Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00000000	ALL	NOP	4	(No Operation) PC <= PC + 1	NONE
00000001	ALL	LXIB	10	(Load Immediate BC) C <= @(PC + 1) B <= @(PC + 2) PC <= PC + 3	NONE
00000010	ALL	STAXB	7	(Store A Indirect BC) @(BC) <= A PC <= PC + 1	NONE
00000011	ALL	INXB	6	(Increment BC) BC <= BC + 1 PC <= PC + 1	UF, VF
00000100	ALL	INRB	4	(Increment B) B <= B + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00000101	ALL	DCRB	4	(Decrement B) B <= B- 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00000110	ALL	MVIB	7	(Move Immediate B) B <= @(PC + 1) PC <= PC + 2	NONE
00000111	ALL	RLC	4	(Rotate A Left Circular) CF <= A /128) A <= (A * 2) + (A / 128) PC <= PC + 1	CF
00001000	8085A	?	?	Undocumented	?
	8085B 8085C S8085D	DSUB	10	(Subtract BC from HL) HL <= HL - BC PC <= PC + 1	SF, ZF, UF, AF, PF, CF
00001001	ALL	DADB	10	(Add BCto HL) HL <= HL + BC PC <= PC + 1	UF, VF, CF
00001010	ALL	LDAXB	7	(Load A Indirect BC) A <= @(BC) PC <= PC + 1	NONE
00001011	ALL	DCXB	6	(Decrement BC) BC <= BC - 1 PC <= PC + 1	UF, VF
00001100	ALL	INRC	4	(Increment C) C <= C + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00001101	ALL	DCRC	4	(Decrement C) C <= C- 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00001110	ALL	MVIC	7	(Move Immediate C) C <= @(PC + 1) PC <= PC + 2	NONE
00001111	ALL	RRC	4	(Rotate Right Circular) CY <= A % 2) A <= (A / 2) + ((A % 2)* 128) PC <= PC + 1	CY
00010000	8085A	?	?	<Undocumented>	?
	8085B 8085C S8085D	ARHL	7	(Arithmetic Shift Right HL) CF <= HL % 2 HL <= (HL / 2) + ((H / 128) * 128) PC < = PC + 1	CF
00010001	ALL	LXID	10	(Load Immediate DE) E <= @(PC + 1) D <= @(PC + 2) PC <= PC + 3	NONE
00010010	ALL	STAXD	7	(Store A Indirect DE) @(DE) <= A PC <= PC + 1	NONE
00010011	ALL	INXD	6	(Increment DE) DE <= DE + 1 PC <= PC + 1	UF, VF
00010100	ALL	INRD	4	(Increment D) D <= D + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00010101	ALL	DCRD	4	(Decrement D) D <= D- 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00010110	ALL	MVID	7	(Move Immediate D) D <= @(PC + 1) PC <= PC + 2	NONE
00010111	ALL	RAL	4	(Rotate Left through Carry) CF <= A /128 A <= (A * 2) + CF PC <= PC + 1	CF
00011000		?	?	Undocumented	?
	8085B 8085C S8085D	RDEL	10	(Rotate DE Left Through Carry) CF = D / 128 DE <= (DE * 2) + CF PC <= PC + 1	CF, VF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00011001	ALL	DADD	10	(Add DE to HL) HL <= HL + DE PC <= PC + 1	UF, VF, CF
00011010	ALL	LDAXD	7	(Load A Indirect DE) A <= @(DE) PC <= PC + 1	NONE
00011011	ALL	DCXD	6	(Decrement DE) DE <= DE - 1 PC <= PC + 1	UF, VF
00011100	ALL	INRE	4	(Increment E) E <= E + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00011101	ALL	DCRE	4	(Decrement E) E <= E- 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00011110	ALL	MVIE	7	(Move Immediate E) E <= @(PC + 1) PC <= PC + 2	NONE
00011111	ALL	RAR	4	(Rotate Right through Carry) CF <= A % 2 A <= (A/ 2) + (CF * 128) PC <= PC + 1	CF
00100000	ALL	RIM<	4	Read Interrupt Mask A <= Interrupt mask (On first read previous state of INTDS is returned) OC <= PC + 1	NONE
00100001	ALL	LXIH	10	(Load Immediate HL) L <= @(PC + 1) H <= @(PC + 2) PC <= PC + 3)	NONE
00100010	ALL	SHLD	16	(Store HL Direct) address <= @(PC + 1) + (@(PC + 2) * 256) @(address) <= L @(address + 1) <= H PC <= PC + 3	NONE
00100011	ALL	INXH	6	(Increment HL) HL <= HL + 1 PC <= PC + 1	UF, VF
00100100	ALL	INRH	4	(Increment H) H <= H + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00100101	ALL	DCRH	4	(Decrement H) H <= H - 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00100110	ALL	MVIH	7	(Move Immediate H) H <= @(PC + 1) PC <= PC + 2	NONE
00100111	8085A 8085B	DAA	4	(Decimal Adjust for Addition) A <= BCD(A) PC <= PC + 1	ZF, AF,CF
	8085C 8085D			(Decimal Adjust for Addition and Subtraction) A <= BCD(A) PC <= PC + 1	ZF, AF, CF, SF, NF
00101000	8085A	?	?	Undocumented	?
	8085B 8085C S8085D	LDHI	10	(Load DE with HL + immediate) DE <= HL + @(PC + 1) PC <= PC + 2	None
00101001	ALL	DADH	10	(Add HL to HL) HL <= HL + HL PC <= PC + 1	UF, VF, CF
00101010	ALL	LHLD	16	(Load HL Direct) address <= @(PC + 1) + (@(PC + 2) * 256) L <= @(address) H <= @(address + 1) PC <= PC + 3	NONE
00101011	ALL	DCXH	6	(Decrement HL) HL <=HL - 1 PC <= PC + 1	UF, VF
00101100	ALL	INRL	4	(Increment L) L <= L + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00101101	ALL	DCRL	4	(Decrement L) L <= L- 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00101110	ALL	MVIL	7	(Move Immediate L) L <= @(PC + 1) PC <= PC + 2	NONE
00101111	ALL	CMA	4	(Complement Accumulator) A <= ~A PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00110000	ALL	SIM	4	(Set Interrupt Mask) interrupt mask <= A PC <= PC + 1	NONE
00110001	ALL	LXISP	10	(Load Immediate SP) SP <= ((@(PC + 2) * 256) + @(PC + 1)) PC <= PC + 3	NONE
00110010	ALL	STA	13	(Store A Direct) address <= @(PC + 1) + (@(PC + 2) * 256) @(address) <= A PC <= PC + 3	NONE
00110011	ALL	INXSP	6	(Increment SP) SP <= SP +1 PC <= PC + 1	UF, VF
00110100	ALL	INRM	10	(Increment Memory) @HL <= @HL + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00110101	ALL	DCRM	10	(Decrement Memory) @HL <= @HL - 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00110110	ALL	MVIM	104	(Move Immediate Memory) @HL <= @(PC + 1) PC <= PC + 2	NONE
00110111	ALL	STC		(Set Carry) CF <= 1	CF
00111000	8085A	?	?	(Undocumented)	?
	8085B 8085C S8085D	LDSI	10	(Load DE with SP + immediate) DE <= SP + @(PC + 1) PC <= PC + 2	None
00111001	ALL	DADSP	10	(Add SP to HL) HL <=SP + HL PC <= PC + 1	UF, VF, CF
00111010	ALL	LDA	13	(Load A Direct) address <= @(PC + 1) + (@(PC + 2) * 256) A <= @(address) PC <= PC + 3	NONE
00111011	ALL	DCXSP	6	(Decrement SP) SP <= SP - 1 PC <= PC + 1	UF, VF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
00111100	ALL	INRA	4	(Increment A) A <= A + 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF
00111101	ALL	DCRA	4	(Decrement A) A <= A - 1 PC <= PC + 1	ZF,AF,CF, PF, VF, UF, NF
00111110	ALL	MVIA	7	(Move Immediate A) A <= @(PC + 1) PC <= PC + 2	NONE
00111111	ALL	CMC	4	(Complement Carry) CF <= ~CF	CF
01000000	ALL	MOVBB	4	(Move B to B (NOP)) PC <= PC + 1	NONE
01000001	ALL	MOVBC	4	(Move C to B) B <= C PC <= PC + 1	NONE
01000010	ALL	MOVBD	4	(Move D to B) B <= D PC <= PC + 1	NONE
01000011	ALL	MOVBE	4	(Move E to B) B <= E PC <= PC + 1	NONE
01000100	ALL	MOVBH	4	(Move H to B) B <= H PC <= PC + 1	NONE
01000101	ALL	MOVBL	4	(Move L to B) B <= L PC <= PC + 1	NONE
01000110	ALL	MOVBM	7	(Move Memory to B) B <= @(HL) PC <= PC + 1	NONE
01000111	ALL	MOVBA	4	(Move A to B) B <= A PC <= PC + 1	NONE
01001000	ALL	MOVCB	4	(Move B to C) C <= B PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
01001001	8085A 8085B 8085C	MOVCC	4	(Move C to C (NOP)) PC <= PC + 1	NONE
	S8085D	HLSP	4	(Move SP to HL) HL <= SP PC <= PC + 1	NONE
01001010	ALL	MOVCD	4	(Move D to C) C <= D PC <= PC + 1	NONE
01001011	ALL	MOVCE	4	(Move E to C) C <= E PC <= PC + 1	NONE
01001100	ALL	MOVCH	4	(Move H to C) C <= H PC <= PC + 1	NONE
01001101	ALL	MOVCL	4	(Move L to C) C <= L PC <= PC + 1	NONE
01001110	ALL	MOVCM	7	(Move Memory to C) C <= @(HL) PC <= PC + 1	NONE
01001111	ALL	MOVCA	4	(Move A to C) C <= A PC <= PC + 1	NONE
01010000	ALL	MOVDB	4	(Move B to D) D <= B PC <= PC + 1	NONE
01001001	ALL	MOVDC	4	(Move C to D) D <= C PC <= PC + 1	NONE
01010010	8085A 8085B 8085C	MOVDD	4	(Move D to D (NOP)) PC <= PC + 1	NONE
	8085D	MUL	10	(Multiply B by C) BC <= B * C	ZF
01010011	ALL	MOVDE	4	(Move E to D) D <= E PC <= PC + 1	NONE
01010100	ALL	MOVDH	4	(Move H to D) D <= H PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
01010101	ALL	MOVDL	4	(Move L to D) D <= L PC <= PC + 1	NONE
01010110	ALL	MOVDM	7	(Move Memory to D) D <= @(HL) PC <= PC + 1	NONE
01010111	ALL	MOVDA	4	(Move A to D) D <= A PC <= PC + 1	NONE
01011000	ALL	MOVEB	4	(Move B to E) E <= B PC <= PC + 1	NONE
01011001	ALL	MOVEC	4	(Move C to E) E <= C PC <= PC + 1	NONE
01011010	ALL	MOVED	4	(Move D to E) E <= D PC <= PC + 1	NONE
01011011	8085A 8085B 8085C	MOVEE	4	(Move E to E (NOP) PC <= PC + 1	NONE
	S8085D	DIV	10	(Divide BC by A) BC <= BC / A A <= BC % A PC <= PC + 1	OF, ZF
01011100	ALL	MOVEH	4	(Move H to E) E <= H PC <= PC + 1	NONE
01011101	ALL	MOVEL	4	(Move L to E) E <= L PC <= PC + 1	NONE
01011110	ALL	MOVEM	7	(Move Memory to E) E <= @(HL) PC <= PC + 1	NONE
01011111	ALL	MOVEA	4	(Move A to E) E <= A PC <= PC + 1	NONE
01100000	ALL	MOVHB	4	(Move B to H) H <= B PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
01100001	ALL	MOVHC	4	(Move C to H) H <= C PC <= PC + 1	NONE
01100010	ALL	MOVHD	4	(Move D to H) H <= D PC <= PC + 1	NONE
01100011	ALL	MOVHE	4	(Move E to H) H <= E PC <= PC + 1	NONE
01100100	8085A 8085B	MOVHH	4	(Move H to H) PC <= PC + 1	NONE
	8085C S8085D	STAR	10	(Store A Relative HL + Immediate) address <= HL + @(PC + 1) @address = A PC <= PC + 2	NONE
01100101	ALL	MOVHL	4	(Move L to H) H <= L PC <= PC + 1	NONE
01100110	ALL	MOVHM	7	(Move Memory to H) H <= @(HL) PC <= PC + 1	NONE
01100111	ALL	MOVHA	4	(Move A to H) H <= A PC <= PC + 1	NONE
01101000	ALL	MOVLB	4	(Move B to L) L <= B PC <= PC + 1	NONE
01101001	ALL	MOVLC	4	(Move C to L) L <= C PC <= PC + 1	NONE
01101010	ALL	MOVLD	4	(Move D to L) L <= D PC <= PC + 1	NONE
01101011	ALL	MOVLE	4	(Move E to L) L <= E PC <= PC + 1	NONE
01101100	ALL	MOVLH	4	(Move H to L) L <= H PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
01101101	8085A 8085B	MOVLL	4	(Move L to L (NOP)) PC <= PC + 1	NONE
	8085C S8085D	LDAR	10	(Load A Relative HL + Immediate) address = HL + @(PC + 1) A <= @address PC <= PC + 2	NONE
01101110	ALL	MOVLM	7	(Move Memory to L) L <= @(HL) PC <= PC + 1	NONE
01101111	ALL	MOVLA	4	(Move A to L) L <= A PC <= PC + 1	NONE
01110000	ALL	MOVMB	7	(Move B to Memory) @(HL) <= B PC <= PC + 1	NONE
01110001	ALL	MOVMC	7	(Move C to Memory) @(HL) <= C PC <= PC + 1	NONE
01110010	ALL	MOVMD	7	(Move D to Memory) @(HL) <= D PC <= PC + 1	NONE
01110011	ALL	MOVME	7	(Move E to Memory) @(HL) <= E PC <= PC + 1	NONE
01110100	ALL	MOVMH	7	(Move H to Memory) @(HL) <= H PC <= PC + 1	NONE
01110101	ALL	MOVML	7	(Move L to Memory) @(HL) <= L PC <= PC + 1	NONE
01110110	ALL	HALT	4	(Stop Processor)	NONE
01110111	ALL	MOVMA	7	(Move A to Memory) @(HL) <= A PC <= PC + 1	NONE
01111000	ALL	MOVAB	4	(Move B to A) A <= B PC <= PC + 1	NONE
01111001	ALL	MOVAC	4	(Move C to A) A <= C PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
01111010	ALL	MOVAD	4	(Move D to A) A <= D PC <= PC + 1	NONE
01111011	ALL	MOVAE	4	(Move E to A) A <= E PC <= PC + 1	NONE
01111100	ALL	MOVAH	4	(Move H to A) A <= H PC <= PC + 1	NONE
01111101	ALL	MOVAL	4	(Move L to A) A <= L PC <= PC + 1	NONE
01111110	ALL	MOVAM	7	(Move Memory to A) A <= @(HL) PC <= PC + 1	NONE
01111111	ALL	MOVAA	4	(Move A to A (NOP)) PC <= PC + 1	NONE
10000000	ALL	ADDB	4	(Add B to A) A <= A + B PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000001	ALL	ADDC	4	(Add C to A) A <= A + C PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000010	ALL	ADDD	4	(Add D to A) A <= A + D PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000011	ALL	ADDE	4	(Add E to A) A <= A + E PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000100	ALL	ADDH	4	(Add H to A) A <= A + H PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000101	ALL	ADDL	4	(Add L to A) A <= A + L PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000110	ALL	ADDM	7	(Add Memory to A) A <= A + @(HL) PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10000111	ALL	ADDA	4	(Add A to A) A <= A + A PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
10001000	ALL	ADCB	4	(Add B to A With Carry) A <= A + B + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001001	ALL	ADCC	4	(Add C to A With Carry) A <= A + C + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001010	ALL	ADCD	4	(Add D to A With Carry) A <= A +D + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001011	ALL	ADCE	4	(Add E to A With Carry) A <= A + E + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001100	ALL	ADCH	4	(Add H to A With Carry) A <= A + H + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001101	ALL	ADCL	4	(Add L to A With Carry) A <= A + L + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001110	ALL	ADCM	7	(Add Memory to A With Carry) A <= A + @(HL) + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10001111	ALL	ADCA	4	(Add A to A With Carry) A <= A + A + CF PC <= PC + 1	SF,ZF,UF, AF,PF,VF, CF
10010000	ALL	SUBB	4	(Subtract B from A)) A <= A - B PC <= PC + 1	ALL
10010001	ALL	SUBC	4	(Subtract C from A)) A <= A - C PC <= PC + 1	ALL
10010010	ALL	SUBD	4	(Subtract D from A)) A <= A - D PC <= PC + 1	ALL
10010011	ALL	SUBE	4	(Subtract E from A)) A <= A - E PC <= PC + 1	ALL
10010100	ALL	SUBH	4	(Subtract H from A)) A <= A - H PC <= PC + 1	ALL

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
10010101	ALL	SUBL	4	(Subtract L from A)) A <= A - L PC <= PC + 1	ALL
10010110	ALL	SUBM	7	(Subtract Memory from A)) A <= A - @(HL) PC <= PC + 1	ALL
10010111	ALL	SUBA	4	(Subtract A from A)) A <= A - A PC <= PC + 1	ALL
10011000	ALL	SBBB	4	(Subtract B from A With Borrow)) A <= A - B - CF PC <= PC + 1	ALL
10011001	ALL	SBBC	4	(Subtract C from A With Borrow)) A <= A - C - CF PC <= PC + 1	ALL
10011010	ALL	SBBD	4	(Subtract D from A With Borrow)) A <= A - D - CF PC <= PC + 1	ALL
10011011	ALL	SBBE	4	(Subtract E from A With Borrow)) A <= A - E - CF PC <= PC + 1	ALL
10011100	ALL	SBBH	4	(Subtract H from A With Borrow)) A <= A - H - CF PC <= PC + 1	ALL
10011101	ALL	SBBL	4	(Subtract L from A With Borrow)) A <= A - L - CF PC <= PC + 1	ALL
10011110	ALL	SBBM	7	(Subtract Memory from A With Borrow)) A <= A - @(HL) - CF PC <= PC + 1	ALL
10011100	ALL	SBBA	4	(Subtract A from A With Borrow)) A <= A - A - CF PC <= PC + 1	ALL

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
10100000	ALL	ANAB	4	(Logical AND B with A)) A <= A & B PC <= PC + 1	SF, ZF,AF, PF, CF
10100001	ALL	ANAC	4	(Logical AND C with A)) A <= A & C PC <= PC + 1	SF, ZF,AF, PF, CF
10100010	ALL	ANAD	4	(Logical AND D with A)) A <= A & D PC <= PC + 1	SF, ZF,AF, PF, CF
10100011	ALL	ANAE	4	(Logical AND E with A)) A <= A & E PC <= PC + 1	SF, ZF,AF, PF, CF
10100100	ALL	ANAH	4	(Logical AND H with A)) A <= A & H PC <= PC + 1	SF, ZF,AF, PF, CF
10100101	ALL	ANAL	4	(Logical AND L with A)) A <= A & L PC <= PC + 1	SF, ZF,AF, PF, CF
10100110	ALL	ANAM	7	(Logical AND Memory with A)) A <= A & @(HL) PC <= PC + 1	SF, ZF,AF, PF, CF
10100111	ALL	ANAA	4	(Logical AND A with A)) A <= A & A PC <= PC + 1	SF, ZF,AF, PF, CF
10101000	ALL	XRAB	4	(Logical Exclusive OR B with A)) A <= A ^ B PC <= PC + 1	SF, ZF,AF, PF, CF
10101001	ALL	XRAC	4	(Logical Exclusive OR C with A)) A <= A ^ C PC <= PC + 1	SF, ZF,AF, PF, CF
10101010	ALL	XRAD	4	(Logical Exclusive OR D with A)) A <= A ^ D PC <= PC + 1	SF, ZF,AF, PF, CF
10101011	ALL	XRAE	4	(Logical Exclusive OR E with A)) A <= A ^ E PC <= PC + 1	SF, ZF,AF, PF, CF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
10101100	ALL	XRAH	4	(Logical Exclusive OR H with A)) A <= A ^ H PC <= PC + 1	SF, ZF,AF, PF, CF
10101101	ALL	XRAL	4	(Logical Exclusive OR L with A)) A <= A ^ L PC <= PC + 1	SF, ZF,AF, PF, CF
10101110	ALL	XRAM	7	(Logical Exclusive OR Memory with A) A <= A ^ @(HL) PC <= PC + 1	SF, ZF,AF, PF, CF
10101111	ALL	XRAA	4	(Logical Exclusive OR A with A)) A <= A ^ A PC <= PC + 1	SF, ZF,AF, PF, CF
10110000	ALL	ORAB	4	(Logical OR B with A)) A <= A | B PC <= PC + 1	SF, ZF,AF, PF, CF
10110001	ALL	ORAC	4	(Logical OR C with A)) A <= A | C PC <= PC + 1	SF, ZF,AF, PF, CF
10110010	ALL	ORAD	4	(Logical OR D with A)) A <= A | D PC <= PC + 1	SF, ZF,AF, PF, CF
10110011	ALL	ORAE	4	(Logical OR E with A)) A <= A | E PC <= PC + 1	SF, ZF,AF, PF, CF
10110100	ALL	ORAH	4	(Logical OR H with A)) A <= A | H PC <= PC + 1	SF, ZF,AF, PF, CF
10110101	ALL	ORAL	4	(Logical OR L with A)) A <= A | L PC <= PC + 1	SF, ZF,AF, PF, CF
10110110	ALL	ORAM	7	(Logical OR Memory with A)) A <= A | @(HL) PC <= PC + 1	SF, ZF,AF, PF, CF
10110111	ALL	ORAA	4	(Logical OR A with A)) A <= A | A PC <= PC + 1	SF, ZF,AF, PF, CF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
10111000	ALL	CMPB	4	(Compare A with B (A - B)) PC <= PC + 1	ALL
10111001	ALL	CMPC	4	(Compare A with C (A - C)) PC <= PC + 1	ALL
10111010	ALL	CMPD	4	(Compare A with D (A - D)) PC <= PC + 1	ALL
10111011	ALL	CMPE	4	(Compare A with E (A - E)) PC <= PC + 1	ALL
10111100	ALL	CMPH	4	(Compare A with H (A - H)) PC <= PC + 1	ALL
10111101	ALL	CMPL	4	(Compare A with L (A - L)) PC <= PC + 1	ALL
10111110	ALL	CMPM	7	(Compare A with Memory (A - @(HL))) PC <= PC + 1	ALL
10111111	ALL	CMPA	4	(Compare A with A (A - A)) PC <= PC + 1	ALL

11000000	ALL	RNZ	6/12	(Return if Not Zero)
				if(ZF == 0) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11000001	ALL	POPB	10	(Pop BC from Top of Stack) C <= @SP) B <= @(SP + 1) SP <= SP + 2 PC <= PC + 1	NONE

11000010	ALL	JNZ	7/10	(Jump if Not Zero)
				if(ZF == 0) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11000011	ALL	JMP	10	(Jump to Absolute Location) PC <= @(PC) + (@(PC + 1) * 256)	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS

11000100	ALL	CNZ	9/18	(Call if not zero)
				if(ZF == 0) {
				@(SP - 1) <= (PC + 3) / 256
				@(SP - 2) <= (PC + 3) % 256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11000101	ALL	PUSHB	12	(Push BC to top of stack) @(SP - 1) <= B @(SP - 2) <= C SP <= SP - 2 PC <= PC + 1	NONE
11000110	ALL	ADI	7	(Add Immediate Data to A) A <= A + @(PC + 1) PC <= PC + 2	SF,ZF, UF,AF,PF, VF,CF
11000111	ALL	RST0	12	(Restart at Location 0) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0	NONE

11001000	ALL	RZ	6/12	(Return if Zero)
				if(ZF == 1) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11001001	ALL	RET	12	(Return unconditional) PC <= @(SP)+ \ @((SP + 1) * 256) SP <= SP + 2}	NONE

11001010	ALL	JZ	7/10	(Jump if Zero)
				if(ZF == 1) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
11001011	8085A	?	?	Undocumented	?

	8085B
	8085C
	S8085D
		RSTV	6/12	(Restart on Overflow)
				if(VF == 1) {
				@(SP - 1) <= (PC + 1)/ 256
				@(SP - 2) <= (PC + 1) % 256)
				SP <= SP - 2}
				PC <= 0x40
				}
				else
				PC <= PC + 1
					NONE

11001100	ALL	CZ	9/18	(Call if zero)
				if(ZF == 1
				)
				@(SP - 1) <= (PC + 3) / 256
				@(SP - 2) <= (PC + 3) % 256
				SP<= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11001101	ALL	CALL	18	(Call unconditional) @(SP - 1) <= (PC + 3)/ 256 @(SP - 2) <= (PC + 3)% 256S PC <= @(PC + 1} + \ @((PC + 2) * 256)	NONE
11001110	ALL	ACI	7	(Add Immediate Data to A with carry) A <= A + @(PC + 1) + CF PC <= PC + 2	SF,ZF, UF,AF,PF, VF,CF
11001111	ALL	RST8	12	(Restart at Location 8) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x08	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS

11010000	ALL	RNC	6/12	(Return if Not Carry)
				if(CF == 0) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11010001	ALL	POPD	10	(Pop DE from Top of Stack) E <= @SP) D <= @(SP + 1) SP <= SP + 2 PC <= PC + 1	NONE

11010010	ALL	JNC	7/10	(Jump if Not Carry)
				if(CF == 0) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11010011	ALL	OUT	7	(Output to I/O device) I/O @(PC + 1) <= A PC <= PC + 2	NONE

11010100	ALL	CNC	9/18	(Call if not carry)
				if(CF == 0) {
				@(SP - 1) <= (PC + 3) /
				256
				@(SP - 2) <= (PC + 3) %
				256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11010101	ALL	PUSHD	12	(Push DE to top of stack) @(SP - 1) <= D @(SP - 2) <= E SP <= SP - 2 PC <= PC + 1	NONE
11000110	ALL	SUI	7	(Subtract Immediate Data from A) A <= A - @(PC + 1) PC <= PC + 2	SF,ZF, UF,AF,PF, VF,CF, NF

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
11010111	ALL	RST10	12	(Restart at Location 10) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x10	NONE

11011000	ALL	RZ	6/12	(Return if Carry)
				if(CF == 1) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11011001	8085A	?	?	Undocumented	?
	8085B 8085C S8085D	SHLX	10	(Store HL indirect DE) @(DE) <= L @(DE + 1) <= H PC <= PC + 1	NONE

11011010	ALL	JC	7/10	(Jump if Carry)
				if(CF == 1) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11011011	ALL	IN	7	(Input from I/O device) A <= I/O @(PC + 1) PC <= PC + 2	NONE

11011100	ALL	CC	9/18	(Call if carry)
				if(CF == 1) {
				@(SP - 1) <= (PC + 3) /
				256
				@(SP - 2) <= (PC + 3) %
				256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
11011101	8085A	?	?	Undocumented	?

	8085B
	8085C
	S8085D
		JNUI	7/10	(Jump on not unsigned)
				if(UF == 0) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11011110	ALL	SBI	7	(Suntract Immediate Data from A with borrow) A <= (A - @(PC + 1)) - CF PC <= PC + 2	SF,ZF, UF,AF,PF, VF,CF, NF
11011111	ALL	RST18	12	(Restart at Location 18) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x18	NONE

11100000	ALL	RPO	6/12	(Return if Odd Parity)
				if(PF == 0) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11100001	ALL	POPH	10	(Pop HL from Top of Stack) L <= @SP) H <= @(SP + 1) SP <= SP + 2 PC <= PC + 1	NONE

11100010	ALL	JPO	7/10	(Jump if Parity Odd)
				if(PF == 0) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11100011	ALL	XTHL	18	(Exchange HL with top of stack) L <= @(SP) H <= @(SP + 1) @(SP + 1) <= H @(SP) <= L	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS

11100100	ALL	CPO	9/18	(Call if Parity Odd)
				if(PF == 0) {
				@(SP - 1) <= (PC + 3) / 256
				@(SP - 2) <= (PC + 3) % 256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11100101	ALL	PUSHH	12	(Push HL to top of stack) @(SP - 1) <= H @(SP - 2) <= L SP <= SP - 2 PC <= PC + 1	NONE
11100110	ALL	ANI	7	(Logical AND Immediate Data with A) A <= A & @(PC + 1) PC <= PC + 2	SF,ZF, AF,PF,CF
11100111	ALL	RST20	12	(Restart at Location 20) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x20	NONE

11101000	ALL	RPE	6/12	(Return if Even Parity)
				if(PF == 1) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11101001	ALL	PCHL	4	PC <= HL	NONE

11101010	ALL	JPE	7/10	(Jump if Parity Even)
				if(PF == 1) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11101011	ALL	XCHG	4	DE <= HL HL <= DE	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS

11101100	ALL	CPE	9/18	(Call if Parity Even)
				if(PF == 1) {
				@(SP - 1) <= (PC + 3) / 256
				@(SP - 2) <= (PC + 3) % 256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11101101	8085A	?	?	Undocumented	?
	8085B 8085C S8085D	LHLX	10	(Load HL indirect DE) L <= @(DE) H <= @(DE + 1) PC <= PC + 1	NONE
11101110	ALL	ANI	7	(Logical Exclusive OR Immediate Data with A) A <= A ^ @(PC + 1) PC <= PC + 2	SF,ZF, AF,PF,CF
11101111	ALL	RST28	12	(Restart at Location 28) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x28	NONE

11110000	ALL	RP	6/12	(Return if Positive)
				if(SF == 0) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

11110001	ALL	POPPSW	10	(Pop Accumulator and flags from Top of Stack) A <= @SP) Flags <= @(SP + 1) SP <= SP + 2 PC <= PC + 1	NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS

11110010	ALL	JP	7/10	(Jump if Positive)
				if(SF == 0) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11110011	ALL	DI	4	(Disable Interrupts) Interrupt Enable <= OFF PC <= PC + 1	NONE

11110100	ALL	CP	9/18	(Call if Positive)
				if(SF == 0) {
				@(SP - 1) <= (PC + 3) /
				256
				@(SP - 2) <= (PC + 3) %
				256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11110101	ALL	PUSHPSW	12	(Push Accumulator and flags to top of stack) @(SP - 1) <= Flags @(SP - 2) <= A SP <= SP - 2 PC <= PC + 1	NONE
11110110	ALL	ORI	7	(Logical OR Immediate Data with A) A <= A | @(PC + 1) PC <= PC + 2	SF,ZF, AF,PF,CF
11110111	ALL	RST30	12	(Restart at Location 30) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x30	NONE

11111000	ALL	RM	6/12	(Return if Minus)
				if(SF == 1) {
				PC <= @(SP)+ \
				@((SP + 1) * 256)
				SP <= SP + 2}
				}
				else
				PC <= PC + 1
					NONE

Table 1: Instruction Sets

OPCODE	MODEL	MNEMONIC	CYCLES	OP	FLAGS
11111001	ALL	SPHL	4	(HL to SP) SP <= HL PC <= PC + 1	NONE

11111010	ALL	JM	7/10	(Jump on Minus)
				if(SF == 1) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11111011	ALL	EI	4	(Enable Interrupts) Interrupt Enable <= ON PC <= PC + 1	NONE

11111100	ALL	CM	9/18	(Call on Minus)
				if(SF == 1) {
				@(SP - 1) <= (PC + 3) /
				256
				@(SP - 2) <= (PC + 3) %
				256
				SP <= SP - 2}
				PC <= @(PC + 1)+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11111101	8085A	?	?	Undocumented	?

	8085B
	8085C
	S8085D
		JUI	7/10	(Jump on unsigned)
				if(UF == 1) {
				PC <= @(PC) + 1+ \
				@((PC + 2) * 256)
				}
				else
				PC <= PC + 3
					NONE

11111110	ALL	CPI	7	(Compare Immediate Data with A) A - @(PC + 1) PC <= PC + 2	SF,ZF, UF,AF,PF, VF, CF
11111111	ALL	RST38	12	(Restart at Location 38) @(SP - 1) <= (PC + 1)/ 256 @(SP - 2) <= (PC + 1)% 256 SP <= SP - 2 PC <= 0x38	NONE