MINI 6

GCOS 6 MOD 400/MFS

System Building

SOFTWARE

SOFTWARE

SOFTWARE

SOFTWARE

SOFTWARE

Bull

69 A2 CB23 REV3

Bull

MINI 6

GCOS 6 MOD 400/MFS

System Building

Software

Subject : Description of the procedures necessary to configure a MOD 400/MFS system to meet the characteristics and requirements of the site.

Special Instructions : This revision supersedes 69 A2 CB23 REV2. Due to the large amount of changes, change bars have been omitted.

Software Supported : GCOS 6 MOD 400/MFS Release 2.1. For further information on software supported, see the GCOS 6 MOD 400/MFS Software and Documentation Directory.

Date : March 1982

Bull **CEDOC - CELOG**

Boîte Postale 110 Parc Industriel d'INCARVILLE
27100 Ensemble Urbain du VAUDREUIL - FRANCE

69 A2 CB23 REV3

Dépôt légal
1er trimestre 1982

Printed in France

PREFACE

This manual is intended for the person who is charged with configuring the GCOS 6 MOD 400 system software to meet the characteristics and requirements of his installation site. This person is assumed to be well acquainted with the GCOS 6 MOD 400 Executive and with CII HONEYWELL BULL's Series 60 (Mini 6) minicomputer systems. In addition, this person is assumed to have read the System Concepts manual, and if he intends to build a system supporting communications devices, the Communications Processing manual. The Manual Directory at the beginning of this manual lists the titles and orders numbers of pertinent documents.

This publication describes in detail the procedures necessary to build the system, the system building directives from which the MOD 400 Executive is assembled, and the utility programs used in the building process.

Section I of this manual introduces the subject of system building and briefly describes the contents of the document.

Section II outlines information the system builder must gather prior to configuring a system to his specifications.

Section III describes the three stages of system startup and the purpose of each stage. Section III also describes the operator's and the system's actions during system startup.

Section IV describes how to use the interactive CLM directive generation program, M4_SYSDEF.

The information and specifications in this document are subject to change without notice. This document contains information about CII HONEYWELL BULL products or services that may not be available in all countries. Consult your CII HONEYWELL BULL Marketing Representative.

Section V describes in detail the noncommunications directives that are processed by the Configuration Load Manager -- a system software tool -- as it configures the system to stipulated specifications.

Section VI describes communications-related directives processed by the Configuration Load Manager.

Section VII lists the contents of the system disks on which GCOS 6 MOD 400 and other related software components are available.

Section VIII includes a number of miscellaneous but important technical notes, most of which pertain only to certain installation sites. This section should be read before attempting to use the system.

Appendix A presents an overview of the various types of halts possible during system startup.

Appendix B lists the names and approximate sizes of all system overlays, which may be used as overlays or, where optimum speed is desired for the related function, selectively made permanently resident in memory.

Appendix C contains minimum system guidelines and a list of supported equipment.

Appendix D describes configuration considerations relative to the Power Resumption Facility.

Appendix E deals with configuration information (including linking) pertaining to the Data Entry Facility.

Appendix F describes configuration information for the Remote Batch Facility.

Appendix G presents configuration information relative to the file transmission capability.

Appendix H and Appendix J provide configuration information pertaining to IBM Workstation Facilities.

Appendix K presents configuration information for the Programmable Facility/3271.

Appendix L describes installation and activation of the Listener and Login capability.

Appendix M provides a handy checklist for use with the M4_SYSDEF interactive CLM directive generation program.

Appendix N provides configuration information for the error logging capability.

Appendix P provides configuration information for the Display Formatting and Control software.

Appendix R provides configuration information for the Asynchronous Character Terminal Driver.

After reading this manual, the user should be able to build, the system for initial operation and onsite personnel training, and rebuild the system later on, if necessary, to satisfy the installation's changing needs.

The examples presented in this publication are for illustrative purposes only; they not intended for execution.

For the sake of clarify in examples, user typeins (commands and directives) are distinguished from system typeouts (responses) by arrows, as shown:

```
=>  COMMAND LINE
    SYSTEM RESPONSE
```

The following conventions are used to indicate the relative levels of topic headings used in this manual:

Level 1 (highest)	ALL CAPITALS, UNDERLINED
Level 2	Initial Capitals, Underlined
Level 3	ALL CAPITALS, NOT UNDERLINED
Level 4	Initial Capitals, Not Underlined

Symbols used in this manual are described below:

- Brackets [ ] indicate an optional directive entry.
- Braces { } enclose information from which a choice must be made.
- The character Δ indicates a space.

CONTENTS

FIGURES

TABLES

SECTION I

SYSTEM BUILDING OVERVIEW

System building consists of several discrete operations, each described in different sections of this manual. The purpose of this section is to place these operations into perspective, and to enable you to obtain a general view of the system building process.

CII HONEYWELL BULL delivers a disk volume containing the software you ordered plus an operating MOD 400 Executive. System building consists of specifying your system variables, identifying your peripheral devices and (optional) communications environment and tailoring main memory to suit system and user needs. A "system build," starting with the initially supplied system, consists of the operations described below.

PRELIMINARY CONSIDERATIONS

Before starting to build your system, you must take stock of the hardware in your configuration. You must be familiar with the characteristics of each peripheral and (optional) communications device. You must also be aware of what software packages you wish to build into your system, consulting the appropriate software manuals as necessary. These preliminary considerations are discussed in more detail in Section II.

INITIAL SYSTEM STARTUP

System building is a progressive process and is typically performed in three discrete stages, each of which is initiated by an appropriate type of system startup. To build your new MOD 400 system, you must first start up the initially supplied system on your hardware configuration. The system is designed to start up without difficulties on most hardware configurations.

The three stages of system startup, and the purpose of each stage, are described in Section III. Each stage of system startup involves (1) bootstrapping the system from disk into main memory, (2) execution of the Configuration Load Manager (CLM) which reads a file of CLM directives and causes the system to be configured accordingly, and (3) creation of a system task group. As described in Section III, additional actions are possible at system startup, depending on whether a special file (named START_UP.EC) exists in the initial working directory of the system task group.

SYSTEM BUILDING

System building consists of creating a file containing a series of system building directives which completely specify the characteristics of the system. These characteristics include the hardware options and physical memory present in the central processor, the complement of peripheral and communications devices present in your configuration, and the manner in which memory is to be allocated among system and application tasks. (The system building directives are described in detail in Sections V and VI).

You create this directive file on disk, using either an interactive building program or the standard text editor. Once created, the file contains configuration directives, which when executed, will configure a system that corresponds to the actual installation hardware.

HOW TO USE THIS MANUAL

Sections I, II and III should be read carefully before you begin system building procedures. Sections I and II provide a general description of system building and outline the information a first-time user needs to know before building a specialized system. Section III describes the three stages of system startup that must be followed to create an operating environment.

An interactive CLM directive generation program, called M4_SYSDEF, is provided for your convenience in creating the CLM directive file, and is fully described in Section IV. Using M4_SYSDEF according to directions will result in a syntactically correct file of CLM directives.

Non-communications CLM directives and communications CLM directives, used to specify the system configuration, are described in detail in Section V and VI, respectively. Refer to these sections when creating or modifying your specialized CLM directive file. If your system supports communications devices, you should be familiar with the Communications Processing manual.

Section VII describes the contents of the system release media. Typically, an installation will receive all of its system software (including any separately priced "extensions" that have been ordered) on one type of release medium; that is, on one or two cartridge disks, on one mass storage unit (also called a "storage module"), on one cartridge module disk, or on multiple diskettes.

Section VIII comprises a number of miscellaneous technical notes, most of which pertain only to certain installation sites. It is important that each prospective system builder familiarize himself with the contents of Section VIII and ascertain which of its topics apply to his installation, before he attempts to use the system.

Most of the appendixes in this manual also describe subjects of interest only to certain installation sites. Among these subjects are configuration information regarding the Data Entry Facility-I (DEF-I), the Remote Batch Facility (RBF), and file transmission.

Appendixes H, J, and K present configuration information for the IBM Workstation Facilities and the Programmable Facility/3271. Appendix L describes Listener and Login capabilities.

An M4_SYSDEF checklist is supplied for your convenience in Appendix M. Use this list as a guide when defining your system's resources prior to invoking the M4_SYSDEF program.

Appendix N describes how to configure the error logging capability so your installation can monitor memory and device performance.

Appendix P describes configuration procedures for the Display Formatting and Control software. This software provides the forms processing capability for your installation.

In general, this manual does not include conceptual information; instead, where necessary, see the System Concepts manual and/or a manual specific to the desired subject area. Commands mentioned in this manual are described in the Commands manual. Those operator commands that are not available to the general user community, for example the CSD (change system directory) operator command, are described in the Operator's Guide. System macro calls are described in the System Service Macro Calls manual.

SECTION II

PRELIMINARY CONSIDERATIONS

Before you actually begin to build your system, you should first compile the information you'll need to configure the system to your specifications. The categories of information required are broken into hardware, software, and communications considerations and are discussed further below. Figure 2-1 is a flowchart of the recommended procedure for gathering information.

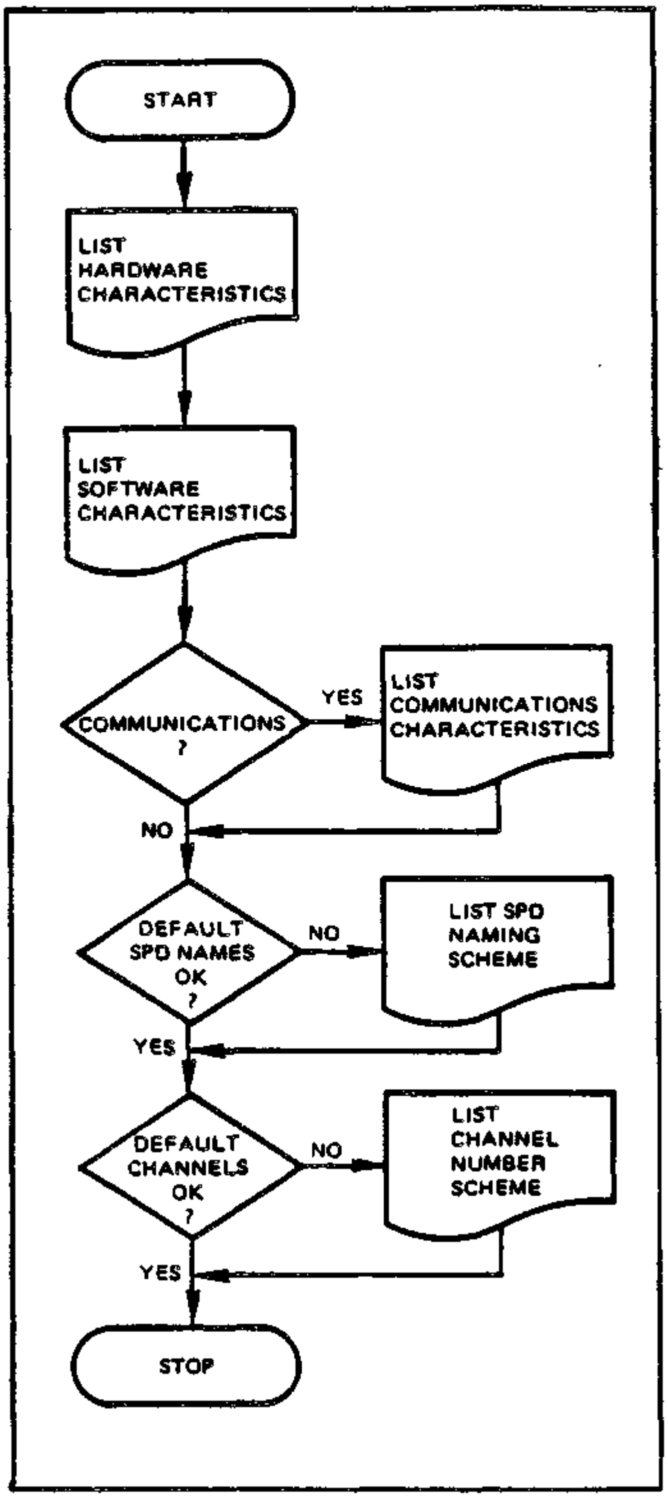


Figure 2-1. Preliminary Considerations Flowchart

HARDWARE CHARACTERISTICS

You must have available the hardware characteristics of the devices in your configuration. You can obtain this information by consulting the appropriate hardware reference manuals as necessary.

Specifically, for the central processor, you must know:

- The model number
- Whether the central processor is a commercial model
- Whether the central processor includes the Scientific Instruction Processor
- The size of main memory
- The number of multiple/dual-line communications processors (MLCPs/DLCPs)

For peripheral devices, you must know:

- The maximum number of devices that you want to configure (regardless of how many are on the bus)
- How many of each type of peripheral device you have in your configuration, identified by marketing identifier (see Appendix C)
- For terminals, you must know the transmission modes, line speeds, desired I/O characteristics, and modem types
- For disk devices, you must know whether each cartridge disk or cartridge module disk has a fixed and removable platter; you must also know the storage capacity of each device
- For tape drives, you must know the recording density and number of tracks.

SOFTWARE CHARACTERISTICS

Once you have compiled a list of hardware characteristics, you should proceed to gather information on the software characteristics of your configuration. You must determine what software subsystems your installation will use. Equipment requirements and characteristics of individual software packages are available in the appropriate software reference manuals.

Specifically, you must know:

- The text strings you wish to use as the system and installation identification of your system

- The anticipated maximum number of concurrent users

- The largest logical file number and logical resource number available to users

- The anticipated number of 512-word system overlay areas

- Whether or not the system will include a batch memory pool (and its size)

- Whether the system will include the commercial or scientific instruction simulators

- Whether or not the system will incorporate record locking or Display Formatting and Control software

- A coherent assignment scheme for symbolic peripheral device names, channel numbers, communications priority levels, and logical resource numbers (see the System Concepts manual)

- The number of communications line protocols (the recommended maximum is two per MLCP/DLCP).

COMMUNICATIONS CHARACTERISTICS

If you intend to include communications devices in your configuration, you must read the Communications Processing manual first, and be thoroughly familiar with the hardware and software characteristics of your communications devices.

Specifically, you must know:

- Which protocols you need

- For polled VIP devices (VIP7700 or VIP7804), a poll address scheme, device type, which stations are control/tributary stations, and characteristics of any receive-only printers

- For H3270 Host Links, a poll address scheme, a select address scheme, and a communications priority level for each line

- A priority level starting scheme

- The number of channel control programs per MLCP/DLCP (the recommended maximum is two)

- For asynchronous terminals, the communications priority level and device type for each

- For BSC lines, the communications priority level and the number of primary and/or secondary lines

- For ACU channels, the lrn of the station associated with each ACU and the list of telephone numbers associated with each data communications channel

- For stations driven by a user-written line protocol handler, whether the line is full- or half-duplex, and the lph-specific word for each station

- For teleprinter compatible devices, the communications priority level and modem type for each device

- Which communications stations will be accessible through the file system

- The polling scheme on a line serviced by the synchronous terminal driver line protocol handler

- For polled or nonpolled VIP terminals, the lrn, level, and channel

- For IBM workstations on a line serviced by the HASP line protocol handler, the lrn, level, and channel

- For stations on a line serviced by the Remote Computer Interface line protocol handler, the lrn, level, and channel.

This is only meant to be a representative sampling of some of the software and hardware characteristics of which you must be aware. For a full description of each communications protocol, see Section VI.

SECTION III

SYSTEM STARTUP

System startup consists of the following steps:

1. The system is bootstrapped into main memory as the result of a procedure performed at the central processor's control panel.

2. The Configuration Load Manager (CLM), a system component, reads a file of CLM directives, which causes the system to be configured according to the contents of the directives.

3. A system task group ($S) is created after system configuration is completed. If the initial working directory for the system task group contains a file named START_UP.EC, the operator commands in that file are executed to further establish the environment that is desired at the end of system startup. (For example, the START_UP.EC file may contain operator commands that spawn additional task groups.)

Thus, system startup establishes an operating environment and brings the system to a "ready" state, at which point system operations may begin.

There are three stages of system startup.

1. Stage 1 Initial System Startup, which is performed at system installation.(1) This type of startup is used to establish a limited environment that will allow you to use the interactive CLM directive generation program M4_SYSDEF or the Editor to create your own file of CLM directives. During a stage 1 system startup, the Configuration Load Manager reads one of two supplied files of CLM directives that define a limited environment. The file of CLM directives read by the Configuration Load Manager depends on which of the following situations exist at system startup:

   a. Bootstrap volume is a cartridge disk, cartridge module disk, diskette, or mass storage unit; one or more terminals are connected to an MDC.

(1)
A stage 1 system startup can also be used at a later time, in the event that an error in a user-created file of CLM directives prevents a successful stage 2 or stage 3 system startup. See "Bootstrap Options" later in this section.

b. bootstrap volume is a cartridge disk, cartridge module disk, diskette, or mass storage unit; no terminal is connected to an MDC, but one or more terminals are connected to an MLCP/DLCP.

In a stage 1 system startup, the initial working directory for the system task group is SID. This working directory contains a START_UP.EC file that causes a second task group ($S) to be spawned. (1)

2. Stage 2 Intermediate System Startup. This stage of system startup is used after you have created your own file of CLM directives, i.e., a file named CLM_USER (under SID), which describes your own installed hardware configuration and other details of the environment you wish to use for application development. (You will have created CLM_USER by using M4_SYSDEF or the Editor in the limited environment established as the result of a stage 1 system startup.)

   In a stage 2 system startup, your tailored CLM_USER file must exist under directory SID on the bootstrap volume. The Configuration Load Manager will read this file instead of the one selected during a stage 1 system startup.

   In a stage 2 system startup, the initial working directory for the system task group is the root directory of the bootstrap disk. This working directory contains a START_UP.EC file that causes a second task group ($H) to be spawned.

   Following a stage 2 system startup, application development can proceed in an environment you have defined to match your own resources and needs. When you have reached a point where you wish to define a fully specialized system refined to precisely match your requirements for a standard operating environment, you can use the M4_SYSDEF program in Rebuild mode or the Editor to modify the contents of CLM_USER so that they describe your final system. In addition, you can modify (or delete) the START_UP.EC file of operator commands immediately subordinate to the root directory of the bootstrap volume. Note that "refining" takes place over a period of time, as you gain experience with the system.

3. Stage 3 Fully Specialized System Startup. This stage of system startup is used after you have made final adjustments to the CLM directives in the CLM_USER file and any necessary modifications to the START_UP.EC file. This stage of system startup is used to achieve your final configuration, one that reflects your total hardware environment, optimized use of central processor main memory, and all other characteristics of your standard system.

(1)
>SID>START_UP.EC is reserved for use during a stage 1 system startup.

Once a satisfactory CLM_USER file and START_UP.EC file (if any) have been established, this type of system startup can be performed routinely, even by noncomputer personnel.

Figure 3-1 summarizes the three stages of system startup.

STAGE 1 INITIAL SYSTEM STARTUP	WHEN PERFORMED
ONE OF TWO CII HONEYWELL BULL SUPPLIED FILES OF CLM DIRECTIVES IS USEDTO DEFINE A LIMITED ENVIRONMENT IN WHICH M4_SYSDEF OR THE EDITOR CAN BE RUN TO CREATE AN INSTALLATION-SPECIFIC FILE OR CLM DIRECTIVES. (>SID>START_UP.EC FILE IS EXECUTED FOLLOWING CONFIGURATION.)	ONCE AT SYSTEM INSTALLATION. THEREAFTER, IF NECESSARY, TO PERMIT CORRECTION OF ERRORS IN USER-CREATED FILE OF CLM DIRECTIVES.
STAGE 2 INTERMEDIATE SYSTEM STARTUP	**WHEN PERFORMED**
USER-CREATED FILE OF CLM DIRECTIVES, CLM USER, IS USED TO DEFINE AN ENVIRONMENT IN WHICH APPLICATION DEVELOPMENT CAN BE PERFORMED. (^ROOT DIRECTORY>START UP.EC FILE IS EXECUTED FOLLOWING CONFIGURATION.)	ONCE AT SYSTEM INSTALLATION, AFTER CREATION OF CLM_USER FOLLOWING A STAGE 1 SYSTEM STARTUP. THEREAFTER, AS NECESSARY.
STAGE 3 FULLY SPECIALIZED SYSTEM STARTUP	
FULLY SPECIALIZED FILE OF CLM DIRECTIVES (ADJUSTED CLM USER) IS USED TO DEFINE THE STANDARD ENVIRONMENT FOR FULL SYSTEM OPERATIONS. (IF PRESENT, ^ROOT DIRECTORY>START_UP.EC FILE IS EXECUTED FOLLOWING CONFIGURATION.)	WHEN STANDARD SYSTEM OPERATIONS ARE FEASIBLE, AFTER NECESSARY APPLICATIONS HAVE BEEN CREATED FOLLOWING A STAGE 2 SYSTEM STARTUP. A STAGE 3 SYSTEM STARUP CAN BE PERFORMED ROUTINELY AFTER SATISFACTORY CLM USER HAS BEEN ESTABLISHED.

Figure 3-1. Stages of System Startup

STAGE I SYSTEM STARTUP (INITIAL SYSTEM STARTUP)

A stage I system startup causes the system to be bootstrapped, the Configuration Load Manager to configure the system using an existing the of CLM directives, and the operator commands in the >SID>START_UP.EC file to be executed. The result is a limited system environment that allows you to use the interactive CLM directive generation program M4_SYSDEF or the Editor to create your own file of CLM directives.

The environment achieved by a stage I system startup is as follows:

- A removable cartridge disk, cartridge module disk, a mass storage unit, or a diskette (depending on the device from which the system was bootstrapped).

- Either an MDC-connected operator termial or an MLCP.DLCP connected operator terminal. (The former is used, if present).

- One online memory pool and one online task group ($H), in addition to the system memory pool and system task group ($S).

A stage I system startup consists of the following steps:

1. Set the central processor control panel POWER switch to ON and turn on the power for all devices.

2. Mount the volume containing the CII HONEYWELL BULL-supplied system software on the bootstrap device (cartridge disk, cartridge module disk, mass storage unit, or diskette).

3. Perform the following bootstrap procedure at the central processor control pannel:

   a. Press Stop.

   b. Press CLeaR.

   c. Press Load.

   d. Press Execute.

   e. Wait for the TRAFFIC light to turn off. (If the bootstrap channel is not 0400 and/or if any of the bootstrap options described in Table 3-2 are to be used, see "Operator's Startup Actions," later in this section).

   f. Press Ready.

   g. Press Execute.

   The system is bootstrapped into main memory. Depending on the terminal configuration, one of the two supplied files of CLM directives under directory SID is read by the Configuration Load Manager and the system is configured accordingly.

(1) CLM_MDC is used if the bootstrap volume is a cartridge disk, cartridge module disk, mass storage unit, or diskette: one or more terminals are connected to an MLCP/DLCP.

(2) CLM_MCP is used if the bootstrap volume is a cartridge disk, cartridge module disk, mass storage unit, or diskette: one or more terminals are connected to an MLCP/DLCP.

If your configuration includes both MDC- and MLCP/DLCP connected terminals, the system uses CLM_MDC.

A system task group ($S) is spawned. The operator commands in >SID>START_UP.EC are executed, and a user task group ($H) is spawned.

If your configuration includes a terminal connected to an MDC, system startup is now complete. Otherwise, your configuration must include a terminal connected to an MLCP/DLCP and you must perform steps h, i, and j (full control panel required).

h. Wait for the TRAFFIC light to turn off. (If the terminal's line speed should be other than 0110 bits per second- the default value -enter the appropriate line speed in the RI-register. See the paragraph below and Table 3-1).

i. Press Ready.

j. Press Execute.

The operator terminal is selected as follows:

1. If there are one or more terminals connected to an MDC, the one with the lowest channel number is selected.

2. If there are no terminals connected to an MDC, there must be one or more terminals connected to an MLCP/DLCP. In this case, the MLCP/DLCP-connected terminal with the highest channel number is the operator terminal.

   Since a communications terminal can run at different line speeds, you must determine the connect line speed of the terminal with the highest channel number and then, in the procedure described above, when the TRAFFIC light turns off in step 3h, you enter the correct line speed in the RI-register. The default value is 0110 bits per second.

Completion of system startup is signalled by the message shown in Figure 3-2. If your terminal is connected to an MDC, don't press any terminal keys until after the system startup message appears. If you do, you can recover by pressing the BREAK key.

```
($S)GCOS6 MOD400 - arrr -mm/dd/hhnn

($H)$H Group ready!
```

NOTE: See the accomoanying text for an explanation of the variables in the first line.

Figure 3-2. System Startup Typeout at Operator Terminal

The system startup message, shown in Figure 3-2, is interpreted as follows:

. ($S)GCOS6 MOD400 - arrr - mm/dd/hhnn

Indicates the completion of system startup. A is S for a SAF system, L for a LAF system. rrr indicates the release number of the system Executive. mm/dd/hhnn indicates the date/time when the Executive was linked (month, day, hours, minutes).

. ($S)$H Group ready!

Indicates that a user taks group, which has the command processor as its lead task, is ready for the entry of commands. This task group has been spawned by a command in >SID>START_UP.EC. The typeout is the result of a command in the supplied >HIS>START_UP.EC file.

Table 3-1. TTY Terminal Line Speeds

Type 2108 Communications-Pac	Type 2100, 2110, or 2118 Communications-Pac
Speed (Entered in RI-Register)	Speed (Entered in RI-Register)
0050	0050
0075	0075
0110	0110
0134	0134
0150	0150
0300	0200
0600	0300
0900	0600
1200	1050
1800	1200
2400	1800
3600	2000
4800	2400
7200	4800
9600	9600

NOTE: A speed of 0134 represents 134.5 bits/second.

STAGE 2 SYSTEM STARTUP (INTERMEDIATE SYSTEM STARTUP)

A stage 2 system startup can be performed only after you have created your own file of CLM directives, named CLM_USER under directory SID on the bootstrap volume. The contents of CLM_USER should described the environment you wish to use for application development.

Creating CLM_USER

The recommended approach for creation of CLM_USER is shown below.

1. Following a stage 1 system startup, invoke M4_SYSDEF to run under task group $H. Although you can create the CLM_USER file directly by using the Editor, CII HONEYWELL BULL recommends that you use M4_SYSDEF, the interactive system building program, even if you have build the system before. See Section IV for a complete description of how to use M4_SYSDEF.

2. After creating a CLM_USER file using M4_SYSDEF and placing it immediately subordinate to directory SID on your bootstrap volume, continue with the steps necessary for a stage 2 system startup.

If you wish to create your own CLM_USER file without using M4_SYSDEF, or if you wish to modify the existing CLM file to reflect changes in your installation's configuration, follow steps 1 through 7 below to use the Editor.

USING THE EDITOR

1. Following a stage 1 system startup, invoke the Editor to run uner task group $H.

   You may be required to change the default task group of the operator's terminal from the $S (System) task group to the $H task group. Enter the following lines at the operator's terminal immediately following system startup:

C :$H:	(transfers default task group of the operator's terminal from $S to $H task group).
RDN	(The ready on (RDN) command causes a prompt message to appear at your terminal. This command is for convenience only and is not required).

   If you have entered RDN, the system will respond with the prompt message:

   ($H)RDY:

Note that no prompt message appears unless you specified RDN.

You are not ready to invoke the Editor, as described in the Commands manual and the Program Preparation manual. Sample Editor work sessions are provided in the Programmer's Guide.

2. If you are modifying an existing CLM file, use the Editor's read directive to read the file's contents into the Editor's "current buffer". If you are creating a CLM file, go to step 5.

3. Use the Editor's print directive to print out the entire contents of the current buffer.

4. Inspect the printout and identify the differences between the contents of the current buffer and your desired file CLM directives.

5. Use the Editor's insert, change, append, and delete directives to modify the contents of the current buffer so as to define the CLM directives that describe your application development configuration.

6. Use the Editor's print directive to print out the entire contents of the current buffer. Ensure that the contents accurately reflect your desired CLM directives. (If any corrections are necessary, repeat steps 5 and 6).

7. Use the Editor's write directive to write the entire contents of the current buffer to >SID>CLM_USER. (You may give your CLM file any name you wish, although it is recommended that you call it CLM_USER. Note that the Configuration Load Manager must be able to locate a file named CLM_USER at system startup.

Stage 2 Startup

One you have defined the desired contents if >SID>CLM_USER, you can perform a stage 2 system startup by taking the following steps:

1. Ensure that the central processor control panel's POWER switch is ON and turn on power to all equipment.

2. Ensure that the volume containing the CII HONEYWELL BULL-supplied system software is mounted on the bootstrap device.

3. Perform the following bootstrap procedure at the central processor control panel:

   a. Press Stop.

   b. Press CLeaR.

      Wait for CHECK light to go out.

c. Press Load.

d. Press Execute.

e. Wait for the TRAFFIC light to turn off. (If the bootstrap channel is not 0400 and/or if any of the bootstrap options describes in Table 3-2 are to be used, see "Operator's Startup Actions," later in this section).

f. Press Ready.

g. Press Execute.

   The system is bootstrapped into main memory. The CLM_USER file in directory SID is read by the Configuration Load Manager and the system is configured accordingly.

   A system task group ($S) is spawned. The operator commands in the CII HONEYWELL BULL-supplied ^root_directory >START_UP.EC file are executed, causing a user task group ($H) to be spawned.

System startup is complete. The system startup message is issued at the operator terminal, which, in the case of a stage 2 system startup, is the terminal associated with logical resource number (lrn) 0 in a CLM directive. (If the operator terminal is connected to an MLCP/DLCP, the terminal's line speed and other communications characteristics are defined in a CLM directive).

If a CLM directive in CLM_USER contains an error caused by incorrect syntax or if the CLM directive is an invalid duplicate, an error message will appear at the operator terminal (if possible). See "Configuration Load Manager Error Reporting" at the end of this section. If an uncorrectable error prevents sucessful completion of a stage 2 system startup, it is possible, by means of the appropriate bootstrap option (see Table 3-2), to bypass the CLM_USER file and perform a stage 1 system startup using one of the two supplied files of CLM directives. This technique allows you to re-establish a limited system configuration in which you can use the Editor or M4_SYSDEF to correct CLM_USER before reattempting a stage 2 system startup.

Note also that the supplied START_UP.EC file directly subordinate to the bootstrap volume's root directory is normally not modified prior to a stage 2 system startup.

Once a satisfactory stage 2 system startup has been achieved, you can use the resulting system for application development or as a means of gaining experience in the use of the system software.

STAGE 3 SYSTEM STARTUP (FULLY-SPECIALIZED SYSTEM STARTUP)

A stage 3 system startup is very similar to a stage 2 system startup. The principal difference is that application development performed following a stage 2 system startup provides you with information you may wish to use in refining the CLM directives in your CLM_USER file. For example, you may wish to change certain MEMPOOL directives and/or cause certain system overlays to be made permanently resident. Moreover, you may wish to describe additional peripheral devices that were not previously identified in CLM_USER because they were not used during application development.

Thus, prior to performing a stage 3 system startup, you will probably have used the Editor or M4_SYSDEF to update the contents of CLM_USER. Additionally, you must decide whether to retain, modify, or delete the START_UP.EC file immediately subordinate to the bootstrap volume's root directory (i.e., the START_UP.EC file used in a stage 2 system startup). If you wish to retain this START_UP.EC file in its original condition (but not use it during a stage 3 system startup), you may use the RENAME command.

The steps performed in a stage 3 system startup are identical to those used for a stage 2 system startup except that the ^root_directoty>START_UP.EC file may have been modified or deleted. If this file exists at the time of a stage 3 system startup, all operator commands therein will be executed.

SUMMARY OF STARTUP PROCEDURE

The following paragraphs and Figure 3-3 summarize the startup procedure.

Every time the system is bootstrapped, a software component ascertains whether an initially-supplied or a user-created file of CLM directives is to be read by the Configuration Load Manager. If an initially-supplied file is used, a software algorithm also ascertains whether the operator terminal is to be an MDC-connected terminal or an MLCP/DLCP-connected terminal. If a user-created CLM file is used, a CLM directive dictates which device is to be the operator terminal.

During a system startup in which an initially-supplied CLM file is used (i.e., a stage 1 system startup), operator commands in the START_UP.EC file in directory >SID will be executed. During a stage 2 system startup (user-created file of CLM directives used), operator commands in the supplied START_UP.EC file under the bootstrap volume's root directory will b executed. During a stage 3 system startup (refined user-created file of CLM directives used), operator commands in the START_UP.EC file under the bootstrap volume's root directory will be executed if such a file is present either in its original form or as modified by the user.

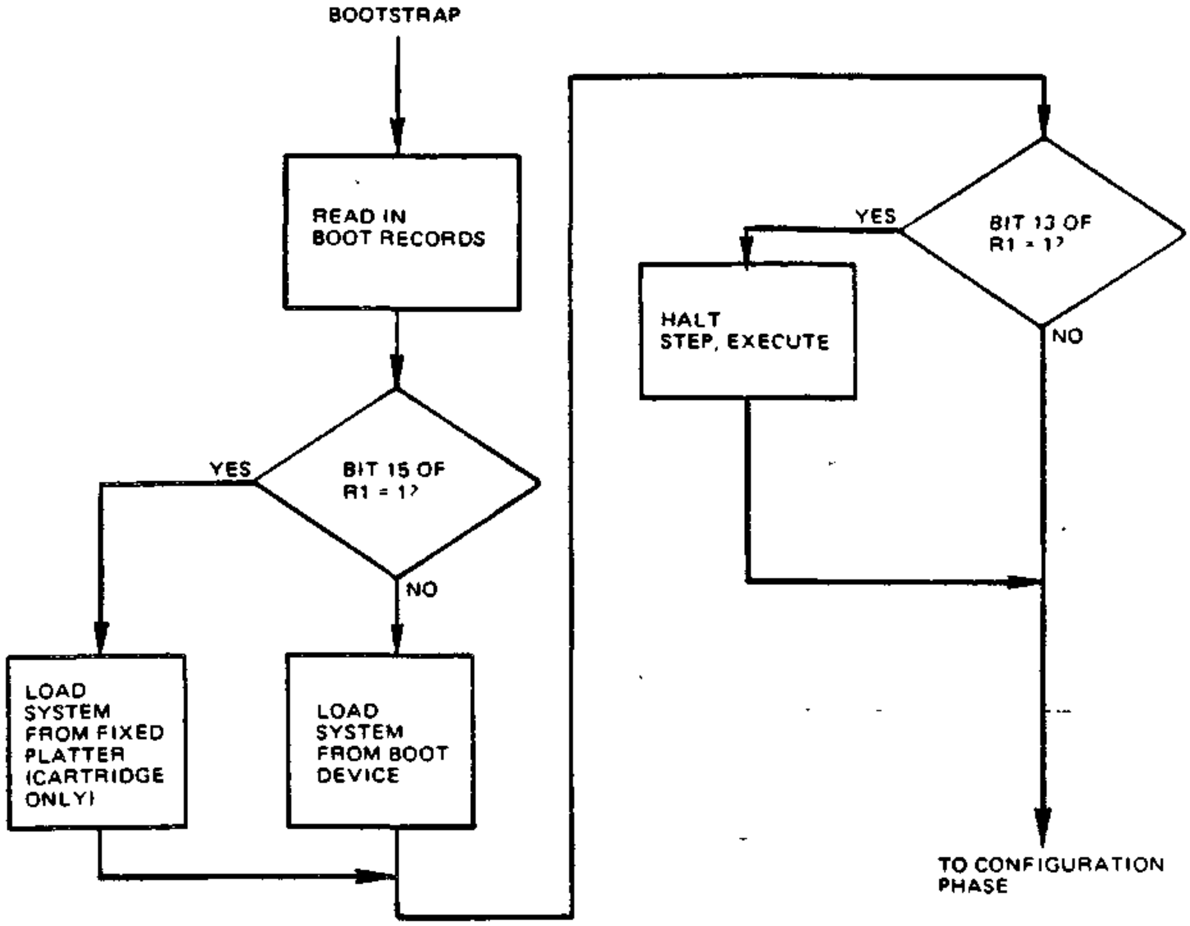


Figure 3-3. Flowchart of System Startup Process

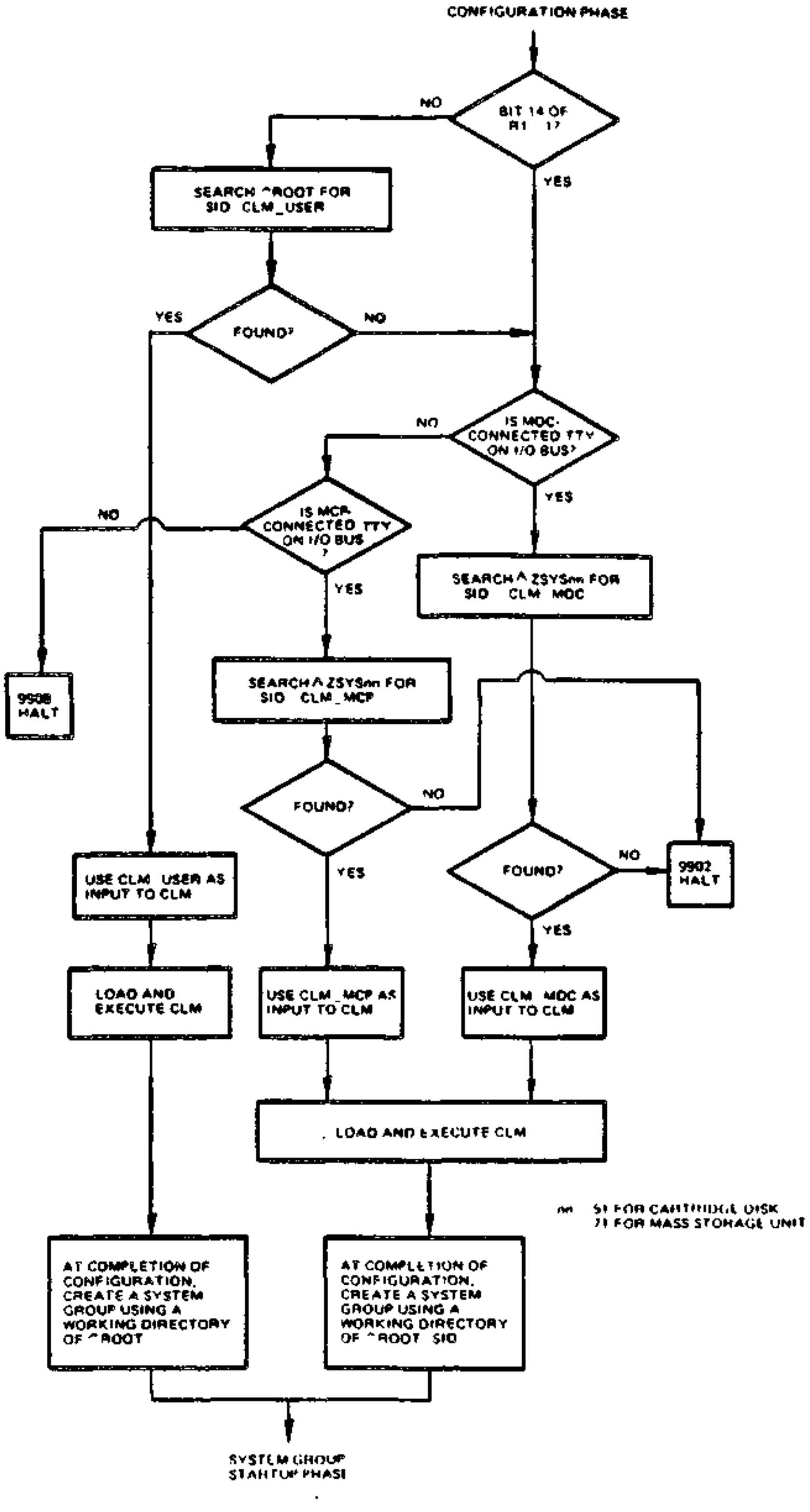


Figure 3-3 (Cont). Flowchart of System Startup Process

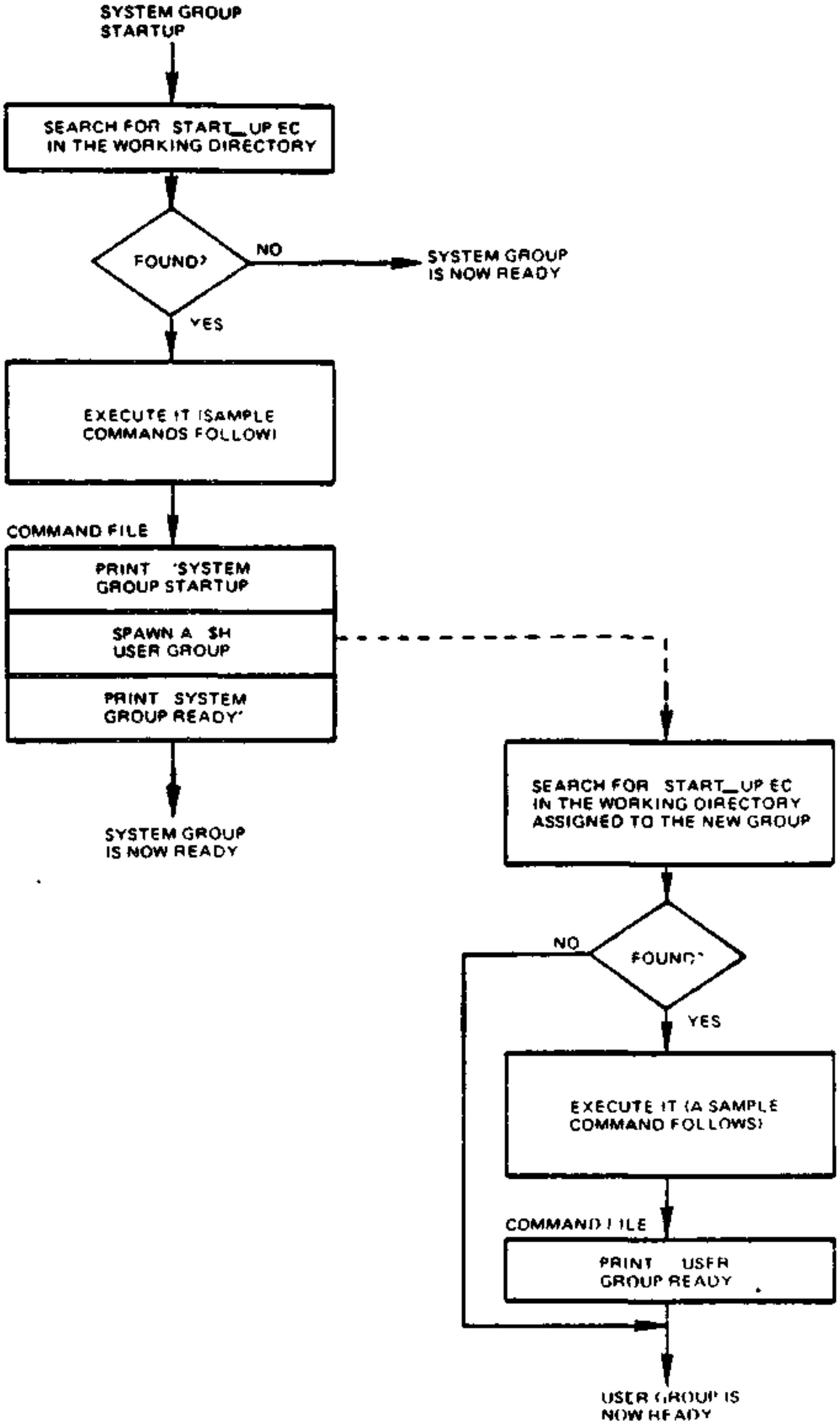


Figure 3-3 (Cont). Flowchart of System Startup Process

Operator's Startup Actions

The operator's actions at system startup are as follows:

1. Set the central processor control panel POWER switch to ON. Turn on power for all devices.

2. Ensure that the volume containing the CII HONEYWELL BULL-supplied system software is mounted on the bootstrap device.

3. Perform the following bootstrap procedure at the central processor control panel:

   a. Press Stop.

   b. Press CLeaR.

      Wait for CHECK light to go out.

   c. Press Load.

   d. Press Execute.

   e. Wait for the TRAFFIC light to turn off.

      If the bootstrap channel is not 0400 and/or if any of the bootstrap options shown in Table 3-2 are to be used, perform the following steps:

      (1) Press Stop.

      (2) Press Select.

      (3) Press D1 on the hexadecimal keypad, so that D1 appears in the LOCATION indicators.

      (4) Press Change.

      (5) Press appropriate keys on the hexadecimal keypad so that the CONTENTS indicators show the desired bootstrap channel and options.

      (6) Press Run.

   f. Press Execute.

      The system is bootstrapped into main memory. (If the bootstrap halt option was specified in step e(5) above, the central processor halts, with D1 equal to what was entered in step e(5), as soon as the system has been bootstrapped; after performing any desired actions, again press Execute to continue. If Stop, Run, or Write was invoked in these actions, press Run and Execute to continue).

(In a stage 1 system startup with an MLCP/DLCP-connected operator terminal, wait for the TRAFFIC light to turn off; then ensure that the RI-register contains the proper line speed for the operator terminal and press Execute to continue).

Table 3-2. Bootstrap Options

Bits in Low-Order Digit of 4-Digit Hexadecimal Bootstrap channel Number	Meaning if Bit Set ON[a]
15 (low-order bit)	After the initial bootstrap record is read from the removable platter of the cartridge disk whose channel number is contained in bits 0 through 9, the remaining records are read from the fixed platter.
14	Ignore CLM_USER file if it exists under directory SID. Instead, use the supplied file of CLM directives appropriate to the bootstrap device and terminal configuration (i.e., CLM_MDC or CLM_MCP).
13	The central processor halts after the system is bootstrapped and before execution of the Executive begins, and before execution of CLM. (To continue after the halt, press Execute).

[a] These bits may be set ON in any desired combination. Thus, the value of the low-order digit of the 4-digit hexadecimal bootstrap channel number may range from 0 (no options) to 7 (all options).

System Startup Actions

System startup actions continues without further operator intervention.

1. The Configuration Load Manager reads CLM directives from the appropriate file:

   a. If CLM_USER does not exist in directory SID on the bootstrap volume, one of the two CII HONEYWELL BULL-supplied files is used.

(1) CLM_MDC is used if the bootstrap volume is a cartridge disk, cartridge module disk, mass storage unit, or diskette, and one or more terminals are connected to an MDC. (The terminal with the lowest channel number will be used as the operator terminal).

(2) CLM_MCP is used if the bootstrap volume is a cartridge disk, cartridge module disk, mass storage unit, or diskette, and one or more terminals are connected to an MLCP/DLCP. (The terminal with the highest channel number will be used as the operator terminal).

b. If CLM_USER does exist under directory SID on the bootstrap volume, it will be used unless bit 14 of the bootstrap channel is set ON, in which case one of the system actions described under a (above) is followed.

2. The system task group is created as if the following command has been executed, without user intervention:

SG	(spawn group command)
$S	(task_group_id)
OPERATOR.SYSTEM.OPR	(user_id)
highest priority level + 2 in CLM directive	(base_priority_level)
!CONSOLE	(in_path, also out_path by default)
-ECL	(lead task)
-LRN highest lrn in CLM directive	(highest lrn)
-LFN 15	(highest lfn)
-POOL system pool	(memory poll)
-WD ^root_directory if CLM directives were read from CLM_USER; SID if CLM directives were read from CLM_MDC or CLM_MCP.	(initial working directory)

3. A system-supplied START_UP.EC file is in directory SID: its operator commands are executed when a stage 1 system startup is performed (i.e., when CLM directives are read from CLM_MDC or CLM_MCP). Another system-supplied START_UP.EC file is immediately subordinate to the root directory of the bootstrap volume: its operator commands are executed when a stage 2 system startup is performed.

Both START_UP.EC files cause a user task group ($H) to be spawned. The working directory for $H is HIS, which in turn contains a system-supplied START_UP.EC file that performs a CWD (change working directory) to ^root_directory.

Regardless of which file of CLM directives is used, the working directory for the system task group is always ^root_directory at the <u>end</u> of system startup.

4. The system startup typeout (see Figure 3-2) is issued at the operator terminal.

When system startup is complete, as signified by the system startup typeout, operator commands may be entered to the system.

CONFIGURATION LOAD MANAGER ERROR REPORTING

If, when the Configuration Load Manager is reading CLM directives from CLM_USER, it detects that a directive contains a syntax error or is an invalid duplicate of another CLM directive, it issues (if possible) a 2-line error message at the operator terminal. The message has the format shown below:

($S) (13yyzz) hh

secondary message indicating the CLM directive causing the error

yy - code identifying the system component that has detected the error

zz - error code (see the <u>System Messages</u> manual)

hh - priority level of the task group executing the Configuration Load Manager

You can (if appropriate) bypass the error by typing * (asterisk) at the operator terminal and pressing RETURN. Otherwise, you must enter a correct CLM directive through the operator terminal and press RETURN. The remainder of the CLM directives in CLM_USER will then be processed.

An error message cannot be issued if an operator terminal is not available because (1) the CLM DEVICE directive for an MDC-connected operator terminal has been omitted or (2) the operator terminal is connected to an MLCP/DLCP in which case it is not operational until system startup is complete or (3) the error occurred before the DEVICE directive defining the console was read. (For this reason, it is important to place the directive early in the CLM_USER file). In these cases, errors are reported through the hardware registers shown below, which are accessible through the central processor control panel. Refer to Section V in the <u>Minicomputer Systems Handbook</u> (CC71) for procedures to follow in reading out registers and memory locations.

R1-register contains yyzz (as described above)

R2-register Zero or a 1-word secondary message in hexadecimal notation.

R4-register zero or the address of the first word of a secondary message in ASCII notation (the left byte of this word is a control byte)

B3-register zero or the address of the first word of the CLM directive causing the error provided the error occurred before the QUIT directive.

As described earlier in this section, in the event of an uncorrectable error in CLM_USER, it is possible to return to a stage I system startup (using the appropriate bootstrap option) and re-create a limited system environment in which M4_SYSDEF or the Editor can be used to correct the error in CLM_USER.

See Appendix A for information about other halts that are possible during system startup.

SECTION IV

USING M4_SYSDEF

M4_SYSDEF, an interactive program that generates a CLM directive file, is provided for your convenience in creating the CLM_USER file, and is fully described in this section.

The system building CLM directives themselves are described in detail in Sections V and VI. Note that some of the directives - and thus some of the MOD 400 software - are not supported by M4_SYSDEF. See "M4_SYSDEF Operating Considerations", later in this section.

INTERACTIVE SYSTEM BUILDING PROGRAM

The interactive CLM directive generation program M4_SYSDEF is designed to help you create the CLM directive file CLM_USER. It converts information you give it, in response to its prompts, into the CLM directive file. M4_SYSDEF is designed to produce a directive file adequate for the average user and which will always result in an operate MOD 400 Executive.

Although you can create the CLM directive file directly, using the Editor, CII HONEYWELL BULL recommends that you use M4_SYSDEF, even if you have built the system before, because M4_SYSDEF:

- Is interactive; it prevents you from forgetting any necessary information.
- Tests each of your responses for validity.
- Includes online explanatations of each of its pompt messages.
- Ensures that a complete, properly ordered directive file is produced.
- Allows you to interactively add, delete, or modify information you specify at any point in the program.
- Performs all calculations of memory required by your Executive; the resulting memory sizes should never be less than your requirements.
- Builds in all hardware devices supported by CII HONEYWELL BULL.

- Generates, at your option, symbolic peripheral device names for all devices in your configuration.

- Generates, at your option, channel number assignments for all noncommunications devices in your configuration.

- Produces, at your option, a listing of the directive file at your terminal, before generating the final output file.

- Automatically generates a file containing configuration information that enables you to verify the success of certain aspects of the system building process.

- At your option, automatically configures all asynchronous communications devices (TTY and ATD) as being reconnectable in the event of a power failure or line drop.

Appendix M contains a checklist for your convenience in gathering the information M4_SYSDEF requires. Appendix C lists the hardware available under MOD 400.

Required Resources

M4_SYSDEF requires 10K words of memory to operate. It can run in the MOD 400 minimum system.

M4_SYSDEF Operating Considerations

You should consider the following when executing M4_SYSDEF:

1. If you allow M4_SYSDEF to dynamically assign channel numbers to your devices, you will only need to supply channel numbers for MLCP/DLCP - connected (communications) devices. If you do not choose this option, you will be asked to explicitly specify channel numbers for all devices in your configuration. (See Section V for information on dynamically assigned channel assignments for non-communications devices).

2. If you choose to let M4_SYSDEF supply the channel numbers for all non-communications devices:

   - M4_SYSDEF will assign the channel to be used for the operator terminal. If the operator terminal is to be an MDC-connected terminal, the program uses the device with the numerically lowest channel number as the operator terminal. If the operator terminal is to be an MLCP/DLCP-connected terminal, the program uses the device with the numerically highest channel number as the operator terminal.

   - M4_SYSDEF will ask you to specify the type of device you plan to use as the boot device.

3. If you do not choose to supply symbolic peripheral device (SPD) names, M4_SYSDEF automatically generates a unique SPD name for every device configured. If you choose to provide your own SPD names, M4_SYSDEF will ask you to specify them as needed.

   CII HONEYWELL BULL recommends that you use the following symbolic peripheral device (SPD) name conventions if you wish to provide your own SPD names:

Symbolic Peripheral Device (SPD) Name	Device
LPTnn	Line Printer
SPTnn	Serial Printer
CONnn	MDC-connected Terminal
CDRnn	Card Reader
CRPnn	Card Reader/Punch
DSKnn	Diskette
RCDnn	Removable Cartridge Disk
FCDnn	Fixed Cartridge Disk
FCMnn	Fixed Cartridge Module Disk
RCMnn	Removable Cartridge Module Disk
MSMnn	Mass Strorage Unit
MT9nn	Magnetic Tape (9-track)

4. CII HONEYWELL BULL recommends configuring Distributed Systems Facilities software e.g., File Transmission Facility HASP Workstation Facility, etc. <u>after</u> you have built your system the first time. This software is described in this manual's appendixes.

5. M4_SYSDEF doesn't support the following:

   - Configuration of dual-purpose operator terminal
   - User-written line protocol handler (LPHn)
   - AutoCall Unit (ACU)
   - 7-track tape devices
   - Device or memory error logging

6. The program converts all lowercase characters in your responses to uppercase characters, unless you enclose the lowercase characters in apostrophes (').

7. When you enter channel number assignments, the program verifies that:

   - The channel number is in the proper format (i.e., that it contains exactly four hexadecimal characters and ends with either "00" or "80")

   - The channel number isn't already assigned to another device

. The channel number isn't less than 0400 or greater than FF80

Since there is no requirement that system building takes place upon the target machine, or that any specified device actually be attached to the machine when M4-SYSDEF is executed, the program doesn't verify whether the specified channel is actually connected to a device whose type matches the one in question. Therefore, if you enter invalid channel number assignments, the directive file will be in error.

MODIFYING YOUR CLM DIRECTIVE FILE

If you make mistaskes in specifying your directives when running the M4_SYSDEF program, or, if you wish to change, add, or delete information, you have several options for making modifications:

- You can interrupt the program at any point to modify any portion of the CLM file. To do this, enter C BSH, and following the BREAK message, enter the program interrupt (PI) command. (It is assumed that you are entering the M4_SYSDEF program from the operator's terminal. If you are not at the operator's terminal, press the BREAK key, and following the BREAK message, enter the PI command). The M4_SYSDEF program will respond by asking you which class of device or system parameter you wish to modify. Depending on your reply, the program will ask you only those questions in the M4_SYSDEF dialog that pertain to the specific device or system parameter that you wish to modify.

- You can finish running the program and do one of the following:

  - Invoke M4_SYSDEF again, this time specifying the pathname of your restart file. The M4_SYSDEF program will enter rebuild mode and will ask you to specify the class of devices, or system parameter that you wish to modify. See "Using M4_SYSDEF in Rebuild Mode," below.

  - Invoke M4_SYSDEF again and redefine the entire file.

  - Use the Editor to edit the file directly.

USING M4_SYSDEF IN REBUILD MODE

Using the M4_SYSDEF program in rebuild mode allows you to enter the program dialog at any point to make changes to an existing CLM file. The M4_SYSDEF program will only ask you those questions in the dialog that pertain to the specific device, class of devices, or system parameter that you wish to modify. Because you need not step through the entire M4_SYSDEF dialog in sequence to get to the section(s) you wish to modify, you can make random modifications quickly and easily.

The first time you create a CLM file using M4_SYSDEF, the program generates a CLM directive file and a file containing information for the rebuild function, called the restart file. Each time you modify a CLM file, the M4_SYSDEF program creates a unique corresponding restart file under the same directory as your CLM file.(1) Whenever you terminal the M4_SYSDEF program normally the full pathnames of your CLM file and its corresponding restart file are displayed at your terminal. You never assign a name to the restart file - the M4_SYSDEF program does this for you automatically. You need only be aware of the pathname of the restart file that corresponds to the CLM directive file you wish to modify.

If you wish to modify a CLM file at some later time, you simply invoke M4_SYSDEF and specify the pathname of the corresponding restart file on the command line. The M4_SYSDEF program reads your restart file. See "M4_SYSDEF Dialog In Rebuild Mode" later in this section for a complete description of the rebuild mode dialog.

(1)
There is no limit to the number of restart files that the M4_SYSDEF program creates. You must delete restart files which are no longer needed with the delete file (DL) command, as described in the Commands manual.

OUTPUT OPTIONS

You have several output options when using M4_SYSDEF:

1. You can request M4_SYSDEF to list all the CLM directives it has built for you. If you spot any errors at this point (or at any point during the interactive dialogue) you can perform any one of the actions described above under "Modifying Your CLM Directive File".

2. You have the option of creating a CLM_USER file which contains the generated directives or you can terminate the interactive session. If you have <u>not</u> listed the CLM directives and you do <u>not</u> wish to create a file using them, M4_SYSDEF warns you that the directives will be lost and processing will terminate, and asks if this is what you intend.

3. You have the option of giving your directive file a name other than the default file name CLM_USER.(1) After you specify a file name, M4_SYSDEF creates the file under your current working directory, assumed to be (but not restricted to) directory SID in the rest of this section.

(1)
If a CLM file already exists in your working directory, M4_SYSDEF cautions you that your new CLM file will replace the existing one, if both have the same file name.

Invoking M4_SYSDEF

Before invoking M4_SYSDEF, you are required to transfer control of the operator's terminal from the $S task group (the system operator's task group) to the $H task group (a system-supplied user task group). (If the terminal is already under control of the $H task group, proceed to the paragraph below that describes how to change your working directory to the directory SID.) Enter the following lines at the operator's terminal immediately following system startup:

C:$H:(Transfers control) of the operator's terminal from $S to $H task group)

RDN (The ready on (RDN) command causes a prompt message to appear at your terminal. This command is for convenience only and is not required).

If you have entered RDN, the system will respond with the prompt message:

($H)RDY:

Note that no prompt message appears unless you specified RDN.

Next, change your working directory to the directory SID by entering:

CWD >SID

Then invoke M4_SYSDEF by entering:

M4_SYSDEF [path]

where the optional argument [path] specifies the pathname of a system-supplied restart file. The pathname must be of the form M4RST.n where n is an integer. You specify this argument only if you wish to enter the M4_SYSDEF program in rebuild mode to modify an existing CLM file. See "M4_SYSDEF Dialog In Rebuild Mode" later in this section.

You can terminate the interactive session at any time by entering C $H and then, following the BREAK message, using the unwind (UW) command. (It is assumed that you are at the operator's terminal. If you are not at the operator's terminal, press the BREAK message, enter the UW command). If you make a mistake while creating the directive file, follow one of the options described elsewhere in this section under "Modifying Your CLM Directive File."

Prompt Modes

M4_SYSDEF operates in two prompt modes - long and short. In the long mode, each question is preceded by a multiline onscreen explanation of the question being asked. In the short mode, only the question is asked. Users of the short prompt mode who need help can see the long explanation printed as a "help message" by asking for help (see below).

M4_SYSDEF asks you which mode you want to use when you invoke the program. If you select the long mode, the explanatory text appears only once. If the program repeats the question (because, perhaps, you answered it incorrectly), the short prompt is repeated unless you ask for help. It is recommended that first-time system builders execute M4_SYSDEF in the long prompt mode.

M4_SYSDEF DIALOGUE

The short mode prompt messages which make up the M4_SYSDEF dialogue are listed below, in order of their appearance. Unless otherwise noted, these responses are always acceptable:

- ? - Help
- YES or Y - Yes (for/yes/no questions)
- NO or N - No (for yes/no questions)
- 0, NONE, or N - None (for questions requiring a numeric response)

A response of [C/R] (RETURN) is never acceptable (i.e., there are no default responses).

Once you invoke M4_SYSDEF, it responds with:

Hello! This tool will help you generate the file containing the Configuration Load Manager directives (the CLM_USER file), which is necessary to build your GCOS6/MOD400 Operating System. Do you wish to use the long prompt mode?

- In the long mode, every prompt message is preceded by the text of the "help" message, explaining what the requested information is, where you can find out if you have it or need it, and/or what will happen if you supply certain values.

  In the short prompt mode, only the prompt message is displayed, unless you enter?. Entering? doesn't change the mode.

- Acceptable answers are YES, Y, NO, or N.

What is the total amount (in words) of real memory on the machine this system is to run on? Enter the value in the form: nnnk.

- This value is the total amount of real (physical) memory installed on the machine for which this system is being built. The value you enter may differ from the amount of memory of the machine on which you invoke M4_SYSDEF. The minimum configuration for MOD 400 is 48K words.

- Acceptable answers range 48K to 1024K. Consult Appendix C for information about minimum system requirements.

Do you wish the system to supply the channel numbers for non-communication devices?

- Every peripheral device attached to the system must have a megabus channel number assignment. The program can use the floating channel feature (see Section V) to relieve you of the responsibility of assigning channel numbers to noncommunications devices: note, however, that you must always provide channel numbers for communications devices.

- Acceptable answer are YES, Y, NO, or N.

- If you reply YES, the program will ask you to supply device types for the operator's console and the boot device.

- If you reply NO, the program will prompt you to supply a channel number for each device in your configuration.

Enter the channel number of the operator terminal which will be the CONSOLE.

- You must specify the channel to which the operator's terminal is connected. M4_SYSDEF verifies for you that this channel number has not already been assigned to another device. However, only you can verify that the specified channel is actually connected to the terminal designated as the operator terminal for your installation: M4_SYSDEF cannot check this information for you.

- The channel number specified must consist of four hexadecimal characters ending in '00' or '80'.

Enter the channel number of the boot device.

- You must specify the channel number of the drive on which you mount the disk containing the system software.

- The channel number specified must consist of four hexadecimal characters ending in '00' or '80'.

- This prompt does not appear if you specified that M4_SYSDEF will supply the channel numbers for all non-communications devices in your configuration.

Do you wish to boot the fixed platter?

- This promt applies only if you wish to boot from the fixed platter of a cartridge disk or cartridge module disk.

- Acceptable answers are YES, Y, NO, or N.

Is this a Commercial System? Reply "yes" or "no".

- If you have a Model 47 or Model 57 central processor, answer YES. If you have any other model central processor, you must answer NO.

- Acceptable answers are YES, Y, NO, or N.

- If you respond NO to this question, the program replies:

Will this system have a Commercial simulator?

- If your system isn't a commercial system, it may require the commercial instruction simulator. Otherwise, it won't be able to execute commercial instructions (used, for instance, by immediate COBOL).

- Acceptable answers are YES, Y, NO, or N.

Will this system have a SIP? Reply "yes" or "no".

- The SIP is the Scientific Instruction Processor software option.

- Acceptable answers are YES, Y, NO, or N.

- If you reply NO to this question, the program replies:

Will this system have a simulator for the SIP? Reply "yes" or "no".

- Acceptable answers are YES, Y, NO, or N.

- If you reply NO, then the system won't be able to support scientific instruction (used, for instance, in FORTRAN).

- If you reply YES to this question, the program replies:

Do you want the double precision simulator SIPSIM?

- Acceptable answers are YES, Y, NO, or N.

- If you reply yes, the double-precision simulator will give you twice as many significant digits in your FORTRAN calculations as the single-precision simulator. (If you chose this option, be aware that numerical values stored in memory will take up twice as much space as values stored for the single-precision simulator).

The default system identification is "GCOS6/MOD400". If you wish to change it, reply "YES"; if you do not wish to change it, reply "NO". Do you?

- The system identification message appears as a comment line near the top of the CLM file that M4_SYSDEF generates for you. You may, if you wish, customize this message for your installation's CLM directive file.

- Acceptable answers are YES, Y, NO, or N.

- If you reply YES to this question, the program replies:

Enter the new System Identification (30 characters or less) in the form: 'SYSTEM_IDENTIFICATION'

- You must enclose the entire string in apostrophes, as shown.

The default installation identification is 'Your Town, USA'. If you wish to change it, reply 'YES'; if you do not wish to change it, reply 'NO'. Do you?

- The installation identification message appears as a comment line near the top of the CLM file that M4_SYSDEF generates for you. You may, if you wish, customize this message for your installation's CLM directive file.

- Acceptable answers are YES, Y, NO, or N.

- If you reply YES, the program responds:

Enter the new installation identification (30 characters or less) in the form: 'Installation id'

- You must enclose the entire string in apostrophes, as shown.

Please enter the maximum number of concurrent users that you expect this system to support.

- This value allows for the calculation of some of the system structure sizes and memory needs which vary depending upon the maximum load for which the system is to be built. M4_SYSDEF assumes that the number of users is synonymous with the number of task groups.

- Note that entering a given number here does not guarantee that the maximum number of concurrent users can in fact be supported. This depends on such dynamic variables as the work each user is performing and memory fragmentation.

- Nonnumeric characters are not acceptable in this answer.

Should I automatically generate the SPD names for you?

- Each peripheral device attached to the system must have a symbolic peripheral device name. This entry is a five-character name which conforms to the file system naming conventions.

- Acceptable answers are YES, Y, NO, or N.

- If you reply YES, the program will generate 5-character SPD names without further prompting. If you reply NO, the program will prompt you to supply an SPD name for every device in your configuration.

Do you wish this system to activate the Record Locking Feature of Data Management?

- Record Locking is the finest granularity of concurrency control provided by data management. Activation of this runtime option allows more than one user to write records to a given file simultaneously. Proper usage of this option ensures that all updates are correctly applied to the file.

- Acceptable answers are YES, Y, NO, or N.

- If you reply YES, record locking is implemented. If you reply NO, data management will enforce file level concurrency (i.e., only one user can have write permission on a file at one time).

Do you wish to include Display Formatting and Control Software?

- The Display Formatting and Control software is described in the Display Formatting and Control manual and in Appendix P of this manual.

- Acceptable answers are YES, Y, NO, or N.

- If you reply, YES, the display processing software will be included in your Executive. If you reply NO, no display processing software will be included.

Do you wish to activate the power fail restart?

- This prompt asks if you wish to configure the power resumption facility. The power resumption facility allows the system execution environment to be restarted after a power interruption, if fewer than two hours have elapsed when power, is returned to the central processor. For more information, refer to Appendix D.

- If you reply YES, each ATD amd TTY communications (MLCP/DLCP-connected) device in your configuration will be automatically reconnected if a power resumption or line drop condition occurs. (Peripheral devices not attached to an MLCP/DLCP are automatically designated as reconnectable when they are configured).

- If you reply NO, no automatic reconnection of MLCP/DLCP-connected terminal devices will occur.

Does this system have any communications type devices? i.e., VIP's, TTY's, etc.

- Acceptable answers are YES, Y, NO, and N.

- A communications device is an MLCP/DLCP-connected device.

- If you reply NO, the program skips to the prompt. "How many MDC-connected console devices do you have?", later in this dialogue. If you reply YES, you will see the following prompts.

How many communications priority levels (1 to 4) do you wish? Enter the number of levels.

- Each device must be assigned to a hardware interrupt level. Noncommunications devices must be assigned to unique interrupt levels; M4_SYSDEF can perform this assignment for you. However, communications devices can share a priority level; therefore, you are given a chance to specify relative priority levels for each communications device.

- Acceptable answers are 1, 2, 3, or 4. If you specify 1, all communications devices will share one interrupt level. If you specify 2 through 4, you will have the capability to assign specific devices to different levels, with those devices on level 1 having the highest relative priority and those devices on level 4 having the lowest relative priority. You do <u>not</u> have to assign at least one devide to each level; M4_SYSDEF reserves any unused levels for you.

How manu MLCP's (1 to 10) will this system have?

- All communications devices must be attached to a multiline communications processor (MLCP).

- Acceptable answers are 1 through 10, inclusive.

How many ATDs do you have?

- The program is now attempting to configure all MLCP/DLCP connected asynchronous terminal devices.

- The following asynchronous terminal devices are supported: VIP 7200, VIP 7205, VIP 7801, VIP 7802, VIP 7207, PRU1001, PRU1003, PRU1005, TWU1001, TWU1003, TWU1005, and various teleprinter-compatible devices.

- If you reply NONE, the program skips to the prompt, "How many VIP's do you have?, below.

- If you reply with a nonzero number, the program continues:

ATD terminal number N

- This message, and the three prompt following it, appear N times, where N is your response to the earlier question, "How many ATDs do you have?".

What channel number will this device be on?

- You must enter exactly four hexadecimal characters, ending in "00" or "80".

Enter the communications priority level for this terminal.

- Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".

- If you specified only one communications priority level, this prompt won't appear.

What is the device type for this ATD device?

- You must specify one of the following values (there is no default):

Value	Physical Device Type
7200	VIP 7200, VIP 7205
7801	VIP 7801, VIP 7802
D7200	VIP 7207
PRU	PRU 1001, PRU1003, PRU 1005, TWU 1001, TWU1003, TWU 1005,
TTY	Can be any asynchronous terminal device-type classified as teleprinter compatible.

How many VIP's do you have?

- The program is now attempting to configure synchronous terminals. The following synchronous terminals are supported by M4_SYSDEF:

  VIP 7700
  VIP 7700R/VIP 7705R
  VIP 7760
  VIP 7804/VIP 7805 (in VIP 7700 mode)

- To use the extended features of the VIP 7804 or VIP 7805, you may want to use an STD directive. STD groups are configured later on in the M4_SYSDEF dialog.

- If you reply 0, the program skips to the prompt, "How many H3270 Host Links do you have?", below.

. If you reply with a nonzero number, the program continues:

VIP terminal number N

. This message, and the seven prompts following it, appear N times, where N is the number you entered in response to the earlier question, "How many VIP's do you have?".

What channel number will this device be on?

. You must provide the Megabus channel number to which this device is attached.

. You must enter exactly four hexadecimal characters, ending in "00" or "80".

How many VIP's are polled on this channel?

. ? is not acceptable here.

. Acceptable answers are the numbers 0 through 31, inclusive.

Enter the communications priority level for this terminal.

. ? is not acceptable here.

. Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".

. If you specified only one communications priority level, this promt won't appear.

Enter the polling address for this VIP.

. Acceptable answers are 0 through 31, inclusive.

. This promt appears only if you specified a non-zero value for the prompt "How manu VIP's are polled on this channel?", above.

Is this VIP a "Control Station"?

. ? is not acceptable here.

. Acceptable answers are YES, Y, NO, or N.

Does this VIP have a ROP (receive-Only Printer)?

. Acceptable answers are YES, Y, NO, or N.

. If you reply NO, the program skips the next two prompts. If you reply YES, the program continues:

What type of ROP do you have?

- There are eight supported ROPs:

Specify One	Devices Supported (See Appendix C)
TN100	TermiNet 100 or PRU 1001
TN300 (default)	TermiNet 300 or PRU 1003
TN1200	TermiNet 1200 or PRU 1005
TTY33	ASR-33
TTY35	ASR-35

Does this ROP support form feed?

- ? is not acceptable here.
- Acceptable answers are YES, Y, NO, or N.

How many H3270 Host Links do you have?

- The program is now attempting to configure stations for the BSC 3270 line protocol handler in your configuration.
- Acceptable answers are a nonzero number or NONE.
- If you reply NONE, the program skips to the prompt, "How many PVE groups do you have?", below.
- If you reply with a nonzero number, the program continues:

H3270 line number N

- This messages, and the 4 prompts following it, appear N times, where N is the number you entered in response to the earlier question, "How many H3270 Host Links do you have?".

What channel number will this line be on?

- You must provide the Megabus channel number to which this line is attached.
- You must enter exactly four hexadecimal characters, ending in "00" or "80".

Enter the communications priority level for this line.

- ? is not acceptable here.
- Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".
- If you specified only one communications priority level, this prompt won't appear.

Enter the polling address (decimal) for this H3270 Host Link.

- ? is not acceptable here.

- Acceptable answers range from 0 through 255, inclusive. M4_SYSDEF converts this number to a hexadecimal value for you when it defines the H3270 directive.

Enter the select address (decimal) for this H3270 Host Link.

- ? is not acceptable here.

- Acceptable answers range from 0 through 255, inclusive. M4_SYSDEF converts this number to a hexadecimal value for you when it defines the H3270 directive.

How many PVE groups do you have?

- The program is now attempting to build facilities for polled VIP emulated (PVE) groups into your configuration. (1)

- If you reply "NONE", the program skips to the prompt, "How many STD groups do you have?".

- If you reply with a nonzero number, the program continues:

PVE group number N

- This message, and the four prompts following it, appear N times, where N is your response to the earlier question, "How many PVE groups do you have?".

Enter channel number for PVE group N

- ? is not acceptable here.

- You must enter exactly four hexadecimal characters, ending in "00" or "80".

How many PVE stations are there in this channel?

- ? is not acceptable here.

- Acceptable answers are 1 through 32, inclusive.

Enter the communications priority level for this group.

- ? is not acceptable here.

- Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".

(1)
A PVE consists of all the PVE stations assigned to the same hardware channel.

- If you specified only one communications priority level, this prompt won't appear.

Enter the polling address (0 to 31) for PVE station N

- ? is not acceptable here.

- Acceptable answers are 0 through 31, inclusive.

- The prompt appears N times, where N is your response to the earlier question, "How many PVE groups do you have?".

How many STD groups do you have?

- The program is now attempting to configure all MLCP/DLCP connected synchronous terminal devices.

- The following synchronous terminal devices are supported: VIP 7804, VIP 7805, VIP 7700, VIP 7700R, VIP 7705R, and VIP 7760.

- If you reply NONE, the program skips to the prompt: "How many MLCP-connected Binary Synchronous Communications (BSC) lines do you have?", below.

- If you reply with a nonzero number, the program continues:

STD group number N

- This message, and the 8 prompts following it, appear N times, where N is your response to the earlier question, "How many STD groups do you have?".

Enter channel number for STD group N

- You must enter exactly four hexadecimal characters, ending in "00" or "80".

Enter the communications priority level for this group

- Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".

- If you specified only one communications priority level, this prompt won't appear.

How many STD stations are there on this channel?

- The reply "?" is nor acceptable here.

- Acceptable answers are 1 through 32, inclusive.

Enter the polling address (0 to 31) for STD station N

- ? is not acceptable here.

- Acceptable answers are 0 through 31, inclusive.

- The prompt appears N times, where N is your response to the earlier question, "How many STD groups do you have?".

What is the device type for this STD device?

- You must specify one of the following device types:

Device type	Supported Devices
V7804	VIP 7804, VIP 7805
V7700	VIP 7700, VIP 7700R, VIP 7705R
V7760	VIP 7760

Does this STD have a ROP (Receive-only printer)?

- If you reply NO, the program skips the next prompt. If you reply YES, the program continues:

What type of ROP do you have?

- There are four supported ROPs:

Specify One	Devices Supported
ROSY24	PRU 1003
ROSY26	PRU 1005
TN300	TermiNet 300
TN1200	TermiNet 1200

How many MLCP-connected Binary Synchronous Communication (BSC) lines do you have?

- The program is now attempting to build facilities for BSC lines into your configuration.

- ? is not acceptable here.

- If you reply 0, the program skips to the prompt, "How many MDC-connected console devices do you have?", below.

- If you reply with a nonzero number, the program continues:

BSC line number N

- This message, and the three prompts following it, appear N times, where N is your response to the earlier question, "How many MLCP-connected BSC lines do you have?".

What channel number will this device be on?

- ? is not acceptable here.

- You must enter exactly four hexadecimal characters, ending in "00" to "80".

Enter the communications priority level for this terminal.

- ? is not acceptable here.
- Acceptable answers range from 1 through your response to the earlier question, "How many communications priority levels (1 to 4) do you wish?".
- If you specified only one communications priority level, this prompt won't appear.

Is this BSC line a "Primary" one.

- ? is not acceptable here.
- Acceptable answers are YES, Y, NO, or N.

How many MDC-connected console devices do you have?

- The program is now attempting to build facilities for all MDC-connected teminal devices.
- If you reply NONE, the program skips to the prompt, "How many diskettes do you have?", below.
- If you reply with a nonzero number, the program continues:

CON number 1

What channel number will this device be on?

- This prompt appears only if you are explicitly entering channel numbers.
- This prompt appears N times, where N is your response to the earlier question, "How many MDC-connected console devices do you have?".
- You must enter exactly four hexadecimal characters, ending in "00" or "80".

What is the device type for the Operator's CONSOLE?

- You must specify one device as the Operator's temrinal:

Value	Physical Device Type
7200	VIP 7200 (DKU9103) VIP 7205 (DKU9102) The value 7200 must be used if MDC-connected VIP 7200 terminals are to be used for forms processing using the Display Formatting and Control Facility.
PRU	PRU1003/PRU1005/TWU1003/TWU1005
D7200	VIP 7207
KSR	The terminal is to be supported in teleprinter mode.
7801	VIP 7801, VIP 7802

CON number N

- This message, and the two prompts following it, appear N - 1 times, where N is your response to the earlier question, "How many MDC-connected console devices do you have?".

What channel number will this device be on?

- You must enter exactly four hexadecimal characters, ending in "00" or "80".
- This prompt appears only if you are explicitly entering channel numbers.
- ? is not acceptable here.

What is the device type for this CON device?

- Specify one of the values previously described for CON number 1.

How many diskettes do you have?

- The program is now attempting to build facilities for diskette devices into your configuration. (See Appendix C for a list of diskette device).
- If you reply 0, the program skips to the prompt, "How many cartridge disks to you have?", below.
- If you reply with a nonzero number, the program continues:

Diskette number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.

- This prompt appears N times, where N is your response to the earlier question, "How many diskettes do you have?".

- ? is not acceptable here.

- You must supply exactly four hexadecimal characters, ending in "00" or "80".

How many cartridge disks do you have?

- The program is now attempting to build facilities for cartridge disk devices into your configuration. (See Appendix C for a list of cartridge disk devices).

- Your configuration must have at least one cartridge disk, cartridge module disk, or mass storage unit.

- If you reply 0, the program skips to the prompt, "How many cartridge module disks do you have?", below.

- If you reply with a nonzero number, the program continues:

Cartridge Disk number N

- This message, and the two prompts following it, appear N times, where N is your response to the earlier question, "How many cartridge disks do you have?".

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.

- ? is not acceptable here.

- You must supply exactly four hexadecimal characters, ending in "00" or "80".

Does this Catridge Disk have a fixed platter?

- ? is not acceptable here.

- Acceptable answers are YES, Y, NO, or N.

How many cartridge module disks do you have?

- The program is now attempting to build facilities for cartridge module disks into your configuration. (See Appendix C for a list of cartridge module disks).

- If you reply 0, the program skips to the prompt, "How many storage modules do you have?", below.

- If you reply with a nonzero number, the program continues:

Catridge Module Disk Number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers

- This prompt appears N time, where N is your response to the earlier question, "How many cartridge module disks do you have?".

- ? is not acceptable here.

- You must supply exactly four hexadecimal characters, ending in "00" or "80".

How many storage modules do you have?

- The program is now attempting to build facilities for mass storage devices into your configuration. (See Appendix C for a list of mass storage devices).

- If you reply NONE, the program skips to the prompt, "How many card reader/punches do you have?", below.

- If you reply with a nonzero number, the program continues:

Storage Module number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.

- This prompt appears N times, where N is your response to the earlier question, "How many storage modules do you have?".

- ? is not acceptable here.

- You must enter exactly four hexadecimal characters, ending in "00" or "80".

How many card readers do you have?

- The program is now attempting to build facilities for card reader devices into your configuration. (See Appendix C for a list of card reader devices).
- If you reply 0, the program skips to the prompt, "How many card reader/punches or punches do you have?", below.
- If you reply with a nonzero number, the program continues:

Card Reader number N

What channel number will this device be on?

- This promot only appears if you are explicitly extering channel numbers.
- This prompt appears N times, where N is your response to the earlier question, "How many card readers do you have?".
- ? is not acceptable here.
- You must supply exactly four hexadecimal characters, ending in "00" or "80".

How many card reader/punches or punches do you have?

- The program is now attempting to build facilities for card reader/punch devices or card punch devices into your configuration. Note that "card reader/punch" doesn't mean card readers or card punches, but devices that both read and punch cards. (See Appendix C for a list of card reader/punch devices).
- If you reply 0, the program skips to the prompt, "How many printers do you have?", below.
- If you reply with a nonzero number, the program continues:

Card reader/punch or punch number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.
- ? is not acceptable here.
- This prompt appears N times, where N is your response to the earlier question, "How many card reader/punches or punches do you have?".
- You must supply exactly four hexadecimal characters, ending in "00" or "80".

How many printers do you have?

- The program is now attempting to build facilities for line printer devices into your configuration. (See Appendix C for a list of line printer devices).

- If you reply 0, the program skips to the prompt, "How many 9-track magnetic tape device do you have?", below.

- If you reply with a nonzero number, the program continues:

Printer number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.

- This prompt appears N times, where N is your response to the earlier question, "How many printers do you have?".

- ? is not acceptable here.

- You must supply exactly four hexadecimal characters, ending in "00" or "80".

How many 9-track magnetic tape devices do you have?

- The program is now attempting to build facilities for 9-track magnetic tape devices into your configuration. (See Appendix C for a list of 9-track magnetic tape devices).

- If you reply 0, the program skips to the prompt "do you wish to have ALL system overlays resident (yes or no)?", below.

- If you reply with a nonzero number, the program continues:

Magnetic Tape Device number N

What channel number will this device be on?

- This prompt only appears if you are explicitly entering channel numbers.

- This prompt appears N times, where N is your response to the earlier question, "How many 9-track magnetic tape devices do you have?

- ? is not acceptable here.

- You must supply exactly four hexadecimal characters, ending in "00" or "80".

Do you wish to have ALL system overlays resident? (yes or no)

- Acceptable answers are YES, Y, NO, or N.

- If you amswer yes, the system optimizes the execution speed of all functions; the resulting system will occupy more memory than a system without resident overlays. The program skips to the prompt, "Do you wish to create a batch memory pool?".

- If you answer no, the program continues:

Do you wish to optimize the reading/writing of relative files?

- By making particular system input/output overlays permanently resident, you can obtain optimum speed for selected file types. This prompt, and the six that follow, ask if you wish to optimize the execution speed necessary to perform different functions.

- Acceptable answers are YES, Y, NO, or N.

Do you wish to optimize the reading/writing of random files?

- Acceptable answers are YES, Y, NO, or N.

Do you wish to optimize the reading/writing of indexed files?

- Acceptable answers are YES, Y, NO, or N.

Do you wish to optimize releative files with deletable records?

- Acceptable answers are YES, Y, NO, or N.

Do you wish to optimize file management open/close functions?

- Acceptable answers are YES, Y, NO, or N.

Do you wish to optimize the reading/writing of tape files?

- Acceptable answers are YES, Y, NO, or N.

- This prompt appears only if you previously specified that your system supports magnetic tape devices.

Do you wish to optimize EBCDIC tape functions?

- Acceptable answers are YES, Y, NO, or N.

Do you wish to create a batch memory pool?

- Acceptable answers are YES, Y, NO, or N.

- If you answer no, the program skips to the prompt, "Enter the number (1 to 10) of 320-word system overlay areas you desire", below. If you answer yes, the program continues:

How big a batch pool? (nnnnn or nnk words)

- Nonnumeric characters (except for "k") are not acceptable here.

- You must supply a numeric value of the form nnnnn or nnk words

Enter the number (1 to 99) of system overlay areas you desire.

- Nonnumeric characters are unacceptable here.

- The number of system overlays that you specify will be loaded into the system area of memory where they will remain resident for the duration of the configured system. The larger the number of system overlays is, the more likely it is that frequently used system overlays will be found in memory.

Would you like to see the generated statements?

- Acceptable answers are YES, Y, NO, or N.

- If you answer yes, M4_SYSDEF lists the contents of the file of CLM directives it has generated for you. At the top of the listing (below the SYS directive statement) the program identifies for you the version of M4_SYSDEF that you are using, as well as the date and time that your CLM directive file was generated.

- If you spot an error at this point, you can enter rebuild mode by hitting the break key and then typing the command PI (program interrupt). See "M4_SYSDEF Dialog In Rebuild Mode" later in this section.

- If you answer no, the program continues:

Do you want to create a CLM directive file now?

- Acceptable answers are YES, Y, NO, or N.

- If you reply no to both this question and the preceding question, the program warns you that the directives will be lost and processing will terminate, and asks if this is what you intend. If you reply yes, this is what you intend, processing terminates, and no directive file is created. If you reply no, the previous question, "Would you like to see the generated statements?", is repeated.

- If you reply no to this question but replied yes to the preceding question, processing terminates.

- If you reply yes, the M4_SYSDEF program automatically generates a map file for you containing configuration information you may find useful. (For a complete explanation of the map file, refer to the description of the MAP directive in Section V). The program assigns this map file the pathname >SID>CLM_MAP. If you wish to examine CLM_MAP after you create your CLM file, you can use a print (PR) command or invoke the Editor and use the Editor's read and print directives.

- If you reply yes, the program continues:

Would you like to use a file name other than CLM_USER for this file?

- Acceptable answers are YES, Y, NO, or N.

- You may wish to use the default file name CLM_USER or a file name of your own. If you answer no, M4_SYSDEF creates CLM_USER under your current working directory SID and processing terminates.

- If this file already exists in your current working directory, M4_SYSDEF informs you and gives you the option of quitting, re-writting the file, or providing another file name.

- Note that although you are free to rename your CLM file, the system must be able to locate a file named CLM_USER under directory SID at system startup. Remaining your CLM file allows you the option of creating more CLM files than just the one your system needs at configuration time. You may, for example, be creating CLM files for several installations, in which case you may want each one to have a unique name for identification purposes.

- If you answer yes, the program continues:

Enter your filename --

- The file name you choose must be no longer than 12-characters and must conform to standard file name conventions.

- After you enter your file name, M4_SYYSDEF creates your CLM file under your working directory SID, and processing terminates. The program lists the full pathnames of the CLM file and its corresponding restart file at your terminal. If, at some later time you wish to modify your CLM file, see "Using M4_SYSDEF in Rebuild Mode", earlier in this section.

- If this file already exists in your current working directory, M4_SYSDEF informs you and gives you the option of quitting, re-writing the file, or providing another file name.

M4_SYSDEF DIALOG IN REBUILD MODE

Once you invoke M4_SYSDEF and specify the pathname of a restart file, the program responds with: You have entered M4_SYSDEF in the rebuild mode. Do you want to see the existing CLM statements?

- If you respond YES, the program lists the CLM directives contained in your restart file.

- If you respond NO, the program continues:

Do you wish to modify devices or system parameters?
Respond with "SYSTEM", "DEVICES", "VIEW", or "DONE".
Response =

- Depending on your response, the program will enter the M4_SYSDEF dialog at the appropriate place so you can make modifications. Table 4-1 lists each question and response in the rebuild mode dialog. The complate M4_SYSDEF dialog is described earlier in this section.

- If you respond SYSTEM, the program asks the question, "Do you wish to modify the prompt (PROMPT), memory size (MEMORY), the number of interactive users (USERS), the automatic generation of SPD names (SPD), the automatic channel assignment (CHANNEL), the record locking feature (RLOCK), the system overlay areas (SOAT), the resident overlays (RESOLA), the display formatting feature (VDAM), the commercial or SIP simulators (SIMUL), or the system and installation identification (IDENT)? Depending on your response to this question, the program will enter the M4_SYSDEF dialog at the appropriate place so you can make modifications. Once you have finished making your modifications, enter the reply DONE.

- If you respond DEVICES, the program asks the question, "Respond with the device type or class that you wish to modify. "COMM", "CON", "DSK", "RCD", "MSM", "CMD", "CDR", "CRP", "LPT", "MTP", or "DONE". Depending on your response to this question, the program will enter the M4_SYSDEF dialog at the appropriate place so you can make modifications. If you reply DONE, the program will repeat the prompt, "Do you wish to modify devices or system parameters?" When you have finished making your modifications, enter the reply DONE.

- If you reply VIEW, the program lists the current CLM file at your terminal.

- If you reply DONE, the program skips through the M4_SYSDEF dialog to the prompt, "Would you like to see the generated statements?" Normal M4_SYSDEF termination prompts follow.

- When M4_SYSDEF processing terminates normally, the program will list the pathnames of your CLM directive file and its associated restart file. If you wish to modify this CLM file at some later time, you invoke M4_SYSDEF, specifying the pathname of this restart file on the command line.

Table 4-1 lists each question and response in the rebuild mode dialog. Using this table, you can trace each of your possible responses to a specific entry point in the M4_SYSDEF program dialog.

To use this table, begin in the left hand column and read across to the right. Note that the questions you are required to answer become more specific as you isolate the specific section of the M4_SYSDEF dialog that you wish to enter. The fifth column lists the actual questions in the M4_SYSDEF program dialog that appear as a result of your response to previous rebuild mode questions. Column six lists the pages in this section of the manual that explain each M4_SYSDEF question in detail.

Table 4-1. Rebuild Mode Dialog

Rebuild Mode Dialog				M4_SYSDEF Dialog	
Question	Response	Question	Response	Prompt	Page Reference
Do you wish to modify devices or system parameters? Respond with "SYSTEM", "DEVICES", "VIEW", or "DONE".	SYSTEM	Do you wish to modify the prompt (PROMPT), memory size (MEMORY), the number of interactive users (USERS), the automatic generation of SPD names (SPD), the automatic channel assignment (CHANNEL), the record locking feature (RLOCK), the system overlay areas (SOAT), the resident overlays (RESOLA), the display formatting feature (VDAM), the commercial or SIP simulators (SIMUL), or the system and installation identification (IDENT)?	PROMPT	Do you wish to use the long prompt mode?	4-8
			MEMORY	What is the total amount (in words) of real memory on the machine this system is to run on? Enter the value in the form: nnnk.	4-8
			USERS	Please enter the maximum number of concurrent users that you expect this system to support.	4-12
			SPD	Should I automatically generate the SHD names for you?	4-12
			CHANNEL	Do you wish the system to supply the channel numbers for noncommunications devices?	4-9
			RLOCK	Do you wish this system to activate the Record Locking Feature of Data Management?	4-12
			SOAT	Enter the number of system overlay areas you desire.	4-28
			RESOLA	Do you wish to have all system overlays resident?	4-26
			VDAM	Do you wish to include Display Formatting and Control Software?	4-13
			SIMUL	Is this a commercial system? Reply "YES" or "NO".	4-10
			IDENT	The default system identification is 'GCOS6/MOD400'. If you wish to change it, reply 'YES'; if you do not wish to change it, reply 'NO'. Do you?	4-11
	DEVICES	Respond with the device type or class that you wish to modify. "COMM", "CON", "DSK" "MCD", "MSM", "CMD", "CDR", "CRP", "LPT", "MTP", or "DONE".	COMM	Do you wish to modify the communications levels and the number of MLCP's (COMM) (see note), the vip's (VIP), the Host Links BH3270), the pve's (PVE), the synchronous terminals (STD), the asynchronous terminals (ATD), or the bsc's (BSC)? NOTE: Type COMM to enter the dialog at the question "Does this system have any communications type devices?" Then type VIP, H3270, PVE, STD, ATD, or BSC to enter the dialog at a more specific point.	COMM,4-13 VIP,4-15 H3270,4-17 PVE,4-18 STD,4-19 ATD,4-14 BSC,4-20

Table 4-1 (Cont). Rebuild Mode Dialog

Rebuild Mode Dialog				M4_SYSDEF DIALOG	
Question	Response	Question	Response	Prompt	Page Reference
Do you wish to modify devices or system parameters? Respond with "SYSTEM", "DEVICES", "VIEW", or "DONE".			CON	How many MDC-connected console devices do you have?	4-21
			DSK	How many diskettes do you have?	4-22
			RCD	How many cartridge disks do you have?	4-23
			MSM	How many storage modules do you have?	4-24
			CMD	How many cartridge module disks do you have?	4-23
			CDR	How many card readers do you have?	4-24
			CRP	How many card reader/punches or punches do you have?	4-25
			LPT	How many printers do you have?	4-25
			MTP	How many 9-track magnetic tape devices do you have?	4-26
			DONE	Do you wish to modify devices or system parameters? Respond with "SYSTEM", "DEVICES", "VIEW", or "DONE".	4-30
	VIEW	(No question is asked. This response causes the current CLM directive file to be listed at your terminal.)			
	DONE	Would you like to see the generated statements?			

EXAMPLES OF M4_SYSDEF USAGE

The following examples illustrate how you can first create a CLM directive file and then modify it at some later time by invoking M4_SYSDEF in rebuilt mode. Example 1 steps through the entire M4_SYSDEF dialog. Example 2 illustrates the rebuild mode dialog.

Example 1:

Following a Stage 1 (initial) system startup, a new user wishes to create a CLM file using M4_SYSDEF. Prior to invoking M4_SYSDEF, the user must transfer control of the operator's terminal from the $S (system) task group to the supplied user task group $H:

Δ C Δ :$H:

The terminal is now under control of the $H task group. The user wishes to see the ready prompt message, and so enters the RDN (ready on) command:

RDN

The system responds:

($H)RDY:

The user invokes M4_SYSDEF and defines a CLM file:

```
M4_SYSDEF
HELLO!  THIS TOOL WILL HELP YOU GENERATE THE FILE CONTAINING
THE CONFIGURATION LOAD MANAGER DIRECTIVES (THE CLM_USER FILE).
WHICH IS NECESSARY TO BUILD YOUR GCOS6/MOD400 OPERATING SYSTEM.
DO YOU WISH TO USE THE LONG PROMPT MODE?
?
THERE ARE TWO MODES IN WHICH TO RUN THIS TOOL, THE LONG
MODE AND THE SHORT MODE.  IF YOU PICK THE LONG MODE, EVERY PROMPT
MESSAGE WILL BE PRECEDED BY AN EXPLANATION OF WHAT THE REQUESTED
INFORMATION IS, WHERE YOU CAN FIND THE REQUESTED INFORMATION, AND
WHAT WILL HAPPEN IF YOU SUPPLY CERTAIN INFORMATION. THE SHORT MODE SHOULD
BE USED BY PERSONS WHO HAVE BUILT THIS SYSTEM BEFORE AND DO NOT NEED
THE LONG PROMPT TEXT.  IF YOU SELECT THE SHORT MODE AND THEN FIND
THAT YOU NEED HELP IN ANSWERING ONE OF THE QUESTIONS, YOU CAN OBTAIN
THE LONG MODE TEXT BY ENTERING A '?'; THIS WILL NOT CHANGE THE MODE
AND ALL SUBSEQUENT QUESTIONS WILL BE OF THE SHORT FORM.
SINCE THIS IS A GENERALIZED TOOL, CERTAIN COMMON DEFAULTS HAVE BEEN
ASSUMED; IF YOUR SYSTEM REQUIRES A CHANGE TO THESE DEFAULTS, YOU CAN
GENERATE THE 'CLM_USER' FILE AND THEN USE THE STANDARD EDITOR
TO MAKE THESE MODIFICATIONS. GOOD LUCK.
DO YOU WISH TO USE THE LONG PROMPT MODE?
YES

 THE TOTAL AMOUNT OF REAL, PHYSICAL MEMORY INSTALLED ON THE MACHINE
FOR WHICH THIS SYSTEM IS BEING BUILT MAY BE DIFFERENT FROM
THE MEMORY AVAILABLE ON THE MACHINE YOU ARE CURRENTLY USING.
THE MINIMUM CONFIGURATION FOR GCOS6/MOD400 IS '48K' WORDS.
WHAT IS THE TOTAL AMOUNT (IN WORDS) OF REAL MEMORY ON THE MACHINE THIS
SYSTEM IS TO RUN ON? ENTER THE VALUE IN THE FORM: NNNK.
128K

 IF YOU WISH, WE CAN GENERATE THE CLM DIRECTIVES WITHOUT SPECIFYING
THE CHANNEL NUMBERS FOR NON-COMMUNICATION TYPE DEVICES. FOR
COMMUNICATION DEVICES YOU MUST ALWAYS PROVIDE A CHANNEL NUMBER.
AT SYSTEM START-UP TIME, THE SYSTEM WILL POLL THE HARDWARE AND
DETERMINE UPON WHICH CHANNEL EACH DEVICE IS LOCATED.
DO YOU WISH THE SYSTEM TO SUPPLY THE CHANNEL NUMBERS FOR
NON-COMMUNICATION DEVICES?
NO

 ENTER THE CHANNEL NUMBER OF THE OPERATOR TERMINAL
WHICH WILL BE THE CONSOLE
0400

 ENTER THE CHANNEL NUMBER OF THE BOOT DEVICE
1400
DO YOU WISH TO BOOT THE FIXED PLATTER?
NO
```

Figure 4-1. Using M4_SYSDEF

```
 A COMMERCIAL SYSTEM IS A MODEL 47 OR MODEL 57.
IS THIS A COMMERCIAL SYSTEM? REPLY "YES" OR "NO."
N
SINCE YOU DO NOT HAVE A COMMERCIAL SYSTEM, WILL YOU
HAVE THE COMMERCIAL SIMULATOR; IF YOU REPLY 'NO', THEN
THIS SYSTEM WILL NOT BE ABLE TO SUPPORT COBOL.
WILL THIS SYSTEM HAVE A COMMERCIAL SIMULATOR?
Y

 SIP MEANS'SCIENTIFIC INSTRUCTION PROCESSOR'.
WILL THIS SYSTEM HAVE A SIP? REPLY "YES" OR "NO."
N
SINCE YOU DO NOT HAVE THE 'SCIENTIFIC INSTRUCTION PROCESSOR', WILL YOU
HAVE THE SIMULATOR FOR THIS HARDWARE OPTION?  IF YOU REPLY 'NO', THEN
THIS SYSTEM WILL NOT BE ABLE TO SUPPORT ANY OF THE SCIENTIFIC
INSTRUCTIONS.
WILL THIS SYSTEM HAVE A SIMULATOR FOR THE SIP? REPLY "YES" OR "NO."
YES
DO YOU WANT THE DOUBLE PRECISION SIMULATOR SIPSIM?
NO

 THE SYSTEM IDENTIFICATION IS ONE OF THE STANDARD PARTS OF THE COMMENT
STATEMENTS THIS UTILITY AUTOMATICALLY GENERATES.
IF YOU ENTER A NEW VALUE MAKE SURE IT IS ENCLOSED WITHIN
APOSTROPHE MARKS OR SOME OF THE INFORMATION MAY BE LOST.
THE DEFAULT SYSTEM IDENTIFICATION IS 'GCOS6/MOD400'.
IF YOU WISH TO CHANGE IT, REPLY 'YES'; IF YOU DO NOT WISH TO
CHANGE IT, REPLY 'NO'.  DO YOU?
Y
ENTER THE NEW SYSTEM IDENTIFICATION (30 CHARACTERS OR LESS) IN THE FORM:
'SYSTEM_IDENTIFICATION'
'GCOS6/MOD 400 VERSION 2.1'

 THE INSTALLATION IDENTIFICATION IS ONE OF THE STANDARD PARTS OF THE
COMMENT STATEMENTS THIS UTILITY AUTOMATICALLY GENERATES.
IF YOU ENTER A NEW VALUE, MAKE SURE IT IS ENCLOSED
WITHIN APOSTROPHE MARKS OR SOME OF THE INFORMATION MAY BE LOST.
THE DEFAULT INSTALLATION IDENTIFICATION IS 'YOUR TOWN, USA'.
IF YOU WISH TO CHANGE IT, REPLY 'YES'; IF YOU DO NOT WISH TO
CHANGE IT, REPLY 'NO'.  DO YOU?
YES
ENTER THE NEW INSTALLATION IDENTIFICATION (30 CHARACTERS OR LESS)
IN THE FORM: 'INSTALLATION ID'
'TEST SITE ONE'

 THE NUMBER OF CONCURRENT USERS ASSISTS IN THE CALCULATION OF SYSTEM
STRUCTURE SIZES AND MEMORY NEEDS WHICH VARY DEPENDING
UPON THE MAXIMUM LOAD THE SYSTEM IS TO BE CONFIGURED
FOR. THERE IS, HOWEVER, NO WAY TO ENSURE THAT THE MAXIMUM
NUMBER OF USERS THAT YOU SPECIFY HERE WILL INDEED
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
BE THE MAX. NUMBER OF USERS THE SYSTEM WILL SUPPORT.  THIS DEPENDS
ON SUCH DYNAMIC VARIABLES AS THE WORK EACH USER IS PERFORMING, AND
MEMORY FRAGMENTATION.
PLEASE ENTER THE MAXIMUM NUMBER OF CONCURRENT USERS THAT YOU
EXPECT THIS SYSTEM TO SUPPORT.
6

 EVERY PERIPHERAL DEVICE ATTACHED TO THIS SYSTEM MUST HAVE A
SPD (SYMBOLIC PERIPHERAL DEVICE) NAME.  THIS NAME IS A
UNIQUE 6 CHARACTER (MAXIMUM) NAME WHICH CONFORMS TO THE FILE SYSTEM
NAMING CONVENTIONS.  IF YOU REPLY 'YES' HERE, I WILL GENERATE THESE
NAMES FOR YOU WITHOUT ANY FURTHER PROMPTS.  IF YOU REPLY 'NO', I WILL
ASK YOU TO SUPPLY THE SPD NAME FOR EACH AND EVERY DEVICE AS WE
CONFIGURE THAT DEVICE
SHOULD I AUTOMATICALLY GENERATE THE SPD NAMES FOR YOU?
Y

 RECORD LOCKING IS THE FINEST GRANULARITY OF CONCURRENCY CONTROL THAT
DATA MANAGEMENT PROVIDES.  ACTIVATION OF THIS RUN-TIME OPTION PROVIDES
TOOLS TO ALLOW MORE THAN ONE USER TO WRITE RECORDS TO A GIVEN FILE AT
THE SAME TIME. PROPER USAGE OF THIS OPTION INSURES THAT ALL UPDATES
ARE CORRECTLY APPLIED TO THE FILE. IF YOU REPLY "YES", THIS OPTION WILL
BE AVAILABLE. IF YOU REPLY "NO", THEN DATA MANAGEMENT WILL ENFORCE
FILE LEVEL CONCURRENCY, I.E. ONLY ONE USER WITH WRITE PERMISSION TO A FILE.
DO YOU WISH THIS SYSTEM TO ACTIVATE THE RECORD LOCKING FEATURE OF
DATA MANAGEMENT?
YES

 IF YOU ARE PLANNING TO USE THE DISPLAY FORMATTING AND CONTROL
SOFTWARE WE NEED TO KNOW SO IT CAN BE INCLUDED IN THE SYSTEM.
DO YOU WISH TO INCLUDE DISPLAY FORMATTING AND CONTROL SOFTWARE?
YES

 DO YOU WISH TO ACTIVATE THE POWER FAIL RESTART?
YES

 DOES THIS SYSTEM HAVE ANY COMMUNICATIONS TYPE DEVICES?
I.E., VIPS, TTYS, ETC.
YES
EVERY DEVICE MUST BE ASSIGNED TO A HARDWARE INTERRUPT LEVEL.
WE WILL PICK THE ABSOLUTE VALUE FOR ALL DEVICES, SINCE THEY MUST
(WITH ONE EXCEPTION) BE UNIQUE.  COMMUNICATIONS DEVICES, HOWEVER,
MAY SHARE HARDWARE LEVELS AND THUS TO FACILITATE THE ASSIGNMENT
YOU MUST PROVIDE A RELATIVE LEVEL FOR EACH COMMUNICATION DEVICE
CONFIGURED.   THE ANSWER TO THE NEXT QUESTION WILL
SET UP THE RANGE OF LEVELS YOU WISH TO HAVE.  THE ACCEPTABLE VALUES
ARE: 1,2,3,4.  IF YOU SPECIFY 1, THEN ALL COMMUNICATION DEVICES
WILL BE ON THE SAME HARDWARE LEVEL.  IF YOU SPECIFY 3, THEN YOU
WILL HAVE THE OPPORTUNITY TO SPREAD THE COMMUNICATION DEVICES
OVER 3 SEPARATE LEVELS (1, 2, 3), WITH THOSE DEVICES ON LEVEL
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
1 HAVING THE HIGHEST PRIORITY AND THOSE ON LEVEL 3 THE LOWEST.
HOW MANY COMMUNICATION PRIORITY LEVELS (1 TO 4) DO YOU WISH?
ENTER THE NUMBER OF LEVELS.
4
ALL COMMUNICATION TYPE DEVICES MUST BE ATTACHED TO A MULTI-LINE
COMMUNICATIONS PROCESSOR (MLCP).
HOW MANY MLCPS (1 TO 10) WILL THIS SYSTEM HAVE?
4

 WE ARE READY TO CONFIGURE ANY MLCP CONNECTED ASYNCHRONOUS TERMINAL
DEVICES (ATD) THAT YOU HAVE. THESE INCLUDE THE 7200 RUN IN TELEPRINTER
MODE AND ALL OTHER ATDS. IF YOU DO NOT HAVE ANY OF THESE DEVICES TO
CONFIGURE REPLY "NONE" TO THE FOLLOWING QUESTION.
HOW MANY ATDS DO YOU HAVE?
2

 ATD TERMINAL NUMBER     1
YOU NEED TO PROVIDE THE BUS CHANNEL NUMBER TO WHICH THIS DEVICE
HAS BEEN ATTACHED. THE CHANNEL NUMBER IS A UNIQUE FOUR CHARACTER
HEXADECIMAL VALUE WHICH ENDS WITH EITHER "00" OR "80".
YOU MUST ENTER EXACTLY FOUR CHARACTERS.
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
FD80
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
2
DEVICE TYPES SUPPORTED BY THE ASYNCHRONOUS TERMINAL DEVICE DRIVER INCLUDE
TTY, 7200, 7801, D7200 AND PRU.
WHAT IS THE DEVICE TYPE FOR THIS ATD DEVICE?
TTY

 ATD TERMINAL NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
FC00
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
3
WHAT IS THE DEVICE TYPE FOR THIS ATD DEVICE?
7801


WE ARE NOW READY TO CONFIGURE THE VIP7700S OR VIP7800S WHICH
ARE TO BE RUN IN VIP7700 EMULATION MODE.  IF YOU DO NOT HAVE ANY
VIPS TO CONFIGURE, REPLY 'NONE' TO THE FOLLOWING QUESTION.
HOW MANY VIPS DO YOU HAVE?
3

 VIP TERMINAL NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
FF00
HOW MANY VIPS ARE POLLED ON THIS CHANNEL?
0
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
2
IS THIS VIP A 'CONTROL STATION'?
YES
DOES THIS VIP HAVE A ROP (RECEIVE ONLY PRINTER)?
YES
THERE ARE 5 TYPES OF ROP: TTY33, TTY35, TN100, TN300, AND TN1200.
WHAT TYPE OF ROP DO YOU HAVE?
TN1200
DOES THIS ROP SUPPORT FORM FEED?
YES

 VIP TERMINAL NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
FF80
HOW MANY VIPS ARE POLLED ON THIS CHANNEL?
NONE
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
3
IS THIS VIP A 'CONTROL STATION'?
N
DOES THIS VIP HAVE A ROP (RECEIVE ONLY PRINTER)?
N

 VIP TERMINAL NUMBER     3
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
FE00
HOW MANY VIPS ARE POLLED ON THIS CHANNEL?
1
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
2
ENTER THE POLLING ADDRESS FOR THIS VIP
12
DOES THIS VIP HAVE A ROP (RECEIVE ONLY PRINTER)?
NO


WE ARE NOW READY TO CONFIGURE THE H3270 HOST LINKS WHICH ARE TO
BE RUN BY THE BSC LINE PROTOCOL HANDLER.  IF YOU DO NOT HAVE ANY
H3270 HOST LINKS TO CONFIGURE, REPLY NONE TO THE FOLLOWING QUESTION.
HOW MANY H3270 HOST LINKS DO YOU HAVE?
2

 H3270 LINE NUMBER     1
WHAT CHANNEL NUMBER WILL THIS LINE BE ON?
?
YOU NEED TO PROVIDE THE BUS CHANNEL NUMBER TO WHICH THIS DEVICE
HAS BEEN ATTACHED. THE CHANNEL NUMBER IS A UNIQUE FOUR CHARACTER
HEXADECIMAL VALUE WHICH ENDS WITH EITHER "00" OR "80".
YOU MUST ENTER EXACTLY FOUR CHARACTERS.
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
WHAT CHANNEL NUMBER WILL THIS LINE BE ON?
F880
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS LINE
2
ENTER THE POLLING ADDRESS (DECIMAL) FOR THIS H3270 HOST LINK
123
ENTER THE SELECT ADDRESS (DECIMAL) FOR THIS H3270 HOST LINK
234

 H3270 LINE NUMBER     2
WHAT CHANNEL NUMBER WILL THIS LINE BE ON?
F800
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS LINE
2
ENTER THE POLLING ADDRESS (DECIMAL) FOR THIS H3270 HOST LINK
124
ENTER THE SELECT ADDRESS (DECIMAL) FOR THIS H3270 HOST LINK
235


WE ARE NOW READY TO CONFIGURE THE POLLED VIP EMULATOR (PVE) GROUPS
(A PVE GROUP CONSISTS OF ALL THE PVE STATIONS ASSIGNED TO THE SAME
HARDWARE CHANNEL).  IF YOU DO NOT HAVE ANY PVE STATIONS TO CONFIGURE,
REPLY "NONE" TO THE FOLLOWING QUESTION.
HOW MANY PVE GROUPS DO YOU HAVE?
2

 PVE GROUP NUMBER    1
ENTER CHANNEL NUMBER FOR PVE GROUP    1
F900
HOW MANY PVE STATIONS ARE THERE ON THIS CHANNEL?
2
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS GROUP
2
ENTER THE POLLING ADDRESS (0 TO 31) FOR PVE STATION    1
1
ENTER THE POLLING ADDRESS (0 TO 31) FOR PVE STATION    2
2

 PVE GROUP NUMBER    2
ENTER CHANNEL NUMBER FOR PVE GROUP    2
F980
HOW MANY PVE STATIONS ARE THERE ON THIS CHANNEL?
3
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS GROUP
1
ENTER THE POLLING ADDRESS (0 TO 31) FOR PVE STATION    3
1
ENTER THE POLLING ADDRESS (0 TO 31) FOR PVE STATION    4
2
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
ENTER THE POLLING ADDRESS (0 TO 31) FOR PVE STATION    5
3

WE ARE NOW READY TO CONFIGURE THE SYNCHRONOUS TERMINAL DEVICE (STD)
GROUPS (A STD GROUP CONSISTS OF ALL THE STD STATIONS ASSIGNED TO THE
SAME HARDWARE CHANNEL).  THE POLLING SEQUENCE WITHIN A GROUP IS
DETERMINED BY THE ORDER ENTERED.
IF YOU DO NOT HAVE ANY STD STATIONS TO CONFIGURE,
REPLY "NONE" TO THE FOLLOWING QUESTION.
HOW MANY STD GROUPS DO YOU HAVE?
2

 STD GROUP NUMBER    1
ENTER CHANNEL NUMBER FOR STD GROUP    1
F680
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS GROUP
3
HOW MANY STD STATIONS ARE THERE ON THIS CHANNEL?
2
ENTER THE POLLING ADDRESS (0 TO 31) FOR STD STATION    1
23
DEVICE TYPES SUPPORTED BY THE SYNCHRONOUS TERMINAL DEVICE DRIVER
INCLUDE V7804, V7700 AND V7760
WHAT IS THE DEVICE TYPE FOR THIS STD DEVICE?
V7804
DOES THIS STD HAVE A ROP (RECEIVE ONLY PRINTER)?
YES
THERE ARE 4 TYPES OF ROP: ROSY24, ROSY26, TN300, AND TN1200.
WHAT TYPE OF ROP DO YOU HAVE?
TN300
ENTER THE POLLING ADDRESS (0 TO 31) FOR STD STATION    2
24
WHAT IS THE DEVICE TYPE FOR THIS STD DEVICE?
V7804
DOES THIS STD HAVE A ROP (RECEIVE ONLY PRINTER)?
Y
WHAT TYPE OF ROP DO YOU HAVE?
TN300

 STD GROUP NUMBER    2
ENTER CHANNEL NUMBER FOR STD GROUP    2
F600
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS GROUP
4
HOW MANY STD STATIONS ARE THERE ON THIS CHANNEL?
3
ENTER THE POLLING ADDRESS (0 TO 31) FOR STD STATION    3
23
WHAT IS THE DEVICE TYPE FOR THIS STD DEVICE?
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
V7700
DOES THIS STD HAVE A ROP (RECEIVE ONLY PRINTER)?
YES
WHAT TYPE OF ROP DO YOU HAVE?
TN1200
ENTER THE POLLING ADDRESS (0 TO 31) FOR STD STATION     4
24
WHAT IS THE DEVICE TYPE FOR THIS STD DEVICE?
V7700
DOES THIS STD HAVE A ROP (RECEIVE ONLY PRINTER)?
NO
ENTER THE POLLING ADDRESS (0 TO 31) FOR STD STATION     5
25
WHAT IS THE DEVICE TYPE FOR THIS STD DEVICE?
V7700
DOES THIS STD HAVE A ROP (RECEIVE ONLY PRINTER)?
YES
WHAT TYPE OF ROP DO YOU HAVE?
TN1200

HOW MANY MLCP CONNECTED BINARY SYNCHRONOUS COMMUNICATIONS (BSC) LINES
DO YOU HAVE?
2

 BSC LINE NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
F000
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
1
IS THIS BSC LINE A 'PRIMARY' ONE?
YES

 BSC LINE NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
F080
ENTER THE COMMUNICATIONS PRIORITY LEVEL FOR THIS TERMINAL
1
IS THIS BSC LINE A 'PRIMARY' ONE?
NO

WE ARE NOW READY TO CONFIGURE ALL MDC CONNECTED CONSOLE DEVICES.
THESE CONSOLE DEVICES ARE ALL HANDLED BY THE CONSOLE DEVICE
DRIVER (CON). IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY
"NONE" TO THE FOLLOWING QUESTION.
HOW MANY MDC CONNECTED CONSOLE DEVICES DO YOU HAVE?
2

 CON NUMBER    1
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
0400
DEVICE TYPES SUPPORTED BY THE CONSOLE DEVICE DRIVER INCLUDE
KSR, 7200, 7801, D7200 AND PRU.
WHAT IS THE DEVICE TYPE FOR THE OPERATOR'S CONSOLE?
KSR

 CON NUMBER    2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
0480
WHAT IS THE DEVICE TYPE FOR THIS CON DEVICE?
7801

 WE ARE NOW READY TO CONFIGURE ALL DISKETTE DEVICES. IF
YOU DO NOT HAVE ANY DISKETTE DEVICES, REPLY 'NONE' TO THE
FOLLOWING QUESTION.
HOW MANY DISKETTES DO YOU HAVE?
4

 DISKETTE NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
0500

 DISKETTE NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
0580

 DISKETTE NUMBER     3
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1200

 DISKETTE NUMBER     4
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1280

 WE ARE NOW READY TO CONFIGURE ALL CARTRIDGE DISK
DEVICES. IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE
FOLLOWING QUESTION.
HOW MANY CARTRIDGE DISKS DO YOU HAVE?
2

 CARTRIDGE DISK NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1400
DOES THIS CARTRIDGE DISK HAVE A FIXED PLATTER?
YES

 CARTRIDGE DISK NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1480
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
DOES THIS CARTRIDGE DISK HAVE A FIXED PLATTER?
NO


WE ARE NOW READY TO CONFIGURE ALL CARTRIDGE MODULE DISK
DEVICES. IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE
FOLLOWING QUESTION.
HOW MANY CARTRIDGE MODULE DISKS DO YOU HAVE?
2

 CARTRIDGE MODULE DISK NUMBER    1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1500

 CARTRIDGE MODULE DISK NUMBER    2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1580

 WE ARE NOW READY TO CONFIGURE ALL STORAGE MODULE
DEVICES. IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE
FOLLOWING QUESTION.
HOW MANY STORAGE MODULES DO YOU HAVE?
2

 STORAGE MODULE NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1800

 STORAGE MODULE NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1880

 WE ARE NOW READY TO CONFIGURE CARD READER DEVICES. IF YOU
DO NOT HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE FOLLOWING QUESTION.
HOW MANY CARD READERS DO YOU HAVE?
2

 CARD READER NUMBER    1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
2200

 CARD READER NUMBER    2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
YES
INVALID CHANNEL NUMBER!  MUST BE EXACTLY 4 CHARACTERS LONG!
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
2280

 WE ARE NOW READY TO CONFIGURE CARD PUNCH AND READER/PUNCH DEVICES.
NOTE THAT THIS IS EITHER A CARD READER/PUNCH OR A CARD PUNCH.
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY
"NONE" TO THE FOLLOWING QUESTION.
HOW MANY CARD READER/PUNCHES OR PUNCHES DO YOU HAVE?
NO

 WE ARE NOW READY TO CONFIGURE ALL PRINTER DEVICES. IF YOU DO NOT
HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE FOLLOWING QUESTION.
HOW MANY PRINTERS DO YOU HAVE?
1

 PRINTER NUMBER    1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1780

 WE ARE NOW READY TO CONFIGURE ALL 9-TRACK MAGNETIC TAPE DEVICES.
IF YOU DO NOT HAVE ANY TO CONFIGURE, REPLY 'NONE' TO THE FOLLOWING
QUESTION.
HOW MANY 9-TRACK MAGNETIC TAPE DEVICES DO YOU HAVE?
2

 MAGNETIC TAPE DEVICE NUMBER     1
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1700

 MAGNETIC TAPE DEVICE NUMBER     2
WHAT CHANNEL NUMBER WILL THIS DEVICE BE ON?
1900

 IF YOU WISH YOU MAY OPTIMIZE ALL FUNCTIONS FOR THE GREATEST
POSSIBLE SPEED, BUT AT THE EXPENSE OF HAVING A LARGE RESIDENT SYSTEM.
DO YOU WISH TO HAVE ALL SYSTEM OVERLAYS RESIDENT? (YES OR NO)
NO

BY MAKING PARTICULAR OPERATING SYSTEM INPUT/OUTPUT OVERLAYS PERMANENTLY
RESIDENT YOU CAN OBTAIN OPTIMUM SPEED IN SOME CATEGORIES.
DO YOU WISH TO OPTIMIZE THE READING/WRITING OF RELATIVE FILES?
YES
DO YOU WISH TO OPTIMIZE THE READING/WRITING OF RANDOM FILES?
YES
DO YOU WISH TO OPTIMIZE THE READING/WRITING OF INDEXED FILES?
YES
DO YOU WISH TO OPTIMIZE RELATIVE FILES WITH DELETABLE RECORDS?
YES
DO YOU WISH TO OPTIMIZE FILE MANAGEMENT OPEN/CLOSE FUNCTIONS?
YES
DO YOU WISH TO OPTIMIZE THE READING/WRITING OF TAPE FILES?
YES
DO YOU WISH TO OPTIMIZE EBCDIC TAPE FUNCTIONS?
YES
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
 DO YOU WISH TO CREATE A BATCH MEMORY POOL?
YES
HOW BIG A BATCH POOL? (NNNNN OR NNK WORDS)
20K


ENTER THE NUMBER (1 TO 99) OF SYSTEM OVERLAY AREAS
YOU DESIRE.
6
WOULD YOU LIKE TO SEE THE GENERATED STATEMENTS?
YES


*PRODUCED BY INTERACTIVE BUILD (VERSION 2.0), ON  1980/09/04 1145:42.5
SYS ,,SCIP,6,18,48,,E
DEVICE CON00,0,5,X'0400',CONSOLE,140,,'KSR'
LDBU CIPSIM
LDBU SIPSIM_SP
LDBU ZXPFR
* 'GCOS6/MOD 400 VERSION 2.1','TEST SITE ONE'
RLOCK
VDAM
LDBU ZNVDAM
COMM 9
ATD 3,11,X'FD80',,'HI','TTY'
DEVICE AT000,3,11,X'FD80'
STTY AT000  -RECONNECT YES
ATD 4,12,X'FC00',,'HI','7801'
DEVICE AT001,4,12,X'FC00'
STTY AT001  -RECONNECT YES
VIP 5,11,X'FF00',,,,C,6,TN1200,FU
DEVICE VIP00,5,11,X'FF00'
DEVICE ROP00,6,11,X'FF00'
VIP 7,12,X'FF80',,,,T
DEVICE VIP01,7,12,X'FF80'
VIP 8,11,X'FE00',,12
DEVICE VIP02,8,11,X'FE00'
H3270 9,11,X'F880',,X'7B',X'EA'
DEVICE H3200,9,11,X'F880'
H3270 10,11,X'F800',,X'7C',X'EB'
DEVICE H3201,10,11,X'F800'
PVE 11,11,X'F900',,1
DEVICE PVE00,11,11,X'F900'
PVE 12,11,X'F900',,2
DEVICE PVE01,12,11,X'F900'
PVE 13,10,X'F980',,1
DEVICE PVE02,13,10,X'F980'
PVE 14,10,X'F980',,2
DEVICE PVE03,14,10,X'F980'
PVE 15,10,X'F980',,3
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
DEVICE PVE04,15,10,X'F980'
STDLN 12,X'F680'
STD 16,23,,V7804
DEVICE STD00,16,12,X'F680'
ROP TN300
DEVICE STD01,16,12,X'F680'
STD 17,24,,V7804
DEVICE STD02,17,12,X'F680'
ROP TN300
DEVICE STD03,17,12,X'F680'
POLIST 1
STAPOL 23,24
STDLN 13,X'F600'
STD 18,23,,V7700
DEVICE STD04,18,13,X'F600'
ROP TN1200
DEVICE STD05,18,13,X'F600'
STD 19,24,,V7700
DEVICE STD06,19,13,X'F600'
STD 20,25,,V7700
DEVICE STD07,20,13,X'F600'
ROP TN1200
DEVICE STD08,20,13,X'F600'
POLIST 1
STAPOL 23,24,25
BSC 21,10,X'F000',,P
DEVICE BSC00,21,10,X'F000'
BSC 22,10,X'F080',,S
DEVICE BSC01,22,10,X'F080'
DEVICE CON01,23,15,X'0480',,,,'7801'
DEVICE DSK00,24,16,X'0500'
DEVICE DSK01,25,17,X'0580'
DEVICE DSK02,26,18,X'1200'
DEVICE DSK03,27,19,X'1280'
* DEVICE RCD00,1,6,X'1400'
DEVICE FCD00,28,6,X'1400'
DEVICE RCD01,29,20,X'1480'
DEVICE RCM00,30,21,X'1500'
DEVICE FCM00,31,21,X'1500'
DEVICE RCM01,32,22,X'1580'
DEVICE FCM01,33,22,X'1580'
DEVICE MSM00,34,23,X'1800'
DEVICE MSM01,35,24,X'1880'
DEVICE CDR00,36,25,X'2200'
DEVICE CDR01,37,26,X'2280'
DEVICE LPT00,38,27,X'1780'
DEVICE MT900,39,28,X'1700'
DEVICE MT901,40,29,X'1900'
MEMPOOL S,,10000,P
MEMPOOL B,,20480,P
```

Figure 4-1 (Cont). Using M4_SYSDEF

```
MEMPOOL ,AB,*,PCU
MEMPOOL ,L0,AB
MEMPOOL ,L1,AB
MEMPOOL ,L2,AB
MEMPOOL ,L3,AB
MEMPOOL ,L4,AB
MEMPOOL ,L5,AB
RESOLA OYUPX1
RESOLA OYWRX1
RESOLA OYRDX1
RESOLA OYRWX1
RESOLA OZXINV
RESOLA OZXSUV
RESOLA OZXSUB
RESOLA OYDMR3
RESOLA OYUMC
RESOLA OYUPC
RESOLA OZCINV
RESOLA OZCKEY
RESOLA OZCLNK
RESOLA OZCPSN
RESOLA OZCSUB
RESOLA OZUCCB
RESOLA OZUSUB
RESOLA OYDMT
RESOLA OZTUTL
RESOLA OZIETA
RESOLA OZTATE
RESOLA OYDMR2
RESOLA OYUPN
MAP >SID>CLM_MAP
QUIT

 DO YOU WANT TO CREATE THE CLM DIRECTIVE FILE NOW?
YES
WOULD YOU LIKE TO USE A FILE NAME OTHER THAN  CLM_USER FOR THIS FILE?
YES
ENTER YOUR FILE NAME ==
EXAMPLE A
CLM FILE NAME IS:
 ^SYSSRM>UDD>SYS_ADMIN>SANBORN>M4R2.1>EXAMPLE.SA
RESTART FILE NAME IS:
 ^SYSSRM>UDD>SYS_ADMIN>SANBORN>M4R2.1>M4RST.1
```

Figure 4-1 (Cont). Using M4_SYSDEF

Example 2:

To modify the CLM directive file created in Figure 4-1, the user invokes the M4_SYSDEF program in rebuild mode, specifying the name of the restart file on the command line. Figure 4-2 illustrates the rebuild mode dialog.

```
M4_SYSDEF M4RST.1
YOU HAVE ENTERED M4_SYSDEF IN THE REBUILD MODE.
DO YOU WANT TO SEE THE EXISTING CLM STATEMENTS?
YES



*PRODUCED BY INTERACTIVE BUILD (VERSION 2.0), ON  1980/09/05 1427:45.3
SYS ,,SCIP,6,18,48,,F
DEVICE CON00,0,5,X'0400',CONSOLE,140,,'KSR'
LOBU CTPSIM
LOBU STPSIM_SP
LOBU ZXPES
* 'GCOS6/MOD 400 VERSION 2.1','PEPPERELL MA.'
BLOCK
VDAM
LOBU ZBVDAM
COMM 9
AID 3,11,X'FD80',,'HI','TTY'
DEVICE AT000,3,11,X'FD80'
STTY AT000  -RECONNECT YES
AID 4,12,X'FC00',,'HI','7601'
DEVICE AT001,4,12,X'FC00'
STTY AT001  -RECONNECT YES
VIP 5,11,X'FF00',,,,C,6,TN1200,F0
DEVICE VIP00,5,11,X'FF00'
DEVICE ROP00,6,11,X'FF00'
VIP 7,12,X'FF80',,,,T
DEVICE VIP01,7,12,X'FF80'
VIP 8,11,X'FF00',,12
DEVICE VIP02,8,11,X'FF00'
H3270 9,11,X'FA80',,X'7A',X'FA'
DEVICE H3200,9,11,X'FA80'
H3270 10,11,X'FA00',,X'7C',X'EA'
DEVICE H3201,10,11,X'FA00'
PVE 11,11,X'F900',,1
DEVICE PVE00,11,11,X'F900'
PVE 12,11,X'F900',,2
DEVICE PVE01,12,11,X'F900'
```

Figure 4-2. Using M4_SYSDEF In Rebuild Mode

```
PVE 13,10,X'F9A0',,1
DEVICE PVE02,13,10,X'F9A0'
PVE 14,10,X'F9A0',,2
DEVICE PVE03,14,10,X'F9A0'
PVE 15,10,X'F9A0',,3
DEVICE PVE04,15,10,X'F9A0'
STDLN 12,X'F6A0'
STD 16,23,,V7804
DEVICE STD00,16,12,X'F6A0'
ROP TN300
DEVICE STD01,16,12,X'F6A0'
STD 17,24,,V7804
DEVICE STD02,17,12,X'F6A0'
ROP TN300
DEVICE STD03,17,12,X'F6A0'
POLIST 1
STAPOL 23,24
STDLN 13,X'F600'
STD 18,23,,V7700
DEVICE STD04,18,13,X'F600'
ROP TN1200
DEVICE STD05,18,13,X'F600'
STD 19,24,,V7700
DEVICE STD06,19,13,X'F600'
STD 20,25,,V7700
DEVICE STD07,20,13,X'F600'
ROP TN1200
DEVICE STD08,20,13,X'F600'
POLIST 1
STAPOL 23,24,25
HSC 21,10,X'F000',,2
DEVICE HSC00,21,10,X'F000'
HSC 22,10,X'F0A0',,3
DEVICE HSC01,22,10,X'F0A0'
DEVICE CON01,23,15,X'0480',,,,'7A01'
DEVICE DSK00,24,16,X'0500'
DEVICE DSK01,25,17,X'0580'
DEVICE DSK02,26,18,X'1200'
DEVICE DSK03,27,19,X'1280'
* DEVICE RCD00,1,6,X'1400'
DEVICE FCD00,28,6,X'1400'
DEVICE RCD01,29,20,X'1480'
DEVICE RCM00,30,21,X'1500'
DEVICE FCM00,31,21,X'1500'
DEVICE RCM01,32,22,X'1580'
DEVICE FCM01,33,22,X'1580'
DEVICE MSM00,34,23,X'1800'
DEVICE MSM01,35,24,X'1880'
DEVICE CDR00,36,25,X'2200'
DEVICE CDR01,37,26,X'2280'
```

Figure 4-2 (Cont). Using M4_SYSDEF In Rebuild Mode

```
DEVICE LPT00,38,27,X'1780'
DEVICE MT900,39,28,X'1700'
DEVICE MT901,40,29,X'1900'
MEMPOOL S,,10000,P
MEMPOOL 8,,20480,P
MEMPOOL ,AB,*,PCU
MEMPOOL ,L0,AB
MEMPOOL ,L1,AB
MEMPOOL ,L2,AB
MEMPOOL ,L3,AB
MEMPOOL ,L4,AB
MEMPOOL ,L5,AB
RESOLA OYUPX1
RESOLA OYWRX1
RESOLA OYROX1
RESOLA OYRWX1
RESOLA OZXINV
RESOLA OZXSOV
RESOLA OZXSUR
RESOLA OYDMR3
RESOLA OYDMC
RESOLA OYUPC
RESOLA OZCINV
RESOLA OZCKEY
RESOLA OZCLNK
RESOLA OZCPSN
RESOLA OZCSUR
RESOLA OZUCCR
RESOLA OZUSUR
RESOLA OYOMT
RESOLA OZTUTL
RESOLA OZTETA
RESOLA OZTATE
RESOLA OYDMR2
RESOLA OYOPN
MAP >STD>CLM_MAP
QUIT

DO YOU WISH TO MODIFY DEVICES OR SYSTEM PARAMETERS?
RESPOND WITH "SYSTEM", "DEVICES", "VIEW", OR "DONE".
RESPONSE =
DEVICES

RESPOND WITH THE DEVICE TYPE OR CLASS THAT YOU WISH TO MODIFY.
"COMM", "CON", "DSK", "RCD", "MSM", "CMO", "CDR",
"CRP", "LPT", "MTP", OR "DONE".
RESPONSE =
COMM

DO YOU WISH TO MODIFY THE COMMUNICATIONS LEVELS AND THE NUMBER OF
MLCP'S (COMM), THE VIP'S (VIP), THE HOST LINKS (H3270), THE
PVES (PVE), THE SYNCHRONOUS TERMINALS (STD),
THE ASYNCHRONOUS TERMINALS (ATD), OR THE BSC'S (BSC)?
RESPONSE =
COMM
```

Figure 4-2 (Cont). Using M4_SYSDEF In Rebuild Mode

```
DOES THIS SYSTEM HAVE ANY COMMUNICATIONS TYPE DEVICES?
I.E., VIPS, TTYS, ETC.
NO

DO YOU WISH TO MODIFY THE COMMUNICATIONS LEVELS AND THE NUMBER OF
MLCP'S (COMM), THE VIP'S (VIP), THE HOST LINKS (H3270), THE
PVES (PVE), THE SYNCHRONOUS TERMINALS (STD),
THE ASYNCHRONOUS TERMINALS (ATD), OR THE BSC'S (BSC)?
RESPONSE =
DONE

RESPOND WITH THE DEVICE TYPE OR CLASS THAT YOU WISH TO MODIFY.
"COMM", "CON", "DSK", "RCD", "MSM", "CMD", "CDR",
"CRP", "LPT", "MTP", OR "DONE".
RESPONSE =
DONE

DO YOU WISH TO MODIFY DEVICES OR SYSTEM PARAMETERS?
RESPOND WITH "SYSTEM", "DEVICES", "VIEW", OR "DONE".
RESPONSE =
SYSTEM

DO YOU WISH TO MODIFY THE PROMPT (PROMPT), MEMORY SIZE (MEMORY), THE
NUMBER OF INTERACTIVE USERS (USERS), THE AUTOMATIC GENERATION OF
SPD NAMES (SPD),THE AUTOMATIC CHANNEL ASSIGNMENT (CHANNEL), THE
RECORD LOCKING FEATURE (RLOCK), THE SYSTEM OVERLAY AREAS (SOAT), THE
RESIDENT OVERLAYS (RESOLA), THE DISPLAY FORMATING FEATURE (VOAM),
THE COMMERCIAL OR SIP SIMULATORS (SIMUL),
OR THE SYSTEM AND INSTALLATION IDENTIFICATION (IDENT)?
RESPONSE =
RESOLA


IF YOU WISH YOU MAY OPTIMIZE ALL FUNCTIONS FOR THE GREATEST
POSSIBLE SPEED, BUT AT THE EXPENSE OF HAVING A LARGE RESIDENT SYSTEM.
DO YOU WISH TO HAVE ALL SYSTEM OVERLAYS RESIDENT? (YES OR NO)
YES

DO YOU WISH TO MODIFY THE PROMPT (PROMPT), MEMORY SIZE (MEMORY), THE
NUMBER OF INTERACTIVE USERS (USERS), THE AUTOMATIC GENERATION OF
SPD NAMES (SPD),THE AUTOMATIC CHANNEL ASSIGNMENT (CHANNEL), THE
RECORD LOCKING FEATURE (RLOCK), THE SYSTEM OVERLAY AREAS (SOAT), THE
RESIDENT OVERLAYS (RESOLA), THE DISPLAY FORMATING FEATURE (VOAM),
THE COMMERCIAL OR SIP SIMULATORS (SIMUL),
OR THE SYSTEM AND INSTALLATION IDENTIFICATION (IDENT)?
RESPONSE =
DONE

DO YOU WISH TO MODIFY DEVICES OR SYSTEM PARAMETERS?
RESPOND WITH "SYSTEM", "DEVICES", "VIEW", OR "DONE".
RESPONSE =
DONE
WOULD YOU LIKE TO SEE THE GENERATED STATEMENTS?
YES
```

Figure 4-2 (Cont). Using M4_SYSDEF In Rebuild Mode

```
*PRODUCED BY INTERACTIVE BUILD (VERSION 2.0), ON  1980/09/05 1434:38.3
SYS ,,SCIP,6,18,48,,E
DEVICE CON00,0,5,X'0400',CONSOLE,140,,'KSR'
LDBU CIPSIM_
LDBU STPSIM_SP
LDBU ZXPFH
* 'GCOS6/MOD 400 VERSION 2.1','PEPPERELL MA.'
BLOCK
VDAM
LDBU ZNVDAM
DEVICE CON01,23,15,X'0480',,,,'7801'
DEVICE DSK00,24,16,X'0500'
DEVICE DSK01,25,17,X'0580'
DEVICE DSK02,26,18,X'1200'
DEVICE DSK03,27,19,X'1280'
* DEVICE RCD00,1,6,X'1400'
DEVICE FCD00,28,5,X'1400'
DEVICE RCD01,29,20,X'1480'
DEVICE RCM00,30,21,X'1500'
DEVICE FCM00,31,21,X'1500'
DEVICE RCM01,32,22,X'1580'
DEVICE FCM01,33,22,X'1580'
DEVICE MSM00,34,23,X'1800'
DEVICE MSM01,35,24,X'1880'
DEVICE CDR00,36,25,X'2200'
DEVICE CDR01,37,26,X'2280'
DEVICE LPT00,38,27,X'1780'
DEVICE MT900,39,28,X'1700'
DEVICE MT901,40,29,X'1900'
MEMPOOL S,,10000,P
MEMPOOL B,,20480,P
MEMPOOL ,AB,*,PCU
MEMPOOL ,L0,AB
MEMPOOL ,L1,AB
MEMPOOL ,L2,AB
MEMPOOL ,L3,AB
MEMPOOL ,L4,AB
MEMPOOL ,L5,AB
CLMIN RESOLA
MAP >STO>CLM_MAP
QUIT

DO YOU WANT TO CREATE THE CLM DIRECTIVE FILE NOW?
YES
WOULD YOU LIKE TO USE A FILE NAME OTHER THAN  CLM_USER FOR THIS FILE?
YES
ENTER YOUR FILE NAME --
EXAMPLE7A
CLM FILE NAME IS:
 ^SYSSRM>UDD>SYS_ADMIN>SANBORN>DON>EXAMPLE7A
RESTART FILE NAME IS:
 ^SYSSRM>UDD>SYS_ADMIN>SANBORN>DON>MANST.2
```

Figure 4-2 (Cont). Using M4_SYSDEF In Rebuild Mode

SECTION V

BASIC CLM DIRECTIVES

This section describes the CLM directives used to define the basic characteristics of the system. The CLM directives for the optional communications environment of the system are described in Section VI. If you are a first-time user, you should be thoroughly familiar with all the topics discussed at the beginning of this section before you create your CLM directive file.

Appropriately coded CLM directives (including those for communications, if applicable) must be placed in the CLM_USER file under directory SID on the bootstrap volume before a stage 2 or stage 3 system startup is performed. (The stages of system startup are described. (The stages of system startup are described in Section III).

Table 5-1 summarizes the basic CLM directives, which are described in alphabetic order later in this section. Briefly, these CLM directives define the following aspects of the system:

- System variables such as power line frequency, scan cycle of real-time clock, Scientific and/or Commercial Instruction Processor options, number of system overlay areas, number of trap save areas, number of intermediate request blocks, and expanded error message code option (SYS directive).

- Characteristics and sizes of system memory pool, online memory pools, and batch memory pool (if any) (MEMPOOL directives).

- Physical and logical characteristics of MDC-connected peripheral devices (DEVICE directives).

- System overlays that are to be made permanently resident (RESOLA directives).

- Operating system extensions that are to be made permanently resident (LDBU directives).

- System physical attributes, system structures, device status, user-defined variables, and memory pool specifications (MAP directive).

- Single character expressions that may be used in certain directives instead of constant values (VARIABLE directives).

- User-written device drivers that are to be made permanently resident and characteristics of the devices driven by them (DRIVER directives).

Table 5-1. Summary of Basic CLM Directives

Directive	Meaning
CLMIN	Causes subsequent CLM directives to be read from a different device/file.
* (Comment)	Used to include a comment in the file of CLM directives.
DEVICE	Identifies a peripheral device and its characteristics, both physical and logical.
DRIVER	Identifies a user-written device driver and one device driven by it.
LDBU	Identifies a bound unit (usually user-written) that is to be made permanently resident in the system area of memory.
MAP	Provides configuration information that allows you to verify the success of certain aspects of the system building process.
MEMPOOL	Defines the system memory pool, the batch memory pool, or one or more online memory pools.
QUIT	Indicates the end of the input file of CLM directives.
RESOLA	Identifies one or more system overlays that are to be made permanently resident in the system area of memory.
RLOCK	Defines a pool of structures to be used by data management when record locking is requested.
SYS	Defines a number of system variables.
VARIABLE	Allows you to define variables for parameters in certain CLM directive statements.

RULES FOR ARRANGING CLM DIRECTIVES

Only a few rules apply to the order of basic directives in the input file read by the Configuration Load Manager. These rules are listed below. (Rules governing the order of communications-related CLM directives are described near the beginning of Section VI).

1. Each communications (MDC-connected) peripheral device in the system must be identified in a separate DEVICE directive. (As described in Section VI, communications devices must be identified in DEVICE directives only if they are to be accessible through the file system).

2. If the operator terminal is connected to an MDC, the DEVICE directive for the operator terminal should be the first CLM directive. This arrangement allows the operator terminal to be available for possible error messages related to subsequent CLM directives.

3. If your configuration requires more than the default number of trap save areas (7) and/or more than the default number of intermediate request blocks (20), a SYS directive with appropriate tsa and irb argument values should be placed as close as possible to the beginning of the file of CLM directives. The addtional trap save areas and/or intermediate request blocks are available as soon as the SYS directive is read.

4. If the CLMIN directive is used, it must not precede the DEVICE directive that identifies the device implied by the pathname argument of the CLMIN directive.

5. If LDBU directives are used to identify both a SIP Simulator and a Commercial Simulator, the simulator identified in the latter LDBU directive will be the first to process a trap to the trap vector shared by the two simulators.

6. A variable must be defined by a VARIABLE directive in the CLM file before it can be used in subsequent directives.

7. QUIT must be the last CLM directive in the input file. Any directives that follow QUIT wil not be processed.

FORMAT OF CLM DIRECTIVES

A CLM directive consists of a string of up to 72 ASCII characters. The format of a CLM directive is shown below:

```
mnemonic  argument_1[,argument_2][,argument_3]...[,argument_n]
```

In the directive descriptions in this section and in Section VI, the following conventions apply:

1. The directive mnemonic is shown in uppercase. It must be specified exactly as shown.

2. Exactly one space must separate the directive mnemonic from the first argument.

3. In arguments, lowercase words and letters indicate values that must be specified by the user.

4. Arguments within square brackets [] are optional.

5. Vertically stacked arguments within braces {} represent options, one of which must be chosen when the argument is used.

6. Arguments in a directive are separated by commas. Embedded blanks are not allowed within arguments. Blanks are not allowed between arguments.

7. A blank terminates the list of arguments in a directive. Anything following this blank is considered a comment.

8. All arguments are positional, meaning that they must be specified in the order shown in the directive description. If a given argument is omitted and a following argument is used, the omitted argument must be signified by delimiting commas (e.g., DEVICE LPT02,15,22, X'1380',,,N).

9. Trailing commas are not required after the last argument.

10. Continuation lines are not allowed.

11. Unless stated otherwise, unsigned positive integers less than or equal to 65,535 (10) (FFFF (16)) can be expressed in decimal or hexadecimal.

    If a value exceeds 65,535 (10), it must be expressed in double-word hexadecimal format (i.e., D'hhhhhhhh').

    A decimal integer consists of one or more decimal digits (e.g., 1234).

    A hexadecimal channel number consists of the constant "X" followed by four hexadecimal digits expressed in the format: X'h...' (e.g., X'FF80').

12. If an ASCII character string <u>begins</u> with a decimal digit, the string must be enclosed within apostrophes (e.g., '1A').

13. The line length specified for a terminal or other input device at system building time overrides the command processor line length of 127 characters. For example, at a terminal configured with a line length of 80 characters, you cannot enter a continuous command line longer than 80 characters unless you type an ampersant (&) at the end of the line and then continue on one or more additional lines. You can alter

the terminal's line length by using the set terminal characteristics (STTY) command (described in the Commands manual) after system initialization or by modifying the STTY directive in your CLM file.

14. For each noncommunications device configured, you have the option of specifying a channel number or allowing the system to dynamically assign or "float" channel assignments at startup time (1). To float a channel, you specify the channel number parameter in the appropriate DEVICE or DRIVER directive with a single zero (0). The system then assigns appropriate devices to any available channels. Note that if you specify a zero channel and the system cannot locate an appropriate device for it, no error message is issued.

M4_SYSDEF, the interactive CLM directive generation program, allows you to float channel numbers for all noncommunications devices, if you so choose. See Section IV for details.

Channel numbers are assigned according to the order in which DEVICE or DRIVER directives with zero (0) channels are specified in the CLM file. To prevent devices with zero channels from inadvertently preempting devices whose channel numbers are explicitly specified, all devices of a given type with explicitly specified channel numbers should precede all devices of that type with zero channels in the CLM file.

For noncommunications devices, the first available device having the lowest bus address and the correct device identification is assigned to the first available MDC-connected DEVICE or DRIVER directive encountered in the CLM file, and so on.

If a system is configured with devices containing zero channel numbers, you may determine all channel numbers assigned by the system after system initialization. The system operator can issue the command STS -ALL to list each device and its channel number assignment.

Specifying a DEVICE KSR directive with a zero channel number causes the system to search for the first available MDC-connected keyboard/typewriter terminal. If none is available, the system will assign an MDC-connected CRT/kewboard terminal to the KSR driver. Note that the KSR driver is capable of operating only in command mode and single character mode.

Specifying a DEVICE CON directive with a zero channel number causes the system to search for the first available MDC-connected CRT/keyboard terminal. If none is available, the system assigns an MDC-connected keyboard/typewriter terminal to the CON driver, but

(1)
By assigning floating channel numbers to devices in the CLM file, the system builder need not know the channel numbers at each installation that will receive a copy of the system software.

only if the "terminal-type" parameter (the eighth parameter in the DEVICE directive) is specified as KSR or TTY. Note that the CON driver operates in field and command modes only.

The system identifies the presence of a device adapter and assumes that the corresponding channel numbers are valid and usable. In such a case, the system may associate an unusable channel with a physical device. If no device is available to satisfy the directive, the user is not informed. If a system included device adapters but not associated devices, then explicit channel numbers should be assigned to all devices within the device class.

PRELIMINARY CONSIDERATIONS

The following paragraphs describe:

- How to incorporate overlays and user-written bound units into your configuration.
- Memory allocation and usage.
- Performance considerations.
- How to identify your peripheral devices.
- How to configure the operator terminal.
- File system pathnames in CLM directives.

You should review each of these topics before you create your CLM directive file.

System Overlays

Within the system software, many system functions are implemented as overlays. Each system overlay runs in a 512-word system overlay area in the system area of memory. By default, one 512-word system overlay area is created. If you wish, you may use the olan argument of the SYS directive to create additional 512-word system overlay areas (up to maximum of 99) so that multiple system overlays can be co-resident at any time. In this case, the system will not be forced to repeatedly change the contents of the single system overlay area as various system overlay functions are required.

In addition to, or instead of creating multiple system overlay areas, you may wish to make certain frequently used system overlays permanently resident in the system area of memory (increassing its size at the expense of memory available for user tasks). This can be done by means of the RESOLA directive. Each RESOLA directive allows specification of one or more system overlays, each of which is loaded for permanent residancy during system startup. The names and approximate sizes of all system overlays are listed in Appendix B.

System Extensions

A system extension is an optional, user-written or system-supplied bound unit (consisting of re-entrant code) that is identified in an LDBU directive and loaded, for permanent residency in the system area of memory, during system configuration (1). Multiple system extensions can be used.

System extensions are most efficient when they are more or less continuously used through symbolic references from multiple application task groups and this usage does not permit a possible delay in the initial loading of the extension.

System extensions may be used to define system-wide global address symbols. These symbols must have been defined in EDEF Linker directives as each extension bound unit was linked. Later, as the bound unit is loaded during system configuration (by virtue of the LDBU directive), the system-wide global address symbols are added to the system symbol table, where they can be used (by the system loader) to resolve any unresolved references to them occurring in <u>subsequently loaded</u> bound units.

Any dynamic requests for memory originating from a permanently resident system extension are fulfilled from the memory pool of the task group that is using the extension.

Note that if system extension code is <u>not</u> concurrently used by multiple applications, it can be individually linked into applicable user-written bound units and not loaded for permanent residency at system startup. In this case, an LDBU directive is not used, and the extension will reside in memory only as part of each user-written bound unit with which it has been linked. Appropriate symbols in system extensions used in this way can be identified in XDEF Assembler control statements and thus resolved at link time.

Memory Allocation and Usage

System startup allows main memory to be divided into the following areas (as a maximum):

- System area
- System memory pool
- Online memory pools
  - Exclusive
  - Nonexclusive
- Batch memory pool

(1)
The SYS Linker directive must be used when the system extension is linked.

The system area contains resident system software, one or more system overlay areas (as specified by the olan argument of the SYS directive), any permanently resident system extensions (as specified in LDBU directives), and any user-written device drivers (as specified in DRIVER directives). The system memory pool, online memory pools, and batch memory pool are described in separate MEMPOOL directives (the batch memory pool is optional). Each MEMPOOL directive establishes the type and size of one or more memory pools. In the case of online memory pools, the MEMPOLL directive also establishes (1) the 2-character identity of each memory pool defined therein and (2) whether, for each memory pool defined, additional memory can be obtained, as necessary, from the batch memory pool (if any).

Roots of bound units linked as sharable are loaded into the system memory pool, whereas roots of bound units linked as non-sharable are loaded into an online memory pool or into the batch memory pool. Each task group is associated with one and only one memory pool, but more than one task group may be associated with the same memory pool. See the <u>System Comcepts</u> manual for a more detailed description of memory pools.

ACCESS TO MEMORY POOLS

Following a system startup, all nondisk devices and files are initially accessible to task groups in online memory pools and in the batch memory pool (if any). The MFA (modify file) command can be used (in an online memory pool task group) to make a nondisk device or file inaccessible to the batch memory pool. However, when a subsequent system startup is performed, all nondisk devices and files are once again accessible to task groups in online memory pools and in the batch memory pool (if any).

FRAGMENTATION

One memory-related phenomenon, "fragmentation," warrants mention here. Fragmentation is the development, within a memory pool, of unusable "holes" or "fragments" of memory. It usually results from the dynamic acquisition and release of memory within a pool by a number of concurrently active tasks. Since the system's memory manager satisfies dynamic requests for memory within a pool on a first-come/first-served basis in consecutive multiples of 32-word blocks, it is possible that the <u>order</u> of memory acquisition and release may create a situation wherein a request for memory cannot be satisfied even though sufficient free memory exists.

This situation might occur, for example, when a task has successfully obtained four areas of memory within a pool and the pool's free memory is nearly exhausted. If the task released the <u>second</u> area (e.g., 128 words) and the <u>fourth</u> area (e.g., 96 words), the task's subsequent request for 192 words might be unsuccessful because that much <u>consecutive</u> memory is not available (even though at least 224 words — the amount released — is available in two nonconsecutive areas).

Note that if tasks obtain and release memory on a last-obtained/first-release basis, the potential for fragmentation is reduced, especially when only a few tasks share a particular memory poll.

The potential for fragmentation increases with the number of tasks sharing a memory pool and with these tasks's dynamic use of memory. In general, the sytem memory pool becomes fragmented before user memory pools do. Calculations for memory pool size later in this section include a factor to allow for potential fragmentation.

<u>Performance Considerations</u>

- Systems can be configured on hardware having an MMU without utilizing the MMU. This is accomplished by use of CLM MEMPOOL directives. If no memory pools are protected or contained, and no batch pool exists (or an existing batch pool is specified as not contained), then the features of the MMU cannot be used.

- In a given configuration, if any single nonexclusive pool has protect and/or contain attributes, all nonexclusive pools have the same attributes.

- MMU Inrush Overhead

  MMU inrush is defined as the load of MMU by firmware at context restore time.

  - If pools other than the system pool are protected and/or if pools other than the batch pool are contained, there will be an MMU inrush each time any task except a system task is scheduled. On a system with an MMU, the inrush will take 19 microseconds for SAF and 37 microseconds for LAF.

  - If only the system pool is protected and/or only the batch pool is contained, there is no MMU inrush overhead. A single load of the MMU occurs during system initialization and the MMU contents are not changed unless the system is reconfigured. (The only exception is that the MMU contents are temporarily changed at batch rollout).

  - If the system is configured without orotection or containment, there is no inrush overhead.

  - For configurations in which MMU inrushes occur, a data structure of 32 words for SAF (64 words for LAF) is required for each protected or contained pool and for the set of nonprotected noncontained pools.

Identifying Peripheral Devices

Each communications (MDC-connected) peripheral device in the system must be identified in a separate DEVICE directive. The device may be designated as accessible through the file system or accessible only through physical input/output. (As described in Section VI, communications devices must be identified in DEVICE directives only if they are to be accessible through the file system).

The DEVICE directive's arguments specify the characteristics of the device:

- Unique device unit identifier
- Priority level number
- Channel number on Megabus
- Unique logical resource number
- Unique device name for file system references
- Maximum record size
- Buffered/unbuffered input/output

Configuring the Operator Terminal

The following paragraphs describe characteristics of the operator terminal and the various configuration options available to you.

OPERATOR TERMINAL CHARACTERISTICS

The operator terminal is the single device designated as the control terminal for the system. It must be assigned to lrn 0 at system configuration time in order to be identified by the system.

The operator terminal is the only terminal from which communication with the system task group $S is possible. Capabilities unique to the operator terminal include the creation of the batch task group, suspension and reactivation of batch and online task groups, peripheral device control, print/punch daemon startup, and the monitoring of system status.

Normally, an operator terminal is configured during a stage 2 or stage 3 system startup. (An operator terminal is always configured automatically in a stage 1 system startup). The operator terminal may be a terminal connected to an MLCP/DLCP.

CONFIGURATION OPTIONS

As the system builder, you have three options with respect to configuring the operator terminal:

- You can configure a terminal that will function as the operator terminal for the life of the system.

- You can configure a dual-purpose terminal that can funtion alternately as an operator terminal and as a non operator terminal.

- You can choose not to configure an operator terminal.

You might choose the first option if you wish to always have one terminal reserved for the operator's exclusive use. This terminal would always have the capability to perform those control and administrative functions unique to the operator. Note that the operator terminal cannot support forms processing software.

The second configuration option allows you to assign a dual function and identify to the operator terminal. You assign two unique names to the terminal: a "device name" that identifies the terminal when it is functioning as a non-operator terminal, and an "operator terminal name" that identifies the terminal when it is functioning as an operator terminal. You might choose this configuration option if operator functions will be infrequently performed and you wish to free the operator terminal for user applications (especially forms processing). The ability to perform forms processing will only be available to users when the terminal is functioning as a non-operator terminal.

Operating procedures for an operator's console configured by one of the above methods are fully described in the <u>Operator's Guide.</u>

The third configuration option does not permit commands and procedures unique to the operator terminal to be performed. You might choose this option if your system configuration will run under a dedicated application.

Configuring an Operator Terminal to Function for the Life of the system.

A device configured to function as the operator terminal for the life of the system will always allow users the capability to perform those control and administrative functions unique to the operator.

If the permanently assigned operator terminal is connected to an MDC, the operator terminal must be identified by a DEVICE directive. This DEVICE directive should be <u>first</u> in the CLM directive file so that the operator terminal is available to receive any error message that relates to subsequent CLM directives.

If the permanently assigned operator terminal is connected to an MLCP/DLCP, both a DEVICE directive and an appropriate communication directive (e.g., ATD) are required. (A COMM directive is also required). An MLCP/DLCP-connected operator terminal is activated only after system startup is complete.

In either case, the logical resource number on the DEVICE directive must be explicitly specified as 0. Any appropriate level number and channel number may be specified. The record size argument should normally be set at 140 bytes (characters). Note that the specified record size affects only the file system's use of the operator terminal: regardless of the value of record_size, the Operator Interface Manager always uses 140 characters as the maximum for its control of input and output operations to the operator terminal. The device name of the operator terminal may be any user-selected name (customarily, CONSOLE is used).

Example 1:

```
DEVICE CON00,0,8,X'0500',CONSOLE,140,N,'7200'
```

In this example, the operator terminal is conencted to an MDC. The device unit name is CON00. The logical resource number is 0 (a requirement for the operator terminal). The priority level is 8. The channel number is 0500. The device name is CONSOLE. The record size for file system access is 140 bytes. I/O to the device will be unbuffered. The terminal is a VIP7200. The default for visual display mode, 'T', has been accepted.

Example 2:

```
COMM 9
DEVICE TTY00,0,11,X'FF80',CONSOLE,140
TTY 0,11,X'FF80'
```

In this example, the operator terminal is connected to an MLCP. The device unit name is TTY00. The logical resource number is 0 (a requirement for the operator terminal). The priority level is 11. The channel number is FF80. The device name is CONSOLE. The record size for file system access is 140 bytes. In the COMM directive, 9 is the priority level at which the MLCP will interrupt the central processor. In the TTY directive, default values have been accepted for modem type and line speed.

Configuring a Dual-Purpose Operator Terminal

You have the option to configure a dual-purpose operator terminal. A dual-purpose terminal can function alternatively as an operator terminal and as a non-operator terminal. Only one device in your configuration may be configured in this manner. Users control in which capacity the terminal functions (as an operator terminal or non-operator terminal) by means of the OPER command described later in this section.

If the device that will function alternatively as an operator terminal and a non-operator terminal is connected to an MDC, the device must be identified by a DEVICE directive with the first parameter as CONnn. This DEVICE directive should be the first in the file of CLM directives so that the terminal is available to receive any error message that relates to subsequent CLM directives.

For an MDC-connected device, the directive is of the form:

```
DEVICE  CONnn,lrn,level,X'channel',device_name, record_size,
        N,terminal_type, [{C}] ,operator_terminal_name
                          [{T}]
```

If the device is connected to an MLCP/DLCP, both a DEVICE directive and an ATD directive are required. (A COMM directive is also required). An MLCP/DLCP-connected operator/non-operator terminal is activated only after system startup is complete. It is recommended that the terminal not be a dialed line.

For an MLCP-connected device, the ATD and DEVICE directives are of the form:

```
DEVICE  ATDnn,lrn,level,X'channel',device_name,rec_size,
        N,,,operator_terminal_name

ATD     lrn,level,X'channel',modem,speed,device_type,[del],
        [stop-bit][,parity]
```

In either case, the logical resource number (lrn) on the DEVICE directive must be explicitly specified and must be non-zero. The lrn you specify will be assigned to the device and the file having the name "device-name" (fifth parameter). The system implicitly assigns lrn 0 to the file having the name "operator terminal_name" (tenth parameter). Lrn 0 may not be assigned to any other device. An appropriate level number and channel number may be specified. The device name (fifth parameter) you specify will be assigned to the terminal when it is functioning as a non-operator terminal. An application task group's standard I/O files (e.g., command-in, user-in) will be assigned to the device name you specify.(1) You must specify that this device is unbuffered: the seventh parameter must be N.

For an MDC-connected terminal, the tenth parameter in the DEVICE directive designates this device as being able to function as both an operator terminal and a non-operator terminal. For an MLCP/DLCP-connected terminal, the tenth-parameter, operator terminal_name, must be separated from the seventh parameter, N, by exactly 3 commas. The operator terminal name you specify will be assigned to users' standard I/O files when the terminal is functioning as an operator terminal. CONSOLE is the curtomary designation.(2)

(1)
Only one application task group can be assigned to the device name you specify.

(2)
Any number of application task groups can be assigned to the operator terminal name you specify.

Example 1:

```
DEVICE  CON00,9,12,X'1380',VIP01,100,N,'7200',,CONSOLE
```

In this example, the operator terminal is connected to an MDC. The device unit name is CON00. The logical resource number (lrn) is 9. The explicitly stated lrn value of 9 is assigned to the terminal when it functions as a non-operator terminal. An lrn of 0 is implicitly assigned to the terminal when it functions as an operator terminal. The priority level is 12. The channel number is 1380. The name assigned to the terminal is VIP01. The record size for file system access is 100 bytes. The seventh parameter must be N, specifying that this device is unbuffered. The terminal type for this device is 7200. The default for visual display mode has been accepted. The name assigned to the terminal when it is functioning as an operator terminal is CONSOLE.

Example 2:

```
COMM 8
DEVICE ATD00,9,15,X'FF80',VIP00,80,N,,,CONSOLE
ATD 9,15,X'FF80',0,9600,7200
```

In this example, the operator terminal is connected to an MLCP. The device unit name is ATD00. The logical resource number is 9. The explicitly stated lrn value of 9 is assigned to the terminal when it functions as a nonoperator terminal. An lrn of 0 is implicitly assigned to the terminal when it functions as an operator terminal. The priority level is 15. The channel number is FF80. In the COMM directive, 8 is the priority level at which the MLCP/DLCP will interrupt the central processor.

In the DEVICE directive, the name assigned to the device when it is functioning as a non-operator terminal is VIP00. The record size for file system access is 80 bytes. The terminal must be designated as unbuffered (the seventh parameter must be N). The name assigned to the terminal when it is functioning as an operator terminal is CONSOLE.

In the ATD directive, the modem type is 0 (direct connect). The line speed is 9600. The device type is 7200. Defaults have been taken for del, stop-bit, and parity.

At system startup, the terminal is running under the system task group and functions as an operator terminal. The initial switch from operator control to user control <u>must</u> be accomplished through an EC file you create. The location and purpose of this EC file should be communicated only to selected users.

The EC file must contain the following commands:

OPER -OFF $\begin{bmatrix}\text{-IM}\\ \text{-DEF}\end{bmatrix}$

```
SG group_id user_id level user_in -WD path -POOL pool_id
```

The OPER -OFF command transfers control of the terminal from operator control to user control. The -IM control argument (the default) allows all messages processed by the Operator Interface Manager (OIM) to be displayed while the terminal is running under an application task group.

The -DEF control argument defers all OIM-processed messages until the terminal is returned to system task group control. However, messages sent to the terminal by two system routines, OPMSG and OPRSP, are never deferred. These messages are displayed immediately to allow the operator or user to respond to volume mount requests or 'device not ready' messages. Messages sent to the terminal from other terminals using the MSG command are also not deferred. If a user specifies the -DEF argument and the OIM has deferred an output order, the OIM will issue a message at regular intervals stating that output is pending.

The spaw group (SG) command initiates the appropriate user group for the application. The spawn group command creates an application task group whose lead task must be the command processor. (The CG and EGR commands may be substitued for the SG command). the user-in file must be the same as the 'device-name' specified in the DEVICE directive when the terminal was configured.

If a user wishes the terminal to function as an operator terminal, he must execute the command OPER -ON. A message will appear at the terminal stating "OPERATOR HAS CONTROL". If at any time, a user wishes to return control of the terminal to his application task group, he must type the command OPER -OFF.

System Configured Without Operator Terminal

A system configured without an operator terminal has the following characteritics: (1)

- A $OPMSG (operator information message) or $OPRSP (operator response message) macro call to the nonexistent operator terminal results in a 0802 error return (invalid logical resource number) to the issuing application.

- No operator terminal is available to record system error messages, ready device messages, and mount volume messages. As a result, error returns to the issuing application occur immediately:

  - An input/output order to a device that is offline or not ready results in a 0105 error return (device not ready) to the application.

(1)
In a system configured with an operator terminal, these same characteristics apply within any task group that has used the $CMSUP (console message suppression) macro call ti temporarily suppress messages to the operator terminal.

- A GET command or $GTFIL (get file) macro call to reserve a volume that is not mounted results in a 020C error return (volume not found) to the application.

- A read or write hardware error results in a 0107 error return (hardware error) to the application.

- No system-supplied message can be issued to signal the termination of system startup.

File System Pathname in CLM Directives

Whenever the format of a Configuration Load Manager (CLM) directive indicates that a file system pathname is to be supplied as an argument, the pathname must be expressed in one of three forms:

1. A full absolute pathname.

   This form of pathname is required to identify a file on a disk volume other than the system (bootstrap) volume.

   A full abosulte pathname begins with a circumflex (^) and a disk volume root directory name (which is the same as the volume_id). Each successive element in the pathname is preceded by a greater-than sign (>) and is hierarchically subordinate to the preceding element. Every element in the pathname except the last is a directory name. The last (rightmost) element is a file name.(1) The last element is not followed by any symbol or punctuation.

   Example:

   ```
   CLMIN  ^USRVOL>DIR_1>FILE_A
   ```

   In this example, a full absolute pathname is used in a CLMIN directive. The pathname indicates a target file that exists on a disk volume other than the system (bootstrap) volume.

2. An absolute pathname.

   This form of pathname is used to identify a file on the system (bootstrap) volume or a nondisk peripheral device.

   An absolute pathname begins with a greater-than sign (>) and the name of a directory or file that is immediately subordinate to the root directory of the system volume. The rest (if any) of the pathname is similar to the full absolute pathname described above.

(1)
In non-CLM environments, the last element in a pathname may be either a directory name or a file name, as appropriate to the situation.

For a magnetic tape file, the absolute pathname must be !device_unit>volume_id>file_name. The value for device_unit must be specified in the related DEVICE directive.

For other nondisk peripheral devices, the absolute pathname must be !device_name. The value for device_name is as specified in the related DEVICE directive. (If device_name is not specified in the DEVICE directive, the value of the device_unit argument is used for device_name.

Example:

```
DRIVER >DIR_Y>USRDRV
```

In this example, an absolute pathname is used in a DRIVER directive. This pathname indicates a target file that exists on the system volume. DIR_Y is a directory immediately subordinate to the root directory of the system volume.

3. A <u>Relative</u> pathname.

   During execution of the Configuration Load Manager, this form of pathname is used to identify a file in this system task group's working directory, in this case directory SID, which is immediately subordinate to the root directory of the system volume. ("Relative" means relative to the working directory of the task group).

   In this situation, the relative pathname is expressed as a simple file name with no preceding or following symbols or punctuation.(1) The named file must exist in directory SID on the system volume.

   Example:

   ```
   LDBU RBRCIP
   ```

   In this example, a relative pathname is used in an LDBU directive. This pathname indicates a target file that exists immediately subordinate to directory SID on the system volume.

In all types of pathnames, a directory name cannot exceed 12 characters, a file name cannot exceed 12 characters, and the total pathname (including ^ and > characters) cannot exceed 57 characters. In a full absolute pathname, the volume root directory cannot exceed six characters. The first character of each directory name and each file name must be alphabetic or a dollar sign ($).

(1)
Other forms of the relative pathname exist, but their use with the Configuration Load Manager is discouraged.

CLMIN DIRECTIVE

Directive Name: CLMIN

A CLMIN (CLM input) directive causes the source of CLM directives to be changed from the current device/file to the one indicated by the pathname argument. All subsequent CLM directives are read from the indicated device/file (unless it too contains a CLMIN directive).

Format:

```
CLMIN pathname
```

Argument Description:

pathname

Consists of an ASCII string that identifies the file system pathname from which subsequent CLM directives will be read.

Functional Description:

In a file of CLM directives, a CLMIN directive cannot appear ahead of the DEVICE directive that identifies the device implied by the pathname argument.

When a CLMIN directive is read by the Configuration Load Manager, the input file is immediately transferred to the beginning of the device/file indicated by the pathname. Subsequent CLM directives are read from the indicated device/file.

If a disk or magnetic tape volume is indicated by pathname, that volume must be mounted before system startup begins.

Example 1:

```
CLMIN ^ABCDVOL>BETA1
```

In this example, subsequent CLM directives will be read from a file named BETA1 on disk volume ABCVOL. An absolute disk pathname is used. Volume ABCVOL must be mounted on a disk device whose DEVICE directive has already been read by the Configuration Load Manager.

Example 2:

```
CLMIN  !CDR00
```

In this example, subsequent CLM directives will be read from card reader whose device_unit name is CDR00. (Assume that the DEVICE directive for CDR00 did not include a device_name argument; in this case, the device_unit argument value is used for file system references to the device). The DEVICE directive for CDR00 must already have been read by the Configuration Load Manager.

COMMENT DIRECTIVE

Directive Name: * (asterisk)

The COMMENT directive is used to include a comment in the file of CLM directives.

Format:

```
* Comment text
```

Functional Description:

When an * (asterisk) is the first character of an entry in the file of CLM directives, the remainder of the entry is treated as a comment and is not processed by the Configuration Load Manager.

Comments can also be included on other CLM directives if they follow a blank after the last argument of the directive.

DEVICE DIRECTIVE

Directive Name: DEVICE

A DEVICE directive is required for each MDC-connected peripheral device in the system. You may allow the system to dynamically assign channel numbers to appropriate devices on the bus. You must include a special format of the DEVICE directive in your CLM file if your installation will use a dual-purpose operator terminal. Refer to "Configuring a Dual-Purpose Operator Terminal" earlier in this section.

Format:

```
DEVICE   device_unit,lrn,level,X'channel',[device_name],
                         {B}
         [record_size],[ {N} ],['terminal-type'][,C/T]
```

Argument Description:

device_unit

A string of five ASCII characters. This first three characters identify the type of device: the last two characters (alphanumeric) must be unique within the device type so as to identify one specific device of that type. Table 5-2 indicates the permissible values of device_unit, the device type indicated by each value, and the default physical record size (in bytes) for each device type.

Except where indicated, the default record size can be overridden by use of an explicit record_size argument in the DEVICE directive.

In the case of KSRnn, the device_unit argument also establishes the following characteristics for the terminal:

- Trailing carriage return option specified.
- Trailing line feed option specified.
- Leading control byte option specified.
- Echo mode option specified.

If you wish to change any of these characteristics of the terminal, you can do so by means of an STTY (set terminal characteristics) command, a $STTY macro call, or a STTY directive.

Table 5-2. Unit Values and Default Record for Various Device (d)

Device_Unit Value	Device Type Indicated	Default Physical Record Size in Bytes (Decimal)
KSRnn/CONnn	Terminal (MDC-connected)[a]	KSR-73 CON-80
LPTnn	Line Printer	137
SPTnn	Serial Printer	133
CDRnn	Card Reader	80
CRPnn	Card Reader/Punch	80
DSKnn	Diskette	128(b)
RCDnn	Removable Cartridge Disk	256(b)
FCDnn	Fixed Cartridge Disk	256(b)
MSMnn	Mass Storage Unit	256(b)
RCMnn	Removable Cartridge Module Disk	256(b)
FCMnn	Fixed Cartridge Module Disk	256(b)
MT7nn	Magnetic Tape (7-track)	0(c)
MT9nn	Magnetic Tape (9-track)	0(c)

a
The KSR device_unit value is included for compatibility with previous releases. The CON device_unit value should be used when creating a new CLM directive file. The CON device_unit value must be used if the user intends to perform forms processing from an MDC-connected VIP 7200 or VIP 7205 terminal. Using the KSR device_unit value and the CON device_unit value will cause two different device handlers to be loaded.

b
The default record size for this device cannot be altered by means of the record_size argument of the DEVICE directive.

c
It is not necessary to issue a GET command or a $GTFIL (get file) macro call to establish the logical record size (and block size). The record size cannot be altered by the record_size argument of the DEVICE directive.

d
Communications devices are described in Section VI.

lrn

Specifies the logical resource number by which the device is requested. The value for lrn is a decimal integer from 3 through 252. (lrn 0 is used for the operator terminal. lrn 1 is used for the bootstrap device. lrn 2 is reserved for system use).

DEVICE

level

Specifies the priority level used by the device driver for this device. The value for level is a decimal integer from 7 through 59.

X'channel'

Specifies the 4-digit hexadecimal channel number of the device. (The leftmost 10 bits specify the channel number; the righmost six bits must be zero).

If you specify the channel number by a single zero (0), the system will automatically assign this device to an appropriate channel.

If a DEVICE directive for a disk device contains an lrn of 1 and a channel number of zero, the bootstrap device channel number replaces the channel number of zero.

[device_name]

For disk and magnetic tape devices, the device_name argument must be omitted or specified as * (asterisk) because the device identity is established by the value of the device_unit argument. (The significance of using the asterisk is described below).

For devices other than disk or magnetic tape, device_name can be one of the following:

1. A string of up to 12 ASCII characters that establish a unique name by which the device is referred to within the file system. The first character of the string must be alphabetic.

2. Omitted, in which case the value of the device_unit argument is used for file system references to the device.

3. An * (asterisk), which indicates a "private" device to be accessible only through physical input/output.

[record_size]

The length, in bytes, of one physical record. If record_size is not specified, the default record size is as established by the device_unit argument. (The default record size for disk devices and magnetic tapes cannot be modified by the record_size argument).

$\left[\begin{Bmatrix} B \\ N \end{Bmatrix}\right]$

This argument is meaningful only for devices other than disk and magnetic tape.

For normally unbuffered devices (viz., CDR, CRP), B indicates that input/output to the device is to be buffered.

For normally buffered devices (viz., CON, KSR, LPT, SPT), N indicates that input/output to the device is to be unbuffered.

Input/output to a buffered device may be either asynchronous or synchronous: tabulation characters are expanded into space characters. Input/output to an unbuffered device is always synchronous: tabulation characters are not expanded.

NOTE: All arguments of a DEVICE directive must be unique (i.e., not duplicated on another DEVICE directive) with these exceptions: (1) the B/N argument need not be unique, (2) for pairs of removable/fixed cartridge disks, the same level number and channel number must be specified for both platters (3) multiple communications devices may be configured on the same level and (4) multiple (polled) VIPs may be configured on the same channel.

['terminal_type']

Applies only if the device_unit value is CON. Specifies the physical terminal type. Possible values are described below:

Value	Physical Device Type
'7200'	VIP 7205 (DKU9103) VIP 7205 (DKU9102) The value 7200 must be used if MDC-connected VIP 7200 terminals are to be used for forms processing using the Display Formatting and Control Facility.
'PRU'	PRU1001/PRU1003/PRU1005/TWU1001/TWU1003/TWU1005
'D7200'	VIP 7207
'KSR'	The value KSR is used if the physical terminal type is to be supported in teleprinter mode.

*

[C/T]

Applies only if the device_unit value is CON. Specifies the visual display mode for the device. The value C indicates CRT mode. The value T indicates teleprinter mode. If C is specified, typed input will appear without edit control characters (as it appears in the memory buffer). If T is specified, typed input will appear as it was keyed-in. The default value is T.

DEVICE

Functional Description:

Each peripheral device in the system must be identified in a separate DEVICE directive. The device_unit argument (e.g., DSK01) identifies both the type of device (DSK) and one specific device (01) of the indicated type. The lrn, level, and channel number of each device must be unique (except in the case of a pair of removable/fixed cartridge disks, in which case the level and channel for both platters must be the same).

The first DEVICE directive for each device type causes the appropriate device driver to be loaded as part of the system. Each device driver is re-entrant, so that only one copy of each required driver is loaded to service all devices of the same type.

A DEVICE directive for the bootstrap device is optional. If none is specified, the bootstrap device is assigned priority level 6 (the default). If you wish to override the default priority level, you should specify a DEVICE directive that indicates the appropriate device_unit, logical resource number 1, desired priority level, and a channel number of zero (e.g., DEVICE DSK00,1,level,X'0'). If the channel number is not zero, the following message is issued: 1345 CMD (DEVICE) SPECIFIES DUPLICATE DEVICE TYPE/UNIT.

Example 1:

```
DEVICE  LPT01,12,20,X'1380'
```

In this example, the line printer whose unit number is 01 is assigned logical resource number 12 and priority level 20. This device is connected to channel 1380. The following characteristics are established by default:

- The device_name used within the file system will be the same as the device_unit (i.e., LPT01).
- The record size will be 137 bytes.

Example 2:

```
DEVICE  RCD01,9,10,X'1400'
DEVICE  FCD01,6,10,X'1400'
```

This example illustrates DEVICE directives for a pair of removable/fixed cartridge disks. The logical resource number for each platter must be unique. The level and channel number for both platters must be the same.

DRIVER

DRIVER DIRECTIVE

Directive Name: DRIVER

The DRIVER directive is used to identify a user-written device driver and one device driven by the driver.

Format:

```
DRIVER   pathname,lrn,level,X'channel'[,X'stack_size:
         RCT_size'][,X'hhhh']
```

Argument Description:

pathname

An ASCII string that identifies the file system pathname of the device driver bound unit.

lrn

Specifies the logical resource number used to identify a device driven by this driver. The value for lrn is a decimal integer from 3 through 252.

level

Specifies the priority level used by the device driver. The value for level is a decimal integer from 7 through 59.

X'channel'

Specifies the 4-digit hexadecimal channel number of a device driven by this driver. (The leftmost 10 bits specify the channel number: the rightmost six bits must be zero). You have the option to allow the system to assign a channel number for this device by specifying a single zero for this argument. If you choose this option, you must also specify the X'hhhh' argument described below.

[X'stack_size:RCT_size]

Specifies the sizes in words of (1) the stack required by this driver and (2) the resource control table (RCT) used for a device driven by this driver. The sizes are specified by a four digit hexadecimal number: the leftmost two digits specify the stack size;

the righmost two digits specify the RCT size. If the stack size and/or the RCT size exceed 255 words (decimal) you may use the double word hexadecimal format, D'hhhhhhhh', where the first four hexadecimal digits represent the stack size and the second four hexadecimal digits represent the RCT size. If the driver is to use the system routine ZIOSUB, the stack size must be at least 22 words for a SAF configuration and 40 words for a LAF configuration. The minimum size for the RCT is 3 words. If a smaller value is entered, the system replaces it with 3. The default values are 8 words for stack size and 16 words for RCT size. You must take the default values for both sizes or specify both. You cannot specify one and take the default for the other.

[X'hhhh']

Specifies a four-digit hexadecimal number corresponding to the internal hardware id of the device being driven by this driver.

Functional Description:

The DRIVER directive causes the user-written device driver identified by the pathname argument to be loaded into the system area of memory. (Because this driver is loaded into the system area of memory, the SYS Linker directive must have been specified when this bound unit was linked). The user-written device driver becomes a permanently resident part of the system software. (If the driver includes initialization routines to be executed when the driver is loaded, these routines also remain permanently resident).

The DRIVER directive causes creation of a resource control table (RCT) of the indicated size. It also causes creation of a task control block (TCB) with a dedicated priority level. Additional data structure initialization is achieved as described below.

The data structures allocated for use by this device driver are sufficient only for physical input/output operations. The RCT will be used by the driver to interface with the indicated device. Input/output from a program can be initiated by a $RQIO (request input/output) macro call that identifies an appropriate input/output request block (IORB). (The format of the IORB is described in the System Service Macro Calls manual).

Each user-written device driver runs at a dedicated (unique) priority level. If a given device driver (single task) drives multiple devices on different channels, one DRIVER directive is required for each device. Each DRIVER directive must specify a unique lrn and channel number, and a common priority level and pathname (see the example). In this case a separate RCT and stack are created for each device but a common TCB is used. Multiple devices of the same type, each with its own work area, are thus driven at a common priority level by a common task. (If a given device driver drives multiple taks, the pathname arguments in each DEVICE directive must be identical, but each task must have a unique priority level).

The initialization of data structures caused by the DRIVER directive is summarized below.

- An RCT of the indicated size is created. Immediately below the RCT, space for a pointer to the TCB is provided: one word for a SAF system, two words for a LAF system.

- The RCT is immediately preceded by a TCB pointer which is in turn immediately preceded by the stack.

- The pointer at the appropriate lrn offset in the logical resource table (LRT) is set to point to the channel/level word (the first word) of the RCT.

- The TCB pointer immediately below the RCT is set to point to the TCB.

- The channel and level values of the DRIVER directive are entered into the channel/level word of the RCT.

- The RCT is marked as having no volume descriptor block (private volume). Thus the device is accessible only through physical input/output (normally, the $RQIO macro call) and not through the file system.

- The TCB is marked as having a dedicated priority level. (The TCB is also set to point to the entry point of the device driver).

- The remaining area of the RCT is set to zero. The driver must initialize the RCT as required.

Example:

```
DRIVER DIRECTORY_1>OWN_DRIVER,12,9,X'0300'
DRIVER DIRECTORY_1>OWN_DRIVER,13,9,X'0380'
```

In this example, a user-written device driver whose file name in OWN_DRIVER is identified. It will be used to drive two devices under a single task. The level and pathname arguments of the two DRIVER directives are the same: the logical resource numbers and channel numbers for the devices are unique. The RCT for each device will be 16 words. The size of the stack required by this driver is 8 words.

LDBU DIRECTIVE

Directive Name: LDBU

The LDBU directive causes the indicated bound unit to be added to the end of the bound unit load list. The bound unit's root is loaded into the system area of memory after all CLM directives have been read. (Since the root of the bound unit will be loaded into the system area of memory, the SYS Linker directive must have been specified when this bound unit was linked). Once loaded, the root of the indicated bound unit is permanentely resident in the system.

If you wish to incorporate an optional system capability into your configuration, such as error logging or defective memory trap handling, you might be required to load one or more bound units using LDBU directives. Refer to Sections VII and VIII for descriptions of the bound units associated with optional system capabilities.

Format:

```
LDBU pathname
```

Argument Description:

pathname

An ASCII string that identifies the file system pathname of the bound unit.

Functional Description:

Application-specific code (usually in the form of subroutines shared among multiple task groups) that is referred to symbolically during application execution can be permanentely brought into memory at system startup by LDBU directives. The pathname in each LDBU directive is added to a bound unit load list.

The order of LDBU directives governs the order in which pathnames are added to the list. If two LDBU directives specify the same pathname, the second one is ignored.

After the QUIT directive (the last CLM directive) is read by the Configuration Load Manager, the roots of the bound units identified by LDBU directives are permanently loaded into the system area of memory. The symbol table for each such bound unit is added to the system's resident symbol table list. Once a bound unit has been loaded, any symbol defined therein by an EDEF Linker directive serves to resolve an unresolved reference to that same symbol in a subsequently loaded bound unit. Otherwise, if a bound unit contains an unresolved reference to a symbol, loading stops with an error halt.

Example:

```
     LDBU  ^ABCVOL>ALPHA1
```

In this example, a bound unit named ALPHA1, which is a file immediately subordinate to the root directory of a volume named ABCVOL, will be added to the bound unit load list.

A bound unit loaded by means of the LDBU directive may contain an initialization subroutine table (IST). The IST defines one or more subroutines of the bound unit that are to be executed once only, immmediately after the root is loaded. If a bound unit contains initialization subroutines, the Linker directive IST must be specified when the bound unit is linked. The label in the IST section is the argument to be entered in the IST directive.

The format of an IST section is given below:

```
label     DC    0[1]
          RESV  $AF,0       RFU

          DC    <sub1       parameters to subroutine sub1 (may
          DC    0           be any sixteen bit value)
          DC    0

          DC    <sub2       parameters to subroutine sub2 (may
          DC    0           be any sixteen bit value)
          DC    0
               .
               .
               .
```

(1)
If the entry is 0, all memory used by the initialization subroutines is returned to the system after the subroutines have been executed (provided the subroutines have not caused the system to create data structures on their behalf). To retain a subroutine in memory, specify n, where n is the number of words to be retained. In this case, take care not to extend the memory requirement beyond that allocated to the bound unit's root when it was loaded.

```
          DC    <subn  )   parameters to subroutine subn (may
          DC    0      }   be any sixteen bit value)
          DC    0      )

          RESV $AF,0       End of table sentinel

sul1      ........
          ........
          ........

sul2      ........
          ........
          ........

subn      ........
          ........
          ........
```

Upon entrance to a subroutine, register B5 contains the return address. Before exit from the subroutine, register R1 must contain the status. A value of zero indicates correct execution. A nonzero value is the error code for this subroutine. If the status is non-zero, the following message is written to the error-out file, and a halt occurs. Execution of CLM cannot continue.

    1348 INITIALIZATION SUBROUTINE ERROR

    (error code returned to register R1)

MAP DIRECTIVE

Directive Name: MAP

The MAP directive provides configuration information that enables you to verify the success of certain aspects of the system building process. The information presented by this directive includes the system physical attributes, system structures, device status, user-defined variables, and memory pool specifications. This information can be used to ease the debugging of system extensions and user-written device handlers. In addition, information (such as system symbols) is presented that would otherwise be unavailable.

Format:

    MAP [path] [,map_form,] [,map_form2]...

Argument Description:

path

Pathname of the file to which the map lising is to be written. The pathname cannot be that of a communications device. The pathname must be able to be verified when the CLM reads the MAP directive.

If the pathname refers to a device file (e.g., !LPT00), that device must have been configured when the MAP directive is read.

If the pathname refers to a disk file, the indicated volume must be mounted on the drive of a device that is already configured and recognized by the file system when the directive is read. If the pathname refers to a disk file that does not exist, the file will be created (as long as the directory structure identified in the pathname already exists). If the file already exists, it must be a variable sequential file. In this case, it will be opened in "renew" mode.

The default map path value is >SID>CLM_MAP.

[map_form,]...[map_formn]

Any combination of the map format arguments described below can be entered in any order. If none of these arguments is specified, the entire system map is written to the map file. The complete map presents the following information (in the order shown):

- System physical attributes
- System structures

- User-written device driver attributes
- Device status for devices accessed by the file system
- User-defined variables
- Memory pool attributes
- System extensions
- User-selected permanently resident system overlays and system overlays made resident to serve as device drivers
- System symbol table

{ -BRIEF | -BF }

Requests display of only the physical attributes, system structures, user-written driver attributes, and user-selected permanently resident overlays.

{ -NDEV | -ND }

Omits from the map the data pertaining to device status (devices accessed through the file system).

{ -NVAR | -NV }

Omits from the map the data pertaining to user-defined variables.

{ -NPOOL | -NP }

Omits from the map the data pertaining to memory pools.

{ -NRES | -NR }

Omits from the map the data pertaining to user-selected permanently resident system overlays and system overlays made resident to serve as device drivers.

{ -NSYM | -NS }

Omits from the map the data pertaining to the system symbol table.

Example 1:

```
MAP -NS
```

The system symbol table data is omitted from the map. The map is written to the map file whose pathname is >SID>CLM_MAP.

Example 2:

```
MAP !CONSOLE
```

The entire map is produced and written to the console. The console output in this example is as follows:

```
CLM MAP

PHYSICAL ATTRIBUTES:

     GCOS6 MOD400-L2.1-03/28/1450
     SIP PRESENT
     NO  CIP PRESENT
     MMU PRESENT
     HIGH PHYSICAL MEMORY ADDRESS: 3FFFF
     HIGH VIRTUAL MEMORY ADDRESS:  6EDFF
     TIME BETWEEN REALTIME CLOCK INTERRUPTS:    32
     OPERATOR CONSOLE CONFIGURED
     ERROR MESSAGE LIBRARY IN USE

SYSTEM STRUCTURES: (DECIMAL NUMERIC VALUES)

     45    IRBS,  36    TSAS,  10    SYSTEM OVERLAY AREAS,
     0     RECORD LOCKS CONFIGURED

USER-WRITTEN DRIVERS:

     ZIO488       RELOCATION= 5501   END= 5B80   ENTRY ADDRESS= 590A
         LRN=20  LEVEL=21  CHANNEL=FE00    TCB AT  5886  RCT AT  5DED
         STACK ALLOCATION  5BEA    TO 5DEA

     ZIO488       RELOCATION= 5FA0   END= 661F   ENTRY ADDRESS= 6479
         LRN=21  LEVEL=22  CHANNEL=C000    TCB AT  6655  RCT AT  670C
         STACK ALLOCATION  6689    TO 6709

     ZIO488       RELOCATION= 5FA0   END= 661F   ENTRY ADDRESS= 6479
         LRN=22  LEVEL=22  CHANNEL=A000    TCB AT  6655  RCT AT  691C
         STACK ALLOCATION  690B    TO 6919


DEVICE STATUS: (DEVICES VISIBLE TO FILE SYSTEM)

  SYMBD  CHANNEL DEVICE   VOLUME USAGE AVAILABLE_SECTORS  VOLUME  MEMBER
   NAME           TYPE      ID         PHYSICAL  LOGICAL SET NAME NUMBER
B RCD00    1400    2332   ^CHECKL  34       0        0
B MSM01    1800    2361   ^         0
B RCD01    1480    2332   ^         0
B DSK00    0400    2010   ^VL6226   1      502      502
B DSK01    0480    2010   ^CLM      0      154      154
B DSK02    1200    2010   ^         0
B DSK03    1280    2010   ^RBT      0     1952     1952
B CONSOLE   500    201A   ^         2
B CDR00    1300    2008   ^         0
B LPT00    1380    2002   ^         1

USER-DEFINED CLM VARIABLES:

     A = 5          B = 50        C = 500       D = 5000        E = 50000
     F = 5555       J = FFFFFFFF  K = FFFFFFCE  O = 0           P = 12C
     Q = 0          W = FFFFFCCB  Z = FFFFED40

MEMORY POOLS:

     $$  START=9980     END=10EFF    SIZE= 7580    VIRTUAL BIAS= 0
```

```
     ATTRIBUTES= P        USER'S RING NUMBER= 0
AL   START=20000   END=200FF   SIZE= 100     VIRTUAL BIAS= F100
     ATTRIBUTES= EP       USER'S RING NUMBER= 1
AK   START=30000   END=301FF   SIZE= 200     VIRTUAL BIAS= 1F000
     ATTRIBUTES= EPCSU    USER'S RING NUMBER= 2
AB   START=40000   END=6EDDF   SIZE= 2EDE0   VIRTUAL BIAS= 2EE00
     ATTRIBUTES= PC       USER'S RING NUMBER= 1
AM   START=40000   END=4003F   SIZE= 40      VIRTUAL BIAS= 2EE00
     ATTRIBUTES= CS       USER'S RING NUMBER= 1
AN   START=40000   END=4005F   SIZE= 60      VIRTUAL BIAS= 2EE00
     ATTRIBUTES= PSU      USER'S RING NUMBER= 2

LDBU'S: (SYSTEM EXTENSIONS)

     ZIO488       START= 5501
     ZIO488       START= 5FA0
     ZXDEFM       START= 7DDB
     OVLYCT       START= 7FFE
     VLDCD        START= 843E

RESOLA'S: (PERMANENTLY RESIDENT EXECUTIVE OVERLAYS)

     OIX     START=4B9D  , OIODK    START=31C2  , OIOLP    START=53F3  ,
     OIOCR   START=520B  , OXPCL1   START=8F66  , OIOIM5   START=8AEB  ,
     OXPCL2  START=9170  , OXGRT2   START=8777  , OXPCL3   START=9001  ,
     OIOIM0  START=896A  , OIOIM3   START=88A9  , OORIND   START=8533  ,
     OOROO2  START=840E  , OOERS2   START=933A  , OYASF    START=9777  ,
     OYCRR   START=8D40  , OYTT     START=8617  , OZAVR2   START=94A1  ,
     OZACCB  START=8360  , OZXEP    START=96A0  , OYCKPT   START=8BD2  ,

SYSTEM SYMBOL TABLE:

     ZSEXEC
          LOCATIONS
               ZIOSUB =  E5D      ZXSTMX = 116E      ZXVLD  = 11E2
               ZIORFQ = 1097      ZIATON = 111D      ZMBSVU =  B0D
               ZMBSPO =  AFE      ZMRSVO =  B10      ZMRSPO =  B01
               ZMSWAP =  AF2      ZMRLOU =  AF5      ZMRLIN =  AF8
               ZMRASD =  AFB      ZIOADP = 1137      ZIOERP = 113B
               ZIOEND =  F05      ZIOTRW =  F0A      ZMROWN =  B1F
               ZXMCLX = 1DAF      ZMSSG  = 1E34      ZXSPRQ = 1BBE
               ZXSPI  = 1CCB      ZXCRT  = 1CCB      ZMACG  = 1E3F
               ZXSFIP = 174B      ZXC_SI = 20F5      ZXIPSI = 143B
               ZXPISA = 1DCC      ZXTWI  = 124A      ZXTSTK =  D12
               ZYOPF3 = 2215      ZYCLF3 = 2220      ZYFM   = 215A
               ZYRDR  = 21D6      ZYSM2  = 213B      ZYSM   = 216A
               ZXTRML = 1F4A      XXDATA =    0      XYCS   =  9AB
               XM_MGR =  ABB      XM_GET =  B2B      XM_RIN =  BC8
               XXCKIT =  C2C      XXLRCT =  C72      XXMVCP =  C90
               XXDTAB =  CCF      XXTRB  =  CF2      XXTSTK =  D12
               XIO    =  DAC      XIOTMR =  F1A      XIOMGR = 108E
               XISPNS = 11BD      XXWAIT = 11F3      XXDQ   = 1267
               XXPOST = 13F3      XXTERM = 14EA      XXSEM  = 1579
               XXTRAP = 168A      XOAMGR = 17A6      XOLOAD = 184F
               XORQUT = 1807      XOVMGR = 19BC      XOERS  = 1B62
               XXREQ  = 1B7F      XXSIO  = 1C49      XXTSKR = 1CA2
               XXSTSK = 1D6B      XXMCL  = 1DBB      XXGRPR = 1E13
               XXAVR  = 1EC3      XXQDQ  = 1ED8      XXTRML = 1F4A
               XXRCS  = 1F5C      XXMAX  = 1F61      XXC_AI = 1F76
```

```
        XXC_W   = 1FA0        XXC_CI  = 1F95        XXC_IN  = 1FB1
        XXC_RW  = 2034        XXCMGH  = 20B8        XXFSW   = 2100
        XXMISC  = 2112        XYEXT   = 2117        XYINT   = 21C4
        XZLLFN  = 2237        XZM400  = 2F6E        XZOVM4  = 3072
        XZSLCD  = 2324        XZVL    = 247C        XZVLM4  = 24AC
        XZXDEF  = 3118        XYDMD   = 250E        XYDMR1  = 252C
        XYDMS2  = 257D        XZDSUB  = 2650        XZBF    = 2AA3
        XZRSUB  = 2763        XZSM    = 2820        XYSM    = 28FD
        XZBPAS  = 2CF7        XZSMD   = 29AE        XZSUB   = 2D5B
        XZCONC  = 2DAC        XZGFDB  = 2E7A        XXGSPN  = 3118
        XXUPFD  = 31BC        XXPTCH  = 329A        XIODK   = 31C2
        XXINIT  = 347A        ZYXPA2  = 2200        ZICDR   =  DB1
        ZIDK1   =  DAC        ZIERLG  = 1108        ZIKSR   =  DB1
        ZILP1   =  DB1        ZIDAVR  = 1141        ZIOCR   =    0
        ZIODK   =    0        ZIDERL  = 1139        ZIOINT  =  DB1
        ZIOLD   = 1115        ZIOTMO  = 111F        ZIDORG  = 112A
        ZX_PVC  = 1130        ZIOSCH  = 1132        ZIOASD  = 1133
        ZIOLP   =    0        ZIOTRB  =   49        ZIOTVC  =  E28
        ZIOXRW  =   65        ZIO_AD  = 1135        ZIO_ST  =  E08
        ZIO_XA  =   1B        ZIOTER  =  DE9        ZITAPF  =    0
        ZITYP   = 1043        ZITYPR  = 1053        ZMGETH  =  830
        ZMGFIL  =  832        ZMRTRN  =  BCC        ZOALDC  = 1824
        ZOARWC  = 1839        ZOLDRT  = 1909        ZORELS  = 1AC5
        ZORLDS  = 1A7C        ZTVIOD  = 113E        ZOERS   = 1B6A
        ZXCNSR  = 1637        ZXCNRW  = 1326        ZXC_CR  = 20C3
        ZXC_GI  = 2099        ZXC_RW  = 2034        ZXDW    = 1270
        ZXD_DF  = 12AF        ZXD_DH  = 12C4        ZXD_PP  = 12CE
        ZXD_PR  = 12EA        ZXD_RC  = 12F4        ZXD_IR  = 1529
        ZXD_IR  = 1529        ZXECA   =  C2C        ZXECA   =  C2C
        ZXGCH   =  389        ZXISTR  = 1558        ZXLMVE  =  C90
        ZXLRCI  =  C75        ZXMCLT  = 1007        ZXPOST  = 142F
        ZXREW   = 187F        ZXSCH   =  1BE        ZXSRCT  =  C72
        ZXSWT   = 15C0        ZXUOUT  = 1C55        ZXUPFD  = 31BC
        ZXVPST  = 1598        ZXWT    = 1213        ZXXCA   =  C4E
        ZXXCA   =  C4E        ZYDM2   = 2136        ZYFM2   = 2155
        ZYUPF2  = 2212        ZMDM01  =  C18        ZMDM02  =  C18
        ZXDFIP  = 168F        ZXTHEX  = 1776        ZUDM01  =  1A2
        ZFRMIN  =   1A        ZFRMRW  =   7D        ZFRMAT  =  952
EDEFS
        $ITCH0  =  231        $XGCH   =  1CB        $XSFIP  = 158A
        $MDM01  =  A5A        $MDM02  =  A5D
OVERLAYS
        OIX     =  0          OIODK   =  1          OFMCR1  =  2
        OIOLP   =  3          OIOCR   =  4          OIOTPE  =  5
        OXPCL1  =  6          OIOIM5  =  7          OXPCL2  =  8
        OIOIM6  =  9          OXCTSK  =  A          OXRXFR  =  B
        OXGRT2  =  C          OXCDVT  =  D          OIOIM8  =  E
        OXPCL3  =  F          OXPCBU  =  10         OXDTSK  =  11
        OXCGRP  =  12         OXSMCL  =  13         OXDFRT  =  14
        OXDFR2  =  15         OXDFSP  =  16         OXCGR1  =  17
        OXRWGP  =  18         OXGRWS  =  19         OXRWG2  =  1A
        OXRWG3  =  1B         OXGRS2  =  1C         OXGRT3  =  1D
        OXDTK1  =  1E         OXGRQT  =  1F         OXDLGP  =  20
        OXCDRV  =  21         OXACSP  =  22         OIOIM0  =  23
        OOERS1  =  24         OIOIM1  =  25         OXC_XD  =  26
        OIOIM2  =  27         OXC_IT  =  28         OIOIM3  =  29
        OXSIO1  =  2A         OXSIO2  =  2B         OXSIO3  =  2C
        OXSIO4  =  2D         OXTRR   =  2E         OXWCS1  =  2F
        OXAVR1  =  30         OIOIM4  =  31         OXTRAP  =  32
        OXGACT  =  33         OXTR17  =  34         OXWAIT  =  35
```

```
UXGSSP =  36     UXESWU =  37     UXDGRP =  38
UXUPFU =  39     UXUPF1 =  3A     UMSTAT =  3B
UUBIND =  3C     UUMSCO =  3D     UUPACK =  3E
UUROU1 =  3F     UUROU2 =  40     UUCRU1 =  41
UUERS2 =  42     UUERS3 =  43     UUEML2 =  44
UYASF  =  45     UYCRF  =  46     UYCRR  =  47
UYDLF  =  48     UYLKF  =  49     UYMDF  =  4A
UYULF  =  4B     UZALX  =  4C     UZALX2 =  4D
UZCRF2 =  4E     UZCRF3 =  4F     UZCRF4 =  50
UZCRF5 =  51     UZCRR2 =  52     UZLKDR =  53
UZULDR =  54     UYGLF  =  55     UYGIF  =  56
UYPLFN =  57     UYRMF  =  58     UZGIF2 =  59
UZGIF3 =  5A     UZCODB =  5B     UZCFDB =  5C
UZGFCB =  5D     UZGIRF =  5E     UYOPN  =  5F
UZUPFD =  60     UZUPFF =  61     UZUPFU =  62
UYCLS  =  63     UZCLFD =  64     UYACL  =  65
UZACL2 =  66     UYAVR  =  67     UYCMR  =  68
UYI1   =  69     UYSWP  =  6A     UZAVR2 =  6B
UZAVR3 =  6C     UZVM1  =  6D     UZVM12 =  6E
UZVM13 =  6F     UYCWD  =  70     UYXPA  =  71
UYSTTY =  72     UYWTFL =  73     UZSME  =  74
UZSME1 =  75     UZSME2 =  76     UZSMT  =  77
UYBPCH =  78     UYGIBP =  79     UZBF2  =  7A
UYDMR2 =  7B     UYDMR3 =  7C     UZDMR  =  7D
UYDMS1 =  7E     UYUPS2 =  7F     UZDMS2 =  80
UZACCB =  81     UZUCCB =  82     UZUSUB =  83
UYDMI1 =  84     UYDMI2 =  85     UYUPI1 =  86
UZIUPF =  87     UZIPSN =  88     UYDMC  =  89
UZCER  =  8A     UZCINV =  8B     UZCKEY =  8C
UZCLNK =  8D     UZCUPF =  8E     UZCPSN =  8F
UZCSUB =  90     UZDMC  =  91     UYDMX1 =  92
UZDMX  =  93     UZXENT =  94     UZXER  =  95
UZXLVL =  96     UZXPSN =  97     UZXSPL =  98
UZXSUB =  99     UYDMX2 =  9A     UYDMT  =  9B
UZDMT  =  9C     UZTCLF =  9D     UZTEBC =  9E
UZTEXD =  9F     UZTUP1 =  A0     UZTUP2 =  A1
UZTUP3 =  A2     UZTUP4 =  A3     UZIPSN =  A4
UZISUB =  A5     UZIUTL =  A6     UYCKPT =  A7
UZCKPT =  A8     UYRSRT =  A9     UYUREC =  AA
UZBEF  =  AB     UZBIIR =  AC     UZBIOR =  AD
UZBIWR =  AE     UZCLPT =  AF     UZBOLB =  B0
UXMBC1 =  B1     UXMBC2 =  B2     UXMBC3 =  B3
UXMBC4 =  B4     UXMBC5 =  B5     UXMBA1 =  B6
UXMBA2 =  B7     UXMBA3 =  B8     UXMBI1 =  B9
UXMBI2 =  BA     UXMBS1 =  BB     UXMBS2 =  BC
UXMBS3 =  BD     UXMBR1 =  BE     UXMBR2 =  BF
UXMBR3 =  C0     UXMBR4 =  C1     UXMBD1 =  C2
UXMBI1 =  C3     UXMBI2 =  C4     UXMBI3 =  C5
UXMBI4 =  C6     UXELG1 =  C7     UXELG2 =  C8
UXCKP1 =  C9     UXCKP1 =  CA     UXCKP2 =  CB
UXCKPA =  CC     UXCPA2 =  CD     UXRST  =  CE
UXRST0 =  CF     UXRST1 =  D0     UXRST2 =  D1
UXRST3 =  D2     UXRSTF =  D3     UXRSER =  D4
UXRSTP    D5     UXCKPD =  D6

ZIO4BB
    LOCATIONS
        ZIO4B8 = 551B     ZIO4B9 = 551B     ZIO4BA = 5621
        ZIO4BB = 590A
```

```
ZIO488
    LOCATIONS
            ZIO488 = 5FBA      ZIO488 = 5FBA      ZIO4BA = 60C0
            ZIO481 = 6479

ZXDEFM
    LOCATIONS
            ZXINIT = 7ECB

OVLYCT
    LOCATIONS
            TAB1   = 7F1B      TAB2   = 801C      TAB3   = 811D
            TAB4   = 821E

VLDCD
```

MEMPOOL

MEMPOOL DIRECTIVE

Directive Name: MEMPOOL

A MEMPOOL directive defines a system memory pool, a batch memory pool, an exclusive online memory pool, or a nonexclusive online memory pool. Each MEMPOOL directive describes only one type of memory pool, but a single MEMPOOL directive may describe more than one exclusive online memory pool or more than one nonexclusive online memory pool. (You have the option of expressing the size of a memory pool symbolically by defining the size in a VARIABLE directive. See the description of the VARIABLE directive elsewhere in this section).

Format 1:

```
     MEMPOOL     S,,size [,P]
```

Format 2:

```
     MEMPOOL     B,,size   [ ,P { NC } ]
                           [ ,  { C  } ]
```

Format 3:

```
     MEMPOOL   { E, }  pool_name,  { size    ,[X][U][P][C][S] }  ...
               { ,  }              { poll_name]               }
```

Argument Description Format 1

S

The letter S designates a system memory pool. The system memory pool must be defined in a separate MEMPOOL directive. If multiple MEMPOOL directives have S as the first argument, only the last one is used.

size

The size argument is a positive integer that defines the number of words requested for the memory pool. To specify a pool size greater than 65535(10) words, use the double-word hexadecimal integer format (e.g., D'10A00').

The size of the system pool may also be specified by an asterisk (*), in which case CLM will allocate to the system pool all memory remaining after MEMPOOL directives specifying explicit sizes have been processed. Only one pool may be defined with "*" in its size field. Once any pool has been defined this way, no other pool may be so defined.

If a system memory pool is not defined in an explicit MEMPOOL directive, its size is set at 3K (3072) words.

The size of pools must be in multiples of a certain number of words and will be rounded to the next highest multiple by CLM. For pool that are neither protected nor contained, the size of the increment is 32 words. For pools that are protected or contained the size of the increment depends on the Level 6 system in which they are to be used as shown in Table 5-3. If a pool that is protected or contained has the potential or extending into the batch pool, the increment for the batch pool must comply with that of the extendable pool.

Table 5-3. Increments for Memory Pools

System[a]	Increment (Words)
LAF (with 1024K memory)	64K
LAF (with less than 1024K memory)	256
SAF (with 64K memory)	4K
SAF (with less than 64K memory)	256

[a] Memory pool protection/containment requires that the Level 6 central processor possesses a memory management unit.

Suggestions for calculating the sizes of memory pools are given after the functional description of the MEMPOOL directive.

P

The letter P specifies that the pool is protected; i.e., users of other non-overlapping memory pools are prevented from writing into it. If the argument is omitted the pool is not protected. If the pool being protected is the system pool, the executive code and the system data structures are protected along with the system pool. If the system being configured does not have the MMU, this option is not effective, and a warning diagnostic is issued.

Argument Description Format 2:

B

The letter B designates a batch memory pool. A batch memory pool can be specified only once, in a separate MEMPOOL directive. If two or more MEMPOOL directives have B as a first argument, an error occurs.

size

Same as for Format 1.

P

Same as for Format 2.

[NC]
[C]

The letter C specifies that the pool is contained; i.e., users of the pool are prevented from writing outside of it. The letters NC specify that the pool is not contained. The default for batch is contained. If the system being configured does not have an MMU, no pool may be contained and non-containment is the default.

Argument Description Format 3:

{E   }
{null}

E indicates an exclusive (non-overlapping) online memory pool. A MEMPOOL directive that has E as its first argument defines one or more exclusive online memory pools. Even if your MEMPOOL directive defines more than one exclusive online memory pool, you need to specify E only once (as the first argument). See Example 5. All exclusive online memory pools defined in <u>all</u> MEMPOOL directives with E as their first argument constitute a pool set.

A null (no entry) first argument indicates a nonexclusive (potentially overlapping) online memory pool. A MEMPOOL directive with a null first argument defines one or more nonexclusive online memory pools.

Each of these pool sets is an alternative definition of the same physical area of memory. All pools defined by a single MEMPOOL directive must be either exclusive or nonexclusive.

pool_name

The 2-character ASCII name that uniquely identifies each exclusive and nonexclusive online memory pool. The pool_name is used in CG (create group) and SG (spawn group) commands and in $CRGRP and $SPGRP macro calls.

If the pool_name begins with a decimal digit, the pool_name argument must be surrounded by apostrophes (e.g., '1A').

A pool_name argument should not be included in a MEMPOOL directive for a system memory pool or a batch memory pool. The omission of the pool_name argument in these cases must be signified by a comma.

NOTE: At login you can (1) specify the two character identification of the memory pool, or (2) specify a single character and accept the second character assigned by the facility, or (3) omit the pool identification and accept a two-character identification whose first character is L.

If you login by method (2), all online memory pools to be used by the login facility must have the same first character in their pool name arguments. This shared first character should not be assigned to any other memory pool. In addition, it is advisable to assign second characters starting with digits 0 through 9 (in order), then letters A through Z (in order); this practice will optimize the efficiency of the algorithm used by the system's logic facility when it searches for an available memory pool.

You can login by method (3) in the same way as for (2) except that the first character must be L.

If you login by methods (2) or (3), you should define at least as many memory pools as the number of users who may consurrently gain access to the system.

size

The description of the size argument for Format I also applies to Format 3. If an asterisk is not used to define the size of the system pool, it may be used to define the size of an exclusive or a nonexclusive online memory pool. In this case the pool size is calculated by CLM. CLM will allocate to this pool all memory remaining after MEMPOOL with explicit sizes have been processed. Once the size of an exclusive pool has been defined with an asterisk, no subsequent pool may have its size defined by an asterisk. Once the size of a nonexclusive pool has been defined with an asterisk, no more pools may be defined on that directive line. A subsequent pool directive may define the size of a nonexclusive pool with an asterisk if the pool is the last pool defined by that directive. The size of an exclusive pool may be defined with an asterisk if the size of a nonexclusive pool has been defined with an asterisk.

pool_name1

This option allows the user to refer to a given pool by two (or more) names. Pool_name1 is the name of a pool specified by a previous MEMPOOL directive; pool_name is the new name. The attributes of the pool are specified by the original directive and cannot be changed by subsequent directives that use this option.

[X]

The letter X indicates that the online pool can be extended into the batch pool area. If a memory request cannot be satisfied within the defined size of the online pool, the contents of the batch pool are rolled out of memory. If X is omitted, the online pool is not extendable. If the contents of the batch pool are to be rolled out of memory upon a SSPB (suspend batch) operator command, at least online must be defined with an X argument.

[U]

The letter U specifies that groups using this pool run in the unprivileged mode. Groups running in the unprivileged mode may not execute privileged instructions and will trap if a privileged instruction is attempted. The default is privileged.

[P]

Same as for Format 1.

[C]

Same as for Format 2, except that for pools defined by Format 3 the default is <u>not contained</u>.

[S]

The letter S specifies serial usage of the pool; i.e., only one group at a time can use it. An attempt to create a second group using a serial usage pool results in an error return and the second group is not created. If this argument is omitted, the pool can be used by more than one group at a time.

Functional Description:

Each MEMPOOL directive causes a pool descriptor list to be created. Later, after the memory requirements of the system and its extensions are known, each pool set is checked to ascertain whether it can fit into the remaining available space. If any element or pool set is too large, an error occurs.

The Configuration Load Manager creates a file named ROLLOUT under the SID directory of the bootstrap volume if a batch memory pool is defined <u>and</u> at least one online memory pool definition includes an X argument. If a ROLLOUT file is to be created on a bootstrap <u>diskette</u>, it is usually necessary to delete unneeded elements from the bootstrap diskette (^ZSYS00) and then copy the remaining contents of this diskette to a backup volume (where the remaining contents will be consolidated into a consecutive area). This technique will provide the largest possible consecutive area of unused space on the backup volume. Using this backup volume as the bootstrap volume, you can optimize the possibility of successful creation of a ROLLOUT file large enough to accommodate the batch memory pool.

The following discussion of the protect and contain attributes applies only to a system having an MMU. The system pool may be protected or not. It may not be contained. The batch pool may be contained or not. It may be protected or not; but if it is protected, it will be protected only from those pools that have no potential for extending into te batch pool.

All other exclusive nonextendable pools may be protected, or contained, or protected and contained, or neither protected nor contained. An exclusive pool that is extendable may also be contained. However, when such a pool is defined, the system does not prevent the user from accidentally writing into the batch pool area before the batch program is rolled out.

An exclusive pool that is extendable may also be protected. If such a pool extends into the batch pool area, the part in the batch area is protected only if there are no other extendable pools.

The protect and contain attributes apply to all nonexclusive pools collectively. If _any_ nonexclusive pool is protected or contained, then the set of nonexclusive pools have the same attribute(s). Nonexlusive pools may not be individually contained or protected from each other. A protected nonexclusive pool, therefore, is protected from exclusive pools or from the batch pool overwriting its space. It is not protected from other nonexclusive pools overwriting its space.

Guidelines for Calculating System Pool Memory Size:

The following tables provide guidance in calculating the size of the system memory pool.

Table 5-4 lists the number of words of resident code required for various system components: Table 5-4 also lists the number of sharable and/or exclusive files that are used.

Table 5-5 lists formulas that can be used to calculate file memory space.

Table 5-6 describes memory requirements for various system control structures.

Table 5-7 describes memory requirements for elements in online or batch memory pools.

Table 5-8 describes memory requirements for the Message Facility and Checkpoint/Restart (describes in the _System Concepts_ manual).

The user is advised that the system memory pool must be large enough to handle the "worst case" situation. To compensate for pool fragmentation, increase the total pool size by at least 10 percent. The pool size must be a multiple of 32 words.

Table 5-4. Resident Code Required for System Components

All system components not listed below (and not also the subject of a separate release bulletin) are less than 200 words.

Component	Resident Code	Sharable Files	Exclusive Files
Access Control Utilities			
Delete Access Control	1,080		
Delete Common Access Control	1,080		
List Access Control	1,545		
List Common Access Control	1,545		
Set Access Control	1,180		
Set Common Access Control	1,180		
BU_CONVERT	4,969	Input file	Output file
Check Mass Storage Volume	7,273		
Compare	4,986	Input file Ouptut file	
Compare ASCII	2,616	Input file Ouptut file	
Copy	6,856	Input file	Output file
Commercial Instruction Simulator	4,500		
Create Directory	380		
Create File	1,280		
Create Index	1,120		

Table 5-4 (Cont). Resident Code Required for System Components

Component	Resident Code	Sharable	Exclusive Files
Create Mailbox			
Create Volume	10,634	Output File (MDUMP etc.)	Output Volume
Data Exchange Utilities			
Compare Data Exchange	3,936		Input File Output File
Copy Data Exchange	3,945		Input file Output File
Create Volume Data Exchange	1,500		Outut file
File Change Data Exchange	1,286		Input file
List Data Exchange	3,284		Input file Output file
Debug (SD)	2,270		Work file
Debug (Multi-User)	1,400 (system pool) 1,500 (user pool)		Work File(s)
Quick Breakpoint Processing	3,750 (system pool + user-specified size of quick memory buffers		
Deferred Print/ Punch	1,055	Input file	
Delete Directory	790		
Delete File	820		

Table 5-4 (Cont). Resident Code Required for System Components

Component	Resident Code	Sharable	Exclusive Files
Delete Index	845		
Display Formatting			
Create Print VDAM VFORMS command	4,140 1,406 7,663 592	 FORMS>VDAM_FORM_O1	
DPEDIT	SAF 7,345 LAF 7,748	Input file Output file	
Editor	7,111 (+ 2KW per user)		Input file Output file Temporary work files (CI=128)
Export	2,300	Input files Output files	
File Change	1,354		Input file
File Dump	1,625 (+ 300W/ user)	Input file Output file	
Get file	735		
Get Quota	5,700		
Interactive System Definition (M4_SYSDEF	4,777 (+ 4KW Get Memory)		Output file
Linker	10,750 (+ default Get Memory of 2KW)	Input files	Output files
List Creation Date	2,990	Input file	
List Names	4,982	Input file Output file	

Table 5-4 (Cont). Resident Code Required for System Components

Component	Resident Code	Sharable	Exclusive Files
List Tape Contents	1,417	Output file	Input file
Listener	2,231	Input file	
Load Index	1,885	Output file	
Modify File	1,030		
Patch	10,299	Input file Output file	
Prime Index	5,790	Output file	
Print	1,507	Input file	
Print/Punch Daemon	6,800	Input file Output file	
Queue Report	1,772	Input file	Output file
Rename	1,025		
Report Queue Maintenance	3,000		Input file Output file
Restore	3,800 (+ 2.5K Get Memory)		Input file Output file + 1 temporary work file
Reorganize Indexed File	2,020	Input file Work file	
Save	2,525 (+ 2.3K Get Memory)	Input file	Output file + 1 temporary work file
Set Terminal Characteristics	2,135		
SIP Simulator (Double)	2,000		
SIP Simulator (Single)	600		

Table 5-4 (Cont). Resident Code Required for System Components

Component	Resident Code	Sharable	Exclusive Files
Tape Positioning	610		Input file
TCLC	2,900		
TCLP	See Note		
Unspool	5,700	Input files	
Walk Subtree	800		
ZXDEFM (Defective Memory Trap Handler)sl ZXPRF (Power Resumption Facility)	300 262		

NOTE: The actual number of words required by the TCLP to perform transaction processing depends on the number of concurrent users and the mix of transactions they are running. Refer to Appendix F in the Transaction Control Language Facility manual for the formulas used to calculate the TCLP memory requirements.

Table 5-5. Formats for Calculating File Memory Space

Memory Requirements in a MOD 400 LAF System		
File Reserved Exclusively	Memory Required for System Memory Pool (in words)	96 + (M x 32) where M is the number of buffers
	Memory Required for Online (User) memory Pool (in words)	64 + (M x $\frac{\text{CI size}}{2}$ where M is the number of buffers
Sharable Files	Memory Required for System Memory Pool (in words)	96 + M(32 + $\frac{\text{CI size}}{2}$ where M is the number of buffers
	Memory Required for Online (User) Memory Pool (in words)	64

Table 5-5 (Cont). Formats for Calculating File Memory Space

Memory Requirements in a MOD 400 LAF System		
File Reserved Exclusively	Memory Required for System Memory Pool (in words)	64 + (M x 32) where M is the number of buffers
	Memory Required for Online (User) Memory Pool (in words)	$32 + (M \times \frac{\text{CI size}}{2})$ where M is the number of buffers
Sharable Files	Memory Required for System Memory Pool (in words)	$64 + M(32 + \frac{\text{CI size}}{2})$ where M is the number of buffers
	Memory Required for Online (User) Memory Pool (in words)	32

CI size = control interval size

NOTE: For all system commands, the number of buffers per file is one (unless the user has acted to change this value).

The control interval size for each command is 512 (unless the user has allocated a file with a different control interval size).

Table 5-6. Memory Requirements for System Control Structures

Element	Word Size SAF	 LAF	Occurrences	Product
The number and size of the various control structures will vary depending on the installation's processing requirements. For each item in the "Element" column multiply the word size (using the SAF or LAF figure, as appropriate) by the number of estimated occurrences to obtain a "Product"; then add all the products.				
Batch GCB (includes LFT, LRT for LFT+LRN < 47(SAF) and LFN+LRN < 28(LAF) for every LFN+LRN above these limits, add 1 (SAF) or 2 (LAF) words.	96	128	0 or 1	
Batch TCB's (includes lead task, execution commands) Without SIP With SIP	 64 96	 96 128	 n...	
Batch Request Blocks	64	96	n	
Rollout Capability (Rollout File and Code)	367	447	0 or 1	
Each Active Disk Directory and File (include bound unit as files)	64	96	n	
Per Command_Input/User_Input File Operator Terminal (MDC) Disk(a) TTY (MLCP) PRU, CR	 32 160 64 96	 32 160 64 96	 n n n n	
Per User Disk File With Concurrency of "Sharable" or per "Device File"(b) Management (buffer size) Disk(a) PR, PRU CR, KSR	 CI+32 96 64	 CI+32 96 64	 n n	
Per Active File (a) (buffer control block)	32	32	n	
Per Bound Unit (File overhead - sharable or overlays)	128	192	n	

Element	Word Size SAF	Word Size LAF	Occurrences	Product
Per Sharable Bound Unit SBUS (SAF) = 18 + 5(number of overlays) + 4(number of EDEFed symbols) + code size	SBUS		Sum of SUBS's	
Tape Volume Name Block	32	32	n	
Concurrently Outstanding Mount Volume Requests	32	32	n	
Expanded Trap Save Areas for Each Task	128	192	n	
System Overhead	1504	1856	1	
Total Pool Size Without Fragmentation (sum of Product column)				
Fragmentation Allowance (10 percent of item immediately above)				

Legend:

GCB - Group control block
LFT - Logical file table
LRN - Logical resource table
TCB - Task control block
SIP - Scientific Instruction Processor
CI - Disk control interval

[a] Indexed sequential files must be counted as two files.

[b] Double the word size value if the "device file" is bidirectional and buffered.

Guidelines for Calculating Online and Batch Memory Pool Size.

The following table provides guidance in calculating the size of the online pools and the batch pool.

Table 5-7. Memory Requirements for Elements in Online or Batch Memory Pools

Element	Word Size SAF	Word Size LAF	Occurrences	Product
The number and size of various elements will vary depending on the installation's processing requirements. For each item in the "Elements" column, multiply the word size (using the SAF or LAF figure, as appropriate) by the number of estimated occurrences to obtain a "Product"; then add all the products.				
GCB (includes LFT, LRT)(a) for LFN+LRN < 47 (SAF) and LFN+LRN < 28 (LAF) For every LFN+LRN above these limits, add 1 (SAF) or 2 (LAF) words.	96	128	n	
TCB's (includes lead task and execution commands)(a)				
Without SIP	64	96	n	
With SIP	96	128	n	
GROUP REQUEST BLOCKS(a)	64	96	n	
Work Space for Commands Execution Commands	192	192	n	
Commands From Interactive Terminal	32	32	n	
TRB (exclude lead task, include execution commands)	64	96	n	
Per Command_Input/User_Input File				
Device	32	64	n	
Disk/Tape	64	64		
Per Error_Out/User_Out File				
Nondisk	64	64	n	
Disk	224	224	n	
Per User File (FCB)	32	64	n	
Per User Disk File With Concurrency of "Exclusive" Under Control of Data Management(b) (buffer size)	CI+32	CI+32	n	
Per associate File	32	32	n	

Table 5-7 (Cont). Memory Requirements for Elements in Online or Batch Memory Pools

Element	Word size		Occurrences	Product
	SAF	LAF		
Per User File Under Control Of Storage Management	32	32	n	
Per Nonsharable Bound Unit NSBUS (SAF) = 18 + 5(number of overlays) + 4(number of EDEFed symbols) + code size	NSBUS		Sum of NSBUS's	
Per Nonsharable Bound Unit (cont) NSBUS (LAF) = 25 + 6(number of overlays) + 5(number of EDEFed symbols) + code size		NSBUS	Sum of NSBUS's	
Any Bound Unit With Show Load Section	160	160	0 or 1	
User Requested Memory				
Total Pool Size Without Fragmentation (sum of Product column)				
Fragementation Allowance (10 percent of item immediately above)				

Legend:

GCB - Group control block
LFT - Logical file table
LRT - Logical resource table
TCB - Task control block
SIP - Scientific Instruction Processor
TRB - Task request block
FCB - File control block
CI - Disk control interval

[a] When calculating the size of the batch memory pool, do not include this item since, in the case of batch processing, it appears in the system memory pool (see Table 5-6).

[b] Indexed sequential files must be counted as two files.

Table 5-8. Memory Requirements for Message Facility and Checkpoint/Restart

Component	User Pool	System Pool	Occurrences
Message Facility			
System overhead		192	1
For each task group using Message Facility	64		n
For each mailbox	160	128	n
For each mailbox with disk queuing		128	n
Checkpoint/Restart			
Permanently allocated (for each group using Checkpoint/Restart)	320	256	n
Dynamically used during every Checkpoint (For more than 2 tasks in the group, add 192 for each additional task)	576		n
Dynamically used during every Restart		480	n

Example 1:

```
MEMPOOL S,,4096
```

In this example, a system memory pool of 4096 words is defined. Because a system memory pool is being defined, no pool_name argument is specified, but the second comma indicates omission of the argument.

Example 2:

```
MEMPOOL B,,12288
```

In this example, a batch memory pool of 12,288 words is defined. Because a batch memory pool is being defined, no pool_name argument is specified, but the second comma indicates omission of the argument. The batch pool is implicitly contained unless the pool size is followed by ",NC", which specifies not contained.

Example 3:

```
MEMPOOL E,AB,12768,,CD,1024,X
MEMPOOL E,EF,2048
```

In this example, three exclusive online memory pools are defined. The three pools constitute the set of exclusive online memory pools. The first MEMPOOL directive defines two exclusive online memory pools: (1) The first, whose pool name is AB, comprises 12,768 words; it cannot dynamically extend into the batch memory pool because argument 4 of the MEMPOOL directive is null. (2) The second, whose pool name is CD, comprises 1024 words; it <u>can</u> dynamically extend (if necessary) into the batch memory pool because argument 7 is X, indicating rollout. The second MEMPOOL directive defines an exclusive online memory pool named EF. This memory pool comprises 2048 words. It cannot dynamically extend into the batch memory pool because argument 4 is null.

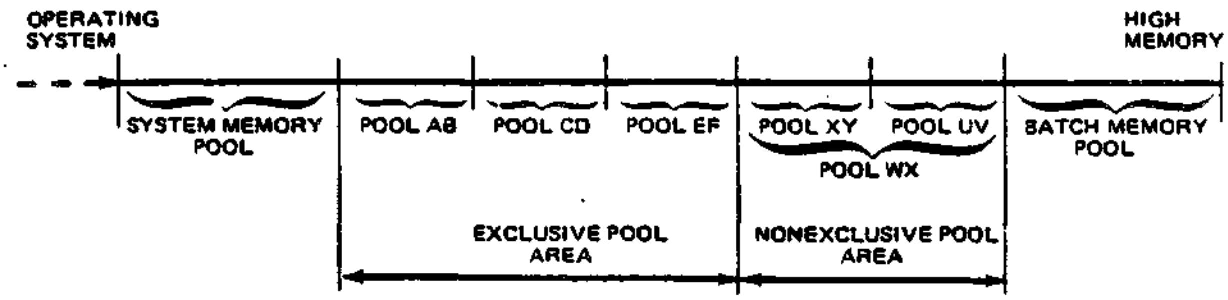
Example 4:

```
MEMPOOL ,XY,512,,UV,*
MEMPOOL ,WX,*,X
```

In this example, three nonexclusive online memory pools are defined (argument 1 in each MEMPOOL directive is null). Each MEMPOOL directive defines a pool set: the first pool set comprises two pools, the second pool set comprises one pool. Because two pool sets are nonexclusive, they define the same physical area of memory (i.e., they overlap). Any contention for space within this area of memory will be resolved by the system.

The first MEMPOOL directive defines two nonexclusive online memory pools: (1) The first, whose pool name is XY, comprises 512 words; it cannot dynamically extend into the batch memory pool because argument 4 of the MEMPOOL directive is null. (2) The second, whose pool name is UV, occupies all memory remaining after the pools with explicit sizes have been allocated (because argument 6 is an asterisk); pool because argument 7 is null. The second MEMPOOL directive defines a nonexclusive online memory pool area named WX. It <u>can</u> dynamically extend (if necessary) into the batch memory pool (argument 4 is X, indicating rollout).

The diagram below (not drawn to scale) indicates the memory pools established by the MEMPOOL directives in examples 1 through 4. Note that the two sets of nonexclusive online memory pools define the <u>same</u> physical area of memory.



Example 5:

```
MEMPOOL E,JJ,4096,PCUSX,RR,8000,SU
```

This example defines two memory pools: one named JJ, the other RR. Both pools are exclusive. The attributes for pool JJ are designated by the two arguments that follow the entry JJ. The first designates a size of 4096 words. The second specifies that the pool is protected, contained, and unpriviledged, that only one group can use it at a time (serial usage), and that it can extend into the batch pool area. Note that the elements of this argument can be entered in any order, and that no commas are inserted between elements.

Pool RR is assigned 8000 words of memory. It is neither protected, nor contained, nor expandable: it is unpriviledged and requires serial usage.

Example 6:

```
MEMPOOL   E,KK,4000,PS
   .
   .
   .
MEMPOOL   E,LL,KK,,MM,KK
```

This example illustrates the use of the renames option. Pool KK is declared first. The second directive specifies alternate names for this pool. Pools LL and MM have the same attributes as KK. Both are protected since KK is protected. All three names define the same 4000-word area in memory. Since KK is serial usage, LL and MM are serial usage also. The serial usage attribute is a function of the pool name. Since these three pool names describe the same memory pool, three groups can possibly use the pool area LL, and a group using MM could all be active at the same time. But, for example, two groups using the pool name KK could not both be active at the same time.

QUIT DIRECTIVE

Directive Name: QUIT

The QUIT directive must be the last directive in the file of CLM directives.

Format:

```
QUIT [VER]
```

Argument Description:

[VER]

The VER argument causes the version (date and time created) of each Configuration Load Manager module used to be typed out at the operator terminal. The format of the typeout is shown below under the example.

Functional Description:

The QUIT directive causes the Configuration Load Manager to cease reading CLM directives and to begin its loading phase. The following actions occur:

1. Any required final data structures are created.

2. If a communications environment has been defined, it is initialized.

3. Bound units (if any) identified in LDBU directives are loaded into the system area of memory.

4. System overlays (if any) identified in RESOLA directives are loaded into the system area of memory.

5. Memory pool descriptors are created, based on the contents of MEMPOOL directives. Each memory pool set is checked to ascertain whether it can fit into available memory space.

6. The Configuration Load Manager terminates.

Example:

```
QUI VER
```

This example causes the Configuration Load Manager to cease reading CLM directives, type out the version identifier of each CLM module used, and enter its loading phase. The typeout might appear as shown below.

```
($S) CLMVAR 80/10/03/0836
($S) CLMST1 80/09/14/0903
($S) CLMST2 80/10/02/0808
($S) CLMDEV 80/10/10/1206
($S) CLMMMU 80/08/17/1021
($S) CLMCOM 80/11/08/1530
($S) CLMCM2 80/10/15/2105
($S) CLMCM1 80/10/15/2104
($S) CLMCM3 80/10/15/2106
($S) CLMCM4 80/10/15/2107
($S) CLMFLT 80/10/02/0811
($S) CLM    80/09/21/0750 COMPLETED
```

RESOLA DIRECTIVE

Directive Name: RESOLA

The RESOLA directive identifies one or more system overlays that are to be loaded into the system area of memory, where they will remain resident for the duration of the configured system.

```
RESOLA   overlay_name[,overlay_name]...
```

Argument Description:

overlay_name

A string of up to six ASCII characters that identify a system overlay that is to be made resident for the duration of the configured system. Appendix B identifies the system overlays that can be specified in this argument and the funtion(s) provided by each one.

Functional Description:

Each RESOLA directive adds one or more system overlay names to a list of resident system overlays. During the Configuration Load Manager's loading phase, each name on the list is compared with a table of system overlay names. A match causes the indicated system overlay to be loaded into the system area of memory.

If no match occurs, an error is indicated. If one overlay name on a RESOLA directive causes an error, all other correctly specified overlays in the same directive can nevertheless be loaded.

Example:

```
RESOLA OYDMT,OZTSUB,OZTUTL
```

This example identifies three system overlays (these three are used for processing magnetic tape files) that will be loaded into the system area of memory, where they will remain resident for the duration of the configured system.

RLOCK DIRECTIVE

Directive Name: RLOCK

The RLOCK directive defines a pool of structures to be used by data management when record locking is requested at file reservation time. The size of the pool determines the total number of records that may be locked at one time.

One structure is used for each record to be locked. The size of each structure is six words for SAF and eight words for LAF. These structures are reusable after records become unlocked. The RLOCK directive must be included if the record lock facility is to be used. If more than one RLOCK directive is given, the last one takes effect.

Format:

```
RLOCK [init],[inc],[max]
```

Argument Description:

[init]

Specifies the number of initial record lock structures. The default is 15.

[inc]

Specifies the number of record lock structures that can be added at one time. The default is 5.

[max]

Specifies the total number of record lock structures permitted. The default is 30. An entry of 0 signifies that the total number of lock structures is unlimited. If the initial size or the increment size is greater than the maximum size, the following CLM message is generated.

133A RLOCK ARGUMENTS INCONSISTENT WITH EACH OTHER

NOTES:

1. The initial record lock pool is allocated in system memory at the time of the first record lock request.
2. The sizes of all three arguments are rounded up so that the amount of a memory allocated is a multiple of 32 words minus 2 for SAF, or 32 words minus 3 for LAF.

Functional Description:

The record lock facility of the file system provides multi-user interferance protection for records within shared disk files. For a detailed description of this facility see the Data File Organizations and Formats manual. This facility requires that a pool of record lock structures be created by means of the RLOCK directive. Because record lock entries are used and surrendered dynamically, the RLOCK pool need contain only enough entries to service the maximum number of records that will be locked at the same time. The following procedure shows how to calculate the size of the record lock pool for a system where a number of users perform simultaneous transactions.

Assume that each transaction consists of an update of three records. The file system will lock the control intervals (CI's) in which the three records reside, and for indexed sequential files it will also lock the record preceding each of the accessed records. Thus, a maximum of six CI's may be locked. However, two consecutive records are usually in the same CI; therefore three locked CI's is a good approximation for an indexed sequential file and the exact value for other file organizations.

Assume that ten users perform transactions but never more than four users access the data base simultaneously. For these assumptions, the initial number of record locks is 12 (4 x 3) and the natural increment is 3.

The accuracy of these estimates may be improved by taking into account the memory management features. Record lock entries are allocated in system memory, and system memory is obtained in blocks. (A multiple of 32 minus 2 words for SAF; a multiple of 32 minus 3 words for LAF). In a SAF environment, record lock entries are six words long. Therefore, for SAF, better estimates for initial and increment values would be 15 and 5.

Example 1: RLOCK

```
init = 15
 inc = 5
 max = 30
```

Example 2: RLOCK ,,45

```
init = 15
 inc = 5
 max = 45
```

Example 3: RLOCK 45,,45

```
init = 45
 inc = 5
 max = 45     Error because 45+5>45
```

```
Example 4: RLOCK 45,0,45

          init = 45
           inc = 0
           max = 45    No error

Example 5: RLOCK 20,5,60
           RLOCK ,4

          init = 15
           inc = 4
           max = 30    Because the last RLOCK
                       directive is the effective
                       one.

Example 6: RLOCK 20,5,60
           RLOCK 20,4,60

          init = 20
           inc = 4
           max = 60

Example 7: RLOCK ,,0

          init = 15
           inc = 5
           max = unlimited
```

SYS DIRECTIVE

Directive Name: SYS

The SYS directive defines a number of system variables.

Format:

```
             ┌⎛ SSIP ⎞┐
             │⎜ DSIP ⎟│
SYS [Hz],[scan-cycle],│⎨ CIP  ⎬│,[olan],[tsa],[irb][,,E]
             │⎜ SCIP ⎟│
             └⎝ null ⎠┘
```

Argument Descriptions:

[Hz]

Specifies the line frequency at which the system's realtime clock operates. Possible values are 60 (for 60 Hz) and 50 (for 50Hz). Model 20's and 30's should be configured with a value of 50 if the line frequency is 50 Hz, and 60 if the line frequency is 60 Hz. Model 40's and 50's should always be configured with a value of 60. The default value is 60 (the U.S. standard).

[scan_cycle]

Specifies the time, in milliseconds, between interrupts by the system's real-time clock. The default value is 50 (milliseconds).

The following values (in milliseconds) are possible for scan_cycle.

50 Hz Line	60 Hz Line
10	8
20	16
50	25
100	33
	50
	100

Millisecond intervals in cyclic clock request blocks should be expressed (if possible) in multiples of the scan_cycle. This practice ensures long-term accuracy of event synchchronization. Otherwise, the full deviation between the specified interval and the nearest scan_cycle multiple is always realized.

SYS

```
[ ( SSIP ) ]
[ ( DSIP ) ]
[ { CIP  } ]
[ ( SCIP ) ]
[ ( null ) ]
```

Specifies that scientific instructions and/or commercial instructions will be used in applications. Commercial instructions are processed by the Model 47 or Model 57 Central Processor in configurations so equipped; otherwise, commercial instructions are processed by a software simulator (CIPSIM). Scientific instructions are processed by the Scientific Instruction Processor if it is present; otherwise, scientific instructions are processed by the software simulator (SIPSIM).

Note that in configurations with a Model 47 or Model 57 Central Processor, the commercial extension to the interrupt save area is automatically created for each priority level except for those specified for this argument. The presence of a hardware Scientific Instruction Processor automatically creates the commercial and scientific extensions to the interrupt save area for each priority level except for those levels occupied by drivers, regardless of the option specified for this argument.

SSIP indicates that single-precision scientific instructions are to be processed. If the hardware Scientific Instruction Processor is not present, the single-precision SIP simulator (SIPSIM_SP) will automatically be loaded during the system startup. If the hardware Scientific Instruction Processor is present, it is used and the single-precision SIP simulator is not loaded. (SSIP is permissible for use with BES 2xx FORTRAN programs and FORTRAN programs without double precision).

DSIP indicates that double-precision scientific instructions are to be processed. If the hardware Scientific Instruction Processor is not present, the double-precision SIP simulator (SIPSIM) will automatically be loaded during system startup. If the hardware Scientific Instruction Processor is present, it will be used and the double-precision SIP simulator is not loaded. (DSIP is permissible for BES 203 -- and later -- FORTRAN programs).

CIP indicates that commercial instructions are to be processed. If the required central processor model is not present, you must include an LDBU directive that identifies the commercial simulator (CIPSIM). If the required central processor model is present, it will be used and the Commercial simulator should not be identified in an LDBU directive.

SCIP indicates that both scientific and commercial instructions are to be processed. If the hardware Scientific Instruction Processor is not present, you must include an LDBU directive that identifies the single-precision SIP simulator (SIPSIM_SP) or the double-precision SIP simulator (SIPSIM). If a Commercial Central Processor model is not present, you must include an LDBU directive that identifies the Commercial simulator (CIPSIM). If the hardware Scientific Instruction Processor and a commercial central processor are present, they will be used and the SIP and commercial simulators should not be identified in LDBU directives.

NOTE: If both SIP and commercial simulators are used, the order of the LDBU directives governs the order in which the simulators process a trap to the single trap vector they share. The simulator identified in the last LDBU directive will process the trap first.

null (no entry) indicates that no scientific or commercial instructions are to be processed or that any required hardware Scientific Instruction Processor and/or a commercial central processor model is present and will be used. (Thus the corresponding simulator is not used).

[olan]

Specifies the number of 512-word sysytem overlay areas to be created. The value for olan must be a decimal integer from 2 through 99. The default value is 1, indicating one 512-word area to be used for system overlays.

[tsa]

Specifies the number of trap save areas to be created in addition to the default number of 6 (system without communications), 7 (LAF system with communications), or 8 (SAF system with communications).

The size of each additional trap save area is the same as the size of each of the default trap save areas (viz., 64 words in a SAF system and 104 words in a LAF system).

You have the option of expressing this parameter symbolically by defining it with a VARIABLE directive. See the description of the VARIABLE directive elsewhere in this section.

[irb]

Specifies the number of intermediate request blocks to the created in addition to the default number of 20 for the system. Each intermediate request block is 8 words in a SAF system and 13 words in a LAF system.

You have the option of expressing this parameter symbolically by defining it with a VARIABLE directive. See description of the VARIABLE directive elsewhere in this section.

[E]

Specifies that expanded error messages (text in addition to code) are to be issued. If this argument is omitted, only the error code is issued.

Once this argument is specified, it remains in effect for the duration of the configured system, even if subsequent SYS directives appear with this argument null.

This argument is not meaningful if the system is bootstrapped from diskette: space limitations on diskette prevent the presence of the error message library file (EMLFILE).

Functional Description:

The SYS directive defines a number of system variables. If all of the SYS directive's default values are acceptable, it can be omitted. These default values are summarized below.

Hz - 60 Hz line frequency

scan_cycle - 50 milliseconds

SSIP/DSIP/CIP/SCIP - null (no SIP simulator or commercial simulator required)

olan - 1 system overlay area

tsa - 7 trap save areas

irb - 20 intermediate request blocks

E - error code only, no text

If multiple SYS directives are specified, only the last one is effective: exceptions: (1) all trap save areas and intermediate request blocks specified on <u>all</u> SYS directives are added to the system defaults of 7 and 20, respectively, and (2) if E is specified in <u>any</u> SYS directive, expanded error messages (text in addition to code) will be issued.

The size of the interrupt save area (and hence the size of the task control block, which includes the interrupt save area) depends on the CPU Model and SIP/CIP characteristics as shown in Table 5-9.

Table 5-9. Decision Table for Calculating TCB Size, Based on CPU Model and SIP/CIP Characteristics

Conditions:																									
Model 23, 33, or 6/34	Y					Y					Y					Y					Y				
Model 43, 47, 53, or 57		Y	Y	Y	Y		Y	Y	Y	Y		Y	Y	Y	Y		Y	Y	Y	Y		Y	Y	Y	Y
SIP hardware present?		N	N	Y	Y		N	N	Y	Y		N	N	Y	Y		N	N	Y	Y		N	N	Y	Y
Model 47 or 57		N	Y	N	Y		N	Y	N	Y		N	Y	N	Y		N	Y	N	Y		N	Y	N	Y
Null argument?	Y	Y	Y	Y	Y																				
SSIP argument?						Y	Y	Y	Y	Y															
DSIP argument?											Y	Y	Y	Y	Y										
CIP argument?																Y	Y	Y	Y	Y					
SCIP argument?																					Y	Y	Y	Y	Y
Resultant ISA Type:																									
Basic ISA	X					X																			
+ Model 40 and 50 registers		X					X																		
+ CIP context			X					X								X	X	X							
+ SIP context				X	X				X	X	X	X	X	X	X				X	X	X	X	X	X	X

TCB Size, Based on ISA Type

ISA Type	SAF Size (words)	LAF Size (words)
Basic ISA	64	Not applicable
+ Registers	64	96
+ CIP Context	96	96
+ SIP Context	96	128

NOTES:

1. Entry meanings: Y - yes N - no X - ISA type
2. The basic ISA ends with the M1-register save word.
3. The ISA for Model 43, 47, 53, or 57 registers includes the basic ISA, save words for the M2- through M7-registers, and save word(s) for the stack address register.
4. The CIP context includes all elements in item 2, plus the CIP-specific context area of 2 words (SAF) or 3 words (LAF).
5. The SIP context includes all elements in item 3, plus the SIP-specific context area of 13 words.
6. All TCB sizes are rounded up to the nearest multiple of 32 words.

Example:

```
SYS ,25,SSIP,5,,,,E
```

This example defines the system variables as shown below. (Assume that only one SYS directive is used).

- The system line frequency is 60 Hz (default).
- The scan-cycle of the system's real-time clock is 25 milliseconds.
- The single-precision SIP simulator (SIPSIM_SP) will be loaded to process scientific instructions, unless the hardware Scientific Instruction Processor is present.
- Five system overlay areas will be created.
- Six trap save areas will be created (default if no communications).
- Twenty intermediate request blocks will be created (default).
- Error codes will be expanded to include message text.

VARIABLE DIRECTIVE

Directive Name: VARIABLE

The VARIABLE directive defines a single-character symbolic expression (chosen from the letters A through Z) that may be specified in certain CLM directive statements.

Variables can define values for the fields in the SYS directive representing additional trap save areas or intermediate request blocks, or memory pool sizes in a MEMPOOL directive. A variable <u>must</u> be defined by a VARIABLE directive <u>before</u> it can be specified in other CLM directives.

Format:

```
VARIABLE variable=expression
```

Argument Description:

variable

A single character designation, upper case only, chosen from the letters A through Z.

expression

Any valid expression that defines the value of the variable. Expression may equal any of the following:

- An integer having a value between - 32767 and +32767.
- KSR (represents the total number of MDC-connected terminals finally configured using the KSR value)
- CON (represents the total number of MDC-connected terminals finally configured with the CON value)
- TTY (represents the total number of communications devices finally configured by DEVICE TTY directives)
- VIP (represents the total number of communications devices finally configured by DEVICE VIP directives)
- ATD (Represents the total number of devices finally configured by DEVICE ATD directives)

. $expression_1 \begin{Bmatrix} + \\ - \\ * \\ / \end{Bmatrix} expression_2$

An arithmetic combination of two expressions (chosen from this list) using addition (+), subtraction (-), multiplication (*) or division (/). The value of an expression is always an integer: all fractional values are truncated, <u>not</u> rounded. For example, 5/3 will be passed to the system as having a value of one (1).

. (expresion (1))

expression, is to be evaluated before being combined with any other expression term. The expression is always evaluated by the system as an integer: all fractional values are truncated, <u>not</u> rounded. For example, 5/3 will be passed to the system as having a value of 1. At each step of the evaluation of an expression, fractional values are truncated, i.e., this is strictly integer arithmetic.

. $\begin{Bmatrix} + \\ - \end{Bmatrix}$ expression

Specifies a signed value, plus or minus, for the expression. Note that "+expression" has the same value as "expression".

. x (where x is the name of an already defined variable)

Variables may be used when specifying the number of trap save areas and intermediate request blocks in a SYS directive, or the size of a memory pool in a MEMPOOL directive. Using a VARIABLE directive, you can assign symbolic values for these fields that will be subsequently resolved by the system. By specifying variables in SYS or MEMPOOL directives, the system builder creates a generalized CLM file that may be used by several similar but not identical hardware configurations.

Example 1:

```
         .
         .
         .
VARIABLE A=TTY*3000
MEMPOOL  ,T1,A
MEMPOOL  S,,*
         .
         .
         .
```

Note that in all cases, a variable must be defined in the CLM file before it can be used in subsequent directives. The variable A has been defined to be equal to the total number of TTY terminals finally configured, multiplied by 3000. In the MEMPOOL directive, A represents the 3000 words of memory that will be reserved for each

TTY configured. The first MEMPOOL directive defines the name and size of the memory pool that has reserved 3000 words of memory for each TTY terminal. The second MEMPOOL directive allocates all remaining memory to the system memory pool.

By defining memory pools with variables, the system builder can accurately account for as many or as few devices as are actually configured. He need not know the channel number assigned to each device, the memory pool size, or the number of devices actually configured. The system assigns these characteristics accurately and automatically.

Example 2:

```
          .
          .
          .
VARIABLE B=TTY*3
VARIABLE C=TTY*2
SYS  ,,,,C,B
          .
          .
          .
```

In this example, two variables have been defined for the trap save area and intermediate request block fields in the SYS directive. C is defined as the number of trap save areas that will be allocated for each TTY terminal, in addition to the default number. (The value of C equals twice the number of TTY terminals configured). C specifies that each terminal will have twp trap save areas in addition to the default number. B is defined in the SYS directive as the number of intermediate request blocks that will be allocated for each TTY terminal, in addition to the default number. (The value of B equals three times the number of TTY terminals configured). B specifies that each terminal will have 3 intermediate request blocks in addition to the default number.

Example 3:

Sample CLM File

```
1 { VARIABLE T=TTY+VIP
  { VARIABLE I=KSR+10
  { SYS ....I,T,,E
    DEVICE DSK00,6,10,X'400
    DEVICE DSK01,7,11,0
    DEVICE FCD00,8,12,0
    DEVICE RCD00,9,12,0
    COMM 13
  { DEVICE TTY01,14,14,0
  { DEVICE TTY02,15,15,0
  { DEVICE TTY03,16,16,0
2 { DEVICE TTY04,17,17,0
  { TTY 14,14,0,0,300
  { TTY 15,15,0,0,300
  { TTY 16,16,0,0,300
  { TTY 17,17,0,0,300
3 { VARIABLE Z=TTY*500
  { MEMPOOL ,L0,Z
4 { MEMPOOL S,,*
  { QUIT
```

Example 3 Comments:

1. Variable T and I have been defined in VARIABLE directives replacing constant values in the SYS directive. T equals the total number of TTY and VIP terminals configured. I equals 10 plus the number of KSR terminals configured. In the SYS directive, T replaces some constant value in the intermediate request block field. In this context, T specifies one additional trap save area for each TTY or VIP configured; I specifies that the number of additional intermediate request blocks equal 10 more than the number of KSR terminals configured.

2. A maximum of four TTY terminals may be configured depending on the number of available channels (with asynchronous line adapters). All terminals have floating channel assignments and are configured in the order in which they appear in the CLM file.

3. The variable Z is defined such that 500 words of memory are allocated for each configured TTY terminal). (Z equals 500 times the number of configured TTY terminals). In the MEMPOOL directive that follows, Z specifies that pool L0 allocates 500 words of memory for each TTY configured. The size of pool L0 could be 500, 1000, 1500, or 2000 words, depending on the final number of TTY teminals actually configured.

4. This MEMPOOL directive allocates all unreserved memory to the system memory pool.

SECTION VI

CLM DIRECTIVES FOR A COMMUNICATIONS CONFIGURATION

This section describes the CLM directives used to define the communications environment of the system software. Communications CLM directives may be entered from the CLM_USER file or from any file or device specified in a CLMIN directive. There after, a system startup can be performed.

If your installation includes communications (MLCP/DLCP-connected) devices, you should have the Communications Processing manual available for reference.

The communications-related CLM directives cause the following functions to be performed:

- Data structures (i.e., tables) are established corresponding to the communications hardware available to your system.
- The following bound units are loaded into the central processor's main memory:
  - Communications supervisor and multiline/dual line communications processor (MLCP/DLCP) driver.
  - One or more line protocol handlers (i.e., TTY, VIP, ACTD, ATD, STD, BSC, PVE, HASP, RCI, or user-written).
- The following elements are loaded into the RAM (random access memory) of one or more Communications controllers:
  - Data set channel control program.
  - Channel control programs of one or more line protocol handlers.

Table 6-1 summarizes the communications-related CLM directives, which are described in alphabetic order in this section. Table 6-2 provides detailed information on physical devices supported by various directives.

Table 6-1. Summary of Communications-Related CLM Directives

Directive	Meaning
ACTD	Identifies a station on a line serviced by the Asynchronous Character Terminal Driver.
ACU	Defines an Auto Call Unit and associates the ACU channel with a data communications channel. Optionally, provides one or more telephone numbers to be used in establishing a connection for the associated data communications channel.
ATD [a,b,c]	Identifies a station on a line serviced by the asynchronous terminal device (ATD) line protocol handler.
BSC	Identifies a station on a line serviced by the BSC (binary synchronous communications) line protocol handler.
COMM	Establishes the priority level(s) at which the MLCP/DLCP interrupts the central processor.
DEVICE	Indicates that the designated communications station is to be accessible through the file system interface.
EQLRN	Defines multiple logical resource numbers (lrn's) for the same physical communications device.
HASP	Identifies a remote workstation serviced by the HASP protocol.
H3270	Identifies a station on a line serviced by the BSC 3270 protocol.
LPHn	Identifies the first (or only) station on a line serviced by a user-written line protocol handler.
LPHDEF	Indicates nonstandard table sizes for channels and stations controlled by a user-written line protocol handler.
MODEM	Defines a nonstandard modem type.
POLIST	Identifies the start of the pool list on a line serviced by the synchronous terminal device line protocol handler. The line is identified by a STDLN directive.
PVE	Identifies a polled VIP emulator "station" on a line serviced by the VIP line protocol handler.

Table 6-1 (Cont). Summary of Communications-Related CLM Directives

Directive	Meaning
RCI	Identifies a remote batch terminal serviced by the RCI protocol.
ROP	Indicates that a receive-only printer is connected to a station on a line serviced by the synchronous terminal device line protocol handler. The line is identified by an STDLN directive and the station is identified by an STD directive.
STAPOL	Defines a sequence of station poll addresses to be added to the poll list of a line serviced by the synchronous terminal device (STD) line protocol handler. The line is identified by an STDLN directive.
STD [d]	Identifies a station on a line serviced by the synchronous terminal device (STD) line protocol handler. The line is identified by an STDLN directive.
STDLN [d]	Identifies a line serviced by the synchronous terminal device (STD line protocol handler.
STATION	Identifies the second or subsequent station on a line serviced by a user-written line protocol handler.
STTY	Specifies the file characteristics of a device that is not a disk device or a unit-record device.
TTY [c]	Identifies a station on a line serviced by the TTY (teleprinter) line protocol printer.
VDAM	Incorporates the Display Formatting and Control software component in the configuration.
VIP [d]	Identifies a polled or nonpolled synchronous visual information projection (VIP) terminal on a line serviced by the VIP line protocol handler. Optionally, identifies an ROP (receive-only printer) station on the same line.
VROSY [a]	Identifies a PRU1001/1003/1005 device on a line serviced by the ATD line protocol handler.

Table 6-1 (Cont). Summary of Communications-Related CLM directives

Directive	Meaning
VTTY [a]	Identifies a teletype-like device on a line serviced by the ATD line protocol handler.
V7200 [a,b]	Identifies a VIP7200 or VIP7205 device on a line serviced by the ATD line protocol handler.

[a] The ATD directive is intended as a replacement for the VROSY, VTTY, and V7200 directives. The protocol handler called by the ATD directive offers additional support capabilities e.g., block mode support of VIP7801/02 devices, support of the VIP7207 device) not offered by the line protocol handler called by the VROSY, VTTY, and V7200 directives. The VROSY, VTTY, and V7200 directives can still be used.

[b] The ATD directive or the V7200 directive must be specified if VIP7200 or VIP7801/02 terminals are to be used for forms processing using CII HONEYWELL BULL Display Formatting and Control Facility.

[c] The TTY directive and the ATD directive support asynchronous terminal devices. The line protocol handler called by the TTY directive supports the physical terminal only in teleprinter mode. The TTY line protocol handler does not provide block mode support of VIP7801/02 devices; neither does it provide support for VIP7200 devices and VIP7801/02 devices in forms processing mode. (For these functions the ATD directive is required). ATD line protocol handler does not provide the transparent I/O or single character mode functions of the TTY line protocol handler.

[d] The STD directive and the VIP directive support synchronous terminal devices. The line protocol handler called by the STD directive offers additional support capabilities (e.g., line protocol handler called by the VIP directive.

Table 6-2. CLM Directives and Supported Communications Devices

Directive	Devices Supported
ACTD	VIP 7200 DKU 7001 DKU7002 + auxiliary badge + auxiliary printer
ATD	VIP 7200 VIP 7205 VIP 7801 VIP 7802 PRU 1001/1003/1005 TWU 1001/1003/1005 Receive-only printers: PRU 1001/1003/1005
BSC	Level 6 central processor
HASP	IBM central processor
H3270	IBM central processor
PVE	Level 6 central processor
RCI	Level 6 central processor
STD	VIP 7700 VIP 7700R VIP 7705R VIP 7760 VIP 7804 VIP 7805 Receive-only printers: TN 300 TN1200 PRU 1003/1005
TTY	VIP 7100 VIP 7200 VIP 7105 VIP 7205 VIP 7801 VIP 7802 TWU 1001/1003/1005
VIP	VIP 7700 VIP 7700R VIP 7705R VIP 7760 VIP 7805 VIP 7804 Receive-only printers: TN 300 ASR-33 TN 1200 ASR-35 PRU 1003/1005
VROSY	PRU 1001/1003/1005

Table 6-2 (Cont). CLM Directives and Supported Communications Devices

Directive	Devices Supported
V7200	VIP 7200 VIP 7801 VIP 7205 VIP 7802
VTTY	ASR-33 KSR-33 ASR-35

You should observe the following rules for arranging communications-related CLM directives:

1. The COMM directive must precede all other communications-related CLM directives.

2. If an LPHDEF directive is used in association with a user-written line protocol handler, the LPHDEF directive must precede all related LPHn directives.

3. If STATION directives are used in association with an LPHn directive (in cases where a user-written line protocol handler drives more than one station per line), the STATION directives must immediately follow the related LPHn directive.

4. The DEVICE directive should follow related LPHDEF, STATION, and LPHn directives and related POLIST, STAPOL, ROP, STDLN, and STD directives.

5. If an ACU directive is used, the station whose lrn appears as an argument in the ACU directive must be defined in the directive that immediately precedes this ACU directive.

6. When polled VIP's are connected to a communications line, the VIP directives for the stations on the line must be consecutive.

7. When polled VIP emulator (PVE) stations are associated with a communications line, the PVE directives for all the stations on the line must be consecutive.

8. The value assigned to the communications interrupt level (by the COMM directive) is normally restricted to the range 7 through 57. At initial startup, the bootstrap device is assigned level 6 by the system. If the level assigned to the bootstrap device is changed to a lower level (higher number) by a DEVICE directive, the communications interrupt level can be specified as 6. The level that is assigned to a communications device associated with a COMM directive is normally 8 through 58.

9. When an MLCP/DLCP operator console is configured, it must be assigned LRN 0.

10. Configuring a line serviced by the STD line protocol handler requires at least the directives STDLN, STD, POLIST, and STAPOL. An STD line may also use a ROP directive. The following diagram illustrates sequence rules for STD directives. Brackets enclose optional directives. One set of these directives must be specified for each line:

    STDLN   must specify one for each line

    STD 1
    [ROP]   specify one for each   } specify one set for each
               station, if needed   } station on the line: all
                                        } must precede POLIST

    STD 2
    [ROP]
      .
      .
      .
    POLIST   must specify one for each line

    STAPOL   must specify one or more for
               each line: all STAPOL
               directives should be specified
               in the order in which stations
               are to be polled

    [DEVICE]   (optional) specifies that the device will be
               accessible to the file system

11. As a general rule, two line protocol handlers are allowed per MLCP/DLCP.

12. The STTY command keyword format, as described immediately following the STTY directive, may be used as an alternative to the STTY directive format in your CLM file. The STTY command format provides additional features not supported by the STTY directive format. For example, if you wish to specify a terminal as being automatically reconnectable following a power failure or line drop, you must configure the terminal using the STTY command format in your CLM file. (For a complete description of this capability, see Appendix D).

13. Each directive for a communications device must reference a unique logical resource number (lrn). If you wish to specify multiple communications directives which reference the same physical device, you should use the EQLRN directive. The EQLRN directive is described later in this section.

TOPICS RELATED TO COMMUNICATIONS DIRECTIVES

The following paragraphs describe:

- How to assign channel numbers to your communications devices, or, how to let the system dynamically assign channel numbers
- How to modify a terminal's line length
- How to let the system dynamically assign a terminal's line speed
- The amount of resident code required for communications modules.

You should review each of these topics before you create your CLM directive file.

Assigning Channel Numbers

Channel control programs are loaded into the MLCP on the basis of channel numbers you specify for communications devices. You should observe the following guidelines when assigning channel numbers to your communications devices:

a. As a general rule, no more than two line protocol handlers can be associated with a given MLCP.

b. The MLCP with which a particular line protocol handler is associated is determined by the channel number(s) you enter for communications devices. The system checks the first six bits of the hexadecimal channel number you specify to determine LPH allocation.

This range of channel numbers specified for a device	FC00 - FF80	causes the LPH to go into	first MLCP
	F800 - FB80		second MLCP
	F400 - F780		third MLCP
	F000 - F380		fourth MLCP
	EC00 - EF80		fifth MLCP
	.		.
	.		.
	.		.

c. Each channel number must be a 4-digit hexadecimal number ending in 00 or 80. Note that channel numbers lower then 0400 are unavailable for communications devices.

DYNAMICALLY ASSIGNED CHANNEL NUMBERS

For MLCP/DLCP-connected devices, channel numbers may be dynamically assigned or "floated" in TTY, VIP, VROSY, VTTY, ATD, V7200, or DEVICE directives.

Assigning zero channel numbers to directives in the CLM file relieves the system builder of the need to know the channel assignments for each installation that will receive a copy of file system software.

For each communications device configured, you have the option of specifying a channel number in the appropriate directive or allowing the system to dynamically assign or "float" channel assignments at startup time. To float a channel, you specify the channel number parameter in the appropriate directive with a single zero (0). The system then assigns directives with floating channels to the first available devices on the bus. Note that if you specify a floating channel and the system cannot locate an appropriate device, no error message is issued; the system ignores this directive.

If your system includes device adapters but not associated devices, it is recommended that explicit channel numbers be assigned to all devices within the device class.

Channel numbers are assigned as soon as the first floating channel is encountered in the CLM file. To prevent devices with floating channels from inadvertently preempting devices whose channel numbers are explicitly specified, all directives of a given type with explicitly specified channel numbers should precede all directives of that tyoe with floating channels in the CLM file. All active channels should have higher bus addresses than all active channels to prevent the system fom inadvertently assigning a device to an inactive channel.

For MLCP/DLCP-connected devices, the first available channel with the highest bus address and appropriate channel adapter (asynchronous or synchronous) will be assigned to the first floatable channel in a communications directive, and so on.

NOTE: Assigning floatable channels to communications devices is recommended only if all devices will be configured identically (if, for example, a system specifies that all asynchronous devices will be directly-connected VIP 7200's with identical line speeds, DEL characters, stop bits, and parity). The system cannot distinguish between different types of MLCP/DLCP-connected devices.(1)

(1)
The M4-SYSDEF program does not permit you to configure MLCP/DLCP connected devices with floating channel numbers. M4_SYSDEF requires the system builder to supply all channel numbers for communications devices.

If a system is configured with devices containing floatable channel numbers, the system, builder may determine all channel numbers assigned by the system as soon as possible after a system initialization. The system operator can issue the command STS -ALL to list each device and its channel number assignment.

See Section V for an example illustrating the use of floating channel number assignments.

Modifying Terminal Line Length

The line length specified for a terminal or other input device at system building time overrides the command processor line length of 127 characters. For example, at terminal configured with a line length of 80 characters, you cannot enter a continuous command line longer than 80 characters unless you type an ampersand (&) at the end of the line and then continue on one or more additional lines. You can alter the terminal's line length by using the set terminal characteristics (STTY) command (described elsewhere in this manual) after system initialization or by modifying the STTY directive in your CLM file.

Terminal Line Speed Selection Capability (Asynchronous Terminals Only)

When you specify an ATD, TTY, LPHn, VTTY, or V7200 directive, you have the option of deferring selection of an asynchronous terminal's line speed (specified by the fifth parameter) until the terminal comes online (1). To defer selection of a terminal's line speed, you specify the fifth parameter as either "HI" or "LO" in one of the above-mentioned directives. Choosing this option frees you from specifying an exact line speed for every asynchronous terminal in your configuration. The interactive system building program, M4_SYSDEF, incorporates this deferred speed selection capability.

In order to allow the system to determine the terminal's line speed, the operator presses the RETURN key on the terminal's keyboard when the terminal comes online. If, after several seconds, normal terminal operations have not begun, the operator should check for one of the following problems:

(1)
If you choose this option, an Auto Call Unit must not be configured for the terminal.

1. The terminal's line speed may not be within the range specified in the appropriate CLM directive. The operator may be able to adjust the terminal's line speed to fall within the range specified by the directive. The operator should then press the RETURN key. If, however, the terminal's line speed is fixed, adjustments must be made to the appropriate CLM directive. You may be required to change the designation of the terminal's line speed in the appropriate CLM directive.

2. If the carriage return character was garbled and the system could not determine the line speed of the terminal, the operator should press the RETURN key again.

3. If the terminal's modem has gone offline, the operator must dial up the line again.

If the system fails to receive a character transmission within 60 seconds, the terminal will be disconnected.

Resident Code Requirements for Communications Modules

The following table defines the number of words of resident code required for various communications modules; the table also list the memory required in the MLCP/DLCP for associated channel control programs (CCP's). Memory requirements for noncommunications modules are defined in Section V.

Table 6-3. Communications Memory Requirements

Component	Resident Memory (SAF) (in words)	Resident Memory (LAF) (in words)	MLCP Memory Required for CCP (1) (in bytes)	DLCP Memory Required for CCP (2) (in bytes)
Comm Supervisor	2.4K	2.5K	N/A	N/A
Autocall	.4K	.4K	128	206
Speed select	.2K	.2K	N/A	N/A
TTY Line Protocol Handler	1.7K	1.7K	1429	1772
VIP Line Protocol Handler	2.4K	2.5K	1346	1760
BSC Line Protocol Handler	2.6K	2.6K	1100 EBCDIC 870 ASCII	1574 EBCDIC 1126 ASCII
PVE Line Protocol Handler	2.3K	2.3K	742	1182
RCI Line Protocol handler	2.0K	2.0K	188	235
ATD Line Protocol Handler	5.9K	6.1K	824	1212
ACTD Line Protocol Handler	-	5.3K	1080	-
Auxiliary subsystems	-	0.15K	-	-
Badge	-	0.6K	-	-
Printer	-	.1K	-	-
STD Line Protocol Handler	3.6K	3.5K	800	1226
3270 Line Protocol Handler	2.1K	2.1K	1158	1758
HASP	6.8K	7.2K	282	446

NOTE: Sizes do not include patch space sizes. Patch space is only required as patches are added.

(1)
2942 bytes of RAM memory are available in the MLCP for the loading of channel control programs. Any combination of the various CCP's can be loaded into the MLCP provided the total memory does not exceed 2942 bytes.

(2)
3844 bytes of RAM memory are available in the DLCP for the loading of channel programs. Any combination of the various CCP's can be loaded into the DLCP provided the total memory does not exceed 3844 bytes.

ACTD

ACTD DIRECTIVE

Directive Name: ACTD

The ACTD directive identifies a station on a line serviced by the asynchronous character terminal device line protocol handler. This directive is used when configuring stations of the following types:

DKU 7002, DKU 7001, VIP 7200, VIP 7100 or TTY device.

See Appendix R to configure a station serviced by ACTD line protocol handler.

ACU DIRECTIVE

Directive Name: ACU

The ACU directive identifies an Auto Call Unit and associates the ACU channel with a data communications channel. The ACU directive must immediately follow the directive describing the station whose lrn is specified in the ACU directive. The ACU directive cannot be associated with any data communications channel that supports the speed select option.

The ACU directive permits the addition of telephone numbers to a list of numbers maintained for a data communications channel. The list of telephone numbers for a data communications channel can be unlimited and has the following format:

```
Entry 0  empty (initially)

[Entry 1  phone_number 1]

[Entry 2  phone_number 2]

[etc.]
```

The Auto Call Unit dials each number in the list three times at 40-second intervals until the list is exhausted or a connection is made.

The first ACU directive that relates an Auto Call Unit to this data communications channel causes an empty entry 0 to be established; this entry in the list may be loaded and reloaded, as desired, by means of an SDL (set ACU telephone number) command or $SDL macro call. The first ACU directive also creates an entry in the table (starting with entry 1) for each telephone number (if any) specified in the directive. Any subsequent ACU directive relating the same ACU to the same data communications channel causes one or more <u>additional</u> entries to be added to the list: the additional entries are added to the end of the list in the order in which the telephone numbers appear in the ACU directive.

Format:

```
ACU  lrn,level,X'acu_channel'['phone_#1'[,'phone_#2'[,...]]]
```

Argument Desctiption:

lrn

The logical resource number of any station on the data communications channel with which this Auto Call Unit is associated.

level

The priority level of the station whose lrn appears in the lrn argument of this directive.

X'acu_channel'

A 4_digit hexadecimal number (from X'400' to X'FF80') specifying the channel number of the Auto Call Unit.

Bits	0 through 9	-	The 10-bit channel address of the send or receive channel on the line.
Bits	10 through 15	-	Must be set to zero.

['phone_#n']

A string of 1 to 16 ASCII characters chosen from the set 0 1 2 3 4 5 6 7 8 9 - (separator) * #.

FUNCTIONAL DESCRIPTION:

n ACU directive causes the Auto Call Unit to initiate a line connection with a remote auto answer data set. When the software issues a connect order and bit 2 of the IORB is set to one, the Auto Call Unit attempts to dial a line using the list of telephone numbers established at configuration time. The Auto Call Unit dials each number in the list three times at 40-second intervals until a connection is made or the list is exhausted.

Example:

```
TTY  26,8,X'FC00'

ACU  26,8,X'FD00','1-555-240-0281'
```

In this example, an Auto Call Unit on channel FD00 is associated with the data communications channel (FC00) servicing a TTY whose logical resource number is 26 and whose priority level is 8. Since this is the first ACU directive for this Auto Call Unit, the telephone number in the ACU directive is established as entry 1 in the list of telephone numbers for the indicated data communications channel (FC00). Note that the ACU directive <u>immediately follows</u> the TTY directive that describes the station whose lrn appears in the ACU directive.

ATD DIRECTIVE

Directive Name: ATD

The ATD directive identifies a station on a line serviced by the asynchronous terminal device line protocol handler. This directive is used to configure asynchronous terminals as part of the communications system. The ATD directives must follows the COMM directive in the CLM file. You can float the channel number assignments in this directive, if you wish.

Format:

```
ATD  lrn,level,X'channel',[modem],[speed],['device-type']

     [del],[stop-bit][,parity]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the ATS line protocol handler processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive.

X'channel'

You may specify a single zero (0) to float this channel assignment or a four-digit hexadecimal number (from X'040n' to X'FF8n'), that specifies the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9	- The 10-bit channel address of the send or receive channel on the line.
Bits 10 through 13	- Must be set to zero.
Bits 14 and 15	- Specifies n, the priority level at which a

communications line interrupts the central processor. n may have a value of 0,1,2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct Connect.

1 - Bell 1xx-type modem (103A, etc.). Both data-set-ready and carrier-detect signals are required for a connection; absence of these signals is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive," earlier in this section).

The default value is modem type 1.

[speed]

The data rate in bits per second.

For an asynchronous line with a communications-pac whose id is 2108(16), use one of the following values for speed:

	50	300	2400
	75	600	3600
(Default)	110	900	4800
	134	1200	7200
	150	1800	9600

For an asynchronous line with a communications-pac whose id (in hexadecimal notation) is 2100, 2110, or 2118, or 3118, use one of the following values for speed:

	50	200	1800
	75	300	2000
(Default)	110	600	2400
	134	1050	4800
	150	1200	9600

The deferred speed selection option functions with the second set or adapter types only.

NOTE: If the data rate is 134.5, specify 134.

You have the option to defer selection of the line speed until the terminal comes online. You select this option by specifying HI or LO.

'HI'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 1200 through 9600 bits per second.

'LO'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 110 through 1200 bits per second.

For more information about the line speed selection capability, refer to the discussion under "Topics Related to CLM Directives" near the beginning of this section.

['device-type']

Specifies the type of terminal used. If this argument is not specified, the default is TTY. Possible values are:

Value	Physical devices supported
7200	VIP 7200, VIP 7205
7801	VIP 7801, VIP 7802
D7200	VIP 7207
PRU	PRU 1001, PRU 1003, PRU 1005, TWU 1001, TWU 1003, TWU 1005
TTY	An asynchronous terminal that is to be supported as teleprinter-compatible
TN300	Terminet 300, PRU 1004

[del]

In TTY mode, a head-of-form sequence (pre-order control) consisting of three LF's and the number of DEL characters specified by this argument is generated if the following conditions are met. (If the conditions are not met, no head-of-form sequence is generated).

- The first byte of the application's buffer must be designated as a control byte; i.e., bit 4 of the IORB's device specific word (I_DVS) must be set to 0 to write time.
- The control byte must specify that a head-of-form sequence is to be generated; i.e., bit 3 of the control byte must be set to 1.
- Device type PRU is not specified. If it is specified, a true head-of-form is issued.

ATD

In the field or TTY modes, an LF in an end-of-message sequence (post order control) is followed by the number of DEL characters specified by this control argument.

The EOM sequence is controlled by the B- and C-bits of the IORB's device-specific word I_DVS, as specified by the application at write time. The TTY line protocol handler sends an EOM sequence according to the following B- and C-bit values:

I_DVS Bits		
B	C	EOM Sequence
0	0	CR
0	1	None
1	0	CR,LF,DEL characters
1	1	LF,DEL characters

At read time, the application can specify the same B- and C- bit values in order to send an EOM sequence back to the terminal when the message is successfully received.

Note that an LF character in a pre-order control specified by the I_CON word of IORB is never followed by a DEL character, i.e., this control argument has no effect in this case.

This argument allows you to select 1 to 32 DEL characters. The default for each type of device is as follows:

Device-Type	Number of DEL Characters
7200	0
7800	0
PRU1001/1003/1005	1
TTY	1
PRU1004	32

[stop-bit]

Specifies the number of stop bits that are to follow each character. A value of 1 or 2 can be chosen for each device. Default values are as follows:

Number of Stop Bits
1 (For speeds greater than 110 bits per second)
2 (For a speed of 110 bits per second or less)

[parity]

Specifies the type of parity ("ODD" or "EVEN") to be used. The default is EVEN.

Example:

```
ATD  15,17,X'F800',0,150,'7200'
```

In this example a VIP 7200 terminal is connected to a line serviced by the ATD line protocol handler. The device defaults for del, stop-bit, and parity are assumed.

BSC DIRECTIVE

Directive Name: BSC

The BSC directive identifies a station serviced by the binary synchronous communications line protocol handler.(1)

Format:

```
BSC  lrn,level,X'channel',[modem],[primary/secondary],
     [character_set][,multi_block_count]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 252. A program may use this number to identify the station when it requests an input/output operation to the station.

level

This priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 62; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

(1)
In the station is to be accessible through the file system interface, an appropriate DEVICE directive must be "paired" with te BSC directive (See "DEVICE Directive," later in this section). If input/output to the station is to be asynchronous, the B (buffered) argument must be included in the DEVICE directive.

X'channel'

A four-digit hexadecimal number (from X'040n to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specified n, the priority level at which a communications line interrupts the central processor. n may have a value of 0,1,2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

2 - Bell 2xx-type modem (201A, 208B, etc). The data-set-ready signal is needed for a connection: absence of this signal is a disconnected.

3 a greater - User-defined modem type (see "MODEM directive," later in this section).

The default value is modem type 2.

[primary/secondary]

Values may be specified as P or S: indicates whether this is a primary or secondary endpoint of the transmission. A primary endpoint (the default) has priority in contention mode.

[character-set]

One of the following may be specified:

AS ASCII (the default).

EB EBCDIC.

The user is responsible for using the correct character set. The BSC protocol does not perform character translation.

[multi_block_count]

An integer from 1 to 7 specifies the maximum number of data blocks for a single transmission in multi_block mode. The default is that multi-block mode is not used.

Example:

```
BSC  29,6,X'FD00',2,S,EB

DEVICE BSC00,29,6,X'FD00',HOST,,B
```

In this example, line FD00 is used for communications with another computer. Modem type 2 is used for the line. The Level 6 computer is the secondary endpoint on the line. A file system interface is established for the station by the DEVICE directive. Multiblock mode is not used.

COMM DIRECTIVE

Directive Name: COMM

The COMM directive is mandatory in a system that includes communications. It specifies from one to four priority levels at which communications lines can interrupt the central processor. The COMM directive must precede all other communications-related CLM directives.

Format:

```
COMM level_0 [,[level_1][,[level_2][,[level_3]]]]
```

Argument Description:

level_0
[level_1]
[level_2]
[level_3]

The four possible priority levels at which a communications line interrupts the central processor. At least one priority level must be specified. Values for level_0 through level_3 must be in the range of 7 through 61 and must not duplicate any other non-communications device priority levels. Each value chosen for level_0 through level_3 must be greater or equal (have a lower or equal priority) to the preceding level. The default values for level_1 through level_3 are equal to the value assigned to the next highest priority level (level_(n-1)). For example, if a priority value has not been assigned for level_2 in a COMM directive, the priority value specified for level_1 is assumed. (See the third TTY directive in the example that follows).

Functional Description:

The COMM directive specifies from one to four priority levels at which communications lines can interrupt the central processor, as described above.

The interrupt level for a communications line is specified by the values of bits 14 and 15 of the channel number argument in the directive that identifies the line. Bits 14 and 15 correspond to the positions of the level_n arguments specified in the COMM directive.

Example:

```
                                            priority
                                            (interrupt)
                                            level
                                            -----------
     COMM  10,11
     TTY   20,20,X'FF00',....     level_0      10
     BSC   21,21,X'FC01',....     level_1      11
     BSC   22,22,X'FC81',....     level_1      11
```

In this example, three line protocol handlers are configured. The TTY associated with lrn 20 processes interrupts on level 10; the BSC station with lrn 21 processes interrupts on level 11, as does the BSC station with lrn 22.

DEVICE DIRECTIVE

Directive Name: DEVICE

The DEVICE directive is required for a communications station only if it is to be accessible through the file system interface. In this case, the DEVICE directive must be "paired" with the appropriate station-defining directive (i.e., TTY, ATD, STD, VIP, BSC, PVE, HASP, or RCI) so that each pair contains the same lrn, level, and channel number. You may float channel numbers, if you wish. The DEVICE directive should follow related LPHDEF, STATION, and LPHn directives and related POLIST, STAPOL, ROP, STDLN, and STD directives.

You must include a special format of the DEVICE directive in your CLM file if your installation will use a dual-purpose operator terminal. Refer to "Configuring a Dual-Purpose Operator Terminal" in Section V.

Format:

```
DEVICE device_unit,lrn,level,X'channel',[device_name],

        [record_size] [,{B|N}]
```

Argument Description:

device_unit

A string of up to six ASCII characters: the first three or four characters identify the type of station and the last character (alphanumeric) identifies one specific station of that type. The permissible values of device_unit are as follows:

TTYnn
BSCnn
XBSCnn
VIPnn
PVEnn
ROPnn
ATDnn
STDnn
HASPnn
RCInn

The default characteristics of various types of stations are given below. Some of the default characteristics are specified by the default value of the device specific words (dsw). Certain default

DEVICE

characteristics for any station in the configuration can be changed by use of the CLM directive STTY which is described later in this section. The description of the STTY directive includes an explanation of the device specific words.

In addition, a user can override (temporarily change) certain default characteristics of a station he is using by the system command STTY or the macro call $STTY.

TTYnn

- Record size: 73 bytes (including control byte)
- Device specific word for connect/disconnect = 0
  - No autodial used
  - Hang up phone on disconnect
  - Queue abort
- Device specific word for read/write = 0030 (hexadecimal)
  - Trailing carriage return
  - Trailig line feed
  - Echo mode
  - Leading control byte
- Detab is on
- Input is asynchronous
- Output is asynchronous
- Type is bidirectional

BSCnn

- BCS 2780 protocol
- Record size: 137 bytes (including control byte)
- Device specific word for connect/disconnect = 0
  - No autodial used
  - Hand up phone or disconnect
  - Leading control byte
  - Queue abort
  - Buffer mode is single record
- Device specific word for read/write = 0000 (hexadecimal)
  - Leading control byte
- Detab is OFF
- Input is nonbuffered synchronous

- Output is nonbuffered synchronous
- Type is input only between connects or output only between connects

XBSCnn

- BSC 3780 protocol
- Record size: 137 bytes (including control byte)
- Device specific word for connect/disconnect = 0
  - Hang up phone on disconnect
  - Queue abort
- Device specific word for read/write = 0040 (hexadecimal)
  - Leading control byte
- Detab is OFF
- Input is asynchronous
- Output is asynchronous
- Type is bidirectional

VIPnn

- Record size: 81 bytes (including control byte)
- Device specific word for connect/disconnect = 0100 (hexadecimal)
  - Logical read time-out interval is 10 minutes
  - No autodial is used
  - Home cursor on page overflow
  - Leading control byte
  - Poll interval of one second (ignored is nonpolled VIP)
  - Hang up phone on disconnect
  - Queue abort
  - Do not save function codes in read IORB (Input/Output Request Block)
- Device specific word for read/write = 0010 (hexadecimal)
  - Trailing carriage return
  - Trailing line feed
  - Input/Output is asynchronous
  - Type is bidirectional

PVEnn

- Record size: 81 bytes (including control bytes)

. Device specific word for connect/diconnect = 0

- No autodial is used
- Do not save function codes in read IORB.
- Hang up phone on disconnected
- Queue abont or disconnect

. Device specific word for read/write = 0

. Detab is ON

. Input is asynchronous

. Output is asynchronous

. Type is bidirectional

ROPnn

. Record size: 73 bytes (including control byte)

. Device specific word = 0000 (hexadecimal)

- Trailing carriage return
- Leading control byte
- Physical disconnect
- Queue abort

. Detab is ON

. Input is asynchronous

. Ouptut is asynchronous

. Type is output

ATDnn

. Record size: 73 bytes

. Device specific word for connect/disconnect = 0

- No autodial is used
- Hang up phone on disconnect
- Queue abort

. Device specific word for read/write = 30 (hexadecimal)

- Echo input character (echo mode)
- Line feed at end of message (EOM)

. Detab is ON

. Input is asynchronous

. Output is asynchronous

. Type is bidirectional

STDnn

. Record size: 81 bytes

. Device specific word for connect/disconnect = 103 (hexadecimal)

- No autodial is used
- Home cursor on page overflow
- Leading control byte
- Logical poll interval = 1 second
- No space suppress
- No roll
- Hardware function codes are specified in write requests
- No timeout on read request
- Send DLE EOT (Data Link Escape: End of Transmission) (7804) on disconnect
- Hang up phone on disconnect
- Queue abort

. Device specific word for read/write = 10 (hexadecimal)

- Carriage return at end of message
- Line feed at end of message
- Print one copy (7804)

HASPnn

. Record size: 520 bytes (including control bytes)

. Device specific word for connect and disconnect = 0000 (hexadecimal)

- Queue abort
- Hang up phone on disconnect

. Device specific word for read/write = 0100

- Leading control byte

. Detab is ON

. Input is asynchronous

- Output is asynchronous

. Type is bidirectional

RCInn

. Record size: 132 bytes (including control byte)

. Device specific word for connect and disconnect = 0003

- Do not hang up phone on disconnect

- No queue abort

. Device specific word for read/write = 0100

- Do not quit upon receiving break
- Leading control byte

. Detab is ON

. Input is synchronous

. Output is synchronous

. Type is bidirectional

lrn

The logical resource number of the station identified by the ATD, STD, TTY, VIP, BSC, PVE, HASP, or RCI directive with which this DEVICE directive is "paired". (A ROP is identified in a VIP directive). The value of lrn is an integer from 3 through 255.

level

The priority level of the station identified by the ATD, STD, TTY, VIP, BSC, PVE, HASP, or RCI directive with which this DEVICE directive is "paired".

The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

X'channel

The channel number of the station identified by the ATD, STD, TTY, VIP, BSC, PVE, HASP, or RCI directive with which this DEVICE directive is "paired". Bits 10 through 15 of the channel number specified in a DEVICE directive should equal 0 even though they may not equal 0 in the "paired" directive. Bits 15 and 16 of the channel number specified in a DEVICE directive do not indicate the interrupt level of the communications line. You can float this channel assignment by specifyig a single zero (0), if you wish.

[device_name]

A string of 1 to 12 ASCII characters, the first of which must be alphabetic. This device_name is a unique name by which the station can be referred to within the file system. If a device_name is not specified, the device_unit argument is used as the device_name.

[record_size]

The length, in bytes, of one physical record. If record_size is not specified, the default record_size is as established by the device_unit argument.

$\left[\begin{Bmatrix} B \\ N \end{Bmatrix}\right]$

For normally unbuffered stations (viz., BSC, HASP, RCI), B indicates that input/output to the station is to be buffered.

For normally buffered stations (viz., TTY, VIP, PVE, ROP), N indicates that input/output to the station is to be unbuffered.

For TTY, VIP, ROP, PVE, ATD, and STD stations, the default is buffered. For HASP, RCI, BSC, and XBSC stations, the default is unbuffered.

Input/output to a buffered station may be asynchronous or synchronous. Input/output to an unbuffered station is always synchronous; tabulation characters are not expanded.

For a BSC 3780 (XBSC) station, input/output must be asynchronous. Therefore, this argument must not be specified as N for an XBSC station.

For the following station types:

TTY
VIP
PVE
ROP
ATD
STD
HASP
RCI

tabulation characters are normally expanded into space characters. Tabulation characters are not expanded if N is specified for these station types. Tabulation characters are never expanded for BSC or XBSC station types.

Functional Description:

If a communications station is to be accessible through the file system interface, the station's TTY, VIP, ATD, STD, BSC, PVE, HASP, or RCI directive must be "paired" with a DEVICE directive. The lrn, level, and channel numbers for each pair of directives must be identical. (Note that this requirement applies to an MLCP/DLCP-connected operator terminal).

Multiple DEVICE directives that specify the same lrn and level are invalid. The EQLRN directive (described elsewhere in this section) allows you to equate multiple lrn's for a device.

Example 1:

```
TTY  21,8,X'FF80',,300

DEVICE  TTY00,21,8,X'FF80',TTYFILE
```

In this example, a TTY is to be accessible through the file system interface. The DEVICE directive contains the same lrn, level, and channel number as the TTY directive. The default characteristics of the station are shown under the device_unit argument above. The default record size (73 bytes) is also to be used since no record_size argument is specified in the DEVICE directive. The device name TTYFILE is to be used for references to the station within the file system.

Example 2:

```
SYS ........
-
-
STDLN 10,X'FF00',2,2400,W4
STD 20,0,,V7805
ROP ROSY26
STD 21,1,,V7804
STD 22,2
POLIST 2
STAPOL 0,1,0,2
DEVICE STD00,20,10,X'FF00',V7805,80,B
DEVICE STD02,21,10,X'FF00',V7804,80,B
DEVICE STD03,22,10,X'FF00',V7804,80,B
-
-
```

In this example, three synchronous devices are to be accessible through the file system interface. DEVICE STD00, DEVICE STD02, and DEVICE STD03 directives are paired with the first, second, and thrid STD directives in the CLM file. Note that each DEVICE directive has the same lrn, level, and channel number as specified in its paired STD directive. DEVICE directives may appear anywhere in your CLM file.

EQLRN DIRECTIVE

Directive Name: EQLRN

The EQLRN directive allows you to specify multiple logical resource numbers (lrn's) for the same physical device. Although each device in a communications configuration must be assigned a unique lrn, the EQLRN directive permits you to "equate" two or more lrn's to reference the same physical device.

Format:

$$\text{EQLRN} \quad lrn_p, lrn_s\ [, lrn_s, \ldots]$$

Argument Description:

$lrn_{(p)}$

The primary logical resource number associated with the physical device. This argument must be specified.

$lrn_{(s)}$

The secondary logical resource number(s) that will be associated with the device being referenced by lrn(p). You must specify one or more lrn(s) arguments. Values you specify for lrn(s) must be numerically less than the largest lrn used in a communications DEVICE directive or any other communications directive.

The follwing example illustrates one possible application of the EQLRN directive:

A communmications device can be driven with two different sets of device characteristics via the same user-written driver. Each different logical device is referenced by a unique file name. This requires two separate communications DEVICE directives, each specifying a unique file name. The device characteristics are specified by STTY directives.

The following set of CLM directives illustrate those directives which must be included at configuration time:

```
   •
   •
   •
COMM 9
LPH3 32,32,X'FF80',,4800
DEVICE TTY05,32,32,X'FF80',CDR00,80
STTY CDR00,,X'0C00',,S
DEVICE TTY06,33,32,X'FF80',CRP00,80
STTY CRP00,,X'0800',OFF
EQLRN 32,33
   •
   •
   •
ATD 35,35,X'FF00',0,9600,'7200'
DEVICE TTY07,35,35,X'FF00',,,N
   •
   •
   •
```

Note that the last two directives above specify an lrn that is numerically larger than those specified in the EQLRN directive. The lrn's specified in the EQLRN directive must be numerically less than the largest lrn specified in any other communications directive.

HASP DIRECTIVE

Directive Name: HASP

The HASP directive identifies an IBM workstation on a line serviced by the HASP line protocol handler.(1)

Format:

```
HASP lrn,level,X'channel',[modem]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications station, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

(1)
If te station is to be accessible through the file system interface, an appropriate DEVICE directive must be "paired" with the HASP directive (See DEVICE Directive," earlier in this section). If input/output to the station is to be asynchronous, the B (buffered) argument must be included in the DEVICE directive.

X'channel'

A four-digit hexadecimal number (from X'040n' to X'FF8n') specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 – The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 – Must be set to zero.

Bits 14 and 15 – Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0,1,2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 – Direct connect.

2 – Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater – User-defined modem type (see "MODEM directive", later in this section).

The default value is modem type 2.

H3270 DIRECTIVE

Directive Name: H3270

The H3270 directive identifies a station on a line serviced by the BSC 3270 line protocol handler. Only one station may be configured on a line.

Format:

```
H3270  lrn,level,X'channel',[modem],X'poll_address',
       X'select_address'
```

Argument Description:

lrn

The logical resource number associated with the station. The value of lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

X'channel

A 4-digit hexadecimal number (from X'040n' to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9	- The 10-bit channel address of the send or receive channel on the line.
Bits 10 through 13	- Must be set to zero.
Bits 14 and 15	- Specifies n, the priority level at which a communications line interrupts the central processor. n may have value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

2 - Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", later in this section).

The default value is modem type 2.

X'poll_address'

A 2-digit hexadecimal number (from X'00' to X'FF') specifying the poll address of the 3270 control unit.

X'select_address'

A 2-digit hexadecimal number (from X'00' to X'FF') specifying the select address of a 3270 device.

Example:

```
H3270 25,20,X'FC00',2,X'60',X'40'
```

In this example the host will use an address of X'60' to poll this station and an address of X'40' to select this station.

LPHn DIRECTIVE

Directive Name: LPHn

The LPHn directive identifies the first (or only) station on a line serviced by a user-written line protocol handler.(1)

Format:

```
LPHn   lrn,level,X'channel',[modem],[speed],
       [FDX/HDX][,lph_specific_word]
```

In the directive name LPHn, n is an integer from 0 through 3 and identifies a specific line protocol handler. If an LPHDEF directive is used in associated with this line protocol handler, the value of the lph argument in the LPHDEF directive must match n.(2)

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

(1)
The LPHn directive is also used in conjunction with the Remote Batch Facility. See Appendix F.

(2)
The value No. 1 is used to configure the ACTD LPH.

X'channel'

A four-digit hexadecimal number (from 'X040n' to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

1 - Bell 1xx-type modem (103A, etc.). Both data-set-ready and carrier-detect signals are needed for a connection; absence of both signals is a disconnection.

2 - Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", later in this section).

The default value is modem type 2.

[speed]

The data rate in bits per second.

For an asynchronous line with a communications-pac whose id is 2108(16), use one of the following values for speed:

	50	300	2400
	75	600	3600
(default)	110	900	4800
	134	1200	7200
	150	1800	9600

For an asynchronous line with a communications-pac whose id is 2100(16), 2110(16), or 2118(16), use one of the following values for speed:

	50	200	1800
	75	300	2000
(default)	110	600	2400
	134	1050	4800
	150	1200	9600

NOTE: If the data rate is 134.5, specify 134.

For asynchronous terminals you have the option to defer selection of the line speed until the terminal comes on-line. You select this option by specifying HI or LO.

'HI'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 1200 through 9600 bits per second.

'LO'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 110 through 1200 bits per second.

For more information about the line speed selection capability, refer to the discussion under "Topics Related to CLM Directives" near the beginning of this section.

[FDX/HDX]

Specifies whether the line is full- or half-duplex. If it is full-duplex (FDX), two channel tables will be assigned. The default value is HDX.

[lph_specific_word]

A word containing user-defined information to be passed to the line protocol handler through the station table at offset Z0SSTS. The default is zero.

Functional Description:

The LPHn directive must be included once for each line (i.e., pair of channels) of an MLCP/DLCP on which there are stations to be driven by a user-written line protocol handler. An LDBU directive (described in Section V) must be included among the CLM directives so that the Configuration Load Manager will load the user-written line protocol handler bound unit and execute its initialization code. If the sizes of the channel and station tables are different from the default sizes for these tables, an LPHDEF directive must be included before the related LPHn directive(s). The values specified in the LPHDEF directive apply only to the LPHn and STATION directives that immediately follow the LPHDEF directive in a CLM file.

If there is more than one station on a line driven by the user-written line protocol handler, the additional stations on the line must be identified by STATION directives that immediately follow the LPHn directive (See "STATION Directive," later in this section).

Example:

```
LPH0   27,8,X'FD80',,,FDX
STATION   28,1
```

In this example, there are two stations on a synchronous, full-duplex line controlled byy a user-written line protocol handler.

LPHDEF DIRECTIVE

Directive Name: LPHDEF

For each line protocol handler you write, you can include an LPHDEF directive to define the sizes of tables used for the channels and stations controlled by the line protocol handler. If the LPHDEF directive is not included, channel table and station table default sizes will be used for channels and stations controlled by the line protocol handler (see channel_table_size and station_table_size arguments below).

Format:

```
LPHDEF  lrh,[channel_table_size]  [,station_table_size]
```

Argument Description:

lph

An integer from 0 through 3 that associates this LPHDEF directive with a line protocol handler identified in one or more LPHn directives.

[channel_table_size]

Specifies the number of words needed for the channel table and the CQB's (communications queue blocks). It must have a value of at least 10 words. The default value is 33 words.

[station_table_size]

Specifies the number of words needed for this line protocol handler's station table (resource control table). It must have a value of at least 10 words. The default value is 10 words.

NOTE: The values specified for channel_table_size and station_table_size apply only to the LPHn and STATION directives that immediately follow the LPHDEF directive in a CLM file.

Example:

```
LPHDEF  0,30

LPH0  27,8,X'FD80',,,FDX

STATION  28,1
```

In this example, line FD80 has two stations on a synchronous.

full-dupplex line controlled by a user-written line protocol handler. Each of the two channel tables for the line has a size of 30 words, as defined by the channel_table_size argument in the LPHDEF directive. The default value (10 words) is accepted for station_table_size.

MODEM DIRECTIVE

Directive Name: MODEM

The MODEM directive defines a nonstandard modem type. The information provided in this directive is used to test entries in the appropriate line control table of the MLCP/DLCP to verify a connection or disconnection.

Standard modem types 3 and 4 have a connect feature for some European data sets that require data-terminal-ready to remain low until the ring indicator is turned on. This feature is activated by setting bit 7 of the data-set-control argument.

Format:

```
MODEM  type_number,connection_AND_mask,connection_XOR_mask,
       disconnection_AND_mask,disconnection_XOR_mask,
       data_set_control
```

Argument description:

type_number

An integer from 3 to 15 that is assigned to this modem definition and may then be used in a communications station directive (i.e., TTY, VIP, BSC, HASP, RCI, PVE, and LPHn directives).

connection_AND_mask

A 2-digit hexadecimal number whose value governs which bits (i.e., from 0 through 3) of line register 5 (LR5) will be examined when a connect request is processed.

connection_XOR_mask

A 2-digit hexadecimal number whose value governs which bits (from 0 through 3) of LR5 must be ON (i.e., set to 1) for a connection.

disconnection_AND_mask

A 2-digit hexadecimal number whose value governs which bits (i.e., from 0 through 3) of LR5 will be examined when a disconnect request is processed or when a test for the occurrence of a disconnect is made.

disconnection_XOR_mask

A 2-digit hexadecimal number whose value governs which bits (from 0 through 3) must be ON (i.e., set to 1) for a disconnection.

data_set_control

Specifies a 2-digits heaxdecimal number. Bits 0 through 4 are loaded unconditionally into bits 0 through of byte 20 of the appropriate line protocol table (LCT). The contents of LCT byte 20 are loaded into line register 2 of the communications-pac when a line is to be connected.

NOTES:

1. To test for a successful connection, the contents of LR5 are first subjected to a logical AND operation against the (user-supplied) connection_AND_mask; then a logical exclusive OR operation is performed on the result of the first operation, against the (user-supplied) connection_XOR_mask. If the result is zero, a connection has been established.

2. To test for a disconnect, the same operations are carried out using the analogous disconnection masks. A zero result indicates a disconnection.

3. The following shows the mask and data set control values for the standard CLM-recognized modem types:

Adapter Type	Modem Type	CONNECT AND	Mask XOR	DISCONNECTION AND	Mask XOR	Data Set Control
ASYNC	0	X'80'	X'80'	X'80'	X'00'	X'80'
SYNC	0	X'80'	X'80'	X'80'	X'00'	X'80'
ASYNC	1	X'0A'	X'A0'	X'A0'	X'00'	X'80'
SYNC	2	X'80'	X'80'	X'80'	X'00'	X'80'
ASYNC	3	X'A0'	X'A0'	X'A0'	X'00'	X'81'
SYNC	4	X'80'	X'80'	X'80'	X'00'	X'81'

Line register 5 and byte 20 of the LCT are shown below See the Communications Handbook for a detailed description of these entities.

Line Register 5:

0	1	2	3	4	5	6	7
DATA SET STATUS				COMMUNICATIONS-PAC STATUS			
DATA SET READY	RSU	CARRIER DETECTOR	RSU	RSU	RSU	RSU	RSU

LCT Byte 20/Data Set Control:

0	1	2	3	4	5	6	7
DATA SET STATUS				COMMUNICATIONS-PAC STATUS			
Data Terminal Ready	RSU	RSU	RSU	Synch-ronous Direct Connect	RSU	RSU	Monitor for Ring Indicator

Example 1:

```
MODEM  3,X'20',X'20',X'20',X'00',X'88'
```

In this example, a modem type requiring only the carrier-detect signal for a connection and absence of this signal for a disconnection is defined.

POLIST

POLIST DIRECTIVE

Directive Name: POLIST

POLIST specifies the time interval between successive scans of the poll list. The POLIST directive must precede the poll list.

Format:

```
POLIST [poll_cycle_delay]
```

Argument Description:

[poll_cycle_delay]

Defines the time interval in seconds between successive scans of the poll list. After completing a scan of the poll list the driver waits the specified time interval before re-scanning the poll list. If this parameter is specified, it must be in the range of 1 to 10 (seconds). If not specified, the default is 1 (second).

PVE DIRECTIVE

Directive Name: PVE

The PVE directive identifies a polled VIP or STD emulated station on a line serviced by the VIP or STD line protocol handler. (1,2,3,4)

Format:

```
PVE  lrn,level,X'channel',[modem],poll_address[,poll_response]
     [,controller_poll_address]
```

Argument Description:

lrn

The logical resource number associated with this station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

(1)
PVE directives are used in configuring the system at the end of the communications line where polled VIP emulation is to be performed. The system at the other end of the communications line must be configured with a VIP directive for each PVE directive: the poll address in each pair of directives must be the same.

(2)
A single communications line have up to 32 polled VIP stations. The total of 32 may be achieved by any combination of actual VIP stations or emulated stations. At your option, up to 32 stations may be combined in groups of 8 when emulating a VIP 7760 line. All stations in a group must specify the same controller poll address, as if they were connected to one VIP 7760 controller.

(3)
If the station is to be accessible through the file system interface, an appropriate DEVICE directive must be "paired" with the PVE directive (see "DEVICE Directive", earlier in this section).

(4)
A station can consist of up to 3 addressable components (scrren/keyboard, ROP, and/or cassette).

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications station, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks. The level of all PVE stations on a common MLCP/DLCP channel must be the same.

X'channel'

A four-digit hexadecimal number (from X'040n' to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 — The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 — Must be set to zero.

Bits 14 and 15 — Specifies n, the priority level at which a communication line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 — Direct connect.

2 — Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater — User-defined modem (see "MODEM Directive" earlier in this section).

The default is modem type 2.

poll_address

Specifies the poll_address of this station on the line identified by the channel argument. The poll_address argument is an integer from 0 through 31. Each station on the line must have a unique poll_address.

[poll_response]

Specifies the type of response that PVE will generate if a select and/or poll is received for a station that is not logically connected. The possible values are:

QA - If a non-connected station is selected to receive data, PVE will respond to the subsequent poll with a positive acknowledgement (ACK).

If a non-connected station is simply polled, PVE will respond with a quiescent (Q) frame.

QN - If a non-connected station is selected to receive data, PVE will respond to the subsequent poll with a negative acknowledgement (NAK).

If a non-connected station is simply polled, PVE will respond with a quiescent (Q) frame.

Default: No response to poll.

[controller_poll_address] [VIP 7760 only]

The controller poll address associated with this station on the line; it equals the poll address of the VIP 7760 controller to which this station is connected. This station emulates a VIP 7760 station having the same controller poll address. controller_poll_address must be an integer ranging from 0 through 7. Several stations on the line may have the controller poll address.

Default: No controller poll address is assigned for this station.

Example:

```
PVE  30,9,X'FD80',,0

PVE  31,9,X'FD80',,1

PVE  32,9,X'FD80',,2
```

In this example, three polled VIP emulated stations are defined for a communications line (FD80). Each station has a unique lrn and poll_address. The default value (2) is established for modem type.

The system at the other end of the communications line must be configured with VIP directives for the same physical line: each poll_address in a VIP directive (in system "A").

RCI DIRECTIVE

Directive Name: RCI

The RCI directive identifies a station on a line serviced by the Remote Computer Interface (RCI) line protocol handler.(1)

Format:

```
RCI  lrn,level,X'channel',[modem]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61: it may be the same as the level specified for other communications stations, but is must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

(1)
If the station is to be accessible through the file system interface, an appropriate DEVICE directive must be "paired" with te RCI directive (see "DEVICE Directive", earlier in this section). If input/output of the station is to be asynchronous, the B (buffered) argument must be included in the DEVICE directive.

X'channel'

A A four-digit hexadecimal number (from X'40n' to X'FF8n') specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

2 - Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", earlier in this section).

The default value is modem type 2.

ROP

ROP DIRECTIVE

Directive Name: ROP

The ROP directive specifies that a receive-only printer (ROP) is connected to a station on a line serviced by the STD line protocol handler. (See the STDLN directive for information on how to specify ROP directives in a CLM file).

Format:

```
ROP  rop_type
```

Argument Description:

rop_type

Specifies the type of receive-only printer device that is connected to the station. Possible values for this argument are:

rop type	corresponding device type
1003	PRU1003
1005	PRU1005
TN300	TermiNet 300
TB1200	TermiNet 1200

One value for rop_type must be specified.

STAPOL DIRECTIVE

Directive Name: STAPOL

The STAPOL directive defines the order in which stations are polled on a line serviced by the STD line protocol handler. The order in which stations are polled is determined by the position of arguments in the STAPOL directive. (See the STDLN directive for example).

Format:

```
STAPOL station_poll_address [,station_poll_address  ]
                                                   2
      ........[,station_poll_address  ]
                                    15
```

Argument Description:

$station\_poll\_address_n$

Specifies the poll address of a station on this line. Up to 15 station poll addresses may be specified. The station poll addresses must have been previously specified in the STD directive associated with this line. The value of this argument must be in the range 0 through 31. It may be specified as many times as is necessary to create the polling priority list for the line.

STATION DIRECTIVE

Directive Name: STATION

The STATION directive identifies the second or subsequent station(s) on a line controlled by a user-written line protocol handler that drives multiple stations per line. One station on the line must be identified by an LPHn directive; additional stations are identified by STATION directives, one per station, immediately following the related LPHn directive (1). (See the LPHn directive for an example). STATION directives and LPHn directives must immediately follow the LPHDEF directive that defines them in a CLM file.

Format:

```
STATION  lrn[,lph_specific_word]
```

Argument Description:

lrn

The logical resource number associated with the station. The station for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

[lph_specific_word]

Specifies a word containing user-defined information that is to be passed to the line protocol handler through the station table, at offset ZQSSTS. The default is 0.

NOTE: The priority level, channel number, modem type, line speed, and line procedure (FDX/HDX) of stations described in STATION directives are obtained from the LPHn directive that precedes the STATION directive (See "LPHn Directive", earlier in this section).

1
For additional (polled) VIP's on a line, use additional VIP directives with the same level and channel number, rather than STATION directives.

STD DIRECTIVE (1)

Directive Name: STD

The STD directive identifies a station on a line serviced by the synchronous terminal device (STD) line protocol handler. See the STDLN directive for examples.

Format:

```
STD  lrn,station_poll_address[,,device_type]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

station_poll_address

Specifies the poll address of this station. The poll address is an integer from 0 through 31. Each station on the line must have a unique station_poll_address.

[device_type]

Specifies the type of VIP terminal used. Possible values for device_type are:

(1)
The STD directive and the VIP directive support synchronous terminals. The line protocol handler called by the STD directive offers additional support capabilities (e.g., support of VIP7804 and VIP7805 terminals) not offered by the line protocol handler called by the VIP directive.

device_type	corresponding terminal type
V7804	VIP 7804/VIP 7805
V7700	VIP 7700/VIP 7700R/VIP 7705R
V7760	VIP 7760, DKU 7005, DKU 7007

If this argument is not specified, the default device_type is V7700.

When using the Synchronous Terminal Driver (STD) line protocol handler, you must observe the following:

- The master LRN station must be the last station to be disconnected.
- The 7804 Control Byte Head of Form support translates to a 7804 clear escape sequence. This sequence, in addition to positioning the cursor to the Home row, puts the 7804 in text mode.
- To use the STD LPH, the VIP 7804 hardware switches must be configured as follows:

  Reset, Text Mode, Verify before process,
  Transmit next block.
  Optional settings are:
    Roll/No Roll
    Space/No Space
    Return = Transmit/Normal
    Set Block Size

- VIP 7804 terminals are not compatible with VIP7700 terminals in the handling of end-of-message CR's and LF's. It is recommended that the new STD line protocol handler be used for VIP7804 support. If the VIP line protocol handler is used to support a VIP7804 terminal, the following must be observed:

  1. RETURN=NORMAL must be selected at the terminal.
  2. The XMT key should be used to enter commands.
  3. When using the Editor, Linker, or other utility directives with the prompt option, the RETURN key should be depressed before the SMT key. This will assure that the promt appears on the next line rather than at the and of the current line.

STDLN DIRECTIVE

Directive Name: STDLN

The TSDLN directive identifies a line serviced by the STD line protocol handler. (The STDLN directive does not support floating channel number assignments. You must explicitly specify the channel number argument).

Format:

```
STDLN   level,X'channel',[modem],[speed][,2/4 wire]
```

Argument Description:

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value of the level is an integer from 8 through 58; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for non-communications devices or tasks.

X'channel

A four-digit hexadecimal number (from X'40n' to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 through 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct Connect

2 - Bell 2xx-type modem (201A, 208B, ect.). The data-set-ready signal is needed for a connection: absence of this signal is a disconnection.

3 or greater - User-defined modem (See "MODEM Directive", in this section).

The default is modem type 2.

[speed]

The data rate in bits per second. The default value is 2000. Other possible values for speed include.

2400
4800
9600
19200

[2/4 wire]

Specifies a 2 or 4 wire connection. Possible values are:

	Value	meaning
(Default)	W2	2-wire connection
	W4	4-wire connection

Functional Description:

The STDLN directive defines a line serviced by the STD line protocol handler: it precedes all other directives which characterize this line. Configuring a line serviced by the STD line protocol handler requires that at least four different directives be specified for the line and all stations on the line. These directives must be specified according to the following guidelines:

Required directives

STDLN (specify one for each line)
STD (specify one for each station on the line)
POLIST (specify one for each line)
STAPOL (specify one or more as needed)

Optional directive

ROP

Sequence rules

The following diagram illustrates the sequence rules that apply when specifying STDLN and other STD directives. One set of these directives must be specified for each line. Note that all directives which characterize the line must immediately follow the STDLN directive that identifies the line. Brackets enclose optional directives.

```
STDLN      must specify one for each line

STD
   1                                        specify one set for each
[ROP] specify one for each                  station on the line; all must
      station, if needed                    precede POLIST

STD
   2
[ROP]
  .
  .
  .
POLIST     must specify one for each line

STAPOL     must specify one or more
           for each line; all STAPOL
           directives should be specified
           in the order in which stations
           are to be polled
```

Example 1:

```
    .
    .
    .
    STDLN 20,X'FC00',,2400,W4
    STD 20,1,,V7804
    ROP  TN300
    STD 21,2,,V7804
    ROP ROSY26
    STD 22,3,,V7804
    POLIST 5
    STAPOL 1,2,1,3
    .
    .
    .
```

In this example, an STD line has been configured with three stations. As specified in the STDLN directive, this line's request level is 20, the channel number is FC00, the default modem used is type 2, the line speed is 2400 bits per second, and the line connection is 4-wire. All three stations support VIP7804 terminals.

The first station has an lrn of 20, a poll address of 1, and includes a TN300 ROP. The second station has an lrn of 21, a poll address of 2 and includes a PRU 1005 ROP. The third station has an lrn of 22, a poll address of 3 and no ROP. The POLIST directive identifies the start of the poll list and specifies a 5-second delay between that the station with poll address 1 (lrn 20) is polled twice as often as the other two stations, implying that this station has higher priority.

Example 2:

```
SYS ........
-
-
-
-
STDLN 10,X'FF00',2,2400,W4
STD 20,0,V7805
ROP ROSY26
STD 21,1,V7804
STD 22,2
POLIST 2
STAPOL 0,1,0,2
DEVICE STD00,20,10,X'FF00',V7805,80,B
DEVICE STD02,21,10,X'FF00',V7804,80,B
DEVICE STD03,22,10,X'FF00',V7700,80,B
-
-
-
```

The above example describes an STD line connected to the system through address "FF00", with an interrupt level of 10. The line uses a type "2" modem at 2400 baud and is a 4-wire connection. There are three VIP's on the line. The terminals will be polled in the following sequence for data: 0,1,0,2: after which there will be a delay of two seconds before they are polled for data again. Address "0" well be polled twice as frequently as the other two terminals. All of the devices are configured for the file system, as shown in the DEVICE directives. (The DEVICE directives may appear anywehere in your CLM file).

STTY DIRECTIVE

Directive Name: STTY

The STTY directive specifies the file characteristics of a device that is not a disk device or a private device. The characteritics specified by this directive override the default characteritics established by the associated DEVICE directive (or the characteristics established by a previous STTY directive). The STTY command keyword format (as described in Appendix D) may be used as an alternative to the STTY directive format in your CLM file.

Format:

```
STTY  { device_unit  }   [length],[D'dsw  dsw  '],[detab],[in],
      { device_name, }              1    2
                         [out] [,type]
```

Argument Description:

{device_unit}
{device_name}

This entry, which identifies the device, must be the same as one entered in a previous DEVICE directive. If device_name was specified in a previous DEVICE directive, it must be entered here: i.e., the entry of device_unit will result in error. The device_unit and device_name arguments each consists of a string of ASCII characters as described under the DEVICE directive. If there is no previous DEVICE directive having the string specified here, an error message is generated.

[length]

An integer giving the line length (record size) in bytes.

(1)
You must use the STTY command format in your CLM directive file if you wish to configure a device that can be automatically reconnected if a power resumption or line drop condition occurs. See Appendix D for information on how to configure devices as reconnectable. The STTY command format is described in Appendix D.

[D'dsw_1 dsw_2']

Specifies the terminal's device specific words for connect/disconnect (dsw1) and read/write (dsw2). The values of dsw1 and dsw2 are expressed in double-word format as a pair of 4-character hexadecimal numbers enclosed in apostrophes.

dsw1 specifies certain characteristics of a device at open (connect) or close (disconnect) time, as shown in Table 6-4 and 6-6. Refer to these table when defining the 16-bit settings for dsw1 that characterize connect/disconnect functions.

dsw2 specifies certain characteristics of a device when reading or writing to a file, as shown in Table 6-5 and 6-6. Refer to these tables when defining the 16-bit settings for dsw2 that characterize read/write functions.

If dsw1 and dsw2 are not specified, certain default values are assumed. Default values are listed in Table 6-7.

NOTE: The STTY command keyword format (as described in Appendix D) may replace the STTY directive format in your CLM file. When used at CLM time, either STTY format will cause the initial dsw1 and dsw2 assignments (as well as the current device specific word assignments) to be modified. However, if the STTY command is specified at any time after system initialization, only the current device specific word assignments will be modified. Only at system building (CLM) time will the STTY directive format or the STTY command keyword format modify both the current and the initial device specific word assignments.(1)

[detab]

An alphabetic string that controls tabulation

-ON Tab characters in the input stream are deleted and replaced by the number of spaces required to bring the cursor (or printer) to the next tab position. Tab positions are set by the MOD 400 system in increments of 10 (to print positions 11, 21, 31,...) and cannot be changed by the user.

-OFF Tab characters in the input stream are not replaced by spaces.

(1)
In a diskette-based system, using the STTY command format in your CLM file results in the loader error 1609. If you wish to use the STTY command format in your CLM file, you must create a directory under the system root named SYSLIB2 and load in it the bound unit SSTY.

[in]

An alphabetic character specifying the type of input accepted by the device.

A - The device is to receive asynchronous input

S - The device is to receive synchronous input

N - The device is to receive nonbuffered synchronous input

[out]

An alphabetic character specifying the type of data transmitted by the device

A - The device transmits asynchronous output

S - The device transmits synchronous output

N - The device transmits nonbuffered synchronous output

[type]

An alphabetic character specifying the device type

I - The device is an input device

O - The device is an output device

B - The device is bidirectional: it receives and transmits

Messages

Error messages that may be issued during execution of this directive are as follows:

133E	CMD	(STTY)	ERROR.	No previous DEVICE directive.
133F	CMD	(STTY)	ERROR.	Invalid detab argument.
1341	CMD	(STTY)	ERROR.	Invalid in or out argument.
1342	CMD	(STTY)	ERROR.	Invalid type argument.

Device Specific Words for Communications Devices

Table 6-4 list the bit settings for dsw1, the device specific word for connect/disconnect functions. Table 6-5 lists the bit settings for dsw2, the device specific word for read/write functions. Use these tables in conjunction with Table 6-6 to determine the definitions of each bit.

If you do not wish to specify dsw1 and/or dsw2 in the STTY directive, the system assumes certain default values for them. Refer to Table 6-7 for a list of device specific word defaults.

Table 6-4. Values for dsw1

Function	LPH	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
CONNECT	TTY	0	0	D	0	X	0	0	0	0	0	0	0	0	Z	0	0
	VIP	R	R	D	W	X	P	P	P	F	0	0	0	0	0	0	0
	BSC	0	0	D	0	X	s	0	q	B	G	0	0	0	0	0	0
	PVE	0	0	D	0	0	0	0	0	F	0	v	0	0	0	w	x
	ATD	a	0	D	0	0	0	b	0	c	d	u	0	0	0	0	0
	STD	m	0	D	N	I	P	P	P	J	K	N	F	e	y	0	0

Function	LPH	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
DISCONNECT	TTY	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H
	VIP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H
	BSC	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H
	PVE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H
	ATD	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H
	STD	0	0	0	0	0	0	0	0	0	0	0	0	0	0	A	H

Table 6-5. Values for dsw2

Function	LPH	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
READ	TTY	0	0	0	0	0	M	0	0	Y	0	E	L	C	0	0	0
	VIP	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	BSC	0	0	0	0	0	0	0	0	0	G	0	0	0	0	0	0
	PVE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	ATD	f	0	0	0	0	0	0	Q	0	0	E	L	C	0	0	0
	STD	g	0	0	0	X	0	0	0	0	0	0	0	0	0	0	0

Function	LPH	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
WRITE	TTY	0	0	0	0	0	0	0	Q	0	0	0	L	C	0	0	0
	VIP	0	0	0	0	0	0	0	0	0	0	0	L	C	0	0	0
	BSC	0	0	0	0	0	0	0	0	0	G	S	V	M	T	0	0
	PVE	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	ATD	h	0	0	i	X	0	j	Q	0	0	0	L	C	0	0	0
	STD	g	0	0	0	X	0	0	0	0	0	0	L	C	k	k	k

Table 6-6. Bit Definitions

BIT	If bit = 0:	If bit = 1:
A	Dequeue remaining IORB's for channel	Don't dequeue remaining IORB's for channel
a	Block mode not supported	ATD Block mode supported
B	Buffer mode is single-block	Buffer mode is double-block. (Note: Double-block is not supported by file manager)
b	ETX mode (ATD Block mode only)	ETB/ETX specified on write (ATD Block mode)
C	Send CR after text of message (Not supported for supervisory message reads by ATD)	Don't send CR after text of message
c	No space suppress (VIP 7801)	Space suppress (VIP 7801) (Only applicable in block mode)
D	No Auto-dial used	Use Auto-dial handler to dial connection
d	Roll (VIP 7801)	No roll (VIP 7801)
E	Don't echo input characters	Echo input characters
e	No time-out for read	Time-out for read (immediate)
F	Don't save in read IORB function codes received in text header	Save up to 2 function codes received in text header
f	Don't abort read IORB requests (ATD)	Abort read IORB requests (ATD)
G	Use BSC2780 protocol	Use 3780 protocol
g	Don't abort read/write requests	Abort read/write requests
H	Hang up the phone on disconnect	Don't hang up the phone on disconnect (ATD Block mode)
h	Don't abort write IORB requests (ATD)	Abort write IORB requests (ATD)
I	Control word present for read/write	No control word for read/write

Table 6-6 (Cont). Bit Definitions

BIT	If bit = 0:	If bit = 1:
I	Don't support preemptive write	Support preemptive write (ATD Block mode only)
J	Don't set VIP 7804 CRT space suppress	Set VIP 7804 CRT space suppress
j	Send a block ended by ETX	Send a block ended by ETB
K	Set VIP 7804 CRT roll mode	Don't set VIP 7804 CRT roll mode
L	Don't send LF after text of message (Not supported for supervisory message reads by ATD)	Send LF after text of message
M	Data mode is non-transparent	Data mode is transparent
m	Don't set BIP 7804 CRT to block mode (RECV)	Set VIP 7804 CRT to block mode (RECV)
N	Set transparent printer mode	Set non-transparent printer mode
n	Not reserved for cross-compatibility with STD lph	Reserved for cross-compatibility with STD lph
Q	Quit on receiving break (Not supported for supervisory message writes by ATD)	Don't quit on receiving break
q	Buffer mode is not multi-block	Buffer mode is multi-block
S	Don't send EOT characters	Send EOT characters
s	Don't support DLE EOT	Support DLE EOT
T	Send ITB and ETB characters	Send ETX characters
t	Include control byte	Do not include control byte
U	Don't send DLE EOT for VIP 7804	Send DLE EOT for VIP 7804
u	No echoplex (VIP 7801)	Echoplex (VIP 7801)
V	Don't send RVI characters	Send RVI characters

Table 6-6 (Cont). Bit Definitions

Bit	If bit = 0:	If bit = 1:
v	Include received DEL characters in buffer	Strip received DEL characters
W	Home cursor on page overflow	Don't home cursor on page overflow
X	Include control byte as first byte	No control byte (first byte is data)
Y	Include ATTENTION character	Exclude ATTENTION character
y	Return key equal transmit key	No specific meaning to return key
Z	Transfer mode is character	Transfer mode is page

k VIP 7804/5 Printer copy

000 = 1 copy	010 = 3 copies	100 = 5 copies	110 = 7 copies
001 = 2 copies	011 = 4 copies	101 = 6 copies	111 = 8 copies

P Logical Poll interval

000 = 0 sec	010 = 2 sec	100 = 4 sec	110 = 15 sec
001 = 1 sec	011 = 3 sec	101 = 5 sec	111 = 30 sec

R Logical read time-out interval (poll duration):

00 = 10 min	01 = no time-out	10 = 0 min	11 = reserved for future use

w,x LPH response to application when LPH receives data but no real IORB available. VIP status codes:

00 = Send NAK	10 = Return busy status
01 = Send ACK	11 = Send NAK (same as 00)

NOTES:

1. 0 indicates that you cannot use the bit.

2. This table is organized by function for convenience only; there is no overlapping of bits. dsw1 sets all bits for connect/disconnect functions. dsw2 sets all bits for read/write functions.

Table 6-7. System Defaults for dsw1 and dsw2

Device	dsw$_1$	dsw$_2$
TTY	0000	0030
VIP	0000	0110
ROP	0000	0000
BSC	0000	0000
PVE	0000	0000
XBSC	0000	0040
ATD	0000	0030
STD	0103	0010

TTY DIRECTIVE (1)

Directive Name: TTY

The TTY directive identifies a station on a line serviced by the TTY (teleprinter) line protocol handler. The TTY directive supports asynchronous terminal devices as teleprinter-compatible devices. You may float the channel assignment in this directive, if you wish.

Format:

```
TTY  lrn,level,X'channel',[modem],[speed][,'device-type']
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

(1)
The TTY directive and the ATD directive support asynchronous terminal devices. The line protocol handler called by the TTY directive supports the physical terminal only in teleprinter mode. The TTY line protocol handler does not provide block mode support of VIP7800 devices; either does it provide support for VIP 7200 devices and VIP 7800 devices in forms processing mode. (For these functions the ATD directive is required). The ATD line protocol handler does not provide the transparent read I/O or single character mode functions of the TTY line protocol handler. The number of words of resident memory required by the TTY line protocol handler is less than the number of words of resident memory required by the ATD line protocol handler. See the MOD 400 Executive Software Release Bulletin for comparative figures.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value of level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

X'channel'

You may specify a single zero(0) to float the channel number assignment or a four-digit hexadecimal number (from X'040n' to X'FF8n'), that specifies the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bit 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

1 - Bell 1xx-type modem (103A, etc.). Both data-set-ready and carrier-detect signals are required for a connection; absence of these signals is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", earlier in this section).

The default value is modem type 1.

[speed]

The data rate is bits per second.

For an asynchronous line with a communications-pac whose id is 2108(16), use one of the following values for speed:

	50	300	2400
	75	600	3600
(default)	110	900	4800
	134	1200	7200
	150	1800	9600

For an asynchronous line with a communications-pac whose id is 2100(16), 2110(16), or 2118(16), use one of the following values for speed:

	50	200	1800
	75	300	2000
(default)	110	600	2400
	134	1050	4800
	150	1200	9600

NOTE: If the data rate is 134.5, specify 134.

You have the option to defer selection of the line speed until the terminal comes online. You select this option by specifying HI or LO.

'HI'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 1200 through 9600 bits per second.

'LO'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 110 through 1200 bits per second.

For more information about the line speed selection capability, refer to the discussion under "Topics Related to CLM Directives" near the beginning of this section.

['device-type']

Specifies the type of terminal used. If this argument is not specified, the default is TTY. Possible values are:

Value	Physical devices supported
7200	VIP 7200, VIP 7205
7801	VIP 7801, VIP 7802
PRU	PRU 1001, PRU 1003, PRU 1005, TWU 1001, TWU 1003, TWU 1005
TTY	An asynchronous terminal that is to be supported as teleprinter-compatible.

Example 1:

```
TTY    21,8,X'FF80'

DEVICE    TTY01,21,8,X'FF80',TTY1
```

In this example, the TTY is connected by a Bell 1xx-type modem and operates at 110 bits per second. Default values for modem type and line speed have been used. The TTY is to be accessible through the file system interface (by virtue of the DEVICE directive).

Example 2:

```
TTY    22,8,X'FF00',0,1200

DEVICE    TTY02,22,8,X'FF00',TTY2
```

In this example, the TTY is connected by a direct cable connection and operates at 1200 bits per second. These characteristics are reflected by explicit arguments in the TTY directive. Again, the TTY to be accessible through the file system interface.

VDAM DIRECTIVE

Directive Name: VDAM

The VDAM directive must be specified if the Display Formatting and Control software is to be used. This directive causes the software component to be incorporated in the configuration(1). You must configure the Display Formatting and Control Sofware if your installation will support DEF-II or DEF-II and TCLF. See Appendix P for further information on configuring the Display Formatting and Control Software.

Format:

    VDAM   [lrn,level][,maximum terminals]

Argument Description:

lrn

The logical resource number associated with the VDAM task. The value for lrn is an integer from 3 through 255. This value must be specified if asynchronous VDAM processing is desired. The next two sequential lrns will be reserved for VDAM processing.

level

The priority level at which VDAM processing operates. The value for level must be an integer from 7 through 61. This value must be specified if asynchronous VDAM processing is desired.

NOTE: Both lrn and level must be specified, or neither argument should be specified. If neither one is specified, the default is that not asynchronous instructions are issued to VDAM.

[maximum terminal]

The maximum number of terminals that use VDAM processing in one task group. If a number is not specified, the default value is 10.

NOTE: When using VDAM with Block Mode Terminals i.e. DKU 7005 or DKU 7007 (configured using an STD directive) another bound unit must be loaded via:

    LDBU ZNV77F

1
Display processing requires VIP 7200/7205 or VIP 7801/7802 terminals. The terminals can be configured using an ATD or a V7200 directive.

VIP DIRECTIVE

Directive Name: VIP

The VIP directive identifies a polled or nonpolled visual information projection station on a line serviced by the VIP line protocol handler. Optionally, the VIP directive identifies a ROP (receive-only printer) station on the same line. (1,2,3,4) VIP 7700, VIP 7700R/7705R, VIP 7804/7805 (in VIP7700 mode) and VIP 7760 terminals are supported. You may float channel assignments in this directive, if you wish.

Format:

$$\text{VIP lrn,level,X'channel',[modem],[poll\_address],}\ \left[\begin{Bmatrix}C\\T\end{Bmatrix}\right].$$

```
VIP lrn,level,X'channel',[modem],[poll_address],
    [ROP_lrn],[ROP_type] [,ROP_form_feed]
```

(1)
The VIP directive and the STD directive both support synchronous terminals. The line protocol handler called by the STD directive offers additional support capabilities (e.g., support of VIP7804 and VIP7805 terminals) not offered by the line protocol handler called by the VIP directive.

(2)
If the Configuration Load Manager detects an error in a VIP directive, the error must be connected in the CLM-USER file and the system bootstrapped again.

(3)
If the station is to be accessible through the file system interface, an appropriate DEVICE directive must be "paired" with the VIP directive (see "DEVICE Directive", earlier in this section).

(4)
If a "remote" system is to perform polled VIP emulation, see "PVE Directive", earlier in this section, for a description of the relationship between VIP directives for the "local" system and PVE directives for the "remote" system.

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

The level for all polled VIP's on a common MLCP channel must be the same.

X'channel

You may specify a single zero (0) to float this channel assignment or a four-digit hexadecimal number (from X'040n' to X'FF8n'), that specifies the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

2 - Bell 2xx-type modem (201A, 208B, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", earlier in this section).

The default value is mode type 2.

[poll_address]

Specifies the address of the VIP station on the line specified by the channel argument. The value for poll_address is an integer from 0 through 31. If no polling address is specified, no polling can occur on the line: in this case, only one VIP station can be on the line.

$\left[\begin{Bmatrix} C \\ T \end{Bmatrix}\right]$

This argument is meaningful in only two situations:
(1) a file-transmission (nonpolled) environment and
(2) a nonpolled environment wherein the VIP is to be accessible through the file system interface. (In the second situation, the VIP directive must be "paired" with an appropriate DEVICE directive).

C specifies that the central processor is the control station: T specifies that the central processor is the tributary station. The tributary station must send the first Q (quiescent) frame on the line. The default value is C.

(In the nonpolled, file system access environment, T should be specified in the VIP directive in order to avoid excessive delays in input/output operations).

[ROP_lrn]

Specified the logical resource number of a receive-only printer connected to the VIP controller. The value for ROP_lrn is an integer from 3 through 252. The default is that no receive-only printer is connected to the VIP controller: in this case, the ROP_type and ROP_form_feed arguments must not be specified.

[ROP_type]

Specifies the type of receive-only printer. Choose from one of the following:

ROP_type	Corresponding device_type
TN300 (default)	TermiNet 300 or PRU1003
TN1200	TermiNet 1200 or PRU1005
TTY33	ASR-33
TTY35	ASR-35

[ROP_form_feed]

Specifies whether the receive-only printer has a form-feed option. FORM or FO indicates that the receive_only_printer does have the form_feed option; NOFORM or NO indicates that it does not have the form_feed option.

The default is that TermiNets have the form_feed option and the TTY's do not have the form_feed option.

NOTE: The following rules only to lines that have more than one VIP station (i.e., a polled VIP environment):
(1) a separate VIP directive is required for each station on the line,
(2) the VIP directives be consecutive, and
(3) the logical resource numbers of the stations on the line must be consecutive.

Examples:

In the following examples, VIP stations are connected by Bell 2xx-type modems. Default values for modem type have been used. DEVICE directives are "paired" with the VIP directives because the VIP's are to be accessible through the file system interface.

Example 1:

```
VIP    23,8,X'FE80',,0

DEVICE    VIP00,23,8,X'FE80',VIP0
```

In this example, the VIP has a poll_address of 0.

Example 2:

```
VIP  24,8,X'FE80',,1,,25,TN1200

DEVICE    VIP01,24,,8,X'FE80',VIP1

DEVICE    ROP01,25,8,X'FE80',ROP1
```

In this example, the VIP has a poll_address of 1. In addition to the VIP screen, the VIP controller has a TermiNet 1200 as a receive-only printer. The VIP screen's logical resource number is 24 and the receive-only printer's logical resource number is 25.

VROSY, VTTY, AND V7200 DIRECTIVES (1), (2)

Directive Name: VROSY, VTTY, or V7200

These diretives define devices that are on lines serviced by the ATD line protocol handler. Except for the directive name, the format is the same for all three directives. You can float channel assignments in these directives.

Format:

```
( VROSY )
{ VTTY  }   lrn,level,X'channel',[modem],[speed],[device_type],
( V7200 )                        [del],[stop_bit] [,parity]
```

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 3 through 255. A program may use this number to identify the station when it requests an input/output operation to the station.

(1)
The ATD directive intended as a remplacement for the VROSY, VTTY, and V7200 directives. The line protocol handler called by the ATD directie offers additional support capabilities (e.g., block mode support of VIP7801/7802 devices, support of the VIP7207 device) not V7200 directives. The VROSY, VTTY, and V7200 directives can still be used. The directives, however, call the ATD line protocol handler.

(2)
The ATD directive or the 7200 directive must be specified if VIP7200/7205 or VIP7801/7802 terminals are to be used for forms processing using HONEYWELL's Display Formatting and Control Facility. The VIP 7200 line protocol handler "nonforms mode" is not supported by ATD. The equivalent functions can be obtained by using the forms mode with a form defined as fully unprotected.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 61; it may be same as the level specified for other communications stations, but it must be a <u>higher number than</u> the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may <u>not</u> also be used for noncommunications devices or tasks.

X'channel'

You may specify a single zero (0) to float this channel assignment or a four-digit hexadecimal number (from X'040n' to X'FF8n'), that specifies the channel number of the station and the inetrrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 0, 1, 2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

1 - Bell 1xx-type modem (103A, etc.). Both data-set-ready and carrier-detect signals are required for a connection; absence of these signals is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", earlier in this section).

The default value is modem type 1.

[speed]

The data rate in bits per second.

For an asynchronous line with a communications-pac whose id is 2108(16), use one of the following values for speed:

	50	300	2400
	75	600	3600
(default)	110	900	4800
	134	1200	7200
	150	1800	9600

For an asynhronous line with a communications-pac whose id is 2100(16), 2110(16), or 2118(16), use one of the following values for speed:

	50	200	1800
	75	300	2000
(default)	110	600	2400
	134	1050	4800
	150	1200	9600

In VTTY or V7200 directives, you have the option to defer selection of the line speed until the terminal comes on-line. You select this option by specifying HI or LO.

'HI'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 1200 through 9600 bits per second.

'LO'

Specifies that all terminals associated with this line will be permitted to function only at speeds from 110 through 1200 bits per second.

For more information about the line speed selection capability, refer to the discussion under "Topics Related to CLM Directives" near the beginning of this section.

[device_type]

Specifies the type of terminal used. The allowable types and the default for each directive are as follows:

Directive	Devices supported	device_type
V7200	VIP 7200, VIP 7205	7200
	VIP 7801, VIP 7802	7801
VROSY	PRU 1001, PRU 1003, PRU 1005, TWU 1001, TWU 1003, TWU 1005	PRU
VTTY	An asynchronous terminal that is to be supported as teleprinter-compatible	TTY

[del]

In the TTY mode, a head-of-form sequence (pre-order control) consisting of three LF's and the number of DEL characters specified by this argument is generated if the following conditions are met. (If the conditions are not met, no head-of-form sequence is generated).

- The first byte of the application's buffer must be designated as a control byte; i.e., bit 4 of the IORB's device specific word (I_DVS) must be set to 0 at write time.
- The control byte must specify that a head-of-form sequence is to be generated; i.e., bit 3 of the control byte must be set to 1.
- Device type PRU is not specified. If it is specified, a true head-of-form is issued.

In the field or TTY modes, an LF in an end-of-message sequence (post order control) is followed by the number of DEL characters specified by this control argument.

The EOM sequence is controlled by the B- and C-bits of the IORB's device specific word I_DVS, as specified by the application at write time. The TTY line protocol handler sends an EOM sequence according to the following B- and C-bit values.

I_DVS Bits		
B	C	EOM Sequence
0	0	CR
0	1	None
1	0	CR,LF,DEL characters
1	1	LF,DEL characters

At read time, the application can specify the same B- and C- bit values in order to send an EOM sequence back to the terminal when the message is successfully received.

Note that an LF character in a pre-order control specified by the I-CON word of an IORB is never followed by a DEL character; i.e., this control argument has no effect in this case.

This argument allows you to select 1 to 32 DEL characters. The default for each type of device is as follows:

device type	Number of DEL characters
7200	0
7801	0
PRU	1
TTY	1

[stop_bit]

Specifies the number of stop bits that are to follow each character. A value of 1 or 2 can be chosen for each device. Default values are as follows:

Number of Stop Bits
1 (for speeds greater than 110 bits per second)
2 (For a speed of 110 bits per second or less)

[parity]

Specifies the type of parity (enter ODD or EVEN) to be used. The default is EVEN.

SECTION VII

MOD 400/MFS PROGRAM MATERIALS AND DISTRIBUTION MEDIA

A listing of MOD 400/MFS program materials is provided in Figures 7-1 through 7-19. Figure 7-1 lists the program materials made available with the MOD 400/MFS Executive. Figures 7-2 through 7-19 list the program materials made available with the various separately-priced products that can be used with the MOD 400/MFS Executive. Order numbers for these products are also provided.

MOD 400/MFS software is available on the following media:

- Cartridge Disk
- Cartridge Module Disk
- Mass Storage Unit
- Diskettes

MOD 400/MFS Executive software can be distributed on a Mini Cartridge disk (^ZSYS62), two cartridge disks (^ZSYS51 and ^ZSYS52), a single cartridge module disk (^ZSYS61), or on a single mass storage unit (^ZSYS71). If an installation orders any of the separately-priced products, appropriate modules are added to ^ZSYS51 and ^ZSYS52, ^ZSYS61, ^ZSYS62, or ^ZSYS71, or they are provided on separate diskettes.

MOD 400/MFS Executive software is made available on up to five diskettes (^ZSYS00, ^ZSYS0A, ^ZSYS1A, ^ZSYS1B). The software will be packaged to make most effective use of the distribution media. Software will be packaged up on to five diskettes if the user receives the software on single-sided diskettes. If the user submits diskettes with a larger capacity, then fewer diskettes are required.

Users performing system installation with CII HONEYWELL BULL-supplied distribution media should note the following:

- If the MOD 400/MFS Executive was distributed on a Mini Cartridge disk, cartridge disks, a cartridge module disk, or a mass storage unit, then all Executive modules will be present on the volume(s).

- If the MOD 400/MFS Executive was distributed on multiple diskettes, then each diskette will contain a subset of the total number of Executive modules. The diskette labeled ^ZSYS00 must be used as the bootstrap volume at initial system startup. The user must be aware, however, that ^ZSYS00 weill not contain sufficient software to permit use of the full range of MOD 400/MFS functionality. Once the user attains the limited processing environment permitted with ^ZSYS00, then the user should examine the contents of the other diskettes and delete and transfer software modules as desired. The user can consult Figure 7-1 to identify Executive modules required for various MOD 400/MFS functions.

In Figures 7-1 through 7-19, elements enclosed within boxes are file system directories. Other elements are files.(1) Indentation signifies subordination - that is, all files described in a given directory are indented from the margin of the boxed directory name. In a few instances, one directory is shown to be subordinate to another directory; again, indentation is used to signify this relationship.

SOFTWARE TO BE PLACED ON THE BOOTSTRAP VOLUME

Certain modules must be present on the bootstrap volume in order to achieve system startup. Other modules must be present on the bootstrap volume if certain types of processing are to be supported. These required modules are described in the following diagrams.

(1)
Note that in some cases files are immediately subordinate to the volume root directory.

Z3EXECUTIVES[a]	SAF system executive; required
Z3EXECUTIVEL[b]	LAF system executive; required
START_UP.EC	File used with CLM_USER; required
GROUP$H.EC	EC file to spawn $H task group
GROUP$P.EC	EC file to spawn $P task group
SUPER.EC	EC file required for a dual-purpose operator terminal
GROUP$D.EC	EC file to spawn $D task group
DEBUG.WORK	Work file for $D DEBUG
DEBUGDB	Lead task of $D DEBUG utility
RECOVERY	File recovery
FILES	
GROUPS	
HIS	
START_UP.EC	File used for $H task group; required if CLM_MDC or CLM_MCP used
EML	
EMLFILE	Error message library
MDD	
$PR.Q2	mailbox queue
$PR.Q3	mailbox queue
UDD	
REPORTS	
LDD	
OBJECT	
EXECUTIVE	
ZIMLCC.O	
ZGQISB.O	

Figure 7-1. MOD 400 Executive Program Materials

MACRO

INCLUDE

CCP_MAC.IN.A
MGIRB.IN.C
MGRRB.C
MGCRB.C

SIJ

START_UP.EC	File used with CLM_MDC or CLM_MCP; required onlu if either is present
CLM_SAMPLE	Sample CLM file; optional
CIPSIM	Commercial simulator
CLM	Configuration Load Manager; required
CLMCM1	CLM functions; required
CLMCM2	CLM functions; required
CLMCM3	CLM functions; required
CLMCM4	CLM functions; required
CLMCOM	CLM functions; required
CLMDEV	CLM functions; required
CLMFLT	CLM functions; required
CLMMMJ	CLM functions; required
CLMLHD	CLM functions; required
CLMMAP	CLM functions; required
CLMST1	CLM tables; required
CLMST2	CLM tables; required
CLMST3	CLM tables; required
CLMVAR	CLM functions; required
RESOLA	Makes all Executive overlays resident; optional c

Figure 7-1 (Cont). MOD 400 Executive Program Materials

ROLLCD	CLM code for rollout function; required only if rollout capability is configured
SIPSIM	Single/Double Precision SIP Simulator c
SIPSIM_SP	Single Precision SIP Simulator c
VLOCD	CLM code for MMU functions; required only if MMU present
ZERRST	Error statistics module; optional c
ZQQCDS	CLM communications module c
ZQEACU	Auto Call module; optional
ZQESEL	Speed select module; optional c
ZQEXEC	Communications supervisor c
ZQHLPH	Communications HASP line protocol handler; optional c
ZQPATD	Communications asynchronous terminal line protocol handler; optional c
ZQPCRT	Communications synchronous terminal line protocol handler; optional c
ZXPFR	Power Resumption Facility; optional c
ZQPTTY	Communications TTY line protocol handler c
ZQPVIP	Communications VIP line protocol handler c
ZQRCI	Communications RCI line protocol handler; optional c
ZQTDUM	Required for a communications configuration c
ZQTRAX	Internal Diagnostic Trace Capability c
ZXDEFM	Defective Memory Trap Handler c
ZQECP1	Auto Call channel control program; optional c
ZQPATC	ATD channel control program; optional c

Figure 7-1 (Cont). MOD 400 Executive Program Materials

ZQPCCP	STD channel control program; optional
ZQPTCP	TTY channel control program; optional
ZQPPVE	
ZQPACT	ACTD asynchronous LPH
ZQPACC	ACTD channel control program
ZQPPRT	Printer management
ZQPBDG	Badge management
ZQPSUR	common subroutines to ZQPPRT and ZQPBDG
ZQXVIP	
ZQXPVE	
ZQXBSC	
ZQXPBS	
ZQXATD	
ZQXTTY	
ZQXCRT	
ZQXXEC	
CLM_MCP	File of CLM directives used during stage 1 system startup with MLCP/DLCP-connected operator terminal; required if your installation has an MLCP/DLCP-connected operator terminal.
CLM_MDC	File of CLM directives used during stage 1 system startup with MDC-connected operator terminal; required if your installation has an MDC-connected operator terminal.
SYSLIB1	
EC	Execution command used for EC files; required if any START_UP.EC file used
CSD	Change system directory operator command; required

Figure 7-1 (Cont). MOD 400 Executive Program Materials

SYSLIB2

ABORT_BATCH
ABORT_GROUP
ABP
ABR
ACTB
ACTG
AGR
AMM
AMT
BYE
CB
CG
CGRQ
CMBX
CQR
CT
CVD
DB
DEBUG
DEL_CUM
DG
DMNC
DMON
ZELUPD
ZXLOG
ASSOC
BC
BPI
BPS
CBP
CD
CF
CKMSV
CKPTFILE
CMR
CP
CPA
CPAS
CPADE
CPDE
CPR
CR
d
CSD
CV
CVCY
CVDE
d
CWD
CX
DA

DP
DPEDIT
DPN
DT
EBR
d
EC
EGR
EQUAL
ETR
FO
LISTENER
LOAD
LQR
LSR
MAIL
MSG
MSW
NEW_PROC
NOW
OPER
PR_ELOG
PR_CUM
DBP
DCA
DD
DISSOC
DL
DX
ED
EX_PAM
FC
FCDE
FD
FDDE
GET
GQ
IM_PAM
ISLCON
IT
PX
RX
LA
LCA
LCD
HD
LINKER
LMR
LS
LSDE

PR_HOLD
QUERY
RDF
RDN
RESTART
SD
SET_ELOG
SET_LISTEN
SG
SMM
SSPB
SSPG
ST
START_ELOG
START_MAIL
STG
STOP_ELOG
STS
TIME
UNLD
UPD_CUM
USER
LTC
LWD
LX
M4_SYSDEF
MFA
PATCH
PR
PR_OK
QRPT
RECOVER
REMOVE
RESTORE
RL
RN
RQM
SA
SAVE
SCA
SDL
STTY
TPOS
UNSP
VALIDCKPT
WH
WS
DMBX

Figure 7-1 (Cont). MOD 400 Executive Program Materials

FORMS	TV	TV (cont.)	PROGS
VDAM_FORM01	CDUG1	MT9G1	TRANS
RDYT	DIMG1	MT7G1	
	PRMG1	ZTVM	
	TCSG1	ZTVIOD	

a
Present only on SAF medium

b
Present only on LAF medium

c
If the system is to include this capability, this system software must be present on the bootstrap volume before configuration.

NOTE: Software modules listed as being optional may be judiciously deleted if the user does not wish to configure a particular capability; otherwise, the appropriate software modules <u>must</u> be included with the Executive software if the user wishes to configure an optional capability.

d
Required for Stage 1 and Stage 2 system startup.

Figure 7-1 (Cont). MOD 400/MFS Executive Program Materials

```
SYSLIB2
  ASSEM
  MACROP
LDD
  OBJECT
    EXECUTIVE
      ZGQISB.O
      ZIMLCC.O
MACRO
  EXEC_LIB
    macro routines
  XNP_USER
    macro routines
```

Figure 7-2. Assembler/Macro Preprocessor Program Materials (SHL937)

```
SYSLIB2
  FORTRAN
  ZFRT
    object modules
```

Figure 7-3. FORTRAN Program Materials (SHL936)

```
SYSLIB2
  RPG
  AUTO
  AUTOR
```

Figure 7-4. RPG Program Materials (SHL926)

SYSLIB2

SORT

SORTC

MERGE

Figure 7-5. SORT/MERGE Program Materials (SHF910)

SID

ZQPBSC[1]

ZQHLPH[1]

ZQPPBS

SYSBIB2

TRANB

[1] These program materials must be transferred to the bootstrap volume before configuration.

Figure 7-6. File Transmission (Non-HONEYWELL Host) (SHC953)

SYSLIB2

COBOLI (intermediate level)

ZCIRT

object modules

Figure 7-7. Intermediate COBOL Program Materials (SHL925)

ZDRT

DEF_SAMPLE.S

object modules

Figure 7-8. DEF-I Program Materials (SHC917)

SYSLIB2

RBT

Figure 7-9. RBF/66 Program Materials (SHC915)

SID

ZQPPVE[1]

ZQRCI[1]

SYSLIB2

TRAN

[1] These program materials must be transferred to the bootstrap volume before configuration.

Figure 7-10. File Transmission (CII HONEYWELL BULL Host) Program Materials (SHC951)

SYSLIB2

HASP

Figure 7-11. HASP Workstation Facility Program Materials (SHC959)

SYSLIB2

WS2780

WS3780

Figure 7-12. 2780/3780 Workstation Facility Program Materials (SHC958)

SID
ZQPPBS[1]
SYSLIB2
IF3271
ZF32IN
ZCOBIF.0

[1] These program materials must be transferred to the bootstrap volume before configuration.

Figure 7-13. Programmable Facility/3271 Program Materials (SHC988)

SYSLIB2
TCLC TCLP

Figure 7-14. TCLF Program Materials (SHS941)

SYSLIB2
FORTRANA
ZFIRT
runtime routines

Figure 7-15. Advanced FORTRAN Program Materials (SHL944)

SYSLIB2

BASIC
BRTNUC
RESEQ
LNKBPRG.EC

ZBRT

runtime routines

ZBRTS

runtime routines

Figure 7-16. BASIC Program Materials (SHL942) Interpreter, (SHL943) Interpreter/Compiler, (SHL932) runtimes

SYSLIB2

DEF
DEFCMPL
DEFEXEC
DEFFIG
DEFPRNT
DEFSTA
DEFSPEC
DEFSUPR
DEFXFER

DEFMENU

DEF-II system forms

Figure 7-17. DEF-II Program Materials (SHC989)

SYSLIB2

COBOLA

ZCART

object modules

Figure 7-18. Advanced COBOL Program Materials (SHL945)

```
BCF                              FORMS

   ZQDMNT                           DTFF
   ZQDOPR
   ZQDSTA                        LDD
   ZQDBYE
   ZQDSTR                           INCLUDE
   ZQDSND
                                       BCFMAC.IN.A
   User-written TBUs
                                 BCFWRK

ZQDBCF                              BCFGEN

   ZQDATD                              BCFGEN.EC
   ZQDDMP                              User-written BCF generation
   ZQDDUM
   ZQDFWK
   ZQDINI
   ZQDIOC
   ZQDISN
   ZQDIO1
   ZQDIO2
   ZQDJB5
   ZQDLST
   ZQDMIS
   ZQDOIM
   ZQDPAI
   ZQDPRM
   ZQDPVE
   ZQDRES
   ZQDSCH
   ZQDSTD
   ZQDSUB
   ZQDTCL
   ZQDTSK
   ZQDTSV
   ZQDTTY
   ZQDTO1
   ZQDTO2
   ZQDT72
   ZQDOVL
   ZQDUSN
   ZQDUTR
```

Figure 7-19. BCF Program Materials (SHC987)

DTF

ZQDBYE
ZQDBCF
ZQDDBG
ZQDMNT
ZQDSND
ZQDSTA

ZQDDTF

ZQDATD
ZQDDMP
ZQDDUM
ZQDFWK
ZQDINI
ZQDIOC
ZQDISN
ZQDIO1
ZQDIO2
ZQDJBS
ZQDLST
ZQDMIS
ZQDOIM
ZQDPAI
ZQDPRM
ZQDPVE
ZQDRES
ZQDSCH
ZQDSTD
ZQDSUB
ZQDTCL
ZQDTSK
ZQDTSV
ZQDTTY
ZQDT01
ZQDT02
ZQDT72
ZQDOVL
ZQDUSN
ZQDUTR
ZQDATL
ZQDCNX
ZQDBYX
ZQDDFF
ZQDSTY
ZQDSPU
ZQDTPU

Figure 7-20. DTF Program Materials (SHC957)

FORMS

DTFF

LDD

INCLUDE

DTFMAC.IN

DTFWRK

DTFGEN

DTFGEN.EC
DTF.P

DTFTBU

COBTBU.EC
TBUI
TBUn

Figure 7-20 (Cont). DTF Program Materials (SHC957)

```
ZQDDRR                    DRR

   ZQDATD                    ZQDBYE
   ZQDDMP                    ZQDBCF
   ZQDDUM                    ZQDDBG
   ZQDFWK                    ZQDMNT
   ZQDINI                    ZQDSND
   ZQDIOC                    ZQDSTA
   ZQDISN
   ZQDIO1                 FORMS
   ZQDIO2
   ZQDJB5                    DTFF
   ZQDLST
   ZQDMIS                 LDD
   ZQDOIM
   ZQDPAI                    INCLUDE
   ZQDPRM
   ZQDPVE                       DRRMAC.IN.A
   ZQDRES
   ZQDSCH                 DRRWRK
   ZQDSTD
   ZQDSJB                    DRRGEN
   ZQDTCL
   ZQDTSK                       DRRGEN.EC
   ZQDTSV                       DRR.P
   ZQDTTY
   ZQDT01                    DRRTBU
   ZQDR02
   ZQDT72                       COBTBU.EC
   ZQDOVL                       TBU
   ZQDUSN                        .
   ZQDUTR                        .
   ZQDATL                        .
   ZQDCNX                       TBUn
   ZQDBYX
   ZQDDFF
   ZQDSTY
   ZQDSPJ
   ZQDTPU
   ZQDFIL
   ZQDDFB
```

Figure 7-21. DTF With Restart/Recovery Program Materials (SHC986)

```
SYSLIB2

  TRAND
  FTF64
  RBF64
  UNSPOL
```

Figure 7-22. Program Materials For Remote Batch 64/DPS (SHC954)

```
SID

  TMMLPH

SYSLIB2

  RB6
```

Figure 7-23. Program Materials For Remote Batch IRIS (SHC990)

```
SID

  ZNVDAM
  ZNV72F
  ZNV77F

SYSLIB2

  VFORMS
  ZNVUPR
  ZNVCRM
```

Figure 7-24. Program Materials For Display Formatting and Control (SHC961)

SECTION VIII

TECHNICAL NOTES

This section describes a number of miscellaneous topics, most of which pertain only to certain installation sites. Before attempting any use of the system, you should ascertain which of these topics apply to your installation and proceed accordingly. The following topics are described:

- CSD (Change System Directory) operator commands in START_UP.EC file for system task group
- Transferring the contents of ^ZSYS51 or ^ZSYS61 to the fixed platter of a two-platter drive
- Syystem search rules and the system commands
- Procedure for transferring software modules

CSD OPERATOR COMMANDS IN START_UP.EC FILE OR SYSTEM TASK GROUP

If a START_UP.EC file is used for the system task group during a stage 3 system startup (1), it must be immediately subordinate to the root directory of the bootstrap volume. This START_UP.EC file should contain one or more CSD (Change System Directory) operator commands to reassign the system libraries -LIB1 and -LIB2 away from their defaults (the default assignment fo both -LIB1 and -LIB2 is SYSLIB1 on the bootstrap volume).

If the system is to be bootstrapped from a non-diskette volume, the START_UP.EC file for the system task group should contain the following two CSD operator commands (in addition to any other appropriate operator commands):

```
CSD -LIB1 pathname_to_user-library

CSD -LIB2 >SYSLIB2
```

(1)
The stages of system startup are described in Section III.

These operator commands will appropriately specialize the system's search rules for this environment. When the system's loader seeks a bound unit to be loaded, it will first search the user's working directory (default), then (if necessary) the user_library (-LIB1), then (if necessary) SYSLIB2 on the bootstrap volume (-LIB2).

If the system is to be bootstrapped from diskette (^ZSYS00), the START_UP.EC file for the system task group should contain the following CSD operator command:

    CSD -LIB1 ^ZSYS01>SYSLIB1

This operator command will appropriately specialize the system's search rules for a diskette environment. When the system's loader seeks a bound unit to be loaded, it will first search the user's working directory (default), then (if necessary) ^ZSYS01>SYSLIB1 (-LIB1). Typically, in a diskette environment, the operator uses dynamic CSD operator commands to assign -LIB2 to the pathname that will be searched (if necessary) following ^ZSYS01>SYSLIB1.

TRANSFERRING CONTENTS OF ^ZSYS51 OR ^ZSYS61 TO FIXED PLATTER

If your system is cartridge-disk-based or cartridge-module-disk-based, you may wish to copy the contents of your system volume to the fixed platter of a two-platter drive. The original HONEYWELL-supplied system volume may then be dismounted and saved as backup. The following procedure is recommended.(1)

1. Mount system volume ^ZSYS51 on RCD00, or mount system volume ^ZSYS61 on RCM00.

2. Perform a stage 2 system startup, bootstrapping the system from RCD00 or RCM00.

(1)
Assume that this action involves a stage 2 system startup: the CLM_USER file is assumed to include the following DEVICE directives:

For cartridge disk-based systems:

    DEVICE RCD00,1,6,X'1400' (removable cartridge disk)

    DEVICE FCD00,3,6,X'1400' (fixed cartridge disk)

For cartridge module disk-based systems:

    DEVICE RCD00,1,6,X'1400' (removable cartridge module disk)

    DEVICE FCM00,3,6,X'1400' (fixe cartridge module disk)

3. Use the following CV (Create Volume) command to initialize the fixed platter of the cartridge disk:

   CV !FCD00 -FT ZUSR51

   (ZUSR51 is merely an example of a user-supplied volume_id).

   If your system is cartridge module disk-based, use the following CV (Create Volume) command to initialize the fixed platter of the cartridge module disk:

   CV !FCM00 -FT ZUSR61

   (ZUSR61 is merely an example of a user-supplied volume_id).

4. Use the following CP (copy) command to copy the entire contents (except volume_id) of ^ZSYS51 to ^ZUSR51:

   CP !RCD00>ZSYS51 !FCD00 -VOL

   If your system is cartridge module disk-based, use the following copy (CP) command to copy the entire contents (except volume_id) of ^ZSYS61 to ^ZUSR61:

   CP !RCM00>ZSYS61 !FCM00 -VOL

5. After the copy is completed, press STOP.

6. Performs another stage 2 system startup, this time bootstrapping the system from FCD00 or FCM00, as appropriate. (Enter 1401 into the RI-register after the TRAFFIC light turns off following Stop, CLeaR, Load, Execute).

7. When system startup is complete, dismount ^ZSYS51 from RCD00 or ^ZSYS61 from RCM00.

8. Mount a new cartridge disk on RCD00 or RCM00, as appropriate.

9. For cartridge disk-based systems, use the following CV command to initialize the new cartridge disk:

   CV !RCD00 -FT USRVOL

   For cartridge module disk-based systems, use the following CV command to initialize the new cartridge module disk.

   CV !RCM00 -FT USRVOL

   (USRVOL is merely an example of a user-supplied volume_id).

10. Use the following CV command to create a bootstrap record on ^USRVOL: (The bootstrap record must be on the removable volume, even through the system is to be bootstrapped from the fixed platter).

```
CV !RCD00>USRVOL -BOOT [X'1401']1
```

For cartridge module disk-based system, use the following command:

```
CV !RCM00>USRVOL -BOOT [X'1401']1
```

NOTE: After a bootstrap record on a removable cartridge disk or cartridge module disk has been used to bootstrap the system executive from the fixed cartridge disk or cartridge module disk, the volume_id of the fixed cartridge disk or cartridge module disk is assumed to be to the left of an absolute pathname that begins with a greater-than sign (>).

SYSTEM SEARCH RULES AND THE SYSTEM COMMANDS

The system builder should insure that system commands (or at least that subset of system commands most frequently used) are accessible to the system's loader.

When the system's loader seeks a bound unit to be loaded, it proceeds as follows:

If the user supplies a full pathname (one beginning with a circumflex and the volume id), then the system loader will search for that specific element.

If the user supplies a simple (single-element) pathname, then the system loader will search for the element acording to a set of search rules.

The search rules specify the order in which the loader is to search one or more directories for the element in the simple pathname. The default search rules for MOD 400 are (1) first search the user's working directory, (2) next search the system directory SYSLIB1 on the root volume and (3) finally search the system directory SYSLIB2 on the root volume.

(1)
If X'1401' is not specified, 1401 must be entered into the R1-register (as shown in step 6) whenever the system is bootstrapped from the fixed platter; however, other bootstrap options (see Table 3-2) can be used more flexibly if the bootstrap channel number (plus options) is entered into the R1-register each time the system is bootstrapped.

The operator can use the CSD command to change the system search rules. In normal operations the operator should use the CSD command to specify that one of the directories to be searched is the system directoryy containing the system commands. The user should note that those system command not accessible to the loader under the search rules must be specified with a full pathname. (For example, the CP command would require the user to specify ^volume_id>SYSLIB1>CP).

PROCEDURE FOR TRANSFERRING SOFTWARE

A procedure for transferring modules is described below.

Prior to performing the procedure, the user should, of course, determine the modules to be transferred and the unneeded elements on the volume to which new software is to be added. Section VII contains a description of the contents of all disks.

1. Mount the volume containing the modules to be transferred on an available drive.

2. Use a list names command (LS) to ascertain the size of the modules to be transferred.

3. Mount the volume to which the modules are to be transferred.

4. Use an LS command to ascertain the size of unneeded elements.

5. Release unneeded elements.

6. Mount a new diskette on an available drive. Use a create volume command (CV) to initialize this diskette. (Use a temporary, unique volume identifier).

7. Use a create directory command (CD to create appropriate directories on the new diskette.

8. Use a copy command (CP) to copy the remaining contents of the volume mentioned in step 3 to the new volume.

9. Ensure that the volume referenced in step 1 above is mounted and use a copy command to copy the appropriate modules to the new volume.(1)

10. Use a CV command to rename the new volume. (If the new name duplicates the name of a currently-mounted volume, the system will issue a dismount message, directing you to dismount the newly-renamed volume).

(1)
When adding elements to the bootstrap volume, the system builder should remember that if a rollout capability is to be established, sufficient space must be available on the volume for the ROLLOUT file created during system configuration; this rollout file must be large enough to accommodate the batch memory pool.

APPENDIX A

STARTUP HALTS

Startup halts may be classified in three categories:

- Halts related to bootstrap operation.
- Error halts related to Configuration Load Manager.
- Error halts related to other aspects of system initialization.

A halt related to the bootstrap operation may have been intentionally requested, or it may reflect an error condition. A bootstrap halt is intentionally requested by setting ON bit 13 of the 16-bit (four hexadecimal digits) bootstrap channel number (see Table 3-2). In the event of this type of bootstrap halt, the following register contents are significant:

- R1-register contains bootstrap channel number.
- R2-register contains address mode flag: 0 indicates SAF; 1 indicates LAF.
- R3-register identifies the bootstrap device type:

  0 indicates cartridge disk, cartridge module disk, or mass storage unit; 1 indicates diskette.

Error halts during bootstrap result in a 16nn value in the R1-register. See the <u>System Messages</u> manual for a description of the 1611, 1612, and 1616 error halts.

NOTE: If a bootstrap halt occurs with a 1616 error condition, there is a possibility that the D7 register will contain no error status. To obtain status in such circumstances, select D7 prior to retrying the operation and observe D7 during bootstrap processing.

Error halts related to the Configuration Load Manager result in a 13nn value in the R1-register, usually additional information relative to the halt is available in, or through, other registers. See the <u>System Messages</u> manual. Note that if an operator terminal is connected to an MDC (and if its DEVICE directive has already been read), a Configuration Load Manager error condition produces an error message at the operator terminal; in this case, a halt occurs (in addition to the message) <u>only</u> under certain error conditions.

Configuration Load Manager error messages are described in the System Messages manual.

Error halts related to other aspects of system initialization result in a 99nn value in the RI-register; in some cases additional information relative to the halt is available in, or through, other registers. See the <u>System Messages</u> manual.

APPENDIX B

SYSTEM OVERLAYS

Table B-1 provides a list of all system overlays that can be made permanently resident in the system area of memory. Each overlay to be made permanently resident must be named in a RESOLA directive (described in Section V). Multiple system overlays can be named in a single RESOLA directive.

Beside each overlay name in Table B-1 is the approximate size (in decimal words) the overlay requires in memory if it is made permanently resident. In certain cases, footnotes indicate groups of overlays that should all be made permanently resident if optimum speed is desired for that function.

Table B-1. System Overlays

Overlay Name	Size (in Words)	Function
Executive Services		
OIOIMS	415	OIM dispatcher (all LRN 0 orders)
OIMIMO	321	OIM input processor
OIOIM1	401	OIM command processor, output completion, and break handler
OIOIM2	331	OIM output control
OIOIM3	417	OIM diagnostics
OIOIM4	451	OIM abort group request purge and OIM monitor call handler
OMSTAT	277	$STMP (status memory pool) macro handler, and get specific memory block
OOBIND	226	Loader - Processing of relocation items

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OOCROT	169	Loader - Create OAT and release, wait and recall overlay MCL handler
OOMSCO	234	Loader - Processing of SLIC load units, unloading of bound units
OOPACK	255	Loader - Resolution of symbols defined by LDBU's and of root entry names
OOR001	243	Loader - Load of bound unit (phase 1)
OOR002	296	Loader - Load of bound unit (phase 2)
OUEML2	241	Error message library processing
OUERS1	302	Error reported (phase 1)
OUERS2	357	Error reported (phase 2)
OUERS3	319	Error reported: trap errors only
OXACSP	149	Break task activation/suspension
OXAVR1	50	Automatic volume recognition
OXBXFR	114	Bound-unit transfer MCLs
OXC_IX	397	Convert from/to external/internal date/time
OXCTSK	491	Create and spawn task
OXCDRV	245	Create driver (initialization use only)
OXCGRP	499	Create and spawn group
OXCKPT	323	Checkpoint, determination of checkpointability and phase 1 of checkpoint
OXCKP1	226	Checkpoint - phase 2
OXCKP2	498	Checkpoint - phase 3
OXCKPA	479	Checkpoint file assignment - part 1
OXCKPA2	474	Checkpoint file assignment - part 2, and checkpoint file disassignment

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
0XDFRT	511	Place deferred spawn task request
		Spawn deferred task (after defer interval)
0XDLGP	448	Delete/abort group (and abort group request)
0XDTSK	488	Task Deletion (created or spawned tasks)
		Request task deletion (created tasks)
0XELOG	471	Error logging MCLs
0XESWO	84	External switch handler
0XGACT	323	Group activate (rollout phase 2)
0XGSSP	274	Group suspend (phase 1)
0XGRQS	312	Start group request (phase 1)
0XGRS2	422	Group request startup (phase 2)
0XGRQT	461	Terminate group request (phase 1)
0XGRT2	263	Terminate group request (phase 2)
0XGRT3	308	Group request termination (phase 3)
0XMF01	337	Message facility: MCL dispatcher
0XMF02	480	Message facility task request dispatcher: Call MF MCLs
0XMF03	489	Message facility activate mailbox: open mailbox file (accept and receive - disk queuing)
0XMF04	393	Message facility: accept MCLs
0XMF05	370	Message facility: accept MCLs and read record from message facility file (accept and receive - disk queuing)
0XMF06	398	Message facility: initiate MCLs
0XMF07	257	Message facility: send MCLs

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OXMF08	383	Message facility: send MCLs (disk queuing/backup); terminate message MCLs (send)
OXMF09	377	Message facility: receive MCLs
OXMF10	479	Message facility: cancel enclosure level MCLs, delete quarantine unit (receive MCLs - disk queuing/backup)
OXMF11	427	Message facility: terminate message MCLs (receive)
OXMF12	340	Message facility: close mailbox (all terminate message MCLs - disks queuing/backup; initiate MCLs - disk queuing; message count MCLs) terminate message MCLs (waiting acceptor)
OXPCBJ	139	Process command line: activate functions
OXPCL1	421	Process command line: (phase 1)
OXPCL2	434	Process command line: (phase 2)
OXPCL3	365	Process command line: break interrupt handler
OXRQJP	241	Request and spawn group (phase 1)
OXRQG2	328	Request and spawn group (phase 2)
OXRQG3	261	Request and spawn group (when mailbox queued request)
OXRST	504	Restart phase 1
OXRST0	428	Restart phase 2
OXRST1	483	Restart phase 3
OXRST2	507	Restart phase 4
OXRST3	495	Restart phase 5
OXRSTB	428	Restart - terminal substitution
OXRSTF	498	Restart phase 6 (last)
OXRSTH	164	Restart - bound units preservation

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OXRSTP	317	Restart - trap processing and suspension activation during restart
OXRSER	495	Restart - error processing
OXSIO1	319	Standard input/output (new user, command_in)
OXSIO2	275	Standard input/output (new user, user_out)
OXSIO3	186	Standard input/output (input line expansion)
OXSIO4	119	Standard input/output (system error_out handler)
OXSMCL	200	Process semaphore MCLs
OXTR17	204	Defective memory trap handler
OXTRP0	315	Connect user trap handler (enable, disable user traps)
OXTRR	139	Terminal request handler
OXUPF0	265	Identification/information macros
OXUPF1	138	Identification/information macros
OXWCS1	62	Load Writable Control Store
OXWTL0	296	Wait list processing
File Managment		
OYASF	97	Associate/disassociate file - MCL
OYCRF	358	Create file - MCL
OYCRR	295	Create record descriptors - MCL
OYDLF	506	Delete file - MCL
OYLKF	388	Link file - MCL
OYMDF	428	Modify disk file attribute/name - MCL
OYULF	247	Unlink file - MCL

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OZALX	292	Allocate disk space - record extent in directory
OZALX2	459	Allocate disk space - GET space for one extent
OZCRF2	408	Create file - primary/alternate index parameter
OZCRF3	298	Create file - calc/ids-2 parameter
OZCRF4	259	Create file - permanent file
OZCRF5	427	Create file - temporary file and subroutines
OZCRR2	511	Build record descriptor
OZLKDR	248	Link and rewrite directory record
OZULDR	107	Unlink directory record
OYGIDV	83	Set device information - MCL
OYGIF	506	Get file information - MCL
OYGTF	403	Get file - MCL
OYRMF	390	Remove file - MCL
OZGTF2	217	Get file -save/restore interface
OZGTF3	442	Get file - tape parameter processing
OZCDDB	313	Get file - create Device Descriptor Block (DDB)
OZCFDB	509	Get file - create File Descriptor Block (FDB)
OZGFCB	504	Get file - get File Control Block (FCB)
OZGTRF	277	Get file - get related files
OYOPN	443	Open file - MCL
OZOPFD	278	Open file - device files
OZOPFF	111	Open file - fixed relative file

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
0Z0PFU	357	Open file - sequential/relative/alternate indexes
0YCLS	380	Close file - MCL[1]
0ZCLFD	160	Close file - device files
0ZGRPT	147	Group termination
0YACL	471	Set/delete Access Control (ACL) - MCL
0ZACL2	486	Set/delete Access Control (ACL) - set ACL
0YAVR	484	Automatic Volume Recognizer (AVR) - MCL
0YCMR	437	Cancel Mount Request (CMR) - MCL
0YIT	352	Initialize Tape (IT) - MCL
0YSWP	348	Swap volume - MCL
0ZAVR2	507	AVR - device file other than tape
0ZAVR3	460	AVR - initialize disk Volume Descriptor Block (VDB)
0ZVMT	403	Volume mount - request disk volume mount
0ZVMT2	403	Volume mount - request device/tape volume mount
0ZVMT3	406	Volume mount - search for free tape
0YCWD	396	Change working directory - MCL
0YXPA	437	Expand file pathname - MCL
0YSTTY	447	Set TTY - MCL
0YWTFL	241	Wait file - MCL
Storage Management		
0ZSME	181	I/O error handler - common to all files
0ZSME1	181	I/O error handler - disk files
0ZSME2	222	I/O error handler - message handler

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OZSMT	111	Storage management positioning function for tape
OYBMCR	506	Buffer pool - create/delete (MCL) [j]
OYBMGI	227	Buffer pool - statistics (MCL)
OYBMDL	366	Buffer poil - Delete adjust [j]
OZBM2	287	Buffer pool - flush buffers and disassociate buffers
Data Management		
OYDMR2	167	Fixed relative - MCL (read/write/rewrite/delete record)
OYDRM3	203	UFAS relative - MCL (read/write/rewrite/delete record) [b]
OZDMR	284	UFAS relative - positioning
OYDMS1	255	String - MCL (read/write/rewrite record)
OZDMS2	511	UFAS sequential - MCL (rewrite/delete record)
OZACCB	173	UFAS indexed/random - allocate CCB (Currency Control)
OZUCCB	448	UFAS indexed/random - update CCB (Currency Control) [c,d,e]
OZUSUB	242	UFAS indexed/random - common subroutines [a,c,d]
OYDMI1	510	Primary indexed - MCL (read/rewrite record) [a]
OYDMI2	299	Primary indexed - MCL (write record in load mode) [a]
OYUPI1	476	Primary indexed - MCL (insert/delete record) [a]
OZIEXT	512	Primary indexed - initialize for extend mode
OZIOPF	500	Primary indexed - open and close
OZIPSN	502	Primary indexed - positioning
OYDMC	364	Calc - MCL (read/write/rewrite/delete record) [c]

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OZCER	76	Calc - error processing
OZCINV	202	Calc - inventory processing [c]
OZCKEY	268	Calc - hash/compare key [c]
OZCLNK	133	Calc - link/unlink calc record [c]
OZCOPF	262	Calc - open
OZCPSN	239	Calc - locate/position record [c]
OZCSUB	222	Calc - subroutines [c]
OZDMC	72	Calc record modification
OYDMX1	507	Alternate index - MCL (read/write/rewrite/delete record [e]
OZDMX	391	Alternate index - update alternate indexes [e]
OZXENT	461	Alternate index - add/delete index entries [e]
OZXER	415	Alternate index - error processing
OZXLVI	325	Alternate index - create index levels
OZXPSN	470	Alternate index - positioning function
OZXSPL	499	Alternate index - split CI [e]
OZXSUB	366	Alternate index - subroutines
OYDMX2	395	Alternate index - MCL (read/write/rewrite/delete entry)
OYDMT	501	Tape - MCL (read/write record) [f]
OZDMT	487	Tape - spanned record (read and write)
OZTCLF	285	Tape - close
OZTEBC	276	Tape - EBCDIC and ASCII conversion [g]
OZTEXD	469	Tape - expiration date checking
OZTOPI	483	Tape - open (common)

Table B-1 (Cont). System Overlays

Overlay Name	Size (in words)	Function
OZTOP2	512	Tape - labeled tape
OZTOP3	316	Tape - unlabeled tape
OZTOP4	491	Tape - process tape labels
OZTPSN	279	Tape - positioning
OZTSUB	237	Tape -subroutines, ASCII and binary conversion [f]
OZTUTL	380	Tape - subroutines
Checkpoint, Restart, Record Lock, and Recovery Services		
OYCKPT	293	Checkpoint - MCL
OZCKER	427	Checkpoint - error reporting
OZCKPT	479	Checkpoint - internal check point
OYRSRT	438	Restart - MCL
OYUREC	293	Unlock record - MCL
OZBEF	372	Before image - data management interface [h]
OZBIGF	255	Before image - reservation
OZBIIR	417	Before image - initialization
OZBIOR	296	Before image - offline recovery
OZBIWR	275	Before image - write before image [h]
OZCLPT	488	Clean point - MCL [h]
OZROLB	413	Roll back

[a] This overlay should be made permanently resident if optimum speed is desired in processing UFAS indexed files.

[b] This overlay should be made permanently resident if optimum speed is desired in processing UFAS relative files.

[c] This overlay should be made permanently resident if optimum speed is desired in processing UFAS random files.

[d] This overlay should be made permanently resident if optimum speed is desired in processing indexed and random files, and alternate indices.

[e] This overlay should be made permanently resident if optimum speed is desired in processing files with alternate indices.

[f] This overlay should be made permanently resident if optimum speed is desired in processing UFAS tape files.

[g] This overlay should be made permanently resident if optimum speed is desired in processing EBCDIC tape functions.

[h] This overlay should be made permanently resident if optimum speed is desired in processing using record locking/recorvery.

[i] This overlay should be made permanently resident if optimum speed is desired in processing file management functions.

[j] This overlay should be made permanently resident if optimum speed is desired in processing using buffer pool mangement.

APPENDIX C

EQUIPMENT REQUIREMENTS

This appendix contains minimum system guidelines and a list of supported equipment.

MINIMUM SYSTEM GUIDELINES

Minimum system guidelines are offered as aids to system builders who must determine the resources required to perform meaningful processing at the installation. Minimum system guidelines provide base configurations from which the system builder can calculate what additional memory and peripheral devices (if any) are necessary for efficient and effective operations. These guidelines will remain in effect until the next major release of MOD 400.

Minimum System for Program Development

Program development activity in a single-user environment can be accomplished with 64K words of memory (SAF or LAF) and a compliment of peripheral devices limited to the following: one operator terminal, one printer, and up to four cartridge disks, cartridge module disks or mass storage units.

In such an environment, the user can obtain an online memory pool of 38K words (remaining memory would be required to accommodate the resident executive and a 6K-word system memory pool). Additionally, this minimum system would allow the user to provide for the following exhaustible resources: two system overlay areas, 17 traps save areas, and 30 intermediate request blocks.

Minimum System for Online Applications (Execute-Only) SAF Mode

A configuration with 48K words of memory (SAF mode) and a compliment of peripheral devices limited to those described below will permit the execution of a wide variety of user applications in a single-user environment.(1)

(1)
The 2780/2780 Workstation Facility can be used in an environment as described above with 32K words of memory.

- Two disks (1)

- One printer

- One teleprinter-compatible terminal connected to an MLCP or DLCP

- A PVE link

In such an environment, the user can obtain an online memory pool of 13K words (remaining memory would be required for the resident executive and a 6K word system memory pool). Additionally, this minimum system would allow the user to provide for the following: the commercial simulator, two system overlay areas, 17 trap save areas, and 30 intermediate request blocks.

Minimum System for ONline Applications (Execute-Only) LAF Mode

A configuration with 64K words of memory and a compliment of peripheral devices limited to those described below will permit the execution of a weide variety of user applications in a single-user environment.(1)

- Two disk (2)

- One printer

- One VIP7200/05 or VIP7801/02 terminal connected to an MLCP or DLCP

- A PVE link

In such an environment, the user can obtain an online memory pool of 16K words (remaining memory would be required for the resident executive and a 6K word system memory pool). Additionally, this minimum system would allow the user to provide for the intermediate request blocks and the Display Formatting and Control Facility software.

(1)
The 2780/2780 Workstation Facility can be used in an environment as described above with 32K words of memory.

(2)
In an all-diskette system, it is recommended for ease-of-use that four diskettes be selected.

SUPPORTED HARDWARE

Table C-1 lists Level 6 equipment and options supported by MOD 400 when operating on the Model 23 central processor.
Table C-2 lists the Level 6 equipment and options supported by MOD 400 on other central processor models.(1)

Table C-1. Supported Hardware - Model 23

Category	Marketing Identifier	Description
Central Processor	CPS9351	23 Minimount, 32K, FCP, 5-slot, no-disk
	CPS9352	23 Minimount, 32K, FCP, 9-slot, no-disk
	CPS9353	23 Minimount, 64K, FCP, 5-slot, no-disk
	CPS9354	23 Minimount, 64K, FCP, 9-slot, no-disk
	CPS9356	23 Minimount, 32K, BCP, 5-slot, no-disk
	CPS9357	23 Minimount, 32K, BCP, 9-slot, no-disk
	CPS9358	23 Minimount, 64K, BCP, 5-slot, no-disk
	CPS9359	23 Minimount, 64K, BCP, 9-slot, no-disk
	CPS9370	23 Minimount, 32K, FCP, 7-slot, 26MB cartridge module disk
	CPS9371	23 Minimount, 64K, FCP, 7-slot, 26MB cartridge module disk
	CPS9372	23 Minimount, 64K, FCP, 7-slot, 80MB cartridge module disk
	CPS9373	23 Minimount, 32K, BCP, 7-slot, 26MB cartridge module disk
	CPS9374	23 Minimount, 64K, BCP, 7-slot, 26MB cartridge module disk
	CPS9375	23 Minimount, 64K, BCP, 7-slot, 80MB cartridge module disk
Console Adapters	KCM9301 KCM9302	Single console adapter Dual console adapter

Table C-1 (Cont). Supported Hardware - Model 23

Category	Marketing Identifier	Description
Consoles	DKU9101 DKU9102 DKU9103 DKU9104 TWU9104 TWU9106 TWU9108 TWU9110	VIP 7100 console, 64-character set VIP 7100 console, 96-character set VIP 7200 console, 64-character set VIP 7205 console, 96-character set 30CPS console typewriter 120CPS console typewriter 30CPS console with numeric keypad 120CPS console with numeric keypad
Printer Adapter	PRM9301	M23 printer adapter
Line Printers	PRU9103 PRU9104 PRU9105 PRU9106 PRU9108 PRU9109	240LPM printer; 96-character set 300LPM printer; 64-character set 440LPM printer; 96-character set 600LPM printer; 64-character set 660LPM printer; 96-character set 900LPM printer; 64-character set
Matrix Printers	PRU9112 PRU9114	Matrix printer; 120CPS Matrix printer; 160CPS
Diskette Adapter	DIM9301	M23 diskette adapter
Diskettes	DIU9101 DIU9102 DIU9103 DIU9104	Single-diskette (single-sided); 256KB Dual-diskette (single-sided); 2x256KB Single-diskette (double-sided); 512KB Dual-diskette (double-sided); 2x512KB
Cartridge Module Disks	CDS9336 CDS9337 CDS9338 CDS9339 CDU9136 CDU9137 CDU9138 CDU9139	26MB cartridge module disk in cabinet with 23 controller 2x26MB cartridge module disk in cabinet with 23 controller 80MB cartridge module disk in cabinet with 23 controller 2x80MB cartridge module disk in cabinet with 23 controller 26MB cartridge module disk in cabinet without controller 2x26MB cartridge module disk in cabinet without controller 80MB cartridge module disk in cabinet without controller 2x80MB cartridge module disk in cabinet without controller

Table C-1 (Cont). Supported Hardware - Model 23

Category	Marketing Identifier	Description
Communication Adapters	DCM9301	M23 communication adapter: 2 asynchronous lines
	DCM9302	M23 communication adapter: 1 synchronous lines
	DCM9303	M23 communication adapter: 2 synchronous lines
	DCM9304	M23 communication adapter: 1 synchronous and 1 asynchronous line
Terminals	<u>Asynchronous Terminals</u>	
	VIP7100	CRT terminal, 64-character set
	VIP7105	CRT terminal, 96-character set
	VIP7200	CRT terminal, 64-character set
	VIP7205	CRT terminal, 96-character set
	VIP7801	CRT terminal, 12-inch screen
	VIP7802	CRT terminal, 15-inch screen
	VAF7821	Buffered printer adapter for all VIP 7800 terminals
	TWU1001	Keyboard/printer, 30 cps
	TWU1003	Keyboard/printer, 30 cps
	TWU1005	Keyboard/printer, 120 cps
	PRU1001	Printer terminal, 30 cps
	PRU1003	Printer terminal, 30 cps
	PRU1005	Printer terminal, 120 cps
	<u>Synchronous Terminals</u>	
	VIP7700	CRT terminal, 62-character set
	VIP7705	CRT terminal, 95-character set
	VIP7700R	CRT terminal, 63-character set
	VIP7705R	CRT terminal, 95-character set
	VIP7760 Subsystem	CRT terminal, 95-character set
	VIP7804	CRT terminal, 12-inch screen
	VIP7805	CRT terminal, 15-inch screen
	VAF7821	Buffered printer adapter for all VIP 7800 terminals
	ROP Printers	ROP printers are generally supported

Table C-1 (Cont). Supported Hardware - Model 23

Category	Description
Modems	Modems supported for asynchronous communications terminals are: . 103, 113, 202 type modems. (These modems must be equipped with the option to disconnect the data set after a carrier drop of 110 milliseconds). . CII HONEYWELL BULL modem bypass. . Modem types where the connection, disconnection, and data set control settings can be user-specified. Modems supported for synchronous communications terminals are: . 201, 201A, 201C, 208, and 208B type modems. . CII HONEYWELL BULL modem bypass. . Modem types where the connection, disconnection, and data set control settings can be user-specified. . Broadband modem. A modem is not required for a direct-connect asynchronous terminal, or a synchronous terminal with a timing source in the terminal or in the MLCP.
System Building Products	Writable Control Store (CPF9509) is offered for system builder use only. Although Writable Control Store is usable under GCOS 6 MOD 400, full end-user software support is not available.

Table C-2. Hardware Supported - Model 3X, 4X and 5X

Category	Marketing Identifier	Description
Central Processors	CPS9470 CPS9471 CPS9472 CPS9473	Model 33 Central Processors: memory to 64K words
	CPS9560 CPS9561 CPS9562 CPS9563	Model 43 Central Processors: memory to 1024K words
	CPS9566 CPS9567	Model 47 Central Processors: Commercial Model: memory to 1024K words
	CPS9570	Model 53 Central Processors: includes MMU and Cache: memory to 1024K words
	CPS9572	Model 57 Central Processors: Commercial Model weith MMU and Cache: memory to 1024K words
Multiple Device Controller	MDC9101	Multiple device controller
Console Device-pac	KCM9101	Keyboard console device-pacs
Consoles	DKU9101 DKU9102 DKU9103 DKU9104 TWU9104 TWU9106 TWU9108 TWU9110	VIP 7100 console VIP 7105 console VIP 7200 console VIP 7105 console 300CPS console typewriter 120CPS console typewriter 30CPS console with numeric keypad 120CPS console with numeric keypad
Printer Device-pac	PRM9101	Printer device-pac
Line Printers	PRU9103 PRU9104 PRU9105 PRU9106 PRU9108 PRU9109	240LPM, 96-character set 300LPM, 64-character set 440LPM, 96-character set 600LPM, 64-character set 660LPM, 96-character set 900LPM, 64-character set
Matrix Printers	PRU9112 PRU9114	120CPS matrix printer 160CPS matrix printer

Table C-2 (Cont). Hardware Supported - Model 3X, 4X and 5X

Category	Marketing Identifier	Description
Card Reader Device-pac	CRM9101	Card reader device-pac
Card Readers	CRU9108 CRU9109 CRU9110 CRU9111 CRU9112 CRU9113	300CPM 300CPM, IBM mark sense 300CPM, HIS mark sense 500CPM 500CPM, IBM mark sense 500CPM, HIS mark sense
Card Punch and Card Reader/ Punch Device-Pac (Adapter)	CRM9103	Device-Pac for card punch and card reader/punch
Card Punch	PCU9101	Card punch, 100 cpm
Card Reader/ Punch	CCU9101	Card reader/punch, 400/100 cpm
Diskette Device-pacs	DIM9101 DIM9102	Diskette device-pac for 2 single-sided diskettes Diskette device-pac for (2) 1- or 2-sided diskettes
Diskettes	DIU9101 DIU9102 DIU9103 DIU9104	Single diskette (single-sided), 256KB Dual diskette (single sided), 2x256KB Single diskette (double-sided), 512KB Dual diskette (double-sided), 2x512KB
Cartridge Disks	CDU9116 CDS9116	10MB (5+5) cartridge disk with controller 10MB (5+5) additional disk without controller
Cartridge Module Disks	CDS9136 CDS9137 CDS9138 CDS9139 CDU9136 CDU9137	26MB (13F + 13R) in cabinet with controller 26MB in cabinet with controller 80MB (67F + 13R) in cabinet with controller Dual 80MB in cabinet with controller 26MB in cabinet without controller Dual 26MB in cabinet without controller

Table C-2 (Cont). Hardware Supported - Model 3X, 4X and 5X

Category	Marketing Identifier	Description
	CDU9138	80MB in cabinet without controller
	CDU9139	Dual 80MB in cabinet without controller
	CDS9123	26MB (13F + 13R) with controller
	CDS9125	80MB (67F + 13R) with controller
	CDU9123	26MB add-on unit without controller
	CDU9125	80MB add-on unit without controller
Mass Storage Units	MSS9102	Mass storage unit in cabinet with controller
	MSS9104	256MB mass storage unit in cabinet with controller
	MSS9107	Dual 67MB mass storage unit in cabinet with controller
	MSU9102	67MB mass storage add-on unit without controller
	MSU9104	256MB mass storage add-on unit without controller
	MSU9107	Dual 67MB mass storage add-on unit without controller
Magnetic Tape Controllers	MTC9101	Magnetic tape controller, NRZI drives
	MTC9102	Magnetic tape controller, PE/NRZI drives
Magnetic Tape Device-Pacs	MTM9101	Magnetic tape Device-pac, 7-track drives
	MTM9102	Magnetic tape Device-pac, 9-track drives
Magnetic Tape Drives	MTU9104	Magnetic tape drive, 9-track NRZI, 45 inches per second (ips)
	MTU9105	Magnetic tape drive, 9-track NRZI, 75 ips
	MTU9109	Magnetic tape drive, 9-track NRZI/PE, 45 ips
	MTU9110	Magnetic tape drive, 9-track NRZI/PE, 75 ips
	MTU9112	Magnetic tape drive, 7-track NRZI, 45 ips
	MTU9113	Magnetic tape drive, 7-track NRZI, 75 ips
	MTU9114	Magnetic tape drive, 9-track PE, 45 ips
	MTU9115	Magnetic tape drive, 9-track PE, 75 ips

Table C-2 (Cont). Hardware Supported - Model 3X, 4X and 5X

Category	Marketing Identifier	Description
Communication Controllers	MLC9103	Multiline communications processor
	MLC9101	MLCP with 8 asynchronous lines; includes 4 communication-pacs
	MLC9102	MLCP with 8 synchronous lines; includes 4 communication-pacs
	MLC9104	MLCP with 8 current loop lines; includes 4 communication-pacs
Communications-Pacs	DCM9101	Communications-pac, two asynchronous lines, cable
	DCM9102	Communications-pac, one asynchronous lines, cable
	DCM9103	Communications-pac, two synchronous lines, cable
	DCM9104	Communications-pac, one synchronous lines, cable
	DCM9105	Communications-pac, one current loop line (Bell 301/303 modem compatible)
	DCM9106	Communications-pac, one medium speed, synchronous, HDLC line
	DCM9108	Communications-pac, broadband line (CCITT/V35 compatible)
	DCM9109	Communications-pac, two synchronous lines (MIL 188C compatible)
	DCM9110	Communications-pac, Autocall Unit for one or two synchronous or asynchronous lines
	DCM9111	Communications-pac, one asynchronous current loop line
	DCM9112	Communications-pac, broadband HDLC line (Bell 301/303 modem compatible to 72KB)
	DCM9113	Communications-pac, broadband HDCL line (CCITT/V35 compatible to 72KB)
	DCM9114	Communications-pac, two asynchronous current loop lines
	DCM9115	Communications-pac, broadband synchronous line (MIL 188C compatible to 56KB)
	DCM9116	Communications-pac, two asynchronous lines (MIL 188C compatible)
	DCM9120	Communications-pac, HDLC to 19 2KB MIL 188C
	DCM9121	Communications-pac broadband HDLC MIL 188C
	DCM9126	Communications-pac INCOTERM connection

Table C-2 (Cont). Hardware Supported - Model 3X, 4X and 5X

Category	Marketing Identifier	Description
Terminals	Asynchronous Terminals	
	VIP7100	CRT terminal, 64-character set
	VIP7105	CRT terminal, 96-character set
	VIP7200	CRT terminal, 64-character set
	VIP7205	CRT terminal, 96-character set
	VIP7300	CRT terminal, 12-inch screen
	VIP7802	CRT terminal, 15-inch screen
	VAF7821	Buffered printer adapter for all VIP 7800 terminals
	TWU1001	Keyboard/printer, 30 cps
	TWU1003	Keyboard/printer, 30 cps
	TWU1005	Keyboard/printer, 120 cps
	PRU1001	Printer terminal, 30 cps
	PRU1003	Printer terminal, 30 cps
	PRU1005	Printer terminal, 120 cps
	Synchronous Terminals	
	VIP7700	CRT terminal, 62-character set
	VIP7705	CRT terminal, 95-character set
	VIP7700R	CRT terminal, 63-character set
	VIP7705R	CRT terminal, 95-character set
	VIP7760 Subsystem	CRT terminal, 95-character set
	VIP7804	CRT terminal, 12-inch screen
	VIP7805	CRT terminal, 15-inch screen
	VAF7821	Buffered printer adapter for all VIP 7800 terminals
	ROP Printers	ROP printers are generally supported

Table C-2 (Cont). Hardware Supported - Model 3X, 4X and 5X

Category	Description
Modems	Modem supported for asynchronous communications terminals are: . 103, 113, 202 type modems. (These modems must be equiped with the option to disconnect the data set after a carrier drop of 110 milliseconds). . CII HONEYWELL BULL modem bypass. . Modem types where the connection, disconnection, and data set control settings can be user-specified. Modems supported for synchronous communications terminals are: . 201, 201A, 201C, 208, and 208B type modems. . CII HONEYWELL BULL modem bypass. . Modem types where the connection, disconnection, and data set control settings can be user-specified. . Broadband modem. A modem is not required for a direct-connect asynchronous terminal, or a synchronous terminal with a timing source in the terminal or in the MLCP.
System Building Products	Writable Control Store (CPF9509) is offered for system builder use only. Although Writable Control is usable under GCOS 6 MOD 400, full end-user software support is not available.

APPENDIX D

POWER RESUMPTION

NOTE: The Power Resumption facility is not supported on systems with an MLCP-connected operator's console.

The power resumption facility allows the system execution environment to be restarted after a power interruption. The central processor must have the memory save and autorestart unit. This unit can preserve the memory image through a power failure lasting up to two hours. It cannot, however, preserve the state of the I/O controllers, ensure that no operational changes have been made to the mounted volumes, nor preserve the contents of the multiline communication processors. Restoration of this system information is accomplished through the power resumption facility.

If fewer than two hours have elasped when power is returned to the central processor, the power resumption facility performs the following actions:

- Reinitializes the I/O controllers
- Reconnects terminal devices (TTY and ATD devices)
- Reestablishes the integrity of mounted volumes
- Restarts the communications subsystem (TTY and ATD devices only)
- Restarts the application tasks that were active at the time of the failure
- Signals a power resumption software trap to tasks that have a handler enabled for this trap
- Restart Display Formatting and Control processing

If the power remains off for more than two hours, the memory image is destroyed and the power resumption facility is disabled. The "memory on" indicator contained in the memory save and autorestart unit is lit. The operator must manually reset this indicator before rebootstrapping the system to perform a restart. See the Mini 6 Minicomputer Systems Handbook for detailed information about the memory save and autorestart unit. See the Operator's Guide for procedures used in restarting after a system failure.

POWER RESUMPTION CONFIGURATION REQUIREMENTS

To implement the power resumption capability, you must configure the following:

- Memory save and autorestart unit
- Power resumption facility
- Automatic terminal reconnect

Note that peripheral devices (those devices not attached to an MLCP) are designated automatically reconnectable when they are configured at system building. These devices are reconnected when power is restored.

Configuring Memory Save and Autorestart Unit

Pour small recker switches, located behind the full control panel on the control panel circuit board, supply configuration information to the central processor. The switch on the extreme left (when the panel is open) is the volatile memory switch. It must be set off (pushed down) to inform the system that the memory save and autorestart unit is present.

NOTE: On some control panels, the volatile memory switch is on the extreme right. See the Mini 6 Minicomputer Systems Handbook for complete details.

Configuring Power Resumption Facility

You add the power resumption facility to your system by including the following LDBU directive in the Configuration Load Manager (CLM) file.

```
LDBU ZXPFR
```

See Section V for a description of the LDBU directive.

Configuration Automatic Terminal Reconnect

Each communication device (each terminal connected to an MLCP) that is to be automatically reconnected by the power resumption facility must be so designated using the STTY command with the -RECONNECT argument in your CLM file. See the description of the STTY command below.

STTY COMMAND

Command Name: STTY

Change or display the file characteristics of a communications terminal. Can be used as an alternative to the STTY directive format in the Configuration Load Manager (CLM) file. Provides features not supported by the STTY directive. You must include an STTY command in your CLM file for each terminal that you wish to configure as being reconnectable in the event of a system power failure.

Format:

```
STTY [device_name] ctl_arg
```

Argument Description:

device_name

The 1- to 6-character device name of the terminal: use of the exclamation point character as a prefix is optional (e.g., TTY01 or !TTY01 can be used).

Default: The terminal from which this command is issued.

ctl_arg

One or more of the following control arguments must be specified, in any order. If an argument is unspecified the corresponding current value for the terminal remains in effect.

-LL n

Decimal integer specifying the desired line length. This value excludes the control byte.

-TYPE { I | O | B }

Set the device type to input-only (I), output-only (O), or bidirectional (B).

-TABSIM { YES | NO }

Insert space characters, where needed, to simulate tabbing. If the terminal does not support tabbing and the output data has tab characters, specify YES.

-IN $\begin{Bmatrix} A \\ S \\ N \end{Bmatrix}$

Prepare the device to receive asynchronous buffered (A), synchronous buffered (S), or nonbuffered synchronous (N) input. (See the "Function Description" for a description of asynchronous, synchronous, and buffered).

-OUT $\begin{Bmatrix} A \\ S \\ N \end{Bmatrix}$

Prepare the device to transmit asynchronous buffered (A), synchronous buffered (S), or nonbuffered synchronous (N) output. (See the "Function Description" for a description of asynchronous, synchronous, and buffered).

-RECONNECT $\begin{Bmatrix} YES \\ NO \end{Bmatrix}$

Specifies whether or not this terminal should be automatically reconnected following a power failure or line drop condition. If YES is specified, the terminal will be automatically reconnection. If NO is specified, no automatic reconnection will take place. The default is NO.

-PRINT

Display the current characteristics (name, device-type, line length, etc.) for the terminal.

-MODES mode ...

Set the terminal to operate in the specified mode(s). To reset a mode, precede the mode with a circumflex (^). For example, -MODES ECHO, ^HANG_UP sets the terminal to echo keyboard input and to "not" hang up the phone when the file is closed. Any number of modes can be set or reset in any combination, in any order. If a particular mode is not specified, that mode is not changed. Possible modes are listed below. To set a mode, you must know which line protocol handler (LPH) is servicing the terminal. The -PRINT argument can be used to display the current mode settings to determine which modes can be set or reset.

Mode	Meaning	LPH
ECHO	Echo keyboard input.	TTY,ATD(1)
{AUTO_CALL / AC}	Use the Auto-Call unit when the file is opened.	ALL
{TRANSPARENT / TR}	Input data is in transparent mode.	TTY,ATD,BSC
{STOP_OUTPUT / SO}	If an output order is in progress, stop it immediately if a BRK is issued from the terminal.	TTY,ATD
{LINE_FEED / LF}	Line feed at end of message.	TTY,ATD,VIP, STD
{CARRIAGE_RETURN / CR}	Carriage return at end of message.	TTY,ATD,VIP STD
{CONTROL_BYTE / CB}	Output sent to the terminal starts with a control byte.	TTY,ATD,VIP STD,BSC
{BLK_XFER / BLK}	Data transfer is in buffered (block) mode, not character mode.	TTY,ATD
{HANG_UP / HU}	Hang up the phone when the file is closed.	ALL
HOME	Set cursor to home on page overflow.	VIP,STD
FCTN	Support VIP function codes.	VIP,STD,PVE
DEL	Include received DEL characters.	PVE

The following special keywords can be used with the VIP line protocol handler to set various time-out and polling intervals:

Mode	Meaning	LPH
TO	Time-out immediately (after 1 poll).	VIP,STD
TO10	Time-out after 10 minutes of polling.	VIP,STD
^TO	No time-out (i.e., indefinite).	VIP,STD
^PI	No polling interval (i.e., continuous).	VIP,STD
PI1	1-second poll interval.	VIP,STD

Mode	Meaning	LPH
PI2	2-second poll interval.	VIP,STD
PI3	3-second poll interval.	VIP,STD
PI4	4-second poll interval.	VIP,STD
PI5	5-second poll interval.	VIP,STD
PI15	15-second poll interval.	VIP,STD
PI30	30-second poll interval.	VIP,STD

-RESET

Reset the modes to those designated at system building.

DSW1 h

Set the terminal's device-specific word to be used at open (connect) and close (disconnect) time; this argument is an alternative to using the -MODES argument. h is a 4-character hexadecimal value. This argument requires detailed knowledge of terminal I/O and should be used only if you understand line protocol handlers and have read the Communications Processing manual.

-DSW2 h

Set the terminal's device_specific word to be used when the file is read or written; this argument is an alternative to using the -MODES argument. h is a 4-character hexadecimal value. This argument requires detailed knowledge of terminal I/O and should be used only if you understand line protocol handlers and have read the Communications Processing manual.

NOTES:

1. Only the current device-specific word assignments can be modified; i.e., not the initial device-specified words established through the STTY directive during system building.

2. No consistency checks are made; if the -DSW1 or -DSW2 argument specifies logically inconsistent or impossible conditions, relative to the actual device type, no notification of error is given and the indicators are set as requested.

FUNCTION DESCRIPTION:

The set terminal characteristics (STTY) command permits you to modify or display the file characteristics associated with a terminal or endpoint that is not currently reserved. The original terminal characteristics, established during system building, can be altered to reflect your current needs. To use the -IN and -OUT arguments, you must understand the meanings of the terms buffered, asynchronous, and synchronous. These terms are described below.

Buffered means that an intermediate data storage area is used before data is transferred to/from a device. For input, data is received in system memory and then moved to used memory. For output, data is moved from used memory to system memory and then to the device. Nonbuffered means that data is transferred directly from user memory to the device.

Asnchronous data transfer is the concurrent transfer of data and execution of an application program; i.e., processing can continue during a read/write operation. Asynchronous operations are always buffered.

In synchronous data transfers, processing waits for completion of a read/write request. Control returns to the requestor after the transfer is complete. Synchronous operations may be buffered or nonbuffered.

The operating characteristics of the application program determine whether asynchronous buffered (A), synchronous buffered (S), or synchronous nonbuffered (N) are specified in the -IN and -OUT arguments. Generally, if an application program is directed towards multiple terminals, specify A so there is full user interaction. If S or N is specified and someone issues a read or write, all other users have to wait. This is explained in more detail below.

With asynchronous input, if an application program issues a read request, the system waits for previous read operations to complete. The data is moved to the application program, the read request just issued is placed in a queue, and control immediately returns to the application program. If the application program is interacting with only one terminal, asynchronous input is efficient, but not necessary: the application program can process a message while the file system reads the next message. Asynchronous input is necessary for an application program to interact with multiple terminals. For multiterminal applications, an application program may poll terminals and wait for input from a list of terminals.

Asynchronous output permits "double buffering" output requests. When an application program issues a write request, the system waits for previous write requests to complete. Data is moved from the application program to a system buffer, the write request is placed in a queue, and control immediately returns to the application program.

NOTE: For single-terminal applications using the roll-out feature, A or S provides more efficient operation.

Example:

```
STTY !VIP01 _MODES ^CONTROL_BYTE,^STOP_OUTPUT,
AUTO_CALL,^HANG_UP
```

or

```
STTY VIP01 -MODES ^CB,^SO,AC,^HU
```

File input/output to VIP01 will not expect a control byte before each output message. Finish printing output message that may be in progress when a BREAK is detected. Use the Auto-Call unit when file is opened. Do not hang up the phone when the file is closed.

ACTIONS FOLLOWING POWER RESUMPTION

When the power resumption facility restarts the execution environment, it automatically performs a number of system functions. When the operator and system users are notified that power resumption has occurred, they may be required to perform certain actions.

Automatic Functions

The power resumption facility automatically performs the following functions:

- Restarts the device drivers, clock, and communications subsystem.
- Reconnects peripheral devices. Reconnects terminal devices that were designated as being automatically reconnectable.
- Resets the system date and time. (The date/time clock has a separate battery backup unit: it is supported through the real-time adapter).
- Reloads the memory management unit.
- Restarts all tasks that were active when the power failure occurred.

In addition to these automatic functions, the system operator and system users may be required to perform certain actions. These power resumption procedures are described in the Operator's Guide.

APPENDIX E

DATA ENTRY FACILITY - I

This Appendix describes a typical configuration procedure that can be used to accommodate DEF-I, and list the Linker directives that are required to link a DEF-I system.

CONFIGURATION

GCOS 6 MOD 400 software requires that the following steps be performed before the Data Entry Facility-I can be used:

1. During system building, ensure that the directives required by the DEF-I system are included in the CLM_USER file.

2. Generate a DEF-I system Linker directive file that supports the user's CLM_USER file.

   NOTE: To aid in steps 1 and 2, a sample DEF-specific Linker directive file (DEF_SAMPLE.S) is supplied on the media. It may be modified to meet the requirements of the particular installation.

3. Execute a Linker run to produce a DEF-I bound unit.

4. Optionally, execute a Linker run to produce a user supplied date entry program.

5. Generate the appropriate DEF-I system user group and Start-Up task command files.

To aid in an understanding of this process, a sample DEF-I system configuration is used throughout this appendix.
Configuration parameters include:

- 4 operator display stations
- 1 line printer
- 1 data entry program overlay area
- 2 DEF-I background tasks

CLM_USER FILE

After the GCOS 6 MOD 400 system has been initially loaded by a stage I system startup, you must define a file of configuration directives, called the CLM_USER file, for the Configuration Load Manager (CLM). This file will contain the Symbolic Peripheral Device names (SPD), communications arguments, and memory pool assignments necessary to use GCOS 6 MOD 400 software on a Mini 6 system.

The CLM_USER file directives need not be completely DEF-specific. That is, the devices, memory, etc., described in the CLM_USER file may be shared by other system function facilities and may include specifications not required for the Data Entry Facility-I.

Sample CLM_USER File

Figure E-1 illustrates a CLM_USER file for a Data Entry Facility. The file specifies a system with four operator display stations, one attached through an MDC (CON01) and three attached through an MLCP/DLCP (TTX). Additionally, there are two diskettes and a line printer. (The bootstrap device is left implicit here).

```
DEVICE KSR00,0,5,X'0500',CONSOLE,140

COMM 7

TTX 12,8,X'FF00',0,9600

TTX 13,9,X'FF80',0,9600

DEVICE CON01,14,10,X'0580',CON01,80,N,'7200',T

TTX 15,11,X'FE80',0,9600

PVE 16,16,X'FC00',2,00

PVE 17,16,X'FC00',2,01

DEVICE DSK00,5,14,X'0480'

DEVICE DSK01,6,15,X'0480'

DEVICE LPT00,8,17,X'1380',LPT00

MEMPOOL S,,6000

MEMPOOL ,AB,*

SYS ,,,5,20,80,,E

QUIT
```

Figure E-1. Sample CLM_USER File

CLM_USER File Directives

The following CLM_USER file directives are used for a system that includes DEF-I.

DEVICE DIRECTIVE

A DEVICE directive is required for each device--except a communications (MLCP/DLCP-connected) operator display station--used with DEF-I. (A DEVICE directive may be paired with a TTX directive used for a communications operator display station if the device is to be accessible through the file system interface). The DEVICE directive is described in Sections V and VI; in addition, note the following points:

- A noncommunications (MDC-connected) operator display station requires the device arguments specifying non-buffered teletype mode of the DEF-I workstation be used (see Section V).

- If a DEVICE directive is paired with a TTX directive (described below), the DEVICE directive's device_unit argument value must be TTYnn (See Section VI to ascertain the device characteristics achieved by TTYnn). For example:

  DEVICE TTY00,15,11,X'FE80',TTY00,30

Table E-1 list the devices configured in Figure E-1 and their associated device_unit argument values.

Table E-1. Devices Configured in Figure E-1

Device Type	Number Configured	Device_Unit Argument Values
Console	1	KSR00
Operator Display Station--MDC-connected TTY	1	CON01
Diskette	2	DSK00,DSK01
Line Printer	1	LPT00

When more than one operator display station is configured for DEF-I, those CRT's must be assigned consecutive logical resource number (LRN's). Similarly, all system line printers used by DEF-I must be assigned consecutive LRN's.

In Figure E-1, the four operator display stations use the sequential LRN's 12, 13, 14, and 15.

Also note that the number of devices configured in the CLM_USER file may be greater than those specified when linking the DEF-I system (see below). Thus the DEF-I bound unit may be linked for three CRT's even if the CLM_USER file specifies four.

COMMUNICATIONS SYYSTEM DIRECTIVE

The COMM directive specifies that a communications subsystem is used. In Figure E-1, the communications subsystem is given a hardware interrupt level of 7. High priority hardware levels should be used for communications interrupts to ensure satisfactory operation.

TTX DIRECTIVE

The TTX directive specifies a TTY-type communications device. However, unlike the TTY directive, it provides an 80-character buffer. This buffer, and therefore the TTX directive, is required for all DEF-I operator display stations attached through an MLCP/DLCP.

The TTX directive appears as follows:

```
TTX lrn,level,X'channel',[modem][,speed]
```

Refer to the TTY directive in Section VI for a description of the TTX arguments.

In Figure E-1, there are three MLCP/DLCP connected operator display stations, and therefore three TTX directives.

MEMORY POOL DIRECTIVES

A MEMPOOL directive is used to specify the system memory pool size. Additionally, an online user memory pool--either exclusive or nonexclusive--must be specified for the DEF-I system. The minimum required size depends upon which DEF-I system functions are linked as resident and which are overlays, and upon the number of DEF-I workstation buffers required. Memory requirements are further described under "Linking a DEF-I System" below. The MEMPOOL directive is described in Section V.

In Figure E-1, the DEF-I system is configured with a single nonexclusive online memory pool with the name AB: this pool extends from the end of the system memory pool to high memory. This single pool can be used by the DEF-I system and by any other function in a group assigned to this pool, e.g., the COBOL compiler.

SYSTEM DEFINITION DIRECTIVE

The SYS (system definition) directive is described in Section V. The following consideration applies to DEF-I.

It is advisable to create several system overlay areas to improve the likelihood that a needed overlay is already present in memory (because it was recently used) and does not have to be reloaded. This is done through the "plan" argument of the SYS directive: 5 is a recommended value.

It is also recommended that the user allocate additional interrupt request blocks for each DEF-I workstation. In Figure E-1, 80 interrupt request blocks have been added for four workstations. This configuration supports activities in a high-throughout data entry environment. In a more casual data entry environment, approximately 10 additional interrupt request blocks per workstation would be adequate.

LINKING A DEF-I SYSTEM

The following paragraphs describe the requirements for linking:

- A DEF-I system bound unit
- A DEF-I data entry programs bound unit

The Linker directives are discussed in the Program Executive and Checkout manual.

DEF-I System Directories

The DEF-I system object units are located in the directories listed in Table E-2, depending upon the system software medium.

Table E-2. DEF-I Object Unit Directory Pathname

Medium	Directory Pathname
Diskette	^ZSYS10>ZDRT
Cartridge Disk	^ZSYS51>ZDRT
Mass Storage Unit	^ZSYS71>ZDRT
Cartridge Module Disk	^ZSYS61>ZDRT

DEF System Object Units

The DEF system object units are grouped in Table E-3 according to function. Some of these units are linked internally, and do not need to be specified when linking DEF-I.

Table E-3. DEF-I Object Units

	Object Units	
Function	Specified	Not Specified
System Object Units		
DEF-I Control	ZDFLTD, ZDFCTL ZDFIAD	ZDFCRT, ZDFINI, ZDFSTB, ZDFDDH, ZDFBM, ZDFPH, ZDFPRW, ZDFOLY, ZDFMSG, ZDFLTH, ZDFSRI, ZDFLOK, ZDFSRD
Background Control	ZDFBGR	
Operaton Security	ZDFOSR	
Operator Statistics	ZDFSTA	
Function Object Units		
Applications Processor	ZDFAP	
Data Entry Processor	ZDFDE1	ZDFDE2, ZDFDE3, ZDFDE4
Forms Processor	ZDFFM1	ZDFFM2, ZDFFM3, ZDFFM4, ZDFFM5
File Print Processor	ZDFFP	
Tables Processor	ZDFTB	
Utility Functions	ZDFUT	

Table E-2 (Cont). DEF_I Object Units

Function	Specified	Not Specified
Supervisor Functions	ZDFSP1, ZDFSP2, (ZDFPUP)	
Verification Processor	ZDFVE	

Additionally, dummy units listed in Table E-4 are provided so that specific DEF-I functions such as operator security, operator statistics, and password updating need not be included in the system.

Table E-4. DEF-I Dummy Object Units

Function	Dummy Object Unit
No Applications	ZDFAPD
No Data Entry	ZDFDED
No Form Creation	ZDFFMD
No File Printing	ZDFFPD
No Table Creation	ZDFTBD
No Utilities	ZDFUTD
No Supervisor	ZDFSPD
No Verification	ZDFVED
No Operator Security	ZDFOSD
No Operator Statistics	ZDFSTD
No Password Updating	ZDFPUD

Not required if supervisor is nor configured).

To create the DEF-I system bound unit, either the real unit or the corresponding dummy unit must be included for each function. For example, you must link either ZDFFP or ZDFFPF.

Data Entry Program Object Units

All data entry programs are optional user-supplied programs that are linked into one or more bound units separate from the DEF-I system bound unit. Table E-5 lists the object units that must be linked in order to process data entry programs. These object units are linked with the data entry programs in a Linker run separate from the linking of the DEF-I bound unit itself.

Table E-5. DEF-I Data Entry Program Object Units

Function	Object unit
Memory Resident Data Entry Program Directory	ZDFMR
Data Entry Program Overlay Directors	ZDFD1, ZDFD2, ZDFD3, ZDFD4
Data Entry Program Dope Vector Generator	ZDFDVG

Bound Unit Organization Considerations

Before generating a DEF-I bound unit, you should consider the following questions:

- Which DEF-I functions are required?
- Which DEF-I functions may be overlayed?
- Is the operator security feature required?
- Is password updating required?
- Are operator statistics required?

The available DEF-I functions are as follows:

- Applications Processing
- Data Entry Processing
- Forms Generation
- File Printing
- Table Generation
- Utility Functions
- Supervisor Functions
- Verification Processor

If Data Entry is configured you must also consider whether user-supplied data entry programs are required.

RESIDENT AND OVERLAYED FUNCTIONS

Each DEF-I function may be linked as memory resident or overlayed. Functions linked as overlays are not necessarily available to the display station station operator at all times. Figure E-2 illustrates three possible DEF-I system bound unit organizations. The first one is the bound unit as linked by DEF_SAMPLE.S in Figure E-7. Here, only the DEF-I control function is resident and the remaining functions are overleys. Functions which share a common overlay base address cannot be used concurrently. That is, it would not be possible to use the data entry and forms functions at the same time, but it is possible to use the data entry and verification functions concurrently. It would not be possible to use the interactive and supervisor functions at the same time because the data entry function is loaded into memory when the verification function is selected and the data entry and supervisor functions have a common base address. Note that the effect of linking the application function so that it is never overlayed by any other function would be to leave it always in memory once it or any of the other functions has been used. This has the same effect as linking the applications function as memory resident. The size of the particular DEF-I system bound unit would include the sum of the sizes of the DEF-I control, applications, data entry, and verification functions.

The second bound unit organization in Figure E-2 allows more functions to be accessed concurrently. Here it would be possible to use the applications, file print, data entry, forms, and tables functions all at the same time. In this particular organization, the size of the bound unit is greater than the size of the first illustration but the difference is only by the size of the file print function.

The third DEF-I bound unit organization in Figure E-2 would take up considerably less memory but available functions are limited. This organization would be suitable once you have created all necessary forms and tables for data entry and had no further use for those or any other functions.

When planning the DEF-I function overlays, you must consider which functions are required to run concurrently in the system and arrange the overlayys accordingly. Use of concurrent functions causes the size of the DEF-I system bound unit to increase according to the size of the concurrent functions.

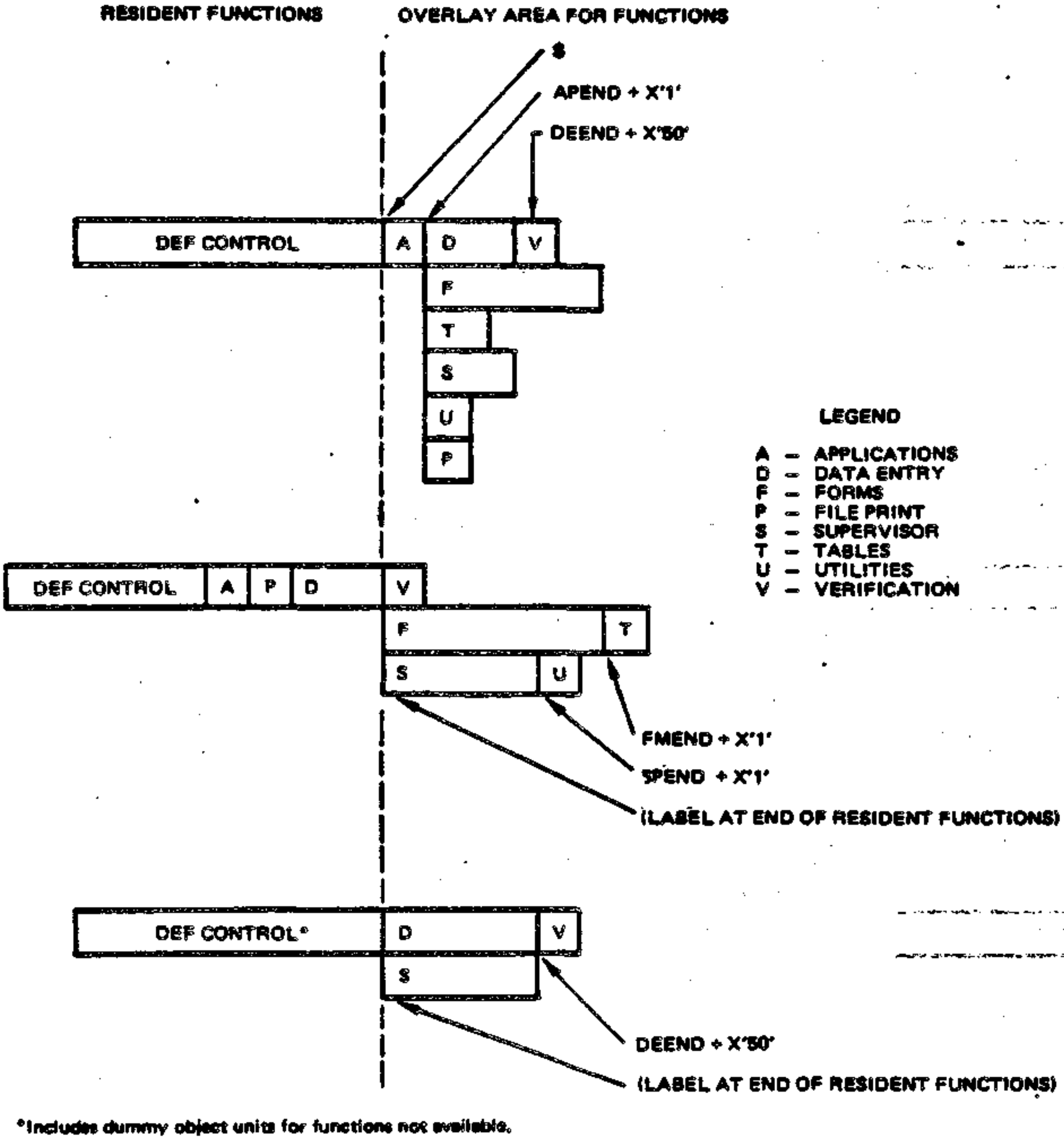
If a function is linked as an overlay but is not overlayed by any other function, it will be read into memory when first selected and remain in memory for the duration of the DEF-I task group.

DATA ENTRY PROGRAMS

All data entry programs are user-supplied and linked into one or more bound units separate from the main DEF-I system bound unit.

Data entry programs can be made memory resident for speed of access or overlayed if efficient memory is not available.

The linkage procedure for user programs is described in the Data Entry Facility-I User's Guide.



*Includes dummy object units for functions not available.

Figure E-2. DEF-I Function Resident/Overlay Organizations

MEMORY CONSIDERATIONS

The following memory considerations must be kept in mind when deciding the DEF-I system bound unit organization.

The amount of memory required for the DEF-I system bound unit includes the memory required for the resident functions plus the memory required for all overlayed functions that can reside in memory simultaneously. When all overlays start at the same base address, this memory size is that of the largest overlay.

When a given function is processing either as resident or as an overlay, all operator display stations can access that function concurrently (the function is reentrant).

User-written data entry programs are linked separately from the DEF-I system bound unit and have their own overlay structure. Up to four bound units of overlayed data entry programs are supported. The user may link up to 99 different data entry overlay programs in each bound unit or overlay area. The amount of memory available must be large enough to contain the largest data entry overlay program. User-written data entry programs can also be linked as memory resident. If no, they are linked into a single, separate bound unit. Memory is required for the data entry program director plus the sum of the sizes of all user programs up to a maximum of 20 programs.

Additional buffer and system memory considerations are discussed later in this Appendix.

DEF-I System Bound Unit Creation

Creation of the DEF-I system bound unit consists of a Linker run using a file with Linker directives that define four logical system categories:

- DEF-I system object units
- DEF-I memory resident DEF-I function object units
- DEF-I overlayed DEF-I function object units
- DEF-I system parameters.

Figure E-7 is a complete listing of sample Linker directives required by a DEF-I system. In the linking procedure described below, the directives in that listing are used to illustrate the text.

LINKING DEF-I SYSTEM OBJECT UNITS

Only the DEF-I system object units listed in Table E-5 must be linked. The remaining system object units listed in E-3 are internally linked and do not need to be specified.

Table E-6. System Object Unit Linker Directives

Function	Linker Directives	
System Routines: DEF-I Control and Background control	LINKN ZDFCTL, ZDFBGR, ZDFLTD, ZDFIAD	
	Configured	Not Configured
Operator Security	LINKN ZDFOSR	LINKN ZDFOSD
Operator Statistics	LINKN ZDFSTA	LINKN ZDFSTD

LINKING MEMORY RESIDENT DEF-I FUNCTION OBJECT UNITS

If a configured DEF-I function is to be memory resident or if it is to be omitted from the DEF-I system, a Linker directive listed in Table E-7 must be used.

Table E-7. Memory Resident Object Unit Linker Directives(1)

Function	Memory Resident	Not Configured
Applications Processor	LINKN ZDFAP	LINKN ZDFAPD
Data Entry Processor	LINKN ZDFDE1	LINKN ZDFDED
Forms Processor	LINKN ZDFFM1	LINKN ZDFFMD
File Print Processor	LINKN ZDFFP	LINKN ZDFFPD
Tables Processor	LINKN ZDFTB	LINKN ZDFTBD
Utility Functions	LINKN ZDFUT	LINKN ZDFUTD
Supervisor Functions	LINKN ZDFSP1, ZDFPUP, ZDFSP2 (See note 2)	LINKN ZDFSPD ZDFPUD
Verification Processor	LINKN ZDFVE	LINKN ZDFVED

NOTES:

1. Several of the object units listed in Tables E-3 and E-4 for the functions in this table do not have to be specified in the LINKN directives because they are internally linked.

2. If the supervisor function is linked and password updating is not required, specify ZDFPUD in place of ZDFPUP.

LINKING DEF-I FUNCTION OVERLAY OBJECT UNITS

Linker directives are required to link overlayed DEF-I functions, if any. For each function, use Linker directives to:

1. Name the overlay
2. Specify the base
3. Link the function object units
4. Protect the overlay base and end addresses.

Table E-8 lists the DEF-I system functions and the directives required to link each as an overlay.

Table E-8. DEF-I Function Overlay Linker Directives

Function	Overlay Directives
Application Processor	OVLY APOLAY BASE --- LINKN ZDFAP PROT APBASE,APEND
Data Entry	OVLY DEOLAY BASE --- LINKN ZDFDE1 PROT DEBASE,DEEND
Forms Processor	OVLY FMOLAY BASE --- LINKN ZDFFM1 PROT FMBASE,FMEND
File Print	OVLY FPOLAY BASE --- LINKN ZDFFP PROT FPBASE, FPEND
Supervisor Functions	OVLY SPOLAY BASE --- LINKN ZDFSP1,ZDFPUP,ZDFSP2 PROT SPBASE,SPEND

Table E-8 (Cont). DEF-I Function Overlay Linker Directives

Function	Overlay Directives
Table Generation	OVLY TBOLAY BASE --- LINK ZDFTB PROT TBBASE,TBEND
Utility Functions	OVLY UTOLAY BASE --- LINKN ZDFUT PROT UTBASE,UTEND
Verification(1)	OVLY VEOLAY BASE --- LINKN ZDFVE PROT VEBASE,VEEND

NOTE:

1. The verification function requires data entry processor subroutines. Therefore, its base starting address must follow the last data entry overlay address, DEEND.

The BASE directive must be set up according to the user required overlay structure. Refer to the <u>Program Execution and Checkout</u> manual for a full discussion of the BASE directive.

If patches are applied to any DEF-I function object unit and the end address of that function is used as the base address for overlayed DEF-I functions, then the base address for that overlayed function should be specified as:

[function id] END + X'50'

where [function id] is one of the two-character abbreviations shown above.

Each overlay has "start" and "end" labels which must be "PROTected" as shown.

Figure E-7 is a sample listing of the Linker directives necessary to specify the DEF-I system and function object units.

DEFINING DEF-I SYSTEM PARAMETERS

Additional Linker directives are required to provide DEF-I system parameter values. The values must be placed in the half of a word as illustrated.

A complete listing of the examples that illustrate the text is in Figure E-7.

Specifying LRNs

The following directives specify sample logical resource numbers (LRNs) required for DEF-I. Unless otherwise mentioned these LRNs are not related to system-wide LRNs (e.g., those in the CLM_USER file).

Directive	Comment
VDEF DKLRN,X'0100'	/LRN OF DISK HANDLER=1
VDEF BMLRN,X'0200'	/LRN OF BUFFER MANAGER=2
VDEF LOKLRN,X'0300'	/LRN OF DEF FILE LOCKER=3
VDEF INILRN,X'0500'	/LRN OF CRT1=5
VDEF LOLRN,X'0019'	/FIRST(=25) AND LAST (=X'1A'=26)LRNS
VDEF HILRN,X'001A'	/FOR BACKGROUND AND OPERATIONS. 2 OPERATIONS ARE ALLOWED IN THIS EXAMPLE.
VDEF CRLRN1,X'0'	/LRN OF MEMORY RESIDENT PROGRAM DIRECTOR =0 (NO MEMORY RESIDENT PROGRAMS)
VDEF OVLRN1,X'1B00'	/LRN OF DATA ENTRY PROGRAM OVERLAY DIRECTOR 1 =X'1B'=27 DECIMAL
VDEF OVLRN2,X'0'	/LRN OF DATA ENTRY PROGRAM OVERLAY DIRECTOR 2=0 (OVERLAY AREA NOT USED)
VDEF OVLRN3,X'0'	/LRN OF DATA ENTRY PROGRAM OVERLAY DIRECTOR 3=0
VDEF OVLRN4,X'0'	/LRN OF DATA ENTRY PROGRAM OVERLAY DIRECTOR 4=0

Note that data entry overlay directors 2 through 4 are given LRNs of 0 because they are not being used.

The LOLRN, HILRN arguments define the range of LRNs to execute file prints and applications in the background. In the example above, two LRNs are available which would allow two background file prints, two background applications, or one of each.

Specifying the Number of Function Overlays

The following directive specifies the number of OVLY directives used in linking the DEF-I functions.

Directive	Comment
VDEF OLYNO,X'0008'	/Number of function overlays = 8

Specifying the Size of the CRT Screen

The following directives specify the dimensions of the screen of the operator display station. The forms processor will not permit fields on a form to be created outside of the positions specified here. The size of the CRT screen must be specified as follows:

Directive	Comment
VDEF CRTC,X'0050'	/COLS PER CRT LINE X'50'=80 DECIMAL
VDEF CRTL,X'0017'	/LINES PER CRT PAGE X'17'=13 DECIMAL

Specifying Continuous or Noncontinuous Keyin

The following directive specifies whether keying is continous: that is, whether the system will tab forward automatically between fields or whether the RETURN key must be entered to reach the next field.

Directive	Comment
VDEF NCONKN,X'0'	/X'0' MEANS CONTINUOUS, NON X'0' MEANS NONCONTINUOUS KEYIN.

AUTCAL Argument

This argument is required and is reserved for system use:

VDEF AUTCAL,X'0000'	/RESERVED

CLM_USER-Related Directives

The following directives are specified in the CLM_USER file. They define the following:

1. The LRN of the first DEF-I CRT. The value must match the lowest LRN value specified in the CLM_USER file for a device to be used as an operator display station. In the example, the X'0C00' corresponds to the (decimal) 12 in Figure E-1.

Directive	Comment
VDEF CRTLRN,X'0C00'	/LRN OF FIRST CRT DRIVER = 12 DECIMAL

2. The LRN of the first printer used by DEF-I. The value must match the lowest LRN value specified in the CLM_USER file for a printer to be used with DEF-I. In the example, the X'0800' corresponds to the (decimal) 8 in Figure E-1.

Directive	Comment
VDEF PRTLRN,X'0800'	/LRN OF FIRST PRINTER DRIVER

3. The presence or absence of an operator console. A value of X'0059' (ASCII Y) in the VDEF OPCONS directive indicates that a console is present; a value of X'004E' (ASCII N) indicates that there is no console.

Directive	Comment
VDEF OPCONS,X'0059'	/ASCII, Y, OPERATOR'S CONSOLE IS PRESENT

4. The number of CRT's available as DEF-I operator display stations. Note that this number must not exceed the number of CRT's specified in the CLM_USER file.

Directive	Comment
VDEF CRTNO,X'0004'	/NUMBER OF CRT'S SUPPORTED

5. The number of printers available to DEF-I.

Directive	Comment
VDEF PRTNO,X'0001'	/NUMBER OF PRINTERS=1

Printer Formatting

The following directives define the number of characters per line and the number of lines per page for each printer. Note that 16 directives must be present, even if only one printer is used.

Directive	Comment
VDEF PRTC1,X'0088'	/PRINTER 1 LINE WIDTH=136 DECIMAL characters (Hexadecimal 88)
VDEF PRTL1,X'003E'	/PRINTER 1 LINES/PAGE=62 DECIMAL (Hexadecimal 3E)
VDEF PRTC2,X'0088'	/PRINTER 2 LINE WIDTH=136
VDEF PRTL2,X'003E'	/PRINTER 2 LINES/PAGE=62
VDEF PRTC3,X'0088'	/PRINTER 3 LINE WIDTH=136
VDEF PRTL3,X'003E'	/PRINTER 3 LINES/PAGE=62
VDEF PRTC4,X'0088'	/PRINTER 4 LINE WIDTH=136
VDEF PRTL4,X'003E'	/PRINTER 4 LINES/PAGE=62
VDEF PRTC5,X'0088'	/PRINTER 5 LINE WIDTH=136
VDEF PRTL5,X'003E'	/PRINTER 5 LINES/PAGE=62
VDEF PRTC6,X'0088'	/PRINTER 6 LINE WIDTH=136
VDEF PRTL6,X'003E'	/PRINTER 6 LINES/PAGE=62
VDEF PRTC7,X'0088'	/PRINTER 7 LINE WIDTH=136
VDEF PRTL7,X'003E'	/PRINTER 7 LINES/PAGE=62
VDEF PRTC8,X'0088'	/PRINTER 8 LINE WIDTH=136
VDEF PRTL8,X'003E'	/PRINTER 8 LINES/PAGE=62

Fixed Buffer Allocation

Each active operator display station (CRT) requires a certain amount of workspace. For example, each active CRT requires a constant memory storage area of 357 words. The remaining workspace requirements depend upon the function executing at the CRT and are provided by a set of fixed and dynamic buffers. Dynamic buffers are requested as they are needed from the user pool. Fixed buffers must be specified using Linker directives. There are four sizes of DEF-I fixed buffers specified using the VDEF directive NB1, NB2, NB3, and NB5 arguments. These buffer sizes are listed in Table E-9.

Table E-9. DEF-I Fixed Buffer Sizes

Buffer Argument	Buffer Length (Words) Decimal
NB1	1024
NB2	256
NB3	Configurable
NB5	10
Buffer argument NB4 is reserved.	

Depending upon the particular application executing at the CRT some of the DEF-I fixed buffers are used only temporarily and are returned to the DEF-I fixed buffer pool as soon as their use is completed. The buffers may then be used by some other operator display station, thus reducing the number of buffers that must be linked.

Table E-10 shows the DEF-I fixed (F) and dynamic (D) buffer requirements for the DEF-I functions and indicates whether the buffers are temporarily held or are held as long as the function is being used. The figures given are for a given operator display station.

Table E-10. Fixed and Dynamic Buffer Requirements

	NB1 1024 Wds	NB2 256 Wds		NB3 Configurable		NB5 **
Function	F	F	D	F	D	10 Wds
DATA ENTRY	1*	1	1	1	1	As many as needed
DATA ENTRY WITH TABLES	1*	1	1	1	1	As many as needed
DATA ENTRY WITH PROGRAMS	1*	1	1	1	1	As many as needed
DATA ENTRY WITH TABLES/PROGRAMS	1*	1	1	1	1	As many as needed
DATA ENTRY - MODIFICATION	1*	1	1	1	2	As many as needed
DATA VERIFICATION	1*	1	1	1	2	As many as needed

Table E-10 (Cont). Fixed and Dynamic Buffer Requirements

	NB1 1024 Wds	NB2 256 Wds		NB3 Configurable		NB5**
FORMS CREATION	1	1	-	-	-	-
FORMS MODIFY	2	1	1	-	-	-
CREATE A VERIFY TABLE	1	-	-	-	-	-
CREATE AN EXTRACT TABLE	2	-	-	-	-	-
COPY A FORM OR TABLE	2	1	-	-	-	-
VIEW A FORM OR VERIFICATION TABLE	1	1	-	-	-	-
VIEW AN EXTRACT TABLE	2	1	-	-	-	-
FILE PRINTING (FORMATTED)	1*	1*	-	1*	-	As many as needed
PRINT A FORM OR TABLE	1	1	-	1	-	-
SHIFT F7 PRINT	1*	-	-	-	-	-

*Indicates that the buffers are used temporarily and then released.

**NB5 buffers are used for extended field (program calls, upper/lower limits, tables, constants) used on the form. There is one NB5 required per extended field used on a form.

General rules based upon the above table for linking NB1, NB2, NB3, and NB5 size buffers are as follows:

- For operator display stations used for data entry only use:

  1 NB1 per 2 operator display stations

  2 NB2 per operator display station

  3 NB3 per operator display station

  5 NB5 per operator display station per extended field on form

- For operator display stations used for development (form creation, tables, copies, etc.):

2 NB1 per operator display station

1 NB2 per operator display station

1 NB3 per operator display station

1 NB5 per operator display station per extended field on form

NOTE: Attempts to process with insufficient buffers linked produce unspecified results. In most cases, an INSUFFICIENT USER MEMORY message will appear on line 24 of the operator display station.

The following directives define the number of buffers allocated for each size and the size of buffer 3. Buffer 3 is used for the Data Entry buffer and contains the current data entry record. Therefore, the specified size must be the same as the buffer size specified when linking any data entry programs. Also, the number of data characters the buffer will contain is twice the buffer size in words less 16 characters for the header. (see the DEF-I User's Guide).

Directive	Comment
VDEF NB1,X'0004'	/NUMBER OF SIZE 1 BUFFERS=4
VDEF NB2,X'0004'	/NUMBER OF SIZE 2 BUFFERS=4
VDEF NB3,X'0004'	/NUMBER OF SIZE 3 BUFFERS=4
VDEF NB4,X'0000'	/NUMBER OF SIZE 4 BUFFERS=0
VDEF NB5,X'0028'	/NUMBER OF SIZE 5 BUFFERS=40 (hexadecimal 28)
VDEF SB3,X'0080'	/SIZE OF BUFFER3 = 128 (Hex 80) words=256 characters

Data Entry Program General Purpose Buffer

The following directive specifies the size of the general purpose buffer, buffer number 6, used by any data entry program. You must specify a buffer size as large as the largest general purpose buffer required by any data entry program that will execute on your system. The buffer size is specified in words, half the number of characters the buffer will hold. One buffer is used for each operator display station executing a data entry program.

Directive	Comment
VDEF DUPWRD,X'0020'	/SIZE OF DATA ENTRY PROGRAM /GENERAL PURPOSE BUFFER=32 /(hex 20) WORDS = 64 /CHARACTERS

First LRN

The following directive defines the first LFN to be used by DEF-I. This number affects the value of the maximum LFN argument specified in the spawn group command for the task group. In the following example, LFNs 1 to 8 are available for use by data entry programs. Note, however, that the DEF-I LFNs may also start at 1. In that case, the program LFNs would start following the last DEF-I LFN. DEF-I requires 8 LFNs per operator display station and 8 LFNs per background area.

```
VDEF CRTLFN,X'0009'          /FIRST CRT LFN=9
```

Number of Background Tasks

The following directive defines the number of background tasks to be run concurrently. It cannot exceed the number of CRT's nor can it exceed the total number of LRNs defined by the LOLRN and HILRN arguments. File printing and application programs optionally run in the background. File print needs a background task even when the print operation is not in the background. At least one background task must be specified.

```
VDEF BAKNO,X'0002'           /NUMBER OF BACKGROUND TASKS=2
```

Assigning Volume Name

Each of the following four directives defines two characters of the name of the volume which DEF-I will initially use for forms, tables, and data entry. The hexadecimal values for the appropriate ASCII characters are specified. Only eight characters are allowed, the first must be circumflex (hexadecimal 5E), the eighth must be hexadecimal 3E. Note that the volume assignments can be changed by a DEF-I supervisory function, but this link specified volume name must be present when DEF-I is loaded.

```
  Directive                   Comment

VDEF SVN1,X'5E44'           /SYSTEM VOLUME= ^D
VDEF SVN2,X'4546'           /              =EF
VDEF SVN3,X'5752'           /              =WR
VDEF SVN4,X'4B3E'           /              =K>
```

Specifying Password Parameters

The following directives define the size and total number of passwords available and one initial password used to gain entry to DEF-I. Note that the initial password is specified using five directives, one for each two characters, even if the password is less than 10 characters long. The excess characters must be specified as blanks (hex 20). The directives are only required if the operator security feature is linked. The passwords may be maintained by the password update facility.

Directive	Comment
VDEF PASSNO,X'0010'	/NUMBER OF PASSWORD AVAILABLE=16 (hex 10)
	/MUST BE 0 IF NO OPERATOR SECURITY
VDEF PASSIZ,X'0004'	/LENGTH OF PASSWORD=4. MAXIMUM
	/SIZE IS 10 CHARACTERS
VDEF PASS1,X'4445'	/INITIAL PASSWORD IS DEF3.2
VDEF PASS2,X'4633'	/(HEX 44, 45, 46, 33, 2E, 32=ASCII DEF3.2)
VDEF PASS3,X'2E32'	(HEX 2E,32=ASCII.2)
VDEF PASS4,X'2020'	/(HEX 20=BLANK)
VDEF PASS5,X'2020'	

Specifying Password Accessible Functions

The following directive specifies the functions which may be accessed using the first password specified above. If any additional passwords are required the initial password must be given access to the supervisor function. The functions available are specified by the first 9 bits of the 16 bit, 4 hex character argument. If the bit value for the associated function is zero, the function is not available; if the bit value is one, the function is available. Table E-11 lists the bits and their associated functions. Bit 0 is the high order bit.

Table E-11. Accessible Function Specification

Bit	Function
0	Applications
1	Data Entry
2	Forms
3	File Print
4	(Reserved)
5	Supervisor
6	Tables
7	Utilities
8	Verification

Thus, a directive specifying that all functions are available using the first password would be as follows. This directive content is recommended.

Directive	Comment
VDEF PASSFN,X'FFFF'	/ALL FUNCTIONS ARE AVAILABLE

Specifying Allowable Operator Functions

The following directives determine which DEF-I functions can be selected by the operator display station users. If X'0000' is entered for any of these directives, the corresponding DEF-I function cannot be used by any data entry operator. If the function is to be allowed, the hexadecimal value of the appropriate ASCII character must be specified, as illustrated in the example.

Directive	Comment
VDEF APPL,X'0041'	/ALLOW APPLICATIONS=A
VDEF DATA,X'0044'	/ALLOW DATA ENTRY=D
VDEF FORM,X'0046'	/ALLOW FORM CREATE=F
VDEF INTRA,X'0000'	/RESERVED
VDEF FILPRN,X'0050'	/ALLOW FILE PRINT=P
VDEF SUPER,X'0053'	/ALLOW SUPERVISOR FUNCTION=S
VDEF TABLE,X'0054'	/ALLOW TABLES CREATE=T
VDEF UTIL,X'0055'	/ALLOW UTILITIES=U
VDEF VERIF,X'0056'	/ALLOW VERIFICATION=V

Printer Assignments

The following directives determine the initial printer assignments for each CRT. All 20 assignments must be made, even though fewer than 20 CRT's are available. Printer assignments may be temporarily changed by a DEF-I Supervisory Function.

Directive	Comment
VDEF PCRT1,X'0001'	/CRT 1 USES PRINTER 1
VDEF PCRT2,X'0001'	/CRT 2 USES PRINTER 1
VDEF PCRT3,X'0001'	/CRT 3 USES PRINTER 1
VDEF PCRT4,X'0001'	/CRT 4 USES PRINTER 1
VDEF PCRT5,X'0001'	/CRT 5 USES PRINTER 1
VDEF PCRT6,X'0001'	/CRT 6 USES PRINTER 1
VDEF PCRT7,X'0001'	/CRT 7 USES PRINTER 1
VDEF PCRT8,X'0001'	/CRT 8 USES PRINTER 1
VDEF PCRT9,X'0001'	/CRT 9 USES PRINTER 1
VDEF PCRT10,X'0001'	/CRT 10 USES PRINTER 1
VDEF PCRT11,X'0001'	/CRT 11 USES PRINTER 1
VDEF PCRT12,X'0001'	/CRT 12 USES PRINTER 1

```
VDEF PCRT13,X'0001'          /CRT 13 USES PRINTER 1
VDEF PCRT14,X'0001'          /CRT 14 USES PRINTER 1
VDEF PCRT15,X'0001'          /CRT 15 USES PRINTER 1
VDEF PCRT16,X'0001'          /CRT 16 USES PRINTER 1
VDEF PCRT17,X'0001'          /CRT 17 USES PRINTER 1
VDEF PCRT18,X'0001'          /CRT 18 USES PRINTER 1
VDEF PCRT19,X'0001'          /CRT 19 USES PRINTER 1
VDEF PCRT20,X'0001'          /CRT 20 USES PRINTER 1
```

Data Entry Overlay Areas

The following directives define which of the data entry overlay areas is to be used by each operator display station for execution of overlayed data entry programs. The hexadecimal value is the number of the data entry overlay area to be used and must be in the range 0001 through 0004.

As with the printer assignments, the data entry overlay area assignments must be made for 20 CRT's, even though fewer are available. Note that these overlays are different from the DEF-I function overlay area.

Directive	Comment
VDEF OCRT1,X'0001'	/CRT 1 USES OVERLAY 1
VDEF OCRT2,X'0001'	/CRT 2 USES OVERLAY 1
VDEF OCRT3,X'0001'	/CRT 3 USES OVERLAY 1
VDEF OCRT4,X'0001'	/CRT 4 USES OVERLAY 1
VDEF OCRT5,X'0001'	/CRT 5 USES OVERLAY 1
VDEF OCRT6,X'0001'	/CRT 6 USES OVERLAY 1
VDEF OCRT7,X'0001'	/CRT 7 USES OVERLAY 1
VDEF OCRT8,X'0001'	/CRT 8 USES OVERLAY 1
VDEF OCRT9,X'0001'	/CRT 9 USES OVERLAY 1
VDEF OCRT10,X'0001'	/CRT 10 USES OVERLAY 1
VDEF OCRT11,X'0001'	/CRT 11 USES OVERLAY 1
VDEF OCRT12,X'0001'	/CRT 12 USES OVERLAY 1
VDEF OCRT13,X'0001'	/CRT 13 USES OVERLAY 1
VDEF OCRT14,X'0001'	/CRT 14 USES OVERLAY 1
VDEF OCRT15,X'0001'	/CRT 15 USES OVERLAY 1
VDEF OCRT16,X'0001'	/CRT 16 USES OVERLAY 1
VDEF OCRT17,X'0001'	/CRT 17 USES OVERLAY 1
VDEF OCRT18,X'0001'	/CRT 18 USES OVERLAY 1
VDEF OCRT19,X'0001'	/CRT 19 USES OVERLAY 1
VDEF OCRT20,X'0001'	/CRT 20 USES OVERLAY 1

Completing the Link

The link must be completed with the QUIT directive. The MAP directive is optional.

```
MAP
QUIT
```

GROUP GENERATION FOR A DEF-I SYSTEM

Use the SG (Spawn Group) command to spawn a task group in which DEF-I can operate. A complete description of the SG command appears in the Commands manual.

Task Group Considerations

The task group in which DEF-I is to execute requires the following specifications for the maximum LRN and LFN:

LRN - the value must be greater than the maximum LRN used in the

START_UP.EC file's CT (Create Task) commands.

LFN - the value must be equal to or greater than the result of the following algorithm:

base LFN specified in the Linker VDEF CRTLFN directive

+ 8 x (number of CRT's)

+ 8 x (number of background tasks allowed at one time)

+ 1 if operator statistics are required

+ the number of LFNs required by data entry programs and applications.

<u>Example of a Spawn Group Command</u>

For example, a typical DEF-I SG (Spawn Group) command might be as follows:

SG AA D.E.F. 0 !CONSOLE -LRN/28 -LFN 58 -POOL AB -WD DEFDIR

SG

The spawn group command

AA

The id assigned to this task group

D.E.F.

The user_id assigned

0

Relative priority level for this task group's lead task

!CONSOLE

Input path for commands and user input

-LRN 28

Highest logical resource number that will be referred to by any task in the task group

-LFN 58

Highest logical file number used by any task in the spawned task group

-POOL AB

Name of the memory pool from which all dynamic memory required by the task group is to be taken

-WD DEFDIR

The working directory to be used by the task group. The directory used should have a START_UP.EC file containing the CT (Create Task) command to activate DEF-I.

The LRN in the example is greater than the maximum LRN used in the START_UP.EC file's CT (Create Task) command below.

The LFN value of 58 is greater than the value determined by the following calculation:

LFN = 8x4 (terminals) + 8x2 (background tasks) + 9 (base LFN, specified in Linker VDEF CRTLFN directive) = 57

DEF START_UP.EC FILE (TASK GENERATION)

Once the SG (Spawn Group) command has been executed, several DEF-I tasks must be created and activated. Appropriate CT (Create Task) commands may be stored in a START_UP.EC file immediately subordinate to the initial working directory of the spawned task group or they may be specified individually at initialization time. START_UP.EC files are discussed in the System Concept manual.

Create Task

Tasks are created by CT (Create Task) commands as follows:

CT lrn rel_level ctl_arg

lrn is the logical resource number by which the task group refers to the task

rel_level is the priority level of the task

ctl_arg is either the control argument

-EFN root?symbolic-start-address

or

-SHARE lrn symbolic-start-address

where the -EFN argument includes the name of the bound unit root segment (here DEF-I) followed by a question mark, followed by the symbolic-start-address (entry point) within the root segment, DEF initialization (INFNTR).

The -SHARE control arguments specify the additional entry points which the DEF bound unit, and the LRN used is that associated with the initialization (INFNTR) entry point.

RELATIVE LEVEL REQUIREMENTS

The CT (create task) command priority level requirements for DEF-I tasks are shown in Table E-12.

Table E-12. DEF-I Relative Priority Level Requirements

Relative Level	Symbolic-Start-Address	Description
1	DSHTAS	DEF Disk Handler
2	DBENT	DEF Buffer Manager
3	LOCK01	DEF File Lock Routine
5	INFNTR	DEF Initialization/CRT #1 Control
6-n	CENTRY	DEF CRT #2 (and up) Control
n+1	(none)	DEF Background Functions
n+m+1	COBMR1 if memory resident COBOV1 is overlay area 1 COBOV2 is overlay area 2 COBOV3 is overlay area 3 COBOV4 is overlay area 4	Optional for Data Entry Programs

In this example CENTRY, background tasks, and data entry programs are not given relative level values because one CT (Create Task) command is required per CRT, background task, and bound unit for data entry programs. Note that a maximum of four overlay areas and one memory resident bound unit for data entry programs may be specified. Note also that these are minimum relative level requirements within DEF-I and do not reflect any other LRNs within the task group.

Sample START_UP.EC File

The commands listed in Figure E-3 are typical for a DEF-I START_UP.EC file. The filename is ^ZSYS71>DEFDIR>START_UP.EC.

```
CWD >DEFDIR
CT 5 5 -EPN DEF?INFNTR
CT 1 1 -SHARE 5 DSHTAS
CT 2 2 -SHARE 5 DBENT
CT 3 3 -SHARE 5 LOCKO1
CT 6 6 -SHARE 5 CENTRY
CT 7 7 -SHARE 5 CENTRY
CT 8 8 -SHARE 5 CENTRY
CT 25 25 -SHARE 5
CT 26 26 -SHARE 5
CT 27 27 -EPN ENTRY1?COBOV1
RDN
ETR 5
&Q
```

Figure E-3. Sample DEF-I START_UP.EC File

In this example, DEF-I has the following:

- 4 CRT's
- 2 background (file print/application) tasks.
- 1 data entry program overlay area named ENTRY1

The name of the DEF-I bound unit is DEF.

(1)
n - Last relative level used for CRT
m - Number of backgroud tasks created.

SAMPLE SYSTEM FILE STRUCTURE

Figure E-4 illustrates the path structure of the DEF-I related system files, as they are supplied on the release media (the directory root name ^ZSYS71 indicates a mass storage unit). Included are the sample CLM_USER file, CLM_SAMPLE the sample DEF_I linker directive file DEF_SAMPLE.S, IEF_SAMPLE.S, and all DEF-I required object units.

Figure E-5 illustrates a typical path structure of DEF-I related system files once the DEF-I system has been built. The following files have been added to those shown in Figure E-4.

- The CLM_USER file (for example, the file in Figure E-1). This is placed in the directory SID.
- The DEF-I spawn group command contained in an EC file. The file is illustrated in Figure E-6. Note that this is directly under the system root.
- The DEF-I linker directives used to create the bound unit called DEF_USERS.S. This file is under the directory ZDRT.
- The DEF-I bound unit called DEF.
- The START_UP.EC that creates the DEF-I tasks is in the DEFDIR directory. An example of the startup file is illustrated in Figure E-3.
- The bound units for the DEF-I data entry programs.

The user-created DEF-I and date entry program bound units and the user-created DEF START_UP.EC are in a user-created directory DEFDIR.

Note that this set of files, naming conventions, and file paths is used as an illustration only.

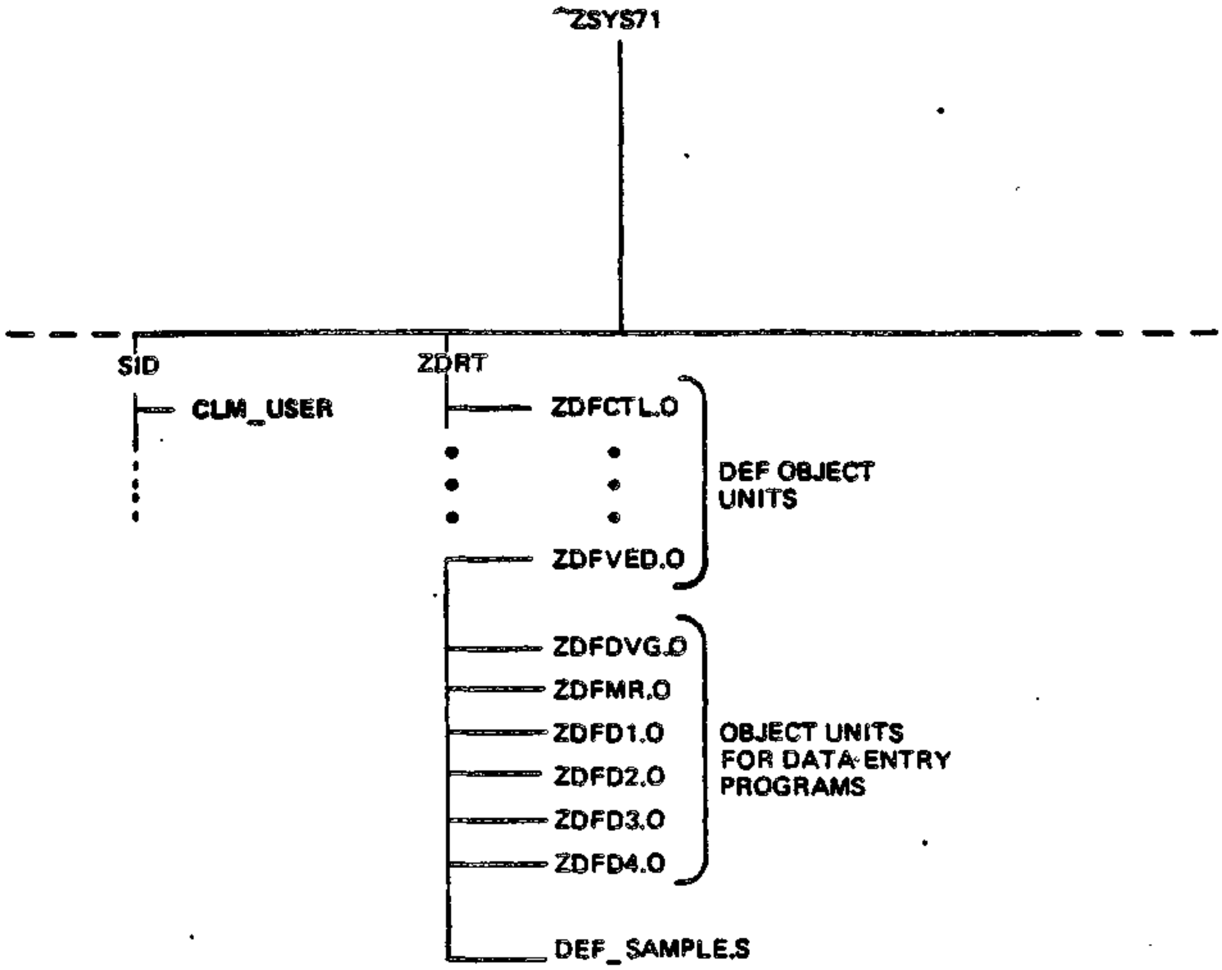
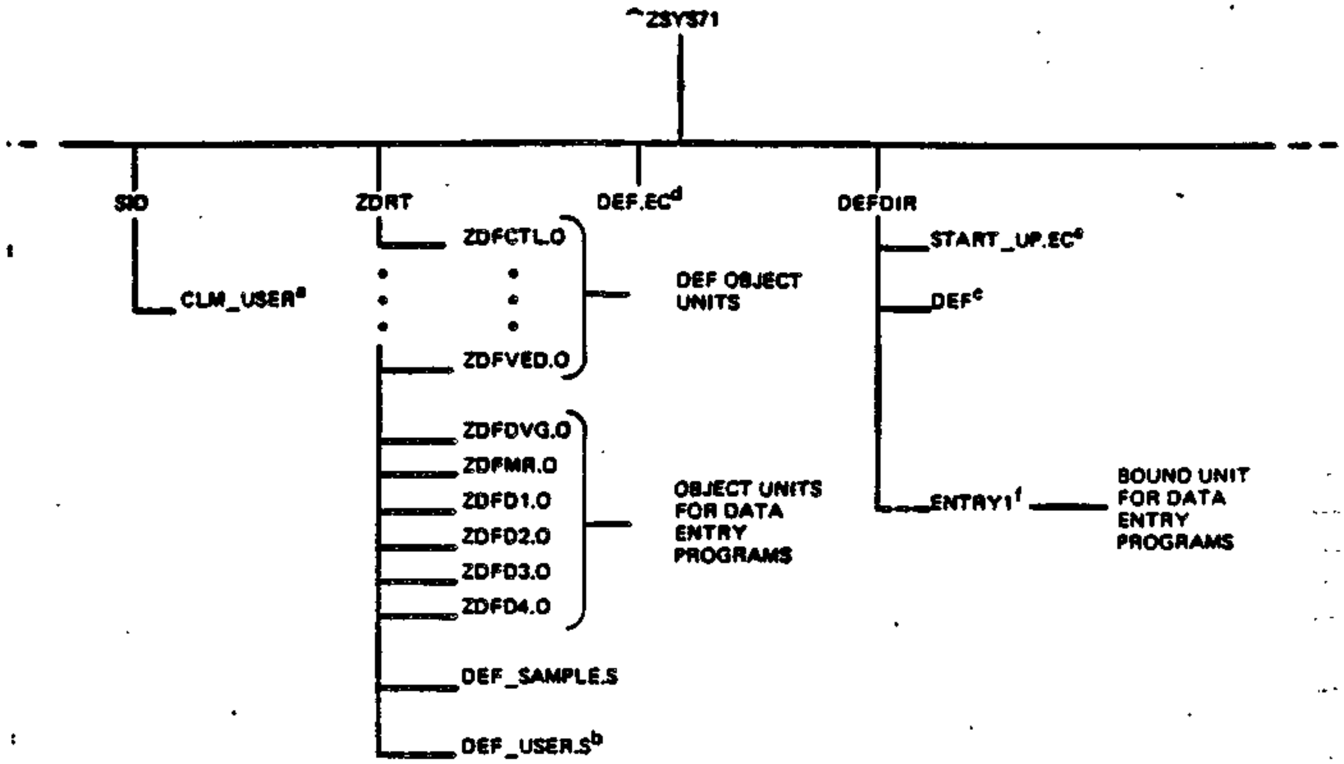


Figure E-4. Paths of System Files



[a]User-created CLM

[b]User-created LINKER directives for DEF bound unit

[c]DEF bound unit

[d]EC file that spawns DEF group in working directory DEFDIR

[e]START-UP.EC file that creates DEF tasks and issues group ready-on message

[f]User-created data entry program (see the Data Entry Facility User's Guide.)

Figure E-5. Sample Disk Organization of System and User DEF-I Files

```
&P *********************************
&P *  SAMPLE DEF GROUP GENERATION  *
&P *********************************
SG AA D.E.F. 0 >SPD>CONSOLE -LRN 28 -LFN 56 -POOL AH -WD >DEFDIR
&Q
```

Figure E-6. Sampe DEF-I SPAWN GROUP EC File

<u>Operator Startup Actions</u>

After all the preparatory procedures described in this section are completed and checked out, a DEF-I system can be started up by following a two-step procedure. Before presenting the procedure, the files that are required to be on disk are described:

- A CLM_USER file specific to DEF-I under the directory SID

- Bound units for the DEF-I system and user data entry programs. In the sample disk organization in Figure E-5, they are named DEF and ENTRY1.

- A START_UP.EC file in the working directory to create DEF-I tasks and start executing at the proper entry point. In figure E-5, the file START_Up.EC is in sample working directory DEFDIR.

- An EC file to spawn a group for DEF-I, for example, DEF.EC in Figure E-5.

The operator actions are then:

1. Bootstrap the system software

2. Enter the following command at the operator terminal:

```
EC DEF
```

A task group is created and the command processor loaded. The START_UP.EC files are then executed. They create and load the tasks of the DEF-I system, and then start execution of the initial DEF-I task, DEF?INFNTR, in the sample START_UP.EC file in Figure E-3.

SAMPLE LINKER DIRECTIVE FILES

Figure E-7 is a listing with supplied sample argument values, of all the Linker directives required to link a DEF-I system.

```
/ LINKER COMMANDS FOR DEF 3.2  (FOR NON-RESIDENT SYSTEM)
LIB >ZDRT                   / LIBRARY OF OBJECT MODULES
LINKN ZDFCTL                / DEF SYSTEM ROUTINES
LINKN ZDFIAD                / REQUIRED FOR DEF SYSTEM
LINKN ZDFLTD                / REQUIRED FOR DEF SYSTEM
LINKN ZDFOSH                / OPERATOR SECURITY
LINKN ZDFBGR                / BACKGROUND CONTROL
LINKN ZDFSTA                / OPERATOR STATISTICS
/                           / DEF FUNCTION OVERLAYS
OVLY APOLAY                 / APPLICATIONS OVERLAY
BASE $                      / STARTS AFTER END OF ROOT
LINKN ZDFAP                 / LINK APPLICATIONS
PROT APBASE,APEND           / PROTECT START/END ADDRESS
OVLY DEOLAY                 / OVERLAY DATA ENTRY
BASE APEND+X'1'             / START AFTER END OF APPLICATIONS
LINKN ZDFDE1                / LINK DATA ENTRY
PROT DEBASE,DEEND           / PROTECT START/END ADDRESS
OVLY FMOLAY                 / FORMS PROCESSOR OVERLAY
BASE APEND+X'1'             / STARTS AFTER END OF APPLICATIONS
LINKN ZDFFM1                / LINK FORMS PROCESSOR
PROT FMBASE,FMEND           / PROTECT START/END ADDRESS
OVLY FPOLAY                 / FILE PRINT PROCESSOR OVERLAY
BASE APEND+X'1'             / STARTS AFTER END OF APPLICATIONS
LINKN ZDFFP                 / LINK FILE PRINT PROCESSOR
PROT FPBASE,FPEND           / PROTECT START/END ADDRESS
OVLY SPOLAY                 / SUPERVISOR FUNCTIONS OVERLAY
BASE APEND+X'1'             / STARTS AFTER END OF APPLICATIONS
LINKN ZDFSP1,ZDFPUP,ZDFSP2  / LINK SUPERVISOR/PWRD UPDATE
PROT SPBASE,SPEND           / PROTECT START/END ADDRESS
OVLY TBOLAY                 / TABLES PROCESSOR OVERLAY
BASE APEND+X'1'             / STARTS AFTER END OF APPLICATIONS
LINKN ZDFTB                 / LINK TABLES PROCESSOR
PROT TBBASE,TBEND           / PROTECT START/END ADDRESS
OVLY UTOLAY                 / UTILITIES PROCESSOR OVERLAY
BASE APEND+X'1'             / STARTS AFTER END OF APPLICATIONS
LINKN ZDFUT                 / LINK UTILITIES PROCESSOR
PROT UTBASE,UTEND           / PROTECT START/END ADDRESS
OVLY VEOLAY                 / VERIFICATION PROCESSOR OVERLAY
BASE DEEND+X'50'            / STARTS AFTER DATA ENTRY
LINKN ZDFVE                 / LINK VERIFICATION PROCESSOR
PROT VEBASE,VEEND           / PROTECT START/END ADDRESS
/*************************************************************
/                           / DEF SYSTEM LRNS
VDEF DKLRN,X'0100'          / LRN OF DISK HANDLER
VDEF BMLRN,X'0200'          / LRN OF BUFFER MANAGER
VDEF LOKLRN,X'0300'         / LRN OF FILE LOCKER
VDEF INILRN,X'0500'         / LRN OF CRT 1 (INITIALIZATION LEVEL)
VDEF LOLRN,X'0019'          / LOW LRN AVAIL FOR IEF /BACKGROUND
VDEF HILRN,X'001A'          / HIGH LRN AVAIL FOR IEF/BACKGROUND
VDEF CRLRN1,X'0000'         / LRN OF D.E. MEM. RESIDENT PROGRAM DIRECTOR
VDEF OVLRN1,X'1800'         / LRN OF D.E. OVERLAY DIRECTOR 1
VDEF OVLRN2,X'0000'         / LRN OF D.E. OVERLAY DIRECTOR 2
VDEF OVLRN3,X'0000'         / LRN OF D.E. OVERLAY DIRECTOR 3
VDEF OVLRN4,X'0000'         / LRN OF D.E. OVERLAY DIRECTOR 4
/
VDEF OLYNO,X'8'             / NUMBER OF FUNCTIONS OVERLAYED
```

Figure E-7. Sample DEF-I Linker Directive File

```
VDEF CRTC,X'0050'         / COLS PER CRT LINE
VDEF CRTL,X'0017'         / LINES PER CRT PAGE
VDEF NCONKN,X'0'          / 0 = CONTINUOUS, NON 0 = NON CONTINUOUS KEY.IN
VDEF AUTCAL,X'0000'       / RESERVED
/                         / REFERENCES TO LRN'S IN CLM_USER
VDEF CRTLRN,X'0C00'       / LRN OF CRT 1  DRIVER
VDEF PRTLRN,X'0800'       / LRN OF PRINTER 1 DRIVER
VDEF OPCONS,X'59'         / 59 = OP. CONSOLE, 4E = NO OP. CONSOLE
VDEF CRTNO,X'0004'        / TOTAL NUMBER OF CRTS
VDEF PRTNO,X'0001'        / NUMBER OF PRINTERS CONFIGURED
/                         / PARAMETERS TO DEFINE PRINTERS
VDEF PRTC1,X'0088'        / PRINTER 1 LINE WIDTH
VDEF PRTL1,X'003E'        / PRINTER 1 LINES PER PAGE
VDEF PRTC2,X'0088'        / PRINTER 2 LINE WIDTH
VDEF PRTL2,X'003E'        / PRINTER 2 LINES PER PAGE
VDEF PRTC3,X'0088'        / PRINTER 3 LINE WIDTH
VDEF PRTL3,X'003E'        / PRINTER 3 LINES PER PAGE
VDEF PRTC4,X'0088'        / PRINTER 4 LINE WIDTH
VDEF PRTL4,X'003E'        / PRINTER 4 LINES PER PAGE
VDEF PRTC5,X'0088'        / PRINTER 5 LINE WIDTH
VDEF PRTL5,X'003E'        / PRINTER 5 LINES PER PAGE
VDEF PRTC6,X'0088'        / PRINTER 6 LINE WIDTH
VDEF PRTL6,X'003E'        / PRINTER 6 LINES PER PAGE
VDEF PRTC7,X'0088'        / PRINTER 7 LINE WIDTH
VDEF PRTL7,X'003E'        / PRINTER 7 LINES PER PAGE
VDEF PRTC8,X'0088'        / PRINTER 8 LINE WIDTH
VDEF PRTL8,X'003E'        / PRINTER 8 LINES PER PAGE
/                         / BUFFER ALLOCATIONS
VDEF NB1,X'0004'          / NO. OF SIZE 1 BUFFERS
VDEF NB2,X'0004'          / NO. OF SIZE 2 BUFFERS
VDEF SB3,X'0080'          / SIZE OF DATA ENTRY BUFFER
VDEF NB3,X'0004'          / NO. OF SIZE 3 BUFFERS
VDEF NB4,X'0000'          / NO. OF SIZE 4 BUFFERS
VDEF NB5,X'0028'          / NO. OF SIZE 5 BUFFERS
VDEF DUPWRD,X'0020'       / SIZE OF GENERAL PURPOSE BUFFER
/
VDEF CRTLFN,X'0009'       / FIRST LFN AVAILABLE TO DEF
VDEF BAKNO,X'0002'        / NUMBER OF BACKGROUND TASKS
/                         / DEF SYSTEM VOLUME NAME
VDEF SVN1,X'5E44'         / SYSTEM VOLUME IS :- ^D
VDEF SVN2,X'4546'         /                     EF
VDEF SVN3,X'5752'         /                     WR
VDEF SVN4,X'4B3E'         /                     K>
/                         / PARAMETERS TO DEFINE SYSTEM PASSWORDS
VDEF PASSNO,X'0010'       / NO. OF PASSWORDS (=0 IF NO OPERATOR SECURITY)
VDEF PASSIZ,X'0006'       / LENGTH OF PASSWORD
VDEF PASS1,X'4445'        / INITIAL PASSWORD = DEF3.2
VDEF PASS2,X'4633'
VDEF PASS3,X'2E32'
VDEF PASS4,X'2020'
VDEF PASS5,X'2020'
VDEF PASSFN,X'FFFF'       / FUNCTIONS FOR INITIAL PASSWORD
/                              BIT 0 = AP, BIT 1 = DE, BIT 2 = FM,
/                              BIT 3 = FP, BIT 4 = (RESERVED), BIT 5 = SP,
/                              BIT 6 = TB, BIT 7 = UT, BIT 8 = VE
/                         / FUNCTION LIST (0 = NOT CONFIGURED)
```

Figure E-7 (Cont). Sample DEF-I Linker Directive File

```
VDEF APPL,X'41'            / APPLICATIONS
VDEF DATA,X'44'            / DATA ENTRY
VDEF FORM,X'46'            / FORM CREATION
VDEF INTRA,X'00'           / RESERVED
VDEF FILPRN,X'50'          / FILE PRINT
VDEF SUPER,X'53'           / SUPERVISOR
VDEF TABLE,X'54'           / TABLES
VDEF UTIL,X'55'            / UTILITIES
VDEF VERIF,X'56'           / VERIFICATION
/                          / PRINTER ASSIGNMENTS FOR EACH CRT
VDEF PCRT1,X'0001'         / PRINTER ASSIGNMENT FOR CRT 1
VDEF PCRT2,X'0001'         / PRINTER ASSIGNMENT FOR CRT 2
VDEF PCRT3,X'0001'         / PRINTER ASSIGNMENT FOR CRT 3
VDEF PCRT4,X'0001'         / PRINTER ASSIGNMENT FOR CRT 4
VDEF PCRT5,X'0001'         / PRINTER ASSIGNMENT FOR CRT 5
VDEF PCRT6,X'0001'         / PRINTER ASSIGNMENT FOR CRT 6
VDEF PCRT7,X'0001'         / PRINTER ASSIGNMENT FOR CRT 7
VDEF PCRT8,X'0001'         / PRINTER ASSIGNMENT FOR CRT 8
VDEF PCRT9,X'0001'         / PRINTER ASSIGNMENT FOR CRT 9
VDEF PCRT10,X'0001'        / PRINTER ASSIGNMENT FOR CRT 10
VDEF PCRT11,X'0001'        / PRINTER ASSIGNMENT FOR CRT 11
VDEF PCRT12,X'0001'        / PRINTER ASSIGNMENT FOR CRT 12
VDEF PCRT13,X'0001'        / PRINTER ASSIGNMENT FOR CRT 13
VDEF PCRT14,X'0001'        / PRINTER ASSIGNMENT FOR CRT 14
VDEF PCRT15,X'0001'        / PRINTER ASSIGNMENT FOR CRT 15
VDEF PCRT16,X'0001'        / PRINTER ASSIGNMENT FOR CRT 16
VDEF PCRT17,X'0001'        / PRINTER ASSIGNMENT FOR CRT 17
VDEF PCRT18,X'0001'        / PRINTER ASSIGNMENT FOR CRT 18
VDEF PCRT19,X'0001'        / PRINTER ASSIGNMENT FOR CRT 19
VDEF PCRT20,X'0001'        / PRINTER ASSIGNMENT FOR CRT 20
/                          / OVERLAY DIRECTOR ASSIGNMENT PER CRT
VDEF OCRT1,X'0001'         / OVERLAY AREA FOR CRT 1
VDEF OCRT2,X'0001'         / OVERLAY AREA FOR CRT 2
VDEF OCRT3,X'0001'         / OVERLAY AREA FOR CRT 3
VDEF OCRT4,X'0001'         / OVERLAY AREA FOR CRT 4
VDEF OCRT5,X'0001'         / OVERLAY AREA FOR CRT 5
VDEF OCRT6,X'0001'         / OVERLAY AREA FOR CRT 6
VDEF OCRT7,X'0001'         / OVERLAY AREA FOR CRT 7
VDEF OCRT8,X'0001'         / OVERLAY AREA FOR CRT 8
VDEF OCRT9,X'0001'         / OVERLAY AREA FOR CRT 9
VDEF OCRT10,X'0001'        / OVERLAY AREA FOR CRT 10
VDEF OCRT11,X'0001'        / OVERLAY AREA FOR CRT 11
VDEF OCRT12,X'0001'        / OVERLAY AREA FOR CRT 12
VDEF OCRT13,X'0001'        / OVERLAY AREA FOR CRT 13
VDEF OCRT14,X'0001'        / OVERLAY AREA FOR CRT 14
VDEF OCRT15,X'0001'        / OVERLAY AREA FOR CRT 15
VDEF OCRT16,X'0001'        / OVERLAY AREA FOR CRT 16
VDEF OCRT17,X'0001'        / OVERLAY AREA FOR CRT 17
VDEF OCRT18,X'0001'        / OVERLAY AREA FOR CRT 18
VDEF OCRT19,X'0001'        / OVERLAY AREA FOR CRT 19
VDEF OCRT20,X'0001'        / OVERLAY AREA FOR CRT 20
MAP
QUIT
```

Figure E-7 (Cont). Sample DEF-I Linker Directive File

APPENDIX F

REMOTE BATCH FACILITY/66

CONFIGURATION

After the GCOS 6 MOD 400 software has been initially loaded by a stage 1 system startup, you must define a file of input directives, called the CLM_USER file, for the Configuration Load Manager (CLM). This file will contain the device symbolic names, communications arguments, and memory pool assignments necessary to use GCOS 6 MOD 400 software on your hardware system.

Any future changes to the Remote Batch facility configuration are made by editing the CLM_USER directive file to reflect these changes.

Configuration Directives

To create a Remote Batch Facility with Remote Computer Interface (RCI), you must specialize your GCOS 6 MOD 400 system software by providing the necessary configuration directives.

NOTE: Only information specifically required to create an RCI Remote Batch Facility is provided. Full descriptions of configuration directives are given in previous sections of this manual.

DIRECTIVES APPLICABLE TO RCI

SYS 60,100,,10,tsa,irb

The plan argument must be >2.

Suggested values for tsa:

30 for RCI

Suggested values for irb:

60 for RCI

COMM 8

This directive is used in the normal manner; i.e., it must precede the LPHn directives and specify the communications interrupt level. Both the communications interrupt level and the normal mode level should be the highest priority hardware levels in the system to ensure satisfactory operation.

Recommended levels are 7 or 8 in the COMM directive and 8 or 9 in the second argument of all LPHn directives (see below).

ACU lrn,level,X'acu_channel'[,'phone_#1'][,'phone_#2'][,...]

This directive is used in the normal manner to configure the optional Auto Call Unit for automatic dialing see section VI.

MEMPOOL S,,size

The Remote Batch Facility requires a system memory pool size of 9200 words.

For example, if your configuration requires 5500 words for system functions and 9200 words for the Remote Batch Facility, you would specify:

MEMPOOL S,,14700

LDBU ZERRST

You must include this directive in your CLM file if you wish to collect peripheral device or memory error statistics. Error logging configuration is fully described elsewhere in this manual. Error logging procedures are described in the <u>Operator's Guide</u>.

Example of Remote Batch Configuration Directives

The following directives give configuration information for a Remote Batch Facility having four RCI streams.

```
        .
        .
        .
MEMPOOL S,,16000
MEMPOOL ,AB,*
COMM 8
RCI   32,10,X'FC00'
DEVICE  RCI00,32,10,X'FC00',LINE32
RCI   33,10,X'FC80'
DEVICE  RCI01,33,10,X'FC80',LINE33
LDBU ZERRST
        .
        .
        .
QUIT
```

NOTE: The bound unit ZQRCI must reside in directory SID of the root directory. The LDBU ZERRST directive allows the user to configure error logging, as described in Appendix P. Using this CLM file, the user may designate his host link by one of the following:

```
RBT 32
RBT !LINE32
RBT >SPD>LINE32
```

INITIALIZATION

Once configuration is complete, you can execute an RBT in the task group $H or create your own task group. If multiple terminals are used, there must be a task group for each terminal. The different methods that can be used to create a task group are described below. Whether you use $H or create your own group, you must also assign devices/files, modify external switches if necessary, and invoke the task group.

Using the $H Task Group

To use the CII HONEYWELL BULL $H task group, enter:

EC GROUP$H(C/R)

The command file creates the task group.

Initializing with the Spawn Group Command

Use the Spawn Group (SG) command to create your own task group in which the remote batch terminal can operate. A complete description of the SG command can be found in the Commands manual.

The following is an example of the SG command used to create an RBT task group:

```
SG A1 SMITH.RBT.OPR 2 !CONSOLE -OUT !CONSOLE -POOL AB
     -WD RBT_DIR -LRN 2
```

SG

The Spawn Group command.

A1

The id assigned to this task group.

SMITH.RBT.OPR

The user_id assigned

2

Relative priority level for this task group: each RBT task group you spawn must be assigned a unique priority level.

!CONSOLE

Input path for commands and user input.

-OUT !CONSOLE

File that is to receive user output and error output.

-POOL AB

Name of the memory pool from which all dynamic memory required by the task group is to be taken.

-WD RBT_DIR

The working directory to be used by the task group. The directory used should have a START_UP.EC file containing the Ready On (RDN) command. This command causes the system to issue a ready message when the task group has been created.

-LRN 2

Two Logical Resource Numbers (LRN) are needed by RBT. This number should increase if there are any other requirements for this task group.

MAKING INITIAL DEVICE ASSIGNMENTS

Once the task group has been created, the input and output files are assigned to devices by using the ASSOCIATE command. The format for the command is:

```
                    ASSOC lfn path_name (C/R)
    lfn is:
            1 - input file
            2 - print file
            3 - punch file
```

The following is an example of the typeins:

```
ASSOC 1   !CDR00
ASSOC 2   !LPT00
ASSOC 3   !MT900>vol>file
```

The first line assigns the input file to card reader CDR00. The second line assigns the print file to line printer LPT00. The third line assigns the punch file to magnetic tape unit MT900. Vol and file must be given names by which they can be accessed.

MODIFYING EXTERNAL SWITCHES

The External Switch Word (ESW) should be set, if desired. The ESW should first be set to all zeros and then the desired bits turned on. The format for the typein is:

MSW -ALL OFF -ON nn...n(C/R)

MSW - Command to modify ESW

-ALL OFF - Turn off all ESW bits

-ON - Turn on the following bits

n - The bit (hexadecimal digit) that is to be turned on

The following is a breakdown of the bit in the External Switch Word:

Bit	On	Off
0	LIST	NOLIST
1	PUNCH	NOPUNCH
2	AUTOCALL	NOAUTOCALL
3	Reserved for system use	
4	STATS	NOSTATS
5	ASCII (READ)	GBCD
6	Reserved for system use	
8	LFN 3 out ASCII	LFN 3 out VBT
9, A, B, C, D, E, F	Reserved for system use	

Default values can be used for ESW settings: bits 0 and 1 will be set for LIST and PUNCH. The relationship between ESW settings and remote batch disposition codes is explained in the Remote Batch Facility User's Guide.

INVOKING THE RBT TASK GROUP

To invoke the RBT task group and associate it with a logical stream, enter the following command.

```
RBF lrn(C/R)                (Cartridge disk environment)
    or
^ZSYSII RBF lrn(C/R)        (Diskette environment)
    or
RBT device_name
```

The lrn value or device_name is taken from the configuration directive in the CLM_USER file. It specifies the logical stream you wish to use.

The system responds with:

```
(tg) hhmm RBF xnnn mm/dd/hhmm AA/BB/CC/DDDD
(tg) hhmm RBF RDY
(tg) hhmm RBF STATION NAME = 9999

            tg - Task group id
          hhmm - Hours, minutes
             x - S for SAF mode, L for LAF mode
           nnn - Software revision number
AA/BB/CC/DDDD - Data/Time identification of RBF
          9999 - NPS Station Name for LHDLC only.
```

The system is now ready to accept commands.

Initializing with a Command File

An alternate mathod of performing initialization is to build a command file with the system Editor and use the EC command for execution. The following steps are necessary:

1. Refer to the System Messages manual for a description of the Execution Command (EC) and the EC command arguments, and to the Program Preparation manual for instructions on the use of the Editor.

   An example of a command file to spawn a task group is:

   ```
   $ SP_GRP.EC           1980/06/08

   &N

   SG AI RBT.USER.OPR 2 !CONSOLE -OUT !CONSOLE
      -WD RBT_DIR -POOL AB
   $Q
   ```

   To execute this command, you must be operating in the system task group ($S) and type in:

EC SP_GRP(C/R)

The system responds with:

($S)SG AI RBT.USER.OPR ...
($S)RDY:

You can now either remain in the system default task group or change to a new task group by typing:

C :AI:(C/R)

Again, it is advisable to turn ready on by typing:

RDN(C/R)

The task group responds with:

(AI)RDY:

2. The second command file, as stated above, performs the ASSOC and MSW commands and invokes the RBT. An example of this command file is:

& RBT.EC 1980/06/08

&N

ASSOC 1 !CDR00

ASSOC 2 !LPT00

ASSOC 3 PUNCH_FILE

MSW -ALL OFF -ON 01 (Optional: defines default state)

RBT 03 (Cartridge disk: specify ^ZSYS!! RBT 03 for diskette)

&Q

The typein for this file is:

EC RBT(C/R)

The system responds by printing out the commands in the command file, and:

(tg) hhmm RBT xnnn mm/dd/hhmm
(tg) hhmm RBT RDY

You are now ready to perform remote batch processing.

3. If the system is to have multiple terminals, individual command files can be generated for the task groups and they

can be grouped into a single command file.

Example:

```
& SP_GRPS.EC
&N
EC SP_GRP1
EC SP_GRP2
EC SP_GRP3
$Q
```

By typing in:

```
EC SP_GRPS(C/R)
```

you can spawn three task groups with a single command, or, if desired, you can spawn the task groups individually.

APPENDIX G

FILE TRANSMISSION

This appendix described the Mini 6 configuration particulars that relate to file transmission in conjunction with the configuration information already presented in this manual. In this appendix, "TRAN" is a general term used to denote the various Mini 6 file transmission utility designations (TRAN and TRAN8).

LINE PROTOCOL CONFIGURATION

The VIP 7700 line protocol is utilized in file transmission between the Mini 6 and CII HONEYWELL BULL host systems. Through the use of the PVE (polled VIP Emulator) line protocol handler, the Mini 6 appears to the host system as a VIP 7700 terminal. The PVE is configured using the PVE directive as described in Section VI of this manual.

The following argument specifications of the PVE directive must be noted in configuring a Mini 6 for file transmission:

- The poll address specified in the directive must match that configured in the host system for the desired communications line.
- High-speed lines should be configured at the highest priority levels (lowest numeric); i.e., line speeds of 2400 and 9600 baud could be configured at priority levels of 20 and 10, respectively, but not 10 and 20. The maximum line speed supported in the file transmission facility is 9600 baud.
- The disk device containing the files to be transmitted or received must be configured at a lower priority level (higher numeric) than the communications device.
- The lrn selected for the directive is the argument used in the TRAN invocation for file transmission (see the appropriate File Transmission manual). If a DEVICE directive is coupled with a PVE directive, a filename may be used in the TRAN invocation as an alternative to a numeric lrn.

In a Mini 6 to Mini 6 file transmission configuration, the Mini 6 designated as the host system (acceptor) is configured with a VIP (see Section VI) for the communications line, while the Mini 6 acting as the remote station (initiator) is configured with a PVE for the same line. Again, the poll addresses must match.

NOTE: If nonswitched lines are used in Mini 6 to Mini 6 file transmission, the system configured with a PVE must have TRAN invoked first. Tis will insure that polls sent from the host system will be answered.

Mini 6/BSC 2780 File Transmission

The Mini 6 configures BSC protocol on the desired communications line for file transmission with an IBM system. The BSC directive must specify EBCDIC (see Section VI, "BSC Directive"). The same argument specifications listed above apply.

MEMORY SIZE

The memory requirements of the TRAN program, as detailed in the appropriate *File Transmission* manual, must be included to calculate the size of the system poll area as described in this manual. This is done to meet the requirement of shareable bound units.

The memory pool associated with the task group(s) in which TRAN will be initiated must meet the size requirements of task group control, lead task control block, ECL task, plus the additional work spaces as detailed in the appropriate File Transmission manual.

NOTE: A task group must be created for each concurrent execution of TRAN. The available memory for each group must, again, meet the size requirements as stated above.

APPENDIX H

2780/3780 WORKSTATION FACILITY CONFIGURATION

For the 2780/3780 Workstation Facility (WF) to interface effectively with the host system, the following Mini 6 configuration environment must be established:

- Noncommunication directives to define the following noncommunications aspects of the system software:
  - System variables
  - Characteristics and sizes of memory pools
  - Physical and logical characteristics of peripheral devices used by the 2780/3780 WF.
  - System overlays
  - System software extensions
- Two communications directives used to define the following:
  - The priority level at which the MLCP (Multiline Communications Processor) interrupts the central processor (COMM communications directive)
  - The binary synchronous line protocol handler (BSC communications directive).
  - Use (or non-use) of multi-record feature.

Usage of both communications and noncommunications directives are described in details in this manual.

Figure H-1 shows a typical Mini 6 configuration required to execute the 2780 WF.

(1)
IOR 2780 WF, the multi-block-count must be either 2 or 7 to match the host configuration.

```
($H)DEVICE KSR00,0,5,X'0050',CONSOLE,140
($H)SYS ,,,10,40,40,,E
($H)DEVICE RCD01,4,9,X'1480'                    Noncommunications
($H)DEVICE LPT00,5,18,X'1380',LPT00
(#H)DEVICE CDR00,6,19,X'1300',CRD00
($H)DEVICE DSK00,7,12,X'400'                    Directive
($H)DEVICE DSK01,8,13,X'480'
($H)DEVICE DSK02,9,14,X'1200'
($H)DEVICE DSK03,10,15,X'1280

($H)COMM20
($H)BSC 22,22,X'FC00',2,P,EB,7      Communications Directives

($H)MEMPOOL S,,10000
($H)MEMPOOL ,AB,*                   Noncommunications Directives
($H)*
($H)QUIT
```

Figure H-1. Typical Mini 6 Configuration to Execute the 2780 WF

APPENDIX J

HASP WORKSTATION FACILITY CONFIGURATION

For the HASP Workstation Facility (WF) to effectively interface with the host system, the following Mini 6 environment must be established:

- Noncommunications directives to define the following noncommunications aspects of the system software:
  - System variables
  - Charactertistics and sizes of memory pools
  - Physical and logical characteristics of pheripheral devices used by the HASP WF
  - System software extensions
- Two communications directives used to define the following:
  - The priority level at which the MLCP (Multiline Communications Processor) interrupts the central processor (COMM communications directive).
  - The HASP line protocol handler (HASP communications directive)

Figure J-1 shows a typical Mini 6 configuration required to execute the HASP WF. In this example, the Host Link parameter (-N) could be any one of the following: -N 33, -N !HASP1, or -N>SPD>HASP1.

```
SYS ...,10,20,20,E
DEVICE KSR00,0,5,X'0500',CONSOLE,140
DEVICE RCD01,4,9,X'1480'

DEVICE RCD02,5,10,X'1500'
DEVICE RCD03,6,11,X'1580'
DEVICE FCD01,8,9,X'1480'

DEVICE FCD02,9,10,X'1500'
DEVICE FCD03,10,11,X'1580'
DEVICE KSR01,11,12,X'0580',TTY01            Noncommunications
                                            Directives
```

```
DEVICE DSK00,12,13,X'0400'
DEVICE DSK01,13,14,X'0480'
DEVICE DSK02,14,15,X'1200'

DEVICE DSK03,15,16,X'1280'
DEVICE MT900,16,17,X'1600'
DEVICE LPT00,17,18,X'1300',LPT00

DEVICE CDR00,18,19,X'1300',DCR00

COMM 25                                       Communications
                                              Directives
HASP 33,34,X'FC00'
DEVICE HASP, 33,34,X'FC00',HASP1

MEMPOOL S,, 10000

MEMPOOL ,AB,*                                 Noncommunications
QUIT                                          Directives
```

Figure J-1. Typical Mini 6 Configuration to Execute the HASP MF

APPENDIX K

PROGRAMMABLE FACILITY/3271

NOTE: See the 3270 Interactive Facility User's Guide for Programmable Facility/3271 (PF/3271) operating instructions.

After an initial system startup, invoke the Editor to build: (1) a CLM_USER file used in configuration of the system software that supports the PF/3271 and (2) a login terminals file used to describe the characteristics of each VIP 7200 terminal used in the dynamic login of VIP 7200 terminals. A CII HONEYWELL BULL supplied generation program (ZF32IN) must then be invoked to build an intialization file. If COBOL programming extensions are used in conjunction with PF/3271, the COBOL program must be linked to a special interface routine (ZCOBIF). The details of the initialization file, CLM_USER file, login terminals file, and COBOL interface are described in this Appendix .

CLM_USER FILE

The CLM_USER file is used to configure GCOS to support the PF/3271. The PF/3271 is composed of the following two bound units: (1) one defining the line protocol handler (LPH) for the host system and (2) one to define the application portion of the PF. The line protocol handler must be loaded at stage 2 system startup (see "Stage 2 System Startup (Intermadiate System Startup)", Section III). The directive that defines (loads) the line protocol handler for the host system is described in Section V of this manual (see H3270 Directive).

Configuration information identifying VIP 7200 terminals, on lines serviced by the Asynchronous Terminal Driver (ATD), is supplied through use of ATD directives (See Section V of this manual). When the system is rebooted, at stage 2 system startup from the CLM_USER file, a communications line is configured for each ATD terminal display plus one communications line for the host system.

CLM_USER FILE EXAMPLE

The following is a typical example of a CLM_USER file used in the generation of the PF/3271:

```
CLM_USER FILE
DEVICE KSR00,5,X'0600','CONSOLE',140
SYS 60,50,,2,20,60,5,E
DEVICE LPT00,30,10,X'0680'
COMM 7
DEVICE ATD00,13,9,X'FF80',VIP00,80,N
DEVICE ATD01,14,9,X'FE80',VIP01,80,N
DEVICE ATD02,15,9,X'FF00',VIP02,80,N
DEVICE ATD03,16,9,X'FE00',VIP03,80,N
DEVICE BSC00,12,8,X'FC00',BSC00
DEVICE BSC01,12,8,X'FC80',BSC01
ATD00 13,9,X'FF80',0,9600,'7200'
ATD01 14,9,X'FF00',0,9600,'7200'
ATD02 15,9,X'FE00',0,9600,'7200'
ATD03 16,9,X'FE00',0,9600,'7200'
H3270 12,8,X'FC00',0,X'40',X'60'
H3270 11,8,X'FC80',2,X'40',X'60'
MEMPOOL 5,,17000
MEMPOOL ,AB,*
MEMPOOL ,AC,*
MEMPOOL ,L0,*
MEMPOOL ,L1,*
MEMPOOL ,L2*
MEMPOOL ,L3*
QUIT
```

In this example of a CLM_USER FILE, the following is shown:

- Four ATDnn directives have been used causing four communications lines to be configured, one per VIP 7200 terminal.
- A DEVICE directive has been "provided" with each ATD00 directive so that each pair contains the same lrn, level, and channel number.

  NOTE: See "DEVICE Directive", in Section 5 of this manual.

- Two H3270 directies have been used indicating that the line protocol handler (lph) will be loaded once defining two host system links.
- A DEVICE directive has been "paired" with each H3270 directive so that each pair contains the same lrn, level, and channel number.

  NOTE: See "DEVICE Directive", Section V of this manual.

LOGIN TERMINALS FILE

The login terminals file is used to describe the characteritics of each VIP 7200 terminal to be monitored by the listener (a system software component) for access requests issued by the PF/3271. The file is created with the Editor and consists of: (1) variable-length G-, T- and A-type records and, optionally, (2) a LOGIN command.

The following is an example of building a login terminals file.

<u>Example:</u>

Building the login terminals file (after initial system startup).

<u>Entries</u>	<u>Comments</u>
Editor Commands And Directives:	
ED	Command to load the Editor
A	Append directive used to enter login records (comprising the terminals file) into the current buffer
<u>Login Record Entries:</u>	
G 1 3 T VIP00 A A A B A C	G-, T- and A-type records comprise this example of the login terminals file to be named "TERM1". In this example, only VIP00 will have access to host system links BSC01, BSC02 and BSC03. The host system links will be established through issuance of three separate login commands from the operator's console.
<u>Editor Command And</u> <u>Directives</u>	
!F	Editor escape character
P	Print directive used to display contents of the current buffer.
W TERM1	Write directive used to write contents of the current buffer to the login terminals file named TERM1
QUIT	Causes the Editor to cease reading commands and directives.

The following example shows three possible login terminal files incorporating login commands.

Example:

1. Building the login terminals file (after initial system startup).

Entries	Comments
Editor Commands And Directives	
ED	Command to load the Editor
A	Append directive used to enter login records (comprising the terminals file) into the current buffer
Login Record Entries	
	Possibility #1
G 2 1 T VIP00 L HOST1 AA -ARG !BSC01 C1	G- and T-type record entries comprise this example of a login terminals file. In this example, only VIP00 has access to host link BSC01.
	Possibility #2
G 2 3 T VIP00 A A L HOST1 AA -ARG !BSC01 C1 A B L HOST2 AB -ARG !BSC02 C2 A C L HOST3 AC	G-, T- and A-type records comprise this example of a login terminals file. In this example, only VIP00 has access to host links BSC01, BSC02, and BSC03 because the T-record precedes the A-records. Note: If multiple T-records precede the A-records, only the last entered T-record is recognized.
	Possibility #3
G 2 4 A A L HOST1 AA -ARG !BSC01 C1 A B L HOST2 AB -ARG !BSC02 C2 A C L HOST3 AC -ARG !BSC02 C3 T VIP00 T VIP01 T VIP02 T VIP03	G-, Ap and T-type records comprise this example of a login terminals file. In this example, all VIPs have access to host links BSC01, BSC02 and BSC03 because the T-records follow the A-records.

Editor Commands And Directives	
!F	Editor escape character
P	Print directive used to display contents of the current buffer
W TERM1	Write directive is used to write the contents of the current buffer (for one of the previous three possibilities) to a login terminals file named TERM1.
QUIT	QUIT directive causes the Editor to cease reading directives.

INITIALIZATION FILE

The initialization file is used to define the site-specific information used to interface with the PF/3271. A CII HONEYWELL BULL supplied generation program, ZF32IN, is provided to perform the following:

- Create the initialization file
- Update an existing initialization file
- List (partially or totally) an existing initialization file on the operator's console.

The generation program operates in an interactive mode consisting of alternating questions and responses between the generation program and you. Depending on the site-specific information to be included in the initialization file, two levels of functionality are provided by the generation program.

Level 1 Functionality

At this level, the generation program allows the following initialization file/record manipulation:

- Inclusion of records defining host system linkages (through use of the HOST directive, described later in this appendix).
- Deletion of existing host system records and affiliated device assignment records (through use of the DELETE directive, described later in this appendix).

- Positioning of the initialization file to either receive new device assignment records or update existing device assignment records affiliated with host system records (through use of the POSITION directive, described later in this appendix).

- Listing, on the operator's console, of all information pertaining to either: (1) all records defining host system linkages or (2) a specific record defining a host system linkage (in both cases, through use of the CURRENT directive, described later in this appendix).

- Specification of the number of Screen Image Buffers (through the BUFFERS directives, described later in this appendix).

Directives, which interface with the generation program in the interactive mode, are used to implement the previously described initialization file/record manipulation. Table K-1 shows the level 1 directives which are used to define host system linkages.

Level 2 Functionality

At this level, the generation program allows the following initialization file/record manipulation:

- Definition of device (address) assignment records (through use of the PRINTER, VIRTUAL and SCREEN directives).

- Updating or deletion of device (address) assignment records (through use of the Y, N and D directives).

Directives, which interface with the generation program in the interactive mode, are used to implement the previously described initialization file/record manipulation. Table K-2 shows the Level 2 directives which are used to define or update device (address) assignment records. depending on the desired application, both Level 1 and Level 2 directives can be used together.

In the format descriptions shown in the tables which follow, brackets ([ ]) enclose optional arguments and braces ({ }) indicate a choice.

Table K-1. Level 1 Directives

Generation Program Prompts	Level 1 Directives/Responses
ENTER CURRENT FILENAME	Enter name of new initialization file to be created or existing file previously created by present generation program. name consists of from 1 to 12 alphanumeric characters; the first character must be alphabetic or a dollar sign ($).
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT	HOST pathname [block size] CURRENT (pathname] BUFFERS nn QUIT HOST If a new pathname is specified, entry of this directive creates a new host system record in the initialization file which defines the host system linkage. pathname consists of !xxxx, where xxxx represents a communications device_unit (see "DEVICE Directive," Section V of this manual). blocksize (nnnn) uses a default value of 2000 if not otherwise specified. If the host system record already exists, you can position, delete or update host system records or device (address) assignment records (see POSITION and DELETE directives, later in this table). If you decide to position records, you can then choose to update device (address) assignment records (see PRINTER, VIRTUAL and SCREEN directives, Table K-2, later in this appendix).

Generation Program Prompts	Level 1 Directives/Responses
	CURRENT [pathname] If pathname is specified, all information pertaining to a specific record (defined through a HOST directive) identifying a host system linkage is listed. If no pathname is specified, all information pertaining to all records (defined through several HOST directives) identifying several host system linkages is listed. pathname consists of !xxxx where xxxx represents a communications device_unit (see "DEVICE Directive," Section V of this manual). BUFFERS nn Enter indicates the number of Screen Image Buffers (SIBs). nn consists of values from 1 to 99. If the number of SIBs have been previously specified, entry of nn overlays any previous number. NOTE: 'COPY' minimally requires two buffers. The number of buffers should equal one-half the number of screens. NOTE: The prompt ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT again appears following the BUFFERS entry. QUIT Entry indicates the initialization file will be closed and generation program ZF32In will be exited.

Table K-1 (Cont). Level 1 Directives

Generation Program Prompts	Level 1 Directives/Responses
DO YOU WISH TO POSITION OR DELETE	POSITION DELETE POSITION Entry positions the initialization file to receive new device (address) assignment records or to update existing device (address) assignment records. Once the file is positioned, entry or updating of device assignment records can occur following the prompt: ENTER COMMAND OR QUIT (See Table K-2 for details). DELETE Entry deletes a specified (via HOST directive) host system record and any device (address) assignment records affiliated with it.

Table K-2. Level 2 Directives

Generation Program Prompts	Level 2 Directives/Responses
ENTER COMMAND OR QUIT	{ PRINTER dn pathname / VIRTUAL dn pathname / SCREEN dn [pathname] [1920] } PRINTER dn pathname Entry indicates a printer device (address) assignment. dn consists of a valid device address selected from Table K-4, later in this appendix. If the device (address) assignment record (affiliated with the particular host system record (has not been previously defined, a new device (address) assignment record has been previously created, you must either update or delete the device (address) assignment record. NOTE: Update and delete functions are described later in this table. pathname consists of !xxxx, where xxxx represents a standard device_unit (see "DEVICE Directive," Table 4-2 in this manual). VIRTUAL dn pathname Entry indicates an application program device (address) assignment. dn consists of a valid device address selected from Table K-4, later in this appendix. If the device (address) assignment record (affiliated with the particular host system record) has not been previously defined, a new device (address) assignment record will be created. However, if the device (address) assignment record has been previously created, you must either update or delete the device (address) assignment record.

Table K-2 (Cont). Level 2 Directives

Generation Program Prompts	Level 2 Directives/Responses
	NOTE: Update and delete functions are described later in this table. pathname consists of the pathname of the application program accessing the virtual device. For a description of pathnames, see the appropriate Operator's Guide. SCREEN dn [pathname] [1920] Entry indicates a VIP 7200 terminal display unit. dn consists of a valid device address selected from Table K-4, later in this appendix. If the device (address) assignment record (affiliated with the particular host system record) has not been previously defined, a new device (address) assignment record will be created. However, if the device (address) assignment record has been previously created, you must either update or delete the device (address) assignment record. NOTE: Update and delete functions are described later in this table. Optional: pathname is the pathname of the user's application program. NOTE: Level 1 directives HOST, CURRENT or BUFFERS can be entered following the prompt: ENTER COMMAND OR QUIT. However, the effect will be to terminate any other entries, update any device (address) assignment records and return to Level 1 processing (see Table K-1, earlier in this appendix).

Table K-2 (Cont). Level 2 Directives

Generation Program Prompts	Level 2 Directives/Responses
UPDATE OR DELETE EXISTING DEVICE ADDRESS? (YES, NO, DELETE)	Y[ES] Entry indicates device (address) assignment record will be updated with information previously entered through a device (address) assignment record. N[O] Entry indicates device (address) assignment record will not be deleted. D[ELETE] Entry indicates device (address) assignment record will be deleted (in effect, disassociated from the current host system record).

Error Messages

The following error messages, shown in Table K-3, are produced by the generation program.

Table K-3. Error Messages

Error Messages	Description
ILLEGAL PATHNAME	The pathname used on a CURRENT directive could not be located. The following generation program prompt is repeated: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT.
PATHNAME MISSING	Either: (1) A HOST directive without a pathname was entered. The following generation program prompt is repeated: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT. Or: (2) A PRINTER or VIRTUAL directive without a pathname was entered. The following generation program prompt is repeated: ENTER COMMAND OR QUIT.
INVALID DEVICE: COMMAND IGNORED	PRINTER, VIRTUAL or SCREEN directive does not have proper device (address) assignment (see Table K-4 for proper device addresses). The following generation program prompt is repeated: ENTER COMMAND OR QUIT.
INVALID PARAM	For the BUFFER directive, nn exceeds two digits. The following generation program prompt is repeated: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT. For the HOST directive, nnnn (block-size) exceeds four digits. The following generation program prompt is repeated: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT.
NO HOSTS CREATED	Before any host system records were created, a CURRENT directive was specified without, in this case, a required pathname. The following generation program prompt is repeated: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT.

Table K-3 (Cont). Error Messages

Error Messages	Descriptions
ILLEGAL ENTRY (CURRENT,HOST, BUFFERS,QUIT)	An invalid entry was made in response to the following generation program prompt: ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT. Note: The prompt is repeated.
ILLEGAL ENTRY (SCREEN,VIRTUAL, PRINTER,CURRENT, (HOST,BUFFERS,QUIT)	An invalid entry was made in response to the following generation program prompt: ENTER COMMAND OR QUIT. Note: The prompt is repeated.

Device Addresses

The following table shows the valid addresses used with the VIRTUAL, SCREEN and PRINTER directives (shown in Table K-2).

Table K-4. Device Addresses

Device or Control Unit Numbers	Addresses
0	40
1	C1
2	C2
3	C3
4	C4
5	C5
6	C6
7	C7
8	C8
9	C9
10	4A
11	4B
12	4C
13	4D
14	4E
15	4F
16	50
17	D1
18	D2
19	D3
20	D4
21	D5
22	D6

Table K-4 (Cont). Device Addresses

Device or Control Unit Numbers	Addresses
23	D7
24	D8
25	D9
26	5A
27	5B
28	5C
29	5D
30	5E
31	5F

Examples of Initialization Files

In Example 1, an initialization file has been created containing one host system record and its associated device (address) assignment records:

Example 1:

```
RDY:
ZF32IN
3271 INITIALIZATION PROGRAM 9066
ENTER CURRENT FILENAME
TESTFL
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
BUFFERS 06
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
HOST !BSC1
ENTER COMMAND OR QUIT
PRINTER C1 !LPT01
ENTER COMMAND OR QUIT
PRINTER C2 !LPT02
ENTER COMMAND OR QUIT
SCREEN D1, PROG1
ENTER COMMAND OR QUIT
SCREEN D2, PROG1
ENTER COMMAND OR QUIT
SCREEN D3, PROG1
ENTER COMMAND OR QUIT
VIRTUAL 5A >DONW>PROG1
ENTER COMMAND OR QUIT
CURRENT
CURRENT DEVICE ASSIGNMENTS:
BUFFER COUNT 06
NUMBER OF HOSTS 01
HOST LINK !BSC1
```

Example 1 (Cont):

```
BLOCKING 0512
SCREEN D1 1920 PROG1
SCREEN D2 1920 PROG1
SCREEN D3 1920 PROG1
VIRTUAL 5A >DONW>PROG1
PRINTER C1 !LPT01
PRINTER C2 !LPT02

ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
QUIT
```

In Example 2, an initialization file has been created containing two host system records and their associated device (address) assignment records:

Example 2:

```
RDY:
ZF32IN
3271 INITIALIZATION PROGRAM 9066
ENTER CURRENT FILENAME
TESTF1
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
HOST !BSC1 256
ENTER COMMAND OR QUIT
PRINTER C1 !LPT01
ENTER COMMAND OR QUIT
PRINTER C2 !LPT02
ENTER COMMAND OR QUIT
SCREEN D1, PROG1
ENTER COMMAND OR QUIT
SCREEN D2, PROG1
ENTER COMMAND OR QUIT
SCREEN D3, PROG1
ENTER COMMAND OR QUIT
VIRTUAL 5A >DONW>PROG1
ENTER COMMAND OR QUIT
BUFFERS 09
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
HOST !BSC2 1024
ENTER COMMAND OR QUIT
PRINTER C3 !LPT03
ENTER COMMAND OR QUIT
PRINTER C4 !LPT04
ENTER COMMAND OR QUIT
SCREEN,D4,!TTY04
ENTER COMMAND OR QUIT
VIRTUAL 5F >DOWN>PROG2
ENTER COMMAND OR QUIT
CURRENT
```

Example 2 (Cont):

```
CURRENT DEVICE ASSIGNMENTS:
BUFFER COUNT 09
NUMBER OF HOST 02

HOST LINK !BSC1
BLOCKING 0256
SCREEN D1 1920 PROG1
SCREEN D2 1920 PROG1
SCREEN D3 1920 PROG1
VIRTUAL 5A >DOWN>PROG1
PRINTER C1 !LPT01
PRINTER C2 !LPT02

HOST LINK !BSC2
BLOCKING 1024
SCREEN D4 1920 PROG1
VIRTUAL 5F >DOWN>PROG2
PRINTER C3 !LPT03
PRINTER C4 !LPT04

ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
QUIT
RDY:
```

In Example 3, an initialization file (containing the host system records created in Example 2) is positioned to the second host system record and updating and deletion functions are performed on existing device (address) assignment records:

Example 3:

```
RDY:
ZF32IN
3271 INITIALIZATION PROGRAM 9066
ENTER CURRENT FILENAME
TESTF1
ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
HOST !BSC2
DO YOU WISH TO POSITION OR DELETE?
POSITION
ENTER COMMAND OR QUIT
SCREEN,5F,!TTY05,1920
UPDATE OR DELETE EXISTING DEVICE ADDRESS?(YES,NO,DELETE)
YES
ENTER COMMAND OR QUIT
PRINTER C4 !LPT04
UPDATE OR DELETE EXISTING DEVICE ADDRESS?(YES,NO,DELETE)
DELETE
ENTER COMMAND OR QUIT
CURRENT

CURRENT DEVICE ASSIGNMENTS:
BUFFER COUNT 09
NUMBER OF HOSTS 02
```

Example 3 (Cont):

```
HOST LINK !BSC1
BLOCKING 0256
SCREEN D1 1920 PROG1
SCREEN D2 1920 PROG1
SCREEN D3 1920 PROG1
VIRTUAL 5A >DONW>PROG1
PRINTER C1 !LPT01
PRINTER C2 !LPT02

HOST LINK !BSC2
BLOCKING 1024
SCREEN D4 1920 PROG1
SCREEN 5F 1920 PROG1
PRINTER C3 !LPT03

ENTER CURRENT OR HOST WITH PATHNAME, BUFFER OR QUIT
QUIT
RDY:
```

COBOL Interface Program

To run a COBOL program on the PF/3271, the COBOL program needs a special interface. The object module of the COBOL program must be linked with the CII HONEYWELL BULL supplied interface program (ZCOBIF). The following is an example of creating such a bound unit:

```
LINKER XXXXXX
LIB >ZCIRT
LINKN  ZCOBIF, YYYYYY: MAP: QUIT
```

where XXXXXX is the bound unit name assigned by the user and YYYYYY is the name of the COBOL program object file.

APPENDIX L

LISTENER COMPONENT AND LOGIN CAPABILITY

INSTALLING A SYSTEM LOGIN CAPABILITY

The listener component enables access to the system from a designated set of terminals. It must run as the lead task of task group $L, and can be activated using commands in the system START_UP.EC file or by entering commands from the operator terminal after system startup is complete and the system is operational.

When the listener is active, a user can log in to the system from the noncommunications terminals (MDC-connected) or communications terminals (MLCP/DLCP-connected) listed in a "terminals" file. The terminals monitored by listener for a login command cannot be directly reserved by system applications.

To provide a system with the login capability, make the following preparations:

1. Create the terminals file which describes the login characteristics of each terminal to be used for login. This file determines in which one of three ways login is to be achieved. A terminal can (a) require a LOGIN command typein, (b) allow a user to type an abbreviation for the login command line, or (c) be immediately logged in, without a typein, when it is ready or connected.
2. Allocate memory pools for login by use of the CLM MEMPOOL directive.

Once these preparations are completed, you can activate the listener as the lead task group $L. Listener activation is described later in this appendix.

MEMORY POOLS FOR LOGIN TASKS

When you log in as a primary user at a terminal, a task group is spawned which has that terminal as its user_in and user_out files. When you enter a LOGIN command, you may specify the task group identification (a two-character group id) or accept a default group id. For each task group to be spawned by a login command, you must define a memory pool whose pool_id is the same as the group_id. The first character of a default group_id/pool_id is an alphabetic character that may be specified by the user when the listener is activated. If no character is specified, the letter L becomes the first character. The second character is the next unused character from the sets 0,...,9 and A,...,Z. If only defaults are taken, the number of default pools must at least equal the number of users who may concurrently gain access to the system. This number is specified in the G-record of the terminals file. You can define pools that are completely overlapping by specifying each pool in a separate MEMPOOL directive. (This also applies to any other type of memory pool that you wish to define). For example, to make available all of the nonexclusive pool area for the three users who accept default pool_ids, use the following directives:

```
MEMPOOL ,L0,*
MEMPOOL ,L1,*
MEMPOOL ,L2,*
```

Additional examples illustrating the allocation of memory are included in the description of the MEMPOOL directive.

TERMINALS FILE

Listener determines which terminals to monitor for system access from information in a terminals file. The pathname of the terminals file may be specified in the command that requests the listener task, or may be defaulted to >SID>TERMINALS. The file is created with the Editor and consists of variable-length G-,T-, and A-type records. Arguments within a record are separated by one or more blank characters. For a terminal to have the direct login characteristic, the LOGIN command must be specified in the T-record for that terminal.

For a terminal to have the option of accepting abbreviations for LOGIN commands it requires A-records with the desired command line image and the absence of a login line in the T-record for that terminal. One or more abbreviations can be specified. The A-records following a T-record are associated only with that terminal. The A-records following the G-records allow all terminals to use those abbreviations for command lines. When the same abbreviation is used in an A-record following a G-record, and in an A-record following a T-record, the command line image in the A-record following the T-record is used for the terminal. The layout of the records of the terminals file is shown below.

G-Record (only one per file) [A-Records — one or more for all terminals]
T-Record — for a specified terminal [A-Records — one or more for the above terminal]
T-Record — for another specified terminal [A-Records — one or more for the above terminal]

G-Record In Login File

There is one G-record in the login terminals file, in the format:

G base-lvl max_user

base_lvl

Level, relative to the lowest numeric (highest priority) level not used by the system group, on which the lead task of a group spawned by listener for a terminal is to execute unless a level is specified in the login line.

max_user

Maximum number of concurrent logged-in users allowed on the system. This value does not include task groups created or spawned by commands other than LOGIN. Logins that exceed this limit are terminated and the listener issues the message.

3915 NUMBER OF CURRENT USERS EXCEEDS THE MAXIMUM

T-record In Login File

There is one T-record in the terminal login file for each terminal on which a user may log in, in the format:

T [A] dev_name [login_line]

A

Specifies that only abbreviated logins will be allowed at this terminal. If a user specifies a nonabbreviated login line at this terminal, the listener issues the following error message:

3916 : LOGIN MUST BE BY ABBREVIATION

dev_name

Symbolic device name of the terminal, as specified at configuration.

login_line

The login command line image (including the LOGIN or L characters) used instead of a user typein when a terminal is to be used for direct login.

A-Record In Login File

An A-record contains an abbreviation character and the associated LOGIN command line image that the listener will use when a user types in the abbreviation. A variable number of A-records may follow the G-record and/or any T-record. When a user enters an abbreviation, listener scans the A-records following the T-record for that terminal and if a match is found, uses that login line for logging in. If the abbreviation is not found, listener scans the A-records following the G-record for a match, and if a match is found, uses that login line for logging in. If no match is found, the listener issues the message:

390E ABBREVIATION FOR TERMINAL NOT FOUND

The format for the A-record is:

A abbrev login_line

abbrev

A 1-character abbreviation that a user can optionally type in when logging in on this terminal.

login_line

The LOGIN command line image associated with the abbreviation.

LOGIN COMMAND

The abbreviated description of the LOGIN command that follows shows the type of entries that may be included in the A- and T-records of the terminals file. For a detailed description of the LOGIN command, see the Commands manual. Note that the length of the login command line is limited to 110 characters.

The LOGIN command causes (1) a task group associated with the user's terminal to be spawned, or, (2) the terminal to be attached to an existing task group as a secondary terminal. Once he has access to the system, the user cannot again invoke login unless he first uses the BYE command or the task group is otherwise terminated.

FORMAT:

```
L   login_id  [destination_id]   [ctl_arg]
```

ARGUMENT DESCRIPTION:

login_id

Establishes the identity of the user who is attempting to gain access to the system. Provides the user identification for the spawned task group. The login_id argument consists of from one to three fields having the following meanings:

person

person.account

person.account.mode

person

Name of person who may access system: can be from 1 through 12 characters. (For example, WDSMITH could be the value for the person field).

account

Name of an account under which the user is to work: can be from 1 through 12 characters. (For example, JSINVENTORY could be used as the value for the account field).

mode

Provides a further identification of the user: can be from 1 through 3 characters. (For example, VER could be used as the value for this field).

[destination_id]

Optional argument that permits the user to log in as a secondary user of an existing task group. (A request for a secondary user terminal must have been previously issued by that task group). To log in as a secondary user of a user-created applications program, enter the value id, where id is the task group id of the task group in which the application is running. If this argument is specified, no other control arguments may be specified.

[ctl_arg]

None or any number of the following control arguments can be selected:

$$\text{-PO} \begin{Bmatrix} * \\ \text{path} \end{Bmatrix} \begin{Bmatrix} * \\ \text{id} \end{Bmatrix} \text{[level]}$$

Used to override the default lead task, group id/pool id, and relative level specifications for the task group spawned as a result of this login procedure.

path

Pathname of the bound unit to be executed as the lead task of the spawned task group. If the star character (*) is entered the lead task is the command processor.

id

Group_id/pool of the spawned task. The group_id and the pool_id are represented by the same 2-character value. If this argument is not specified, a default id is assigned as for memory pools.

Default: The group id is a 2-character value whose first character was specified when the listener component was activated and whose second character is the next unused character in the sequence 0 through 9 and A through Z, assigned by the system.

level

Base priority level at which the spawned group is to execute (relative to the system group level). A base level of 0 is the next higher level. The sum of the highest system physical level plus 1, and the base level of the group, and the relative level of a task within that group, must not exceed 62 (decimal).

Default: The base priority level is that specified in the terminals file, relative to the system group level.

-HD path

Used to specify the home directory for the task group spawned as a result of the login procedure. The home directory is your initial working directory when you log in under an account.

path

Pathname of the initial working directory for the spawned task group. You must specify a full or relative pathname (not a simple pathname). If this argument is omitted, the working directory pathname is null.

-LRN n

Used to override the default maximum logical resource number (LRN) value for the task group spawned as a result of this login procedure.

n

Maximum LRN value to be used for the spawned task group. (The maximum possible LRN value is 252). If this argument is omitted, the maximum LRN value is 1.

-LFN n

Used to override the default logical file number (LFN) value for the task group spawned as a result of this login procedure.

n

Maximum LFN value to be used for the spawned task group. (The maximum possible LFN value is 255). If this argument is omitted, the maximum value is 15.

-HOLD

Set the current connect/disconnect device-specific word to specify that the terminal is to be disconnected without phone hangup. This argument allows you to reuse the terminal without redialing after logging off the system. (See the set terminal characteristics (STTY) command or directive for information about the connect/disconnect device-specific word).

Default: The terminal will be disconnected according to the setting of the current connect/disconnect device-specific word.

-ARG arg arg . . . arg

Used to pass additional arguments to the lead task of the task group spawned as a result of this login procedure.

LISTENER ACTIVATION

Listener is activated with the CG (create group) and EGR (enter group request) operator commands, or with an SG (spawn group) operator command, using the arguments shown below. These commands and their arguments are described in the Commands manual.

Once activated listener can be terminated only if the system shuts down, or by using the -QT argument of the SET_LISTEN operator command. Once terminated, the $L task group can be recreated as described below.

```
CG $L base _lvl -EFN LISTENER -POLL id

                                        [{'path  '}]
EGR $L user_id -OUT !CONSOLE -ARG       [{"path  "}]    [X] ["message"]

SG $L user_id base_lvl -EFN LISTENER -POOL id

                                        [{'path  '}]
             -OUT  !CONSOLE - ARG       [{"path  "}]    [X] ["message"]
```

[{'path'}
{"path"}]

Pathname of the terminals file, which list the terminals on which users may log in, and which contains the terminal characteristics records.

The last character in the pathname must be a blank and the entir pathname must be enclosed in either single or double quotes. An omitted (default) pathname must be written as a pair of enclosing single or double quotes (' ') or (" "), and results in the default pathname >SID>TERMINALS.

[x]

The first character in the 2-character pool_id and group_id when default values are used. The second character, from 0 through 9 or A through Z, is appended when a task group is spawned as a result of the LOGIN command. When this argument is omitted, its default value is L.

When a user specifies a group_id in a LOGIN command or in a login line for a T-record or A-record, listener uses that as a group_id instead of generating a group_id.

["message"]

The message-of-the-day, enclosed in quotes to provide for embedded blanks, which listener transmits to all terminals for display.

Terminal State After Listener is Activated

When first activated and again when the session terminates, listener performs specific operations affecting the state of a terminal. The output on the terminal that a user sees and the state of the terminal depend on whether it is a noncommunications or a communications terminal.

NONCOMMUNICATIONS TERMINAL STATE WITH LISTENER

If a terminal is not ready when listener is activated, no initial output messages from listener are displayed when the terminal comes on line.

When listener is activated:

1. If there are terminals online, ready for direct login, they display the message-of-the-day. A task group is spawned for each such terminal if the login_line image contained in that terminal's T-record in the terminal login file specifies primary login. The lead task defined in the login line is executed. The application should display a prompter message to the terminal indicating that it is ready to accept input. When the lead task terminates, the message-of-the-day is displayed and a task group is immediately spawned again.

   If the login line in the terminal's T-record specifies secondary login, the terminal is given to the group named in the destination field, provided that the group has an outstanding request for a secondary terminal. The group returns the terminal to listener's control by executing a release terminal macro call.

2. Terminals that require a user login, display the message-of-the-day and the user login promoter message identifying the system and giving the date and time:

   LOGIN system id yyyy/mm/dd/ hhmm:ss.t

   The user can then type in the LOGIN command. When the lead task terminates, the message-of-the-day is displayed followed by the login promoter message.

COMMUNICATIONS TERMINAL STATE WITH LISTENER

Although a communications terminal may not be ready when the listener is activated, listener displays a message when the terminal comes online. Otherwise, when listener is activated, the same operations are done for communications terminals as for noncommunications terminals described above.

When a terminal is released:

1. A terminal connected by phone and with the hangup option, is disconnected. The user must dial in again to use the terminal.

2. A terminal connected through a modem bypass or by phone without the hangup option, displays the message-of-the-day; either the login promoter message is displayed or, for a direct login, a login task group is spawned.

CHANGING THE LOGIN MESSAGE OF THE DAY

After listener is activated, it places an operator response request to the operator terminal. The request number must be used in the response to listener from the operator terminal that changes the message-of-the-day. The message to the listener cannot exceed 63 characters and is in the form:

Δmsg_noΔ message-of-the-day

EXAMPLES OF LISTENER OPERATION

For these examples, the CLM file includes the following directives:

```
DEVICE    KSR01,11,21,X'0580',KSR01

DEVICE    KSR02,12,22,X'0600',KSR02

MEMPOOL    S,,10000

MEMPOOL   ,LO,*

MEMPOOL   ,L1,*

MEMPOOL   ,M1,*

MEMPOOL   ,AB,*
```

Note that memory pools LO, L1, M1, and AB all share the same memory.

Example 1:

The terminals file for this example has the pathname ^ZSYS51>TERM>T' and contains the following records:

```
G  1  3
T  KSR01
A  X  L  X.X.X
A  Y L Y.Y.Y -PO * M1 -HD ^ZSYS51?SYSLIB2 -LRN 40 -LFN 20
T  KSR02
A  W L W.W.W
```

The listener is activated by command 1 made through the operator terminal. Commands 2,3, and 4 illustrate alternative logins made through terminal KSR01. Note that they cannot be executed in sequence unless each is terminated with a BYE command.

Command 1:

```
Δ$S SG $L A 10 -EFN LISTENER -POOL AB -OUT !CONSOLE
                  -ARG '^ZSYS51>TERM>T1Δ '
```

Command 2:

ΔX

Command 3:

ΔY

Command 4:

ΔW

The operations that result from each command are as follows:

Command 1: Spawns group $L with a user_id of A, a relative level of 10, bound unit of listener as the lead task, uses memory pool AB, the in_path and out_path is the console and argument passed to listener is the location of the terminals file. The character used for default value of the first character of pool_id and group_id was not given and this defaults to L.

Command 2: Listener spawns group L0 with a user_id of X.X.X., lead task of command processor. The command processor issues the message: 170222 (No Working Directory) because -HD option was not used. User may continue after typing in a CWD command.

Command 3: Spawns group M1 with a working directoryy of ^ZSYS51>SYSLIB2 and the lead task is the command processor.

Command 4: Because the A record is under the T record for KSR02 and not under the G record or the T record for KSR01, the listener issues the message.

390E ABBREVIATION FOR TERMINAL NOT FOUND

For a valid login, the W must be typed in on KSR02.

Example 2:

The terminals file for this example has the pathname ^ZSYS51>TERM>T2 and contains the following records:

```
G  1  3

A  Z  L  Z.Z.Z -PO >SYSLIB2>PR M1 -HD ^ZSYS51>SYSLIB2
                -ARG >SUD>CLM_USER  -SP 1

T  KSR01

T  KSR02
```

The listener is activated by command 1 made through the operator terminal. Command 2 is a login made through terminal KSR01.

Command 1:

```
Δ$S SG $L B 10 -EFN LISTENER -POOL LO -OUT !CONSOLE
              -ARG  '^ZSYS51>TERM>T2Δ '
```

Command 2:

ΔZ

The operations that result from each command are as follows:

Command 1: Spawns group $L with a user_id of B, a relative level of 10, bound unit of listener as the lead task: uses memory pool LO: the argument passed to listener is the location of the terminals file. The first chatacter of pool_id and group_id was not specified and defaults to L.

Command 2: Spawns group M1 which prints CLM_USER and then terminates.

Example 3:

The terminals file for this example has the pathname ^ZSYS51>TERM>T3 and contains the following records:

```
G   1   3

T   KSR01   L A.A.A
```

The listener is activated by the following command which is made through the operator terminal.

```
Δ$S SG $L C 10 -EFN LISTENER -POOL M1 -OUT !CONSOLE
              -ARG '^ZSYS51>TERM>T3Δ' M
```

This is an example of direct login. The login command is specified in the T record. The spawn group command, typed on the operator terminal, spawns listener group $L which uses memory pool M1. Listener then spawns group M1 with the command processor as the lead task, M1 as its memory pool, and KSR01 as user_in and user_out.

The functionality of this spawn group command can also be achieved by the joint use of the create group and the enter group request commands as shown below.

```
CG $L 10 -EFN LISTENER -POOL M1

EGR $L C -OUT !CONSOLE -ARG '^ZSYS51>TERM>T3Δ' M
```

A message enclosed in quotation marks may be added to the entries following -ARG. This message will appear on the terminal when the group is activated.

APPENDIX M

M4_SYSDEF CHECKLIST

This appendix is included as a convenience for users of the M4_SYSDEF utility. You can use the checklist provided to write down information which the utility requires to generate your directive file. The information you write down on the checklist will be requested by the utility in the same order.

Note that this checklist does not include all the questions which the utility asks. For example, it doesn't list whether VIP devices support form feed. Also, the number of spaces provided for writing information about peripheral devices isn't meant to imply any limits on how many devices of a given type you can include in your configuration.

M4_SYSDEF CHECKLIST

ITEM		(Options/Units)
1) Total amount of physical memory	________	(nnnK)
2) Default peripheral channel numbers? (If YES, ignore "Channel Number" entries for noncommunications devices)	________	(YES/NO)
3) Commercial system (Model 47/57)?	________	(YES/NO)
4) (If NO to #3): Simulator?	________	(YES/NO)
5) Scientific processor?	________	(YES/NO)
6) (If NO #5): Simulator?	________	(YES/NO)
7) System identification message: (Default: "GCOS 6/MOD400")	____________________	
8) Installation identification: (default: "YOUR TOWN, USA")	____________________	
9) Maximum number of concurrent users:	________	
10) Default SPD names? (If YES, ignore "SPD name" column of subsequent entries)	________	(YES/NO)
11) Record locking?	________	(YES/NO)
12) Display Formattind and Control Software?	________	(YES/NO)
13) Power fail restart?	________	(YES/NO)
14) Communications devices? (If NO, skip items 15 through 21)	________	(YES/NO)
15) Number of communications priority levels:	________	(1 through 4)
16) Number of MLCPs:	________	(1 through 10)

17) MLCP-connected asynchronous terminals:

SPD name	Channel number	Comm. priority level	Device type
____	______	_______	______
____	______	_______	______
____	______	_______	______
____	______	_______	______
____	______	_______	______
____	______	_______	______

18) VIP 7700's or VIP 7800's to be run in VIP 7700 emulation mode:

Channel Number	Comm. priority level	ROP type
______	_______	_____
______	_______	_____
______	_______	_____
______	_______	_____
______	_______	_____

19) H3270 Host Links:

SPD name	Channel number	Comm. priority level	Polling address (dec.)	Select address (dec.)
____	______	_______	______	______
____	______	_______	______	______
____	______	_______	______	______
____	______	_______	______	______
____	______	_______	______	______

20) Polled VIP Emulation Groups:

SPD name	Channel number	Stations/ channel	Comm. priority level	Polling address 1	2	3	4
____	______	________	_______	____	____	____	____
____	______	________	_______	____	____	____	____
____	______	________	_______	____	____	____	____
____	______	________	_______	____	____	____	____

21) Synchronous Terminals:

SPD name	Channel number	VIPs/ channel	Comm. priority level	Control Stations? (YES/NO)	ROP type
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____
____	______	______	_______	________	____

22) Binary synchronous communications lines:

#	Channel number	Comm. priority level	Primary/ secondary? (P/S)
1	______	_______	_________
2	______	_______	_________
3	______	_______	_________
4	______	_______	_________

23) MDC-connected console (terminal) devices:

SPD name	Channel number	Device type	
____	______	_____	<= CONSOLE
____	______	_____	
____	______	_____	
____	______	_____	

24) Diskette devices:

SPD name	Channel number
____	_____
____	_____
____	_____
____	_____

25) Cartridge disk devices:

SPD name	Channel number	Fixed platter? (YES/NO)
____	______	_______
____	______	_______
____	______	_______
____	______	_______

26) Cartridge module disk devices:

SPD name	Channel number
____	______
____	______
____	______
____	______

27) Storage module devices:

SPD name	Channel number
____	_______
____	_______
____	_______
____	_______

28) Card reader devices:

SPD name	Channel number
____	_______
____	_______
____	_______
____	_______

29) Card reader/punch or punch devices:

SPD name	Channel number
____	_______
____	_______
____	_______
____	_______

30) Line printer devices:

SPD name	Channel number
____	_______
____	_______
____	_______
____	_______

31) Magnetic tape drives:

SPD name	Channel number
____	______
____	______
____	______
____	______

32) Batch memory pool? (YES/NO)

size of batch pool — nnnnn or nnk words

33) Number of 512-word system overlay areas (1-99) — (1 to 10)

APPENDIX N

ERROR LOGGING

Error logging is an optional feature that allows the system operator to collect memory or hardware-related error statistics for selected noncommunications peripheral devices. Error logging is intended primarily as a preventive maintenance tool, allowing for early detection and correction of potential memory or hardware failures. The error statistics collected can be used by CII HONEYWELL BULL field engineering personnel to monitor memory and peripheral device performance to determine if corrective actions are required.

As the system builder, you have the option of configuring error logging or not(1). Once configured, the system operator must activate error logging through commands or macro calls. Error logging commands and operating procedures are described in the Command manual and the Operator's Guide; error logging macro calls are described in the System Service Macro Calls manual.

(1)
The interactive building program, M4_SYSDEF, des not allow you to configure error logging. You must incorporate the appropriate software module by hand.

CONFIGURATION REQUIREMENTS

Error logging must be incorporated into your configuration before memory or device error logging can be activated. The following hardware and software requirements are necessary to configure error logging for your installation.

Hardware Requirements

Error logging requires that you have one or more noncommunications devices in your configuration that you wish to monitor. (If you wish to log memory errors exclusively, peripheral devices need not be attached to the central processor). In addition, cumulative file processing (described elsewhere in this appendix) requires that your installation has a CIP (or a CIP simulator).

Software Requirements

To implement error logging, you must load the bound unit ZERRST, located in directory SID, using an LDBU directive. If your working directory is SID, include the following directive in your CLM file:

LDBU ZERRST

Section V of this manual describes the LDBU directive in more detail.

MEMORY REQUIREMENTS

The error logging facility requires a minimum memory area equivalent to the sum of the following components:

- 254(16) words (SAF) or 263(16) words (LAF) in the system memory pool for the bound unit ZERRST (of which 17(16) words, the initialization section, is overwritten after use).
- 22(16) words (SAF) or 23(16) words (LAF) in the system memory pool for each device that is monitored.
- 39(16) words (SAF or LAF) in the system or user memory pool for data for each error logging command.
- Space for the bound unit to process a particular error logging command.

Error logging can be run in the system task group ($S) or in a user group (e.g., $H). Since the memory area necessary to run the error logging bound unit can be quite large, error logging should not be executed from the $S group. The operator should initially create a group large enough to accommodate the error logging session. (If the system operator wishes to create and maintain error logging disk files, additional file space is required as well).

APPENDIX P

CONFIGURING DISPLAY FORMATTING AND CONTROL SOFTWARE

This appendix describes the requirements for configuring the Display Formatting and Control software. Unless you configure this software for your installation, forms processing will not be available to users. If your installation uses the Data Entry Facility-II (DEF-II), you must configure Display Formatting and Control Software. Users should have available for reference the Display Formatting and Control manual.

CONFIGURATION REQUIREMENTS

The following hardware and software requirements are necessary to configure the Display Formatting and Control Software.

Hardware Requirements

To performs forms processing, your installation requires one or more of the following asynchrnous terminals connected to an MDC or an MLCP/DLCP: VIP 7200, VIP 7205, VIP 7207, VIP 7801, or VIP 7802, and a full-duplex modem such as type 103, or one or more of the following synchronous terminals on MLCP: DKU 7007, DKU 7005.

Software Requirements

To configure forms processing, you must include the appropriate directives in your CLM file as described below.

CONFIGURATION DIRECTIVES

The CLM directives required to configure the Display Formatting and Control Software depend on whether you wish to configure your terminals as noncommunications (MDC-connected) terminals or communications (MLCP-connected) terminals. The directives mentioned below are fully described in Sections V and VI.

Configuring Noncommunications Terminals for Forms Processing

The CLM directives required to configure the Display Formatting and Control software with communications (MDC-connected) asynchronous terminals are as follows:

VDAM

This directive must be specified if the Display formatting and Control software is to be incorporated in the configuration. (See Section VI for a full description of this directive).

```
DEVICE CONnn,lrn,level,X'channel',[device_name],
     [record_size], [{B}] , {'7200'  , {C}
                    [{N}]   {'D7200'    {T}
```

For each asynchronous terminal capable of running Display Formatting Software, you must specify a DEVICE directive with the first parameter as CONnn. The eighth parameter must be specified as either "7200" (for VIP 7200 or VIP 7205 support) or "D7200" (for VIP 7207 support). The ninth parameter must be specified as either C (for CRT visual display mode) or T (for teleprinter mode). The terminal runs in the mode you specify whenever it is not running in forms mode (i.e., running the forms processing software). Forms mode overrides visual display mode or teleprinter mode.

Configuring Communications Terminals for Forms Processing

The CLM directives required to configure the Display Formatting and Control software with communications (MLCP-connected) asynchronous terminals are as follows:

VDAM

This directive must be specified if the Display Formatting and Control software is to be incorporated in the configuration. (See Section VI for a complete description of this directive).

COMM n

This directive is required for a system that includes communications. It must precede all other communications-related directives. See Section VI for a description of n, the number of interrupt priority levels.

```
LDBU    ZQPACT

LPHDEF      1,96,120

LPHI lrn,level,X'channel',[modem],speed,FDX,lph-specific-word

DEVICE ATDnn,lrn,level,X'channel',[device_name],
       [record_size]    , [{B}]
                          [{N}]
```

For each asynchronous terminal capable of running Forms Processing Software, you must specify an ATD directive and its "paired" DEVICE directive(1). The sixth parameter of the ATD directive must be specified as either 7200 (for VIP 7200 or VIP 7205 support), or F7200 (for VIP 7207 support). VIP terminals require a full duplex modem (such as type 103) to run forms processing. Additionally, the data rate (speed) in bits per second for any asynchronous terminal that will run forms processing should be i the range of 2400 to 9600.

For use of the VDAM Block Mode terminal support, you must add

```
LDBU ZNV77F
```

and you must describe, through

```
STDLN
STD
POLIST
STAPOL
```

The configuration of synchronous terminals to be used, the common device-type being V7760, both for DKU 7005 and DKU 7007.

(1)
Alternatively, you may specify a V7200 directive and its "paired" DEVICE directive. However, CII HONEYWELL BULL recommands using the Asynchronous Terminal Device (ATD) driver.

CONFIGURATION OPTIONS

In addition to the configuration requirements described above, you have two options to consider when configuring forms processing for your installation. You may choose to activate the listener capability for terminals in your configuration. You also have the option of configuring the operator terminal such that you can perform forms processing in a one-terminal configuration.

Listener performs specific operations affecting the state of a terminal. Whether you wish to activate the listener or not depends entirely on your installation's processing requirements. Refer to Appendix L for a complete description of the listener capability.

The terminal configured as a standard operator terminal in your configuration does not support forms processing. If, however, the operator terminal has been configured as a dual-purpose terminal that runs alternately under operator control and user control, users can utilize the forms processing capability when the terminal is under user control (i.e., not functioning as an operator console). In this way it is possible to run forms processing in a one-terminal configuration. Refer to Section V for information on configuring a dual-purpose operator terminal.

APPENDIX R

ASYNCHRONOUS CHARACTER TERMINAL DRIVER (ACTD)

This appendix provides information necessary to configure a station on a line serviced by ACTD (asynchronous character terminal driver) through the Communication System.

DIRECTIVES:

The Asynchronous Character Terminal Driver must be loaded and initialized at CLM time when configuring terminals as DKU 7002 and its auxilary components, DKU 7001, VIP 7200, VIP 7100 or teletype-like devices, in the Communications System.

The configuration directives to be provided are:

LDBU

LPHDEF

LPH1

which are mandatory

DEVICE

Which is to be provided only if the associated device is to be accessed thru File Management System.

LDBU Directive

The main part of the ACTD handler software (ZQPACT) must be explicitly called by an LDBU directive.

LDBU ZQPACT

This directive may be put anywhere in the CLM file. It allows to load, residently, in memory the ACTD software.

LPHDEF directive

The ACTD line protocol handler makes use of two channel-tables and one station table for each serviced station. The sizes of these tables must be defined by an LPHDEF directive (involved by the further described LPHI directive)

Format:

```
LPHDEF    1,95,120
```

Argument Description

1 value led by, the use of the further described directive LPH.

95 channel-table size in words (LAF mode)

120 Station-table size in words.

Usage conditions

There must be only one LPHDEF directive regardless of the number of serviced station.

This directive must be below the COMM directive (defining the Communication System part of the CLM file) and must precide any LPHI directive.

LPHI directive

This directive allows to define a station serviced by ACT.

Format:

```
LPHI lrn,level,X'channel',[modem],speed,
     FDX,lph_specific_word
```

In this directive name LPHI, I is the reserved value which has been associated with the ACTD line orotocol handler.

Argument Description:

lrn

The logical resource number associated with the station. The value for lrn is an integer from 2 through 252. A program may use this number to identify the station when it requests an input/output operation to the station.

level

The priority level at which the communications supervisor processes requests for an input/output operation to the station. The value for level is an integer from 7 through 58; it may be the same as the level specified for other communications stations, but it must be a higher number than the communications interrupt level(s) specified in the COMM directive. The level specified for one or more communications stations may not also be used for noncommunications devices or tasks.

X'channel'

A four-digit hexadecimal number (from X'040n' to X'FF8n'), specifying the channel number of the station and the interrupt priority level (n) of the line. The channel number has the following format:

Bits 0 through 9 - The 10-bit channel address of the send or receive channel on the line.

Bits 10 through 13 - Must be set to zero.

Bits 14 and 15 - Specifies n, the priority level at which a communications line interrupts the central processor. n may have a value of 1,2, or 3 as specified in the COMM directive. See the description of the COMM directive for further information.

[modem]

A number specifying the type of data set. Possible values are as follows:

0 - Direct connect.

1 - Bell 1xx-type modem (103A, 113F, etc.). Both data set-ready and carried-detect signals are needed for a connection; absence of both signals is a disconnection.

2 - Bell 2xx-type modem (201A, 201X, 208A, etc.). The data-set-ready signal is needed for a connection; absence of this signal is a disconnection.

3 or greater - User-defined modem type (see "MODEM Directive", later in this section).

The default value is modem type 2.

Speed

The data rate in bits per second. The default value is zero and signifies a synchronous line.

For an asynchronous line with a communications-pac whose id is 2108(16), use one of the following values for speed:

50	300	2400
75	600	3600
110	900	4800
134	1200	7200
150	1800	9600

For an asynchronous line with a communications-pac whose id is 2100(16), 2110(16), or 2118(16), use one of the following values for speed.

50	200	1800
75	300	2000
110	600	2400
134	1050	4800
150	1200	9600

NOTE: If the data rate of the line is 134.5, specify 134.

FDX

Specifies that two channel tables will be assigned. This parameter is mandatory.

lph_specific_word

A word containing information to be passed to the line protocol handler through the station table at offset ZOSSTS.

The structure of this word is

L	L	L	P	S		I	I		F		D	D	D	D	D
0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F

L These three bits indicate the presence of auxiliary units of the device. These bits will be ignored if the device type equals 0 or 1.

- Bit 0 indicates the presence of a badge-reader
- Bit 1 indicates the presence of an auxiliary printer
- Bit 2 MBZ

P Bit of parity specifies the parity to be used.

0 = odd

1 = even

S Stop bit-specifies the number of stoo bits that are to follow each character.

0 = 1 stop bit will be used

1 = 2 stop bit will be used

II DEVICE type-specifies the type of terminal used

0 = VIP7100, teletype like device

1 = VIP7200, DKU 7001

3 = DKU 7002

F Field-mode specifies whether the terminal will be allowed to be connected in field mode or not.

0 = field mode allowed

1 = field mode not allowed

D These five bits specify the number of DEL characters that are to follow LF characters in the pre-order ore-space or in the postorder control. The value chosen may be 1 up to 32.

NOTE: The support of the badge reader subsystem and the support of the printer subsystem are two parts of the ACTD software, each part is residently loaded only if the presence of the correspondent subsystem has been indicated in one of the LPHI directives in the CLM file.

Usage conditions

There must be as many LPH directives as there are stations serviced by ACTD line protocol handler.

All these directives must appear after the COMM directive and the LPHDEF directive to which they are associated. No others directives may be mixed with the set of LPHI directives.

DEVICE Directive

This directive must be used if

- either the driver is accessed by the file System in TTY mode.

- or the device referenced is used like a "secondary" terminal.

Format:

```
DEVICE device_unit,lrn,level,X'channel',device_name,
        [record_size] [,{B|N}]
```

Argument Description:

device_unit

A string of five ASCII characters: the first three must be ATD and the last two characters (alphanumeric) identify one specific station of that type. Must be in the format ATDnn.

lrn,level and X'channel'

are previously described

device_name

A string of 1 to 6 ASCII characters, the first of which must be alphabetic. This device_name is the unique File System name.

Record_size

The length in bytes of one physical record (range)

B/N

File System definition of buffered/nonbuffered device.

B = buffered

N = nonbuffered

Usage conditions

There must be a DEVICE directive each time a station is serviced by ACTD in TTY mode three the file system or each time a station is to be logged as a "secondary" station.

All the DEVICE DIRECTIVES IN A CLM file must appear after the set of related LPHI directives.

```
COMM 24

LPHDEF    1,96,120

* DKU 7002 with a badge reader
*    (even parity, 1 stop bit, 1 DEL character)

LPHI    16,26,X'1C00' ,0,9600,FDX,X'9301'

* DKU 7001
*    (even parity, 1 stop bit, 3 DEL characters)

LPH    17,27,X'1C80',0,9600,FDX,X'1103'

* ROSY 24
* (odd parity, 2 stop bits, 4 Del characters)

LPHI   18,28,X'1D00',1,300,FDX,X'0844'

* VIP 7200 driven by ATD line protocol handler

ATD    19,29,X'1D80',9600,'7200'

DEVICE   ATD00, 16, 26, X'1C00', VISU1, 140, N
DEVICE   ATD00, 17, 27, X'1C80', VISU2, 140, N
DEVICE   ATD00, 18, 28, X'1D00', VISU3, 140,
DEVICE   ATD00, 19, 29, X'1D80', VISU4. 140, N

LDBU ZQPACT
```

INDEX

D

E

L

M

O

P

S

T,U

V,W,X,Y,Z

Vos remarques sur ce document / Technical publications remarks form

Titre / Title :

MINI 6 GCOS 6 MOD 400/MFS System Building

Nº Référence / Reference No. : 69 A2 CB23 REV3

Date / Dated : March 1982

ERREURS DETECTEES / ERRORS IN PUBLICATION

AMELIORATIONS SUGGEREES / SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

03 84 - 438

▶ Vos remarques et suggestions seront attentivement examinées.
Si vous désirez une réponse écrite, veuillez indiquer ci-après votre adresse postale complète.

▶ *Your comments will be promptly investigated by qualified technical personnel and action will be taken as required.*
If you require a written reply, furnish your complete mailing address below.

NOM/NAME ____________________ DATE ____________

SOCIETE/COMPANY : ____________________

ADRESSE/ADDRESS ____________________

■ Remettez cet imprimé à un responsable BULL ou envoyez le directement à

■ Please give this technical publications remarks form to your BULL representative or mail to :

Bull **CEDOC-CELOG**
Boîte Postale 110
Parc Industriel d'INCARVILLE
27100 Ensemble Urbain du Vaudreuil-FRANCE

Distribution codes/Codes de diffusion			
Customers Clients	SHS938D		
Internal Interne	69340000		

DELIVERY ADDRESS
ETIQUETTE ADRESSE

Goudy-Hélio s.a. - 06/84

Bull Systèmes

94, Av. Gambetta
75990 PARIS Cedex 20

69A2CB23 03

Bull Systèmes