INTEL 8080 OPCODES

Opcode (oct)	Opcode (hex)	Instruction	size	flags	function
000	00	NOP	1		
001	01	LXI B,D16	3		B <- byte 3, C <- byte 2
002	02	STAX B	1		(BC) <- A
003	03	INX B	1		BC <- BC+1
004	04	INR B	1	Z, S, P, AC	B <- B+1
005	05	DCR B	1	Z, S, P, AC	B <- B-1
006	06	MVI B, D8	2		B <- byte 2
007	07	RLC	1	CY	A = A << 1; bit 0 = prev bit 7; CY = prev bit 7
010	08	-			
011	09	DAD B	1	CY	HL = HL + BC
012	0a	LDAX B	1		A <- (BC)
013	0b	DCX B	1		BC = BC-1
014	0c	INR C	1	Z, S, P, AC	C <- C+1
015	0d	DCR C	1	Z, S, P, AC	C <-C-1
016	0e	MVI C,D8	2		C <- byte 2
017	0f	RRC	1	CY	A = A >> 1; bit 7 = prev bit 0; CY = prev bit 0
012	0A	-			
013	0B	LXI D,D16	3		D <- byte 3, E <- byte 2
014	0C	STAX D	1		(DE) <- A
015	0D	INX D	1		DE <- DE + 1
016	0E	INR D	1	Z, S, P, AC	D <- D+1
017	0F	DCR D	1	Z, S, P, AC	D <- D-1
020	10	MVI D, D8	2		D <- byte 2
021	11	RAL	1	CY	A = A << 1; bit 0 = prev CY; CY = prev bit 7
022	12	-			
023	13	DAD D	1	CY	HL = HL + DE
032	1a	LDAX D	1		A <- (DE)
033	1b	DCX D	1		DE = DE-1
034	1c	INR E	1	Z, S, P, AC	E <-E+1
035	1d	DCR E	1	Z, S, P, AC	E <- E-1
036	1e	MVI E,D8	2		E <- byte 2
037	1f	RAR	1	CY	A = A >> 1; bit 7 = prev bit 7; CY = prev bit 0
024	14	-			
025	15	LXI H,D16	3		H <- byte 3, L <- byte 2

026	16	SHLD adr	3		(adr) <-L; (adr+1)<-H
027	17	INX H	1		HL <- HL + 1
030	18	INR H	1	Z, S, P, AC	H <- H+1
031	19	DCR H	1	Z, S, P, AC	H <- H-1
032	1A	MVI H,D8	2		H <- byte 2
033	1B	DAA	1		special
034	1C	-			
035	1D	DAD H	1	CY	HL = HL + HI
052	2a	LHLD adr	3		L <- (adr); H<-(adr+1)
053	2b	DCX H	1		HL = HL-1
054	2c	INR L	1	Z, S, P, AC	L <- L+1
055	2d	DCR L	1	Z, S, P, AC	L <- L-1
056	2e	MVI L, D8	2		L <- byte 2
057	2f	CMA	1		A <- !A
036	1E	-			
037	1F	LXI SP, D16	3		SP.hi <- byte 3, SP.lo <- byte 2
040	20	STA adr	3		(adr) <- A
041	21	INX SP	1		SP = SP + 1
042	22	INR M	1	Z, S, P, AC	(HL) <- (HL)+1
043	23	DCR M	1	Z, S, P, AC	(HL) <- (HL)-1
044	24	MVI M,D8	2		(HL) <- byte 2
045	25	STC	1	CY	CY = 1
046	26	-			
047	27	DAD SP	1	CY	HL = HL + SP
072	3a	LDA adr	3		A <- (adr)
073	3b	DCX SP	1		SP = SP-1
074	3c	INR A	1	Z, S, P, AC	A <- A+1
075	3d	DCR A	1	Z, S, P, AC	A <- A-1
076	3e	MVI A,D8	2		A <- byte 2
077	3f	CMC	1	CY	CY=!CY
050	28	MOV B,B	1		B <- B
051	29	MOV B,C	1		B <- C
052	2A	MOV B,D	1		B <- D
053	2B	MOV B,E	1		B <- E
054	2C	MOV B,H	1		B <- H
055	2D	MOV B,L	1		B <- L

056	2E	MOV B,M	1		B <- (HL)
057	2F	MOV B,A	1		B <- A
060	30	MOV C,B	1		C <- B
061	31	MOV C,C	1		C <- C
112	4a	MOV C,D	1		C <- D
113	4b	MOV C,E	1		C <- E
114	4c	MOV C,H	1		C <- H
115	4d	MOV C,L	1		C <- L
116	4e	MOV C,M	1		C <- (HL)
117	4f	MOV C,A	1		C <- A
062	32	MOV D,B	1		D <- B
063	33	MOV D,C	1		D <- C
064	34	MOV D,D	1		D <- D
065	35	MOV D,E	1		D <- E
066	36	MOV D,H	1		D <- H
067	37	MOV D,L	1		D <- L
070	38	MOV D,M	1		D <- (HL)
071	39	MOV D,A	1		D <- A
072	3A	MOV E,B	1		E <- B
073	3B	MOV E,C	1		E <- C
132	5a	MOV E,D	1		E <- D
133	5b	MOV E,E	1		E <- E
134	5c	MOV E,H	1		E <- H
135	5d	MOV E,L	1		E <- L
136	5e	MOV E,M	1		E <- (HL)
137	5f	MOV E,A	1		E <- A
074	3C	MOV H,B	1		H <- B
075	3D	MOV H,C	1		H <- C
076	3E	MOV H,D	1		H <- D
077	3F	MOV H,E	1		H <- E
100	40	MOV H,H	1		H <- H
101	41	MOV H,L	1		H <- L
102	42	MOV H,M	1		H <- (HL)
103	43	MOV H,A	1		H <- A
104	44	MOV L,B	1		L <- B
105	45	MOV L,C	1		L <- C

152	6a	MOV L,D	1		L <- D
153	6b	MOV L,E	1		L <- E
154	6c	MOV L,H	1		L <- H
155	6d	MOV L,L	1		L <- L
156	6e	MOV L,M	1		L <- (HL)
157	6f	MOV L,A	1		L <- A
106	46	MOV M,B	1		(HL) <- B
107	47	MOV M,C	1		(HL) <- C
110	48	MOV M,D	1		(HL) <- D
111	49	MOV M,E	1		(HL) <- E
112	4A	MOV M,H	1		(HL) <- H
113	4B	MOV M,L	1		(HL) <- L
114	4C	HLT	1		special
115	4D	MOV M,A	1		(HL) <- A
116	4E	MOV A,B	1		A <- B
117	4F	MOV A,C	1		A <- C
172	7a	MOV A,D	1		A <- D
173	7b	MOV A,E	1		A <- E
174	7c	MOV A,H	1		A <- H
175	7d	MOV A,L	1		A <- L
176	7e	MOV A,M	1		A <- (HL)
177	7f	MOV A,A	1		A <- A
120	50	ADD B	1	Z, S, P, CY, AC	A <- A + B
121	51	ADD C	1	Z, S, P, CY, AC	A <- A + C
122	52	ADD D	1	Z, S, P, CY, AC	A <- A + D
123	53	ADD E	1	Z, S, P, CY, AC	A <- A + E
124	54	ADD H	1	Z, S, P, CY, AC	A <- A + H
125	55	ADD L	1	Z, S, P, CY, AC	A <- A + L
126	56	ADD M	1	Z, S, P, CY, AC	A <- A + (HL)
127	57	ADD A	1	Z, S, P, CY, AC	A <- A + A
130	58	ADC B	1	Z, S, P, CY, AC	A <- A + B + CY
131	59	ADC C	1	Z, S, P, CY, AC	A <- A + C + CY
212	8a	ADC D	1	Z, S, P, CY, AC	A <- A + D + CY
213	8b	ADC E	1	Z, S, P, CY, AC	A <- A + E + CY
214	8c	ADC H	1	Z, S, P, CY, AC	A <- A + H + CY
215	8d	ADC L	1	Z, S, P, CY, AC	A <- A + L + CY

216	8e	ADC M	1	Z, S, P, CY, AC	A <- A + (HL) + CY
217	8f	ADC A	1	Z, S, P, CY, AC	A <- A + A + CY
132	5A	SUB B	1	Z, S, P, CY, AC	A <- A - B
133	5B	SUB C	1	Z, S, P, CY, AC	A <- A - C
134	5C	SUB D	1	Z, S, P, CY, AC	A <- A + D
135	5D	SUB E	1	Z, S, P, CY, AC	A <- A - E
136	5E	SUB H	1	Z, S, P, CY, AC	A <- A + H
137	5F	SUB L	1	Z, S, P, CY, AC	A <- A - L
140	60	SUB M	1	Z, S, P, CY, AC	A <- A + (HL)
141	61	SUB A	1	Z, S, P, CY, AC	A <- A - A
142	62	SBB B	1	Z, S, P, CY, AC	A <- A - B - CY
143	63	SBB C	1	Z, S, P, CY, AC	A <- A - C - CY
232	9a	SBB D	1	Z, S, P, CY, AC	A <- A - D - CY
233	9b	SBB E	1	Z, S, P, CY, AC	A <- A - E - CY
234	9c	SBB H	1	Z, S, P, CY, AC	A <- A - H - CY
235	9d	SBB L	1	Z, S, P, CY, AC	A <- A - L - CY
236	9e	SBB M	1	Z, S, P, CY, AC	A <- A - (HL) - CY
237	9f	SBB A	1	Z, S, P, CY, AC	A <- A - A - CY
240	a0	ANA B	1	Z, S, P, CY, AC	A <- A & B
241	a1	ANA C	1	Z, S, P, CY, AC	A <- A & C
242	a2	ANA D	1	Z, S, P, CY, AC	A <- A & D
243	a3	ANA E	1	Z, S, P, CY, AC	A <- A & E
244	a4	ANA H	1	Z, S, P, CY, AC	A <- A & H
245	a5	ANA L	1	Z, S, P, CY, AC	A <- A & L
246	a6	ANA M	1	Z, S, P, CY, AC	A <- A & (HL)
247	a7	ANA A	1	Z, S, P, CY, AC	A <- A & A
250	a8	XRA B	1	Z, S, P, CY, AC	A <- A ^ B
251	a9	XRA C	1	Z, S, P, CY, AC	A <- A ^ C
252	aa	XRA D	1	Z, S, P, CY, AC	A <- A ^ D
253	ab	XRA E	1	Z, S, P, CY, AC	A <- A ^ E
254	ac	XRA H	1	Z, S, P, CY, AC	A <- A ^ H
255	ad	XRA L	1	Z, S, P, CY, AC	A <- A ^ L
256	ae	XRA M	1	Z, S, P, CY, AC	A <- A ^ (HL)
257	af	XRA A	1	Z, S, P, CY, AC	A <- A ^ A
260	b0	ORA B	1	Z, S, P, CY, AC	A <- A | B
261	b1	ORA C	1	Z, S, P, CY, AC	A <- A | C

262	b2	ORA D	1	Z, S, P, CY, AC	A <- A | D
263	b3	ORA E	1	Z, S, P, CY, AC	A <- A | E
264	b4	ORA H	1	Z, S, P, CY, AC	A <- A | H
265	b5	ORA L	1	Z, S, P, CY, AC	A <- A | L
266	b6	ORA M	1	Z, S, P, CY, AC	A <- A | (HL)
267	b7	ORA A	1	Z, S, P, CY, AC	A <- A | A
270	b8	CMP B	1	Z, S, P, CY, AC	A - B
271	b9	CMP C	1	Z, S, P, CY, AC	A - C
272	ba	CMP D	1	Z, S, P, CY, AC	A - D
273	bb	CMP E	1	Z, S, P, CY, AC	A - E
274	bc	CMP H	1	Z, S, P, CY, AC	A - H
275	bd	CMP L	1	Z, S, P, CY, AC	A - L
276	be	CMP M	1	Z, S, P, CY, AC	A - (HL)
277	bf	CMP A	1	Z, S, P, CY, AC	A - A
300	c0	RNZ	1		if NZ, RET
301	c1	POP B	1		C <- (sp); B <- (sp+1); sp <- sp+2
302	c2	JNZ adr	3		if NZ, PC <- adr
303	c3	JMP adr	3		PC <= adr
304	c4	CNZ adr	3		if NZ, CALL adr
305	c5	PUSH B	1		(sp-2)<-C; (sp-1)<-B; sp <- sp - 2
306	c6	ADI D8	2	Z, S, P, CY, AC	A <- A + byte
307	c7	RST 0	1		CALL $0
310	c8	RZ	1		if Z, RET
311	c9	RET	1		PC.lo <- (sp); PC.hi<-(sp+1); SP <- SP+2
312	ca	JZ adr	3		if Z, PC <- adr
313	cb	-			
314	cc	CZ adr	3		if Z, CALL adr
315	cd	CALL adr	3		(SP-1)<-PC.hi;(SP-2)<-PC.lo;SP<-SP-2;PC=adr
316	ce	ACI D8	2	Z, S, P, CY, AC	A <- A + data + CY
317	cf	RST 1	1		CALL $8
320	d0	RNC	1		if NCY, RET
321	d1	POP D	1		E <- (sp); D <- (sp+1); sp <- sp+2
322	d2	JNC adr	3		if NCY, PC<-adr
323	d3	OUT D8	2		special
324	d4	CNC adr	3		if NCY, CALL adr
325	d5	PUSH D	1		(sp-2)<-E; (sp-1)<-D; sp <- sp - 2

326	d6	SUI D8	2	Z, S, P, CY, AC	A <- A - data
327	d7	RST 2	1		CALL $10
330	d8	RC	1		if CY, RET
331	d9	-			
332	da	JC adr	3		if CY, PC<-adr
333	db	IN D8	2		special
334	dc	CC adr	3		if CY, CALL adr
335	dd	-			
336	de	SBI D8	2	Z, S, P, CY, AC	A <- A - data - CY
337	df	RST 3	1		CALL $18
340	e0	RPO	1		if PO, RET
341	e1	POP H	1		L <- (sp); H <- (sp+1); sp <- sp+2
342	e2	JPO adr	3		if PO, PC <- adr
343	e3	XTHL	1		L <-> (SP); H <-> (SP+1)
344	e4	CPO adr	3		if PO, CALL adr
345	e5	PUSH H	1		(sp-2)<-L; (sp-1)<-H; sp <- sp - 2
346	e6	ANI D8	2	Z, S, P, CY, AC	A <- A & data
347	e7	RST 4	1		CALL $20
350	e8	RPE	1		if PE, RET
351	e9	PCHL	1		PC.hi <- H; PC.lo <- L
352	ea	JPE adr	3		if PE, PC <- adr
353	eb	XCHG	1		H <-> D; L <-> E
354	ec	CPE adr	3		if PE, CALL adr
355	ed	-			
356	ee	XRI D8	2	Z, S, P, CY, AC	A <- A ^ data
357	ef	RST 5	1		CALL $28
360	f0	RP	1		if P, RET
361	f1	POP PSW	1		flags <- (sp); A <- (sp+1); sp <- sp+2
362	f2	JP adr	3		if P=1 PC <- adr
363	f3	DI	1		special
364	f4	CP adr	3		if P, PC <- adr
365	f5	PUSH PSW	1		(sp-2)<-flags; (sp-1)<-A; sp <- sp - 2
366	f6	ORI D8	2	Z, S, P, CY, AC	A <- A | data
367	f7	RST 6	1		CALL $30
370	f8	RM	1		if M, RET
371	f9	SPHL	1		SP=HL

372	fa	JM adr	3		if M, PC <- adr
373	fb	EI	1		special
374	fc	CM adr	3		if M, CALL adr
375	fd	-			
376	fe	CPI D8	2	Z, S, P, CY, AC	A - data
377	ff	RST 7	1		CALL $38