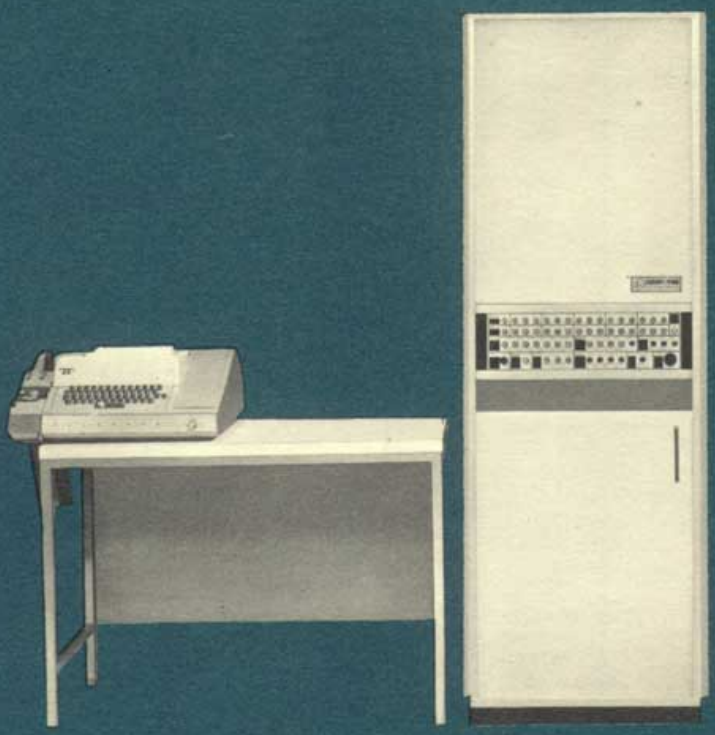
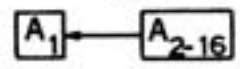
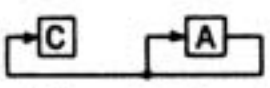
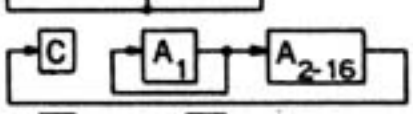
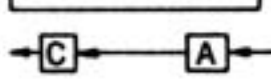
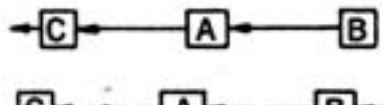
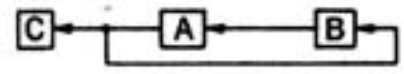
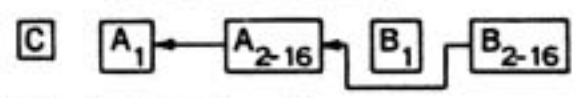
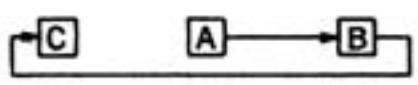
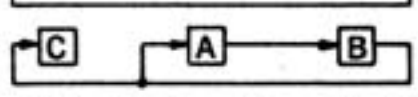
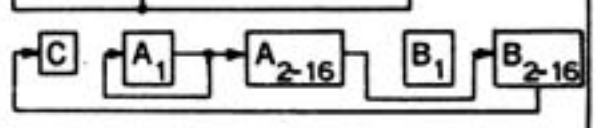
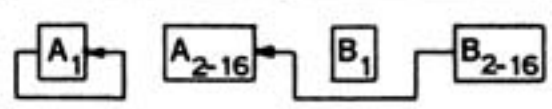
DDP-116 digital data processor



PROGRAMMERS REFERENCE CARD

Honeywell

COMPUTER CONTROL DIVISION

CATEGORY	OP-CODE MNEMONIC	OCTAL	FUNCTION	TYPE
Load and Store	CRA	140040	0 → (A)	G
	IAB	000201	(A) ⇄ (B)	G
	IMA	13	(A) ⇄ (EA)	MR
	LDA	02	(EA) → (A)	MR
	SCA*	000041	(SCTR) → $(A)_{11-16}$ 0 → $(A)_{1-10}$	G
	STA	04	(A) → (EA)	MR
Arithmetic	ADD††	06	(A) + (EA) → (A), 0V bit → (C)	MR
	DIV*†	17	(A) (B) ÷ (EA) → (A) quotient (B) remainder	MR
	IRS	12	(EA) + 1 → (EA)	MR
	MPY*	16	(A) × (EA) → (A) (B) A = high order, B = low order	MR
	SUB††	07	(A) − (EA) → (A), OV bit → (C)	MR
Logical	ANA	03	(A) ∧ (EA) → (A)	MR
	ERA	05	(A) ∀ (EA) → (A)	MR
Shift	ALR	0416		S
	ALS††	0415	 If A_1 changes, 1 → (C) If A_1 does not change, 0 → (C)	S
	ARR	0406		S
	ARS	0405		S
	LGL	0414		S
	LGR	0404		S
	LLL	0410		S
	LLR	0412		S
	LLS††	0411	 If A_1 changes, 1 → (C) If A_1 does not change, 0 → (C)	S
	LRL	0400		S
	LRR	0402		S
	LRS	0401		S
	NRM*	000101	 Until $(A_1) \neq (A_2)$	G

Transfer of Control	CAS	11	If (A) > (EA), take next instruction.
			If (A) = (EA), skip next instruction.
			If (A) < (EA), skip next two instructions.
				MR

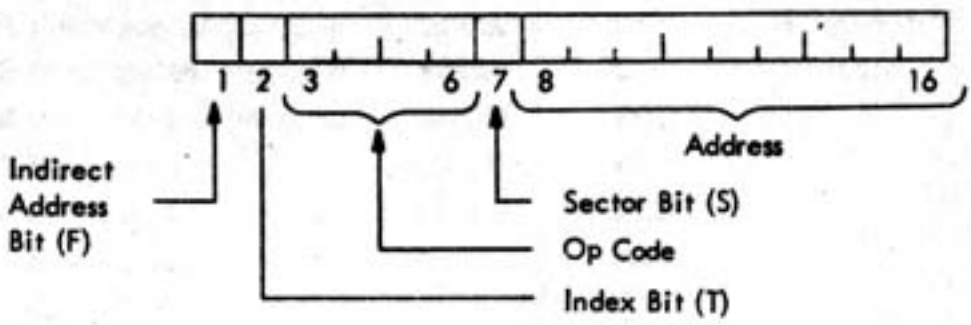
	JMP	01	EA → (P)	MR
	JST	10	$(P)_{3-16}$ → $(EA)_{3-16}$ EA + 1 → (P)	MR
	SLN	101100	Skip next instruction if $A_{16} \neq 0$	G
	SLZ	100100	Skip next instruction if $A_{1-16} \neq 0$	G
	SMI	101400	Skip next instruction if $A_1 = 1$	G
	SNZ	101040	Skip next instruction if $A_{1-16} \neq 0$	G
	SPL	100400	Skip next instruction if $A_1 = 0$	G
	SRC	100001	Skip next instruction if (C) = 0	G

CATEGORY	OP-CODE MNEMONIC	OP-CODE OCTAL	FUNCTION	TYPE
Transfer of Control (cont)	SR1	100020	Skip next instruction if SS1 is off.	G
	SR2	100010	Skip next instruction if SS2 is off.	G
	SR3	100004	Skip next instruction if SS3 is off.	G
	SR4	100002	Skip next instruction if SS4 is off.	G
	SSC	101001	Skip next instruction if (C) = 1	G
	SSR	100036	Skip next instruction if SS1∧SS2∧SS3∧SS4 are off.	G
	SSS	101036	Skip next instruction if SS1∨SS2∨SS3∨SS4 is on.	G
	SS1	101020	Skip next instruction if SS1 is on.	G
	SS2	101010	Skip next instruction if SS2 is on.	G
	SS3	101004	Skip next instruction if SS3 is on.	G
	SS4	101002	Skip next instruction if SS4 is on.	G
	SZE	100040	Skip next instruction if $A_{1-16} = 0$	G
Input/Output	INA	54	If not ready, no input, take next instruction. If ready, and $(IW)_7 = 1$, (INB) → (A), and skip next instruction. If ready, and $(IW)_7 = 0$, (INB) ∨ (A) → (A), skip next instruction.	IOT
	OCP	14	$(IW)_{7-16} \rightarrow (ADB)_{7-16}$	IOT
	OTA	74	If not ready, no output, take next instruction. If ready, (A) → (OTB), skip next instruction.	IOT
	SKS	34	Skip next instruction on sense condition.	IOT
Control	ACA††	141216	(C) +(A)→(A)	G
	AOA††	141206	(A) + 1→(A)	G
	CHS	140424	Complement $(A)_1 \rightarrow (A)_1$	G
	CMA	140401	One's complement (A) → (A)	G
	CSA	140320	$(A)_1 \rightarrow (C)$, $0 \rightarrow (A)_1$	G
	ENB	000401	Enable interrupt.	G
	HLT	000000	Stop computer operation.	G
	INH	001001	Inhibit Interrupt.	G
	NOP	101000	No operation.	G
	RCB	140200	0 → (C)	G
	SCB	140600	1 → (C)	G
	SSM	140500	$1 \rightarrow (A)_1$	G
	SSP	140100	$0 \rightarrow (A)_1$	G
	TCA	140407	Two's complement (A) → (A)	G

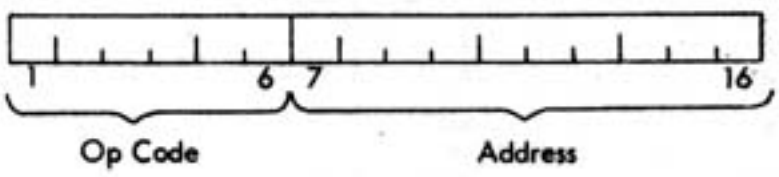
ADB = Address bus
C = Carry bit
EA = Effective address
G = Generic instruction
INB = Input bus
IOT = I/O and test instruction
IW = Instruction word
MR = Memory reference instruction (index bit, indirect address bit, and sector bit applicable)
OTB = Output Bus
OV = Overflow
S = Shift instruction
SCTR = Shift counter
( ) = Contents of
∧ = Logical AND
∨ = Logical OR
⊻ = Exclusive OR
* = Optional instruction
† = Improper divide possible
†† = Overflow possible

WORD FORMATS

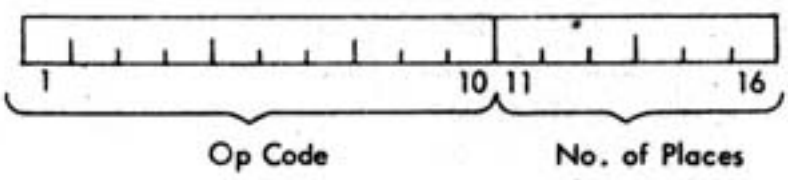
TYPE MR (MEMORY REFERENCE)



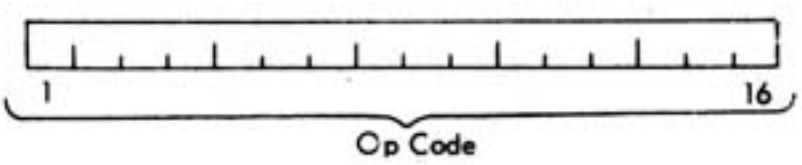
TYPE IOT (I/O AND TEST)



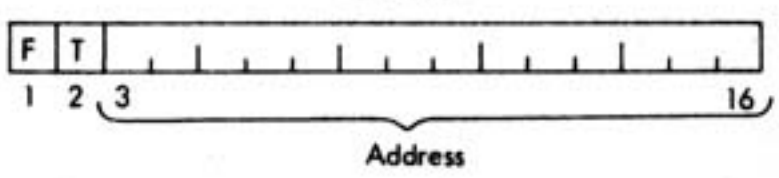
TYPE S (SHIFT)



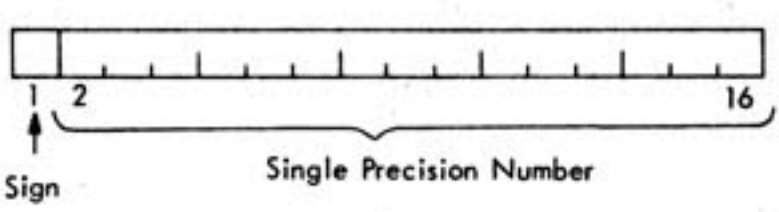
TYPE G (GENERIC)



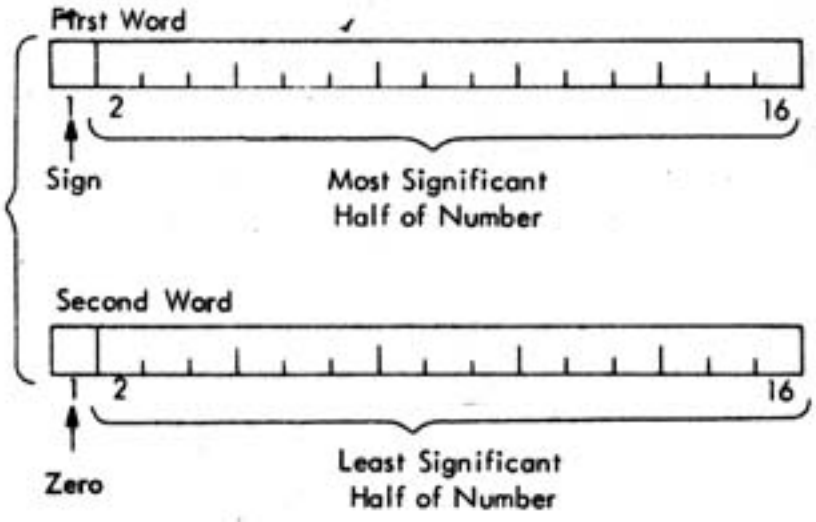
INDIRECT ADDRESS



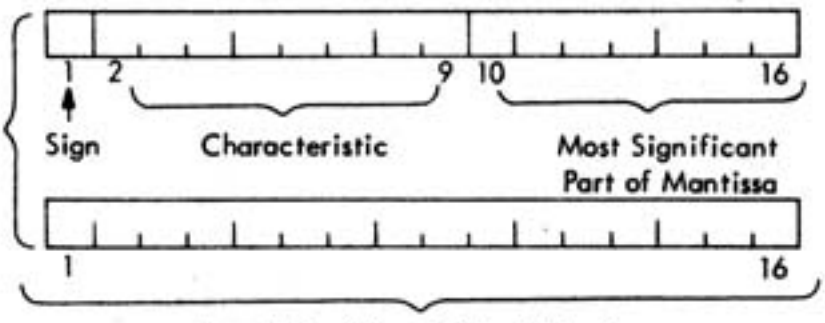
FIXED-POINT SINGLE PRECISION DATA



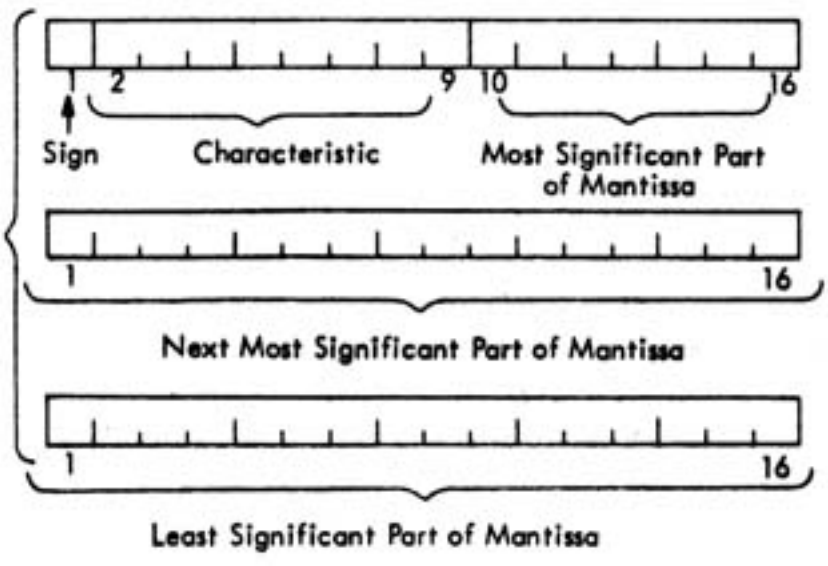
FIXED-POINT DOUBLE PRECISION DATA



FLOATING-POINT SINGLE PRECISION DATA



FLOATING-POINT DOUBLE PRECISION DATA



DDP-116 PERIPHERAL DEVICE CODES

CHARACTER	ASCII CODE	HOLLERITH CARD CODE	MAG TAPE CODE
0	260	0	12[1]
1	261	1	01
2	262	2	02
3	263	3	03
4	264	4	04
5	265	5	05
6	266	6	06
7	267	7	07
8	270	8	10
9	271	9	11
A	301	12-1	61
B	302	12-2	62
C	303	12-3	63
D	304	12-4	64
E	305	12-5	65
F	306	12-6	66
G	307	12-7	67
H	310	12-8	70
I	311	12-9	71
J	312	11-1	41
K	313	11-2	42
L	314	11-3	43
M	315	11-4	44
N	316	11-5	45
O	317	11-6	46
P	320	11-7	47
Q	321	11-8	50
R	322	11-9	51
S	323	0-2	22
T	324	0-3	23
U	325	0-4	24
V	326	0-5	25
W	327	0-6	26
X	330	0-7	27
Y	331	0-8	30
Z	332	0-9	31
Space	240	Blank	20
!	241[2]	8-6	16
"	242[2]	0-8-7	37
$	244[2]	11-8-3	53
%	245[2]	12-8-5	75
'	247[2]	8-4	14
(	250[2]	0-8-4	34
)	251[2]	12-8-4	74
*	252[2]	11-8-4	54
+	253[2]	12	60
,	254	0-8-3	33
—	255	11	40
.	256	12-8-3	73
/	257	0-1	21
:	272	8-5	15
;	273	11-8-2	52
<	274[2]	11-8-7	57
=	275[2]	8-3	13
>	276[2]	8-7	17
?	277[2]		
[	333[3]	11-8-5	55
\	334[4]	12-8-6	76
]	335[5]	0-8-6	36
↑	336[6]	12-8-2	72
←	337	12-8-7	77

NOTES:

1. When writing magnetic tapes in even parity (BCD) mode, 00_8 is written as 12_8; conversely when reading in even parity, 12_8 is read as 00_8.
2. Upper case characters on ASR-33
3. Upper case VT on ASR-33
4. Upper case FORM on ASR-33
5. Upper case M on ASR-33
6. Not on ASR-33

CONTROL AND SENSING CODES FOR ASR-33 AND HIGH-SPEED PAPER TAPE READER/PUNCH

OCP	'0001	Start paper tape reader
OCP	'0101	Stop paper tape reader
OCP	'0002	Enable paper tape punch
OCP	'0102	Paper tape punch power off
OCP	'0004	Enable ASR-33 in input mode
OCP	'0104	Enable ASR-33 in output mode
SKS	'0001	Paper tape reader is ready
SKS	'0002	Paper tape punch is ready
SKS	'0102	Paper tape punch is enabled
SKS	'0004	ASR-33 ready in ASCII mode
SKS	'0104	ASR-33 not busy
SKS	'0204	ASR-33 ready in binary mode
SKS	'0504	ASR-33 input not stop code
INA	'0001	Input if paper tape reader ready
INA	'1001	Clear A and input if paper tape reader ready
INA	'0004	Input ASCII code if ASR-33 ready
INA	'0204	Input binary code if ASR-33 ready
INA	'1004	Clear A and input ASCII code if ASR-33 ready
INA	'1204	Clear A and input binary code if ASR-33 ready
OTA	'0002	Output to paper tape punch if ready
OTA	'0004	Output ASCII code if ASR-33 ready
OTA	'0204	Output binary code if ASR-33 ready

SUMMARY OF DAP-116 PSEUDO-OPERATIONS

OPERATION MNEMONIC	MEANING
***	Op code set by program
BCI	Binary coded information
BES	Block ending with symbol
BSS	Block starting with symbol
BSZ	Block storage of ZEROs
CALL	Call subroutine
COMN	Put in common storage
DAC	Define address constant
DEC	Decimal to binary
END	End of source program
EQU	Equals
LIST	Generate listing
MOR	More
NLST	No listing
OCT	Octal to binary
ORG	Origin
POOL	Establish pool table
PZE	Plus ZERO
REL	Relocatable mode
SUBR	Subroutine entry point

ASR-33 FUNCTION CONTROLS

FUNCTION	OCTAL CODE
Bell	207
Line Feed	212
Return	215
X on	221
X off	223
Rub out	377

OP CODE OCTAL	MNEMONIC	INSTRUCTION	TYPE	EXECUTION TIME (μSEC)
01	JMP	Unconditional Jump	MR	1.7
02	LDA	Load A	MR	3.4
03	ANA	AND to A	MR	3.4
04	STA	Store A	MR	3.4
05	ERA	Exclusive OR to A	MR	3.4
06	ADD	Add	MR	3.4
07	SUB	Subtract	MR	3.4
10	JST	Jump and Store Location	MR	5.1
11	CAS	Compare Memory and A	MR	5.1
12	IRS	Increment, Replace Memory and Skip	MR	5.1
13	IMA	Interchange Memory and A	MR	5.1
14	OCP	Output Command Pulse	IOT	3.4
16	MPY*	Multiply*	MR	9.5
17	DIV*	Divide*	MR	17.9 (max)
34	SKS	Skip If Ready Line Set	IOT	3.4
54	INA	Input to A	IOT	5.1
74	OTA	Output from A	IOT	5.1
0400	LRL	Long Right Logical Shift	S	1.7 + 0.34N
0401	LRS	Long Arithmetic Right Shift	S	1.7 + 0.34N
0402	LRR	Long Right Rotate	S	1.7 + 0.34N
0404	LGR	Logical Right Shift	S	1.7 + 0.34N
0405	ARS	Arithmetic Right Shift	S	1.7 + 0.34N
0406	ARR	Logical Right Rotate	S	1.7 + 0.34N
0410	LLL	Long Left Logical Shift	S	1.7 + 0.34N
0411	LLS	Long Arithmetic Left Shift	S	1.7 + 0.34N
0412	LLR	Long Left Rotate	S	1.7 + 0.34N
0414	LGL	Logical Left Shift	S	1.7 + 0.34N
0415	ALS	Arithmetic Left Shift	S	1.7 + 0.34N
0416	ALR	Logical Left Rotate	S	1.7 + 0.34N
000000	HLT	Halt	G	
000041	SCA*	Shift Counter to A*	G	1.7
000101	NRM*	Normalize*	G	2.04 + 0.34N
000201	IAB	Interchange A and B	G	1.7
000401	ENB	Enable Interrupt	G	1.7
001001	INH	Inhibit Interrupt	G	1.7
100001	SRC	Skip If C Reset	G	1.7
100002	SR4	Skip If Sense Switch No. 4 Reset	G	1.7
100004	SR3	Skip If Sense Switch No. 3 Reset	G	1.7
100010	SR2	Skip If Sense Switch No. 2 Reset	G	1.7
100020	SR1	Skip If Sense Switch No. 1 Reset	G	1.7
100036	SSR	Skip If All Sense Switches Reset	G	1.7
100040	SZE	Skip If A Zero	G	1.7
100100	SLZ	Skip If LSB A Zero	G	1.7
100400	SPL	Skip If A Sign Plus	G	1.7
101000	NOP	No Operation	G	1.7
101001	SSC	Skip If C Bit Set	G	1.7
101002	SS4	Skip If Sense Switch No. 4 Set	G	1.7
101004	SS3	Skip If Sense Switch No. 3 Set	G	1.7
101010	SS2	Skip If Sense Switch No. 2 Set	G	1.7
101020	SS1	Skip If Sense Switch No. 1 Set	G	1.7
101036	SSS	Skip If Any Sense Switch Set	G	1.7
101040	SNZ	Skip If A Non-Zero	G	1.7
101100	SLN	Skip If LSB of A Non-Zero	G	1.7
101400	SMI	Skip If A Sign Minus	G	1.7
140024	CHS	Change Sign of A	G	1.7
140040	CRA	Clear A	G	1.7
140100	SSP	Set A Sign Plus	G	1.7
140200	RCB	Reset C Bit	G	1.7
140320	CSA	Copy Sign to C Bit and Set A Sign Plus	G	1.7
140401	CMA	Complement A	G	1.7
140407	TCA	2's Complement A	G	1.7
140500	SSM	Set A Sign Minus	G	1.7
140600	SCB	Set C Bit	G	1.7
141206	AOA	Add One to A	G	1.7
141216	ACA	Add C to A	G	1.7

* Optional (part of high-speed arithmetic option)

N = Number of shifts