SY33-8571-6
File No. S370-20

Systems

DOS/VS Handbook

Volume 1

Release 34

IBM

Preface

This manual is the first in a series of two volumes. The reference information contained in these two manuals is provided as a DOS/VS serviceability aid and is, therefore, a summary of other DOS/VS documentation. These manuals are intended for use by persons involved in program support.

This manual does not contain information on Advanced Functions DOS/VS. This information represent licensed material and must be ordered separatly.

It will be distributed as System Library Supplement which logically fit into this manual.

The two volumes contain the following information:

- Volume 1, SY33-8571:
  Chapter I : System/370 General Information
  II : DOS/VS General Information
  III: DOS/VS IOCS (General, SAM, DAM, ISAM)
  IV: DOS/VS Supervisor Control Blocks and Areas
  V : DOS/VS Service Aids
- Volume 2, SY33-8572:
  Chapter I : POWER/VS
  II : VTAM Control Blocks
  III: VSAM Control Blocks
  IV: Model 20 Emulator
  V : 14xx Emulator
  VI: BTAM

If there is any discrepancy between the information contained in this manual and the DOS/VS optional programming material (e.g., PLMs and listings), the latter is assumed to be correct.

Seven Edition (July, 1977)

This is a major revision of, and obsoletes, SY33-8571-5. It applies to Version 6, Release 34, of the IBM Disk Operating System/Virtual Storage, DOS/VS, and to all subsequent versions and releases until otherwise indicated in new editions or Technical Newsletters. Changes are continually made to the information herein; before using this publication in connection with the operation of IBM systems, consult the latest *System/370 Bibliography*, GC20-0001, for the editions that are applicable and current.

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

A handbook-sized binder, FE Part Number 453559, may be purchased from IBM. Customers may order it through their IBM marketing representative. IBM personnel should order it as an FE part from Mechaniesburg.

This manual has been prepared by WT-DP/CE Technical Operations, Poststraße 40, 7030 Boeblingen, Germany.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be sent to the above address. Comments become the property of IBM.

TABLE OF CONTENTS

TABLE OF CONTENTS (continued)

CHAPTER III DOS/VS IOCS (GENERAL/SAM/DAM/ISAM) (continued)

CHAPTER IV DOS/VS SUPERVISOR CONTROL BLOCKS AND AREAS

TABLE OF CONTENTS (continued)

CHAPTER IV DOS/VS SUPERVISOR CONTROL BLOCK AND AREAS (continued)

TABLE OF CONTENTS (continued)

CHAPTER V DOS/VS Service AIDS

CHAPTER I
SYSTEM 370 GENERAL INFORMATION

MACHINE INSTRUCTIONS

Name	Mnemonic	Op Code	Format	Operands
Add (c)	AR	1A	RR	R1,R2
Add (c)	A	5A	RX	R1,D2(X2,B2)
Add Decimal (c)	AP	FA	SS	D1(L1,B1),D2(L2,B2)
Add Halfword (c)	AH	4A	RX	R1,D2(X2,B2)
Add Logical (c)	ALR	1E	RR	R1,R2
Add Logical (c)	AL	5E	RX	R1,D2(X2,B2)
AND (c)	NR	14	RR	R1,R2
AND (c)	N	54	RX	R1,D2(X2,B2)
AND (c)	NI	94	SI	D1(B1),I2
AND (c)	NC	D4	SS	D1(L,B1),D2(B2)
Branch and Link	BALR	05	RR	R1,R2
Branch and Link	BAL	45	RX	R1,D2(X2,B2)
Branch on Condition	BCR	07	RR	M1,R2
Branch on Condition	BC	47	RX	M1,D2(X2,B2)
Branch on Count	BCTR	06	RR	R1,R2
Branch on Count	BCT	46	RX	R1,D2(X2,B2)
Branch on Index High	BXH	86	RS	R1,R3,D2(B2)
Branch on Index Low or Equal	BXLE	87	RS	R1,R3,D2(B2)
Clear I/O	CLRIO	9D01	S	D2(B2)
Compare (c)	CR	19	RR	R1,R2
Compare (c)	C	59	RX	R1,D2(X2,B2)
Compare and Swap	CS	BA	RS	R1,R3,D2,(B2)
Compare Decimal (c)	CP	F9	SS	D1(L1,B1),D2(L2,B2)
Compare Double and Swap	CDS	BB	RS	R1,R3,D2,(B2)
Compare (c)	CR	19	RR	R1,R2
Compare (c)	C	59	RX	R1,D2(X2,B2)
Compare Decimal (c)	CP	F9	SS	D1(L1,B1),D2(L2,B2)
Compare Halfword (c)	CH	49	RX	R1,D2(X2,B2)
Compare Logical (c)	CLR	15	RR	R1,R2
Compare Logical (c)	CL	55	RX	R1,D2(X2,B2)
Compare Logical (c)	CLC	D5	SS	D1,(L,B1),D2(B2)
Compare Logical (c)	CLI	95	SI	D1(B1),I2
Compare Logical Characters under Mask (c)	CLM	BD	RS	R1,M3,D2(B2)
Compare Logical Long (c)	CLCL	0F	RR	R1,R2
Convert to Binary	CVB	4F	RX	R1,D2(X2,B2)
Convert to Decimal	CVD	4E	RX	R1,D2(X2,B2)
Diagnose (p)		83	SI	
Divide	DR	1D	RR	R1,R2
Divide	D	5D	RX	R1,D2(X2,B2)
Divide Decimal	DP	FD	SS	D1(L1,B1),D2(L2,B2)
Edit (c)	ED	DE	SS	D1(L,B1),D2(B2)
Edit and Mark (c)	EDMK	DF	SS	D1(L,B1),D2(B2)
Exclusive OR (c)	XR	17	RR	R1,R2
Exclusive OR (c)	X	57	RX	R1,D2(X2,B2)
Exclusive OR (c)	XI	97	SI	D1(B1),I2
Exclusive OR (c)	XC	D7	SS	D1(L,B1),D2(B2)
Execute	EX	44	RX	R1,D2(X2,B2)
Halt I/O (c,p)	HIO	9E00	S	D1(B1)
Halt Device (c,p)	HDV	9E01	S	D1(B1)
Insert Character	IC	43	RX	R1,D2(X2,B2)
Insert Characters under Mask (c)	ICM	BF	RS	R1,M3,D2(B2)
Insert Storage Key (p)	ISK	09	RR	R1,R2
Load	LR	18	RR	R1,R2
Load	L	58	RX	R1,D2(X2,B2)
Load Address	LA	41	RX	R1,D2(X2,B2)
Load and Test (c)	LTR	12	RR	R1,R2
Load Complement (c)	LCR	13	RR	R1,R2
Load Control (p)	LCTL	B7	RS	R1,R3,D2(B2)
Load Halfword	LH	48	RX	R1,D2(X2,B2)
Load Multiple	LM	98	RS	R1,R3,D2(B2)
Load Negative (c)	LNR	11	RR	R1,R2

MACHINE INSTRUCTIONS (....Cont'd)

Name	Mnemonic	Op Code	Format	Operands
Load Positive (c)	LPR	10	RR	R1,R2
Load PSW (n,p)	LPSW	82	SI	D1(B1)
Load Real Address (c,p)	LRA	B1	RX	R1,D2(X2,B2)
Monitor Call (m)	MC	AF	SI	D1(B1),I2
Move	MVI	92	SI	D1(B1),I2
Move	MVC	D2	SS	D1(L,B1),D2(B2)
Move Long (c)	MVCL	0E	RR	R1,R2
Move Numerics	MVN	D1	SS	D1(L,B1),D2(B2)
Move with Offset	MVO	F1	SS	D1(L1,B1),D2(L2,B2)
Move Zones	MVZ	D3	SS	D1(L,B1),D2(B2)
Multiply	MR	1C	RR	R1,R2
Multiply	M	5C	RX	R1,D2(X2,B2)
Multiply Decimal	MP	FC	SS	D1(L1,B1),D2(L2,B2)
Multiply Halfword	MH	4C	RX	R1,D2(X2,B2)
OR(c)	OR	16	RR	R1,R2
OR (c)	O	56	RX	R1,D2(X2,B2)
OR (c)	OI	96	SI	D1(B1),I2
OR (c)	OC	D6	SS	D1(L,B1),D2(B2)
Pack	PACK	F2	SS	D1(L1,B1),D2(L2,B2)
Purge Table (p)	PTLB	B20D	S	
Read Direct (a,p)	RDD	85	SI	D1(B1),I2
Reset Reference Bit (c,p)	RRB	B213	S	D1(B1)
Set Clock (c,p)	SCK	B204	S	D1(B1)
Set Clock Comparator (p)	SCKC	B206	S	D1(B1)
Set CPU Timer (p)	SPT	B208	S	D1(B1)
Set Program Mask (n)	SPM	04	RR	R1
Set Storage Key (p)	SSK	08	RR	R1,R2
Set System Mask (p)	SSM	80	SI	D1(B1)
Shift and Round Decimal (c)	SRP	F0	SS	D1(L1,B1),D2(B2),I3
Shift Left Double (c)	SLDA	8F	RS	R1,D2(B2)
Shift Left Double Logical	SLDL	8D	RS	R1,D2(B2)
Shift Left Single (c)	SLA	8B	RS	R1,D2(B2)
Shift Left Single Logical	SLL	89	RS	R1,D2(B2)
Shift Right Double (c)	SRDA	8E	RS	R1,D2(B2)
Shift Right Double Logical	SRDL	8C	RS	R1,D2(B2)
Shift Right Single (c)	SRA	8A	RS	R1,D2(B2)
Shift Right Single Logical	SRL	88	RS	R1,D2(B2)
Start I/O (c,p)	SIO	9C00	S	D1(B1)
Start I/O Fast Release (c,p)	SIOF	9C01	S	D1(B1)
Store	ST	50	RX	R1,D2(X2,B2)
Store Channel ID (c,p)	STIDC	B203	S	D1(B1)
Store Character	STC	42	RX	R1,D2(X2,B2)
Store Characters under Mask	STCM	BE	RS	R1,M3,D2(B2)
Store Clock (c)	STCK	B205	S	D1(B1)
Store Clock Comparator (p)	STCKC	B207	S	D1(B1)
Store Control (p)	STCTL	B6	RS	R1,R3,D2(B2)
Store CPU ID (p)	STIDP	B202	S	D1(B1)
Store CPU Timer (p)	STPT	B209	S	D1(B1)
Store Halfword	STH	40	RX	R1,D2(X2,B2)
Store Multiple	STM	90	RS	R1,R3,D2(B2)
Store then AND System Mask (p)	STNSM	AC	SI	D1(B1),I2
Store then OR System Mask (p)	STOSM	AD	SI	D1(B1),I2
Subtract (c)	SR	1B	RR	R1,R2
Subtract (c)	S	5B	RX	R1,D2(X2,B2)
Subtract Decimal (c)	SP	FB	SS	D1(L1,B1),D2(L2,B2)
Subtract Halfword (c)	SH	4B	RX	R1,D2(X2,B2)
Subtract Logical (c)	SLR	1F	RR	R1,R2
Subtract Logical (c)	SL	5F	RX	R1,D2(X2,B2)
Supervisor Call	SVC	0A	RR	I
Test and Set (c)	TS	93	SI	D1(B1)

MACHINE INSTRUCTIONS (....Cont'd)

Name	Mnemonic	Op Code	Format	Operands
Test Channel (c,p)	TCH	9F	SI	D1(B1)
Test I/O (c,p)	TIO	9D	SI	D1(B1)
Test under Mask (c)	TM	91	SI	D1(B1),I2
Translate	TR	DC	SS	D1(L,B1),D2(B2)
Translate and Test (c)	TRT	DD	SS	D1(L,B1),D2(B2)
Unpack	UNPK	F3	SS	D1(L1,B1),D2(L2,B2)
Write Direct (a,p)	WRD	84	SI	D1(B1),I2
Zero and Add Decimal (c)	ZAP	F8	SS	D1(L1,B1),D2(L2,B2)

Floating Point Instructions

Name	Mnemonic	Op Code	Format	Operands
Add Normalized, Extended (c,x)	AXR	36	RR	R1,R2
Add Normalized, Long (c)	ADR	2A	RR	R1,R2
Add Normalized, Long (c)	AD	6A	RX	R1,D2(X2,B2)
Add Normalized, Short (c)	AER	3A	RR	R1,R2
Add Normalized, Short (c)	AE	7A	RX	R1,D2(X2,B2)
Add Unnormalized, Long (c)	AWR	2E	RR	R1,R2
Add Unnormalized, Long (c)	AW	6E	RX	R1,D2(X2,B2)
Add Unnormalized, Short (c)	AUR	3E	RR	R1,R2
Add Unnormalized, Short (c)	AU	7E	RX	R1,D2(X2,B2)
Compare, Long (c)	CDR	29	RR	R1,R2
Compare, Long (c)	CD	69	RX	R1,D2(X2,B2)
Compare, Short (c)	CER	39	RR	R1,R2
Compare, Short (c)	CE	79	RX	R1,D2(X2,B2)
Divide, Long	DDR	2D	RR	R1,R2
Divide, Long	DD	6D	RX	R1,D2(X2,B2)
Divide, Short	DER	3D	RR	R1,R2
Divide, Short	DE	7D	RX	R1,D2(X2,B2)
Halve, Long	HDR	24	RR	R1,R2
Halve, Short	HER	34	RR	R1,R2
Load and Test, Long (c)	LTDR	22	RR	R1,R2
Load and Test, Short (c)	LTER	32	RR	R1,R2
Load Complement, Long (c)	LCDR	23	RR	R1,R2
Load Complement, Short (c)	LCER	33	RR	R1,R2
Load, Long	LDR	28	RR	R1,R2
Load, Long	LD	68	RX	R1,D2(X2,B2)
Load Negative, Long (c)	LNDR	21	RR	R1,R2
Load Negative, Short (c)	LNER	31	RR	R1,R2
Load Positive, Long (c)	LPDR	20	RR	R1,R2
Load Positive, Short (c)	LPER	30	RR	R1,R2
Load Rounded, Extended to Long(x)	LRDR	25	RR	R1,R2
Load Rounded, Long to Short (x)	LRER	35	RR	R1,R2
Load, Short	LER	38	RR	R1,R2
Load, Short	LE	78	RX	R1,D2(X2,B2)
Multiply, Extended (x)	MXR	26	RR	R1,R2
Multiply, Long	MDR	2C	RR	R1,R2
Multiply, Long	MD	6C	RX	R1,D2(X2,B2)
Multiply, Long/Extended (x)	MXDR	27	RR	R1,R2
Multiply, Long/Extended (x)	MXD	67	RX	R1,D2(X2,B2)
Multiply, Short	MER	3C	RR	R1,R2
Multiply, Short	ME	7C	RX	R1,D2(X2,B2)
Store, Long	STD	60	RX	R1,D2(X2,B2)
Store, Short	STE	70	RX	R1,D2(X2,B2)
Subtract Normalized, Ext'd (c,x)	SXR	37	RR	R1,R2
Subtract Normalized, Long (c)	SDR	2B	RR	R1.R2
Subtract Normalized, Long (c)	SD	6B	RX	R1,D2(X2,B2)

Floating Point Instructions (....Cont'd)

Name	Mnemonic	Op Code	Format	Operands
Subtract Normalized, Short (c)	SER	3B	RR	R1,R2
Subtract Normalized, Short (c)	SE	7B	RX	R1,D2(X2,B2)
Subtract Unnormalized, Long (c)	SWR	2F	RR	R1,R2
Subtract Unnormalized, Long (c)	SW	6F	RX	R1,D2(X2,B2)
Subtract Unnormalized, Short (c)	SUR	3F	RR	R1,R2
Subtract Unnormalized, Short (c)	SU	7F	RX	R1,D2(X2,B2)

(a) Direct Control Feature
(c) Condition Code is set
(m) Monitoring Feature
(n) New Condition Code is loaded
(p) Privileged Instruction
(x) Extended precision floating point feature

EXTENDED MNEMONIC INSTRUCTION CODES

GENERAL

Extended Code		Machine Instruction		Meaning
B	D2(X2,B2)	BC 15,	D2(X2,B2)	Branch Unconditionally
BR	R2	BCR 15,	R2	Branch Unconditionally
NOP	D2(X2,B2)	BC 0,	D2(X2,B2)	No Operation
NOPR	R2	BCR 0,	R2	No Operation (RR)

AFTER COMPARE INSTRUCTIONS (A:B)

BH	D2(X2,B2)	BC 2,	D2(X2,B2)	Branch on A High
BL	D2(X2,B2)	BC 4,	D2(X2,B2)	Branch on A Low
BE	D2(X2,B2)	BC 8,	D2(X2,B2)	Branch on A equal B
BNH	D2(X2,B2)	BC 13,	D2(X2,B2)	Branch on A not High
BNL	D2(X2,B2)	BC 11,	D2(X2,B2)	Branch on A not Low
BNE	D2(X2,B2)	BC 7,	D2(X2,B2)	Branch on A not Equal B

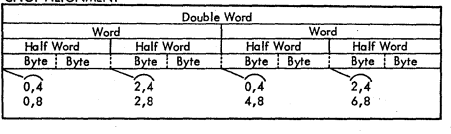
AFTER ARITHMATIC INSTRUCTIONS

BO	D2(X2,B2)	BC 1,	D2(X2,B2)	Branch on Overflow
BP	D2(X2,B2)	BC 2,	D2(X2,B2)	Branch on Plus
BM	D2(X2,B2)	BC 4,	D2(X2,B2)	Branch on Minus
BZ	D2(X2,B2)	BC 8,	D2(X2,B2)	Branch on Zero
BNP	D2(X2,B2)	BC 13,	D2(X2,B2)	Branch on not Plus
BNM	D2(X2,B2)	BC 11,	D2(X2,B2)	Branch on not Minus
BNZ	D2(X2,B2)	BC 7,	D2(X2,B2)	Branch on not Zero

AFTER TEST UNDER MASK INSTRUCTIONS

BO	D2(X2,B2)	BC 1,	D2(X2,B2)	Branch if Ones
BM	D2(X2,B2)	BC 4,	D2(X2,B2)	Branch if Mixed
BZ	D2(X2,B2)	BC 8,	D2(X2,B2)	Branch if Zero's
BNO	D2(X2,B2)	BC 14,	D2(X2,B2)	Branch if not Ones

CNOP ALIGNMENT



EDIT AND EDMK PATTERN CHARACTERS (In Hex)

20 - Digit selector	40 - Blank	5C - Aterisk
21 - Start of significance	4B - Period	6B - Comma
22 - Field separator	5B - Dollar sign	C3D9-CR

ASSEMBLER INSTRUCTIONS

Function	Mnemonic	Meaning
Data Definition	DC	Define Constant
	DS	Define Storage
	CCW	Define Channel Command Word
Program Sectioning and Linking	START	Start Assembly
	CSECT	Identify Control Section
	DSECT	Identify Dummy Section
	COM	Identify blank common Control Section
	ENTRY	Identify Entry Point Symbol
	EXTRN	Identify External Symbol
	WXTRN	Identify weak External Symbol
Base Register Assignment	USING	Use Base Address Register
	DROP	Drop Base Address Register
Control of listings	TITLE	Identify Assembly Output
	EJECT	Start new Page
	SPACE	Space Listing
	PRINT	Print Optional Data
Program Control	ICTL	Input Format Control
	ISEQ	Input Sequence Checking
	PUNCH	Punch a Card
	REPRO	Reproduce following Card
	ORG	Set Location Counter
	EQU	Equate Symbol
	LTORG	Begin Literal Pool
	CNOP	Conditional No-Operation
	COPY	Copy predefined Source Coding
	END	End Assembly
Macro Definition	MACRO	Macro Definition Header
	MNOTE	Request for Error Message
	MEXIT	Macro Definition Exit
	MEND	Macro Definition Trailer
Conditional Assembly	ACTR	Conditional Assembly Loop Counter
	AGO	Unconditional Branch
	AIF	Conditional Branch
	ANOP	Assembly No-Operation
	GBLA	Define global SETA Symbol
	GBLB	Define global SETB Symbol
	GBLC	Define global SETC Symbol
	LCLA	Define local SETA Symbol
	LCLB	Define local SETB Symbol
	LCLC	Define local SETC Symbol
	SETA	Set Arithmetic Variable Symbol
	SETB	Set Arithmetic
	SETC	Set Character Variable Symbol

SUMMARY OF CONSTANTS

Type	Implied Length Bytes	Alignment	Format	Truncation Padding
C	-	Byte	Characters	Right
X	-	Byte	Hexadecimal Digits	Left
B	-	Byte	Binary Digits	Left
F	4	Word	Fixed-point Binary	Left
H	2	Halfword	Fixed-point Binary	Left
E	4	Word	Short Floating-point	Right
D	8	Doubleword	Long Floating-point	Right
L	16	Doubleword	Extended Floating-point	Right
P	-	Byte	Packed Decimal	Left
Z	-	Byte	Zoned Decimal	Left
A	4	Word	Value of Address	Left
Y	2	Halfword	Value of Address	Left
S	2	Halfword	Address in Base-Displacement form	-
V	4	Word	Externally defined Address Value	Left

CONDITION CODES

Condition Code Setting Mask Bit Position	0 8	1 4	2 2	3 1
Floating Point Arithmetic				
Add Normalized S/L/E	zero	< zero	> zero	-
Add Unnormalized S/L	zero	< zero	> zero	-
Compare S/L (A:B)	equal	A low	A high	-
Load and Test S/L	zero	< zero	> zero	-
Load Complement S/L	zero	< zero	> zero	-
Load Negative S/L	zero	< zero	-	-
Load Positive S/L	zero	-	> zero	-
Subtract Normalized S/L/E	zero	< zero	> zero	-
Subtract Unnormalized S/L	zero	< zero	> zero	-
Fixed Point and Decimal Arithmetic				
Add H/F/Dec.	zero	< zero	> zero	overflow
Add Logical	zero, no carry	not zero, no carry	zero, carry	not zero, carry
Compare H/F/Dec. (A:B)	equal	A low	A high	-
Compare and Swap/Double	equal	not equal	-	-
Load and Test	zero	< zero	> zero	-
Load Complement	zero	< zero	> zero	overflow
Load Negative	zero	< zero	-	-
Load Positive	zero	-	> zero	overflow
Shift and Round Decimal	zero	< zero	> zero	overflow
Shift Left Single/Double	zero	< zero	> zero	overflow
Shift Right Single/Double	zero	< zero	> zero	-
Subtract H/F/Dec.	zero	< zero	> zero	overflow
Subtract Logical	-	not zero, no carry	zero, carry	not zero, carry
Zero and Add	zero	< zero	> zero	overflow

CONDITION CODES (....Cont'd)

Condition Code Setting	0	1	2	3
Mask Bit Position	8	4	2	1
Logical Operations				
AND	zero	not zero	-	-
Compare Logical (A:B)	equal	A low	A high	-
Edit	zero	<zero	> zero	-
Edit and Mark	zero	<zero	> zero	-
Exclusive OR	zero	not zero	-	-
Insert Characters under Mask	all zero	1st bit one	1st bit zero	-
Move Long (A:B)	equal	A low	A high	overlap
OR	zero	not zero	-	-
Test under Mask	zero	mixed	-	one
Translate and Test	zero	incomplete	complete	-
Input/Output Operations				
Clear I/O	no oper in	CSW stored	chan busy	not oper
Halt I/O, Halt Device	interruption pending	CSW stored	See Prin Op	not oper
Start I/O, SIOF	started	CSW stored	busy	not oper
Store Channel ID	ID stored	CSW stored	ID not stored	not oper
Test I/O	available	CSW stored	busy	not oper
Test Channel	available	interruption pending	burst mode	not oper
Miscellaneous Operations				
Set Clock	set	secure	-	not oper
Store Clock	set	not set	error	not oper
Test and set	zero	one	-	-
Load Real Address	Translation available	Segment table entry invalid	Page table entry invalid	Segment- or Page table length violation
Reset Reference Bit	Ref bit zero, change bit zero	Ref bit zero, change bit one	Ref bit one, change bit zero	Ref bit one, change bit one
Signal Processor	accepted	stat stored	busy	not oper

CODE TRANSLATION TABLE

Dec	Hex	Instruction (RR)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
0	00			NUL	NUL		12-0-1-8-9	0000 0000
1	01			SOH	SOH		12-1-9	0000 0001
2	02			STX	STX		12-2-9	0000 0010
3	03			ETX	ETX		12-3-9	0000 0011
4	04	SPM		PF	EOT		12-4-9	0000 0100
5	05	BALR		HT	ENO		12-5-9	0000 0101
6	06	BCTR		LC	ACK		12-6-9	0000 0110
7	07	BCR		DEL	BEL		12-7-9	0000 0111
8	08	SSK			BS		12-8-9	0000 1000
9	09	ISK			HT		12-1-8-9	0000 1001
10	0A	SVC		SMM	LF		12-2-8-9	0000 1010
11	0B			VT	VT		12-3-8-9	0000 1011
12	0C			FF	FF		12-4-8-9	0000 1100
13	0D			CR	CR		12-5-8-9	0000 1101
14	0E	MVCL		SO	SO		12-6-8-9	0000 1110
15	0F	CLCL		SI	SI		12-7-8-9	0000 1111
16	10	LPR		DLE	DLE		12-11-1-8-9	0001 0000
17	11	LNR		DC1	DC1		11-1-9	0001 0001
18	12	LTR		DC2	DC2		11-2-9	0001 0010
19	13	LCR		TM	DC3		11-3-9	0001 0011
20	14	NR		RES	DC4		11-4-9	0001 0100
21	15	CLR		NL	NAK		11-5-9	0001 0101
22	16	OR		BS	SYN		11-6-9	0001 0110
23	17	XR		IL	ETB		11-7-9	0001 0111
24	18	LR		CAN	CAN		11-8-9	0001 1000
25	19	CR		EM	EM		11-1-8-9	0001 1001
26	1A	AR		CC	SUB		11-2-8-9	0001 1010
27	1B	SR		CU1	ESC		11-3-8-9	0001 1011
28	1C	MR		IFS	FS		11-4-8-9	0001 1100
29	1D	DR		IGS	GS		11-5-8-9	0001 1101
30	1E	ALR		IRS	RS		11-6-8-9	0001 1110
31	1F	SLR		IUS	US		11-7-8-9	0001 1111
32	20	LPDR		DS	SP		11-0-1-8-9	0010 0000
33	21	LNDR		SOS	!		0-1-9	0010 0001
34	22	LTDR		FS	"		0-2-9	0010 0010
35	23	LCDR			#		0-3-9	0010 0011
36	24	HDR		BYP	$		0-4-9	0010 0100
37	25	LRDR		LF	%		0-5-9	0010 0101
38	26	MXR		ETB	&		0-6-9	0010 0110
39	27	MXDR		ESC	'		0-7-9	0010 0111
40	28	LDR			(		0-8-9	0010 1000
41	29	CDR			)		0-1-8-9	0010 1001
42	2A	ADR		SM	*		0-2-8-9	0010 1010
43	2B	SDR		CU2	+		0-3-8-9	0010 1011
44	2C	MDR			,		0-4-8-9	0010 1100
45	2D	DDR		ENQ	-		0-5-8-9	0010 1101
46	2E	AWR		ACK	.		0-6-8-9	0010 1110
47	2F	SWR		BEL	/		0-7-8-9	0010 1111
48	30	LPER			0		12-11-0-1-8-9	0011 0000
49	31	LNER			1		1-9	0011 0001
50	32	LTER		SYN	2		2-9	0011 0010
51	33	LCER			3		3-9	0011 0011
52	34	HER		PN	4		4-9	0011 0100
53	35	LRER		RS	5		5-9	0011 0101

CODE TRANSLATION TABLE (....Cont'd)

Dec	Hex	Instruction (RR)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
54	36	AXR		UC	6		6-9	0011 0110
55	37	SXR		EOT	7		7-9	0011 0111
56	38	LER			8		8-9	0011 1000
57	39	CER			9		1-8-9	0011 1001
58	3A	AER			:		2-8-9	0011 1010
59	3B	SER		CU3	;		3-8-9	0011 1011
60	3C	MER		DC4	<		4-8-9	0011 1100
61	3D	DER		NAK	=		5-8-9	0011 1101
62	3E	AUR			>		6-8-9	0011 1110
63	3F	SUR		SUB	?		7-8-9	0011 1111

1 EBCDIC graphics shown are standard bit pattern assignments. For specific print train/chain: See printer manual.
2 Add C (check bit) for odd or even parity as needed, exept as noted.
3 For even parity use CA

Dec	Hex	Instruction (RX)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
64	40	STH		SP	@	(3)	no punches	0100 0000
65	41	LA			A		12-0-1-9	0100 0001
66	42	STC			B		12-0-2-9	0100 0010
67	43	IC			C		12-0-3-9	0100 0011
68	44	EX			D		12-0-4-9	0100 0100
69	45	BAL			E		12-0-5-9	0100 0101
70	46	BCT			F		12-0-6-9	0100 0110
71	47	BC			G		12-0-7-9	0100 0111
72	48	LH			H		12-0-8-9	0100 1000
73	49	CH			I		12-1-8	0100 1001
74	4A	AH		¢	J		12-2-8	0100 1010
75	4B	SH	.	.	K	B A 8 2 1	12-3-8	0100 1011
76	4C	MH	¤	<	L	B A 8 4	12-4-8	0100 1100
77	4D		(	(	M	B A 8 4 1	12-5-8	0100 1101
78	4E	CVD	<	+	N	B A 8 4 2	12-6-8	0100 1110
79	4F	CVB	≠	I	O	B A 8 4 2 1	12-7-8	0100 1111
80	50	ST	&+	&	P	B A	12	0101 0000
81	51				Q		12-11-1-9	0101 0001
82	52				R		12-11-2-9	0101 0010
83	53				S		12-11-3-9	0101 0011
84	54	N			T		12-11-4-9	0101 0100
85	55	CL			U		12-11-5-9	0101 0101
86	56	O			V		12-11-6-9	0101 0110
87	57	X			W		12-11-7-9	0101 0111
88	58	L			X		12-11-8-9	0101 1000
89	59	C			Y		11-1-8	0101 1001
90	5A	A		!	Z		11-2-8	0101 1010
91	5B	S	$	$	[	B 8 2 1	11-3-8	0101 1011
92	5C	M	*	*	\	B 8 4	11-4-8	0101 1100
93	5D	D	]	)	]	B 8 4 1	11-5-8	0101 1101
94	5E	AL	;	;	¬ ^	B 8 4 2	11-6-8	0101 1110
95	5F	SL	Δ	¬	-	B 8 4 2 1	11-7-8	0101 1111
96	60	STD	-	-	`	B	11	0110 0000
97	61		/	/	a	A 1	0-1	0110 0001
98	62				b		11-0-2-9	0110 0010

CODE TRANSLATION TABLE (....Cont'd)

Dec	Hex	Instruction (RX)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
99	63				c		11-0-3-9	0110 0011
100	64				d		11-0-4-9	0110 0100
101	65				e		11-0-5-9	0110 0101
102	66				f		11-0-6-9	0110 0110
103	67	MXD			g		11-0-7-9	0110 0111
104	68	LD			h		11-0-8-9	0110 1000
105	69	CD			i		0-1-8	0110 1001
106	6A	AD		¦	j		12-11	0110 1010
107	6B	SD	.	.	k	A 8 2 1	0-3-8	0110 1011
108	6C	MD	%	%	l	A 8 4	0-4-8	0110 1100
109	6D	DD	Ÿ	-	m	A 8 4 1	0-5-8	0110 1101
110	6E	AW	\	>	n	A 8 4 2	0-6-8	0110 1110
111	6F	SW	≠	?	o	A 8 4 2 1	0-7-8	0110 1111
112	70	STE			p		12-11-0	0111 0000
113	71				q		12-11-0-1-9	0111 0001
114	72				r		12-11-0-2-9	0111 0010
115	73				s		12-11-0-3-9	0111 0011
116	74				t		12-11-0-4-9	0111 0100
117	75				u		12-11-0-5-9	0111 0101
118	76				v		12-11-0-6-9	0111 0110
119	77				w		12-11-0-7-9	0111 0111
120	78	LE			x		12-11-0-8-9	0111 1000
121	79	CE			y		1-8	0111 1001
122	7A	AE		:	z	A	2-8	0111 1010
123	7B	SE	≠ =	≠	{	8 2 1	3-8	0111 1011
124	7C	ME	@'	@	l	8 4	4-8	0111 1100
125	7D	DE	:	'	}	8 4 1	5-8	0111 1101
126	7E	AU	>	=	~	8 4 2	6-8	0111 1110
127	7F	SU	v	"	DEL	8 4 2 1	7-8	0111 1111

1 EBCDIC graphics shown are standard bit pattern assignments. For specific print train/chain: See printer manual.

2 Add C (check bit) for odd or even parity as needed, except as noted.

Dec	Hex	Instruction (RS, SI, S)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
128	80	SSM					12-0-1-8	1000 0000
129	81			a			12-0-1	1000 0001
130	82	LPSW		b			12-0-2	1000 0010
131	83	Diagnose		c			12-0-3	1000 0011
132	84	WRD		d			12-0-4	1000 0100
133	85	RDD		e			12-0-5	1000 0101
134	86	BXH		f			12-0-6	1000 0110
135	87	BXLE		g			12-0-7	1000 0111
136	88	SRL		h			12-0-8	1000 1000
137	89	SLL		i			12-0-9	1000 1001
138	8A	SRA					12-0-2-8	1000 1010
139	8B	SLA					12-0-3-8	1000 1011
140	8C	SRDL					12-0-4-8	1000 1100
141	8D	SLDL					12-0-5-8	1000 1101
142	8E	SRDA					12-0-6-8	1000 1110
143	8F	SLDA					12-0-7-8	1000 1111
144	90	STM					12-11-1-8	1001 0000

CODE TRANSLATION TABLE (....Cont'd)

Dec	Hex	Instruction (RS, SI, S)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
145	91	TM		j			12-11-1	1001 0001
146	92	MVI		k			12-11-2	1001 0010
147	93	TS		l			12-11-3	1001 0011
148	94	NI		m			12-11-4	1001 0100
149	95	CLI		n			12-11-5	1001 0101
150	96	OI		o			12-11-6	1001 0110
151	97	XI		p			12-11-7	1001 0111
152	98	LM		q			12-11-8	1001 1000
153	99			r			12-11-9	1001 1001
154	9A						12-11-2-8	1001 1010
155	9B						12-11-3-8	1001 1011
156	9C	SIO, SIOF					12-11-4-8	1001 1100
157	9D	TIO					12-11-5-8	1001 1101
158	9E	HIO, HDV					12-11-6-8	1001 1110
159	9F	TCH					12-11-7-8	1001 1111
160	A0						11-0-1-8	1010 0000
161	A1			~			11-0-1	1010 0001
162	A2			s			11-0-2	1010 0010
163	A3			t			11-0-3	1010 0011
164	A4			u			11-0-4	1010 0100
165	A5			v			11-0-5	1010 0101
166	A6			w			11-0-6	1010 0110
167	A7			x			11-0-7	1010 0111
168	A8			y			11-0-8	1010 1000
169	A9			z			11-0-9	1010 1001
170	AA						11-0-2-8	1010 1010
171	AB						11-0-3-8	1010 1011
172	AC	STNSM					11-0-4-8	1010 1100
173	AD	STOSM					11-0-5-8	1010 1101
174	AE	SIGP					11-0-6-8	1010 1110
175	AF	MC					11-0-7-8	1010 1111
176	B0						12-11-0-1-8	1011 0000
177	B1	LRA					12-11-0-1	1011 0001
178	B2	See below					12-11-0-2	1011 0010
179	B3						12-11-0-3	1011 0011
180	B4						12-11-0-4	1011 0100
181	B5						12-11-0-5	1011 0101
182	B6	STCTL					12-11-0-6	1011 0110
183	B7	LCTL					12-11-0-7	1011 0111
184	B8						12-11-0-8	1011 1000
185	B9						12-11-0-9	1011 1001
186	BA	CS					12-11-0-2-8	1011 1010
187	BB	CDS					12-11-0-3-8	1011 1011
188	BC						12-11-0-4-8	1011 1100
189	BD	CLM					12-11-0-5-8	1011 1101
190	BE	STCM					12-11-0-6-8	1011 1110
191	BF	ICM					12-11-0-7-8	1011 1111

Op Code:

B202 - STIDP
B203 - STIDC
B204 - SCK
B205 - STCK
B206 - SCKC
B207 - STCKC
B208 - SPT
B209 - STPT
B20D - PTLB
B213 - RRB

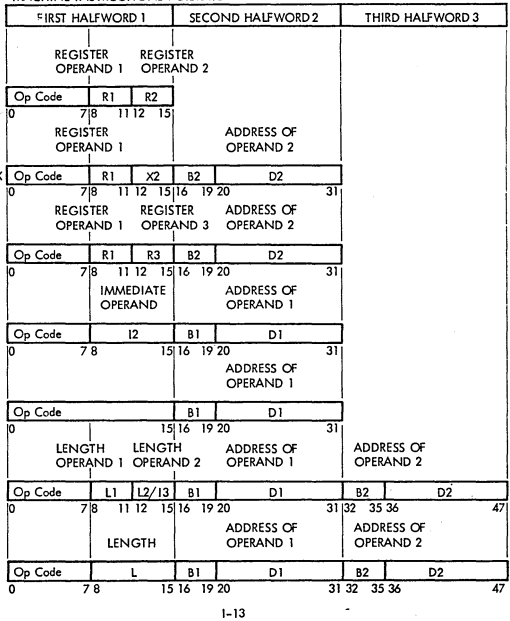
CODE TRANSLATION TABLE (....Cont'd)

Dec	Hex	Instruction (SS)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
192	C0		?	{		B A 8 2	12-0	1100 0000
193	C1		A	A		B A 1	12-1	1100 0001
194	C2		B	B		B A 2	12-2	1100 0010
195	C3		C	C		B A 2 1	12-3	1100 0011
196	C4		D	D		B A 4	12-4	1100 0100
197	C5		E	E		B A 4 1	12-5	1100 0101
198	C6		F	F		B A 4 2	12-6	1100 0110
199	C7		G	G		B A 4 2 1	12-7	1100 0111
200	C8		H	H		B A 8	12-8	1100 1000
201	C9		I	I		B A 8 1	12-9	1100 1001
202	CA						12-0-2-8-9	1100 1010
203	CB						12-0-3-8-9	1100 1011
204	CC			⌡			12-0-4-8-9	1100 1100
205	CD						12-0-5-8-9	1100 1101
206	CE			┤			12-0-6-8-9	1100 1110
207	CF						12-0-7-8-9	1100 1111
208	D0		!	}		B 8 2	11-0	1101 0000
209	D1	MVN	J	J		B 1	11-1	1101 0001
210	D2	MVC	K	K		B 2	11-2	1101 0010
211	D3	MVZ	L	L		B 2 1	11-3	1101 0011
212	D4	NC	M	M		B 4	11-4	1101 0100
213	D5	CLC	N	N		B 4 1	11-5	1101 0101
214	D6	OC	O	O		B 4 2	11-6	1101 0110
215	D7	XC	P	P		B 4 2 1	11-7	1101 0111
216	D8		Q	Q		B 8	11-8	1101 1000
217	D9		R	R		B 8 1	11-9	1101 1001
218	DA						12-11-2-8-9	1101 1010
219	DB						12-11-3-8-9	1101 1011
220	DC	TR					12-11-4-8-9	1101 1100
221	DD	TRT					12-11-5-8-9	1101 1101
222	DE	ED					12-11-6-8-9	1101 1110
223	DF	EDMK					12-11-7-8-9	1101 1111
224	E0		‡	\		A 8 2	0-2-8	1110 0000
225	E1						11-0-1-9	1110 0001
226	E2		S	S		A 2	0-2	1110 0010
227	E3		T	T		A 2 1	0-3	1110 0011
228	E4		U	U		A 4	0-4	1110 0100
229	E5		V	V		A 4 1	0-5	1110 0101
230	E6		W	W		A 4 2	0-6	1110 0110
231	E7		X	X		A 4 2 1	0-7	1110 0111
232	E8		Y	Y		A 8	0-8	1110 1000
233	E9		Z	Z		A 8 1	0-9	1110 1001
234	EA						11-0-2-8-9	1110 1010
235	EB						11-0-3-8-9	1110 1011
236	EC			┼			11-0-4-8-9	1110 1100
237	ED						11-0-5-8-9	1110 1101
238	EE						11-0-6-8-9	1110 1110
239	EF						11-0-7-8-9	1110 1111
240	F0	SRP	0	0		8 2	0	1111 0000
241	F1	MVO	1	1		1	1	1111 0001
242	F2	PACK	2	2		2	2	1111 0010
243	F3	UNPK	3	3		2 1	3	1111 0011
244	F4		4	4		4	4	1111 0100
245	F5		5	5		4 1	5	1111 0101

CODE TRANSLATION TABLE (....Cont'd)

Dec	Hex	Instruction (SS)	Graphics and Controls BCDIC	EBCDIC (1)	ASCII	7-Track Tape BCDIC (2)	Card Code	Binary
246	F6		6	6		4 2	6	1111 0110
247	F7		7	7		4 2 1	7	1111 0111
248	F8	ZAP	8	8		8	8	1111 1000
249	F9	CP	9	9		8 1	9	1111 1001
250	FA	AP		1			12-11-0-2-8-9	1111 1010
251	FB	SP					12-11-0-3-8-9	1111 1011
252	FC	MP					12-11-0-4-8-9	1111 1100
253	FD	DP					12-11-0-5-8-9	1111 1101
254	FE						12-11-0-6-8-9	1111 1110
255	FF						12-11-0-7-8-9	1111 1111

MACHINE INSTRUCTIONS FORMATS



CONTROL REGISTER ALLOCATION

	0	1	2	3
0	SYSTEM CONTR	TRANSL CONTR	EXTERNAL INTERRUPTION MASKS	
1	SEGM TBL LENGTH	SEGMENT TABLE ORIGIN ADDRESS		
2	CHANNEL MASKS			
3				
4				
5				
6				
7				
8			MONITOR MASKS	
9	PER EVENT MASKS		PER GR ALTERATION MASKS	
10		PER STARTING ADDRESS		
11		PER ENDING ADDRESS		
12				
13				
14	ERROR RECOVERY CONTR & MASKS			
15		MCEL ADDRESS		

ASSIGNMENT OF CONTROL REGISTER FIELDS

Word	Bits	Name of Field	Facility	Initial value
0	0	Block-Multiplexing Mode	Block Multiplexing Control	0
0	1	SSM Suppression	Extended Control	0
0	8-9	Page Size	Dynamic Address Translation	0
0	10	Reserved	Dynamic Address Translation	0
0	11-12	Segment Size	Dynamic Address Translation	0
0	20	Clock Comparator Mask	Clock Comparator	0
0	21	CPU Timer Mask	CPU Timer	0
0	24	Interval Timer Mask	External Interruption	1
0	25	Interrupt Key Mask	External Interruption	1
0	26	External Signal Mask	External Interruption	1
1	0-7	Segment Table Length	Dynamic Address Translation	0
1	8-25	Segment Table Address	Dynamic Address Translation	0
2	0-31	Channel Masks	I/O Interruptions	1
8	16-31	Monitor Masks	Monitoring	0
9	0-7	PER* Event Masks	Program-Event Recording	0
9	16-31	PER GR Aletration Masks	Program-Event Recording	0
10	8-31	PER Staring Address	Program-Event Recording	0

ASSIGNMENT OF CONTROL REGISTER FIELDS (....Cont'd)

Word	Bits	Name of Field	Facility	Initial value
11	8-31	PER Ending Address	Program-Event Recording	0
14	0	Check-Stop Control	Machine-Check Handling	1
14	1	Synchronous MCEL** Control	Machine-Check Handling	1
14	2	I/O Extended Logout Control	Machine-Check Handling	0
14	4	Recovery Report Mask	Machine-Check Handling	0
14	5	Degradation Report Mask	Machine-Check Handling	0
14	6	External Damage Report Mask	Machine-Check Handling	1
14	7	Warning Mask	Machine-Check Handling	0
14	8	Asynchronous MCEL Control	Machine-Check Handling	0
14	9	Asynchronous Fixed Log Contr.	Machine-Check Handling	0
15	8-28	MCEL Address	Machine-Check Handling	512***

Explanation:

The fields not listed are unassigned.
The initial value of unassigned register positions is unpredictable.

* PER means Program-Event Recording
** MCEL means machine-check extended logout
*** Bit 22 is set to one, with all other bits set to zero, thus yielding a decimal byte address of 512

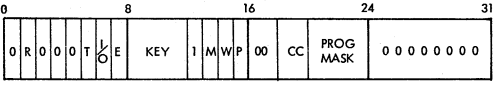
PERMANENT MAIN STORAGE ASSIGNMENT

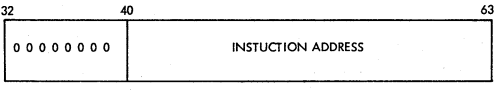
Storage Loc Hex	Storage Loc Dec	Byte	Byte	Byte	Byte
0	0	Restart New PSW (IPL PSW)			
4	4				
8	8	Restart Old PSW (IPL CCW1)			
C	12				
10	16	(IPL CCW2)			
14	20				
18	24	External Old PSW			
1C	28				
20	32	Supervisor Call Old PSW			
24	36				
28	40	Program Old PSW			
2C	44				
30	48	Machine Check Old PSW			
34	52				
38	56	Input/Output Old PSW			
3C	60				
40	64	Channel Status Word			
44	68				
48	72	Channel Address Word			
4C	76				
50	80	Interval Timer			
54	84				
58	88	External New PSW			
5C	92				
60	96	Supervisor Call New PSW			
64	100				
68	104	Program New PSW			
6C	108				
70	112	Machine Check New PSW			
74	116				
78	120	Input/Output New PSW			
7C	124				
80	128				
84	132	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0		External Interruption Code
88	136	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	ILC 0	SVC Interruption Code
8C	140	0 0 0 0 0 0 0 0	0 0 0 0 0 0 0 0	ILC 0	Program Interruption Code

PERMANENT MAIN STORAGE ASSIGNMENT (....Cont'd)

Storage Loc Hex	Dec	Byte	Byte	Byte	Byte
90	144	0 0 0 0 0 0 0 0	Translation Exception Address		
94	148	0 0 0 0 0 0 0 0	Monitor Class ≠	PER Code	0 0 0 0 0 0 0 0
98	152	0 0 0 0 0 0 0 0	PER Address		
9C	156	0 0 0 0 0 0 0 0	Monitor Code		
A0	160				
A4	164				
A8	168	Channel ID			
AC	172		IOEL Address		
B0	176	Limited Channel Logout			
B4	180				
B8	184		0 0 0 0 0 0 0 0	I/O Address	
BC ⋮ D4	188 ⋮ 212				
D8	216	Machine Check CPU-Timer Save Area			
DC	220				
E0	224	Machine Check Clock-Comparator Save Area			
E4	228				
E8	232	Machine Check Interruption Code			
EC	236				
F0	240				
F4	244				
F8	248	0 0 0 0 0 0 0 0	Failing Storage Address		
FC	252	Region Code			
100 ⋮ 15C	256 ⋮ 348	Machine Check Fixed Logout Area			
160 ⋮ 17C	352 ⋮ 380	Machine Check Floating Point Register Save Area			
180 ⋮ 1BC	384 ⋮ 444	Machine Check General Register Save Area			
1C0 ⋮ 1FC	448 ⋮ 508	Machine Check Control Register Save Area			

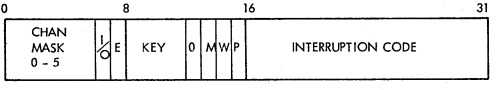
PROGRAM STATUS WORD (EC-mode)

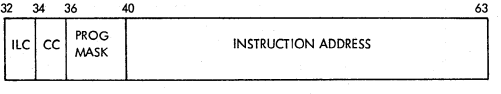




Bit	Description	Bit	Description
0	Always zero	15	Problem state
1	PER mask	16-17	Always zero
2-4	Always zero	18-19	Condition code
5	Translate mode	20	Fixed-point overflow mask
6	I/O interrupt mask	21	Decimal overflow mask
7	External interrupt mask	22	Exponent overflow mask
8-11	KEY	23	Significance mask
12	Always one (EC mode)	24-31	Always zero
13	Machine check mask	32-39	Always zero
14	Wait state	40-63	Instruction address

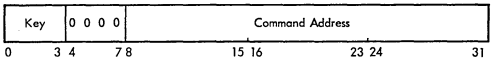
PROGRAM STATUS WORD (BC-mode)



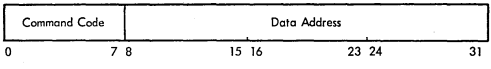


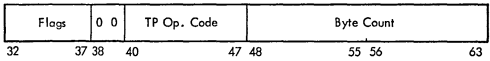
Bit	Description	Bit	Description
0	Channel 0 mask	14	Wait state
1	Channel 1 mask	15	Problem state
2	Channel 2 mask	16-31	Interruption code
3	Channel 3 mask	32-33	Instruction length code
4	Channel 4 mask	34-35	Condition code
5	Channel 5 mask	36	Fixed-point overflow mask
6	Mask for channel 6 and up	37	Decimal overflow mask
7	External interrupt mask	38	Exponent underflow mask
8-11	Protection key	39	Significance mask
12	Always 0 (BC mode)	40-63	Instruction address
13	Machine check mask		

CHANNEL ADDRESS WORD (CAW) (X'48')



CHANNEL COMMAND WORD (CCW)

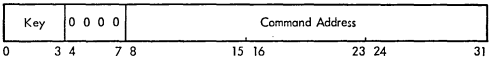


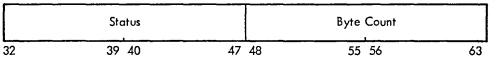


Flags

Bit	Description	
32	CD-bit (80)	: causes use of address portion of next CCW
33	CC-bit (40)	: causes use of command code and data address of next CCW
34	SLI-bit (20)	: causes suppression of possible incorrect length indication
35	Skip bit(10)	: supresses transfer of information to main storage
36	PCI-bit(08)	: causes a channel Program Controlled Interruption
37	IDA-bit(04)	: specifies indirect data addressing

CHANNEL STATUS WORD (CSW) (X'40')



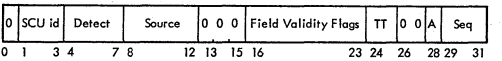


Status

Bit	Description		Bit	Description	
32	(8000)	Attention	40	(0080)	Program controlled interruption
33	(4000)	Status Modifier	41	(0040)	Incorrect length
34	(2000)	Control unit end	42	(0020)	Program check
35	(1000)	Busy	43	(0010)	Protection check
36	(0800)	Channel end	44	(0008)	Channel data check
37	(0400)	Device end	45	(0004)	Channel control check
38	(0200)	Unit check	46	(0002)	Interface control check
39	(0100)	Unit exception	47	(0001)	Chaining check

Byte count: Bits 48-63 form the residual count for the last CCW used.

LIMITED CHANNEL LOGOUT (X'B0')



Detect

Bit	Description
4	CPU
5	Channel
6	Storage control unit
7	Storage unit

Source

Bit	Description
8	CPU
9	Channel
10	Storage control unit
11	Storage unit
12	Control unit

Field Validity Flags

Bit	Description
16	Interface address
17	Reserved (0)
18	Reserved (0)
19	Sequence code
20	Unit status
21	Command address and key
22	Channel address
23	Device address

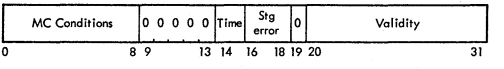
TT (Type of termination)

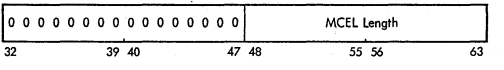
Bit 24-25	Description
0 0	Interface disconnect
0 1	Stop, stack or normal
1 0	Selective reset
1 1	System reset

A (bit 28) : I/O error alert

Seq (bits 29-31) : Sequence code

MACHINE CHECK INTERRUPTION CODE (X'E8')





MC Conditions

Bit	Description
0	System damage
1	Instr.proc'g damage
2	System recovery
3	Timer damage
4	TOD clock damage
5	External damage
6	-
7	Auto configuration
8	Warning

Time

Bit	Description
14	Backed-up
15	Delayed

Stg error

Bit	Description
16	Storage error uncorrected
17	Storage error corrected
18	Storage-error key uncorrected

MACHINE CHECK INTERRUPTION CODE (....Cont'd)

Validity

Bit	Description	Bit	Description
20	PSW bits 12-15	26	-
21	PSW masks and key	27	Floating-point registers
22	Program mask and cond. code	28	General registers
23	Instruction address	29	Control registers
24	Failing storage address	30	CPU extended logout
25	Region code	31	Storage logical

CODES FOR PROGRAM INTERRUPTION (X'8E')

Interruption Code		Program Interruption Cause	Interruption Code		Program Interruption Cause
Dec	Hex		Dec	Hex	
1	0001	Operation	12	000C	Exponent overflow
2	0002	Privileged operation	13	000D	Exponent underflow
3	0003	Execute	14	000E	Significance
4	0004	Protection	15	000F	Floating point divide
5	0005	Addressing	16	0010	Segment translation
6	0006	Specification	17	0011	Page translation
7	0007	Data	18	0012	Translation specification
8	0008	Fixed-point overflow	19	0013	Special operation
9	0009	Fixed-point divide	64	0040	Monitor event
10	000A	Decimal overflow	128	0080	Program event
11	000B	Decimal divide			

CODES FOR EXTERNAL INTERRUPTIONS (X'86')

Interruption Code (Hex)	External Interruption Cause
0080	Interval timer
0040	Interrupt key
0020	External signal 2
0010	External signal 3
0008	External signal 4
0004	External signal 5
0002	External signal 6
0001	External signal 7
1004	Clock comparator
1005	CPU timer

CODES FOR INPUT/OUTPUT INTERRUPTIONS (X'8A')

Interruption Code (Hex)	Input/Output Interruption Cause
00 dd	Channel 0
01 dd	1
02 dd	2
03 dd	3
04 dd	4
05 dd	5
06 dd	6

Note: d= device address

CODES FOR SUPERVISOR CALL INTERRUPTIONS (X'8A')

Interruption Code (Hex)	Supervisor Call Interruption Cause
00 rr	Instruction (0A)

Note: r= R1 and R2 field of SUPERVISOR CALL

CHANNEL COMMANDS

Standard Command Code Assignments (CCW bits 0-7) for I/O Operations

xxxx 0000	Invalid	†††† †† 01	Write
†††† 0100	Sense	†††† †† 10	Read
xxxx 1000	Transfer in Channel	†††† †† 11	Control
†††† 1100	Read backward	0000 00 11	Control No Operation

x-Bit ignored † Modifier bit for specific type of I/O device

3210, 3215 CONSOLES Source: GA24-3557

Write, No Carrier Return	01	Sense	04
Write, Auto Carrier Return	09	Audible Alarm	0B
Read Inquiry	0A		

3505 CARD READER/3525 CARD PUNCH Source: GA21-9124

Command	Binary	Hex
Sense	0000 0100	04
Feed, Select Stacker	SS10 F011	
Read Only *	11D0 F010	
Diagnostic Read	1101 0010	D2
Read, Feed, Select Stacker *	SSD0 F010	
Write RCE Format * †	0001 0001	11

Bit Meanings

SS	Stacker
00	1
01	2
10	2

Format Mode

0	Unformatted
1	Formatted

* Special feature on 3525 † Special feature on 3505

3505 CARD READER/3525 CARD PUNCH (....Cont'd)

3505 only		
Write OMR Format	0011 0001	31
3525 only		
Write, Feed, Select Stacker	SSD0 0001	
Print Line *	LLLL L101	

D	Data Mode
0	1-EBCDIC
1	2-Card image

L	Line Position
5 bit binary value	

* Special feature on 3525

3211 PRINTER/3811 CONTROL UNIT — Source: GA24-3543

	After Write	Immed.
Space 1 Line	09	0B
Space 2 Lines	11	13
Space 3 Lines	19	1B
Skip to Channel 0	-	83
Skip to Channel 1	89	8B
Skip to Channel 2	91	93
Skip to Channel 3	99	9B
Skip to Channel 4	A1	A3
Skip to Channel 5	A9	AB
Skip to Channel 6	B1	B3
Skip to Channel 7	B9	BB
Skip to Channel 8	C1	C3
Skip to Channel 9	C9	CB
Skip to Channel 10	D1	D3
Skip to Channel 11	D9	DB
Skip to Channel 12	E1	E3

Write without spacing	01
Sense	04
Load UCSB	FB
Fold	43
Unfold	23
Load FCB	63
Block Data Check	73
Allow Data Check	7B
Read PLB	02
Read UCSB	0A
Read FCB	12
Check Read	06
Diagnostic Write	05
Raise Cover	6B
Diagnostic Gate	07

3803/3420 MAGNETIC TAPE — Source: GA32-0020

Write	01
Read Forward	02
Read Backward	0C
Sense	04
Sense Reserve *	F4
Sense Release *	D4
Request Track in Error	1B
Loop Write to Read	8B
Set Diagnose	4B
Rewind	07
Rewind Unload	0F
Erase Gap	17
Write Tape Mark	1F
Backspace Block	27
Backspace File	2F
Forward Space Block	37
Forward Space File	3F

Data Security Erase	97
Diagnostic Write Mode Set	0B
Set Mode 1 (7-track) †	

Density	Parity	DC	Trans	Cmd
556	odd	on	off	53
		off	off	73
			on	7B
	even	off	off	63
			on	6B
800	odd	on	off	93
		off	off	B3
			on	BB
	even	off	off	A3
			on	AB

Set Mode 2 (9-track)

1600 bpi	C3
800 bpi †	CB

* Two-channel switch required † Special feature for NRZI operation

Source: GA26-1592 for 3830/3330
GA26-1589 for 2835/2305
GA26-3599, GA26-1606 for 2314, 2319

DIRECT ACCESS DEVICES

	Command	MT Off	MT On *	Count
Control	Orient (c)	2B		Nonzero
	Recalibrate	13		Nonzero
	Seek	07		6
	Seek cylinder	0B		6
	Seek Head	1B		6
	Space Count	0F		3 (a); nonzero (d)
	Set File Mask	1F		1
	Set Sector (a)	23		1
	Restore (a)	17		Nonzero
	Vary Sensing (c)	27		1
	Diagnostic Load (a)	53		1
	Diagnostic Write (a)	73		512
Search	Home Address Equal	39	B9	4
	Identifier Equal	31	B1	5
	Identifier High	51	D1	5
	Identifier Equal or High	71	F1	5
	Key Equal	29	A9	KL
	Key High	49	C9	KL
	Key Equal or High	69	E9	KL
	Key and Data Equal (d)	2D	AD	Number of bytes (incl. mask bytes) in search argument
	Key and Data High (d)	4D	CD	
	Key and Data Equal or High(d)	6D	ED	
Continue Scan	Search Equal (d)	25	A5	
	Search High (d)	45	C5	
	Search High or Equal (d)	65	E5	
	Set Status Modifier (d)	35	B5	
	Set Status Modifier (d)	75	F5	
	No Status Modifier (d)	55	D5	
Read	Home Address	1A	9A	5
	Count	12	92	8
	Record 0	16	96	Number of bytes to be transferred
	Data	06	86	
	Key and Data	0E	8E	
	Count, Key and Data	1E	9E	
	IPL	02		
	Sector (a)	22		1
Sense	Sense I/O	04		24 (a); 6 (d)
	Read, Reset Buffered Log (b)	A4		24
	Read Buffered Log (c)	24		128
	Device Release (e)	94		24 (a); 6 (d)
	Device Reserve (e)	B4		24 (a); 6 (d)
	Read Diagnostic Status 1 (a)	44		16 or 512
Write	Home Address	19		5
	Record 0	15		8+ KL+ DL or R0
	Erase	11		8+ KL+ DL
	Count, Key and Data	1D		8+ KL+ DL
	Special Count, Key and Data	01		8+ KL= DL
	Data	05		DL
	Key and Data	0D		KL+ DL

* Code same as MT Off except as listed
a 3830/3330 and 2835/2305 only
b 3830/3330 only c 2835/2305 only
d 2314, 2319 only
e Channel attachment and 2-channel switch feature required; standard on 2314 with 2844

HEXADECIMAL AND DECIMAL CONVERSION

From Hex: Locate each hex digit in its corresponding column position and note the decimal equivalents. Add these to obtain the decimal value.

From Dec: Locate the largest decimal value in the table that will fit into the decimal number to be converted.
Note its hex equivalent and hex column position.
Find the decimal remainder.
Repeat the process on this and subsequent remainders.

Hexadecimal Columns												
6		5		4		3		2		1		
Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	Hex	Dec	
0	0	0	0	0	0	0	0	0	0	0	0	
1	1 048 576	1	65 536	1	4 096	1	256	1	16	1	1	
2	2 097 152	2	131 072	2	8 192	2	512	2	32	2	2	
3	3 145 728	3	196 608	3	12 288	3	768	3	48	3	3	
4	4 194 304	4	262 144	4	16 384	4	1 024	4	64	4	4	
5	5 242 880	5	327 680	5	20 480	5	1 280	5	80	5	5	
6	6 291 456	6	393 216	6	24 576	6	1 536	6	96	6	6	
7	7 340 032	7	458 752	7	28 672	7	1 792	7	112	7	7	
8	8 388 608	8	524 288	8	32 768	8	2 048	8	128	8	8	
9	9 437 184	9	589 824	9	36 864	9	2 304	9	144	9	9	
A	10 485 760	A	655 360	A	40 960	A	2 560	A	160	A	10	
B	11 534 336	B	720 896	B	45 056	B	2 816	B	176	B	11	
C	12 582 912	C	786 432	C	49 152	C	3 072	C	192	C	12	
D	13 631 488	D	851 968	D	53 248	D	3 328	D	208	D	13	
E	14 680 064	E	917 504	E	57 344	E	3 584	E	224	E	14	
F	15 728 640	F	983 040	F	61 440	F	3 840	F	240	F	15	
0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7		0 1 2 3		4 5 6 7		
Byte				Byte				Byte				

POWERS OF 2

2^n	n
256	8
512	9
1 024	10
2 048	11
4 096	12
8 192	13
16 384	14
32 768	15
65 536	16
131 072	17
262 144	18
524 288	19
1 048 576	20
2 097 152	21
4 194 304	22
8 388 608	23
16 777 216	24

$2^n = 16^{n:4}$

POWERS OF 16

16^n	n
1	0
16	1
256	2
4 096	3
65 536	4
1 048 576	5
16 777 216	6
268 435 456	7
4 294 967 296	8
68 719 476 736	9
1 099 511 627 776	10
17 592 186 044 416	11
281 474 976 710 656	12
4 503 599 627 370 496	13
72 057 594 037 927 936	14
1 152 921 504 606 846 976	15

SENSE INFORMATION SUMMARY

1017 - Paper Tape Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention reject 2 : bus-out check 3 : - 4 : data check 5 : - 6 : - 7 : broken tape

1018 - Paper Tape Punch

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : - 6 : - 7 : -

1287 - Optical Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : non-recovery 7 : keyboard correction (tape only)
1	Bit 0 : tape mode 1 : late stacker select 2 : no document found 3 : - 4 : invalid operation 5 : - 6 : - 7 : -

SENSE INFORMATION SUMMARY (...Continued)

1288 - Optical Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : non-recovery 7 : -
1	Bit 0 : - 1 : end-of-page 2 : no document found 3 : - 4 : invalid operation 5 : - 6 : - 7 : -

1403 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : parity check (UCS storage) 6 : - 7 : channel 9
1	Not used
2	Bit 0 : chain interlock 1 : forms check 2 : coil protect check 3 : subscan ring check 4 : chain buffer address register check 5 : - 6 : any hammer on check 7 : -
3	Not used
4	Bit 0 : hammer reset failure check 1 : no fire check 2 : misfire check 3 : print data buffer parity check 4 : check bit buffer parity check 5 : chain buffer parity check 6 : buffer address register parity check 7 : clock check

1403 - Printer (...Continued)

Sense Byte	Designation
5	Bit 0 : open hammer coil check 1-7 : -

1419 - PCU - MICR

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : - 4 : data check 5 : overrun 6 : autoselect 7 : -
1	Bit 0 : - 1 : - 2 : document under read head 3 : amount field valid 4 : process-control field valid 5 : account-number field valid 6 : transit field valid 7 : serial-number field valid

1419 - SCU - MICR

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : - 4 : - 5 : late stacker select 6 : autoselect 7 : operator attention

1442 - Card Read-Punch / Card Punch

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : - 7 : -

SENSE INFORMATION SUMMARY (...Continued)

1443 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : type bar 5 : type bar 6 : - 7 : -

2260 - Display Station

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4-7 : -

2311 - Disk Storage

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : track condition check 7 : seek check
1	Bit 0 : data check in count area 1 : track overrun 2 : end of cylinder 3 : invalid sequence 4 : no record found 5 : file protect 6 : missing address marker 7 : overflow incomplete
2	Bit 0 : unsafe 1 : - 2 : serializer check 3 : - 4 : ALU check 5 : unselected file status 6 : - 7 : -

2311 - Disk Storage (...Continued)

Sense Byte	Designation
3	Bit 0 : ready 1 : on line 2 : unsafe 3 : - 4 : on line 5 : end of cylinder 6 : - 7 : seek incomplete
4	Bit 0-7: -
5	Bit 0-7: command in progress when overflow incomplete occurs

2314/2319 - Direct Access Storage

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : track condition check 7 : seek check
1	Bit 0 : data check in count area 1 : track overrun 2 : end of cylinder 3 : invalid sequence 4 : no record found 5 : file protect 6 : missing address marker 7 : overflow incomplete
2	Bit 0 : unsafe 1 : - 2 : SERDES check 3 : - 4 : ALU check 5 : unselected status 6 : - 7 : -

2314/2319 - Direct Access Storage

Sense Byte	Designation
3	Bit 0 : busy 1 : on line 2 : unsafe 3 : wr current sense 4 : pack change 5 : end of cylinder 6 : multi-module select 7 : seek incomplete
4	Bit 0 : wrong length record (2314 with multiplex storage control feature only) 1 : pending status (2314 with multiplex storage control feature only) 2 : - 3 : - 4-7 : Module indentification

bits 4567	physical drive
0000	A
0001	B
0010	C
0011	D
0100	E
0101	F
0110	G
0111	H
1000	J
1111	module not defined

5	Bit 0-7: command in progress when overflow incomplete occurs.

2321 - Data Cell

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : track condition check 7 : seek check

SENSE INFORMATION SUMMARY (...Continued)

2321 - Data Cell (...Continued)

Sense Byte	Designation
1	Bit 0 : data check in count area 1 : track overrun 2 : end of cylinder 3 : invalid sequence 4 : no record found 5 : file protect 6 : missing address marker 7 : overflow incomplete
2	Bit 0 : unsafe 1 : - 2 : serializer check 3 : - 4 : ALU check 5 : unselected file status 6 : - 7 : -
3	Bit 0 : drive ready 1 : drive operative 2 : read safety 3 : write safety 4 : strip ready 5 : invalid address 6 : auto restore 7 : CE cell located
4	Bit 0-7:-
5	Bit 0-7:command in progress when overflow incomplete occurs

2400 - Magnetic Tape

Sense Byte	Designation
0	Bit 0 :command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : word count zero 7 : data converter check

2400 - Magnetic Tape (...Continued)

Sense Byte	Designation
1	Bit 0 : noise 1-2 : B'00' = not existent B'01' = not ready B'10' = ready and not rewinding B'11' = ready and rewinding 3 : seven-track 4 : at loadpoint 5 : selected and write status 6 : file protect 7 : not capable
2	Bit 0-7: contains the track-in-error indicator bits that are set at the end of a read, or read-backward command if a data check has been encountered. Bits 6 and 7 on together indicate either more than one error or no error found.
3	Bit 0 : R/W VRC 1 : LRCR 2 : skew 3 : CRC 4 : skew register VRC 5 : phase encoding 6 : backward 7 : C compare
4	Bit 0 : echo check 1 : reject TU 2 : read clock error 3 : write clock error 4 : delay counter 5 : sequence indicator C 6 : sequence indicator B 7 : sequence indicator A

2495 - Tape Cartridge Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : should-not-occur 6 : position check 7 : should-not-occur

2501 - Card Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : - 7 : -

2520 - Card Read-Punch/Card Punch

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : - 7 : -

2540 - Card Reader/Card Punch

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : - 6 : unusual command 7 : -

2560 - Multifunction Card Machine

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4 : data check 5 : feed/machine check 6 : no card available 7 : print operation in progress

2560 - Multifunction Card Machine (...Continued)

Sense Byte	Designation
1	Bit 0 : cover interlock/punch pusher check 1 : jam bar check 2 : corner station check 3 : call 8 to 9 feed check 4 : print station feed check 5 : punch station feed check 6 : read station feed check 7 : input station feed check
2	Location of individual card : Bit 0 : secondary select 1 : card in punch station 2 : preprint SC7 exposed 3 : prepunch SC5 exposed 4 : prepunch SC4 exposed 5 : preread SC 3 exposed 6 : preread SC2 exposed 7 : input station SC1 exposed
3	Stacker Select Information : Bit 0 : primary card 0) 1 : binary value 4) card at primary 2 : binary value 2) prepunch station 3 : binary value 1) 4 : secondary card 1) 5 : binary value 4) card at secondary 6 : binary value 2) prepunch station 7 : binary value 1)
4	Stacker Select Information : Bit 0 : primary (0)) secondary (1)) card at punch or 1 : binary value 4) preprint station 2 : binary value 2) 3 : binary value 1) 4 : primary (0)) secondary (1)) card after print 5 : binary value 4) station 6 : binary value 2) 7 : binary value 1)

2560 - Multifunction Card Machine (...Continued)

Sense Byte	Designation	
5	Stacker Select Information :	
	Bit 0 : primary (0))	
	secondary (1))	
	1 : 4)	card at corner station
	2 : 2)	
	3 : 1)	
	4 : primary (0))	
	secondary (1))	card in stacker pocket
	5 : 4)	(was just stacked)
	6 : 2)	
	7 : 1)	
6	Card column in which first (possibly only) error was detected :	
	Bit 0 : multi data check	
	1 : binary value 64	
	2 : binary value 32	
	3 : binary value 16	
	4 : binary value 8	
	5 : binary value 4	
	6 : binary value 2	
	7 : binary value 1	

2596 - Card Read-Punch

Sense Byte	Designation
0	Bit 0 : command reject
	1 : intervention required
	2 : bus-out check
	3 : equipment check
	4 : data check
	5 : overrun
	6 : -
	7 : -

2671 - Paper Tape Reader

Sense Byte	Designation
0	Bit 0 : command reject
	1 : intervention required
	2 : bus-out check
	3 : equipment check
	4 : data check
	5 : -
	6 : -
	7 : -

3203 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4 : data check 5 : chain buffer parity check 6 : no channel found 7 : channel 9
1	Bit 0-7: -
2	Bit 0 : interlock (chain gate open) 1 : form check (jam) 2 : coil protect check 3 : subscan ring check 4 : chain buffer address register check 5 : hammer unit shift check (model 1 only) 6 : any hammer on check 7 : device ready check
3	Bit 0 : - 1 : - 2 : - 3 : carriage inhibit check 4 : - 5 : - 6 : step check 7 : move check
4	Bit 0 : hammer reset failure check 1 : no fire check 2 : misfire check 3 : print data buffer parity check 4 : check bit buffer parity check 5 : chain buffer parity check 6 : buffer address register check 7 : clock check
5	Bit 0 : open coil check 1-7 : -

3203-4 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4 : data check 5 : buffer parity check 6 : load check (load FCB/UCSB) 7 : channel 9
1	Bit 0 : command retry 1 : print check 2 : - 3 : line position check 4 : forms check 5 : command suppress 6 : controller check 7 : -
2	Bit 0 : carriage failed to move 1 : carriage motion check 2 : - 3 : - 4 : - 5 : forms jam 6 : - 7 : train velocity
3	Bit 0 : USCB parity 1 : PLB parity 2 : - 3 : coil protect 4 : hammer fire check 5 : - 6 : sync check 7 : -
4	Bit 0-7: always X'84', Device ID 3203-4 Printer Subsystem Bytes 5-23 contain Att. and device specific information. (pls. refer to MLM)

3210/3215 - Console Printer Keyboard

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4-7 : -

SENSE INFORMATION SUMMARY (...Continued)

3211 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : buffer parity check 6 : load check 7 : channel 9
1	Bit 0 : command parity 1 : print check 2 : print quality 3 : line position check 4 : forms check 5 : command suppress 6 : mechanical motion 7 : -
2	Bit 0 : carriage failed to move 1 : carriage sequence check 2 : carriage stop 3 : platen failed to advance 4 : platen failed to retract 5 : forms jam 6 : ribbon motion 7 : train overload
3	Bit 0 : UCSB parity 1 : PLB parity 2 : FCB parity 3 : coil protect 4 : hammer fire check 5 : service aid 6 : UCSAR sync check 7 : PSE sync check
4	Bit 0-7 : information used by service personnel
5	Bit 0-7 : -

3272 - (3270 Local)

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus out check 3 : equipment check 4 : data check 5 : unit specify 6 : control check 7 : operation check

3330 - Disk Storage

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : - 7 : -

3330 - Disk Storage (...Continued)

Sense Byte	Designation
1	Bit 0 : permanent error 1 : invalid track format 2 : end of cylinder 3 : - 4 : no record found 5 : file protected 6 : write inhibited 7 : operation incomplete
2	Bit 0 : - 1 : correctable 2 : - 3 : environmental data present 4 : - 5 : - 6 : - 7 : -
3	Bit 0-7:restart command
4	Bit 0-1:storage control identification 2-7:physical drive identification bits 2 to 7 / physical drive: 111000 A 110001 B 101010 C 100011 D 011100 E 010101 F 001110 G 000111 H
5	Bit 0-7:identify the eight low-order bits of the cylinder address in the most recent seek argument
6	Bit 0 : reverse 1 : cylinder number (high order bit of cylinder address) 2 : difference 3 : 16) 4 : 8) 5 : 4) head number 6 : 2) 7 : 1)
7	Bit 0-3:format type of remaining sense bytes (8-23) 4-7 : encoded error message
8-23	Meaning depends on format type

3340 - Disk Storage

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : track condition check 7 : seek check
1	Bit 0 : permanent error 1 : invalid track format 2 : end of cylinder 3 : - 4 : no record found 5 : file protected 6 : write inhibited 7 : operation incomplete
2	Bit 0 : RPS feature present 1 : correctable 2 : - 3 : environmental data present 4 : - 5 : - 6 : data module size) 01 = 35MB 7 : data module size) 10 = 70MB
3	Bit 0-7: restart command
4	Physical drive identification : Bit 0 : drive A 1 : drive B 2 : drive C 3 : drive D 4 : drive E 5 : drive F 6 : drive G 7 : drive H
5	Bit 0-7: Identifies the eight low-order bits of the cylinder address in the most recent seek argument
6	Bit 0-2: identifies the three high-order bits of the cylinder address 3 : - 4 : 8) 5 : 4) head number 6 : 2) 7 : 1)

3340 - Disk Storage (...Continued)

Sense Byte	Designation
7	Bit 0-3: format type of remaining sense bytes (8-23) 4-7: encoded error message
8-23	Meaning depends on format type

3344 - DIRECT ACCESS STORAGE (3340 Mode)

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : track condition check 7 : seek check
1	Bit 0 : permanent error 1 : invalid track format 2 : end of cylinder 3 : - 4 : no record found 5 : file protected 6 : write inhibited 7 : operation incomplete
2	Bit 0 : RPS feature present 1 : correctable 2 : - 3 : environmental data present 4 : Compatibility Mode / 3344 5 : HDA Size 3 Bit 6 : HDA Size 2 Bit 7 : HDA Size 1 Bit
3	Bit 0-7: restart command
4	Controller device address Bit 0 : Controller Addr. Bit 2 1 : Controller Addr. Bit 1 2 : 3 : } Controller Device Address 4 : 5 : Device Addr. Bit 4 6 : Device Addr. Bit 2 7 : Device Addr. Bit 1
5	Bit 0-7: Identifies the eight low-order bits of the cylinder address

3344 - DIRECT ACCESS STORAGE (3340 Mode) (...Continued)

Sense Byte	Designation
6	Bit 0-3: identifies the four high-order bits of the cylinder address 0 : 1024 1 : 512 2 : 256 3 : 2048 4 : 8) 5 : 4) head number 6 : 2) 7 : 1)
7	Bit 0-3: format type of remaining sense bytes (8-23) 4-7: encoded error message
8-23	Meaning depends on format type

3350 - Direct Access Storage

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : channel bus out parity 3 : equipment check 4 : overrun 5 : not used 7 : not used
1	Bit 0 : permanent error 1 : invalid track format 2 : end of cylinder 3 : not used 4 : no record found 5 : file protected 6 : write inhibited 7 : operation incomplete
2	Bit 0 : not used 1 : correctable 2 : not used 3 : environmental data present 4 : computibility mode 5 : not used 6 : not used 7 : not used
3	Bit 0 - 7 : restart command

3350 - Direct Access Storage (...Continued)

Sense Byte	Designation
4	Bit 0 - 7: physical drive identification Bit :physical drive 0 : A 1 : B 2 : C 3 : D 4 : E 5 : F 6 : G 7 : H
5	Bit 0 - 7 : low order logical cylinder address Bit :Value 0 : 128 1 : 64 2 : 32 3 : 16 4 : 8 5 : 4 6 : 2 7 : 1
6	Bit 0 - 2 : identifies the three high-order bits of the cylinder address Bit 0: CE Cylinder 1: 3330 - 11 = 512 3330 - 1 = 256 2: 3330 - 11 = 256 3330 - 1 = 0 3: 16) 4: 8) 5: 4) Head number 6: 2) 7: 1)
7	Bit 0 - 3 : FORMAT TYPE of remaining sense bytes (8 - 23)
8-23	Meaning depends on format type (see 3350 MLM)

3410/3411 - Magnetic Tape

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : word count zero 7 : data converter check
1	Bit 0 : noise 1-2 : B'00' = non-existent B'01' = not ready B'10' = ready and not busy B'11' = ready and busy 3 : seven track 4 : at load point 5 : write status 6 : file protected 7 : not capable
2	Bit 0-7: track in error bits
3	Bit 0 : VRC 1 : multiple track error (PE) or LRC (NRZI) 2 : skew 3 : end data check (PE) or CRC (NRZI) 4 : envelope check (PE only) 5 : phase encoding 6 : backword 7 : -
4	Bit 0 : tape unit positioning check 1 : tape unit reject 2 : end of tape 3 : - 4 : - 5 : diagnostic track check 6 : tape unit check 7 : illegal command

3410/3411 - Magnetic Tape (...Continued)

Sense Byte	Designation
5	Bit 0-1: new subsystem 2 : write tape mark check 3 : PE identification burst 4 : PE compare 5 : tachometer check 6 : false end mark 7 : RPO
6	Bit 0 : seven track 1 : short gap mode 2 : dual density 4-7 : tape unit model
7	Bit 0 : lamp check 1 : left column check 2 : right column check 3 : ready reset 4 : data security erase 5-7 : -
8	Bit 0 : - 1 : feedthrough 2 : - 3 : end velocity check 4 : no read-back data 5 : start velocity check 6 : - 7 : -

3420/3803 - Magnetic Tape

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : overrun 6 : word count zero 7 : data converter check

3420/3803 - Magnetic Tape (Continued)

Sense Byte	Designation
1	Bit 0 : noise 1-2 : B'00' = non-existent B'01' = not ready B'10' = ready and not rewinding B'11' = ready and rewinding 3 : seven track 4 : at load point 5 : write status 6 : file protected 7 : not capable
2	Bit 0-7: track in error bits
3	Bit 0 : VRC 1 : multiple track error (PE) or LPC (NRZI) 2 : skew 3 : end data check (PE) or CRC (NRZI) 4 : envelope check (PE only) 5 : phase encoding 6 : backward 7 : C-compare
4	Bit 0 : ALU hardware error 1 : reject tape unit 2 : tape indicate 3 : write trigger VRC 4 : microprogram detected error 5 : LWR 6 : tape unit check 7 : RPQ
5	Bit 0 : new subsystem 1 : new subsystem 2 : write tape mark check 3 : PE ID burst check 4 : start read check 5 : partial record 6 : excessive postable or tape mark 7 : RPQ
6	Bit 0 : seven track 1 : write current failure 2 : dual density 3 : NRZI density 4-7 : tape unit model
7	Bit 0 : lamp failure 1 : tape bottom left 2 : tape bottom right 3 : reset key 4 : data security erase 5 : erase head 6 : air bearing pressure 7 : load failure

3420/3803 - Magnetic Tape (...Continued)

Sense Byte	Designation
8	Bit 0 : IBG drop while writing 1 : feed through check 2 : SDR counter 3 : early begin readback check 4 : early ending readback check 5 : slow begin readback check 6 : slow ending readback check 7 : velocity retry/restart
9	Bit 0 : SDR counter 1 : velocity change during write 2-3 : SDR counter 4 : - 5 : - 6 : - 7 : tape control reserved
10	Bit 0 : command status reject 1 : - 2 : control status reject 3 : no block on record readback check 4 : WTM not detected block 5 : tachometer start fail 6 : - 7 : velocity check
11	Bit 0 : B bus parity error, ALU 1 1 : - 2 : low ROS parity/low IC/ parity on branch instr. 3 : high IC/high ROS reg parity 4 : micro program detected hardware error 5 : D bus parity error, ALU1 6 : - 7 : branch condition error, ALU2
12	Bit 0 : B bus parity error, ALU 2 1 : - 2 : low ROS parity/low IC/parity on branch instr. 3 : high IC/BC/high ROS reg parity 4 : microprogram detected hardware error 5 : D bus parity error, ALU 2 6 : - 7 : branch condition error, ALU 2
13	Bit 0-1:tape control density 2-7:tape control unique ID high
14	Bit 0-7:tape control unique ID low

3420/3803 - Magnetic Tape (...Continued)

Sense Byte	Designation
15	Bit 0-7:tape unit unique ID
16	Bit 0-7:tape unit unique ID
17	Bit 0 : two-channel switch 1-3 : tape control device switch features 4-7 : EC level of tape control
18	Bit 0 : Power check/air flow 1-3 : - 4-7 : EC level of tape unit
19	Bit 0 : primed for device and tape unit 7 1 : primed for device and tape unit 6 2 : primed for device and tape unit 5 3 : primed for device and tape unit 4 4 : primed for device and tape unit 3 5 : primed for device and tape unit 2 6 : primed for device and tape unit 1 7 : primed for device and tape unit 0
20	Bit 0 : primed for device and tape unit F 1 : primed for device and tape unit E 2 : primed for device and tape unit D 3 : primed for device and tape unit C 4 : primed for device and tape unit B 5 : primed for device and tape unit A 6 : primed for device and tape unit 9 7 : primed for device and tape unit 8
21	Bit 0 : load button depressed 1 : left reel turning 2 : right reel turning 3 : tape present 4 : reels loaded 5 : load rewind 6 : load complete 7 : load check
22	Bits 0-7:FRU identifiers for tape control
23	Bits 0-7:FRU identifiers for tape control

SENSE INFORMATION SUMMARY (...Continued)

3504/3505/3525 - Card I/O

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : - 6 : abnormal format reset 7 : permanent error key
1	Bit 0 : permanent error 1 : automatic retry 2 : motion malfunction 3 : retry after intervention complete 4-7 : -
2 - 3	Used for diagnostic purposes only.

3540 - Diskette

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5-7 : -
1	Bit 0 : permanent error 1 : automatic retry 2 : motion malfunction 3 : retry after intervention complete 4 : special record transferred 5-7 : -
2	Used for diagnostic purposes only
3	Bit 0-7:cylinder address in binary
4	Bit 0-7:head address, must be binary zero
5	Bit 0-7:record address in binary

3881 - Optical Mark Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : - 5 : - 6 : unusual command sequence 7 : -

3886 - Optical Character Reader

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : - 5 : - 6 : non-initialized 7 : RCP error
1	Bit 0 : - 1 : mark check 2 : invalid format 3 : - 4 : incomplete scan 5 : - 6 : non-recovery 7 : outboard

5203 - Printer

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4 : data check 5 : chain buffer parity check 6 : no channel found 7 : channel 9

SENSE INFORMATION SUMMARY (...Continued)

5203 - Printer (...Continued)

Sense Byte	Designation
1	Bit 0-7:-
2	Bit 0 : interlock (chain gate open) 1 : forms check (jam) 2 : coil protect check 3 : subscan ring check 4 : chain buffer address register check 5 : hammer unit shift check 6 : any-hammer-on check 7 : thermal overload
3	Bit 0-7:-
4	Bit 0 : hammer reset failure check 1 : no fire check 2 : misfire check 3 : print data buffer parity check 4 : check bit buffer parity check 5 : chain buffer parity check 6 : buffer address register check 7 : clock check
5	Bit 0 : open coil check 1-7 : -

5425 - Multifunction Card Unit

Sense Byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : bus-out check 3 : equipment check 4 : data check 5 : - 6 : no card available 7 : -
1	Bit 0 : read check 1 : punch check 2 : - 3 : print data check 4 : print clutch check 5 : hopper check 6 : feed check 7 : -

5425 - Multifunction Card Unit (...Continued)

Sense Byte	Designation
2	Bit 0 : - 1 : - 2 : card in primary wait station 3 : card in secondary wait station 4 : NPRO allowed 5 : hopper cycle not complete 6 : card in transport counter bit 2 7 : card in transport counter bit 1
3	Contains a hexadecimal number whose value can represent feedchecks and emitter checks in the 5425
4	Defines the card column group and tier where the error was detected which caused the first read check or punch check of a card cycle : Bit 0 : multiple error 1-2 : B'00' Tier 1 B'01' Tier 2 B'10' Tier 3 3-7 : B'00000' column group 1 B'00001' column group 2 ⋮ B'11111' column group 32
5	Bit 0 : D row miscompare 1 : C row miscompare 2 : B row miscompare 3 : A row miscompare 4 : 8 row miscompare 5 : 4 row miscompare 6 : 2 row miscompare 7 : 1 row miscompare
6 - 10	Forms a table of the five most recent command strings Bit 0 : Secondary 1 : print four lines 2 : stacker select M2 3 : stacker select M3 4 : punch 5 : feed command sample 6 : print 7 : read

SENSE INFORMATION SUMMARY (...Continued)

DOC - Display Operator Console

Sense byte	Designation
0	Bit 0 : command reject 1 : intervention required 2 : - 3 : equipment check 4 : - 5 : - 6 : - 7 : operation check

CHAPTER II
DOS/VS GENERAL INFORMATION

Operation	Operand	Remarks
ADD	X'cuu'[(k)],devicetype[,X'ss' ,X'ssss' ,X'ssssss']	Add a device to the PUB table. X'cuu': Channel and unit number (k): Can be specified as either (S) or a decimal number from 0 to 255. (S) indicates that the device can be switched (that is, physically attached to two adjacent channels). The designated channel is the lower of the two channels. (0)-(255) indicates the priority of a device that cannot be switched, with 0 indicating the highest priority. If (k) is not given, the assumed priority is 255 device type: actual device (See device codes list) X'ss' X'ssss' X'ssssss': device specification (See ASSGN statement). If absent the following values are assigned: X'C0' for 9-track tapes X'90' for 7-track tapes X'00' for nontapes. X'00', X'01', X'02' and X'03' are invalid as X'ss' for magnetic tape. X'ss' specifies SADxxx (Set ADdress) requirements for IBM 2702 lines: X'00' for SAD0 X'01' for SAD1 X'02' for SAD2 X'03' for SAD3 X'ss' is required for MICR/OCR device types. It specifies the external interrupt bit in the old PSW, which is used by this device to indicate "read complete". The specifications are: X'01' PSW bit 31 X'02' PSW bit 30 X'04' PSW bit 29 X'08' PSW bit 28 X'10' PSW bit 27 X'20' PSW bit 26

Operation	Operand	Remarks
ADD (Cont'd)		The X'ss' parameter specifies whether or not the error correction feature is present on an IBM 1018 Paper Tape Punch with 2826 Control Unit. These specifications are: X'00' No error correction feature X'01' Error correction feature For the ICA of the M 115/125, X'ss' X'ssss' or X'ssssss' is used to specify the line mode setting for a Start/Stop line or a BSC line. This is not accepted on the ASSGN statement. If a one or two byte value is specified the specified value is right-justified and the rest of the three bytes is filled with zeros. Note: Optional statement; if required it must be entered before SET command
CAT	UNIT= X'cuu'	Assigns the system logical unit SYSCAT X'cuu': Indicates the hexadecimal channel (c) and unit (uu) number of the device that is to contain the VSAM master catalog. Note: Optional statement; if required the CAT command must follow the SET command and precede the DPD com'd.
DEL	X'cuu'	Delete a device from the PUB table. X'cuu': Channel and unit number. Note: Optional statement; if required it must be entered before SET command
DPD	[TYPE= {N / F}][, UNIT= X'cuu', CYL= xxx][, VOLID= xxxxxx]	Defines the page data set. TYPE= N: Indicates that the page data set need not be formatted and the extent limits have not been changed. If TYPE= N is specified but the page data set does not exist or the extent limits have been changed, TYPE= N is ignored and the page data set is formatted during IPL. In this case, the UNIT and CYL operands must either have been supplied during system generation, or they must be specified in the DPD command.

Operation	Operand	Remarks
DPD (Cont'd)		TYPE= F indicates that the page data set is to be formatted during IPL. Formatting during IPL is required if the page data set is to be extended or if it is to be reallocated. UNIT= X'cuu' specifies the channel and unit number of the device that is to contain the page data set. If UNIT is specified, CYL must also be specified. CYL= xxx: Specifies the sequential number of the cylinder, relative to zero, where the page data set is to begin. (The size of the page data set extent is calculated by the system) If CYL is specified, UNIT must also be specified. VOLID= xxxxxx identifies the alphanumeric volume serial no of the disk pack that contains the page data set. If this operand is omitted both during system generation and in the DPD command, the volume serial number is not checked. Notes: Required statement. The DPD command must be the last command entered during IPL procedures. The operands of the DPD command may be given in any order.
SET	[DATE= value1 [, CLOCK=value2]] [, ZONE= {EAST / WEST} /hh/mm]	value1: In one of the following formats: mm/dd/yy or dd/mm/yy, mm: month (01-12) dd : day (01-31) yy : year (00-99) value2: In the following format: hh/mm/ss, hh : hours (00-23) mm: minutes(00-59) ss : seconds(00-59) EAST: Specifies a geographical position east of Greenwich.

Operation	Operand	Remarks
SET (Cont'd)		WEST: Specifies a geographical position west of Greenwich. hh/mm: A decimal value which indicates the difference in hours and minutes between local and Greenwich Mean Time. hh : 0-12 mm: 0-59 Note: Required statement. If any ADD or DEL commands are required, they must precede the SET command.

Name	Operation	Operand	Remarks	Accepted by
	ALLOC	F1=nK [,F2=nK] [,F3=nK] [,F4=nK]	Allocates foreground program areas in the virtual address area. Value of n is an even number. The order of operands is arbitrary. At least one operand must be specified.	JCC AR
	ALLOCR	[BGR=nK] [,F1R=nK] [,F2R=nK] [,F3R=nK] [,F4R=nK]	Allocates real address area among foreground and background programs. Value of n is an even number. The order of operands is arbitrary. At least one operand must be specified.	JCC
	ALTER	XXXXXX	Alters 1 to 16 bytes of virtual storage. XXXXXX is the hex address where alteration is to start.	AR
[//]	ASSGN	For any device: SYSxxx, { X'cuu' / UA / IGN / 'address-list' / SYSyyy } [,TEMP / ,PERM] For disks: SYSxxx, { X'cuu' / (address-list) / SYSyyy / DISK / 2311 / 2314 / 3330 / 3330B / 3340 / 3350 } [,TEMP / ,PERM] [, VOL=volserno][, SHR] For diskettes: SYSxxx, { X'cuu' / (address-list) / SYSyyy / DISKETTE / 3540 } [TEMP / PERM] For tapes: SYSxxx, { X'cuu' / (address-list) / SYSyyy / TAPE / 2400T7 / 2400T9 / 3410T7 / 3410T9 / 3420T7 / 3420T9 } [X'ss' / ALT] [,TEMP / ,PERM] [,VOL=volserno]	For remarks see end of this statement	JCS JCC

Name	Operation	Operand	Remarks	Accepted by
	ASSGN (Cont'd)	For printers: SYSxxx, { X'cuu' / (address-list) / SYSyyy / PRINTER / PRT 1 / 1403 / 1403U / 1443 / 3203 / 3211 / 5203 / 5203U } [,TEMP / ,PERM] See Note For card (read) punches: SYSxxx, { X'cuu' / (address-list) / SYSyyy / PUNCH / 1442N1 / 1442N2 / 2520B1 / 2520B2 / 2520B3 / 2540P / 2560 [,H1 / ,H2] / 2596 / 3525P / 3525RP / 5425 [,H1 / ,H2] } [,TEMP / ,PERM] For card readers: SYSxxx, { X'cuu' / (address-list) / SYSyyy / READER / 1442N1 / 2501 / 2520B1 / 2540R / 2560 [,H1 / ,H2] / 2596 / 3504 / 3505 / 3525RP / 5425 [,H1 / ,H2] } [,TEMP / ,PERM]	For remarks see end of this statement	

Name	Operation	Operand	Remarks	Accepted by
	ASSGN (Cont'd)	SYSxxx :	can be SYSRDR, SYSIPT, SYSIN, SYSPCH, SYSLST, SYSOUT, SYSLOG, SYSLNK, SYSREC, SYSRLB, SYSSLB, SYSCLB (JCC only,), or SYS000-SYSnnn.	
		X'cuu':	c= 0-6. uu = 00-FE (0-254)in hex	
		address-list :	a list of up to seven device addresses in the form: (X'cuu',...,X'cuu')	
		UA:	unassign	
		IGN:	unassign and ignore (invalid for SYSCLB, SYSRDR, SYSIPT, SYSIN)	
		SYSyyy:	any system or programmer logical unit.	
		device-class:	READER, PRINTER, PUNCH, TAPE, DISK, or DISKETTE	
		device-type:	device code of any supported device	
		X'ss':	density (magn.tape only)	

ss	BPI	Parity	Transl. feat	Conv. feat
10	200	odd	off	on
20	200	even	off	off
28	200	even	on	off
30	200	odd	off	off
38	200	odd	on	off
50	556	odd	off	on
60	556	even	off	off
68	556	even	on	off
70	556	odd	off	off
78	556	odd	on	off
90	800	odd	off	on
A0	800	even	off	off
A8	800	even	on	off
B0	800	odd	off	off
B8	800	odd	on	off
C0	800	single dens 9 tr.		
C0	1600	single dens. 9 tr.		
C0	1600	dual dens. 9 tr.		
C8	800	dual dens. 9 tr.		
D0	6250	single dens. 9 tr.		
D0	6250	dual dens. 9 tr.		

Name	Operation	Operand	Remarks	Accepted by
	ASSGN (Cont'd)	ALT:	specifies alternate tape unit. (Invalid for SYSIPT)	
		H1:	specifies input hopper 1 for input on 2560 or 5425; is assumed if neither H1 nor H2 is specified.	
		H2:	specifies input hopper 2 for input on 2560 or 5425; (invalid for programmers units)	
		PERM:	the assignment is permanent	
		TEMP:	the assignment is temporary	
		VOL=volserno :	volume serial number of the tape or disk required.	
		SHR:	indicates the shared option for disk devices	
		Note:	PRT 1 stands either for a 3211 or a 3204-3 printer. (The 3203-4 Printer cannot be assigned as 3203.)	
	BATCH	{BG / Fn}	Start or continue processers	AR
	CANCEL	{BG / Fn}	Cancels execution of current job in specified area	AR
	CANCEL	blank	Cancels execution of current job	JCC
[//]	CLOSE	SYSxxx [,X'cuu' [,X'ss'] / ,UA / ,IGN / ,ALT]	SYSxxx : for magnetic tape SYSPCH SYSLST SYSOUT SYS000-SYSnnn for DASD (JCC only) SYSIN SYSRDR SYSIPT SYSPCH SYSLST X'cuu', X'ss', UA, IGN, ALT: Values as described in ASSGN command.	JCS JCC

Name	Operation	Operand	Remarks	Accepted by
//	DATE	mm/dd/yy or dd/mm/yy	mm : month (01-12) dd : day (01-31) yy : year (00-99)	JCS
//	DLAB	'label fields 1-3' C xxxx,yyddd,yyddd, 'system code' [,type]	'label fields 1-3': first three fields of Format 1 DASD file label. Is a 51-byte character string, contained within apostrophes and following by a comma. Entire 51-byte field must be contained in the first of the two statements. Field 1 is the file name (44-byte alphameric); field 2 is the format identifier (1-byte numeric); field 3 is the file serial number (6-byte alphameric) C: Any nonblanc character in column 72. xxxx: Volume sequence number (4-digit num.) Must begin in column 16 of the continuation statement. Columns 1-15 are blank. yyddd, yyddd: File creation date followed by file expiration date. Each is 5-digit numeric. 'system-code' : Not required. When used, a 13-character-string within apostrophes. type: SD, DA, ISC or ISE. If omitted, SD is assumed.	JCS
//	DLBL	filename, ['file-ID'], [date], [codes], [,DSF][,BUFSP=n] [,CAT=filename] (See Note 1)	filename : One to seven alphameric characters, the first of which must be alphabetic 'file-ID': One to forty-four alphameric characters (one to eight alphameric characters for the 3540 diskette) date : One to six characters(yy/ddd) codes : Two to four alphabetic characters(SD,DA,DU, ISC,ISE,VSAM) DSF : specifies that a data secured file is to be created or processed BUFSP=n: specifies,for a VSAM file to be processed, the number of bytes of virtual storage(0-999999) to be allocated as bufferspace CAT=filename : specifies filename (1 to 7 alphameric characters)of the DLBL statement for the catalog owing this VSAM file. BLKSIZE=n: a number from 1 to 32,768 (only valid for SAM files on 3350 and 3330-11)	JCS

Name	Operation	Operand	Remarks	Accepted by
	DSPLY	XXXXXX	Displays 16 bytes of virtual storage	AR
	DUMP	blank S BG Fn BGS FnS PDAREA address, address (BG / Fn)	Dumps specified areas of virtual storage Parameter causes dump on the SYSLST assigned to the specified partition. Default is BG SYSLST. blank: General registers plus all real and virtual partitions currently occupied by programs S: General registers, all real and virtual partitions currently occupied by programs, and supervisor area BG, Fn: applicable real or virtual partition currently occupied by progr. and associated registers BGS,FnS: Applicable real or virtual partition currently occupied, registers and supervisor area PDAREA: PD table, PD area and AAA address, address: Specified storage area between the two hexadecimal addresses and associated registers	AR
	DVCDN	X'cuu'	X'cuu': c= 0-6 uu= 00-FE(0-254) in hex	JCC
	DVCUP	X'cuu'	X'cuu': c= 0-6 uu= 00-FE(0-254) in hex	JCC
	END or ENTER	blank	End of SYSLOG communications END for the 3210 and 3215 printer keyboards ENTER for DOC	JCC AR
	ENDSD	blank	Terminates execution of SD aids program	AR

Name	Operation	Operand	Remarks	Accepted by
[//]	EXEC	{[[PGM=] progname] [,REAL] [,SIZE=size]] PROC= procname [,OV]	PGM= progname : one to eight alphameric characters. Used only if the program is in the core image library REAL: The respective program is to be executed in real mode SIZE=size: can be nK, AUTO, or (AUTO,nK) nK : size of area required AUTO : take program size (AUTO,nK) : take program size plus nK PROC=procname : Name of cataloged procedure to be retrieved. One to eight alphameric characters, the first of which must be alphabetic. OV: Indicates that overwrite statements follow EXEC statement	JCC JCS
//	EXTENT	[symbolic unit], [serial number], [type], [sequence number], [relative track], [number of tracks], [split cylinder track], [B=bins]	symbolic unit : Six alphameric characters serial number : One to six alphameric characters type : One numeric character sequence number : One to three numeric characters relative track : One to five numeric characters number of tracks: One to five numeric characters split cylinder track : One or two numeric characters bins : One or two numeric characters	JCS
	HOLD	FN [,Fn] ...	Causes the assignments for the specified foreground partition(s) to remain in affect until the end of the next job	JCC
	IGNORE	blank	Ignore abnormal condition	AR JCC

Name	Operation	Operand	Remarks	Accepted by
//	JOB	jobname [accounting information]	jobname: One to eight alphameric characters accounting information: One to sixteen characters	JCS
//	LBLTYP	{TAPE [(nn)] {NSD (nn)	TAPE: Used when tape files requiring label information, are to be processed and no non-sequential disk files are to be processed (nn): Optional and is present only for future expansion (ignored by job control) NSD: Nonsequential disk files are to be processed (nn): Largest number of extents per single file	JCS
	LFCB	X'cuu',phasename [, FORMS=xxxx] [, LPI=n][, NULMSG]	Causes the FCB of printer X'cuu' to be loaded	AR
[//]	LISTIO	SYS PROG Fn ALL SYSxxx UNITS DOWN UA X'cuu'	Causes listing of I/O assignments on SYSLST for JCS and SYSLOG for JCC	JCS JCC
	LOG	blank	Causes logging of job control statements on SYSLOG	JCC AR
	LUCB	X'cuu',phasename [, FOLD][,NOCHK] [, TRAIN=xxxxxx] [, NULMSG]	Causes the UCB of printer X'cuu' to be loaded	AR
	MAP	blank	Causes a map of area in real and virtual storage to appear on SYSLOG	JCC AR
	MODE	IR CR CE, cuu [, I [, xx,y] , D [, xx,y] , N] R STATUS HIR ECC [, {M} {C}] {[, {R} {Q} {TH}] [, E= eeee] [, T= tttt]}		AR

Name	Operation	Operand	Remarks	Accepted by
	MODE (Cont'd)		Changes the mode of operation, changes the EFL threshold values and gives status information. Note: When HIR or ECC is specified, at least one of the optional operands within these braces must be selected. TH is only valid for the Model 145 when ECC, C is specified with the MODE command	
	MSG	[BG / Fn]	Transfers control to message routine	AR
1 // 1	MTC	opcode, {SYSxxx / X'cuu'} [,nn]	opcode: BSF, BSR, DSE, ERG, FSF, FSR, REW, RUN, or WTM SYSxxx: Any logical unit X'cuu': (only valid for JCC) c=0-6 uu=00-FE (in hex) nn: dec. number (01-99)	JCS JCC
	NEWVOL	[BG / Fn]	Indicates that a new volume has been mounted for the specified partition	AR
	NOLOG	blank	Suppresses logging of job control statements on SYSLOG	JCC AR
//	OPTION	option 1 [,option 2,...]	option: can be any of the following LOG Log control statements on SYSLST NOLOG Suppress LOG option DUMP Dump registers, supervisor area, and temporary real or virtual partition on SYSLST in case of abnormal program end. PARTDUMP Dump registers, selected supervisor control blocks, and temporary real or virtual partition on SYSLST in case of abnormal program end. NODUMP Suppress DUMP or PARTDUMP option LINK Write output of language translator on SYSLNK for linkage editing	JCS

Name	Operation	Operand	Remarks		Accepted by
	OPTION (Cont'd)		NOLINK	Suppress LINK option	
			DECK	Output object module on SYSPCH	
			NODECK	Suppress DECK option	
			EDECK	Punch source macro definitions on SYSPCH	
			NOEDECK	Suppress EDECK option	
			ALIGN	Align constants and date areas on boundaries	
			NOALIGN	Suppress ALIGN option	
			LIST	Output listing of source module on SYSLST	
			NOLIST	Suppress LIST option	
			LISTX	Output of object module on SYSLST	
			SYM	Produce symbol table or data division map on SYSLST	
			NOSYM	Suppress SYM option	
			XREF	Output symbolic cross-reference list on SYSLST	
			NOXREF	Suppress XREF option	
			ERRS	Output listing of all errors in source program on SYSLST	
			NOERRS	Suppress ERRS option	
			RLD	Output listing of RLD information on SYSLST	
			NORLD	Suppress RLD option	
			ACANCEL	Cancel job if attempt to assign device is unsuccessful	
			NOACANCEL	Await operator action if a device cannot be assigned	

Name	Operation	Operand	Remarks		Accepted by
	OPTION (Cont'd)		CATAL	Catalog program or phase in core image library after completion of Linkage Editor run	
			STDLABEL	Causes all DASD or tape labels to be written on the standard label track	
			USRLABEL	Causes all DASD or tape labels to be written on the user label track	
			PARSTD	Causes all DASD or tape labels to be written on the partition standard label track	
			48C	48 - character set	
			60C	60 - character set	
			SYSPARM='string'	specifies a value for assembler system variable symbol and SYSPARM	
			SUBLIB=DF	Causes assembler and ESERV program to retrieve non-edited macros and copy-books from D-sublibrary and edited macros from the F-sublibrary of the source statement library.	
			SUBLIB=AE	Causes assembler and ESERV program to retrieve non-edited macros from the E-sublibrary and edited macros from the E-sublibrary of the source statement library.	

Name	Operation	Operand	Remarks	Accepted by
[//]	OVEND	[comments]	Indicates end of overwrite statements for a cataloged procedure	JCS JCC
[//]	PAUSE	[comments]	Causes pause immediately after processing this statement. PAUSE statement is always printed on SYSLOG. If no 3210, 3215 or DOC is available the statement is ignored.	JCS JCC
	PAUSE	[{BG / Fn} [,EOJ]] where n= 1,2,3 or 4	Causes pause at end of current job step or at end of job	AR
	PRTY	blank partition, partition	Pn= BG,F1,F2,F3 or F4. Allows the operator to display or change the priority of partitions	AR
[//]	RESET	{SYS / PROG / ALL / SYSxxx}	Resets I/O device assignments	JCS JCC
	ROD	blank	Causes all SDR counters for all non-teleprocessing devices on the recorder file on SYSREC to be updated from the SDR counters in main storage	JCC
//	RSTRT	SYSxxx,nnnn[,filename]	SYSxxx: Symbolic unit name of the device on which the checkpoint records are stored. Can be SYS000-SYSnnn nnn: four character identification of the checkpoint record to be used for restart filename: symbolic name of the DASD file to be used for restarting	JCS
	SET	[UPSI=value1] [,LINECT=value2] [,RCLST=value3] [,RCPCH=value4] [,RF=value5] [,DATE=value6] [,HC=value7]	value1: 0, 1 or X value2: standard number of lines for output on each page of SYSLST value3: decimal number indicating minimum number of SYSLST disk records remaining to be written before operator warning	JCC

Name	Operation	Operand	Remarks	Accepted by
	SET (Cont'd)	[,SVA=value 8] [,SDL=value 9]	value 4: decimal number indicating mimimum number of SYSPCH disk records remaining to be written before operator warning value 5: defines to the system the status of the recorder file (IJSYSREC) on SYSREC used by the RMSR feature RF= {YES / CREATE} -file exists / -create file value 6: in one of the following formats: mm/dd/yy or dd/mm/yy mm : month (01-12) dd : day (01-31) yy : year (00-99) value 7: HC= {YES / NO / CREATE} YES: hard-copy file exists NO: No recording performed CREATE: Create a hard-copy file value 8: storage size in the format nK, nK for SVA and GETVIS area, respectively value 9: specify CREATE to have the system directory list (SDL) built in the SVA.	
	START	{BG / Fn}	Same as BATCH	AR
	STOP	blank	Stops batched-job progr. processing	JCC
//	TLBL	filename, ['file-ID'], [date], [File serial number], [volume sequence number], [file sequence number], [generation number], [version number]	filename : One to seven alphameric characters, the first of which must be alphabetic 'file-ID': One to seventeen alphameric characters date: One to six characters (yy/ddd or d-dddd)	JCS

Name	Operation	Operand	Remarks	Accepted by
	TLBL (Cont'd)	Note : For ASCII file processing the fourth and fifth operands are called set identifier and file section number, respectively	{[File serial number (EBCDIC): One to six alphameric characters] [set identifier (ASCII) : Six alphameric characters]} {[volume sequence number (EBCDIC)] [file section number (ASCII)]} One to four numeric characters file sequence number : One to four numeric characters generation number : One to four numeric characters version number : One to two numeric characters	
	TPBAL	[n]	n= number of partitions in which processing can be delayed(0,1,2,.., number of partitions minus one). Allows the operator to display or alter the status of the Tele-processing Balancing function.	AR
//	TPLAB	'label fields 3-10'	'label fields 3-10' : Indicated fields of the standard tape file label for either EBCDIC or ASCII. A 49-byte character string, contained within apostrophes	JCS
//	TPLAB	'label fields 3-10 C label fields 11-13'	'label fields 3-10' : same as above C : Any nonblanc character in column 72 label fields 11-13' : 20 character direct continuation of the same character string begun with fields 3-10 (no blanks, apostrophes or commas separating)	JCS
	UCS	SYSxxx, phasename [,FOLD] [,BLOCK] [,NULMSG]	Causes the 240-character universal character set contained in the core image library phase specified by phasename to be loaded as buffer storage in the IBM 2821 CU. SYSxxx must be assigned to a 1403 or 5203 Printer with the UCS feature.	JCC
	UNBATCH	blank	Terminates foreground processing	JCC
//	UPSI	nnnnnnnn	n : 0, 1 or X	JCS
//	VOL	SYSxxx, filename	SYSxxx Can be SYS000-SYSnnn filename: One to seven alphameric characters, the first of which must be alphabetic	JCS

Name	Operation	Operand	Remarks	Accepted by
//	XTENT	type, sequence, lower, upper, 'serial no.', SYSxxx [,B2]	type: 1 for data area (no split cylinder) 2 for overflow area (for indexed sequential file) 4 for index area (for indexed sequential file) 128 for data area (split cylinder) sequence: sequence number of extent within multiextent file. Can be 0-255 lower: Lower limit of extent in the form $B_1C_1C_1C_2C_2C_2H_1H_2H_2$ where: B_1= 0 for 2311 or 2314/2319; 0-9 for 2321 C_1C_1 = 00 for 2311 or 2314/2319; 00-19 for 2321 $C_2C_2C_2$= 000-199 for 2321 or 2314/2319; 000-009 for 2321 H_1= 0 for 2311 or 2314/2319; 0-4 for 2321 H_2H_2= 00-09 for 2311; 00-19 for 2321 or 2314/2319 Note that the last four strips of subcell 19 are reserved for alternate track for 2321 upper: Upper limit of extent in the same form as for lower limit. 'serial no.': 6-alphameric-character volume serial number contained within apostrophes SYSxxx: Can be SYS000-SYSnnn B2 : 0 for 2311 or 2314/2319; 0-9 for 2321	JCS

Name	Operation	Operand	Remarks	Accepted by
//	ZONE	{EAST / WEST} /hh/mm	EAST: A geographical position east of Greenwich WEST: A geographical position west of Greenwich hh/mm: A decimal value which indicates difference in hours and minutes between local time and Greenwich Mean Time. hh may be in the range 0-12; mm in the range 0-59	JCS
/+	ignored	[comments]	Indicates end of procedure	JCS
/*	ignored	ignored	Columns 1 and 2 are the only columns checked	JCS
/&	ignored	[comments]	Columns 1 and 2 are the only columns checked. Comments appear on SYSLOG and SYSLST at EOJ	
*		comments	Column 2 must be blank	

Note 1: If the DLBL and EXTENT statements for a private core image library are in the input stream (that is, the information is not contained on the label cylinder), they must precede the ASSGN SYSCLB command.

Name	Operation	Operand	Remarks
blank	PHASE	name,origin[,NOAUTO] [,SVA][,PBDY]	name: Symbolic name of the phase. One to eight alphameric characters origin: Specifies the load address of the phase. Load address can be in one of the following formats: 1) symbol[(phase)] [± relocation] 2) *[± relocation] } * 3) S[+ relocation] } * addresses relative to begin of virtual partition 4) ROOT 5) +displacement } ** 6) F +address } ** absolute addresses Note: A phase is eligible for relocation by the relocating loader if its origin is specified as a relative address (formats 1-4 above). However, if a phase is relative to another phase whose origin is specified as an absolute address (formats 5 or 6 above), none of the phases can be made relocatable during this linkage editor execution. Refer to ACTION statement for additional information about the relocating loader. NOAUTO: Indicates that the automatic library lookup (AUTOLINK) feature is suppressed for both the private- and system relocatable libs. SVA: Indicates that the phase is SVA-eligible. PBDY: Indicates that the phase is to be link-edited on a page boundery.
blank	INCLUDE	[modulename][,(namelist)]	If both operands are omitted the object module to be included is assumed to be on SYSIPT If the first operand is present, the object module is assumed to be in either the private- or the system relocatable library If the first operand is omitted and the second operand is present, the object module to be included is assumed to be in the input stream (SYSLNK)

Name	Operation	Operand	Remarks
	INCLUDE (Cont'd)		modulename: Symbolic name of the module as used when cataloged in the relocatable library. It consists of one to eight alphameric characters (namelist): The Linkage Editor constructs a phase from only the control sections specified. The namelist is in the following format: (csname1, csname2, ..) Entries within the parentheses are the names of the control sections that are used to constitute the phase
blank	ENTRY	[entrypoint]	entrypoint: Symbolic name of an entry point If the operand field is blank, the Linkage Editor uses as transfer address the first significant address provided in an END record encountered during generation of the first phase
blank	ACTION	{REL / NOREL} [,CLEAR] [,MAP / ,NOMAP] [,NOAUTO] [,Cancel] [,BG / ,Fn]	Indicates Linkage Editor options: CLEAR: Indicates that the unused portion of the core image library will be set to binary zero before the beginning of the Linkage Editor function MAP: Indicates that SYSLST is available for diagnostic messages. In addition, a virtual storage map is printed on SYSLST NOMAP Indicates that SYSLST is not available when performing the linkedit function NOAUTO Indicates that the AUTO LINK function is to be suppressed CANCEL Cancels the job automatically if any of the errors 21001 through 21701 occur

Name	Operation	Operand	Remarks
	ACTION (Cont'd)		BG: Sets the end-of-supervisor address used in Linkage Editor calculation to the beginning of the partition specified, plus the length of the label area and of the save area If none of these operands are present, the program is link-edited to execute in the virtual partition in which linkedit takes place, unless otherwise specified in the PHASE statement REL Indicates that the phase(s) produced during this execution of the Linkage Editor is to be made relocatable if possible. Refer to origin operand in PHASE statement Note: If support for the relocating loader was generated in the supervisor, ACTION REL is the default NOREL Indicates that the phase(s) produced during this execution of the Linkage Editor is not to be made relocatable Note: If support for the relocating loader was not generated in the supervisor, ACTION NOREL is the default

Maintenance Functions

Function	Unit	Element	Control Statements

Catalog	Core Image Library	Phase	// OPTION CATAL
			(Linkage Editor control statements and if in card form, the phase to be cataloged)
			/*
			// EXEC LNKEDT

	Relocatable Library	Module	// EXEC MAINT CATALR modulename [,v,m] (module to be cataloged)
	Source statem. Library	Book	// EXEC MAINT CATALS sublib.bookname [,v.m],c]] (book to be cataloged)
	Procedure Library	Proce-dure	// EXEC MAINT CATALP procedurename [,VM=v.m] [,EOP=yy][,DATA=NO/YES] (procedure to be cataloged) /+ (or delimiter as specified in EOP parameter)
Delete	Core Image Library	Phase	// EXEC MAINT DELETC phase 1[,phase 2,.....]
		Program	// EXEC MAINT DELETC prog1.ALL[,prog2.ALL,....]
	Relocatable Library	Module	// EXEC MAINT DELETR module1[,module2,.....]
		Program	// EXEC MAINT DELETR prog1.ALL[,prog2.ALL,....]
		Library	// EXEC MAINT DELETR ALL
	Source Statement Library	Book	// EXEC MAINT DELETS sublib.book1[,sublib.book2,..]
		Sub Library	// EXEC MAINT DELETS sublib.ALL
		Library	// EXEC MAINT DELETS ALL
	Procedure Library	Proce-dure	// EXEC MAINT DELETP procedurename[,procedure-name2,.....]
		Library	// EXEC MAINT DELETP ALL
Rename	Core Image Library	Phase	// EXEC MAINT RENAMC oldname,newname[,oldname, newname,....]

Function	Unit	Element	Control Statements
Rename (Cont'd)	Relocatable Library	Module	// EXEC MAINT RENAMR oldname,newname [,oldname, newname,....]
	Source Statement Library	Book	// EXEC MAINT RENAMS sublib.oldname,sublib.newname[,sublib.oldname,sublib.newname, ...]
	Procedure Library	Procedure	// EXEC MAINT RENAMP oldname,newname[,oldname, newname,....]
Update	Source Statement Library	Book	// EXEC MAINT UPDATE sublib.bookname,[s.book1], [v.m],[nn]) ADD,) DEL, or) REP statements as required with source statements to be added) END [v.m],C]]
Condense	Core Image Library	Library	// EXEC MAINT CONDS CL

	Relocatable Library	Library	// JOB jobname
			// EXEC MAINT
			CONDS RL

	Source Statement Library	Library	// EXEC MAINT CONDS SL
	Procedure Library	Library	// EXEC MAINT CONDS PL
	Libraries	All	// EXEC MAINT CONDS CL, RL, SL, PL
Set Parameter for Automatic Condense	Libraries	Any or All	// EXEC MAINT CONDL lib= nnnnn[,lib= nnnnn[,lib= nnnnn]] Notes: Values to be substituted for lib: CL - Core image library RL - Relocatable library SL - Source statement library PL - Procedure library Values to be substituted for nnnnn: One to five decimal digits with a maximum value of 65536.

Reallocation	System	Library	// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE' date, code
			// EXTENT SYSRES, balance of extent information
			// EXEC MAINT
			ALLOC CL=cylin(tracks),RL=cylin(tracks),SL=cylin(tracks),PL=cylin(tracks)

LIBRARIAN (....Cont'd)

Function	Unit	Element	Control Statements
Reallocation (Cont'd)	System (Cont'd)	Library (Cont'd)	Notes: CL- Core image library RL - Relocatable library SL - Source ststement library PL - Procedure library Values to be substituted for cylin and track: Any integer

Note: //JOB, /* and /& must be included where needed

Service Functions

Display Unit	Element	Control Statements
Core Image Library	Phase	// EXEC CSERV DSPLY phase1[,phase2,....]
	Program	// EXEC CSERV DSPLY prog1.ALL[,prog2.ALL,....]
	Library	// EXEC CSERV DSPLY ALL
	Directory	// EXEC DSERV DSPLY CD or DSPLYS CD
	Phase(s) with Version and Modification Level	In the standard position: // EXEC DSERV DSPLY[S] CD(phasename) or CD(phasename) In the nonstandard position or higher than DSERV in use: // EXEC DSERV DSPLY[S] CD(phasename,nn) or CD(phasename ,nn)
Relocatable Library	Module	// EXEC RSERV DSPLY module1[,module2,....]
	Program	// EXEC RSERV DSPLY prog1.ALL[.prog2.ALL,....]
	Library	// EXEC RSERV DSPLY ALL
	Directory	// EXEC DSERV DSPLY RD or DSPLYS RD
Source Statement Library	Book	// EXEC SSERV DSPLY sublib.book1[.sublib.book2,....]
	Sublibrary	// EXEC SSERV DSPLY sublib1.ALL[,sublib2.ALL,....]
	Library	// EXEC SSERV DSPLY ALL
	Directory	// EXEC DSERV DSPLY SD or DSPLYS SD
	Macro-Sublibrary	// EXEC ESERV GENEND DSPLY E. book 1 (,E. book 2, ...)

Display Unit	Element	Control Statements
Procedure Library	Procedure	// EXEC PSERV DSPLY procedurename1[,procedurename2,....]
	Library	// EXEC PSERV DSPLY ALL
	Directory	// EXEC DSERV DSPLY PD or DSPLYS PD
Transient Directory	Directory	// EXEC DSERV DSPLY TD or DSPLYS TD
System Directory	Directory	// EXEC DSERV
Directories	All	// EXEC DSERV DSPLY ALL or DSPLYS ALL
Punch Unit	**Element**	**Control Statements**
Core Image Library	Phase	// EXEC CSERV PUNCH phase1[,phase2,...]
	Program	// EXEC CSERV PUNCH prog1.ALL[,prog2.ALL,....]
	Library	// EXEC CSERV PUNCH ALL
Relocatable	Module	// EXEC RSERV PUNCH module1[,module2,....]
	Program	// EXEC RSERV PUNCH prog1.ALL[.prog2.ALL,....]
	Library	// EXEC RSERV PUNCH ALL
Source Statement Library	Book	// EXEC SSERV PUNCH sublib.book1[,sublib.book2,...][,CMPRSD]
	Sublibrary	// EXEC SSERV PUNCH sublib1.ALL[,sublib2.ALL,....][,CMPRSD]
	Library	// EXEC SSERV PUNCH ALL[,CMPRSD]
	Macro Sublibrary	// EXEC ESERV GENEND PUNCH E.book 1 (,E.book 2, ...)
Procedure Library	Procedure	// EXEC PSERV PUNCH procedurename1[,procedurename2,....]
	Library	// EXEC PSERV PUNCH ALL
Display and Punch Unit	**Element**	**Control Statements**
Core Image Library	Phase	// EXEC CSERV DSPCH phase1[,phase2,....]
	Program	// EXEC CSERV DSPCH prog1.ALL[,prog2.ALL,....]

LIBRARIAN (....Cont'd)

Display and Punch Unit	Element	Control Statements
Core Image Library (Cont'd)	Library	// EXEC CSERV DSPCH ALL
Relocatable Library	Module	// EXEC RSERV DSPCH module1[,module2,....]
	Program	// EXEC RSERV DSPCH prog1.ALL[,prog2.ALL,....]
	Library	// EXEC RSERV DSPCH ALL
Source Statement Library	Book	// EXEC SSERV DSPCH sublib.book1[,sublib.book2,....][,CMPRSD]
	Sublibrary	// EXEC SSERV DSPCH sublib1.ALL[,sublib2.ALL,....][,CMPRSD]
	Library	// EXEC SSERV DSPCH ALL[,CMPRSD]
	Macro Sublibrary	// EXEC ESERV GENEND DSPCH e.book 1 (,E.book 2, ...)
Procedure Library	Procedure	// EXEC PSERV DSPCH procedurename1[,procedurename2,....]
	Library	// EXEC PSERV DSPCH ALL

Note: //JOB, /* and /& must be included where needed

Copy Functions

Copy Unit	Element	Control Statements

Core Image	Phase	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYC phase1[,phase2,....]

	Program	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYC prog1.ALL[,prog2.ALL,....]

Copy Unit	Element	Control Statements

Core Image Library (Cont'd)	Library	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYC ALL

	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYC NEW

Relocatable Library	Module	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYR module1[,module2,....]

	Program	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYR prog1.ALL[,prog2.ALL,....]

	Library	// ASSGN SYS002,X'cuu'
		// DLBL IJSYSRS,'DOS SYSTEM RESIDENCE FILE',
		date,code
		// EXTENT SYS002,balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks),RL=cylin(tracks)
		SL=cylin(tracks),PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYR ALL

Copy Unit	Element	Control Statements

Relocatable Library Cont'd	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYR NEW

Source Statement Library	Book	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYS sublib.book1[,sublib.book2,....]

	Sublibrary	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYS sublib1.ALL[,sublib2.ALL,....]

	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYS ALL

	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE',
		date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks)
		SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYS NEW

Copy Unit	Element	Control Statement

Procedure Library	Procedure	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks) SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYP procedurename1[,procedurename2,....]

	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks) SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYP ALL

	Library	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks) SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPYP NEW

Libraries	All	// ASSGN SYS002, X'cuu'
		// DLBL IJSYSRS, 'DOS SYSTEM RESIDENCE FILE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		ALLOC CL=cylin(tracks), RL=cylin(tracks) SL=cylin(tracks), PL=cylin(tracks)
		* ALLOC STATEMENT MUST ALWAYS BE PRESENT
		COPY ALL

Copy Unit	Element	Control Statements

Definition of a Private Library (See note 2)	Core Image	// ASSGN SYS003, X'cuu'
		// DLBL IJSYSPC, 'user identification of private library', date, code
		// EXTENT SYS003, balance of extent information
		// EXEC CORGZ
		NEWVOL CL= cylin(tracks)

	Relocatable	// ASSGN SYSRLB, X'cuu'
		// DLBL IJSYSRL, 'user identification of private library', date, code
		// EXTENT SYSRLB, balance of extent information
		// EXEC CORGZ
		NEWVOL RL= cylin(tracks)

	Source Statement	// ASSGN SYSSLB, X'cuu'
		// DLBL IJSYSSL, 'user identification of private library', date, code
		// EXTENT SYSSLB, balance of extent information
		// EXEC CORGZ
		NEWVOL SL= cylin(tracks)

Definition and Creation of a Private Library (See note 2)	Core Image	// ASSGN SYS003, X'cuu'
		// DLBL IJSYSPC, 'user identification of private library', date, code
		// EXTENT SYS003, balance of extent information
		// EXEC CORGZ
		NEWVOL CL= cylin(tracks)
		COPYC operands

	Relocatable	// ASSGN SYSRLB, X'cuu'
		// DLBL IJSYSRL, 'user identification of private library', date, code
		// EXTENT SYSRLB, balance of extent information
		// EXEC CORGZ
		NEWVOL RL= cylin(tracks)
		COPYR operands

	Source Statement	// ASSGN SYSSLB, X'cuu'
		// DLBL IJSYSSL, 'user identification of private library', date, code
		// EXTENT SYSSLB, balance of extent information
		// EXEC CORGZ
		NEWVOL SL= cylin(tracks)
		COPYS operands

Merge System Residence to New System Residence		// ASSGN (statements as required)
		// DLBL IJSYSRS, 'NEW SYSTEM RESIDENCE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		MERGE RES, NRS
		COPY statements (COPYC, COPYR, COPYS, COPYP, COPYI) as required

Copy Unit	Element	Control Statements

Merge New System Residence to System Residence		// ASSGN (statements as required)
		// DLBL IJSYSRS, 'NEW SYSTEM RESIDENCE', date, code
		// EXTENT SYS002, balance of extent information
		// EXEC CORGZ
		MERGE NRS, RES
		COPY statements (COPY1, COPYC, COPYR, COPYS, COPYP, COPY) as required

Merge System Residence to Private Libraries		// ASSGN (statements as required)
		// DLBL IJSYSRL, 'PRIVATE RELOCATABLE LIBRARY', date, code
		// EXTENT SYSRLB, balance of extent information
		// DLBL IJSYSSL, 'PRIVATE SOURCE STATEMENT LIBRARY', date, code
		// EXTENT SYSSLB, balance of extent information
		// DLBL IJSYSCL, 'PRIVATE CORE IMAGE LIBRARY', date, code
		// EXTENT SYSCLB, balance of extent information
		ASSGN SYSCLB, X'cuu'
		// EXEC CORGZ
		MERGE RES, PRV
		COPY statements (COPY1, COPYR, COPYS, COPYC) as required

Merge New System Residence to Private Libraries		// ASSGN (statements as required)
		// DLBL IJSYSRS, 'NEW SYSTEM RESIDENCE', date, code
		// EXTENT SYS002, balance of extent information
		// DLBL IJSYSRL, 'PRIVATE RELOCATABLE LIBRARY', date, code
		// EXTENT SYSRLB, balance of extent information
		// DLBL IJSYSSL, 'PRIVATE SOURCE STATEMENT LIBRARY', date, code
		// EXTENT SYSSLB, balance of extent information
		// DLBL IJSYSCL, 'PRIVATE CORE IMAGE LIBRARY', date, code
		// EXTENT SYSCLB, balance of extent information
		ASSGN SYSCLB, X'cuu'
		// EXEC CORGZ
		MERGE NRS, PRV
		COPY statements (COPYR, COPYS, COPYC) as required

Merge Private Libraries to System Residence		// ASSGN (statements as required)
		// DLBL IJSYSPR, 'PRIVATE RELOCATABLE LIBRARY', date, code
		// EXTENT SYS001, balance of extent information
		// DLBL IJSYSPS, 'PRIVATE SOURCE STATEMENT LIBRARY', date, code
		// EXTENT SYS000, balance of extent information
		// DLBL IJSYSPC, 'PRIVATE CORE IMAGE LIBRARY', date, code
		// EXTENT SYS003, balance of extent information

Copy Unit	Element	Control Statements
Merge Private Libraries to System Residence (Cont'd)		// EXEC CORGZ MERGE PRV,RES COPY statements (COPYR,COPYS,COPYC) as required

Merge Private Libraries to New System Residence		// ASSGN (statements as required)
		// DLBL IJSYSRS,'NEW SYSTEM RESIDENCE',
		date,code
		// EXTENT SYS002,balance of extent information
		// DLBL IJSYSPR,'PRIVATE RELOCATABLE LIBRARY',
		date,code
		// EXTENT SYS001,balance of extent information
		// DLBL IJSYSPS,'PRIVATE SOURCE STATEMENT
		LIBRARY',date,code
		// EXTENT SYS000,balance of extent information
		// DLBL IJSVSPC,'PRIVATE CORE IMAGE LIBRARY',
		date,code
		// EXTENT SYS003,balance of extent information
		// EXEC CORGZ
		MERGE PRV,NRS
		COPY statements (COPYR,COPYS,COPYC) as required

Merge Private Libraries to Private Libraries		// ASSGN (statements as required)
		// DLBL IJSYSRL, 'NEW PRIVATE RELOCATABLE
		LIBRARY', date, code
		// EXTENT SYSRLB,balance of extent information
		// DLBL IJSYSPR, 'EXISTING PRIVATE RELOCATABLE
		LIBRARY', date, code
		// EXTENT SYS001,balance of extent information
		// DLBL IJSYSSL, 'NEW PRIVATE SOURCE STATEMENT
		LIBRARY, date, code
		// EXTENT SYSSLB,balance of extent information
		// DLBL IJSYSPS, 'EXISTING PRIVATE SOURCE
		STATEMENT LIBRARY', date, code
		// EXTENT SYS000,balance of extent information
		// DLBL IJSYSCL, 'NEW PRIVATE CORE IMAGE
		LIBRARY', date, code
		// EXTENT SYSCLB,balance of extent information
		ASSGN SYSCLB,X'cuu'
		// DLBL IJSYSPC, 'EXISTING PRIVATE CORE IMAGE
		LIBRARY', date code
		// EXTENT SYS003,balance of extent information
		// EXEC CORGZ
		MERGE PRV,PRV
		COPY statements (COPYR,COPYS,COPYC) as required
		To define the private library in the same jobstep, precede MERGE with NEWVOL statement (except for COPYC statements).

Notes : 1) //JOB, /* and/& must be included where needed
2) The private library can be updated with either a MAINT or a copy MERGE function

LIBRARIAN (....Cont'd)

Direction of Transfer for Merge Operations

File Name	IJSYSRS	IJSYSRS	IJSYSRL	IJSYSPR	IJSYSSL	IJSYSPS	IJSYSCL	IJSYSPC
Logical Unit	SYSRES	SYS002	SYSRLB	SYS001	SYSSLB	SYS000	SYSCLB	SYS003
Merge RES to NRS	from	to						
Merge NRS to RES	to	from						
Merge RES to PRV	from		to		to		to	
Merge NRS to PRV		from	to		to		to	
Merge PRV to RES	to			from		from		from
Merge PRV to NRS		to		from		from		from
Merge PRV to PRV			to	from	to	from	to	from

ESERV

Function: De-editing and/or updating of edited macro's.
(Refer to: Guide to DOS/VS Assembler (GC33-4024))

Control Statement	Meaning
// EXEC ESERV	
GENEND	Causes generation of an assembler END statement, and a /* statement. GENEND, when present, must be the first statement after the // EXEC ESERV statement.
GENCATALS	Causes generation of the appropriate CATALS statement before each macro in the stream, and a /* statement after the last macro. GENCATALS, when present, must be the first statement after the // EXEC ESERV statement. Note: If neither GENEND nor GENCATALS is used, GENCATALS is assumed.
DSPLY sublib1.mac1,sublib1.mac2,.....	Produces a printout of the de-edited macro on the device assigned to SYSLST.
PUNCH sublib1.mac1,sublib1.mac2,....	Produces a de-edited deck on the device assigned to SYSPCH.
DSPCH sublib1.mac1,sublib1.mac2,.....	Produces a deck and printout of the de-edited macro on the devices assigned to SYSPCH and SYSLST respectively.
) COL startcol,n	Specifies the columns containing the sequence numbers in the statement of a macro definition. startcol: A decimal number within the range 73-80, which identifies the start column of the sequence number. n: A decimal number within the range 1-8, specifying the number of columns used by the sequence number. COL, when present must be the first statement following DSPLY, PUNCH or DSPCH statement. If the COL statement is omitted startcol receives a default value of 73 and n a default value of 6.
) VER seqno+rel,l	Specifies the source statement of a macro definition which contents are to be verified. seqno+rel: Identifies the source statement which is to be compared with the statement following the VER statement. seqno: The sequence number of a source statement.

Control Statement	Meaning
) VER seqno+rel,l (Cont'd)	rel: A decimal number of 1-4 digits in length. If omitted, 0 is assumed. l: A decimal number within the range 1-80. If omitted, 72 is assumed. Only the first l characters are used in the comparison.
) ADD seqno+rel	Add statements to a source macro definition. seqno+rel: Identifies the source statement after which the new statements following the ADD statement are to be inserted. (seqno: See VER statement) (rel : See VER statement)
) DEL seqno+rel,seqno+rel	Delete statements from a macro source definition. seqno+rel,seqno+rel: Identifies the first and the last source statement of the section to be deleted. If the second operand is omitted, only the source statement identified by the first operand is deleted. (seqno: See VER statement) (rel : See VER statement)
) REP seqno+rel,seqno+rel	Replace statements in a source macro definition. seqno+rel,seqno+rel: Identifies the first and the last source statement of the section which is to be replaced by the statements following the REP statement. If the second operand is omitt omitted, only the source statement identified by the first operand is replaced. (seqno: See VER statement) (rel : See VER statement)
) RST seqno+rel	Specifies a new sequence number serial starts. seqno+rel: Identifies the source statement after which the new serial starts. (seqno: See VER statement) (rel : See VER statement)
) END	Indicates the end of an update to a macro definition. Required statement for all updating.

Sample coding for de-editing without updating a macro definition

```
// JOB NOUPDATE
// EXEC ESERV
 PUNCH E.MAC1,E.MAC2
/*
/&
```

Sample coding for de-editing and updating a macro definition

```
// JOB UPDATE
// EXEC ESERV
 GENEND
 DSPCH E.MAC1
) COL 77,4
) VER 72+1,5
.PP9
) ADD 72+1
   AIF (&PCH NE 1400)D4
) DEL 102,103
) REP 245
JOYCE  CLC 0(4,REG6),BLANKS
) END
/*
// PAUSE  CHECK LIST,MOVE DECK TO READER
// OPTION EDECK,NODECK
// EXEC ASSEMBLY
      :
      :     deck produced by ESERV
      :
// PAUSE MOVE SYSPCH DECK TO READER
// EXEC MAINT
      :
      :     deck produced by assembler
      :
/*
/&
```

The label information cylinder is on the first full cylinder after the last system library on SYSRES. A display of all labels on the cylinder, with the exception of Data Set Secured labels, can be obtained by executing LSERV. Illustrations in this section show the location of the label information cylinder on SYSRES, and the layout of label information and record format.

System requirements

LSERV may be executed in any partition, with a minimum of 8192 bytes of the real or virtual address areas. LSERV assumes that the SYSRES label cylinder is formatted as described in DOS/VS DASD Labels.

Executing LSERV

The control statements necessary to execute LSERV in a virtual partition are :

From the console :

```
// EXEC LSERV
```

From the reader:

```
// JOB jobname
// EXEC LSERV
/*
/&
```

LSERV can also be executed in a real partition. The output of LSERV shows the contents of the label cylinder on the device assigned to SYSRES. The output is directed to the device assigned to SYSLST.

When and How to use

1. Operator action given in DOS/VS Messages indicates when LSERV must be executed.
   Programmer action, also given in DOS/VS Messages,explains how to use the LSERV printout.
   For example, under the message :
   OP36 NO REC FND

2. It is useful to execute LSERV prior to running a program that is known to have been run sometime in the past, but whose workfile assignments and partition allocations are unknown.

3. LSERV can be used for error analysis. LSERV displays the TLBL and the DLBL and EXTENT information contained on the SYSRES label cylinder. Information about secured data files is not displayed.

Summary of information provided

The printout of LSERV will show you the following details about the previous run :

- Whether the correct DLBL/EXTENT information is still on the label cylinder
- The permanent files
- The temporary files
- Extent type
- File type

For more information,
Refer to DOS/VS Serviceability Aids and Debugging Procedures (GC33-5380).

Supervisor Macro Instruction

Operation	Operand	Explanation
SUPVR	AP= {NO / YES}	Specify if there is to be multiprogramming within a partition (multitasking) support. AITM=YES is assumed if AP=YES. If the operand (NO or YES) is incorrectly entered, YES is assumed. If AP=YES is specified and NPARTS is omitted or specifies 1, NPARTS=3 is forced. AP=YES must be specified if support for Power/VS with RJE, SNA is required.
	ASCII= {NO / YES}	Specify YES if supervisor support of ASCII code is desired.
	CHAN= {NO / YES} (Models 115 and 125 only)	Specify if channel attached devices, tape units or tp devices are to be added to a Model 115 or 125 CPU during system generation or at IPL time. CHAN=YES causes RMSR support to be generated. CHAN=NO is ignored if RMS=YES. RMS=YES overrides a specification of chan; it should be specified if full error recovery is required in addition to error recording.
	ERRLOG= {YES / RDE}	Specify RDE to include Reliability Data Extractor recording in addition to normal RMSR recording (YES). For the Model 115 or 125, CHAN=YES or RMS=YES must be specified if RDE is to be used.
	EU= {NO / YES}	YES must be specified if mixed parity tape processing is required for the emulators, the tape preprocessor and the tape post-processor programs.
	MCH= {YES / NO} (models 115 and 125 only)	Specify if the supervisor is to provide MCAR/CCH support for a Model 115 or 125 CPU. RMS=YES overrides MCH=NO. MCH=YES in conjunction with CHAN=YES is equivalent to specify in RMS=YES. In addition to the software recording services generated by specifying CHAN=YES.
	MICR= {NO / 1419 / 1419D}	Indicates whether the supervisor is to support magnetic ink character readers or optical reader/sorter. If 1419's or 1255/1270/1275's are attached to a multiplexer channel, the PIOCS parameter BMPX=YES is supported; however, burst mode and MICR devices cannot run concurrently on the same byte multiplexer channel. (3886 OCR's do not require MICR support).

Supervisor Macro Instruction (...Cont'd)

Operation	Operand	Explanation
SUPVR (Cont'd)	NPARTS= {1 / 3 / n}	Specifies the number of partitions to be supported. The maximum value for n is 5. The default value is 1 if the AP parameter is omitted, or if AP=NO. The default value is 3 if AP=YES. If POWER=YES is specified, NPARTS=2, 3, 4, or 5 must be specified unless AP=YES is specified.
	ID= {1 / c}	Specify this parameter if you plan to use the Supervisor Select option. You can assign a unique name to the supervisor being generated by specifying any alphameric character (A-Z, 1-9) in this field.
	PAGEIN= {NO / n}	Specify that paging activity is to be controlled by the PAGEIN, RELPAG, and FCEPGOUT macros. The value n indicates the maximum of page-in requests that can be queued at any one time for execution.
	PHO= {NO / YES}	Specify if page fault handling overlap is desired for private multitasking. YES is required if SETPFA macro instructions are used. When PHO=YES, PFIX=YES is assumed.
	POWER= {NO / YES}	Specify YES for support of POWER/VS. If NPARTS=1 or omitted, POWER=YES causes NPARTS=3 to be generated. For details of generation parameters, refer to Power Section.
	RMS= {NO / YES} (models 115 and 125 only)	Specify YES for RMS support (MCAR/CCH and RMSR) for a Model 115 or 125 CPU. RMS=YES overrides specification of CHAN and/or MCH. RMS=YES is required if the string switch feature is installed. RMS=YES should be specified if a multiple partition system is used.

Supervisor Macro Instruction (...Cont'd)

Operation	Operand	Explanation
SUPVR Cont'd	TP= {NO / VTAM / BTAM [,VTAM] / QTAM [,VTAM] / QTAMn [,VTAM]}	Specify if teleprocessing support is desired. QTAM includes BTAM support. If AP=YES, n specifies the maximum number of active QTAM message processing programs in the system. From 2 to 12 may be specified. The default value for n is 2. If AP=NO, n is always 2. If QTAM is specified, then NPARTS=2, 3, 4, or 5 must be specified unless AP=YES is specified. If BTAM runs virtual, PFIX=YES is required. If VTAM is specified, AP=YES must be specified. Support is included for TP Balancing, unless TP=NO is specified or TP=BTAM and NPARTS=1. TP= if Power =YES with RJE, SNA, TP=VTAM must be specified.

Describe the Hardware Features

Operation	Operand	Explanation
CONFG	FP= {NO / YES}	Floating Point Feature
	MODEL= {115 / 125 / 135 / 138 / 145 / 148 / 155-II / 158}	Specify the CPU model number. Model=135, 138, 145, 148, 155-II or 158 forces MCAR/CCH and RMSR to be generated. When Model=125 is specified, support for DOC and 3330 is always generated. When Model=115 is specified, support for DOC and 3340 is always generated.

Specify Standard settings for Job Control

Operation	Operand	Explanation
STDJC	ALIGN= {YES / NO}	Specify if the assembler is to align data on halfword or fullword boundaries.
	ACANCEL= {NO / YES}	Specify if Job Control is to cancel jobs (ACANCEL=YES) or await operator intervention (ACANCEL=NO) after an unsuccessful attempt to assign a device.
	CHARSET= {48C / 60C}	Specify the 48- or 60 character set for PL/1 translator input on SYSIPT.
	DATE= {MDY / DMY}	Format of the date.

Specify Standard settings for Job Control (...Cont'd)

Operation	Operand	Explanation
STDJC (cont'd)	DECK= {YES / NO}	Output of object modules of language translators on SYSPCH.
	DUMP= {YES / NO / PART}	Dump of registers and virtual storage on SYSLST. PART dump of supervisor control blocks and virtual storage of the partition on SYSLST.
	EDECK= {NO / YES}	Specify if the assembler is to create and punch edited macros on SYSPCH.
	ERRS= {YES / NO}	COBOL, PL/1, FORTRAN and basic FORTRAN summarize all errors in source programs on SYSLST.
	LINES= {56 / nn}	Number of lines per page on SYSLST (nn must be between 30 and 99).
	LIST= {YES / NO}	Source module listings and diagnostics from language translators on SYSLST.
	LISTX= {NO / YES}	Hexadecimal object module listings from PL/1 and COBOL on SYSLST.
	LOG= YES / NO	Listing of all control statements on SYSLST
	RLD= NO / YES	Specifies whether RLD information is to be printed.
	SPARM= {NO / YES}	Support of assembler system variable symbol &SYSPARM
	SYM= {NO / YES}	PL/1 outputs symbol tables on SYSPCH. The American National Standard COBOL compiler produces a data division glossary.
	XREF= {YES / NO}	Assembler and American National Standard COBOL compiler output symbolic cross-reference lists on SYSLST.

Specify Optional Support in the Supervisor

Operation	Operand	Explanation
FOPT	AB= {NO / YES}	Specify if the abnormal termination exit function is to be supported. The abnormal termination exit allows the user to exit to a user's routine before an abnormal end of job causes a program to be cancelled. Specify YES for American National Standard COBOL I/O error recovery. AB=YES is forced by VSAM=YES or TP=VTAM.

Specify Optional Support in the Supervisor (...Cont'd)

Operation	Operand	Explanation
FOPT (Cont'd)	CBF= {NO / n}	Specify if I/O requests are to be appraised for console buffering and indicate the number of buffers (3-25) to be generated. CFB=NO is forced if Model = 115, 125, 138 or 148 is specified together with doc = 125D or 3277.
	DASDFP= {NO / n1, n2[,2321]}	Specification of n^1, n^2 provides file protection for disk devices. For reasons of compatibility with previous releases, specifications for all disk devices are accepted, but treated as documentation (that is, the parameter is ignored)
	DOC= {NO / 1250 / 3277}	Specify if support for the Display Operator Console (DOC) is to be generated. The default is NO if MODEL=135, 145, 155-II, or 158. For MODEL=115 or 125, DOC=125D is forced. If Model=138,148 the default is 3277.
	ECPREAL= {NO / YES}	Specification of this parameter allows use of the VIRTAD macro, and of the REALAD macro and of the EXCP macro with the REAL parameter.
	ERRQ= {5 / 3 / n}	Specify the number of entries for the error queue. Without multiprogramming n may be from 3 to 25. With multiprogramming support n may be from 5 to 25. The lower value is the default value in each case.
	EVA= {NO / r, w}	Specify if error volume analysis is supported. r is the read error parameter; w is the write error parameter.
	FASTTR= {NO / YES}	Specify if fast CCW translation is to be supported. Fast translation is attempted for all channel programs except : 1. Those containing non-contiguous CCW strings. 2. Those associated with translation requests from BTAM.
	GETVIS= {NO / YES}	Specify storage management support, using GETVIS and FREEVIS macro instructions. YES is assumed if VSAM=YES or TP=VTAM. GETVIS=YES forces RELLDR=YES. Getvis=yes if 3344/3350 AP 1 program will be invoked.
	IDRA= {NO / YES}	Specify the independent directory read-in area (IDRA). If IDRA=YES, NPARTS must be >1.

Operation	Operand	Explanation
FOPT (cont'd)	IT={NO / YES}	STXIT option is available for interval timer interruption. IT=YES generates timer support for all tasks in all partitions. YES is forced by TP=VTAM. Specify yes if 3344/3350 AP-1 program will be invoked.
	JA={NO / YES / (n1,n2,n3,n4,n5)}	Specify if job control job accounting interface is to be supported.
	JALIOCS={NO / (n1,n2)}	Specify for support of user-written job control job accounting interface routines containing LIOCS, and LIOCS with label processing
	OC={NO / YES}	STXIT option is available for external interruptions (except timer) to problem programs. YES is required if emulator program operator services are to be requested through the INTERRUPT key. OC=YES is forced if RETAIN=YES, OLTEP=YES or TP=VTAM.
	OLTEP={YES / NO}	Specify if the on-line testing function is desired. If OLTEP=YES then OC=YES and RELLDR=YES are forced. OLTEP=YES is forced if RETAIN=YES.
	PC={NO / YES}	STXIT option is available for program check interruption. Included in supplied supervisor because QTAM, FORTRAN, COBOL-D, PL/1 and RPG require PC=YES.
	PCIL={NO / YES}	Specify if private core image libraries are to be supported. PCIL=NO forces PSLD=NO.
	PD={NO / YES / n}	Specify the number of bytes to be allocated to the problem determination programs. Any amount between 1400 and 10240 may be specified for n. Specification of YES provides the minimum number of 1400 bytes.
	PFIX={NO / YES}	Specify if the supervisor is to support the fixing and freeing of pages in real partitions. PFIX=YES is forced by PHO=YES, POWER=YES, or TP=VTAM.
	PRTY=(partition name sequence)	Specify the desired dispatching priority for each partition (from low to high).
	PSLD={NO / n}	Specify the number of entries in the Private Second Level Directories. A directory is created for each partition. The minimum value for n is 5. PCIL=NO forces PSLD=NO.

Specify Optional Support in the Supervisor (...Cont'd)

Operation	Operand	Explanation
FOPT (Cont'd)	RELLDR= {NO / YES}	Specify relocating load support in the supervisor. RELLDR=YES is forced by VSAM=YES, GETVIS=YES, OLTEP=YES, RPS=YES, RETAIN=YES, and TP=VTAM.
	RETAIN= {YES / NO}	Generates support for relocating relocatable phases. The default value is YES. YES is forced by VSAM=YES, OLTEP=YES, GETVIS=YES, RPS=YES, RETAIN=YES, and TP=VTAM.
	SKSEP={NO / YES / n}	Specify if seeks are to be separated from the remainder of channel programs. Seek separation allows other devices on the channel to be accessed (including other seeks) during the seek. YES indicates support for all DASD type devices specified by the DVCGEN macro. n is the number of DASD devices to be supported and can not be less than the number of DASD devices specified at system generation. The maximum number is 254.
	SLD= {5 / n}	Specify the number of entries in the Second Level Directory. The minimum value for n is 5.

Specify Optional Support in the Supervisor (...Cont'd)

Operation	Operand	Explanation
FOPT (cont'd)	SYNCH= {NO / YES}	Specify SYNCH=YES if the use of an IBM provided program requires the synchronous exit facility (SVC) screening.
	SYSFIL= {NO / YES / (YES,n1,n2)}	Specify if system input and system output files (SYSRDR, SYSIPT, SYSLST, SYSPCH) may be assigned to a disk device or an 3540 Diskette or, if extended support for the procedure library is desired. In a disk only configuration SYSFIL=YES is required for system maintenance. n1 = residual capacity for beginning of operator notification where SYSLST assigned to a disk device. 100≤n1≤65535. If n1 is omitted, 1000 is assumed. n2= residual capacity for beginning of operator notification where SYSPCH assigned to a disk device. 100≤n2≤65535. If n2 is omitted, 1000 is assumed. Note : If neither n1 or n2 is specified, the operand need not be placed between parentheses.
	TEB= {NO / n}	Specify if tape error statistics are to be accumulated and logged for the 2495 Tape Cartridge Reader, where n is the number of tape cartridge readers attached to the system. Allow extra TEB's for future expansion of the system.
	TEBV= {IR / CR}	Specify the type of error recording (combined or individual) to be performed for unlabeled or non-standard tapes.

Specify Optional Support in the Supervisor (...Cont'd)

Operation	Operand	Explanation
FOPT (cont'd)	TOD= {NO / YES}	Specify time of day clock support. TP=VTAM forces TOD=YES.
	TTIME= {NO / partition ID}	Specify if the task timer support is to be generated in the supervisor. The partition ID (BG, F1, F2, etc.) designates the partition that owns the task timer. Support must be generated for the partion that owns the task timer by corresponding specification in the NPARTS parameter of SUPVR macro. Example: with NPARTS=2 specified, valid partition ID specifications are BG or F1. If NPARTS=3, Part. ID, BG, F1 or F2 is valid.
	TRKHLD= {NO / n}	Specify of the track hold function is to be supported. The maximum number of tracks that can be held at one time is 225. The default is 10 if n is an invalid operand (non-numeric or outside the range 1-255). If TRKHLD=n is specified, NPARTS must be 1.
	USERID= id	Specify if you want a supervisor id to be printed as part of the IPL COMPLETE message. id may be up to 16 bytes long.
	VSAM= {NO / YES}	Specify if supervisor support of the Virtual Storage Access Method (VSAM) is desired. If VSAM=YES, YES is assumed for GETVIS, AB and RELLDR.
	WAITM= {NO / YES}	Specify if multiple wait function is to be supported. WAITM=YES is assumed if AP=YES or TP=VTAM.

Specify Optional Support in the Supervisor (...Cont'd)

Operation	Operand	Explanation
FOPT (Cont'd)	XECB= {NO / YES / n}	Specifies whether Cross Partition Event Control is to be supported. If YES is specified, four XECB's are generated per partition (as specified in NPARTS) n must be specified as a numeric value and indicates the number of XECB's for which an entry is to be generated in a supervisor internal table.
	ZONE= {NO / EAST / WEST ,hh,mm}	Specify the difference between Greenwich Mean Time and local time in hours (hh) and minutes (mm). Use EAST for areas east of Greenwich and WEST for areas west of Greenwich. If TOD=NO is specified, ZONE=NO is assumed. If the first operand is incorrectly specified, EAST is assumed.
	RPS= {NO / YES}	Provides support for the Rotational Position Sensing (RPS) capabilities of DASD devices supporting the feature. RPS=YES forces BLKMPS=YES in the PIOCS macro instruction and GETVIS=YES and RELLDR=YES in the FOPT macro.

Define options and Configuration requirements to be included in Physical IOCS

Operation	Operand	Explanation
PIOCS	BLKMPX= {NO / YES}	Specify if block multiplexer mode is to be supported for integrated block multiplex channels. Must be NO for disk emulation.
	BMPX= {NO / YES}	Burst mode device support on multiplexer channel. If overrunnable devices are attached to the multiplexer channel, BMPX= YES is supported. However, burst mode and MICR devices cannot run concurrently on the same byte multiplexer channel.
	CHANSW= {NO / RWTAU / TSWTCH}	Specify if channel switching. For a 2404 or 2804, enter CHANSW=RWTAU. Enter CHANSW=TSWTCH for a 2816 (with a 2403 or 2803) or a 3803.

Operation	Operand	Explanation
PIOCS (Cont'd)	DISK={2311 / 2314 / 3330 / 3340 / 3350 / (3330,3340) / (3330,3350) / (3340,3350) / (3330,3340,3350)}	Support for the IBM 2311, 2314/2319. Specification of DISK=33xx gives support for all attachable DASD devices of the type specified. If support for more than one type of DASD device is required, each type must be specified. For example, to provide support for 3330s and 3340s, specify DISK=(3330,3340). This provides no support for 3350s.
	MRSLCH={NO / YES}	Specify if MICR device is on the selector channel (can only be specified for the 1419 single address device).
	TAPE={7 / 9 / NO}	Specify PIOCS tape support. Support is generated for both 7- and 9 track tape unless NO is specified.

Specify Size of Real and Virtual Address Areas

Operation	Operand	Explanation
VSTAB	RSIZE=nK	Specifies the maximum size of real storage. n may be up to 81900 and must be a multiple of 2. If not specified, or specified incorrectly, 96 is assumed.
	VSIZE= nK	Specify the size of the virtual address area. n must be≥64 and a multiple of 2. The maximum value that can be substituted for n is 16K (16384) minus the value substituted for n in the RSIZE parameter.
	BUFSIZE=n	Specify the number of copy blocks to be used by the Channel Program Translation routine. The following table shows the minimum value that can be specified and default value that is used if the operand is omitted.
	SVA={(64K,0K) / (nK, mK)}	nK specifies the size of the Shared Virtual Area (SVA); mK specifies the size of the system GETVIS area in the SVA. n must be at least 64 and a multiple of 2. m must be smaller than n and also a multiple of 2.

FASTTR=	NPARTS=	BUFSIZE= Default	BUFSIZE= Minimum
NO	1	10	10
NO	2-5	30	10
YES	1	30	30
YES	n(=2-5)	30+(n x 20)	30

Specify partitioning of Virtual Address Area

Operation	Operand	Explanation
ALLOC	Fn=mK [,Fn=mK]	Operands may be specified in random order. m must be a multiple of 2 and should not be less than 64, unless 0 is specified.

Specify partitioning of Real Address Area (less Supervisor and Main Page Pool)

Operation	Operand	Explanation
ALLOCR	BGR= mK [,FnR=mK]	Operands may be specified in random order. m must be a multiple of 2. Minimum amount of real storage to be reserved for the main page pool is : 18K bytes (+2K if AP=YES) if PFIX=YES, or 18K bytes (+2K if AP=YES) minus the size of the smallest real partition if PFIX=NO.

Define the necessary Input/Output Tables for the System

Operation	Operand	Explanation
IOTAB	BGPGR= {10 / n}	Specify the number of logical unit blocks (LUB) for programmer units i.e., the number of symbolic programmer logical units (SYS000-SYSnnn).
	Fn PGR ={5 / m}	Specify the number of symbolic programmer logical units (of the class SYSnnn) for the foreground partition F1. The minimum value for m is 5, and the maximum is 241. For the remaining foreground partitions the minimum value for m is 5 while the maximum depends on the number of partitions specified. Refer to SYS Gen. Manual.
	JIB= {5 / n}	Number of JIB's for the system. Minimum value generated is 5. Max. is 255
	NRES= {10 / n}	Specifies the number of named resources in the system. n can have any value between 2 and 255.

<table>
<tr><th>Operation</th><th>Operand</th><th>Explanation</th></tr>
<tr><td>IOTAB
(Cont 'd)</td><td>CHANQ={10 / 8 / 6 / n}</td><td>Number of entries in the channel queue
Minimum value is 6 if NPARTS=1, 2 or 3; 8 if NPARTS=4; 10 if NPARTS=5 etc.
For default values see Sys.Gen.Man.</td></tr>
<tr><td></td><td>IODEV={10 / n}</td><td>Specify the number of I/O device attached to the system. The minimum value is 5.</td></tr>
<tr><td></td><td>D2311=n
D2314=n
D2321=n
D2400=n
D3330=n
D3340=n
D3350=n
D3410=n
D3420=n
D3540=n
D3886=n</td><td>Specify, for each I/O device listed at the left, the number of drives that are attached to our system. If the device is not specified, the system assumes that no devices of that type are attached. However, defaults other than zero are taken depending on the CPU Model specified and (for tape devices) whether PIOCS tape support is generated or not (see table below):
<table>
<tr><th rowspan="2">Default if</th><th rowspan="2">PIOCS Macro: TAPE={7 / 9}</th><th colspan="3">Config. Macro Model</th></tr>
<tr><th>115</th><th>125</th><th>135,135-3,
138,145,
145-3,148,
155-II,158</th></tr>
<tr><td>D2314=2</td><td></td><td></td><td></td><td>X</td></tr>
<tr><td>D2400=4</td><td>X</td><td></td><td></td><td>X</td></tr>
<tr><td>D3330=2</td><td></td><td></td><td>X</td><td></td></tr>
<tr><td>D3340=2</td><td></td><td>X</td><td></td><td></td></tr>
<tr><td>D3350=2</td><td></td><td></td><td></td><td>X</td></tr>
<tr><td>D3410=2</td><td>X</td><td>X*</td><td>X*</td><td></td></tr>
</table>
* Either one applies</td></tr>
<tr><td></td><td>SSLNS={4 / n}
(Model 115/125 only)</td><td>Specify the number of Start/Stop lines for the Model 115 or 125 CPU. n may be any value between 0 and 8 for Model 115 and 0 and 16 for Model 125.</td></tr>
<tr><td></td><td>BSCLNS={1 / n}
(Model 115/125)</td><td>Specify the number of BSC lines for the Model 115 or 125 CPU. n may be any value between 0 and 4 for Model 115 and 0 and 6 for Model 125.</td></tr>
</table>

Specify the Physical I/O units attached to the System

Operation	Operand	Explanation
DVCGEN (Note 1)	CHUN= X'cuu'	Hexadecimal number of channel and unit.

Operation	Operand	Explanation
DVCGEN (Note 1) (Cont'd)	DVCTYP= xxxxxx	Specify the Type of device.
	CHANSW= {NO / YES}	YES indicates that the device is attached to more than one selector channel (the device is switchable)
	MODE= {X'ss' / X'ssss' / X'ssssss'}	1) 2400T9; MODE is used to specify the tape mode. X'C0' is the default value. 2) 3410T9 or 3420T9; MODE specifies the tape mode. X'C0' is the default value. 3) 3420T9; MODE specifies the tape mode. X'D0' is the default value for single and dual density. 4) 2400T7 or 3420T7; MODE is used to specify the tape mode. X'90' is the default. 5) 2702; MODE designates the SADxx command. X'00' is the default value. X'00' SAD0, X'01' SAD1, X'02' SAD2, X'03' SAD3. 6) 2260 (local) and 3270 (local) MODE is used to specify the 1053, 3284 or 3286 printer when CHUN= X'cuu' refers to a 1053 attached to a 2848 or to either 3284 or a 3286 attached to a 3272. This operand must be entered as X'01'. 7) 1419, 1255, 1259, 1270, 1275 MODE designates the external interrupt bit associated with magnetic ink character reader. X'01' External line 7 X'02' External line 6 X'04' External line 5 X'08' External line 4 X'10' External line 3 X'20' External line 2 8) 1018 MODE specifies whether the Error Correction feature is present or not. X'00' is the default value. The operand must be entered as X'01' if the feature is present. 9) 2703 For a Model 115 and 125 with ICA, MODE=X'ss', MODE=X'ssss', or MODE=X'ssssss' specifies the line mode setting for a start/stop or BSC line. 10) 3705 MODE specifies the type of channel adapter. No default is assumed X'01' Type 1 or 4 channel adapter. X'02' Type 2 or 3 channel adapter.

Assign LUB's to PUB's as Standard System Assignments

Operation	Operand	Explanation
ASSGN (Note 2)	SYSnnn, X'cuu' [,bG][,H1] [Fn][,H2]	Symbolic unit is assigned a hexadecimal channel and unit number within a partition, and an input hopper for an IBM 2560 or 5425

Define the Page Data Set

Operation	Operand	Explanation
DPD	UNIT= X'cuu'	Specify channel and unit number of the disk device that is to contain the page data set.
	CYL = nnn	Specify the number of the cylinder where the page data set is to begin.
	VOLID= xxxxxx	Specify the volume serial number of the disk pack that holds the page data set.
SEND	(n)	End of supervisor macro instructions. n= beginning address of the problem program area.

Note 1: Rules for using DVCGEN

1 A separate DVCGEN macro instruction is required for each device.

2 The total number of DVCGEN macros must not exceed the total number of devices specified in the IODEV parameter of the IOTAB macro.

3 DVCGEN macros must be specified in ascending channel address sequence.

4 Switchable units (attached to more than one selector channel) must be defined once. They are defined on the lowest channel on which they are addressable.

5 The sequence of the DVCGEN cards determines the priority of the devices on their channel. Switchable units must be the last device for each channel, and must be on consecutive channels.

6 The specifications of these macros may be altered by IPL and DEL statements.

Note 2: Rules for using ASSGN

1 The ASSGN macro allows SYSRDR, SYSLST, SYSPCH and SYSIPT to be assigned to a tape or DASD. However, IPL unassigns any such assignments.

2 SYSLOG must also be assigned in BG, if assigned in foreground partition.

3 SYSLNK can only be assigned to a foreground partition if PCIL has been specified and NPARTS>1.

DEVICE TYPE CODES

Card Code	Actual IBM Device	Device-Type X'nn'	Device Type
2400T9	9-track Magnetic Tape Tape units	50	Magnetic Tape devices
2400T7	7-track Magnetic Tape units	50	
3410T9	9-track 3410 Magnetic Tape units	53	
3410T7	7-track 3410 Magnetic Tape units	53	
3420T9	9-track 3420 Magnetic Tape units	52	
3420T7	7-track 3420 Magnetic Tape units	52	
1442N1	1442N1 Card Read Punch	30	Card Read Punches
2520B1	2520B1 Card Read Punch	31	
2560	2560 Multifunction Card machine	33	
2596	2596 Card Read Punch	30	
3525RP	3525 Card Punch (with optional read feature)	32	
5425	5425 Multifunction card Unit	34	
2501	2501 Card Reader	10	Card Readers
2540R	2540 Card Reader	11	
3504	3504 Card Reader	12	
3505	3505 Card Reader	12	
2540P	2540 Card Punch	21	Card Punches
2520B2	2520B2 Card Punch	20	
1442N2	1442N2 Card Punch	22	
2520B3	2520B3 Card Punch	20	
3525P	3525 Card Punch	23	
PRT 1	3211 or 3203-4 Printer	43	Printers
1403	1403 Printer	40	
1403U	1403 Printer with UCS feature	42	
1443	1443 Printer	41	
2260(local)	1053 Printer with 2848 Control Unit. MODE operand must be entered as X'01'	C0	
3203	3203 Printer	4A	
3211	3211 or 3203-4 Printer	43	
3277 (local 3270)	3284 or 3286 Printer with 3272 Control Unit. MODE operand must be entered as X'01'	B0	
3277B (local 3270)	3284 or 3286 Printer with 3272 Control Unit, attached in burst mode to a multiplexer channel. MODE operand must be entered as X'01'	B0	
5203	5203 Printer	4C	
5203U	5203 Printer with UCS feature	4D	

DEVICE TYPE CODES (...Cont'd)

Card Code	Actual IBM device	Device Type X'nn'	Device Type
1050A	3210,3215 Console Printer Keyboards	00	Printer Keyboards
125D	Model 115/125 Integrated Display Operator Console	B2	Display Operator Consoles
125DP	Model 115/125 Integrated Display Operator Console with 5213 Console Printer attached Note: Codes are valid for native Alpha Numeric Replacement mode and for 1052 emulation. The 1052 mode is not supported with Indirect Data Addressing (IDA)	B2	
3277	3277 Operator Console	B0	
UNSP	Unsupported device	FF	Unsupported no burst mode on multiplexor channel
UNSPB	Unsupported device	FF	Unsupported with burst mode on multiplexor channel.
2311	2311 Disk Storage Device	60	DASD
2314	2314 Direct Access Storage Facility	62	
2314	2319 Disk Storage Facility	62	
2321	2321 Data Cell Drive	61	
3330	3330-1,3330-2 or 3333-1 Disk Storage	63	
3330	3350 Direct Access Storage in 3330 Compatibility mode	63	
3330B	3330 Disk Storage Model 11	65	
3340	3340 Disk Storage (General)	68	
3340R	3340 Disk Storage with RPS feature (with or without 3340 Data Module, Model 35)	69	
3340R	3340 Disk Storage with RPS feature (with or without 3340 Data Module, Model 70)	6A	
3340	3340 Disk Storage without RPS feature (with or without 3340 Data Module, Model 35)	69	
3340	3340 Disk Storage without ROS feature (with or without 3340 Data Module, Model 70)	6A	
3340R	3344 Direct Access Storage	6A	
3350	3350 Direct Access Storage	67	
1419	1255 Magnetic Character Reader	72	MICR-Magnetic Ink Character Recognition devices
1419	1259 Magnetic Character Reader	72	
1419	1419 Magnetic Character Reader	72	
1419P	1419 Dual Address Adapter Primary Control Unit	73	
1419S	1419 Dual Address Adapter Secondary Contr. Unit	74	
2955	2955 Data Adapter Unit	D7	Data Link for RETAIN

DEVICE TYPE CODES (....Cont'd)

Card Code	Actual IBM device	Device-Type X'nn'	Device Type
2701	2701/2715 Data Adapter unit	D0	Teleprocessing lines
2703	Integrated Communications Adapter (Model 135)	D2	
2702 A 2702 B 2702 C 2702 D	2702 Transmission Control unit	D1	A=SAD0 comm'd when enabling the line B=SAD1 comm'd C=SAD2 comm'd D=SAD3 comm'd
2703	2703 Transmission Control unit	D2	
2703	Integrated Communication Adapter (Mod. 115/125)	D2	
2703	3704/3705 Communication Controller in Emulation Mode	D2	
3704	3704 Communication Controller in Network Control Mode	DC	
3705	3705 Communication Controller in Network Control Mode	DC	
3791L	3791 Local Communication Controller	DE	
1419	1270 Optical Reader/ Sorter	72	
1419	1275 Optical Reader/ Sorter	72	
1419P	1275 Dual Address Adapter Primary Control Unit	73	
1419S	1275 Dual Address Adapter Secondary Control Unit	74	Optical Readers
1287	1287 Optical Reader	77	
1288	1288 Optical Page Reader	77	
3881	3881 Optical Mark Reader	11	
3886	3886 Optical Character Reader	7C	
3540	3540 Diskette Input/Output Unit	80	Diskette
2260	2260 Display Station	C0	
3277 (local 3270)	3277 Display Station; MODE operand need not be entered		
3277B (local 3270)	3277 Display Station; attached in burst mode to a multiplexor channel. MODE operand need not be entered	B0	Display Stations
7770	7770 Audio Response Unit	D3	Audio Response Unit
1017	1017 Paper Tape Reader with 2826 Control Unit	78	Paper Tape Readers
2671	2671 Paper Tape Reader	70	
1018	1018 Paper Tape Punch with 2826 Control Unit	79	Paper Tape Punch
2495TC	2495 Tape Cartridge Reader	51	Tape Cartridge Reader

FORMAT OF THE ESD CARD

Card Columns	
1	Multiple punch (12-2-9). Identifies this as a loader card.
2- 4	ESD--External Symbol Dictionary card.
11-12	Number of bytes of information contained in this card.
15-16	External symbol identification number (ESID) of the first SD,PC, CM or ER on this card. Relates the SD,PC,CM or ER to a particular control section.
17-72	Variable information. 8 positions - Name 1 position - Type code hex '00','01','02','04','05', or '0A' to indicate SD,LD,ER,PC,CM, or WX respectively. 3 positions - Assembled origin 1 position - Blank 3 positions - Lenghth, if an SD type, CM type, or a PC type. If an LD type, this field contains the external symbol identification number (ESID) of the SD containing the label.
73-80	May be used by the programmer for identification.

FORMAT OF THE TXT CARD

Card Columns	
1	Multiple punch (12-2-9). Identifies this as a loader card.
2- 4	TXT --Text card.
6- 8	Assembled origin (Address of first byte to be loaded from this card).
11-12	Number of bytes of text to be loaded.
15-16	External symbol identification number (ESID) of the control section (SD or PC) containing the text.
17-72	Up to 56 bytes of text -- data or instructions to be loaded.
73-80	May be used for program identification.

FORMAT OF THE RLD CARD

Card Columns	
1	Multiple punch (12-2-9). Identifies this as a loader card.
2- 4	RLD -- Relocation List Dictionary Card.
11-12	Number of bytes of information contained in this card.
17-72	Variable information (multiple items). a. Two positions - (relocation identifier) pointer to the ESID number of the ESD item on which the relocation factor of the contents of the address constant is dependent. b. Two positions

FORMAT OF THE RLD CARD (....Cont'd)

b. Two positions - (position identifier) pointer to the ESID number of the ESD item on which the position of the address constant is dependent.

c. One position - flag indicating type of constant, as follows:

Bits	
0-2	Ignored
3	0 - a non branch type load constant 1 - a branch type load constant
4-5	00 - load constant length= 1 byte 01 - load constant length= 2 bytes 10 - load constant length= 3 bytes 11 - load constant length= 4 bytes
6	0 - relocation factor is to be added 1 - relocation factor is to be subtracted
7	0 - Next load constant has different R and P identifiers; therefore, both R and P must be present. 1 - Next load constant has the same R and P identifiers; therefore they are both omitted.

Five significant bits of this byte are expanded in the RSERV printout.

d. Three positions - assembled origin of load constant.

73-80 May be used for program identification.

FORMAT OF THE END CARD

Card Columns	
1	Multiple punch (12-2-9). Identifies this as a loader card.
2 -4	END
6 -8	Assembled origin of the label supplied to the Assembler in the END card (optional).
15-16	ESID number of the control section to which this END card refers (only if 6-8 present).
17-22	Symbolic label supplied to the Assembler if this label was not defined within the assembly.
29-32	Control section length (if not specified in the last SD or PC).
73-80	Not used.

FORMAT OF THE REP (User Replace) CARD

Card Columns	
1	Multiple punch (12-2-9) Identifies this as a loader card.
2 -4	REP - Replace text card.
5 -6	Blank.
7 -12	Assembled address of the first byte to be replaced (hexadecimal). Must be right justified with leading zero's if needed to fill the field.
13	Blank

FORMAT OF THE REP (User Replace) CARD (.....Continued)

Card Columns	
14-16	External symbol identification number (ESID) of the control section (SD) containing the text (hexadecimal). Must be right justified with leading zero's if needed to fill the field.
17-70	From 1-11 4-digit hexadecimal fields seperated by comma's, each replacing two bytes. A blank indicates the end of information in this card.
71-72	Blank
73-80	May be used for program identification.

DEBLOCK UTILITY

Description:

Purposes :

- To block an 80/81-byte record file to a 3440-byte record file.
- To deblock a blocked 3340-byte file in order to create an 80-byte SYSIN file.
- To copy files.
- To print (list) job control statements and comments from a blocked input file.
- To select records (or a group of records) from a blocked 3440-byte file in order to create an 80-byte SYSIN file.

Functions :

The program is only meant to support IBM distribution files that contain only 3440-byte blocked records. Eighty-byte deblocked records as output and 80 and/or 81-byte records as input will be processed.

The devices used for input and output are defined by assigning the input device to SYS004 and the output device to SYS005. For the list function the output device is SYSLST.

Block :	To block an 80 or 81-byte record file to a 3440-byte record file.
Deblock :	To deblock the blocked 3440-byte file in order to create an 80-byte SYSIN file.
Copy :	The card-to-card copy function includes 80-column to 96-column conversion for the IBM 5425 Multi-Function Card Unit.
List :	To determine the contents of a file with blocked 3440-byte records.
Select :	To deblock selected PTFs from a blocked PTF file. The function can be used for any other 3440-byte blocked sequential file.

Supported devices :

- IBM 2501 Card Reader
- IBM 2540 Card Read Punch
- IBM 2560 Multi-Function Card Machine
- IBM 3504 Card Reader
- IBM 3505 Card Reader
- IBM 3525 Card Punch
- IBM 5425 Multi-Function Card Unit
- IBM 2400/3400 Series Magnetic Tape Unit *
- IBM 2311 Disk Storage Drive
- IBM 2314 Direct Access Storage Facility
- IBM 2319 Disk Storage
- IBM 3330 Disk Storage
- IBM 3333 Disk Storage
- IBM 3340 Direct Access Storage Facility

* The IBM 2495 Tape Cartridge Reader does not belong to the IBM 2400 Series Magnetic Tape Unit.

When a disk is assigned, a //DLBL and //EXTENT card are required. The file names that are used for the DLBL card are :

```
// DLBL UIN, 'file ID'          (for input)
// DLBL UOUT, 'file ID'         (for output)
```

Tape labels and the UPSI byte are not supported, except for deblocked output tapes.

To create a deblocked tape, a //TLBL card or a //UPSI card are required. The entries that are used for the TLBL and UPSI cards are :

```
// TLBL UOUT, 'file ID'
// UPSI 00100000
```

NO REWIND is always assumed for input/output tapes. Be sure that the tapes are correctly positioned.

Input/Output

Following is a table showing the input/output devices for the block, deblock, select and list functions :

		Block		Deblock/Select		List
		Input	Output	Input	Output	Input
	Record Format	80/81 bytes unblocked	80 bytes 3440 blocked	80 bytes 3440 blocked	80 bytes unblocked	80 bytes 3440 blocked
Devices	Card	yes	no	no	yes	no
	Tape	yes	yes	yes	yes	yes
	Disk	no	yes	yes	yes	yes

Note : SYSLST is the output device for the list function.

Following is a table showing the input/output devices for the copy function:

		Copy			
		Input	Output	Input	Output
	Record Format	80/81 bytes unblocked	80 bytes unblocked	80 bytes 3440 blocked	80 bytes 3440 blocked
Devices	Card	yes	yes	no	no
	Tape	yes	yes	yes	yes
	Disk	no	yes	yes	yes

Note : When a card device is assigned to SYS004 or SYS005, the program supports unblocked files ; otherwise, blocked files are assumed. Be sure you mount a tape or disk with records of the required length.

Input/Output (...Continued)

When the block function is used, you are not allowed to define a disk storage as an input device or a card punch as an output device. When the deblock function is used, you are not allowed to define a card reader as an input device.

Utility Modifier Statement

Contains information to run the program.

The format and entries are :

// UDS ffff

// U	Identifies the utility modifier statement.
DS	Indicates the Deblock program.
	Can be omitted
ffff	Indicates the function specification. Can be omitted. The default is DBL.
ffff=BLK6	Block function.
ffff=COP6	Copy function.
ffff=DBL6	Deblock function.
ffff=LST6	List function.
ffff=SEL,	Select function.

The following parameters are only required for the select function (see Control Statement Stream example 6).

n	Indicates the start position (column number) of the fixed part of the select identifier. One or two numerics, ranging from 1 to 80, are required.
'ii...ii'	Indicates the fixed part of the select identifier. All characters are allowed.
m1	Indicates the start position (column number) of the variable part of the select identifier. One or two numerics, ranging from 1 to 80, are required.
m2	Indicates the end position (column number) of the variable part of the select identifier. One or two numerics, ranging from 1 to 80, are required.

Note : It is not allowed to use apostrophes in a comment in the utility modifier statement for the select function.

The select identifier consists of two parts, a fixed part that is not changed during the select operation, and a variable part that is changed for each select by reading the next selector card.

The variable part read in from the selector card, is moved to the select identifier positions m1 and m2.

Utility Modifier Statement (...Continued)

The selector cards are placed behind the END card.

If the fixed and variable parts overlap each other, the variable part overwrites the fixed part.

The selected stream of records starts with the record that has the required characters in the positions specified by the select identifier. The select operation ends by reading a /& card from SYS004.

The identifiers are searched in the order in which they are read in. Therefore, the sequence of the selector cards and the way in which they appear on the input file, should be identical.

Default values

In the following cases DBL is the default :

- //UDS
- //U
- not specified.

End Statement

This is the last control statement and cannot be omitted.

The format is :

// END

Control Statement Stream

Six examples of control statement streams to run the program from the core image library are given.

1. Blocking from card to tape :

```
// JOB BLOCK CARD TO TAPE
// ASSGN SYS004,X'00C'
// ASSGN SYS005,X'282'
// EXEC DSTRB
// END
//  .
    .
    .
  data cards
    .
    .
    .
/*EOD          (no embedded blanks)
/&
```

Control Statement Stream (...Continued)

2. Deblocking from tape to disk.

```
// JOB CREATE SYSINFILE
// ASSGN SYS004,X'283'
// ASSGN SYS005,X132'
// DLBL UOUT,'file ID',99/365
// EXTENT SYS005,,,,20,780
// EXEC DSTRB
// UDS DBL
// END
/&
```

3. Deblocking from tape to tape.

```
// JOB DEBLOCK TAPE TO TAPE
// ASSGN SYS004,X'180'
// ASSGN SYS005,X'181'
// TLBL UOUT,'file ID'
// EXEC DSTRB
// UDS DBL
// END
/&
```

4. Copying card to tape.

```
// JOB COPY CARD TO TAPE
// ASSGN SYS004,X'00C'
// ASSGN SYS005,X'181'
// TLBL UOUT,'file ID'
// EXEC DSTRB
// UDS COP
// END
/&
```

5. Listing a blocked tape.

```
// JOB LIST BLOCKED TAPE
// ASSGN SYS004,X'181'
// EXEC DSTRB
// UDS LST
// END
/&
```

6. Selecting from tape.

```
// JOB SELECT FROM TAPE
// ASSGN SYS004,X'180'
// ASSGN SYS005,X'00C'
// EXEC DSTRB
// UDS SEL,1,'ii...ii',9,14
// END
123456
437298
/*
/&
```

Control Statement Stream (...Continued)

If the identifiers overlap, the second part overwrites the first part. The identifiers are searched in the order in which they are read in. Be sure that the cards are in the same sequence as the selected records on the input file.

Record Limits

During blocking, messages will be generated to inform you that the blocked file, when deblocked, will fit on 90% of a 2400-foot 7 or 9-track tape or on a 2311 disk. Processing continues. These limits (90%) are :

31,000 records (7-track tape)
40,000 records (9-track 1600 BPI tape)
45,000 records (2311 disk).

Be sure you mount an output tape or disk that can contain the whole file.

NAME	PROGRAM DESCRIPTION/FUNCTION
CSERV	(core image library service) to punch out (or write on magnetic tape, disk, or diskette) programs from the core image library.
SSERV	(source statement library service) to punch out (or write on magnetic tape, disk, or diskette) macro definitions.
RSERV	(relocatable library service) to punch out (or write on magnetic tape, disk, or diskette) the relocatable modules used to build IBM-supplied processor programs.
PSERV	(procedure library service) to display on SYSLST or to punch out (or write on magnetic tape, disk, or diskette) procedures from the procedure library.
DSERV	(directory service) to display on SYSLST the current contents of one or more library directories and their remaining library capacities. The directory display may be either an alphabetically sorted listing or a listing of the entries in the order they appear in the directory.
LSERV	(label cylinder display) to display (on SYSLST) the label cylinders located on SYSRES. SYSLST may be assigned to disk, tape, printer, or diskette.
ESERV	(de-editor program) to de-edit pre-edited macros from the E-sublibrary. It provides SYSLST and SYSPCH output of the original macros. It also provides an update facility for one macro per job step.
MAINT	(library maintenance) to delete and/or catalog library elements, and also to condense and reallocate library extents.
BACKUP	to create a backup of the system on tape and/or to create a backup of private libraries on tape. The tape is suitable as input for the RESTORE program.
RESTORE	to restore the DOS/VS Distribution Tape in a partition of a current release of DOS/VS. The program may be executed by statements from SYSIPT or SYSLOG.
COPYSERV	to build new libraries. The program automates the comparing of the current library directories with the new directories and provides input for the CORGZ program.
CORGZ	(copy or merge) to selectively copy or merge library entries from one disk pack to another disk pack. CORGZ allows larger or smaller allocations for each library of the new pack.

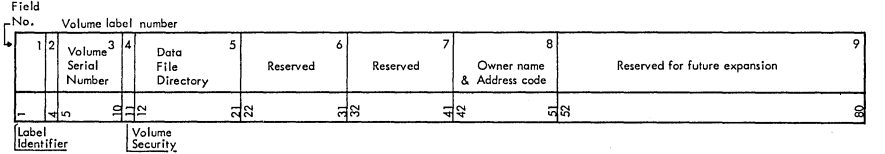
Note: BACKUP and RESTORE can be used efficiently to condense your libraries and also to migrate libraries from one DASD type to another.

For more details on these programs, refer to DOS/VS System Control Statements, GC33-5376.

CHAPTER III
DOS/VS IOCS (GENERAL/SAM/DAM/ISAM)

STANDARD VOLUME LABEL, TAPE OR DASD

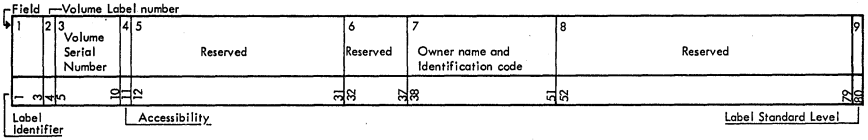
IBM Standard Volume Label Format (80 bytes) for EBCDIC Tape or DASD



FIELD	NAME AND LENGTH	DESCRIPTION
1	LABEL IDENTIFIER 3 bytes	Must contain VOL to indicate that this is a Volume Label.
2	VOLUME LABEL NR 1 byte	Indicates the relative position (1-8) of a volume label within a group of volume labels.
3	VOLUME SERIAL NR 6 bytes	A unique identification code which is assigned to a volume when it enters an installation. This code may also appear on the external surface of the volume for visual identification. It is normally a numeric field 000001 to 999999, however any or all of the 6 bytes may be alphameric.
4	VOLUME SECURITY 1 byte (OS/VS only)	Indicates security status of the volume: 0: no further identification for each file of the volume is required. 1: Further identification for each file of the volume is required before processing.
5	DATA FILE DIRECTORY 10 bytes	For DASD only. The first 5 bytes contain the starting address (CCHHR) of the VTOC. The last 5 bytes are blank. For tape files this field is not used and should be recorded as blanks.
6	RESERVED 10 bytes	Reserved
7	RESERVED 10 bytes	Reserved
8	OWNER NAME AND ADDRESS CODE 10 bytes	Indicates a specific customer, installation and/or system to which the volume belongs. This field may be a standardized code, name, address etc. (OS/VS only).
9	RESERVED 29 bytes	Reserved

Note: All reserved fields should contain blanks to facilitate their use in the future. Any information appearing in these fields at the present time will be ignored by the DOS/VS and OS/VS programs.

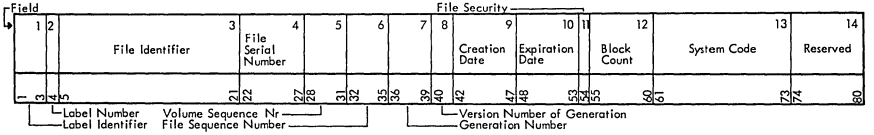
ANSI Standard Volume Label format and contents for ASCII tapes



FIELD	NAME AND LENGTH	DESCRIPTION
1	LABEL IDENTIFIER 3 bytes	Must contain VOL to indicate that this a Volume Label.
2	VOLUME LABEL NR 1 byte	Must be 1. If any other standard volume labels are present (indicated by an entry other than 1 in this field)they are ignored.
3	VOLUME SERIAL NR 6 bytes	Uniquely identifies this volume. Must consist of 6 bytes and may be any character exept a quote(').
4	ACCESSIBILITY 1 byte	Indicates accessibility protection: Space: No accessibility protection. Nonspace: Accessibility protection.
5	RESERVED 20 bytes	Reserved for future use as required by the American National Standards Institute, Inc. Should contain spaces.
6	RESERVED 6 bytes	Reserved for future use as required by American National Standards Institute, Inc. Should contain spaces.
7	OWNER NAME AND IDENTIFICATION CODE 14 bytes	Indicates a specific customer, installation and /or system to which the volume belongs. This field may be a standardized code, name, address etc.
8	RESERVED 28 bytes	Reserved for future use as required by the American National Standards Institute, Inc. Should contain spaces.
9	LABEL STANDARD LEVEL 1 byte	Indicates whether this volume observes the American National Standards: Dec. 1: Volume observes the standards (1 is also the default value) Space: Volume does not observe the standards, but it follows an agreed format.

STANDARD MAGNETIC TAPE FILE LABEL

IBM Standard Tape File Label Format and Contents



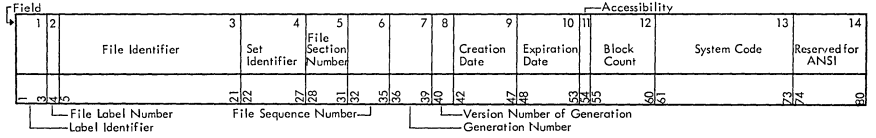
III-03

FIELD	NAME AND LENGTH	DESCRIPTION
1	LABEL IDENTIFIER 3 bytes EBCDIC	Identifies the type of label: HDR: Header--beginning of data file. EOF: End of File--end of a set of data. EOV: End of Volume--end of the physical reel.
2	FILE LABEL NUMBER 1 byte EBCDIC	Always a 1
3	FILE IDENTIFIER 17 bytes EBCDIC	Uniquely identifies the entire file, may contain only printable characters.
4	FILE SERIAL NUMBER 6 bytes EBCDIC	Uniquely identifies a file-volume relationship. This field is identical to the Volume Serial Number in the volume label on the first or only volume of a multi-volume file or a multi-file set. This field will normally be numeric(000001 to 999999) but may contain any six alphameric characters.
5	VOLUME SEQUENCE NUMBER 4 bytes	Indicates the order of a volume in a given file or multi-file set. This number must be numeric(0000-9999). Multiple volumes of an output file will be numbered in consecutive sequence.
6	FILE SEQUENCE NUMBER 4 bytes	Assign numeric sequence to a file within a multi file set.
7	GENERATION NUMBER 4 bytes	Numerically identifies the various editions of the file.
8	VERSION NUMBER OF GENERATION 2 bytes	Indicates the version of the generation of a file.

FIELD	NAME AND LENGTH	DESCRIPTION
9	CREATION DATE 6 bytes	Indicates the year and the day of the year that the file was created: Position / Code / Meaning 1 / blank / none 2-3 / 00-99 / year 4-6 / 001-366 / day of year (e.g., January 31, 1965 would be entered as 65031)
10	EXPIRATION DATE 6 bytes	Indicates the year and the day of the year when the file may become a scratch tape. The format of this field is identical to field 9. On a multi-file reel, processed sequentially, all files are considered to expire on the same day.
11	FILE SECURITY 1 byte	Indicates the security status of the file. 0: No security protection. 1: Security protection. Additional identification of the file is required before it can be processed.
12	BLOCK COUNT 6 bytes	Indicates the number of data blocks written on the file from the last header label to the first trailer label, exclusive of tape marks. Count does not include checkpoint records. This field is used in trailer labels.
13	SYSTEM CODE 13 bytes	Uniquely identifies the programming system. (For DOS/VS: IBMDOSVS)
14	RESERVED 7 bytes	Reserved

STANDARD MAGNETIC TAPE FILE LABEL

ANSI Standard Tape File Label Format and Contents

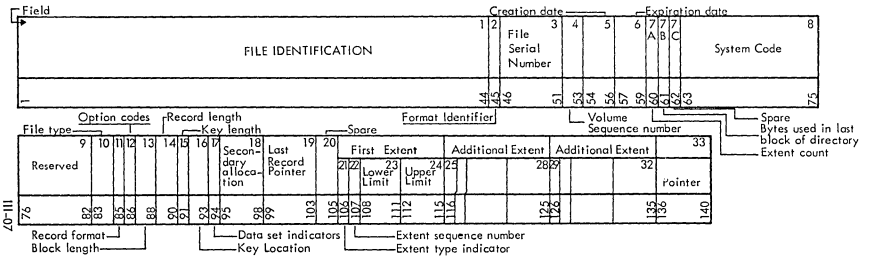


FIELD	NAME AND LENGTH	DESCRIPTION
1	LABEL IDENTIFIER 3 bytes, ASCII	Identifies the type of label: HDR: Header--beginning of a data file. EOF: End of File--end of a set of data. EOV: End of Volume--end of the physical reel.
2	FILE LABEL NUMBER 1 byte, ASCII	Indicates the sequence of this label within a label group (HDR, EOF, EOV). DOS/VS supports File Label 1 only and ignores subsequent numbers.
3	FILE IDENTIFIER 17 bytes, ASCII	Identifies the entire file. May be any character exept a quote(').
4	SET IDENTIFIER 6 bytes, ASCII	Identifies the volume-file relationship. Generally, this field is identical to the volume serial number from the VOL label or the first or only volume of the logical file.
5	FILE SECTION NUMBER 4 bytes	Indicates the order of a volume in a given file or multi-file set. (The first file must be numbered 0001).
6	FILE SEQUENCE NUMBER 4 bytes	Assigns numeric sequence to a file within a multi-file set. (The first file must be numbered 0001).
7	GENERATION NUMBER 4 bytes	Numerically identifies this edition of the file. (Must be numerical or blank).
8	VERSION NUMBER OF GENERATION 2 bytes	Indicates this version of the generation in field 7. (Must be numerical or blank).
9	CREATION DATE 6 bytes	Indicates the year and the day of the year that this file was created (byyddd), where: b= blank yy= year (00-99) ddd= day (001-366)

FIELD	NAME AND LENGTH	DESCRIPTION
10	EXPIRATION DATE 6 bytes	Indicates the year and the day of the year that this file may become a scratch tape. Same format as above (Field 9).
11	ACCESSIBILITY 1 byte	Indicates the accessibility protection of the file. Space: no accessibility protection. Nonspace: accessibility protection.
12	BLOCK COUNT 6 bytes	Indicates the number of data blocks (physical records) written for this logical file.
13	SYSTEM CODE 13 bytes	Name of programming system. (For DOS/VS: IBMDOSVS$$$$$)
14	RESERVED 7 bytes	Reserved for future use as required by ANSI. (American National Standards Institute, Inc.). Should be recorded as spaces.

FIELD	NAME AND LENGTH	DESCRIPTION

STANDARD DASD FILE LABEL , FORMAT 1 (Format 1: This format is common to all data files on Direct Access Storage Devices)



FIELD	NAME AND LENGTH	DESCRIPTION
1	FILE NAME 44 bytes, alphameric EBCDIC	This field serves as the key portion of the file label. Each file must have a unique file name. Duplication of file names will cause retrieval errors. The file name can consist of three sections: 1 File ID is an alphameric name assigned by the user and identifies the file. Can be 1-35 bytes if generation and version numbers are used, or 1-44 bytes if they are not used. 2 Generation number. If used, this field is separated from File ID by a period. It has the format Gnnnn, where G identifies the field as the generation number and nnnn (in decimal) identifies the generation of the file. 3 Version Number of Generation. If used, this section immediately follows the

STANDARD DASD FILE LABEL , FORMAT 1 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
1	FILENAME (Cont'd)	generation number and has the format Vnn, where V identifies the field as the version of generation number and nnn (in decimal) identifies the version of generation of the file. Note: DOS/VS compares the entire field against the filename given in the DLBL card. The generation and version numbers are treated differently by OS/VS.

The remaining fields comprise the DATA portion of the file label:

2	FORMAT IDENTIFIER 1 byte, EBCDIC numeric	1 = Format 1
3	FILE SERIAL NR 6 bytes, EBCDIC alphameric	Uniquely identifies a file/volume relationship. It is identical to the Volume Serial Number of the first or only volume of a multivolume file.
4	VOLUME SEQUENCE NR, 2 bytes, binary	Indicates the order of a volume relative to the first volume on which the data file resides.
5	CREATION DATE 3 bytes, discontinuous binary	Indicates the year and the day of the year the file was created. It is of the form YDD, where Y signifies the year (0-99) and DD the day of the year (1-366).
6	EXPIRATION DATE 3 bytes, discontinous binary	Indicates the year and the day of the year the file may be deleted. The form of this field is the same as that of field 5.

FIELD	NAME AND LENGTH	DESCRIPTION
7A	EXTENT COUNT	Contains a count of the number of extents for this file on this volume. If user labels are used, the count does not include the user label track. This field is maintained by the DOS/VS programs.
7B	BYTES USED IN LAST BLOCK OF DIRECTORY 1 byte, binary	Used by OS/VS
7C	SPARE 1 byte	Reserved
8	SYSTEM CODE 13 bytes	Uniquely identifies the programming system. The character codes that can be used in this field are limited to EBCDIC characters. On input, IOCS ignores this field. On output, IOCS inserts IBMDOSVS$\not\not\not\not\not$.
9	RESERVED 7 bytes	Reserved
10	FILE TYPE 2 bytes	The contents of this field uniquely identify the type of data file: Hex 4000: Consecutive organization Hex 2000: Direct access organization Hex 8000: Indexed sequential organization Hex 0200: Library organization Hex 0000: Organization not defined in the file label Hex 0008: VSAM

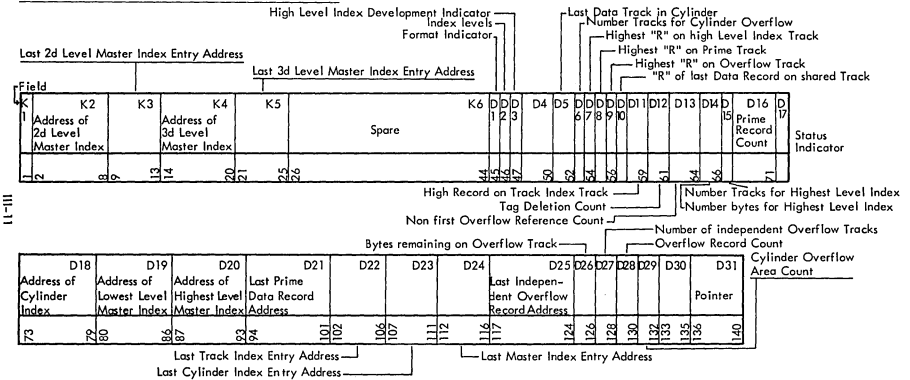
STANDARD DASD FILE LABEL , FORMAT 1 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
11	RECORD FORMAT 1 byte	Used by OS/VS
12	OPTION CODES	Bits within this field indicate various options used in building the file: Bit 0: 0 Bit 1: Reserved Bit 2: Master index present (ISAM) Bit 3: Independent overflow present (ISAM) Bit 4: Cylinder overflow present (ISAM) Bit 5: Reserved Bit 6: Delete record (OS/VS) Bit 7: Reorganize (OS/VS)
13	BLOCK LENGTH 2 bytes, binary	Indicates the block length for fixed length records or maximum block size for variable length blocks.
14	RECORD LENGTH 2 bytes, binary	Indicates the record length for fixed length records or the maximum record length for variable length records.
15	KEY LENGTH 1 byte, binary	Indicates the length of the key portion of of the data records in the file.
16	KEY LOCATION 2 bytes, binary	Indicates the high order position of the data records
17	DATA SET INDICATORS 1 byte	Bits within this field are used to indicate the following: Bit 0: If on, indicates that this is the last volume on which this file normally resides. Bit 1,2,4,6,7: 0 for DOS/VS-Used by OS/VS. Bit 3: If on, data set security is invoked. Bit 5: Used by DOS/VS and OS/VS.
18	SECONDARY ALLOCATION 4 bytes, binary	Used by OS/VS
19	LAST RECORD POINTER 5 bytes, discontinuous binary	Used by OS/VS
20	SPARE 2 bytes	Reserved
21	EXTENT TYPE INDICATOR 1 byte	Indicates the type of extent with which the following fields are associated: HEX CODE 00: Next three fields do not indicate any extent. 01: Prime data area(Indexed sequential), or Consecutive area, etc., (i.e., the extent containing the user's data records) 02: Overflow area of an indexed sequential file. 04: Cylinder index or master index area of an indexed sequential file. 40: User label track area. 80: Shared cylinder indicator.

STANDARD DASD FILE LABEL , FORMAT 1 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
22	EXTENT SEQUENCE NR 1 byte, binary	Indicates the extent sequence in a multi-extent file.
23	LOWER LIMIT 4 bytes, discontinuous binary	The cylinder and the track address specifying the starting point (lower limit) of this extent component. This field has the format CCHH.
25-28	ADDITIONAL EXTENT 10 bytes	These fields have the same format as the fields 21-24 above.
29-32	ADDITIONAL EXTENT 10 bytes	These fields have the same format as the fields 21-24 above.
33	POINTER TO NEXT FILE LABEL WITHIN THIS LABEL SET 5 bytes, discontinuous binary	The address (format CCHHR) of a continuation label if needed to further describe the file. If field 10 indicates Indexed Sequential organization, this field points to a Format 2 file label within this label set. Otherwise, it points to a Format 3 file label, and then only if the file contains more than three extent segments. This field contains all binary zeros if no additional file label is pointed to.

STANDARD DASD FILE LABEL , FORMAT 2



SEE NEXT PAGE FOR FURTHER EXPLANATION

STANDARD DASD FILE LABEL , FORMAT 2 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
K1	KEY IDENTIFICATION 1 byte	This byte contains the hex code 02 in order to avoid conflict with a file name.
K2	ADDRESS OF 2d LEVEL MASTER INDEX 7 bytes, discontinuous binary	This field contains the address of the first track of the second level of the master index, in the form MBBCCHH. (OS/VS only)
K3	LAST 2d LEVEL MASTER INDEX ENTRY 5 bytes, discontinuous binary	This field contains the address of the last index entry in the second level of the master index, of the form CCHHR. (OS/VS only)
K4	ADDRESS OF 3d LEVEL MASTER INDEX 7 bytes, discontinuous binary	This field contains the address of the first track of the third level of the master index, in the form MBBCCHH.
K5	LAST 3d LEVEL MASTER INDEX ENTRY 5 bytes, discontinuous binary	This field contains the address of the last entry in the third level of the master index, in the form CCHHR. (OS/VS only)
K6	SPARE 19 bytes	Reserved
D1	FORMAT IDENTIFIER 1 byte, EBCDIC numeric	2: Format 2
D2	NUMBER OF INDEX LEVELS 1 byte, binary	The contents of this field indicate how many levels of index are present with an Indexed Sequential File.
D3	HIGH LEVEL INDEX DEVELOPMENT INDICATOR 1 byte, binary	This field contains the number of tracks determining development of Master Index. (OS/VS only)
D4	FIRST DATA RECORD IN CYLINDER 3 bytes	This field contains the address of the last data track on each cylinder in the form HHR.
D5	LAST DATA TRACK IN CYLINDERS 2 bytes	This field contains the address of the last data track on each cylinder, in the form HH.
D6	NUMBER OF TRACKS FOR CYLINDER OVERFLOW 1 byte, binary	This field contains the number of tracks in cylinder overflow area. (OS/VS only)
D7	HIGHEST "R" ON HIGH LEVEL INDEX TRACK 1 byte	This field contains the highest possible R on track containing high-level index entries.
D8	HIGHEST "R" ON PRIME TRACK 1 byte	This field contains the highest possible R on prime data tracks for form F records.
D9	HIGHEST "R" ON OVERFLOW TRACK 1 byte	This field contains the highest possible R on overflow data tracks for form F records.

STANDARD DASD FILE LABEL , FORMAT 2 (....Cont'd)

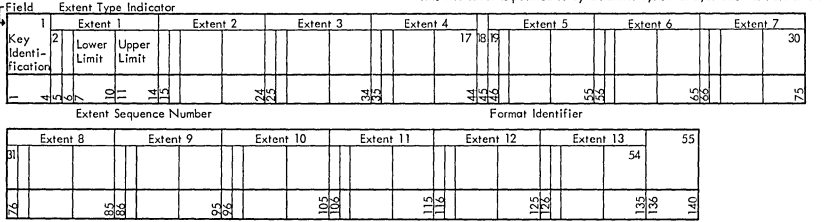
FIELD	NAME AND LENGTH	DESCRIPTION
D10	"R" OF LAST DATA RECORD ON SHARED TRACK - 1 byte	This field contains the R of the last data record on a shared track.
D11	HIGH RECORD ON TRACK INDEX TRACK 2 bytes	The first byte of this 2-byte field indicates the high(0-256) record on the track index track. The second byte is reserved.
D12	TAG DELETION COUNT 2 bytes, binary	This field contains the number of records that have been tagged for deletion.
D13	NONFIRST OVERFLOW REFERENCE COUNT 3 bytes, binary	This field contains a count of the number of random references to a nonfirst overflow record.
D14	NUMBER OF BYTES FOR HIGHEST LEVEL INDEX-2 bytes binary	The contents of this field indicate how many bytes are needed to hold the highest level index in main storage.
D15	NUMBER OF TRACKS FOR HIGHEST LEVEL INDEX 1 byte, binary	This field contains a count of the number of tracks occupied by the highest level index.
D16	PRIME RECORD COUNT 4 bytes, binary	This field contains a count of the number of records in the prime data area.
D17	STATUS INDICATOR 1 byte	The eight bits of this byte are used for the following indications: Bit 0-1 : must remain off Bit 2 : file closed for ADD or ADDRTR Bit 3-5 : must remain off Bit 6 : last block full-Bit 7: last track full
D18	ADDRESS OF CYLINDER INDEX 7 bytes	This field contains the address of first track of the cylinder index, in the form MBBCCHH.
D19	ADDRESS OF LOWEST LEVEL MASTER INDEX 7 bytes	This field contains the address of the first track of the lowest-level index of the high level indexes, in the form MBBCCHH.
D20	ADDRESS OF HIGHEST LEVEL INDEX 7 bytes	This field contains the address of the first track of the highest level master index, in the form MBBCCHH.
D21	LAST PRIME DATA RECORD ADDRESS 8 bytes	This field contains the address of the last data record in the prime data area, in the form MBBCCHHR.
D22	LAST TRACK INDEX ENTRY ADDRESS 5 bytes	This field contains the address of the last normal entry in the track index on the last cylinder in the form CCHHR
D23	LAST CYLINDER INDEX ENTRY ADDRESS 5 bytes	This field contains the address of the last index entry in the cylinder index in the form CCHHR.
D24	LAST MASTER INDEX ENTRY ADDRESS 5 bytes	This field contains the address of the last index entry in the master index, in the form CCHHR.
D25	LAST INDEPENDENT OVERFLOW RECORD ADDRESS 8 bytes	This field contains the address of the last record written in the current independent overflow area, in the form MBBCCHHR.

STANDARD DASD FILE LABEL , FORMAT 2 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
D26	BYTES REMAINING ON OVERFLOW TRACK 2 bytes, binary	This field contains the number of bytes remaining on current independent overflow track. (OS/VS only)
D27	NUMBER OF INDEPEN-OVERFLOW TRACKS 2 bytes, binary	This field contains the number of tracks remaining in independent overflow area.
D28	OVERFLOW RECORD COUNT 2 bytes, binary	This field contains a count of the number of records in the overflow area.
D29	CYLINDER OVERFLOW AREA COUNT 2 bytes, binary	This field contains the number of cylinder overflow areas full.
D30	DUMMY TRACK INDEX ENTRY 3 bytes	This field contains the HHR portion of the dummy track index entry. (OS/VS only)
D31	POINTER TO FORMAT 3 FILE LABEL 5 bytes	This field contains the address(in the form CCHHR) of a Format 3 file label if more than 3 extent segments exist for the data file within this volume. Otherwise it contains binary zeros. (OS/VS only)

STANDARD DASD FILE LABEL , FORMAT 3

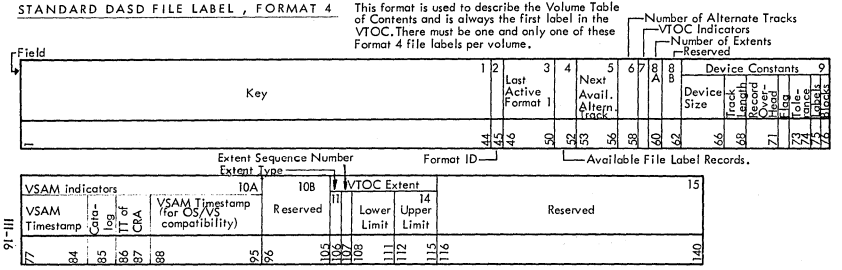
Format 3: This format is used to describe extra extent segments on the volume if there are more than can be described in the Format 1 (and Format 2 if it exists) file label. This file label is pointed to by a Format 1, Format 2, or another Format 3 file label.



FIELD	NAME AND LENGTH	DESCRIPTION
1	KEY IDENTIFICATION 4 bytes	Each byte of this field contains the Hex code 03 in order to avoid conflict with a data file name.
2-17	EXTENTS (in KEY) 40 bytes	Four groups of fields identical in format to fields 21-24 in the Format 1 label are contained here.
18	FORMAT IDENTIFIER 1 byte, EBCDIC numeric	3 : Format 3
19-54	ADDITIONAL EXTENTS 90 bytes	Nine groups of fields identical in format to fields 21-24 in the Format 1 label are contained here.
55	POINTER TO NEXT FILE LABEL 5 bytes	This field contains the address(in the form CCHHR) of another Format 3 label if additional extents must be described. Otherwise, it is all binary zeros.

STANDARD DASD FILE LABEL , FORMAT 4

This format is used to describe the Volume Table of Contents and is always the first label in the VTOC. There must be one and only one of these Format 4 file labels per volume.



FIELD	NAME AND LENGTH	DESCRIPTION
1	KEY FIELD 44 bytes, binary	Each byte of this field contains the Hex code 04 in order to provide a unique key.
2	FORMAT ID 1 byte, EBCDIC numeric	4 : Format 4.
3	LAST ACTIVE FORMAT 1 5 bytes	Contains the address (in the form CCHHR) of the last active Format 1 file label. It is used to stop a search on a file name. (OS/VS only)
3	NEXT AVAILABLE ALTERNATE TRACK	Contains the address (in the form CCHH) of the next track to be assigned as an alternate for a bad track.
5	4 bytes	
6	NUMBER OF ALTERNATE TRACKS 2 bytes, binary	Contains the number of alternate tracks available.
7	VTOC INDICATORS	Bit 0, if on indicates no DADSM (Format 5) label, or DADSM label does not reflect true status of volume. Bits 1,2 not used. Bit 3, if on indicates a stacked pack for the 1401/1440/1460 Emulator Program. Bits 4-7 are not used.

STANDARD DASD FILE LABEL , FORMAT 4 (....Cont'd)

FIELD	NAME AND LENGTH	DESCRIPTION
8A	NUMBER OF EXTENTS 1 byte	Contains the hexadecimal constant 01, to indicate one extent in the VTOC.
8B	RESERVED 2 bytes	Reserved
9	DEVICE CONSTANTS 14 bytes	This field contains constants describing the device on which the volume was mounted when the VTOC was created. The following describes each of the subfields:

Device Size: (4 bytes)-The number of cylinders(CC) and tracks per cylinder(HH).

Track Length: (2 bytes)-The number of available bytes on a track exclusive of home address and record zero. (record zero is assumed to be a nonkeyed record with an eight bytes data field)

Record Overhead: (3 bytes)-The number of bytes required for gaps, check bits, and count field for each record. This value varies according to the record characteristics and thus is broken down into three subfields.

- I: Overhead required for a keyed record other than the last record on the track.
- L: Overhead required for a keyed record that is the last record on the track.
- K: Overhead bytes to be subtracted from I or L if the record does not have a key field.

Flag: (1 byte)-Further defines unique characteristics of the device.

Bits: 0-5 : Reserved
6/7 : CC and HH must be used as 1-byte values, as in the case of the 2321. A tolerance factor must be applied to all but the last record on the track.

Tolerance: (2 bytes)-A value that is to be used to determine the effective length of a record is calculated in the following manner :
1 Add the key length to the data length of the record
2 Test bit 7 in the flag byte :
   a. If 0, go to step 3
   b. Multiply value from step 1 by the tolerance factor
   c. Shift result 9 bits to the right
3. Add overhead bytes to the result.
Note : Step 2 is not required if the calculation is for the last record on the track.

Labels/track: (1 byte)-A count of the number of labels that can be written on each track in the VTOC. (Number of full records of 44-bytes key and 96-bytes data lengths that can be contained on one track of this device)

Directory Blocks/Track: (1 byte)-A count of the number of directory blocks that can be written on each track for an OS/VS partitioned data set. (Number of full records of 8-bytes key and 256-bytes data lengths that can be contained on one track of this device)

STANDARD DASD FILE LABEL, FORMAT 4 (...Cont'd)

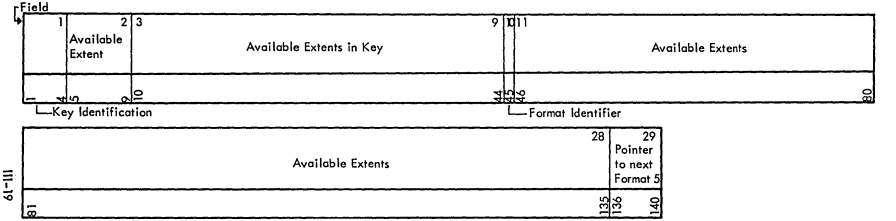
FIELD	NAME AND LENGTH	DESCRIPTION
9	DEVICE CONSTANTS (...Cont'd)	The following illustrates the device constants field for the various direct access devices:

Device	CC	HH	Track Length	I	L	K	Flag	Tol.	Labels Track
2311	203	10	3625	81	20	20	1	537	16
2314/2319	203	20	7294	146	45	45	1	534	25
2321	20/10	5/20	2000	100	16	16	3	537	8
2301	0	200	20616	186	186	53	0	512	63
2302	250	46	5070	82	55	20	1	537	22
3330	411	19	13030	191	191	56	0	512	39
3330-11	815	19	13030	191	191	56	0	512	39
3340/35MB	349	12	8368	242	242	75	0	512	22
3340/70MB	698	12	8368	242	242	75	0	512	22
3350	560	30	19069	267	267	82	8	512	46
7320	0	400	2129	111	43	14	1	537	8

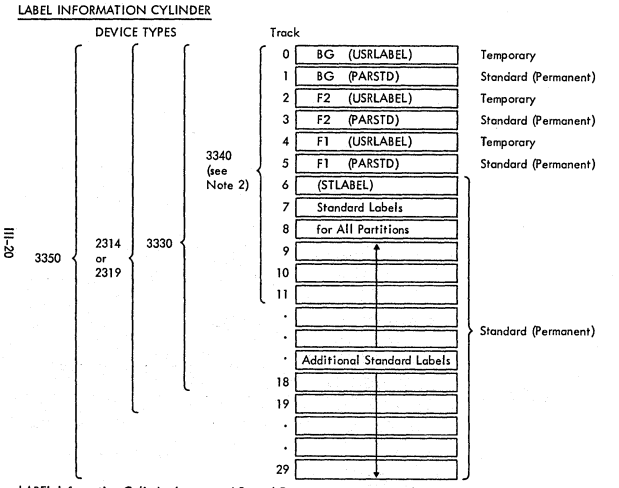
Note : CCHH for the 2321 above are separate 1 byte quantities.

FIELD	NAME AND LENGTH	DESCRIPTION
10A	VSAM INDICATORS (19 bytes)	The VSAM indicators are present on any volume that contains space allocated to VSAM. If a volume does not contain any VSAM-owned space, these fields are set to zeros.
	VSAM TIMESTAMP (8 bytes)	Is the time the most recent VSAM data space was added to the volume.
10A (Cont'd)	VSAM CATALOG (1 byte)	Bit0-1 : This volume is owned by a VSAM catalog. Bit 1-7: Unused
	TT of CRA (2 bytes)	Relative track number of first track of catalog recovery area
	VSAM TIMESTAMP (8 bytes)	
10B	RESERVED (10 bytes)	Reserved
11-14	VTOC EXTENT	These fields describe the extent of the VTOC and are identical in format to fields 21-24 of the Format 1 file label. Extent type 01 (prime date area).
15	RESERVED (25 bytes)	Reserved

STANDARD DASD FILE LABEL , FORMAT 5 (Note: Format 5 Label used by OS/VS only)



FIELD	NAME AND LENGTH	DESCRIPTION
1	KEY IDENTIFICATION 4 bytes	Each of these four bytes is an hex 05.
2	AVAILABLE EXTENT 5 bytes	This field indicates an extent of space available for allocation to a data file. The first two bytes are relative track address. The next two are the number of full cylinders included in the extent. The last byte is the number of tracks in addition to the cylinders in the extent.
3-9	AVAILABILITY EXTENTS IN KEY 35 bytes	These fields are identical to field 2. They are in relative track address sequence.
10	FORMAT IDENTIFIER 1 byte EBCDIC numeric	5: Format 5
11-28	AVAILABLE EXTENTS 90 bytes	These fields are the same as field 2. There are 26 available extents fields in the Format 5 label.
29	POINTER TO NEXT FORMAT 5	Contains the address (in the form CCHHR) of the next Format 5 file label if one exists.



LABEL Information Cylinder Layout and Record Format (see Note 1)

Note 1: The layout of the label information cylinder depends on the number of partitions defined in your system. This example assumes that three partitions are present. For more information see DOS/VS Serviceability Aids and Debugging Procedures.

Note 2: For the 3340, a second label information cylinder is provided containing 12 standard label tracks for all partitions.

LIOCS MODULE NAME VERSUS OPTIONS

character # ➡	1	2	3	4	5	6	7	8
CDMOD	I	J	C	F RECFORM=FIXUNB (always for TYPEFLE= INPUT, TYPEFLE= CMBND or FUNC= I files) V RECFORM= VARUNB U RECFORM= UNDEF	A CTLCHR= ASA(not specified if CMBND) Y CTLCHR= YES C CONTROL= YES Z CTLCHR or CONTROL not specified	B RDONLY= YES and TYPEFLE= CMBND C TYPEFLE= CMBND H RDONLY= YES and TYPEFLE= INPUT I TYPEFLE= INPUT N RDONLY= YES and TYPEFLE= OUTPUT O TYPEFLE= OUTPUT	Z WORKA and IOAREA2 not specified W WORKA= YES I IOAREA2= YES B WORKA and IOAREA2 Z WORKA= YES not specified (CMBND file only)	0 DEVICE=2540, 3881 1 DEVICE= 1442, 2596 2 DEVICE=2520 3 DEVICE=2501 4 DEVICE=2540 and CRDERR 5 DEVICE=2520 and CRDERR 6 DEVICE=3505 or 3504 7 DEVICE=3525 and FUNC=R/P or omitted 8 DEVICE=2560 and FUNC=R/P or omitted A DEVICE=3525 and FUNC=RP B DEVICE=3525 and FUNC=RW C DEVICE=3525 and FUNC=PW D DEVICE=3525 and FUNC=I E DEVICE=3525 and FUNC=RPW F DEVICE=2560 and FUNC=RP

LIOCS MODULE NAME VERSUS OPTIONS (....Cont'd)

character # ➡	1	2	3	4	5	6	7	8
CDMOD (Cont'd)								G DEVICE=2560 and FUNC=RW H DEVICE=2560 and FUNC=PW I DEVICE=2560 and FUNC=I J DEVICE=2560 and FUNC=RPW K DEVICE=5425 and FUNC=RP L DEVICE=5425 and FUNC=RW M DEVICE=5425 and FUNC=PW N DEVICE=5425 and FUNC=I O DEVICE=5425 and FUNC=RPW
DIMOD	I	J	J	F always	C non-RPS version of module V RPS data set	B TYPEFLE=OUTPUT (processes both in put and output) I TYPEFLE=INPUT	I IOAREA2=YES Z IOAREA2=YES is not specified	C RDONLY=YES D RDONLY=YES is not specified
DRMOD	I	J	M	Z	S SETDEV=YES Z SETDEV=YES not specified	R RDONLY=YES Z RDONLY=YES not specified	D always	0 always
DUMOD	I	J	N	D	I =DUMODFI O =DUMODFO	C ERROPT=YES and ERREXT=YES E ERROPT=YES Z neither is specified	Z	Y RDONLY=YES Z RDONLY not specified

LIOCS MODULE NAME VERSUS OPTIONS (....Cont'd)

character # ➡	1	2	3	4	5	6	7	8
MRMOD	I	J	U	S Address=SINGLE D Address=DUAL	Z	Z	Z	Z
MTMOD (GET/PUT)	I	J	F	F RECFORM=FIXUNB (or FIXBLK) (EBCDIC mode) I RECFORM=FIXUNB (or FIXBLK) (ASCII mode) V RECFORM=VARUNB (or VARBLK) (EBCDIC mode) R RECFORM=VARUNB (or VARBLK) (ASCII mode) S RECFORM=SPNUNB (or SPNBLK) (spanned records) U RECFORM=UNDEF (EBCDIC code) N RECFORM=UNDEF (ASCII mode)	B READ=BACK Z READ=FORWARD or if READ is not specified	C CKPTREC=YES Z CKPTREC=YES is not specified	W WORKA=YES Z WORKA=YES is not specified	E ERREXT=YES and RDONLY=YES N ERREXT=YES Y RDONLY=YES Z ERREXT and RDONLY not specified
MTMOD (WORKFILE)	I	J	F	W always	E ERROPT=YES Z ERROPT is not specified	N NOTEPNT=YES S NOTEPNT=POINTS Z NOTEPNT is not specified	Z always	M ERREXT=YES and RDONLY=YES N ERREXT=YES Y RDONLY=YES Z ERREXT and RDONLY not specified

LIOCS MODULE NAME VERSUS OPTIONS (....Cont'd)

character # ➡	1	2	3	4	5	6	7	8
ORMOD	I	J	M	F RECFORM=FIXUNB X RECFORM=FIXBLK U RECFORM=UNDEF D RECFORM=UNDEF and BLKFAC=YES	C CONTROL=YES Z CONTROL=YES is not specified	I IOAREA2=YES W WORKA=YES B both are specified Z neither is specified	T device is in tape mode D device is in document mode	Z always
PRMOD	I	J	D	F RECFORM=FIXUNB V RECFORM=VARUNB U RECFORM=UNDEF	A CTLCHR=ASA Y CTLCHR=YES C CONTROL=YES S STLIST=YES Z none of these is specified T DEVICE=3525 with 2-line printer U DEVICE=2560 V DEVICE=5425	B ERROPT=YES and PRINTOV=YES P PRINTOV=YES, DEVICE is not 3525 and ERROPT is not specified I PRINTOV=YES, DEVICE=3525 and FUNC=W[T] or omitted F PRINTOV=YES, DEVICE=3525 and FUNC=RW[T] C PRINTOV=YES, DEVICE=3525 and FUNC=PW[T] D PRINTOV=YES, DEVICE=3525 and FUNC=RPW[T] Z PRINTOV=YES and ERROPT not specified and DEVICE is not 2560, 3525, or 5425	I IOAREA2=YES Z IOAREA2=YES is not specified	V RDONLY=YES and WORKA=YES W WORKA=YES Y RDONLY=YES Z neither is specified

character # ➡	1	2	3	4	5	6	7	8
PRMOD (Cont'd)						O PRINTOV=YES not specified, DEVICE= 3525 and FUNCT= W[T] or omitted R PRINTOV=YES not specified, DEVICE= 3525 and FUNCT= RW[T] S PRINTOV=YES not specified, DEVICE= 3525 and FUNCT= PW[T] T PRINTOV=YES not specified, DEVICE= 3525 and FUNCT= RPW[T] E ERROPT=YES and PRINTOV=YES is not specified U FUNC=W or omitted and DEVICE=2560 or 5425 V FUNC=RW and DEVICE=2560 or 5425 W FUNC=PW and DEVICE=2560 or 5425 X FUNC=RPW and DEVICE=2560 or 5425		

LIOCS MODULE NAME VERSUS OPTIONS (....Cont'd)

character # →	1	2	3	4	5	6	7	8
PTMOD	I	J	E	S SCAN=YES Z SCAN=YES is not specified	T TRANS=YES (SCAN)=YES is not specified) Z TRANS=YES is not specified	F RECFORM=FIXUNB and SCAN=YES U RECFORM=UNDEF and SCAN=YES Z SCAN=YES is not specified and/or DEVICE=1018	1 DEVICE=1017 2 DEVICE=1018 Z DEVICE=2671 or if this entry is omitted	Z always
SDMOD (GET/PUT)	I	J	G	C SDMODFx specifies HOLD=YES F SDMODFx does not specify HOLD=YES R SDMODUx specifies HOLD=YES U SDMODUx does not specify HOLD=YES P SDMODVx specifies HOLD=YES and RECFORM=SPNBLK/SPNUNB Q SDMODVx does not specify HOLD=YES and specifies RECFORM=SPNBLK/SPNUNB S SDMODVx specifies HOLD=YES V SDMODVx does not specify HOLD=YES	I SDMODxI (non-RPS version of module) O SDMODxO (non-RPS version of module) U SDMODxU (non-RPS version of module) W SDMODxI (RPS input data set) X SDMODxO (RPS output data set) Y SDMODxU (RPS update data set)	C ERROPT=YES and ERREXT=YES E ERROPT=YES Z neither is specified	M TRUNCS=YES and FEOVD=YES T TRUNCS=YES W FEOVD=YES Z neither is specified	B CONTROL=YES and RDONLY=YES C CONTROL=YES Y RDONLY=YES Z neither is specified

LIOCS MODULE NAME VERSUS OPTIONS (....Cont'd)

Character # ➡	1	2	3	4	5	6	7	8
SDMOD (WORK FILES)	I	J	G	T HOLD=YES W HOLD=YES not specified	C ERROPT=YES and EREXT=YES E ERROPT=YES W RPS version of module Z neither is specified	N NOTEPNT=YES R NOTEPNT=POINTRW Z NOTEPNT is not specified	C CONTROL=YES Z CONTROL=YES is not specified	T RDONLY=YES and UPDATE=YES U UPDATE=YES Y RDONLY=YES Z neither is specified
DAMOD	I	J	I	F RECFORM=FIXUNB B RECFORM=UNDEF handles both UNDEF and FIXUNB S RECFORM=SPNUNB V RECFORM=VARUNB	A AFTER=YES W RPS version of module Z AFTER is not specified	E IDLOC=YES and FEOVD=YES I IDLOC=YES R FEOVD=YES Z neither is specified	H ERREXT=YES and RELTRK=YES P ERREXT=YES R RELTRK=YES Z neither is specified	W HOLD=YES and RDONLY=YES X HOLD=YES Y RDONLY=YES Z neither is specified
ISMOD	I	J	H	A RECFORM-BOTH, IOROUT=ADD or ADDRTR B RECFORM=FIXBLK, IOROUT=ADD or ADDRTR U RECFORM-FIXUNB, IOROUT=ADD or ADDRTR Z RECFORM is not specified (IOROUT=LOAD or RETRVE)	A IOROUT=ADDRTR (non-RPS version of module) I IOROUT=ADD L IOROUT=LOAD (non-RPS version of module) R IOROUT=RETRVE V IOROUT=ADDRTR (RPS version of module) X IOROUT=LOAD (RPS version of module)	B TYPEFLE=RANSEQ G IOAREA2=YES TYPEFLE=SEQNTL or IOROUT=LOAD R TYPEFLE=RANDOM S TYPEFLE=SEQNTL Z neither is specified (IOROUT=LOAD or ADD)	B CORINDX=YES and HOLD=YES C CORINDX=YES O HOLD=YES Z neither is specified	F CORDATA=YES, ERREXT=YES, RDONLY=YES G CORDATA=YES and ERREXT=YES O CORDATA=YES and RDONLY=YES P CORDATA=YES S ERREXT=YES and RDONLY=YES T ERREXT=YES Y RDONLY=YES Z neither is specified

Bytes Dec	Bytes Hex	Bits	Contents	Function
0-15 8	00-FF 8			CCB 1= 2501 double - CCW support
16	10	0		1= OMR [1]; 0= Omitted
		1		1= ERROPT [2]; 0= Omitted
		2		COBOL open; ignore option
		3		1= GET issued [3]; 0= GET not issued [7]
		4		DTF table address constants relocated by OPENR
		5-7		File association: 000= READ only 010= READ/PRINT [4] 101= READ/PUNCH/PRINT [5] 001= READ/PUNCH [5]
17-19	11-13			Address of logic module
20	14		X'02' X'05'	DTF type DTF type for 2560 or 5425
21	15	0		1= Open; 0= Closed
		1		First time switch
		2		1= 1442 or 2596; 0= Other
		3		1= 2560, 3525 or 5425; 0= Other
		4		1= 3504, 3505; 0= Other
		5		1= 2 I/O areas; 0= 1 I/O area
		6		1= 2520; 0= Other
		7		1= 2540; 0= Other
22	16		B'SSF0X010'	Normal command code (not for 2560 or 5425): SS: 00= pocket 1; 01= pocket 2; 10= pocket 3 [6] F : 1= Column binary [3]; 0= EBCDIC X: 1= OMR or RCE [3]; 0= neither
			B'H0B00010'	Read command code (2560): H : 0= hopper 1; 1= hopper 2 B : 0= EBCDIC; 1= column binary
			B'HMMM0011'	SS command code (5425). H: 0= hopper1, 1=hopper 2. MMM: 001=stacker 1,010=stacker 2. 011=stacker 3, 100=stacker 4.
23	17		B'H0B00010'	Control command code (not for 2560 or 5425) Read command code (2560, 5425)
24-27	18-1B			Address of IOAREA2
28	1C	0		1= 2560; 0= Other
		1		1= 5425; 0= Other
		2-7		Not used
29-31	1D-1F			Address of EOF routine
32-39	20-27			Read CCW
Bytes 40-49 as used for all files except 2560 and 5425 files				
40-43	28-2B		LA &IOREG,0(14) NOP 0	Load user pointer register
44-49	2C-31		MVC 0(&BLKSIZE,13),0(14) NOP 0 DC X'0000'	Move IOAREA to WORKA

Bytes Dec	Bytes Hex	Bits	Contents	Function
The following bytes (50-105) are used for 2501 double-CCW support.				
50-55	32-37			Unused CCB
56-71	38-47			
72	48	0		1= OMR[1], 0= omitted.
		1		1= ERROPT[2], 0= omitted.
		2		COBOL open; ignore option
		3		1= GET issued[3], 0 = GET not issued[7]
		4		DTF table address constants relocated by OPENR.
		5-7		File Association 000 = READ only[4] 010 = READ/PRINT 101 = READ/PUNCH/PRINT[5] 001 = READ/PUNCH[6].
73-75	49-4B			Address of logic module.
76	4C		X'02'	DTF type.
			X'05'	DTF type for 2560 or 5425.
77	4D	0		1=open; 0= closed
		1		First time switch
		2		1= 1442 or 2596; 0 = other.
		3		1= 2560, 3525, or 5425; 0 = other.
		4		1= 3504 or 3505; 0= other.
		5		1= 2 I/O areas; 0 = 1 O/I area.
		6		1= 2520; 0 = other.
		7		1= 2540; 0 = other.
78	4E		B'SSF0X010'	Normal command code (not for 2560 or 5425). SS : 00 = pocket1, 01 = pocket2, 10 = pocket 3[6]. [3] F : 1= column binary[3], 0= EBCDIC. X : 1 = OMR or RCE[3], 0 = neither.
			B'H0B00010'	Read command code (2560, 5425) H : 0= hopper 1, 1 = hopper2. B : 0 = EBCDIC, 1 = column binary.
79	4F			Control command code (not for 2560 or 5425).
			B'H0B00010'	Read command code (2560, 5425).
80-83	50-53			Address of IOAREA2. (If IOAREA2 is not specified, address of IOAREA1).

Bytes Dec	Bytes Hex	Bits	Contents	Function
84	54	0 1 2-7		1 = 2560; 0 = other. 1 = 5425; 0 = other. Not used.
85-87	55-57			Address of EOF routine.
88-95	58-5F			Read CCW.
96-99	60-63		LA &IOREG, 0(14) NOP 0	Load user pointer register.
100-103	64-67		MVC 0 (&BLKSIZE, 13), 0(14) NOP 0	Move IOAREA to WORKA
104-105	68-69		DC X'0000'	
The following bytes (50-57) are used for 3504, 3505, and 3525 associated files.				
50-53	32-35		DC A (name) B 16(15) B 20(15) DC F'0'	If ERROPT=name[2]. If ERROPT=SKIP. If ERROPT=IGNORE. If ERROPT=omitted.
54-57	36-39		DC A(ASOCFLE)	Address of associated DTF table[7]. (3525 only).
Bytes 40 onward as used for 2560 and 5425 files.				
40-47	28-2F			Stacker select CCW (2560). Read CCW (5425).
48-51	30-33		LA &IOREG,0(14) NOP 0	
52-57	34-39		MVC 0(&BLKSIZE, 13), 0(14) NOP 0 DC X'0000'	Move IOAREA to WORKA
58-63	3A-3F		CLC 0(L, 14), 64(1)	Test for end of file. L= 4 if MODE=C; L=2 in other cases.
64-67	40-43		DC C'/* ' DC X' 0C001022'	End-of-file indicator if MODE=E In other cases.

Bytes		Bits	Contents	Function
Dec	Hex			
68-71	44-47		DC A(name) B 16(15) B 20(15) DC F'0'	If ERROPT=name[2]. If ERROPT=SKIP. If ERROPT=IGNORE. If ERROPT=omitted.
72-75	48-4B		DC A(ASOCFLE)	Address of associated DTF table[7].
76-81	4C-51		MVC 0 (&BLKSIZE, 14), 82(1)	Move card image to IOAREA1
82	52		DC &BLKSIZE.C' '	Buffer for card image.

1 OMR only for 3504 and 3505.
2 ERROPT for 2560, 3504, 3505, 3525, or 5425 READ file.
3 3504, 3505, and 3525 with or without CONTROL=YES specified.
4 2560, 3525, or 5425 with or without CONTROL=YES specified.
5 2560, 3525, or 5425 without CONTROL=YES specified.
6 Defaults to pocket2 for 3504, 3505, and 3525.
7 Present only when 2560, 3525, or 5425 associated files are specified for the input DTF.

DTFCD (Punch)

Bytes Dec	Bytes Hex	Bits	Contents	Function
0-15	00-FF			CCB
16	10	0		Not used
		1		1= ERROPT [3]; 0= Omitted
		2		COBOL open; ignore option
		3		1= PUT issued [2]; 0= PUT not issued
		4		DTF table address constants relocated by OPENR
		5-7		File Association: 000= PUNCH only 011= PUNCH/PRINT [3] 001= READ/PUNCH [3] 101= READ/PUNCH/PRINT [3] 100= PUNCH/INTERPRET [3]
17-19	11-13			Address of logic module
20	14		X'04'	DTF type
21	15	0		1= Open; 0= Closed
		1		First time switch
		2		1= CTLCHR
		3		1= Fixed unblocked
		4		1= Variable unblocked
		5		1= 2 I/O areas
		6		1= Workarea
		7		1= 2 CCWs in table; 0= 1 CCW in table
22	16		B'SSF00001'	Normal command code: SS : 00= pocket 1; 01= pocket 2; 10= pocket 3 [4] F : 1= column binary; 0= EBCDIC
			B'HSSS0011'	Normal stacker select command code (2560 or 5425). H : 0= hopper 1; 1= hopper 2 SSS: stacker information
23	17		B'HSSS0011'	Control command code (not for 2560 or 5425). Actual stacker select command code (2560 or 5425)
24-27	18-1B		DC A(IOAREA1+x)	Address of data in IOAREA1
28-31	1C-1F			Bucket [1]
32-33	20-21		LR 12,(RECSIZE)	Undefined records only
34-37	22-25		LA &IOREG,4(14) NOPR 0	Load user pointer register
38	26	0-2		Not used
		3		1= 5425
		4		1= 2560
		5		1= 3525
		6		1= 1442 or 2596
		7		1= 2520B1
39	27		DC C' '	Blank for eject last card

DTFCD (Punch) (....Cont'd)

Bytes Dec	Bytes Hex	Bits	Contents	Function
For all files except 2560 and 5425 files:				
40-47	28-2F			Punch CCW
48-55	30-37			Eject CCW for last card if 2520
For 2540 files if CRDERR is specified				
48-55	30-37			Retry CCW
56-135	38-87		DC CL80' '	Savearea card image
For 3525 PUNCH/INTERPRET files				
48-55	30-37			Load CCW
56-63	38-3F			Print CCW
64-127	40-7F		DC 64C' '	Print buffer
For 3525 Associated files				
48-51	30-33		DC A(ASOCFLE)	Pointer to associated file
For 2560 and 5425 files				
40-47	28-2F		DC D'0'	Eject CCW If FUNC= RP or RPW
48-55	30-37			Stacker select CCW
56-63	38-3F			Punch and Feed CCW
For 2560 PUNCH/INTERPRET files				
64-71	40-47			Load print head buffer 1 CCW
72-79	48-4F			Load print head buffer 2 CCW
80-87	50-57			Print CCW
88-151	58-97		DC 64C' '	Save area for printing line 2
For 5425 PUNCH/INTERPRET files				
64-71	40-47			Print CCW
For 2560 and 5425 Associated files				
64-67	40-43		DC A(ASOCFLE)	
68	44		DC C' '	If mode is EBCDIC
			DC X'00'	If mode is Column Binary
69-	45-		DC &BLKSIZE.C' '	Buffer for card image

1) The bucket bytes handle undefined length records
2) Valid for 2560 or 3525 READ/PUNCH, PUNCH/PRINT, and READ/PUNCH/PRINT files
3) Valid for 2560 or 3525 only. (3504, 3505, 5425) non-associated files
4) Defaults to pocket 2 for 3525

DTFCD (Combined Reader/Punch)

Bytes Dec	Bytes Hex	Bits	Contents	Function
0 -15	00-0F			CCB
16	10	0-1 2 3 4 5-7		Not used COBOL open; ignore option Not used OPENR relocates DTF address constants Not used
17-19	11-13			Address of logic module
20	14		X'00'	DTF type
21	15			Command code (X'02' for 1442, X'C2' for 2520, 2540)
22	16			Command code (X'01' for 1442, X'09' for 2520, 2540)
23	17			Command code (X'01' for 1442, X'09' for 2520, 2540)
24-31	18-1F			CCW
32-35	20-23			Input area address
36-39	24-27			Output area address
40-41	28-29			Input blocksize
42-43	2A-2B			Ouput blocksize
44-49	2C-31		MVC 0 (&BLKS, 13,0(14)	
50-55	32-37		MVC 0 (&OUBL, 14),0(13)	
56-59	38-3B			End-of-file address
60-67	3C-43			Save area
68-73	44-49		MVC 1 (&OUBL-1, 13),0(13)	
74-77	4A-4D		MVI 0(13), X'40'	
78-79	4E-4F			Constant (blanks)
80-83	50-53			Constant address (bytes 78-79)

Bytes Dec	Bytes Hex	Bits	Contents	Function
0 -15	00-0F			CCB
16	10	0		1= 2-line printer (3,4); 0= Other
		1		1= ERROPT (3,4); 0= Omitted
		2		COBOL open; ignore option
		3		1= 3525; 0= Other
		4		OPENR relocates DTF address constants
		5-7		000= PRINT only 011= PUNCH/PRINT 3) 010= READ/PRINT 3) 101= READ/PUNCH/PRINT 3)
17-19	11-13			Address of logic module
20	14		X'08' X'07'	DTF type DTF type for 2560 and 5425
21	15	0		1= Open; 0= Closed
		1		First time switch
		2		1= Control character
		3		1= Fixed unblocked records
		4		1= Variable unblocked records
		5		1= 2 I/O areas
		6		1= Workarea
		7		1= Print overflow channel 9
For Printer and Card Punch devices				
22	16		X'09'	Normal command code 5)
23	17		X'09'	Control command code 5)
24-27	18-1B		DC A(IOAREA1+x)	Address of data in IOAREA1
28-31	1C-1F			Bucket 1)
32-33	20-21		LR 12,(RECSIZE) NOPR 0	For undefined records only
34-37	22-25		LA &IOREG,4(14) NOP 0	Only if IOREG= (r)
38-39	26-27			Bucket 2)
40-47	28-2F		11,*,X'60',1 9,IOAREA,X'20',121	CCW- Set up Selective Tape List Control STLIST not specified 6)
48-55	30-37		9,IOAREA,X'20',121 A (Name)	CCW- STLIST specified 6) Address of user error routine (3211 only)
			DC A(ASOCFLE)	If ASOCFLE= filename 3)
For the 2560 and 5425 Multi Function Card Machine				
22	16		X'00'	Not used
23	17		B'HHHHHH00'	Print head selection byte H= 1 specifies the corresponding head
24-27	18-1B			Address of IOAREA1

Bytes Dec	Bytes Hex	Bits	Contents	Function
For the 2560 Multi Function Card Machine (Cont'd)				
28-31	1C-1F			Bucket
32-33	20-21		LR 12,(RECSIZE) NOPR 0	For undefined records only
34-37	22-25		LA &IOREG,4(14) NOP 0	Only if IOREG= (r)
38-39	26-27			Number of bytes to be printed by the last specified print head
40-43	28-2B		DC A(ASOCFLE) DC F'0'	If FUNC= RW, PW or RPW In all other cases
44	2C	0 1 3 4-7		1= 2560 Not used 1= Print control switch for 2560 associated files Not used
45-47	2D-2F		DC 3X'00'	Reserved for future use
For 2560 simple files				
48-55	30-37			Eject CCW
56-63	38-3F			Load print head buffer CCW
64-71	40-47			Print CCW
For 2560 associated files				
48-55	30-37			Load print head buffer CCW
56-63	38-3F			Print CCW
For 5425 files				
48-55	30-37			Print CCW

1) The bucket bytes handle undefined records. Bit 0 of byte 28 at open time determines the mode set of a printer with UCS. If bit 0= 1, the mode is set so that data checks occur if an invalid character is printed. Otherwise, mode is set to suppress data checks. The use of the UCS parameter determines the setting of this bit. If STLIST= YES, byte 31 saves the STLIST control byte provided by the PUT macro.

2) The 2 byte bucket saves print overflow conditions if CTLCHR= ASA. If STLIST= YES, byte 38 contains the current STLIST control byte. Byte 39 is set by the PUT macro to indicate spacing or skipping. (X'00' no spacing, no skipping; X'01' spacing; X'02' skipping).

3) Valid for 2560, 3525 READ/PRINT, PUNCH/PRINT and READ/PUNCH/PRINT files.

4) Valid for 3525 PRINT only files.

5) X'05' for 3525; X'09' for other devices

6) Valid for 1403 only

Bytes Dec	Bytes Hex	Contents	Function
0 -15	00-0F		CCB
16	10	X'20' X'08'	COBOL open; ignore option DTF table address constants relocated by OPENR
17-19	11-13		Address of logic module: GET and PUT logic if TYPEFLE= INPUT; PUT logic if TYPEFLE= OUTPUT; GET,PUT and PUTR logic if TYPEFLE= CMBND
20	14	X'03'	DTF type
21-23	15-17		For input and output: not used For combined: byte 21 contains X'01' and bytes 22-23 contain INPSIZE
24-31	18-1F	X'09', IOAREA1, X'00', BLKSIZE	CCW
End of table if RECFORM=FIXUNB and WORKA not specified. The following bytes are added if WORKA is specified.			
32-35	20-23	DC A(IOAREA1)	Address of I/O area
36-39	24-27	DC F'0'	Register save area
40-43	28-2B	DC F'0'	Register save area
End of table if RECFORM=FIXUNB. The following bytes are added if RECFORM=UNDEF			
		DC F'0'	Register save area
		DC F'0' DC H'BLKSIZE' DC AL2(BLKSIZE-1)	Register save area I/O area size For input files only
The following bytes are added to the table if TYPEFLE= CMBND			
32-35	20-23	DC A(IOAREA1+BLKSIZE)	I/O area address for input
36-37	24-25	DC H'BLKSIZE'	Blocksize

DTFDR

Bytes Dec	Bytes Hex	Bits	Contents	Function
0 -15	00-0F			CCB
16	10	0-1		Not used
		2		COBOL open; ignore option
		3		Not used
		4		OPENR relocates DTF table addresses
		5-7		Not used
17-19	11-13			Address of logic module
20	14		X'C0'	DTF type
21	15			PIOCS switches:
		0		1= open; 0= closed
		1	B'1'	Input
		2-5	B'0000'	Not used
		6	B'1'	Device is 3886
		7	B'0'	Not used
22	16			Error indicator byte
23	17			LIOCS switches:
		0-4	B'00000'	Not used
		5		1= SETDEV
		6		1= Control passed to COREXIT
		7		1= FR loaded from disk
24-31	18-1F			FR phasename at open time
32-39	20-27			Phasename of currently used FR
40-43	28-2B		X'00000000'	Not used
44-47	2C-2F			Start address of FR area in DTF
48-51	30-33			Address of four-byte pointer at the end of the FR area in the DTF
52-55	34-37			EOF routine address
56-63	38-3F			Scan CCW
64-71	40-47			Read CCW
72-79	48-4F			Read CCW
80-87	50-57			Control CCW
88-95	58-5F			Load format record CCW
96-99	60-63			COREXIT routine address
100-103	64-67			IOAREA1 area address
104-107	68-6B			Header area address
108-111	6C-6F			Exit indicator address
112	70			Start of FR area

Bytes Dec	Bytes Hex	Bits	Function
0 -15	00-0F		Dummy CCB
16	10	0-1	Not used
		2	COBOL open; ignore option
		3	Not used
		4	DTF table address constants relocated by OPENR
		5-7	Not used
17-19	11-13		Address of logic module
20	14		DTF type, (X'09')
			DTF type, (X'0A' if HEADER=YES)
21	15		PIOCS switches:
		0	1= Open; 0= closed
		1	1= Input
		2	1= Control
		3	1= Device is 1287
		4	1= Header
		5	Reserved for future use
		6	1= RDLNE
		7	Not used
22	16		Not used
23	17	0-6	Not used
		7	1= LIOCS posts a hopper empty condition to DTF
24-39	18-27		CCB
40-47	28-2F		Sense CCW
48-51	30-33		Lost lines (equipment check)
52-55	34-37		After 9 retries for journal tape, or after 2 retries for documents
56-59	38-3B		Wrong length records
60-63	3C-3F		After 4 retries for journal tape, or after 2 retries for documents
64-67	40-43		Keyboard corrections
68-71	44-47		Count of data check errors
72-75	48-4B		Lines marked
76-79	4C-4F		Total lines read (CCW chains executed)
80	50		Error indicators:
		0	1= EOP
		1	1= Lost reference mark indicator
		2	1= Late stacker selection
		3	1= Non-recovery error
		4	1= Equipment check
		5	1= Wrong length record
		6	1= Hopper empty
		7	1= Data check
81	51		LIOCS switches:
		0	1= First time
		1	1= 2 I/O areas

Bytes		Bits	Function
Dec	Hex		
81 (Cont'd)		2 3 4 5-7	1= WORKA= YES 1= RECFORM= FIXUNB 1= RECFORM= UNDEF Not used
82	52		Normal command code
83	53		Control command code
84-87	54-57		IOAREA2 address
88-95	58-5F		Read CCW
96-103	60-67		Go to next line CCW
104-111	68-6F		Control CCW
112-115	70-73		EOF address
116-119	74-77		Correction exit address
120-123	78-7B		IOAREA1 address
124-127	7C-7F		DC A(&BLKS-1)
128-129	80-81		SR 13, &RECS
130-131	82-83		LR &RECS, 13
132-133	84-85		LR &IOR, 13
134-135	86-87		Sense

Bytes Dec	Bytes Hex	Bits	Function
0 -5	00-05		CCB indicators
6 -7	06-07		Logical class and unit numbers (primary if DUAL addressing)
8	08		Zero
9 -11	09-0B		CCW address
12-15	0C-0F		Zeros
16	10	0-1	Not used
		2	COBOL open; ignore option
		3	Not used
		4	DTF table address constants relocated by OPENR
		5-7	Not used
17-19	11-13		Address of logic module
20	14		DTF type= X'0B'
21	15		Logic module option switches:
		0	User disengage 0= off; 1= on
		1	Program sort mode 0= no; 1= yes
		2	First time switch (after engage) 0= no; 1= yes
		3	Addressing= DUAL 0= no; 1= yes
		4	Waiting 0= no; 1= yes
		5	Read logic indicator 0= no; 1= yes
		6	Not used
		7	Supervisor initial read (after open) 0= no; 1= yes
22-29	16-1D		Symbolic filename
30	1E		Open/Close switch:
		0	0= closed; 1= open
31-33	1F-21		Open/Close option switches
34-35	22-23		Logic module option switches
36-39	24-27		Error information status
40-41	28-29		Length of DTF table
42-43	2A-2B		Device type indicator
44-45	2C-2D		Record type
46-49	2E-31		Reserved for future use
50-51	32-33		I/O register
52-55	34-37		End-of-file address
56-59	38-3B		IOAREA2/1 address
60-63	3C-3F		Document buffer size
64-65	40-41		Blocking factor/Number of buffers
66-67	42-43		I/O area size
68-71	44-47		Record length
72-76	48-4C		Sense information

Bytes Dec	Bytes Hex	Bits	Function
77	4D		Supervisor switch
78-79	4E-4F		Logical class and unit numbers (secondary, for DUAL addressing only)
80-81	50-51		Register alignment bytes
82-83	52-53		Logical class and unit numbers (primary, for DUAL addressing)
84-87	54-57		Document buffer size
88	58		Command code (4C)
89-91	59-5B		Address of last byte of first document buffer
92	5C		Command code (4C)
93-95	5D-5F		Address of last byte of last document buffer
96-99	60-63		Stacker select routine address
100-103	64-67		Address of stacker select CCW chain
104-107	68-6B		Current buffer address pointer (Supervisor)
108-111	6C-6F		Supervisor count
112-113	70-71		Number of buffers minus 7
114-115	72-73		Message indicator
116-119	74-77		ERROPT routine address
120-121	78-79		Logical class and unit numbers (secondary, for DUAL addressing only)
122-123	7A-7B		Reserved for future use
124-127	7C-7F		Address of last buffer given to user
128-131	80-83		Address of first byte of last buffer
132-139	84-8B		Channel status word (CSW)
140-143	8C-8F		Address of active GET record
144-147	90-93		GET counter
148-159	94-9F		Reserved for future use
For single addressing			
160-167	A0-A7		CCW - Engage
168-175	A8-AF		CCW - Read
176-183	B0-B7		CCW - Sense
184-191	B8-BF		CCW - NOP
192-199	C0-C7		CCW - Stacker select
200-207	C8-CF		CCW - TIC
208-215	D0-D7		CCW - Control
216-223	D8-DF		CCW - BN

Bytes		Bits	Function
Dec	Hex		
224-231	E0-E7		CCW - Read
232-239	E8-EF		CCW - Sense
240-247	F0-F7		CCW - Disengage
For DUAL Address Adapter			
160-167	A0-A7		CCW - Engage
168-175	A8-AF		CCW - Read buffer 1
176-183	B0-B7		CCW - Sense
184-191	B8-BF		CCW - NOP
192-199	C0-C7		CCW - Read buffer 2
200-207	C8-CF		CCW - MOD Sense
208-215	D0-D7		CCW - Read buffer 1
216-223	D8-DF		CCW - MOD Sense
224-231	E0-E7		CCW - TIC to NOP
232-239	E8-EF		CCW - NOP
240-247	F0-F7		CCW - MOD CTL
248-255	F8-FF		CCW - Stacker select
256-263	100-107		CCW - MOD Sense

Bytes Dec	Bytes Hex	Bits	Contents*	Function	Record Format*
0 -15	00-0F			CCB	
8	08		Input: X'00'-X'63' Output: X'00'-X'04' (variable) X'00' (undefined)	Buffer offset length, ASCII	
16	10	0		First time entered MTMOD for a file	
		1		Not used	
		2		COBOL open; ignore option	
		3		American National Standard COBOL	
		4		DTF table address constants relocated by OPENR	
		5		1= spanned records	
		6		1= ASCII - 0= EBCDIC	V-V, S
		7		ASCII input: 1= Length check	V
				ASCII output: 1= Buffer offset length = 4	V
17-19	11-13			Address of logic module	
20	14		X'11'	Nonstandard or unlabeled	
			X'12'	Standard labeled, output	
			X'13'	Standard labeled, input, backwards	
			X'14'	Standard labeled, input, forwards	
21	15	0		First time switch: 1= not first-time entry 0= first-time entry	
		1		1= blocked 0= unblocked	
		2		1= 2 I/O area's 0= 1 I/O area	
		3		1= workarea	F, U, V
				0= no workarea	F, U, V
				0= workarea, spanned	S
		4		1= input 0= output	
		5		1= backwards 0= forwards	
		6		1= checkpoint 0= no checkpoint	
		7		1= TRUNC required during Close	
22-29	16-1D			Symbolic filename	
30	1E			Same as command code in CCW; (X'01', X'02' or X'0C')	
31	1F	0-4		Bits 0-4 are used as displacements by OPEN to determine the location of variable fields of the DTF.	

Bytes Dec	Bytes Hex	Bits	Contents *	Function	Record Format
31 (Cont'd)	1F		B'01110'	Input	F
			B'01100'	Output	F
			B'10001'	Input	V
			B'01111'	Output	V
			B'01101'	Input	U
			B'01011'	Output	U
		5		1= Tape label information included in DTF (see bytes 88-95) 0= Tape label information not included in DTF	
		6		Used by COBOL	
		7		1= Header label and EOV information wanted 0= No header label and EOV information wanted	
32	20	0		Standard labels: 1= yes; 0= no	
		1		Labels: 1= nonstandard; 0= unlabeled	
		2		Rewind unload: 1= yes; 0= no	
		3		Rewind option: 1= no rewind; 0= rewind	
		4		Drive direction: 1= backwards; 0= forwards	
		5		User label address: 1= yes; 0= no	
		6		Tapemark option: 1= no; 0= yes	
		7		EOF-EOV switch (used by IBM SORT): 1= yes; 0= no	
33-35	21-23			User label routine address	
36	24	0		DTFPH: 1= yes; 0= no	
		1		COBOL indicator: 1= yes; 0= no	
		2		File type: 1= input; 0= output	
		3		FEOV switch: 1= yes; 0= no	
		4		EOF-EOV switch (output): 1= EOF 0= EOV	
		5		Open indicator: 1= open; 0= closed	
		6		1= variable or spanned records	V, S
		7		1= undefined records	U
37-39	25-27			EOF address	
40-43	28-2B			Block count	
44-47*	2C-2F		BXH 11,12, 24(15)	Forward	F
			BXLE 11,12 24(15)	Backward	F
			L &VARBLD, DEBLOCKER	If VARBLD parameter is used	V
			NOP 0(0)		S
			DC F'0'	DEBLOCKER1	U

Bytes Dec	Bytes Hex	Bits	Contents*	Function	Record Format
48-51	30-33		LA 14,1(14)		F,V,S
			BCTR 14,0 + NOPRO	Backward	F,V,S
			L &RECSIZE, DEBLOCKER1	If RECSIZE given	U
			NOP 0(0)	For input if not NOP	U
52-55	34-37		L &IOREG, DEBLOCKER1	If IOREG specified	F
			L &IOREG, DEBLOCKER5	If IOREG specified	V
			L EIOREG, DEBLOCKER2	If IOREG specified	U
			NOP 0(0)	If no IOREG	
			L &RECSIZE, IJFVSREC	If spanned input	S
			ST &RECSIZE, IJFVSREC	If spanned output	S
56-63	38-3F			CCW	
64-67	40-43		DC A(IOAREA1)	One I/O area	
			DC A(IOAREA1 +BLKSIZE-1)	One I/O area, read backward	
			DC A(IOAREA2)	Two I/O area's	
			DC A(IOAREA2 +BLKSIZE-1)	Two I/O area's, read backward	
68-71	44-47		DC F'0'	Input	F
			DC A(IOAREA1 +BLKSIZE-RECSIZE)	Input backward: DEBLOCKER1	F
			DC A(IOAREA1)	1 I/O area, output: DEBLOCKER1	F
			DC A(IOAREA2)	2 I/O area's, output: DEBLOCKER1	F
			DC A(BLKSIZE)	DEBLOCKER1: EBCDIC	V,S
			DC A(IOAREA1)	1 I/O area: DEBLOCKER2	U
			DC A(IOAREA2)	2 I/O area's: DEBLOCKER2	U
72-75	48-4B		DC F'RECSIZE'	Forward: DEBLOCKER2	F
			DC F'-RECSIZE'	Backward: DEBLOCKER2	F
			DC A(IOAREA1)	1 I/O area: DEBLOCKER2	V,S
			DC A(IOAREA2)	2 I/O area's: DEBLOCKER2	V,S
			LA 14,1(14)	Forward	U
			BCTR 14,0 + NOPRO	Backward	U
76-79	4C-4F		DC F'0'	Input forward: DEBLOCKER3	F
			DC A(IOAREA1 +BLKSIZE - RECSIZE)	Input backwards: DEBLOCKER3	F
			DC A(IOAREA1 +BLKSIZE-1)	Output, 1 I/O area: DEBLOCKER3	F
			DC A(IOAREA2 +BLKSIZE-1)	Output, 2 I/O area's: DEBLOCKER3	F

Bytes Dec	Bytes Hex	Bits	Contents	Function	Record Format
76-79 (Cont'd)			DC F'0' DC Y(BLKSIZE) DC Y(BLKSIZE -1)	DEBLOCKER3 (Bytes 76-77 only) (Bytes 78-79 only)	V,S U U
80-83	50-53		DC Y(BLKSIZE) + Y(BLKSIZE-1) DC &(BLKSIZE) +Y(BLKSIZE+1) DC F'0' LR 12,RECSIZE DC H'0'	Forward Backward DEBLOCKER4 (Bytes 80-81 only) (Bytes 82-83 only)	F F V,S U U
84-87	54-57		DC Y(RECSIZE -1) DC 2X'00' DC A(IOAREA1 +4) DC A(IOAREA2 +4) DC A(IOAREA1 +BUFOFF) DC A(IOAREA2 +BUFOFF) DC 2X'00'	(Bytes 84-85) (Bytes 86-87) Output, Standard labels 1 I/O area: DEBLOCKER 5, EBCDIC 2 I/O area's: DEBLOCKER 5, EBCDIC 1 I/O area: DEBLOCKER5, ASCII 2 I/O area's: DEBLOCKER5, ASCII (Bytes 84-85 output only) Standard labels; reserved for OPEN	F F V,S V,S V V F,U
			B 28(15) B 24(15) B 28(15) DC A(ERROPT)	Input only, ERROPT= omitted Input only, ERROPT= SKIP Input only, ERROPT= IGNORE Input only, ERROPT= ADDRESS	U U U U
88-91	58-5B		DC A(WLRERR) B 24(15) B 28(15) DC 2X'00' DC A(ERROPT)	Input only, WLRERR= ADDRESS Input only, WLRERR omitted and ERROPT= SKIP Input only, WLRERR omitted and ERROPT= IGNORE or omitted Output only, standard labels (bytes 88-89), reserved for OPEN Input only, WLRERR omitted and ERROPT= ADDRESS	For fixed-length records only
90-95	5A-5F		DC 6X'00'	File serial number, Standard labels, Output only	
92-95	5C-5F		DC A(ERROPT) B 28(15) B 24(15) B 28(15)	Input only, ERROPT= ADDRESS Output, nonstandard labels only. ERROPT= ADDRESS Input only, ERROPT= omitted Input only, ERROPT= SKIP Input only, ERROPT= IGNORE	
96-99	60-63		DC 4X'00'	Volume sequence number, Standard labels, output only	
96-97	60-61		DC 2X'00'	Standard labels, input only, reserved for OPEN	

Bytes Dec	Bytes Hex	Bits	Contents*	Function	Record Format
98-103	62-67		DC 6X'00'	File serial number, Standard labels, Input only.	For fixed-length records only
100-103	64-67		DC 4X'00'	File sequence number, Standard labels, output only	
104-107	68-6B		DC 4X'00'	Volume sequence number, Standard labels, input only	
			DC A(ERROPT)	Output only, Standard labels only. ERROPT=ADDRESS	
108-111	6C-6F		DC 4X'00'	File sequence number, Standard label, input only	
88-91	58-5B		DC F'0'	DEBLOCKER6	
92-95	5C-5F		DC A(ERROPT)	Output only, Nonstandard labels only. ERROPT= ADDRESS	
92-93	5C-5D		DC Y(BLKSIZE)	Input only	
			DC Y(BLKSIZE -4)	Output only: EBCDIC	
			DC Y(BLKSIZE -BUFOFF)	Output only: ASCII	(V only)
94-95	5E-5F		DC Y(BLKSIZE -1)		
96-97	60-61		DC Y(RECSIZE -1)		For variable-length and spanned records
98-99	62-63		DC H'0'	Input only: Residual count	
100-103	64-67		DC A(WLRERR)	Input only, WLRERR= ADDRESS	
			B 24(15)	Input only, WLRERR= omitted and ERROPT= SKIP	
			B 32(15)	Input only, WLRERR= omitted and ERROPT= IGNORE or omitted	
100-101	64-65		DC 2X'00'	Output only, Standard labels, reserved for OPEN	
104-107	68-6B		DC A(ERROPT)	Input only, ERROPT= ADDRESS	
			B 28(15)	Input only, ERROPT= omitted	
			B 24(15)	Input only, ERROPT= SKIP	
			B 28(15)	Input only, ERROPT= IGNORE	
			DC A(ERROPT)	Output, Nonstandard labels only (version 3 onward); ERROPT=ADDRESS	
108-111	6C-6F		DC 4X'00'	Volume sequence number; Standard labels, output only	
108-109	6C-6D		DC 2X'00'	Standard labels, input only; Reserved for OPEN	
110-115	6E-73		DC 6X'00'	File serial number; Standard labels, input only	
112-115	70-73		DC 4X'00'	File sequence number; Standard labels, output only	

Bytes Dec	Bytes Hex	Bits	Contents*	Function	Record Format
116-119	74-77		DC A(ERROPT)	Output only, ERROPT= ADDRESS, Standard labels only	For variable length and spanned records
116-119	74-77		DC 4X'00'	Volume sequence number; Standard labels, input only	
121-123	78-7B		DC 4X'00'	File sequence number; Standard labels, input only	
86-91	56-5B		DC 6X'00'	File serial number; Standard labels, output only	For undefined records only
88-91	58-5B		DC A(WLRERR)	Input only; WLRERR= ADDRESS	
			B 24(15)	Input only; WLRERR= omitted and ERROPT= SKIP	
			B 28(15)	Input only; WLRERR= omitted and ERROPT= IGNORE or omitted	
			DC A(ERROPT)	Input only; WLRERR= omitted and ERROPT= ADDRESS. Output only; nonstandard labels, ERROPT= ADDRESS	
92-95	5C-5F		DC 4X'00'	Volume sequence number; Standard labels, output only	
92-93	5C-5D		DC 2X'00'	Standard labels; input only, Reserved for OPEN	
94-99	5E-63		DC 6X'00'	File serial number; Standard labels, input only	
96-99	60-63		DC 4X'00'	File sequence number; Standard labels, output only	
100-103	64-67		DC 4X'00'	Volume sequence number; Standard labels, input only	
100-103	64-67		DC A(ERROPT)	Output only; Standard labels only, ERROPT= ADDRESS	
100-103	64-67		DC 4X'00'	Volume sequence number; Standard labels, output only	
104-107	68-6B		DC 4X'00'	File sequence number; Standard labels, input only	
100-103	64-67		DC A(WLRERR)	Input only; WLRERR= ADDRESS	For spanned records only
			B 24(15)	Input only; WLRERR= omitted and ERROPT= SKIP	
			B 32(15)	Input only; WLRERR= omitted and ERROPT= IGNORE or omitted	
100-101	64-65		DC 2X'00'	Output only; Standard labels, reserved for OPEN	
102-107	66-6B		File serial number	Standard labels, output only	
100-103	64-67		DC 4X'00'	Output only; ERROPT= ADDRESS Nonstandard labels only	

Bytes Dec	Bytes Hex	Bits	Contents*	Function	Record Format
100-123	64-7B		DC 24X'00'	Output only; ERROPT= omitted, nonstandard labels	
104-107	68-6B		DC A(ERROPT)	Input only; ERROPT= ADDRESS	
			B 24(15)	Input only; ERROPT= omitted	
			B 24(15)	Input only; ERROPT= SKIP	
			B 28(15)	Input only; ERROPT= IGNORE	
104-107	68-6B		DC A(ERROPT)	Output only; ERROPT= ADDRESS, nonstandard labels	
108-123	6C-7B		DC 16X'00'	Output only; ERROPT= ADDRESS, nonstandard labels	
108-111	6C-6F		Volume sequence number	Standard labels, output only	
112-115	70-73		File sequence number	Standard labels, output only	
116-119	74-77		DC A(ERROPT)	Output only, ERROPT= ADDRESS, standard labels	
120-123	78-7B		DC 4X'00'	Output only, ERROPT= ADDRESS, standard labels	For spanned records only
108-123	6C-7B		DC 16X'00'	Input only, nonstandard labels	
108-109	6C-6D		DC 2X'00'	Standard labels, input only, reserved for OPEN	
110-115	6E-73		File serial number	Standard labels, input only	
116-119	74-77		Volume sequence number	Standard labels, input only	
120-123	78-7B		File sequence number	Standard labels, input only	
124-127	7C-7F		DC F'0'	Full word for loading and storing USER RECSIZE: IJFVSREC	
128	80		DC X'00'	IJFVSFLG	
		0		Sign bit, not used	
		1		Skip to first segment	
		2		First segment	
		3		Segment out of sequence, input only	
		4		Read back for EOV, output only	
		5		File reversed for logical spacing; Input CNTRL only; Trunc issued, output only	
		6		User Trunc issued, output only	
		7		Multi segment, output only; Skip Get segment, input CNTRL only	
129-131	81-83		DC 3X'00'	Pointer within WORKA	

See Notes on next page

* The format of the tape data file DTF is different starting at byte 44.
The location indicated by the numbers in the left hand column can contain only one of the factors listed under Contents .
The factor used for any given DTF table is determined by whether the file record format is fixed, variable or undefined, and by other DTF parameters as indicated.
A blank in the record column indicates that the contents apply to all record types.

- Record Format explanation

F = Fixed Record
V = Variable
U = Undefined
S = Spanned (variable format superset)

The deblockers are scratch areas used by the modules to save data from one GET/PUT macro instruction to another. In the text and listings, they are referred to by the names DEBLOCKER1 to 6. These are not labels; they are comments used to make it easier to follow the listings.

DTFMT (Workfiles)

Bytes		Bits	Function
Dec	Hex		
0 -15	00-0F		CCB
16	10	0-1	Not used
		2	COBOL open; ignore option
		3	1= VOL1 label is at user specified density
		4	1= DTF table address constants relocated by OPENR
		5-7	Not used
17-19	11-13		Address of logic module
20	14		DTF type= X'10'
21	15	0	1= No rewind
		1	1= Rewind unload
		2	1= Workfile
		3	1= Read backward
		4	1= Write
		5	1= POINTW
		6	Not used
		7	1= Forward-space file before next operation
22-23	16-17		Not used
24-25	18-19		Record length
26-27	1A-1B		Maximum BLKSIZE
28	1C		Read command code (X'02' for read forward; X'0C' for read backward)
29-31	1D-1F		EOF address
32-39	20-27		CCW
40-43	28-2B		Block count, initialized 00000000 for read forward, 00400000 for read backward
44	2C	0	1= Error routine
		1	1= Ignore
		2	Not used
		3	1= Record fixed unblocked
		4-7	Not used
45-47	2D-2F		DC A(ERROPT) Address of error routine

DTFSD (Data Files)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		Command Control Block (CCB)
	16	10	0	1= Dequeue old volume extents
			1	1= Dummy OPEN to obtain extents from label track
			2	1= File assigned 'IGN' (COBOL)
			3	1= Track hold option specified
			4	1= DTF relocated by OPENR
			5	1= Input trailer labels to be processed at close time (COBOL only)
			6	1= Spanned processing
			7	1= COBOL end-of-extent option specified
	17-19	11-13		Address of logic module
	20	14		DTF type for OPEN/CLOSE (X'20'= sequential access DASD files)
	21	15	0	1= 2321 (version 1/2 only)
			1	1= Blocked file
			2	1= Work file
			3	1= Workarea specified
			4	1= Not a Version 1 type table
			5	1= Open; 0= closed
			6	1= Input; 0= output
			7	1= User labels specified
	22-28	16-1C		Filename (DTF name)
	29	1D		Device type code: X'00'= 2311 X'01'= 2314,2319 X'02'= 2321 X'04'= 3330-1,-2 X'05'= 3330-11 X'07'= 3350 X'08'= 3340 general X'09'= 3340 35MB X'0A'= 3340 70MB Note: In previous versions, last byte of filename contains device type code
	30-35	1E-23		Address of Format 1 label in VTOC (BCCHHR)
	36-37	24-25		Volume sequence number
	38	26		Open communication byte: Input file
			0	1= No more extents
			1	1= Update file
			2	1= Process trailer labels
			3	1= Exit to user's EOF routine
			4	1= Next extent on new volume
			5	1= Return to close routine
			6	1= Process header labels
			7	1= Extent switch
				Output file
			0	1= No more extents
			1	1= Extents needed at close time
			2	1= Process trailer labels
			3	1= Process header labels
			4	1= Next extent on new volume
			5	1= Extents entered via console

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	38			Output file (Cont'd)
	(Cont'd)		6	1= Process trailer labels at close
			7	1= Check extent for minimum of 2 tracks
	39	27	0	1= Extent bypassed before file is opened (input only)
			1	1= FEOVD has been issued (input only)
			0-7	Sequence number of current extent opened (Output only)
	40	28		Sequence number of last extent opened
	41-43	29-2B		Address of user's label routine
	44	2C	0	not used
			1	1= Device supports RPS
			2-6	not used
			7	1= DTF has been extended into the partition virtual area
	45-47	2D-2F		Address of IOAREA1
	48-51	30-33		CCHH address of user's label track (X'80000000')
	52-53	34-35		Lower head limit (HH)
	54-57	36-39		Extent upper limit (CCHH)
&Filename.S	58-59	3A-3B		Seek address (BB): X'0000' if a disk device X'00nn' if 2321; where nn= bin number
	60-63	3C-3F		Search argument (CCHH)
	64	40		Record number
	65-67	41-43		EOF address if input file ; Key length and data length if output file
	68-71	44-47		CCHH control field:
				CCHH= X'00C80009' if 2311 - type 1
				CCHH= X'00C80013' if 2314 or 2319 - type 1
				CCHH= X'13090413' if 2321 - type 1
				CCHH= X'01940012' if 3330 - type 1
				CCHH= X'03280012' if 3330-11 - type 1
				CCHH= X'015C000B' if 3340 35MB
				CCHH= X'02B8000B' if 3340 70MB
				CCHH= X'022B001D' if 3350 - type 1
				where nn= current upper head number
	72	48		Number of records per track (input) or number of records per track - minus one (output)
	73	49		Switch byte used by the logic modules for various switching purposes. Functions indicated are for the ON condition(1) of the respective bit.
				Fixed length Record Modules
			0	Not first entry after Open (INPUT and UPDATE) Not first write after Open (OUTPUT)
			1	Short record (INPUT and UPDATE without truncation)
			2	Partial block written (OUTPUT)
			3	ERROPT= SKIP (INPUT); TRUNC= YES (OUTPUT)
			4	End-of-file record written (OUTPUT) End of extent (UPDATE)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	73 (Cont'd)			Fixed length Record Modules (Cont'd)
			5	Truncation not specified (used by OPEN routines)
			6	Write block of records (UPDATE)
			7	End of file (UPDATE)
				Variable length Record Modules
			0	Not first entry after OPEN (INPUT and UPDATE) Write record (OUTPUT)
			1	Wrong length record (INPUT); TRUNC=YES (OUTPUT); Second GET operation performed (UPDATE)
			2	Return to close routine (OUTPUT) Update specified (UPDATE)
			3	Not first entry after OPEN (OUTPUT)
			4	New extent required by CLOSE
			5	Capacity of I/O area exceeded (OUTPUT) Second GET required (UPDATE)
			6	Not first read (INPUT) Second GET issued (UPDATE)
			7	Unnecessary to read (INPUT) Track capacity exceeded (OUTPUT) Save record count (UPDATE)
				Undefined length Record Modules
			0	Not first entry after OPEN (ALL modules)
			1	Save record count (UPDATE)
			2	Return to close routine (OUTPUT)
			3	Second GET issued (UPDATE)
			4	Not used
			5	PUT command issued (UPDATE)
			6	End of file reached (UPDATE)
			7	Multi-track operation (UPDATE)
	74-75	4A-4B		Block size minus 1
	76-80	4C-50		CCHHR= Extent lower limit and record number. Field is used as a search argument bucket by the logic modules
	81	51	1	1=FEOVD has been issued (output only)
	81-83	51-53		Address of user wrong-length record routine if input file; Track capacity counter if output file
	84-87	54-57		Instruction to load user's register IOREG. (Note: This field is a NOP unless blocked records are processed in one I/O area, or two I/O areas are specified and records are processed in the I/O areas)
	88-91	58-5B		Address of current available input/output area
	92-95	5C-5F		Logical record size
	96-99	60-63		Address of end of input/output area

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	100	64		Logical indicators:
			0	1= ERROPT= address
			1	1= ERROPT= IGNORE
			2	1= ERROPT= SKIP
			3	1= VERIFY= YES
			4	1= 2 I/O areas
			5	1= WLRERR= address (fixed length and variable records) 1= Output file (undefined length records)
			6	1= Fixed-length records 0= Variable or undefined length records
			7	Control parameter specified
	101-103	65-67		Address of user's read error routine
	104-111	68-6F		Seek CCW
	112-119	70-77		Search ID Equal CCW
	120-127	78-7F		TIC CCW
	128-135	80-87		Read/Write Data CCW

This is the end of the common portion of the DTFSD table. The following sections are a added depending on the parameters specified in the operand of the DTFSD macro instruction.

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function	Record Format
If RECFORM= FIXBLK and TRUNCS= YES					Fixed length record input files
	136-143	88-8F		Read count CCW	
	144-151	90-97		Count field input area	
	If CONTROL= YES, the following section is added				
	152-167	98-A7		Control CCB	
	168-175	A8-AF		Control CCW	
If UPDATE= YES					
	136-143	88-8F		Search ID Equal CCW	
	144-151	90-97		TIC CCW	
	152-159	98-9F		Verify CCW	
	If CONTROL= YES, the following section is added				
	160-175	A0-AF		Control CCB	
	176-183	B0-B7		Control CCW	
If RECFORM= FIXBLK, TRUNCS= YES and UPDATE= YES					
	136-143	88-8F		Read Count CCW	
	144-151	90-97		Search ID Equal CCW	

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function	Record Format
If RECFORM=FIXBLK, TRUNCS= YES and UPDATE= YES (Cont'd)					Fixed record length input files
	152-159	98-9F		TIC CCW (Bytes 158-159 contain saved block length if two files are using same logic module)	
	160-167	A0-A7		Verify CCW	
	168-175	A8-AF		Count field input area	
	If CONTROL= YES, the following section is added				
	176-191	B0-BF		Control CCB	
	192-199	C0-C7		Control CCW	
If TRUNCS or UPDATE are not specified, no additions are made to the DTFSD table except when CONTROL= YES is specified, the following section is added.					
	136-151	88-97		Control CCB	
	152-159	98-9F		Control CCW	
	136-143	88-8F		Search ID Equal CCW	Fixed record length output files
	144-151	90-97		TIC CCW	
	152-159	98-9F		Verify CCW	
	If CONTROL is not specified				
	160-163	A0-A3		End-of-extent routine address (primarily used by COBOL compiler)	
	If CONTROL= YES				
	160-175	A0-AF		Control CCB	
	176-183	B0-B7		Control CCW	
	184-187	B8-BB		End-of extent routine address (primarily used by COBOL compiler)	
	136-143	88-8F		Read count CCW	Variable length record, undefined length record, spanned record input files
If UPDATE is not specified:					
	144-151	90-97		Count field input area	
	If CONTROL= YES *				
	152-167	98-A7		Control CCB	
	168-175	A8-AF		Control CCW	
	176-179	B0-B3		Logical record length	
	180-183	B4-B7		RX type instruction	

* These bytes are always generated when spanned processing is specified

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function	Record Format
If UPDATE is not specified: (Cont'd)					Variable length record, undefined length record, spanned record input files
	If CONTROL= YES (Cont'd)				
	184	B8	0	Not used	
			1	1= Skip segment	
			2	1= Spanned first time	
			3-7	Not used	
	185-187	B9-BB		Pointer in logical record	
If UPDATE= YES:					Variable length record and undefined length record input files
	144-151	90-97		Search ID Equal CCW	
	152-159	98-9F		TIC CCW	
	160-167	A0-A7		Verify CCW	
	168-175	A8-AF		Count field input area	
	176-183	B0-B7		Count field save area if one I/O area	
	184-191	B8-BF		Count field save area if two I/O areas	
	If CONTROL= YES *				
	192-207	C0-CF		Control CCB	
	208-215	D0-D7		Control CCW	
	216-219	D8-DB		Logical record length	Variable length spanned record update files
	220-223	DC-DF		RX type instruction	
	224	E0	0	Not used	
			1	1= Skip segment	
			2	1= Spanned first time	
			3	1= Null segment	
			4	1= Spanned PUT return	
			5	Not used	
			6	Not used	
			7	1= No update	
	225-227	E1-E3		Pointer in logical record	
	228-235	E4-EB		Count save area	
	236-239	EC-EF		Extent status save area	
	136-143	88-8F		Search ID Equal CCW	Variable length record output files
	144-151	90-97		TIC CCW	
	152-159	98-9F		Verify CCW	
	160-163	A0-A3		Space remaining in output area	
	164-165	A4-A5		Track capacity	
	166-169	A6-A9		Instruction to load user's register VARBLD (If VARBLD is not specified, instruction is NO-OP)	

* These bytes are always generated when spanned processing is specified

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function	Record Format
	If CONTROL= YES *				Variable length record output files
	170-172	AA-AC		Not used	
	173-175	AD-AF		End-of-extent routine address (primarily used by COBOL compiler)	
	176-191	B0-BF		Control CCB	
	192-199	C0-C7		Control CCW	
	200-203	C8-CB		Logical record length	Variable length spanned record output files
	204-207	CC-CF		RX type instruction	
	208	D0	0 1 2 3 4 5 6 7	Not used Not used 1= Leading segment 1= Output block truncated 1= End of track 1= Track truncated 1= Save count 1= Volume spanned	
	209-211	D1-D3		Pointer in logical record	
	212-219	D4-DB		Count save area	
	220-223	DC-DF		Extent status save area	
	136-143	88-8F		Search ID Equal CCW	Undefined length record output files
	144-151	90-97		TIC CCW	
	152-159	98-9F		Verify CCW	
	160-161	A0-A1		Track capacity	
	If CONTROL= YES				
	162-164	A2-A4		Not used	
	164-167	A4-A7		End-of-extent routine address (primarily used by COBOL compiler)	
	168-183	A8-B7		Control CCB	
	184-191	B8-BF		Control CCW	

* These bytes are always generated when spanned processing is specified

DTFSD (Workfiles)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0-15	00-0F		Command Control Block (CCB)
	16	10	0-1	Not used
			2	1= File assigned 'IGN' (COBOL)
			3	1= Track hold option specified
			4	1= DTF relocated by OPENR
			5-7	Not used
	17-19	11-13		Address of logic module
	20	14		DTF type for OPEN/CLOSE (X'20' = sequential access DASD files)
	21	15	0	0= Disk device
			1	1= CLOSE macro is not to delete Format 1 and Format 3 file labels
			2	1= Work file
			3	Type of open : 1= Point; 0= Normal
			4	1= Routine entered from close routine
			5	1= File opened; 0= File closed
			6	Not used
			7	1= Re-entry to close routine
	22-28	16-1C		Filename (DTF name)
	29	1D		Device type Code : X'00'= 2311 X'01' =2314, 2319 X'04'= 3330-1, 2 X'05'= 3330-11 X'07'= 3350 X'08'= 3340 general X'09'= 3340 35MB X'0A'= 3340 70MB Note : In previous versions, last byte of filename contains device type code
	30-31	1E-1F		Track capacity counter
	32-35	20-23		Address of Format 1 label in VTOC (CCHR)
	36	24		Extent sequence number
	37	25		Open communication byte
			0	Not used
			1	1= Device supports RPS
			2	Not used
			3	1= symbolic unit in DTF
			4	1= next extent on new volume
			5	1= extent opened
			6	Not used
			7	1= DTF has been extended into the partition virtual area
	38	26		Lower head limit
	39	27		Upper head limit
&Filename.L	40-41	28-29		Record length
	42-45	2A-2D		Initial extent lower limit
	46-49	2E-31		Current extent lower limit
	50-53	32-35		Extent upper limit

DTF Assembly Label	Bytes		Bits	Function
	Dec	Hex		
&Filename.S	54-55	36-37		Seek address (BB= X'0000')
	56-59	38-3B		Search address (CCHH)
	60	3C		Record number
	61	3D		Switch byte used by logic module
			0	1= First write entry indicator
			1	1= Write update indicator
			2	1= POINTS macro issued
			3	Not first record of a track (RECFORM= UNDEF)
			4	1= Track upper limit reached
			5	Not used
			6	1= Check after read/write
			7	Not used
	62-63	3E-3F		Maximum record length
	64	40		Verify chain bit
	65-67	41-43		Address of user's EOF routine
	68	44		Logical indicators
			0	1= ERROPT= address
			1	1= ERROPT= IGNORE
			2	1= Fixed-length unblocked records
			3	1= Verify specified
			4	1= ERROPT= SKIP
			5	1= Re-read after read error
			6-7	Not used
	69-71	45-47		Address of user read/write error routine
	72-143	48-8F		CCW chain for work files
	144-151	90-97		Input area for Verify CCW and Read Count CCW

DTFDA

DTF Assembly Label	Module DSECT Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	IJICCB	0-15	00-0F		Command Control Block (CCB)
	IJIMOD	16	10	0	1= Trailer labels
				1	Used by FREE macro
				2	1= COBOL Open/Ignore option
				3	1= Track hold option specified
				4	1= DTF relocated by OPENR
				5	Not used
				6	1= SPNUNB
				7	Used by CNTRL macro
		17-19	11-13		Address of logic module
		20	14		DTF type for OPEN/CLOSE (X'22'= direct access files)
	IJISWI	21	15	0	1= Output; 0= Input
				1	1= Verify option specified
				2	1= Search multiple track (SRCHM) specified
				3	1= WRITE AFTER or WRITE RZERO macro used
				4	1= IDLOC specified
				5	1= Undefined; 0 = FIXUNB, VARUNB or SPNUNB
				6	1= RELTYPE= DEC
				7	1= End of file
	IJIFNM	22-28	16-1C		Filename (DTF name)
	IJIDVTP	29	1D		Device type code: X'05'=3330-11 X'00'=2311 X'07'=3350 X'01'=2314,2319 X'08'=3340 general X'02'=2321 X'09'=3340 35MB X'04'=3330-1,-2 X'0A'=3340 70MB
	IJIUNT	30-31	1E-1F		Starting logical unit address of the first volume containing the data file. This value is supplied by the OPEN from EXTENT cards (can be initially zero)
	IJIRPS	32	20	0	Not used
				1	1= Device supports RPS
				2-6	Not used
				7	1= DTF has been extended into the partition virtual area
	IJIULB	33-35	21-23		Address of user's label routine
	IJIUXT	36-39	24-27		Address of user's routine for processing EXTENT information
	IJIRELPT	40	28		Pointer to relative address area: $\frac{\&Filename.P - \&Filename}{2}$
	IJIERC	41-43	29-2B		Address of a 2-byte field in which IOCS can store the error condition or status codes
	IJITST	44-45	2C-2D		Macro code switch for internal use: X'0000'= READ ID X'0001' =READ KEY X'0002'= WRITE ID X'0003'= WRITE KEY X'0004'= WRITE RZERO X'0005'= WRITE AFTER

DTFDA (....Cont'd)

DTF Assembly Label	Module DSECT Label	Bytes Dec	Bytes Hex	Bits	Function
	IJIBPT	46-47	2E-2F		Pointer to channel program build area (&Filename.B) minus 32
	IJICB2	48-63	30-3F		Control seek CCB
&Filename.Z	IJICCW	64-71	40-47		Control Seek CCW for overlap seek routine
	IJIXMD	72-75	48-4B		Channel program builder instruction: XI 36(2),C'0'
	IJIMSZ	76-77	4C-4D		Maximum data length for FIXUNB or UNDEF records; BLKSIZE for VARUNB or SPNUNB records
	IJISPT	78	4E		Pointer to READ ID string (File name.0); X'00' if no READ ID issued
		79	4F		Pointer to READ KEY string (File name.1); X'00' if no READ KEY issued
		80	50		Pointer to WRITE ID string (File name.2); X'00' if no WRITE ID issued
		81	51		Pointer to WRITE KEY string (File name.3); X'00' if no WRITE KEY issued
		82	52		Pointer to WRITE RZERO string (Filename.4); X'00' if no WRITE RZERO issued
		83	53		Pointer to WRITE AFTER string (Filename.5); X'00' if no WRITE AFTER issued
	IJITRK	84-85	54-55		Track constant: 2311: H'0' if key length=0 H'20' if key length≠0 2314/2319: H'0' if key length=0 H'45' if key length≠0 3330: H'135' if key length=0 H'191' if key length≠0 3340 : H'167' if key length=0 H'242' if key length≠0 3350: H'185' if key length=0 H'267' if key length≠0 2321 : H'0' if key length=0 H'16' if key length≠0
	IJIRIC	86-87	56-57		2311 : H'61' 2314/2319: H'101' 3330 : H'135' 3340 : H'167' 3350: H'185' 2321 : H'84'
	IJILAT	88	58	0 1 2 3 4	Not used 1= Wrong-length record 1= non data transfer error Not used 1= no room found

DTF Assembly Label	Module DSECT Label	Bytes Dec	Bytes Hex	Bits	Function
		88 (Cont'd)		5-6 7	Not used 1= Record out of extent area
		89	59	0 1 2 3 4 5 6 7	1= Data check in count area 1= Track overrun 1= End of cylinder 1= Data check when reading key or data 1= No record found 1= End of file 1= End of volume Not used
	IJILBTK	90-95	5A-5F		Label track address, XBCCHH, where X is the volume sequence number of the device on which the label track is located.
The following section is included if UNDEF, AFTER or RZERO is specified					
&Filename.L	IJILST	96-143	60-8F		Basic CCW's to build channel program
		144-183	90-B7		Basic CCW's for undefined length or formatting macros
	IJIVIT	184-185	B8-B9		Instruction to give record length to user if record length is undefined (NOPR 0 if no RECSIZE specified)
	IJIFRU	186-187	BA-BB		Instruction to get record length from user if record length is undefined. (NOPR 0 if no RECSIZE specified)
&Filename.F	IJIFLD	188-192	BC-C0		Work area (used for R0 address - CCHH0)
&Filename.K	IJICNT	193-200	C1-C8		Work area (used for R0 data field)
&Filename.C	IJICTS	201-208	C9-D0		Work area (included only for spanned or variable records for record count field)
The channel program builder strings are generated following the DTFDA table and preceding the channel program building area					
&Filename.0		Variable			Channel program builder string for READ ID macro. If READ ID is not specified, the string is not generated
&Filename.1		Variable			Channel program builder string for READ KEY macro. If READ KEY is not specified, the string is not generated
&Filename.2		Variable			Channel program builder string for WRITE ID macro. If WRITE ID is not specified, the string is not generated

DTF Assembly Label	Module DSECT Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.3		Variable			Channel program builderstring for WRITE KEY macro. If WRITE KEY is not specified, the string is not generated
&Filename.4		Variable			Channel program builder string for WRITE RZERO macro. If WRITE RZERO or WRITE AFTER is not specified, the string is not generated
&Filename.5		Variable			Channel program builder string for WRITE AFTER macro. If WRITE RZERO or WRITE AFTER is not specified, the string is not generated
The following section contains the channel program build areas and varies in size					
&Filename.B		0-7	00-07		Seek CCW that is generated at program assembly time and used by all channel programs
		Variable			Area to build: 1) Eight CCW's if AFTER is not specified 2) Eight CCW's if spanned or variable length records and AFTER= YES is specified 3) Seven CCW's if undefined or fixed records and AFTER= YES is specified
		Variable			Area to build: 1) Eight CCW's if AFTER is not specified and VERIFY= YES is specified 2) Eight CCW's if spanned or variable length records and AFTER= YES and VERIFY= YES are specified 3) Five CCW's if undefined or fixed records and AFTER= YES and VERIFY= YES are specified
The following section is added for spanned records only					
		8 bytes			Count save area
		8 bytes			SEEKADR save area
		1 byte		0 1 2 3 4 5-7	1= Relative addressing 1= IJIGET switch on 1= Ignore hold switch on 1= Reserved for use by DAMODV 1= New volume SEEKADR Not used

DTFDA (....Cont'd)

DTF Assembly Label	Module DSECT Label	Bytes Dec	Bytes Hex	Bits	Function
		1 byte			Reserved
		2 bytes			Record size
		12 bytes			Work area
		8 bytes			Control word save area
The following section is added to the DTFDA table if DSKXTNT (relative addressing) is specified					
&Filename.P		3 bytes			3X'00' for padding
&Filename.I		5 bytes			IDLOC record area (bucket used by module)
&Filename.S		8 bytes			SEEKADR in the form: M,B1,B2,C1,C2,H1,H2,R
		4 bytes			DC A(&SEEKADR)
		4 bytes			DC A(&IDLOC)
		8 bytes			Work area for RELTYPE=DEC
&Filename.X		4 bytes			Save area for CCHH portion of actual DASD address
		4 bytes			Alteration factor for C1 in SEEK ADR (see bytes 112-119) 2311 : X'00000001' 2314/2319: X'00000001' 3330 : X'00001300' 3340 : X'00000C00' 3350 : X'00001E00' 2321 : X'000003E8'
		4 bytes			Alteration factor for C2 in SEEK ADR (see bytes 112-119) 2311 : X'0000000A' 2314/2319: X'00000014' 3330 : X'00000013' 3340 : X'0000000C' 3350 : X'0000001E' 2321 : X'00000064'
		4 bytes			Alteration factor for H1 in SEEK ADR (see bytes 112-119) 2311 : X'00000001' 2314/2319: X'00000001' 3330 : X'00000001' 3340 : X'00000001' 3350 : X'00000001' 2321 : X'00000014'
		Variable to end of DTF table			DSKXTNT table composed of a variable number of 8-byte entries containing extent information in the following format: Bytes 0-2 TTT2 - cumulative number of tracks in the DSK XTNT table entries up to and including the current entry 3 M- volume sequence number 4 B - bin number (0 for disk devices) 5-7 TTT1 - relative track number of lower limit of this entry A 2-byte end-of-table indicator containing X'FFFF' follows the last entry in the DSKXTNT table

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		Command Control Block (CCB)
	16	10	0	Used by ISAM Interface Program
			1	Not used
			2	1= COBOL open; ignore option
			3	Not used
			4	1= DTF table address constants relocated by OPENR
			5	Not used
			6	1= Data set security
			7	1= Wrong blocksize error during file extension
	17-19	11-13		Address of logic module
	20	14		File type for OPEN/CLOSE (X'24'= LOAD)
	21	15		Option byte:
			0	1= 2321 (Version 1-2 only)
			1	Not used
			2	1= Cylinder overflow option
			3	Not used
			4	1= Blocked records (used by previous versions)
			5	1= Verify
			6	1= Indexes on 2321 (Version 1-2 only)
			7	1= 2 I/O areas present
	22-28	16-1C		File name
	29	1D		Prime data device type indicator: X'00'= 2311 ; X'01'= 2314/2319 ; X'02'= 2321 ; X'04'= 3330 ; X'08'= 3340 general; X'09'= 3340 35MB; X'0A'= 3340 70MB
&Filename.C	30	1E		Status byte:
			0	1= Uncorrectable DASD error (except WLR)
			1	1= WLR error
			2	1= Prime data area full
			3	1= Cylinder index area not large enough to reference prime data area. Set on only if error detected at SETFL time
			4	1= Master index not large enough to reference prime data area. Set on only if error detected at SETFL time
			5	1= Duplicate record
			6	1= Sequence error
			7	1= No EOF record written in prime data area
	31	1F		High level index device type indicator: X'00'= 2311 ; X'01'= 2314/2319 ; X'02'= 2321 ; X'04'= 3330 ; X'08'= 3340 general ; X'09'= 3340 35MB ; X'0A'= 3340 70MB
	32	20		Relative position of the DSKXTN (logical unit, cell number) table (in words). This value is the length of the DTF table divided by 4

DTF Assembly Label	Bytes		Bits	Function
	Dec	Hex		
	33-34	21-22		First prime track in cylinder (HH)
	35	23		First prime data record in cylinder (R)
	36-37	24-25		Last prime track in cylinder (HH)
	38	26		High record on master index/cylinder index track (R)
	39	27		High record on prime data track (R)
	40	28		High record on overflow track (R)
	41	29		High record on last track index track in cylinder (whether shared or unshared)
	42	2A		High record on track index track other than last in cylinder. If only one track index track in cylinder, it is equal to byte 41
	43	2B		Condition code :
			0	1= WLR checks requested (for extension)
			1	1= First record in file
			2	1= Prime data extent full
			3	1= Master index/cylinder index extent too small
			4	1= Prime data upper limit has been increased (for extension)
			5	1= Extension
			6-7	Not used
	44-50	2C-32		Prime data lower limit (MBBCCHH)
	51-57	33-39		Cylinder index lower limit (MBBCCHH)
	58-64	3A-40		Master index lower limit (MBBCCHH)
	65	41		Number of index levels
			0-3	Not used
			4	1= RPS type device (data)
			5	1= RPS type DTF
			6	1= Master index
			7	1= RPS type device (index)
&Filename.H	66-73	42-29		Address of last prime data record (MBBCCHHR)
	74-75	4A-4B		Logical record length
	76-77	4C-4D		Key length
	78-79	4E-4F		Block length (logical record length times number of records)
	80-81	50-51		Overflow record length (logical record length plus 10)
	82-83	52-53		Blocking factor (number of logical records)
	84-85	54-55		Index entry length (key length plus 10)
	86-87	56-57		Prime data record length (key length plus physical record length)
	88-89	58-59		Overflow record length with key (key length plus logical record length plus 10)
	90-91	5A-5B		Prime data record format length (key length plus physical record length plus 8)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	92-93	5C-5D		Overflow record format length (key length plus logical record length plus 18)
	94-95	5E-5F		Key location (in blocked records)
This is the end of the common DTF area. The format of the remainder of the table is variable and is generated according to the parameters specified in the DTFIS macro instruction				
&Filename.S	96-103	60-67		Seek/search address area (MBBCCHHR)
&Filename.P	104-105	68-69		Logical record counter (for blocking)
	106-107	6A-6B		Number of bytes for high level index
	108-111	6C-6F		Prime data record counter (logical records)
	112	70		Status indicators:
			0-1	Not used
			2	1= File closed
			3-5	Not used
			6	1= Last prime data track full
			7	1= Last block full
	113-117	71-75		Last track index normal entry address (CCHHR)
	118-122	76-7A		Last cylinder index entry address (CCHHR)
	123-127	7B-7F		Last master index entry address (CCHHR)
&Filename.B				CCW build area. See description of SETFL macro, phase 1 - $$BSETFL
	128-135	80-87		Seek CCW
	136-143	88-8F		Search ID Equal CCW
	144-151	90-97		TIC CCW
	152-159	98-9F		Read/Write CCW
	160-167	A0-A7		Search ID Equal CCW
	168-175	A8-AF		TIC CCW
	176-183	B0-B7		Verify CCW
&Filename.M	184-187	B8-BB		Address of IOREAL
	188-191	BC-BF		Address of data in WORKL. (FIXBLK= address of WORKL; FIXUNB= address of WORKL plus key).
	192-195	C0-C3		Address of key in WORKL. (FIXBLK= address of WORKL plus KEYLOC minus 1; FIXUNB= address of WORKL.)
	196-199	C4-C7		Block position indicator (address of logical record in IOAREAL)
	200	C8		Master index, extension indicator:
			0-2	Not used
			3	1= Extending file; 0= Creating file
			4-6	Not used
			7	1= Master index being used; 0= No master index being used

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	201-204	C9-CC		Cylinder index upper limit (CCHH)
	205-208	CD-D0		Master index upper limit (CCHH)
	209-215	D1-D7		Prime data upper limit (old upper limit if extension) (MBBCCHH)
	216-222	D8-DE		Prime data new upper limit (for extension) (MBBCCHH)
	223	DF		Last prime data track in cylinder minus 1
	224-225	E0-E1		Key length minus one
	226-227	E2-E3		Logical record length minus 1
	228-229	E4-E5		Address of track index dummy record (HR)
	230-231	E6-E7		Address of record before first prime data record in cylinder (HR)
	232	E8		Number of records on master index/cyl inder index track minus 1
	233-236	E9-EC		Master index/cyl inder index DASD address control field (CCHH): 2311 = X'00C70009' 2314/2319= X'00C70013' 2321 = X'13090413' 3330 = X'01FF0012' 3340 = X'01FF000B'(35MB), X'02FF000B'(70MB)
	237-239	ED-EF		Prime data address control field (CCH): 2311 = X'00C700' 2314/2319= X'00C700' 2321 = X'130904' 3330 = X'01FF00' 3340 = X'01FF00'(35MB), X'02FF00'(70MB)
	240-242	F0-F2		Prime data beginning of volume (CCH): 2311 = X'000100' 2314/2319= X'000100' 2321 = X'000001' 3330 = X'000100' 3340 = X'000100'
	243-245	F3-F5		Prime data end of volume (CCH): 2311 = X'00C700' 2314/2319= X'00C700' 2321 = X'130504' 3330 = X'019300' 3340 = X'015800'(35MB), X'02B700'(70MB)
	246-247	F6-F7		Used for alignment
	248-251[1]	F8-FB		First entry in DSKXTN table (logical unit, cell number)
	256-259[2]	100-103		X'FFFFFFFF'= End of DSKXTN table
	260-263	104-107		Address of IOAREA2
	264-267	108-10B		Address used to relocate IOAREA2

1) Each entry in the DSKXTN table is 4 bytes long. The minimum number of entries is two. There is one entry per extent.
2) Location of the end-of-table indicator depends on length of DSKXTN table

DTFIS (Add) - part 1

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		CCB
	16	10	0	Used by ISAM Interface Program
			1	Not used
			2	1= COBOL open; ignore option
			3	1= Track hold specified
			4	1= DTF table address constants relocated by OPENR
			5	Not used
			6	1= Data set security
			7	1= Wrong blocksize error during addition to file
	17-19	11-13		Logic module address
	20	14		File type for OPEN/CLOSE (X'25'= ADD)
	21	15		Option byte:
			0	1= 2321 (Version 1-2 only)
			1	1= Prime data in core
			2	1= Cylinder overflow
			3	1= Cylinder index in core
			4	1= Blocked records
			5	1= Verify
			6-7	Not used
	22-28	16-1C		DTF file name
	29	1D		Prime data device type indicator: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
&Filename.C	30	1E		Status byte:
			0	1= Uncorrectable DASD error (except WLR)
			1	1= WLR error
			2	1= EOF (sequential)
			3	1= No record found
			4	1= Illegal ID specified
			5	1= Duplicate record sensed
			6	1= Overflow area full
			7	1= Record retrieved from overflow area
	31	1F		Highest level index device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
	32	20		Relative position of the DSKXTN (logical unit, cell number) table (in words). This value is the length of the DTF table divided by 4
	33-35	21-23		First prime data record in cylinder (HHR)
	36-37	24-25		Last prime data track in cylinder (HH)
	38	26		High record number on master index/cylinder index track (R)

DTFIS (Add) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	39	27		High record number on prime data track (R)
	40	28		High record number on overflow track (R)
	41	29		High record number on shared track (R)
	42	2A		High record number on track index (TI) track (R)
	43	2B		Retrieval byte :
			0	1= WORKR area specified
			1	1= WORKS area specified
			2	Overflow switch
			3	1= Read
			4	Not used
			5	1= Output
			6	1= Write key
			7	1= PUT macro issued
	44-50	2C-32		Prime data lowet limit (MBBCCHH)
	51-57	33-39		Cylinder index lower limit (MBBCCHH)
	58-64	3A-40		Master index lower limit (MBBCCHH)
	65	41		Index level number, WAITF indicator :
			0	1= From WAITF routine
			1	1= WAITF seek check bit
			2-3	Not used
			4	1= RPS type device (data)
			5	1= RPS type DTF
			6	1= Master index
			7	1= RPS type device (index)
	66-73	42-49		Last prime data record address (MBBCCHHR)
	74-75	4A-4B		Logical record length (RECSIZE)
	76-77	4C-4D		Key length (KEYLEN)
	78-79	4E-4F		Block size (logical record length times number of records)
	80-81	50-51		Overflow record length (logical record length plus 10)
	82-83	52-53		Blocking factor (number of logical records in block (NRECDS)
	84-85	54-55		Index entry length (key length plus 10)
	86-87	56-57		Prime data record length (key length plus physical record length (block size)
	88-89	58-59		Overflow record length plus key (key length plus logical record length plus 10)
	90-91	5A-5B		Prime data record format length (key length plus blocksize plus 8)
	92-93	5C-5D		Overflow record format length (key length plus logical record length plus 18)
	94-95	5E-5F		Key location (KEYLOC) for blocked records
	96-97	60-61		Constant = 5
	98-99	62-63		Constant = 10

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	100-101	64-65		Displacement of part 2 of the DTFIS table from start of part 1
	102-103	66-67		Displacement of part 3 of the DTFIS table from start of part 1
&Filename.S	104-113	68-71		Seek/search address area (MBBCCHHRFP)
&Filename.W	114-123	72-7B		Random/sequential retrieval work area
&Filename.P	124-127	7C-7F		Prime data record count
	128	80		Status indicators:
			0-1	Not used
			2	1= File closed
			3-5	Not used
			6	1= Last prime data track full
			7	1= Block complete
	129-133	81-85		Last track index normal entry address (CCHHR)
	134-138	86-8A		Last cylinder index entry address (CCHHR)
	139-143	8B-8F		Last master index entry address (CCHHR)
	144-151	90-97		Last independent overflow record address (MBBCCHHR)
&Filename.I	152-153	98-99		Number of independent overflow tracks
&Filename.A	154-155	9A-9B		Number of full cylinder overflow areas
&Filename.O	156-157	9C-9D		Overflow record count
	158-164	9E-A4		Independent overflow area lower limit (MBBCCHH)
	165-171	A5-AB		Independent overflow area upper limit (MBBCCHH)
	172-175	AC-AF		A(&Filename.D) - Address of work area for cylinder overflow control record (COCR)
	176-179	B0-B3		A(&Filename.D+8) - Address of workarea for the current track index normal entry count field
	180-183	B4-B7		A(&Filename.D+16) - Address of work area for current track index overflow entry count field
	184-187	B8-BB		A(&Filename.D+24) - Address of workarea for current prime data record count field
	188-191	BC-BF		A(&Filename.D+32) - Address of work area for current overflow record count field
	192-195	C0-C3		A(&Filename.D+40) - Address of work area for track index normal entry data field
	196-199	C4-C7		A(&Filename.D+50) - Address of work area for current overflow record linkage field
	200-203	C8-CB		A(&IOREAL) - Address of IOREAL, the I/O area used for adding records to a file

DTFIS (Add) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	204-207	CC-CF		A(&WORKL) - Address of WORKL, workarea containing user data records to be added to the file
	208-211	D0-D3		A(&Filename.K) - Address of the ADD key area
	212-215	D4-D7		A(&IOAREAL+8) - Address of key position in IOAREAL
	216-219	D8-DB		A(&IOAREAL+8+&KEYLEN) - Address of data position in IOAREAL

DTFIS (Add) - part 2

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.2	0 -3	00-03		A(&Filename.S+3) - Address of the seek/search address area plus 3
	4	04	0	1= Seek check indicated
			1-5	Not used
			6	1= Over/under seek has occurred
			7	1= An error has been found, but a seek check is indicated
	5 -7	05-07		A(&Filename.W) - Address of random/sequential retrieval work area
The following information is generated if the cylinder index in core option is specified				
	12-15	0C-0F		A(&INDAREA) -Starting address of main storage area specified for cylinder index
	16-17	10-11		AL2(&INDSIZE) - Number of bytes in main storage available for cylinder index
	18-25	12-19		Next cylinder index entry to be read (MBBCCHHR)
	26-30	1B-1E		Last cylinder index entry (CCHHR)
	31	1F		Core index byte:
			0	1= First time through B-transient, $$BINDEX
			1	1= End of cylinder index reached
			2	1= Index skip option specified
			3	1= Suppress in-core option and read cylinder index
			4-7	Not used
	32-35	1D-23		Pointer to key (stored by module)

DTFIS (Add) - part 2 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
The following information is generated if the prime data in core add function is specified This information is aligned on a double word boundary				
	36-37	24-25		Size of IOAREAL
	38-39	26-27		Maximum number of prime data records in main storage
	40-43	28-2B		Address of write CCW's
	44-47	2C-2F		Address of read CCW's
	48	30		Switch byte:
			0	1= EOF
			1-7	Not used
	49	31		Reserved.
IJHDCWRK	50-51	32-33		Work field for I/O Module.

DTFIS (Add) - part 3

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.B	0 -7	00-07		CCW X'07', &Filename.S+1, X'40', 6 - Long seek CCW with command chaining
	8 -127	08-7F		Channel program build area.
&Filename.D	128-135	80-87		Cylinder overflow control record (COCR)
	136-143	88-8F		Current track index normal entry count field address
	144-151	90-97		Current track index overflow entry count field address
	152-159	98-9F		Current prime data record count field address
	160-167	A0-A7		Current overflow record count field address
	168-177	A8-B1		Track index normal entry data field
	178-187	B2-BB		Current overflow record sequence link field
	188-197	BC-B5		Current track index overflow entry data field
	198	C6		X'01' - Add to EOF X'02' - Add to independent overflow area
	199-201	C7-C9		Overflow control bytes (CCH)
	202-203	CA-CB		High HR on overflow track
	204-211	CC-D3		Volume upper limit for prime data records (MBBCCHHR)
	212-217	D4-D9		CLC 0 (&KEYLEN, 13),0 (6) - Unblocked CLC 0 (&KEYLEN, 13), &KEYLOC-1 (6) - Blocked Utility CLC for key

DTF Assembly Label	Bytes		Bits	Function
	Dec	Hex		
	218-223	DA-DF		MVC 0 (&KEYLEN, 13), 0 (12) - Unblocked MVC 0 (&KEYLEN, 13), &KEYLEN-1 (12) - Blocked Utility MVC for key
&Filename.E	224-227[1]	E0-E3		First entry in DSKXTN table (logical unit, cell number)
	232-235[2]	E8-EB		4X'FF' - End of DSKXTN table
&Filename.K	236 +	EC-end		Key area for ADD only. Number of bytes depends on key length, KEYLEN

1) Each entry in the DSKXTN table is four bytes long. The minimum number of entries is two. There is one entry per extent.

2) Location of the end-of-table indicator depends on length of DSKXTN table.

DTFIS (RETRVE,RANDOM) - part 1

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		Command Control Block (CCB)
	16	10	0	Used by ISAM Interface Program
			1	1= GET issued
			2	1= COBOL open; ignore option
			3	1= HOLD option specified
			4	1= DTF table address constants relocated by OPENR
			5-6	Not used
			7	1 = Different blocksize in format
	17-19	11-13		Address of logic module
	20	14		File type for OPEN/CLOSE (X'26'= RETRVE)
	21	15		Option byte:
			0	1= 2321 (Version 1-2 only)
			1	1= Prime data in core
			2	1= Cylinder overflow option
			3	1= Cylinder index in core option
			4	1= Blocked records
			5	1= Verify
			6-7	Not used
	22-28	16-1C		File name (DTF name)
	29	1D		Prime data device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
&Filename.C	30	1E		Status byte:
			0	1= Uncorrectable DASD error (except WLR error)
			1	1= WLR error
			2	1= EOF (sequential)
			3	1= No record found
			4	1= Illegal ID specified
			5	1= Duplicate record sensed
			6	1= Overflow area full
			7	1= Record retrieved from overflow area
	31	1F		High level index device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
	32	20		Relative position of the DSKXTN (logical unit, cell number) table (in words). This value is the length of the DTF table divided by 4
	33-35	21-23		First prime data record in cylinder (HHR)
	36-37	24-25		Last prime data track in cylinder (HH)
	38	26		High record number on master index/cylinder index track (R)
	39	27		High record number on prime data track (R)

DTFIS (RETRVE, RANDOM) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	40	28		High record number on overflow track (R)
	41	29		High record number on shared track (R)
	42	2A		High record number on track index track (R)
	43	2B		Retrieval byte :
			0	1= WORKR specified
			1	1= WORKS specified
			2	Overflow switch
			3	1= Read key
			4	Not used
			5	1= Output
			6	1= Write key
			7	1= PUT macro issued
	44-50	2C-32		Prime data lower limit (MBBCCHH)
	51-57	33-39		Cylinder index lower limit (MBBCCHH)
	58-64	3A-40		Master index lower limit (MBBCCHH)
	65	41		Index level number, WAITF, and track hold indicators :
			0	1= From WAITF routine
			1	1= Seek check from WAITF
			2	1= Index track held
			3	1= Data track held
			4	1= RPS type device (data)
			5	1= RPS type DTF
			6	1= Master index
			7	1= RPS type device (index)
	66-73	42-49		Last prime data record address (MBBCCHHR)
	74-75	4A-4B		Logical record length
	76-77	4C-4D		Key length
	78-79	4E-4F		Block size (logical record length times number of records)
	80-81	50-51		Overflow record length (logical record length plus 10)
	82-83	52-53		Blocking factor
	84-85	54-55		Index entry length (key length plus 10)
	86-87	56-57		Prime data record length key (key length plus physical record length)
	88-89	58-59		Overflow record length with key (key length plus logical record length plus 10)
	90-91	5A-5B		Prime data record format length key (key length plus physical record length plus 8)
	92-93	5C-5D		Overflow record format length key (key length plus logical record length plus 18)
	94-95	5E-5F		Key location (blocked records)

DTFIS (RETRVE,RANDOM) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	96-97	60-61		Constant= 5
	98-99	62-63		Constant= 10
	100-101	64-65		Displacement of part 2 of the DTFIS table from part 1
	102-103	66-67		Displacement of part 3 of the DTFIS table from part 1
&Filename.S	104-113	68-71		Seek/search address area (MBBCCHHRFP)
&Filename.W	114-123	72-7B		Random/sequential retrieval work area

DTFIS (RETRVE,RANDOM) - part 2

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.2	0 -3	00-03		Address of seek/search address area plus 3
	4	04	0	1= Seek check indicated
			1-5	Not used
			6	1= Over/under seek has occurred
			7	1= An error has been found, but a seek check is indicated
	5 -7	05-07		Address of random/sequential retrieval work area
	8 -11	08-0B		Address of IOAREAS
	12-15	0C-0F		Address of IOAREAR
	16-19	10-13		Address of KEYARG
	20-23	14-17		Address of WORKR
	24-27	18-1B		Current sequential I/O area address
	28-31	1C-1F		4-byte NO-OP instruction, or L IOREG, * - 4 if IOREG specified.
	32	20		X'00'= No verify; X'40'= Verify
	33	21		X'08'= Unblocked; X'00'= Blocked
	34	22		R= First prime data record on shared track
	35-39	23-27		Upper limit for sequential retrieval (CCHHR)
	40-41	28-29		H'0'= Blocked records H'2'= Overflow record H'8'= Unblocked records
	42	2A		X'C7'= 2311,2314 or 2319; X'09'= 2321; X'FF'= 3330, 3340
	43-47	2B-2F		Initial values for sequential retrieval
&Filename.H	48-55	30-37		Current DASD address for sequential (MBBCCHHR)
	56-63	38-3F		Current overflow DASD address for sequential (MBBCCHHR)

DTFIS (RETRVE,RANDOM) - part 2 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	64-65	40-41		Sequential record counter
	66-67	42-43		Current track index entry for sequential (HR)
&Filename.T	68-69	44-45		Number of records tagged for deletion
	70-71	46-47		Load IOREG for random retrieval
&Filename.G	72-79	48-4F		DASD address save area (MBBCCHHR)
	80-83	50-53		Record pointer within I/O area for write operation
&Filename.R	84-87	54-57		Nonfirst overflow record count
The following information is generated when the cylinder index in core option is specified				
	92-95	5C-5F		A(&INDAREA) - Starting address of main storage area specified for cylinder index
	96-97	60-61		AL2(&INDSIZE) - Number of bytes in main storage available for cylinder index
	98-105	62-69		Next cylinder index entry to be read (MBBCCHHR) (Initialized by $$BINDEX to cylinder index starting address)
	106-110	6A-6E		Last cylinder index entry
	111	6F		Core index byte:
			0	1= First time through transient
			1	1= End of index reached
			2	1= Index skip option
			3-7	Not used
	112-115	70-73		Pointer to key (stored by the module)
	116-131	74-83		Reserved

DTFIS (RETRVE,RANDOM) - part 3

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.B	0 -7	00-07		X'07',&Filename.S+1,X'40',6 - Long seek CCW with command chaining
	8 -63	08-3F		Area to build CCW-string
	64-67[1]	40-43		First entry in DSKXTN table (logical unit,cell number)
	72-75[2]	48-4B		4X'FF' End of DSKXTN table

1) The length of one entry is the four bytes shown here. The minimum number of entries is 2. There is one entry per extent.

2) The location of the end-of-table indicator depends on the length of DSKXTN table.

DTFIS (RETRVE,SEQNTL) - part 1

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		Command Control Block (CCB)
	16	10	0	Used by ISAM Interface Program
			1	1= GET issued
			2	1= COBOL open; ignore option
			3	1= Track Hold specified
			4	1= DTF table address constants relocated by OPENR
			5	1= EOF on sequential retrieve
			6	1= Data set security
			7	1= Different blocksize in format 1 label than in DTFIS
	17-19	11-13		Address of logic module
	20	14		File type for OPEN/CLOSE (X'26'= RETRVE)
	21	15		Option byte:
			0	1= 2321 (Version 1-2 only)
			1	1= Prime data in core
			2	1= Cylinder overflow option
			3	1= Cylinder index in core option
			4	1= Blocked records
			5	1= Verify
			6	1= IOAREAS just used; 0= IOAREA2 just used
			7	1= 2 I/O areas present
	22-28	16-1C		File name (DTF name)
	29	1D		Prime data device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
&Filename.C	30	1E		Status byte:
			0	1= Uncorrectable DASD error (except WLR error)
			1	1= WLR error
			2	1= EOF (sequential)
			3	1= No record found
			4	1= Illegal ID specified
			5	1= Duplicate record sensed
			6	1= Overflow area full
			7	1= Record retrieved from overflow area
	31	1F		High level index device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
	32	20		Relative position of the DSKXTN (logical unit, cell number) table (in words). This value is the length of the DTF table divided by 4)
	33-35	21-23		First prime data record in cylinder (HHR)
	36-37	24-25		Last prime data track in cylinder (HH)

DTF Assembly Label	Bytes		Bits	Function
	Dec	Hex		
	38	26		High record number on master index/cylinder index track (R)
	39	27		High record number on prime data track (R)
	40	28		High record number on overflow track (R)
	41	29		High record number on shared track (R)
	42	2A		High record number on track index track (R)
	43	2B		Retrieval byte :
			0	1= WORKR specified
			1	1= WORKS specified
			2	Overflow switch
			3	1= Read key
			4	1= First record being processed (after issuing SETL macro)
			5	1= Output
			6	1= Write key
			7	1= PUT macro issued
	44-50	2C-32		Prime data lower limit (MBBCCHH)
	51-57	33-39		Cylinder index lower limit (MBBCCHH)
	58-6	3A-40		Master index lower limit (MBBCCHH)
	65	41		Index level number, WAITF indicator :
			0	1= From WAITF routine
			1	1= WAITF seek check bit
			2-3	Not used
			4	1= RPS type device (data)
			5	1= RPS type DTF
			6	1= Master index
			7	1= RPS type device (index)
	66-73	42-49		Last prime data record address (MBBCCHHR)
	74-75	4A-4B		Logical record length
	76-77	4C-4D		Key length
	78-79	4E-4F		Block size (logical record length times number of records)
	80-81	50-51		Overflow record length (logical record length plus 10)
	82-83	52-53		Blocking factor
	84-85	54-55		Index entry length (key length plus 10)
	86-87	56-57		Prime data record length (key length plus physical record length
	88-89	58-59		Overflow record length with key (key length plus logical record length plus 10)
	90-91	5A-5B		Prime data record format length (key length plus physical record length plus 8)
	92-93	5C-5D		Overflow record format length (key length plus logical record length plus 18)
	94-95	5E-5F		Key location (blocked records)

DTFIS (RETRVE, SEQNTL) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	96-97	60-61		Constant= 5
	98-99	62-63		Constant= 10
	100-101	64-65		Displacement of part 2 of the DTFIS table from part 1
	102-103	66-67		Displacement of part 3 of the DTFIS table from part 1
&Filename.S	104-113	68-71		Seek/search address area (MBBCCHHRFP)
&Filename.W	114-123	72-7B		Random/sequential retrieval work area

DTFIS (RETRVE, SEQNTL) - part 2

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.2	0 -3	00-03		Address of seek/search address area plus 3
	4	04	0 1-5 6 7	1= Seek check indicated Not used 1= Over/under seek has occurred 1= An error has been found, but a seek check is indicated
	5 -7	05-07		Address of random/sequential retrieval work area
	8 -11	08-0B		Address of IOAREAS
	12-15	0C-0F		Address of IOAREA2
	16-19	10-13		Address of KEYARG
	20-23	14-17		Address of WORKR
	24-27	18-1B		Current sequential I/O area address
	28-31	1C-1F		L IOREG, *-4 - Load IOREG or a 4 byte NO-OP instruction
	32	20		X'00'= No verify; X'40'= Verify
	33	21		X'08'= Unblocked records; X'00'= Blocked rec'ds
	34	22		R= First prime data record on shared track
	35-39	23-27		Upper limit for sequential retrieval (CCHHR)
	40-41	28-29		H'0'= Blocked records H'2'= Overflow record H'8'= Unblocked records
	42	2A		X'C7'= 2311, 2314 or 2319; X'09'= 2321; X'FF'= 3330, 3340
	43-47	2B-2F		Initial values for sequential (CCHHR)
&Filename.H	48-55	30-37		Current DASD address for sequential retrieval (MBBCCHHR)
	56-63	38-3F		Current overflow DASD address (MBBCCHHR)

DTFIS (RETRVE, SEQNTL) - part 2 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	64-65	40-41		Sequential record counter
	66-67	42-43		Current track index entry (HR)
&Filename.T	68-69	44-45		Number of records tagged for deletion.
	70-75	46-4B		For boundary alignment.
	76-91	4C-5E		Reserved.

DTFIS (RETRVE, SEQNTL) - part 3

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.B	0 -7	00-07		X'07', &Filename.S+1, X'40',6 - Long seek CCW with command chaining
	8 -63	08-3F		Area to build CCW-string
&Filename.E	64-67[1]	40-43		First entry in DSKXTN table (logical unit, cell number)
	72-75[2]	48-4B		4X'FF' - End of DSKXTN table

1) The length of one entry is the four bytes shown here. The minimum number of entries is 2. There is one entry per extent.

2) The location of the end-of-table indicator depends on the length of DSKXTN table.

DTFIS (ADDRTR) - part 1

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename	0 -15	00-0F		Command Control Block (CCB)
	16	10	0	Used by ISAM Interface Program
			1	1= GET issued
			2	COBOL open; ignore option
			3	1= Track hold option specified
			4	1= DTF table address constants relocated by OPENR
			5	EOF switch
			6	1= Data set security
			7	1= Wrong blocksize error during addition to file
	17-19	11-13		Logic module address
	20	14		File type for OPEN/CLOSE (X'27'= ADDRTR)
	21	15		Option byte:
			0	1= 2321 (Version 1-2 only)
			1	1= Prime data in core
			2	1= Cylinder overflow
			3	1= Cylinder index in core
			4	1= Blocked records

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	21 (Cont'd)		5 6 7	1= Verify 1= IOAREAS just used; 0= IOAREA2 just used 1= 2 I/O areas present
	22-28	16-1C		DTF file name
	29	1D		Prime data device type indicator: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
&filename.C	30	1E	 0 1 2 3 4 5 6 7	Status byte: 1= Uncorrectable DASD error (except WLR error) 1= WLR error 1= EOF (sequential) 1= No record found 1= Illegal ID specified 1= Duplicate record sensed 1= Overflow area full 1= Record retrieved from overflow area
	31	1F		Highest level index device type: X'00'= 2311 X'01'= 2314/2319 X'02'= 2321 X'04'= 3330 X'08'= 3340 general X'09'= 3340 (35MB) X'0A'= 3340 (70MB)
	32	20		Relative position of the DSKXTN (logical unit, cell number) table (in words). This value is the length of the DTF table divided by 4.
	33-35	21-23		First prime data record in cylinder (HHR)
	36-37	24-25		Last prime data track in cylinder (HH)
	38	26		High record number on master index/cylinder index track (R)
	39	27		High record number on prime data track (R)
	40	28		High record number on overflow track (R)
	41	29		High record number on shared track (R)
	42	2A		High record number on track index (T1) track (R)
	43	2B	 0 1 2 3 4 5 6 7	Retrieval byte: 1= WORKR area specified 1= WORKS area specified Overflow switch 1= Read 1= First record being processed (after issuing SETL macro) 1= Output 1= Write key 1= PUT macro issued
	44-50	2C-32		Prime data lower limit (MBBCCHH)

DTFIS (ADDRTR) - part 1 (....Cont'd)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	51-57	33-39		Cylinder index lower limit (MBBCCHH)
	58-64	3A-40		Master index lower limit (MBBCCHH)
	65	41		Index level number, WAITF and track hold indicators :
			0	1= From WAITF routine
			1	1= Seek check from WAITF
			2	1= Index track held
			3	1= Data track held
			4	1= RPS type device (data)
			5	1= RPS type DTF
			6	1= Master index; 0= cylinder index
			7	1= RPS type device (index)
&Filename.H	66-73	42-49		Last prime data record address (MBBCCHHR)
	74-75	4A-4B		Logical record length (RECSIZE)
	76-77	4C-4D		Key length (KEYLEN)
	78-79	4E-4F		Block size (logical record length times number of records)
	80-81	50-51		Overflow record length (logical record length plus 10)
	82-83	52-53		Blocking factor (number of logical records in block (NRECDS))
	84-85	54-55		Index entry length (key length plus 10)
	86-87	56-57		Prime data record length (key length plus physical record length (block size)
	88-89	58-59		Overflow record length with key (key length plus logical record length plus 10)
	90-91	5A-5B		Prime data record format length (key length plus block size plus 8)
	92-93	5C-5D		Overflow record format length (key length plus logical record length plus 18)
	94-95	5E-5F		Key location (KEYLOC) for blocked records
	96-97	60-61		Constant = 5
	98-99	62-63		Constant = 10
	100-101	64-65		Displacement of part 2 of the DTFIS table from start of part 1
	102-103	66-67		Displacement of part 3 of the DTFIS table from start of part 1
&Filename.S	104-113	68-71		Seek/search address area
&Filename.W	114-123	72-7B		Random/sequential retrieval work area
&Filename.P	124-127	7C-7F		Prime data record count
	128	80		Status indicators :
			0-1	Not used
			2	1= File closed
			3-5	Not used
			6	1= Last prime data track full
			7	1= Block complete

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	129-133	81-85		Last track index normal entry address (CCHHR)
	134-133	86-8A		Last cylinder index entry address (CCHHR)
	139-143	8B-8F		Last master index entry address (CCHHR)
	144-151	90-97		Last independent overflow record address (MBBCCHHR)
&Filename.I	152-153	98-99		Number of independent overflow tracks
&Filename.A	154-155	9A-9B		Number of full cylinder overflow areas
&Filename.O	156-157	9C-9D		Overflow record count
	158-164	9E-A4		Independent overflow area lower limit (MBBCCHH)
	165-171	A5-AB		Independent overflow area upper limit (MBBCCHH)
	172-175	AC-AF		A(&Filename.D)- Address of work area for cylinder overflow control record (COCR)
	176-179	B0-B3		A(&Filename.D+8) - Address of work area for the current track index normal entry count field
	180-183	B4-B7		A(&Filename.D+16) - Address of work area for current track index overflow entry count field
	184-187	B8-BB		A(&Filename.D+24) - Address of work area for current prime data record count field
	188-191	BC-BF		A(&Filename.D+32) - Address of work area for current overflow record count field
	192-195	C0-C3		A(&Filename.D+40) - Address of work area for track index normal entry data field
	196-199	C4-C7		A(&Filename.D+50) - Address of work area for current overflow record sequence-link field
	200-203	C8-CB		A(&IOAREAL) - Address of IOAREAL, the I/O area used for adding records to a file
	204-207	CC-CF		A(&WORKL) - Address of WORKL, work area containing user data records to be added to a file
	208-211	D0-D3		A(&Filename.K) - Address of the ADD key area
	212-215	D4-D7		A(&IOAREAL+8) - Address of key position in IOAREAL
	216-219	D8-DB		A(&IOAREAL+8+&KEYLEN) - Address of data position in IOAREAL

DTFIS (ADDRTR) - part 2

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.2	0 -3	00-03		A(&Filename.S+3) - Address of the seek/search address area plus 3
	4	04	0 1-5 6 7	1= Seek check indicated Not used 1= Over/under seek has occurred 1= An error has been found, but a seek check is indicated
	5 -7	05-07		A(&Filename.W) - Address of the random/sequential retrieval work area
	8 -11	08-0B		Address of IOAREAS, I/O area used for sequential retrieval
	12-15	0C-0F		Address of IOAREAR, I/O area used for random retrieval or address of IOAREA2 (if specified) for sequential retrieval
	16-19	10-13		Address of KEYARG, field containing user supplied key used for random READ/WRITE operations and sequential retrieval initiated by key
	20-23	14-17		Address of WORKR, work area used for random retrieval
	24-27	18-1B		Current sequential I/O area address
	28-31	1C-1F		1) L IOREG,*-4 - Load I/O register for sequential or 2) 4- byte NO-OP instruction for random
	32	20		X'00'= No verify; X'40'= Verify
	33	21		X'00'= Blocked; X'08'= Unblocked
	34	22		R= First prime data record on shared track
	35-39	23-27		Limits for sequential (CCHHR)
	40-41	28-29		H'0'= Blocked records H'2'= Overflow records H'8'= Unblocked records
	42	2A		X'C7'= 2311, 2314 or 2319; X'09'= 2321; X'FF'= 3330, 3340
	43-47	2B-2F		Initial values for sequential
&Filename.H	48-55	30-37		Current sequential DASD address (MBBCCHHR)
	56-63	38-3F		Current overflow DASD address (MBBCCHHR)
	64-65	40-41		Sequential record count
	66-67	42-43		Current track index entry for sequential (HR)
&Filename.T	68-69	44-45		Number of records tagged for deletion
	70-71	46-47		LR &IOREG,0 for random (or 2-byte NO-OP for sequential)
&Filename.G	72-79	48-4F		DASD address save area for random retrieval (MBBCCHHR)

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
	80-83	50-53		Record pointer within I/O area for write (for random retrieval)
&Filename.R	84-87	54-57		Non-first overflow record count
The following information is generated if the cylinder index in core option is specified. Bytes 88-91 (58-5B) are not used.				
	92-95	5C-5F		A(&INDAREA) - Starting address of main storage area specified for cylinder index
	96-97	60-61		AL2(&INDSIZE) - Number of bytes in main storage avialable for cylinder index
	98-105	62-69		Next cylinder index entry to be read (MBBCCHHR)
	106-110	6A-6E		Last cylinder index entry (CCHHR)
	111	6F		Core index byte:
			0	1= First time through B-transient, $$BINDEX
			1	1= End of cylinder index reached
			2	1= Index skip option specified
			3	1= Suppress index in-core option and read cylinder index
			4-7	Not used
	112-115	70-73		Pointer to key (stored by module)
The following information is generated if the prime data in core add function is specified. This information is aligned on a double word boundary. If both cylinder index in core and prime data in core add functions are specified, the following information is found in bytes (116-131) (74-83).				
	116-117	74-75		Size of IOAREAL
	118-119	76-77		Maximum number of prime data records in main storage
	120-123	78-7B		Address of write CCW's
	124-127	7C-7F		Address of read CCW's
	128	80		Switch byte:
			0	1= EOF
			1-7	Not used
	129	81		Reserved.
IJHDCWRK	130-131	82-83		Work field for I/O module.

DTFIS (ADDRTR) - part 3

DTF Assembly Label	Bytes Dec	Bytes Hex	Bits	Function
&Filename.B	0 -7	00-07		X'07', &Filename.S+1, X'40', 6 - Long seek CCW with command chaining
	8 -63	08-3F		Channel program build area
	64-127	40-7F		Channel progr.build area for add function only
&Filename.D	128-135	80-87		Cylinder overflow control record (COCR)
	136-143	88-8F		Current track index normal entry count field
	144-151	90-97		Current track index overflow entry count field
	152-159	98-9F		Current prime data record count field
	160-167	A0-A7		Current overflow record count field
	168-177	A8-B1		Track index normal entry data field
	178-187	B2-BB		Current overflow record sequence-link field
	188-197	BC-C5		Current track index overflow entry data field
	198	C6		X'01' - Add to EOF X'02' - Add to independent overflow area
	199-201	C7-C9		Overflow control bytes (CCH)
	202-203	CA-CB		High HR on overflow track
	204-211	CC-D3		Volume upper limit for prime data records (MBBCCHHR)
	212-217	D4-D9		CLC 0(&KEYLEN,13),0(6) - Unblocked CLC 0(&KEYLEN,13),&KEYLOC-1(6) - Blocked Utility CLC for key
	218-223	DA-DF		MVC 0(&KEYLEN,13),0(12) - Unblocked MVC 0(&KEYLEN,13),&KEYLOC-1(12) - Blocked Utility MVC for key
&Filename.E	224-227[1]	E0-E3		First entry in DSKXTN table (logical unit,cell number)
	232-235[2]	E8-EB		4X'FF' - End of DSKXTN table
&Filename.K	236+	EC-end		Key area for add only. Number of bytes depends on key length, KEYLEN

1) Each entry in the DSKXTN table is four bytes long. The minimum number of entries is 2. There is one entry per extent.

2) Location of the end-of-table indicator depends on length of DSKXTN table.

DTFDU

Bytes Dec	Bytes Hex	Bits	Contents	Function
0-15	0-F			Command Control Block (CCB).
16	10	0-3	B'0000'	Not used.
		4		1=DTF relocated by OPENR.
		5-7	B'000'	Not used.
17-19	11-13			Address of logic module.
20	14		X'1A'	DTF type for OPEN/CLOSE (X'1A'=diskette file).
21	15	0		1=Command chained file.
		1-2	B'00'	Not used.
		3		1=Work area specified.
		4	B'0'	Not used.
		5		1=Open; 0=Close.
		6		1=Input; 0=Output.
		7		Not used.
22-28	16-1C			Filename.
29	1D		X'06'	Device type code (X'06=3540).
30-35	1E-23		C'00CHR00'	Address of HDR1 label in VTOC.
36-37	24-25			Volume sequence number.
38	26			Open communications byte.
				Input File
		0		1=No more extents
		1-2	B'00'	Not used.
		3		1=Exit for user's EOF routine.
		4		1=Next extent on new volume.
		5-6		Not used.
		7		1=Extent switch.
				Output File
		0		1=No more extents.
		1		1=Extents needed at Close time.
		2-3	B'00'	Not used.
		4		1=Next extent on new volume.
		5		1=Extent entered via console.
		6-7		Not used
39	27	0		1=Extent bypassed before file opened (input).
		0-7		Sequence number of current extent opened (output).
40	28			Sequence number of last extent opened.
41-43	29-2B		X'000000'	Reserved.
44-47	2C-2F			Address of IOAREA1.
48-51	30-33			Address of last Read/Write CCW in chain.
52-53	34-35		X'0001'	Lower record limit.
54-57	36-39		X'00CC00RR'	End-of-data seek address (last record + 1)
58-59	3A-3B			Number of records in I/O area (used in short chain processing).
60-63	3C-3F		X'00FF0001'	Seek argument (0CHR).
64-67	40-43			End-of-file routine address (input); 4X'00' (output).
68-71	44-47		X'0049001A'	Seek argument control field.
72	48			Command chaining factor.

Bytes				
Dec	Hex	Bits	Contents	Function
73	49			Switch byte 1.
		0		1=Not first entry after open.
		1	B'0'	Not used.
		2		1=In close routine (output).
		3		1=Error chain to be skipped.
		4		1=End of extent.
		5-7	B'000'	Not used.
74-75	4A-4B			(record size multiplied by command chain factor)-1.
76-80	4C-50		X'FFFFFFFFFF'	Seek argument bucket.
81-83	51-53		X'000000'	Reserved.
84-87	54-57			Instruction to load user's I/O register (or NOP).
88-91	58-5B			Address of current I/O area.
92-95	5C-5F			Logical record size.
96-99	60-63			Address of last byte of the I/O area.
100	64			Logical indicators.
		0		1: ERROPT=address.
		1		1: ERROPT=IGNORE.
		2		1: ERROPT=SKIP.
		3		Not used.
		4		1=Two I/O areas.
		5-7		Not used.
101-103	65-67			Address of user's error handling routine.
104	68			CCW count (write command only).
105	69			Allowed operations
		0		1=Allow read commands.
		1		1=Allow write commands.
		2		1=Suppress unit check on C4/C6.
		3-7	B'00000'	Not used.
106	6A		X'00'	Sector factor (X'00'=128).
107	6B		X'00'	Reserved.
108	6C	0		1=Write protect.
		1		1=No feed at EOF.
		2		1=Check multivolume sequence.
		3		1=Multivolume file.
		4		1=Verify requested.
		5		1=c6s written (update ERMAP).
		6		1=Read/Write security.
		7	B'0'	Not used.
109-111	6D-6F		X'000000'	Not used.
112-119	70-77			Feed CCW.
120-127	78-7F			Define ops CCW (output); 8X'00' (input).
128-135	80-87			Seek CCW.
136-143	88-8F			TIC CCW.
144-X	90-Y		X=143+8*(# of CCWs)	Read/Write data CCWs, 1, 2, 13, or 26.
			Y=8F+8*(# of CCWs)	Read/Write CCWs.
X+1	Y+1			NOP CCW (output only).

DTFPH (Magnetic Tape)

Bytes		Bits	Contents	Function
Dec	Hex			
0 -15	00-0F			CCB
16	10	0-1		Not used
		2		COBOL open; ignore option
		3		Not used
		4		DTF Table address; constants relocated by OPENR
		5		Not used
		6		1= ASCII 0= EBCDIC
		7		Not used
17-19	11-13		3X'00'	
20	14		X'12'	Standard labeled, output
			X'14'	Standard labeled, input, forward
21	15	0-3		Not used
		4		1= input; 0= output
		5-7		Not used
22-29	16-1D			Symbolic filename
30	1E			Not used
31	1F	0-4	B'01100'	Used as displacement by OPEN
		5		
		6-7		Reserved
32	20	0		1= Standard labels
		1-2		Not used
		3		1= No rewind
		4		Not used
		5		User label address; 1= yes, 0= no
		6-7		Not used
33-35	21-23			User label routine address
36	24	0		1= DTFPH table
		1		Not used
		2		File switch: 1= input, 0= output
		3		Not used
		4		1= EOF switch
		5-7		Not used
37-39	25-27			User label exit
40-43	28-2B		DC F'0'	Reserved for OPEN
44-87	2C-57			EOV routine
88-89	58-59		DC 2X'00'	Reserved for OPEN
90-95	5A-5F		DC 6X'00'	File serial number
96-99	60-63		DC 4X'00'	Volume sequence number
100-103	64-67		DC 4X'00'	File sequence number

DTFPH (Sequential Disk)

Bytes Dec	Bytes Hex	Bits	Function
0 -15	00-0F		CCB
16	10	0	1= Dequeue old volume extents
		1	Not used
		2	1= File assigned 'IGN' (COBOL)
		3	Not used
		4	1= DTF relocated by OPENR
		5-7	Not used
17-19	11-13		3X'00'
20	14		DTF type (X'21')
21	15		Open/Close indicators
		0	1= 2321 (Version 1-2 only)
		1	1= Blocked files
		2	1= Work file
		3	1= Work area
		4	1= Not version 1 table type
		5	1= Open; 0= Closed
		6	1= Input; 0= Output
		7	1= User labels specified
22-28	16-1C		Filename (See byte 29)
29	1D		Device type code: X'05'= 3330-11 X'00'= 2311 X'07'= 3350 X'01'= 2314,2319 X'08'= 3340 general X'02'= 2321 X'09'= 3340 35MB X'04'= 3330-1, -2 X'0A'= 3340 70MB
30	1E		C'F'= EOF indicator for DTFPH
30-35	1E-23		(BCCHHR) Address of F1 label in VTOC (output) (BCCHHR) Address of next DLBL-EXTENT record (input)
36-37	24-25		Volume sequence number
38	26		Open communication byte: Output
		0	1= No more EXTENTS
		1	1= EXTENTS for LIOCS at close
		2	1= Process trailer labels
		3	1= Process header labels
		4	1= New volume on next extent
		5	1= EXTENTS entered via console
		6	1= Process trailer labels at close
		7	1= Check EXTENT for minimum of 2 tracks
			Input
		0	1= No more EXTENTS
		1	Not used
		2	1= No F1 label, process EXTENTS only
		3	Not used
		4	1= New volume on next EXTENT
		5	Not used

Bytes Dec	Bytes Hex	Bits	Function
38 (Cont'd)	26		Open communications byte (Input) (Cont'd)
		6	1= Process header labels
		7	Not used
39	27		Sequence number of current EXTENT being opened
40	28		Sequence number of last EXTENT opened (not a console EXTENT entry)
41-43	29-2B		Address of user's label routine
44-47	2C-2F		Address of IOAREA1
48-51	30-33		CCHH address of user's label track. Initially X'80000000'
52-53	34-35		Lower head limit (HH) X'0000' if type 1; X'00nn' if type 128 (n= head limit)
54-57	36-39		EXTENT upper limit (CCHH)
58-59	3A-3B		BB seek address: X'0000' if disk device X'00nn' if 2321 whre 'nn'= bin number
60-63	3C-3F		EXTENT lower limit (CCHH)
64	40		Record number: 1= Input; 0= Output
65-67	41-43		Not used
68-71	44-47		CCHH control bucket CCHH= X'00C80009' if 2311 - type 1 CCHH= X'00C80013' if 2314 or 2319 - type 1. CCHH= X'13090413' if 2321 - type 1 CCHH= X'01940012' if 3330 - type 1 CCHH= X'03280012' if 3330-11 - type 1 CCHH= X'015C000B' if 3340 35MB CCHH= X'02B8000B' if 3340 70MB CCHH= X'0228001D' if 3350 - type 1
72	48		Record number
73	49		Not used
74-75	4A-4B		Not used
76-80	4C-50		CCHHR bucket= extent lower limit and record number
81-83	51-53		Not used

Note : where nn = current upper head number

DTFPH (DAM FILES)

Bytes		Bits	Function
Dec	Hex		
0-15	0-F		CCB
16	10		X'08' indicates DTF relocated by OPENR.
17-19	11-13		3X'00'.
20	14		DTF type (X'23').
21	15		Option codes.
		0	1=Output, 0=Input.
		1	Not used.
		2	Not used.
		3	Not used.
		4	Not used.
		5	Not used.
		6	1=2321 (Version 1/2 only).
		7	Not used.
22-28	16-1C		Filename
29	1D		Device type code :
			X'00' = 2311
			X'01' = 2314, 2319
			X'02' = 2321
			X'04' = 3330-1, -2
			X'05'= 3330-11
			X'07'= 3350
			X'08' = 3340 general
			X'09' = 3340 35MB
			X'0A'= 3340 70MB.
30-31	1E-1F		Logical unit address of first volume containing the file.
32	20		X'40'=Device supports RPS.
33-35	21-23		Address of user label routine.
36-39	24-27		Address of user routine to process EXTENT information.

DTFPH (DISKETTE)

Bytes			
Dec	Hex	Bits	Function
0-15	0-F		CCB.
16	10	0	1=Dequeue old volume extents.
		1-3	Not used.
		4	1=DTF relocated by OPENR.
		5-7	Not used.
17-19	11-13		3X'00'.
20	14		DTF type (X'21').
21			Open/close indicators.
		0-2	Not used.
		3	1=Work area.
		4	1=Not version 1 DTF table type.
		5	1=Open; 0=Closed.
		6	1=Input ; 0=Output.
		7	Not used.
22-28	16-1C		Filename (see byte 29).
29	1D		Device type code (3540=X'06').
30	1E		C'F'=EOF indicator for DTFPH.
30-35	1E-23		(0CHR00) Address of HDR1 label in VTOC (output).
36-37	24-25		Volume sequence number.
38	26		Open communications byte.
			Input
		0	1=No more extents.
		1-3	Not used.
		4	1=New volume or new extent.
		5-7	Not used.
			Output
		0	1=No more extents.
		1	1=Extents for LIOCS at close.
		2-3	Not used.
		4	1=New volume on next extent.
		5	1=Extents entered via console.
		6	Not used.
		7	1=Check extent for minimum of 2 tracks.
39	27		Sequence number of current extent being opened.
40	28		Sequence number of last extent opened (not a console extent entry).
41-43	29-2B		Not used.
44-47	2C-2F		Address of IOAREA1.
48-51	30-33		Not used.
52-53	34-35		X'0000'.
54-57	36-39		Extent upper limit (0CHR).
58-59	3A-3B		Not used.
60-63	3C-3F		Extent lower limit (0CHR).
64	40		Record number. 1=Input, 0=Output.
65-67	41-43		Not used.
68-71	44-47		0CHR control bucket. 0CHR= X'0049001A' for 3540 (output only).
72	48		Record number.
73	49		X'10' - multivolume file (input) X'40' - last volume on multivolume file (input).

Bytes		Bits	Function
Dec	Hex		
74	4A		Record size (maximum of 128)
75	4B		Not used.
76-80	4C-50		0CHR bucket = extent lower limit and record number (output).
81-83	51-53		Not used.

DTFDI

Bytes Dec	Bytes Hex	Bits	Function
0-15	00-0F		CCB. If RPS is supported, the CCW address in bytes 9-11 (09-0B) is changed by OPEN to point to an RPS CCW string in the user virtual area. CLOSE restores it.
16	10	0-1	Not used.
		2	COBOL open; ignore option.
		3	Not used.
		4	DTF table address constants relocated by OPENR.
		5-7	Not used.
17-19	11-13		Address of logic module. If RPS is supported, OPEN changes this address to point to an RPS version of the logic module in the system virtual area. CLOSE restores it.
20	14		DTF Type =X'33'.
21	15		Open/Close indicators - X'02'=input, X'00'=output.
22-28	16-1C		Symbolic filename.
29	1D		DASD or diskette device indicators X'00'=2311 X'01'=2314, 2319 X'04'= 3330-1, -2 X'05'= 3330-11 X'07'= 3350 X'08'=3340 general X'09'=3340 35MB X'0A'=3340 70MB.
30-35	1E-23		DASD address of format-1 label.
36-37	24-25		DASD or diskette volume sequence number.
38	26		Open communications switch.
		0	1=No more extents --diskettes.
		1-3	Not used.
		4	Always 1.
		5-7	Not used.
39	27		Sequence number of current extent.
40	28		Sequence number of last extent, or X'80' for 1442 reader punch.
41	29		Open indicator = X'20'.
42	2A		Device type indicators :
		0	Unused.
		1	1=DTF has been extended into the partition GETVIS area.
		2	1=DASD
		3	1=tape
		4	1=printer
		5	1=punch
		6	1=reader
		7	1=RPS supported.
43	2B		Logic module device indicators : X'F3' = DASD or diskette device. X'F1' = reader or tape device. X'F0' = other type devices.

Bytes		Bits	Function
Dec	Hex		
44	2C		Logic module option switches
		0	1=input, 0=output.
		1	1=eject for RDR-PCH; 0=no eject.
		2	1=not first pass; 0=first pass.
		3	1=two I/O areas; 0=one I/O area.
		4	1=2540 Punch.
		5	1=SYSLST/SYSPCH.
		6	1=Tape SYSLST/SYSPCH.
		7	1=ASCII ; 0=EBCDIC code.
45-47	2D-2F		Alternate I/O area address.
48-51	30-33		Reserved for future use.
52-53	34-35		Extent lower head limit.
54-57	36-39		Extent upper head limit.
58-64	3A-40		DASD seek address.
			Diskette seek address at byte 60 (3C).
65-67	41-43		Users EOF address.
68-72	44-48		Control bucker CCHHR.
			Byte 72 (48) always X'01' for diskettes.
73	49		Logic module switches
			X'01'=input
			X'00'=output
			X'00'=both input and output on diskettes.
74-75	4A-4B		Logic module constants
			X'0020' DASD output
			X'0018' DASD input
			X'0008' Diskette devices
			X'0000' Non-DASD devices.
76-80	4C-50		Count field CCHHR (0CHR0 for diskettes).
81	51		Key length.
82-83	52-53		Data length.
84-87	54-57		Instruction to load IOREG with correct I/O area address.
88-103	58-67		Seek, Search CCWs.
			Seek, Read/Write CCW for diskette files.
104-111	68-6F		TIC CCW.
			NOP CCW for diskette output files; unused for diskette input files.
112-119	70-77		Input/output CCW.
120-127	78-7F		Second output CCW.
128-151	80-97		Verify CCWs for output.
152-159	98-9F		Error CCW1.
160-167	A0-A7		Error CCW2.
168-231	A8-E7		Save area (64 bytes).
232-235	E8-EB		DC A(WLRERR) if WLRERR=Address.
			B 28(15) if ERROPT=omitted.
			B 25(15) if ERROPT=SKIP.
			B 28(15) if ERROPT=IGNORE.
236-239	EC-EF		DC A(ERROPT) if ERROPT=Address.
			B 0(15) if ERROPT=omitted.
			B 24(15) if ERROPT=SKIP.
			B 28(15) if ERROPT=IGNORE.

DTFCP (DISK=YES)

Bytes Dec	Bytes Hex	Bits	Function
0-15	00-0F		CCB. If RPS is supported, the CCW address in bytes 9-11 (09-0B) is changed by OPEN to point to an RPS CCW string in the user virtual save area. CLOSE restores it.
16	10	0	Not used
		1	Set by Maint; indicates that LIOCS must retrieve extents from the VTOC instead of the label cylinder.
		2	COBOL open; ignore option.
		3	X'10' indicates an unlabeled FORTRAN tape.
		4	DTF table address constants relocated by OPENR.
		5	Used by FORTRAN (Sequential Disk Backspace and Rewind).
		6	1 = ASCII, 0 = EBCDIC.
		7	FORTRAN is calling DTFCP.
17-19	11-13		Logic module address. If RPS is supported, OPEN changes this address to point to an RPS version of the logic module in the system virtual save area. CLOSE restores it.
20	14		DTF type X'32' except in the case of disk assigned to units SYS000 to SYSnnn. In this case, a DTFCP open phase changes it to X'20'.
21	15		Open indicators: X'02' input, X'00' output, except for tapes assigned to SYS000 to SYSnnn when X'00' = input and X'08' is output. X'08' DISK=YES indicator.
		0	1 = no rewind, 0 = rewind.
22-28	16-1C		Filename (see byte 29).
29	1D		Device type code : X'00' = 2311 X'01' = 2314, 2319 X'04' =3330-1, -2 X'05' = 3330-11 X'07' = 3350 X'08' = 3340 general X'09' = 3340 35MB X'0A' = 3340 70MB.
30-35	1E-23		File address for disk; block count if bit 7 of byte 16 is on.
36-37	24-25		Volume sequence number or work area.
38	26		Open switch
39	27		Sequence number of current extent.
40	28		Sequence number of last extent, or X'80' if 1442 punch.
41	29		X'80' indicates request for standard label tape OPEN.
42	2A		X'80' device is a 2560. X'40' DTF has been extended into the user virtual save area X'20' device is a DASD X'10' device is a tape X'08' device is a printer X'04' device is a punch X'02' device is a reader X'01' RPS is supported.
43	2B		X'F3' device is a DASD X'F1' device is a reader X'F0' device is other type.

DTFCP (DISK=YES) (...Continued)

Bytes Dec	Bytes Hex	Bits	Function
44	2C	0	1=input, 0=output.
		1	1=eject needed for a reader punch; 0=no eject.
		2	0=first pass, 1=not first pass.
		3	1=two I/O areas, 0=one I/O area.
		4	1=2540 punch.
		5	1=SYSLST or SYSPCH.
		6	1=SYSLST or SYSPCH on output tape.
		7	1=TLBL is present and tape is labeled.
45-47	2D-2F		IOAREA2 address.
48-51	30-33		X'80000000'.
52-53	34-35		Lower head limit.
54-57	36-39		Extent upper limit.
58-64	3A-40		BBCCHHR seek address.
65-67	41-43		EOF address.
68-71	44-47		Control bucket CCHH.
72	48		Number of record per track for output, number of record per track +1 for input.
73	49		X'00' for output, X'01' for input.
74-75	4A-4B		X'0020' for output, X'0018' for input for DASD X'0008' for 2560 and 5425 output. X'0000' for nondisk device.
76-80	4C-50		CCHHR for count field.
81	51		Key length.
82-83	52-53		Data length.
84-87	54-57		Instruction to load user I/O area address to I/O register.
88-111	58-6F		Seek, search, TIC CCWs.
112-119	70-76		CCW for DASD input and first CCW for DASD output. This CCW can be used for other device if unit is not a DASD
End-of-table if DTF is defined for an input file			
120-127	77-7F		Second CCW for output.
128-151	80-97		Verify CCWs for output.
End-of-table if DTF is defined for output file and DEVADDR does not equal SYSPCH.			
152-159	98-9F		2540 punch error recovery CCW 1.
160-167	A0-A7		2540 punch error recovery CCW 2.
168-231	A8-E7		Reserved.
When the CP open initializes the table and determines that the device is a 2540 punch, the following bytes in the table are changed:			
30	1F		X'FF' indicator to DTFCP open phases and logic module.
32-35	20-23		Instruction to load user I/O area to I/O register.
48-55	30-37		CCW.
56-63	38-3F		2540 punch error recovery CCW 1.
64-71	40-47		2540 punch error recovery CCW 2.
72-151	48-97		80-byte card image, savearea 1.
152-231	98-E7		80-byte card image, savearea 2.

Bytes Dec	Bytes Hex	Bits	Function
When the CP open initializes the table and determines that the device is a 2560 or 5425, the following bytes in the table are changed :			
32-35	20-23		Instruction to load user I/O area to I/O register.
48-55	30-37		First output CCW.
56-63	38-3F		Second output CCW.
64	40		Stacker select character V for ASCII.
65	41		Stacker select character W for EBCDIC.

DTFCP (DISK=NO)

Bytes Dec	Bytes Hex	Bits	Function
0-15	00-0F		CCB.
16	10	0-1	Not used.
		2	COBOL open; ignore option.
		3	Not used.
		4	DTF table address constants relocated by OPENR.
		5	Not used.
		6	1=ASCII (used only if DISK=YES), 0=EBCDIC (used only if DISK=YES).
		7	FORTRAN is calling DTFCP.
17-19	11-13		Logic module address.
20	14		DTF type X'32' except in the case of tape assigned to units SYS000 to SYSnnn. In this case, a DTFCP open phase changes it to X'10'.
21	15		Open indicators X'02' input, X'00' output (except for tapes assigned to SYS000 to SYSnnn when it is X'00' input, X'08' output).
22-28	16-1C		Filename (see byte 29).
29	1D		Reserved for future use.
30	1E		Indicator to DTFCP open phase and logic module. X'FF' for input files. X'00' for output files.
31	1F		Reserved for future use.
32-35	20-23		Instruction to load user's I/O area address into I/O register.
36-37	24-25		Volume sequence number or work area.
38	26		Open switch.
39	27		Sequence number of current extent.
40	28		Sequence number of last extent, or X'80' if 1442 punch.
41	29		X'20'.
42	2A		X'80' device is a 2560. X'40' device is a 5425. X'10' device is a tape. X'08' device is a printer. X'04' device is a punch. X'02' device is a reader.
43	2B		X'F1' device is a reader or tape. X'F0' device is other type.
44	2C	0	1=input, 0=output.
		1	1=eject needed for a reader-punch, 0= no eject.
		2	1=not first pass, 0=first pass.
		3	1=two I/O areas, 0=one I/O area.
		4	1=2540 punch.
		5	1=SYSLST or SYSPCH.
		6	1=SYSLST or SYSPCH on output tape.
		7	Reserved for future use.
45-47	2D-2F		IOAREA2 address.
48-55	30-37		CCW.
End-of-table if DTF is defined as output file and DEVADDR is not equal to SYSPCH.			

Bytes Dec	Bytes Hex	Bits	Function
56-63	38-3F		2540 punch error recovery CCW 1.
64-71	40-47		2540 punch error recovery CCW 2.
65-67	41-43		EOF address, input only.
End-of-table if DTF is defined as input file			
72-151	48-97		80-byte card image, save area 1.
152-231	98-E7		80-byte card image, save area 2.
If the device is a 2560 or 5425, bytes 56 onward contain the following information.			
56-63	38-3F		Second output CCW.
64	40		Stacker select character V for ASCII.
65	41		Stacker select character W for EBCDIC.
66-75	42-4B		Reserved for future use.
76-235	4C-EB		First I/O area.
236-237	EC-ED		Reserved.
238-317	EF-13D		Second I/O area.
318-319	13E-13F		Reserved.

DTFCP (DISK=PARAMETER OMITTED)

Bytes Dec	Bytes Hex	Bits	Function
0-15	00-0F		CCB
16	10	0-1	Not used.
		2	COBOL open; ignore option.
		3	Not used.
		4	DTF table address constants relocated by OPENR.
		5	Not used.
		6	1=ASCII (used only if DISK=YES), 0=EBCDIC (used only if DISK=YES).
		7	Used by FORTRAN
17-19	11-13		Logic module address.
20	14		DTF type X'31' except in the case of tape assigned to units SYS000 to SYSnnn. In this case DTFCP open phase changes it to X'10'.
21	15		Open indicators X'02' input, X'00' output (except for tapes assigned to SYS000 to SYSnnn when it is X'00' input, X'08' output.
22-28	16-1C		Filename.
29	1D		Reserved for future use.
30	1E		X'00' indicator to DTFCP open phases and logic module.
31	1F	0	1=input, 0=output.
		1	1=eject needed for a read punch, 0=no eject.
		2	1=not first pass, 0=first pass.
		3	1=two I/O areas, 0=one I/O area.
		4	1=2540 punch.
		5	1=SYSLST or SYSPCH.
		6	1=SYSLST or SYSPCH on output tape.
		7	1=TLBL specified and tape is labeled.
32	20		Open indicators.
33-35	21-23		IOAREA2 address.
36-39	24-27		Instruction to load user's I/O area address into I/O register.
40-47	28-2F		CCW.
End of table if DTF is defined as output file and DEVADDR is not equal to SYSPCH.			
48-55	30-37		2540 punch error recovery CCW 1.
56-63	38-3F		2540 punch error recovery CCW 2.
57-59	39-3B		EOF address, input only.
End of table if DTF is defined as input file.			
64-143	40-8F		80-byte card image, save area 1.
144-223	90-13F		80-byte card image, save area 2.

DTFCP (DISK=PARAMETER OMITTED)

Bytes		Bits	Function
Dec	Hex		
For 2560 and 5425 bytes 48 onwards contain the following information :			
48-207	30-CF		IOAREA1.
208-209	D0-D1		Reserved
210-369	D2-171		IOAREA2.
370-371	172-173		Reserved
372-451	174-1C3		Compare area.

DTF - Table Types

DTF Type Code (Byte 20) of DTF Table	DTF	Description
X'00'	DTFCD	Combined files
X'01'	DTFPT	Paper tape files
X'02'	DTFCD	Reader and 3881 Optical Mark Reader files
X'03'	DTFCN	Console
X'04'	DTFCD	Punch files
X'05'	DTFCD	Reader files on 2560, 5425
X'07'	DTFPR	Printer files on 2560
X'08'	DTFPR	Printer files
X'09'	DTFOR	Optical Reader files except 3881 and 3886 files
X'0A'	DTFOR	Optical Reader files (HEADER=YES)
X'0B'	DTFMR	Magnetic Ink Character Recognition (MICR) and Optical Reader/Sorter files
X'0C'	DTFDR	3886 Optical Character Reader files
X'10'	DTFMT	Magnetic tape workfiles
	DTFCP	Magnetic tape workfiles (compiler). (Note 1)
X'11'	DTFMT	Nonstandard or unlabeld tape files
X'12'	DTFMT	Standard labeled, output tape files
	DTFPH	Standard labeled, output tape files (physical IOCS)
X'13'	DTFMT	Standard labeled, input tape files (read backward)
X'14'	DTFMT	Standard labeled, input tape files (read forward)
X'1A'	DTFDU	Diskette Input/Output Unit files
X'20'	DTFSD	Sequential DASD workfiles and data files
	DTFCP	DASD workfiles (compiler)
X'21'	DTFPH	Sequential DASD files, MOUNTED=SINGLE (physical IOCS)
X'22'	DTFDA	Direct access files
X'23'	DTFPH	Direct access files, MOUNTED=ALL (physical IOCS)
X'24'	DTFIS	Indexed sequential, LOAD file
X'25'	DTFIS	Indexed sequential, ADD file
X'26'	DTFIS	Indexed sequential, RETRVE file
X'27'	DTFIS	Indexed sequential, ADDRTR file
X'28'	ACB	Access Method Control Block for VSAM
X'30'	DTFCP	Compiler file for DOS Version 1 (Note 1)
X'31'	DTFCP	Compiler file for DOS Versions 2 onward
X'32'	DTFCP	Compiler file for DOS Versions 2 onward (Note 2)
X'33'	DTFDI	Device independent system unit files
X'40'	DTFBT	Basic Telecommunication Access Method (BTAM) file (Note 3)
X'50'	DTFQT	Queued Telecommunication Access Method (QTAM) file (Note 3)
X'60' - X'67'		

Notes

1. DTF type is X'30' except for tape or DASD assigned to units SYS000 to SYSnnn. In this case, the DTFCP open phases change the DTF type to X'10' for tape workfiles, or X'20' for DASD workfiles.

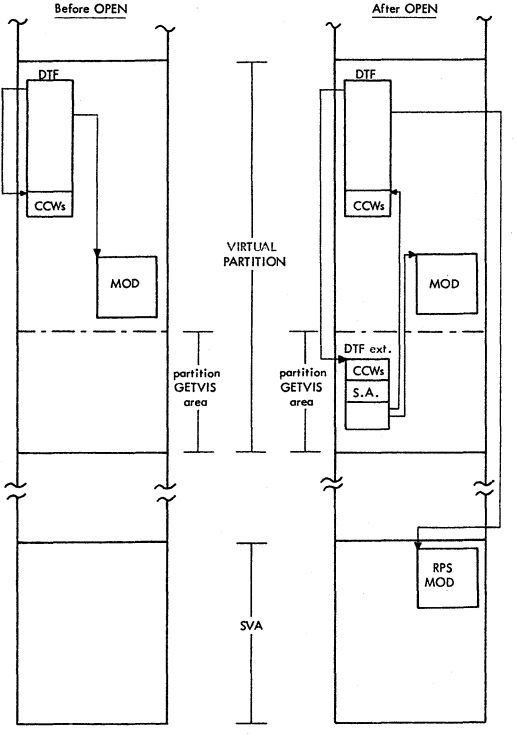
Notes (continued)

2. DTF type is X'32' except for DASD assigned to units SYS000 to SYSnnn. In this case, the DTFCP open phases change the DTF type to X'20' for DASD workfiles.

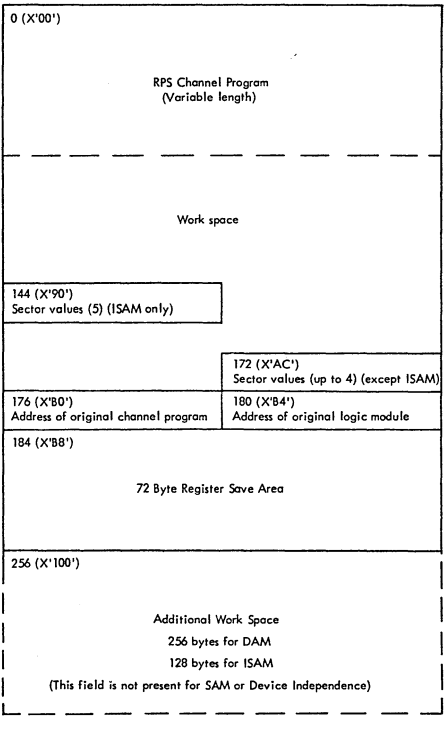
3. The following control unit codes are ORed into the low-order 4 bits of the DTF type code.

Control Unit	Code
7770	1
2848	3
2701	4
2702	5
2703	6

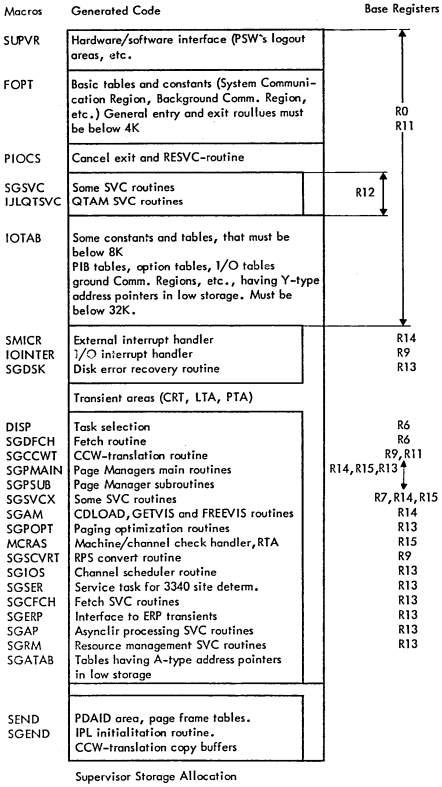
RPS DTF/MODULE RELATIONSHIP



RPS DTF-Extention



CHAPTER IV
DOS/VS SUPERVISOR CONTROL BLOCKS AND AREAS



Supervisor Storage Allocation

SUPERVISOR CALLS

*optional

SVC Dec	SVC Hex	Macro supported	Function
0	0	EXCP	Execute Channel Program
1	1	FETCH	Fetch any phase
2	2		Fetch a logical transient (B-transient)
3	3		Force dequeue
4	4	LOAD	Load any phase
5	5	MVCOM	Modify supervisor communication region (if issued by MVCOM macro) Fetch another physical transient (if issued by a physical transient)
6	6	CANCEL	Cancel a problem program or task
7	7	WAIT	Wait for a CCB or TECB
8	8		Transfer control to the problem program from a logical transient (B-transient)
9	9	LBRET	Return to a logical transient (B-transient) from the problem program after an SVC 8
10*	A	SETIME	Set timer interval
11	B		Return from a logical transient (B-transient)
12	C		Logical AND (Reset) to second job control byte (displ.57 in comm.region), or reset PCIL being condensed bit (displ.59 in comm.region)
13	D		Logical OR (Set) to second job control byte (displ.57 in comm.region), or reset PCIL being condensed bit (displ.59 in comm.region)
14	E	EOJ	Cancel job and go to job control for end of job step
15	F	SYSIO	Headqueue and execute channel program
16*	10	STXIT(PC)	Provide supervisor with linkage to user's PC routine for program check interrupts
17*	11	EXIT(PC)	Return from user's PC routine
18*	12	STXIT(IT)	Provide supervisor with linkage to user's IT routine for interval timer interrupts
19*	13	EXIT(IT)	Return from user's IT routine
20*	14	STXIT(OC)	Provide supervisor with linkage to user's OC routine for external or attention interrupts (operator comm.)
21*	15	EXIT(OC)	Return from user's OC routine
22	16		Seize/Release system; Enable/disable external and I/O interrupts; Set key in user's PSW
23*	17		Load phase header. Phase load address is stored at user's address
24*	18	SETIME	Set timer interval and provide supervisor with linkage to user's TECB, if any

*optional

SVC Dec	SVC Hex	Macro Supported	Function
25*	19		Issue HALT I/O on a teleprocessing device, or HALT I/O on any device if issued by OLTEP. With multiprogramming, dequeue an unstarted OLTEP I/O request to a shared device
26*	1A		Validate address limits
27*	1B		Special HIO on teleprocessing devices
28*	1C	EXIT(MR)	Return from user's stacker select routine (MICR type devices only)
29*	1D	WAITM	Provide support for multiple wait macro WAITM
30*	1E	QWAIT	Wait for a QTAM element
31*	1F	QPOST	Post a QTAM element
32	20		Reserved
33	21	COMRG	• Provide address of partition com. reg. • Force task select for system tasks
34	22	GETIME	Provides Time-of-Day and updates the DATE field
35*	23	HOLD	Hold a track for use by the requesting task only
36*	24	FREE	Free a track held by the task issuing the FREE
37*	25	STXIT(AB)	Provide supervisor with linkage to user's AB routine for abnormal termination of a task
38*	26	ATTACH	Initialize a subtask and establish its priority
39*	27	DETACH	Perform normal termination of a subtask. It includes calling the FREE routine to free any tracks held by the subtask
40*	28	POST	Inform the system of the termination of an event and ready any waiting tasks
41*	29	DEQ	Inform the system that a previously enqueued resource is now available
42*	2A	ENQ	Prevent tasks from simultaneous manipulation of a shared data area (resource)
43	2B		Reserved
44*	2C		Provide supervisor support for external creation of unit check records by specific request
45*	2D		Provide emulator interface
46*	2E		Provide OLTEP with the facility to operate in supervisory state
47*	2F	WAITF	Provide support for multiple wait macro WAITF for MICR type devices
48	30		Fetch a CRT transient
49	31		Used by VTAM to init. exec. of channel program
50	32		Reserved for LIOCS error diagnostic
51	33		Return phase header

*optional

SVC Dec	SVC Hex	Macro supported	Function
52*	34	TTIMER	Return the remaining time interval, or cancel a time interval
53*	35		Used by VTAM to schedule user exit in applic.progr.
54	36	FREEREAL	Release page frames to selection pool
55	37	GETREAL	Provide interface between SDAID and PDAID initialization routine and page management routine, to create the PDAID alternate area or the SDAID buffer area
56	38		Reserved
57	39		Reserved
58	3A		Provide interface between job control and the supervisor. Get real storage for real jobs
59	3B		Provide interface between EOJ and the supervisor. Reset the storage key for virtual jobs
60	3C	GETDADR	Provide virtual address of location within I/O areas for ERP and CRT routines
61*	3D	GETVIS	Get storage in virtual partition
62*	3E	FREEVIS	Free storage in virtual partition
63	3F	USE	Use a resource
64	40	RELEASE	Release a resource
65*	41	CDLOAD	Load VSAM or CI phase
66	42	RUNMODE	Return mode in which program is running
67*	43	PFIX	Fix page(s) in real storage
68*	44	PFREE	Free page(s) in real storage
69*	45	REALAD	Return real address corresponding to a given virtual address
70*	46	VIRTAD	Return virtual address corresponding to a given real address
71*	47	SETPFA	Establish or terminate the linkage between the supervisor and a user page-fault appendage routine
72*	48	GETCBUF/FREECBUF	Get or free copy buffer for IDAL or tape ERP
73*	49	SETAPP	Allow linkage to channel and appendage routines
74*	4A		Fix page(s) in real storage for restart
75*	4B	SECTVAL	Calculate a sector value (disk dev. with RPS feature)
76	4C		Initiate recording on SYSRLC file
77	4D	TRANSCSW	Returns the virtual address of a copied CCW

* optional

SVC		Macro supported	Function
Dec	Hex		
78*	4E	CHAP	Change subtask priority (supported if AP=YES)
79*	4F	SYNCH	Give control to synchronous exit.
80*	50	SETT	Set task time interval
81*	51	TESTT	Return the remaining task time interval or cancel a time interval.
82-84	52-54		Reserved
85*	55	RELPAG	Release contents of one or more pages.
86*	56	FCEPGOUT	Force a page-out of one or more pages
87*	57	PAGEIN	Page-in one or more pages.
88*	58	TPIN	Start TP Balancing
89*	59	TPOUT	Stop TP Balancing
90*	5A	PUTACCT	Provide interface with POWER/VS for additional account information (by user)
91*	5B		Provide interface with POWER/VS for standard account information (DOS/VS)
92*	5C	XECBTAB	XECB table manipulation (DEFINE, DELETE, CHECK an entry)
93*	5D	XPOST	Post an XECB
94*	5E	XWAIT	Wait on an XECB
96*	60	EXIT (TT)	Return form user's task timer exit
97*	61	STXIT(TT)	Provide supervisor with linkage to user's task timer exit routine for task time interval end.

COMMAND CONTROL BLOCK (CCB)

Count	Transmission information	CSW Status Bits	Type Code and Logical Unit	Reserved for logical IOCS	CCW Address	Reserved for physical IOCS	CCW Address in CSW	Optional Sense CCW
0 1	2 3	4 5	6 7	8 9	11	12 13	15	16 23

Byte(s)		Description		
0-1	Used for residual Count.	Byte 1 is also used by the BTAM Channel End Appendage for fix requests.		
2-3	Transmitting information between Physical IOCS & Problem Program	Byte 2		Set on by:
		Bit 0:	Traffic Bit (Wait) (Note 5)	PIOCS*
		Bit 1:	End of File (/* or /&) 3211-UCSB Parity Check (Line Complete) (Note 2)	PIOCS
		Bit 2:	Unrecoverable I/O error	PIOCS
		Bit 3:	Accept unrecoverable I/O error	Pr.Pr.**
		Bit 4:	Return DASD Data Checks, 3540 Data Checks, 2671 errors, or 1017/1018 errors to the user; indicate action-type messages for DOC; Return 5425 not ready.	Pr.Pr.
		Bit 5:	Post at Device End (Note 5)	Pr.Pr.
		Bit 6:	Return Tape Read Data Check; 1018 or 2560 Data Check; 2520, 2540, 2560, 3881 or 5425 Equipment Check; Accept 3504, 3505 or 3525 Perm. Error; DASD Data Checks on Read or Verify Command on 3203, 3211, or 5203 Passback Requested. (Notes 3, 6 and 8)	Pr.Pr.
		Bit 7:	User Error Routine (Note 10)	Pr.Pr.
		Byte 3		Set on by:
		Bit 0:	DASD Data Check in Count Area; Permanent Error for 3330, 3340 or 3350 MICR-SCU Not Operational; 1287/1288 Data Check; 3203, 3211, or 5205 Print Check/Equipment Check; 3540 Special Record Transferred	PIOCS
		Bit 1:	DASD Track Overrun; MICR Intervention required; 1287-Keyboard Correction in Journal Tape Mode; 1017-Broken Tape; 3211-Print Quality/Equipment Check.	PIOCS
		Bit 2:	DASD End of Cylinder; MICR-(Note 4) 1287/1288-Hopper Empty in Document Mode. 3211/2245 Line Position Error. (Note 7)	

* Physical IOCS ** Problem Program

Count	Transmission information	CSW Status Bits	Type Code	Reserved for logical IOCS	CCW Address	Reserved for physical IOCS	CCW Address in CSW	Optional Sense CCW
0 1	2 3	4 5	6 7	8 9	11	12 13	15	16 23

Byte(s)	Description	
2-3 (....Cont'd)	Byte 3	Set on by
	Bit 3: 2520,2540,3881-Equipment Check;2560 3203,5203,5425 Data Check/ Equipment Check; Tape Read Data Check; DASD-Any Data Check; 1287-Equipment Check; 1017/1018 Data Check; 3211-Print Check/ Data Check; 3504,3505,3525 Perm. Error (Note 8); 3540 Diskette Data Check.	PIOCS
	Bit 4: Non-Recovery Questionable Condition: Card- Unusual Command Sequence DASD- No Record Found 1287/1288- Document Jam or Torn Tape; 3211- UCSB Parity Check(Command retry);5425 Not Ready.	PIOCS
	Bit 5: No Record Found Condition(Retry on 2311,2314,2319,3330,3340 or 3350	Pr.Pr.
	Bit 6: Carriage Channel 9 Overflow or Verify Error for DASD; 1287-Document Mode-Late,Stacker Select; 1288-End of Page	PIOCS
	Bit 7: Command Chaining,Retry from the next CCW to be executed	Pr.Pr.
4-5 CSW Status Bits	Byte 4 (Note 1)	Byte 5
	Bit 0: Attention 1: Status Modifier 2: Control Unit End 3: Busy 4: Channel End 5: Device End 6: Unit Check 7: Unit Exeption	Bit 0: Program Controlled Interruption 1: Incorrect Length 2: Program Check 3: Protection Check 4: Channel Data Check 5: Channel Control Check 6: Interf.Control Check 7: Chaining Check
6-7 Type Code and Logical Unit	Byte 6	
	X'0u' Original CCB X'08' Physical addressing is requested. Byte 7 contain the PUB index. X'2u' Translated CCB X'4u' BTAM request original CCB X'6u' BTAM request translated CCB X'8u' User-translated CCB in virtual partition Note: Any one of the above incremented by X'10'(bit 3 on) indicates automatic switching to the beginning of the next cylinder at End of Cylinder condition.	

Count	Transmission information	CSW Status Bits	Type Code	Reserved for logical IOCS	CCW Address	Reserved for physical IOCS	CCW Address in CSW	Optional Sense CCW
0 1	2 3	4 5	6 7	8	9 11	12	13 15	16 23

Byte(s)	Description
6-7 (....Cont'd)	Byte 6 (Cont'd) u: 0= The address in byte 7 refers to a System Logical Unit. 1= The address in byte 7 refers to a Programmer Logical Unit.
	Byte 7 Hexadecimal representation of SYSnnn: SYSRDR = 00 SYSREC = 0A SYSIPT = 01 SYSCLB = 0B SYSPCH = 02 SYSVIS = 0C SYSLST = 03 SYSCAT = 0D SYSLOG = 04 SYS000 = 00 SYSLNK = 05 SYS001 = 01 SYSRES = 06 SYS002 = 02 SYSSLB = 07 ⋮ SYSnnn SYSRLB = 08 (Note 9) SYSUSE = 09
8 Reserved for Logical IOCS	Buffer Offset: ASCII Input Tapes X'00'-X'63' ASCII Output Tapes Fixed X'00' Variable X'00' or X'04' Undefined X'00'
9-11 CCW Address	Virtual or real address of CCW associated with this CCB depending on byte 6: Real address if byte 6= X'2u', X'6u', or X'8u' ; Virtual address if byte 6= X'0u', or X'4u'.
12 Reserved for Physical IOCS	X'80' CCB being used by ERP X'40' Channel Appendage Routine present X'20' Sense Information desired (Note 10) X'10' Message writer X'08' EU Tape Error X'04' OLTEP Appendage available X'02' Tape ERP Read Opposite Recovery X'01' Seek Separation
13-15 CCW Address in CSW	Virtual Address of CCW pointed to by CSW at Channel End (if byte 6= X'8u', it is the real address) or address of the Channel End Appendage Routine for TP devices, VSAM or POWER.
16-23 Optional Sense CCW	8 bytes appended to the CCB when Sense Information is desired.

Note 1: Bytes 4 and 5 contain the status bytes of the Channel Status Word(Bits 32-47). If byte 2,bit 5 is on and device end results as a separate interrupt,device end will be OR-ed into CCB byte 4.

Note 2: Indicates /* or /& statement on SYSRDR or SYSIPT. Byte 4,bit 7 (unit exception) is also on.

Note 3: DASD data checks on count not returned.

Note 4: For 1255/1259/1270/1275/1419,disengage. For 1275/1419D, I/O Error is external interrupt routine (Channel data check or bus-out check).

Note 5: The traffic bit (Byte 2,bit 0) is normally set on at channel end to signify that the I/O was completed. If byte 2,bit 5 has been set on,the traffic bit and bits 2 and 6 in byte 3 will be set on at device end. Also see Note 1.

Note 6: 1018 ERP does not support the Error Correction Function.

Note 7: This error occurs as an equipment check,data check or FCB parity check. For 2245,this error occurs as a data check or FCB parity check.

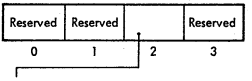
Note 8: For 3504,3505,3525 input or output files using ERROPT, byte 3-bit 3 is set on if a permanent error occurs. Byte 2-bit 6 is set on to allow you to accept permanent errors.

Note 9: SYSnnn= 255-(Number of partitions x 14).

Note 10: If user Error Routine is specified and the user needs the sense info to further process the error, byte 12-bit 2 must also be set. Otherwise the supervisor error routine will mask off the status on return and the sense info is not available.

Note 11: Byte 8, Buffer Offset : This is not used in EBCDIC files, and must than contain Hex 0 or 80, the latter indicating that double-CCW support for an IBM 2501 Card Reader is active.

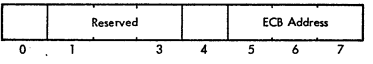
EVENT CONTROL BLOCK (ECB)



X'80' : Normal termination of subtask

X'C0' : Abnormal termination of subtask

RESOURCE CONTROL BLOCK (RCB)



Byte 0 : X'FF' if resource is in use, X'00' if resource is not in use

Bytes 1-3 : Reserved

Byte 4 : Bit 0= 1 Another task waiting for the resource
= 0 No other task waiting for the resource

Bytes 5-7 : ECB address of current resource owner

DENSITY DATA

Density (Bytes per inch)	Parity	Convert Feature	Translate	SS Code *
200	odd	on	off	10
200	odd	off	off	30
200	odd	off	on	38
200	even	off	off	20
200	even	off	on	28
556	odd	on	off	50
556	odd	off	off	70
556	odd	off	on	78
556	even	off	off	60
556	even	off	on	68
800	odd	on	off	90
800	odd	off	off	B0
800	odd	off	on	B8
800	even	off	off	A0
800	even	off	on	A8
800	dual density nine-track ----------			CB
1600	dual density nine-track ----------			C0
6250	dual density nine-track ----------			D0
800	single density nine-track----------			C0
1600	single density nine-track----------			C0
6250	single density nine-track----------			D0

* Refer to PUB Table, byte 5

SYSTEM CONTROL PROGRAM COMMUNICATION REGION (SYSCOM)

SYSCOM

Displacement Hex / Dec									
0 0	4 4	8 8	0A 10	0C 12	10 16	14 20	18 24		
Address of Error Block	Address of Attention Exit	Address of Operator Option Cancel Exit	Address of Operator Request Cancel Exit	Address of SYSRES PUB	Address of Fetch Routine	Address of I/O Interr. Routine	Address of Ext. Interr. Routine		
XXXX	XXXX	XX	XX	XXXX	XXXX	XXXX	XXXX		
1C 28	20 32	24 36	25 37	28 40	2A 42	2C 44	2E 46	2F 47	30 48
Address of Logical Transient Area	Address of 1st byte of Problem Program Area	Free List Pointer	Address of Channel Queue	Number of Channel Queue Entries	Length of One Error Queue Entry	Number of Partitions	Not used	Flag Byte (see expansion)	Address of Channel Buckets
XXXX	XXXX	X	XXX	XX	XX	XX	X	X	XXXX

34 52	35 53	38 56	3C 60	40 64	44 68	46 70	48 72	4C 76
Configuration Byte (see expansion)	Address of CRT Table	Address of SAB Table	Address of Channel Control Table	Flags and Switches (See expansion) *	System Task Selection Control Field *	Address of Task Selection	Address of PD Area	TH Free List Pointer
X	XXX	XXXX	XXXX	XXXX	XX	XX	XXXX	X

* See end of tables for further explanation

Note : The address of SYSCOM can be found at fixed location X'80' - X'83'.

4D 77	50 80	54 84	58 88	5A 90	5C 92	60 96	64 100	68 104
Address of TH Table	Address of Timer Request Table	Address of AB Table	Key of Task owning LTA (LIK)	Key of Task running (TIK)	Address of POWER/VS partition	Address of VTAM Address Vector Table	Address of RF Table	Address of EU ECB Table
xxx	xxxx	xxxx	xx	xx	xxxx	xxxx	xxxx	xxxx

6C 108	70 112	74 116	78 120	7C 124	80 128	84 132	88 136
Address of OLTEP bucket	Address of RAS Linkage Area	Address of ASCII Translate Table	Address of PUB Ownership Table	Address of Job Accounting Common Table	Base Address of Page Management Routine	Base Address of Channel Program Translation Routine	Address of SDAID Save Area
xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx

8C 140	90 144	94 148	98 152	9C 156	A0 160	A1 161	A2 162	A3 163
Address of Line Mode Table	Address of VSAM Communication Area	Address of PTA	Address of first System Task Block	Address of Task Block of Active System Task	1 byte for Alignment	Pointer to SENSE Task Block	Pointer to Disk ERP Task Block	Pointer to RAS Task Block
xxxx	xxxx	xxxx	xxxx	xxxx	x	x	x	x

A4 164	A5 165	A6 166	A7 167	A8 168	A9 169	AA 170	AB 171	AC 172	AF 175	B0 176
Pointer to PMGR Task Block	Reserved	Pointer to PAGEIN Task Block	Pointer to FETCH Task Block	Reserved	Pointer to CRT Task Block	Pointer to ASYN Task Block	Pointer to ERP Task Block	Reserved	Pointer to SVF Task Block	Address of Task Timer Table
x	x	x	x	x	x	x	x	xxx	x	xxxx

B4 180	B8 184	BC 188	BE 190	C0 192	CB 203	CC 204	CE 206	D0 208
Address of MVCFLD	TRTMSK pointer	TP Balancing Parameter	Key of partition owning Task Times	Repositioning Information for MFCM ERP	Number of Error Queue Entries	Length of PUB Table in bytes	Number of Active Partitions	Address of Segment Table
xxxx	xxxx	xx	xx	xxxxxxxxxx	x	xx	xx	xxxx

D4 212	D8 216	DC 220	E0 224	E4 228	E8 232	EC 236	F0 240
Address of Page Frame Table	Address of Page Frame Table Extension	Address of Boundary Box	Address of DPD Table	Reserved	Address of VIRTAD Routine	Address of End of Real Storage	Address of Fetch Table
xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx	xxxx

F4 244	F5 245	F8 248	FC 252	100 256	104 260
SVA Flag (see expansion) *	Address of SVA	Address of System GETVIS	Address of RPS local directory list	Address of RPS sector calculation routine	Address of System Code patcharea start address
x	xxx	xxxx	xxxx	xxxx	xxxx

* See end of tables for further explanation

Expansion of SYSCOM Flag Bytes

Byte Dec	Byte Hex	Description
64	40	Reserved for RMS in Model 115 and 125 X'80' RMSR for channel attached devices, tapes and TP devices X'40' Full RMS support (MCAR/CCH and RMSR) X'20' MCAR/CCH support
65	41	X'80' Initial selection of ERP X'40' Reserved X'20' Timer interrupt pending X'10' MICR Stacker-select active X'08' Invalid address during fetch X'04' SIO routine entered after interrupt X'02' TP in progress X'01' IPL in progress
66	42	X'80' Initial RAS request X'40' RAS WAIT request outstanding X'20' RAS IPL in progress X'10' Reserved X'08' POWER/VS supported X'04' POWER/VS initialized X'02' GETREAL for SDAID or PDAID in progress X'01' Fetch for system task in progress (used by PDAID's)
67	43	X'80' Reserved X'40' ECPREAL supported X'20' VSAM supported X'10' Reserved X'08' XECB support generated X'04' TP Balancing reset request X'02' Batch deactivated X'01' Reserved
244	F4	SVA Flag X'80' Do not test for warm start copy of SVA X'40' SDL active X'20' No "Set SVA" or "Set SDL" allowed X'10' Build of DSL in progress X'08' SDL overflow X'04' Reserved X'02' Reserved X'01' Reserved
252	FG	X'00000000' RPS not initialized X'00XXXXXX' Pointer to RPS LDC in SVA
256	100	X'00000000' No RPS support X'00XXXXXX' Pointer to Sector Calculation Routine

LAYOUT OF SYSTEM TASK SELECTION CONTROL FIELD

Byte Dec	Byte Hex	Description
68	44	Always zero
69	45	SELECT byte : X'00' No system task active X'01' SNS active X'02' DSK active X'03' RAS active X'04' PMGR active X'06' PGIN active X'07' SUP active X'09' CRT active X'0B' ERP active X'0F' SVT active

Note: The address of SYSCOM can be found at fixed location X'80'- X'83'.

LAYOUT OF FLAG BYTE

Byte Dec	Byte Hex	Description
47	2F	X'01' APRT support generated

LAYOUT OF CONFIGURATION BYTE

Byte Dec	Byte Hex	Description
52	34	Configuration Byte X'06' Support for 3277 generated X'02' Support for 125D generated X'00' Support for 3210/3215 generated

PARTITION COMMUNICATION REGION

nnCOMREG

Hex	0	8	0A	0C	17	18	20	24	28	2C
Dec	0	8	10	12	23	24	32	36	40	44
	Date	Address of PPBEG	Address of EOSSP	Problem Program Use	UPSI Byte	Job Name	Highest Storage Address of the Partition	End Address of Last Phase Fetched or loaded	Address of upper-most Byte of Phase with highest Ending Address	Label Area Length
	XXXXXXXX	XX	XX	XXXXXXXXXX	X	XXXXXXXX	XXXX	XXXX	XXXX	XX

Displacement											
2E	30	34	35	36	37	38	39	3A	3B	3C	3E
46	48	52	53	54	55	56	57	58	59	60	62
PIK	End of Virtual Storage Address	Machine Configur. Byte	System Configur. Byte 1	Standard Language Translator I/O Options	Dump, Log, RELLDR and ASCII Options	Job Control Byte	Linkage Control Byte	Language Translator Control Byte	Job Duration Indicator Byte	Disk Address of Label Cylinder	Address of FOCL
XX	XXXX	X	X	X	X	X	X	X	X	XX	XX

Job Control Switches

40	42	44	46	48	4A	4C	4E	4F	58	5A	5C
64	66	68	70	72	74	76	78	79	88	90	92
Address of PUBTAB	Address of FAVP	Address of JIBTAB	Address of TEBTAB	Address of FICL	Address of NICL	Address of LUBTAB	Line Count for SYSLST	System Date	LIOCS Comm. Byte	Address of 1st Part of PIB Table	ID Number of last Checkpoint or DASDFP Indicator
XX	XX	XX	XX	XX	XX	XX	X	XXXXXXXX	XX	XX	XX

Note: A communication region exists for each partition supported by the system
The address of the communication region of the active partition is in fixed loc. X'14' - X'17'

PARTITION COMMUNICATION REGION (....Cont'd)

5E 94	60 96	62 98	64 100	66 102	68 104	6A 106	6C 108	6E 110
Job Zone in Minutes	Address of Disk Information Block (DIB)	Reserved	Address of PC Option Table less 8 bytes	Address of IT Option Table less 8 bytes	Address of OC Option Table less 8 bytes	Key of Program with Timer Support	Reserved	Logical Transient Key (LTK)
xx	xx	xx	xx	xx	xx	xx	xx	xx

70 112	74 116	78 120	7C 124	7E 126	80 128	84 132	86 134	87 135
Address of SYSPARM	Address of J.A.Partition Table	Address of TOD clock Common Area	Address of 2d part of PIB Table	Address of MICR DTF Table(PDTABB)	Address of QTAM Vector Table	Address of BG Comm. Region	Option Indicator	System Configuration Byte 2 and RMSR Open Flag Byte
xxxx	xxxx	xxxx	xx	xx	xxxx	xx	x	x

88 136	8C 140	8D 141	8E 142	8F 143	97 151	98 152	9F 159
Reserved for compatability reasons	Standard Options	Temporary Options	Disk Configuration	Catalog Procedure Name	Switch for Catalog Procedure	JCL Statement Name	81 bytes SYSIN Indicator
xxxx	x	x	x	xxxxxxx	x	xxxxxx	x

Note: A communication region exists from each partition supported by the system.
The address of the communication region of the active partition is in fixed loc. X'14'-X'17'.

PARTITION COMMUNICATION REGION (...Cont'd)

A0 160	A4 164	A5 165
Address of POWER/VS Partition Control Block	POWER/VS Flag Byte 1	POWER/VS Flag Byte 2
xxxx	x	x

Note: A communication region exists for each partition supported by the system
The address of the communication region of the active partition is in fixed loc. X'14' - X'17'

Key to Communication Region displacement

0	MM/DD/YY or DD/MM/YY either set permanently by the job control date statement, or updated every time a GETIME macro is issued when time-of-day support is provided. Format controlled by BGCOMREG + 53. (System Configuration Byte, date convention bit 0)
8	Address of the problem program area. } Maintained for compatibility; correct value found in SYSCOM at displacement hex 20
10	Address of the beginning of the problem program area. Y(EOSSP) equals Y(PPBEG) }
12	User area. If seek separation option is specified, bytes 12 and 13 are used at IPL time for the address of the seek address block.
23	User program switch indicator.
24	Job name set by the job control program from information found in the job statement.
32	Address of the uppermost byte available to the problem program, that is either the address of the uppermost byte of the partition as determined during processing of the ALLOC or ALLOCR macro or statement, or the end address of the area specified by the SIZE parameter in the EXEC statement.
36	Address of the uppermost byte of the last phase of the problem program fetched or loaded. Not filled in when phase is in SVA.
40	Highest ending main-storage address of all phases having the same first four characters as operand on the EXEC statement. For the phase $LNKEDT this field is not filled in. The address value may be incorrect if the program loads any of these phases above its link-edited origin address. If the EXEC statement has no operand, job control places in this location the ending address of the program just link-edited.
44	Length of the problem program label area.
46	Partition Identification Key (PIK). The low order byte identifies the active partition. Only significant for BG communication region.
48	End address of virtual storage.

Expansion of COMREG Flag Bytes

Byte Dec	Byte Hex	Description
52	34	Machine Configuration Byte (Values set as supervisor generation time) X'80' Always set to indicate standard storage protect X'40' Decimal feature (always set) X'20' Floating point feature X'10' Physical transient overlap option X'08' Always set to indicate standard timer feature X'04' Channel switching device X'02' Burst mode on multiplex channel support X'01' Indicates MCH/CCH in system
53	35	System Configuration byte X'80' DDMMYY (Date convention bit set at generation time by STDJC) X'40' Multiprogramming environment (2 or more partitions) X'20' DASD file-protect supported X'10' DASD SYSIN - SYSOUT X'08' Teleprocessing X'04' Two or more partitions X'02' Asynchronous processing X'01' Track Hold
54	36	This byte contains the standard language translator I/O options (set by STDJC macro). X'80' DECK option — output object modules on SYSPCH X'40' LIST option — output source module listings and diagnostics on SYSLST X'20' LISTX option — output hexadecimal object module listings on SYSLST (compilers only) X'10' SYM option — output symbol tables on SYSLST/SYSPCH X'08' XREF option — output symbolic cross-reference list on SYSLST X'04' ERRS option — output diagnostics on SYSLST (compilers only) X'02' CHARSET option — input on SYSIPT is 48 or 60 character set X'02' Reserved

Expansion of COMREG Flag Bytes

Byte Dec	Byte Hex	Description
55	37	This byte contains the standard supervisor options for abnormal EOJ, Relocating Loader and Control statement display and the indicator for the presence of the ASCII-EBCDIC and EBCDIC-ASCII translation tables. X'80' Always on X'40' DUMP option (DUMPXYES OR DUMP=PART) X'20' yes, dump registers and storage on SYSLST partition in wait state, because a volume is to be mounted X'10' LOG option X'08' yes, list all control statements on SYSLST dummy, device search in progress; do not enter ERP X'04' Not used X'02' Relocating Load option yes, Relocating Loader supported X'01' ASCII option yes, ASCII supported
56	38	Job Control byte X'80' Job accounting interface (JA) not supported X'40' Return to caller on LIOCS disk open failure X'20' Job control input from SYSRDR X'10' Job control output on SYSLOG X'08' Cancel job X'04' Pause at end-of-job step X'02' SYSLOG is not a console printer-keyboard or DOC X'01' SYSLOG is assigned to the same device as SYSLST
57	39	Linkage control byte X'80' SYSLNK open for output X'40' Update of Second Level Directory and RAS loadlist in progress (interface between $MAINDIR and Supervisor) X'20' Allow EXEC X'10' Catalog linkage editor output X'08' Supervisor has been updated X'04' PCIL open in progress X'02' Update of System Core Image Library in progress (interface between $MAINDIR and Supervisor) X'01' Check automatic condens limits at end of job (interface between Librarian and Job Control)

Expansion of COMREG Flag Bytes

Byte Dec	Byte Hex	Description
59	3B	Job duration indicator byte X'80' Within a job condition X'40' Dump or partition dump on an abnormal EOJ condition (see byte 141) X'20' Pause at EOJ step Set by attention routine for Job Control X'10' Job control output on SYSLST X'08' Job is being run out of sequence with a temporary assignment for SYSRDR X'04' PCIL is being condensed X'02' //DATE statement processed for current job X'01' Batch command just issued
93	5D	Used at IPL time and set by PIOGS X'80' 3330 file protection X'40' 3340 file protection X'20' 3350 file protection
134	86	Option Indicator byte X'80' Reserved X'40' = EU interface active X'20' = Teleprocessing request X'10' = Supervisor support for tape X'08' Reserved X'04' = RETAIN support generated X'02' = Linkage to Channel End Appendage Routine allowed X'01' = GETVIS function has been initiated
135	87	System Configuration byte 2 and RMSR Open Flag byte X'80' PCIL supported X'40' TOD clock supported X'20' PFIX macro supported X'10' Fetch $$BOPEN by $JOBCTLJ X'08' Fetch $$BOPEN by $JOBCTLD X'04' Fetch $$BOPEN by $JOBCTLJ for WAITM X'02' Reserved X'01' RPS supported

Expansion of COMREG Flag Bytes

Byte Dec	Byte Hex	Description
140	8C	Standard Option byte X'80' = EDECK Standard Option X'40' = ALIGN Standard Option X'20' = PARTDUMP Standard Option X'10' = RLD Standard Option X'02' Reserved X'01' = ACANCEL Standard Option
141	8D	Temporary Option byte X'80' =EDECK Temporary Option X'40' =ALIGN Temporary Option X'20' =PARTDUMP Temporary Option X'10' =RLD Temporary Option Not used (Bit 4 and 5) X'02' SUBLIB=DF Temporary Option X'01' 1=ACANCEL Temporary Option
142	8E	Disk Configuration Byte 0 - 3 Not used X'08' 3350 supported X'04' 3340 supported X'02' 3330 supported X'01' Always 1: indicates 2311 and 2314/2319 supported.
151	97	Interface byte for Catalogued Procedures X'80' Procedure being executed X'40' Overwrite processing X'20' Procedure with data X'10' Overwrite request for Job Control X'08' Insert request for Job Control X'04' Procedure end X'02' SYSLOG procedure X'01' Overwrite request for Supervisor
159	9F	SYSIN 81 bytes indicator X'80' Permanent 81 bytes on SYSRDR X'40' Permanent 81 bytes on SYSIPT X'20' Temporary 81 bytes on SYSRDR X'10' Temporary 81 bytes on SYSIPT Not used X'01' Allow/& for MAINT CATALS
164	A4	POWER/VS Flag Byte 1 X'80' POWER/VS Accounting Supported X'40' Partition under control of POWER/VS X'20' POWER/VS Partition Reserved
165	A5	POWER/VS Flag Byte 2 (Not used)

PARTITION IDENTIFICATION KEY (PIK)

Task or Partition	PIK value				
	Number of partitions supported				
	5	4	3	2	1
Attention	X'00'	X'00'	X'00'	X'00'	X'00'
BG	X'10'	X'10'	X'10'	X'10'	X'10'
F4	X'20'				
F3	X'30'	X'20'			
F2	X'40'	X'30'	X'20'		
F1	X'50'	X'40'	X'30'	X'20'	

TASK INTERRUPT KEY (TIK)

Task	TIK value			
	Number of partitions supported			
	5	4	3	2
Attention	X'00'	X'00'	X'00'	X'00'
BG	X'10'	X'10'	X'10'	X'10'
F4	X'20'			
F3	X'30'	X'20'		
F2	X'40'	X'30'	X'20'	
F1	X'50'	X'40'	X'30'	X'20'
Subtask*	X'60'-X'F0'	X'50'-X'F0'	X'40'-X'F0'	X'30'-X'F0'

* If AP=YES

LOGICAL TRANSIENT OWNER IDENTIFICATION KEY (LIK)

The halfword LIK at displacement 88 in SYSCOM contains the same value as the TIK when the Logical Transient Area (LTA) is in use and therefore identifies the owner of the LTA. When LTA is free, the halfword LIK contains zero's. The SVC2 routine sets the LIK, and the SVC11 routine resets it to zero. If AP is not supported, the LIK contains zero's.

LOGICAL TRANSIENT KEY (LTK)

The halfword LTK at displacement 110 in each partition communication region has a zero value in the high-order byte and a key value in the low order byte. In a foreground communication region, the key value in the LTK is not significant. The LTK in the background communication region(BGCOMREG) has the same value as the PIK of the partition of the task that owns the LTA, or contains zero's when the LTA is free. The SVC2 routine sets the LTK, and the SVC11 routine resets it to zero.

RAS LINKAGE AREA

RASLINK

Hex	0	8	9	0A	0B	0C	10	14	18	1C
Dec	0	8	9	10	11	12	16	20	24	28
	CPU ID Field	Damaged Channel Byte	RAS Flag Byte	Machine Check Flags	Largest CPU Model	RAS Table (RASTAB) Address	Base Address for RAS Monitor	Address of Headqueue PUB Table	Address of CCH Savearea	Address of Page Queue
	xxxxxxxx	x	x	x	x	xxxx	xxxx	xxxx	xxxx	xxxx

Key to RAS Linkage Area displacement

Displacement

0 CPU ID Field

8 Address of damaged channel, or X'FF' if no channel damaged

9 RAS Flag Byte

bit	flag	description
0	X'80'	RAS active
1	X'40'	RAS SIO flag
2	X'20'	RTA in control
3	X'10'	RAS I/O delayed
4	X'08'	Channel check on error SIO
5	X'04'	Reserved
6	X'02'	Channel check on SIO
7	X'01'	I/O active for SIO

10 Machine Check Flags

bit	flag	description
0-4		Reserved
5	X'04'	Hard machine check
6	X'02'	All machine records built
7	X'01'	All channel check records built

11 Largest CPU Model

12 Address of RAS MOnitor Table (RASTAB)

16 Address used for base register in RAS Monitor Program

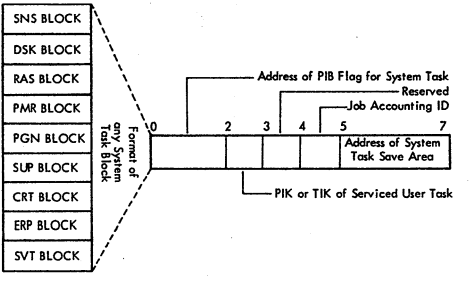
20 Address of Headqueue PUB Table

24 Address of CCH Savearea

28 Address of Page Queue Table

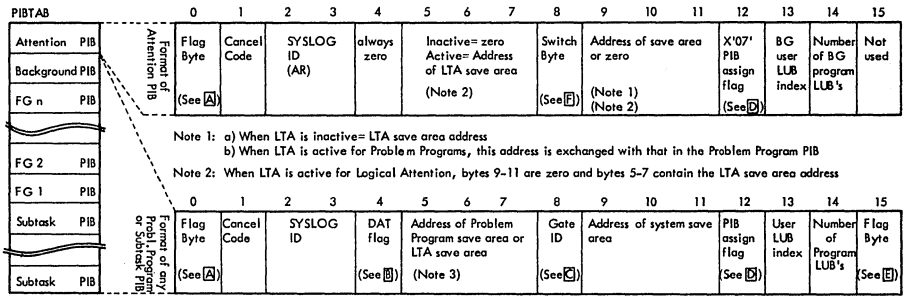
Bytes 112-115 (X'70'-X'73') of the System Communication Region (SYSCOM) contain the address of the area.
Label RASLINK indentifies the first byte of the area.

SYSTEM TASK BLOCKS



Notes: Bytes 152-155 (X'98'-X'9B') of the System Communication Region (SYSCOM) contain the address of the first System Task Block.

PROGRAM INFORMATION BLOCK (PIB) - First part



Note 1: a) When LTA is inactive= LTA save area address
b) When LTA is active for Problem Programs, this address is exchanged with that in the Problem Program PIB

Note 2: When LTA is active for Logical Attention, bytes 9-11 are zero and bytes 5-7 contain the LTA save area address

Note 3: When the Logical Transient Area is active the save area address in the Problem Program PIB is exchanged with that in the Attention PIB.

The number of Problem Program PIB's generated depends on the number of partitions specified during system generation.
Subtask PIB's are generated only if AP= YES has been specified during system generation.
The number of subtask PIB's generated depends on the number of partitions, it is 15 minus the number of partitions. For example, in a four partition system the number of subtask PIB's generated is 11.

Bytes 90-91 (X'5A'-X'5B') of the partition communication region(s) contain the address of the first part of the PIB table. Label PIBTAB identifies the first byte of the table.

A FLAG BYTE (First byte in PIB)

Flags which are always used:

X'71'= Program is waiting for SVC58
X'73'= Program is waiting because system is seized
X'75'= Program is waiting for copy block
X'77'= Program is waiting for TFREE
X'79'= Program is waiting for channel queue entry
X'7B'= Program is waiting for CCW translation
X'7F'= Program is waiting for XECB table
X'80'= Program is not active
X'81'= Program is SVC2-bound (waiting for the LTA to be released)
X'82'= Program is SVC7-bound (waiting for an I/O interruption)
X'83'= Program is ready to run
X'85'= Program is SVC5-bound (waiting for the PTA to be released)
X'86'= Initial selection of RAS (used only for RAS PIB flag)
X'87'= Program is set to common-bound condition

Flags used only under certain conditions:

Flags with partition-dependent values:

The table below shows the various possible values of these flags and the partition to which a given value refers, depending on the number of partitions. The meanings of the flags types A, B, C, and D are explained below the table.

Flag				Partition referenced			
				NPARTS =			
A	B	C	D	2	3	4	5
X'29'	X'3D'	X'4B'	X'59'	BG	BG	BG	BG
X'2B'	X'3F'	X'4D'	X'5B'	F1	F2	F3	F4
X'2D'	X'41'	X'4F'	X'5D'		F1	F2	F3
X'2F'	X'43'	X'51'	X'5F'			F1	F2
X'31'	X'45'	X'53'	X'61'				F1

Flag A is used only if AP=YES and VSAM=YES. The codes are used to gate the CDLOAD routine for tasks running in one partition.

Flag B is used only if AP=YES and GETVIS=YES. The codes are used to gate the GETVIS routine for tasks running in one partition.

Flag C is used only if AP=YES and PFIX=YES. The codes are used by the PFIX routine to set a partition PFIX-bound.

Flag D is used only if PARTS 1. The codes are used by the load leveller to deactivate a partition.

Flags with partition-independent values:

The following flags are used only if NPARTS 1:

X'6A'= Program is SVC35-bound
X'6B'= Program is SVC35-bound — only if TRKHLD=n
X'6D'= Program is waiting for the next freed page frame.

A Flag Byte (Cont'd)

The following flags are used only if AP=YES:
X'67' = Program is SVC38-bound
X'69' = Program is SVC41/42-bound

The following flag is used only if CBF=n:
X'7D' = Program is waiting for free console buffer table entry

The following flag is used only if TP=QTAM:
X'8B' = Task in QTAM wait

The following flag is used only if IDRA=YES or DOC=125D/3277:
X'6F' = Program is IDRA-bound

The flags X'37', X'39', and X'3B' are used if TP=VIAM to gate the SVC53 routine.

Explanation of the contents of an entry in the PIB

B PIB DAT Flag

X'01' = Return to re-entrant supervisor routine
X'02' = Return to gated supervisor routine
X'04' = Move CCB at dispatching time
X'08' = Service delayed external interrupt
X'10' = Deactivation for this task is being delayed
X'20' = Operator communications to be serviced
X'40' = Task has seized the system
X'80' = Program is running in virtual mode

C Gate Identifier

X'71' = Gating of SVC58 required
X'53' = Gating of SVC41/42 required

The flags are only used if the PIB DAT Flag is X'03', that is, the first two flags are on (See B).

D PIB Assign Flag

X'80' = SYSRES DASD file protect inhibited (allow write operation on SYSRES)
X'40' = Channel appendage exit allowed (BTAM)
X'20' = Cancel in progress (used in terminator function)
X'10' = Cancel control (set on a foreground cancel)
X'08' = Hold foreground assignments
X'07' = Attention PIB

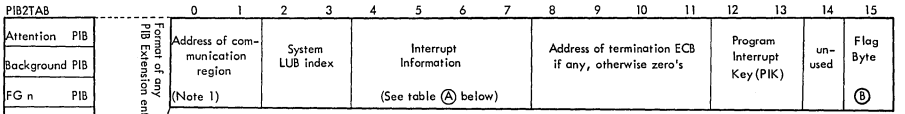
E Problem Program PIB Flag (Last byte in PIB)

Bit 0: 1=Batched job in foreground (has no function in DOS/VS)
Bit 1: 1=Cancel in LTA and device not assigned
Bit 2: 1=/& on SYSIN if DASD
Bit 3: 1=Partition in stopped state
Bit 4: 1=Fetch EOJ monitor
Bit 5: 1=Task is canceled
Bit 6: 1=Subtask(s) attached
Bit 7: 1=in AB routine

F Attention PIB Switch Byte

Bit 0: Reserved
Bit 1: 1= Fetch Physical Attention Transient $$ABERRZ
Bit 2: 1= Delay cancelation (fetch $$ABERZ1)
Bit 3: 1= Emergency cancel request
Bit 4: Reserved
Bit 5: 1= Command available (DOC)
Bit 6: 1= Fetch Logical Attention Routine ($$BATTNA)
Bit 7: 1= External Interrupt request

PROGRAM INFORMATION BLOCK EXTENSION - Second Part



Note 1: Always BG communication region in Attention- and Background PIB extension. Appropriate communication region in other PIB extensions when a multiprogram system has been generated.

For each PIB Table entry (first part) an entry exists in the PIB Table Extension (second part).

Bytes 124-125 (X'7C'-X'7D') of the partition communication region(s) contain the address of the PIB Table extension. Label PIB2TAB identifies the first byte of the PIB Table extension.

Ⓐ

Type of interruption	Contents of PIB Extension Bytes			
	4	5	6	7
SVC	00	ILC *)	Interruption Code	
PC	00	ILC *)	Interruption Code	
I/O	00	00	I/O Address	

*) ILC (Instruction Length Code) is in bits 5 and 6, other bits are zero's.

Ⓑ

Byte 15	
Bits 0 -1 :	Reserved
Bit 2 :	1=Task owns CRT
Bit 3 :	VSAM automatic Close in progress
Bit 4:	Not used
Bit 5:	1= SVC screening
Bit 6:	NOT used r interval set
Bit 7 :	+ ECB - Task has issued SVC 92, 93 or 94.

SAVE AREAS

Layout of LTA and Problem Program Save Area

Note 1		PSW		Reg 9	Reg A
Reg B	Reg C	Reg D	Reg E	Reg F	Reg 0
Reg 1	Reg 2	Reg 3	Reg 4	Reg 5	Reg 6
Reg 7	Reg 8	Note 2		FLTPT R0	
FLTPT R2		FLTPT R4		FLTPT R6	

Note 1 : Problem Program Save Area : Program Name

Note 2: Problem Program Save Area: First half-word : Label area length
Second half-word : Reserved
Third and fourth half-word: Job start time

LTA Save Area: Reserved

Layout of User Save Area (for AB, PC, OC and IT options)

Interrupt Status Information *		Reg 0	Reg 1	Reg 2	Reg 3
Reg 4	Reg 5	Reg 6	Reg 7	Reg 8	Reg 9
Reg A	Reg B	Reg C	Reg D	Reg E	Reg F

* Interrupt Status Information :

Byte 0 : Unused

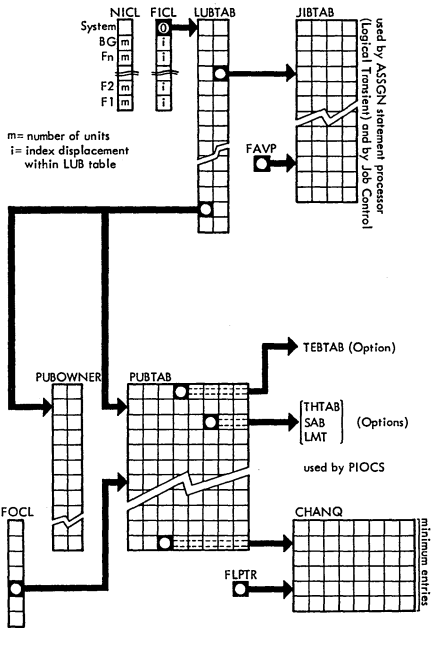
Byte 1 : Protection key and Mask (from PSW byte 1)

Bytes 2-3 : Interruption Code

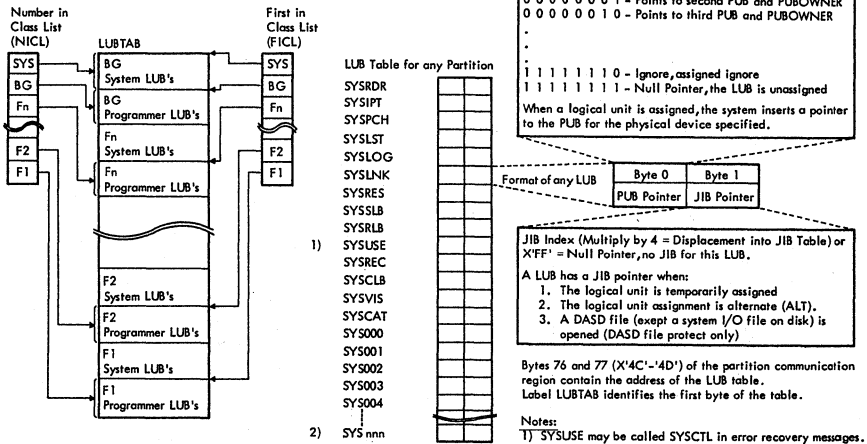
Byte 4 : Bits 0-1 : Instruction Length Code
Bits 2-3 : Condition Code
Bits 4-7 : Program Mask

Bytes 5-7 : Instruction Address

I/O TABLE INTERRELATIONSHIP



LOGICAL UNIT BLOCK (LUB) TABLE

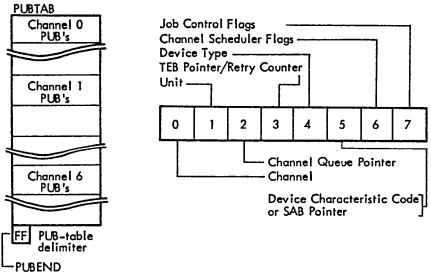


Notes:

1) SYSLUSE may be called SYSCTL in error recovery messages.

2) The value that may be specified for the highest programmer logical unit in the system is: SYS241 if NPARTS= 1, SYS226 if NPARTS=2, SYS212 if NPARTS=3, SYS198 if NPARTS=4, or SYS184 if NPARTS=5.

PHYSICAL UNIT BLOCK (PUB) TABLE



Byte 0: Channel number. (Hex 0-6, FF= NULL)

Byte 1: I/O device unit number

Byte 2: Hex 0, 1, 2, ... points to the first channel queue entry for this device

Byte 3: If device is a 2495 Tape Cartridge Reader and TEB's are specified, this byte is a TEB pointer. (Hex 1, 2, 3, ...). Otherwise this byte is a retry counter.

Byte 4: Section 2 for device type codes.

Byte 5: SS of the MODE= parameter in the DVCGEN macro for tape unit. (See Section 2)

For an ICA line Model 115 or 125, this byte contains the displacement index of the entry in the Line Mode Table (LMT). The address of the LMT is in SYSCOM.

For DASD with seek separation, this byte is used as the SAB Pointer. With Track Hold but not seek separation supported, this byte contains a pointer to the Track Hold Table entry or X'FF' (with both SKSEP and TRKHLD specified, the track hold pointer is found in the SAB entry).

For MICR type devices, this byte indicates which external interrupt line is in use.

For a 3704/3705 Communications Controller, this byte contains the type number of the Channel Adapter

For 2560 or 5425 :

Bit 0: 1= Repositioning required
1: 0= SYSPCH temporarily assigned to hopper 1
1= SYSPCH temporarily assigned to hopper 2
2: 0= SYSIPT temporarily assigned to hopper 1
1= SYSIPT temporarily assigned to hopper 2
3: 0= SYSRDR temporarily assigned to hopper 1
1= SYSRDR temporarily assigned to hopper 2
5: 0= SYSPCH permanently assigned to hopper 1
1= SYSPCH permanently assigned to hopper 2

Byte 5: (Cont'd)
- Bit 6: 0= SYSIPT permanently assigned to hopper 1
  1= SYSIPT permanently assigned to hopper 2
- 7: 0= SYSRDR permanently assigned to hopper 1
  1= SYSRDR permanently assigned to hopper 2

Byte 6: Channel Scheduler Flags

- Bit 0: 1= Device busy
- 1: 1= Switchable device
- 2: Reserved
- 3: 1= I/O error queued for recovery
- 4: 1= Operator intervention required
- 5: 1= Device End posting required
- 6: 1= Burst or overrunnable device
- 7: 1= 7-track tape unit

Byte 7: Job Control Flags

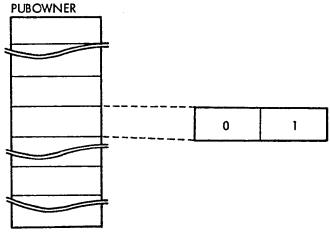
- Bit 0-4: Standard MODE assignment for 7-track tape (all ones if not tape, all zero's if device is down)
- 5: 1= Device supports RPS
- 6-7: B'11' (both on)= Headqueue in progress
  B'01' = Headqueue requested

Notes: A null is generated for each device to be supported by the supervisor. Standard physical unit assignments are made to the PUB table at supervisor generation time.
PUB's are ordered by channel and priority within a channel.

An entry in the PUB Ownership Table is associated with each entry in the PUB Table, if the supervisor has been generated to support multiprogramming.

Bytes 64 and 65 (X'40'-'41') of the partition communication region contain the address of the PUB Table entry. Label PUBTAB identifies the first byte of the table.

PHYSICAL UNIT BLOCK OWNERSHIP TABLE



Byte 0: Bit 0: 1= Volume is owned by VTAM
Bit 1: 1= waiting for volume to be mounted
Bit 2-7: Reserved

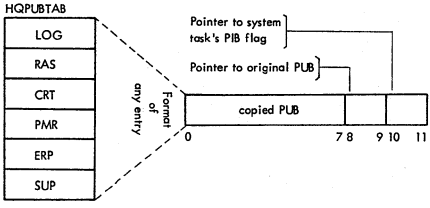
Byte 1: Identifies the partition that owns the PUB according to following table:

Flag	Partition owning PUB if number of partitions is:			
	2	3	4	5
X'01' X'02' X'04' X'08' X'10'	BG F 1	BG F 2 F 1	BG F 3 F 2 F 1	BG F 4 F 3 F 2 F 1
X'80'	PUB owned by VTAM			

Note: The number of entries in the PUB Ownership table is equal to the number of entries in the PUB table. Associated with each PUB entry is an entry in the PUB Ownership table.

Bytes 120-123 (X'78'-'7B') of the system communications region(SYSCOM) contain the address of the PUB Ownership table.
Label PUBOWNER identifies the first byte of the table.

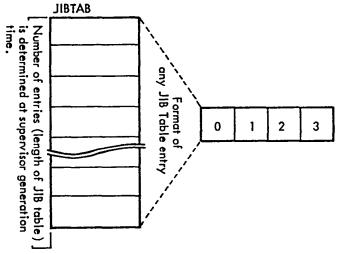
HEADQUEUE PUB TABLE



The Headqueue Table has a maximum of 6 entries (for RAS, CRT, ERP, PMR and SUPV system tasks). The entries for the RAS and CRT tasks are not generated when these functions are not supported. The LOG headqueue entry is only generated for models 115 and 125.
The order of the entries determines the headqueueing priority of the system tasks.

Bytes 20-23 (X'14'-X'17') of the RASLINK area contain the address of the Headqueue PUB Table.
Label HQPUBTAB identifies the first byte of the table.

JOB INFORMATION BLOCK (JIB) TABLE



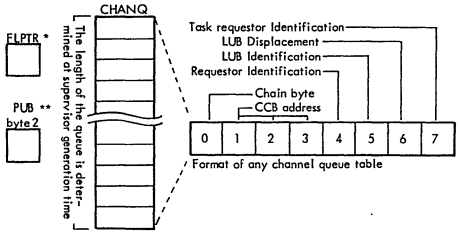
Byte(s)	Description	
	Bit setting Byte 2	Contents
0-1 (Contents depends on the bit setting in byte 2)	Bit 0= 1 Stored standard assignment	LUB entry of stored standard assignment. (PUB and JIB pointer)
	Bit 1= 1 Alternate assignment	Byte 0 : PUB pointer Byte 1 : X'00'
	Bit 2= 1 2311/2314/ 2319 Extent	Byte 0 : Cylinder lower limit Byte 1 : Cylinder upper limit *) Note 1
	Bit 3= 1 2321/3330/ 3340/3350 Extent	For 2321 : Lower limit (Cell or combined subcell and strip) or Upper limit (Cell or combined subcell and strip) For 3330 / 3340 / 3350 Cylinder lower limit or Cylinder upper limit (One cylinder number uses two bytes) *) Note 1 *) Note 2
2	Meaning if bit = 1 : Bit 0 : Stored standard assignment Bit 1 : Alternate assignment Bit 2 : 2311/2314/2319 extent Bit 3 : 2321/3330/3340/3350 extent Bit 4 : The alternate assignment indicated in bit 1 is permanent. This bit is also on when one of the extent indicators (bit 2 and bit 3) is on. Bit 5 : Catalogued procedures processing Bit 6/7: Reserved	

Byte(s)	Description
3	Chain byte Contains the displacement index of the next JIB. X'FF' defines the end of the chain.

* Note 1 : Only when file-protect on DASD.
* Note 2 : Two JIB's are required for a 2321/3330/3340/3350 extent; one for lower limit and one for upper limit.
The lower limit defining JIB must be chained to the upper limit defining JIB.
For 2321, byte 1 of this type JIB contains the subcell number times 10 plus the strip number in binary.

Bytes 68-69 (X'44'-'45') of the partition communication region contain the address of the JIB table entry. Label JIBTAB identifies the first byte of the table.

CHANNEL QUEUE TABLE



Byte 0: Contains the displacement within the channel queue of the next entry in the free list or in the list for a specific device, or X'FF' when it is the last entry in the free list or a device list.

Bytes 1-3: Contains the CCB address for the specified device.

Byte 4: Contains a code identifying the task making the I/O request. This one byte entry indicates to which task the CCB belongs, and is in the form X'nk' where:
n= user storage protection key. (Attention or system task= 0, BG task= 1, FG tasks= 2-5 depending on the number of partitions).
k= 0 for Attention and all user tasks.
1 for SNS
2 for DSK
3 for RAS
4 for PMGR
6 for PAGEIN
7 for SUPVR
9 for CRT
B for ERP
F for SVT

nK= FF for any unused channel queue entries.

Byte 5: Contains pointer(displacement index) to the entire LUB table identifying the logical unit making the I/O request. This is doubled to get the actual displacement into the full LUB table.

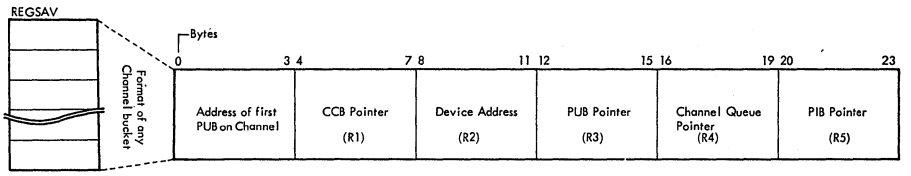
Byte 6: Contains X'FF' if the LUB is a programmer class, or the displacement within the partition LUB if it is a system class.
(Not used by a non MPS Supervisor).

Byte 7: Contains the displacement within the PIB table of the PIB of the task requesting I/O, or X'FF' when the channel queue entry is in the free list.
(Not used by a non MPS Supervisor).

* FLPTR: The free list pointer contains the index within the channel queue of the first entry in the free list or X'FF' when the channel queue is full. Byte 36 (X'24') of the System Communication Region (SYSCOM) contains the Free List Pointer. Label FLPTR identifies the location of the pointer (1 byte)

** PUB byte 2: The PUB channel queue pointer contains the index within the channel queue of the first entry for a specific device.
Bytes 37-39 (X'25'-X'27') of the System Communication Region (SYSCOM) contain the address of the Channel Queue table entry. Label CHANQ identifies the first byte of the table.

CHANNEL BUCKETS



A channel bucket contains information related to the last I/O started on the channel.

The number of channel buckets in a system equals the number of I/O channels in the system.

Bytes 48-51 (X'30'-X'33') of the System Communication Region (SYSCOM) contain the address of the Channel Buckets.
Label REGSAV identifies the first byte of the table.

CROSS PARTITION ECB (XECB) TABLE

XECB name	Access code	XECB address	Task ID of owner	Task ID of waiter/ poster
0 7	8	9 11	12 13	14 15

Byte 8

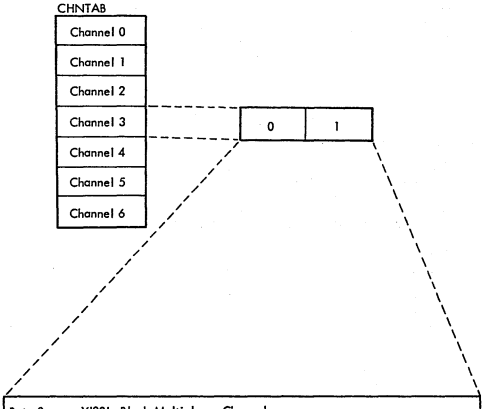
Parameterlist for TYPE = DEFINE

Bytes 0 - 7: XECB name
Byte 8: X'0n', bits 4 - 7 contain the access code

Parameterlist for TYPE = DELETE or DELETALL or RESET or CHECK

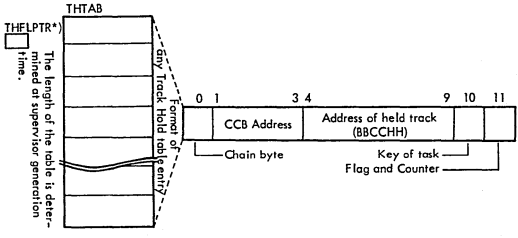
Bytes 0 - 7: XECB name
Bytes 8 - 9: X'1000' TYPE = RESET
X'4000' TYPE = DELETE
X'4800' TYPE = DELETALL
X'8000' TYPE = CHECK

CHANNEL CONTROL TABLE



Byte 0:	X'20'= Block Multiplexer Channel X'10'= Byte Multiplexer Channel X'11'= Byte Multiplexer Channel with burst mode support X'13'= Byte Multiplexer Channel running in burst mode X'00'= Selector Channel X'80'= Channel not operational or not present in the system
Byte 1:	Always zero
Note:	Byte 60-63 (X'3C'-'3F') of the system communications region(SYSCOM) contain the address of the Channel Control Table. Label CHNTAB identifies the first byte of this table.

TRACK HOLD (THTAB) TABLE

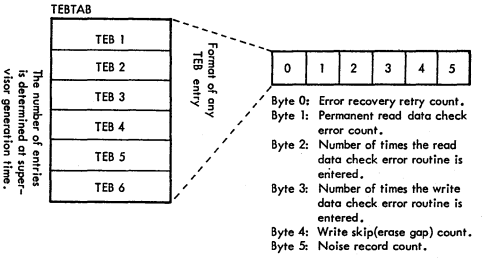


Byte 0:	Initially, pointer to next sequential entry or X'FF'(table delimiter)in last entry. After requests have been issued, this byte points to the entry for the next request for a track on the same device, or contains X'FF' (in entry for the last request), or it is a pointer in the free list chain.	
Bytes 1-3:	Address of CCB associated woth the task requesting the hold.	
Bytes 4-9:	Disk address of the track being held (in the form BBCCHH).	
Byfe 10:	Key of the task owning the track.	
Byte 11:	Bit 0=1:	Indicates a task is waiting for this track
	Bits 1-3:	Unused
	Bits 4-7:	Counter of number of holds on the track. (The number of holds is one more than this value).
*) Note:	THFLPTR:	The track hold free list pointer (1 byte) contains the displacement within the Track Hold table of the first entry in the free list or X'FF' when the Track Hold table is full.

Bytes 77-79 (X'4D'-'4F') of the system communication region (SYSCOM) contain the address of the Track Hold table. Label THTAB identifies the first byte of the table.

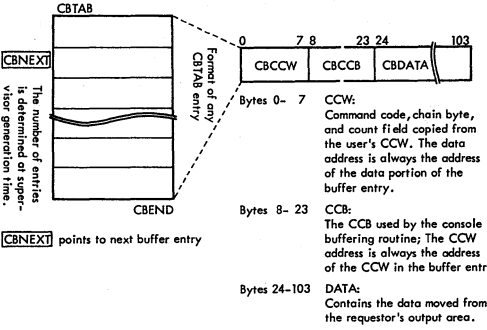
Byte 76 (X'4C') of the System Communication Region (SYSCOM) contains the address of the Track Hold Free List Pointer. Label THFLPTR identifies the location of the pointer.

TAPE ERROR BLOCK (TEB) TABLE



One TEB is generated for each 2495 Tape Cartridge Reader unit if the FOPT macro contains the TEB=n parameter. Job Control resets each TEB at normal or abnormal End-of-Job. An unused TEB contains HEX'FF0000000000'. A TEB is referenced from byte 3 of a Tape Cartridge Reader unit PUB.
Bytes 70 and 71 (X'46'-'47') of the partition communications region(s) contain the address of the TEB Table entry. Label TEBTAB identifies the first byte of the table.

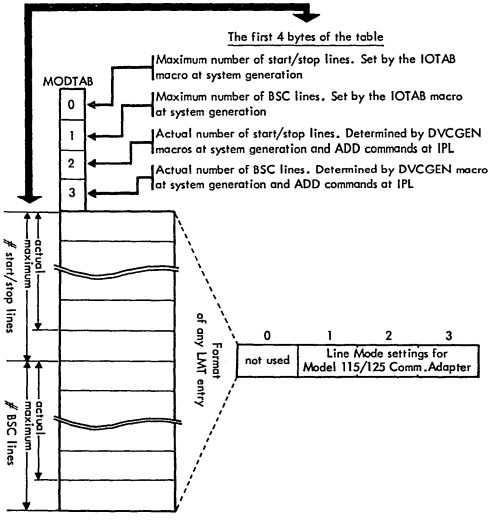
CONSOLE BUFFERING TABLE



CBNEXT points to next buffer entry

Label CBTAB identifies the first byte of the Console Buffer.

LINE MODE TABLE



Bytes 140-143 (X'8C'-X'8F') of the System Communication Region (SYSCOM) contain the address of the table.

Label MODTAB identifies the first byte of the table.

PAGE DATA SET TABLE (DPDTAB)

Extent Information	Number of Pages supported	Volume ID for Page Data Set	Address of Constant Table for Load Leveller
0 11	12 13	14 19	20 23

Extent Information (Bytes 0-11)

Bytes 0 and 1 : Channel and Unit number of SYSVIS Device

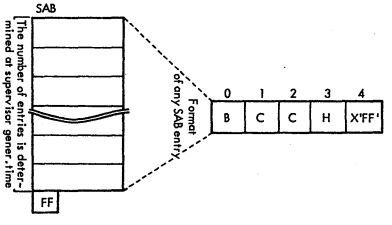
Bytes 2 and 3 : Internal device code of SYSVIS Device (offset in Page Manager Device Constants List) 0= 2314 , 4= 3330 , 8=3340, C=3350

Bytes 4 - 7 : Relative track address of Lower Limit

Bytes 8 - 11 : Relative track address of Upper Limit

Bytes 224-227 (X'E0'-X'E3') of the System Communication Region (SYSCOM) contain the address of the DPDTAB.
Label DPDTAB identifies the first byte of the table.

SEEK ADDRESS BLOCK (SAB)



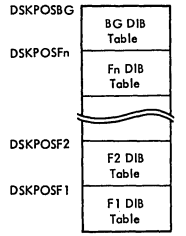
Bytes 0-3 : Current Disk Address (BCCH)

Byte 4 : X'FF' or: Pointer to the Track Hold table entry

Note: A SAB table is only generated if Seek Separation has been specified at supervisor generation time.

Bytes 56-59 (X'38'-X'3B') of the System Communication Region (SYSCOM) contain the address of the SAB Table.
Label SAB identifies the first byte of the table.

DISK INFORMATION BLOCK (DIB) TABLE



The number of DIB tables depends on the number of partitions specified at supervisor generation.

Format of any DIB table if SYSFIL= YES

	0						6	7		9	10							16	17	18	19	20		21	22	23
	Current Address							K	D	D	End Address								UL	LL	M	RC				
SYSLNK	C	C	H	H	00	00	00	P	P	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
SYSIN	B	B	C	C	H	H	R	00	00	50	B	B	C	C	H	H	R	H	H	XX	XX	XX	00	00		
SYSPCH	B	B	C	C	H	H	R	00	00	51	B	B	C	C	H	H	R	H	H	XX	XX	XX	00	00		
SYSLST	B	B	C	C	H	H	R	00	00	78	B	B	C	C	H	H	R	H	H	XX	XX	XX	00	00		
PRCDIB	B	B	C	C	H	H	R	00	00	50	B	B	C	C	H	H	R	H	H	XX	XX	XX	00	00		

Format of any DIB table if SYSFIL= NO

	0						6	7		9	10							16	17	18	19	20		21	22	23
	Current Address							K	D	D	End Address								UL	LL	M	RC				
SYSLNK	C	C	H	H	00	00	00	P	P	00																
PRCDIB	B	B	C	C	H	H	R	00	00	50	B	B	C	C	H	H	R	H	H	XX	XX	XX	00	00		

■) BG SYSLNK DIB contains the PUB pointer for CLB.
For FG SYSLNK DIB's this byte is unused.

Format of 3540 Diskette DIB table

	0						6	7		9	10							16	17	18	19	20		21	22	23
	Current Address							K	D	D	End Address										M	RC				
SYSIN	00	00	00	00	C	H	R	00	00	00	FLG	EXT #	HDR #	00	C	H	R	00	00	XX	00	00	00	00		
SYSPCH	00	00	00	00	C	H	R	00	00	00	FLG	EXT #	HDR #	00	C	H	R	00	00	XX	00	00	00	00		
SYSLST	00	00	00	00	C	H	R	00	00	00	FLG	EXT #	HDR #	00	C	H	R	00	00	XX	00	00	00	00		

DISK INFORMATION BLOCK (DIB) TABLE (....Cont'd)

Bytes 0 - 6 : Current address
- Bytes 0 - 5 : Current address of key: the next address to be used (both for input and output).
- 6 : Record number of current address.

Bytes 7 - 9 : KDD
Key and data length of the symbolic device.
PP: starting cylinder of private core image library if PCIL is assigned; otherwise zero.

Bytes 10-16: End address
- Bytes 10-15: End address of key: the last address within the limits of the extent.
- 16: Record number of end address.
- (if 3540, byte 10: flag byte
  - X'80' = no more extents
  - X'40' = last volume
  - X'10' = continued volume
- byte 11: extent number
- byte 12: HDR 1 number)

Byte 17 : UL — Upper head limit (3540 not used)

Byte 18 : LL — Lower head limit (3540 not used)

Byte 19 : M — Maximum number of records per track.

Byte 20-21: RC — Record count: residual capacity for beginning of operator notification. Set at system generation time with SYSFIL parameter, or after IPL with SET statement (RCLST and/or RCPCH operands). A warning message is issued by job control after end-of-job step when the minimum number of remaining records has been reached or exceeded during the previous job (not supported for 3540).

Byte 22 : X'40' Indicates RPS support

Byte 23 : Reserved

Note : There is one DIB table for each partition.
Label DSKPOSBG identifies the first byte of the BG DIB table (DSKPOSF1, DSKPOSF2 etc. for the other partitions).

The addresses of the DIB tables are contained in bytes 96 and 97 (X'60' - X'61') of the appropriate partition communication region.

TABLES FOR MICR DTF ADDRESSES AND POINTERS

The table of DTF addresses (PDTABB) contains six 8-byte entries; one for each line of the direct control feature on the system.

PDTABB

Byte	0	1	2	3	4	5	6	7
	AND instruction				Owner ship Flags	DTF address for MICR		
0	NI	PDSTAT+1		X'FE'		Device on line 7		
8	NI	PDSTAT+1		X'FD'		Device on line 6		
16	NI	PDSTAT+1		X'FB'		Device on line 5		
24	NI	PDSTAT+1		X'F7'		Device on line 4		
32	NI	PDSTAT+1		X'EF'		Device on line 3		
40	NI	PDSTAT+1		X'DF'		Device on line 2		

Bytes 0-3 Contain an 'AND'instruction that is executed in main line coding to turn off the external line status after its detection.

PDSTAT+1 will contain one or more of the following interrupt codes:

External interrupt code bit	Interrupt Code (byte X'87')	External interrupt cause
15	nnnnnnn1	External signal 7
14	nnnnnn1n	External signal 6
13	nnnnn1nn	External signal 5
12	nnnn1nnn	External signal 4
11	nnn1nnnn	External signal 3
10	nn1nnnnn	External signal 2

n= other external interrupt conditions

Byte 4 Contains the flag of the partition containing the DTF
Background= 10
Foreground= 20-50, depending on the number of partitions

Bytes 5-7 Contain the address of the DTF table

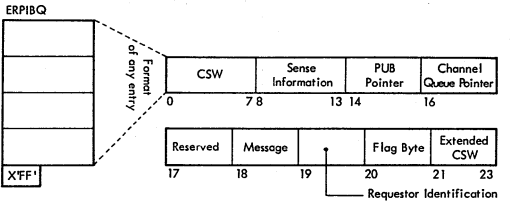
Table of pointers (PDTABA) to DTF addresses with the external interrupt line. The table is set up to handle the status in descending order from bit 15 to bit 10 of the external interrupt code.

PDTABA

Byte	0	1	2	3	4	5	6	7
0	00	08	00	10	00	08	00	18
8	00	08	00	10	00	08	00	20
16	00	08	00	10	00	08	00	18
24	00	08	00	10	00	08	00	28
32	00	08	00	10	00	08	00	18
40	00	08	00	10	00	08	00	20
48	00	08	00	10	00	08	00	18
56	00	08	00	10	00	08	00	

Bytes 126 and 127 (X'7E'-X'7F') of the partitions communication region contain the address of these tables. Label PDTABB identifies the first byte of the first table. The tables are also used for optical reader/sorters.

ERROR RECOVERY PROCEDURE INFORMATION BLOCK (ERPIB)



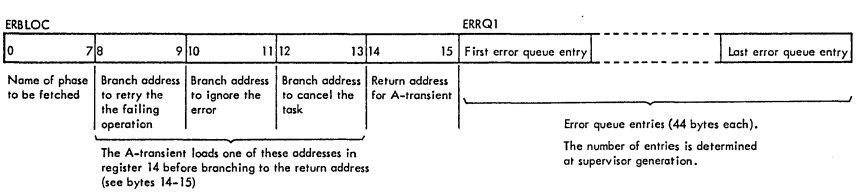
Byte 20: Flag Byte:

Bit 0 : 1= Channel Check on SIO
1-4 : Not used
5 : 1= Sense data stored
6 : 1= DASD ERPIB active
7 : Not used

Note: A free entry is indicated by X'FE' in byte zero, end of queue is indicated by 'FF' in byte zero.

Lable ERPIBQ indentifies the first byte of Queue.

ERROR RECOVERY BLOCK AND ERROR QUEUE ENTRY



Layout of any Error Queue entry

Bytes 0- 7: CSW

Bytes 8- 9: Address of PUB for device in error

Byte 10: Flag byte:

Bit 0: 1= No record found on DASD
1: 1= Intervention required
2: 1= Passback (Set by device ERP)
3: 1= Allow ignore
4: Not used
5: 1= Allow retry
6: Not used
7: 1= Active entry

Byte 11: Message code: may refer to a device error recovery message generated by physical IOCS (See Cancel codes and Messages) or:

Byte 11: (Cont'd) This location may contain one of the following:
X'E2' = The error is recoverable
X'AE' = A record is to be recorded on the system recorder file for SVC44 or a BTAM appendage routine, and a physical transient is to be fetched (last two characters of phase name are in bytes 20-21).

Bytes 12-15: Disk seek address or address of effected PIB (for alternate entry)

Bytes 16-19: Address of CCB

Bytes 20-43: Sense data: The number of sense bytes generated depends on the options specified; the minimum is 24 bytes.
or:
Alternate entry name: If byte 11 contains X'AE', bytes 20-21 contain the last two characters of the phase name of the physical transient to be fetched for SVC44 (A3) or BTAM (A5). X'AF' in byte 22 indicates that the I/O area associated with an alternate entry has been fixed temporarily.

Note: The address of the Error Block can be found in SYSCOM at displacement 0 (X'00').

CANCEL CODES AND MESSAGES

Cancel Code (hex)	Message Code	Descriptive part of Message or Condition	Label
00	----	In all other cases than the ones mentioned hereafter. Default value.	
10	-----	Normal EOJ	ERR10
11	0V07I	No channel program translation for unsupported device	ERR11
12	0V06I	Insufficient buffer space for channel program translation	ERR12
13	0V05I	CCW with count greater than 32 K	ERR13
14	0V04I	Page pool too small	ERR14
15	0V02I	Page fault in disabled program	ERR15
16	0V01I	Page fault in MICR stacker select or PHO routine	ERR16
17	0S02I	Program request (Same as 23 but causes dump because subtasks were attached when maintask issued CANCEL macro)	ERR17
18	-----	Eliminates cancel message when maintask issues DUMP macro with subtasks attached	
19	0P74I	I/O operator option	ERR19
1A	0P73I	I/O error	ERR1A
1B	0P82I	Channel failure	ERR1B
1C	0S14I	CANCEL ALL macro	
1D	0S12I	Main task termination	
1E	0S13I	Unknown ENQ requestor	
1F	0P81I	CPU failure	ERR1F
20	0S03I or 0S11I	Program check	ERR20
21	0S04I or 0S09I	Illegal SVC	ERR21
22	0S05I or 0S06I	Phase not found	ERR22
23	0S02I	Program request	ERR23
24	0S01I	Operator intervention	ERR24
25	0P77I	Invalid address	ERR25
26*	0P71I	SYSxxx not assigned (unassigned LUB code)	ERR26
27	0P70I	Undefined logical unit	ERR27
28	-----	QTAM cancel in progress	
29	0S15I	No relocating loader support (Fetch or load request for relocatable phase while supervisor does not support relocating load)	ERR29

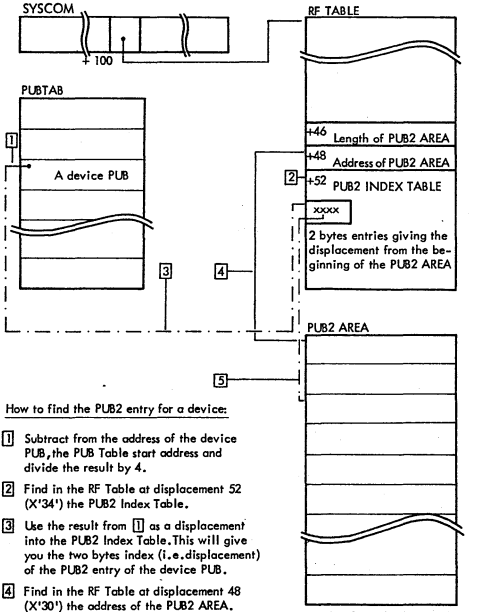
Cancel Code (hex)	Messages Code	Descriptive part of Message or Condition	Label
2A	0V101	I/O error on page data set	
2B	0P841	I/O error during fetch from PCIL	ERR2B
2C	0V091	Illegal parameter passed by PHO routine	ERR2C
2D	0P881	Program cannot be executed/restarted due to failing storage block	ERR2D
2E	0S161	Invalid resource request (possible deadlock)	ERR2E
2F	0V031	More than 255 PFIX requests for 1 page	ERR2F
30	0P721	Reading past/& statement (on SYSRDR or SYSIPT)	ERR30
31	0P751	I/O error queue overflow (error queue overflow)	ERR31
32	0P761	Invalid DASD address	ERR32
33	0P791	No long seek (disk)	ERR33
34		Reserved	
35	0P851	Job control open failure	ERR35
36	0V081	Page fault in I/O appendage routine	ERR36
37		Reserved	
38	0V111	Wrong privately translated CCW	ERR38
39		Reserved	
40	5J951	VTAM termination	
41	5J961	VTAM termination	
FF	0P781	Unrecognized cancel code	
	0P83A**	Supervisor catalog failure	
	0P87A**	IPL failure	

* If the CCB is not available, the logical unit is SYSxxx.

** The cancel code is not significant in case of a supervisor catalog or IPL failure, because the system is placed in the wait state without any further processing by the Terminator.

Note : In addition to recognizing the cancel codes above, the Terminator also recognizes the same codes with the X'80' bit on (cancel occurred in LTA). The X'80' bit is tested by $$BEOJ and subsequently reset.

PUB2 ENTRY ADDRESSING



How to find the PUB2 entry for a device:

1. Subtract from the address of the device PUB, the PUB Table start address and divide the result by 4.
2. Find in the RF Table at displacement 52 (X'34') the PUB2 Index Table.
3. Use the result from 1 as a displacement into the PUB2 Index Table. This will give you the two bytes index (i.e. displacement) of the PUB2 entry of the device PUB.
4. Find in the RF Table at displacement 48 (X'30') the address of the PUB2 AREA.
5. Use the value from 3 as a displacement in the PUB2 AREA.

See PUB2 for the format of the entries for the various device types

Note: PUB2AREA is not generated for model 115, 125 without RMSR support.

PUB 2 TABLE

PUB 2 Table Entry Format for Unit Record and Unsupported Devices

Decimal Displacement	Byte Length	Description
0	3	Usage Count (number of non-ERP SIO's)
3	1	Flag Byte: Bit 0: 1= Device in intensive mode 1: 1= Device in diagnostic mode 2: 1= No recording mode 3: 1= Call statistics transient 2 4: 1= Use PUB 2 name completion field 5: 1= Volume opened on this device 6 , 7: Reserved
4	1	CE mode limit byte
5	1	CE mode byte/bit mask
6	6	Statistical data counters
Total length 12		

PUB 2 Table Entry Format for DASD

Decimal Displacement	Byte Length	Description
0	3	Usage Count (number of non-ERP SIO's)
3	1	Flag byte: Bit 0: 1= Device in intensive mode 1: 1= Device in diagnostic mode 2: 1= No recording mode 3: 1= Call statistics transient 2 4: 1= Use PUB 2 name completion field 5: 1= Volume opened on this device 6 , 7: Reserved
4	1	CE mode limit byte
5	1	CE mode byte/bit mask
6	1	Flag byte: Bit 0: 1= Soft DASD error is queued Bit 1: 1= ERP requests logging of error 2 - 7: Reserved
7	2	Reserved
9	1	Physical module identifier
10	6	Volume serial number
End 3340 Total length 16		
16	8	Statistical data counters
End DASD except 33XX Total length 24		
24	8	Additional statistical data counters (3330 only)
End 3330 and 3350 Total length 32		

PUB 2 TABLE (....Cont'd)

PUB 2 Table Entry Formats for Tapes

Decimal Displacement	Byte Length	Description
0	3	Usage Count (number of non-ERP SIO's)
3	1	Flag Byte: Bit 0: 1= Device in intensive mode 1: 1= Device in diagnostic mode 2: 1= No recording mode 3: 1= Call statistics transient 2 4: 1= Use PUB2 name completion field 5: 1= Volume opened on this device 6-7: Reserved
4	1	CE mode limit byte
5	1	CE mode byte/bit mask
6	2	Name of ERP that wants control
8	1	Flag byte 1: Bit 0: Reserved 1: 1= Unsolicited interrupt 2: 1= ERP is in control 3: 1= ERP requests repositioning 4: 1= Use original TIE byte 0= Use opposite TIE byte 5: 1= Intercept next SIO request 6: 1= ERP read opposite request 7: 1= Restart user's CCW chain
9	1	Flag byte 2: Bit 0: 1= Last ERP operation was ERG 1: 1= Last ERP operation was reposition 2: 1= Cleaner action in progress 3: 1= Read Opposite Recovery in progress 4: 1= Message stored in P2ORGTIE 5: 1= Error on attempt to recover by repositioning 6: 1= Data check after ERP in control 7: Reserved
10	1	Flag byte 3: Bit 0: 1= Failing CCW is Write or Control command 1: 1= User reading backwards 2: 1= Read Opposite Recovery (ROR) 3: 1= Maximum ROR retries 4: 1= Command chaining ROR 5: 1= ROR suppressed incorrect length 6: 1= User used SILI (Suppress Incorrect Length Indicator) 7: Reserved
11	1	Temporary read count
12	1	Temporary write count
13	1	Noise record count
14	2	Erase gap count
16	2	Cleaner action count
18	1	Permanent read errors count
19	1	Permanent write errors count

PUB2 TABLE (....Cont'd)

PUB 2 Table Entry Formats for Tapes

Decimal Displacement	Byte Length	Description
20	1	TIE original direction
21	1	TIE opposite direction
22	1	ERP counter 0
23	1	ERP counter 1
24	8	ERP work area
32	6	Tape serial number
38	2	Block Length
40	4	User ROR command address from CSW
44	2	User ROR residual count from CSW
46	2	Reserved
48	16	Work area for rewind-unload error action
64	10	2400-series statistical data counter area
74	2	Reserved
End 2400-series : Total length 76		
48	20	3410/3420 statistical data counter area
End 3410/3420 : Total length 84		

Note : PUB2 Table is not generated for Model 115 and 125, without RMSR support.

PUB 2 Table Entry Format for 3886 Optical Character Reader

Decimal Displacement	Byte Length	Description
0	3	Usage count (number of non-ERP SIO's)
3	1	Flag byte: Bit 0: 1= Device in intensive mode 1: 1= Device in diagnostic mode 2: 1= No recording mode 3: 1= Call statistics transient 2 4: 1= Use PUB2 name completion field 5: 1= Volume opened on this device 6-7: Reserved
4	1	CE mode limit byte
5	1	CE mode byte/bit mask
6	20	Statistical data counters
End 3886	Total length 26	

PUB2 TABLE (....Cont'd)

PUB 2 Table Entry Format for 3540 Diskette

Decimal Displacement	Byte Length	Description
0	3	Usage Count (number of non-ERP SIO's)
3	1	Flag byte: Bit 0: 1=Device in intensive mode
		1: 1=Device in diagnostic mode
		2: 1=No recording mode
		3: 1=Call statistics transient 2
		4: 1=Use PUB2 name completion field
		5: 1=Volume opened on this device
		6-7: Reserved
4	1	CE mode limit byte
5	1	CE mode byte/bit mask
6	8	Statistical data counters
14	1	Bit 0-5: Reserved
		6: 1=System file opened by Job Control
		7: 1=System file opened by Problem Program
15	1	Reserved
End 3540		Total length 16

RECORDER FILE TABLE (RF TABLE)

Displacement Dec	Displacement Hex	Label	Byte Length	Description
		RFTABLE		Label of Starting Address
0	0	RFFLAGS 1	1	Bit 0: 1= File full 1: 1= RDE option included 2: 1= Initial IPL 3: Reserved 4: 1= File is to be created 5: 1= File has been created 6: Reserved 7: 1= File open and ready
1	1	RFFLAGS 2	1	Bit 0: 1= File full message request 1: 1= Last track message request 2: 1= I/O error message request 3: 1= Data lost message request 4: 1= EVA message request 5: 1= File owned by RTA recorder 6: 1= File owned by PTA recorder 7: 1= File being accessed by EREP
2	2	RFFLAGS 3	1	Bit 0: 1= Last track message issued once 1: 1= Error is to be recorded 2: 1= Short form record request 3: 1= Individual records for unlabeled tapes 4: 1= Reserved 5: 1= Reserved 6: 1= Exit to $$BOMTØ5 indicator for $$BOPEN 7: 1= Exit to $$BOMTØ1 indicator for $$BOPEN
3	3	RFFLAGS 4	1	Work area switches for various transients including BTAM Bit 0: 1 = multiple records required (3211 recording) 1: 1 = PLB record required (3211 recording) 2: 1 = FCB record required (3211 recording) 3: 1 = UCB record required (3211 recording) 4: 1 = Ignore exit requested (3211 recording) 5: Not used 6: Not used 7: 1 = record not written
4	4	RFFLAGS 5	1	Reserved
5	5	RFNOFN	1	N of N records (low order 4 bits contain the number of records to be recorded and high order 4 bits contain the record number being processed
6	6	RFRECTYP	1	Record type code
7	7	RFREL	1	DOS/VS ID-Release Level code

Displacement Dec	Displacement Hex	Label	Byte Length	Description
8	8	RFRDSW 1	1	Record dependent switch 1
9	9	RFRDSW 2	1	Record dependent switch 2
10	A	RFEXIT	2	Exit phase name or exit address
12	C	RFMCONST	2	Multiplier for track balance
14	E	RFDCONST	2	Divisor for track balance
16	10	RFOCONST	2	Overhead track for balance
18	12	RFRECLEN	2	Length of record
20	14	RFTIMEA	4	Address of RMSR time entry
24	18	RFRECADR	4	Address of record
28	1C	RFSEEK	7	Work area for seek address(BBCCHHR)
35	23	RFEREPK	1	Key of EREP partition
36	24	RFHDRCH	4	SYSREC cylinder/head
40	28	RFCHMAP	1	Map of supported channels
41	29	RFCHIDC	3	Channel ID codes
The following entries are not generated for Model 115, 125 without RMSR support				
44	2C	RFEVARTH	1	EVA read treshold
45	2D	RFEVAWTH	1	EVA write treshold
46	2E	RFP2ENTL	2	Length of PUB2 Entry Area
48	30	RFP2ENT	4	Address of PUB2 Entry Area
52	34	RFP2ITAB	*	PUB2 Index Table

* Two bytes are generated for each PUB2 entry in the system.(See PUB2 Entry Addressing for using the PUB2 Index Table to access the PUB2 entries.

Bytes 100-103 (X'64'-'67') of the system communication region (SYSCOM) contain the address of the Recorder File Table.
Label RFTABLE identifies the first byte of the table.

CRT CONSTANT TABLE (CRTTAB)

Name of CRT Routine	Dummy Sense Byte	Address of CRT Save Area	Flag Byte 1	Address of CRT Trans. Area	Flag Byte 2	Address of System Task Save Area	CRT error Information *)
0 7	8	9 11	12	13 15	16	17 19	20 27

*)limited channel logout is saved here

Byte 8 - Sense byte:

Bit 0 : 1= Command reject
Bit 1 : 1= Intervention required (only if Console Printer is attached)
Bit 4 : 1= Equipment check (only if Console Printer is attached)
Bit 7 : 1= Operation check

or: this byte is used for saving Name Indicator

Byte 12 - Flag Byte 1:

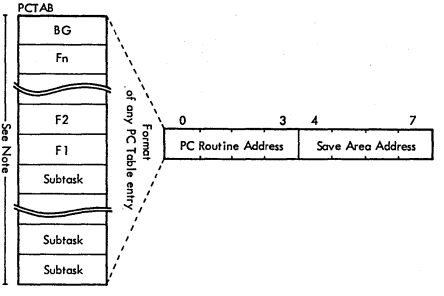
X'80' = ERP message
X'40' = Unit check for CRT
X'20' = CRT Fetch bound
X'10' = Device End simulated
X'08' = Validation error
X'04' = Re-display mode
X'02' = CRT error

Byte 16 - Flag Byte 2:

X'80' = CRT busy
X'40' = Sense byte prepared
X'20' = End of CRT routine
X'10' = Data already read
X'08' = Attention pending
X'04' = Request pending
X'02' = Attention request
X'01' = EOJ on CRT

Bytes 52-55 (X'34'-X'37') of the System Communication Region (SYSCOM) contain the address of the CRT CONSTANT Table.
Label CRTTAB identifies the first byte of the table.

PROGRAM CHECK OPTION TABLE

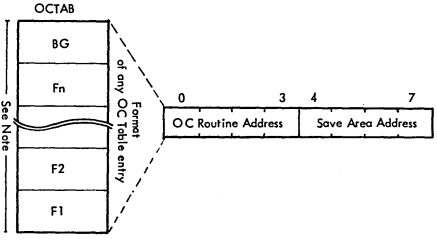


Bytes 0-3 :	No STXIT issued	:	zero
	STXIT issued	:	Address of the user Program Check Routine
	STXIT issued and the user routine is already in use	:	Complement of user Program Check Routine address
Bytes 4-7 :	No STXIT issued	:	zero
	STXIT issued	:	Address of the user Save Area

Note: In a supervisor without multiprogramming support, there is only one entry (BG) in each generated table. With multiprogramming support, there is is one entry for each partition supported.
With asynchronous processing support, each generated table always comprises 15 entries; the subtask entries occupy the higher address locations in the table.

Bytes 100-101 (X'64'-X'65') of the partition communication region contain the address of the PC Option Table. Label PCTAB identifies the frist byte of the table.

OPERATOR COMMUNICATION OPTION TABLE

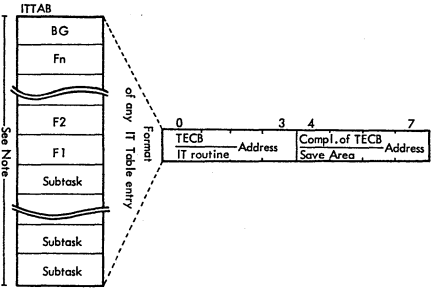


Bytes 0-3 :	No STXIT issued	:	zero
	STXIT issued	:	Address of the user Operator Communication Routine
	STXIT issued and the user routine is already in use	:	Complement of the user Operator Communication Routine address
Bytes 4-7 :	No STXIT issued	:	zero
	STXIT issued	:	Address of the user Save Area

Note: In a supervisor without multiprogramming support, there is only one entry (BG) in each generated table.
With multiprogramming support, there is one entry for each partition supported.

Bytes 104-105 (X'68'-X'69') of the partition communication region contain the address of the OC Option Table. Label OCTAB identifies the first byte of the table.

INTERVAL TIMER OPTION TABLE



Bytes 0-3 :

No TECB or STXIT issued	:	zero
TECB issued	:	Address of the timer event control block
STXIT issued	:	Address of user interval timer routine
STXIT issued and user routine is already in use	:	Complement of the user interval timer routine

Bytes 4-7 :

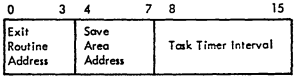
No TECB or STXIT issued	:	zero
TECB issued	:	Complement of the TECB address
STXIT issued	:	Address of the user save area

Note: One table entry is built for each partition supported and an IT Request table is also built.

With timer and asynchronous processing supported, the table always comprises 15 entries ; the subtask entries occupy the higher address locations in the table.

Bytes 102-103 (X'66'-X'67') of the partition communication region contain the address of the IT Option Table. Label ITTAB identifies the first byte of the table.

TASK TIMER OPTION TABLE



Bytes 0 - 3 : No STXIT issued : zero

STXIT issued: Address of entry point of user's task timer routine address.

STXIT issued and user's routine already in use: Complement of the task timer routine address

Bytes 4 - 7 : No STXIT issued : zero.

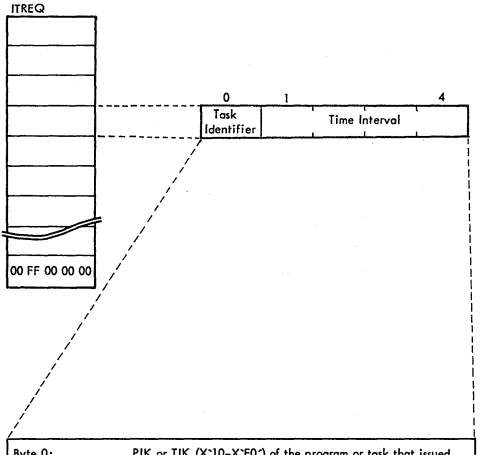
STXIT issued : Address of the user's save area.

Bytes 8 - 15 : No SETT issued : zero or negative.

SETT issued : Time remaining of the interval specified. The time is expressed in micro-seconds in bits 4 - 51. Bits 52-63 are ignored.

Note : Bytes 176 - 179 (X"B0-B3) of the System Communications region (SYSCOM) contain the address of Task Timer Table. Lable TTTAB identifies the first byte of the table.

INTERVAL TIMER REQUEST TABLE

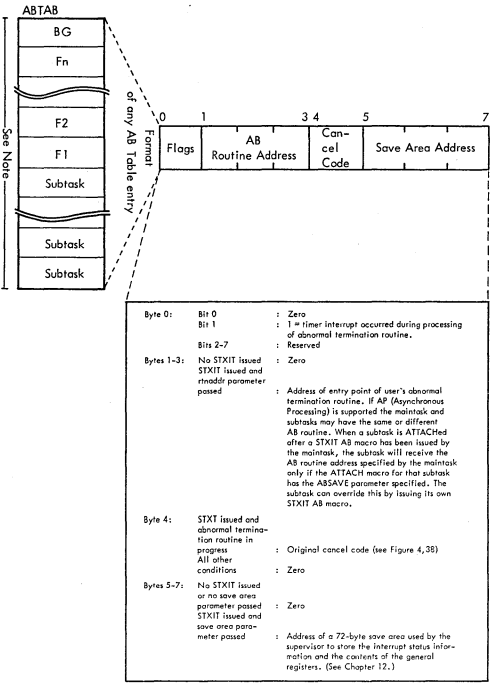


Byte 0:	PIK or TIK (X'10'-X'F0') of the program or task that issued the SETIME macro. If the entry is not used or is the last entry, byte 0 contains X'00'.
Bytes 1 - 4:	Time interval that will elapse between the occurrence of the currently pending interrupt in SYSTIMER and the occurence of the interrupt itself. The entries are in ascending order of magnitude, that is, the smallest interval appears first. If the entry is not used or is the last entry, bytes 1 - 4 contain the highest interval possible (X'FF000000').
	The interval corresponding to the currently pending interrupt is in SYSTIMER (address X'50') and the key of the respective program or task is in TWTIMS.

Note: This table is only used when timer interrupt occurs while an interrupt from another partition of task is pending. See SVC 24 in DOS/VS Supervisor.

Bytes 80-83 (X'50' - X'53') of the System communication region (SYSCOM) contain the address of the IT Request Table. Label ITREQ identifies the first byte of the table.

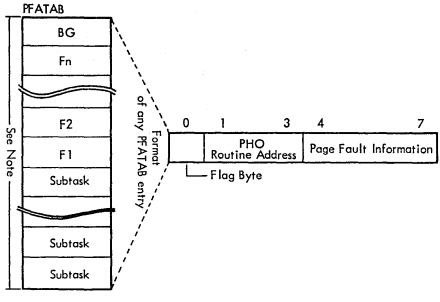
ABNORMAL TERMINATION OPTION TABLE



Note: One table entry is generated for each partition supported. With asynchronous processing support, the table always comprises 15 entries; the subtask entries occupy the higher address locations in the table.

Bytes 84-87 (X'54'-X'57') of the System Communication region (SYSCOM) contain the address of the AB Option Table. Label ABTAB identifies the first byte of the table.

PAGE FAULT HANDLING OVERLAP OPTION TABLE



SETPFA issued

Byte 0 : Bit 0 : 1= A page fault from the task is in the Page Queue Table

Bit 1 : 1= A page fault occuring in a supervisor service of the task is queued in the Page Queue Table

Bits 2-7: Unused

Bytes 1-3 : Address of the user's page fault appendage routine

Bytes 4-7 : Page fault information (for detailed layout see Page Queue Table)

This information is stored here only when a page fault occurs in a supervisor service working for the task, while another page fault is queued for the task in the Page Queue Table, otherwise zero

NO SETPFA issued

Bytes 0-7 : Zero

Note: PFATAB is only built if PHO=YES was specified in the SUPVR macro at supervisor generation.
One table entry is generated for each partition supported. With asynchronous processing support, the table always comprises 15 entries; the subtask entries occupy the higher address locations in the table.

Label PFATAB identifies the first byte of the table.

JOB ACCOUNTING INTERFACE COMMON TABLE

Displacement	Label	Description
	(ACCTCOMN)	
0 -15	ACCTSVRG	Temporary register save area
16-17	ACCTSVRX	Save area for remainder of overhead counter times distributed by partition on exit
18-19	ACCTSVRE	Save area for remainder of all-bound counter times distributed by partitions on entry
20-23	ACCTPCNT	Count of partitions using the Job Accounting interface
24	ACCTSAID	Owner of physical transient area *)
25	ACCTFAID	Interrupted program *)
26	ACCTRAID	Active program *)
27	ACCTSWCH	Accounting switches: if bit= 1,true ; if bit= 0, not true bit 0: cancel accounting bit 4: IPL indicator bit 1: no active partitions bit 5: not used bit 2: catalog in process bit 6: not used bit 3: alternate label area bit 7: not used
28-31	ACCTIME	Start time of current accounting interval, in complement format
32-33	ACCTRESC	Reserved
34-35	ACCTUSEP	Address of user save area (ACCTUSER)
36-37	ACCTUSEL	Length of user save area (Set with 1st operand of FOPT macro parameter JALIOCS)
38-39	ACCT$JOB	Job accounting partition indication
40-43	ACCTBLES	Address of BG Job Accounting Table

If multiprogramming is supported, this table is to be extended (depending on the number of supported partitions) according to the pattern shown below otherwise the table ends here.

NPARTS=n

44-47		Address of Fn Job Accounting Table . . . .
+4		Address of F2 Job Accounting Table
+4		Address of F1 Job Accounting Table
+ (n*2)-1	ACCTSEAS	Control Field: prevents the accounting routine being loaded twice

*) These values are the same as the PIK values for the relevant tasks

Byte 124-127(X'7C'-'7F') of the system communication region (SYSCOM) contain the address of the Job Accounting Interface Common Table.
Label ACCTCOMN identifies the first byte of the table.

JOB ACCOUNTING INTERFACE PARTITION TABLE

Displacement	Label	Description
	(ACCTABLE)	
0 -3	ACCTWK1	Work area used in SIO update
4 -7	ACCTWK2	Work area used with ACCTWK1 in start/stop time routine
8 -11	ACCTSVPT	Job card pointer; address of job card field following jobname
12	ACCTPART	ID of partition in charge (partition switch name)
13	ACCTRES2	Reserved
14-15	ACCTLEN	Length of SIO area= 6n+1, where n= number of devices for this partition in SYSGEN option JA= (n,n,...)
16-21	ACCTLOAD	Label area instruction; moves JAI label area address to OPEN/CLOSE transients
22-23	ACCTRES3	Reserved
24-27	ACCTLADD	Address of alternate label area
28-31	ACCTCPUT	Counter for CPU time elapsed in a jobstep, counted in 300th of a second
32-35	ACCTOVHT	Counter for overhead time; time not charged to any partition
36-39	ACCTBNDT	Counter for all-bound time; system wait state time divided between running partitions
40-47	ACCTSVJN	Save area for job name during simulated EOJ
48-55	ACCTJBNM	Job name; taken from job card
56-71	ACCTUSRS	User information; 16 bytes from job card
72-73	ACCTPTID	Partition ID: 'BG', 'F1', 'F2', etc. in EBCDIC format
74	ACCTCNCL	Cancel code; see Cancel Codes and Messages
75	ACCTYPER	Type of record: 'S'= job step, 'L'= last step of job
76-83	ACCTDATE	Date in format specified at SYSGEN (MM/DD/YY or DD/MM/YY)
84-87	ACCTSTRT	Start time of a job-step, in packed decimal (DHHMMSSF ; F = sign)
88-91	ACCTSTOP	Stop time of a job-step in the same format as ACCTSTRT. This value is used as the start time for the next step
92-95	ACCTRES	Reserved
96-103	ACCTEXEC	Phase name; taken from execute card

Job Accounting Table (user's portion of Partition Table)

Job Accounting Table (user's portion of partition table)

Displacement	Label	Description
104-107	ACCTHICR	Real mode : the problem program end address reflecting ALLOCR or the SIZE = parameter on the EXEC statement Virtual mode : the virtual start address of the partition incremented by the amount of virtual storage referenced during this step
108-111	ACCTIMES	CPU time elapsed in a job-step; counted in 300th of a second
112-115	--------	Overhead time; elapsed time not charged to any partition, in 300th of a second
116-119	--------	All-bound time; system wait state time divided between running partitions, in 300th of a second
120	ACCTSIOS	SIO tables: 6bytes for each device specified by SYSGEN options, as follows: 2 bytes for device address (0cuu), 4 bytes for count of SIO's in current job-step
--------	--------	Overflow byte: normally X'20', but is X'30' if more devices are used within a partition than specified by SYSGEN options

Notes: DSECT ACCTABLE symbolically addresses the JAI Partition Tables with labels as shown. Each partition in which JAI is supported has its own JAI Partition Table. This table is labeled ACCTBG for the active partition BG; ACCTF1 for the active partition F1; ACCTF2 for F2, etc.

The address of this table is in the partition Communication Region at displacement 116 (X'74')

ASCII EBCDIC TRANSLATION TABLES

ASCII/EBCDIC (0/0 to 3/2)

ASCII					EBCDIC				
Charact.	Col	Row	Bit Pattern		Col Row (in hex)		Bit Pattern		Comments
NUL	0	0	0000	0000	0	0	0000	0000	
SOH	0	1	0000	0001	0	1	0000	0001	
STX	0	2	0000	0010	0	2	0000	0010	
ETX	0	3	0000	0011	0	3	0000	0011	
EOT	0	4	0000	0100	3	7	0011	0111	
ENQ	0	5	0000	0101	2	D	0010	1101	
ACK	0	6	0000	0110	2	E	0010	1110	
BEL	0	7	0000	0111	2	F	0010	1111	
BS	0	8	0000	1000	1	6	0001	0110	
HT	0	9	0000	1001	0	5	0000	0101	
LF	0	10	0000	1010	2	5	0010	0101	
VT	0	11	0000	1011	0	B	0000	1011	
FF	0	12	0000	1100	0	C	0000	1100	
CR	0	13	0000	1101	0	D	0000	1101	
SO	0	14	0000	1110	0	E	0000	1110	
SI	0	15	0000	1111	0	F	0000	1111	
DLE	1	0	0001	0000	1	0	0001	0000	
DC1	1	1	0001	0001	1	1	0001	0001	
DC2	1	2	0001	0010	1	2	0001	0010	
DC3	1	3	0001	0011	1	3	0001	0011	
DC4	1	4	0001	0100	3	C	0011	1100	
NAK	1	5	0001	0101	3	D	0011	1101	
SYN	1	6	0001	0110	3	2	0011	0010	
ETB	1	7	0001	0111	2	6	0010	0110	
CAN	1	8	0001	1000	1	8	0001	1000	
EM	1	9	0001	1001	1	9	0001	1001	
SUB	1	10	0001	1010	3	F	0011	1111	
ESC	1	11	0001	1011	2	7	0010	0111	
FS	1	12	0001	1100	1	C	0001	1100	
GS	1	13	0001	1101	1	D	0001	1101	
RS	1	14	0001	1110	1	E	0001	1110	
US	1	15	0001	1111	1	F	0001	1111	
SP	2	0	0010	0000	4	0	0100	0000	
!	2	1	0010	0001	4	F	0100	1111	Logical OR
"	2	2	0010	0010	7	F	0111	1111	
#	2	3	0010	0011	7	B	0111	1011	
$	2	4	0010	0100	5	B	0101	1011	
%	2	5	0010	0101	6	C	0110	1100	
&	2	6	0010	0110	5	0	0101	0000	
'	2	7	0010	0111	7	D	0111	1101	
(	2	8	0010	1000	4	D	0100	1101	
)	2	9	0010	1001	5	D	0101	1101	
*	2	10	0010	1010	5	C	0101	1100	
+	2	11	0010	1011	4	E	0100	1110	
,	2	12	0010	1100	6	B	0110	1011	
-	2	13	0010	1101	6	0	0110	0000	Hyphen, Minus
.	2	14	0010	1110	4	B	0100	1011	
/	2	15	0010	1111	6	1	0110	0001	
0	3	0	0011	0000	F	0	1111	0000	
1	3	1	0011	0001	F	1	1111	0001	
2	3	2	0011	0010	F	2	1111	0010	

ASCII					EBCDIC				
Charact.	Col	Row	Bit Pattern		Col (in hex)	Row	Bit Pattern		Comments
3	3	3	0011	0011	F	3	1111	0011	
4	3	4	0011	0100	F	4	1111	0100	
5	3	5	0011	0101	F	5	1111	0101	
6	3	6	0011	0110	F	6	1111	0110	
7	3	7	0011	0111	F	7	1111	0111	
8	3	8	0011	1000	F	8	1111	1000	
9	3	9	0011	1001	F	9	1111	1001	
:	3	10	0011	1010	7	A	0111	1010	
;	3	11	0011	1011	5	E	0101	1110	
<	3	12	0011	1101	4	C	0100	1100	
=	3	13	0011	1101	7	E	0111	1110	
>	3	14	0011	1110	6	E	0110	1110	
?	3	15	0011	1111	6	F	0110	1111	
@	4	0	0100	0000	7	C	0111	1100	
A	4	1	0100	0001	C	1	1100	0001	
B	4	2	0100	0010	C	2	1100	0010	
C	4	3	0100	0011	C	3	1100	0011	
D	4	4	0100	0100	C	4	1100	0100	
E	4	5	0100	0101	C	5	1100	0101	
F	4	6	0100	0110	C	6	1100	0110	
G	4	7	0100	0111	C	7	1100	0111	
H	4	8	0100	1000	C	8	1100	1000	
I	4	9	0100	1001	C	9	1100	1001	
J	4	10	0100	1010	D	1	1101	0001	
K	4	11	0100	1011	D	2	1101	0010	
L	4	12	0100	1100	D	3	1101	0011	
M	4	13	0100	1101	D	4	1101	0100	
N	4	14	0100	1110	D	5	1101	0101	
O	4	15	0100	1111	D	6	1101	0110	
P	5	0	0101	0000	D	7	1101	0111	
Q	5	1	0101	0001	D	8	1101	1000	
R	5	2	0101	0010	D	9	1101	1001	
S	5	3	0101	0011	E	2	1110	0010	
T	5	4	0101	0100	E	3	1110	0011	
U	5	5	0101	0101	E	4	1110	0100	
V	5	6	0101	0110	E	5	1110	0101	
W	5	7	0101	0111	E	6	1110	0110	
X	5	8	0101	1000	E	7	1110	0111	
Y	5	9	0101	1001	E	8	1110	1000	
Z	5	10	0101	1010	E	9	1110	1001	
[	5	11	0101	1011	4	A	0100	1010	
\	5	12	0101	1100	E	0	1110	0000	Reverse Slant
]	5	13	0101	11'1	5	A	0101	1010	
¬	5	14	0101	1110	5	F	0101	1111	Logical Not
—	5	15	0101	1111	6	D	0110	1101	Underscore
`	6	0	0110	0000	7	9	0111	1001	Grave accent
a	6	1	0110	0001	8	1	1000	0001	
b	6	2	0110	0010	8	2	1000	0010	
c	6	3	0110	0011	8	3	1000	0011	
d	6	4	0110	0100	8	4	1000	0100	
e	6	5	0110	0101	8	5	1000	0101	
f	6	6	0110	0110	8	6	1000	0110	

ASCII					EBCDIC				
Charact.	Col Row		Bit Pattern		Col Row		Bit Pattern		Comments
g	6	7	0110	0111	8	7	1000	0111	
h	6	8	0110	1000	8	8	1000	1000	
i	6	9	0110	1001	8	9	1000	1001	
j	6	10	0110	1010	9	1	1001	0001	
k	6	11	0110	1011	9	2	1001	0010	
l	6	12	0110	1100	9	3	1001	0011	
m	6	13	0110	1101	9	4	1001	0100	
n	6	14	0110	1110	9	5	1001	0101	
o	6	15	0110	1111	9	6	1001	0110	
p	7	0	0111	0000	9	7	1001	0111	
q	7	1	0111	0001	9	8	1001	1000	
r	7	2	0111	0010	9	9	1001	1001	
s	7	3	0111	0011	A	2	1010	0010	
t	7	4	0111	0100	A	3	1010	0011	
u	7	5	0111	0101	A	4	1010	0100	
v	7	6	0111	0110	A	5	1010	0101	
w	7	7	0111	0111	A	6	1010	0110	
x	7	8	0111	1000	A	7	1010	0111	
y	7	9	0111	1001	A	8	1010	1000	
z	7	10	0111	1010	A	9	1010	1001	
{	7	11	0111	1011	C	0	1100	0000	
|	7	12	0111	1100	6	A	0110	1010	Vertical Line
}	7	13	0111	1101	D	0	1101	0000	
~	7	14	0111	1110	A	1	1010	0001	Tilde
DEL	7	15	0111	1111	0	7	0000	0111	

EBCDIC					ASCII				
Charact	Col Row (in hex)		Bit Pattern		Col Row		Bit Pattern		Comments
NUL	0	0	0000	0000	0	0	0000	0000	
SOH	0	1	0000	0001	0	1	0000	0001	
STX	0	2	0000	0010	0	2	0000	0010	
ETX	0	3	0000	0011	0	3	0000	0011	
HT	0	5	0000	0101	0	9	0000	1001	
DEL	0	7	0000	0111	7	15	0111	1111	
VT	0	B	0000	1011	0	11	0000	1011	
FF	0	C	0000	1100	0	12	0000	1100	
CR	0	D	0000	1101	0	13	0000	1101	
SO	0	E	0000	1110	0	14	0000	1110	
SI	0	F	0000	1111	0	15	0000	1111	
DLE	1	0	0001	0000	1	0	0001	0000	
DC1	1	1	0001	0001	1	1	0001	0001	
DC2	1	2	0001	0010	1	2	0001	0010	
DC3	1	3	0001	0011	1	3	0001	0011	
BS	1	6	0001	0110	0	8	0000	1000	
CAN	1	8	0001	1000	1	8	0001	1000	
EM	1	9	0001	1001	1	9	0001	1001	
FS	1	C	0001	1100	1	12	0001	1100	
GS	1	D	0001	1101	1	13	0001	1101	
RS	1	E	0001	1110	1	14	0001	1110	
US	1	F	0001	1111	1	15	0001	1111	

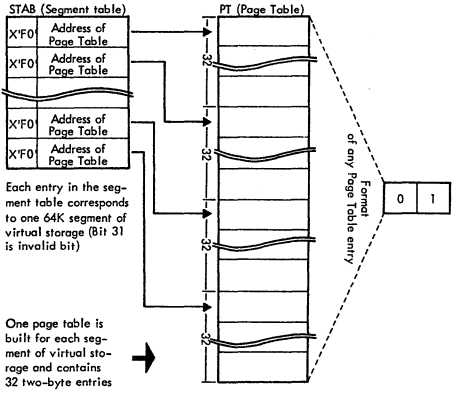
EBCDIC to ASCII (X'00' to 'IF')

ASCII EBCDIC TRANSLATION TABLES(....Cont'd) EBCDIC to ASCII (X'25 to X'93')

EBCDIC					ASCII				
Charact.	Col (in hex)	Row	Bit Pattern		Col	Row	Bit Pattern		Comments
LF	2	5	0010	0101	0	10	0000	1010	
ETB	2	6	0010	0110	1	7	0001	0111	
ESC	2	7	0010	0111	1	11	0001	1011	
ENQ	2	D	0010	1101	0	5	0000	0101	
ACK	2	E	0010	1110	0	6	0000	0110	
BEL	2	F	0010	1111	0	7	0000	0111	
SYN	3	2	0011	0010	1	6	0001	0110	
EOT	3	7	0011	0111	0	4	0000	0100	
DC4	3	C	0011	1100	1	4	0001	0100	
NAK	3	D	0011	1101	1	5	0001	0101	
SUB	3	F	0011	1111	1	10	0001	1010	
SP	4	0	0100	0000	2	0	0010	0000	
[	4	A	0100	1010	5	11	0101	1011	
.	4	B	0100	1011	2	14	0010	1110	
<	4	C	0100	1100	3	12	0011	1100	
(	4	D	0100	1101	2	8	0010	1000	
+	4	E	0100	1110	2	11	0010	1011	
!	4	F	0100	1111	2	1	0010	0001	Logical OR
&	5	0	0101	0000	2	6	0010	0110	
]	5	A	0101	1010	5	13	0101	1101	
$	5	B	0101	1011	2	4	0010	0100	
*	5	C	0101	1100	2	10	0010	1010	
)	5	D	0101	1101	2	9	0010	1001	
;	5	E	0101	1110	3	11	0011	1011	
¬	5	F	0101	1111	5	14	0101	1110	Logical Not
-	6	0	0110	0000	2	13	0010	1101	Hyphen, Minus
/	6	1	0110	0001	2	15	0010	1111	
¦	6	A	0110	1010	7	12	0111	1100	Vertical Line
,	6	B	0110	1011	2	12	0010	1100	
%	6	C	0110	1100	2	5	0010	0101	.
_	6	D	0110	1101	5	15	0101	1111	Underscore
>	6	E	0110	1110	3	14	0011	1110	
?	6	F	0110	1111	3	15	0011	1111	
`	7	9	0111	1001	6	0	0110	0000	Grave accent
:	7	A	0111	1010	3	10	0011	1010	
#	7	B	0111	1011	2	3	0010	0011	
@	7	C	0111	1100	4	0	0100	0000	
'	7	D	0111	1101	2	7	0010	0111	
=	7	E	0111	1110	3	13	0011	1101	
"	7	F	0111	1111	2	2	0010	0010	
a	8	1	1000	0001	6	1	0110	0001	
b	8	2	1000	0010	6	2	0110	0010	
c	8	3	1000	0011	6	3	0110	0011	
d	8	4	1000	0100	6	4	0110	0100	
e	8	5	1000	0101	6	5	0110	0101	
f	8	6	1000	0110	6	6	0110	0110	
g	8	7	1000	0111	6	7	0110	0111	
h	8	8	1000	1000	6	8	0110	1000	
i	8	9	1000	1001	6	9	0110	1001	
j	9	1	1001	0001	6	10	0110	1010	
k	9	2	1001	0010	6	11	0110	1011	
l	9	3	1001	0011	6	12	0110	1100	

EBCDIC					ASCII				
Charact.	Col (in hex)	Row	Bit Pattern		Col	Row	Bit Pattern		Comments
m	9	4	1001	0100	6	13	0110	1101	
n	9	5	1001	0101	6	14	0110	1110	
o	9	6	1001	0110	6	15	0110	1111	
p	9	7	1001	0111	7	0	0111	0000	
q	9	8	1001	1000	7	1	0111	0001	
r	9	9	1001	1001	7	2	0111	0010	
~	A	1	1010	0001	7	14	0111	1110	Tilde
s	A	2	1010	0010	7	3	0111	0011	
t	A	3	1010	0011	7	4	0111	0100	
u	A	4	1010	0100	7	5	0111	0101	
v	A	5	1010	0101	7	6	0111	0110	
w	A	6	1010	0110	7	7	0111	0111	
x	A	7	1010	0111	7	8	0111	1000	
y	A	8	1010	1000	7	9	0111	1001	
z	A	9	1010	1001	7	10	0111	1010	
{	C	0	1100	0000	7	11	0111	1011	
A	C	1	1100	0001	4	1	0100	0001	
B	C	2	1100	0010	4	2	0100	0010	
C	C	3	1100	0011	4	3	0100	0011	
D	C	4	1100	0100	4	4	0100	0100	
E	C	5	1100	0101	4	5	0100	0101	
F	C	6	1100	0110	4	6	0100	0110	
G	C	7	1100	0111	4	7	0100	0111	
H	C	8	1100	1000	4	8	0100	1000	
I	C	9	1100	1001	4	9	0100	1001	
}	D	0	1101	0000	7	13	0111	1101	
J	D	1	1101	0001	4	10	0100	1010	
K	D	2	1101	0010	4	11	0100	1011	
L	D	3	1101	0011	4	12	0100	1100	
M	D	4	1101	0100	4	13	0100	1101	
N	D	5	1101	0101	4	14	0100	1110	
O	D	6	1101	0110	4	15	0100	1111	
P	D	7	1101	0111	5	0	0101	0000	
Q	D	8	1101	1000	5	1	0101	0001	
R	D	9	1101	1001	5	2	0101	0010	
\	E	0	1110	0000	5	12	0101	1100	Reverse Slant
S	E	2	1110	0010	5	3	0101	0011	
T	E	3	1110	0011	5	4	0101	0100	
U	E	4	1110	0100	5	5	0101	0101	
V	E	5	1110	0101	5	6	0101	0110	
W	E	6	1110	0110	5	7	0101	0111	
X	E	7	1110	0111	5	8	0101	1000	
Y	E	8	1110	1000	5	9	0101	1001	
Z	E	9	1110	1001	5	10	0101	1010	
0	F	0	1111	0000	3	0	0011	0000	
1	F	1	1111	0001	3	1	0011	0001	
2	F	2	1111	0010	3	2	0011	0010	
3	F	3	1111	0011	3	3	0011	0011	
4	F	4	1111	0100	3	4	0011	0100	
5	F	5	1111	0101	3	5	0011	0101	
6	F	6	1111	0110	3	6	0011	0110	
7	F	7	1111	0111	3	7	0011	0111	
8	F	8	1111	1000	3	8	0011	1000	
9	F	9	1111	1001	3	9	0011	1001	

SEGMENT AND PAGE TABLE



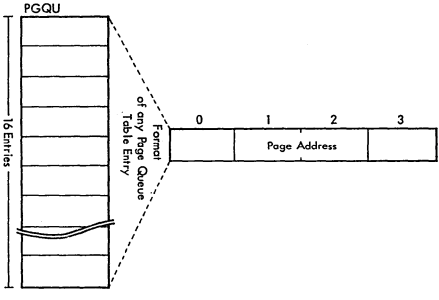
Bytes 0 and 1 :

Bit 0	:	1= Address of any byte in the corresponding page is invalid
Bits 0-12	:	Leftmost 13 bits of address of page in real storage if bit 13= 0
Bits 8-11	:	Storage key of corresponding partition if bit 13= 1
Bit 13	:	0= Page is in real storage 1= Page is not in real storage
Bit 14	:	Always zero
Bit 15	:	Used bit: 0= Page must be read from page data set (a valid copy of the page is on the page data set) 1= Page need not be read from page data set (no valid copy of the page on the page data set)

Note: Bits 4-7 may contain PIK/16 of partition which requires PFIXing of the page currently located in corresponding page frame. The page must be PFIXed in a reserved page frame in the correct real partition.

Bytes 208-211 (X'D0'-X'D3') of the System Communication Region (SYSCOM) contain the address of the Segment Table.
Label STAB identifies the first byte of the Segment Table.
Label PT identifies the first byte of the Page Table.

PAGE QUEUE TABLE



Byte 0 : Bits 0-3 : Bits 0-3 of PIK or TIK (if AP supported) of user task

Bits 4-7 : B'0000' Page Translation Exception
B'1000' Dummy Request
B'0100' PFIX Request
B'0010' TFIX Request
B'0001' GETREAL Request

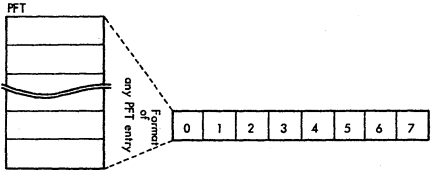
Bytes 1-2 : Leftmost 16 bits of the address of the page (which is also the address of the page frame, if GETREAL) to be handled; the remaining 8 bits of the address are assumed to be zero

Byte 3 : Task identifier (PIK or TIK) of user task or, task identifier (select byte in STID field) of system task

Note: Label PGQU identifies the first byte of the table

Bytes 28-31 (X'1C'-X'1F') of the RASLINK area contain the address of the Page Queue Table

PAGE FRAME TABLE



Bytes 0 and 1 :

Bits 0-10:	Counter for temporary fixes (tfix counter)
Bit 11 :	Reserved for later use
Bit 12 :	1= PFIX request (into another page frame) pending for the page (NFF bit)
Bit 13 :	If on, page frame is unusable (hardware failure, DRAP bit)
Bit 14 :	If 0, page frame belongs to selection pool (SP bit)
Bit 15 :	1= PFIX (different page) or GETREAL request pending for the page frame (NF bit)
Bytes 2 and 3:	Pointer to next page frame in queue or to queue header if last element in the queue (all page frames in selection pool are queued)
Bytes 4 and 5:	If the frame is occupied: the page number (the virtual address of the page divided by 2048); if unused X'FFFF'
Bytes 6 and 7:	Pointer to the previous frame in the queue or to queue header if first element in the queue (all page frames in the selection pool are queued)

Note: The page frame table is built at supervisor generation time and contains one 8-byte entry for each 2K block of real storage (page frame) as specified in the RSIZE parameter of the VSTAB macro.

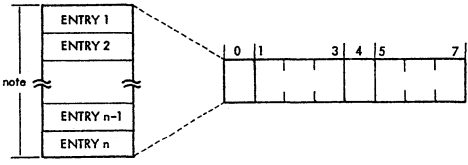
Bytes 212-215 (X'D4'-X'D7') of the System Communication Region (SYSCOM) contain the address of the Page Frame Table. Label PFT identifies the first byte of the table.

PAGE FRAME TABLE EXTENSION

For each Page Frame Table entry in the system there is a 1-byte entry in the Page Frame Table Extension. Each entry is a counter for the number of times a page has been permanently fixed in the page frame (PFIX counter)

Bytes 216-219 (X'D8'-X'DB') of the System Communication Region (SYSCOM) contain the address of the Page Frame Table Extension. Label PFTX identifies the first byte of the table.

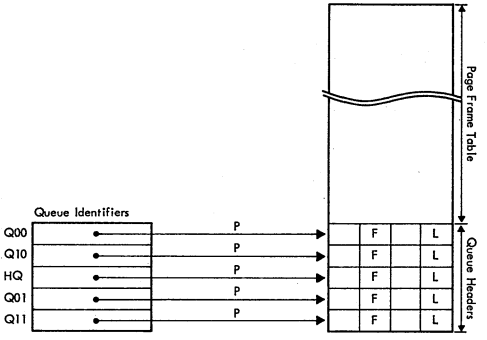
PAGETAB



Byte 0 :	TIK (identifier of task that issued the PAGEIN macro).
Bytes 1-3:	Pointer to a list of areas that are to be paged in.
Byte 4 :	bit 0 : 1 = PAGEIN request has been completed bit 1 : reserved bit 2 : 1 = one or more of the requested pages are outside the requesting program's partition bit 3 : 1 = one or more negative length specifications were detected bit 4-7: reserved
Byte 5-7 :	Pointer to ECB (if used) or zero.

Note : The number of entries is determined at supervisor generation time. Label PAGETAB identifies the first byte of the table.

SELECTION POOL IDENTIFIERS AND QUEUE HEADERS



F : Pointer to first PFT entry in queue

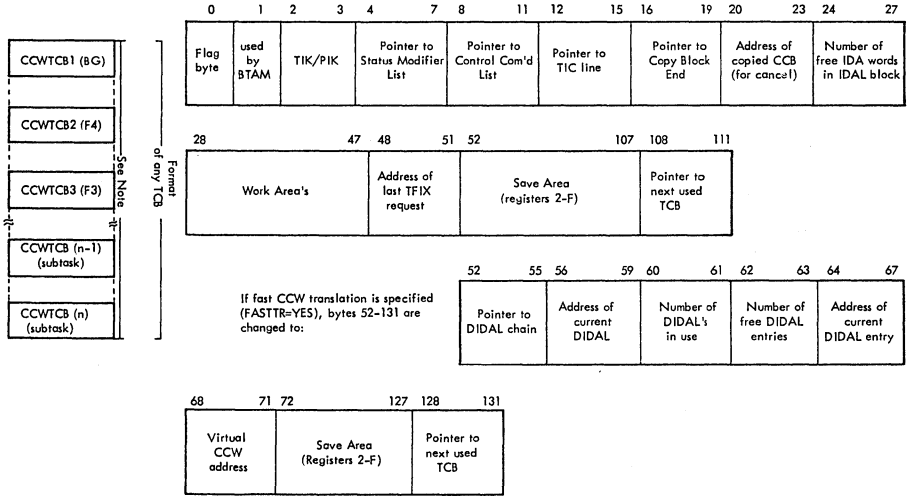
L : Pointer to last PFT entry in queue

P : Initially these pointers are as drawn. They may be changed by queue switching.

Note:

The Queue Headers are located directly behind the Page Frame Table.
Each Queue Header is 8 bytes long.
Bytes 2 and 3 contain a pointer to the first queue entry and bytes 6 and 7 contain a pointer to the last queue entry.

TRANSLATION CONTROL BLOCK (TCB)



TRANSLATION CONTROL BLOCK (TCB) (....Cont'd)

Byte 0: bit 0 = 1 : data chaining specified
1 = 1 : Read/Sense command specified
2 = 1 : Read backward command specified
3 = 1 : Status modifier command with data chaining
4 = 1 : Status modifier command only
5-6 : Reserved
7 = 1 : REPLICA creation flag (fast CCW translation)

Byte 1: Number of copy blocks needed in addition to those required for current CCW-translation request.

Bytes 4-7: Pointer to Status Modifier List belonging to handled device. Zero if device does not support Status Modifier commands.

Bytes 8-11: Pointer to Control Command List belonging to handled device. Zero if device does not support Control Commands with data area.

Bytes 12-15: Chain of knots of tree structure caused by TIC following Status Modifier command. Zero if no knots exist in CCW chain.

Bytes 16-19: Chain of knots build because Status Modifier command is last one fitting in CCW copy block. Zero if no Status Modifier commands at end of CCW copy blocks.

Note: One TCB is generated for each partition supported. With asynchronous processing support 15 TCB's are generated.

To locate the TCB (associated with the partition/task), add X'50' to the address of the System Save Area (displacement X'09' of the appropriate PIB). Labels CCWTCB 1- CCWTCBn identify the first byte of the appropriate TCB.

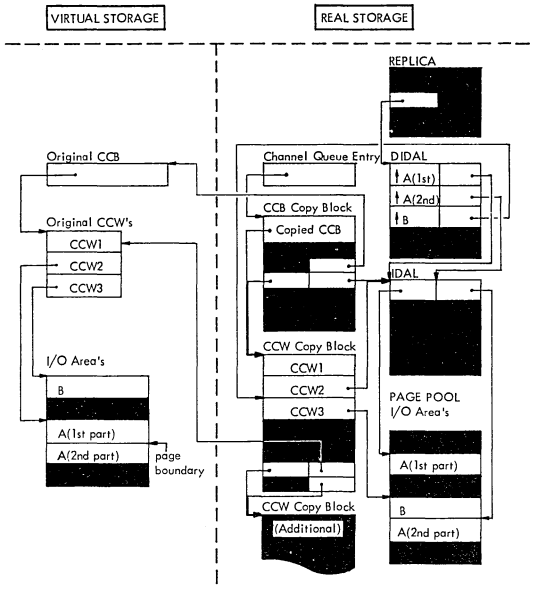
COPY BLOCKS

The copy blocks are located at the end of the supervisor nucleus. Each block is 72 bytes long and is used to store one of the following:

- One CCB plus some additional information (CCB copy block), or
- Up to seven copied CCW's and pointers (CCW copy block), or
- One or more Indirect Data Address Lists (IDAL block).

Chains are formed for both the occupied and the free copy blocks. The occupied blocks are chained by means of pointers in the CCB copy blocks. The address of the first CCB copy block in the chain is contained in the field ACCBB, which is located at the beginning of the channel program translation routine (CCWTADR).
The address of CCWTADR is contained in the System Communication Region (SYSCOM), bytes 132-135 (X'84'-X'87').

The free copy blocks are chained by means of pointers in the copy blocks. The address of the first copy block in the chain is contained in the first AFCB.



Note: The DIDAL and REPLICA blocks are created by the fast CCW translation routine.

Layout of CCB Copy Block

	0	1	2	3	4	5	6	7
0	CCBCNT		CCB COM1	CCB COM2	CCB STA1	CCB STA2	CCB CLS *	CCB LNO
8	CCBCCW Address of first CCW				CCBBY3	CCBCSWW		
16	CCBSENS Sense CCW if any							
24	CCBPIK User PIK		CCB FLAG**	Unused	CCBVA Virtual Address of CCB			
32	CCBACB Address of first CCW copy block in channel program				CCB1CB Address of first IDAL block in channel program			
40	CCBXINF (Fix information; 24 bytes)							
48	Each bit in this field represents one page frame. If a bit is on, the associated page frame contains a page fixed for this I/O request. If more than 384K of real storage are available, the address in CCBXPTR will point to any additional field which contains bits for the page frames beyond 384K							
56								
64	CCBXPTR Address of additional Fix information				CCBNEXT Address of next CCB copy block			

If fast CCW translation is specified (FASTTR=YES), the last twelve bytes of the block are changed to:

		CCBRCB Address of REPLICA block
64	CCBFPT Forward chain pointer	CCBBPT Backward chain pointer

* Set to X'20' (=copied CCB)

** Legend CCBFLAG: Bit 0: Indicates that CCW-translation of this request is complete; indicator is set before I/O request is enqueued in channel queue.

1: Indicates that at least one time during CCW-translation control has been transferred to TFIX routine.
in case TTFIX is 0 scan through CCBXINF for freeing pages is skipped. Indicator is set immediately before control is transferred to TFIX routine.

2: Unused.

3: Indicates that the next CCW-translation request from BTAM is from the BTAM channel appendage. This indicator is set immediately after 1st time request from BTAM has been completed.

4: Valid for fast translation.

5: CCB copy block is on saved CCB queue.

6: Data area pages require fixing.

7: Unused.

COPY BLOCKS (....Cont'd)

Layout of CCW Copy Block

	0	1	2	3	4	5	6	7
0	1st Copy location for CCW							
8	2nd Copy location for CCW							
16	3rd Copy location for CCW							
24	4th Copy location for CCW							
32	5th Copy location for CCW							
40	6th Copy location for CCW							
48	7th Copy location for CCW							
56	X'80'*	X'000000'			Virtual address of first CCW in the Copy block			
64	X'88'**	X'000000'			Address of next CCW Copy block in the chain			

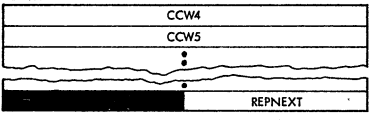
* X'80' indicates the end of the CCW copy locations in the block. It is replaced by a TIC (Transfer in Channel command) if the 7th copy location contains a copied CCW with data- or command chaining. Bytes 57-59 will then point to the copy location of the CCW following the CCW in the 7th copy location.
Bytes 56-59 will not be changed if the CCW in the 7th copy location is a TIC.

** X'88' indicates the last 8-byte entry in the block. It is replaced by a TIC if the CCW in the 7th copy location is a status modifier CCW. Bytes 65-67 will then point to the copy location of the second CCW following the status modifier CCW.

MAIN REPLICA BLOCK

VCCBA		RCCBA		
TIMTEST		REPPIK	REPLCNT	CCWSTRL
REPDIDAL		REPDCB		
CCB (16 or 24 Bytes)				
CCW1				
CCW2				
CCW3				
REPFPT	REPBPT	REPNEXT		

ADDITIONAL REPLICA BLOCK



Legend:

VCCBA	- Virtual CCB address
RCCBA	- Address of copied CCB
TIMEST	- Timestamp
REPPIK	- Partition Identification Key
REPLCNT	- The number of tasks currently testing this REPLICA for a match with their channel program
CCWSTRL	- Length of CCW string (number of CCW's)
REPDIDAL	- Address of DIDAL block
REPDCB	- Pointer to chain of DIDAL blocks
REPFPT, REPBPT	- Pointers used for chaining REPLICA's (forward and backward pointer)
REPNEXT	- Pointer to (next) additional REPLICA block

DIDAL BLOCK

ENTRY	1
	2
	3
	4
	5
	6
	7
	8
	CHAIN POINTER***

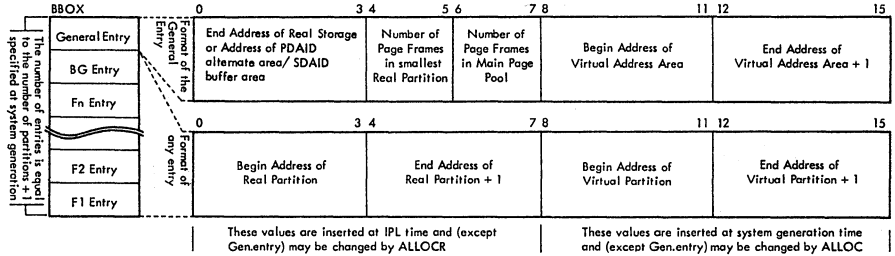
DIDAL ENTRY

* FLAGBYTE: bit 0 - indicates that TFIXing is not is not neccessary because the page has already been TFIXed for this request.
bit 7 - indicates that TFIX request for this entry has been completed.

** Real location (either copied CCW or IDA word) that should contain the translated I/O area address.

*** Points to (next) additional DIDAL. Contains 4X'00' in last DIDAL.

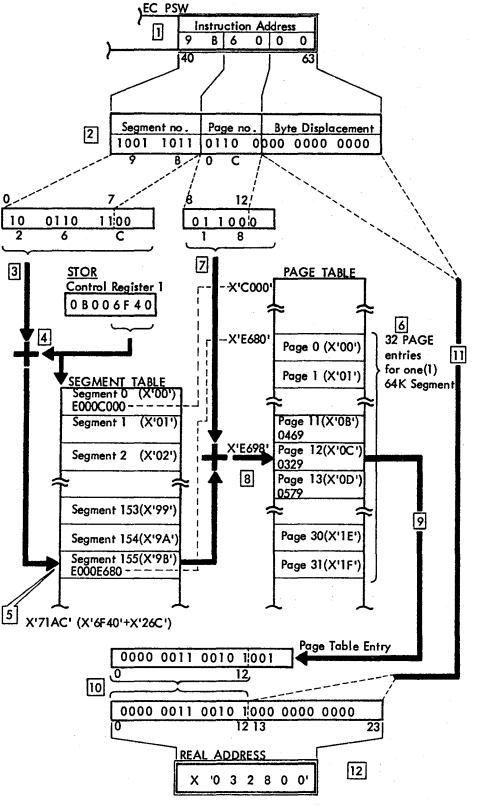
BOUNDARY BOX



Notes: If a partition is not allocated, its beginning- and end address are identical with the beginning address of the following partition.

Bytes 220-223 (X'DC'-X'DF') of the System Communication Region (SYSCOM) contain the address of the Boundary Box. Label BBOX identifies the first byte of the table.

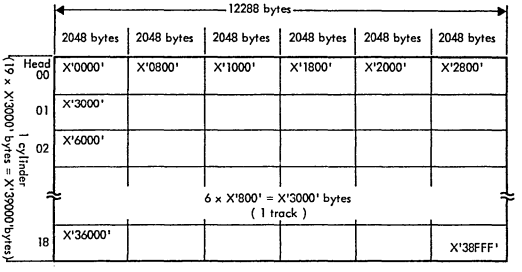
CONVERTING VIRTUAL TO REAL ADDRESS



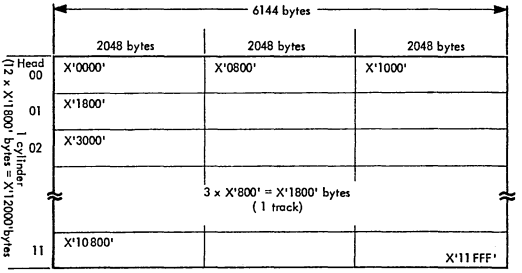
Note : Values used in the figure are hypothetical.

PAGE DATA SET FORMAT

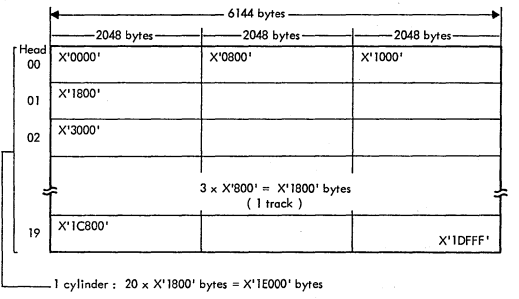
Virtual Disk Format 3330



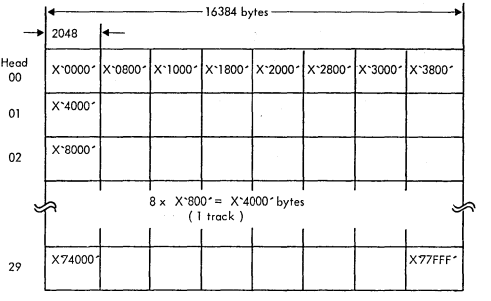
Virtual Disk Format 3340



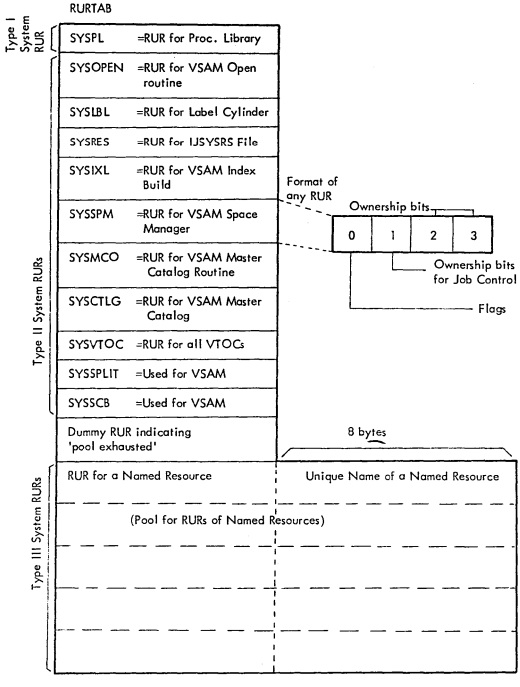
Virtual Disk Format 2314



Virtual Disk Format 3350



RESOURCE USAGE RECORD TABLE (RURTAB)



Byte 0 :

Bit 3: 1= Another task waiting for this resource
Bit 5: 1= Resource shared among owners indicated
Bit 6: 1= Resource used exclusively by indicated task
Bit 7: 1= Shared or exclusive use of resource

Byte 1 : Used by Job Control to indicate that the resource is to be held for the duration of a job.

Bit 0: 1= TIK or PIK is X'70'
Bit 1: 1= TIK or PIK is X'60'
Bit 2: 1= TIK or PIK is X'50'
Bit 3: 1= TIK or PIK is X'40'
Bit 4: 1= TIK or PIK is X'30'
Bit 5: 1= TIK or PIK is X'20'
Bit 6: 1= TIK or PIK is X'10'

Byte 2 :

Bit 0: 1= TIK is X'F0'
Bit 1: 1= TIK is X'E0'
Bit 2: 1= TIK is X'D0'
Bit 3: 1= TIK is X'C0'
Bit 4: 1= TIK is X'B0'
Bit 5: 1= TIK is X'A0'
Bit 6: 1= TIK is X'90'
Bit 7: 1= TIK is X'80'

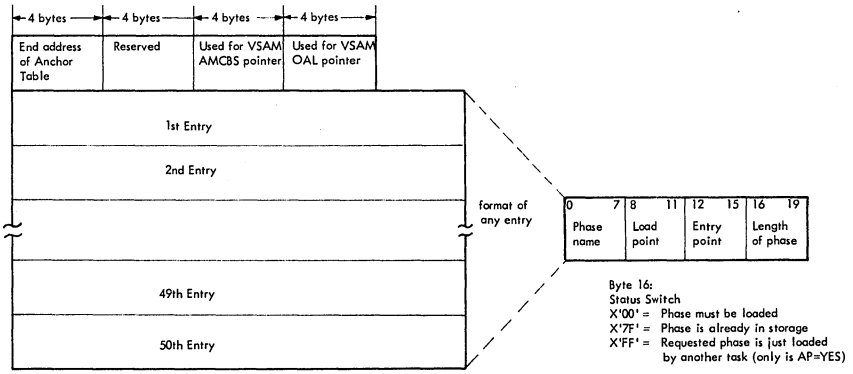
Byte 3 :

Bit 0: 1= TIK or PIK is X'70'
Bit 1: 1= TIK or PIK is X'60'
Bit 2: 1= TIK or PIK is X'50'
Bit 3: 1= TIK or PIK is X'40'
Bit 4: 1= TIK or PIK is X'30'
Bit 5: 1= TIK or PIK is X'20'
Bit 6: 1= TIK or PIK is X'10' (BG)
Bit 7: 1= TIK or PIK is X'00' (Attention)

Note: For explanation of TIK and PIK see Task Interrupt Key and Partition Identification Key.

Label RURTAB identifies the first byte of the table.

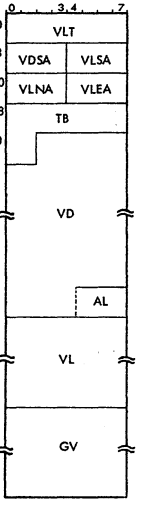
ANCOR TABLE (ANCHTAB)



Note :

The Anchor Table is located at the beginning of the GETVIS area in the virtual partition.

LAYOUT of SHARED VIRTUAL AREA (SVA)



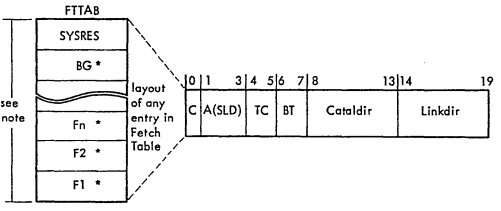
Field	Length	Content
VLT	8	Date + time SVA has been updated (note 1)
VDSA	4	Start address of the System D Directory List
VLSA	4	Start address of the virtual library (note 2)
VLNA	4	Address of next available location in SVA
VLEA	4	Address of end of the SVA
TB	10	Communications area for Job Control and $MAINDIR
VD	up to 32K	System Directory List (note 3)
AL	<8	Alignment bytes for doubleword boundary
VL	any	Virtual library containing re-entrant and relocatable phases
GV	any	GETVIS area for the system (starts on page boundary)

Note 1 : As one of the tests to determine whether a warm start copy of the SVA is available, job control compares this entry after IPL against the date and time in the SCIL descriptor entry.

Note 2 : Address of first doubleword aligned byte after SDL.

Note 3 : The layout of this area is compatible with a directory block in the Core Image Library. The SDL has fixed length entries of 34 bytes. The last entry contains 8X'FF' as phasename.

FETCH TABLE (FTTAB)

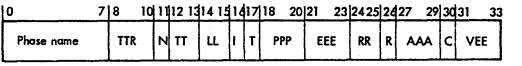


Byte 0 :	Condense counter (incremented for each condense of the library . Reset to zero when a new library is assigned - PCIL only)
Bytes 1-3 :	Address of the corresponding Second Level Directory (zero if no SLD present)
Bytes 4-5:	Number of tracks per cylinder
Bytes 6-7:	Number of library blocks per track
Bytes 8-13:	Address of directory of cataloged phases in the form BBCCHH
Bytes 14-19:	Address of Linkarea in the form BBCCHH (zero if no Linkarea present)

Note : The address of the FETCH TABLE is contained in bytes 240-243 (X'F0'-'F3') of the System Communication region (SYSCOM). Label FTTAB identifies the first byte of the table.

* The entries for the partitions will not exist if the support for Private Core Image Libraries is not included in the Supervisor.

LAYOUT of DIRECTORY ENTRY

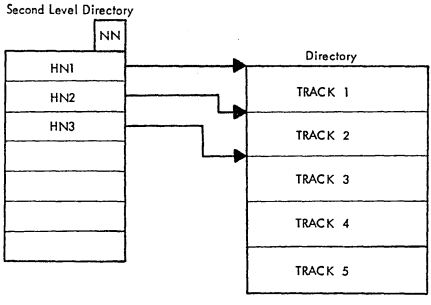


Bytes 0-7 :	Phase name	
Bytes 8-10:	TTR	Relative track address and record number of phase (note 2)
Byte 11 :	N	Number of halfwords containing user data
Bytes 12-13:	TT	Number of text blocks
Bytes 14-15:	LL	Number of text bytes in last text block
Byte 16 :	I	Switch indicating type of phase X'80' : selfrelocating phase X'40' : relocatable phase X'20' : SVA eligible X'10' : phase is in the SVA (note 1) X'08' : phase found in a PCIL (note 1) X'04' : phase not found (note 1) X'02' : active entry (filled in) X'01' : not used
Byte 17 :	T	Always X'00' (used as type byte for stow table)
Bytes 18-20:	PPP	Load point at LNKEDT time (note 3)
Bytes 21-23:	EEE	Entry point at LNKEDT time (note 3)
Bytes 24-25:	RR	Number of RLD items (note 4)
Byte 26 :	R	Number of additional RLD blocks (note 4)
Bytes 27-29:	AAA	Partition start address at LNKEDT time (note 4)
Byte 30 :	C	Condense counter at the time when entry was activated (note 6)
Bytes 31-33:	VEE	Entry point of phase in SVA (notes 1 and 5)

Note 1 : Only used for directory entries that are in storage.
Note 2 : The TTR is relative to the beginning of the directory.
Note 3 : PPP and EEE are not present if both are zero and the phase is not relocatable.
Note 4 : RR, R and AAA are only present if the phase is relocatable.
Note 5 : VEE is only present if the phase is SVA eligible.
Note 6 : Condense counter is used for incore directory entries (not SDL)

Note : The last entry in the directory is a 12-byte entry with a dummy phasename (contains 8X'FF'), a dummy TTR (contains XL3'00') and a dummy N (contains X'00'). Directory entries in storage always have the standard length of 34 bytes (including the last entry).

RELATIONSHIP between DIRECTORY and its SECOND LEVEL DIRECTORY



Field	Length	Content
NN	2	Number of entries
HN	8	Highest phase name on corresponding directory track

Note : Track 4 and 5 are not used.

DIRECTORY SEARCH SEQUENCES (INCLUDING SDL CONSIDERATION)

	Options					Directories					
	SDL	DE or LDL	or SYS	PCIL	LINK	SLD	SCIL LD	PSLD	PCIL LD	DE or LDL	SDL
1	No	No	No	No	No	1					
2	No	No	No	No	Yes	2	1				
3	No	No	No	Yes	No	2		1			
4	No	No	No	Yes	Yes	3		2	1		
5	No	No	Yes	No	No	1					
6	No	No	Yes	No	Yes	1	2				
7	No	No	Yes	Yes	No	1		2			
8	No	No	Yes	Yes	Yes	1		2	3		
9	No	Yes	No	No	No	2				1	
10	No	Yes	No	No	Yes	3	2			1	
11	No	Yes	No	Yes	No	3		2		1	
12	No	Yes	No	Yes	Yes	4		3	2	1	
13	No	Yes	Yes	No	No	2				1	
14	No	Yes	Yes	No	Yes	2	3			1	
15	No	Yes	Yes	Yes	No	2		3		1	
16	No	Yes	Yes	Yes	Yes	2		3	4	1	
17	Yes	No	No	No	No	2					1
18	Yes	No	No	No	Yes	3	1				2
19	Yes	No	No	Yes	No	3		1			2
20	Yes	No	No	Yes	Yes	4		2	1		3
21	Yes	No	Yes	No	No	2					1
22	Yes	No	Yes	No	Yes	2	3				1
23	Yes	No	Yes	Yes	No	2		3			1
24	Yes	No	Yes	Yes	Yes	2		3	4		1
25	Yes	Yes	No	No	No	3				1	2
26	Yes	Yes	No	No	Yes	4	2			1	3
27	Yes	Yes	No	Yes	No	4		2		1	3
28	Yes	Yes	No	Yes	Yes	5		3	2	1	4
29	Yes	Yes	Yes	No	No	3				1	2
30	Yes	Yes	Yes	No	Yes	3	4			1	2
31	Yes	Yes	Yes	Yes	No	3		4		1	2
32	Yes	Yes	Yes	Yes	Yes	3		4	5	1	2

VTAM ADDRESS VECTOR TABLE (ISTAVT)

Label	Length	Description
ISTACVT	4	Address of Communications vector table in VTAM partition
ISTAS49	4	SVC 49 code routine address
ISTAS53	4	SVC 53 code routine address
ISTAPSTA	4	Address of APS table
ISTTXTSZ	4	Address of size of text record
ISTAPSEX	4	Address of APS disposition exit
ISTPHNM	5	Phase name
ISTX1	1	Used for TOLTEP to save the condition code
ISTARID	2	Address of RID
ISTATRT	2	Address of VTAM gates in resource table
ISTAGTWT	2	Address of gatewait routine
ISTVTTP	2	Address of SVC 11 routine

Bytes 96-99 (X'60'-X'63') of the system communication region (SYSCOM) contain the address of the VTAM Address Vector Table.

ENTRY IN THE ASYNCHRONOUS PROCESS SCHEDULER (APS) OPTION TABLE

Label	Length	Description
APSFLAG	1	X'80' = Exit requested
	11	Reserved
VTAMFLAG	1	X'01' = Subtask runs in supervisor state X'04' = VTAM SVC active X'02' = VTAM appendage active X'08' = VTAM user exit in control X'10' = VTAM timer delay
	3	Reserved

ROUTINE IDENTIFIERS (RID)

NAME	ID	MEANING	ACTION
SYSTEMID	00	System error condition, for example, page fault in I/O interrupt handler	Hard Wait
REENTRID	04	Page fault or GETREAL request in a reenterable routine	Save PSW and regs to user task's system save area, set PIB DAT flag to call SVRETURN, and ENQU request
USERTID	08	Page fault from a disabled user task or disabled B-transient	Cancel user task error code X'15'.
APPENDID	0C	Page fault in I/O appendage routine	Cancel usertask error code X'36'.
RESVCID	10	Page fault in SVC 29 or 47	Set saved PSW to re-issue SVC when task is dispatched, ENQU page fault request.
DISPID	14	Page fault in a routine which requires no information to be saved, for example, a page fault in the dispatcher.	ENQU page fault request.
PFARID	16	Page fault in a page fault appendage routine	Cancel user task error code X'15'.
	47 thru 4F	Request for a page which is to be PFIXed. RID identifies the partition which issued the PFIX request. Used only if AP=YES.	Save PSW and registers to user task's system save area, set PID DAT flag to call INITSVC when task is dispatched next. Close gate to routine (routine cannot be used until gate is opened). ENQU page request. (Any task trying to use a gated resource is placed in a wait state and marked resource bound. It is released from the wait state when the resource is ungated after the page request has been handled.)
G41BND	53	Page fault in the reusable SVC 41 or 42.	
G58BND	71	GETREAL request from the reusable SVC 58.	

CHAPTER V
DOS/VS SERVICE AIDS

PD AIDS

System Generation Requirements

The following System Generation macro parameters apply to PDAIDS:

Macro	Parameter	Remarks
FOPT	PD={NO / YES / n}	The PD parameter must be specified if the problem determination programs (PDAIDS) are desired. PD= YES specifies that a minimum of 1400 bytes is reserved for this function. Any amount between 1400 and 10,240 bytes may be specified for n.

The PDAIDS routines must be catalogued in the Core Image Library.

Linkage Edit statements:

```
   INCLUDE  IJBPDAID
/*
// EXEC  LNKEDT

   INCLUDE  IJBPDLST
/*
// EXEC  LNKEDT
```

INITIALIZATION

After linkage editing PDAIDS routines into the Core Image Library, the initializer program is loaded with standard DOS/VS job control statements.
The statement: // EXEC PDAID causes the initializer to be loaded. The initializer types out the first keyword PDAID= and opens the keyboard for a response.
A choice can be made at this point to enter the parameters from the printer keyboard or from SYSIPT via cards. Following is a description of each of these methods:

Initialization via card:

1 Punch desired keywords and parameters into cards. (See PDAID control statements table for the appropriate PDAID functions, in this chapter).
Entries may be punched one - per - card, or multiple entries (separated by commas) can be made in a single card. An entry may not be split between two cards.
All 80 columns of a card may be used, but a card is terminated by the first blank following an entry, or a GO entry. The last entry of the last card must be GO and the last card must be followed by a /* card.

2 Place keyword entry card(s) in SYSIPT.

3 Respond to the SYSLOG message PDAID= by pressing only the END key on the printer keyboard.

4 The initializer reads cards from SYSIPT until the GO entry is encountered.

Note : If a wrong parameter is encountered on a card, corrections are requested on SYSLOG.

Initialization via printer keyboard:

1 Enter one of the parameters (IT,FT,TD,GT, QT, VT or VP) indicating the specific PDAID function you want to initiate.

2 The initializer will respond with the next keyword:
Respond with the appropriate parameter
(See PDAID control statements table for the appropriate PDAID functions, in this chapter).
The initializer will continue to prompt with the appropriate keywords until all options have been satisfied or a GO response is received, whereupon defaults are selected for any remaining options.
Pressing the END key in response to any keyword causes selection of the default option and causes the initializer to respond with the next option available.

4 The sequence in which keywords are prompted depends on previous responses. Because AAA is valid only in core-wrap mode, it does not occur unless the response to "OUTPUT DEVICE= " is END(key). (which selects core-wrap mode by default).
Because "IGNORE DEVICE/SVC= " and "TRACE DEVICE/SVC= " are mutually exclusive, "TRACE DEVICE/SVC= " is prompted only after a default response to "IGNORE DEVICE/SVC= " .

TERMINATION

Any PDAID function can be terminated by re-initializing the PDAID program with the job control statement: //EXEC PDAID
and responding to the message PDAID with XX.
It is also possible to reset one function by loading another.

Note: If the PDAID function is terminated and an alternate area was assigned for core-wrap mode before this area is released to the main page pool, the contents of this area is printed on SYSLST.
If SYSLST is unassigned, the printout of this area is suppressed and the content of this area will be destroyed.

PDAID FUNCTIONS

1 I/O Trace

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{IT XX}	IT: Initiate I/O trace XX: Terminates	None Function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specifiy the hexadecimal channel and unit number of either a magnetic tape unit or a printer for the output device of the PDAID.	Core-wrap mode
AAA= (note 1)	[{nK END GO}]	nK: Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode.*) The value n must be an even integer. If not an even integer, n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode, using PD-save area

1 I/O Trace (....Continued)

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
IGNORE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be ignored by I/O trace. Note: A maximum of 3 may be specified.	Trace all devices
TRACE DEVICE=	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be traced by the I/O trace. Note: A maximum of 3 may be specified.	Trace all devices
GO (valid SYSIPTparameter)		GO terminates the PDAID control input and uses the default for those options that are not specified.	None

Note 1: OUTPUT DEVICE and AAA are mutually exclusive.
Note 2: TRACE DEVICE and IGNORE DEVICE are mutually exclusive.

2 Fetch/Load trace

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{FT XX}	FT: Fetch/Load trace XX: Terminates	None Function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specify the hexadecimal channel and unit number of either a magnetic tape unit or a printer for the output device of the PDAID.	Core-wrap mode
AAA= (note 1)	[{nK END GO}]	nK: Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode.*) The value n must be an even integer. If it is not an even integer, n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode, using PD-save area
TRACE PARTITION=	[{SP BG Fn END GO}]	SP: Supervisor BG: Background Fn: Foreground n	Trace all partitions and the supervisor

2 Fetch/Load Trace (....Continued)

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
GO (Valid SYSIPT parameter)		GO terminates the PDAIDcontrol input and uses the default for those options that are not specified.	None

Note 1: OUTPUT DEVICE and AAA are mutually exclusive.

3 GSVC Trace

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{GT XX}	GT: GSVC trace XX: Terminates	None function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specify the hexadecimal channel and unit number of either a magnetic tape unit or a printer for the output device of the PDAID.	Core-wrap mode
AAA= (note 1)	{nK END GO}	nK: Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode.*) The value n must be an even integer. If it is not an even integer, n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode using PD-save area
TRACE PARTITION=	[{SP BG Fn END GO}]	SP: Supervisor BG: Background Fn: Foreground n	Trace all partitions and the supervisor
IGNORE SVC= (note 2)	[{nn END GO}]	Specify the hexadecimal SVC number to be ignored by the GSVC trace. Amaximum of 6 may be specified.	Trace all SVC's
TRACE SVC= (note 2)	[{nn END GO}]	Specify the hexadecimal SVC number to be traced by the GSVC trace. A maximum of 6 may be specified.	Trace all SVC's
GO (valid SYSIPT parameter)		GO terminates the PDAID control input and uses the default for those options that are not specified.	None

3 GSVC Trace (....Continued)

Note 1: OUTPUT DEVICE and AAA are mutually exclusive.
Note 2: TRACE SVC and IGNORE SVC are mutually exclusive.

4 QTAM Trace

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{QT XX}	QT: QTAM Trace XX: Terminates	None Function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specify the hexadecimal channel and unit number of a magnetic tape unit for the output device of the PDAID. Note: A printer is invalid for QTAM trace.	Core-wrap mode
AAA= (note 1)	{nK END GO}	nK: Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode.*) The value n must be an even integer. If it is not an even integer n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode, using PD-save area
TRACE PARTITION=	[{SP BG Fn END GO}]	SP: Supervisor BG: Background Fn: Foreground n	Trace all partitions and the supervisor
IGNORE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be ignored by the QTAM trace. A maximum of 3 may be specified.	Trace all devices
TRACE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be traced by the QTAM trace. A maximum of 3 may be specified.	Trace all devices
GO (valid SYSIPT parameter)		GO terminates the PDAID control input and causes the default for those options that are not specified.	None

Note 1: OUTPUT DEVICE and AAA are mutually exclusive.
Note 2: TRACE DEVICE and IGNORE DEVICE are mutually exclusive.

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{VT XX}	VT : VTAM Trace XX : Terminates	None Function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specify the hexadecimal channel and unit number of a magnetic tape unit for the output device of the PDAID. Note : A printer is invalid for VTAM trace.	Core-wrap mode
AAA= (note 1)	{nK END GO}	nK : Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode. *) The value n must be an even integer. If it is not an even integer n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode, using PD-save area
TRACE PARTITION=	[{SP BG Fn END GO}]	SP: Supervisor BG: Background Fn: Foreground n Only SVCs 49 and 53 are recorded per partition	Trace all partitions and the supervisor
IGNORE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be ignored by the VTAM trace. A maximum of 3 may be specified.	Trace all devices
TRACE DEVICE = (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be traced by the VTAM trace. A maximum of 3 may be specified.	Trace all devices
GO (valid SYSIPT parameter)		GO terminates the PDAID control input and causes the default for those options that are not specified.	None

Note 1: OUTPUT DEVICE and AAA are mutually exclusive.
Note 2: TRACE DEVICE and IGNORE DEVICE are mutually exclusive.

6 VTAM BUFFER pool trace

SYSLOG message SYSIPT parameter	SYSLOG reponse SYSIPT operand	Meaning	Default
PDAID=	{VP XX}	VP : VTAMBUFFER pool trace XX : Terminates	None Function continues
OUTPUT DEVICE= (note 1)	{cuu X'cuu' END GO}	Specify the hexadecimal channel and unit number of a magnetic tape unit for the output device of the PDAID. Note : For VTAM BUFFER pool trace only a magnetic tape unit is valid.	Core-wrap mode
AAA= (note 1)	{nK END GO}	nK : Specifies the number (n) of "K" bytes to be allocated as alternate area for core-wrap mode.*) The value n must be an even integer. If it is not an even integer n+1 is taken. *) This area will be taken from the main page pool.	Core-wrap mode, using PD-save area
TRACE PARTITION=	[{SP BG Fn END GO}]	SP: Supervisor BG: Background Fn: Foreground n Only SVCs 49 and 53 are recorded per partition	Trace all partitions and the supervisor
IGNORE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be ignored by the VTAM trace. A maximum of 3 may be specified.	Trace all devices
TRACE DEVICE= (note 2)	[{cuu X'cuu' END GO}]	Specify the hexadecimal channel and unit number of the device to be traced by the VTAM trace. A maximum of 3 may be specified.	Trace all devices
GO (valid SYSIPT parameter)		GO terminates the PDAID control input and causes the default for those options that are not specified.	None

Note 1 : OUTPUT DEVICE and AAA are mutually exclusive.
Note 2 : TRACE DEVICE and IGNORE DEVICE are mutually exclusive.

7 Transient Dump

SYSLOG message SYSIPT parameter	SYSLOG response SYSIPT operand	Meaning	Default
PDAID=	{TD} {XX}	TD: Transient Dump XX: Terminates	None Function continues
OUTPUT DEVICE=	{cuu} {X'cuu'}	Specify the hexadecimal channel and unit number of either a magnetic tape unit or a printer for the output device of the PDAID.	None
GO (valid		GO terminates the PDAID control input.	None

TAPE TO PRINTER UTILITY PROGRAM (PDAIDLST)

The tape-to-printer utility program reads the tape input(created by the PDAID tape routines) from SYS 005, and then prints the information on SYSLST in the same format that is used by the PDAID print routines.
The tape-to-printer utility is loaded with standard DOS/VS job control statement:
// EXEC PDLIST
Input on tape may contain output from more than one PDAID function.

Note: When tape is used as output device, always reset(terminate) the function before executing PDLIST

SD AIDS

GENERAL

Requirements

- DOS/VS System
- Line printer
- At least 6K of page pool

Linkage Edit Statements

```
   INCLUDE IJBSDLNK
// EXEC LNKEDT
```

Restrictions

- Printer is only output device
- No burst mode device must be running on same channel with printer
- Debugging of printer error recovery routines only possible with output class FASTREC
- Only real storage can be dumped
- Time dependent programs may not be debugged
- Alternate Area of PDAID cannot be used concurrently

FUNCTIONS

Events

There are two groups of events : Elementary events and Dedicated events.

Events appear as program checks.

Elementary events are :

Mnemonic	Event
BR	Successful branching
IF	Instruction fetching
SA	Storage alteration
GA	General register alteration
TE	Page translation exception

The output for the elementary events depends on the specified output class

Dedicated events are :

Mnemonic	Event
PGMCHK	Program interruption codes X'01'-X'0F', X'10' and X'12'
PAGENQ	Request for page is enqueued
PAGEHDL	Request for page is handled

The events PAGENQ and PAGEHDL are signalled by hooks in the page management routines.

The output for dedicated events is predefined and specific to the event.

OUTPUT

Dedicated events

The following predefined and events specific data is printed :

Recorded Information	Dedicated Event		
	PGMCHK	PAGENQ	PAGEHDL
Event Mnemonic, Program old PSW, Time of day (usec), Complete supervisor, Instruction at time of PGMCHK, Control Registers, General purpose registers	•		
Event Mnemonic, Requestor ID, (TE/GETR/TFIX/PFIX) Task ID, Virtual address of page to be handled		•	•
Protection key associated with virtual page to be handled		•	•
Real address of page frame to which the page is assigned			•

Elementary events

The output class (OUTCL) determines the amount of the information to be recorded for events BR, IF, SA, GA, and TE.

Recorded Information	Output Classes Mnemonic								
	01	02	03	04	05	06	07	08	00
	PSW	GPR	LOW CORE	COM REG	PAGE TAB	SUPV	DUMP REAL NDD	PDUMP	FAST REC**
Event ID*,progr old PSW, time of day (usec)	•	•	•	•	•	•	• ***	•	•
Instruction causing event	•	•	•	•	•	•	• ***	•	
GPR's		•	•	•	•	•	•	•	
Low Core			•	•	•	•	•		
Current COMREG and SYSCOM				•		•	•		
Control registers			•	•	•	•	•	•	
Segment-, page- and page frame tables					•	•	•		
Complete superv						•	•		
Complete real storage							•		
Area between virtual address limits								•	
TE-MASK PER mask(ctlr 9) GPR mask (ctlr 9) PER start address (ctlreg 10) PER end address (ctlreg 11) GPR's 13,14,15, 0,1,2									•

* Event ID for BR, IF, SA and GA= event mnemonic and instruction address
Event ID for TE= mnemonic TE and address of the virtual page causing TE

** FASTREC is an output class that stores the described information into an SDAID internal buffer. Information for several events is stored and printed as one block

*** INSTR and PSW are not printed if NDD is forced after STOP ON EVENT VIA NDD BYTE X'FF'

COMMUNICATION

Initialization

Communication with SDAID

Console	Operator Action	Remarks
	//EXEC SDAID [,REAL, SIZE= 12K]	SDAID may only be used in a DOS/VS system. If it is initialized in a real partition, SIZE parameter is also required.
4C55D GIVE SPACE FOR SDAID=	[6K / nK] END/ENTER	SD-area requires at least 6K at page pool. If OUTCL=FASTREC is used it is recommended to give at least 8K.
4C70E DUPLICATE REQUEST FOR PDAID AND/OR SDAID		Use of SDAID and alternate area of PDAID is exclusive.
4C56E INSUFFICIENT SDAID SPACE, REALLOCATE	Reallocate real partitions	Message appears only if increasing of the page pool is required.
4C58D OUTPUT DEVICE=	[X'00E' / X'CUU' [,GO] / GO] END/ENTER	Device must be a line printer; No burst mode device must be running on the same channel.
4C60D STOP ON EVENT=	[YES / NO [,GO]] END/ENTER	A wait PSW with address part X'0000 EEEE' will be loaded. The NDD byte may be set to X'FF' to get the non-destroying DUMP. Processing continues after pressing the external interrupt key.
4C61D EVENT LIMITS=	[X'000000' ,X'FFFFFF' [,GO] / X'LLLLLL' [,X'LLLLLL' [,X'hhhhhh' [,GO] / ,GO] / ,GO]] END/ENTER	Lower and upper limit of virtual storage to be traced: . Instruction address on events BR, IF, TE and GA . Altered address on event SA If lower limit is higher than upper limit, wrap around occurs.

Communication with SDAID

Console	Operator Action	Remarks
4C59D OUTCL=	[PSW / GPR / LOCORE / COMREG / PAGETAB / SUPVISOR / DUMPREAL [,GO]] PDUMP [,lower and upper event limit [,GO] / ,X'aaaaaa' [,upper event limit / ,X'bbbbbb' [,GO] / ,GO] / ,GO] FASTREC [,PGMCHK / ,AUTOMATIC [,GO] / ,GO] END/ENTER	OUTCL may be specified for events BR, IF, SA, GA and TE. For the other events, predefined data is recorded. DUMPREAL and PDUMP cause dumping of real storage only. FASTREC may be used for printer debugging. PGMCHK causes wrap around mode of internal buffer which is written out each time a PGMCHK occurs. AUTOMATIC causes writing out of the internal buffer if it is full.
4C62D EVENT BR=	[YES / NO [,GO]] END/ENTER	Event appears after successful branching.
4C63D EVENT IF=	[YES / NO [,GO]] END/ENTER	Event appears after instruction fetching.
4C64D EVENT SA=	[YES / NO [,GO]] END/ENTER	Event appears after storage alteration.
4C65D EVENT GA=	[X'0 1 2 . . . E F' / NO [,GO]] END/ENTER	Event appears after alteration of one or more specified General Purpose Register(s).

Communication with SDAID

Console	Operator Action	Remarks
4C66D EVENT TE=	[YES / NO] [,GO] END/ENTER	Event appears after page translation exception
4C67D EVENT PGMCHK=	[YES / NO] [,GO] END/ENTER	Event appears after PGMCHK interruption with codes X'01'-X'10' and X'12'. Output is predefined.
4C68D EVENT PAGENQ=	[YES / NO] [,GO] END/ENTER	Request for page is enqueued. Event is signalled by a hook in Page Management Routine. Output is predefined.
4C69D EVENT PAGEHDL=	[YES / NO] END/ENTER	Request for page is handled. Event is signalled by a hook in Page Management Routine. Output is predefined.

If GO is entered for a parameter, the dialogue is terminated immediately and defaults (underscored) are taken for the remaining parameters.

Changing

When the SDAID is initialized, parameters can be changed with the CPU in stopped state.
The procedure (see example) printed at initialization time shows which control registers or which locations in real storage have to be altered.

Example

These are the parameters that you can change from the console:
Procedure: Machine in stopped state, hit ALTER/DISPLAY. Examples below:

To alter ctl reg		9 to 40000000		
....type in	AC	9 40000000	END key	
To alter MS location		03CC8C to FF		
....type in	AM	03CC8C FF	END key	
Printer Address		on hex 033480	Format 0XXX	
Output Class		on hex 033490	Form UUYY	
where IF:		UU=00 FASTREC		
		UU=01 PSW		
		UU=02 GPR		
		UU=03 LOCORF		
		UU=04 COMREG		
		UU=05 PAGETAB		
		UU=06 SUPVISOR		
		UU=07 DUMPREAL		
		UU=08 PDUMP		
		YY=00 Automatic SDBUFFER Output		
		YY=FF SDBUFFER Output on PGMCHK		
Stop on Event		on hex 003348C	X'FF'=YES, X'00'=NO	
Non-destroying Dump		on hex 003348E	X'FF'=YES	
Event Limits	Start Address	Ctl Reg A	Hex 00XXXXXX	ADDR
	End Address	Ctl Reg B	Hex 00XXXXXX	ADDR
PDump Limits	Start Address	On hex NNNNNN	00XXXXXX	ADDR
	End Address	On hex NNNNNN	00XXXXXX	ADDR
SDAID Events		To enable - Switch on the BIT/BYTE -		
BR Event		Ctl Reg 9	Hex 80000000	BIT
IF Event		Ctl Reg 9	Hex 40000000	BIT
SA Event		Ctl Reg 9	Hex 20000000	BIT
GA Event		Ctl Reg 9	Hex 10000000	BIT
	GPR Mask (0-15) ..	Ctl Reg 9	Hex 1000XXXX	POSITIONAL BITS (see note)
TE Event	Main Storage	033484	Hex FF	BYTE
PGMCHK Event	Main Storage	033486	Hex FF	BYTE
PAGENQ Event	Main Storage	033488	Hex FF	BYTE
PAGEHDL Event	Main Storage	03348A	Hex FF	BYTE

Note: SDAID needs external interrupt if printer became unready

****

CPU-ID is .... Hex 0072050301450000

4057 I SDAID successfully initiated

Note: see next page for GPR Alteration Masks

Control Register 9 , General Purpose Register (GPR) Masks:

GPR	Masks
0	80 00
1	40 00
2	20 00
3	10 00
4	08 00
5	04 00
6	02 00
7	01 00
8	00 80
9	00 40
A	00 20
B	00 10
C	00 08
D	00 04
E	00 02
F	00 01

Termination

The SDAID is terminated and the SDAREA is released to the page pool, by using either one of the following:

- The AR (Attention Routine) command ENDSD
- The Job Control Statement // EXEC ENDSD

Note: SDAID can be made inactive (elementary events only) by clearing control register 9.
Abort DUMP or PDUMP: Stop line printer and make ready again, then hit extemal interrupt key twice within one second.

SYSVIS DUMP

General

This program should be used in addition to the Stand-alone Dump to dump information from the Page Data Set.
Control cards may be read from SYSIPT or entered via SYSLOG.
Continuation cards are not allowed, but multiple cards may follow.

CORE IMAGE LIBRARY
Phase name: PDSDM

RELOCATABLE LIBRARY
Module name: IJBPDSDM

Two ways of dumping exist:

- Dump and select from SYSVIS to SYSLST
- Dump whole SYSVIS to SYS001 (tape or disk), then select and print from SYS001 to SYSLST lateron.

from / to		SYSVIS on DISK	SYS001 on DISK	SYS001 on TAPE
SYS001 on	DISK	•		
	TAPE	•		
SYSLST		• *	• *	• *

* with selection

Assign/label requirements:

from SYSVIS:
```
// ASSGN SYS000,X'CUU'
// DLBL PDSDISK,'PAGE DATA SET',99/365,SD
// EXTENT SYS000,,1,,,
```

from SYS001 (disk):
```
// ASSGN SYS001,X'CUU'
// DLBL PDSDISK,'BACKUP FOR PDS',99/365,SD
// EXTENT SYS001,....
```

from SYS001 (tape):
```
// ASSGN SYS001,X'CUU'
// TLBL PDSTAPE,'BACKUP FOR PDS'
```

to SYS001 (disk):
```
// ASSGN SYS001,X'CUU'
// DLBL S01DISK,'BACKUP FOR PDS',99/365,SD
// EXTENT SYS001,....
```

to SYS001 (tape):
```
// ASSGN SYS001,X'CUU'
// TLBL S01TAPE,'BACKUP FOR PDS'
```

SYSVIS DUMP (....Cont'd)

To define output device:

TO= [SYSLST [,T]
     [SYS001   ]

meaning:

TO= SYSLST : from SYSVIS to SYSLST
TO= SYSLST,T : from SYS001 to SYSLST
TO= SYS001 (default) : from SYSVIS to SYS001 (without selection possibility)

To select amount of data to be printed

- Partition ID : Total virtual partition will be dumped
- (from,to) virtual addresses : All pages belonging to specified address space will be dumped
- Virtual address : Only one page containing the specified address will be dumped

Note: The virtual address must be in hex and always 6 bytes long.

Examples:

1) Print and select from SYSVIS to SYSLST immediately after IPL :

```
// JOB DUMPPDS1
// ASSGN SYSLST,X'CUU'
// ASSGN SYS000,X'CUU'
// DLBL PDSDISK,'PAGE DATA SET',99/365,SD
// EXTENT SYS000,,1,,,
// EXEC PDSDM
TO= SYSLST
F3,F1,03D170
/*
/&
```

2) Dump all data from SYSVIS to SYS001 (tape) immediately after IPL :

```
// JOB DUMPPDS2
// ASSGN SYS000,X'CUU'
// DLBL PDSDISK,'PAGE DATA SET',99/365,SD
// EXTENT SYS000,,1,,,
// ASSGN SYS001,X'CUU'
// TLBL S01TAPE,'BACKUP FOR PDS'
// EXEC PDSDM
/*
/&
```

3) Print and select from SYS001 (disk) to SYSLST

```
// JOB DUMPPDS3
// ASSGN SYSLST,X'CUU'
// ASSGN SYS001,X'CUU'
// DLBL PDSDISK,'BACKUP FOR PDS',99/365,SD
// EXTENT SYS001,,1,,,
// EXEC PDSDM
TO= SYSLST,T
F3,F1,03D170
(03A891,03(005),F4
/*
/&
```

Terminating the dump

This can be done in any of the three ways given below:

- Enter EOJ on SYSLOG
- Having a /* or a /& card at the end of the jobstream when entering parameters through SYSIPT
- Pressing the END key in response to the message: OV21D GIVE PARAMETERS after at least one address has been processed

DUMPGEN AND STAND-ALONE DUMP

DUMPGEN

DUMPGEN allows you to generate a stand-alone dump program that must be used to obtain information about the system under certain conditions of system malfunction.

The dump consists of a printout of real storage (except bytes X'00' - X'17', X'40'-X'4B', X'BA' - X'BB' and 214 bytes of a non-critical area in the supervisor). Two types of dump programs can be generated using DUMPGEN :

- Translating dump
- Formatting dump.

Both programs produce a convential dump with translation. In addition, the formatting dump produces a pre-formatted printout of the DOS/VS interface tables. This dump is generated if the DUMPGEN option FORMAT=YES is specified.

Executing DUMPGEN

Before being able to execute DUMPGEN you must catalog it to the core image library. Execute it in any partition by the job control statement or command :

```
// EXEC DUMPGEN
```

You enter DUMPGEN and read its control statements from SYSIPT.

Note that SDAIDS may not be initiated during execution of DUMPGEN.
The two types of control statements used with DUMPGEN are ASSGN and OPTN, described as follows :

ASSGN Statement : ASSGN defines the output device for the stand-alone dump program.

Name	Operation	Operand
(blank)	ASSGN	SYSLST, X'cuu'

SYSLST The only valid logical unit assignment.
X'CUU' Must define the address of the SYSLST Printer. If the ASSGN statement is omitted, then X'00E' is assumed.

OPTN Statement : OPTN defines the type of output generated by the DUMPGEN program.

Name	Operation	Operand
(blank)	OPTN	INTR=NO/YES DECKS=nnnnnn PPOOL= NO/YES FORMAT= NO/YES TAPEIPL= NO/YES DISKIPL= NO/YES

Operands for the DUMPGEN option statement

INTR YES produces a DUMP program that, when loaded, enters the WAIT state. Either press the INTERRUPT button on the CPU operating panel to print the output on X'00E', or press the STOP button and then START button of the printer desired for the output device.
NO produces a DUMP program that, when loaded, prints out the contents of real storage either on the SYSLST printer defined with the ASSGN statement or on X'00E'.

DECKS Specifies the number of DUMP card decks (punched out on SYSPCH) desired, nnnnnn may be any decimal number from 1 to 99,999,999. A blank card separates each deck produced. If DECKS is omitted, one deck is produced.

PPOOL YES produces a dump program that, after printing out real storage, will print the formatted contents of the Boundary Box and the contents of the real storage in sequence of ascending virtual addresses.
If NO is specified, the last two items are not printed.

FORMAT YES produces a translating stand-alone dump that formats and displays the DOS/VS supervisor tables after displaying the contents of real storage. This formatted display depends upon the location of the communications region.
If the communications region cannot be related, the program is terminated when the formatted display is to occur.
In this case the following message is printed on the dump output :

COULD NOT FIND COMREG BETWEEN C0 AND A00,
FORMATTING WILL NOT OCCUR.

If the information to be formatted has been destroyed, results are unpredictable.

If NO is specified or FORMAT is omitted, a non-formatting translating dump is generated.

TAPEIPL If YES is specified and SYSPCH is assigned to a tape unit, the stand-alone dump written on tape may be IPLed directly from the tape unit.
If NO is specified, or TAPEIPL is omitted and SYSPCH is assigned to a tape unit, the stand-alone dump records are written on tape preceded by an ASA control character.

DISKIPL If YES is specified and SYS 006 is assigned to a diskette I/O unit, the standalone dumpprogram is written on diskette and can be IPLed directly from diskette.
If NO is specified and SYSPCH is assigned to a diskette I/O unit, the stand-alone dump program is written on diskette, and each record is preceded by an ASA control character.

Control statements for the DUMPGEN operands.

Control statements may be specified in any order, however, the following rules apply:

- All statements may be omitted, but if they are DUMPGEN assigns printer X'00E', INTR=NO, FORMAT=NO, and PPOOL=NO options.

- Only one operation and only one operand per control statement is allowed.

Control statements for the DUMPGEN operands (...cont'd)

- The last statement processed of a duplicate operation overrides all previous statements of the same operation with similar operands (if DECKS=2 is followed by DECKS=5, five decks are punched).
- The name field must be blank.
- Decimal operands may contain leading zeros.
- One or more blanks must follow the operand if comments are to be made.

1. To create a stand-alone dump program on cards or tape:

```
// JOB CARD DUMP
// ASSGN SYSPCH,X'cuu'
// EXEC DUMPGEN
ASSGN SYSLST,X'00F'
OPTN FORMAT=YES
OPTN PPOOL=YES
OPTN DECK=1
/*
/&
```

This will create a single stand-alone dump program on cards. The program will use the printer X'00F' for dump output.

2. To create a stand-alone dump program on diskette:

```
// JOB DISKETTE DUMP
// DLBL IJSYS06, 'filename',,DU
// EXTENT
// ASSGN SYS006, DISKETTE
// EXEC DUMPGEN
OPTN DSKTIPL=YES
OPTN FORMAT=YES
OPTN PPOOL=YES
OPTN DECK= 1
/*
/&
```

This will create one single stand-alone dump program diskette. The program will use the default output printer X'00E', as no ASSGN statement was provided.

DUMPGEN messages

The functions of DUMPGEN-to-operator error message routines are :

- Cancel the job is SYSLOG is not a 3215/3210 or a System/370 Model 125/115 video display unit.
- Reissue the message if operator response is to press the CANCEL key.
- Process an operator response of END/ENTER as IGNORE.
- Cancel the job if operator response is CANCEL.
- Ignore the control card in question when the operator response is IGNORE.

If none of the preceding operator responses is issued, then DUMPGEN assumes that a correction has been made and processes it.

DUMPGEN AND STAND-ALONE DUMP (...Cont'd)

Stand-alone Dump Program (REAL DUMP)

This program is generated for your installation using the IBM program DUMPGEN.

DUMPGEN produces a dump program that is either punched into a card deck or stored on magnetic tape or to SYS006 (diskette). When required, the dump program thus generated can be loaded into the system via the standard IPL procedure.

The stand-alone dump program that is generated by DUMPGEN provides either a conventional dump or a formatted dump, depending on the FORMAT option used in the DUMPGEN program.

Operation

During execution of the stand-alone dump program, a non-critical area in the supervisor is used to load the program. The LOAD ADDRESS of the non-critical area is punched (in decimal) in the first card of the stand-alone dump card deck punched by the DUMPGEN program. Because of this use of the non-critical area it is recommended to use the stand-alone program for a system using a supervisor that was used for the generation of that dump.

The conventional dump prints the contents of real storage locations, but does not dump the floating point registers. In addition to the areas dumped by the conventional dump, the formatted dump prints the DOS/VS interface tables in a more readable form.

For both types of dump the following is printed :

1. The contents of the general registers, the old and new PSWs, the interruption codes, CSW, CAW, and TIMER.
2. The contents of real storage in 2K blocks. Each block is preceded by a sequence number.
3. At the end of the real storage dump, page address and status information is printed that contains the following information for each page frame :

   - The virtual address
   - The real address of the associated page
   - The sequence number of the 2K block
   - Information that indicates whether the contents of the page frames has been changed.

4. The contents of the control registers are printed after page address and status information.
5. Depending on the options selected, the following then occurs :

   If PPOOL=YES

   - The formatted contents of the boundary box is printed after the control registers.
   - The contents of real storage is printed in 2K blocks in sequence of ascending virtual addresses.

Operation (...Cont'd)

If FORMAT=YES
the formatted contents of the tables listed below are printed at the end of the dump .

PARTITION SAVE AREAS
COMREGS
PIBs
AP SUBTASK PIBs (if AP supported)
LUBs
PCIL LUBs (if PCIL supported)
PUB
ERROR RECOVERY BLOCK
CHANNEL QUEUE
FLOATING POINT REGISTERS
COPIES AND TRANSLATED CCB
FIXINF EXT. BLOCKS
COPIED AND TRANSLATED CHANNEL PROGRAM
IDAL BLOCK QUEUE
FIXINF BLOCK
BOUNDARY BOX
SEGMENT TABLE
PAGE TABLE
PAGE FRAME TABLE and PAGE FRAME TABLE EXT
SELECTION POOL

DOSVSDMP AND STAND-ALONE DUMP

DOSVSDMP

This program creates a stand-alone dump program which writes a storage dump in a file on tape or disk. This file can be printed during normal system operation at the user's convenience.

The dump program may reside on tape or disk. When the dump program is executed, the dump is written onto the same device from which it was IPLed.

The program produces a copy of virtual storage in virtual page order, and a copy of real storage in real page order. Pages which are allocated but paged out to SYSVIS are included in the virtual dump.

The storage dump can be printed either formatted or unformatted. In a formatted printout each field identified is printed with certain system control blocks. The unformatted printout is only a conventional translated copy of storage.

Note : 7-track tape drives without data convert are not supported by DOSVSDMP.

1. Record any error or status indicators on the system console.
2. Using ALTER/DISPLAY, record the current PSW, general registers, and floating point registers.
3. Perform the Store Status procedure for your system.
4. Mount and IPL your dump program tape or disk.
5. If message "DOSVSDMP COMPLETED" is displayed, restart DOS/VS and prepare to print the dump as soon as the system has been restarted.
6. If the stand-alone dump program cannot be completed, due to errors, it enters the hard wait state with an error stored in the current PSW. See the section "DOSVSDMP Hard Wait Error Codes" in DOS/VS Messages for details of these codes and subsequent action.

Creating a stand-alone dump program

Before you can execute DOSVSDMP, it must be cataloged to the core image library. The dump program can be generated in a tape resident version or in a disk resident version.

How to create a tape resident stand-alone program

1. Select a non-labeled tape to be used as the dump volume.
2. Execute DOSVSDMP:

```
// JOB
// ASSGN SYS006,TAPE
// EXEC DOSVSDMP
/&
```

3. When message 4C75D is issued, reply : TAPECREATE.
4. Message 4C79I indicates that creation of the dump program tape is completed.
5. Message 4C75D is issued again. Reply : EOJ.

Refer to DOS/VS Messages for an explanation of the above messages.

How to create a disk resident stand-alone dump program

1. Select a disk pack to be used as the dump volume. Note that the IPL text on the pack will be replaced by the dump IPL text. The selected pack cannnot be a system residence volume.
2. Before you can execute DOSVSDMP, the file which is to hold the dump output must be allocated on the selected volume. This file must be named DOSDMPF, and begin and end on a cylinder boundary. The size of the file is calculated using the following formula :

   N = ((2+V+R)/T) + 2

   where

   N is the number of tracks needed,
   V is the number of pages in the virtual address area,
   R is the number of pages in real storage,
   T is: 3 for 2314/3340 devices,
   6 for 3330/3330-11 devices,
   8 for 3350 devices

   The result N must be rounded to the next higher number of tracks equivalent to the next whole number of cylinders.
3. Execute DOSVSDMP:

```
// JOB
// ASSGN SYS006,DISK
// DLBL DOSDMPF,'filename'
// EXTENT SYS006, balance of extent information
// EXEC DOSVSDMP
/&
```

4. When message 4C75D is issued, reply : DISKCREATE
5. Message 4C79I indicates that creation of the dump program disk is completed.
6. Message 4C75D is issued again. Reply : EOJ.

Refer to DOS/VS Messages for an explanation of the above messages.

Executing the stand-alone dump program

The program is activated via an IPL of the tape or disk containing the dump program. The IPL procedure loads the program into and overlays storage locations X'A00' to X'AFF'. Once loaded, the program proceeds to write out storage in virtual page order. Following the virtual storage dump, or as a result of an error in attempting to dump virtual storage, a dump of real storage in real page order is taken. Upon completion of the dump message DOSVSDMP COMPLETE is issued and the system is placed in the wait state with a completion code in bits 48 to 63 of the current PSW.

Printing the stand-alone dump output

For printing the stand-alone dump output, the program requires at least 36K of storage, not including the size of the supervisor.

1. Execute DOSVSDMP:

```
// JOB
// ASSGN SYS006,device
// DLBL DOSDMPF,'filename'
// EXTENT SYS006,balance of extent information
// EXEC DOSVSDMP
/&
```

   SYS006 is assigned to the device which contains the dump file.

Printing the stand-alone dump output (Cont'd)

2. When message 4C75D is issued, reply with one of the following options:

FORMAT	for a formatted printout of the dump file.
FORMAT,REAL	for a formatted printout and a conventional translated printout of real storage following the virtual storage printout.
DUMP	for an unformatted printout of the dump file.
DUMP, REAL	for an unformatted printout and a conventional translated dump of real storage.

3. Message 4C81I will be issued, indicating the default print selection options.
4. Then message 4C80D is issued to allow you to change the print selection options. The options may be entered separately or as one reply, in which case they must be separated by a comma. The options are :

SYSVIS=	YES NO	If SYSVIS=YES is specified the pages retrieved from SYSVIS during the dump will be printed. For unformatted dumps SYSVIS=YES is the only valid option.
SVA=	YES NO	If SVA=YES is specified the Shared Virtual Area will be printed.
PARTITION=	ALL xx	If PARTITION=ALL is specified the supervisor and all partitions will be printed. If PARTITION=xx is specified the supervisor and only the specified partition are printed. xx must be a valid partition ID.

Press END/ENTER to terminate option selection. Message 4C81I is issued to display the selected options.

5. When message 4C75D is issued again, repeat step 2 or reply with EOJ.

PDZAP

This program allows you to make changes to programs cataloged in a system or private core image library. It provides a printout of the changes on SYSLST. This printout should be kept on the installation to keep track of the changes made.

System requirements

PDZAP can be executed in any partition. Since PDZAP accesses a core image library, other programs running currently should not use the phases PDZAP is operating on the same library. SYSLOG must be assigned to the operator console. When card input is used, SYSIPT must be assigned to a card reader (hopper 1 on 5425/2560). SYSLST should be assigned to a printer.

Executing PDZAP

The PDZAP program can be executed from the operator console or from a card reader.

How to execute PDZAP from the operator console

The following is an example in which the program PROG is used as a phase to be modified.

- Call in the program

  // EXEC PDZAP

- The system will respond :

  4C86I IJBPDZAP REL.32.0 program name and version
  4C99A Enter your name. Reply to this message by typing your name.
  4C85A ENTER PHASENAME (XCIL= xxxxxxxx)

- Reply to this message in one of the following ways:

  a. if PROG is in the system core image library enter SCIL=PROG, or simply PROG, as SCIL is the default.
  b. if PROG is in the private core image library enter PCIL=PROG.

- When the phase is found, the following messages are issued:

  4C87I LOAD ADDRESS=xxxxxx
  4C88A ADDRESS XXXXXX, OFFSET+XXXXXX, SCAN=XX..XX, REF=XXXXXX

- Reply to this message in one of the following ways:

  - option 1 specify the hexadecimal address of the data you want displayed : 08072A
  - option 2 specify the offset to the reference point of the data you want displayed (the initial reference point is the load address): +6D4
  - option 3 specify a character string to be searched: SCAN='LABPROG
  - option 4 specify a hexadecimal string to be searched : SCAN=D3C1C2D7D9D6C7
  - option 5 set a reference point : REF=08071C.

How to execute PDZAP from the operator console (continued)

If an address is invalid or a string cannot be found, an error message will be issued.

To options 3 and 4 the system will first respond with :

```
4C941  SCAN ADDRESS xxxxxx
```

Options 1-4 will result in the display of up to 16 bytes of data in the format

```
4C89D
xxxx..xx      cccc..cc
```

where xxxx..xx is the hexadecimal representation and cccc..cc is the corresponding character representation.

The data printed is contained in a single library block. If less than 16 bytes are displayed, it is either the end of the program or the end of the library block.

If only a display is wanted, press END/ENTER.

To modify the displayed data, type in replacement data for the number of bytes to be changed. For instance, if the data displayed is :

```
4C89D
05B012110746410000014790842807F6  ..............6
```

to change the first four bytes to NOP, type in :

```
07000700
```

- The system will respond with message 4C88A again and you can use one of the five options mentioned above to display or modify another portion of the same phase, or to set another reference point.

- If you are finished with this phase, but want to access another phase, use

  - option 6 terminate processing with this phase by typing END PHASE

  Now the program repeats message 4C85A, which allows you to specify the name of another phase.

  If you want to terminate the execution of PDZAP, use
  - option 7 end PDZAP operation by typing: END

How to execute PDZAP from a card reader

Executing PDZAP from SYSIPT is the same as from SYSLOG, with the following exceptions:

- // UPSI 1 must be specified to indicate card input
- data must be verified before it can be changed.

How to execute PDZAP from a card reader (continued)

Example :

	// UPSI 1	indicate card input
	// EXEC PDZAP	call the program
	NAME=your name	specify your name
	SCIL=PROG	specify the phase to be accessed
	+6D4	specifies the position of the data to be displayed (option 2)
	VER=0580,1211	specifies the data to be verified; if the data is
or	VER=05801211	not found, no update will take place
	REP=07000700	specifies the data which is to replace the date
or	REP=0700,0700	just verified
	END	terminates the run

The format of the VER and REP data can be :

- a hexadecimal string (full bytes, or an even number of digits)
- a set of 2-byte entries, separated by commas
- a character string, preceded by a quote (VER='LABPROG).

OLTEP

Functions:

- Diagnosing I/O errors
- Verifying I/O device repairs and Engineering Changes
- Checking I/O devices

System Generation requirements:

The following system generation macro parameters apply to OLTEP :

Macro	Parameter	Remarks
FOPT	OLTEP= YES	Default value.
FOPT	OLTEP= NO	If OLTEP support is not required.
FOPT	OC= YES	1) Forced when OLTEP= YES is specified. 2) Allows you to force a communications interval using the console interrupt.
FOPT	RETAIN= YES	This option permits the use of the RETAIN/370 function of OLTEP.
FOPT	RETAIN= NO	Default value.
FOPT	IT= YES	Will improve the performance of OLTEP in a multiprogramming environment.
FOPT	RELLDR= YES	Relocating load support (Forced when OLTEP= YES)

JCL TO INVOKE OLTEP

Statement	Comments
// JOB XXXX	Mandatory.
// ASSGN SYSnnn, X'cuu'	One ASSGN statement is necessary for each device tested or accessed by a test. None is required if the device was permanently assigned.
// UPSI 01	This statement is necessary if a console device is available but the test run definition is to be entered via the input job stream.
// EXEC IJZADOLT, REAL, [SIZE=NK]	Mandatory, OLTEP will run only in real. Size= parameter must be minimum of 16K or 20K if RETAIN is invoked. This will allow a 4K OLT to execute. If OLTs larger than 4K are to be run, the SIZE= parameter must specify a size equal to 12K plus the size of the OLT. The size specified must be multiple of 2K.
dev/test/opt/	This statement is included if the test-run definitions are entered via the input job stream.
/*	Mandatory.
/&	Mandatory.

Note: If SYSLOG is not a console, card input must be used.

TABLE OF OPTIONS

Option	Entry	Description
Testing Loop	TL(n) NTL	Recycle the test. If you specify a value (n), OLTEP runs the test the number of times indicated. If you do not specify a value, the test cycles 10 times. The maximum value allowed is 32,767 decimal. (See note 1)
Error Loop	EL(n) EL(I) NEL	Authorizes any error loop coded in the OLT to be executed the specified number of times. If you specify a value (n), the test loops the number of times indicated. If you do not specify a value, the test loops the number of times indicated in the preface of the OLT. If you specify the character I, a flag is set which indicates to the OLT, that it must loop indefinitely on the error. You can terminate the loop by specifying NEL following a request for communications. (See note 1)
Print	PR NPR	Print messages from the OLT. If you enter NPR, all messages originated by the OLT and normally designated for SYSLST are supressed. (See note 2)
Error Print	EP NEP	Print diagnostic error messages from the OLT. The FE option overrides NEP when a first error is encountered(once per section). (See note 3)
Control Print	CP NCP	Print OLT start and termination messages on SYSLST and SYSLOG.
Parallel Print	PP(n) NPP	Use the console device, in addition to SYSLST for OLT messages. Four levels of print are available on the parallel printer by entering one of these numbers at (n). 0: HEADER only 1: HEADER, DESCRIPTION and COMMENTS 2: HEADER and RESULTS 3: HEADER, DESCRIPTION, COMMENTS and RESULTS
First error Communications	FE NFE	Forces a communications interval when the first error is encountered. (See note 3). A message is printed indicating the test being run and the device being tested. This is followed by the 01E105D message that allows you to: . Change the device and/or test fields. . Continue the test by entering R01,'///' or R01,'//(Option change)/'. . Enter any OLTEP verb. . Cancel OLTEP by entering R01,'CANCEL'. There cannot be a first error communication if a console device is not available.
Manual Intervention	MI NMI	Informs the OLT section to run all manual intervention routines within the test request. (Manual Intervention and RE are mutually exclusive options)
Remote	RE NRE	RETAIN/370 only. If RE is specified, OLTEP is controlled by the remote specialist, who enters NRE to return control to the on-site customer engineer.

TABLE OF OPTIONS (....Continued)

Option	Entry	Description
Trace	TR NTR	Trace all functions called by OLT. (See note 4)
EXT=	EXT=	Information following this option is passed to the OLT section by way of a 56-byte buffer. This information must be the last entry in the option field and can contain any character but a slash. EXAMPLE: R01,'181/2400C/TL,EXT= BLOCK 4FFPRINT/' BLOCK 4FFPRINT goes into a buffer area within OLTEP and then passes on to the OLT section.

The default options are underlined.

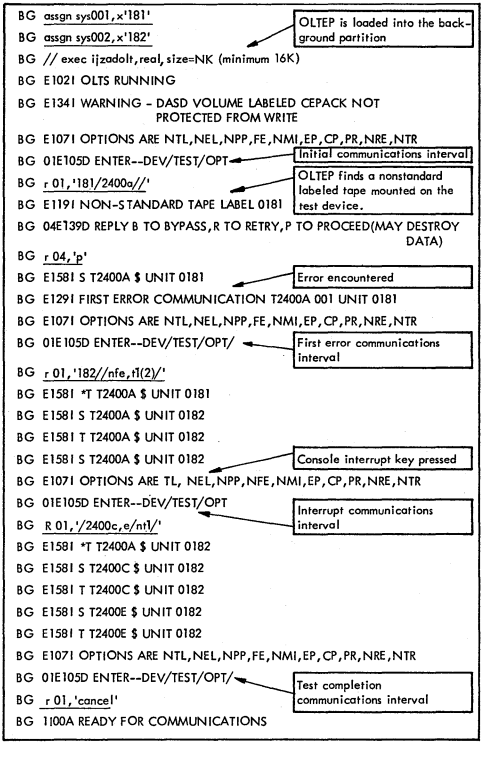
Note 1: The FE option overrides the TL and EL options, unless NPR is also in effect. However if you enter R 01,'///' or R 01,'///(option change)/' at a first error communications interval, the TL and EL options, if specified, are in effect.

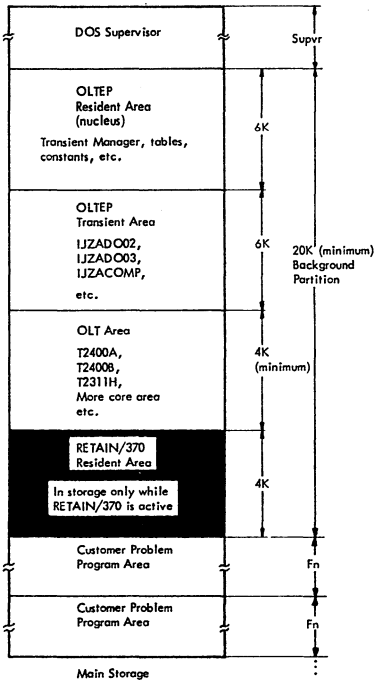
Note 2: NPR without EL and/or TL is ignored.

Note 3: Error print and FE are ignored if no print and either EL or TL are specified.

Note 4: Routine-to routine linkage is not traced. Do not attempt to use Trace function when SYSLST is assigned to the test device.

EXAMPLE OF OLTEP OPERATION





OLTEP Storage allocation with RETAIN/370 Active

DOS/VS EREP

DOS/VS ENVIRONMENTAL RECORDING, EDITING, AND PRINTING (EREP)

EREP is a program used by the customer or CE to print error statistics records that have been stored on the System Recorder File (SYSREC). When EREP is invoked, SYSREC prints out:

UC	Unit check records
MDR	Miscellaneous data records
CCH	Channel check handler
MCAR	Machine check analysis and recording records
TES	Tape error statistics
RDE	Reliability data extractor records
IPL/EOD	Initial program load/end of day records

EREP can perform any combination of the following options (refer to Figures V-2, V-3 and V-4 for details of the options):

- Edit/print the entire SYSREC file.
- Create or update the history/RDW tapes.
- Selectively retrieve records from the SYSREC file or history/RDE tapes for editing and printing.
- Summarize the SYSREC files.
- Create or update a TES history tape.
- Edit/print TES data from the SYSREC file.
- Summarize TES data from the SYSREC file or history tape.
- Clear the SYSREC file.

EREP is a self-relocating program and can be executed (// EXEC EREP) in any partition. Earlier editions of EREP required the program to be run in BACKGROUND only.

Executing DOS/VS EREP

When the configuration data is needed or the SYSREC file becomes full, EREP can be executed from SYSLOG or SYSRDR by:

```
// EXEC EREP
```

Then EREP issues a message to the operator via SYSLOG, requesting the logical unit (Figure V-1), either SYSLOG or SYSIPT, that is to be used for entering the EREP options. The operator must respond with one of the following:

C.followed by EOB or END for SYSIPT (system input reader).
S.followed by EOB or END for SYSLOG
N.followed by EOB or END, or END for the default option, EDIT.

DOS/VS EREP (...Cont'd)

LOGICAL UNIT	COMMENTS
SYSIPT	Optional
SYSLOG	Required, must be assigned to a 3210, 3215 or a Model 125 video display unit
SYSREC	Required
SYS007	Optional; must be assigned to a magnetic tape unit when a TES option is specified.
SYS008	Optional; required for TES options. Must be assigned to a magnetic tape unit. Used as a work tape.
SYS009	Optional; must be assigned to a magnetic tape unit for history/RDE options.

Figure V-1. Logical units required by EREP.

ROD Command

The ROD command can be submitted on the console typewriter or by a JCL card. The command is issued before the system is shut down, or at the end of the logical day, for systems that operate on a 24-hour basis. This command ensures that any statistical data held in main storage is added to the recorder file. The command also allows the end-of-day (EOD) record to be written if RDE is supported by the system.

Note: If the message, 3E13I (recorder file not opened), is issued, enter a // JOB statement to open the file.

EREP Options

When entering the EREP options via SYSIPT, column 1 must be blank and only one option for each card is allowed. SYSIPT input must be delimited by an end-of-data file card (/* in columns 1 and 2).

When entering the EREP options via SYSLOG, the entry must not exceed 80 positions:

1. Start with operation, OPTION.
2. Follow with a blank.
3. Follow with the option.
4. Follow with EOB or END.

Repeat this procedure for each option. When all the options have been specified, enter an EOB or END to continue processing.

Embedded blanks within the operation and the option are not allowed. A misspelled word, system error, duplicate option, or unsupported option can be corrected at SYSLOG by the operator. However, if they are not corrected, they are ignored. Multiple options are allowed by EREP.

It is recommended that the first EREP be run, using the option SUM. After reviewing the output, the SELECT option can be used to print specific data.

Figure V-2 shows a list of available EREP options.

When the EREP options are entered via SYSLOG, it is possible to execute the SUM and SELECT options more than once during an EREP run. After the SUM or SELECT function has been executed, the message, either:

3E03A ENTER SELECT PARAMETERS

or

3E05A ENTER SUMMARY PARAMETERS

is printed on SYSLOG. You may execute the SUM or SELECT function again by entering parameters at this time. If you wish to terminate the SUM or SELECT option, press END.

Figure V-3 is a condensed list of the EREP options and the parameters required if the SUM option is specified.

By means of specified search parameters, EREP selects records to be printed. The SELECT option initiates the search of these records on SYSREC, SELECT, TAPE causes a search of the history tape to be performed.

The parameters of the SELECT option are checked for validity, but not for logical relationship. For example, although an MCAR record has no VOL field, the following parameters are considered valid:

TYPE=MCAR
VOL=123456

The select parameters are listed in Figure V-4.

Any combination of parameters may be entered, the EREP program assumes that you will select only parameters that apply to the records wanted. If no select parameters are specified with the SELECT option the MCAR records are selected and printed.

An example of System/370 DOS/VS EREP is shown in Figure V-5. It is recommended that the SUM option be used. After reviewing the summary, use the SELECT option to print the desired records.

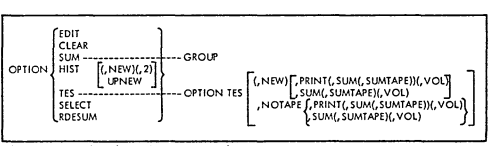
OPTION	RESULT
OPTION EDIT	Edits and prints SYSREC onto SYSLST.
OPTION CLEAR	1. Edits and prints SYSREC onto SYSLST. 2. Clears SYSREC.
OPTION SUM GROUP= {DISK, DISKETTE, TAPE, MICR/OCR, UNITREC, TP, CPU, 2715, ALL} CPU=xxxxxx, yyyy	Prints the summarization of SYSREC onto SYSLST. The file is summarized by the hardware group(s) listed in the GROUP parameter. If records from multiple CPUs appear on the SYSREC file, specify the serial number (xxxxxx) and model number (yyyy) of the CPU whose records you wish to have summarized. If CPU data is not supplied, records from all CPUs appearing on the SYSREC file ar summed together.
OPTION HIST,NEW(,2)	1. Creates the history /RDE tape on SYS009 (see note 1) 2. Clears SYSREC.
OPTION HIST(,2)	1. Updates the history/RDE tape on SYS009 (see note 1) 2. Clears SYSREC.
OPTION HIST, UPNEW	1. Updates the history/RDE tape on SYS009 (see note 1) 2. Creates a new tape file on SYS009 3. Clears SYSREC.
OPTION EDIT followed by OPTION HIST, NEW or OPTION HIST	1. Edits and prints SYSREC onto SYSLST 2. Creates or updates the history/RDE tape on SYS009 (see note 1) 3. Clears SYSREC.
OPTION TES,NEW	Creates a TES history tape on SYS007.
OPTION TES	Updates a TES history tape on SYS007.
OPTION TES, NOTAPE, PRINT	Edits and prints tape error data from SYSREC onto SYSLST. The data is printed in the detail tape unit format.
OPTION TES,PRINT,NEW	1. Creates a TES history tape on SYS007 2. Edits and prints tape error data from SYSREC onto SYSLST in the detail tape unit format.
OPTION TES, PRINT	1. Updates the TES history tape on SYS007 2. Edits and prints tape error data from SYSREC onto SYSLST in the detail tape unit format.
OPTION TES, NOTAPE, SUM	Prints the summarized tape data from SYSREC onto SYSLST in the detail tape unit format.
OPTION TES, NOTAPE, PRINT, SUM	1. Edits and prints the tape error data from SYSREC onto SYSLST in the deteil tape unit format. 2. Prints the summarization of the tape data from SYSREC onto SYSLST in the summarized tape unit format.
OPTION TES,SUM,VOL	1. Updates the TES history tape on SYS007 2. Summarizes the tape error data on SYSREC by volume serial number.

Figure V-2. EREP Options (Part 1 of 2)

OPTION TES,PRINT, VOL	1. Updates the TES history tape on SYS007 2. Edits and prints the tape error data from SYSREC onto SYSLST in the detail volume serial number format.
OPTION TES,PRINT, SUM,SUMTAPE,VOL	1. Updates the TES history tape on SYS007. 2. Edits and prints the tape error data from SYSREC onto SYSLST in the detail volume serial number format. 3. Summarizes the tape error data on the history tape and prints it on SYSLST in the summarized volume serial number format.
OPTION TES,NOTAPE, SUM,SUMTAPE	Summarizes the tape error data on the history file and prints it on SYSLST in the summarized tape unit format.
OPTION SELECT (see note 2)	Selectively prints records from SYSREC onto SYSLST.
OPTION SELECT,TAPE (see note 2)	Selectively prints records from the history/RDE tape onto SYSLIST (see note 1).
OPTION RDESUM	Summarizes the IPL, EOD, MCAR, CCH and Unit Check records for a specified period of from one to 30 days. These records are on the history/RDE tape (see note 3).
(none)	Edits and prints SYSREC onto SYSLST.

Notes:
1. RDE is only available if ERLOG = RDE is specified at generation time.
2. Records are selected by specifying select parameters.
3. RDESUM does not summarize across multiple volumes. If EOF is encountered before the entire requested reporting period has been covered (this can be checked through the end data printed on the RDESUM listing), rerun RDESUM using the next volume history/RDE file and the same reporting period you specified during the first RDESUM execution. A listing with the reminder of the requested information is thus generated.

Figure V-2. The EREP Options (Part 2 of 2)



The options for TES (Tape Error Statistics)

GROUP=	DISK DISKETTE TAPE MICR/OCR UNITREC TP CPU 2715 ALL	CPU=xxxxxx,yyyy where xxxxxx = the CPU volume serial number yyyy = the CPU model number.

Figure V-3. Parameters for the SUM option.

SELECT PARAMETER	RESULT
CPU=xxxxxx	All error records associated with a CPU may be selected for printing by entering the six digit CPU serial number.
TYPE = {MCAR, CCH, IPL, EOD, TP, UNIT, 2715}	A specific type of error record may be selected for printing. Any number of different types, separated by commas, may be selected for each search. If TYPE=UNIT is selected, all error records except volume dismount records are handled.
DATE= {yyddd,yyddd / yyddd}	All recordings made within a time span (measured in days) may be selected for printing. If two dates, separated by a comma, are specified, all recordings made in that time span are selected. If only one date is specified, all recordings made on that day are selected for printing.
TIME hhmm,hhmm	All recordings made within a time span (measured in hours and minutes) may be selected for printing.
JOB=xxxxxxxx	All recordings made during the execution of a specific job may be selected for printing by specifying the eight-byte jobname from the job statement.
VOL=xxxxxx	The error records for a specific volume may be selected for printing by entering the six-byte volume serial number.
TERM=xxxxxxxx	The error records for a terminal may be selected by entering the eight-byte terminal name.
CUA=xxxx	Records may be selected for printing by entering the channel and unit address (in hexadecimal) or the line number for TP.CUA=normalized address.
DEVICE=xxxxxx	The records associated with a specific type of device may be selected by entering the device type code (for example, 1403,1442N1). To retrieve records associated with 3340 model 70 enter 334A. To retrieve records associated with 3330 Model 11 enter 3330B.
FORMAT=TES	Whenever a tape (2400 or 3400-series) error record is encountered, it is printed in the detail TES format by volume serial number. If FORMAT=TES is not specified, all tape error records are printed in the unit check format. All error records except volume dismount records are handled.
SEL2715= {AREA, ADAPTER, SPECIAL}	The 2715 records are printed in area station format if the SEL 2715 parameter is not specified. If printing by area, adapter, or special is required, however, the SEL2715 parameter must be specified.

Figure V-4 EREP Select Parameter

```
BG      1100A  READY FOR COMMUNICATION
BG      //EXEC EREP(B)
BG      3E11D  ENTER OPTION SOURCE, C=CARD, S=CONSOLE, N=NONE
BG      S (B)
BG      3E14A  ENTER OPTION
BG      Option sum (B)
BG      Option clear (B)
        (B)
BG      READY FOR COMMUNICATIONS
[ ]     =      OPERATOR responses or command
(B)     =      EOB
```

Figure V-5 Example of Erep Options Using Syslog.

Creating RDE/TES History Tapes

History/RDE and TES history tapes are created and updated using EREP options HIST and TES.

Entering options via SYSIPT: When entering the EREP options via SYSIPT, column 1 must be blank and only one option per card is allowed (for example, HIST with UPNEW or with NEW and/or 2 is considered one option). Each option may only be entered once for each execution of the EREP program.

Example job streams for executing EREP:

```
// JOB EXAMPLE1
// TLBL EREPNEW
// TLBL TAPEIN
// ASSGN SYS007,X'cuu'
// ASSGN SYS008,X'cuu'
// ASSGN SYS009,X'cuu'
// LBLTYP TAPE
// EXEC EREP
OPTION HIST,NEW
OPTION TES,NEW
/*
/&
// JOB EXAMPLE2
// TLBL TESUP
// TLBL EREPUP
// ASSGN SYS007,X'cuu'
// ASSGN SYS008,X'cuu'
// ASSGN SYS009,X'cuu'
// LBLTYP TAPE
// EXEC EREP
OPTION EDIT
OPTION TES
OPTION HIST
/*
/&
```

Figure V-6. Sample to Create History Tape

EREPNEW and EREPUP must be the filenames for new history files or for updating. TAPEIN and TESUP must be the file names for a new TES history tape or an update TES history tape.

MAINTAIN SYSTEM HISTORY UTILITY

1.0 DESCRIPTION

MSH utility is a program to select, deblock, and apply PTFs from a PTF file. The program creates a System History, simplifies installation and reduces the time required to remove a PTF, which proves to be ineffective or unstable. For this purpose the program provides backup records and automatically updates the PTF history which is kept in the system in form of a book in the source statement library.

This program runs under control of DOS/VS Release 33 and upwards. This includes application in a cardless environment. The program requires a virtual partition of 64K Bytes.

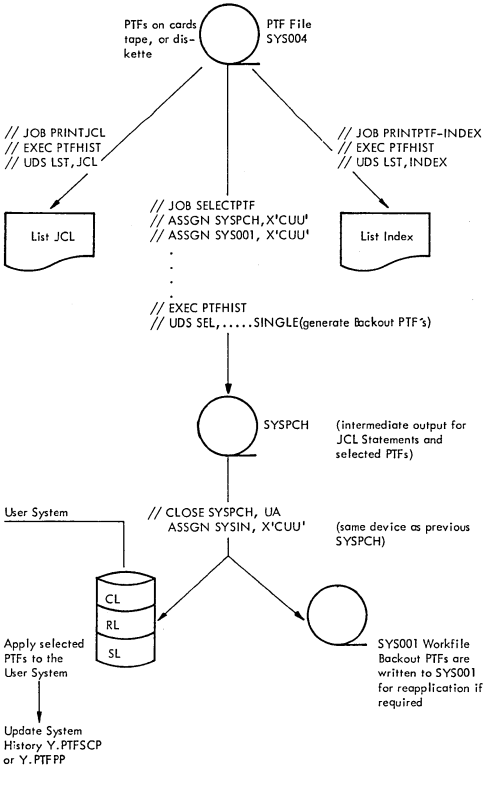
1.1 Functions

- Selects PTFs from a PTF file as specified in select control statements (if the input file is blocked, deblocking is done automatically).
- Generates Job Control Statements to punch a backout PTF (statements which can be used to remove the PTF at a later time, if this should be necessary).
- Generates Job Control Statements to update the System History.
- Prints an index of a PTF file, or all Job Control Statements within a PTF file.

The program can also be used for the installation of some Independent Distributed Components (IRs) and Program Products (PPs). The description of the control statements in this chapter includes the parameters needed to process such IRs and PPs. However, for information on how to use the program for IR and PP installation, refer to the individual IR or PP.

FOR DETAILED INFORMATION, PLEASE REFER TO DOS/VS SYSTEM UTILITIES MANUAL GC33-5381-2.

FUNCTIONAL OVERVIEW



2.0 SYSTEM HISTORY

The system history is kept in the form of two books in the source statement library.

- The SCP History Y.PTFSCP.
- The Program Product History Y.PTFPP.

The following table shows the history format:

```
           BKEND     Y.PTFSCP

*********  PTFS AND  LOCAL FIXES APPLIED TO DOS/VS REL NN.N
*********  ------------------------------------------------------------
*********  CUSTOMER -----------------------EVA CORPORATION
*********  ADDRESS ------------------------2 MAIN STREET, ANYWHERE
*********  PHONE NO ---------------------- 017-723-977
*********  SYSTEM PROGRAMMER ---------- TOM SMITH
*********  ENVIRONMENT-----------------
*********  ------------------------------------------------------------
*********  TO ADD A NEW ENTRY USE FOLLOWING CONTROL STATEMENTS:
*********              // EXEC MAINT
*********              UPDATE Y.PTFSCP
*********              ) ADD 0024
*********              NEW ENTRY
*********              ) END
*********              /*
*********              /&
*********  ------------------------------------------------------------
*********
*********  IDENTIFIERS IN COLUMN ONE ARE AS FOLLOWS:
*********  P=PTF, A=APAR FIX, L=LOCAL FIX, I=INSTALLATION FIX, C=COMMENT
*********  B=BACKOUT PTF
*********
*********
* 0         1                                            7
* 3 ---------2 ------------------------------------------ 1
B N04074    5745TAP                                21/01/76
B           E05335 E05908
B           E.DTFMT
P N04089    574510X AR=NNN                         21/01/76
P           E04348 E04906 E05372 E05630 E05786
P           C-$$BOPEN C-$$BOPEN0  C-$$BOPEN1
P           C-$$BOPEN 3 C-$$BOPIGN C-$$BOSD01
P N04074    5745TAP                                19/01/76
P           E05335 E05908
P           E.DTFMT
           BKEND
```

Each new entry in the history is inserted at the top of the history book, and the previous entries are shifted downwards.

Note 1 : The Identifier I has to be used as installation record for ICRs (Independent Component Release). The entry will be automatically at ICR installations or has to be done manually by the CE/FE.

```
Example :        Power VS ICR Installation
Col.1  .....12...
    I  bN00000bbb5745PWRbRel=nn
                    PWR ICR installed      date
```

2.1 History Macro Description

```
HIST REL = 'RELEASE', CUST = 'NAME'
      [,ADDR='ADDRESS']  [,PHONE='NUMBER']
      [,PROG='PROGRAMMER']  [,ENV=ENVIRONMENT']
      [,SCPHIST=SYSTEM/PRIVATE]
      [,PPHIST=PRIVATE/SYSTEM]
```

2.2 History Generation Example

```
// Job MSH Installation
// Option Deck
// Exec Assembly
      Hist  Rel= '33.0'
            Cust= 'EVA CORPORATION'
            Addr = '2 MAIN STREET, ANYWHERE'
            Phone= '017-723-977'.
            Prog= 'TOM SMITH',
            PPHIST=SYSTEM
      END
/*
/&
```

SYSPCH output of the above assembly is a job stream to initialize the history book and link the maintain system history utility.

3.0 THE UDS CONTROL STATEMENT

This statement has the format :

```
// UDS SEL, [ (ww,'xxxxxxxx',yy,zz) ]  , [ SINGLE/MASS/MASSB]
```

// UDS identifies the statement as a UDS statement

SEL requests the PTF select function

ww,'xxxxxxxx' — These two operations describe the fixed part of the select argument, ww specifies the starting position of the fixed part of the argument within the statement in the PTF file, and xxxxxxxx is the character representation of the fixed part of the argument.
The length of the fixed part may be 20 characters. Any apostrophes within the character string must be coded as double apostrophes.

yy,zz — These two operands specify the starting and ending positions of the variable part of the argument within the statement in the PTF file. The contents of the field delimited by these positions are compared with the parameter "nnnnnn" in the select statements. The length of this variable part may be up to 8 characters.

SINGLE
MASS
MASSB
SUM — This parameter controls the BACKOUT-PTF creation and history updating.

If SINGLE is specified, statements for a BACKOUT-PTF are generated for each selected PTF. The history is updated at the end of each PTF job.

If MASS is specified, the creation of BACKOUT-PTFs is suppressed and the history entries are collected in a table. A history update job is generated at the end of the selection run or between two PTFs if the table becomes full.

If MASSB is specified, statements for a BACKOUT-PTF are generated for each selected PTF. A history update job is generated at the end of selection run or between two PTFs if the table becomes full.

SUM is used only for IR and PP installation.

Name — used only for IR and PP installation.

The specification of a UDS statement without parameters (// UDS SEL) results in the generation of the "Standard" default :

```
// UDS SEL,(1,'//JOB N',9,13),SINGLE
```

which is suitable for selection of PTF statements in the form:

```
// JOB Nxxxxx
```

where xxxxx is a five figure identification number.

4.0 THE SELECT STATEMENT

This statement has the following format :

nnnnnn [,C=kkkkkkk]

nnnnnn — The variable part of the select argument for the PTF to be selected. The length of this parameter must match the length implicitely specified by the parameters yy and zz in the UDS statement.

C=kkkkkkk — This parameter specifies the component which is affected by this PTF. This parameter is optional, but it is transferred to the history if it is specified. Its length must be seven bytes (for example, 5745UTL for component 5745-SC-UTL), and a parameter with an incorrect length is ignored.

CL=SIP
RL=SIP
SL=SIP — Used only for IR and PP Installation.

5.0 THE COMMENT STATEMENT

The comment statement is read in from SYSIPT and merged into the selected PTF. It is also treated as a JCL comment within the PTF jobstream. A comment may also appear within the PTF itself. A maximum of 20 comment statements can be processed for one PTF. The statement has the following format :

* 'nnnnnn' [ttt...tt]

An asterisk in column 1 followed by at least one blank identifies a comment statement.

'nnnnnn' — Identifies the PTF into which the comment is to be inserted. This parameter may be up to eight characters long. If this parameter is shorter than the PTF name, comparison with the PTF name is made only with the length of this parameter. The comment is inserted into the first PTF which matches these characters.

ttt...tt — Up to 60 text characters, seperated from the preceding parameter by one or more blanks.

6.0 THE LIST FUNCTION (LST)

The list function permits the user to print a PTF file index or all the job statements contained in a PTF file.

LIST INDEX

The list index function is requested with the statement :

// UDS LST,INDEX

The Control Statement is read from SYSIPT and the PTF File is read from SYS004, which may be a disk, a tape, or a diskette. The input record length may be 80 or 81 bytes, or the records may be blocked into 3440-byte blocks.

The output of the function is printed on SYSLST, and contains the following information :

a) the control statement read,
b the PTF names, their card counts, and the word "BACKOUT".

If this is a BACKOUT-PTF (BACKOUT-PTFs produced by previous PTF application runs may exist in the user environment).

LIST JCL STATEMENTS

A list of all job control statements is requested with the statement :

```
// UDS LST,JCL
```

A list of all job control statements found in the file (including CATALR, CATALS, CATALP, INCLUDE, PHASE, COPYC, COPYR, and COPYS statements) is printed on SYSLST.

SAMPLE JOBS USING THE LIST FUNCTION

The following I/O device assignments are required for the LIST function :

SYSIPT	Control statement input
SYSLST	System output
SYS004	PTF input file (filename = PTF)
or	
SYSSLB	IR or PP installation history input

SAMPLE INDEX JOB

```
// JOB PRINT PTF-INDEX
// ASSGN SYS004, X'281'          PTF-TAPE
// EXEC PTFHIST
// UDS LST, INDEX
/*
/&
```

SAMPLE JCL LIST JOB

```
// JOB PRINT JOB-CONTROL-CARDS
// ASSGN SYS004,DISK,VOL=DOS330
// DLBL PTF,.....
// EXTENT SYS004,DOS330
// EXEC PTFHIST
// UDS LST,JCL
/*
/&
```

7.0 THE SELECT FUNCTION (SEL)

Selection is controlled by a UDS control statement, which is followed by the appropriate PTF selection statements.
These may be in any order, but they must follow the UDS statement. Up to 100 select statements may follow the UDS control statement.

The following I/O device assignments are required for running the select function.

SYSLOG	Accounting information input Message output
SYSIPT	UDS control statement PTF select statements (Filename IJSYSIN) Comment statements
SYS004	PTF input (filename=PTF) Devices supported : card reader diskette tape DASD (2314,3330, 3340, 3350) Record Length : 80,81 or 3340 bytes
SYSSLB	used only for IR and PP installation
SYS001	Workfile (filename=IJSYS01) Devices supported : tape DASD (2314,3330, 3340, 3350) Note : This is required only if BACKOUT-PTFs are to be generated.
SYSPCH	Jobstream output (filename = IJSYSPH)
SYSLST	System output (filename = IJSYSLS)

Notes :

If the input is on a multi-file tape, the tape must first be positioned to the correct file. The MTC command may be used for this.

User job accounting information is requested via SYSLOG at the start of the program. This information, which has a maximum length of 56 bytes is transferred to the JOB statement(s), starting at column 17.

The system input is printed on SYSLST as it is read. During selection, the PTF ID is printed together with its card count (or "NOT FOUND" if the PTF could not be found). A summary at the end of the program shows how many PTFs were selected.

To conclude processing, SYSPCH must be closed. The output on SYSPCH is in a form suitable for input via SYSIN for application of the PTFs to the system.

SAMPLE JOB USING SELECT FUNCTION

```
// JOB SELECT
// ASSGN SYS004, TAPE
// ASSGN SYSPCH, TAPE
// ASSGN SYS001, DISK,VOL=111111
// DLBL IJSYS01,.....
// EXTENT SYS001,.....
// EXEC PTFHIST
// UDS SEL, (1,'//JOB N',9,13)
04054,C=5745RMS
04076
/*
// CLOSE SYSPCH,UA
/&
```

OUTPUT ON SYSPCH

```
// JOB N04054*3
// OPTION NOLOG
// EXEC PTFREPRO
*// JOB N04054*3
** CAUTION  THIS BACKOUT-PTF REMOVES THE APAR'S LISTED BELOW
        -
        -
        -
        JCL For BACKOUT-PTF generation
        -
        -
* * *  N04054 3   * * *
* APPLICABLE REL..
        -
        -
        -
        PTF fetched from input tape
        -
        -
RLD
END
/*
// EXEC LNKEDT
// ASSGN SYSSLB,UA
        -
        -
        -
        JCL for history update step
        -
        -
) END
/*
/&
```

8.0 JOB CONTROL EXAMPLE FOR A DISK OR DISKETTE ORIENTED SYSTEM

```
// JOB SELECT PTF FOR APPLICATION
// DLBL IJSYSPH, ....                          Note 1
// EXTENT SYSPCH,....
// ASSGN SYSPCH,X'cuu'OUTPUT PTF FOR APPL.
// DLBL PTF,....                               Note 2
// EXTENT SYS004,....
// ASSGN SYS004,X'cuu' INPUT PTF
// EXEC PTFHIST
   - control statements for selection -
/*
CLOSE SYSPCH, UA
/&
// OPTION PARSTD
// DLBL IJSYSIN,....                           Note 3
// EXTENT SYSIN,....
   - assgn (permanently) private libraries if desired -
ASSGN SYSIN,X'cuu'                  SAME AS SYSPCH IN PREV.JOB
   - SYSIN-job stream will apply the PTF to the system -
```

Note 1

DLBL, EXTENT-cards are required (if not available, as STDLABEL) in case of SYSPCH to disk or diskette. For a single volume diskette file specify.

```
// EXTENT SYSPCH.
```

For a multi-volume diskette file code one EXTENT card per diskette volume.

For example for a 3 volume diskette file :

```
// DLBL IJSYSPH,....
// EXTENT SYSPCH
// EXTENT SYSPCH
// EXTENT SYSPCH
```

The file ID in the DLBL card for SYSPCH on diskette should be DTTEPTF, i.e.

```
// DLBL IJSYSPH, 'DTTEPTF'.
```

In this case the SYSIN-Labels for diskette can be omitted, if they have been added correctly to the STDLABEL-Set or if IBM supplied STDLABEL are used (see SYSGEN-Book, GC33-5377).

Note 2

Same as in Note 1 for SYS004 instead of SYSPCH. Remember, the filename in the DLBL Card is PTF in this case.

Note 3

Same as in Note 1 for SYSIN instead of SYSPCH. Remember, the filename in the DLBL card is IJSYSIN in this case. The DLBL and EXTENT information (same as for SYSPCH of previous job) has to be made PARSTD which means that all existing PARSTD-Label for that partition will disappear.

The Job Control Commands

```
// OPTION PARSTD
// DLBL IJSYSIN,....
// EXTENT SYSIN,....
```

can be omitted for input from diskette if they have been added correctly to the STDLABEL-Set (for cardless system) or if IBM supplied STDLABEL are used (see note 1).

8.1 Job Control Examples

If a BACKOUT PTF must be created during a select run, SYS001 is required as a workfile.
DLBL-,EXTENT,-AND ASSIGN- statements for IJSYS01 can be omitted if a standard assignment exists and the label information is stored on the standard label cylinder.

```
// JOB BACKOUT
// ASSGN SYS001,X'161'
// DLBL IJSYS01,'WORK',70/001/SD
// EXTENT SYS001,.......
// ASSGN SYSPCH, X'280'
ASSGN SYSIN,UA
   ENTER FROM CONSOLE...
ASSGN SYS004,X'00C'
// EXEC PTFHIST
// CLOSE SYSPCH,X'00D   BACKOUT PTF OUTPUT
ASSGN SYSIN,X'280'
/&
```

EXAMPLE FOR CARD INCORPORATION. THE PTF IS READ BY THE CARD READER, STORED ON DISK AND THE JOBSTREAM IS PUNCHED ON TAPE. THE ENTIRE INPUT IS PROCESSED BECAUSE SELECT CARDS ARE MISSING.

8.2

```
// JOB MASS APPLICATION
// DLBL IJSYSPCH,.....
// EXTENT SYSPCH,.....
// ASSGN SYS004,X'162'
// DLBL PTF,.....
// EXTENT SYS004,..
// ASSGN SYSPCH,X'162'
// EXEC PTFHIST
// UCS SEL,1,'// JOB N',9,13,MASS
52511
53008
59084
53016
* 'N52511' APAR E52784 NOT FIXED
* 'N53008A3' C=5745IOX
* 'N59084' C=6745 PWR FIX FOR APAR E55186 INCLUDED
/*
// CLOSE SYSPCH.UA
/&
// DLBL IJSYSIN,.....
// EXTENT
ASSGN SYSIN,X'162'
```

MASS APPLICATION OF PTFs N52511, N53008, N53016, AND N 59084. FOR MASS APPLICATION, GENERATION OF BACKOUT PTF IS SUPPRESSED, THEREFORE WORKFILE SYS001 IS NOT REQUIRED. ADDITIONAL INFORMATION IS SUPPLIED FOR THREE DIFFERENT PTFs.

8.3

```
// JOB BACKOUT PTF
// ASSGN SYS001,X'280'
// ASSGN SYSPCH,X'281'
// ASSGN SYS004,X'282'
// EXEC PTFHIST
// UDS SEL,1,'// JOB N',9,13
59021 C=5745PWR
59035
* 'N59035' C=5745PWR LOCAL FIX INCLUDED
/*
// MTC WTM, SYSPCH
// MTC REW, SYSPCH
/&
// ASSGN SYSPCH,TAPE                BACKOUT PTF OUTPUT
ASSGN SYSIN,X'281'
```

EXAMPLE FOR A TAPE ORIENTED SYSTEM.
PTFs 59021 AND 59035 ARE SELECTED AND A COMMENT FOR PTF 59035 IS SUPPLIED.

REAPPLICATION OF BACKOUT PTFs

For re-application of a BACKOUT PTF the BACKOUT PTF File is processed in the same manner as the normal PTF-File by using the UDS and the select statements.

LIST SYSTEM HISTORY (HISTLIST)

1.0 Description

The List System History utility can be executed in any real or virtual partition. The storage requirements depend on the number of entries in the history book to be processed, because the program requires sufficient space for the complete history book and additional working storage. The minimum virtual partition size of 64K is sufficient for processing a book with approximately 1200 entries. If the book to be processed is larger, the partition size will have to be increased accordingly.

The List System History utility produces its output in three parts: the book list, the cross-reference lists, and the lost APAR and error report. The program allows you to request specific parts of the output as well as the complete output (see the section Program Execution)

1.1 Functions

- To provide a complete printout of either of the history books Y.PTFSCP and Y.PTFPP, which are generated and maintained by the utility program Maintain System History (PTFHIST), or of any other history book with the same format.

- To provide edited and sorted cross-reference lists of APARs, local fixes, PTFs, and affected library members, with pointers to the entries in the book printout.

- To provide an edited list of lost APARs and an error report.

For detailed information, please refer to DOS/VS System Utilities Manual GG33-5381.

2.0 BOOK LIST

The BOOK LIST is a complete listing of the selected HISTORY BOOK with its entries in the original sequence.
The following table shows the history format:

```
********* PTF'S AND LOCAL FIXES APPLIED TO DOS/VS REL 32.0        00000000
********* ----------------------------------------------------- 00000001
********* CUSTOMER --------------- EVA CORPORATION               00000002
********* ADDRESS ----------------- 2 MAIN STREET, ANYWHERE      00000003
********* PHONE NO -------------- 017-723-977                    00000004
********* SYSTEM PROGRAMMER ---- TOM SMITH                       00000005
********* ENVIRONMENT ----------                                 00000006
********* ----------------------------------------------------- 00000007
********* TO ADD A NEW ENTRY USE FOLLOWING
********* CONTROL STATEMENTS:                                    00000008
*********             // EXEC MAINT                              00000009
*********             UPDATE Y.PTFSCP                            00000010
*********             ) ADD 0024                                 00000011
*********             NEW ENTRY                                  00000012
*********             ) END                                      00000013
*********             /*                                         00000014
*********             /&                                         00000015
*********                                                        00000016
********* ----------------------------------------------------- 00000017
*********                                                        00000018
********* IDENTIFIERS IN COLUMN ONE ARE AS FOLLOWS:              00000019
********* P=PTF, A=APAR FIX, L=LOCAL FIX, I=INSTALLATION FIX,    00000020
********* C=COMMENT, B=BACKOUT PTF                               00000021
*********                                                        00000022
* 0        1                                                   7 00000023
* 3 --------2 ------------------------------------------------ 1 00000024
B N10099:1  5745PDA AR=320                              09/22/76 00000025
B           E10001                                               00000026
B           B-IJBDMPGN                                           00000027
P N10084:B  5745BTM AR=310,320 SUP=N10083,N10081,N10080
            PRE=N10261,                                        * 00000028
P           N10158                                      09/27/76 00000029
P           E06010 E06061 E06062 E06082 E06087 E06102 E06110 E06114 00000030
P           E06132 E07210 E07211 E07212 E07217 E07218 E07219 E07220 00000031
P           E07221                                               00000032
P           E.BTMODIH2 E.BTCSE E.BTBTFIX E.DTFBT C-$$ABERP5
            C-$$ABERP6                                           00000033
P           C-$$ABERP7 C-$$ABERP8 C-$$BCTC01 C$$BOTC02           00000034
P N10099:1  5745PDA AR=320                              09/22/76 00000035
P           E10001                                               00000036
P           R-LJBDMPGN                                           00000037
P N11190:3  5745UTL AR=330                              09/22/76 00000038
P           E11313                                               00000039
P           Z.FASTCOPY                                           00000040
1           5745SUP  ***  FT TRACE INSTALLED  ***       09/22/76 00000041
1           E.SGDFCH                                             00000042
```

3.0. CROSS-REFERENCE LISTS

This part of the output is produced when you specify the XREF function. It is also produced for book Y.PTFSCP when the program is executed without console communication.

Fix/PTF Cross-Reference

This is a cross-reference list of all APAR fixes, local fixes, and PTFs installed on the system. The list has four columns with the following meanings:

FIX
Installed APAR, local fix, or PTF. APARs and PTFs are identified by APAR and PTF numbers, respectively. A local fix is identified by its entry sequence number.

COMP
Component identifier of the APAR fix or PTF.

LAST INST.DATE
Last installation date of the APAR fix or PTF.

ENTRIES
The sequence numbers of the corresponding history book entries, as an aid to finding these entries in the first part of the program output.

APAR Cross-Reference

This is a sorted list of all installed APARs. The list has four columns with the following meanings:

APAR
The numbers of all APARs for which PTFs or APAR fixes are installed.

COMP
The component identifier of the affected component.

LAST FIX
Either the number of the last PTF containing the APAR or the indication 'A-FIX' for APAR fix.

ENTRIES
The sequence numbers of the corresponding history book entries as an aid to finding these entries in the first part of the output.

3.0. CROSS-REFERENCE LISTS

Module Cross Reference

A sorted list of all affected library members. The list is produced for the XREF function and has four columns with the following meanings:

MODULE
The names of all library members affected by APAR fixes, local fixes, of PTFs. Names of phases are identified by the prefix 'C-'. Names of modules have the prefix 'R-', for names of books the sublibrary identifier is used (for example, E.PIOCS).

COMP
The identifier of the affected component.

LAST CHNG. DATE
The date of the last change installed by means of a fix or PTF.

FIXES
This column contains one of the following:

- The PTF number
- If no PTF is installed, the APAR number of the fix
- If the fix has no APAR number, the sequence number of the history book entry.

The following sample shows the format of the three cross-reference lists.

3.0 CROSS-REFERENCE LISTS

*** CROSS REFERENCE OF ALL INSTALLED PTF'S ***

FIX	COMP	LAST INST.DATE	ENTRY SEQUENCE NUMBERS
E06011	5745BTM	08/13/76	0059
E07220	5745BTM	09/22/76	0043
* INST. FIXES W/O APAR NO. *			0041
* LOCAL FIXES W/O APAR NO. *			0061 0062
N10081:A	5745BTM	09/22/76	S0050
N10083:3	5745BTM		S0045
N10084:B	5745BTM	09/27/76	0028
N10099:1	5745PDA	09/22/76	B0025 B0035
N10158:2	5745SUP	09/22/76	0054
N11190:3	5745UTL	09/22/76	0038

TOTAL FIX COUNT 00010

A 'B' IN FRONT OF A SEQUENCE NUMBER INDICATES A BACKOUT PTF.
A 'S' INDICATES THIS PTF HAS BEEN SUPERSEDED.

*** CROSS REFERENCE OF ALL APAR NUMBERS ***

APAR	COMP	LAST FIX	ENTRY SEQUENCE NUMBERS
E06010	5745BTM	N10084:B	0030 S0046
E06011	5745BTM	A-FIX	0059
E06061	5745BTM	N10084:B	0030 S0046
E06062	5745BTM	N10084:B	0030 S0046
E06082	5745BTM	N10084:B	0030 S0046
E06087	5745BTM	N10084:B	0030 S0046
E06102	5745BTM	N10084:B	0030 S0046
E06110	5745BTM	N10084:B	0030 S0046
E06114	5745BTM	N10084:B	0030 S0046
E06132	5745BTM	N10084:B	0031 S0047
E07210	5745BTM	N10084:B	0031 S0047
E07211	5745BTM	N10084:B	0031 S0047
E07212	5745BTM	N10084:B	0031 S0047
E07217	5745BTM	N10084:B	0031 S0047
E07218	5745BTM	N10084:B	0031 S0047
E07219	5745BTM	N10084:B	0031
E07220	5745BTM	N10084:B	0031 0043
E07221	5745BTM	N10084:B	0032
E09050	5745SUP	N10158:2	0057
E09171	5745SUP	N10158:2	0057
E09172	5745SUP	N10158:2	0057
E10001	5745PDA	N10099:1	B0026 B0036
E10123	5745SUP	N10158:2	0057
E10134	5745SUP	N10158:2	0057
E10155	5745SUP	N10158:2	0057
E10159	5745SUP	N10158:2	0057
E10161	5745SUP	N10158:2	0057
E10165	5745SUP	N10158:2	0058
E10170	5745SUP	N10158:2	0058
E11313	5745UTL	N11190:3	0039

TOTAL FIX COUNT 0030

A 'B' IN FRONT OF A SEQUENCE NUMBER INDICATES A BACKOUT PTF.
A 'S' INDICATES THIS PTF HAS BEEN SUPERSEDED.

3.0 CROSS-REFERENCE LISTS

*** CROSS REFERENCE OF ALL CHANGED MODULES ***

MODULE	COMP	LAST CHNG.DATE	FIXES OR ENTRY SEQUENCE NO'S
C-$$ABERP5	5745BTM	09/27/76	N10084:B N10083:3
C-$$ABERP6	5745BTM	09/27/76	N10084:B N10083:3
C-$$ABERP7	5745BTM	09/27/76	N10084:B N10083:3
C-$$ABERP8	5745BTM	09/27/76	N10084:B N10083:3
C-$$ABERP9	5745BTM	09/22/76	N10081:A
C-$$BCTC01	5745BTM	09/27/76	N10084:B N10083:3 N10081:A
C-$$BOTC02	5745BTM	09/27/76	N10084:B
E.ALLOC	5745SUP	09/22/76	N10158:2
E.ALLOCR	5745SUP	09/22/76	N10158:2
E.BTBTFIX	5745BTM	09/27/76	N10084:B N10083:3
B.BTCSE	5745BTM	09/27/76	N10084:B N10083:3
E.BTMOD	5745BTM	09/22/76	N10081:A
E.BTMODIH2	5745BTM	09/27/76	N10084:B N10083:3 N10081:A
E.BTNCKID	5745BTM	09/22/76	N10081:A
E.BTOMLOA	5745BTM	09/22/76	N10081:A
E.DTFBT	5745BTM	09/22/76	N10084:B E07220 N10083:3 N10081:A E06011
E.DVCGEN	5745SUP	09/22/76	N10158:2
E.FOPT	5745SUP	09/22/76	N10158:2
E.TOINTER	5745SUP	09/22/76	N10158:2
E.SGCCWT	5745SUP	09/22/76	N10158:2
E.SGDFCH	5745SUP	09/22/76	**0042**
E.SGEND	5745SUP	09/22/76	N10158:2
E.SGMAIN	5745SUP	09/22/76	N10158:2
E.SGSUP	5745SUP	09/22/76	N10158:2
R-IJL03Z	5745BTM	09/22/76	N10081:A
Z.FASTCOPY	5745UTL	09/22/76	N11190:3

TOTAL COUNT OF CHANGED MODULES 00026

4.0 LOST APAR and ERROR REPORT

This report provides a summary of all PTFs for which one of the following conditions was detected:

- Incorrect syntax or format,
- Prerequisite PTF(s) missing.

It also provides a list of lost APARs. These are APAR or local fixes which were lost due to installation of PTFs.

This part of the output is produced when you specify the CHECK funetion. It is also produced for book Y.PTFSCP when the program is executed without console communication.

The following sample shows the format of this report.

4.0 LOST APAR and ERROR REPORT

****** LOST APAR AND ERROR REPORT ******

PTF	COMP	DATE	PHASE MODULE	REFERENCED ENTRY OF LOCAL-FIX OR APAR	DATE	SEQ.NO
N10084:B	5745BTM	09/27/76		***THIS PTF HAS NO PREREQUSITE		
				PTF N10262 INSTALLED		0028
			E.DTFBT	A E06011 5745BTM		
				LOOP IN BTBTFIX	08/13/76	0059
			E.DTFBT	L 5745BTM CICS ABEND		
				WITH PROGRAM CHECK	08/13/76	0061
			E.DTFBT	L BTBTFIX		0062
N11190:3	5745UTL	09/22/76		***THIS PTF IS NOT APPLICABLE TO THE INSTALLED RELEASE		0038
N10061:A	5745BTM	09/22/76		***SYNTAX ERROR ON PTF, ENTRIES BYPASSED		0050

NOTE: LOCAL AND INSTALLATION FIXES OR ENTRIES WITH PRE APAR NO'S WILL BE DISPLAYED IF THE APAR NO. IS MISSING.

*** SYNTAX ERRORS FOUND ON ENTRIES 0051 0053 0061 0062

5.0 JOB CONTROL STATEMENTS

The HITLIST utility runs in a batch partition of an operational DOS/VS system and is controlled by the following job control statements:

// JOB jobname	Required.
// UPSI	
1	Optional. Indicates that console communication is desired.
01	Optional. Suppresses listing of the history book.
// ASSGN SYSSLB,X'cuu'	Required if the requested book is stored in a private source statement library.
// EXEC HISTLIST	Required.
	Note: It is assumed that SYSLOG, SYSLST, and SYSRES are already asigned.

INDEX

INDEX (continued)

C (continued)

D

D (continued)

INDEX (continued)

J

L

M (continued)

P (continued)

Q

R

R (continued)

S (continued)

T

T (continued)

U

V

W

X

Y

Z

THIS PAGE HAS BEEN ADDED FOR YOUR OWN NOTES

THIS PAGE HAS BEEN ADDED FOR YOUR OWN NOTES

THIS PAGE HAS BEEN ADDED FOR YOUR OWN NOTES

THIS PAGE HAS BEEN ADDED FOR YOUR OWN NOTES

THIS PAGE HAS BEEN ADDED FOR YOUR OWN NOTES

SY33-8571-6

DOS/VS Handbook Volume 1 (S370-20) Printed in U.S.A. SY33-8571-6

IBM

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

READER'S COMMENT FORM

This sheet is for comments and suggestions about this manual. We would appreciate *your* views, favorable or unfavorable, in order to aid us in improving *this* publication. This form will be sent directly to the author's department. Please include your name and address if you wish a reply. Contact your IBM branch office for answers to technical questions about the system or when requesting additional publications. Thank you.

Your comments* and suggestions:

*** We would especially appreciate your comments on any of the following topics:**

Clarity of the text	Accuracy	Index	Illustrations	Appearance	Paper
Organization of the text	Cross-references	Tables	Examples	Printing	Binding

FIRST CLASS
PERMIT NO. 1359
WHITE PLAINS, N. Y.

BUSINESS REPLY MAIL

NO POSTAGE STAMP NECESSARY IF MAILED IN THE UNITED STATES

POSTAGE WILL BE PAID BY . . .

IBM Corporation
1133 Westchester Avenue
White Plains, N.Y. 10604

Attention: Department 813DP