IBM

System/370 Reference Summary

GX20-1850-6
File No. S370/4300-01

IBM Corporation, Product Publications, Dept. B98,
PO Box 390, Poughkeepsie, NY, USA 12602

GX20-1850-6

IBM®

GX20-1850-06

Printed in USA

Seventh Edition (July 1986)

This major revision obsoletes GX20-1850-5. Additions include information about expanded storage, the vector facility, and new tape and DASD command codes. Minor technical and editorial revisions have been made throughout.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

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

Please direct any comments on the contents of this publication to IBM Corporation, Product Publications, Department B98, PO Box 390, Poughkeepsie, NY, USA 12602. IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

PREFACE

This publication is intended primarily for use by System/370 assembler language application programmers. It contains basic machine information summarized from the *IBM System/370 Principles of Operation,* GA22-7000, about System/370 Models 115 through 195; the 3031, 3032, 3033, 3081, 3083, 3084, and 3090 Processor Complexes; and the 4321, 4331, 4341, 4361, and 4381 Processors. It also contains frequently used information from *IBM System/370 Vector Operations,* SA22-7125, and the OS/VS, DOS/VSE, and VM/370 assembler language manual, GC33-4010, command codes for various I/O devices, and a multicode translation table. This publication will be updated from time to time. However, the above publications and others cited in this publication are the authoritative reference sources and will be first to reflect changes.

The floating-point instructions, as well as the instructions listed below, are not provided on every model. For instructions that are provided on a particular model, either as standard or optional features on that model, the user should refer to the appropriate System Library publication.

Facility	Instructions
Branch and save	BAS, BASR
Channel-set switching	CONCS, DISCS
Conditional swapping	CS, CDS
CPU timer and clock comparator	SCKC, SPT, STCKC, STPT
Direct control	RDD, WRD
Dual address space	EPAR, ESAR, IAC, IVSK, LASP, MVCP, MVCS, MVCK, PC, PT, SAC, SSAR
Extended facility	IPTE, TPROT
Extended-precision floating point	AXR, LRDR, LRER, MXR, MXDR, MXD, SXR
Move inverse	MVCIN
Multiprocessing	SPX, SIGP, STAP, STPX
PSW-key handling	IPK, SPKA
Storage-key-instruction extensions	ISKE, RRBE, SSKE
Suspend and resume	RIO
Test block	TB
Translation	LRA, PTLB, RRB, STNSM, STOSM
Vector	(All instructions with mnemonics that start with "V")

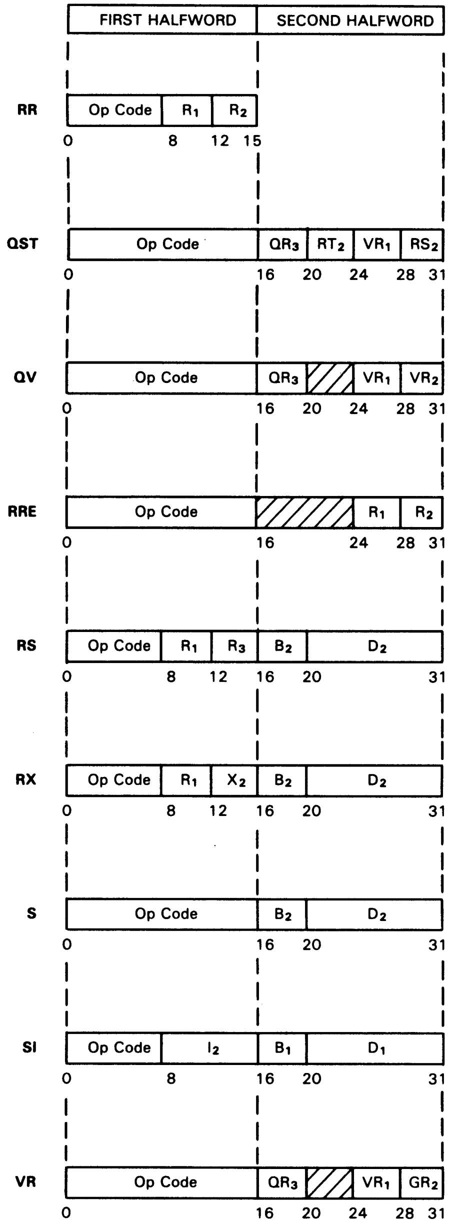
The operation of the following I/O instructions may differ depending on the model, the designated channel, and the installed facilities: CLRCH, CLRIO, HDV, and SIOF. To determine the operation, the user should refer to the appropriate System Library publications.

For information about System/370 extended architecture, refer to *IBM System/370 Extended Architecture Principles of Operation,* SA22-7085, *IBM System/370 Extended Architecture Interpretive Execution,* SA22-7095, and *IBM System/370 Extended Architecture Reference Summary,* GX20-0157.

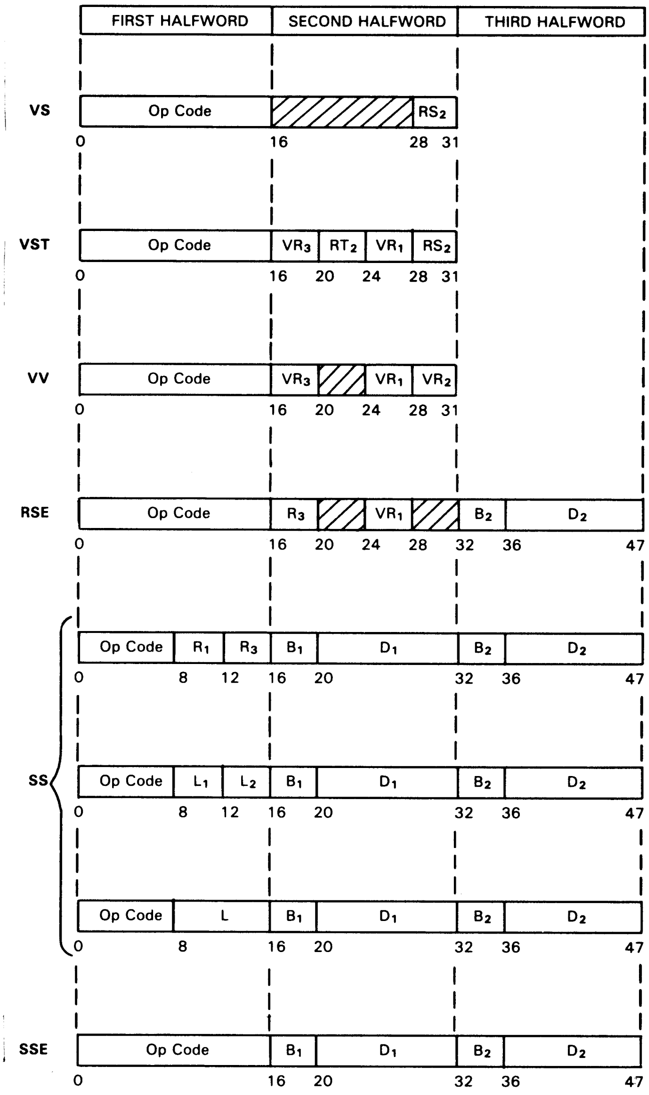
CONTENTS

NOTES

MACHINE INSTRUCTION FORMATS



MACHINE INSTRUCTION FORMATS (Cont'd)



1, 2, 3:	Denotes association with first, second, or third operand
B_1, B_2:	Base register designation field
D_1, D_2:	Displacement field
GR_2:	Register designation field (general register)
I_2:	Immediate operand field
L, L_1, L_2:	Length field
QR_3:	Register designation field (equivalent to GR_3 if general register, or FR_3 if floating-point register)
R_1, R_2, R_3:	Register designation field
RS_2:	Register designation field (starting address of vector)
RT_2:	Register designation field (stride of vector)
VR_1, VR_2, VR_3:	Register designation field (vector register)
X_2:	Index register designation field

MACHINE INSTRUCTIONS

By Mnemonic

Mnemonic	Operands	Name	Format	Op Code	Class & Notes
A	$R_1,D_2(X_2,B_2)$	Add	RX	5A	c
AD	$R_1,D_2(X_2,B_2)$	Add Normalized (L)	RX	6A	c
ADR	R_1,R_2	Add Normalized (L)	RR	2A	c
AE	$R_1,D_2(X_2,B_2)$	Add Normalized (S)	RX	7A	c
AER	R_1,R_2	Add Normalized (S)	RR	3A	c
AH	$R_1,D_2(X_2,B_2)$	Add Halfword	RX	4A	c
AL	$R_1,D_2(X_2,B_2)$	Add Logical	RX	5E	c
ALR	R_1,R_2	Add Logical	RR	1E	c
AP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Add Decimal	SS	FA	c
AR	R_1,R_2	Add	RR	1A	c
AU	$R_1,D_2(X_2,B_2)$	Add Unnormalized (S)	RX	7E	c
AUR	R_1,R_2	Add Unnormalized (S)	RR	3E	c
AW	$R_1,D_2(X_2,B_2)$	Add Unnormalized (L)	RX	6E	c
AWR	R_1,R_2	Add Unnormalized (L)	RR	2E	c
AXR	R_1,R_2	Add Normalized (E)	RR	36	c
BAL	$R_1,D_2(X_2,B_2)$	Branch and Link	RX	45	
BALR	R_1,R_2	Branch and Link	RR	05	
BAS	$R_1,D_2(X_2,B_2)$	Branch and Save	RX	4D	
BASR	R_1,R_2	Branch and Save	RR	0D	
BC	$M_1,D_2(X_2,B_2)$	Branch on Condition	RX	47	
BCR	M_1,R_2	Branch on Condition	RR	07	
BCT	$R_1,D_2(X_2,B_2)$	Branch on Count	RX	46	
BCTR	R_1,R_2	Branch on Count	RR	06	
BXH	$R_1,R_3,D_2(B_2)$	Branch on Index High	RS	86	
BXLE	$R_1,R_3,D_2(B_2)$	Branch on Index Low or Equal	RS	87	
C	$R_1,D_2(X_2,B_2)$	Compare	RX	59	c
CD	$R_1,D_2(X_2,B_2)$	Compare (L)	RX	69	c
CDR	R_1,R_2	Compare (L)	RR	29	c
CDS	$R_1,R_3,D_2(B_2)$	Compare Double and Swap	RS	BB	c
CE	$R_1,D_2(X_2,B_2)$	Compare (S)	RX	79	c
CER	R_1,R_2	Compare (S)	RR	39	c
CH	$R_1,D_2(X_2,B_2)$	Compare Halfword	RX	49	c
CL	$R_1,D_2(X_2,B_2)$	Compare Logical	RX	55	c
CLC	$D_1(L,B_1),D_2(B_2)$	Compare Logical	SS	D5	c
CLCL	R_1,R_2	Compare Logical Long	RR	0F	i c
CLI	$D_1(B_1),I_2$	Compare Logical	SI	95	c
CLM	$R_1,M_3,D_2(B_2)$	Compare Logical Characters under Mask	RS	BD	c
CLR	R_1,R_2	Compare Logical	RR	15	c
CLRCH	$D_2(B_2)$	Clear Channel	S	9F01	pc
CLRIO	$D_2(B_2)$	Clear I/O	S	9D01	pc
CONCS	$D_2(B_2)$	Connect Channel Set	S	B200	pc
CP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Compare Decimal	SS	F9	c
CR	R_1,R_2	Compare	RR	19	c
CS	$R_1,R_3,D_2(B_2)$	Compare and Swap	RS	BA	c
CVB	$R_1,D_2(X_2,B_2)$	Convert to Binary	RX	4F	
CVD	$R_1,D_2(X_2,B_2)$	Convert to Decimal	RX	4E	
D	$R_1,D_2(X_2,B_2)$	Divide	RX	5D	
DD	$R_1,D_2(X_2,B_2)$	Divide (L)	RX	6D	
DDR	R_1,R_2	Divide (L)	RR	2D	
DE	$R_1,D_2(X_2,B_2)$	Divide (S)	RX	7D	
DER	R_1,R_2	Divide (S)	RR	3D	
DISCS	$D_2(B_2)$	Disconnect Channel Set	S	B201	pc
DP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Divide Decimal	SS	FD	
DR	R_1,R_2	Divide	RR	1D	
ED	$D_1(L,B_1),D_2(B_2)$	Edit	SS	DE	c
EDMK	$D_1(L,B_1),D_2(B_2)$	Edit and Mark	SS	DF	c
EPAR	R_1	Extract Primary ASN	RRE	B226	q
ESAR	R_1	Extract Secondary ASN	RRE	B227	q
EX	$R_1,D_2(X_2,B_2)$	Execute	RX	44	

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mnemonic	Operands	Name	Format	Op Code	Class & Notes
HDR	R_1,R_2	Halve (L)	RR	24	
HDV	$D_2(B_2)$	Halt Device	S	9E01	pc
HER	R_1,R_2	Halve (S)	RR	34	
HIO	$D_2(B_2)$	Halt I/O	S	9E00	pc
IAC	R_1	Insert Address Space Control	RRE	B224	qc
IC	$R_1,D_2(X_2,B_2)$	Insert Character	RX	43	
ICM	$R_1,M_3,D_2(B_2)$	Insert Characters under Mask	RS	BF	c
IPK		Insert PSW Key	S	B20B	q
IPTE	R_1,R_2	Invalidate Page Table Entry	RRE	B221	p
ISK	R_1,R_2	Insert Storage Key	RR	09	p
ISKE	R_1,R_2	Insert Storage Key Extended	RRE	B229	p
IVSK	R_1,R_2	Insert Virtual Storage Key	RRE	B223	q
L	$R_1,D_2(X_2,B_2)$	Load	RX	58	
LA	$R_1,D_2(X_2,B_2)$	Load Address	RX	41	
LASP	$D_1(B_1),D_2(B_2)$	Load Address Space Parameters	SSE	E500	pc
LCDR	R_1,R_2	Load Complement (L)	RR	23	c
LCER	R_1,R_2	Load Complement (S)	RR	33	c
LCR	R_1,R_2	Load Complement	RR	13	c
LCTL	$R_1,R_3,D_2(B_2)$	Load Control	RS	B7	p
LD	$R_1,D_2(X_2,B_2)$	Load (L)	RX	68	
LDR	R_1,R_2	Load (L)	RR	28	
LE	$R_1,D_2(X_2,B_2)$	Load (S)	RX	78	
LER	R_1,R_2	Load (S)	RR	38	
LH	$R_1,D_2(X_2,B_2)$	Load Halfword	RX	48	
LM	$R_1,R_3,D_2(B_2)$	Load Multiple	RS	98	
LNDR	R_1,R_2	Load Negative (L)	RR	21	c
LNER	R_1,R_2	Load Negative (S)	RR	31	c
LNR	R_1,R_2	Load Negative	RR	11	c
LPDR	R_1,R_2	Load Positive (L)	RR	20	c
LPER	R_1,R_2	Load Positive (S)	RR	30	c
LPR	R_1,R_2	Load Positive	RR	10	c
LPSW	$D_2(B_2)$	Load PSW	S	82	pn
LR	R_1,R_2	Load	RR	18	
LRA	$R_1,D_2(X_2,B_2)$	Load Real Address	RX	B1	pc
LRDR	R_1,R_2	Load Rounded (E/L)	RR	25	

Floating-point operand lengths:

(E)	Extended source and result.
(E/L)	Extended source, long result.
(L/E)	Long source, extended result.
(L)	Long source and result.
(L/S)	Long source, short result.
(S/L)	Short source, long result.
(S)	Short source and result.

Notes:

c.	Condition code set.
i.	Interruptible instruction.
n.	New condition code loaded.
p.	Privileged instruction.
q.	Semiprivileged instruction.
x.	Execution in problem state and supervisor state differs.
y.	Condition code may be set.

Class (for instructions subject to vector-control bit, CR 0 bit 14)

IC:	Interruptible; (VCT — VIX) elements processed.
IG:	Interruptible; either (bit count in a general register) elements or (section-size — VIX) elements processed, whichever is fewer.
IM:	Interruptible; (VCT — VIX) elements processed, vector-mask mode.
IP:	Interruptible; (partial-sum-number — VIX) elements processed.
IZ:	Interruptible; (section-size) elements processed.
NC:	Not interruptible; (VCT) elements processed.
NZ:	Not interruptible; (section-size) elements processed.
NO:	Not interruptible; no elements processed (VSR/VAC housekeeping).
N1:	Not interruptible; one element processed.

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mne- monic	Operands	Name	For- mat	Op Code	Class & Notes
LRER	R_1,R_2	Load Rounded (L/S)	RR	35	
LTDR	R_1,R_2	Load and Test (L)	RR	22	c
LTER	R_1,R_2	Load and Test (S)	RR	32	c
LTR	R_1,R_2	Load and Test	RR	12	c
M	$R_1,D_2(X_2,B_2)$	Multiply	RX	5C	
MC	$D_1(B_1),I_2$	Monitor Call	SI	AF	
MD	$R_1,D_2(X_2,B_2)$	Multiply (L)	RX	6C	
MDR	R_1,R_2	Multiply (L)	RR	2C	
ME	$R_1,D_2(X_2,B_2)$	Multiply (S/L)	RX	7C	
MER	R_1,R_2	Multiply (S/L)	RR	3C	
MH	$R_1,D_2(X_2,B_2)$	Multiply Halfword	RX	4C	
MP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Multiply Decimal	SS	FC	
MR	R_1,R_2	Multiply	RR	1C	
MVC	$D_1(L,B_1),D_2(B_2)$	Move	SS	D2	
MVCIN	$D_1(L,B_1),D_2(B_2)$	Move Inverse	SS	E8	
MVCK	$D_1(R_1,B_1),D_2(B_2),R_3$	Move with Key	SS	D9	qc
MVCL	R_1,R_2	Move Long	RR	0E	i c
MVCP	$D_1(R_1,B_1),D_2(B_2),R_3$	Move to Primary	SS	DA	qc
MVCS	$D_1(R_1,B_1),D_2(B_2),R_3$	Move to Secondary	SS	DB	qc
MVI	$D_1(B_1),I_2$	Move	SI	92	
MVN	$D_1(L,B_1),D_2(B_2)$	Move Numerics	SS	D1	
MVO	$D_1(L_1,B_1),D_2(L_2,B_2)$	Move with Offset	SS	F1	
MVZ	$D_1(L,B_1),D_2(B_2)$	Move Zones	SS	D3	
MXD	$R_1,D_2(X_2,B_2)$	Multiply (L/E)	RX	67	
MXDR	R_1,R_2	Multiply (L/E)	RR	27	
MXR	R_1,R_2	Multiply (E)	RR	26	
N	$R_1,D_2(X_2,B_2)$	AND	RX	54	c
NC	$D_1(L,B_1),D_2(B_2)$	AND	SS	D4	c
NI	$D_1(B_1),I_2$	AND	SI	94	c
NR	R_1,R_2	AND	RR	14	c
O	$R_1,D_2(X_2,B_2)$	OR	RX	56	c
OC	$D_1(L,B_1),D_2(B_2)$	OR	SS	D6	c
OI	$D_1(B_1),I_2$	OR	SI	96	c
OR	R_1,R_2	OR	RR	16	c
PACK	$D_1(L_1,B_1),D_2(L_2,B_2)$	Pack	SS	F2	
PC	$D_2(B_2)$	Program Call	S	B218	q
PT	R_1,R_2	Program Transfer	RRE	B228	q
PTLB		Purge TLB	S	B20D	p
RDD	$D_1(B_1),I_2$	Read Direct	SI	85	p
RIO	$D_2(B_2)$	Resume I/O	S	9C02	pc
RRB	$D_2(B_2)$	Reset Reference Bit	S	B213	pc
RRBE	R_1,R_2	Reset Reference Bit Extended	RRE	B22A	pc
S	$R_1,D_2(X_2,B_2)$	Subtract	RX	5B	c
SAC	$D_2(B_2)$	Set Address Space Control	S	B219	q
SCK	$D_2(B_2)$	Set Clock	S	B204	pc
SCKC	$D_2(B_2)$	Set Clock Comparator	S	B206	p
SD	$R_1,D_2(X_2,B_2)$	Subtract Normalized (L)	RX	6B	c
SDR	R_1,R_2	Subtract Normalized (L)	RR	2B	c
SE	$R_1,D_2(X_2,B_2)$	Subtract Normalized (S)	RX	7B	c
SER	R_1,R_2	Subtract Normalized (S)	RR	3B	c
SH	$R_1,D_2(X_2,B_2)$	Subtract Halfword	RX	4B	c
SIGP	$R_1,R_3,D_2(B_2)$	Signal Processor	RS	AE	pc
SIO	$D_2(B_2)$	Start I/O	S	9C00	pc
SIOF	$D_2(B_2)$	Start I/O Fast Release	S	9C01	pc
SL	$R_1,D_2(X_2,B_2)$	Subtract Logical	RX	5F	c
SLA	$R_1,D_2(B_2)$	Shift Left Single	RS	8B	c
SLDA	$R_1,D_2(B_2)$	Shift Left Double	RS	8F	c
SLDL	$R_1,D_2(B_2)$	Shift Left Double Logical	RS	8D	
SLL	$R_1,D_2(B_2)$	Shift Left Single Logical	RS	89	
SLR	R_1,R_2	Subtract Logical	RR	1F	c
SP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Subtract Decimal	SS	FB	c

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mne- monic	Operands	Name	For- mat	Op Code	Class & Notes
SPKA	$D_2(B_2)$	Set PSW Key from Address	S	B20A	q
SPM	R_1	Set Program Mask	RR	04	n
SPT	$D_2(B_2)$	Set CPU Timer	S	B208	p
SPX	$D_2(B_2)$	Set Prefix	S	B210	p
SR	R_1,R_2	Subtract	RR	1B	c
SRA	$R_1,D_2(B_2)$	Shift Right Single	RS	8A	c
SRDA	$R_1,D_2(B_2)$	Shift Right Double	RS	8E	c
SRDL	$R_1,D_2(B_2)$	Shift Right Double Logical	RS	8C	
SRL	$R_1,D_2(B_2)$	Shift Right Single Logical	RS	88	
SRP	$D_1(L_1,B_1),D_2(B_2),I_3$	Shift and Round Decimal	SS	F0	c
SSAR	R_1	Set Secondary ASN	RRE	B225	q
SSK	R_1,R_2	Set Storage Key	RR	08	p
SSKE	R_1,R_2	Set Storage Key Extended	RRE	B22B	p
SSM	$D_2(B_2)$	Set System Mask	S	80	p
ST	$R_1,D_2(X_2,B_2)$	Store	RX	50	
STAP	$D_2(B_2)$	Store CPU Address	S	B212	p
STC	$R_1,D_2(X_2,B_2)$	Store Character	RX	42	
STCK	$D_2(B_2)$	Store Clock	S	B205	c
STCKC	$D_2(B_2)$	Store Clock Comparator	S	B207	p
STCM	$R_1,M_3,D_2(B_2)$	Store Characters under Mask	RS	BE	
STCTL	$R_1,R_3,D_2(B_2)$	Store Control	RS	B6	p
STD	$R_1,D_2(X_2,B_2)$	Store (L)	RX	60	
STE	$R_1,D_2(X_2,B_2)$	Store (S)	RX	70	
STH	$R_1,D_2(X_2,B_2)$	Store Halfword	RX	40	
STIDC	$D_2(B_2)$	Store Channel ID	S	B203	pc
STIDP	$D_2(B_2)$	Store CPU ID	S	B202	p
STM	$R_1,R_3,D_2(B_2)$	Store Multiple	RS	90	
STNSM	$D_1(B_1),I_2$	Store Then AND System Mask	SI	AC	p
STOSM	$D_1(B_1),I_2$	Store Then OR System Mask	SI	AD	p
STPT	$D_2(B_2)$	Store CPU Timer	S	B209	p
STPX	$D_2(B_2)$	Store Prefix	S	B211	p
SU	$R_1,D_2(X_2,B_2)$	Subtract Unnormalized (S)	RX	7F	c
SUR	R_1,R_2	Subtract Unnormalized (S)	RR	3F	c
SVC	I	Supervisor Call	RR	0A	
SW	$R_1,D_2(X_2,B_2)$	Subtract Unnormalized (L)	RX	6F	c
SWR	R_1,R_2	Subtract Unnormalized (L)	RR	2F	c
SXR	R_1,R_2	Subtract Normalized (E)	RR	37	c

Floating-point operand lengths:

(E) Extended source and result.
(E/L) Extended source, long result.
(L/E) Long source, extended result.
(L) Long source and result.
(L/S) Long source, short result.
(S/L) Short source, long result.
(S) Short source and result.

Notes:

c. Condition code set.
i. Interruptible instruction.
n. New condition code loaded.
p. Privileged instruction.
q. Semiprivileged instruction.
x. Execution in problem state and supervisor state differs.
y. Condition code may be set.

Class (for instructions subject to vector-control bit, CR 0 bit 14)

IC: Interruptible; (VCT — VIX) elements processed.
IG: Interruptible; either (bit count in a general register) elements or (section-size — VIX) elements processed, whichever is fewer.
IM: Interruptible; (VCT — VIX) elements processed, vector-mask mode.
IP: Interruptible; (partial-sum-number — VIX) elements processed.
IZ: Interruptible; (section-size) elements processed.
NC: Not interruptible; (VCT) elements processed.
NZ: Not interruptible; (section-size) elements processed.
N0: Not interruptible; no elements processed (VSR/VAC housekeeping).
N1: Not interruptible; one element processed.

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mne- monic	Operands	Name	For- mat	Op Code	Class & Notes
TB	R_1,R_2	Test Block	RRE	B22C	ipc
TCH	$D_2(B_2)$	Test Channel	S	9F00	pc
TIO	$D_2(B_2)$	Test I/O	S	9D00	pc
TM	$D_1(B_1),I_2$	Test under Mask	SI	91	c
TPROT	$D_1(B_1),D_2(B_2)$	Test Protection	SSE	E501	pc
TR	$D_1(L,B_1),D_2(B_2)$	Translate	SS	DC	
TRT	$D_1(L,B_1),D_2(B_2)$	Translate and Test	SS	DD	c
TS	$D_2(B_2)$	Test and Set	S	93	c
UNPK	$D_1(L_1,B_1),D_2(L_2,B_2)$	Unpack	SS	F3	
VA	$VR_1,VR_3,RS_2(RT_2)$	Add	VST	A420	IM
VACD	$VR_1,RS_2(RT_2)$	Accumulate (L)	VST	A417	IM
VACDR	VR_1,VR_2	Accumulate (L)	VV	A517	IM
VACE	$VR_1,RS_2(RT_2)$	Accumulate (S/L)	VST	A407	IM
VACER	VR_1,VR_2	Accumulate (S/L)	VV	A507	IM
VACRS	$D_2(B_2)$	Restore VAC	S	A6CB	NO p
VACSV	$D_2(B_2)$	Save VAC	S	A6CA	NO p
VAD	$VR_1,VR_3,RS_2(RT_2)$	Add (L)	VST	A410	IM
VADQ	VR_1,FR_3,VR_2	Add (L)	QV	A590	IM
VADR	VR_1,VR_3,VR_2	Add (L)	VV	A510	IM
VADS	$VR_1,FR_3,RS_2(RT_2)$	Add (L)	QST	A490	IM
VAE	$VR_1,VR_3,RS_2(RT_2)$	Add (S)	VST	A400	IM
VAEQ	VR_1,FR_3,VR_2	Add (S)	QV	A580	IM
VAER	VR_1,VR_3,VR_2	Add (S)	VV	A500	IM
VAES	$VR_1,FR_3,RS_2(RT_2)$	Add (S)	QST	A480	IM
VAQ	VR_1,GR_3,VR_2	Add	QV	A5A0	IM
VAR	VR_1,VR_3,VR_2	Add	VV	A520	IM
VAS	$VR_1,GR_3,RS_2(RT_2)$	Add	QST	A4A0	IM
VC	$M_1,VR_3,RS_2(RT_2)$	Compare	VST	A428	IC
VCD	$M_1,VR_3,RS_2(RT_2)$	Compare (L)	VST	A418	IC
VCDQ	M_1,FR_3,VR_2	Compare (L)	QV	A598	IC
VCDR	M_1,VR_3,VR_2	Compare (L)	VV	A518	IC
VCDS	$M_1,FR_3,RS_2(RT_2)$	Compare (L)	QST	A498	IC
VCE	$M_1,VR_3,RS_2(RT_2)$	Compare (S)	VST	A408	IC
VCEQ	M_1,FR_3,VR_2	Compare (S)	QV	A588	IC
VCER	M_1,VR_3,VR_2	Compare (S)	VV	A508	IC
VCES	$M_1,FR_3,RS_2(RT_2)$	Compare (S)	QST	A488	IC
VCOVM	GR_1	Count Ones in VMR	RRE	A643	NC c
VCQ	M_1,GR_3,VR_2	Compare	QV	A5A8	IC
VCR	M_1,VR_3,VR_2	Compare	VV	A528	IC
VCS	$M_1,GR_3,RS_2(RT_2)$	Compare	QST	A4A8	IC
VCVM		Complement VMR	RRE	A641	NC
VCZVM	GR_1	Count Left Zeros in VMR	RRE	A642	NC c
VDD	$VR_1,VR_3,RS_2(RT_2)$	Divide (L)	VST	A413	IM
VDDQ	VR_1,FR_3,VR_2	Divide (L)	QV	A593	IM
VDDR	VR_1,VR_3,VR_2	Divide (L)	VV	A513	IM
VDDS	$VR_1,FR_3,RS_2(RT_2)$	Divide (L)	QST	A493	IM
VDE	$VR_1,VR_3,RS_2(RT_2)$	Divide (S)	VST	A403	IM
VDEQ	VR_1,FR_3,VR_2	Divide (S)	QV	A583	IM
VDER	VR_1,VR_3,VR_2	Divide (S)	VV	A503	IM
VDES	$VR_1,FR_3,RS_2(RT_2)$	Divide (S)	QST	A483	IM
VL	$VR_1,RS_2(RT_2)$	Load	VST	A409	IC
VLBIX	$VR_1,GR_3,D_2(B_2)$	Load Bit Index	RSE	E428	IG c
VLCDR	VR_1,VR_2	Load Complement (L)	VV	A552	IM
VLCER	VR_1,VR_2	Load Complement (S)	VV	A542	IM
VLCR	VR_1,VR_2	Load Complement	VV	A562	IM
VLCVM	RS_2	Load VMR Complement	VS	A681	NC
VLD	$VR_1,RS_2(RT_2)$	Load (L)	VST	A419	IC
VLDQ	VR_1,FR_2	Load (L)	QV	A599	IC
VLDR	VR_1,VR_2	Load (L)	VV	A519	IC
VLE	$VR_1,RS_2(RT_2)$	Load (S)	VST	A409	IC
VLEL	VR_1,GR_3,GR_2	Load Element	VR	A628	N1
VLELD	VR_1,FR_3,GR_2	Load Element (L)	VR	A618	N1

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mne- monic	Operands	Name	For- mat	Op Code	Class & Notes
VLELE	VR_1,FR_3,GR_2	Load Element (S)	VR	A608	N1
VLEQ	VR_1,FR_2	Load (S)	QV	A589	IC
VLER	VR_1,VR_2	Load (S)	VV	A509	IC
VLH	$VR_1,RS_2(RT_2)$	Load Halfword	VST	A429	IC
VLI	$VR_1,VR_3,D_2(B_2)$	Load Indirect	RSE	E400	IC
VLID	$VR_1,VR_3,D_2(B_2)$	Load Indirect (L)	RSE	E410	IC
VLIE	$VR_1,VR_3,D_2(B_2)$	Load Indirect (S)	RSE	E400	IC
VLINT	$VR_1,RS_2(RT_2)$	Load Integer Vector	VST	A42A	IC
VLM	$VR_1,RS_2(RT_2)$	Load Matched	VST	A40A	IC
VLMD	$VR_1,RS_2(RT_2)$	Load Matched (L)	VST	A41A	IC
VLMDQ	VR_1,FR_2	Load Matched (L)	QV	A59A	IC
VLMDR	VR_1,VR_2	Load Matched (L)	VV	A51A	IC
VLME	$VR_1,RS_2(RT_2)$	Load Matched (S)	VST	A40A	IC
VLMEQ	VR_1,FR_2	Load Matched (S)	QV	A58A	IC
VLMER	VR_1,VR_2	Load Matched (S)	VV	A50A	IC
VLMQ	VR_1,GR_2	Load Matched	QV	A5AA	IC
VLMR	VR_1,VR_2	Load Matched	VV	A50A	IC
VLNDR	VR_1,VR_2	Load Negative (L)	VV	A551	IM
VLNER	VR_1,VR_2	Load Negative (S)	VV	A541	IM
VLNR	VR_1,VR_2	Load Negative	VV	A561	IM
VLPDR	VR_1,VR_2	Load Positive (L)	VV	A550	IM
VLPER	VR_1,VR_2	Load Positive (S)	VV	A540	IM
VLPR	VR_1,VR_2	Load Positive	VV	A560	IM
VLQ	VR_1,GR_2	Load	QV	A5A9	IC
VLR	VR_1,VR_2	Load	VV	A509	IC
VLVCA	$D_2(B_2)$	Load VCT from Address	S	A6C4	NO c
VLVCU	GR_1	Load VCT and Update	RRE	A645	NO c
VLVM	RS_2	Load VMR	VS	A680	NC
VLY	$VR_1,RS_2(RT_2)$	Load Expanded	VST	A40B	IC
VLYD	$VR_1,RS_2(RT_2)$	Load Expanded (L)	VST	A41B	IC
VLYE	$VR_1,RS_2(RT_2)$	Load Expanded (S)	VST	A40B	IC
VLZDR	VR_1	Load Zero (L)	VV	A51B	IC
VLZER	VR_1	Load Zero (S)	VV	A50B	IC
VLZR	VR_1	Load Zero	VV	A50B	IC
VM	$VR_1,VR_3,RS_2(RT_2)$	Multiply	VST	A422	IM
VMAD	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Add (L)	VST	A414	IM
VMADQ	VR_1,FR_3,VR_2	Multiply and Add (L)	QV	A594	IM
VMADS	$VR_1,FR_3,RS_2(RT_2)$	Multiply and Add (L)	QST	A494	IM
VMAE	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Add (S/L)	VST	A404	IM
VMAEQ	VR_1,FR_3,VR_2	Multiply and Add (S/L)	QV	A584	IM

Floating-point operand lengths:

(E) Extended source and result.
(E/L) Extended source, long result.
(L/E) Long source, extended result.
(L) Long source and result.
(L/S) Long source, short result.
(S/L) Short source, long result.
(S) Short source and result.

Notes:

c. Condition code set.
i. Interruptible instruction.
n. New condition code loaded.
p. Privileged instruction.
q. Semiprivileged instruction.
x. Execution in problem state and supervisor state differs.
y. Condition code may be set.

Class (for instructions subject to vector-control bit, CR 0 bit 14)

IC: Interruptible; (VCT — VIX) elements processed.
IG: Interruptible; either (bit count in a general register) elements or (section-size — VIX) elements processed, whichever is fewer.
IM: Interruptible; (VCT — VIX) elements processed, vector-mask mode.
IP: Interruptible; (partial-sum-number — VIX) elements processed.
IZ: Interruptible; (section-size) elements processed.
NC: Not interruptible; (VCT) elements processed.
NZ: Not interruptible; (section-size) elements processed.
NO: Not interruptible; no elements processed (VSR/VAC housekeeping).
N1: Not interruptible; one element processed.

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mnemonic	Operands	Name	Format	Op Code	Class & Notes
VMAES	$VR_1,FR_3,RS_2(RT_2)$	Multiply and Add (S/L)	QST	A484	IM
VMCD	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Accumulate (L)	VST	A416	IM
VMCDR	VR_1,VR_3,VR_2	Multiply and Accumulate (L)	VV	A516	IM
VMCE	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Accumulate (S/L)	VST	A406	IM
VMCER	VR_1,VR_3,VR_2	Multiply and Accumulate (S/L)	VV	A506	IM
VMD	$VR_1,VR_3,RS_2(RT_2)$	Multiply (L)	VST	A412	IM
VMDQ	VR_1,FR_3,VR_2	Multiply (L)	QV	A592	IM
VMDR	VR_1,VR_3,VR_2	Multiply (L)	VV	A512	IM
VMDS	$VR_1,FR_3,RS_2(RT_2)$	Multiply (L)	QST	A492	IM
VME	$VR_1,VR_3,RS_2(RT_2)$	Multiply (S/L)	VST	A402	IM
VMEQ	VR_1,FR_3,VR_2	Multiply (S/L)	QV	A582	IM
VMER	VR_1,VR_3,VR_2	Multiply (S/L)	VV	A502	IM
VMES	$VR_1,FR_3,RS_2(RT_2)$	Multiply (S/L)	QST	A482	IM
VMNSD	VR_1,FR_3,GR_2	Minimum Signed (L)	VR	A611	IM
VMNSE	VR_1,FR_3,GR_2	Minimum Signed (S)	VR	A601	IM
VMQ	VR_1,GR_3,VR_2	Multiply	QV	A5A2	IM
VMR	VR_1,VR_3,VR_2	Multiply	VV	A522	IM
VMRRS	$D_2(B_2)$	Restore VMR	S	A6C3	NZ
VMRSV	$D_2(B_2)$	Save VMR	S	A6C1	NZ
VMS	$VR_1,GR_3,RS_2(RT_2)$	Multiply	QST	A4A2	IM
VMSD	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Subtract (L)	VST	A415	IM
VMSDQ	VR_1,FR_3,VR_2	Multiply and Subtract (L)	QV	A595	IM
VMSDS	$VR_1,FR_3,RS_2(RT_2)$	Multiply and Subtract (L)	QST	A495	IM
VMSE	$VR_1,VR_3,RS_2(RT_2)$	Multiply and Subtract (S/L)	VST	A405	IM
VMSEQ	VR_1,FR_3,VR_2	Multiply and Subtract (S/L)	QV	A585	IM
VMSES	$VR_1,FR_3,RS_2(RT_2)$	Multiply and Subtract (S/L)	QST	A485	IM
VMXAD	VR_1,FR_3,GR_2	Maximum Absolute (L)	VR	A612	IM
VMXAE	VR_1,FR_3,GR_2	Maximum Absolute (S)	VR	A602	IM
VMXSD	VR_1,FR_3,GR_2	Maximum Signed (L)	VR	A610	IM
VMXSE	VR_1,FR_3,GR_2	Maximum Signed (S)	VR	A600	IM
VN	$VR_1,VR_3,RS_2(RT_2)$	AND	VST	A424	IM
VNQ	VR_1,GR_3,VR_2	AND	QV	A5A4	IM
VNR	VR_1,VR_3,VR_2	AND	VV	A524	IM
VNS	$VR_1,GR_3,RS_2(RT_2)$	AND	QST	A4A4	IM
VNVM	RS_2	AND to VMR	VS	A684	NC
VO	$VR_1,VR_3,RS_2(RT_2)$	OR	VST	A425	IM
VOQ	VR_1,GR_3,VR_2	OR	QV	A5A5	IM
VOR	VR_1,VR_3,VR_2	OR	VV	A525	IM
VOS	$VR_1,GR_3,RS_2(RT_2)$	OR	QST	A4A5	IM
VOVM	RS_2	OR to VMR	VS	A685	NC
VRCL	$D_2(B_2)$	Clear VR	S	A6C5	IZ
VRRS	GR_1	Restore VR	RRE	A648	IZ xc
VRSV	GR_1	Save VR	RRE	A64A	IZ c
VRSVC	GR_1	Save Changed VR	RRE	A649	IZ pc
VS	$VR_1,VR_3,RS_2(RT_2)$	Subtract	VST	A421	IM
VSD	$VR_1,VR_3,RS_2(RT_2)$	Subtract (L)	VST	A411	IM
VSDQ	VR_1,FR_3,VR_2	Subtract (L)	QV	A591	IM
VSDR	VR_1,VR_3,VR_2	Subtract (L)	VV	A511	IM
VSDS	$VR_1,FR_3,RS_2(RT_2)$	Subtract (L)	QST	A491	IM
VSE	$VR_1,VR_3,RS_2(RT_2)$	Subtract (S)	VST	A401	IM
VSEQ	VR_1,FR_3,VR_2	Subtract (S)	QV	A581	IM
VSER	VR_1,VR_3,VR_2	Subtract (S)	VV	A501	IM
VSES	$VR_1,FR_3,RS_2(RT_2)$	Subtract (S)	QST	A481	IM
VSLL	$VR_1,VR_3,D_2(B_2)$	Shift Left Single Logical	RSE	E425	IM
VSPSD	VR_1,FR_2	Sum Partial Sums (L)	VR	A61A	IP
VSQ	VR_1,GR_3,VR_2	Subtract	QV	A5A1	IM
VSR	$VR_1,VR_3.VR_2$	Subtract	VV	A521	IM
VSRL	$VR_1,VR_3,D_2(B_2)$	Shift Right Single Logical	RSE	E424	IM

MACHINE INSTRUCTIONS (Cont'd)

By Mnemonic (Cont'd)

Mnemonic	Operands	Name	Format	Op Code	Class & Notes
VSRRS	$D_2(B_2)$	Restore VSR	S	A6C2	IZ x
VSRSV	$D_2(B_2)$	Save VSR	S	A6C0	NO x
VSS	$VR_1,GR_3,RS_2(RT_2)$	Subtract	QST	A4A1	IM
VST	$VR_1,RS_2(RT_2)$	Store	VST	A40D	IC
VSTD	$VR_1,RS_2(RT_2)$	Store (L)	VST	A41D	IC
VSTE	$VR_1,RS_2(RT_2)$	Store (S)	VST	A40D	IC
VSTH	$VR_1,RS_2(RT_2)$	Store Halfword	VST	A42D	IC
VSTI	$VR_1,VR_3,D_2(B_2)$	Store Indirect	RSE	E401	IC
VSTID	$VR_1,VR_3,D_2(B_2)$	Store Indirect (L)	RSE	E411	IC
VSTIE	$VR_1,VR_3,D_2(B_2)$	Store Indirect (S)	RSE	E401	IC
VSTK	$VR_1,RS_2(RT_2)$	Store Compressed	VST	A40F	IC
VSTKD	$VR_1,RS_2(RT_2)$	Store Compressed (L)	VST	A41F	IC
VSTKE	$VR_1,RS_2(RT_2)$	Store Compressed (S)	VST	A40F	IC
VSTM	$VR_1,RS_2(RT_2)$	Store Matched	VST	A40E	IC
VSTMD	$VR_1,RS_2(RT_2)$	Store Matched (L)	VST	A41E	IC
VSTME	$VR_1,RS_2(RT_2)$	Store Matched (S)	VST	A40E	IC
VSTVM	RS_2	Store VMR	VS	A682	NC
VSTVP	$D_2(B_2)$	Store Vector Parameters	S	A6C8	NO
VSVMM	$D_2(B_2)$	Set Vector Mask Mode	S	A6C6	NO
VTVM		Test VMR	RRE	A640	NC c
VX	$VR_1,VR_3,RS_2(RT_2)$	Exclusive OR	VST	A426	IM
VXEL	VR_1,GR_3,GR_2	Extract Element	VR	A629	N1
VXELD	VR_1,FR_3,GR_2	Extract Element (L)	VR	A619	N1
VXELE	VR_1,FR_3,GR_2	Extract Element (S)	VR	A609	N1
VXQ	VR_1,GR_3,VR_2	Exclusive OR	QV	A5A6	IM
VXR	VR_1,VR_3,VR_2	Exclusive OR	VV	A526	IM
VXS	$VR_1,GR_3,RS_2(RT_2)$	Exclusive OR	QST	A4A6	IM
VXVC	GR_1	Extract VCT	RRE	A644	NO
VXVM	RS_2	Exclusive OR to VMR	VS	A686	NC
VXVMM	GR_1	Extract Vector Mask Mode	RRE	A646	NO
VZPSD	VR_1	Zero Partial Sums (L)	VR	A61B	IP
WRD	$D_1(B_1),I_2$	Write Direct	SI	84	p
X	$R_1,D_2(X_2,B_2)$	Exclusive OR	RX	57	c
XC	$D_1(L,B_1),D_2(B_2)$	Exclusive OR	SS	D7	c
XI	$D_1(B_1),I_2$	Exclusive OR	SI	97	c
XR	R_1,R_2	Exclusive OR	RR	17	c
ZAP	$D_1(L_1,B_1),D_2(L_2,B_2)$	Zero and Add	SS	F8	c
– –	Model-dependent	Diagnose	– –	83	py

Floating-point operand lengths:

(E) Extended source and result.
(E/L) Extended source, long result.
(L/E) Long source, extended result.
(L) Long source and result.
(L/S) Long source, short result.
(S/L) Short source, long result.
(S) Short source and result.

Notes:

c. Condition code set.
i. Interruptible instruction.
n. New condition code loaded.
p. Privileged instruction.
q. Semiprivileged instruction.
x. Execution in problem state and supervisor state differs.
y. Condition code may be set.

Class (for instructions subject to vector-control bit, CR 0 bit 14)

IC: Interruptible; (VCT – VIX) elements processed.
IG: Interruptible; either (bit count in a general register) elements or (section-size – VIX) elements processed, whichever is fewer.
IM: Interruptible; (VCT – VIX) elements processed, vector-mask mode.
IP: Interruptible; (partial-sum-number – VIX) elements processed.
IZ: Interruptible; (section-size) elements processed.
NC: Not interruptible; (VCT) elements processed.
NZ: Not interruptible; (section-size) elements processed.
NO: Not interruptible; no elements processed (VSR/VAC housekeeping).
N1: Not interruptible; one element processed.

MACHINE INSTRUCTIONS (Cont'd)

By Operation Code

Op Code	Mne- monic
04	SPM
05	BALR
06	BCTR
07	BCR
08	SSK
09	ISK
0A	SVC
0D	BASR
0E	MVCL
0F	CLCL
10	LPR
11	LNR
12	LTR
13	LCR
14	NR
15	CLR
16	OR
17	XR
18	LR
19	CR
1A	AR
1B	SR
1C	MR
1D	DR
1E	ALR
1F	SLR
20	LPDR
21	LNDR
22	LTDR
23	LCDR
24	HDR
25	LRDR
26	MXR
27	MXDR
28	LDR
29	CDR
2A	ADR
2B	SDR
2C	MDR
2D	DDR
2E	AWR
2F	SWR
30	LPER
31	LNER
32	LTER
33	LCER
34	HER
35	LRER
36	AXR
37	SXR
38	LER
39	CER
3A	AER
3B	SER
3C	MER
3D	DER
3E	AUR
3F	SUR
40	STH
41	LA
42	STC
43	IC
44	EX
45	BAL
46	BCT

Op Code	Mne- monic
47	BC
48	LH
49	CH
4A	AH
4B	SH
4C	MH
4D	BAS
4E	CVD
4F	CVB
50	ST
54	N
55	CL
56	O
57	X
58	L
59	C
5A	A
5B	S
5C	M
5D	D
5E	AL
5F	SL
60	STD
67	MXD
68	LD
69	CD
6A	AD
6B	SD
6C	MD
6D	DD
6E	AW
6F	SW
70	STE
78	LE
79	CE
7A	AE
7B	SE
7C	ME
7D	DE
7E	AU
7F	SU
80	SSM
82	LPSW
83	Diagnose
84	WRD
85	RDD
86	BXH
87	BXLE
88	SRL
89	SLL
8A	SRA
8B	SLA
8C	SRDL
8D	SLDL
8E	SRDA
8F	SLDA
90	STM
91	TM
92	MVI
93	TS
94	NI
95	CLI
96	OI
97	XI
98	LM

Op Code	Mne- monic
9C00	SIO
9C01	SIOF
9C02	RIO
9D00	TIO
9D01	CLRIO
9E00	HIO
9E01	HDV
9F00	TCH
9F01	CLRCH
A400	VAE
A401	VSE
A402	VME
A403	VDE
A404	VMAE
A405	VMSE
A406	VMCE
A407	VACE
A408	VCE
A409	VL
A409	VLE
A40A	VLM
A40A	VLME
A40B	VLY
A40B	VLYE
A40D	VST
A40D	VSTE
A40E	VSTM
A40E	VSTME
A40F	VSTK
A40F	VSTKE
A410	VAD
A411	VSD
A412	VMD
A413	VDD
A414	VMAD
A415	VMSD
A416	VMCD
A417	VACD
A418	VCD
A419	VLD
A41A	VLMD
A41B	VLYD
A41D	VSTD
A41E	VSTMD
A41F	VSTKD
A420	VA
A421	VS
A422	VM
A424	VN
A425	VO
A426	VX
A428	VC
A429	VLH
A42A	VLINT
A42D	VSTH
A480	VAES
A481	VSES
A482	VMES
A483	VDES
A484	VMAES
A485	VMSES
A488	VCES
A490	VADS
A491	VSDS
A492	VMDS

MACHINE INSTRUCTIONS (Cont'd)

By Operation Code (Cont'd)

Op Code	Mne- monic
A493	VDDS
A494	VMADS
A495	VMSDS
A498	VCDS
A4A0	VAS
A4A1	VSS
A4A2	VMS
A4A4	VNS
A4A5	VOS
A4A6	VXS
A4A8	VCS
A500	VAER
A501	VSER
A502	VMER
A503	VDER
A506	VMCER
A507	VACER
A508	VCER
A509	VLER
A509	VLR
A50A	VLMER
A50A	VLMR
A50B	VLZER
A50B	VLZR
A510	VADR
A511	VSDR
A512	VMDR
A513	VDDR
A516	VMCDR
A517	VACDR
A518	VCDR
A519	VLDR
A51A	VLMDR
A51B	VLZDR
A520	VAR
A521	VSR
A522	VMR
A524	VNR
A525	VOR
A526	VXR
A528	VCR
A540	VLPER
A541	VLNER
A542	VLCER
A550	VLPDR
A551	VLNDR
A552	VLCDR
A560	VLPR
A561	VLNR
A562	VLCR
A580	VAEQ
A581	VSEQ
A582	VMEQ
A583	VDEQ
A584	VMAEQ
A585	VMSEQ
A588	VCEQ
A589	VLEQ
A58A	VLMEQ
A590	VADQ
A591	VSDQ
A592	VMDQ
A593	VDDQ
A594	VMADQ
A595	VMSDQ

Op Code	Mne- monic
A598	VCDQ
A599	VLDQ
A59A	VLMDQ
A5A0	VAQ
A5A1	VSQ
A5A2	VMQ
A5A4	VNQ
A5A5	VOQ
A5A6	VXQ
A5A8	VCQ
A5A9	VLQ
A5AA	VLMQ
A600	VMXSE
A601	VMNSE
A602	VMXAE
A608	VLELE
A609	VXELE
A610	VMXSD
A611	VMNSD
A612	VMXAD
A618	VLELD
A619	VXELD
A61A	VSPSD
A61B	VZPSD
A628	VLEL
A629	VXEL
A640	VTVM
A641	VCVM
A642	VCZVM
A643	VCOVM
A644	VXVC
A645	VLVCU
A646	VXVMM
A648	VRRS
A649	VRSVC
A64A	VRSV
A680	VLVM
A681	VLCVM
A682	VSTVM
A684	VNVM
A685	VOVM
A686	VXVM
A6C0	VSRSV
A6C1	VMRSV
A6C2	VSRRS
A6C3	VMRRS
A6C4	VLVCA
A6C5	VRCL
A6C6	VSVMM
A6C8	VSTVP
A6CA	VACSV
A6CB	VACRS
AC	STNSM
AD	STOSM
AE	SIGP
AF	MC
B1	LRA
B200	CONCS
B201	DISCS
B202	STIDP
B203	STIDC
B204	SCK
B205	STCK
B206	SCKC
B207	STCKC

Op Code	Mne- monic
B208	SPT
B209	STPT
B20A	SPKA
B20B	IPK
B20D	PTLB
B210	SPX
B211	STPX
B212	STAP
B213	RRB
B218	PC
B219	SAC
B221	IPTE
B223	IVSK
B224	IAC
B225	SSAR
B226	EPAR
B227	ESAR
B228	PT
B229	ISKE
B22A	RRBE
B22B	SSKE
B22C	TB
B6	STCTL
B7	LCTL
BA	CS
BB	CDS
BD	CLM
BE	STCM
BF	ICM
D1	MVN
D2	MVC
D3	MVZ
D4	NC
D5	CLC
D6	OC
D7	XC
D9	MVCK
DA	MVCP
DB	MVCS
DC	TR
DD	TRT
DE	ED
DF	EDMK
E400	VLI
E400	VLIE
E401	VSTI
E401	VSTIE
E410	VLID
E411	VSTID
E424	VSRL
E425	VSLL
E428	VLBIX
E500	LASP
E501	TPROT
E8	MVCIN
F0	SRP
F1	MVO
F2	PACK
F3	UNPK
F8	ZAP
F9	CP
FA	AP
FB	SP
FC	MP
FD	DP

CONDITION CODES

Condition Code →	0	1	2	3
Mask Bit Value →	8	4	2	1
Binary and Logical Instructions (See Note)				
Add	Zero	< Zero	> Zero	Overflow
Add Halfword	Zero	< Zero	> Zero	Overflow
Add Logical	Zero, no carry	Not zero, no carry	Zero, carry	Not zero, carry
AND	Zero	Not zero	– –	– –
Compare	Equal	First op low	First op high	– –
Compare and Swap	Equal	Not equal	– –	– –
Compare Double and Swap	Equal	Not equal	– –	– –
Compare Halfword	Equal	First op low	First op high	– –
Compare Logical	Equal	First op low	First op high	– –
Compare Logical Characters under Mask	Equal, or mask is zero	First op low	First op high	– –
Compare Logical Long	Equal, or lengths both = 0	First op low	First op high	– –
Exclusive OR	Zero	Not zero	– –	– –
Insert Characters under Mask	All zero, or mask is zero	Leftmost bit = 1	Not zero, but with leftmost bit = 0	– –
Load and Test	Zero	< Zero	> Zero	– –
Load Complement	Zero	< Zero	> Zero	Overflow
Load Negative	Zero	< Zero	– –	– –
Load Positive	Zero	– –	> Zero	Overflow
Move Long	Operand lengths equal	First op shorter	First op longer	Overlap
OR	Zero	Not zero	– –	– –
Shift Left Double	Zero	< Zero	> Zero	Overflow
Shift Left Single	Zero	< Zero	> Zero	Overflow
Shift Right Double	Zero	< Zero	> Zero	– –
Shift Right Single	Zero	< Zero	> Zero	– –
Subtract	Zero	< Zero	> Zero	Overflow
Subtract Halfword	Zero	< Zero	> Zero	Overflow
Subtract Logical	– –	Not zero, no carry	Zero, carry	Not zero, carry
Test and Set	Leftmost bit zero	Leftmost bit one	– –	– –
Test under Mask	All zeros, or mask is zero	Mixed 0's and 1's	– –	All ones
Translate and Test	All zeros	Not zero, scan incomplete	Not zero, scan complete	– –

Note: Vector instructions with binary or logical operands do not set the condition code.

CONDITION CODES (Cont'd)

Condition Code →	0	1	2	3
Mask Bit Value →	8	4	2	1
Decimal Instructions				
Add Decimal	Zero	< Zero	> Zero	Overflow
Compare Decimal	Equal	First op low	First op high	– –
Edit	Zero	< Zero	> Zero	– –
Edit and Mark	Zero	< Zero	> Zero	– –
Shift and Round Decimal	Zero	< Zero	> Zero	Overflow
Subtract Decimal	Zero	< Zero	> Zero	Overflow
Zero and Add	Zero	< Zero	> Zero	Overflow
Floating-Point Instructions (See Note)				
Add Normalized	Zero	< Zero	> Zero	– –
Add Unnormalized	Zero	< Zero	> Zero	– –
Compare	Equal	First op low	First op high	– –
Load and Test	Zero	< Zero	> Zero	– –
Load Complement	Zero	< Zero	> Zero	– –
Load Negative	Zero	< Zero	– –	– –
Load Positive	Zero	– –	> Zero	– –
Subtract Normalized	Zero	< Zero	> Zero	– –
Subtract Unnormalized	Zero	< Zero	> Zero	– –
General Instructions				
Count Left Zeros in VMR	Active bits all zeros	Active bits 0's and 1's	– –	Active bits all ones
Count Ones in VMR	Active bits all zeros	Active bits 0's and 1's	– –	Active bits all ones
Load Bit Index	VCT = 0 and bit count = 0	VCT = 0 and bit count < 0	VCT = section size and bit count > 0	VCT > 0 and bit count not > 0
Load VCT and Update	VCT = 0 and new count = 0	VCT = 0 and new count < 0	VCT = section size and new count > 0	VCT > 0 and new count = 0
Load VCT from Address	VCT = 0 and eff addr = 0	VCT = 0 and eff addr < 0	VCT = section size and eff addr > section size	VCT > 0 and eff addr ≤ section size
Restore VR	VR14-VR15 examined and not loaded	VR0-VR13 examined and not loaded	VR14-VR15 loaded	VR0-VR13 loaded

Note: Vector instructions with floating-point operands do not set the condition code.

CONDITION CODES (Cont'd)

Condition Code →	0	1	2	3
Mask Bit Value →	**8**	**4**	**2**	**1**
General Instructions (Continued)				
Save VR	VR14-VR15 examined and not stored	VR0-VR13 examined and not stored	VR14-VR15 stored	VR0-VR13 stored
Store Clock	Set state	Not-set state	Error state	Stopped state or not oper
Test VMR	Active bits all zeros	Active bits 0's and 1's	– –	Active bits all ones
Control Instructions				
Connect Channel Set	Successful	Connected to other CPU	– –	Not oper
Diagnose	See Note	See Note	See Note	See Note
Disconnect Channel Set	Successful	Connected to other CPU	– –	Not oper
Insert Address Space Control	Zero	One	– –	– –
Load Address Space Parameters	Parameters loaded	Primary not available	Secondary not authorized or not available	Space-switch event
Load PSW	See Note	See Note	See Note	See Note
Load Real Address	Translation available	Segment-table entry invalid	Page-table entry invalid	Table length exceeded
Move to Primary	Length ≤ 256	– –	– –	Length > 256
Move to Secondary	Length ≤256	– –	– –	Length > 256
Move with Key	Length ≤256	– –	– –	Length > 256
Reset Reference Bit	Ref = 0, Chg = 0	Ref = 0, Chg = 1	Ref = 1, Chg = 0	Ref = 1, Chg = 1
Reset Reference Bit Extended	Ref = 0, Chg = 0	Ref = 0, Chg = 1	Ref = 1, Chg = 0	Ref = 1, Chg = 1

Note: For Diagnose, the resulting condition code is model-dependent. For Load PSW, the condition code is loaded from a field of the second operand (the new PSW's condition code field).

CONDITION CODES (Cont'd)

Condition Code →	0	1	2	3
Mask Bit Value →	**8**	**4**	**2**	**1**
Control Instructions (Continued)				
Save Changed VR	VR14-VR15 examined and not stored	VR0-VR13 examined and not stored	VR14-VR15 stored	VR0-VR13 stored
Set Clock	Set	Secure	– –	Not oper
Signal Processor	Accepted	Status stored	Busy	Not oper
Test Block	Usable	Unusable	– –	– –
Test Protection	Fetch and store allowed	Fetch allowed; no store allowed	No fetch or store allowed	Translation not available
Input/Output Instructions				
Clear Channel	Reset signaled	– –	Channel busy	Not oper
Clear I/O	No operation in progress	CSW stored	Channel busy	Not oper
Halt Device	Busy or interruption pending	CSW stored	Channel working	Not oper
Halt I/O	Interruption pending	CSW stored	Burst op stopped	Not oper
Resume I/O	Successful	– –	– –	Not oper
Start I/O	Successful	CSW stored	Busy	Not oper
Start I/O Fast Release	Successful	CSW stored	Busy	Not oper
Stored Channel ID	Chan ID stored	CSW stored	Busy	Not oper
Test Channel	Available	Interruption pending	Working in burst mode	Not oper
Test I/O	Available	CSW stored	Busy	Not oper

ASSEMBLER INSTRUCTIONS

Function	Mnemonic	Meaning
Data definition	DC	Define constant
	DS	Define storage
	CCW	Define channel command word
	CCW0**	Define format-0 channel command word
	CCW1**	Define format-1 channel command word
Program sectioning and linking	START	Start assembly
	LOCTR**	Specify multiple location counters
	CSECT	Identify control section
	DSECT	Identify dummy section
	DXD*	Define external dummy section
	CXD*	Cumulative length of external dummy section
	COM	Identify blank common control section
	AMODE**	Specify addressing mode
	RMODE**	Specify residence mode
	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
	OPSYN*	Equate operation code
	PUSH*	Save current PRINT or USING status
	POP*	Restore PRINT or USING status
	LTORG	Begin literal pool
	CNOP	Conditional no operation
	COPY	Copy predefined source coding
	END	End assembly
Macro definition	MACRO	Macro definition header
	MEXIT	Macro definition exit
	MEND	Macro definition trailer
	AREAD**	Assign card to SETC symbol
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
	MNOTE	Generate error message
	MHELP**	Trace macro flow
	SETA	Set arithmetic variable symbol
	SETB	Set binary variable symbol
	SETC	Set character variable symbol

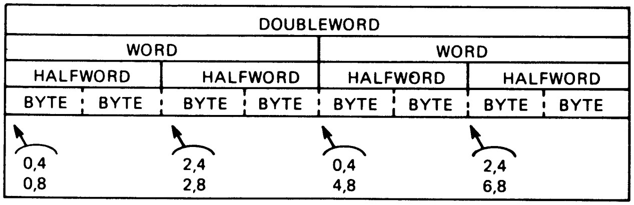
Source: GC33-4010 for the OS/VS, VM/370, and DOS/VSE Assembler, and GC26-4037 for Assembler H Version 2.
*Not for use with the DOS/VSE Assembler.
**Assembler H Version 2 only.

EXTENDED MNEMONIC INSTRUCTIONS

Use	Extended Mnemonic* (RX or RR)	Meaning	Machine Instr.* (RX or RR)
General	B or BR	Unconditional Branch	BC or BCR 15,
	NOP or NOPR	No Operation	BC or BCR 0,
After Compare Instructions (A:B)	BH or BHR	Branch on A High	BC or BCR 2,
	BL or BLR	Branch on A Low	BC or BCR 4,
	BE or BER	Branch on A Equal B	BC or BCR 8,
	BNH or BNHR	Branch on A Not High	BC or BCR 13,
	BNL or BNLR	Branch on A Not Low	BC or BCR 11,
	BNE or BNER	Branch on A Not Equal B	BC or BCR 7,
After Arithmetic Instructions	BP or BPR	Branch on Plus	BC or BCR 2,
	BM or BMR	Branch on Minus	BC or BCR 4,
	BZ or BZR	Branch on Zero	BC or BCR 8,
	BO or BOR	Branch on Overflow	BC or BCR 1,
	BNP or BNPR	Branch on Not Plus	BC or BCR 13,
	BNM or BNMR	Branch on Not Minus	BC or BCR 11,
	BNZ or BNZR	Branch on Not Zero	BC or BCR 7,
	BNO or BNOR	Branch on No Overflow	BC or BCR 14,
After Test under Mask Instruction	BO or BOR	Branch if Ones	BC or BCR 1,
	BM or BMR	Branch if Mixed	BC or BCR 4,
	BZ or BZR	Branch if Zeros	BC or BCR 8,
	BNO or BNOR	Branch if Not Ones	BC or BCR 14,
	BNM or BNMR	Branch if Not Mixed	BC or BCR 11,
	BNZ or BNZR	Branch if Not Zeros	BC or BCR 7,

Source: GC33-4010 for the OS/VS, VM/370, and DOS/VSE Assembler, and GC26-4037 for Assembler H Version 2.
*Second operand, not shown, is $D_2(X_2,B_2)$ for RX format and R_2 for RR format.

CNOP ALIGNMENT



Source: GC33-4010 for the OS/VS, VM/370, and DOS/VSE Assembler, and GC26-4037 for Assembler H Version 2.

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
Q*	4	word	symbol naming a DXD or DSECT	left

Source: GC33-4010 for the OS/VS, VM/370, and DOS/VSE Assembler, and GC26-4037 for Assembler H Version 2.
*Not for use with the DOS/VSE Assembler.

FIXED STORAGE LOCATIONS

Area, dec.	Addr type	Hex addr	EC only	Function
0- 7	A	0		Initial-program-loading PSW
0- 7	R	0		Restart new PSW
8- 15	A	8		Initial-program-loading CCW1
8- 15	R	8		Restart old PSW
16- 23	A	10		Initial-program-loading CCW2
24- 31	R	18		External old PSW
32- 39	R	20		Supervisor-call old PSW
40- 47	R	28		Program old PSW
48- 55	R	30		Machine-check old PSW
56- 63	R	38		Input/output old PSW
64- 71	R	40		Channel-status word (see diagram)
72- 75	R	48		Channel-address word (see diagram)
80- 83	R	50		Interval timer
84- 87	L	54		Trace-table designation (0 control, 8-31 address)
88- 95	R	58		External new PSW
96-103	R	60		Supervisor-call new PSW
104-111	R	68		Program new PSW
112-119	R	70		Machine-check new PSW
120-127	R	78		Input/output new PSW
128-131	R	80		External-interruption parameter for service signal
132-133	R	84		CPU address associated with external interruption, or unchanged
132-133	R	84	X	CPU address associated with external interruption, or zeros
134-135	R	86	X	External-interruption code (see table)
136-139	R	88	X	SVC interruption (0-12 zeros, 13-14 ILC, 15:0, 16-31 code)
140-143	R	8C	X	Program interruption (0-12 zeros, 13-14 ILC, 15:0, 16-31 code)
144-147	R	90	X	Translation-exception ID (see table)
148-149	R	94		Monitor class (0-7 zeros, 8-15 class number)
150-151	R	96	X	PER code (0-3 code, 4-15 zeros)
152-155	R	98	X	PER address (0-7 zeros, 8-31 address)
156-159	R	9C		Monitor code (0-7 zeros, 8-31 code)
168-171	R	A8		Channel ID (0-3 type, 4-15 model, 16-31 max. IOEL length)
172-175	R	AC		I/O-extended-logout address (0-7 unused, 8-31 address)
176-179	R	B0		Limited channel logout (see diagram)
185	R	B9	X	Measurement byte (0-1 delay, 2-4 count, 5-7 zeros)
186-187	R	BA	X	I/O address
216-223	A	D8		Store-status CPU-timer save area
216-223	R	D8		Machine-check CPU-timer save area
224-231	A	E0		Store-status clock-comparator save area
224-231	R	E0		Machine-check clock-comparator save area
232-239	R	E8		Machine-check-interruption code (see diagram)
244-247	R	F4		External-damage code (see diagram)
248-251	R	F8		Failing-storage address (0-5 zeros, 6-31 address)
252-255	R	FC		Region code*
256-263	A	100		Store-status PSW save area
256-351	R	100		Fixed-logout area*
264-267	A	108		Store-status prefix save area
268-271	A	10C		Store-status model-dependent save area*
352-383	A	160		Store-status floating-point-register save area
352-383	R	160		Machine-check floating-point-register save area
384-447	A	180		Store-status general-register save area
384-447	R	180		Machine-check general-register save area
448-511	A	1C0		Store-status control-register save area
448-511	R	1C0		Machine-check control-register save area
795	L	31B		CPU identity for DAS tracing

A = Absolute address
L = Logical address
R = Real address
*May vary among models; see System Library manuals for specific model.

CONTROL REGISTERS

CR	Bits	Name of field	Associated with	Init.*
0	0	Block-multiplexing control	Block-multiplexing	0
	1	SSM-suppression control	SSM instruction	0
	2	TOD-clock-sync control	Multiprocessing	0
	3	Low-addr-protection control	Low-address protection	0
	4	Extraction-authority control	Dual address space	0
	5	Secondary-space control		0
	7	Storage-key exception control	Storage-key 4K-byte block	0
	8-12	Translation format	Dynamic address trans	0
	14	Vector control	Vector facility	0
	16	Malfunc-alert subclass mask	Multiprocessing	0
	17	Emergency-signal subcl mask		0
	18	External-call subclass mask		0
	19	TOD-clk sync-chk subcl mask		0
	20	Clk-comparator subclass mask	Clock comparator	0
	21	CPU-timer subclass mask	CPU timer	0
	22	Service-signal subclass mask	Service signal	0
	24	Interval-timer subclass mask	Interval timer	1
	25	Interrupt-key subclass mask	Interrupt key	1
	26	External-signal subcl mask	External signal	1
1	0-7	Primary segment-table length	Dynamic address trans	0
	8-25	Primary segment-table origin		0
	31	Space-switch-event control	Dual address space	0
2	0-31	Channel masks	Channels	1
3	0-15	PSW-key mask	Dual address space	0
	16-31	Secondary ASN		0
4	0-15	Authorization index	Dual address space	0
	16-31	Primary ASN		0
5	0	Subsystem-linkage control	Dual address space	0
	8-24	Linkage-table origin		0
	25-31	Linkage-table length		0
7	0-7	Secondary segment-table length	Dual address space	0
	8-25	Secondary segment-table origin		0
8	16-31	Monitor masks	MC instruction	0
9	0	Successful-branching-event mask	Program-event recording	0
	1	Instruction-fetching-event mask		0
	2	Storage-alteration-event mask		0
	3	GR-alteration-event mask		0
	16-31	PER general-register masks		0
10	8-31	PER starting address	Program-event recording	0
11	8-31	PER ending address	Program-event recording	0
14	0	Check-stop control	Machine-check handling	1
	1	Synch.-MCEL control		1
	3	I/O-extended-logout control	I/O extended logout	0
	4	Recovery subclass mask	Machine-check handling	0
	5	Degradation subclass mask		0
	6	External damage subclass mask		1
	7	Warning subclass mask		0
	8	Asynch.-MCEL control		0
	9	Asynch.- fixed-log control		0
	12	ASN-translation control	Dual address space	0
	20-31	ASN-first-table origin		0
15	8-28	MCEL address	Machine-check handling	512

*Value after initial CPU reset.

VECTOR-STATUS REGISTER

0000 0000 0000 000	M	VCT	VIX	VIU	VCH
0	15	16	32	48	56 63

15 (M) Vector-mask-mode bit
16-31 (VCT) Vector count
32-47 (VIX) Vector interruption index
48-55 (VIU) Vector in-use bits
56-63 (VCH) Vector change bits

PROGRAM-STATUS WORD (EC Mode)

0	R	000	T	I	E	PSW key	C	M	W	P	S	0	CC	Program mask	0000 0000

0 8 12 18 20 24 31

0000 0000	Instruction address

32 40 63

1 (R) Program-event-recording mask
5 (T = 1) DAT mode
6 (I) Input/output mask
7 (E) External mask
12 (C = 1) Extended-control mode
13 (M) Machine-check mask
14 (W = 1) Wait state
15 (P = 1) Problem state
16 (S = 1) Secondary-space mode
18-19 (CC) Condition code
20 Fixed-point-overflow mask
21 Decimal-overflow mask
22 Exponent-underflow mask
23 Significance mask

PROGRAM-STATUS WORD (BC Mode)

Channel masks	E	PSW key	C	M	W	P	Interruption code

0 7 8 12 16 31

ILC	CC	Program mask	Instruction address

32 34 36 40 63

0-5 Channel 0 to 5 masks
6 Mask for channel 6 and up
7 (E) External mask
12 (C = 0) Basic-control mode
13 (M) Machine-check mask
14 (W = 1) Wait state
15 (P = 1) Problem state
32-33 (ILC) Instruction-length code
34-35 (CC) Condition code
36 Fixed-point-overflow mask
37 Decimal-overflow mask
38 Exponent-underflow mask
39 Significance mask

EXTERNAL-INTERRUPTION CODES

For EC mode, at real storage address 134-135 (hex 86-87)
For BC mode, at real storage address 26-27 (hex 1A-1B)

Code (binary)	Condition	Code (binary)	Condition
00000000 1eeceeee	Interval timer	00010010 00000000	Malfunction alert
00000000 e1eeeeee	Interrupt key	00010010 00000001	Emergency signal
00000000 ee1eeeee	External sig 2	00010010 00000010	External call
00000000 eee1eeee	External sig 3	00010000 00000011	TOD-clock-sync check
00000000 eeee1eee	External sig 4	00010000 00000100	Clock comparator
00000000 eeeee1ee	External sig 5	00010000 00000101	CPU timer
00000000 eeeeee1e	External sig 6	00100100 00000001	Service signal
00000000 eeeeeee1	External sig 7		

e- if 1, the bit indicates a concurrent external interruption condition.

PROGRAM-INTERRUPTION CODES

For EC mode, at real storage address 142-143 (hex 8E-8F)
For BC mode, at real storage address 42-43 (hex 2A-2B)

Code (hex)	Condition
0001	Operation exception
0002	Privileged-operation exception
0003	Execute exception
0004	Protection exception
0005	Addressing exception
0006	Specification exception
0007	Data exception
nn08*	Fixed-point overflow exception
0009	Fixed-point divide exception
000A	Decimal-overflow exception
000B	Decimal-divide exception
nn0C*	Exponent-overflow exception
nn0D*	Exponent-underflow exception
nn0E*	Significance exception
nn0F*	Floating-point divide exception
0010	Segment-translation exception
00i1	Page-translation exception
0012	Translation-specification exception
0013	Special-operation exception
0017	ASN-translation specification exception
0019	Vector-operation exception
001C	Space-switch event
nn1E*	Unnormalized-operand exception
001F	PC-translation specification exception
0020	AFX-translation exception
0021	ASX-translation exception
0022	LX-translation exception
0023	EX-translation exception
0024	Primary-authority exception
0025	Secondary-authority exception
0040	Monitor event
0080	PER event (code may be combined with another code)

*Use the Exception-Extension Code table below for bits 0-7 (nn) of the program-interruption code.

EXCEPTION-EXTENSION CODE

a	v	w	w	r	r	r	r

0 7

Bit **Meaning**

0(a) Arithmetic-partial-completion bit
  0 Completion or suppression of instruction and bits 1-7 of the exception-extension code are also zero
  1 Partial completion of vector instruction
1(v) Arithmetic-result location
  0 Scalar register
  1 Vector register
2-3(ww) Arithmetic-result width
  01 4-byte result
  10 8-byte result
4-7(rrrr) Register number of result designated by the interrupted instruction

DYNAMIC ADDRESS TRANSLATION

Dynamic-Address-Translation Format

			Virtual -Address Fields			
Cntl Reg 0 Bits 8 - 12	Segment Size	Page Size		Segment Index	Page Index	Byte Index
0 1 0 0 0	64K	2K	Bits	8-15	16-20	21-31
0 1 0 1 0	1M	2K	0-7	8-11	12-20	21-31
1 0 0 0 0	64K	4K	are	8-15	16-19	20-31
1 0 0 1 0	1M	4K	ignored	8-11	12-19	20-31

Any other combination of bits 8-12 of control register 0 is invalid for translation. 1M-byte segments are not provided on some models; 2K-byte pages are not provided on some models.

Segment-Table Entry

PT length	0000*	Page-table origin	P	C	I
0	4	8	29	30	31

29 (P) Segment-protection bit.
30 (C) Common-segment bit
31 (I) Segment-invalid bit
*Normally zeros; ignored on some models.

Page-Table Entry (4K)

Page-frame real address	I	EA	▨
0	12	13	15

12 (I) Page- invalid bit
13-14 (EA) Extended-address bits

Page-Table Entry (2K)

Page-frame real address	I	0	▨
0	13	14	15

13 (I) Page-invalid bit

TRANSLATION-EXCEPTION IDENTIFICATION

At real storage location 144-147 (hex 90-93)

Interruption Code	Format of the Information Stored
0010 (4K pg)	0 secondary address, 1-7 zeros, 8-19 address, 20-31 unpredictable
0010 (2K pg)	0 secondary address, 1-7 zeros, 8-20 address, 21-31 unpredictable
0011 (4K pg)	0 secondary address, 1-7 zeros, 8-19 address, 20-31 unpredictable
0011 (2K pg)	0 secondary address, 1-7 zeros, 8-20 address, 21-31 unpredictable
001C	0 old space-switch-event control, 1-15 zeros, 16-31 old PASN
0020	0-15 zeros, 16-31 address-space number
0021	0-15 zeros, 16-31 address-space number
0022	0-11 zeros, 12-31 program-call number
0023	0-11 zeros, 12-31 program call number
0024	0-15 zeros, 16-31 address-space number
0025	0-15 zeros, 16-31 address-space number

DUAL-ADDRESS-SPACE CONTROL

Program-Call Number

▨	Linkage index	Entry index	
0	12	24	31

Linkage-Table Entry

I	000 0000	Entry-table origin	ETL	
0	1	8	26	31

0 (I) LX-invalid bit
26-31 (ETL) Entry-table length

Entry-Table Entry

Authorization key mask	ASN	0000 0000	Entry instruction address	P
0	16	32	40	63

Entry parameter	Entry key mask	▨	
64	96	112	127

63(P) Entry problem state

ASN-First-Table Entry

I	000 0000	ASN-second-table origin	0000	
0	1	8	28	31

0 (I) AFX-invalid bit

ASN-Second-Table Entry

I	000 0000	Authority-table origin	00	Authorization index	Authority-table length	0000	
0	1	8	30	32	48	60	63

STL	Segment-table origin	▨	X	V	000 0000	Linkage-table origin	LTL	
64	72	90	95	96	97	104	121	127

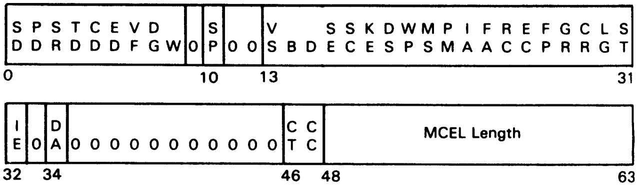
0 (I) ASX-invalid bit
64-71 (STL) Segment-table length
95 (X) Space-switch-event bit
96 (V) Subsystem-linkage control
121-127 (LTL) Linkage-table length

Trace-Table-Entry Header

Current-entry control	First-entry control	Last-entry control	
0	32	64	95

MACHINE-CHECK INTERRUPTION CODE

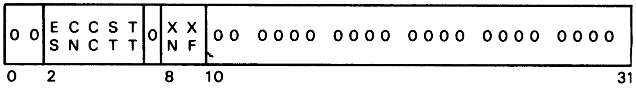
At real storage address 232-239 (hex E8-EF)


SD PD SR TD CD ED VF DG W 0 SP 0 0 VS B D SE SC KE DS WP MS PM IA FA RC EC FP GR CR LG ST
0 10 13 31
IE 0 DA 0 0 0 0 0 0 0 0 0 0 0 CT CC MCEL Length
32 34 46 48 63

Bit	Meaning
0	(SD) System damage
1	(PD) Instruction-processing damage
2	(SR) System recovery
3	(TD) Interval-timer damage
4	(CD) Timing-facility damage
5	(ED) External damage
6	(VF) Vector-facility failure
7	(DG) Degradation
8	(W) Warning
10	(SP) Service-processor damage
13	(VS) Vector-facility source
14	(B) Backed up
15	(D) Delayed
16	(SE) Storage error uncorrected
17	(SC) Storage error corrected
18	(KE) Storage-key error uncorrected
19	(DS) Storage degradation
20	(WP) PSW-CMWP validity
21	(MS) PSW mask and key validity
22	(PM) PSW program-mask and condition-code validity
23	(IA) PSW-instruction-address validity
24	(FA) Failing-storage-address validity
25	(RC) Region-code validity
26	(EC) External-damage-code validity
27	(FP) Floating-point-register validity
28	(GR) General-register validity
29	(CR) Control-register validity
30	(LG) Logout validity
31	(ST) Storage logical validity
32	(IE) Indirect storage error
34	(DA) Delayed access exception
46	(CT) CPU-timer validity
47	(CC) Clock-comparator validity
48-63	Machine-check-extended-logout (MCEL) length

EXTERNAL-DAMAGE CODE

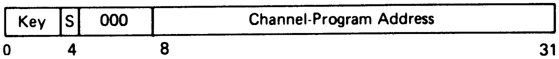
At real storage address 244-247 (hex F4-F7)



Bit	Meaning
2	(ES) External secondary report
3	(CN) Channel not operational
4	(CC) Channel-control failure
5	(ST) I/O-instruction timeout
6	(TT) I/O-interruption timeout
8	(XN) Expanded storage not operational
9	(XF) Expanded storage control failure

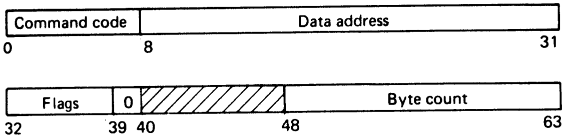
CHANNEL-ADDRESS WORD

At real storage address 72-75 (hex 48-4B)



4 (S) Suspend-control bit

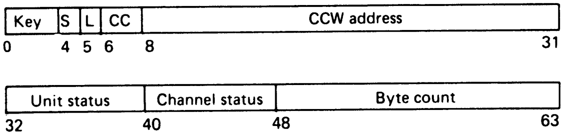
CHANNEL-COMMAND WORD



CD — bit 32 (80) causes use of data-address portion of next CCW.
CC — bit 33 (40) causes use of command code and data address of next CCW.
SLI — bit 34 (20) causes suppression of possible incorrect-length indication.
Skip — bit 35 (10) suppresses transfer of information to main storage.
PCI — bit 36 (08) causes a channel-program-controlled interruption.
IDA — bit 37 (04) causes bits 8-31 of CCW to specify location of first IDAW.
Suspend — bit 38 (02 causes suspension before execution of this CCW.

CHANNEL-STATUS WORD

At real storage address 64-71 (hex 40-47)

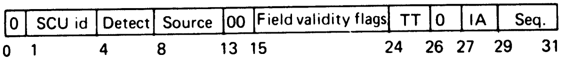


4 Suspended (only in CSW stored by PCI)
5 Logout pending
6-7 Deferred condition code
32 (80) Attention
33 (40) Status modifier
34 (20) Control-unit end
35 (10) Busy
36 (08) Channel end
37 (04) Device end
38 (02) Unit check
39 (01) Unit exception

40 (80) Program-controlled interruption
41 (40) Incorrect length
42 (20) Program check
43 (10) Protection check
44 (08) Channel-data check
45 (04) Channel-control check
46 (02) Interface-control check
47 (01) Chaining check
48-63 Residual byte count for the last CCW used

LIMITED CHANNEL LOGOUT

At real storage address 176-179 (hex B0-BC)



4 CPU
5 Channel
6 Main-storage control
7 Main storage
8 CPU
9 Channel
10 Main-storage control
11 Main storage
12 Control unit
15 Full channel logout
16-18 Reserved (000)
19 Sequence code

20 Unit status
21 CCW address and key
22 Channel address
23 Device address
24-25 Type of termination
00 Interface disconnect
01 Stop, stack or normal
10 Selective reset
11 System reset
27 (I) Interface inoperative
28 (A) I/O-error alert
29-31 Sequence code

I/O COMMAND CODES

Standard Command-Code Assignments (CCW bits 0-7)

xxxx	0000	Invalid Command	mmmm	0100	Sense
mmmm	mm01	Write	0000	0100	– Basic Sense
mmmm	mm10	Read	1110	0100	– Sense ID
0000	0010	–Read IPL	xxxx	1000	Transfer in Channel
mmmm	mm11	Control	mmmm	1100	Read Backward
0000	0011	– Control No Operation			

x — Bit ignored
m — Modifier bit for specific type of I/O device

Standard Meanings of Bits of First Sense Byte

Bit	Designation	Bit	Designation
0	Command reject	4	Data check
1	Intervention required	5	Overrun
2	Bus-out check	6	(Device—dependent)
3	Equipment check	7	(Device—dependent)

Console Printer Channel Commands

Write, No Carrier Return	01	Sense	04
Write, Auto Carrier Return	09	Audible Alarm	0B
Read Inquiry	0A	No Operation	03

Card Reader and Card Punch Channel Commands

3504, 3505 Card Readers/3525 Card Punch (GA21-9124)

Channel Command	Binary	Bit Meanings	
Sense	0000 0100	SS	Stacker
Feed, Select Stacker	SS10 F011	00	1
Read Only*	11D0 F010	01/10	2
Diagnostic Read (invalid for 3504)	1101 0010		
Read, Feed, Select Stacker*	SSD0 F010	F	Format Mode
Write RCE Format*	0001 0001	0	Unformatted
		1	Formatted
3504, 3505 only			
Write OMR Format†	0011 0001	D	Data Mode
		0	1—EBCDIC
3525 only		1	2—Card image
Write, Feed, Select Stacker	SSD0 0001		
Print Line*	LLLL L101	L	Line Position
		(5-bit binary value)	

*Special feature on 3525.

†Special feature.

I/O COMMAND CODES (Cont'd)

Printer Channel Commands

COMMANDS VALID FOR ALL PRINTERS (Except 3800-3 when in Page Mode)	
No Operation	03
Space 1 Line Immediate	0B
Space 2 Lines Immediate	13
Space 3 Lines Immediate	1B
Block Data Check	73
Allow Data Check	7B
Skip to Channel 1 Immediate	8B
Skip to Channel 2 Immediate	93
Skip to Channel 3 Immediate	9B
Skip to Channel 4 Immediate	A3
Skip to Channel 5 Immediate	AB
Skip to Channel 6 Immediate	B3
Skip to Channel 7 Immediate	BB
Skip to Channel 8 Immediate	C3
Skip to Channel 9 Immediate	CB
Skip to Channel 10 Immediate	D3
Skip to Channel 11 Immediate	DB
Skip to Channel 12 Immediate	E3
Write Without Spacing	01
Write and Space 1 Line	09
Write and Space 2 Lines	11
Write and Space 3 Lines	19
Write and Skip to Channel 1	89
Write and Skip to Channel 2	91
Write and Skip to Channel 3	99
Write and Skip to Channel 4	A1
Write and Skip to Channel 5	A9
Write and Skip to Channel 6	B1
Write and Skip to Channel 7	B9
Write and Skip to Channel 8	C1
Write and Skip to Channel 9	C9
Write and Skip to Channel 10	D1
Write and Skip to Channel 11	D9
Write and Skip to Channel 12	E1
Basic Sense	04

3800-3 PAGE MODE COMMANDS (See Note X)	
No Operation	03
Load Font Index	0F
Load Font Control	1F
Load Font	2F
Execute Order Any State	33
Load Font Equivalence	3F
Delete Font	4F
Begin Page Segment	5F
Delete Page Segment	6F
Include Page Segment	7F
Execute Order Home State	8F
Set Home State	97
Load Copy Control	9F
Begin Page	AF
End Page	BF
Load Page Description	CF
Begin Overlay	DF
Delete Overlay	EF
Write Factored Text Control	0D
Write Text	2D
Write Image Control	3D
Write Image	4D
End	5D
Load Page Position	6D
Basic Sense	04
Sense Intermediate Buffer	14
Sense Error Log	24
Sense ID	E4

3800-1 Reference: GA26-1635
3800-3 Reference: GA32-0050

Note X: Other 3800-3 commands are rejected with command retry; the retry will succeed because Page Mode will have been reset.

Note Y: For 3800-3 only, the Set Home State (97) command will be rejected with command retry; the retry will succeed because Page Mode will have been set.

IMPACT PRINTERS – ADDITIONAL COMMANDS

Printer	Column	Reference
1403-N1	A	GA24-3312
3203-1, -2	B	GA33-1515
3203-4	C	GA33-1515
3203-5	C	GA33-1529
3211	C	GA24-3543
4248 <3211 mode>	C	GA24-3927
3262-1, -11	D	GA24-3733
3262-5 <3262-1 mode>	D	GA24-3936
4245-1	D	GA33-1541
4245-12, -20	D	GA33-1579
3262-5 <4248 mode>	E	GA24-3936
4248 <native mode>	E	GA24-3927

Use column A, B, C, D, or E.		A	B	C	D	E
Unfold	23	.	.	X	X	X
Execute Order	33	.	.	.	.	X
Fold	43	.	.	X	X	X
Advance to End of Sheet	5B	.	X	.	.	.
Load Forms Control Buffer	63	.	X	X	X	X
Raise Cover	6B	.	.	1	2	
Signal Attention	6B	.	.			3
Skip to Channel 0 Immediate	83	.	.	4	.	2
Clear Printer	87	.	.	.	X	X
UCS Gate Load	EB	X	2	.	.	.
Load UCS Buffer and Fold	F3	X	X	.	.	
Verify Band ID	F3			.	.	X
Load UCS Buffer (No Fold)	FB	X	X	X	X	
Verify Band ID	FB					X
Release CU and Device	14	5	.	.	.	
Sense Intermediate Buffer	14		.	.	.	X
Release CU, Reserve Device	34	5	.	.	.	.
Reserve CU, Release Device	54	5	.	.	.	.
Reserve CU and Device	74	5	.	.	.	.
Release Device	94	5	.	.	.	.
Reserve Device	B4	5	.	.	.	.
Release CU	D4	5	.	.	.	.
Sense ID	E4	.	X	.	X	X
Reserve CU	F4	5	.	.	.	.
Read Band ID	0A	.	.			X
Diagnostic Read PLB	02	X	.	X	6	2
Diagnostic Write	05	7	.	8	6	2
Diagnostic Check Read	06	X	.	X	X	2
Diagnostic Gate	07	.	.	X	X	2
Diagnostic Read UCS Buffer	0A	.	.	X	X	
Diagnostic Read FCB	12	.	.	X	X	X

X = Valid; . = Invalid; Blank = N/A
1 = No action occurs (except 3211).
2 = No action occurs.
3 = No action occurs (except 4248).
4 = 3211 only (no action occurs on 4248 <3211 mode> and 3203-4).
5 = Two-channel switch feature only.
6 = No action occurs (except 4245).
7 = 1403-N1 also uses command codes 0D, 15, 1D, 8D, 95, 9D, A5, AD, B5, BD, C5, CD, D5, DD, and E5.
8 = 3211 and 4248 <3211 mode> only.

3800-1, -3 – ADDITIONAL COMMANDS (Except 3800-3 when in Page Mode; see Note Y)	
End of Transmission	07
Mark Form	17
Load Copy Number	23
Execute Order Any State	33
Initialize Printer	37
Load Forms Overlay Seq Control	43
Select Translate Table 0	47
Load Writable Char Gen Module	53
Select Translate Table 1	57
Load Forms Control Buffer	63
Select Translate Table 2	67
Select Translate Table 3	77
Load Translate Table	83
Clear Printer	87
Load Graphic Char Modification	25
Load Copy Modification	35
Sense Intermediate Buffer	14
Sense Error Log	24
Sense ID	E4

Direct Access Storage Devices

Use this chart to find the proper column in the DASD Channel Commands table and to find order numbers for DASD reference manuals. See DASD manuals for the restrictions and details of operations.

	Count/Key/Data Devices							FBA Devices		
Controller	2305	3330 3333	3340 3344	3350	3375	3380 -0-A	3380 -D-E	3310	3370	Controller Manual
DASD-A1								col6		GA26-1660
DASD-A4			col2							GA33-1526
DASD-A6			col2							GA33-1566
DASD-A7									col6	GA33-1539
DDA-30		col2								GA33-1510
DDA-40			col2							GA33-1506
IFA		col2	col2							GA24-3632
ISC		col2	col2	col2						GA26-1620
ISC-SA		col2		col2						GA32-0036
2835	col1									GA26-1589
3830-1		*col2								GA26-1592
3830-2		col2	col2	col2						GA26-1617
3830-3		col2		col2						GA32-0036
3880-1		col2	col2	col2	col4				col6	GA26-1661
3880-2		col2	col2	col2	col4	col4			col6	GA26-1661
3880-3						col4	col4			GA26-1661
3880-4					col4				col6	GA26-1661
3880-11 (ND)		col2		col2						GA32-0061
3880-11 (PD)				col2						GA32-0061
3880-11 (PP)				col3						GA32-0061
3880-13						col5				GA32-0067
3880-21 (PD)				col2						GA32-0081
3880-21 (PP)				col3						GA32-0081
3880-23						*col5	col5			GA32-0083
Device Manual	GA26 1589	GA26 1615	GA26 1619	GA26 1638	GA26 1666	GA26 4193	GA26 4193	GA26 1660	GA26 1657	

DASD-A1	=	4321/4331/4361 DASD Adapter for 3310
DASD-A4	=	4321/4331 DASD Adapter for 3340/3344
DASD-A6	=	4361 DASD Adapter for 3340/3344
DASD-A7	=	4321/4331/4361 DASD Adapter for 3370
DDA-30	=	S/370 125-0, -2 3330/3333 Direct Disk Attachment
DDA-40	=	S/370 115-0, -2, 125-0, -2 3340/3344 Direct Disk Attachment
IFA	=	S/370 135, 135-3, 138 Integrated File Adapter
ISC	=	Integrated Storage Controller
ISC-SA	=	Integrated Storage Controller with Staging Adapter
ND	=	Nonpaging director
PD	=	Paging director, direct mode
PP	=	Paging director, paging mode
3380-0-A	=	3380 Direct Access Storage Models AA4, A04, and B04
3380-D-E	=	3380 Direct Access Storage Models AD4, AE4, BD4, and BE4
*	=	3333 does not attach to 3830-1, nor does 3380-A04 to 3880-23

DASD Channel Commands

Channel Command	Hex Code	2305	3330 3340 3350	Page Swap 3350	3375 3380	Data Cache 3380	FBA 3310 3370	Typical Transfer Count
		1	2	3	4	5	6	
Control								
No Operation	03	X	X	X	X	X	X	None
Seek	07	X	X	X	X	X		6
Seek Cylinder	0B	X	X	X	X	X		6
Space Count	0F	X	X		X	X		3
Recalibrate (No-Op on 2305-1, -2)	13	X	X		X	X		None
Restore (executed as No-Op)	17	X	X		X	X		None
Seek Head	1B	X	X	X	X	X		6
Set File Mask	1F	X	X		X	X		1
Set Sector (3340 RPS is optional)	23	X	X	X	X	X		1
Vary Sensing	27	X						1
Orient (No-Op on 2305-2)	2B	X						None
Set High Performance Storage Limits	3B					(a)		10
Locate	43						X	8
Locate Record	47				(b)	(c)		16
Suspend Multipath Reconnection	5B				(d)	X		None
Define Extent	63				(b)	X	X	16
Set Subsystem Mode	87		(e)			X		2
Set Paging Parameters	8B			X				10
Discard Block	8F			X				2+(5 × n)
Set Path Group ID	AF				(d)	X		12
Search								
Search Key Equal (*A9)	29	X	X		X	X		KL
Search ID Equal (*B1)	31	X	X	X	X	X		5
Search Home Address Equal (*B9)	39	X	X		X	X		4
Search Key High (*C9)	49	X	X		X	X		KL
Search ID High (*D1)	51	X	X		X	X		5
Search Key Equal or High (*E9)	69	X	X		X	X		KL
Search ID Equal or High (*F1)	71	X	X		X	X		5
Read								
Read Initial Program Load	02	X	X		X	X	X	DL or 512
Read Data (*86)	06	X	X	X	X	X		DL
Read Key & Data (*8E)	0E	X	X		X	X		KL+DL
Read Count (*92)	12	X	X		X	X		8
Read Record Zero (*96)	16	X	X		X	X		8+KL+DL
Read Home Address (*9A)	1A	X	X		X	X		5
Read Count Key & Data (*9E)	1E	X	X		X	X		8+KL+DL
Read Sector (3340 RPS is optional)	22	X	X		X	X		1
Read	42						X	512 × n
Read Multiple Count Key & Data	5E		(f,g)		X	X		n × (8+KL+DL)
Write								
Write Special Count Key & Data	01	X	X					8+KL+DL
Write Data	05	X	X	X	X	X		DL
Write Key & Data	0D	X	X		X	X		KL+DL
Erase	11	X	X		X	X		8+KL+DL
Write Record Zero	15	X	X		X	X		8+KL+DL
Write Home Address	19	X	X		X	X		5, 7, or 11
Write Count Key & Data	1D	X	X		X	X		8+KL+DL
Write	41						X	512 × n
Write Update Data	85				(b)	(c)		DL
Write Update Key & Data	8D				(b)	(c)		KL+DL
Write Count Key & Data Next Track	9D				(b)	(c)		8+KL+DL
Sense								
Basic Sense	04	X	X	X	X	X	X	24
Unconditional Reserve (h,j,k)	14		(m,p)		(d,m,p)	X	(m,p)	24
Read Buffered Log	24	X						128
Sense Path Group ID	34				(d)	X		12
Sense Subsystem Status	54		(q)			X		40
Read Device Characteristics	64					(c)	X	32
Sense Subsystem Counts	74		(q)			X		80
Device Release (g,j)	94	(p)	(m,p)		(d,m,p)	X	(m,p)	24
Read and Reset Buffered Log (g)	A4		X		X	X	X	24 or 32
Device Reserve (g,j)	B4	(p)	(m,p)		(d,m,p)	X	(m,p)	24
Sense ID (f,g)	E4		X	X	X	X	X	7
Diagnostic								
Diagnostic Write Home Address	09				X	X		27 or 28
Diagnostic Read Home Address	0A				X	X		27 or 28
Diagnostic Sense # (k)	44	X	(r)					16 or 512
Diagnostic Load (k)	53	X	(r)					1
Diagnostic Write (k)	73	X	(r)					8 or 512
Diagnostic Sense/Read	C4		(s)		X	X	X	Variable
Diagnostic Control	F3		(t)		(t)	X	X	4+n
		1	2	3	4	5	6	

a Valid only for 3880-13

b Speed-matching-buffer feature

c Valid only for 3880-23

d Dynamic path selection (only valid on 3380-AA4, -AD4, -AE4 strings)

e Valid only for 3880-21

f Not valid for 3330/3333 on ISC-SA or 3830-1; 3830-2, -3, DDA-40, IFA, and ISC require 3344/3350 microcode

g Not valid on DDA-30

h Not valid on IFA, ISC-SA, or 3830-1; not valid on 3330/3333, 3340/3344

j Executed as Basic Sense on DASD-A1, -A4, -A6, -A7 if no string-switch (for Unconditional Reserve, see note g)

k Not valid on DDA-30, -40, DASD-A4, -A6

m String-switching feature

p Channel-switching feature

q Valid only for 3880-11 paging director and 3880-21

r Not valid on 3880-21

s Valid only for 3880-1, -2, -11, -21

t Valid only for 3330/3350 on 3880-1, -2, and for 3380 on 3880-2, -3 without 3380-speed-matching-buffer feature

* Multitrack command codes (standard)

Also called "Read Diagnostic Status 1"

I/O COMMAND CODES (Cont'd)

Magnetic-Tape Channel Commands

Channel Command	Hex Code	3410 3411	3420-3 3420-5 3420-7	3420-4 3420-6 3420-8	3422 3430	3480	8809
No Operation	03	X	X	X	X	X	X
Rewind	07	X	X	X	X	X	X
Rewind Unload	0F	X	X	X	X	X	X
Modeset-1 (200/Odd/DC)	→ 13	(a)	(b)	(c)		(c)	—
Set Long Gap	→ 13	—	—	—		—	X
Erase Gap	17	X	X	X	X	X	X
Request Track-In-Error	1B	X	X	X	(d)		
Write Tape Mark	1F	X	X	X	X	X	X
Modeset-1 (200/Even/Normal)	→ 23	(a)	(b)	(c)		(c)	—
Set Normal Gap	→ 23	—	—	—		—	X
Backspace Block	27	X	X	X	X	X	X
Modeset-1 (200/Even/TR)	2B	(a)	(b)	(c)		(c)	
Backspace File	2F	X	X	X	X	X	X
Modeset-1 (200/Odd/Normal)	→ 33	(a)	(b)	(c)		(c)	—
Set High Speed/Normal Gap	→ 33	—	—	—		—	X
Forward Space Block	37	X	X	X	X	X	X
Modeset-1 (200/Odd/TR)	3B	(a)	(b)	(c)		(c)	
Forward Space File	3F	X	X	X	X	X	X
Synchronize	43					X	
Locate Block	4F					X	
Modeset-1 (556/Odd/DC)	→ 53	(a)	(a)	(c)		(c)	—
Set Low Speed/Long Gap	→ 53	—	—	—		—	X
Suspend Multipath Reconnection	5B					(c)	
Modeset-1 (556/Even/Normal)	→ 63	(a)	(a)	(c)		(c)	—
Set Low Speed/Normal Gap	→ 63	—	—	—		—	X
Modeset-1 (556/Even/TR)	6B	(a)	(a)	(c)		(c)	
Modeset-1 (556/Odd/Normal)	73	(a)	(a)	(c)		(c)	
Modeset-1 (556/Odd/TR)	7B	(a)	(a)	(c)		(c)	
Set Low Speed	83						X
Modeset-1 (800/Odd/DC)	→ 93	(a)	(a)	(c)		(c)	—
Set High Speed/Long Gap	→ 93	—	—	—		—	X
Data Security Erase	97	X	X	X	X	X	X
Load Display	9F					X	
Modeset-1 (800/Even/Normal)	A3	(a)	(a)	(c)		(c)	
Modeset-1 (800/Even/TR)	AB	(a)	(a)	(c)		(c)	
Set Path Group ID	AF					X	
Modeset-1 (800/Odd/Normal)	B3	(a)	(a)	(c)		(c)	
Assign	B7					X	
Modeset-1 (800/Odd/TR)	BB	(a)	(a)	(c)		(c)	
Modeset-2 (1600 bpi PE)	→ C3	(e)	(e)	(f)	X	—	(c)
Set Tape-Write-Immediate	→ C3	—	—	—	—	X	—
Unassign	C7					X	
Modeset-2 (800 bpi NRZI)	CB	(e)	(e)	(c)		(c)	
Modeset-2 (6250 bpi GCR)	D3			(f)	X	(c)	
Mode Set	DB					X	
Control Access	→ E3					X	—
Set High Speed	→ E3					—	X
Write	01	X	X	X	X	X	X
Read	02	X	X	X	X	X	X
Read Buffer	12					X	
Read Block ID	22					X	
Read Backward	0C	X	X	X	X	X	
Basic Sense	04	X	X	X	X	X	X
Read Buffered Log	24					X	
Sense Path Group ID	34					X	
Read/Reset Buffered Log	A4						X
Release	D4		(g)	(g)	(g)		
Sense ID	E4				X	X	X
Reserve	F4		(g)	(g)	(g)		
Diagnostic Mode Set	0B		X	X			
Set Diagnose	4B		X	X	(d)		
Loop Write-To-Read	8B		X	X	X		X

Notes:

a No action occurs unless 7-track feature is installed.
b No action occurs unless 7-track feature is installed; if present, density set is 200 bpi by 3803-2 Tape Control, 556 bpi by 3803-1.
c Valid command, but no action occurs.
d Invalid command for 3422.
e No action occurs unless 800 bpi density feature is installed.
f No action occurs unless 1600 bpi density feature is installed.
g Requires two-channel switch feature; invalid for 3430.

Where arrows appear, the meaning of the hex code depends on the machine type; hyphens signify that the alternative meaning is used.

Modeset-1 command (for 7-track drives): density (200, 556, 800 bpi)/parity (even, odd)/mode (Normal, DC = data converter, TR = translator). Modeset-2 command (for 9-track drives): density (800, 1600, 6250 bpi).

Sources:

3410/3411 (GA32-0022)	3422 (GA32-0089)	8809 (GA26-1659)
3420-3, -5, -7 (GA32-0020)	3430 (GA32-0076)	
3420-4, -6, -8 (GA32-0021)	3480 (GA32-0042)	

CODE ASSIGNMENTS

Two-Character BSC Data Link Controls

Function	EBCDIC	ASCII
ACK-0	DLE,X'70'	DLE,0
ACK-1	DLE,X'61'	DLE,1
WACK	DLE,X'68'	DLE,;
RVI	DLE,X'7C'	DLE,<

Commonly Used Editing Pattern Characters

Code (hex)	Meaning
20	digit selector
21	start of significance
22	field separator
40	blank
4B	period

Code (hex)	Meaning
5B	dollar sign
5C	asterisk
6B	comma
C3D9	CR (credit)
C4C2	DB (debit)

ANSI-Defined Printer Control Characters

(**A** in **RECFM** field of DCB)

Code	Action before Printing Record
blank	Space 1 line
0	Space 2 lines
–	Space 3 lines
+	Suppress space
1	Skip to line 1 on new page

Control Character Representations

ACK	Acknowledge	IT	Indent Tab
BEL	Bell	IUS	Interchange Unit Separator
BS	Backspace	ITB	Intermediate Transmission Block
BYP	Bypass	LF	Line Feed
CAN	Cancel	MFA	Modify Field Attribute
CR	Carriage Return	NAK	Negative Acknowledge
CSP	Control Sequence Prefix	NBS	Numeric Backspace
CU1	Customer Use 1	NL	New Line
CU3	Customer Use 3	NUL	Null
DC1	Device Control 1	POC	Program-Operator Communication
DC2	Device Control 2	PP	Presentation Position
DC3	Device Control 3	RES	Restore
DC4	Device Control 4	RFF	Required Form Feed
DEL	Delete	RNL	Required New Line
DLE	Data Link Escape	RPT	Repeat
DS	Digit Select	SA	Set Attribute
EM	End of Medium	SBS	Subscript
ENP	Enable Presentation	SEL	Select
ENQ	Enquiry	SFE	Start Field Extended
EO	Eight Ones	SI	Shift In
EOT	End of Transmission	SM	Set Mode
ESC	Escape	SO	Shift Out
ETB	End of Transmission Block	SOH	Start of Heading
ETX	End of Text	SOS	Start of Significance
FF	Form Feed	SPS	Superscript
FS	Field Separator	STX	Start of Text
GE	Graphic Escape	SUB	Substitute
HT	Horizontal Tab	SW	Switch
IFS	Interchange File Separator	SYN	Synchronous Idle
IGS	Interchange Group Separator	TRN	Transparent
INP	Inhibit Presentation	UBS	Unit Backspace
IR	Index Return	VT	Vertical Tab
IRS	Interchange Record Separator	WUS	Word Underscore

Formatting Character Representations

NSP	Numeric Space	SP	Space
RSP	Required Space	SHY	Syllable Hyphen

CODE ASSIGNMENTS (Cont'd)

Code Tables

Dec.	Hex	Graphics and Controls			7-Track Tape	Card Code	Binary
		BCDIC	EBCDIC	ASCII	BCDIC	EBCDIC	
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		SEL	EOT		12-4-9	0000 0100
5	05		HT	ENQ		12-5-9	0000 0101
6	06		RNL	ACK		12-6-9	0000 0110
7	07		DEL	BEL		12-7-9	0000 0111
8	08		GE	BS		12-8-9	0000 1000
9	09		SPS	HT		12-1-8-9	0000 1001
10	0A		RPT	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		SO	SO		12-6-8-9	0000 1110
15	0F		SI	SI		12-7-8-9	0000 1111
16	10		DLE	DLE		12-11-1-8-9	0001 0000
17	11		DC1	DC1		11-1-9	0001 0001
18	12		DC2	DC2		11-2-9	0001 0010
19	13		DC3	DC3		11-3-9	0001 0011
20	14		RES/ENP	DC4		11-4-9	0001 0100
21	15		NL	NAK		11-5-9	0001 0101
22	16		BS	SYN		11-6-9	0001 0110
23	17		POC	ETB		11-7-9	0001 0111
24	18		CAN	CAN		11-8-9	0001 1000
25	19		EM	EM		11-1-8-9	0001 1001
26	1A		UBS	SUB		11-2-8-9	0001 1010
27	1B		CU1	ESC		11-3-8-9	0001 1011
28	1C		IFS	FS		11-4-8-9	0001 1100
29	1D		IGS	GS		11-5-8-9	0001 1101
30	1E		IRS	RS		11-6-8-9	0001 1110
31	1F		ITB/IUS	US		11-7-8-9	0001 1111
32	20		DS	SP		11-0-1-8-9	0010 0000
33	21		SOS	!		0-1-9	0010 0001
34	22		FS	''		0-2-9	0010 0010
35	23		WUS	#		0-3-9	0010 0011
36	24		BYP/INP	$		0-4-9	0010 0100
37	25		LF	%		0-5-9	0010 0101
38	26		ETB	&		0-6-9	0010 0110
39	27		ESC	'		0-7-9	0010 0111
40	28		SA	(		0-8-9	0010 1000
41	29		SFE	)		0-1-8-9	0010 1001
42	2A		SM/SW	*		0-2-8-9	0010 1010
43	2B		CSP	+		0-3-8-9	0010 1011
44	2C		MFA	,		0-4-8-9	0010 1100
45	2D		ENQ	-		0-5-8-9	0010 1101
46	2E		ACK	.		0-6-8-9	0010 1110
47	2F		BEL	/		0-7-8-9	0010 1111
48	30			0		12-11-0-1-8-9	0011 0000
49	31			1		1-9	0011 0001
50	32		SYN	2		2-9	0011 0010
51	33		IR	3		3-9	0011 0011
52	34		PP	4		4-9	0011 0100
53	35		TRN	5		5-9	0011 0101
54	36		NBS	6		6-9	0011 0110
55	37		EOT	7		7-9	0011 0111
56	38		SBS	8		8-9	0011 1000
57	39		IT	9		1-8-9	0011 1001
58	3A		RFF	:		2-8-9	0011 1010
59	3B		CU3	;		3-8-9	0011 1011
60	3C		DC4	<		4-8-9	0011 1100
61	3D		NAK	=		5-8-9	0011 1101
62	3E			>		6-8-9	0011 1110
63	3F		SUB	?		7-8-9	0011 1111

CODE ASSIGNMENTS (Cont'd)

Code Tables (Cont'd)

Dec.	Hex	Graphics and Controls				7-Track Tape	Card Code	Binary
		BCDIC	EBCDIC(1)		ASCII	BCDIC(2)	EBCDIC	
64	40	SP	SP	SP	@	(3)	no punches	0100 0000
65	41		RSP		A		12-0-1-9	0100 0001
66	42				B		12-0-2-9	0100 0010
67	43				C		12-0-3-9	0100 0011
68	44				D	ı	12-0-4-9	0100 0100
69	45				E		12-0-5-9	0100 0101
70	46				F		12-0-6-9	0100 0110
71	47				G		12-0-7-9	0100 0111
72	48				H		12-0-8-9	0100 1000
73	49				I		12-1-8	0100 1001
74	4A		¢	¢	J		12-2-8	0100 1010
75	4B	.	.	.	K	B A 8 2 1	12-3-8	0100 1011
76	4C	⌑)	<	<	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	<	+	+	N	B A 8 4 2	12-6-8	0100 1110
79	4F	⧧	|	|	O	B A 8 4 2 1	12-7-8	0100 1111
80	50	& +	&	&	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				T		12-11-4-9	0101 0100
85	55				U		12-11-5-9	0101 0101
86	56				V		12-11-6-9	0101 0110
87	57				W		12-11-7-9	0101 0111
88	58				X		12-11-8-9	0101 1000
89	59				Y		11-1-8	0101 1001
90	5A		!	!	Z		11-2-8	0101 1010
91	5B	$	$	$	[	B 8 2 1	11-3-8	0101 1011
92	5C	*	*	*	\	B 8 4	11-4-8	0101 1100
93	5D	]	)	)	]	B 8 4 1	11-5-8	0101 1101
94	5E	;	;	;	^	B 8 4 2	11-6-8	0101 1110
95	5F	Δ	¬	¬	_	B 8 4 2 1	11-7-8	0101 1111
96	60	—	—	—	`	B	11	0110 0000
97	61	/	/	/	a	A 1	0-1	0110 0001
98	62				b		11-0-2-9	0110 0010
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				g		11-0-7-9	0110 0111
104	68				h		11-0-8-9	0110 1000
105	69				i		0-1-8	0110 1001
106	6A		¦		j		12-11	0110 1010
107	6B	,	,	,	k	A 8 2 1	0-3-8	0110 1011
108	6C	%(	%	%	l	A 8 4	0-4-8	0110 1100
109	6D	γ	_	_	m	A 8 4 1	0-5-8	0110 1101
110	6E	\	>	>	n	A 8 4 2	0-6-8	0110 1110
111	6F	⧻	?	?	o	A 8 4 2 1	0-7-8	0110 1111
112	70				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				x		12-11-0-8-9	0111 1000
121	79				y		1-8	0111 1001
122	7A	ƀ	:	:	z	A	2-8	0111 1010
123	7B	# =	#	#	{	8 2 1	3-8	0111 1011
124	7C	@'	@	@	¦	8 4	4-8	0111 1100
125	7D	:	'	'	}	8 4 1	5-8	0111 1101
126	7E	>	=	=	~	8 4 2	6-8	0111 1110
127	7F	√	''	''	DEL	8 4 2 1	7-8	0111 1111

Code Tables (Cont'd)

Dec.	Hex	Graphics and Controls				7-Track Tape	Card Code	Binary
		BCDIC	EBCDIC(1)		ASCII	BCDIC	EBCDIC	
128	80						12-0-1-8	1000 0000
129	81		a	a			12-0-1	1000 0001
130	82		b	b			12-0-2	1000 0010
131	83		c	c			12-0-3	1000 0011
132	84		d	d			12-0-4	1000 0100
133	85		e	e			12-0-5	1000 0101
134	86		f	f			12-0-6	1000 0110
135	87		g	g			12-0-7	1000 0111
136	88		h	h			12-0-8	1000 1000
137	89		i	i			12-0-9	1000 1001
138	8A						12-0-2-8	1000 1010
139	8B			{			12-0-3-8	1000 1011
140	8C			≤			12-0-4-8	1000 1100
141	8D			(See Note			12-0-5-8	1000 1101
142	8E			+ See Note			12-0-6-8	1000 1110
143	8F			+			12-0-7-8	1000 1111
144	90						12-11-1-8	1001 0000
145	91		j	j			12-11-1	1001 0001
146	92		k	k			12-11-2	1001 0010
147	93		l	l			12-11-3	1001 0011
148	94		m	m			12-11-4	1001 0100
149	95		n	n			12-11-5	1001 0101
150	96		o	o			12-11-6	1001 0110
151	97		p	p			12-11-7	1001 0111
152	98		q	q			12-11-8	1001 1000
153	99		r	r			12-11-9	1001 1001
154	9A						12-11-2-8	1001 1010
155	9B			}			12-11-3-8	1001 1011
156	9C			⌑			12-11-4-8	1001 1100
157	9D			) See Note			12-11-5-8	1001 1101
158	9E			±			12-11-6-8	1001 1110
159	9F			■			12-11-7-8	1001 1111
160	A0			‾ See Note			11-0-1-8	1010 0000
161	A1		~	°			11-0-1	1010 0001
162	A2		s	s			11-0-2	1010 0010
163	A3		t	t			11-0-3	1010 0011
164	A4		u	u			11-0-4	1010 0100
165	A5		v	v			11-0-5	1010 0101
166	A6		w	w			11-0-6	1010 0110
167	A7		x	x			11-0-7	1010 0111
168	A8		y	y			11-0-8	1010 1000
169	A9		z	z			11-0-9	1010 1001
170	AA						11-0-2-8	1010 1010
171	AB			└			11-0-3-8	1010 1011
172	AC			┌			11-0-4-8	1010 1100
173	AD			[			11-0-5-8	1010 1101
174	AE			≥			11-0-6-8	1010 1110
175	AF			•			11-0-7-8	1010 1111
176	B0			^{0} See Note			12-11-0-1-8	1011 0000
177	B1			^{1} See Note			12-11-0-1	1011 0001
178	B2			^{2} See Note			12-11-0-2	1011 0010
179	B3			^{3} See Note			12-11-0-3	1011 0011
180	B4			^{4} See Note			12-11-0-4	1011 0100
181	B5			^{5} See Note			12-11-0-5	1011 0101
182	B6			^{6} See Note			12-11-0-6	1011 0110
183	B7			^{7} See Note			12-11-0-7	1011 0111
184	B8			^{8} See Note			12-11-0-8	1011 1000
185	B9			^{9} See Note			12-11-0-9	1011 1001
186	BA						12-11-0-2-8	1011 1010
187	BB			┘			12-11-0-3-8	1011 1011
188	BC			┐			12-11-0-4-8	1011 1100
189	BD			]			12-11-0-5-8	1011 1101
190	BE			≠			12-11-0-6-8	1011 1110
191	BF			—			12-11-0-7-8	1011 1111

Note: This character is an EBCDIC superscript character.

Code Tables (Cont'd)

Dec.	Hex	Graphics and Controls				7-Track Tape	Card Code	Binary
		BCDIC	EBCDIC(1)		ASCII	BCDIC(2)	EBCDIC	
192	C0	?	{			B A 8 2	12-0	1100 0000
193	C1	A	A	A		B A 1	12-1	1100 0001
194	C2	B	B	B		B A 2	12-2	1100 0010
195	C3	C	C	C		B A 2 1	12-3	1100 0011
196	C4	D	D	D		B A 4	12-4	1100 0100
197	C5	E	E	E		B A 4 1	12-5	1100 0101
198	C6	F	F	F		B A 4 2	12-6	1100 0110
199	C7	G	G	G		B A 4 2 1	12-7	1100 0111
200	C8	H	H	H		B A 8	12-8	1100 1000
201	C9	I	I	I		B A 8 1	12-9	1100 1001
202	CA		SHY				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	J	J	J		B 1	11-1	1101 0001
210	D2	K	K	K		B 2	11-2	1101 0010
211	D3	L	L	L		B 2 1	11-3	1101 0011
212	D4	M	M	M		B 4	11-4	1101 0100
213	D5	N	N	N		B 4 1	11-5	1101 0101
214	D6	O	O	O		B 4 2	11-6	1101 0110
215	D7	P	P	P		B 4 2 1	11-7	1101 0111
216	D8	Q	Q	Q		B 8	11-8	1101 1000
217	D9	R	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						12-11-4-8-9	1101 1100
221	DD						12-11-5-8-9	1101 1101
222	DE						12-11-6-8-9	1101 1110
223	DF						12-11-7-8-9	1101 1111
224	E0	ǂ	\			A 8 2	0-2-8	1110 0000
225	E1		NSP				11-0-1-9	1110 0001
226	E2	S	S	S		A 2	0-2	1110 0010
227	E3	T	T	T		A 2 1	0-3	1110 0011
228	E4	U	U	U		A 4	0-4	1110 0100
229	E5	V	V	V		A 4 1	0-5	1110 0101
230	E6	W	W	W		A 4 2	0-6	1110 0110
231	E7	X	X	X		A 4 2 1	0-7	1110 0111
232	E8	Y	Y	Y		A 8	0-8	1110 1000
233	E9	Z	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	0	0	0		8 2	0	1111 0000
241	F1	1	1	1		1	1	1111 0001
242	F2	2	2	2		2	2	1111 0010
243	F3	3	3	3		2 1	3	1111 0011
244	F4	4	4	4		4	4	1111 0100
245	F5	5	5	5		4 1	5	1111 0101
246	F6	6	6	6		4 2	6	1111 0110
247	F7	7	7	7		4 2 1	7	1111 0111
248	F8	8	8	8		8	8	1111 1000
249	F9	9	9	9		8 1	9	1111 1001
250	FA						12-11-0-2-8-9	1111 1010
251	FB						12-11-0-3-8-9	1111 1011
252	FC						12-11-0-4-8-9	1111 1100
253	FD						12-11-0-5-8-9	1111 1101
254	FE						12-11-0-6-8-9	1111 1110
255	FF		EO				12-11-0-7-8-9	1111 1111

1. Two columns of EBCDIC graphics are shown. The first gives IBM standard U.S. bit pattern assignments. The second shows the T-11 and TN text printing chains (120 graphics).
2. Add C (check bit) for odd or even parity as needed, except as noted.
3. For even parity, use CA.

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 decimal: (1) locate the largest decimal value in the table that will fit into the decimal number to be converted, and (2) note its hex equivalent and hex column position. (3) Find the decimal remainder. Repeat the process on this and subsequent remainders.

Note: Decimal, hexadecimal, and binary equivalents of all numbers from 0 to 255 are listed in the code tables.

WORD															
HALFWORD								HALFWORD							
BYTE				BYTE				BYTE				BYTE			
BITS: 0123		4567		0123		4567		0123		4567		0123		4567	
Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal	Hex	Decimal
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	268,435,456	1	16,777,216	1	1,048,576	1	65,536	1	4,096	1	256	1	16	1	1
2	536,870,912	2	33,554,432	2	2,097,152	2	131,072	2	8,192	2	512	2	32	2	2
3	805,306,368	3	50,331,648	3	3,145,728	3	196,608	3	12,288	3	768	3	48	3	3
4	1,073,741,824	4	67,108,864	4	4,194,304	4	262,144	4	16,384	4	1,024	4	64	4	4
5	1,342,177,280	5	83,886,080	5	5,242,880	5	327,680	5	20,480	5	1,280	5	80	5	5
6	1,610,612,736	6	100,663,296	6	6,291,456	6	393,216	6	24,576	6	1,536	6	96	6	6
7	1,879,048,192	7	117,440,512	7	7,340,032	7	458,752	7	28,672	7	1,792	7	112	7	7
8	2,147,483,648	8	134,217,728	8	8,388,608	8	524,288	8	32,768	8	2,048	8	128	8	8
9	2,415,919,104	9	150,994,944	9	9,437,184	9	589,824	9	36,864	9	2,304	9	144	9	9
A	2,684,354,560	A	167,772,160	A	10,485,760	A	655,360	A	40,960	A	2,560	A	160	A	10
B	2,952,790,016	B	184,549,376	B	11,534,336	B	720,896	B	45,056	B	2,816	B	176	B	11
C	3,221,225,472	C	201,326,592	C	12,582,912	C	786,432	C	49,152	C	3,072	C	192	C	12
D	3,489,660,928	D	218,103,808	D	13,631,488	D	851,968	D	53,248	D	3,328	D	208	D	13
E	3,758,096,384	E	234,881,024	E	14,680,064	E	917,504	E	57,344	E	3,584	E	224	E	14
F	4,026,531,840	F	251,658,240	F	15,728,640	F	983,040	F	61,440	F	3,840	F	240	F	15
8		7		6		5		4		3		2		1	

HEXADECIMAL AND DECIMAL CONVERSION (Cont'd)

Powers of 2 and 16

m	*n*	2^m and 16^n
0	0	1
1		2
2		4
3		8
4	1	16
5		32
6		64
7		128
8	2	256
9		512
10		1 024
11		2 048
12	3	4 096
13		8 192
14		16 384
15		32 768
16	4	65 536
17		131 072
18		262 144
19		524 288
20	5	1 048 576
21		2 097 152
22		4 194 304
23		8 388 608
24	6	16 777 216
25		33 554 432
26		67 108 864
27		134 217 728
28	7	268 435 456
29		536 870 912
30		1 073 741 824
31		2 147 483 648

m	*n*	2^m and 16^n
32	8	4 294 967 296
33		8 589 934 592
34		17 179 869 184
35		34 359 738 368
36	9	68 719 476 736
37		137 438 953 472
38		274 877 906 944
39		549 755 813 888
40	10	1 099 511 627 776
41		2 199 023 255 552
42		4 398 046 511 104
43		8 796 093 022 208
44	11	17 592 186 044 416
45		35 184 372 088 832
46		70 368 744 177 664
47		140 737 488 355 328
48	12	281 474 976 710 656
49		562 949 953 421 312
50		1 125 899 906 842 624
51		2 251 799 813 685 248
52	13	4 503 599 627 370 496
53		9 007 199 254 740 992
54		18 014 398 509 481 984
55		36 028 797 018 963 968
56	14	72 057 594 037 927 936
57		144 115 188 075 855 872
58		288 230 376 151 711 744
59		576 460 752 303 423 488
60	15	1 152 921 504 606 846 976
61		2 305 843 009 213 693 952
62		4 611 686 018 427 387 904
63		9 223 372 036 854 775 808

Symbol	Value
K (kilo)	$1,024 = 2^{10}$
M (mega)	$1,048,576 = 2^{20}$
G (giga)	$1,073,741,824 = 2^{30}$

NOTES

NOTES

NOTES

NOTES

NOTES