

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
BS	136	10001000	210	88	ACA	1	Accum	5	20	C,Z,S,P	Add A to A with carry, Results in A
NULL	128	10000000	200	80	ADA	1	Accum	5	20	C,Z,S,P	Add A to A, Results in A
HT	137	10001001	211	89	ACB	1	Accum	5	20	C,Z,S,P	Add A to B with carry, Results in A
SOH	129	10000001	201	81	ADB	1	Accum	5	20	C,Z,S,P	Add A to B, Results in A
LF	138	10001010	212	8A	ACC	1	Accum	5	20	C,Z,S,P	Add A to C with carry, Results in A
STX	130	10000010	202	82	ADC	1	Accum	5	20	C,Z,S,P	Add A to C, Results in A
VT	139	10001011	213	8B	ACD	1	Accum	5	20	C,Z,S,P	Add A to D with carry, Results in A
ETX	131	10000011	203	83	ADD	1	Accum	5	20	C,Z,S,P	Add A to D, Results in A
FF	140	10001100	214	8C	ACE	1	Accum	5	20	C,Z,S,P	Add A to E with carry, Results in A
EOT	132	10000100	204	84	ADE	1	Accum	5	20	C,Z,S,P	Add A to E, Results in A
CR	141	10001101	215	8D	ACH	1	Accum	5	20	C,Z,S,P	Add A to H with carry, Results in A
ENQ	133	10000101	205	85	ADH	1	Accum	5	20	C,Z,S,P	Add A to H, Results in A
ACK	134	10000110	206	86	ADL	1	Accum	5	20	C,Z,S,P	Add A to L, Results in A
SO	142	10001110	216	8E	ACL	1	Accum	5	20	C,Z,S,P	Add A to L, with carry Results in A
	4	00000100	004	04	ADI	2	Accum	8	32	C,Z,S,P	Add A to Memory
SI	143	10001111	217	8F	ACM	1	Accum	8	32	C,Z,S,P	Add A to Memory HL with carry, Results in A
BEL	135	10000111	207	87	ADM	1	Accum	8	32	C,Z,S,P	Add A to Memory HL, Results in A
	12	00001100	014	0C	ACI	2	Accum	8	32	C,Z,S,P	Add A to Memory with carry
Space	160	10100000	240	A0	NDA	1	Accum	5	20	C=0,Z,S,P	AND A with A, results in A
!	161	10100001	241	A1	NDB	1	Accum	5	20	C=0,Z,S,P	AND A with B, results in A
"	162	10100010	242	A2	NDC	1	Accum	5	20	C=0,Z,S,P	AND A with C, results in A
#	163	10100011	243	A3	NDD	1	Accum	5	20	C=0,Z,S,P	AND A with D, results in A
\$	164	10100100	244	A4	NDE	1	Accum	5	20	C=0,Z,S,P	AND A with E, results in A
%	165	10100101	245	A5	NDH	1	Accum	5	20	C=0,Z,S,P	AND A with H, results in A
&	166	10100110	246	A6	NDL	1	Accum	5	20	C=0,Z,S,P	AND A with L, results in A
'	36	00100100	044	24	NDI	2	Accum	8	32	C=0,Z,S,P	AND A with Memory
.	167	10100111	247	A7	NDM	1	Accum	8	32	C=0,Z,S,P	AND A with Memory HL, results in A
	70	01000110	106	46	CAL	3	PC/Stack	11	44		Call Subroutine
	78	01001110	116	4E	CAL	3	PC/Stack	11	44		Call Subroutine
	86	01010110	126	56	CAL	3	PC/Stack	11	44		Call Subroutine
	94	01011110	136	5E	CAL	3	PC/Stack	11	44		Call Subroutine
	102	01100110	146	66	CAL	3	PC/Stack	11	44		Call Subroutine
	110	01101110	156	6E	CAL	3	PC/Stack	11	44		Call Subroutine
	118	01110110	166	76	CAL	3	PC/Stack	11	44		Call Subroutine
	126	01111110	176	7E	CAL	3	PC/Stack	11	44		Call Subroutine
	66	01000010	102	42	CFC	3	PC/Stack	9/11	36/44		Call Subroutine if Carry flag clear
	98	01100010	142	62	CTC	3	PC/Stack	9/11	36/44		Call Subroutine if Carry flag set
	122	01111010	172	7A	CTP	3	PC/Stack	9/11	36/44		Call Subroutine if Parity Even
	90	01011010	132	5A	CFP	3	PC/Stack	9/11	36/44		Call Subroutine if Parity Odd
	82	01010010	122	52	CFS	3	PC/Stack	9/11	36/44		Call Subroutine if Sign bit clear
	114	01110010	162	72	CTS	3	PC/Stack	9/11	36/44		Call Subroutine if Sign bit set
	74	01001010	112	4A	CFZ	3	PC/Stack	9/11	36/44		Call Subroutine if Zero flag clear
	106	01101010	152	6A	CTZ	3	PC/Stack	9/11	36/44		Call Subroutine if Zero flag set
8	184	10111000	270	B8	CPA	1	Accum	5	20	C,Z,S,P	Compare A with A
9	185	10111001	271	B9	CPB	1	Accum	5	20	C,Z,S,P	Compare A with B
:	186	10111010	272	BA	CPC	1	Accum	5	20	C,Z,S,P	Compare A with C
:	187	10111011	273	BB	CPD	1	Accum	5	20	C,Z,S,P	Compare A with D
<	188	10111100	274	BC	CPE	1	Accum	5	20	C,Z,S,P	Compare A with E
=	189	10111101	275	BD	CPH	1	Accum	5	20	C,Z,S,P	Compare A with H
>	190	10111110	276	BE	CPL	1	Accum	5	20	C,Z,S,P	Compare A with L
	60	00111100	074	3C	CPI	2	Accum	8	32	C,Z,S,P	Compare A with Memory
?	191	10111111	277	BF	CPM	1	Accum	8	32	C,Z,S,P	Compare A with Memory at HL
	9	00001001	011	09	DCB	1	Index Reg	5	20	Z,S,P	Decrement B
	17	00010001	021	11	DCC	1	Index Reg	5	20	Z,S,P	Decrement C
	25	00011001	031	19	DCD	1	Index Reg	5	20	Z,S,P	Decrement D
	33	00100001	041	21	DCE	1	Index Reg	5	20	Z,S,P	Decrement E
	41	00101001	051	29	DCH	1	Index Reg	5	20	Z,S,P	Decrement H
	49	00110001	061	31	DCL	1	Index Reg	5	20	Z,S,P	Decrement L
(168	10101000	250	A8	XRA	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with A, results in A
)	169	10101001	251	A9	XR B	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with B, results in A
*	170	10101010	252	AA	XRC	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with C, results in A
+	171	10101011	253	AB	XRD	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with D, results in A
,	172	10101100	254	AC	XRE	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with E, results in A

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
-	173	10101101	255	AD	XRH	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with H, results in A
.	174	10101110	256	AE	XRL	1	Accum	5	20	C=0,Z,S,P	Exclusive OR A with L, results in A
/	44	00101100	054	2C	XRI	2	Accum	8	32	C=0,Z,S,P	Exclusive OR A with Memory
/	175	10101111	257	AF	XRM	1	Accum	8	32	C=0,Z,S,P	Exclusive OR A with Memory HL, results in A
	0	00000000	000	00	HLT	1	Machine	4	16		Halt
	1	00000001	001	01	HLT	1	Machine	4	16		Halt
DEL	255	11111111	377	FF	HLT	1	Machine	4	16		Halt
	8	00001000	010	08	INB	1	Index Reg	5	20	Z,S,P	Increment B
	16	00010000	020	10	INC	1	Index Reg	5	20	Z,S,P	Increment C
	24	00011000	030	18	IND	1	Index Reg	5	20	Z,S,P	Increment D
	32	00100000	040	20	INE	1	Index Reg	5	20	Z,S,P	Increment E
	40	00101000	050	28	ICH	1	Index Reg	5	20	Z,S,P	Increment H
	48	00110000	060	30	INL	1	Index Reg	5	20	Z,S,P	Increment L
	65	01000001	101	41	INP	1	In/Out	8	32		Input from Port 0 to A
	67	01000011	103	43	INP	1	In/Out	8	32		Input from Port 1 to A
	69	01000101	105	45	INP	1	In/Out	8	32		Input from Port 2 to A (Mark 8 Front panel switches)
	71	01000111	107	47	INP	1	In/Out	8	32		Input from Port 3 to A
	73	01001001	111	49	INP	1	In/Out	8	32		Input from Port 4 to A
	75	01001011	113	4B	INP	1	In/Out	8	32		Input from Port 5 to A
	77	01001101	115	4D	INP	1	In/Out	8	32		Input from Port 6 to A
	79	01001111	117	4F	INP	1	In/Out	8	32		Input from Port 7 to A
	68	01000100	104	44	JMP	3	PC/Stack	11	44		Jump
	76	01001100	114	4C	JMP	3	PC/Stack	11	44		Jump
	84	01010100	124	54	JMP	3	PC/Stack	11	44		Jump
	92	01011100	134	5C	JMP	3	PC/Stack	11	44		Jump
	100	01100100	144	64	JMP	3	PC/Stack	11	44		Jump
	108	01101100	154	6C	JMP	3	PC/Stack	11	44		Jump
	116	01110100	164	74	JMP	3	PC/Stack	11	44		Jump
	124	01111100	174	7C	JMP	3	PC/Stack	11	44		Jump
	64	01000000	100	40	JFC	3	PC/Stack	9/11	36/44		Jump if Carry flag clear
	96	01100000	140	60	JTC	3	PC/Stack	9/11	36/44		Jump if Carry flag set
	120	01111000	170	78	JTP	3	PC/Stack	9/11	36/44		Jump if Parity Even
	88	01011000	130	58	JFP	3	PC/Stack	9/11	36/44		Jump if Parity Odd
	80	01010000	120	50	JFS	3	PC/Stack	9/11	36/44		Jump if Sign bit clear
	112	01110000	160	70	JTS	3	PC/Stack	9/11	36/44		Jump if Sign bit set
	72	01001000	110	48	JFZ	3	PC/Stack	9/11	36/44		Jump if Zero flag clear
	104	01101000	150	68	JTZ	3	PC/Stack	9/11	36/44		Jump if Zero flag set
	6	00000110	006	06	LAI	2	Index Reg	8	32		Load A
@	192	11000000	300	C0	LAA	1	Index Reg	5	20		Load A with A
A	193	11000001	301	C1	LAB	1	Index Reg	5	20		Load A with B
B	194	11000010	302	C2	LAC	1	Index Reg	5	20		Load A with C
C	195	11000011	303	C3	LAD	1	Index Reg	5	20		Load A with D
D	196	11000100	304	C4	LAE	1	Index Reg	5	20		Load A with E
E	197	11000101	305	C5	LAH	1	Index Reg	5	20		Load A with H
F	198	11000110	306	C6	LAL	1	Index Reg	5	20		Load A with L
G	199	11000111	307	C7	LAM	1	Index Reg	8	32		Load A with Memory address L,H
	14	00001110	016	0E	LBI	2	Index Reg	8	32		Load B
H	200	11001000	310	C8	LBA	1	Index Reg	5	20		Load B with A
I	201	11001001	311	C9	LBB	1	Index Reg	5	20		Load B with B
J	202	11001010	312	CA	LBC	1	Index Reg	5	20		Load B with C
K	203	11001011	313	CB	LBD	1	Index Reg	5	20		Load B with D
L	204	11001100	314	CC	LBE	1	Index Reg	5	20		Load B with E
M	205	11001101	315	CD	LBH	1	Index Reg	5	20		Load B with H
N	206	11001110	316	CE	LBL	1	Index Reg	5	20		Load B with L
O	207	11001111	317	CF	LBM	1	Index Reg	8	32		Load B with Memory address L,H
	22	00010110	026	16	LCI	2	Index Reg	8	32		Load C
P	208	11010000	320	D0	LCA	1	Index Reg	5	20		Load C with A
Q	209	11010001	321	D1	LCB	1	Index Reg	5	20		Load C with B
R	210	11010010	322	D2	LCC	1	Index Reg	5	20		Load C with C
S	211	11010011	323	D3	LCD	1	Index Reg	5	20		Load C with D
T	212	11010100	324	D4	LCE	1	Index Reg	5	20		Load C with E
U	213	11010101	325	D5	LCH	1	Index Reg	5	20		Load C with H
V	214	11010110	326	D6	LCL	1	Index Reg	5	20		Load C with L
W	215	11010111	327	D7	LCM	1	Index Reg	8	32		Load C with Memory address L,H
	30	00011110	036	1E	LDI	2	Index Reg	8	32		Load D
X	216	11011000	330	D8	LDA	1	Index Reg	5	20		Load D with A
Y	217	11011001	331	D9	LDB	1	Index Reg	5	20		Load D with B

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
Z	218	11011010	332	DA	LDC	1	Index Reg	5	20		Load D with C

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
[219	11011011	333	DB	LDD	1	Index Reg	5	20		Load D with D
\	220	11011100	334	DC	LDE	1	Index Reg	5	20		Load D with E
]	221	11011101	335	DD	LDH	1	Index Reg	5	20		Load D with H
^	222	11011110	336	DE	LDL	1	Index Reg	5	20		Load D with L
_	223	11011111	337	DF	LDM	1	Index Reg	8	32		Load D with Memory address L,H
`	38	00100110	046	26	LEI	2	Index Reg	8	32		Load E
a	224	11100000	340	E0	LEA	1	Index Reg	5	20		Load E with A
b	225	11100001	341	E1	LEB	1	Index Reg	5	20		Load E with B
c	226	11100010	342	E2	LEC	1	Index Reg	5	20		Load E with C
d	227	11100011	343	E3	LED	1	Index Reg	5	20		Load E with D
e	228	11100100	344	E4	LEE	1	Index Reg	5	20		Load E with E
f	229	11100101	345	E5	LEH	1	Index Reg	5	20		Load E with H
g	230	11100110	346	E6	LEL	1	Index Reg	5	20		Load E with L
h	231	11100111	347	E7	LEM	1	Index Reg	8	32		Load E with Memory address L,H
i	46	00101110	056	2E	LHI	2	Index Reg	8	32		Load H
j	232	11101000	350	E8	LHA	1	Index Reg	5	20		Load H with A
k	233	11101001	351	E9	LHB	1	Index Reg	5	20		Load H with B
l	234	11101010	352	EA	LHC	1	Index Reg	5	20		Load H with C
m	235	11101011	353	EB	LHD	1	Index Reg	5	20		Load H with D
n	236	11101100	354	EC	LHE	1	Index Reg	5	20		Load H with E
o	237	11101101	355	ED	LHH	1	Index Reg	5	20		Load H with H
p	238	11101110	356	EE	LHL	1	Index Reg	5	20		Load H with L
q	239	11101111	357	EF	LHM	1	Index Reg	8	32		Load H with Memory address L,H
r	54	00110110	066	36	LLI	2	Index Reg	8	32		Load L
s	240	11110000	360	F0	LLA	1	Index Reg	5	20		Load L with A
t	241	11110001	361	F1	LLB	1	Index Reg	5	20		Load L with B
u	242	11110010	362	F2	LLC	1	Index Reg	5	20		Load L with C
v	243	11110011	363	F3	LLD	1	Index Reg	5	20		Load L with D
w	244	11110100	364	F4	LLE	1	Index Reg	5	20		Load L with E
x	245	11110101	365	F5	LLH	1	Index Reg	5	20		Load L with H
y	246	11110110	366	F6	LLL	1	Index Reg	5	20		Load L with L
z	247	11110111	367	F7	LLM	1	Index Reg	8	32		Load L with Memory address L,H
{	62	00111110	076	3E	LMI	2	Index Reg	9	36		Load Memory at H, L with data
	248	11111000	370	F8	LMA	1	Index Reg	7	28		Load Memory L,H with A
}	249	11111001	371	F9	LMB	1	Index Reg	7	28		Load Memory L,H with B
~	250	11111010	372	FA	LMC	1	Index Reg	7	28		Load Memory L,H with C
0	251	11111011	373	FB	LMD	1	Index Reg	7	28		Load Memory L,H with D
1	252	11111100	374	FC	LME	1	Index Reg	7	28		Load Memory L,H with E
2	253	11111101	375	FD	LMH	1	Index Reg	7	28		Load Memory L,H with H
3	254	11111110	376	FE	LML	1	Index Reg	7	28		Load Memory L,H with L
4	176	10110000	260	B0	ORA	1	Accum	5	20	C=0,Z,S,P	OR A with A, results in A
5	177	10110001	261	B1	ORB	1	Accum	5	20	C=0,Z,S,P	OR A with B, results in A
6	178	10110010	262	B2	ORC	1	Accum	5	20	C=0,Z,S,P	OR A with C, results in A
7	179	10110011	263	B3	ORD	1	Accum	5	20	C=0,Z,S,P	OR A with D, results in A
8	180	10110100	264	B4	ORE	1	Accum	5	20	C=0,Z,S,P	OR A with E, results in A
9	181	10110101	265	B5	ORH	1	Accum	5	20	C=0,Z,S,P	OR A with H, results in A
A	182	10110110	266	B6	ORL	1	Accum	5	20	C=0,Z,S,P	OR A with L, results in A
B	52	00110100	064	34	ORI	2	Accum	8	32	C=0,Z,S,P	OR A with Memory
C	183	10110111	267	B7	ORM	1	Accum	8	32	C=0,Z,S,P	OR A with Memory HL, results in A
D	81	01010001	121	51	OUT	1	In/Out	6	24		Output A to Port 0 (Mark 8 front panel LEDs)
E	83	01010011	123	53	OUT	1	In/Out	6	24		Output A to Port 1 (Mark 8 "A" port)
F	85	01010101	125	55	OUT	1	In/Out	6	24		Output A to Port 2 (Mark 8 "B" port)
G	87	01010111	127	57	OUT	1	In/Out	6	24		Output A to Port 3 (Mark 8 "C" port)
H	89	01011001	131	59	OUT	1	In/Out	6	24		Output A to Port 4 (Mark 8 "D" port)
I	91	01011011	133	5B	OUT	1	In/Out	6	24		Output A to Port 5
J	93	01011101	135	5D	OUT	1	In/Out	6	24		Output A to Port 6
K	95	01011111	137	5F	OUT	1	In/Out	6	24		Output A to Port 7
L	97	01100001	141	61	OUT	1	In/Out	6	24		Output A to Port 8
M	99	01100011	143	63	OUT	1	In/Out	6	24		Output A to Port 9
N	101	01100101	145	65	OUT	1	In/Out	6	24		Output A to Port 10
O	103	01100111	147	67	OUT	1	In/Out	6	24		Output A to Port 11
P	105	01101001	151	69	OUT	1	In/Out	6	24		Output A to Port 12
Q	107	01101011	153	6B	OUT	1	In/Out	6	24		Output A to Port 13
R	109	01101101	155	6D	OUT	1	In/Out	6	24		Output A to Port 14
S	111	01101111	157	6F	OUT	1	In/Out	6	24		Output A to Port 15

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
	113	01110001	161	71	OUT	1	In/Out	6	24		Output A to Port 16
	115	01110011	163	73	OUT	1	In/Out	6	24		Output A to Port 17
	117	01110101	165	75	OUT	1	In/Out	6	24		Output A to Port 18
	119	01110111	167	77	OUT	1	In/Out	6	24		Output A to Port 19
	121	01111001	171	79	OUT	1	In/Out	6	24		Output A to Port 20
	123	01111011	173	7B	OUT	1	In/Out	6	24		Output A to Port 21
	125	01111101	175	7D	OUT	1	In/Out	6	24		Output A to Port 22
	127	01111111	177	7F	OUT	1	In/Out	6	24		Output A to Port 23
	5	00000101	005	05	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,000
	13	00001101	015	0D	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,010
	21	00010101	025	15	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,020
	29	00011101	035	1D	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,030
	37	00100101	045	25	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,040
	45	00101101	055	2D	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,050
	53	00110101	065	35	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,060
	61	00111101	075	3D	RST	1	PC/Stack	5	20		Restart (Call Subroutine) at 00,070
	7	00000111	007	07	RET	1	PC/Stack	5	20		Return from Subroutine
	15	00001111	017	0F	RET	1	PC/Stack	5	20		Return from Subroutine
	23	00010111	027	17	RET	1	PC/Stack	5	20		Return from Subroutine
	31	00011111	037	1F	RET	1	PC/Stack	5	20		Return from Subroutine
	39	00100111	047	27	RET	1	PC/Stack	5	20		Return from Subroutine
	47	00101111	057	2F	RET	1	PC/Stack	5	20		Return from Subroutine
	55	00110111	067	37	RET	1	PC/Stack	5	20		Return from Subroutine
	63	00111111	077	3F	RET	1	PC/Stack	5	20		Return from Subroutine
	3	00000011	003	03	RFC	1	PC/Stack	3/5	12/20		Return from Subroutine if Carry clear
	35	00100011	043	23	RTC	1	PC/Stack	3/5	12/20		Return from Subroutine if Carry set
	59	00111011	073	3B	RTP	1	PC/Stack	3/5	12/20		Return from Subroutine if Parity Even
	27	00011011	033	1B	RFP	1	PC/Stack	3/5	12/20		Return from Subroutine if Parity Odd
	19	00010011	023	13	RFS	1	PC/Stack	3/5	12/20		Return from Subroutine if Sign clear
	51	00110011	063	33	RTS	1	PC/Stack	3/5	12/20		Return from Subroutine if Sign Set
	11	00001011	013	0B	RFZ	1	PC/Stack	3/5	12/20		Return from Subroutine if Zero flag clear
	43	00101011	053	2B	RTZ	1	PC/Stack	3/5	12/20		Return from Subroutine if Zero flag set
	18	00010010	022	12	RAL	1	Accum	5	20	C	Rotate A Left through Carry
	2	00000010	002	02	RLC	1	Accum	5	20	C	Rotate A Left, bit 7 to Carry
	26	00011010	032	1A	RAR	1	Accum	5	20	C	Rotate A Right through Carry
	10	00001010	012	0A	RRC	1	Accum	5	20	C	Rotate A Right, bit 0 to Carry
CAN	152	10011000	230	98	SBA	1	Accum	5	20	C,Z,S,P	Subtract A from A with borrow, results in A
DLE	144	10010000	220	90	SUA	1	Accum	5	20	C,Z,S,P	Subtract A from A, results in A
EM	153	10011001	231	99	SBB	1	Accum	5	20	C,Z,S,P	Subtract B from A with borrow, results in A
DC1	145	10010001	221	91	SUB	1	Accum	5	20	C,Z,S,P	Subtract B from A, results in A
SUB	154	10011010	232	9A	SBC	1	Accum	5	20	C,Z,S,P	Subtract C from A with borrow, results in A
DC2	146	10010010	222	92	SUC	1	Accum	5	20	C,Z,S,P	Subtract C from A, results in A
ESC	155	10011011	233	9B	SBD	1	Accum	5	20	C,Z,S,P	Subtract D from A with borrow, results in A
DC3	147	10010011	223	93	SUD	1	Accum	5	20	C,Z,S,P	Subtract D from A, results in A
FS	156	10011100	234	9C	SBE	1	Accum	5	20	C,Z,S,P	Subtract E from A with borrow, results in A
DC4	148	10010100	224	94	SUE	1	Accum	5	20	C,Z,S,P	Subtract E from A, results in A
GS	157	10011101	235	9D	SBH	1	Accum	5	20	C,Z,S,P	Subtract H from A with borrow, results in A
NAK	149	10010101	225	95	SUH	1	Accum	5	20	C,Z,S,P	Subtract H from A, results in A
RS	158	10011110	236	9E	SBL	1	Accum	5	20	C,Z,S,P	Subtract L from A with borrow, results in A
SYN	150	10010110	226	96	SUL	1	Accum	5	20	C,Z,S,P	Subtract L from A, results in A
	20	00010100	024	14	SUI	2	Accum	8	32	C,Z,S,P	Subtract Memory from A
	28	00011100	034	1C	SBI	2	Accum	8	32	C,Z,S,P	Subtract Memory from A, with borrow
US	159	10011111	237	9F	SBM	1	Accum	8	32	C,Z,S,P	Subtract Memory HL from A with borrow, results in A
ETB	151	10010111	227	97	SUM	1	Accum	8	32	C,Z,S,P	Subtract Memory HL from A, results in A
	34	00100010	042	22	!						Unimplemented / unknown
	42	00101010	052	2A	!						Unimplemented / unknown
	50	00110010	062	32	!						Unimplemented / unknown
	56	00111000	070	38	!						Unimplemented / unknown
	57	00111001	071	39	!						Unimplemented / unknown
	58	00111010	072	3A	!						Unimplemented / unknown

8008 Instruction Set

Function Sort

By Bryan Blackburn, <http://members.cox.net/oldcomp>

ASCII	Decimal	Binary	Octal	Hex	Mnem	Byte length	Type	States	Time, uS	Flags Affected	Description
ASCII Abbreviations for Control Characters											
NUL									DC1		Device Control 1
SOH									DC2		Device Control 2
STX									DC3		Device Control 3
ETX									DC4		Device Control 4
EOT									NAK		Negative Acknowledge
ENQ									SYN		Synchronous Idle
ACK									ETB		End of Transmission Block
BEL									CAN		Cancel
BS									EM		End of Medium
HT									SUB		Substitute
LF									ESC		Escape
VT									FS		File Separator
FF									GS		Group Separator
CR									RS		Record Separator
SO									US		Unit Separator
SI									DEL		Delete
DLE											Data Link Escape