CENTRAL PROCESSOR INSTRUCTION FORMAT

MACHINE CODE

1st. CHARACTER	2nd. CHARACTER	3rd. CHARACTER	4th. CHARACTER	5th. CHARACTER
Operation Code	Most Significant Row	Most Significant Column	Least Significant Row	Least Significant Column

OPERATION **M ADDRESS (OPERAND)**

MODULE ADDRESSING:

For Central Processor Instructions the Module Designation of the Operand is determined by the "X" Bits of the 4th. and 5th. Characters. The following table of bits illustrates Bank Addressing.

"X" BIT CHAR. 4	"X" BIT CHAR. 5	MODULE DESIGNATION
ABSENT	ABSENT	1
PRESENT	ABSENT	2
ABSENT	PRESENT	3
PRESENT	PRESENT	4

EXTERNAL FUNCTION INSTRUCTION FORMAT

MACHINE CODE

1st. CHARACTER	2nd. CHARACTER	3rd. CHARACTER	4th. CHARACTER	5th. CHARACTER
Function	Sub Function A	Sub Function B	Sub Fuction C	Sub Function D

1005 80-COLUMN CODE

80-Col. Card Code	Printable Characters	XS-3 Code	80-Col. Card Code	Printable Characters	XS-3 Code
12-1	A	01 0100	7	7	00 1010
12-2	B	01 0101	8	8	00 1011
12-3	C	01 0110	9	9	00 1100
12-4	D	01 0111	12	&	01 0000
12-5	E	01 1000	11	-(minus)	00 0010
12-6	F	01 1001	12-0	?	01 0011
12-7	G	01 1010	11-0	!(exclam.)	10 0011
12-8	H	01 1011	0-1	/	11 0100
12-9	I	01 1100	2-8	+	11 0011
11-1	J	10 0100	3-8	#	01 1101
11-2	K	10 0101	4-8	@	10 1110
11-3	L	10 0110	5-8	:(colon)	01 0001
11-4	M	10 0111	6-8	>	11 1110
11-5	N	10 1000	7-8	'(apos.)	10 0000
11-6	O	10 1001	12-3-8	.(period)	01 0010
11-7	P	10 1010	12-4-8	¤	11 1101
11-8	Q	10 1011	12-5-8	[	00 1111
11-9	R	10 1100	12-6-8	<	01 1110
0-2	S	11 0101	12-7-8	=	01 1111
0-3	T	11 0110	11-3-8	$	10 0010
0-4	U	11 0111	11-4-8	*	10 0001
0-5	V	11 1000	11-5-8	]	00 0001
0-6	W	11 1001	11-6-8	;(semi-col)	00 1110
0-7	X	11 1010	11-7-8	Δ	10 1111
0-8	Y	11 1011	0-2-8	≠	11 0000
0-9	Z	11 1100	0-3-8	,(comma)	11 0010
0	0	00 0011	0-4-8	%	11 0001
1	1	00 0100	0-5-8	(	10 1101
2	2	00 0101	0-6-8	\	00 1101
3	3	00 0110	0-7-8	)	11 1111
4	4	00 0111			
5	5	00 1000	Blank	Space N.P.	00 0000
6	6	00 1001			

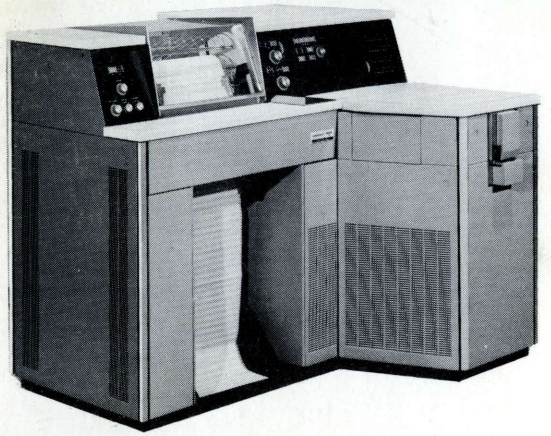
MEMORY MATRIX

ROW AND COLUMN MACHINE CODE

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
Sp	]	¤	4	;	1	F	.	1	5	:	-	2	7	B	8	D	[	<	#	H	C	\	G	A	6	?	3	9	E	&	=
!	*	!	M	@	Z	W	,	J	N	%	$	K	P	S	Q	U	Δ	>	¤	Y	T	(	X	/	O	+	L	R	V	≠	)

X ABSENT

X PRESENT



UNIVAC® 1005

EXTENDED SYSTEM

CODE CARD

CENTRAL PROCESSOR INSTRUCTION REPERTOIRE

	SAAL OP	OPER	DESCRIPTION	MACH CODE r1	MACH CODE r2	OPERATION
TRANSFER	LAr	M,L	Load Ascending AR1 or 2	Sp	,	(Mem)→(AR1 or 2)
	LDr	M,L	Load Descending AR1 or 2	]	*	(Mem)→(AR1 or 2)
	LPR	M,L	Load Print Descending	Ø		(Mem)→(Print Buffer)
	SAr	M,L	Store Ascending AR1 or 2	4	M	(AR1 or 2)→(Mem)
	SDr	M,L	Store Descending AR1 or 2	;	@	(AR1 or2)→(Mem)
	SPR	M,L	Store Print Descending	I		(Print Buffer)→(Mem)
	SHR	M,L S	Shift Right	F		(Mem)→(Mem Ascending)
	SHL	M,L S	Shift Left	.		(Mem)→(Mem Descending)
	CLR	M,L	Clear Area to Spaces	1		Spaces→(Mem)
	SC	M,L C	Store Character		≠	C→(Mem)
COMPARE	CAr	M,L	Compare Alpha AR1 or 2	5	N	(AR1 or 2) : (Mem) Alpha/Numeric
	CNr	M,L	Compare Numeric AR1 or 2	:	%	(AR1 or 2) : (Mem) Algebraic
	IC	M	Increment and Compare	—		(Mem)→Update→(Mem)
	CCA	M,L C	Compare Character Alpha		≠	(Mem) : C Alpha/Numeric
JUMP LOGIC	J	M	Jump Unconditional	2		(Branch→(Mem)
	J6	M	Jump Greater (Numeric)	B		(Branch→(Mem) if Greater
	JL	M	Jump Less Than (Numeric)	7		(Branch→(Mem) if Less
	JE	M	Jump Equal (Numeric)	8		(Branch→(Mem) if Equal
	JEA	M	Jump Equal (Alpha/Numeric)	8		(Branch→(Mem) if Equal
	JUA	M	Jump Unequal (Alpha/Numeric)	7		(Branch→(Mem) if Unequal
	JP	M	Jump Positive (Arithmetic)	7		(Branch→(Mem) if Positive
	JN	M	Jump Negative (Arithmetic)	B		(Branch→(Mem) if Negative
	JZ	M	Jump Zero (IC & Arithmetic)	8		(Branch→(Mem) if Zero
	JR	M	Store PAK in X REG→Jump	D		(PAK)→(X REG), (Branch→(Mem)
	JX	M	Store X REG in M	[		(X REG)→(Mem)
	JS3	M	Jump Alt Switch 3		V	(Branch→(Mem) if Alternate Switch 3 on
	JOF	M	Jump Arithmetic Overflow		V	(Branch→(Mem) if Arithmetic Overflow
ARITHMETIC	AMr	M,L	Add Alg. AR1 or 2 to M	<	>	(AR1 or 2) + (Mem)→(Mem) w/Sign Compare
	ARr	M,L	Add Alg. M to AR1 or 2	#	¤	(Mem)+(AR1 or 2)→(AR1 or 2) w/Sign Compare
	SMr	M,L	Subtract Alg. AR1 or 2 from M	H	Y	(Mem) – (AR1 or 2)→(Mem) w/Sign Compare
	SRr	M,L	Subtract Alg. M from AR1 or 2	C	T	(AR1 or 2)–(Mem)→(AR1 or 2) w/Sign Compare
	MUL	M,L	Multiply	\		(Mem) x (AR1)→(AR2)
	DIV	M,L	Divide	G		(AR2 ÷ (Mem)→(AR1 and 2)
EDIT	TRL	M,L	Translate	A		(Mem)→(Mem) Translated
	SZS	M,L	Ø Supress AR2 & Store Ascending		Θ	(AR2 Edited)→(Mem)
	LWS	M,L	Load AR2 w/Sign & Zone Delete		?	(Mem)→(AR2 with Edit)
	LNr	M,L	Zone Delete AR1 or 2	3	L	(Mem)→(AR1 or 2 w/Zone Delete)
	SED	M,L	Edit ,, . AR2 & Store Ascending		R	(AR2 Edited)→(Mem)
	LAN	M,L C	Logical And		≠	(M) ∧ C→(M)
	LOR	M,L C	Logical Or		≠	(M) ∨ C→(M)
	BSH	M,L	Bit Shift Circularly		≠	Shift (1 Chr Mem) Circularly One Bit

r = Register 1 or 2
M = Most Significant Location
L = Length of Operand
S = Number of Shift Positions
C = Character Stored in the Instruction
∧ = Logical And
∨ = Logical Or

INPUT/OUTPUT - INSTRUCTION REPERTOIRE

	SAAL OP	OPER	DESCRIPTION	MACH CODE	OPERATION
CARD SYSTEM	PTE		Punch Test	E	(Processor Interlock) If Punch Active
	XF	REA	Read Card	&	Card→(Card Buffer)
	XF	PR1	Print – SP1	&	(Print Buffer)→Printer SP1
	XF	PR2	Print – SP2	&	(Print Buffer)→Printer SP2
	XF	PR7	Print – SK7	&	(Print Buffer)→Printer→Channel 7 On Loop
	XF	PUN	Punch	&	(Punch Buffer)→Punch
	XF	RPR	Read – Print –SP1	&	Card→(Card Buffer) (Print Buffer)→Printer – SP1
	XF	RP2	Read – Print – SP2	&	Card→(Card Buffer) (Print Buffer)→Printer – SP2
	XF	RPH	Read – Punch	&	Card→(Card Buffer) (Punch Buffer)→Punch
	XF	RPP	Read – Print – SP1 – Punch	&	Card→(Card Buffer) (Print Buffer)→Printer – SP1 (Punch Buffer)→Punch
	XF	SK2	Skip 2	&	(Advance Paper)→Channel 2 On Loop
	XF	SK4	Skip 4	&	(Advance Paper)→Channel 4 On Loop
	XF	SK7	Skip 7	&	(Advance Paper)→Channel 7 On Loop
	XF	RCI	Read Code Image	&	Card→(Buffer Code Image)
	XF	PCI	Punch Code Image	&	(Punch Buffer Code Image)→Punch
	XF	RXC	Read Auxiliary Code Image	&	Card Aux Reader→(Buffer Code Image)
	XF	RX1	Read Auxiliary Stkr Sel 1	&	Card Aux Reader→(Buffer) Stkr Sel 1
	XF	RX2	Read Auxiliary Stkr Sel 2	&	Card Aux Reader→(Buffer) Stkr Sel 2
	XF	RX3	Read Auxiliary Stkr Sel 3	&	Card Aux Reader→(Buffer) Stkr Sel 3
	XF	PSS	Punch Stkr Sel	&	(Punch Buffer)→Punch Stkr Sel
	XF	RRP	Read/Read Punch	&	Card Punch→(Read/Punch Input Buffer) (Read/Punch Output Buffer)→Punch
	XF	RRS	Read/Read Punch Stkr Sel	&	Card Punch→(Read/Punch Input Buffer) (Read/Punch Output Buffer)→Punch Stkr Sel
	XF	RRC	Read/Read Punch Code Image	&	Card Punch→(Read/Punch Input Buffer Code Image) (Read/Punch Output Buffer Code Image)→Punch
	XFC	MACHINE CODE	Special Instructions	&	Input/Output Card System Combinations
TAPE	XF	RP1	Read Paper Tape 1 Frame	&	Paper Tape→(Buffer) 1 Frame
	XF	RP8	Read Paper Tape 80 Frames	&	Paper Tape→(Buffer) 80 Frames
	XF	RPS	Read Paper Tape Through Sentinel	&	Paper Tape→(Buffer) Through Sentinel
	XF	PP1	Punch Paper Tape 1 Frame	&	(Punch Buffer 1 Frame)→Paper Tape Punch (No Parity)
	XF	PPS	Punch Paper Tape to Sentinel	&	(Punch Buffer To Sentinel)→Paper Tape Punch (No Parity)
	XF	P1P	Punch Paper Tape 1 Frame	&	(Punch Buffer 1 Frame)→Paper Tape Punch (w/Parity)
	XF	PSP	Punch Paper Tape to Sentinel	&	(Punch Buffer To Sentinel)→Paper Tape Punch (w/Parity)
	JPE	M	Jump Parity Error	V	Branch→(Mem) If Parity Error
	JC8	M	Jump Channel 8	V	Branch→(Mem) If Channel 8
MAGNETIC TAPE	XF	RT_s,BF_n,L	Read Tape Normal Gain	≠	Mag Tape→(Buffer) Normal Gain
	XF	RT_{s+4},BF_n,L	Read Tape High Gain	≠	Mag Tape→(Buffer) High Gain
	XF	WT_s,BF_n,L	Write Tape	≠	(Buffer)→Mag Tape
	XF	ER_s,BF_n,L	Erase Before Write	≠	Erase Mag Tape – (Buffer)→Mag Tape
	XF	BS_s	Backspace 1 Block	≠	Backspace
	XF	RW_s	Rewind Servo	≠	Rewind
	JPE	M	Jump Parity Error	V	Branch→(Mem) If Parity Error
	JET	M	Jump End of Tape	V	Branch→(Mem) If End of Tape
DATA LINE TERMINAL 3	XF	SN8	Send DLT 80 Characters	V	(Buffer)→DLT 80 Characters
	XF	SNS	Send DLT Through Sentinel	V	(Buffer)→DLT
	XF	RCD	Receive DLT to EOM	V	(DLT)→Buffer
	JPE	M	Jump Parity Error	V	Branch→(Mem) If Parity Error
	JET	M	Jump End of Time	V	Branch→(Mem) End of Time (No Traffic 20 Seconds)
	PTE		Pause Test DLT	E	(Processor Interlock) If DLT Active
1001	XF	SI_1,BF_n,L	Send Buffer to 1001	V	L Buffer Characters→1001 Buffer
	XF	RI_1,BF_n,L	Receive from 1001	V	(Data)→Buffer
	J11	M	Jump Interrupt Unit 1	V	Branch→(Mem) If 1001 Ready
	JAL	M	Jump Alert	V	Branch→(Mem) If 1001 Interlocked

s = Servo Number
BF_n = Memory Module (1 through 4)
L = Length of Operand

1005 INPUT/OUTPUT-STORAGE AREAS

MODULE 1

ROW \ COLUMN	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	32
1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	
3	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84	85	86	87	88	89	90	91	92	93	
4	94	95	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	112	113	114	115	116	117	118	119	120	121	122	123	124	
5	125	126	127	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	144	145	146	147	148	149	150	151	152	153	154	155	
6	156	157	158	159	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	176	177	178	179	180	181	182	183	184	185	186	
7	187	188	189	190	191	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	208	209	210	211	212	213	214	215	216	217	
8	218	219	220	221	222	223	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	240	241	242	243	244	245	246	247	248	
9	249	250	251	252	253	254	255	256	257	258	259	260	261	262	263	264	265	266	267	268	269	270	271	272	273	274	275	276	277	278	279	
10	280	281	282	283	284	285	286	287	288	289	290	291	292	293	294	295	296	297	298	299	300	301	302	303	304	305	306	307	308	309	310	
11	311	312	313	314	315	316	317	318	319	320	321	322	323	324	325	326	327	328	329	330	331	332	333	334	335	336	337	338	339	340	341	
12	342	343	344	345	346	347	348	349	350	351	352	353	354	355	356	357	358	359	360	361	362	363	364	365	366	367	368	369	370	371	372	
13	373	374	375	376	377	378	379	380	381	382	383	384	385	386	387	388	389	390	391	392	393	394	395	396	397	398	399	400	401	402	403	
14	404	405	406	407	408	409	410	411	412	413	414	415	416	417	418	419	420	421	422	423	424	425	426	427	428	429	430	431	432	433	434	
15	435	436	437	438	439	440	441	442	443	444	445	446	447	448	449	450	451	452	453	454	455	456	457	458	459	460	461	462	463	464	465	
16	466	467	468	469	470	471	472	473	474	475	476	477	478	479	480	481	482	483	484	485	486	487	488	489	490	491	492	493	494	495	496	
17	497	498	499	500	501	502	503	504	505	506	507	508	509	510	511	512	513	514	515	516	517	518	519	520	521	522	523	524	525	526	527	
18	528	529	530	531	532	533	534	535	536	537	538	539	540	541	542	543	544	545	546	547	548	549	550	551	552	553	554	555	556	557	558	
19	559	560	561	562	563	564	565	566	567	568	569	570	571	572	573	574	575	576	577	578	579	580	581	582	583	584	585	586	587	588	589	
20	590	591	592	593	594	595	596	597	598	599	600	601	602	603	604	605	606	607	608	609	610	611	612	613	614	615	616	617	618	619	620	
21	621	622	623	624	625	626	627	628	629	630	631	632	633	634	635	636	637	638	639	640	641	642	643	644	645	646	647	648	649	650	651	
22	652	653	654	655	656	657	658	659	660	661	662	663	664	665	666	667	668	669	670	671	672	673	674	675	676	677	678	679	680	681	682	
23	683	684	685	686	687	688	689	690	691	692	693	694	695	696	697	698	699	700	701	702	703	704	705	706	707	708	709	710	711	712	713	
24	714	715	716	717	718	719	720	721	722	723	724	725	726	727	728	729	730	731	732	733	734	735	736	737	738	739	740	741	742	743	744	
25	745	746	747	748	749	750	751	752	753	754	755	756	757	758	759	760	761	762	763	764	765	766	767	768	769	770	771	772	773	774	775	
26	776	777	778	779	780	781	782	783	784	785	786	787	788	789	790	791	792	793	794	795	796	797	798	799	800	801	802	803	804	805	806	
27	807	808	809	810	811	812	813	814	815	816	817	818	819	820	821	822	823	824	825	826	827	828	829	830	831	832	833	834	835	836	837	
28	838	839	840	841	842	843	844	845	846	847	848	849	850	851	852	853	854	855	856	857	858	859	860	861	862	863	864	865	866	867	868	
29	869	870	871	872	873	874	875	876	877	878	879	880	881	882	883	884	885	886	887	888	889	890	891	892	893	894	895	896	897	898	899	
30	900	901	902	903	904	905	906	907	908	909	910	911	912	913	914	915	916	917	918	919	920	921	922	923	924	925	926	927	928	929	930	
31	931	932	933	934	935	936	937	938	939	940	941	942	943	944	945	946	947	948	949	950	951	952	953	954	955	956	957	958	959	960	961	
32																																

STATIC REGISTERS

READ TRANSLATE TABLE PRINT PUNCH

MODULE 2

	ARITHMETIC REGISTER 1										ARITHMETIC REGISTER 2																					
32	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	

MACHINE ADDRESSES:

	FROM		TO
MEMORY MODULE 1	0001	—	0961
MEMORY MODULE 2	0962	—	1922
MEMORY MODULE 3	1923	—	2883
MEMORY MODULE 4	2884	—	3844