

Copies of this and other IBM publications can be obtained through IBM Branch Offices. Comments concerning the contents of this publication may be addressed to: IBM, Product Publications Department, San Jose, Calif. 95114

### 1620 STORAGE REGISTER FUNCTIONS

IR-1	Contains address of next instruction if machine is stopped with Stop key or Halt instruction. Saves return address when interrupt is serviced.
IR-2	Saves return address when BT or BTM instructions are executed.
IR-3	Contains interrupt address — used in place of IR-1 during interrupt program operation (1710 Control System only).
IR-4	Saves return address when BT or BTM instructions are executed in the interrupt program (1710 Control System only).
OR-1	Contains Q address after I cycle of an instruction. In disk storage operations, used to store and control disk sector address.
OR-2	Contains P address after I cycle of an instruction. In disk storage operations, contains core storage address where data from disk storage is written to or read from.
OR-3	Retains address of low-order augend digit during addition for recomplement operation.
OR-4	Used to store and control the exponent address E <sub>4</sub> during automatic floating-point operations.
OR-5	Used to store and control the exponent address E <sub>9</sub> during automatic floating-point operations.
PR-1	Saves return address when a Save key operation occurs. Decremental for each new multiplier digit during multiply.
PR-2	Decremental for each new multiplicand digit during multiply.
PR-3	Used to add partial product to each multiply cycle result. In disk storage operations, used to store and control number of sectors in operation.
MAR	Addresses core storage.
MBR	Receives digits entering or leaving core storage.
MDR	Receives addressed digit entering or leaving core storage.
Digit	Stores partial product during multiplication.
OP	Contains Op code of instruction just executed if machine is stopped with Stop key or Halt instruction.
CR-1	Used to store the algebraic difference between E <sub>9</sub> and E <sub>4</sub> for determination of decimal alignment during automatic floating-point operations. CR-1 is also used during floating-point operations to count high-order zeros when normalizing — the contents of CR-1 are subtracted from E <sub>9</sub> .
Multiplier/Quotient	Contains multiplier digits during multiply and quotient digits during automatic divide operations. Q <sub>n</sub> digit on disk operations.
Digit and Branch	Decodes Q <sub>9</sub> and Q <sub>8</sub> digits of BI, BNI, and I/O instructions. Stores partial product digits during multiply instructions. Stores digits affecting MARS during all I cycles.

### BIT CONFIGURATION OF DECIMAL DIGITS

Digit	Bit Configuration					
	C	F	8	4	2	1
0	X					
1						X
2					X	
3	X				X	X
4				X		
5	X			X	X	X
6	X			X	X	
7				X	X	X
8			X			
9	X		X			X

### SWITCH AND INDICATOR CODES

Q <sub>9</sub> Q <sub>8</sub>	Name
01-04	1620 Program Switches 1-4
06	Read Check
07	Write Check
09	Last Card
11	High/Positive (H/P)
12	Equal/Zero (E/Z)
13	H/P or E/Z
14	Arithmetic Overflow Check
*15	Exponent Check
16	MBR-E Check
17	MBR-O Check
19	Any Data Check
*25	Printer Check
*33	Channel 9
*34	Channel 12
*35	Printer Busy
*36	Address Check
*37	Wrong-Length Record/ Read-Back Check
*38	Cylinder Overflow
*39	Any Disk Check

\*Special Feature

### CORE STORAGE TABLE AREAS

Address	Area
00000-00099	Console Area
00080-00099	Product Area
00100-00299	Multiply Table
00300-00399	Add Table

### INPUT/OUTPUT DEVICE CODES

Q <sub>9</sub> Q <sub>8</sub>	Device
01	Typewriter — 10 char/sec.
02	Tape Punch — 15 char/sec.
02	Plotter — Model 1: 18,000 steps/min. Model 2: 12,000 steps/min.
03	Paper Tape Reader — 150 char/sec.
04	Card Punch — Model 1: 125 cpm Model 2: 250 cpm
05	Card Reader — Model 1: 250 cpm Model 2: 500 cpm
07	Disk Storage — speed varies with function
09	Printer — Model 1: 150 to 450 lines/min. Model 2: 240 to 600 lines/min.

### TYPEWRITER CONTROL CODES

Q <sub>11</sub>	Function
1	Space
2	Return Carriage
8	Tabulate

### 1443 PRINTER CARRIAGE CONTROL (OP 34)

Q <sub>10</sub> Q <sub>11</sub>	Immediate Skip to	Q <sub>10</sub> Q <sub>11</sub>	Skip after Print to
71	Channel 1	41 (A)	Channel 1
72	Channel 2	42	Channel 2
73	Channel 3	43	Channel 3
74	Channel 4	44	Channel 4
75	Channel 5	45	Channel 5
76	Channel 6	46	Channel 6
77	Channel 7	47	Channel 7
78	Channel 8	48	Channel 8
79	Channel 9	49	Channel 9
70	Channel 10	40	Channel 10
33	Channel 11	03	Channel 11
34	Channel 12	04	Channel 12
	Immediate Space		Space after Print
51	1 Space	21	1 Space
52	2 Spaces	62	2 Spaces
53	3 Spaces	63	3 Spaces

### MULTIPLY TABLE

High-Order Positions of Address	Units Position of Address									
	0	1	2	3	4	5	6	7	8	9
0010	0	0	0	0	0	0	0	0	0	0
0011	0	0	1	0	2	0	3	0	4	0
0012	0	0	2	0	4	0	6	0	8	0
0013	0	0	3	0	6	0	9	0	2	1
0014	0	0	4	0	8	0	2	1	6	1
0015	0	0	5	0	0	1	5	1	0	2
0016	0	0	6	0	2	1	8	1	4	2
0017	0	0	7	0	4	1	1	2	8	2
0018	0	0	8	0	6	1	4	2	2	3
0019	0	0	9	0	8	1	7	2	6	3
0020	0	0	0	0	0	0	0	0	0	0
0021	5	0	6	0	7	0	8	0	9	0
0022	0	1	2	1	4	1	6	1	8	1
0023	5	1	8	1	1	2	4	2	7	2
0024	0	2	4	2	8	2	2	3	6	3
0025	5	2	0	3	5	3	0	4	5	4
0026	0	3	6	3	2	4	8	4	4	5
0027	5	3	2	4	9	4	6	5	3	6
0028	0	4	8	4	6	5	4	6	2	7
0029	5	4	4	5	3	6	2	7	1	8

### ADD TABLE

High-Order Positions of Address	Units Position of Address									
	0	1	2	3	4	5	6	7	8	9
0030	0	1	2	3	4	5	6	7	8	9
0031	1	2	3	4	5	6	7	8	9	0
0032	2	3	4	5	6	7	8	9	0	1
0033	3	4	5	6	7	8	9	0	1	2
0034	4	5	6	7	8	9	0	1	2	3
0035	5	6	7	8	9	0	1	2	3	4
0036	6	7	8	9	0	1	2	3	4	5
0037	7	8	9	0	1	2	3	4	5	6
0038	8	9	0	1	2	3	4	5	6	7
0039	9	0	1	2	3	4	5	6	7	8

### COMPARE RESULTS

Condition (Algebraic)	Indicator	
	High/Positive	Equal/Zero
P greater than Q	ON	OFF
P less than Q	OFF	OFF
P equal to Q	OFF	ON

P=Data in Field at P address  
Q=Data in Field at Q address

OP Code	SPS Mnemonic	Instruction	Significance of P and Q Address		Operation	Allowable Ind. Add.		Notes	Instruction Time In $\mu$ sec (except as otherwise noted)	OP Code
			P Address	Q Address		P Address	Q Address			
*01	FADD	Floating Add	Location of units position of exponent of Augend and Result.	Location of units position of exponent of Addend.	$M_P + M_Q$ replaces $M_P$ . exponent modified if required.	X	X	Recomplement and change the sign of P if signs initially unlike and numerical value of Q data is greater than P data.	400 + 100L basic time 80L recomb time.	*01
*02	FSUB	Floating Subtract	Location of units position of exponent of Minuend and Product.	Location of units position of exponent of Subtrahend.	$M_P - M_Q$ replaces $M_P$ . exponent modified if required.	X	X	Recomplement and change the sign of P if signs initially alike and numerical value of Q data is greater than P data.	400 + 100L basic time 80L recomb time.	*02
*03	FMUL	Floating Multiply	Location of units position of exponent of Multiplicand and Product.	Location of units position of exponent of Multiplier.	$M_P \times M_Q$ replaces $M_P$ . $E_P + E_Q$ replaces $E_P$ .	X	X	$L' =$ Number of digits mantissa is increased by shift left.	1120 + 80L + 168L'	*03
*05	FSL	Floating Shift Left	Location of high-order position of resulting field.	Location of units position of field shifted.	$M_Q$ shifted left so that high-order position = $L_P$ .	X	X		200 + 40L + 40L'	*05
*06	TFL	Transmit Floating	Location of units position of exponent of resulting field.	Location of units position of exponent of field transmitted.	$F_Q$ to $F_P$ .	X	X		240 + 40L.	*06
*07	BTFL	Branch and Transmit Floating	P - 1: location of units position of field to which Q field is transmitted. P: location of next instruction executed.	Location of units position of exponent of field transmitted.	Save $A_S$ , $F_Q$ to $L_P - 1$ , $I_P$ .	X	X		240 + 40L.	*07
*08	FSR	Floating Shift Right	Location of units position of resulting field.	Location of units position of field shifted.	$F_Q$ shifted right to $L_P$ .	X	X		200 + 40L.	*08
*09	FDIV	Floating Divide	Location of units position of exponent of Dividend and Quotient.	Location of units position of exponent of divide.	$M_P \div M_Q$ replaces $M_P$ . $E_P - E_Q$ replaces $E_P$ .	X	X	Average quotient digit = 4.5	880 + 940L + 520L'	*09
11	AM	Add Immediate	Location of units position of Augend and Result.	$Q_n$ is units position of Addend.	$F_P + Q$ replaces $F_P$ .	X			160 + 80D <sub>r</sub> basic time 80 D <sub>r</sub> recomb time.	11
21	A	Add	Same as Code 11.	Location of units position of Addend.	$F_P + F_Q$ replaces $F_P$ .	X	X	Recomplement and change the sign of P if signs initially unlike and numerical value of Q data is greater than P data.	Same as Code 11.	21
12	SM	Subtract Immediate	Location of units position of Minuend and Result.	$Q_n$ is units position of Subtrahend.	$F_P - Q$ replaces $F_P$ .	X		Recomplement and change the sign of P if signs initially alike and numerical value of Q data greater than P data.	160 + 80D <sub>r</sub> basic time. 80 D <sub>r</sub> recomb time.	12
22	S	Subtract	Same as Code 12.	Location of units position of Subtrahend.	$F_P - F_Q$ replaces $F_P$ .	X	X	Same as Code 12.	Same as Code 12.	22
13	MM	Multiply Immediate	Location of units position of Multiplicand.	$Q_n$ is units position of Multiplier.	$Q \times F_P$ (result at 00099).	X			560 + 40D <sub>r</sub> + 168D <sub>r</sub> D <sub>q</sub>	13
23	M	Multiply	Same as Code 13.	Location of units position of Multiplier.	$F_Q \times F_P$ (result at 00099).	X	X		560 + 40D <sub>r</sub> + 168 D <sub>r</sub> D <sub>q</sub> .	23
14	CM	Compare Immediate	Location of units position of field compared with Q field.	$Q_n$ units position of field compared with P field.	$F_P$ compared with Q.	X		$D_2 =$ number of positions compared until a digit other than zero is detected in either field.	200 + 80D <sub>r</sub> - Unlike signs. 160 + 80D <sub>r</sub> - Like signs.	14
24	C	Compare	Same as Code 14.	Location of units position of field compared with P field.	$F_P$ compared with $F_Q$ .	X	X	Same as Code 14.	Same as Code 14.	24
15	TDM	Transmit Digit Immediate	Location to which digit is transmitted.	$Q_n$ is digit transmitted.	$Q_n$ to location defined by P.	X			200	15
25	TD	Transmit Digit	Same as Code 15.	Location of digit transmitted.	$d_Q$ to $L_P$ .	X	X		200	25
16	TFM	Transmit Field Immediate	Location to which units position of field is transmitted.	$Q_n$ is units position of field transmitted.	Q to $L_P$ .	X			160 + 40D <sub>r</sub>	16
26	TF	Transmit Field	Same as Code 16.	Location of units position of field transmitted.	$F_Q$ to $L_P$ .	X	X		160 + 40D <sub>r</sub> .	26
17	BTM	Branch and Transmit Immediate	Same as Code 07.	$Q_n$ is units position of field transmitted.	Save $A_S$ , Q to $L_P - 1$ , do instruction defined by P.	X			200 + 40D <sub>r</sub>	17
27	BT	Branch and Transmit	Same as Code 07.	Same as Code 26.	Save $A_S$ , $F_Q$ to $L_P - 1$ , do instruction defined by P.	X	X		200 + 40D <sub>r</sub> .	27
*18	LDM	Load Dividend Immediate	Location in Product area to which units position of Dividend is transmitted.	$Q_n$ is units position of Dividend.	Q to $L_P$ .	X			400 + 40D <sub>r</sub>	*18
*28	LD	Load Dividend	Same as Code 18.	Location of units position of Dividend	$F_Q$ to $L_P$ .	X	X		Same as Code 18.	*28
*19	DM	Divide Immediate	Location in product area of units position of Divisor for first subtraction.	$Q_n$ is units position of Divisor.	Product area (00080 - 00099) $\div$ Q.	X		Average quotient digits = 4.5	160 + 520D <sub>r</sub> Q <sub>r</sub> + 740Q <sub>r</sub>	*19
*29	D	Divide	Same as Code 19.	Location of units position of Divisor.	Product area (00080 - 00099) $\div$ $F_Q$ .	X	X	Same as Code 19.	Same as Code 19.	*29
31	TR	Transmit Record	Location to which high-order position of record is transmitted.	Location of high-order digit of record transmitted.	Record defined by Q to $L_P$ .	X	X		160 + 40D <sub>r</sub>	31
32	SF	Set Flag	Location where flag is set.	Not used.	Place flag bit at $L_P$ .	X			200	32
33	CF	Clear Flag	Location where flag is cleared.	Not used.	Remove flag bit from $L_P$ .	X			200	33
*34	SK	Seek	Address of disk control field.	$Q_n$ and $Q_s$ specify disk storage (07). $Q_n = 1$ .	Move arm to cylinder designated by sector address in disk control field.	X		Average seek time = 250 ms Maximum seek time = 392 ms	320	*34

OP Code	SPS Mnemonic	Instruction	Significance of P and Q Address		Operation	Allowable Ind. Add.		Notes	Instruction Time In $\mu$ sec (except as otherwise noted)	OP Code
			P Address	Q Address		P Address	Q Address			
34	K	Control	Not used.	$Q_n$ and $Q_0$ specify I/O device. $Q_n$ specifies control function performed.	Do $Q_n$ on I/O <sub>0</sub> .			Depends on control function and speed of I/O unit.	34	
35	DN	Dump Numeric	Location of first character written.	$Q_n$ and $Q_0$ specify output device.	I/O <sub>0</sub> writes from $L_n$ to 19,999, 39,999 or 59,999.	X		Depends on speed of I/O unit and number of characters involved.	35	
*36	CDGN CDN CTGN CTN	Check Disk	Address of disk control field.	$Q_n Q_0$ must be 07, $Q_n$ specifies function performed.	Check one or more sectors with WLRC. Check one or more sectors WLRC. Check full track with WLRC. Check full track no WLRC.	X		$Q_n = 1$ All times average. $Q_n = 3$ $Q_n = 5$ $Q_n = 7$	(S = number of sectors). 22 + 2S ms. 22 + 2S ms. 60 ms. 60 ms.	*36
*36	RDCN RDN RTGN RTN	Read Disk	Same as above.	Same as above.	Read one or more sectors with WLRC. Read one or more sectors no WLRC. Read full track with WLRC. Read full track no WLRC.	X		$Q_n = 0$ All times average. $Q_n = 2$ $Q_n = 4$ $Q_n = 6$	(S = number of sectors). 22 + 2S ms. 22 + 2S ms. 60 ms. 60 ms.	*36
36	RN	Read Numeric	Location where first character is stored.	$Q_n Q_0$ specify input device.	I/O <sub>0</sub> reads at $L_n$ .	X		Card I/O 3.4 ms. Depends on speed of I/O device	36	
37	RA	Read Alphanumeric	P - 1: location where zone digit of first character is stored. P: location where numerical digit of first character is stored. Must be odd.	$Q_n$ and $Q_0$ specify input device.	I/O <sub>0</sub> reads at $L_n - 1$ .	X		Card I/O 3.4 ms. Depends on speed of I/O device	37	
*38	WDGN WDN WTGN WTN	Write Disk	Address of disk control field. Must be even.	$Q_n Q_0$ must be 07. $Q_n$ specifies function performed.	Write one or more sectors with WLRC. Write one or more sectors no WLRC. Write full track with WLRC. Write full track no WLRC.	X		$Q_n = 0$ All times average. $Q_n = 2$ $Q_n = 4$ $Q_n = 6$	22 + 2S ms. 22 + 2S ms. 60 ms. 60 ms.	*38
*38	PRN	Print Numerically	Location from which first character is printed.	$Q_n Q_0$ must be 09. $Q_n$ controls spacing.	Print from $L_n$ .	X		$Q_n = 0 =$ Space $Q_n = 1 =$ No space.	8.06 ms. (buffer read in).	*38
38	WN	Write Numerically	Location from which first character is written.	$Q_n$ and $Q_0$ specify output device.	I/O <sub>0</sub> writes from $L_n$ .	X		Card I/O 3.4 ms. Depends on speed of I/O device.	38	
*39	PRA	Print Alphanumerically	P - 1: location of zone digit of first character printed; P: location of numerical digit of first character.	$Q_n Q_0$ must be 09.	Print from $L_n - 1$ .	X		$Q_n = 0 =$ Space. $Q_n = 1 =$ No space.	8.06 ms. (buffer read in).	*39
39	WA	Write Alphanumerically	P - 1: location of zone digit of first character written. P: location of numerical digit of first character written. Must be odd.	$Q_n Q_0$ specify output device.	I/O <sub>0</sub> writes from $L_n - 1$ .	X		Card I/O 3.4 ms.	39	
41	NOP	No Operation	Not used.	Not used.	Go to address of next sequential instruction.			160	41	
42	BB	Branch Back	Not used.	Not used.	Do instruction at saved address.			200	42	
43	BD	Branch On Digit	Branch: location of next instruction executed. Must be even. No Branch: Not used.	Location tested for digit other than zero.	If digit at $L_n$ not zero, do instruction defined by P.	X	X	240 Branch, 200 No Branch.	43	
44	BNF	Branch No Flag	Same as Code 43.	Location tested for flag bit.	If no flag bit at $L_n$ , do instruction defined by P.	X	X	Same as Code 43.	44	
45	BNR	Branch No Record Mark	Same as Code 43.	Location tested for record mark character.	If no record mark or group mark at $L_n$ , do instruction defined by P.	X	X	Same as Code 43.	45	
46	BI	Branch Indicator	Same as Code 43.	$Q_n Q_0$ specify program switch or indicator tested.	If indicator defined by $Q_n Q_0$ on, do instruction defined by P.	X		300 Branch 160 No Branch.	46	
47	BNI	Branch No Indicator	Same as Code 43.	Same as Code 46.	If indicator defined by $Q_n Q_0$ off, do instruction defined by P.	X		Same as Code 46.	47	
48	H	Halt	Not used.	Not used.	Stop.			160	48	
49	B	Branch	Location of next instruction to be executed. Must be even.	Not used.	Do instruction defined by P.	X		200	49	
*55	BNG	Branch No Group Mark	Branch: location of next instruction. Must be even. No Branch: not used.	Location tested for group mark.	If no group mark at $L_n$ , do instruction defined by P.	X	X	240 Branch. 200 No Branch	*55	
*71	MF	Move Flag	Location to which flag is moved.	Location of flag to be moved.	Move flag from $L_n$ to $L_r$ .	X	X	240	*71	
*72	TNS	Transfer Numerical Strip	Location of units position of alphanumeric field. Must be odd.	Location of units position of numerical field.	$F_r$ to $F_n$ .	X	X	160 + 40 D <sub>r</sub> .	*72	
*73	TNF	Transfer Numerical Fill	Same as Code 72.	Same as Code 72.	$F_n$ to $F_r$ .	X	X	160 + 40 D <sub>r</sub> .	*73	

$A_n$  - Address of next sequential instruction.

$D_n$  - Number of digits, including high-order zeros, in dividend.

$D_r$  - Number of digits, including high-order zeros, in the field at P.

$D_0$  - Number of digits, including high-order zeros, in the field at Q.

$D_n'$  - Number of digits, including high-order zeros, in Q part of instruction.

$d_n$  - Digit at location defined by Q.

$D_r$  - Number of digits, including high-order zeros, in divisor.

$D_n$  - Number of digits compared until a digit other than zero is detected in either field.

$E_r$  - Exponent of field at P address.

$E_n$  - Exponent of field at Q address.

$F_r$  - Field defined by P.

$F_n$  - Field defined by Q.

$I_n$  - Instruction defined by P.

$I/O_n$  - I/O defined by  $Q_n Q_0$ .

$L$  - Number of digits in mantissa.

$L'$  - Number of digits mantissa is increased by shift left.

$L_n$  - Location defined by P.

$L_n'$  - Location defined by Q.

$M_r$  - Mantissa of field at P address.

$M_n$  - Mantissa of field at Q address.

ms = Milliseconds.

P - P part of instruction.

Q - Q part of instruction.

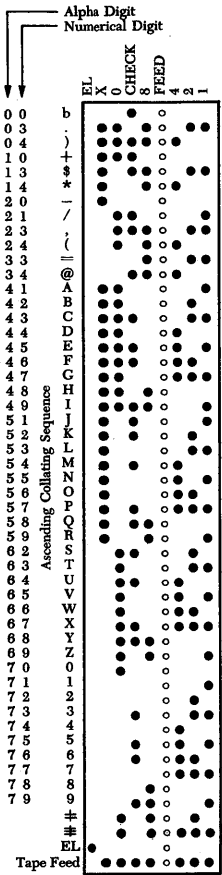
$Q_r$  - Number of digits, including high-order zeros, in quotient.

WLRC - Wrong-Length Record Check.

$\mu$ sec = Microseconds.

\* - Special Feature

PAPER TAPE CODE



ALPHAMERIC MODE

NUMERICAL MODE

CHARACTER CODING

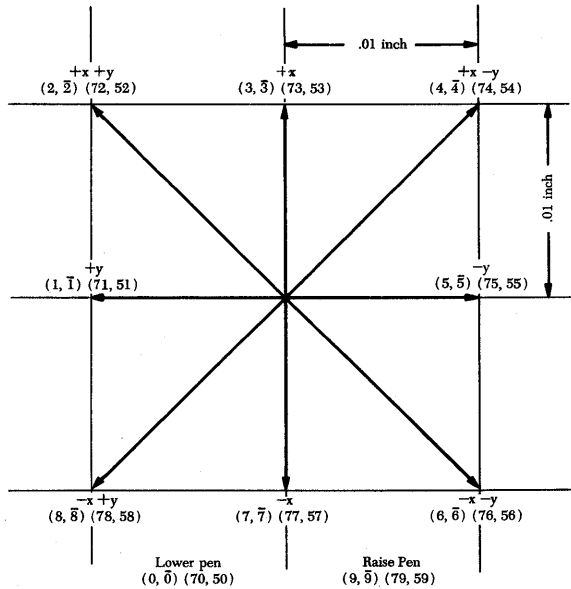
Character	Input			Core Storage		Output			
	Typewriter	Tape	Card	Alpha	Num	Typewriter	Tape	Card	Printer
(Blank)	(Space)	C	(Blank)	C	C	(Space)	C	(Blank)	(Blank)
(Period)	.	X0821	12, 3, 8	C	3	.	X0821	12, 3, 8	.
+	+	X0C84	12, 4, 8	C	4	+	X0C84	12, 4, 8	+
\$	\$	X0C	12	1	C	\$	X0C	12	\$
-(Hyphen)	-	XC821	11, 3, 8	1	3	-	XC821	11, 3, 8	-
/	/	X84	11, 4, 8	1	4	/	X84	11, 4, 8	/
(Comma)	,	X	11	2	C	,	X	11	,
=	=	OC1	0, 1	2	1	=	OC1	0, 1	=
@	@	OC821	0, 3, 8	2	3	@	OC821	0, 3, 8	@
A-I	A-I	084	0, 4, 8	2	4	A-I	084	0, 4, 8	A-I
0 (-)	(None)	821	3, 8	3	3	0 (-)	821	3, 8	0 (-)
J-R	J-R	C84	4, 8	3	4	J-R	C84	4, 8	J-R
1-9 (-)	J-R	X0, 1-9	12, 1-9	4	1-9	1-9 (-)	X0, 1-9	12, 1-9	1-9 (-)
S-Z	J-R	X, 1-9	11, 1-9	5	1-9	S-Z	X, 1-9	11, 1-9	S-Z
0 (+)	0	X, 1-9	11, 1-9	5	1-9	0 (+)	0	0	0
1-9 (+)	1-9	S-Z	0, 2-9	6	2-9	1-9 (+)	1-9	1-9	1-9
≠	≠	0	0 or 12, 0	7	C	≠	0	0	≠
≠	≠	1-9	1-9	7	1-9	≠	1-9	1-9	≠
≠	≠	082	0, 2, 8	C	C82	≠	082	0, 2, 8	≠
≠	≠	08421	0, 7, 8	C	C8421	≠	08421	0, 7, 8	≠
									Printer Dump
(Blank)	(Space)	C	(Blank)	C	0	0	0	0	0
0 (+)	0	0	0	C	0	0	0	0	0
0 (-)	0	X, X0C	11, 0	F	0	X	11, 0	1-9	1-9
1-9 (+)	1-9	1-9	1-9	1-9	1-9	1-9	1-9	1-9	1-9
1-9 (-)	1-9	X, 1-9	11, 1-9	F, 1-9	1-9	X, 1-9	11, 1-9	J-R	J-R
≠	≠	082	0, 2, 8	C82	(Stop, WN)	EOL (WN)	0, 2, 8	(Stop)	(Stop)
EOL	≠	EOL		C82	(DN)	082 (DN)			
≠	≠	X82	11, 8, 2	F82	≠	X82	11, 8, 2	(Stop)	W
≠	≠	08421	0, 7, 8	*C8421	≠	08421	0, 7, 8	(Stop)	C
≠	≠	X8421	12, 7, 8	F8421	≠	X8421	12, 7, 8	(Stop)	X
Num Blank†	@	C84	4, 8	C84	@	C84	(Blank)	(Blank)	@

\* Recorded as 0, 8, 4, 2, 1 in disk storage  
† For Card Format Use Only

SIGN CONTROL CHART

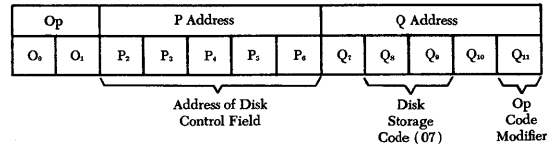
	Add				Subtract			
	+	-	+	-	+	-	+	-
Sign of P Field	+	+	-	-	+	+	-	-
Sign of Q Field	+	-	+	-	+	-	+	-
Stored P Field Sign	+	+	-	-	+	+	-	-
True or Complement Add/Q Field	True	Comp	Comp	True	True	Comp	True	Comp
Recomplement Answer and reverse sign of the P field if Q Field Value is Greater than P Field Value		X	X	-	X			X

PLOTTER RECORD CONTROL CHARACTERS

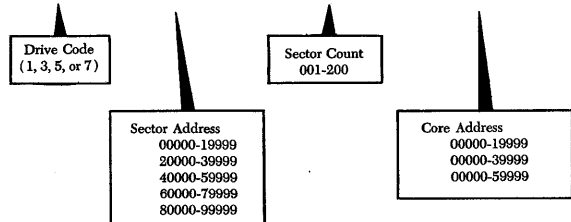
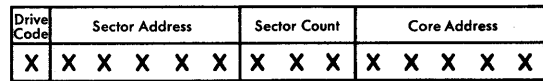


NOTE: Figures in parenthesis are (left) positive and negative numeric characters and (right) positive and negative alphabetic characters that correspond to the associated plotting movement.

DISK STORAGE INSTRUCTION FORMAT



DISK CONTROL FIELD FORMAT



DISK DRIVE CODING

Drive Code	or	Address Range	Selects Drive
1		00000-19999	0
3		20000-39999	1
5		40000-59999	2
7		60000-79999	3



International Business Machines Corporation  
Data Processing Division  
112 East Post Road, White Plains, New York