Intel 8085 instruction set

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	NOP 1 4 -------	LXI B,d16 3 10 -------	STAX B 1 7 -------	INX B 1 6 --K----	INR B 1 4 SZKAPV-	DCR B 1 4 SZKAPV-	MVI B,d8 2 7 -------	RLC 1 4 -----VC	*DSUB 1 10 SZKAPVC	DAD B 1 10 -----VC	LDAX B 1 7 -------	DCX B 1 6 --K----	INR C 1 4 SZKAPV-	DCR C 1 4 SZKAPV-	MVI C,d8 2 7 -------	RRC 1 4 -----0C
1x	*ARHL 1 7 -----0C	LXI D,d16 3 10 -------	STAX D 1 7 -------	INX D 1 6 --K----	INR D 1 4 SZKAPV-	DCR D 1 4 SZKAPV-	MVI D,d8 2 7 -------	RAL 1 4 -----VC	*RDEL 1 10 -----VC	DAD D 1 10 -----VC	LDAX D 1 7 -------	DCX D 1 6 --K----	INR E 1 4 SZKAPV-	DCR E 1 4 SZKAPV-	MVI E,d8 2 7 -------	RAR 1 4 -----0C
2x	RIM 1 4 -------	LXI H,d16 3 10 -------	SHLD a16 3 16 -------	INX H 1 6 --K----	INR H 1 4 SZKAPV-	DCR H 1 4 SZKAPV-	MVI H,d8 2 7 -------	DAA 1 4 SZKAPVC	*LDHI d8 2 10 -------	DAD H 1 10 -----VC	LHLD a16 3 16 -------	DCX H 1 6 --K----	INR L 1 4 SZKAPV-	DCR L 1 4 SZKAPV-	MVI L,d8 2 7 -------	CMA 1 4 -------
3x	SIM 1 4 -------	LXI SP,d16 3 10 -------	STA a16 3 13 -------	INX SP 1 6 --K----	INR M 1 10 SZKAPV-	DCR M 1 10 SZKAPV-	MVI M,d8 2 10 -------	STC 1 4 ------1	*LDSI d8 2 10 -------	DAD SP 1 10 -----VC	LDA a16 3 13 -------	DCX SP 1 6 --K----	INR A 1 4 SZKAPV-	DCR A 1 4 SZKAPV-	MVI A,d8 2 7 -------	CMC 1 4 ------C
4x	MOV B,B 1 4 -------	MOV B,C 1 4 -------	MOV B,D 1 4 -------	MOV B,E 1 4 -------	MOV B,H 1 4 -------	MOV B,L 1 4 -------	MOV B,M 1 7 -------	MOV B,A 1 4 -------	MOV C,B 1 4 -------	MOV C,C 1 4 -------	MOV C,D 1 4 -------	MOV C,E 1 4 -------	MOV C,H 1 4 -------	MOV C,L 1 4 -------	MOV C,M 1 7 -------	MOV C,A 1 4 -------
5x	MOV D,B 1 4 -------	MOV D,C 1 4 -------	MOV D,D 1 4 -------	MOV D,E 1 4 -------	MOV D,H 1 4 -------	MOV D,L 1 4 -------	MOV D,M 1 7 -------	MOV D,A 1 4 -------	MOV E,B 1 4 -------	MOV E,C 1 4 -------	MOV E,D 1 4 -------	MOV E,E 1 4 -------	MOV E,H 1 4 -------	MOV E,L 1 4 -------	MOV E,M 1 7 -------	MOV E,A 1 4 -------
6x	MOV H,B 1 4 -------	MOV H,C 1 4 -------	MOV H,D 1 4 -------	MOV H,E 1 4 -------	MOV H,H 1 4 -------	MOV H,L 1 4 -------	MOV H,M 1 7 -------	MOV H,A 1 4 -------	MOV L,B 1 4 -------	MOV L,C 1 4 -------	MOV L,D 1 4 -------	MOV L,E 1 4 -------	MOV L,H 1 4 -------	MOV L,L 1 4 -------	MOV L,M 1 7 -------	MOV L,A 1 4 -------
7x	MOV M,B 1 7 -------	MOV M,C 1 7 -------	MOV M,D 1 7 -------	MOV M,E 1 7 -------	MOV M,H 1 7 -------	MOV M,L 1 7 -------	HLT 1 5 -------	MOV M,A 1 7 -------	MOV A,B 1 4 -------	MOV A,C 1 4 -------	MOV A,D 1 4 -------	MOV A,E 1 4 -------	MOV A,H 1 4 -------	MOV A,L 1 4 -------	MOV A,M 1 7 -------	MOV A,A 1 4 -------
8x	ADD B 1 4 SZKAPVC	ADD C 1 4 SZKAPVC	ADD D 1 4 SZKAPVC	ADD E 1 4 SZKAPVC	ADD H 1 4 SZKAPVC	ADD L 1 4 SZKAPVC	ADD M 1 7 SZKAPVC	ADD A 1 4 SZKAPVC	ADC B 1 4 SZKAPVC	ADC C 1 4 SZKAPVC	ADC D 1 4 SZKAPVC	ADC E 1 4 SZKAPVC	ADC H 1 4 SZKAPVC	ADC L 1 4 SZKAPVC	ADC M 1 7 SZKAPVC	ADC A 1 4 SZKAPVC
9x	SUB B 1 4 SZKAPVC	SUB C 1 4 SZKAPVC	SUB D 1 4 SZKAPVC	SUB E 1 4 SZKAPVC	SUB H 1 4 SZKAPVC	SUB L 1 4 SZKAPVC	SUB M 1 7 SZKAPVC	SUB A 1 4 SZKAPVC	SBB B 1 4 SZKAPVC	SBB C 1 4 SZKAPVC	SBB D 1 4 SZKAPVC	SBB E 1 4 SZKAPVC	SBB H 1 4 SZKAPVC	SBB L 1 4 SZKAPVC	SBB M 1 7 SZKAPVC	SBB A 1 4 SZKAPVC
Ax	ANA B 1 4 SZKAPVC	ANA C 1 4 SZKAPVC	ANA D 1 4 SZKAPVC	ANA E 1 4 SZKAPVC	ANA H 1 4 SZKAPVC	ANA L 1 4 SZKAPVC	ANA M 1 7 SZKAPVC	ANA A 1 4 SZKAPVC	XRA B 1 4 SZKAPVC	XRA C 1 4 SZKAPVC	XRA D 1 4 SZKAPVC	XRA E 1 4 SZKAPVC	XRA H 1 4 SZKAPVC	XRA L 1 4 SZKAPVC	XRA M 1 7 SZKAPVC	XRA A 1 4 SZKAPVC
Bx	ORA B 1 4 SZKAPVC	ORA C 1 4 SZKAPVC	ORA D 1 4 SZKAPVC	ORA E 1 4 SZKAPVC	ORA H 1 4 SZKAPVC	ORA L 1 4 SZKAPVC	ORA M 1 7 SZKAPVC	ORA A 1 4 SZKAPVC	CMP B 1 4 SZKAPVC	CMP C 1 4 SZKAPVC	CMP D 1 4 SZKAPVC	CMP E 1 4 SZKAPVC	CMP H 1 4 SZKAPVC	CMP L 1 4 SZKAPVC	CMP M 1 7 SZKAPVC	CMP A 1 4 SZKAPVC
Cx	RNZ 1 12/6 -------	POP B 1 10 -------	JNZ a16 3 10/7 -------	JMP a16 3 10 -------	CNZ a16 3 18/9 -------	PUSH B 1 12 -------	ADI d8 2 7 SZKAPVC	RST 0 1 12 -------	RZ 1 12/6 -------	RET 1 10 -------	JZ a16 3 10/7 -------	*RSTV 1 12/6 -------	CZ a16 3 18/9 -------	CALL a16 3 18 -------	ACI d8 2 7 SZKAPVC	RST 1 1 12 -------
Dx	RNC 1 12/6 -------	POP D 1 10 -------	JNC a16 3 10/7 -------	OUT d8 2 10 -------	CNC a16 3 18/9 -------	PUSH D 1 12 -------	SUI d8 2 7 SZKAPVC	RST 2 1 12 -------	RC 1 12/6 -------	*SHLX 1 10 -------	JC a16 3 10/7 -------	IN d8 2 10 -------	CC a16 3 18/9 -------	*JNK a16 3 10/7 -------	SBI d8 2 7 SZKAPVC	RST 3 1 12 -------
Ex	RPO 1 12/6 -------	POP H 1 10 -------	JPO a16 3 10/7 -------	XTHL 1 16 -------	CPO a16 3 18/9 -------	PUSH H 1 12 -------	ANI d8 2 7 SZKAPVC	RST 4 1 12 -------	RPE 1 12/6 -------	PCHL 1 6 -------	JPE a16 3 10/7 -------	XCHG 1 4 -------	CPE a16 3 18/9 -------	*LHLX 1 10 -------	XRI d8 2 7 SZKAPVC	RST 5 1 12 -------
Fx	RP 1 12/6 -------	POP PSW 1 10 SZKAPVC	JP a16 3 10/7 -------	DI 1 4 ------	CP a16 3 18/9 -------	PUSH PSW 1 12 -------	ORI d8 2 7 SZKAPVC	RST 6 1 12 -------	RM 1 12/6 -------	SPHL 1 6 -------	JM a16 3 10/7 -------	EI 1 4 -------	CM a16 3 18/9 -------	*JK a16 3 10/7 -------	CPI d8 2 7 SZKAPVC	RST 7 1 12 -------

Misc/control instructions
Jumps/calls
8bit load/store/move instructions
16bit load/store/move instructions
8bit arithmetic/logical instructions
16bit arithmetic/logical instructions

Length in bytes →

INS reg	← Instruction mnemonic
2 7	← Duration in cycles
SZKAPVC	← Flags affected

Duration of conditional calls and returns is different when action is taken or not. This is indicated by two numbers separated by "/". The higher number (on the left side of "/") means duration of instruction when action is taken, the lower number (on the right side of "/") means duration of instruction when action is not taken.
All instructions marked by "*" are not documented.

Flags affected are always shown in **S Z K A P V C** order. If flag is marked by "**0**" it means it is reset after the instruction. If it is marked by "**1**" it is set. If it is marked by "**-**" it is not changed. If it is marked by "**S**", "**Z**", "**K**", "**A**", "**P**", "**V**" or "**C**" corresponding flag is affected by particular instruction.

d8 means immediate 8 bit data, in LDHI and LDSI instructions the argument is treated as unsigned
d16 means immediate 16 bit data
a16 means 16 bit address

JNK has alternative mnemonic **JNX5** or **JNUI**
JK has alternative mnemonic **JX5** or **JUI**

Registers

15 . . . 8	7 . . . 0	
A (accumulator)	F (flags)	← PSW
B	C	← B
D	E	← D
H	L	← H

15 . . . 0
SP (stack pointer)
PC (program counter)

Flag register (F) bits:

7	6	5	4	3	2	1	0
S	Z	K	A	0	P	V	C

S - Sign Flag
Z - Zero Flag
K - also called X5 or UI (underflow or overflow indicator), undocumented
A - also called AC, Auxiliary Carry Flag
0 - Not used, always zero
P - Parity Flag
V - Overflow Flag, undocumented
C - Carry Flag