

# Honeywell



**LEVEL 64**

---

GCOS

---

**SYSTEM  
OPERATION  
OPERATOR GUIDE**

SERIES 60 (LEVEL 64)  
SYSTEM OPERATION  
OPERATOR GUIDE  
ADDENDUM B

SUBJECT

Modifications to the Existing Manual

SPECIAL INSTRUCTIONS

This is the second addendum to AQ13 Rev. 1, dated September 1978. Insert the attached pages in the manual according to the collating instructions. Change bars in the margin indicate technical changes or additions; asterisks indicate deletions.

SOFTWARE SUPPORTED

Release 0400 Level 64 GCOS

ORDER NUMBER

AQ13-01B

December 1979

COLLATING INSTRUCTIONS

To update this manual, remove old pages and insert new pages as follows:

<u>Remove</u>	<u>Insert</u>
3-13 through 3-16	3-13 through 3-16
6-05, 6-06	6-05, 6-06
6-11, 6-12	6-11, 6-12
6-15, 6-16	6-15, 6-16

## LEVEL 64 DOCUMENT LIST

Order Number	Title
AQ02	<i>Series 100 Program Mode Operator Guide</i>
AQ03	<i>Series 100 Conversion Guide</i>
AQ04	<i>Series 200/2000 Conversion Guide</i>
AQ05	<i>System 360/370 Conversion Guide</i>
AQ09	<i>System Management Guide</i>
AQ10	<i>Job Control Language (JCL) Reference Manual</i>
AQ11	<i>Job Control Language (JCL) User Guide</i>
AQ13	<i>System Operation Operator Guide</i>
AQ14	<i>System Operation Console Messages</i>
AQ18	<i>Operator Reference Manual</i>
AQ20	<i>Data Management Utilities Manual</i>
AQ21	<i>Series 200/2000 Program Mode User Guide</i>
AQ22	<i>Series 200/2000 Program Mode Operator Guide</i>
AQ26	<i>Series 100 File Translator</i>
AQ27	<i>Series 200/2000 File Translator</i>
AQ28	<i>Library Management Manual</i>
AQ40	<i>System 3 Conversion Guide</i>
AQ49	<i>Network Control Terminal Operation Manual</i>
AQ50	<i>Terminal Operations Manual</i>
AQ52	<i>Program Checkout Facility Manual</i>
AQ53	<i>Communications Processing Facility Manual</i>
AQ55	<i>TDS/64 Standard Processor Site Manual</i>
AQ56	<i>TDS/64 User Guide</i>
AQ57	<i>Standard Processor Programmer Reference Manual</i>
AQ59	<i>Unit Record Devices User Guide</i>
AQ63	<i>COBOL User Guide</i>
AQ60	<i>Interactive Operation Facility</i>
AQ64	<i>COBOL Language Reference Manual</i>
AQ65	<i>FORTTRAN Language Reference Manual</i>
AQ66	<i>FORTTRAN User Guide</i>
AQ67	<i>FORTTRAN Mathematical Library</i>
AQ68	<i>RPG Language Reference Manual</i>
AQ69	<i>RPG User Guide</i>
AQ72	<i>Series 200/2000 COBOL to Level 64 COBOL Translator</i>
AQ73	<i>IBM COBOL Translator</i>
AQ82	<i>BFAS User Guide</i>
AQ83	<i>HFAS User Guide</i>
AQ84	<i>UFAS User Guide</i>
AQ85	<i>Sort/Merge Manual</i>
AQ86	<i>Catalog Management Manual</i>
AQ87	<i>Library Maintenance User Guide</i>
AQ88	<i>I-D-S/II User Guide, Volume 1</i>
AQ89	<i>I-D-S/II User Guide, Volume 2</i>
AQ90	<i>COBOL Reference Card</i>
AQ92	<i>Operator's Reference Card</i>
AQ93	<i>RPG Reference Card</i>
AQ94	<i>FORTTRAN Reference Card</i>

## CONTENTS

Section I	Introduction.....	1-01
	How to Use this Manual.....	1-01
	Operator Responsibilities.....	1-02
Section II	Hardware.....	2-01
	The Processors.....	2-01
	The Central Processor Unit.....	2-01
	The Integrated Unit Record Processor .....	2-03
	The Integrated Mass Storage Processor.....	2-04
	The Operator Station.....	2-04
	The Visual Display Unit.....	2-05
	The Console Keyboard.....	2-05
	The Console Printer.....	2-05
	The Console Cassette Handler.....	2-16
	Peripherals.....	2-16
Section III	System Start-Up and Power Down.....	3-01
	System Operator Panel (SOP).....	3-01
	SOP Controls.....	3-02
	SOP Indicators.....	3-03
	Power.....	3-03
	Initialization.....	3-04
	Status.....	3-04
	Powering-Up the System.....	3-05
	Initialization.....	3-06
	Operator Dialog on the Console.....	3-07
	Message from GCOS to the Operator.....	3-07
	Operator Command.....	3-08
	Question With Immediate Reply.....	3-10
	Question With Deferred Reply.....	3-10
	Storage Load.....	3-11
	System Previous Status.....	3-13
	Question ILO1.....	3-14
	Date and Time.....	3-15
	To Indicate Resident Disks.....	3-16
	To Specify a General Handling of Telecom-	
	munications Messages Queues.....	3-16
	Dump Session.....	3-17
	SYSDUMP Utility.....	3-18
	Question ILO3.....	3-18
	Dumping to a Disk.....	3-19
	Question IL14.....	3-19
	Restore Session.....	3-20
	Release States.....	3-20
	Device Testing.....	3-22

Question DV42.....	3-22
Restart Session.....	3-23
Warm Restart.....	3-23
File Salvager.....	3-24
Question RRO1.....	3-25
Question OOU2.....	3-26
Cold Restart.....	3-27
Clean Restart.....	3-27
System Ready.....	3-28
Error Messages from GCOS Loading.....	3-28
GCOS Session Termination.....	3-31
Normal Termination.....	3-31
Abnormal Termination.....	3-31
Information Left Over at System Termination.....	3-32
System Reset.....	3-32
Powering Down the System.....	3-33
System Functions.....	3-34
Section IV	
Running User Jobs Under GCOS.....	4-01
User Jobs - General Concepts.....	4-02
The Job State Concept.....	4-03
Multiprogramming.....	4-05
The Job Class Concept.....	4-05
Priorities.....	4-06
Scheduling Priority.....	4-06
Dispatching Priority.....	4-06
Output Priority.....	4-07
DPR Use.....	4-07
Service Jobs.....	4-08
Job Introduction.....	4-10
The Input Reader.....	4-11
The Stream Reader.....	4-11
Stream Reader Execution.....	4-12
The JCL Translator.....	4-12
Hold State.....	4-13
Job Scheduling.....	4-13
Job Execution.....	4-15
Events During Execution.....	4-15
Step Execution.....	4-16
Step Repeat.....	4-18
Preinitialized Load Modules.....	4-19
Sysout Overflow.....	4-20
Suspended Jobs.....	4-21
Before Journal.....	4-22
After Journal.....	4-22
Backing Store.....	4-22
Thrashing.....	4-23
Job Termination.....	4-24
Output Writer.....	4-25
Standard and Permanent Outputs.....	4-25
Output Names and Sequence Numbers.....	4-26
Output Priorities.....	4-28
Output Class.....	4-28
Output Selection.....	4-28
Question OOU2.....	4-29
Output Writer Incidents.....	4-30

	Incidents on the Output Device.....	4-30
	Incidents on the Volume Containing the Output File.....	4-31
	Paper Form or Card Deck Cannot be Used.....	4-32
	The Sysout File cannot be Used.....	4-32
	The Sysout File is Incorrect.....	4-32
Section V	Peripheral Operations Under GCOS.....	5-01
	Device Handling Overview.....	5-01
	Device states.....	5-03
	Device Allocation in Device Mode.....	5-04
	Unknown Device.....	5-04
	Disk, Magnetic Tape and Cassette Operations....	5-05
	Media Characteristics.....	5-05
	Disk Packs.....	5-05
	Native Disks.....	5-05
	Non-Native Disks.....	5-06
	Magnetic Tape Reels.....	5-07
	Native Tapes.....	5-07
	Tapes Without a Native Label.....	5-08
	Cassettes.....	5-09
	Disks, Tapes and Cassettes : Premounting, Mounting and Dismounting.....	5-10
	Premounting Disk and Tape Volumes.....	5-10
	Normal Mounting of Disk, Tape and Cassette Volumes.....	5-11
	Disk and Tape Dismounting.....	5-12
	Disk and Tape Incidents.....	5-12
	Input/Output Operation Incidents.....	5-13
	Transition to Standby or Power Off.....	5-13
	Write Protect on Disk.....	5-14
	Write Protect on Magnetic Tape.....	5-16
	Printer Operations.....	5-16
	Paper Form Operations.....	5-17
	Paper Premounting.....	5-17
	Paper Mounting.....	5-18
	Print Test.....	5-19
	Print Belt Operations.....	5-19
	Print Belt Premounting.....	5-20
	Print Belt Mounting.....	5-20
	Printer Incidents.....	5-21
	Card Device Operations.....	5-22
	Card Deck Premounting.....	5-22
	Card Deck Mounting and Device Modes.....	5-23
	Card Device Incidents.....	5-24
	Hardware Incidents.....	5-25
	Device Incidents.....	5-26
	Peripheral Processor Incidents.....	5-26
	Error Logging.....	5-27
	Monitoring Device Incidents (Device Trace)....	5-29
Section VI	Operator Command Language.....	6-01
	Broadcast Telecommunication Command.....	6-03
	Cancel Load Module Command.....	6-04
	Cancel Main Memory Command.....	6-05
	Cancel Output Command.....	6-06

Cancel Request Command.....	6-07
Display Backing Store Command.....	6-08
Display Device Command.....	6-09
Display Device Trace Command.....	6-11
Display Dispatching Priority Command.....	6-12
Display Load Module Command.....	6-13
Display Log Command.....	6-14
Display Main Memory Command.....	6-15
Display Output Command.....	6-16
Display Output Class Command.....	6-19
Display Scheduling Command.....	6-20
Display Time Command.....	6-24
End Session Command.....	6-25
Force Job Command.....	6-26
Force Output Command.....	6-27
Hold Device Command.....	6-28
Hold Job Command.....	6-29
Hold Output Command.....	6-30
Modify Class Command.....	6-31
Modify Device Trace Command.....	6-32
Modify Dispatching Priority Command.....	6-33
Modify Job Command.....	6-34
Message of Today Command.....	6-36
Modify Log Command.....	6-37
Modify Output Command.....	6-38
Modify Output Class Command.....	6-39
Modify Scheduling Command.....	6-40
Name Volume Command.....	6-41
Preinitialize Load Module Command.....	6-42
Preload Main Memory Command.....	6-43
Release Device Command.....	6-44
Release Job Command.....	6-45
Release Output Command.....	6-46
Start Class Command.....	6-47
Start Device Trace Command.....	6-48
Start Input Reader Command.....	6-49
Start Job Command.....	6-50
Start Output Writer.....	6-51
Start Telecommunications Command.....	6-52
Terminate Class Command.....	6-53
Terminate Device Trace Command.....	6-54
Terminate Input Reader Command.....	6-55
Terminate Job Command.....	6-56
Terminate Output Writer Command.....	6-57
Terminate Telecommunications Command.....	6-58
Use Catalog Command.....	6-59
Use Device Command.....	6-60
write Log Command.....	6-61

Section VII Communications.....	7-01
Starting Communications Activities.....	7-03
Control of Communications.....	7-04
Terminating Communications.....	7-04
BT.....	7-06
DT.....	7-06
HT.....	7-06



MTE.....	7-06
MTL.....	7-06
MTP.....	7-07
MTT.....	7-07
RT.....	7-07
ST.....	7-07
IT.....	7-07

## ILLUSTRATIONS

Figure 2-1	Level 64 Operator Station.....	2-02
Figure 2-2	Console Keyboard, Alphabetic Keys .....	2-06
Figure 2-3	Console Keyboard, Control Keys.....	2-07
Figure 2-4	Console Keyboard, Numeric and Graphic Keys .....	2-08
Figure 2-5	Console Printer (Version C).....	2-10
Figure 2-6	Console Printer (Version D).....	2-12
Figure 2-7	CRT With Metering Box.....	2-14
Figure 2-8	Console Cassette Handler.....	2-17
Figure 3-1	The System Operator Panel.....	3-02
Figure 4-1	Job States.....	4-03
Figure 5-1	Summary of Disk Categories.....	5-06
Figure 5-2	Summary of Magnetic Tape Categories.....	5-08
Figure 5-3	Summary of Cassette Categories.....	5-09

## TABLES

Table 2-1	Console Keyboard Control Keys.....	2-09
Table 2-2	Console Printer Controls (Version C).....	2-11
Table 2-3	Console Printer Controls (Version D).....	2-13
Table 2-4	CRT Controls.....	2-15
Table 2-5	Cassette Handler Controls.....	2-18
Table 4-1	Stages in the Life of a Job.....	4-04
Table 4-2	Service Job Characteristics.....	4-09
Table 4-3	Output Commands.....	4-27



## SECTION I

### INTRODUCTION

This section describes firstly the way in which this manual can best be used and secondly the responsibilities of the Level 64 operator with regard to the system.

#### HOW TO USE THIS MANUAL

All users of this manual should study at an early stage:

- . The opening paragraphs of Section IV which introduce the concepts and terminology relating to the running of jobs.
- . The opening paragraphs of Section V which introduce the concepts and terminology relating to peripheral operations under GCOS.

The above is essential information for all readers. Further advice can only be given according to the type of reader:

1. The general reader (including operators who are training prior to starting work on a Level 64 site).

Each subject is described in general terms and then in detail: the reader may choose to skip the details.

2. The on-site operator. Here the need is for swift access and the start-point will almost always be an OCL (Operator Control Language) command, which are covered in Section VI.

For console messages, see the manual "System Operation: Console Messages".

## OPERATOR RESPONSIBILITIES

The responsibilities of the Level 64 operator can be summarized as follows:

1. Powering up the hardware of the system.
2. Performing Hardware Initialization: loading the firmware from disk.
3. Performing Storage Load : loading the Operating System, GCOS, from disk.
4. Performing GCOS Initialization : here the operator has to decide whether to take a dump listing of the contents of memory stored at the time of the previous system termination. He may have to perform a Restore session in order to create a new (or re-create the old) version of GCOS. He may also, if he is restarting after a fatal system incident, have to decide whether to attempt to rerun certain user jobs.
5. Initiation and monitoring jobs from submission to termination. This includes:
  - . Introduction of jobs in the correct order according to rules laid down within the installation.
  - . Ensuring, wherever possible, that the machine has a good mix of work to do.
  - . Control of multiprogramming and the use of system resources by means of the various OCL commands.
  - . Control of the printed and punched output from jobs which is "spooled" until the operator initiates the Output Writer.
  - . Dispatching the output to the user.
6. Performing the necessary peripheral operations. This includes mounting disk packs, magnetic tape reels, card decks, printer stationary, print belts etc. when requested by the system. Alternatively the operator can anticipate these system requests by "premounting" a medium which he knows is about to be requested.
7. [Under Operations Management supervision] Controlling certain system files, both through the use of OCL commands and by initiating utilities which deal with these files. (There is, for example, a utility which lists the contents of the System Error Log.)

8. Attempting to recover from any hardware or software incidents by following the procedures described in this manual and in the "System Operation: Console Messages" manual. The operator should also note down the circumstances of the incident, particularly any error indications on the System Operator Panel.
9. Closing down and powering off the system.

Note : To the above list must be added any installation-defined responsibilities concerning, for example, the control of disks and magnetic tapes, stationery etc.



## SECTION II

### HARDWARE

Level 64 of the Series 60 is a full Range System consisting of one basic central processor which can be expanded in main memory size, peripheral complement and processor power. In the early product life of Level 64 it consisted of five submodels, the 64/20, 64/30, 64/40, 64/50 and 64/60, which are still in operation today. Although the Level 64 system provides a wide range of power and configurability, operation remains the same at all performance levels and on all models.

The Level 64 central subsystem integrates the processors, the operator station and their associated power supplies.

For more detailed information, refer to the "System Operation: Central Subsystem Operation" manual

#### THE PROCESSORS

The Level 64 processors, integrated in a single unit, are:

- . The Central Processor Unit
- . Integrated Unit Record Processor
- . The Integrated Mass Storage Processor
- . The Magnetic Tape Processor (which is optional)

#### The Central Processor Unit (CPU)

The CPU comprises:

- . The Control Store, which contains microprograms to perform logic functions
- . The Arithmetic Logic Unit, which initiates all system activities including microprogram execution, memory access, and data transfers

- The Channel Buffering Unit, which provides the input-output interface between the CPU and the peripheral processor subsystem

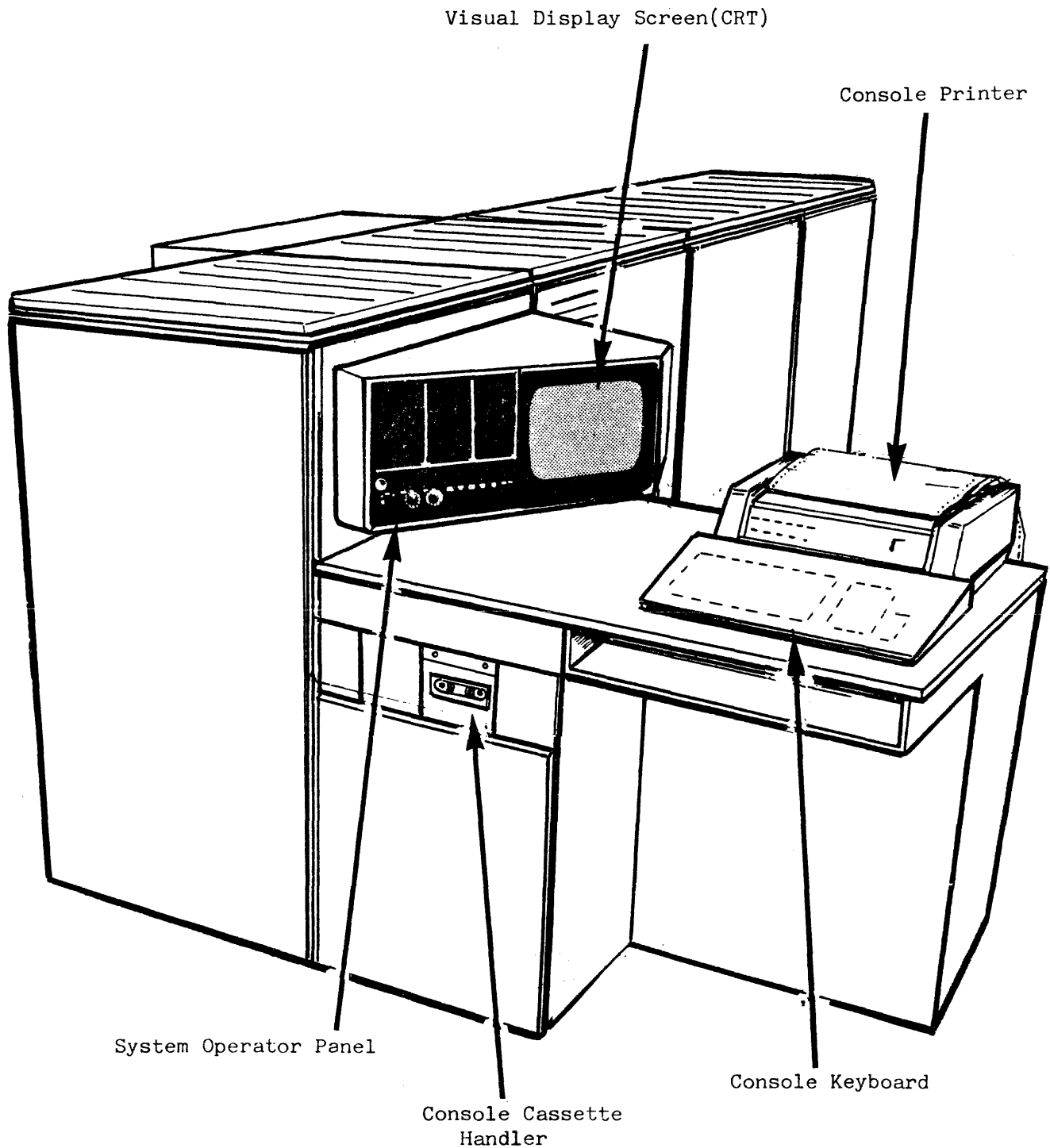


Figure 2-1. Level 64 Operator Station



These three units described above interact to perform the following CPU functions:

- . Addressing the memory subsystem for storing and retrieving data
- . Performing arithmetic and logical operations on the data
- . Controlling the sequence of the various system and user processes
- . Initiating data transfers between the memory subsystem and other parts of the system (or vice-versa)
- . Performing the various automatic integrity and status checks which ensure efficient and error-free operations

The Level 64 Memory Subsystem is capable of:

- . Memory cycle time of from .74 to 1.0 microseconds per 4-byte word
- . Automatic correction of any single-bit error (hamming code)
- . Several retries to correct a double-bit error before flagging the system software
- . Reconfiguration to bypass a faulty memory module

#### The Integrated Unit Record Processor (IURP)

The main function of the IURP is to control the following equipment:

- The Operator Station
  - System Operator Panel (SOP)
  - Console
  - Console Cassette Handler
- The Line Printer
- The Card Reader and Punch
- The Data Communication Controller and Terminals
- The Paper Tape Reader and Punch
- The Document Reader and Sorter

Much of the routine burden of message handling, encoding and decoding is performed by the IURP. This leaves the CPU free to perform more useful work.

## The Integrated Mass Storage Processor

The Integrated Mass Storage Processor controls the mass storage units.

### THE OPERATOR STATION

A brief description of the Operator Station is given below. For more detailed information refer to the "System Operation: Operator Station" manual.

The Operator Station (see Figure 2-1) allows the operator to initiate dialog with the system.

The term dialog is associated with the following functions:

- . System activation to set the system fully operational
- . Indications of the system state for all phases of activation
- . Input facility for entering information to the system
- . Output facility for displaying both information generated by software and information entered as input
- . Diagnostic capability to test the subsystem and device operability

The Operator Station consists of:

- . System Operator Panel
  - Control Panel
  - Visual Display Unit (optional)
- . Console
  - Console Keyboard
  - Console Printer
- . Console Cassette Handler

The two types of information display available at the Operator Station are:

- . Hard Copy, console listing on the console printer
- . Soft Copy, temporary display on the visual display unit

## The Visual Display Unit

The screen of the visual display unit is referred to as the CRT (Cathode Ray Tube).

The CRT provides a display of 24 lines of 64 characters per line.

(See Figures 2-7 and Table 2-4).

The first displayable character appears in the leftmost free column, usually column 1, of the bottom line, line 24.

When the entire line is filled, it is displaced upwards to the next uppermost adjacent line, line 23. The procedure is repeated until the entire screen is now filled, i.e. the original entry in line 24 is now displaced upwards to line 1.

Subsequent line entries from now on result in roll-up by which process the top line is lost to allow for the entry line, line 24 to accommodate further text.

## The Console Keyboard

The console keyboard is similar to a typewriter keyboard with the addition of certain control keys.

The keyboard is shown with the different keys highlighted below.

The function of these control keys, which are described fully in the "System Operation: Operator Station" manual, are shown in Table 2-1.

## The Console Printer


The Level 64 console printer is a 30 character-per-second serial printer. The character set contains 68 printable characters and the printer allows up to 80 characters per line.

The console printer controls are shown in Tables 2-2 and 2-3.


The two different versions of the console printer are shown in Figures 2-5 and 2-6.


**LEGEND**


 set of Control keys

 set of Graphic keys

*NOTE: The character SPACE is represented by a number of symbols, the most common of which are:*

 delta inverse

 *s.* circumscribed

 lower case *b.* slant overprint

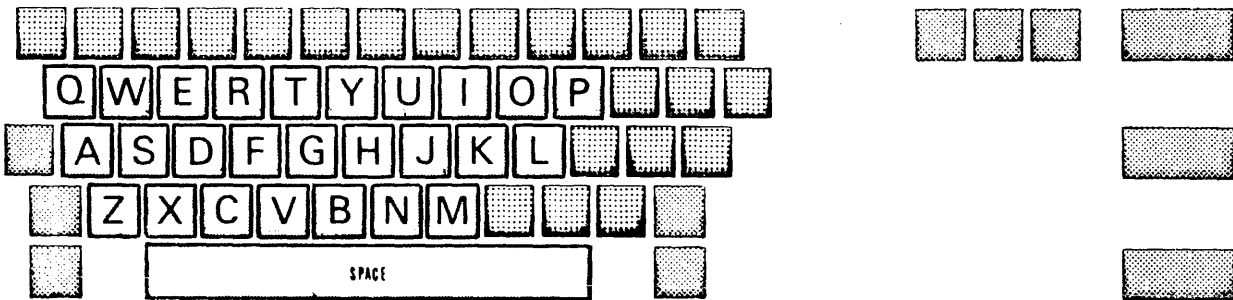


Figure 2-2. Console Keyboard, Alphabetic Keys

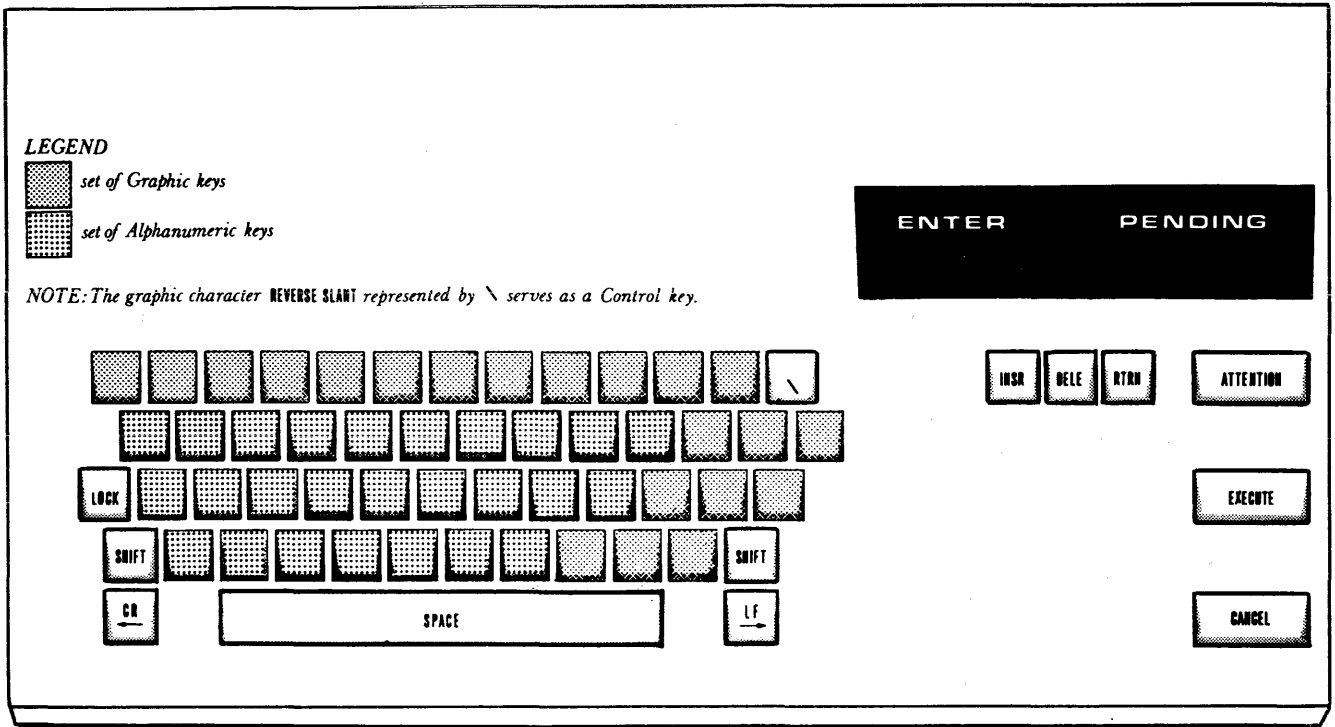


Figure 2-3. Console Keyboard, Control keys

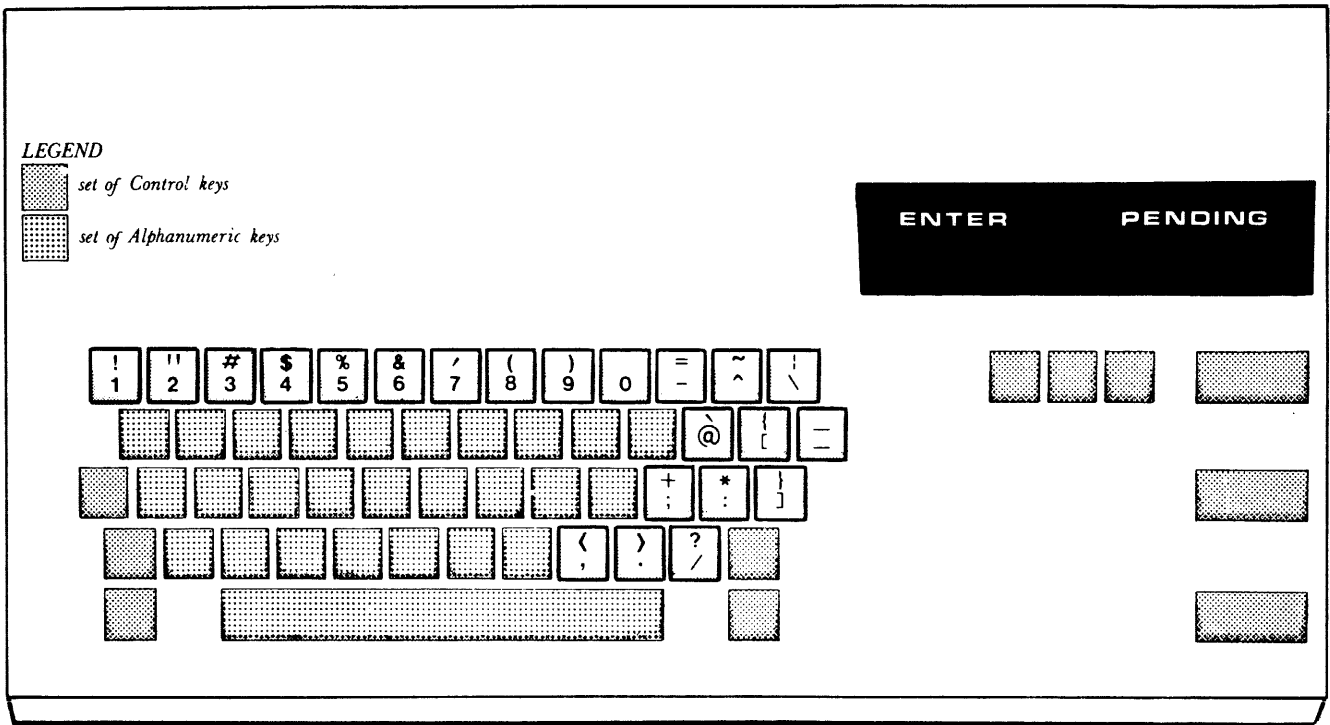


Figure 2-4. Console Keyboard, Numeric and Graphic Keys

Table 2-1. Console Keyboard Control Keys

Keyboard Control	Type/Function
. ATTENTION	Touch-key/System - Generates system interrupt
. CANCEL	Touch-key/Edit - Cancels message keyed in
. CR (Carriage Return)	Touch-key/Edit: Visual Display - Moves cursor 1 character position left  : Serial Printer - Next character at start of same line
. DELE (DELEte)	Touch-key/Edit: Visual Display - Deletes character overlaid by cursor - Moves left characters right of delete
. ENTER	Indicator/System - Interrupt to System acknowledged
. EXECUTE	Touch-key/System - Transmits message to system
. INSR ( INSeRt)	Touch-key/Edit: Visual Display - Moves right characters overlaid by cursor
. LF (Line Feed)	Touch-key/Edit: Visual Display - Moves cursor 1 character position right  : Serial Printer - Causes 1 line skip without return
. LOCK	Shift Lock/Mechanical - Locks SHIFT key in upper case select
. PENDING	Indicator/System - Request for system interrupt
. REVERSE SLANT (See Graphic Keys)	Graphic Key/Edit (Symbol \) - Deletes preceding character printed
. RTRN (ReTuRN)	Touch-key/Edit: Visual Display - Returns cursor to original position
. SPACE BAR	Touch-key/Edit or Data - Generates non-printable "spaces"
. SHIFT	Graphic Select/Data - Selects upper case/lower case graphic

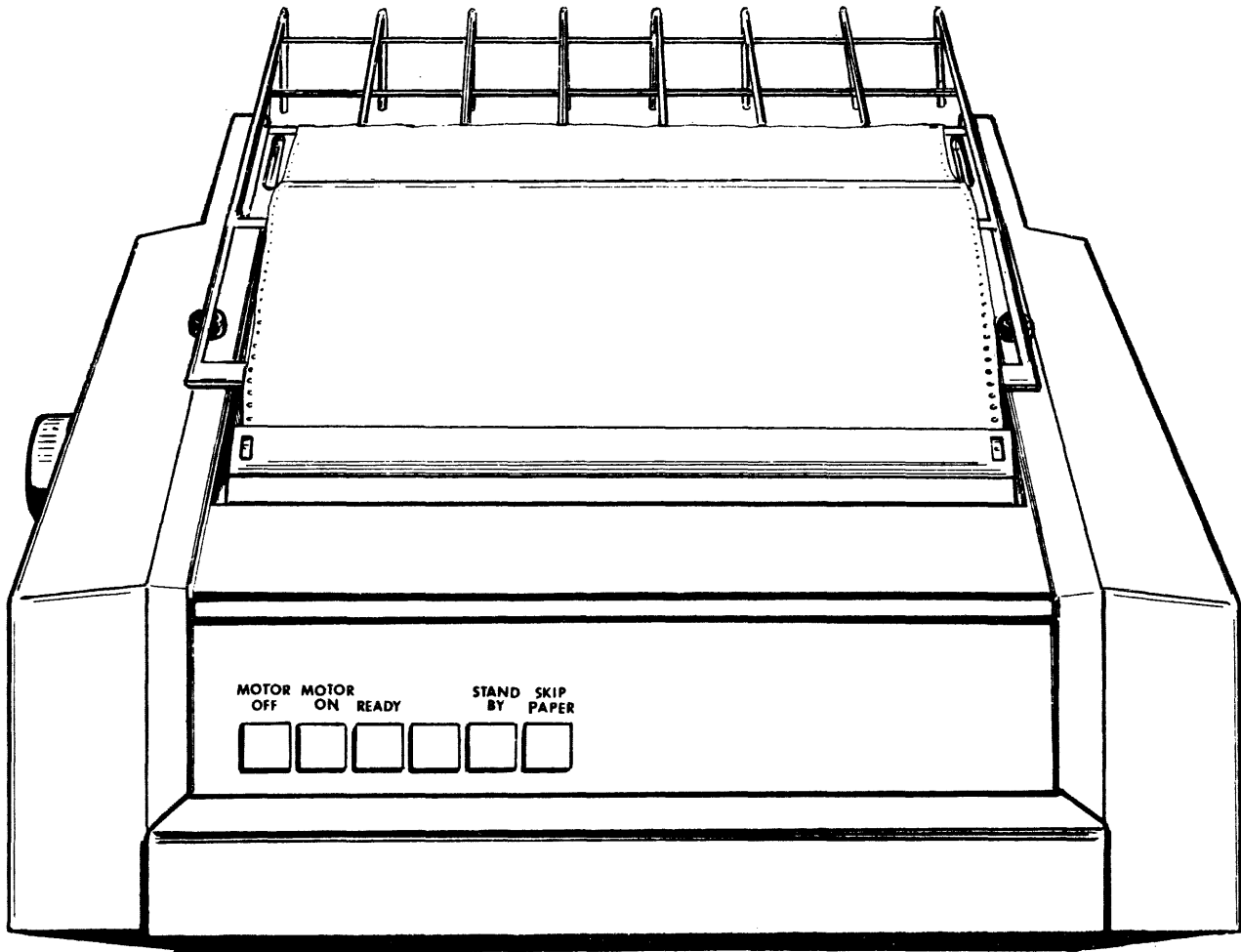


Figure 2-5. Console Printer Version C



Table 2-2. Console Printer Controls (Version C)

Control	Type/Function
. Master Switch	Saddle Breaker at rear of console printer - Connects/Disconnects Power Supply to the console printer
. MOTOR OFF	Indicator/Pushbutton - Removes power from the console
. MOTOR ON	Indicator/Pushbutton - Applies power to the console
. READY	Indicator Console is mechanically operational - No malfunctions have been detected - No operator intervention required  NOTE : READY state of the console is independent of SYSTEM INITIALIZATION
. SKIP PAPER	Pushbutton - Advances paper without printing
. STANDBY	Indicator Operator intervention required - Paper Low detected - Shield Up - Undervoltage * - Cover Interlock tripped
* Not operator-rectifiable	

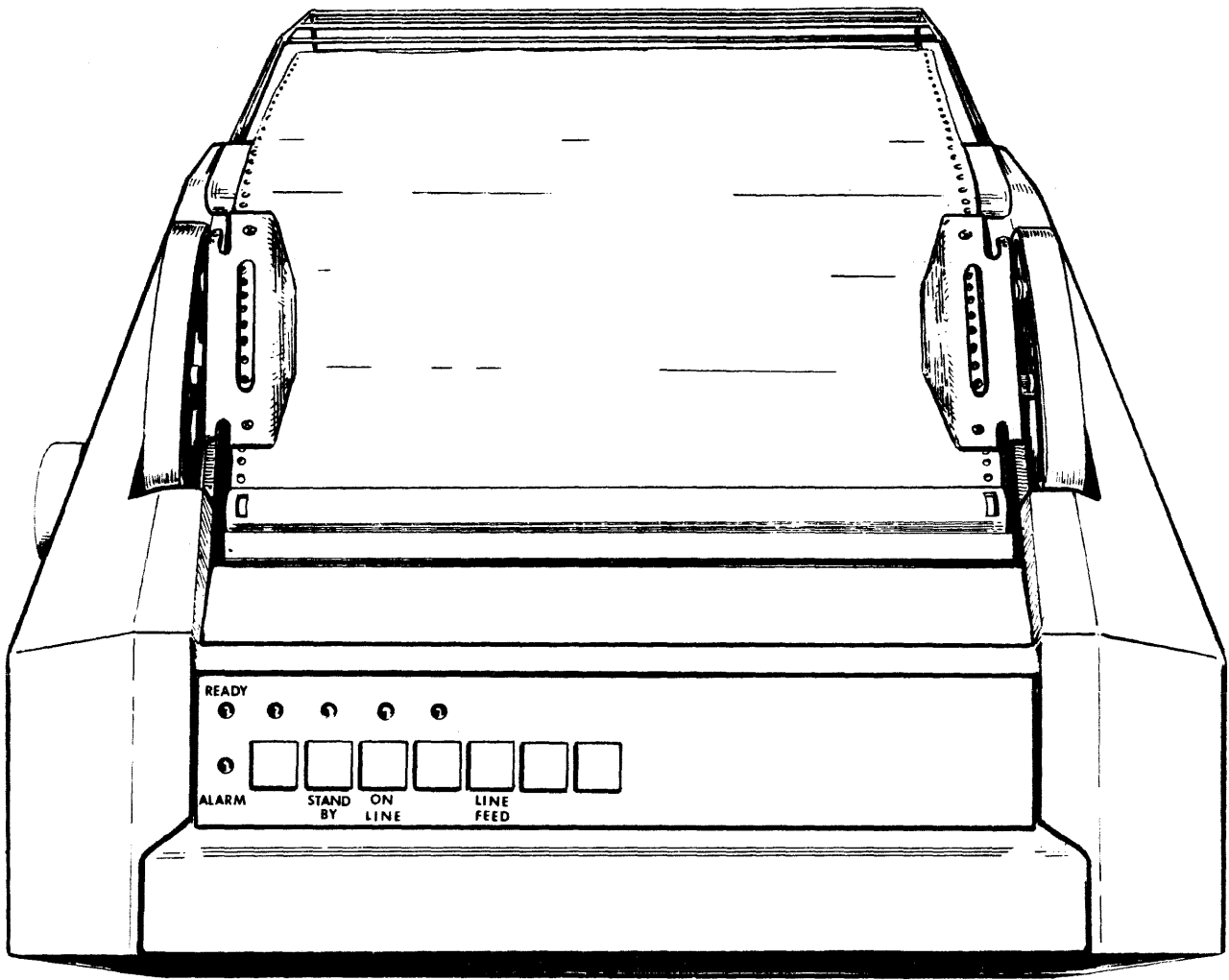


Figure 2-6. Console Printer Version D

Table 2-3. Console Printer Controls (Version D)

Control	Type/Function
. Master Switch	Saddle Breaker at rear of console printer - Connects/Disconnects Power Supply to the console printer
. ALARM	Indicator Operator intervention required - Paper Low detected - Shield Up - Undervoltage, see Note 2 - Cover Interlock tripped
. LINE FEED	Pushbutton - Advances paper without printing
. ON LINE	Indicator/Pushbutton - Applies power to the console
. READY	Indicator Console is mechanically operational - No malfunctions have been detected - No operator intervention required  NOTE : READY state of the console is independent of SYSTEM INITIALIZATION
. STANDBY * (see Note 1)	Indicator/Pushbutton - Removes power from the console
<p>NOTE: 1. The STANDBY indicator/pushbutton for Version D is denoted as "STANDBY *" to distinguish it from the STANDBY indicator for Version C.</p> <p>2. Undervoltage is not operator-rectifiable.</p>	

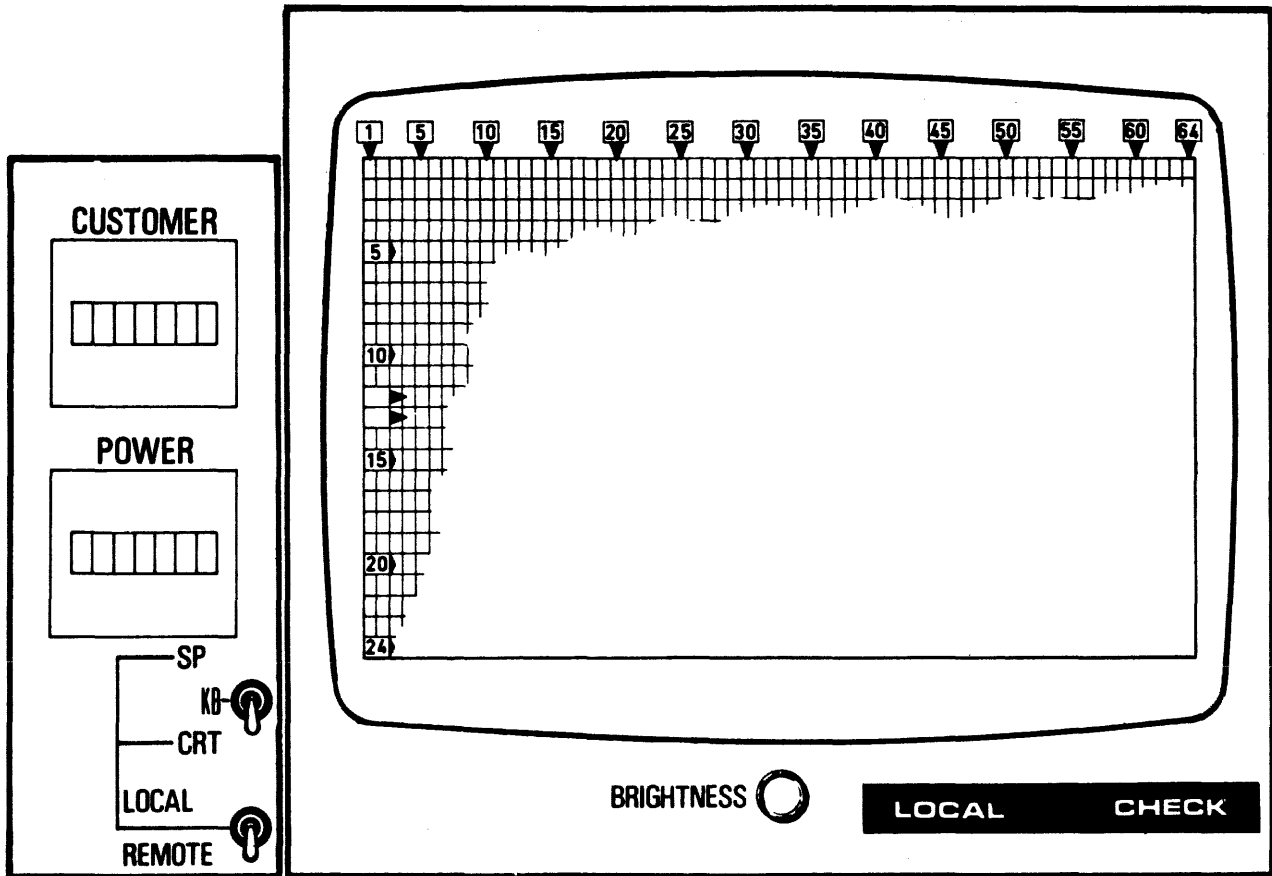


Figure 2-7. CRT With Metering Box (at left)

Table 2-4. CRT Controls

Control Name	Type of Control	Function
. BRIGHTNESS	Dial	Adjusts CRT brightness/contrast
. CHECK	Indicator	If remains illuminated, denotes malfunction in the visual display unit
(CRT)		See SP/CRT KB
. Customer	7-digit counter (upper)	Register I-O activity in hours
(KB)		See SP/CRT KB
. LOCAL	Indicator	LOCAL/REMOTE "lock" toggle switch is slotted in LOCAL
. LOCAL/REMOTE	"Lock" Toggle Switch (lower)	Enables (REMOTE) or disables (LOCAL) central subsystem control of the console
. Power	7-digit counter (lower)	Registers in hours the time in which the central subsystem has been successfully powered up
(REMOTE)		See LOCAL/REMOTE
. SP/CRT KB	"Lock" Toggle Switch (upper)	Enables either the CRT or the serial printer to be used as the principal console when slotted in the position indicated

## The Console Cassette Handler

The console cassette handler provides for the diagnostic capability of the system. (Figure 2-8)

The operator interface with the cassette handler is as a "read only" device to load software specific to launch checkout routines and to perform the System Check Report (SCR).

Accessible parts of the cassette handler are,

- . Handler Panel
- . Operator Panel

The handler panel comprises the shutter assembly and pushbuttons.

Pushbuttons above the shutter allow mechanical handling of the device and medium.

The operator panel has a set of indicators for the various states of the handler and the cassette (see Table 2-5).

## PERIPHERALS

Full operating details for each of the peripherals available on Level 64 can be found in the respective peripheral operation manual.

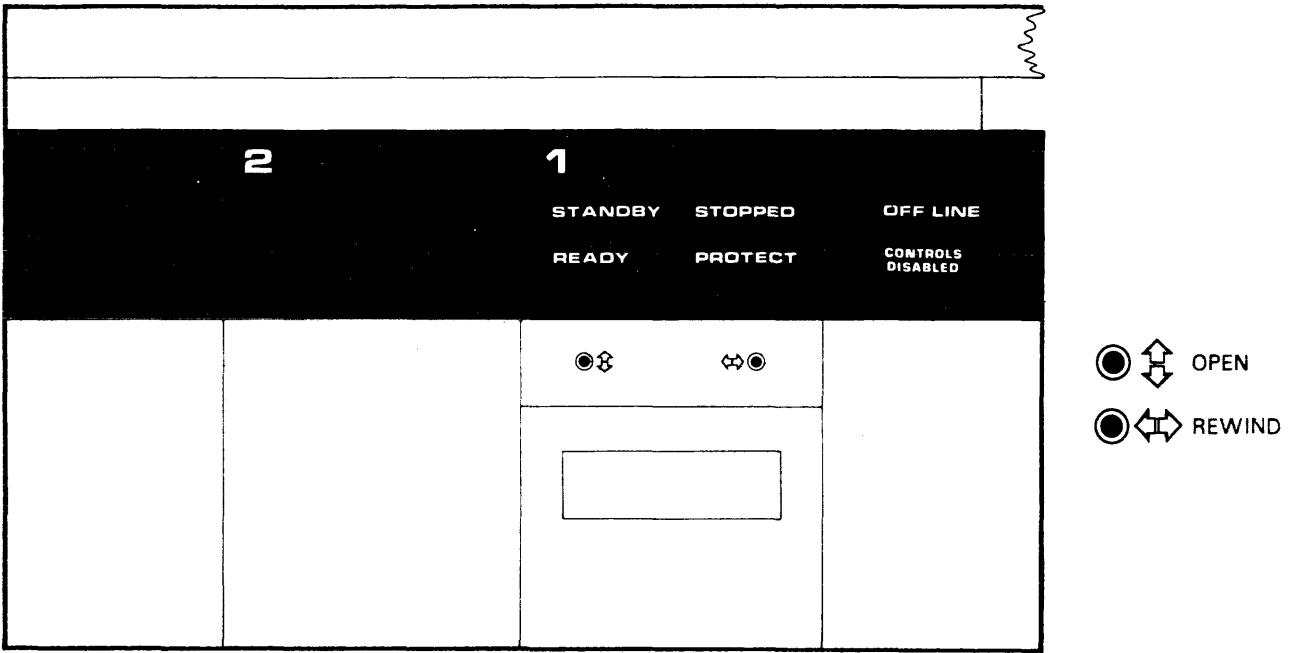


Figure 2-8. Console Cassette Handler

Table 2-5. Cassette Handler Controls

Control	Type/Function
. CONTROLS DISABLED	Indicator Handler controls are impaired - REWIND is not operational - OPEN may allow access to shutter
. OFF LINE	Indicator Handler is not available to the central subsystem NOTE : This condition is present before SYSTEM INITIALIZATION
. OPEN	Pushbutton Allows shutter to open for access to cassette medium
. PROTECT	Indicator WRITE ENABLE plugs on cassette medium have been removed
. READY	Indicator Handler is now available to the central subsystem, i.e. after SYSTEM INITIALIZATION - Cassette tape at BOT - Shutter has been closed securely - No malfunctions have been detected
. REWIND	Pushbutton Rewinds tape to the transparent leader
. STANDBY	Indicator Handler is not connected logically to the central subsystem Cassette may now be changed
. STOPPED	Indicator Movement on cassette tape has stopped - REWIND is now completed - Tape is at the transparent leader



## SECTION III

### SYSTEM START-UP AND POWER DOWN

This section describes Level 64 operations from System Start-Up until the SYSTEM READY message is displayed on the console (this message denotes that user jobs may now be entered and run) until the powering down of the system.

The different phases of system start-up and close-down are:

- . Power-Up
- . Operator Dialog
- . Initialization
- . Storage Load
- . Power Down

However, the first subsection below introduces the System Operator Panel (SOP). It is described here because it is during the period of system start-up that the operator's need to use and to understand the SOP is greatest.

#### SYSTEM OPERATOR PANEL (SOP)

The System Operator Panel is shown in Figure 3-1. The function of the SOP is twofold:

1. To provide the operator with a set of controls and selector switches for starting up, restarting and closing down the system.
2. To display by means of illuminated indicators the current system status, especially any errors or abnormal conditions in the hardware which require operator action. The SOP is discussed below under 2 headings:
  - . SOP Controls
  - . SOP Indicators

For more detailed information, including recognition of error conditions, see "System Operation: Operator Station".

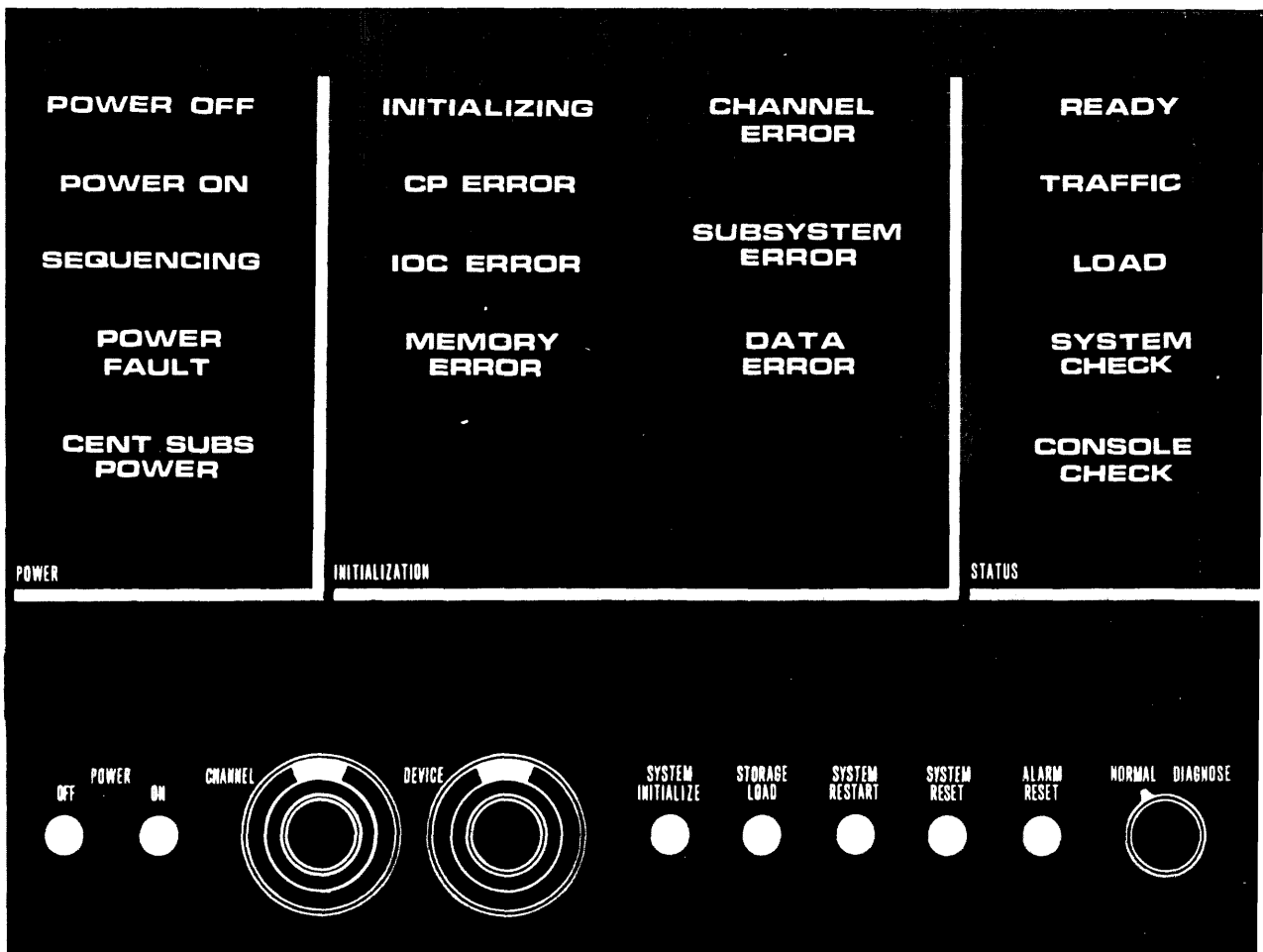


Figure 3-1. The System Operator Panel

### SOP Controls

Various controls are shown at the bottom of Figure 3-1. The precise circumstances in which they are used are discussed in the appropriate places elsewhere in this manual, but their basic functions are as follows:

- POWER ON** This pushbutton initiates the DC power-up sequence. (AC power from the mains supply must already be present).
- POWER OFF** This pushbutton initiates DC power-down.
- CHANNEL** This rotary switch enables the operator to select the subsystem from which hardware initialization and storage load are to be performed. Channel 0 is used for disk. Channel 2 for magnetic tape.
- DEVICE** This rotary switch enables the operator to select the device from which hardware initialization or storage load is to be performed. e.g. Channel 0, Device 2 is the 2nd disk transport.

SYSTEM INITIALIZE	This pushbutton causes hardware initialization to be performed and the system firmware to be loaded from the device specified on the rotary switches.
STORAGE LOAD	This pushbutton causes the system software to be loaded or reloaded from the device specified by the two rotary switches described above.
SYSTEM RESTART	This pushbutton should not normally be used. Its function is to cause a deliberate system crash and a consequent dump of memory to disk. It must always be preceded by SYSTEM RESET and it should only be used in cases where operations management suspects that the system is functioning incorrectly and requires a dump to give to Field Engineering.
ALARM RESET	This pushbutton permits a visual test of the SOP bulbs by causing all the SOP indicators (except for the Power indicators) to illuminate. Its other function is to switch off and reset the system alarm buzzer which warns the operator when certain exception conditions occur.
NORMAL/DIAGNOSE	For normal running this 2-way switch is always set to NORMAL. DIAGNOSE transfers control to the Service and Unit Record Processor which may then perform various diagnostic tests on the system.

### SOP Indicators

There are 17 indicators on the SOP (illustrated in Figure 3-1). Their significance when illuminated is described below under the headings Power, Initialization and Status.

#### Power:

POWER OFF	Indicates that AC power is present in the system but DC power is not yet applied.
POWER ON	Indicates that DC power is present in the system.
SEQUENCING	Indicates that the power-up sequence is being performed. If it fails to extinguish after a reasonable time, a fault has occurred which prevents the sequence from terminating.
POWER FAULT	Indicates fault in power supply.
CENT SUBS POWER	Indicates that the central subsystem has been successfully powered up.

Initialization:

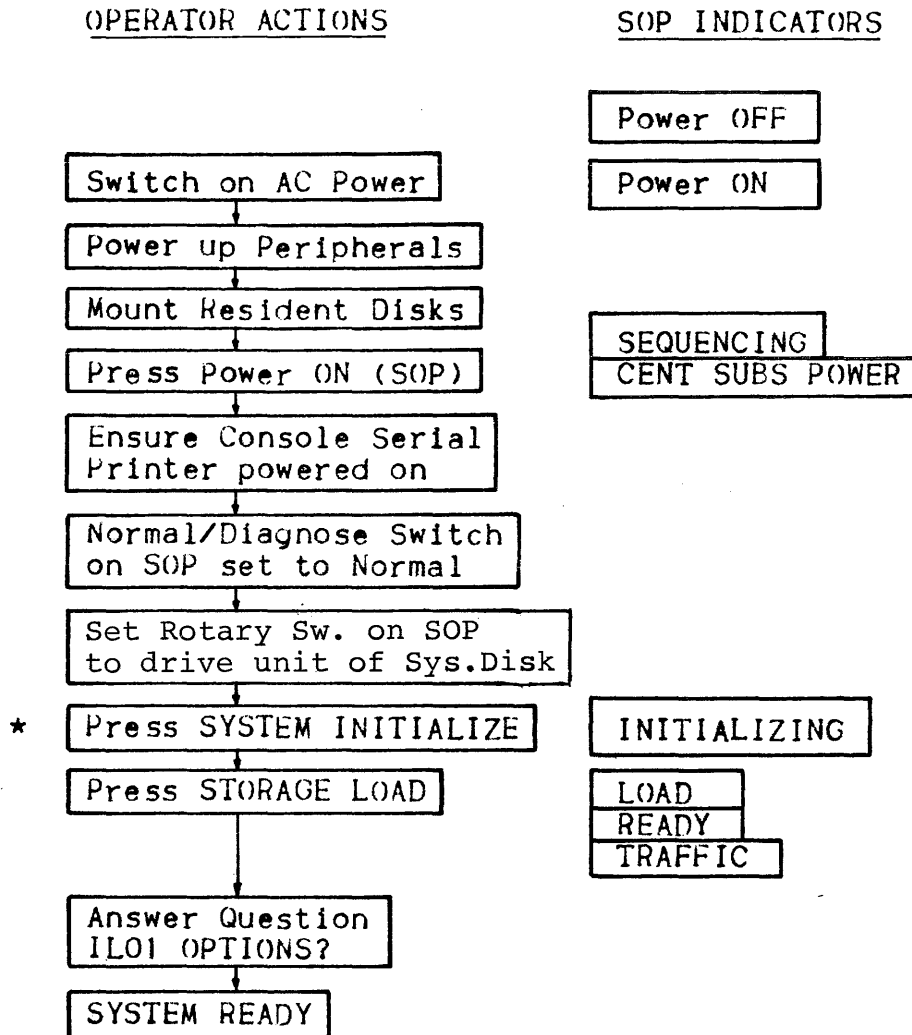
- INITIALIZING Indicates that hardware initialization is being performed.
- CP ERROR Indicates a Central Processor error.  
[Note: It is temporarily lit during each hardware initialization when the system circuits are tested.]
- IOC ERROR Indicates an error in the Input/Output Controller (but Note on CP ERROR also applies here).
- MEMORY ERROR Indicates an error in the memory subsystem (but Note on CP ERROR also applies here).
- CHANNEL ERROR Indicates that the channel selected (by rotary switch on SOP) for hardware initialization or storage load is inoperative.
- SUBSYSTEM ERROR Indicates that an error exists in one of the subsystems.
- DATA ERROR Indicates that a fault exists either on the device (e.g. disk drive), or the volume (the disk pack itself) which is being used for either hardware initialization or storage load.

Status:

- READY Indicates that Storage Load has been successfully completed. It remains lit during normal running.
- TRAFFIC Indicates that a Central Processor or Input/Output activity is occurring.
- LOAD Indicates that hardware initialization is complete and that the operator may proceed with Storage Load.
- SYSTEM CHECK Indicates an irrecoverable error, normally during hardware initialization or Storage Load. If this indicator is lit it must be followed by SYSTEM INITIALIZE
- CONSOLE CHECK Indicates an error in the Unit Record subsystem.

## POWERING UP THE SYSTEM

The operator actions required to power up the system are set out below. The SOP indicators are shown on the right, at the time when they will illuminate during the powering-up procedure.



\* When performing SYSTEM INITIALIZE, if LOAD and CHANNEL ERROR, SUBSYSTEM ERROR or DATA ERROR are illuminated, press the following pushbuttons:

SYSTEM RESET  
STORAGE LOAD  
SYSTEM INITIALIZE  
STORAGE LOAD

If DATA ERROR or DATA ERROR and LOAD are still illuminated, check that the system disk (or the correct system disk) is mounted.

## INITIALIZATION

INITIALIZATION checks the system components and the identity of all the hardware elements, and in doing so takes the system from the POWER ON state to the LOAD state. To do this the operator has to:

1. Mount the system disk on a drive which is to be used as the LOADER and set the drive to READY.
2. On the SOP, set the channel selector switch to 0 and the device selector switch to the device number indicated on the LOADER disk drive.
3. Press the SYSTEM INITIALIZE pushbutton.

Each of the following illuminates and extinguishes in turn:

CP ERROR

MEMORY ERROR

IOC ERROR

and a question is asked:

DO YOU RUN OS ?

The operator replies YES to this question and the LOAD indicator illuminates, indicating that the hardware initialization has been completed, and that GCOS loading can now be performed.

GCOS loading is performed by pressing the STORAGE LOAD pushbutton (LOAD is extinguished, READY is illuminated); the operation of loading GCOS is described under Storage Load in this section. The next part of this section deals with the types of dialog between GCOS and the operator.

## OPERATOR DIALOG ON THE CONSOLE

Four types of dialog can occur during a GCOS session:

### 1. MESSAGE FROM GCOS TO THE OPERATOR

Messages will take one of two forms (question messages are described below).

The first form is normally just informative, the message is preceded by a time stamp (hh.mm), which is followed by the object (i.e. device - ddnn or job - ron.step) and then the message itself. Two examples are shown below:

```
hh.mm ddnn HELD      (meaning that the specified device is Held)
hh.mm ron  STARTED job-id user-id class (meaning that the
                                         specified job has
                                         started execution)
```

The second type of message like the first is normally just informative. It differs from the first type in that it has a two character, two digit identifier instead of a time stamp (i.e. DV06), the rest of the format is fairly similar. The first two characters of the identifier relate to message set (i.e. DV = devices, IL = initial load etc...), and the second two digits are just the occurrence of that message in the set. Two examples are shown below:

```
DV24 ddnn VOLUME UNKNOWN (meaning GCOS cannot recognize the
                           volume mounted on the indicated
                           device)
CG05 UCG SUCCESSFUL      (meaning that a Use Catalog command
                           has been successful)
```

If operator action is required the message is preceded by an asterisk "\*":

```
* hh.mm ddnn MOUNT vol-name FOR ron (meaning the indicated job
                                       needs the indicated volume
                                       mounted, the device shown is
                                       free and can be used)
```

Messages may extend to more than one line, in this case the following lines are indented but the time stamp or identifier are not repeated:

```
O2/hh.mm X6.2 KILLED INOUT SEV5
          REPEAT ?
```

Note : the first three character positions on the log are reserved exclusively for the action indicator (\*) and deferred reply identifier (O2/ in the above example, see 4 below)

## 2. OPERATOR COMMAND

This is a communication from the operator to the system and is performed by:

1. Pressing the ATTENTION key on the keyboard, which causes the PENDING indicator to illuminate.
2. The message S: appears at the start of a new line.
3. PENDING extinguishes at the same time as ENTER illuminates.
4. Once ENTER illuminates, the operator keys in the message.
5. When the message is correctly composed, press the EXECUTE key.
6. The effect of this action is:
  - If both the CRT and the Console Serial Printer are functional as the console then the message appearing first on the CRT is now printed on the serial printer
  - The message text is sent to the system
  - ENTER extinguishes.

Example:

S: S0 PRO1

(the underlined part is entered by the operator)



## ABNORMAL CASES

- . The type-in is incorrect and the command has not yet been entered. The operator can press the ATTENTION key instead of the EXECUTE key. The type-in is ignored and the "S:" is re-issued.
- . The command does not exist. GCOS will answer:

OP02      CC      UNKNOWN COMMAND

- . The command syntax is illegal. GCOS will answer:

OP03      CC      ILLEGAL SYNTAX

OP11      CC      INVALID PARAMETER : xxxxxxxx

- . The system queues for dialog are full or the processor (for BTNS) is not active, the command is not taken into account

OP07      CC      RETYPE THE COMMAND LATER

OP08      CC      COMMAND NOT CURRENTLY AVAILABLE

- NOTE :
- . Some commands require several parameters; these parameters are separated by a space (or several spaces).
  - . A command can be more than one line long; in this case the first line should be terminated by a hyphen. GCOS will type S: at the beginning of the next line and the continuation of the command can be typed in.

### 3. QUESTION WITH IMMEDIATE REPLY

This type of question is used when the system cannot continue until the answer has been provided (for example during GCOS loading)

The question is asked with a message, in which the message is preceded by an "\*", and which ends with a "?". After the message is given, on the next line the symbol I: is printed. The operator must then type in the answer.

Example:

```
* SV05 GCOS : FILE SALVAGING ?
```

```
I: NO
```

(the underlined part is the operator answer)

If the answer is not given after a certain time the question is asked again.

After the system has printed I: no other message will be given until the operator has given an answer.

### 4. QUESTION WITH DEFERRED REPLY

When this type of question is asked, the answer can be given whenever the operator wishes.

A reply identifier is given together with the question. The answer will be given, with the same mechanism as a command, but the reply identifier is used instead of the command code and is followed by the answer.

Example:

```
07/0U02 PRO2 OUTPUT X18:4 RESTART FROM ?
```

```
S: 07 BEG
```

(the underlined part is the operator's answer)

If the answer is not correct the question will be asked again but the reply identifier will change.

## Abnormal cases

- . The reply identifier corresponds to no question or the job which asked the question has been terminated.

### OP01      INVALID REPLY IDENTIFIER

- . The system queues for dialog are full, the reply is not taken into account.

### OP06      nn    RETYPE THE REPLY LATER

## STORAGE LOAD

The purposes of a storage load are:

- . To load the operating system GCOS
- . To establish the status of the session (date and time, resident volumes)
- . To treat the information left over from a previous session (dump, preinitialized load modules, information on jobs and their associated outputs)

Storage Loading proceeds as follows:

1. The system status at the end of the previous session is displayed
2. Question ILO1 is asked which allows the selection of various options (Dump; Restore; Warm, Cold or Clean restart) and to specify or modify elements of the session identity (date and time, resident disks, MAM)
3. A Dump session is executed if required
4. A Restore session is executed if required
5. GCOS ascertains the state of all the devices
6. A Clean, Cold or Warm restart is executed
7. A message, indicating that GCOS is available, is displayed.

A typical dialog for a first GCOS loading after creation of the system disk, a first loading of the day and a loading after a system crash are shown below.

1. At first GCOS Loading after creation of the system disk a typical answer will be:

```
RESIDENT(C117),DATE(11.5.78/14.0)
```

which initiates the name of the disks to be used in the Dump and Restore sessions, the list of resident disks and the date and time.

2. First GCOS loading after System Initialize. Beginning of a day's work, the last session was terminated by a shutdown (END session command)  
Operator actions are underlined.

```
SYSTEM PREVIOUS STATUS ON 10.04.78 AT 19.02.35  
SHUTDOWN
```

```
* ILO1 OPTIONS ?  
I: DATE(10.5.78/8.0)  
08.00 DV21 MS01 PREMOUNTED SYSIC  
08.00 DV21 MS03 PREMOUNTED TPRC4  
08.00 DV21 MT04 PREMOUNTED BO1794  
08.00 GCOS LEVEL 64 - RELEASE 1D.18** READY ON APR 11  
*****
```

3. GCOS loading after a system crash. The dump corresponding to it is stored on disk D12, job X47 was executing and will be re-executed from the beginning.  
Operator actions are underlined.

```
SYSTEM PREVIOUS STATUS ON 10.05.78 AT 13.45.22  
CALL TO PANIC FROM J = 00 P = 08 CODE = 2080  
PARAMETERS : 2080 4040 00080904 0B47 0132
```

```
* ILO1 OPTIONS ?  
I: DUMP  
* ILO3 DUMP ? 0,D12.MS  
* ILO3 DUMP ? FW  
* ILO3 DUMP ? (Press EXECUTE key)  
13.49.DV21 MS01 PREMOUNTED SYSIC  
13.49 DV21 MS03 PREMOUNTED TPRC4  
13.49 DV21 MS02 PREMOUNTED AB1245  
13.49 DV21 MS04 PREMOUNTED D12  
* RRO1 X47.1 ABORTED SEV6  
CRASH REPEAT WHOLE JOB ?
```

```
I: YES
```

```
13.50 GCOS LEVEL 64 - RELEASE 1D.18** READY ON MAY 11  
*****
```

The remainder of this section describes the loading operations in detail, in the order in which they happen.

### SYSTEM PREVIOUS STATUS

GCOS indicates whether or not a System Dump is available as a result of the previous session.

No dump is available:

IL28 NO PREVIOUS DUMP

A dump exists:

SYSTEM PREVIOUS STATUS ON mm.dd.yy AT hh.mm.ss

{ UNKNOWN

{ HARDWARE CHECK IOC = nn PC = nn LC = nn

{ CALL TO PANIC FROM J = j P = p CODE = nnnn

{ SYSTEM EXCEPTION }

{ CHANNEL EXCEPTION } ON J = j P = p CLASS = nn TYPE = nn

{ HARDWARE FAILURE }

Then in both cases GCOS asks question ILO1 to know what work is required.

NOTES - When a System Dump exists it must be processed by a Dump Session during this loading, either to be printed or stored on magnetic disk. This is necessary because when the dump was taken only the segments in memory are stored. Other segments are dumped from backing store and might be modified during the new GCOS session.

- If the System Previous Status was SHUTDOWN there is no dump available. The Dump area contains no valid information and any attempt to perform a dump will produce unpredictable results.

## QUESTION ILO1

After indicating the system previous status, GCOS asks the question

\* ILO1 OPTIONS ?  
I:

	DATE
	HELP
	DUMP
	RESIDENT
	RESTART
	RESTORE
	MAM
	DUMP_CD

where the answer given to this question directs the remaining loading operations. The answer to question ILO1 consists of possibly several elements, depending upon the choice of the operator:

- . Whether or not a Dump session should be executed.

A Dump session is needed if the previous session was abnormally terminated and a dump is available. It will allow the dump to be printed either directly on to the line printer or to be sent to a disk file for printing later.

- . Whether or not a Restore Session should be executed. (Certain Restore options may be executed at the request of Field Engineering)
- . Whether or not to perform a Cold Restart or a Clean Restart. The default option - a Warm Restart - will normally be the case, but sometimes a Cold Restart will be desired. A Clean Restart will be performed only in very rare circumstances.
- . Whether or not to include MAM (specifies the handling of telecommunication message queues).

The answer to Question ILO1 is also used to establish the status of the session by indicating or modifying

- . The date and time
- . The list of resident disks

After a normal shutdown with no dumps the operator just presses the EXECUTE key. A Warm Restart is performed. The date and time will be current and the list of resident disks will be unchanged.

## Remarks

It is possible to use the card reader as an input device for answering question IL01. To do so, CARD is typed in reply to IL01 and the cards present in the reader will be read. The answer format on cards is identical to that used for the console, including the use of a hyphen (-) as a continuation character.

If the last entry on the cards is CONS then control is passed back to the console, otherwise the answer will be considered complete.

## To request Help in replying to IL01

If the operator wants a reminder of the answers to IL01 he can type HELP as the first reply. This results, in all the possible answers being printed on the printer, together with those for question IL03. Question IL01 is re-issued at the end of the list.

## DATE AND TIME

DATE (mm.dd.yy/hh.mm)

The date and time are memorized from one GCOS session to the next, as long as:

- . The CPU is not powered off and/or
- . The INITIALIZE pushbutton is not used or
- . A new date is not given.

When entering the date and time all values can be given with 1 or 2 digits as appropriate.

If the date element is omitted at the first GCOS loading after using the INITIALIZE pushbutton (or after giving an impossible date):

\* IL44 DAY / TIME M.D.Y. / H.M?

I: 06.22.78/10.15

is asked and awaits an immediate reply, an example of which is also shown. The date and time can be ascertained at any time by use of the Display Time command (Section VI).

## TO INDICATE RESIDENT DISKS

RESIDENT(vol-name1,vol-name2...)

This list is used during the initial checking of device states.

It is memorized from session to session (even if in the meantime the system has been powered off).

This parameter should therefore be given at the first GCOS loading after creation of the system disk, or later on if the list is to be changed. In all other cases it can be omitted.

The system disk need not be indicated in the list. If it is desired that only the system disk be considered as resident the format is then: RESIDENT (/).

The resident disks are assumed to be MS/M400's if other disks are to be used the device class must be specified in the RESIDENT element: for example:

RESIDENT(vol-name1,DVC-MS/M350,....)

## TO SPECIFY A GENERAL HANDLING OF TELECOMMUNICATION MESSAGE QUEUES

MAM ( ( { YES  
REFORMAT } ) )  
NO

- YES : leaves all queues as they were. A roll back of queues assigned to each job will be requested; (see 'File Salvager' in this section) this is only possible after a warm restart. It should be noted that IOF, BTNS and ROF service jobs must be started again after a warm restart.
- REFORMAT : all queues are cleaned up. All messages are erased.
- NO : no telecommunication job can be executed until a new generation of telecommunication is performed.



## DUMP SESSION

This subsection assumes that all dialog is performed on the console. It is possible to answer questions ILO3 on cards (this question is dealt with later in this section) using the same format as via the console.

It is also possible to store on disk a series of answers. This is done by answering question ILO1 with:

```
DUMP_CD(answer to ILO3; [answer to ILO3]...)
```

Note : the last entry should be "N" or more answers will be requested from the console when DUMP(DISK) is used.

This series of answers will be stored and any future dumps needed will be obtained by answering question ILO1 with:

```
DUMP(DISK)
```

A Dump Session allows the operator to either print the dump left at the end of the previous GCOS session or to store it on disk (in the SYS.SYSDUMP file) to be printed later by the SYSDUMP utility. The latter is preferable as it makes more efficient use of CPU time. If a dump is not treated by a Dump Session during GCOS loading it may be corrupted by the GCOS session over-writing the backing-store, where most of the dump is situated.

The Dump session is entered if the answer to question ILO1 contains the element:

$$\text{DUMP}[(\text{vol-name1} : \text{volname2}) [ , \left. \begin{array}{l} \text{CONS} \\ \text{CARD} \\ \text{DISK} \end{array} \right\} 1 )]$$

vol-name1 is the disk containing the SYS.BKST1 file; normally the System disk, in which case vol-name 1 can be replaced by a "/".

vol-name2 is the disk containing the SYS. BKST2 file, if it is the same disk as vol-name1 then it can be replaced by an "\*".

<u>CONS</u>	-	Device from which the answer to questions in
<u>CARD</u>		the Dump Session will be given. The default
<u>DISK</u>		(console) is the usual device.

The normal form of this element is to enter:

```
DUMP(/:*)
```

the first time a Dump Session is needed and:

```
DUMP
```

and press the EXECUTE key if any further dumps are needed.

If a dump is available from the previous GCOS session then question

\* ILO3 DUMP?

is asked.

### SYSDUMP Utility

A utility known as SYSDUMP allows the operator to print a dump which has been stored on a disk whilst the system is running.

SYSDUMP is introduced by the operator command:

SJ SYSDUMP

The SYSDUMP job then prints the stored dump automatically.

Note : if the Field Engineers request a tape copy of a dump for further analysis, this is achieved by using the FILESAVE utility on the SYS.SYSDUMP file to copy it to a tape.

### QUESTION ILO3

This question is asked when a dump is available from the previous GCOS session. It allows the answer of one function at a time and is asked until you terminate the dump session by entering N and pressing the EXECUTE key (or by just pressing the EXECUTE key). Note DUMP(DISK) or DUMP(CARD) answers must also be terminated by an "N", or question ILO3 is asked on the console as well.

The most usual dialog is (with operator actions underlined):

\* ILO3 DUMP? FW

\* IL14 CRASH COMMENTS?

I: (press EXECUTE key)

ILO3 DUMP? (press EXECUTE key)

Where FW indicates that a full dump (fully writable) is to be sent to the line printer. IL14 is asked as an opportunity to the operator to enter any useful comments, which will appear on the first page of the dump printout. (If more than one line of comments are to be given, terminate the line with a hyphen).

## Dumping to a disk

To store a dump on disk (SYS.SYSDUMP file) the console dialog would be:

```
* ILO3 DUMP ? O,volname.MS[,LONG]
* ILO3 DUMP ? N (or just press execute key)
```

vol-name - is the volume name of the disk which holds the SYS.SYSDUMP file, if it is the System disk it can be replaced by "/" i.e. O,/.MS (Operator replies are underlined; note the O in the operator reply is alphabetic.)

The SYS.SYSDUMP file can be put on any disk. The dump will overwrite any previously stored dump on the SYS.SYSDUMP file as only one dump at a time may be written to this file. Thus if this file holds a dump from a previous GCOS session which has not been printed and a new dump occurs, the new dump must be printed. If the new dump is sent to disk the old dump will be overwritten unless there is another disk holding a SYS.SYSDUMP file which can be used.

The LONG option is used to print a summary edit of the dump (mainly for Field Engineers) while the dump is being stored on disk.

In both the above cases the dumps are output as fully writable (FW). It is possible to edit dumps with other options, but this is mainly for Field Engineer use and is not described in this manual.

## Question IL14

```
* IL14 CRASH COMMENTS ?
I:
```

This question is asked after the first reply to question ILO3 (provided the reply was not "N" or disk-dump). It allows the operator to type in comments which will appear on the first page of the dump. If the comments use more than one line the lines should be terminated with a hyphen (-), except the last line. If no comments are required the operator just presses the EXECUTE key.

## RESTORE SESSION

A Restore session is used to restore the system segments in the SYS.BKST file on the systems disk from the original copy stored (in the SYS.SYSTEM file) on the system disk, taking into account system patches. See the Installation Guide.

### Release States

From time to time improvements to GCOS are released to customers. These improvements are incorporated by adding patches to the system, the patches are provided on a cassette or tape and incorporated via a special job; these patches are stored in the SYS.SYSTEM file. These new patches, plus selected parts of the previous version of the release form the new version, however a copy of the entire previous release is stored in another part of the SYS.SYSTEM file. When the old version is moved to the new part of the file it overwrites the previous old version. Thus there are two versions of the release in the SYS.SYSTEM file, the new version, and the old version; the relevance of this is explained below in OLD.

The Restore session is entered if the answer to Question ILO1 contains the element:

```
RESTORE[(vol-name[,OLD])]
```

vol-name - volume name of the disk containing the SYS.SYSTEM file. The vol-name is kept from session to session and will normally be indicated once and for all at the first GCOS loading after Systems Generation. "/" can be used instead of vol-name if the disk is located on the drive whose physical address was posted on the rotary switches when the ISL push button was pressed (i.e. the system disk).

The option listed below (OLD) is the only one that the operator should need; there are other options available, but these are for the use of the Field Engineers.

The most usual form of doing a Restore therefore is to answer question ILO1 by entering

```
RESTORE(/)
```

the first time a Restore is needed and simply

```
RESTORE
```

and press the EXECUTE key for any further Restore sessions. This uses the new version of this release for the Restore session.

OLD - This parameter restores the SYS.BKST from the original copy kept in the SYS.SYSTEM file, taking into account only the previously stored patches of the previous version of the Release state. This option should be used if a new version of

the release state is unsatisfactory and allows the use of the previous version.

If the SYS.SYSTEM file cannot be found on the specified disk, the following messages are displayed. The operator then types in the correct vol-name after the "I:"

```
IL38  SYS.SYSTEM NOT FOUND ON vol-name  
* IL30  GIVE NAME OF vol-name ?  
I:
```

## DEVICE TESTING

After performing a Dump session and/or a Restore session, if this is required, the system then tests the state of all its devices.

- Disk and tapes in Ready are treated as if they had just been premounted (Section V)
- On each Ready printer (that uses a print belt) the belt is treated as if it had just been premounted.

A premounting message is displayed for each disk and tape in the Ready state:

```
DV21 ddnn PREMOUNTED vol-name
```

The list of Resident disks, as given in answer to question IL02 or as memorized from a previous session, is used at this time. If this list is not empty (the System disk must be in this list) and all disks are in the Ready state, no dialog takes place.

If the list is empty, question DV42 is asked.

If some disk in the list is not in the Ready state the message:

```
DV25 GCOS:RESIDENT vol-name NOT FOUND
```

is displayed then question DV42 is asked.

### Question DV42

This question is asked only if a problem is found relating to the list of resident disks as given in answer to question IL01, or as memorized from a previous session.

The answer to the question is used during this session, but does not modify the list of resident disks memorized for the next session; to do so the RESIDENT element must be used in reply to question IL01.

```
* DV42 GCOS : RESIDENT VOLUMES ?
```

Answer I: vol-name1,volname2...

The volume name of all disks which are to be considered as Resident are separated by commas. The system disk need not be given in the answer. If only the system disk is to be resident, an empty answer can be given (press the EXECUTE key).

## RESTART SESSION

A Restart session treats all information concerning jobs and outputs left over from the previous session.

- . The Warm Restart, which will be performed unless the operator specifically requests a Cold or Clean Restart as part of his answer to question ILO1, attempts to put the system back into its previous status.
- . The Cold Restart can be used to erase information left over from the previous session and leads to an "empty" system.
- . The Clean Restart has a much more drastic effect and should be used only in exceptional conditions.

RESTART( { COLD }  
          { WARM } )  
          { CLEAN }

### WARM RESTART

The purpose of a Warm Restart is to resume a GCOS session as near as is possible to the state it was in when it terminated.

All characteristics set by operator commands are left unaltered; i.e., the multiprogramming level, job class characteristics, etc..

After a system shutdown (End Session command):

1. All jobs and outputs are restored in the same state that they were in at the time of the system shutdown, and everything appears as if there were no interruption of the GCOS session (except that certain service jobs need to be started again, see 4 below).
2. A message is displayed for each executing or suspended job. The Display Output or Display Scheduling commands can be used to check jobs and outputs in other states.

```
RS06  ron  REACTIVATED
RR05  ron.step  REMAINS SUSPENDED
```

After an Abnormal Termination (System Crash, System RESET/RESTART pushbutton):

1. The File Salvager is called to check the status of all files in use and to put them back into the correct state. (see use of File Salvager below).

2. All jobs not executing are unmodified. Jobs executing or jobs suspended between steps have the appropriate RS05 or RR05 message displayed. For jobs suspended inside a step question RR01 is asked so the operator can decide what action should be taken. This question is not asked if the user specified the job as nonrepeatable in JCL. See question RR01 in this section for full details.
3. A report is prepared (in output class A), listing all the actions taken concerning files and jobs.

Note : Outputs will remain in the states they were in when the previous session terminated. Question OU33 is asked for each output that was being printed or punched at the time of termination (see Output Writer in Section IV).

4. The Output Writer, BTNS, IOF, ROF service jobs are terminated and should be started again.
5. Any job being read by the Stream Reader will be lost and must be re-introduced.

### File Salvager

The File Salvager is called to check the state of all user files which were being used. A user file is said to be stable if it was either not opened or opened in input mode only. If all the user files are stable, no message is displayed and the salvager operation is terminated. If some user files are unstable, the message:

SV01 GCOS: UNSTABLE efn FILE (ON volname FOR ron

is displayed for each unstable file, followed by the question:

\* SV05 GCOS: FILE SALVAGING?

If the answer is NO, salvaging is not attempted and no further messages are displayed.

If the answer is YES, file salvaging is attempted and a SV09 message is displayed for each file indicating its state.

SV09 efn FILE	}	VERIFIED RECOVERED UNSTABLE UNALLOCATED NOT SUPPORTED
---------------	---	---



## Where

- VERIFIED = the file is stable
- RECOVERED = the file has been corrected and is stable
- UNSTABLE = the file cannot be corrected
- UNALLOCATED = a space allocation was being performed at the time of the abnormal system termination, and has not been completed. The file must be re-allocated.
- NOT SUPPORTED = the file is in a specific format not supported by the salvager.

Any other response: no salvaging is attempted and there is no message.

If a disk containing one of the files is not ready then the question

```
* SV07 GCOS : MOUNT vol-name ?  
I:
```

is asked. If the file is to be salvaged the operator should mount the disk and reply YES. If the reply is NO the salvage is bypassed for this file.

## Question RRO1

This question is asked during a Warm Restart following an abnormal system termination for every job that was executing.

```
* RRO1 ron.step n ABORTED lm-name SEVn[=mmm]  
      { REPEAT[FROM CHKPT nn.xx]? (i)  
        REPEAT WHOLE JOB ? (ii)  
        ROLLBACK ? (iii)  
I:
```

The first form is used if the executing step has been declared as being repeatable by the user. (Checkpoint corresponds to the beginning of the step).

The second form is used if the step is not repeatable

The third form is used if the step is not repeatable but the file and communication queues were journalized.

The severity code will normally be 6 (i.e., SEV6) indicating a System Crash. If n= 7 question RRO1 will have a reply code. This indicates that the step was about to abort (issue question SP07) when the system crashed.

## Answers

YES if the appropriate file and communication queues are to be rolled back. The execution of the step is then restarted from the checkpoint nn.xx or else the job is restarted from the beginning. Output created, during the part of execution which is repeated, is erased (except the Job Report).

NO to abort the current step and the job

ROLLBACK is the same as NO except that file and communication queues are rolled back. If no journal was being used, this answer is incorrect (this reply is applicable only for REPEAT? or REPEAT WHOLE JOB?).

ALL Indicates that an answer of YES is to be assumed for this and all following RRO1 questions. The following RRO1 questions will not be asked, but the Salvage session will progress as if they had.

YES,NO  
ROLLBACK is applicable to the second form of the question RRO1 only. The effect is the same as the YES response except that the file and communication queues are not rolled back before repeating the job execution. If no journal was being used, this answer is incorrect and question RRO1 is asked again.

HOLD is used only with the second form; the job is held and should be released before it can be executed.

HOLD, NO  
ROLLBACK is similar to HOLD but file and communication queues are not rolled back.

Note : if a job was not actually executing a step (i.e. job or step initiation or termination) its execution will proceed without any questions being asked.

## Question OU02

This question is asked during a Warm Restart for each standard output which was being treated by the Output Writer.

\* OU02 ddnn OUTPUT out-name FOR ron RESTART FROM?

The operator responses to this question are described under Output Writer in Section IV.

## COLD RESTART

A Cold Restart is given by answering question ILO1 with:

RESTART(COLD)

In the case of a Cold Restart:

- . All the jobs and outputs existing on the system are lost.
- . The system multiprogramming level, job class characteristics, accounting information and Message Of Today are left unaltered.
- . Preinitialized load modules are left unaltered.
- . No report is prepared.

This restart will therefore be used to reach an "empty" GCOS, for example when all the jobs and outputs of the previous session are of no use, or if a Warm Restart has failed.

Note: If temporary files were being used they are not deleted. To delete these files run the VOLCHECK utility.

## CLEAN RESTART

A Clean Restart is given by answering question ILO1 with

RESTART(CLEAN)

but will never be performed except in very rare circumstances (after Question VM02). This is for the following reasons:

1. All the jobs and outputs existing on the system are lost.
2. All the Preinitialized Load Modules are cancelled.
3. All the accounting information which is not already stored on file is erased.
4. All the job class characteristics have to be reinitialized.
5. All the shareable modules (used for data communications and transaction processing) are unloaded.
6. The network definition is destroyed.
7. No report is prepared.

Note : certain temporary files are not deleted. These files can be deleted by using the VOLCHECK utility.

## SYSTEM READY

After the Restart session is completed the system becomes Ready, and the message:

```
hh.mm GCOS LEVEL 64 - RELEASE ID.0 13** READY ON NOV 30, 1978
```

```
*****
```

is displayed. This signifies that Storage Load is now completed and that GCOS is fully operational.

- . Some jobs may be in the In Scheduling state if a Warm Restart was used. They will be started as shown under Scheduling in Section IV.
- . Some jobs may be Executing. This will happen if the previous session was terminated by an End session command; jobs which were executing will have been interrupted between steps, and will now be resumed.
- . Some outputs may be ready and the Output Writer must be started by using the Start Output Writer command (SO).
- . New jobs can be introduced by using the Input Reader or the Start Job Command.

### Error Messages for GCOS Loading

- Error in the answer syntax

IL06 ILLEGAL SYNTAX

or

IL29 OPTION ERROR AT element

Question IL01 is asked again, the operator types in the correct answer.

- Date element is required

\* IL44 DAY / TIME M.D.Y/H.M ?

I:

If the DATE element is omitted after an INITIALIZE or the date is impossible, question IL44 is asked. After a correct reply has been typed Loading will continue.

- An I/O operation has failed on disk

IL11 MSnn ERROR EV=cc PSB=ccc  
DSB=ccc . . .

\* IL15 MSnn RETRY ANSWER R OR C ?

I:

Replying "R" will cause a retry of the operation, "C" will cancel the operation. If the reply is "C" GCOS may issue message IL20, IL21 or IL22, a new ISL is then necessary.

Warning : An I/O error may be caused by an inappropriate Write Protect on the disk drive specified (indicated by "EV = 8E" in message IL11), in this case put the drive into Write Permit and reply "R". Only the disk holding the SYS.SYSTEM file can be in Protect, providing there is no Restore session or the Restore session has the "OLD" parameter. The System disk must always be in Permit, due to system backing store requirements.

- The System Resource and Status Table (SRST) has been changed without RESTORE

IL23 NEW SRST INIT AND RESTORE NECESSARY

Perform new INITIALIZE and Load.

- SRST structure error

IL24 SRST ERROR IN MAINTAINABILITY AREA

Perform new INITIALIZE and Load.

- Discrepancy between SRST and GCOS tables

IL25 INITIALIZE ERROR FOR ddnn

An Initialize error means that the specified drive cannot support a System disk in regard of GCOS tables; to recover the System disk it should be mounted on another drive that can support it, CONFIG should then be run.

IL43 WARNING PATH ccccc IGNORED DURING ISL

For IL43 Loading will continue, but it is advisable to run CONFIG.

- Resident disk is not ready

\* IL30 GIVE NAME OF vol-name ?  
I:

Operator mounts the disk and when the disk is ready replies, or if the vol-name specified was incorrect, replies with the correct disk name.

- Error in disk Volume Table of Contents (VTOC)

IL32 UNABLE TO READ VTOC FROM vol-name

The disk cannot be used. Mount appropriate stand-by disk and perform new INITIALIZE and Load. It is possible that the use of VOLCHECK on the disk will clear the VTOC error.

- System file absent or unreadable

IL35 UNABLE TO READ FILE LABEL OF SYS BKST  
IL38 SYS.ccc NOT FOUND ON vol-name

The operator should check that the correct media was specified, and perform a new load.

- Memory Reconfiguration

IL47 WARNING MEMORY RECONFIGURED

This is a warning message only, the system will run normally, but with a slower reponse time due to the reconfiguration.

- Two disks with the same volume name (vol-name) are mounted

IL50 vol-name ALREADY MOUNTED, DISMOUNT

The operator should dismount one of the disks.

- SRST segments missing from SYS.BKST file

IL51 NEW SRST : INITIALIZE NECESSARY

Perform new INITIALIZE and Load.

## GCOS SESSION TERMINATION

### Normal termination

A GCOS session is terminated by using the End Session (END) command. This is for a normal systems shut down where all the systems activities are stopped at the end of the session. A normal termination means that:

- . No step is executing; an END command will cause all executing jobs to be suspended at the end of their current STEP or at the next STEP check point.
- . No job is in a suspended state (except jobs suspended between STEPs).
- . The Output Writer is stopped; and END command will terminate all WRITER service jobs at the end of their current OUTPUTs.
- . The systems files are closed.

Note : some outputs may remain (i.e. some jobs may be in the output state).

After the End Session command, the installation can either be powered down, or GCOS reloaded by using the SYSTEM RESET, SYSTEM LOAD pushbuttons. The system can be powered down after an End Session command without pressing the SYSTEM RESET pushbutton, but this is not advisable as the machine may not be left in a determinate state.

### Abnormal Termination

If an error is detected which prevents any further processing because the error cannot be by-passed then the system terminates in a system crash. The message:

```
*   AVOI GCOS: CRASH {DUMP AVAILABLE}
                        {NO DUMP}
```

is displayed.

If the system is stopped by the use of the SYSTEM RESET pushbutton, the effect on the system is the same as if a systems crash had occurred except that no systems dumps is taken.

The system user files, jobs and outputs are in different states at the time of a system crash. The use of the Warm Restart during the next system loading will enable the recovery of these elements. After a system crash, the installation can be powered off or processing recommenced by using the SYSTEM RESET and SYSTEM LOAD pushbuttons.

NOTE : NEVER (under normal circumstances) terminate the system by using the INITIALIZE or POWER OFF pushbuttons. NO DUMP indicates that GCOS was unable to store the dumps on disk; it may be an indication of a problem related to either the System disk, the drive the disk is mounted on, or the Mass Storage Processor.

#### INFORMATION LEFT OVER AT SYSTEM TERMINATION

At system termination information is still left in the system files:

- . Jobs known to the system
- . Input enclosures associated with these jobs
- . Output prepared by (or for) these jobs

Some information remains in the SYS.BKST2 system file; the system dump (if it exists) is stored in the SYS.BKST and SYS.BKST1 system files. In the event of a system shut down, the systems files are closed. In any other case the system files (as well as user files) are left opened so that they can be recovered with the File Salvager whilst doing a Warm Restart during the next Storage Load.

#### SYSTEM RESET

The SYSTEM RESET pushbutton terminates the system activity by preventing any further instructions from being executed.

The TRAFFIC indicator on the System Operator Panel is off and the LOAD indicator is on.

After this state:

- . The installation can be powered down or
- . GCOS can be reloaded by use of SYSTEM LOAD or
- . A dump can be taken by using the SYSTEM RESTART pushbutton. After this, the installation can be powered down or the SYSTEM RESET pushbutton can be used again.

If the SYSTEM RESET pushbutton is used while the system is running, the system user files are left as if a system crash had occurred and no automatic system dump is taken. This condition must be followed by use of the SYSTEM RESTART pushbutton, this will cause a system dump to be taken and up to date information will be stored for a Warm Restart.



## POWERING DOWN THE SYSTEM

In order to power down the System the following actions have to be performed:

1. Terminate GCOS by use of the End Session command. (see End Session command, Section VI and 'Normal Termination' in this Section)
2. Press SYSTEM RESET on the System Operator Panel.
3. Power down each peripheral in turn as follows:
  - a) For Unit Record Devices: Press the POWER OFF pushbutton
  - b) For Mass Storage Devices: Press the POWER OFF pushbutton or DC OFF (depending on the type of device)
  - c) For Magnetic Tape Devices: Switch off the Cabinet Circuit Breaker situated at the rear of the cabinet.
4. Return to the System Operator Panel and press the POWER OFF pushbutton
5. Switch off the mains supply of power to the computer installation.

Note : The "END" command will not become effective if there are any suspended jobs. If there are suspended jobs GCOS will issue the following message:

```
hh.mm JOBS SUSPENDED,END NOT ALLOWED
```

When all jobs are released and have finished executing their current step (or have reached a step checkpoint), GCOS issues the following message:

```
hh.mm GCOS : NO MORE JOBS RUNNING
```

If Output Writer is in use it is terminated at the end of its current output. When shutdown is complete GCOS issues the following message:

```
* AV02 GCOS : END OF SESSION.
```

and the session is terminated.

## SYSTEM FUNCTIONS

Certain system functions are not resident in main memory and have to be loaded into main memory from backing store each time they are used. If they are not used very often, the time taken for the load is of no importance. However, in certain sites some of these functions are used all the time or very often, and the time taken to constantly load and unload these function segments is then quite noticeable. To overcome this time waste it is possible to preload certain desired functions into main memory and thus make them resident for the duration of the session, thus obviating the need to constantly move function segments to and from backing store.

The Preload Main Memory (PMM) command is used to load a function segment into main memory.

The preloaded functions can be cancelled, and thus taken out of main memory with the Cancel Main Memory (CMM) command.

The Information on the size, usage and availability of main memory can be ascertained by use of the Display Main Memory (DMM) command.

It is the Operations Manager's responsibility to decide which functions should be preloaded.

Systems functions should only be preloaded when they will be used, and not too far in advance. As soon as a system function is no longer needed it should be cancelled. These two points should be observed, otherwise main memory will not be available for other uses.

Note : it may be necessary to limit the amount of memory a step requires when the PMM command has been used. This is achieved via the \$SIZE JCL statement.

The following is a list of Systems Function that can be used with the PMM and CMM commands.

- BFAS, BFASI, BFASII
- HFASS, HFASSD, HFASI, HFASRDM
- UFASS, UFASREL, UFASI, UFASIK, UFASIP, UFASIKP, UFASTDS
- QUEUED - IDSR, IDSU - MAMM, MAMD - VCAM

Note : MAMM, MAMD and VCAM must not be preloaded in main memory when running Communications Network Configurator (CNC). If these functions are preloaded and CNC has to be run, then they should be cancelled with the CMM command.

For the Main Memory commands refer to Section VI; for more information on System Function Usage refer to the Systems Management Guide and Storage Estimates Manual.

SECTION IV  
RUNNING USER JOBS UNDER GCOS

This section comprises the following subsections:

- . "User Jobs - General Concepts" introduces the main concepts and terminology of GCOS.
- . "Multiprogramming" describes the various commands which enable the operator to control the system resources.
- . "Job Introduction" describes the ways in which jobs can be introduced to the system particularly through the Input Reader.
- . "Job Scheduling and Execution" defines the respective roles of the operator and the system in controlling the passage of jobs through the system.
- . "Job Termination" describes the ways in which a job may terminate
- . "Output Writer" describes the use of the Output Writer

## USER JOBS - GENERAL CONCEPTS

All computer systems have their own terminology and Series 60 is no exception. Operators will find that the terms become gradually more familiar, but the following broad picture may be helpful:

1. The programmers who send their work to be done on the computer are Users.
2. A user submits his work in discrete units, each of which is called a job. Accounting related to these jobs is performed by the system.
3. A job normally takes the form of a pack of cards containing a series of Job Control Language (JCL) statements. JCL is the language with which the user tells the system the details of how he wants his job to be run and what work is to be performed.
4. The user gives his job a name: the job identification.
5. Each job consists of one or several steps: the system allocates a step number to each step.
6. A step is an executable load module.
7. A load module is simply a compiled program, but a program in a form such that it can be loaded into memory and executed.
8. The whole life of a job in the system is a run.
9. The system generates a run occurrence number (ron) for each job in the system. This ron is the means by which the operator will normally identify a job once it has been introduced (The ron consists of one or more digits preceded by an "X". GCOS itself is X1, X0 is not a permissible ron).
10. Some of GCOS 64 functions are carried out by specific jobs called service jobs (e.g. JCL Translator, WRITER). These jobs run in a similar manner to user jobs, but in general they do not display any messages on the system console (see Service Jobs in this section).

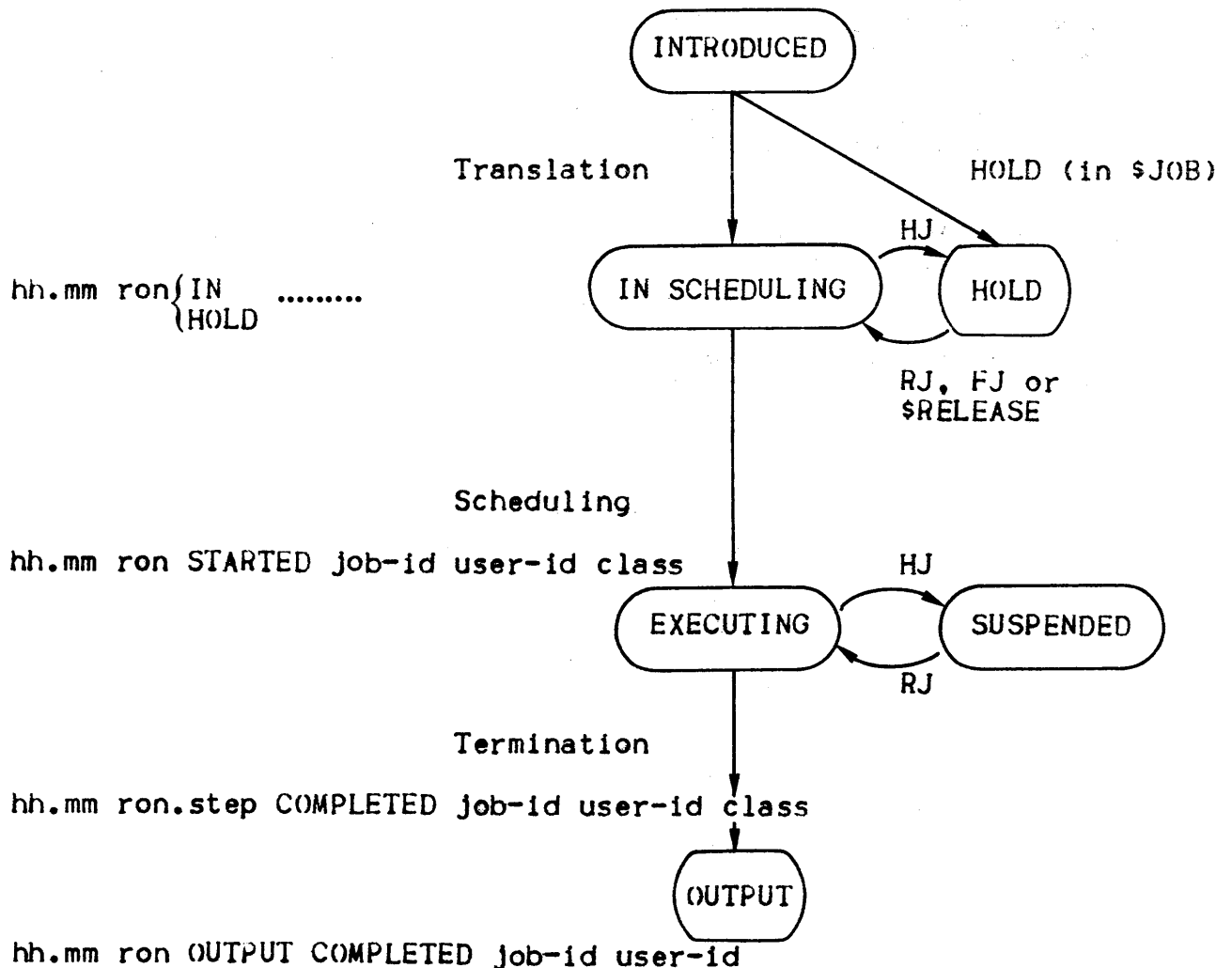
The Job State Concept

A job passes through various stages between the time of its introduction and the time of its deletion from the system. Dependent upon which stage it has reached, GCOS attributes to it a Job State.

The significance of these Job States is outlined below in Figure 4-1 and Table 4-1.

Corresponding Message

Job State



- HJ - Hold Job Command
- RJ - Release Job Command
- FJ - Force Job Command

Figure 4-1. The Job State Concept

Table 4-1. Job States

Stages in the life of a Job	Job State
<p>1. A job introduced by the Input Reader must first be translated. Until translation is completed the job is said to be Introduced.</p>	<p>INTRODUCED</p>
<p>2. Once the job is submitted to the system it can either be :</p> <p>a. Held until released by the operator or another job.</p> <p>b. Eligible to be selected for execution.</p>	<p>HOLD</p> <p>IN SCHEDULING</p>
<p>3. GCOS selects, from the eligible jobs, the one which is to be executed. The selected job is then started.</p> <p>While in the Executing state, a job can be temporarily suspended; the jobs resources are released and no further instruction or execution takes place until it is released for execution by the operator.</p>	<p>EXECUTING</p> <p>SUSPENDED</p>
<p>4. Eventually, job execution is terminated but the job remains in the system waiting for the printing or punching of its outputs to be completed.</p>	<p>OUTPUT</p>
<p>5. Once the output is complete the job is deleted from the system.</p>	
<p>6. Two types of service job (JCL Translator and WRITER) go into a specific state called IDLE whenever they have no more jobs to treat. They automatically start EXECUTING whenever new jobs need treating.</p>	<p>IDLE</p>

## MULTIPROGRAMMING

GCOS allows more than one job to be executing at any one time.

The number of jobs in execution at any one time is called the System Load. The maximum System Load is called the Multiprogramming Level

This can be modified by the operator by using the Modify Scheduling command (see Section VI)

It is part of the operator's job to monitor and control the use of system resources by the different jobs in the system. To enable him to do this the operator can display information on all jobs by using the DISPLAY SCHEDULING command (see Section VI).

Job Class, Scheduling and Dispatching Priorities are also means by which the operator can control the different jobs. Start Class, Modify Class, Terminate Class, Hold Job, Modify Job and Release Job commands are available for this function.

Note : The System (X1) and the Output Writer and Stream Reader are not taken into account in the System Load, but the JCL Translator (X2) is (when it exists).

The maximum multiprogramming level is set to 24, and this level is kept from one GCOS session to the next.  
If a job is suspended, it is not taken into account for the System Load.

### The Job Class Concept

(for all commands see section VI)

Jobs are grouped into classes which are identified by a letter : A,B,C,D,E,F,G,H,I,J,K,L,M,N,O,P for user jobs; P being the default value and Q,S,U,W,X for service jobs. (Users may not use these classes).

Job classes are mainly used with scheduling priorities to ensure a particular suite of programs are run in a particular order without upsetting the multiprogramming level.

Each class may be made available or not for scheduling by use of the Start Class and Terminate Class commands. When a class is held by the Terminate Class command no job of that class can start executing, except if the Force Job command is used.

Each class has a Class Multiprogramming Level which specifies the maximum number of jobs in that class which can be executing at any one time. The number of jobs in a class executing at the same time is called the Class Load.

Each class has a default value for Scheduling Priority and Dispatching Priority. These default priorities are defined at System Configuration.

The class status (i.e. started or not), current load, multiprogramming level and priority default values can be ascertained using the Display Scheduling command. They can be modified using the Modify Class command. These parameters are left unchanged from one session to the next. It should be noted that Modify Class commands will not be effective on jobs already executing. To alter the Dispatching Priority of an executing job the Modify Job command has to be used.

## PRIORITIES

Three types of priority are used by GCOS; one controls the order in which jobs are queued while waiting to run (Scheduling Priority), another controls the availability of CPU time for each job (step) whilst executing (Dispatching Priority) and the third priority is associated with Outputs (Output Priority).

### SCHEDULING PRIORITY (SPR)

This priority is mainly used in conjunction with job classes to control the order in which jobs are queued and selected for scheduling (see Job Class Concept above). Jobs are queued in priority order whilst waiting for translation, scheduling and machine resources. Jobs with the same priority are treated on a first in, first out (FIFO) basis.

The scheduling priority range is from 0 - 7, with 0 being the highest. Thus all jobs with a priority of 0 will start before jobs with a priority of 1 etc... (this is assuming that the jobs are not in the Hold state and the job classes are started). The fact that a job is held or its class is not started does not affect the job's position in the queue (apart from jobs leaving the queue to be run or being inserted in the queue).

Unless a scheduling priority is explicitly given in a \$JOB JCL statement or a Start Job (SJ) command, the job will assume the default priority of its job class.

Scheduling priorities can be displayed by using the Display Scheduling (DS) command and changed with the Modify Class (MC) or Modify Job (MJ) commands.

### DISPATCHING PRIORITY (DPR)

This priority controls the sharing of CPU time, between executing steps, during multiprogramming. Dispatching priorities are used to streamline step-execution flow and thus optimize CPU utilization.

The DPR range is from 0 - 9 (0 being the highest). GCOS itself has a DPR higher than 0 and thus can always gain control of the CPU.

Unless the DPR is explicitly given in the \$STEP JCL statement it is assumed to be the same as the default priority of the job class.



DPRs can be displayed using the Display Dispatching Priorities (DDPR) command or the Display Scheduling (DS) command. The DPR can be altered using either the Modify Dispatching Priority (MDPR) command or the Modify Job (MJ) command, the MDPR has a wider use and is explained in DPR Use below.

Service jobs use the default priorities of their class and should not, under normal conditions, be modified.

#### OUTPUT PRIORITY (OPR)

Job-outputs are assigned classes and priorities in an analogous way to jobs being assigned a job-class and SPR, even to the assumption of defaults. These classes and priorities are used in controlling how and when an output can be treated by WRITER. For a full description see Output Writer later in this section.

#### DPR Use

The basic rule for DPRs is that the step with the highest priority will gain control of the CPU whenever it needs to use it. The step with the next highest priority will be able to use the CPU during the time the first step has left it vacant (e.g. when the first step is executing an I/O operation). This effect applies to other executing steps. If a new step starts and its DPR is higher than any of the others it will gain control of the CPU as soon as it requires it and assume the position of the of the "first step" discussed above.

The step that loses control will wait in front of steps of the same priority. This lowers the number of times that data and programs are read from the backing store to the main memory as the steps should find its executing-environment still in main memory when it regains control of the CPU.

The allocation of DPRs can have a major impact on system throughput and job execution time. Since an I/O operation takes much more time than CPU manipulation of the same amount of data, an I/O-bound job only uses the CPU for a small percentage of its executing time. The converse is true for a CPU-bound job.

Thus if two jobs are executing, one I/O-bound and the other CPU-bound, it is best for the I/O-bound job to have the higher priority. This way the CPU-bound job can use the CPU during the other job's I/O operations and so share CPU time. In the reverse situation the I/O-bound job would get very little CPU time until the other job had terminated. In conclusion, I/O-bound jobs should normally have the higher DPR.

GCOS provides three additional methods of allocating CPU time by modification of the DPRs:

1. Executing steps using the same DPR can be made to equally share available CPU time. The priority is then said to be "sliced".
2. A limit can be placed on the amount of CPU time available to steps using a particular DPR bracket. These priorities are said to be "limited", they are also sliced.
3. Two consecutive priorities can be declared as optimized. In this case the DPRs of the steps involved will be decreased or increased dynamically so that the I/O-bound steps will have an effectively higher priority and the CPU-bound steps a lower one. These priorities are said to be "optimized, they are also sliced.

In all three cases the DPR is said to be "controlled" as opposed to its normal state.

These functions are used via the Modify Dispatching Priority (MDPR) command. The current state of the DPRs can be displayed using the Display Dispatching Priority (DDPR) command.

### Service Jobs

Some of GCOS's functions are executed as service jobs; this subsection is only to show how they differ from user jobs. These jobs are described in full later in the manual.

The following are GCOS service jobs:

- JCL Translator (JTRA)
- Writer part of Output Writer
- Basic Terminal Network Support (BTNS)
- Interactive Operation Facility (IOF)
- Remote Operator Facility (ROF)

All service jobs are assigned a ron when introduced to the system.

The job class is exclusive for that particular service job. See table below.

There can only be one JTRA and BTNS job executing at any one time.

There is one WRITER job for each device on which Output Writer has been started with a maximum of eight executing at any one time. Note the maximum number of WRITER + ROF jobs is also eight.

There can be one IOF or ROF job for each connected terminal user.

## Service Job Characteristics

Function	Jobid	Userid	Class	Comments
JCL Translator	JTRA	JTRA	X	uses IDLE state
BTNS	BTNS	BTNS	S	creates ouput
Writer	WRITER	ddnn	W	uses IDLE state
IOF	IOF	log on	Q	
ROF	ROF	log on	U	

Service jobs do not display log messages concerning run states except in error conditions (i.e. error messages).

Service jobs go through the same basic states as user jobs, but with certain differences.

Both JTRA and WRITER use a specific state called IDLE. They enter the idle state when they run out of work. However, as soon as work is available (i.e. jobs to be translated or outputs to be printed or punched), they automatically enter the In-Scheduling state. The IDLE state economizes the use of system resources and removes the need for operator intervention.

Service job scheduling is controlled by their class SPR, class multiprogramming level and the maximum number of J. It is not limited by the system multiprogramming level. Service jobs use the scheduling and dispatching priorities of their class; these priorities can be altered by use of the Modify Class command (except class X and S) and the Modify Job command (except JTRA and BTNS) for executing service jobs.

BTNS, IOF and ROF service jobs never wait for scheduling or system resources. If they cannot start executing at once, they abort and have to be reintroduced. The job displays a message, either on the log or the appropriate terminal that there was a start failure.

Only BTNS creates an output, and it is normally the only service job to go through the OUT state. However, if a service job aborts during execution it enters the OUT state and a dump of the job is taken. The job will remain in the OUT state until the job report and dump have been printed.

Accounting information for service jobs is stored on the accounting file in the same way as for user jobs.

## JOB INTRODUCTION

A job may be introduced to the system by one of the following means:

1. Input Reader
  2. Start Job command
  3. "Spawning" of a job during the execution of another job (by use of the \$RUN JCL statements)
  4. From a terminal, by use of ROF or FTU service jobs.
- } described below

When a job is introduced, the system displays a message whose form depends on the way in which the job was introduced:

- . If introduced by use of the Input Reader:

```
hh.mm ron { IN } job-id user-id class SPR = spr
           { HOLD }
```

Notes : The job is given a ron by the system at introduction.

If the JCL statements include a HOLD, the job is put into hold after translation. The job will not be started until a Release Job command is issued, or another job executes a \$RELEASE JCL statement.

A limit on the number of jobs that the system may know simultaneously is defined during configuration. When this limit is reached GCOS displays the following messages:

```
INI4 JOB READING STOPPED : JOB MAX. NUMBER
```

After an End Session command has been issued new jobs cannot be introduced to the system (see "System Termination").

Service jobs are introduced by specific operator (or terminal user) commands or actions. No message is displayed on their introduction.

## The INPUT READER

The Input Reader has two main functions:

- To read the input stream (one or more job descriptions) from cards, disk or magnetic tape.
- To translate job descriptions and put them into the In Scheduling state or into the Hold state (if requested by use of a \$JOB statement or by a Hold Job command)

These two functions are performed by the Stream Reader and the JCL Translator, both of which are described below.

## The Stream Reader

The Stream Reader is part of the system and not a separate job, thus it can be activated regardless of the system load. The Stream Reader can read the input stream from either cards, sequential disk or magnetic tape files or a member of a disk library. In all these cases the job descriptions are held in card image format.

The Stream Reader is activated by any of the following:

- Mounting a card deck in the card reader and pressing the START button.
- A Start Input Reader (SI) command.
- A Start Job (SJ) command.
- The execution of a \$RUN JCL statement in an executing job.
- From a terminal using either the IOF or ROF service jobs.

Stream Reader activation can be requested even if the Stream Reader is already in use. The requests are queued and then treated sequentially.

The Stream Reader is deactivated when:

- The input stream has been completely read; i.e. when a \$EOS card image has been read.
- A Terminate Input Reader (TI) command is issued.
- An End Session (END) command is issued.
- Error conditions arise (see Console Messages manual)

When the Input Stream is a sequential disk or tape file or a member of a source library, the operator initiates the Input Reader with the Start Input Reader or, from a library only, with the Start Job command. These commands are described in Section VI.

## Stream Reader Execution

Once the Input Reader has been initiated (either automatically or by use of the SI or SJ command) it proceeds in the following way:

1. If it finds that the specified device contains nothing for it to read, it issues a message asking the operator to mount the volume or card deck containing the Input Stream.
2. The Input Reader reads all the job descriptions in the Input Stream and for each displays the message:

```
hh.mm    ron IN  job-id user-id class SPR = spr
```

If, however, the \$JOB card or record indicated that the job was to be put into the HOLD state, the message is in the form:

```
hh.mm    ron HOLD job-id user-id class SPR = spr
```

3. When an End of Stream (\$EOS) card or record is read, or an End or Terminate Input Reader command is received, the Input Reader stops reading and warns the JCL Translator that there are jobs to be translated. If necessary the JCL Translator is activated from the IDLE to the In Scheduling state.

## THE JCL TRANSLATOR (JTRA)

The JCL Translator is one of the GCOS service jobs. It is always known by the system; it does not terminate but enters the Idle state. Only one JTRA can be known to the system at any one time.

The JTRA is the only job that can use class X and it always has a Scheduling Priority of 0. When it is in the In Scheduling state it will start before all other jobs except a Forced job.

The JTRA is activated whenever there is an Introduced job present. If more jobs are introduced while the JTRA is executing these will also be translated before JTRA enters the Idle state.

If, after a Warm Restart, there are untranslated jobs in the Introduced state, they will cause JTRA to enter the In Scheduling state and then be translated.

After translation JTRA puts the jobs into the scheduling queue, they either go into the Hold state or into the In Scheduling state, depending upon the job description.

When there are no more jobs for translation, JTRA enters the Idle state.

The only control command that can be used on JTRA is the Modify Job (MJ) command, even then it may only be used to alter the Dispatching Priority.

JTRA does not normally display messages or generate an output. Occasionally messages with a key of "TNnn" may be output, these should be referred to the Field Engineers. Users may cause a message to be displayed at translation by use of the \$MESSAGE JCL statement.

If during translation a JCL error is discovered or JCL "INVOKED" is not ready or is being used by another job the job in error will abort. Log messages tell the operator when a job has aborted, refer to the Console Messages manual for the corrective actions.

## HOLD STATE

After introduction and before execution a job can be put into the HOLD state. While in this state the job cannot be started unless a Force Job command is used on the job. Jobs are put into the HOLD state after translation.

A job can be placed in the Hold state on introduction by use of the HOLD parameter in the \$JOB JCL statement. While the job is in the In Scheduling state the operator can use the Hold Job (HJ) command to put a job or all the jobs into the Hold state.

HJ ron list (if no parameter is used, all In Scheduling jobs are held)

A job can be put back to the In Scheduling state by the operator using the Release Job (RJ) command or by a job executing a \$RELEASE JCL statement.

RJ {ron list}  
  {ALL} (if ALL is used, all held jobs are put into the In Scheduling state; not applicable to suspended jobs)

Note : the HJ and RJ commands can be used to suspend and release executing jobs. For full details see Hold Job and Release Job in section VI.

## JOB SCHEDULING

Whenever a job terminates or a new job is introduced GCOS will try to start another job. Jobs in the In Scheduling state are queued for selection in scheduling priority order and using a FIFO basis for jobs of the same priority. GCOS will try to start the first job in the scheduling queue provided the following conditions are satisfied.

1. The job must not be in the HOLD state and its associated job class must be started.
2. The maximum number of "J" has not been reached.
3. The system multiprogramming level has not been reached (for user jobs only).
4. The class multiprogramming level has not been reached.

If the first job in the queue cannot start then GCOS will try to start the next job, and so on. This is repeated until no more jobs can start due to the above limitations, or because no more jobs are waiting to be started.

Service jobs are not limited by the system multiprogramming level but they are by the other conditions.

A job can be pointed out to be selected immediately by use of the Force Job command. This job is said to be forced.

Forced jobs ignore all restrictions except for the maximum number of "J". Thus a forced job may overload either the system or class multiprogramming levels. In this case no new jobs, dependant upon these levels, will start until the levels return to their normal limits.

User jobs display the following message when they start.

```
hh.mms ron STARTED job-id user-id class
```

Service jobs do not display any message.

The order of jobs waiting to be started can be found using the Display Scheduling (DS) command and altered using the Modify Job (MJ) command.

In summary, job scheduling tests the following relations:

1. Total number of executing and suspended jobs < maximum number of "J"
2. System load < system multiprogramming level (for user jobs only)
3. Class load < class multiprogramming level

Service jobs ignore case 2.  
Forced jobs ignore cases 2 & 3.



## JOB EXECUTION

Once started the execution of a job proceeds sequentially through the JCL statements (unless the user, using \$JUMP statements, has requested some alternative).

The main part of the job execution is the execution of steps, but the user may:

- Introduce new jobs by using \$RUN JCL statements, which have the same function as the Start Job (SJ) command
- Release jobs in the HOLD state by using \$RELEASE JCL statement
- Display messages to the operator by using the \$SEND JCL statements. These messages are also shown in the corresponding Job Report (JOR or JOB-REP). These messages appear in the format:

SPO6 ron users-message

- Execute operator commands as if entered from the console. This is by use of the \$OCL JCL statements.

Eventually the job execution will terminate.

## EVENTS DURING JOB EXECUTION

If an event occurs during job execution it will normally be caused by a device or by the media. Other events during execution are reported to the user in the Job Report (prepared by the system).

Occasionally operator intervention is necessary and this is signified by console messages (not all console messages need action, some are purely informative).

The rest of this subsection deals with the component parts of a job run and certain possible events are also described where appropriate.

## STEP EXECUTION

Resources are needed for each step to be executed. The job may be enqueued to wait for a resource, in which case one of the following messages is displayed:

- . A file is being used by another job and cannot be shared:

RS05 ron.step WAITS FOR FILE e fn

- . A volume is being used by another job and cannot be shared:

RS05 ron.step WAITS FOR VOLUME vol-name

- . A device is needed and none is currently free:

RS05 ron.step WAITS FOR DEVICE dd

- . More memory space is needed than is currently available:

RS05 ron.step WAIT FOR nnK

- . The step is in Program Mode, and another Program Mode step is Executing or Suspended (this also applies to suspended jobs):

RS05 ron.step WAITS FOR PROCESSING

- . Not enough free space exists on the SYS.BKST file to load the step:

RS05 ron.step WAITS FOR LOADING

The list of jobs waiting for resources can be ascertained by using the DISPLAY SCHEDULING command.  
As soon as the resource becomes available, because another job stops using it, the message:

RS06 ron.step REACTIVATED

is displayed and the step execution proceeds. No operator intervention is necessary. The operator can however react to the conflict indicated by message RS05 in various ways:

1. Prevent other jobs from entering the resource conflict
2. Suspend or terminate one of the jobs in conflict if the conflict is found to be abnormal.  
This is achieved by using the Modify Scheduling, Modify Job, Modify Class or Hold Job commands described in Section VI

Volume mounting may also be requested and the execution will wait for these mounting operations to be completed before proceeding.

The execution of each step may cause operator visible events such as :

- Creation of an operator command:

OP04 ron.step COMMAND : cc CREATED

- Deletion of a previously created operator command:

OP05 ron.step COMMAND : cc DELETED

- Printing of a user message or question. These messages are stamped with the ron of the job followed by the step number:

ron.step message itself

- Transformation of a work tape into a normal tape (in particular in case of dynamic extension of a tape file):

\* FP01 ron TAG vol-name FOR EFN external file-name

The step execution can be temporarily suspended by using the Hold Job (HJ) command (see Suspended Jobs below).

If step tracing has been requested with the Modify Scheduling (MS) command the following message is displayed when each specified step starts to execute:

```
JB08 ron.step STEP lm-name DPR=dpr
```

If the step execution terminates normally the job execution proceeds. If it terminates abnormally the job will be aborted, unless the step has been declared as being repeatable (particularly with checkpoint restart) or the user has used \$JUMP JCL statements to take care of the situation.

#### STEP REPEAT

If a step terminates abnormally and it has been declared as being repeatable, or if use of file or communication queue journal has been requested, then the following question is asked (with a deferred reply):

```
xx/hh.mm ron.step STEP load module. ABORTED SEVn(= nnnn)
```

```
RC = ccccccc → siu, return code { REPEAT [FROM CHECKPOINT n] ? }
                               { ROLLBACK ? }
```

xx is the deferred reply code and must be used in the reply.

REPEAT - if no checkpoint is indicated the step is repeated from the beginning, else from the indicated checkpoint.

ROLLBACK - the step is not repeatable and only journal rollback is allowed.

SEVn - is the severity of the step abortion (i.e. SEV3); the following number is the status of the severity when the severity is not unary, see the table below.

ccccccc - is the Return Code register value and gives information on how the function was performed for evaluation by Field Engineering.

SEVERITY	STATUS RANGE	MEANING (when set by GCOS)
SEV3	10000 — 19999	USER ERROR
SEV4	20000 — 49999	SYSTEM ERROR (I/O error, system overload, etc..)
SEV5	50000 — 59999	TJ (operator used Terminate Job command without the STRONG element)
SEV6	60000	SYSTEM CRASH

If checkpoints were taken during the execution of the step, the last checkpoint is indicated. Checkpoint 0 is the beginning of the step.

#### Possible replies.

- YES - The execution of the step is repeated from the indicated checkpoint, after rolling back of files and communication queue journal if applicable. If the step was not declared as being repeatable (question ROLLBACK) only the journal rollback will take place.
- NO - The step execution is not restarted; no rollback of journal takes place. In most cases the job will be aborted unless the user has used \$JUMP statements, which will cause the job execution to be resumed.
- ROLLBACK - Same as NO except that the file and communication queue journal is rolled back. If a journal was not used, this answer is considered as incorrect and question SPO7 is asked again.

Any other reply is incorrect and the question will be repeated.

If a Hold Job (HJ) command is issued on the job before replying YES, the job will be suspended after rollback of the journal, if rollback is applicable.

If the step abort is due to an I/O error (a message would have been displayed) the operator should check the media and device (tape or disk) before replying. If needed the operator should mount the media on another device before replying.

#### PREINITIALIZED LOAD MODULES

Load modules (or "programs") can be stored in SYS.BKST2 file, they are then said to be preinitialized. This speeds up the loading of the load module when it is requested for execution in a job step. When a load module has been preinitialized the original copy (in a load module library) is not used. It is the operator's responsibility to control the SYS.BKST2 file:

- . To ascertain which load modules are preinitialized, use the DISPLAY LOAD MODULE command.
- . To preinitialize a load module, use the PREINITIALIZE LOAD MODULE command.
- . To delete a Load module from SYS.BKST2 use the CANCEL LOAD MODULE command. These 3 commands are described in Section VI.

## SYSOUT OVERFLOW

### OUI3 ron.step SYS.OUT OVERFLOW

This message indicates that an attempt to write to the SYS.OUT file by process n of the indicated step of the job was not satisfied because the SYS.OUT file was full.

If some output can be treated by the Output Writer this message is only a warning; the step will resume its execution when some space becomes available without any operator intervention.

Sometimes an operator intervention is compulsory :

- If the Output Writer is not started for the classes and devices of the outputs filling up the SYS.OUT file. The operator should then issue a START OUTPUT WRITER command on those outputs that are waiting to be treated. If there are outputs in the READY or HOLD state the Release Output (STRONG) commands should be used (see Output Writer in this section and the command in Section VI).
- If all the outputs in SYS.OUT belong to the same job and are to be treated only after job termination. There is a problem with that job which will never be able to run with this size of SYS.OUT file. The operator should either use the Release Output (STRONG) command on outputs in the READY state, terminate the job, or cancel some of the outputs to free some SYS.OUT space. (see Note)
- In similar cases if the all outputs in SYS.OUT belong to several jobs which will never be able to terminate if nothing is done (for example if they are waiting for some resource currently used by the job for which the overflow has been detected). The operator should then either use the Release Output (STRONG) command when appropriate, or terminate some of the jobs, or cancel some of the outputs.

In any event it is advisable:

- To ascertain the current contents of the SYS.OUT file by using the Display Output command.
- To issue a Start Output Writer command on classes and devices for which some outputs are waiting.
- Use the Release Output command with the STRONG parameter on outputs in the Ready state.

- To prevent other jobs from generating outputs by using the Hold Job, Modify Scheduling, Modify Class, Modify Job, Terminate Job commands as appropriate.
- Under extreme circumstances (i.e. printer failure) the Output Writer should be started on a tape file to free SYS.OUT space. The tape may then be printed later using the \$WRITER JLC statements.

Note : It is possible to prevent a step from using too much SYS.OUT space by use of the \$STEP JCL statement. Action should be taken to prevent SYS.OUT overflow from happening again: e.g. submit jobs in a different order, use private sysout files, correct a possible endless loop in a program, etc... .

## SUSPENDED JOBS

Executing jobs can be suspended using the Hold Job (HJ) command:

```
HJ ron list [ENDSTEP]
```

Suspending a job will prevent any further execution of the job. If the job is suspended at the beginning of a step, before devices have been allocated, device resources will be freed. Swappable memory is also freed when a job is suspended but other memory (such as I/O buffers) is not.

Suspending a job will not lower either the system or class load.

If the ENDSTEP element is used the job is suspended at the end of the current step. When a job is suspended between steps (e.g. after use of ENDSTEP, but not necessarily so) it will free almost all resources; the only exceptions are "passed" files and the system and class loads.

A suspended job can only be released by the Release Job (RJ) command. When a job is released it will start to execute straight away, in accordance with its Dispatching Priority.

Service jobs should not be suspended except under exceptional circumstances. JTRA and BTNS cannot be suspended at all.

Note : An End Session (END) command will not become effective until all suspended jobs are released.

## BEFORE JOURNAL

The Before Journal is a disk-based System file containing "checkpointing" information. This file is used to restore a user's file to its previous state (last checkpoint) before a restart is performed.

The use of the Before Journal is not normally visible to the operator except when the file becomes full; then the following message is displayed:

```
JL02  ron  MORE SPACE NEEDED FOR BEFORE JOURNAL.  WHERE ?
```

There are three possible answers:

vol-name - the operator replies with the name of a disk with room for the file extension and the job continues. The extra area used is freed when the job or step terminates.

NO - this is the normal reply if no more space is available. The step is aborted.

CONTINUE - if the operator replies continue the job continues, but it does so as if no checkpointing has been used. This reply should only be used with the user's express permission.

If the extension is not possible on the specified disk; or the journal is full and cannot be extended any more; or an I/O error occurs on the journal; or no Before Journal exists in the System; questions JL03 - JL06 are asked respectively. The replies to these are the same as for JL01 (see Console Messages guide).

## AFTER JOURNAL

The After Journal is a magnetic tape or disk file used with a Transaction Driven System (TDS). This file is used in restoring a TDS file's updated contents from a previous copy of the file after a failure; it is also used dynamically by TDS to handle cases where a transaction is unsuccessful. If the journal is on tape and needs extending, more work-tapes are requested. If the journal is on disk and needs extending, it requests more disk area in a similar way as the Before Journal (see Console Messages guide, questions JA01 - JA04).

## BACKING STORE (BKST)

```
VM01  GCOS : WARNING : BKST ALMOST FULL
```

This message warns the operator that the backing store SYS.BKST file is almost full. If this file does become full the System will crash, so immediate action is required.



The operator should:

- Prevent any more jobs starting. Use Modify Scheduling (MS) or Hold Job (HJ) commands.
- Suspend execution jobs with the Hold Job (HJ) command.
- Use the Cancel Load Module (CLM) command. This will free disk area that the SYS.BKST file will be able to use.
- Ascertain the amount of BKST available with the Display Backing Store (DBS) command.

If there is sufficient backing store available restart some of the suspended jobs. When a job terminates backing store will be freed and another job restarted.

If necessary terminate jobs using the Terminate Job command with the Strong element (TJ STRONG).

Possible reasons for backing store being full is that a job or jobs use an exceptional amount of backing store or there are too many jobs running. In the latter case it may be advisable to lower the system (and/or class) multiprogramming levels. For this use the Modify Scheduling (MS) command.

#### THRASHING

VM02 GCOS: THRASHING SITUATION  $\left\{ \begin{array}{l} A \\ B \\ C \end{array} \left[ \begin{array}{l} I/O \\ CPU \\ MS \end{array} \right] \right\}$

This message informs the operator that a job or jobs are constantly moving segments into and out of memory from the backing store. This indicates that the System is overloaded.

It is not recommended to react immediately, as any GCOS command will require memory and only increase the problem.

If the problem persists use the Hold Job (HJ) command, to suspend a job until more memory becomes available due to a job terminating. It may be necessary to free some memory by terminating a system activity (i.e. Output Writer).

If thrashing is still persistent it may be necessary to modify the multiprogramming level. The use of the \$SIZE JCL statement can be used (for some jobs) to prevent a step from being started if there is not enough memory available (refer to the Systems Management Guide).

## JOB TERMINATION

Whenever a user job terminates the following message is displayed:

```
hh.mm ron.step { COMPLETED } job-id user-id class [SEVn]
                  { ABORTED   }
                  { KILLED   }
                  [G4 = ccccccc siu, return-code]
```

COMPLETED - the job has been executed and terminated normally.

ABORTED - means that the job has been aborted because:

- (a) either a fatal error was found in the job description during translation
- (b) or an abnormal condition occurred during execution, or a Cancel Request command was used in reply to a system request for a certain device or volume

SEVn indicates the severity setting of the step.  
G4 = etc..., indicates the return code setting when the step aborted

KILLED - the job has been terminated by the operator's use of the Terminate Job command (described in Section VI)

Note : Service jobs do not display this message.

When a job is terminated by any means it does not disappear from the system until all its outputs have been processed by the Output Writer.

When a job is terminated, GCOS will look for another job to run, unless an END session command has been issued.

If no jobs can start (i.e. there are no jobs in the In Scheduling state) and there are no more jobs executing, the following message is displayed:

```
hh.mm GCOS : NO MORE JOBS RUNNING
```

Unless the operator wants an empty machine for a specific reason more jobs should be introduced or made available for scheduling.

## OUTPUT WRITER

During a job run various outputs are prepared for printing or punching. Instead of printing or punching these outputs directly it is possible to store them on an intermediate file. This allows more jobs to be run simultaneously than there are printers on the site (printers refers to card-punches as well in this section); this makes more efficient use of CPU time and the printers.

The Output Writer is part of GCOS and is in control of selecting outputs for printing and printing them via the WRITER service job.

When all of a job's outputs are printed the job is no longer known to GCOS, and the Output Writer displays the following message:

```
hh.mm  ron  OUTPUT COMPLETED  job-id  user-id  class
```

The Output Writer may also be started on a sequential tape or disk file instead of a printer. Each output is considered as being printed as soon as it has been transferred to the file. A file created in this manner can be printed by a job using the \$WRITER JCL statement.

Certain outputs created by jobs introduced via the ROF service job are sent to terminals and not printed. These "special" outputs are handled by an Output Writer function which is performed within each ROF service job. The maximum number of Writer jobs running is 8 at any one time, the maximum for ROF jobs is 6; however the maximum of ROF + WRITER must not exceed 8.

## STANDARD AND PERMANENT OUTPUTS

The file in which an output waits for printing can be either the system file SYS.OUT or a permanent file; temporary user files are not permitted. In the first case the output is said to be standard and in the second to be permanent.

Once a standard output is printed, the area it used in SYS.OUT is freed. Since all jobs using standard outputs share the SYS.OUT file an overflow condition may arise (see SYS.OUT OVERFLOW above).

Permanent outputs may be on non-resident media, tape or disk, and a mounting request is issued if necessary. Permanent outputs are not erased when they are printed and can be printed again by use of the \$WRITER JCL statement.

ROF service jobs handle standard outputs without operator intervention. Permanent outputs can be transferred back to a station (terminal) in the same way as any other permanent file.

## OUTPUT NAMES AND SEQUENCE NUMBERS

Outputs created by a job may have a specific name given by the user and they are also given a sequence number (within the run) by GCOS. Therefore the naming of an output can take one of two forms:

- 1)            ron:output-name                    e.g. -     X 17:TRUC
- 2)            ron:sequence-number                e.g. -     X17:3

For each job run GCOS prepares an output called the Job Occurrence Report (JOB-REP or JOR) in which all system information concerning the execution of the job is given. The JOB-REP always has a sequence number of 1. JOB-REPs are not produced for service jobs except BTNS; if however a service job aborts then a JOB-REP is prepared with the dump.

Jobs may have another system generated output called JOB-OUT. This output is created for each step that writes an output with the same characteristics as the JOB-REP. The JOB-OUT always has the sequence number of 2.

The output corresponding to the dump of a step (if the step aborted) will have a name in the following format:

DUMP\_sequence number

This helps in identifying the dump.

## OUTPUT STATES

**KNOWN** - The life of an output begins when its creation by a job begins; from this time until its treatment is completed by Output Writer, the output is known by GCOS.

**READY** - When the job closes the output file it is then said to be Ready; however, it is not yet eligible for selection by the Output Writer.

**WAIT** - As soon as the "when condition" indicated in the JCL statements is met the output enters the Wait state, it is then eligible for selection to be treated by Output Writer. The "when condition" can be end of job (which is the default), end of step, closing of the file etc...

Note : If the HOLDOUT parameter is used on the \$JOB card, then all the outputs of that job will remain in the Ready state even when the job terminates. These outputs will remain in the Ready state until released by the Release Output (RO) command.

**OUT** - While the output is actually being treated by the WRITER service job (i.e. printing or punching) it is in the Out state. At the end of this operation the output is no longer known to GCOS.

HOLD - An output can be put into the HOLD state by the Hold Output (HO) command or by use of the HOLD parameter in the JCL statements \$SYSOUT, \$WRITER or \$OUTVAL. An output in the HOLD state will not be selected until a Release Output (RO) command is issued. An output that is being treated can be put into the HOLD state by use of the HO STRONG command, question OU02 is then asked (see HOLD OUTPUT command later in this section).

Output Commands	Meanings
Cancel Output (CO)	- cancel the notification to Output Writer to treat this output.
Display Output (DO)	- ascertains the state of an output or outputs.
Force Output (FO)	- position a specified output as first to be treated.
Hold Output (HO)	- puts specified outputs or output into HOLD.
Modify Output (MO)	- modifies an output's class and/or priority.
Release Output (RO)	- makes an output available for treatment.
Start Output Writer (SO)	- starts the Output Writer on devices or sequential files.
Terminate Output Writer (TO)	- terminates the Output writer on devices or sequential files.
for full information see the relevant commands in Section VI.	

Table 4-3. Output Commands

## OUTPUT PRIORITIES (OPR)

Outputs are assigned priorities in a similar way to that in which jobs are assigned Scheduling Priorities. The OPR is used to queue outputs waiting to be treated by the WRITER service job.

Output Priorities are in the range of 0 - 7, with 0 being the highest. The outputs are queued in priority order and on a FIFO basis for the same priority. Outputs waiting to be treated are said to be in the WAIT state.

Output Priorities are either given explicitly in the job description or implicitly by default in relation to the output class.

Output Priorities can be ascertained with the Display Output (DO) command and modified with the Modify Output (MO) command.

## OUTPUT CLASS (OC)

There are 26 Output Classes in the range of A - Z; these are used to control outputs in a similar way to jobs being controlled by Job Classes.

Each Output Class has an associated default output priority, the value of which is set at system configuration. This default value can be modified by the Modify Output Class (MOC) and ascertained with the Display Output Class (DOC) command. The default output priority is used whenever an output of that class has not explicitly been given a priority.

The class of an output is given explicitly in the job description, if it is not then a default class is assumed. The class to be used as default is specified at system configuration.

The class of an output, as well as the number of outputs in a class can be ascertained with the Display Output (DO) command. The class of an output can be changed using the Modify Output (MO) command.

Each Output Class can be started or terminated on each output device, it is also possible to start more than one class on a device which is the normal case (see Start Output command in section VI).

If at system configuration no values are specified, the following default values will be assumed by GCOS:

```
OUTPUT CLASS -   A B C D E F - Z      CLASS "C" IS THE DEFAULT CLASS
OUTPUT PRIORITY - 1 2 3 4 5   6
```

## OUTPUT SELECTION

When an output enters the wait state it is put into the output queue as the last one of its priority, this is also true for a held output even though it is not eligible for selection.

In certain circumstances the incident is more serious and in this case message OU27 is displayed and followed by question OU02 (for OU02 see this section). The general format of OU27 is:

```
OU27 ddnn ron.out sequence [ {HELD      } ] [WRITER TERMINATED]
                             {CANCELLED}
```

[RC = cccccc → sru, return code] [ON SYSOUT FILE]  
(see Console Messages manual)

After the operator has recovered the device and answered OU02 the output treatment resumes at the point specified in the reply to question OU02.

Normally message OU27 ends before HELD or CANCELLED and the above holds true; if however the output has been HELD it must be released before treatment is resumed (in fact Output Writer may begin treating another output as soon as question OU02 is answered).

If CANCELLED is shown the output has been cancelled and in the case of a standard output it will have been erased.

If WRITER TERMINATED is shown the Output Writer will have to be restarted on that device.

Note : If question OU02 was not asked, after recovery the Output Writer will start printing from the current line; this may result in that line being duplicated. In this case mark the duplicate line and inform the user.

#### Incidents on the Volume Containing the Output File

These incidents are treated like any other tape of disk incidents (see section V) and if recovery is successful the output will continue without any further operator action. It should be mentioned that tapes are automatically repositioned if the recovery was successful.

If the incident could not be recovered the operator issues a Cancel Request (CR) command. Message OU27 is displayed indicating that the output has been HELD, question OU02 is then asked. After the operator has replied to OU02 the Output Writer will select the next available output. The output must be released before it can be treated by Output Writer again.

## Paper Form or Card Deck Cannot Be Used (or Print Belt)

If the output media cannot be used this may be due to an error in the \$WRITER or \$OUTPUT JCL parameters; in this case message OU27 is displayed indicating that the output is HELD, the return code contains information concerning the erroneous parameter. The operator response is indicated by the return code, refer to the JCL Reference manual.

If the output media cannot be used for any other reason, in particular issuing a Cancel Request command in reply to a mount media request, then message OU28 is displayed:

```
OU28  ddnn  WRITER TERMINATED  RC = ccccccc  siu,return-code
```

The Output Writer is terminated on the specified device and the output being treated is left in the WAIT state. The Output Writer can be started again on that device but only for outputs that do not need the unavailable media; thus it is advisable to hold these outputs before starting the Output Writer.

## The Sysout File Cannot be Used

If a Permanent Sysout file cannot be found, i.e. the wrong file was specified or the file has been deleted, then message OU26 is displayed, indicating the output sequence number of the file in error.

```
OU26  ddnn  ron.output-sequence-number CANCELLED  
      RC = ccccccc → siu, return code  
      EFN = efn [SUBFILE = subfile-name]
```

The output is cancelled and Output Writer selects the next output for treatment.

In any other case where the Sysout file cannot be used, i.e. the operator refused to mount the Volume containing the file, then message OU27 is displayed showing that the output has been HELD and that the Output Writer has been terminated on the indicated device.

## The Sysout File is Incorrect

Sometimes unedited Permanent Sysout files have errors that prevent Output Writer from being able to print a line. In these circumstances message OU27 is displayed and then question OU02 is asked. The operator should reply CONT to OU02 and printing will resume on the next line, this bypasses the line at fault. For each line that is at fault the above messages are displayed. If there are a large number of these errors the operator should use the HOLD option in reply to OU02 and consult the user.



## SECTION V

### PERIPHERAL OPERATIONS UNDER GCOS

The purpose of this current section is to describe the "software characteristics" of the devices: how each device type (disks, printers etc.) behaves under GCOS.

The section is therefore broken down into subsections which describe the characteristics of the device and its volumes, how these volumes are mounted, and incident recovery. The following topics are discussed:

- |   |  |
|---|--|
| . Device Handling Overview                    | introduces the terminology, characteristics, commands and messages common to all devices |
| . Disk, Magnetic Tape and Cassette Operations | } describes the operational characteristics of each device class                         |
| . Printer Operations                          |  |
| . Card Device Operations                      |  |
| . Hardware Incidents                          | describes incidents, recovery, error logging etc.  |
| . Monitoring Device Incidents                 | describes use of Device Trace commands to monitor occurrence of device incidents.        |

#### DEVICE HANDLING OVERVIEW

Level 64 operators will need to understand the following terminology: (key terms are listed in the left-hand margin).

DEVICE TYPE                      Each device belongs to a device type identified by two letters (e.g. PR = printer). Each device in a device type is identified by a device number (two digits).

DEVICE IDENTIFIER              The device identifier is made up of 2 letters for the device type and two digits for number (e.g. PRO1)

MEDIUM Each device uses one or more types of medium : disk pack, magnetic tape reel, paper form, printer belt, card deck, paper tape...

VOLUME Disk packs and tape reels are referred to as volumes. A volume contains in fact one or more files, each of these files being, for the user, a set of information.

FILE

PREMOUNTING The action of loading a volume on a device before it is actually needed or requested is described as premounting.

MOUNTING The action of loading a volume on a device only when it is requested by a job is described as mounting. If a mounting request from a job cannot be satisfied, the operator indicates this to the system using a Cancel Request command.

AVR Whenever a device is put into the Ready state, the system tries to identify the volume. This operation is called Automatic Volume Recognition.

AVR is capable of identifying Level 64 native disks and magnetic tapes. G100 disks and magnetic tapes, and H200 disks. The facility is also available for print belts which are identified by means of a hardware "label" which exists on each belt.

For these volumes, no operator action is required other than mounting.

Non-standard disks and magnetic tapes have to be presented to the system by the operator using the Name Volume command.

VOLUME NAME Disk and Tape volumes are identified by their Volume Name (vol-name).

[Note : The term volume name is used throughout this manual for convenience, but the volume name may in fact be alphabetic, alphanumeric or numeric.]

DEVICE  
RECONFIGURATION

A job asks for access to a volume by specifying its volume name. If the volume is not premounted the system suggests in the mounting request a device to be used. If however the operator mounts it on another device AVR will read the volume name and direct the job to this new device. This operation is called device reconfiguration. It can be performed only for disks and magnetic tapes and only when the user has not specified a particular device in the JCL statements.

INCIDENT  
RECOVERY

While using a device some incidents may occur. If the intervention of the operator is needed to correct this situation a recovery request is issued by the system. The operator can then correct the incident but if it is not possible he warns the system using the Cancel Request command. All incidents are recorded in the SYS.ERLOG system files. The system provides functions for handling these files and, when necessary, for excluding a faulty device from the configuration by use of the Hold Device command. (The Release Device command enables the operator to put this device back in the configuration.)

DEVICE STATES

It follows that a device may be in various states of availability. These device states are described below.

Device States

Under GCOS a device is always in one of the following Device States:

. Held

The device has been taken out of the system configuration by use of the Hold Device command. This can be performed when the device is free. It can be put back into the configuration by use of the Release Device command. (These 2 commands are described in Section VI). A device can also be unavailable because it is not known to the system; see "Unknown Device", below.

. Free

The device is not being used. It may be in the hardware state Power off, Standby or Ready. If it is Ready a volume may be premounted (but a Name Volume command may still be needed to complete the premounting). When a volume is premounted the device may become allocated without any operator action.

. Allocated

The device is being used by one or several users. It is normally in Ready. When an incident has occurred a recovery request may be pending. If the recovery is denied, using a Cancel Request command, the device will become Free.

. Allocated in Device Mode

The device is being used directly by a user. All actions concerning this device are the responsibility of the user. [See "Device Allocation in Device Mode"].

To ascertain the state of a device use the Display Device command.

### Device Allocation In Device Mode

When a job needs the exclusive use of a device it can request (in its JCL) that the device be allocated in device mode.

When the device is allocated in device mode the system displays the message :

```
DV23 dev-id ALLOCATED IN DEVICE MODE TO run
```

From that instant the handling of the device is entirely under the responsibility of the user. He will request mounting or dismounting of volumes through specific user messages (see 100 Program Mode Operator Guide).

Incident handling and device availability is still the system responsibility.

When a disk drive is allocated in device mode native disks should not be mounted on it; if a native disk is mounted the following message is displayed:

```
DV01 ddnn DISMOUNT MEDIA
```

All access to the disk is rejected until the device is no longer in device mode.

### Unknown Device

When a device that is not known to the system - i.e. a device not listed in the System Resource and Status Table (SRST) - is made Ready or set to Standby, the message:

```
AV11 UNAVAILABLE DEVICE (PC=xx, LC=xx)
```

is displayed, giving the physical path to the unknown device. The system must be re-initialized (start with pressing the Initialize pushbutton).

## DISK, MAGNETIC TAPE AND CASSETTE OPERATIONS

This subsection is divided into three parts:

- |                           |  |
|---------------------------|--|
| "Media Characteristics"   | describes the logical characteristics of the various types of the disk, tape and cassette volumes.                     |
| "Physical Operations"     | describes the system's responses to volume premounting, mounting and dismounting.                                      |
| "Disk and Tape Incidents" | discusses various non-fatal incidents which can occur when the system is reading or writing to disk, tape or cassette. |

### Media Characteristics

The following is a description of the logical characteristics of the different categories of disk, tape and cassette volumes which are supported by GCOS.

#### DISK PACKS

When GCOS first tries to access a disk pack, it looks for the volume name which is contained in the disk label. If the volume name is present and in the correct format, the disk is identified as Native; otherwise it is regarded as Non-native (non-standard).

#### Native Disks

The system is capable of recognizing four categories of native disk (see also Figure 5-1.):

1. System disk. The System disk is recognized as such because it is the disk from which Storage Load is performed. It must remain online and in Write Permit on the same disk drive throughout the whole session. This is because it contains the key system files, such as the system backing store (SYS.BKST).

Note : If any free space is available on the System disk it may be used to store user files.

2. Resident disks. The operator may define any disk as RESIDENT during GCOS Loading (See question ILOI in Section III).

Note : There is no necessity to define any Resident volume provided that all the system files are held on the System disk. Resident disks and the System disk should not be put in Write Protect, Standby or Powered Off.

3. User disks. These disks contain user files. Files belonging to more than one user may be held on the same disk.
4. Work disks. Any native disk with a volume name in the format of W.....X (where X is a character) is a work disk. A user can use space on a work disk without specifying the volume name. Work disks are prepared using the VOLPREP utility.

### Non-Native Disks

The system is capable of recognizing four categories of Non-Native disks:

1. G100 disks. These disks can be identified by GCOS but can only be read from or written to by 100 Program Mode.
2. H200 disks. These disks can be identified by GCOS and can be accessed either through 200 Program Mode or the HFAS method.
3. Non-standard. When a disk is found to contain legible data in the correct recording density but with no recognizable volume-name, it is treated by GCOS as a Non-Standard Volume. In order for a user job to access such a disk, the operator must identify the disk to the system by using Name Volume command.
4. Unknown. When GCOS finds that a disk contains either nothing at all or nothing intelligible (e.g. a disk with the wrong recording density) the disk is indicated as Unknown by Automatic Volume Recognition, but can from that point onwards be treated as a Non-Standard disk.

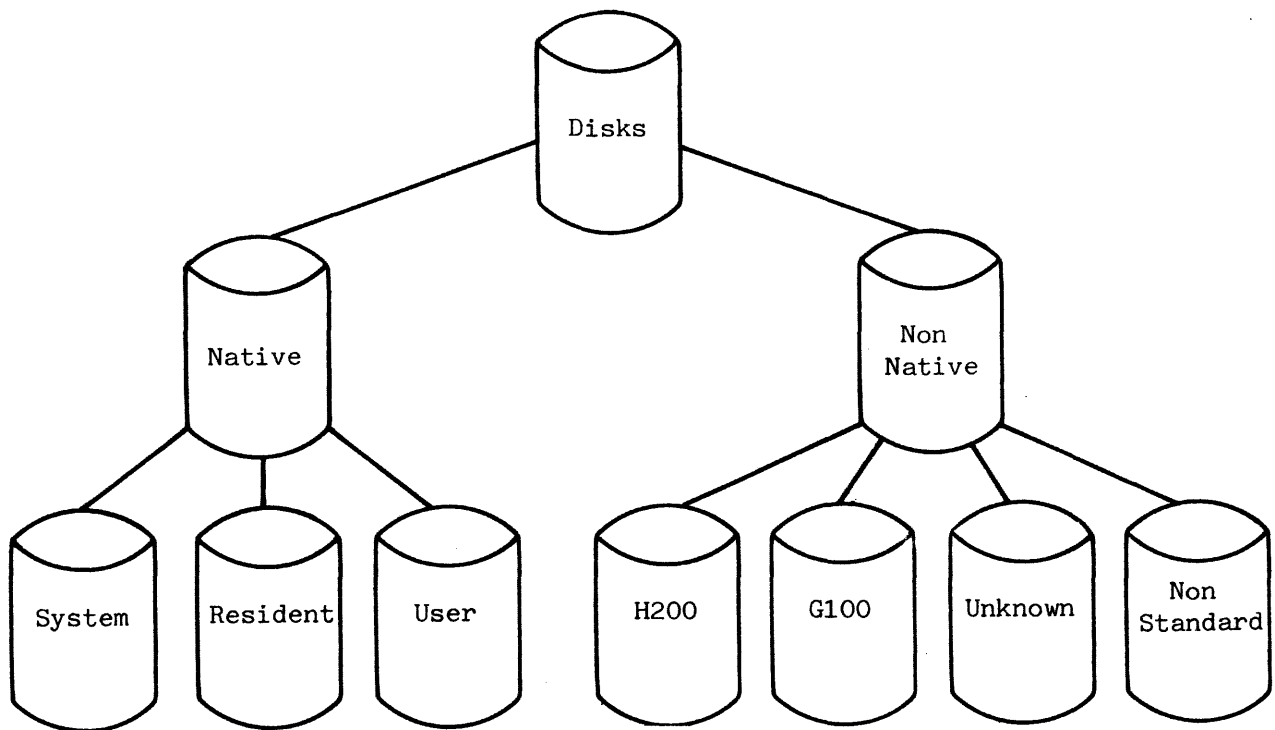


Figure 5-1. Summary of Disk Categories

## MAGNETIC TAPE REELS

A reel of magnetic tape may display various physical characteristics: it may contain 7 or 9 tracks and may be recorded in a density of 200, 556, 800 or 1600 b.p.i. It is therefore essential that it be mounted on a drive which has the corresponding physical characteristics. If not, it will be treated as an Unknown Volume (see below).

In addition to these physical differences, a tape may display logical differences which are detected by GCOS and used to categorize the tape as either Native or Non-Native : see below (and summary in Figure 5-2).

### Native Tapes

When GCOS finds that a tape contains a standard tape label with a volume name in the correct format, it treats the tape as a Labelled Native tape. In other cases, the operator must identify the tape using the Name Volume Command.

Labelled Native tapes belong to one of two types:

1. Work tapes. These are created by the \$VOLPREP utility. They are recognized as Work tapes by their tape labels.

Any job may ask for a Work tape which is then mounted by the operator and is normally assigned only for the duration of that job after which it reverts to a pool of Work tapes which is controlled by the Operations Manager.

If a work tape is used for dynamic permanent-file extension they lose their work quality. The operator is informed by the following message:

```
FPO1 ron TAG media FOR EFN external-file-name
```

The operator should then note that the tape belongs to the indicated file and inform the user accordingly.

2. User tapes. These are created by user jobs and are used to store permanent files. These tapes belong to the user who created them either until the expiry date (specified in the tape label) has been reached, or until they are released by a user action.

Note : Any user tape whose expiry date has been reached can be overwritten by a job which specifies the correct volume-name (assuming that Write Permit is enabled).

## Tapes Without a Native Label

Tapes without a Native tape label fall into one of four categories:

1. G100 tapes. These are identified by GCOS but can only be read from or written to by 100 Program Mode.
2. G100 Work tapes. These are recognized by GCOS simply as G100 tapes, but the operator can use the WORK parameter in the Name Volume command to tell the system that they are available as G100 Work tapes.
3. Unlabelled tapes. These are tapes which contain recognizable data but which do not have a standard native tape label. H200 tapes are treated as unlabelled even if they have a H200(0) tape label. If a job wishes to use such a tape, the operator must introduce it to the system with the Name Volume command.
4. Unknown tapes. A tape is regarded by GCOS as Unknown if it contains either nothing at all or nothing in any recognizable format. Thus a 7-track tape mounted on a 9-track drive is treated as Unknown, as is a tape recorded in a density different from the density of the tape-drive. If a job requires to write on to an Unknown tape, the operator must specify the tape name with the Name Volume command.

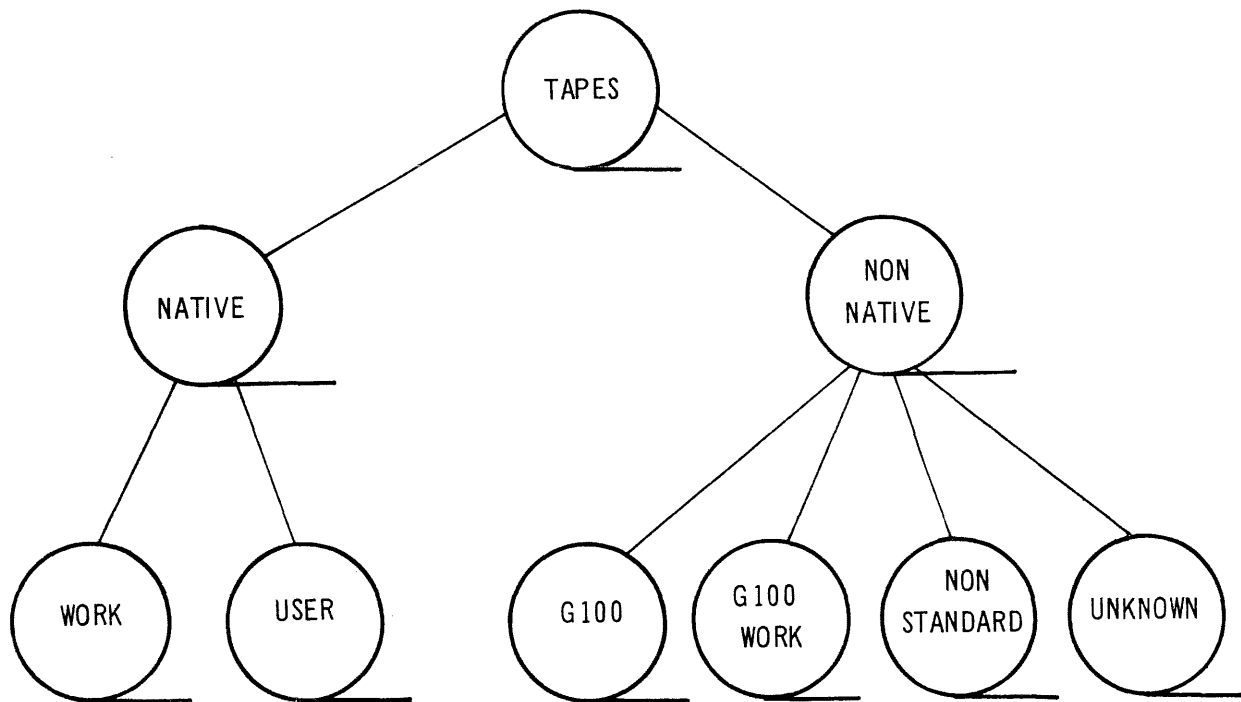


Figure 5-2. Summary of Magnetic Tape Categories



## CASSETTES

Cassettes can be Native, Compact, or Non-standard depending on the way they are accessed by GCOS and the user (see Figure 3-1 below)

On each cassette, except non-standard, there is a volume name indicating the cassette name. On non-standard cassettes the operator has to indicate the name using the Name Volume (NV) command.

If GCOS is unable to read the cassette it is referred to as Unknown and is treated as non-standard. This is the normal case for new cassettes.

Some native cassettes are Work cassettes and can be used for cassette file extension. If this happens the FPOI message is displayed as for a Work tape (see note above).

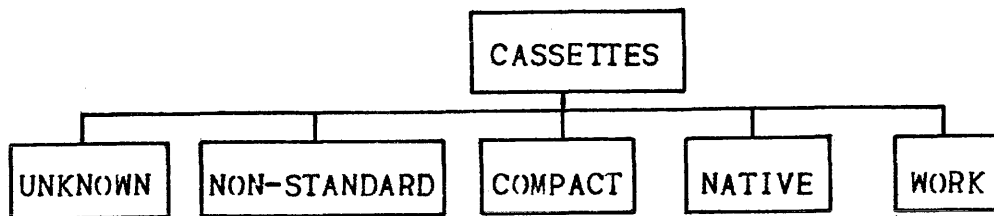


Figure 5-3. Summary of Cassette Categories

Cassettes have two sides and side 1 must always be used before side 2. Side 1 also contains the volume label so when a cassette is changed to side 2 it must be mounted on the same device.

When a cassette needs to be mounted on side 2 the following message is displayed:

```
hh.mm CSnn MOUNT vol-name { NSDT
                             COMPACT } SIDE 2
                             NATIVE }
```

When a cassette handler is switched to standby GCOS is not aware of it. Thus a Display Device command will show the last situation known by GCOS. If an I/O operation is attempted on a handler in standby a recovery request is issued. Another result of this situation is that a cassette is considered to remain premounted until another cassette is mounted and its label read.

Unlike a tape, cassettes are not repositioned after a device incident.

Apart from the above mentioned differences cassettes are treated in a similar way to tapes; in particular for premounting, mounting, dismounting and incidents (including the write protect mechanism).

## Disks, Tapes and Cassettes: Premounting, Mounting and Dismounting

This subsection describes the way in which the system treats the physical disk and tape operations of premounting and mounting volumes (and the important distinction between them). Finally the system's treatment of volume dismounting is discussed.

### PREMOUNTING DISK AND TAPE VOLUMES

The operator is able to save computer time by anticipating requests by jobs for disk and tape volumes. He does this by premounting the volume before the job has asked for it.

The volume is loaded on any available drive (with the required characteristics) and the drive is set to Ready. The system now tries to recognize the volume and it may or may not succeed:

- . If the volume has been recognized, the following message is displayed:

```
hh.mm ddnn PREMOUNTED vol-name [type]
```

The premounting is complete. (See Note for premounting Work tapes and G100 Work tapes.)

- . If the volume has not been recognized, the system replies with one of the following:

```
DV22 ddnn NON STANDARD VOLUME
```

```
DV24 ddnn UNKNOWN VOLUME
```

The name of the volume is then indicated using the Name Volume command. After this the following message is displayed:

```
hh.mm ddnn PREMOUNTED vol-name NSTD
```

Note : If a premounted tape is a Work tape, any job can use it as such without any further operator intervention. If resources permit, the operator may even wish to keep a Work tape permanently mounted for use as required.

If the tape is to be used as a G100 Work tape the operator must first premount it and then indicate that it is a G100 Work tape using the Name Volume command. After this the following message is displayed:

```
hh.mm Mttn PREMOUNTED vol-name "WORK G100"
```

## NORMAL MOUNTING OF DISK, TAPE AND CASSETTE VOLUMES

When a job requires a disk, tape or cassette volume and the operator has not anticipated this requirement by premounting it, the system displays the following message:

```
hh.mm ddnn MOUNT { "WORK G100" } [type] FOR rom(SIDE 1)
                  { vol-name }          (SIDE 2)
                  { WORK }              [IMP][PROTECT]
```

WORK indicates that a native WORK tape or cassette is required.

The operator should then take the following actions:

1. If necessary, dismount the volume currently loaded on the device.
2. Mount the requested volume in the following cases:
  - . If IMP is in the message, on the indicated device, if any other volume is mounted on this device message DV02 is displayed.
  - . If IMP is not specified, on any free device (including the device specified) which is of the same type and characteristics.
  - . If PROTECT is specified in the message the Write Permit ring should be removed from the tape or the Write Enable plug removed, from the cassette.
3. The device is switched to Ready.

The subsequent events depend on the type of volume:

- . For Native, G100 or H200 disk and for Native or G100 tape (except G100 Work tape)

The mounting is completed. No message is displayed.

- . For G100 Work tape

A Name Volume command must be issued to indicate that the tape can be used as G100 Work tape.

After that the mounting is completed. No message is displayed.

- . For Non-Standard Volume

The following message is displayed:

```
DV22 ddnn NON STANDARD VOLUME
    or
DV24 ddnn UNKNOWN VOLUME
```

A Name Volume command must now be issued to indicate the name of the volume. After that the mounting is completed.

If the mounting request cannot be satisfied (because the volume is not available) or should not be satisfied (because the volume is not to be over-written) the operator answers the mounting request with a Cancel Request command. This will cause the job to be aborted, unless the corresponding file was an 'optional' file.

If the magnetic tape PROTECT has been indicated it is mandatory to mount the tape without the write permit ring. If this is not done, GCOS will display the message:

\* DVO6 MTnn SWITCH vol-name FOR ron TO PROTECT

Write PROTECT for a disk is not checked at mounting time. If PROTECT is shown for a disk, the drive the disk is mounted on should be switched to PROTECT before switching to READY.

## DISK AND TAPE DISMOUNTING

A volume can be dismounted when the device is not allocated. When a mounting request is issued, the device indicated in the message is always unallocated and can be used. To ascertain whether a particular device is unallocated, the Display Device command may be used. GCOS is not aware that a cassette is dismounted until an I/O operation is performed or a new cassette is mounted.

The dismounting operation is not acknowledged by the system in any way (unless, of course, the device was allocated to a job, see the next paragraph). A user may specify in the JCL that a tape is to be dismounted after its use, in this case the tape is fully rewound and the tape deck window lowers.

## Disk and Tape Incidents

Incidents involving disk or tape devices can be of the following nature:

- . An input/output operation attempt is unsuccessful, but the device affected remains in the Ready state.
- . Transition of the device into Standby or Power Off: this may result either from a physical action (the operator pressing the appropriate button and/or dismounting the volume) or from a hardware/firmware problem.
- . Write Protect violation on disk, tape or cassette: this leads to a system request for operator action which might involve the Use Device command or the Cancel Request command.

## INPUT/OUTPUT OPERATION INCIDENTS

Several retries (the actual number is dependent upon the type of operation) are attempted by hardware and software to complete an unsuccessful I/O operation in which the device affected has remained Ready.

If a retry is successful:

- . There is no notification (although the original incident is reported if the Start Device Trace command, described later in this section, was entered).
- . The event is recorded by Error Logging.
- . There is no effect on the step to which the I/O operation applies (the retries are invisible to the step).

If all retries fail it is an unrecoverable I/O error and the message

```
hh.mm ddnn I/O ERROR ON vol-name FOR ron [AT ccc/tt]
```

is displayed. The cylinder (ccc) and track (tt) location of the error are given when the error occurs on a Native disk. The step attempting the operation is notified of the unrecoverable condition and is then usually aborted.

Note : An unrecoverable I/O error involving a system file can cause a system crash.

## TRANSITION TO STANDBY OR POWER OFF

A disk or tape device which is allocated to a user or to the system may be switched into the Standby or Power Off state either by an operator action or as the result of hardware failure. Two cases may occur:

- . The drive containing the system disk has reverted to Standby or Power Off. This causes the following message to be output:

```
* AV15 ddnn SWITCH vol-name TO READY
```

The operator should then remount the system disk on the same device. If this is successful, no further action is requested and no message displayed.

If it is unsuccessful AV05 is displayed, there is clearly a hardware problem with the device. Hardware Initialization must now be performed from a different drive (but see also "Hardware Failure").

```
* AV05 ddnn MOUNT vol-name IMPERATIVE
```

AV05 is also displayed if another disk other than the system disk is mounted on this drive.

Note : As long as the system disk is in Standby, system activity is strictly limited by its inability to access system files.

- Another disk, tape or cassette device has reverted to Standby or Power Off. This causes the following message to be output:

```
*   DV12   ddnn   RECOVER vol-name FOR ron
```

The operator has a choice at this point. He may:

1. Try to recover : by remounting the volume which was on the indicated device either on the same device or on any other of same type and characteristics. (Note : if the mounting request for that volume was IMPERATIVE, it must be mounted again on the same device)
2. Give up recovery : by using the Cancel Request command to indicate this to the System. The step which tried to access the volume will be aborted. This should not be performed on Resident disks.

NOTE : If recovery was possible the I/O operation is retried. Tapes are repositioned before a retry except during open or close operations; or if they are used for a Volsave, Volrest, Filesave or Filrest utility.

## WRITE PROTECT ON DISK

A Write Protect pushbutton exists on all Level 64 disk drives. When Write Protect is on, no writing can be performed on the disk mounted on the drive.

Two abnormal cases may occur:

1. The System Disk is in Write Protect.

The System Disk (particularly the SYS.BKST file) is constantly being written to by GCOS. If it is put in Write Protect, the following message is displayed:

```
*   AV15   ddnn   SWITCH vol-name TO PERMIT
```

The operator must switch off Write Protect on the appropriate drive. No further action is required and no message displayed. While the System Disk drive is in PROTECT system activity is halted.

2. A disk (other than System Disk) is in Write Protect at a time when either the system or a user job wishes to write to it. This causes the following message to be displayed.

```
DV07  ddnn  SWITCH vol-name FOR ron TO PERMIT
        AND REPLY (UD, CR...)
```

The operator action depends on whether or not it is normal for the disk to be in Write Protect. (Note : Resident disks should not be in Write Protect.)

- a. Abnormal Write Protect - no request has been received by the operations staff for this disk to be mounted in Write Protect.

The operator must now make the disk available to the job which wishes to write to it. He may do this in either of two equally valid ways:

- i) - Switch disk to Write Permit (i.e. Write Protect off)

- Indicate this to the system using the Use Device command (see below).

- ii)- Dismount the disk.

- Remount it on another drive (or possibly the same one).

- In this case there is no need to use the Use Device command, but, of course, if the new drive also has Write Protect on, the message DV07 will be printed again.

- b. Normal Write Protect - the user has specifically demanded that this disk be mounted in Write Protect. The operator in this case must refuse to switch to Write Permit and indicates this to the system using the Cancel Request command. The step which attempted to access the disk will be aborted.

## WRITE PROTECT ON MAGNETIC TAPE

If a tape reel is mounted without its permit ring it is said to be in Write Protect. If, on the other hand, the permit ring is present the tape is in Write Permit. No writing can be performed to a tape in Write Protect.

If the system or a user attempts to write to a tape in Write Protect the following message is displayed:

```
* DV07 ddnn SWITCH vol-name FOR ron TO PERMIT
```

The operator action depends on whether or not it is normal for this tape to be in Write Protect (i.e. whether requested by the user):

- . Abnormal Write Protect (not requested by the user).

The operator:

- dismounts the tape
- remounts it with its permit ring

No further action is required and no message is printed.

- . Normal Write Protect (as requested by the user) i.e. the mounting request indicated PROTECT. The operator indicates to the System that he will not set the tape to Write Permit, by use of the Cancel Request command (see "Device Handling Overview"). In this case, the step which attempted to access the tape will be aborted.

## PRINTER OPERATIONS

All Printers use paper forms and a print medium such as belt or a drum. PUR0600/0800 printers use drums as their print medium; drums are fixed and cannot be changed. The PRU1200/1600 and PRU0640/0840 printers use a print belt as their print medium; print belts can be changed so that a different character set can be used. The system can recognize belts but cannot do so for paper forms.

The alignment of a paper form can be visually checked using the Print test routine.



These aspects of printer operations are described below under the main headings:

- . "Paper Form Operations"
- . "Print Belt Operations"
- . "Printer Incidents"

Note : During GCOS Loading the printer is handled in a manner different from the one described below.

### Paper Form Operations

An installation may define its own in-house paper form standards. Each type of paper form has its own characteristics: page-length, perforations, preprinting etc. Each of these form types is entered into a system file (called SYS.URCINIT) and is allocated a 4-character name by which it is known both to the users and to the system.

Thus a user job can call for a particular paper form to be used for its output. If this paper form is not mounted at the time of output - the system remembers which form is mounted - the system asks the operator to mount the correct form.

Once the correct form is mounted, output will commence. The handling of page lengths and paper throwing is performed automatically by the system which holds all the necessary details in its SYS.URCINIT file.

Paper premounting and paper mounting are described in more detail below.

#### PAPER PREMOUNTING

In order to premount a paper form the operator performs the following actions:

1. Switch the printer to Standby (if it is not already so).
2. Mount the paper.
3. Preferably leave the printer in Standby (see Note). If the printer is already allocated, switching it to standby will cause an incident and message

\* DV12 PRnn RECOVER media FOR ron

is displayed. This request is periodically repeated until it is either satisfied or cancelled.

At loading, GCOS treats the printer (if it is switched to Ready) as if the standard paper were mounted, and message:

```
hh.mm PRnn PREMOUNTED 0000
```

is displayed.

Note : If, after changing the paper, the printer is switched to Ready, GCOS will continue as if no paper change has occurred. This is why it may well be advantageous to leave the printer switched to Standby (in case the wrong paper is mounted) otherwise if a request is made for the paper which was mounted prior to the change, access will be given without any mounting request.

## PAPER MOUNTING

When a paper form is required by a job or the Output Writer (see Note) and is not already mounted, the system displays the message :

```
* hh.mm PRnn MOUNT vol-name FOR ron
```

and switches the printer to Standby with the Print Check Indicator on. The operator now performs the following actions:

1. Mount the paper
2. Then the operator can either:
  - Press the Print Test button to use the Print Test routine (described below).
  - Or press the Start button to switch the printer to Ready.

The mounting is now complete.

The following abnormal case may occur during paper mounting:

- . The request cannot be satisfied (for example the paper form is not available). The Cancel Request command should be used. If the paper form was requested by the Output Writer the WRITER service job is terminated and the output being treated is left in the Wait state; If it was requested by a user, the user step is aborted.

Note : If the user requests a print belt together with a paper form, mounting messages are issued for both; thus the needed belt can then be mounted at the same time as the paper.

## Print Test

The Print Test routine allows the operator to verify at any time that the stationery is correctly aligned. During the mounting of a paper form the system sets the printer to Standby with the Print Check indicator on. Once the paper is mounted, the operator presses the Print Test push button which has the following effect:

- . The system sets the printer to Ready.
- . The Print Test routine is executed.
- . The system resets the printer to Standby with Print Check lit and displays the message:
  - \* DV45 PRnn SWITCH TO READY

The operator can then visually check that the mounting is correct:

- . If the paper form alignment is not correct it can be corrected and Print Test can be used again.
- . If it is correct, switch the printer to Ready.

The print test routine can be performed at any time by switching the printer to Standby (if it is not already so) and pushing the Print Test button. The result is the same as that described above.

Note : The standard print test (a user can specify another print test, but it must be stored in the SYS. URCINIT file) is the printing of a full line just underneath the paper fold. The line is:

```
SYS. URCINIT CREATION DATE mm/dd/yy EEE.....EEE
```

## Print Belt Operations

The PRU 1200/1600 and PRU0640/0840 printers may be supplied with a choice of print belts which correspond to a range of different character sets which are available to users. Each print belt contains a hardware "label" which is recognizable by the system in the same way that disks and tapes are identified. Users are therefore able to specify in their JCL the name of the character set required for their output, although most users will probably be content with the standard set. No action is required and no message displayed in cases where the belt containing the required character set is already mounted.

When a different belt is needed, however, the operator is required either to premount it or to mount it. The procedures, which are not unlike those for disk and tape, are described below.

#### PRINT BELT PREMOUNTING

If a job is running and is known to require a special print belt, the operator may (subject to the needs of other jobs) anticipate the system mounting request.

The procedure is as follows:

1. Set the printer to Standby if it is not already so.
2. Change the belt.
3. Set the printer to Ready.

No message is displayed and the premounting is complete.

If the operator mounts a belt on a printer that is allocated and the new belt has an incompatible character set the following message is displayed:

```
* DV01 PRnn DISMOUNT MEDIA
```

The operator should dismount the belt and remount the belt that was used before.

If GCOS can not recognize the belt for any reason the following message is displayed:

```
DV19 Prnn UNKNOWN BELT
```

The operator should repeat the premounting operation, if the problem persists he should try another belt with the same character set. If this does not succeed the Field Engineers should be informed.

#### PRINT BELT MOUNTING

When a specific print belt is required by a job and the system ascertains that it is not already mounted, the following console message is displayed:

```
* hh.mm PRnn MOUNT FOR ron A BELT: belt-name,belt-name...
```

GCOS automatically switches the printer to standby if it is not already so.

In order to meet this request the operator performs the following actions:

1. Change the Belt to one of those indicated.

2. Set the printer to Ready.

The mounting is now complete. No message is displayed.

The following abnormal situations may occur after the "mount" message:

- . The request cannot be satisfied (for example none of the indicated belts is available).  
The Cancel Request command should now be used. If the belt was requested by the Output Writer the WRITER service job is terminated and the output is left in the Wait state, if requested by the user, the user step is aborted.
- . The name of the mounted belt is not one of those specified by the "mount" message. The system repeats the "mount" message and the operator now either mounts the correct belt or uses the Cancel Request command as above.

- Notes:
1. If both paper form and belt are requested by separate mounting messages both can be mounted at the same time.
  2. It is possible for a single character set to be contained in two or more print belts. This explains why the "mount" message may specify more than one belt. It is up to the user to say which belt he wants in such cases.

### Printer Incidents

Most printer incidents (end of paper etc.) result in the printer being set to Standby. Some more serious incidents result in the printer being switched to Power off. When an incident occurs, and the printer is being used, the following message is displayed:

\* DV12 PRnn RECOVER media FOR ron

The operator may either:

- Try to recover: correct any obvious cause (end of paper, paper jam, printer open...), then make the printer Ready. Printing resumes with the line at which the incident was detected, which may result in a line or two being duplicated. If the incident involves a declutched tractor, press the SKIP button prior to making the printer Ready, to avoid a line overlapping the previously printed line.
- Give up the recovery: inform the system of this choice through the Cancel Request command. If the printer is being used by a step, that step is aborted (100 and 200 Program Mode jobs are not aborted but the emulated job is). This is also used to ask the Output Writer for a page recovery (see Incidents on Output Writer, Section IV). For Power off conditions the job will react as if a Cancel Request command had been issued.

In certain cases (normally after a Power Off incident) GCOS may indicate that it is unable to recognize the printer by giving the message :

DV17 PRnn MEDIA RECOGNITION IMPOSSIBLE  
RC = cccc → siu ,return-code

If this should happen, press the STOP and then the START pushbuttons on the printer.

## CARD DEVICE OPERATIONS

A Card Device is defined as either a card reader, a card punch or a card reader-punch. All these devices belong to the type CD.

The medium used by the card devices are the cards themselves. As with paperforms the system cannot recognize card decks, it has to be informed.

The main user of the card reader is the Input Reader. The main user of the card punch is the Output Writer.

Note : The card reader is handled in a non-standard way during GCOS Initialization.

As with other media, cards may be either premounted or mounted. These operations are described below.

### Card Deck Premounting

1. Job description deck and Stream Reader activation  
(only for card reader or card reader/punch in reader-mode)
  - Switch the card reader to Standby if it is not already so  
(and possibly take out all remaining cards)
  - Put the deck in the hopper
  - Switch the card reader to Ready

Note : This results in the activation of the Stream Reader.

- 2) Other cases on card reader
  - Switch the card reader to Standby if it is not already so  
(and possibly take out all remaining cards)
  - Put the deck in the hopper

- Leave the card reader in Standby otherwise the Stream Reader will be activated.

### 3) Card punch

- Switch the card punch to Standby if it is not already so (and possibly take out all remaining cards)
- Put the deck in the hopper
- Leave the card punch in Standby or switch it to Ready

Note : As long as a card punch remains in the Ready state GCOS remembers the name of the card deck and will give access to it if requested without issuing a new mounting request. As soon as the card punch is switched to Standby this name is forgotten.

### Card Deck Mounting and Device Modes

When a job requires a card deck to be mounted (either for reading or punching), the system outputs the following message:

```
*   hh.mm   CDnn   MOUNT   media   FOR ron{PUNCH }
                                     {CARD51}
```

where PUNCH is 80 column cards in punch-mode (indicated by PROTECT on card reader/punch devices)

CARD51 is 51 column cards in read-mode

If neither PUNCH or CARD51 is indicated 80 columns cards in read-mode (for card reader/ punch device) is assumed. On a card punch device PUNCH is never indicated.

GCOS switches the device to Standby if it is not already so.

The operator actions are to:

1. Take out all remaining cards
2. Switch the device to the right mode and put the requested card deck in the hopper
3. Switch the device to Ready

The mounting is then completed. There is no message.

The Following abnormal cases may occur during card mounting:

- . If the mounting request cannot be satisfied, use the Cancel Request command, the job requesting the cards is aborted; see Stream Reader in previous section.
- . If the device is not in the correct mode after mounting GCOS switches it to standby and reissues the mounting request; the operator switches to the correct mode and switches the device to ready.

### Card Device Incidents

Most card device incidents (read error, punching error ...) result in the card device being put in Standby. Some more serious incidents result in the card device being switched to off-line. Incidents also occur if the Stop or Power Off pushbuttons are used.

- . Device not being used

No message is displayed. The operator may recover the device (if he wishes) or wait until a mounting request.

- . Device being used

The message:

```
*   DV12  CDnn  RECOVER media FOR ron, [ {READ AGAIN LAST CARD    } ]  
                                     {DELETE LAST PUNCHED CARD }
```

is displayed. The following three paragraphs explain the operator actions required by the various forms of this message:

- . RECOVER (only)

- Take out the card remaining in the card device by using the Runout pushbutton (it is possible that there is no such card)
- Put back the card taken out (if any) as the first of input deck
- Correct the cause (no more cards, \$EOS missing, throat jam..)
- Switch the card device to Ready

- . READ AGAIN LAST CARD (2)

- Take out the card remaining in the card device by using the Runout pushbutton



- Put back the last card of the output deck and the card just taken out as the first and second of the input deck.
- Switch the card device to Ready.

. DELETE LAST PUNCHED CARD

- Reserve or mark, according to operating instruction, the last card in the stacker
- Switch the card punch back to Ready

Note : 1. In case of a power off incident it is not always possible to determine exactly the last card to be reread or repunched. A specific repositioning is therefore needed. If the user has allowed for checkpointing (\$CKP cards) then it is possible to do a checkpoint/restart (using cards as input only); if this is not the case the step will usually abort.

2. If a Card Reader is switched, because of an incident, to Standby the READ indicator is, most of the time, on.

If it is Off two cases are possible

- Card being read in Hollerith code:

- . The last card read contains 2 holes in line 1 to 7 of the same column where only one is legal. It should be corrected before being reread.

- Card being read in binary:

- . The last card read contains in column 1 the perforation of a \$Hollerith code. This character (being protected) has been detected. Reread the card and it will be read correctly.

## HARDWARE INCIDENTS

Hardware incidents, and the way in which GCOS reacts to them, are discussed below at three levels:

- . Device Incidents
- . Peripheral Processor Incidents
- . Error Logging

## Device Incidents

GCOS provides a set of functions which help in maintaining devices in a good operational state. When incidents do occur, these functions allow GCOS to handle them with as little damage to user programs as possible.

The handling of incidents on each particular type of device has been described in the corresponding incident paragraph above.

More serious incidents might cause the peripheral processor to fall in the "stop" state - Specific operator action is then required.

All incidents on devices, or peripheral processors are recorded by the Error Logging mechanism, for further use by Field Engineers. Specific commands are provided for Error logging control (see Error Logging later in this section).

Two commands (Hold Device and Release Device) are provided to take a device out of the configuration (for handling by Field Engineers, or to prevent its use by GCOS) and then to put it back in the configuration.

The Device Trace mechanism provides the operator with a visibility of device marginal conditions, and the Field Engineer with a help for on site device maintenance.

Tests and Diagnosis programs are also provided to help in problem detection and diagnosis. These will not be described in this manual, but for further information see the "System Operation: Checkout Procedures" manual.

## Peripheral Processor Incidents

If an incident occurs on one of the peripheral processors, the GCOS switches the processor to Stop and reacts as follows :

- . Mass Storage Processor failure which controls the drive the System disk is mounted on: since the System disk can no longer be accessed, no more useful work can be performed and the following message is output:

\* AVOI GCOS : CRASH, NO DUMP

- . Magnetic Tape or Mass Storage Processor failure : the following message is output:

\* hh.mm ddnn PSI n FAILED

It is possible for the system to still run but the devices controlled by the processor cannot be used.

- Integrated Unit Record Processor failure: since the operator's console is connected to the IURP, no message is displayed. Evidence of the failure is provided by the fact that the console is "dead" and the CONSOLE CHECK and SUBSYSTEM ERROR indicators are illuminated on the SOP. [Note : if only the console typewriter has failed, there will be no indication on the SOP].

In any of the above cases the operator procedure is the same:

1. Note any abnormal indications on the SOP
2. Perform the appropriate Checkout Procedure (see "Error Symptoms" in the System Operation: Checkout Procedures manual)
3. Perform a complete initialization

### Error Logging

Every event which could be of help to the Field Engineers is recorded in a system file called SYS. ERLOG.

Such events are, for example, the successful retry of an I/O operation or a marginal condition detected by a peripheral.

It is essential to maintain enough free space in the SYS. ERLOG file otherwise information may be overwritten and lost. When half of the file is full, the messages:

```
*      ELO1   SYS.ERLOG FILE 50% FULL
      ELO3   RUN PRLOG
```

are displayed.

The operator should now run the PRLOG utility by typing in:

```
SJ PRLOG
```

This will cause the execution of the PRLOG utility, which will print the information stored in the SYS.ERLOG file and free the corresponding space.

If the PRLOG utility is not run before the SYS.ERLOG file is completely full, the message:

```
EL01  SYS.ERLOG FILE FULL
```

is displayed, and if PRLOG is still not run before SYS.ERLOG has been completely overwritten then the message:

```
EL01  SYS.ERLOG FILE FULL  
EL01  DATA LOST FROM date-1 TO date-2
```

is displayed, indicating the dates between which all the information in the SYS.ERLOG file have been lost.

Certain abnormal situations will lead to the following messages being displayed:

```
EL01  SYS.ERLOG FILE UNAVAILABLE
```

This message, indicating an error, (probably on the SYS.ERLOG file) will cause Error Logging to work in "Reduced Mode". In this state PRLOG cannot be run and the Device Trace commands cannot be used. It can also indicate that due to an error the whole set of Error Logging functions are unavailable. This is called "Trace Only Mode" because all events that would have been stored on the SYS.ERLOG file are displayed on the console instead.

```
EL01  SYS.ERLOG FILE REINITIALIZED
```

This message is issued when the System Disk is used on a different hardware system and informs the operator that the previously stored events are erased as they do not relate to the current system.

The operator can write comments on the SYS.ERLOG file using the Write Log (WLOG) command. The Display Log (DLOG) and Modify Log (MLOG) commands are provided for use of the Field Engineers. See the comments on these commands in Section VI.

## MONITORING DEVICE INCIDENTS (DEVICE TRACE)

For each device in the configuration, a threshold value is declared at system generation which defines the acceptable number of logged events on a given device. When that threshold is reached, the message:

```
* AV16 ddnn n% EVENTS (events-i/o operation)
```

is displayed, giving the number of logged events over the number of input/output operations (and the corresponding percentage).

If the threshold declared at generation proves to be unrealistic, the value can be revised for a device class through the Modify Device Trace command. The operator can ascertain the present value set for a device class (or all classes) by using the Display Device Trace command, which also displays the number of events detected so far for each device of the class (or all classes), the number of events can be reset to zero by use of the Modify Log command or by use of the PRL0G utility.

It may also be appropriate at some time (at the direction of Field Engineering, for instance) to implement the Start Device Trace command; this command causes an informative message to be displayed each time an event is detected. A Terminate Device Trace command discontinues this activity.

The Start Device Trace command causes the following message to be displayed for the original I/O incident and for the last retry of the I/O operation.

```
AV17 ddnn IOcc = cccc EV = cc PSB = cc.... DSB = cccc....
```

The Device Trace commands are described in Section VI.



## SECTION VI

### OPERATOR COMMAND LANGUAGE

This section describes the commands which are available under the GCOS Operating System. The commands are listed in alphabetic order for easy reference. A list is given of the commands, with their parameters.

The normal response by GCOS is given after every command; for any other response see "System Operation : Console Messages".

The variable parts of the messages and commands are listed below:

ddnn - device number  
devclass - device class  
dpr - dispatching priority (0 to 9)  
efn - external file name  
function-name - system function name  
jclass - job class (A to P)  
jclasslist - job class list  
job user - job name user name  
j-state - job state (IN, SCH, HOLD, EX, SUSP, OUT)  
libname - library name  
lm-name - load module name  
multi - multiprogramming level  
oclass - output class (A to Z)  
oclass list - output class list  
oname - output name  
ostate - output state (KNOWN, READY, WAIT, HOLD, OUT)  
ostation - output station  
osequence - output sequence number  
out-devclass-output device class  
ron - run occurrence number (eg X12)  
sfn - subfile name  
spr - scheduling priority (0 to 7)  
volname - volume name

Where ddnn appears, it refers to a particular device (e.g. MT03 refers to magnetic tape unit number 03).

Device class refers to the specific "type" of device. For example there are several classes of disks, but each specific disk class is a device class (i.e. MS/M400 and MS/M350 are two separate device classes).

Note : when a list is indicated (i.e. ron list or class list etc...)

the list can either be entered in full or, if consecutive, just the first and last entries in the list separated by a hyphen (-).

Using a ron list for example:

- 1) X32/X34/X35      (just these 3 rons)
- 2) X14/X15/X16/X17/X18/X23      (example 2 and 3 will have the same effect. The two methods of forming lists can be mixed)
- 3) X14-X18/X23



## BROADCAST TELECOMMUNICATION COMMAND

This command can be used to broadcast a message to a specific terminal or all terminals. It can also be used to send messages to a program using communications or a program queue (MCS jobs).

```
BT   { terminal
      { program
      { program queue } text
      { ALL
```

### Normal case

The text is broadcast to the specified recipient.  
The text must not exceed 128 characters, if it does it will be truncated.

## CANCEL LOAD MODULE COMMAND

This command allows the operator to de-preinitialize a preinitialized load module coming from a certain library.

CLM    lm name [:libname]

Normal case

The load module is deleted from backing store. There is no message.

If the load module is busy the command must be retyped later.

## CANCEL MAIN MEMORY COMMAND

This command is used to remove system functions, that have been made resident by the PMM command, from main memory.

CMM function name

### Normal Case

The specified function is no longer marked as being resident and its memory area can be used by another job.

The following message is displayed:

```
VM06 cmm function-name CANCELLED
```

If the function was not preloaded, the following message is displayed:

```
VM06 cmm function name NOT PRELOADED
```

Note : cancelling a function being used for telecommunications will lead to an increase in response time.

## CANCEL OUTPUT COMMAND

This command is used to cancel the request that an output or outputs be printed or punched. A single output can be selected or all the outputs of a job or list of jobs. In the last two cases it can be specified that only outputs in certain classes will be cancelled.

```
CO      { ron : oname
        { ron : osequence
        { ron list      } oclass list } } [STRONG]
                          { ALL
```

### Normal case

The output (or outputs) having the defined characteristics is (are) cancelled. If all the outputs of a job are cancelled, the message:

```
hh.mm ron OUTPUT COMPLETED jobid userid class
```

is displayed.

The indicated output or outputs are cancelled. If the outputs are on the SYS.SYSOUT file they will be erased; permanent outputs are not erased, only the notification to Output Writer is cancelled.

The ALL parameter is compulsory when cancelling all the outputs of a job or list of jobs. The JOB-REP (job occurrence report; output sequence number of 1) must be cancelled by name only e.g.

```
CO ron: JOB-REP [STRONG]
```

If the output is being printed or punched (i.e. in the OUT state), the STRONG parameter must be used.

After a CO command a message is displayed showing the number of outputs cancelled.

```
OU10 CO n OUTPUTS CANCELLED
```

Note : outputs can only be cancelled if they are in the WAIT, HOLD or OUT states.

## CANCEL REQUEST COMMAND

This command is used to indicate that:

- . A requested volume cannot be mounted
- . A requested volume will not be mounted
- . A device will not be switched to write permit
- . There is no recovery to be made following an incident
- . The use of the output writer recovery mechanism is requested after an incident on a printer or card punch.

CR ddnn

### Normal Case

The operator decision is indicated to the job responsible for the request. As a general rule the job will abort but refer to the sections corresponding to the device indicated in the request message for further details.

# DISPLAY BACKING STORE COMMAND

This command can be used to ascertain the current use of backing store.

DBS { SM  
ALL  
SYS  
ron  
CKPT  
VMF  
MAP }

## Normal case

SYS (system) the answer is a series of four messages indicating the number of K-bytes used in the SYS.BKST2, SYS.BKST1 and the SYS.BKST0 files by the permanent Virtual Memory files, preinitialized load modules and the system process group.

```
VM04 DBS BS2 BS1 VMF BS0
VM04 DBS VMF nnK
VM04 DBS PLM nnK
VM04 DBS SYS nnK nnK
```

ron (run occurrence number) - the answer is a series of 2 messages indicating the number of K-bytes the job is using in the SYS.BKST1 and temporary Virtual Memory files. Note that the temporary VMF is part of SYS.BKST1.

```
VM04 DBS BS2 BS1 VMF BS0
VM04 DBS ron nnK nnK
```

ALL - the answer is the same as SYS and ron (for all executing jobs), the SYS.BKST2 space used for loaded Sharable Modules is also shown.

MAP - the answer indicates the SYS.BKST0, SYS.BKST1 and SYS.BKST2 file sizes added together and the amount currently used. The start address and the size of each unit of allocation is also shown.

```
VM04 DBS : nnK/USED nnk (CYL nn, UNIT nnK)
```

SM, CKPT and VMF - the answer is a series of messages, one for each loaded Sharable Module, memorized checkpoint or Virtual Memory file respectively. The size and name of the element is shown, for VMF the name is given in characters and hexadecimal.

```
VM04 BDS nnK : name
```

## DISPLAY DEVICE COMMAND

This command ascertains for a particular device or for all the devices of a certain type:

- The state of the device
- The volume mounted on this device
- The jobs to which the device is allocated
- The requests related to this device which are still pending

DD            { devclass }  
              { ddnn        }

### Normal case

For the device indicated (or for each of the devices of the type indicated) one message is displayed.

This message depends on the state of the device and can be:

- The device has been taken out of the System configuration (using Hold Device command):

DV26    ddnn   HELD

- No volume is mounted; no request pending:

DV26    ddnn   STANDBY

- A volume is mounted but the device is not allocated:

hh.mm   ddnn   PREMOUNTED vol-name   [type]

- A volume is mounted but a Name Volume command must be issued:

DV22    ddnn   NON STANDARD VOLUME

- A mounting request is pending:

\* hh.mm   ddnn   MOUNT volname   [type]   FOR    ron [mode]

\* hh.mm   PRnn   MOUNT FOR ron A BELT : belt name   [,belt name   ...]

- The device is allocated:

DV26 ddnn ALLOCATED TO ron, volname type

or

DV26 ddnn ALLOCATED TO ron

- The device is allocated but a Recover request (including Write Protect) is pending:

\* DV12 ddnn RECOVER volname FOR ron { DELETE LAST CARD  
READ AGAIN LAST CARD  
READ AGAIN LAST RECORD  
LAST RECORD MISPUNCHED }

- The device is allocated in device mode:

DV23 ddnn ALLOCATED IN DEVICE MODE TO ron

A step may reserve a pool of devices; these devices can only be used by that step even though they are in "STANDBY" (see second case). In this case a Display Device for the device class will ascertain the number of free devices. The following message is displayed:

DV26 DD n devclass FREE



## DISPLAY DEVICE TRACE COMMAND

This command displays:

- . The current number of events recorded on a device, all the devices of a certain class, a certain volume or a certain volume on a certain device.
- . The associated threshold values
- . The number of logged events

```
DDT      { ddnn [vol-name] }  
         { devclass   }  
         { volname    }
```

For the indicated device, all the devices of a class, a message or messages is displayed indicating the number of I/O operations and various logged events (and the corresponding percentage) together with the current warning threshold (see Modify Device Trace command).

```
hh.mm DDT ddnn      I/O#: nnn  ABN: nnn (%)  
                        THRESHOLD: nnn  WARN: nnn (%)  
                        ATN: nnn
```

If vol-name was specified as well, or on its own, a message is displayed giving similar information for the volume:

```
hh.mm DDT vol-name ON ddnn      I/O#: nnn  ABN: nnn (%)  
                                    WARN: nnn (%)
```

The MLOG command as well as the PRLOG utility (see Error Logging in Section IV) can be used to reset the counts of logged incidents and input/output operations on a device to zero.

## DISPLAY DISPATCHING PRIORITY COMMAND

This command is used to ascertain the controlled DPRs with their maximum and actual CPU time usage, or just with DPRs are controlled.

DDPR [ALL]

Normal case

No parameters - For each controlled DPR a message is displayed:

TK05 DDPR dpr1 [,dpr2] : n1/n2

Meaning: n1 is the authorized percentage of CPU time and n2 is the percentage currently being used. If a second DPR is shown it means that the factors apply to the range of DPRs.

ALL - Three messages are displayed:

TK06 DDPR SLICED dpr1 [,dpr2] [dprn [, dprn] .....  
TK06 DDPR LIMITED dpr1 [,dpr2] :n .....  
TK06 DDPR OPTIMIZED dpr1 [dpr2] .....

Meaning : for sliced the DPR or range of sliced DPRs is shown.

for limited the DPR or range of limited DPRs is shown with the limiting percentage.

for optimized the DPR with the lowest actual priority is shown.

For all of these more than one "set" may be shown.

\* If there are no controlled DPRs, no message is displayed.

## DISPLAY LOAD MODULE COMMAND

This command ascertains

- If a load module is preinitialized and the space it uses
- All the preinitialized load modules

DLM [lm name: [libname]]

Normal case

- . If a load module name is specified
  - The load module is not preinitialized. The message:  
LD03 DLM LOAD MODULE NOT PREINITIALIZED  
is displayed.
  - The load module is preinitialized:  
LD03 load module name/library name = n UNITS  
n-number of units of backing store used.
- . If the load module is not specified (or in case of an error on the command syntax)
  - No load modules are preinitialized. The message:  
LD03 DLM NO PREINITIALIZED LOAD MODULES  
is displayed.
  - Some load modules are preinitialized. The message:  
LD03 load module name/library name = n UNITS  
is displayed for each of the preinitialized load modules followed by the message:  
LD03 BSMAX = n UNITS BUSED = n UNITS  
BSMAX - the maximum number of usable units in backing store,  
BSUSED - the number of units currently used  
A last message:  
LD03 n PREINITIALIZED LOAD MODULES  
is displayed giving the number of preinitialized load modules.

## DISPLAY LOG COMMAND

This command is used to display information about the SYS.ERLOG system file and information stored on the file. The information can be selected according to the type of events which are being analysed.

```
DLOG { STATE
      CHEXC
      SYSDOWN
      SYSUP
      ITR
      OPR [string]
      SYSERR
      SYSEXC
      MISC
      ddnn [ATN] [WARN] [ABN] } [n][date]
```

### Normal cases

STATE - this is the normal use of this command, the following message is displayed:

```
EL01 SYS.ERLOG FILE n% FULL
```

This message shows the percentage of space that has been used, if the file is full the following message is displayed as well:

```
* EL01 SYS.ERLOG FILE FULL
```

```
DATA LOST FROM date1/time1 TO date2/time2
```

This message indicates what data on the file has been overwritten.

For the other cases one of several DLOG messages are displayed with the contents of the SYS.ERLOG records of the indicated type.

```
CHEXC - channel exceptions (device controller in stop)
SYSDOWN - GCOS terminations and crashes
SYSUP - GCOS loadings
ITR - Integrity Test Routines
OPR - Operator (stored by WLOG command)
SYSERR - system errors
SYSEXC - system exceptions
MISC - miscellaneous, not any of the other cases
ddnn - I/O events on this device, in this case attention
      (ATN) warnings (WARN) or abnormal terminations (ABN)
      can be selected as desired
```

The number of records displayed can be specified by "n", if none are specified 1 is assumed.

The date and time that the search is started from can be specified in the form mm.dd.yy/hh.mm, if it is not specified "now" is assumed.

## DISPLAY MAIN MEMORY COMMAND

This command ascertains the total size of Main Memory and the way it is being used.

DMM [SYS]

Normal case

If no parameter is used the following message is displayed:

```
RS01 MAIN=nK RESERVED=nK LOCKED=nK
```

where: MAIN - shows the size of main memory, in K(1024) bytes.

RESERVED - shows, in K bytes, the amount of main memory that currently executing steps have "declared" they are going to use, (this is specified in the \$SIZE JCL statements), and the amount of memory used by preloaded functions as a result of the PMM command.

LOCKED - shows, in K bytes, the amount of main memory that has been "locked" for certain functions and cannot be used by steps. Locked memory includes I/O buffers, control structures, memory images for Program Mode and system functions made resident with the PMM command.

If the SYS parameter is used a list of all the system functions made resident by use of the Preload Main Memory (PMM) command is displayed:

```
VM07 DMM PRELOADED FUNCTIONS : function name, function name...
```

If there are no preloaded functions the following message is displayed:

```
VM07 DMM NO PRELOADED FUNCTIONS
```

## DISPLAY OUTPUT COMMAND

This command can be used to ascertain:

- the number of outputs in various states.
- a list of outputs, their classes and their states for specified jobs.
- the number of outputs in each state for each job.
- all details for a specific output.
- the output classes started and the current output on a device.
- the amount of free space in the SYS.OUT file.

```
DO [ { ALL
      ron list [output state1] [output class list]
      [output state2] [output class list]
      ron:output name
      ron:sequence number
      ddnn
      SPACE } ]
```

```
output state1 = { KNOWN
                  READY
                  WAIT
                  HOLD
                  OUT
                }
output state2 = { QUEUE
                  WAIT
                  HOLD
                  OUT
                }
```

Various forms of the command follow with the associated response.

### 1 - DO

```
OU21 DO HOLD = n WAIT = n OUT = n
```

The number of outputs in the HOLD, WAIT and OUT states is shown.

### 2 - DO ALL

```
OU24 DO ron RDY = n HOLD = n WAIT = n OUT = n SIZE = n
```

This is displayed for each job with at least one output, SIZE is the total number of lines or cards to be printed or punched.

### 3 - DO ron list [output state] [output class list]

```
OU20 DO ron:sequence-number output-class output-state ron:seq...
```

The outputs for each job indicated is shown in sequence number

order for each job giving their class and state. Only the outputs in the specified state and of a class in the list are shown; if no state is specified all states are shown, the same is true for class.

If there is no output in the specified category the following message is displayed:

OU01 DO NO OUTPUT

4 - DO output state2 output class list

OU02 DO ron:sequence etc..... (as above)

All outputs in the indicated state and of a class in the list are shown along with their state and class. The outputs are shown in the same order as they are in the queue. If no state is specified QUEUE is assumed as default, this means that all outputs in the WAIT, HOLD and OUT states will be shown if their class is one of those shown in the class list. If no class list is specified then all outputs in the state or states specified will be shown.

If there is no output in the specified category then message OU01 is displayed, as above.

5 - DO ron:output name  
or  
DO ron:sequence number

OU08 DO ron:seq-number output-name output-priority output-class

{PR}  
{CD} output state [SIZE = n][x number of copies] [efn [:member]]

For the specified output the output sequence number, name (if any), priority, class, device required (PR = printer, CD = card punch), and its output state is shown. Also, if known, the size of the output, in number of lines or cards, is shown; this is only for outputs created by this run (i.e., not a WRITER JCL statement). If the number of copies is greater than one this is shown also.

In the case of a permanent output the external file name is given; the member name is also given if it is part of a library.

6 - DO ddnn

OU22 ddnn STARTED ON CLASSES :output class list

WORKING ON ron:sequence number output class

The started classes and the current output, with its class, is shown for the specified device. If no classes are started then message OU11 is displayed:

OU11 DO NO CLASS STARTED ON ddnn

7 - DO SPACE

OU03 DO n% OF SYS.OUT SPACE FREE

The percentage of available SYS.OUT space is shown.



## DISPLAY OUTPUT CLASS COMMAND

This command is used to ascertain the default priority associated to a list of output classes and the devices on which these classes are started.

DOC [oclass list]

### Normal cases

For each class in the list message:

OU18 DOC DEFAULT PRIORITY = n FOR CLASS oclass list

is displayed and is followed by a message for each device on which at least one of the output classes is started:

OU23 DOC ddn STARTED ON CLASSES oclass list

## DISPLAY SCHEDULING COMMAND

This command allows the operator to get information on the job executing from various points of view :

- . Detailed information on a job (ron,jobid,userid, status,priorities...)
- . List of jobs currently in a certain state
- . Information related to a class (started or not, load, multiprogramming level...)
- . Number of jobs in all states and system multiprogramming level.
- . Number of executing jobs of all classes and corresponding multiprogramming levels.

DS	[	LOAD	]	Job state being any of : IN,HOLD, SCH,EX,SUSP,OUT,IDLE or ALL
		jclasslist		
		jclass		
		ron		
		jobid        (userid)		
		job-state (submitter)		

Normal cases

The answer depends on the form used:

### 1. NO PARAMETER

```
SH21 DS IN=n HOLD=n SCH=n EX=n SUSP=n  
      OUT=n IDLE = n
```

This is displayed for each job which is known to GCOS.

### 2. CLASS LIST

For each class mentioned the following message is displayed :

```
SH22 DS class [NOT]. STARTED LOAD = n/n SPR = n DPR = N
```

indicating whether the class is started or not, the current number of executing jobs of that class, the maximum class load, and the default scheduling and dispatching priorities associated with that class.

### 3. RON , JOBID

The message given depends on the current job status

- Neither executing nor suspended

```
SH14  ron      { IN      }  jobid userid class SPR = n
              { HOLD    }
              { SCH     }
              { OUT     }
```

It must be noted that the class is the job class, not an output class even if the job is in OUTPUT.

- Executing or Suspended

```
SH15 ron.step {EX  }  jobid userid class SPR= n DPR = n
              {SUSP}
              CPU = n.n ELAPSED = n.n
```

CPU = shows the amount of CPU time the job has used, ELAPSED = shows the time the job has been executing; the time is shown in minutes and thousandths of minute.

This message may be followed by one of :

```
SH19 ron.step REPLY nn          : waiting for a reply
                                to question nn

SH19 ron.step RECOVER ON ddnn    : waiting for a
                                recovery on device
                                nn

SH19 ron.step WAITS FOR VOLUME MOUNTING: waiting for a volume
                                to be mounted

SH19 ron.step WAITS FOR FILE efn : the file is being
                                ON volname used by another job,
                                and the sharing is
                                not possible

SH19 ron.step WAITS FOR PROCESSING : a suspended job has
                                been released but
                                cannot execute
                                because the maximum
                                system load is
                                reached, or a
                                Program Mode step is
                                to be executed but
                                another Program Mode
                                step is executing or
                                suspended.
```

SH19 ron.step WAITS FOR DEVICE dd	: all devices of type dd are being used by other jobs
SH19 ron.step WAITS FOR MEMORY nnK	: not enough memory space is currently available
SH19 ron.step WAITS FOR LOADING	: not enough space is currently available in the SYS.BKST file
SH19 ron.step WAITS FOR BEFORE JOURNAL	: the permissible maximum number of jobs using the Before Journal, at one time, has been reached.
SH19 non.step WAITS FOR SYSTEM RESOURCE:	not enough central system resources available.
SH19 ron.step WAITS FOR SYS.OUT SPACE	: overflow on SYS.OUT system file.
SH19 ron.step WAITS FOR SYS.IN SPACE	: overflow on SYS.IN system file.

If a second message is indicated, in the first three cases an operator action is mandatory, in the other cases (WAITS FOR...) the resource will normally become available when another job (or a step in that job) terminates (see STEP EXECUTION , in section IV).

#### 4. LOAD

SH16 LOAD STARTED = n/n USER = n+n/n SERVICE = n

SH16 LOAD jclass = n/n [jclass = n/n]...

The first line indicates the number of executing and suspended jobs (service jobs included) over the maximum number of "J", the number of executing plus suspended user jobs (batch) over the current multiprogramming limit and also the number of executing service jobs.

The following lines give, for each started class in which at least one job is executing, the number of executing jobs over the class multiprogramming level.

5. IN, HOLD, SCH, IDLE or OUT

```
SH17      ron      { IN
                    HOLD
                    SCH
                    IDLE
                    OUT } jobid  userid  class
```

This message is displayed for each job in the specified state.. indicating the job identifier, the user identifier and the job class.

6. EX or SUSP

```
SH17      ron.step { EX
                    SUSP } jobid  userid  class
```

This message is displayed for each executing or suspended job (depending upon which was specified) and shows the ron, current step, job identifier, user identifier and the job class. This first line may be followed by a message SH19 (see 3 above) for each job which is waiting for a resource.

7. ALL

Same as IN + SCH + HOLD + EX + SUSP + OUT: this should be used sparingly as it uses a lot of CPU time.

Remark: (for cases 5, 6, 7,)

If there is no job in a 'State' the message will be:

```
SH17 DS NO JOBS
```

Note : The System is not taken into account by the Display Scheduling command, but service jobs are.

For 5, 6 and 7 a submitter can be added so that only those jobs that qualify are shown. The submitters are:

- 1) MAIN - Operator introduced jobs only (i.e via SJ or Input Reader).
- 2) Userid - Jobs introduced by an IOF user with this identifier, or spawned by other jobs (via the \$RUN JCL statement) which have this user identifier.

## DISPLAY TIME COMMAND

This command gives the date and time through typing:

DTM

and the system replies with the message:

hh.mm DATE = mm.dd.yy

The date and time are memorized from one GCOS session to the next, providing that the CPU is not powered down or the INITIALIZE push button used.

## END SESSION COMMAND

This command is used to reach a system shut down

END

Normal case

No more jobs will be scheduled.

If some jobs are suspended the message:

```
* SH02 SUSPENDED JOB,END NOT ALLOWED
```

is displayed. All suspended jobs must be released before the shutdown can be completed.

- . All step execution is completed and job execution is stopped (execution will be resumed at the next step, provided there is a Warm Restart). Note: if a checkpoint is reached before the end of a step, execution will terminate here;
- . When all jobs have been thus treated the message:

```
hh.mm GCOS : NO MORE JOBS RUNNING
```

is displayed.

The Output Writer is terminated at the end of its current output. When the shutdown is completed, the message:

```
* AV02 GCOS : END OF SESSION [NO DUMP]
```

is displayed. (The indicator TRAFFIC on the System Operator Panel is off)

Note : As regards the Stream Reader activity (see section IV) the END SESSION command does not force the termination of communication steps. In certain cases (Program Mode, communications...) appropriate action should be performed to terminate an executing step.

If mounting or recovery requests are pending they must be treated in the normal manner.

## FORCE JOB COMMAND

This command allows the operator to :

- Position a job at the front of the queue for job scheduling.
- Ignore any restriction due to the job class, or to the fact that the job was in the Hold state.

FJ ron

### Normal case

The indicated job will be started as soon as it can be, independently of whether the class is started or not, whether the class or system thus becomes over-loaded, or whether the job was in the Hold state. The job even starts before the JCL translator. (but if the job is introduced it must be Translated before being started).

Note : . A Forced Job can be held before it is started by using the Hold Job command, in which case it is no longer considered as Forced.

- . The only limitation that will prevent a forced job from starting is if it would cause the maximum number of "J" to be exceeded. J is the maximum number of executing and suspended jobs (including service jobs) permissible, this number is set at system configuration (by default J = 15).
- . Only one forced job may be present in the In Scheduling state at any one time. If there is a forced job waiting to start and another FJ command is issued on another job, the first job returns to its original position in the scheduling queue and is no longer considered forced.
- . The FJ command must not be used on service jobs; if it is the following message is displayed:

SH13 FJ NOT ALLOWED

- . If the specified job is executing or in the Out state the following message is displayed:

SH11 FJ MEANINGLESS

- . If the specified job does not exist the following message is displayed:

SH09 FJ JOB UNKNOWN



## FORCE OUTPUT COMMAND

With this command an output in the READY, WAIT or HOLD state can be released and positioned as the first output to be selected for treatment.

F0 {ron:sequence}  
    {ron:aname } [STRONG]

### Normal case

The indicated output is forced and will be started as soon as a device of the correct type, on which the Output Writer has been started for the output class of this output, becomes free.

For outputs in the READY state the STRONG option must be used.

A message is displayed

```
0017 F0 ron:osequence FORCED
```

Only one output may be forced at one time, if a F0 command is issued before a previous forced output has started the previous output is no longer forced and is put into the WAIT state.

A forced output may be held (put into the HOLD state) by use of the Hold Output (HO) command.

## HOLD DEVICE COMMAND

This command takes a device out of the system configuration (providing it is not presently allocated).

HD) ddnn

Normal case

The indicated device is taken out of the system configuration and all action on it will be ignored by the system until the device is released by using the Release Device command.

The message:

hh.mm ddnn HELD

is displayed.

Note : the effects of this command are not kept from session to session, thus any device that has to be held should be held again after GCOS has been loaded.

## HOLD JOB COMMAND

This command allows the operator

- To put in the Hold state jobs which are Introduced or In Scheduling
- To put in the Suspended state jobs which are Executing

HJ [ronlist [ENDSTEP]]

### Normal case

The indicated jobs are put in the Hold state (if it is Introduced or In Scheduling) or in the Suspended state (if it is Executing)

If ENDSTEP is used the Executing jobs will be suspended at the end of their current step. This ensures that memory and devices are freed.

If no parameter is indicated all jobs Introduced or In Scheduling are put in the Hold state.

There is no message.

- Note:
- . Jobs in the Introduced state are translated before they are held.
  - . This command does not apply to the JCL Translator or to BTNS.
  - . If this command is used on a Forced job, it is considered as being held and no longer forced.
  - . Jobs in the Out state cannot be Suspended.
  - . Suspended jobs are still considered as being in the system or class load.

## MODIFY DEVICE TRACE COMMAND

### HOLD OUTPUT

This command modifies the number of events on a device warning message will be issued. This can be done for a device class, or for a particular device.

This command modifies the number of events on a device warning message will be issued. This can be done for a device class, or for a particular device.

MDT { devclass } [nnn]  
      { ddnn }

HO { ron :  
      ron :  
      ronli

### Normal case

Normal case

The value is taken as the new threshold. Should the number detected reach this threshold a warning message (AV16) displayed.

The output defined ch

Value 0 indicates that message AV16 should never be displayed.

The output that they in they will in the JCL

If no devices or device classes are indicated, all devices assumed.

The STRONG being printed and put in informed here released.

If no parameters are used message AV16 will not be displayed until the value is reset.

After a HOLD outputs here

Note: The threshold value is left unaltered by System Initialization, as well as the number of detected devices. This last number can be reset to 0 by using the MDT command (see Error Logging, Section V) and the Modify Device Trace command.

OU10 HO

To release

## HOLD JOB COMMAND

This command allows the operator

- To put in the Hold state jobs which are Introduced or In Scheduling
- To put in the Suspended state jobs which are Executing

HJ [ronlist [ENDSTEP]]

### Normal case

The indicated jobs are put in the Hold state (if it is Introduced or In Scheduling) or in the Suspended state (if it is Executing)

If ENDSTEP is used the Executing jobs will be suspended at the end of their current step. This ensures that memory and devices are freed.

If no parameter is indicated all jobs Introduced or In Scheduling are put in the Hold state.

There is no message.

- Note:
- . Jobs in the Introduced state are translated before they are held.
  - . This command does not apply to the JCL Translator or to BTNS.
  - . If this command is used on a Forced job, it is considered as being held and no longer forced.
  - . Jobs in the Out state cannot be Suspended.
  - . Suspended jobs are still considered as being in the system or class load.

## HOLD OUTPUT COMMAND

This command is used to put into the Hold state an output, all the outputs, all the outputs of a list of jobs, or in the last two cases just those outputs in the indicated output class or classes.

```
H0 { ron : oname  
   { ron : osequence  
     ronlist } [oclass list] } [STRONG]
```

### Normal case

The output (or outputs) currently in the WAIT state having the defined characteristics is (are) put in HOLD.

The output (or outputs) in the Known or Ready state are marked so that they will enter the Hold state as soon as the "when condition", in they will enter the Hold state as soon as the "when condition", in the JCL, is met.

The STRONG parameter is used if an output in the Out state (i.e. being printed or punched) is to be held. The output is interrupted and put in the Hold state, GCOS asks question OU02 so that it can be informed how treatment should be resumed when the output is released.

After a H0 command a message is displayed indicating the number of outputs held or marked to be held.

```
OU10 H0 n OUTPUTS HELD
```

To release the output, use the Release Output command.

## MODIFY CLASS COMMAND

This command allows the operator to modify for a job class (or several classes) the associated values of :

- . The class multiprogramming level
- . The default scheduling priority
- . The default dispatching priority

MC [class list] [ $\left\{ \begin{array}{c} \text{class-multi} \\ * \end{array} \right\}$ ] [ $\left\{ \begin{array}{c} \text{spr} \\ * \end{array} \right\}$ ] [dpr]]

### Normal case

The given values are associated with the indicated classes. The values replaced by an \* or not mentioned are left unchanged. For example MC G \* \* 3 will set the default value for dispatching priority of class G to 3, leaving the multiprogramming level and the scheduling priority unchanged.

### Note :

1. These values are left unmodified from one session to the next
2. Since the scheduling priority of a job is determined when the job is introduced, modification of the default value for a class will have no effect on jobs already known by GCOS. The new value will be taken into account for jobs which will be introduced in the future
3. Since the dispatching priority of a step is determined when the step execution begins, modification of the default value for a class will have no effect on steps already executing. It will be taken into account for steps which will be started in the future (including the steps of jobs already executing)
4. The Modify Job command is used to change the dispatching priority of an executing step

## MODIFY DEVICE TRACE COMMAND

This command modifies the number of events on a device after which a warning message will be issued. This can be done for each device of a device class, or for a particular device.

MDT  $\left[ \begin{array}{l} \{ \text{devclass} \} \\ \{ \text{ddnn} \} \end{array} \right] \text{ [nnn]}$

### Normal case

The value is taken as the new threshold. Should the number of events detected reach this threshold a warning message (AV16) will be displayed.

Value 0 indicates that message AV16 should never be displayed

If no devices or device classes are indicated, all devices is assumed.

If no parameters are used message AV16 will not be displayed for any device until the value is reset.

Note: The threshold value is left unaltered by System Initialization, as well as the number of detected events. This last number can be reset to 0 by using the PRL0G utility (see Error Logging, Section V) and the Modify Log (MLOG) command.



## MODIFY DISPATCHING PRIORITY COMMAND

This command is used to indicate which Dispatching Priorities (DPR) should be sliced, limited or optimized.

MDPR [dpr[:dpr]] [ { SLICE }  
                  [ { n  
                  { OPT } } ]

### Normal case:

- SLICE - the indicated DPR or all from the first to the second indicated DPRs are sliced. This means that executing steps using the same DPR have an equal share of the CPU time for that CPU
- n - The indicated DPR or range of DPRs are limited, they cannot use more than "n"% of the CPU time.
- OPT - Only one priority is indicated. For this priority and the adjacent one (DPR+1) dynamic optimization takes place. This optimization ensures that I/O-bound steps have an effectively higher priority than CPU-bound steps for steps executing in these priorities.

If no second parameter is indicated the DPR or DPR range is set back to normal.

The control of DPRs becomes effective immediately, even on executing steps.

The effects of this command are kept from session to session.

If a DPR has previously been modified by use of the MDPR command it cannot be re-modified until it has been set back to normal. Every time a MDPR command is used successfully a message is displayed:

TD04 MDPR SUCCESSFUL

## MODIFY JOB COMMAND

This command allows the modification of a class or priorities of a job or several jobs.

```
MJ ron list [jclass] [ {spr} [dpr]] [switch list [OFF]]
                    { * }
```

### Normal case

The class and/or priorities and/or job switches are modified as indicated. An \* (asterisk) instead of a number will leave the scheduling priority unmodified.

The job switches can be indicated only for a job which is executing or suspended. Class can only be indicated for jobs before they begin executing.

### If the class is modified:

- The default value for Scheduling priority of the new class will have no effect on the jobs.
- The default value for Dispatching priority will be taken into account only for steps which will be started in the future. The job class can only be modified before the job starts executing.

### If the scheduling priority is modified:

- If the jobs are not yet executing, it will be taken into account immediately, leading possibly to repositioning in the queue
- If the jobs are executing, it will be taken into account for any future resource enqueueing, but will not lead to repositioning of jobs already waiting for resources.

### If the dispatching priority is modified:

- The dispatching priority of the executing steps are immediately modified (leading to a possible speed up or slow down of their execution)
- It will be used as a default value for any subsequent steps (unless a priority is specifically given in \$STEP statements by the user) instead of the class associated default value.

### If the switches are set:

- They are used as a communication between the operator and the executing job
- The switches are of the form SWnn and are in the range SW0 up

to SW31. They can be tested by JCL or by a user program.

**Note :** This command should not be used without the agreement of Operations Manager because of its possible effects on response time and system throughput. This command can be used to modify the characteristics of WRITER, IOF and ROF service jobs. The effects will last only for this execution of the job.

## MESSAGE OF TODAY COMMAND

This command allows the operator to:

- . Send a message to be printed in the Job Occurrence Report of all jobs and displayed to each IOF user when they log on.
- . Ascertain the current value of this message

MOT { RESET }  
      { text }

### Normal case

If a text is indicated it will appear in the Job Occurrence Report of all jobs which will be started after the command has been issued and it will be displayed to each new IOF user as they log-on.

If no text is indicated the currently active text is displayed:

MOT : text

- . RESET indicates that the text is to be erased.

### Note :

- 1 The text can be up to 176 characters long. If necessary the command should be continued on more than one line.
- 2 The text remains from one GCOS session to the next.

## MODIFY LOG COMMAND

This command is to be used only under Field Engineer supervision. It is used to modify the management of the SYS.ERLOG file; for more information refer to the Error Logging manual.

MLOG { INIT  
RESTART  
TERM  
{ RESET }  
TRACE }  
{ NTRACE }  
WARN } [ { ddnn } ]  
NWARN }

## MODIFY OUTPUT COMMAND

This command is used to modify the output class or the output priority or both for one output, all the outputs of a job or series of jobs; in the last two cases it can be specified that only outputs in a certain class or classes are modified. Also the destination of an IOF or ROF output can be altered.

```
MO {ron:osequence } [oclass-list oclass] [opriority] [ostation]
   {ron:oname }
   {ron-list }
```

### Normal case

The indicated outputs are put in the new output class or given a new output priority or both. If the output priority is modified and the output is in the output queue, its position is modified accordingly.

For ROF and IOF outputs, the terminal or terminal station to which the output is to be sent, can be changed.

After the use of an MO command a message is displayed showing the number of outputs that were modified.

```
OWIO MO nOUTPUTS MODIFIED
```

## MODIFY OUTPUT CLASS COMMAND

This command is used to modify the default output priority associated with an output class.

```
MOC   oclass   opriority
```

### Normal case

The default priority of the indicated class is set to the specified value. This will have no effect on outputs already known to GCOS. A message is displayed showing the achieved change.

```
OU18 MOC DEFAULT PRIORITY = n FOR oclass
```

## MODIFY SCHEDULING COMMAND

This command allows the operator to set the multiprogramming level to any value between 0 and the maximum value set at system configuration, and to enable or disable the tracing of every step of every job.

MS [multiprogramlevel] [ {TRACE }  
 {NTRACE} ]

### Normal case

The multiprogramming level is set to the value indicated. No message is displayed. If the multiprogramming level is set to 0 no new job can be started, but jobs already executing will not be affected.

If TRACE is used message JB08 is displayed for each step that begins its execution. This message shows the ron and step, load module name and Dispatching Priority.

JB08 ron.step STEP load module name DPR = n

Note : the result of the MS command is kept from one session to the next.





## PREINITIALIZE LOAD MODULE COMMAND

This command allows the operator to take a load module from a load module library on disk and put it into Backing Store (SYS.BKST2 file). The load module is then said to be preinitialized. This saves time whenever the load module is loaded for execution.

```
PLM  lm-name [:libname [:volname : devclass]]
```

devclass = MS/M300,MS/M350,MS/M400,MS/402 or MS/452

The default library-name is SYS.HLMLIB. If the volume and device class are not specified it is assumed that the library is a catalogued file or that it can be located on a resident disk.

### Normal case

The load module is preinitialized. The message:

```
LD01  PLM SUCCESSFUL
```

is displayed.

## PRELOAD MAIN MEMORY COMMAND

This command is used to make certain system functions resident in main memory. This saves time and increases throughput; the main use for this command is with telecommunications where if certain functions are not resident response time becomes unsatisfactory.

PMM function-name

Normal case

The function is loaded into main memory. This function will remain resident until cancelled with the Cancel Main Memory (CMM) command or the session is terminated.

If the preload is successful the following message is displayed:

```
VM05 PMM function name PRELOAD
```

If the function is already preloaded the following message is displayed:

```
VM05 PMM function name ALREADY PRELOADED
```

If there is not enough memory available the following message is displayed, the operator should re-issue the command when more memory is available:

```
VM05 PMM TRY LATER : NOT ENOUGH MEMORY
```

Note : This command should only be used with the operations manager's approval.

For various functions see System Functions in section III.

## RELEASE DEVICE COMMAND

This command puts a previously held device back into the configuration.

RD ddnn

Normal case

The indicated device is put back into the system configuration. The message :

AV03 ddnn RELEASED

is displayed

If it is a disk or tape drive it is treated (if in Ready) as if it had just been premounted. Unit record devices are considered as being free.

Note : Following a Release Device command on a printer it may happen that GCOS prints the message:

DV17 PRnn MEDIA RECOGNITION IMPOSSIBLE.

In this case press the STOP and START pushbuttons on the indicated printer.

MTL { line { in-timeout } { out-timeout } { I/O-ratio } } [speed] { TRACE }  
 { \* \* \* } [NTRACE]

MTP : modify telecom poll modify the polling list for a line from the "start" of the list indicated in the command.

MTP line { 1 } station-index-1/.../station-index-n  
 { start }

MTT : modify telecom terminal modifies the characteristics of a terminal.

MTT { terminal-queue1 { terminal-queue-2 } [ALTER]  
 { NALTER }  
 terminal { program  
 program-queue }  
 { NASGN }

RT : release telecom releases the component declared by the appropriate operand.

RT { line  
 program-queue }  
 { station  
 terminal }

ST : start telecom starts a telecommunications session.

ST

TT : terminate telecom causes the telecommunications session to "shut-down".

TT [STRONG]



## RELEASE JOB COMMAND

This command allows the operator to:

- . Put back into the In Scheduling state a job or all jobs which are in the Hold state.
- . Release suspended jobs so that they can resume execution.

```
RJ    {ronlist}  
      {ALL    }
```

Normal case

ALL :

All jobs in Hold are put back into the In Scheduling state. No message is given.

The indicated jobs are in Hold:

They are put back in the In Scheduling state. No message is given.

The indicated jobs are Suspended:

The jobs are released and will resume execution as soon as resources are available. The message:

```
RS06  ron  REACTIVATED
```

will be displayed for each job as it resumes execution.

## RELEASE OUTPUT COMMAND

This command can be used on one output, all the outputs of a job or list of jobs; or in the last two cases, just those outputs in the classes specified.

This command is used to:

- 1) Release outputs from the Hold state to the Wait state.
- 2) Cancel the request that outputs in the Known or Ready state will be put in the Hold state (from a previous Hold Output command).
- 3) To put outputs currently in the Ready state into the Wait state without waiting for the "when condition" (in the JCL) to be met.

```
RO { ron : oname  
   { ron : osequence  
     { ron list [oclass list] } } } [STRONG]
```

### Normal case

The outputs currently in the Hold state and having the defined characteristics is (are) put in the Wait state.

Outputs in the Known or Ready states are marked so that they will be put into the Wait state as soon as the "when condition", in the JCL, is met.

The STRONG parameter must be used if an output in the Ready state is to be put into the Wait state without waiting for the "when condition".

A message is displayed, after an RO command, showing the number of outputs put into or marked to enter the Wait state:

```
OU10 RO n OUTPUTS RELEASED
```



## START CLASS COMMAND

This command enables jobs of a job class (or of several classes) in the In Scheduling state to be selected by Job Scheduling.

SC        [jclass list]

### Normal case

The class or classes indicated are started.

If no parameter is given all the classes are started.

No message is given.

Note : This command has no effect on the class multiprogramming level.

This command can be used on class Q (IOF service job) and class U (ROF service job) and allows new users to log-on. This command is not valid for other service jobs.

## START DEVICE TRACE COMMAND

This command displays a message each time an event is logged for a device, on all devices of a certain class, or on all devices.

```
SDT [ {ddnn } ] [ATN] [WARN] [ABN]
    [ {devclass} ]
```

ATN - selects attention

WARN - selects normal terminations with marginal conditions

ABN - selects abnormal terminations

### Normal case

Each time an event of the selected categories is logged for a device, any device of the class or all devices (if no parameter was given) an informative message (AVI7) will be displayed.

If nothing is indicated ATN and ABN are used as default

To terminate this option use the Terminate Device Trace (TDT) command.

## START INPUT READER COMMAND

This command activates the Stream Reader to read an input stream from a tape or disk sequential file, or from the member of a library on disk.

```
SI [sfn :]efn [: volname : devclass] [MFT] [ { Job 1
                                                Job 1 :
                                                : Job 2
                                                Job 1 : Job 2 } ]
```

### Normal case

The Stream reader reads the file or the subfile indicated as the input stream. This file or subfile is assumed to be a catalogued file or on a Resident disk unless a volume is indicated with the corresponding type of device.

The whole file or subfile contents are taken as input stream unless a third parameter limits the portion to be read:

job 1            only that job  
job 1 :          all jobs starting from job 1 (included)  
: job 2          all jobs up to job 2 (included)  
job 1 : job 2    all jobs from job 1 to job 2 (both included)

MFT - this indicates that the file is located on a multifile tape so that GCOS looks for the file on the whole tape.

A volume mounting is requested if necessary

The following message is displayed for each job that is started:

```
hh.mm ron { IN
           } job-id user-id class SPR = spr
           { HOLD }
```

## START JOB COMMAND

This command introduces a job whose description has been stored in a library.

```
SJ Jobid [efn[:media:devclass]] [class] [spr] [switch-list]
                                         [parameters]
```

### Normal case

The job whose description is named by jobid in the library named efn is introduced, it can be more than one job (see Note below). The element ":efn" need not be used if the library is the SYS.HSLLIB file and the element " :media:declass" need not be used if the library is a catalogued file or on a resident disk.

The class parameter, if given overrides the class possibly stored in the description; class P is used if no class parameter is given in either the command or the job description.

The scheduling priority parameter overrides the scheduling priority possibly stored in the description. The default value associated with the job class is used if no scheduling priority is given and the stored description does not indicates any.

If Switches are indicated, they are set to one.

The parameters are to be used in connection with the description of the job (see the \$RUN statement in the JCL Reference manual).

The following message is displayed for each job introduced indicating if it is held or not.

```
hh.mm  ron  IN  Jobid  userid  class  SPR = spr
                HOLD
```

Note : Service Jobs cannot be introduced by use of this command.

Since this command starts the Stream Reader the library member may be an input stream (more than one job description) and the IN/HOLD message is displayed for each job. If the actual jobid is different from the library member jobid (in the SJ command) the actual jobid is shown in the IN/HOLD message.

## START OUTPUT WRITER

This command starts the activity of the Output Writer on a specific device or sequential file and for specific output classes (possibly all).

```
SO { ALL
    { ddnn [oclass list]
      { [sfn:]efn[:volname:]devclass [oclass list] [out-devclass] }
```

### Normal case

The printing or punching of outputs of the indicated classes is initiated on the indicated device. (If no class is specified all classes are initiated). If the Output Writer was not already started on the device this will correspond to the introduction of a WRITER service job.

The mounting of a volume on the device will be requested if necessary.

If the Output Writer is started on a sequential file (efn specifies the external filename) a copy of all the indicated output classes for the type of device (CD = card punch, PR = printer) will be stored on the file. Once this is done these outputs will be considered as having been treated. The "attributes" for printers indicates line length and the print speed. These files can later be treated via the JCL & WRITER statements.

If the sequential file is a catalogued file or on a resident disk "[:media:]devclass" need not be specified.

## START TELECOMMUNICATIONS COMMAND

This command introduces the BTNS service job, this is a prerequisite for any communications activity.

ST

Normal case

The BTNS service job is introduced and started, no message is displayed.

## TERMINATE CLASS COMMAND

This command stops jobs of a particular job class or of several classes which are in the In Scheduling state from being selected by Job Scheduling.

TC      [jclass list]

### Normal case

The class or classes indicated are terminated. No jobs of these classes will then be started until a Start Class command is issued (except for Forced Jobs).

If no class is indicated all classes are terminated. There is no message.

This command has no effect on the class multiprogramming level.

This command has no effect on the jobs of a class which are already executing, nor on jobs which have been Suspended then Released and which wait for processing.

Note : This command can be used on class Q (IOF service job) and class U (ROF service job) and thus prevent any more users logging-on. This command is not valid for other service jobs.

## TERMINATE DEVICE TRACE COMMAND

This command terminates the displaying of a message each time an event is logged for a device, on all devices of a certain class, or on all devices.

```
TDT  { ddnn  
      } [ATN] [WARN] [ABN]  
      { devclass }
```

ATN - selects attention  
WARN - selects normal terminations with marginal conditions  
ABN - selects abnormal terminations

### Normal case

The effect of a previous Start Device Trace command will be terminated for the device indicated, or every device of the indicated class, or all devices (if no parameter is given) and the indicated type of events (if none is indicated, they will all be terminated).



## TERMINATE INPUT READER COMMAND

This command stops the reading of any new job descriptions by the Stream Reader.

TI

Normal case

The reading of the job descriptions is terminated just before the next \$JOB record. The message:

```
IN13  TI  JOB READING STOPPED
```

is displayed (unless the command is entered after the last \$JOB record has been reached).

## TERMINATE JOB COMMAND

This command terminates a job which is in the Introduced, Scheduling, Hold, Executing or Suspended state.

TJ ronlist  $\left[ \begin{array}{c} \{ \text{STRONG} \} \\ \{ \text{FORCE} \} \end{array} \right] \left[ \begin{array}{c} \{ \text{DUMP} \} \\ \{ \text{NDUMP} \} \end{array} \right]$

### Normal case

#### STRONG is not indicated

If a step is executing it is aborted. This will cause the job to be aborted unless the step has been declared as being repeatable or the user, by using \$JUMP JCL statements, has taken care of this situation.

#### STRONG is indicated

The executing step (if any) and the job terminates abnormally regardless of any repeatable steps or \$JUMP JCL statements.

#### FORCE is indicated

FORCE is used only after a TJ or TJ STRONG when the executing step has an exit routine for when the step aborts; the FORCE parameter causes the exit routine to be aborted also.

#### NDUMP/DUMP

If a step execution is aborted as a result of the Terminate Job command, a dump of this step is taken if it was requested in the JCL. However if NDUMP is indicated in the command, no dump will be taken regardless of the JCL instructions, conversely DUMP will force a dump to be taken.

A message :

hh.mm ron.step KILLED jobid userid class

is displayed for each job that is aborted by a TJ command.

Note : If the job is in the Introduced state it will be terminated only after its Translation has been completed.

## TERMINATE OUTPUT WRITER COMMAND

This command terminates the Output Writer activity on a device or sequential file for specific output classes (possibly all).

```
TO { ALL  
   { ddnn [oclass list] } [STRONG]  
   { [sfm:]e fn[:media:devclass] [:oclasslist] }
```

### Normal case

The printing or punching-out of outputs of the indicated classes on the indicated device or sequential file is terminated.

- If no class is specified, all classes are terminated
- If STRONG is specified the current output is stopped immediately (question OU02 is asked), else the stop will occur at the end of the current output.
- If ALL is indicated all Output Writer functions are terminated
- If the sequential file is a catalogued file or on a resident disk then the "[:media:devclass]" parameter is not needed.
- If all output classes are stopped on a device, the device is released, and can be used by another job. The corresponding WRITER service job is no longer known to GCOS.
- The Output Writer activities are also terminated as part of the End Session command. This case is equivalent to a Terminate Output Writer ALL without the STRONG option.

## TERMINATE TELECOMMUNICATIONS COMMAND

This command is used to terminate a telecommunications session.

TT [STRONG]

Normal case

If the STRONG option is not used the session will terminate by degrees. No new users may log-on. At the end of their current steps users will be shutdown and will not be able to do any further processing. When all the steps using the BTNS service job have ended the telecommunications session is shutdown and the BTNS service job is terminated.

If the STRONG option is used the telecommunications session is stopped after any current I/O operations have been completed, the BTNS service job is terminated, all the current user steps linked to the BTNS service job will abort.

Note : Before issuing a TT command all users should be informed and given time to finish their work tidily. The users can be informed via the Broadcast Telecom (BT) command, see the list of Telecom Commands in Section VII.

## USE CATALOG COMMAND

This command is used to indicate to GCOS that in following Operator Commands any catalogued files mentioned must be looked for in the catalog specified in the UCG command, before a general search of the resident disks is made.

UCG [efn [:volname:devclass:]]

### Normal case

The indicated catalog will be used the next time a file description is looked for (i.e. when the OCL command does not indicate the volume and device class). If the command is successful the following message is displayed:

```
CG05 UCG SUCCESSFUL
```

If the volume and device are not specified in the UCG command, resident disks are assumed.

The effect of a UCG command is kept from one session to the next.

If just UCG is used with no parameter then the previous command is negated and SITE.CATALOG is assumed again.

Note : this command must not be used when any IOF users are logged-on.

## USE DEVICE COMMAND

This command indicates to the system that a disk drive has been set to Write Permit in response to message DV07.

UD ddnn

### Normal case

The activity that was temporarily blocked because of the disk being in Write Protect is resumed. There is no message.

The Terminate Class (TC) command used on classes Q and/or U will prevent any new log-ons for IOF and/or ROF functions respectively.

The TDS command M NONEW will prevent any new log-ons for the TDS.

The Terminate Telecom (TT) command will prevent any new log-ons for any telecommunications function. It also implies that the BTNS service job will terminate as soon as all terminals are logged-off and the MCS programs or programs have been deassigned from the program queues.

The operator can log-off a specific terminal.

For terminals using IOF or ROF service jobs the operator issues a Terminate Job (TJ) on the corresponding service job, however this does not stop the user from logging-on again; to prevent this the operator will have to modify the corresponding class multiprogramming level.

The TDS command M REMOVE will log-off a specific user at the end of the current transaction, the M CANCEL command will do this at once without waiting for the transaction to finish.

The Hold Telecom (HT) command can be used to log-off a terminal, all of a station's terminals, all terminals using a specific line or all terminals using a specific program queue. This command also prevents any log-ons from the affected part of the network till it is released with a Release Telecom (RT) command. The effects of the Hold Telecom command are kept from session to session.

The programs (jobs) using communications can be terminated.

A MCS program can be terminated by using the Terminate Job (TJ) command. This will not cause the corresponding terminal to be logged-off, but it will not be able to get any response from the program.

A TDS can be terminated using the M STOP TDS-command. This will cause all terminals connected to the TDS to be logged-off at the end of their current transaction, then the TDS step will terminate.

IOF and ROF service jobs have to be terminated using the Terminate Job (TJ) command, which has to be issued on each job separately.

A list of communications operator commands is shown below with a brief description; these commands are entered on the Network Control Terminal (NCT). If the system console is not the NCT then only the BT, ST and TT commands may be entered via the system console.

**BT** : broadcast telecom  
sends a message to destinations declared by the appropriate operands.

BT { program  
program-queue } message-text  
terminal  
ALL }

**DT** : display telecom  
displays the status of the component declared by the appropriate operand.

DT [ { line [STRONG]  
program [STRONG]  
program-queue [STRONG]  
QUEUE  
station  
[terminal] [ { QUEUE }  
[STRONG] } ] }

**HT** : hold telecom  
inhibits the component declared by the appropriate operand.

HT { line  
program-queue  
station  
terminal }

**MTE** : modify terminal edit  
modifies data format sent from or received by the terminal.

MTE terminal [ { BLOCK } [ { line-length } { block-size } ] [ { IMARK } ] [ { INEDT } ] [ { ONEDT } ] ]  
[ { NBLOCK } ] [ \* ] [ \* ] [ { INORM } ] [ { ONORM } ] ]

**MTL** : modify telecom line  
modifies the functionality of a line or makes available maintenance facilities for the line.



## WRITE LOG COMMAND

This command allows the operator to write up to 255 characters of text in the SYS.ERLOG system file.

WLOG text

Normal case

The text is entered into the SYS.ERLOG file.

Note : this facility should only be used with Field Engineer and Site Manager approval.



## SECTION VII

### COMMUNICATIONS

In addition to the local devices in the computer room or site, GCOS provides a set of functions allowing the use and control of remotely situated devices.

These devices are called terminals. Several terminals may be subservient to one device, called a station, which is the only device of the group to be directly connected to the central system.

Terminals or the station are connected to the central system via lines, these lines are named using the same rules as local devices i.e. LNO1, LNO2 etc...; the terminals and stations however are user named and do not have to follow these rules.

The set of all terminals, stations and lines is called the Network. The physical control of the network is carried out by the Basic Telecommunications Network Support (BTNS) service job.

The terminals can be used in four different ways:

MCS - A user program can use a terminal in a similar way to how it uses a local device, to be able to do this it uses MCS (Message Control System). Messages to and from the program go through queues (located on a disk or in memory); the program receives messages via the program queue and sends messages via the terminal queue, and vice-versa for a terminal user.

Each terminal queue corresponds to one terminal and uses the same name as the terminal. The names of the program queue are user defined.

Only MCS programs use queues. they are not used by ROF, IOF or TDS jobs.

An executing MCS program is an executing step of a job.

TDS - it is possible to use a network for TDS (Transaction Driven System) applications. In this case a predefined set of GCOS functions is used with user programs for the application. An executing TDS is an executing step of a job.

ROF - Terminals can be used to introduce streams of jobs, which are executed as if the operator had introduced them; it is also

possible for the outputs to be sent back to the corresponding terminal or station. This type of usage is handled by the ROF (Remote Operations Facility) service job. This facility can be used to transfer files in and out of the central system.

Each terminal or station that is using this facility has an ROF service job executing.

IOF - This last type of usage is IOF (Interactive Operation Facility), it is similar to ROF but has a wider range of functions. IOF enables terminal users to perform source program and JCL editing functions, to introduce jobs, be aware of the execution of these jobs and to examine the output of these jobs.

For each terminal user using IOF facilities there is an IOF service job executing.

There are no practical limitations on the number of different MCS, TDS, ROF or IOF functions being used simultaneously, but an installation would need only a limited set of these functions to fulfil its requirements.

The responsibilities of the system operator in an installation using communications include:

- starting communications activities
- control of communications functions since they compete for resources with user jobs as well as between themselves
- reaction to possible incidents
- terminating communications activities

Note : all dialog with BTNS service jobs take place on the Network Control Terminal; in the rest of this section this is assumed to be the system console.

For each TDS one terminal is used as the TDS Master Terminal; in the rest of this section this is assumed to be the system console. All TDS Master commands have to be entered on the Master Terminal and when the Master Terminal is the System Console preceded by the TDS name (i.e. TDS! M START; the M indicates a Master command).

Only some BTNS commands are shown here. For more information on BTNS and TDS commands refer to the TDS Site manual and the Network Control Terminal Operations manual.

MCS programs use the MAM (Message Access Method) system function and IOF, ROF and TDS all use the VCAM (Virtual Communications Access Method) system function. See "System Functions" in Section III.

## STARTING COMMUNICATIONS ACTIVITIES

All communications activities must be supported by the BTNS service job, thus the first command is always Start Telecommunications (ST) which introduces BTNS (see section VI for all commands).

All communications have two elements in addition to BTNS:

- a program using communications, i.e. a TDS or MCS program, or an IOF or ROF service job.
- a terminal and a terminal user which uses the program. The link between the terminal and the program (or program queue for MCS) is established by a log-on command; this command is issued at the terminal and the necessary processing carried out by BTNS. A message is sent back to the terminal informing the user that the log-on has been performed.

A program using MCS is started like any other job, by a Start Job (SJ) or Start Input Reader (SI) command. No specific message is given when a terminal is logged-on to a program queue unless the terminal is a read only terminal; in this case any message that would have been sent to the terminal is sent to the Network Control Terminal instead. Because Programs and terminals using MCS interact via queues they do not need to be active simultaneously.

TDS is started by introducing a job, a step of which is the TDS. As soon as the TDS step starts a message is displayed:

```
OP04  ron.step  COMMAND : tds name  CREATED
      ron.step  TDS STARTED. YOU ARE MASTER TERMINAL
```

A M START command is then sent to indicate which type of start is required. After the M START command is completed a M ANEW command is issued so that terminals can be logged-on to the TDS. No specific message are given when terminals are logged-on to a TDS.

IOF and ROF do not require any action by the operator to be started (except BTNS must be executing). When a terminal is logged-on for IOF or ROF activities the corresponding service job is introduced and scheduled by GCOS. No specific messages are displayed for IOF or ROF log-ons.

## CONTROL OF COMMUNICATIONS

Communications activities use system resources just like batch jobs, the main difference is that the resources used by communications depend upon the terminal user's activities and are thus difficult to plan beforehand. Another point worth mentioning is that users are on-line and much more sensitive to delays in responses to requests etc., than batch jobs. GCOS provides a set of functions to ascertain the current use of communications and to control it, the commands listed below are the most likely to be used for this.

The Display Telecommunication (DT) command can be used to display information on connected terminals, stations, lines and users.

The TDS command M STATUS provides information on users connected to the specified TDS.

The Display Scheduling (DS) command can be used to ascertain the current usage of ROF and IOF service jobs, either by displaying a specific job by the job identifier IOF or ROF or for all by use of their class.

The Broadcast Telecom (BT) command allows the operator to send messages to one or all the terminals, or to a program or program queue.

The TDS command M SEND is used to send a message to a specific user, while the M BROADCAST command is used to send a message to all users.

The Modify Telecom Terminal, Modify Telecom Editing, Modify Telecom Polling and Modify Telecom Line commands can be used to modify the characteristics of the Telecommunications Network. These four commands are not described in this manual and should be referred to in the Communications Processing Facilities manual.

Several of the commands described in relation with termination of telecommunications can be used to log-off a terminal, prevent any new log-ons etc... . These commands can also be used as a means of controlling telecommunications activities.

## TERMINATING COMMUNICATIONS

Terminating communications can occur at a variety of levels; this is due to the number of activities that could be used simultaneously.

Each terminal user can break the link between the terminal and the central system, this operation is called log-off. No specific message is displayed on the system console when a user logs-off.

It is possible for the operator to prevent any new users from logging-on to a specific function or all functions. Thus communications activity will slowly decrease as more users log-off having completed their work.

HONEYWELL INFORMATION SYSTEMS  
Technical Publications Remarks Form

TITLE

SERIES 60 LEVEL 64  
SYSTEM OPERATIONS OPERATOR GUIDE  
ADDENDUM B

ORDER NO.

AQ13-01B

DATED

DECEMBER 1979

ERRORS IN PUBLICATION

[Empty box for errors in publication]

SUGGESTIONS FOR IMPROVEMENT TO PUBLICATION

[Empty box for suggestions for improvement to publication]



Your comments will be investigated by appropriate technical personnel and action will be taken as required. Receipt of all forms will be acknowledged; however, if you require a detailed reply, check here.

FROM: NAME \_\_\_\_\_

DATE \_\_\_\_\_

TITLE \_\_\_\_\_

COMPANY \_\_\_\_\_

ADDRESS \_\_\_\_\_

\_\_\_\_\_

PLEASE FOLD AND TAPE—  
NOTE: U. S. Postal Service will not deliver stapled forms

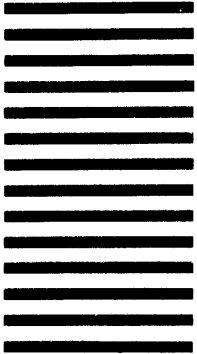


NO POSTAGE  
NECESSARY  
IF MAILED  
IN THE  
UNITED STATES

**BUSINESS REPLY MAIL**  
FIRST CLASS PERMIT NO. 39531 WALTHAM, MA02154

POSTAGE WILL BE PAID BY ADDRESSEE

HONEYWELL INFORMATION SYSTEMS  
200 SMITH STREET  
WALTHAM, MA 02154



ATTN: PUBLICATIONS, MS486

**Honeywell**