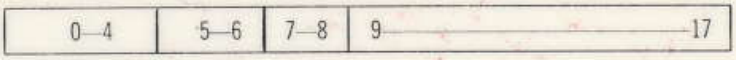
UNIVAC® 1818 ILAAS COMPUTER
INSTRUCTION REPERTOIRE

f	b	i	FUNCTION	DESCRIPTION	Time (μsec)
01	*	*	Enter A	(Y) → A	6
02	*	*	Enter Q	(Y) → Q	6
03	*	0, 1	Enter B	(Y) → B^b	10
	*	2	Enter B	y# → B^b	6
	*	3	Increment B	$(B^b)+y\# \rightarrow B^b$	10
04	*	*	Store A	(A) → Y	6
05	*	*	Store Q	(Q) → Y	6
06	*	0	Store B	$(B^b) \rightarrow y$	10
	N.I.	1	Store E	(E) → y	6
	N.I.	2	Load E	y → E	2
	N.I.	3	Load E	(y) → E	6
07	*	*	Ent. Log. Prod.	L [(Y) (Q)] → A	6
10	*	*	Substitute	$(Y)_n \rightarrow A_n$ for $(Q)_n=1$	8
11	*	*	Add A	(A)+(Y) → A	6
12	N.I.	0	Enter A Constant	y# → A	4
	N.I.	1	Modify A	$(A)+y\# \rightarrow A$	4
13	*	*	Subtract A	(A)−(Y) → A	6
14	*	*	Multiply	(Y)×(A) → AQ	24
15	*	*	Divide	(AQ)/(Y) → Q, Remainder → A	24
16	0	all	Shift	Shift (A) by y	(3 or 4)+n
	1	all	Shift	Shift (AQ) by y	(3 or 4)+n
	2	all	Shift	Shift (A) by (y)	(7 or 8)+n
	3	all	Shift	Shift (AQ) by (y)	(7 or 8)+n
	all	0	Shift	Right, sign prop.	Determined by the b designator
	all	1	Shift	Left, zero prop.	
	all	2	Shift	Right, zero prop.	
	all	3	Shift	Left, end around	
17	0	*	Index Y	(Y)+1 → Y	10
	1	*	Index Y	(Y)−1 → Y	10
	2	*	Index Y	(Y)+1 → Y, Set C.D.	10
	3	*	Index Y	(Y)−1 → Y, Set C.D.	10
20	*	*	Masked Compare	Compare (A)−L[(Y) (Q)], Set C.D.	6
21	0	*	Compare A	Compare (A)−(Y), Set C.D.	6
	1	*	Compare Q	Compare (Q)−(Y), Set C.D.	6
22	*	*	Compare B	Compare (B^b)−(Y), Set C.D.	10
23	0	N.I.	Designator Jump	If C.D. > Stage Not Set, y → P	2
	1	N.I.	Designator Jump	If C.D. > State Set, y → P	2
	2	N.I.	Designator Jump	If C.D. = Stage Not Set y → P	2
	3	N.I.	Designator Jump	If C.D. = Stage Set, y → P	2
24	*	N.I.	Index B Jump	Decrement B^b; If $(B^b) \geq 0$, y → P	10
25	0	N.I.	Test A Jump	If $(A) \geq 0$, y → P	2
	1	N.I.	Test Q Jump	If $(Q) \geq 0$, y → P	2
	2	N.I.	Test A Jump	If (A)=0, y → P	2
	3	N.I.	Test Q Jump	If (Q)=0, y → P	2
26	0	*	Store P Jump	(P) → PSTORE, (Y) → P	10
	1	*	Store P Jump	(P) → PSTORE+1, (Y) → P	10
	2	*	Store P Jump	(P) → PSTORE+2, (Y) → P	10
	3	*	Store P Jump	(P) → PSTORE+3, (Y) → P	10
27	N.I.	*	Indirect Jump	(Y) → P	10
30	N.I.	N.I.	Direct Jump	y → P	2
31	0	N.I.	Complement	−(A) → A	4
	1	N.I.	Complement	−(Q) → Q	4
32	0, 1	all	Input	CH. bi → y	6
	2, 3	all	Output	(y) → CH. bi	6
36	N.I.	N.I.	Program Stop	Stop if Stop Switch Set	2

INSTRUCTION WORD FORMAT



f—Function code designator
b—Index register designator
i—Memory bank designator
y—Operand designator

MEMORY ADDRESS ASSIGNMENT

Address Range (Octal)	Type		Description
00000-01777	DRO	First	1024 words of the DRO memory
02000-03777	DRO	Second	1024 words of the DRO memory
04000-05777	NDRO	First	1024 word NDRO memory module
06000-07777	NDRO	Second	1024 word NDRO memory module
10000-11777	NDRO	Third	1024 word NDRO memory module
12000-13777	NDRO	Fourth	1024 word NDRO memory module
14000-15777	NDRO	Fifth	1024 word NDRO memory module
16000-17777	NDRO	Sixth	1024 word NDRO memory module
20000-21777	NDRO	Seventh	1024 word NDRO memory module
22000-23777	NDRO	Eighth	1024 word NDRO memory module
24000-25777	NDRO	Ninth	1024 word NDRO memory module
26000-27777	NDRO	Tenth	1024 word NDRO memory module

SPECIAL ADDRESS ALLOCATIONS

Address (Octal)	Description
00001	B^1 (Index register 1)
00002	B^2 (Index register 2)
00003	B^3 (Index register 3)
00004	P-Store (Program P-Store address 1)
00005	P-Store+1 (Program P-Store address 2)
00006	P-Store+2 (Program P-Store address 3)
00007	P-Store+3 (Program P-Store address 4)
00010	P-Store (Interrupt P-Store address 1)
00011	P-Store+1 (Interrupt P-Store address 2)
00012	P-Store+2 (Interrupt P-Store address 3)
00013	P-Store+3 (Interrupt P-Store address 4)
04000	$Interrupt_1$ special address (Power Interrupt)
04001	$Interrupt_2$ special address (Real Time Clock Interrupt)
04002	$Interrupt_5$ special address
04003	$Interrupt_6$ special address
04004	$Interrupt_7$ special address
04005	$Interrupt_8$ special address
04006	$Interrupt_9$ special address
04007	Initial starting address
04010	$Interrupt_{10}$ special address
04011	$Interrupt_{11}$ special address (End of Input Cycle Int.)
04012	$Interrupt_{12}$ special address (Fault Interrupt)

NOTES

B^b—The index register indicated by the b designator
$Y=y+B^b$ interpreted as indicated by the value of the i designator
*—Indicates normal interpretation
N.I.—Indicates no interpretation
C.D.—Comparision designator
#—Indicates y is a nine-bit two's complement number

Execution times listed are for instructions performed without indexing, and assume that the instruction is located in permanent memory and the operand in varible memory. Indexing adds two microseconds to the listed execution times. During address modification bits 9 through 17 of the specified B register are added, modulo 2^9 to the address specified by y.