SPERRY UNIVAC
COMPUTER SYSTEMS

##
ATTN: CHARLIE GIBBS

00447
CAV208M45541 UP 8511 R2A

UAS

SPERRY UNIVAC
1 - 1816 CORNWALL STREET
VANCOUVER B C
V6J 1C7

##

PUBLICATIONS UPDATE

90/25, 90/30B Systems

Operations Handbook

Operator Reference

UP-8511 Rev. 2–A

This Library Memo announces the release and availability of Updating Package A to "SPERRY UNIVAC Operating System/3 (OS/3) Operations Handbook Operator Reference", UP-8511 Rev. 2.

This update documents the following new features for the 8.0 release:

- A new system command – SHUTDOWN – provides the operator with a method for orderly terminating system activity.
- The SWITCH command has been enhanced with an ALL parameter to allow the new priority level to be in effect for all subsequent job steps.
- The HOLD, BEGIN, DELETE, and DISPLAY console commands have two new features: (1) a HOST parameter, which specifies that only those queued jobs having the specified host-id will be affected; and (2) a DDP parameter, which specifies that only distributed data processing jobs will be affected.
- The SI/SC commands have a new parameter allowing the user to run job control streams that have been saved in an alternate library file.

The following changes are also in effect for this release:

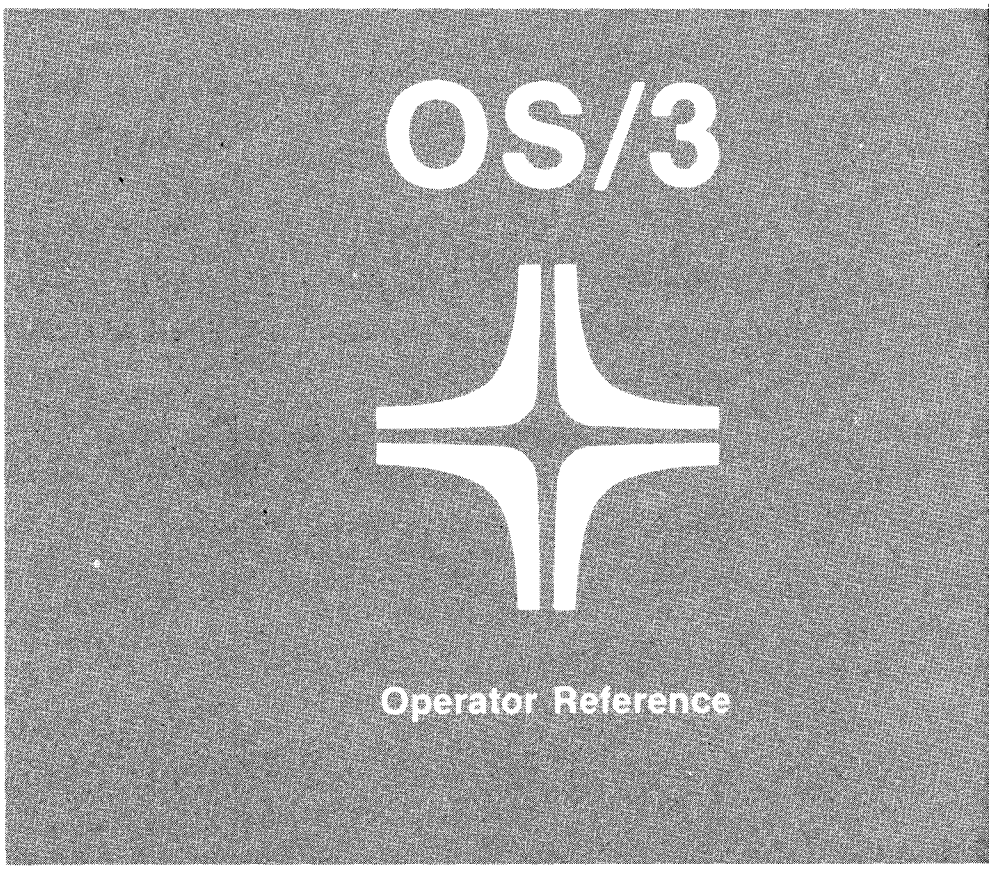
- The 90/30 B system operator information has been removed from UP-8072 and included in this update.
- Support has been decommitted for split cylinder disk file allocation.

All other changes are corrections or expanded descriptions applicable to features present prior to the 8.0 release.

Copies of Updating Package A are now available for requisitioning. Either the updating package only or the complete manual with the updating package may be requisitioned by your local Sperry Univac representative. To receive only the updating package, order UP-8511 Rev. 2–A. To receive the complete manual, order UP-8511 Rev. 2.

LIBRARY MEMO ONLY	LIBRARY MEMO AND ATTACHMENTS	THIS SHEET IS
Mailing Lists BZ, CZ and MZ	Mailing Lists A00, AA0, AA2, 18, 18U, 19, and 19U (Package A to UP-8511 Rev. 2, Cover and 147 pages plus Memo)	Library Memo for UP-8511 Rev. 2–A
		RELEASE DATE: September, 1982

Operations Handbook



Environment: 90/25 and 90/30 B Systems

This document contains the latest information available at the time of preparation. Therefore, it may contain descriptions of functions not implemented at manual distribution time. To ensure that you have the latest information regarding levels of implementation and functional availability, please consult the appropriate release documentation or contact your local Sperry Univac representative.

Sperry Univac reserves the right to modify or revise the content of this document. No contractual obligation by Sperry Univac regarding level, scope, or timing of functional implementation is either expressed or implied in this document. It is further understood that in consideration of the receipt or purchase of this document, the recipient or purchaser agrees not to reproduce or copy it by any means whatsoever, nor to permit such action by others, for any purpose without prior written permission from Sperry Univac.

Sperry Univac is a division of the Sperry Corporation.

This document was prepared by Systems Publications using the SPERRY UNIVAC UTS 400 Text Editor. It was printed and distributed by the Customer Information Distribution Center (CIDC), 555 Henderson Rd., King of Prussia, Pa., 19406.

PRINTED IN U.S.A.

PAGE STATUS SUMMARY

ISSUE: Update A – UP-8511 Rev. 2
RELEASE LEVEL: 8.0 Forward

Part/Section	Page Number	Update Level
Cover/Disclaimer		A
PSS	1	A
Preface	1 2	A Orig.
Contents	1 thru 11 12	A A*
PART 1	Title Page	Orig.
1	1, 2 3 4 thru 12 13 thru 16	A Orig. A A*
PART 2	Title Page	A
2	1 2 thru 18 19 20 thru 42	A Orig. A Orig.
3	1 thru 37	Orig.
4	1 thru 24	Orig.
5	1 thru 50	Orig.
6	1 thru 36	Orig.
7	1 thru 29	Orig.
PART 3	Title Page	Orig.
8	1	A
9	1, 2 2a 3 4 5 thru 8 9	A A* A Orig. A A*
10	1 thru 4 5 6 7 8 9, 10	Orig. A Orig. A Orig. A
10 (cont)	11 12 12a 13, 14 15 16 thru 20 20a 21 thru 24 24a 25 26 thru 30 31 thru 38 39 thru 41 42, 43 44 44a 45 thru 48 48a 49 50 51, 52 53 54, 55 56, 57 58 59 60 thru 64 65, 66 66a 67 thru 70 71, 72 72a 73 thru 84 85 thru 90 91, 92 93 thru 96 97 98 99 100, 101 102, 103 104 105	Orig. A A* A Orig. A A* A A* Orig. A Orig. A Orig. A A* A A* Orig. A Orig. A Orig. A A* Orig. A A* Orig. A Orig. A Orig. A Orig. A Orig.
PART 4	Title Page	Orig.
Appendix A	1, 2	Orig.
Appendix B	1 thru 6 7	A Orig.
Appendix C	1 thru 3 4	A A*
Index	1 2 3 thru 13	A Orig. A
User Comment Sheet		

*New pages

All the technical changes are denoted by an arrow (➔) in the margin. A downward pointing arrow (↓) next to a line indicates that technical changes begin at this line and continue until an upward pointing arrow (↑) is found. A horizontal arrow (➔) pointing to a line indicates a technical change in only that line. A horizontal arrow located between two consecutive lines indicates technical changes in both lines or deletions.

Preface

This manual is designed to instruct and guide the operator, at the system console, in the procedures required to operate the SPERRY UNIVAC 90/25 and 90/30 B Systems under control of the SPERRY UNIVAC Operating System/3 (OS/3). Its intended audience is the operator with a basic knowledge of data processing operations but whose experience is limited to systems other than those of Sperry Univac.

Whereas the system operator uses the procedures and commands described in this handbook to control the entire system, many of these commands are also available to a workstation user under certain limitations. For a description and explanation of workstation operation procedures and available functions, see the workstation user guide, UP-8845 (current version).

One other document relating to the operation of the 90/25 and 90/30 B systems under control of OS/3 is the system messages programmer/operator reference manual, UP-8076 (current version). This manual describes all the system messages you could encounter while operating the 90/25 and 90/30 B data processing systems and the appropriate responses, when necessary.

This operations handbook is divided into the following parts:

- PART 1. SYSTEM DEFINITION

  Briefly describes the minimum and maximum hardware configurations of the 90/25 and 90/30 B systems and the components comprising OS/3.

- PART 2. HARDWARE CHARACTERISTICS

  Describes the function and use of all the operating controls and indicators in the system, the turn-on and turn-off procedures for each device in the system, the operating procedures peculiar to each device, and the recovery procedures applicable to each device. This information is organized and presented under individual device headings to allow you to remove information that does not pertain to your installation.

  We recommend that you go through this part of the manual and remove the information that does not pertain to you, and then flag, or in some way identify, the entries in the table of contents that pertain to the information that was removed. For example, if your system uses only SPERRY UNIVAC 8415 Disk Subsystems, you would remove those pages pertaining to the SPERRY UNIVAC 8413 Diskette Subsystem and the 8416 and 8418 Disk Subsystems (3.3, 3.5, and 3.6). You would then flag these entries in the table of contents.

  Entries in the table of contents should not be obliterated, nor should any removed pages be discarded, as this would preclude their future use in the event your installation should obtain a new device.

- PART 3. STANDARD OPERATING PROCEDURES

  Contains the procedures required to power up, initialize, and run jobs on the 90/25 and 90/30 B systems under the control of OS/3.

- PART 4. APPENDIXES

  Describes the conventions used to illustrate the message and command formats presented in this manual, the procedure to modify the supervisor during IPL, and other aids deemed necessary to help you perform your job.

Each of these parts consists of one or more sections.

Contents

PART 2. HARDWARE CHARACTERISTICS

2. CENTRAL HARDWARE

3. DISK SUBSYSTEMS

7. CARD PUNCH SUBSYSTEM

PART 3. STANDARD OPERATING PROCEDURES

8. SYSTEM TURN-ON AND TURN-OFF PROCEDURES

9. SYSTEM INITIALIZATION PROCEDURES

10. JOB PROCESSING PROCEDURES

PART 4. APPENDIXES

FIGURES

TABLES

1. System Orientation

1.1. GENERAL

The SPERRY UNIVAC 90/25 and 90/30 B Data Processing Systems are general-purpose, disk-oriented computers designed to function in many different data processing environments with equal operating efficiency. This efficiency is achieved through the use of the SPERRY UNIVAC Operating System/3 (OS/3), a multiprogramming software system specifically designed to make maximum use of the capabilities of the system hardware. Figure 1-1 illustrates a typical 90/25 system. The 90/30 B system is similar in appearance.

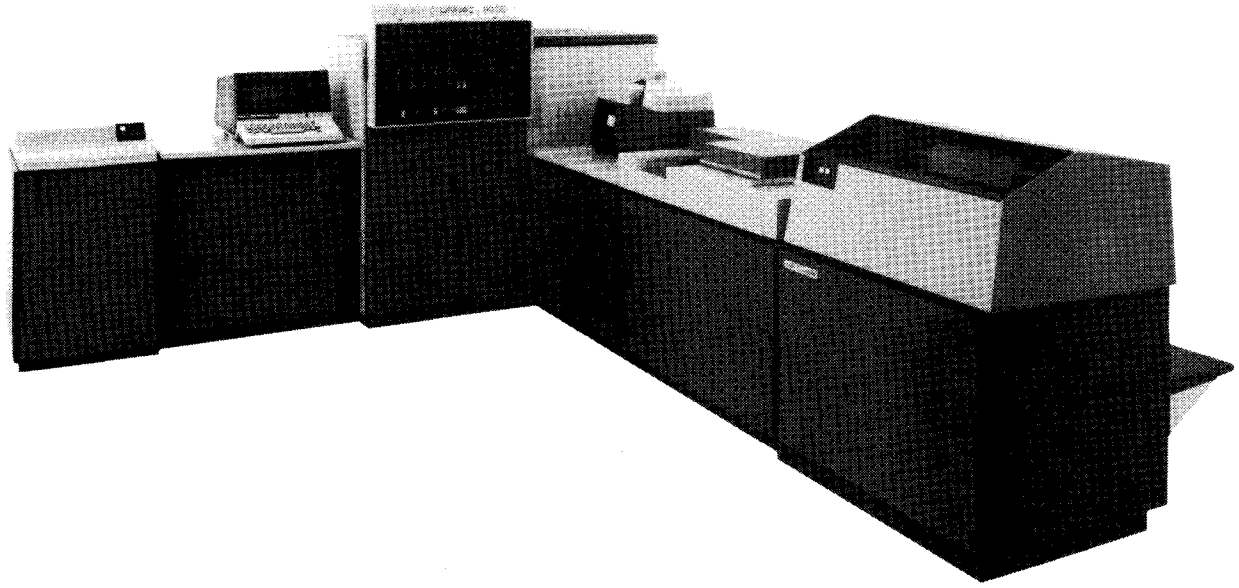


Figure 1—1. 90/25 System

1.2. SYSTEM CONFIGURATION

A diagram of the basic 90/25 and 90/30 B systems configuration is presented in Figure 1-2. Expanded hardware configurations for the 90/25 and 90/30 B systems are presented in Figures 1-3 and 1-4, respectively.

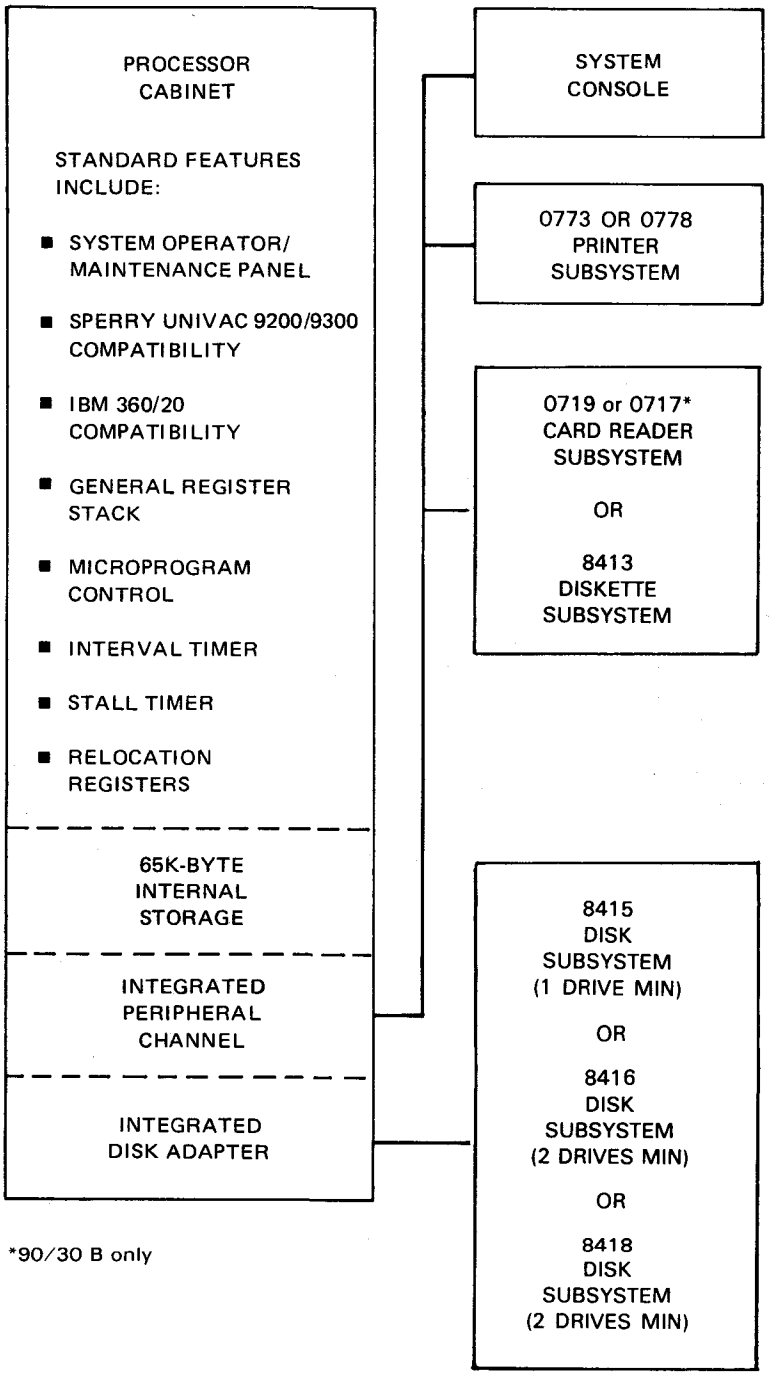


Figure 1—2. Basic 90/25 and 90/30 B Systems Configuration

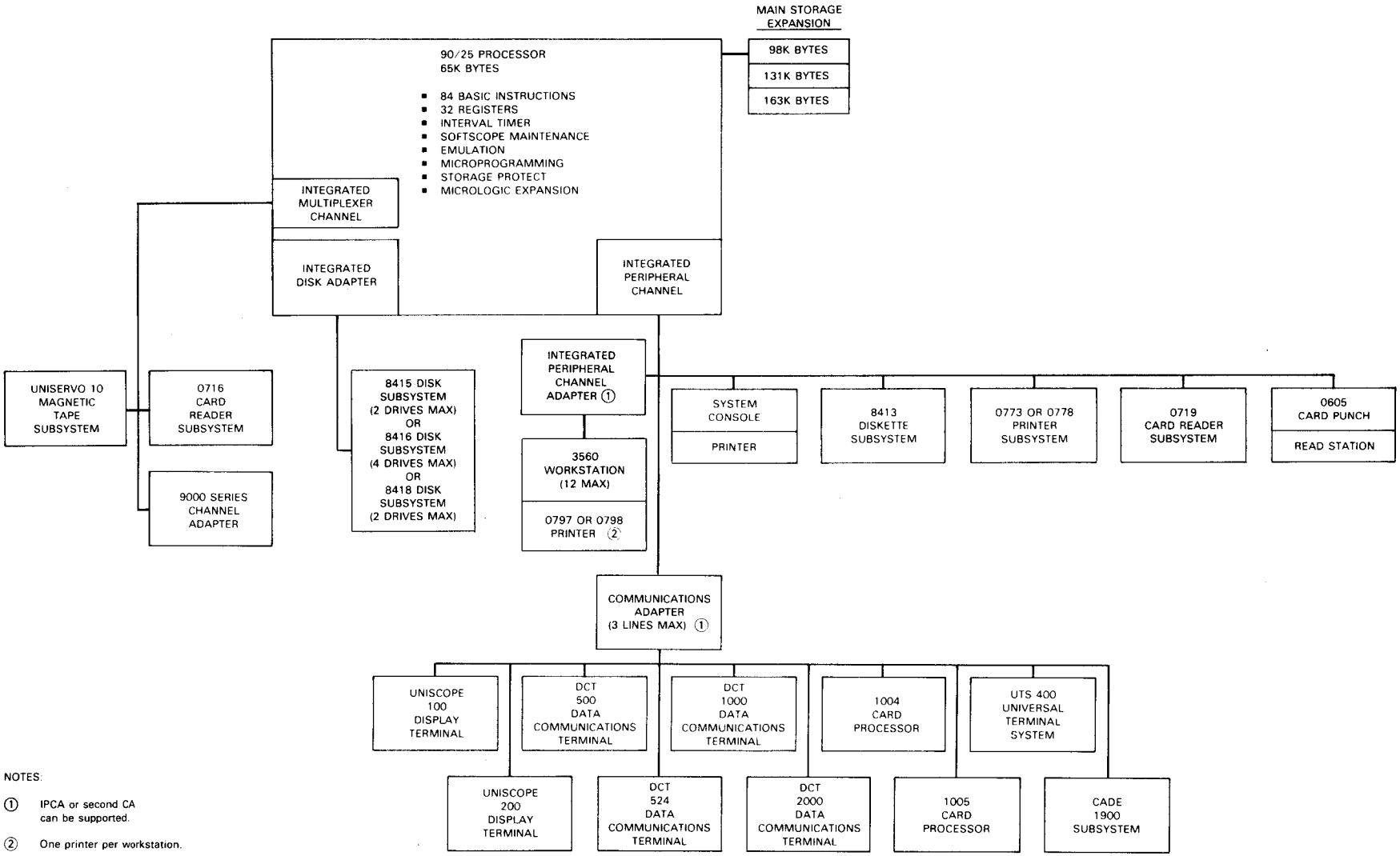
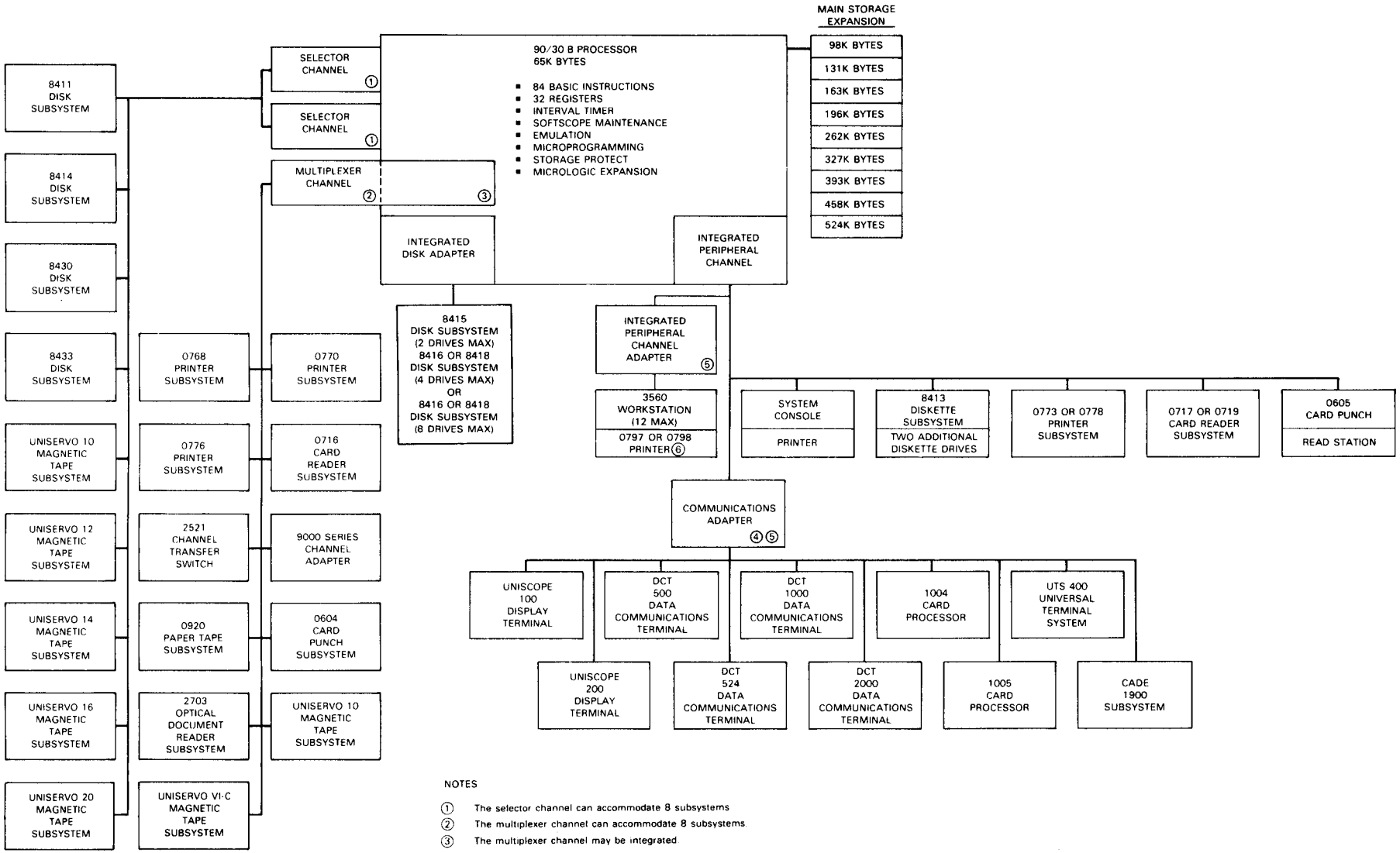


Figure 1—3. 90/25 System with All Hardware Options Shown



NOTES

① The selector channel can accommodate 8 subsystems.
② The multiplexer channel can accommodate 8 subsystems.
③ The multiplexer channel may be integrated.
④ Supports up to 12 full-duplex or 24 half-duplex lines.
⑤ IPCA or second CA can be supported.
⑥ One printer per workstation.

Figure 1—4. 90/30 B System with All Hardware Options Shown

1.3. CENTRAL HARDWARE

The central hardware of a basic 90/25 or 90/30 B system consists of the processor cabinet and a system console, which is a modified UNISCOPE 100 Display Terminal. The central hardware for a fully expanded system also includes a communications output printer (COP) and the processor cabinet optional features shown in Figures 1-3 and 1-4. All of the controls and input/output (I/O) channels required to process a job in the system are built into the basic central hardware components. Enhanced system performance is obtained with the addition of the optional features mentioned.

The characteristics of the basic central hardware components are briefly described in Table 1-1 and are described in detail in Part 2.

A brief description of each of the functional components comprising the central hardware is provided in 1.3.1 through 1.3.3. More detailed descriptions are presented in Section 2.

1.3.1. Processor

The processor is a general-purpose, microprogram-controlled processor that includes the following:

- Basic instruction set
- 32 general registers, 8 working registers, and 8 floating-point registers (optional)
- Interval timer
- Stall timer
- Integrated peripheral channel
- Relocation registers
- Integrated disk adapter
- Operator/maintenance panel
- 65K bytes of main storage expandable to 163K bytes on the 90/25 system; or 65K bytes of main storage expandable to 524K bytes on the 90/30 B system
- Input/output control section

1.3.1.1. Main Storage

Main storage is of the semiconductor type with a 600-nanosecond half-word read/write cycle time. The minimum main storage is 65K bytes, expandable to 163K bytes on the 90/25 system and 524K bytes on the 90/30 B system. It is constructed in modular form and is packaged as an internal part of the processor. The system automatically monitors and protects the main storage contents to ensure data integrity. Power losses experienced by the system result in loss of all data in main storage.

1.3.1.2. Input/Output Control Section

The input/output control section initiates, directs, and monitors the transfer of data between main storage and the peripheral subsystems. After an I/O instruction is initiated, the data is transferred independently of other processor functions; i.e., the I/O and the processor operate concurrently. The I/O control section is the processor interface to the integrated peripheral channel, the integrated disk adapter, and the selector and multiplexer channels.

1.3.1.3. Integrated Peripheral Channel

The integrated peripheral channel (IPC) coordinates all information transfers between main storage and the integrated peripheral devices: system console, card reader, card punch, printer, diskette subsystem, integrated peripheral channel adapter and workstation, and the communications adapter. The IPC is a half-duplex channel that transfers commands, data, status, and sense information. Input/output activity is initiated by the processor upon issuance of a start I/O instruction to IPC. This instruction results in the transfer of a command to the control logic of a specific peripheral device. The command specifies the type of operation to be performed and is executed on an individual basis. The high transfer rate of the IPC permits simultaneous operation of all integrated peripherals.

1.3.1.4. Integrated Disk Adapter

The integrated disk adapter (IDA) acts as a combination channel and control unit.

On the 90/25 system, the IDA is designed to operate with a minimum of one 8415 or two 8416 or 8418 disk drive units. The IDA operates with a maximum of:

- two 8415;
- four 8416; or
- two 8418 disk drive units.

The 90/30 B system IDA operates with a minimum of one 8415 or two 8416 or 8418 disk drive units, and a maximum of:

- two 8415 and four 8416 or 8418 disk drive units; or
- eight 8416 or 8418 disk drive units.

In all cases, 8416 and 8418 units may be used in any combination.

1.3.1.5. Micrologic Expansion Feature

The micrologic expansion feature provides a repertoire of 64 additional instructions, four registers (each 64 bits long), and expanded control storage. It provides micrologic for execution of 44 floating-point instructions in both long and short, normalized and unnormalized formats, and micrologic for the execution of 20 additional nonprivileged instructions.

1.3.1.6. Storage Protection Feature

The storage protection feature provides read/write protection on access to main storage and two additional privileged instructions (SSK, ISK). It protects up to 131,072 bytes of main storage for the 90/25 system, and up to 524,288 bytes of main storage for the 90/30 B system.

1.3.1.7. Storage Expansion Feature

The storage expansion feature in the 90/25 system provides for increasing the size of internal main storage up to 163K bytes. Storage expansions are in 32K-byte increments. In the 90/30 B system, the storage expansion feature allows main storage increases up to 524K bytes in either 32K- or 64K-byte increments.

1.3.1.8. Integrated Communication Adapter Feature

The integrated communication adapter feature provides for interfacing the IPC with a communication adapter.

1.3.1.9. Integrated Multiplexer Channel

- 90/25 System

  The integrated multiplexer channel on the 90/25 system provides I/O capability between the processor and up to three low-speed subsystems, with a throughput rate of 83K bytes per second. It operates in multiplexed mode; that is, the channel services several concurrently operating subsystems by assigning the input/output interface to a subsystem only long enough to transfer one or a few bytes of information. The multiplexer channel controls up to three subsystems and initiates all input/output operations by issuing input/output instructions to a selected subchannel subsystem. When the operation is successfully initiated, the multiplexer channel controls the flow of data between the main storage and the subsystem, independent of the processor. At the completion of the input/output operation, the status of the multiplexer channel is presented to the processor.

- 90/30 B System

  The integrated multiplexer channel feature on the 90/30 B system provides I/O capability between the processor and up to eight subsystems with a throughput rate of 83K bytes per second. The multiplexer channel may or may not be integrated, depending on the system configuration. If the system configuration includes an I/O expansion cabinet, this feature cannot be used.

1.3.2. System Console

The system console provides the main interface for operator interaction with the processor. The system console is a modified UNISCOPE 100 Display Terminal, which accepts data from the keyboard of the console control unit, displays the data, and transfers the data to the integrated peripheral channel.

Data entered into the keyboard is displayed on the screen in a 64-character-per-line by 16-line format, providing a total display of 1024 characters. Displayable characters consist of the 64-character (including space) ASCII set plus control characters.

1.3.3. Communications Output Printer (COP)

The COP is a freestanding auxiliary output device for the system console. Capable of printing at a maximum rate of 30 characters per second, the COP can produce from one to six printed copies on edge-sprocketed forms 11 inches (27.9 cm) long and 3-5/8 inches (9.19 cm) wide to 14-7/8 inches (37.76 cm) wide. Operation is asynchronous. The COP requires only ac power connection and an interface connection to the system console.

1.3.4. I/O Expansion Cabinet (90/30 B System)

The I/O expansion cabinet for the 90/30 B system provides increased processor I/O capability by providing up to two selector channels and one multiplexer channel. Addition of these channels allows standard peripheral subsystems to operate with the system, in addition to the integrated peripheral subsystems.

1.3.4.1. Selector Channels (90/30 B System)

Each selector channel for the 90/30 B system controls the exchange of information between subsystems (no more than eight) and processor main storage. The selector channels operate in the burst mode. (For example, one of eight possible subsystems retains control of the interface for the duration of its I/O operation. Simultaneously, other subsystems can be executing previously initiated operations that do not involve data transfer over the I/O interface.) The processor initiates all I/O operations to the selector channel and the specific subsystem connected to the channel. When the operation is successfully initiated, the channel maintains control of the data transfers between main storage and the subsystem independently of the processor. Upon completion of the I/O operation, the status of the channel and the subsystem is presented to the processor. One or two selector channels may be added to an expanded 90/30 B system configuration.

1.3.4.2. Multiplexer Channel (90/30 B System)

The multiplexer channel for the 90/30 B system is similar in operation to the selector channel except that it operates in multiplexed mode. That is, the channel services several concurrently operating subsystems by assigning the input/output interface to a subsystem only long enough to transfer one or a few bytes of information. The multiplexer channel controls up to eight subsystems and initiates all input/output operations by issuing input/output instructions to a selected subchannel and subsystem. When the operation is successfully initiated, the multiplexer channel controls the flow of data between the main storage and the subsystem, independent of the processor. At the completion of the input/output operation, the status of the multiplexer channel is presented to the processor. One multiplexer channel, either integrated or with an I/O expansion cabinet (external) may be added to an expanded 90/30 B system configuration.

1.4. INPUT/OUTPUT SUBSYSTEMS

The I/O subsystems available for use with the 90/25 and 90/30 B systems include: a workstation subsystem, disk and diskette subsystems, magnetic tape subsystems, printer subsystems, card reader and card punch subsystems, and a communications adapter with remote I/O subsystems. The specific local subsystems and their characteristics are briefly described in Table 1-1. Those subsystems common to both the 90/25 and 90/30 B systems are described in more detail in Sections 3 through 7, with the exception of the workstation, the communications adapter, and the subsystems exclusive to the 90/30 B system. The hardware references for these subsystems are listed in Table 1-2.

Table 1—1. Input/Output Subsystems (Part 1 of 3)

Subsystem	Description
Workstation Subsystem	
3560 workstation subsystem	12 or 24 lines, 64-character set Standard and extended function keyboards
Disk and Diskette Subsystems	
8411 disk subsystem*	7.25 million bytes/pack 156K bytes/second transfer time 75 milliseconds average access time

Table 1—1. Input/Output Subsystems (Part 2 of 3)

Subsystem	Description
Disk and Diskette Subsystems (cont)	
8413 diskette subsystem	242,944 bytes/diskette 4680 records/minute read rate 3120 records/minute write rate
8414 disk subsystem*	29.17 million bytes/pack 312K bytes/second transfer time 60 milliseconds average access time
8415 disk subsystem	24.9 million bytes fixed 8.3 million bytes removable 33.1 million bytes/pack 625K bytes/second transfer rate 33 milliseconds average access time
8416 disk subsystem	28.95 million bytes/pack, maximum 625K bytes/second transfer time 27 milliseconds average access time
8418 disk subsystem	28.9 million bytes/pack, maximum 625K bytes/second transfer time 27 milliseconds average access time
8430 disk subsystem*	100 million bytes/pack 806K bytes/second transfer time 27 milliseconds average access time
8433 disk subsystem*	200 million bytes/pack 806K bytes/second transfer time 30 milliseconds average access time
Magnetic Tape 7- and 9-Track, Phase and NRZI	
UNISERVO 10 Magnetic Tape Subsystem	40K bytes/second transfer rate
UNISERVO 12 Magnetic Tape Subsystem*	68.3K bytes/second transfer rate
UNISERVO 14 Magnetic Tape Subsystem*	96K bytes/second transfer rate
UNISERVO 16 Magnetic Tape Subsystem*	192K bytes/second transfer rate
UNISERVO 20 Magnetic Tape Subsystem*	320K bytes/second transfer rate
UNISERVO VI-C Magnetic Tape Subsystem*	34.1K bytes/second transfer rate
Printers	
0768 printer subsystem*	900–1100 lines/minute, 63-character set 840–2000 lines/minute, 94-character set 1200–1600 lines/minute, 63-character set
0770 printer subsystem*	1435, 2320, or 3000 lines/minute, 24-character set 800, 1400, or 2000 lines/minute, 48-character set 112, 213, or 337 lines/minute, 348-character set

Table 1—1. Input/Output Subsystems (Part 3 of 3)

Subsystem	Description
Printers (cont)	
0773 printer subsystem	500 lines/minute, 48-character set 400 lines/minute for the 48-character set of a 48/16 combination character set 670 lines/minute for the 16-character set of a 48/16 combination character set 310 lines/minute, 84-character set 217 lines/minute for the 128-character special 96/(6-16) character set 217 or 500 lines/minute for ASCII 114 lines/minute, 256-character set
0776 printer subsystem*	760, 900, or 940 lines/minute, 48-character set 600, 730, or 750 lines/minute, 64-character set 420, 520, or 540 lines/minute, 96-character set 325, 400, or 420 lines/minute, 128-character set 225, 280, or 290 lines/minute, 192-character set 115, 145, or 150 lines/minute, 384-character set
0778 printer subsystem	300 or 500 lines/minute, 48-character set 240 or 560 lines/minute for the 48-character set of a 48/16 combination character set 240 or 415 lines/minute, 64-character set 180 or 333 lines/minute, 85-character set 120 or 240 lines/minute, 128-character set
Card Punches	
0604 card punch subsystem*	80 columns, 250 cards/minute
0605 card punch subsystem	80 columns, 75 cards/minute 28 columns, 160 cards/minute
Card Readers	
0716 card reader subsystem	51/66/80 columns, 600 or 1000 cards/minute
0717 card reader subsystem*	51/66/80 columns, 500 cards/minute
0719 card reader subsystem	51/66/80 columns, 300 cards/minute
Paper Tape Subsystem*	
0920 paper tape subsystem*	300 characters/second, 10 characters/inch
Optical Document Reader*	
2703 optical document reader (ODR)*	300 documents/minute, 6-inch documents 600 documents/minute with speed-up feature
Communication Interface	
Communications adapter	Supports up to 3 half-duplex or 1 full-duplex and 2 half-duplex communication lines at speeds up to 56K bits/second
90/30 communications adapter*	Supports up to 12 half-duplex or 6 full-duplex (24 half-duplex or 12 full-duplex with expansion feature) communication lines at speed up to 56K bits/second

*90/30 B only

Table 1—2. Hardware Documentation (Part 1 of 2)

Processor/Device	Document Number (Current Version)			
	General Description, UP-	Subsystem/ Programmer Reference, UP-	Operator Reference, UP-	Programmer/ Operator Reference, UP-
Processors				
90/25		8460	8459	
90/30 B		8052	8097	
Integrated Peripheral Channel (IPC)				
90/25		8458		
90/30 B		8041		
Card Punch				
0604*		7772	7773	
0605	8192		8088	
Card Reader				
0716			7921	
0717*	8196		8089	
0719	8493			
Printer				
0768*			7931	7688
0770*		8016	7938	
0773	8191		8086	
0776*	8354	8441	8250	
0778	8524		8525	
8541 (COP)	7939			
0797	9159		9160	
0798	8871		8882	
UNISERVO Magnetic Tape				
10	8206	8205	8207	
12*				7661
14*	8206	8205	8207	
16*				7661
20*			7956	
VI-C*				7644
Disks				
8411*	7605	7977	7802	
8414*	7691	7977	7802	
8415			8511 and Section 3 in this manual	
8416			8511 and Section 3 in this manual	
8418		8362	8361	
8430*		8344	8343	
8433*		8344	8343	
Diskette				
8413	8463		8490	
Workstation				
3560		8742	8880	

Table 1—2. Hardware Documentation (Part 2 of 2)

Processor/Device	Document Number (Current Version)			
	General Description, UP-	Subsystem/ Programmer Reference, UP-	Operator Reference, UP-	Programmer/ Operator Reference, UP-
Paper Tape 0920*	7595	7998	7830	
Optical Document Reader (ODR) 2703*	7710	7993	7994	
Channel Transfer Switch 2521*			8489	
Communications Adapters	8273	8247		
UNISCOPE 100/200	8155		7788	7807
DCT 475/500/524 Data Communications Terminal	7804		7832	7836
DCT 1000 Data Communications Terminal	7782		7827	7859
DCT 2000 Data Communications Terminal	7511		7545	7532
Card Processor 1004/1005			7839	
UTS 400 Universal Terminal System	8358		8357	
CADE 1900	8335			

*90/30 B only

1.5. OPERATING SYSTEM

OS/3 (Figure 1-5) is composed of a group of major programs: supervisor, job control, data management, integrated communications access method, language processors, system service programs, emulators and transition aids, information management system, data base management system, application programs, and interactive services.

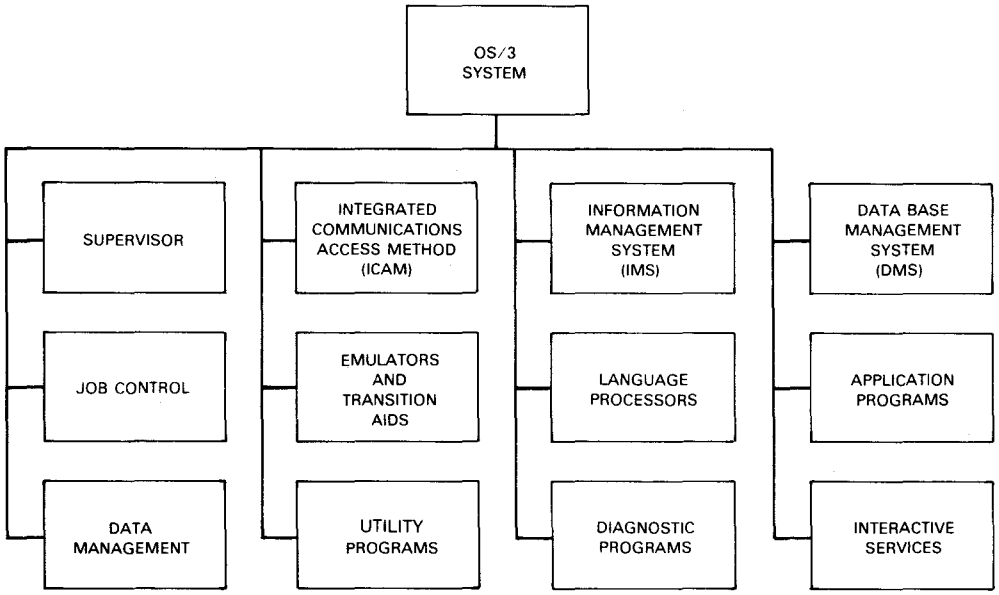


Figure 1—5. Operating System/3 Components

1.5.1. Supervisor

The supervisor is the part of the operating system that interfaces with the user programs to provide the necessary control for the optimum utilization of the system hardware and software. It controls the physical I/O operations, system resource allocation on a dynamic basis, task switching to achieve multitasking, hardware interrupt servicing, communications with the system operator, and interface to user programs with the system hardware. To accomplish this, the supervisor is composed of the following program elements:

- Interrupt control
- Priority control
- Transient control
- Physical I/O controls
- Resource allocation
- Task control
- Interrupt timer and day clock services
- Program management
- System console management
- File services

- Program error handling
- Cooperative/symbiont operations
- Physical input/output control system (PIOCS)
- Debugging aids

1.5.2. Job Control

Job control is a nonresident program of the operating system responsible for controlling the orderly initiation and termination of jobs within a multiprogrammed environment. The job control services are performed prior to execution of the initial job step of a job, during the transition between job steps, and at the conclusion of a job. Some of the services of job control are:

- Volume label and file label storage
- Job control stream file maintenance
- Job scheduling by priority
- Main storage allocation and reallocation
- Peripheral device assignment
- Program restart

The functions of job control are implemented by the programmer through the job control language or by the operator through the system console commands. These sequenced control statements form the control stream that defines a job's facility requirements and directs the execution of the job. The job control statements, through the job control stream, function as an interface between the programmer and OS/3.

1.5.3. Data Management

Data management provides the interface between the hardware-oriented I/O facility and the user program. The data management facilities consist of logical input/output control stream (IOCS) modules, transient routines, declarative macroinstructions, and imperative macroinstructions.

1.5.4. Integrated Communications Access Method

The communication software necessary to support remote terminals or processors is controlled by two logical levels of software. These levels are:

- the communications physical input/output control system (CPIOCS) and the communication symbionts; and
- the message control program (MCP).

There are four user levels (interfaces) that communicate with the logical control levels via declarative and imperative macroinstructions. These are:

- CPI - communications physical interface
- DDI - direct data interface
- STDMCP - standard GET/PUT interface
- TCI - transaction control interface

1.5.5. Emulators and Transition Aids

There are two emulation programs and one transition program that adapt the instruction repertoire and peripheral characteristics of existing systems to OS/3. They are the:

- IBM 360/20 emulator
- SPERRY UNIVAC 9200/9300 Emulator
- IBM System/3 transition program

1.5.6. Utility Programs

The OS/3 utility programs make available to the system the means for sorting data into a specified order, merging data to facilitate processing, maintaining files on magnetic disk storage, linking output modules of language processors into executable programs, copying input cards, magnetic tapes, disk or diskette files to any other card, magnetic tape, disk or diskette, or printer device.

The major utility programs include:

- Data utilities
- Linkage editor
- System librarian
- Sort/merge
- Disk, diskette, and tape prep routines
- System utility

1.5.7. Language Processors

Six language processors are available with OS/3: assembler, COBOL, FORTRAN, report program generator (RPG II), BASIC, and ESCORT. All language processor input can be on punched cards, magnetic tape, or disk files; all output can be recorded on magnetic tape or disk files. All processor output is in a common system output format.

1.5.8. Diagnostic Programs

The diagnostic programs provided with OS/3 are hardware maintenance routines that can be executed concurrently with user programs. These programs are intended to be run as confidence tests by the system operator, and as diagnostic and maintenance tests by the customer engineer.

1.5.9. Application Programs

Application programs are specialized programs available to a user but not provided as part of the standard software package. These programs are directed towards handling problems distinctive to a particular user, and include program evaluation and review techniques/critical method analysis (PERT/CPM) and linear programming (LP).

1.5.10. Information Management System

The information management system (IMS) used with OS/3 is common to the 9000 Series and facilitates access to information stored in data files. IMS provides a terminal-oriented data retrieval and update capability for managerial and clerical personnel and, thereby, relieves them of needing to learn complex methods employed by programming personnel. IMS is supplied by Sperry Univac in the form of application programs called UNIQUE, which require programming effort. Also, coding required for line and device handlers is provided.

1.5.11. Data Base Management System

The data base management system (DMS) is a collection of system programs that support the development of integrated data bases. These programs provide for the description, initialization, creation, accessing, maintenance, backup, and recovery of data base. The languages used in the description and manipulation of DMS data bases are derived from the CODASYL data base specifications. A data base may be accessed by batch application programs and communications application programs.

1.5.12. Interactive Services

Interactive services provides an extensive interactive command set that is available to a user as part of the standard software package. These commands enable you to control the interactive system environment, including all jobs within the system, all workstation users (local locations), and all terminal users (local and remote locations).

Interactive services also performs a variety of functions including program creation and file/data manipulation. The interactive facilities used to perform these functions are:

- Screen format services
- Interactive data utilities
- General editor
- Interactive job stream preparation

PART 2. HARDWARE CHARACTERISTICS

The hardware described in the following sections is common to both the 90/25 and 90/30 B systems. The additional hardware available with the 90/30 B system is described in the documents listed in Table 1-2.

2. Central Hardware

2.1. PROCESSOR CABINET

The processor cabinets (Figure 2-1) for the 90/25 and the 90/30 B systems are identical in appearance. The cabinet houses the arithmetic and control logic circuits of the central processor unit, 62K bytes of main storage, the integrated disk adapter (IDA), integrated peripheral channel (IPC), and, optionally, the integrated communication attachment and adapter and the integrated multiplexer channel. It also contains the primary control panel for the system (Figure 2-2, operator/maintenance panel), from which most of the system operations are controlled. The housing unit used for the integrated communications equipment also doubles as a desk for the system console.

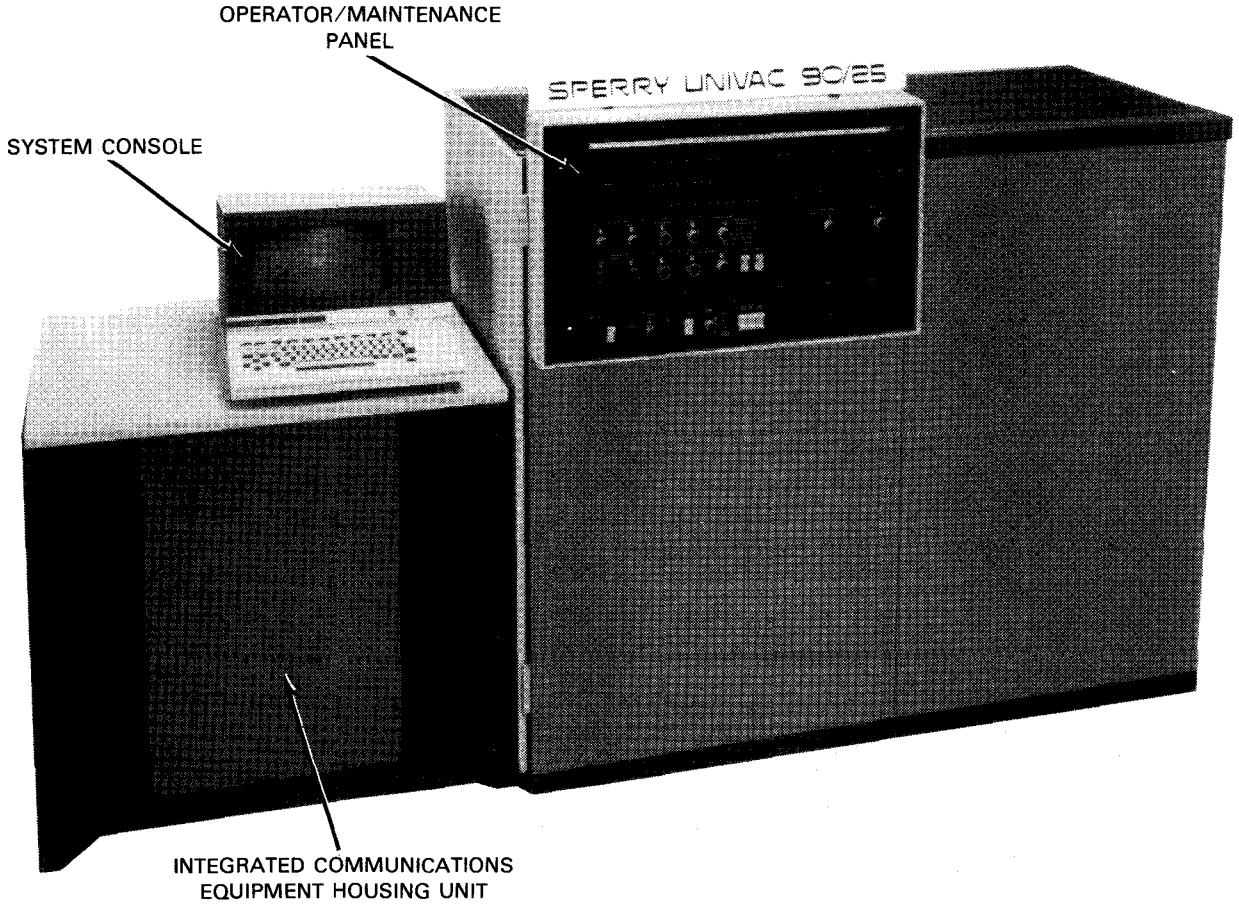


Figure 2—1. System Processor Cabinet (Front View)

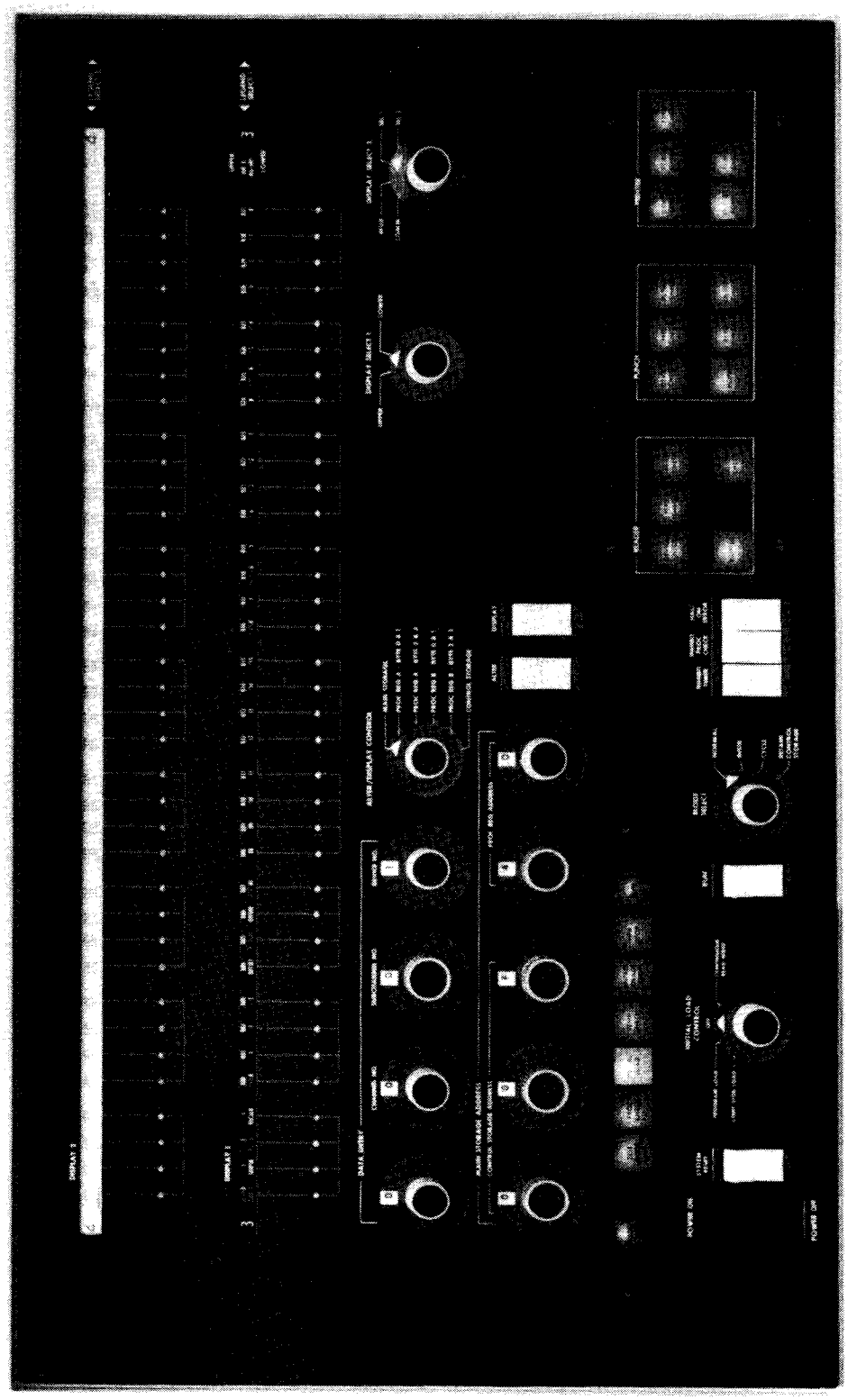


Figure 2—2. Processor Cabinet, Operator/Maintenance Panel

2.2. SYSTEM CONSOLE

The system console (Figure 2—8) is the primary means available to the operator for communication with the system. It consists of a display screen, a control/indicator panel (Figure 2—9), and a keyboard (Figure 2—10) for entering information.

The system console operates in a protected and unprotected format; that is, messages transmitted to the system console from the processor cannot be overwritten by the operator and are thus said to be in protected format. Messages generated by the operator, however, can be overwritten (edited and changed) by the operator and are thus said to be in unprotected format.

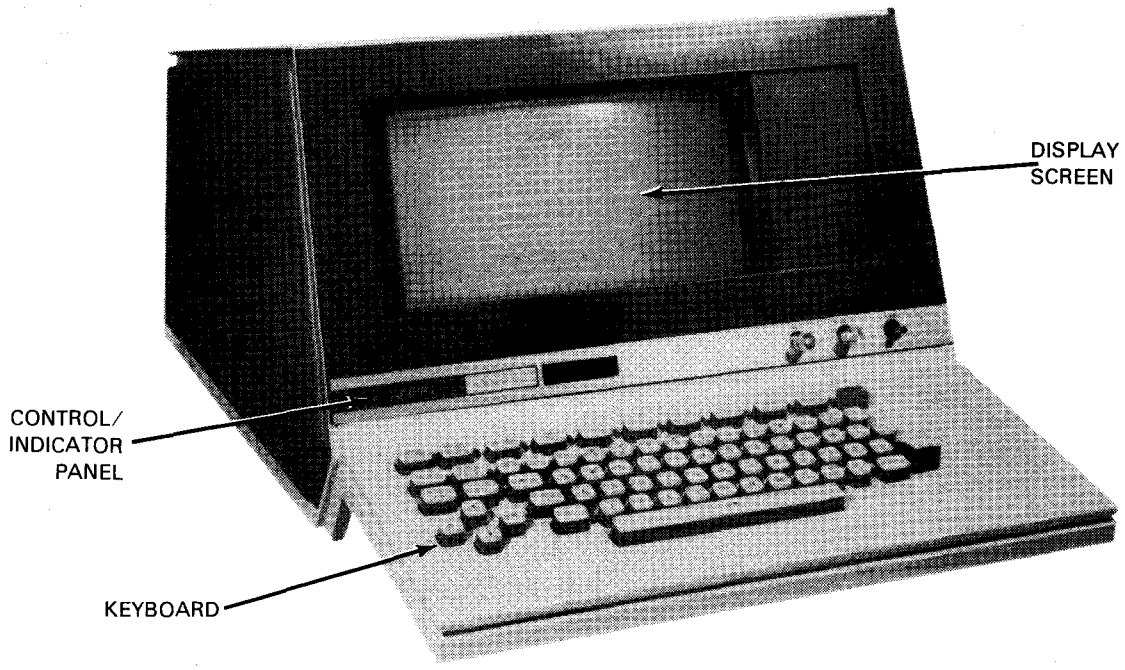


Figure 2—8. System Console

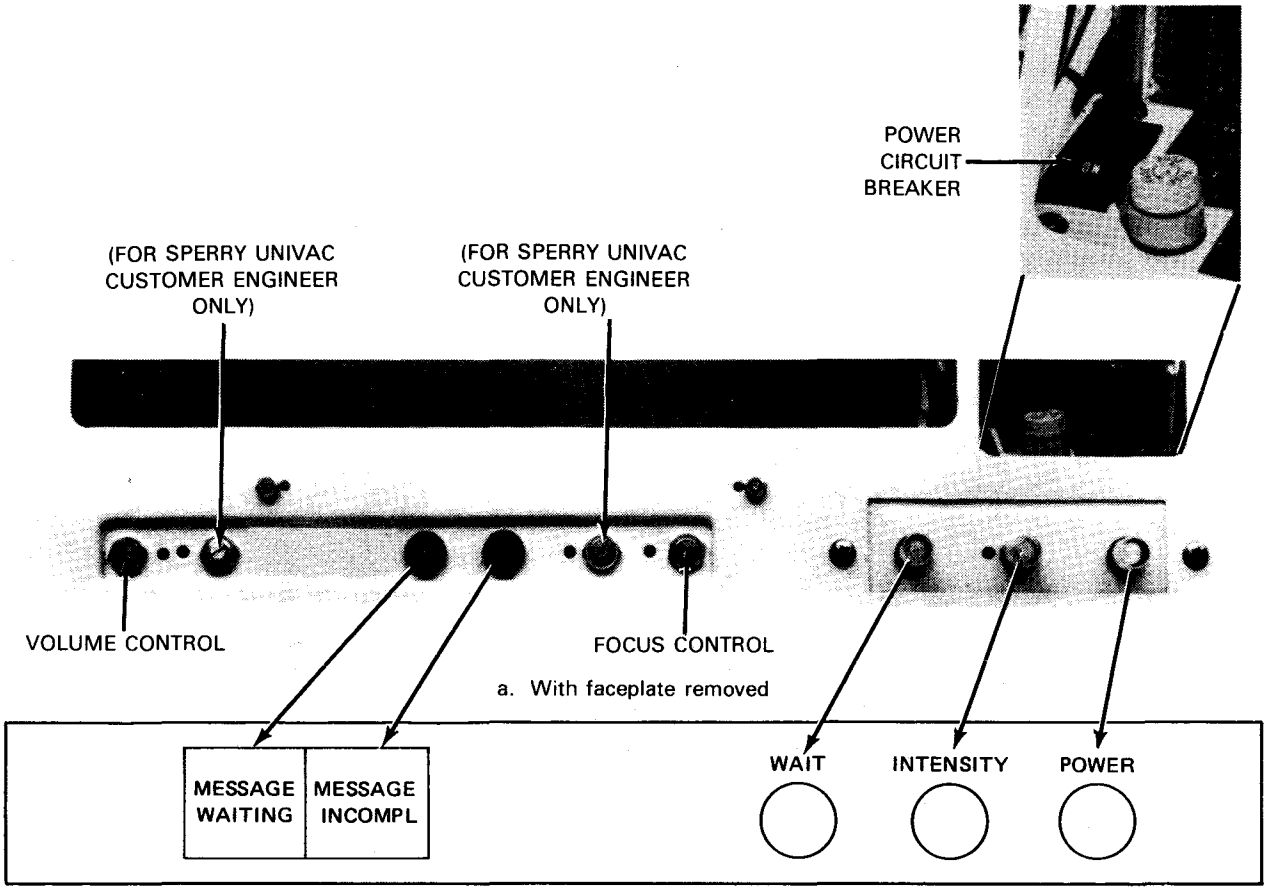


Figure 2—9. System Console, Control/Indicator Panel

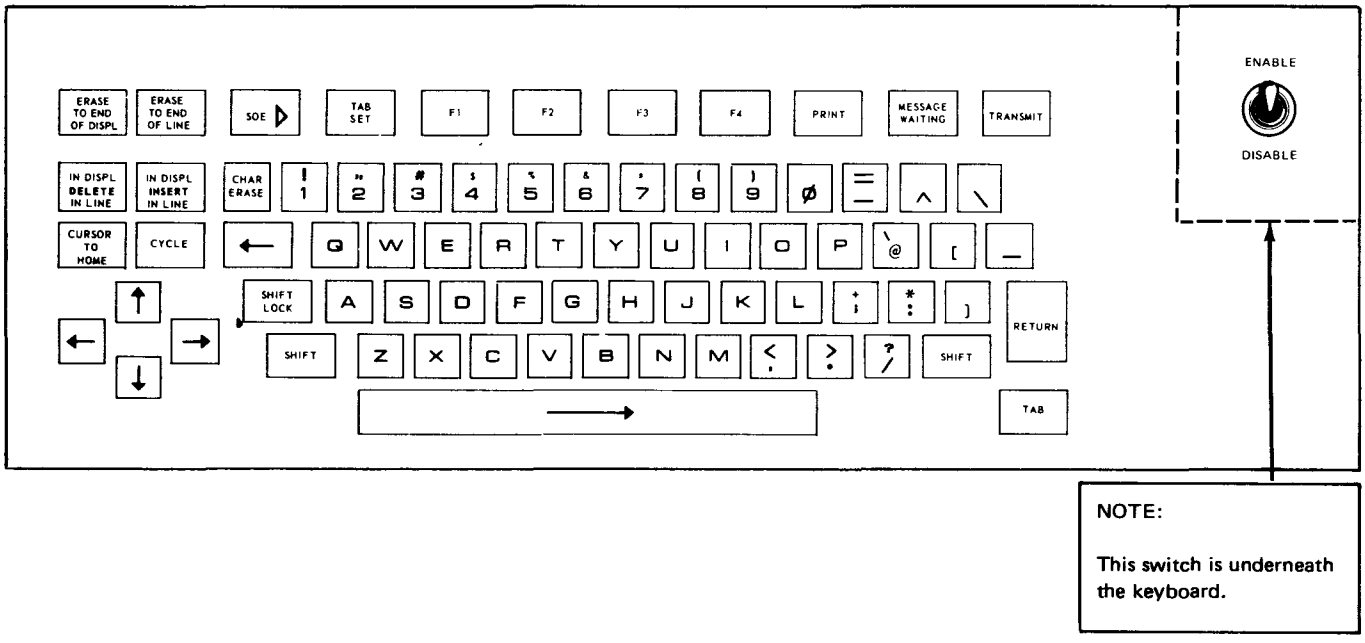


Figure 2—10. System Console Keyboard

8. System Turn-On and Turn-Off Procedures

8.1. SYSTEM TURN ON

To turn on your 90/25 or 90/30 B system from a full power-off condition, proceed as follows:

1. Set the system circuit breakers (wall mounted) to the **ON** position.

2. Set the **POWER ON/POWER OFF** switch, on the processor operator/maintenance panel (Figure 2—2), to the **POWER ON** position. The **POWER CYCLING** indicator lights during the power-up sequence. All other indicators remain off.

   When the power-up sequence is complete, the **POWER ON** indicator on the operator/maintenance panel lights and all other indicators are extinguished. This condition indicates that operating power is applied to all the central hardware and on-line peripheral devices.

   If the **POWER CYCLING** indicator, as well as the **POWER ON** indicator, on the operator/maintenance panel lights at the end of the power-up sequence, and all the other indicators remain extinguished, a stall condition in the power-up sequencing of the system has occurred. To remedy this condition, set the **POWER ON/POWER OFF** switch on the operator/maintenance panel to the **POWER OFF** position. Check the peripheral devices and central hardware circuit breakers and switches to ensure that they are set to their respective power-on positions. Then set the **POWER ON/POWER OFF** switch to the **POWER ON** position.

   If, after this power-up sequence is completed, the **POWER CYCLING** indicator, as well as the **POWER ON** indicator lights, and all other system indicators remain extinguished, refer the situation to your Sperry Univac customer engineer.

8.2. SYSTEM TURN OFF

To turn off your 90/25 or 90/30 B system, proceed as follows:

1. Set the **POWER ON/POWER OFF** switch, on the processor operator/maintenance panel, to the **POWER OFF** position. All the indicators on the operator/maintenance panel light momentarily and then extinguish.

2. If the system is to remain unused for a period of time, set the system circuit breakers (wall mounted) to the **OFF** position.

9. System Initialization Procedures

9.1. GENERAL

Initialization of the 90/25 or 90/30 B system is a 2-step operation:

1. Loading and initializing control storage (9.2)

2. Loading and initializing the resident portion of the supervisor (9.3).

Whenever power is removed from the system, both these operations must be performed to initialize the system after power is reapplied. Whenever a nonrecoverable error occurs, only the supervisor need be reloaded and initialized. The only time control storage needs to be reloaded and initialized, after a nonrecoverable error occurs, is when you are unable to load and initialize the supervisor.

9.2. CONTROL STORAGE LOAD PROCEDURE

Whenever the 90/25 or 90/30 B system is turned on from a full power-off condition or the initial program load (IPL) operation cannot be completed successfully, control storage must be loaded and initialized. To load and initialize control storage, proceed as follows:

1. Place the system resident (SYSRES) disk pack containing the control storage code on a suitable disk drive unit and set the disk drive to the run state in accordance with the operating procedures presented in Section 3.

2. Perform the following operations at the operator/maintenance panel.

   a. Set the **INHIBIT TIMER** switch to its off (down) position.

   b. Set the **INHIBIT PROC CHECK** switch to its off (down) position.

   c. Set the **HALT ON ERROR** switch to its off (down) position.

      NOTE:

      The **INHIBIT TIMER, INHIBIT PROC CHECK,** *and* **HALT ON ERROR** *switches are never set to the on position during normal system operation.*

   d. Set the **MODE SELECT** switch to **NORMAL**.

   e. Set the **INITIAL LOAD CONTROL** switch to **CONT STOR LOAD**.

NOTE:

At this point you specify the 3-digit number for the control storage load device identification (did). The did is a concatenation of the channel, subchannel, and selected device number. The did is represented by the device address you set in steps f, g, and h.

f. Set the **DATA ENTRY CHANNEL NO.** switch to the channel number on which the selected disk drive is connected. The number assigned to this channel is 3 for the 90/25 system. On the 90/30 B system, the numbers assigned to these channels are:

Channel	Assigned Number
Integrated disk adapter	3
Selector channel 1	4
Selector channel 2	6

g. Set the **DATA ENTRY SUBCHANNEL NO.** switch to the subchannel number assigned to the selected disk drive unit. The subchannel assignment is 0 for the 90/25 system. On the 90/30 B system, the numbers assigned to these channels are:

Channel	Assigned Number
Integrated disk adapter	0
Selector channel 1	0–7
Selector channel 2	0–7

h. Set the **DATA ENTRY DEVICE NO.** switch to the actual physical unit number assigned to the selected disk drive unit (0 through F).

i. Press the top portion of the **SYSTEM RESET** switch twice. The **TEST MODE** indicator lights.

j. Press the top portion of the **RUN** switch. The **INITIAL LOAD** indicator lights and remains lit until control storage has been successfully loaded (**HPR STOP** indicator lights).

If the **INITIAL LOAD** indicator fails to extinguish and the **PROC CHECK** or **CONTROL STORAGE** indicators light, control storage was not successfully loaded and steps 2i and 2j must be repeated.

NOTE:

Do not disturb any switch settings until control storage loading is complete.

If the proper indications cannot be obtained, try loading control storage from another disk drive unit. If the abnormal indications persist, refer the problem to your Sperry Univac customer engineer.

9.3. INITIAL PROGRAM LOAD PROCEDURE

Before the system can be used to run productive jobs, the resident portion of OS/3 (the supervisor) must be loaded into main storage and initialized. This operation can only be performed after control storage is loaded and initialized, and must be performed whenever control storage is loaded or a nonrecoverable error occurs.

To load and initialize the supervisor (the IPL procedure), proceed as follows:

CAUTION

Make certain the system is in the idle condition (no jobs are active) before you perform the IPL procedure. Otherwise, the SYSRES volume table of contents (VTOC) may be left in a nonrecoverable state requiring that a new SYSRES volume be generated.

1. Place the system resident (SYSRES) disk pack containing the initial program load (IPL) routine on a suitable disk drive unit and set the disk drive to the run state in accordance with the operating procedures presented in Section 3.

   NOTES:

   1. *The control storage code and initial program load routine are stored on the same disk pack.*
   2. *If an 8418 disk is being used for the SYSRES, the density setting on the drive must match the density of the recorded disk pack. The PROC CHECK indicator lights if the densities do not match.*

2. Set the controls and switches on the operator/maintenance panel to the following positions:

   a. Set the **INHIBIT TIMER** switch to its off (down) position.

   b. Set the **INHIBIT PROC CHECK** switch to its off (down) position.

   c. Set the **HALT ON ERROR** switch to its off (down) position.

      NOTE:

      The **INHIBIT TIMER, INHIBIT PROC CHECK,** *and* **HALT ON ERROR** *switches are never set to the on position during normal system operation.*

   d. Set the **MODE SELECT** switch to **NORMAL** position.

   e. Set the **INITIAL LOAD CONTROL** switch to **PROGRAM LOAD** position.

   NOTE:

   At this point you specify the 3-digit number for the program load device identification (did). The did is a concatenation of the channel, subchannel, and selected device number. The did is represented by the device address you set in steps f, g, and h.

f. Set the **DATA ENTRY CHANNEL NO.** switch to the channel number on which the selected disk drive is connected. The number assigned to this channel is 3 for the 90/25 system. On the 90/30 B system, the numbers assigned to these channels are:

Channel	Assigned Number
Integrated disk adapter	3
Selector channel 1	4
Selector channel 2	6

g. Set the **DATA ENTRY SUBCHANNEL NO.** switch to the subchannel number assigned to the selected disk drive unit. The subchannel assignment is 0 for the 90/25 system. On the 90/30 B system, the numbers assigned to these channels are:

Channel	Assigned Number
Integrated disk adapter	0
Selector channel 1	0–7
Selector channel 2	0–7

h. Set the **DATA ENTRY DEVICE NO.** switch to the actual physical unit number assigned to the selected disk drive unit (0 through F).

i. Press the top portion of the **SYSTEM RESET** switch. The **TEST MODE** indicator lights.

j. Press the top portion of the **RUN** switch. The **INITIAL LOAD** and **RUN** indicators light. After the initial program load routine is loaded into the system, the **INITIAL LOAD** indicator goes out, and only the **POWER ON**, **TEST MODE**, and **RUN** indicators remain lit.

If either the **PROC CHECK** or **HPR STOP** indicators light, the IPL operation was unsuccessful and steps 2i and 2j must be repeated until the proper indications are obtained. If the **PROC CHECK** indicator continues to light, refer the problem to the Sperry Univac customer engineer. If the **HPR STOP** indicator continues to light, perform the following procedure to determine the cause of the error:

(1) Set the **DISPLAY SELECT 1** switch to **UPPER** position.

(2) Set the **LEGEND SELECT 1** switch to position 7.

(3) Read the first four sets of the **DISPLAY 1** indicators (I0 through I15). HPR codes displayed by these indicators are listed and described in the OS/3 system messages programmer reference, UP-8076 (current version).

If possible, correct the cause of the **HPR STOP**, and then repeat steps 2i and 2j. If the HPR code identified a condition that cannot be corrected by the operator or if the **HPR STOP** persists, contact your Sperry Univac customer engineer.

After this phase of the IPL operation is successfully completed, proceed with step 2k.

NOTE:

The system automatically sets all devices or subsystems not online (not turned on) during IPL time to not available. The devices or subsystems are not available for system use until they are identified as available by the operator via the SET IO command (10.4.7) or until a disk pack is mounted on them and they are initialized.

k. Set the **INITIAL LOAD CONTROL** switch to the **OFF** position. The **TEST MODE** indicator extinguishes and the following message appears on the system console screen:

IPL TO LOAD STANDARD SUPERVISOR UNLESS NEW NAME KEYED IN ¬_ _ _ _ _ _,{L C D}

NOTE:

At this point, the cursor (¬) is at the point of entry for the type-in of the desired supervisor name (six characters). The final character after the comma is for special types of loading as follows:

L = Special load for stand alone programs

C = Supervisor control storage card read

D = Supervisor debug option (for details, see supervisor user guide, UP-8075 (current version))

l. If the standard supervisor is to be loaded into the system, press the **TRANSMIT** key on the system console. No keyin is required. If a different supervisor is to be loaded in, key in the name of the new supervisor and then press the **TRANSMIT** key. The following statements are now displayed on the system console. Respond to each statement as directed. After all statements are answered, again press the **TRANSMIT** key.

NOTE:

The cursor (¬) is initially positioned at the point of entry for the date (second statement). Upon completion of a keyin, the cursor is automatically positioned on the following line at the point where keyin is to begin. Lines not requiring answers may be bypassed by pressing the **RETURN** *key on the system console, thus selecting the displayed default value.*

The **MESSAGE WAITING** *key need not be pressed before initiating any keyin during this procedure.*

OS/3 VERSION nn
Indicates the release version (nn) of the OS/3 system; no reply is required.

DATE? ({YY/MM/DD MM/DD/YY DD/MM/YY}) _ _/_ _/_ _
Requires the date to be entered in the format displayed.

where:

DD
Specifies the day.

MM
Specifies the month.

YY
Specifies the year.

The date format displayed in the message is selected at SYSGEN time for operator convenience. The format of the date, as used by the system for all processing operations and output messages, is YY/MM/DD.

The date entered is compared to the date keyed in at the last load from the same SYSRES. If the date entered is six days less than or six days greater than the date of the last load, the message DATE QUESTIONABLE appears. If the date entered is correct, press the **TRANSMIT** key and the date is accepted. If the date is incorrect, key in the correct date.

TIME? (HH/MM/SS) _ _:_ _:_ _
Requires the time of day to be entered in hours (HH), minutes (MM), and seconds (SS) in 2-digit format.

RUN LIBS DEVICE ADDR? (DEFAULT=system-generation-option) _ _ _
Questions whether the system job run library file, YRUN, is to be located on the disk volume specified during SYSGEN and identified as the default value or on another disk volume. To locate the run library on the default volume, no keyin is required; press the RETURN key on the system console to position the cursor on the next line. To locate YRUN on a different volume, key in the device address of the disk unit containing the desired disk volume.

Remember, the volume identified as the YRUN volume must be online for the system to be operational. Should the specified volume not be online, a system message to mount the volume will appear on the system console.

RECOVER FILES?
Requests operator action on the following three statements:

JOB QUEUE (N, Y, H DEFAULT=N)
To retain jobs previously filed in the job queue for processing, key in Y; to place them into hold status, key in H; to delete them from the queue, press the **RETURN** key to advance to the next statement.

ERROR LOG (N, Y DEFAULT=Y)
This message appears only if the error log option is configured in your system. To clear all accumulated errors in the error log and start a new error log file, key in N. Otherwise, to retain the present error log file and continue to list errors in the error log, press the **RETURN** key to advance to the next line.

SPOOL FILES (N, A, C, L, H DEFAULT=SYSGEN option)
This message appears only if the spooling option is configured in your system. It requests that you specify the level of recovery desired for the spool file in your system. To specify the level of recovery that was specified during SYSGEN, press the RETURN key to advance to the next line. To specify a different level of recovery, key in one of the following responses:

N

Previously spooled input and output was processed before the system was turned off and the spool file is empty; therefore, no recovery is required.

A

Recover all spooled subfiles when the spool file is reinitialized because previously spooled input or output files in the spool file are to be processed. With this response, all spooled subfiles, whether complete or incomplete, are saved. This response must be specified to recover the console and workstation log file (if configured in the system). Console and workstation messages that were not copied from the main storage buffer to the spool file are not recovered; copying is done only when the buffer has been filled.

C

Recover only completed subfiles when the spool file is initialized.

L

Recover only the user log directory when the spool file is initialized.

H

May be selected only if the operator has taken a system dump of the previously loaded system (see 2.1.3). When the spool file is reinitialized, all spool subfiles (same as with A option) as well as system console messages that have been accumulated in the main storage buffer but not copied onto the spool file are recovered; copying is done only when the buffer has been filled. (Workstation messages that accumulated in the main storage buffer are not recovered, however.) Use this method of recovery (sometimes referred to as the hot start) only if a system crash occurs. When there is a planned shutdown of the system and spool files are to be recovered at a later date, the operator should breakpoint the console log and then select the A, C, or L spool recovery option when the system is reinitialized.

SPOOLING DVC ADDR? (DEFAULT={blank / vsn / SYSRES}) - - -

This message appears only if the spooling option is configured in your system. Further, if multivolume spooling was configured, it appears once for each volume that the spool file can be on, as specified at SYSGEN time. Each of these messages requests that you identify the disk volumes that are to contain the spool file. To use the volume specified during SYSGEN, identified in the message as the default value, press the space bar three times. To locate the spool volume being referenced on a different volume from that displayed in the message, key in the device address of the disk unit containing the desired disk volume. Repeat this procedure for each message displayed.

If no default volume is identified (DEFAULT=blank), you have the option of identifying another volume for use by the spool file or limiting the spool file to those volumes already identified. To identify another volume, key in the device address of the disk unit containing the desired disk volume. To indicate that no more volumes are to be used, press the RETURN key, as required, to position cursor to point of entry for the next statement. Remember, all volumes identified as spool volumes at IPL time must be online for the system to be operational.

MODIFY SUPERVISOR? (N, Y DEFAULT=N)

If no modification to the supervisor is required (the SYSGEN selections for the supervisor are to be used), press the TRANSMIT key on the system console. This action causes the message presented in step m to be displayed.

If the supervisor is to be modified, key in Y. The screen clears and the system output messages are displayed. Proceed with the appropriate operator action described in Appendix c.

NUMBER OF 32K BYTES BLOCKS FOR CACHE (0-4: 0=NO CACHE, DEF=3) #

This message appears only if the disk cache feature is configured in your system. Reply with a number from 0 to 4:

0

Tells the system to run without the cache feature.

1, 2, 3, or **4**

Tells the system to set up a cache of 32K, 65K, 98K, or 131K bytes, respectively.

If your reply does not fall between 0 and 4 or if you do not reply at all, the system defaults to 98K bytes. This cache operates at its present size until the next IPL.

CAUTION

Be sure to respond to all the above statements requiring keyin before pressing the **TRANSMIT** *key. Once the* **TRANSMIT** *key is pressed, the questions and answers are lost and the entire procedure must be restarted if the procedure was not performed correctly.*

m. After the requested information has been responded to as required, press the **TRANSMIT** key on the system console. When the selected supervisor is loaded and initialized, the following header message appears on the system console screen:

(1) (2) (3) (4) (5) (6) (7)
90/nn OS/3 version-no supnam COS-n yy/mm/dd hh:mm:ss

Message Description:

(1) through **(7)**
These seven numbers represent the numbers assigned to the seven jobs that can concurrently run in the system. When a job is initiated, the number disappears and the job name takes its place. The position of the job name signifies its job number.

Example:

If jobs named A, B, and C are running in the system, the screen format is:

A B C (4) (5) (6) (7)

NOTE:

On the 90/30 B system, the first IPL of a supervisor that is configured for spooling with the spool file being located on a selector channel disk subsystem (8411, 8414, 8430, or 8433) requires that the spool file be formatted. This formatting operation takes an appreciable amount of time (approximately 1 minute on an 8430 for the default specification of 50 cylinders) and is no cause for concern.

nn
Signifies your machine type (25 or 30).

version-no
Signifies the OS/3 software version loaded into the system.

supnam
Signifies the name of the supervisor loaded into the system.

n
Signifies the amount of control storage loaded into the system (1, 2, or 3K).

yy/mm/dd
Signifies the year, month, and day used by the system.

hh:mm:ss
Signifies the time in hours, minutes, and seconds used by the system.

n. If Y or H was selected for the JOB QUEUE message (refer to step l), the following message appears on the system console screen:

JOB QUEUE RECOVERED - n JOBS QUEUED

where:

n JOBS QUEUED
Specifies the number of jobs in the scheduling queues.

o. If N was selected for the ERROR LOG message (refer to step l), the following message appears on the system console screen:

ERROR LOG NOT RECOVERED

When one job terminates, the next job to be run takes the place and job number of the terminated job on the system console screen heading.

If either the **PROC CHECK** or **HPR STOP** indicators (on the operator/maintenance panel) light during these phases (m, n, o) of the IPL operation, the preceding messages won't be displayed because the supervisor wasn't loaded or initialized properly, and you must repeat the IPL operation, beginning with step 2e of this procedure.

If, after repeating this procedure, the **PROC CHECK** indicator remains lit, reload control storage and then repeat the IPL operation. If the **HPR STOP** indicator continues to light, perform the following procedure at the operator/maintenance panel to determine the cause of the error:

(1) Set the **DISPLAY SELECT 1** switch to **UPPER**.

(2) Set the **LEGEND SELECT 1** switch to position 7.

(3) Read the first four sets of the **DISPLAY 1** indicators (I0 through I15). The possible HPR codes that can be displayed by these indicators during this phase of the IPL operation are listed and described in the OS/3 system messages manual, UP-8076 (current version). If one of the special supervisor initialization HPR codes is displayed, set the **LEGEND SELECT 1** switch to position 6 and read the **DISPLAY 1** indicators representing registers Z00 through Z15 to further identify the cause of the HPR stop. The first two hexadecimal digits displayed (Z00 through Z07) identify the HPR stop code (01 through 0F). The last two hexadecimal digits further identify the cause of the HPR stop. The meaning of the last two digits is also described in the OS/3 system messages manual, UP-8076 (current version).

If possible, correct the cause of the HPR stop and then retry the IPL operation. If the IPL operation still cannot be completed successfully or if the HPR code identified a condition that cannot be corrected, reload control storage and then retry the IPL operation.

If either the **PROC CHECK** or **HPR STOP** error condition persists, contact your Sperry Univac customer engineer.

After the header is displayed on the system console, the system is ready to process user jobs. Note, however, that any integrated disk units (8415's, 8416's, and 8418's) that were offline when the supervisor was initialized are not available for system use until they are identified as available by the operator via the SET IO command (10.4.7) or until a disk pack is mounted on them and they are initialized.

Note also that the supervisor cannot distinguish between a low-density 8418 disk unit and an 8416 disk unit. Thus, if a low-density 8418 is placed online after system initialization, the following SET IO command must be keyed in to identify this fact to the supervisor:

SET IO,did,TY,2002

Jobs requiring more than the available devices will be terminated with an R277 message. The operator may review device ready status with the MIX command (10.4.3) and change the status with the SET IO command.

```
     PROGRAM1    (2) _ _ _ _ _ _ (3) _ _ _ _ _ _ PROGRAM4 PROGRAM5    (6) _ _ _ _ _ _ PROGRAM7
1    ▷ 71 THIS IS A COMMENT FROM PROGRAM7
2    ▷ 12? ANSWER A QUESTION FROM PROGRAM1?
3    ▷ 12 THIS IS THE ANSWER TO PROGRAM1
4    ▷ 50 ACTIVATE USER ISLAND CODE FOR PROGRAM5
5    ▷ 43* MOUNT DEV=440 VSN=DSP614 LU=050 DEV=441 VSN=DSP633 LU=051
6    ▷ 44* MOUNT DEV=442 VSN=DSP554 LU=052 GO?
7    ▷ GO PROGRAM4
8    ▷ DISPLAY 140,7
9    ▷ 0A? THIS IS A QUESTION FROM THE 'DISPLAY' SYMBIONT
10   ▷ 0A THIS IS THE ANSWER TO THE 'DISPLAY' SYMBIONT'S QUESTION
11   ▷ 00 IO ACTIVATE IO SYMBIONT ISLAND CODE
12   ▷ CANCEL IO,S
13   ▷ DUMP PROGRAM5
14   ▷ 0B THIS IS A COMMENT FROM THE DUMP ROUTINE
15   ▷ END DUMP,PROGRAM5
```

Figure 10—1. Typical System Console Messages

10.2.2.3. Unsolicited Input Messages

Unsolicited messages are those messages input by the operator that are not in direct response to an output message that requires a reply. Unsolicited messages may be entered from the console or workstation that initiated the job or symbiont. The format for all unsolicited input messages is:

△j0△[symbiont-name][(did)]△message-text

where:

j
: Is the job number of the job you want to receive the unsolicited message.

0
: Is the message number used to identify the message as an unsolicited message.

symbiont-name
: Is the 2-character alphanumeric name of the supervisor symbiont to receive the unsolicited message (the job and messge numbers are 00). If a symbiont is not the recipient of the message, no symbiont name is required.

did
: Is the address of the device used or controlled by a specific copy of a symbiont in memory. The symbiont having this device allocation receives the unsolicited message. If the specified device is not assigned to the symbiont, the unsolicited message is not acknowledged. The did must be enclosed in parentheses.

message-text
: Is the actual text of the message.

NOTE:

When you key in an unsolicited message to a symbiont, the system task control blocks (TCB) are searched to locate an active symbiont identified by the name specified in the message. When no address is specified, the unsolicited message is transferred to the buffer of the first symbiont encountered in the switch list that bears this name. If there is more than one copy of the same symbiont active, only the first will get the message. Figure 10—1, line 11, is an example of an unsolicited type-in to a symbiont.

The remainder of this section describes the operator procedures for entering commands, unsolicited messages, and solicited messages, according to the function required. Specific command and message formats are included, specifying what parameters are appropriate and in what order for that particular function.

10.3. JOB PROCESSING COMMANDS

Job processing commands enable the operator to:

- Read job control streams into the system and assign scheduling priorities to them (job initialization)
- Control jobs awaiting execution within the scheduling priority queues (schedule jobs)
- Control jobs being executed (execute jobs)
- Stop jobs under execution (terminate jobs)

In the job processing commands that follow, scheduling priorities are defined as preemptive, high, or normal to specify in what order jobs begin execution. Jobs to be run are placed in one of the three scheduling priority queues:

- PRE (preemptive) Queue

  Contains jobs to be executed first, i.e., before any jobs assigned HIGH or NOR scheduling priority. If rollin/rollout is configured, a PRE job initiated for execution when sufficient main storage is not available may cause HIGH or NOR jobs being processed to be rolled out to make main storage space available for the PRE job. Rolled out jobs are rolled in and continue processing when main storage is again available.

- HIGH Queue

  Contains jobs to be executed before any jobs are assigned a NOR scheduling priority. HIGH scheduling priority jobs are not executed unless the PRE queue is either empty or placed on hold.

- NOR (normal) Queue

  Contains jobs to be executed only when there are no jobs left in the PRE or HIGH queues or when those queues are placed on hold. NOR scheduling priority is the default for a job control stream and for some of the job scheduling commands (10.3.2).

10.3.1. Job Initialization

Job control streams are read into the system by using one of the job initialization commands (FILE, RUN/RV, SI/SC, and OCL/OV). These commands enable you to file the job for future use or to process the job immediately.

When a job is filed for future use, it is placed in the job control stream library (YJCS) file or in an alternate library file, as specified in the operator command.

A job may be initiated for processing from one of three places:

1. from an input device (card reader, diskette, or spool file);
2. from YJCS or an alternate library where the job is filed; or
3. from the YSAVE file where it has been saved in its expanded *run* state via a statement included in the job control stream (// OPTION SAVE or // OPTION NOSCHED statement).

When a job is initiated, it is placed in a scheduling priority queue to await execution. This scheduling priority queue can be specified by the programmer submitting the job in the job control stream itself. The operator can override this specification by entering another scheduling priority in the job initialization command. The default is to use the normal priority queue. ←

10.3.1.1. Filing Job Control Streams (FILE)

Function:

The FILE command files jobs and JPROCs, read from an input device, into the permanent JCS library file (YJCS) or an alternate library file. (The alternate file may be a MIRAM or a SAT file, but keep in mind that the RUN processor cannot access data in a MIRAM file.) The input device can be a card reader, a diskette drive, or the input spool file. ←

If no device and label are identified, the first available card reader, as defined when the system was generated, is expected to contain the job control streams and/or JPROCs to be filed. If the job control stream is on a diskette, the label is required; if it is in the input spool file, RDR and label are required. (See 10.2.1.)

Jobs filed from the card reader must terminate with a //FIN job control statement. When jobs are filed from a diskette or the spool file, the // FIN job control statement is not needed. Jobs input from diskette to the spool file must be single volume.

The FILE command cannot be issued from an enter stream. ←

Format:

Positional Parameter 1:

Identifies an alternate library file where job control streams and/or JPROCs are to be filed. Omit this parameter when no alternate file is required; the job is filed into YJCS.

:alt-filename

Specifies the name of the alternate library file, residing on SYSRES, to receive the job and/or JPROC. If the alternate file name is cataloged, the vsn of that file in the catalog is used. There must not be a write password for the alternate file in the catalog.

:(alt-filename,{RES / RUN / vsn})

Specifies the name of the alternate library file to receive the job and/or JPROC and identifies a volume serial number (RES, RUN, or vsn) for the file. You specify RES to identify SYSRES as the volume to contain the file, RUN to identify the system RUN pack as the volume to contain the file, or you may specify the volume serial number (vsn) of a disk pack to be used. If a file with the same file name is in the catalog, the volume serial number you include in the command makes the distinction between the files and overrides the catalog vsn. There must not be a write password for the alternate file in the catalog. The colon and enclosing parentheses are optional and included only for consistency with the format of the RUN/RV command.

:(alt-filename,[{RES / RUN / vsn}],write-password)

Specifies the name of the alternate library file to receive the job and/or JPROC and includes the write password, identified in the catalog, required to write to that file. You specify RES to identify SYSRES as the volume to contain the file, RUN to identify the system RUN pack as the volume to contain the file, or you may specify the volume serial number (vsn) of a disk pack to be used. If you omit a volume serial number (RES, RUN, or vsn), your file is written to the volume associated with that file name in the catalog. You must specify RES, RUN, or a vsn if you want the file written to a different volume; the volume serial number you specify overrides the catalog vsn. The colon and enclosing parentheses are optional and included only for consistency with the format of the RUN/RV command.

Example:

Operator keyin:

FI :(ALTJCS,RUN)

Function requested:

The job control stream, residing on a card reader, is to be written to the alternate job control library file called ALTJCS on the system RUN pack.

10.3.1.2. Running Job Control Streams (RUN/RV)

Function:

The RUN/RV commands initiate the reading of a job control stream from either an input device, or the YJCS or alternate job control library file. The input device can be a card reader, a diskette drive, or the input spool file.

The commands cause the job control stream to be written to the YRUN library file and expanded to its *run* state (JPROCs are expanded), then scheduled for execution. When an // OPTION SAVE job control statement is included in the job control stream, the job is scheduled to be run from YRUN and a copy of the expanded job is saved in the YSAVE file. When an // OPTION NOSCHED job control statement is included in the job control stream, a copy of the expanded job is saved in YSAVE; however, the job is not scheduled to be run.

The expanded job to be saved in YSAVE or an alternate save file can be run using the SI/SC commands. (SC/SI commands are described in 10.3.1.3.)

The RV command initiates the reading of a prefiled job control stream that does not contain a // CR statement indicating that there is input (cards, diskette, or spool file) to be read and inserted into the stream. The RUN command initiates the reading of a job control stream that requires the use of an input device (i.e., card reader, diskette, or spool file). This means if the job is initiated from an input device or if the job contains a // CR statement to read input, you must use RUN. When the RUN command is issued, it is accepted only if an input device is available, whether or not one is needed by the job control stream being read. The RV command allows a job control stream to be initiated that does not require the use of an input device. You must include a job name when you enter an RV command.

The operator should remember that when a system card reader is placed online, the RUN command to read cards in the hopper is initiated when the RUN switch on the card reader is pressed, or when the RUN command is keyed in at the system console. The RUN command should be initiated from either location, but not from both. If a duplicate RUN command is initiated for the same job, the supervisor queues the second command until the input device is available. Presuming that nothing is in the hopper when the second RUN command is executed, a hopper empty message results.

Format:

```
{ RUN [ [ (did)              ] ]  [ { jobname[(new-name)] } ] }
{     [ { ([did],label)      } ]  [ { (new-name)          } ] }
{     [ { (RDR,label)        } ]                              }
{ RV  jobname[(new-name)]                                     }

  [ { :alt-filename                                                  } ]
  [ { :(alt-filename,{RES})                                          } ]
  [ {               {RUN}                                            } ]
  [ {               {vsn}                                            } ]
  [ { :(alt-filename,[{RES}],read-password)                          } ]
  [ {               [{RUN}]                                          } ]
  [ {               [{vsn}]                                          } ]

  [ ,{PRE } ] [,key-1=val-1,...,key-n=val-n]
  [  {HIGH} ]
  [  {NOR } ]
```

Command Code:

RUN

Initiates the running of a job control stream that requires an input device. You must specify a job name when the job control stream is prefiled. If it is prefiled and you don't specify an input device, the first available card reader is assigned to the job. If you omit the input device and job name, the first available card reader, as defined when the system was generated, is expected to contain the job control stream to be run.

You must specify a label if the job control stream is on a diskette. If it is in the input spool file, you must specify RDR and a label (10.2.1). For diskette and spool file input, the last // FIN job control statement is not needed because it is only used to terminate card reader operation. However, the // FIN statements that separate groups of card images read with // CR statements are still necessary. Jobs input from diskette to the spool file must be single volume.

RV

Initiates the running of a prefiled job control stream that does not require an input device; that is, does not contain a // CR (read card reader, diskette, or spool file) statement. You must specify a job name.

Positional Parameter 1:

jobname[(new-name)]

Identifies the name of the job to be read from YJCS or an alternate job control library file and stored in a scheduling priority queue to await execution. The job name consists of one to eight alphanumeric characters. The job name is required with RV.

You include *new-name* to assign a new 1- to 8-character alphanumeric name to a job already stored in YJCS or an alternate job control library file. The job identified by the *jobname* parameter is read from YJCS or an alternate file, and stored in a scheduling priority queue under the name identified by the *new-name* parameter to await execution. The new name cannot contain blanks.

(new-name)

Used with the RUN command to assign a new 1- to 8-character alphanumeric name to a job input from the card reader. The job is read and stored in a scheduling priority queue under the new name to await execution. The new name cannot contain blanks.

If omitted from the RUN command, the job is read and stored in a queue under the jobname on the // JOB statement in the job control stream.

Positional Parameter 2:

Used when the job control stream resides in an alternate job control library file, rather than in YJCS. When the job resides in an alternate library file, this parameter identifies the library file to be read. If omitted, the job is read from YJCS.

:alt-filename

Specifies the name of the alternate library file, residing on SYSRES, that contains the job. If the alternate file name is cataloged, the vsn of that file in the catalog is used. There must not be a read password for the alternate file in the catalog.

:(alt-filename,{RES / RUN / vsn})

Specifies the name of the alternate library file that contains the job or JPROC and identifies a volume serial number (RES, RUN, or vsn) for the file. You specify RES to identify SYSRES as the volume that contains the file, RUN to identify the system RUN pack as the volume that contains the file, or you may specify the volume serial number (vsn) of a disk pack to be read. If a file with the same file name is in the catalog, the volume serial number you include in the command makes the distinction between the files and overrides the catalog vsn. There must not be a read password for the alternate file in the catalog.

:(alt-filename,[{RES / RUN / vsn}],read-password)

Specifies the name of the alternate library file that contains the job stream and includes the read password, identified in the catalog, required to read from that file. You specify RES to identify SYSRES as the volume that contains the file, RUN to identify the system RUN pack as the volume that contains the file, or you may specify the volume serial number (vsn) of a disk pack to be read. If you omit a volume serial number (RES, RUN, or vsn), your file is read from the volume associated with that file name in the catalog. You must specify RES, RUN, or a vsn if you want the file read from a different volume; the volume serial number you specify overrides the catalog vsn.

Positional Parameter 3:

PRE
Places the job in the preemptive scheduling priority queue to await execution.

HIGH
Places the job in the high scheduling priority queue to await execution.

NOR
Places the job in the normal scheduling priority queue to await execution.

If positional parameter 3 is omitted, the scheduling priority specified in the job control stream is used. If not specified in the job stream, the normal priority is used.

Positional Parameters 4 through n:

key-1=val-1,...,key-n=val-n
Are the keywords and their values, which may be used by the job being run. The keywords and their values must be supplied by the user requesting the job.

NOTE:

The total length of all the parameters specified in this command, from the first character of positional parameter 1 to the last character of the last keyword value specified, is limited to 60 characters.

Examples:

1. Operator keyin:

   RU MYJOB:(ALTJCS,RUN)

   Function requested:

   The job named MYJOB, filed in the alternate job control library file ALTJCS on the system RUN pack, is to be run under the priority specified in the job control stream. The first available card reader is expected to contain some input for MYJOB, which contains a // CR statement.

2. Operator keyin:

   RV MYJOBA(NETPAY)

   Function requested:

   The job named MYJOBA, filed in YJCS, is to be run under the new name NETPAY according to the priority specified in the job control stream.

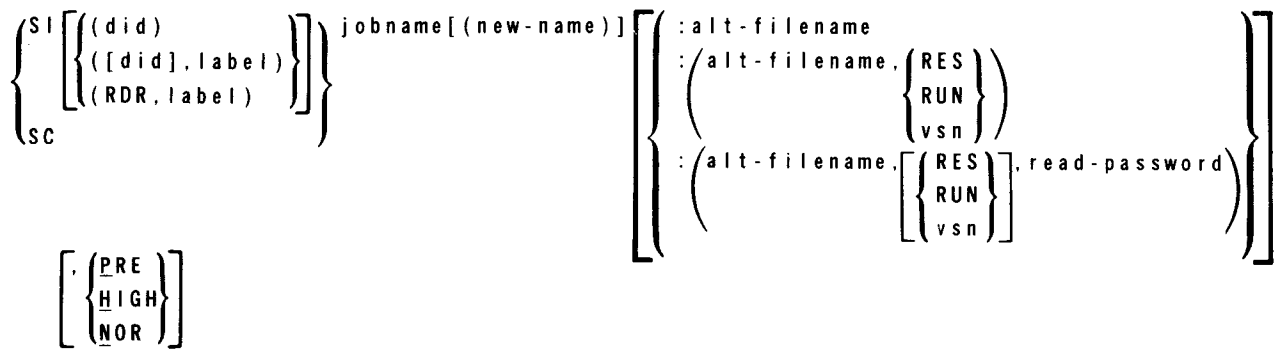
10.3.1.3. Running Saved Job Control Streams (SI/SC)

Function:

The SI/SC commands initiate the running of a job control stream from the YSAVE MIRAM library file or from an alternate library file, then schedule the job for execution. In either file, the control stream has been saved in its expanded *run* state. A control stream is expanded in the YRUN file when the RUN/RV or OCL/OV command is issued for the job. When an // OPTION SAVE or // OPTION NOSCHED job control statement is included in the job control stream, a copy of the expanded control stream is stored in the YSAVE or alternate file for subsequent runs using SI or SC.

The SC command is used only to initiate the reading of a job control stream that does not require an input device to replace embedded data. The SI command initiates the reading of a job control stream that requires an input device (i.e., card reader, diskette, or spool file) to replace embedded data. When the SI command is issued, it is accepted only if an input device is available. The SC command allows a job control stream to be initiated that does not require an input device.

Format:



Command Code:

SI

Initiates the running of a job control stream that requires an input device. If no device and label are identified, the first available card reader, as defined when the system was generated, is expected to contain the replacement data required by the job. If the data is on a diskette, the label is required; if it is in the input spool file, RDR and label are required. (See 10.2.1.) For diskette and spool file input, the last // FIN job control statement is not necessary.

SC

Initiates the running of a job control stream that does not require an input device to replace embedded data.

Positional Parameter 1:

jobname [(new-name)]

Identifies the name of the job to be read from YSAVE or an alternate library file and stored in a scheduling priority queue to await execution. The job name consists of one to eight alphanumeric characters.

You include *new-name* to assign a new 1- to 8-character alphanumeric name to a job stored in YSAVE or an alternate file. The job identified by the *jobname* parameter is read from YSAVE or an alternate file and stored in a scheduling priority queue under the name identified by the *new-name* parameter to await execution. The new name cannot contain blanks.

Positional Parameter 2:

Used when the job control stream resides in an alternate job control library file on disk or format label diskette, rather than in YSAVE. When the job resides in an alternate library file, this parameter identifies the library file to be read. If omitted, the job is read from YSAVE.

:alt-filename

Specifies the name of the alternate library file, residing on SYSRES, that contains the job. if the alternate file name is cataloged, the vsn of that file in the catalog is used. There must not be a read password for the alternate file in the catalog.

:(alt-filename, {RES / RUN / vsn})

Specifies the name of the alternate library file that contains the job or JPROC and identifies a volume serial number (RES, RUN, or vsn) for the file. You specify RES to identify SYSRES as the volume that contains the file; RUN to identify the system RUN pack as the volume that contains the file; or you may specify the volume serial number (vsn) of a disk pack or format label diskette to be read. If a file with the same file name is in the catalog, the volume serial number you include in the command makes the distinction between the files and overrides the catalog vsn. There must not be a read password for the alternate file in the catalog.

$$:\left(\text{alt-filename},\left[\left\{\begin{matrix}\text{RES}\\\text{RUN}\\\text{vsn}\end{matrix}\right\}\right],\text{read-password}\right)$$

Specifies the name of the alternate library file that contains the job stream and includes the read password, identified in the catalog, required to read from that file. You specify RES to identify SYSRES as the volume that contains the file; RUN to identify the system RUN pack as the volume that contains the file; or you may specify the volume serial number (vsn) of a disk pack or format label diskette to be read. If you omit a volume serial number (RES, RUN, or vsn), your file is read from the volume associated with that file name in the catalog. You must specify RES, RUN, or a vsn if you want the file read from a different volume; the volume serial number you specify overrides the catalog vsn.

Positional Parameter 3:

PRE
Places the job in the preemptive scheduling priority queue to await execution.

HIGH
Places the job in the high scheduling priority queue to await execution.

NOR
Places the job in the normal scheduling priority queue to await execution.

If positional parameter 2 is omitted, the scheduling priority assigned to the job via the job control stream is in effect.

Example:

Operator keyin:

SI MYJOB(AVGYTD),H

Function requested:

The job called MYJOB, expanded and filed in YSAVE, is to be run under the new name AVGYTD. The job is to be placed in the high scheduling priority queue to await execution. Replacement embedded data for AVGYTD is expected to be found on the first available card reader.

10.3.1.4. Running IBM System/3 Operation Command Language Jobs (OCL/OV)

Function:

The OCL/OV commands enable the operator to run an IBM System/3 control stream in an OS/3 environment. When the OCL or OV command is entered, the entire System/3 control stream is read and interpreted by the OCL processor. Once the OCL processor verifies that no syntax or sequence errors exist, the job is placed in a scheduling priority queue.

The OV command initiates the reading of a prefiled control stream that does not contain a // CR statement indicating that there is input (cards, diskette, or spool file) to be read and inserted into the stream. The OCL command initiates the reading of a control stream that requires an input device (i.e., card reader, diskette, or spool file). When the OCL command is issued, it is accepted only if an input device is available, whether or not one is needed by the control stream being read. The OV command allows a control stream to be initiated that does not require an input device. You must include a job name when you enter an OV command.

Format:

```
{OCL[[{(did)            }]   [(new-name)]  ][,{PRE }][,key-1=val-1,...,key-n=val-n]
 {   [{([did],label)}]                      }[,{HIGH}]
 {   [{(RDR,label)  }]                      }[,{NOR }]
 {OV  jobname-library-unit[(new-name)]      }
```

Command Code:

OCL

Initiates the running of a control stream that requires an input device. If you omit the input device, the first available card reader, as defined when the system was generated, is expected to contain the control stream to be run.

You must specify a label if the control stream is on a diskette. If it is in the input spool file, you must specify RDR and a label (10.2.1). For diskette and spool file input, the last // FIN job control statement is not needed because it is used only to terminate card reader operation. However, the // FIN statements that separate groups of card images read with // CR statements are still necessary. Jobs input from diskette to the spool file must be single volume.

OV

Initiates the running of a prefiled control stream that does not require an input device; that is, does not contain a // CR (read card reader, diskette, or spool file) statement. You must specify a jobname-library-unit.

Positional Parameter 1:

(new-name)

Used with the OCL command to assign a new 1- to 8-character alphanumeric name to the job; otherwise, the job name is taken from the // JOB statement or is defined as OCLnnnn by default, where nnnn is a decimal number from 0001 to 9999. The new name cannot contain blanks.

jobname-library-unit[(new-name)]

Identifies the name of the job to be read from the library specified by the library-unit code. The job name must be appended with a dash (11-punch), then the library-unit code. Jobname-library-unit is required with OV.

You include *new-name* to assign new 1- to 8-character alphanumeric name to a job that is stored in the library specified by the library-unit code. The job identified by the *jobname* parameter is read from the library-unit specified, and stored in a scheduling priority queue under the name identified by the *new-name* parameter to await execution. The job name and library-unit must be separated by a dash (11-punch). The new name cannot contain blanks.

If omitted from the OCL command, the job name is taken from the // JOB statement or is defined as OCLnnnn by default.

Positional Parameter 2:

PRE
Places the job in the preemptive scheduling priority queue to await execution.

HIGH
Places the job in the high scheduling priority queue to await execution.

NOR
Places the job in the normal scheduling priority queue to await execution.

If positional parameter 2 is omitted, the scheduling priority assigned to the job via the control stream is in effect.

Positional Parameters 3 through n:

key-1=val-1,...,key-n=val-n
Are the keywords and their values, which may be used by the job being run. The keywords and their values must be supplied by the user requesting the job.

Examples:

1. Operator keyin:

   OC

   Function requested:

   The control stream is to be run from the first available card reader under the priority assigned in the control stream. The job name is taken from the // JOB statement or will be OCLnnnn by default.

2. Operator keyin:

   OV MYJOB-F1(PAYROLL)

Function requested:

The job called MYJOB, filed in the library F1, is to be run under the new name PAYROLL according to the priority assigned in the control stream.

10.3.2. Job Scheduling

A job is placed in a scheduling priority queue to await the availability of system resources (e.g., main storage, disk drive, printer) to execute that job. While waiting for these resources, the operator can exercise control over any specific job in a queue, all jobs in a specific queue, and all jobs in all queues, by using job scheduling commands. The operator may also control, in a similar manner, jobs initiated by a specific workstation user. These commands allow you to:

- Defer jobs from being executed
- Permit jobs to be executed
- Delete jobs from a queue

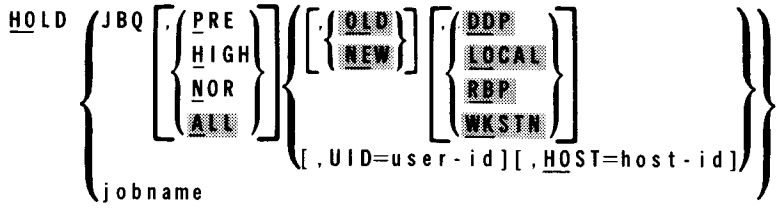
- Display contents of a queue
- Change a job's scheduling priority

10.3.2.1. Deferring Jobs Scheduled for Execution (HOLD)

Function:

The HOLD command permits the operator to defer the scheduling of jobs according to the command parameters specified. You can defer scheduling for all jobs in all queues or in a specific queue; for a specific job within a queue; for a specific workstation user's jobs in all queues or a specific queue; or for a specific host's jobs in all queues or in a specific queue. Scheduling remains deferred until the jobs are reactivated via the BEGIN command.

Format:



Positional Parameter 1:

Specifies that the command applies to the jobs in a job scheduling priority queue, as further defined by PRE, HIGH, NOR, or ALL, where:

PRE
Defers the jobs in the preemptive scheduling priority queue.

HIGH
Defers the jobs in the high scheduling priority queue.

NOR
Defers the jobs in the normal scheduling priority queue.

ALL
Defers the jobs in all scheduling priority queues.

jobname
Defers a particular job from being scheduled for execution. No further parameters are permitted.

Keyword Parameters:

OLD
Defers only jobs already in the scheduling priority queue defined in parameter 1. Jobs subsequently entered in this queue are not deferred.

NEW
Defers only jobs subsequently placed in the scheduling priority queue defined in parameter 1. All existing jobs are still available for execution.

Once a new job is placed in queue, it becomes an old job for any subsequent commands to defer or permit execution. Therefore, the NEW parameter is used with the HOLD command to defer new jobs entering a queue, while the old jobs already residing in the queue remain unchanged and are still able to be scheduled for execution. Likewise, the OLD parameter is used with the HOLD command to defer old jobs already residing in the queue while the new jobs entering the queue are still able to be scheduled for execution. Once a new job enters a queue that is under the influence of a HOLD NEW command, a subsequent command to permit the old jobs in that queue to be scheduled for execution will also release the new job from its deferred status.

DDP
Defers only distributed data processing (DDP) jobs.

LOCAL
Defers only locally entered jobs.

RBP
Defers only jobs entered remotely (i.e., from a remote batch terminal).

WKSTN
Defers only jobs initiated from a workstation.

NOTE:

All of the above keyword parameters may be interchanged.

UID=user-id

Defers all old jobs associated with a particular workstation user-id in the scheduling priority queue defined in parameter 1. Jobs subsequently entered with this user-id are not deferred. The user-id is a 1- to 6-character alphanumeric identification that does not start with Y.

HOST=host-id
Defers all old jobs associated with a particular host in the scheduling priority queue defined in parameter 1. Jobs subsequently entered with this host-id are not deferred. The host-id is one to four alphanumeric characters.

NOTES:

1. *The UID and HOST parameters may be interchanged. No other keyword parameters are permitted.*
2. *If a command is entered from the system console (local or remote), the HOST parameter should not be used; that site's host-id is used automatically.*
3. *The special user-id OPERATOR indicates that the command applies to console-initiated jobs.*
4. *Any command from the system console (local or remote) pertains only to jobs with a matching host-id regardless of user-id.*
5. *DDP users submitting the HOLD command via a parameterized enter stream may use HOST=$HOST to specify that the host-id of the command's submitter be used.*

Examples:

1. Operator keyin:

   HO JBQ,A,NE,WKSTN

   Function requested:

   All jobs to be subsequently initiated from workstations and placed in all scheduling priority queues are to be deferred.

2. Operator keyin:

   HO JBQ,H,UID=WKSTA1

   Function requested:

   All jobs in the high scheduling priority queue that are associated with the workstation user WKSTA1 are to be deferred.

3. Operator keyin:

   HO JBQ,N,HOST=B

   Function requested:

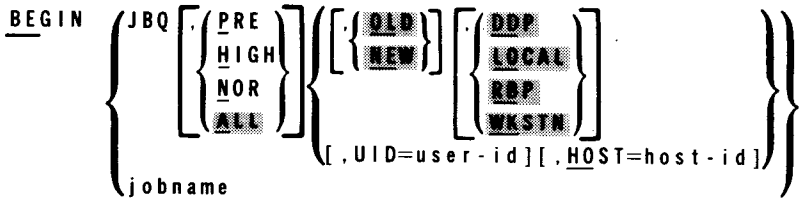
   All jobs in the normal priority queue and associated with host B are to be deferred. New jobs entered into this queue are not to be deferred.

10.3.2.2. Scheduling Deferred Jobs (BEGIN)

Function:

The BEGIN command reinstitutes the scheduling for execution of currently deferred jobs, according to the command parameters specified. Scheduling may be reinstituted for all jobs in all queues or in a specific queue; for a specific job within a queue; for a specific workstation user's jobs in all queues or a specific queue; or for a specific host's jobs in all queues or in a specific queue. Jobs remain deferred by a HOLD command until a BEGIN command is entered to permit their rescheduling for execution.

Format:



Positional Parameter 1:

Specifies the command applies to the jobs in a job scheduling priority queue, as further defined by PRE, HIGH, NOR, or ALL, where:

PRE
Permits the jobs in the preemptive scheduling priority queue to be scheduled for execution.

HIGH
Permits the jobs in the high scheduling priority queue to be scheduled for execution.

NOR
Permits the jobs in the normal scheduling priority queue to be scheduled for execution.

ALL
Permits the jobs in all scheduling priority queues to be scheduled for execution.

jobname
Permits the particular job to be scheduled for execution. No further parameters are permitted.

Keyword Parameters:

OLD
Permits only jobs either initiated from a workstation or jobs using the // OPTION ORI= or // OPTION MASS= job control statement, as defined in parameter 1, to be scheduled for execution.

NEW
Permits newly entered jobs placed in the scheduling priority queue defined in parameter 1 to be permitted to be scheduled for execution.

If omitted, both old and new jobs are permitted to be scheduled for execution. (See 10.3.2.1.)

DDP
Permits only distributed data processing (DDP) jobs to be scheduled for execution.

LOCAL
Permits only locally entered jobs to be scheduled for execution.

RBP
Permits only jobs entered remotely (i.e., from a remote batch terminal) to be scheduled for execution.

WKSTN
Permits only jobs either initiated from a workstation or jobs using the // OPTION ORI= or // OPTION MAS= job control statement to be scheduled for execution.

NOTE:

All of the above keyword parameters may be interchanged.

UID=user-id
Permits all jobs associated with a particular workstation user-id to be scheduled in the scheduling priority queue defined in parameter 1. The user-id is a 1- to 6-character alphanumeric identification that does not start with Y.

HOST=host-id
Permits all jobs associated with a particular host to be scheduled in the priority queue defined in parameter 1. The host-id is one to four alphanumeric characters.

NOTES:

1. *The UID and HOST parameters may be interchanged, but no other parameters are permitted.*
2. *If a command is entered from the system console (local or remote), the HOST parameter should not be used; that site's host-id is used automatically.*
3. *The special user-id OPERATOR indicates that the command applies to console-initiated jobs.*
4. *Any command from the system console (local or remote) pertains only to jobs with a matching host-id regardless of user-id.*

Examples:

1. Operator keyin:

   BE MYJOB

   Function requested:

   The currently deferred job named MYJOB is permitted to be scheduled for execution.

2. Operator keyin:

   BE JBQ,UID=WKSTA2

   Function requested:

   All currently deferred jobs in all scheduling priority queues associated with the workstation user WKSTA2 are permitted to be scheduled for execution.

3. Operator keyin:

   BE JBQ,H,HOST=ABC

   Function requested:

   All currently deferred jobs in the high scheduling priority queue and associated with host ABC are scheduled for execution.

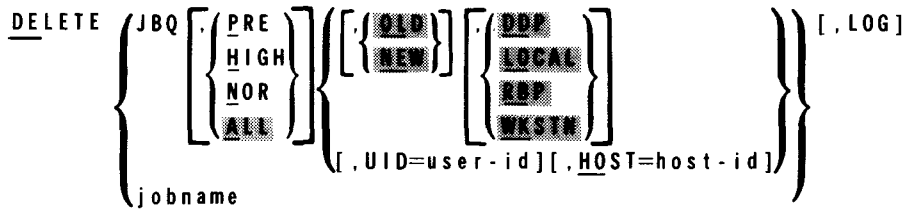
10.3.2.3. Deleting Jobs from Scheduling Priority Queues (DELETE)

Function:

The DELETE command permits the operator to delete jobs according to the command parameters specified. You can delete all jobs in all queues or in a specific queue; a specific job within a queue; a specific workstation user's jobs in all queues or in a specific queue; or a specific host's jobs in all queues or in a specific queue. Only those jobs residing in a scheduling priority queue, and thus waiting to begin execution, can be deleted.

The DELETE command cannot be issued from an enter stream.

Format:



Positional Parameter 1:

JBQ,(PRE / HIGH / NOR / ALL)

Specifies the command applies to the jobs in a job scheduling priority queue, as further defined by PRE, HIGH, NOR, or ALL, where:

PRE
: Deletes the jobs in the preemptive scheduling priority queue.

HIGH
: Deletes the jobs in the high scheduling priority queue.

NOR
: Deletes the jobs in the normal scheduling priority queue.

ALL
: Deletes the jobs in all scheduling priority queues.

jobname
: Deletes a particular job from being scheduled for execution.

Keyword Parameters:

DDP
: Deletes only distributed data processing jobs.

LOCAL
: Deletes only locally entered jobs.

RBP
: Deletes only jobs entered remotely (i.e., from a remote batch terminal).

WKSTN
: Deletes only jobs either initiated from a workstation or jobs using the // OPTION ORI= or // OPTION MAS= job control statement.

NOTE:

All of the above keyword parameters may be interchanged.

UID=user-id
Deletes all old jobs associated with a particular workstation user-id in the scheduling priority queue defined in parameter 1. Jobs subsequently entered with this user-id are not deleted. The user-id is a 1- to 6-character alphanumeric identification that does not start with Y.

HOST=host-id
Deletes all old jobs associated with a particular host in the scheduling priority queue defined in parameter 1. Jobs subsequently entered with this host-id are not deleted. The host-id is one to four alphanumeric characters.

LOG
Specifies the job log is printed for all jobs deleted. If omitted, the log is not printed.

NOTES:

1. *The UID and HOST parameters may be interchanged, but no other parameters are permitted.*
2. *If a command is entered from the system console (local or remote), the HOST parameter should not be used; that site's host-id is used automatically.*
3. *The special user-id OPERATOR indicates that the command applies to console-initiated jobs.*
4. *Any command from the system console (local or remote) pertains only to jobs with a matching host-id regardless of user-id.*

Examples:

1. Operator keyin:

   DE JBQ,H,HOST=CDE

   Function requested:

   All jobs originating from host CDE and residing in the high scheduling priority queue are to be deleted.

2. Operator keyin:

   DE JBQ,A,UID=WKSTA1,LOG

   Function requested:

   All jobs in all scheduling priority queues associated with the workstation user WKSTA1 are to be deleted and their logs are to be printed.

10.3.2.4. Displaying Jobs in Scheduling Priority Queues (DISPLAY)

Function:

The DISPLAY command permits the operator to display the contents of any or all job scheduling queues on the system console screen, as specified by command parameters. You can display all jobs in all queues or in a specific queue; a specific job within a queue; a specific workstation user's jobs in all queues or in a specific queue; or a specific host's jobs in all queues or in a specific queue. All requested jobs within the specified queue are displayed. Jobs in a deferred status (HOLD command) are displayed with parentheses around the job name. When all queues are requested, PRE is displayed first, followed by HIGH, and then NOR. If no jobs are found in the queue you request, a system output message is displayed stating that condition.

A system output message is displayed before the list of jobs. The message specifies:

- whether the request was for LOCAL, RBP, DDP, WKSTN, or for all jobs (QUEUED);
- the user-id and host-id;
- the priority of the queue display to follow; and
- whether a hold local (HL), hold remote (HR), hold workstation (HW), or hold DDP (HD) status is in effect for that queue.

Jobs are displayed in three different formats, depending on the parameters you specify on the DISPLAY command. In all cases, jobs in hold status are displayed in parentheses.

- Two jobs per line

  If the DDP parameter is specified, the host-id is included in the following format:

  ```
  jobname: host-id/user-id      jobname: host-id/user-id
  ```

- Three jobs per line

  This format is the most common and is used when you specify the HOST parameter or the WKSTN parameter or the command is from the system console (local or remote) with no other parameters. The format is:

  ```
  jobname: user-id      jobname: user-id      jobname: user-id
  ```

- Five jobs per line

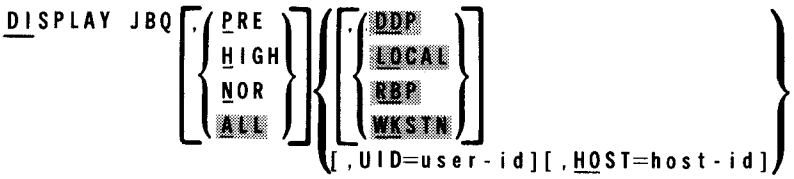
  If there can be no user-id (LOCAL or RBP) or if there is only one user-id to be selected (UID= or DI JBQ from workstation), the jobs are displayed in the following format:

  ```
  jobname    jobname    jobname    jobname    jobname
  ```

  NOTE:

  The DISPLAY command cannot be issued from an enter stream.

Format:



Positional Parameter 1:

Specifies the command applies to the jobs in a job scheduling priority queue, as further defined by PRE, HIGH, NOR, or ALL, where:

PRE
Displays the jobs in the preemptive scheduling priority queue.

HIGH
Displays the jobs in the high scheduling priority queue.

NOR
Displays the jobs in the normal scheduling priority queue.

ALL
Displays the jobs in all scheduling priority queues.

Keyword Parameters:

DDP
Displays only distributed data processing jobs.

LOCAL
Displays only locally entered jobs.

RBP
Displays only jobs entered remotely (i.e., from a remote batch terminal).

WKSTN
Displays only jobs either initiated from a workstation or jobs using the // OPTION ORI= or // OPTION MAS= job control statement.

NOTE:

All of the above keyword parameters may be interchanged.

UID=user-id
Displays all old jobs associated with a particular workstation user-id from the scheduling priority queue defined in parameter 1. Jobs subsequently entered with this user-id are not displayed. The user-id is a 1- to 6-character alphanumeric identification that does not start with Y.

HOST=host-id
Displays all old jobs associated with a particular host and residing in the scheduling priority queue defined in parameter 1. The host-id is one to four alphanumeric characters.

NOTES:

1. *The UID and HOST parameters may be interchanged, but no further parameters are permitted.*
2. *If a command is entered from the system console (local or remote), the HOST parameter should not be used; that site's host-id is used automatically.*
3. *The special user-id OPERATOR indicates that the command applies to console-initiated jobs.*
4. *Any command from the system console (local or remote) pertains only to jobs with a matching host-id regardless of user-id.*

Examples:

1. Operator keyin:

   DI JBQ,N,LO,RBP

   Function requested:

   Displays all locally and remotely entered jobs in the normal scheduling priority queue.

2. Operator keyin:

   DI JBQ,UID=WKSTA2

   Function requested:

   Displays all jobs in all scheduling priority queues that are associated with the workstation user WKSTA2.

10.3.2.5. Changing a Job Scheduling Priority (CHANGE)

Function:

The CHANGE command changes the scheduling priority of a specific job. If you place a deferred job into a new (changed) scheduling priority queue, the job retains its deferred status. Likewise, if you place a job into a queue that is under the influence of a HOLD NEW command, it too will become deferred. The job is put on the end of the new queue; it is the last examined for scheduling for execution in that queue.

Format:

CHANGE jobname, {PRE / HIGH / NOR}

Positional Parameter 1:

jobname
Specifies the particular job to have its scheduling priority changed.

Positional Parameter 2:

PRE
Moves the job defined in positional parameter 1 into the PRE scheduling priority queue.

HIGH
Moves the job defined in positional parameter 1 into the HIGH scheduling priority queue.

NOR
Moves the job defined in positional parameter 1 into the NOR scheduling priority queue.

Example:

Operator keyin:

CH JOBABC,H

Function requested:

Moves JOBABC from a previously assigned (via the job control stream, command entry for running the job, or NOR by default) scheduling priority queue into the HIGH scheduling priority queue. The following message appears when the change has been made:

CH OF JOB JOBABC

10.3.3. Job Execution

When a job is being executed, the operator can control the processing of that job through the use of job execution commands. These commands allow you to:

- Suspend a job under execution
- Restart a job that has been suspended
- Raise or lower the switching priority level of a job being executed

A job is never executing unless the job name is displayed in the top line of the system console screen. If a job is rolled out, an asterisk (*) is displayed next to the job name on this line. Any command or unsolicited message to that job will be rejected; reenter the message or command when the asterisk is no longer displayed.

10.3.3.1. Suspending a Job in Progress (PAUSE)

Function:

The PAUSE command suspends processing of a job. You may enter the command at any time, and job processing suspends immediately. If the job is between job steps, PAUSE takes effect at the beginning of the next job step. The PAUSE command permits you to mount a new volume on a tape unit or disk drive, replace paper on the printer, or place more cards in the card reader. The suspended job is reactivated by the GO command.

Format:

```
PAUSE jobname
```

Positional Parameter:

`jobname`
Specifies the name of the job whose processing is suspended.

10.3.3.2. Activating a Suspended Job (GO)

Function:

The GO command reactivates a job suspended by the PAUSE command or by job control operations. Job control suspends processing of a job when it issues instructions to mount a new volume on a tape unit or disk drive. The GO command also is required as a response to a system message preceded by an asterisk (*).

Format:

```
GO jobname
```

Positional Parameter:

`jobname`
Specifies the job to be reactivated after execution has temporarily suspended.

10.3.3.3. Changing a Job Switching Priority (SWITCH)

Function:

The SWITCH command changes the switching priority level for a job under execution. The switching priority level for the currently executing job step is changed for the duration of the job step. Any subsequent job step executes under the priority established for it (via // EXEC job control statement or default to the lowest level established at SYSGEN) unless changed by another SWITCH command. A job assigned a higher switching priority level has priority over lower switching priority level jobs for control of the central processor.

If a job is changed to a higher switching priority level than another job currently being executed, the lower switching level job will often be processed slower than the higher switching level job. The number of switching priority levels a job can be raised or lowered is governed by the number of switching priority levels established at system generation time (maximum 60 levels).

Switching priority levels are from 1 to n, where 1 is the highest priority level and n is the lowest. If the SWITCH command exceeds the upper or lower limit of these levels, the system automatically changes the number of levels the job can be raised or lowered, so that the job remains within the preset switching priority limits.

When you change any job switching priority, all tasks of that job retain the same switching priority relative to each other; therefore, if a job task exceeds the upper or lower switching priority limit, all the job tasks move only by the number of priority levels that the highest or lowest priority task can be moved within the switching priority limits. For this reason, the system may automatically reduce the number of priority levels that the job switching priority may be changed.

Format:

```
SWITCH jobname,{+number-of-priority-levels}[,ALL]
               {-number-of-priority-levels}
```

Positional Parameter 1:

`jobname`
Specifies the name of the job whose task switching priority is changed.

Positional Parameter 2:

`+number-of-priority-levels`
Specifies the number of switching priority levels a job is raised.

`-number-of-priority-levels`
Specifies the number of switching priority levels a job is lowered.

Positional Parameter 3:

`ALL`
Indicates that the priority to which the job is switched will be in effect for all subsequent job steps. If omitted, the priority is in effect only for the current job step.

10.3.4. Job Termination

Job termination commands permit the operator to terminate the processing of a job or a symbiont or transient as defined by the command parameters.

10.3.4.1. Cancelling a Job in Progress (CANCEL)

Function:

The CANCEL command immediately halts all processing of a job or symbiont. You enter the CANCEL command at any time during job processing to immediately terminate the job step currently being executed, plus any subsequent job steps scheduled for the job. The job run library file for the job also is deleted.

Format:

```
CANCEL { jobname [,{D}] }
       {       [,{N}] }
       { symbiont,S[,N] }
```

Positional Parameter 1:

jobname
: Specifies the name of the job whose processing is immediately terminated and whose job run library file is deleted.

symbiont
: Specifies the 2-character console command that called the symbiont to be terminated.

Positional Parameter 2:

D
: Specifies that a dump is taken when the job terminates, regardless of the dump option specified in its job control stream.

N
: Specifies that no dump is taken when the job terminates, regardless of the dump option specified in its job control stream.

S
: Specifies that the name in positional parameter 1 is the name of a symbiont.

If omitted, the job control dump options remain in effect. Positional parameter 2 must be specified when a symbiont is cancelled.

Positional Parameter 3:

N
: Specifies that no dump is taken when the symbiont terminates.

If omitted, a symbiont dump is taken.

10.3.4.2. Stopping Execution of a Dump (END)

Function:

The END command terminates execution of a cancelor end-of-job dump for a particular job.

Format:

```
END DUMP,jobname
```

Positional Parameter 1:

DUMP
Specifies that the execution of a dump is stopped.

Positional Parameter 2:

jobname
Specifies the name of the job whose cancel or end-of-job dump you want stopped.

10.3.4.3. Terminating a Job (STOP)

Function:

The STOP command terminates a specific job at the end of the currently executing job step. This command provides for orderly termination of the job.

Format:

```
STOP jobname
```

Positional Parameter:

jobname
Specifies the job whose processing is to be terminated in an orderly sequence.

10.4. SELECTED-OCCASION OPERATOR COMMANDS

During the course of processing a job, the operator may be required to enter system-oriented commands to obtain information or make changes not involved with the execution of a particular job. These commands enable you to:

- display an area of main storage;
- display information on active jobs and symbionts, current system I/O device status, and outstanding requests and commands;
- clear the system console screen of all but outstanding output messages;
- change the system time or date;
- control software-detected hardware error logging;
- set an I/O device down or up as required for normal maintenance or device malfunction;
- set the seek separation feature on or off for all devices or a particular device;

- read the volume serial number of a mounted disk or tape volume;
- display the status of jobs in the system;
- dump contents of main storage;
- set the YDUMP file to an unlocked condition; and
- terminate system activity.

The commands enabling you to perform these functions are described in 10.4.1 through 10.4.13. Commands required to control interactive services, system utility functions, and data communications are described in 10.5 through 10.7.

10.4.1. Displaying Portions of Main Storage (DISPLAY)

Function:

The DISPLAY command displays selected areas of main storage at the system console. You usually enter the command when your system administrator needs a job address displayed for program debugging purposes. This command cannot be issued from an enter stream.

Format:

```
DISPLAY addr[,{jobnumber}]
              {0        }
```

Positional Parameter 1:

addr
Is a hexadecimal number used for a specific (absolute) main storage address or a job-relative main storage address. A job-relative address is identified by the job number for the job using it; otherwise, an aboslute address is displayed.

Positional Parameter 2:

jobnumber
Specifies the job number (1 through 7) of the job in main storage whose relative address is displayed.

0
Specifies the address entered in positional parameter 1 is an absolute address.

Operator Considerations:

After you enter DISPLAY to load the display symbiont (DI), the following output message appears on the system console:

```
▷ 0i? addr[JOB#n] contents-of-selected-addr Y,N,NXT?
```

where:

i
Is a 1-digit hexadecimal message number (1 through F) consecutively assigned to output messages generated by the supervisor (in this case, in behalf of the display symbiont).

addr
Is the address of the main storage location being displayed, in hexadecimal.

JOB#n
Identifies the address being displayed as a job-relative address and identifies the job region by the job's number. If JOB#n is not displayed, the address being displayed is an absolute address.

contents-of-selected-addr
Is the hexadecimal representation of the contents of the selected main storage address.

Y
Is a message response to display the next sequential main storage location.

N
Is a message response code to terminate the display symbiont.

NXT
Is a reminder that you can display another nonconsecutive main storage location without recalling the display symbiont by responding to this output message with the solicited input message:

▷ Øi addr [,{jobnumber / 0}]

where:

i
Is the 1-digit hexadecimal message number (1 through F) of the display symbiont output message you are responding to.

addr
Is the address of the main storage location to be displayed in hexadecimal.

jobnumber
Identifies the job number (1 through 7) of the next job whose relative address is to be displayed.

0
Specifies the address entered in the input message is an absolute address.

When there are no more addresses to display, terminate the display by responding to the last display message with the solicited input message:

▷Øi N

The console messages that may be displayed during execution of the ONUERL program follow. An explanation and operator action, where required, is also provided.

ONUERL - OS/3 ERROR-LOG EDITOR. VER nn/nn

This message appears at start of ONUERL program execution, where nn/nn is the version number loaded.

ONUERL - ERROR READING PARAM. R0=error-code. TERMINATE? (Y, N)

An error has been detected in reading // PARAM cards from the job control stream, where error-code specifies the error encountered. Respond Y to terminate. Respond N to ignore the parameter.

INVALID ENTRY card-contents IGNORE? (Y OR N)

A // PARAM card has the wrong format, where card-contents is a display of the // PARAM card in error. Respond Y if the default condition is to be used; otherwise, respond N if ONUERL is to be retried with a corrected // PARAM card.

ONUERL - YELOG IS EMPTY.

YELOG contains no new data since its last reading. ONUERL is terminated.

ONUERL - ACCESS ERROR ON YELOG. R0=error-code

An error has been detected in attempting to access YELOG, where error-code specifies the error encountered. ONUERL is terminated.

FIRST RECORD NOT HISTORICAL RECORD. JOB TERMINATED.

The beginning of YELOG cannot be found. ONUERL is terminated.

ONUERL - INVALID DATA IN YELOG. JOB CANCELLED.

The data in YELOG is meaningless. ONUERL is terminated.

The ELOG file can become completely full only if the ONUERL program was not initiated when the LOG FILE IS NEARLY FULL message was displayed.

If the ELOG file becomes full, ELOG asks whether you want to turn off error logging or wraparound to the beginning of the log file. When the message

LOG FILE IS FULL W(RAP) OR O(FF)

appears, the operator responds by using the solicited message format:

Ø i {O / W}

where:

i
Identifies the console message being answered.

O
Informs the PIOCS error logging processor to turn off.

W
Informs the PIOCS error logging processor to wraparound the log fiel and continue logging.

If printing of the ELOG file has previously been initiated (with error logging set to ON), you should respond with the letter W.

If printing has not been initiated, enter O to turn off logging; then initiate the printing of the ELOG file as previously directed.

The summary report obtained as a result of running the ONUERL program provides a comprehensive listing of all errors contained in the YELOG file, as defined by the program parameters. The main body of the report consists of a single line for each error log entry. The entries are sorted chronologically by channel device number. The report also includes a summary of total I/O count and sense byte errors per device and a listing of machine check errors.

A sample ONUERL summary report is illustrated in Figure 10-2.

The description of each summary report heading follows:

DEVICE
A 4-character identification of device type assembled into the PUB at supervisor generation time.

CHDV
The channel, subchannel, and device number representing the did for the device identified.

FEATURE BYTES
A hexadecimal identification of the features installed on the device.

```
ONUERL-ERROR LOG EDIT VERSION nn/nn  ---------------------------------  yy/mm/dd hh.mm  PAGE   1

CUSTOMER ID: xxxxxxxxx OS/3 VER. xx , REV.  x, DATE yy/mm/dd, TIME hh.mm.ss, FLAGS xxxxxxxx, CHARACTERISTICS xxxx

SYSRES xxx, PRINTER xxx, READER xxx, $Y$ELOG xxx, MAIN STORAGE SIZE xxxxxxxx, USER MEMORY SIZE xxxxxxxx



***** UNISCOPE 100 CONSOLE *****
      CHAN. DVC ADDR.: 00 00
      FEATURES :
      PRINTER, 24x80 SCREEN
      LGERMSK: xxxx, LGEMSK: xxxx



DEVICE: xxxx, CHDV: xxxx, FEATURE BYTES: xxxx, PHYSICAL ADDR: x, DATE yy/mm/dd

                                        (*=BCW)       CCW / BCW              DISK ADR DV/SC
  TIME   JOB NAME  VSN   RT R/U OPR I/O'S * CCAAAAAAFFBBBBBB CCAAAAAAFFBBBBBB CCC HH RR STATS. EMSK SNS BYTES 0-5
hh.mm.ss _______ ______ __ xxx  x   xxxxx  ---------------- ---------------- --- -- -- -- -- ---- ---- ---- ----
 xxxx SSB 6-23: ____ ____ ____ ____ ____ ____ ____ ____ ____
   .
   .
   .

DEVICE: xxxx, CHDV: xxxx, MODE: xx, DATE yy/mm/dd

                                        (*=BCW)       CCW / BCB                   XPCT ACT. DV/SC
  TIME   JOB NAME  VSN   RT R/U OPR I/O'S * CCAAAAAAFFBBBBBB CCAAAAAAFFBBBBBB MODE BLCK BLCK STATS. EMSK SNS BYTES 0-5
hh.mm.ss _______ ______ __ xxx  x  xxxxx   ---------------- ----------------  --  ---- ---- -- --  ---- ---- ---- ----
   .
   .
   .

  TIME   JOB NAME START   SIZE  SIO MACH CHK OLD PSW PGM CHK OLD PSW  RELOC RG SUP R-0  SUP R-1
hh.mm.ss _______ ______ ______  -  ---------------- ----------------  -------- -------- --------
```

Figure 10—2. Sample Error Log Summary Report

PHYSICAL ADDR
The starting address in main storage for the user job running at time of error report.

DATE, TIME
The date and time the error log was written to the error file.

JOB NAME
The name of the job to which the device was allocated at time of error.

VSN
The volume serial number of the volume mounted on the device at time of error.

RT
Record Type: 20 = I/O error, 31 = operator response, 40 = termination

R/U
The number indicates the number of retries the supervisor initiated in trying to recovery from the error. The letter following the retry number indicates the error's disposition.

R = recovered by supervisor.

U = not recovered by supervisor; error passed to user program.

OPR
Operator response to error message displayed on system console screen, if any. Possible response is U (unrecovered), I (ignore), C (cancel), or R (retry).

I/O'S
Contains the number of valid I/Os prior to the error. It indicates the frequency of errors for the device.

BCW and CCW fields
The error or failing buffer control word. (Entire BCW is indicated by an * preceding BCW.)

CCAAAAAA: CC=command code, AAAAAA=address

FFBBBBBB: FF=flag, BBBBBB=count

For a description of the BCW and CCW fields, see MH2737, SPERRY UNIVAC Processor Type 3029-02, -30 Functional Analysis and Servicing. On selector channels, two CCW's will be displayed.

DISK ADR
If the failing device was a disk and the first CCW was a seek, the cylinder (CCC) and the head (HH) identifications are recorded. If, in scanning the chain, a search equals command is recognized, the record (RR) is recorded.

DV/SC STATS
The device and subchannel status recognized when the error recovery sequence was completed.

EMSK
The error log mask in effect at time of error.

SNS BYTES
The bytes recognized when the first unit check of the error recovery sequence was initiated. Up to six sense bytes are displayed for all devices (except the 8430 disk drives, which display 24 sense bytes) on the following line.

MODE
On tape devices, normal and current tape mode will be displayed.

XPCT BLCK
Expected block number.

ACT.BLCK
Actual block number.

The MODE, XPCT BLCK, and ACT BLCK headings, along with CCCHHRR, will not be displayed on unit record devices.

Machine Check Error Display
MACH CHK OLD PSW
PSW at time of error.

PGM CHK OLD PSW
PSW at time of error.

RELOC RG
Contents of the relocation register.

SUP R-0
Contents of supervisor.

SUP R-1
Registers 0 and 1.

10.4.7. Setting Up Physical Unit Blocks (SET IO)

Function:

The SET IO command allows you to set specific bits in the physical unit blocks (PUB), which define operational characterstics and assignments of I/O devices. There is one physical unit block, comprising a 3- or 4-character did (device address), for each physical device in your system. You must set all devices or subsystems *DOWN* before attempting operation on the device, such as forms loading or changing ribbon; or in case of malfunction, before turning device off. This is required if the processor is to continue operation with other peripheral devices while the subject device undergoes isolated operations. Before performing a procedure or turning power off for the device, key in SET IO,did,DOWN. After the offline procedure completes or after turning on power independently of the processor, key in SET IO,did,UP.

NOTE:

The system automatically sets all devices or subsystems not online during IPL time to not available. After IPL, any attention interrupt from a device causes that device to be set available.

Format:

```
SET IO,did,{AV
           |DOWN
           |EON
           |EOF
           |FEA,type-code
           |HOME
           |NA
           |NOSHARE
           |RDR
           |SHARE
           |TYPE,type-code
           |UP
           }
```

Positional Parameter 1:

IO

Specifies the change is made in the device address specified by positional parameter 2.

Positional Parameter 2:

did

Specifies a 1- to 4-character numeric field identifying the device address to be changed (device addresses are usually attached at a visible location on the device cabinet) as follows:

0000	Integrated peripheral channel (console)
0001	Integrated peripheral channel (card reader)
0002	Integrated peripheral channel (printer)
0003	Integrated peripheral channel (card punch)
0100 through 0177	Multiplexer channel
0300 through 0307	Integrated disk adapter
0400 through 0477	Selector channel 4 (90/30 B only)
0600 through 0677	Selector channel 6 (90/30 B only)

Positional Parameter 3:

AV

Specifies the device identified by positional parameter 2 is recognized by the system and available for assignment to user jobs.

DOWN

Specifies the device identified in positional parameter 2 is not to be considered available for assignment to user jobs, although the device is recognized by the system.

EON
Turns on error logging for the specified device.

EOF
Turns off error logging for the specified device.

FEA
Modifies the feature bytes of the device specified in positional parameter 2. The type code in positional parameter 4 is the new feature bytes.

HOME
Synchronizes the operating system with the physical paper position of a printer during a home operation.

NA
Specifies the device identified by positional parameter 2 is not recognized by the system and is not available for assignment to user jobs.

NOSHARE
Forbids allocation of the device specified in positional parameter 2 to more than one program simultaneously.

RDR
Assigns the new did specified in positional parameter 2 as the system card reader.

SHARE
Permits the device specified in positional parameter 2 to be shared by more than one program simultaneously.

TYPE
Modifies the type bytes of the device specified in positional parameter 2. The type code specified in positional parameter 4 describes the modification.

UP
Specifies the device identified in positional parameter 2 is considered available for assignment to user jobs. The device remains recognized by the system.

Positional Parameter 4:

type-code
A 1- to 4-character field specifying the device, its options, and features desired.

10.4.8. Setting the Seek Separation Feature (SET SEEKSEP)

The SET SEEKSEP command activates the seek separation feature. This feature allows you to access more than one disk device during input/output operations by separating seek time from read/write time.

Format:

```
SET SEEKSEP{OFF} [,{ALL}]
           {ON }  [,{did}]
```

Positional Parameter 1:

OFF
Disables prevously enabled seek separation feature.

ON
Enables the seek separation feature.

Positional Parameter 2:

ALL
Sets the seek separation feature for all disks.

did
Sets the seek separation feature for all this device only. If this parameter is omitted, the feature is set for all devices.

10.4.9. Reading a Mounted Volume Serial Number (AVR)

Function:

The AVR command reads the volume serial number of a premounted prepped disk pack, diskette, or magnetic tape volume and stores it in the device physical unit block. This command is required when a disk pack or magnetic tape is mounted on a unit that does not have an attention interrupt capability (i.e., the card reader, punch, or printer of a 9300 system attached to the 90/25 processor).

Format:

```
AVR did[,did][,did]
```

Positional Parameters 1 through 3:

did
Specifies the device addresses for the volumes to be recognized.

10.4.10. Displaying Job Status (DISPLAY JS)

Function:

The DISPLAY JS command displays the status of jobs in the system at the system console. You can display the status of a specific job or all jobs in main storage, or display the status of a specific job in a scheduling priority queue. The display includes the job name along with the CPU time used when the job is under execution, or the reason why the job is not executing (such as waiting for IO, waiting for mount message, under a pause), or the scheduling priority queue in which the job resides.

Format:

```
DISPLAY JS[,jobname]
```

Positional Parameter 1:

JS
Displays that job status at the system console.

Positional Parameter 2:

jobname
Displays the status of the job name specified. If omitted, the status of all jobs in main storage is displayed.

Example:

Operator keyin:

DI JS,MYJOB

Function requested:

Produce a display of status information about a job named MYJOB.

Typical informational messages:

The following are examples of the job status informational messages produced by the DISPLAY JS command. The examples show what information DI JS might display if it were entered against MYJOB as the job proceeds through the various steps in job processing.

For the message:

MYJOB IN STEP 01(LNKEDT00)-PRI=10 CPU-TIME=00:01:43.874

MYJOB is active in its first step, performing linkage editing. The CPU-TIME portion of the display indicates the linkage editor had control of the CPU for 1 minute, 43 seconds, and 874 milliseconds. If the job is proceeding, you can reenter DI JS for MYJOB and see an increase in the CPU TIME figure.

For the message:

MYJOB IN STEP 02(LIBS0000)-WAITING FOR I/O #00005736

MYJOB is in its second step, executing the librarian. Currently the 5736th I/O operation of this step is being performed. If you reenter DI JS, you may see the I/O number increase. If MYJOB remains at #00005736, it might be stuck, requiring your intervention.

For the message:

MYJOB IN STEP 03 - IN STEP PROCESSOR

MYJOB is between job steps. Step 03 either has just completed or is about to start.

For the message:

MYJOB NOT YET SCHEDULED - INSUFFICIENT MAIN STORAGE

MYJOB is not executing; it is placed on a job queue but isn't scheduled for execution because not enough main storage is available.

10.4.11. Dumping the Contents of Main Storage (SYSDUMP)

Function:

The SYSDUMP command is used to dump the entire contents of main storage to the YDUMP file on SYSRES. Use the SYSDUMP command whenever a system dump is required without supervisor reloading (no re-IPL is required). After the contents of main storage are dumped, the job SYSDMPnn (where nn is a unique number assigned by the system) is automatically initiated to print the YDUMP file. The YDUMP file locks until the SYSDMPnn job completes to prevent other system functions (caused, for example, by an //OPTION SYSDUMP statement or by system errors) from also using the file. (For further details, see the current version of the dump analysis manual, UP-8837.)

If you don't want a printout of the dump, enter NONE when the SYSDMPnn job asks what type of dump should be printed. If you delete the SYSDMPnn job from the job queue or cancel it before it sends you the SD01 DUMP OPTION message, you must enter the SET SY command (10.4.11) to unlock the YDUMP file.

You can't use the system console keyboard until the main storage contents are completely written (only a few seconds). After the keyboard unlocks, you can resume system activity without impairing the integrity of your dump.

Format:

SYSDUMP

There are no positional parameters.

NOTE:

When a re-IPL is required, the dump should be obtained by pressing SYSTEM RESET and RUN on the maintenance panel to preserve low-order main storage.

10.4.12. Setting the YDUMP File to Unlocked Condition (SET SY)

Function:

The SET SY command unlocks the YDUMP file after a SY command (10.4.10) or system error has locked it. You use the SET SY command to unlock YDUMP if you previously entered the SYSDUMP command and deleted the SYSDMPnn job from the job queue or cancelled it before it displayed the SD01 output message.

Format:

```
SET SY,LOFF
```

Positional Parameter 1:

SY
Specifies the YDUMP file condition is to be set.

Positional Parameter 2:

LOFF
Specifies the YDUMP file is to be unlocked.

10.4.13. Terminating System Activity (SHUTDOWN)

Function:

The SHUTDOWN command terminates system activity in an orderly manner. The spooler and job scheduler will not start any new files or jobs. Interactive services will not start any new functions and will terminate when its current activity ceases.

Format:

```
SHUTDOWN [DDP]
```

Positional Parameter 1:

DDP
Specifies that only distributed data processing will be terminated when its activity ceases.

NOTE:

DDP is shut down either by using the SHUTDOWN DDP command or automatically after 10 minutes of no DDP activity. During DDP shutdown processing, all DDP activity is logged into the interactive services log file. The activity information remains in the log file and will be printed only when interactive services is shut down by using the IS SHUTDOWN message (10.5.3.4).

10.5. INTERACTIVE SERVICES

10.5.1. General

The system operator uses interactive services with an extended set of commands and messages to control the interactive system environment. (For a description of all interactive services commands, see the current version of the interactive services command and facilities user guide/programmer reference, UP-8845.) These commands and messages enable you to exercise control over the interactive OS/3 operating system, all jobs within the system, all workstation users (local locations), and all terminal users (local and remote locations).

10.5.2. Interactive Services Commands

When interactivity is included in your system at SYSGEN time, the interactive services components are loaded automatically whenever required for the system operator and workstation users. Provided ICAM and the global user service task (GUST) are ready, a workstation user's interactive entry or a command from the operator loads interactive services for terminal users. Interactive services commands are used to:

- send messages to workstation and terminal users;
- ask questions of workstation and terminal users;
- display the volumes in use and the status of active workstations, terminals, jobs, and functions;
- display status of system resources in use and available;
- run workstation and terminal user sessions in a batch mode; and
- load interactive services.

10.5.2.1. Sending Messages to Users (TELL)

Function:

The TELL command sends a message not requiring a response to a specific workstation or terminal user or to all users.

Format:

```
TELL {ALL      } 'text'
     {user-id'}
```

Positional Parameter 1:

`ALL`
Displays the message on all active user screens.

`user-id`
Specifies the 1- to 6-character alphanumeric identification of the user to receive the message display.

Positional Parameter 2:

`text`
Is the text of the message to be sent. The text may be a maximum of 48 characters long. Text must be preceded and followed by apostrophes.

10.5.2.2. Asking Questions of Users (ASK)

Function:

The ASK command sends a message that requires a response to a specific workstation or terminal user. The command displays your question to the specified user, accepts the reply, and returns the reply to the console screen.

Format:

```
ASK user-id,'text'
```

Positional Parameter 1:

`user-id`
Specifies the 1- to 6-character alphanumeric identification of the local or remote user that is to receive the message display.

Positional Parameter 2:

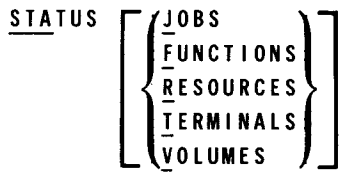
`text`
Is the text of the message to be sent. The text may be a maximum of 48 characters long. Text must be preceded and followed by apostrophes.

10.5.2.3. Displaying System Status (STATUS)

Function:

The STATUS command displays the volumes currently in use; the status of active workstations, terminals, jobs, and functions; and the status of system resources in use and available.

Format:



Positional Parameter 1:

JOBS
Displays a listing of the jobs and symbionts currently active in the system. The list includes the job or symbiont name and the amount of storage and CPU time used by each, information on what program each job is executing and its job step number, the job slot number for the job, and the master user-id for the job or symbiont. A summary line displays the amount of free main storage in the system and the largest contiguous region.

FUNCTIONS
Displays a listing of all active commands and tasks initiated by users.

RESOURCES
Displays a listing of the amount of storage being used by all users; the number of interactive, batch, and batch with interactive commands (ENTER) tasks; the number of active jobs and configured job slots; and the amount of total system storage.

TERMINALS
Displays a listing of all currently active (logged on) workstations and terminals and the associated user-id.

VOLUMES
Displays a listing of the tape, disk, and diskette volumes currently mounted on the system.

10.5.3. Interactive Services Messages

You use unsolicited messages provided by interactive services for additional control over the interactive environment. These messages are used to:

- terminate workstation and terminal user sessions;
- restrict and release new workstation and terminal user sessions;
- terminate interactive services; and
- control interactive services for terminals.

10.5.3.1. Terminating User Sessions (REMOVE Message)

Function:

The REMOVE unsolicited message terminates a specific workstation or terminal user session, all user sessions, or a single command for a specific task. Following the termination, a cancellation message is displayed on the terminated user's screen.

Message Format:

```
ØØ IS REMOVE {user-id}
             {ALL    }
             {task-id}
```

where:

`user-id`
Specifies the particular user whose session is to terminate. User-ids can be determined by using the STATUS TERMINAL command. All interactive functions for the user-id specified are terminated and the user is logged off. If the user is running an interactive session as a batch job (via ENTER), the user is logged off; however, the session is not affected. A message is displayed on the console screen to indicate when the user cannot be logged off.

`ALL`
Specifies all user sessions and batch runs are to terminate. All interactive functions for all users are terminated and the users are not logged off. Users running interactive sessions as batch jobs (via ENTER) are logged off. A message indicating which users cannot be logged off is displayed on the console screen.

`task-id`
Specifies the particular task under which a command is to terminate. Task-ids can be determined by using the STATUS FUNCTION command.

NOTE:

When distributed data processing (DDP) is present in the system, the IS REMOVE ALL command will have no effect on the task with a user-id of DDP△MR and a command identification of DDP△MAIL. Also, this task cannot be removed by the user-id or the task-id. This is a DDP task that has several responsibilities including DDP termination processing. This task is removed when DDP is shut down (by you using the SHUTDOWN DDP command or automatically after 10 minutes of no DDP activity). During DDP shutdown processing, all DDP activity is logged into the interactive services log file. The activity information remains in the log file and will be printed only when interactive services is shut down (using the IS SHUTDOWN message).

10.5.3.2. Restricting New User Sessions (CLOSE Message)

Function:

The CLOSE unsolicited message restricts any new workstation or terminal user sessions from starting. Currently active sessions are not affected.

Message Format:

```
ØØ IS CLOSE
```

10.5.3.3. Releasing New User Session Restrictions (OPEN Message)

Function:

The OPEN unsolicited message removes a previously entered CLOSE or reverses an incomplete SHUTDOWN to permit new workstation and terminal user sessions to start.

Message Format:

```
ØØ IS OPEN
```

10.5.3.4. Terminating Interactive Services (SHUTDOWN Message)

Function:

The SHUTDOWN unsolicited message terminates interactive services after all sessions have completed. If no sessions are active when you enter the command, it takes effect immediately. Otherwise, interactive services do not terminate until all active sessions complete. An OPEN message can be used to stop a shutdown in progress (incomplete). After the interactive services facility is completely shut down, you can restart it only via the IS command (10.5.2.4); it is not automatically loaded for workstation users via an interactive services command entry.

Format:

```
ØØ IS SHUTDOWN
```

10.5.3.5. Controlling Interactive Services for Terminals (REMOTE Message)

Function:

The REMOTE unsolicited message provides additional control over interactive services for terminals. You must load ICAM and run the global user service task (GUST) before you start interactive services for terminals. (See 10.7 for a description of the ICAM and GUST operator procedures.)

Format:

```
ØØ IS REMOTE{START   }
            {SHUTDOWN}
            {CANCEL  }
```

Parameters:

START
Starts interactive services for terminals after ICAM and GUST are ready. You use this parameter when interactive services are already active in the system (via a prior IS command or an automatic start by a workstation user). If you enter the REMOTE START message, with ICAM and GUST not ready, an error message is displayed on the console screen.

SHUTDOWN
Terminates the interactive services facility for terminals as soon as the last terminal session completes. (Workstation users are not affected.)

CANCEL
Immediately terminates the interactive services facility for terminals. Terminal users are not logged off.

10.6. SYSTEM UTILITY SERVICES

10.6.1. General

The system utility symbiont (SL$$SU) is a multipurpose utility that allows you to perform many different functions using cards, tapes, disks, or diskettes. Table 10-2 breaks down the different functions to the media associated with them.

Table 10—2. System Utility Functions (Part 1 of 2)

Function Code	Function Performed
Card Functions	
CC	Reproducing cards punched in Hollerith code
CC96	Reproducing 96-column cards
CCB	Reproducing cards punched in binary and Hollerith code
CCS	Reproducing and resequencing source programs
CS96	Reproducing and resequencing source programs contained on 96-column cards
CT	Writing card to tape in unblocked format
CT96	Writing 96-column cards to tape in unblocked format
CTR	Writing card to tape in blocked format
CP	Listing cards
CP96	Listing 96-column cards in character format
CH	Listing cards containing compressed mode
CH96	Listing 96-column cards in vertical hexadecimal format
JCP	Punching cards from the system console

Table 10—2. System Utility Functions (Part 2 of 2)

Function Code	Function Performed
Tape Functions	
TT	Copying a tape to another tape
TH	Printing a tape in character and hexadecimal format
THR	Printing a tape in character, hexadecimal, deblocked format
TP	Printing a tape containing only standard characters
TPR	Printing a tape in character and deblocked format
TRS	Locating a specific record on tape
TRL	Changing existing records on tape
TC	Punching cards from tape
INT	Prepping a tape
FSF	Forward space to a specific file
BSF	Backward space to a specific file
FSR	Forward space to a specific record
BSR	Backward space to a specific record
WTM	Writing tape marks
REW	Rewinding a tape
RUN	Rewinding a tape with interlock
ERG	Erasing a portion of a tape
Disk Functions	
DD	Printing a disk in unblocked format
DDR	Printing a disk in reblocked format
VTP	Printing the volume table of contents of a disk
SVT	Printing short format VTOC file
AVX	Displaying available disk extents on console screen
DID	Changing volume serial number (VSN) of a disk
Diskette Functions	
DD	Printing a diskette in unblocked format
VTP	Printing the data set labels of a diskette
DID	Changing volume serial number (VSN) of a diskette

NOTE:

If XXX is entered in place of the function code, all function codes are displayed.

10.6.2. System Utility Commands and Messages

Function:

The SU/TU command loads the system utility symbiont. The SU and TU symbionts can be used interchangeably for all functions. However, we recommend you use the TU symbiont for tape operations since it increases the buffer size for all selector channel tapes from 8189 to 32,767 bytes.

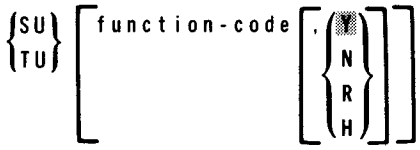
You can include the required funciton as a parameter with the SU/TU command. A spooling parameter can also be entered with the command, if spooling is configured in your system. (For a description of all spooling commands, see the current version of spooling and job accounting concepts and facilities, UP-8869.) When you enter the command alone to load the symbiont, you enter the function as a solicited message. After the symbiont is loaded, you control it by responding with solicited messages. You use unsolicited messages only to terminate the symbiont or current symbiont functions on certain occasions.

NOTES:

1. *The TU symbiont should be run only when no other job will be starting up or performing multiple steps that will allocate the same volumes. This could cause the system utility to terminate abnormally.*

2. *The SU/TU command cannot be issued from DDP enter streams.*

Format:



Positional Parameter 1:

`function-code`
Specifies the appropriate 2- or 3-character function code. (Function codes are shown in Table 10–2.) If omitted, the symbiont displays a message requesting that you enter a function.

All possible function codes that are recognized by either symbiont can be displayed on the system console by entering XXX in place of the function code. Following this display, the symbiont requests that you enter the required function code.

Positional Parameter 2:

Y
Spools the system utility output.

N
Specifies the system utility output is not spooled.

R
Retains system utility output in HOLD condition in the spool file after it is printed or punched. The retained output is unavailable for additional processing until released via the BEGIN SPL command.

H
Places system utility output in HOLD condition in the spool file before it is printed or punched. The output must be released via the BEGIN SPL command to permit processing.

This parameter is entered only if spooling is configured in your system. When Y is specified or taken as the default condition, the output writer automatically prints or punches any spooled output at the end of every SU function.

Operator Considerations:

When the symbiont is loaded, the following message is displayed:

0i SYSTEM UTILITY SYMBIONT LOADED

where:

i

Is a 1-digit hexadecimal message number (1-F) consecutively assigned to output messages generated by the supervisor (in this case, in behalf of the system utility symbiont).

If the command entry includes a function code, the symbiont completes the requested function and then displays an ENTER REQUIRED FUNCTION message to allow you to either terminate the symbiont or request another function.

If you omit the function code, the symbiont displays the ENTER REQUIRED FUNCTION message to allow you to enter a function. The message is displayed as follows:

- With spooling:

  0i? ENTER REQUIRED FUNCTION AND SPOOL OPTION [,Y,N,R,H] DEFAULT=Y

- Without spooling:

  0i? ENTER REQUIRED FUNCTION

Initiate the required function with the following keyin:

- With spooling:

  0i function-code, {Y / N / R / H}

- Without spooling:

  0i function-code

Each time the symbiont completes a requested function, it transmits the ENTER REQUIRED FUNCTION message to allow you to either terminate the symbiont or request another function.

If your system supports spooling and the spool option is incorrectly entered, (i.e., a character other than Y, N, R, or H is entered), the following message is displayed:

0i? IS {PRINTED / PUNCHED} OUTPUT TO BE SPOOLED FROM SU Y,N,R,H

Enter the required spooling option with the keyin:

0i {Y / N / R / H}

To terminate the symbiont, you reply to the ENTER REQUIRED FUNCTION with the end-of-job keyin:

0i EOJ

The symbiont then terminates and displays the message:

0i SYSTEM UTILITY SYMBIONT ENDED

You can also end the symbiont by pressing the **MESSAGE WAITING** key on the system console and keying in one of the following unsolicited messages:

00 SU EOJ
00 TU EOJ

This permits you to terminate the symbiont before it completes a function.

To terminate only the current function of the symbiont, key in one of the following unsolicited messages:

00 SU END
00 TU END

NOTE:

When message replies are keyed in incorrectly or the reply cannot be honored, the symbiont requests the information to be keyed in again. If no determination can be made on why the keyed input is not accepted, use the unsolicited message (above) to terminate the current function or to terminate the symbiont.

10.6.3. Card Functions

All the card functions you can perform are described in this subsection. All input card files must be terminated by a card with the words END OF DATA punched in columns 1 through 11.

Proceed as follows to perform the card functions.

1. As described in 10.6.2:

   a. Enter the SU symbiont command.

   b. Enter the appropriate function code either as a command parameter or as a solicited message response to the ENTER REQUIRED FUNCTION message.

   c. Enter the spooling option; otherwise, default is Y (applicable only if spooling is configured).

2. If a card file is being read, place it in the card reader designated the system reader (SYSRDR). If this card reader is unavailable, the first available card reader is assigned to the symbiont, causing the following message to be displayed on the system console:

   0i USE READER did

   where:

   did
   Is the device address of the card reader assigned to read the input file.

   If no card readers are available, the function aborts and the following message is displayed on the system console:

   0i NO READER AVAILABLE

   Likewise, if the required output device is not available, the function aborts and the following message is displayed on the system console:

   0i NO {PUNCH / TAPE / PRINTER} AVAILABLE

If the required devices are available, the operation continues for each function code, as described in 10.6.3.1 through 10.6.3.13.

10.6.3.1. Reproducing Cards Punched in Hollerith Code (CC)

You use the CC function code to reproduce cards in 80 x 80 format containing the Hollerith code. All job control cards, even the /*, can be reproduced by using this function. You must submit an END OF DATA card with your input deck indicating the end of file to the symbiont.

10.6.3.2. Reproducing 96-Column Cards (CC96)

You use the CC96 function code to read 96-column cards and punch 80-column cards. Columns 81 through 96 of the input cards are truncated. The input card file must be terminated with an END OF DATA card.

10.6.3.3. Reproducing Cards Containing Binary Data (CCB)

You use the CCB function to reproduce cards containing binary data in addition to the Hollerith code. Again, you must submit an END OF DATA card as the last card in your input deck. When punching column binary, the output must not be spooled.

10.6.3.4. Reproducing and Resequencing Source Programs (CCS)

You use the CCS function code to reproduce and resequence an assembler (BAL), COBOL, or RPG II source language program. For a BAL program, the program name can be up to three characters in length; COBOL can be up to eight characters in length; and RPG II can be up to six characters in length. If you supply a name having fewer characters than the number permitted, the name is left-justified and space-filled. You must submit an END OF DATA card as the last card of your source program. Tables 10–3 through 10–5 show the formats of the source programs being reproduced and resequenced.

10.6.3.10. Listing 96-Column Cards in Character Format (CP96)

You use the CP96 function code to list your 96-column cards in character format on the printer. The processing constraints for this function are the same as those described for the CP function code.

10.6.3.11. Listing Cards Containing Compressed Mode (CH)

You use the CH function code to list cards containing the compress mode (hexadecimal characters) and the standard characters. Again, you must have an END OF card as the last card in your input deck.

10.6.3.12. Listing 96-Column Cards in Vertical Hexadecimal Format (CH96)

You use the CH96 function code to list your 96-column cards in character and vertical hexadecimal format. An END OF DATA card is required to terminate the function.

10.6.3.13. Punching Cards from the System Console (JCP)

The JCP function punches job control or data cards entered through the system console. Up to 60 columns may be entered in reply to the scale message. If 60 or more columns are needed, position the cursor under the 0 of 60 in the scale message and transmit. Another message requesting 20 more characters will appear. If a card having a blank in column 1 or column 61 is required, key in a right parenthesis instead of a blank in that column. The symbiont replaces a right parenthesis in column 1 or 61 with a blank. The right parenthesis is needed because all messages received by SU are returned left-justified. To terminate the JCP function, key in END and transmit immediately.

10.6.4. Tape Functions

The tape functions you can perform are described in 10.6.4.1 through 10.6.4.3.17. We recommend you use the SU command if your tape block size does not exceed 8189 bytes; TU supports tape blocks up to 32,767 bytes.

10.6.4.1. Tape Addressing

You identify the tape used for a TU function by its tape unit device address, mode setting, and block count characteristics for the tape in the following format:

```
cuummb
```

where:

cuu
Is the device address.

mm
Is the tape mode setting. If mm is blank, 00, or not entered (e.g., response is 102), the SYSGEN mode settings are assumed. (Refer to the current version of the job control user guide, UP-8065.)

b
Is the tape block count characteristics. If blank or omitted, the tape is assumed not to have a block count. If b is entered, the tape is assumed to have a block count.

NOTES:

1. *The block count specification is not needed for tape functions used to position a tape (e.g., FSF).*

2. *The record number (REC) printed by the system utility for the TRS function is relative to the beginning of scan. For other tape functions, it is relative to where printing begins. The REC number does not correspond to the 3-byte block number on block-numbered tapes.*

3. *The tape block number (TBLK) printed for the TRS function is a display of the 3-byte block number prefixed to block numbered tapes.*

10.6.4.2. Tape Error Processing

If an error is encountered on an input tape, control is turned over to a tape error correction routine, where communication is established with you to determine whether the error should be ignored, bypassed, or the function terminated.

If an error is ignored, the record is processed as is. If an error is bypassed, the input tape is reread before returning to the active function; therefore, no processing is performed on the error block.

For the messages:

0i TAPE ERROR ON INPUT TAPE

0i? B-BYPASS, I-IGNORE, OR E-END FUNCTION

A tape error has occurred and one of the following options may be replied:

0i BYPASS

This reply is not applicable during a TRL function.

0i IGNORE

0i END

The message:

0i WARNING: TAPE BLK FILLS AVAILABLE BUFFER OR MAY BE TRUNCATED

warns that a tape input or output record completely fills the allocated buffer. SU reads/writes a maximum of 8189 bytes and TU reads/writes a maximum of 32,767 bytes; any tape block exceeding this is truncated.

10.6.4.3. Tape Operating Instructions

Proceed as follows to perform a tape function:

1. As described in 10.6.2:

   a. Enter the TU symbiont command (or enter SU if the tape block size does not exceed 8192 bytes).

   b. Enter the appropriate function code either as a command parameter or as a solicited message response to the ENTER REQUIRED FUNCTION message.

c. Enter the spooling option; otherwise, default is Y (applicable if spooling is configured).

2. Place the input tape volume on an available tape unit and identify the tape unit to the TU (or SU) symbiont by keying in its unit address as described in 10.6.4.1.

If the required devices are available, the operation continues for each function code as described in 10.6.4.3.1 through 10.6.4.3.17.

Procedure:

You identify the scan field by its length and location within the logical record. The logical record length is also required for deblocking purposes. When you enter the scan argument in either hexadecimal or character format, it is compared to the scan field of each logical tape record and printed on the printer for future reference. When an equal is located, the block having the record is printed in character and hexadecimal format. After each find, you are given the option of continuing the scan for more finds or terminating the function. Tape positioning does not occur at either the beginning or the end of this function. The tape error correction route is enabled, so you can ignore or bypass tape errors (10.6.4.2). The function is terminated either at the end of file (two tape marks side by side) or when a find is made and you terminate the function. The scale is printed to improve readability. You also enter the scan argument for use in the search.

For the messages:

0i INVALID HEX CHAR

An invalid hexadecimal character is detected when you entered the scan argument in hexadecimal. You reenter the sequence to obtain the scan argument.

0i NO MATCH FND

An end-of-file mark (two tape marks back to back) is encountered before any finds are made. The function terminates.

0i? CONTINUE SCAN? Y/N

Message displayed each time a find is made. You enter Y to continue the scan or N to terminate the scan.

The displayed record (find) is preceded by a header that includes:

REC

Specifies the sequential number of the record, relative to the beginning of the search.

DATA

Specifies the logical record length.

TBLK

Specifies the block number, which is maintained by the system in a 3-byte prefix to each tape block.

An example of a tape block count search for a record containing a key field of 4637275467 in its first 10 bytes is as follows:

For the message:

0i? CUUMMB - INPUT TAPE B=BLK CNT

Key in:

0i 10000B

For the message:

0i? LOGICAL REC LENGTH

Key in:

0i 125

For the message:

0i? LENGTH ARGUMENT (1-30)

Key in:

0i 10

For the message:

0i? STARTING DATA POSITION IN REC

Key in:

0i 1

For the message:

0i ENTER IN HEX-H, CHAR-C

Key in:

0i C

For the messages:

0i ENTER 10 BYTES, 1 CHAR PER BYTE
0i? 1...5...10

Key in:

0i 4637275467

For the message:

0i? CONTINUE SCAN? Y/N

Key in:

0i N

10.6.4.3.7. Changing Existing Records (TRL)

You use the TRL function code to change an existing block in your tape file. You can change either character or hexadecimal data. In order for the symbiont to change your block, the following information is needed:

- relative block number;
- record number;
- data to be changed; and
- position where the change is to take place.

10.6.4.3.15. Rewind Tape (REW)

You use the REW function code to rewind your tape to load point.

Procedure:

For the message:

0i? CUUMM - OUTPUT TAPE

Enter the tape unit device address and the mode setting of the tape to be rewound (refer to 10.6.4.1).

10.6.4.3.16. Rewind and Unload Tape (RUN)

You use the RUN function to rewind either UNISERVO 16 or UNISERVO 20 tapes to load point with interlock.

Procedure:

For the message:

0i? CUUMM - OUTPUT TAPE

Enter the tape unit device address and the mode setting of the tape to be rewound (refer to 10.6.4.1).

10.6.4.3.17. Erasing Tape Record Gap (ERG)

You use the ERG function code to erase a portion of your tape. This function is useful to erase known defective areas on your tape. It erases the specific tape for approximately 3.5 inches.

Procedure:

For the message:

0i? CUUMM - OUTPUT TAPE

Enter the tape unit device address and the mode setting of the tape to be erased (refer to 10.6.4.1).

10.6.5. Disk Functions

All the disk functions that can be performed are described in 10.6.5.1 through 10.6.5.2.8.

10.6.5.1. Operating Considerations

When operating with the system utility on disks, an end-of-file record is a disk record on which data length is in binary zeros (not applicable on IDA disk subsystems).

10.6.5.2. Disk Operating Instructions

Proceed as follows to perform a disk function:

1. Place the subject disk volume on an available disk unit.
2. As described in 10.6.2.:
   a. Enter the SU symbiont command.
   b. Enter the appropriate function code either as a command parameter or as a solicited message response to the ENTER REQUIRED FUNCTION message.
   c. Enter the spooling option; otherwise, default is Y (applicable if spooling is configured).

10.6.5.2.1. Printing a Disk in Unblocked Format (DD)

You use the DD function code to print your disk pack in character and hexadecimal format. The DD function does not deblock your logical records.

Procedure:

For the messages:

0i? ENTER DVC ADDRESS

Enter the disk unit device address of the disk pack to be displayed.

0i? CCCHH - BEGIN OR FILE-ID

Enter in decimal the beginning cylinder (CCC) and head (HH) to be displayed, or the file identifier (up to 44 characters) as used on the // LBL job control statement when the file was created. If you enter less than 44 characters, the file ID is padded with blanks on the right. Printing begins at the start of the file when a file is entered.

0i? CCCHH - END

Enter the last cylinder (CCC) and head (HH) in decimal to be displayed.

0i? NO OF TRACKS TO PRINT UP TO 9

If you entered the file ID, now enter the number of tracks to be printed. Note that printing begins with the low cylinder and head numbers of the first extent. If the file is not laced, printing continues for the requested number of tracks, or until end of file (EOF) is encountered, whichever comes first. If the file is laced, printing continues for the required number of tracks. Printing is not confined to the extents specified in the format label. (Laced files are discussed in the system service programs user guide, UP-8062 (current version) in the section describing diskette prep.)

Example 1:

Cylinder 3, head 4 through cylinder 6, head 6 on device 300 are to be printed.

For the message:

0i? ENTER DVC ADDRESS

Key in:

0i 300

For the message:

0i? CCCHH - BEGIN OR FILE-ID

Key in:

0i 00304

For the message:

0i? CCCHH END

Key in:

0i 00606

Example 2:

The first three tracks from a file called SEQUENTIAL DISC on device 440 are to be printed.

For the message:

0i? ENTER DVC ADDRESS

Key in:

0i 440

For the message:

0i? CCCHH - BEGIN OR FILE-ID

Key in:

0i SEQUENTIAL DISC

For the message:

0i? NO OF TRACKS TO PRINT UP TO 9

Key in:

0i 3

10.6.5.2.2. Printing a Disk in Deblocked Format (DDR)

You use the DDR function code to print your disk pack in deblock format in both character and hexadecimal formats. This function is similar to the DD function, with the only exception that your logical records are deblocked.

Procedure:

For the messages:

0i? ENTER DVC ADDRESS

Enter the disk unit device address of the disk pack to be displayed.

0i? CCCHH - BEGIN

Enter in decimal the first cylinder (CCC) and head (HH) to be printed.

0i? CCCHH - END

Enter in decimal the last cylinder (CCC) and head (HH) to be printed.

Oi? RECORD SIZE

Enter in decimal the logical record size.

Oi? BLOCK SIZE

Enter in decimal the logical block size. However, if the block size is not an exact multiple of the record size, reenter the record size/block size sequence.

10.6.5.2.3. Printing the Disk Volume Table of Contents (VTP)

You use the VTP function code to get a copy of your volume table of contents (VTOC). You can print:

- a full VTOC listing, giving you all the device information plus extents and other information for all your allocated files;
- device information only, giving you the available space left on your volume and other information regarding your volume; or
- file information only, giving you the extent and other information regarding the file.

VTP edits and prints VTOC information for the requested volume. VTP will not process other than OS/3-created VTOCs. Use the DD function to print a VTOC from a disk not created by OS/3.

Procedure:

Three list options are available:

- VSN, DI - Device Information Only

  Lists the available free extents and other information on the requested volume.

- VSN, FILE ID. - File Information

  Lists device information plus the extent and other information on the requested file.

- VSN, ALL - Full VTOC Listing

  Lists device information plus the extent and other information for all files allocated on the requested file.

For the message:

0i? ENTER DVC/VSN, DI, ALL, FILE-ID, END OR EOJ

Key in one of the following, where vsn is the volume serial number of the disk pack to be printed.

For full VTOC listing:

0i vsn,ALL

For a listing of only the device information:

0i vsn,DI

For a listing of up to 44 characters as used on the // LBL card when the file was created.

0i vsn,FILE-ID

To terminate the VTP function:

0i END

To terminate SU:

0i EOJ

NOTES:

1. *The device address of the disk pack may replace the vsn in any of the preceding messages.*

2. *When a VTP request is completed, SU produces the message:*

   0i ENTER DVC/VSN, DI, ALL, FILE-ID, END OR EOJ

   You may specify another volume for VTP display, END to end VTP, or EOJ to terminate SU.

10.6.5.2.3.1. Disk VTP Listing Summary

The various information listed by VTP is either taken directly from the disk labels or calculated from data contained in the labels. A summary of the VTOC information listed by the VTP function is provided in Table 10-6. (Refer to Appendix D in the current version of the data management manual, UP-8068, for details.)

Table 10—6. Summary of Disk VTP Information (Part 1 of 5)

Field Heading	Field Label
VOLUME INFORMATION	
VOLUME SERIAL NUMBER	DL$VSN
VTOC ADDRESS (CCC HHRR)	DL$VTC
VOLUME SECURITY	DL$VSB
OWNER NAME/ADDR CODE	DL$ONR

Table 10—6. Summary of Disk VTP Information (Part 4 of 5)

Field Heading	Field Label
PHYSICAL EXTENT INFORMATION	
EXTENT TYPE	DL$XT1
EXTENT SEQ NO.	DL$SX1
EXTENT START CCC HH	DL$XL1
EXTENT END CCC HH	DL$XU1
CUM CCC HH	(calculated)
AUTO EXTENT INCREMENT	DL$SA1
ISAM FILE INFORMATION	
KEY LENGTH	DL$SLF2
KEY LOCATION	DL$KL1
LAST PRIME DATA REC ID	DL$PID2
FULL OVFL CYL COUNT	DL$NMA2
INDEP OVFL CYL ADDRESS	DL$10F2
PRIME DATA LOAD COUNT	DL$PDLC2
OVERFLOW REC COUNT	DL$NMO2
LAST INDEX REC ADDRESS	DL$BID2
PRIME DATA RECORD COUNT	DL$NMP2
BYTES REQD FOR MAIN STOR	DL$NMS2
DELETED REC COUNT	DL$NMT2
BLOCKS PER CYL COUNT	DL$NMT2+2(2)
PERCENT OVERFLOW	(calculated)
SAT FILE INFORMATION	
DIRECTORY PCA LACE FACTOR	DL$DIRL2
DIRECTORY PCA LACE ADJUST	DL$DIRF2
TEXT PCA LACE FACTOR	DL$TXTL2
TEXT PCA LACE ADJ	DL$TXTF2

Table 10—6. Summary of Disk VTP Information (Part 5 of 5)

Field Heading	Field Label
(M)IRAM FILE INFORMATION	
KEY LOCATION	DL$XILOC
KEY LENGTH	DL$XILOC +2
DUPLICATES ALLOWED	DL$XILOC+3, X'80'
CHANGES ALLOWED	DL$XILOC+3, X'40'
DATA RECORD COUNT	DL$COUTR
RECORD SIZE	DL$DREC
INDEX BUFF SIZE	DL$CSIZ

10.6.5.2.4. Printing the Short Format Volume Table of Contents (SVT)

You use the SVT function to obtain an abbreviated VTOC listing, consisting of a single print line for each physical extent for each file.

Procedure:

For the message:

0i? ENTER DVC OR VSN OR END

Enter disk unit device address or the volume serial number (up to six characters) of the VTOC to be printed; either entry prints the abbreviated VTOC. Enter END to terminate the SVT function.

NOTE:

If SVT is attempted on a data set label diskette, an error message is displayed.

10.6.5.2.4.1. Disk SVT Listing Summary

The various information listed by SVT is either taken directly from the disk labels or calculated from data contained in the labels. A summary of the short format VTOC information listed by the SVT function is provided in Table 10-7. (Refer to Appendix D in the current version of the data management manual, UP-8068, for details.)

Table 10—7. Summary of Disk SVT Information

Field Heading	Field Label
VSN	DL$VSN
SECUR	DL$VSB
OWNER	DL$ONR
FILE-NAME	DL$ID1
FILE TYPE	DL$FT1
FILE SER. NO	DL$FS1
SEQ NO	DL$VS1
CREATION DATE	DL$CD1
EXPIRATION DATE	DL$ED1
OPT COD	DL$OC1
PCA CT	DL$PC1
START CCC HH	DL$XL1
END CCC HH	DL$XU1
CUM CCC HH	(calculated: cumulative)
AI	DL$SA1

10.6.5.2.5. Displaying the Available Disk Extents (AVX)

You use the AVX function to display a list of available disk extents on the system console screen. The display is similar to the listing printed by the VTP function with the DI option; however, output is displayed at system console and printed at the console output printer rather than at the customary line printer.

Procedure:

For the message:

0i? ENTER DEVICE ADDRESS

Enter the disk unit device address of the disk whose available extents are to be displayed.

NOTE:

If AVX is attempted on a data set label diskette, an error message is displayed.

10.6.5.2.5.1. AVX Listing Summary

The various information listed by AVX is either taken directly from the disk labels or calculated from data contained in the labels. A summary of the available disk extent information listed by the AVX function is provided in Table 10-8. (Refer to Appendix D in the current version of the data management manual, UP-8068, for details.)

Table 10—8. Summary of Disk AVX Information

Field Heading	Field Label
VSN	DL$VSN
SECUR	DL$VSB
VTOC ADRS	DL$VTC
OWNR	DL$ONR
START CCC HH	DL$XT5
END CCC HH	DL$XC5 (calculated)
SIZE CCC HH	(calculated)

10.6.5.2.6. Changing the Disk Volume Serial Number (DID)

You use the DID function code to change the VSN of your disk. The DID function will only change the VSN if the device is not allocated to any other job in the system, including SYSRES, SYSRUN, and SYSPOOL.

Procedure:

For the messages:

0i? ENTER DEVICE ADDRESS

Enter the disk unit device address of the disk pack VSN to be changed.

0i? ENTER OLD VSN OF DISK

Enter the old VSN to be changed (up to six characters). The old VSN entered must match the VSN in the disk VOL label; otherwise, the function terminates with an error message.

0i? ENTER NEW VSN OF DISK

Enter the new VSN (up to six characters).

10.6.6. Diskette Functions

Diskette functions are described in 10.6.6.1 through 10.6.6.1.3.

10.6.6.1. Diskette Operating Instructions

Proceed as follows to request one of these functions:

1. Place the subject diskette volume on an available diskette unit.

2. As described in 10.6.2:

   a. Enter the SU symbiont command.

   b. Enter the appropriate function code either as a command parameter or as a solicited message response to the ENTER REQUIRED FUNCTION message.

   c. Enter the spooling option; otherwise, default is Y (applicable if spooling is configured).

10.6.6.1.1. Printing a Diskette in Unblocked Format (DD)

You use the DD function code to print your diskette in character and hexadecimal format. The DD function does not deblock your logical records.

Procedure:

For the messages:

0i? ENTER DVC ADDRESS

Enter the diskette unit device address of the diskette to be displayed. ←

0i? TTRR-BEGIN OR FILE-ID

Enter in decimal the beginning track (TT) and sector (RR) to be displayed, or the file identifier (up to 17 characters) as used on the // LBL job control statement when the file was created. If you enter less than 17 characters, the file ID is padded with blanks on the right. Printing begins at the start of the file when a file ID is entered. ←

0i? TTRR-END

Enter the last track (TT) and sector (RR) in decimal to be displayed.

10.6.6.1.2. Printing the Diskette Volume Table of Contents (VTP)

You use the VTP function code to get a copy of your VTOC. You can print:

- a full VTOC listing, giving you all the device information plus extents and other information for all your allocated files;
- device information only, giving you defective track information, owner-id, number of recording surfaces, and the physical sector length; or
- file information only, giving you the extent and other information regarding the file.

The VTP function code edits and prints the volume table of contents (VTOC) for the requested volume for the requested volume.

Procedure:

Three list options are available, as follows:

- VSN, DI – Device Information Only

  Lists the available free extents and other information on the requested volume.

- VSN, FILE ID. – File Information

  Lists device information plus the extent and other information on the requested file.

- VSN, ALL – Full VTOC Listing

  Lists device information plus the extent and other information for all files allocated on the requested file.

For the message:

0i? ENTER DVC/VSN, DI, ALL, FILE-ID, END OR EOJ

Key in one of the following, where vsn is the volume serial number of the diskette to be printed:

For a full VTOC listing:

0i vsn,ALL

For a listing of only the device information:

0i vsn,DI

For a listing of up to 17 characters as used on the // LBL card when the file was created:

0i vsn,FILE-ID

To terminate the VTP function:

0i END

To terminate SU:

0i EOJ

NOTES:

1. *The device address of the diskette may replace the vsn in the preceding messages.*
2. *When a VTP request is completed, SU reproduces the message:*

   0i ENTER DVC/VSN, DI, ALL, FILE-ID, END OR EOJ

   You may specify another volume for VTP display, END to end VTP, or EOJ to terminate SU.

10.6.6.1.2.1. Diskette VTP Listing Summary

The information listed by VTP is either taken directly from the diskette volume and data set labels or calculated from data contained in the labels. A summary of the VTOC information listed by the VTP function is provided in Table 10–9. (Refer to Appendix D in the current version of the data management manual, UP-8068, for details.)

Table 10—9. Summary of Diskette VTP Information

Field Heading	Byte Position
VOLUME INFORMATION	VOL1 LABEL
VOLUME SERIAL NUMBER	4
OWNER ID	37
NO. RECORDING SURFACES	71
EXTENT ARRANGEMENT CONSTRAINTS	73 C'P' = YES, else NO
PHYSICAL SECTOR LENGTH	75 C'1' = 256, else 128
DEFECTIVE TRACKS	6, 10 of ERMAP record
FILE INFORMATION	DATA SET LABEL
FILE IDENT	5
BEGIN (OF EXTENT)	28, 30
END (OF EXTENT)	34, 36
END OF DATA	74, 76
BLOCK LENGTH	22
RECORD LENGTH	53
BYPASS INDICATOR	40
FILE SECURITY	41
WRITE PROTECT	42
EXCHANGE TYPE IND	43
VOL SEQ (UENCE)	45
(FILE) CREATION DATE	47–52
(FILE) EXPIR DATE	66–71

*Not applicable to IRAM/MIRAM files on PCA basis.

10.6.6.1.3. Changing the Diskette Volume Serial Number (DID)

You use the DID function code to change the VSN of your diskette. The DID function changes the VSN only if the device is not allocated to any other job in the system, including SYSRES, SYSRUN, and SYSPOOL.

Procedure:

For the messages:

0i? ENTER DEVICE ADDRESS

Enter the diskette unit device address of the diskette VSN to be changed.

0i? ENTER OLD VSN OF DISK

Enter the old VSN to be changed (up to six characters). The old VSN entered must match the VSN in the diskette VOL label; otherwise, the function terminates with an error message.

0i? ENTER NEW VSN OF DISK

Enter the new VSN (up to six characters).

10.7. INTEGRATED COMMUNICATIONS ACCESS METHOD (ICAM)

10.7.1. General

The integrated communications access method (ICAM) is an extension of the supervisor (a symbiont) that handles data communications tasks. At system generation time, the ICAM symbionts are tailored to each user's requirements. Each symbiont may contain multiple network definitions (CAAs), and each CCA can handle one or more communications lines. One or more ICAM symbionts can be configured during SYSGEN. Each symbiont satisfies specific communications network requirements; or a single ICAM symbiont can be configured to satisfy all communications requirements. The operator must load the appropriate ICAM symbiont before the programs requiring it can execute or before interactive services can start for terminals. In addition, when interactive global networks or interactive services are required, the operator must initiate the running of the global user service task (GUST) after loading ICAM. The ICAM symbiont remains in main storage until the last program or interactive service terminal session is completed and GUST is shut down. Then ICAM shuts itself down unless the system operator loaded ICAM with a KEEP operand. In this case, ICAM must be terminated with a CANCEL command.

The following subsections describe how to load ICAM, change the ICAM name established during SYSGEN, run and terminate the global user service task, and use messages to control the active communications environment.

10.7.2. ICAM Operator Commands

10.7.2.1. Loading the ICAM Symbiont (Cn/Mn)

The Cn/Mn command loads the ICAM symbiont. The symbiont name (C1-C9 or M1-M9) you specify in the command is normally assigned during SYSGEN. The Cn/Mn command brings in the symbiont specified to handle the required communications network functions.

Format:

$$\left\{\begin{matrix}\mathbf{Cn}\\ \mathbf{Mn}\end{matrix}\right\}$$

where:

$\left\{\begin{matrix}Cn\\Mn\end{matrix}\right\}$

Specifies the name of the required ICAM symbiont, where n is 1 to 9.

When ICAM is successfuly loaded, the output message is displayed on the console screen:

ICAM READY

10.7.2.2. Changing the ICAM Name (SET IC)

The SET IC command changes the name of the ICAM symbiont (C1-C9, M1-M9) that is loaded if remote batch output has output ready and ICAM is not loaded. C? causes the system to ask the operator to supply the symbiont name to be used the next time output is ready and ICAM is not loaded.

Format:

$\underline{SET}\ IC,\left\{\begin{matrix}Cn\\Mn\\C?\end{matrix}\right\}$

Positional Parameter 1:

IC

Specifies the SPOOLICAM SYSGEN parameter is to change.

Positional Parameter 2:

$\left\{\begin{matrix}Cn\\Mn\\C?\end{matrix}\right\}$

Specifies the ICAM name to use (C1-C9 or M1-M9) or specifies that you be asked for a valid ICAM name if ICAM is not currently loaded to send remote batch output (C?).

10.7.3. Initializing and Terminating the Global User Service Task

You must initialize the global user service task before starting interactive services for terminals or before executing user programs requiring global networks. You initiate the running of the job that executes the global user service task program ML$$GI through a system console command entry. See your system administrator for the name of the GUST job to initiate. When global network processing is no longer required for interactive services at terminals or for user programs, you enter an unsolicited message to shut GUST down.

10.7.3.1. Running the Global User Service Task Job

The RUN command is entered with the global user service task job name to initiate ML$$GI program execution.

Format:

```
RUN jobname
```

where:

jobname
Is the name of the prefiled job control stream to run for executing the global user service task ML$$GI program.

10.7.3.2. ML$$GI Program Operator Messages

When you execute ML$$GI, it sends the following messages to the system console to obtain the information it requires to initialize the global network. Respond to each message as indicated. (Only those messages pertaining to GUST initialization are presented here. See the current version of the system messages programmer/operator reference, UP-8076, for all global user service task messages.)

For the message:

MC#420 ENTER NETREQ: CCANAME, PASSWORD, LINE REQUEST

Enter the name, password, and line information associated with the global network to be initialized. When multinode global networks are used, ML$$GI automatically requests all virtual lines.

where:

CCANAME

Is the name of the global network to be initialized. This name must be the same as the label of the CCA macroinstruction that begins the network definition for the global network.

PASSWORD

Is a 1- to 8-character password.

LINE REQUEST

Specifies the lines for which line requests are to be issued. Response may be one of the following:

ALL

Issue line requests for all lines defined in the global network.

line-1,...,line-n

Specifies the lines for which line requests are to be issued. Lines specified must be identical to those specified in label fields of related LINE macroinstructions.

*

Issue line requests for the lines specified in forthcoming message MC#421.

For the message:

MC#421 ENTER LNEREQS: LINE-1, LINE-2,...* OR BLANK

Enter the lines for which line requests are to be issued. This message is displayed only when you respond to the MC#420 message with an asterisk for the line request.

where:

LINE-1, LINE-2,...line-n
Specifies the lines for which line requests are to be issued. Lines specified must be identical to those specified in label fields of related LINE macroinstructions.

*OR BLANK
Redisplays this message to permit additional line requests to be entered.

For the message:

MC#433 RESTART DESIRED FOR NETWORK ---? ANS Y OR N

Enter Y or N to set the RESTART flag for use when a NETREQ macroinstruction is issued.

where:

Y
Sets a restart flag during global network initialization to recover previous network messages from existing disk files.

N
Specifies previous network messages are not recovered during global network initialization.

For the message:

MC#430 GUST ACTIVE FOR CCA network-name

This specifies the global user service task (GUST) initialization is complete for the network named. No response is required.

10.7.3.3. Terminating the Global User Service Task

The following unsolicited message ends global network processing by cancelling the global user service task job.

Format:

```
ØØ {Cn} GU S,network-name,jj
   {Mn}
```

where:

{Cn / Mn}
Specifies the currently loaded ICAM symbiont name (C1–C9 or M1–M9).

GU
Specifies this message is for GUST.

S
Specifies a shutdown is required.

network-name
Is the 4-character name of the active global network. This name must be the same as the label of the CCA macroinstruction in the global network definition. If the name is less than four characters, pad it with blanks on the right.

jj
Is the GUST job number (01–07).

If the shutdown request is accepted, the following message is displayed when the global user service task job is cancelled:

```
MC#401 GUST SHUTDOWN COMPLETE
```

NOTE:

To resume communications processing with global networks or to start interactive services for terminals, the operator procedures for loading ICAM and initializing the global user service task must be repeated. You should never cancel the GUST job by means of the CANCEL operator command.

10.7.4. ICAM Operator Messages

On occasion, you enter unsolicited messages to ICAM to facilitate processing. These messages have the following format:

```
ØØ {Cn} cc f,{xxxx},jj
   {Mn}       {xx  }
```

where:

{Cn / Mn}
Specifies the name of the required ICAM symbiont.

cc
Is a 2-character command code.

f
Is a 1-character facility type (L=line, P=port, T=terminal).

xxxx
Is a 1- to 4-character name of line or terminal as defined in the label field of a LINE or TERM macroinstruction. If the name is less than four characters, pad it with blanks on the right.

xx
Is the 2-character port number on the communications adapter or the SLCA number.

jj
Is a 2-digit job number (01–07).

Appendix B. Operating Procedures for 9200/9300 Series Subsystem

B.1. GENERAL

This appendix outlines the procedures required for operating the SPERRY UNIVAC 9200/9300 Series Subsystem (Figure B-1) online with the SPERRY UNIVAC 90/25 and 90/30 B Systems under control of Operating System/3 (OS/3). For complete details about the 9200/9300 subsystem, see the 9200/9300 series processor and storage operator reference, UP-7781 (current version).

The 9200/9300 subsystem is connected to the 90/25 or 90/30 B system by means of a 9000 channel adapter attached to multiplexer channels on both units. When operated online with the 90/25 or 90/30 B system, the 9200/9300 acts as an I/O controller through which the integrated printer, card reader, and card punch are used as 90/25 or 90/30 B I/O devices. Other 9200/9300 devices and the read/punch feature on the integrated card punch are not supported. To perform the function of an I/O controller, the 9200/9300 uses a highly modified MOS supervisor, which is provided as a deck of punched cards.

OS/3 interfaces with the 9200/9300 I/O devices through a device handler that is configured at system generation time if the keyword parameter TYPE=9200 or TYPE=9300 is specified for any printer, reader, or punch category. OS/3 supports each I/O device independently, regardless of the status or condition of the other 9200/9300 devices.

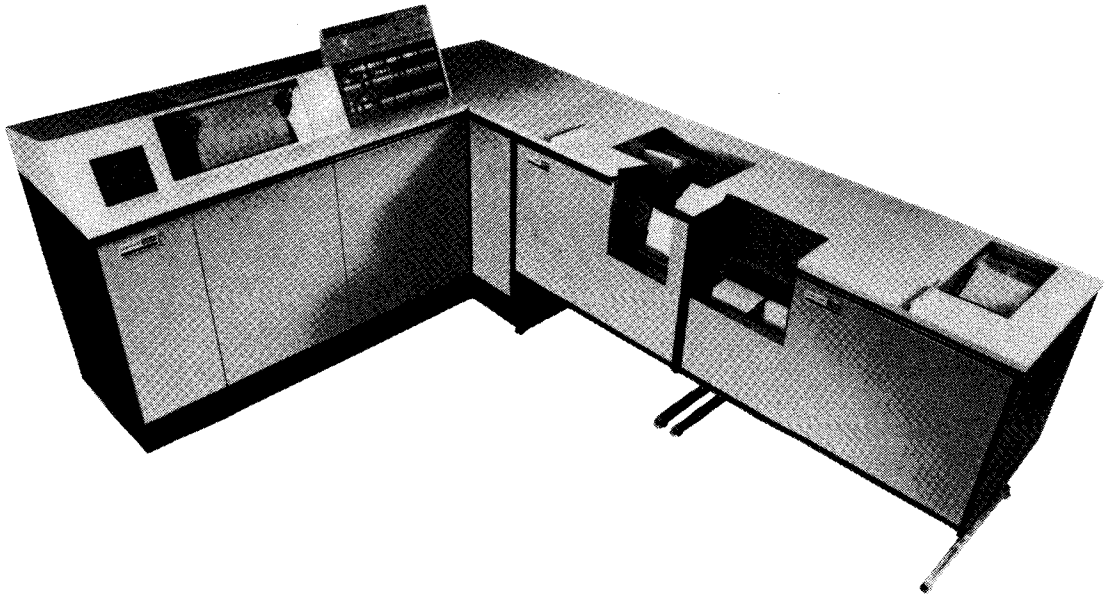


Figure B—1. 9200/9300 Series Subsystem

B.2. OPERATING CONTROLS AND INDICATORS

The control console (Figure B-2) contains the controls and indicators required to:

- perform an initial program load of the 9200/9300 subsystem;
- initialize and reset the printer, reader, and punch;
- produce a main storage dump; and
- receive and respond to error messages.

Controls on the control console are of two types: momentary and 2-position. Momentary controls may be buttons or switches. Momentary switches are pressed on the upper portion; when released, they return to the normal position. Two-position switches are set to either of two positions by pressing the upper or lower portion.

Data is entered by setting the **DATA ENTRY** switches to represent hexadecimal values. Storage locations are indicated by setting the **MEMORY ADDRESS** switches. In both cases the upper portion of ech switch is pressed to designate a bit value of 1 and the lower portion to designate a bit value of 0.

Error messages are displayed in hexadecimal code on the **NEXT INSTRUCTION/HALT DISPLAY** indicators. A lighted indicator designates a bit value of 1; an unlighted indicator, a bit value of 0. Error messages relating to the 9200/9300 devices are also received on the 90/25 or 90/30 B console.

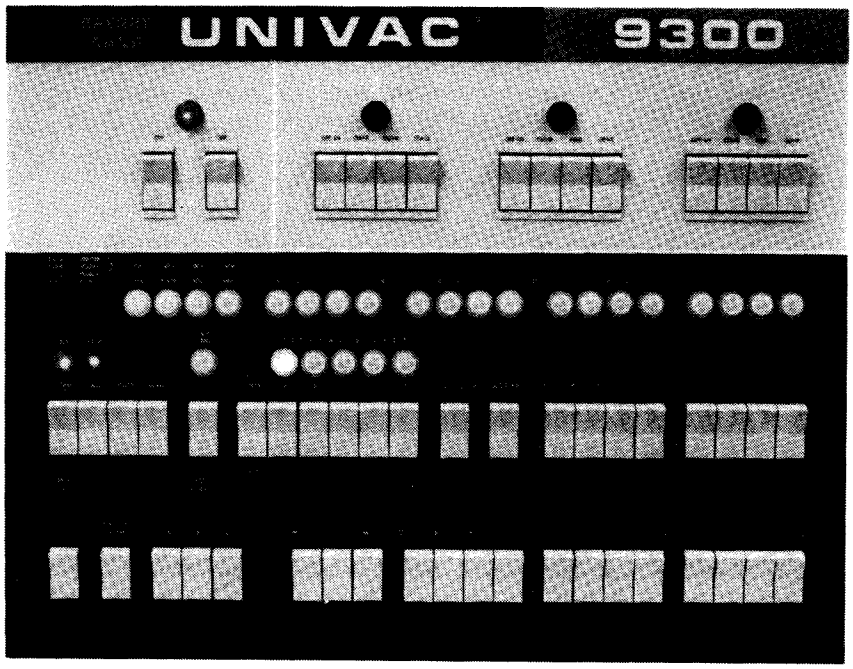


Figure B—2. Control Console, Controls and Indicators

B.3. INITIAL PROGRAM LOAD PROCEDURE

The initial program load for the 9200/9300 supervisor may be performed either before or after the IPL for the 90/25 or 90/30 B system (9.3). The procedure is as follows:

1. Place program deck in the reader input hopper.
2. Press **READER CLEAR** and **READER FEED** switches.
3. Set **DATA ENTRY** switches to X'01'.
4. Press **CLEAR** switch.
5. Press **LOAD** switch.
6. Press **START** switch.
7. Reset **LOAD** switch.
8. Reset **START** switch.

B.4. INITIALIZATION AND RESET PROCEDURES

Initialization and reset procedures are performed by means of the OPERATOR REQUEST function and the low-order (right position) **DATA ENTRY** switches. To perform each procedure, set the indicated hexadecimal value in the **DATA ENTRY** switches and press the **OP REQ** button. The high-order **DATA ENTRY** switches have no significance.

B.4.1. Initializing the Printer

Before attempting to operate the printer, be certain that the printer form control tape meets the specifications of the vertical format buffer being used. Procedures for preparing the printer form control tape are detailed in the 9200/9300 series processor and storage operator reference, UP-7781 (current version).

To initialize the vertical format buffer, enter the value X'3' and press the **OP REQ** button. This sets the VFB to the home-paper position but does *not* move the forms.

To initialize the VFB and move the forms to the home paper position at the same time, enter the value X'B' and press the **OP REQ** button.

B.4.2. Initializing the Card Reader

To initialize the card reader, enter the value X'9' and press the **OP REQ** button. Two cards are read and any existing errors or images in the card reader buffers are discarded. Be sure to place at least two blank cards at the end of each job deck.

Although it is required only that a card be in the card reader wait station before attempting to read cards, it is advisable to reinitialize the card reader between jobs to ensure that residual information is not inadvertently carried over from job to job.

When this procedure is performed, the channel adapter is also initialized (B.4.3).

B.4.3. Initializing the Channel Adapter

To initialize the channel adapter, enter the value X'C' and press the **OP REQ** button. An acknowledge message containing the current status of all the 9200/9300 I/O devices is transmitted to OS/3. This can be helpful when misoperation has stalled the system.

B.4.4. Obtaining a Storage Dump and Reinitializing the System

When the 9200/9300 subsystem has failed as a result of stalling, power failure, or other reason, the following procedure may be performed to obtain a storage dump and/or restart the input/output operation.

1. Do either of the following:

   a. Enter the value X'F' and press the **OP REQ** button.

   b. Press the **CLEAR** and **RUN** controls.

2. The **NEXT INSTRUCTION/HALT DISPLAY** indicators display the value X'1FFF'.

3. Set the low-order **MEMORY ADDRESS** switches to location 4 (X'4').

4. Key in a value in the **DATA ENTRY** switches and press the **RUN** button:

   a. If 0 is entered, the system will proceed from where it was prior to the interruption.

   b. If 1 is entered, a storage dump is printed. At the conclusion of the dump, the **HALT DISPLAY** of '1FFF' is repeated, and one of the three keyins must be made again.

   c. If any value other than 0 or 1 is entered, the system is reinitialized. It goes into an idle loop and is in the same condition as after a successful IPL.

B.5. RECOVERY PROCEDURES

Error conditions are communicated to the operator through messages on the 90/25 or 90/30 B console or halt displays on the 9200/9300 control panel.

B.5.1. 90/25 and 90/30 B Console Messages

All console messages pertaining to the 9200/9300 devices are in the standard device, status, sense format. There are three basic messages:

1. ji? DEVICE=14x STATUS=00FF SENSE=0000 9300-NAK RU*C

2. ji? DEVICE=14x STATUS=00FF SENSE=0000 9300-CLEAR RU*C

3. ji? DEVICE=14x STATUS=0200 SENSE=ss00 9300-6xss RU*C

where:

j
Is the job number

i
Is the message number

x
Is the device number:

1 = card reader

2 = card punch

3 = printer

ss
Is the device status.

RU*C
Indicates the allowable operator responses:

R = retry

U = unrecoverable - return control to issuing program

C = cancel user program

Message 1 (9300-NAK) indicates that an I/O order was accepted by the channel adapter but has not been executed within 20 seconds thereafter. Probably the 9200/9300 subsystem is stopped. If so, press **RUN** on the 9200/9300, and respond R on the console.

Message 2 (9300-CLEAR) indicates that a retry on message 1 has also timed out. If the 9200/9300 is running, but idle, try reinitializing (B.4.4) and respond R to this message.

Message 3 (9300-6xss) indicates that one of the 9200/9300 devices has developed an unrecoverable error. Correct the condition as described in Table B-1 and respond R, U, or C to the message.

Table B—1. 90/25 and 90/30 B Console Messages for the 9200/9300 Series Subsystem (Part 1 of 2)

Error Code	Module	Condition	Operator Action
6108	Reader	Multistrobe check error	Place the last card in the output stacker and the card in the wait station on the bottom of the input deck. Feed one card; then press **READER CLEAR.**
6140	Reader	Hopper is empty or stacker is full.	Correct the condition and press **READER CLEAR.**
6140	Reader	Misfeed	If there is a card in the wait station, place it on the bottom of the input deck; feed a card and press **READER CLEAR.**
6180	Reader	Card jam or photocell check	See operator action for 6108 error code.

Table B—1. 90/25 and 90/30 B Console Messages for the 9200/9300 Series Subsystem (Part 2 of 2)

Error Code	Module	Condition	Operator Action
6202	Punch	Hopper is empty or stacker is full.	Correct the condition and press **PUNCH CLEAR.**
6220	Punch	Punch check error	Press **PUNCH CLEAR.**
6280	Punch	Interlock check, misfeed, stacker jam, punch entry, or exit check	Correct the condition and press **PUNCH CLEAR.**
6301	Printer	Low paper supply	Correct the condition.
6308	Printer	Wrong print bar setting	Insert the correct bar or reset the bar switch.
6320	Printer	Storage overload	No action required.
6340	Printer	Skip code cannot be found on the paper loop.	Install the correct paper loop and press **PRINTER CLEAR.**
6340	Printer	Skip code cannot be found in the VFB.	NO recovery is possible. Either the VFB has been destroyed or an incorrect skip has been issued.
6380	Printer	Abnormal condition on the printer	Correct the condition and press **PRINTER CLEAR.** An extra line may print or a print line may be missing.

B.5.2. 9200/9300 Halt Displays

Error conditions may be indicated by hexadecimal displays on the **NEXT INSTRUCTION/HALT DISPLAY** indicators on the 9200/9300 control panel. Halt displays, their causes, and recovery procedures are listed in Table B—2.

Table B—2. 9200/9300 Control Panel Halt Displays (Part 1 of 2)

Hexadecimal Display	Module	Cause	Operator Action
03ss	Printer	Error during VFB initialization and "home paper" (OP REQ X'B')	Follow procedure in Table B—1 for comparable 63ss message.
12FF	Channel adapter	Illogical command sequence between 90/30 and 9200/9300	Press **CLEAR** and **RUN** to dump storage.
12ss	Channel adapter	Error on I/O command to the channel adapter	Press **RUN** to retry. Press **CLEAR** and **RUN** to reinitialize or dump storage.
1FFF	Storage dump	See B.4.4.	See B.4.4.
4300	Loader	Card count discrepancy	Repeat IPL procedure.
6100	Loader	Hole-count check	Repeat IPL procedure.
6100	Reader	Illogical sequence	Press **CLEAR** and **RUN** to dump storage.

Appendix C. Supervisor Modification Procedure

For special processing requirements, your system administrator may tell you to modify the selections made for the supervisor during SYSGEN. Respond with Y to the MODIFY SUPERVISOR? IPL statement, then proceed with the appropriate operator action described in Tables C-1 and C-2.

NOTE:

For a description of system output messages and how to respond to them (via solicited input messages), see Section 10.

Table C—1. How to Modify List of Resident Shared Code Modules (Part 1 of 3)

IPL Output Message	Operator Response
0i? IS THE SHARED CODE DIRECTORY INDEX TO BE BUILT? (Y,N)	0i Y Specifies an index is to be built for the shared code library. The system will do a binary search on the index rather than a serial directory search. This allows faster loading of shared code modules into main storage and faster processing between two or more shared code modules. The directory itself uses approximately 4000 bytes of main storage. 0i N No shared code directory index is to be built.
0i? ANY RESIDENT SHARED MODULES TO ADD OR DELETE? (Y,N)	0i N Terminates output messages for modifying shared code module list. Specifies no modification is required to SYSGEN list of shared code modules to be made resident. Proceed with operator action described in Table C-2. 0i Y Specifies SYSGEN list of shared code modules requires modification. Output message requesting a function is displayed.

Table C—1. How to Modify List of Resident Shared Code Modules (Part 2 of 3)

IPL Output Message	Operator Response
0i? FUNCTION? (ADD,DEL,LIST,END,HELP)	0i END Terminates output messages for modifying shared code module list. Specifies no further supervisor modification is required. The SYSGEN list of shared code modules is updated by the changes previously specified with the ADD and DEL funcitons. Modules in updated list are loaded and made resident until next IPL. Proceed with operator action described in Table C-2. 0i HELP Displays brief description of the ADD, DEL, LIST, and END functions, followed by an output message requesting another function. 0i LIST Displays a list of the individual shared code modules currently specified to be made resident, followed by an output message requesting another function. 0i {ADD / DEL} Specifies an individual module or group of modules is to be added to or deleted (DEL) from the current list of shared code modules. Displays the following list of shared code module groups, followed by an output message requesting a group of module names. POSSIBLE SYSTEM GROUPS ARE: SF - SCREEN FORMAT SERVICES ISB - BASIC INTERACTIVE SERVICES ISF - FULL INTERACTIVE SERVICES EDT - EDITOR DP - DIALOGUE PROCESSOR RPG - RPG EDITOR DDPR - REMOTE DISTRIBUTED DATA PROCESSING DDPL - LOCAL DISTRIBUTED DATA PROCESSING ESC - ESCORT ESCF - ESCORT FULL BAS - BASIC

Table C—1. How to Modify List of Resident Shared Code Modules (Part 3 of 3)

IPL Output Message	Operator Response
0i? WHICH GROUP TO {ADD? / DELETE?}	0i {group-name / module-name} [,L] Specifies the group of shared code modules or the individual shared code module to be added or deleted. Provides option to display (L) individual module names as they are added or deleted from the current list. After processing the additon or deletion, an output message requests another function. Run the job SCLIST to get a list of all system shared-code modules filed in YSCLOD.

Table C—2. How to Modify Buffer Sizes Used by Dynamic Buffer Management

IPL Output Message	Operator Response
Oi? ANY CHANGES TO DYNAMIC BUFFER MANAGEMENT PARAMETERS? (Y,N)	Oi N Terminates output messages for modifying buffer sizes. Specifies no changes are required to the expansion region and resident buffer sizes specified at SYSGEN. Proceed with step m of the interactive IPL procedure (Section 9). Oi Y Specifies SYSGEN-specified expansion region and resident buffer sizes require modification. Output message requesting new expansion region size is displayed.
Oi? ENTER NEW EXPANSION REGION SIZE? (CURRENTLY decimal-byte-size)	Oi new-size Specifies new size*, in decimal bytes, of expansion region and terminates output messages for modifying buffer sizes. Expansion region size is changed until next IPL. Proceed with step m of the interactive IPL procedure (Section 9). Oi O Specifies no expansion regions are to be allocated until the next IPL. Dynamic buffer management must use resident buffer pool. Output message requesting new resident buffer size is displayed.
Oi? ENTER NEW RESIDENT BUFFER SIZE? (CURRENTLY decimal-byte-size)	Oi new-size Specifies new size*, in decimal bytes, of resident buffer pool and terminates output messages for modifying buffer sizes. Resident buffer size is changed until next IPL. Proceed with step m of the interactive IPL procedure (Section 9).

*The new decimal size can be specified with or without a comma, or as a multiple of K. For example:

200000	is 200,000 decimal bytes
200,000	is 200,000 decimal bytes
200K	is 200 x 1024 = 204,800 decimal bytes

Index

T

SPERRY UNIVAC

USER COMMENT SHEET

Your comments concerning this document will be welcomed by Sperry Univac for use in improving subsequent editions.

Please note: This form is not intended to be used as an order blank.

_____

(Document Title)

_____ _____ _____

(Document No.) *(Revision No.)* *(Update No.)*

Comments:

Cut along line.

From:

_____

(Name of User)

_____

(Business Address)

Fold on dotted lines, and mail. (No postage stamp is necessary if mailed in the U.S.A.)
Thank you for your cooperation

FOLD

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 21 BLUE BELL, PA.

POSTAGE WILL BE PAID BY ADDRESSEE

SPERRY UNIVAC

ATTN.: SYSTEMS PUBLICATIONS

P.O. BOX 500
BLUE BELL, PENNSYLVANIA 19424

CUT

FOLD