Appendix

SYSTEM/360 MACHINE INSTRUCTIONS

STANDARD INSTRUCTION SET

NAME	MNEMONIC	TYPE	CODE	OPERANDS (Assembler Format)
* Add	AR	RR	1A	R1, R2
* Add	A	RX	5A	R1, D2 (X2, B2)
* Add Halfword	AH	RX	4A	R1, D2 (X2, B2)
* Add Logical	ALR	RR	1E	R1, R2
* Add Logical	AL	RX	5E	R1, D2 (X2, B2)
* AND	NR	RR	14	R1, R2
* AND	N	RX	54	R1, D2 (X2, B2)
* AND	NI	SI	94	D1 (B1), I2
* AND	NC	SS	D4	D1 (L, B1), D2 (B2)
Branch and Link	BALR	RR	05	R1, R2
Branch and Link	BAL	RX	45	R1, D2 (X2, B2)
Branch on Condition	BCR	RR	07	M1, R2
Branch on Condition	BC	RX	47	M1, D2 (X2, B2)
Branch on Count	BCTR	RR	06	R1, R2
Branch on Count	BCT	RX	46	R1, D2 (X2, B2)
Branch on Index High	BXH	RS	86	R1, R3, D2 (B2)
Branch on Index Low or Equal	BXLE	RS	87	R1, R3, D2 (B2)
* Compare	CR	RR	19	R1, R2
* Compare	C	RX	59	R1, D2 (X2, B2)
* Compare Halfword	CH	RX	49	R1, D2 (X2, B2)
* Compare Logical	CLR	RR	15	R1, R2
* Compare Logical	CL	RX	55	R1, D2 (X2, B2)
* Compare Logical	CLC	SS	D5	D1 (L, B1), D2 (B2)
* Compare Logical	CLI	SI	95	D1 (B1), I2
Convert to Binary	CVB	RX	4F	R1, D2 (X2, B2)
Convert to Decimal	CVD	RX	4E	R1, D2 (X2, B2)
Diagnose		SI	83	
Divide	DR	RR	1D	R1, R2
Divide	D	RX	5D	R1, D2 (X2, B2)
* Exclusive OR	XR	RR	17	R1, R2
* Exclusive OR	X	RX	57	R1, D2 (X2, B2)
* Exclusive OR	XI	SI	97	D1 (B1), I2
* Exclusive OR	XC	SS	D7	D1 (L, B1), D2 (B2)
Execute	EX	RX	44	R1, D2 (X2, B2)
* Halt I/O	HIO	SI	9E	D1 (B1)
Insert Character	IC	RX	43	R1, D2 (X2, B2)
Load	LR	RR	18	R1, R2
Load	L	RX	58	R1, D2 (X2, B2)
Load Address	LA	RX	41	R1, D2 (X2, B2)
* Load and Test	LTR	RR	12	R1, R2
* Load Complement	LCR	RR	13	R1, R2
Load Halfword	LH	RX	48	R1, D2 (X2, B2)
Load Multiple	LM	RS	98	R1, R3, D2 (B2)
* Load Negative	LNR	RR	11	R1, R2
* Load Positive	LPR	RR	10	R1, R2
† Load PSW	LPSW	SI	82	D1 (B1)
Move	MVI	SI	92	D1 (B1), I2
Move	MVC	SS	D2	D1 (L, B1), D2 (B2)
Move Numerics	MVN	SS	D1	D1 (L, B1), D2 (B2)
Move with Offset	MVO	SS	F1	D1 (L1, B1), D2 (L2, B2)
Move Zones	MVZ	SS	D3	D1 (L, B1), D2 (B2)
Multiply	MR	RR	1C	R1, R2
Multiply	M	RX	5C	R1, D2 (X2, B2)
Multiply Halfword	MH	RX	4C	R1, D2 (X2, B2)
* OR	OR	RR	16	R1, R2
* OR	O	RX	56	R1, D2 (X2, B2)
* OR	OI	SI	96	D1 (B1), I2
* OR	OC	SS	D6	D1 (L, B1), D2 (B2)
Pack	PACK	SS	F2	D1 (L1, B1), D2 (L2, B2)
† Set Program Mask	SPM	RR	04	R1
Set System Mask	SSM	SI	80	D1 (B1)
* Shift Left Double	SLDA	RS	8F	R1, D2 (B2)
* Shift Left Single	SLA	RS	8B	R1, D2 (B2)
Shift Left Double Logical	SLDL	RS	8D	R1, D2 (B2)
Shift Left Single Logical	SLL	RS	89	R1, D2 (B2)
* Shift Right Double	SRDA	RS	8E	R1, D2 (B2)
* Shift Right Single	SRA	RS	8A	R1, D2 (B2)
Shift Right Double Logical	SRDL	RS	8C	R1, D2 (B2)
Shift Right Single Logical	SRL	RS	88	R1, D2 (B2)
* Start I/O	SIO	SI	9C	D1 (B1)
Store	ST	RX	50	R1, D2 (X2, B2)
Store Character	STC	RX	42	R1, D2 (X2, B2)
Store Halfword	STH	RX	40	R1, D2 (X2, B2)
Store Multiple	STM	RS	90	R1, R3, D2 (B2)
* Subtract	SR	RR	1B	R1, R2
* Subtract	S	RX	5B	R1, D2 (X2, B2)
* Subtract Halfword	SH	RX	4B	R1, D2 (X2, B2)
* Subtract Logical	SLR	RR	1F	R1, R2
* Subtract Logical	SL	RX	5F	R1, D2 (X2, B2)
Supervisor Call	SVC	RR	0A	I
* Test and Set	TS	SI	93	D1 (B1)
* Test Channel	TCH	SI	9F	D1 (B1)
* Test I/O	TIO	SI	9D	D1 (B1)
* Test Under Mask	TM	SI	91	D1 (B1), I2
Translate	TR	SS	DC	D1 (L, B1), D2 (B2)
* Translate and Test	TRT	SS	DD	D1 (L, B1), D2 (B2)
Unpack	UNPK	SS	F3	D1 (L1, B1), D2 (L2, B2)

DECIMAL FEATURE INSTRUCTIONS

* Add Decimal	AP	SS	FA	D1 (L1, B1), D2 (L2, B2)
* Compare Decimal	CP	SS	F9	D1 (L1, B1), D2 (L2, B2)
Divide Decimal	DP	SS	FD	D1 (L1, B1), D2 (L2, B2)
* Edit	ED	SS	DE	D1 (L, B1), D2 (B2)
* Edit and Mark	EDMK	SS	DF	D1 (L, B1), D2 (B2)
Multiply Decimal	MP	SS	FC	D1 (L1, B1), D2 (L2, B2)
* Subtract Decimal	SP	SS	FB	D1 (L1, B1), D2 (L2, B2)
* Zero and Add	ZAP	SS	F8	D1 (L1, B1), D2 (L2, B2)

DIRECT CONTROL FEATURE INSTRUCTIONS

Read Direct	RDD	SI	85	D1 (B1), I2
Write Direct	WRD	SI	84	D1 (B1), I2

PROTECTION FEATURE INSTRUCTIONS

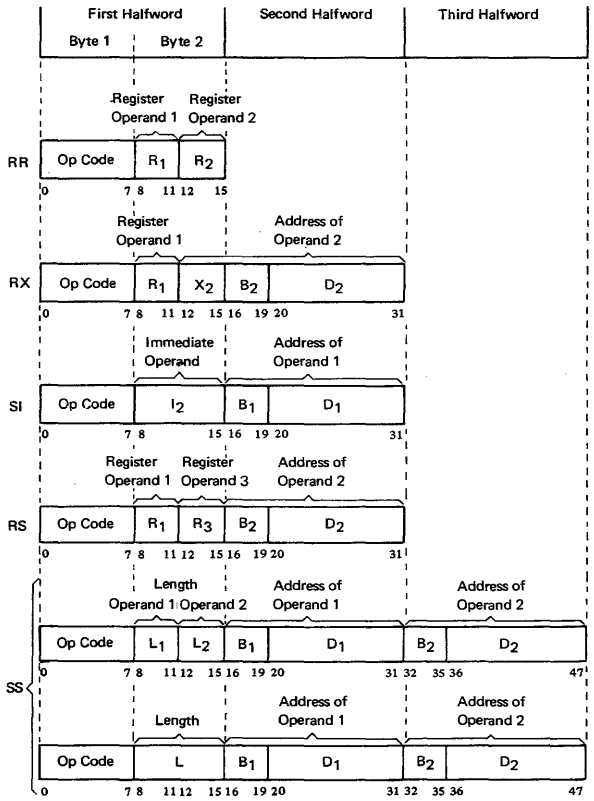
Insert Storage Key	ISK	RR	09	R1, R2
Set Storage Key	SSK	RR	08	R1, R2

Floating-point feature instructions are listed in Chapter 9.

* Condition code is set

† New condition code is loaded

MACHINE FORMAT



CONDITION CODE SETTINGS

Code State	0	1	2	3
Mask Bit Position	8	4	2	1
Fixed-Point Arithmetic				
Add H/F	zero	< zero	> zero	overflow
Add Logical	zero	not zero	zero	not zero
	no carry	no carry	carry	carry
Compare H/F (A:B)	equal	A low	A high	--
Load and Test	zero	< zero	> zero	carry
Load Complement	zero	< zero	> zero	overflow
Load Negative	zero	< zero	--	--
Load Positive	zero	--	> zero	overflow
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 H/F	zero	< zero	> zero	overflow
Subtract Logical	--	not zero	zero	not zero
		no carry	carry	carry
Decimal Arithmetic				
Add Decimal	zero	< zero	> zero	overflow
Compare Decimal (A:B)	equal	A low	A high	--
Subtract Decimal	zero	< zero	> zero	overflow
Zero and Add	zero	< zero	> zero	overflow
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	--	--
Or	zero	not zero	--	--
Test Under Mask	zero	mixed	--	one(s)
Translate and Test	zero	incomplete	complete	--

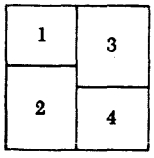
EXTENDED MNEMONIC CODES FOR THE BRANCH ON CONDITION INSTRUCTION

Assembler Code		Meaning	Machine Instruction Generated	
B	D2(X2,B2)	Branch Unconditional	BC	15,D2(X2,B2)
BR	R2	Branch Unconditional (RR format)	BCR	15,R2
NOP	D2(X2,B2)	No Operation	BC	0,D2(X2,B2)
NOPR	R2	No Operation (RR format)	BCR	0,R2
		Used after compare instructions (A:B)		
BH	D2(X2,B2)	Branch on High	BC	2,D2(X2,B2)
BL	D2(X2,B2)	Branch on Low	BC	4,D2(X2,B2)
BE	D2(X2,B2)	Branch on Equal	BC	8,D2(X2,B2)
BNH	D2(X2,B2)	Branch on Not High	BC	13,D2(X2,B2)
BNL	D2(X2,B2)	Branch on Not Low	BC	11,D2(X2,B2)
BNE	D2(X2,B2)	Branch on Not Equal	BC	7,D2(X2,B2)
		Used after arithmetic instructions		
BO	D2(X2,B2)	Branch on Overflow	BC	1,D2(X2,B2)
BP	D2(X2,B2)	Branch on Plus	BC	2,D2(X2,B2)
BM	D2(X2,B2)	Branch on Minus	BC	4,D2(X2,B2)
BZ	D2(X2,B2)	Branch on Zero	BC	8,D2(X2,B2)
BNP	D2(X2,B2)	Branch on Not Plus	BC	13,D2(X2,B2)
BNM	D2(X2,B2)	Branch on Not Minus	BC	11,D2(X2,B2)
BNZ	D2(X2,B2)	Branch on Not Zero	BC	7,D2(X2,B2)
		Used after Test under Mask instructions		
BO	D2(X2,B2)	Branch if Ones	BC	1,D2(X2,B2)
BM	D2(X2,B2)	Branch if Mixed	BC	4,D2(X2,B2)
BZ	D2(X2,B2)	Branch if Zeros	BC	8,D2(X2,B2)
BNO	D2(X2,B2)	Branch if Not Ones	BC	14,D2(X2,B2)

EBCDIC CHART

The 256-position chart at the right, outlined by the heavy black lines, shows the graphic characters and control character representations for the Extended Binary-Coded Decimal Interchange Code (EBCDIC). The bit-position numbers, bit patterns, hexadecimal representations and card hole patterns for these and other possible EBCDIC characters are also shown.

To find the card hole patterns for most characters, partition the chart into four blocks as follows:



Block 1: Zone punches at top of table; digit punches at left

Block 2: Zone punches at bottom of table; digit punches at left

Block 3: Zone punches at top of table; digit punches at right

Block 4: Zone punches at bottom of table; digit punches at right

Fifteen positions, indicated by circled numbers, are exceptions to the above arrangement. The card hole patterns for these positions are given below the chart.

Following are some examples of the use of the EBCDIC chart:

Character	Type	Bit Pattern	Hex	Hole Pattern	
				Zone Punches	Digit Punches
PF	Control Character	00 00 0100	04	12 - 9	- 4
%	Special Graphic	01 10 1100	6C	0	- 8 - 4
R	Upper Case	11 01 1001	D9	11	- 9
a	Lower Case	10 00 0001	81	12 - 0	- 1
	Control Character, function not yet assigned	00 11 0000	30	12 - 11 - 0 - 9	- 8 - 1

Bit Positions
01 23 4567