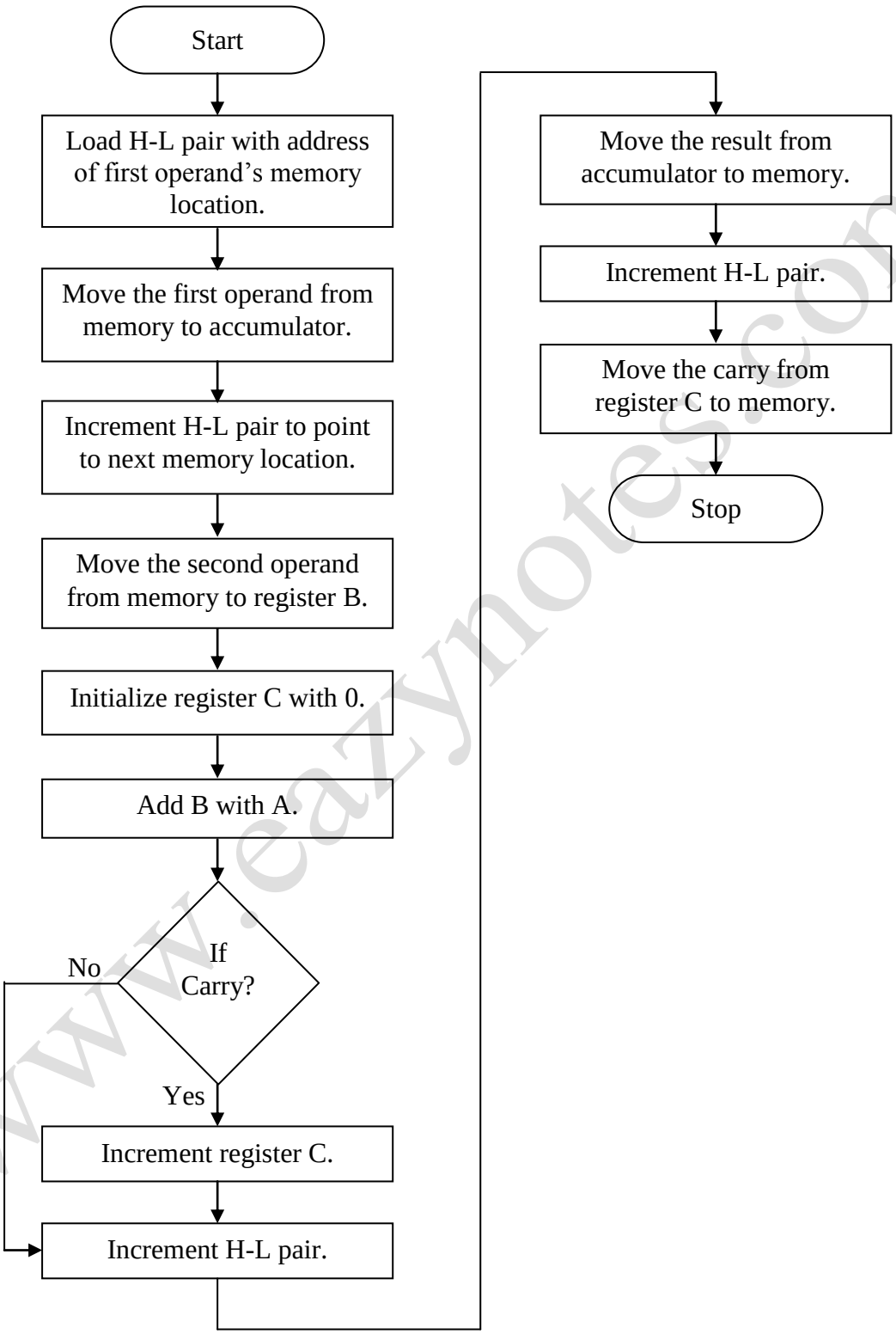
Program 10: Add two 8-bit numbers along with considering the carry.

Flowchart:



Program:

Address	Mnemonics	Operand	Opcode	Remarks
2000	LXI	H, 3000H	21	Load H-L pair with address 3000H.
2001			00	Lower-order of 3000H.
2002			30	Higher-order of 3000H.
2003	MOV	A, M	7E	Move the 1st operand from memory to reg. A.
2004	INX	H	23	Increment H-L pair.
2005	MOV	B, M	46	Move the 2nd operand from memory to reg. B.
2006	MVI	C, 00H	0E	Initialize reg. C with 00H.
2007			00	Immediate value 00H.
2008	ADD	B	80	Add B with A.
2009	JNC	200D	D2	Jump to address 200DH if there is no carry.
200A			0D	Lower-order of 200DH.
200B			20	Higher-order of 200DH.
200C	INR	C	0C	Increment reg. C.
200D	INX	H	23	Increment H-L pair.
200E	MOV	M, A	77	Move the result from reg. A to memory.
200F	INX	H	23	Increment H-L pair.
2010	MOV	M, C	71	Move carry from reg. C to memory.
2011	HLT		76	Halt.

Explanation:

- This program adds two operands stored in memory location 3000H and 3001H, along with considering the carry produced (if any).
- Let us assume that the operands stored at memory location 3000H is FAH and 3001H is 28H.
- Initially, H-L pair is loaded with the address of first memory location.
- The first operand is moved to accumulator from memory location 3000H and H-L pair is incremented to point to next memory location.
- The second operand is moved to register B from memory location 3001H.
- Register C is initialized to 00H. It stores the carry (if any).
- The two operands stored in register A and B are added and the result is stored in accumulator.
- Then, carry flag is checked for carry. If there is a carry, C register is incremented.
- H-L pair is incremented and the result is moved from accumulator to memory 3002H.
- H-L pair is again incremented and carry (either 0 or 1) is moved from register C to memory location 3003H.

Output:

Before Execution:

3000H: FAH

3001H: 28H

After Execution:

3002H: 22H

3003H: 01H