File No. S360-20
Order No. GC28-6691-2

OS

Dump on USER01

IBM Systems Reference Library

IBM System/360 Operating System: Operator's Reference

This book tells the computer operator how to run the IBM System/360 Operating System. It presents general operating procedures for using the various types of operator's console configurations. It describes specific operating techniques for each of the three major system types: systems with the primary control program (PCP); systems that provide multiprogramming with a fixed number of tasks (MFT); and systems that provide multiprogramming with a variable number of tasks (MVT). These operating techniques include:

- How to start, stop, and restart the operating system.
- How to control input and output.
- How to control jobs.

This book also presents operating techniques that apply to all three major system types, as well as a discussion of:

- The remote job entry (RJE) facility.
- The conversational remote job entry (CRJE) facility.
- The telecommunications access method (TCAM).
- The time sharing option (TSO).

Many technical terms are defined.

Information in this publication for TSO, the Model 165, and extended 2880 channel support is for planning purposes until those items are available.

Second Edition (January, 1971)

This is a major revision of, and obsoletes, GC28-6691-1 and Technical Newsletters GN28-2433 and GN28-2444. See the Summary of Major Changes preceding the Contents. Other changes to the text, and small changes to illustrations, are indicated by a vertical line to the left of the change; changed or added illustrations are denoted by the symbol • to the left of the caption.

This edition with Technical Newsletter GN28-2463 applies to release 20.1, of IBM System/360 Operating System, and to all subsequent releases until otherwise indicated in new editions or Technical Newsletters. Changes are continually made to the information herein; before using this publication in connection with the operation of IBM systems, consult the latest IBM System/360 SRL Newsletter, Order No. GN20-0360, for the editions that are applicable and current.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Programming Systems Publications, Department D58, PO Box 390, Poughkeepsie, N. Y. 12602

Preface

You should be familiar with the information described in the publications:

IBM System/360 Operating System:

Concepts for Operators Learner's Guide (Text), SR20-4114

Concepts for Operators Learner's Guide (Illustrations), SR20-4115

To run the operating system, you should have the required books listed below, as well as this reference. In addition, you should be familiar with the recommended books.

REQUIRED BOOKS

IBM System/360 Operating System:

Operator's Procedures, GC28-6692

Messages and Codes, GC28-6631

RECOMMENDED BOOKS

IBM System/360 Operating System:

Job Control Language User's Guide, GC28-6703

Utilities, GC28-6586

Service Aids, GC28-6719

HOW TO USE THE OPERATOR'S REFERENCE

Each chapter in this manual is like a separate book. It has its own table of contents and index and is numbered from page 1 to page n. This allows you to remove a chapter from the book and use it by itself.

Using the separate chapters of this manual you can put together a book customized to your needs. To build your operator's book, do the following:

- Keep the front part of the book. This part contains introductory information and the Summary of Changes.
- The first three chapters are on the three OS control program configurations (PCP, MFT, MVT). Keep the chapter that pertains to your system.
- Keep the chapters "General Operating Techniques" and "Operator's Console." The information in these two chapters is for all three control program configurations.
- Keep the last two chapters in the book. The chapter "Technical Terms and Their Meanings" is where you will find the meaning of a technical term. The chapter "Operator's Reference Master Index" is a combined index of the entire book.
- Keep any other chapters that describe facilities used by your operating system. For example, if your system has TCAM, keep the chapter "Telecommunications Access Method."

You now have an Operator's Reference manual customized to your needs.

If you are familiar with the book and want to know what's new and changed, look for the vertical line to the left of a column and a bullet next to a figure caption. There's a summary of what's new in the Summary of Changes.

If you have a display operator console, refer to IBM System/360 Operating System: Operator's Guide for Display Consoles, GC27-6949 for operating instructions.

If you have any comments or corrections, fill out and mail the form at the back of the book. Many of the features of this book are the result of readers' comments on earlier versions.

Summary of Major Changes--Release 20

Item	Description	Areas Affected
	The chapter "Understanding Messages" has been deleted from the publication. The information can be found in Messages and Codes, GC28-6631.	
Technical Terms	Many new terms have been added to the chapter "Technical Terms and Their Meanings." Terms which are not used in this publication, but which you may find useful in your job, have also been added.	TT 1 to 24
Starting a job from the console. (MFT-MVT)	You can start a job from the console with the START command. You can also stop, modify and cancel a job that was started by the START command or entered through the input stream. See the START command writeup for an explanation.	MFT 23 to 24, MFT 38 to 39, MFT 47 to 53, MVT 23 to 24, MVT 37 to 39, MVT 50 to 58
Telecommunications Access Method (TCAM)	A new chapter "Telecommunications Access Method" (TCAM) has been added to the publication. TCAM is an access method used for communication between terminals and a central computer complex. This information is for planning purposes only.	TCAM 1 to 21
MONITOR command (MFT-MVT)	The new command MONITOR has been added to the system. The MONITOR command will cause a continuous display (The display will be continuously updated.) The parameters JOBNAMES, STATUS, DSNAME, and SPACE have been moved from the DISPLAY command to the MONITOR command. To give you time to get used to the MONITOR command, the four parameters have been left in the DISPLAY command but removed from the DISPLAY command documentation (Release 20 only). So if you enter a DISPLAY STATUS command it will still work, but you should familiarize yourself with the new MONITOR command.	MFT 26 to 28, MFT 40, MFT 52 to 53, MVT 25 to 27, MVT 40 to 41, MVT 57 to 58
Time Sharing Option (TSO) (MVT)	The time sharing option (TSO) has been added to the MVT chapter. TSO parameters have been added to the CANCEL, MODIFY, MONITOR, and START commands. The SEND command has been added so that you can communicate with the TSO terminal users. A discussion on the time sharing option background reader has been added to the section on input readers. This information is for planning purposes only.	MVT 9, MVT 23 to 24, MVT 37 to 41, MVT 46 to 47, MVT 50 to 58
American National Standard labels (PCP, MFT and MVT)	Label information for American National Standard labels has been added to the VOL= parameter of the MOUNT command.	PCP 15, MFT 41, MVT 42, TT 3
IMDPRDMP print dump program. (PCP, MFT and MVT)	Changes have been made to the "Printing the Core Image Dump" section of the "Hardware Debugging Aids" chapter. Use the IEAPRINT dump program for PCP and the IMDPRDMP print dump program for MFT and MVT.	GOT 7 to 9

(Continued)

Item	Description	Areas Affected
Recovery Management Switching (RMS) for the Models 155 and 165. (MFT-MVT)	The MODE command for the Models 155 and 165 has been added to obtain information on the current state of the hardware facilities that attempt recovery from machine check interrupts. The Model 165 information is for planning purposes only.	MFT 32 to 37, MVT 31 to 36, TT 13
Channel Check Handler (CCH) for the Models 155 165, and 195.	The Channel Check Handler (CCH) analyzes channel error information for error recovery procedures. The Model 165 information is for planning purposes only.	GOT 20 to 21, TT 4
System/370 SET command (MFT-MVT)	The System/370 SET command is entered at IPL time as text in the REPLY command. All other times the regular SET command is used.	MFT 5 to 6, MVT 5 to 6
Multiple Console Support (MCS)	Changes have been made to the Multiple Console (MCS) section on how to do an alternate console and varying a console device.	OC 9, OC 11
VARY command	A note has been added to the VARY command about varying a device with a reserved volume mounted on it.	PCP 24, MFT 58, MVT 61 to 63
Writers	You can stop a class A or B writer by starting a new class A or B writer.	PCP 22
Display active	If a job is having devices allocated to it or being terminated, no indication that a task is active will appear in the display of active tasks.	MFT 26
Controlling jobs through HOLD and RELEASE commands	Additional information has been added to the section "How to Control Jobs Through HOLD and RELEASE Commands".	MFT 64, MVT 67
DSNAME parameter	Information has been added to the DSNAME parameter of the DISPLAY command (PCP) and the MONITOR command (MFT and MVT).	PCP 14, MFT 40, MVT 40 to 41
LOG command	Information has been added to the text field description of the LOG command.	MFT 31, MVT 30
STOP command	The parameter INIT.Pn has been added to the STOP command.	MFT 52
Resetting job priority	If you try to reset the priority of a job that is in execution, only the priority of the system output for that job will be reset.	MFT 44
System failures	This is explained in the section "How to Handle System Failures." If a must complete task in MFT has insufficient main storage, system messages may not appear.	GOT 7 to 8

Summary of Major Changes--Release 20.1

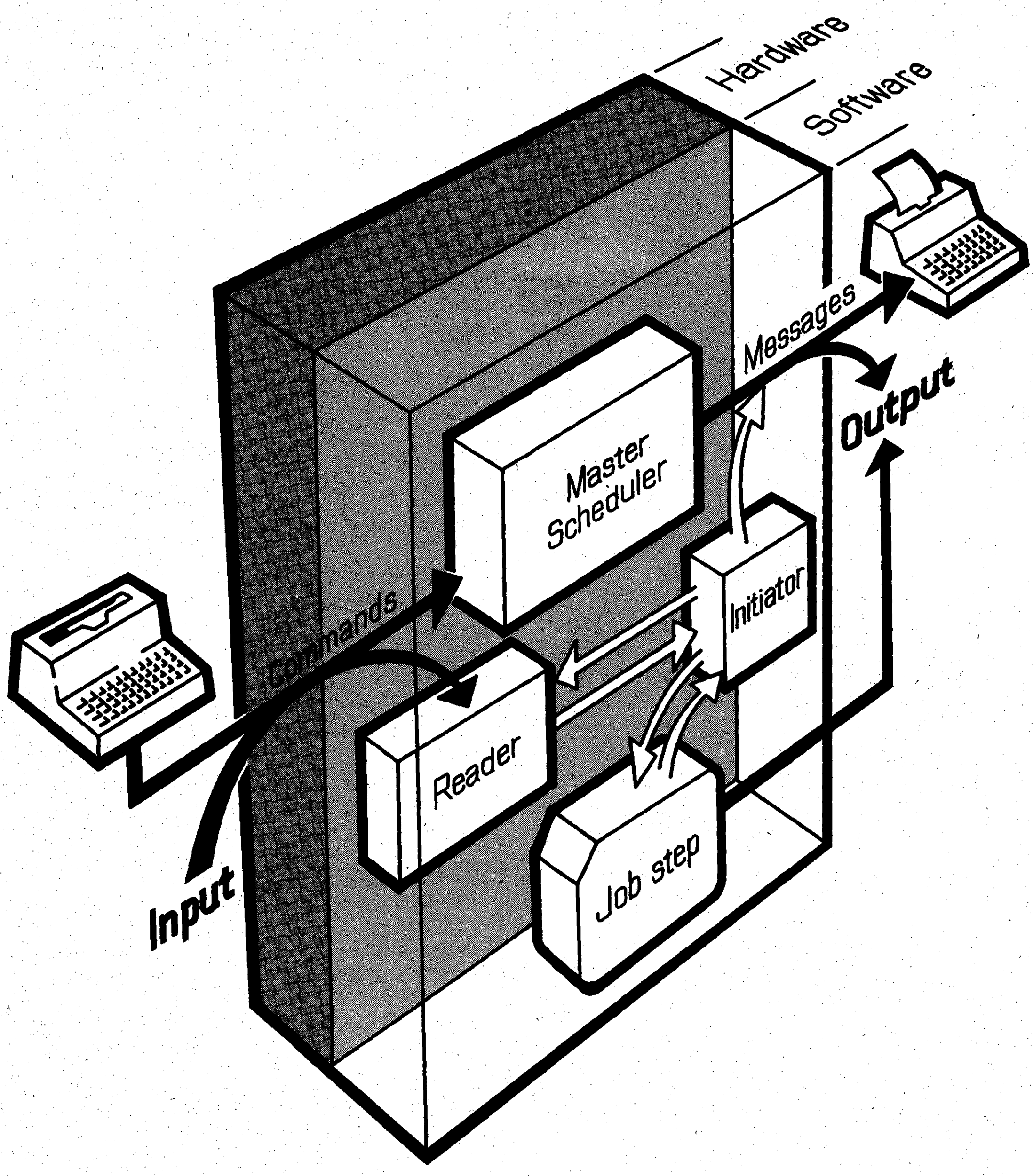
Time Sharing Option (TSO)	The time sharing option (TSO) has been deleted from the MVT chapter and put into a separate TSO chapter. The chapter "Time Sharing Option" discusses how to start, control, modify, and stop TSO.	TSO 1 to 21

CONTENTS

SYSTEMS WITH PCP

PCP

Contents



How the Scheduler Works in Systems With the Primary Control Program

Systems With PCP

In systems with the primary control program, the main parts of the job scheduler are the reader and the initiator.

Control is passed from the reader to the initiator, and then to the job step.

At first, control is given to the reader, which reads in and arranges job control records till data from the current job, or control records for the next job, are found in the input stream.

Control is then passed to the initiator, which assigns devices, starts the job, and passes control to the job step.

When the job or job step ends, control is returned to the reader by the initiator, and the process is repeated.

At any given time, only one of these parts can be in control of the CPU.

Under the primary control program, one job at a time is brought from the input stream into main storage. That job can use all of main storage beyond an area set aside for the control program routines that make up the nucleus of the system.

No other job can be brought into main storage until the first job is ended.

Starting, Stopping, and Restarting the System

System operations are controlled mainly through a console I/O device -- the IBM 1052 Printer Keyboard. You give commands to the system, and receive messages from it, through the console.

There may be one console or two, depending on your installation. If you have two, one of them is called the primary console and the other is called the alternate console.

The primary console is active until you shift to the alternate by hitting the INTERRUPT key on the system control panel. Before shifting, give the VARY command to switch the alternate console offline to avoid conflicts with executing programs.

You can enter most commands through a SYSIN (system input) stream by using command statements. Each command statement, or group of command statements, must precede a JOB, an EXEC, a null, or a delimiter statement. Commands are accepted as soon as they are read.

<u>Composite Console</u>: A composite console is made up of a card reader and a printer. If your installation uses a composite console, it will be your only console -- there will be no alternate.

To give the system a command through a composite console (made up of a card reader and a printer) hit the card reader's STOP key, place the command in the reader, and hit the START and EOF keys.

STARTING THE SYSTEM

Starting the system includes initial program loading (IPL), readying the nucleus, and readying the scheduler.

Initial Program Loading

Initial program loading is a procedure carried out at the beginning of a shift, after a power-on following an electrical shutoff, after malfunctions that require reloading the control program into main storage, after scheduled maintenance, and as part of switching from one system to another.

Begin initial program loading by selecting the direct access storage device on which the operating system resides: set the three LOAD UNIT switches on the control panel to the proper unit address (made up of the channel, control unit, and device numbers); then hit the LOAD key on the panel.

Hitting the LOAD key turns off the MANUAL light, turns on the LOAD light, and starts reading the IPL program from the input device.

After the IPL program is read into lower main storage, control is passed to it, and the LOAD light turns off. If either the reading operation or the passing of control is unsuccessful, the CPU pauses and the LOAD light stays on.

When the IPL program gets control, it loads the nucleus of the control program into main storage.

The IPL program loads a standard, or primary, nucleus unless you cause it to load a secondary nucleus. For a description of this procedure, see "How to Load a Secondary Nucleus" Operator's Procedures, GC28-6692.

After the nucleus is loaded, control is given to a nucleus initialization program (NIP).

If the IPL program does not finish successfully, or if I/O errors occur while NIP is running, the WAIT light turns on and an error code is placed in the low-order 12 bits of the program status word (PSW).

Whenever the WAIT light turns on without a message, display the PSW, note the error code, and follow the instructions for that code given in the publication IBM System/360 Operating System: Messages and Codes.

Readying the Nucleus

The nucleus initialization program (NIP) does general preparatory work for the system. If the communication option was specified at the time the system was generated, you'll receive message IEA101A SPECIFY SYSTEM PARAMETERS, requesting any changes.

If you receive this message, your system programmer may ask you to alter one or more options, such as the BLDL option, the RAM option, or the RSVC option.

Explanations of the various options, and instructions on how to alter the options, are given in the Operator's Procedures, GC28-6692, under the heading "How to Specify System Parameters."

If no changes are to be made, issue REPLY id,'U', or simply signal EOB.

After NIP completes its preparation of the system, it passes control to the master scheduler. You'll receive a READY message from the system and the WAIT light will turn on.

If an error other than an I/O error occurs during the running of NIP, the WAIT light turns on, and you will receive a message identifying the error. No message is sent if the system console is not ready, but a code can be found in the low-order 12 bits of the current PSW as the system waits.

Readying the Scheduler

When initial program loading is complete and the system is ready to run, you will receive a READY message, and the WAIT light will go on. You may then enter commands to start the job scheduler, which in turn begins the flow of work through the system.

Your first command should be a SET command specifying the date.

When the system includes the timer option, the SET command should also give the time of day.

Optionally, SET can specify the addresses of the devices for the input queue and a procedure library (SYS1.PROCLIB), and can also specify input queue formatting.

Normally, the formatting parameter Q=(unitaddr,F) can be left out after the first IPL, causing the scheduler to use the input queue as it was formatted earlier.

START RDR and START WTR must be issued if they have not been specified by your installation at system generation time.

If your installation has already specified START RDR and START WTR, I/O devices are automatically allocated to an input reader and an output writer, and the commands are written out on the console as if you had keyed them in yourself.

If you want to override an automatic START RDR or START WTR command, wait until after the command appears on the console and then enter your own command manually.

You must enter a START command with no parameters as the last command in readying the scheduler.

Examples:

1. To start a system with automatic START RDR and START WTR:

```
SET DATE=yy.ddd,Q=(unitaddr,F)
START
```

2. To start a system and to remove an I/O device from the system before processing:

```
SET DATE=yy.ddd,Q=(unitaddr,F)
START RDR,unitaddr
START WTR,unitaddr
VARY unitaddr,OFFLINE
START
```

3. To start a system with a timer, and to remove two I/O devices before processing:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss, Q=(unitaddr,F)
START RDR,unitaddr
```

```
START WTR,unitaddr
VARY unitaddr,OFFLINE
VARY unitaddr,OFFLINE
START
```

STOPPING THE SYSTEM

Stopping the operating system -- to prepare for turning power off, for example, or for loading another system or an independent utility program -- means taking all jobs out of the input stream, and canceling the job in progress or waiting for it to end.

RESTARTING THE SYSTEM

Follow the same steps you took in starting the system, but leave out the formatting parameter, Q=(unitaddr,F), when you issue the SET command.

Controlling Input and Output

This section describes how to work with the operating system in controlling input and output, particularly in the area of allocating or assigning devices to jobs.

INPUT

A system input (SYSIN) stream is made up of job control language (JCL) statements and problem program data routed by the control program to their destinations within the system.

Sets of input data entering the system through the SYSIN stream are called SYSIN data sets.

Input data can also be read directly by the problem program and not be a part of a SYSIN stream.

Input Reader

An input reader is that part of the scheduler that reads a system input stream from a single device. You assign an input device to an input reader by issuing a START RDR command.

Input is in the form of cards or 80-character unblocked records (card images) on tape.

There is only one input reader in the system, but you can change the source of the SYSIN stream from one unit to another by another START RDR command naming the new unit.

Because systems with the primary control program handle only one job at a time, the input reader will not begin reading from the new unit until the end of the current job.

The processing of JCL statements halts temporarily when a SYSIN data set is met in the input stream. The unit handling the SYSIN stream is then assigned to the job step. The job step reads the data set directly from the unit until a JCL delimiter statement (/*) is met. The delimiter statement marks the end of data for the problem program, separating it from the JCL statements for the next job step.

OUTPUT

A system output (SYSOUT) stream consists of system messages and problem program output data sets routed by the control program to a common output device.

Problem program data sets that leave the system through the output stream are called SYSOUT data sets.

Output data can also be written directly by a problem program and not be a part of a SYSOUT stream.

Output Writers

An output writer is that part of the scheduler that writes a single class of output to a single device.

A maximum of eight output writers can be active at any one time, each writing one class to one device. Each output class is designated by a one-character class name.

Valid class names include the letters A through Z, and the numbers 0 through 9. Class A is the system output (SYSOUT) class. An output device must be active for class A at all times.

An output device is assigned to an output class by issuing a START WTR command with a class name. The programmer must supply DCB information in the processing program or on the DD SYSOUT= statement for that output class. The device is unassigned from the output class by issuing either a STOP WTR command or a START WTR command to a different device for that class.

The starting of a new writer for an output class automatically stops the writer assigned earlier to that device and class.

System messages are written either before the job step is started or after it ends.

When a SYSOUT data set is to be written, the job scheduler allocates the unit handling the SYSOUT stream to the job step, which then writes directly to the unit.

ALLOCATING DEVICES

Device allocation is the assignment of I/O devices for use by a job step or the system.

Device allocations are made in response to requests from three sources:

- Data definition (DD) statements. These may specify the I/O requirements of a job step in the system.
- System generation statements. These may request fixed assignments for system processes, such as the automatic starting of an input reader.
- Operator commands. These may request assignments for system processes, or may modify assignments made when the system was generated.

Assignment by the Scheduler

Through data definition statements, the programmer describes the input/output requirements for the data sets of a job step.

Using this information, the job scheduler allocates I/O devices directly to the job step, attempting to provide overlapped operation, conserve input/output resources, and recognize items that increase input/output efficiency.

Assignment by the Operator

You are responsible for device assignments made for the starting of input readers and output writers. If assignments were specified at system generation time, you don't have to respecify them, unless you wish to make a modification.

All of your assignment requests are made through operator commands.

Your ability to switch input/output devices to online or offline status lets you modify their allocation.

Online devices can be assigned to problem programs, while offline devices cannot be. Your main reason for placing a device offline is to reserve it for preventive maintenance.

To switch a device's status, enter a VARY command indicating the device and the desired status.

By entering the MOUNT command, you can cause an input/output device to be assigned to those job steps that require the particular volume mounted on it.

For example, you might give a MOUNT command when you know that a volume will soon be used by many independent jobs.

In systems with the automatic volume recognition option, you can mount volumes on online devices that are not ready, thus anticipating the later needs of jobs you are scheduling.

Volume Mounting: In most installations, your role with respect to I/O devices is to mount and demount volumes.

The job scheduler, using information from data definition statements, determines the input/output resources to be assigned to a job and the volumes that are required. If these volumes are not mounted, the job scheduler writes you a mounting message.

Each message states that either a specific volume or a scratch volume is to be mounted. Mount the requested volume and hit the START key on the device to continue processing.

Never mount a blank tape volume, unless specifically directed to do so.

The system checks for a tape label and the absence of data causes the whole volume to be scanned for a data record. If an unlabeled tape is required, a tape mark should be written to avoid unnecessary scanning.

For a description of a program you can use to create standard labels on tapes, see the section on the program IEHINITT in the publication IBM System/360 Operating System: Utilities.

After you mount the volume and ready the drive, the system reads the volume label. If an incorrect volume is mounted, the system repeats the mounting message and unloads the incorrect volume, if possible (some device types, such as the 2311, can only be unloaded manually).

If a scratch tape volume is incorrect only because the job specified a nonstandard or unlabeled tape and you have mounted a standard-labeled tape, the system will rewrite the tape label to conform to job specifications.

If a scratch tape is incorrect only because the job specified a labeled tape and you have mounted one without a label, you will receive message IEC704A L ddd.

If you want the system to write a new volume label for the tape respond REPLY id,'ser,owner' to define the volume serial number and, optionally, the owner's name for the new label.

If you want to use a different tape, respond REPLY id,'M' to cause the system to unload the mounted tape and repeat the mounting message. You can then mount a different tape.

Note: You will receive a message describing any changes the system makes to the volume label; note these changes on the tape reel's external label.

If a request was made for a tape volume without a standard label and if the volume mounted does not have a standard label, that volume will be accepted. The volume is treated as unlabeled, or as a volume labeled with nonstandard labels, according to the DD statement.

The following volume mounting options can be selected at system generation time:

- Imperative mount. Mounting messages are written when the job step requiring the volumes is started.
- Automatic volume recognition. You take the initiative and mount labeled volumes on any unused drives. The system recognizes and remembers these volumes and assigns the drives to later job steps.

If your system has the automatic volume recognition feature, mount volumes you want the system to find for the first job at IPL before entering the START command.

Also before entering START, be sure that all offline devices are known to the system by using the VARY OFFLINE command.

After the first job, you can mount ahead for several jobs at a time.

In addition, the system may ask you to mount other volumes, and you can mount these on any appropriate online devices that are not ready. (Do not unload any devices -- you can only mount on devices unloaded by the system.)

Automatic volume recognition handles nine-track tape, seven-track tape, and 2311 and 2314 direct access devices. The density for seven-track tape is set at 200, 556, or 800 bytes per inch at the time the system is generated.

When volumes are to be demounted, the system unloads the devices, if possible, and writes you messages identifying the volumes being unloaded.

When you receive a mounting message for a 2321 data cell, hit the RESET button on the device if the requested cell is already positioned properly. If you have to open the door on the unit to position the cell, then you don't have to hit RESET -- closing the door performs the same function.

Occasionally you may be asked by your installation not to mount volumes on certain devices, and not to make those devices ready, because you are running a version of the operating system that was generated for a slightly different set of devices.

Allocation Guidelines

When the scheduler cannot satisfy requests for allocation from available (online) devices, it sends you a message and a list of offline devices. You then either cancel the job or make an offline device available by replying to the message with a three-character unit address.

Use the MOUNT command to reserve a volume on a device, when you know that several jobs are going to need that volume.

Volumes reserved through a MOUNT command are not demounted by the system until an UNLOAD command is given, causing the system to unload the volume.

Unit Addresses: When referring to I/O devices in the unitaddr parameters of operator commands, you must use the unique unit address assigned to each device.

Group Names: Symbolic group names of one to eight alphameric characters may be defined by your installation, but these are for use by your programmers in their data definition statements. Don't use symbolic group names in operator commands.

Shared DASD Option: If your installation is using a system with this option, be sure to read the section "How to Use the Shared DASD Option" in the chapter "General Operating Techniques."

Restarting a Job

A job may be abnormally ended as a result of a hardware, programming, or system error. Such an ending may occur at any time during program execution.

Valuable machine time would be lost if an abnormal end occurred during the processing of one of the last job steps of a multistep program or in the middle of a long job step and execution of the program had to be started again at the beginning of the first job step. The checkpoint/restart feature of the system is provided to allow a restart on an abnormally ended job either at the beginning of a job step or at a checkpoint within this step.

Your programmer determines whether an automatic restart or a deferred restart is to be performed.

AUTOMATIC RESTART

If your programmer provides for an automatic restart and the job does abnormally end, you will receive system message IEF225D (SHOULD job RESTART) requesting a reply to authorize this restart. This allows you to prevent repeated restarts at the same checkpoint or job step.

When you are requested to authorize an automatic restart, you can reply YES or NO.

- Reply YES if a restart at a specific checkpoint or job step is to be performed for the first time.

When you authorize a restart, the system must again process the JCL statements referring to the job that has been abnormally ended. (For procedural details, see "How to Restart a Job" in the Operator's Procedures, GC28-6692.) If these JCL statements are punched into cards, you must, before replying YES, return all cards in the input stream to the hopper. If the JCL statements are on tape, your system retrieves them automatically.

PCP

Also if the failing job step used one or more card input data sets, you must return, to the appropriate hoppers, all data cards that may have been read during this job step.

The system may request you to exchange data volumes on the various devices that have been allocated to the job step being restarted.

- Reply NO if a restart at a specific checkpoint or job step has been requested repeatedly. When your reply is NO and your programmer wants a restart to be performed, he must resubmit the job for a deferred restart.

DEFERRED RESTART

If your programmer provides for a deferred restart and his job does abnormally end, he must resubmit the job to have this restart performed.

To restart the job, your programmer must provide a restart deck for submission to the system through the system input reader. The JCL statements to be included in this restart deck are described in detail in the publication IBM System/360 Operating System: Job Control Language Reference.

The device configuration of your system at the time of restart need not be the same as it was when the job was abnormally ended. However, enough devices must be available to satisfy the needs of the job step being restarted. In addition, when the restart does not involve initial program loading, all volumes containing system data sets must be mounted on the same devices as they were mounted originally.

The system may request you to mount data volumes other than those required at the beginning of the job. In addition, any card input data sets that have been used by the failing job step must again be made available to the system.

Operator Commands

This section contains a description of the commands you use to control the operating system. The commands are given in alphabetical order.

You can use abbreviations as well as full command names when entering commands. The usable names and abbreviations are:

*CANCEL	C	*SET	T
DISPLAY	D	START	S
MOUNT	M	STOP	P
*REPLY	R	UNLOAD	U
+*REQ		VARY	V

* These commands cannot be entered in the input stream.
\+ The REQ command cannot be abbreviated.

Console commands, other than SET and START, are accepted whenever you enter them.

To enter a SET or START command, you must get a READY message.

A REQ command will give you a READY message. It makes the system pause and issue a message requesting further commands at the end of a job step. Any commands entered in response to such a message must be followed by a START command with no parameters. This command gives control to the reader to start the next job step.

Be sure to use the correct abbreviations for operator commands. For example, use S for START and T for SET. If you inadvertently key in S for SET, the system issues an error message and waits for you to enter a SET command.

The following conventions are used in illustrating the format of commands:

- Required letters (those shown in upper case) must be entered, but can be entered in either upper or lower case.
- Lower-case letters indicate that a parameter must be substituted.
- Dotted lines ... (indicating a series of terms), brackets [ ], and braces { } are not entered.
- Entries within brackets [ ] are optional.
- Entries within braces { } are required - you must select one.
- Numbers and punctuation marks (other than dotted lines, brackets, and braces) must be entered as shown.

Command formats are essentially free form, but one or more blanks must follow the operation field.

Commands cannot occupy more than one line. For example, if a command is entered through a card reader, it may not be more than 80 characters in length.

If comments on commands are necessary, they must appear to the right of the operand field and be separated from it by at least one blank. If the operand field is null, a comma followed by at least one blank indicates that comments will follow.

PCP

CANCEL -- Terminate Job Immediately

Use the CANCEL command to immediately terminate the scheduling or execution of a job. This command cannot be entered into the input stream.

Optionally, you may request that an abnormal-end-of-task storage dump be taken if the command is received while the job is running.

This command is always executed as soon as it is received.

If you enter a CANCEL command for a job that is neither running nor in the process of being scheduled, you will be informed that the command cannot be executed.

You may be asked by your programmers not to use the CANCEL command on certain jobs. These jobs alter data sets containing information vital to the system -- canceling the jobs might make the data unusable.

Operation	Operand
{CANCEL / C}	jobname[,DUMP]

jobname
: the name of the job to be terminated. The maximum length of a job name is eight characters.

DUMP
: an abnormal-end-of-task storage dump is to be taken if a step of the job is being executed when the command is received. If the programmer has put in the SYSABEND data definition statement, a full dump is taken. If he has not included this card, an indicative (partial) dump is taken.

DISPLAY -- Cause Console Display

Use the DISPLAY command to cause a console display of the name of each job at the time it is initiated, and at the time it is terminated. (If a job is terminated due to unusual circumstances, you'll receive a message even if you have not used the DISPLAY command.)

This command provides you with job name information needed for effective use of the CANCEL command and, together with system messages, keeps you informed of which job is currently being executed.

This command provides, via mount and K(keep)-type demount messages, the names of non-temporary data sets. It also provides, via demount messages, the available space on a direct access device.

Operation	Operand
{DISPLAY / D}	{JOBNAMES[,T] / STATUS / DSNAME / SPACE}

JOBNAMES[,T]

the name of each job is to be displayed both when the job starts and when it ends, and that unit record allocation is to be displayed when the step starts.

If a job terminates abnormally, the job name will appear in the diagnostic message.

If the T parameter is used in conjunction with the JOBNAMES parameter, the system displays the time of the day in addition to the jobnames. The time is shown in the format

```
hh.mm.ss.
|  |  |
|  |  └--Seconds(00-59)
|  └-----Minutes(00-59)
└--------Hours  (00-23)
```

In systems without the timer option, the T parameter is ignored.

STATUS

the data set names and volume serial numbers of data sets with dispositions of KEEP, CATLG, or UNCATLG are to be displayed on the console at step termination and job termination.

DSNAME

the system is to display, within the mount and K (keep)-type demount messages, the name of the first non-temporary data set allocated to the volume to which the messages refer. Mount messages for data sets with a disposition of DELETE will not contain the data set name.

SPACE

the system is to display, in demount messages, the available space on a direct access volume.

MOUNT -- Allocate Device

PCP

Use the MOUNT command to allow allocation of an input/output device to all job steps that require a particular volume, without intervening demountings and remountings of that volume. The volume must be removable.

The required volume should be mounted immediately after the MOUNT command is issued.

Operation	Operand
{MOUNT / M}	unitaddr[,VOL=(NL,serial)] [,USE={STORAGE / PUBLIC / PRIVATE}]

unitaddr
: the address of the input/output device to be allocated. Unitaddr must specify a device that has been unloaded by the system. When issuing this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unitaddr for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

VOL=(NL,serial)
: the volume does not have standard labels. The serial number, up to six characters long, is used for allocation references. This parameter must not be used for direct access volumes, but must be used for nonlabeled tapes. If you do not specify both parameters, NL and the serial number, the system assumes the volume is labeled. The system determines, by examining the label of the volume you mount, whether the label is standard or American National Standard. The volume's label type is included in message IEF279I in the form of SL for standard type or AL for American National Standard type.

USE=STORAGE or PUBLIC or PRIVATE
: a direct access volume will be used as a storage volume, public volume, or a private volume. If this operand is not used, the system treats the volume as a private volume.

: A storage volume is the most freely allocated kind of volume, open to use by the largest variety of data sets, temporary or non-temporary. Slightly restricted is a public volume, which can be allocated freely for temporary data sets, but which must be specified by volume serial number to be allocated to non-temporary data sets. A private volume is the least freely allocated kind of volume -- it is allocated only if its volume serial number is specified.

REPLY -- Reply to Information Request

Use the REPLY command to reply to messages from the operating system and from problem programs that request information. This command cannot be entered into the input stream.

The message requesting the reply must be directly followed by the REPLY command. No other messages are printed until the reply has been entered.

Operation	Operand
{REPLY / R}	id,'text'

id
: the 2-character reply identification field of the message requesting the reply.

text
: the text to be entered in response to a message. The information passed to the program expecting the reply does not include the enclosing apostrophes. When using the REPLY command to answer system messages, it is always correct to use upper case letters in the text.

REQ -- Request Commands

Use the REQ command to cause the system to tell you when you can enter commands. This is indicated by a READY message from the master scheduler. The REQ command cannot be entered into the input stream.

Other than at IPL-time, you must issue a REQ command, and wait for a READY message from the master scheduler, before entering a START or SET command.

Operation	Operand
REQ	

SET -- Set Date, Time, and Location

Use the SET command to establish the date, the time of day, the device for the input work queue and whether the queue is to be formatted, or the location of the procedure library. This command is also used to specify the device on which the accounting data set (SYS1.ACCT) resides. Any combination of the above may be specified.

The SET command cannot be entered into the input stream.

The SET command can be entered only after you have received a READY message.

The first SET command after initial program loading must always include the DATE operand.

In systems that don't include the timer option, the CLOCK parameter is ignored.

Operation	Operand
{SET} {T}	DATE=yy.ddd[,CLOCK=hh.mm.ss][,Q=([unitaddr][,F])] [,PROC=unitaddr][,ACCT=([unitaddr][,N])]

DATE=yy.ddd
the date in the following format:

```
yy.ddd
|   |
|   └----Day of year  (001-366)
└--------Year         (00-99)
```

CLOCK=hh.mm.ss
the time of day in the following format:

```
hh.mm.ss
|  |  |
|  |  └--Seconds      (00-59)
|  └-----Minutes      (00-59)
└--------Hours  (00-23)
```

If the new clock setting implies a change of date, the new date must be explicitly stated using the DATE parameter.

Q=(unitaddr) or (unitaddr,F) or (,F)
(1) the address of the direct access device (other than an IBM 2321) on which the volume containing the input work queue (SYS1.SYSJOBQE) resides or (2) that the system is to format the input work queue, or both. This parameter is used only in the first SET command after IPL. Note: If you are specifying only unitaddr, you do not need parentheses.

Space for the input work queue must have already been allocated on the volume which is mounted on the specified device.

You need not specify a 3-character unit address if one of the following conditions exists:

- The SYS1.SYSJOBQE data set is cataloged.
- The SYS1.SYSJOBQE data set is contained on the system residence volume.

If the system is to format the input work queue prior to the first job initiation, you must specify ,F following or without the 3-character unit address. For example, Q=(191,F) specifies that the system is to format the input work queue on the volume residing on the direct access device with an address of 191; Q=(,F) specifies that the system is to format the input work queue either on the volume to which the SYS1.SYSJOBQE data set is cataloged or on the system residence volume.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.SYSJOBQE data set. If this data set is not cataloged, the system assumes that the data set is contained on the system residence volume.

The system issues an error message when either of the following conditions exists:

1. The volume to which the SYS1.SYSJOBQE data set is cataloged is not mounted.

2. The system cannot locate the SYS1.SYSJOBQE data set on the selected volume.

PROC=unitaddr
: the address of the direct access device on which the volume that contains the procedure library resides.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.PROCLIB data set. If this data set is not cataloged, the system assumes that the data set is contained on the system residence volume.

The system issues an error message if either of the following conditions exists:

1. The volume to which the SYS1.PROCLIB data set is cataloged is not mounted.

2. The system cannot locate the SYS1.PROCLIB data set on the selected volume.

This parameter is used only in the initial SET command issued immediately after IPL and should only specify a device that is <u>ready</u>.

ACCT=([unitaddr][,N])
: unitaddr specifies the address of the direct access device on which the SYS1.ACCT data set resides; N specifies that accounting data is to be written starting at the beginning of the SYS1.ACCT extent.

If you omit the ACCT parameter, the system assumes that the SYS1.ACCT data set is contained on the system residence volume.

If you omit N, the system attempts to write accounting data from the end of the last accounting record written into the SYS1.ACCT data set.

START -- Start System Process

Use the START command to start an input reader or output writer.

The START command can be issued only after you have received a READY message.

Any sequence of commands submitted in response to a READY message must be followed by a START command with no parameters.

If the output writer is associated with a tape unit, the volume mounted must have a standard label.

Positional Parameters: Positional parameters, such as unitaddr, volumeserial, and parmvalue in the START command, must be entered exactly in the order in which they appear. If you leave one out, put a comma in its place. You do not need to use a replacing comma if:

- The parameter to be left out is the last one in the series.
- All positional parameters following the absent one are also to be left out.
- All optional positional parameters, such as volumeserial and parmvalue, are to be left out. Example:

```
START RDR,284,FILESEQ=2
```

Keyword Parameters: Keyword parameters, such as DSN and FILESEQ, can appear in any order. To leave one out, simply omit it; do not replace a missing keyword parameter with a comma.

Operation	Operand
{START / S}	{RDR / WTR},unitaddr[,volumeserial][,parmvalue] [,DSN=datasetname][,FILESEQ=filesequencenumber]

RDR
: an input reader is to be started.

WTR
: an output writer is to be started.

unitaddr
: specifies the address of the unit record or magnetic tape input/output device associated with the input reader or output writer that is to be started.

volumeserial
: the serial number, up to six characters long, of a magnetic tape volume. If this parameter is specified, label checking is performed.

  If you do not use this parameter but specify parmvalue, you must indicate the absence of volumeserial by a comma. Example:

```
START RDR,282,,JOBX,DSN=YRTODATE
```

PCP

parmvalue
either an up-to-eight-character name of a job in the input stream or a one-character alphameric output class name. A job name is used only with RDR: when starting an input reader, giving the job name causes forward spacing through the input stream until the named job is found. An output class name is used only with WTR: giving a class name indicates the class of output the writer is to handle; if no class is specified, the writer is assigned to class A.

If you specify parmvalue but not volumeserial, you must indicate the absence of volumeserial by a comma. Example:

```
START WTR,282,,A,DSN=YRTODATE
```

DSN=data set name
the name of the data set associated with the input reader or output writer. The maximum length of a data set name is 44 characters. If this parameter is not specified, the data set name SYSIN is assumed for the reader.

FILESEQ=file sequence number
the file sequence number, up to four digits long, of a data set on a magnetic tape volume. This parameter is optional and is used only with RDR when unitaddr designates a magnetic tape device.

STOP -- Stop System Process

Use the STOP command to stop an output writer or to stop a console display of job names, data set names, or the available space on a direct access volume.

Operation	Operand
{STOP, P}	{JOBNAMES, WTR,unitaddr, STATUS, DSNAME, SPACE}

JOBNAMES
: a console display of the names of jobs, initiated by the JOBNAMES parameter of the DISPLAY command, is to be terminated. For more information about JOBNAMES, see the discussion of the DISPLAY command.

WTR,unitaddr
: the output writer using the addressed unit is to be stopped by the system. This operand will not stop a writer assigned to class A or B -- only a new START WTR to class A or B will cause the presently active one to stop. The new START WTR command for class A or B may address the same unitname used for the active writer being stopped.

STATUS
: the system is to stop the console display at step termination and job termination of the names and volume serial numbers of data sets with dispositions of KEEP, CATLG, or UNCATLG.

DSNAME
: the system is to stop the display of the names of non-temporary data sets as initiated by the DSNAME parameter of the DISPLAY command. For more information, see the discussion of the DISPLAY command.

SPACE
: the system is to stop displaying, in demount messages, the available space on a direct access volume. (The display was initiated by the SPACE parameter of the DISPLAY command.)

PCP

UNLOAD -- Prepare Volume for Demounting

You normally use the UNLOAD command to remove a volume previously mounted in response to a MOUNT command, but you can use UNLOAD to remove any tape or removable direct access volume.

The UNLOAD command causes a volume on an input/output device to be prepared for demounting.

When the volume is ready to be demounted, you'll receive a message. (The message may not be received until the current job is completed.)

Operation	Operand
{UNLOAD / U}	unitaddr

unitaddr
: the unit address of the input/output device to be prepared for demounting. When entering this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unitaddr for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

VARY -- Vary Status of Device

Use the VARY command to place an input/output device (other than a communication line) into an online or offline status.

Operation	Operand
{VARY / V}	unitaddr{,ONLINE / ,OFFLINE}

unitaddr
: the unit address of the input/output device whose status is to be changed. To vary the status of an <u>entire</u> 2321 data cell, use its three-character unit address -- 263, for example. To vary the status of a particular 2321 <u>bin</u>, use its five-character unit address -- 263/8, for example, where 8 is the number of the particular bin being addressed.

ONLINE
: the device is to be made available for allocation by the job scheduler to problem programs. If a device is made ONLINE from an OFFLINE status, and you want the system to recognize a volume that was mounted on the device when you took it OFFLINE, use the MOUNT command to identify the volume to the system.

OFFLINE
: the device is to be removed from the recognition of the job scheduler, and that any further allocation of the device to problem programs is to be prevented. If the device is in use (allocated to a problem program or to an input reader or output writer), the status is not changed until the current user is finished with the device. When the status is changed to offline, you will receive a message.

A device can be removed from the offline status by a subsequent VARY command or, if an appropriate system message is received, by issuing REPLY id,'unitaddr'.

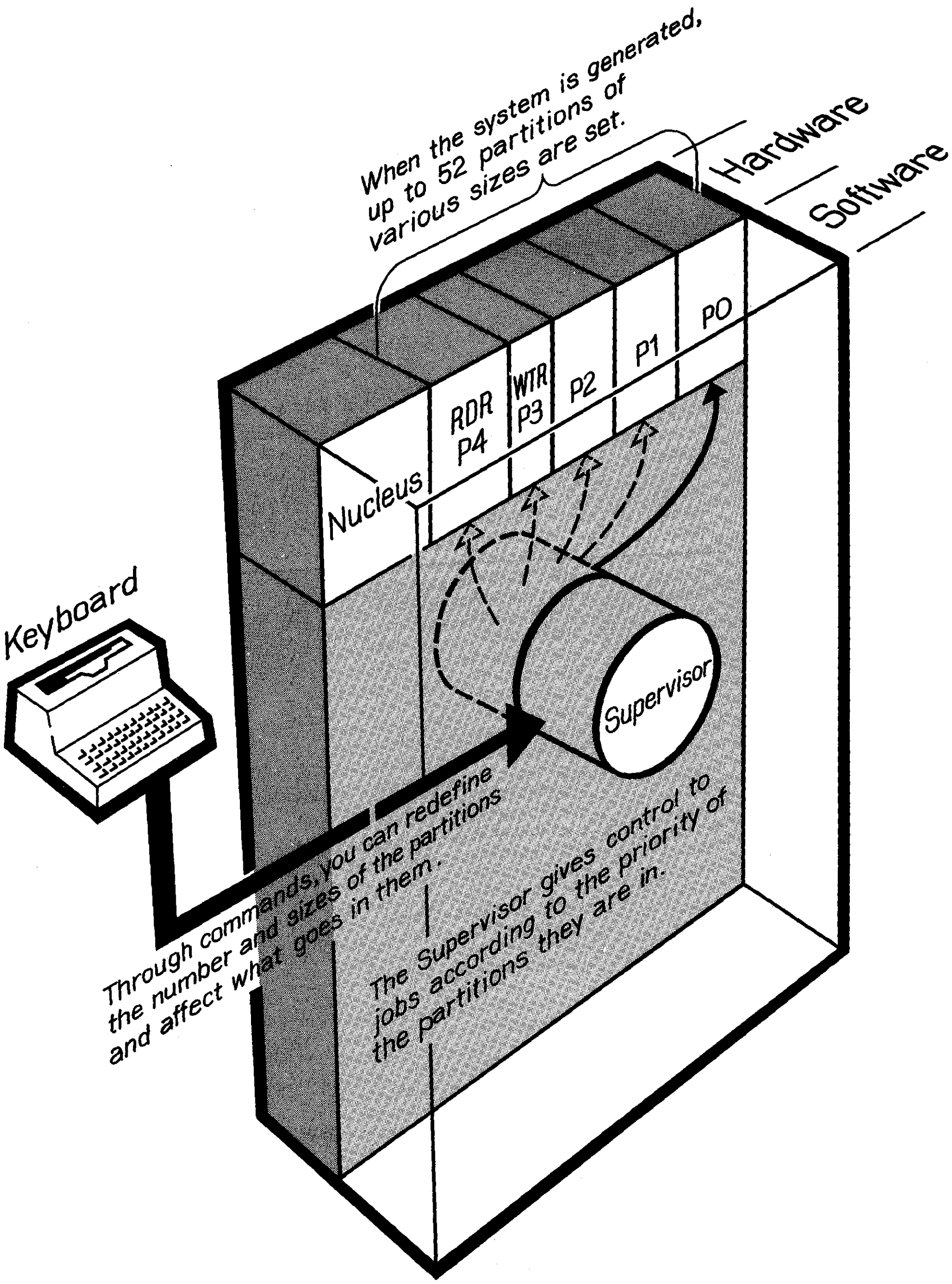
<u>Note</u>: A volume that has been reserved through use of either a PRESRES entry or a user-issued MOUNT command cannot be moved or removed from the system by varying offline the device on which it resides. A reserved volume can be moved or removed only if you issue an UNLOAD command, or if the system issues an action-type message (such as IEF234A), instructing that the volume be demounted.

Index

SYSTEMS WITH MFT

Contents

MFT



How Main Storage is Partitioned in Systems With MFT

Systems With MFT

MFT

In systems with MFT, main storage beyond the nucleus area is separated into from 1 to 52 areas called partitions.

These areas may consist of as many as 15 problem program partitions, as many as 3 resident input reader partitions, and as many as 36 resident output writer partitions, providing the total number of partitions does not exceed 52.

The number and size of these partitions are set when the system is generated, but you can redefine the sizes and reduce the number of partitions when you start the system or while the system is running.

Jobs, read in by system input readers, are placed in input work queues according to CLASS and PRTY parameters, specified on the JOB statement. MFT schedules jobs independently for each partition.

Each partition accepts jobs from as many as three input work queues.

The partition at the end of main storage with the highest addresses is called P0 and is the highest-priority partition. Any job read into P0 is given top priority whenever the system must decide which job in main storage is to receive control next.

When P0 enters a wait state, P1 is given control. If P0 and P1 are both in a wait state, P2 is given control. In a system which contains 15 problem program partitions, P14 is not given control unless all of the higher priority partitions are unable to execute.

Starting, Stopping, and Restarting the System

System operations are controlled mainly through a console I/O device. You give commands to the system, and receive messages from it, through the console. For a discussion on the various types of console configurations see the chapter "Operator's Consoles."

STARTING THE SYSTEM

Starting the system includes initial program loading (IPL), readying the nucleus, and readying the scheduler.

Initial Program Loading

Initial program loading is a procedure carried out at the beginning of a shift, after a power-on following an electrical shutoff, after malfunctions that require reloading the control program into main storage, after scheduled maintenance, and as part of switching from one system to another.

Begin initial program loading by selecting the direct access storage device on which the operating system resides: set the three LOAD UNIT switches on the control panel to the proper unit address (made up of the channel, control unit, and device numbers); then hit the LOAD key on the panel.

Hitting the LOAD key turns off the MANUAL light, turns on the LOAD light, and starts reading the IPL program from the input device.

After the IPL program is read into lower main storage, control is passed to it, and the LOAD light turns off. If either the reading operation or the passing of control is unsuccessful, the CPU pauses and the LOAD light stays on.

When the IPL program gets control, it loads the nucleus of the control program into main storage.

The IPL program loads a standard, or primary, nucleus unless you cause it to load a secondary nucleus. For a description of this procedure, see "How to Load a Secondary Nucleus" in Operator's Procedures, GC28-6692.

After the nucleus is loaded, control is given to a nucleus initialization program (NIP).

If the IPL program does not finish successfully, or if I/O errors occur while NIP is running, the WAIT light turns on and an error code is placed in the low-order 12 bits of the program status word (PSW).

Whenever the WAIT light turns on without a message, display the PSW, note the error code, and follow the instructions for that code given in the publication IBM System/360 Operating System: Messages and Codes.

Readying the Nucleus

The nucleus initialization program (NIP) does general preparatory work for the system. If the communication option was specified at the time the system was generated, you'll receive a message, IEA101A SPECIFY SYSTEM PARAMETERS, requesting any changes.

If you receive this message, your system programmer may ask you to alter one or more options, such as the BLDL option, the RAM option, the RSVC option, or the SQS option. Instructions on how to reply to this message are given in the Operator's Procedures, GC28-6692, under the heading "How to Specify System Parameters."

If no changes are to be made, enter REPLY id'U', or signal EOB.

You will then receive the message:

IEE801D CHANGE PARTITIONS - REPLY YES/NO (,LIST)

Instructions on how to reply to this message are given in the Operator's Procedures, GC28-6692, under the heading "How to Change Partitions." After you have replied to message IEE801D you will receive the READY message and the WAIT light will turn on when the system is ready to run.

Readying the Scheduler

System/360 SET Command: When the nucleus is initialized and the system is ready to run, the primary console (or the MCS master console) receives a READY message, and the WAIT light on the system control panel comes on. You can then enter commands to start the job scheduler, which in turn begins the flow of work through the system.

Your first command should be a SET command specifying the date. When the system includes the timer option, the SET command should also give the time of day. Optionally, SET can specify the names of the devices for the input queue and a procedure library (SYS1.PROCLIB), and can also specify input queue formatting.

Normally, the formatting parameter Q=(unitaddr,F) can be left out after the first IPL, causing the scheduler to use the input queue as it was formatted earlier.

You will receive message IEF423A SPECIFY JOB QUEUE PARAMETERS after you enter the SET command with the formatting parameter Q=(unitaddr,F). Instructions telling how to answer this message are given under the heading "How to Specify Job Queue Parameters" in Operator's Procedures, GC28-6692.

System/370 SET Command: When the nucleus is initialized and the system is ready to run, you will receive the READY message, and the WAIT light on the system control panel comes on. After the READY message you will receive one of two messages:

IEE114A DATE=yy.ddd, clock=hh.mm.ss REPLY WITH SET PARAMETERS OR U

or

IEE116A TOD CLOCK INVALID - REPLY WITH SET PARAMETERS

Message IEE114A displays the date and time of day as they are in storage at the time of the message. If the date and time are satisfactory and no other SET command parameters (Q, PROC, and AUTO) are to be specified, enter r id,'U' , where r is the REPLY command, id is the message identifier, and ,'U' means there are no changes.

If the date or time is to be changed or if additional parameters are to be entered, reply with the information to be changed as text in the REPLY command. For example, to change the time to seven-thirty P.M., enter r id,'CLOCK=19.30.00'.

If you receive message IEE116A, your reply must include at least the date; you may enter the time and other SET command parameters at your discretion.

You must depress the Time of Day Clock Enable Switch on the control panel immediately after entering the date or time in response to either message, or the clock cannot be set with the new information. If you do not depress the switch within approximately 15 seconds after you enter the REPLY command, you will receive the message:

IEE117A INTERVENTION REQUIRED ON TOD CLOCK ENABLE SWITCH

The message will be repeated approximately one minute later if the Time of Day Enable Switch has not been depressed. If, approximately one minute after you receive the message the second time, you have not depressed the switch you will receive the following messages:

IEE119A SET PARAMETER(S) NOT ACCEPTED - ENABLE SWITCH NOT DEPRESSED
IEE112I RESPECIFY SET PARAMETERS TO INITIALIZE SYSTEM
IEE114A or message IEE116A

After receiving the above messages you must enter the REPLY command again. This time be sure to depress the Time of Day Enable Switch before the 2 minute and 15 second time limit elapses.

If you are setting the Time of Day Clock at a time other than IPL time, do not use the REPLY command; use the SET command. When you enter the SET command to set the Time of Day Clock, you have approximately 2 minutes and 15 seconds to depress the Time of Day Enable Switch. If the allotted time elapses and you have not depressed the switch, you will receive the message:

IEE119I SET PARAMETER(S) NOT ACCEPTED - ENABLE SWITCH NOT DEPRESSED

You must now enter the SET command again and don't forget to depress the Time of Day Enable Switch before the 2 minutes and 15 seconds is up.

When the Time of Day Clock has been successfully set, you will receive the message:

IEE118I SET PARAMETER(S) ACCEPTED

START Commands For All Systems: The START RDR, START WTR, and START INIT commands must be entered if automatic START commands have not been specified by your installation at system generation.

Automatic START Commands: If your installation has specified automatic START commands, I/O devices are automatically allocated to an input reader and an output writer.

The automatic START commands will appear on the console in the following format (assuming that the input device is a card reader with the unit address 00C and the output device is a printer with the unit address 00E):

```
IEE103I START WTR.Pn,00E *
IEE103I START RDR.S,00C *
IEE103I START INIT.ALL *
```

If you want to override automatic START commands, use the SET command with the AUTO operand.

Output Writer: The automatic START command for the system output writer will have the partition number of the highest priority partition which is defined as a writer placed in the command (Pn will be replaced with the partition number by the master scheduler).

If a partition has not been defined as a writer, no automatic START commands can be issued by the system.

If you don't want to start a writer in that partition, you must override the command and then issue a START command specifying the partition number in which you want the writer started. See Example 4.

Input Reader: The automatic START command for the system input reader will start a system-assigned transient reader in the first available (highest priority) problem program partition.

If your system configuration includes a partition which has been defined as a resident reader partition, you will probably want to override the automatic command and issue a START command for the resident reader partition (the command specifying a transient reader will not start a reader in the resident reader partition). See Example 5 for the command sequence to start a resident reader.

MFT

Initiator: The automatic START command for the initiator will cause an initiator to be started in all large (scheduler-size) problem program partitions.

If you want to start the initiator only in selected partitions, override this automatic command and enter a START INIT.Pn command for each selected partition.

DISPLAY JOBNAMES: The DISPLAY command with the JOBNAMES parameter should be used as one of the initialization commands. If it is not used, you will have great difficulty in reconstructing the day's work from console messages to find out which jobs are active at any given time.

In addition, if you use DISPLAY JOBNAMES, you'll have the job names available if you have to cancel any jobs.

Examples:

1. To start a system that has automatic START reader, START writer, and START INIT commands (the CLOCK operand in each example is optional):

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss
DISPLAY JOBNAMES
```

2. To start a system that has automatic commands, but to suppress them:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss,AUTO=NONE
DISPLAY JOBNAMES
START WTR.identifier
START RDR.identifier
START INIT.P2,,,ABC
```

3. To start a system that does not have automatic commands, and to remove an I/O device from the system before processing:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss
VARY unitaddr,OFFLINE
START writer.identifier
START reader.identifier
START INIT.P0,,,A
START INIT.P2
```

4. To start a system that has automatic START WTR, START RDR and START INIT commands, but to start a system output writer in a lower priority partition than that specified in the command:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss,AUTO=NYY
START WTR.P8,00E
```

5. To start a system that has automatic START WTR, START RDR, and START INIT commands, but to start a resident reader in partition 0, and start initiators in partitions 1, 3, and 5 only.

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss,AUTO=YNN
START RDR.P0,00C
START INIT.P1
START INIT.P3
START INIT.P5
```

Planning the Work for Each Partition

Through careful use of job classes your installation can achieve optimum balance between its resource requirements and the utilization of those resources.

The CLASS parameter on the JOB statement informs the system of the non-conflicting nature of jobs, permitting them to be scheduled for concurrent execution.

RESPONDING TO A PROBLEM PROGRAM FAILURE

When message IEA029I jobname stepname TASK REINSTATEMENT FAILED appears on the console with names other than MASTER SCHEDULER in the jobname stepname fields, a problem program has failed, but other tasks that are already scheduled will try to reach normal termination. Return the job to the programmer. Do not reallocate main storage, I/O devices, and data sets used by the failing task.

The failing problem program task may or may not have any impact on system operation. If, for example, a program that used a great deal of storage were to fail, this storage would not be released and might impact the system by severely limiting the amount of main storage available to new jobs. Use DISPLAY A to verify that the system is functioning normally. If system operation is impacted, you should quiesce the system (allow the system to come to a stop) by taking the following actions:

- Enter a HOLD Q command to stop the scheduling of jobs.
- Stop all readers and writers.
- After the scheduled tasks have completed, restart the system.

STOPPING THE SYSTEM

Enter a STOP INIT command for each initiator in the system, a CANCEL command (optional) for each job, a STOP RDR for each reader, a STOP WTR for each output writer, and a STOP GFX for the graphics interface task (if the task was initiated).

A STOP GFX command will not take effect until all units assigned to the graphic job processor or satellite graphic job processor are no longer in use.

After these commands, wait for all activity to cease and then enter the HALT EOD command to preserve the status of the system log data sets (if present) and to move internal data from main storage to the SYS1.LOGREC data set. The system will send you a message when the data has been moved.

RESTARTING THE SYSTEM

To restart the system after a failure, follow the same steps you took in starting the system, but when entering the SET command omit the "F" suffix from the Q=(unitaddr,F) parameter or omit the Q=(unitaddr,F) parameter entirely.

To restart the system enter the command (CLOCK is optional):

SET DATE=yy.ddd,CLOCK=hh.mm.ss

Omitting the Q=(unitaddr,F) parameter saves job queue data set information.

Make sure that all direct access volumes containing system input data and system output data are ready.

System input and system output volumes may be switched to a different unit of the same type as the one they were on when the system went down. If, however, the volumes were on a control unit having the record overflow feature, and the track overflow feature was used to record the data, then the volume that was on the control unit having the record overflow feature can only be switched to another control unit having the record overflow feature.

All volumes with SYSIN should be mounted.

All volumes with SYSOUT should be mounted.

When a system output writer encounters a data set all of whose volumes are not mounted, the data set is bypassed. Message IEF304I will go to the console and output device and will tell you the data set name and the volume serials, which you can use to dump the data set at a later time.

The job scheduler will send you the names of any jobs and job steps being handled by the reader or initiator before the failure, and the output writer will print out all the data on the system output data sets.

You'll receive the message IEF420I to indicate the jobs that were handled by the reader and the message IEF421I to indicate the jobs and job steps that were handled by the initiator.

A job referred to in the message IEF420I can be reentered as is. Message IEF421I looks like this:

$$\text{INIT=job.step.procstep(x)}\left\{\begin{array}{l}\text{RESTART}\\ \text{NO RESTART}\\ \text{CANCELLED}\\ \text{ENDED}\\ \text{CONTINUING}\end{array}\right\}$$

where x is

1 If the step was being processed by the initiator and had not started executing (the message will indicate CANCELLED).

2 If the step was executing (the message may indicate RESTART or NO RESTART). In this case, you will have already received the ABEND message IEF450I, followed by this message or by a message requesting you to authorize automatic restart. Should you receive the latter message, follow the procedure described under "Restarting a Job" in <u>Operator's Procedures</u>, GC28-6692.

3 If the step was being ended (the message may indicate CANCELLED, ENDED, or CONTINUING).

Whenever the message indicates NO RESTART or CANCELLED, the particular job may require programmer analysis before it can be rerun.

Controlling Input and Output

This section describes how to work with the operating system in controlling input and output, particularly in the area of allocating or assigning devices to jobs.

INPUT

A system input (SYSIN) stream is made up of job control language (JCL) statements and problem program data routed by the control program to their destinations within the system.

Sets of input data entering the system through the SYSIN stream are called SYSIN data sets. Input data can also be read directly by the problem program and not be a part of a SYSIN stream.

Data in the input stream is written on a direct access volume for later reading by the problem program.

Input Reader

An input reader is that part of the scheduler that reads a system input stream from a single device. You assign an input device to an input reader by issuing a START RDR command.

In systems with MFT, as many as three input readers can bring jobs into the system concurrently. Each input reader is assigned its own SYSIN stream. You issue a START command to start a new input reader, and a STOP command to stop a reader. (Input readers also stop automatically on encountering an end-of-file condition.) SYSIN data sets in an input stream are stored on non-demountable direct access volumes while other programs are running. Job steps can later read the data sets at high speed from the direct access volumes.

Job Classes

Jobs can be grouped, depending upon your installation needs, into as many as 15 different job classes (A-O).

Each class has a one-character classname. All job control information for jobs with the same classname is placed on the same input queue. Jobs with a common characteristic can be grouped together by using the same classname. The characteristic might be a job with much input/output, a job requiring a large amount of CPU processing, or possibly one which requires special control volumes.

The programmer uses the CLASS parameter in the JOB statement to define his job's input class.

OUTPUT

A system output (SYSOUT) stream consists of system messages and problem program output data sets routed by the control program to an output device.

Problem program data sets that leave the system through the output stream are called SYSOUT data sets. Output data can also be written directly by a problem program and not be a part of a SYSOUT stream.

Problem programs write their output on a direct access volume. The output writer later picks up the output for transmittal to the device specified by the programmer for its output. System output can also be written directly from a problem program, to a specified output device, by using Direct System Output Processing. See the chapter "General Operating Techniques" on "How to Use Direct System Output Processing (DSO)."

Output Classes

Your installation, depending on its needs, can group SYSOUT data into as many as 36 different output classes.

Each class has a one-character classname and includes all system messages and SYSOUT data sets the system associates with the classname, allowing the grouping of output data with a common characteristic. The characteristic might be a type of output device, a priority, or possibly a location -- for example, the data for the third floor programmers might have a classname of 3.

Examples of possible output classes are:

A High priority printed output, such as messages from the control program.

B High priority punched output, such as error records to be hand-corrected and rerun.

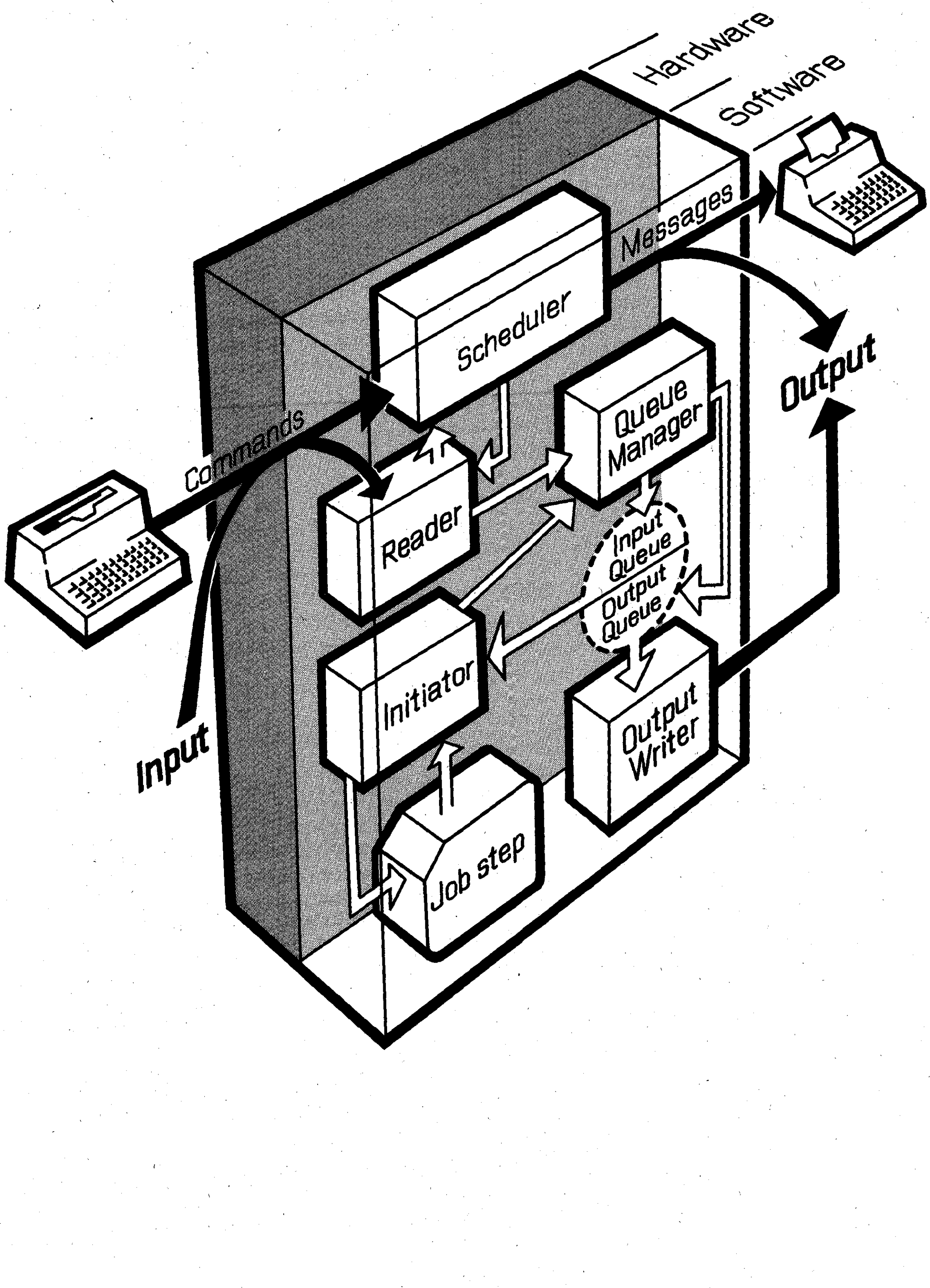
C Low priority printed output, such as a summary dump of disk records for auditing.

If your system has a universal character set (UCS) printer that will be used as an output writer, you must assign a separate output class to each character set image stored in the system library. To associate the character set-code with a SYSOUT class, either specify the UCS parameter in the START WTR command or add the UCS parameter to the user-designed output writer procedure. This assignment preserves the identity of character set-dependent data through the SYSOUT stream. The output is grouped by character set-code, which minimizes your changing of printer chains and trains.

The programmer specifies the SYSOUT class for a set of system messages or a SYSOUT data set for his job. These statements are described in the manual IBM System/360 Operating System: Job Control Language Reference.

The system messages for a job are in the output class specified by the MSGCLASS=parameter of the JOB statement. A SYSOUT data set is in the output class specified by the SYSOUT=parameter of the DD (data definition) statement for the data set.

If a job specifies a class for a SYSOUT data set that is different from the MSGCLASS, and the output data set contains data, message IEF298I jobname SYSOUT=x will appear on the MSGCLASS output immediately preceding the JOB statement.



How the Scheduler Works in Systems With MFT

The JCL statements in the following example describe a one-step job that executes PROGRAM in a system with MFT. The system messages for this job are in class B, while the output data set is in class C. Therefore, if there is data in the output data set, this message will appear:

IEF298I FIRSTJOB SYSOUT=C.

```
//FIRSTJOB    JOB    ,'AUSER',MSGCLASS=B,CLASS=N,PRTY=12
//STEP1       EXEC   PGM=PROGRAM
//RESULT      DD     UNIT=2311,SYSOUT=C,SPACE=(132,(1000,10))
```

MFT

Output Writers

An output writer is that part of the scheduler that writes a single system output (SYSOUT) stream to a single device. You assign an output device to an output writer by issuing a START command.

In systems with MFT, several output writers can run at the same time on several SYSOUT streams, and more than one writer can work on an output class. In these systems, you not only issue a START command to start a new output writer, but also must issue a STOP command to stop one.

By using START, you specify the association between the output classes and devices, by naming (a) the device to be allocated to the output writer and (b) the output classes that the output writer is to process.

You can thus ensure that each SYSOUT class is written on the device that is most suitable for that class.

Given the example in the preceding section, you might issue the following command, assuming that 00E is a printer and WTR is the procedurename of a writer.

```
START WTR.P3,00E,,C
```

This would cause the output data set to be printed. On the other hand, (assuming 185 is a magnetic tape drive) if you issued:

```
START WTR.P3,185,,C
```

the data set would be written on magnetic tape. If you wanted a printed listing from this tape, you could use the tape SYSOUT program described under the heading "How to Print a SYSOUT Tape" in the Operator's Procedures, GC28-6692.

A job step writes SYSOUT data sets at high speed onto a direct access device. Later, after the entire job is completed, an output writer will handle the data sets concurrently with other jobs.

System Log

The system log is kept by the system on a permanently mounted direct access volume. The system log is an optional feature that may be used by both problem programs and operators.

There are several kinds of information that can appear in the system log:

- Job time, step time, and data from the JOB and EXEC statements of a job that has ended. This information is entered on the log by an accounting routine written at your installation.
- Operating data entered by problem programs using a write to log (WTL) macro instruction.
- Descriptions of unusual events that occurred during your shift. You can enter these descriptions by issuing a LOG command.
- WTO and WTOR messages, including the routing codes used to route the message and the time the message was executed (systems with the MCS option only).
- Commands entered through operator's consoles and the input stream, and commands issued by the operating system (systems with the MCS option only).

Use the LOG command to make entries into the system log.

The system log consists of two data sets -- SYS1.SYSVLOGX and SYS1.SYSVLOGY. System message IEE041I LOG NOW RECORDING ON SYS1.SYSVLOG (X/Y) tells you which data set is currently being used. When the current data set is filled, the system will close it, open the other, and issue message IEE042I LOG DATA SET SYS1.SYSVLOG (X/Y) ON device CLOSED. The system then begins recording the log on the second data set and issues messages IEE041I to inform you that log recording has switched to the second data set.

Message IEE043I LOG DATA SET (X/Y) QUEUED TO SYSOUT CLASS x tells you that the filled data set has been placed on the output queue for the SYSOUT class specified in the message, though the actual scheduling cannot take place until a job currently running in the system has reached termination.

The class specified in message IEE043I is the default output class for the system log. The default value is class L unless your installation has modified it during system generation.

If you want the system log to be part of an output class other than the default class, enter a WRITELOG command with the desired classname each time you wish to modify the default class. Also use the WRITELOG command if you want the system log written out at a specific point in time.

Note: Whether the data set is scheduled for output by the system or by your WRITELOG command, make sure a writer has been started for the appropriate output class.

ALLOCATING DEVICES

Device allocation is the assignment of I/O devices for use by job steps or the system. If a job step is using a tape drive, no other job step may use the tape drive until the first job step is finished -- the entire device is assigned. If the job step is writing on a direct access volume, however, it may only be necessary to assign a particular set of tracks; data sets for many job steps can be on the same volume.

Device allocations are made in response to requests from three sources:

- Data definition (DD) statements. These may specify the I/O requirements of the job steps in the system.
- System generation statements. These may request fixed assignments for system processes, such as the automatic starting of an input reader.
- Operator commands. These may request assignments for system processes, or may modify assignments made when the system was generated.

MFT

Assignment by the Scheduler

Through data definition statements, the programmer describes the input/output requirements for the data sets of a job step.

Using this information, the job scheduler allocates I/O devices directly to the job step, attempting to provide overlapped operation, conserve input/output resources, and recognize items that increase input/output efficiency.

Assignment by the Operator

You are responsible for device assignments made for the starting of input readers and output writers. If assignments were specified at system generation time, you don't have to respecify them, unless you wish to make a modification.

All of your assignment requests are made through operator commands.

Your ability to switch input/output devices to online, offline, or console status lets you modify their allocation.

Online devices can be assigned to problem programs, while offline devices and consoles cannot be. All input/output devices can be switched online or offline, but only certain types can be consoles. Devices capable of being consoles are discussed in the chapter "Operator's Consoles."

Your main reason for placing a device offline is to reserve it for preventive maintenance. To switch a device's status, enter a VARY command indicating the device and the desired status.

By entering the MOUNT command, you can cause an input/output device to be assigned to those job steps that require the particular volume mounted on it. For example, you might give a MOUNT command for a device when you know that a volume will soon be used by many independent jobs.

In systems with the automatic volume recognition option, you can mount volumes on online devices that are not ready, thus anticipating the later needs of jobs you are scheduling.

Volume Mounting: In most installations, your role with respect to I/O devices is to mount and demount volumes.

The job scheduler, using information from data definition statements, determines the input/output resources to be assigned to a job and the volumes that are required. If these volumes are not mounted, the job scheduler writes you a mounting message.

Each message states that either a specific volume or a scratch volume is to be mounted. Mount the requested volume and press the START key on the device to continue processing.

Never mount a blank tape volume, unless specifically directed to do so.

The system checks for a tape label and the absence of data causes the whole volume to be scanned for a data record. If an unlabeled tape is required, a tape mark should be written to avoid unnecessary scanning.

For a description of a program you can use to create standard labels on tapes, see the section on the program IEHINITT in the publication IBM System/360 Operating System: Utilities.

After you mount the volume and ready the drive, the system reads the volume label. If an incorrect volume is mounted, the system repeats the mounting message and unloads the incorrect volume, if possible (some device types, such as the 2311, can only be unloaded manually).

If a scratch tape volume is incorrect only because the job specified a nonstandard or unlabeled tape and you have mounted a standard-labeled tape, the system will rewrite the tape label to conform to job specifications.

If a scratch tape is incorrect only because the job specified a labeled tape and you have mounted one without a label, you will receive message IEC704A L ddd.

If you want the system to write a new volume label for the tape, respond REPLY id,'ser,owner' to define the volume serial number and, optionally, the owner's name for the new label.

If you want to use a different tape, respond REPLY id,'M' to cause the system to unload the mounted tape and repeat the mounting message. You can then mount a different tape.

Note: You will receive a message describing any changes the system makes to the volume label; note these changes on the tape reel's external label.

If a request was made for a tape volume without a standard label and if the volume mounted does not have a standard label, that volume will be accepted. The volume is treated as unlabeled, or as a volume labeled with nonstandard labels, according to the DD statement.

The following volume mounting options can be selected at system generation time:

- Imperative mount. Mounting messages are written when the job step requiring the volumes is started.
- Automatic volume recognition. You take the initiative and mount labeled volumes on any unused drives. The system recognizes and remembers these volumes and assigns the drives to later job steps.

If your system has the automatic volume recognition feature, mount volumes you want the system to find for the first job at IPL before entering the START command.

Also before entering START, be sure that all offline devices are known to the system by using the VARY OFFLINE command.

After the first job, you can mount ahead for several jobs at a time.

In addition, the system may ask you to mount other volumes, and you can mount these on any appropriate online devices that are not ready. (Do not unload any units -- you can only mount on units unloaded by the system.)

Automatic volume recognition handles nine-track tape, seven-track tape, and 2311 and 2314 direct access devices. The density for seven-track tape is set at 200, 556, or 800 bytes per inch at the time the system is generated.

Occasionally you may be asked by your installation not to mount volumes on certain devices, and not to make those devices ready, because you are running a version of the operating system that was generated for a slightly different set of devices.

Work Volumes: Make sure there are sufficient work volumes available in the system to satisfy requests for temporary data sets at peak loads. Failure to do this can delay the allocation of a job step while it waits for direct access space to become free.

Where to Mount Active Volumes: At IPL, mount relatively full and active disk packs (for example, system packs) at the highest address locations available on the channel. Also, place system residence and system data sets (SYS1.MACLIB, for example) at the highest available address locations of different channels.

Shared DASD Option: If your installation is using a system with this option, be sure to read the section "How to Use the Shared DASD Option" in the chapter "General Operating Techniques."

Wait Messages

During operation, you may receive a message informing you that a job is waiting in a partition for resources to become available.

When these resources become available, the job resumes processing. If, however, the job is waiting in the only large partition in the system (or the only large partition not containing an unending job), the needed resources will never become available unless you free the partition for use by the system.

If the reader is waiting for space or if the initiator is waiting for space or a device, reply 'CANCEL' in order to free the partition for further work.

If a transient writer is waiting for work, issue a STOP command in order to free the partition. If another large partition is available in the system, no reply is required.

In some cases a message series will be issued indicating that a job cannot be initiated because needed data sets are reserved by another job.

You will have to make a decision, based on your knowledge of the jobs in the system, the system configuration, and the data sets requested.

You will receive message IEF864D REPLY 'RETRY' or 'CANCEL' as the last message in the series. Your reply to this message will probably be 'RETRY' unless you know that:

1. There is only one scheduler-size partition in the system and therefore no other jobs can be terminated if the scheduler is waiting in that partition.

2. The data sets listed are reserved by an unending job, and therefore will not be likely to be released.

3. The job waiting is being scheduled into a small partition and the scheduler is operating in the only large partition which does not have an unending job in it.

If you reply 'CANCEL' to this message, you can later reenter the job at a time when the data sets will be available.

Your reply to 'RETRY' need not be made immediately; you can wait until other jobs have terminated and freed the necessary data sets.

Restarting a Job

A job may be abnormally ended as a result of a hardware, programming, or system error. Such an ending may occur at any time during program execution.

Valuable machine time would be lost if an abnormal end occurred during the processing of one of the last job steps of a multistep program or in the middle of a long job step and execution of the program had to be started again at the beginning of the first job step.

The checkpoint/restart feature of the system is provided to allow a restart of an abnormally ended job either at the beginning of a job step or at a checkpoint within this step.

Your programmer determines whether an automatic restart or a deferred restart is to be performed.

AUTOMATIC RESTART

If your programmer provides for an automatic restart and the job does abnormally end, you will receive message IEF225D SHOULD job RESTART requesting a reply to authorize the restart. When a program is to be restarted at a checkpoint, it may be delayed if the system must wait to allocate the required storage area. See "Checkpoint Restart Storage Allocation" for the special considerations involved. The SHOULD job RESTART message also allows you to prevent repeated restarts at the same checkpoint or job step.

When you are requested to authorize an automatic restart, you can reply YES, HOLD, or NO.

- Reply YES if the restart is to be performed at a specific checkpoint or job step for the first time.

  If a step restart is to occur and the step to be restarted used a card input data set that was not part of the SYSIN stream, you must return, to the appropriate hoppers, all cards read by the job step before it ended abnormally. If a checkpoint restart is to occur, follow the programmer's instructions for replacing the input cards.

- Reply HOLD if you do not wish to perform the restart at once, but want to do so soon: for example, to permit another job to be run first.

  To restart the job at a later time, you must issue an appropriate RELEASE command. Or, if desired, you may cancel the job.

- Reply NO if a restart at a specific checkpoint or job step has been requested repeatedly. When your reply is NO, and your programmer wants a restart to be performed, he must resubmit the job for a deferred restart.

The system may request you to mount data volumes other than those required at the beginning of the job.

After you authorize a restart and if you are displaying jobnames, you'll get jobname STARTED and jobname ENDED messages describing system job IEFREINT (the restart reader). Ignore them. Should failure occur during execution of this job, you'll get an ABEND message.

Note: Any operator commands in the input stream of the job step being restarted will not be executed.

DEFERRED RESTART

If your programmer provides for a deferred restart, and his job does abnormally end, he must resubmit the job to have this restart performed.

To restart the job, your programmer must provide a restart deck for submission to the system through the system input reader. The JCL statements to be included in the restart deck are described in detail in the publication IBM System/360 Operating System: Job Control Language Reference.

The device configuration of your system at the time of restart need not be the same as it was when the job was abnormally ended. However, enough devices must be available to satisfy the needs of the job step being restarted.

The system under which a step restart is run need not be the same as it was for the job's original execution. However, a checkpoint restart should be run under the original system unless the alternate system can meet the following restrictions:

- The type (PCP, MFT, MVT) and release number are the same.
- SYS1.SVCLIB should be the same as it was originally.
- The resident access method modules in use at the checkpoint are in the same storage locations.

* The boundary between hierarchy 1 and hierarchy 0 must be the same for a system with Main Storage Hierarchy Support for IBM 2361 Models 1 and 2.

* An area of main storage identical to the original area must be available to the job to be restarted.

When the job restarts correctly, you will receive a message, IHJ008I job RESTARTED. If you do not get this message, you'll receive a message describing the delay. See Messages and Codes for descriptions of the responses you should make.

The system may request you to mount data volumes other than those required at the beginning of the job. In addition, any card input data sets that have been used by the failing job step must again be made available to the system.

Checkpoint Restart Storage Allocation

A checkpoint restart can be either automatic or deferred, but the restart step must occupy the same area of main storage as it did during the original execution. The restart will be delayed if the system is presently unable to allocate the required main storage because:

* The required area is divided between two or more partitions.

* The required area is contained in a RDR or WTR partition.

* There is a pending DEFINE command for the partition that contains the required area.

System messages will be issued to inform you of the delay. See Messages and Codes for a detailed explanation of the actions you must take, such as redefining partitions, to allow the restart to continue.

If the nucleus has expanded into the required area or if a deferred restart is being executed on a different system and the required storage is not entirely in the dynamic area, message IEF209I MAIN STORAGE UNAVAILABLE is issued, and the system cancels the job.

Operator Commands

MFT

This section contains a description of the commands you use to give control information to the operating system. The formats, functions, parameters, and options of the commands are included. The commands are presented in alphabetical order.

Abbreviations as well as the full command name can be used when keying in the commands. The usable names and abbreviations are:

CANCEL	C	RELEASE	A
*DEFINE	N	REPLY	R
DISPLAY	D	RESET	E
*HALT	Z	SET	T
HOLD	H	START	S
LOG	L	STOP	P
*+MODE		*SWAP	G
MODIFY	F	UNLOAD	U
MONITOR	MN	VARY	V
MOUNT	M	WRITELOG	W

* These commands cannot be entered into the input stream.
\+ This command cannot be abbreviated.

When you issue certain commands, such as DISPLAY, you should be aware that confusion may arise due to intermingled messages. While displaying the status of jobs, you may also be informed of system messages, of WTOR messages, or of jobs beginning or ending or both.

If the system does not respond promptly to a command, such as DISPLAY A or DISPLAY R, it may be because a DEFINE command is in progress. Check the console output to determine if any replies to the DEFINE command are outstanding.

A CANCEL command is executed when it is entered, even if it refers to a job in a partition other than the one the scheduler is in.

Be sure to use the correct abbreviations for operator commands. For example, use S for START and T for SET. If you inadvertently key in S for SET, the system assumes you are giving a START command, queues the command, and waits for a SET command.

The following conventions are used in illustrating the format of commands:

- Required letters (those shown in upper case) must be entered, but can be entered in either upper or lower case.
- Lower-case letters indicate that a parameter must be substituted.
- Dotted lines ... (indicating a series of terms), brackets [ ], and braces { } are not entered.
- Entries within brackets [ ] are optional.
- Entries within braces { } are required - you must select one.
- Numbers and punctuation marks (other than dotted lines, brackets, and braces) must be entered as shown.
- Stacked items represent alternative items. Only one of the stacked items is to be coded.

Except for letters between apostrophes, lower case letters are translated to upper case before being handled by the command scheduler.

Some commands require apostrophes in their operands. Be sure letters between apostrophes in these commands are upper case if they are meant to be processed as upper case.

When using the REPLY command to answer system messages, it is always correct to use upper case letters in the text between apostrophes.

Command formats are essentially free form, but one or more blanks must follow the operation field. Commands cannot occupy more than one line.

For example, if a command is entered through a card reader, it may not be more than 80 characters in length.

If comments on commands are necessary, they must appear to the right of the operand field and be separated from it by at least one blank. If the operand field is null, a comma followed by at least one blank indicates that comments will follow.

CANCEL -- Terminate Process Immediately

Use the CANCEL command to immediately terminate:

- The scheduling or execution of a job from the system input stream or a job started by the START command (a procedure in SYS1.PROCLIB). You may be asked by your system programmers not to use the CANCEL command on certain jobs that alter data sets containing information vital to the system; canceling these jobs might make the data unusable.

- A system task in the device allocation process. A system task is in allocation if you receive a message with the prefix IEF (the exception is a mount message for tape with the prefix IEE). If a system task needs operator intervention during device allocation (such as mounting a pack or canceling a job because of a mount message that you cannot satisfy), it can be canceled.

  The CANCEL command will not work for a system task after the device allocation process has completed. If a system task does not have a device associated with it, the system task cannot be canceled. To stop a running system task you must enter the STOP command.

- The writing of an output data set currently being processed by an output writer.

<u>Note</u>: If the initiator has issued any messages requiring a REPLY, you must reply to those messages to allow related jobs to be removed from the system -- before you cancel. Otherwise, the system will enter a wait state.

Operation	Operand
{CANCEL | C}	{jobname [[,DUMP][,ALL] | IN[=class] | OUT[=class]] | identifier}

jobname
: the name of the job (process) to be terminated. The jobname is the name of a job from the system input stream.

  If a partition identifier (any one from P0 through P51) of an active partition -- for example, P4 -- is used as a job name, a CANCEL command will cancel both the job named P4 and any system task waiting for allocation in partition 4.

DUMP
: specifies that an abnormal-end-of-task storage dump is to be taken if a step of the job is being executed when the command is received. If the programmer has put in the SYSABEND data definition statement, a full dump is taken. If this card was omitted, an indicative (partial) dump is taken.

ALL
: all system input and/or system output for the specified job is canceled.

IN=class
: the system will search for the job on the input queue specified by the class parameter. If IN is used without the class parameter, all input queues will be searched for the job.

OUT=class

the system will search for the job on the output queue specified by the class parameter. If OUT is used without the class parameter, all output queues will be searched for the job.

Note: If neither the IN or OUT parameter is used the system will search all the input queues and the hold queue for the job.

identifier

the identifier, from the START command, of the system task to be terminated during device allocation or the job started from the console.

The following can be used to cancel a system task in the device allocation process.

- The identifier used in the START command.
- The unit type (e.g., 1403 or 2311) associated with a unit address in the START command.
- The unit type associated with a cataloged procedure, in SYS1.PROCLIB, started by the START command.

The identifier for a job started from the console is the partition number in which it is run.

To stop a system output writer from completing the writing of a system output data set, cancel the device address (e.g., 00E or 00F) associated with the system output writer processing the data set. The device address is taken from the devicename parameter of the START WTR command. The system output writer will continue processing further output.

DEFINE -- Invoke Dynamic Partition Definition

Use the DEFINE command after nucleus initialization to change the size and description of any partition while unaffected partitions continue processing. This command cannot be entered into the input stream.

The system will respond to your DEFINE command with IEE866I DEFINE COMMAND BEING PROCESSED followed by IEE802A ENTER DEFINITION. Respond to message IEE802A by using the REPLY command. (See How to Change Partitions in the Operator's Procedures manual.)

MFT

Operation	Operand
{DEFINE / N}	[LIST]

LIST
    the current partition definitions will be listed. Job classes associated with currently active partitions and, if time slicing is used, the time-slicing specifications are listed also.

CAUTION: When using the time-slicing option, do not define job classes across the boundaries between a time-sliced group of partitions and partitions that are not time-sliced. For example, don't specify a partition with job classes A,B,C in a time-sliced group, and a partition with job classes C,D,E outside the group. Doing so would allow a job in class C to be put either inside or outside the time-sliced group regardless of the intentions of the programmer of that job.

DISPLAY -- Cause Current Display

Use the DISPLAY command to cause a current display of:

- The time of day and the date.
- Job status or the activity of tasks within the system.
- The status of input/output devices attached to the system.
- Outstanding requests for operator replies.
- The status of the queues.
- The status of a specified job.
- The system console configuration.

This command provides you with job name information needed for effective use of the CANCEL command and, together with system messages, keeps you informed of which jobs are currently being executed.

If you are using a display operator console, use the DISPLAY command to display the CONTROL command, its operands, and an explanation of each operand. See Operators Guide for Display Consoles, GC27-6949 for a description of the CONTROL command.

```
┌──────────┬───────────────────────────────────────────────────────────┐
│Operation │Operand                                                    │
├──────────┼───────────────────────────────────────────────────────────┤
│{DISPLAY} │/T                                              \          │
│{   D   } │ A                                               |         │
│          │   [,TP     ]                                    |         │
│          │   [,GRAPHIC]                                    |         │
│          │ U [,TAPE   ] [,ONLINE ][,xxx][,nnn]             |         │
│          │   [,DASD   ] [,OFFLINE]                         |         │
│          │{  [,UR     ]                                     }        │
│          │ R                                               |         │
│          │ Q[=list]                                        |         │
│          │ N[=list]                                        |         │
│          │ jobname                                         |         │
│          │ C,K                                             |         │
│          │\CONSOLES                                       /          │
└──────────┴───────────────────────────────────────────────────────────┘
```

T
: the time of day and the date are to be displayed in the following format:

```
hh.mm.ss                  yy.ddd
|  |  |                   |  |
|  |  └--Seconds (00-59)  |  |
|  └-----Minutes (00-59)  |  └--day of year (000-366)
└--------Hours (00-23)    └-----year (00-99)
```

A
: the jobname, current stepname, and partition are to be displayed for each job currently executing. If a system task is running, its name and partition will be displayed. In MFT systems with subtasking, the number of attached subtasks within each partition is displayed. If the number is zero the field is left blank. If you have started Direct System Output Processing it will not be displayed. If a job is currently in the process of having devices allocated to it, or of being terminated, no indication that a task is active in that partition is given.

U

a listing of the Unit Status information about the devices indicated is to be displayed.

Unit status can be displayed about the following device types:

TP	communication equipment.
GRAPHIC	graphic devices.
TAPE	magnetic tape units.
DASD	direct access storage devices.
UR	unit record devices.

MFT

If you do not specify a device type, the unit status of all devices in the system will be displayed.

You may display those devices that are ONLINE, or those devices that are OFFLINE. If you do not use either of these parameters, both ONLINE and OFFLINE devices will be displayed.

Unit Status information can be displayed as specified in the second and third operands starting at address xxx for nnn number of devices. If xxx is omitted the starting address is 000. If nnn is omitted the number of devices is 100.

R

the system is to display:

- The id of each message that required a reply and has not yet been replied to.
- The unit address of each device for which a mount message has been issued but has not been complied with.
- An indication if any AVR mount messages are pending.

When you use the DISPLAY R command, the system issues message IEE110I if any operator action (reply to messages or mount volumes) is required from previous messages. If you have complied with all system requests, you will receive system message IEE111I NO OUTSTANDING REQUESTS.

Q

a listing of the number of entries on each of the non-empty input, hold, and output queues is to be displayed. Also included in the display may be the Remote Job Entry (RJE) queue.

N

a listing of jobnames on the hold, input, and output queues is to be displayed. Also included in the display may be the Remote Job Entry (RJE) queue.

list

any combination of up to three of the following items:

specific input work queue name (job class A through O)

HOLD (system hold queue)

If list includes more than one item, you must separate the specified items by commas and enclose them in parentheses. If no list value is specified, all 15 input work queues, the hold queue, and the output queue are assumed.

jobname
the name of the job for which the following are to be displayed: job name; class; job priority; type of queue the job is in -- JOB Q, HOLD Q, or SOUT Q (SYSOUT queue); and position in the queue. The maximum length of a job name is eight characters. If JOBNAMES, STATUS, T, A, R, Q, N, C, U, SPACE, DSNAME, CONSOLES, USER, or SESS is used as a job name, it must be in parentheses.

C,K
the system is to display the CONTROL command operands and an explanation of each operand. This display is referred to as a status display. C,K is valid only for display operator consoles.

CONSOLES
the system's console configuration is to be displayed. This operand is valid only when the operating system has the multiple console support (MCS) option. The display for each console includes:

- The unit address of the console (or the input and output addresses for composite consoles).
- The unit address of the alternate console (or the input and output addresses for composite consoles).
- The status of the console, where A indicates an active secondary console, A,P indicates a pending request to make the device an active console, N indicates an inactive secondary console, N,P indicates a pending request to make the console device inactive, M indicates the master console, and H indicates the hard copy device.
- The command groups that the console is authorized to enter.
- The routing codes that the console is authorized to receive.

In addition, the display includes:

- The unit address of the hard copy log, or "SYSLOG" if the system log is the hard copy log.
- The routing codes the hard copy log is authorized to receive.
- Whether or not the log is receiving operator and system commands and their responses.

HALT -- Prepare for Power-off

Use the HALT command before you turn the power off at the end of the day, or any time the computer is not to continue under the control of the operating system. This command cannot be entered in the input stream.

You must use this command to ensure that important statistics and data records in main storage are not lost permanently.

MFT

Use of this command also closes the system log and discontinues the log function. The contents of the system log may, if your installation chooses, be written by a system output writer before power is turned off. Before you enter the HALT command, use the WRITELOG command to initiate this writing.

Operation	Operand
{HALT, Z}	EOD

EOD
the end-of-day storing is to be done of internal I/O device error counts. The information is stored in the SYS1.LOGREC data set (see the topic "Hardware Debugging Aids" in the chapter "General Operating Techniques." This parameter is required.

When the storing is done, the system sends you message IEE334I HALT EOD SUCCESSFUL. At this point, you can safely turn the power off.

HOLD -- Temporarily Suspend Job Selection

Use the HOLD command to temporarily prevent one job, or all jobs in the input work queue, from being selected for processing.

If the named job has already been selected, or if it is not in the input work queue, a message will be received.

Jobs temporarily suspended by HOLD are subject to CANCEL and RESET commands.

The HOLD command works in two different ways, depending on whether you use the jobname operand or the Q operand.

A HOLD jobname command causes the job to be withheld from initiation until a RELEASE jobname command is given.

A HOLD Q command, on the other hand, prevents the selection of all jobs in the specified input work queues until a RELEASE Q command is given.

A RELEASE Q command will not release a job held by a HOLD jobname command, nor will a RELEASE jobname command release a job that is in a queue which is held because of a HOLD Q command.

Operation	Operand
$\left\{\begin{matrix}\text{HOLD}\\ \text{H}\end{matrix}\right\}$	$\left\{\begin{matrix}\text{jobname}\\ \text{Q[=list]}\end{matrix}\right\}$

jobname
: the name of the job whose selection is to be suspended. The maximum length of a job name is eight characters. Although any job name can be in parentheses, a job with the name Q must have the Q in parentheses in the command statement.

Q
: the selection of jobs from a specified input work queue is suspended. If the parameter Q is specified without the =list option, selection of jobs from all work queues is suspended. A RELEASE Q command will release the input work queue(s).

list
: any combination of up to four input work queue names (job class A through O). If no list value is specified, all 15 input work queues are assumed. If list includes more than one item, you must separate the specified items by commas and enclose them in parentheses.

LOG -- Store Information in Log

Use the LOG command to enter information into the system log.

The system log consists of two data sets on a direct access device. (Since it is an optional feature of the system, you should know whether or not your system has the option before you use the LOG command. When you enter the LOG command and the system log option is not in your system, you receive a message informing you that the log is not supported.)

MFT

When your system has the multiple console support (MCS) option, the time that the LOG command was received by the system is appended to the message text entered in the system log. (Your system must have the timer option to make the time stamp on the message meaningful.)

Operation	Operand
{LOG, L}	'text'

text
: the exact text you wish to enter into the system log. The message written in the system log does not include the enclosing apostrophes. The type of letters and special characters that can be entered (such as upper case alphabetic only, or upper and lower case alphabetics) depends upon the character set installed in your printer. For a description of the available character sets, see Operator's Procedures.

The system will send your LOG command to the primary console or the MCS master console if the system log is temporarily inactive (as when both data sets are being written out). Enter the command again when the log becomes active (as when the data sets have been completely written).

MODE -- Recovery Management Mode Switching for Models 85, 155 and 165

The MODE command is used to control recovery management activity in the CPU of Models 85, 155, and 165. The format and uses of the command are explained in the following sections.

MODEL 85

The MODE command for Model 85 specifies the mode of recording recoverable machine check interruptions: (either a message for each error processed or a message only after a certain number have been processed). You also use MODE to reactivate deleted sectors of the high speed buffer area, to re-enable the entire buffer, or to reactivate the high speed multiply feature. Use MODE with the STATUS operand to obtain a message describing the current state of recovery management facilities.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	STATUS INIT HIR,x[,dddddddd] ECC,x[,dddddddd] SECT= { n[,n,...] / ALL / BUF } HSM

STATUS
: a message describing the status of machine recovery facilities is to be displayed. See "Status Message" below for a description of the information you will find in this message.

INIT
: recovery management functions are to be set to their initial values: the buffer is enabled with all sectors active, the high speed multiply feature is activated, and the Hardware Instruction Retry (HIR) and Error Correction Code (ECC) circuitry, machine facilities that attempt recovery from machine check interruptions, are set to recording mode with the default threshold count specified. Threshold count values control the number of errors allowed in a given amount of time before a change is made from recording the errors in messages to simply counting them.

HIR
: the Hardware Instruction Retry circuitry is to be set.

ECC
: the Error Correction Code circuitry is to be set.

x
: the mode to which the HIR or ECC circuitry is to be set; use R to indicate Recording Mode and C to indicate Count Mode.

  Recording Mode means that a message is written to the operator for each error processed by the recovery management facilities.

  Count Mode means that each error processed by recovery management facilities causes a counter in the CPU to be updated. No message is written until the threshold count is reached; then you receive a message and the counter is reset to zero.

dddddddd

the threshold count in decimal. The function of the threshold count depends on the mode of recording machine check interruptions. When the threshold is reached in Recording Mode, the system automatically switches to Count Mode. When the threshold is reached in Count Mode, a message is written and a counter in the CPU is reset to zero. If you omit this parameter, the threshold count remains unchanged.

MFT

SECT

a previously deleted and repaired sector of the high speed buffer is to be reactivated. You can specify any number of sectors, as long as the message does not exceed the message length for your console device.

n

the individual sector(s) to be reactivated.

ALL

all sectors are to be reactivated.

BUF

the entire high speed buffer is to be re-enabled.

HSM

the High Speed Multiply feature in the Model 85 CPU is to be reactivated. Use MODE HSM only after the errors in high speed multiply that caused the system to switch to slower speed multiply have been corrected.

Model 85 Status Message: The message that you receive in response to your MODE STATUS command includes the following information about the current state of recovery management facilities:

1. HIR mode, threshold count, present error count
2. ECC mode, threshold count, present error count
3. SECTOR deletions
4. BUFFER status
5. HSM status

Note: The threshold count is set at system generation and can be reset by use of the MODE command during system operation.

MODEL 155

Use the MODE command for Model 155 to obtain information on the current state of the hardware facilities that attempt recovery from machine check interruptions. Use this command also to enable or disable soft (recoverable) machine check interruptions and error recovery report messages, and to change threshold counts that control the number of errors allowed in a given interval of time before a change is made from a mode that records errors to a mode in which no record of errors is kept.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	{ STATUS HIR,x[,eeee][,tttt] ECC,x[,eeee][,tttt] }

STATUS
: a message describing the current status of machine recover facilities is to be displayed. See "Model 155 Status Message Format" below for a description of the information you will find in this message.

HIR
: the Hardware Instruction Retry circuitry is to be set.

ECC
: the Error Correction Code circuitry is to be set.

x
: the mode to which the HIR or ECC circuitry is to be set. Use R to indicate Recording Mode and Q to indicate Quiet Mode.

<u>Recording Mode</u>, the normal mode for Model 155, provides for a soft (recoverable) machine check interruption and an error recovery report message to the operator after each machine check condition.

<u>Quiet Mode</u> means that soft machine check interruptions will be disabled, and that no error recovery report messages will be written to the operator.

eeee
: a four-digit decimal value to be inserted in the error count threshold. This value is the number of errors that will be allowed in a specified time before the HIR or ECC circuitry is placed in the quiet mode. This parameter is used with the R parameter only; if you omit it, the default error count threshold value will be used.

tttt
: a four-digit decimal value to be inserted in the time threshold. This value is the time in hours allowed for a specified number of errors before the HIR or ECC circuitry is placed in the quiet mode. This parameter is used with the R parameter only; if you omit it, the default time threshold value will be used.

<u>Note</u>: Because of Model 155 hardware design, ECC may be in either recording mode or quiet mode when HIR is in recording mode; but when HIR is in quiet mode, ECC is placed in quiet mode automatically. Note also that the eeee and tttt parameters are positional; when the time parameter is used alone, it must be written: <u>,,tttt</u>.

Mode 155 Status Message: The message that you receive in response to your MODE STATUS command contains the following information about the current state of recovery management facilities:

```
HIR,{R},aaaa/bbbb/,cccc/dddd ECC,{R},aaaa/bbbb/,cccc/dddd BUF DEL=xxxx
    {Q}                          {Q}
```

where:

R = recording mode

Q = quiet mode

aaaa = current error count

bbbb = error count threshold

cccc = elapsed time, or INVL when TOD clock is invalid

dddd = time threshold

xxxx = number of buffer pages deleted

The MODE command may be used at the operator's discretion, or at the request of an IBM Field Engineer, to check the number of errors that have occurred, or to change threshold values, either to prevent the hardware facilities HIR or ECC from going into the quiet mode, or to return to the recording mode after the quiet mode has been entered. Or, if messages indicating that ECC or HIR has successfully recovered from a machine check interruption are being written out too frequently, your programmers may advise you to use the MODE command with the Q parameter to suppress these messages. Some examples of MODE command usage are:

1. MODE HIR,R,0009,0009

   This entry sets the mode to record errors by inserting new system error count and time threshold values. The counters will be reset to zero and the system will record until the new threshold values are exceeded.

2. MODE ECC,R

   This entry switches the mode of ECC from quiet to record. IBM default values will be inserted as the new threshold values. Note that this command will not be executed if HIR is in the quiet mode at the time it is entered.

3. MODE ECC,Q

   This entry switches the mode of ECC from recording to quiet. No threshold values are required since no record of interruptions will be kept and no counters will be updated.

MODEL 165

Use the MODE command for Model 165 to determine if the system is running in recording mode or quiet mode, what the soft error (recoverable machine check) and soft error threshold counts (the maximum number of soft machine-check interruptions recorded by the system) are, and if the buffer is enabled or disabled. Also use the MODE command to set the Model 165 in recording mode or quiet mode and to enable the buffer.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	{ STATUS RECORD QUIET ENABLE }

STATUS
: the current status of the recovery management facilities is to be displayed. The message indicates if the Model 165 is operating in recording mode or quiet mode. The message will also tell you how many soft errors have occurred and how many soft errors the system will accept before it switches to quiet mode. The status of the buffer is also given.

Some examples of STATUS messages might be:

```
IGF053I  MODE STATUS - QUIET COUNT-12 THRESHOLD-12 BUFFER ENABLED
```

In this case the Model 165 has just switched to QUIET mode on the occurrence of the last error. If the Model 165 is now switched back to RECORD mode, the soft error counter will be reset to zero.

```
IGF053I  MODE STATUS - RECORD COUNT-0 THRESHOLD-12 BUFFER DISABLED
```

Here the Model 165 is in RECORD mode; there have been no soft errors, but the buffer has been disabled. This indicates that four buffer errors have taken place. A good way to handle this situation would be to use the ENABLE parameter of the MODE command to re-enable the buffer and a short time later to again issue the STATUS parameter. If the buffer was again disabled, it indicates that there is a recurring problem with the Model 165 which keeps disabling the buffer.

RECORD
: the Model 165 is to go into recording mode. As stated before, all soft machine checks in this mode will generate machine-check interruptions. In addition, by specifying RECORD, the soft error counter will be reset to zero. The machine does not have to be in QUIET mode to specify RECORD. The RECORD parameter of the MODE command can be used just as a way of resetting the soft error counter.

QUIET
the Model 165 is to go into QUIET mode. A case in example of which it would be desirable to switch to QUIET mode is the case where SYS1.LOGREC is almost full. Rather than use the remaining space to log information about soft errors, the space could be saved for a possible hard error. When the data is retrieved from the SYS1.LOGREC data set, the Model 165 could again be switched to RECORD mode to record information about all errors.

ENABLE
the high-speed buffer, which had previously been disabled, is to be reactivated. The buffer is disabled when four buffer errors occur. When ENABLE is specified, the counter for buffer errors is reset to zero.

MODIFY -- Change Process Characteristics

Use the MODIFY command to change the characteristics of:

- An output writer. You can change the output classes associated with an output writer and the conditions under which the output writer pauses for servicing of its device. When an output writer pauses, it sends you a message requesting you to perform any necessary action on its device. If the pause results from a new form number specification, you are given the form number.
- Direct System Output Processing (DSO). You can change the job classes and the output classes associated with DSO.
- A job. You can change programmer-specified values in a job started from the console by the START command (a cataloged procedure in SYS1.PROCLIB) or a job entered from the system input stream. If the programmer has not set the proper indicator, the job will not respond to the MODIFY command.

Operation	Operand
{MODIFY / F}	[procname.]identifier[,CLASS=classnames] [,PAUSE={FORMS / DATASET}] [,JOBCLASS=jjj][,OUTCLASS=s] [,'job parameters']

procname
: the name of a cataloged procedure (a writer or a job started by the START command) to be modified.

identifier
: the identifier of a writer or a job as defined by the START command. Or, the name of a job from the system input stream.

CLASS=classnames
: one to eight single-character names of the classes to be associated with the output writer; for example, CLASS=ABCD. If more than one classname is specified, the writer treats the specified classes on a priority basis, where the left-most character indicates the highest-priority output class. Note: The system will also accept classnames as a series of characters in parentheses and separated by commas; for example, CLASS=(A,B,C,D). But the form ABCD is preferred to (A,B,C,D).

PAUSE=FORMS
: the output writer is to pause when a change in forms on its device is necessary.

PAUSE=DATASET
: the output writer is to pause before starting to process each data set.

JOBCLASS=jjj
: job class(s) that may use Direct System Output Processing (DSO). From one to eight job classes can by specified (A-O). If you modify an initiator's job classes, this will not have any effect on the DSO job classes.

  If this parameter has been omitted from the DSO START command, and the job class parameter was omitted from the DSO procedure, then the partition's present job classes will be used.

OUTCLASS=s

the system output class that is to be processed by Direct System Output Processing. One system output class can be specified (A-Z,0-9). This parameter is required if the system output class was omitted from the DSO procedure. It will also override an output class specified in the DSO procedure.

'job parameters'

parameters specified by a programmer to change values in a job currently being processed.

MONITOR -- Cause Continual Display

Use the MONITOR command to cause continual display of:

- The name of each job at the time it is initiated or terminated. If a job is terminated abnormally, you will receive a message even if you have not entered the MONITOR command.
- The names of non-temporary data sets (in mount and keep-type demount messages).
- The available space on a direct access volume (in demount messages).
- The names of data sets and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG.

Operation	Operand
{MONITOR / MN}	{JOBNAMES[,T] / DSNAME / SPACE / STATUS}

JOBNAMES[,T]

the name of each job is to be displayed when the job starts and when it terminates. Unit record allocation is to be displayed when the step starts. If a job terminates abnormally, the jobname will appear in the diagnostic message.

If the T parameter is used with the JOBNAMES parameter, the system displays the time of the day in addition to the jobname. In systems with MCS, the time of day is displayed at all consoles. The format of the time display is:

```
hh.mm.ss
|  |  |
|  |  └─ Second  (00-59)
|  └──── Minutes (00-59)
└─────── Hours   (00-23)
```

DSNAME

the system is to display, within the mount and K (keep) type demount messages, the name of the first non-temporary data set allocated to the volume to which the messages refer. Mount messages for data sets with a disposition of DELETE will not contain the data set name.

SPACE

the system is to display, in demount messages, the available space on a direct access volume.

STATUS

the data set names and volume serial numbers of data sets with dispositions of KEEP, CATLG, or UNCATLG, are to be displayed whenever they are freed.

MOUNT -- Allocate Device

Use the MOUNT command to allow allocation of an input/output device to all job steps that require a particular volume, without intervening demountings and remountings of that volume. The volume must be removable.

There will be a short delay after you issue the command before the volume is mounted (the MOUNT command must be scheduled by the system).

Operation	Operand
{MOUNT / M}	unitaddr [,VOL=(NL,serial)],[USE={STORAGE / PUBLIC / PRIVATE}]

unitaddr
: The address of the input/output device to be allocated. Unitaddr must specify a device that has been unloaded by the system.

When issuing this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unit address for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

VOL=(NL,serial)
: the volume specified does not have a standard label. The serial number, up to six characters long, is used for allocation references. This parameter must not be used for direct access volumes, but must be used for nonlabeled tapes. If you do not specify both parameters, NL and the serial number, the system assumes the volume is labeled. The system determines, by examining the label of the volume you mount, whether the label is standard or American National Standard. The volume's label type is included in message IEF279I in the form of SL for standard type or AL for American National Standard type.

USE=STORAGE or PUBLIC or PRIVATE
: a direct access volume will be used as either a storage volume or a public volume or a private volume. If this operand is not used, the system treats the volume as a private volume.

A <u>storage volume</u> is the most freely allocated kind of volume, open to use by the largest variety of data sets, temporary or non-temporary. Slightly restricted is a <u>public volume</u>, which can be allocated freely for temporary data sets, but which must be specified by volume serial number to be allocated to non-temporary data sets. A <u>private volume</u> is the least freely allocated kind of volume -- it is allocated only if its volume serial number is specified.

RELEASE -- Make Job Available for Selection

Use the RELEASE command to resume job selection that has been suspended by the HOLD command, or to initiate a job restart that has been suspended by a HOLD reply. If the job to be selected or restarted is in the input queue in a canceled status or the job is not found, you'll receive a message.

To release a specific job that was held through a HOLD jobname command, or TYPRUN=HOLD on the JOB statement, enter a RELEASE jobname command.

To release jobs held because they were on the specified input work queue when a HOLD Q command was given, issue a RELEASE Q command.

A RELEASE Q command will not release a job held by a HOLD jobname command, nor will a RELEASE jobname command release a job that is in a queue which is held because of a HOLD Q command.

If the job is in the input queue in a canceled status, or if the job is not found, you will receive a message.

Operation	Operand
{RELEASE A}	{jobname Q[=list]}

jobname
: the name of the job to be made available for processing. The maximum length of a job name is eight characters. Although any job name can be in parentheses, a job with the name Q <u>must</u> have the Q in parentheses in the command statement.

Q
: all jobs in the input work queue are to be made available for processing.

list
: any combination of up to four input work queue names (job class A through O). If no list value is specified, all 15 input work queues are assumed. If list includes more than one item, you must separate the specified items by commas and enclose them in parentheses.

REPLY -- Reply to Information Request

Use the REPLY command to reply to messages from the operating system and from problem programs that request information.

The REPLY command need not directly follow the message requesting the reply. The reply id ensures that the message is routed by the system to the correct job. Because other messages can be printed before the reply has been entered, be sure that, in replying to the last message printed, previous messages are not ignored.

MFT

Operation	Operand
{REPLY / R}	id,'text'

id
: a 2-character reply identification field of the message requesting the reply.

text
: the text to be entered in response to a message. The information passed to the program expecting the reply does not include the enclosing apostrophes. When using the REPLY command to answer system messages, it is always correct to use upper case letters in the text.

RESET -- Change Class or Priority of Job

Use the RESET command to change the class, selection priority, or both of a job in an input, hold, or system output queue. The priority of all system output for a job can be reset, or the priority of a job in a specified system output class may be reset. If you try to reset the priority of a job that is in execution, only the priority of the system output for that job will be reset. A canceled job cannot be reset.

You will receive message IEE316I jobname JOB NOT FOUND if the job is not found in an input, hold, or system output queue. The job may not be in the system, may be on a queue other than the one specified in the command, or may have been selected by a writer.

Operation	Operand
{RESET / E}	jobname {,PRTY=priority / ,CLASS=class} [,OUT=outclass]

jobname
: the name of the job whose priority and/or class will be changed. The maximum length of a jobname is eight characters. If as a result of a previous RESET command, a system output class has more than one entry for the same jobname, a new RESET command for the same jobname will apply only to the highest priority entry.

PRTY=priority
: the value to which the job's priority is to be set. The value is a two-digit number that may range from a low of 00 to a high of 13.

CLASS=class
: the input or output class to which the job is to be moved. If a jobname qualifier is present, class is an output class. If no qualifier is present, class is an input class. Class names are A-O for input (job class), A-Z and 0-9 for output (system output class).

NOTE: Both the PRTY and the CLASS parameters can be used in the same RESET command.

OUT=outclass
: a system output class from which the job is to be moved, or on which its priority is to be changed. This field contains a one character class identifier A-Z, or 0-9.

SET -- Set Date, Time, and Location

Use the SET command to establish the date, the time of day, the device for the input work queue and whether the queue is to be formatted, the location of the procedure library, or the automatic commands you wish to override. Any combination of these may be specified.

Use of the optional CLOCK operand in systems with MFT is suggested if the timer option is included. These systems use the data in this operand when they name system data sets.

MFT

Operation	Operand
{SET / T}	DATE=yy.ddd[,CLOCK=hh.mm.ss][,Q=([unitaddr[,F])] [,PROC=unitaddr][,AUTO=characters]

DATE=yy.ddd
: the date in the following format:

```
yy.ddd
|   |
|   └----Day of the year  (001-366)
└--------Year             (00-99)
```

CLOCK=hh.mm.ss
: the time of day in the following format:

```
hh.mm.ss
|  |  |
|  |  └--Seconds(00-59)
|  └-----Minutes(00-59)
└--------Hours  (00-23)
```

If the new clock setting implies a change of date, the new date must be explicitly stated using the DATE parameter.

Q=(unitaddr) or (unitaddr,F) or (,F)
: (1) the address of the direct access device (other than a 2321) on which the volume containing the input work queue (SYS1.SYSJOBQE) resides or (2) that the system is to format the input work queue, or both. This parameter is used only in the first SET command after IPL. Note: If you are specifying only unitaddr, you do not need parentheses.

Space for the input work queue must have already been allocated on the volume which is mounted on the specified device.

You need not specify a 3-character unit address if one of the following conditions exists:

- The SYS1.SYSJOBQE data set is cataloged.
- The volume containing the SYS1.SYSJOBQE data set resides on the device that has been specified at the time of system generation.
- The SYS1.SYSJOBQE data set is contained on the system residence volume.

If the system is to format the input work queue prior to the first job initiation, you must specify ,F following or without the 3-character unit address. For example, Q=(unitaddr,F) specifies that the system is to format the input work queue on the volume residing on the direct access device referred to by unitaddr;

Q=(,F) specifies that the system is to format the input work queue either on the volume to which the SYS1.SYSJOBQE data set is cataloged, or on the volume that resides on the device specified at the time of system generation, or on the system residence volume.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.SYSJOBQE data set. If this data set is not cataloged, the system uses the values provided at the time of system generation, if any such values were provided. If no such values were provided, the system assumes that the data set is contained on the system residence volume.

The system issues an error message when either of the following conditions exists:

1. The volume to which the SYS1.SYSJOBQE data set is cataloged is not mounted.
2. The system cannot locate the SYS1.SYSJOBQE data set on the selected volume.

PROC=unitaddr
the address of the direct access device on which the volume that contains the procedure library resides.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.PROCLIB data set. If this data set is not cataloged, the system uses the values provided at the time of system generation, if any such values were provided. If no such values were provided, the system assumes that the data set is contained on the system residence volume.

The system issues an error message if either of the following conditions exists:

1. The volume to which the SYS1.PROCLIB data set is cataloged is not mounted.
2. The system cannot locate the SYS1.PROCLIB data set on the selected volume.

This parameter is used only in the initial SET command issued immediately after IPL and should only specify a device that is ready.

Note: When the system determines the location of the SYS1.PROCLIB data set, the data set is recataloged. Therefore, if you specify the SYS1.PROCLIB data set to reside on a volume other than the volume it normally resides on, the data set will be cataloged to that volume. On the next IPL, the PROC parameter will have to be used to reset the data set to its normal volume.

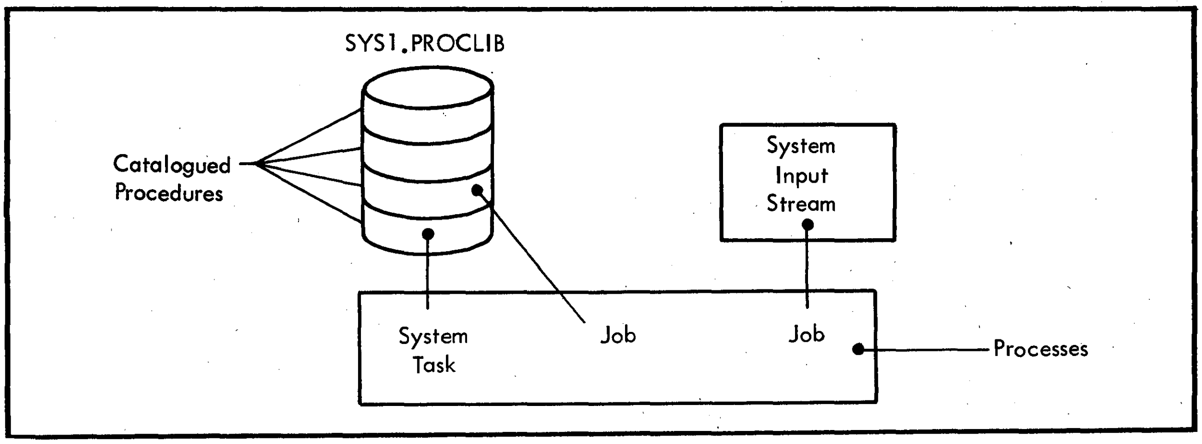
AUTO=characters
in systems with automatic START commands, whether you wish to retain any of those commands. For each automatic command printed out by the system, follow the equals sign by a Y if you want to retain the command, or by an N if you want to override the command. For example, if the system prints out S RDR, S WTR, S INIT, and you want to retain the automatic reader and writer but not the initiator, key in AUTO=YYN. If you want to reject all automatic commands, key in AUTO=NONE or AUTO=NNN.

This parameter is used only in the initial SET command after IPL. If this parameter is omitted, the system will use the automatic start commands.

MFT

START -- Start Process

Use the START command to start a process that is a cataloged procedure in SYS1.PROCLIB. A process can be a cataloged procedure that resides in SYS1.PROCLIB or a job from the system input stream. A catalogued procedure in the SYS1.PROCLIB can be either a system task or a job. If a cataloged procedure is a system task, the process name is the task name. If a cataloged procedure is a job, the process name is the job name.



To Stop or Cancel a Process Do One of the Following

System Task: When a system task is in allocation, stop it with the CANCEL command. Once a system task is running, stop it with the STOP command. There are two exceptions: one, an input reader will stop when it encounters an end-of-file; two, if you cancel an output writer it will stop writing the current job and start writing the next job.

Job: To stop a job use the STOP command. The stop command will work only if the programmer has set the proper indication to cause the job to respond to the STOP command. To cancel a job, use the CANCEL command. This will immediately terminate the scheduling or execution of a job. It will also stop the writing of the job's output.

Additional Format Conventions Used in the START Command

Positional Parameters: Positional parameters, such as devicename, volumeserial, and parmvalue in the START command, must be entered exactly in the order in which they appear. If you leave one out, put a comma in its place. You do not need to use a replacing comma if:

- The parameter to be left out is the last one in the series.
- All positional parameters following the absent one are also to be left out.
- All optional positional parameters are to be left out. Example:

```
START RDR.Pn,,DSNAME=IN
```

Keyword Parameters: Keyword parameters, such as DSNAME=name, can appear in any order. To leave one out, simply omit it; do not replace a missing keyword parameter with a comma.

Operation	Operand
START or S	procname.identifier [,devicename][,volumeserial][,parmvalue] [,keyword=option,...]

procname
    the name of a cataloged procedure which defines the process to be started. This procedure must reside in SYS1.PROCLIB. The name can be either an IBM assigned name or a user-assigned name provided by your system programmer.

IBM Assigned Names and the Processes They Describe

CRJE	Conversational Remote Job Entry
DSO	Direct System Output Processing
GFX	Graphic Job Processor or Satellite Graphic Job Processor
INIT	Initiator*
RDR	Input Reader (SYSIN data sets unblocked)
RDR400	Input Reader (SYSIN data sets in 400-byte blocks)
RDR3200	Input Reader (SYSIN data sets in 3200-byte blocks)
RJE	Remote Job Entry Subsystem
WTR	Output Writer

*INIT, though used as a procname to define an initiator, calls a system command-handling routine rather than a standard, IBM-supplied cataloged procedure.

When a START INIT command is entered in a system containing small partitions (any size between 8K and the size of the scheduler), the partition containing the initiator places the job into the specified (small) partition and initiates the job. You must start an initiator for each partition in which a job from the system input stream will run.

User Assigned Names and the Processes They Describe

A user-assigned name is any name your system programmer assigns to a cataloged procedure that resides in SYS1.PROCLIB. The processes they describe are the jobs that are run as a result of starting the user named procedure. In other words, you can start a job (process) from the console by starting a user-named procedure in the SYS1.PROCLIB. Use the process name from the START command to refer to a started job in other operator commands. The started job will be run as soon as the storage and the necessary devices have been allocated.

identifier
    one of the following:

ALL (used with the procname INIT) to indicate that the initiator is to be started in all partitions, if this is desired.

Pn where n is a partition number (0 through 51) to indicate the partition in which the process is to be started.

S to indicate that the reader or writer is to be system-assigned. Only one system-assigned reader or writer can be active in a system at any time.

The identifier identifies a specific process to the system. Use the identifier from the START command to enter STOP or MODIFY commands or to cancel the system task during allocation.

Example: To start a writer for job classes ABC, enter:

```
START WTR.P3,00E,,ABC
```

If you have to cancel the writer during allocation, enter:

```
CANCEL P3
```

To modify the writer, for example, to add class D to the classes to be processed, enter:

```
MODIFY P3,CLASS=ABCD
```

Note: With STOP or MODIFY, you can use the procname with the identifier. With CANCEL, you must use just the identifier.

Initiators with the same procname can have the same identifier. All other processes must have unique identifiers.

devicename
: the input or output device associated with an input reader or output writer. This can be either a unit address (such as 293) or a device type (such as 2400). If specified, devicename will override any corresponding unit specification in the cataloged procedure.

If you don't use this parameter but specify one or both of the parameters volumeserial and parmvalue, you must indicate the absence of the device parameter by a comma. Example:

```
START RDR.Pn,,123456
```

If you use devicename but do not use volumeserial and parmvalue, you do not need commas to indicate their absence. Example:

```
START RDR.Pn,282,DSNAME=IN
```

volumeserial
: the up-to-six-character serial number of a magnetic tape or direct access volume. If specified, this parameter will override any corresponding volume serial specification in the cataloged procedure. Do not specify a serial number for a direct access volume when starting a writer.

If you don't use this parameter but specify a parmvalue parameter, you must indicate the absence of the volumeserial parameter by a comma. Example:

```
START WTR,Pn,282,,A
```

No comma is necessary to indicate the absence of parmvalue. Example:

```
START RDR.Pn,282,111111,DSNAME=N
```

parmvalue
parameter values to be passed to the program receiving control as a result of the START command. If parmvalue contains any non-alphameric character, such as an equals sign, parmvalue must be enclosed in parentheses.

No comma is necessary to indicate the absence of parmvalue. If you are omitting all three parameters (devicename, volumeserial, parmvalue), you do not need any comma in their place.

```
START RDR.Pn,DSNAME=N
```

The particular form of parmvalue that may be used depends on the system process that is being started. The parmvalues that may be used with the system processes described by the standard procedures are explained below.

Input Reader -- parmvalue may specify a jobname. Use this parmvalue to cause forward spacing through the input stream so that processing begins with the named job. For example, the command S RDR.Pn,,,JOBX will start an input reader that will space forward until reaching the first statement of JOBX, and then begin processing.

Initiator or Output Writer -- parmvalue may specify class names when starting an initiator or output writer; use this parmvalue to limit the classes which the initiator or writer is to process.

The class name parmvalue may be, for an initiator, one to three characters or, for an output writer, one to eight characters which represent the job classes or output classes to be processed. The specified classes are treated on a priority basis where the left-most character indicates the queue to be processed first. A class name parmvalue entry overrides all class names specified in the cataloged procedure.

The initiator and output writer will also accept a class name as a series of characters in parentheses and separated by commas. For example, S WTR.P1,00E,,(A,B,C). But the form ABC is preferred to (A,B,C).

Job class identifiers specified in a START INIT command override job classes previously assigned to a particular partition. For example, the command S INIT.Pn,,,B will start an initiator that will select for execution only jobs of job class B (specified via the CLASS parameter in the JOB statement). The command S WTR.Pn,,,AB will start an output writer that will attempt to write only output of output class A (specified via the MSGCLASS parameter in the JOB statement or the SYSOUT parameter in the DD statement). Whenever no output of class A is ready, the writer will attempt to write output of output class B. If no output of either output class A or B is ready, the writer will wait rather than process output of any other output class.

Direct System Output Processing -- parmvalue may specify job classes(s) and an output class when starting direct system output processing.

JOBCLASS=jjj
job class(s) which may use Direct System Output Processing (DSO). From one to eight job classes can by specified (A-O).

If this parameter has been omitted from the DSO START command, and the job class parameter was omitted from the DSO procedure, then the partition's present job classes will be used.

OUTCLASS=s
the system output class that is to be processed by Direct System Output Processing. One system output class can be specified (A-Z,0-9). This parameter is required if the system output class was omitted from the DSO procedure. It will also override an output class specified in the DSO procedure.

Graphic Job Processor or Satellite Graphic Job Processor -- parmvalue specifies a new value for and overrides the corresponding GFX option value selected at system generation. You can override just one or more or all of the GFX option values specified at system generation. For a description of the graphics parmvalue, see "How to Specify Use of the Graphic Job Processor or Satellite Graphic Job Processor" later in this chapter.

Remote Job Entry Subsystem -- For a description of parmvalues that may be used with a Remote Job Entry subsystem see the chapter "Remote Job Entry and Conversational Remote Job Entry."

keyword=option
any appropriate keyword syntax allowable on a DD statement or any symbolic parameter keyword that is defined in the procedure specified in the START command. (For detailed information on these keywords, refer to IBM System/360 Operating System: Job Control Language Reference.) Symbolic parameter keywords can be provided by your system programmer.

If such DD keyword parameters are specified, they will override the corresponding parameters on the DD statement for the input or output device in the cataloged procedure. If the devicename positional parameter is used, the UNIT keyword cannot be used. If the volumeserial positional parameter is used, the VOLUME keyword cannot be used.

If the input device is a disk, you must use the keyword DSNAME=name to specify the correct data set. If the data set is not cataloged, you must use either the volumeserial parameter of the START command or the keyword VOLUME=SER=volumeid. But whether or not the data set is cataloged, you must specify DISP=OLD when using the IBM-supplied reader procedures, unless you want the data set to be deleted.

You cannot specify a keyword=option parameter in a START command for a graphics interface task.

STOP -- Stop Process or Continual Display

Use the STOP command to stop the operation of:

- A Process (a system task, initiator, or job started by the START command, or a job from the system input stream).
- A continual display of job names.
- A continual display of non-temporary data set names (in mount and keep-type demount messages).
- A continual display of available space on a direct access volume (in demount messages).
- A continual display of names of data sets and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG.

When you use the STOP command to stop a process, the process does not stop immediately. Instead, the system begins the stopping after the process finishes handling its current task. If the process you are stopping is a job, the STOP command will work only if the programmer has set the proper indicator to cause the job to respond to the STOP command.

After a reader is started, it must process one job before a STOP RDR command will take effect, In addition, input readers stop automatically when an end-of-file condition is encountered. If a MOUNT message for tape has been received, the mount request must be satisfied before the reader will stop.

Operation	Operand
{STOP / P}	{[procname.]identifier / INIT.Pn / JOBNAMES / DSNAME / SPACE / STATUS}

procname
: the name of a process to be stopped (e.g., a RDR, WTR, or a job).

identifier
: the identifier of the process as defined by the START command or the name of a job from the system input stream.

  If the process to be stopped is a:

  - System task with a device allocated to it, use the device address as the identifier.
  - System task with no device allocated to it or a started job, use the process name (cataloged procedure name in SYS1.PROCLIB) as the identifier.
  - Job from the system input stream, use the job name as the identifier.

INIT.Pn
: the initiator to be stopped, where n is the partition number in which the initiator is running.

JOBNAMES
a continual display of the names of jobs, initiated by the JOBNAMES parameter of the MONITOR command, is to be ended.

DSNAME
the system is to stop the continual display of the names of non-temporary data sets, as initiated by the DSNAME parameter of the MONITOR command.

SPACE
the system is to stop the continual display, in demount messages, of the available space on a direct access volume. The display was initiated by the SPACE parameter of the MONITOR command.

STATUS
the system is to stop the continual display, at step end and job end, of the names and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG. The display was initiated by the STATUS parameter of the MONITOR command.

SWAP -- Swap Volumes for Dynamic Device Reconfiguration (DDR)

Use the SWAP command to request Dynamic Device Reconfiguration of two volumes, or in response to a system request for Dynamic Device Reconfiguration. (DDR must be specified at system generation time.)

The command can be entered to use DDR to exchange any two demountable volumes (on 2311's, 2314's, 2321's, or 2400's), the request can come from you or the system. See "How to Use Dynamic Device Reconfiguration" in the chapter "General Operating Techniques."

Operation	Operand
{SWAP | G}	{OFF | ON | xxx,yyy}

OFF
: system-initiated Dynamic Device Reconfiguration is stopped and all permanent errors will bypass DDR processing.

ON
: activates system-initiated Dynamic Device Reconfiguration.

xxx
: the primary channel unit address of the device "from" which a volume is to be swapped.

yyy
: the primary channel unit address of the device "to" which a volume is to be swapped.

Note: The devices specified by xxx and yyy must be of the same device type. xxx and yyy may specify the same device to permit maintenance of a particular volume or device.

UNLOAD -- Prepare Volume for Demounting

You normally use the UNLOAD command to remove a volume previously mounted in response to a MOUNT command, but you can use UNLOAD to remove any tape or removable direct access volume. The UNLOAD command causes a volume on an input/output device to be prepared for demounting.

When the volume is ready to be demounted -- when all job steps using it have terminated -- you will receive message IEF282I device NOW UNLOADED. If SYSOUT data sets are allocated to the volume, you will not be able to unload until a system writer has processed the data sets.

Operation	Operand
{UNLOAD / U}	unitaddr

unitaddr
: the unit address of the input/output device to be prepared for demounting. When using this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unit address for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

VARY -- Vary Status of Device or VARY Status of a Path

Use the VARY command to place input/output devices (other than a communication line) into an online or offline status. This command is used with the graphic job processor (GJP) and satellite graphic job processor (SGJP) to designate the display units and graphic subsystems to be available or unavailable for graphic job control operations.

If Alternate Path Retry is in the system, use the VARY command to place a channel/control unit path online or offline. The last path to a device cannot be varied offline, and TP paths cannot be varied at all.

See "The VARY Command in Systems with MCS" in the chapter "Operator's Consoles" for instructions on how to use this command to perform functions related specifically to MCS.

```
┌─────────┬──────────────────────────────────────────────────────────────┐
│Operation│Operand                                                       │
├─────────┼──────────────────────────────────────────────────────────────┤
│ {VARY}  │                            ⎧,ONLINE                 ⎫        │
│ { V  }  │                            ⎪,OFFLINE                ⎪        │
│         │{ALL                      } ⎪,ONGFX                  ⎪        │
│         │{(unitaddr[,unitaddr]...) } ⎨          ⎡  ⎧F⎫⎤       ⎬        │
│         │                            ⎪,OFFGFX   ⎢, ⎨M⎬⎥       ⎪        │
│         │                            ⎪          ⎣  ⎩S⎭⎦       ⎪        │
│         │                            ⎪,PATH,xxx,{ONLINE }     ⎪        │
│         │                            ⎩          {OFFLINE}     ⎭        │
└─────────┴──────────────────────────────────────────────────────────────┘
```

ALL
: for GFX, all units named for GJP or SGJP during system generation be made available (ONGFX) or unavailable (OFFGFX) for graphic job control operations.

unitaddr
: the unit address of the input/output device whose status is to be changed. You don't need parentheses around only one unitaddr. For GFX, unitaddr specifies the 2250s (for GJP) or telecommunication lines (for SGJP) that are to be made available or unavailable for graphic job control operations. For GJP or SGJP, do not specify more than 15 devices.

  When both GJP and SGJP are functioning in an operating system, unit addresses of 2250s and telecommunication lines can be intermixed.

  For PATH, the unitaddr is the channel unit address of the primary path to a device. There may be as many as four paths to a device, each path having a channel unit address. When you VARY the path of a device use the lowest channel unit address associated with the device. Only one channel unit address may be specified with each VARY PATH command.

  To vary the status of an entire 2321 data cell, use its three-character unit address -- 263, for example. To vary the status of a particular 2321 bin, use its five-character address -- 263/8, for example, where the 8 is the number of the particular bin being addressed.

ONLINE
: the input/output devices identified in this command are to be made available for allocation by the job scheduler to problem programs. If a device is made ONLINE from an OFFLINE status, and you want the system to recognize a volume that was mounted on the device when you took it OFFLINE, use the MOUNT command to identify the volume to the system.

OFFLINE

specifies that the input/output devices identified in this command are to be removed from the recognition of the job scheduler, and that any further allocation of the devices to problem programs and system tasks is to be prevented. If the devices are in use (allocated to a problem program or to an input reader or output writer), their status is not changed until all the current users have finished with the devices. When the status is changed to offline, you will receive a message.

A device can be removed from the offline status only by a subsequent VARY command or, if an appropriate system message is received, by issuing REPLY id,'unitaddr'.

MFT

ONGFX

for GFX, the 2250s and/or telecommunication lines identified in this command are to be made available for graphic operations.

Note: This command does not vary the device online. To do this, use the VARY ONLINE command.

OFFGFX

for GFX, the 2250s and/or telecommunication lines identified in this command are to be made unavailable for graphic operations.

Note: This command does not vary the device offline. To do this, use the VARY OFFLINE command.

F

(used with OFFGFX) the operating system jobs being processed for the specified 2250s and/or 1130/2250 subsystems are to be ended immediately without notifying the users that this ending has occurred (F = fast stop). A printed record of previous job control operations cannot be obtained. The operator should notify the user at the 1130/2250 subsystem.

M

(used with OFFGFX) the operating system jobs being processed for the specified 2250s and/or 1130/2250 subsystems are to be ended immediately and the users requested to log off (M = medium stop). A printed record of previous control operations can be obtained.

S

(used with OFFGFX) the operating system jobs currently being defined for or processed for the specified 2250s and/or 1130/2250 subsystems are to be allowed to reach normal or abnormal end before the users are requested to log off (S = slow stop). A printed record of previous job control operations can be obtained. The operand is underscored to indicate that the system assumes S if none of the parameters F, M, and S has been specified. 0

PATH

this is a VARY PATH request, (to be used only if Alternate Path Retry was specified at System Generation Time.)

xxx

the channel unit address of the path to be varied.

ONLINE

the specified path is to be logically added to the system and is made available for allocation by the job scheduler to problem programs.

OFFLINE
: the specified path is to be logically removed from the system. Any further allocation of the path to problem programs is prevented. A path can be removed from offline status only by a subsequent VARY PATH command or a re-IPL.

Note: A volume that has been reserved through use of either a PRESRES entry or a user-issued MOUNT command cannot be moved or removed from the system by varying offline the device on which it resides. A reserved volume can be moved or removed only if you issue an UNLOAD command, or if the system issues an action-type message (such as IEF234A), instructing that the volume be demounted.

Example 1:

```
VARY ALL,ONGFX
```

This example makes available for graphic job control operations all 2250s and 1130/2250 subsystems that were identified by the UNITS parameter of the GJOBCTL macro instruction during system generation.

Example 2:

```
VARY (1E0,024),OFFGFX,S
```

This example makes the 2250 unit with the address 1E0 and the subsystem attached to telecommunication line 024, which were made available by Example 1, no longer available to GJP or SGJP after the current job has been processed.

WRITELOG -- Write Out System Log

Use the WRITELOG command to have the system log written out or to close the system log.

Operation	Operand
{WRITELOG / W}	[classname / CLOSE]

MFT

classname
: the name of the system output class you want to use to write out the contents of the system log. If you do not specify classname, the default value is class L, unless your installation modified the default value during system generation.

CLOSE
: the system log is to be closed and the log function is to be discontinued. The system log can be reopened only by IPLing the system.

WRITELOG schedules the writing of the system log, but you must start a system output writer (if one does not already exist) to actually write the data set to a SYSOUT device. Also, a job must terminate after the WRITELOG command is entered for the system log to be scheduled for output.

Example: If you want the log to be written as output class D, enter WRITELOG D. If no output writer for class D exists, enter S WTR.Pn,00E,,D to actually write out the system log.

Summary of Special MFT Operating Techniques

This section covers special procedures that apply to MFT only.

After reading this section, skip to the chapter "General Operating Techniques."

HOW TO EXTRACT A JOB FROM TAPE INPUT STREAM

To extract one particular job from a tape input stream, enter:

S RDR.Pn,,,jobname

and then

P Pn

before readying the tape. Jobs preceding "jobname" will be skipped; "jobname" is entered into the queues. Then the STOP is processed.

Note: Devicename or volumeserial or both can be added to the START command. Pn (the identifier) is all that is needed to stop the reader.

HOW TO RUN JOBS THAT UPDATE SYSTEM DATA SETS

Do not run other jobs concurrently with jobs that update system data sets (SYS1.LINKLIB, SYS1.SVCLIB, and SYS1.PROCLIB); for example, don't run jobs that may try to use a cataloged procedure with a job that is updating that procedure. Run jobs that update system data sets as follows:

- Make sure no other jobs are active in the system; use the HOLD command to prevent the initiation of jobs on the job queue.
- Stop all readers, writers, and initiators.
- Place the jobs that are to update the system data sets in an input device.
- Start a reader to that device.
- Start one initiator when the reader stops after processing all the input.
- Start normal processing -- that is, start required readers, writers, and initiators -- when the first initiator stops after completing all the jobs. Use the RELEASE command to free the jobs held on the job queue by your earlier HOLD command.

HOW TO DETERMINE SYSTEM STATUS

Information describing current system activity is necessary to detect potential problems, and whenever possible, to prevent them from occurring. The commands listed below are particularly useful.

DISPLAY A Command

The DISPLAY A (active) command is your primary means of finding out what the system is doing. Use it frequently. Entering DISPLAY A causes each job name and associated stepname and the active system task and the partition in which it is running to be displayed on the console.

If one job seems to be running longer than it should (the job name continually appears on the console when the DISPLAY A is entered), this job may be contending for some resource such as:

- A device (check all operator messages; you may have missed a mount request).
- A data set (the initiator is waiting for a data set).

Regularly check the console sheet for messages that may need replies (the DISPLAY R command will help you determine if there are unanswered messages). It may be necessary to cancel a job if an interlock condition exists. One job may be waiting for a data set that has been allocated to another job, and, in turn, the first job has a data set that the second job is waiting for. Neither job can continue until one of them relinquishes its data set. This is known as a system interlock. You should cancel one of the jobs and reenter it into the input stream.

DISPLAY JOBNAMES Command

This command gives you continuing data on jobs starting and stopping. After DISPLAY JOBNAMES is entered, message IEF403I jobname STARTED is issued during initiation of each job. During termination, message IEF404I jobname ENDED is issued to notify you that the job has ended (except when the job terminated abnormally -- then you get a diagnostic message). Allocation of unit record equipment is also displayed when each job step starts.

DISPLAY N Command

Jobs assigned an unserviced CLASS parameter on the JOB card remain on their input queue indefinitely. Therefore, you should enter DISPLAY N periodically to obtain a listing of the jobs on the hold, input, and output work queues.

If there are jobs on input work queues that have no partitions assigned to service them, a partition can be redefined to service these classes (see "Redefining Partitions" earlier in this chapter.) In addition, if you want to find out whether a job previously entered in the input stream has been run, enter DISPLAY N. If the job is not on any of the queues, it is either being run, or has already been run.

DISPLAY Q Command

If you have not started a resident writer (by entering a START command for a partition that has a job class identifier of WTR), DISPLAY Q should be entered periodically to determine the status of the output queues. If the number of entries in the output queues increases steadily, you should try to clean out the queues. This can be done by starting one or more writers in problem program partitions, specifying the output classes that have the most work on their queues.

When the number of entries on the queues has been reduced (enter DISPLAY Q again), you can STOP the writer, and the partition will again be available to process user jobs.

If you have started a resident reader (by entering a START command for a partition that has a job class identifier of RDR), DISPLAY Q should also be entered periodically to determine the status of the input queues. If the number of entries in the input queues increases steadily, you can redefine the resident reader partition to a problem program partition (with valid job class identifiers, A-O) that will process the user jobs. You can also issue a STOP command for any readers that you started in problem program partitions. This makes the partition available to process user jobs.

When the number of entries on the input queues has decreased, you can restart the reader in a problem program partition, or redefine the problem program partition back to a resident reader.

Display U Command

The DISPLAY U command provides you with information about each device in the system. When DISPLAY U is entered, the message:
IEE450I UNIT STATUS TIME=hh.mm.ss will be entered on your console followed by a heading line. If the offline parameter is not specified in the DISPLAY U command, the heading line will contain: UNIT STATUS VOLSER VOLSTATE. If the offline parameter is specified, the heading line will contain: OFFLINE DEVICES, and seven UNIT TYPE headings.

Each subsequent line contains information for two devices or seven devices if offline status was requested.

DISPLAY CONSOLES Command

The DISPLAY CONSOLES command gives you the status of each operator's console in a system with the multiple console support (MCS) option. The command can be entered by any active console; only the console that enters the command receives the display. The display identifies each console and its alternate, indicates whether the console is active or inactive, and shows the command groups and routing codes that are valid. The display also identifies the hard copy log, shows which routing codes the log is receiving, and shows whether it is receiving commands and their responses.

Use the DISPLAY CONSOLES command to give you the system's console configuration and to help you construct alternate chains. An alternate chain is created when several consoles are logically connected through their alternate consoles. For example, the alternate to console 1 is console 2, the alternate to console 2 is console 3, the alternate to console 3 is console 1. The chain used in this example forms a loop, but this is not required. If console 2 becomes inoperative, the operating system examines the status of console 3. Console 3, if active, assumes the routing codes and command of authority of console 2. If console 3 is inactive, the status of console 1 is examined. This examination continues until an active console is located or the entire loop is examined. If an active console cannot be found, the master console is used.

Once the console configuration has been displayed and you have decided which changes need to be made, use the VARY command to modify the configuration. A description of who can modify the configuration and how to enter the VARY command is part of the chapter "Operator's Consoles."

Low System Activity

Low system activity (indicated by the wait light frequently being on) can be due to the following conditions:

- Lack of system direct access space. Readers that do not read to end-of-file may have been delayed waiting for direct access space for the jobs and their associated data. Also, jobs that are waiting for allocation frequently cannot be assigned system output space. Both conditions can occur in normal system operation. They persist until space is made available as a result of writing the output data sets.

  If you have started several readers and writers in the system, the above delays may be reduced by entering STOP commands for readers to make direct access space available for output data sets. STOP commands can also be entered for writers which make direct access space available for initiation of new jobs.

- Mount Requests. When messages are written on the console to mount certain volumes, mount these volumes immediately. If you cannot do this, because of a lack of tape drives or disk drives, cancel the job.

- Small Partition Interlocks. If the system has only one scheduler-size partition and the system enters an enabled wait state, it may be in a small partition interlock; i.e., the initiator may be waiting for a data set that has been allocated to a job in a small partition. You will receive message IEF244I jobname UNABLE TO ALLOCATE. You should cancel the job in the small partition, and reenter it in the input stream.

  In a Shared DASD system where two or more systems are sharing the same device, an interlock can be caused by a programming error. This situation is usually characterized by all CPUs being in an enabled wait state. In this case, it may be necessary to cancel a job in only one of the systems to resolve the interlock. If not all CPUs are in an enabled wait state, proceed with caution; a system could simply be waiting for a device to be released.

HOW TO CONTROL JOBS THROUGH HOLD AND RELEASE COMMANDS

In MFT, in addition to the control provided by the job class facility, you can use HOLD and RELEASE commands to control scheduling of jobs. All jobs of one type (e.g., production) initially can be put on the HOLD queue in two ways:

1. Use the TYPRUN=HOLD parameter on the JOB statements.

2. After you START a reader, enter a HOLD command for the jobs that you want to be put on the HOLD queue. Do this before you enter any START INIT command.

You can then RELEASE these jobs one at a time (e.g., when the first production job is terminated, the next can be released). DISPLAY JOBNAMES can be extremely helpful in determining when jobs end, even though production-type jobs usually end in a predictable manner (rewinding tapes, receiving KEEP messages, etc.).

The HOLD command remains in effect until a RELEASE command is entered, even if the system is restarted. Jobs that were held individually or as part of the queues are still held after the system is restarted. They must be made available by entering the appropriate RELEASE command.

Be careful of the sequence of your specific commands when controlling job class execution through HOLD Q, RELEASE Q, and S INIT. For example, the command sequence: HOLD Q; S INIT.P1,,,GFEDCBA; RELEASE Q may not result in a G class job being executed first. Instead, the initiator may pick up one A or B class job prior to beginning execution of the job classes in the specified order G through A. This is because job classes G, F, etc., have not yet been released by the RELEASE Q command when the initiator looks for work in response to the S INIT.P1,,,GFEDCBA command.

HOW TO SPECIFY USE OF THE GRAPHIC JOB PROCESSOR OR SATELLITE GRAPHIC JOB PROCESSOR

MFT

The graphic job processor (GJP) is a program that displays and requests job control information at a local IBM 2250 Display Unit. The satellite graphic job processor (SGJP) is a program that displays and requests job control information at a remote 1130/2250 subsystem (attached to a System/360 via a telecommunication line).

The user responds to the displays by entering requested information, by selecting appropriate options, or both. GJP and SGJP convert the entered information into job control statements and pass them to the operating system to initiate the job.

System Considerations When Using GJP or SGJP

A separate partition must be available for the graphics interface task (GFX), for each 2250 used with GJP, for each 1130/2250 subsystem used with SGJP, and for each initial processor (when used with SGJP to activate telecommunication lines). The GFX partition and each initial processor partition must contain at least 10K bytes. Each 2250 and each subsystem partition must contain at least 60K bytes and should be time-sliced. Initial processor partitions should be included in the time slice group with the 2250 and subsystem partitions. Time-sliced partitions must be contiguous.

After GFX has started, no other jobs should use the foreground message class assigned to GJP or SGJP (see "GFX Options"). After GFX has stopped, the partitions for GFX, GJP, and SGJP are available for other uses.

Starting GJP or SGJP

To use GJP or SGJP, first start the graphics interface task (GFX).

$\begin{Bmatrix}\text{START}\\ \text{S}\end{Bmatrix}$ GFX.Pn[,,,(gfx=option,...)]

Pn is a partition with a priority greater than or equal to the priority of GJP or SGJP. The gfx=option parameters are described under "GFX Options" following.

Note: When using the START GFX command for SGJP, you may need the CLSI parameter to designate the job class for the initial processor. If a user wants initial processors to activate telecommunication lines, he will set aside as many partitions as initial processors during system generation or initial program loading, assigning the same job class to each partition. You must specify this job class in the CLSI parameter.

After issuing the START GFX command, enter a VARY ONGFX command to designate the 2250s and/or telecommunication lines to be made available.

$\begin{Bmatrix}\text{VARY}\\ \text{V}\end{Bmatrix}$ (unitaddr[,unitaddr]...),ONGFX

Keep in mind that a VARY ONGFX for SGJP merely activates the telecommunication line between the System/360 and the 1130/2250 subsystem. Then there's a waiting period until a user logs on -- thus letting the initial processors or SGJP routines sit in main storage with nothing to do. To avoid wasting storage (processors take 10K bytes each and SGJP routines take 60K bytes), don't issue VARY ONGFX until a user is ready to begin SGJP operations at the subsystem.

Stopping GJP or SGJP

At any time, enter a VARY OFFGFX command to designate the 2250s and/or telecommunication lines that are not to be used any longer.

$$\left\{\begin{array}{l}\text{VARY}\\ \text{V}\end{array}\right\}\text{(unitaddr[,unitaddr]...),OFFGFX}\left[,\left\{\begin{array}{c}\text{F}\\ \text{M}\\ \text{S}\end{array}\right\}\right]$$

If you include the S (slow stop) parameter in your command (explicitly or by default), you let any jobs being defined or processed reach normal or abnormal completion before the unit becomes unavailable.

If you include the M (medium stop) or F (fast stop) parameter, you force immediate termination of GJP or SGJP activity for the specified unit. (See the VARY command for the effects of the M and F parameters.) If you've already issued a command with an S parameter and feel that an M or F stop is necessary, issue another VARY OFFGFX command with an M or F.

Note: Under some conditions, such as during certain displays at the 1130/2250 subsystem, a fast or medium stop (VARY OFFGFX,F or M), will not take place immediately.

After a fast stop (VARY OFFGFX,F), do not issue VARY ONGFX until SGJP has been restarted in the 1130/2250 subsystem.

To stop the graphics interface task (GFX), use a STOP GFX command. The task will end when all units have logged off. If GJP or SGJP activity continues and you want the STOP command to take effect quickly, issue the appropriate VARY OFFGFX commands (even though you've already issued the STOP GFX command).

GFX Options

These are special parmvalue parameters of the format gfx=option. Use them in any sequence within the parmvalue field of the START GFX command to override options selected during system generation. Always enclose GFX options in parentheses.

PRT=printer output class
PCH=punch output class
MSGF=foreground message class
MSGB=background message class
PRIF=foreground priority
PRIB=background priority
CLSF=foreground job class
CLSB=background job class
CLSI=initial processor job class

PRT=printer output class
: specifies the class name that identifies the SYSOUT class for printed output from jobs defined at a 2250 by means of GJP or SGJP.

PCH=punch output class
: specifies the class name that identifies the SYSOUT class for punched output from jobs defined at a 2250 by means of GJP or SGJP.

MSGF=foreground message class
specifies the class name that identifies the SYSOUT class for messages pertaining to foreground jobs defined at a 2250 by means of GJP or SGJP.

MSGB=background message class
specifies the class name that identifies the SYSOUT class for messages pertaining to background jobs defined at a 2250 by means of GJP or SGJP.

PRIF=foreground priority
specifies the scheduling priority (0-13) for foreground jobs defined at a 2250 by means of GJP or SGJP.

PRIB=background priority
specifies the scheduling priority (0-13) for background jobs defined at a 2250 by means of GJP or SGJP.

CLSF=foreground job class
specifies the job class (A-O) for GJP or SGJP and for foreground jobs defined at a 2250 by means of GJP or SGJP. This job class determines the partitions in which GJP or SGJP and foreground jobs are executed, thereby establishing their dispatching priorities. This job class should be assigned to at least one partition for each 2250 that will use GJP and one partition for each subsystem that will use SGJP.

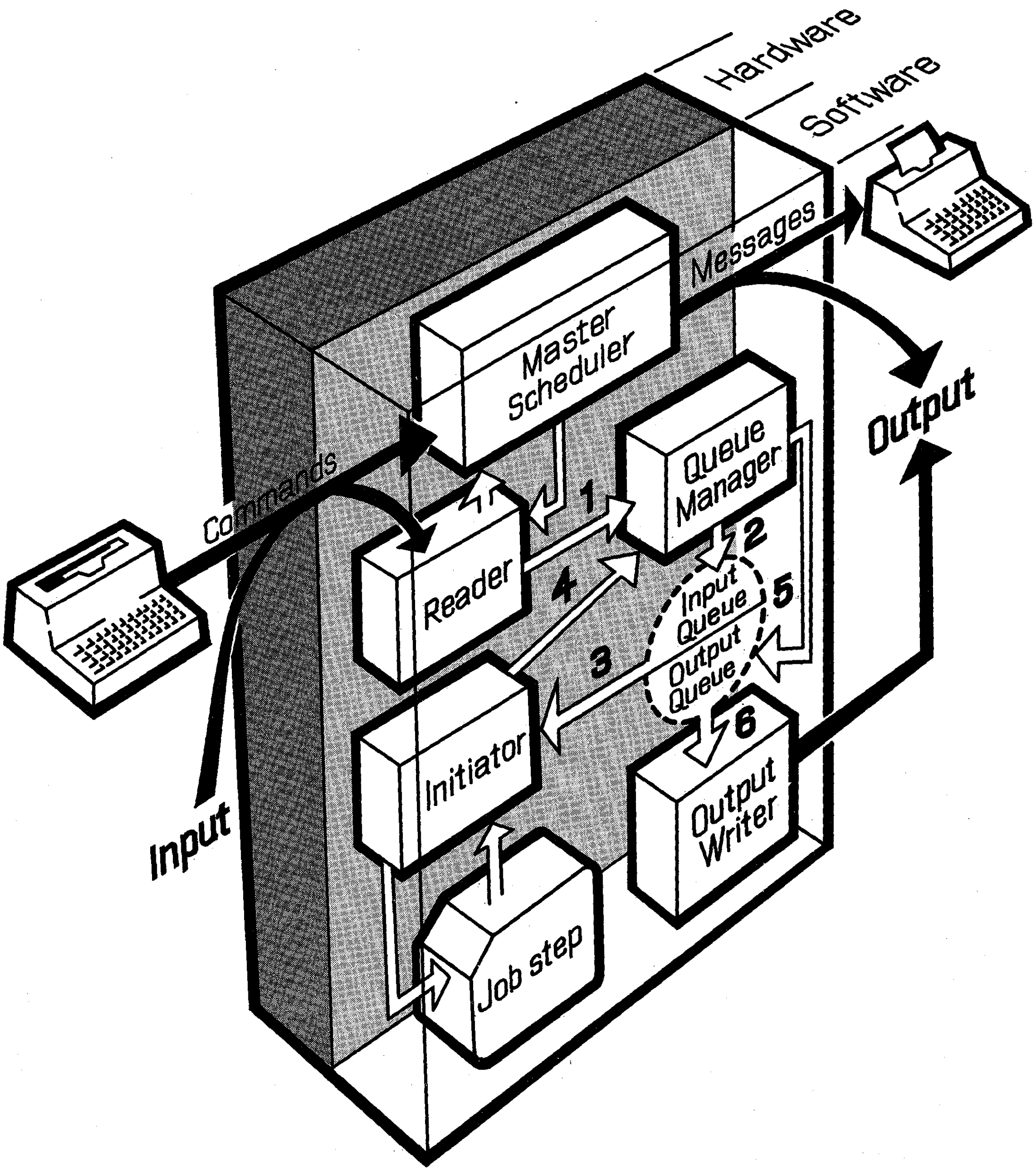
CLSB=background job class
specifies the job class (A-O) for background jobs defined at a 2250 by means of GJP or SGJP. This job class determines the partitions in which background jobs are executed, thereby establishing their dispatching priorities.

CLSI=initial processor job class
(used with SGJP) specifies the job class (A-O) for the initial processor, which activates telecommunication lines. This job class determines the partition in which the initial processor is executed, thereby establishing its dispatching priority. The priority should not be lower than that for the foreground job class. The partition must contain at least 10K bytes.

MFT

SYSTEMS WITH MVT

Contents



How the Scheduler Works in Systems With MVT

Systems With MVT

The job scheduler in a system with MVT has a queue manager and the ability to start multiple readers, writers, and initiators.

The queue manager controls the input and output work queues. The input queue is a list of jobs in the order in which they are to be processed internally. The output queue is a list of data sets in the order in which they are to be handled by the output writer.

Instead of simply running one after the other, jobs in this system are rearranged and stored in a queue to run according to their priorities.

Later, an initiator takes each job from the input queue on a priority basis and starts it. When the job is finished, the initiator puts requests for system output in the output queue for later handling by an output writer.

Several readers can put job control records in the input queue while, at the same time, the initiators are starting job steps and the output writers are handling system output.

The number of input streams existing at one time depends on the number of START RDR commands you issue -- one command starts one reader and one input stream, two commands start two readers and two input streams, and so on.

Each reader reads data and control statements, places the information from them in proper format for later use by the initiator, and arranges the jobs in priority sequence in the input queue.

An initiator takes jobs from the input queue, assigns I/O devices for each data set, sends you I/O mounting messages, and introduces each job step to the supervisor as a separate "task" -- a unit of work to be done at the same time as other units of work.

When the job step or job ends, the initiator removes it from the system and releases the no-longer-needed I/O devices. At the end of the job, the initiator completes information in the output queue for the output writer to handle. The initiator then goes on to the next job in the queue.

Like the reader, the number of initiators in the system depends on your commands -- one initiator is started for each START INIT command.

The degree of multijobbing being done is thus within your control -- you start many input streams or few, and start many job steps or few, depending on your installation's needs.

An output writer, activated by your command, START WTR, writes system output data sets, according to job priority and output class, on external devices such as printers and punches, depending on specifications given in commands or in job control language in an input stream.

The number of output streams depends on the number of output writers you start.

Starting, Stopping, and Restarting the System

System operations are controlled mainly through a console I/O device. You give commands to the system, and receive messages from it, through the console. For a discussion on the various types of console configurations see the chapter "Operator's Consoles."

STARTING THE SYSTEM

Starting the system includes initial program loading (IPL), readying the nucleus, and readying the scheduler.

Initial Program Loading

Initial program loading is a procedure carried out at the beginning of a shift, after a power-on following an electrical shutoff, after malfunctions that require reloading the control program into main storage, after scheduled maintenance, and as part of switching from one system to another.

Begin initial program loading by selecting the direct access storage device on which the operating system resides: set the three LOAD UNIT switches on the control panel to the proper unit address (made up of the channel, control unit, and device numbers); then hit the LOAD key on the panel.

Hitting the LOAD key turns off the MANUAL light, turns on the LOAD light, and starts reading the IPL program from the input device.

After the IPL program is read into lower main storage, control is passed to it, and the LOAD light turns off. If either the reading operation or the passing of control is unsuccessful, the CPU pauses and the LOAD light stays on.

When the IPL program gets control, it loads the nucleus of the control program into main storage.

The IPL program loads a standard, or primary, nucleus unless you cause it to load a secondary nucleus. For a description of this procedure, see "How to Load a Secondary Nucleus" in Operator's Procedures, GC28-6692.

After the nucleus is loaded, control is given to a nucleus initialization program (NIP).

If the IPL program does not finish successfully, or if I/O errors occur while NIP is running, the WAIT light turns on and an error code is placed in the low-order 12 bits of the program status word (PSW).

Whenever the WAIT light turns on without a message, display the PSW, note the error code, and follow the instructions for that code given in the publication IBM System/360 Operating System: Messages and Codes.

Readying the Nucleus

The nucleus initialization program (NIP) does general preparatory work for the system. If the communication option was specified at the time the system was generated, you will receive a message, IEA101A SPECIFY SYSTEM PARAMETERS, requesting any changes.

If you receive this message, your system programmer may ask you to alter one or more options, such as the BLDL option, the RAM option, or the RSVC option.

Explanations for the various options, and instructions on how to alter the options, are given in the Operator's Procedures, GC28-6692, under the heading "How to Specify System Parameters."

If no changes are to be made, issue REPLY id,'U', or simply signal EOB.

If your system includes the multiple console support (MCS) option, you may receive message IEA150A SPECIFY HARDCPY. Instructions on replying to this message are given in the chapter "General Operating Techniques." under "How to Specify Hard Copy." Only the master console is initialized and active during NIP.

If an error other than an I/O error occurs during the running of NIP, the WAIT light turns on, and you will receive a message identifying the error. No message is sent if the system console is not ready, but a code can be found in the low-order 12 bits of the current PSW as the system waits.

After NIP completes its preparation of the system, it passes control to the master scheduler and the remaining consoles and the system log will be initialized. You'll receive a READY message from the system and the WAIT light will turn on.

Readying the Scheduler

System/360 SET Command: When initial program loading is complete and the system is ready to run, the primary console (or the MCS master console) will receive a READY message and the WAIT light will go on. You may then enter commands to start the job scheduler, which in turn begins the flow of work through the system.

Your first command should be a SET command specifying the date and the time of day. Optionally, SET can specify the names of the devices for the input queue and a procedure library (SYS1.PROCLIB), and can also specify input queue formatting.

Normally, the formatting parameter Q=(unitaddr,F) can be left out after the first IPL, causing the scheduler to use the input queue as it was formatted earlier.

You will receive message IEF423A SPECIFY JOB QUEUE PARAMETERS after you issue the SET command with the formatting parameter Q=(unitaddr,F). Instructions telling how to answer this message are given under the heading "How to Specify Job Queue Parameters" in the Operator's Procedures, GC28-6692.

System/370 SET Command: When the nucleus is initialized and the system is ready to run, you will receive the READY message, and the WAIT light on the system control panel comes on. After the READY message you will receive one of two messages:

IEE114A DATA=yy.ddd, clock=hh.mm.ss REPLY WITH SET PARAMETERS OR U

or

IEE116A TOD CLOCK INVALID - REPLY WITH SET PARAMETERS

Message IEE114A displays the date and time of day as they are in storage at the time of the message. If the date and time are satisfactory and no other SET command parameters (Q, PROC, and AUTO) are to specified enter r id,'U' , where r is the REPLY command id is the message identifier, and ,'U' means there are no changes.

If the date or time is to be changed, or if additional parameters are to be entered, reply with the information to be changed as text in the REPLY command. For example, to change the time to seven-thirty P.M., enter r id,'CLOCK=19.30.00'.

If you receive message IEE116A, your reply must include at least the date; you may enter the time and other SET command parameters at your discretion.

You must depress the Time of Day Clock Enable Switch on the control panel immediately after entering the date or time in response to either message, or the clock cannot be set with the new information. If you do not depress the switch within approximately 15 seconds after you enter the REPLY command, you will receive the message:

IEE117A INTERVENTION REQUIRED ON TOD CLOCK ENABLE SWITCH

the message will be repeated approximately one minute later if the Time of Day Enable Switch has not been depressed. If, approximately one minute after you receive the message the second time, you have not depressed the switch you will receive the following messages:

IEE119A SET PARAMETER(S) NOT ACCEPTED - ENABLE SWITCH NOT DEPRESSED
IEE112I RESPECIFY SET PARAMETERS TO INITIALIZE SYSTEM
IEE114A or message IEE116A

After receiving the above messages you must enter the REPLY command again. This time be sure to depress the Time of Day Enable Switch before the 2 minute and 15 second time limit elapses.

If you are setting the Time of Day Clock at a time other than IPL time, do not use the REPLY command; use the SET command. When you enter the SET command to set the Time of Day Clock, you have approximately 2 minutes and 15 seconds to depress the Time of Day Enable Switch. If the allotted time elapses and you have not depressed the switch, you will receive the message:

IEE119I SET PARAMETER(S) NOT ACCEPTED - ENABLE SWITCH NOT DEPRESSED

You must now enter the SET command again and don't forget to depress the Time of Day Enable Switch before the 2 minutes and 15 seconds is up.

When the Time of Day Clock has been successfully set, you will receive the message:

IEE118I SET PARAMETER(S) ACCEPTED

START Commands For All Systems: The START RDR, START WTR, and START INIT commands must be issued if they have not been specified by your installation at system generation time.

If your installation has already specified START commands, I/O devices are automatically allocated to an input reader and an output writer, and the commands are written out on the console as if you had keyed them in yourself.

If you want to override automatic START commands in systems with MVT, use the SET command with the AUTO operand.

Examples:

1. To start a system that has automatic START RDR, START WTR, and START INIT commands:

   SET DATE=yy.ddd,CLOCK=hh.mm.ss

2. To start a system that has automatic commands, but to suppress them, and to start a reader, a writer, and an initiator:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss,AUTO=NONE
START WTR.identifier
START RDR.identifier
START INIT.identifier
```

3. To start a system that does not have automatic commands, and to start a reader, a writer, and two initiators, and to remove an I/O device from the system before processing:

```
SET DATE=yy.ddd,CLOCK=hh.mm.ss
START RDR.identifier
START WTR.identifier
START INIT.identifier
START INIT.identifier
VARY unitaddr,OFFLINE
```


To Improve Storage Use: The order in which you issue START commands can improve main storage utilization.

Particularly on systems with 256K of main storage, start the longest-running tasks first, the shortest-running ones last. This practice will reduce the possibility of main storage fragmentation.

Number of Initiators to Start: The number of concurrent initiators to be started should be carefully regulated according to the needs of your installation.

As a rule of thumb, if the wait light is on most of the time, you could probably start more initiators for greater performance. If the wait light is flickering rapidly, you are likely to have a good number of initiators going. If the wait light is out almost all the time, you may have too many initiators processing at the same time.

RESPONDING TO A PROBLEM PROGRAM FAILURE

When message IEA029I jobname stepname TASK REINSTATEMENT FAILED appears on the console with names other than MASTER SCHEDULER in the jobname stepname fields, a problem program has failed, but other tasks that are already scheduled will try to reach normal termination. Return the job to the programmer. Do not reallocate main storage, I/O devices, and data sets used by the failing task.

The failing problem program task may or may not have any impact on system operation. If, for example, a program that used a great deal of storage were to fail, this storage would not be released and might impact the system by severely limiting the amount of main storage available to new jobs. Use DISPLAY A to verify that the system is functioning normally. If system operation is impacted, you should quiesce the system (allow the system to come to a stop) by taking the following actions:

- Enter a HOLD Q command to stop the scheduling of jobs.
- Stop all readers and writers.
- After the scheduled tasks have completed, restart the system.

STOPPING THE SYSTEM

Enter a STOP command for each initiator in the system, a CANCEL command (optional) for each job, a STOP RDR for each reader, a STOP WTR for each output writer, and a STOP GFX for the graphics interface task (if such a task was initiated). A stop GFX command will not take effect until all devices assigned to the graphic job processor or satellite graphic job processor are no longer in use.

Wait for all activity to cease after these commands, and then issue the HALT EOD command to preserve the status of the system log data sets (if present) and to move internal data from main storage to the SYS1.LOGREC data set. The system will send you a message when the data has been moved.

RESTARTING THE SYSTEM

Follow the same steps you took in starting the system, but leave out the formatting parameter, Q=(unitaddr,F), when you issue the SET command.

Make sure that all direct access volumes containing system input data and system output data are ready.

System input and system output volumes may be switched to a different device of the same type as the one they were on when the system went down. If, however, the volumes were on a control unit having the record overflow feature, and the track overflow feature was used to record the data, then the volume that was on the control unit having the record overflow feature can only be switched to another control unit having the record overflow feature.

All volumes with SYSIN should be mounted.

All volumes with SYSOUT should be mounted.

When a system output writer encounters a data set all of whose volumes are not mounted, the data set is bypassed. Message IEF304I will go to the console and output device and will tell you the data set name and the volume serials, which you can use to dump the data set at a later time.

The job scheduler will send you the names of any jobs and job steps being handled by the reader or initiator before the failure, and the output writer will print out all the data on the system output data sets. You'll receive the message IEF420I to indicate the jobs that were handled by the reader and the message IEF421I to indicate the jobs and job steps that were handled by the initiator.

A job referred to in the message IEF420I can be reentered as is. Message IEF421I looks like this:

$$\text{INIT=job.step.procstep(x)} \left\{\begin{array}{l}\text{RESTART}\\ \text{NO RESTART}\\ \text{CANCELLED}\\ \text{ENDED}\\ \text{CONTINUING}\end{array}\right\}$$

where x is

1 If the step was being processed by the initiator and had not started executing (the message will indicate CANCELLED).

2 If the step was executing (the message may indicate RESTART or NO RESTART). In this case, you will have already received the ABEND message IEF450I, followed by this message or by a message requesting you to authorize automatic restart. Should you receive the latter message, follow the procedure described under "Restarting a Job" in the Operator's Procedures, GC28-6692.

3 If the step was being ended (the message may indicate CANCELLED, ENDED, or CONTINUING).

Whenever the message indicates NO RESTART or CANCELLED, the particular job may require programmer analysis before it can be rerun.

Controlling Input and Output

This section describes how to work with the operating system in controlling input and output, particularly in the area of allocating or assigning devices to jobs.

INPUT

A system input (SYSIN) stream is made up of job control language (JCL) statements and problem program data routed by the control program to their destinations within the system.

Sets of input data entering the system through the SYSIN stream are called SYSIN data sets. Input data can also be read directly by the problem program and not be a part of a SYSIN stream.

Data in the input stream is written on a direct access volume for later reading by the problem program.

Input Readers

An input reader is that part of the scheduler that reads a system input stream from a single device. You assign an input device to an input reader by issuing a START RDR command.

An input reader with Automatic SYSIN Batching (ASB) collects the JCL statements for multiple jobs in a batching queue, and then interprets the entire batch of statements at one time. This type of input reader normally takes less storage than other readers, and expands temporarily when JCL statements are being interpreted. You assign an input device to an input reader with ASB by issuing a START RDRA command.

Although multiple ASB input readers may be started at the same time to service multiple input streams, all concurrent ASB readers must reference the same procedure library.

Several input readers can bring jobs into the system from several SYSIN streams at the same time. In these systems, you not only issue a START RDR command to start a new input reader, but also can issue a STOP RDR command to end one.

Input readers stop automatically on encountering an end-of-file condition.

SYSIN data sets in an input stream are stored on non-demountable direct access volumes while other programs are running. Job steps can later read the data sets at high speed from the direct access volumes.

Job control information is placed in the input queue for use by the job scheduler in selecting the jobs and job steps to be processed.

Job Classes

Jobs can be grouped, depending upon your installation needs, into as many as 15 different job classes (A-O).

Each class has one-character classname. All job control information for jobs with the same classname is placed on the same input queue. Jobs with a common characteristic can be grouped together by using the same classname. The characteristic might be a job with much input/output, a job requiring a large amount of CPU processing, or possibly one which requires special control volumes.

The programmer uses the CLASS parameter in the JOB statement to define his job's input class.

Initiators

An initiator is that part of the job scheduler which selects jobs and job steps to be processed.

By using START INIT, or any other procedure name which your system programmer may supply, you specify the classnames that the initiator is to process.

For example, to process jobs from the input queue which have been grouped in class D, you would issue:

```
START INIT,,,D
```

You can also specify a force or limit priority when entering the START INIT command.

A force priority is set for an entire job class; all jobs for that class will be assigned -- or forced to have -- the same priority.

A limit priority is set for an initiator. No job step processed by the initiator can have a priority higher than the limit priority. The priority a job or step specifies will be used only if it is lower than the limit priority. If it is higher, the limit priority overrides it.

Any value that you specify with START INIT will override any class name and force and limit priorities in the cataloged procedure.

For instructions on setting class name, and force and limit priorities, see the description of the initiator under the START command.

OUTPUT

A system output (SYSOUT) stream consists of system messages and problem program output data sets routed by the control program to a common output device.

Problem program data sets that leave the system through the output stream are called SYSOUT data sets. Output data can also be written directly by a problem program and not be a part of a SYSOUT stream.

Problem programs using the SYSOUT stream write their output on a direct access volume. The output writer later picks up the output for printing. System output can also be written directly from a problem program, to a specified output device by using Direct System Output Processing see the chapter "General Operating Techniques"; on "How to Use Direct System Output Processing (DSO)."

Output Classes

Your installation, depending on its needs, can group SYSOUT data into as many as 36 different classes.

Each class has a one-character classname and includes all system messages and SYSOUT data sets the system associates with the classname. This permits the grouping of output data with a common characteristic. The characteristic might be a type of output device, a priority, or possibly a location -- for example, the data for the third floor programmers might have a classname of 3.

Examples of possible output classes are:

A High priority printed output, such as messages from the control program.

B High priority punched output, such as error records to be hand-corrected and rerun.

C Low priority printed output, such as a summary dump of disk records for auditing.

If your system has a universal character set (UCS) printer that will be used as an output writer, you must assign a separate output class to each character set image stored in the system library. To associate the character set-code with a SYSOUT class, either specify the UCS parameter in the START WTR command or add the UCS parameter to the user-designed output writer procedure. This assignment preserves the identity of character set-dependent data through the SYSOUT stream. The output is grouped by character set-code which minimizes your changing of printer chains and trains.

The programmer specifies the SYSOUT class for a set of system messages or a SYSOUT data set in the JCL statements for his job. These statements are described in the manual IBM System/360 Operating System: Job Control Language.

The system messages for a job are in the class specified by the MSGCLASS parameter of the JOB statement. A SYSOUT data set is in the class specified by the SYSOUT parameter of the DD statement for the data set.

If a job specifies a class for a SYSOUT data set that is different from the MSGCLASS, and the output data set contains data, message IEF298I jobname SYSOUT=x will appear on the MSGCLASS output immediately preceding the JOB statement.

The JCL statements in the following example describe a one-step job that executes PROGRAM in a system with MVT. The system messages for this job are in class B, while the output data set is in class C. Therefore, if there is data in the output data set, this message will appear:

IEF298I FIRSTJOB SYSOUT=C.

```
//FIRSTJOB     JOB     ,'AUSER',MSGCLASS=B
//STEP1        EXEC    PGM=PROGRAM
//RESULT       DD      UNIT=2311,SYSOUT=C,SPACE=(132,(1000,10))
```

Output Writers

An output writer is that part of the scheduler that writes a single system output (SYSOUT) stream to a single device. You assign an output device to an output writer by issuing a START WTR command.

In systems with MVT, several output writers can run at the same time on several SYSOUT streams, and more than one writer can work on an output class. In these systems, you not only issue a START WTR command to start a new output writer, but also must issue a STOP WTR command to end one.

By using START WTR, you specify the association between class names and devices, by naming (a) the device to be allocated to the output writer and (b) the class names that the output writer is to process.

You can thus ensure that each SYSOUT class is written on the device that is most suitable for that class. Given the example in the preceding section, you might issue the following command, assuming that 00E is a printer.

```
START WTR.A,00E,,C
```

This would cause the output data set to be printed. On the other hand, (assuming 185 is a magnetic tape drive) if you issued:

```
START WTR.B,185,,C
```

the data set would be written on magnetic tape.

If you wanted a printed listing from this tape, you could use one of the tape SYSOUT programs described under the heading "How to Print a SYSOUT Tape" in the Operator's Procedures, GC28-6692.

A job step writes SYSOUT data sets at high speed onto a direct access device. Later, after the job is done, an output writer will handle the data sets concurrently with other jobs.

System Log

The system log is kept by the system on a permanently mounted direct access volume. The system log is an optional feature that may be used by both problem programs and operators.

There are several kinds of information that can appear in the system log:

- Job time, step time, and data from the JOB and EXEC statements of a job that has ended. This information is entered on the log by an accounting routine written at your installation.

- Operating data entered by problem programs using a write to log (WTL) macro instruction.

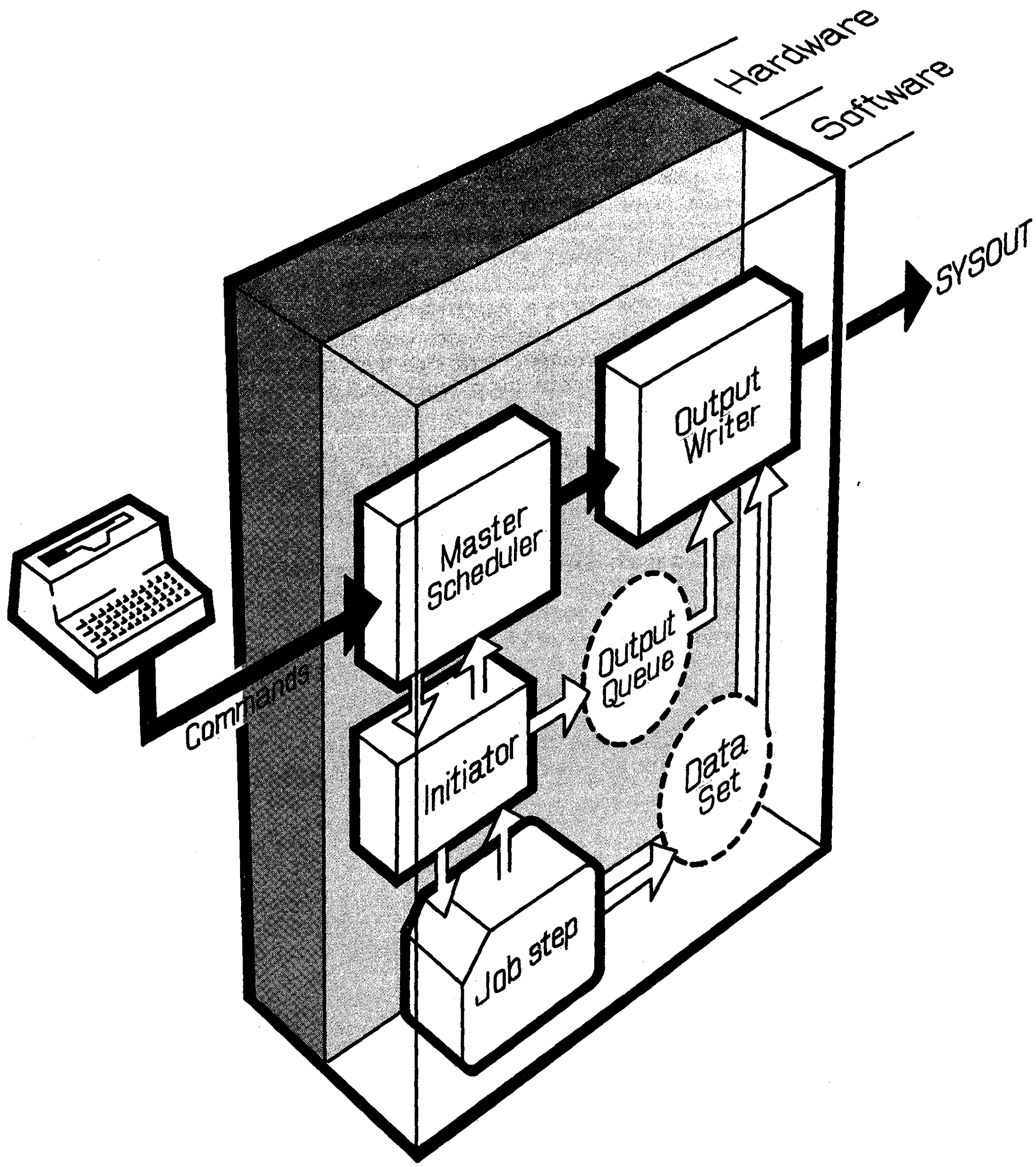
- Descriptions of unusual events that occurred during your shift. You can enter these descriptions by issuing a LOG command.

- WTO and WTOR messages, including the routing codes used to route the message and the time the message was executed (systems with the MCS option only).

- Accepted replies to WTOR messages (systems with the MCS option only).

- Commands issued through operator's consoles and the input stream, and commands issued by the operating system (systems with the MCS option only).

MVT



Output Process in Systems With MVT

Use the LOG command to make entries into the system log.

The system log consists of two data sets -- SYS1.SYSVLOGX and SYS1.SYSVLOGY. System message IEE041I LOG NOW RECORDING ON SYS1.SYSVLOG (X/Y) tells you which data set is currently being used. When the current data set is filled, the system will close it, open the other, and issue message IEE042I LOG DATA SET SYS1.SYSVLOG (X/Y) ON device CLOSED. The system then begins recording the log on the second data set and issues messages IEE041I to inform you that log recording has switched to the second data set.

Message IEE043I LOG DATA SET (X/Y) QUEUED TO SYSOUT CLASS x tells you that the filled data set has been placed on the output queue for the SYSOUT class specified in the message, though the actual scheduling cannot take place until a job currently running in the system has reached termination.

The class specified in message IEE043I is the default output class for the system log. The default value is class L unless your installation has modified it during system generation.

If you want the system log to be part of an output class other than the default class, enter a WRITELOG command with the desired classname each time you wish to modify the default class. Also use the WRITELOG command if you want the system log written out at a specific point in time.

Note: Whether the data set is scheduled for output by the system or by your WRITELOG command, make sure a writer has been started for the appropriate output class.

ALLOCATING DEVICES

Device allocation is the assignment of I/O devices for use by job steps or the system.

If a job step is using a tape drive, no other job step may use the tape drive until the first job step is finished -- the entire device is assigned.

If the job step is writing on a direct access volume, however, it may only be necessary to assign a particular set of tracks; data sets for many job steps can be on the same volume.

Device allocations are made in response to requests from three sources:

- Data definition (DD) statements. These may specify the I/O requirements of the job steps in the system.
- System generation statements. These may request fixed assignments for system processes, such as the automatic starting of an input reader.
- Operator commands. These may request assignments for system processes, or may modify assignments made when the system was generated.

Assignment by the Scheduler

Through data definition statements, the programmer describes the input/output requirements for the data sets of a job step.

Using this information, the job scheduler allocates I/O devices directly to the job step, attempting to provide overlapped operations, conserve input/output resources, and recognize items that increase input/output efficiency.

<u>Assignment by the Operator</u>

You are responsible for device assignments made for the starting of input readers and output writers. If assignments were specified at system generation time, you don't have to respecify them, unless you wish to make a modification.

All of your assignment requests are made through operator commands.

Your ability to switch input/output devices to online, offline, or console status lets you modify their allocation.

Online devices can be assigned to problem programs, while consoles and offline devices cannot be. All input/output devices can be switched online or offline, but only certain types can be consoles. Devices capable of being consoles are described in the chapter "Operator's Consoles."

Your main reason for placing a device offline is to reserve it for preventive or corrective maintenance. To switch a device's status, enter a VARY command indicating the device and the desired status.

By entering the MOUNT command, you can cause an input/output device to be assigned to those job steps that require the particular volume mounted on it.

For example, you might give a MOUNT command for a device when you know that a volume will soon be used by many independent jobs.

In systems with the automatic volume recognition option, you can mount volumes on online devices that are not ready, thus anticipating the later needs of jobs you are scheduling.

<u>Volume Mounting</u>: In most installations, your role with respect to I/O devices is to mount and demount volumes.

The job scheduler, using information from data definition statements, determines the input/output resources to be assigned to a job and the volumes that are required. If these volumes are not mounted, the job scheduler writes you a mounting message. Each message states that either a specific volume or a scratch volume is to be mounted.

Mount the requested volume and press the START key on the device to continue processing.

Occasionally you will receive two mount messages in a row for the same volume -- one message starting with IEF and the other with IEC. Treat the two messages as though they were one. The second is merely a reminder if you have not yet answered the first.

Never mount a blank tape volume, unless specifically directed to do so.

The system checks for a tape label and the absence of data causes the whole volume to be scanned for a data record. If an unlabeled tape is required, a tapemark should be written to avoid unnecessary scanning.

For a description of a program you can use to create standard labels on tapes, see the section on the program IEHINITT in the publication <u>IBM System/360 Operating System: Utilities</u>, GC28-6586.

After you mount the volume and ready the drive, the system reads the volume label. If an incorrect volume is mounted, the system repeats the mounting message and unloads the incorrect volume, if possible (some devices, such as the 2311, can only be unloaded manually).

If a scratch tape volume is incorrect only because the job specified a nonstandard or unlabeled tape and you have mounted a standard-labeled tape, the system will rewrite the tape label to conform to job specifications.

If a scratch tape is incorrect only because the job specified a labeled tape and you have mounted one without a label, you will receive message IEC704A L ddd.

If you want the system to write a new volume label for the tape, respond REPLY id,'ser,owner' to define the volume serial number and, optionally, the owner's name for the new label.

If you want to use a different tape, respond REPLY id,'M' to cause the system to unload the mounted tape and repeat the mounting message. You can then mount a different tape.

Note: You will receive a message describing any changes the system makes to the volume label; note these changes on the tape reel's external label.

If a request was made for a tape volume without a standard label and if the volume mounted does not have a standard label, that volume will be accepted. The volume is treated as unlabeled, or as a volume labeled with nonstandard labels, according to the DD statement.

The following volume mounting options can be selected at system generation time:

- Imperative mount. Mounting messages are written when the job step requiring the volumes is started.
- Automatic volume recognition. You take the initiative and mount labeled volumes on any unused drives. The system recognizes and remembers these volumes and assigns the drives to later job steps.

If your system has the automatic volume recognition feature, mount volumes you want the system to find for the first job at IPL before entering the START command.

Also before entering START, be sure that all offline devices are known to the system by using the VARY OFFLINE command.

After the first job, you can mount ahead for several jobs at a time.

In addition, the system may ask you to mount other volumes, and you can mount these on any appropriate online devices that are not ready. (Do not unload any units -- you can only mount on units unloaded by the system.)

Automatic volume recognition handles nine-track tape, seven-track tape, and 2311 and 2314 direct access devices. The density for seven-track tape is set at 200, 556, or 800 bytes per inch at the time the system is generated.

When volumes are to be demounted, the system unloads the devices, if possible, and writes you messages identifying the volumes being unloaded.

When you receive a mounting message for a 2321 data cell, hit the RESET button on the device if the requested cell is already positioned properly. If you have to open the door on the unit to position the cell, then you don't have to hit RESET -- closing the door performs the same function in this case.

Occasionally you may be asked by your installation not to mount volumes on certain devices, and not to make those devices ready, because you are running a version of the operating system that was generated for a slightly different set of devices.

Allocation Guidelines

When the scheduler cannot satisfy a request for allocation, it may be because other jobs or the system itself may be using the device needed by the job being scheduled.

When the scheduler sends you a message because it cannot allocate, it lists any offline devices as well as any devices assigned to the system, such as tapes or disks assigned to the control program's reader or writer routines and may suggest that you reply with WAIT, NOSEP or CANCEL.

When the scheduler has no other alternative but to wait for devices to become free, it puts itself in a wait state after sending you an IEF388I WAITING FOR DEVICES message. The system waits for devices to become free unless you cancel the job while the system is waiting.

When the scheduler has no other alternative but to ignore the SEP option requested by the job control language, it attempts allocation recovery after sending you an IEF389I SEP REQUEST IGNORED message.

When the scheduler has a number of alternatives, it will send you a message listing them. The alternatives will be some combination of CANCEL, unit address, WAIT, and NOSEP. Depending on the message you receive, reply with the word 'CANCEL', with a unit address, with 'WAIT', or with 'NOSEP'.

Replying with 'WAIT' tells the system to wait for devices to become free, while 'NOSEP' tells it to ignore job control language requests to provide channel and unit separation for the job step's data sets.

When canceling, be sure to use the REPLY command and to spell your reply word in upper case letters only. If instead you use the CANCEL command, the system will not terminate the job even though you may get a message saying the job is canceled -- the system will wait for you to use the REPLY command.

You can however use the CANCEL command later after using the REPLY command to say WAIT, if you change your mind about wanting the job to continue waiting.

If you ask the system to wait, you can issue STOP RDR or STOP WTR to free a device needed by the present job step. If you ask the system to wait, and later change your mind, you can still cancel the job.

If you reply with NOSEP and this still does not help the scheduler to assign a device, you will get another message and another chance to tell the system to wait, cancel, or activate an offline device.

Use the MOUNT command to reserve a volume on a device, when you know that several jobs are going to need that volume. Volumes reserved through a MOUNT command are not demounted by the system until an UNLOAD command is given, causing the system to unload the volume.

Unit Addresses: When referring to I/O devices in the unitaddr parameters of operator commands, you must use the unique unit address assigned to each device.

Group Names: Symbolic group names of one to eight alphameric characters may be defined by your installation, but these are for use by your programmers in their data definition statements.

Don't use symbolic group names in operator commands, except in the optional devicename parameter of the MOUNT and START commands. (Using devicename is better than using unitaddr -- if you specify the unit address of a device the system is using, or is about to use, your command will be rejected.)

The number of devices associated with a symbolic name may range from one to the total number of devices in your installation. This allows the devices to be grouped according to whatever attributes your installation considers significant - device type, for example, or special equipment, or installation configuration.

Work Volumes: Make sure there are sufficient work volumes available in the system to satisfy requests for temporary data sets at peak loads. Failure to do this can delay the allocation of a job step while it waits for direct access space to become free.

Where to Mount Active Volumes: At IPL, mount relatively full and active disk packs (for example, system packs) at the highest address locations available on the channel. Also, place system residence and system data sets (SYS1.MACLIB, for example) at the highest available address locations of different channels.

SYSOUT Writers with a Single Initiator: If device allocation has entered a wait for direct access space in a single initiator environment with no SYSOUT writer running, a SYSOUT writer cannot be started to free space. If this occurs, you may have to cancel the waiting job, depending on your installation's procedures. You can avoid this situation by starting a writer after IPL and having one always in the system.

Freeing Direct Access Space: A wait for direct access space will not always result in the allocation of a waiting data set. Even the eventual deletion of the temporary data sets on the candidate volumes still may not leave sufficient space on the volumes to satisfy the request. In addition if SYSOUT data sets in previous steps of the job (or previous steps of other currently active jobs) still occupy space that the data set requires, you may have to use the CANCEL command.

Shared DASD Option: If your installation is using a system with this option, be sure to read the section "How to Use the Shared DASD Option" in the chapter "General Operating Techniques."

Restarting a Job

A job may be abnormally ended as a result of a hardware, programming, or system error. Such an ending may occur at any time during program execution.

Valuable machine time would be lost if an abnormal end occurred during the processing of one of the last job steps of a multistep program or in the middle of a long job step and execution of the program had to be started again at the beginning of the first job step.

The checkpoint/restart feature of the system is provided to allow a restart of an abnormally ended job either at the beginning of a job step or at a checkpoint within this step.

Your programmer determines whether an automatic restart or a deferred restart is to be performed.

AUTOMATIC RESTART

If your programmer provides for an automatic restart and the job does abnormally end, you will receive system message IEF225D (SHOULD job RESTART) requesting a reply to authorize this restart. This allows you to prevent repeated restarts at the same checkpoint or job step.

When you are requested to authorize an automatic restart, you can reply YES, HOLD, or NO.

- Reply YES if the restart is to be performed at a specific checkpoint or job step for the first time.

  If a step restart is to occur and the step to be restarted used a card input data set that was not part of the SYSIN stream, you must return, to the appropriate hoppers, all cards read by the job step before it ended abnormally. If a checkpoint restart is to occur, follow the programmer's instructions for replacing the input cards.

- Reply HOLD if you do not wish to perform the restart at once, but want to do so soon: for example, to permit another job to be run first.

  To restart the job at a later time, you must issue an appropriate RELEASE command. Or, if desired, you may cancel the job.

- Reply NO if a restart at a specific checkpoint or job step has been requested repeatedly. When your reply is NO, and your programmer wants a restart to be performed, he must resubmit the job for a deferred restart.

The restart may be delayed by the system waiting for the allocation of main storage. If another job is using the required storage, you will get no message -- only a delay. Do a DISPLAY A to see if a system task or other job is using the required main storage. You may then STOP or CANCEL the conflicting task or job.

The system may request you to mount data volumes other than those required at the beginning of the job.

After you authorize a restart and if you are displaying jobnames, you'll get jobname STARTED and jobname ENDED messages describing system job IEFREINT (the restart reader). Ignore them. Should failure occur during execution of this job, you'll get an ABEND message.

Note: Any operator commands in the input stream of the job step being restarted will not be executed.

DEFERRED RESTART

If your programmer provides for a deferred restart, and his job does abnormally end, he must resubmit the job to have this restart performed.

To restart the job, your programmer must provide a restart deck for submission to the system through the system input reader. The JCL statements to be included in the restart deck are described in detail in the publication IBM System/360 Operating System: Job Control Language Reference.

The device configuration of your system at the time of restart need not be the same as it was when the job was abnormally ended. However, enough devices must be available to satisfy the needs of the job step being restarted.

The system under which a step restart is run need not be the same as it was for the job's original execution. However, a checkpoint restart should be run under the original system unless the alternate system can meet the following restrictions:

- The type (PCP, MFT, MVT) and release number are the same.
- SYS1.SVCLIB should be the same as it was originally.
- The link pack area modules in use at the checkpoint must reside in the same main storage locations.
- The boundary between hierarchy 0 and hierarchy 1 must be the same for a system with Main Storage Hierarchy Support for IBM 2361 Models 1 or 2.
- An area of main storage identical to the original area must be available to the job to be restarted.

You will receive message IEF209I MAIN STORAGE UNAVAILABLE and the system will cancel the restart if the required main storage is not available for any of the following reasons:

- The link pack area expands into the required main storage. This may occur if an initial program loading has been performed after the original execution of the job and prior to the restart. If it does occur, contact your system programmer for a respecification of the RAM, RSVC, and RBLDL system parameters and repeat initial program loading using the new values.

- The system queue area expands into the required main storage. When this occurs, contact your system programmer for a respecification of the SQS system parameter and repeat initial program loading using the new SQS value.

When the job restarts correctly, you will receive two messages saying so. One is IHJ006I job RESTARTING; the other is IHJ008I job RESTARTED. If you do not get both these messages, do a DISPLAY A to see if a system task or other job is using the required main storage. You may then STOP or CANCEL the conflicting system task or job.

The system may request you to mount data volumes other than those required at the beginning of the job. In addition, any card input data sets that have been used by the failing job step must again be made available to the system.

Operator Commands

This section contains a description of the commands you use to control the operating system. The formats, functions, parameters, and options of the commands are included. The commands are presented in alphabetical order.

In systems with MVT, all commands can be entered at any time. For example, the VARY command can be used before the initial SET command.

You can use abbreviations as well as the full command name when keying in your commands. The usable names and abbreviations are:

CANCEL	C	REPLY	R
DISPLAY	D	RESET	E
*HALT	Z	SET	T
HOLD	H	START	S
LOG	L	STOP	P
*+MODE		*SWAP	G
MODIFY	F	UNLOAD	U
MONITOR	MN	VARY	V
MOUNT	M	WRITELOG	W
RELEASE	A		

*This command cannot be entered into the input stream.
+This command cannot be abbreviated.

Be sure to use the correct abbreviations for operator commands. For example, use S for START and T for SET. If you inadvertently key in S for SET, the system assumes you are giving a START command, queues the command, and waits for a SET command.

The following conventions are used in illustrating the format of commands:

- Required letters (those shown in upper case) must be entered, but can be entered in either upper or lower case.
- Lower-case letters indicate that a parameter must be substituted.
- Dotted lines ... (indicating a series of terms), brackets [ ], and braces { } are not entered.
- Entries within brackets [ ] are optional.
- Entries within braces { } are required - you must select one.
- Numbers and punctuation marks (other than dotted lines, brackets, and braces) must be entered as shown.
- Stacked items represent alternative items. Only one of the stacked items is to be coded.

Except for letters between apostrophes, lower case letters are translated to upper case before being handled by the command scheduler.

Some commands require apostrophes in their operands. Be sure letters between apostrophes in these commands are upper case if they are meant to be processed as upper case.

When using the REPLY command to answer system messages, it is always correct to use upper case letters in the text between apostrophes.

Command formats are essentially free form, but one or more blanks must follow the operation field.

Commands cannot occupy more than one line. For example, if a command is entered through a card reader, it may not be more than 80 characters in length.

If comments on commands are necessary, they must appear to the right of the operand field and be separated from it by at least one blank. If the operand field is null, a comma followed by at least one blank indicates that comments will follow.

Many operator commands need a region of main storage to execute in. Most of these commands require about 8K bytes of storage. But START and MOUNT need 52K. The system may reject these commands if enough main storage space is not available.

CANCEL -- Terminate Process Immediately

Use the CANCEL command to immediately terminate:

- The scheduling or execution of a job from the system input stream or a job started by the START command (a procedure in SYS1.PROCLIB). You may be asked by your system programmers not to use the CANCEL command on certain jobs that alter data sets containing information vital to the system; canceling these jobs might make the data unusable.

  If a job is in the Automatic SYSIN Batching Queue, you cannot cancel it. Retry the command later.

- A system task in the device allocation process. A system task is in allocation if you receive a message with the prefix IEF (the exception is a mount message for tape with prefix IEF). If a system task needs operator intervention during device allocation (such as mounting a pack or receiving a mount message that you cannot satisfy), it can be canceled.

  The CANCEL command will not work for a system task after the device allocation process has completed. If a system task does not have a device associated with it, the system task cannot be canceled. To stop a running system task you must enter the STOP command.

- The writing of an output data set currently being processed by an output writer.

<u>Note</u>: If the initiator has issued any messages requiring a REPLY, you must REPLY to those messages to allow related jobs to be removed from the system -- before you cancel. Otherwise, the system will enter a wait state.

Operation	Operand
{CANCEL C}	{jobname [[,DUMP][,ALL] ,IN[=class] ,OUT[=class]] identifier}

jobname
: the name of the job (process) to be terminated. The jobname can be the name of a job from the system input stream or the procname (a cataloged procedure name in SYS1.PROCLIB) from a job started by the START command.

DUMP
: an abnormal-end-of-task storage dump is to be taken if a step of the job is being executed when the command is received. If the programmer has put in the SYSABEND data definition statement, a full dump is taken.

ALL
: all system input and/or system output for the specified job is canceled.

IN=class
: the system will search for the job on the input queue specified by the class parameter. If IN is used without the class parameter, all input queues will be searched for the job.

OUT=class
: the system will search for the job on the output queue specified by the class parameter. If OUT is used without the class parameter, all output queues will be searched for the job.

Note: If neither the IN or OUT parameter is used the system will search all the input queues and the hold queue for the job.

identifier
: the identifier, from the START command, of the system task to be terminated during device allocation.

The following can be used to cancel a system task in the device allocation process.

- The identifier used in a START command.
- The unit type (e.g., 1403 or 2311) associated with a unit address in the START command.
- The unit type associated with a cataloged procedure, in SYS1.PROCLIB, started by the START command.

To stop a system output writer from completing the writing of a system output data set, cancel the device address (e.g., 00E OR 00F) associated with the system output writer processing the data set. The device address is from the identifier or devicename parameters of the START WTR command. The system output writer will continue processing further output.

DISPLAY -- Cause Current Display

Use the DISPLAY command to cause a current display of:

- The time of day and the date.
- Job status or the activity of tasks within the system.
- The status of input/output devices attached to the system.
- Outstanding requests for operator replies.
- The status of the queues.
- The status of a specified job.
- The system console configuration.

This command provides you with job name information needed for effective use of the CANCEL command and, together with system messages, keeps you informed of which jobs are currently being executed.

If you are using a display operator console, use the DISPLAY command to display the CONTROL command, its operands, and an explanation of each operand. See Operator's Guide for Display Consoles, GC27-6949 for a description of the CONTROL command.

```
┌──────────┬──────────────────────────────────────────────────────────┐
│Operation │Operand                                                   │
├──────────┼──────────────────────────────────────────────────────────┤
│{DISPLAY} │/T                                                  \     │
│{   D   } │ A                                                        │
│          │   ┌,TP      ┐                                           │
│          │   │,GRAPHIC │                                           │
│          │  U│,TAPE    │ ┌,ONLINE ┐ [,xxx][,nnn]                   │
│          │   │,DASD    │ └,OFFLINE┘                         >     │
│          │   └,UR      ┘                                           │
│          │ R                                                        │
│          │ Q[=list]                                                 │
│          │ N[=list]                                                 │
│          │ jobname                                                  │
│          │ C,K                                                      │
│          │\CONSOLES                                           /     │
└──────────┴──────────────────────────────────────────────────────────┘
```

T

the time of day and the date are to be displayed in the following format:

```
hh.mm.ss                 yy.ddd
|  |  |                  |   |
|  |  └--Second (00-59)  |   |
|  └-----Minutes (00-59) |   └--Day of year (000-366)
└--------Hours (00-23)   └------Year (00-99)
```

A

the system is to display information about all the jobs and job steps recognized by the supervisor as tasks.

The names of the job and job step associated with each task are displayed, as well as a number telling how many subordinate tasks (those pieces of work that support the main or job step tasks) are operating within the same region of main storage, the beginning and end addresses of the region, (or all parts of the region if it's in more than one storage hierarchy) controlled by the main task, and the amount of supervisor queue space used for system control blocks related to the main task.

U

a listing of the Unit Status information about the devices indicated is to be displayed.

Unit status can be displayed about the following device types:

TP	communication equipment.
GRAPHIC	graphic devices.
TAPE	magnetic tape units.
DASD	direct access storage devices.
UR	unit record devices.

If you do not specify a device type, the unit status of all devices in the system will be displayed.

You may display only those devices that are ONLINE, or those devices that are OFFLINE. If you do not use either of these parameters, both ONLINE and OFFLINE devices will be displayed.

Unit Status information can be displayed as specified in the second and third operands starting at address xxx for nnn number of devices. If xxx is omitted the starting address is 000. If nnn is omitted the number of devices is 100.

R

the system is to display:

- The id of each message that required a reply and has not yet been replied to.
- The unit address of each device for which a mount message has been issued but has not been complied with.
- An indication if any AVR mount messages are pending.

When you use the DISPLAY R command, the system issues message IEE110I if any operator action (reply to messages or mount volumes) is required from previous messages. If you have complied with all system requests, you will receive system message IEE111I NO OUTSTANDING REQUESTS.

Q

a listing of the number of entries on each of the non-empty input, hold, and output queues is to be displayed. Also included in the display may be the Remote Job Entry (RJE) queue, the Automatic SYSIN Batching reader (RDRA) queue, and the Time Sharing Option Background Reader (BRDR) queue.

N

a listing of job names on the hold, input, and output, queues is to be displayed. Also included in the display may be the Remote Job Entry (RJE) queue, the Automatic SYSIN Batching reader (RDRA) queue, and the Time Sharing Option Background (BRDR) queue.

list

any combination of up to four of the following items:

- Specific input work queue name (job class A through O).
- SOUT (system output queues collectively).
- HOLD (system hold queue).
- BRDR (Background Reader queue for the time sharing option.)

If list includes more than one item, you must separate the specified items by commas and enclose them in parentheses. If no list value is specified, all 15 input work queues, the hold queue, and the output queue are assumed.

jobname
the name of the job for which the following is to be displayed: job name; class; job priority; type of queue the job is in -- JOB Q, HOLD Q, or SOUTQ (SYSOUT queue); and position in the queue. The maximum length of a job name is eight characters. If JOBNAMES, STATUS, T, A, R, Q, N, C, U, SPACE, DSNAME, CONSOLES, USER, or SESS is used as a job name, it must be in parentheses.

If a job is in the batching queue used by an input reader with Automatic Sysin Batching, its status will not be displayed. Retry the command later.

C,K
the system is to display CONTROL command operands and an explanation of each operand. This display is referred to as a status display. C,K is valid only for display operator consoles.

CONSOLES
the system console configuration is to be displayed. This operand is valid only when the operating system has the multiple console support (MCS) option. The display for each console includes:

- The unit address of the console (or the input and output addresses for composite consoles).
- The unit address of the alternate console (or the input and output addresses for composite consoles).
- The status of the console, where A indicates an active secondary console, A,P indicates a pending request to make the device an active console, N indicates an inactive secondary console, N,P indicates a pending request to make the console device inactive, M indicates the master console, and H indicates the hard copy device.
- The command groups that the console is authorized to enter.
- The routing codes that the console is authorized to receive.

In addition, the display includes:

- The unit address of the hard copy log, or "SYSLOG" if the system log is the hard copy log.
- The routing codes the hard copy log is authorized to receive.
- Whether or not the log is receiving operator and system commands and their responses.

HALT -- Prepare for Power-off

Use the HALT command before you turn the power off at the end of the day, or any time the computer is not to continue under the control of the operating system. This command cannot be entered in the input stream.

You must use this command to ensure that important statistics and data records in main storage are not lost permanently. Use of this command also closes the system log and discontinues the log function.

The contents of the system log may, if your installation chooses, be written by a system output writer before power is turned off. Before you enter the HALT command, use the WRITELOG command to initiate this writing.

Operation	Operand
{HALT, Z}	EOD

EOD

the end-of-day storing is to be done of internal I/O device error counts. The information is stored in the SYS1.LOGREC data set (see the topic "Hardware Debugging Aids" in the chapter "General Operating Techniques"). This parameter is required.

When the storing is done, the system sends you message IEE334I HALT EOD SUCCESSFUL. At this point, you can safely turn the power off.

HOLD -- Temporarily Suspend Job Selection

Use the HOLD command to temporarily prevent one job, or all jobs in the input work queue from being selected for processing.

If the named job has already been selected, or if it is not in one of the input work queues, you will receive a message. Jobs temporarily suspended by HOLD are subject to CANCEL and RESET commands.

The HOLD command works in two different ways, depending on whether you use the jobname operand or the Q operand.

A HOLD jobname command causes the job to be withheld from initiation until a RELEASE jobname command is given.

MVT

A HOLD Q command, on the other hand, prevents the selection of all jobs in the specified input work queues until a RELEASE Q command is given.

A RELEASE Q command will not release a job held by a HOLD jobname command, nor will a RELEASE jobname command release a job that is in a queue which is held because of a HOLD Q command.

Operation	Operand
{HOLD / H}	{jobname / Q[=list]}

jobname
: the name of the job whose selection is to be suspended. The maximum length of a job name is eight characters. Although any job name can be in parentheses, a job with the name Q must have the Q in parentheses in the command statement.

Q
: the selection of jobs from a specified input work queue is suspended. If the parameter Q is specified without the =list option, selection of jobs from all work queues is suspended. A RELEASE Q command will release the input work queue(s).

list
: any combination of up to four input work queue names (job class A through O). If no list value is specified, all 15 input work queues are assumed. If list includes more than one item, you must separate the specified items by commas and enclose them in parentheses.

LOG -- Store Information in Log

Use the LOG command to enter information into the system log. The system log consists of two data sets on a direct access device. (Since it is an optional feature of the system, you should know whether or not your system has the option before you use the LOG command. When you enter the LOG command and the system log option is not in your system, you receive a message informing you that the log is not supported.)

When your system has the multiple console support (MCS) option, the time that the LOG command was received by the system is appended to the message text entered in the system log. (Your system must have the timer option to make the time stamp on the message meaningful.)

Operation	Operand
$\left\{\begin{matrix}\text{LOG}\\ \text{L}\end{matrix}\right\}$	'text'

text
: the exact text you wish to enter into the system log. The message written in the system log does not include the enclosing apostrophes. The type of letters and special characters that can be entered (such as upper case alphabetic only, or upper and lower case alphabetics) depends upon the character set installed in your printer. For a description of the available character sets, see Operator's Procedures.

The system will send your LOG command to the primary console or the MCS master console if the system log is temporarily inactive (as when both data sets are being written out). Enter the command again when the log becomes active (as when the data sets have been completely written).

MODE -- Recovery Management Mode Switching for Models 85, 155, and 165

The MODE command is used to control recovery management activity in the CPU of Models 85 and 155. The format and uses of the command are explained in the following sections.

MODEL 85

The MODE command for Model 85 specifies the mode of recording recoverable machine check interruptions (either a message for each error processed or a message only after a certain number have been processed). You also use MODE to reactivate deleted sectors of the high speed buffer area, to re-enable the entire buffer, or to reactivate the high speed multiply feature. Use MODE with the STATUS operand to obtain a message describing the current state of recovery management facilities.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	STATUS INIT HIR,x[,dddddddd] ECC,x[,dddddddd] n[,n,...] SECT= ALL BUF HSM

STATUS
: a message describing the status of machine recovery facilities is to be displayed. See "Status Message" below for a description of the information you will find in this message.

INIT
: recovery management functions are to be set to their initial values: the buffer is enabled with all sectors active, the high speed multiply feature is activated, and the Hardware Instruction Retry (HIR) and Error Correction Code (ECC) circuitry, machine facilities that attempt recovery from machine-check interruptions, are set to recording mode with the default threshold count. Threshold count values control the number of errors allowed in a given amount of time before a change is made from recording the errors in messages to simply counting them.

HIR
: the Hardware Instruction Retry circuitry is to be set.

ECC
: the Error Correction Code circuitry is to be set.

x
: the mode to which the HIR or ECC circuitry is to be set; use R to indicate Recording Mode and C to indicate Count Mode.

  <u>Recording Mode</u> means that a message is written to the operator for each error processed by the recovery management facilities.

  <u>Count Mode</u> means that each error processed by recovery management facilities causes a counter in the CPU to be updated. No message is written until the threshold count is reached; then you receive a message and the counter is reset to zero.

dddddddd
: the threshold count in decimal. The function of the threshold count depends on the mode of recording machine check interruptions. When the threshold is reached in Recording Mode, the system automatically switches to Count Mode. When the threshold is reached in Count Mode, a message is written and a counter in the CPU is reset to zero. If you omit this parameter, the threshold count remains unchanged.

SECT
: a previously deleted and repaired sector of the high speed buffer is to be reactivated. You can specify any number of sectors, as long as the message does not exceed the message length for your console device.

n
: the individual sector(s) to be reactivated.

ALL
: all sectors are to be reactivated.

BUF
: the entire high speed buffer is to be re-enabled.

HSM
: the High Speed Multiply feature in the Model 85 CPU is to be reactivated. Use MODE HSM only after the errors in high speed multiply that caused the system to switch to slower speed multiply have been corrected.

Model 85 Status Message: The message that you receive in response to your MODE STATUS command includes the following information about the current state of recovery management facilities:

1. HIR mode, threshold count, present error count
2. ECC mode, threshold count, present error count
3. SECTOR deletions
4. BUFFER status
5. HSM status

Note: The threshold count is set at system generation and can be reset by use of the MODE command during system operation.

MODEL 155

Use the MODE command for Model 155 to obtain information on the current state of the hardware facilities that attempt recovery from soft (recoverable) machine-check interruptions. Use this command also to enable or disable machine-check interruptions and error recovery report messages, and to change threshold counts that control the number of errors allowed in a given interval of time before a change is made from a mode that records errors to a mode in which no record of errors is kept.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	{ STATUS HIR,x[,eeee][,tttt] ECC,x[,eeee][,tttt] }

STATUS
: a message describing the current status of machine recover facilities is to be displayed. See "Model 155 Status Message Format" below for a description of the information you will find in this message.

HIR
: the Hardware Instruction Retry circuitry is to be set.

ECC
: the Error Correction Code circuitry is to be set.

x
: the mode to which the HIR or ECC circuitry is to be set. Use R to indicate Recording Mode and Q to indicate Quiet Mode.

  Recording Mode, the normal mode for Model 155, provides for a soft (recoverable) machine check interruption and an error recovery report message to the operator after each machine check condition.

  Quiet Mode means that soft machine check interruptions will be disabled, and that no error recovery report messages will be written to the operator.

eeee
: a four-digit decimal value to be inserted in the error count threshold. This value is the number of errors that will be allowed in a specified time before the HIR or ECC circuitry is placed in the quiet mode. This parameter is used with the R parameter only; if you omit it, the default error count threshold value will be used.

tttt
: a four-digit decimal value to be inserted in the time threshold. This value is the time in hours allowed for a specified number of errors before the HIR or ECC circuitry is placed in the quiet mode. This parameter is used with the R parameter only; if you omit it, the default time threshold value will be used.

Note: Because of Model 155 hardware design, ECC may be in either recording mode or quiet mode when HIR is in recording mode; but when HIR is in quiet mode, ECC is placed in quiet mode automatically. Note also that the eeee and tttt parameters are positional; when the time parameter is used alone, it must be written: ,,tttt.

Mode 155 Status Message: The message that you receive in response to your MODE STATUS command contains the following information about the current state of recovery management facilities:

```
HIR,{R},aaaa/bbbb/,cccc/dddd ECC,{R},aaaa/bbbb/,cccc/dddd BUF DEL=xxxx
    {Q}                          {Q}
```

where:

R = recording mode

Q = quiet mode

aaaa = current error count

bbbb = error count threshold

cccc = elapsed time, or INVL when TOD clock is invalid

dddd = time threshold

xxxx = number of buffer pages deleted

The MODE command may be used at the operator's discretion, or at the request of an IBM Field Engineer, to check the number of errors that have occurred, or to change threshold values, either to prevent the hardware facilities HIR or ECC from going into the quiet mode, or to return to the recording mode after the quiet mode has been entered. Or, if messages indicating that ECC or HIR has successfully recovered from a machine check interruption are being written out too frequently, your programmers may advise you to use the MODE command with the Q parameter to suppress these messages. Some examples of MODE command usage are:

1. MODE HIR,R,0009,0009

   This entry sets the mode to record errors by inserting new system error count and time threshold values. The counters will be reset to zero and the system will record until the new threshold values are exceeded.

2. MODE ECC,R

   This entry switches the mode of ECC from quiet to record. IBM default values will be inserted as the new threshold values. Note that this command will not be executed if HIR is in the quiet mode at the time it is entered.

3. MODE ECC,Q

   This entry switches the mode of ECC from recording to quiet. No threshold values are required since no record of interruptions will be kept and no counters will be updated.

MODEL 165

Use the MODE command for Model 165 to determine if the system is running in recording mode or quiet mode, what the soft error (hardware recoverable machine check) and soft error threshold counts (the maximum number of soft machine-check interruptions recorded by the system) are, and if the buffer is enabled or disabled. Also use the MODE command to set the Model 165 in recording mode or quiet mode and to enable the buffer.

The MODE command cannot be entered into the input stream.

Operation	Operand
MODE	{STATUS RECORD QUIET ENABLE}

STATUS
: the current status of the recovery management facilities is to be displayed. The message indicates if the Model 165 is operating in recording mode or quiet mode. The message will also tell you how many soft errors have occurred and how many soft errors the system will accept before it switches to quiet mode. The status of the buffer is also given.

Some examples of STATUS messages might be:

IGF053I MODE STATUS - QUIET COUNT-12 THRESHOLD-12 BUFFER ENABLED

In this case the Model 165 has just switched to QUIET mode on the occurrence of the last error. If the Model 165 is now switched back to RECORD mode, the soft error counter will be reset to zero.

IGF053I MODE STATUS - RECORD COUNT-0 THRESHOLD-12 BUFFER DISABLED

Here the Model 165 is in RECORD mode; there have been no soft errors, but the buffer has been disabled. This indicates that four buffer errors have taken place. A good way to handle this situation would be to use the ENABLE parameter of the MODE command to re-enable the buffer and a short time later to again issue the STATUS parameter. If the buffer was again disabled, it indicates that there is a recurring problem with the Model 165 which keeps disabling the buffer.

RECORD
: the Model 165 is to go into recording mode. As stated before, all soft machine checks in this mode will generate machine-check interruptions. In addition, by specifying RECORD, the soft error counter will be reset to zero. The machine does not have to be in QUIET mode to specify RECORD. The RECORD parameter of the MODE command can be used just as a way of resetting the soft error counter.

QUIET
the Model 165 is to go into QUIET mode. A case in which it would be desirable to switch to QUIET mode is the case where SYS1.LOGREC is almost full. Rather than use the remaining space to log information about soft errors, the space could be saved for a possible hard error. When the data is retrieved from the SYS1.LOGREC data set, the Model 165 could again be switched to RECORD mode to record information about all errors.

ENABLE
the high-speed buffer, which had previously been disabled, is to be reactivated. The buffer is disabled when four buffer errors occur. When ENABLE is specified, the counter for buffer errors is reset to zero.

MODIFY -- Change Process Characteristics

Use the MODIFY command to change the characteristics of:

- An output writer. You can change the output classes associated with an output writer and the conditions under which the output writer pauses for servicing of its device. When an output writer pauses, it sends you a message requesting you to perform any necessary action on its device. If the pause results from a new form number specification, you are given the form number.

- An initiator. You can change the job classes associated with an initiator.

- Direct System Output Processing (DSO). You can change the job classes and the output classes associated with DSO.

- A job. You can change programmer specified values in a job started from the console (a cataloged procedure in SYS1.PROCLIB) or a job from the system input stream. If the programmer has not set the proper indicator, the job will not repsond to the MODIFY command.

MVT

Operation	Operand
{MODIFY / F}	[procname.]identifier[,CLASS=classnames] [,PAUSE={FORMS / DATASET}] [,JOBCLASS=jjj][,OUTCLASS=s][,'job parameters']

procname
: the name of a process to be modified (WTR, INIT, a job name).

identifier
: the identifier of the process as defined by the START command or the name of a job from the system input stream.

If the process to be modified is a:

- System task with a device allocated to it, use the device address as the identifier.

- System task with no device allocated to it or a job, use the process name (cataloged procedure name in SYS1.PROCLIB) as the identifier.

- Job from the input system input stream, use the job name as the identifier.

CLASS=classnames
: for an output writer, the one to eight single-character names of the output classes to be associated with the output writer; for example, CLASS=ABCD. If more than one class name is specified, the classes are treated on a priority basis, where the left-most character indicates the class to be processed first. Note: The output writer will also accept class names as a series of characters in parentheses and separated by commas; for example, CLASS=(A,B,C,D).

For an initiator, CLASS=classnames specifies the job classes to be processed, an optional force priority for each class, and an optional limit priority for the initiator. The format of the classnames field for an initiator is:

([X[(n)]],...[LIMIT=k])

X the one-character name of the job class to be processed. Up to eight class names can be given. When there is more than one class name, the classes are treated on a priority basis, where the leftmost character indicates the class to be processed first. For example,

```
CLASS=(A,B)
```

the initiator is to process only class A. When no jobs of class A are available, the initiator will then attempt to process job class B. If no jobs of either class A or B are available, the initiator will wait rather than process jobs of any other class.

Note: If X is the only value you are specifying, you do not need parentheses; for example, CLASS=AB would also be correct.

n the force priority at which all jobs from the class specified by X will be run; n must be a number from 0 to 15. For example,

```
CLASS=(A(5),B)
```

the initiator is only to process jobs of class A and class B, and will run class A jobs only with a priority of 5.

LIMIT=k the limit priority (the highest priority at which the initiator will process a job) for the initiator; k must be a number from 0 to 15. For example,

```
CLASS=(A(5),B,C,LIMIT=7)
```

the initiator is only to process jobs of class S1A, B, and C. Class B and C jobs will be run at the priority specified by the job, up to the limit of 7. If the priority specified by the job exceeds the limit, the job will be run with the limit priority of 7. All jobs for class A will be run with a priority of 5.

PAUSE=FORMS
the output writer is to pause when a change in forms on its device is necessary.

PAUSE=DATASET
the output writer is to pause before starting to process each data set.

JOBCLASS=jjj
job class(s) which may use Direct System Output Processing (DSO). From one to eight job classes can be specified (A-O). If you modify an initiator's job classes, this will not have any effect on the DSO job classes.

OUTCLASS=s
the system output classes that are to be processed by Direct System Output Processing. One system output class can be specified (A-Z,0-9). This parameter is required if the system output class was omitted from the DSO procedure. It will also override an output class specified in the DSO procedure.

'job parameters'
parameters specified by a programmer to change values in a job currently being processed.

MVT

MONITOR -- Cause Continual Display

Use the MONITOR command to cause a continual display of:

- The name of each job at the time it is initiated or terminated. If a job is terminated abnormally, you will receive a message even if you have not entered the MONITOR command.
- The names of non-temporary data sets (in mount and keep-type demount messages).
- The available space on a direct access volume (in demount messages)
- The names of data sets and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG.

Operation	Operand
{MONITOR / MN}	{JOBNAMES[,T] / DSNAME / SPACE / STATUS}

JOBNAMES[,T]
: the name of each job is to be displayed both when the job starts and when it terminates, and that unit record allocation is to be displayed when the step starts. If a job terminates abnormally, the jobname will appear in the diagnostic message.

If the T parameter is used with the JOBNAMES parameter, the system displays the time of the day in addition to the job name. In systems with MCS, the time of day is displayed at all consoles. The format of the time display is:

```
hh.mm.ss
|  |  |
|  |  └─ Second  (00-59)
|  └──── Minutes (00-59)
└─────── hours   (00-23)
```

DSNAME
: the system is to display, within the mount and K (keep) type demount messages, the name of the first non-temporary data set allocated to the volume to which the messages refer. Mount messages for data sets with a disposition of DELETE will not contain the data set name.

SPACE
: the system is to display, in demount messages, the available space on a direct access volume.

STATUS
: the data set names and volume serial numbers of data sets with dispositions of KEEP, CATLG, or UNCATLG, are to be displayed whenever they are freed.

MOUNT -- Allocate Device

Use the MOUNT command to allow allocation of an input/output device to all job steps that require a particular volume, without intervening demountings and remountings of that volume. The volume must be removable. There will be a short delay after you issue the command before the volume is mounted (the MOUNT command must be scheduled by the system).

Operation	Operand
{MOUNT | M}	{unitaddr | devicesname} {,VOL=(NL,serial) | ,VOL=(SL,serial) | ,VOL=(AL,serial)} [,USE={STORAGE | PUBLIC | PRIVATE}] [,H={0 | 1}]

unitaddr
: the address of the input/output device to be allocated. In systems with MVT, you can specify a loaded or an unloaded device. The system will request that the correct volume be loaded if it is not already mounted. When issuing this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unit address for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

devicename
: the type of device to be allocated. After a device is allocated, you receive a mounting message.

VOL=(NL,serial)
: the volume specified does not have a standard label. The serial number, up to six characters long, is used for allocation references. This parameter is not used for direct access volumes.

VOL=(SL,serial)
: the volume specified has a standard label. The serial number, up to six characters long, is used in label checking and for allocation references.

VOL=(AL,serial)
: specifies that the volume has American National Standard labels. The serial number, up to six characters long, is used in label checking and for allocation references.

USE=STORAGE or PUBLIC or PRIVATE
: a direct access volume will be used as either a storage volume or a public volume or a private volume. If this operand is not used, the system treats the volume as a private volume. A storage volume is the most freely allocated kind of volume, open to use by the largest variety of data sets, temporary or non-temporary. Slightly restricted is a public volume, which can be allocated freely for temporary data sets, but which must be specified by volume serial number to be allocated to non-temporary data sets. A private volume is the least freely allocated kind of volume -- it is allocated only if its volume serial number is specified.

H=0 or 1
: main storage is obtained in hierarchy 0 or in hierarchy 1. The H parameter is used only when Hierarchy Support has been specified at system generation time. If the H parameter is omitted (when running with Hierarchy Support), hierarchy 0 is assumed.

RELEASE -- Make Job Available for Selection

Use the RELEASE command either to resume job selection that has been suspended by a preceding HOLD command or to initiate a job restart that has been suspended by a HOLD reply. If the job to be selected or restarted is in the input queue in a canceled status or the job is not found, you'll receive a message.

To release a job that was held through a HOLD jobname command or through a TYPRUN=HOLD on the JOB statement, or to initiate a job restart that was suspended through a HOLD reply, issue a RELEASE jobname command.

To release jobs held because they were on the specified input work queue when a HOLD Q command was given, issue a RELEASE Q command.

MVT

A RELEASE Q command will not release a job held by a HOLD jobname command, nor will a RELEASE jobname command release a job that is in a queue which is held because of a HOLD Q command.

Operation	Operand
$\left\{\begin{array}{c}\text{RELEASE}\\ \text{A}\end{array}\right\}$	$\left\{\begin{array}{l}\text{jobname}\\ \text{Q[=list]}\end{array}\right\}$

jobname
: the name of the job to be made available for processing. The maximum length of a job name is eight characters. Although any job name can be in parentheses, a job with the name Q must have the Q in parentheses in the command statement.

Q
: all jobs in the input work queue are to be made available for processing.

list
: any combination of up to four input work queue names (job class A through O). If no list value is specified, all 15 input work queues are assumed. If list includes more than one item, you must separate them by commas and enclose them in parentheses.

REPLY -- Reply to Information Request

Use the REPLY command to reply to messages from the operating system and from problem programs that request information.

The REPLY command need not directly follow the message requesting the reply. The reply id ensures that the message is routed by the system to the correct job. Because other messages can be printed before the reply has been entered, be sure that, in replying to the last message printed, previous messages are not ignored.

Operation	Operand
$\begin{Bmatrix}\text{REPLY}\\ \text{R}\end{Bmatrix}$	id,'text'

id
: the 2-character reply identification field of the message requesting the reply.

text
: the text to be entered in response to a message. The information passed to the program expecting the reply does not include the enclosing apostrophes. When using the REPLY command to answer system messages, it is always correct to use upper case letters in the text.

RESET -- Change Class or Priority of Job

Use the RESET command to change the class, selection priority, or both of a job in an input, hold, or system output queue. The priority of all system output for a job can be reset, or the priority of a job in a specified system output class may be reset. A canceled job cannot be reset.

You will receive message IEE316I jobname JOB NOT FOUND if the job is not found in an input, hold, or system output queue. The job may not be in the system, may be on a queue other than the one specified in the command, or may have been selected by a writer.

Operation	Operand
{RESET E}	jobname {,PRTY=priority ,CLASS=class} [,OUT=outclass]

jobname
: the name of the job whose priority and/or class will be changed. The maximum length of a jobname is eight characters. If as a result of a previous RESET command, a system output class has more than one entry for the same jobname, a new RESET command for the same jobname will apply only to the highest priority entry.

PRTY=priority
: the value to which the job's priority is to be set. The value is a two-digit number that may range from a low of 00 to a high of 13.

CLASS=class
: the input or output class to which the job is to be moved. Class names are A-O for input (job class), A-Z and 0-9 for output (system output class).

Note: Both the PRTY and CLASS parameters can be used in the same RESET command.

OUT=outclass
: a system output class from which the job is to be moved, or on which its priority is to be changed. This field contains a one character class identifier A-Z, or 0-9.

SET -- Set Date, Time, and Location

Use the SET command to establish the date, the time of day, the device for the input work queue and whether the queue is to be formatted, the location of the procedure library, or the automatic commands you wish to override. Any combination of these may be specified.

Always use the CLOCK operand in systems with MVT. These systems use the data in this operand when they name system data sets. If you don't use the CLOCK operand, the system sets the clock value to zero.

Operation	Operand
{SET} { T }	DATE=yy.ddd[,CLOCK=hh.mm.ss][,Q=([unitaddr][,F])] [,PROC=unitaddr][,AUTO=characters]

DATE=yy.ddd
the date in the following format:

```
yy.ddd
|   |
|   └----Day of year   (001-366)
└--------Year          (00-99)
```

CLOCK=hh.mm.ss
the time of day in the following format:

```
hh.mm.ss
|  |  |
|  |  └--Seconds (00-59)
|  -------Minutes (00-59)
└--------Hours   (00-23)
```

If the new clock setting implies a change of date, the new date must be explicitly stated using the DATE parameter.

Q=(unitaddr) or (unitaddr,F) or (,F)
(1) the address of the direct access device (other than an IBM 2321) on which the volume containing the input work queue (SYS1.SYSJOBQE) resides or (2) that the system is to format the input work queue, or both. This parameter is used only in the first SET command after IPL. Note: If you are specifying only unitaddr, you do not need parentheses.

Space for the input work queue must have already been allocated on the volume which is mounted on the specified device.

You need not specify a 3-character unit address if one of the following conditions exists:

- The SYS1.SYSJOBQE data set is cataloged.
- The volume containing the SYS1.SYSJOBQE data set resides on the device that has been specified at the time of system generation.
- The SYS1.SYSJOBQE data set is contained on the system residence volume.

If the system is to format the input work queue prior to the first job initiation, you must specify ,F following or without the 3-character unit address. For example, Q=(unitaddr,F) specifies that the system is to format the input work queue on the volume residing on the direct access device referred to by unitaddr; Q=(,F) specifies that the system is to format the input work queue either on the volume to which the SYS1.SYSJOBQE data set is cataloged, or on the volume that resides on the device specified at the time of system generation, or on the system residence volume.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.SYSJOBQE data set. If this data set is not cataloged, the system uses the values provided at the time of system generation, if any such values were provided. If no such values were provided, the system assumes that the data set is contained on the system residence volume.

The system issues an error message when either of the following conditions exists:

1. The volume to which the SYS1.SYSJOBQE data set is cataloged is not mounted.
2. The system cannot locate the SYS1.SYSJOBQE data set on the selected volume.

PROC=unitaddr
: the address of the direct access device on which the volume that contains the procedure library resides.

If this parameter is not given, the system searches the system catalog (SYSCTLG) to determine and select the volume that contains the SYS1.PROCLIB data set. If this data set is not cataloged, the system uses the values provided at the time of system generation, if any such values were provided. If no such values were provided, the system assumes that the data set is contained on the system residence volume.

The system issues an error message if either of the following conditions exists:

1. The volume to which the SYS1.PROCLIB data set is cataloged is not mounted.
2. The system cannot locate the SYS1.PROCLIB data set on the selected volume.

This parameter is used only in the initial SET command issued immediately after IPL and should only specify a device that is ready.

Note: When the system determines the location of the SYS1.PROCLIB data set, the data set is recataloged. Therefore, if you specify the SYS1.PROCLIB data set to reside on a volume other than the volume it normally resides on, the data set will be cataloged to that volume. On the next IPL, the PROC parameter will have to be used to reset the data set to its normal volume.

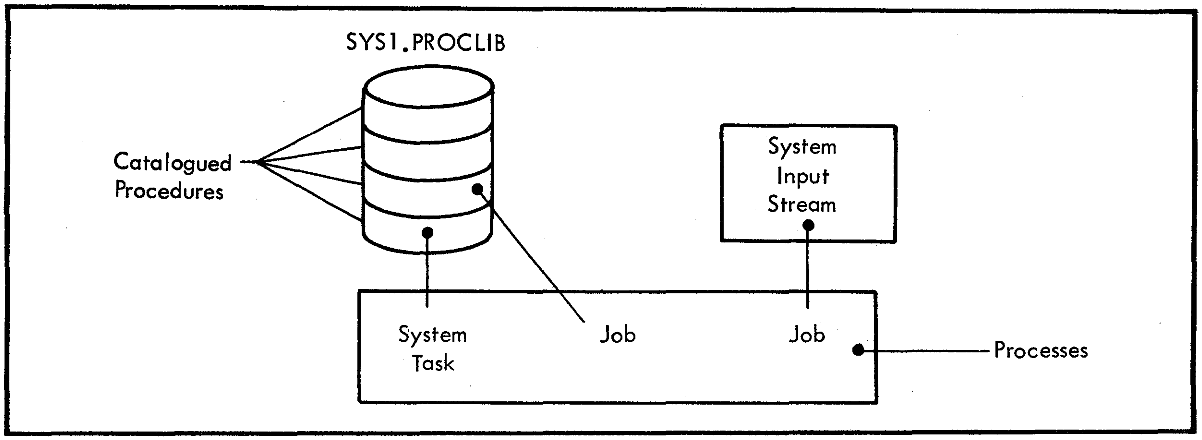
AUTO=characters
: in systems with automatic START commands, whether you wish to retain any of those commands. For each automatic command printed out by the system, follow the equals sign by a Y if you want to retain the command, or by an N if you want to override the command. For example, if the system prints out S WTR, S RDR, S INIT, and you want to retain the automatic reader and writer but not the initiator, key in AUTO=YYN. If you want to reject all automatic commands, key in AUTO=NONE or AUTO=NNN.

This parameter is used only in the initial SET command after IPL.

If this parameter is omitted, the system will use the automatic START commands.

START -- Start Process

Use the START command to start a process that is a cataloged procedure in SYS1.PROCLIB. A process can be a cataloged procedure that resides in SYS1.PROCLIB or a job from the system input stream. A catalogued procedure in the SYS1.PROCLIB can be either a system task or a job. If a cataloged procedure is a system task, the process name is the task name. If a cataloged procedure is a job, the process name is the job name.



To Stop or Cancel a Process Do One of the Following

System Task: When a system task is in allocation, stop it with the CANCEL command. Once a system task is running, stop it with the STOP command. There are two exceptions: one, an input reader will stop when it encounters an end-of-file; two, if you cancel an output writer it will stop writing the current job and start writing the next job.

Job: To stop a job use the STOP command. The stop command will work only if the programmer has set the proper indicator to cause the job to respond to the STOP command. To cancel a job, use the CANCEL command. This will immediately terminate the scheduling or execution of a job. It will also stop the writing of the jobs output.

Additional Format Conventions Used in the START Command

Positional Parameters: Positional parameters, such as devicename, volumeserial, and parmvalue in the START command, must be entered exactly in the order in which they appear. If you leave one out, put a comma in its place. You do not need to use a replacing comma if:

- The parameter to be left out is the last one in the series.
- All positional parameters following the absent one are also to be left out.
- All optional positional parameters are to be left out. Example:

```
START RDR.A,282,DSNAME=N
```

Keyword Parameters: Keyword parameters, such as DSNAME=name, can appear in any order. To leave one out, simply omit it; do not replace a missing keyword parameter with a comma.

Operation	Operand
{START / S}	procname[.identifier][,devicename][,volumeserial] [,parmvalue][,keyword=option,...] [,H={0/1}]

procname
the name of a cataloged procedure which defines the process to be started. This procedure must reside in SYS1.PROCLIB. The name can be either an IBM assigned name or a user assigned name provided by your system programmer.

MVT

IBM Assigned Names and the Processes They Describe

CRJE	Conversational Remote Job Entry
BRDR	Input reader for time-sharing option background jobs.
DSO	Direct System Output Processing
GFX	Graphic Job Processor or Satellite Graphic Job Processor
INIT	Initiator
RDR	Input Reader (SYSIN data sets unblocked)
RDR400	Input Reader (SYSIN data sets in 400-byte blocks)
RDR3200	Input Reader (SYSIN data sets in 3200-byte blocks)
RDRA	Input Reader with ASB (SYSIN data sets in 3200-byte blocks)
RJE	Remote Job Entry Subsystem
WTR	Output Writer

User Assigned Names and the Processes They Describe

A user assigned name is any name your system programmer assigns to a cataloged procedure that resides in SYS1.PROCLIB. The processes they describe are the jobs that are run as a result of starting the user named procedure. In other words, you can start a job (process) from the console by starting a user named procedure in the SYS1.PROCLIB. Use the process name from the START command to refer to a started job in other operator commands. The started job will be run as soon as the storage and the necessary devices have been allocated.

identifier
an optional name of up to eight characters, the first of which must be alphabetical. The identifier identifies a specific system task to the system. Although initiators with the same procname can have the same identifier, all other system tasks must have unique identifiers. You use this identifier to enter STOP or MODIFY commands or to cancel the system task during allocation. For this reason, it is good practice to assign an identifier on the START command. The system will assign one if you do not; however, the system-assigned identifier might not be unique.

Example: To start a writer for job classes ABC, enter:

```
START WTR.FIRST,00E,,ABC
```

If you have to cancel the writer during allocation, enter:

```
CANCEL FIRST
```

Note: With STOP or MODIFY, you can use the procname with the identifier. With CANCEL, you must use just the identifier.

The identifier to be used in the MODIFY, STOP, and CANCEL commands is derived in the following way. When an identifier is specified in the START command, it unconditionally becomes the identifier. If an identifier is not specified, the identifier is assigned either the unitaddr of the device allocated to the system process started by the START command or, when no unit is allocated, the procname specified in the START command. The following table indicates the identifier associated with each started system task.

START Command	Unit allocated	identifier
S RDR.A,00C	00C	A
S RDR,00C	00C	00C
S INIT	none	INIT
S WTR	00E (from proc)	00E

devicename
: an input or output device. This can be either a unit address (such as 280) or a device type (such as 2400). If specified, devicename will override any corresponding unit specification in the cataloged procedure.

If you don't use this parameter but specify one or both of the parameters volumeserial and parmvalue, you must indicate the absence of the device parameter by a comma. Example:

```
START RDR.A,,123456
```

If you use devicename but do not use volumeserial and parmvalue, you do not need commas to indicate their absence. Example:

```
START RDR.A,282,DSNAME=IN
```

volumeserial
: the up-to-six-character serial number of a magnetic tape or direct access volume. If specified, this parameter will override any corresponding volume serial specification in the cataloged procedure.

If you don't use this parameter but specify a parmvalue parameter, you must indicate the absence of the volumeserial parameter by a comma. Example:

```
START RDR.A,282,,JOBX
```

No comma is necessary to indicate the absence of parmvalue. Example:

```
START RDR.A,282,111111,DSNAME=IN
```

parmvalue
: the parameter values to be passed to the program receiving control as a result of the START command. If parmvalue contains any non-alphameric character, such as an equals sign, parmvalue must be enclosed in parentheses.

No comma is necessary to indicate the absence of parmvalue. If you are omitting devicename, volumeserial, and parmvalue, you do not need to use commas in their place. Example:

```
START RDR.A,DSNAME=IN
```

The particular form of parmvalue that may be used depends on the system process that is being started. The parmvalues that may be used with the system processes described by the standard procedures are:

Input Reader -- parmvalue may specify a jobname. Use this parmvalue to cause forward spacing through the input stream so that processing begins with the named job. For example, the command S RDR.A,,,JOBX will start an input reader that will space forward until reaching the first statement of JOBX, and then begin processing.

Output Writer -- parmvalue may specify class names when starting an output writer. Use this parmvalue to limit the classes the writer is to process.

The class name parmvalue may be from one to eight characters that represent the output classes to be processed. The specified classes are treated on a priority basis where the left-most character indicates the queue to be processed first. A class name parmvalue entry causes all class names specified in the cataloged procedure to be overridden.

For example, the command S WTR.A,,,AB will start an output writer that will only write output of class A. Whenever no output of class A is ready, the writer will then write output class B. If no output of either class A or B is ready, the writer will wait rather than process output of any other output class.

Note: The output writer will also accept classnames as a series of characters in parentheses and separated by commas; for example,

S WTR.ONE,00E,,(A,B,C).

Initiator -- parmvalue may specify class names, a force priority associated with each class, and a limit priority associated with the initiator. Use this parmvalue to limit the classes processed by an initiator, set their force priority, and establish the limit priority for the initiator.

The format of the parmvalue field is:

    ([X[(n)]],...[LIMIT=k])

X the one-character class name of the job class to be processed. Up to eight class names can be given. When there is more than one class name, the classes are treated on a priority basis, where the left-most character indicates the class to be processed first. For example,

    S INIT.A,,,(A,B)

will start an initiator that will process only class A. When no jobs of class A are available, the initiator will process job class B. If no jobs of either class A or B are available, the initiator will wait rather than process jobs of any other class.

Note: If X is the only value you are specifying, you do not need parentheses; for example, S INIT.A,,,AB would also be correct.

n the force priority at which all jobs from the class specified by X will be run; n must be a number from 0 to 15. For example,

S INIT.A,,,(A(5))

will start an initiator that will select for execution only jobs of class A and will run these jobs only with a priority of 5.

LIMIT=k specifies the limit priority (the highest priority at which the initiator will process a job) for the initiator; k must be a number from 0 to 15. For example,

S INIT.A,,,(A,B(5),C,LIMIT=7)

will start an initiator that will select for execution only jobs of classes A, B, and C. All jobs run from classes A and C will be run at the priority specified by the job, up to the limit priority of 7. If the priority specified by the job exceeds the limit, the job will be run with the limit priority of 7. All jobs from class B will be run with a priority of 5.

Direct System Output Processing -- parmvalue may specify job class(s) and output class when starting direct system output processing.

JOBCLASS=jjj
job class(s) that may use Direct System Output Processing (DSO). From one to eight job classes can be specified (A-O).

OUTCLASS=s
the system output class that is to be processed by Direct System Output Processing. One system output class can be specified (A-Z,0-9). This parameter is required if the system output class was omitted from the DSO procedure. It will also override a output class specified in the DSO procedure.

Time Sharing Option -- for a description of parmvalue that may be used with the Time Sharing Option, see the chapter "Time Sharing Option."

Graphic Job Processor or Satellite Graphic Job Processor -- parmvalue specifies a new value for and overrides the corresponding GFX option value selected at the time of system generation. You can override just one or more or all of the GFX option values specified at the time of system generation. For a description of the graphics parmvalue, see "How to Specify the Use of the Graphic Job Processor or Satellite Graphic Job Processor" later in this chapter.

Remote Job Entry Subsystem -- for a description of parmvalue that may be used with a Remote Job Entry Subsystem, see the chapter "Remote Job Entry and Conversational Remote Job Entry."

keyword=option
any appropriate keyword syntax allowable on a DD statement or any symbolic parameter keyword that is defined in the procedure specified in the START command. (For detailed information on these keywords, refer to IBM System/360 Operating System: Job Control Language Reference.) Symbolic parameter keywords can be provided by your system programmer.

If such DD keyword=option parameters are specified, they will override the corresponding parameters on the DD statement for the input or output device in the cataloged procedure. If the devicename positional parameter is used, the UNIT keyword may not be used. If the volumeserial positional parameter is used, the VOLUME keyword may not be used.

MVT

If the input device is a disk, you must use the keyword DSNAME=name to specify the correct data set. If the data set is not cataloged, you must use either the volumeserial parameter of the START command or the keyword VOLUME=SER=volumeid. But whether or not the data set is cataloged, you must specify DISP=OLD when using the IBM-supplied reader procedures, unless you want the data set to be deleted.

You cannot specify a keyword=option parameter in a START command for a graphics interface task.

H=0 or 1
In systems that have been generated with hierarchy support, system tasks (like a reader or writer) can be started in storage hierarchy 0 (H=0) or hierarchy 1 (H=1). If the H parameter is omitted, hierarchy 0 is assumed.

Example:
```
s rdr,00c,H=0
s wtr,00e,H=1
```

STOP -- Stop Process or Continual Display

Use the STOP command to stop the operation of:

- A Process (a system task or job started by the START command, or a job from the system input stream).
- A continual display of job names.
- A continual display of non-temporary data set names (in mount and keep-type demount messages).
- A continual display of available space on a direct access volume (in demount messages).
- A continual display of names of data sets and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG.

When you use the STOP command to stop a process, the process does not stop immediately. Instead the system begins the stopping after the process finishes handling its current task. If the process you are stopping is a job, the STOP command will work only if the programmer has set the proper indicator to cause the job to respond to the STOP command.

After a reader is started, it must process one job before a STOP RDR command will take effect. In addition, input readers stop automatically when an end-of-file condition is encountered. If a MOUNT message for tape has been received, the mount request must be satisfied before the reader will stop.

Operation	Operand
{STOP P}	{[procname.]identifier JOBNAMES DSNAME SPACE STATUS}

procname
: the name of a process to be stopped (e.g., RDR, WTR, or RJE).

identifier
: the identifier of the process as defined by the START command or the name of a job from the system input stream.

If the process to be stopped is a:

- System task with a device allocated to it, use the device address as the identifier.
- System task with no device allocated to it or a started job, use the process name (cataloged procedure name in SYS1.PROCLIB) as the identifier.
- Job from the system input stream, use the job name as the identifier.

JOBNAMES
: a continual display of the names of jobs, initiated by the JOBNAMES parameter of the MONITOR command, is to be ended.

DSNAME
the system is to stop the continual display of the names of non-temporary data sets as initiated by the DSNAME parameter of the MONITOR command.

SPACE
the system is to stop the continual displaying, in demount messages, the available space on a direct access volume. The display was initiated by the SPACE parameter of the MONITOR command.

STATUS
the system is to stop the continual display, at step end and job end, of the names and volume serial numbers of data sets with dispositions of KEEP, CATLG, and UNCATLG. The display was initiated by the STATUS parameter of the MONITOR command.

SWAP -- Swap Volumes for Dynamic Device Reconfiguration (DDR)

Use the SWAP command to request Dynamic Device Reconfiguration of two volumes or in response to a system request for Dynamic Device Reconfiguration. (DDR must be specified at system generation time.)

The command can be entered to use DDR to exchange any two demountable volumes (on 2311's, 2314's, 2321's, or 2400's), the request can come from you or the system. See "How to Use Dynamic Device Reconfiguration" in the chapter "General Operating Techniques."

Operation	Operand
{SWAP | G}	{OFF | ON | xxx,yyy}

OFF
: system/initiated Dynamic Device Reconfiguration is stopped and all permanent errors will bypass DDR processing.

ON
: activates system/initiated Dynamic Device Reconfiguration.

xxx
: the primary channel unit address of the device from which a volume is to be swapped.

yyy
: the primary channel unit address of the device to which a volume is to be swapped.

Note: The devices specified by xxx and yyy must be of the same device type. xxx and yyy may specify the same device to permit maintenance of a particular volume or device.

UNLOAD -- Prepare Volume for Demounting

You normally use the UNLOAD command to remove a volume previously mounted in response to a MOUNT command, but you can use UNLOAD to remove any tape or removable direct access volume.

The UNLOAD command causes a volume on an input/output device to be prepared for demounting. When the volume is ready to be demounted -- when all job steps using it have terminated -- you will receive message IEF282I xxx NOW UNLOADED. If SYSOUT data sets are allocated to the volume, you will not be able to unload until a system writer has processed the data sets.

MVT

Operation	Operand
$\left\{\begin{matrix}\text{UNLOAD}\\ \text{U}\end{matrix}\right\}$	unitaddr

unitaddr
: the unit address of the input/output device to be prepared for demounting. When issuing this command for a 2321 data cell, unitaddr must contain a bin number in addition to the channel, control unit, and device. For example, 263/8 is the unit address for bin number 8 of a 2321 with a device number of 3, a control unit number of 6, and a channel number of 2.

VARY -- Vary Status of Device

Use the VARY command to place input/output devices (other than a communication line) into an online or offline status. This command is also used with the graphic job processor (GJP) and satellite graphic job processor (SGJP) to designate the display units and graphic subsystems to be available or unavailable for graphic job control operations.

If Alternate Path Retry is in the system, use the VARY command to place a channel/control unit path online or offline. The last path to a device cannot be varied offline, and paths to Shared DASD or TP paths cannot be varied at all.

See "The VARY Command in Systems with MCS" in the chapter "Operator's Consoles." for instructions on how to use the command to perform functions related specifically to MCS.

Operation	Operand
{VARY V}	{ALL (unitaddr[,unitaddr]...)} {,ONLINE ,OFFLINE ,ONGFX ,OFFGFX[,{F M S}] ,PATH,xxxy,{ONLINE OFFLINE}}

ALL
: for GFX, all units named for GJP or SGJP at system generation are to be made available (ONGFX) or unavailable (OFFGFX).

unitaddr
: the unit address of the input/output devices whose status is to be changed. You don't need parentheses around only one unitaddr. For GFX, unitaddr specifies the 2250s (for GJP) or telecommunication lines (for SGJP) that are to be made available or unavailable for graphic job control operations. For GJP or SGJP, do not specify more than 15 devices.

  When both GJP and SGJP are functioning in an operating system, unit addresses of 2250s and telecommunication lines can be intermixed.

  For PATH, the unitaddr is the channel unit address of the primary path to a device. There may be as many as four paths to a device, each path having a channel unit address. When you VARY the path of a device use the lowest channel unit address associated with the device. Only one channel unit address may be specified with each VARY PATH command.

  To vary the status of an <u>entire</u> 2321 data cell, use its three-character unit address -- 263, for example. To vary the status of a particular 2321 <u>bin</u>, use its five-character unit address -- 263/8, for example, where the 8 is the number of the particular bin being addressed.

ONLINE
: the input/output devices identified in this command are to be made available for allocation by the job scheduler to problem programs. If a device is made ONLINE from an OFFLINE status, and you want the system to recognize a volume that was mounted on the device when you took it OFFLINE, use the MOUNT command to identify the volume to the system.

MVT

OFFLINE
the input/output devices identified in this command are to be removed from the recognition of the job scheduler, and that any further allocation of the devices to problem programs and system tasks is to be prevented. If the devices are in use (allocated to a problem program or to an input reader or output writer), the status is not changed until all the current users have finished with the devices. When the status is changed to offline, you will receive a message.

A device can be removed from the offline status only by a subsequent VARY command or, if an appropriate system message is received, by issuing REPLY id,'unitaddr'.

ONGFX
the 2250s and/or telecommunication lines identified in this command are to be made available for graphic operations.

Note: This command does not vary the device online. To do this, use the VARY ONLINE command.

OFFGFX
the 2250s and/or telecommunication lines identified in this command are to be made unavailable for graphic operations.

Note: This command does not vary the device offline. To do this, use the VARY OFFLINE command.

F
(used with OFFGFX) the operating system jobs being processed for the specified 2250s and/or 1130/2250 subsystems are to be ended immediately without notifying the users that this ending has occurred (F = fast stop). A printed record of previous job control operations cannot be obtained. The operator should notify the user at the 1130/2250 subsystem.

M
(used with OFFGFX) the operating system jobs being processed for the specified 2250s and/or 1130/2250 subsystems are to be ended immediately and the user requested to log off (M = medium stop). A printed record of previous control operations can be obtained.

S
(used with OFFGFX) the operating system jobs currently being defined or processed for the specified 2250s and/or 1130/2250 subsystems are to be allowed to reach normal or abnormal end before the users are requested to log off (S = slow stop). A printed record of previous job control operations can be obtained. The operand is underscored to indicate that the system assumes S if none of the parameters F, M, and S has been specified.

PATH
this is a VARY PATH request. (To be used only if Alternate Path Retry was specified at system generation time.)

xxx
the channel unit address of the path to be varied.

y
the CPU identifier (for M65MP only, this parameter must be specified).

ONLINE
the specified path is to be logically added to the system and is made available for allocation by the job scheduler to problem programs.

OFFLINE
    the specified path is to be logically removed from the system. Any further allocation of the path to problem program is prevented. A path can be removed from offline status only by a subsequent VARY PATH command or a re-IPL.

Note: A volume that has been reserved through use of either a PRESRES entry or a user-issued MOUNT command cannot be moved or removed from the system by varying offline the device on which it resides. A reserved volume can be moved or removed only if you issue an UNLOAD command, or if the system issues an action-type message (such as IEF234A), instructing that the volume be demounted.

Example 1:

```
VARY ALL,ONGFX
```

This example makes available for graphic job control operations all 2250s and 1130/2250 subsystems that were identified by the UNITS parameter of the GJOBCTL macro instruction during system generation.

Example 2:

```
VARY (1E0,024),OFFGFX,S
```

This example makes the 2250 unit with the address 1E0 and the subsystem attached to telecommunication line 024, which were made available by Example 1, no longer available to GJP or SGJP after the current job has been processed.

<u>WRITELOG -- Write Out System Log</u>

Use the WRITELOG command to have the system log written out or to close the system log.

Operation	Operand
{WRITELOG / W}	[classname / CLOSE]

MVT

classname
    specifies the name of the system output class you want to use to write out the contents of the system log. If you do not specify classname, the default value is class L, unless your installation modified the default value during system generation.

CLOSE
    specifies that the system log is to be closed and the log function is to be discontinued. The system log can be reopened only by IPLing the system.

WRITELOG schedules the writing of the system log, but you must start a system output writer (if one does not already exist) to actually write the data set to a SYSOUT device. Also, a job must terminate after the WRITELOG command is entered for the system log to be scheduled for output.

<u>Example</u>: If you want the log to be written as output class D, enter WRITELOG D. If no output writer for class D exists, enter S WTR.Pn,00E,,D to actually write out the system log.

Summary of Special MVT Operating Techniques

This section covers procedures that apply to MVT only.

After reading this section, skip to the chapter "General Operating Techniques," which describes general techniques.

HOW TO DETERMINE SYSTEM STATUS

The DISPLAY Active (d a) command is your primary means of knowing what the system is doing. Use it frequently. Be alert to identical entries occurring on successive uses of d a. These entries indicate resource contention problems, such as contention for:

- devices (check all operator messages)
- main storage (probable problem if proper volumes are mounted)
- data sets (initiators may be waiting for data sets or main storage -- but they take up no region space)

You may need to determine system status more precisely when the system enters an enabled wait state or when one or more identical d a entries keep recurring. The following sequence of commands is a guideline for this.

HOLD Q	(prevents initiation of new jobs)
DISPLAY JOBNAMES	(get continuing data on job starting and stopping)
DISPLAY Active	(get current status)
STOP RDRs and WTRs	(to assure no main storage fragmentation)
CANCEL jobs	(cancel jobs in the reverse order of their importance; should be done only as a last resort)
DISPLAY Active	(should be used repeatedly to trace changing environment)
RELEASE Q	(to resume normal processing)

This sequence can be terminated whenever the situation appears to be corrected.

Regularly check the console sheet for messages that may need replies (the DISPLAY R command will help you determine if there are unanswered messages).

When using a display operator console, regularly check the display screen for messages that may need replies or action taken. In addition, check to see if the message area is full; if it is, some messages will have to be deleted.

Reduced system activity can also be due to the following:

- Exhausted system direct access space. Readers that don't read to end-of-file may have been delayed waiting for space. Also, jobs that are waiting for allocation frequently cannot be assigned SYSOUT space. Both cases are normal, and will persist until space is freed as a result of printing and punching activity.

- Unsatisfied mount requests. Proper volumes should be mounted promptly, or the job canceled.

- When the system enters an enabled wait state and you have not held the job queue, the system is out of work if the number of IEF429I INITIATOR WAITING FOR WORK messages issued since the <u>last</u> IEF161I READER CLOSED ddd message is equal to the number of times you gave the START INIT command, provided no readers are currently active.

- Space in the system input queue has been exhausted. This can occur if the jobs being executed cannot complete because they are waiting for main storage, as indicated by DISPLAY Active output, and the readers are allowed to continue. The mechanism is: the readers continue filling the queue with jobs, the initiators cannot free queue space by starting jobs because there is not enough main storage, and the writers cannot free queue space by processing output because no jobs are progressing to completion.

- In a Shared DASD system where two or more systems are sharing the same device, an interlock has been caused by a programming error. This situation is usually characterized by all CPUs being in an enabled wait state. In this case, it may be necessary to cancel a job in only one of the systems to resolve the interlock.

  If not all CPUs are in an enabled wait state, proceed with caution; a system could simply be waiting for a device to be released.

If in response to your DISPLAY Active command you get a list with more than one master scheduler task on it, only the first is actually the master task. The others are system tasks such as altering the job queue or creating readers, writers, and initiators.

If you get a list with more than one entry for a task, that task requires more than one region for its execution.

DISPLAY CONSOLES

The DISPLAY CONSOLES command gives you the status of each operator's console in a system with the multiple console support (MCS) option. Use the DISPLAY CONSOLES command to identify each console and its alternate, whether the console is active or inactive, what command groups are valid, and what routing codes each console is receiving. The DISPLAY CONSOLES command also displays the status of the hard copy log.

HOW TO CONTROL JOBS THROUGH HOLD AND RELEASE

In addition to the control provided by the job class facility, you can use HOLD and RELEASE to control scheduling of jobs in a multijobbing environment (more than one initiator).

For example, assume that production jobs (setup) are multijobbed with one or more scientific (CLG) jobs. However, only one production job is running at a time.

All production jobs can be initially put on the hold queue either through a TYPRUN=HOLD parameter on a JOB statement or through a HOLD command after a reader has processed the jobs but before any initiators are started.

The jobs can then be released by name through a RELEASE command one at a time -- when the first production job is complete the next can be released.

DISPLAY JOBNAMES may be helpful in telling when jobs end, but since production jobs usually end in a predictable manner -- tapes rewind, keep messages are received, and so on -- it may not be needed.

The HOLD command remains in effect across system restarts -- that is, when the job queue is not reformatted. Jobs that were held individually or as part of the queue are still held after the system is restarted, and they must be freed by the appropriate form of the RELEASE command.

Be careful of the sequence of your specific commands when controlling job class execution through HOLD Q, RELEASE Q, and S INIT. For example, the command sequence: HOLD Q; S INIT.A,,,GFEDCBA; RELEASE Q may not result in a G class job being executed first. Instead, the initiator may pick up one A or B class job prior to beginning execution of the job classes in the specified order G through A. This is because job classes G, F, etc., have not yet been released by the RELEASE Q command when the initiator looks for work in response to the S INIT.A,,,GFEDCBA command.

HOW TO EXTRACT A JOB FROM TAPE INPUT STREAM

To extract one particular job from a tape input stream, enter:

```
S RDR.A,,,jobname
```

and then:

```
P A
```

before readying the tape. Jobs preceding "jobname" will be skipped; "jobname" will be entered into the queues.

Note: A is the identifier for the reader in this particular example. Substitute your own identifier and add, if needed, the devicename and/or volumeserial parameters to the START command. The identifier is all that is needed to stop the reader.

HOW TO RUN JOBS THAT UPDATE SYSTEM DATA SETS

Do not run other jobs concurrently with jobs that update system data sets (SYS1.LINKLIB, SYS1.SVCLIB, and SYS1.PROCLIB); for example, don't run jobs that may try to use a cataloged procedure with a job that is updating that procedure. Run jobs that update system data sets as follows:

- Make sure no other jobs are active in the system; use the HOLD command to prevent the initiation of jobs on the job queue.
- Stop all readers, writers, and initiators.
- Place the jobs that are to update the system data sets in an input device.
- Start a reader to that device.
- Start one initiator when the reader stops after processing all the input.
- Start normal processing -- that is, start required readers, writers, and initiators -- when the first initiator stops after completing all the jobs. Use the RELEASE command to free the jobs placed on the hold queue earlier.

HOW TO SPECIFY USE OF THE GRAPHIC JOB PROCESSOR OR SATELLITE GRAPHIC JOB PROCESSOR

The graphic job processor (GJP) is a program that displays and requests job control information at a local IBM 2250 Display Unit. The satellite graphic job processor (SGJP) is a program that displays and requests job control information at a remote 1130/2250 subsystem (attached to a System/360 via a telecommunication line).

The user responds to the displays by entering requested information, by selecting appropriate options, or both. GJP and SGJP convert the entered information into job control statements and pass them to the operating system to initiate the job.

System Considerations When Using GJP or SGJP

To ensure uniform operation, the priority for GJP or SGJP should be high and equivalent to a time-slice priority. Furthermore, the priority and SYSOUT classes assigned to GJP or SGJP should not be assigned to any other job. After GFX has stopped, the regions, 2250s, and telecommunication lines allocated for GJP and SGJP are freed.

Starting GJP or SGJP

To use GJP or SGJP, first start the graphics interface task (GFX).

$\begin{Bmatrix}\text{START}\\ \text{S}\end{Bmatrix}$ GFX[,,,(gfx=option,...,gfx=option)]

The gfx=option parameters are described under "GFX Options."

After issuing the START GFX command, enter a VARY ONGFX command to designate the 2250s and/or telecommunication lines to be made available.

$\begin{Bmatrix}\text{VARY}\\ \text{V}\end{Bmatrix}$(unitaddr[,unitaddr]...),ONGFX

Keep in mind that a VARY ONGFX for SGJP merely activates the telecommunication line between the System/360 and the 1130/2250 subsystem. Then there's a waiting period until a user logs on -- thus letting the SGJP routines sit in main storage with nothing to do. To avoid wasting storage, don't issue VARY ONGFX until a user is ready to begin SGJP operations at the subsystem.

Stopping GJP or SGJP

At any time, enter a VARY OFFGFX command to designate the 2250s and/or telecommunication lines that are not to be used any longer.

$\begin{Bmatrix}\text{VARY}\\ \text{V}\end{Bmatrix}$(unitaddr[,unitaddr]...),OFFGFX $\left[,\begin{Bmatrix}\text{F}\\ \text{M}\\ \text{S}\end{Bmatrix}\right]$

If you include the S (slow stop) parameter in your command (explicitly or by default), you let all jobs being defined or processed reach normal or abnormal completion before the unit becomes unavailable.

If you include the M (medium stop) or F (fast stop) parameter, you force immediate termination of GJP or SGJP activity for the specified unit. (See the VARY command for the effects of the M and F parameters.) If you've already issued a command with an S parameter and feel that an M or F stop is necessary, issue another VARY OFFGFX command with an M or F.

Note: Under some conditions, such as during certain displays at the 1130/2250 subsystem, a fast or medium stop (VARY OFFGFX,F or M), will not take place immediately.

After a fast stop (VARY OFFGFX,F), do not issue VARY ONGFX until SGJP has been restarted in the 1130/2250 subsystem.

To stop the graphics interface task (GFX), use a STOP GFX command. The task will end when all units have logged off. If GJP or SGJP activity continues and you want the STOP command to take effect quickly, issue the appropriate VARY OFFGFX commands (even though you've already issued the STOP GFX command).

GFX Options

These are special parmvalue parameters of the format gfx=option. Use them in any sequence within the parmvalue field of the START GFX command to override options selected during system generation. Always enclose GFX options in parentheses. Use any of the following gfx=option parameters:

PRT=printer output class
PCH=punch output class
MSGF=foreground message class
MSGB=background message class
PRIF=foreground priority
PRIB=background priority
CLSF=foreground job class
CLSB=background job class
RGNG=gjp region size
RGNF=foreground region size
RGNB=background region size

PRT=printer output class
: specifies the class name that identifies the SYSOUT class for printed output from jobs defined at a 2250 by means of GJP or SGJP.

PCH=punch output class
: specifies the class name that identifies the SYSOUT class for punched output from jobs defined at a 2250 by means of GJP or SGJP.

MSGF=foreground message class
: specifies the class name that identifies the SYSOUT class for messages pertaining to foreground jobs defined at a 2250 by means of GJP or SGJP. This class name should not be assigned to other jobs while GFX is active.

MSGB=background message class
: specifies the class name that identifies the SYSOUT class for messages pertaining to background jobs defined at a 2250 by means of GJP or SGJP.

PRIF=foreground priority
: specifies the scheduling priority (0-13) for foreground jobs defined at a 2250 by means of GJP or SGJP. The priority within an input work queue, along with the job class, determines the dispatching priority for these jobs.

PRIB=background priority
: specifies the scheduling priority (0-13) for background jobs defined at a 2250 by means of GJP or SGJP. The priority within an input work queue, along with the job class, determines the dispatching priority for these jobs.

CLSF=foreground job class
specifies the job class (A-O) for GJP or SGJP and for foreground jobs defined at a 2250 by means of GJP or SGJP. The job class determines the input work queue in which the jobs will be placed; this determines, along with priority, how quickly the jobs will be initiated.

CLSB=background job class
specifies the job class (A-O) for background jobs defined at a 2250 by means of GJP or SGJP. The job class determines the input work queue in which background jobs will be placed; this determines, along with priority, how quickly the jobs will be initiated.

RGNG=gjp region size
an integer that specifies the number of 1024-byte blocks of main storage to be included in a region (which must be at least 60K bytes) occupied by GJP or SGJP.

RGNF=foreground region size
an integer that specifies the number of 1024-byte blocks of main storage to be included in regions in which foreground jobs, defined at a 2250 by means of GJP or SGJP, are executed.

RGNB=background region size
an integer that specifies the number of 1024-byte blocks of main storage to be included in regions in which background jobs, defined by means of GJP or SGJP, are executed.

Index

MVT

TIME SHARING OPTION

Contents

TSO

Time Sharing Option

The time sharing option (TSO) makes the facilities of the operating system available to programmers at remote terminals. They can develop, test, and execute programs without turnaround delays that occur when submitting jobs to a computer center. Remote terminal users can control certain aspects of system operation if they are authorized.

Terminal communications are handled by the Telecommunications Access Method (TCAM). See the chapter "Telecommunications Access Method."

With TSO a large number of jobs can share the resources of the system concurrently, and the execution of each job is controlled primarily by a remote terminal user. Thus, time sharing can be defined as the shared, conversational, and concurrent use of a computing system by a number of users at remote terminals.

Time-sharing processing differs from batch processing in three ways:

1. A terminal user concurrently shares the resources of a computing system with other terminal users.

2. A terminal user can enter his problem statements and other input into the system as he develops them and receive immediate results. Thus the problem of turnaround time is greatly reduced.

3. A terminal user is constantly aware of the progress of his job. He requests results from the system one step at a time, receives notification of the status of his work, and is notified of errors as soon as the system detects them. The terminal user can change his problem statements or correct errors immediately after entering each statement or at any time during the current terminal session.

Once your installation has the time sharing option in your system, you can start, modify, and stop time-sharing operations. The system resources may be shared by the time-sharing jobs entered from the terminals and by batch jobs that are being processed at the same time. You can also dedicate the system to time sharing or batch processing when you need to.

Starting TSO

To start TSO, use the START command with a user-assigned cataloged procedure name. A user-assigned name is assigned to the cataloged procedure by your system programmer.

When starting TSO, you can specify START command parameters requested by your system programmer. With START command parameters, you can:

- Request a listing of the time-sharing parameters.
- Format one or more SWAP data sets.
- Specify the number of time-sharing users.
- Specify the number of logical tracks on the job queue for jobs entered at a terminal but run with the batch processing jobs.
- Change a time-sharing region size.
- Specify the percentage of CPU time that is given to batch processing.
- Specify System Management Facilities (SMF) options.

```
{START}  procname[,,[,membername][,LIST][,FORM[=ALL]][,USERS=number]
{  S  }
         [,SUBMIT=queuesize][,REGSIZE(regno)=(nnnnnK,xxxxxK),...]

         [,DRIVER=(parameters)][,SMF=(parameters)]]
```

Note: The two commas after procname are required when entering one or more parameters. They indicate the absence of the positional parameters devicename and volumserial not used when starting the time sharing option. If you use a parameter that contains any non-alphameric character, such as an equals sign, enclose all of the START command parameters in parentheses. For example:

```
S TS,,,(membername,USERS=number)
```

procname
: the name of the cataloged procedure for the time-sharing option.

membername
: a member of SYS1.PARMLIB which contains values to override values specified at system generation time. If member is omitted, the member name used is IKJPRM00.

LIST
: requests a listing of the time-sharing parameters. Any overrides that have been issued will be in the list, including the member in SYS1.PARMLIB if it was specified. You can change the LIST information after it is displayed with the MODIFY command.

FORM[=ALL]
: one or more swap data sets are to be formatted. Swap data sets are used to swap foreground jobs between main storage and auxiliary storage. The parameter ALL will cause all of the swap data sets to be formatted. If ALL is omitted, the system will format all unformatted swap data sets. Then the system will send you the message:

IKJ034D FORMAT ddname SWAP DATA SET, REPLY 'YES' or 'NO'

asking you if a formatted swap data set is to be reformatted. You will receive the format request message for each of the formatted swap data sets in your system. A serious I/O error is a good reason to reformat a swap data set.

USERS=number
: the number of terminal users permitted. The maximum number of users depends on your installation.

SUBMIT=queuesize
: the maximum number of logical tracks that can be used at one time on the job queue for the Time Sharing Option Background Reader Queue (BRDR queue). The BRDR queue is used to enqueue jobs from the foreground to the background. If the number specified is greater than the number of logical tracks available, then the number of logical tracks available is used. The BRDR queue contains jobs entered from a time-sharing terminal which will be run with the batch processing jobs.

REGSIZE(regno)=(nnnnnK,xxxxxK),...
: the time-sharing region number and the size of each region that is to be changed. The parameter regno is the region number and nnnnnK is the size of the region. If the region size (nnnnn) is set to zero the region will be freed up and any time-sharing terminal users in that region will be logged off the system. The size of the Local System Queue Area (LSQA) is specified by xxxxxK. The LSQA area is used for time-sharing system control blocks. The LSQA area cannot be larger than the region size. nnnnn and xxxxx are the number of 1024 byte areas wanted; these numbers range from one to five digits and cannot exceed 16384. The numbers should be specified as even numbers. If an odd number is specified the system will treat it as the next higher even number.

TSO

DRIVER=(parameters)
: the parameter list for the time-sharing driver. The time-sharing driver can be either IBM supplied or your own driver. If your system uses the IBM supplied driver you can use the keyword parameter BACKGROUND=nn. The BACKGROUND=nn keyword parameter indicates the percentage of CPU (execution) time which is given to batch processing.

SMF=(parameters)
: the TSO System Management Facilities (SMF) options to be changed. You system programmer should specify these for you; they are explained in <u>System Management Facilities</u>, GC28-6712.

EXAMPLES OF STARTING THE TIME SHARING OPTION

- Start the time sharing option with the cataloged procedure name of TS and no parameters.

```
S TS
```

- Start the time sharing option with the cataloged procedure name of TS and the parameters ,LIST,USERS=20 given to you by your system programmer.

```
S TS,,,(LIST,USERS=20)
```

  The LIST parameter will cause a listing of the time-sharing parameters, and the USERS=20 parameter specifies that only 20 terminal users may connect to the system at once.

- Start the time sharing option with the cataloged procedure name TOP and the parameter ,DRIVER=(BACKGROUND=45) given to you by your system programmer.

```
S TOP,,,(DRIVER=(BACKGROUND=45))
```

  The DRIVER=(BACKGROUND=45) parameter indicates that 45 percent of the CPU (execution) time is given to batch processing. If time-sharing does not use all of the remaining 55 percent of the CPU time, it will be given to batch processing.

Modifying TSO

To modify TSO, use the MODIFY command with the cataloged procedure name specified in the START command. Modifications to the time sharing option should be made at the direction of your system programmer. The following is a list of values that can be changed with the MODIFY command.

- The number of time-sharing users allowed at one time.
- The number of logical tracks on the job queue for jobs entered at a terminal but run with the batch processing jobs.
- The size of a time-sharing region.
- The percentage of CPU time that is given to batch processing.
- System Management Facilities (SMF) options.

You can also put a time-sharing region on hold with the MODIFY command. When a time sharing region is on hold the system cannot assign other time-sharing terminal users to that region.

{MODIFY / F}	procname[,USERS=number][,SUBMIT=queuesize] [,DRIVER=(parameters)][,REGSIZE(regno)=(nnnnnK,xxxxxK),...] [,SMF=(parameters)][,HOLD=regno]

procname
    the name of the cataloged procedure that was used in the START command to start the time sharing option.

USERS=number
    the number of terminal users which may connect to time sharing.

SUBMIT=queuesize
    the maximum number of logical tracks that can be used at one time on the job queue for the Time Sharing Option Background Reader Queue (BRDR queue). If the number specified is greater than the number of logical tracks available the number of available tracks is used. The jobs on the BRDR queue are entered only from a time-sharing terminal but run with the batch processing jobs.

DRIVER=(parameters)
    the parameter list for the time-sharing driver. The time-sharing driver can be either IBM supplied or your own dirver. If your system uses the IBM supplied driver you can use the keyword parameter BACKGROUND=nn. The BACKGROUND=nn keyword parameter indicates the percentage of CPU time (execution time) which is given to batch processing.

REGSIZE(regno)=(nnnnnK,xxxxxK),...
the time-sharing region number and new region size. The parameter regno is the region number, and nnnnnK is the size of the region. If the region size (nnnnn) is set to zero the region will be freed up and any time-sharing terminal users in that region will be logged off the system. The Local System Queue Area (LSQA) is specified by xxxxxK. The LSQA area is used for time-sharing system control blocks. The LSQA area cannot be larger than the region size. "nnnnn" and "xxxxx" are the number of 1024 byte areas wanted; these numbers range from one to five digits and cannot exceed 16384. The numbers should be specified as even numbers. If an odd number is specified the system will treat it as the next higher even number.

SMF=(parameters)
the TSO System Management Facilities (SMF) options to be changed. (See System Management Facilities, GC28-6712.)

HOLD=regno
a time-sharing region is to be held and no new time-sharing terminal users are to be assigned to that region.

EXAMPLES OF MODIFYING THE TIME SHARING OPTION

- Modify the time sharing option with the cataloged procedure name TS and the parameter ,SUBMIT=20 given to you by your system programmer.

  ```
  F TS,SUBMIT=20
  ```

  The SUBMIT=20 parameter specifies that the maximum number of logical tracks that can be used at one time on the job queue for the BRDR queue is 20.

- Modify the time sharing option with the cataloged procedure name TOP and the parameters

  ```
  ,REGSIZE(1)=(100K,6K),REGSIZE(2)=(90K,6K)
  ```

  given to you by your system programmer.

  ```
  F TOP,REGSIZE(1)=(100K,6K),REGSIZE(2)=(90K,6K)
  ```

  The REGSIZE parameters define region 1 to be 100K and region 2 to be 90K. The LSQA for region 1 is 6K and the LSQA for region 2 is 6K.

Example of Using the HOLD Parameter

Your system programmer has instructed you to shut down time-sharing region 2 (set the size of region 2 to 0) in 5 minutes. The time-sharing cataloged procedure name used in the START command was TS. To shut down region 2 do the following:

1. Put region 2 on hold.

   F TS,HOLD=2

2. Enter the DISPLAY USER command to find out which terminal users, if any, are in region 2. (See the section "Displaying Terminal User Ids".)

   DISPLAY USER

   Example reply:

   IEE327I USERS=4 FLICK(1) JOE(2) SAM(2) GAD(3)

   The message is telling you that there are 4 active time-sharing terminal users on the system and their userids are FLICK in region 1, JOE and SAM in region 2, and GAD in region 3.

3. If there are terminal users in region 2 send them a message to log off the system in 5 minutes. (See the section "Communicating with Specified Terminal Users".)

   SEND 'LOG OFF THE SYSTEM IN 5 MINUTES',USER=(JOE,SAM)

4. Set the size of region 2 to 0.

   F TS,REGSIZE(2)=(0K)

Controlling TSO

To control the time sharing option (TSO) you should be aware of time-sharing terminal users logging on and off the system. Also, it will be necessary for you to send messages to specific users or all terminal users about system operation. Information on the number of terminal users active on the system at one time, the userid of the active users, and the region they are running in may be requested by your system programmer. The information can be used to modify the characteristics of the time sharing option.

MONITORING TSO TERMINAL USERS

To keep track of terminal users logging on and off the system, use the MONITOR command with the SESS parameter.

{MONITOR / MN}	SESS[,T]

SESS[,T]
: the userid for each time-sharing terminal is to be displayed both when the session is initiated and when it is terminated. If the terminal session terminates abnormally, the userid will appear in the diagnostic message.

  If the T parameter is used, the system displays the time of day in addition to the userid. The format of the time display is:

```
hh.mm.ss
|  |  |
|  |  └--Second  (00-59)
|  └-----Minutes (00-59)
└--------Hours   (00-23)
```

Response to the MONITOR SESS Command

When a time-sharing terminal user logs on the system you will receive the message:

IEF125I userid LOGGED ON TIME=hh.mm.ss

When a time-sharing terminal user logs off the system you will receive the message:

IEF125I userid LOGGED OFF TIME=hh.mm.ss

You will receive the log off message only if the terminal session was terminated normally. If there was an abnormal termination, you will receive a diagnostic message.

COMMUNICATING WITH TSO USERS

To communicate with time-sharing terminal users use the SEND command. You can communicate with specified terminal users or all terminal users:

- Currently logged on the system.
- Currently logged on the system but the message is held in the SYS1.BRODCAST data set until a terminal user requests messages. This is to prevent the terminal listing from being interrupted with messages.
- When they log on the system. The message is saved in the SYS1.BRODCAST data set and a user will receive the message when he logs on the system.

You can also:

- List the notice section of the SYS1.BRODCAST data set or a specific message from the notice section. The notice section is where messages to all users, whether they are logged on the system or not are stored.
- Delete messages from the notice section of the SYS1.BRODCAST data set.

Communicating With Specified Terminal Users

To send a message to terminal users currently logged on the system, enter the SEND command with the message and the terminal user id. If you want to send a message to a terminal user currently logged on the system, but do not want to interrupt the users terminal listing, use the LOGON parameter. The LOGON parameter will cause the message to be stored in the mail section of the SYS1.BRODCAST data set. When the user is through with is current task he can request messages that have been stored. Also, if the specified user is not logged on the system when the message with the LOGON parameter is sent, the user will receive the message when he logs on the system.

Note: never send a message directly to a terminal user currently logged on the system (the LOGON parameter is not used) unless it is absolutely necessary.

TSO

SEND	'message',USER=(userid,...) [,LOGON]

message
: the message that is to be sent to specified time-sharing terminal users.

USER=(userid,...)
: one or more terminal users, specified by their userid are to receive the message.

LOGON
: the message is stored in the mail section of the SYS1.BRODCAST data set. If the specified user is currently logged on the system he will receive the message only when he requests that he receive messages. If the specified user is not logged on the system he will receive the message when he logs on the system.

Example: To send a message to two specific terminal users whether they are logged on the system or not, enter:

```
SEND 'message',USER=(userid,userid),LOGON
```

If the specified users are currently logged on the system, they will receive the message only if they request messages. If a user is not logged on the system he will receive the message when he logs on the system.

Communicating With All Terminal Users

To send a message to all terminal users currently logged on the system, enter the SEND command with the message. If you want to send a message to all terminal users currently logged on the system, but do not want to interrupt the terminal users listing, use the LOGON parameter. The LOGON parameter will cause the message to be stored in the notice section of the SYS1.BRODCAST data set. When users are through with their current task they can request messages that have been stored. Also if users logging on the system are to receive the message, use the LOGON parameter. When a user logs on the system he will receive the message.

Note: Never send a message directly to all terminal users currently logged on the system (the LOGON parameter is not used) unless it is absolutely necessary.

SEND	'message'[,LOGON]

message
: the message that is sent to all time-sharing terminal users.

LOGON
: the message is stored in the notice section of the SYS1.BRODCAST data set. If a user is currently logged on the system he will receive the message only when he requests that he receive messages. When a user logs on the system he will also receive the message.

Example: To send a message to all terminal users currently logged on the system and all terminal users logging on the system, enter:

```
SEND 'message',LOGON
```

The message is stored in the notice section of the SYS1.BRODCAST data set. All users currently logged on the system will receive the message only if they request messages. All terminal users logging on the system will receive the message.

The system will then return the message:

```
IKJ578I - BROADCAST MSGNO=nnnn
```

The number on the system message (nnnn) is the number the system has assigned to the message that has been added to the notice section of the SYS1.BRODCAST data set.

Example: To send a message to all terminal users currently logged on the system, enter:

    SEND 'message'

Use this command only when absolutely necessary, for example, if you must shut down the system in 4 minutes. Thsi istuation would make it necessary for you to interrupt all terminal sessions to warn all users.

Listing the Notice Section of the SYS1.BRODCAST Data Set

When you send a message to all terminal users with the LOGON parameter in the SEND command, the message is put in the notice section of the SYS1.BRODCAST data set. To keep track of the accumulation of messages in the notice section, list the notice section with the SEND command and the LIST parameter.

SEND	LIST

LIST
: all messages in the notice section of the SYS1.BRODCAST data set are listed cn your console.

Example: To list the notice section of the SYS1.BRODCAST data set, enter:

    SEND LIST

To list just one message from the notice section, enter the SEND command with the number of the message to be listed and the LIST parameter. Each message in the notice section has a unique number assigned to it by the system.

TSO

SEND	msgno,LIST

msgno
: the number of the message in the notice section of the SYS1.BRODCAST data set to be listed.

LIST
: the message requested in the msgno parameter is to be listed.

Example: To list number 21, enter:

    SEND 21,LIST

Deleting a Message From the Notice Section

If you find after listing the notice section of the SYS1.BRODCAST data set that a message is no longer needed, use the SEND command and the msgno parameter to delete the message from the notice section. Each message in the notice section has a unique number assigned to it by the system.

SEND	msgno

msgno
: the number of the message in the notice section of the SYS1.BRODCAST data set to be deleted.

Example: To delete message number 23 from the notice section of the SYS1.BRODCAST data set, enter:

```
SEND 23
```

DISPLAYING TERMINAL USER IDS

To find out the number of time-sharing terminal users on the system, the userid of the terminal users on the system, and the regions they are running in, use the DISPLAY command with the USER parameter.

{DISPLAY / D}	USER[=NMBR]

USER[=NMBR]
the system is to display the number of active time-sharing terminal users on the system, the identification of each terminal user, and the region number the user is running in. When the parameter NMBR is used, only the number of active time-sharing terminal users is displayed.

System Response to the DISPLAY USER Command

After you enter the DISPLAY USER command you will receive the message:

```
IEE326I USERS=nnnn id(Rn) id(Rn)
```

Where nnnn is the number of active terminal users on the system, id is the identification of each active terminal user, and Rn is the region number of each active user.

When the parameter USER=NMBR is specified, only the number of active terminal users id displayed.

The information obtained from this command is used to assist your system programmer to modify the time sharing option.

CANCELING A TSO TERMINAL SESSION

TSO

To cancel a time-sharing terminal session use the CANCEL command and the U=userid parameter. It is necessary for you to know the userid. If you are not sure the user is active enter the DISPLAY USER command to display the userid's of the active TSO terminal users.

{CANCEL / C}	U=userid[,DUMP]

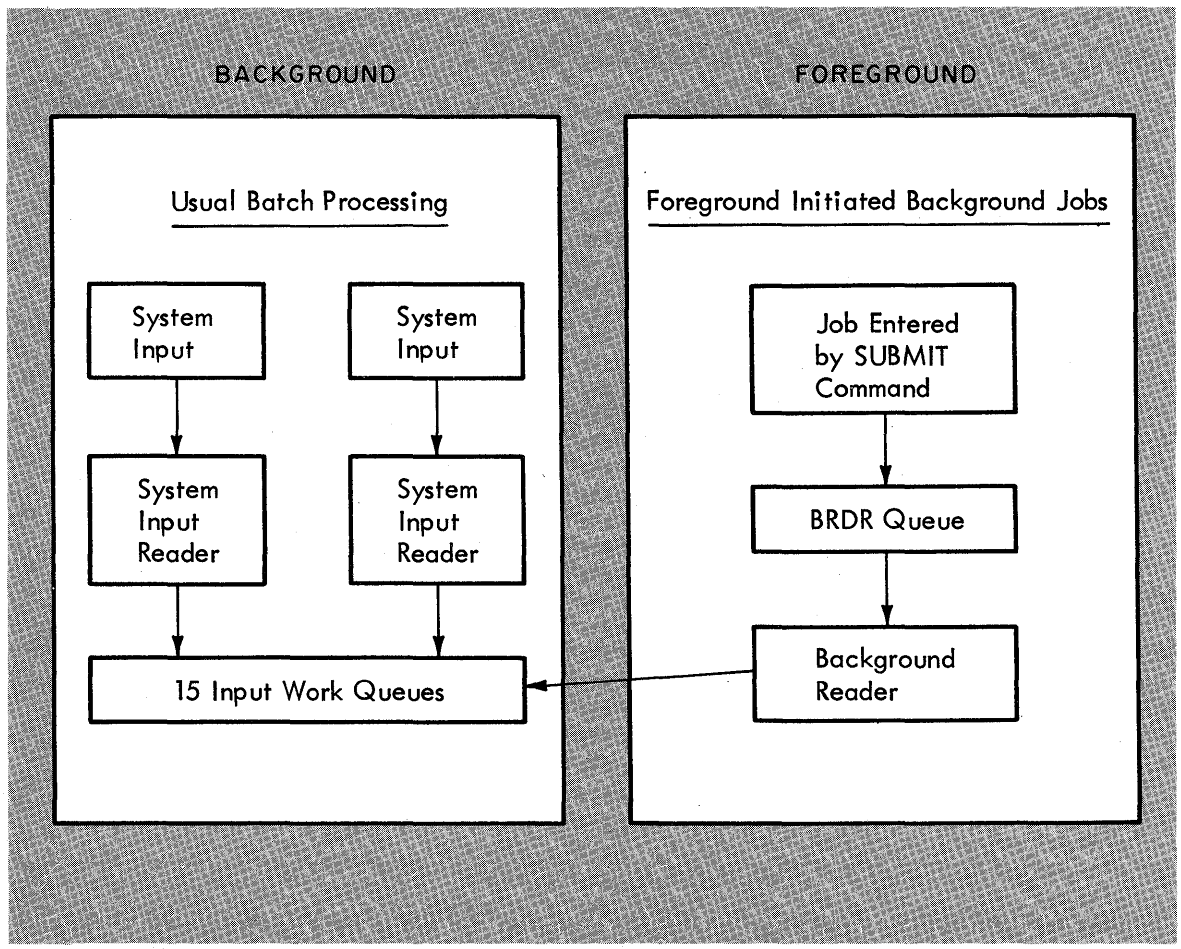
U=userid[,DUMP]
the id of the time-sharing terminal user associated with the terminal session to be canceled. If a dump is requested use the DUMP parameter.

Example: To cancel a terminal session with the associated terminal userid of MOTOR, enter:

```
C U=MOTOR
```

Foreground Initiated Background Jobs

A foreground initiated background job is a job entered by a time-sharing terminal user through a terminal and run with the regular batch processing jobs. The terminal user enters the TSO terminal command SUBMIT with the job names. The SUBMIT command puts the job name on the background reader queue (BRDR queue) and stores the job. The jobs are moved from the foreground to the background by the background reader. The background reader, which you must start, takes the job from the BRDR queue and puts it on one of the 15 input work queues. Now the job is in the background.



Foreground Initiated Background Jobs

STARTING THE BACKGROUND READER

To start the background reader use the START command with the cataloged procedure name BRDR. BRDR is the IBM assigned name of the cataloged procedure in the SYS1.PROCLIB. A user assigned name can be assigned to the background reader cataloged procedure by your system programmer.

After the background reader is started it continually checks the background reader queue (BRDR queue) for jobs. When the background reader finds a job on the BRDR queue, it will then move the job from the foreground to an input work queue in the background. When the BRDR queue is empty the background reader will release the main storage used by the queue. Because the background reader uses only a small amount of main storage when it is not reading jobs, you can start the reader and leave it active as long as it is needed.

$\begin{Bmatrix}\text{START}\\ \text{S}\end{Bmatrix}$	BRDR

BRDR
    the IBM assigned cataloged procedure name for the background reader. A user assigned name can also be used.

STOPPING THE BACKGROUND READER

To stop the background reader use the STOP command with the cataloged procedure name BRDR. BRDR is the IBM assigned name of the cataloged procedure in the SYS1.PROCLIB. A user-assigned name can also be assigned by your system programmer. In either case use the same name that was used to start the reader.

$\begin{Bmatrix}\text{STOP}\\ \text{P}\end{Bmatrix}$	BRDR

BRDR
    the IBM assigned catalog procedure name for the background reader. A user assigned name can also be used, just be sure you are stopping with the same procedure name used in the START command.

DISPLAYING THE BACKGROUND READER QUEUE (BRDR QUEUE)

You can display the number of jobs or a list of job names on the BRDR queue. Before stopping the background reader or TSO, check the BRDR queue for jobs. If there are jobs on the BRDR queue, wait until the jobs are moved from the foreground to the background before stopping the reader or TSO. To check the BRDR queue for jobs use the DISPLAY command with the parameter Q=BRDR.

If you are requested to cancel a foreground initiated background job and the job has not been run or is not active in the system, check the BRDR queue. To check the BRDR queue for a job name use the DISPLAY command and the parameter N=BRDR.

TSO

{DISPLAY | D}	{Q=BRDR | N=BRDR}

Q=BRDR
the number of jobs on the BRDR queue is displayed.

N=BRDR
a list of job names on the BRDR queue is displayed.

CANCELING A JOB ON THE BACKGROUND READER QUEUE (BRDR QUEUE)

When you are requested to cancel a foreground initiated background job and you have located the job on the BRDR queue, cancel the job off the queue. To cancel the job use the CANCEL command, the job name, and the parameter IN=BRDR.

{CANCEL | C}	jobname,IN=BRDR

jobname
the name of the job on the BRDR queue.

IN=BRDR
the job to be canceled is on the BRDR queue.

Example: To cancel the job MOTORA on the BRDR queue, enter:

```
C MOTORA,IN=BRDR
```

Stopping TSO

To stop TSO use the STOP command with the cataloged procedure name used in the START command. The STOP command will terminate all TSO terminal sessions and TSO activity in the system.

{STOP | P}	procname

procname
the name of the cataloged procedure used in the START command for the time-sharing option.

Index

OPERATOR'S CONSOLES

Contents

OC

Operator's Consoles

This chapter describes the two basic console arrangements or configurations: the primary console configuration and the multiple console configuration.

The primary console configuration has one basic console; if there is a second console, it acts as a backup for the primary console and is called the alternate console.

The multiple console configuration has a master console, that is the basic console, and one or more secondary consoles that perform assigned functions. The master console and the secondary consoles each have an alternate console assigned to it.

OC

For information on how to use an IBM 1052 Printer-Keyboard see Operator's Procedures, GC28-6692. For information on how to use an operator display console see Operator's Guide for Display Consoles, GC27-6949.

Primary Console Configuration

The primary console configuration consists of either a single primary console or one primary plus one alternate console. The primary console is the console you normally use to communicate with the operating system. You use the alternate console only when it is necessary to bypass the primary console, perhaps because of a console malfunction. You shift to the alternate console by hitting the INTERRUPT key on the system control panel. Either the primary or the alternate console may be a composite console. A composite console is made up of an input and an output device, usually a card reader and a printer.

You enter your commands to the system through your console or through the system input (SYSIN) stream by using command statements. Commands entered through the SYSIN stream are accepted as soon as they are read -- they aren't synchronized with the jobs in the SYSIN stream.

Commands included in the job stream may not take effect immediately. For example, if you wish to stop a job stream and to do so you include a STOP command in that job stream, a few of the cards that follow the command may be read. However, the command will prevent any subsequent jobs from being placed on the job queue.

How to Shift to an Alternate Console Device

- Issue the VARY command on the primary console to make the alternate console offline.
- Hit the INTERRUPT key on the control panel.

If the alternate console is an IBM 2250 Display Unit Model 1, see Operator's Guide for Display Consoles, GC27-6949 for specific considerations.

How to Bypass a Console Malfunction

If a console malfunction occurs on a console device with an alarm bell, the system normally sounds an alarm three times in a row and suspends console activity. A malfunction can also be recognized by the appropriate indicator lights on the console. One of two actions can be taken:

1. Correct the malfunction, if possible, and make the console not ready, then ready. Processing will continue using the same console. The operation will be retried.
2. If an alternate console is available, hit the INTERRUPT key on the system control panel to switch to the alternate console. Next, make the alternate console not ready, then ready, and issue an operator command (for example, DISPLAY A) on the alternate console. Processing will continue using the alternate console.

How to Enter a Command Into the Input Stream

- Place the command statement card (or group of cards) into the card reader. Do not place the command statement card into the input stream at random; it must be followed by a JOB, EXEC, null, or delimiter statement card.

- If the command you have entered into the input stream appears on the console and you receive message IEF166D REPLY Y/N TO EXECUTE/SUPPRESS COMMAND, enter REPLY id,'Y' if you decide the command should be executed. Enter REPLY id,'N' to suppress execution of the command. If only the command appears on the console, it is being executed.

Command Statement Format: The command statement card has a // in columns 1 and 2, followed by a blank and the text of the command. The command must fit on one card; it cannot be continued.

Multiple Console Configuration

The multiple console support (MCS) option of the operating system allows one operating system configuration to have the use of many operator consoles. Each console performs one or more defined functions; for example, a console near the tape drives might be defined to handle all messages and commands related to tape operations.

Each console in a multiple console configuration is defined (either at system generation time or during system operation) by specifying:

- The operator commands that the system will accept from that console.
- The console that will act as the alternate. In a system with MCS, every console is assigned an alternate to take over its functions if a failure occurs. If a malfunction is detected, the system switches automatically to its specified alternate.
- The message groups that the console will receive. This definition is accomplished by specifying one or more routing codes that a console is eligible to receive. Routing codes are decimal numbers assigned to each system message according to its function. For example, all mount messages for tape would have the same routing code, and the console near the tape drives would be assigned to receive that routing code.

OC

The following list briefly defines the function of each routing code. See "Operator Messages" in Messages and Codes, GC28-6631, for a more detailed discussion of routing codes.

Code	Meaning
1	Master Console Action
2	Master Console Information
3	Tape Pool
4	Direct Access Pool
5	Tape Library
6	Disk Library
7	Unit Record Pool
8	Teleprocessing Control
9	System Security (such as password checking)
10	System Error Maintenance
11	Programmer Information
12	Emulator Programs
13-15	Reserved for customer use
16	Reserved for future expansion

In a system with MCS, one console acts as the master console and the rest (up to thirty-one) are secondary consoles. Any of the consoles in the configuration can be a composite console.

Only one master console can be active at any one time. The master console is the basic console required for operator-system communication; it alone can perform the following functions:

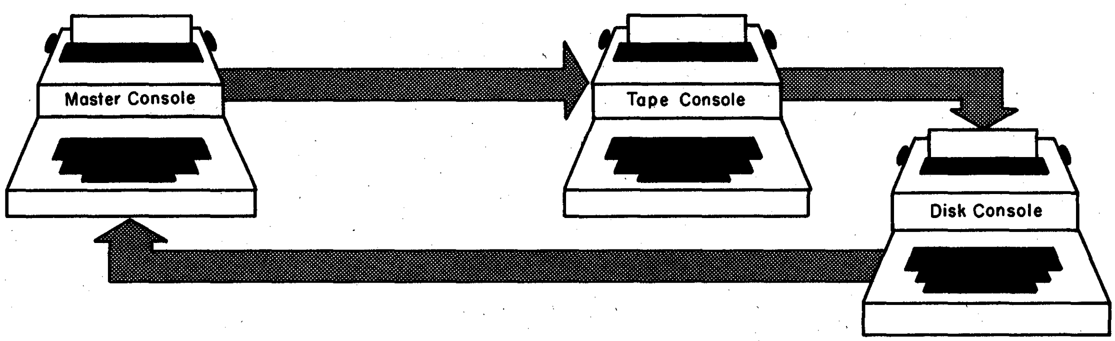
- Enter all possible operator commands.
- Change the status of the hard copy log and the messages to be recorded on it.
- Switch to a different master console.

In addition, the master console receives all messages not specifically assigned to any other console.

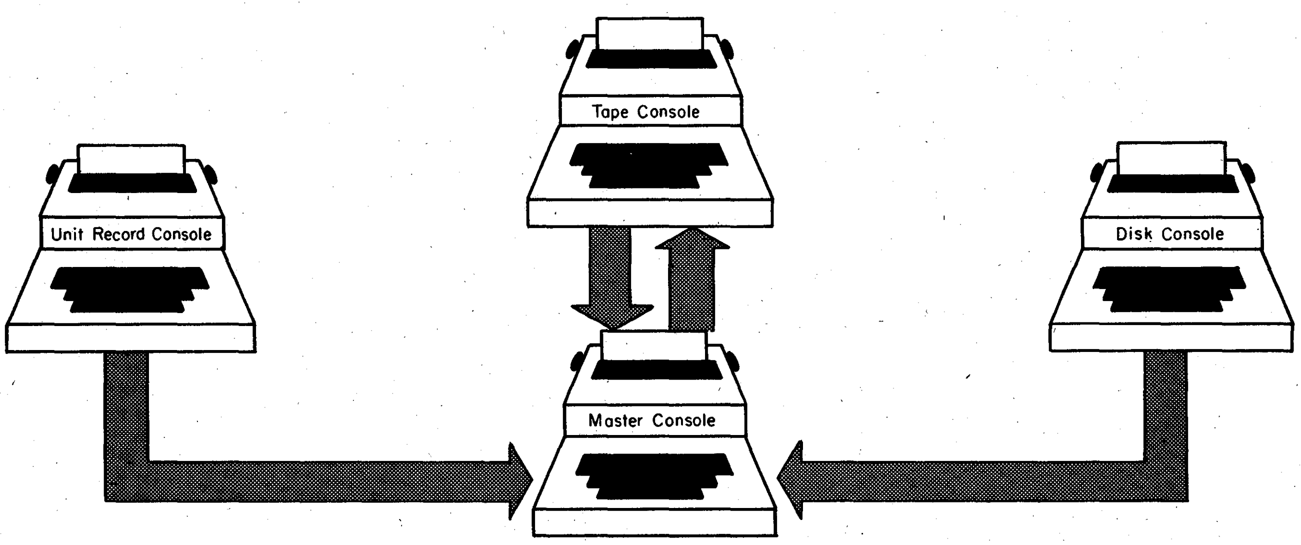
A secondary console is any console other than the master console. It can be local or remote. A secondary console can handle more than one system function, and the same system function can be handled by many consoles.

Each console in a system with the MCS option is assigned another console in the configuration as an alternate console to provide backup when the console malfunctions. The alternate console will usually be another functioning console in the system configuration. For example, the alternate to the console handling tape activity might be the console handling disk activity. If the tape console malfunctions, the operator at the disk console will receive a message telling him his console has now taken over the functions of the failed console.

Alternate consoles can be arranged in alternate chains or loops. In the chain illustrated below, the tape console will automatically take over if the master console malfunctions, the disk console is the alternate for the tape console, and the master console will take over if a malfunction occurs on the disk console.



Alternate consoles need not be arranged in alternate chains, however. In the example below, the master console is the alternate for all consoles in the configuration; the tape console is the alternate for the master console.



The hard copy log provides a permanent record of system activity. It is mandatory in systems with either an active graphic console or more than one active console of any type. The hard copy log can be written on either a non-graphic console device or the system log; it provides a permanent record of needed maintenance information. If your installation chooses, the log can also record operator and system commands and their responses, as well as messages issued by the system and problem programs.

See "How to Specify Hard Copy" in the chapter "General Operating Techniques." for instructions on how to define the contents of the hard copy log during system initialization. It can also be defined at system generation or by you during system operation. See "How to Change the Status of the MCS Hard Copy Log" later in this chapter for instructions on defining the log during system operation.

How to Bypass a Console Malfunction

OC

If a console malfunction is detected on any console in a system with the MCS option, the system will try to bypass the inoperative console by automatic console switching. To bypass a failing master console, however, it may be necessary for secondary console operators to use command-initiated console switching. Manual console switching, which also applies only to the master console, is used when the system cannot detect the hardware malfunction.

Automatic Console Switching: When the operating system discovers a console failure, it automatically tries to switch to the alternate of the failing console. The system inspects the alternate chain of the failing console to find an active console. The first active console found is assigned the functions of the failing console. If no active console is found in the alternate chain, the master console is assigned as the alternate.

When an alternate is found, unanswered WTOR messages and unissued WTO and WTOR messages are transferred from the failing console to the alternate console. The operator of the alternate console receives a message stating which functions his console has assumed from the failing console.

Command-Initiated Console Switching: If an automatic console switch fails for the master console (perhaps because there are no active consoles in the master console's alternate chain), all active consoles in the system are sent a message stating that the system is operating without a master console. Any operator can then enter the VARY command with the MSTCONS operand to define his console as the master console. See "How to Change the Status of the Master Console" later in this chapter for more details.

Manual Console Switching: The INTERRUPT key on the system control panel permits you to bypass a hardware malfunction on the master console that cannot be detected by the operating system. By pressing the INTERRUPT key, you cause the alternate to the master console to become the new master console. (In an M65 MP system, the INTERRUPT key pressed must be on the side from which the IPL program was executed, and there must be an available alternate console.)

A message is sent to the new master console operator informing him that his console is now the master console. The message also informs him of the unit address of the old master console so that corrective action may be taken.

The INTERRUPT key on the control panel of an IBM 2150 Console can also be used to switch to a new master console.

Preserving the Hard Copy Log: When the failing console is the hard copy log, the operating system inspects the alternate chain of the failing console to find an active non-graphic console. The first active non-graphic console found is assigned the hard copy log function. If the search is not successful, the master console and its alternate chain are inspected. If this search is not successful, any active non-graphic console may be selected. A graphic console is selected only if no non-graphic consoles are available. See the discussions on the IBM 2250 Display Unit and the Model 85 Operator Console with CRT Display on additional requirements when a graphic console is the hard copy log.

Recovering Lost Messages: Some console malfunctions will prevent waiting messages from being displayed on the console. The following procedure can be used to avoid losing the waiting messages:

1. Use the VARY command to make the console with the malfunction the master console.

2. Hit the INTERRUPT key on the control panel. This action will cause the inoperative console's functions and waiting messages to be automatically switched to the alternate console.

How to Place a Console Device in Offline Status

When a secondary console must be bypassed for some reason, enter a VARY command to place the secondary console in an offline status. Command activity from the console is immediately suspended but messages continue to be displayed until all waiting messages have been issued.

As a result of the VARY command, the functions of the bypassed console are not assigned to another console. However, any messages that would be routed exclusively to the bypassed console are now routed to the master console by default.

Before you use the VARY command to bypass a console that was the hard copy log device, change the hard copy log function to another console device or to the system log. See "How to Change the Status of the MCS Hard Copy Log" below for instructions.

How to Recover From a No Console Condition

A no console condition exists when all consoles appear inoperative to the operating system. The condition is indicated in two possible ways:

1. The system continues to process jobs but all message activity stops.

2. The console alarm on a 1052, 2250, or Model 85 Operator Console with CRT Display will ring three times.

To recover from the no console condition, do the following:

1. Ready the master console.

2. Hit the INTERRUPT key on his operator control panel. (In an M65 MP system, the INTERRUPT key pressed must be on the side from which the IPL program was executed.)

If an initialization message is not issued (or if the alarm rings again), repeat these two steps.

Then, if the no console condition still persists, tell the supervisor and perform the following procedures:

1. Ready one or more secondary consoles.

2. Perform another IPL. The system will search the master console, the alternates to the master console, and all secondary consoles in that order. The system assigns the first active console that it encounters as the master console, and initializes all consoles that you have made ready.

OPERATOR COMMAND GROUPS

Every operator command that may be used with an MFT or MVT system is placed in one of four command groups according to the function of the command. The four command groups and their functions are:

- Informational commands (group 0)
- System Control commands (group 1)
- I/O Control commands (group 2)
- Console Control commands (group 3)

The commands that make up each group are listed below.

OC

Informational commands may be entered from any console. However, for a secondary console to enter System Control, I/O Control, or Console Control commands, the particular command group must have been assigned to that console during system generation or system operation. The master console, on the other hand, can enter all possible operator commands. In addition, only the master console can enter the special operands of the VARY command that are listed below.

Command Group	Function	Commands	
0	Informational (INFO)	BRDCST DISPLAY LOG MONITOR	MSG REPLY SEND SHOW
1	System Control (SYS)	CANCEL CENOUT DEFINE (MFT only) HALT HOLD MODE (M85 only) MODIFY QUIESCE	RELEASE RESET SET START STOP USERID WRITELOG
2	I/O Control (IO)	MOUNT UNLOAD	VARY SWAP
3	Console Control (CONS)	VARY	
	Master Console only	All command groups are valid, plus VARY MSTCONS, VARY HARDCPY, VARY CPU, VARY CHANNEL, and VARY STORAGE.	

Notes:
1. VARY (Group 2) is accepted only to place a non-console device online or offline. VARY (Group 3) provides only for console switching and console reconfiguration.
2. CONTROL, a command used only in a Model 85 system to control the CRT Display of the Model 85 Operator Console, has not been assigned a group since it is never seen by the command processing routines of the system.

To determine the commands that each secondary console is authorized to enter, use the DISPLAY CONSOLES command to display the system console configuration. If a secondary console enters a command it is not authorized to enter, the system rejects the command and sends an error message to the issuing console.

You enter your commands to the system through your console or through the system input (SYSIN) stream by using command statements. Your installation will decide which operator commands are valid when entered into the input stream by assigning a command group to the reader. Your command is rejected if it is not one that the reader is authorized to enter.

Commands entered through the SYSIN stream are accepted as soon as they are read -- they aren't synchronized with the jobs in the SYSIN stream.

Commands included in the job stream may not take effect immediately. For example, if you wish to stop a job stream and to do so you include a STOP command in that job stream, a few of the cards that follow the command may be read. However, the command will prevent any subsequent jobs from being placed in the job queue.

How to Enter a Command Into the Input Stream

- Place the command statement card (or group of cards) into the card reader. Do not place the command statement card into the input stream at random; it must be followed by a JOB, EXEC, null, or delimiter statement card.

- If the command you have entered into the input stream appears on the console and you receive message IEF166D REPLY Y/N TO EXECUTE/SUPPRESS COMMAND, enter REPLY id,'Y' if you decide the command should be executed. Enter REPLY id,'N' to suppress execution of the command. If only the command appears on the console, it is being executed.

Command Statement Format: The command statement card has a // in columns 1 and 2, followed by a blank and the text of the command. The command must fit on one card; it cannot be continued.

THE VARY COMMAND IN SYSTEMS WITH MCS

In a system with MCS, the VARY command is used to change the status of secondary consoles, the master console, and the hard copy log. The VARY command operands that relate to each MCS function are discussed under the section for that function: "How to Reassign Secondary Consoles," "How to Change the Status of the MCS Hard Copy Log," and "How to Change the Status of the Master Console."

Note: Other functions of the VARY command are described under "Model 65 Multiprocessing (M65 MP)" and "VARY -- Vary Status of Device" for MFT and MVT.

How to Reassign Secondary Consoles

The VARY command can be used to change the status of operator's consoles. An operator's console is any device with input/output capability that has been specified as a console at system generation. Any secondary console can be assigned this command capability. The VARY command in group 3 can be used to:

- Place one or more secondary consoles in an online, offline, or console status.

* Change a secondary console's routing code assignment, command entering capability, and alternate console.

* Change the master console's routing code assignment and alternate console.

When you use a VARY command to place an active console in an online or offline status, the command-entering and command-receiving capabilities of the console are not assumed by any other console. These capabilities are assumed by an alternate console only when an unrecoverable I/O error on a console causes automatic console switching.

An unauthorized entry or an improperly specified command causes an error message to be returned to the console entering the command.


```
Operation   Operand

{VARY}      ({unit             } [,{unit             }]...)
{ V  }      ({O-unit           } [,{O-unit           }]
            ({(I-unit,O-unit)  } [,{(I-unit,O-unit)  }]

            {,ONLINE                                                     }
            {                                                            }
            {,OFFLINE                                                    }
            {                                                            }
            {,CONSOLE [,AUTH={ALL                          }]             }
            {         [      {INFO                         }]             }
            {         [      {([SYS][,IO][,CONS])          }]             }
            {                                                            }
            {         [,ROUT={ALL                           }]            }
            {         [      {NONE                          }]            }
            {         [      {(routecode[,routecode]...)    }]            }
            {                                                            }
            {         [,ALTCONS={unit             }]                      }
            {         [         {O-unit           }]                      }
            {         [         {(I-unit,O-unit)  }]                      }
```

unit or O-unit or (I-unit,O-unit)
    the unit address of the operator's console whose status you want to change.

unit
    the unit address of a device with input and output capability.

O-unit
    the unit address of a device with output capability (1403 or 1443).

(I-unit,O-unit)
    the unit addresses of a composite console, where I-unit is the input and O-unit the output address. (Composite consoles are logically paired at system generation and cannot be separated by the VARY command.)

If you are specifying only one unit or O-unit, you do not need parentheses.

You can specify multiple unit addresses and include units, O-units, and composite consoles in the same sequence.

OFFLINE
: the device is not to be used as an operator's console or as an input/output device assigned to a problem program. If the device is in an online status, the status is not changed until all the current users have finished with the device. If the device is being used as a console, command activity is terminated, but message activity continues until the console output queue is empty. The screen on a graphic console is blanked when all messages have been displayed.

Your VARY OFFLINE command will be rejected if you attempt to remove the hard copy device from the system when hard copy is required (when there is an active graphic console or more than one active console in the system). Before using VARY OFFLINE to remove the device, you should reassign the hard copy log. See "How to Change the Status of the MCS Hard Copy Log" for details.

An attempt to place the master console offline is ignored.

ONLINE
: the job scheduler can allocate the device to problem programs, an input reader, or an output writer. The device is made available for general system use. If the device is being used as a console, command activity is terminated, but message activity continues until the console output queue is empty. The screen on a graphic console is blanked when all messages have been displayed.

Your VARY ONLINE command will be rejected if you attempt to remove the hard copy device from the system console configuration when hard copy is required (when there is an active graphic console or more than one active console in the system). Before using VARY ONLINE, you should reassign the hard copy log. See "How to Change the Status of the MCS Hard Copy Log" for details.

An attempt to place the master console online is ignored.

CONSOLE
: (1) that a device is to be added (initialized) to the active console configuration or (2) that the command, routing, and/or alternate console status of an active console is to be modified by using the AUTH, ROUT, and/or ALTCONS subparameters with the CONSOLE parameter.

A console is activated at system initialization with the command, routing, and alternate console assignments made at system generation. Subsequent VARY commands may change the assignments of the console, as well as changing it from active to inactive status and back again.

If the device is being added to the active console configuration after system initialization, the command, routing, and alternate status will remain the same as when the device was last in an active console status unless you change them by using AUTH, ROUT, and ALTCONS. The console initialization process includes sending a message to the issuing console. This message includes the same information provided by the DISPLAY CONSOLES operator command defined under "DISPLAY -- Cause Console Display" for MFT or MVT.

AUTH
: the operator command groups that the console is authorized to enter. Use AUTH to change the previous command group assignment of the console; omit AUTH if you want the command-entering capability to remain the same.

ALL
: specifies that all operator command groups are valid except those reserved for the master console; command groups 1, 2, and 3 are assigned.

INFO
: the command-entering capability is to include only the DISPLAY, REPLY, LOG, SHOW, BRDCST, and MSG commands. No command group is assigned since these commands are valid for every system console.

SYS
: the command-entering capability of the console is to include group 1 commands.

IO
: the command-entering capability of the console is to include group 2 commands.

CONS
: the command-entering capability of the console is to include group 3 commands.

If you are entering only SYS or IO or CONS, you do not need parentheses.

The AUTH parameter is ignored if the specified console is:

1. A console with only output capability.
2. The master console. The master console is authorized to enter all commands.

ROUT
: the routing codes the console is assigned to receive. These routing codes replace previous routing code assignments.

ALL specifies that all routing codes are desired, thus permitting all operator messages (WTO and WTOR) to be received.

If all operator messages are not desired, one or more specific routing codes may be specified with the routecode subparameter, which must be a number from 1 through 16. If you are specifying only one routing code, you do not need parentheses.

NONE specifies that no routing codes are desired. If NONE is specified for the master console, the default value is routing code 1.

If ROUT is not specified, the routing code assignment of the console is not modified.

ALTCONS
: the unit address of the alternate console. If this parameter is omitted, the alternate console is not changed.

unit
: the unit address of a console with both input and output capability.

O-unit
: the unit address of an output device (1403 or 1443). O-unit can only be used to specify the alternate to another O-unit console.

OC

(I-unit,O-unit)
: the unit addresses of a composite console, with I-unit specifying the input console and O-unit specifying the output console.

I-unit and O-unit devices must be logically paired. A console with input and output capability can be an alternate to a console with only output capability, but the relationship cannot be reversed. Specification of a console with only output capability as the alternate to a console with input and output capability is ignored, with the issuing console receiving a message stating the error.

If you specify the unit address of a non-console device, your VARY command will be rejected. (Your command will also be rejected if you specify a console as its own alternate console.)

Note: When the specified console is also functioning as the hard copy log, the AUTH and ROUT parameters only pertain to the console function and do not affect the hard copy output.

Example: To dynamically respecify an active console with a physical device address 009, giving it system and I/O command control, and authorizing routing codes 2, 3, 4, 7, and 13, specify:

```
VARY 009,CONSOLE,AUTH=(SYS,IO),ROUT=(2,3,4,7,13)
```

How to Change the Status of the MCS Hard Copy Log

The VARY command with the HARDCPY operand can be used to assign or modify the hard copy log function. VARY HARDCPY can only be entered through the master console.

The multiple console support option provides for either buffered or immediate recording of messages and commands. This recording, which is called the hard copy log, produces a permanent record that may be retained by your installation to obtain information about operating conditions and for maintenance purposes. It can be used as a collection point for all messages and commands, or it can record a few selected messages.

The system log (which is the only buffered hard copy device supported) or a non-graphic console can be specified as the hard copy log. The system log must be specified by your installation at system generation if you are to use it during system operation.

Operation	Operand
{VARY V}	{unit SYSLOG},HARDCPY [{,CMDS ,NOCMDS ,OFF}] [,ROUT={ALL NONE (number[,number]...)}]

unit
: the unit address of the device that is to assume the function of the hard copy log. It must be an active system console. If it is not, an error message is issued.

  If both unit and SYSLOG are omitted, the current hard copy log is modified. If you omit both unit and SYSLOG (indicating you intend to modify the current hard copy log), you will receive an error message if no hard copy log has been previously defined. Use a comma when you omit both unit and SYSLOG.

A graphic device cannot be specified as the hard copy log device (an attempt will be rejected). If a composite console is to be used, unit specifies the unit address of the output device.

SYSLOG
: the system log is to be used as the hard copy log device. If the system log does not exist, or if the system log is not supported, an error message is issued if this operand is specified. When SYSLOG and unit are omitted, the specified modifications are performed on the current hard copy log device. To indicate that you are not using either the unit or SYSLOG parameter, use a comma. (See Example 1 below.)

HARDCPY
: the status of the hard copy log is to be changed. In MCS systems without a previously-defined hard copy log, use of this operand initiates the log function. This parameter must be specified.

OC

CMDS
: the operator and system commands, and the responses to those commands, are to be written on the hard copy log.

NOCMDS
: no operator and system commands and responses are to appear on the hard copy log. When CMDS and NOCMDS are omitted, the command logging status of the hard copy log is not modified unless hard copy is mandatory.

  Note: When hard copy is mandatory, commands and responses are sent to the log by the operating system, regardless of the CMDS or NOCMDS specification.

OFF
: the hard copy function is to be eliminated. If you use this parameter when hard copy is mandatory, your command will be rejected. When you use OFF, it must be the last parameter; you then cannot use ROUT.

ROUT
: the routing codes the hard copy log is assigned to receive.

  ALL
  : all routing codes are desired. If all operator messages are not desired, one or more specific routing codes may be specified (see Note below).

  NONE
  : no routing codes are desired.

  number
  : a routing code from 1 through 16.

If ROUT is not specified, the routing code assignment of the hard copy log device is not modified.

Note: The hard copy log routing code assignment may be reduced when the hard copy log is mandatory, but only to the routing code values that are required.

If your operating system has an active graphic operator's console, the hard copy log is required to eliminate the loss of information when the operator of a graphic console deletes messages and operator commands from his screen. The operating system ensures that those messages that go only to graphic devices go also to the hard copy log. An attempt to remove the hard copy device from the console configuration will be rejected.

The hard copy log is also required when there is more than one active console. If required, the log must receive commands, command responses, and messages with routing codes 1 through 4, 7, 8, and 10. As the system will automatically assign these requirements if you do not, you need use the ROUT operand of the VARY HARDCPY command only when your installation wishes to record more than the basic information needed.

An attempt to add a second active console to the system when hard copy is not present causes the system to assign the master console as the hard copy log.

If you try to add a second active console when the master is a graphic console acting as its own hard copy log, the system will assign the new console as the hard copy log if the new console is non-graphic. For the Model 85 only, the new console can also be a graphic console. If it is, the system will check to see if the system log is present. If the system log is present, the system will assign the hard copy log function to the system log; if the system log is not present, the graphic master console will remain as the hard copy log.

An attempt to remove a console (or the system log) from the system when it is functioning as a hard copy log when hard copy is mandatory will be rejected.

Example 1: Remove the hard copy log function (when the hard copy log is not mandatory):

```
VARY ,HARDCPY,OFF
```

Example 2: Change hard copy recording from a console to the system log.

```
VARY SYSLOG,HARDCPY
```

How to Change the Status of the Master Console

The VARY command can be entered through the master console to shift to a new master console. This enables the master console to be serviced without disrupting system operations.

When the master console is operative, the entry of the VARY MSTCONS command by a secondary console operator causes no action to be taken. The secondary console operator receives a message indicating that no console switch was made.

When the master console is inoperative, the entry of the VARY MSTCONS command is valid from any secondary console. Normally the operating system detects that the master console is inoperative and performs an automatic console switch to the alternate for the master console. Occasionally a failure is not detected. The master console operator could press the INTERRUPT key on the operator's control panel and cause a console switch to the alternate of the master console. (In an M65 MP system, the INTERRUPT key pressed must be on the side from which the IPL program was executed.) Otherwise, he should place his console's ON/OFF switch in the off position; another operator should issue the VARY command with the MSTCONS operand from an active secondary console.

An automatic or manual (INTERRUPT key) console switch to a new master console cannot be made when all the alternate consoles in the master console alternate chain are inactive. When this occurs, all active secondary consoles receive a message asking that the VARY command with the MSTCONS operand be entered to assign a new master console. The first VARY command entered with the correct syntax is accepted and processed.

The new master console operator receives a message informing him that he is now the master console operator. The message also provides him with the unit address of the previous master console.

Operation	Operand
{VARY V}	{unit (I-unit,O-unit)} ,MSTCONS

OC

unit or (I-unit,O-unit)
: the unit address of the new master console. If a composite console is to be used, I-unit specifies the unit address of the input device and O-unit specifies the unit address of the output device. The unit address must specify an active system console with input/output capability. If the console is not active, or is an output unit, you will receive message IEE313I UNIT REF INVALID; your command will be ignored.

MSTCONS
: a master console switch is to be performed.

Index

OC

GENERAL OPERATING TECHNIQUES

Contents

GOT

General Operating Techniques

This chapter is a grouping of techniques and related data you will need to use in running the operating system. You will probably learn these techniques thoroughly, making revisions to them to suit your own needs. And of course you will be starting a "run book" of your own, adding tips and procedures as you discover them.

How to Specify a Hard Copy Log (Systems With MCS Option Only)

If you receive message IEA150A SPECIFY HARDCPY when the nucleus is initialized, you must specify a hard copy log. The hard copy log can be any system operator's console or it can be the system log if the system log data sets were provided at system generation. Only the master console operator receives this message.

You receive the SPECIFY HARDCPY message only when the hard copy log is mandatory and it has not been specified at system generation or by the HARDCPY option of message IEA101A SPECIFY SYSTEM PARAMETERS. The log is mandatory when an active graphic console exists in the console configuration, or when more than one active console is in use.

If the system log was designated as the hard copy log at system generation, you must use the HARDCPY option when you receive the message: IEA101A SPECIFY SYSTEM PARAMETERS. After the HARDCPY keyword, give the address of an operator console with output capability (like the 1052). After the message:

IEE041A LOG NOW RECORDING ON SYS1.SYSVLOG $\left\{\begin{matrix}X\\Y\end{matrix}\right\}$ ON ddd

use the VARY command to change the hard copy log back to the system log.

See "How to Change the Status of the MCS Hard Copy Log."

The operating system will not proceed until a hard copy log device has been accepted. To respond to the message, enter:

REPLY id,'HARDCPY=($\left\{\begin{matrix}\text{address}\\\text{SYSLOG}\end{matrix}\right\}\left[\left\{\begin{matrix}\text{,(number[,number]...)}\\\text{,ALL}\end{matrix}\right\}\right]$ [,NOCMDS])'

SYSLOG
: the system log is to be used as the hard copy log. Both system log data sets must be provided at system generation for the SYSLOG operand to be accepted.

address
: a unit address of an operator's console with output capability. Specification of a graphic console is not accepted.

ALL
: all messages to the operator are to be written on the hard copy log.

number
: a routing code. Only those messages to the operator that have this routing code are written on the hard copy log. The number must be a routing code from 1 through 16, and more than one number can be specified. If this operand and the ALL operand are omitted, no routing codes are assigned to the hard copy log. The system will modify your specifications if the hard copy log is mandatory by assigning those mandatory routing codes that have not been specified.

NOCMDS
: no operator and system commands and responses are desired. This parameter is ignored when the hard copy log is mandatory. If this parameter is not specified, both operator and system commands and command responses are to be written on the hard copy log.

Although the log is mandatory when an active graphic console is in the console configuration, you do not have to specify the routing code or commands parameters when asked to SPECIFY HARDCPY. Messages are written on the log automatically if they go only to a graphic console. Any message that goes to both a graphic and a non-graphic console is not sent to the log unless its routing code has been assigned to the log. The operating system also writes commands on the log if they are entered by a graphic console, even if NOCMDS has been specified.

When more than one active console is in use, the hard copy log must receive operator and system commands and their responses, and messages with routing codes 1, 2, 3, 4, 7, 8, and 10. These requirements provide essential maintenance information for IBM service personnel.

In addition to the maintenance information, your installation may want you to specify additional routing codes. You may specify only the routing codes that your installation needs over and above the mandatory routing codes, or you may specify all routing codes.

If your system includes the Timer option, a time stamp precedes all messages and commands sent to the hard copy log. All messages sent to the log also include a field that identifies the routing code or codes for the message. See the <u>Messages and Codes</u> publication for a description of these fields. If your system does not include the Timer option, the time stamp field will be all zeros.

Using the System Management Facilities (MFT & MVT)

The System Management Facilities (SMF) option of the operating system enables the user to:

- Determine system usage on the basis of each program's use of the CPU, I/O devices, and storage.
- Evaluate system usage by comparing the information gathered by SMF against installation standards or against recorded usage at similar installations.

SMF gathers, formats, and records system management information in accordance with SMF values specified at system generation time. If operator intervention is a part of your installation's SMF, you may also enter these values through the console.

OPERATOR INTERVENTION

At IPL time you will receive the message:

```
IEE354I  SMF PARAMETERS
```

The system will then list the SMF parameters specified at system generation time. The list is followed by the message:

```
id IEE357A  REPLY WITH SMF VALUES OR U
```

If you have no requests from your system programmer to change the SMF values, enter:

```
REPLY id, 'U'
```

If you have requests from your system programmer to change the SMF values, enter:

```
REPLY id, 'text'
```

'text'
: SMF parameters to be changed. The entire text is enclosed in apostrophes. Each parameter consists of a keyword, an equal sign, and the values. Separate each parameter with a comma.

  Example: REPLY id,'ALT=123456,BUF=5000'

SMF OUTPUT

The information gathered by SMF and user routines is recorded on a SMF system data set, which may be either a direct access or a tape volume.

DIRECT ACCESS: SMF data is recorded on the primary data set SYS1.MANX and the alternate data set SYS1.MANY; they need not be on the same volume. Be sure that both data sets are online at IPL time.

When one data set is full, you will receive a message telling you to dump the filled data set; data recording will switch automatically to the alternate data set.

GOT

To dump the filled data set, use the program IFASMFDP. To speed execution, submit the dump job before it is needed, hold it in the job queue, and release it when necessary. Sample JCL is as follows:

```
//DUMP{X}   JOB  ...
      {Y}

//STEP1     EXEC PGM=IFASMFDP

//DUMPIN    DD   DSN=SYS1.MAN {X}
                              {Y}

//               UNIT=diskaddr,

//               VOL=SER=serial,

//               DISP=OLD

//DUMPOUT   DD   UNIT=tapeaddr,

//               VOL=SER=serial,

//               DISP=(NEW,KEEP)

//SYSPRINT DD    SYSOUT=A
```

The variables are:

diskaddr
    the disk drive you will use for input.

serial
    the volume serial number of the disk volume used for input and the tape volume used for output.

tapeaddr
    the tape drive you will use for output.

TAPE: SMF data is recorded on data set SYS1.MAN, which you must mount on a device that is online at IPL time. SMF positions the tape to its beginning at IPL time. Up to 20 tape volumes may be filled before SMF requires you to re-IPL. To avoid destroying existing data, mount a new tape each time you IPL.

ERROR RECOVERY: If an I/O error occurs while writing to SYS.MANX or SYS1.MANY on a direct access volume, you will receive a message and SMF will switch to the alternate data set. Allow the system to become inactive, allocate a new data set to replace the bad one, and re-IPL. When the SYS1.MAN data set is on tape, mount a new tape in response to the I/O error message.

How to Handle System Failures

When a system failure occurs, the system attempts to write a core image dump on a system data set, SYS1.DUMP, recover from the failure, and continue processing.

Creating SYS1.DUMP

SYS1.DUMP can reside either on direct access storage or on tape. If it is on direct access storage, it is created before IPL time. If SYS1.DUMP is to be on tape, your programmer will tell you to create the data set during nucleus initialization. To do so, follow these procedures:

- Respond to message IEA135A SPECIFY SYS1.DUMP TAPE UNIT OR NO with the unit address of a 2400 series tape drive. The message is repeated if there is an error in your response. Reply NO if you have been told not to create the dump data set.

- You'll then receive a message asking you to mount an unlabeled tape on the device you have specified. If you mount a labeled tape, the MOUNT message will be repeated.

GOT

Note: When you expect the system to encounter a large number of errors, creating SYS1.DUMP on tape is most efficient.

Responding to a System Task Failure (MFT and MVT)

When the system sends you the message IEA029I jobname stepname TASK REINSTATEMENT FAILED or the message IEE404I (MVT only), a system task has failed. The system has abnormally terminated the failing task, but will attempt to complete other tasks that are already scheduled.

The resources (storage, I/O devices, and data sets) in use by the failing task are not freed. Do not reallocate these resources. To continue normal system operation, allow the system to come to a stop by taking the following actions:

- Enter a HOLD Q command to stop the scheduling of jobs.

- Stop all readers and writers.

- After the scheduled tasks have completed, restart the system. If you have received the message IEA023I or the message IEE404I (with a code of 30) run the IMPRDMP service aid to print the SYS1.DUMP data set.

CAUTION: Blanks may appear in the jobname stepname field of message IEA029I for a problem program also, but only if the programmer has not provided a jobname and/or stepname for his job. See "Responding to a Problem Program Failure" in the MFT or MVT chapters for a discussion of handling problem program failures.

Abnormally Ending 'Must Complete' Tasks (MFT and MVT only)

A task that is exclusively using a system resource (the task must finish using the resource before any other task can use it) is called a 'Must Complete' task. The following processing will always occur in MVT. However in MFT, insufficient main storage may prevent the writing of the messages.

If a 'Must Complete' task ends abnormally, you will first receive message IEA027I jobname stepname ENQUEUED RESOURCES, followed by a list of the resources being exclusively used by the failing task. In response to the next message, IEA028A REPLY 'C' OR 'N' enter REPLY id, 'C; and inform your system programmer of the failure. Respond 'N' only if you have been authorized to release the task from 'Must Complete' status.

Printing the Core Image Dump

The SYS1.DUMP data set can hold only one core image dump at a time. When you receive message IEA023I jobname stepname CORE IMAGE COMPLETE-1, or message IEE404I with an error code of 30, perform one of the two following procedures:

- If SYS1.DUMP is on a direct access volume, schedule the print dump program (IEAPRINT for PCP or IMDPRDMP for MFT and MVT) to print out the contents of the data set and make it ready to receive another core image dump.

- If SYS1.DUMP is on tape, the tape will not be rewound and unloaded after the core image dump was taken. The tape must be manually unloaded if a fresh unlabeled tape is to be mounted. Schedule the print dump program (IEAPRINT for PCP or IMDPRDMP for MFT and MVT) if you wish to print out the core image dump recorded on the tape.

System message IEA022I jobname stepname CORE IMAGE FAILED-3 indicates that the core image was not completed because of an I/O error. If you receive this message, at least part of the core image dump has probably been lost. Proceed as if you had received a CORE IMAGE COMPLETE message and ready the data set for the next dump.

MFT and MVT Examples: Your system programmer can describe the procedures for using the print dump program (IMDPRDMP) or a complete description may be found in the publication OS/360 Service Aids, GC28-6719. The first example is when SYS1.DUMP is on disk; the second example is when SYS1.DUMP is on tape.

Example 1:

```
//DUMP       JOB  MSGLEVEL=(1,1)
//           EXEC PRDMP
//DMP.SYSIN  DD   *
             GO
             END
/*
```

Example 2:

```
//DUMP       JOB  MSGLEVEL=(1,1)
//           EXEC PRDMP
//DMP.TAPE   DD   DSN=SYS1.DUMP,UNIT=2400,
//  LABEL=(,NL),DISP=(OLD,KEEP),VOL=SER=xxxxxx
//DMP.SYSIN  DD   *
             GO
             END
/*
```

PCP Example: Your system programmer will describe the procedures for using the print dump program (IEAPRINT). In this example of the JCL that could be used to run IEAPRINT when SYS1.DUMP is on tape, xxxxxx indicates information to be supplied by your installation.

```
//xxxxxx    JOB  xxxxxx,MSGLEVEL=(1,1)
//          EXEC PGM=IEAPRINT
//SYSPRINT  DD   SYSOUT=A
//SYSUT1    DD   DSNAME=SYS1.DUMP,UNIT=2400,LABEL=(,NL),
//               DISP=(OLD,KEEP),VOLUME=SER=xxxxxx
```

GOT

How to Analyze Input/Output Commands

- Look at the channel address word (CAW) at location 48 hex to find the channel command word (CCW). If command chaining was used, the CAW points to the <u>first</u> CCW in the chain.

- Look at the channel status word (CSW) at location 40 hex:

  1. The command address (bits 8-31) points to the <u>last</u> CCW executed plus 8 bytes.

  2. The status portion (bits 32-47) tells the status of the channel control unit or subchannel, and the status of the device the command was issued to (each device has its own meanings for the status bits -- see your hardware manuals). The address of the device is found in bits 16-31 of the I/O old PSW at location 3A hex.

  3. The residual byte count should be zero. If it is not, one of three things is indicated: a wrong-length record was met, indicating a problem with the channel program; the command was rejected by the channel, also indicating a fault in the channel program; a data check during a read or write stopped data transfer and device motion.

     Channel end, device end, unit check, and incorrect length indications are in the CSW, and the residual byte count may show how much data was not transferred.

  4. When working with variable-length records, the wrong-length indicator (bit 34 of the CCW) should be on to prevent I/O interruptions.

- Look at the channel command word (CCW) to find the data address, byte count of the data, command code naming the actual operation, and flag bits for command and data chaining. Except for the transfer-in-channel (TIC) command, there must be a byte count of one or more for any I/O operations.

How to Use the Shared DASD Option

The shared direct access storage device (DASD) option extends the operating system with PCP, MFT, or MVT by letting independently-running systems share the 2301, 2311, 2314, and 2321 direct access storage devices. The option also allows up to three paths from a CPU to a shared 2314.

The option is included in the operating system during system generation. The option allows two or more CPUs to efficiently use a shared device in a controlled way. This offers such advantages as:

- Lessening the amount of time you have to spend moving volumes from one system to another.

- Minimizing the updating of data sets -- you have to update only one instead of two or more duplicates.

- Simplifying scheduling -- you can run a job needing a specific data set on a shared device on any of the sharing systems, that is, you don't have to run it on one and only one system, unless the job has some other requirements, such as that it needs a unique set of devices, a certain control program (MFT for example), or extra amounts of main storage that all the systems don't have.

If your installation is using this option, you may receive special instructions from your management on what to do in certain cases. For example, you may be told:

- How you should keep track of the status of shared devices -- whether they are used by only one system for certain runs, whether you can change their status, on whose authorization, and how.

- How you should tell other operators in your installation when a parallel set of procedures must be carried out on all sharing systems. For example, when a VARY command is necessary on your system to change the status of a shared device, the operators on the sharing systems must also issue VARY commands to maintain a correct combination of mount characteristics.

- What you should do in the case of an enabled wait state on one or more CPUs -- a general discussion of this topic is covered for each system under the topic "How to Determine System Status."

These and other considerations are wholly dependent on your installation's needs. You must get directions on what to do in specific instances from your installation supervision. The following instructions and guidelines are only to be used in conjunction with what your installation has already decided to be the best course of action.

HARDWARE NOTES

A two-channel hardware switch allows the control programming to actually carry out the reserving and releasing of the shared device and data. The two-channel switch allows a common control unit to be switched on a first-come, first-served basis between two channels, each from a different system.

On systems with the 2314, be sure any hardware toggle switches marked <u>tagged</u> and <u>untagged</u> are always set to the <u>tagged</u> position.

IPL FROM A SHARED DEVICE

An attempt to IPL from a shared device will hang if the control unit or device is busy with the other sharing system. Retry the IPL until successful. No adverse effects will arise in the other systems.

SYSTEM RESET

System reset only resets the functions of a shared control unit or device that belongs to the system that was reset -- any function having to do with another system is undisturbed. A selective or malfunction reset has no effect on device reservations or status.

MOUNT CHARACTERISTICS

Volume characteristics, device status, and volume mounting and demounting are all affected by the shared DASD option. One of the following combinations of mount characteristics and status must be in effect for each device which is physically shared between CPUs.

	System A	Other Systems
1.	Permanently Resident	Permanently Resident
2.	Reserved	Reserved
3.	Removable	Offline
4.	Offline	Reserved or Removable -- if a device is removable in one system, it must be offline in all others.

No other combination of mount characteristics is supported. You and your fellow operators must maintain the proper relationships between systems since the systems themselves do not and cannot.

<u>Note</u>: Device allocation must not proceed until valid mount characteristics are established. If the correct combination of mount characteristics is not maintained, the system may issue an unsatisfiable allocation request to which your only alternative is to cancel the job requiring the action; that is, the system may ask you to mount or demount a volume currently being used by another CPU.

<u>Initializing Mount Characteristics</u>

After initial program loading, a valid set of mount characteristics must be established before the system begins device allocation. This can be accomplished in one of the following two ways:

1. By specifying mount characteristics of shared devices in PRESRES, as described in <u>IBM System/360 Operating System: System Programmers Guide</u>, GC28-6550.

2. By varying all shareable devices offline prior to issuing START commands and then following the parallel mount procedures (described in the next section) to establish appropriate mount characteristics. However, if you use this method, do not allow readers, writers, or initiators to start automatically, that is, device allocation must not begin until the valid mount characteristics are established.

Changing Mount Characteristics

The mount characteristics of a volume or the status of a device or both can be changed in one system as long as the resulting combination across all systems is valid.

Volumes that aren't permanently resident must be put in reserved status via the MOUNT command before jobs needing the shared volume are started.

To change volumes that are "reserved" you must:

- Use the VARY command to put the device in offline status in each sharing system and wait for the offline message in each system. If the system is in a quiesced state (no jobs in process), the offline message will not appear on the console.
- Use the MOUNT command to notify each sharing system of the units on which the new volume is being placed, and to put that volume in reserved status.
- Mount the volume.
- Use the VARY command to put the shared device in online status in each sharing system with PCP or MFT. In a system with MVT you will receive a normal allocation recovery message.

GOT

In MFT or MVT systems, no job requiring this new volume should be selected by an initiator before the volume is mounted. Selection before the volume is mounted could result in the abnormal ending of the job or could allow the volume to be allocated to a nonshared device when sharing is desired.

Job selection can be held up by:

1. Using the TYPRUN=HOLD parameter on the job card, or
2. By using the HOLD command, or
3. By assigning the job to a job class and not starting an initiator to that class. After the volume is mounted, use the RELEASE command to let the job be selected in cases one and two, start an initiator to the job class in the third case.

Any shared device can be allocated by only one system by using the VARY command in all the other systems to put the device in offline status in those systems.

OPERATING GUIDELINES

Just how complicated your job is when using the shared DASD option depends on what your installation has told you to do. You may or may not have to change the mount characteristics of a volume or device and perform parallel procedures on sharing systems. Thus the following statements are only general guidelines that don't necessarily apply to your installation.

MOUNTING A SHARED VOLUME

If shared devices are not permanently mounted, you must mount a new volume on a shared device before the job needing that volume is started. If a shared volume is not available when the requesting job is started, you must cancel the job. Don't reply to the allocation recovery message with the three-character unit address of an offline shared device unless the device is offline on all sharing systems.

Don't change volumes until jobs using them are shown as ended by job ended messages displayed in response to your DISPLAY JOBNAMES command. When all of the jobs needing the volume currently mounted have completed, follow the procedures described under "Changing Mount Characteristics" above.

In systems with MFT or MVT, however, you must also temporarily suspend the selection of all jobs requiring volumes not yet mounted. Use the HOLD command, or the TYPRUN=HOLD parameter on the job card, for each job needing the volume. Later use the RELEASE command for each job, after the required volume has been mounted. Or you could assign the job to a job class but not start an initiator to that class until after the volume has been mounted.

Data Sets That Can Be Shared

Any of your installation's own data sets can be shared. In addition, the following system data sets can be shared:

- SYSCTLG when not on the IPL volume
- Volume tables of contents (VTOCs) on all shared volumes
- SYS1.LINKLIB if all sharing systems have identical operating system configurations
- SYS1.PROCLIB when not on the IPL volume.

Data Sets That Cannot Be Shared

The following system data sets cannot be shared.

- SYSCTLG when on the IPL volume
- SYS1.SVCLIB
- SYS1.NUCLEUS
- SYS1.LOGREC
- SYS1.SYSJOBQE
- SYS1.SYSVLOGX
- SYS1.SYSVLOGY
- SYS1.ROLLOUT
- SYS1.ACCT
- PASSWORD
- SYS1.ASRLIB
- SYS1.MAN$\begin{Bmatrix}X\\Y\end{Bmatrix}$
- SYS1.DUMP

TAKING POWER DOWN

When taking power down on any system using the shared DASD option, be sure to follow this sequence:

- Hit RESET
- Disable the shared control units connected to the system
- Hit POWER OFF

SUMMARY

Be sure that you know and follow all your installation's rules and procedures when you operate a system with the shared DASD option. Remember that sharing systems cannot communicate with each other and that therefore it is up to you and your fellow operators to cooperate fully to avoid errors.

You can resume system activity after entering an enabled wait state on all CPUs by using the CANCEL command to cancel one or more jobs as necessary.

When you perform initial program loading on a system, you must ensure that the proper mount characteristics are established before allowing device allocation to proceed. You must know which devices are shared and must not reply to an allocation recovery message with a shared device address unless you are in the process of mounting to that device in an MVT system.

Whenever you change the mount characteristics of a shared volume or the status of a device, or both, you must maintain a valid combination of these characteristics across all sharing systems. Remember that the VARY OFFLINE command is always the first step of the mounting procedure and that the procedure must be done in parallel on all sharing systems.

If there is a hardware malfunction on other than the system residence device, you must vary the device offline on all sharing systems. You can then mount the shared volume on another shared device, if one is available, as long as you follow parallel mount procedures on all sharing systems.

How to Use Dynamic Device Reconfiguration (MFT-MVT)

Dynamic Device Reconfiguration (DDR) allows you to move a demountable volume from one device to another. The request to move the volume may be initiated by either you or the system. You can move volumes to or from SYSRES or non-SYSRES devices. DDR cannot be used for TP devices. Shared DASDs can only be swapped to themselves (a volume is demounted then remounted on the same device).

OPERATOR REQUESTS

If you decide cleaning or maintenance of a device is required, initiate DDR by entering the SWAP command.

- For non-SYSRES devices, provide the address of the device "from" which and "to" which the volume will be moved. Wait for confirmation of the command; then use one of the following to reply:

  YES - proceed as indicated in the SWAP command.

  NO - cancel the request.

  zzz - indicate a new "to" device.

  If you replied YES or zzz, wait for the "PROCEED" message and then swap the volumes.

  For readers, printers, and punches, the system will never request DDR. You can request DDR for these devices when an "intervention required" condition is noted by the message IEA000A, or cause an "intervention required" condition by making the device not-ready. Issue the SWAP command and proceed as above.

- For SYSRES devices, provide the address of the device "from" which and "to" which the volume will be moved.

  You will receive a message reminding you that SYSRES resides on the "from" device. Wait for confirmation of your SWAP command; then use one of the following to reply:

  YES - proceed as indicated in the SWAP command.

  NO - cancel the request.

  zzz - indicate a new "to" device.

  If you replied YES or zzz, wait for the message:

  SWAP SYSRES FROM xxx TO yyy

  then move the SYSRES volume to the "to" device and ready it.

SYSTEM REQUESTS

If a permanent I/O error occurs on a DDR-supported device, and it is an error supported by DDR, the system will request a volume swap.

- For non-SYSRES devices, you are given the choice not to use DDR or not to use DDR. Use one of the following to reply:

  YES - proceed as indicated by the system.

  NO - no DDR action, and the error will be posted.

  zzz - a device name other than the "to" device requested by the system.

  If you replied YES or zzz, wait for the "PROCEED" message and then swap the volumes.

- For SYSRES devices, an alternate device is specified at system generation time; at IPL time, you are given the option to change the device.

  When the system issues the message:

  SWAP SYSRES FROM xxx TO yyy

  move the volume on xxx to yyy, move the volume on yyy to xxx, then ready the devices.

How to Communicate with an Integrated Emulator Program

Your operating system can have several integrated emulator programs. Integreted emulator programs are executed under the control of the operating system.

You communicate with an integrated emulator program by replying to the operator messages with the REPLY command. The 'text' field of the REPLY command is used to contain another command, the emulator command. Since each emulator program has its unique commands, refer to IBM System/360 Operating System: Master Index, GC28-6644, for the publication describing your integrated emulator.

How to Use Direct System Output Processing (DSO) (MFT-MVT)

If Direct System Output Processing (DSO) is used, system output (SYSOUT) data is written directly from a problem program to a specified output device. DSO is started with the START command, stopped with the STOP command, and can be modified with the MODIFY command.

When starting DSO, you are telling the system what device the system output will go to, the output class that is to be written out on that device, and the classes of jobs that can use the DSO processing that you are starting. You can also use the "keyword=option" parameter in the START command to change the DSO procedure (proclib).

Example: VOL=(,,,volcount,SER=(serno))

volcount
: the number of output volumes that you can use without restarting DSO. The IBM-supplied parameter is 5 volumes; you can change this number with the VOL= keyword when you need a larger number of output volumes.

serno
: the serial number of the first output volume. If this parameter is not used, the system will request a scratch volume.

GOT

Using the MODIFY command you can change DSO processing as the need arises.

Example: If you have DSO processing started for class A, and need it for class C, use the MODIFY command to change DSO processing to class C.

In an MVT system, DSO will appear in a DISPLAY ACTIVE display. In an MFT system, you must keep track of the Direct System Output Processing that you have started, as DSO will not appear in any DISPLAY command displays.

Examples of starting DSO:

MFT - Start DSO in partition 3, for job class N, output class A, and output to tape device 182, enter:

S DSO.P3,182,,(JOBCLASS=N,OUTCLASS=A)

MVT - Start DSO for job class N, output class B, and output onto printer 00E.

S DSO.00E,,(JOBCLASS=N,OUTCLASS=B)

Hardware Debugging Aids

The operating system uses error recovery programs which record hardware bugs in the CPU, main storage, and the channels, and, in most cases, attempt to recover the affected program from these errors. These error recovery programs include the system environment recording (SER0 or SER1), the Machine-Check Handler (MCH), and the Channel-Check Handler (CCH) programs.

SER0 or SER1 can be used on the Models 40, 50, 65, or 75, SER1 can be used on the Models 91 and 195. MCH supports the System/360 Models 65 and 85, and the System/370 Models 155 and 165. The CCH program is provided for any configuration using the 2860, 2870, or 2880 channel which attach to System/360 or System/370 processing units. CCH also supports the System/370 integrated channels on the Model 155.

Always run the system with the check control switch (Models 30 and 50) or the CPU check switch (Models 40, 65, 75, 85, 91, 155, 165, and 195) set to PROCESS. This setting is required by error recovery programs, as well as by a standard, stand-alone program called SEREP (system environment recording edit and print).

When a machine check interruption occurs while the CPU check switch is in process mode, the system gives control to an error recovery management program.

If control goes to SER0, it figures out the type of bug and, if possible, writes out a record describing the error on a data set called SYS1.LOGREC. This data set resides on the primary residence volume.

If SER0 cannot write the record, a message is sent to the primary console (master console in systems with the MCS option) telling you to use SEREP, and then the system is placed in wait state. If the recording is partially or fully successful, a message is sent to the primary console (or master console) telling you to reload the operating system, and then the system is placed in wait state. Either of these messages causes the console alarm to ring if the console is a 1052 Printer-Keyboard or a 2250 Display Console.

In systems with the MCS option, the messages are also sent to the hard copy log if the log is a 1052, 2740, or composite console. After all messages have been sent, the appropriate wait state code is placed in register 0 and the system is placed in wait state.

If control goes to SER1, it also collects and writes out hardware data, but in addition, it attempts to link the error with a particular piece of work being done. If the error can be linked with a particular piece of work and if the control program has not been damaged by the error, that piece of work is terminated abnormally; otherwise, the system goes into a wait state.

Messages issued by SER1 use the same criteria as discussed for SER0. If SER1 does not have to abnormally terminate the whole system, the console alarm is not rung and the system is not placed in wait state.

If MCH receives control, it analyzes the error and attempts a recovery by retrying the failing instruction if possible. If retry is not possible or is unsuccessful, MCH will attempt to repair the malfunction, or isolate it to a task, or both. If the program damage cannot be repaired, the affected job is ended abnormally, and the system operation can continue. If neither retry nor ending a single job is possible, the system is put in the wait state. If possible, MCH formulates an error record, writes it out on the SYS1.LOGREC data set, and informs you of results in a message.

Note: For the Models 85, 155, and 165, machine recovery facilities retry the failing instruction. MCH attempts recovery processing only if retry is not possible or is unsuccessful.

However, if the primary console is an IBM 2250 Display Unit MCH messages are not displayed unless the alternate console is an IBM 1052 Printer-Keyboard. A corresponding wait state code is displayed in the D register.

If the channel check handler gets control when a channel error occurs, CCH analyzes the channel logout information stored in the error recovery procedure information block (ERPIB) for System/360 models, or the extended channel status word (ECSW) for System/370 models. The result of this analysis is used by the I/O supervisor in retrying the failing operation. CCH also records the channel error on SYS1.LOGREC, and writes an operator message reporting the error.

When the SYS1.LOGREC data set has been filled -- by any of the recovery error routines -- print out the data set's contents by using the environment recording edit and print routine, called IFCEREP0. This routine formats and writes the records on SYS1.LOGREC onto printer, tape, or disk according to your installation's specifications.

GOT

IFCEREP0 is described in the publication IBM System/360 Operating System: Utilities, GC28-6586.

To use IFCEREP0 to write out the contents of the SYS1.LOGREC data set and at the same time clear the data set for re-use, you can use a JOB card followed by:

```
//            EXEC   PGM=IFCEREP0
//SERLOG      DD     DSNAME=SYS1.LOGREC,DISP=(OLD,KEEP)
//EREPPT      DD     SYSOUT=A
```

HOW TO USE SEREP

When a message or PSW code indicates that you should run the SEREP program to print out debugging information:

- Load the SEREP deck in the card reader.
- Set the LOAD UNIT switches to the address of the card reader.
- Hit the LOAD button on the system control panel.
- Save the SEREP printout for later use by your customer engineer.
- Reload the operating system.

If you are repeatedly asked to run the SEREP program, call your customer engineer.

Creating Volume Labels

For a description of a program you can use to create standard labels on tapes, see the section on the program IEHINITT in the publication IBM System/360 Operating System: Utilities.

Index

GOT

THE MODEL 65 MULTIPROCESSING SYSTEM

Contents

M65 MP

Figures

The Model 65 Multiprocessing System

The two central processing units (CPUs) of the Model 65 Multiprocessing System operate under one control program as a single system. The control program is MVT with Model 65 multiprocessing, an adaption of the MVT version of the operating system.

When MVT controls a single-CPU system, the CPU processes the highest-priority task that resides in main storage. Under MVT with Model 65 multiprocessing, the CPUs simultaneously process the two highest-priority tasks; and the control program balances the workload among resources (the CPUs, main storage, and I/O devices).

There is flexibility in the way the CPUs operate. An independent subsystem, with its own control program, main storage, and I/O devices, can be set up on each CPU.

Survey of the System

The CPUs are Model 65 Multiprocessors that have been connected so that they share main storage. From two to eight 2365 Model 13 Processor Storage Units, each having 256K bytes, can be attached to a Model 65 Multiprocessing System.

From one to seven channels can be attached to each CPU. No more than two channels can be 2870 Multiplexers, and the remaining channels are 2860 Selectors.

An installation customizes its Model 65 Multiprocessing System. There are two CPUs and as many channels, I/O devices, and storage units as are necessary to meet data-processing needs.

THE OPERATOR AND THE SYSTEM

Many steps you follow when running the Model 65 Multiprocessing System are identical to steps you use when running MVT on any other System/360 CPU. You must be completely familiar with MVT in order to run the Model 65 Multiprocessing System. (See the chapter, "Systems With MVT.") This chapter assumes you know MVT and discusses the differences between running MVT on other System/360 CPUs. and running the Model 65 Multiprocessing System.

When you run any computing system under MVT, you use operator commands to control the processing of tasks. There are additional commands that are used only under MVT with Model 65 multiprocessing that allow you to change the way the hardware interacts. For example, you can set up the Model 65 Multiprocessing System so that the CPUs operate as a single unit, sharing the control program, all of main storage, and most I/O devices. Or, you can set up the system so that the CPUs operate as independent subsystems. When you make such changes, you are reconfiguring the system.

The initial program loader and nucleus initialization procedure (IPL/NIP) for the Model 65 Multiprocessing System issues several operator messages that aid reconfiguration. These messages tell you what storage is offline, which channels are offline, the status of all devices in the system, and the type of system that was initiated. Three types of systems can be initiated with Model 65 multiprocessing. The type of system initiated is specified in one of the following messages:

IEA256I PARTITIONED SYSTEM INITIATED
IEA257I ONE CPU MULTIPROCESSOR INITIATED
IEA258I MULTIPROCESSOR INITIATED

The way you reconfigure each type of system is discussed under "Setting Up the Configuration Control Panel." Other IPL/NIP messages indicate that the system is in a wait state and that certain switch settings must be changed for the initiation of the system to be successful.

RECONFIGURATION OF THE SYSTEM

The way the hardware is physically arranged at your installation is called the configuration of the system. When you select the resources (CPUs, channels, storage, and devices) that the control program is to use on certain jobs, you are performing reconfiguration. To reconfigure the Model 65 Multiprocessing System, use the configuration control panel and the reconfiguration commands.

Configuration Control Panel

As you face the system, look at the frame that connects the two CPUs. You will see a group of dials and switches in the middle of the frame. This is the configuration control panel, and most of the dials and switches you use to perform reconfiguration are located here. A drawing of the configuration control panel is presented in M65MP Figure 1. Refer to the items in the drawing when reading the description of the configuration control panel. Actual manipulation of the controls will be discussed under "Setting Up the Configuration Control Panel."

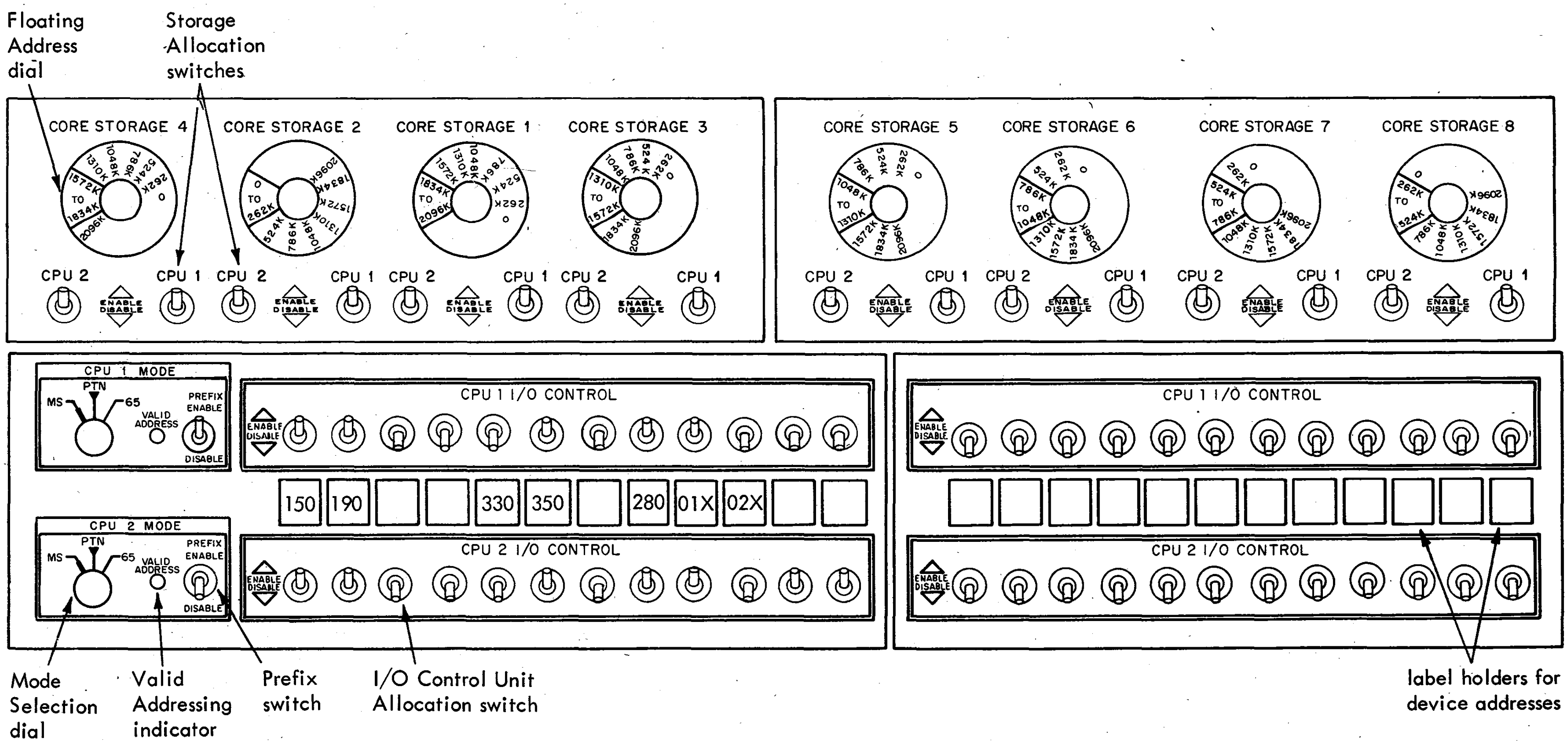
There are dials across the top of the panel called Floating Address dials. Set each dial so that one number appears above and another number appears below the "TO" on the dial. This assigns an address range to a physical storage unit. (A "Core Storage" label above each dial tells you which storage unit that dial is for.) The address ranges you can select are 0K TO 262K, 262K TO 524K, 524K TO 786K, 786K TO 1048K, 1048K TO 1310K, 1310K TO 1572K, 1572K TO 1834K, and 1834K TO 2096K. The last four dials are put on the system control panel only if more than four storage units are configured in the system.

Under the dials is a row of Storage Allocation switches. There are two switches for each Floating Address dial: one switch is for CPU 1 and the other is for CPU 2. Each switch can be set to ENABLE or DISABLE position, which determines whether or not a CPU can use that storage unit.

The three controls that are discussed next -- the Mode Selection dial, the Valid Address indicator, and the Prefix switch -- are grouped for each CPU. The two groups of controls are located on the left-hand side of the panel, one group above the other. The top group is for CPU 1 and the bottom group is for CPU 2.

The Mode Selection dial is turned to one of three positions, MS, PTN, or 65. If both dials are set to MS, the two CPUs are operating as a single system (sharing I/O devices, storage, and the control program). If either dial is set to PTN or 65, each CPU is operating independently of the other CPU. MS and PTN are abbreviations for multisystem and partitioned.

The light next to the Mode Selection dial is the Valid Address indicator. The light must be on or the associated CPU is stopped. The way you set the Floating Address dials and the Storage Allocation switches determines whether the light will go on.



M65MP Figure 1. The Configuration Control Panel

The Prefix switch can be set to ENABLE or to DISABLE position. The position you select determines the area of storage the associated CPU uses to store its hardware status (for example, PSWs) and programming information. This storage area is called the prefixed storage area (PSA). Each CPU must have its own PSA. More on PSA will be presented under "Setting Up the Configuration Control Panel."

The remaining two rows of toggle switches across the middle and bottom of the configuration control panel are for I/O control. These switches are paired -- an upper switch for CPU 1 and a lower switch for CPU 2. Each switch can be set to ENABLE (to allow a CPU to access a control unit or a manual switching unit in the system) or to DISABLE (to prevent a CPU from accessing a control unit or manual switching unit). Between the two rows of I/O controls is a row of label holders. You can put the control unit designation for each pair of switches in the appropriate holder. The configuration control panel shown in Figure 7 was set up for the system to operate as a multiprocessor; that is, both CPUs are in MS mode. Each storage unit that is enabled for CPU 1 is enabled for CPU 2, and the address ranges for all the enabled storage are different. The Valid Address lights are on. The Prefix switch for one CPU is set to ENABLE, while the switch for the other CPU is set to DISABLE. I/O Control Unit Allocation switches are set to ENABLE for both CPUs if the control unit is available to both CPUs. If the control unit is available to one CPU only, the switch for that unit is enabled for the one CPU.

Refer to the Floating Address dials in Figure 7 and note that the address ranges dialed to the storage units are not in ascending or descending order across the panel. This is because the time a CPU needs to access a storage unit is different from the time the same CPU needs to access another storage unit. The address range assigned to each storage unit takes advantage of access speed.

Reconfiguration Commands

Seven commands can be used when performing reconfiguration under MVT with Model 65 multiprocessing. Two of the commands, VARY device and VARY path, are used in MVT; but because there are special restrictions on their use in MVT with Model 65 multiprocessing, they are presented again here. The formats and general information on the use of commands are shown under "Using Reconfiguration Commands."

When ONLINE is specified in the operation field of a VARY command, the resource specified is made available for use by the system. When OFFLINE is specified, the resource is made unavailable. These commands are used for all resources (CPUs, channels, and storage, as well as devices and paths) in the Model 65 Multiprocessing System.

The QUIESCE command tries to let all activity in the Model 65 Multiprocessing System complete normally before stopping the system. The system does not send a message if QUIESCE has been successful, but you can tell when the system has stopped. The Manual light on each of the CPU's system control panel comes on. You should not change controls at the configuration control panel until both Manual lights come on. (Although there are controls elsewhere in the system that may have to be changed, for example, at a 2816 Manual Switching Unit or at a 2860 Selector channel, you should use the QUIESCE command before making the change.)

You do not issue a command when you have finished changing controls. Hit START on the CPU(s) to restart the activity.

Use the DISPLAY M command whenever you want to know the status of system resources (CPUs, channels, storage, and devices). This command is especially useful when you are performing reconfiguration.

When you look at the CPUs in your installation, and when you look at the CPU identifications on the configuration control panel, you will see the labels CPU 1 and CPU 2. When you issue commands, you will use A and B to identify the CPUs. Remember that CPU 1 is CPU A and CPU 2 is CPU B.

Types of Reconfiguration

There are two types of reconfiguration: (1) logical reconfiguration, which occurs when you issue reconfiguration commands; and (2) physical reconfiguration, which occurs when you change the settings of controls. The effect of each type of reconfiguration on the system is different. You must consider the effect when you perform reconfiguration.

In logical reconfiguration, system tables are changed to notify the control program of the number and types of resources available for use. You can perform logical reconfiguration without performing physical reconfiguration.

In physical reconfiguration, resources are connected to or disconnected from the system. Whenever you perform physical reconfiguration, you should perform logical reconfiguration.

Refer back to Figure 7. Although both Mode Selection dials are set to MS, CPU 1 could be idle. A VARY CPUA,OFFLINE command could have been issued and would have made CPU 1 logically unavailable to the system.

Reconfiguration and the Operator

Reconfiguration gives you flexibility in using the resources of the Model 65 Multiprocessing System. External conditions can be met because of this flexibility. For example, to keep up with changes in workload, you can set up one or two computing systems (CPU, channels, storage, and I/O devices) by using the configuration control panel and the reconfiguration commands.

The flexibility provided by reconfiguration also gives you the ability to continue running the system if a noncritical resource fails. For example, several jobs are being processed, and machine failures occur unpredictably in a storage unit (one that does not contain a critical system region). You can issue a VARY command that will remove the storage unit from use and the system can continue running.

How to Reconfigure the System

You reconfigure the Model 65 Multiprocessing System to meet certain conditions, such as the types of jobs that will be run. To handle the workload, you may have to reconfigure the system several times a day. This section defines the types of systems that can be reconfigured and explains the rules you follow when performing reconfiguration.

SETTING UP THE CONFIGURATION CONTROL PANEL

The way storage and I/O devices are reconfigured depends, mainly, on the settings you choose for the Mode Selection dials. If both Mode Selection dials are set to MS, the CPUs share all of main storage and all I/O devices that are enabled for them. One copy of the control program, MVT with Model 65 multiprocessing, coordinates the activity of the CPUs.

If the Mode Selection dials are set to any other combination (MS and PTN, MS and 65, PTN and PTN, PTN and 65, or 65 and 65), you must enable some storage and some I/O devices to each CPU, and you must provide a copy of some control program for each CPU. Use MVT with Model 65 Multiprocessing when a CPU is being run in MS mode or PTN mode; and use MVT, MFT, PCP, or any other control program that can be run on a Model 65 when a CPU is being run in 65 mode. The controls on the configuration control panel, again, are Floating Address dials, Storage Allocation switches, Mode Selection dials, Prefix switches, and I/O Control Unit Allocation switches.

Settings for a Multiprocessor

When conditions require that the system operate as a multiprocessor, turn both Mode Selection dials to MS. Then set up the rest of the panel following these general rules:

1. Set the Floating Address dials so that each enabled storage unit has a different address range. Generally, having contiguous ranges will lessen the possibility of storage fragmentation. However, there are times when having contiguous address ranges is not advantageous; for example, when you plan to add storage units later, leave a 262K-byte address gap between the high and low address ranges for each unit that will be added. Then, when you add storage, you will not have to perform an IPL. You must dial at least two storage units (524K bytes) into the system, and the low address range must be 0K TO 262K.

   The time CPU 1 takes to store data into or to fetch data from a storage unit is not the same as the time CPU 2 takes to perform the same operation; and these two times will be different for every storage unit in the system. How efficiently the system operates depends, partly, on the address range you assign to each storage unit.

   There is a base access time that is the same for all storage units from either CPU. There is also a delay time, which partly depends on the distance the storage unit is from the CPU, that must be added to the base access time to get the total access time. If the system is configured as shown in Figure 8, with the maximum amount of storage available (eight units, equivalent to 2096K bytes), the maximum delay times will be as follows when the Prefix switch for the CPU is set to DISABLE:

Physical Storage Unit	Delay Time per Reference (in ns) from: CPU 1	CPU 2
1	30	65
2	65	30
3	30	100
4	100	30
5	230	230
6	265	265
7	300	300
8	335	335

If the Prefix switch for a CPU is set to ENABLE, the delay time for each access of the PSAs will be longer. (See Item 3, below.)

The addressing established in Figure 7 takes advantage of the different storage access speeds for the configuration shown in M65MP Figure 2. Your branch office representative should provide you with the recommended settings for your system.

2. Set the Storage Allocation switches for CPU 1 and CPU 2 to ENABLE if the storage will be used. Set the switches to DISABLE if the storage will not be used. Don't set the switches for the same unit in opposite positions. Don't set a pair of switches to DISABLE if you plan to use the address range dialed to the storage unit.

3. Set one Prefix switch to ENABLE and the other to DISABLE. The CPU for which the Prefix switch is set to ENABLE will use the PSA established in the storage unit that has the highest address range, while the CPU for which the Prefix switch is set to DISABLE will use the PSA established in the lowest address range. The storage unit that can be accessed fastest by a given CPU should have that CPU's PSA.

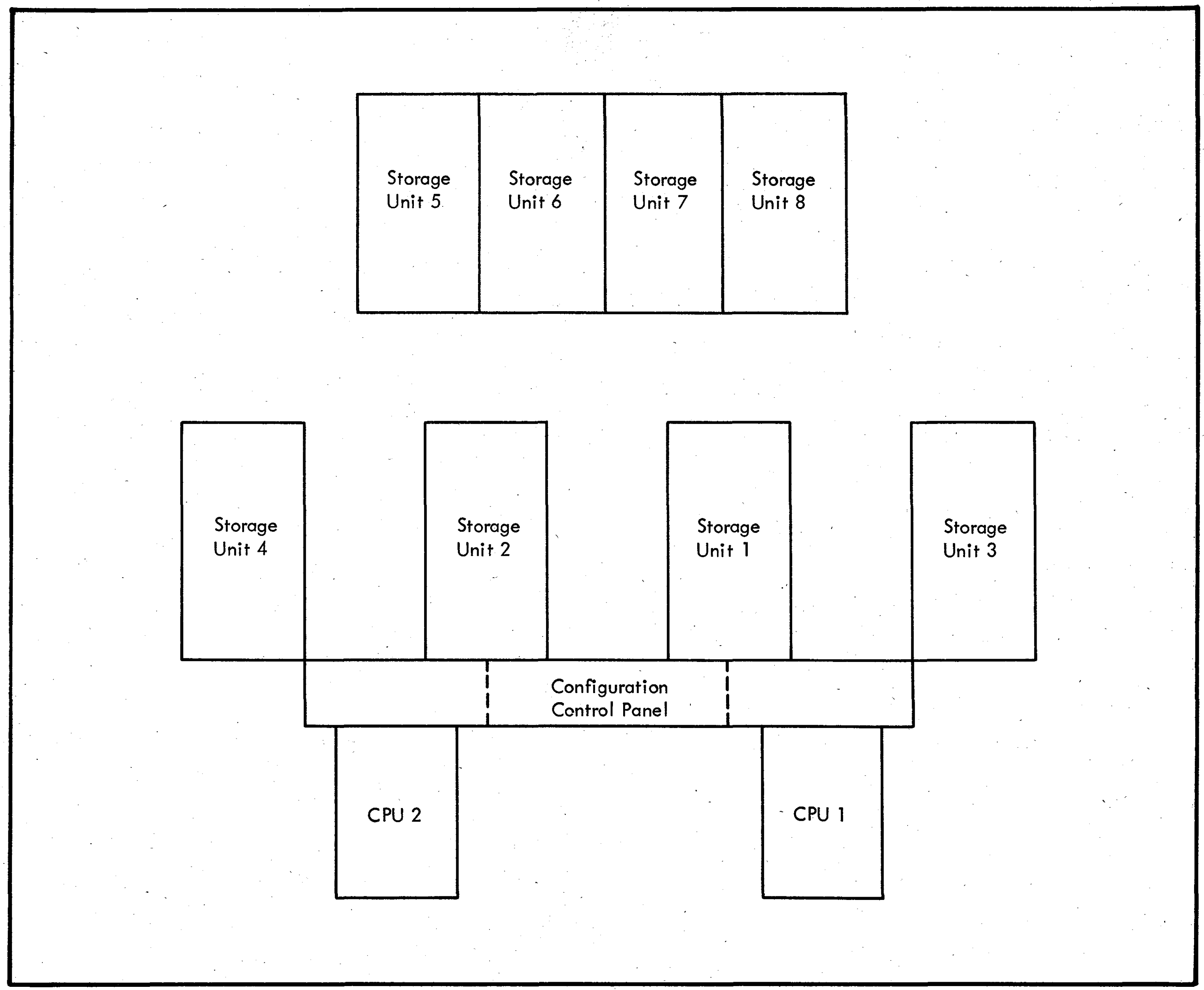
4. Set each pair of I/O Control Unit Allocation switches to ENABLE if the CPUs are to share the devices that are accessed through the control unit. If the CPUs will not share the devices, set the switch to ENABLE for the CPU that will use the devices and to DISABLE for the other CPU. For example, teleprocessing and graphics devices are not shared by both CPUs, even if both I/O Control Unit Allocation switches are set to ENABLE. In this case, such devices are reserved to the CPU you IPL. You may wish to enable the control unit to one CPU and disable the control unit from the other CPU. This physically reserves the device to the enabled CPU. Of course, you can set both switches to DISABLE if neither CPU will use the devices.

Now, look at the Valid Address indicators. If these lights are not on, the CPUs cannot access any storage in the system because two of the storage units that have been enabled for both CPUs have duplicate address ranges. The system will not run until one of the address ranges is changed.

Settings for a One-CPU Multiprocessor

When you turn one Mode Selection dial to MS, a one-CPU multiprocessor can be initiated on that CPU. Follow these general rules:

1. Set the Floating Address dials so that each CPU has a predetermined amount of storage: the CPU in MS mode must have a minimum of 524K bytes of main storage; and if the other CPU is in PTN mode and is running under MVT with Model 65 multiprocessing, it must have a minimum of 524K bytes of main storage. If it is intended that more



M65MP Figure 2. Sample Model 65 Multiprocessing System Configuration storaage will be assigned to the CPU in MS mode later, leave a 262K-byte address gap between the high and low address ranges for each storage unit that will be assigned. If no additional storage will be assigned to the CPU in MS mode, setting address ranges contiguously will help avoid unnecessary storage fragmentation.

2. Set each Storage Allocation switch to ENABLE for the CPU that is to use a storage unit; the corresponding switch for the other CPU must be set to DISABLE. For example, you can dial a 0K TO 262K range for two storage units (0K TO 262K for the storage unit labeled "Core Storage 1" and 0K TO 262K for the storage unit labeled "Core Storage 4"). If CPU 1 is enabled for Core Storage 4, CPU 2 must be disabled from that unit. The reverse is done for Core Storage 1. You must not enable the same storage to both CPUs, but you can disable a storage unit from both CPUs.

3. You can set the Prefix switch for the CPU in MS mode to ENABLE or DISABLE, but it is recommended that you set the Prefix switches for both CPUs to DISABLE whenever the CPUs are operating independently. If you will later run both CPUs in MS mode, it will be necessary to change the setting of one Prefix switch because when both CPUs operate in MS mode, the Prefix switches must be set oppositely. Alter only the Prefix switch of the CPU being put into MS mode so that you will not have to perform an IPL. Leave the Prefix switch for the CPU in MS mode alone or the system will fail.

4. Set the I/O Control Unit Allocation switches to ENABLE for the CPU that is to access the control units and to DISABLE for the other CPU. Do not enable the same control units to both CPUs when one CPU is operating in MS mode and the other CPU is operating in another mode.

Settings for a Partitioned System

When you turn one Mode Selection dial to PTN or to 65, a partitioned system can be initiated on that CPU. Follow these general rules:

1. Set the Floating Address dials so that each CPU has at least the minimum amount of storage required by the control program.

2. Set the Storage Allocation switches so that the CPUs do not share storage.

3. Set each Prefix switch to DISABLE.

4. Set the I/O Control Unit Allocation switches to ENABLE for those control units to be accessed by either CPU. Set the corresponding switch to DISABLE for the other CPU. If you are running Shared DASD (both Mode Selection dials are set to 65), you can set both switches for the shared control unit to ENABLE.

USING RECONFIGURATION COMMANDS

When you reconfigure the system while jobs are running, use the reconfiguration commands in conjunction with changes you make at the configuration control panel. System tables, modified by reconfiguration commands, must reflect changes in the status of hardware so that the system will allocate resources correctly.

The reconfiguration commands are VARY (status of channels, CPUs, devices, paths, and storage) and QUIESCE. You should use one additional command, DISPLAY M, along with VARY and QUIESCE to show the results of reconfiguration. Formats of the reconfiguration commands, in alphabetical order, follow.

DISPLAY M -- Cause Current Display

Use the DISPLAY M command to display the status of system resources (CPUs, channels, devices, and storage). The display can be selective (for example, status of devices on a specific channel) or collective (for example, all resources in the system).

Operation	Operand
{DISPLAY | D}	M [=CPU | =DEV | =n | =STOR | =(sublist)]

CPU
: status of CPUs and channels.

DEV
: status of all devices in the system.

n
: status of all devices on channel n.

STOR
: address ranges of offline storage, in hexadecimal.

(sublist)
: status of resources specified by the operands in parentheses. Operands must be separated by a comma. Do not use blanks within parentheses.

A combined display of CPUs, channels, devices, and storage will be printed if you issue DISPLAY M with no operands.

QUIESCE -- Stop the System

Use the QUIESCE command to stop the system before changing controls at the configuration control panel. This command prevents new I/O operations from starting and waits up to four minutes for I/O activity in progress to complete. If I/O activity stops, both CPUs stop and the Manual lights on the system control panels go on. (If you issue the command to a system in which only one CPU is in MS mode, only that CPU will stop. You can issue the command to a CPU in PTN mode also.)

The QUIESCE command is used in conjunction with VARY, but the sequence of issuing commands is not the same for all operations. When resources are being removed from use by the system, issue the necessary VARY offline commands, wait for the message that says the resource is offline, issue the QUIESCE command, perform the physical reconfiguration, and restart the system. When resources are being returned to use by a system, issue the QUIESCE command, perform the physical reconfiguration, restart the system, issue the necessary VARY online commands, and wait for message verification that the resource is online.

The QUIESCE command may not stop a system that has teleprocessing and graphics devices. I/O operations to these types of devices are often lengthy, exceeding the four-minute time limit. If the QUIESCE command does not stop the system, a message is issued that gives the address or addresses of busy devices:

```
IEE511E  QUIESCE IS NOT POSSIBLE.  BUSY DEVICES
         ARE:  ddd
```

M65MP

Once the QUIESCE has taken effect, don't cause any interruptions. For example, don't cause an I/O interruption (by changing a disk pack) or an external interruption (by pushing the INTERRUPT key).

After you finish changing controls, hit START to restart processing.

Operation	Operand
$\left\{\begin{matrix}\text{QUIESCE}\\ \text{Q}\end{matrix}\right\}$	

VARY -- Change Status of System Resource

Use the VARY command to change the status of channels, CPUs, devices, paths, and storage.

When you plan to use the VARY command to change the status of a CPU or a channel, you must consider the relationship of I/O devices and paths to that CPU or channel. I/O devices are attached to the Model 65 Multiprocessing System either symmetrically or asymmetrically. A device that is attached symmetrically is connected to a control unit to which a pair of corresponding channels (channels that have the same number; one channel from each CPU) is wired. Such a device can be reached by either channel of the pair. A device that is attached asymmetrically is connected to a control unit to which only one of two corresponding channels is wired. To get better performance, do not have permanently resident data sets on devices that are connected asymmetrically.

A path is the logical connection to a device. The address of a path is the combined channel, control unit, and device designations. Corresponding paths from both CPUs have the same address; there can be one or two paths from each CPU to a device. A path can be available or unavailable to the system. An available path is one on which I/O operations can be performed successfully.

Channel: A VARY CHxy,ONLINE command makes the specified channel available to the system if the channel is operational. If the channel cannot be brought online, the system sends a message. For example, if the channel cannot be used by the CPU specified in the command, you will be sent the following message:

IEE500E CHANNEL xy NOT OPERATIONAL

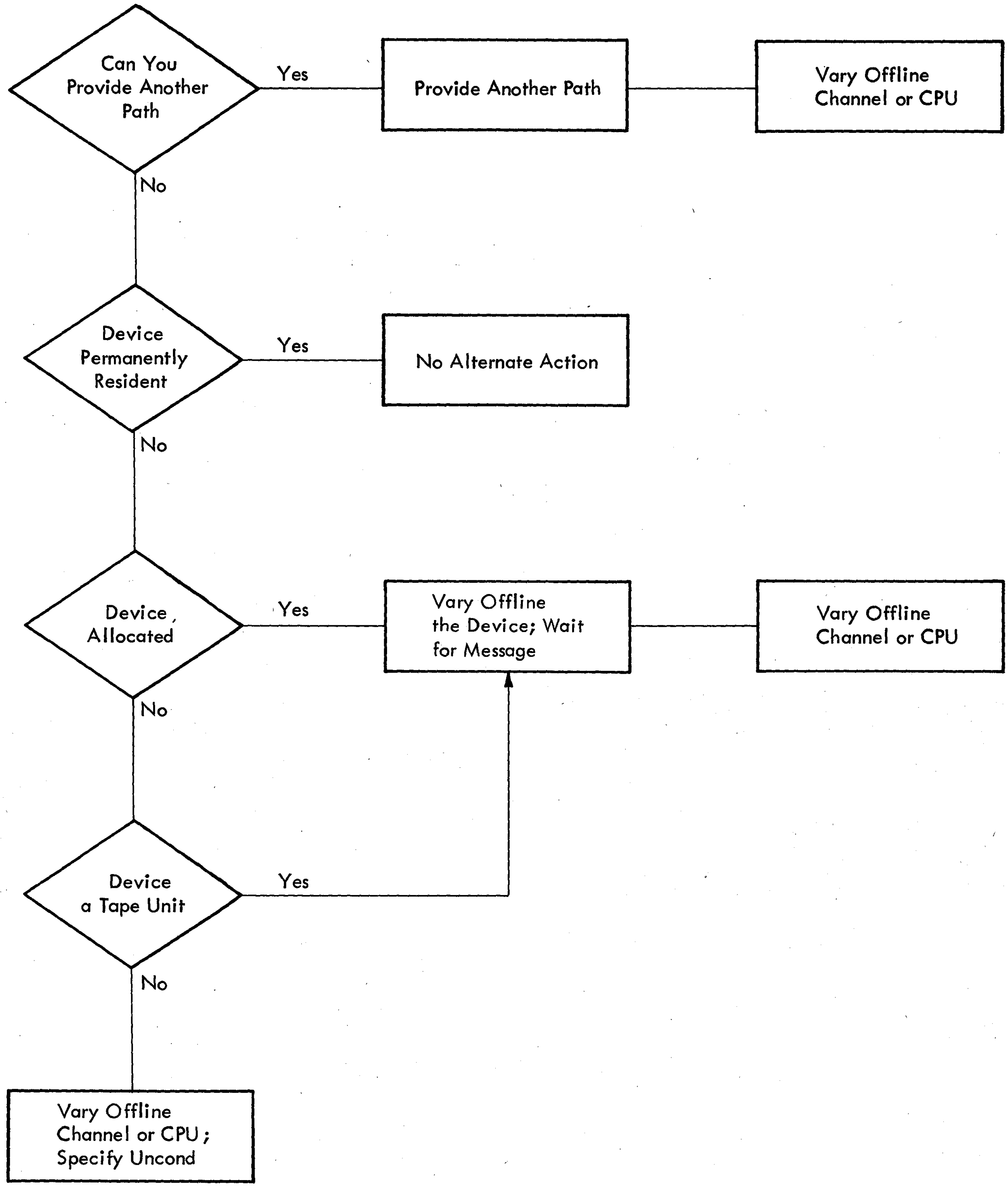
If during execution of the command a channel error is encountered and recovery is not possible, the system sends another message:

IEE509I VARY REJECTED, CATASTROPHIC CHANNEL ERROR

A VARY CHxy,OFFLINE removes the specified channel from use. The command is rejected if the only remaining operator's console is attached to the channel or if the channel makes up the address of the last path to the device. The command can be entered as VARY CHxy,OFFLINE,UNCOND to remove a channel that makes up the address of the last path to a device unless the device is allocated, contains a permanently resident data set, or is an online tape drive. You must specify the UNCOND parameter if the CPU designated in the command is in PTN mode or is the only CPU in MS mode.

If the channel makes up the address of the last path to a device, the safest measure is to provide another path to the device. There are other options available, however: for an allocated device or an online tape drive, take the device offline (you may have to cancel the job), wait for the "device offline" message, and issue a VARY CHxy,OFFLINE (or VARY CHxy,OFFLINE,UNCOND) command for the channel. You have no alternate course of action if the channel makes up the address of the last path to a device that contains a permanently resident data set. M65MP Figure 3. shows alternatives available when you are faced with a "last-path" situation.

You may not be able to remove any channel from system use while teleprocessing or graphics devices are active. I/O operations to these devices are lengthy so the devices are usually allocated.



M65MP Figure 3. "Last-Path" Decision Chart

If you remove the primary (active) console (for example, the 1052 Printer-Keyboard) from the system with a VARY CHxy,OFFLINE command, a later VARY CHxy,ONLINE command for the same channel will return the console to the system. The alternate console that became active when the primary console went offline remains active; and when the "old" primary console is returned online, it becomes the alternate console.

In a system with the multiple console support (MCS) option, a VARY CHxy,OFFLINE command is rejected if the channel is to an asymmetric operator's console (for example, the 1052 Printer-Keyboard). A message giving the device address of the asymmetric console(s) is printed. A VARY CHxy,ONLINE command does not bring operator's consoles online in an MCS system. You must issue a separate VARY command to bring the console online.

Note: Do not hit the INTERRUPT key while the VARY CH commands are being executed.

Operation	Operand
{VARY / V}	CHxy,{ONLINE / OFFLINE}[,UNCOND]

xy
the channel to be made online or offline, where x is the channel number (0, 1, 2, 3, 4, 5, or 6) and y is the CPU (A or B). The CPU must be in MS mode or PTN mode. Do not use blanks in the field.

ONLINE
the channel is to be brought into the system.

OFFLINE
the channel is to be removed from the system.

UNCOND
used with OFFLINE to remove the last path to a device that is not allocated, that does not have a permanently resident data set, or that is not an online tape drive.

CPU: A VARY CPUx,ONLINE command makes the specified CPU available to the system. If the CPU cannot be brought online, the system issues a message to the operator. For example, if a CPU does not respond to an external start from the other CPU, the message is

IEE506E 2ND CPU DID NOT RESPOND TO EXTERNAL START

or if both CPUs are in MS mode and the Prefix switches are set the same, the message is

IEE508E BOTH PREFIX SWITCHES SET THE SAME

Both CPUs (the one specified in the command the one from which the command is issued) must be in MS mode.

A VARY CPUx,OFFLINE command removes the specified CPU from use. The command is rejected if the CPU is the only online CPU in MS mode or has the last path to the device. The command can be entered as VARY CPUx,OFFLINE,UNCOND to remove the CPU that has the last path to a device unless the device is allocated, contains a permanently resident data set, or is an online tape drive.

If the CPU has the last path to a device, first try to provide another path to the device. If this cannot be done, take the device offline. (You may have to cancel the job.) After the "device offline" message is printed, issue a VARY CPUx,OFFLINE (or VARY CPUx,OFFLINE,UNCOND) command.

Before a CPU can be removed from the system, I/O in the system must be complete. There is a three-minute time limit for I/O to complete. Teleprocessing and graphics I/O operations may not finish within the time limit; therefore, when these devices are active, the command may be rejected.

The VARY CPUx,OFFLINE command (with or without the UNCOND parameter) works only for a CPU that is in MS mode.

A VARY CPUx,OFFLINE command will remove an active asymmetric operator's console (for example, a 1052) unless the system has MCS. After the CPU is removed from the system, the alternate console becomes the new active console. If a VARY CPUx,ONLINE command also brings online an operational 1052, that 1052 is made the alternate console.

In an MCS system, the VARY CPUx,OFFLINE command is rejected if the CPU has any active asymmetric operator's consoles. The system prints a message giving the address of the asymmetric console(s). A VARY CPUx,ONLINE command brings devices online, but does not reinstate them as MCS consoles. You must issue a separate command for the device.

M65MP

Note: Do not hit the INTERRUPT key while the VARY CPU commands are being executed.

Operation	Operand
{VARY / V}	CPUx,{ONLINE / OFFLINE}[,UNCOND]

x
    the CPU (A or B) that is to be made online or offline. Use A for CPU 1 and B for CPU 2. The CPU must be in MS mode. Do not use blanks in the field.

ONLINE
    the CPU is to be brought into the system.

OFFLINE
    the CPU is to be removed from the system.

UNCOND
    used with OFFLINE to remove the last path to a device that is not allocated, that does not have a permanently mounted data set, or that is not an online tape drive.

Device: This command is used in MVT as well as in MVT with Model 65 multiprocessing.

Before you issue a VARY ddd,ONLINE command for a device that is asymmetrically connected to the system, make sure the CPU and channel to which the device is attached are online.

The status of the control units cannot be changed, but you can specify all devices attached to the control unit in a VARY ddd,OFFLINE command and create the effect of having a control unit offline.

Operation	Operand
{VARY / V}	(ddd[,ddd]...),{ONLINE / OFFLINE}

ddd
: the unit address of the I/O device(s) to be made online or offline. Parentheses are not needed if you specify one device.

ONLINE
: the device is to be brought into the system.

OFFLINE
: the device is to be removed from the system.

Path: This command is used in MVT as well as in MVT with Model 65 multiprocessing. In MVT, the command can be used only if Alternate Path Retry is specified at system generation. Alternate Path Retry is mandatory in MVT with Model 65 multiprocessing.

A VARY ddd,PATH,dddx,OFFLINE command will be rejected if the specified path is the last available path to a device. A path is considered available only if the channel and CPU that help make up the path address are online.

Operation	Operand
{VARY / V}	ddd,PATH,dddx,{ONLINE / OFFLINE}

ddd
: the primary unit address of an I/O device.

dddx
: channel-control unit-device number and CPU identifier (A or B) of the path.

ONLINE
: the path is to be brought into the system.

OFFLINE
: the path is to be removed from the system.

Storage: A VARY STORx,ONLINE command makes the logical storage unit specified by x available for system use. A VARY STOR(yyyyyy,zzzzzz),ONLINE command makes the storage within the specified address range available. You cannot use the command to bring a range online higher than the highest range online at IPL or to bring a range online that is not dialed into the system.

If storage is being taken offline, all activity in that storage must stop before the command can take effect. There may be a delay between the time the command is issued and the time the message is printed that the storage is offline. If a long job or system task is using the storage, you may have to cancel the operation so reconfiguration can occur. If you issue three or more commands to take storage offline, but three or more of the units or ranges specified cannot be marked offline immediately, the system issues the following message:

IEE342I V REJECTED-TASK BUSY

The VARY STORx,OFFLINE and VARY STOR(yyyyyy,zzzzzz),OFFLINE commands will be rejected if you attempt to remove storage that overlaps the supervisor.

Operation	Operand
{VARY / V}	STOR{x / (yyyyyy,zzzzzz)},{ONLINE / OFFLINE}

M65MP

x
: the number (1, 2, 3, 4, 5, 6, 7, or 8) of the logical storage unit to be made online or offline.

yyyyyy
: the beginning address of storage, in hexadecimal. Use from one to six hex digits. The system rounds the address down to the nearest 2K boundary. Do not use blanks in this field.

zzzzzz
: the ending address of storage, in hexadecimal. Use from one to six hex digits. The system rounds the address up to the nearest 2K boundary. Do not use blanks in this field.

ONLINE
: the storage is to be brought into the system.

OFFLINE
: the storage is to be removed from the system.

RECONFIGURATION EXAMPLES

The following exercises will help you become familiar with the format of commands, the messages the system sends in response to commands, and control changes you must make. Start with a multiprocessing system set up as shown in M65MP Figure 4. (This type of form will help you simplify setting up the configuration control panel. One form should be marked for each type of system you run. IBM does not supply these forms.) How to issue commands and what messages the system sends in response to the commands are shown below.

```
V CPUA,OFFLINE
IEE505I  CPU A OFFLINE
```

The Manual light on the system control panel for CPU 1 turns on.

```
IEE390I  NO ALTERNATE CONSOLE
V CPUZ,ONLINE
IEE522I  CPU Z INVALID
V CPUA,ONLINE
IEE504I  CPU A ONLINE
IEE391I  ALTERNATE CONSOLE IS 01F
V CH1A,OFFLINE
IEE503I  CH OFFLINE
V CH1B,OFFLINE
IEE541E  VARY REJECTED, VARYING OUT LAST PATH TO DEVICE 190
V STOR2,ONLINE
IEE526I  LOGICAL STORAGE UNIT 2 IS NOW ONLINE
V STOR(40000,50000),OFFLINE
IEE510I  STORAGE LOCATIONS 040000 TO 050000 OFFLINE
V STOR(40000,50000),ONLINE
IEE524I  STORAGE LOCATIONS 040000 TO 050000 ONLINE
```

<u>How to Power Down a CPU</u>

```
V CPUB,OFFLINE
IEE505I  CPU B OFFLINE
IEE390I  NO ALTERNATE CONSOLE
Q      (Enter this command on CPU 1.)
```

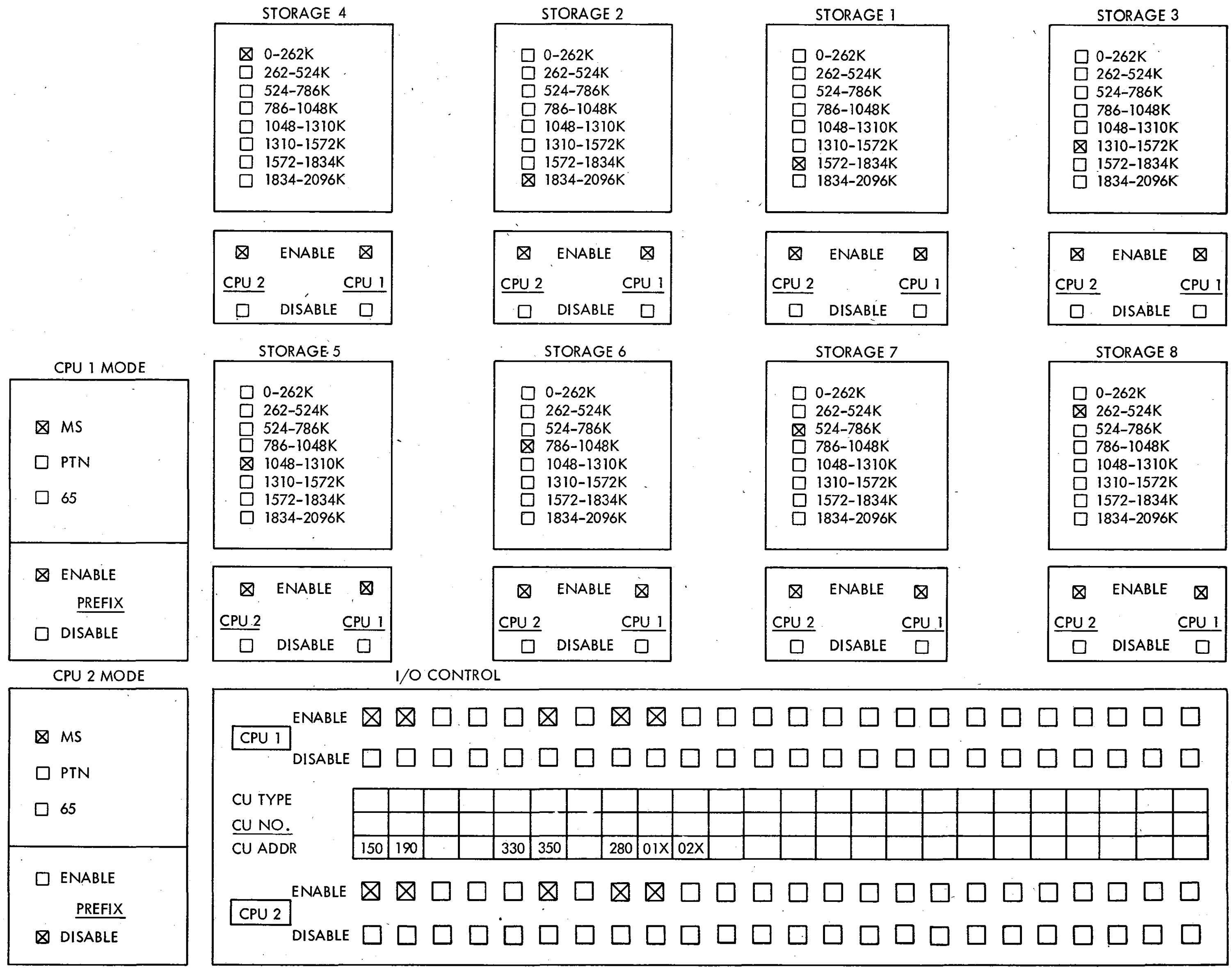
Both Manual lights should be on. Set the Mode Selection dial for CPU 2 to 65 and set the local-remote switches to LOCAL for all channels attached to CPU 2. (This last change prevents a power down of the channels.) Take power down on CPU 2 then hit START on CPU 1.

<u>How to Power Up a CPU</u>

```
Q      (Enter this command on CPU 1.)
```

The Manual light for CPU 1 turns on. Bring up power on CPU 2. Set the Mode Selection dial for CPU 2 to MS. Hit START on CPU 1.

```
V CPUB,ONLINE
IEE504I  CPU B ONLINE
IEE391I  ALTERNATE CONSOLE IS 009
```



M65MP Figure 4. Sample Form for Setting the Configuration Control Panel

How to Power Down a Channel

```
V CH0B,OFFLINE
IEE503I  CH 0B OFFLINE
Q
```

Both Manual lights should be on. At the configuration control panel, set the Control Unit Allocation switches for control units attached to Channel 0B to DISABLE for CPU 2. Put the channel in Test mode. Hit START on both CPUs.

How to Power Up a Channel

```
Q
```

Both Manual lights should turn on. Restore the channel to Auto mode and enable all control units to CPU B for Channel 0B. Hit START on both CPUs.

```
V CH0B,ONLINE
IEE502I  CH 0B ONLINE
```

How to Split the System

The next example shows the steps you might use to reconfigure the system set up as shown in Figure 9.2. The result will be two systems, one with a CPU in MS mode and one with a CPU in PTN mode. Reconfiguration will take place while the initial system is operating. The I/O devices that will be made available to the CPU in PTN mode are one reader-punch (019-01A), one printer (018), eight 2314s (150-157), one 2311 (190), eight tapes (280-287) and one 1052 (01F).

When logically reconfiguring a system, you can enter as many commands as you wish. The reconfiguration will not have been completed until you get the proper messages. The underlined items are system messages that have been included in this example to show continuing activity.

```
IEF403I  JOB000 STARTED
IEF403I  JOB001 STARTED
V STOR7,OFFLINE
V STOR6,OFFLINE
IEE525I  LOGICAL STORAGE UNIT 6 IS NOW OFFLINE
V STOR5,OFFLINE
IEE525I  LOGICAL STORAGE UNIT 5 IS NOW OFFLINE
V STOR2,OFFLINE
```

When you split the system, knowing it will be put back together, remove storage units that are between the high-address and low-address units for use in the subsystem. Because the high-address and low-address units contain the control program, you must not reconfigure them.

```
V (018,019,01A),OFFLINE
V (150,151,152,153,154,155,156,157,190),OFFLINE
V (280,281,282,283,284,285,286,287),OFFLINE
IEE303I  150 OFFLINE
IEE303I  155 OFFLINE
IEE303I  156 OFFLINE
IEF404I  JOB001 ENDED
IEF403I  JOB002 STARTED
IEF281I  DEVICE 281 NOW OFF-LINE JOB002
IEF281I  DEVICE 282 NOW OFF-LINE JOB002
IEF404I  JOB000 ENDED
IEF403I  JOB003 STARTED
IEF281I  DEVICE 152 OFF-LINE JOB003
IEF281I  DEVICE 280 OFF-LINE JOB003
IEE525I  LOGICAL STORAGE UNIT 7 IS NOW OFFLINE
D A
```

The system responds to this DISPLAY request with the message IEE102I. (The MVT commands, except MODE, can be used in MVT with Model 65 multiprocessing.) As jobs are initiated and terminated, resources are freed and are marked offline. If resources do not become free, you may have to cancel a job so that reconfiguration can occur.

M65MP

```
V CPUA,OFFLINE
```

This command also removes the 1052 (01F) and the channels (0,1,2) attached to CPU A.

```
IEE505I  CPU A OFFLINE
IEE390I  NO ALTERNATE CONSOLE
Q      (Enter this command on CPU 2.)
```

When both Manual lights are on, you are ready to reconfigure the system physically. Make the following changes at the configuration control panel.

- Turn the Mode Selection dial for CPU 1 to PTN.
- Set the Prefix switch for CPU 1 to DISABLE.
- Set the Storage Allocation switches for "Core Storage" 1, 3, 5, and 8, to DISABLE for CPU 2.
- Set the Storage Allocation switches for "Core Storage" 2, 4, 6, and 7 to DISABLE for CPU 1.
- Turn the Floating Address dials for "Core Storage" 1, 3, 5, and 8 to the desired address ranges (0K TO 262K, 786K TO 1048K, 524K TO 786K, and 262K TO 524K, respectively).
- Set the Control Unit Allocation switches for CPU 2 to DISABLE if CPU 1 will use the control unit, and set the Control Unit Allocation switches for CPU 1 to DISABLE if CPU 2 will use the unit.
- Hit START on CPU 2.
- Set the Load Unit dials on the system control panel for CPU 1 to the address of the SYSRES device.
- Hit LOAD on CPU 1.

After IPL is complete, you can check the status of resources in the partitioned system by entering the DISPLAY M command on the 1052 attached to CPU 1.

```
D M
IEE070I  CPU AND CHANNEL STATUS
IEE070I  AB01234567
IEE070I  P.AAA.....

IEE071I  STATUS OF DEVICES
IEE071I    0123456789ABCDEF
IEE071I  01 ........AAA....A
IEE071I  15 AAAAAAAA .......
IEE071I  19 A...............
IEE071I  28 AAAAAAAA........

IEE072I  NO STORAGE OFFLINE
```

You may also use the DISPLAY M command at this point to check the status of the resources left in the MS system.

How to Restore a CPU to Multisystem Mode

When the work to be done on CPU 1 has completed, you may have to return the CPU to MS mode. You would do the following:

- Stop CPU 1.
- Q   (Enter this command on CPU 2.)
- Set the Mode Selection dial for CPU 1 to MS.
- Set the Prefix switch for CPU 1 to ENABLE.
- Set the Storage Allocation switches for "Core Storage" 1, 3, 5, and 8, to ENABLE for CPU 2.
- Set the Storage Allocation switches for "Core Storage" 2, 4, 6, and 7 to ENABLE for CPU 1.
- Turn the Floating Address dials for "Core Storage" 1, 3, 5, and 8 to 1572K TO 1834K, 1310K TO 1572K, 1048K TO 1310K, and 262K TO 524K, respectively.
- Set the Control Unit Allocation Switches for CPU 1 and CPU 2 to ENABLE if both CPUs share the control unit.
- Hit START on CPU 2.
- V CPUA, ONLINE
- IEE391I ALTERNATE CONSOLE IS 01F

The last command also brings online (1) operational channels attached to the CPU, except channels that were taken offline by the VARY CH command; and (2) devices, except direct access devices that are not online.

How to Run the System

You can run most jobs under MVT with Model 65 multiprocessing that were run under MVT. Shared DASD and Main Storage Hierarchy Support are not available to a system that is running under MVT with Model 65 multiprocessing.

The procedures operators use in running the Model 65 Multiprocessing System can affect the system's efficiency and availability. A few recommended procedures for running under MVT with Model 65 multiprocessing follow.

STARTING, STOPPING, AND RESTARTING THE SYSTEM

After the configuration control panel has been set up and the address of the SYSRES device selected, hit LOAD. If both CPUs are in MS mode, the CPU on which you perform IPL will electronically start the other CPU. If the message

IEA257I ONE CPU MULTIPROCESSOR INITIATED

is printed, one of the following could be the problem:

1. The Rate switch on one system control panel is set to INSN STEP. Set the Rate switch to PROCESS. You can re-IPL or issue a VARY CPUx,ONLINE command for the CPU that was not initiated to get the second CPU into the system.
2. One CPU has no storage. Set the Storage Allocation switches for that CPU to ENABLE. You can re-IPL or issue a VARY CPUx,ONLINE command for the CPU that was not initiated to get the second CPU into the system.
3. The prefixed CPU's PSA is in a storage unit that is accessed too slowly. Select a faster-access storage unit for the high address range. Re-IPL.

If you cannot get the second CPU into the system, try to IPL that CPU. The IPL may be successful; but if it is not, you will get a message or the LOAD light will stay on. If the LOAD light stays on, check the settings of the Floating Address dials and the Storage Allocation switches.

As in a uniprocessing MVT system, an IPL failure may be caused by an I/O problem. Display the contents of General Register 10 on the system control panel of the CPU you IPL to get the address of the last I/O device used by the program. There are other causes of IPL failures. Also, display the current PSW for a wait state code.

After one type of IPL failure, you get neither a device address nor a wait state code. When you are using the SYS1.DUMP data set, the tape drive on which the data set is located must be available to the CPU you IPL. The system will stop immediately after your reply to the message

IEA135A SPECIFY SYS1.DUMP TAPE UNIT ADDRESS OR NO

if you specify a unit address and the device is not available to the CPU you performed IPL on. You must make the tape device available to the CPU you IPL and re-IPL or you must re-IPL and reply NO to the message.

During nucleus initialization, an I/O device that is accessible to the CPU you IPL must be selected for messages. This device is called the primary, active console (or master console in a system with MCS).

Usually, the same device will be used as the primary, active console. You must make sure that the device is ready and online to the CPU you IPL. If a primary, active console (or master console) cannot be found, nucleus initialization ends and the system enters a wait state. You must make a console available to the CPU and re-IPL, or you can perform the IPL procedure on the other CPU.

After the primary, active console has been selected, the system attempts to find an alternate console from among all available consoles. If an alternate is found, a message is printed that gives the address of the device. If none is found, the "no alternate console" message is printed. In a system that has MCS, a message that gives the status of all consoles is printed.

If you want to make the alternate console the primary console, hit the INTERRUPT button on one system control panel. In a non-MCS system, this console switch will be reversed if you hit INTERRUPT a second time. In an MCS system, the new primary will be switched with the new alternate each time you hit INTERRUPT until either (1) you run out of consoles or (2) you get back to the original primary-alternate pair (because the chain of consoles is a loop). Never hit INTERRUPT while the system is quiesced or while the VARY CPU or VARY CH commands are being executed.

Although it doesn't matter which CPU you IPL, you might consider performing IPL from the CPU that has prefixing disabled. If the operator's console on that CPU becomes the active, primary console, system communication could be slightly faster.

The QUIESCE command can be used to stop a system that is running under MVT with Model 65 multiprocessing. To restart a two-CPU multiprocessing system that has been quiesced, hit START on both CPUs. To restart a one-CPU system, hit START on that CPU.

MODIFYING THE IPL PROCEDURE

If your installation's Model 65 Multiprocessing System has the Rollout-Rollin option, you can restrict an important job to the storage units that contain the control program (the highest and lowest address ranges). By doing this, you ensure that whenever the system is running the important job is running. Also, you can split the system while the important job is running and save an IPL.

There are three requirements for using this modified IPL procedure: (1) job control language must specify that the critical job can cause rollout; (2) there must be at least 786K bytes (three units) of main storage available; and (3) the critical job must be divided into two parts, each of which will fit into the remaining storage of the first two storage units online, if all of the job will not fit into one unit. The procedure is as follows:

- Have the storage units with the lowest and highest address ranges online (for example, if you are using three storage units, "Core Storage" 1 is 0K TO 262K and "Core Storage" 3 is 524K TO 786K).

- IPL and override the automatic START commands by specifying AUTO=NNN.

```
S RDR,ddd      (This command starts to bring in the job.)
P RDR,ddd      (This command frees reader storage.)
```

If the critical job was split, bring in the first part of the job by issuing

S INIT

The first part of the job will get the second part (by a LINK, LOAD, or XCTL that will cause a rollout and will obtain a second region in the other online storage unit). The initiator will cause the rest of the job to be put into storage. Then, assign an address range between the high and low address ranges online to another storage unit (for example, dial the 262K TO 524K address range for "Core Storage" 2).

V STOR2,ONLINE

You can bring in other jobs that will occupy the remaining available storage (S WTR, S RDR, S INIT). If the system must be split, a V STOR2,OFFLINE command will remove storage that is not being used by the critical job.

TAKING A STAND-ALONE DUMP

There are several ways of taking a stand-alone dump on this multiprocessing system. In one method:

- Set both Prefix switches to DISABLE.
- Set both CPU Mode Selection dials to PTN.
- Disable both timers.
- Record the address range of each storage unit.
- Split the storage units between the CPUs, setting the address ranges of both sets of units contiguously, beginning at 0K.
- Load the stand-alone dump program on each CPU.
- Keep a record of which dump came from which CPU.

When using the following alternate method of taking a stand-alone dump on the system, do not change either Prefix switch.

- Hit STOP on both CPUs.
- Record the address range of each storage unit.
- Set the Storage Allocation dials to contiguous ranges (0K TO 262K, 262K TO 524K, etc.), if necessary.
- Set the Mode Selection dial for the CPU on which the dump is being taken to 65.
- Disable the timer on the CPU on which the dump is being taken.
- Hit LOGOUT then STOP on the CPU on which the dump is <u>not</u> being taken.
- Load the dump program.

RUNNING WITH TELEPROCESSING AND GRAPHICS DEVICES

Most teleprocessing and graphics devices are physically attached to both CPUs through a control unit that has a two-processor switch. When the switch is in NEUTRAL position, the control unit is reserved to the first CPU that issues a TEST I/O instruction and finds the devices available. You can reserve devices for a specific CPU by setting the I/O Control Unit Allocation switch at the configuration control panel to ENABLE for the CPU and to DISABLE for the other CPU. You can use this technique to balance teleprocessing or graphics devices between the two CPUs.

When two CPUs are operating in MS mode, a re-IPL must be performed if a teleprocessing control unit is to be switched from the reserving CPU to the other CPU. When CPUs are operating independently, the teleprocessing control unit must be assigned to the desired CPU by manual switching, usually at the configuration control panel.

As was mentioned in the section that gives the formats of QUIESCE and VARY, the commands may not take effect while teleprocessing and graphics devices are active. If the commands do not take effect, you will have to cancel the job in order to perform reconfiguration.

Index

M65 MP

REMOTE JOB ENTRY AND CONVERSATIONAL REMOTE JOB ENTRY

Contents

RJE/ CRJE

Remote Job Entry and Conversational Remote Job Entry

The Remote Job Entry (RJE) or the Conversational Remote Job Entry (CRJE) facility of the operating system provide a way of entering jobs submitted from remote work stations into the job stream. This allows a remote user the same batch computing facility that is available at the central installation.

RJE or CRJE provide a means for efficient operation of computing facilities by equipment centralization, and also gives substantial computing power on a demand basis to locations not requiring it on a regular basis.

In addition, it allows sharing of a common body of information within a company by widely separated departments having related needs.

RJE provides fast turnaround of computer requirements for people in all parts of a company by placing the computer facilities close to the source of input with high speed communication lines.

CRJE provides for more than one user at a time, a means of obtaining access to the data processing facilities of a central computing installation from a remote terminal. Programs and data can be entered at terminals, stored in the central system, and updated as necessary before being entered into the job stream for normal batch processing at the central installation.

General Concepts

Remote job entry or conversational remote job entry control a flow of data and processes that data as required. Data entering from remote sources is the primary input to the RJE or CRJE system.

Jobs submitted by remote users are passed to the operating system for scheduling and execution. When the output resulting from these jobs becomes available, it is returned to the user as requested -- either immediately (RJE only) or on command (RJE and CRJE).

RJE or CRJE operate in conjunction with MFT and MVT as a system task, much like a combined reader and output writer. Jobs received from the work station are scheduled for subsequent execution.

When a remotely submitted job is completed, the job output is placed in a common SYSOUT class for RJE or CRJE. RJE or CRJE removes the output from this class and returns it to the work stations.

The CRJE system uses a keyboard terminal for input and a printing device for output. The CRJE user and the system communicate with each other like two people talking on the telephone. The user is aware at all times what the system is doing with his job. CRJE provides a means of creating and manipulating data sets, creating job streams a line at a time, and of entering these jobs from the remote terminals into a job stream to be processed in the background. The user can insert, replace, delete or change information submitted.

Using commands, you can supervise the central system and communicate with remote users. The central commands are:

1. BRDCST - Maintain information in the broadcast data set.
2. CENOUT - Cause output from remotely submitted jobs to be written locally.
3. MODIFY - Cause a CRJE communication line to be activated or deactivated.
4. MSG - Send a message to a work station.
5. SHOW - Display information pertaining to RJE.
6. START - Begin RJE operation at the central installation.
7. STOP - Cease RJE operation at the central installation.
8. USERID - Add users to or delete users from the system.

Startup and Closedown

When the central RJE or CRJE task is in operation, remote work stations may begin and end RJE or CRJE activities at will. When the central RJE or CRJE task ceases operation, all work stations are closed down.

CENTRAL STARTUP AND CLOSEDOWN

Central startup and closedown are achieved by the START and STOP commands. When startup is initiated by the START command, you will get a message indicating that the RJE or CRJE task is in operation. When closedown is initiated by the STOP command, you will get a message indicating completion of the RJE or CRJE task.

System Overload

An overload condition results if direct access storage space at the central installation is insufficient to meet the demands of the RJE or CRJE system.

Input already received and acknowledged by RJE or CRJE is not affected by an overload condition. Any input transmission causing an overload condition is ended and must be wholly resubmitted at a later time.

In each overload situation (RJE only), a message is sent to both you and the work station operator indicating the particular resource depleted. If the system continually overloads, the direct access storage space allotted for the resources must be increased to reflect the peak traffic requirements of the system more realistically.

An alternate solution may be to reschedule the work load to take advantage of periods of relative inactivity.

The total system input capacity for RJE or CRJE is specified by the central installation and is dependent on the following resources:

- The quantity of SYS1.SYSJOBQE space -- specified at system generation.
- The number of concurrent jobs RJE or CRJE is to maintain -- specified at RJE or CRJE assembly by your system programmer.
- The quantity of space for remotely submitted SYSIN data -- specified in an RJE or CRJE cataloged procedure referred to in the START command for RJE.

SYS1.SYSJOBQE depletion results when job input submitted both locally and from attached work stations exceeds the limit specified by the central installation. If this condition continues to occur, the size of the SYS1.SYSJOBQE must be increased to reflect the requirements of the system -- both local and remote. This requires that SYS1.SYSJOBQE be scratched and reallocated.

An overload condition also occurs when the number of remote jobs resident in the central system exceeds the limit specified when the RJE or CRJE program was assembled.

Remote jobs have residence until the output is removed from the RJE or CRJE SYSOUT class. This condition is relieved by requesting the output of completed remote jobs in the system. You have this facility with the CENOUT command. If the condition continues to occur, the number of remote jobs RJE or CRJE can maintain must be increased. This requires an RJE or CRJE assembly by your system programmer.

Depletion of SYSIN space is the final cause of a system overload. In its cataloged RJE or CRJE procedure, the installation specifies SYSIN data sets on a communication line basis. In this procedure, the installation specifies the direct access type, the volume serial number to be used for the SYSIN data sets for this line, the blocking factor for SYSIN data sets, and the maximum space available for any one input data set.

Specifying the maximum amount of space allocated for one input data set prevents one job from getting all the SYSIN space. This is a system protection feature, and no special action is necessary if a job exceeds this limit.

On the other hand, a regular depletion of the total SYSIN allocation necessitates a new cataloged procedure, referred to at START RJE or START CRJE time, which makes more SYSIN space available to the system.

Operator Awareness

A message is displayed whenever an unrecoverable communication error occurs. In addition, error counts for each line are displayed.

The system keeps for each line an error count for data check, for intervention required, and for non-text time-out, and also records the number of transmissions occurring on the line.

The installation can specify threshold values for these counters when the RJE or CRJE support is assembled.

If any one of the three error counters reaches its threshold count before the transmission threshold count is reached, a message is displayed. This message identifies the line and gives all three error counts and the transmission count. After the message is displayed, the error counts and the transmission count are added to accumulators, and the counters reset to zero.

You can display the value of the accumulators at any time with the SHOW LERB command. The accumulators for one line or for every support line can be requested. These accumulators are reset to zero each time the central system starts up.

Note: If the transmission count reaches its threshold count before any error count, the counters are added to the accumulators and reset to zero. No message is displayed.

RJE OR CRJE RESTART

RJE or CRJE must be restarted whenever the operating system is restarted because of an unrecoverable error. The procedure followed to restart RJE or CRJE at the central system includes three steps.

1. The condition causing the error is corrected.

2. The operator re-IPLs the system.

3. The operator issues the START command for RJE or CRJE.

These three steps restart the RJE or the CRJE support in the central system. RJE or CRJE provides the necessary information to the work stations to insure that no information is lost.

RJE and CRJE Commands

You communicate with the RJE or the CRJE system through RJE or CRJE commands. These commands provide the additional capabilities needed to control and maintain the RJE or the CRJE application and to communicate with RJE or CRJE users and work stations. If the RJE or the CRJE task is not active when the commands are issued, the commands are not accepted and a message is issued.

The restrictions imposed on format and placement of these commands are identical to those for JCL command statements; that is, the commands are introduced from the keyboard-printer or the system input device.

When entered from the system input device, the commands contain the JCL identifier (//) in the first two positions of the command statement.

The commands available to the central RJE or the CRJE operator provide a number of capabilities.

1. Commands used to control the RJE or the CRJE application:

   START
   STOP

2. Commands used to maintain the RJE or the CRJE application:

   MODIFY (CRJE only)
   USERID
   CENOUT
   SHOW

3. Commands used to communicate in the RJE or the CRJE system:

   MSG
   BRDCST

An RJE or CRJE command statement cannot be continued. It must be coded on one card or card image. No abbreviation of these commands (except START, STOP, and MODIFY) is allowed.

RJE or CRJE will buffer central commands, up to 100 (defined in the assembly of RJE or CRJE), so that the operator can have more than one command pending at a time. All buffered commands will be processed.

BRDCST -- Maintain the RJE or CRJE Broadcast Messages

Use the BRDCST command to maintain broadcast messages which are sent to work stations on request and when the stations become active. The messages tell, for instance, when the central station is closing down and when a central resource is added or deleted.

Up to 100 of these messages are kept in a data set, on a direct access device. Each message is numbered to correspond with a numbered slot. These slots are numbered from 00 to 99 for RJE and from 0 to 9999 for CRJE and they are either active (containing a message), or inactive (containing no message).

The BRDCST command lets you:

- Insert a new message.
- Add a new message.
- Change an existing message.
- Remove an existing message.
- Collect active messages into the low-numbered slots.
- Clear the data set of all existing messages.

You can get a copy of the active broadcast messages through the SHOW BRDCST command.

Enter message text in upper case. The text must be enclosed in apostrophes and can include up to 40 printable characters and blanks. Apostrophes used as text must be paired; each pair counts as one text character.

Operation	Operand
BRDCST	[c,] { nn,'text' | 'text' | nn | DELETE | Ann,'text' | PACK }

c
: indicates that this command refers to CRJE. This operand is optional and is required only when CRJE and RJE are operating concurrently.

nn,'text' (RJE)
: the text is placed into slot number 'nn' (00-99).

nnnn,'text' (CRJE)
: the message consisting of the specified text is to be entered as the broadcast message with the identifier specified by nnnn (0-9999). If there is already a message with the specified identifier, it is replaced.

'text' (RJE)
: the text is placed into the lowest numbered inactive slot. If no slots are inactive, the command is refused.

'text' (CRJE)
: the message consisting of the specified text is to be added following the last existing message. The identifier for the new message is obtained by adding 10 to the identifier of the last existing message.

nn (RJE)
the text is deleted from slot number 'nn'.

nnnn (CRJE)
the message identified by nnnn (0-9999) is to be deleted.

DELETE (RJE and CRJE)
requests the deletion of all broadcast messages.

Ann,'text' (RJE)
the text is placed into slot number 'nn' after moving sequentially the texts of that slot and of all consecutive active slots up into higher numbered inactive slots. This preserves the contents and sequential order of all original texts. If there is no high inactive slot to receive the pushed up texts (or if slot 99 is specified), this command is refused. This is useful if you want to insert a message between two existing messages.

PACK (RJE)
specifies the collection of all active slots at the 00 end of the directory and of all inactive slots at the 99 end. This preserves the contents and sequential order of all original texts. Use this function to recover when an insert command (see previous paragraph) is refused because of no higher inactive slot.

CENOUT -- Give RJE Output to Local Output Writers

Use the CENOUT command to remove job output in the RJE or CRJE SYSOUT class and process it with the central installation output writers.

This command lets you retrieve output of completed remotely-submitted jobs which cannot be transmitted or has not been requested by an RJE or CRJE user.

The RJE or CRJE system places the output data sets and system messages in the originally-specified SYSOUT class. The disposition of the output is the same as that of any other data for the SYSOUT class at the central installation.

Only output of completed jobs is placed in the originally-specified SYSOUT class. Jobs completing after the CENOUT command has been processed remain in the RJE or CRJE SYSOUT class.

Operation	Operand
CENOUT	[C,]J=jobname,C=class

C
: indicates that this command refers to CRJE. This operand is optional and is required only when CRJE and RJE are operating concurrently.

J=jobname
: the name of the remote job whose output is to be handled by the central installation output writers. If this job is not complete or is not in the RJE or the CRJE system, or is already queued for delivery, the command is rejected and a message is issued.

C=class
: the SYSOUT class in which the job output is to be placed (use for CRJE only).

MODIFY -- Activate or Deactivate a CRJE Communication Line

Use the MODIFY command to activate or to deactivate a CRJE communication line. When you activate a line it becomes operative, when you deactivate a line it suspends the CRJE use of the line.

The following is the result of deactivating a CRJE communications line while a user is active at the terminal.

- A write operation will be completed and the user automatically logged off.
- A read operation from the communications line is halted and the user is automatically logged off.
- A user entering data at a terminal is not notified, and is automatically logged off. If the terminal is a 2741 waiting for the user to enter data, the users active data set, if one exists, is saved and he is notified of the action taken by the CRJE system.

Operation	Operand
$\left\{\begin{array}{c}\text{MODIFY}\\ \text{F}\end{array}\right\}$	[procname.]identifier,$\left\{\begin{array}{c}\text{D}\\ \text{A}\end{array}\right\}$=(address,...)

procname
: the name of the cataloged procedure for CRJE operation. This procedure name is the same as that specified in the START command.

identifier
: the identifier assigned to the CRJE task with the START command.

D=(address,...)
: the address of the communication line to be deactivated.

A=(address...)
: the address of the communication line to be activated.

MSG -- Communicate with RJE or CRJE Users

Use the MSG command to send messages to users and work stations. You can selectively route a message to:

- A specific user currently logged on.
- A specific work station.
- A specific user or, if the user is not logged on, a specific station.
- All work stations logically attached to the RJE or CRJE system.

In addition, the MSG command lets you remove from the system messages whose transmission is waiting for a work station startup. This option is normally used either when communication between the central installation and a work station is not possible due to some failure, or when the data set containing pending messages becomes full.

Operation	Operand

MSG	```[C,]{M='text'[,U=userid[,Q]][,T=termid]}```
	 ```{D={termid}```
	 ```{ {userid}```

C
: indicates that this command refers to CRJE. This operand is optional and is required when CRJE and RJE are operating concurrently.

M='text' (RJE and CRJE)
: (RJE and CRJE) the message text. Text must be framed with apostrophes, can include as many as 40 printable characters and blanks for RJE only, text must be upper-case. Apostrophes included as part of the text must be paired; each pair counts as one text character. <u>If you omit the U and T keyword parameters for RJE and the U parameter for CRJE, the message is sent to all active work stations</u>.

U=userid (RJE)
: the message is sent to the user identified by three alphameric characters (the first must be alphabetic) -- for example, U2T or T23. The message is sent if the user is logged on. If he is not logged on and the T parameter is omitted, the message is not sent. (You will get a message indicating this.) If both T and U parameters are specified and the user is not logged on, the message is either sent to the work station or held until work station startup.

U=userid[,Q] (CRJE)
: the message is sent to the user identified by the userid (1 to 7 alphameric characters). If the Q parameter is specified and the user is inactive, the message will be delivered when the user next logs on the system. If Q is not specified, the message is delivered only if the user is active.

T=termid (RJE)
: the message is sent to the work station identified by one to eight alphameric characters, the first of which must be alphabetic -- for example, RALEIGH or TERM23. If the work station is inactive, the message waits for it to start up.

D=termid (RJE)

the deletion of pending messages for the work station identified by termid. To obtain a copy of these messages issue a SHOW MSGS,termid command before entering the MSG D=termid command.

D=userid (CRJE)

deletes all delayed messages for the user identified by the userid (1-7 alphameric characters). This userid need not be that of the current CRJE terminal user, allowing the central operator the capability of deleting a user from the system and then deleting that user's messages. A copy of these messages can be obtained with a SHOW MSGS, userid command before entering the MSG D=userid command.

SHOW -- Display RJE or CRJE Information

Use the SHOW command to request a console display of RJE or CRJE information.

Specify only one parameter for each command. If you want more than one type of information, use a separate SHOW command for each type.

Operation	Operand
SHOW	[c,] { JOBS[,jobname] TERMS[,termid] USERS[,userid] DEFER[,userid] ACTIVE[,number] BRDCST MSGS[,termid][,userid] LERB[,linename][,lineaddress] SESS[,userid] SESSREL[,userid] }

C
: indicates that this command refers to CRJE. This operand is optional and is required only when CRJE and RJE are operating concurrently.

JOBS
: requests a list of the RJE or RJE jobs in the central system and the status of these jobs. The response for each job includes the jobname, the identifier for the user who submitted the job (userid), the completion status (complete or incomplete), and output disposition (immediate or deferred). If the output is deferred, the response also includes the number of normal central closedowns since the job was received and, if there is an alternate user, his userid.

JOBS,jobname
: requests the status of a particular RJE or CRJE job.

TERMS
: requests a list of the RJE work stations. The response for RJE includes the RJE work station identifier (termid) and the state of the work station -- inactive, active or processing. If the work station is active, the unitaddr of the communication line being used is included. If the work station is processing, the unitaddr and the userid of the user logged on are included.

TERMS,termid
: requests the state of a particular RJE work station.

USERS
: requests a list of all RJE or RJE users. The response includes the userid and protection key and indicates whether or not the user is logged on. If user has previously logged on, no matter how much time has passed since the logging on and off, the termid of that work station is included. If the user has not logged on before, the termid is omitted. If the user is currently logged on, the termid and the unitaddr of the communication line being used are included.

The response for CRJE includes an indication of whether each user is active. For active users, the time each has been logged on the system and the address of the line he is on, are also returned.

USERS,userid
: requests the status of a particular RJE or CRJE user.

DEFER
: requests a list of all RJE jobs which have deferred output. The response includes the jobname, the identifier for the user who submitted the job (userid), the alternate user's identifier (if available), completion status (complete or incomplete), the output disposition, and the number of normal central closedowns since the job was received.

DEFER,userid
: requests a list of those RJE jobs with deferred output for which the specified user is a valid recipient.

ACTIVE (RJE)
: requests a list of all RJE work stations in the active or processing state, plus a list of the users logged on the processing work stations. The response includes the RJE work station identifier, the state of the work station, and the unitaddr of the communication line being used. If a work station is processing, the userid of user currently logged on is also included.

  For CRJE requests a list of currently active users, including the line each is on and the time each has been logged on the system.

ACTIVE,NUMBER (CRJE)
: Requests a display of the number of currently active CRJE users.

BRDCST
: requests a copy of the current broadcast messages.

RJE CRJE

MSGS
: requests a copy of all messages whose transmission is waiting for a work station to start up. The work station to which each message is directed is indicated in the list.

MSGS,termid (RJE)
: requests a copy of messages pending for a particular work station. If termid does not correspond to a work station in the system, the command is rejected.

MSGS,userid (CRJE)
: Requests a copy of all the delayed messages for the user identified by the userid displayed, which need not be that of a current user. This permits the central operator to see if there are any delayed messages remaining in the system for a user that has been deleted from the system.

LERB
: specifies a request for the error and transmission counts for all communication lines. The list indicates for each line the three error counter values (data check, non-text time-out, and intervention required) and the transmission counter values. These values are cumulative values since the last RJE or CRJE central startup.

  Note: For CRJE if the multiple console support (MCS) option of the operating system is present, all SHOW LERB command responses are sent to the primary central console, regardless of the location of the issuing console.

LERB,linename (RJE)
: specifies a request for the error and transmission counts for a particular line. The linename is that specified when the central RJE program was assembled. You will receive a message containing the three error counter values (data check, non-text time-out, and intervention required) and the transmission counter value for the line designated.

LERB,lineaddress (CRJE)
: requests the error and transmission counts for the specified line. The lineaddress is the address that was specified for the line when the CRJE system was assembled.

SESS (CRJE)
: requests notification when all CRJE users log on and off the system.

SESS,userid (CRJE)
: requests notification when the specified user logs on and off the system.

SESSREL (CRJE)
: specifies that no notification is given when users log on and off the system. If you are running with Multiple Console Support (MCS) and a SHOW SESS command is in effect, a SHOW SESSREL command must be issued before CRJE closedown when the operating system is not to be restarted before the next CRJE startup. If a SHOW SESSREL command is not issued, the SHOW SESS command facility will not work properly.

SESSREL,userid (CRJE)
: specifies that notification for the user identified by the userid is terminated.

Note: If a SHOW SESS command is currently in effect, a SHOW SESS or SHOW SESSREL request for a specific userid is not accepted for processing by CRJE.

START -- Start RJE or CRJE System Process

Use the START command to start operation of RJE at the central installation.

Operation	Operand
{START S}	procname[.identifier],,,({FORM NFMT NONE}{,ABNO ,NORM})

Note: The two commas following the delimiting comma after procname are required to indicate the absence of the parameters devicename and volumeserial.

procname
: the name of the cataloged procedure for remote job entry or conversational remote job entry operation. This procedure name must begin with characters RJE or CRJE.

identifier
: the RJE or CRJE task identifier. When RJE or CRJE is run under MVT, this parameter is optional. Under MFT, it must be specified as P0. In addition, P0 must be defined as a resident reader partition, which may be done either at system generation or with a DEFINE command prior to starting RJE or CRJE. To use the DEFINE command:

  Operator enters: DEFINE

  System responds: id IEE802A ENTER DEFINITION

  Operator enters: REPLY id,'P0=RDR,END'

FORM
: RJE or CRJE is to start from scratch. Code this form only if the operating system has been loaded with the following IPL option since RJE or CRJE closed down: in the SET command, 'F' was specified in the Q keyword subparameter list. 'F' indicates that the job queue data set is to be formatted during IPL. The operating system formatting of the job queue data set deletes all jobs in the system, including those submitted remotely. The FORM parameter removes all references to jobs in the RJE or CRJE job table. If FORM is specified, all jobs within the RJE System are deleted regardless of the operating system startup.

NFMT
: RJE or CRJE is to restart. Code NMFT only if the operating system has been loaded since RJE or CRJE closed down and the sub-parameter 'F' of the keyword Q in the SET command has not been specified.

  If you have reloaded the operating system more than once since RJE or CRJE closed down, and, during any of the IPL procedures, specified 'F' in the SET command, then code FORM on the START statement for RJE or CRJE.

  If you specified Q=([unitaddr] [,F]) in the SET command, the RJE or CRJE job table will retain its references to the jobs deleted by reformatting the job queue data set. These references can be removed only by specifying FORM.

If you have not loaded the operating system since RJE or CRJE closed down and you specify NFMT, no job output existing prior to closedown can be retrieved during this execution of RJE or CRJE. To recover the output, reload (do not specify 'F' in the SET command) and start RJE or CRJE specifying NFMT.

NONE

specified when the operating system has not been loaded since RJE or CRJE closed down. If you specify None on the START statement and have loaded the operating system since RJE or CRJE down, all jobs within the RJE or CRJE System are deleted. In addition, if you specified 'F' in the SET command during the IPL, the RJE CRJE job table will retain its references to the deleted jobs. Any attempts to access the deleted jobs will cause unpredictable table results. The RJE or CRJE job table references can be freed only by specifying FORM.

ABNO

CRJE is to execute its active area recovery procedure.

NORM

CRJE is to execute its normal active area startup procedures. This operand must be specified if the active area has been reallocated since the last CRJE closedown.

STOP -- Stop RJE or CRJE System Process

Use STOP command to stop operation of RJE or CRJE at the central installation.

Operation	Operand
{STOP / P}	{procname / identifier}

procname
: the name of the cataloged procedure for remote job entry or conversational remote job entry operations. This procedure name is the same as that specified in the START command.

identifier
: the RJE or CRJE task identifier. If you used an identifier in the START command, use the identifier to STOP (procname is unnecessary).

USERID -- Modify the RJE User Directory

Use the USERID command to modify and maintain the RJE user directory. The USERID command lets you:

- Add a userid/key pair to the user directory.
- Remove a userid/key pair from the user directory.

These facilities are provided dynamically and don't require a reassembly of the RJE program.

Each userid in the directory must be unique.

If you submit a command to add a userid that is currently in the user directory, RJE rejects the command and returns a message indicating such.

When a userid is removed from the directory, all jobs currently in the system associated with that userid are also removed.

Operation	Operand
USERID	userid,key,{ADD / DELETE}

userid
: the userid, identified by three alphameric characters (the first must be alphabetic -- for example, U2T or T23), be added to or removed from the user directory.

key
: the protection key, which consists of three alphameric characters (the first must be alphabetic -- for example, T23 or U51), assigned to the userid. The same key may be assigned to several userids.

ADD
: the userid and key be added to the user directory. If no space is available in the user directory, the RJE program must be reassembled to increase the size of the user directory.

DELETE
: the userid and key to be deleted from the user directory.

USERID -- Control the Availability of the CRJE System

Use the USERID command to control the availability of the CRJE system to terminal users. Users can be added to and deleted from the list of authorized users of the system. Also, initiation of new sessions can be suspended and resumed.

Operation	Operand
USERID	[C,] { A[DD] =(userid,password) D[ELETE] S[UPPRESS] R[ESUME] }

C
: indicates that this command refers to CRJE. This operand is optional and is required only when CRJE and RJE are operating concurrently.

A[DD]=(userid,password)
: the user with the specified userid and password is to be added to the list of authorized users. The userid consists of 1-7 and the password consists of 1-8 alphameric characters, the first of which must be alphabetic.

D[ELETE]=(userid,password)
: the user with the specified userid and password is to be deleted from the list of authorized users. A user cannot be deleted while he is active.

S[UPPRESS]
: users not currently logged on the system are not allowed to log on. On switched lines, calls will not be answered. On nonswitched lines, LOGON commands are rejected and the terminal user receives a message indicating that sessions are suppressed.

R[ESUME]
: users are allowed to log on the system. This operand nullifies the effect of a previously entered USERID SUPPRESS command.

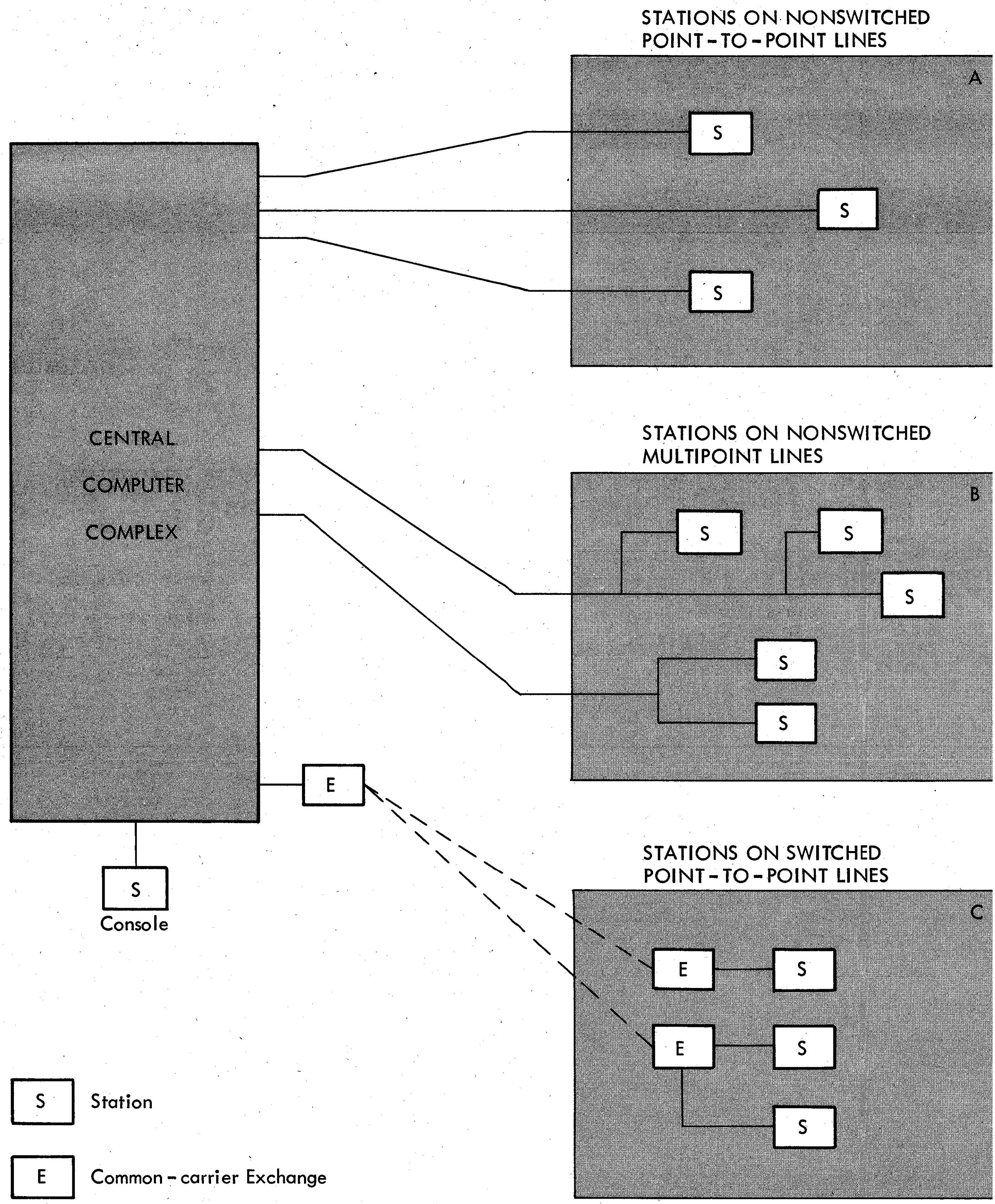
Note: If a new userid is added to the system while CRJE is running, the terminal user identified by the userid will not have access to his user library, even if a DD card for his library was included at system startup time.

Index

TELECOMMUNICATIONS ACCESS METHOD

Contents

TCAM



• Example Telecommunications Network

Telecommunications Access Method

The Telecommunications Access Method (TCAM) is an access method used for communication between terminals and a central computer complex. The time sharing option (TSO) of the operating system uses TCAM as its access method. Other telecommunications applications may also use TCAM. TCAM can be used with either the MFT or MVT configurations of the control program.

With operator commands, you can display and modify current TCAM status information.

General Concepts

To use TCAM commands effectively, you need to understand the structure of a telecommunications network. An example of such a telecommunications network is shown on page TCAM 2.

In a telecommunications network several stations are connected to the central computer complex by communications lines. Stations can be either terminals or other computers. Communications lines are simply referred to as lines.

In TCAM there are three types of lines:

- Nonswitched point-to-point.
- Nonswitched multipoint.
- Switched point-to-point.

TCAM

A nonswitched point-to-point line connects the central computer complex to a single station. Each station has its own line. (A on page TCAM 2.)

A nonswitched multipoint line connects the central computer complex to several stations. Stations on the same nonswitched multipoint line can be in the same remote location or different remote locations. (B on page TCAM 2.)

A switched point-to-point line connects a station to the central computer complex only during the time it is in use. The central computer complex and the station are connected to a common-carrier exchange (such as a telephone exchange). When the station is in use, it is connected to the central computer complex by a common-carrier communications line, such as a telephone line or a microwave channel. (C on page TCAM 2.)

All the lines of one type in a telecommunications network can be classified as a line group. Each line group in the network is given a name when TCAM is set up. This name is used in some commands to refer to the group of lines. Each line within a group is also assigned a relative line number. The combination of the line name for the line group and the relative line number within the group may be used to identify a specific line. For example, if PPTRM is the name of three nonswitched point-to-point lines in a network and you want to specify the line assigned relative line number 3, you can refer to the line in many TCAM commands as PPTRM,3 instead of using the channel unit address of the line.

All stations are given names when TCAM is set up. The names of all stations in your network can be obtained by using the DISPLAY command.

Your console is also a station. Its name is SYSCON. It is different from all other stations in the network because it cannot be used for data entry.

When TCAM is set up, several stations can be designated as operator control stations, which means that they are eligible to enter TCAM commands. Normally, your console is designated as the primary operator control station, which makes it eligible to receive error recovery procedure messages. All other operator control stations are known as secondary operator control stations. The primary operator control station can be reassigned to any secondary operator control station by using the MODIFY command.

Option fields (named fields which specify station options) are also set up for each station in the network when TCAM is set up. These fields are referenced in operator commands using their field names. Changes to option fields will be specified by your systems programmer.

STATION AND LINE STATUS

Although stations and lines may remain physically connected to the telecommunications network, the message transmission status of a station or a line may be changed.

Stations and lines are said to be active if they are capable of transmitting messages to or receiving messages from the central computer complex. Inactive stations and lines are those on which all message transmission has been stopped.

Message transmission in either direction from a station may also be stopped. A station is said to be intercepted when it may enter messages into the telecommunications network, but may not receive any messages. Stopping message transmission from a station while the station is still capable of receiving messages is also possible. So, a station can be active, inactive, or intercepted.

POLLING (AUTOMATIC AND PROGRAMMED)

Polling is a procedure used to coordinate messages being sent from many different stations at the same time. The systems checks each active station in turn to see if there are any messages to be transmitted.

Automatic polling is a hardware feature in which the hardware checks each station in turn for new messages.

Programmed polling is under control of a program. It may also be used when automatic polling is unavailable or to override automatic polling. The MODIFY command may be used to change a station from automatic polling to programmed polling.

The order in which stations are polled is determined by their position in a list which is called an invitation list. It is called this because the polling procedure -- be it hardware or program -- invites the station to transmit messages.

Programmed polling sometimes has a variable interval of time set for each station to indicate how often that station should be checked for new data. The MODIFY command may be used to change this interval for a station.

THE SYSTEM INTERVAL

The system interval is the duration of programmed delay which allows users on switched point-to-point lines to make connection to the central computer complex.

During this delay, all polling stops. When this delay is called for by the MODIFY command, the system may enter a wait state for the specified time, if there are no other jobs running in the system. TCAM displays a message at the beginning of the programmed delay that tells you what is happening:

IED080I START OF TCAM SYSTEM DELAY

After the specified time, TCAM will resume normal processing with a message stating that the delay has ended:

IED081I END OF TCAM SYSTEM DELAY

Startup and Closedown

You can start TCAM by entering the START command, if TCAM is a cataloged procedure on SYS1.PROCLIB, or by entering the TCAM procedure through a system input stream. Closedown is accomplished by the HALT command.

Immediately following startup, you may receive a message requesting that TCAM parameters be specified:

IED002A SPECIFY TCAM PARAMETERS

The response to this request should be specified by your systems programmer.

Closedown can be specified in one of two modes -- quick or flush -- in quick mode closedown is effective immediately and in flush mode all pending messages are sent before the system closes down.

Operator Awareness

A message is displayed whenever an unrecoverable input/output error occurs. Three types of error message are possible.

The message:

IED008I TCAM OPEN ERROR

If an error occurs on a communication line, TCAM sends you the message IEA000I which includes the channel unit address for the line and other related diagnostic information.

It is also possible for a disk threshold error to occur, in which case TCAM displays a message and closes down. The message:

IED076I TCAM NON-REUSABLE DISK THRESHOLD CLOSEDOWN

TCAM Commands

The TCAM commands are described in this section. You use these commands to start and stop TCAM, display and modify status information, and interrupt or resume message transmission to and from stations.

TCAM commands are entered from the system console, a system input device, or any station designated in the TCAM system as an operator control station.

When entered from a system input device, the commands contain the JCL identifier (//) in the first two positions of the command statement.

TCAM commands provide you with a number of capabilities. All can be entered in abbreviated form.

- Commands used to control the TCAM process:

  HALT Z
  START S

- Commands used to display and modify control information:

  DISPLAY D
  MODIFY F

- Commands used to interrupt and resume message transmission to and from stations:

  HOLD H
  RELEASE A
  VARY V

TCAM command statements cannot be continued on a second line or a second card.

TCAM will buffer up to 255 control commands so that you can have more than one command pending at a time.

The TCAM commands are explained in detail and listed in alphabetical order.

DISPLAY -- Cause Current Display

Use the DISPLAY command to cause a current display of:

- The names of all stations on a line.
- The name of a line group, the relative line number of a line within that line group, and the machine address of that line.
- The names of all inactive stations on a line.
- The names of all intercepted stations in the TCAM network.
- The status field and error record for a line.
- An indication as to whether the invitation list for a line may be polled, and if so, whether automatic polling is being used.
- Any field in the option table for a station.
- The name of the current primary operator control station.
- Queue status information for a station.
- The names of all current secondary operator control stations.
- The station status and input/output sequence numbers for a station.

Some of the information provided by the DISPLAY command is for systems programmer use, such as a field in the option table for a station. However, much of the information made available by this command is of use to you. For example, you can determine the status of operating stations -- whether they are active, inactive, or intercepted. You can also determine the names of all stations from which TCAM commands can be entered.

TCAM

```
Operation   | Operand
------------+------------------------------------------------
 {DISPLAY}  |     {ACT,lineaddress                        }
 {   D    } |     {INACT,lineaddress                      }
            |     {LINE,lineaddress                       }
            |     {LIST,lineaddress                       }
            |     {ADDR,stationname                       }
            | TP,{QUEUE,stationname                       }
            |     {TERM,stationname                       }
            |     {OPTION,stationname,optionfieldname     }
            |     {INTER                                  }
            |     {PRITERM                                }
            |     {SECTERM                                }
```

TP
indicates that this command is used with TCAM.

ACT
the names of all active stations on the line addressed are to be displayed.

INACT
the names of all inactive stations on the line addressed are to be displayed.

LINE
the status field and error record for the line are to be displayed.

LIST
    the system is to indicate whether the invitation list for the line may be polled and whether the auto poll feature is being used to poll the list.

lineaddress
    the specific line for which information is to be displayed. This may be entered in the form of a channel unit address or as name,rln where name is the name of the line group and rln is the relative line number within the line group.

ADDR
    the system is to display for the station named:

- The name of the line group of which it is a part.
- The relative line number of the line on which the station is located.
- The machine address of the line on which the station is located.

QUEUE
    the queue control block for the station is to be displayed. This information includes the number of messages queued, the queue status, and the priority levels permitted for messages.

TERM
    the station status and input/output sequence numbers for the station are to be displayed.

OPTION
    the contents of the field that is reserved in the option table for the station are to be displayed.

stationname
    the specific station for which information is to be displayed. This field may be from one to eight characters long beginning with an alphabetic character.

optionfieldname
    the specific option field for which information is to be displayed. This field may be from one to eight characters long beginning with an alphabetic character.

INTER
    the names of all stations in the network that are intercepted are to be displayed. An intercepted station is one which may enter messages, but to which transmission of message is suspended.

PRITERM
    the name of the current primary operator control station is tc be displayed.

SECTERM
    the names of all secondary operator control stations are to be displayed.

HALT -- Stop TCAM

Use the HALT command to stop TCAM in either quick or flush closedown mode.

Operation	Operand
{HALT Z}	TP,{QUICK FLUSH}

TP
: indicates that this command is used with TCAM.

QUICK
: the message traffic is to stop on each line as soon as transmission of any message currently being sent or received on the line is completed. Messages remaining in the system are sent to the appropriate destinations after TCAM is restarted.

FLUSH
: message transmission from stations is to stop on each line as soon as transmission of any message currently being sent is completed. Then, all message tranmissions to stations are completed before the system is halted. Intercepted messages which cannot be sent to stations are sent to the appropriate destinations after TCAM is restarted.

TCAM

HOLD -- Intercept a Station

Use the HOLD command to intercept a station. When a station is intercepted, no messages may be sent to it. However, messages may be entered from the station.

Operation	Operand
{HOLD / H}	TP=stationname

TP
    indicates that this command is used with TCAM.

stationname
    the name of the station to be intercepted. This field may be one to eight characters long beginning with an alphabetic character.

MODIFY -- Change Process Characteristics

Use the MODIFY command to change these characteristics of TCAM:

- Switch the polling method of a nonswitched line from automatic to programmed or from programmed to automatic.
- Cause records of the number of recoverable I/O errors to be kept.
- Change the polling interval for the whole system or for a single line within the system.
- Change the primary operator control station designation.
- Change the contents of a field in the option table for a station.
- Start or stop the TCAM I/O trace facility for a line.

Some of the changes that can be made by the MODIFY command should be made only by a systems programmer, such as changing the contents of a field in the option table. However, many of the changes possible are useful to you. For example, you may wish to keep records of recoverable I/O errors to diagnose possible system difficulties.

```
Operation   | Operand
------------+-----------------------------------------------------------
{MODIFY}    | /[procname.] identifier                                  \
{  F    }   | |                                                        |
            | |  AUTOPOLL=lineaddress{,ON }                            |
            | |                      {,OFF}                            |
            | |                                                        |
            | |  INTENSE={LINE,lineaddress } ,sense[,sensecount]       |
            | |          {TERM,stationname }                           |
            | |                                                        |
            | <  INTERVAL={SYSTEM[,value]             }                >
            | |           {POLL,stationname,seconds   }                |
            | |                                                        |
            | |  OPERATOR={stationname}                                |
            | |           {SYSCON     }                                |
            | |                                                        |
            | |  OPT=stationname,optionfieldname,data                  |
            | |                                                        |
            | |  TRACE=lineaddress{,ON }                               |
            | \                   {,OFF}                               /
```

procname
: the name of the TCAM cataloged procedure in SYS1.PROCLIB.

identifier
: the TCAM identifier used in the START command or the name of the job to start TCAM in the system input stream.

AUTOPOLL=lineaddress,ON or OFF
: the polling method for the line is to be changed from automatic to programmed or from programmed to automatic. Lineaddress, which specifies the line to be changed, may be entered in the form of a channel unit address or as name,rln, where name is the name of the name of the line group and rln is the relative line number within the line group. ON specifies a change from programmed to automatic polling. OFF specifies a change from automatic to programmed polling.

INTENSE=LINE,lineaddress

records are to be made of recoverable I/O errors occurring on the line specified by lineaddress. Lineaddress may be entered in the form of a channel unit address or as name,rln, where name is the name of the line group and rln is the relative line number within the line group.

INTENSE=TERM,stationame

records are to be made of recoverable I/O errors occurring on the station specified by stationname. Stationname may be one to eight characters long beginning with an alphabetic character.

sense

the code specifying the type of errors to be recorded:

BO	busout check
CR	command reject
DC	data check
EC	equipment check
IM	general intensive mode
IR	intervention required
LD	lost data
M2	leading graphics for 2740 Model 2 terminal
OR	overrun
TO	timeout exceeded
UE	unit exception

sensecount

the number of times error recording is to take place. The value is limited to 1 to 15. One digit values may be entered as a single digit or with one leading zero. The default value is 15.

INTERVAL=SYSTEM

the system is to enter a programmed delay for the duration of the system interval.

INTERVAL=SYSTEM,value

the duration of the system interval is to be changed. The value field is used to replace the system interval with a decimal number of seconds not to exceed 65,535. Digits may be left or right adjusted (with leading zeros) in the five-digit field.

INTERVAL=POLL,stationname,seconds

the polling interval of a line group is to be changed. Stationname is the name (one to eight characters beginning with an alphabetic character) of any station associated with the line to be changed. Seconds is used to replace the polling interval with a decimal number of seconds not to exceed 255. Digits may be left or right adjusted (with leading zeros) in the three-digit field.

OPERATOR=stationname

the secondary operator control station specified by stationname (one to eight characters beginning with an alphabetic character) is to be changed to the primary operator control station.

OPERATOR=SYSCON

the system console (specified by SYSCON) is to be made the primary operator control station.

OPT=stationname,optionfieldname,data
the contents of the operation field for a station are to be changed. Stationname (one to eight characters beginning with an alphabetic character) identifies the station. Optionfieldname (one to eight characters beginning with an alphabetic character) specifies the option field to be changed. Data is the data tc be inserted in the option field.

TRACE=lineaddress,ON or OFF
the TCAM I/O trace facility for a line is either to be activated or deactivated. Lineaddress specifies the line and may be entered in the form of a channel unit address or as name,rln, where name is the name of the line group and rln, where name is the name of the line group and rln is the relative line number within the group.

TCAM

RELEASE -- Release an Intercepted Station

Use the RELEASE command to release an intercepted station so that messages cn be transmitted to that station.

Operation	Operand
{RELEASE / A}	TP=stationname

TP
    indicates that this command is used with TCAM.

stationname
    the name of the station to be released. This field may be frcm one to eight characters long beginning with an alphabetic character.

START -- Start TCAM

Use the START command to start operation of TCAM.

TCAM is a cataloged procedure on SYS1.PROCLIB. TCAM is an IBM-supplied name, but your systems programmer can give TCAM a user name.

Operation	Operand
{START / S}	procname[.identifier]

procname
: the name of the cataloged procedure in SYS1.PROCLIB for TCAM. The name can be either an IBM-assigned name or a user-assigned name provided by your systems programmer.

identifier
: an optional name of up to eight characters, the first of which must be alphabetic. This name identifies the TCAM task.

TCAM

VARY -- Vary Status of Lines or Stations

Use the VARY command to activate or deactivate lines and stations in the following manner:

- Stop two-way transmission for a station.
- Stop a station from entering messages only.
- Start two-way transmission for a station.
- Start a station entering messages only.
- Stop the transmission on a line immediately.
- Stop the transmission on a line after all current messages are completed.
- Start transmission on a line or line group.

```
┌────────────┬─────────────────────────────────────────────────┐
│ Operation  │ Operand                                         │
├────────────┼─────────────────────────────────────────────────┤
│  {VARY}    │ /                {ONTP    }  \                  │
│  { V  }    │ | lineaddress,   {OFFTP,C }  |                  │
│            │ |                {OFFTP,I }  |                  │
│            │ {                            }                  │
│            │ |                {ONTP,B  }  |                  │
│            │ | stationname,   {ONTP,E  }  |                  │
│            │ |                {OFFTP,B }  |                  │
│            │ \                {OFFTP,E }  /                  │
└────────────┴─────────────────────────────────────────────────┘
```

lineaddress
: the name of the line or line group which is to be activated or deactivated. This field may be entered in the form of a channel unit address or as name,rln, where name is the name of the line group and rln is the relative line number within the group. For this command, lineaddress may indicate either a single line or a whole line group. Transmission of messages can be restarted (or started initially), stopped after current messages are completed, or stopped immediately. The parameters specifying the type of action to be taken are:

  ONTP -- restart (or start initially) transmission.

  OFFTP,C -- stop transmission after current messages are completed.

  OFFTP,I -- stop transmission immediately.

stationname
: the name of the station which is to be activated or deactivated. This may be from one to eight characters long beginning with an alphabetic character. Only nonswithced stations may be specified. In all cases transmission on the line with which the station is associated must be stopped before a VARY command to activate or deactivate a station is given. The line must be restarted after the desired change in the station's status is specified. The stopping and starting of transmission on a line is achieved by the VARY command option in which lineaddress is specified. Stations can be activated and deactivated in one of two ways:

  - Two-way message transmission can be started or stopped.
  - Only the entering of messages can be started or stopped.

The parameters specifying the type of action to be taken are:

ONTP,B -- start, entering and accepting

ONTP,E -- start, entering only

OFFTP,B -- stop, entering and accepting

OFFTP,E -- stop, entering only

In all cases, when entering of information is stopped, any message currently in process is completed.

TCAM

TCAM Operating Techniques

This section describes procedures that apply to TCAM only. General operating techniques for the MVT and MFT configurations of the control program are discussed in the chapter "General Operating Techniques."

HOW TO REASSIGN THE PRIMARY OPERATOR CONTROL STATION

When TCAM is set up, several stations can be designated as operator control stations. Normally the system console is designated as the primary operator control station, which makes it eligible to receive error recovery procedure messages, as well as to enter TCAM commands. All other operator control stations are known as secondary operator control stations.

You can determine the name of the primary operator control station and the names of all secondary operator control stations by using the PRITERM and SECTERM operands of the DISPLAY command.

If at any time it is desirable to reassign the primary operator control station to a different station within the telecommunications network, use the OPERATOR operand of the MODIFY command. By specifying OPERATOR=stationname, where stationname is the name associaed with any secondary operator control station in the network other than the system console, that station becomes eligible to receive all error recovery procedure messages. To reassign the primary operator control station to the system console, you enter OPERATOR=SYSCON. SYSCON is the name that is always assigned to the system console.

HOW TO DEACTIVATE AND REACTIVATE A STATION OR LINE

Using combinations of several TCAM commands, you can stop two-way message transmission for a line or a station, or message transmission in either direction from a station.

The operator commands used to accomplish changes in message transmission status are the HOLD, RELEASE, and VARY commands. Use the DISPLAY command to determine the current transmission status of all stations at any time.

Intercepting a station is the simplest form of transmission status change. When a station is intercepted, it may enter messages into the telecommunications network, but it may not receive any messages. This facility allows the operator of a remote station to leave the station for short periods of time without missing any messages sent to that station. You might be called upon to intercept a station for such a reason.

To intercept a station, use the HOLD command. When a station in intercepted, any messages for that station are retained in the central computer complex until the station is ready to receive them. To release a station from an intercept, use the RELEASE command. The INTER operand of the DISPLAY command is used to determine which stations within the network are intercepted at a given time.

If it is necessary to completely disconnect a station from a telecommunications network -- for machine failure or preventive maintenance, for example -- use the VARY command. The ACT and INACT operands of the DISPLAY command are used to determine which stations within the network are currently active or inactive.

Whenever a change to the status of a station is necessary using the VARY command, you must use the VARY command to stop message transmission on the line with which the station is associated before the status of the station can be changed.

In a deactivation procedure, you deactivate the line or line group, then the station. The line or line group may then be reactivated.

In a reactivation procedure for a station, you reactivate the station, then the line or line group. If the line or line group had been reactivated previously, you must deactivate the line or line group before you reactivate the station.

Index

TECHNICAL TERMS AND THEIR MEANINGS

This chapter defines technical terms used in this publication and in your every day work.

TT

Technical Terms and Their Meanings

a

accept: In TCAM, the process in which a destination station obtains a message transmitted to it from the computer (entering and accepting are functions of a station).

access method: A method for transferring data between main storage and a direct access storage or input/output devices.

active station: In TCAM, a station that is currently eligible for entering and/or accepting messages on the line. A station may be active for entering or active for accepting, or both.

address constant: A number, or a symbol representing a number, used in calculating storage addresses.

address stop: A capability to specify at the system console an address which causes a halt in processing.

alias: Another name for a member of a partitioned data set; another name for an entry point of a program.

allocate: To assign a resource for use in performing a specific job; job step, subtask of a job step, or job support task.

Alternate Path Retry (APR): A feature that allows an I/O operation that has developed an error to be retried on another channel assigned to the device performing the I/O operation. APR also provides the capability to establish other paths to an online or offline device.

American National Standard Institute: An organization sponsored by the Business Equipment Manufacturers Association (BEMA), which establishes industry standards.

American National Standard Labels (ANL): Tape labels that conform to the conventions established by the American National Standard Institute.

application program: A problem state program written by a user. A job.

ASCII: The American National Standard Code for Information Interchange.

asynchronous: Without regular time relationship; unexpected or unpredictable with respect to the execution of a programs instructions.

attach (task): To create a task and present it to the supervisor.

attribute: A trait; for example, attributes of data include record length, record format, data set name, associated device type and volume identification, use, creation date, etc.

automatic command: A command specified during system generation and executed after nucleus initialization in response to the AUTO parameter of the SET command.

automatic polling: A hardware feature of a telecommunications unit that processes a polling list. (See polling and polling list.)

automatic restart: A restart requested by the programmer after a job has been abnormally ended. The restart is dependent on operator action.

auxiliary storage: Data storage other than main storage.

availabilty: The degree to which a software/hardware system is available when needed to process data.

b

background: In TSO, the environment in which jobs submitted through the SUBMIT command or SYSIN are executed. One job step at a time is assigned to a region of main storage, and remains in storage to completion. Contrast with "foreground."

background job: A data processing job that is entered into an MFT or MVT job input queue for normal batch processing. The entry of a background job may be initiated locally, via a job input stream, or from a remote terminal. A background job may be performed concurrently with a higher priority foreground jobs.

<u>background reader</u>: In TSO, a system task started by the operator to process foreground-initiated background jobs.

<u>basic access method</u>: Any access method in which each input/output statement causes an input/output operation to occur.

<u>batch processing</u>: (See batched job processing.)

<u>batched job processing</u>: A technique whereby job definitions are placed one behind another on a common input device to form a batch of job definitions that are processed by the CPU with as little operator intervention as possible.

<u>binary</u>: A numbering system having a base of two; therefore the valid digits are 0 and 1.

<u>blocked data</u>: (See block.)

<u>block (records)</u>:

1. To group records to conserve storage space or to increase the efficiency of access or processing.

2. A blocked record.

3. A portion of a telecommunications message defined as a unit of data transmission.

<u>block loading</u>: Bringing the control section of a load module into adjoining positions of main storage.

<u>block prefix</u>: An optional field of variable length which may precede unblocked records, or blocks of records, on magnetic tapes recorded in ASCII.

<u>broadcast data set</u>: In TSO, a system data set containing messages and notices from the system operator, administrators, and other users. Its contents are displayed to each terminal user when he logs on the system, unless suppressed by the user.

<u>buffer (program)</u>: (See buffer (main storage).)

<u>buffer, main storage</u>: An area of main storage that is temporarily reserved for use in performing an input/output operation.

<u>byte</u>: Continuous storage equal to eight bits. (Eight bits in the IBM System/360 and IBM System/370.)

C

<u>call</u>: The transfer of control from one routine to another routine.

<u>catalog</u>:

1. In the operating system, a collection of data set indexes that are used by the control program to locate a volume containing a specific data set.

2. To include the volume identification of a data set in the catalog.

<u>cataloged data set</u>: A data set that is represented in an index or series of indexes.

<u>cataloged procedure</u>: A set of job control statements that has been placed in a cataloged data set, called the procedure library, and can be retrieved by naming it in an execute (EXEC) statement or started by the START command.

<u>cathode ray tube</u>: An electronic vacuum tube, such as a television picture tube, that can be used to graphically display numbers, letters, symbols, and images.

<u>CAW (channel address word)</u>: A word in main storage at location 72 that specifies the location in main storage where a channel program begins.

<u>CCW (channel command word)</u>: A doubleword at the location in main storage specified by the CAW. One or more CCWs make up the channel program that directs channel operations.

<u>central processing unit (CPU)</u>: A unit of a computing system that performs the work of processing data by executing predefined sequences of instructions, such as add, subtract, mulitply, and divide instructions.

<u>channel</u>: A hardware device that connects a CPU and main storage with input/output control units.

<u>channel address word</u>: (See CAW.)

<u>channel check handler</u>: A feature that, when a channel error occurs, records information about the error and issues a message to the operator.

<u>channel command word</u>: (See CCW.)

<u>channel-to-channel adapter</u>: A hardware device that can be used to connect a channel on one computing system to a channel on another system. It can also connect two channels on the same system.

checkpoint:

1. A point at which information about the status of a job step can be recorded so that the job step can be later restarted.

2. To record such information.

checkpoint data set: A physical sequential or partitioned data set containing a collection of checkpoint entries. If a checkpoint data set is a partitioned data set, each checkpoint entry is a member.

checkpoint/restart facility: A facility of the operating system that can minimize time lost in reprocessing a job step that terminated due to a program of system failure or intervention by the operator. A restart may begin from a checkpoint or from the beginning of a job step.

checkpoint restart: A restart that begins at a checkpoint. At the operator's discretion, the restart may be automatic or deferred, where deferred involves resubmitting the job.

command language: The set of commands, subcommands, and operands recognized by the system.

command processing: The reading, analyzing, and performing of commands issued via a console device or an system input stream.

commands, operator: A set of control language statements that are used by an operator to communicate with the operating system control program and direct the overall operation of the commuting system/operating system combination.

communication line: Any medium, such as a wire or a telephone circuit, that connects a remote terminal with a computer.

computing system: A central processing unit together with the main storage, input/output channels, control units, direct access storage devices, and input/output devices connected to it.

concatenated data set: A group of logically connected data sets.

control block: A storage area used by the operating system to hold control information.

control dictionary: The external symbol dictionary and relocation dictionary, collectively, of an object or load module.

control program: A program that is designed to schedule and supervise the performance of data processing work by a computing system.

control section: That part of a program specified by the programmer to be a relocatable unit, all of which is to be loaded into adjoining main storage locations.

control terminal: In TSO, any terminal at which a user authorized to enter commands affecting TSO operation is logged on.

control unit; input/output: A hardware unit of a computing system that controls the reading and recording of data on one or more direct access storage or input/output devices.

control volume: A volume that contains one or more indexes of the catalog.

conversational: Describing a program or a system that carries on a dialog with a remote terminal user, alternately accepting input and then responding to the input quickly enough for the user to maintain his train of thought.

Conversational Remote Job Entry (CRJE): An operating system component for entering job control language statements from a remote terminal, and causing the scheduling and execution of the jobs described. The terminal user is prompted for missing operands or corrections.

CPU (central processing unit): The unit of a system that contains the circuits that control and perform the execution of instructions.

CPU time: The amount of time denoted by the central processing unit to the execution of instructions.

CSW (channel status word): A doubleword in main storage at location 64 that provides information about the termination of an input/output operation.

cursor symbol: A short line (underscore) displayed on the cathode ray tube to indicate where the next character entered will be positioned.

TT

d

D format: a data set format, used only with ASCII data, in which logical records are of varying lengths and include a zoned decimal length indicator and in which records may be blocked, with each block containing a block prefix.

data cell: A direct access storage volume containing strips of tape on which data is stored. When being used, a data cell is mounted within an IBM 2321 Data Cell Drive.

data collection: A type of teleprocessing application in which data is received from remote terminals and stored for later processing.

data control block (DCB): A control block used by access routines in storing and retrieving data.

data definition name (ddname): A name appearing in the data control block of a program which corresponds to the name field of a data definition statement.

data definition (DD) statement: A job control statement that describes a data set associated with a particular job step.

data file:

1. A collection of related data records organized in a specific manner. For example, a payroll file (one record for each employee showing his rate of pay, deductions, etc.) or an inventory file (one record for each inventory item, showing the cost, selling price, number in stock, etc.).

2. In the operating system, a data set.

data management: A major function of the operating system that involves organizing, cataloging, locating, storing, retrieving, and maintaining data.

data organization: The arrangement of a data set.

data record: (See record.)

data set:

1. A named organized collection of data. (See data file.)
2. A telephone device used to transmit data.

data set catalog: (See catalog.)

data set control block (DSCB): A data set label for a data set in direct access storage.

data set label (DSL): A collection of information that describes the attributes of a data set, and that is normally stored with the data set; a general term for data set control blocks and tape data set labels.

data set utility programs: A collection of problem state programs that are provided with the operating system. The programs are designed to perform such functions as updating, maintaining, editing, and transcribing sets.

DD statement: (See data definition DD statement.)

debug: To detect, locate, and remove mistakes from a routine.

decimal: A numbering system having a base ten; therefore valid digits range from 0 to 9.

dedication: Describing the assignment of a system resource (e.g., an I/O device, a program, or a whole system) to one application or purpose.

default option: A language statement option that is selected by the operating system control program or a processing program in the absence of a selection by a user.

deferred restart: A restart performed by the system on resubmission of a job by the programmer. Operators submit the restart deck to the system through a system input reader.

delimiter: A character that groups or separates the words or values in a line of input.

density: The number of recorded bytes of data per unit of length; normally bytes or characters per inch (BPI or CPI).

destination code: A code in teleprocessing message that indicates the terminal to which the message is to be sent.

device independence: The ability to request input/output operations without regard to the characteristics of the input/output devices.

device type: The general term for a kind of device, specified at the time the system is generated. For example, 2311 or 2400. (See unit address and group name.)

dial-up terminal: A terminal on a switched network.

dialing: Establishing a connection, through common communication lines, between a central computing system and a remote terminal.

direct access: Retrieval or storage of data by a reference to its location on a volume, rather than relative to the previously retrieved or stored data.

direct access application: A type of computing system application in which master data file records are normally processed and interrogated in a random (non-sequential) order.

direct access device: A device in which the access time is effectively independent of the location of the data.

directory: An index that is used by the operating systems control program to locate one or more sequential blocks of data (called numbers) that are stored in separate partitions of a partitioned data set in direct access storage.

disabled: A state of the CPU that prevents the occurrence of certain types of interruptions.

disk pack: A direct access storage volume containing magnetic disks on which data is stored. When being used, a disk pack is mounted on a disk storage drive, such as the IBM 2311 Disk Storage Drive.

dispatching priority: A number assigned to tasks to determine the order in which they will use the central processing unit in a multitask situation.

documentation, program: The flowcharts, instruction lists, and other documents that are used to define or describe a program.

dump (main storage):

1. To copy the contents of all or part of main storage onto an output device, so that it can be examined.
2. The data resulting from number 1.
3. A routine that will accomplish number 1.

dynamic area: An area of main storage that is allocated for performing job step or job support tasks.

dynamic data set definition: The process of defining a data set and allocating auxiliary storage space for it during job step execution rather than before job step execution.

Dynamic Device Reconfiguration (DDR): A feature that allows a demountable volume to be moved from one device to another, and repositioned if necessary, without abnormally terminating the affected job or reloading the IPL program. A request to move a volume may be initiated by either the operator or the system. For M65MP DDR is required for SYSRES or non-SYSRES devices.

e

enabled: A state of the CPU that allows the occurence of certain types of interruptions determined by the current program status word.

enter: In TCAM, the process in which a station places on the line a message to be transmitted to the computer (a station enters and accepts messages, while a computer sends and receives messages).

entry point: Any location in a program to which control can be passed by another program.

error recovery routines: A control program routine that attempts a recovery from a machine error condition by repeating the operation that was performed in error. If the error does not persist, normal processing is continued at the point at which it was interrupted by the error condition. Some error routines also attempt to repair any program change caused by an error.

event: An occurrence of significance to a task; typically, the completion of an asynchronous operation, such as in input/output operation.

event control block (ECB): A control block used to represent the status of an event.

exclusive segments: Segments in the same region of an overlay program, that are not in the same path. They cannot be in main storage simultaneously.

TT

execute (EXEC) statement: A job control statement that designates a job step by identifying the load module to be fetched and executed to perform the step.

extent: The physical locations on input/output devices occupied by or reserved for a particular data set.

extended binary coded decimal interchange code (EBCDIC): A coded method of representing data.

external reference: A reference to a symbol defined in another module.

external symbol: A control section name, entry point name, or external reference; a symbol contained in the external symbol dictionary.

external symbol dictionary (ESD): Control information associated with an object or load module which identifies the external symbols in the module.

f

F format: A data set record format in which the logical records are the same length.

facility:

1. A measure of how easy it is for people to operate, use, and manage the use of a software/hardware system. Together with system performance, the facility of a system is a major factor on which the total productivity of an installation depends.
2. A feature of the operating system designed to serve a particular purpose -- for example, the checkpoint/restart facility.

fetch:

*1. To locate and load a quantity of data from storage.
2. A control routine that accomplishes number 1.

field, data: One or more items of information that together make up a record, such as an account number, the name of a person, or the calculated interest on a loan.

file: (See data file.)

file protect: The process by which a volume of recorded data can be 'protected' by disabling the writing mechanism (WRITE head) of a unit.

fixed storage area: That portion of main storage occupied by the resident portion of the control program (nucleus).

foreground: Describing the environment in which programs invoked by commands are executed. Programs are swapped in and out of main storage as necessary to efficiently use main storage. Contrast with "background".

foreground-initiated background job: A job submitted from a remote terminal for scheduling and execution in the background environment.

foreground job: Usually a teleprocessing or graphic display job that has an indefinite running time during which communications is established with one or more users at local or remote terminals.

With an MFT or MVT control program configuration, a foreground job may be performed concurrently with lower priority background jobs.

foreground message processing program: In TSO, a problem program run in the foreground using TCAM to handle messages for one or more terminals.

g

general purpose operating system: A operating system designed to handle a wide variety of computing system applications.

generalized sort/merge program: A program that is designed to sort or merge a wide variety of records in a variety of formats.

generation data group: A collection of successive, historically related data sets.

GJP: (See graphic job processor.)

graphic display program: A program designed to display information, in graphic or alphameric form, on the face of TV-like display tube.

graphic job processing: An optional feature of an MFT or MVT control program configuration that enables users at remote display units to quickly and conveniently define and start jobs that are processed by the operating system. The feature also allows interactive use of graphic display programs.

graphic job processor (GJP): A program that elicits job control information from a user as he selects and performs job control operations at a display unit. It interprets the information entered by the user at the display unit and converts it into job control language.

graphic programming services: A number of services provided by the operating system for use in designing and executing programs that communicate with a user at an IBM 2250 Display Unit, or an IBM 2260 Display Station.

group name: The name for a particular collection of devices, specified at the time the system is generated. For example, SYSDA or TAPE. (See unit address and device type.)

h

hardcopy log: A permanent record of system activity that is mandatory for systems with an active graphic console or multiple active consoles. For a system with a graphic console, the hard copy log is kept on another console device; for a system with multiple consoles, it can also be kept on the system log.

hardware: The physical equipment that you operate. For example, magnetic tape drives, disk drives, printer, console, CPU, etc.

hardware resources: CPU time, main storage space, input/output channel time, direct access storage space, and input/output devices, all of which are required to do the work of processing data automatically and efficiently.

hexadecimal: A numbering system with a base of 16; therefore, valid digits range from 0 through F, where F represents the highest units position (15).

hierarchy storage: A division of main storage that allows hierarchy 0 and hierarchy 1 to be addressed separately. For PCP, MFT, and MVT systems with hierarchy support and an IBM 2361 Core Storage Unit, processor storage is addressed as hierarchy 0, and the 2361 is addressed as hierarchy 1. For MVT with hierarchy support, but with no 2361, there are still two hierarchies: both are in processor storage.

home address: That address written on a direct access volume, denoting the track's address relative to the beginning of the recording volume.

"human oriented" language: A programming language that is more like a human language than a machine language.

human resources: The members of the applications planning and programming staff, the operations staff, and the system programming staff at a data processing installation.

i

I/O-processor overlap: The automatic process by which channels control I/O operations while the CPU carries out normal instruction execution.

IBM problem state programs: Problem state programs that are provided by IBM as part of the operating system. They include the master and job schedulers, language translators, and service programs.

IBM System/360: A large collection of computing system devices that can be connected together in many combinations to produce a wide range of computing systems that share many characteristics, including a common machine language.

IBM System/370: An upward compatible extension of the IBM System/360, offering new functions and improved price/performance.

IBM System/360 Operating System: A comprehensive collection of control program options, language processors, I/O support, application programs, and service programs designed to meet the needs of the users who require the extensive facilities of a large operating system.

IBM 2250 Display Unit: An input/output unit that can be used to provide high-speed visual communication between a computing system and its users. Information directed to a user is displayed on the face of a cathode ray tube in the form of alphameric data, tables, graphs, charts, etc. A keyboard and a light pen on the unit make it possible for a user to alter the display by entering or changing information.

IBM 2260 Display Station: A table-top input/output device for displaying computer output information in alphameric form, either locally or at remote location. Information from the computer, such as insurance records or airline reservations, are displayed on the face of a cathode ray tube. A keyboard can be attached to the device so that messages can be keyed in by a user, displayed, and sent to the computer.

IBM 2361 Core Storage: Magnetic core storage that can be added to the processor main storage of some System/360s. The core storage is an extension of and is addressed in the same way as processor main storage. The number of bytes obtained per storage cycle, and all other features of 2361 storage except its 8-microsecond speed, are the same as those of the processor storage of the system to which it is attached.

inactive station: In TCAM, a station that is currently ineligible for entering and/or accepting messages.

inclusive segments: Segments in the same region of an overlay program that can be in main storage simultaneously.

independent utility programs: A group of utility programs that support, but are not part of, the operating system. They are used chiefly by the system programmer to initialize and prepare direct access storage devices for use under operating system control.

index (data management):

1. A table in the catalog structure used to locate data sets.
2. A table used to locate the records of an indexed sequential data set.

information management system: A system designed to organize, catalog, locate, store, retrieve, and maintain information.

information resources: Programs executed by a CPU and the data processed by a CPU.

information retrieval system: A computing system application designed to recover specific information from a mass of data.

initial program loading (IPL): The initialization procedure which loads the nucleus and begins normal operations.

initialize: To set counters, switches, address, etc., to zero or other starting values at the beginning of, or at prescribed points in, a computer program.

initiator: (See initiator/terminator.)

initiator/terminator: A part of the job scheduler. In an MFT or MVT configuration of the control program, the initiator/ternator selects a job from the input work queue, allocates resources required to perform a step of the job, loads and transfers control to the program that is executed to perform the job step, and terminates the job step when execution of the program is completed. In a PCP control program configuration the initiator performs the same basic functions except that each job is received directly from the reader/interpreter.

input buffer: An area of main storage used to store a data block received from an input device for processing by the CPU.

input job queue: A collective term for the fifteen queues of job information which the job scheduler uses to select the jobs and job steps to be processed. Each of the fifteen queues is associated with one input job class. (See input work queue.)

input/output channel: (See channel.)

input stream: Job control statements entering the system; may also include data.

input queue: (See input work queue.)

input work queue: A queue (waiting list) of job definitions in direct access storage assigned to a job class and arranged in order of priority assignment. Job definitions are entered into an input work queue by one or more readers, interpreters, and are selected and removed by one or more initiator/terminators. An input work queue is not used in the primary control program (PCP) configuration.

inquiry and transaction processing: A type of teleprocessing application in which inquiries and records of transactions are received from a number of remote terminals and are used to interrogate or update one or more master files maintained by the central system.

installation: A particular computing system, in terms of the overall work it does and the people who manage it, operate it, apply it to problems, service it, and use the results it produces.

interaction: In TSO, a basic unit used to record system activity, consisting of acceptance of a line of terminal input, processing of the line, and response, if any. Interactions are recorded when a user task starts its wait for a line of terminal input.

interaction time: In TSO, the duration of an interaction.

intercepted station: In TCAM, a station to which no messages may be sent. (An intercepted station can still enter messages -- only messages queued for the destination are stopped.)

interruption: A transfer of CPU control to the supervisor that is initiated automatically by the computing system or by a problem state program through the execution of a supervisor call (SVC) instruction. The transfer of control occurs in such a way that control can later be restored to the interrupted program, or, in systems that perform more than one task at a time, to a different program.

interruption network: A network of circuits throughout a computing system that continuously monitors its operation. When an event occurs that would normally require intervention and direction by the supervisor, it is detected by the network and an interruption is initiated.

invitation list: In TCAM, a sequence of polling characters or identification sequences associated with the stations on line; the order in which the characters are specified determines the order in which the stations are invited to enter a message.

j

job:

1. The major unit of work performed under operating system control. A job consists of one or several related steps. It is defined by a series of job control language statements consisting of a JOB statement, one or more job step (EXEC) statements, and, for each step, one or more data definition (DD) statements.
2. In TSO, in the foreground environment, the processing done on behalf of one user from LOGON to LOGOFF -- one terminal session.

job batch: A succession of job definitions that are placed one behind another to form a batch. Each job batch is placed on an input device and processed by the operating system with a minimum of delay between one job or job step and another.

job class: Any one of a number of job categories that can be defined at an installation when using an MFT or MVT control program configuration. Each job can be assigned to any one of several predefined job classes and each initiator/terminator can be directed to initiate jobs from one to three different classes. By classifying jobs and directing initiator/terminators to initiate specific classes of jobs, it is possible to control the mixture of jobs that are performed concurrently.

job control language: A high-level programming language used to code job control statements.

job control statement: A control statement in the input stream that identifies a job and defines its needs.

job definition: A series of job control statements that define a job. (See job.)

job input device: A device assigned by the operator for use by the operating systems control program in reading a batch or streams of job definitions and any accompanying data.

job input file: A data file (or data set) consisting of a series of job definitions and accompanying input data.

job input stream: A continuous series or stream of job definitions, and accompanying data, that is read and processed by an operating system control program without necessarily stopping between one job step and another.

job library: A set of user-identified partitioned data sets used as the main source of load modules for a given job.

job management: A major function of the operating system involving the reading and interpretation of job definitions, the scheduling of jobs, the initiation and termination of jobs and job steps, and the recording of job output data.

job output device: A device assigned by the operator for common use in recording output data for a series of jobs.

job output file: A data file (or data set) consisting output data produced by a series of jobs.

job output stream: A continuous series or stream of output data from a number of jobs that is recorded by the operating system control program.

job priority: A value assigned to an MVT job that, together with an assigned job class, determines the priority (relative to other jobs) to be used in scheduling the job and allocating resources to it.

job processing: The reading of control statements from an input job stream, the intiating of job steps defined in these statements, and the writing of system messages and SYSOUT data sets.

job queue: (See input work queue.)

job scheduler: A part of the operating system control program that reads and interprets job definitions, schedules the jobs for processing, initiates and terminates the processing of jobs and job steps, and records job output data.

job (JOB) statement: A job control statement that is used to identify the beginning of a job. It contains such information as the name of the job, an account number, and the class and priority assigned to the job.

job step: A unit of work associated with one processing program or one cataloged procedure, and related data.

job step restart: (See step restart.)

job step task: A task that is initiated by an initiator/terminator in the job scheduler in accordance with specifications in an execute (EXEC) statement. In an MVT control program configuration, a job step task can initiate any number of other tasks.

job stream: (See job input stream and job output stream.)

job support tasks: Tasks that read and interpret job definitions and convert job input and output data from one direct access storage or input/output medium to another.

k

keyword: A part of a command operand that consists of a specific character string (such as DSNAME=).

l

label, magnetic tape: One or more records at the beginning of a magnetic tape reel that identifies and describes the data recorded on the tape, and contains other information, such as the serial number of the tape reel.

language statement: A statement that is coded by a programmer, operator, or user of a computing system, and is used to convey information to a processing program such as a language translator program, service program, or control program. A statement may signify that an operation be performed or may simply contain data that is to be passed to the processing system.

language translator: A program that transforms statements from one language to another without significantly changing their meaning.

library:

1. A collection of objects (for example, data sets, volumes, card decks) associated with a particular use, and identified in a directory. See job library, system library, and link library.

2. Any partitioned data set.

library reference system: A collection of indexes, tables, and control blocks that are used by the operating system control program to classify and locate data sets.

light pen: A pen-shaped instrument that can be used to sense light at a point on the 2250 screen.

limit priority: A number associated with a task in a multitask operation, representing the highest dispatching priority that the task can assign to itself or to any of its subtasks.

line group: In TCAM, a set of one or more communications lines of the same type, over which stations can communicate with the central computer.

line number: In TSO, a number associated with a line in a line data set, which can be used to refer to the line.

link library: A partitioned data set which, unless otherwise specified, is used in fetching load modules referred to in execute (EXEC) statements and in ATTACH, LINK, LOAD, and XCTL macro instructions.

link pack area: An area in upper main storage containing a list of track addresses (BLDL list) for routines that reside in SYS1.LINKLIB, routines from SYS1.SVCLIB and SYS1.LINKLIB as selected by the user, types 3 and 4 routines, and master scheduler and system modules required resident by system tasks. The link pack area is set up by the nucleus initialization program (NIP) at the time of initial program loading.

linkage: The way two routines or modules communicate.

linkage conventions: A set of operating system conventions that should be adhered to when passing control from one program module to another. Adherence to the conventions helps to ensure program sharing and compatibility.

linkage editor: A processing program that can be used to combine program segments or modules that are independently compiled or assembled. The linkage editor also enables a program that is too large for the space available in main storage to be divided so that executed segments of the program can be overlaid by segments yet to be executed.

listing: A printout that lists the source language statements and contents of a program. Usually prepared by a language translator.

load: To place a program into main storage so that it can be executed.

load module: A program or part of a program, formed of one or more object modules or other load modules, that is ready to be loaded into main storage by the control program for execution by the CPU.

load module library: A partitioned data set used to store and retrieve load modules.

loader: A service program that combines the basic editing and loading functions of the linkage editor and program fetch. It loads object modules into main storage for execution; however, it does not produce load modules.

Local System Queue Area (LSQA): In TSO, a portion of the foreground (swapped) region used for control blocks that are to be swapped out along with a terminal job.

locate mode: A way of providing data by pointing to its location instead of moving it.

logical record: A record that is defined in terms of the information it contains rather than by its physical traits.

LSQA: (See Local System Queue Area.)

m

machine check handler for Model 65 (MCH/65): A feature that analyzes an error and attempts recovery by retrying the failing instruction. If retry is not possible, or if it is unsuccessful, MCH/65 will attempt to repair the malfunction, or isolate the task, or both.

machine check handler for Models 85, 155, and 165 (MCH/85, MCH/155, and MCH/165): A standard feature for MFT and MVT that constructs a record of errors successfully or unsuccessfully retried by machine recovery facilities. In the case of an unsuccessful retry, the MCH programs analyze the error and attempt repair and/or isolation and termination of the affected task.

machine language: A number language consisting of a set of arithmetic and other instructions that can be executed directly by a computing system. The language is almost always based on some form of the binary numbering system.

"machine oriented" language: A programming language that is more like a machine language than a human or mathematical language.

macro instruction: An instruction in a source language that is equivalent to a specific sequence of machine instructions.

macro library: A partitioned data set that is used for storing and retrieving instruction sequences whose execution can be called for by macro instructions.

main storage: The storage in a computing system from which a central processing unit can directly obtain instructions and data and to which it can directly return results.

main storage partition: In an MFT control program configuration, a part of the dynamic area of main storage assigned by the operator for use in performing specific job support tasks or classes of jobs.

main storage region: In an MVT control program configuration, a section of main storage that is allocated by the control program for use in performing a job step or a job support task.

major time slice: In TSO, the period of time for which a terminal job is swapped into main storage.

management information system: An application of a computing system in which the system provides information used in the management of an enterprise.

master console: In a system with multiple consoles, the basic console required for operator-system communication. There can be only one master console through which the operator can enter all operator commands and change the status of the hard copy log.

master data file: A file containing such data as names and addresses, current credits and debits, running inventories, and other identifying or historical facts about individuals, accounts, items, or services.

master file: (See master data file.)

master scheduler: A part of the control program that serves as a two-way communications link between the operator and the system, usually by way of the operator's console. It is used to relay messages from the system to the operator, to execute operator commands, and to respond to replies from the operator. In MFT and MVT control program configurations, the master scheduler is used to start and stop the reader/interpreter, initiator/terminator, and output writer tasks.

member: A partition of a partitioned data set.

merge: To combine records from two or more similarly ordered data sets into one set that is arranged in the same order.

message: A combination of characters and symbols transmitted from one point to another.

message control program: A program that is used to control the sending or reception of messages to or from remote terminals.

message header: The leading part of a message that contains information concerning the message, such as the source or destination code of the message, the message priority, and the type of message.

message, operator: A message from the operating system (or a problem program (job)) directing the operator to perform a specific function, such as mounting a tape reel, or informing him of specific conditions within the system, such as an error condition.

message processing program: A program that processes or otherwise responds to messages received from remote terminals.

message queue: A queue (waiting list) of messages that are awaiting processing or waiting to be sent to a remote terminal.

message switching: A teleprocessing application in which a message received by a central system from one remote terminal is sent to one or more other remote terminals.

message, teleprocessing: A unit of information that is transferred to or from a remote terminal by way of telecommunication lines.

message text: A part of a message consisting of the actual information that is routed to a user at a terminal or to a program in a central system that is to process it. (See message header.)

MFT: Multiprogramming with a fixed number of tasks.

microsecond: One millionth of a second.

module:

1. A program or part of a program that can be modified or replaced without affecting other code.

2. A program unit that is input to, or output from, a single execution of an assembler, compiler, or linkage editor; a source, object, or load module.

monitor: A name sometimes used to refer to all or a part of an operating system control program.

multiple-job processing: Controlling the performance of more than one data processing job at a time.

multiple-task management: Managing the performance of more than one data processing task at a time.

multiprocessing: A technique whereby the work of processing data is shared among two or more interconnected central processing units under integrated control that directly or indirectly communicate with one another, other than through direct human intervention.

multiprocessing system: A computing system employing two or more interconnected processing units to execute programs simultaneously.

multiprogramming: Executing more than one program concurrently by interleaving the execution of one with that of another. The term "multiprogramming" is also broadly used to refer to the performance of more than one data processing task concurrently whether or not a single reenterable program, or several programs, are executed to perform the tasks.

multisystem mode: An operating mode of the Model 65 Multiprocessing System. When operating in this mode, all of main storage is shared by both CPUs. Also, most input/output and auxiliary storage devices, along with their associated control units, are shared by the two CPUs.

multitask operation: Multiprogramming; called multitask operation to express parallel processing not only of more than one program, but also of a single reenterable program used to do many tasks.

MVT: Multiprocessing with a variable number of tasks.

n

network: In teleprocessing, a number of communication lines connecting a computer with remote terminals.

non-resident portion/control program: Those control program routines that are loaded into main storage as they are needed, and can be overlaid after their completion.

non-switched line: A connection between a remote terminal and a computer that does not have to be established by dialing.

nucleus: The portion of a control program that always remains in main storage.

o

object module: The output of a language translator for a particular translation that contains a program or part of a program in the form of machine language instructions.

object module library: A partitioned data set that is used to store and retrieve object modules.

object program: A program that has been compiled or assembled by a language translator.

offline: Pertaining to resources with which the central processing unit has no direct communication or control.

offline applications: A data processing application in which the computing system does not communicate directly with the original source of final destination of the data it processes.

online: Pertaining to resources with which the central processing unit has direct communication or control.

online application: A data processing application in which the computing system communicates directly with the source and destination of the data it processes.

online direct access system: A data processing application in which the computing system communicates directly with the source and destination of the data it processes and which transaction data records are processes in a random (non-sequential) order; usually in the order in which they are received.

online problem solving: A teleprocessing application in which a number of users at remote terminals can concurrently use a computing system in solving problems online. Often, in this type of application, a dialogue or conversation is carried on between a user at a remote terminal and a program within the central computing system.

operands: In the operator command language, information entered with a command name to define the data on which a command processor operates and to control the execution of the command processor. Some operands are positional, identified by their sequence in the command input line, others are identified by keywords.

operating system: An application of a computing system, in the form of organized collections of programs and data, that is specifically designed for use in creating and controlling the performance of other applications.

operations staff: The members of a data processing installation who receive jobs from the programmers, schedule the order in which the jobs are presented to the system and performed, and direct the overall operation of the system in performing the jobs.

operator: A member of a data processing installation operations staff who is responsible for directing the operation of a computing system. The same, or a different operator, may perform routine functions such as mounting tape reels and loading card decks.

operator command: A statement to the control program, entered through a console device, which causes the control program to provide requested information, alter normal operations, initiate new operations, or terminate existing operations.

operator control station: In TCAM, any station that is eligible to enter operator commands. An application program and the system console may serve also as operator control stations.

operator message: (See message, operator.)

output buffer: An area of main storage used to store a data block before it is transferred to an output device.

output class: In an MFT or MVT control program configuration, any one of up to 36 different output classes, defined at an installation, to which output data produced during a job step can be assigned. When an output writer is started, it can be directed to process from one to eight different classes of output data.

output queue: (See output work queue.)

output work queue: A queue (waiting list) of data definitions in direct access storage assigned to an output class and arranged in order of priority assignment. Data definitions are removed from an output work queue by one or more output writers. The definitions specify for the output writers the location and disposition of each system data set. An output work queue is not used in the primary control program configuration.

output writer: A part of the job scheduler that writes output data sets onto a system output unit, independently of the program that produced such data sets.

overlay: To place a load module or segment of a load module into main storage locations occupied by another (already executed) load module or segment.

overlay module: A load module that has been divided into overlay segments, and has been provided by linkage editor with information that enables overlay supervisor to load the segments when required.

overlay region: A contiguous area of main storage within which segments can be loaded independently of paths in other overlay regions. Only one path within a region can be in main storage at any one time.

P

partition: (See main storage partition.)

partitioned data set: A special type of data set that is stored in direct access storage and can be cataloged like any other data set. A partitioned data set is usually called a program library. It is divided into independent partitions (members), each of which normally contains a program, or part of a program, in the form of one or more sequential data blocks. Each program library contains a built-in directory (or index) that the control program can use to locate a program in the library.

partitioned mode: An operating mode of the Model 65 Multiprocessing System. When operating in this mode, main storage, control units, auxiliary storage units, and input/output devices are apportioned between the two CPUs, each of which operates as a separate and distinct system.

password: In TSO, a one- to eight-character symbol assigned to a user that he can be required to supply at the time he logs on the system. The password is confidential, as opposed to the user identification. Users can also assign passwords to data sets.

patching: A makeshift technique for modifying a program or correcting programming errors by changing the object code of the program, usually to avoid recompiling or reassembling the program.

path:

1. A series of segments that form the shortest distance in a region of an overlay program between a given segment and the root segment.

2. In Alternate Path Retry, the path through a channel/control unit to an I/O device.

PCP: Primary Control Program. A PCP control program is designed to perform one job or task at a time.

performance: Together with facility, one of the two major factors on which the total productivity of a hardware/software system depends. Performance is largely determined by a combination of three other factors: throughput, response time, and availability.

physical record: A record that is defined in terms of physical qualities rather than by the information it contains.

PL/I: Programming Language I. A high-level programming language that has features of both COBOL and FORTRAN, plus additional features.

polling: A technique by which each of the stations sharing a communications line is periodically checked to determine if it requires servicing.

polling list: In TCAM, a list that specifies a sequence in which stations are to be periodically checked to determine if they require servicing.

Post: Note the occurrence of an event.

Primary Operator Control station: In TCAM, the Operator Control station that can receive an Error-Recovery Procedure message and send operator commands and receive related responses.

priority: The relative standing a job or task has in the system as opposed to the other jobs and tasks in the system at a given time. See dispatching priority, job priority.

priority scheduling system: A form of job scheduler which uses input and output work queues to improve system performance.

private library (of a job step): A partitioned data set other than the link library or the job library.

privileged instruction: An instruction that can only be executed when the CPU is in the supervisor state.

problem program: (See problem state program.)

problem state: A state of the central processing unit during which input/output and other privileged instructions cannot be executed. Opposite of supervisor state.

problem state program: Any program that is executed when the central processing unit is in the problem state. This includes IBM-distributed programs, such as language translators and service programs, as well as programs written by a user.

procedure library: A program library in direct access storage containing job definitions. The reader/interpreteer can be directed to read and interpret by an execute statement in a job stream.

process control: A computing system application in which a manufacturing or other process is automatically monitored and controlled.

processing program: Any program capable of operating in the problem program state. This includes IBM-distributed language processors, application programs, service and utility programs, and user-written programs.

processor:

1. In hardware, a central processing unit (CPU).

2. In software, a problem state program such as a language translator or service program that is usually provided by IBM and is widely used at an installation.

productivity: A measure of the work performed by a software/hardware system. Productivity largely depends on a combination of two factors: the facility (ease of use) of the system and the performance (throughput, response time, and availability) of the system.

program: A logically self-contained sequence of instructions that can be executed by a computing system to attain a specific result.

program fetch: A part of the operating system supervisor that retrieves load modules from auxiliary storage and places them in main storage, relocating them as necessary.

program library: (See partitioned data set.)

program loader: A program or routine that is used to load other programs and routines into main storage.

program status word (PSW): A doubleword in main storage used to control the order in which instructions are executed, and to hold and indicate the status of the computing system in relation to a particular program.

programmed polling: In TCAM, a software facility that processes a polling list. (See polling and polling list.)

prompting: In TSO, a system function that helps a terminal user by requesting him to supply operands necessary to continue processing.

protection key: A task-associated indicator (key) that appears in the current PSW whenever a task is active (i.e., has control of the system); this indicator must match the storage keys of all main storage blocks that the task is to use.

PSW: (See program status word.)

q

qualified name: A data set name that is composed of multiple names separated by periods (for example, TREE.FRUIT.APPLE)

qualifier: All names in a qualified name other than the rightmost, which is called the simple name.

queue: A waiting line or list.

queued access method: An access method that automatically governs the movement of data between the program using the access method and input/output devices. (The primary macro instructions used are GET and PUT.)

r

reader: That part of the scheduler that reads into the system an input stream.

reader/interpreter: A part of the job scheduler that reads and interprets a series of job definitions from a job input stream.

ready condition: The condition of a task that is ready to be performed by the central processing unit.

real-time application: An application in which a computing system is used to assist in or guide a process while the process actually transpires.

record: One or more data fields that represent an organized body of related data, such as all of the basic accounting information concerning a single sales transaction.

reenterable program: A program that can be executed to perform each of several tasks concurrently and produce several independent sets of results. A reenterable program must be designed so that it is not changed in any way when the CPU executes it.

region: (see main storage region, overlay region.)

relocatability: The ability of a program (in the form of a load module) to be dynamically loaded anywhere in main storage.

remote job entry (RJE): A type of teleprocessing application in which jobs (like those entered into a system locally) are received from one or more remote locations and processed. Remote job entry is provided as an optional feature of an MFT or MVT control program configuration.

remote job processing: (See remote job entry.)

remote terminal: An input/output control unit and one or more input/output devices usually attached to a system through an IBM telecommunications control unit. It may or may not be physically remote from the system.

report program generator (RPG): A processing program that can be used to generate object programs that produce reports from existing sets of data.

resource, data processing: Any one of the human, hardware, and information assets of a data processing organization that are required in order to process data automatically and efficiently. (See human resources, hardware resources, and information resources.)

resource queue: A logical list of control blocks or other data elements that represent tasks awaiting the allocation of a particular resource.

response time:

1. The time between the submission of an item of work to a computing system and the return of results. Loosely, turnaround time.
2. In online systems, the time between the end of a block of user input and the display of system response at the terminal.

restart: (See checkpoint restart and step restart.)

restart deck: A series of JCL statements which constitute a complete job description for a job that is to be resumed at the beginning of a particular job step or at a checkpoint within this job step. If the job is to be resumed at a checkpoint, a DD statement named SYSCHK must precede the first EXEC statement in the restart deck.

return code: A number placed in a designated register (the "return code register") at the completion of a program. The number is established by user-convention and may be used to influence the execution of succeeding programs or, in the case of an abnormal end of task, it may simply be printed for programmer analysis.

return code register: A register in which a user-specified condition code is placed at the completion of a program.

reusable: Usable by two or more tasks. (See reenterable, serially reusable.)

rollout/rollin: An optional feature of the MVT control program configuration that enables an additional region (or regions) of main storage to be temporarily reassigned from one job step to another.

root segment: The first segment in an overlay program.

routine: A part of a program or subprogram that may have general or frequent use.

routing code: A code assigned to each operator message and used, in systems with the multiple console support (MCS) option, to route the message to the proper console.

S

satellite computer: An offline auxiliary computer. (See auxiliary computing system.)

satellite graphic job processor: A program that elicits job control information from a user at an IBM 2250 Display Unit attached to an IBM 1130 computing system allowing him to define and initiate jobs to be processed by a remote IBM System/360 Operating System. The program interprets the information entered by the 2250 user and converts it into job control language. Jobs so defined can be processed either independently in the System/360 or in conjunction with a related program in the 1130.

scatter loading: Placing the control sections of a load module into nonadjoining positions of main storage.

scheduler: (See master scheduler and job scheduler.)

secondary console: In a system with multiple consoles, any local or remote console except the master console. The secondary console handles one or more assigned functions on the multiple console system.

secondary operator control station: In TCAM, and operator control station that can send operator commands and receive related responses.

secondary storage: Auxiliary storage.

seek: To position the access mechanism of a direct access device at a specified location.

segment:

1. The smallest unit (one or more control sections) that can be loaded during execution of an overlay program.
2. As applied to telecommunications, a portion of a message that can be contained in a buffer.

self-initializing program: A program that initializes itself each time it is executed.

sequential access application: A type of data processing application in which master data records are arranged, updated, and interrogated in a prescribed sequence and each new batch of transaction data records are first sorted and then processed in the same sequence.

serially reusable: Usable by another task after the current use has been concluded.

service program: A processing program such as the linkage editor, sort/merge program, or a utility program that is designed mainly to perform specific services for a user of the program.

session: In TSO, the elapsed time from when the terminal user logs on the system until he logs off the system.

setting up: The act of preparing a computing system to perform a job or job step. Setting up is usually performed by an operator or assistant operator and often involves performing routine functions, such as mounting tape reels and loading card decks.

setup time: The time required by an operator to prepare a computing system to perform a job or job step.

SGJP: (See satellite graphic job processor.)

short block: A block of F-format data which contains fewer logical records than are specified for a block.

simple name: The rightmost component of a qualified name (for example, APPLE is the simple name in TREE.FRUIT.APPLE).

simultaneous processing: The performance of two or more data processing tasks at the same instant of time. Contrast with concurrent processing.

SMF: (See "System Management Facilities.")

software resources: The program and data resources that represent the software associated with a computing system.

sort/merge program: A generalized processing program of the operating system that can be used to sort or merge records in a prescribed sequence.

sorting: The act of collating records of a data set or file in accordance with a prescribed sequence.

source code, message: A code that indicates the source of a message received from a remote location.

source language: The input to a language translator; for example, FORTRAN, COBOL, PL/I.

source module: A series of language statements that represent the input to a language translator for a particular translation.

source module library. A partitioned data set that is used to store and retrieve source modules.

source program: A program written in a source language.

stacked job processing: (See batched job processing.)

station: In TCAM, either a remote terminal, or a remote computer used as a terminal.

step restart: A restart that begins at the beginning of a job step. The restart may be automatic or deferred, where deferred involves resubmitting the job.

storage block: An area of main storage consisting of 2048 bytes to which a storage key can be assigned.

storage dump: A recording of the contents of main or auxiliary storage so that it can be examined by a programmer or operator.

storage reconfiguration: An M65MP function that makes an area of defective storage unavailable and frees any system resources associated with it.

subprogram: A sequence of instructions stored in a library, that can be incorporated as part of a compiler.

subroutine: A relatively short sequence of instructions that can be incorporated into a program to perform a specific function, such as finding the square root of a number.

subsystem, operating system: A major part of an operating system which, in itself, is a system; such as, a language translator or a supervisor.

subtask: A task that is initiated and terminated by a higher order task.

summary report: A report that represents a summary and selection of detailed information; such as, an annual earnings report.

supervisor: A major part of the operating system control program that is executed when the CPU is in the supervisor state. The supervisor directs and controls the execution of problem state programs and provides them with a variety of services.

supervisor call (SVC) instruction: An instruction that interrupts the program being executed and passes control to the supervisor for the purpose of performing a specific service indicated by the instruction.

supervisor state: A state of the central processing unit during which input/output and other privileged instructions can be executed. Opposite of problem state.

swap: In TSO, to write an image of a foreground job's main storage region to auxiliary storage, and to read in another job's main storage image into the region.

swap allocation unit: In TSO, an arbitrary unit of auxiliary storage space into which a swap data set is divided, and by which it is allocated.

swap data set: In TSO, a data set dedicated to the swapping operation.

swap data set control block: In TSO, a control block that describes a swap set. It contains a DCB, a space queue, and device dependent control information.

swap in: The process of reading an image of a terminal job's main storage from a swap data set into the appropriate main storage locations.

swap out: In TSO, the process of writing an image of a terminal job's main storage region from main storage to a swap data set.

switching: In TSO, describing a connection established by dialing between a remote terminal and a computer.

symbolic I/O assignment: A means by which problem programs can refer to an I/O device by a symbolic name.

synchronous: Occuring with a regular or predictable time relationship.

syntax checker: A program that tests source statements in a programming language for violations of that language's syntax.

SYSCTLG: A system data set on the primary system residence device containing addresses relating installation data set names to specific volume numbers.

SYSIN: A system input stream, also, a name used as the data definition name of a data set in the input stream.

SYSOUT: A system output stream. Also, an indicator used in data definition statements to signify that a data set is to be written on a system output unit.

system application: An application of a computing system involving the design and use of a software system; for example, a management information system, an information retrieval system, or an operating system.

system generation: The process of using one operating system to assemble and link together into a coherent whole all the required, alternative and optional parts that form a new operating system.

system input device: A device that is assigned to read a job input stream.

system library: A program library in auxiliary storage in which the various parts of an operation are stored.

system library device: An auxiliary storage device on which the system library is stored.

system macro instruction: An assembler language macro instruction that is provided by IBM or an installation staff, for general purpose use.

system management facilities (SMF): An optional control program feature that provides the means for gathering and recording information that can be used to evaluate system usage.

system monitor: (See monitor)

system output device: A device assigned to record output data for a series of jobs.

system programmer:

1. A programmer who is assigned to plan, generate, maintain, extend, and control the use of an operating system with the aim of improving the overall productivity of an installation.

2. A programmer who designs programming systems and other applications.

system queue area (SQA): An area in main storage adjacent to the fixed main storage area. The system queue area is set up by the nucleus initialization program (NIP) at the time of initial program loading.

system resdence volume: The volume that contains the IPL program, the volume index of the SYSCTLG data set, and the system data sets SYS1.NUCLEUS, SYS1.SVCLIB, and SYS1.LOGREC. The system residence volume must reside on the I/O device which is addressed when initial program loading is performed.

system utility device: A device that is assigned for the temporary storage of intermediate data for a series of job steps.

system utility programs: A collection of problem state programs that are provided with the operating system. The programs are designed for use by a system programmer in performing such functions as changing or extending the indexing structure of the system library catalog.

systems analyst: An expert on accounting, record keeping, and other business systems and practices, who formulates and plans data processing applications.

SYS1.DUMP: A system data set on which the system writes a core image dump when a system failure occurs.

SYS1.GENLIB: A data set, normally kept offline, used for system generation.

SYS1.LINKLIB: A system data set containing the system program modules that are neither permanently resident in main storage nor resident in the SYS1.SVCLIB.

SYS1.LOGREC: A system data set on the primary system residence device containing information regarding system failures.

SYS1.MACLIB: A system data set containing the macro definitions for the system macro instructions used by the assembler.

SYS1.MAN: A system data set on which System Management Facilities record the system, job, or job step information specified by an installation.

SYS1.NUCLEUS: A system data set on the primary system residence device containing the IPL program and the primary nucleus.

SYS1.PARMLIB: A system data set containing system parameter lists. If SYS1.PARMLIB is cataloged, it may reside on any volume in the system; if not cataloged, it is on the primary system residence volume.

SYS1.PROCLIB: A system data set containing cataloged procedures. The cataloged procedures can be either system tasks or jobs.

SYS1.SVCLIB: A system data set on the primary system residence device containing all of those SVC routines, machine check handler recovery management routines, and access method routines, that are not permanently resident in main storage.

SYS1.SYSJOBQ: A system data set used by the scheduler as a storage and work area for information about the input and output streams. Contains the input queue and output queue.

t

tape oriented system: A computing system that uses magnetic tape as auxiliary storage and for the bulk of input and output data.

task: An independent unit of work that can compete for the resources of the system.

task control block (TCB): The consolidation of control information related to a task.

task dispatcher: The control program routine that selects from the task queue the task that is to be performed by the central processing unit.

task management: In an MFT or MVT control program configuration, the part of the supervisor that controls and directs the concurrent performance of data processing tasks.

task queue: A queue of all the task control blocks present in the system at any one time.

TCAM: (See "Telecommunications Access Method.")

telecommunication lines: Telephone and other communication lines that are used to transmit messages from one location to another.

telecommunications: The transmission of messages from one loation to another over telephone and other communication lines.

Telecommunications Access Method (TCAM): The combination of an access technique and a given data set organization in a teleprocessing application that allows the programmer to transfer data between main storage and remote I/O devices.

Telecommunications Control Unit (TCU): An input/output control unit that addresses messages to and receives messages from a member of remote terminals.

teleprocessing: The processing of data that is received from or sent to remote locations by way of telecommunication lines.

terminal job: In TSO, a foreground job, a session from the time the terminal user logs on the system until he logs off the system. Also used to refer to the main storage region assigned to a user and associated system control blocks.

terminal user: (See "user.")

test translator: A facility that allows various debugging procedures to be specified in assembler language programs.

TESTRAN: A set of assembler language macro instructions that can be used to test programs written in the assembler language.

text:

1. The control sections of an object or load module.
2. The data to be sent in a telecommuncations message.

throughput: The total volume of work performed by a computing system over a given period of time.

time sharing: The concurrent sharing of the hardware and information resources of a data processing installation among two or more users who may or may not be located at remote terminals.

time sharing driver: A TSO addition to the dispatcher that determines which task is to execute next.

Time Sharing Option (TSO): An option of the operating system providing conversational time sharing from remote terminals.

time slice: A uniform interval of CPU time allocated for use in performing a task. Once the interval is over, CPU time is allocated to another task. Thus, a task cannot monopolize CPU time beyond a fixed limit.

time slicing: An optional feature of the MFT and MVT control program configurations that can be used to prevent each of several tasks in a group from monopolizing the CPU and thereby delaying the assignment of CPU control to other tasks in the group.

transactions: Business or other activities such as, sales, expenditures, shipments, reservations, and inquiries.

transaction data: Data produced as a result of a transaction. Such data is usually processed to update or interrogate records in a master file.

transient error: An error that occurs once or at unpredictable intervals.

transmission code: A code for sending information over communications lines.

transmit interruption: The interruption of a transmission from a terminal by a higher priority transmission to the terminal. Also called a "reverse break".

transmittal mode: The way the contents of an input buffer are made available to the program, and the way a program makes records available for output.

turnaround time: The elapsed time between submission of a job to a computing center and the return of results.

U

U format: A data set format in which blocks are of unknown length.

unit address: The three-character address of a particular device, specified at the time a system is installed. For example, 191 or 293. (See device type.)

user:

1. Anyone who requires the services of a computing system.
2. Under TSO, anyone with an entry in the User Attribute Data Set; anyone eligible to log on the system.

User Attribute Data Set (UADS): A partitioned data set with a member for each authorized system user. Each member contains the appropriate passwords, user identification, account numbers, LOGON Procedure names, and user characteristics defining the user profile.

USERID: (See "user identification.")

user identification (USERID): A one- to seven-character symbol identifying each system user.

utility programs: A collection of problem state programs that are designed to perform everyday tasks; such as transcribing data from one storage or input/output device to another. In the operating system there are three types of utility programs: data set, system, and independent.

V

V format: A data set format in which logical records are of varying length and include a binary length indicator; and in which V format logical records may be blocked, with each block containing a block length indicator.

volume: A section or unit of auxiliary storage space that is serviced by a single read/write mechanism whose operation is entirely independent of any other read/write mechanism.

volume table of contents (VTOC): A table of information in a direct access volume that defines the sets of data and unassigned space in the volume and indicates where they are located. The table of contents of a volume is used by the control program to locate a data set or assigned space on the volume.

W

wait condition: The condition of a task that needs one or more events to occur before the task can be ready to be performed by the central processing unit.

wait state: The state of the system when no instructions are being processed, but the system is not fully stopped. The system can accept I/O and external interruptions, and can be put through the IPL procedure.

OPERATOR'S REFERENCE MASTER INDEX

Indexes to systems reference library manuals are consolidated in the publication IBM System/360 Operating System: Systems Reference Library Master Index, GC28-6644. For additional information about any subject in this index, refer to other publications listed for the same subject in the Master Index.

ORM

Operator's Reference Master Index

ORMI

ORM

ORMI

ORMI

ORM

ORMI

ORMI

GC28-6691-2

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

READER'S COMMENT FORM

IBM System/360 Operating System:
Operator's Reference

Order No. GC28-6691-2

Please use this form to express your opinion of this publication. We are interested in your comments about its technical accuracy, organization, and completeness. All suggestions and comments become the property of IBM.

Please do not use this form to request technical information or additional copies of publications. All such requests should be directed to your IBM representative or to the IBM Branch Office serving your locality.

- Please indicate your occupation: ______________________________

- How did you use this publication?
  - ☐ Frequently for reference in my work.
  - ☐ As an introduction to the subject.
  - ☐ As a textbook in a course.
  - ☐ For specific information on one or two subjects.

- Comments (Please include page numbers and give examples.):

- Thank you for your comments. No postage necessary if mailed in the U.S.A.

GC28-6691-2

YOUR COMMENTS, PLEASE . . .

This manual is part of a library that serves as a reference source for systems analysts, programmers and operators of IBM systems. Your answers to the questions on the back of this form, together with your comments, will help us produce better publications for your use. Each reply will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Note: Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.

Cut Along Line

Fold Fold

FIRST CLASS
PERMIT NO. 81
POUGHKEEPSIE, N.Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY ...

IBM Corporation
P.O. Box 390
Poughkeepsie, N.Y. 12602

Attention: Programming Systems Publications
Department D58

Fold Fold

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
[U.S.A. only]

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
[International]