MICRO LOGIC CORP.
HACKENSACK, NJ

Z80 CPU
MICROPROCESSOR INSTANT REFERENCE CARD

MICRO CHART

Example of reading instruction set tables: ADC A,A ... ADC A, - entry says to see table; table shows opcode 8F, 4 states; and flag code 'A' which is defined under 'Flag Codes'. ADC HL,BC ... 2 byte opcode is ED,4A; flag code is H; takes 15 states. CALL C, address ... opcode is DC followed by 2 byte address; flag code is Z; states are described by note 5.

Instruction Set

A	ADC	A,—	A		
	ADC	HL,BC	H15		
	ADC	HL,DE	H15		
	ADC	HL,HL	H15		
	ADC	HL,SP	H15		
	ADD	A,—	A		
	ADD	HL,BC	G11		
	ADD	HL,DE	G11		
	ADD	HL,HL	G11		
	ADD	HL,SP	G11		
	ADD	IX,BC	G15		
	ADD	IX,DE	G15		
	ADD	IX,IX	G15		
	ADD	IX,SP	G15		
	ADD	IY,BC	G15		
	ADD	IY,DE	G15		
	ADD	IY,IY	G15		
	ADD	IY,SP	G15		
	AND	—	C		
	BIT	—	V		
C	CALL	aa	Z17		
	CALL	C,aa	Z(5)		
	CALL	M,aa	Z(5)		
	CALL	NC,aa	Z(5)		
	CALL	NZ,aa	Z(5)		
	CALL	P,aa	Z(5)		
	CALL	PE,aa	Z(5)		
	CALL	PO,aa	Z(5)		
	CALL	Z,aa	Z(5)		
	CCF		G4		
	CP	—	B		
	CPD		T16		
	CPDR		T(1)		
	CPI		T16		
	CPIR		T(1)		
	CPL		N4		
D	DAA		M4		
	DEC	(HL)	F11		
	DEC	(IX+d)	F23		
	DEC	(IY+d)	F23		
	DEC	A	F4		
	DEC	B	F4		
	DEC	BC	Z6		
	DEC	C	F4		
	DEC	D	F4		
	DEC	DE	Z6		
	DEC	E	F4		
	DEC	H	F4		
	DEC	HL	Z6		
	DEC	IX	Z10		
	DEC	IY	Z10		
	DEC	L	F4		
	DEC	SP	Z6		
	DI		Z4		
	DJNZ	d	Z(2)		
	EI		Z4		
E	EX	(SP),HL	Z19		
	EX	(SP),IX	Z23		
	EX	(SP),IY	Z23		
	EX	AF,AF'	Z4		
	EX	DE,HL	Z4		
	EXX		Z4		
	HALT		Z4		
I	IM	0	Z8		
	IM	1	Z8		
	IM	2	Z8		
	IN	A,(C)	W12		
	IN	A,(n)	Z11		
	IN	B,(C)	W12		
	IN	C,(C)	W12		
	IN	D,(C)	W12		
	IN	E,(C)	W12		
	IN	H,(C)	W12		
	IN	L,(C)	W12		
	INC	(HL)	E11		
	INC	(IX+d)	E23		
	INC	(IY+d)	E23		
	INC	A	E4		
	INC	B	E4		
	INC	BC	Z6		
	INC	C	E4		
	INC	D	E4		
	INC	DE	Z6		
	INC	E	E4		
	INC	H	E4		
	INC	HL	Z6		
	INC	IX	Z10		
	INC	IY	Z10		
	INC	L	E4		
	INC	SP	Z6		
	IND		P16		
	INDR		Q(1)		
	INI		P16		
	INIR		Q(1)		
J	JP	(HL)	Z4		
	JP	(IX)	Z8		
	JP	(IY)	Z8		
	JP	aa	Z10		
	JP	C,aa	Z10		
	JP	M,aa	Z10		
	JP	NC,aa	Z10		
	JP	NZ,aa	Z10		
	JP	P,aa	Z10		
	JP	PE,aa	Z10		
	JP	PO,aa	Z10		
	JP	Z,aa	Z10		
	JR	C,d	Z(3)		
	JR	d	Z12		
	JR	NC,d	Z(3)		
	JR	NZ,d	Z(3)		
	JR	Z,d	Z(3)		
L	LD	(BC),A	Z7		
	LD	(DE),A	Z7		
	LD	(HL),A	Z7		
	LD	(HL),B	Z7		
	LD	(HL),C	Z7		
	LD	(HL),D	Z7		
	LD	(HL),E	Z7		
	LD	(HL),H	Z7		
	LD	(HL),L	Z7		
	LD	(HL),n	Z10		
	LD	(IX+d),A	Z19		
	LD	(IX+d),B	Z19		
	LD	(IX+d),C	Z19		
	LD	(IX+d),D	Z19		
	LD	(IX+d),E	Z19		
	LD	(IX+d),H	Z19		
	LD	(IX+d),L	Z19		
	LD	(IX+d),n	Z19		
	LD	(IY+d),A	Z19		
	LD	(IY+d),B	Z19		
	LD	(IY+d),C	Z19		
	LD	(IY+d),D	Z19		
	LD	(IY+d),E	Z19		
	LD	(IY+d),H	Z19		
	LD	(IY+d),L	Z19		
	LD	(IY+d),n	Z19		
	LD	(aa),A	Z13		
	LD	(aa),BC	Z20		
	LD	(aa),DE	Z20		
	LD	(aa),HL	Z16		
	LD	(aa),IX	Z20		
	LD	(aa),IY	Z20		
	LD	(aa) SP	Z20		
	LD	A,(BC)	Z7		
	LD	A,(DE)	Z7		
	LD	A,(aa)	Z13		
	LD	A,I	U9		
	LD	A,R	U9		
	LD	A,—	Z		
	LD	B,—	Z		
	LD	BC,(aa)	Z20		
	LD	BC,aa	Z10		
	LD	C,—	Z		
	LD	D,—	Z		
	LD	DE,(aa)	Z20		
	LD	DE,aa	Z10		
	LD	E,—	Z		
	LD	H,—	Z		
	LD	HL,(aa)	Z16		
	LD	HL,aa	Z10		
	LD	I,A	Z9		
	LD	IX,(aa)	Z20		
	LD	IX,aa	Z14		
	LD	IY,(aa)	Z20		
	LD	IY,aa	Z14		
	LD	L,—	Z		
	LD	R,A	Z9		
	LD	SP,(aa)	Z20		
	LD	SP,HL	Z6		
	LD	SP,IX	Z10		
	LD	SP,IY	Z10		
	LD	SP,aa	Z10		
	LDD		R16		
	LDDR		S(1)		
	LDI		R16		
	LDIR		S(1)		
	NEG		B8		
	NOP		Z4		
O	OR	—	D		
	OTDR		Q(1)		
	OTIR		Q(1)		
	OUT	(C),A	Z12		
	OUT	(C),B	Z12		
	OUT	(C),C	Z12		
	OUT	(C),D	Z12		
	OUT	(C),E	Z12		
	OUT	(C),H	Z12		
	OUT	(C),L	Z12		
	OUT	(n),A	Z11		
	OUTD		P16		
	OUT I		P16		
P	POP	AF	Z10		
	POP	BC	Z10		
	POP	DE	Z10		
	POP	HL	Z10		
	POP	IX	Z14		
	POP	IY	Z14		
	PUSH	AF	Z11		
	PUSH	BC	Z11		
	PUSH	DE	Z11		
	PUSH	HL	Z11		
	PUSH	IX	Z15		
	PUSH	IY	Z15		
R	RES	—	Z		
	RET		Z10		
	RET	C	Z(4)		
	RET	M	Z(4)		
	RET	NC	Z(4)		
	RET	NZ	Z(4)		
	RET	P	Z(4)		
	RET	PE	Z(4)		
	RET	PO	Z(4)		
	RET	Z	Z(4)		
	RETI		Z14		
	RETN		Z14		
	RL	—	K		
	RLA		J4		
	RLC	—	K		
	RLCA		J4		
	RLD		L18		
	RR	—	K		
	RRA		J4		
	RRC	—	K		
	RRCA		J4		
	RRD		L18		
	RST	00H	Z11		
	RST	08H	Z11		
	RST	10H	Z11		
	RST	18H	Z11		
	RST	20H	Z11		
	RST	28H	Z11		
	RST	30H	Z11		
	RST	38H	Z11		
S	SBC	A,—	B		
	SBC	HL,BC	I15		
	SBC	HL,DE	I15		
	SBC	HL,HL	I15		
	SBC	HL,SP	I15		
	SCF		Q4		
	SET	—	Z		
	SLA	—	K		
	SRA	—	K		
	SRL	—	K		
	SUB	—	B		
	XOR	—	D		

	A	B	C	D	E	H	L	(HL)	(IX+d)	(IY+d)	
BIT 0,	CB,47	CB,40	CB,41	CB,42	CB,43	CB,44	CB,45	CB,46	DD,CB,d,46	FD,CB,d,46	V
BIT 1,	CB,4F	CB,48	CB,49	CB,4A	CB,4B	CB,4C	CB,4D	CB,4E	DD,CB,d,4E	FD,CB,d,4E	V
BIT 2,	CB,57	CB,50	CB,51	CB,52	CB,53	CB,54	CB,55	CB,56	DD,CB,d,56	FD,CB,d,56	V
BIT 3,	CB,5F	CB,58	CB,59	CB,5A	CB,5B	CB,5C	CB,5D	CB,5E	DD,CB,d,5E	FD,CB,d,5E	V
BIT 4,	CB,67	CB,60	CB,61	CB,62	CB,63	CB,64	CB,65	CB,66	DD,CB,d,66	FD,CB,d,66	V
BIT 5,	CB,6F	CB,68	CB,69	CB,6A	CB,6B	CB,6C	CB,6D	CB,6E	DD,CB,d,6E	FD,CB,d,6E	V
BIT 6,	CB,77	CB,70	CB,71	CB,72	CB,73	CB,74	CB,75	CB,76	DD,CB,d,76	FD,CB,d,76	V
BIT 7,	CB,7F	CB,78	CB,79	CB,7A	CB,7B	CB,7C	CB,7D	CB,7E	DD,CB,d,7E	FD,CB,d,7E	V
STATES:				8				12	20		

	A	B	C	D	E	H	L	(HL)	(IX+d)	(IY+d)	
RES 0,	CB,87	CB,80	CB,81	CB,82	CB,83	CB,84	CB,85	CB,86	DD,CB,d,86	FD,CB,d,86	Z
RES 1,	CB,8F	CB,88	CB,89	CB,8A	CB,8B	CB,8C	CB,8D	CB,8E	DD,CB,d,8E	FD,CB,d,8E	Z
RES 2,	CB,97	CB,90	CB,91	CB,92	CB,93	CB,94	CB,95	CB,96	DD,CB,d,96	FD,CB,d,96	Z
RES 3,	CB,9F	CB,98	CB,99	CB,9A	CB,9B	CB,9C	CB,9D	CB,9E	DD,CB,d,9E	FD,CB,d,9E	Z
RES 4,	CB,A7	CB,A0	CB,A1	CB,A2	CB,A3	CB,A4	CB,A5	CB,A6	DD,CB,d,A6	FD,CB,d,A6	Z
RES 5,	CB,AF	CB,A8	CB,A9	CB,AA	CB,AB	CB,AC	CB,AD	CB,AE	DD,CB,d,AE	FD,CB,d,AE	Z
RES 6,	CB,B7	CB,B0	CB,B1	CB,B2	CB,B3	CB,B4	CB,B5	CB,B6	DD,CB,d,B6	FD,CB,d,B6	Z
RES 7,	CB,BF	CB,B8	CB,B9	CB,BA	CB,BB	CB,BC	CB,BD	CB,BE	DD,CB,d,BE	FD,CB,d,BE	Z
SET 0,	CB,C7	CB,C0	CB,C1	CB,C2	CB,C3	CB,C4	CB,C5	CB,C6	DD,CB,d,C6	FD,CB,d,C6	Z
SET 1,	CB,CF	CB,C8	CB,C9	CB,CA	CB,CB	CB,CC	CB,CD	CB,CE	DD,CB,d,CE	FD,CB,d,CE	Z
SET 2,	CB,D7	CB,D0	CB,D1	CB,D2	CB,D3	CB,D4	CB,D5	CB,D6	DD,CB,d,D6	FD,CB,d,D6	Z
SET 3,	CB,DF	CB,D8	CB,D9	CB,DA	CB,DB	CB,DC	CB,DD	CB,DE	DD,CB,d,DE	FD,CB,d,DE	Z
SET 4,	CB,E7	CB,E0	CB,E1	CB,E2	CB,E3	CB,E4	CB,E5	CB,E6	DD,CB,d,E6	FD,CB,d,E6	Z
SET 5,	CB,EF	CB,E8	CB,E9	CB,EA	CB,EB	CB,EC	CB,ED	CB,EE	DD,CB,d,EE	FD,CB,d,EE	Z
SET 6,	CB,F7	CB,F0	CB,F1	CB,F2	CB,F3	CB,F4	CB,F5	CB,F6	DD,CB,d,F6	FD,CB,d,F6	Z
SET 7,	CB,FF	CB,F8	CB,F9	CB,FA	CB,FB	CB,FC	CB,FD	CB,FE	DD,CB,d,FE	FD,CB,d,FE	Z
STATES:				8				15	23		

	A(8)	B	C	D	E	H	L	(HL)	(IX+d)	(IY+d)	
RLC	CB,07	CB,00	CB,01	CB,02	CB,03	CB,04	CB,05	CB,06	DD,CB,d,06	FD,CB,d,06	K
RRC	CB,0F	CB,08	CB,09	CB,0A	CB,0B	CB,0C	CB,0D	CB,0E	DD,CB,d,0E	FD,CB,d,0E	K
RL	CB,17	CB,10	CB,11	CB,12	CB,13	CB,14	CB,15	CB,16	DD,CB,d,16	FD,CB,d,16	K
RR	CB,1F	CB,18	CB,19	CB,1A	CB,1B	CB,1C	CB,1D	CB,1E	DD,CB,d,1E	FD,CB,d,1E	K
SLA	CB,27	CB,20	CB,21	CB,22	CB,23	CB,24	CB,25	CB,26	DD,CB,d,26	FD,CB,d,26	K
SRA	CB,2F	CB,28	CB,29	CB,2A	CB,2B	CB,2C	CB,2D	CB,2E	DD,CB,d,2E	FD,CB,d,2E	K
SRL	CB,3F	CB,38	CB,39	CB,3A	CB,3B	CB,3C	CB,3D	CB,3E	DD,CB,d,3E	FD,CB,d,3E	K
STATES:				8				15	23		

Flag Codes

	C	Z	P/V	S	N	H	
A	C	Z	V	S	0	H	
B	C	Z	V	S	1	H	
C	0	Z	P	S	0	1	
D	0	Z	P	S	0	0	
E	=	Z	V	S	0	H	
F	=	Z	V	S	1	H	
G	C	=	=	=	0	U	
H	C	Z	V	S	0	U	
I	C	Z	V	S	1	U	
J	C	=	=	=	0	0	
K	C	Z	P	S	0	0	
L	=	Z	P	S	0	0	
M	C	Z	P	S	=	H	
N	=	=	=	=	1	1	
O	1	=	=	=	0	0	
P	U	F	U	U	1	U	(1)
Q	U	1	U	U	1	U	
R	=	U	F	U	0	0	(2)
S	=	U	0	U	0	0	
T	=	F	F	S	1	U	(3)
U	=	Z	F	S	0	0	(4)
V	=	Z	U	U	0	1	(5)
W	=	Z	P	S	0	U	
X							
Y							
Z	=	=	=	=	=	=	

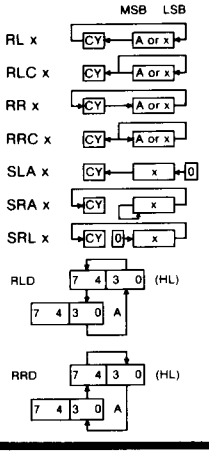
Codes:
0: reset
1: set
C: Carry*
F: Footnote
H: Half carry*
N: Add/Sub*
P: Parity*
S: Sign*
U: Undefined
V: oVerflow*
Z: Zero*
=: not affected

* Indicated flag affected by result

(1)Z=1 iff B becomes 0
(2)PV=0 iff BC becomes 0
(3)PV=0 iff BC becomes 0 and Z=1 iff A=(HL)
(4)PV=IFF2
(5)Z=b̄it

	A	B	C	D	E	H	L	(HL)	n	(IX+d)	(IY+d)	
ADC A,	8F	88	89	8A	8B	8C	8D	8E	CE,n	DD,8E,d	FD,8E,d	A
ADD A,	87	80	81	82	83	84	85	86	C6,n	DD,86,d	FD,86,d	A
AND	A7	A0	A1	A2	A3	A4	A5	A6	E6,n	DD,A6,d	FD,A6,d	C
CP	BF	B8	B9	BA	BB	BC	BD	BE	FE,n	DD,BE,d	FD,BE,d	B
OR	B7	B0	B1	B2	B3	B4	B5	B6	F6,n	DD,B6,d	FD,B6,d	D
SBC A,	9F	98	99	9A	9B	9C	9D	9E	DE,n	DD,9E,d	FD,9E,d	B
SUB	97	90	91	92	93	94	95	96	D6,n	DD,96,d	FD,96,d	B
XOR	AF	A8	A9	AA	AB	AC	AD	AE	EE,n	DD,AE,d	FD,AE,d	D
LD A,	7F	78	79	7A	7B	7C	7D	7E	3E,n	DD,7E,d	FD,7E,d	Z
LD B,	47	40	41	42	43	44	45	46	06,n	DD,46,d	FD,46,d	Z
LD C,	4F	48	49	4A	4B	4C	4D	4E	0E,n	DD,4E,d	FD,4E,d	Z
LD D,	57	50	51	52	53	54	55	56	16,n	DD,56,d	FD,56,d	Z
LD E,	5F	58	59	5A	5B	5C	5D	5E	1E,n	DD,5E,d	FD,5E,d	Z
LD H,	67	60	61	62	63	64	65	66	26,n	DD,66,d	FD,66,d	Z
LD L,	6F	68	69	6A	6B	6C	6D	6E	2E,n	DD,6E,d	FD,6E,d	Z
STATES:				4				7	19			

Rotates and Shifts



Addressing

n	n is immediate 8-bit data.
aa	aa is immediate 16-bit data or address to CALL/to JP to.
(aa)	aa is address of data.
(rr)	16-bit reg rr holds address of data or address to CALL or to JP to.
(n)	n is port number.
(r)	8-bit reg r holds port number.
(IX+d)	IX+d is address of data (d is a 1 byte signed displacement).
d	In relative jumping, address to jump to is d + address of next instruction (d is signed).

Full 2 byte addresses in code, stack, and data areas are stored low byte followed by high byte. Thus JP 1234H is: C3,34,12.

SP points to used byte at top of stack. PUSH decrements SP by 2.

Notes

(1) 21 except 16 at termination
(2) 13 except 8 at termination
(3) 12 for success; 7 for failure
(4) 11 for success; 5 for failure
(5) 17 for success; 10 for failure
(6) A to A15..A8 and n to A7..A0
(7) B to A15..A8 and C to A7..A0
(8) See faster version of 'Rotate A' instructions

Intentionally Blank

AUTHOR
JAMES D. LEWIS

MI0033893817

#104A

MICRO LOGIC CORP.
HACKENSACK, NJ

Z80 CPU
MICROPROCESSOR INSTANT REFERENCE CARD

MICRO CHART

Single-Byte-Opcode to Instruction Conversion

LSD ⇒

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	NOP	LD BC,nn	LD (BC),A	INC BC	INC B	DEC B	LD B,n	RLCA	EX AF,AF'	ADD HL,BC	LD A,(BC)	DEC BC	INC C	DEC C	LD C,n	RRCA	0
1	DJNZ n	LD DE,nn	LD (DE),A	INC DE	INC D	DEC D	LD D,n	RLA	JR n	ADD HL,DE	LD A (DE)	DEC DE	INC E	DEC E	LD E,n	RRA	1
2	JR NZ,n	LD HL,nn	LD (nn),HL	INC HL	INC H	DEC H	LD H,n	DAA	JR Z,n	ADD HL,HL	LD HL,(nn)	DEC HL	INC L	DEC L	LD L,n	CPL	2
3	JR NC,n	LD SP,nn	LD (nn),A	INC SP	INC (HL)	DEC (HL)	LD (HL),n	SCF	JR C,n	ADD HL,SP	LD A,(nn)	DEC SP	INC A	DEC A	LD A,n	CCF	3
4	LD B,B	LD B,C	LD B,D	LD B,E	LD B,H	LD B,L	LD B,(HL)	LD B,A	LD C,B	LD C,C	LD C,D	LD C,E	LD C,H	LD C,L	LD C,(HL)	LD C,A	4
5	LD D,B	LD D,C	LD D,D	LD D,E	LD D,H	LD D,L	LD D,(HL)	LD D,A	LD E,B	LD E,C	LD E,D	LD E,E	LD E,H	LD E,L	LD E,(HL)	LD E,A	5
6	LD H,B	LD H,C	LD H,D	LD H,E	LD H,H	LD H,L	LD H,(HL)	LD H,A	LD L,B	LD L,C	LD L,D	LD L,E	LD L,H	LD L,L	LD L,(HL)	LD L,A	6
7	LD (HL),B	LD (HL),C	LD (HL),D	LD (HL),E	LD (HL),H	LD (HL),L	HALT	LD (HL),A	LD A,B	LD A,C	LD A,D	LD A,E	LD A,H	LD A,L	LD A,(HL)	LD A,A	7
8	ADD A,B	ADD A,C	ADD A,D	ADD A,E	ADD A,H	ADD A,L	ADD A,(HL)	ADD A,A	ADC A,B	ADC A,C	ADC A,D	ADC A,E	ADC A,H	ADC A,L	ADC A,(HL)	ADC A,A	8
9	SUB B	SUB C	SUB D	SUB E	SUB H	SUB L	SUB (HL)	SUB A	SBC A,B	SBC A,C	SBC A,D	SBC A,E	SBC A,H	SBC A,L	SBC A,(HL)	SBC A,A	9
A	AND B	AND C	AND D	AND E	AND H	AND L	AND (HL)	AND A	XOR B	XOR C	XOR D	XOR E	XOR H	XOR L	XOR (HL)	XOR A	A
B	OR B	OR C	OR D	OR E	OR H	OR L	OR (HL)	OR A	CP B	CP C	CP D	CP E	CP H	CP L	CP (HL)	CP A	B
C	RET NZ	POP BC	JP NZ,nn	JP nn	CALL NZ,nn	PUSH BC	ADD A,n	RST 00H	RET Z	RET	JP Z,nn	table	CALL Z,nn	CALL nn	ADC A,n	RST 08H	C
D	RET NC	POP DE	JP NC,nn	OUT (n), A	CALL NC,nn	PUSH DE	SUB n	RST 10H	RET C	EXX	JP C,nn	IN A,(n)	CALL C,nn	table	SBC A,n	RST 18H	D
E	RET PO	POP HL	JP PO,nn	EX (SP), HL	CALL PO,nn	PUSH HL	AND n	RST 20H	RET PE	JP (HL)	JP PE,nn	EX DE,HL	CALL PE,nn	table	XOR n	RST 28H	E
F	RET P	POP AF	JP P,nn	DI	CALL P,nn	PUSH AF	OR n	RST 30H	RET M	LD SP,HL	JP M,nn	EI	CALL M,nn	table	CP n	RST 38H	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Multi-Byte-Opcode to Instruction Conversion

CB00	RLC B	ED40	IN B,(C)	%%09	ADD XY,BC	%%CBd06	RLC (XY+d)	
CB01	RLC C	ED41	OUT (C),B	%%19	ADD XY,DE	%%CBd0E	RRC (XY+d)	
CB02	RLC D	ED42	SBC HL,BC	%%21aa	LD XY,aa	%%CBd16	RL (XY+d)	
CB03	RLC E	ED43aa	LD (aa),BC	%%22aa	LD (aa),XY	%%CBd1E	RR (XY+d)	
CB04	RLC H	ED44	NEG	%%23	INC XY	%%CBd26	SLA (XY+d)	
CB05	RLC L	ED45	RETN	%%29	ADD XY,XY	%%CBd2E	SRA (XY+d)	
CB06	RLC (HL)	ED46	IM 0	%%2Aaa	LD XY,(aa)	%%CBd3E	SRL (XY+d)	
CB07	RLC A	ED47	LD I,A	%%2B	DEC XY	%%CBd46	BIT 0,(XY+d)	
CB08	RRC B	ED48	IN C,(C)	%%34d	INC (XY+d)	%%CBd4E	BIT 1,(XY+d)	
CB09	RRC C	ED49	OUT (C),C	%%35d	DEC (XY+d)	%%CBd56	BIT 2,(XY+d)	
CB0A	RRC D	ED4A	ADC HL,BC	%%36dn	LD (XY+d),n	%%CBd5E	BIT 3,(XY+d)	
CB0B	RRC E	ED4Baa	LD BC,(aa)	%%39	ADD XY,SP	%%CBd66	BIT 4,(XY+d)	
CB0C	RRC H	ED4D	RETI	%%46d	LD B,(XY+d)	%%CBd6E	BIT 5,(XY+d)	
CB0D	RRC L	ED4F	LD R,A	%%4Ed	LD C,(XY+d)	%%CBd76	BIT 6,(XY+d)	
CB0E	RRC (HL)	ED50	IN D,(C)	%%56d	LD D,(XY+d)	%%CBd7E	BIT 7,(XY+d)	
CB0F	RRC A	ED51	OUT (C),D	%%5Ed	LD E,(XY+d)	%%CBd86	RES 0,(XY+d)	
CB10	RL B	ED52	SBC HL,DE	%%66d	LD H,(XY+d)	%%CBd8E	RES 1,(XY+d)	
CB11	RL C	ED53aa	LD (aa),DE	%%6Ed	LD L,(XY+d)	%%CBd96	RES 2,(XY+d)	
CB12	RL D	ED56	IM 1	%%70d	LD (XY+d),B	%%CBd9E	RES 3,(XY+d)	
CB13	RL E	ED57	LD A,I	%%71d	LD (XY+d),C	%%CBdA6	RES 4,(XY+d)	
CB14	RL H	ED58	IN E,(C)	%%72d	LD (XY+d),D	%%CBdAE	RES 5,(XY+d)	
CB15	RL L	ED59	OUT (C),E	%%73d	LD (XY+d),E	%%CBdB6	RES 6,(XY+d)	
CB16	RL (HL)	ED5A	ADC HL,DE	%%74d	LD (XY+d),H	%%CBdBE	RES 7,(XY+d)	
CB17	RL A	ED5Baa	LD DE,(aa)	%%75d	LD (XY+d),L	%%CBdC6	SET 0,(XY+d)	
CB18	RR B	ED5E	IM 2	%%77d	LD (XY+d),A	%%CBdCE	SET 1,(XY+d)	
CB19	RR C	ED5F	LD A,R	%%7Ed	LD A,(XY+d)	%%CBdD6	SET 2,(XY+d)	
CB1A	RR D	ED60	IN H,(C)	%%86d	ADD A,(XY+d)	%%CBdDE	SET 3,(XY+d)	
CB1B	RR E	ED61	OUT (C),H	%%8Ed	ADC A,(XY+d)	%%CBdE6	SET 4,(XY+d)	
CB1C	RR H	ED62	SBC HL,HL	%%96d	SUB (XY+d)	%%CBdEE	SET 5,(XY+d)	
CB1D	RR L	ED67	RRD	%%9Ed	SBC A,(XY+d)	%%CBdF6	SET 6,(XY+d)	
CB1E	RR (HL)	ED68	IN L,(C)	%%A6d	AND (XY+d)	%%CBdFE	SET 7,(XY+d)	
CB1F	RR A	ED69	OUT (C),L	%%AEd	XOR (XY+d)			
CB20	SLA B	ED6A	ADC HL,HL	%%B6d	OR (XY+d)	%%E1	POP XY	
CB21	SLA C	ED6F	RLD	%%BEd	CP (XY+d)	%%E3	EX (SP),XY	
CB22	SLA D	ED72	SBC HL,SP	%% means DD or FD and for DD, XY means IX for FD, XY means IY		%%E5	PUSH XY	
CB23	SLA E	ED73aa	LD (aa),SP			%%E9	JP (XY)	
CB24	SLA H	ED78	IN A,(C)			%%F9	LD SP,XY	
CB25	SLA L	ED79	OUT (C),A					
CB26	SLA (HL)	ED7A	ADC HL,SP					
CB27	SLA A	ED7Baa	LD SP,(aa)					
CB28	SRA B	EDA0	LDI					
CB29	SRA C	EDA1	CPI					
CB2A	SRA D	EDA2	INI					
CB2B	SRA E	EDA3	OUTI					
CB2C	SRA H	EDA8	LDD					
CB2D	SRA L	EDA9	CPD					
CB2E	SRA (HL)	EDAA	IND					
CB2F	SRA A	EDAB	OUTD					
CB38	SRL B	EDB0	LDIR					
CB39	SRL C	EDB1	CPIR					
CB3A	SRL D	EDB2	INIR					
CB3B	SRL E	EDB3	OTIR					
CB3C	SRL H	EDB8	LDDR					
CB3D	SRL L	EDB9	CPDR					
CB3E	SRL (HL)	EDBA	INDR					
CB3F	SRL A	EDBB	OTDR					
CB40	see BIT							
....	see RES							
CBFF	see SET							

Hex and Decimal Conversion

LSD ⟶

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	0
1	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	1
2	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47	2
3	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63	3
4	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	4
5	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95	5
6	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111	6
7	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127	7
8	128	129	130	131	132	133	134	135	136	137	138	139	140	141	142	143	8
9	144	145	146	147	148	149	150	151	152	153	154	155	156	157	158	159	9
A	160	161	162	163	164	165	166	167	168	169	170	171	172	173	174	175	A
B	176	177	178	179	180	181	182	183	184	185	186	187	188	189	190	191	B
C	192	193	194	195	196	197	198	199	200	201	202	203	204	205	206	207	C
D	208	209	210	211	212	213	214	215	216	217	218	219	220	221	222	223	D
E	224	225	226	227	228	229	230	231	232	233	234	235	236	237	238	239	E
F	240	241	242	243	244	245	246	247	248	249	250	251	252	253	254	255	F
	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	

Powers of Two

1	2	9	512
2	4	10	1,024
3	8	11	2,048
4	16	12	4,096
5	32	13	8,192
6	64	14	16,384
7	128	15	32,768
8	256	16	65,536

17	131,072
18	262,144
19	524,288
20	1,048,576
21	2,097,152
22	4,194,304
23	8,388,608
24	16,777,216

Unsigned Comparisons

example: CP B

A < B	JP C,YES
A ≤ B	JP C,YES ① JP Z,YES
A = B	JP Z,YES
A ≠ B	JP NZ,YES
A ≥ B	JP NC,YES
A > B	JR C,3 ① JP NZ,YES

YES represents label for code to be executed if condition is true. Internally, A-B is computed to determine flags as for 'SUB B'.

①Requires both instructions.

Status Flags

MSB						LSB
S	Z	-	H	-	P/V	N	C

S = Sign (MSB) of result
Z = 1 when result is Zero
H = Half carry from bit 3
P/V = 1 = Parity even for logic op or oVerflow for arithmetic op
N = 1 when last op was subtract (0 for add)
C = Carry (CY)

Interrupts and Reset

Falling edge sensitive NMI does a RST 66H regardless of IFF1, 2 (Interrupt Flip Flop).

If interrupts are enabled (IFF1=1), low level sensitive INT depends on mode:

MODE 0: Interrupting device puts instruction on bus (e.g. RST or CALL). Takes 2 extra time states.

MODE 1: Does a RST 38H (Z13).

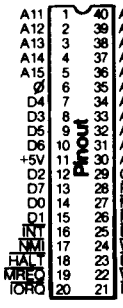
MODE 2: Location pointed to by

15	8 7	0
I	ivi	0

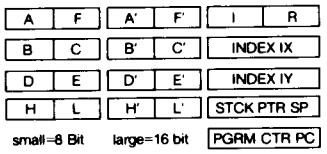
and next hold vector of service subroutine. ivi (7 bit int vector index) is put on data bus by interrupting device (Z19).

IFF1 and IFF2 are both cleared by INT or DI. Both are set by EI. NMI clears IFF1. RETN loads IFF1 from IFF2. LD A,I and LD A,R set P/V flag to IFF2. Reset sets PC=0, IFF1=IFF2=0, I=0, R=0, MODE =0.

Pinout



Registers



A=Accumulator
F=Flags
I= Interrupt vector
R=Memory refresh

When AF,BC,DE,HL used as pairs A,B,D,H are high order.

General Instruction Description (except shifts)

ADC x, y	Add y+CY to x.
ADD x, y	Add y to x.
AND x	AND x to A.
BIT b, x	Test bit b of x.
CALL c, x	If condition c is true call subroutine at x.
CALL x	Call subroutine at x (push PC and jump to x).
CCF	Complement carry flag.
CP x	Compare A with x (see "Unsigned Comparisons")
CPD	Compare A with (HL); DEC HL; DEC BC.
CPDR	Like CPD, but repeat until A=(HL) or BC=0.
CPI	Compare A with (HL); INC HL;DEC BC.
CPIR	Like CPI, but repeat until A=(HL) or BC=0.
CPL	Complement A (1's comp.).
DAA	Decimal adjust A (after add or sub of BCD data).
DEC x	Decrement x by 1.
DI	Disable interrupts.
DJNZ d	Decrement B, jump relative by d if B not zero.
EI	Enable interrupts after next instruction.
EX x,y	Exchange x with y.
EXX	Exchange BC, DE, HL with BC', DE', HL'.
HALT	Halt (wait for interrupt or reset).
IM x	Set interrupt mode to x.
IN A, (n)	Input port n into A (6).
IN r, (C)	Input port (C) into r (7).
INC x	Increment x by 1.
IND	Load (HL) from port (C); DEC B; DEC HL; (7).
INDR	Like IND, but repeat until B=0 (7).
INI	Load (HL) from port (C); DEC B; INC HL; (7).
INIR	Like INI, but repeat until B=0 (7).
JP c, x	If condition c is true jump to location x.
JP x	Jump to location x.
JR c, d	If condition c is true jump relative by d.
JR d	Jump relative by d.
LD x, y	Load x with y (move y to x).
LDD	Load (DE) with (HL); DEC DE; DEC HL; DEC BC.
LDDR	Like LDD, but repeat until BC=0.
LDI	Load (DE) with (HL); INC DE; INC HL; DEC BC.
LDIR	Like LDI, but repeat until BC=0
NEG	Negate A (2's comp.).
NOP	No operation.
OR x	OR x to A.
OTDR	Like OUTD, but repeat until B=0 (7).
OTIR	Like OUTI, but repeat until B=0 (7).
OUT (C), r	Output r to port (C) (7).
OUT (n), A	Output A to port n (7).
OUTD	Output (HL) to port (C); DEC B; DEC HL; (7).
OUTI	Output (HL) to port (C); DEC B; INC HL; (7).
POP x	Pop x from top of stack updating SP.
PUSH x	Push x onto top of stack updating SP.
RES b, x	Reset bit b of x (to 0).
RET	Return from subroutine (pop PC)
RET c	If condition c is true return from subroutine.
RETI	Return from interrupt.
RETN	Return from NMI (see "Interrupts").
RST x	Call subroutine at x (1 byte inst).
SBC x, y	Subtract y+CY from x.
SCF	Set carry flag (to 1).
SET b, x	Set bit b of x (to 1).
SUB x	Subtract x from A.
XOR x	XOR x to A.

ASCII Character Set

LSD \ MSD	0	1	2	3	4	5	6	7
	000	001	010	011	100	101	110	111
0	0000	NUL	DLE	SP	0	@	P	`	p
1	0001	SOH	DC1	!	1	A	Q	a	q
2	0010	STX	DC2	"	2	B	R	b	r
3	0011	ETX	DC3	#	3	C	S	c	s
4	0100	EOT	DC4	$	4	D	T	d	t
5	0101	ENQ	NAK	%	5	E	U	e	u
6	0110	ACK	SYN	&	6	F	V	f	v
7	0111	BEL	ETB	'	7	G	W	g	w
8	1000	BS	CAN	(	8	H	X	h	x
9	1001	HT	EM	)	9	I	Y	i	y
A	1010	LF	SUB	*	:	J	Z	j	z
B	1011	VT	ESC	+	;	K	[	k	{
C	1100	FF	FS	,	<	L	\	l	|
D	1101	CR	GS	-	=	M	]	m	}
E	1110	SO	RS	.	>	N	↑	n	~
F	1111	SI	US	/	?	O	←	o	DEL

INSTANT ACCESS

MICRO CHARTS: Z80, 6502-65XX, 8080-8085, 8086-8088, 8048 Family, 54/7400 TTL pinouts, BASIC Algorithms, Wordstar, Electronic Components, Sampling Statistics, C Language.