

Contents

PREFACE

CONTENTS

1. NEW FEATURES

1.1.	SUBBATCH BALANCING	1-3
1.2.	DISK DATA CACHE	1-3
1.3.	FORMATTED READ FACILITY	1-4
1.4.	NONCONTIGUOUS FIELD SORT	1-4
1.5.	ICL 1900 VARIABLE LENGTH RECORDS UNDER REPORT WRITER	1-5
1.6.	NEW EXTENDED MEMORY TWAS	1-5
1.7.	KEYSTROKE-BUFFER EXPANDED	1-5
1.8.	DISK FULL AT 98 PERCENT	1-6
1.9.	DATA ENTRY PRIORITY SELECTION	1-6
1.10.	AUTO CHECKBOX FUNCTIONALITY CORRECTION	1-6
1.11.	IBM VARIABLE LENGTH RECORDS	1-7
1.12.	INTERCOM	1-7
1.13.	DEFAULT FORMAT=ANY FOR SV WRITE AND SV PRINT	1-7
1.14.	NO TOP OF FORM ON PRINTER OPEN	1-8
1.15.	TEMP FILE DELETE IN STARTUP	1-8
1.16.	LOG DISK ERRORS	1-8
1.17.	DS COMPILE WITH RECORD NUMBERS	1-10
1.18.	PRINT COMPILE WITH PAGE EJECT	1-10
1.19.	SET TIME AND DATE OPTION	1-11
1.20.	RUNTIME 6-CHARACTER TIME	1-11
1.21.	BK LOG MESSAGE EXPANSION	1-11
1.22.	"ENTER MESSAGE" REMOVED	1-12
1.23.	REBROADCAST BACKGROUND MESSAGES	1-12
1.24.	BACKGROUND MESSAGE REPORT	1-12
1.25.	U100 DATACOM	1-13
1.26.	SFTW DIAG-MODIFICATIONS	1-13
1.27.	ANY DATA KEY TO CONTINUE	1-14
1.28.	SEARCH BACKWARD BY NUMBER	1-14
1.29.	FREEFORM SEARCH	1-14
1.30.	WRITE CONTINUE—TOTAL RECORD COUNT	1-15
1.31.	DISPLAY-ONLY RECORDS	1-15
1.32.	"FORMAT NOT FOUND" MESSAGE	1-16
1.33.	SINGLE RECORD REPLAY	1-16
1.34.	DOUBLE CHECKPOINT	1-17
1.35.	WARM START ENHANCEMENTS	1-17
1.35.1.	WARM START Speedup	1-17
1.35.2.	WARM START Fast Recovery	1-17
1.36.	LINK BACK TO CHECKPOINT	1-18

- 1.37. BREAK UP LONG COMMUNICATION RECORDS 1-18
- 1.38. DELETE COMMAND DEFAULT CHANGE 1-18
- 1.39. THREE-CYCLE VERIFY OPTION 1-18
- 1.40. MESSAGE QUEUING 1-19
- 1.41. MAPPED TWA SPACE 1-20
- 1.42. EXTENDED MEMORY OPTION REMOVED 1-20
- 1.43. WRITE CONTINUE FOR H6000 1-20
- 1.44. MONITOR ENHANCEMENT 1-21
- 1.45. CORRECTION COUNT OPTION 1-21
- 1.46. DISK RECORD SIZE IN DETAILED STATUS 1-21
- 1.47. SEQUENTIAL I/O ACCESS TO SPOOL FILES 1-21
- 1.48. LOG FIRST RECORD OF DATACOM FILES 1-21
- 1.49. INDIO MODULES UNLOADED AT POWEROFF 1-22
- 1.50. DATA COMMUNICATIONS ENHANCEMENTS 1-22
- 1.50.1. Directory Construction at Transmission Time 1-22
- 1.50.2. Reformat of Data at Transmission Time 1-22
- 1.51. LARGE CAPACITY DISK SUPPORT 1-23
- 1.52. IMPROVED DISK UTILIZATION PROCEDURES 1-23
- 1.53. UNIQUE USER'S PROGRAM LIBRARY BATCH NUMBERS 1-24

2. COMPATIBILITY

- 2.1. COMPATIBILITY WITH PREVIOUS LEVELS 2-1
- 2.1.1. Batch Compatibility 2-1
- 2.1.2. Format Program Compatibility 2-1
- 2.1.3. Autoprompt Batch Compatibility 2-1
- 2.1.4. Patch Compatibility 2-2
- 2.2. RELATED PRODUCTS 2-2
- 2.2.1. Products Executed to Construct This Product 2-2
- 2.2.2. Products Executed With This Product 2-3
- 2.2.3. Products Executed With a Program Produced
by This Product 2-3
- 2.3. LONG-TERM COMPATIBILITY CONSIDERATIONS 2-3

3. RESTRICTIONS

4. RELEASE MEDIA DESCRIPTION

5. CORRECTIONS

6. SUPPORT

- 6.1. RELEASE HISTORY 6-1
- 6.2. RESOURCE UTILIZATION 6-1
- 6.2.1. Memory 6-1
- 6.2.2. Disk 6-5
- 6.2.3. CPU 6-5
- 6.3. PROBLEM REPORTING PROCEDURES 6-6
- 6.3.1. Reporting the Error 6-6
- 6.3.2. Priorities 6-6
- 6.3.3. Reviewing the Error 6-7

APPENDIXES

A. OPERATIONAL CONSIDERATION

A.1.	1900/10 USERS	A-1
A.1.1.	DELETE	A-1
A.1.2.	DISK MGMT	A-1
A.1.3.	PAGE TWO	A-2
A.1.4.	SET	A-2
A.1.5.	WRITE and PRINT	A-3
A.2.	XL/40 USERS	A-3
A.2.1.	BUILD DIR	A-4
A.2.2.	DELETE	A-4
A.2.3.	DISK MGMT	A-5
A.2.4.	SET	A-6
A.2.5.	WRITE and PRINT	A-9

B. KNOWN SITUATIONS

C. USER'S PROGRAM LIBRARY

C.1.	THE LIBRARY	C-1
C.1.1.	Loading the Program Library	C-1
C.1.2.	Loading Specific Library Programs	C-2
C.1.3.	Using the Program Library	C-4
C.2.	SOURCE LISTINGS	C-4
C.3.	FORMAT PROGRAMS	C-6
C.3.1.	AUPROMPT	C-8
C.3.2.	DATATYPE	C-8
C.3.3.	DATECONV	C-10
C.3.4.	DK-CONV	C-11
C.3.5.	FORMAT	C-12
C.3.6.	FORMATI	C-17
C.3.7.	FORMATIS	C-19
C.3.8.	FREEFORM	C-21
C.3.9.	F80, F128, F133, and F134	C-22
C.3.10.	INDEX	C-23
C.3.11.	LABEL and LABEL-TR	C-24
C.3.12.	MEM11	C-26
C.3.13.	NC-SORT	C-33
C.3.14.	SORT	C-37
C.3.15.	SUBPROG	C-39
C.3.16.	TABLE	C-40
C.3.17.	TRANSLAT	C-42
C.3.18.	VFU-FORM	C-43
C.4.	SUBPROGRAMS	C-44
C.4.1.	CALENDAR	C-45
C.4.2.	CHKDG-07	C-46
C.4.3.	CHKDG-10	C-48

C.4.4.	CHKDG-11	C-50
C.4.5.	DEC2HEX	C-52
C.4.6.	EDIT	C-53
C.4.7.	HEX-CONV	C-58
C.4.8.	ICL-NEG	C-59
C.4.9.	JULIAN	C-60
C.4.10.	MEM11S	C-62
C.5.	TABLE PROGRAMS	C-66
C.5.1.	DEV-CODE	C-67
C.5.2.	FISTATUS	C-69
C.5.3.	FMSTATUS	C-70
C.5.4.	HEX2DEC	C-70
C.5.5.	HW-TAB	C-71
C.5.6.	ICL-TAB	C-71
C.5.7.	NTBL	C-72
C.5.8.	ST-TABLE	C-72
C.5.9.	STATE-TB	C-73
C.5.10.	TESTTAB	C-73
C.5.11.	TYPTBL	C-74
C.5.12.	YNTBL	C-74
C.5.13.	YTBL	C-75
C.6.	LABEL PROGRAMS	C-75
C.6.1.	HW6000SL	C-76
C.6.2.	IBMCR	C-78
C.6.3.	IBMCRV	C-79
C.6.4.	IBMDL	C-80
C.6.5.	IBMDLV	C-81
C.6.6.	IBMFD	C-83
C.6.7.	IBMNL	C-85
C.6.8.	IBMSL	C-86
C.6.9.	IBMSLV	C-88
C.6.10.	ICL-LBL	C-90
C.6.11.	ICL29CR	C-92
C.6.12.	ICL29SL	C-94

D. SOFTWARE DIAGNOSTICS

D.1.	STARTUP	D-1
D.2.	COMMAND SYNTAX	D-2
D.3.	LOGICAL UNIT NUMBERING	D-3
D.4.	SUGGESTED OPERATIONAL PROCEDURES	D-3
D.5.	COMMANDS	D-4
D.5.1.	Command Quick Reference	D-4
D.5.2.	BSF	D-6
D.5.3.	BSR	D-6
D.5.4.	COPY	D-6
D.5.5.	DIFFD	D-7
D.5.6.	DIFFM	D-9
D.5.7.	DUMPM	D-11

D.5.8.	DV	D-14
D.5.9.	EOF	D-15
D.5.10.	HEADER	D-15
D.5.11.	HELP	D-16
D.5.12.	INPUT	D-16
D.5.13.	LIST	D-17
D.5.14.	LPP	D-20
D.5.15.	MAKEMD	D-20
D.5.16.	NOSIM	D-21
D.5.17.	OUTPUT	D-21
D.5.18.	P	D-22
D.5.19.	PLTH	D-22
D.5.20.	R	D-22
D.5.21.	RECOVER	D-23
D.5.22.	RESTORE	D-24
D.5.23.	REW	D-25
D.5.24.	SAVE	D-26
D.5.25.	SIM	D-27
D.5.26.	SKE	D-28
D.5.27.	VTOC	D-29
D.5.28.	WPL	D-31
D.5.29.	X	D-31

E. SYSTEM GENERATION (SYSGEN)

F. MESSAGES

F.1.	ERROR MESSAGES	F-1
F.2.	MESSAGE TYPES	F-1
F.3.	ERROR TYPES	F-1
F.3.1.	Keying Errors	F-1
F.3.2.	Field Check Errors	F-1
F.3.3.	Format Programming Errors	F-2
F.3.4.	Data Communication Errors	F-3
F.3.5.	System Errors	F-3
F.3.6.	Task Errors	F-4
F.4.	RESPONSE TO MESSAGES	F-4
F.4.1.	RESET Key Response	F-4
F.4.2.	Keyed Response Required	F-5
F.5.	GENERAL MESSAGES	F-8
F.6.	SYSTEM ERROR MESSAGES	F-50
F.6.1.	Disk Error Recovery	F-50
F.6.2.	System Errors	F-51
F.7.	TASK ERROR MESSAGES	F-65
F.7.1.	Task Error Recovery	F-65
F.7.2.	Task Errors	F-65
F.8.	FILE STATUS CODES	F-71
F.8.1.	File Management Status Codes	F-72
F.8.2.	Host Interactivity Status Codes	F-75

F.9.	SYSGEN ERROR MESSAGES	F-76
F.9.1.	SYSGEN Errors	F-76
F.9.2.	IOCTL Errors	F-79
F.10	MEMORY DUMP PROCEDURES	F-81
F.10.1	System Error Dump Procedure	F-81
F.10.2.	Alternate Dump Procedure	F-82

G. SUBBATCH BALANCING

G.1.	OPERATING PROCEDURES	G-1
G.1.1.	Customer Use	G-1
G.1.2.	Format Programming	G-3
G.1.3.	SYSGEN Procedures	G-15
G.2.	OPERATIONAL CONSTRAINTS	G-15
G.2.1.	Parameter Limitations	G-15
G.2.2.	Performance Limitations	G-16
G.2.3.	Compatibility	G-16

H. ICL 1900 VLR WITH REPORT WRITER

H.1.	COMPARISON OF REPORT WRITER AND NORMAL WRITE	H-1
H.2.	COMPARISON OF REPORT WRITER AND SEQUENTIAL I/O	H-2
H.3.	PROGRAMMING FOR REPORT WRITER	H-2

I. IBM VLR

I.1.	DESCRIPTION	I-1
I.1.1.	Data Block	I-2
I.1.2.	Block Header	I-2
I.1.3.	Record Header	I-2
I.2.	COMPATIBILITY	I-3
I.3.	OPERATOR INTERFACE	I-3
I.3.1.	IBM VLR Tape Output	I-3
I.3.2.	IBM VLR Tape Input	I-5
I.4.	PROGRAMMING REQUIREMENTS	I-5
I.4.1.	Writing IBM VLR Tapes	I-5
I.4.2.	Reading IBM VLR Tapes	I-6
I.4.3.	General Format Requirements	I-6
I.5.	PROCESSING ERRORS	I-9
I.6.	EXCLUDED CAPABILITIES	I-9

J. U100 DATACOM

J.1.	LOGICAL DEVICE DEFINITION	J-1
J.2.	CONTROLLING DATA COMMUNICATIONS	J-2
J.3.	UNRECOVERABLE LINE ERRORS	J-2
J.4.	KEYSTATION MESSAGES	J-2
J.5.	AUTO ANSWER	J-3
J.6.	EXAMPLES OF 1100 SYSGENS FOR 1900/10 OR XL/40 U100	J-3

K. FIELD RELEASE MASTER BREAK-OUT PROCEDURE

K.1. COPY FIELD RELEASE MASTER WITH ONLY ONE
TAPE UNIT K-1

K.1.1. Copy Software Diagnostics K-1

K.1.2. Copy the SMB K-2

K.2. COPY FIELD RELEASE MASTER WITH TWO TAPE
UNITS K-2

K.2.1. Copy Software Diagnostics K-2

K.2.2. Copy the SMB K-3

L. HARDWARE FEATURES

L.1. 1900/10 AND XL/40 SYSTEMS L-1

L.2. 1900/10 SYSTEM ONLY L-1

L.3. XL/40 SYSTEM ONLY L-2

M. SOFTWARE PATCHING PROCEDURES

M.1. OMT PATCHING AND CORRECTIONS M-1

M.2. CDM PATCHING M-7

M.3. PATCHING ERRORS M-10

INDEX

USER COMMENT SHEET

FIGURES

6-1. Average TWA/Compile Ranges 6-3

C-1. Memory Analysis Display C-29

C-2. Memory Analysis Printed Report C-29

C-3. Disk Record Layout C-30

G-1. Coding for Subbatch Balancing, Level A G-9

G-2. Coding for Subbatch Balancing, Level B G-10

G-3. Coding for Subbatch Balancing, Level C G-11

G-4. Coding for Subbatch Balancing, Level Z G-12

H-1. Example of an ICL 1900 VLR Report
Writer Format H-4

I-1. IBM VLR Tape Output--Option Selection Form I-4

I-2. IBM VLR Tape Output--Packing Type I-4

I-3. IBM VLR Tape Output--Block Size I-4

I-4. Example of an IBM VLR Report Writer Format I-8

J-1. Dedicated Line SYSGEN J-3

J-2. Dial-Up Line SYSGEN J-4

TABLES

1-1.	Disk Function Codes	1-9
1-2.	Disk Status Codes	1-10
6-1.	Supervisor Command Memory Usage	6-2
6-2.	Preinstallation Planning	6-4
C-1.	Data Communication Device Control Codes	C-69
F-1.	IOCTL Error Coding	F-80

1. New Features

The following features are new to either the 1900/10 or the XL/40, or to both systems:

- Sub-batch Balancing
- Disk Data Cache
- Formatted Read Facility
- Noncontiguous Field Sort
- ICL 1900 Variable Length Records under Report Writer
- New Extended Memory TWAs
- Keystroke Buffer Expanded
- Disk Full at 98 Percent
- Data Entry Priority Selection
- AUTO Checkbox Functionality Correction
- IBM Variable Length Records
- Intercom
- Default FORMAT = ANY for SV WRITE and SV PRINT
- No Top of Form on Printer OPEN
- Temp File Delete in STARTUP
- Log Disk Errors
- DS COMPILE with Record Numbers
- Print Compile with Page Ejects
- SET Time and Date Options
- Runtime 6-Character Time
- BK LOG Message Expansion
- "ENTER MESSAGE" Removed

Rebroadcast Background Messages
Background Message Report
U100 Datacom
SFTW DIAG Modifications
Any Data Key to Continue
Search Backward By Number
Freeform SEARCH
WRITE CONTINUE -- Total Record Count
Display-Only Records
"FORMAT NOT FOUND" Message
Single Record Replay
Double Checkpoint
WARM START Enhancements
Link Back to Checkpoint
Break Up Long Communication Records
DELETE Command Default Changed
Three-Cycle VERIFY Option
Message Queuing
Mapped TWA Space
Extended Memory Option Removed
WRITE CONTINUE for H6000
MONITOR Enhancement
Correction Count Option
Disk Record Size in Detailed Status

Sequential I/O Access to Spool Files
Log First Record of DCM Files
INDIO Modules Unloaded at POWEROFF
Data Communications Enhancements
Large Capacity Disk Support
Improved Disk Utilization Procedures
Unique User's Program Library Batch Numbers

1.1. SUBBATCH BALANCING

Subbatch balancing enables the performance of multiple-balance operations within a single data batch. Several subbatches containing logically corresponding documents may be balanced without the need to create separate data batches. This reduces the overhead involved in opening and closing files; also, the number of disk batches is reduced.

Refer to Appendix G for detailed information concerning the use of this feature.

1.2. DISK DATA CACHE

The Disk Data Cache (DDC) option has been added to improve disk I/O throughput. The DDC maintains a pool of one or more Standard Memory (SM) buffers. Each buffer is large enough to contain one disk AU (1344 bytes).

All disk I/O requests are monitored by the DDC at the physical I/O level. The DDC dynamically selects frequently referenced AUs of data to retain in the cache. Whenever possible, it services READ/WRITE requests from the cache instead of from the disk.

The DDC option is user-selectable both at SYSGEN time and through the Supervisor menu. If the DDC option was selected during SYSGEN, the DDC will automatically be loaded as part of system startup. The supervisor has the option to load/unload the DDC, enable/disable DDC write, and change number of buffers.

The degree of improvement in disk I/O throughput produced by using DDC depends on the number of buffers in the cache and the nature of the disk I/O activity. In general, each function which has a high rate of disk I/O, where the same AU is referenced several times, can be improved by the DDC. For example, the disk I/O for a SEARCH/MODIFY on a data batch with 20 records per AU could be reduced by a maximum factor of 20 but would probably be reduced by a factor of 10. If the Search/Modify accounted for 20 percent of all disk I/O, a corresponding reduction of 18 percent in total disk I/O could be realized. In this example, only one DDC buffer would be required. If too many DDC buffers are used, throughput may actually degrade. It is, therefore, advisable to start with few DDC buffers and work up to find the optimum number of buffers for the particular application. For indexed file activity, a good estimate is one buffer per file.

These improvements are obtained at the expense of system costs in the form of additional resident code, memory-resident data buffers, and increased CPU overhead on disk I/O.

The system functions improved by the DDC are:

VERIFY	INPUT
SEARCH/MODIFY	INDEXED FILE ACTIVITY
UPDATE	DISK SEQUENTIAL FILE ACTIVITY
NEW NAME	SPOOLING
PRINT	SORT
WRITE	DATA COM

1.3. FORMATTED READ FACILITY

A facility has been implemented for reading formats from disk through a 1344-byte AU buffer to allow format programs of any size to function properly under the SAVE, transmit in SAVE TAPE FORM, and POWEROFF commands.

1.4. NONCONTIGUOUS FIELD SORT

The Noncontiguous Field Sort (NC-SORT) feature enables SORT to assemble a SORT key from several fields. This allows the key to be composed of as many as 16 fields with a total length of up to 328 characters. A single key may be specified. NC-SORT cannot be used with indexed files; however, the addition of NC-SORT does not affect the standard sort capability.

A new format program, NC-SORT, has been added to the 11.R1.00 User's Library to allow the specification of field and key data.

1.5. ICL 1900 VARIABLE LENGTH RECORDS UNDER REPORT WRITER

ICL 1900 Variable Length Records (VLR) may now be generated using Report Writer.

ICL 1900 VLR capability has been previously available for the WRITE, PRINT, and INPUT commands, but was not implemented to allow Report Writer to use ICL 1900 VLR packing option. ICL 1900 VLR packing may now be specified when a Report Writer format program is called by a WRITE, or INPUT Supervisor command.

Refer to Appendix H for detailed information concerning the use of this.

1.6. NEW MEMORY TWAS

Previously foreground TWAs were eligible for Extended Memory residency. Several background TWAs are also eligible for Extended Memory, freeing Standard Memory space for other tasks.

The following foreground TWAs are eligible to reside in Extended Memory:

Stat TWAs

BK TWAs which include:

- DA (Disk Management)
- TWA (INDEX File)
- (SORT)
- VA (INDEX File)
- TWA (INDEX File)

1.7. CHARACTER EXPANDED

The character has been expanded from 10 characters to 40 characters, which reduces the frequency of "CHARACTER LOST--REKEY" messages during periods of rapid keying.

1.8. DISK FULL AT 98 PERCENT

The disk full condition has been adjusted to occur when there is only 2 percent of the disk space left. Previously, the disk was full with 10 percent left. This enhancement could make available as much as an extra 1.5 MB on a D3400 disk and 3MB on a large capacity disk. The associated system messages have been modified to reflect this percentage change.

1.9. DATA ENTRY PRIORITY SELECTION

This feature enables the use of the Supervisor SET...SHIFT command sequence to change the amount of processing time allocated for data entry functions. Data entry is assigned a low, medium, or high priority by keying 7, 8, or 9 (respectively) as the shift number in the Supervisor SET...SHIFT command sequence. The actual shift number then remains unchanged; only the priority is affected. Selecting a low priority (SHIFT=7) may impede data entry because a higher percentage of processor time is allocated to nondata entry functions such as compiles, data communications, tape I/O, etc. Selecting a high priority (SHIFT=9) may enhance data entry throughput by allocating more time to data entry functions and less time to nondata entry functions. Medium (SHIFT=8), which is the default priority, strikes a balance between low and high data entry priority. A COLD START or WARM START resets the priority to medium.

1.10. AUTO CHECKBOX FUNCTIONALITY CORRECTION

A correction has been made to the functionality of the AUTO checkbox to bring it into conformity with both the original design and the System Supervisor's Reference Manual. An s (skip) in the AUTO checkbox now causes the skipped field to be filled with the character designated in the FILL checkbox or, if the FILL checkbox is blank, with the appropriate default fill character (zeros for numeric fields; blanks for nonnumeric fields). Previously, auto-skipped numeric fields were erroneously blank-filled by default.

Format programs using auto-skipped numeric fields may require modification to accommodate this change.

1.11. IBM VARIABLE LENGTH RECORDS

IBM Variable Length Record (VLR) capability, previously implemented on the 1900/10, is available under the INPUT and WRITE commands and Report Writer in Version 11.R1.00 software.

Refer to Appendix J for detailed information concerning the use of this feature.

1.12. INTERCOM

This feature enables communication of one-line messages from any active keystation to one or all other active keystations. This is done by keying one of the following sequences:

```
CTRL/DISP  
MSG,ALL xxxxx,  
FLD REL
```

or

```
CTRL/DISP  
MSG,n xxxxx  
FLD REL
```

Where:

xxxxx
is the message.

n
is a valid keystation number.

The message will appear on the message line of the receiving keystation.

1.13. DEFAULT FORMAT=ANY FOR SV WRITE AND SV PRINT

The default format name displayed during WRITE and PRINT selection of format has been changed to ANY. Selection of the new default, by pressing the FIELD REL key or keying ANY, will result in each batch being output under control of the format program name in its own batch header.

Existing autoprompts may require modification in this regard.

1.14. NO TOP OF FORM ON PRINTER OPEN

The Sequential I/O OPEN for line printer and station printer output has been modified not to issue a top of form. This is true for both direct and station printer shared output. Previously, the OPEN OUTPUT statement for a sequential I/O file going to a line printer or station printer forced the page eject. If a page eject is desired on the Sequential I/O files, the user must now issue the top of form explicitly in the format program. This modification does not affect the form release function. Each form release still causes a page eject on shared output.

1.15. TEMP FILE DELETE IN STARTUP

Previously, SORT temporary files and INDIO temporary files remained on the disk if the system shut down before they were deleted. Since these files have special names, they could not be subsequently deleted. The disk recovery procedures for both system STARTUP and disk MOUNT have been modified to delete these files, zeroing them on the disk.

1.16. LOG DISK ERRORS

Hardware disk errors are now tracked by recording the disk number, segment number, sector address, function code, and status in low-core locations 2, 3, and 4 as follows:

Loc 2	Status
Loc 3	Disk number (first 6 bits), and segment number (last 10 bits)
Loc 4	Sector address

These locations can be examined using the maintenance panel. They contain only the information for the most recent error. In addition, if the system log option is on, the errors are logged with the following message:

DK ERROR: ADR=D,SS,AAAA FUNC=FF STAT=XXXX

Where:

D
represents the disk number.

SS
represents the segment number.

AAAA
represents the sector address.

FF
represents the function code (see Table 1-1).

XXXX
represents the status code (see Table 1-2).

Table 1-1. Disk Function Codes

Disk Type D3400 (17.6 MB)		Disk Type LCD (ISS)	
Code	Meaning	Code	Meaning
01	RESTORE	01	HEADER LOCK
02	SEEK	02	SENSE DEVICE
10	WRITE (DATA XFER)	03	RESERVE
11	READ (DATA XFER)	04	RELEASE DEVICE
12	WRITE ZEROES	10	FORMAT TRACK
13	READ (NO DATA XFER)	11	INIT TRACK
14	INIT WITHOUT XFER	13	READ IMMEDIATE
15	READ DIAG	14	WRITE (DATA XFER)
20	UNLOAD	15	READ (DATA XFER)
30	STATUS	20	VERIFY TRACK
FF	FAILURE DUE TO: CONTROLLER BUSY, DEVICE BUSY, BAD DISK ADR OR MEMORY ADR SPECIFIED IN REQUEST	30	READ TRACK
		40	READ HEADER
		50	REWRITE HEADER
		60	READ (SEEK)
		70	WRITE (SEEK)
		D1	END CHANNEL PROGRAM
		FF	FAILURE DUE TO CONTROLLER BUSY, DEVICE BUSY, BAD DISK ADR OR MEMORY ADR SPECIFIED IN REQUEST

TABLE 1-2. Disk Status Codes

Disk Type D3400 (17.6MB)		Disk Type LCD (ISS)
Bit	Meaning	Meaning
0	DK READY BIT	LCD READY BIT
1	DK READ ONLY BIT	LCD READ ONLY OR ADDR ERR BIT
2	DK SEEKING BIT	LCD DATA END BIT
3	DK SEEK COMPLETE BIT	LCD SEEK INCOMPLETE BIT
4	DK SEEK CHECK BIT	LCD CMND ERR BIT
5	DK 4 PLATTER BIT	LCD CP ERR BIT
6	DK 2 PLATTER BIT	LCD SEQUENCE ERR BIT
7	DK PHYIO NO ACCESS BIT	LCD PHYIO NO ACCESS BIT
8	DK NORMAL END BIT	LCD UNIT SELECT ERR BIT
9	DK HEADER, ERR BIT	LCD ERR TYPE BITS (3 BITS)
A	DK CRCC ERR BIT	LCD ERR TYPE BITS
B	DK DATA LATE BIT	LCD ERR TYPE BITS
C	DK SCTR OVERRUN BIT	LCD DEV OR LCDC ERR BIT
D	DK DRIVE MALFUNCTION BIT	LCD UNSAFE BIT
E	DK ADDRESS ERR BIT	LCD ABN END BIT
F	DK BUS ERR BIT	LCD BUS ERR BIT

1.17. DS COMPILER WITH RECORD NUMBERS

The COMPILER display has been modified to show the record number at the left of each line. The record number display field handles up to four digits and is followed by one blank.

1.18. PRINT COMPILER WITH PAGE EJECT

The COMPILER to a print device has been modified to print a maximum of N-6 lines per page, where N is the number of lines on a page. This prevents printing over perforations. N is normally set to 66 (decimal) and is kept in a low-core cell so that it can be easily modified by a CDM patch. In this release, the cell is in LOW CORE (LC) at location 2C5. This location should be verified with Marketing Support.

1.19. SET TIME AND DATE OPTION

The SET command has been modified to allow the reset of the DATE and TIME by combining the DATE and TIME options with the SHIFT option under a single SET command. The SHIFT option in SET has been changed to read "TIME/DAY". The prompts for setting DATE and TIME will display the current DATE or TIME and the default selection will be "NO CHANGE". If the Supervisor keys a response to the SET DATE or "SET TIME", the same checks will be performed as at the startup of the DATE and TIME specification. Only the system date cells in low-core will be modified on a DATE change, and the two-second clock in core will be modified by the TIME change. No attempt is made to adjust other structures which may have been previously set with the old DATE and TIME.

Autoprompts that change the SHIFT may need to be modified to accommodate this feature.

1.20. RUNTIME 6-CHARACTER TIME

The runtime TIME register obtained by the GET TIME verb has been modified to have six characters of significance instead of four. The six characters are hours, minutes, and seconds. All format programs which access the time register as a four-character quantity will still receive hours and minutes only.

1.21. BK LOG MESSAGE EXPANSION

By pressing the RESET key and then the CTRL/DISP keys, the operator may key onto the message input line (line 12 on the small screen or line 2 on the large screen) in order to invoke Command Manager or Datacom options, or to answer system messages of the form:

xxxxx KEY n,ANS **yyyyy**

Where:

xxxxx
is the message.

n
is the message number (1 through 9).

yyyyy
names the module responsible for the message.

If background message-logging is enabled, the log will include message line responses keyed by the operator (using the RESET, CTRL/DISP key sequence) in addition to background messages displayed in response to operator-keyed messages. This allows the log to show the operator responses to messages requiring an answer, and it indicates whether the operator response was accepted.

1.22. "ENTER MESSAGE" REMOVED

The "ENTER MESSAGE" prompt is no longer displayed after the operator keys RESET and then CTRL/DISP. This prevents the "ENTER MESSAGE" prompt from overwriting the line 3 message. The cursor still moves to the message input line to indicate that the operator should key a response.

1.23. REBROADCAST BACKGROUND MESSAGES

All unretired messages are held in a buffer until they are retired. The messages are modified slightly as they are placed in the buffer.

The KEY n,ANS is changed to:

MSG n,xx

Where

n
is the message number (1-9).

xx
is the decimal number of the keystation which must respond to retire the message.

At one minute update, all messages will be rebroadcast to the original terminal (provided the terminal is still active, the message has not been broadcast in the last minute, and the message line is not busy). Messages are displayed on the message line with a tone, but no reset is required.

1.24. BACKGROUND MESSAGE REPORT

The MONITOR command has been modified to allow the monitoring of terminals or unretired messages. If the unretired messages option is selected, all messages requiring response are logged to the supervisor screen. The message is logged in the modified form as in

the previous paragraph. "MSG n,xx" replaces "KEY n,ANS". If there are no unretired messages, the message "NO OUTSTANDING MESSAGES TO REPORT" is issued.

1.25. U100 DATACOM

The UNISCOPE 100 terminal protocol (U100 protocol) allows communication in a batch input and output mode with mainframe computers using that protocol. Although the peripherals normally controlled through UNISCOPE terminals may not be connected to the shared processor, all existing shared processor peripherals may be used under this U100 protocol emulation.

Operating procedures for using the U100 protocol are similar to those for data communications using other protocols. A keystation must be dedicated to controlling data communications when U100 protocol is selected.

The U100 protocol may be selected during SYSGEN or by using the SET command. A Remote ID (RID) identifying the communication routing and a Site ID (SID) identifying the workstation also are assigned.

Refer to Appendix J for further information concerning its use.

1.26. SETW DIAG MODIFICATIONS

The Software Diagnostics tape for support of Version 11.R1.00 has been enhanced with the following modifications:

1. Directories, as well as jobs and compiles, may be recovered during the recovery process.
2. The SAVE command has been modified to save all contents of a drive with multireel capability. (See D.5.24.)
3. The RESTORE command has been modified to restore all the contents of a drive with multireel capability. (See D.5.22.)
4. Defaults have been added for the following commands:

Command	Default Parameter
SAVE	OUTPUT= T1
RESTORE	INPUT = T1
RECOVER	OUTPUT= T1
COPY	INPUT = T1
	OUTPUT = T2

5. Dump output has been updated to match the current TWA configuration.
6. The HEADER command has been modified to prompt for a 30-character header rather than a 10-character header.
7. New facilities have been added to DUMPM to unwind the SCL and TENANT TABLE in addition to TWAs.
8. The ability to load a VFU for the printer has been added.
9. A new command, PLTH=, has been implemented to enable specification of the physical page length in support of the VFU load feature.

Procedures for using these new Software Diagnostic features are detailed in Appendix J.

1.27. ANY DATA KEY TO CONTINUE

This feature allows any data key to act like a REC FWD in response to a "REC FWD TO CONTINUE" message when in the Supervisor Mode. It also allows any data key to put up the Operating Mode prompt from the "DONE", "READY", or "BATCH STATISTICS" displays.

1.28. SEARCH BACKWARD BY NUMBER

In the Search/Modify mode, this feature allows the operator to search forward to a record number and also search backward to any record number. Searching backward may cause replay from the beginning of the batch to occur.

NOTE:

Zero is not a valid record number.

1.29. FREEFORM SEARCH

Additional flexibility has been provided in Version 11.R1.00 when searching (on content or on level and content), enabling the operator to search for a text string either in a specific position in a record or wherever it may be found.

The new Search mode is invoked by pressing the -SKIP key after the search text is entered. When searching by content or by level and content, the operator may change the mode back and forth between positionally-dependent and freeform searches any time a new search mask is entered by terminating the search text with a FLD REL or the -SKIP key respectively. For example, to search for the text, ABC, in the positionally dependent mode, the operator might key the mask as:

ABC ... (FLD REL)

This invokes a search for the first record containing the text, ABC, in columns 4, 5, and 6 of the record. To invoke a freeform search of the same text, the operator enters:

ABC (-SKIP)

This causes a search for the next record containing the contiguous text, ABC, anywhere within the record.

1.30. WRITE CONTINUE — TOTAL RECORD COUNT

Previously, the total record count displayed on the completion of a WRITE CONTINUE showed only the record count of the last WRITE. With this feature, the record count will be the total record count on the tape (old record count plus new record count).

NOTE:

The maximum record count is 65,536. If the total record count exceeds the maximum value, the counter will start over at zero. For example, if a batch containing 536 records is added to a tape with 65,000 records, the displayed record count will be 00000, if 537 records are added, the count will be 00001.

1.31. DISPLAY-ONLY RECORDS

This feature allows the compilation and execution of format programs with records greater than one AU in length. This is intended to make the job of filling the large 2000-character screen easier. The maximum record size has been increased to 2540 characters.

In order to use the new feature, the programmer must place a NO-DISK statement in each level of the format program containing the large records. If the NO-DISK statement is omitted, the system detects the attempt to write the larger record, displays the message

"LOGICAL RECORD SIZE EXCEEDED", and closes the batch. The error is detected at run time. No error displayed when the program is compiled. When the "LOGICAL RECORD BUFFER SIZE EXCEEDED" message is displayed, the programmer should check the program for a "NO-DISK" statement and verify that the record size does not exceed 1334. The change should be made and the program recompiled.

NOTE:

For a definition of Logical Record Size, see the 1900 Systems Analyst's Reference, UP-8693.

1.32. "FORMAT NOT FOUND" MESSAGE

The "FORMAT NOT FOUND" message has been modified to include the type of format. The messages are as follows:

FORMAT NOT FOUND	:xxxxxxxx	message for format programs
FORMAT NOT FOUND	L:xxxxxxxx	message for label programs
FORMAT NOT FOUND	S:xxxxxxxx	message for subprograms
FORMAT NOT FOUND	T:xxxxxxxx	message for tables

Where:

xxxxxxxx
is the missing program name.

1.33. SINGLE RECORD REPLAY

Previously, when an operator backed up beyond the last checkpoint and made a change, the system would reposition back to the first record of the batch and replay all the records and procedures down to the modified field.

This feature provides the option to selectively replay records (by format program) from only the start of the current record.

The feature is invoked by placing a numeric 1 in the REPLAY FLAG field (formerly the AVERAGE KEYSTROKE RATE field) of the format program header. Refer to the 1900/10 System Supervisor's Reference, UP-9103.

The programs FORMAT and FORMATI, in the User Library, are examples of programs that can use the Single Record Replay feature.

NOTE:

The Single Record Replay feature must not be used in programs that use registers, DUP fields, or working storage to carry information from level to level or record to record. Only programs which use registers, DUP fields, or working storage to store information within each level may use this feature.

1.34. DOUBLE CHECKPOINT

Without this feature, the system maintains only one checkpoint in the batch header. Normally this checkpoint is taken on every 10th record, unless the checkpoint verb is used.

This feature increases the speed of checkpointing by providing two checkpoints, the regular one in the batch header on the disk and a new one maintained in the TWA. This means that the operator should always have a checkpoint within the last 10 records, eliminating the need for long replay. This feature is active only when the FAST REPLAY option is selected during SYSGEN, and does not require programming changes.

1.35. WARM START ENHANCEMENTS

1.35.1. WARM START Speedup

This feature speeds up the time required to do a WARM START by not initializing AUs that have never been used.

The amount of time saved depends on the amount of data on the disk. If the disk contains more data, less WARM START time is saved.

Neither data nor free disk space is lost.

1.35.2. WARM START Fast Recovery

This feature speeds up WARM START significantly by not recovering free disk AUs. When a WARM START is requested, a new prompt is displayed:

FAST RECOVERY? Y=YES, REL=NO

If REL is selected, normal WARM START is performed. If the response is Y (YES), unused AUs on the platter that were in use before the

restart are not recovered. Any subsequent WARM START without Fast Recovery will recover all unused AUs.

1.36. LINK BACK TO CHECKPOINT

This feature was implemented to increase system throughput. It does not affect system operational procedures.

The link back to checkpoint feature causes replay to take the shortest route to the checkpoint record instead of always starting from the beginning of the file.

1.37. BREAK UP LONG COMMUNICATION RECORDS

This feature makes it possible to break records which have record lengths of more than 80 characters into multiple 80-character records for BSC protocols.

New prompts have been added to the SYSGEN procedure and the Supervisor SET=DATA COM options to facilitate selection of this feature.

1.38. DELETE COMMAND DEFAULT CHANGE

In order to reduce the risk of accidentally deleting data from the disk, the JOB NAME default for the DELETE command has been changed from ALL to SELECT, and the COMPILE NAME and DIRECTORY NAME defaults have been changed from ALL to NONE.

Because of this change in the prompt sequence, Autoprompts using the DELETE command may require modification.

1.39. THREE-CYCLE VERIFY OPTION

A SYSGEN option has been added to allow selection of either a 2-key or a 3-key correction for VERIFY. Immediately following the prompt for "CORRECTION COUNT BY RECORD (Y,N)" in the SYSGEN dialogs, a new prompt appears as follows:

THREE CYCLE VERIFY (Y,N) = N

If the default of NO is selected, the system will use two-key VERIFY correction as on previous versions.

If YES is selected, the VERIFY correction process will function as follows:

Current Field Value	AAA
Operator Keys	B
System Responds	VERIFY MISCOMPARE
Operator Keys	B
System Responds	VERIFY MISCOMPARE
Operator Keys	B
Field Contents	BAA

1.40. MESSAGE QUEUING

A 15-second timer has been established for background message responses to prevent a system lockup when a response is not fully entered. The timer is initiated after RESET, CTRL/DISP is keyed and the cursor moves to the message input line. After 15 seconds has elapsed, 'MESSAGE CANCELLED' is displayed and the response operation is cancelled as if the BYPASS key had been pressed. The next terminal in the queue then displays the cursor on the message input line to indicate that the system is ready to receive the next message response. This prevents any incomplete message response from halting the next terminal needing to respond to a message.

All message response procedures are unchanged except that if an operator initiates a message response (keys CTRL/DISP), and does not complete the response within 15 seconds, the message is cancelled. This allows any other terminal in the message queue to enter a message. The previously-cancelled message can be completed by again keying CTRL/DISP and entering the response within 15 seconds.

An example of message queuing is as follows:

Action	Response
1. Key CTRL/DISP.	
2. Wait 15 seconds.	MESSAGE CANCELLED
3. Key CTRL/DISP.	
4. Key CTRL/DISP on another terminal.	Terminal will hang.
5. Wait fifteen seconds.	MESSAGE CANCELLED on first terminal; second terminal unlocks.

6. Wait 15 seconds. **MESSAGE CANCELLED** on second terminal.
7. Execute an SV RESTORE with tape NOT ready. **TAPE NOT ON-LINE KEY 1,ANS**
8. Key RESET, CTRL/DISP.
9. Wait 15 seconds. **MESSAGE CANCELLED**
10. Key CTRL/DISP.
11. Put tape on-line. No response.
12. Key 1,R (within 15 seconds of CTRL/DISP). Drive will read tape.

1.41. MAPPED TWA SPACE

The Mapped TWA Space (MTS) feature utilizes Extended Memory (EM) for certain terminal work areas, relieving contention for Standard Memory (SM) and increasing the total amount of memory which may be used for TWA storage.

MTS allows normal Supervisor and Operator TWAs to reside in EM. The general memory management functions were expanded so that TWAs may be allocated, maintained, accessed and released in EM as well as in SM. EM-resident TWAs are supported by revised TWA management functions and structures. MTS accommodates concurrent EM residency of a foreground TWA, a format program and a system overlay. All I/O for EM resident TWAs will be routed indirectly through SM buffers.

1.42. EXTENDED MEMORY OPTION REMOVED

The prompt "EXTENDED MEMORY = YES OR NO" has been removed from the GDS and GTS dialogs of SYSGEN. The system now checks all available memory and configures the system accordingly. For this reason, "EXTENDED MEMORY NOT AVAILABLE" is no longer displayed at STARTUP on 1900/10s having only 128K bytes of memory.

1.43. WRITE CONTINUE FOR H6000

This feature adds the CONTINUE option to the Supervisor WRITE command for H6000 SSF format tapes. All current rules for continuing a tape apply to this feature. The CONTINUE operation

must use the same record size, block factor, packing, and label as the original WRITE operation.

1.44. MONITOR ENHANCEMENT

The message, "INVALID TERMINAL NUMBER", which occurred when using the MONITOR command, has been replaced with five self-explanatory messages to clarify the nature of the error:

MONITORING TERM 0 OR 1 INVALID
MONITORING TERM OUTSIDE RANGE INVALID
MONITORING YOURSELF INVALID
MONITORING LARGE WITH SMALL INVALID
MONITORING SMALL WITH LARGE INVALID

1.45. CORRECTION COUNT OPTION

The correction count has been changed from a SET command to a SYSGEN option. Errors during data entry functions may be accumulated on a field or record basis. The prompt, "CORRECTION COUNT BY RECORD (Y,N)=Y", now appears after the "LOAD FILE MGMT AT STARTUP (Y,N)=N" prompt in the GDS sequence. The SET command for "CORR CNT" has been deleted.

1.46. DISK RECORD SIZE IN DETAILED STATUS

This feature displays the Disk Record Size (DRS) in the statistics of the detailed STATUS report. The DRS is defined in bytes and is calculated as the data record length (rounded up to an even number of bytes). This number is displayed on line two of the detailed STATUS as "DRS" with the number following.

1.47. SEQUENTIAL I/O ACCESS TO SPOOL FILES

It is now possible to process spool files as if they were sequential files. The only condition that must be met to enable this capability is that the record size in the Data Definition Level of the sequential program must match the Disk Record Size (DRS) given in the detailed status of the spool file (1.4.6).

1.48. LOG FIRST RECORD OF DATACOM FILES

The first 40 characters of the first record of each file received from the communications line will be logged to the LP if logging

was selected . Previously, this occurred only when the Multiple Auto Answer option was selected. This feature is implemented for 2770, 2780, 3780, HASP and U100 protocols.

1.49. INDIO MODULES UNLOADED AT POWEROFF

Previously, if the supervisor performed POWEROFF with the file management (INDIO) modules loaded into memory, it was necessary to perform BYPASS, UNLOAD, FILE MGMT operations, then a POWEROFF to complete the system shutdown. The POWEROFF command now performs an INDIO unload and proceeds with shutdown. A message, "INDIO UNLOAD IN PROGRESS", displays on the supervisor terminal during this function.

1.50. DATA COMMUNICATIONS ENHANCEMENTS

Enhancements have been added to the front-end processing of data communication transmission. These enhancements enable implementation of the two new features which are described in the following paragraphs. Both are implemented through the use of prompts added to the BUILD DIR command sequence.

1.50.1. Directory Construction at Transmission Time

The supervisor may, at the time of directory creation, specify data batches to be included in the communication directory whether or not the batches are currently on disk. If this feature is selected, all batches on disk at the time of the transmission and that fit the directory specifications will be transmitted. This allows the user to predefine the range of batch names to be transmitted where batches have not yet been created.

1.50.2. Reformat of Data at Transmission Time

This function enables the user to transmit reformatted data directly from the disk, alleviating the need to first reformat the data by writing it to tape under a specified format program. Now the user may specify reformatting from disk and provide a format program name when creating the directory. When the user requests transmission of this directory, the system first reformats the data according to the output specifications of the format program, and then transmits the reformatted data.

1.51. LARGE CAPACITY DISK SUPPORT

Version 11.R1.00 software is capable of supporting large capacity Winchester technology disks. This enhancement is provided through the expansion of disk addressing capacity to 100 MBytes and the addition of a new disk driver module called LCD DRVR.

These modifications enable Version 11.R1.00 software to support a PCC D3400 disk with one to four drives consisting of either two or four platters. The system also supports configurations in quarter-disk increments of up to two ISS disk drives.

The operating system must be created to run on a specific type of disk. The disk type is specified in response to a new prompt added to the GDS procedure. This prompt appears as:

DISK TYPE? REL OR D=D34 I=ISS

If a large capacity disk is used, the LCD DRVR must be loaded into memory. This is done by setting LCD DRVR to LOAD status and D34 DRVR to SKIP status on the SMB. (If both are at LOAD status, the system will default to D34 DRVR.)

1.52. IMPROVED DISK UTILIZATION PROCEDURES

The following major improvements have been added to the various disk utilization procedures:

1. Platter-spanning for data batches is made possible by expansion of the disk address to 24 bits. This enables a batch to reside over the range of a disk drive, instead of being restricted to a 4.4 MByte platter.
2. The disk maps are partitioned so that only a part of each map is resident in memory.
3. Disk space now is normally allocated in blocks of contiguous AUs and disk recovery procedures are performed by multiple AUs. This speeds up the recovery process.
4. Disk drives are now treated as single units. Disk initialization, recovery, mount and dismount functions are modified to work on a disk drive basis, instead of a platter basis. Disk Mount is a background task. There may be system degradation during a mount.

5. The STARTUP procedures have been changed to include a WARM AUTOMOUNT function which allows each disk drive in the hardware READY condition to be mounted as an OLD DISK.
6. A WARM SYSLOAD function is added to allow an SYS TAPE load to be performed while preserving data batches, indexed files, sequential files, and directories. Compiles are not preserved. A RESTORE of FORMAT, SUBPROGS, TABLES, and LABELS will be required.

NOTE:

A WARM SYSLOAD must be preceded by a POWEROFF.

7. A new DISK STATUS function is added which gives detailed status information about disk drive utilization on a drive basis.
8. The DISK RESERVE function is expanded and made available as both part of the MOUNT function and independent of the MOUNT. The expansion gives the supervisor control over disk drive selection at three different levels:

A disk drive may be accessed normally.
Restrictions = NONE.

A disk drive may be reserved to prevent creation of new batches.
Restrictions = BATCH CREATE.

A disk drive may be reserved to prevent creation of new batches or extension of existing batches.
Restrictions = ALLOCATION.

9. System utilities such as IOCTL5, PERDEBUG, and SFTW DIAG are modified to work with the D3400 and ISS disk drives.

1.53. UNIQUE USER'S PROGRAM LIBRARY BATCH NUMBERS

The batch numbers for each job in the User's Program Library have been changed to 50000. This has been done so that library programs may be more easily identified in a Supervisor status display. This feature does not preclude the use of batch number 50000 for other purposes.

2. Compatibility

2.1. COMPATIBILITY WITH PREVIOUS LEVELS

2.1.1. Batch Compatibility

Regular data batches and directories may be SAVE'd on versions 1.6D, 10.R3, or 10.R4 and RESTORE'd on Version 11.R1.00. Regular data batches SAVE'd on Version 11.R1.00 may be SAVE'd only on Version 10.R4 or 11.R1.00.

Indexed files may be SAVE'd on versions 1.6D, 10.R3, or 10.R4 and IF RESTORE'd only on Version 11.R1.00 or 10.R4. An improperly structured file is created if an IF RESTORE of a Version 11.R1.00 indexed file is attempted on a Version 10.R3 or 1.6D system.

2.1.2. Format Program Compatibility

It is mandatory that all old format programs from previous software levels be recompiled on this release prior to their use.

Due to the addition of new reserved words, 11.R1.00 programs are not backward compatible. Any programs from previous releases will have compile errors if the new reserved words were used as labels or field names in those programs.

Format programs using auto-skipped numeric fields may require modification to accommodate the change in AUTO Checkbox Functionality Correction. (See 1.10.)

Format programs from old User's Program Libraries are compatible with this version with the following exception: If the new Single Record Replay feature described in 1.34 is to be used, only the 11.R1.00 or 10.R4.30 versions of FORMATI or FORMATIS are compatible. Also, several new format and label programs have been added to the User's Program Library. These programs are discussed in Appendix C.

2.1.3. AUTOPROMPT Batch Compatibility

AUTOPROMPT programs involving the Supervisor commands BUILD DIR, DELETE, DISK MGMT, PRINT, SET, and WRITE may require modification before they will run properly on Version 11.R1.00. Refer to Appendix A, Operational Considerations, for a description of new or altered prompt sequences.

2.1.4. Patch Compatibility

Old patches are not compatible with Version 11.R1.00. Version 11.R1.00 patches are not compatible with previous software versions. In addition to numerous changes in code and addresses throughout the software, there are two significant changes to the patch format for Version 11.R1.00 to safeguard patch integrity and minimize procedural errors in the application of patches.

1. A patch number of up to 4 digits, plus a check-digit used to set bits in a patch mask, now follows the version number in every patch.
2. Each L or P command in the patch must be followed by the contents (in hexadecimal) of the address specified in that command. The patch will not be applied if those contents are not actually found at that address. This prevents accidental reapplication of patches already applied, and ensures that patches are applied only at the intended address in the code.

Example:

MODULE NAME:	EXECUTIVE
VERSION NUMBER:	11.61
PATCH NUMBER:	4/8
COMMAND:	L6BR/6
OLD CONTENTS:	1547/4
PATCH:	LDA 57/1 LSR 9./6 G

2.2. RELATED PRODUCTS

2.2.1. Products Executed to Construct this Product

Not applicable.

2.2.2. Products Executed With This Product

Disk Compatibility

An attempt to perform a MOUNT OLD DISK of version 10.R3 or 1.6D disks on version 11.R1.00 will result in an error. These disks must be mounted NEW at conversion time.

SAVE TAPE FORM Compatibility

Data transmission using SAVE TAPE FORM may be performed between any combination of version 10.R4 and 11.R1.00 systems only.

2.2.3. Products Executed with a Program Produced by this Product

Not applicable.

2.3. ~~LONG-TERM COMPATIBILITY CONSIDERATIONS~~

Not applicable.

3. Restrictions

The minimum memory size is 128K bytes. The maximum memory size is 512K bytes.

Up to 62 keystations may be configured in a version 11.R1.00 system.

There is no restriction in version-to-version peripheral interchangeability.

4. Release Media Description

The Version 11.R1.00 software release is contained on 3 600-foot tapes as follows:

1. Field Release Master Tape (FRM) Number 11.R1.00 consisting of the System Master Binary (SMB) and Software Diagnostics tapes copied one after the other onto this tape for ease of back-up and shipping.
2. User's Program Library Tape.
3. Patch Tape Number 11100002.

As it will be necessary to recopy the SMB file from the FRM tape to another tape for use, refer to Appendix K for a description of how this is to be done.

5. Corrections

The following is a list of new patches and the problems they correct:

- BRL-0001** Speeds up the tape write for programs containing prompts.

- BRL-0006** Disables the read after write feature.

- BRL-0008** Causes ZERO PARTIAL FIRST AU to zero only a portion of a sector in the disk buffer, and eliminates map switching on the first disk interrupt, for which no processing is performed.

- BRL-0009** Prevents batches from being written to the restricted area when there is still room available in other segments of the disk.

- BRL-0010** Corrects erroneous code in LCD DRVR.

- BRL-0011** Corrects KSTRKS/HOUR information in detailed STATUS report under VERIFY STATS.

- BRL-0012** Corrects a bad transaction which prevented tenants from being released.

- BRL-0013** Increases screen refresh frequency by a factor of ten to enhance screen response during fast keying.

- BRL-0015** Removes obsolete log device prompt and KS prompt from the SYSGEN procedure.

- BRL-0016** Flushes auxiliary keystroke buffer so buffer will not be shifted after collapse of TWA causing memory to be scrambled.

- BRL-0017** Corrects invalid disk percentage when performing a fast start.

- BRL-0018** Corrects a System Error 24 when selecting "FAST RECOVERY=YES" followed by "FAST RECOVERY=NO".

- BRL-0019** Corrects System Error 40 caused by CALLs which should have been CFOs.

- BRL-0020** Prevents System Errors 18 and 19 when there are bad tracks or data on the current MCB.

- BRI-0021** Removes the "UNIVAC KEYSTATIONS" prompt from the SYSGEN prompt sequence and forces would-be response to YES.
- BRI-0023** Prevents SYSGEN from allowing second entry of LP, since only one LP device is supported.
- BRI-0024** Removes references to Kennedy disk type in the SYSGEN prompts, since Kennedy disks are not supported.
- BRI-0025** Corrects erroneous code in STARTUP2.
- BRI-0026** Corrects the ordering of messages on compiled file conflicts.
- BRI-0027** Speeds up REC<-- on large files.
- BRI-0028** Prevents logical disk errors of H400 from being logged to the log device. For LCD disk configurations.
- BRI-0029** Prevents logical disk errors of H400 from being logged to the log device. For D3400 disk configurations.
- BRI-0030** Corrects "BATCH NUMBER RANGE ERROR" on batch numbers greater than nine when building a DCOM directory.
- BRI-0031** Prevents System Error 13 during POWEROFF.
- BRI-0034** Prevents inadvertant shift locks.
- BRI-0035** Corrects a problem where a system configured for flexible diskettes without data communications erroneously acts as if data communication had also been configured.
- BRI-0036** Prevents System Error 98 and other errors caused by CALLs which should have been CFOs.
- BRI-0037** Ensures that an EOT is sent when a data communications transmission is aborted before the "XMT DONE" message has been received.
- BRI-0038** Implements the Data Entry Priority Selection feature for adjusting the runtime time slice using the SET...SHIFT command sequence.
- BRI-0039** Corrects a problem with the new feature to break up long BSC communications records.

- BRL-0041** Prevents occurrence of negative record count values during file management.
- BRL-0042** Assures proper handshaking with the transmitting station when printer errors interrupt BSC communications RECEIVE operations.
- BRL-0043** Assures proper handshaking with the transmitting station when printer errors interrupt BSC communications RECEIVE operations.
- BRL-0044** Corrects logic errors in RESTORE.
- BRL-0051** Corrects erroneous code in PARM BUF1.
- BRL-0052** Assures proper deallocation of INDIO buffers.
- BRL-0060** Prevents System Error 98 and other errors caused by CALLs which should have been CFOs.
- BRL-0062** Prevents System Error 96 and other errors caused by moving system messages to the wrong location.
- BRL-0064** Corrects logic errors in SUBBATCH.
- BRL-0065** Prevents garbage in AUTO-DUP fields when transmitting data reformatted from disk.
- BRL-0067** Removes redundant code responsible for various system errors.
- BRL-0070** Corrects a bad transaction which was causing UNRECOVERABLE DISK ERRORS.

6. Support

The support level of Version 11.R1.00 software is Category 1. It will receive periodic updates, revisions, enhancements and software correction service for significant errors in SPERRY Computer System's then current version of the software.

6.1. RELEASE HISTORY

Release	Machine	Release Date	Discontinuance Date
11.R1.00	both	Sept 1983	Supported
10.R4.30	1900/10	Jan 1983	Supported
10.R4.19	1900/10	Jun 1982	July 1, 1983
1.6D.91	XL/40	Jan 1982	Supported

6.2. RESOURCE UTILIZATION

6.2.1. Memory

This operating system consists of two parts: Fixed and variable.

Fixed = 78,000 bytes \pm 10 percent, depending on configuration.

Variable = completely dependent upon the number of keystations and functions in use.

To map memory usage on an installed system, see C.4.10.

For preinstallation planning, see Table 6-1 and Figure 6-1. The total memory needed is the sum of memory used by each function, plus the memory for each keystation CTWA, and the compile that will be used at any given time.

Values in Table 6-1 were obtained with typical programs and configurations, but will vary according to the parameters of each site.

Table 6-1. Supervisor Command Memory Usage

Function	SM*	EM**
System	74,500	3,500
Op-stats (collection)	580	3,262
Op-stats (report)	704	8,752
Optional features	1,066	2,208
Communications	5,224	13,994
Write (tape)	2,128	8,156
Print	2,256	6,830
Compile (LP)	3,100	17,198
File Management	5,756	11,556
Disk Data Cache	3,404***	2,404
Restore	3,182	8,542
Save	3,216	6,388
Log (SP)	5,630	4,840
Auto Save	3,272	6,386

*SM is an abbreviation for Standard Memory, limited to 114,688 bytes and reserved for buffers and parts of the operating system.

**EM is an abbreviation for Extended Memory, which is 16,384 to 409,600 bytes of memory. EM is selectable in size to suit the needs of each site (depending upon the number of keystations, format programs, and supervisor functions performed).

***plus 1,510 per buffer over 1.

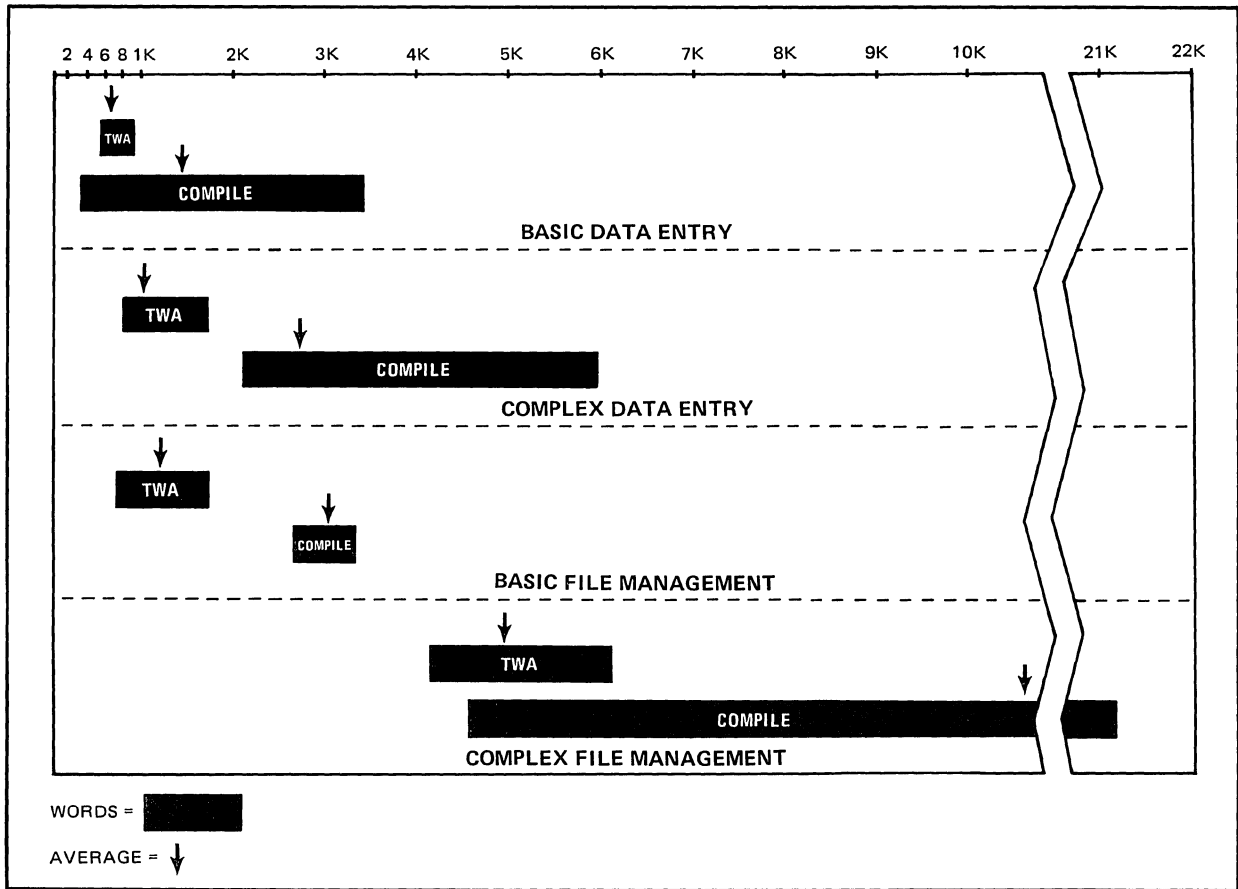


Figure 6-1.
Average TWA/Compile Ranges

The exact amount of memory needed for any given site depends upon variables such as number and size of fields, mode of each keystation, and program register usage. Often, these variables are unknown when the question is first asked. The following general terms are provided as guidelines, but no guarantees are implied:

Program Complexity Levels

1. Typically one line of COBOL per field.
2. Edits plus Indexed File Validation.
3. Complex COBOL.
4. Complex COBOL with multiple Indexed Files.

Supervisor Usage Levels

1. Intermittently used.
2. Some Supervisor function is active most of the time.
3. Frequently, background task is active along with foreground supervisor.

Table 6-2. Preinstallation Planning

Number of Keystations	Program Complexity	Supervisor Usage	Memory
3	1	1	128K
3-6	1	1	160K
3	2	1	160K
3	1	2	160K
3	4	3	192K
3-6	3	1	192K
6-12	2	1	192K
12-24	1	1	192K
3-6	4	3	224K
6-12	4	2	224K
12-24	2	2	224K
24-32	1	1	224K
6-12	4	3	256K

NOTE:

For any other configuration, a more complete study is recommended.

6.2.2. Disk

Disk space required by the operating system is 600,000 bytes. Changes in system configuration will not change this by any appreciable amount (even 1 percent of disk space).

Disk loading with Version 11.R1.00 can be reduced by using a feature called **Disk Data Cache**. Since multiple records may be placed in one disk Allocation Unit (AU), keeping AUs in memory reduces disk activity. Each site must test how many Cache Buffers are needed. Too few buffers increases disk load, and may cause delays. Too many buffers causes increased processor activity to find out which task should get a buffer and which buffer to use.

6.2.3. CPU

Processor loading has been reduced in Version 11.R1.00 by changing how some functions work (Search Mode and Prompting, for example). The 1900/10-XL/40 computers have limited use. Under programs with extensive procedures, it may only be possible to sustain key entry rates at 20 keystations, while other sites (with less COBOL) key at an average 18,000 keystrokes per hour on 32 keystations.

Exact predictions of processor loading are impossible to give. The best predictions are based on experience.

Some key factors to note when comparing the throughput between different sites are:

- Number of keystations
- Average keystroke rate of site
- Type of source documents
- Program complexity
- Operator modes in use
- Average batch size
- Disk percentage during keying
- Frequency of poweroff cleanups

A common misconception is that more memory gives more throughput. More memory actually allows more concurrent activity; it has no effect on throughput.

6.3. PROBLEM REPORTING PROCEDURES

The following paragraphs describe the software error reporting procedures for submitting errors on data entry software supplied and supported by SPERRY Computer Systems and the method with which these errors will be prioritized, distributed, solved, and closed.

6.3.1. Reporting the Error

All software errors are to be submitted to the following address:

SPERRY
SUR Coordinator
322 North 2200 West
M.S. B5B1
Salt Lake City, Utah 84116

Errors are to be submitted on the Software User Report (SUR) form UD1-745 Rev. 10-81, as outlined in SPERRY General Programming Information Exchange Bulletin, UP-8012. Urgent software problems may be called in to the field support group within the data entry organization; however, it is usually necessary to have dump tapes of the problem to perform proper analysis.

6.3.2. Priorities

The reverse side of the SUR form gives instructions for completing and submitting the SUR. Many SURs are not completed according to these instructions, which causes unnecessary delays. The System Analyst designates a priority on the SUR based on the criteria which follows. If this is not done by the originator, the Support Group will designate the priority according to their understanding of the problem. When the Support Group disagrees with the originator-designated priority, they will follow up with the originator and come to a mutual agreement.

- 4 (Critical) Problem stops production. There is no alternative method of bypassing the problem. This is the highest priority. Customer is losing data or the error is causing equipment to be down excessively. Such errors will be treated as urgent.
- 3 (Serious) Problem deters customer from meeting production schedules. This type of error occurs regularly, causing customer to restart operations, it is possible to work around inconvenience. Such errors are prioritized on a first in basis with available resources and consideration given to outstanding critical (4) problems.
- 2 (Minor) Problem causes inconvenience for customer. A nuisance which customer would like corrected eventually. Such errors are worked on when resources become available with consideration given to outstanding critical (4) and serious (3) problems.

Violations of given criteria are made when a minor (2) or serious (3) problem is designated by the SUR originator as critical (4) because of the political ramifications of the overall customer situation. If this situation does occur, the SUR submission should contain the information (either stated on the SUR form or in an attached memo) that this particular problem is "customer critical". Software Support will then handle the resolution of the problem ahead of all other priority 2 and 3 problems and, if necessary, ahead of some priority 4 problems. Upgrading the priority of a problem solely for political reasons defeats the purpose of the priority system.

6.3.3. Reviewing the Error

The following information must be provided on the SUR form before submission. This information is reviewed upon receipt by Software Support. Omitted items can be cause for SUR rejection or SUR closure due to insufficient information.

1. Computer, operating system/level, date prepared, customer name, city, state, priority code, initiator's name, SPERRY system analyst, phone number, problem, materials attached.
2. Frequency of occurrence, size of installation, whether error was recoverable and whether data was lost.

3. A detailed explanation of the error with description of what was happening prior to error (system activity) and any other available information.

Upon review by Software Support, if the error has been solved (or can be solved without further investigation) it is logged and closed, and the originator is informed of the solution by return of the answered SUR.

Software Support will also, when advisable, load the applicable software version and attempt to recreate the problem. Factors affecting the amount of time it may take to solve the problem are priority, difficulty, available resources (with expertise required for solution), and impact to the system.

A telephone or TWX is used to directly transmit the solution (such as a patch) to the System Analyst (originator) in emergency situations. However, the solution is still communicated to the originator formally.

A. Operational Considerations

A.1. 1900/10 USERS

The following system prompts have been added or modified for Version 11.R1.00:

A.1.1. DELETE

CMND=DELETE	JOB OPTIONS
NAME TYPE=JOB	DEFAULT=SELECT
OPTIONS=NONE	
OPTIONS=	ALL
	SELECT
	NONE

A.1.2. DISK MGMT

CMND=PAGE TWO	COMMAND
CMND=DISK MGMT	DEFAULT=NO MORE
FUNCTION=	
	MOUNT
	DISMOUNT
	RESERVE
	STATUS
	DATA CACHE
	NO MORE

CMND=PAGE TWO	COMMAND
CMND=DISK MGMT	DEFAULT=NO MORE
FUNCTION=DATA CHACHE	
FUNCTION=	LOAD
	UNLOAD
	ENALE WRITE
	DISABLE WRITE
	BUFFER CHANGE
	NO MORE

CMND=PAGE TWO	COMMAND
CMND=DISK MGMT	DEFAULT=NO MORE
FUNCTION=DATA CACHE	
FUNCTION=BUFFER CHANGE	DEFAULT=(current number)
NUMBER=	

A.1.3. PAGE TWO

CMND=PAGE TWO	SV CMND PAGE TWO	
CMND=	DEFAULT=PAGE ONE	
	TEST	CHKPT CNTL
	DISK MGMT	BUILD DIR
	SM TAPE	DCOM CNTL
	OP-STAT	SPOOL CNTL
	PRINT	FILE MGMT
	SORT	AUTOPROMPT
	FD UTILITY	

A.1.4. SET

CMND=SET	INPUT
SET=INPUT	DEFAULT=(current input dev)
INPUT=	
	T1,T2,T3,T4,CR
	F1,F2

CMND=SET	01 OUTPUT
SET=01 OUTPUT	DEFAULT=(current output dev)
01 OUTPUT=	
	T1,T2,T3,T4
	F1,F2
	DS,LP,SP

A.1.5. WRITE and PRINT

CMND=WRITE (or PRINT)	PACKING TYPE	
OPTIONS=NONE	DEFAULT=H2000	
JOB NAME=ALL JOBS		
FORMAT=ANY	NONE	
SKIP=NO	H2000	VLR-VLB
OPTIONS=NO MORE	H6000	
(PRINT only)		
OPTIONS=PK-PACKING	4X3 ICL	
PACKING	2970 ICL	
	1900 ICL	

CMND=WRITE (or PRINT)	DATA BLOCK SIZE
OPTIONS=NONE	DEFAULT=1024
JOB NAME=ALL JOBS	
FORMAT=ANY	
SKIP=NO	
OPTIONS=NO MORE	
(PRINT only)	
OPTIONS=PK-PACKING	
PACKING=VLR-VLB	
DATA BLK SIZE=	

A.2. XL/40 USERS

The following system prompts have been added or modified for Version 11.R1.00.

A.2.1. BUILD DIR

CMND=PAGE TWO	REFORMAT OUTPUT
CMND=BUILD DIR	DEFAULT=NO
TYPE=DCOM TRANSMIT	
SAVE TAPE FORM=NO	NO
BATCH NUM=1	YES
REFORMAT=	

CMND=PAGE TWO	FORMAT PROGRAM
CMND=BUILD DIR	DEFAULT= (Same
TYPE=DCOM TRANSMIT	format as batch
SAVE TAPE FORM=NO	was keyed under)
BATCH NUM=1	
REFORMAT=YES	
FORMAT NAME=	

CMND=PAGE TWO	WHEN TO BUILD
CMND=BUILD DIR	BATCH LIST
TYPE=DCOM TRANSMIT	DEFAULT=IMMEDIATE
SAVE TAPE FORM=NO	
BATCH NUM=1	IMMEDIATE
REFORMAT=YES	RUNTIME
FORMAT NAME=ANY	
DIR TIME=	

A.2.2. DELETE

CMND=DELETE	COMPILE OPTIONS
NAME TYPE=COMPILE	DEFAULT=NONE
OPTIONS=	
	ALL
	SELECT
	NONE

CMND=DELETE	OPTIONS
NAME TYPE=DIRECTORY	DEFAULT=NONE
OPTIONS=	
	ALL
	SELECT
	NONE

CMND=DELETE	JOB OPTIONS
NAME TYPE=JOB	DEFAULT=SELECT
OPTIONS=NONE	
OPTIONS=	
	ALL
	SELECT
	NONE

A.2.3. DISK MGMT

CMND=PAGE TWO	COMMAND
CMND=DISK MGMT	DEFAULT=NO MORE
FUNCTION=	
	MOUNT
	DISMOUNT
	RESERVE
	STATUS
	DATA CACHE
	NO MORE

CMND=PAGE TWO	COMMAND
CMND=DISK MGMT	DEFAULT=NO MORE
FUNCTION=DATA CACHE	
FUNCTION=	
	LOAD
	UNLOAD
	ENABLE WRITE
	DISABLE WRITE
	BUFFER CHANGE
	NO MORE

CMND=PAGE TWO	NUMBER OF BUFFERS
CMND=DISK MGMT	CURRENT IS DEFAULT
FUNCTION=DATA CACHE	
FUNCTION=BUFFER CHANGE	DEFAULT=(current number)
NUMBER=	

A.2.4. SET

CMND=SET	SET VARIABLES
SET=	DEFAULT=NO MORE
	O1 OUTPUT PASSWORD
	O2 OUTPUT TIME/DAY
	DC OUTPUT TAPE
	LOG DEVICE INPUT
	AUTO SAVE DATA COM
	MSG TRMNL LENGTH

CMND=SET	SELECT OPTION
SET=TIME/DAY	DEFAULT=NO MORE
OPTION=	
	DATE
	TIME
	SHIFT
	NO MORE

CMND=SET	JULIAN DATE
SET=TIME/DAY	DEFAULT=(current date)
OPTION=DATE	
DATE=	NO CHANGE
	ON DEFAULT

CMND=SET	24 HOUR TIME
SET=TIME/DAY	DEFAULT=(current time)
OPTION=TIME	
TIME=	NO CHANGE ON DEFAULT

CMND=SET	DCOM PROTOCOLS
SET=DATACOM	DEFAULT=28 IBM2780
UNIT=Cn	
PRCL=	27 IBM2770 28 IBM2780 38 IBM3780 U1 U100 HA HASP HI HOST INT

CMND=SET	RID ADDRESS
SET=DATACOM	
UNIT=Cn	
PRCL=U1 U100	
OPTION=NO	
DISABLE=NO	
RATE=HIGH	
RID ADDRESS=	

CMND=SET	SID ADDRESS
SET=DATACOM	
UNIT=Cn	
PRCL=U1 U100	
OPTION=NO	
DISABLE=NO	
RATE=HIGH	
RID ADDRESS=nn	
SID ADDRESS=	

CMND=SET	U100 RECORD LENGTH
SET=LENGTH	DEFAULT=NO MORE
SET=	
	D1 DCOM1
	D2 DCOM2
	D3 DCOM3
	D4 DCOM4

CMND=SET	RECORD LENGTH
SET=LENGTH	DEFAULT=64
SET=D1 DCOM1	
LENGTH=	

CMND=SET	RECORD LENGTH
SET=LENGTH	DEFAULT=80
SET=D2 DCOM2	
LENGTH=	

CMND=SET	RECORD LENGTH
SET=LENGTH	DEFAULT=132
SET=D3 DCOM3	
LENGTH=	

CMND=SET	RECORD LENGTH
SET=LENGTH	DEFAULT=1050
SET=D4 DCOM4	
LENGTH=	

CMND=SET	LONG RECORD BREAKUP
SET=DATA COM	DEFAULT=NO
UNIT=Cn	
PRTCL=(any BSC)	NO
OPTION=NO	YES
DISABLE=YES	
RATE=HIGH	
TRANSLATION=NO	
SUPPRESS=NO	
TRANSPARANCY=NO	
DELETE=YES	
MULTI-POINT=NO	
OPTION=NO	
MULTI-RECORD=NO	
BREAKUP=	

A.2.5. WRITE and PRINT

CMND=WRITE (or PRINT)	PACKING TYPE
OPTIONS=NONE	DEFAULT=H2000
JOB NAME=ALL JOBS	
FORMAT=ANY	NONE
SKIP=NO	H2000 VLR-VLB
OPTIONS=NO MORE	H6000
(PRINT only)	
OPTIONS=PK-PACKING	4X3 ICL
PACKING=	2970 ICL
	1900 ICL

CMND=WRITE (or PRINT) DATA BLOCK SIZE
OPTIONS=NONE DEFAULT=1024
JOB NAME=ALL JOBS
FORMAT=ANY
SKIP=NO
OPTIONS=NO MORE
 (PRINT only)
OPTIONS=PK-PACKING
PACKING=VLR-VLB
DATA BLK SIZE=

B. Known Situations

- Data communications have not been fully certified due to availability of pertinent hardware resources.
- SYSGEN prompts in reference to Kennedy disk drives and Univac keystations have been deleted from the SYSGEN prompt sequence. However, if the SYSGEN procedure is being logged to a printer, these prompts will still appear in the log listing. They should be ignored.
- Under some conditions, keystation number 63 is not recognized by the system. It is therefore recommended that a terminal is not strapped as 63.

9896.2
UP-NUMBER

SPERRY 1900/10 and XL/40 LEVEL 11.R1.00
Release Description

B-2
PAGE

C. User's Program Library

C.1. THE LIBRARY

The User's Program Library is a magnetic tape that stores standard programs for routines common to the needs of many customers. The library contains format programs, subprograms, table programs, and label programs that have general applications.

C.1.1. Loading the Program Library

Before any program contained on the library tape can be referenced, it must be loaded onto the system. Ordinarily the supervisor restores the entire library from tape to disk at one time so that individual programs are available as needed. The process actually restores a machine-language version of data entered originally from a source input form or document. The machine-language version is compiled from the source language, and this is called a **compile**. To restore all library program compiles from the tape to disk, follow these steps:

1. Mount the User's Program Library tape on one of the system tape drive units.
2. Enter the Supervisor Mode.
3. Select SET from the options available on the Supervisor work menu. Set the input device to T1 if the library tape is mounted on the primary tape unit (or to T2, T3, or T4 for the appropriate unit).
4. Display the Supervisor work menu again and select the RESTORE command. Then respond to system prompts as shown in the sample dialogue below. Default responses are shown in parentheses.

PROMPT	RESPONSE	MEANING
CMND=	RESTORE	
NAME TYPE=	(COMPILE)	The system asks for the type of data to be restored; programs are sorted in compiled form on the library tape. Type COMPILE and press the FIELD REL key.
NAME=	(ALL NAMES)	Restore the entire User's Program Library by pressing the FIELD REL key to default to ALL NAMES.
COMPILE=	(ALL)	The system requests the name of the section within the library in which the compiled program is stored. Is selection a compiled format program, subprogram, table program, or label program? Press the FIELD REL key to default to ALL.
OPTIONS=	NONE	The system requests specific tape input options. Finish answering the prompts by defaulting with the FIELD REL key to further inquiries. The contents of the User's Program Library are now restored to disk, ready for use.

C.1.2. Loading Specific Library Programs

To load only those programs that are to be referenced during a given data entry operation, enter the name of the compiled program and its type (format, table, label, or subprogram). Then answer the prompts of the system. Refer to C.3, C.4, C.5, or C.6 for the name and type of programs to load.

For example, to load only the FORMAT program, the CALENDAR subprogram and the STATE-TB table, operator responses to system prompts might be as follows:

PROMPT	RESPONSE	MEANING
CMND=	RESTORE	
NAME TYPE=	COMPILE	The system requests the type of data to be restored.
NAME=	FORMAT	Enter the name of the program to be used.
COMPILE=	FORMAT	A compiled format program that is stored in the format portion of the program library is needed.
COMPILE=	(NO MORE)	If no more subprograms, label programs, or table programs of the same name (FORMAT) are required from the tape (if such programs exist), press the FIELD REL key to default to NO MORE.
NAME=	CALENDAR	Enter the name of the program to be used.
COMPILE=	SUBPROGRAM	A compiled subprogram that is stored in the subprogram portion of the program library is desired.
COMPILE=	(NO MORE)	If no more formats, label programs, or table programs having the same name (CALENDAR) are required from the tape (if such programs exist), press the FIELD REL key to default to NO MORE.
NAME=	STATE-TB	Enter the name of the program to be used.
COMPILE=	TABLE	The compiled program used to prepare a table is required; it is stored in the table portion of the program library.
COMPILE=	(NO MORE)	If no more formats, subprograms, or label programs of the same name (STATE-TB) are required from the tape (if such programs exist), press the FIELD REL key to default to NO MORE.

PROMPT	RESPONSE	MEANING
NAME=	(NO MORE)	If no more compiled programs are required from the tape, press the FIELD REL key to default to NO MORE.
OPTIONS=	(NO MORE)	Finish answering the prompts of the system by defaulting with the FIELD REL key to tape input options. The programs requested are restored to disk. In each case, enter the name of the program as well as the type of program (i.e., where it is stored in the program library).

C.1.3. Using the Program Library

To use programs from the library, specific format program names may be entered. Or programs may be called with instructions written as COBOL statements in format programs, depending on the type of program being used (e.g., subprograms, tables, labels). Usually, all programs are loaded from tape to disk and kept on the system during the operations of the day so that any program will be available when referenced in a format program.

But if programs are restored selectively, read the source documents or this section to determine which programs to load. For example, before referencing a table containing employee names and addresses, restore the TABLE format and enter the necessary table values.

The introductions to the subprogram, table, and label subsections in this manual include instructions on how to access a library program.

More detailed information about writing a format program that calls a library program is included in the appropriate **System Supervisor's Reference Manual**.

C.2. SOURCE LISTINGS

Source language is entered into the system from a source input form. Besides program compiles, the library tape also contains most programs in the form of the source data batches that were originally entered from source input forms. That is, the tape contains a compiled program called DATECONV and also a job called DATECONV. A **source listing** is an exact record of the job that was entered.

In order to facilitate easy identification of library source jobs, all jobs in the library have been given a batch number of 50000.

A few programs are available only in machine language and are maintained on tape in compiled form but not in a source language version.

To print out a copy of the available source listing, proceed as follows:

1. Mount the User's Program Library tape on one of the system tape drive units.
2. Enter the Supervisor mode.
3. If the contents of the library are already restored to disk, go to step 4. If not, select SET from the options available on the supervisor work menu. Set the input device to the tape drive unit on which the User's Program Library tape is mounted (e.g., T1 for the primary unit), and restore source batches from the library tape.
4. Set the 01 output device to LP for line printer or SP for station printer, also entering a printer number for station printer. Answer additional prompts as needed.
5. Display PAGE TWO of the Supervisor work menu and select the PRINT command. Then respond to system prompts as follows:

PROMPT	RESPONSE	MEANING
CMND=	PRINT	
OPTIONS=	(NONE)	Press the FIELD REL key to default.
JOB NAME=	_____	Enter the name of the program for which a source listing is desired (e.g., DATECONV).
BATCH=	(ALL)	Usually a given library program has just one batch (batch number 50000); enter the batch number or press the FIELD REL key to default to ALL.
JOB NAME=	_____	Enter additional program names or press the FIELD REL key to default to NO MORE.
FORMAT=	_____	Press the FIELD REL key to select the default ANY to print each listing using the format that was used to create it.

PROMPT	RESPONSE	MEANING
OPTIONS=	(NO MORE)	Press the FIELD REL key to default to NO MORE. Usually, to print a source listing, pressing the FIELD REL key is used to default to any further system prompts.
DELETE OLD=	(NONE)	Press the FIELD REL key to retain source data on disk. To display a source listing at a terminal, set the 01 output device to DS (Display Station); respond to the spooling prompt as desired.

C.3. FORMAT PROGRAMS

A format program is a set of checkbox field descriptions and associated COBOL procedures that describe data being entered or verified from a source document and specify how the data is to appear when it is output. The library format programs provide routines for entering data under different formats. A format program is used to create other format programs, subprograms, table programs, and label programs. A format program that will call or reference a subprogram, table, or label program may also be written.

For example, to write a new format program, use **FORMAT**, **FORMATI** or **FORMATIS** to control the source data entry. It is recommended that **FORMATIS** be used to enter new format programs because **FORMATIS** includes the expanded capabilities of the most recent version of the operating system. **FORMATIS** may also be used to search and modify programs entered previously under **FORMAT** and **FORMATI**. Where the three format programs are not interchangeable, exceptions are noted within the program descriptions.

Among the format programs in the library are those used to create new subprograms, table programs, and label programs. For example, the format program called **TABLE** is used to create a table containing customer names and account numbers. If called by the format program, the new table (**MAINTABL**) and the data in it would be accessible during data entry operations.

The **System Supervisor's Reference Manual** contains detailed information about writing format programs.

All library format programs appear alphabetically in the following list and are described in detail in the following sections.

Name	Purpose
AUPROMPT	Enters unformatted AUTOPROMPT batches.
DATATYPE	Defines character subsets for data types.
DATECONV	Creates date and 24-hour clock conversion tables.
DK-CONV	Converts disk address and segment number to a cylinder and head, and vice versa.
FORMAT	Enters format programs.
FORMATI	Enters format programs.
FORMATIS	Enters format programs.
FREEFORM	Enters unformatted 80-character records.
F80, F128 F133 and F134	Are freeform programs for accessing communicated records.
INDEX	Enters data used in creating indexed file header.
LABEL and LABEL-TR	Enters label programs.
MEM11	Generates current memory usage analysis for display or report printing.
NC-SORT	Assembles sort key of up to 16 noncontiguous fields in ascending or descending order.
SORT	Creates sort file headers for data sort.
SUBPROG	Enters subprograms.
TABLE	Enters tables.
TRANSLAT	Defines translation tables.
VFU-FORM	Creates electronic VFU record.

C.3.1. AUPROMPT

Application To create a batch to be used by the AUPROMPT command in the Supervisor or Command Manager Mode.

Program Description Each record can be a maximum of 40 characters long. The format is freeform, with each record containing one field in alpha shift. Any character may be entered. The program has only one level.

Operating Procedures

1. Open batch in the ENTER mode using the AUPROMPT format program.
2. Enter AUPROMPT program records. Each prompt response must be contained in a separate record.
3. After all records are entered, close the batch by pressing the CTRL and CMND keys simultaneously; then enter C.

Comments The AUPROMPT statements are entered and stored in the system as a data batch; they are not compiled and can be modified or deleted in the same way as any regular data batch. To get a hard copy of an AUPROMPT program that has been created, print the listing on the line printer, specifying job name and batch number.

See the **System Supervisor's Reference Manual** for instructions on how to plan an AUPROMPT program.

C.3.2. DATATYPE

Application To create records to modify the attributes of the data type subset specified in checkbox 5 of the coding form. The records are written to tape and loaded as part of the system during the system load operation.

Necessary Tables Before using the DATATYPE program, first prepare tables using the following table formats from the program library (see C.5.): HEX2DEC, NTBL, TYPTBL, YNTBL, YTBL.

**Program
Description**

The program has six levels:

- H Identifies the attribute module to be modified.
- I Initializes character registers with the current attributes.
- 1 Selects the data type to be modified.
- 2 Displays the entire character set, 32 characters at a time, with a Y if the corresponding character exists in the system definition of the table, N if it does not.
- Y Creates the output records.
- Z Closes the batch.

The batch contains eight 64-character output records when completed.

**Operating
Procedures**

1. Open batch in ENTER mode using the DATATYPE format program.
2. Enter Name (NAME), Version (VERSION), Release Number (REL-NO) and Identification (ID). Enter the name ATTRIBITS to modify an EBCDIC system; enter the name ATTRIBASC to modify an ASCII system.
3. Enter data type to be modified. The data types or standard characters for each type are:
 - N Digits 0 through 9.
 - A Uppercase alphabetic characters A through Z.
 - a Lowercase alphabetic characters a through z.
 - S Digits 0 through 9 (signed decimals).
 - P Punctuation characters.
 - B Blank (space).
 - C COBOL characters.
 - H Hexadecimal number set (0 through 9 and A through F).

1-8 user character subsets:

- 1 As currently assigned, underscored 0 through 9.
 - 2-8 As currently assigned, the number of the data type itself is the sole character included in the set.
4. When the character set is displayed, with Y or N under each character, enter Y for any character that is to be included, and space for any character that is to remain as displayed.
 5. After all characters have been displayed, select another data type and repeat step 4.
 6. Enter E to format the output records and close the batch.
 7. Output the batch to tape using the WRITE command.
 8. Read the tape into the system at the end of system generation to replace the original ATTRIBITS or ATTRIBASC module.

C.3.3. DATECONV

Application

To create date and 24-hour clock conversion tables.

**Program
Description**

The program has seven levels:

- O** Initializes the year and registers.
- H** Creates date conversion table header for the year.
- I** Creates date conversion table header for the month and day.
- D** Creates the date table.
- T** Creates the 24-hour clock conversion table header.
- C** Creates the 24-hour table.
- Z** Closes the batch.

**Operating
Procedures**

1. Open the batch in the ENTER Mode using the DATECONV format program.
2. Enter the year for which the calendar conversion table is wanted.
3. Both the date and 24-hour tables will be created and the batch will automatically close.
4. Use the PRINT command to output the batch to the line printer.

Comments

The program is formatted for a 132-character printer. Part of the output will be lost if printed on a printer with fewer characters.

The printing can be accomplished without print control, requiring one 11- by 14-inch page, or with one character of print control specified, requiring two pages of output.

C.3.4. DK-CONV

Application

To convert a disk address and segment number on the ISS disk (LCD) to a cylinder and head, and vice versa.

**Program
Description**

The program has three levels:

- 1 Handles prompting for processing mode.
- 2 Handles prompts and processing to convert a segment number and disk address to a cylinder and head. Calls HEX-CONV.
- 3 Handles prompts and processing to convert a cylinder and head to a segment and disk address. Calls DEC2HEX.

**Operating
Procedures**

1. Open a batch in the Enter mode using the format program called DK-CONV. All conversion information generated by this program will be recorded in this batch.
2. Key "C" to convert to a cylinder and head or "D" to convert to a segment and disk address.

3. If "C" was keyed in Step 2, respond to the prompts for a two-digit hexadecimal segment number and a three-digit hexadecimal disk address. The associated cylinder, head, and AU numbers will then be calculated and displayed, and the user will be prompted to continue (FIELD REL) or return to the mode menu (M).
4. If "D" was keyed in Step 2, respond to the prompts for a three-digit decimal cylinder number and a two-digit decimal head number. The associated segment number and disk address range will then be calculated and displayed, and the user will be prompted to continue (FIELD REL) or return to the mode menu (M).
5. Key "RESET/CMND C" to close the batch.

Comments

This program calls the subprograms DEC2HEX and HEX-CONV contained in the User's Program Library.

C.3.5. FORMAT

Application

To enter, search, or modify source code to create format programs. Enter a format program as a job under control of FORMAT. The input document is the source input coding form filled out according to syntax rules described in the appropriate **System Supervisor's Reference Manual**.

NOTE:

Do not use FORMAT for entering file inquiry, sequential files, or report writer formats. Instead, use the FORMATIS program when creating a new format program and when using the file inquiry, report writer, and sequential file features.

FORMAT accepts all field descriptors in sequential order (field name, field size, shift, data type, etc.) and output specifications (output positions, justification, fill, etc.) from the checkbox portion of the source input form. ALLOW statements ensure that only valid checkbox parameters are entered. After entering the checkbox data for a given field, FORMAT can accept COBOL procedure statements for that field. After entering the entire format program, the program must be compiled under control of the Supervisor COMPILE command to convert source code into executable object code.

After an error-free compilation, the system automatically catalogs the object program and writes it to disk. It can then be used at any future time with the Enter, Verify, Search/Modify or Update Modes. Formats are also used during operation of the Supervisor WRITE, PRINT, and INPUT functions.

Program**Description**

The program has four levels:

- O** Creates the first header record.
- V** Creates all other header records.
- F** Creates field descriptor records.
- Z** Closes the batch.

**Operating
Procedures**

1. Fill out the source input coding form according to instructions included in the appropriate **System Supervisor's Reference Manual**.
2. Open a batch in the Enter Mode using the format program called FORMAT.
3. Enter the first header record. In the first position of the first header record, FORMAT automatically places the letter P, which is displayed on the screen; it does not need to be coded or entered.
4. Enter the field descriptor records. In the first position of each record, FORMAT automatically places the letter F; as in step 3, this character does not need to be coded or entered.

5. To move to the next program level, enter END LEVEL in the last procedure field of the current level. In the procedure field this must be the only statement and must begin in the first position.
6. Enter the header record for the next program header. Header records for all levels except the first must have the letter V in first position. FORMAT automatically places a V in that position; it is displayed on the screen; it does not need to be coded or entered.
7. To close the batch, enter END in the procedure field; this causes the system to move to level Z automatically. In the procedure field the END statement must begin in the first position and must be the only statement in the field.
8. To compile the batch, use the Supervisor command COMPILE.

Keying Errors

If the END LEVEL statement is miskeyed (misspelled, for example, as END LEVL or ENDLVL) and then the FIELD REL key is pressed, the system will select a level based on the incorrect entry, but the system will still display level F. To correct the program level selection:

1. Press the REC<-- key to return to the previous record where the error exists.
2. Press the FLD--> key to move the cursor to the procedure field.
3. Correct the END LEVEL statement. Press the FIELD REL key.

4. Even though the correction is made, the system still displays level F. Before entering any information into the new record, press the CTRL and CMND keys (or just the CMND key on the typewriter-style terminal keyboards); then press PROG and enter the code letter V for a new level header. The system will now select and display level V.

The first field in a record is normally a nonkeyed field into which the format program inserts an identifying code letter (P, V, F, etc.). To correct this field manually for V or F records proceed as follows:

1. Press the AUTO OFF key.
2. Move the cursor to the first position in the record and enter the appropriate code letter. After making the correction, the automatic feature is again in effect.

NOTE:

The first record of a format entry (P record) cannot be changed with the AUTO OFF feature. If an attempt is made, the message "INVALID KEY AT 1ST FIELD" is displayed.

**Adding New
Records to a
Closed Batch**

When the END statement in the procedure field of a format program is entered, the system reads that statement and then automatically selects level Z to close the batch. If additional records must be entered, follow one of two steps, depending on whether only a few or many records must be added.

**Adding a Few
Records**

To add only a few records, open the batch in the Search/Modify Mode and search for the END statement.

1. Insert another line above the END statement. Enter the new record and repeat this step for each new record to be added. The original END statement remains intact while new records are inserted above it.
2. To add an additional level, insert a line above the END statement and enter END LEVEL in the new procedure field. Press the FIELD REL key. Press the INSERT key again. Level V is displayed. Inserts for each record needed can now be made. As in step 1, your original END statement remains intact.

Adding Many Records

To add more than a few records to the batch, open the batch in the Search/Modify Mode, search for the end statement, and proceed as follows:

1. **Change** the END statement to END LEVEL. Press the FIELD REL key.
2. Close the batch and output it to tape using the WRITE command. Then return the batch to disk using the INPUT command and the appropriate format program (FORMAT, FORMATI, or FORMATIS).
3. Reopen the batch in the Enter Mode and respond YES to the "CONTINUE=" prompt. Enter any additional records. Enter END in the procedure field of the last record and press the FIELD REL key. The system then automatically selects level Z to close the batch.

NOTE:

When using the above procedure to add a number of records to a batch, do not remove the V or F in the first field position of the records. Using the INPUT command leaves the AUTO feature OFF, so it is possible to accidentally remove the identifying code letters. Without these letters, the compilation will fail.

Comments

Level 0 has a tab stop at the date field; level F has tab stops at the output position and procedure fields.

Programs created using FORMAT are not interchangeable with programs created using FORMATI or FORMATIS.

C.3.6. FORMATI

Application

To enter, search, and modify source code to create format programs. FORMATI can be used to create format programs that access indexed files or are usable for report writing, as well as for creating any format program possible with FORMAT.

NOTE:

Use the FORMATIS program instead of FORMAT or FORMATI when creating a new format program.

**Program
Description**

The program has seven levels:

- O** Creates the first header record.
- E** Creates Environment Division detail records.
- R** Creates Data Definition header records.
- D** Creates Data Definition detail records.
- F** Creates field descriptor records.
- V** Creates all other header records.
- Z** Closes the batch.

**Operating
Procedures**

1. Fill out the source input coding form. Instructions for doing so are included in the appropriate **System Supervisor's Reference Manual**.
2. Open a batch in the Enter Mode using the format program called FORMATI.

3. Enter the first header record. In the first position of the first header record, FORMATI automatically places the letter P, which is displayed on the screen; the letter does not need to be coded or entered.
4. Enter detail records.
5. To move to the next program level, enter END LEVEL in the last procedure field of the current level. This must be the only statement and must begin in the first position in that field.
6. To close the batch, enter END in the procedure field; this causes the system to move to level Z automatically. In the procedure field, the END statement must begin in the first position and be the only statement in that field.
7. To compile the batch, use the Supervisor command COMPILE.

Keying Errors

To correct a keying error that causes the system to select the wrong program level, see the information about **keying errors** under the FORMAT program description (C.3.4).

Adding New Records to a Closed Batch

After closing a batch by entering END in the procedure field, use the following procedure to add more records to that batch. (This procedure cannot be applied to programs created under earlier FORMATI versions.)

1. Open the batch in the Search/Modify Mode and change the END statement to END LEVEL. Close the batch.
2. Reopen the batch in the Enter Mode and respond YES to the CONTINUE prompt. Add new records as needed.

NOTE:

When using this procedure to add more records to a batch, do not remove the V or F in the first position of the records. Using the INPUT command leaves the AUTO feature OFF, so it is possible to accidentally remove the identifying code letters. Without these letters, the compilation will fail. Programs created using FORMATI cannot use the INPUT command for either indexed files or report writing.

Comments

Level 0 has a tab stop at the date field; level F has tab stops at the output position and procedure fields.

Programs created using FORMAT are not interchangeable with FORMATI-created programs for the file inquiry and report writer features.

C.3.7. FORMATIS

Application

To enter, search, or modify source code in order to create format programs. When creating an all new format program, use FORMATIS. It can be used to create format programs that access either indexed or sequential files, as well as for creating any format program possible with FORMAT or FORMATI.

**Program
Description**

The program has six levels:

- O** Creates the first header record.
- E** Creates Environment Division detail records.
- V** Creates both the Data Definition header records and the Procedure Division header records.
- D** Creates Data Definition Division detail records.
- F** Creates field descriptor records.
- Z** Closes the batch.

Operating Procedures

1. Fill out the source input coding form. Instructions for doing so are included in the appropriate **System Supervisor's Reference Manual**.
2. Open a batch in the Enter Mode using the format program called FORMATIS.
3. Enter the first header record. In the first position of the first header record, FORMATIS automatically places the letter P, which is displayed on the screen. The letter does not need to be coded or entered.
4. Enter detail records.
5. To move to the next program level, enter END LEVEL in the last procedure field of the current level. This must be the only statement and must begin in the first position in that field.
6. To close the batch, enter END in the procedure field; this causes the system to move to level Z automatically. In the procedure field the END statement must begin in the first position and must be the only statement in that field.
7. To compile the batch, use the Supervisor command COMPILE.

Keying Errors

To correct a keying error that causes the system to select the wrong program level, see the information about keying errors in C.3.4.

Adding New Records to a Closed Batch

After closing a batch by entering END in the procedure field, use the following procedure to add more records to that batch.

1. Open the batch in the Search/Modify Mode and change the END statement to END LEVEL. Close the batch.

2. Reopen the batch in the Enter Mode and respond YES to the CONTINUE prompt. Add new records as needed.

NOTE:

When using this procedure to add more records to a batch, do not remove the V or F in the first position of the records. Using the INPUT command leaves the AUTO feature OFF, so it is possible to accidentally remove the identifying code letters. Without these letters, the compilation will fail.

Comments

Level 0 has a tab stop at the date field; level F has tab stops at the output position and procedure fields.

Programs created using FORMAT and FORMATI are not interchangeable with FORMATIS-created programs for using sequential files.

C.3.8. FREEFORM

Application

To enter unformatted 80-character records.

**Program
Description**

The program contains one level and allows any character to be entered.

**Operating
Procedures**

1. Open the batch in the Enter Mode using the FREEFORM program.
2. Enter records. If fewer than 80 characters per record are entered, the rest of the field will be filled with blanks.
3. Close the batch by pressing the CTRL and CMND keys simultaneously and then entering the letter C (or press the CMND key plus C on typewriter-style terminal keyboards).

C.3.9. F80, F128, F133, and F134

Application

To search, modify, or transmit to tape, data that has been entered on the system by means of data communication.

F80 defines an 80-character record conforming to punched card requirements.

F128 defines a 128-character record conforming to an IBM 3741 protocol emulation characteristics; the IBM 3741 data station is limited to receiving records no longer than 128 characters (nonexpanded mode).

F133 defines a 133-character record conforming to the line printer requirement of 132 characters and 1 print control character.

F134 defines a 134-character record conforming to the Houston Automatic Spooling Program (HASP) requirement of 132 characters and 2 print control characters.

Program Description

Each program contains one level and allows any character to be entered.

Operating Procedures

To Search/Modify:

1. Open the batch in the Search/Modify Mode using the appropriate program.
2. Search the records. If fewer than the maximum allowable characters are entered into a record, the rest of the field is filled with blanks.
3. Close the batch by pressing the CTRL and CMND keys simultaneously and then entering the letter C.

To output to tape:

1. Output communicated data to tape with the WRITE command using the appropriate program.

2. Input the data for further manipulation.

Comments

Each of these programs contains a DECLARE REGISTER SIZE 1 statement to make it compatible with DCOMDATA batch headers.

C.3.10. INDEX

Application

To create an indexed file header when building an indexed file through the Supervisor FILE MGMT command.

**Program
Description**

The program has two levels:

- 1 Enters general information about the indexed file being created.
- 2 Enters parameters relating to the primary key and each of nine alternative keys.

**Operating
Procedures**

1. Enter the Supervisor Mode and select CMND=INITIALIZE and FORMAT=INDEX.
2. The data area of the screen goes blank. Operator prompts are then displayed on the message line, and the entered responses are displayed on the first two data lines.
3. Enter the general header information.
4. Enter the parameters for each key in the file.
5. Close the program by pressing the CTRL and CMND keys simultaneously and then entering the letter C.

Keying Errors

To correct errors, follow these steps:

1. Use the CHR<-- key to correct a keying error within a field.

2. It is not possible to use REC<--- or FLD<--- when using the INDEX format program. Thus, if an error is made and the field is exited, close the program immediately by pressing the CTRL and CMND keys simultaneously and then entering the letter C (or press the CMND key plus C on typewriter-style terminal keyboards).

If the header and primary key parameters have not already been entered completely, the system displays an error message and does not create the header. In this case simply select the INITIALIZE command again.

If the header and primary key parameters have been entered completely, delete the header before reselecting the INITIALIZE function. While still in the Supervisor File Mgmt Mode, select CMND=DELETE and specify PARTIAL. Then select CMND=INITIALIZE and enter the correct data.

C.3.11. LABEL AND LABEL-TR

Application

To enter source label programs. Use LABEL-TR when a special translation between information codes is necessary; use LABEL when translation is not necessary.

Program Description

Except for the translation field in LABEL-TR, both programs are the same. There are four levels:

- O** Creates the first header record.
- V** Creates all other header records.
- F** Creates field descriptor records.
- Z** Closes the batch.

Operating Procedures

1. Open the batch in the Enter Mode using the LABEL or LABEL-TR format program.
2. Enter the first header record. In the first position of the first header record, the program automatically places the letter L, which displays on the screen; the letter does not need to be coded or entered.

3. Enter the field descriptor records. In the first position of each record, the program automatically places the letter F; as in step 2, the letter does not need to be coded or entered.
4. To move to the next program level, enter END LEVEL in the last procedure field of the current level. This must be the only statement and must begin in the first position in that field.
5. Enter the header record for the next program level. Header records for all levels except the first must have the letter V in the first position. The program automatically places a V in that position; it displays on the screen; it is not necessary to code or enter it.
6. To close the batch, enter END in the procedure field; this causes the system to move to level Z automatically.
7. To compile the batch, use the supervisor command COMPILE.

Comments

To correct keying errors, see the paragraph on Keying Errors under the FORMAT program description.

LABEL and LABEL-TR are not interchangeable. A batch entered under one format cannot be searched or modified under the other.

Level 0 has a tab stop at the date field; level F has tab stops at the output position and procedure fields.

In each case where LABEL or LABEL-TR inserts a code letter in the first position of a record (L, V, etc.), the compilation will fail if the letter is removed accidentally.

C.3.12. MEM11

Application

To generate, in File Mgmt Mode, current memory usage analyses for display and printing in report form. MEM11 is a diagnostic tool and is not intended for general use during a production run of the system.

Version

Compatibility

MEM11 is a version-dependant program. It can only be used with 11.R1.00 systems which, during SYSGEN, have been given a version number of the exact form "11.R1.xx", where xx is the patch level number.

Program

Description

The program has 10 levels and calls one subprogram, MEM11S, which sets up A-registers, 1 through 28, for use by the main format.

The levels are as follows:

- A** Displays information on the use of the program and calls MEM11S to set up size information in the A-registers.
- 1** Outputs the first header record to disk.
- 2** Prompts the operator to key a comment line of up to 100 characters to be included among the header lines.
- 3-4,**
- 6-8** Generate 8 more lines of header information.
- M** Main loop; constructs the actual memory analysis records, displays the analysis information and writes it to disk; calls MEM11S to determine memory sizes.
- Y** Processes errors detected during calls to the MEM11S subprogram.
- Z** End-of-program processing.

Operating Procedures

Each pass of the program (see step 3 of the operating instructions) will yield an instantaneous snapshot of current memory usage. The analyst can obtain exact steady state memory usage values for virtually any system function. However, functions which have extremely short duration peaks may prove to be difficult to map. Manual keying of passes, even as fast as possible, will still be much slower than computer execution time for certain events. The bytes indicated may be lower than actual, in these cases.

Also, note that only the actual memory use is reflected. It may be possible to run out of room even though there is memory available due to the mapping environment restrictions.

If an error is encountered in the subprogram (such as tenant length error, invalid tenant table entry, invalid version number, not enough A-registers), an ALARM will be issued with a message. If the error occurs prior to keying the comment line, Level Y will be selected and no report will be generated. If the error occurs once the report generation starts, the report will continue, but the PASS counter will be zeroed on the report to indicate that an error was encountered. The numbers generated on error passes may be incorrect.

The specific operating instructions are given below:

1. File Inquiry mode must be used to open a batch using the MEM11 format. (Use of any other mode will result in the automatic selection of Level Y and the close of the batch.) Do not simultaneously open more than one terminal using this program name. To do so would result in inaccurate memory usage information.

2. The first level displays information about the program. Assure that the analyzer format is the correct program for the version installed, and press any key except "Z" to proceed to the report second level. Keying a "Z" will result in a batch close.
3. Key a comment line (up to 100 characters) for the printed report. After the comment is entered, the main level will be entered. This level will loop with each press of a key until the batch is closed by keying a "Z".
4. The analyst should carefully note the current system activity captured by each pass of the report, for the later comparison with the hard-copy output.
5. To obtain a hard-copy report, merely print the batch records generated. "SKIP PROC=YES" must be specified when printing the report.
6. The batch should be closed by typing the letter "Z". This will enable a Level Z close with no batch replay. A command C may cause checkpoint replay and give erroneous results in the last few records.

**Output
Description**

The MEM11 program automatically outputs information to the screen and to disk, and will produce a formatted report upon WRITE or PRINT of the disk batch.

Screen Output:

The memory analysis is automatically output to the screen. All sizes are in decimal bytes. The screen layout is given in Figure C-1.

PASS=nnn	HIT ANY KEY TO CONTINUE		
	EM	SM	TOTAL
FIXED AREAS	XXXXXX	XXXXXX	XXXXXX
OVERLAYS	XXXXXX	XXXXXX	XXXXXX
FORMATS	XXXXXX	XXXXXX	XXXXXX
TWAs	XXXXXX	XXXXXX	XXXXXX
BUFFERS AND SCL	XXXXXX	XXXXXX	XXXXXX
TOTAL USED	XXXXXX	XXXXXX	XXXXXX
TOTAL UNUSED	XXXXXX	XXXXXX	XXXXXX

Figure C-1.
Memory Analysis Display

Printed Report:

When a WRITE or PRINT is performed on the data batch produced by MEM11, the printed report is generated. An example of the report is given in Figure C-2.

MEMORY ANALYSIS FOR VERSION 11										DATE:83220									
MEMORY ANALYZER TEST 8/3/83																			
TOTAL MEMORY AVAILABLE IS:					409088 EM					144688 SM					523776 TOTAL				
FIXED MEMORY ITEMS USE:					1584 EM					71244 SM					72828 TOTAL				
MEMORY ANALYSIS TWA AND FORMATS USE:					8620 EM					0 SM					8620 TOTAL				
PASS:	OVERLAYS			FORMATS			TWAS			BUFFERS AND SCL			TOTAL USED MEMORY			TOTAL UNUSED MEMORY			
:	EM	SM	EM+SM:	EM	SM	EM+SM:	EM	SM	EM+SM:	EM	SM	EM+SM:	EM	SM	EM+SM:	EM	SM	EM+SM:	
1	4040	3784	7794	0	0	0	1386	0	1386	0	646	646	6980	75674	82654	393488	39044	432502	
2	4040	4872	10882	0	0	0	1386	0	1386	0	644	644	6980	78780	85740	393488	35908	429396	
3	3580	3558	7138	0	0	0	130	0	130	0	628	628	5294	75430	80724	395474	39258	434432	
4	5984	3558	9542	0	0	0	130	1482	1312	0	2208	2208	7698	78492	85890	392770	36496	429266	
5	5984	3558	9542	0	0	0	130	1280	1440	0	3568	3568	7698	79650	87348	392770	35038	427808	
6	5984	3558	9542	0	0	0	130	1378	1508	0	4928	4928	7698	84408	88806	392770	33580	426350	
7	5984	3558	9542	0	0	0	130	1476	1606	0	6304	6304	7698	82582	90280	392770	32406	424876	
8	6444	3784	10498	0	0	0	1386	1476	2862	0	6322	6322	9384	82826	92240	394084	34862	422946	
9	5984	3558	9542	156	0	156	1436	1476	2912	0	6304	6304	9460	82582	94742	394308	32406	423444	
10	5984	3558	9542	156	0	156	1520	1476	2996	0	6320	6320	9244	82598	94842	394224	32090	423344	
11	5984	3558	9542	0	0	0	130	1476	1606	0	6320	6320	7698	82598	90296	392770	32090	424860	

Figure C-2.
Memory Analysis Printed Report

Disk Record Layout:

The actual data record produced to disk has all the information in the printed report, plus some mapping information not displayed or output in the printed report. The disk record layout is given in Figure C-3 which follows.

1-6	FIXED AREA FM
7-12	FIXED AREA SM
13-18	FIXED AREA TOTAL
19-24	OVERLAYS EM
25-30	OVERLAYS SM
31-36	OVERLAYS TOTAL
37-42	FORMATS EM
43-48	FORMATS SM
49-54	FORMATS TOTAL
55-60	TWAs EM
61-66	TWAs SM
67-72	TWAs TOTAL
73-78	BUFFERS AND SCL EM
79-84	BUFFERS AND SCL SM
85-90	BUFFERS AND SCL TOTAL
91-96	TOTAL USED EM
97-102	TOTAL USED SM
103-108	TOTAL USED TOTAL
109-114	TOTAL UNUSED FM
115-120	TOTAL UNUSED SM
121-126	TOTAL UNUSED TOTAL
127	PASS CHARACTER
128-133	MEM11R1 SPACE EM
134-139	MEM11R1 SPACE SM
140-145	MEM11R1 SPACE TOTAL
146-151	TMT GROWTH
152-157	CMT GROWTH
158-163	SM TENANTS GROWTH
164-169	RME RESIDENT REGISTER COUNT
170-175	TME RESIDENT AND SM TENANTS REGISTER COUNT
176-181	TME TMT REGISTER COUNT
182-187	TME CMT REGISTER COUNT
188-193	TME OVERLAY REGISTER COUNT
194-199	TME TFMP MAP REGISTER COUNT
200-205	TWA NUMBER OF MAX TMT TWA
206-211	EM SYSTEM FLAGG
212-214	PASS NUMBER (0 IF ERROR ON PASS)

Figure C-3.
Disk Record Layout

Interpreting the Output:

As indicated in Figure C-2, memory is allocated among the following general memory areas, all of which are analyzed on a real-time basis with each new pass while running the MEM11 program.

For purposes of the following memory area descriptions, the headings used in the report (Figure C-2) are used to designate the general area.

1. MEMORY ANALYSIS TWA AND FORMAT:

- a. TWA used to run MEM11.
- b. MAIN FORMAT MEM11.
- c. SUBPROGRAM MEM11S.

2. FIXED AREAS:

- a. LOW CORE, RESIDENT, FIXED BUF OVERLAY, MSG HANDLER TWA, DISK MAPS FOR SYSTEM DISK, PHY IO AND MTS BUFFERS, CRT UPDATE TBL in SM.
- b. 400 bytes for SCL growth in SM.
- c. Two bytes for each TWA up to the tenant table TWA.
- d. Gap tenants in EM.

3. OVERLAYS:

Overlay tenants with user count of one or more.

4. FORMATS:

Format programs, subprograms, format tables, and label program with a user count of one or more. The MEM11 format program and MEM11S subprogram will not be included in these sizes.

5. TWAs:

Foreground and background Terminal Work Areas (TWAs) with a standard TWA structure, minus the two-bytes already calculated in the FIXED AREA. This starts at TWA 0 and goes up to the first buffer pool. The MSG HANDLER TWA is excluded from this area since it appears in the FIXED AREA, and the TWA required for running the MEM11 program is excluded since it is already in the MEMORY ANALYSIS TWA AND FORMATS section.

6. BUFFERS AND SCL:

- a. BUFFER POOLS.
- b. DISK MAPS for alternate disks.
- c. TENANT TABLE.
- d. SM BUFFER TENANTS.
- e. DCOM FIXED BUFFER.
- f. SYS TEST PKG BUF POOL.
- g. SYSTEM CONTROL LIST.

TOTAL USED MEMORY and TOTAL UNUSED MEMORY are also given. The space required for running MEM11R1 is not in either the used or unused totals.

Disk Record Mapping Information:

In addition to the information which goes to the screen and to the report, the format program also outputs, to disk only, a set of EM mapping information. The EM information given is:

- 1. Number of map registers required to map:
 - a. RME RESIDENT.
 - b. TME FROM LOW CORE THROUGH SM TENANTS.

- c. TME TWA (MAXIMUM TMT TWA).
 - d. TME CMT TWAs.
 - e. TME MAPPED OVERLAY.
 - f. TME TEMP MAPS (XMAP).
- 2. TWA number for maximum sized TMT TWA.
 - 3. EM system flag (1 if FM, 0 if SM).
 - 4. Maximum growth allowed for the following areas (in bytes).
 - a. SM TWAs and CMT area buffers.
 - b. SM tenants.
 - c. TMT maximum size TWA.

C.3.13. NC-SORT

Application

To create a sort file header, based on one sort key composed of up to 16 noncontiguous fields, for sorting data batches using the Supervisor SORT command. Total key length cannot exceed 328 characters.

The system used the output of this program to control the sorting of the data batches.

Program

Description

This program has five levels:

- 1 Enters general information about the sort file header (SF) being created.
- 2 Enters information about the sort key record.
- 3 and 4 Enter information about the fields (1 to 16) to be used for constructing the sort key.
- Z Closes the batch.

**Operating
Procedures**

1. Select the supervisor SORT command. Press the FIELD REL key, enter the file name, select CMND=INITIALIZE, and specify FORMAT=NC-SORT. The data area of the screen goes blank; the system then displays prompts on the message line and the responses for the operator on the first two data lines.
2. Enter general information about the indexed file header; the following prompts require an operator response:

SF DATA BATCH TYPE:

- 0 Directory.
- 1 Data batch.

SF RETAIN KEYED BATCHES:

- 0 Do not retain batch after the index is built.
- 1 Retain batch.

DATA RECORD LENGTH:

Enter the number of characters in the records being sorted.

SF DATA JOB NAME:

Enter the job name of the directory, batch or set of batches.

SF DATA BATCH NO.:

- 0 All batches.
- 1 1 to 46,999 = specific batch number, greater than 0 but less than or equal to 46,999.

SF PASSWORD:

Enter correct password (must match the supervisor password).

3. Enter information to define the key record; the following prompts require an operator response:

SF SORT RETAIN:

0 Do not retain batch after the sort
is completed.

SF DATA IN SORT:

0 Batch has not been sorted.
1 Batch was sorted previously.

SF KEY LTH:

Enter the number of characters in the
key (1 - 328).

SF DUPLICATES FLAG:

0 Duplicates not allowed.
1 Duplicates allowed.

SF NUMBER OF FIELDS:

Enter the number of fields in the key
(1 to 16).

SF SORT KEY RCD POS - OPTIONAL:

Enter 0 to 995 to specify position of
sort data record.

SF SORT DA RCD POS - OPTIONAL:

Enter 0 to 995 to specify position of
sort data record.

SF SORT JOB NAME - OPTIONAL:

No entry or job name.

SF SORT BATCH NO.:

Enter 1 to 46,999.

SF SORT SEQUENCE:

0 Ascending.
1 Descending.

4. Enter information about the 1 to 16 fields to be used for constructing the SORT key. The proper sequence must be determined; that is, enter information about the most important field first, least important last.

For each field, the system displays the two following prompts, where x represents the number of the field:

SF FIELD x RCD POS:

To specify the position where the desired field begins, enter 1 to 999 (but no number greater than the record length).

SF FIELD x LTH:

To specify the length of the field, enter 1 to 328; the total number of characters in all fields combined cannot exceed 328.

5. After entering all information, close the batch by pressing the CTRL and CMND keys simultaneously and then entering the letter C (or press the CMND key plus C on typewriter-style terminal keyboards).

Comments

When the sort sequence is specified, an ascending or descending numerical or alphabetical order is allowed in NC-SORT.

When executing the sort, the system offers two noncontiguous key options:

LIST DATA NON-C

This sort process operates from an internal work list. The output is an ordered file. LIST can only be selected if the POST KEYS function has already been performed.

IMMED DATA NON-C

This sort process is immediate and is used to initiate or cancel a sort on a specific key of a selected file. The output is an ordered file.

Keying Errors

Correcting an Error within Field

If an error is made, but the field has not been exited yet, move backward in the field by using the CHR<-- key. Then enter the correction.

Correcting an Error After Exiting Field

If an error is made and the field has been exited, the REC<-- or FLD<-- key cannot be used to reenter the field. Close the program, then reopen it, and move forward to the field containing the error.

Close the program by pressing the CTRL and CMND keys simultaneously; then enter the letter C (or press CMND plus C on typewriter-style terminal keyboards). If the header and primary key parameters have not been entered completely, the system displays an error message and does not create the header. In this case, select CMND=INITIALIZE.

If the header and primary key parameters have been entered completely, the system will already have created the header. Delete it before starting over with the correct data. In the Supervisor File Mgmt Mode, select CMND=DELETE and specify PARTIAL. Then select CMND=INITIALIZE and start over.

C.3.14. SORT

Application

To create a sort file header for sorting data batches using the Supervisor SORT command.

**Program
Description**

The program has two levels:

- 1 Enters general information about the data records being sorted.
- 2 Enters parameters relating to the primary sort key and each of nine alternative keys.

**Operating
Procedures**

1. Select the supervisor SORT command. Press the FIELD REL key, enter file name, select CMND=INITIALIZE, and specify FORMAT=SORT. The data area of the screen goes blank. Operator prompts are then displayed on the message line, and the responses entered are displayed on the first two data lines.
2. Enter the general header information.
3. Enter the parameters for each key to be sorted.
4. After entering all information, close the batch by pressing the CTRL and CMND keys simultaneously and then entering the letter C (or pressing the CMND key plus C on typewriter-style terminal keyboards).

Keying Errors

To correct keying errors, follow these steps:

1. Use the CHR<-- key to correct a keying error within a field.

2. Because it is not possible to use REC<-- or FLD<-- key when using the SORT format program, if an error is made and the field is exited, close the program immediately by pressing the CTRL and CMND keys simultaneously and then entering the letter C (or by pressing just the CMND key plus C on the typewriter-style terminal keyboards).

If the header and primary key parameters have not already been entered completely, the system displays an error message and does not create the header. In this case, simply select the INITIALIZE command again.

If the header and primary key parameters have been entered completely, delete the header before reselecting the INITIALIZE function. While still in the Supervisor Sort mode, select CMND=DELETE and specify PARTIAL. Then select CMND=INITIALIZE and enter the correct data.

C.3.15. SUBPROG

Application

To enter the source language of subprograms. A subprogram can be called by a format to perform such COBOL procedural functions as computing a check digit or converting a Julian date to calendar date.

Program Description

The program has three levels:

- 0 Creates the header record.
- 1 Enters the procedure statements.
- Z Closes the batch.

Operating Procedures

1. Fill out the source input coding form according to syntax rules given in the appropriate **System Supervisor's Reference Manual**.
2. Open the batch in the Enter mode using the format SUBPROGRAM.

3. Enter the header record. In the first position of the first record, SUBPROG automatically places the letter S, which is displayed on the screen; the letter need not be coded or entered. If the letter is removed accidentally, the compilation will fail.
4. Enter the procedure statements.
5. To close the batch, enter END in the procedure field; this causes the system to select level Z automatically. END must begin in the first position and must be the only statement in that field.
6. To compile the batch, use the Supervisor command COMPILE.

**Adding New
Procedures to
Close Batch**

After closing a batch by entering END in the procedure field, more procedure statements can be added to that batch only by following these steps:

1. Open the batch in the Search/Modify mode and search for the END statement. Delete the statement.
2. Output the batch to tape using the WRITE command.
3. Use the Supervisor command INPUT to input the batch.
4. Open the batch in the ENTER mode and respond YES to the "CONTINUE=" prompt. Enter additional statements.

C.3.16. TABLE

Application

To enter source language for tables.

**Program
Description**

The program has three levels:

- 0** Creates the header record.
- 1** Enters the elements of the table.
- Z** Closes the batch.

**Operating
Procedures**

1. Fill out the source input coding form according to syntax rules given in the appropriate **System Supervisor's Reference Manual**.
2. Open the batch in the Enter Mode using the format TABLE.
3. Enter the header record. In the first position of the first record, TABLE automatically places the letter T, which is displayed on the screen; the letter need not be coded or entered. If the letter is removed accidentally, the compilation will fail.
4. Enter the table elements.
5. To close the batch, enter END in the procedure field; this causes the system to select level Z automatically. End must begin in the first position and be the only statement in that field.
6. To compile the batch, use the Supervisor command COMPILE.

**Adding New
Procedures to
a Closed Batch**

After closing a batch by entering END in the procedure field, more table elements can be added only by following these steps:

1. Open the batch in the Search/Modify Mode and search for the END statement. Delete the statement.
2. Output the batch to tape using the WRITE command.

3. Use the Supervisor command INPUT to input the batch.
4. Open the batch in the ENTER mode and respond YES to the "CONTINUE=" prompt. Enter additional table elements.

C.3.17. TRANSLAT

Application To create translation tables that are used for data output. The records are written to tape and loaded as part of the system during system generation.

Program Description The program has six levels:

- H** Identifies the translation table being created.
- 1, 2, 3 and 4** Display the entire character set, 32 characters at a time, and provides for entered input.
- Z** Closes the batch.

The batch contains four 64-character output records when completed.

Operating Procedures

1. Open the batch in the Enter Mode using the TRANSLAT format.
2. Enter name (NAME) and version (VERSION) both user-defined; Release Number (REL-NO) and Identification Number, ID (the table number entered on the checkbox coding form). The name must be different from that of any system modules. ID is the number by which the table is requested; numbers available are 5 through 19 and 23 through 25.
3. When the character set is displayed, enter the appropriate hexadecimal equivalent for each character.
4. After all characters have been entered, the batch automatically closes.

5. Output the batch to tape using the WRITE command.
6. The write tape is then read into the system at the end of the system generation process.

Comments

The identification number of a translation table can be specified in checkbox 7 of the source input coding form. If not overridden by the selection of a new translation table option, the table specified on the coding form will be accessed during the WRITE function.

C.3.18. VFU-FORM

Application

To create the record required to load the electronic vertical forms unit (VFU) on a line printer. The printer uses electronic VFU instead of a punched paper tape control unit.

Program

Description

The program has two levels:

- I** Defines the VFU format.
- Z** Closes the batch.

**Operating
Procedures,
System Printers**

1. Open the batch in the Enter Mode using the VFU-FORM format.
2. Enter the total number of lines to be used on the printer form. VFU-FORM automatically places the letter S in the fourth position of the record and displays the character on the screen.
3. Enter one character into the record for each print line on the form. Enter an asterisk (*) if the line is not a VFU channel; enter 1 to 9, A, B, or C for those lines that define each VFU channel. (There are 12 VFU channels.)
4. Enter an E after defining the last line.

5. The program checks that the number of lines defined agrees with the number of lines specified in step 2. If the two numbers do not agree, an error and a field backspace occur. If the numbers agree, the batch automatically closed.
6. Load the VFU by printing the batch using the supervisor PRINT command and specifying translation table 22.

**Operating
Procedures,
Station Printers**

To load the station printer VFU, a sequential format program must READ the record from the disk batch and WRITE the record to the station printer output file using translation table 22.

Comments

To use the PRINT function, first load VFU control to enable the automatic skip-to-channel-1 function.

C.4. SUBPROGRAMS

A subprogram is used when a specific COBOL procedure might be executed more than once in a single format; i.e., by more than one field, or when the procedure is used by several formats. A format program accesses the subprogram through a CALL statement. The subprogram must be referenced by its exact name and must be on disk when the format is selected; otherwise, a run time error occurs and the message "FORMAT NOT FOUND" is displayed.

The appropriate **System Supervisor's Reference Manual** contains detailed information about writing subprograms and about calling subprograms from a format program.

For example, in the User's Program Library we include a routine for converting the calendar date into the Julian date. The main format program would include a statement that begins as follows:

CALL JULIAN USING

The main format program CALLs the JULIAN subprogram to convert a month-day-year calendar date into a Julian date. Under a data entry program the operator might enter October 27, 1980, as the date an item was purchased; the Julian subprogram converts that date to

80301. If the operator also needs to enter the date on which an invoice was prepared, then the format program must include a second statement calling the Julian subprogram.

This section includes the following subprograms:

Name	Purpose
CALENDAR	Converts Julian date to calendar date.
CHKDG-07	Modulo 7 check digit
CHKDG-10	Modulo 10 check digit
CHKDG-11	Modulo 11 check digit
DEC2HEX	Converts decimal numbers to hexadecimal.
EDIT	Sends edit data output to line printer.
HEX-CONV	Converts hexadecimal numbers to decimal.
ICL-NEG	Converts negative numbers to ICL format.
JULIAN	Converts calendar date to Julian date.
MEM11S	Generates memory analysis information for use by format program MEM11.

The following sections describe the subprograms.

C.4.1. CALENDAR

Application To convert a Julian date into a calendar date after a call from a format program.

Program Description The subprogram takes a 5-digit Julian date and returns the calendar month, day, and year. The calendar subprogram also signals an error if an invalid date is used.

Arguments Needed for Sending and Receiving

Julian Date.
Information can be held in a field, in the special register DATE, or in an arithmetic register.

Month, Day and Year.

Information for each can be held in a nonkeyed field or in an arithmetic or character register.

**Example of
CALL
Statement**

CALL CALENDAR USING DATE, THIS FIELD, A1, A2

Where:

DATE

is the special register containing current Julian date.

THIS FIELD

is the name of the field in which the month will be stored.

A1 and A2

are the arithmetic registers where the day and year, respectively, will be stored.

Comments

If a DECLARE REGISTER SIZE statement is used in the main format program, the size specified must be five or greater, or a SUBSCRIPT OVERFLOW error will occur.

C.4.2. CHKDG-07

Application

To generate or validate a modulo 7 check digit after a call from a format program.

**Program
Description**

With the number of digits in the base number and the base number itself as input, the subprogram returns a modulo 7 check digit.

**Arguments Needed
for Sending and
Receiving**

Number of Digits in the Base Number.

The information will be in the form of a numeric literal or may be contained in a field or arithmetic register.

Base Number.

The number can usually be in a keyed field, but it can also be contained in an arithmetic register.

Check Digit.

The check digit can be returned in an arithmetic register if a validation is being performed or in a nonkeyed field if the check digit is being generated.

**Example of
CALL
Statement**

CALL CHKDG-07 USING 8, BASE, A1

Where:

8

is the number of digits in the base number.

BASE

is a keyed field containing the base number.

A1

is an arithmetic register where the calculated check digit will be stored.

Comments

The base number cannot be larger than 35 184 372 088 831, the highest number possible on the system.

A modulo 7 check digit is computed by dividing the base number by 7. The remainder, between 0 and 6, is the check digit and is appended to the base number. For example, if the base is 53 736 483 and it is divided by 7, the quotient is 7676640 and the remainder 3 is the check digit. The entire number is 537364833.

The check digit can be validated in the main format program by using an IF statement.

Example:

```
IF A1 NOT = CHKDG ALARM "INVALID ACCT",  
FIELD BKSP
```

C.4.3. CHKDG-10

Application

To generate or validate a modulo 10 check digit after a call from a format program.

Program Description

With the number of digits in the base number and the base number itself as input, the subprogram returns a modulo 10 check digit.

Arguments Needed for Sending and Receiving

Number of Digits in the Base Number.

The information will be in the form of a numeric literal or may be contained in a field or arithmetic register.

Base Number.

The number can usually be in a keyed field, but it can also be contained in an arithmetic register.

Check Digit.

The check digit can be returned in an arithmetic register, usually the case if a validation is being performed, or in a nonkeyed field if the check digit is being generated.

Example of CALL Statement

```
CALL CHKDG=10 USING 8, BASE A1
```

Where:

8

is the number of digits in the base number

BASE

is a keyed field containing the base number

Al

is an arithmetic register where the calculated check digit will be stored

Comments

A modulo 10 check digit is computed by the following method:

1. Beginning with the right digit in the base number, select out those digits that are in the odd-numbered positions and multiply the series of digits by 2.

Example:

Base number 53736493; $3343 \times 2 = 6686$

2. The individual digits of the above product are added: $6 + 6 + 8 + 6 = 26$.
3. Beginning with the right digit, select the digits in the even-numbered positions in the base number and add the individual digits together.

Example:

**Base number 53736493;
 $5 + 7 + 6 + 9 = 27$**

4. Add the sums from steps two and three:
 $26 + 27 = 53$.
5. Subtract the final sum from the next highest number ending in 0. The result is the check digit.

$60 - 53 = 7$

6. The check digit is appended to the base number, and the entire number in the example would then be 537364937.

The base number cannot be larger than 35 184 372 088 831, the highest number possible on the system.

The check digit can be validated in the main format program by using an IF statement.

Example:

```
IF A1 NOT = CHKDG ALARM 'INVALID ACCT',  
FIELD BKSP
```

C.4.4. CHKDG-11

Application

To generate or validate a modulo 11 check digit after a call from a format program.

Program Description

With the number of digits in the base number and the base number itself as input, the subprogram returns a modulo 11 check digit.

Arguments Needed for Sending and Receiving

Number of Digits in the Base Number.

The information will be in the form of a numeric literal or may be contained in a field or arithmetic register.

Base Number.

The number can usually be in a keyed field, but it can also be contained in an arithmetic register.

Check Digit.

The check digit can be returned in an arithmetic register, which is usually the case if a validation is being performed, or in a nonkeyed field if the check digit is being generated.

Example of CALL Statement

```
CALL CHKDG-11 USING 8, BASE, A1
```

Where:

8

is the number of digits in the base number.

BASE

is a keyed field containing the base number.

AI

is an arithmetic register where the calculated check digit will be stored.

Comments

The base number cannot be larger than 35 184 372 088 831, the highest number possible on the system.

A modulo 11 check digit is computed by the following method:

1. Start at the least significant digit of the base number and work to the left. (The least significant digit is the right-most digit.)

Multiply the least significant digit in the base number by 2. Multiply the next digit to the left by 3; the third digit to the left by 4; and so on, until there are no more digits or a multiplier of 7 is reached.

If a multiplier of 7 is reached and there are still more digits in the number, the multiplier is again set to 2, and the process continues as before.

Example:

Base number = 5 3 7 3 6 4 8 3

(the least significant digit is 3)

$$3 \times 2 = 6$$

$$8 \times 3 = 24$$

$$4 \times 4 = 16$$

$$6 \times 5 = 30$$

$$3 \times 6 = 18$$

$$7 \times 7 = 49$$

$$3 \times 2 = 6$$

$$5 \times 3 = 15$$

- 2. Add all the products computed in step 1.
Divide the sum of the products by 11:

$$6 + 24 + 30 + 18 + 49 + 15 = 164$$

$$164 - 11 = 14, \text{ remainder} = 10$$

- 3. Subtract the remainder from 11:

$$11 - 10 = 1, \text{ the check digit}$$

- 4. The check digit is then appended to the base number as follows:

5 3 7 3 6 4 8 3 1

↑
check digit

If the modulo 11 routine applied to your base number leads to a check digit of:

- 11** the check digit used is 0 (zero)
- 10** the check digit is returned as -1, an **invalid** check digit
- n** (single digit) the check digit is that digit

The check digit can be validated in the main format program by using an IF statement.

Example:

```
IF A1 NOT = CHKDG ALARM 'INVALID ACCT',
FIELD BKSP
```

C.4.5. DEC2HEX

Application

To convert a decimal number to a hexadecimal number (maximum of four digits) when called by a format program.

Program Description

With the decimal number as input, DEC2HEX will return a hexadecimal number.

**Arguments Needed
for Sending and
Receiving**

Decimal Number.

This decimal number may be a numeric literal or in a field, an A-register, or a C-register.

Hexadecimal Number.

This hexadecimal value can be returned in a nonkeyed field or a C-register.

**Example of CALL
Statement**

CALL DEC2HEX USING A10, THIS FIELD.

Where:

A10

is an A-register containing the decimal number

THIS FIELD

is the nonkeyed field in which the converted hexadecimal value will be stored

Comments

The maximum value of the decimal number that this subprogram can convert is 65535 since the hexadecimal number returned is limited to four digits.

C.4.6. EDIT

Application

To edit numeric data being output to a line printer when called by a format program.

**Program
Description**

Data is stored on disk with no punctuation such as dollar signs or decimal points. To insert punctuation into a data field for a printed report, use the EDIT subprogram. The EDIT subprogram makes it possible to create an edit mask. An edit mask is a frame to hold punctuation in specified positions. Data is moved into the frame around the punctuation, and the frame and data are then printed on paper.

The compiled procedure language version of EDIT in the program library has been replaced with an assembly language version that provides a 10 to 1 increase in editing speed. The new version does not require any programming changes in the main format program. Minor operating differences are described in the following paragraphs.

Because the latest version of EDIT is now provided in assembly language, there is no source listing for it. But to provide backup, the original source of the procedure language version in the program library is maintained as a batch that can be recompiled to replace the assembly language version. The procedure language version is identified as:

Job Name: EDIT, Batch: 50000.

NOTE:

This subprogram yields unpredictable results when run on any earlier version. To avoid confusion, use the version of the User's Program Library tape released with a given system version.

**Arguments Needed
for Sending and
Receiving**

The data to be edited must be analyzed so the edit mask may be planned to achieve the desired printed output. The mask must be large enough to contain both the data and the punctuation. Include the following arguments in the COBOL procedure section of the source input form:

Data to be Edited.

Digits must be held in an arithmetic register.

Edit Mask.

The mask can be a literal, a character register, or a field.

Edited Data.

The edited data is usually stored in a nonkeyed field, but can also be held in a character register. The register or the field must be as large as or larger than the original edit mask.

If several data fields use the same edit mask, the mask should be stored in a C register. If a mask is to be used only once, it can be a literal in a procedure statement.

**Example of
CALL
Statement**

CALL EDIT USING A1, '\$__0.____' THIS FIELD

Where:

A1
is the data to be edited.

\$__0.____
the edit mask.

THIS FIELD
the name of the field where the edited data will be stored.

Example:

A1 (Data)	0 2 5 6 4 9
Mask	\$__0.____
Result	\$ 256.49

Comments

The following editing characters are valid:

— An underscore indicates a position to be replaced by data or a fill character.

- 0 A zero indicates zero suppression. It is placed to the immediate right of the position where zero suppression is to begin.
- * An asterisk indicates that asterisks will be used as fill. It is placed in the right-most position where asterisk fill is to begin.
- \$ A dollar sign placed in the left-most position stays fixed. In that position, it does not function as fill.
- If a dollar sign is placed in any other position, it becomes a floating dollar sign and is a zero suppression character. In this case it is placed to the immediate left of the first significant digit and in the right-most position where zero suppression is to begin.
- ,
.
/
- Commas, decimal points, blanks, slashes and dashes are placed in the exact positions where they are to occur in the edited data.
- CR
or
- The characters CR or a minus sign placed in the last positions of the mask will remain if the number is negative, or be replaced with a blank if the number is positive. A zero amount is always positive.

NOTE:

The zero, asterisk and floating dollar sign are mutually exclusive characters. If two or more are placed in the same mask, the character farthest to the right takes precedence.

**Differences
Between EDIT
Versions**

The data, mask, and area for edited data to be returned must be compatible. If the mask size is smaller than the data size in the procedure language version, either a "SUBSCRIPT OVERFLOW" error will occur or the edited data will be incorrect.

If the mask size is smaller than the data size in the assembly language version, the edited data will be truncated on the left; that is, the most significant digits of the data will be dropped, resulting in incorrect data.

If the edited data is to be returned in a character register, the register size must be at least one position larger than the mask. If the register size is smaller in the procedure language version a "SUBSCRIPT OVERFLOW" error will occur.

If the register size is smaller than the mask in the assembly language version, the edited data will be truncated on the left, resulting in incorrect data.

The main format determines character register size. The EDIT subprogram shares the register storage area with the main format and uses that area to perform editing. Therefore, the main format must specify a character register size large enough to support the largest mask value to be used during editing.

To specify register size, include a statement DECLARE REGISTER SIZE in the COBOL procedure section to the size of the character register plus one; or default to the size of the largest field or literal.

**EDIT Mask
Examples**

MASK	+ DATA	+RESULT	- RESULT
___, __. __	00001250	000,012.50	000,012.50
\$ __, __ 0. __	00001250	\$ 12.50	\$ 12.50
___, __ * . __	00001250	*****12.50	*****12.50
___, __ \$. __	00001250	\$12.50	\$12.50
\$ __, __ 0. __ CR	00001250	\$ 12.50	\$ 12.50CR
___, __ \$. __ -	00001250	\$12.50	\$12.50-
\$ __, __ 0. __ CR	00000000	\$.00	\$.00
\$ __, __ * . __ -	00000000	\$*****.00	\$*****.00
___, __ \$. __	00000000	\$.00	\$.00
__ / __ / __	060375	06/03/75	06/03/75
__ - __ - __	060375	06-03-75	06-03-75

C.4.7. HEX-CONV

Application

To convert a hexadecimal number to a decimal when called by a format program.

Program Description

With the hexadecimal number and number of digits in the number as input, HEX-CONV will return a decimal number.

Arguments Needed for Sending and Receiving

Hexadecimal Number.

The number must be contained in a character register.

Number of Digits in Hexadecimal Number.

The number will usually be a numeric literal, but can also be contained in a field or arithmetic register.

Decimal Number.

The number can be returned in a nonkeyed field or arithmetic register.

**Example of
CALL
Statement**

CALL HEX-CONV USING HEX-NUM, 8, THIS FIELD

Where:

HEX-NUM

is a field containing a hexadecimal number.

8

is the number of digits in the hexadecimal number.

THIS FIELD

is the field in which the converted decimal number will be stored.

Comments

The maximum size of the hexadecimal number that the program can convert is 8 digits. An error will occur if a larger number is used.

C.4.8. ICL-NEG

Application

To convert negative numbers to ICL 1900 format when called by a format program.

**Program
Description**

The subprogram takes a negative number stored in the system format (sign information over the least significant digit position) and returns the number in the ICL 1900 format (sign information in the most significant digit position).

**Arguments Needed
for Sending and
Receiving**

Negative Number to be Converted.
The number must be in a field.

Number of Digits in the Negative Number.

The number will usually be a numeric literal, but can also be a field or arithmetic register.

The Converted Number.

The number will usually be a nonkeyed field, but can also be a character register.

**Example of
CALL
Statement**

**IF AMOUNT NOT >0 CALL ICL-NEG USING AMOUNT, 8,
THIS FIELD ELSE MOVE AMOUNT TO THIS FIELD**

Where:

AMOUNT

is a number that can be negative.

8

is the size of the AMOUNT field.

THIS FIELD

is where the converted number will be stored.

Comments

To determine whether the original number is negative, include an IF statement in the format program; see CALL statement example (IF AMOUNT NOT >0.....).

The size of the receiving area must be the same as that of the sending field.

If the original number has one or more leading zeros, the first position will be converted to a minus sign.

C.4.9. JULIAN

Application

To convert any calendar date to the Julian date when called by a format program.

**Program
Description**

The subprogram takes month, day, and year and returns a five-digit Julian date. An error is signaled if an invalid date is used.

**Arguments Needed
for Sending and
Receiving**

Month.

The month will usually be held in a field, but can also be held in a register.

Day.

The day will usually be held in a field, but can also be held in a register.

Year.

The year will usually be held in a field, but can also be held in a register.

Julian Date.

This date can be held in a nonkeyed field or register.

Comments

If the month, day, and year are keyed in one field, it is necessary to transfer each of the components of the date to a separate character register before calling the subprogram JULIAN. In the following example, the date entered is November 20, 1980, entered as 112080.

Example:

```
CONNECT TDATE(1), TDATE(2) GIVING C1
CONNECT TDATE(3), TDATE(4) GIVING C2
CONNECT TDATE(5), TDATE(6) GIVING C3
CALL JULIAN USING C1, C2, C3, THIS FIELD
```

Where:

TDATE (1) and TDATE (2)
is 11 (month)

TDATE (3) and TDATE (4)
is 20 (day)

TDATE (5) and TDATE (6)
is 80 (year)

C1-C3
are character registers.

THIS FIELD
is where the converted date will be stored.

C.4.10. MEM11S

Application

Generates memory analysis information in A-registers 1 through 28 intended for use by the format program MEM11.

Program Description

MEM11S is a machine language subprogram containing COBOL code only for ENTRY, the argument list, EXIT TO ML, and GO BACK. The machine language portion of the subprogram was generated from the module MFM TMT.

Arguments Needed for Sending and Receiving

A-register 33 is used as a dummy parameter in calling MEM11S.

Information is passed back to the calling program in A-registers 1 through 28 as follows:

A1 SM BUFFER SPACE

All standard buffer pools
Tenant Table
Alternate Disk MCBS and RCBS
SM Buffer tenants
DCOM Fixed Buffer
SYS TEST PKG Buffer
SCL

A2 SM TWAs

Foreground and background SM TWAs (excluding 1-word TWAs in Fixed Area MSG HANDLER TWA, and TWA used for running MEM11R1)

A3 FIXED EM

Gap Tenants

A4 FIXED SM

Low Core
Resident
Fixed Buffer Overlay
400 Bytes for SCL Growth
MSG HANDLER TWA
One Word for each TWA up to TENANT Table
System Disk MCBS and RCB
PHY IO and MTS Buffers
CRT Update Table

A5 SM OVERLAYS

Used overlays in SM Tenants

A6 SM FORMATS

Used formats in SM Tenants (excluding
MEM11R1 and MEM11R1S)

A7 TOTAL USED SM

Total of A1, A2, A4, A5, A6 above

A8 TOTAL UNUSED SM

Total of unused SM Tenants and free space
around SCL--200 words for stack growth

A9 FIXED SM MEM TMT

SM space used for MEM11, MEM11S, or the TWA
used to run MEM11

A10 TMT TWA GROWTH

On EM systems, the amount which the largest
TMT TWA may grow assuming no growth
elsewhere

A11 CMT TWA GROWTH

On EM systems, the amount which the CMT TWA
area may grow assuming no growth elsewhere

A12 SM TENANTS GROWTH

On EM systems, the amount which the SM TENANTS area may grow assuming no growth elsewhere

A13 EM BUFFER SPACE

Used EM Buffer tenants

A14 EM TWAS

Used foreground and background EM TWAs (excluding the TWA used for running MEM11)

A15 FIXED EM MEM TMT

EM space used for MEM11, MEM11S, and the TWA used to run MEM11

A16 EM OVERLAYS

Used overlays in EM

A17 EM FORMATS

Used formats in EM tenants (excluding MEM11 and MEM11R1S)

A18 TOTAL USED EM

Total of A3, A13, A14, A16, and A17 above

A19 TOTAL UNUSED EM

Total of unused tenants in EM

A20 RME REGS

Number of registers required for RME resident mapping

A21 TME RES AND SM TENANTS REGS

Number of registers required for TME resident and SM Tenant mapping

A22 TME TMT TWA REGS

Number of registers required for TME maximum TMT TWA mapping

A23 TME CMT TWA REGS

Number of registers required for TME maximum CMT TWA mapping

A24 TME OVLY REGS

Number of registers required for the overlay mapping

A25 TME TEMP REGS

Number of registers required for the TEMP mapping (XMAP)

A26 MAX TMT TWA NO

Number of the maximum sized TMT TWA

A27 EM SYSTEM FLAG

Flag to indicate EM system

A28 ERROR FLAG

Flag to indicate errors: 0 on no errors; may be 2 if a tenant length, TTI, or type error occurs

Note that A10-A12 (growths) and A20-A27 (TME, RME information) are 0 on SM only systems. Also, A10-A12 (growths) on EM systems assumes that the SM free and unused tenants will be squeezed out, but does not assume that EM qualified tenants or TWAs will go to EM.

Example of CALL Statement

CALL MEM11S USING A33

Where:

A33
is used as a dummy parameter.

Comments

MEM11S is a special machine language subprogram which puts special restrictions on its caller. These restrictions are adhered to by MEM11 and must be followed by any other calling program.

1. The caller must reserve the first 28 A-registers for use by MEM11S.
2. A-register 28 should be set up to some value other than 0 or 2 so that the return status in A28 can be interpreted correctly.
3. The caller must not make use of any other subprograms or tables in order to obtain accurate memory evaluation for the task using MEM11S.

C.5. TABLE PROGRAMS

A table is a series of items arranged in a logical order. The information consists of constant data that can be used for data validation or data generation. A table program is written on a source input coding form, entered under control of the TABLE format program from the program library, and then compiled for storage on the system. A format program accesses a table through a FIND or IF statement. For example, a format program may access a table to validate a charge number or to automatically retrieve a customer's name and insert it in a field, based on an entered account number

The **System Supervisor's Reference Manual** contains detailed information about writing table programs and about accessing a table from a format program.

This section includes the following table programs:

Name	Purpose
DEV-CODE	Convert EBCDIC characters to data communication device control codes
FISTATUS	Translate file inquiry status codes
FMSTATUS	Translate file management status codes
HEX2DEC	Convert hexadecimal to decimal
HW-TAB	Convert trailer label data block count to binary for use by HW-6000SL label program
ICL-TAB	Convert ICL labels before translation
NTBL	Used by DATATYPE format program
ST-TABLE	2-digit numeric code representing state names
STATE-TB	2-character alphabetic code representing state names
TESTTAB	Select EBCDIC characters for host interactivity screen formatting
TYPTBL	Used by DATATYPE format program
YNTBL	Used by DATATYPE format program
YTBL	Used by DATATYPE format program

The following sections describe the table programs.

C.5.1. DEV-CODE

Application To supply data communication device control codes. Because the codes have no EBCDIC representation, they cannot be produced by normal means within the system. The control codes are used to transmit data to remote sites. Upon reception, the control codes can be acted upon as if they were data communication device control codes from a mainframe.

**Table
Description**

DEV-CODE is a double entry lookup table. When given a single byte EBCDIC character argument, the table returns a hexadecimal number internally that serves as a data communication device control code. A summary of the codes is included in Table C-1.

**Example of
Use**

When preparing a batch for transmission, enter the following statement:

FIND U IN DEV-CODE GIVING THIS FIELD

This returns a hexadecimal 27, the data communication escape code. The code is then available in the disk record. When transmitted, the code functions like any escape code, signaling that the character following the code is a carriage control character, not a graphic character.

Comments

DEV-CODE is supplied in compiled form on the User's Program Library tape. There is no source code on the tape; the table cannot be compiled by users in the field.

Table C-1.
Data Communication Device Control Codes

Symbolic Input	Output Byte	Type	Abbreviation
A	00	Null	NUL
B	04	Punch Off	PF
C	05	Horizontal Tab	HT
D	07	Delete	DEL
E	0B	Vertical Tab	VT
F	0C	Form Feed	FF
G	0D	Carriage Return	CR
H	11	Device Control 1	DC1
I	12	Device Control 2	DC2
J	13	Device Control 3	DC3
K	14	Restore	RES
L	15	New Line	NL
M	19	End of Medium	EM
N	1B	ASCII Escape Code	CU1
O	1C	Interfield Separator	IFS
P	1D	Intergroup Separator	IGS
Q	1E	Interrecord Separator	IRS
R	1F	Interunit Separator	IUS
S	22	Field Separator	FS
T	25	Line Feed	LF
U	27	Escape	ESC
V	2A	Set Mode	SM
W	2F	Bell	BEL
X	34	Punch On	PN
Y	3C	Device Control 4	DC4
Z	5D	Device Control 4	DC4

C.5.2. FISTATUS

Application

To translate the file status codes that reflect the condition of a file and of the functions being performed on the file in accessing indexed files.

Table

Description

In this double entry table, each file status code is listed with an explanatory translation.

**Example of
Use**

When you access an indexed file using file inquiry verbs, a 2-digit code is returned indicating file status. Codes greater than 02 indicate an error or difficulty. To receive an explanation for the code displayed, enter:

FIND STATUS IN FISTATUS GIVING THIS FIELD

A message will be returned explaining the significance of the particular file status code.

File status codes are also explained in the **System Supervisor's Reference Manual** and in F.8. of this document.

C.5.3. FMSTATUS

Application

To translate the file status codes that reflect the condition of a file and of the functions being performed on the file in accessing indexed or sequential files.

**Table
Description**

When accessing an indexed or sequential file using file management, a 2-digit code is returned indicating file status. Codes greater than 02 indicate an error or difficulty. To receive an explanation for the code displayed, enter:

FIND STATUS IN FMSTATUS GIVING THIS FIELD

A message will be returned explaining the significance of the particular file status code.

File status codes are also explained in the **System Supervisor's Reference Manual** and in F.8. of this document.

C.5.4. HEX2DEC

Application

To convert hexadecimal digits to decimal equivalents required by the DATATYPE format program.

**Table
Description**

This double entry conversion table lists single-digit hexadecimal numbers and the decimal equivalent for each.

**Example of
Use**

FIND HEX-FLD IN HEX2DEC GIVING A1

Where:

HEX-FLD

is a 1-position hexadecimal field.

A1

is an arithmetic register.

C.5.5. HW-TAB

Application

To convert the trailer label data block count to a binary value for use by HW6000SL.

**Table
Description**

HW-TAB is a double entry conversion table.

**Example of
Use**

A modulo 64 division routine executed within the HW6000SL program yields remainders from 0 to 63. These are the input to the conversion table. The result of the conversion will be characters that, when written to tape using the specified translation table, will result in the appropriate binary representation.

C.5.6. ICL-TAB

Application

To convert the label data block count to a binary value for use by ICL-LBL.

**Table
Description**

ICL-TAB is a double entry conversion table.

**Example of
Use**

A modulo 64 division routine executed within the ICL-LBL program yields remainders from 0 to 63. These are the input to the conversion table. ICL-LBL converts these numbers to characters within the ICL character set. When the characters are written to magnetic tape using an ICL output translation table, they are converted to the appropriate binary representation.

C.5.7. NTBL

Application

To reset bits to NO for use by the DATATYPE format program.

**Table
Description**

Given a positive integer variable less than 65, NTBL returns a new value.

**Example of
Use**

The DATATYPE format program allows you to modify the attributes of a given data type. For a specified character in a data type, NTBL returns a code that causes the character to be omitted from the character subset for that data type.

C.5.8. ST-TABLE

Application

To validate state abbreviation codes.

**Table
Description**

ST-TABLE contains 55 two-digit numeric codes.

**Example of
Use**

**IF THIS FIELD NOT IN ST-TABLE ALARM, INVALID
STATE CODE**

Where:

THIS FIELD
is a keyed field.

C.5.9. STATE-TB

Application To validate the state abbreviation in name and address fields.

Table Description STATE-TB contains 54 two-character alphabetic state abbreviation codes.

Example of Use **IF THIS FIELD NOT IN STATE-TB ALARM, INVALID STATE KEYED, FIELD BKSP**

Where:

THIS FIELD
is a keyed field.

C.5.10. TESTTAB

Application To select EBCDIC characters for host inter-activity screen formatting.

Table Description This single entry table uses a decimal value (0-256) passed by the main format program to obtain an EBCDIC character used as a special screen formatting character of the IBM 3270 protocol.

TESTTAB is supplied in compiled form on the User's Program Library tape. There is no source code on the tape; the table cannot be compiled by users in the field.

Example of Use **FIND VALUE IN TEST TAB GIVING CHARACTER**

Where:

VALUE
is any decimal number in the range 0-256.

CHARACTER
is the associated IBM 3270 special screen formatting character.

Consult the IBM 3270 Component Description manual for a complete list of special characters and their associated numeric table values.

C.5.11. TYPTBL

Application To select data type for use by the DATATYPE format program.

Table Description This double entry table returns a subscript to be used in obtaining the valid characters for system data types.

Example of Use
**FIND TYPE IN TYPTBL, GIVING A4 ELSE ALARM
INVALID, FIELD BKSP**

Where:

TYPE
is a 1-character keyed field.

A4
is a register used to obtain the valid characters for the keyed data types.

C.5.12. YNTBL

Application To define the default system subset associated with each data type used by the DATATYPE format program.

Table Description YNTBL contains positive integer variables from 01 to 62, returning either a Y or N for each.

Example of Use
This table is used by the DATATYPE format program to return a Y or N for any character from the subset for a given data type.

C.5.13. YTBL

Application To set bits on YES for use by DATATYPE format program.

Table Description YTBL contains positive 2-digit integer variables.

Example of Use The DATATYPE format program makes it possible to modify the attributes of a given data type. For a specific character in a data type, YTBL returns a code that causes the character to be included in the character subset for that data type.

C.6. LABEL PROGRAMS

A label program processes labeled magnetic tape or flexible diskettes. Some of the functions of label programs are:

- To check tape labels to ensure that the correct volume is mounted.

- To create new labels.

- To position the tape beyond existing labels.

- To process labels during volume switching.

- To examine and modify diskette labels.

A label program is entered under control of the LABEL format program. It can be specified in the format header or through the supervisor WRITE command. The **System Supervisor's Reference Manual** contains detailed information about using label programs.

The following is a listing of the label programs.

Name	Purpose
HW6000SL	Create/verify Honeywell 6000 standard labels
IBMCR	Create IBM OS standard labels
IBMCRV	Create IBM Variable Length Record format IBM OS standard tape labels
IBMDL	Create and process default IBM OS standard labels
IBMDLV	Create and process default IBM Variable Length Record format IBM OS standard tape labels
IBMFD	Examine and modify IBM standard diskette labels
IBMNL	Process the IBM OS no label option
IBMSL	Process IBM OS standard labels
IBMSLV	Process IBM Variable Length Record format IBM OS standard tape labels
ICL-LBL	Create and process ICL standard labels
ICL29CR	Create ICL 2970 format tape labels
ICL29SL	Input/verify ICL format tape labels

The following sections describe the label programs.

C.6.1. HW6000SL

Application To create (output) and verify (input) Honeywell 6000 standard format magnetic tape labels.

Necessary Table HW-TAB

Program Description The program has four levels:

- 0 Selects level 1 if the tape is being opened for input; selects level A if it is being opened for output; and selects level B if the output tape is being closed.
- 1 Reads and validates the header label plus the tape mark preceding the data blocks; the tape is moved so that the beginning of the first data block is at the read-write head.
- A Writes the header label plus one tape mark and returns to WRITE so data blocks may be written.
- B Writes one tape mark followed by the trailer label and two more tape marks, and then terminates label operations; the program inserts the block count into the trailer label. A modulo 64 routine is used to convert the data block count to binary. The HW-TAB conversion table is used in the process.

**Operating
Procedures**

1. Specify HW6000SL in the data format program. The HW6000SL label program will then be called when the system is writing to or inputting from tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE or INPUT command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=HW6000SL

3. Specify HW6000 tape format by responding as follows:

OPTIONS=PK-PACKING
PACKING=H6000
**BLOCK FACTOR=(specify if different
from default**

4. Specify HW6000 tape format by responding as follows:

OPTIONS=TRANS

TRANSLATION=(respond with the user
tape translation table
number of the Honeywell
6000 character set)

5. If performing a write operation, enter the installation ID, tape reel serial number, file serial number, retention days, and file name in response to display messages for the header label.

C.6.2. IBMCR

Application

To create IBM OS standard labels on output tapes, when specified in the format program or by the supervisor as an option in the WRITE function.

Program Description

The program has 10 levels. The first level tests that the batch is OPEN in OUTPUT. If not, the error message "IBMCR INVALID FOR OPEN INPUT" is displayed. Remaining levels create the volume one record, header, and trailer labels.

Operating Procedures

1. Specify IBMCR in the format program. The IBMCR label program will then be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL

LABELS=YES

LABEL=IBMCR

3. When IBMCR is called, enter the following information according to the indicated field descriptions to create the volume one and header records:

Field	Size	Shift	Data	Comment
Volume Number	6	N	U	
Owner	10	A	U	
Data Set Name	17	A	U	
Expiration	6	N	N	Format YYDDD (Date) or 0
Security	1	N	N	0 = No; 1 = Yes

Comments

Processing occurs in this sequence: IBMCR is given control to create the header label. Control then passes to the format program to write the batches to tape. Finally, control returns to IBMCR to generate the trailer label.

C.6.3. IMBCRV**Application**

To create IBM labels on IBM Variable Length Record (VLR) format output tapes, when specified in the format program or by the supervisor as an option in the WRITE function.

**Program
Description**

The program has 10 levels. The first level tests that the batch is OPEN in OUTPUT. If not, the error message "IBMCR INVALID FOR OPEN INPUT" is displayed. Remaining levels create the volume one record, and header and trailer labels.

**Operating
Procedures**

1. Specify IMBCRV in the format program. The IMBCRV label program will then be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMCRV

- Specify IBM VLR tape format by responding as shown:

OPTIONS=PK-PACKING

PACKING=VLR-VLB

**DATA BLK SIZE=(respond with desired
maximum block size)**

- When IBMCRV is called, enter the following information according to the indicated field descriptions to create the volume one and header records:

Field	Size	Shift	Data	Comment
Volume Number	6	N	U	
Owner	10	A	U	
Data Set Name	17	A	U	
Expiration	6	N	N	Format YYDDD (Date) or 0
Security	1	N	N	0 = No, 1 = Yes

Comments

Processing occurs in this sequence: IBMCRV is given control to create the header label. Control then passes to the format program to write the batches to tape. Finally, control returns to IBMCRV to generate the trailer label.

C.6.4. IBMDL

Application

To verify the presence of IBM standard labels on input tapes and to create IBM standard labels, with default values on output tapes.

**Program
Description**

The program has eight levels: Level 0 selects the proper level in the format for processing. Decisions for level selection are based on whether the tape is being input or output, and if output, whether the process is being opened or closed.

**Operating
Procedures**

For input:

- Specify IBMDL in the format program. The IBMDL label program will then be called when tape is input.

2. To override a label program specified in the format program, respond to these prompts as shown in the supervisor INPUT command sequence:

```
OPTIONS=LABEL  
LABELS=YES  
LABEL=IBMDL
```

3. IBMDL does not require input at the terminal. The format program calls IBMDL to verify the presence of IBM standard labels.

For output:

1. Specify IBMDL in the format program. The IBMDL label program will be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the supervisor WRITE command sequence:

```
OPTIONS=LABEL  
LABELS=YES  
LABEL=IBMDL
```

3. IBMDL does not require input at the terminal. IBMDL is given control to create the header label. Control then passes to the format program to write the batches to tape. Finally, control returns to IBMDL to generate the trailer label.

C.6.5. IBMDLV

Application

To verify the presence of IBM standard labels on IBM Variable Length Record (VLR) format input tapes and to create IBM standard labels with default values on IBM VLR format output tapes.

**Program
Description**

For input:

1. Specify IBMDLV in the format program. The IBMDLV label program will then be called when tape is input.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor INPUT command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMDLV

3. Specify IBM VLR tape format by responding as shown:

OPTIONS=PK-PACKING
PACKING=VLR-VLB
**DATA BLK SIZE=(respond with desired
maximum block size)**

4. IBMDLV does not require input at the terminal. The format program calls IBMDLV to verify the presence of IBM standard labels.

For output:

1. Specify IBMDLV in the format program. The IBMDLV label program will be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence

OPTIONS=LABEL
LABELS=YES
LABEL=IBMDLV

3. Specify IBM VLR tape format by responding as shown:

OPTIONS=PK-PACKING

PACKING=VLR-VLB

DATA BLK SIZE=(respond with desired
maximum block size)

4. IBMDLV does not require input at the terminal. IBMDLV is given control to create the header label. Control then passes to the format program to write the batches to tape. Finally, control returns to IBMDLV to generate the trailer label.

C.6.6. IBMFD

Application

To examine or modify basic data exchange (IBM compatible) diskette data set labels and volume labels; to delete diskette data sets and examine track error maps. This program cannot be used with extended format diskettes.

Program Description

The program has 12 levels. The first level selects which task is to be accomplished. The program is tutorial and uses extensive message line displays and data area prompting.

Operating Procedures

1. Initialize the diskette through the Supervisor command "DK UTILITY--INITIALIZE--BASIC".
2. SET the 01 OUTPUT device to the drive containing the basic diskette.
3. Select the FD UTILITY option from the Supervisor menu. Respond to the following prompts as shown:

FUNCTION=LABEL

LABEL=IBMFD

UNIT=(select diskette drive unit
number)

4. The first level prompts for a selector code, which then routes execution to the appropriate program level.
5. In the examine/modify levels:
 - a. To display the full label, press the FIELD RFL key at the first or any subsequent field.
 - b. To change the label, enter the correct data into the field that needs changing.
 - c. To accept current label contents, press the FIELD REL key.
 - d. To change a nonblank field to blank, enter the letter Z and press the FIELD REL key (this action is appropriate only in fields that are allowed to be blank).
6. To modify a label, enter M at the end of the examine/modify level. After modifying the label, the system displays the same label again for the change to be verified visually.

CAUTION:

Modifying a label destroys existing label information.

7. To examine/modify another label after finishing with one label:
 - a. Enter N to access the label in the next higher sector.
 - b. Enter P to access the label in the previous sector.
 - c. Enter L to return to the beginning of the current sector.
 - d. Enter R to return to the selector menu.

8. To delete a data set from a basic diskette, use the IBMFD label program, selecting the delete data set level. (The Supervisor DELETE command has no effect on basic diskette data.)
9. The diskette error map displays the track number of any bad tracks or blanks for a diskette without bad tracks.

Comments

Before you can examine or modify labels, the diskette must be initialized through the Supervisor command DK UTILITY--INITIALIZE--BASIC.

For detailed specifications on basic diskette labels, refer to the appropriate **System Supervisor's Reference Manual**.

C.6.7. IBMNL

Application

To create an output tape without a label from a tape that has a label.

Program Description

The program has three levels:

- 0** Selects level 1 only if the tape is being opened for output.
- 1** Validates the volume label.
- A** Validates the header label and rewinds the tape unit.

Operating Procedures

1. Specify IBMNL in the format program. The IBMNL label program will be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMNL

3. IBMNL does not require terminal input. The label program checks for IBM OS standard header label and expiration date. If the header label is standard and the date has expired, the tape is rewound and an unlabeled tape is written. If the label is nonstandard or the date has not expired, an error message is displayed, the task is canceled, and the output tape is not written.

Comments

The IBMNL label program is not intended to be used for INPUT functions. If so used, however, the label will be verified. If the date is expired, the label will be treated as a data file, and if the formats are compatible, the label will be input as data.

C.6.8. IBMSL

Application

To validate IBM standard labels.

For INPUT functions this program checks for IBM standard labels and positions the tape past the label.

For OUTPUT functions the program checks for IBM standard label format and expiration date. If the header labels are standard and the date has expired, the tape is rewound and a standard labeled tape is then written. If the label is nonstandard or the date has not expired, an error message is displayed, the task is cancelled, and the output tape is not written.

**Program
Description**

The program has 11 levels. Level 0 selects the proper level in the format for processing. Decisions for level selection are based on whether the tape is being input or output, and if output, whether the process is being opened or closed.

**Operating
Procedures**

For input:

1. Specify IBMSL in the format program. The IBMSL label program will then be called when tape is input.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor INPUT command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMSL

3. IBMSL does not require input at the terminal.

For output:

1. Specify IBMSL in the format program. The IBMSL label program will be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMSL

3. When IBMSL is called, enter the following parameters according to the indicated field descriptions:

Field	Size	Shift	Data	Comment
Owner	10	A	U	
Data Set Name	17	A	U	
Expiration	6	N	N	Format YYDDD (Date) or 0
Security	1	N	N	0 = No; 1 = Yes

Comments

No further entries are necessary. IBMSL is given control to create the header label. Control then passes to the format program to write the batches to tape. Finally, control returns to IBMSL to generate the trailer label.

C.6.9. IBMSLV

Application

To validate IBM standard labels on IBM Variable Length Record (VLR) format tapes.

For INPUT functions on IBM VLR format tapes this program checks for IBM standard labels and positions the tape past the label.

For OUTPUT functions on IBM VLR format tapes the program checks for IBM standard label format and expiration date. If the header labels are standard and the date has expired, the tape is rewound and a standard labeled tape is then written. If the label is nonstandard or the date has not expired, an error message is displayed, the task is canceled, and the output tape is not written.

**Program
Description**

The program has 11 levels. Level 0 selects the proper level in the format for processing. Decisions for level selection are based on whether the tape is being input or output and, if output, whether the process is being opened or closed.

**Operating
Procedures**

For input:

1. Specify IBMSLV in the format program. The IBMSLV label program will then be called when tape is input.
2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor INPUT command sequence:

**OPTIONS=LABEL
LABELS=YES
LABEL=IBMSLV**

3. Specify IBM VLR tape format by responding as shown:

OPTIONS=PK-PACKING
PACKING=VLR-VLB
DATA BLK SIZE=(respond with desired
maximum block size)

4. IBMSLV does not require input at the terminal.

For output:

1. Specify IBMSLV in the format program. The IBMSLV label program will be called when the system is writing to tape for the format program.
2. To override a label program specified in the format program, respond to these prompts as shown in the supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=IBMSLV

3. Specify IBM VLR tape format by responding as shown:

OPTIONS=PK-PACKING
PACKING=VLR-VLB
DATA BLK SIZE=(respond with desired
maximum block size)

4. When IBMSLV is called, enter the following parameters according to the indicated field descriptions:

Field	Size	Shift	Data	Comment
Volume Number	6	N	U	
Owner	10	A	U	
Data Set Name	17	A	U	
Expiration	6	N	N	Format YYDDDD (Date) or 0
Security	1	N	N	0 = No; 1 = Yes

Comments

No further entries are necessary. IBMSLV is given control to create the header label; control then passes to the format program to write the batches to tape; and finally control returns to IBMSLV to generate the trailer label.

C.6.10. ICL-LBL

Application

To create or validate standard International Computers Limited (ICL) 1900 series labels.

**Necessary
Table**

ICL-TAB

**Program
Description**

The program has five levels:

- 0 Selects level 1 if the tape is being opened for output or if the data is being output to a new reel (other than the first) of a multireel file. Selects level 2 if the output tape is being closed or the end of tape has been reached. Selects level 3 if the tape is being opened for input.
- 1 Prompts for the tape serial number and file name and computes the volume sequence number. This information is included in the header label written to tape.
- 2 Writes a tape mark and the start-of-data sentinel and returns to WRITE so data blocks may be written.

- 3 Writes one tape mark followed by an end-of-volume or trailer label. This level includes a modulo 64 routine to convert the data block count to binary and insert it into the end-of volume or trailer label; in the process, the ICL-TAB conversion table is used.
- 4 Reads and validates the header label and then reads the tape mark and start-of-data sentinels, leaving the tape positioned to input data blocks.

Operating Procedures

For input:

1. Specify ICL-LBL in the data format program. The ICL-LBL label program will then be called during tape INPUT.
2. To override a label program specified in the data format program, respond to these prompts as shown in the Supervisor INPUT command sequence:

```
OPTIONS=LABEL  
LABELS=YES  
LABEL=ICL-LBL
```

3. No further entries are required for INPUT. During execution the ICL-LBL program confirms that the first four characters of the tape header are HDDR. If not, an error message is displayed and INPUT is canceled.

For output:

1. Specify ICL-LBL in the format program. The ICL-LBL label program will then be called when the system is writing to tape for the format program.

2. To override a label program specified in the format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=ICL-LBL

3. When ICL-LBL is called, enter in the file name and tape serial number in response to display messages.
4. The ICL-LBL label program then creates the header labels, and the format program writes the batches to tape. If more than one tape reel is required to write the data, an end-of-volume label is placed at the end of the reel and the user is instructed to load another reel on the tape drive. Another header label is written (the user is again prompted for the tape serial number and file name), and the writing of data continues. Up to 512 reels may be used to write the file.

C.6.11. ICL29CR

Application

To create tape labels in the International Computers Limited (ICL) 2970 computer series magnetic tape format.

Program Description

The program has five levels:

- 0** Selects level 1 if the file header is to be written and level 2 if the file trailer is to be written.
- 1** Writes the volume header block and then selects level A.
- A** Writes the first file header block and then selects level B.

- B Writes the second file header block plus one tape mark and then returns to WRITE so the data block can be written.
- 2 Writes one tape mark plus the two end-of-file blocks plus two tape marks, and then terminates label operations.

**Operating
Procedures**

- 1. Specify ICL29CR in the data format program. The ICL29CR label program will then be called when the system is writing to tape for the format program.
- 2. To override a label program specified in the data format program, respond to these prompts as shown in the Supervisor WRITE command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=ICL29CR

- 3. Specify ICL 2970 tape format by responding as shown:

OPTIONS=PK-PACKING
PACKING=2970 ICL
**RECORD LTH=(respond with desired
record length)**
**BLOCK FACTOR=(respond with desired
blocking factor)**

- 4. Specify ICL 2970 character set translation by responding as follows:

OPTIONS=TRANS
**TRANSLATION=respond with user tape
translation table
number of the ICL 2970
character set**

NOTE:

Steps 3 and 4 are required procedures for ICL29CR label operations.

- 5. The ICL29CR program requires the following input at the terminal:

Prompt	Response
ENTER VOLUME IDENTIFIER	A 6-character (alphanumeric) volume name
ENTER FILE IDENTIFIER	A 17-character (alphanumeric) file name
ENTER GENERATION NUMBER	A 4-digit (numeric) generation sequence number
ENTER EXPIRATION DATE	A 5-digit (numeric) expiration date in the form of YYDDD

6. The label program generates and writes the volume header block, two file header blocks, one tape mark, and then returns control to WRITE.

WRITE writes the formatted data blocks. The label program generates and writes one tape mark, two end-of-file blocks, and two tape marks. WRITE then terminates.

Comments

This label program does not function with INPUT. ICL 2970 labeled tapes are read by the label program ICL29SL.

ICL 2970 labeled tapes are not compatible with ICL 4X3 packed tapes (ICL 1900 and 2903 systems).

C.6.12. ICL29SL

Application

To input/verify tape labels in the International Computers Limited (ICL) 2970 computer series magnetic tape format.

Program Description

The program has two levels:

- 0 Selects level 3 only if the tape is being opened for INPUT of volume one, file one.

- 3 Reads and validates the volume and file header tape blocks plus the tape mark preceding the data blocks. The tape is moved so that the beginning of the first data block is at the read-write head.

**Operating
Procedures**

1. Specify ICL29SL in the data format program. The ICL29SL label program will then be called during input from tape.
2. To override a label program specified in the data format program, respond to these prompts as shown in the Supervisor INPUT command sequence:

OPTIONS=LABEL
LABELS=YES
LABEL=ICL29SL

3. Specify ICL 2970 tape format by responding as shown:

OPTIONS=PK-PACKING
PACKING=2970 ICL
RECORD LTH=respond with desired
record length
BLOCK FACTOR=respond with desired
blocking factor

4. Specify ICL 2970 character set translation by responding as shown:

OPTIONS=TRANS
TRANSLATION=respond with user tape
translation table
number of ICL 2970
character set

NOTE:

Steps 3 and 4 are required procedures for ICL29SL label operations.

5. ICL29SL does not require input at the terminal. The label program checks for ICL 2970 standard volume and file header records. If the Supervisor function is not specified as INPUT, if volume or file header records are not found, or the tape mark preceding the data is not found, an error message is displayed and label processing is canceled.

Comments

This label program does not function with WRITE. ICL 2970 labeled tapes are written by the label program ICL29CR.

ICL 2970 labeled tapes are not compatible with ICL 4X3 packed tapes (ICL 1900 and 2903 systems).

D. Software Diagnostics

Software Diagnostics is an off-line utility for the SPERRY 1900/10 and XL/40 Distributed Processing and Capture Systems. It is used by operations people, programmer/analysts, and software development programmers.

Operations personnel normally use Software Diagnostics to initialize memory dump tapes (MAKEMD) and to Recover JOBS (Supervisor SAVE) from a disk platter with an unreadable Volume Table of Contents (VTOC). Programmer/analysts may use Software Diagnostics to look at the contents of the VTOC and/or data batches on disk, to copy magnetic tapes, to convert memory dump tapes to printed listings, or to separate a Field Release Master Tape. Software Diagnostics may also be used to compare memory and disk dumps resulting from an Operating System error to known Operating Systems and to recreate Operating System errors.

Software Diagnostics cannot be operated concurrently with the Operating System and is controlled by commands entered at one keystation only. It is not disk resident, however. Normal system operations and keyboarding activity that use the disk can be halted, Software Diagnostics loaded into memory and used, and keyboarding activity that is controlled by the disk resident Operating System can then be resumed with a normal WARM START.

NOTE:

The Software Diagnostic RESTORE and SIM commands can destroy the contents of the disk and should be used with caution.

The version of the Software Diagnostic tape is related to the version of the Operating System. Version 1.65 Software Diagnostics should only be used with Version 11.R1.00 of the Operating System. The following instructions describe Software Diagnostics Version 1.65 operations.

D.1. STARTUP

Starting Software Diagnostics is done as follows:

1. Mount the Software Diagnostic magnetic tape on the load unit (usually tape unit 1).

*Go to Ready Screen and Load
Software DIAG-tape*

- 2. If normal system operations and keyboarding are in progress, perform a supervisor POWEROFF prior to loading Software Diagnostics.
- 3. Do a tape load. *→ 1 + 3 Buttons*

Software Diagnostics will be read from the tape into memory. The contents of disk drives will not be disturbed. Software Diagnostics will prompt the operator at keystation 2 or the lowest numbered keystation on the system with the following message:

SFW DIAG V(1.65)
ENTER COMMAND - *Go to D-23*

Any function that is in progress can be terminated by Tape Load if the Software Diagnostic tape is still mounted on the load unit, or by simultaneously pressing the CTRL and BYPASS keys.

PERDEBUG (an element of Software Diagnostics) can be accessed by pressing the CTRL and BYPASS keys simultaneously. Software Diagnostics may be restarted from PERDEBUG by entering G300 and pressing the FIELD REL key. G is the PERDEBUG command for GO and 300 is the hexadecimal location where instructions will begin executing.

D.2. ~~COMMAND~~ SYNTAX

Software Diagnostic commands may take three formats depending on the modifier or argument associated with the command. Modifiers or arguments specify logical unit numbers, starting and/or ending address, printed lines per page, or character strings for headers. The command formats are as follows:

No argument:

CMNDFIELD REL

Commands with no arguments require only the command followed by FIELD REL. HELP and HEADER commands require no arguments.

Single argument:

CMND=a(FIELD REL)

Multiple arguments:

CMND=a,a,a,a(FIELD REL)

Single and multiple argument commands consist of the command and an equal sign (=) followed by the argument or arguments. Multiple arguments are separated by commas.

After each command is successfully executed, Software Diagnostics displays a DONE message followed by the ENTER COMMAND prompt. Wait until this DONE message appears before entering another command. Software Diagnostics is not multiprogrammed.

D.3. LOGICAL UNIT NUMBERING

Logical unit numbers used as arguments correspond to physical devices as follows:

Logical Unit Number	Physical Device
1-4	Tape units 1 through 4 when a tape unit argument is required.
5	Station Printer
6	Display
10	Line Printer
7,8,9,A	Disk drives 1,2,3,4
0,1,2,3	D-34 disk drive segments (platters)
0-1C	ISS disk drive segments

D.4. SUGGESTED OPERATIONAL PROCEDURES

Always observe the following rules when using Software Diagnostics:

1. Save the system disk drive if there is any doubt about adverse effects of commands that may be used. This is particularly important when using SIM or RESTORE commands.
2. Restore the disk after SAVE using Software Diagnostics.
3. Be sure that the write rings are in place on magnetic tapes to be written using Software Diagnostics. (Some tape write operations may appear to be successful without the presence of a write ring).

4. Use a list unit during a RECOVER to obtain a listing of batches saved.
5. Use the CTRL key to scroll one line at a time.
6. Use the DISP key to roll to the next page.
7. The command menu, HELP, is always available.

D.5. COMMANDS

D.5.1. Command Quick Reference

Mnemonic	Arguments	Function
BSF=	Multiple	Backspace a magnetic tape by 1 file.
BSR=	Multiple	Backspace a magnetic tape by 1 record.
COPY=	Multiple	Copy a magnetic tape.
DIFFD=	Multiple	Compare contents of two disk drives and list only the differences.
DIFFM=	Multiple	Compare two memory dump tapes and list only the differences.
DUMPM=	Multiple	Convert a memory dump tape to EBCDIC for printed listing.
DV=	Multiple	View the contents of a disk drive segment.
EOF=	Multiple	Write a Tape Mark on the magnetic tape unit specified.
HEADER	None	Allow entry of a character string that will appear on each page listed on the list device.
HELP	None	Display command menu.
INPUT=	Single	Set the input tape unit.
LIST=	Single	Set the list device.

Mnemonic	Arguments	Function
LPP=	Single	Set the number of lines per page to be printed for the list device.
MAKEMD=	Multiple	Make initialized memory dump tapes.
NOSIM=	Multiple	Return the tape unit simulated on disk to the physical tape unit.
OUTPUT=	Single	Set the output tape unit.
P=	Single	Set the PERDEBUG Program Counter offset value.
PLTH=	Single	Set the length of a page (in number of lines) for VFU control.
R=	Single	Set the PERDEBUG Program Relative offset value.
RECOVER=	Multiple	Corresponds to a Supervisor SAVE of JOBS from a disk drive.
RESTORE=	Single	Restore the contents of an entire disk drive that was written to magnetic tape using the Software Diagnostic SAVE command.
REW=	Multiple	Rewind the specified magnetic tape unit.
SAVE=	Multiple	Write the contents of a disk drive to a magnetic tape.
SIM=	Multiple	Specify the tape unit to be simulated on disk drive number 7.
SKF=	Multiple	Space a magnetic tape forward to the next tape mark.
VTOC=	Multiple	List the Volume Table of Contents on a disk drive.
WPL=	Multiple	Write a Pertec label on the magnetic tape unit specified.
X=	Single	Set the PERDEBUG X register offset value.

D.5.2. BSF

BSF spaces the magnetic tape on the unit specified backward one file.

BSF Command Syntax:

BSF=n(FIELD REL)
BSF=n,n,n,n(FIELD REL)

Where:

n
is the tape unit to be spaced backward.

D.5.3. BSR

BSR backspaces the magnetic tape one record on the unit specified.

BSR Command Syntax:

BSR=n(FIELD REL)
BSR=n,n,n,n(FIELD REL)

Where:

n
is the tape unit to be spaced backward.

D.5.4. COPY

COPY duplicates variable length blocks on the Output tape unit from the Input tape unit. A block can contain one or more logical records. The operator can specify the number of files to be copied and the maximum number of blocks that will be copied in any one file. The COPY will stop if this number of blocks is exceeded. If the list device has been set, each block will be listed as it is copied showing the file number, block number within the file, and block length in bytes. Tape marks will be listed when copied and the count shown. File number, block number, and block length are shown in decimal; tape mark count is shown in hexadecimal.

COPY Command Syntax:

COPY=NF,Nb(FIELD REL)

Where:

NF

is the number of files to be copied from INPUT to OUTPUT in hexadecimal.

Nb

is the maximum number of blocks that will be copied in any one file in hexadecimal.

D.5.5. DIFFD

DIFFD compares the contents of two disk drives and lists the contents only when differences are encountered. DIFFD provides a listing like the DV command (D.5.8). DIFFD is used to isolate problems. It enables software development programmers to find the differences between a known disk drive and a user disk drive that has problems.

DIFFD Command Syntax:

DIFFD=Dn, Ls, Hs, Dn2

Where:

Dn

is the disk drive logical unit number to be evaluated (7,8,9,A).

Ls

is the low sector; the hexadecimal sector address where the comparison is to start.

Hs

is the high sector; the hexadecimal sector address where the comparison is to end.

Dn2

is the disk drive logical unit number to which Dn is to be compared (7,8,9,A).

DIFFD Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. Save the contents of platters to be used for DIFFD if necessary.	
3. Restore the contents of the disk to be evaluated.	
4. Restore the contents of the disk that the drive in step 3 is to be compared to.	
5. LIST=n (FIELD REL) Enter comments or title information on the first page of the listing by preceding the character string with an asterisk (*).	n is 1 through 4 for a tape unit with a scratch tape mounted where the dump will be written, or 10 for line printer. See the LIST command (D.5.13) for additional prompts and responses.
6. DIFFD=7,78F,83A,8 (FIELD REL)	7 is the disk drive to be evaluated. 78F is the starting sector in hex. 83A is the ending sector in hex. 8 is the disk drive that drive 7 will be compared to.
7. KEY IN HEADER STRING	Allows entry of up to 30 characters of header information that will appear on every subsequent page of the listing.
8. XXXXXXXXXXXXXXXX (FIELD REL)	Header character string.
9. OUTPUT FOR WIDE PAPER? TYPE Y FOR YES,REL=NO	A Y response results in a 129-character print line for 14 7/8 by 11-inch printer paper. Pressing the FIELD REL key gives an 81-character print line.

D.5.6. DIFFM

DIFFM converts a memory dump tape to EBCDIC, compares each line of the dump to the same lines of a known Operating System memory dump, and lists only those that differ on the list device assigned by the LIST command. As with DUMPM, lines can be 128 or 132-character records. The listing shows the absolute memory address, relative address, contents of memory locations in hexadecimal, and the EBCDIC character equivalent of the contents of memory location.

The known Operating System memory dump (for comparison purposes) is made using an initialized dump tape. It should be created when no activities are occurring on the system. The Operating System should be loaded into memory with a WARM or COLD START. The memory dump should be taken when "READY--USE QMND KEY TO START WORK" is displayed on the keystation where date, time, and shift were entered. No keys should have been pressed on any other keystation on the system. This memory dump should be listed for reference purposes using the DUMPM command.

Successful use of DIFFM is based on both memory dumps resulting from the same version of the Operating System. The low-core resident addresses will be the same.

When a difference in low-core resident addresses is encountered, Software Diagnostics prompts the operator with the following:

```
INCOMPATIBLE SYSTEMS, CONTINUE?  
MASTER=XX.XX  
THIS MOD=YY.YY  
TYPE Y FOR YES, REL=NO
```

The operator should ensure that the versions are the same. Pressing the FIELD REL key will terminate the DIFFM and allow the operator to mount the correct tape. A Y response causes the DIFFM to continue; however, the results may not be useful for analysis purposes.

DIFFM Command Syntax:

```
DIFFM=n1,n2(FIELD REL)
```

Where:

- n1 is the tape unit with the memory dump tape to be evaluated.
- n2 is the tape unit with the known Operating System memory dump tape.

DIFFM Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. LIST=n (FIELD REL) Enter comments or title information on the first page of the listing by preceding the character string with an asterisk (*).	n is 1 through 4 for a tape unit with a scratch tape mounted where the dump will be written (not the tape unit with the dump tapes mounted) or 10 for line printer. See the LIST command (D.5.13) for the additional prompts and responses.
3. DIFFM=1,2	1 is the tape drive with the memory dump tape to be evaluated. 2 is the tape drive with the known Operating System memory dump tape.
4. KEY IN HEADER STRING (30 CHAR MAX)	Allows entry of up to 30 characters of header information that will appear on every subsequent page of the listing.
5. XXXXXXXXXXXXX (FIELD REL)	Header character string.
6. OUTPUT FOR WIDE PAPER? TYPE Y FOR YES, REL=NO	A Y response results in a 129-character print line for 14 7/8 by 11-inch printer paper. Pressing the (FIELD REL) key gives an 81-character print line.
7. UNWIND TWAs? TYPE Y FOR YES, REL=NO	A Y response causes each TWA in the dump to be followed by an alphabetical list of TWA (Terminal Work Area) memory cells identified by their Common Name Directory name, their contents, and relative address within that TWA.

D.5.7. DUMPM

DUMPM converts a memory dump tape to EBCDIC and lists it on the list device assigned by the LIST command. A memory dump tape consists of 8192-byte blocks of memory image whether the dump was taken using an initialized dump tape or using the Operating System dump routine. DUMPM converts these blocks of memory to 128 or 132-character records in EBCDIC for analysis purposes. The converted memory dump shows Operating Systems module boundaries, TWA boundaries, absolute memory addresses, addresses relative to the start of the module, contents of memory locations in hexadecimal, and the EBCDIC character equivalent for the contents of memory locations.

DUMPM Command Syntax:

DUMPM=n,Rf(FIELD REL)

Where:

n
is the input magnetic tape unit where the dump tape to be converted is mounted.

Rf
is the **rec flag**. It indicates whether or not module boundaries are to be recognized. Rf=0 specifies recognition of module boundaries. Rf=1 specifies no recognition of module boundaries (memory contents will be dumped as one module).

DUMPM Command Sequence:

The command sequence for a system with two magnetic tape drives or a single tape drive and line printer is as follows:

Command	Definition
1. Load Software Diagnostics.	
2. LIST=n(FIELD REL)	n is 1 through 4 for a tape unit with a scratch tape mounted where the dump will be written (not the tape unit with the dump tape mounted), or 10 for line printer. See the LIST command (D.5.13) for the additional prompts and responses.

Command	Definition
3. * *25 JAN 83 *DUMPED BY HARRY BRANSON *SYSTEM ERROR 93 CAUSED *BY TERMINAL 3 IN COMMAND *MANAGER DOING DATA COMM *USING HASP *	Title information can be entered on the first page of the listing by preceding each line with an asterisk (*).
4. DUMPM=n,0 (FIELD REL)	n is the tape drive where the memory dump tape is mounted. 0 specifies recognition of module boundaries.
5. KEY IN HEADER STRING (30 CHAR MAX)	Allows entry of up to 30 characters of header information that will appear on every subsequent page of the listing.
6. XXXXXXXXXXXXX (FIELD REL)	Header character string.
7. OUTPUT FOR WIDE PAPER? TYPE Y FOR YES, REL=NO	A Y response results in a 129-character print line for 14 7/8 by 11-inch printer paper. Pressing the (FIELD REL) key gives an 81-character print line.
8. UNWIND TWAS? TYPE Y FOR YES, REL=NO	A Y response causes each TWA in the dump to be followed by an alphabetical list of TWA memory cells identified by their Common Name Directory name, their contents, and relative address within that TWA.

The command sequence for a system with one magnetic tape drive is as follows:

Command	Definition
1. Load Software Diagnostics.	
2. Save the system disk drive.	
3. Mount the memory dump tape.	

Command	Definition
4. INPUT=1	Sets the input device to the tape drive with the memory dump tape.
5. SIM=2	Specifies that tape unit 2 will be simulated on the system disk drive.
6. OUTPUT=2	Sets the output tape unit to the simulated drive.
7. COPY=1 (FIELD REL)	Copies the memory dump to the simulated tape unit.
8. REW=2	Rewinds the simulated tape drive to load point.
9. Dismount the dump tape and mount a scratch tape.	
10. LIST=1	Sets the list device to the tape unit. See the LIST command (D.5.13) for the additional prompts and responses.
11. * *25 JAN 83 *DUMPED BY HARRY BRANSON *SYSTEM ERROR 93 CAUSED *BY TERMINAL 3 IN COMMAND *MANAGER DOING DATA COMM *USING HASP *	Comments or title information can be entered on the first page of the listing by preceding each line with an asterisk (*).
12. DUMPM=2,0 (FIELD REL)	Specifies the INPUT TAPE as the simulated drive and recognition of module boundaries.
13. KEY IN HEADER STRING	Allows entry of up to 30 characters to header information that will appear on every subsequent page of the listing.
14. XXXXXXXXXXXXX (FIELD REL)	Header character string.
15. OUTPUT FOR WIDE PAPER? TYPE Y FOR YES, REL=NO	A Y response results in a 129-character print line for 14 7/8 by 11-inch printer paper. Pressing the (FIELD REL) key gives an 81-character print line.

Command	Definition
16. UNWIND TWAS? TYPE Y FOR YES, REL=NO	A Y response causes each TWA in the dump to be followed by an alphabetical list of TWA memory cells identified by their Common Name Directory name, their contents, and relative address within that TWA.
17. Restore the System disk that was saved in step 2.	

D.5.8. DV

DV lists the contents of a disk unit on the list device. DV allows the programmer to view the disk and diagnose problems. The address of data on the disk may be obtained from the VTOC or, in the case of JOBS, from a RECOVER with save flag=0. The operator specifies the disk and the starting and ending sector of the area to be dumped. The disk dump shows sector address, the offset of the line relative to the start of the sector, the contents of the disk in hexadecimal, and the EBCDIC character equivalent of the contents of the disk.

DV Command Syntax:

DV=Dn, Ls, Hs, Sn

Where:

Dn is the disk logical unit number (7,8,9,A).

Ls is the low sector; the hexadecimal sector address where the dump is to start.

Hs is the high sector; the hexadecimal sector address where the dump is to end.

Sn is the starting segment number index (0 through 1C in hexadecimal).

DV Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. LIST=n(FIELD REL)	n is 6 for display, 1 through 4 for a magnetic tape, or 10 for line printer. See the LIST command (D.5.13) for the additional prompts and responses.
3. DV=Dn, Ls, Hs, Sn(FIELD REL)	Dn is the platter number to be viewed. Ls is the starting sector in hexadecimal. Hs is the ending sector in hexadecimal. Sn is the segment number.

D.5.9. EOF

EOF writes a file mark on the magnetic tape on the unit specified.

EOF Command Syntax:

EOF=n(FIELD REL)
EOF=n,n,n,n(FIELD REL)

Where:

n
is the tape unit where the file mark is to be written.

D.5.10. HEADER

HEADER allows the operator to enter up to 30 characters that will appear in every subsequent page header on the list device.

HEADER Command Syntax:

HEADER(FIELD REL)
KEY IN HEADER STRING (30 CHAR MAX)
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXX(FIELD REL)

Where:

X....
is up to 30 characters. Alphanumeric and symbols are allowed.

D.5.11. HELP

HELP lists all the currently implemented commands on the command keystation.

HELP Command Syntax:

HELP (FIELD REL)

HELP Command Menu:

BSF=
BSR=
COPY=
DIFFD=
DIFFM=
DUMPM=
DV=
EOF=
HEADER=
HELP
INPUT=
LIST=
LPP=
MAKEMD=
NOSIM=
OUTPUT=
P=
PLTH=
R=
RECOVER=
RESTORE=
REW=
SAVE=
SIM=
SKF=
VTOC=
WPL=
X=

D.5.12. INPUT

INPUT defines the logical magnetic tape input unit number for RESTORE, COPY, and SIM functions. The tape number must be in the range between 1 and 4.

INPUT Command Syntax:

INPUT=n(FIELD REL)

Where:

n
is the magnetic tape unit number.

D.5.13. LIST

LIST defines the alternate list device. Commands, operator prompts, headers, comments, and the output of DUMPM, DIFFD, DIFFM, DV, and VTOC commands are written to the list device. A list unit can be the display, line printer, magnetic tape, or a simulated magnetic tape.

LIST Command Syntax:

LIST=n(FIELD REL)

Where:

n
is the logical unit number. 1 through 4 are used for magnetic tape units, 6 for display, and 10 for line printer. LIST=0 turns off the list device. In the case of magnetic tape, LIST=0 causes a file mark to be written and the tape rewound.

If **n** is a magnetic tape unit, the following prompts will be displayed:

PACKED?
TYPE Y FOR YES, REL=NO

A packed list tape consists of variable length records containing an end-of-line character (hex EODF) and a page eject character (hex E05F). A block will be a maximum of 2039 characters. A packed list tape requires a special list program—it cannot be listed using the Supervisor TRANSFER command. Pressing FIELD REL causes a fixed length record list tape to be written. The record length and top-of-form characters are specified by the next prompt.

ASA PRINT CONTROL—EBCDIC?
TYPE Y FOR YES, REL=NO

A Y response causes the list tape to be formatted into 128-character records, blocked 5, with a top-of-form character of F140 hex. Pressing FIELD REL causes the list tape to be formatted into 132-character records, blocked 5, with a top-of-form character of 0C hex. This format is suitable for listing the tape using the Supervisor TRANSFER command. The LPP (line per page) command determines the record where the top-of-form character will be placed.

**CONTINUATION TAPE?
TYPE Y FOR YES, REL=NO**

A Y response causes the tape to be skipped to the first file mark and then backspaced over this file mark prior to writing to the tape. Pressing FIELD REL causes writing to begin at the start of the tape.

LIST Command Sequence:

The command sequence for LIST= a magnetic tape is as follows:

Command	Action
1. LIST=n(FIELD REL) TYPE Y FOR YES, REL=NO	n is 1 through 4 for a magnetic tape.
2. PACKED? TYPE Y FOR YES, REL=NO	
3. (FIELD REL)	Write a fixed length record list tape.
4. ASA PRINT CONTROL—EBCDIC? TYPE Y FOR YES, REL=NO	
5. (FIELD REL)	Write 132-character records, blocked 5, suitable for listing using the Supervisor TRANSFER command.
6. CONTINUATION TAPE? TYPE Y FOR YES, REL=NO	
7. (FIELD REL)	Begin writing at the start of the tape.

If n is 10 for line printer, the following prompts will be displayed:

**ASA PRINT CONTROL—EBCDIC?
TYPE Y FOR YES, REL=NO**

A load response causes top-of-form to be issued when a specified number of lines have been printed on each page. The number of lines is determined by the LPP command. Pressing (FIELD REL) indicates no top of form.

LOAD VFU FOR LINE PRINTER?
TYPE Y FOR YES, REL=NO

A (FIELD REL) causes no use of the VFU or no top of form. A Y response, along with a Y response to "ASA PRINT CONTROL--EBCDIC?" will use the VFU. A Y response displays the following:

VFU LOADED WITH THESE PARAMETERS:
PAGE LTH (PLTH=) 33 HEX
PRINT LINES PER PAGE (LPP=) 25 HEX

The page length (in number of lines) and lines that will be printed before top of form are 51 (33 hex) and 37 (25 hex), respectively. These should be changed prior to the LIST command to the correct numbers for the paper being used.

The command sequence for LIST=10, line printer is as follows:

Command	Action
1. PLTH=42 (FIELD REL)	Set the page length at 66 lines for 11-inch paper.
2. LPP=3C (FIELD REL)	Set the number of lines to be printed per page at 60, 3 lines of space before and after the perforation.
3. LIST=10 (FIELD REL)	Set the list device to the line printer.
4. ASA PRINT CONTROL--EBCDIC? TYPE Y FOR YES, REL=NO	
5. Y (FIELD REL)	Use of the vertical format unit in the line printer is planned.
6. LOAD VFU FOR LINE PRINTER? TYPE Y FOR YES, REL=NO	
7. Y (FIELD REL)	

Command	Action
8. VFU LOADED WITH THESE PARAMETERS: PAGE LTH (PLTH=) 42 HEX PRINT LINES PER PAGE (LPP=) 3C HEX	

D.5.14. LPP

LPP sets the number of lines that will be printed on a page on the list unit before top of form to the next page. When the list unit is the line printer, it controls the vertical format unit. When the list unit is a magnetic tape, LPP places a top-of-form character (0C hex) in the first position of the record after the number of lines has been written. See the LIST command (D.5.13) for use.

LPP Command Syntax:

LPP=hh (FIELD REL)

Where:

hh

is the number of lines per page in hexadecimal.

D.5.15. MAKEMD

MAKEMD creates a memory dump tape--an initialized dump tape--on the unit specified. MAKEMD writes a 26-word dump routine at the beginning of the tape. Once initialized, a dump tape can be reused without reinitialization, even after it has been listed using DUMPM or DIFEM commands.

Memory dumps are used by software development programmers to determine the cause of system errors, task errors, and system and keystation lockups. A system error or task error allows access of the System Dump Module by simultaneously pressing the LCA SHIFT, RESET, and DISPLAY keys. Pressing these keys displays the system register contents that should be recorded on the Software User Report. The operator will then be prompted to take a memory dump and disk dump to a scratch magnetic tape mounted on the system load unit. If the Y (YES) response to the memory dump prompt does not cause the dump to be taken (the system error or task error has prevented access to the System Dump Module) or the problem is a system or keystation lockup, the memory dump must be taken using a tape that has been initialized with the Software Diagnostic MAKEMD command. The dump tape must have a write ring in place when it is initialized by Software Diagnostics and when the dump is to be

taken. When the problem is encountered, mount the initialized dump tape on the load unit and press the TAPE LOAD button. The contents of memory will be dumped. After the tape has stopped moving, operations can be continued with a WARM START.

MAKEMD Command Syntax:

MAKEMD=n(FIELD REL)
MAKEMD=n,n,n,n(FIELD REL)

Where:

n
is the tape unit where the initialized dump tape is to be created.

D.5.16. NOSIM

NOSIM causes all future I/O calls to the tape unit specified to be routed to the physical tape unit rather than the TAPE SIMULATOR on disk.

NOSIM Command Syntax:

NOSIM=n(FIELD REL)
NOSIM=n,n,n,n(FIELD REL)

Where:

n
is the tape unit that has been simulated on disk.

D.5.17. OUTPUT

OUTPUT defines the logical magnetic tape output unit number for COPY, RECOVER, SAVE, and SIM functions. The tape number must be in the range between 1 and 4.

OUTPUT Command Syntax:

OUTPUT=n(FIELD REL)

Where:

n
is the magnetic tape unit number.

D.5.18. P

P sets the PERDEBUG program counter offset to the hexadecimal value specified.

P Command Syntax:

P=hhhh; (FIELD REL)

Where:

hhhh

is a four-digit value in hexadecimal.

D.5.19. PLTH

PLTH sets the number of lines that can be printed on a page in the vertical format unit when the alternate list device is the line printer. See the LIST command (D.5.13) for use.

PLTH Command Syntax:

PLTH=hh (FIELD REL)

Where:

hh

is the number of lines in hexadecimal.

An 11-inch-long page (for 14 7/8 by 11-inch computer paper) has 66 lines decimal, 42 hex. An 8 1/2-inch-long page has 51 lines decimal, 33 hex.

D.5.20. R

R sets the PERDEBUG relative offset to the hexadecimal value specified.

R Command Syntax:

R=hhhh (FIELD REL)

Where:

hhhh

is a four-digit memory address in hexadecimal.

Software Diag.

SOURCE BATCHES
DATA BATCHES

D.5.21. RECOVER

RFCOVER is equivalent to a Supervisor SAVE of JOBS, only, from a disk drive. Its objective is to retrieve as many data batches as possible. The Operating System can be reloaded from a System Tape. The User's Library of object programs can be restored from a SAVE tape. Data batches might require several hours to be rekeyed.

RFCOVER reads disk allocation units (16, 84-byte sectors) and saves data batches based on linkages that connect the Allocation Units (AUs) into batches. It does not use the VTOC for that disk drive.

RECOVER Command Syntax:

RECOVER=Dn,Sf (FIELD REL)

Where:

Dn
is the system logical disk drive number (7,8,9,A).

Sf
is the save flag. It indicates whether or not batches are to be saved.

Sf
SF=1 saves batches. SF=0 does not. In conjunction with LIST=6 (Display), SF=0 will show the operator the starting disk address of batches without saving batches to magnetic tape.

RECOVER Command Sequence:

Remove DTA & - Mount Scratch tapes

- | Command | Definition |
|--|---|
| 1. Load Software Diagnostics. | |
| 2. <u>OUTPUT</u> =n(FIELD REL) <i>1 - ju</i>
<i>Close</i> | n is the tape unit with a scratch tape mounted where batches will be saved. (Default = T1.) |
| 3. <u>LIST</u> =n(FIELD REL) <i>5 - ju</i>
<i>= Station printer</i> | n is 6 for display, 1 through 4 for a magnetic tape, or 10 for line printer. Batches that are saved will be listed on the list unit. See the LIST command for the additional prompts and responses. |

Command

Definition

4. RECOVER=Dn,l(FIELD REL)

Dn is the disk drive to be recovered.

*7, 7 for
recover disk drives
logical disk*

Use CTRL key or DISPLAY key to page through any I/O BSR messages and through the list of batches being saved. Respond Y for YES to the "ACCEPT BLOCK" prompt that may result from a link error. A FIELD REL to the "ACCEPT BLOCK" prompt will terminate the RECOVER without the entire drive being searched for batches. The objective is to RECOVER as many batches as possible.

= Y

When the RECOVER is done for this drive, the tape will rewind and the number of batches saved will appear on the list unit.

Can't do a Cold Start

Batches from another disk may be saved to the same tape as follows:

5. SKF=n

Space the magnetic tape in step 2 to the first file mark.

6. BSR=n

Backspace the magnetic tape in step 2 over the file mark encountered in Step 5.

Repeat step 4 for the next disk. Steps 4, 5, and 6 may be repeated as many times as required.

D.5.22. RESTORE

The Software Diagnostic RESTORE is not the same as a Supervisor RESTORE.

RESTORE writes the contents of a magnetic tape created by a Software Diagnostic SAVE to a disk drive. The disk drive for the RESTORE does not have to be the same as the one specified during the SAVE. Each segment on the SAVE tape will be restored to the corresponding segment on the disk drive specified by the RESTORE command.

RESTORE replaces the current contents of the disk drive specified with the disk contents on the magnetic tape in preparation for a Software Diagnostic DIFFD or to reload the user's Operating System for online problem diagnosis. RESTORE is also used to return the original contents of the system drive after it has been used to simulate (SIM=) a magnetic tape.

RESTORE Command Syntax:

RESTORE = Dn

Where:

Dn

is the disk number (7,8,9,A; default 7).

RESTORE Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. INPUT=n(FIELD REL)	n is the tape unit with the SAVE tape mounted. (Default = T1.)
3. RESTORE=Dn(FIELD REL)	Dn is the disk drive where the contents of the SAVE tape will be deposited. This causes the disk drive and segments to be overwritten with the contents of the SAVE tape.

D.5.23. REW

REW rewinds the magnetic tape on the unit specified.

REW Command Syntax:

REW=n(FIELD REL)
REW=n,n,n,n(FIELD REL)

Where:

n

is the tape unit to be rewound.

D.5.24. SAVE

SAVE writes the contents of the disk drive specified to magnetic tape in Software Diagnostic SAVE format. A magnetic tape created using the Software Diagnostic SAVE can be restored only with a Software Diagnostic RESTORE. The Software Diagnostic SAVE command is not equivalent to a Supervisor SAVE.

SAVE is normally used after a system error when a memory dump has been taken and when the contents of a disk drive will be required for diagnosis of the system error. The contents of the system disk drive can be used for a Software Diagnostic DIFFD or to reload the user's Operating System for online diagnosis. It can also be used to save the system disk prior to use of that drive to simulate (SIM=) a magnetic tape. After use of SIM=, the contents of the System disk can be returned using a Software Diagnostic RESTORE.

SAVE Command Syntax:

SAVE = Dn,Sn,Ns

Where:

Dn
is the disk drive number (7,8,9,A; default = 7).

Sn
is the starting segment number in hex (default =0).

Ns
is the number of segments to be saved, in hex (default = 1).

SAVE Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. OUTPUT=n(FIELD REL)	n is the tape unit with a scratch tape mounted. (Default = T1.)

3. **SAVE=Dn,Sn,Ns(FIELD REL)**

Dn is the disk drive number
(7,8,9,A).

Sn is the starting segment number
index.

Ns is the number of segments in
hex. (Maximum of 4 for a D-34
disk drive. Maximum of 1D for an
ISS disk drive).

When the tape is full, Software
Diagnostics will display an error
message (from IOCTL). Press the
space bar once. The tape will be
backspaced to the last EOF. Two
EOFs will be written and the tape
will rewind. Software Diagnostics
will prompt for a continuation
tape. Mount a scratch tape and
enter Y. The prompt should not be
answered while the tape is
rewinding.

D.5.25. SIM

SIM causes all future I/O calls to the tape unit specified to be
routed to the TAPE SIMULATOR on disk. Up to four tapes can be
simulated at one time, each using one-fourth of the system disk.
The TAPE SIMULATOR will destroy the contents of the system disk.
The Software Diagnostic SAVE command should be used to retain the
current contents of the system disk for subsequent use. Simulated
tapes on the disk do not rewind automatically at the completion of
an I/O operation. The operator must be sure to rewind the simulated
device when necessary using the REW command.

SIM Command Syntax:

SIM=n(FIELD REL)

SIM=n,n,n,n(FIELD REL)

Where:

n
is the tape unit to be simulated on disk.

SIM Command Sequence:

SIM command sequence for COPY in a one tape, one disk system:

Command	Action
1. Load Software Diagnostics.	
2. SAVE the contents of the system disk if necessary.	
3. INPUT=1 (FIELD REL)	Set the input tape unit where the tape to be copied is mounted.
4. SIM=2 (FIELD REL)	Define the simulated tape unit.
5. OUTPUT=2 (FIELD REL)	Set the output tape unit to the simulated drive.
6. COPY=1 (FIELD REL)	Copy one file without a record number limitation.
7. Dismount the original and mount a scratch tape with a write ring in place.	
8. REW=2 (FIELD REL)	Rewind the simulated tape.
9. INPUT=2 (FIELD REL)	Set the input tape unit to the simulated drive.
10. OUTPUT=1 (FIELD REL)	Set the output tape unit to the scratch tape.
11. COPY=1 (FIELD REL)	Copy the file to the magnetic tape.

D.5.26. SKF

SKF spaces the magnetic tape on the unit specified forward to the next file mark.

SKF Command Syntax:

SKF=n (FIELD REL)
SKF=n,n,n,n (FIELD REL)

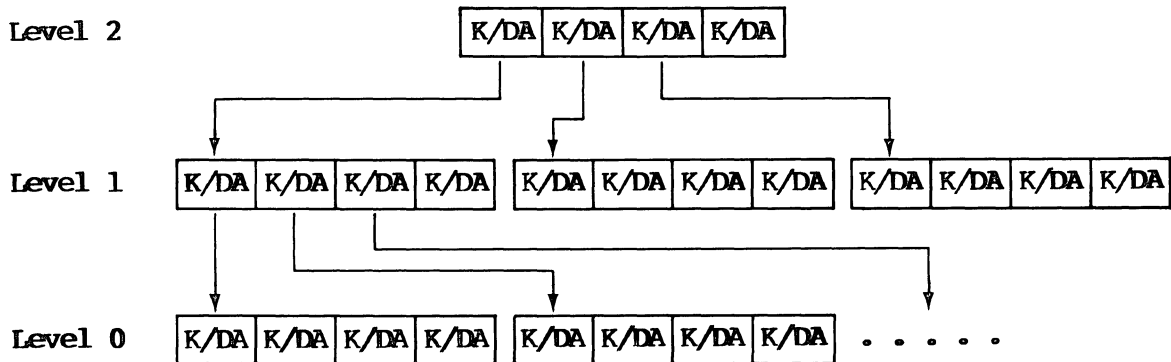
Where:

n
is the tape unit to be spaced forward.

D.5.27. VTOC

VTOC lists the Volume Table Of Contents (VTOC) of the disk drive specified on the list device--display, line printer, or magnetic tape. It is used to obtain a listing of the entries in the VTOC, to aid in diagnosis of problems with the VTOC, or to locate data stored on a disk drive. The VTOC is used by the Operating System to retrieve the disk address of Operating System modules, COMPILES, JOBS, and DIRECTORIES. The VTOC considers indexed files to be JOBS with different, special batch numbers. Each disk drive used by the Operating System contains a VTOC for that drive. The VTOC is built for the System drive by the Operating System during startup after a COLD START. VTOCs for other drives are created during DISK LOAD, MOUNT, or NEW DISK.

The VTOC is indexed by key and disk address and may contain multiple levels. Each level is separated into pages of 17 entries. The following diagram shows the construction of a VTOC with 3 levels.



The VTOC is similar to an indexed file but may be accessed only by the Operating System. Level 0 contains the key (4 words) and disk address (2 words) of all Operating System modules, COMPILES, JOBS, and DIRECTORIES stored on that drive. Level 1 contains the key and disk address of the start of each Level 0 page. Each higher level (2, 3, etc.) follows the format of key and disk address of the start of each lower level page. A VTOC search or location of data on a drive is accomplished by retrieving VTOC pages, starting with the highest level, a binary search in memory, until the key and disk address of the system module, COMPILE, JOB, or DIRECTORY is found in Level 0.

System modules, COMPILES, JOBS, or DIRECTORIES can be located on the disk using the VTOC command by examining the entries in Level 0. After the address is determined, the system module, COMPILE, JOB, or DIRECTORY can be examined using the Disk View (DV) command.

VTOC Command Syntax:

VTOC=Kl,Sd,Dn

Where:

Kl

is the key list flag. It indicates whether or not the listing will include symbolic reference to the keys in the VTOC. Kl=1 specifies a symbolic listing; 0 does not.

Sd

is the sector dump flag. It indicates whether the listing will include a disk dump of all sectors comprising the VTOC as if a DV command were used for these sectors. Sd=1 specifies a disk dump. SD=0 does not. A disk dump is normally useful to software development and support programmers.

Dn

is the disk number (7,8,9,A).

VTOC Command Sequence:

Command	Definition
1. Load Software Diagnostics.	
2. LIST=n(FIELD REL)	n is 6 for display, 1 through 4 for a magnetic tape, or 10 for line printer. If the list unit is a magnetic tape or line printer, see the LIST command (D.5.13) for the additional prompts and responses.

Command	Definition
3. VTOC=1,0,Dn(FIELD REL)	1 specifies a symbolic listing. 0 specifies no disk dump. When n is a magnetic tape, contents of the VTOC will not appear on the display and the magnetic tape must be printed to obtain a hard copy. A response of Dn,0,0 causes no listing to be made and an immediate Software Diagnostic DONE. Dn is the disk drive to be listed.

D.5.28. WPL

WPL writes a Pertec label on the magnetic tape on the unit specified. The Pertec label is an 80-byte record that contains I/O control block information about the tape such as record length, blocking factor, records per block, translation used, number of reverse reads for continue tape, etc. WPL writes two tape marks, the Pertec label, two tape marks and rewinds the tape.

WPL Command Syntax:

WPL=n(FIELD REL)
WPL=n,n,n,n(FIELD REL)

Where:

n
is the tape unit where the Pertec label is to be written.

D.5.29. X

X sets the PERDEBUG index register offset to the hexadecimal value specified.

X Command Syntax:

X=hhhh(FIELD REL)

Where:

hhhh
is a four-digit value in hexadecimal.

E. System Generation (SYSGEN)

The following description details the various responses displayed during system generation (SYSGEN) procedure. Before beginning the SYSGEN procedure, all new data should be saved.

SYSGEN WITH SAMPLE

Display	Operator Action
1.	Mount the System Master Binary (SMB) tape on the load unit (primary tape drive). Simultaneously press the IPL Release key (blank white pushbutton) and the LOAD key on the system control panel. The bootstrap loader will be read into memory and will, in turn, load and transfer control to SYSGEN.

CAUTION

Wait a reasonable period of time for the system to respond to an operator entry. Do not press the FIELD RELEASE key (FIELD REL) repeatedly because nothing appears to be happening. The additional FIELD RELs will be retained in the keystation buffer and used as the response to a prompt, and will prevent the operator from responding at the appropriate time.

2. ***** MASTER VERSION NO. XX.XX.XX *****
LOAD MASTER TAPE TO DISK? REL=YES, N=NO

DISK INITIATION FOR LOAD IN PROGRESS

Display

Operator Action

3. LOAD SYSGEN OVERLAYS TO DISK

NO	NAME	VERSION	TYP	ID	STATUS
001	S BLD LC	11.72	014	001	LOAD
002	S GEN LC	11.59	014	001	LOAD
003	S GEN LC	11.13	014	002	LOAD
004	S LD TBLS	11. 6	014	001	LOAD
005	S P BUF1	11.18	014	001	LOAD

END OF FILE
TO SELECT MODS, TYPE: Y=YES, REL=NO

Press the FIELD REL key. Disk initiation takes approximately 2 minutes.

NO	NAME	VERSION	TYP	ID	STATUS
001	S BLD LC	11.72	014	001	LOADED
002	S GEN LC	11.59	014	001	LOADED
003	S BLD MSG	11.13	014	002	LOADED
004	S LD TBLS	11. 6	014	003	LOADED
005	S P BUF1	11.18	014	001	LOADED

Press the FIELD REL key. SYSGEN modules should not be selected.

*Step 7
msgs*

4. * SYSGEN VERSION NO. XX.XX.XX ***
TYPE SYSGEN COMMAND
ALU=16
ALU=N SET ALTERNATE LIST UNIT = N**

A listing of prompts, messages, operator responses, and patches applied to modules on the SMB should be obtained if the system configuration includes a line printer or a second magnetic tape drive.

Typing ALU=2, then pressing the FIELD REL key directs the listing to tape drive 2.

Typing ALU=5, then pressing the FIELD REL key directs the listing to the station printer.

Typing ALU=16, then pressing the FIELD REL key directs the listing to the line printer. If a listing is not to be produced, go to step 7.

Display

Operator Action

5. TYPE SYSGEN COMMAND
LC=1
LC=N SET LIST COMMAND FLAG = N

Typing LC=1, then pressing the FIELD REL key logs all operator entries to the Alternate List Unit.

6. TYPE SYSGEN COMMAND
COM
COM TYPE COMMENTS FOR LIST UNIT
GDS DONE BY HARRY BRANSON, 5/26/83,
TO DOCUMENT SYSTEMS GENERATION OF
VERSION 11.R1

Typing COM, then pressing the FIELD REL key allows the entry of comments to identify the listing that is obtained. Pressing the FIELD REL key only terminates entry of comments.

✓ 7. TYPE SYSGEN COMMAND
GDS
GDS GENERATE DISK SYSTEM
NO NAME VERSION TYP ID STATUS
001 S BLD LC 11.72 014 001 LOAD
002 S GEN LC 11.59 014 001 LOAD
003 S BLD MSG 11.13 014 002 LOAD
. . .

Type GTS, then press the FIELD REL key; or type GDS, then press the FIELD REL key. GTS (Gen-Tape System) generates tape system. GDS (Generate Disk System) allows the entry of additional patches (patches not resident on the SMB) by the CDM (Correct Disk Module) prior to writing the system tape (step 138).

*revisions
could start here*

328 SYS INFO 11. 1 003 267 LOAD
END OF FILE

The names of the modules in the operating system will be rolled through the screen.

Display

Operator Action

8. TO SELECT MODS, TYPE: Y=YES, REL=NO
Y
TO REVERSE LOAD STATUS, TYPE:
Y=YES, REL=NO, T=TERMINATE

R = RECYCLE

001	S BLD LC	11.72	014	001	LOAD
002	S GEN LC	11.60	014	001	LOAD
003	S BLD MSG	11.13	014	002	LOAD
004	S LD TBL	11. 6	014	003	LOAD
005	S P BUFL	11.18	014	001	LOAD
006	ATTRIBITS	11. 4	000	000	LOAD
007	ATTGERMAN	11. 2	000	000	SKIP
008	TURK ATTR	11. 5	000	000	SKIP
009	ATTRIUK	11. 2	000	000	SKIP
010	NORDICATT	11. 7	000	000	SKIP
011	ISWATTRBT	11. 3	000	000	SKIP
012	ATTIGREEK	11. 4	000	000	SKIP
013	SPAINATT	11. 2	000	000	SKIP
014	THAI ATTR	11. 4	000	000	SKIP
015	KEYPCH KT	11.11	000	254	SKIP
016	ADDMCH KT	11.11	000	254	SKIP
017	TYPE KT	11.13	000	254	SKIP
018	KP KT ASC	11.11	000	254	SKIP
019	KP AM ASC	11.13	000	254	SKIP
020	TYP ASCII	11.14	000	254	SKIP
021	ASC KP 43	11. 4	000	254	SKIP
022	ASC ADD43	11. 5	000	254	SKIP
023	ASC TYP43	11. 5	000	254	SKIP
024	XL KEYPCH	11.12	000	254	LOAD
025	GERMAN KP	11. 7	000	254	SKIP
026	XL TURK	11.11	000	254	SKIP
027	UK KEYPCH	11. 6	000	254	SKIP
028	XLKP SWFI	11.12	000	254	SKIP
029	SFTYPEWR	11.15	000	254	SKIP
030	ISWTYPEWR	11. 3	000	254	SKIP
031	XLKP DANO	11.11	000	254	SKIP
032	XL KANA	11. 8	000	254	SKIP
033	UK ADDMCH	11. 6	000	254	SKIP
034	GREEK KP	11. 8	000	254	SKIP
035	SPANSH KP	11. 5	000	254	SKIP
036	FRCH KP A	11. 5	000	254	SKIP
037	XL ADDMCH	11.10	000	254	SKIP
Y					
037	XL ADDMCH	11.10	000	254	LOAD
038	GERMANADD	11. 4	000	254	SKIP
039	DANOR ADD	11.12	000	254	SKIP
040	SF ADDMCH	11. 8	000	254	SKIP

Type Y, then press the FIELD REL key.

Press FIELD REL to scroll through modifications.

For each line, pressing the FIELD REL key does not change the load status. The next module will be displayed.

Typing Y, then pressing the FIELD REL key reverses the load status---SKIP will be changed to LOAD or LOAD will be changed to SKIP.

Typing T, then pressing the FIELD REL key terminates module selection. Does not affect load status of the module where it is entered (step 9). Typing R, then pressing the FIELD REL key, returns the review to the beginning of the module list. Load status of modules are not affected.

Load status of only modules 015-024, 037, 041, 056, and 057 should be changed to support the system configuration.

Display					Operator Action	
041	XL TYPWRT Y	11.14	000	254	SKIP	Modules 015, 016, and 017 support the T3541 (480 character screen) keystation with keypunch, adding machine, and typewriter keyboards. Modules 024, 037, and 041 support the T3555 (2000 character screen) keystation with keypunch, adding machine, and typewriter keyboards. Modules 018, 019, and 020 support ASCII keyboards for the T3541. Modules 021, 022, and 023 support ASCII keyboards for the T3555 keystation. Version 11.R1.00 is delivered only with module 024 with the load status of LOAD. The example shows changing the load status of modules 037 and 041 from SKIP to LOAD.
041	XL TYPWRT	11.14	000	254	LOAD	
042	GERMANTYP	11. 4	000	254	SKIP	
043	DANW TYP	11.15	000	254	SKIP	
044	XL THATTY	11. 6	000	254	SKIP	
045	UK TYPWRT	11. 4	000	254	SKIP	
046	SPAIN TYP	11. 6	000	254	SKIP	
047	RUNTIME	11.63	000	000	LOAD	
048	SUB BATCH	11.21	004	000	LOAD	
049	MAP CTRL	11.27	000	000	LOAD	

Display					Operator Action		
050	EXECUTIVE	11.62	000	000	LOAD	Module 056 LCD DRVR is required when the disk drives in the system configuration are F3285 120M byte drives.	
051	FILE CTRL	11.79	000	000	LOAD		
052	LOGICALIO	11.43	000	000	LOAD		
053	CRT FMTR	11.64	000	000	LOAD		
054	IDXFILES	11.52	000	000	LOAD		
055	PHY IO 3	11.90	000	000	LOAD		
056	LCD DRVR	11.11	000	000	LOAD		
057	D34 DRVR	11.32	000	000	LOAD		
Y							Module 057 LCD DRVR is required when the disk drives in the system configuration are F3282 8.8M byte drives or F3283 17.6M byte drives. Module 057 should have a load status of LOAD. The example shows reversing the load status of LCD DRVR. The system configuration includes either 8.8M or 17.6M byte disk drives.
057	D34 DRVR	11.32	000	000	SKIP		
058	SYS SUBRS	11.15	000	000	LOAD		
T						Type T, then press the FIELD REL key after the load status of module 056 or 057 has been changed.	

Display

Operator Action

9. STD EQUIP AND PARAMETERS? REL=YES, N=NO

Type N, then press the FIELD REL key to review and change, when necessary, the system configuration, I/O assignments, and parameters (password, field separator character, etc).

Pressing the FIELD REL key allows only specification of disk type (step 10). The system configuration, I/O assignments and parameters cannot be changed.

10. DISK TYPE? REL OR D=D34 I=ISS.

Press the FIELD REL key or type D and press the FIELD REL key to specify F3282 8.8M byte or F3283 17.6M byte drives.

Type I, then press the FIELD REL key to specify F3285 120M byte drives.

If response to step 9 was FIELD REL, goes to step 129.
If response to step 9 was N(FIELD REL), goes to step 11.

Display

11. **TABLES:** REL=STANDARD 1=MINIMUM
 2=DATA COM 3=CURRENT
SELECT TABLE=

*system
tables*

Operator Action

Press the FIELD REL key; or type 1, then press the FIELD REL key; or type 2, then press the FIELD REL key; or type 3, then press the FIELD REL key. Standard configuration (pressing the FIELD REL key) is one 9-track magnetic tape, one disk drive and one line printer.

(1--Minimum configuration is one 9-track magnetic tape and one disk drive.)

2--Data Communications configuration is one 9-track magnetic tape, one disk drive, one line printer, and data communications using 2780 protocol.

3--Current configuration should be used during subsequent reviews otherwise the configuration that has been specified reverts to values in the selected table.)

Display

Operator Action

12. ACCEPT CONFIGURATION? REL=YES, N=NO

Pressing the FIELD REL key accepts the configuration based on the table in step 11. The next step is 85. Typing N and pressing the FIELD REL key allows review and modification of the configurations. The next step is 13.

13. PARAMETER DEFINITIONS ARE:

- CMND = COMMAND (A,D,R, OR H)
- LUN = LOGICAL UNIT NUMBER (1,16)
- CONT = CONTROLLER NUMBER (0,3)
- DEV = DEVICE NUMBER (1,4)
- SPL = DEVICE ALLOW SPOOLING (Y,N)
- T = NUMBER OF TRACKS (7 OR 9)
- FMTR = FORMATTER ADDRESS (0 OR 1)
- DEN = TAPE DENSITY (HIGH OR LOW)
- AUMNT = WARM AUTOMOUNT (Y OR N)

Use the CNTRL key to step through the parameter definitions and commands or DISPLAY key to go immediately to the next prompt for operator action.

- PRTY = TAPE PARITY (ODD OR EVEN)
- MDL = MODEL (TT OR DCT) OR
= MODEL (2770, 2780, 3780, HASP, U100)

Press the FIELD REL key to go to step 14.

COMMANDS ARE:

- A = ADD NEXT DEVICE
- D = DELETE EXISTING DEVICE
- R = REPLACE EXISTING DEVICE
- H = HELP, DISPLAY DEFINITIONS

PRESS REL KEY TO CONTINUE

		Display							Operator Action
14.	TYPE	LUN	CONT	DEV	SPL	TRKS	DEN	PRTY	Determine the number of tape drives in the system configuration. Typing R,1,0,1,7,H,0, then pressing the FIELD REL key replaces the configured 9-track tape with a 7-track tape. Typing A,2,0,2,9, then pressing the FIELD REL key adds a second 9-track tape drive. Typing A,2,0,2,7,H,0, then pressing the FIELD REL key adds a second 7-track tape drive. Maximum number of tape drives is four. Press the FIELD REL key after all tape drives have been specified.
	MT	01	00	01	NO	9T	00		
	9T FORM = CMND,LUN,CONT,DEV,9T,FMTR								
	7T FORM = CMND,LUN,CONT,DEV,7T,DEN,PAR								
	TYPE : INPUT OR REL FOR NEXT DEV TYPE								
	A,2,0,2,9								
	TYPE	LUN	CONT	DEV	SPL	TRKS	DEN	PRTY	
	MT	01	00	01	NO	9T	00		
	MT	02	00	01	NO	9T	00		
	9T FORM = CMND,LUN,CONT,DEV,9T,FMTR								
	7T FORM = CMND,LUN,CONT,DEV,7T,DEN,PAR								
	TYPE : INPUT OR REL FOR NEXT DEV TAPE								

Display

15. TYPE LUN CONT DEV AUMNT
DK 01 00 01 NO

DK FORM = CMND,LUN,CONT,DEV,AUMNT
TYPE : INPUT OR REL FOR NEXT DEV TYPE

key →

R,1,0,1,Y
TYPE LUN CONT DEV AUMNT
DK 01 00 01 YES

DK FORM = CMND,LUN,CONT,DEV,AUMNT
TYPE : INPUT OR REL FOR NEXT DEV TAPE

Operator Action

Determines the number of disk drives in the system configuration. Disk drive type was specified in steps 8 and 10. Only entire drives are now added to the configuration. Individual platters are no longer specified. Typing R,1,0,1, Y, then pressing the FIELD REL key provides for automounting of platters in the first disk drive. Automount from a LOAD of the operating system initializes all platters. Automount from restart recovers existing data on all platters. Typing A,2,0,2,Y, then pressing the FIELD REL key, adds a second disk drive. Maximum number of F3282 8.8M byte or F3283 17.6M byte drives is four. Maximum number of F3285 120M byte drives is four. Press the FIELD REL key after all drives have been specified.

	Display					Operator Action	
16.	TYPE	LUN	CONT	DEV	SPL	Press the FIELD REL key.	
	KS	01	00	01	YES		
	KS FORM = NO INPUT REQUIRED TYPE : INPUT OR REL FOR NEXT DEV TYPE						
17.	TYPE	LUN	CONT	DEV	SPL	MDL	This step is skipped. It appears momentarily on the screen, however.
	LD	(NONE)			
	LD FORM = CMND,LUN,CONT,DEV,SPL,MDL						
18.	TYPE	LUN	CONT	DEV	SPL		If the system configuration selected by table in step 11 did not include a line printer, the line printer may be added by typing A,l,0,l,Y, and pressing the FIELD REL key.
	LP	01	00	01	YES		
	LP FORM = CMND,LUN,CONT,DEV,SPL TYPE : INPUT OR REL FOR NEXT DEV TYPE						
19.	TYPE	LUN	CONT	DEV	SPL		Press the FIELD REL key.
	CR	(NONE)			
	CR FORM = CMND,LUN,CONT,DEV TYPE : INPUT OR REL FOR NEXT DEV TYPE						
20.	TYPE	LUN	CONT	DEV	SPL	MDL	Up to four data communications protocols may be assigned to the four logical communication controllers C1-C4.
	DC	(NONE)			
	DC FORM = CMND,LUN,CONT,DEV,MDL TYPE : INPUT OR REL FOR NEXT DEV TYPE						

Display

Operator Action

*Select:
Set 1, 2, 3*

Typing A,n,0,
n,2770, then
pressing the FIELD
REL key adds IBM
2770 protocol where
n=1-4 and specifies
the logical
controller C1-C4.

Typing A,¹n,0,¹n,2780,
then pressing the
FIELD REL key adds
2780.

Typing A,n,0,n,3770,
then pressing the
FIELD REL key adds
3770.

Typing A,¹n,0,²n,3780,
then pressing the
FIELD REL key adds
3780.

Typing A,n,0,n,HASP,
then pressing the
FIELD REL key adds
HASP.

Typing A,³n,0,³n,U100,
then pressing the
FIELD REL key adds
U100.

Typing A,n,0,n,HI,
then pressing the
FIELD REL key adds
Host Interactivity.

After adding a
protocol, goes to
step 21.

(list of all items added appears here)

After adding all
protocols, press
the FIELD REL key
to go to step 72.
If no protocols are
added, goes to step
83.

Display

Operator Action

21. **DEFAULT DATA COM CONTROLS? REL=YES, N=NO**

Pressing the FIELD REL key accepts the data communications defaults and returns to step 20. Typing N, then pressing the FIELD REL key allows specification of options by protocol; however, these options may be subsequently changed by the Supervisor SET command. If N(FIELD REL) is entered, and 2770 was added, goes to step 22; 2780 goes to step 34; 3770 goes to step 45; 3780 goes to step 51; HASP goes to step 60; U100 goes to step 65; HI (Host Interactivity) goes to step 68.

22. **DISABLE 9600 BAUD CLOCK? REL=YES, N=NO**

(Start of 2770 Options.) Press the FIELD REL key.

23. **DATA SET RATE HIGH? REL=HIGH, L=LOW**

Specifies the switch position on modems with a high/low switch.

Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.

Display	Operator Action
24. SUPPRESS PRINT CONTROL? REL=NO, Y=YES	Pressing the FIELD REL key does not suppress the vertical forms control characters. Type Y, then press the FIELD REL key for vertical forms control characters to be treated as data during receive.
25. MULTI-POINT? REL=NO, Y=YES	Press the FIELD REL key and go step 27. Typing Y, then pressing the FIELD REL key, may be used in a multipoint communications network as a slave station. Go to step 26 to enter the alphabetic polling address.
26. POLLING ADDRESS (ALPHA CHARACTER)?	Type a, then press the FIELD REL key, where a is the one character alphabetic polling address.
27. BREAK UP LONG RECORDS? REL=NO, Y=YES	Press the FIELD REL key for records to be communicated using the disk record length. Type Y, then press the FIELD REL key and records exceeding 80 characters in length will be broken into multiple 80-character words.

Display	Operator Action
28. BUFFER EXPANSION? REL=NO, Y=YES	Pressing the FIELD REL key allocates a 128-byte buffer. Goes to step 30. Typing Y, then pressing the FIELD REL key allocates a 256-byte buffer. Goes to step 29.
29. ADDITIONAL EXPANSION? REL=NO, Y=YES	Pressing the FIELD REL key allocates the 256-byte buffer. Typing Y, then pressing the FIELD REL key allocates a 512-byte buffer.
30. ASCII TRANSLATION? REL=NO, Y=YES	Press the FIELD REL key for no ASCII translation. Type Y, then press the FIELD REL key to support ASCII code to provide compatibility with the device being communicated with. Cannot be used concurrently with EBCDIC transparency.
31. EBCDIC TRANSPARENCY? REL=NO, Y=YES	Press FIELD REL for no EBCDIC transparency. Typing Y, then pressing the FIELD REL key allows communications of data that contains hex characters corresponding to protocol characters, i.e., characters less than hex 40.

Display

Operator Action

32. SPACE COMPRESSION ON? REL=NO, Y=YES

Press the FIELD REL key for no space compression. Type Y, then press the FIELD REL key to compress spaces to increase communications throughput. Cannot be used with EBCDIC transparency or Delete Trailing Blanks.

33. DELETE TRAILING BLANKS? REL=NO, Y=YES

Pressing the FIELD REL key shortens transmission time required for most data by deleting spaces at the end of a record. Cannot be used with EBCDIC transparency or space compression. Go to step 20.

Typing N, then pressing the FIELD REL key returns to step 20.

34. DISABLE 9600 BAUD CLOCK? REL=YES, N=NO

(Start of 2780 options) Press the FIELD REL key.

35. DATA SET RATE HIGH? REL=HIGH, L=LOW

Specifies the switch position on modems with a high/low switch. Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.

Display	Operator Action
36. SUPPRESS PRINT CONTROL? REL=NO, Y=YES	Pressing the FIELD REL key does not suppress vertical forms control characters. Type Y, then press the FIELD REL key for vertical forms control characters to be treated as data during receive.
37. MULTI-POINT? REL=NO, Y=YES	Press the FIELD REL key and go to step 39. Typing Y, then pressing the FIELD REL key allows the system to be used in a multipoint communications network as a slave station. Go to step 38 to enter the alphabetic polling address.
38. POLLING ADDRESS (ALPHA CHARACTER)?	Type a, then press the FIELD REL key (where a is the one character alphabetic polling address).
39. BREAK UP LONG RECORDS REL=NO, Y=YES	Press the FIELD REL key for records to be communicated using the disk record length. Type Y, then press the FIELD REL key for records exceeding 80 characters in length to be broken into multiple 80-character records.

Display

Operator Action

40. BUFFER EXPANSION? REL=NO, Y=Yes

Pressing the FIELD REL key allocates a 400-byte buffer.

Typing Y, then pressing the FIELD REL key, allocates a 500-byte buffer.

41. ASCII TRANSLATION? REL=NO, Y=YES

Press the FIELD REL key for no ASCII translation.

Typing Y, then pressing the FIELD REL key, supports ASCII code to provide compatibility with the device being communicated with. Cannot be used concurrently with EBCDIC transparency.

42. EBCDIC TRANSPARENCY? REL=NO, Y=YES

Press the FIELD REL key for no EBCDIC transparency.

Typing Y, then pressing the FIELD REL key, allows communications of data that contains hex characters corresponding to protocol characters, i.e., characters less than hex 40.

43. MULTIRECORD? REL=NO, Y=YES

Pressing the FIELD REL key allows a maximum of four records per block to be transmitted. Typing Y, then pressing the FIELD REL key, allows up to seven records per block to be transmitted.

Display	Operator Action
44. DELETE TRAILING BLANKS? REL=YES, N=NO	Pressing the FIELD REL key shortens transmission time for most data by deleting spaces at the end of a record. Cannot be used with EBCDIC transparency. Returns to step 20. Type N, then press the FIELD REL key to return to step 20.
45. DISABLE 9600 BAUD CLOCK? REL=YES, N=NO	(Start of 3770 Options). Press the FIELD REL key.
46. DATA SET RATE HIGH? REL=HIGH, L=LOW	Specifies the switch position on modems with a high/low switch. Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.
47. SUPPRESS PRINT CONTROL? REL=NO, Y=YES	Pressing the FIELD REL key does not suppress the vertical forms control characters. Type Y, then press the FIELD REL key for vertical forms control characters to be treated as data during receive.

Display

Operator Action

48. **TERMINAL ID (2 HEX CHARS)=F1** Type tt, then press the FIELD REL key, where tt is used as the SDLC address and also as the ID for the SDLC XID command. Acceptable hexadecimal values are C1 through C6 and F0 through F9.
49. **BUFFER EXPANSION? REL=NO, Y=YES** Pressing the FIELD REL key leaves the buffer size at 256 bytes. Typing Y, then pressing the FIELD REL key expands the buffer size to 512 bytes.
50. **EBCDIC TRANSPARENCY? REL=NO, Y=YES** Press the FIELD REL key for no EBCDIC transparency. Typing Y, then pressing the FIELD REL key allows communications of data that contains hex characters corresponding to protocol characters, i.e., characters less than hex 40.
51. **DISABLE 9600 BAUD CLOCK? REL=YES, N=NO** (Start of 3780 Options). Press the FIELD REL key.
52. **DATA SET RATE HIGH? REL=HIGH, L=LOW** Specifies the switch position on modems with a high/low switch.
- Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.

Display

Operator Action

53. SUPPRESS PRINT CONTROL? REL=NO, Y=YES

Pressing the FIELD REL key does not suppress the vertical forms control characters. Type Y, then press the FIELD REL key for vertical forms control characters to be treated as data during receive.

54. MULTI-POINT? REL=NO, Y=YES

Press the FIELD REL key to return to step 56. Typing Y, then pressing the FIELD REL key allows the system to be used in a multipoint communications network as a slave station. Go to step 55 to enter the alphabetic polling address.

55. POLLING ADDRESS (ALPHA CHARACTER)?

Type a, then press the FIELD REL key, where a is the alphabetic polling address (1 character).

56. BREAK UP LONG RECORDS REL=NO, Y=YES

Press the FIELD REL key for records to be communicated using the disk record length. Typing Y, then pressing the FIELD REL key, allows records exceeding 80 characters in length to be broken into multiple 80-character records.

Display

Operator Action

57. ASCII TRANSLATION? REL=NO, Y=YES

Press the FIELD REL key for no ASCII translation. Typing Y, then pressing the FIELD REL key supports ASCII code to provide compatibility with the device being communicated with. Cannot be used concurrently with EBCDIC transparency.

58. EBCDIC TRANSPARENCY? REL=NO, Y=YES

Press the FIELD REL key for no EBCDIC transparency. Typing Y, then pressing the FIELD REL key allows communications of data that contains hex characters corresponding to protocol characters, i.e., characters less than hex 40.

59. SPACE COMPRESSION ON? REL=NO, Y=YES

Pressing the FIELD REL key allows no space compression. Returns to step 20.

Typing Y, then pressing the FIELD REL key compresses spaces to increase communications throughput. Cannot be used with EBCDIC Transparency. Returns to step 20.

60. DISABLE 9600 BAUD CLOCK? REL=YES, N=NO

(Start of HASP options). Press the FIELD REL key.

Display

Operator Action

61. DATA SET RATE HIGH? REL=HIGH, L=LOW

Specifies the switch position on modems with a high/low switch.

Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.

62. SUPPRESS PRINT CONTROL? REL=NO, Y=YES

Pressing the FIELD REL key does not suppress the vertical forms control characters. Type Y, then press the FIELD REL key for vertical forms control characters to be treated as data during receive.

63. BUFFER EXPANSION? REL=NO, Y=YES

Pressing the FIELD REL key allocates a 400-byte buffer.

Typing Y, then pressing the FIELD REL key allocates a 512-byte buffer.

64. EBCDIC TRANSPARENCY? REL=NO, Y=YES

Press the FIELD REL key for no EBCDIC transparency. Returns to step 20.

Typing Y, then pressing the FIELD REL key allows communications of data that contains hex characters corresponding to protocol characters; i.e., characters less than hex 40. Returns to step 20.

Display	Operator Action
65. DISABLE 9600 BAUD CLOCK? REL=YES, N=NO	(Start of U100 Options). Press the FIELD REL key.
66. DATA SET RATE HIGH? REL=HIGH, L=LOW	Specifies the switch position on modems with a high/low switch. Press the FIELD REL key or type L, then press the FIELD REL key as appropriate.
67. BUFFER EXPANSION? REL=NO, Y=YES	Pressing the FIELD REL key allocates a 1,050-byte buffer (for U100 screen). Returns to step 20. Typing Y, then pressing the FIELD REL key allocates a 2,100-byte buffer (for U200 screen). Returns to step 20.
68. DISABLE 9600 BAUD CLOCK? REL=YES, N=NO	(Start of Host Interactivity Options). Press the FIELD REL key.
69. DATA SET RATE HIGH? REL=HIGH, L=LOW	Specifies the switch position on modems with a high/low switch. Press the FIELD REL key or type L, then press the FIELD REL key, as appropriate.

Display

70. CNTRL UNIT ADR(2 HEX CHARS)=C1

Operator Action

Enter the control unit address that has been assigned to the system and matches the expectation of the host computer. Entered as two hexadecimal digits based on the following table:

<u>Control Unit Address Decimal</u>	<u>Control Unit Address Hex</u>
0	40
1	C1
2	C2
3	C3
4	C4
5	C5
6	C6
7	C7
8	C8
9	C9
10	4A
11	4B
12	4C
13	4D
14	4E
15	4F
16	50
17	D1
18	D2
19	D3
20	D4
21	D5
22	D6
23	D7
24	D8
25	D9
26	5A
27	5B
28	5C
29	5D
30	5E
31	5F

Display

Operator Action

71. **HOST INT TIMEOUT(0,5)=0**

The maximum number of minutes allowed for a format program to await completion of a host interactivity RECEIVE or SEND statement. 0 results in no automatic timeout, regardless of how long a format program waits. If 1 to 5 is entered, any SEND or RECEIVE request causing a format program to wait the specific number of minutes is automatically cancelled.

72. **NUMBER OF COMM BUFS = 2**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the maximum number of buffers for data communications. Increasing the number of buffers may increase communications throughput depending on other activity on the system during communications. Buffer size is determined by protocol and by buffer expansion.

Press the FIELD REL key for no change.

Type n, then press the FIELD REL key to change the number of buffers to n.

Display

Operator Action

73. **MAX NO XMT BUFS = 2**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

The number of transmit buffers should be the same as the total number of communications buffers.

Press the FIELD REL key for no change.

Type n, then press the FIELD REL key to change the number of buffers to n.

74. **MAX NO RCV BUFS = 2**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

The number of receive buffers should be the same as total communications buffers.

Press the FIELD REL key for no change.

Type n, then press the FIELD REL key to change the number of buffers to n.

If a data communication was added, but not U100, go to step 75. If U100 was added as a protocol, go to step 76.

75. **ALLOW U100? (Y,N) = Y**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Type N, then press the FIELD REL key to go to step 83. Press the FIELD REL key or type Y, then press the FIELD REL key to go to step 76.

Display

Operator Action

76. SET U100 RECORD SIZE? Y OR N

Type Y, then press the FIELD REL key to change record sizes assigned to logical data communications output devices regardless of the protocols specified and to specify U100 remote ID (RID) and site ID (SID).

Go to step 77.

Type N, then press the FIELD REL key or press the FIELD REL key to go to step 81.

77. D1 64
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the logical record length assigned to data communications logical output device D1. May be changed by the Supervisor SET command.

Press the FIELD REL key for no change.

Type nn, then press the FIELD REL key to change record length for D1 to nn.

Display

Operator Action

78. D2 80
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the logical record length assigned to data communications logical output device D2. May be changed by the Supervisor SET command.

Press the FIELD REL key for no change.

Type nnn, then press the FIELD REL key to change record length for D2 to nnn.

79. D3 132
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the logical record length assigned to data communications logical output device D3. May be changed by the Supervisor SET command.

Press the FIELD REL key for no change.

Type nnn, then press the FIELD REL key to change record length for D3 to nnn.

Display

Operator Action

80. D4 1050
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the logical record length assigned to data communications logical output device D4. May be changed by the Supervisor SET command.

Press the FIELD REL key for no change.

Type nnn, then press the FIELD REL key to change record length for D4 to nnn.

81. RID ADDRESS (41,117)
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Remote Identification (RID) identifies the communications routing (line) to the host processor when using U100 protocol. Determined by generation of the operating system for the host processor. May be changed by the Supervisor SET command. Type nnn, then press the FIELD REL key to assign the RID address.

Display

Operator Action

82. **SID ADDRESS (121,157)**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Site Identification (SID) identifies the workstation to the host processor when using U100 protocol. Determined by generation of the operating system for the host processor. May be changed by the Supervisor SET command. Type nnn, then press the FIELD REL key to assign the SID address.

83. **TYPE LUN CONT DEV SPL**
FD (NONE)
FD FORM = CMND,LUN,CONT,DEV
TYPE: INPUT OR REL FOR NEXT DEV TYPE

Up to two flexible diskette drives may be configured on XL/40 systems. Flexible diskette configuration is not meaningful on 1900/10 systems. Typing A,n,0,n, then pressing the FIELD REL key adds diskette drive #n to the system where n=1 or 2.

After adding the desired diskette drives, press the FIELD REL key and go to step 84.

Display

Operator Action

84. CURRENT CONFIGURATION IS:

TYPE	LUN	CONF	DEV	SPL	TRKS	DEN	PRTY
MT	01	00	01	NO	9T	00	
MT	02	00	02	NO	9T	00	

Use CNTRL key to step through the current configuration. Press the DISPLAY key to go immediately to the prompt for operator action.

TYPE	LUN	CONF	DEV	AUMNT
DK	01	00	01	YES

Press the FIELD REL key to go to step 85.

TYPE	LUN	CONF	DEV	SPL
KS	01	00	01	YES

Type N, then press the FIELD REL key to return to step 13.

TYPE	LUN	CONF	DEV	SPL	MDL
LD		(NONE)			

TYPE	LUN	CONF	DEV	SPL
LP	01	00	01	YES

TYPE	LUN	CONF	DEV	SPL
CR		(NONE)		

TYPE	LUN	CONF	DEV	SPL	MDL
DC	01	00	01	NO	HI

TYPE	LUN	CONF	DEV	SPL
FD		(NONE)		

ACCEPT CONFIGURATION? REL=YES, N=NO

85. CURRENT I/O ASSIGNMENTS ARE:

LOG DEV	(TYPE,LUN,SPL) =	KS, 01, N
INPUT DEV	(TYPE,LUN) =	MT, 01
01 OUTPUT	(TYPE,LUN,SPL) =	MT, 01, N
02 OUTPUT	(TYPE,LUN,SPL) =	MT, 01, N
D1 DCOM1	(TYPE,LUN,SPL) =	DK, 01, N
D2 DCOM2	(TYPE,LUN,SPL) =	DK, 01, N
D3 DCOM3	(TYPE,LUN,SPL) =	DK, 01, N
D4 DCOM4	(TYPE,LUN,SPL) =	DK, 01, N
ACCEPT ASSIGNMENTS?	REL=YES, N=NO	

Press the FIELD REL key to accept the I/O assignments as shown. These can be changed by the Supervisor SET command during operations. (Forwards to step 94.)

Type N, then press the FIELD REL key to go to step 86.

Display	Operator Action
86. LOG DEV (TYPE,LUN,SPL) = KS, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT LP,1,Y LOG DEV (TYPE,LUN,SPL) = LP, 01, Y TYPE NEW ASSIGNMENT OR REL=ACCEPT	Identifies the device that is to receive output from Autolog if Log Options (step 104) are selected. Devices are Line Printer (LP) or Magnetic Tape (MT). The example shows change of assignment to line printer with spooling.
NOTE:	
Spooling must also have been specified when the line printer was added/included in the configuration at step 18.	
87. INPUT DEV (TYPE,LUN) = MT, 01 TYPE NEW ASSIGNMENT OR REL=ACCEPT	Change or press the FIELD REL key.
88. 01 OUTPUT (TYPE,LUN,SPL) = MT, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Change or press the FIELD REL key.
89. 02 OUTPUT (TYPE,LUN,SPL) = MT, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key.
NOTE:	
02 Output not currently used.	
90. D1 DCOM1 (TYPE,LUN,SPL) = DK, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Change or press the FIELD REL key.
91. D2 DCOM2 (TYPE,LUN,SPL) = DK, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Change or press the FIELD REL key.
92. D3 DCOM3 (TYPE,LUN,SPL) = DK, 01, N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Change or press the FIELD REL key.

Display

Operator Action

93. D4 DCOM4 (TYPE,LUN,SPL) = DK, 01, N
TYPE NEW ASSIGNMENT OR REL=ACCEPT
94. PASSWORD (8 CHARS) =
TYPE NEW ASSIGNMENT OR REL=ACCEPT
95. FIELD SEPARATOR CHAR =
TYPE NEW ASSIGNMENT OR REL=ACCEPT
96. COMMAND ID COUNT(2,6) = 02
TYPE NEW ASSIGNMENT OR REL=ACCEPT
- Change or press the FIELD REL key.
Return to step 85.
- Press the FIELD REL key for no change.
Enter up to eight alphanumeric characters to form the desired password, then press the FIELD REL key.
- Press the FIELD REL key for no change (if blank, separation character is a space).
- Type N, then press the FIELD REL key for no separation between fields.
- Type x, then enter the FIELD REL key for where x is any character except N.
- Specifies the number of characters required to identify a command (the operating system completes the remaining characters in the command).
- Press the FIELD REL key for no change.
- Type n, then press the FIELD REL key, where n is a number between 2 and 6.

Display

Operator Action

97. **MAX COAX ADDR (2,63) = 08**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Specifies the maximum number of devices, such as keystations or keystation printers.

Press the FIELD REL key for no change.

Type nn, then press the FIELD REL key, where nn is a number between 2 and 63.

98. **RESPONSES ARE:**
P = PARTIAL
C = COMPLETE
A = ALL
Y = YES
N = NO
TYPE = MT
LUN = 1-4

These responses apply to the prompts generated by typing Y, then pressing the FIELD REL key as a response to Auto Save (step 108).

99. **OPS-STATS (Y,N) =N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Press the FIELD REL key for no change.

By typing Y, then pressing the FIELD REL key, the operator statistics batch will open automatically during startup.

By typing N, then pressing the FIELD REL key, the operator statistics batch will not open automatically during startup; however, the Supervisor may open operator statistics with the OP-STAT command.

Display

Operator Action

100. TONE A RCD END (Y,N) = Y
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Press the FIELD REL key for no change.

Type Y, then press the FIELD REL key and the keystation in an operator mode will sound the ready tone at the end of a record.

Type N, then press the FIELD REL key, and the tone does not sound at the record end.

101. OPTIONAL FEATURES (Y,N) =N
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Press the FIELD REL key for no change.

Typing Y, then pressing the FIELD REL key causes the MISC I/O and WORK INIT overlays to become resident in memory. Also, the 20 most recently used system messages are retained in memory. System response time is improved, particularly when a batch is opened; however, more memory is required.

Display

102. **FAST REPLAY(Y,N) =N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Operator Action

Press the FIELD REL key for no change.

Typing Y, then pressing the FIELD REL key causes three copies of format program registers (A,B,C,D) to be allocated in each operator Terminal Work Area (TWA) for storing the register contents of the previous record and previous checkpoint. Changes to registers in a record can be based on the prior record, checkpoint in the TWA, or normal checkpoint, substantially reducing the incidence of long replays.

Type N, then press the FIELD REL key for no fast replay or double checkpoint.

Display	Operator Action
103. DBL BUF MT WRITE (Y,N) =Y TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. By typing Y, then pressing the FIELD REL key, the system will use double buffers for tape write whenever adequate memory is available. The system will revert to a single buffer if adequate memory is not available. Type N, then press the FIELD REL key for a single buffer tape write to always be used.
104. LOG OPTIONS (Y,N) =N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. Type Y, then press the FIELD REL key to select log options. Goes to step 105. Typing N, then pressing the FIELD REL key does not select log options. Goes to step 108.

Display	Operator Action
105. LOG SV ERRORS (Y,N) =N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. Type Y, then press the FIELD REL key to log Supervisor messages and errors automatically to the device specified in step 86 when the device is not KS, 01. Typing N, then pressing the FIELD REL key does not log Supervisor messages and errors.
106. LOG BK MESSAGES (Y,N) =N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. Typing Y, then pressing the FIELD REL key logs background messages automatically to the device specified in step 86 when the device is not KS, 01. Typing N, then pressing the FIELD REL key does not log background messages.

Display

Operator Action

107. LOG AUTO MESSAGES (Y,N) =N
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Press the FIELD REL
key for no change.

Typing Y, then
pressing the FIELD
REL key initiates
Autolog
automatically at
startup to the
device specified in
step 86 when the
device is not KS,
01. Log entries
will be Supervisor
prompts and
responses and batch
opens and closes.

Typing N, then
pressing the FIELD
REL key does not
initiate Autolog
automatically.

Display

108. **AUTO SAVE (Y,N) =N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Operator Action

Press the FIELD REL key for no change.

Typing Y, then pressing the FIELD REL key automatically saves batches of data to a magnetic tape at batch close. Autosave is initiated at startup. It can be terminated or begun by the Supervisor. Steps 109 through 116 define the conditions that cause a batch to be saved automatically. Step 117 specifies the tape drive. Go to step 109.

Typing N, then pressing the FIELD REL key does not automatically save data batches. Goes to step 118.

Display	Operator Action
109. STD CNTRL WORD (Y,N) =Y TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. Typing Y, then pressing the FIELD REL key saves all entered, verified, updated, modified, written, balanced, and valid jobs. Goes to step 117.
110. ENTERED JOBS (P,C,A,N) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Typing N, then pressing the FIELD REL key selectively specifies the conditions that cause a batch to be saved automatically. Goes to step 110. Responses that apply to steps 110 through 116 are:
111. VERIFIED JOBS (P,C,A,N) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change.
112. UPDATED JOBS (P,C,A,N) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Type P, then press the FIELD REL key for partial (closed with CMND/I).
113. MODIFIED JOBS (Y,N,A) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Type C, then press the FIELD REL key for complete (closed with CMND/I or Level Z).
115. BALANCED JOBS (Y,N,A) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Type A, then press the FIELD REL key for all.
116. VALID JOBS (Y,N,A) =A TYPE NEW ASSIGNMENT OR REL=ACCEPT	Type Y, then press the FIELD REL key for yes.

Display

Operator Action

Type N, then press the FIELD REL key for no.

For example:

110. C
111. C
112. N
113. N
114. Y
115. Y
116. N

would cause a data batch to be automatically saved only when it was completely entered and verified, was in balance, and after it had been written to magnetic tape.

117. **SAVE DEVICE (TYPE,LUN) = MT,01
TYPE NEW ASSIGNMENT OR REL=ACCEPT**

Press the FIELD REL key for no change.

Type MT, n, then press the FIELD REL key, where n is 1 through 4 specifying the logical tape unit number.

Display

Operator Action

118. **BK TWA POOL CNTRL (2,8) = 02**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Controls the number of background activities that can take place concurrently. Activities that are executed in the background are:

Autosave
Autolog
Data communications
Spooling
Sort
Index file
Creation
Save
Restore

Press the FIELD REL key for no change.

Type n, then press the FIELD REL key where n is 2 through 8 specifying the number of terminal work areas in the background pool.

119. **LOAD FILE MGMT AT STARTUP (Y,N) =N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Type Y, then press the FIELD REL key to load IND RES, the File Management module, at startup. The size of IND RES is approximately 12K bytes.

Typing N, then pressing the FIELD REL key does not load IND RES at startup.

Display	Operator Action
120. CORRECTION COUNT BY RECORD (Y,N) =Y TYPE NEW ASSIGNMENT OR REL=ACCEPT	Press the FIELD REL key for no change. Type Y, then press the FIELD REL key for the correction count in operator statistics for the verify mode to be the number of records corrected. Type N, then press the FIELD REL key for the correction count in operator statistics for the verify mode to be the number of characters corrected.
121. THREE CYCLE VERIFY (Y,N) =N TYPE NEW ASSIGNMENT OR REL=ACCEPT	Typing Y, then pressing the FIELD REL key requires two entries of a character after a verify miscompare to correct the character (i.e., two verify miscompare messages). Typing N, then pressing the FIELD REL key requires one entry of a character after a verify miscompare to correct the character.

Display

Operator Action

122. **DISK DATA CACHE (Y,N) = N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Pressing the FIELD REL key or typing N, then pressing the FIELD REL key, completely disables the Disk Data Cache feature. Goes to step 126.

Typing Y, then pressing the FIELD REL key enables caching of disk I/O.

Select this option if Disk Data Cache is to be used at any time with this system. The Disk Data Cache feature may then be loaded or unloaded at will using the Supervisor Disk MGMT command.

123. **ENABLE WRITE ON DATA CACHE (Y,N) = N**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Typing Y, then pressing the FIELD REL key enables caching of all disk writes, giving a throughput advantage.

Pressing the FIELD REL key or typing N, then pressing the FIELD REL key forces normal physical I/O processing of disk writes while still caching disk reads. Cached writes may be lost in a system failure which presents an undesirable risk in some applications.

Display

Operator Action

124. **LOAD DATA CACHE AT STARTUP (Y,N) = Y**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

By typing Y, then pressing the FIELD REL key or by simply pressing the FIELD REL key, the Disk Data Cache feature will be automatically loaded and active at startup time.

Typing N, then pressing the FIELD REL key requires that the Disk Data Cache be manually loaded after system startup if it is to be used.

125. **NUMBER OF DATA CACHE BUFFERS (1,15) = 04**
TYPE NEW ASSIGNMENT OR REL=ACCEPT

Press the FIELD REL key to select from cache buffers.

Type N, then press the FIELD REL key to select n cache buffers. The optimum number of cache buffers will vary with different applications. Generally, one buffer per Supervisor function will relieve disk contention although, in some cases, fewer buffers may be just as effective. Experiment to determine the optimum number of buffers to select.

Display	Operator Action
126. SUB-BATCH BALANCE (Y,N) = N TYPE NEW ASSIGNMENT OR REL=ACCEPT	By pressing the FIELD REL key or typing N, then pressing the FIELD REL key, subbatch balancing procedures in the format programs will be ignored. By typing Y, then pressing the FIELD REL key, memory will be allocated for subbatch balancing and subbatch balancing procedures in format programs will be executed.
127. ACCEPT ALL DEFINITIONS? Y=YES, N=NO	Type Y, then press the FIELD REL key to continue with system generation. Goes to step 129. Type N, then press the FIELD REL key to review the configuration, I/O assignments and definitions. Goes to step 128 first.
128. ACCEPT CONFIGURATION? REL=YES, N=NO	Press the FIELD REL key if configuration is all right. Start at I/O assignments (step 85). Type N, then press the FIELD REL key to return to step 10, (DISK TYPE?).

Display

Operator Action

129. **LOAD SELECTED MODULES TO DISK**

NO	NAME	VERSN	TYP	ID	STATUS
001	S BLD LC	11.72	014	001	LOAD
002	S GEN LC	11.59	014	001	LOAD
003	S BLD MSG	11.13	014	002	LOAD
004	S LD TBLS	11. 6	014	003	LOAD
005	S P BUFL	11.18	014	001	LOAD
006	ATTRIBITS	11. 4	000	000	LOAD
007	ATTGERMAN	11. 2	000	000	SKIP
008	TURK ATTR	11. 5	000	000	SKIP
009	ATTRIUK	11. 2	000	000	SKIP

NO	NAME	VERSN	TYP	ID	STATUS
010	NORDICATT	11. 7	000	000	SKIP
011	ISWATTRBT	11. 3	000	000	SKIP
012	ATTIGREEK	11. 4	000	000	SKIP
013	SPAINATT	11. 2	000	000	SKIP
014	THAI ATTR	11. 4	000	000	SKIP
015	KEYPCH KT	11.11	000	254	SKIP
016	ADDMCH KT	11.11	000	254	SKIP
017	TYPE KT	11.13	000	254	SKIP
018	KP KT ASC	11.11	000	254	SKIP
019	KP AM ASC	11.13	000	254	SKIP

.
.

.

328 SYS INFO 11. 1 003 267 LOAD

130. **LOAD TABLES FROM TAPE? Y=YES, REL=NO**

Modules of the operating system are read from the SMB, patches are applied and the modules are written to disk.

Type Y, then press the FIELD REL key if user defined Datatype and/or Translation Tables (created with user library format programs DATATYPE and/or TRANSLAT) are to be added to the operating system. Dismount the SMB and mount the DATATYPE/TRANSLAT WRITE tape.

Press the FIELD REL key if user-defined tables are not to be added.

Display

Operator Action

131. **ACCEPT FILE MGMT PARMS?** **Y=**Y**, REL=NO**

Type Y, then press the FIELD REL key to accept the File Management buffer definitions. If GTS was entered at step 7, goes to step 140. If GDS was entered at step 7, goes to step 138.

Typing N, then pressing the FIELD REL key allows review and change of File Management Buffer definitions. Goes to step 132.

132. **NR OF RCD BUFS**
(2, 16)= 3

Record buffers hold records being processed for an index file. Should be set to the number of page buffers plus one.

Press the FIELD REL key for no change.

Type nn, then press the FIELD REL key, where nn is 2 through 16, to change.

Display

Operator Action

133. RCD BUF LTH IN WORDS
(2, 500)= 64

One record buffer expands to accommodate the largest record being processed for an index file. Record buffer size should be set to the second largest record size that will be processed.

Press the FIELD REL key for no change.

Type nnn, then press the FIELD REL key, where nnn is 2 through 500 record length in words.

134. NR OF KEY BUFS
(2, 16)= 3

Key buffers hold keys being used to process records for index files. Should be set to the number of page buffers plus one.

Press the FIELD REL key for no change.

Type nn, then press the FIELD REL key, where nn is 2 through 16, to change.

Display	Operator Action
135. KEY BUF LTH IN WORDS (4, 164)= 16	One key buffer expands to accomodate the largest key being used to process records for an index file. Key buffer size should be set to the second largest key size to be processed. Press the FIELD REL key for no change. Type nnn, then press the FIELD REL key, where nnn is 4 through 164 key length in words.
136. NR OF PAGE BUFS (1, 16)= 4	The number of page buffers should be equal to the number of disk controllers plus one. Press the FIELD REL key for no change. Type 2, then press the FIELD REL key for the current standard configuration.

Display

Operator Action

137. NR OF WORK BUFS
(2, 16)= 8

The number of work buffers should be set to the number of page buffers plus two.

Press the FIELD REL key for no change.

Type 4, then press the FIELD REL key for the current standard configuration. Returns to step 131.

138. TYPE SYSGEN COMMAND
CDM
CDM CORRECT DISK MODULE

(ENTER PATCH)

Continue here if GDS was entered at step 7. Type CDM, then press the FIELD REL key to apply additional patches to the operating system using the CORRECT DISK MODULE command.

139. TYPE SYSGEN COMMAND
WTS
WTS WRITE TAPE SYSTEM
CHANGE XX.XX.XX to YY.YY.YY?
TYPE VERSION NUMBER OR REL TO ACCEPT
11.R1.00
CHANGE XX.XX.XX to 11.R1.00?
TYPE VERSION NUMBER OR REL TO ACCEPT

Dismount SMB or DATATYPE/TRANSLAT write tape. Mount a scratch tape with a write ring.

Type WTS, then press the FIELD REL key.

Enter the version number that appears on the SMB.

Press the FIELD REL key if the new version number is acceptable. Operating system is written to magnetic tape. End GDS.

Display

*Dismount old tape
write new*

Operator Action

140. MOUNT TAPE: REL=READY, A=ABORT
NEW OLD SMP TAP
CHANGE XX.XX.XX to YY.YY.YY?

TYPE VERSION NUMBER OR REL TO ACCEPT

11.R1.00
CHANGE XX.XX.XX to 11.R1.00?
TYPE VERSION NUMBER OR REL TO ACCEPT

TYPE SYSGEN COMMAND

Continue here if GTS was entered at step 7. Dismount SMB or DATATYPE/TRANSLAT write tape. Mount a scratch tape with a write ring.

Press the FIELD REL key to accept the specified version or rekey new version number.

Enter the version number that appears on the SMB. Then press the FIELD REL key if the new version number is acceptable.

Operating system is written to magnetic tape. End GTS.

NOTE:

Pressing the FIELD REL key ends the GTS; however, the operating system is on the disk and may be subsequently written to magnetic tape using the Write Tape System (WTS) command. See step 139.

F. Messages

F.1. ERROR MESSAGES

If a condition occurs that causes the system to display an error message, an alarm tone sounds, a bright checkerboard pattern shows briefly on the entire screen, and a message appears on the message line of the terminal display screen. If the system prints a message, it appears on the source listing near the text that caused the message to be generated, or on the printed log of operations.

F.2. MESSAGE TYPES

Three types of message may appear:

Alarm messages describe an error condition. When an alarm message displays, the keyboard locks. Entries cannot be made until the message and the error are cleared by pressing the RESET key.

Guide messages guide operations or inform the operator of some special terminal status. When a guide message displays, the keyboard locks. Entries cannot be made until the message and the error are cleared by pressing the RESET key.

Requested information messages are related to system operation or are created by operator application programs to inform the operator of an alarm condition or program condition.

F.3. ERROR TYPES

F.3.1. Keying Errors

Keying errors are reported on the screen immediately. They may be caused by pressing an invalid data key, as determined by the format, or an invalid control key. For example, a number might have been entered in a position where the format allows only alphabetic characters. Field keying errors are reported at field end when the last character of the field is keyed, the field is rekeyed, or the field is released.

F.3.2. Field Check Errors

A field check error occurs when data is entered correctly from the

source document, but the data does not meet format requirements. Depending on format design, the system may report these errors while the user is in the ENTER mode or during data validation. The most common field check errors are as follows:

Check digit field error

Although all digits have been entered correctly, the check digit on the source document does not agree with the one computed by the system.

Value set field error

The value in a given field is found to be invalid when compared with a value specified in a system table.

Range check field error

A numeric value has been entered outside the range specified in a system table.

Character check field error

A character has been entered in a position not allowed by a system table.

Procedure check field error

A field checked against specific criteria stored in the system is found to be invalid.

F.3.3. Format Programming Errors

Errors in a format program may be detected by the system during either of two processes: (1) During the compile time in which the system translates the source program written by a programmer into a form the machine can execute, or (2) During the run time in which the system executes, or runs, the compiled program.

1. Compile Time Errors

During compilation, a listing of the source program is produced on the 01 OUTPUT device of the system. If any system-detected errors exist in the source program, the system notes them on the listing as diagnostic error messages. For example, the system can recognize such language errors as the following:

Incorrect syntax, as when the right closing parenthesis is missing:

PARENTHESIS ERROR

Missing or incorrect use of verb in a procedure statement:

VERB ERROR
SUBPROGRAM VERB ERROR
LABEL PROGRAM VERB ERROR
FORMAT PROGRAM VERB ERROR
ENVIRONMENT VERB ERROR

Multiple use of field name or label in one program level:

MULTIPLE DEFINED

2. Run Time Errors

At run time, the system can recognize some errors in logic or arithmetic; for example, attempted comparison of numerics and nonnumerics in an ALLOW/DISALLOW or IF statement in a format program causes this message to appear:

FORMAT ERROR - COMPARE TYPE

Messages about errors detected at run time are displayed on the message line (line two or three, depending on screen size) of the terminal display screen. If the log device is set to the line printer, the message will also print out as part of the log.

F.3.4. Data Communication Errors

Specific transmission protocols determine how to detect and correct errors that occur during data communication. Generally, whether on the basis of individual characters or entire messages, the receiving end validates received data and responds when errors are detected. The simplest method is parity checking. The transmitter generates an extra bit, the parity bit, in each character. The receiver checks incoming characters for the correct parity; upon detecting an error, the receiving end may send a message requesting retransmission.

F.3.5. System Errors

System errors occur because of software anomalies or hardware failures that cause the software to malfunction. When a system error occurs, the system is halted and can be restarted only by initiating a warm or cold start. Section F.6. lists system errors.

F.3.6. Task Errors

A task error usually halts a single terminal and locks a specific task. This reduces some operational inconveniences created when a hard system error shuts down the entire system. An orderly CLOSE and WARM START procedure can usually get recovery from a task error. Section F.7. lists task errors.

F.4. RESPONSE TO MESSAGES

Some messages require no reply. They simply remain on the screen until replaced by another message. Other messages require a response from the user. Two types of messages requiring a response are: messages cleared by the RESET key only, and messages requiring reset plus a keyed reply.

F.4.1. RESET Key Response

Generally, the messages that display at an operator's terminal need to be acknowledged and cleared by use of the RESET key.

Press the RESET key to clear the alarm message and activate the keyboard for continued operation. Then enter the correct data or response as required.

If the correct information is not readily available, the field may be bypassed temporarily so that current operations can continue. Press the RESET key, then press the CTRL/BYPASS keys. The cursor moves to the start of the next field. The system flags the bypassed field, the record, and the batch as invalid, and the correct data must be determined and entered later.

Messages that require only a RESET key response are usually generated by the format program. Depending on how the program is written, such messages may or may not be accompanied by an alarm and/or backspacing in the field. After pressing the RESET key, just follow the instructions included in the message or provided by the data entry supervisor.

F.4.2. Keyed Response Required

The system places messages in a queue that may include several devices or tasks. Certain kinds of tasks, for example, may be active in the background while the terminal that initiated the task is engaged in another activity. Such tasks might involve the use of Autolog, Autosave, Spooling, Data Communication, Sort, or File Management.

After a background task has been established, it normally operates without direct operator intervention. But if an active background task encounters a condition that requires the supervisor's attention, the system sends a message to the terminal that initiated the task or to the terminal designated by the SET command as a message terminal.

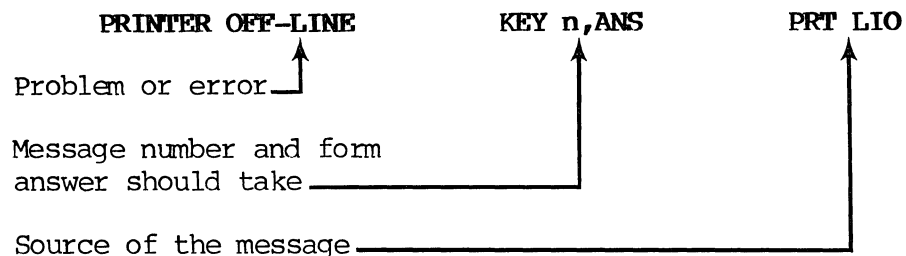
• Message Queue

Up to nine messages can be queued and displayed simultaneously. But after the RESET and CTRL/DISP sequence is keyed to respond to a background message, no other operators can respond to their particular messages until the first response is completed. Thus, a 15-second timer ensures that an incomplete response will be canceled so that other operators can have access to the background message response queue.

To get back into the queue, press the CTRL/DISP keys again and make your response within 15 seconds.

• Message Format

This type of message is provided by the system in the general format as shown. Respond as described in steps 1 through 5 in "Making a Keyed Response."



The first part of the displayed message describes the problem or error. The second part of the message gives a number that must be referred to in a response. The number is assigned by the system and can be 1 through 9, depending on how many messages are backlogged (queued).

The next part of the message may also indicate what form the response should take; for example, enter R for retry or C for cancel; or enter C for next page or R for roll.

The last part, on the extreme right of the message line, names the source of the message--the system module in which the problem area exists. This module name is not part of the response; it is displayed to provide information.

- **Making a Keyed Response**

To respond to a message that requires a keyed response, follow these steps:

1. Note the message number.
2. Correct the problem or error. For example, reestablish a data communication link or put the printer on line.
3. Press the RESET key to clear the message line.
4. Press the CTRL/DISP keys. The system then displays the cursor on the message input line.

The system is ready for a keyed response.

After keying CTRL/DISP, the 15-second timer begins. If a complete response is not entered within 15 seconds, the system will cancel the partial response. To successfully complete a response, repeat steps 3 and 4, wait for the cursor to appear on the message input line, and respond within 15 seconds.

5. Enter response as follows:

n,R or n,C

Where:

n

is the message number noted in step 1. This number **must** be identical to the one the system displayed. It identifies the message being responded to.

R

means retry message.

C

means cancel message.

Example:

PRINTER OFF-LINE KEY 2,R=RETRY C=CANCEL PRT LIO

Note the message number. Check the printer control panel. Press the ON-LINE switch. Press the RESET key.

Key CTRL/DISP. The system displays the cursor on the message input line.

Enter

2,R or 2,C

Press the FIELD REL key.

The system displays:

MESSAGE RECEIVED

The work that was interrupted by the original message can now be resumed.

F.5. GENERAL MESSAGES

This section includes messages that appear on the video display screen to inform, guide, or notify the user that an error has occurred.

If an error is simply a keying error, the message may appear only at the terminal where the error was keyed. Some messages appear also at the supervisor's terminal, or only at the supervisor's terminal. In such cases the supervisor may need to check the source data or format program to find the cause of the problem.

This section also includes messages about format program errors the system detects when compiling or running a program. A diagnostic message about a compile time error prints on the source listing near the text that caused the message to be generated. If the line printer is set as a log device, these messages also print out as part of the operation log.

During data communication, specific transmission protocols determine how errors are detected and corrected. In some situations, the system corrects the error and continues to function. In other cases, direct voice contact with the other end of the communication link to correct the problem may need to be established.

AAAAA

- | | |
|-------------------|---|
| ACCESS | A prompt in the FD UTILITY Supervisor command. Operator must enter a previously defined accessibility code (that has been arranged at the user site) for the volume label. |
| ALL DISKS MOUNTED | All disks have been mounted successfully. |
| ALL SPOOLS BUSY | User attempted to initiate more than three spool files at one time. |
| ALLOCATION | A response to the RESERVE prompt in the DISK MGMT Supervisor command. If this response is chosen, none of the free space on a previously designated disk will be allocated for use by the system. |
| ALREADY COMPLETE | User attempted to create indexed files or sort a data batch, but the process has already been completed. |

ALREADY IN
PROGRESS User attempted to create indexed files or sort
a data batch, but the process has already been
started.

ALREADY RESERVED Batch has already been marked for Sort or Create.

AUTOLOG
TERMINATED General background message indicates autolog
function has ended.

AUTOPROMPT EOF System has found premature end-of-file mark in an
autoprompt batch.

AUTOPROMPT
TERMINATED Autoprompt program is completed.

AUTOSAVE
TERMINATED Background message indicates autosave function
has ended.

BBBBB

BACKGROUND INPUT
BUSY Unable to begin background task; background
already active.

BAD BLOCK CHECK
RCV Too many Negative Acknowledgments (NAKs)
received during data communication.

BAD, DELETED
FROM DISK Error on disk.

BAD DESTINATION
MSG CANCELLED Background message indicates an incorrect message
number has been keyed as part of a message
response.

BAD LABEL The IBM label read from a diskette is bad.

BAD STATUS AUTO
ANS The 2780 autoanswer firmware has indicated bad
status during data communication.

BAD STRUCTURE,
FILE ABORTED Batch name was not a member of file; directory
could not open; tree has no dummy end.

BATCH ACTIVE
or
BATCH ACTIVE
CANNOT LOAD The batch selected is already open at another
terminal. Press the RESET key. Verify that the
batch number was entered correctly. If correct,
either wait for the batch to become available or
select another batch.

BATCH CREATE	A response to the RESERVE prompt in the DISK MGMT Supervisor command. If this response is chosen, no new batches will be opened on a previously designated disk.				
BATCH IS IN BALANCE	All balance registers in the batch are in balance.				
BATCH IS OUT OF BALANCE	One or more of the balance registers in the batch is out of balance.				
BATCH NOT FOUND	Data batch referenced in a directory for data communication has been deleted from disk, the batch is active, or the user selected a nonexistent directory batch number in response to the directory batch number prompt in data communication. Press RESET and verify that the correct number was entered. Then respond appropriately.				
BATCH NUMBER ERROR	Batch number was not a member of file. Valid batch numbers range from 1 through 50,000.				
BATCH NUMBER RANGE ERROR	The batch number keyed is not in the range 1 to 50,000. Press RESET and enter the correct number. Also, if the selected directory batch number is not the last one transmitted, an error results when the transmission is terminated.				
BATCH RANGE ERROR	The directory number given to terminate a datacom transmission does not match the number of the directory currently in use.				
BATCH RECORD COUNT TOO LARGE	Displays if user enters in more than 1000 records per batch.				
BATCH STATISTICS	Message displays on the screen whenever a data batch is closed. See individual entries for specific meanings. The information displays in this format: <p style="text-align: center;">BATCH STATISTICS</p> <table><tr><td>TOTAL REC=</td><td>INV FIELDS=</td></tr><tr><td>VERIFIED REC=</td><td>UPDATED REC=</td></tr></table>	TOTAL REC=	INV FIELDS=	VERIFIED REC=	UPDATED REC=
TOTAL REC=	INV FIELDS=				
VERIFIED REC=	UPDATED REC=				
BKGRND FUNCTION NOT ACTIVE	Invalid command selected for nonactive background function.				

BKGROUND TASK
NOT ACTIVE FOR
THIS FILE Operator is trying to cancel a task that has not
 been initiated.

BLOCKED DISK READ
NOT ACTIVE Insufficient memory to read full disk at one time.

BUFFER OVERFLOW
RCV Number of characters received from host was
 larger than communications buffer.

CCCCC

CANCEL COMPLETE Cancellation of sort process is complete in the
 file management mode.

CANCEL NOT
ALLOWED Cancellation of sort, disk mount or other system
 function cannot be executed at this time.

CANCEL REQUESTED Request has been enqueued, but not yet completed.

CANCEL
UNSUCCESSFUL In sort process, cancelation could not delete
 some scratch files (disk error).

CANNOT ABORT--
OPERATION IN PROG Appears when the supervisor tries to terminate
 host interactivity, but an operation is in the
 process of being executed and the system cannot
 cancel it.

CANNOT FIND PROC
OR PGM An attempt was made to refer to a procedure or
 program that does not exist.

CANNOT FIND
PROGRAM An attempt was made to refer to a program that
 does not exist on the diskette.

CAN'T OPEN
DEVICE Operating system is unable to open the required
 device. Press the RESET key and do a retry. The
 device may already be in use. Or the incorrect
 device may have been specified. For example, if
 the 01 OUTPUT= LP, and a SAVE is attempted, this
 message will appear.

CARD READER NOT
READY System attempted to read card deck before card
 reader was ready.

CC LINE ERROR 1 Host interactivity transmitted a message and
 received a negative acknowledgment from the host.

- CC LINE ERROR 2 An error has been detected within a message received from the host, and the host interactivity subsystem responded with a negative acknowledgment.
- CC LINE ERROR 3 During host interactivity, after a message was sent to the host, the following occurred:
- Host responded with the wrong acknowledgment (ACKO instead of ACKI or vice versa).
- The HI subsystem responded with ENQ to elicit the proper response.
- CC LINE ERROR 4 After acknowledging a received message, the host interactivity subsystem has received an excessive number of ENQ messages from the host.
- CC LINE ERROR 5 During host interactivity, the host has sent an excessive number (15) of consecutive Temporary Text Delay (TTD) data link control messages.
- CC NOT CONFIGURED Data communication controller was not configured
--TASK ABORTED during SYSGEN process. Task is aborted.
- CC NOT RESPONDING Communication controller is not responding. Task
--TASK ABORTED is aborted.
- C = NEXT PAGE,
R = ROLL A data communication R = function outputs to the terminal display screen. It allows the user to select either paging (one page at a time) or rolling (continuous roll to end of job). It is possible to select a roll response after any number of page responses, but a page response cannot be selected after the first roll response.
- CHANGE TRANSMIT
TO RECEIVE RVI In data communication, the host computer wants to transmit and is overriding the current outgoing transmission by sending a reverse interrupt control code (RVI). Press RESET key. The receive mode is then automatically established.
- CHAR BKSP INVALID
AT FIELD START The CHR<-- key cannot be used to move backward from the first position of the first field in a record. Press RESET and press REC<--. Also, the CHR<-- key cannot be used to move back to any nonkeyed field.

CHAR FWD INVALID The CHR--> key cannot be used to move beyond the last character entered or verified. Press the RESET key and enter the next character.

CHARACTER LOST--
REKEY The system received keystrokes faster than it was able to process them, and the 40-character keystroke buffer was full. Press the RESET key and reenter.

CHARACTER
TRANSMISSION
ERROR - REKEY Reflects the same problem as the "CHARACTER LOST--REKEY" message, except that it appears during data communication. Press the RESET key and reenter the last character.

CHECKPOINT RECORD
NOT FOUND System does not find the indicated checkpoint record. Backspacing beyond a file or directory start or advancing beyond a file or directory end causes this message to display. Under control of the CHKPT CNTL command, the user may have keyed in a record number larger than exists in the subject file. Press the RESET key and enter the correct record number.

CLAUSE ERROR An incorrect statement exists in the environment or data definition division of a file management format program.

CLOSE SUBFILE I/O
ERROR Compiled file could not be closed. Compilation was aborted. Try again to compile.

COMM CNTL ACTIVE
- TEST ABORTED System is unable to run the on-line communications test because the communication controller is active.

COMM LINE ACTIVE After the host interactivity subsystem is loaded, the subsystem waits indefinitely for the first poll from the host computer. When the first poll is received, this message displays.

COMMAND MANAGER
TASK COMPLETE Background command manager task is now complete and the terminal returns to the foreground operating mode. This is equivalent to the "READY" message that appears in other operating modes. Press the RESET key and continue other activities.

COMMAND MODULE
BUSY - TASK
CANCELLED In file management a request for load or unload is made, but the function is already busy.

COMMUNICATIONS
ABORTED Remote station forced abort. Press RESET key.
Call other station and reestablish communications
if possible. A checkpoint will have been written.

COMMUNICATIONS
TERMINATED In data communication, this appears following
RECEIVE TERMINATE or TRANSMIT TERMINATE
commands. The appropriate overlays are released
from memory (includes release of HI subsystem).

COMPILATION
ABORTED An unrecoverable error has occurred. Error is
described in the program source listing.

COMPILATION DONE Batch was successfully completed.

COMPILED FILE
CONFLICT A compiled program with the same name is already
on disk. The compilation or restore is aborted.

CONFLICTING
EXTENTS This references a diskette label function. An
attempt was made to allocate the same disk area
to two data sets.

CORRECTIONS= Operator statistics include the total number of
times the VCOR key was used to correct an invalid
field in all modes except in the Verify mode. In
the Verify mode the count is incremented once for
each character changed after a verify miscompare
and once for each character keystroke in a field
that is changed with the use of the VCOR key.

CR FEED CHECK Cards were not fed into card reader properly, or
a given card was rejected because of damage or
staples.

CR READ CHECK The last card in the card reader output stacker
may have leading or trailing edges.

DDDDD

DATA ERROR (describes type)	The last character keyed is not an allowable data type in the field. Press the RESET key and enter the correct character or bypass the field.
- NOT ALPHANUMERIC	
- NOT ALPHA	
- NOT BLANK	
- NOT COBOL	
- NOT HEX	
- NOT LOWERCASE ALPHA	
- NOT NUMERIC	
- NOT PUNCTUATION	
- NOT SIGNED DECIMAL	
- NOT USER n (user 1 through 8)	
DATA LENGTH EXCEEDS KEY LENGTH	Selected data is longer than key length definition.
DATA LOSS RCV	An error causing loss of data occurred during data communication receive.
DATA LOSS XMT	An error causing loss of data occurred during data communication transmission.
DATA NOT MARKED	User requested Sort or Create process on unreserved data.
DATA REC LENGTH TOO SMALL	Record length is inconsistent with definition in the file header.
DATA SET NOT FOUND	A diskette on which a new volume is expected to continue a data set does not have a data set label of the same name as in the previous volume. Replace with correct diskette or let system generate proper label name with different extents (adjacent areas of physical storage).
DATA SET NOT READY	The DATA SET READY (DSR) signal from the modem was lost while attempting to receive or transmit data. Cancel or retry.
DATA SET WRITE PROTECT ERROR	A diskette assigned for output is write-protected.
DATA TYPE INVALID	The data type entered does not match the data type allowed by the format program.

DATA SET NOT READY In data communication, the modem is not yet ready for transmit/receive (i.e., the link has either been broken or has not been established). Retry or cancel the task.

DATE Batch statistics include the current date.

DATE RECORDS DESCEND ASCEND REC REQD Indexed file status headings.

DCOM RECEIVE ACTIVE Data communication receive has been previously initiated and is still active.

DCOM RECEIVE CANCELLED---NO TASK In data communication, a reverse interrupt control code (RVI) has been received, but receive mode has not been initiated.

DDC BUFFER COUNT CHANGED TO n User has changed the number of disk data cache buffers to be used to n.

DELETED TRACK HARD ERROR An attempt to delete a defective track on a diskette results in a hardware error.

DELETIONS= Operator statistics include the total number of records deleted from a data batch.

DESTINATION IS NOT CURRENT FIELD Field name is used as destination but is not current field.

DEVICE ERROR---JOB ABORTED In file management, an unrecoverable disk error was encountered during a sort process.

DEVICE IS BUSY User requested a device that is already in use.

DEVICE UNAVAILABLE User selected a unit not configured on the system or tried to load host interactivity subsystem while other data communications activities were in progress. Retry or cancel the task.

DIRECTORY BATCH COMPLETED Normal completion of a spool-off function.

DIRECTORY NAME MUST BE FULLY KEYED User entered a partial directory name followed by a period. This is not allowed here.

DIRECTORY NOT FOUND	User selected a nonexistent directory batch number.
DIRECTORY NUMBER ERROR	User specified an unallowable directory number.
DISK AUTOMOUNT AND RECOVERY IN PROCESS	Disk is being mounted automatically during a system startup.
DISK FILE OPENED	The system opens a batch automatically and displays the job name and batch number. The disk file begins receiving data.
DISK FULL OR RESERVED	Disk capacity has been reached.
DISK FULL--TASK TERMINATED	May appear when the operator uses the POWEROFF Supervisor command and not enough disk space is available to complete the operation. The result is that the automatic mount feature is suppressed. No data loss occurs. However, for a subsequent WARM START, mount disk platters. May also be seen during sort or input when platter becomes full.
DISK FULL OR RESERVED--TASK TERMINATED	Disk capacity has been exceeded. Batch closes automatically.
DISK HARD ERROR ON WRITE	System cannot write to disk because of a hard error.
DISK INIT FAIL	An attempt was made to mount a diskette (unable to bring the head into contact with the diskette). Try to mount diskette again. During initialization of diskette, diskette was found bad.
DISK NOT MOUNTED	An I/O operation was attempted on a file on a disk drive that does not have the head in contact with the diskette (initialization). Mount the diskette.
DISK...% LEFT...STATUS	Headings under the "STATUS" prompt in the DISK MGMT Supervisor command that indicate the drive number, the percentage of unallocated space left, and the mount and reserve status of all disk drives.

DISK 98 PERCENT FULL WARNING--disk is reaching its capacity.

DISK NOT READY The disk is not in the ready state to mount (appears during DISK MGMT MOUNT procedure).

DISK 1 RESTRICTED AREA 98% FULL The reserved area of the system disk is 98 percent full. Compiles are stored in the reserved area. This message appears on the message line of the Supervisor's terminal screen once a minute. If you are using more than one disk, this message will only appear if the reserved areas of **all** disks are 98 percent full.

DISK XX--DISK NO AUTOMOUNT During automount the system recognizes a disk that is differently formatted from those dismounted during a poweroff.

DISK XX--NEW DISK --NO AUTOMOUNT Disk error is encountered during automount process.

DISKETTE n NOT READY May appear during the flexible diskette mount process when diskette drive n has been assigned as an I/O device, but is not ready. Press the RESET key, enter l,ANS xxx (xxx identifies the origin of the message).

DISKETTE NOT AVAILABLE The system is not configured with a flexible diskette drive.

DISKETTE WRITE PROTECT ERROR The write-protected slot on the diskette to be initialized is left exposed so that the system cannot write to diskette.

DISMOUNT IN PROGRESS This is the status of the disk drive when status is requested through the Supervisor DISK MGMT command.

DISMOUNTED DRIVE BUSY Same as "DISMOUNT IN PROGRESS". When FD DIAG is running, this message appears if the return from the system routine GET DCB PNTR has the DCB DEVICE BUSY bit set.

DRIVE NOT READY When FD DIAG is running, this message appears if the return from the routine FD MOUNT has the Z condition code set.

DUP KEY INVALID ON FIRST RECORD Use of the DUP key is not allowed in the first record of the batch. Press the RESET key and enter the field.

DUPLICATE FILE Batch on a disk being mounted has same name as a batch on a disk already mounted.

DUPLICATE JOB AND BATCH NUMBER The job name and batch number are already entered into the system and may not be assigned to more than one job. Press RESET and verify that the name and number were entered correctly. If the entries were correct, then new numbers must be assigned.

DUPLICATE KEY NOT PERMITTED While building the index tree, the system detects a duplicate where no duplicates are allowed. This may also occur during the Sort process.

DUPLICATE FLAG ERROR In the sort-initialize process used to create a sort file header, the duplicate flag was specified as a number other than 0 or 1.

~~EEEE~~

END-OF-FILE A single character indicating the end of a group of records has been encountered during a tape read.

END OF VOLUME When writing to a BDE diskette, the system has used all the space within the specified extents but more data remains to be written. Change extents or use multi-volume procedure.

ENTER ONE JOB NAME ONLY Appears if user attempts to enter more than one job during an operation (like Sort) when only one name can be entered. Use a directory name instead.

ENTER - VRFY - UPDT When the system displays batch statistics, the name of the current operating mode (ENTER, VERIFY, or UPDATE) also appears.

ENTER/VERIFY SEPARATION TOO SMALL The number of records separating the record being entered and the record being verified is not large enough. Press the RESET key and wait before continuing to verify.

ERROR	ENVIRONMENT VERB specified in the environment division of a file management format program.
EOT, MOUNT NEXT TAPE	During magnetic tape operations, the system has reached the end-of-tape mark and requests the next tape.
ERMAP STRING ERROR	The ID field of the diskette error map does not contain the five characters ERMMap.
ERROR MAP HARD ERROR	The system detects a hardware error while reading or writing to the diskette error map sector.
ERROR TRACK ERROR	The cylinder 1 or cylinder 2 field of a diskette contains an invalid track number and the associated deleted track indicator is not blank.
ERRORS=	Operator statistics include the total of all errors that require the RESET key in all modes except the Verify mode. In the Verify mode, the count is incremented once for each character changed without using the VCOR key and once for each use of the VCOR key.
ERRORSET LABEL ERROR	The first data label sector on an extended diskette does not contain a valid ERRORSET label.
EXTENT RANGE ERROR	The user tries to copy a diskette by a range of addresses not present.
FFFFF	
FAIL ON RECOVER	Appears during startup and means a COLD START will be required. Also issued if user keyed in "1,C" and the recovery required a message or if system is unable to find last file change.
FAST RECOVERY IMPOSSIBLE	The fast recovery feature has been disallowed for one of two reasons: a. A regular batch has been written to a restricted area as would occur, for example, when the disk is 98 percent full; b. A reload has been initiated while the "DISK AUTOMOUNT AND RECOVERY IN PROGRESS" message was displayed.

FAST RECOVERY IN PROGRESS	A speeded-up WARM START is in progress. This type of restart does not recover unused AUs.
FIELD DESCRIPTOR REQUIRES FIELD SIZE	Field size was not entered. Check the format program and enter the correct information.
FIELD FWD INVALID	FLD--> is not allowed beyond the last field entered or verified unless it is designated for sight verification and the Verify mode is in operation. Press the RESET key and rekey the field.
FIELD MUST BE FULLY KEYED	Use of the FIELD REL, REL and -SKIP keys is not allowed until data has been keyed into all positions in the field. Press the RESET key and rekey the field or bypass the field and make note of the bypass.
FIELD MUST BE KEYED	Use of the FIELD REL, REL and -SKIP keys is not allowed until at least one data character is keyed in the field. Press the RESET key and rekey the field or bypass the field.
FILE CLOSED BY SYSTEM	The system closed a file left open when terminal was closed.
FILE IS ACTIVE	User tried to delete an active file.
FILE NAME NOT AN I.F.	File specified is not an indexed file.
FILE NAME NOT FOUND	File specified was not found.
FILE NAME...TYPE ...STATUS ...FORMAT	Indexed file status headings, as requested through the Supervisor's FILE MGMT file status command.
FILE NOT COMPLETE	File does not contain all necessary elements.
FILE NOT FOUND	User tried to delete a file that the system could not find.
FILE PROTECT ERROR	The write ring on a magnetic tape reel has been removed, preventing the system from saving or writing data to tape. Add the ring and retry.

FILE SUMMARY File summary heading.

FILE WAITING ON A CLOSED SPOOL Pending file is waiting for spool to be closed.

FILE WAS MODIFIED Partial delete option was selected, but could not be carried out because the file had been modified.

FILES ACTIVE-- CANCEL? On an unload, a file is still active.

FILES OPEN-- RETRY? On an unload, a file is open.

FIRST CHAR MUST BE COLON (:) User entered a directory name that did not begin with a colon; the colon is required.

FLD BKSP INVALID ON 1ST FIELD FLD--> is not allowed in the first position of the first field in a record. Press the RESET key and press REC--> key to backspace to a previous record. Then press FLD--> key to advance to the desired field in that record.

FMT ERR - SAVE STMT BEFORE RESERVE In a format program using subbatch balancing, an expected SAVE statement has not been found before the next RESERVE statement. Correct and recompile the format program.

FMT ERR - SUB BATCH RCD LTH In using a format program with subbatch balancing, the records being processed are less than 74 characters in length. Subbatch balancing may not be used with such data.

FORMAT ERROR- (describes type) A runtime error has occurred. Check the format program.

FORMAT ERROR- ACCESS TYPE Arguments in a format and a called subprogram are not of the same type.

FORMAT ERROR- ARGUMENT COUNT Number of arguments in format and called subprogram are not the same.

FORMAT ERROR- COMPARE TYPE Mixed data types (numerics and nonnumerics) are being compared in a single ALLOW/DISALLOW or IF statement.

FORMAT ERROR-
DESTINATION TYPE The data type of the destination of a move is not the same as from the field. Moving one type of data field to another type of data field is not possible (e.g., numeric to non-numeric).

FORMAT ERROR-
LABEL REFERENCE System is looking for label address and is unable to access the label.

FORMAT ERROR-
LOGICAL TYPE Destination in IF statement using AND, OR, or NOT is incorrect; or division by zero has been requested.

FORMAT ERROR-
POWER SIZE Operand in an exponentiation operation is too large.

FORMAT ERROR-
SOURCE TYPE Occurs when system is looking for next source line in the format program and is unable to locate it.

FORMAT ERROR-
SUBSCRIPT
OVERFLOW Subscript value has been allowed to become less than 1 or greater than the number of characters in the string being subscripted.

FORMAT ERROR-
TIME LIMIT
EXCEEDED Procedure is looping until time-out occurs.

FORMAT LABEL
RETURN ERROR A RETURN ERROR statement has been executed during processing of a label program. Unrecoverable.

FORMAT NOT FOUND The format requested or associated label programs, subprograms or tables are not stored on disk. (The type of the missing program--label program, subprogram or table program--is indicated in the message by the presence of "L:", "S:", or "T:" respectively.) Press the RESET key and determine whether the format name was keyed correctly. If correct, check through the Supervisor STATUS command that the format and all associated subprograms and tables are present on the system.

FORMAT RANGE
EXCEEDED The mapping range is exceeded and the TWA, the main format and submodule cannot be referenced, or when a requested tenant exceeds the maximum length available.

FORMAT PROGRAM
VERB ERROR

User specified a verb that is not allowed in a format. See the **System Supervisor's Reference Manual** for the list of reserved COBOL words to ensure correct verb use.

GGGGG

GENERAL PART OF
HEADER NEEDED

In the sort-initialize process used to create a sort file header, the general part was not entered. This part includes the data batch type, data record length, data job name, and data batch number.

GENERATE LABEL?

In a multi-volume diskette write, the system has not found the data set name on a subsequent volume and asks whether to generate a label automatically.

GROWTH DEFAULTED
TO 50

User attempted to specify a growth greater than 50 when 50 was defaulted.

HHHHH

HARD ERROR

An operation failed due to a permanent error.

HARDWARE ERROR

Problem with a piece of hardware. Must be corrected before continuing.

HI FILES OPEN

During the process of unloading the host interactivity subsystem from memory, the system detects some HI files still open. Retry or cancel.

HI OPERATION
ABORTED

During host interactivity, if the communication link is accidentally broken and then reestablished, the format program immediately resumes processing at the statement following the canceled SEND or RECEIVE statement.

HI SUBSYSTEM
LOADED

The host interactivity subsystem is successfully loaded into memory.

HOST NOT POLLING

After the first poll is received from the host computer during use of host interactivity, this message displays when there is any 1-minute interval of nonpolling activity.

HOST NOT
RESPONDING

After having been established, the data communication link with host computer was lost. Either the host interactivity subsystem has not received any information from the host within the previous minute, or none of the messages received within the minute has been recognizable.

HYPHEN CANNOT
END WORD

A word cannot end with a hyphen character. Change the word if necessary.

IIIII

ID CONFLICT

Use of name conflicts with previous use (i.e., name is used as both program name and field name).

I.F. BATCH NOT
AVAILABLE

Specified Indexed File Batch is not available for I/O.

IFH DISK ERROR

During the sort-initialize process, the system encountered a disk error when trying to write the Indexed File Header (IFH) to disk.

IFH REC POS=0

During the sort-initialize process, the key record position was set to zero.

I.F. RESTORE
ACTIVE--CAN'T
START

Indexed File Restore process has already been started.

I.F. RESTORE
BKGRND INIT FAILED
--ABORTED

Indexed File Restore process was unsuccessful and the task was aborted.

I.F. RESTORE NOT
ACTIVE--CAN'T
CANCEL

User tried to cancel an Indexed File Restore that ACTIVE--CAN'T has not been initiated.

I.F. SAVE ACTIVE--
CAN'T START

Indexed File Save process has already been started.

I.F. SAVE BKGRND
INIT FAILED--
ABORTED

Indexed File Save process was unsuccessful and the task was canceled.

I.F. SAVE NOT
ACTIVE--CAN'T
CANCEL

User tried to cancel an Indexed File Save that was not active.

ILLEGAL ARGUMENT	Argument in a CALL statement is not a literal, register, or field name.
ILLEGAL CALL	A function is used improperly.
ILLEGAL CHARACTER STRING	A table argument type is designated as C, but table does not have a character string.
ILLEGAL CONDITION	Either an IF statement does not contain a conditional expression, or a FIND statement does not have a table condition.
ILLEGAL DATA RECEIVED	Host sent data after the receiving work station requested data end or after reception had terminated.
ILLEGAL DESTINATION	The term specified cannot be used as a destination in a procedure statement.
ILLEGAL FILL CHARACTER	User entered the incorrect fill character. Press the RESET key and enter the correct character.
ILLEGAL FOR ZERO SIZE FIELDS	An entry has been made in a checkbox where field size was specified as 0, or a zero size field has been used as a source or destination.
ILLEGAL HEADER	The first character of the first record being compiled must be P, S, L, or T (program, subprogram, label, or table).
ILLEGAL IN FIRST FIELD	For the first field of a level, the entry in checkbox 12 (TAB/CONTINUE) cannot be T, B, or C.
ILLEGAL IN SUBPROGRAM	The words THIS FIELD or B and D registers are not allowed as operands in subprograms.
ILLEGAL NUMBER	A minus sign in a table entry must be followed by a number.
ILLEGAL OPERAND	Operand is not a field name, register, literal, or THIS FIELD, or is illegal for expression type.
ILLEGAL PROCEDURE FOR PROMPT FIELD	The only procedure allowed in a prompt field is the PROMPT statement. The PROMPT verb must be followed by a numeric or alphabetic literal or a specific register number.
ILLEGAL REGISTER	The register type specified cannot be used.

ILLEGAL RESPONSE User tried to enter the Command Manager mode from the Supervisor mode.

ILLEGAL VOLUME During volume switching, the volume label of a new diskette is incorrect or unusable.

INCOMPATIBLE
DISKETTES Diskette copy function has found that the diskette to be copied and the diskette to copy to are initialized to different formats or have a different number of sides.

INCOMPATIBLE FMT The job cannot be entered under the format selected. Press the RESET key and verify that the format name was entered correctly. If entered correctly, one of the following errors may exist:

- Field size as specified in the controlling format program does not equal the original field size.
- The end of the data records is reached, but the controlling format calls for one or more fields to be processed.
- Format program indicates end of record, but actual data records contain more fields.

INDEXED FILE= Indexed file number is specified.

INDEXED FILE
PERMANENT ERROR A hardware error has occurred during I/O using an indexed file.

INDIO UNLOAD IN
PROGRESS During poweroff, if file management (INDIO) modules are in memory, the system will automatically unload them and proceed.

INPUT END OF
VOLUME: Fn An end-of-volume condition occurs when data is being input from a basic diskette (F2, F3, or F4). Press the RESET key and respond to the prompt sequence.

INPUT RECORD
CHANGES Record to be changed has been located during a search of the tape (Supervisor SM TAPE command). Modify the record as needed.

INPUT RECORD
CONTENTS Prompt displayed to begin entering data into record.

INSERTIONS= Operator statistics include the total number of records inserted in a data batch.

INV FIELDS= Batch statistics include the number of fields marked invalid in a data batch.

INVALID CHARACTER The current entered record does not meet format-specified criteria.

INVALID COMMAND - BKGRND ACTIVE Invalid command was selected for active background function.

INVALID COMMAND SELECTED Press the RESET key and select another command.

INVALID CONTINUATION Wrong response to continue prompt. Try again.

INVALID DATA Data is not compatible with file structure.

INVALID DATA SET The data set name entered is not in the correct format.

INVALID DATE User gave invalid response to date prompt. Enter date in MMDDYY format.

INVALID DEVICE CODE Invalid response to input or output device prompt.

INVALID ENTRY Checkbox or procedure field contains an invalid entry.

INVALID FIELD - CORRECT OR PASS Data entered does not meet the criteria for validity. Press the RESET key and correct or bypass the field.

INVALID FILE File definition is not compatible with selected procedure.

INVALID FILE NAME The file name entered is not in the correct format.

INVALID GROWTH SIZE An illegal file growth size was specified.

INVALID INPUT DEVICE ASSIGNMENT Any device other than a card reader, tape, or diskette (F1, F2) is assigned.

INVALID JOB NAME The job name entered is not correct. Press the RESET key and verify that the name was entered correctly. If entered correctly, then the wrong name is being used.

INVALID KEY Key definition not compatible with selected procedure.

INVALID KEY AT
LST FIELD Use of the FLD--> or CHR--> key is not valid in the first field of a record. Press the RESET key and enter the field.

INVALID KEY AT
MIDFIELD Cursor must be in the first or last position of a field for you to use the key just pressed. Press the RESET key. Finish entering the field or press FLD--> or FLD<-- and try again.

INVALID KEYSTROKE The last character entered is not allowed by the format. Press the RESET key and enter the correct character or function key.

INVALID LABEL ID Disk error message. Label is not the same as that on disk.

INVALID LEVEL
CHANGE An incorrect level designation has been entered, or the logical point of level selection has been passed. Press the RESET key. Enter the correct level or resume entering at the current level.

INVALID LEVEL
SELECTED User has entered a nonexistent or nonselectable level. Press the RESET key. Enter the correct level identifier or resume entering at the current level.

INVALID MODE
CHANGE AFTER
POWER RECOVERY Previous mode specified does not match new mode.

INVALID OUTPUT
DEVICE ASSIGNMENT The specified output device was not configured on the system during the SYSGEN process.

INVALID PASSWORD User entered an incorrect password.

INVALID RECORD Record definition not compatible with selected procedure.

INVALID REC
POSITION Record position invalid for specified file.

INVALID REQUEST-- User requested activity from a completed
DIR CLOSED directory.

INVALID REQUEST-- User requested activity from a completed file.
FILE CLOSED

INVALID REQUEST-- The requested spool file is already active.
SPOOL ACTIVE

INVALID REQUEST-- A command is invalid because it specifies a
SPOOL INACTIVE spool-off task that is inactive.

INVALID REQUEST-- A command is invalid because it specifies a spool
SPOOL NOT file not in a suspended state.
SUSPENDED

INVALID REQUEST-- A command is invalid because it specifies a spool
SPOOL SUSPENDED file in a suspended state.

INVALID STATEMENT The procedure field contains an invalid statement.
IN PROC

INVALID TIME User gave invalid response to time prompt; enter
in HHMM format.

INVALID UNIT If a device has multiple unit numbers (tape,
NUMBER disk, diskette), this message will display when
an invalid number for that device is entered.
Either the device was not specified at the time
of SYSGEN, or the number entered is out of the
allowable range.

I/O ERROR: A READ RECORD or WRITE RECORD execution has
PERMANENT resulted in a hardware error during label program
execution.

JJJJ

JOB NOT FOUND The job name entered is not stored on disk.
Press the RESET key and verify that the name was
entered correctly. If entered correctly,
possibly the wrong job name is being used.

JOBNAME REQUIRED FIELD REL cannot be pressed in response to the
current prompt. You must enter a job name.

KKKKK

KEY AND FILE INCONSISTENCY Key and file definitions are not consistent. Key length, character positions, etc., must be compatible within records in a file.

KEY CLAUSE MISSING An error exists or clause has been omitted from the environment division of a file management program for indexed files.

KEY...KEY...KEY... SORT...INDEX... SORT... Indexed file status headings.

KEY LENGTH RANGE ERROR During the sort-initialize process used to create a sort file header, the key length was set to a number greater than 328.

KEY NOT SORTED Task was suspended.

KEY OUT OF REC RANGE During the sort-initialize process, key record plus key length was found to be greater than record length.

KEY RECORD During a data communication transmission, a record must be keyed because the job input device was specified as the keystation.

KEY RECORD TO BE INSERTED INS has been pressed in Search/Modify or Verify.

KEY SEARCH ARGUMENT Enter the key specified to search file.

KEY SEQUENCE ERROR While building an index tree from batches that have supposedly been sorted, the system detects an out-of-sequence key.

KEYBOARD DISABLED When operator keys CMND/D, the keyboard becomes inoperable until the keyboard is enabled and this message appears. Keying CMND/D stops the clock on the terminal when open in Enter, Verify, Search/Modify and Update modes. Keying CMND/A reactivates the keyboard and restarts the clock.

KEYED FIELD CANNOT BE A DESTINATION The current field is specified as **keyed** and therefore, cannot be used as a destination.

KEYED SIZE ERROR	Keyed size exceeded. (Message occurs during attempted compile.)
KEYSTROKES=	Operator statistics include the total keystroke count for a given data batch. The count does not include the use of RESET and other keys to correct an error condition.
KEYSTROKES/HR=	Operator statistics include the keystroke-per-hour rate for a given data batch operation.
LLLLL	
LABEL MUST BE SPECIFIED	When mounting diskette, no label available. Run Label program.
LABEL # TOO LARGE	During a Label program execution, a WRITE RECORD or READ RECORD attempt has failed because the contents of the sector number register (All) specify an out-of-range value.
LABEL PROGRAM VERB ERROR	The verb specified is not allowed in label programs. See the System Supervisor's Reference Manual for list of allowed COBOL verbs.
LABEL REQUIRED	Needs a label. Run Label program.
LABEL TOO LONG	An attempt has failed because of an out-of-range value.
LABEL UNREADABLE	No proper identification on diskette.
LAU IO ERROR	This error is typically hardware-related and indicates a bad status was returned on an attempt to read or write in a disk LAU (Label Allocation Unit) containing pointers for all system files.
LENGTH CHECK	Record is longer than specified record length.
LEVEL=	Prompt for assigning the level to a label.
LEVEL ID ERROR	Level identification must be a letter, number or blank.
LIST NOT FOUND	The system tries to sort, but cannot find the specified list.

LOAD POINT
REQUIRED WITH
LABELS Invalid tape parameter. If tape is to be
labeled, the load point option must be specified.

LOGICAL RECORD
BUFFER SIZE
EXCEEDED The output position specified exceeds the maximum
logical record length.

LP UNAVAILABLE The line printer is busy.

LP OFFLINE The line printer is off-line. Note the message
number, press the RESET key and put the printer
on-line. Key CTRL/DISP, the message number, then
,R for retry or ,C for cancel.

MMMM

MAP IO ERROR Segment number does not exist in map.

MAXIMUM REGISTER
SPACE EXCEEDED The defined register size has been exceeded.

MESSAGE CANCELLED This displays when the user does not press the
FIELD REL key after typing an N,ANS response, but
instead keys CTRL/BYPASS.

This message also displays when a background task
needs attention and the cursor is on the message
input line. If response is not entered within
15 seconds, "MESSAGE CANCELLED" displays.

MESSAGE RECEIVED The user has entered a response to a background
task message.

MISSING NAME The header must have a name. Enter name in
header checkbox 2 on the source input coding form.

MISSING QUOTE A nonnumeric table entry or literal is missing
the beginning or ending quotation mark.

MONITORING LARGE
WITH SMALL
INVALID The supervisor cannot monitor the model 3555 or
4143 terminal (13-inch screen) from the model
3541 or 4141 terminal (9-inch screen).

MONITORING SMALL
WITH LARGE
INVALID The supervisor cannot monitor the model 3541 or
4141 terminal (9-inch screen) from the model
3555 or 4143 terminal (13-inch screen).

MONITORING TERM 0 OR 1 INVALID	The supervisor tried to monitor terminal 0 or 1.
MONITERING TERM OUTSIDE RANGE INVALID	The supervisor tried to monitor a terminal unit with a number outside the allowable range.
MONITORING YOURSELF INVALID	The supervisor at a designated supervisor terminal cannot monitor the same terminal. The identical unit number was probably entered accidentally.
MORE THAN 127 DIRECTORY RECORDS	Allowable number of directory records was exceeded.
MOUNT ABORTED	Diskette mount forced to terminate.
MOUNT FAILED	The disk mount failed.
MOUNT IN PROGRESS	The system is in the process of mounting the disk.
MOUNT INPUT FD:Fn	An end-of-volume condition has occurred during a diskette input operation (F1, F2, F3, or F4). Enter CTRL/DISP 1,C to cancel or CTRL/DISP 1,R to retry to allow volume switching.
MOUNT INPUT TAPE: Tn	Tape mount on unit T1, T2, T3, or T4 is required for background data communication input function.
MOUNT NEXT TAPE	Tape error message. Mount another tape to complete task.
MOUNT NOT REQST	Drive n. An alternate drive is configured, but warm automount was not requested. This message occurs during status and is a reminder that drive n has not been mounted automatically. RESET must be pressed to continue the startup procedure.
MOUNT OUTPUT FD:Fn	This message appears after an end-of-volume condition occurs during a diskette write operation (F1, F2, F3, or F4). Enter CTRL/DISP 1,C to cancel or CTRL/DISP 1,R to retry to allow volume switching.
MOUNT OUTPUT TAPE: Tn	Tape mount on unit T1, T2, T3 or T4 is required for background data communication output function.

MOUNT STARTED Status report. Diskette mounting in progress.

MOUNTED Indicates the status of the disk drive when status is requested through the Supervisor DISK MGMT command.

MULTI SURFACE ERROR The multisurface string in the volume label (position 72) is not blank for BDE, 1 for extended type, or M for type H.

MULTIPLE DEFINED LEVEL More than one level has the same identification.

MULTIPLY DEFINED A field name has been used for more than one field in a particular program level, or a label has been used more than once in the same procedure.

MUST BE A FIELD The SET INVALID/VALID statement does not reference a defined field name or the TAB statement does not reference a forward field name.

MUST BE ALPHA Format, subprogram, table, and label names must begin with a letter.

MUST BE AT LOGICAL RECORD START Expression must be at record start.

MUST BE BLANK A field specified as nonkeyed cannot contain an entry in the MUST RELEASE, VERIFY or AUX DUP checkboxes.

MUST BE FIRST STATEMENT Statement is not the first executable statement in the procedure or the ENTRY or ALLOW/DISALLOW statement is not the first procedure.

MUST BE Y FOR KEYED, SIGNED FIELDS The MUST RELEASE checkbox must contain Y or a blank (blank defaults to Y when data type is S for signed fields).

MUST KEY REPLY An answer appropriate to the given background message is required. Key CTRL/DISP followed by a valid response (1,C for example).

NNNNN

NESTED CONDITION NOT ALLOWED An AT END or INVALID KEY clause specifies a condition; such a clause cannot contain another condition statement. Similarly, an IF, FIND, or another I/O statement cannot contain an AT END or INVALID KEY clause.

NINE TRACK TAPE UNIT - NO OPTIONS Tape error message. Options may not be specified when a nine track tape is mounted.

NO ALLOCATION The supervisor can specify that nothing can be added to the disk. Existing data can be read.

NO BATCH CREATION The supervisor has specified that no new batches can be added to the disk. An existing batch can have additions or deletions made to it.

NO DETAIL REPORT ON ACTIVE BATCHES Detailed report was specified for active jobs. Detailed report may be generated only for closed or interrupted batches.

NO FILE, BLOCK SKIP WITH LABELS Invalid tape parameter. If tape is to be labeled, no blocks or files can be skipped.

NO:...FMT IN USE Trying to delete a compiled format program that is being used.

NO HI REQUEST PENDING The Supervisor has requested termination of host interactivity. If there is no active HI processing, HI is terminated and other tasks continue as usual.

NO NEW ACTIVITY No new activity to disk can be performed.

NO OP-STATS BATCHES ON DISK An attempt has been made to run the operator statistics report, but no OP-STATS batches exist on disk.

NO OUTSTANDING MESSAGES TO REPORT Monitoring of the message queue indicates there are no unretired messages on the system.

NO ROOM RETRY/CANCEL In executive memory space allocation, there is not enough room for requested task. Retry or cancel.

NONSYSTEM FILE Tape error message. Tape mounted was not written by system.

NOT A SPOOL DEVICE User attempted to use device not configured for spooling or the selected device is not log, disk, or line printer.

NOT ALLOWED AS FIELD NAME A word reserved for use in COBOL procedure statements cannot be used as a field name.

NOT AT LOAD POINT Tape error message. The tape is positioned beyond load point.

NOT ENOUGH ROOM Not enough room exists in standard memory to accommodate a new overlay or format, even if room is available in extended memory.

NOT ENOUGH ROOM FOR TASK--JOB CANCELLED Memory required during compilation exceeds available memory. Any one of six on-line diagnostic modules attempts to run, but not enough memory is available to set up the large I/O buffer needed (FD DIAG, KEY DIAG, TAPE DIAG, CD DIAG, LP DIAG, and SP DIAG).

NOT ENOUGH ROOM--TASK CANCELLED System does not have enough memory available to complete the task.

NOT FORWARD REFERENCE The TAB statement does not reference a forward field.

NOTE MSG #, CTRL/DISP Check for the message number before responding to a background task. The number displayed is the number to use.

NUM POS SIZE STATUS STATUS SEQUENCE Indexed file status headings.

NUMBER= During Search/Modify, this message is displayed if the user is searching by number. It indicates the number of the record in the batch being searched for.

NUMBER MUST FOLLOW A MINUS SIGN Only a numeric literal can be signed, and the digit must immediately follow the sign.

NUMERIC OVERFLOW A number exceeds 13 digits.

00000

01 OUTPUT NOT LP When FD DIAG is running, this appears if Test 8, Dump Diskette Data, has been selected but the 01 Output device has not been set to Line Printer (LP).

OPEN RECEIVE
TIMEOUT Communications have not occurred within five minutes of data communication receive being initiated. Contact the remote transmission station verbally and determine the problem. Key CTRL/DISP and respond appropriately.

OPEN TRANSMIT
TIMEOUT Communications have not occurred within three minutes of data communication transmit being initiated. Establish voice contact with the remote station and determine the problem. Key CTRL/DISP and respond appropriately.

OPERAND NOT
RECOGNIZED Statement following an operator code is not valid. Can appear in numeric or nonnumeric statements.

OPERATOR
STATISTICS A data batch is closed. See individual entries for specific meanings. The information displays in this format:

OPERATOR STATISTICS

TIME USED=	RECORDS KEYED=
KEYSTROKES=	KEYSTROKES/HR=
ERRORS=	CORRECTIONS=
INSERTIONS=	DELETIONS=

OPERATOR TERMINAL
BUSY Displays during batch replay.

OUT OF MEMORY -
RETRY Insufficient memory is available for requested command.

OUTPUT BUF OVFL Sum of the output position and field length is greater than record length specified by format program.

OUTPUT BUFFER
OVERFLOW ON WRITE Sum of the output position and field length is greater than the record length specified by format program (occurs where output positions have been left blank).

OUTPUT END OF VOLUME: Fn An end-of-volume condition occurs when data is being written to a basic diskette (F1, F2, F3, or F4). Press the RESET key and respond to prompt sequence.

OUTPUT RECORD SIZE EXCEEDED Output position plus field size exceeds record size.

PPPPP

PARENTHESIS ERROR The right parenthesis is missing.

PERMANENT ERROR Hardware error has occurred.

PRESS REC FWD KEY TO CONTINUE Press the record forward key (REC-->) to display the next page of information or to continue processing.

PREVIOUSLY DEFINED Field name or label has been used more than once.

PREVIOUSLY DEFINED LEVEL More than one level has the same identification.

PRIMARY KEY REQUIRED During the sort-initialize process, the user specified a file without any keys.

PRINTER ERROR Background message. A printer error has occurred. Key CTRL/DISP and enter the message number, comma (,) and R for RETRY, or comma (,) and C for CANCEL.

PRINTER OFF-LINE Background message. The printer is off-line. This can happen if the paper jams, if the paper runs out, or if someone presses the ON/OFF LINE button. If due to one of the first two conditions, fix or load the paper, key CTRL/DISP, and enter the message number, comma (,), and R for RETRY, or comma (,) and C for CANCEL.

If the printer is out of paper, the condition has been corrected, and the retry command has been given, printing will continue with the next line. No data will be lost.

If someone presses the ON/OFF LINE button, the rest of the current record being printed will be output before the printer stops.

PRINTER PARITY ERROR Background message. An attempt to correct parity error has been made.

PROCEDURE NOT ALLOWED The stated procedure is not allowed in data definition division.

PROCESSOR MESSAGE WAITING During data communication using U100 protocol, a BELL character has been received from the host.

PROMPT REQUIRED A field is designated as a prompt field, but the procedure statement does not contain a prompt verb.

RRRRR

RCV DONE The data communication receive function is complete.

READ PRIMARY I/O ERROR Disk I/O error has occurred during reading of source batch. Search for the bad record and re-enter it.

READY - USE CMND KEY TO START WORK Press CMND key to display operating mode (work) menu. Six choices are listed (Enter, Verify, Search/Modify, Update, File Management or Supervisor).

REC BKSP INVALID AT FIRST REC Use of REC<-- key is not allowed in the first record of the batch. Press the RESET key and continue entering.

REC BKSP INVALID IN FILE MGMT MODE Use of REC<-- key is restricted with indexed files. Press the RESET key and continue entering.

REC FWD INVALID Use of Rec--> key not allowed beyond the last complete record. Press the RESET key and enter the record.

REC SIZE > BUF SIZE User attempted to transmit a file whose records are larger than the size allowed in the data communication protocol being used. Press the RESET key. Ensure that the record size and protocol buffer size are compatible. In certain cases, file record size can be reduced to provide compatibility.

RECEIVE REC LENGTH OVFL Record length received was too large and record will be truncated.

RECEPTION TERMINATED The data communication receive function has been terminated.

RECORD SIZE ERROR Record size in batch header to be reserved does not match required block length in indexed file header or sort header.

RECORD SIZE NUMBER OF KEYS These appear as indexed file status headings.

RECORDS KEYED= Operator statistics include total number of records created on disk from the initial work menu display through batch close.

RECOVER OR CANCEL Notifies user that file recovery is required.

RECOVER TO OPEN May occur after a system failure when a file was active. After WARM START, the file may need to be recovered before it is accessed.

RECOVERY DONE Status message stating that recovery is complete.

RECOVERY FAILED An attempt to recover a file has failed.

RECOVERY IN PROGRESS Displays during system recovery. If all disks do not get mounted, the system displays a message to indicate which disks were not mountable.

RECOVERY REQUIRED-CANNOT SAVE FILE The file cannot be saved until a recovery is performed. File may have been left active or damaged at a system failure. Open a program that uses file to execute recovery.

REKEY FIELD TO VERIFY After pressing the VCOR key, you must key-verify the field. Press the RESET key and do key verification. Also, if an error is made, such as entering a letter in a numeric field, and then entering the correction, this message appears. Press the RESET key and reenter.

REQUEST INITIATE CANCEL SUSPEND COMPLETE Indexed file status headings.

REQUIRED ENTRY A header or field checkbox requiring an entry has been left blank.

RESERVE IN PROGRESS System is busy reserving a disk as instructed by the Supervisor.

RESERVED WORD ERROR User has tried to compile a format program that contains the words RESERVE or SAVE outside of a comment, label program, format name, or quoted string. These words are reserved for use in sub-batch balancing.

RESOURCES NOT AVAILABLE FOR COMMAND MGR System is unable to acquire necessary resources for command manager task. Press RESET. Wait until resources are available or close down certain system activities.

RESPONSE INVALID An invalid response to a prompt.

RESTART INVALID WHEN XMT ACTIVE User attempted a CONTINUE command on reception or transmission while a transmission was already active. Press the RESET key. The transmission function will continue without interruption.

RESTORE CHECK SUM ERROR A checksum error exists on SAVE tape.

ROLL POINTER EXCEEDED During a compile, more than 4095 characters exist in the prompt field storage of the format program. Too many prompt fields. Compilation will terminate. Examine format.

SSSSS

SAME AREA CONFLICT User has referenced a file in more than one SAME RECORD AREA clause.

SEARCH ARGUMENT NOT FOUND May appear when using the Search/Modify mode.

SECTOR LENGTH ERROR The physical record length is not blank for BDE or 3 for extended diskette.

SELECTED INITIATION INCOMPLETE Memory or device resources are not available for autolog or autosave. Retry when resources are available.

SEQUENCE ERROR Data is not in sequence in a file specified as sequential.

SINGLE MT BUFFER ONLY User tried to use double buffering when single buffering was specified.

SORT FILE NOT FOUND Batches could not be found in sequential file header or indexed file header.

SORT INDEX ACTIVE...
COMPLETE...
PARTIAL... Indexed file status headings.

SORT KEY POSITION ERROR During the Sort-Initialize process, the key record position must be specified as an odd number between 1 and 999.

SPACE NOT ALLOWED This field is not formatted to allow blanks. Press the RESET key. Reenter or bypass the field and note the problem.

SP n UNAVAILABLE Station printer n is busy.

SP n OFFLINE Station printer n is off-line. Note message number. Press the RESET key. Place printer online. Key CTRL/DISP, enter message number, and C for cancel or R for retry.

SPOOL DIRECTORY END-OF-FILE System has reached the end of the spool directory.

SPOOL OFF CMND BUSY The spool-off function is being used by another terminal. Only three spool-off directories can be active concurrently.

SPOOL WITHOUT PRINTER During OPEN OUTPUT processing, a printer is requested but is not available or a format program not using the phrase SHARE-O has opened the file.

 Respond n,C to cancel the OPEN process (returns a status of "UNAVAILABLE RESOURCE" to the program).

 Respond n,R for retry; the file is spooled to disk until the printer is available and another OPEN OUTPUT command is encountered.

SUBPROG ARG MUST BE UNIQUE DUMMY NAME A dummy argument name cannot be a reserved word or register.

SUBPROGRAM VERB ERROR The verb specified is not allowed in subprograms. See the **System Supervisor's Reference Manual** for a list of allowed COBOL verbs.

SUBSCRIPT ERROR A subscript must be an arithmetic expression or numeric literal.

SUPERVISOR TERMINAL ACTIVE A user tries to enter the Supervisor Mode while another user is already active in that mode. Also displays during certain Supervisor functions.

SYNTAX ERROR The procedure language is used incorrectly.

SYSTEM DISK 90 PERCENT FULL The system displays this message at the terminal every 60 seconds when the system disk is over 90 percent full. Suspend work until supervisor makes disk space available (by saving or writing batches to tape, then deleting those batches from the disk).

SYSTEM ERROR (number) An unrecoverable system error has occurred. Note the error number and find the explanation in F.6.

SYSTEM OFF - TURN OFF ALL POWER SWITCHES This appears when the Supervisor POWEROFF command is executed. When the operator is finished, the equipment must be turned off (i.e., keystation, printer).

SYSTEM SOFTWARE ERROR If FD DIAG is running, this appears when the return from the system routine GET DCB PNTR has the Z condition code set.

 This message also appears when one of the modules FD PHY IO or FD MOUNT cannot be found on disk.

TTTTT

TAB OVERRUN ERROR Displays during the data communication receive mode when a mismatch exists between data records and the horizontal tab master record. Press the RESET key. Verbally contact the transmitting station to resolve the problem.

TAPE BLK LENGTH ERROR During a long record transmission in data communication, a block containing less than 14 characters was received. If a tape with IBM VLR format is being read, this message indicates that the physical block length does not coincide with the block length found in the block header.

TAPE BLOCK SIZE ERROR The number of records in a physical block on the tape being read does not match options specified. The block factor times the record length (in bytes) must be no less than 14 and no greater than 32,758.

TAPE DOES NOT MATCH OPTIONS The tape mounted does not match options specified in the format program.

TAPE FILE NO=XXX, BLOCK NO=XXXX System displays the file number and number of the last successfully transmitted block in tape file.

TAPE FUNCTION CODE ERROR An attempt has been made to perform an invalid tape I/O function while using IBM VLR tape format.

TAPE HARD ERROR The tape has a physical defect or needs cleaning by industry-standard procedures.

TAPE LABEL ERROR The label program specified and the label on tape do not match.

TAPE RECORD SIZE ERROR Tape error message. Record size in batch header for reserve does not match IFH or sort header record size.

TAPE NOT ON-LINE The tape unit is off line. Check the unit control panel.

TASK ABNORMAL END Task was not completed normally.

TASK ABORTED—CC LINE ERROR An unrecoverable communication line error has occurred. Press the RESET key. Verbally contact the other station and, if possible, reestablish data communications. A checkpoint was written.

TASK BUSY, COMMAND CANCELLED A task that the user requested is already in use. Press the RESET key and wait until the function is available for retry.

TASK COMPLETED A background task has been completed.

TASK INITIATED The task specified has begun.

TASK NOT
IMPLEMENTED The task referenced has not been implemented.

TERM=XX
TIME=XXXX The terminal number and time of day are displayed
on the message line of the screen if the DISPLAY
key was pressed and T was typed while in the
Enter mode.

TERMINAL ACTIVE A terminal is still active at the time of
attempted poweroff.

TERMINAL BUSY System is processing a previous command or
performing a batch replay. Wait until the
message disappears, then proceed.

TERMINAL NOT
ACTIVE The terminal designated to receive messages is
not active. Also, the supervisor has attempted
to monitor or close a terminal or station printer
that has not been turned on.

TERMINAL SIGNED
ON Sign-on record has been sent and the communication
link established.

32 LEVELS
EXCEEDED The format program has too many levels. It must
have no more than 32 program levels. Modify and
recompile format program.

TIME USED= Operator statistics include the amount of time a
terminal was active from initial work menu
display through batch close.

TOO MANY RETRY
S Too many Negative Acknowledgments (NAKs) have
been transmitted during data communication.

TOTAL REC= Batch statistics include the total number of
records stored on disk for a given data batch.
The count reflects record inserts and deletes.

TOTAL REGISTERS
USED EXCEEDS
AVAIL AU All registers referenced in the format must not
exceed 1334 bytes total.

TRACK ERROR
THRESHOLD
EXCEEDED More than two cylinders on a diskette are found
to be defective.

TRANS TABLE MISSING Selected translation table is not available in system.

TRANSMISSION TERMINATED Transmission has been terminated.

TWO SIDED DISKETTE ERROR The user attempts to initialize a 2-sided diskette in BDE format (must have a 1-sided diskette).

UUUUU

UNABLE TO LOAD COMM CNTL A probable hardware malfunction with the communication controller.

UNABLE TO RUN TEST--CC ACTIVE Occurs during TEST command. There is an on-line diagnostic test for each device and this one is related to communication controller being busy, so test cannot select this device.

UNDEFINED Field name in procedure has not been previously defined.

UNDEFINED LABELS: Label referenced by a procedure has not been defined.

UNDEFINED LEVEL: Level referenced by a SELECT LEVEL statement has not been defined.

UNDEFINED NAMES: Field name referenced in a procedure has not been defined.

UNMARK OR DELETE? User tried to unmark a modified file.

UNRECOVERABLE DISK ERROR Major disk error has occurred.

UNRECOVERABLE DISK ERROR-- BATCH CANCELLED An unrecoverable disk error has occurred. Batches erroneously flagged as active on the disk are inaccessible.

UNRECOVERABLE DISK ERROR, JOB CANCELLED An unrecoverable disk error has occurred. Batches erroneously flagged as active on the disk are inaccessible.

UNRECOVERABLE FD ERROR, JOB CANCELLED During a read or write to a diskette, a diskette error occurred.

UNSOLICITED DATA RCV Dn The host computer sent data to D1, D2, D3, or D4 without requesting permission, or after the data stream had been suspended.

UPDATED REC= Batch statistics include the number of records processed in the Update mode.

USE CONFLICT The name used for file status, prime or alternate key, or data division header record name is also used elsewhere.

www

VERB ERROR Verb is missing or incorrect at the beginning of a statement.

VERIFIED REC= Batch statistics include the number of records processed in the Verify mode.

VERIFY DISKETTE TYPE ERROR The diskette type specified does not match the diskette type encountered.

VERIFY MISCOMPARE The character entered does not match the character in the record. Press the RESET key and enter the correct character.

VOL1 STRING ERROR The ID field of the volume label does not contain the four characters VOL1.

VOLUME LABEL HARD ERROR The system detects a hardware error while reading or attempting to write the diskette volume label sector.

VOLUME STANDARD ERROR Position 80 in a standard field on a BDE diskette is not W or that of an extended diskette is not P.

VTOC IO ERROR The system cannot read or write the volume table of contents.

www

WAIT--SYSTEM PERFORMING SELF TEST System is performing self-test before recovery.

WARNING...DATA
CHECK FROM CC Displays during data communication receive. A software error probably exists at the transmission station. Press the RESET key. Retry occurs automatically at the transmission station. If the message displays repeatedly, verbally contact the other station and resolve the problem.

WARNING - EXCEEDS
RECOMMENDED BLOCK
LENGTH A possible tape error because the block length was exceeded.

WORD TOO LONG A nonreserved word in a format program statement exceeds eight characters.

WRITE HEADER I/O
ERROR Compiled file header could not be written, and compilation was aborted. Recompile.

WRITE SUBFILE I/O
ERROR Disk I/O error occurred during write of compile and compilation was aborted. Recompile.

WRONG VSN During diskette volume switching, the Volume Sequence Number (VSN) of an input diskette does not have the correct value (one greater than previous volume). Replace with diskette that has next higher volume.

XXXXX

XDUP KEY IS
INVALID The field is not designated for auxiliary duplication. Press the RESET key and enter or bypass the field.

XMT BEGUN A transmission task has been initiated. (The first job name and batch number in the directory are also displayed).

XMT DONE A transmission task has been completed or interrupted. (The last job name and batch number in the directory are also displayed.) If the multiple end-of-file option has been selected, transmission of each succeeding list continues without supervisor intervention.

ZZZZZ

ZERO NOT ALLOWED Zero is not allowed in this field.

F.6. SYSTEM ERROR MESSAGES

System errors occur because of software anomalies or hardware failures that cause the software to malfunction. When a system error occurs, the system is halted and can be restarted only by initiating a warm or cold start.

All keyed data is usually recoverable when the system enters the error recovery mode. Recovery is not possible, however, when the information on disk is destroyed. If the system error resulted from a permanent hardware malfunction, the hardware must be repaired before recovery is initiated.

When a major system error is detected by software, the following message is displayed at all keystations and CPU operation is terminated (XX is the number of the error).

SYSTEM ERROR XX

At this point, a dump program which is loaded from disk can be invoked by entering a special character sequence from any keystation. This program displays the register contents at the time the error occurred and prompts the operator for memory and/or disk dumps. When the tasks are completed, the dump program performs a disk load to restart the system. The memory and disk dumps can then be analyzed, using the software diagnostic package.

The system error dump procedure is detailed in F.10.

F.6.1. Disk Error Recovery

Some basic conventions followed by the system-disk file system are as follows:

- Every file on disk is referenced by the volume table of contents (VTOC).
- All files on disk (except cylinder 0, the system messages, and overlay type 1 files) are linked allocation unit (one-quarter track) chains.
- All linked allocation unit chains contain file headers which consist of 11 words including the VTOC entry for the file. Each file is a part of a larger data chain which, in turn, contains a file header at its start. For example, format programs, subprograms, tables and labels combine to make one data chain.

- Each allocation unit on any available disk has an associated list in the main memory indicating the allocated/released status of the unit.

In case of system failure, recovery consists of:

1. Obtaining an initial disk map of allocation unit availability.
2. Reconfiguring the disk map to indicate actual usage/availability of allocation units according to the current VTOC.

Initially, the volume label is read from the disk. This volume label contains the initial disk map and the absolute disk address of the VTOC. If the volume label is missing, the system must be reloaded.

Each file that is part of a data chain in the VTOC is individually inspected, link by link, by reading the first sector of each allocation unit link and making it unavailable for further allocation by the system. Files that are not part of a data chain were previously made unavailable during the construction of the initial disk map. The linking is then conducted forward through the chain until the end is reached. This procedure is followed for every file in the VTOC. For batches that were open at the time of system failure, each data record of the batch is read and, when an end of file is reached, the batch is closed with "partial" status. This procedure allows the recovery of all data that had been keyed up to the time of failure except the data in memory at the time of failure. The amount of data lost may be up to 84 characters or one record, whichever is greater.

F.6.2. System Errors

A list of system errors with a brief description of each error follows. Descriptions include the meaning of each message, response needed, and name of the module from which the error message originated. The meaning given for each system error may not be very helpful. Most software-related system errors mean basically the same thing--there is garbage in memory. The different system error numbers simply tell what process was taking place when the garbage was detected, suggesting where one might logically **begin** looking for the actual problem. In most cases, only a memory dump analysis will reveal the actual cause of the invalid memory contents.

System Error Number	Response	Meaning	Originating Module
2	Do a memory dump. Warm start the system.	No room on system control list. Could not move system control list to get more space to save queue or stack.	Executive
3	Do a memory dump. Warm start the system.	Duplicate in enqueue. Request was made to place a terminal in a waiting queue, but terminal was already in the queue. This type of error could result if the hardware is not responding properly to messages issued by Physical I/O.	Physical I/O (PHY IO 3)
		An Input/Output Control Block (IOCB) was passed via a call to Physical I/O. Physical I/O detected that an I/O operation is still in progress using the designated IOCB. Current call is invalid.	Physical I/O (PHY IO 3)
4	Do a memory dump. Warm start the system.	Invalid device type. Device type code has no device handler associated with it.	Physical I/O (PHY IO 3)
5	Do a memory dump. Warm start the system.	Invalid device type. CALL was made to GET DCB PNTR with a terminal or station printer device type, but device code number given was 0 or 1, invalid for devices on the coaxial cable.	Physical I/O (PHY IO 3)
6	Do a memory dump. Warm start the system.	Bad IOCB pointer. Input/output queuing process has found a bad IOCB pointer on the device control block queue.	Physical I/O

System Error Number	Response	Meaning	Originating Module
7	Do a memory dump. Warm start the system.	Bad IOCB pointer. Indicates a bad IOCB pointer on the channel control block queue.	Physical I/O
10	Do a memory dump. Warm start the system.	Overlay TTI is zero. On return to a resident module. Tenant Table Index (TTI) is found to be missing.	EM MGR
11	Do a memory dump. Warm start the system.	Overlay segment not found during LTO execution. A link from one overlay to another cannot be completed because destination overlay is nonresident. This occurs during execution of an LTO instruction.	File Control
12	Do a memory dump. Warm start the system.	Overlay segment not in memory. A link from one overlay to another cannot be completed because destination overlay is not found in memory. This occurs during execution of a CFO or LTO instruction.	File Control or Editor
13	Do a memory dump. Warm start the system.	Return to nonexistent overlay. Upon return to File Control, source overlay address is found to be missing.	File Control
14	Do a memory dump. Warm start the system.	Swap-in overlay cannot be fetched. An attempt was made to swap an overlay on top of an overlay that cannot be found. This error occurs during execution of an RAL instruction.	File Control
15	Do a memory dump. Warm start the system.	Link to nonexistent overlay. No base address exists for a transfer of control to an overlay.	File Control

System Error Number	Response	Meaning	Originating Module
16	Do a memory dump. Warm start the system.	Segment was not fetched. Attempt was made to build an overlay pointer for an overlay not fetched by requesting terminal. This occurs during execution of an LTO or CFO instruction or during a special overlay call.	File Control or Sys Sub-routines
17	Do a memory dump. Warm start the system. Do not cancel (bypass) at the same point until the problem is resolved.	Link to cancel pointer overlay not in memory. An attempt to link to cancel pointer was made, and overlay ID was found to be zero. This occurs during a Supervisor cancel.	Sys Sub-routines
18	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	Cannot start up. Volume label in the system resident area cannot be written or read. Initial or used disk map cannot be read. Fetch of an indexed file overlay failed. This error is due to hardware or controller failure during a disk access.	Startup, File Control or Sys Sub-routines

System Error Number	Response	Meaning	Originating Module
19	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	Cannot start up. An allocation unit chain cannot be recovered because one of its links cannot be read (probably an unrecoverable hardware disk error).	Startup
		System overlay index table is full.	File Control
		Error on MAP I/O during INIT ALL UNUSED AUS, or error on PROCESS RCB TO DISK.	Startup
20	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	During system disk recovery, an I/O error occurred during attempt to delete scratch files from MVTOC.	Startup
		Disk VTOC unreadable. Attempt to read Volume Table of Contents (VTOC) failed.	Disk Load
		Link to the delete file overlay failed. This is typically a hardware-related error.	Index Files

System Error	Response	Meaning	Originating Module
21	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are . required. Refer to the Software Diagnostic RECOVER command.	VTOC error. System could not locate a key on an alternate disk or could not write a new VTOC page after an entry deletion. I/O error during MERGE of VTOC. I/O error on MAP update to disk or memory during LTC.	Delete File
22	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	An error occurred on initial READ of system Label AU (LAU).	

System Error Number	Response	Meaning	Originating Module
23	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	I/O fail on disk map READ or WRITE.	
24	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	Attempt to allocate from a disk map found AU had already been allocated, indicating that two files link into the same AU. This occurs only during MOUNT or STARTUP.	

System Error Number	Response	Meaning	Originating Module
25	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	Unknown disk error. Unspecifiable error exists on disk.	Checkpoint
26	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	MVTOC delete on startup. Failure occurred on READ/WRITE of DELETE ALL KEYS FOR ALT DISK.	

System Error Number	Response	Meaning	Originating Module
27	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	Out of room in startup: No room for TWA, disk buffers, or overlays. This can also indicate a WRITE LAU I/O error, or a READ MAP error on the platter.	
30	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	This indicates that an extended memory address has been specified as the transfer address for a device that does not provide indirect I/O to EM.	Physical I/O

System Error Number	Response	Meaning	Originating Module
31	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	An EM data transfer request exceeds the size of the indirect I/O buffer.	Physical I/O
32	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	The specified TWA transfer address is not in the actual TWA range.	Physical I/O

System Error Number	Response	Meaning	Originating Module
33	Do a memory dump and a disk dump. A warm start may recover the system. If not, special data recovery procedures and a cold start are required. Refer to the Software Diagnostic RECOVER command.	An invalid TWA number of 0 was specified.	Any
40	Do a memory dump. Warm start the system.	Invalid TTI. Tenant Table Index (TTI) is zero or greater than the maximum allowable.	EM MGR, Executive, or Format Loader 2
46	Do a memory dump. Warm start the system.	Invalid external ID. External ID of an overlay does not match old external ID, indicating the overlay was released and a new one loaded in its place.	EM MGR
47	Do a memory dump. Warm start the system.	Invalid overlay. User count of an overlay is zero, indicating overlay was unintentionally released.	EM MGR
X 51	Do a memory dump. Warm start the system.	Nonexistent Input/Output Control Block (IOCB). System tried to initialize the X register with address of an IOCB that has not yet been allocated.	Logical I/O

4/27/84
Concluded Comm. TEST
10 MIN. BEFORE

System Error Number	Response	Meaning	Originating Module
52	Do a memory dump. Warm start the system.	Illegal post priority argument. This error is generated if an illegal post priority is found.	Executive
58	Do a memory dump. Warm start the system.	Bad status returned when STARTUP calls the disk driver to read status of system disk. Cause may be disk being powered off, a failure during the disk controller self-test process, or some other hardware malfunction.	STARTUP
7X	Do a memory dump. Warm start the system.	Buffer pool X not allocated. Attempt was made to fetch a buffer from pool X, but pool X was not previously allocated. If X is greater than 9, message displays as 7:.	Sys Sub-routines
8X	Do a memory dump. Warm start the system.	Task queue X overflow. The task queue is full. Attempt was made to enqueue another task, but system could not accept it. X may be 0, 1, 2, or 3 and indicates the priority of the task queue. A common cause of this error is using more terminals on the system than system was configured for.	Executive
89	Do a memory dump. Warm start the system.	CRT update table capacity exceeded. This error may be hardware-related: more terminals on system than SYSGENed or SMUX failure.	Executive

NOTE:

System errors 91 through 99 occur in response to interrupts. The interrupts are processed in the EXECUTIVE module. Some of the errors are caused by actual hardware problems, while most are caused by the software but detected by the hardware.

System Error Number	Response	Meaning	Originating Module
91	Check the MCU cabinet for proper ventilation. Warm start the system.	Temperature alarm. Interrupt was received from the hardware indicating that MSU is operating at too high a temperature.	Physical I/O
92	Do a memory dump. Warm start the system.	Invalid program status word. Software error caused by invalid program status word.	Executive
93	Do a memory dump. Warm start the system.	Bus error. Software attempted to address out of the range of existing memory. This can be an application-caused software problem, but can also be caused by a hardware malfunction such as a failure in the XPU or undetected data transfer error.	Any module
94	Do a memory dump. Warm start the system.	Stack overflow. An interrupt was generated by the XPU indicating that contents of the hardware stack pointer were less than contents of cell FF. This is usually caused by too many entries in the stack	Physical I/O
95	Do a memory dump. Warm start the system.	Stack service request. Attempt was made to execute an instruction not recognized by XPU.	Physical I/O

System Error Number	Response	Meaning	Originating Module
96	Do a memory dump. Warm start the system.	Reserved instruction. Attempt was made to execute an instruction not recognized by XPU.	Physical I/O
97	Simultaneously press the CTRL and DISPLAY keys. Note the bus location played on the screen, because this is the address that caused the memory error. Warm start the system. Call Customer Engineer.	Bus error. Hardware interrupt generated by XPU indicating a bus error caused by a parity error within usable memory. This is generated by accessing a bad memory cell.	Any module
98	Do a memory dump. Warm start the system.	Overlay service error. This occurs on XPU because of overlay service errors. On the XPU2, this occurs because of a page fault.	Executive
99	Do a memory dump. Warm start the system.	Miscellaneous errors. Miscellaneous error cache for privileged instruction, Supervisor call, or trace or test panel interrupts.	Executive

F.7. TASK ERROR MESSAGES

Task errors are errors which occur during a particular task affecting only that task and not the entire system.

When a task error occurs, the following message is displayed at the offending keystation (NN is the error number):

TASK ERROR NN

The type of error can be determined by locating the displayed error number in F.7.2.

Usually a task error hangs up only the keystation that initiated the task. Occasionally, however, the system may hang up or the keystation may fail to lock.

F.7.1. Task Error Recovery

Task error recovery can usually be accomplished by performing an orderly CLOSE and WARM-START procedure. All keystations should be closed as soon as possible after a task error has occurred. If a memory dump is desired, the system dump procedures described in F.10 may be used. The memory dump may be taken after all keystations have been closed or during a system hang-up.

F.7.2. Task Errors

A list of task errors with a brief description of each error follows. Descriptions include the meaning of each error and name of the module from which the error message originated.

An asterisk (*) next to a task error indicates that more than one problem may be associated with that task error number.

Task Error Number	Meaning	Originating Module
2	Illegal wait post ID. Wait post identifier is greater than 7.	Executive
3	Illegal IOCB number. Input/output control block number is too large.	Executive
5	Illegal TWA word value. An invalid value has been found in terminal work area.	Test
6	Checksum error. System detected an incorrect checksum.	Test
7	Time exceeds limit. Timeout value was reached before completion of test.	Test
10	DCB not found. System was not configured for the device, or the input/output control block is bad.	Logical I/O, LIO 2780 or HI LIO
11	Device handler not found. Device number rung of the input/output control block is bad or input/output control block pointer is not set up correctly.	Logical I/O
*12	An illegal magnetic tape command was issued.	MT CMND
*12	Receive file not open. Data communication receive command was issued before receive file was opened or after the file was closed.	HASP LIO
*12	Transmit file not open. Data communication transmit command was issued before transmit file was opened or after file was closed.	HASP LIO
*12	Console-out file not open. Data communication output command was issued before console-out file was opened.	HASP LIO
*12	HASP not initialized. Any data communication command, except open-console, was issued before initialization was completed.	HASP LIO

Task Error Number	Meaning	Originating Module
*12	Device not open. During data communication, attempt was made to output or input from a device that is not open.	LIO 2780
*12	Receive file not open. SDLC receive command was issued before receive file was opened or after file was closed.	LIO 3770
*12	Transmit file not open. SDLC transmit command was issued before transmit file was opened or after file was closed.	LIO 3770
*12	Console-out file not open. SDLC output command was issued before console-out file was opened.	LIO 3770
*12	SDLC not initialized. Any SDLC command, except open-console, was issued before SDLC initialization was completed.	LIO 3770
13	Represents a disk error, but error bits are not set correctly to indicate what error. Problem may be hardware related, in which case disk drive or controller may be malfunctioning.	DISK LIO
14	File not open. System tried to input to or output from a disk file that is not open.	DISK LIO
*15	Illegal IOCB allocation. System tried to allocate an existing input/output control block.	GET IOCBS
*15	Duplicate IOCB. The same IOCB appears twice in user's IOCB table in TWA.	GET IOCBS
*15	No such IOCB. Attempt was made to initialize an IOCB that does not exist.	GET IOCBS
16	Incorrect logical function. Function code rung of input/output control block is not correct, or input/output control block pointer is bad.	MT RCD or MT SPECL

Task Error Number	Meaning	Originating Module
20	Illegal appendage. Attempted start overlay appendage with a nonexistent overlay identifier.	FILE CTRL
21	Illegal section identifier. A parameter is being passed to a routine to indicate which section of code to execute, and parameter exceeds number of sections.	MSG HDLR 2
22	No overlay on disk. Overlay identifier is bad; overlay does not exist; input/output control block is bad; or volume table of contents is bad.	FILE CTRL
23	Unsuccessful fixed buffer overlay fetch. Fixed buffer overlay cannot be fetched. SEGS FETCHED table has been exceeded.	FILE CTRL
28	Format error. An incompatible format was called by operator, or a program error exists in the attempt to store in a previous field.	RUNTIME
*29	IOCB out of range. Attempt was made to use an IOCB that is out of range of valid IOCBs for data communication.	DCOM XMT
*29	No subtask to start. Request was made to activate a data communication subtask when there were no tasks waiting to be started.	DCOM XMT
*29	Device code was not found in device table.	UH XMT
*30	Bad overlay release. System failed to release an overlay.	IFH HDLR
*30	An invalid response to a CMND message prompt was received.	MT PROMPT
*30	An incorrect protocol was specified by operator.	HASP LIO or LIO 3770

Task Error Number	Meaning	Originating Module
*31	Incorrect TWA ID. Incorrect terminal work area identifier was received.	INIT TWA2
*31	Shutdown error. Attempt was made to terminate communication while data stream was active.	HASP LIO
32	Incorrect TWA ID. Incorrect terminal work area identifier was received.	INIT TWA
33	DCB not found. False return from the get device control block pointer.	HASP LIO or LIO 3770
34	Logical receive buffer pool was not allocated before use.	HASP LIO or LIO 3770
*35	Transmit buffer count has gone negative.	HASP LIO
*35	Illegal logical function. Function requested was greater than the maximum supported logical function.	CONSOLE
*35	A bad TWA number was used when attempt was made to reset base address.	CONSOLE
*35	Transmit overlay missing or receive overlay missing. Diagnostic error used during system development. Actual error and message should never occur in the field.	CONSOLE
*35	Startup already performed. Startup was attempted more than once.	STARTSTOP
36	Receive buffer count contains a negative value.	HASP LIO
51	MANAGER overlay has discovered an unrecoverable error during station printer spooling.	MANAGER

Task Error Number	Meaning	Originating Module
52	RELEASER overlay has discovered an unrecoverable error during station printer spooling.	RELEASER
61	Illegal overlay index into SCA appendage. Overlay index sent to the error appendage for the synchronous communications adapter (SCA) is bad (overlay does not exist).	SCA ERR
62	DCB not found. System was not configured for device during SYSGEN or the input/output control block is bad.	SCA LDR, HASP LIO, SCA ERR, or LIO 3770
63	A buffer is left after SCA close. System did not deallocate all data communication buffers after a close.	SCA ERR
64	SCA not loaded. Data communication software was not down-loaded from system to communications controller.	SCA ERR
65	Bad SCA logical input/output entry. A bad function code rung exists in input/output control block or a bad input/output block pointer at synchronous communications adapter level.	LIO 2780, HASP LIO, or LIO 3770
*66	No DCB linkage. No device control block linkage has been established.	SCA PHY IO
*66	Logical Host Interactivity error. Error cache for HI logical errors.	HI LIO
67	Bad SCA physical input/output entry. A bad function code rung exists in input/output control block or a bad input/output control block pointer at physical input/output lever.	SCA PHY IO

Task Error Number	Meaning	Originating Module
*68	Bad magnetic tape record length. Either a mismatch of an allocated buffer and record length indicated in the input/output control block, a bad input/output control block, or a bad IOCB pointer.	PHY IO 2
*68	No SCA pool. Could not get background pool area to execute data communications.	SCA PHY IO
70	Send message queue empty. System attempted to send a message, but no messages were waiting in queue. System control list may be bad.	SYS SUBRS
71	Reply queue empty. No reply is waiting in reply queue. System control list may be bad.	SYS SUBRS
*90	IOCB number is out of range. Either system detected an input/output control block number that exceeds the maximum number allocated or system tried to start a subtask when no subtasks were waiting to be started.	DCOM RCV
*90	Receive error. Represents an error cache for UH receive errors. Either UH RCV active task count is invalid, open input failed, or open output failed.	UH RCV

An asterisk (*) next to a task error indicates that more than one problem may be associated with that task error number.

F.8 FILE STATUS CODES

A format program may include FILE STATUS clauses in the following form:

FILE STATUS IS data-name

Whenever this clause is specified, the system returns a two-digit numerical code in data-name after each statement that implicitly or explicitly references a given file. Check this status code to avoid errors that are program controllable. File status codes are explained in F.8.1. and F.8.2.

F.8.1 File Management Status Codes

The following list of status codes includes those pertaining to general and sequential file management and the report writer feature. Status codes specific to the host interactivity file management feature are explained in F.8.2.

Status Code	Status Result	Meaning	Suggested Response
00	None	Successful I/O.	None
02	None	Successful, but duplicate key detected.	Appropriate action.
05	None	Successful, but other file user permanent error.	Close; fix hardware and delete/restore file if necessary.
06	None	Operation was successfully completed, but system discovered a discrepancy between size at the record area in report writer format and size of logical record. Some data may have been lost.	Check for lost data and correct the discrepancy in record size.
10	Invalid key or at end condition	An end-of-file condition has been detected during an input operation.	File end action.

Status Code	Status Result	Meaning	Suggested Response
21	Invalid key or at end condition	Invalid key--out of sequence record error.	Check data or programming.
22	Invalid key or at end condition	Invalid key--unexpected duplicate key.	Check data or programming.
23	Invalid key or at end condition	Invalid key--indexed record not found.	Check data or programming.
30	Invalid key or at end condition	Permanent error.	Close; fix hardware and delete/restore file (if necessary).
90	Invalid key or at end condition	Unsuccessful I/O operation. Reasons for the failure are unknown.	Check programming or operations.
91	Invalid key or at end condition	Unsuccessful--a file opening was not completed because some parameters associated with the file are inconsistent or invalid.	Check programming or operations.
92	Invalid key or at end condition	Unsuccessful--operation was not completed because operation itself was invalid, as when a format program tries to open a file that is already open.	Check programming or operations.

Status Code	Status Result	Meaning	Suggested Response
93	Invalid key or at end condition	Unsuccessful--operation was not completed because operation was not valid. For example, this status would be returned if a file was assigned to tape unit 2 (T2) and the system only has one tape unit.	Check programming or operations.
94	Invalid key or at end condition	Unsuccessful--an input or output operation failed because system could not get at least one system resource needed to complete the operation. For example, a needed device may be busy or not ready; or buffers may not be available.	Close; make the resource available.
95	Invalid key or at end condition	Unsuccessful--file management record temporarily locked.	Retry operation or check programming or operations.
96	Invalid key or at end condition	A severe error has occurred during an I/O operation. Condition of file is uncertain; batches associated with file may still be in the active state.	Perform warm start as soon as possible to remove file from active state.

F.8.2. Host Interactivity Status Codes

The following status codes apply to the host interactivity file management feature. Refer to F.8.1 for all other file management codes.

Status Code	Meaning
00	Operation was successfully completed.
30	A sending operation was tried 15 times without positive acknowledgment from host computer.
91	Occurs only after RECEIVE or SEND statements. After a RECEIVE statement, code indicates that the host computer has transmitted a record containing more than n characters, where n is the record size specified in the Data Definition Division. Text length will be set to n and received record will contain the first n characters of the transmitted record. After a SEND statement, code indicates that the format program attempted to write a record larger than n, and the operation was canceled.
92	Attempted operation was considered invalid by the host interactivity subsystem. Operation was either a RECEIVE or SEND operation referring to an unopened file, or it was a redundant ENABLE or DISABLE statement.
94	Host interactivity subsystem is either unloaded or being unloaded, or data communications link failed before operation was executed.
96	Operation was canceled for one of three reasons: <ol style="list-style-type: none">1. Format program had a timeout.2. Operator cancelled the operation.3. Communication link was broken after a RECEIVE or SEND statement was executed.

F.9. SYSGEN ERROR MESSAGES

When the system is being configured, errors are under control of the SYSGEN logic modules (see Appendix G). At the time a major error is detected by software, the message is generated (XXX is the error number):

SYSGEN ERROR NO: XXX

Whenever a SYSGEN error occurs, the process being attempted at the time of the error is stopped. The operator will not be able to cause SYSGEN to resume processing at the point of interruption. If the intercepted task is to be completed, it must be restarted. A list of SYSGEN errors, their causes, and in some cases, suggestions for resolving or avoiding the error, follows.

F.9.1. SYSGEN Errors

**SYSGEN
Error
Number**

Description

- 01 Disk allocation error. Attempt made to allocate space for a software module failed. Indicates excessive bad spots on the system platter. Contact Customer Engineering immediately.
- 02 Disk read error. Attempt to read a module from disk has failed 10 times. Contact SPERRY Customer Service representative.
- 03 Tape read error. Attempt to read a block from tape failed 10 times. Clean tape drive with alcohol. Restart. If problem continues, try another tape. Contact Customer Engineering.
- 04 Disk write error. Attempt to write a module to disk has failed 10 times. Probable hardware malfunction. If it occurs during system generation, restart. Mandatory to INITIALIZE SYSTEM DISK=YES. Failure to initialize disk prior to GTS command will allow system to allocate modules to possible bad spots.

**SYSGEN
Error
Number****Description**

- 05 Binary loader error. Reloader routine has detected errors reading binary modules. Possible tape drive malfunction. Clean tape drive with alcohol and restart. If errors continue, try another SMB tape.
- 06 Module type error. Resident module ID which is invalid has been detected. Scrap SMB being used and get a new one. Contact Customer Support Services (CSS).
- 07 Maximum overlay ID error. Overlay ID has exceeded allowable maximum for this system version. Scrap the being used and get a new copy. Contact CSS.
- 08 Resident overflow. Loading of resident modules during cold/ warm start has destroyed Perdebug/IOCTL modules. This should never occur. Create a new cold start tape from a known good SMB. Contact CSS.
- 09 Maximum module length error. Reloader routine has constructed a module too large for buffer. Scrap SMB being used and get a new copy.
- 10 VTOC overflow. VTOC has overflowed its maximum allowable length. Should occur only during system generation. Scrap SMB being used and generate system from a known good copy.
- 11 SYS MSG error. System messages contain incorrect values and message file cannot be built. Scrap SMB. Restart with a known good copy.
- 12 Not currently used.
- 13 Write tape error. Attempt to write a block to tape has failed 10 times. Clean tape drive with alcohol. Mount a new scratch tape and restart tape write. If problem persists, contact Customer Engineering.
- 14 Record checksum error. Checksum of last tape record read is incorrect. Hardware did not detect an error. Scrap SMB being used and restart using a known good copy.
- 15 Not currently used.

**SYSGEN
Error
Number**

Description

- 16 LAU not found. Last allocation unit containing volume label, system flags, and disk maps cannot be found. Probable disk malfunction.
- 17 SMB format error. SMB being read in does not contain proper files/file marks. Possible tape drive malfunction. Possible bad SMB.
- 18 Module directory overflow. Number of modules being added to the directory has caused buffer to overflow. Scrap SMB or cold start tape.
- 19 SYSGEN overlay missing. Overlay necessary to the SYSGEN program cannot be found. Scrap SMB being used and restart with known good copy.
- 20 SYS MSG logical error. System message cannot be built in correct form. Scrap SMB and restart using a known good copy. (Also fixed overlay ID error.)
- 21 Correction buffer overflow. Patch added by means of CMT has exceeded buffer size allotted. Scrap SMB and restart with a known good copy. If a new patch has been added by means of CMT, try again. If failure persists, contact CSS.
- 22 Buffer key overflow. Attempt to copy the SYSGEN file from tape to disk has exceeded buffer size. Scrap SMB and restart using a known good copy.
- 25 Bad disk status error. Disk is not yet ready and an attempt to TAPE LOAD or DISK LOAD was made.
- 26 Incompatible disk type error. GDS was for a different disk type than type available on the system.
- 27 Disk load not allowed error. Contents of disk indicate that a system does not yet exist or that a warm tape load is in progress and system status on disk is uncertain.
- 28 Warm mount abort error. Space for warm tape load was 32 AUs or less on disk, and operator aborted warm tape load.

**SYSGEN
Error
Number**

Description

- | | |
|----|--|
| 29 | I/O error on LAU. I/O operation on last AU (LAU) has failed in some way. |
| 30 | IO error on disk map. I/O operation on disk map has failed in some way. |
| 31 | Message file overflow. SYS MSGS file overflowed allocated disk space. To fix, modify SYS MSGS to request an additional cylinder of disk space (in-house diagnostic). |

F.9.2. IOCTL Errors

Besides SYSGEN errors, IOCTL errors can occur during SYSGEN. These errors are hardware-related and are detected by the I/O Control Program. If an IOCTL error occurs, the following messages will be displayed.

IO ERROR U=XXXX F=YYYY X=ZZZZ

Table F-1 lists the IOCTL codings and the meanings of these values.

Table F-1. IOCTL Error Coding (Part 1 of 2)

U=XXXX (Unit Number)	F=YYYY (Function)	S=ZZZZ (Status)
0001: MTC 0, tape drive 1 0002: MTC 0, tape drive 2 0003: MTC 0, tape drive 3 0004: MTC 0, tape drive 4 (See note)	0003: read 0004: write 0005: rewind 0006: write file- mark 000B: read status 000C: backspace re- cord 000D: backspace file 000E: read without transfer 000F: skip file	0100: Nonexistent unit error occurs if the specified unit is not present in the present hardware con- figuration (software mal- function). 0200: Unresolvable request errors occur if the user program specifies a unit number which is not defined (soft- ware malfunc- tion).
0006: keystation	0001: input one character 0002: output one character 0003: input char- acter string 0004: output char- acter string	0400: Unrecoverable error occurs if an opera- tion results in an error and ten re- tries could not solve the problem.
0007: RDC 0, disk drive 0, bottom platter 0008: RDC 0, disk drive 0, top platter 0009: RDC 0, disk drive 1, bottom platter 000A: RDC 0, disk drive 1, top platter	0003: read 0004: write 0005: restore 0006: initialize 000E: read without transfer	

Table F-1. IOCTL Error Coding (Part 2 of 2)

U=XXXX (Unit Number)	F=YYYY (Function)	S=ZZZZ (Status)
000B: RDC 0, disk drive 2, bottom platter 000C: RDC 0, disk drive 2, top platter 000D: RDC 0, disk drive 3, bottom platter 000E: RDC 0, disk drive 3, top platter	0004: output	0800: Unit not ready error occurs if the speci- fied unit is not ready or if the write ring is not installed to perform a tape write. After a character is entered the program will attempt the operation again.
0005: Station printer 0010: ASCII line printer 0011: EBDIC printer		

NOTE:

The following unit modification bits are added to the tape unit numbers to further describe the tape unit:

- 0800: even parity, 7 track
- 1000: even parity, 7 track
- 2000: no conversion mode, 7 track
- 4000: phase encoded read adapter is used
- 8000: read after write drive

F.10. MEMORY DUMP PROCEDURES

To accurately determine the cause of a system error, task error or system hang, it is valuable to examine a "snapshot" of memory or disk contents as they appeared the instant the error occurred. By performing a dump procedure, the contents of memory or disk at a specific moment are copied onto tape. This dump tape can then be printed out and analyzed by SPERRY software support personnel, and a solution devised.

F.10.1. System Error Dump Procedure

This procedure is useful only in the case of an explicit system error as described in F.6.

1. Mount a scratch tape with a write-enable ring in the master tape drive.
2. On any keystation, press the LCA, RESET and DISPLAY keys simultaneously. The contents of various memory locations will be displayed on the screen.
3. Write the displayed numbers (which are in hexadecimal) on the software user report.
4. Type Y in response to the "MEMORY DUMP?" prompt. The contents of memory will be dumped to tape.
5. When the tape has stopped moving, "DISK DUMP?" is displayed.

If a disk dump is required, type Y.

If a disk dump is not required, press the FIELD REL key. Only the system portion of the disk is dumped using this procedure.

If a SAVE of all disk contents is required, use the procedures described in F.10.2 instead.

6. Remove the tape from the drive and label it.
7. Restart the system, using either cold or warm start procedures.

If part of the disk-resident dump routine was destroyed by the cause of the system error, the preceding procedure may not work. In this case, a memory dump tape prepared previously from the software diagnostic tape must be used as described in the following paragraph.

F.10.2. Alternate Dump Procedure

Use the alternate dump procedure when one of the following conditions occurs:

1. A task error or system hang occurs.
2. The system error dump procedure does not work.
3. The nature of the problem suggests that an exhaustive dump of all disk contents is required.

A specially prepared memory dump tape is needed. It should have been previously prepared using the following steps:

A specially prepared memory dump tape is needed. It should have been previously prepared using the following steps:

1. Load the software diagnostic tape by means of a tape load.
2. Mount a tape with a write-enable ring on any tape drive, keeping in mind that the memory dump tape must be compatible with the master tape drive of the system.
3. In response to the "ENTER COMMAND" prompt, type:

MAKEMD=X (,X) (,X) (,X)

Where:

X

is the drive number of each tape mounted in step 2.

The arguments in parentheses are optional.

A memory dump program is then written on the beginning of each tape. It is recommended that several of these tapes be made and kept available onsite.

To perform a dump using a memory dump tape, proceed as follows:

1. While the system error message is still displayed, mount the memory dump tape with a write-enable ring on the master tape drive.
2. Press the TAPE LOAD switch and wait for tape movement to stop.
3. Remove the memory dump tape from the tape drive and label it.
4. If a disk SAVE is not required, press the DISK LOAD switch and perform either a warm or cold start.
5. If a disk SAVE is required, proceed as follows:
 - a. Mount the Software Diagnostic tape and press the TAPE LOAD switch.
 - b. When the "ENTER COMMAND" prompt is displayed, remove the Software Diagnostic tape and mount a scratch tape with a write ring in place.

- c. In response to the next "ENTER COMMAND" prompt, key

OUTPUT=n

Where:

n
is the tape unit to be used. (Default = T1.)

- d. In response to the next "ENTER COMMAND" prompt, key

SAVE=Dn,Sn,Ns

Where:

Dn
is the disk drive logical unit number (7,8,9,A)

Sn
is the starting segment number index (0 through 1C)

Ns
is the number of segments in hexadecimal (up to 4 for D3400 disk drives or 1D for ISS disk drives).

The contents of the disk are then written to tape; e.g.,

SAVE=7,0,4

SAVES four segments (0 through 3) from disk drive 1 (logical unit number 7).

- e. Press the DISK LOAD switch and perform either a WARM or COLD START.

NOTE:

This disk dump procedure may also be performed with a scratch tape rather than the memory dump tape. The only difference is that the "CONTINUATION TAPE?" prompt is answered by typing N.

- f. Press the DISK LOAD switch and perform either a WARM or COLD START.

G. Subbatch Balancing

Subbatch balancing enables the performance of multiple balance operations within a single data batch. Several subbatches, containing logically corresponding documents, may be balanced without the need to create separate data batches. This reduces the overhead involved in opening and closing files and the number of disk batches.

Subbatch balancing provides these capabilities:

- In the Verify and Search/Modify modes, each subbatch can be balanced as if it were a separate batch.
- In the Verify and Search/Modify modes, the operator is notified when a correction to a field changes the out-of-balance status of a balance register to an in-balance status.
- In the Verify mode, if a subbatch has an in-balance status, key verification of associated fields will not be required. A field will be skipped whether the subbatch was originally in-balance, or whether corrections were made to achieve an in-balance status.

G.1. OPERATING PROCEDURES

G.1.1. Customer Use

In a typical customer credit card processing application, merchants would deposit all the credit card charge slips they have received during the day with their banks. Each merchant would submit a total deposit sheet and one or more credit slips. Because there might be only a few charges submitted by each merchant, it would be more efficient for a bank to use subbatch balancing and handle several merchants' charges within a single batch. Otherwise, the bank would have to create separate batches for each merchant.

Using subbatch balancing, the bank would bundle all the credit card documents received from all merchants that day in one batch. A grand total sheet of adding machine tape would be placed on top of the entire bundle to indicate the sum of the merchants' totals. The bundle of documents would then be sent to the data entry department.

The following examples illustrate subbatch balancing in the Enter and Verify modes.

Subbatch Balancing in the Enter Mode

The sequence in which the documents are arranged for data entry might be as follows:

Grand total record

Subbatch	Record 1
total	Record 2
record	:
1	Record n

Subbatch	Record 1
total	Record 2
record	:
2	Record n

How the format program is written determines how the system balances the figures: for example, a format program may subtract credit slip records from the merchant total record, or may subtract merchant total records from the grand total record.

Data entry continues until the entire bundle has been completed. Status information at the time the batch is closed provides the current balance status.

Enter Mode Limitations

Insertions and deletions are not allowed in the Enter mode. Also, concurrent data entry and verification is not possible during subbatch balancing; this is equally true for simple batch balancing.

Once the data entry operator has finished initial input, the bundle of documents would go to the Verify operator. But automatically converting to the Verify mode is not possible; the Enter operator may convert to the Verify mode by following these steps:

1. Close in Enter mode.
2. Open in Search/Modify mode.
3. Make the needed corrections, then close the batch.
4. Open in Enter; "CONTINUE? = YES".
5. Enter the next subbatch.

Verify Mode

The Verify operator follows these steps using the same bundle of documents as those used in the Enter Mode:

1. Open the batch in the Verify mode.
2. As the system reaches the first merchant total record, the balance totals for that subbatch will be restored.
3. In-balance status:
 - a. A field defined as "balance verify" will be skipped over if the balance register associated with that field is in balance.
 - b. If the only verification specified is "balance verify", and the subbatch is in balance, then the entire subbatch is skipped.
4. Out-of-balance status:
 - a. If the balance register associated with a given field is out of balance, the field will require key verification.

G.1.2. Format Programming

At the start of a subbatch, the format program must provide space in the first field of a record to hold the balance total. This is accomplished by using a RESERVE statement. At the end of the subbatch, a SAVE statement enables the balance total to be placed into the field reserved at the beginning of the subbatch.

Sub-batch balancing works only with data records that are equal to or greater than 74 characters. A runtime check is made for this condition; if an error is found, the following message is displayed:

FMT ERR - SUB BATCH RCD LTH

Balance Registers and Verification Type

To write a correct program for subbatch balancing, it is important to understand the balance register concept and the type of verification to specify in the Verify checkbox (i.e., sight-, key-, or required-verify).

1. Subbatch balancing will support up to 60 registers to be defined with RESERVE statements; the highest possible sub-batch balance register number is B60. This upper limit results from the maximum size of the RESERVE field (999 characters).

2. Subbatch balancing allows the definition of a number of subbatch balance registers and several batch balance registers. Fields associated with an out-of-balance register require verification. All in-balance register fields are skipped during the verify process.
3. Even if a subbatch total shows an in-balance status, the subbatch is not automatically skipped during the verify operation. If a field is coded for sight-verify, key-verify, or required-verify in checkbox 10, the cursor will stop at the field regardless of the balance status. Fields that are not balance-related will be skipped only if they are coded N or blank in the VERIFY checkbox.

Assigning Subbatch Balance Registers

Disk space and Terminal Work Area (TWA) record buffer space can be kept to a minimum if subbatch balance registers are assigned consecutively beginning with B01, the lowest-numbered balance register, and then assign batch-wide balance registers. The reason for beginning with the lowest number is that the physical size of the RESERVE field in a subbatch header record is determined by the highest-numbered register used for subbatch balancing. Thus, assigning a high-numbered register causes a large subbatch header record to be required; if this record is longer than the longest data record, all disk records will be increased to the length of the header record.

Beginning assignments with low-numbered balance registers also help to save the TWA space required for a table that is constructed when subbatch balancing is used. The RESERVE RECORD TBL in the TWA contains one entry per balance register used by the format, and table size is calculated as three times the highest-numbered balance register referenced by a RESERVE statement. The required TWA space can be minimized by assigning balance registers consecutively beginning with B01.

Use of the New COBOL Verbs

1. RESERVE Statement

The RESERVE statement is used only in subbatch balance applications. Examples of the RESERVE statement follow:

Statement	Meaning
RESERVE B01.	Use of the statement causes space to be reserved in the current field for balance register totals. A balance register must be specified. The number specified must be the same register number that the associated data records will reference in the checkboxes. If, by mistake, the register numbers are not identical, a compile error will not be generated.
RESERVE B02.	
.	
.	
RESERVE B60.	

The following rules apply to a field containing the RESERVE statement:

- a. The field that contains the RESERVE statement must be the first field in the level. The length of this field must equal 15 times the highest-numbered subbatch balance register number using this field.
- b. The field must not be keyed or displayed.
- c. The field may not be referenced by any other COBOL procedures.
- d. The output position of the RESERVE field must be zero.

2. SAVE Statement

The SAVE statement is used during subbatch balancing to store the contents of the balance register in the field identified by the RESERVE statement. One must specify a balance register.

Examples of the SAVE statement follow:

Statement	Meaning
SAVE B01.	Using this procedure causes the batch to be suspended while the previously entered record containing the last RESERVE field for this balance register is loaded and updated. The reserved field is defined as the first field of a record in which the last RESERVE statement for the same register was executed; the field is indexed into by the balance register number.
SAVE B02.	
.	
.	
SAVE B60.	

When a SAVE statement is executed, a checkpoint is written.

NOTE:

In writing the program, be sure to SAVE the contents of a balance register before the RESERVE statement of the next subbatch is executed. If this precaution is not taken, a runtime error occurs and the following message will appear:

FMT ERR - SAVE SIMT BEFORE RESERVE

Coding For Grand Total Balancing

To write a program that includes grand total balancing in addition to subbatch balancing, coding can be done in several ways, depending on the desired results. The field names mentioned here are examples; the same names appear in the following sample program.

1. Initialize B01 to the figure entered in the grand total field (BANKTOT). When the merchant's total field (MERCHTOT) is coded, reference it as 01 in the balance checkbox; in the procedure statement, INITIALIZE the B02 total to the merchant's total field. Code the credit slips (CARDTOT) field as 02 in the balance checkbox; this allows the credit slips to be subtracted from MERCHTOT, and MERCHTOT to be subtracted from the grand total.
2. Initialize one balance register to the grand total. Use a second balance register for subbatch balancing. As balance fields are automatically subtracted from the subbatch register, the field should also be added to an arithmetic register (A01-A99).

At the end of the batch, after all subbatches have been entered, the contents of the arithmetic register should be moved into a field that is associated with the grand total balance register, ideally causing this register to become zero. The operator could be alerted, via format coding, to the contents of this arithmetic field.

3. Create a nonkeyed field associated with the grand total register following the subbatch balance field. After the subbatch field is entered, its contents can be saved in an arithmetic register, and then moved into the grand total register field. This would cause immediate subtraction from the grand total balance register.

This method has the advantage that a change to a field made in the Search/Modify or Verify mode can immediately bring the grand total register into balance; it has a disadvantage in that the disk record size is increased.

Sample Program

An example of coding for subbatch balancing appears in Figures G-1, G-2, G-3, and G-4. Explanatory comments about the program appear in the procedure field, preceded by an asterisk. The example continues with the bank/credit card application described earlier in this document.

The sample format has four levels: A, B, C, and Z.

Level A

Level A contains bank identification and the grand total for the batch. In this level, the operator is prompted for the bank identification number, then for the grand (bank) total. The grand total field is called BANKTOT. The procedure statement, INITIALIZE B01 TOTAL TO THIS FIELD, places the quantity entered in the BANKTOT field into the balance register B01. There is one Level A record per batch.

Level B

Level B contains merchant identification and a merchant total, which is actually the subbatch total. The operator is prompted to enter the merchant number and then the merchant's total credit card charges (MERCHTOT). The procedure statement, INITIALIZE B02 TOTAL TO THIS FIELD, places the quantity entered in the MERCHTOT field into the balance register B02. There is one Level B record per subbatch and it follows Level A.

In the first field of level B, the programmer created an unnamed, unkeyed field that is reserved for the contents of register B02. This field is used to store the value remaining in B02 after each subbatch balancing operation is completed. In accordance with the rules for coding a sub-batch balance program, the size of the sample field is 30; that is, 15 times the highest-numbered subbatch balance register (B02) associated with the operation.

Level C

Level C controls the entry of individual credit card charges. There are as many Level C records as there are credit card charges. The operator is prompted to enter a credit card number, then the total for each card. Each CARDTOT value is automatically subtracted from the merchant total stored in B02. In the sample, the CARDTOT value is moved to arithmetic register A01; the value in A01 is then moved to a nonkeyed, nondisplayed field that is used for grand total balancing.

The operator is then prompted to describe the status of the entry operation; the response determines the level selection from C:

S = SUB-BATCH COMPLETE (The latest entry was the last one for one merchant's credit card transactions.)

B = BATCH COMPLETE (The latest entry was the last one for the entire batch.) This response causes Level Z to be selected automatically.

FLD REL = CONTINUE (There are more credit card charges to be entered for the current merchant [subbatch].)

After all items are entered for a given merchant or for all the merchants in the entire batch, the sub-batch balance total is saved by means of the procedure statement SAVE B02. As noted previously in this section, the SAVE process must occur before the RESERVE statement is executed for the next subbatch or the value stored in B02 will be lost.

Level Z

If the operator selects "B = BATCH COMPLETE" in response to the status prompt, the system selects level Z and the batch closes automatically.

PREPARED BY Z CREDIT / 1

DATE 6/17/81

LABEL FORMAT
 TABLE SUBPROG

PAGE 1 OF 4

NAME	LEVEL SELECT	RECORD SIZE	AVERAGE KEYSTROKE/HR.	TRANS.	REC/FILL	BLK. PAD	REG. PAD	BLOCK FACTOR	LABEL NAME	DATE	COMMENT	
2	3	4	5	6	7	8	9	10	11	12	13	14
CREDIT	A									06/17/81	BANK TOTAL RECORD	

FIELD NAME	FIELD SIZE	SHIFT	DATA TYPE	REVISED	UPDATE	POST RELEASE	DISPLAY	VERIFY	BALANCE	TAB CONT.	AUTO	AUX. DUP.	OUTPUT POSITION	JUSTIFICATION	FILL	CONVERSION	PROCEDURE
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	13		P										Ø				PROMPT 'BANK NUMBER: '
BANKNO	5	N															
	19		N										Ø				MOVE 'Ø' TO THIS FIELD.
	12		P										Ø				PROMPT 'BANK TOTAL: '
BANKTOT	9	NN	S				Y										IF MODE = 'ENTER' OR MODE = 'SEARCH' INITIALIZE BANK TOTAL TO THIS FIELD.
																	SELECT LEVEL B.
																	END LEVEL.



Figure G-1.
Coding for Subbatch Balancing, Level A

9896.2 SPERRY 1900/10 and XI/40 LEVEL 11.R1.00
 UP-NUMBER Release Description PAGE G-9

PREPARED BY Z CREDIT / 1 DATE 6/17/81

LABEL FORMAT
 TABLE SUBPROG

PAGE 2 OF 4

NAME	LEVEL SELECT	RECORD SIZE	AVERAGE KEYSTROKE/HR.	TRANS.	REC. FILL	BLK. PAD	REC. PAD	BLOCK FACTOR	LABEL NAME	DATE	COMMENT
2	4	5	6	7	8	9	10	11	12	13	14
	B										MERCHANT TOTAL RECORD

FIELD NAME	FIELD SIZE	SHIFT	DATA TYPE	KEYED	UPDATE	POST RELEASE	DISPLAY	VERIFY	BALANCE	TAB CONT.	AUTO	AUX. DUP.	OUTPUT POSITION	JUSTIFICATION	FILL	CONVERSION	PROCEDURE
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	3		NNN	N			N										RESERVE B02.
	1		7				P										PROMPT 'MERCHANT NUMBER: '.
MERCH.NO.	1		6				N										
	2		1				N										MOVE 'B' TO THIS FIELD.
	1		6				P										PROMPT 'MERCHANT TOTAL: '.
MERCH.TOT	1		NNN				Y										IF MODE = 'ENTER' OR MODE = 'SEARCH' INITIALIZE B02 TOTAL TO THIS FIELD.
																	SELECT LEVEL C.
																	END LEVEL.

UD1-993



Figure G-2.
Coding For Subbatch Balancing, Level B

9896.2
 UP-NUMBER
 SPERRY 1900/10 and XL/40 LEVEL 11.R1.00
 Release Description
 G-10
 PAGE

PREPARED BY Z CREDIT / 1

DATE 6/17/81

LABEL FORMAT
 TABLE SUBPROG

PAGE 3 OF 4

NAME	LEVEL SELECT	RECORD SIZE	AVERAGE KEYSTROKE/HR.	TRANS.	REC.FILL	BLK.PAD	REC.PAD	BLOCK FACTOR	LABEL NAME	DATE	COMMENT						
2		5	6	7	8	9	10	11	12	13	14						
CREDIT SLIP ENTRY RECORD																	
FIELD NAME	FIELD SIZE	SHIFT	DATA TYPE	KEYED	UPDATE	MOST RELEASE	DISPLAY	VERIFY	BALANCE	TAB CONT.	AUTO	AUX. DUF.	OUTPUT POSITION	JUSTIFICATION	FILL	CONVERSION	PROCEDURE
2	3												15	16	17	18	19
	1,1,3			P													PROMPT 'CARD NUMBER: '.
CARDNO	1,3N																
	1,1			N													MOVE 'B' TO THIS FIELD.
	1,2			P													PROMPT 'CARD TOTAL: '.
CARDTOT	1,0NNS							B0,2									MOVE THIS FIELD TO A01.
	1,0NNS			N				N 0,1									* DUPLICATED, CARDTOT, FIELD, TO ALLOW GRAND TOTAL BALANCING. MOVE A01 TO THIS FIELD.
	6			N													MOVE 'B' TO THIS FIELD.
	3,9			P													PROMPT 'ENTER: S=SUB-BATCH COMPLETE'.
	3,9			P													PROMPT ' B= BATCH COMPLETE'.
	3,9			P													PROMPT ' FIELD, REL=CONTINUE'.
CODE	1AAB																ALLOW 'S', 'B', 'P'.
																	IF CODE = 'P' GO TO CONT.
																	SAVE B02.
																	IF B02 NOT = 0 ALARM 'SUB-BATCH OUT OF BALANCE'.
																	IF CODE = 'B' SELECT LEVEL Z ELSE SELECT LEVEL B.
																	CONT.
																	END LEVEL.

UD1-993



Figure G-3.
Coding For Subbatch Balancing, Level C

9896.2
UP-NUMBER

SPERRY 1900/10 and XL/40 LEVEL 11.R1.00
Release Description

G-11
PAGE

PREPARED BY Z CREDIT / 1

DATE 6/17/81

LABEL
 TABLE

FORMAT
 SUBPROG

PAGE 4 OF 4

NAME	LEVEL SELECT	RECORD SIZE	AVERAGE KEYSTROKE/HR.	TRANS.	REC/FILL	BLK. PAD	REG. PAD	BLOCK FACTOR	LABEL NAME	DATE	COMMENT	
2	3	4	5	6	7	8	9	10	11	12	13	14
												CLOSE BATCH WHEN 'B' IS SELECTED

FIELD NAME	FIELD SIZE	SHIFT	DATA TYPE	KEYED	UPDATE	MUST RELEASE	DISPLAY	VERIFY	BALANCE	TAB CONT.	AUTO	AUX. DUP.	OUTPUT POSITION	JUSTIFICATION	FILL	CONVERSION	PROCEDURE
2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
	0																END.

UD1-993



Figure G-4.
 Coding For Subbatch Balancing, Level Z

9896.2
 UP-NUMBER
 SPERRY 1900/10 and XL/40 LEVEL 11.R1.00
 Release Description
 G-12
 PAGE

Sample Data Entry

After entering the data using the sample program, an operator might get the following results (two keying errors are included):

RCD NO.	BANK NO.	BANK TOTL	LVL	OPERATOR RESPONSE TO END	BALANCE AFTER B01	REGS FLD REL B02
1	10156	\$800.00	A		800	0
	MERCHANT NO.	MERCHANT TOTL				
2	519033	200.00	B		800	200
	CARD NO.	CARD TOTL				
3	1029-482-911-782	150.00	C	FLD REL	650	50
4	9567-291-200-107	55.00	C	FLD REL	595	5-
5	4024-776-107-244	25.00	C	S	570	30-
	MERCHANT NO.	MERCHANT TOTL				
6	519350	300.00	B		570	300
	CARD NO.	CARD TOTL				
7	6542-765-121-512	150.00	C	FLD REL	420	150
8	5231-512-316-754	25.00	C	FLD REL	395	125
9	7890-343-721-841	125.00	C	S	270	0
	MERCHANT NO.	MERCHANT TOTL				
10	519477	300.00	B		270	300
	CARD NO.	CARD TOTL				
11	4024-123-456-789	175.00	C	FLD REL	95	125
12	7659-148-246-801	45.00	C	FLD REL	50	80
13	1776-197-855-263	60.00	C	FLD REL	10-	20
14	1775-168-898-059	40.00	C	B	50-	20-

Due to keying errors, two of the subbatches are out of balance:

FIRST MERCHANT

The second card total was entered as 55, but the actual credit card item should have been 25 (record 4). Thus -30 is stored as the balance total for the first subbatch.

THIRD MERCHANT

The first card total was entered as 175, but the actual credit card item should have been 155 (record 11). Thus -20 is stored as the balance total for the third subbatch.

Sample Data Verification

The batch must now be verified.

NOTE:

A detailed status of this job would show a total for B01 of -50 and a total for B02 of -20. The B01 total is the balance difference for the entire batch. The B02 total is the balance difference for the last subbatch.

The program specifies that only the field CARDTOT requires batch verification.

Record number 2 is a balance total record. After reading this record, the system stores the contents of the first field (RESERVE field in B02, value -30).

Record 3 is verified at 150.

Record 4 is corrected with a value of 25. This change brings the first subbatch into balance.

Verify now continues and does not stop until record 11. (Record 10 loads a value of -20 in B02.) (Note that it did not stop in second subbatch because that subbatch was in balance.) Record 11 is corrected with a value of 155 and batch closes in balance.

G.1.3. SYSGEN Procedures

A new prompt has been added to the SYSGEN dialog:

SUB-BATCH BALANCE SYSTEM? Y=YES REL=NO

If the default of NO is selected, subbatch-type balancing is not performed. There is no error indication at compile time or run time. The system ignores all subbatch balancing procedures but performs all other procedures. Memory required to perform subbatch balancing is only allocated and procedures are only executed if YES is selected in response to this prompt.

There are no other special procedures which are not already present with batch balance capabilities, that are required of the operators or the supervisor.

G.2. OPERATIONAL CONSTRAINTS

G.2.1. Parameter Limitations

Concurrent Enter/Verify Mode

Concurrency is not supported. This is consistent with current batch balance processing on the system.

Number of Subbatch Registers

Subbatch balancing allows up to sixty registers to be defined with RESERVE statements. No restrictions are placed on batch balancing formats. They continue to operate as in previous versions of the software. The highest-numbered subbatch balance register possible is B60. This restriction is imposed by the maximum size of the RESERVE field (999 characters).

Subbatch Register Assignments

Since the physical size of the RESERVE field in a subbatch header record is determined by the highest-numbered balance register used for subbatch balancing, the balance registers should be assigned so that the lowest-numbered balance registers are used first for subbatch balancing. Disk space and TWA record buffer space are kept to a minimum if subbatch balance registers are assigned consecutively beginning with B01.

Subbatch TWA Memory Size

When subbatch balancing is used by a format program, a RESERVE RECORD TBL is constructed in the TWA. The size of this table is a function of the highest-numbered balance register (subbatch or subwide) used by the format. Therefore, TWA space is kept to a minimum if balance registers are assigned consecutively beginning with B01.

G.2.2. Performance Limitations

Customers using subbatch balance will experience a decrease in available resident memory of approximately 2K bytes.

This feature tends to slow down data entry. Extra disk accesses are necessary for the execution of the SAVE procedure to retrieve and update the AU containing the subbatch header record. Extra processing time is required to maintain subbatch balancing.

G.2.3. Compatibility

Due to the addition of RESERVE and SAVE as reserved words in the compiles, any format program containing the words RESERVE or SAVE outside of a comment, label program, format name, or quoted string will produce a RESERVED WORD ERROR when compiled.

Format programs containing the RESERVE and SAVE verbs will compile and execute on a Version 11.R1.00 system regardless of subbatch balance SYSGEN load status. However, systems not configured for this feature will return an invalid balance status for those batches keyed using those programs.

H. ICL 1900 VLR With Report Writer

ICL 1900 VLR (or 1900 VLR) is a tape recording format that provides for fixed block variable length logical records without record spanning. The following is required:

1. The logical records may be of varying length. When written, this length will be saved with the data so that when the tape is read, the original lengths will be known. (Note: this is rounded up to be a multiple of four in the case of 1900 VLR).
2. As many logical records as will fit will be placed in physical blocks for output. The blocks will be of fixed equal size.
3. There will be no record spanning. If a block is partially filled and there is insufficient remaining space for the next logical record, the block will be padded out to the fixed size (using a fill character) and output. The logical record will then be placed at the beginning of the next block.

Report writer is invoked by the WRITE command and has the features of both the normal write and sequential I/O. Report writer permits output to the 01 output device with features such as labeling, blocking translation and padding. Most of the COBOL procedures may be used to allow flexibility in reformatting the data.

H.1. COMPARISON OF REPORT WRITER AND NORMAL WRITE

Report writer, along with the normal write, is invoked by WRITE. They both output to the 01 output device and have the same tape options, including labeling and packing.

The normal write is simple. For most applications it is used to transfer the data from batch to 01 output device with no reformatting. Except for indexed files, OPEN, CLOSE, READ, and WRITE operations (COBOL procedures) are not allowed. Therefore, the SEQ IO module is not entered. Without SEQ IO, the reformatting capability of the normal write is limited.

With report writer, all input from batches is through READ and all output to tape through WRITE. Report writer has a peculiar requirement for a dummy batch. Report writer has an environmental section to describe the input and output files, field descriptions that define the input and output records, and a procedure division that controls the processing. The only field description in the procedure division is a 0 length unlabeled field at the beginning.

H.2. COMPARISON OF REPORT WRITER AND SEQUENTIAL I/O

Report writer and sequential I/O are very similar. There are only a few differences:

1. Report writer is invoked by a Supervisor WRITE command; therefore:
 - a. Labeling is an option.
 - b. Input is from the batch list (using file name BATCH-LIST).
 - c. Output is to the 01 output device (using file name UTILITY).
 - d. A dummy batch is necessary as the first record of the input.
 - e. A zero length field is necessary at the beginning of the procedure division.
2. Sequential IO is invoked by the File Management operating mode; therefore:
 - a. Labeling is not an option.
 - b. Input or output is directly to the unit, either TAPE, PRINTER, DISK, or the actual unit specification (such as LP, T1, SP2, etc.).
 - c. Neither a dummy batch nor a zero length field is required.

H.3. PROGRAMMING FOR REPORT WRITER

If report writer is used to create the 1900 VLR tape, a simple format (such as FREEFORM) can be used to ENTER the batch. The batch would normally not contain a field for the record length; the record length field would be added by report writer. A simple format could also be used to INPUT, but provisions must be made to handle the four character record length.

The report writer format, used to generate both ICL 1900 VLR and (in ENTER mode) the dummy batch needed for report writer processing, should have the following characteristics: (1) In the environment division, the record size should be set to the largest record expected, including the four bytes for the record size. (2) In the data definition division, the first field of each record of UTILITY (i.e. output), should be:

FIELD SIZE	4
DATA TYPE	N
OUTPUT POSITION	1

The COBOL procedure division should be able to calculate the record size (using some user-defined criterion) and move it into the first field. In order to use this format to generate the dummy batch, the first statement in the COBOL procedure division should be something like "IF MODE = 'ENTER' GO TO FINISHED" where "FINISHED" is a label immediately preceding the "END" statement.

Refer to Figure H-1 for an example of a 1900 VLR report writer format.

```

JOB NAME: ICL-RPRT    BATCH: 1    DATE: 83115    TIME: 0916
REC  PROGRAM  LVL LVL  REC  AVRG  TRAN REC  BLK  BLK  LABEL
NUM  NAME     ID  SEL  SZ  KSTR  FIL  PAD  FAC  NAME    DATE  COMMENT
  1  ICL-RPRT      E   84
  2
  3
  4
  5
  6
  7
  8
  9
 10
 11
 12
 13  BTCHDATA      D   80
 14  IN-DATA      80  U
 15
 16  OUT-FILE      D   84
 17  REC-LTH      4  N
 18  OUT-DATA      80  U
 19
FILE SUMMARY
 20  BTCHDATA
 21  OUT-FILE
 22
 23
 24
 25
 26
 27
 28
 29
 30
 31
 32
 33
 34
 35
 36
 37
 38
 39
 40
 41
 42
 43
 44
 45
 46
COMPILATION DONE
ERRORS: 0 SIZE: 224 WORDS

                                PROCEDURE LEVEL

REC FLD  FIELD  SIZ  DATA  UPD  DSP  BAL  AUT  POS  FIL
NUM  NAME   SHF  KEY  REL  VER  TAB  AUX  JUS  CNV    PROCEDURE

21  1  FLD      0
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
* THIS PROGRAM WILL CREATE ICL 1900 VLR TAPES.
* EACH INPUT RECORD CONSISTS OF A CHARACTER STRING OF UP TO
* 80 BYTES.
* THE REMAINDER OF THE 80 BYTES SHOULD BE FILLED WITH BLANKS
* AND ARE NOT INCLUDED IN THE COMPUTED RECORD LENGTH.
*
IF MODE = 'ENTER'
  GO TO FINISHED.
OPEN INPUT BTCHDATA.
IF BTCHSTAT > 0 GO TO FINISHED.
OPEN OUTPUT OUT-FILE.
IF OUT-STAT > 0 GO TO FINISHED.
DECLARE REGISTER SIZE 4.
IN-REC.
READ BTCHDATA AT END GO TO FINISHED.
MOVE IN-DATA TO OUT-DATA.
MOVE 80 TO A1.
LDOP.
IF OUT-DATA(A1) = ' ' AND A1 IS NOT = 1
  COMPUTE A1 = A1 - 1. GO TO LOOP.
COMPUTE REC-LTH = A1 + 4.
WRITE OUT-REC.
GO TO IN-REC.
FINISHED.
END.

```

Figure H-1.
Example of an ICL 1900 VLR Report Writer Format

I. IBM VLR

Version 11.R1.00 has been enhanced to provide IBM Variable Length Record (VLR) packing capability for the WRITE and INPUT supervisor commands and report writer.

I.1. DESCRIPTION

IBM VLR is a tape format using variable length records in variable length blocks.

The following is a description of the general layout of physical blocks, interblock gaps, and tape marks:

HEADER LABEL

tape mark

QUALIFIER BLOCK

DATA BLOCK

DATA BLOCK

·
·
·

DATA BLOCK

tape mark

TRAILER LABEL

tape mark

tape mark

PERTEC LABEL

tape mark

tape mark

I.1.1. Data Block

The data block contains a variable number of complete logical records (i.e., no record spanning) with block truncation for any unused bytes at the end of the block.

Each variable record is immediately preceded by a four-byte (i.e., two-word) record header consisting of a binary record length count which includes the four-byte field itself.

Each block is preceded by a four-byte (two-word) block header consisting of a binary block length field which includes the four-byte field itself.

The following illustrates a typical block lay-out.

```
BLOCK HEADER  
RECORD HEADER  
RECORD 1  
RECORD HEADER  
RECORD 2  
.  
.  
.  
RECORD HEADER  
RECORD N
```

I.1.2. Block Header

The first four bytes of each block contain the following control information:

Bytes 1 & 2 Contain the binary value designating the length of the block in bytes (including this field). This field is automatically generated by the system and may be different for each block.

Bytes 3 & 4 Contain binary zeroes (reserved for system use).

I.1.3. Record Header

The first four bytes of each logical record contain the following control information:

Bytes 1 & 2 Contain the binary value designating the length of the record in bytes (includes this field). This field is supplied by the format program and may be different for each record.

Bytes 3 & 4 Contain binary zeros (reserved for system use).

I.2. COMPATIBILITY

In order to use IBM VLR, all records input or output using IBM VLR must contain four bytes indicating the length of the record. A format program that reads or writes fixed length records will have to be modified in order to read or write IBM VLR. The modified format program will no longer be capable of reading or writing the original fixed length records. Two different format programs will be necessary if it is desired to have the option of using either fixed length or variable length for the same type of data.

Three new label programs--IBMCRV, IBMDLV, and IBMSLV--have been provided for use with IBM VLR tapes. For VLR applications, these should be used instead of the standard IBM label programs IBMCR, IBMDL, and IBMSL, respectively.

I.3. OPERATOR INTERFACE

The IBM VLR format applies only to magnetic tape and can only be created by the WRITE Supervisor command or report writer, or read by the INPUT Supervisor command.

The format and label programs used to process IBM VLR must be designed or modified for this purpose.

I.3.1. IBM VLR Tape Output

To write an IBM VLR tape, the operator must enter Supervisor mode and then specify the WRITE command. The operator will then be prompted for OPTIONS (batch selection options), a list of batches (Job names/batch numbers), a format program, and the SKIP PROC option.

The operator will then be prompted for tape output OPTIONS as follows:

```
OUTPUT =                TAPE OUTPUT
                        DEFAULT = NO MORE

                        LABELS          PAD
                        TAPE POS        TRANS
                        REC LENGTH
                        PK-PACKING
                        BLK FACTOR
```

Figure I-1.
IBM VLR Tape Output—Option Selection Form

PK-PACKING should be selected. The operator will then be prompted for packing type:

```
PACKING =              PACKING TYPE
                        DEFAULT = H2000

                        NONE
                        H2000          VLR-VLB
                        H6000
                        4X3 ICL
                        2970 ICL
                        1900 VLR
```

Figure I-2.
IBM VLR Tape Output—Packing Type

The operator should select VLR-VLB; then a prompt for block size will appear.

```
PACKING = 2970 ICL      DATA BLOCK SIZE
BLOCK SIZE=            DEFAULT = 1024
```

Figure I-3.
IBM VLR Tape Output—Block Size

The operator should now select the proper block size. The block size must be between 14 and 8188 (inclusive). If these conditions are not met, "INVALID BLOCK SIZE" will be displayed on line 3.

At this point, any previously entered specifications for REC LENGTH, BLK FACTOR and PAD are ignored and any subsequent attempt to enter any of these parameters causes "INVALID COMMAND SELECTED" to be displayed on line 3.

Tape Labels

Tape labels for IBM VLR tapes are not required. However, the operator may specify that tape labeling is desired. This is done by selecting the LABELS option and specifying the appropriate IBM label program in response to the "LABEL NAME" prompt (see I.2).

Translation

IBM VLR tapes are normally written in EBCDIC. If a translation other than standard EBCDIC is desired, the operator should select the TRANS option and enter the number of the desired translation table.

I.3.2. IBM VLR Tape Input

To read an IBM VLR tape, the operator must enter the Supervisor mode and then specify the INPUT command. The operator will then be prompted for batches (JOB name, first batch, and records per batch) and a format program.

The operator will then be prompted for OPTIONS. The subsequent prompts and responses are the same as for WRITE as described in I.3.1.

I.4. PROGRAMMING REQUIREMENTS

I.4.1. Writing IBM VLR Tapes

For the system to handle the IBM variable length records, it must know the length of each record to be written. Due to the nature of the format programs, the length must be specified as part of the record. The following requirements must be met when writing IBM VLR tapes:

1. The first four character positions of each output record must be reserved for a field specifying the record length.
2. The record length is the number of bytes (characters) in the record including the four bytes for the record length field.

3. The record length field is a numeric field. (The number is right justified with leading zeroes.)
4. The record length must be within the limits imposed by the block size.

1.4.2. Reading IBM VLR Tapes

When reading IBM VLR tapes, the records are transferred to the format program as follows:

1. Each record is of the length specified by the format program.
2. The first four bytes of each record will contain the record length in the same form as used in writing IBM VLR tapes (see 1.4.1).
3. The record beyond the record length specified by the first four bytes will be blank filled.

1.4.3. General Format Requirements

Normal Write and Input

A single format may be written to be used for Enter, Write, and Input. In all cases except Enter, the SKIP PROC option may be used.

The first output field of each record should be:

FIELD SIZE	4
DATA TYPE	N
OUTPUT POSITION	1

The operator may either ENTER the appropriate record length into the field or the field may be a nonkeyed field and COBOL procedures may be used to calculate (according to some user-defined criterion) and insert a record length when the record is entered.

Report Writer

If report writer is used to create the IBM VLR tape, a simple format (such as FREEFORM) can be used to Enter the batch. Normally the batch would not contain a field for the record length. The record length field would be added by report writer. A simple format could also be used to Input, but provisions must be made to handle the four-character record length. The report writer format, used to generate both IBM VLR and (in the Enter mode) the dummy batch needed

for report writer processing, should have the following characteristics: (1) In the Environment division, the record size should be set to the largest record expected, including the four bytes for the record size; (2) In the Data Definition division, the first field of each record of UTILITY (i.e. output) should be:

FIELD SIZE	4
DATA TYPE	N
OUTPUT POSITION	1

The COBOL Procedure division should be able to calculate the record size (using some user-defined criterion) and move it into the first field. Refer to Figure I-4 for an example of an IBM VLR report writer format.

REC NUM	PROGRAM NAME	LVL ID	LVL SEL	REC SIZ	AVRG KSTR	TRAN	REC FIL	BLK PAD	BLK FAC	LABEL NAME	DATE	COMMENT
JOB NAME: VLR-RPRT BATCH: 4 DATE: 83115 TIME: 0915												
1	VLR-RPRT		E	84								04-25-83 FORMAT TO WRITE IBM VLR USING REPORT WRITER.
2												SELECT BTCHDATA
3												ORGANIZATION IS SEQUENTIAL
4												ASSIGN TO BATCH-LIST
5												ACCESS IS SEQUENTIAL
6												FILE STATUS IS BTCHSTAT
7												SELECT OUT-FILE
8												ORGANIZATION IS SEQUENTIAL
9												ASSIGN TO UTILITY
10												ACCESS IS SEQUENTIAL
11												FILE STATUS IS OUT-STAT
12												END LEVEL.
13	BTCHDATA		D	80						BTCH-REC		DEFINITION OF INPUT
14	IN-DATA	80	U							1		* INPUT DATA
15												END LEVEL.
16	OUT-FILE		D	84						OUT-REC		DEFINITION OF OUTPUT RECORD
17	REC-LTH	4	N							1		* RECORD LENGTH
18	OUT-DATA	80	U							5		* OUTPUT DATA
19												END LEVEL.
F I L E S U M M A R Y												
												BTCHDATA
												OUT-FILE
20			Z									PROCEDURE LEVEL
PROCEDURE												
REC NUM	FLD NAME	FIELD NAME	SIZ SHF	DATA KEY	UPD REL	DSP VER	BAL TAB	AUT AUX	POS JUS	FIL CNU		PROCEDURE
21	1	FLD	0									* THIS PROGRAM WILL CREATE IBM VLR TAPES.
22												* EACH INPUT RECORD CONSISTS OF A CHARACTER STRING OF UP TO
23												
24												* 80 BYTES.
25												* THE REMAINDER OF THE 80 BYTES SHOULD BE FILLED WITH BLANKS
26												* AND ARE NOT INCLUDED IN THE COMPUTED RECORD LENGTH.
27												*
28												IF MODE = 'ENTER'
29												GO TO FINISHED.
30												OPEN INPUT BTCHDATA.
31												IF BTCHSTAT > 0 GO TO FINISHED.
32												OPEN OUTPUT OUT-FILE.
33												IF OUT-STAT > 0 GO TO FINISHED.
34												DECLARE REGISTER SIZE 4.
35												IN-REC.
36												READ BTCHDATA AT END GO TO FINISHED.
37												MOVE IN-DATA TO OUT-DATA.
38												MOVE 80 TO A1.
39												LOOP.
40												IF OUT-DATA(A1) = ' ' AND A1 IS NOT = 1
41												COMPUTE A1 = A1 - 1. GO TO LOOP.
42												COMPUTE REC-LTH = A1 + 4.
43												WRITE OUT-REC.
44												GO TO IN-REC.
45												FINISHED.
46												END.
COMPILATION DONE												
ERRORS: 0 SIZE: 224 WORDS												

Figure I-4.
Example of an IBM-VLR Report Writer Format

I.5. PROCESSING ERRORS

The following processing errors are unique to IBM VLR:

1. If the record length specified in the first four bytes of an input or output record exceeds the block size, "TAPE BLOCK SIZE ERROR" will be displayed and the operation terminated.
2. On input or output, if the record length specified in the first four bytes of a record exceeds the record size specified by the format, "TAPE BLOCK SIZE ERROR" will be displayed and the operation terminated.

I.6. EXCLUDED CAPABILITIES

The following restrictions apply to IBM VLR:

1. The minimum block size is 14 and the maximum is 8188 bytes due to a tape controller restriction.
2. IBM VLR capability has not been implemented in the PRINT or TRANSFER Supervisor functions.

J. U100 DATACOM

The U100 protocol may be selected during SYSGEN or by using the SET command. A remote ID (RID) identifying the communication routing and a site ID (SID) identifying the workstation also are assigned.

J.1. LOGICAL DEVICE DEFINITION

A device ID (DID) identifying each of the four logical output devices is assigned as follows:

163 selects D1
164 selects D2
165 selects D3
166 selects D4

Default record sizes for the logical devices are:

Device	Length
D1	213 characters
D2	230 characters
D3	0 characters
D4	1 character

These may be changed using the LENGTH option of the SET command during SYSGEN.

Since the software equates an output device to the device ID, selection of a DID by the mainframe will direct data to the appropriate device as it is received. Use by the mainframe of one of the following listed DIDs enables data to be received on the screen of the controlling workstation at the same time data is being received to another output device, or enables the mainframe to close an output file without affecting data transmission. In addition, the following device IDs are recognized:

Device	Function
160	Selects the controlling keystation or message terminal designated using the SET command.
161	Selects the controlling keystation without deselecting any other device.
162	Deselects the presently selected device.
167	Selects directory for transmission of batches.
170-174	Device unable to proceed, closes any open output file without deselecting an input file.
176	Selection and data are ignored; nothing is deselected.

J.2. CONTROLLING DATA COMMUNICATIONS

The controlling keystation initiates transmission by building a directory and using the DCOM CNTL command, just as in other protocols. The SWITCH and SUSPEND controls are not valid for U100.

For data reception, only START and TERMINATE may be selected. Screen messages are received, acknowledged, and displayed during data transmission as well as reception.

J.3. UNRECOVERABLE LINE ERRORS

On occurrence of an unrecoverable line error, the system cancels data communications. Data communications must then be restarted. Records are checkpointed when an unrecoverable line error occurs just as in other protocols, and may be restarted from the last successfully transmitted record using the CHKPT CNTL command.

J.4. KEYSTATION MESSAGES

Messages and data may be input from the keystation even if transmission is active, but the data is not transmitted until the mainframe requests the keystation data. The procedure for sending messages and data is the same as for other protocols.

An urgent processor message for the mainframe is accepted and displayed on the screen of the controlling keystation, regardless of whether the communications buffer is available. An urgent processor

message is recognized by its two STX characters. The message "PROCESSOR MESSAGE WAITING" is displayed on line 3 of a 3441 keystation or on line 2 of the display screen of the controlling keystation when a message from the mainframe is waiting. The Supervisor responds by sending the following message:

@@BELL

The mainframe then transmits the waiting message to the display screen of the controlling keystation.

J.5. AUTO ANSWER

The auto answer feature may be used with U100 protocol. Operating procedures are the same as for other data communications protocols.

J.6. EXAMPLES OF 1100 SYSGENS FOR 1900/10 OR XL/40 U100

Figures J-1 and J-2 are examples of SYSGENS on an 1100 for both dial-up and dedicated lines. In these examples, both have allowed SLC to successfully communicate with an 1100 mainframe.

SYMBIONT CONFIGURATION - EXEC 37R2B1:02-61B					
.	N E T W O R K		N E T T B		- TEST BED
.					
.					
NET	NETTB,	POLL	ODD,	SYNC	PERM
MUX	MUXTB,	NETTB	040		
.					
TERM	TTB050	MUXTB	0100	060,0132,P	
DEVICE	CSS132,0,	TTB050	CRT	PRI 12,80	0160,FULL,I
DEVICE	ATB050,1,	TTB050	MTAPE	AUX 12,80	0167,0163
DEVICE	BTB050,2,	TTB050	MTAPE	AUX 12,80	0175,0164
DEVICE	CTB050,3,	TTB050	MTAPE	AUX 12,80	0176,0165
DEVICE	DTB050,4,	TTB050	MTAPE	AUX 12,80	0174,0166

Figure J-1.
Dedicated Line SYSGEN

SYMBIONT CONFIGURATION - EXEC 37R2A:27-82B						
NET	NETAI,	POLL	ODD,SYNC		DIAL	
MUX	MUXAI,	NETAI	040			
.						
TERM	TAI050	MUXAI	U100			060,0132,P
DEVICE	CSS132,0,	TAI050	CRT	PRI	12,80	0160,FULL,I
DEVICE	AAI050,1,	TAI050	MTAPE	AUX	12,80	0167,0163
DEVICE	BAI050,2,	TAI050	MTAPE	AUX	12,80	0175,0164
DEVICE	CAI050,3,	TAI050	MTAPE	AUX	12,80	0176,0165
DEVICE	DAI050,4,	TAI050	MTAPE	AUX	12,80	0174,0166

Figure J-2.
Dial-Up Line SYSGEN

K. Field Release Master Break-Out Procedure

The following procedures detail the copying of tapes from the Field Release Master (FRM) tape.

There are seven EOFs on the FRM. The entire tape may be copied by means of Software Diagnostics using a "COPY=7" command.

K.1. COPY FIELD RELEASE MASTER WITH ONLY ONE TAPE UNIT

To copy from the FRM using a system where only one tape drive (MT1) is available, it is necessary to copy to a simulated tape on the disk and then copy back to MT1.

CAUTION

This procedure overwrites data on the disk.
Saves of jobs and compiles must have been done previously.

K.1.1. Copy Software Diagnostics

1. Mount FRM on MT1 if it has been removed.
2. Type:
SIM=2
INPUT=1
OUTPUT=2
REW=1,2
COPY=2
3. Remove the FRM and mount scratch tape on MT1 when the copy is done.
4. Type:
INPUT=2
OUTPUT=1
REW=2
COPY=2
5. When done, remove the new Software Diagnostic tape from MT1.

K.1.2. Copy the SMB

1. Mount FRM on MT1.
2. Type:

SIM=2
INPUT=1
OUTPUT=2
REW=1,2
SKF=1,1
COPY=5
3. Remove FRM and mount scratch tape on MT1.
4. Type:

INPUT=2
OUTPUT=1
REW=2
COPY=5
5. When done, remove the new SMB from MT1.

K.2. COPY FIELD RELEASE MASTER WITH TWO TAPE UNITS

To copy tapes from the Field Release Master using a system where MT2 is the same density as MT1:

1. Mount FRM on MT1 and press the TAPE LOAD switch. Software Diagnostics will be loaded. The FRM is left on MT1.

The FRM may be used as the new Software Diagnostics tape, but it is recommended that a copy of Software Diagnostics be made onto a working tape, and the FRM kept as a backup.

K.2.1. Copy Software Diagnostics

1. Mount the scratch tape on MT2.
2. Type the following:

INPUT=1
OUTPUT=2
REW=1
COPY=2

K.2.2. Copy the SMB

1. Mount a scratch tape on MT2.
2. Type:

 INPUT=1
 OUTPUT=2
 RFW=1
 SKF=1,1
 COPY=5
3. Remove the SMB from MT2 when done.

L. Hardware Features

L.1. 1900/10 AND XL/40 SYSTEMS

Version 11.R1.00 software supports the following hardware features for both the 1900/10 and XL/40 systems:

- 10 and 20 MB disk drives (8.8 MB on 1900/10)
- 30, 60, 90 and 120 MB fixed disk drives
- 480 or 2000-character KP, TY and AM keystations (EBCDIC)
- 10.5 and 7 in. 800 and 1600 bpi tape drives (7- and 9-trk)
- Station printer
- 180, 300 and 640 lpm printer
- 300 cpm card reader
- Communications adapter

L.2. 1900/10 SYSTEM ONLY

The following hardware features are supported for 1900/10 systems only:

- Cartridge disk controller
- 480 or 2000-character KP, TY and AM keystations (ASCII)
- RDA (local and remote)
- Autoanswer for RDA (local)
- 640 lpm printer attached via coax

L.3. XL/40 SYSTEM ONLY

The following hardware features are supported for XL/40 systems only:

- Processor upgrade (XPU1 to XPU2)
- Keystation line switch
- Diskette subsystem
- Local control
- CARS local
- Remote control
- Modem ADP CARS
- Modem ADP LC RC
- Printer switch (one printer, two systems)
- Printer switch (two printers, one system)
- 160 cps printer
- 170, 660 and 900 lpm printers
- 600 cpm card reader

M. Software Patching Procedures

This section includes the patching procedures as well as a listing of any errors that might occur during the patching.

The process of patching facilitates the correction of software modules on the System Master Binary (SMB) Tape and the Operating System Tape. This process is performed mainly by the SYSGEN program. Two basic SYSGEN commands can be used for this:

CMT Correct Master Tape

CDM Correct Disk Module

The procedures for performing both types of patching are outlined below.

M.1. CMT PATCHING AND CORRECTIONS

The SYSGEN command CMT creates a new SMB tape from an old SMB tape and incorporates patches entered from the keystation or from a patch tape. The procedure for CMT patching is as follows:

CAUTION

This procedure will destroy the data on the disk. Be sure to save any customer data that is on the disk.

CMT Patch Procedure

Display	Operator Action
1.	Mount the old SMB on the primary tape drive. If a patch tape is to be applied, mount it on any other drive. Press the LOAD and IPL Release keys (blank white buttons) on the system control panel. The system will load tape and respond with the following prompts.

Display

Operator Action

2. ***MASTER VERSION NO.XX.XX.XX***
LOAD MASTER TAPE TO DISK?
REL=YES, N=NO

Press the FIELD REL key.

*misstake
in min*

3. DISK INITIATION FOR LOAD IN PROGRESS
LOAD SYSGEN OVERLAYS TO DISK

Press the FIELD REL key.

NO	NAME	VER	TYP	ID	STATUS
001	S BLD LC	X.X	014	001	LOAD
002	S GEN LC	X.X	014	001	LOAD
003	S BLD MSG	X.X	014	002	LOAD
004	S LD TBLS	X.X	014	003	LOAD
005	S P BUF 1	X.X	014	001	LOAD
END OF FILE					
TO SELECT MODS, TYPE: Y=YES, REL=NO					

4. NO NAME VER TYP ID STATUS

001	S BLD LC	X.X	014	001	LOADED
.
.
.
005	S P BUF 1	X.X	014	001	LOADED

SYSGEN VERSION NO.XX.XX.XX
TYPE SYSGEN COMMAND

(This prompt takes several minutes to be displayed.)

Type ALU=16 then press the FIELD REL key to output the modules to the line printer.

Type ALU=5, then press the FIELD REL key to output the modules to the station printer.

Don't need

5. ALU=N SET ALTERNATE LIST
UNIT = N
TYPE SYSGEN COMMAND

Type LC=1 then press the FIELD REL key to print all commands.

6. LC=N SET LIST COMMAND
FLAG = N
TYPE SYSGEN COMMAND

Type PTU=N then press the FIELD REL key where N is the tape drive on which the patch tape is mounted.

used patch from tapes

7. PTU=N SET PATCH TAPE
UNIT = N
TYPE SYSGEN COMMAND

Type CMT then press the FIELD REL key.

Now Goto step 8 to begin patch

- | Display | Operator Action |
|---------|-----------------|
|---------|-----------------|
8. **CMT CORRECT MASTER TAPE**
- | NO | NAME | VER | TYPE | ID | STATUS |
|-----|----------|-----|------|-----|--------|
| 001 | S BLD LC | X.X | 014 | 001 | LOAD |
| : | : | : | : | : | : |
| : | : | : | : | : | : |
- END OF FILE**
REVIEW AND CORRECT MASTER CORRECTIONS
TYPE: Y=YES, REL=NO
- (All of the software modules will be listed. This takes several minutes).
- Press the FIELD REL key only if patches are to be added to the SMB tape. Go to step 14: 13
- Type Y then press the FIELD REL key if patches are to be deleted.
9. **REVIEW SUB COMMANDS ARE:**
- | | |
|------------|---------------------------------|
| REL | DISPLAY NEXT LINE |
| D | DELETE CURRENT LINE |
| T | TERMINATE CURRENT MODULE |
| R | RECYCLE CURRENT MODULE |
- TYPE MODULE NAME:**
- Type the module name which is to be reviewed, then press the FIELD REL key.
- NOTE:**
- Modules must be reviewed in the order in which they are listed on the line printer in step 9.
10. **TYPE VERSION: FORM IS XX.XX**
- The version number of the module to be reviewed must be entered, followed by pressing the FIELD REL key.
- Significant digits may be entered with or without spaces as fill characters. Zero (0) fill characters are not allowed.

Display

Operator Action

11. ****XX-XXX**XX.XX.XX**PATCH NO.XXXXX**

Press the FIELD REL key to review each line of the patches for that module. If a line is to be deleted, display the line, then type D and press the FIELD REL key.

If the module is to be reviewed again, type R then press the FIELD REL key.

If the module is no longer to be reviewed, type T then press the FIELD REL key.

12. **END OF CORRECTIONS FOR THIS MODULE
TYPE MODULE NAME**

If patches are to be deleted from another module, type the module name then press the FIELD REL key.

If no more modules are to be reviewed, type XXX then press the FIELD REL key.

13. **ENTER NEW CORRECTIONS?
TYPE Y=YES, REL=NO**

T=TAPE

If new patches are to be keyed, type Y then press the FIELD REL key and go to step 16. *16*

If new patches are to be applied from a patch tape, type T then press the FIELD REL key.

If no new patches are to be added, press the FIELD REL key and go to step 19. *19*

Display	Operator Action
14. MOUNT PATCH TAPE...REL WHEN READY	Mount the patch tape on the unit specified in the PTU command (step 7). When the tape is ready, press the FIELD REL key. The system will read the patch tape and display what is read on the CRT.
15. TYPE MODULE NAME	Type the module name then press the FIELD REL key. or If no more modules are to be corrected, or if a patch tape is being applied, type XXX then press the FIELD REL key. Go to step 19. /3
16. TYPE VERSION: FORM IS XX.XX	Type the version number of the module to be patched then press the FIELD REL key. Significant digits may be entered with or without spaces as fill characters. Zero (0) fill characters are not allowed.

Display

Operator Action

17. TYPE CORRECTION LINES

Comments or patch numbers may be entered by keying an asterisk (*) in the first position of each comment line followed by the comment information. Refer to the Patch Form for the comments.

Correction code must be typed exactly as specified on the Patch Transmittal Form. All spaces, including those preceding actual code, must be typed. Attention must be given to entering any periods (.) and spaces () which occur in the code. A slash (/) and a check digit must be entered at the end of each line of code.

After the last line of code is entered for a particular patch, a G is typed to terminate that particular patch. Go to step 16.

18. MOUNT TAPE: REL=READY, A=ABORT

Remove the old SMB tape from the primary tape drive. Mount a write-enabled tape which is to become the new SMB tape. Press the FIELD REL key.

Display

Operator Action

19. **CHANGE XX.XXX.XX TO YY.YY.YY?**
TYPE VERSION NUMBER OR REL TO ACCEPT

If the new version number is correct, press the FIELD REL key. If the new version number is not correct, type the correct new version number, then press the FIELD REL key. The new SMB tape will be written. (This will take several minutes.)

M.2. CDM PATCHING

The CDM command provides a method of correcting disk resident modules under the control of SYSGEN. In turn, a new Operating System Tape may be created from the modified disk-resident operating system.

CAUTION

This procedure will destroy the data on the disk. Be sure to save any customer data that is on the disk.

CDM Patching Procedure

Display

Operator Action

- 1.

Take all SAVES (Job, Compile, Directory) and FILES (under File Inquiry). Mount the existing System Tape on the primary tape drive.

Press the LOAD and IPL Release keys (white pushbuttons) on the MCU System Control Panel simultaneously.

Display	Operator Action
2. ***SYSTEM VERSION NO.XX.XX.XX*** SYS TAPE START OPTION - COLD	Type SYS, then press the FIELD REL key.
3. SYS DEFERRED UNTIL TAPE LOADED SYS TAPE START OPTION - COLD	Type COLD, then press the FIELD REL key.
4. DISK INITIATION FOR LOAD IN PROGRESS TAPE LOAD INITIATED ***SYSGEN VERSION NO. XX.XX.XX***	If a line printer is available on the system, type ALU=16, then press the FIELD REL key. (Use ALU=5 for a station printer.)
5. TYPE SYSGEN COMMAND	Type CDM then press the FIELD REL key.
6. CDM CORRECT DISK MODULE TYPE MODULE NAME	Type the module name from the Patch Form then press the FIELD REL key.
6a. ENTER PATCH NUMBER	Type the 3-digit patch number from the Patch Form, then press the FIELD REL key.

Display

Operator Action

7. **ENTER CORRECTIONS**
R=XXXX B=XXXX U=XXXX L=XXXX
D=XXXX

At this time SYSGEN passes control to PERDEBUB for the entering of the patches.

Comments and cross reference problems may be entered by keying an * in the first position followed by the comment information. Refer to the Patch Form for the comments. Correction code must be entered exactly as specified on the Patch Form. All spaces, including those preceding actual code, must be keyed. Attention must be given to entering periods (.) which occur in the code. Do not enter the slash (/) and check digit at the end of each line of code. Press the FIELD REL key.

The system will display a check digit and should be sight verified with the one shown on the Patch Form. After each line of code is entered, system will display a series of hexadecimal values.

After patches for the above module name have been entered, type G then press the FIELD REL key.

Display	Operator Action
8. G/X R=XXXX B=XXXX U=XXXX L=XXXX D=XXXX TYPE MODULE NAME	If another module is to be patched, type the module name. Go to step 7. If no other modules are to be patched, press the FIELD REL key.
9. TYPE SYSGEN COMMAND	Mount the scratch tape (with a write ring) which is to become the new Operating System Tape on the primary tape drive. Type WTS then press the FIELD REL key. (The system will then write the new corrected System Tape.)
10. CHANGE XX.XX.XX to YY.YY.YY? TYPE VERSION NUMBER OR REL TO ACCEPT	Type in new version number and press the FIELD REL key or press the FIELD REL key to accept given version number (YY.YY.YY).

M.3. PATCHING ERRORS

Since the software patching procedures utilize the SYSGEN program, the errors generated during patching are those that are generated during the Initial System Configuration procedure of SYSGEN. These are SYSGEN and IOCTL errors. Refer to F.9 for a listing of these errors.

Index

Term	Reference	Page
A		
Accept all definitions?	E	F-49
Accept configuration?	E	E-9
Accept file mgmt parms?	E	E-51
Additional expansion?	E	E-16
Alarm messages	F.2	F-1
ALLOW/DISALLOW	F.3.3	F-3
Alphabetic polling address	E	E-15
Alternate dump procedure	F.10.2	F-82
ALU	E	E-2
Any data key to continue	1.27	1-14
Arithmetic register	G.1.2	G-6
ASCII	L.2	L-1
ASCII code	E	E-16
ASCII keyboards?	E	E-5
ASCII translation?	E	E-16
Assigning sub-batch balance registers	G.1.2	G-4
AUPROMPT	C.3	C-7
	C.3.1	C-8
AUTO ANSWER	J.5	J-3
AUTO checkbox functionality correction	1.10	1-6
	2.1.2	2-1
AUTO save	E	E-36
Automount	E	E-9
Autoprompt	2.1.3	2-1
Autoprompt batch compatibility	2.1.3	2-1
AVERAGE KEYSTROKE RATE field	1.33	1-16
Average TWA/Compile ranges	6.2.1	6-3
B		
B01	G.1.2	G-4
Background message report	1.24	1-12
Balance registers and verification type	G.1.2	G-3
Batch compatibility	2.1.1	2-1
Batch header	1.13	1-7
BATCH STATISTICS expansion	1.27	1-14
BK LOG message	1.21	1-11

Term	Reference	Page
BLK FACTOR	I.3.1	I-4
Block header	I.1.2	I-2
Break up long communication records	1.37	1-18
BSC protocols	1.37	1-18
BSF	D.5.2	D-6
BSF command syntax	D.5.2	D-6
BSR	D.5.3	D-6
BSR command syntax	D.5.3	D-6
BUFFER CHANGE	A.1.2	A-1
	A.2.3	A-4
Buffer expansion?	E	E-16
BUILD DIR	1.50	1-22
	A.2.1	A-4

C

CALENDAR	C.4	C-45
	C.4.1	C-45
CARDTOT	G.1.2	G-6
CDM	E	E-3
	M.2	M-7
CDM patching	M.2	M-7
Character check field error	F.3.2	F-2
CHARACTER LOST -- REKEY	1.7	1-5
Check digit field error	F.3.2	F-2
Checkpoint	1.33	1-16
	1.34	1-17
CHKDG-07	C.4	C-45
	C.4.2	C-46
CHKDG-10	C.4	C-45
	C.4.3	C-48
CHKDG-11	C.4	C-45
	C.4.4	C-50
CHKPT CNTL	J.3	J-2
CMT patch procedures	M.1	M-1
CMT patching and corrections	M.1	M-1
CNTRL UNIT ADR	E	E-26
COBOL procedure	C.3	C-6
	C.4	C-44
	G.1.2	G-5
	H	H-1
	H.1	H-1
	H.3	H-3
	I.4.3	I-6
COBOL statements	C.1.3	C-4

Term	Reference	Page
Coding for grand total balancing	G.1.2	G-6
COLD START	1.9	1-6
COM	E	E-3
COMMAND ID COUNT	E	E-35
Command quick reference	D.5.1	D-4
COMMAND syntax	D.2	D-2
COMMANDS	D.5	D-4
Comparison of report writer and normal write	H.1	H-1
Comparison of report writer and sequential I/O	2	2-1
Compatibility	H.2	H-2
	2	2-1
	C.3.12	C-26
	G.2.3	G-16
Compatibility with previous levels	I.2	I-3
	2.1	2-1
COMPILE	1.18	1-10
COMPILE NAME	1.38	1-18
Compile time errors	F.3.3	F-2
Compile time and run time errors	F.3.3	F-3
Concurrent Enter/Verify mode	G.2.1	G-15
Controlling data communications	J.2	J-2
COPY	D.5.4	D-6
COPY command syntax	D.5.4	D-7
Copy FRM with only one tape unit	K.1	K-1
Copy FRM with two tape units	K.2	K-2
Copy software diagnostics	K.1.1	K-1
Copy the SMB	K.1.2	K-2
Correct disk module	E	E-54
CORRECTION COUNT BY RECORD (Y,N)	1.39	1-18
	1.45	1-21
	E	E-46
Correction count option	1.45	1-21
Corrections	5	5-1
CPU	6.2.3	6-5
Current configuration is:	E	E-33
Current IO assignments are:	E	E-33
Customer use	G.1.1	G-1

D

D=	E	E-7
D1 DCOM 1	E	E-34
D2 DCOM 2	E	E-34
D3 DCOM 3	E	E-34
D4 DCOM 4	E	E-35
Data block	I.1.1	I-2

Term	Reference	Page
Data communications	1.25	1-13
	1.50	1-22
	E	E-8
	F.3.4	F-3
	J.3	J-2
Data communications enhancements	1.50	1-22
Data communications errors	F.3.4	F-3
Data entry priority selection	1.9	1-6
Data set rate high?	E	E-14
DATATYPE	C.3	C-6
	C.3.2	C-8
	C.5	C-67
	E	E-50
DATE	1.19	1-11
	A.2.4	A-6
	C.4.1	C-45
	F.5	F-16
DATECONV	C.3	C-6
	C.3.3	C-10
DBL BUF MT WRITE	E	E-39
DCOM CNTL	J.2	J-2
DCOM PROTOCOLS	A.2.4	A-7
DDC	1.2	1-3
Default data COM controls?	E	E-14
Default format = any for SV WRITE and SV PRINT	1.13	1-7
DELETE	A.1.1	A-1
	A.2.2	A-4
DELETE command default change	1.38	1-18
Delete trailing blanks	E	E-17
Description	I	I-1
DEV-CODE	C.5	C-67
	C.5.1	C-67
DID	J.1	J-1
DIFFD	D.5.5	D-7
DIFFD command sequence	D.5.5	D-8
DIFFD command syntax	D.5.5	D-7
DIFFM	D.5.6	D-9
DIFFM command sequence	D.5.6	D-10
DIFFM command syntax	D.5.6	D-9
Directory construction at transmission time	1.50.1	1-22
DIRECTORY NAME	1.38	1-18
Disable 9600 baud clock?	E	E-14
Disk	1.51	1-23
	2.2	2-2
	6.2.3	6-5

Term	Reference	Page
Disk compatibility	2.2.2	2-3
	6.2.2	6-5
Disk data cache	1.2	1-3
Disk error logging	1.16	1-8
Disk error recovery	F.6.1	F-50
Disk full at 98 percent	1.8	1-6
Disk initiation for load in progress	E	E-1
DISK MGMT	A.1.2	A-1
	A.2.3	A-5
Disk record size in detailed status	1.46	1-21
	1.47	1-21
DISK RESERVE	1.52	1-24
Disk status codes	1.16	1-10
Disk throughput	1.2	1-3
Disk type	E	E-7
Disk utilization improved	1.52	1-23
Display-only records	1.31	1-15
DONE	1.27	1-14
	D.2	D-3
Double checkpoint	1.34	1-17
	E	E-38
DRS	1.46	1-21
	1.47	1-21
DS compile	1.17	1-10
DS compile with record numbers	1.17	1-10
DUMPM	D.5.7	D-11
DUMPM command sequence	D.5.7	D-11
DUMPM command syntax	D.5.7	D-11
DV	D.5.8	D-14
DV command sequence	D.5.8	D-15
DV command syntax	D.5.8	D-14
D3400 disk	1.16	1-10
	1.51	1-23

E

EBCDIC	I.3.1	I-5
EBCDIC characters	C.5	C-67
EBCDIC transparency?	E	E-16
EDIT	C.4	C-45
	C.4.5	C-53
Edit mask	C.4.5	C-54
END	H.3	H-3
ENTER MESSAGE removed	1.22	1-12
Enter mode	G.1.1	G-1

Term	Reference	Page
Enter mode limitations	G.1.1	G-2
EOF	D.5.9	D-15
EOF syntax	D.5.9	D-15
Error messages	F.1	F-1
Error types	F.3	F-1
Excluded capabilities	I.6	I-9
Extended memory option removed	1.42	1-20
Extended memory TWAs	1.41	1-20

F

F80, F128, F133 and F134	C.3	C-7
	C.3.8	C-22
F3282	E	E-6
F3283	E	E-6
F3285	E	E-6
FAST RECOVERY	1.35.2	1-17
FAST REPLAY	E	E-38
Field check errors	F.3.2	F-1
FIELD REL	E	E-1
Field release master break-out procedure	K	K-1
	4	4-1
Field release master tape	D	D-1
FIELD SEPARATOR CHAR=	E	E-35
File management	D	D-1
	H.2	H-2
File management modules	1.49	1-22
File management status codes	F.8.1	F-72
File status codes	F.8	F-71
FIND	C.5	C-66
FINISHED	H.3	H-3
FISTATUS	C.5	C-67
	C.5.2	C-69
FMSTATUS	C.5	C-67
	C.5.3	C-70
FORMAT	1.33	1-16
	C.3	C-7
	C.3.4	C-11
	C.4	C-44
FORMAT, default	1.13	1-7
FORMAT NOT FOUND	1.32	1-16
FORMAT NOT FOUND message	1.32	1-16
Format program compatibility	2.1.2	2-1
Format programming	G.1.2	G-3
Format programming errors	F.3.3	F-2

Term	Reference	Page
FORMAT PROGRAMS	2.1.2	2-1
	C.3	C-6
	C.3.2	C-8
Format programs listing	C.3	C-7
FORMATI	1.33	1-16
	2.1.2	2-1
	C.3	C-7
	C.3.5	C-17
FORMATIS	2.1.2	2-1
	C.3	C-7
	C.3.6	C-19
Formatted read facility	1.3	1-4
FREEFORM	C.3	C-7
	C.3.7	C-21
	H.3	H-2
	I.4.3	I-6
Freeform SEARCH	1.29	1-14
FRM	4	4-1
	K	K-1
	K.1	K-1
	K.1.1	K-1
	K.2	K-2
Function codes	1.16	1-8

G

GDS	E	E-3
General format requirements	I.4.3	I-6
General messages	F.5	F-8
Generate disk system	E	E-3
Gen-tape system	E	E-3
GET TIME	1.20	1-11
GTS	E	E-3
Guide messages	F.2	F-1

H

Hardware features	L	L-1
HEADER	D.5.10	D-15
HELP	D.4	D-4
	D.5.11	D-16
HELP command menu	D.5.11	D-16
HELP command syntax	D.5.11	D-16
Hex characters	E	E-16

Term	Reference	Page
HEX-CONV	C.4	C-45
	C.4.6	C-58
HEX2DEC	C.5	C-67
(DEC2#EX)	C.5.4	C-70
Hexadecimal	E	E-26
HI	E	E-13
Honeywell 6000 standard format	C.6.1	C-76
HOST INT TIMEOUT	E	E-27
Host interactivity	E	E-13
Host interactivity status codes	F.8.2	F-75
H6000	L.43	L-20
HW-TAB	C.5	C-67
	C.5.5	C-71
HW 6000SL	C.6	C-76
	C.6.1	C-77

I

I=	E	E-7
IBM label programs	I.3	I-3
IBM labels	C.6	C-77
IBM OS standard labels	C.6	C-77
IBM variable length records	I.11	I-7
	C.6.9	C-88
	I.4.1	I-5
IBM VLR	I	I-1
	I.2	I-3
	I.4.3	I-6
IBM VLR tape input	I.3.2	I-5
IBM VLR tape output	I.3.1	I-3
IBM VLR tapes	I.2	I-3
	I.3.1	I-3
	I.3.2	I-5
	I.4.1	I-5
	I.4.2	I-6
IBM 3270 protocol	C.5.10	C-73
IBMCR	C.6	C-76
	C.6.2	C-79
	I.2	I-3
IBMCR INVALID FOR OPEN INPUT	I	I-1
IBMCRV	C.6	C-76
	C.6.3	C-79
	I.2	I-3
IBMDL	C.6	C-76
	C.6.4	C-80
	I	I-1
	I.2	I-3

Term	Reference	Page
IBMDLV	C.6 C.6.5	C-76 C-81
IBMFD	I.2 C.6 C.6.6	I-3 C-76 C-83
IBMNL	C.6 C.6.7	C-76 C-85
IBMSL	C.6 C.6.8	C-76 C-86
IBMSLV	I I.2 C.6 C.6.9	I-1 I-3 C-76 C-88
ICL character set	I I.2	I-1 I-3
ICL-LBL	C.5.6 C.6 C.6.10	C-72 C-76 C-90
ICL-NEG	C.4 C.4.7	C-45 C-59
ICL-TAB	C.5 C.5.6	C-67 C-71
ICL29CR	C.6 C.6.11	C-76 C-92
ICL29SL	C.6 C.6.12	C-76 C-94
ICL 1900 Variable Length Records under report writer	1.5	1-5
ICL 1900 VLR	H H.3	H-1 H-2
ICL 1900 VLR with report writer	H	H-1
ICL 4X3 packed tapes	C.6.11 C.6.12	C-94 C-96
ID	E J.1	E-7 J-1
IF AMOUNT NOT >0....	C.4.8	C-60
IF RESTORE	2.1.1	2-1
Improved disk utilization procedures	1.52	1-23
INDEX	C.3 C.3.9	C-7 C-23
Index file creation	E	E-45
Indexed files	2.1.1	2-1
IND RES	E	E-45
INDIO	1.15	1-8
INDIO modules unloaded at poweroff	1.49	1-22
INDIO UNLOAD IN PROGRESS	1.49	1-22
INITIALIZE	G.1.2	G-6

Term	Reference	Page
INPUT	D.5.12	D-16
INPUT command syntax	D.5.12	D-17
INPUT DEV	E	E-33
Intercom	1.12	1-7
International Computers Limited (ICL) 1900 series labels	C.6.10	C-90
International Computers Limited (ICL) 2970 computer series	C.6.11 C.6.12	C-92 C-94
INVALID BLOCK SIZE	I.3.1	I-4
INVALID COMMAND SELECTED	I.3.1	I-5
INVALID TERMINAL NUMBER	1.44	1-21
IOTCL errors	F.9.2	F-79
IPL release	E	E-1
ISS disk	1.16 1.51	1-9 1-23

J

JOB NAME	1.38	1-18
JULIAN	C.4 C.4.8	C-45 C-60
Julian date	C.4 C.4.1 C.4.8	C-44 C-45 C-60

K

KEY BUF LTH IN WORDS	E	E-53
Keyed response required	F.4.2	F-5
Keying errors	C.3.4 C.3.9 C.3.12 C.3.13	C-14 C-23 C-37 C-38
Keystation messages	F.3.1	F-1
Keystations	J.4 1.25 3	J-2 1-13 3-1
Keystations, intercommunication	1.12	1-7
Keystroke buffer expanded	1.7	1-5

Term	Reference	Page
L		
LABEL and LABEL-TR	C.3 C.3.10	C-7 C-24
LABEL NAME	I.3.1	I-5
Label programs	C.6	C-75
Label programs listing	C.6	C-76
LABELS	I.3.1	I-5
Large capacity disk support	1.51	1-23
LC	E	E-3
LCD	1.51	1-23
LCD DRVR	1.51	1-23
LENGTH option	J.1	J-1
Level A	G.1.2	G-7
Level B	G.1.2	G-7
Level C	G.1.2	G-8
Level Z	G.1.2	G-8
Library format programs	C.3	C-6
Link back to checkpoint	1.36	1-18
LIST	D.5.13	D-17
LIST command sequence	D.5.13	D-18
LIST command syntax	D.5.13	D-17
LOAD	1.45 E	1-21 E-1
LOAD FILE MGMT AT STARTUP	E	E-45
Load master tape to disk?	E	E-1
LOAD SELECTED MODULES TO DISK	E	E-50
Load SYSGEN overlays to disk	E	E-2
LOAD TABLES FROM TAPE?	E	E-50
Loading specific library programs	C.1.2	C-2
Loading the program library	C.1.1	C-1
LOG AUTO MESSAGES	E	E-41
LOG BK MESSAGES	E	E-40
LOG DEV	E	E-33
Log disk errors	1.16	1-8
Log first record of datacom files	1.48	1-21
Log options	E	E-39
LOG SV ERRORS	E	E-40
Logical device definition	J.1	J-1
LOGICAL RECORD SIZE EXCEEDED	1.31	1-16
Logical unit numbering	D.3	D-3
Long-term compatibility considerations	2.3	2-3
LPP	D.5.14	D-20
LPP command syntax	D.5.14	D-20

Term	Reference	Page
M		
MAINTABL	C.3	C-6
MAKEMD	D	D-1
	D.5.15	D-20
MAKEMD command syntax	D.5.15	D-20
Mapped TWA space	1.41	1-20
Master version no. 11.R1.00	E	E-1
MAX COAX ADDR	E	E-36
MAX NO RCV BUFS	E	E-28
MAX NO XMT BUFS	E	E-28
MEM11	C.3	C-7
	C.3.11	C-24
MEM11S	C.4	C-45
	C.4.9	C-62
Memory	3	3-1
	6.2.1	6-1
Memory dump procedures	F.10	F-81
Memory usage by supervisor command	6.2.1	6-2
MERCHTOT	G.1.2	G-6
MESSAGE CANCELLED	1.40	1-20
Message format	F.4.2	F-5
Message queue	F.4.2	F-5
Message queuing	1.40	1-19
Message types	F.2	F-1
MESSAGES	F	F-1
MISC I/O	E	E-37
Mnemonic	D.5.1	D-4
Modulo 7	C.4.2	C-46
Modulo 10	C.4.3	C-48
Modulo 11	C.4.4	C-50
Modulo 64	C.5.5	C-71
	C.6.1	C-77
MONITOR	1.24	1-12
MONITOR enhancement	1.44	1-21
MOUNT	1.52	1-24
MOUNT OLD DISK	2.2.2	2-3
MOUNT TAPE:	E	E-55
MTS	1.41	1-20
MT1	K.1	K-1
Multi-point?	E	E-15
Multirecord?	E	E-19

Term	Reference	Page
N		
N	E	E-33
NC-SORT	1.2	1-3
	1.4	1-4
	C.3	C-7
	C.3.12	C-33
Negative number to be converted	C.4.7	C-59
New extended memory TWAS	1.6	1-5
New features	1	1-1
	1.3	1-4
No top of form on printer open	1.14	1-8
NO-DISK statement	1.31	1-15
Noncontiguous field sort	1.4	1-4
Normal write and input	I.4.3	I-6
NOSIM	D.5.16	D-1
NOSIM command syntax	D.5.16	D-21
NR OF KEYBUFS	E	E-52
NR OF PAGE BUFS	E	E-53
NR OF RCD BUFS	E	E-51
NR OF WORK BUFS	E	E-54
NTBL	C.5	C-67
	C.5.7	C-72
Number of COMM BUFS	E	E-27
Number of digits in hexadecimal number	C.4.6	C-59
Number of digits in the negative number	C.4.7	C-60
Number of subbatch registers	G.2.1	G-15
O		
O1 OUTPUT	E	E-33
O2 OUTPUT	E	E-33
OPEN OUTPUT statement	1.14	1-8
Operating procedures	G	G-1
Operating system	D	D-1
Operational considerations	A	A-1
Operational constraints	G.2	G-15
Operator interface	I.3	I-3
OPS-STATS	E	E-36
Optional features	E	E-37
OPTIONS	I.3.2	I-5
OUTPUT	D.5.17	D-21
OUTPUT command syntax	D.5.17	D-21

Term	Reference	Page
P		
P	D.5.18	D-22
P command syntax	D.5.18	D-22
Page eject	1.14	1-8
PAGE TWO	A.1.3	A-2
Parameter definitions	E	E-9
Parameter limitations	G.2.1	G-15
PASSWORD	E	F-35
Patch compatibility	2.1.4	2-2
Patches	2.1.4	2-2
	4	4-1
Patching errors	M.3	M-10
PERDEBUG	1.52	1-24
	D	D-1
Performance limitations	G.2.2	G-16
Peripheral interchangeability	3	3-1
Platter-spanning	1.52	1-23
PLTH	D.5.19	D-22
PLTH command syntax	D.5.19	D-22
Polling address	E	E-15
POWEROFF	1.49	1-22
Preinstallation planning	6.2.1	6-1
Press release key to continue	E	E-9
PRINT	1.2	1-4
	1.5	1-5
	1.13	1-7
	A.1.5	A-3
	A.2.5	A-9
Print compile	1.18	1-10
Print compile with page eject	1.18	1-10
Printer	1.14	1-8
Printing a source listing	C.2	C-5
Priorities	6.3.2	6-6
Problem reporting procedures	6.3	6-6
Problems	B	B-1
Procedure check field error	F.3.2	F-2
Processing errors	I.5	I-9
Processor message	J.4	J-2
PROCESSOR MESSAGE WAITING	J.4	J-3
Products executed to construct this product	2.2.1	2-2
Products executed with a program produced by this product	2.2.3	2-3
Products executed with this product	2.2.2	2-3
Programming for report writer	H.3	H-2
Programming requirements	I.4	I-5

Term	Reference	Page
R		
R	D.5.20	D-22
R command syntax	D.5.20	D-22
Range check field error	F.3.2	F-2
RCD BUF LTH IN WORDS	E	E-52
Reading IBM VLR tapes	I.4.2	I-6
READY	1.27	1-14
Rebroadcast background messages	1.23	1-12
REC FWD TO CONTINUE	1.27	1-14
REC LENGTH	I.3.1	I-4
RECEIVE	E	E-27
Record count	1.30	1-15
Record header	I.1.3	I-2
RECOVER	D.4	D-4
	D.5.21	D-23
RECOVER command sequence	D.5.21	D-23
RECOVER command syntax	D.5.21	D-23
References	preface	preface-1
Reformat of data at transmission time	1.50.2	1-22
Related products	2.2	2-2
Release history	6.1	6-1
Release media description	4	4-1
Remote identification	E	E-31
Report writer	1.5	1-5
	I.4.3	I-6
Reporting the error	6.3.1	6-6
Requested information messages	F.2	F-1
RESERVE statement	G.1.2	G-4
RESERVED WORD	G.2.3	G-16
Reset key response	F.4.1	F-4
Resource utilization	6.2	6-1
RESTORE	2.1.1	2-1
	C.1.1	C-2
	D.4	D-3
	D.5.22	D-24
	E	E-45
RESTORE command sequence	D.5.22	D-25
RESTORE command syntax	D.5.22	D-25
Restrictions	3	3-1
Reviewing the error	6.3.3	6-7
REW	D.5.23	D-25
REW command syntax	D.5.23	D-25
RID	1.25	1-13
	F	E-31
	J	J-1
RID ADDRESS	E	E-31
Run time errors	F.3.3	F-3
Runtime 6-character time	1.20	1-11

Term	Reference	Page
S		
Sample data entry	G.1.2	G-13
Sample data verification	G.1.2	G-14
Sample program	G.1.2	G-7
SAVE	D.5.24	D-26
SAVE command sequence	D.5.24	D-26
SAVE command syntax	D.5.24	D-26
SAVE DEVICE	E	E-44
SAVE statement	G.1.2	G-5
SAVE tape	2.2	2-2
SAVE TAPE FORM compatibility	2.2.2	2-3
SEARCH	1.2	1-4
	1.28	1-14
Search backward by number	1.28	1-14
SELECT	1.38	1-18
Select table =	E	E-8
SEND	E	E-27
Sequential I/O	1.14	1-8
Sequential I/O access to spool files	1.47	1-21
SET	1.9	1-6
	1.45	1-21
	A.1.4	A-2
	A.2.4	A-6
	E	E-29
SET=DATA COM	1.37	1-18
SET time and date option	1.19	1-11
Set U100 record size?	E	E-29
SETW DIAG modifications	1.26	1-13
SHIFT	1.9	1-6
	1.19	1-11
	A.2.4	A-6
SHIFT option	1.19	1-11
SID	1.25	1-13
	J	J-1
	E	E-29
SID ADDRESS	E	E-32
SIM	D	D-1
	D.4	D-3
	D.5.25	D-27
SIM command sequence	D.5.25	D-28
SIM command syntax	D.5.25	D-27
Single record replay	1.33	1-16
Site identification	E	E-32
SKF	D.5.26	D-28
SKF command syntax	D.5.26	D-29

Term	Reference	Page
SKIP	E	E-4
SMB	1.51	1-23
	E	E-1
Software diagnostics	1.26	1-13
	D	D-1
Software patching procedures	M	M-1
Software support	6.3.2	6-7
	6.3.3	6-7
	F.9.1	F-76
	F.10	F-81
Software user report (SUR)	6.3.1	6-6
SORT	C.3	C-7
	C.3.13	C-37
	E	E-45
SOURCE LISTINGS	C.2	C-4
Space compression on?	E	E-17
SPERRY general programming infor- mation exchange bulletin (UP-8012)	6.3.1	6-6
SPERRY XL/40 reference manual, XLOS 1.6, 680281-03	preface	preface-1
SPERRY 1900/10 and XL/40 distributed processing and capture systems	D	D-1
SPERRY 1900/10, system supervisor's reference, UP-9103	preface	preface-1
	1.3	1-4
Spool file	1.47	1-21
Spooling	E	E-34
STARTUP	1	1-1
	1.15	1-8
	1.42	1-20
	1.45	1-21
	1.52	1-24
	D.1	D-1
	E	E-45
	F.6.2	F-57
STATE-TB	C.5	C-67
	C.5.9	C-73
STATUS	1.46	1-21
Status code	F.8.1	F-72
	F.8.2	F-75
STD CNTRL WORD	E	E-43
STD equip and parameters?	E	E-7
ST-TABLE	C.5	C-67
	C.5.8	C-72
STX	J.4	J-3
Subbatch balance field	G.1.2	G-3
Subbatch balancing	1.1	1-3
	G	G-1
	G.1.2	G-3

Term	Reference	Page
Subbatch register assignments	G.2.1	G-15
Subbatch TWA memory size	G.2.1	G-16
SUBPROG	C.3	C-7
	C.3.14	C-39
SUBPROGRAMS	C.4	C-44
Subprograms listing	C.4	C-45
SUBSCRIPT OVERFLOW	C.4.1	C-46
Supervisor POWEROFF	D.1	D-2
SUPERVISOR work mode	C.1.1	C-1
SUPERVISOR WRITE	H.2	H-2
Support	6	6-1
Suppress print control?	E	E-15
SUR	6.3.1	6-6
SUR coordinator	6.3.1	6-6
SUR form	6.3.3	6-6
SUSPEND	J.2	J-2
SWITCH	J.2	J-2
SYMBIONT CONFIGURATION	J.6	J-3
SYSGEN	1.2	1-3
	1.37	1-18
	1.39	1-18
	G.1	G-1
SYSGEN error messages	F.9	F-76
SYSGEN error number	F.9.1	F-76
SYSGEN errors	F.9.1	F-76
SYSGEN procedures	E	E-1
	G.1.3	G-15
SYSGEN prompts	B.1	B-1
SYSGENS	J.6	J-3
System error dump procedure	F.10.1	F-81
System error messages	F.6	F-50
System error number	F.6.2	F-51
System errors	F.3.5	F-3
	F.6.2	F-52
System generation	E	E-1
System master binary	E	E-1

T

T3555	E	E-5
TABLE	C.3	C-7
	C.3.15	C-40
Table programs	C.5	C-66
Table programs listing	C.5	C-67
TABLES	E	E-8
Tape labels	I.3.1	I-5
Task error messages	F.7	F-65

Term	Reference	Page
Task error number	F.7.2	F-66
Task error recovery	F.7.1	F-65
Task errors	F.7.2	F-65
Temp file delete in startup	1.15	1-8
TERMINATE	J.2	J-2
TESTTAB	C.5	C-67
	C.5.10	C-73
The library	C.1	C-1
Three-cycle VERIFY option	1.39	1-18
Throughput	6.2.3	6-6
Time	1.19	1-11
Timer, fifteen-second	1.40	1-19
Tone at rcd end	E	E-37
Top of form	1.14	1-8
To select mods	E	E-2
TRANS	I.3.1	I-5
TRANSLAT	C.3	C-7
	C.3.16	C-42
	E	E-50
TRANSLAT WRITE	E	E-50
Translation	I.3.1	I-5
TWA	E	E-38
	G.1.2	G-4
TWAs	1.6	1-5
Type new assignment	E	E-27
Type SYSGEN command	E	E-54
TYPTBL	C.5	C-67
	C.5.11	C-74

U

Uniscope 100 protocol	1.25	1-13
Unrecoverable line errors	J.3	J-2
Use of the new COBOL verbs	G.1.2	G-4
User library	1.4	1-4
User's program library	C	C-1
	C.1	C-1
	C.1.1	C-1
User's program library tape	4	4-1
Using the program library	C.1.3	C-4
UTILITY	H.3	H-2
	I.4.3	I-7
	J	J-1

Term	Reference	Page
U100 datacom	1.25	1-13
	J	J-1
U100 protocol	1.25	1-13
	J.5	J-3
U100 screen	E	E-25
U200 screen	E	E-25

V

Value set field error	F.3.2	F-2
VERIFY	1.39	1-18
Verify mode	F.5	F-14
	G.1.1	G-2
VFU-FORM	C.3	C-7
	C.3.17	C-43
Volume table of contents	D	D-1
VTOC	D	D-1
	D.5.27	D-29
VTOC command sequence	D.5.27	D-30
VTOC command syntax	D.5.27	D-30

W

WARM AUTOMOUNT	1.52	1-24
WARM START enhancements	1.35	1-17
WARM START fast recovery	1.35.2	1-17
WARM START speed-up	1.35.1	1-17
WARM SYSLOAD	1.52	1-24
Winchester technology disk	1.51	1-23
WORK INIT	E	E-37
WPL	D.5.28	D-31
WPL command syntax	D.5.28	D-31
WRITE	1.11	1-7
	1.13	1-7
	1.30	1-15
	A.1.5	A-3
	A.2.5	A-9
WRITE CONTINUE--TOTAL RECORD COUNT	1.30	1-15
WRITE CONTINUE for H6000	1.43	1-20
Writing IBM VLR tapes	I.4.1	I-5

Term	Reference	Page
X		
X	D.5.29	D-31
X command syntax	D.5.29	D-31
XL/40 operational considerations	A.2	A-3
XL/40 system only	L.3	L-2
Y		
Y	E	E-4
YNTBL	C.5	C-67
	C.5.12	C-74
YTBL	C.5	C-67
	C.5.13	C-75
2=minor	6.3.2	6-7
3=serious	6.3.2	6-7
4=critical	6.3.2	6-7
1100 SYSGEN	J.6	
1900 VLR	H	H-1
	H.3	H-2
1900/10 and XL/40 systems	L.1	L-1
1900/10 system only	L.2	L-1
1900/10 users	A.1	A-1

FLAKE A FR

READER'S COMMENT SHEET

Your comments on this manual will help us improve it. Please fill in the requested information.

Name of manual: _____

Manual number: UP-_____ revision number _____, including update numbers: _____

Name of your company: _____

Address of company: _____

What is your position? _____

Your level of experience: Professional _____ Knowledgeable _____ Novice _____

With what system is the equipment used? _____

How do you use this manual?

As a reference source

As a self-instructional text

As a classroom text

As _____

Please rate this manual

As a reference source:

Good

Adequate

Not adequate

As a text:

Good

Adequate

Not adequate

For other uses:

Good

Adequate

Not adequate

Add your specific comments. Give page and paragraph references where appropriate.

Thank you for your cooperation.

PLEASE SEND US YOUR COMMENTS!


We feel that this manual is an essential part of our equipment. We want to be sure it is the best, most usable manual possible. Your comments will help us achieve this goal. When you have become familiar with the manual, please fill in the other side of this form and mail the form to us. Your reply will be carefully reviewed by the persons responsible for writing and publishing this manual.

FOLD

FIRST CLASS
PERMIT NO. 2540
SALT LAKE CITY,
UTAH

BUSINESS REPLY MAIL NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY

SPERRY  **UNIVAC**
322 NORTH 2200 WEST
SALT LAKE CITY, UTAH 84116

ATTN: MANAGER, GSD PUBLICATIONS

FOLD

NOTE:

Requests for copies of this manual and other Sperry Univac publications and for assistance in getting the most use out of your Sperry Univac equipment should be directed to your local Sperry Univac representative.

CUT