

8080 MICROPROCESSOR

Mnemonic	Bytes	Cycles	Description of Operation
INR r	1	1	$(r) \leftarrow (r) + 1$ The content of register r is incremented by one. All the condition flip-flops except carry are affected by the result.
DCR r	1	1	$(r) \leftarrow (r) - 1$ The content of register r is decremented by one. All of the condition flip-flops except carry are affected by the result.
ADD r	1	1	$(A) \leftarrow (A) + (r)$ Add the content of register r to the content of register A and place the result into register A. (All flags affected.)
ADC r	1	1	$(A) \leftarrow (A) + (r) + (\text{carry})$ Add the content of register r and the contents of the carry flip-flop to the content of the A register and place the result into Register A. (All flags affected.)
SUB r	1	1	$(A) \leftarrow (A) - (r)$ Subtract the content of register r from the content of register A and place the result into register A. Two's complement subtraction is used. (All flags affected.)
SBB r	1	1	$(A) \leftarrow (A) - (r) - (\text{borrow})$ Subtract the content of register r and the content of the carry flip-flop from the content of register A and place the result into register A. (All flags affected.)
ANA r	1	1	$(A) \leftarrow (A) \wedge (r)$ Place the logical product of the register A and register r into register A. (Resets carry.)
XRA r	1	1	$(A) \leftarrow (A) \vee (r)$ Place the "exclusive - or" of the content of register A and register r into register A. (Resets carry.)
ORA r	1	1	$(A) \leftarrow (A) \vee (r)$ Place the "inclusive - or" of the content of register A and register r into register A. (Resets carry.)
CMP r	1	1	$(A) - (r)$ Compare the content of register A with the content of register r. The content of register A remains unchanged. The flag flip-flops are set by the result of the subtraction. Equality ($A = r$) is indicated by the zero flip-flop set to "1." Less than ($A < r$) is indicated by the carry flip-flop, set to "1."
ADD M	1	2	$(A) \leftarrow (A) + (M)$ ADD
ADC M	1	2	$(A) \leftarrow (A) + (M) + (\text{carry})$ ADD with carry
SUB M	1	2	$(A) \leftarrow (A) - (M)$ SUBTRACT
SBB M	1	2	$(A) \leftarrow (A) - (M) - (\text{borrow})$ SUBTRACT with borrow
ANA M	1	2	$(A) \leftarrow (A) \wedge (M)$ Logical AND
XRA M	1	2	$(A) \leftarrow (A) \vee (M)$ Exclusive OR
ORA M	1	2	$(A) \leftarrow (A) \vee (M)$ Inclusive OR
CMP M	1	2	$(A) - (M)$ COMPARE
ADI <B ₂ >	2	2	$(A) \leftarrow (A) + \langle B_2 \rangle$ ADD
ACI <B ₂ >	2	2	$(A) \leftarrow (A) + \langle B_2 \rangle + (\text{carry})$ ADD with carry
SUI <B ₂ >	2	2	$(A) \leftarrow (A) - \langle B_2 \rangle$ SUBTRACT
SBI <B ₂ >	2	2	$(A) \leftarrow (A) - \langle B_2 \rangle - (\text{borrow})$ SUBTRACT with borrow
ANI <B ₂ >	2	2	$(A) \leftarrow (A) \wedge \langle B_2 \rangle$ Logical AND
XRI <B ₂ >	2	2	$(A) \leftarrow (A) \vee \langle B_2 \rangle$ Exclusive OR
ORI <B ₂ >	2	2	$(A) \leftarrow (A) \vee \langle B_2 \rangle$ Inclusive OR
CPI <B ₂ >	2	2	$(A) - \langle B_2 \rangle$ COMPARE
RLC	1	1	$A_{m+1} \leftarrow A_m, A_0 \leftarrow A_7, (\text{carry}) \leftarrow A_7$ Rotate the content of register A left one bit. Rotate A ₇ into A ₀ and into the carry flip-flop.
RRC	1	1	$A_m \leftarrow A_{m+1}, A_7 \leftarrow A_0, (\text{carry}) \leftarrow A_0$ Rotate the content of register A right one bit. Rotate A ₀ into A ₇ and into the carry flip-flop.

(M) addressed by the contents of registers H and L.

Flags affected are same as non-memory reference instructions.