8 BIT LOAD GROUP (MOVE)

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
LD r, r' LD r, n	r ← r' r ← n	•	•	X	X	•	•	01 r r' 00 r 110 ← n →		1 2	1 2	4 7	r, r' Reg. 000 B 001 C 010 D 011 E 100 H 101 L 111 A
LD r, (HL) LD r, (IX+d)	r ← (HL) r ← (IX + d)	•	•	X	X	•	•	01 r 110 11 011 101 01 r 110 ← d →	DD	1 3	2 5	7 19	
LD r, (IY+d)	r ← (IY + d)	•	•	X	X	•	•	11 111 101 01 r 110 ← d →	FD	3	5	19	
LD (HL), r LD (IX+d), r	(HL) ← r (IX + d) ← r	•	•	X	X	•	•	01 110 r 11 011 101 01 110 r ← d →	DD	1 3	2 5	7 19	
LD (IY+d), r	(IY + d) ← r	•	•	X	X	•	•	11 111 101 01 110 r ← d →	FD	3	5	19	
LD (HL), n	(HL) ← n	•	•	X	X	•	•	00 110 110 ← n →	36	2	3	10	
LD (IX+d), n	(IX + d) ← n	•	•	X	X	•	•	11 011 101 00 110 110 ← d → ← n →	DD 36	4	5	19	
LD (IY+d), n	(IY + d) ← n	•	•	X	X	•	•	11 111 101 00 110 110 ← d → ← n →	FD 36	4	5	19	
LD A, (BC) LD A, (DE) LD A, (nn)	A ← (BC) A ← (DE) A ← (nn)	•	•	X	X	•	•	00 001 010 00 011 010 00 111 010 ← n → ← n →	0A 1A 3A	1 1 3	2 2 4	7 7 13	
LD (BC), A LD (DE), A LD (nn), A	(BC) ← A (DE) ← A (nn) ← A	•	•	X	X	•	•	00 000 010 00 010 010 00 110 010 ← n → ← n →	02 12 32	1 1 3	2 2 4	7 7 13	
LD A, I	A ← I	:	:	X	0	X	IFF	0 • 11 101 101 01 010 111	ED 57	2	2	9	
LD A, R	A ← R	:	:	X	0	X	IFF	0 • 11 101 101 01 011 111	ED 5F	2	2	9	
LD I, A	I ← A	•	•	X	•	X	•	• • 11 101 101 01 000 111	ED 47	2	2	9	
LD R, A	R ← A	•	•	X	•	X	•	• • 11 101 101 01 001 111	ED 4F	2	2	9	

Notes : r, r' means any of the registers A, B, C, D, E, H, L. IFF the content of the interrupt enable flip-flop, (IFF) is copied into the P/V flag.
For an explanation of flag notation and symbols for mnemonic tables, see Symbolic Notation section following tables.

8-BIT ARITHMETIC AND LOGICAL GROUP

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
ADD A, r ADD A, n	A ← A + r A ← A + n	:	:	X	X	V	0	:	10 000 r 11 000 110 ← n →		1 2	1 2	4 7	r Reg. 000 B 001 C 010 D 011 E 100 H 101 L 111 A
ADD A, (HL) ADD A, (IX+d)	A ← A + (HL) A ← A + (IX + d)	:	:	X	X	V	0	:	10 000 110 11 011 101 10 000 110 ← d →	DD	1 3	2 5	7 19	
ADD A, (IY+d)	A ← A + (IY + d)	:	:	X	X	V	0	:	11 111 101 10 000 110 ← d →	FD	3	5	19	
ADC A, s SUB s SBC A, s AND s OR s XOR s CP s INC r INC (HL) INC (IX + d)	A ← A + s + CY A ← A − s A ← A − s − CY A ← A ∧ s A ← A ∨ s A ← A ⊕ s A − s r ← r + 1 (HL) ← (HL) + 1 (IX + d) ← (IX + d) + 1	:	:	X	X	V	0	:	001 010 011 100 110 101 111 00 r 100 00 110 100 11 011 101 00 110 100 ← d →	DD	1 1 3	1 1 3 6	4 4 11 23	s is any of r, n, (HL), (IX+d), (IY+d) as shown for ADD instruction. The indicated bits replace the 000 in the ADD set above.
INC (IY+d)	(IY + d) ← (IY + d) + 1	:	:	X	:	X	V	0	11 111 101 00 110 100 ← d →	FD	3	6	23	
DEC m	m ← m − 1	:	:	X	:	X	V	1	101					m is any of r, (HL), (IX+d), (IY+d) as shown for INC. DEC same format and states as INC. Replace 100 with 101 in opcode.

16-BIT LOAD GROUP

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
LD dd, nn	dd ← nn	•	•	X	•	X	•	• • 00 dd0 001 ← n → ← n →		3	3	10	dd Pair 00 BC 01 DE 10 HL 11 SP
LD IX, nn	IX ← nn	•	•	X	•	X	•	• • 11 011 101 00 100 001 ← n → ← n →	DD 21	4	4	14	
LD IY, nn	IY ← nn	•	•	X	•	X	•	• • 11 111 101 00 100 001 ← n → ← n →	FD 21	4	4	14	
LD HL, (nn)	H ← (nn + 1) L ← (nn)	•	•	X	•	X	•	• • 00 101 010 ← n → ← n →	2A	3	5	16	
LD dd, (nn)	dd_H ← (nn + 1) dd_L ← (nn)	•	•	X	•	X	•	• • 11 101 101 01 dd1 011 ← n → ← n →	ED	4	6	20	
LD IX, (nn)	IX_H ← (nn + 1) IX_L ← (nn)	•	•	X	•	X	•	• • 11 011 101 01 101 010 ← n → ← n →	DD 2A	4	6	20	
LD IY, (nn)	IY_H ← (nn + 1) IY_L ← (nn)	•	•	X	•	X	•	• • 11 111 101 00 101 010 ← n → ← n →	FD 2A	4	6	20	
LD (nn), HL	(nn + 1) ← H (nn) ← L	•	•	X	•	X	•	• • 00 100 010 ← n → ← n →	22	3	5	16	
LD (nn), dd	(nn + 1) ← dd_H (nn) ← dd_L	•	•	X	•	X	•	• • 11 101 101 01 dd0 011 ← n → ← n →	ED	4	6	20	
LD (nn), IX	(nn + 1) ← IX_H (nn) ← IX_L	•	•	X	•	X	•	• • 11 011 101 00 100 010 ← n → ← n →	DD 22	4	6	20	
LD (nn), IY	(nn + 1) ← IY_H (nn) ← IY_L	•	•	X	•	X	•	• • 11 111 101 00 100 010 ← n → ← n →	FD 22	4	6	20	

Notes : dd is any of the register pairs BC, DE, HL, SP.
qq is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively.
e.g., BC_L = C, AF_H = A.

ROTATE AND SHIFT GROUP

Mnemonic	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	No.of Bytes	No.of Cycles	No.of T States	Comments
RLCA		•	•	X	0	X	•	0 : 00 000 111	07	1	1	4	Rotate left circular accumulator.
RLA		•	•	X	0	X	•	0 : 00 010 111	17	1	1	4	Rotate left accumulator.
RRCA		•	•	X	0	X	•	0 : 00 001 111	0F	1	1	4	Rotate right circular accumulator.
RRA		•	•	X	0	X	•	0 : 00 011 111	1F	1	1	4	Rotate right accumulator.
RLC r		:	:	X	0	X	P	0 : 11 001 011 00 000 r	CB	2	2	8	Rotate left circular register r.
RLC (HL)		:	:	X	0	X	P	0 : 11 001 011 00 000 110	CB	2	4	15	r Reg 000 B 001 C 010 D 011 E 100 H 101 L 111 A
RLC (IX + d)	r, (HL), (IX+d), (IY+d)	:	:	X	0	X	P	0 : 11 011 101 11 001 011 ← d → 00 000 110	CB	4	6	23	
RLC (IY + d)		:	:	X	0	X	P	0 : 11 111 101 11 001 011 ← d → 00 000 110	FD CB	4	6	23	Instruction format and states are as shown for RLC's. To form new opcode replace 000 of RLC's with shown code.
RL m		:	:	X	0	X	P	0 : 010					
RRC m		:	:	X	0	X	P	0 : 001					
RR m		:	:	X	0	X	P	0 : 011					
SLA m		:	:	X	0	X	P	0 : 100					
SRA m		:	:	X	0	X	P	0 : 101					
SRL m		:	:	X	0	X	P	0 : 111					

m = r, (HL), (IX+d), (IY+d)

16-BIT LOAD GROUP (continued)

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
LD SP, HL	SP ← HL	•	•	X	•	X	•	• • 11 111 001	F9	1	1	6	
LD SP, IX	SP ← IX	•	•	X	•	X	•	• • 11 011 101 11 111 001	DD F9	2	2	10	
LD SP, IY	SP ← IY	•	•	X	•	X	•	• • 11 111 101 11 111 001	FD F9	2	2	10	
PUSH qq	(SP − 2) ← qq_L (SP − 1) ← qq_H SP ← SP − 2	•	•	X	•	X	•	• • 11 qq0 101		1	3	11	qq Pair 00 BC 01 DE 10 HL
PUSH IX	(SP − 2) ← IX_L (SP − 1) ← IX_H SP ← SP − 2	•	•	X	•	X	•	• • 11 011 101 11 100 101	DD E5	2	4	15	
PUSH IY	(SP − 2) ← IY_L (SP − 1) ← IY_H SP ← SP − 2	•	•	X	•	X	•	• • 11 111 101 11 100 101	FD E5	2	4	15	
POP qq	qq_H ← (SP + 1) qq_L ← (SP) SP ← SP + 2	•	•	X	•	X	•	• • 11 qq0 001		1	3	10	
POP IX	IX_H ← (SP + 1) IX_L ← (SP) SP ← SP + 2	•	•	X	•	X	•	• • 11 011 101 11 100 001	DD E1	2	4	14	
POP IY	IY_H ← (SP + 1) IY_L ← (SP) SP ← SP + 2	•	•	X	•	X	•	• • 11 111 101 11 100 001	FD E1	2	4	14	

Notes : dd is any of the register pairs BC, DE, HL, SP.
qq is any of the register pairs AF, BC, DE, HL.
$(PAIR)_H$, $(PAIR)_L$ refer to high order and low order eight bits of the register pair respectively.
e.g., BC_L = C, AF_H = A.

16-BIT ARITHMETIC GROUP

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
ADD HL, ss	HL ← HL + ss	•	•	X	X	X	•	0 : 00 ss1 001		1	3	11	ss Reg. 00 BC 01 DE 10 HL 11 SP
ADC HL, ss	HL ← HL + ss + CY	:	:	X	X	X	V	0 : 11 101 101 01 ss1 010	ED	2	4	15	
SBC HL, ss	HL ← HL + ss − CY	:	:	X	X	X	V	1 : 11 101 101 01 ss0 010	ED	2	4	15	
ADD IX, pp	IX ← IX + pp	•	•	X	X	X	•	0 : 11 011 101 01 pp1 001	DD	2	4	15	pp Reg. 00 BC 01 DE 10 IX 11 SP
ADD IY, rr	IY ← IY + rr	•	•	X	X	X	•	0 : 11 111 101 00 rr1 001	FD	2	4	15	
INC ss	ss ← ss + 1	•	•	X	•	X	•	• • 00 ss0 011		1	1	6	
INC IX	IX ← IX + 1	•	•	X	•	X	•	• • 11 011 101 00 100 011	DD 23	2	2	10	
INC IY	IY ← IY + 1	•	•	X	•	X	•	• • 11 111 101 00 100 011	FD 23	2	2	10	
DEC ss	ss ← ss − 1	•	•	X	•	X	•	• • 00 ss1 011		1	1	6	
DEC IX	IX ← IX − 1	•	•	X	•	X	•	• • 11 011 101 00 101 011	DD 2B	2	2	10	
DEC IY	IY ← IY − 1	•	•	X	•	X	•	• • 11 111 101 00 101 011	FD 2B	2	2	10	

Notes : ss is any of the register pairs BC, DE, HL, SP.
pp is any of the register pairs BC, DE, IX, SP.
rr is any of the register pairs BC, DE, IY, SP.

GENERAL-PURPOSE ARITHMETIC AND CPU CONTROL GROUPS

Symbol	Symbolic Operation	S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
DAA	Converters acc content into packed BCD following add or subtract with packed BCD operands	:	:	X	:	X	P	• : 00 100 111	27	1	1	4	Decimal Adjust Accumulator
CPL	A ← $\bar{A}$	•	•	X	1	X	•	1 • 00 101 111	2F	1	1	4	Complement Accumulator (one's complement)
NEG	A ← 0 − A	:	:	X	:	X	V	1 : 11 101 101 01 000 100	ED 44	2	2	8	Negate Acc. (two's complement)
CCF	CY ← $\bar{CY}$	•	•	X	X	X	•	0 : 00 111 111	3F	1	1	4	Complement Carry Flag.
SCF	CY ← 1	•	•	X	0	X	•	0 1 00 110 111	37	1	1	4	Set Carry Flag.
NOP	No Operation	•	•	X	•	X	•	• • 00 000 000	00	1	1	4	
HALT	CPU Halted	•	•	X	•	X	•	• • 01 110 110	76	1	1	4	
DI*	IFF ← 0	•	•	X	•	X	•	• • 11 110 011	F3	1	1	4	
EI*	IFF ← 1	•	•	X	•	X	•	• • 11 111 011	FB	1	1	4	
IM 0	Set Interrupt Mode 0	•	•	X	•	X	•	• • 11 101 101 01 000 110	ED 46	2	2	8	
IM 1	Set Interrupt Mode 1	•	•	X	•	X	•	• • 11 101 101 01 010 110	ED 56	2	2	8	
IM2	Set Interrupt Mode 2	•	•	X	•	X	•	• • 11 101 101 01 011 110	ED 5E	2	2	8	

Notes : IFF indicates the interrupt enable flip-flop.
CY indicates the carry flip-flop.
* indicates interrupts are not sampled at the end of EI or DI.

BIT SET, RESET & TEST GROUP

Symbol	Symbolic Operation	Flags S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
BIT b, r	$Z \leftarrow \overline{r_b}$	X	↕	X	1	X	X	0	•	11 001 011 01 b r	CB	2	2	8	r, r' Reg. 000 B 001 C 010 D 011 E 100 H 101 L 111 A b Bit Tested 000 0 001 1 010 2 011 3 100 4 101 5 110 6 111 7
BIT b, (HL)	$Z \leftarrow \overline{(HL)_b}$	X	↕	X	1	X	X	0	•	11 001 011 01 b 110	CB	2	3	12	
BIT b, (IX+d)	$Z \leftarrow \overline{(IX+d)_b}$	X	↕	X	1	X	X	0	•	11 011 101 11 001 011 ← d → 01 b 110	DD CB	4	5	20	
BIT b, (IY+d)	$Z \leftarrow \overline{(IY+d)_b}$	X	↕	X	1	X	X	0	•	11 111 101 11 001 011 ← d → 01 b 110	FD CB	4	5	20	
SET b, r	$r_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 001 011 11 b r	CB	2	2	8	
SET b, (HL)	$(HL)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 001 011 11 b 110	CB	2	4	15	
SET b, (IX+d)	$(IX+d)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 011 101 11 001 011 ← d → 11 b 110	DD CB	4	6	23	
SET b, (IY+d)	$(IY+d)_b \leftarrow 1$	•	•	X	•	X	•	•	•	11 111 101 11 001 011 ← d → 11 b 110	FD CB	4	6	23	
RES b, m	$m_b \leftarrow 0$ m = r, (HL), (IX + d), (IY + d)	•	•	X	•	X	•	•	•	10					To form new opcode replace 11 of SET b, s with 10. Flags and time states for SET instruction.

Note : The notation m_b indicates bit b (0 to 7) of location m.

SUMMARY OF FLAG OPERATION

Symbol	Operation
S	Sign Flag. S = 1 if the MSB of the result is 1.
Z	Zero Flag. Z = 1 if the result of the operation is 0.
P/V	Parity or Overflow Flag. Parity (P) and overflow (V) share the same flag. Logical operations affect this flag with the parity of the result while arithmetic operations affect this flag with the overflow of the result. If P/V holds parity, P/V = 1 if the result of the operation is even, P/V = 0 if result is odd. If P/V holds overflow, P/V = 1 if the result of the operation produced an overflow.
H	Half-carry Flag: H = 1 if the add or subtract operation produced a carry into or borrow from bit 4 of the accumulator.
N	Add/Subtract Flag. N = 1 if the previous operation was a subtract.
H & N	H and N flags are used in conjunction with the decimal adjust instruction (DAA) to properly correct the result into packed BCD format following addition or subtraction using operands with packed BCD format.
C	Carry/Link Flag. C = 1 if the operation produced a carry from the MSB of the operand or result.
↕	The flag is affected according to the result of the operation.
•	The flag is unchanged by the operation.
0	The flag is reset by the operation.
1	The flag is set by the operation.
X	The flag is a "don't care".
V	P/V flag affected according to the overflow result of the operation.
P	P/V flag affected according to the parity result of the operation.
r	Any one of the CPU Registers A, B, C, D, E, H, L
s	Any 8-bit location for all the addressing modes allowed for the particular instruction.
ss	Any 16-bit location for all the addressing modes allowed for that instruction.
ii	Any one of the two Index registers IX or IY.
R	Refresh Counter
n	8-bit Value in Range < 0..255 >
nn	16-bit Value in Range < 0..65535 >
d	8-bit Value in Range <-128 ; +127> {displacement}

CALL AND RETURN GROUP

Symbol	Symbolic Operation	Flags S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
CALL nn	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$ $PC \leftarrow nn$	•	•	X	•	X	•	•	•	11 001 101 ← n → ← n →	CD	3	5	17	
CALL cc, nn	If condition cc is false continue, otherwise same as CALL nn	•	•	X	•	X	•	•	•	11 cc 100 ← n → ← n →		3	3	10	If cc is false.
												3	5	17	If cc is true.
RET	$PC_L \leftarrow (SP)$ $PC_H \leftarrow (SP+1)$	•	•	X	•	X	•	•	•	11 001 001	C9	1	3	10	
RET cc	If condition cc is false continue, otherwise same as RET	•	•	X	•	X	•	•	•	11 cc 000		1	1	5	If cc is false.
												1	3	11	If cc is true
RETI	Return from interrupt	•	•	X	•	X	•	•	•	11 101 101 01 001 101	ED 4D	2	4	14	
RETN [1]	Return from non-maskable interrupt	•	•	X	•	X	•	•	•	11 101 101 01 000 101	ED 45	2	4	14	
RSTp	$(SP-1) \leftarrow PC_H$ $(SP-2) \leftarrow PC_L$ $PC_H \leftarrow 0$ $PC_L \leftarrow p$	•	•	X	•	X	•	•	•	11 t 111		1	3	11	

cc Condition: 000 NZ Non-zero; 001 Z Zero; 010 NC Non-carry; 011 C Carry; 100 PO Parity Odd; 101 PE Parity Even; 110 P Sign Positive; 111 M Sign Negative

t | p: 000 00H; 001 08H; 010 10H; 011 18H; 100 20H; 101 28H; 110 30H; 111 38H

Note : 1. RETN loads $IFF_2 \rightarrow IFF_1$.

INPUT AND OUTPUT GRUP

Symbol	Symbolic Operation	Flags S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
IN A, (n)	A ← (n)	•	•	X	•	X	•	•	•	11 011 011 ← n →	DB	2	3	11	n to A_0 - A_7 Acc. to A_8 - A_{15}
IN r, (C)	r ← (C) If r = 110 only the flags will be affected	↕	↕	X	↕	X	P	0	•	11 101 101 01 r 000	ED	2	3	12	C to A_0 - A_7 B to A_8 - A_{15}
OUT (n), A	(n) ← A	•	•	X	•	X	•	•	•	11 010 011 ← n →	D3	2	3	11	n to A_0 - A_7 Acc. to A_8 - A_{15}
OUT (C), r	(C) ← r	•	•	X	•	X	•	•	•	11 101 101 01 r 001	ED	2	3	12	C to A_0 - A_7 B to A_8 - A_{15}

SUMMARY OF FLAG OPERATION

Instruction	D_7 S	Z		H		P/V	N	D_0 C	Comments
ADD A, s ; ADC A, s	↕	↕	X	↕	X	V	0	↕	8-bit Add or Add with Carry.
SUB s; SBC A, s ; CP s ; NEG	↕	↕	X	↕	X	V	1	↕	8-Bit subtract, subtract with carry, compare and negate accumulator.
AND s	↕	↕	X	1	X	P	0	0	Logical Operations
OR s, XOR s	↕	↕	X	0	X	P	0	0	
INC s	↕	↕	X	↕	X	V	0	•	8-bit Increment
DEC s	↕	↕	X	↕	X	V	1	•	8-bit Decrement
ADD DD, ss	•	•	X	X	X	•	0	↕	16-bit Add
ADC HL, ss	↕	↕	X	X	X	V	0	↕	16-bit Add with Carry
SBC HL, ss	↕	↕	X	X	X	V	1	↕	16-bit Subtract with Carry.
RLA, RLCA, RRA ; RRCAA	•	•	X	0	X	•	0	↕	Rotate Accumulator.
RL m ; RLC m ; RR m ; RRC m ; SLA m SRA m ; SRL m	↕	↕	X	0	X	P	0	↕	Rotate and Shift Locations.
RLD ; RRD	↕	↕	X	0	X	P	0	•	Rotate Digit Left and Right
DAA	↕	↕	X	↕	X	P	•	↕	Decimal Adjust Accumulator.
CPL	•	•	X	1	X	•	1	•	Complement Accumulator
SCF	•	•	X	0	X	•	0	1	Set Carry
CCF	•	•	X	X	X	•	0	↕	Complement Carry
IN r, (C)	↕	↕	X	0	X	P	0	•	Input Register Indirect
INI, IND, OUTI ; OUTD INIR ; INDR ; OTIR ; OTDR	X X	↕ 1	X X	↕ X	X X	X X	1 1	↕ ↕	Block Input and Output. Z = 0 if B ≠ otherwise Z = 0
LDI ; LDD LDIR ; LDDR	X X	X X	X X	0 0	X X	↕ 0	0 0	• •	Block Transfer Instructions. P/V = 1 if BC ≠ 0, otherwise P/V = 0
CPI ; CPIR ; CPD ; CPDR	X	↕	X	X	X	↕	1	•	Block Search Instructions. Z = 1 if A = (HL), otherwise Z = 0. P/V = 1 if BC ≠ 0, otherwise P/V = 0
LD A, I; LD A, R	↕	↕	X	0	X	IFF	0	•	The content of the interrupt enable flip-flop (IFF) is copied into P/V flag.
BIT b, s	X	↕	X	1	X	X	0	•	The state of bit b of location is copied into the Z flag.

JUMP GROUP

Symbol	Symbolic Operation	Flags S	Z	H	P/V	N	C	Opcode 76 543 210	Hex	N° of Bytes	N° of M Cycles	N° of T States	Comments
JP nn	PC ← nn	•	•	X	•	X	•	•	•	11 000 011 ← n → ← n →	C3	3	3	10	
JP cc, nn	If condition cc is true PC ← nn, otherwise continue	•	•	X	•	X	•	•	•	11 cc 010 ← n → ← n →		3	3	10	cc Condition 000 NZ non-zero 001 Z zero 010 NC non-carry 011 C carry 100 PO parity odd 101 PE parity even 110 P sign positive 111 M sign negative
JR e	PC ← PC - e	•	•	X	•	X	•	•	•	00 011 000 ← e-2 →	18	2	3	12	
JR C, e	If C = 0 continue If C = 1 PC ← PC+e	•	•	X	•	X	•	•	•	00 111 000 ← e-2 →	38	2 2	2 3	7 12	If condition not met. If condition is met.
JR NC, e	If C = 1, continue If C = 0, PC ← PC+e	•	•	X	•	X	•	•	•	00 110 000 ← e-2 →	30	2 2	2 3	7 12	If condition not met. If condition is met.
JR Z, e	If Z = 0 continue If Z = 1 PC ← PC+e	•	•	X	•	X	•	•	•	00 101 000 ← e-2 →	28	2 2	2 3	7 12	If condition not met. If condition is met.
JR NZ, e	If Z = 1, continue If Z = 0, PC ← PC+e	•	•	X	•	X	•	•	•	00 100 000 ← e-2 →	20	2 2	2 3	7 12	If condition not met. If condition is met.

Notes : e represents the extension in the relative addressing mode.
e is a signed two's complement number in the range <- 126, 129>
e - 2 in the opcode provides an effective address of pc + e as PC is incremented by 2 prior to the addition of e.

PIN DESCRIPTIONS

A_0-A_{15}. *Address Bus* (Output, Active High, 3-state). A_0-A_{15} form a 16-bit address bus. The Address Bus provides the address for memory data bus exchanges (up to 64K bytes) and for I/O device exchanges.

BUSACK. *Bus Acknowledge* (Output, Active Low). Bus Acknowledge indicates to the requesting device that the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR have entered their high-impedance states. The external circuitry can now control these lines.

BUSREQ. *Bus Request* (Input, Active Low). Bus Request has a higher priority than NMI and is always recognized at the end of the current machine cycle. BUSREQ forces the CPU address bus, data bus, and control signals MREQ, IORQ, RD, and WR to go to a high-impedance state so that other devices can control these lines. BUSREQ is normally wire-ORed and requires an external pullup for these applications. Extended BUSREQ periods due to extensive DMA operations can prevent the CPU from properly refreshing dynamic RAMs.

D_0-D_7. *Data Bus* (Input/Output, Active High, 3-state). D_0-D_7 constitute an 8-bit bidirectional data bus, used for data exchanges with memory and I/O.

HALT. *Halt State* (Output, Active Low). HALT indicates that the CPU has executed a Halt instruction and is awaiting either a non-maskable or a maskable interrupt (with the mask enabled) before operation can resume. While halted, the CPU executes NOPs to maintain memory refresh.

INT. *Interrupt Request* (Input, Active Low). Interrupt Request is generated by I/O devices. The CPU honors a request at the end of the current instruction if the internal software-controlled interrupt enable flip-flop (IFF) is enabled. INT is normally wire-ORed and requires an external pullup for these applications.

IORQ. *Input/Output Request* (Output, Active Low, 3-state). IORQ indicates that the lower half of the address bus holds a valid I/O address for an I/O read or write operation. IORQ is also generated concurrently with M1 during an interrupt acknowledge cycle to indicate that an interrupt response vector can be placed on the data bus.

M1. *Machine Cycle One* (Output, Active Low). M1, together with MREQ, indicates that the current machine cycle is the opcode fetch cycle of an instruction execution. M1, together with IORQ, indicates an interrupt acknowledge cycle.

MREQ. *Memory Request* (Output, Active Low, 3-state). MREQ indicates that the address bus holds a valid address for a memory read or memory write operation.

NMI. *Non-Maskable Interrupt* (Input, negative edge-triggered). NMI has a higher priority than INT. NMI is always recognized at the end of the current instruction, independent of the status of the interrupt enable flip-flop, and automatically forces the CPU to restart at location 0066H.

RD. *Read* (Output, Active Low, 3-state). RD indicates that the CPU wants to read data from memory or an I/O device. The addressed I/O device or memory should use this signal to gate data onto the CPU data bus.

RESET. *Reset* (Input, Active Low). RESET initializes the CPU as follows : it resets the interrupt enable flip-flop, clears the PC and Registers I and R, and sets the interrupt status to Mode 0. During reset time, the address and data bus go to a high-impedance state, and all control output signals go to the inactive state.

Note that RESET must be active for a minimum of three full clock cycles before the reset operation is complete.

RFSH. *Refresh* (Output, Active Low). RFSH, together with MREQ, indicates that the lower seven bits of the system's address bus can be used as a refresh address to the system's dynamic memories.

WAIT. *Wait* (Input, Active Low). WAIT indicates to the CPU that the addressed memory or I/O devices are not ready for a data transfer. The CPU continues to enter a Wait state as long as this signal is active. Extended WAIT periods can prevent the CPU from refreshing dynamic memory properly.

WR. *Write* (Output, Active Low, 3-state). WR indicates that the CPU data bus holds valid data to be stored at the addressed memory or I/O location.

Z80 Set Assembler & L.M. (NOT COMPLETE)

this side → (macchine le istruzioni composte da I/O)

BIT	CALL	Jump
FLAG	I/O	PIN
	FLAG	

other Side → (Macchine alcune istruzioni: composte, BCD EXANGE BANK)

LD 8	LD16	PUSH
ALU8	ROT SHIFT	ALU_{16}
		CONTROL