Experiment 6

Aim: To add two binary numbers, each 8 byte long

Software Used:

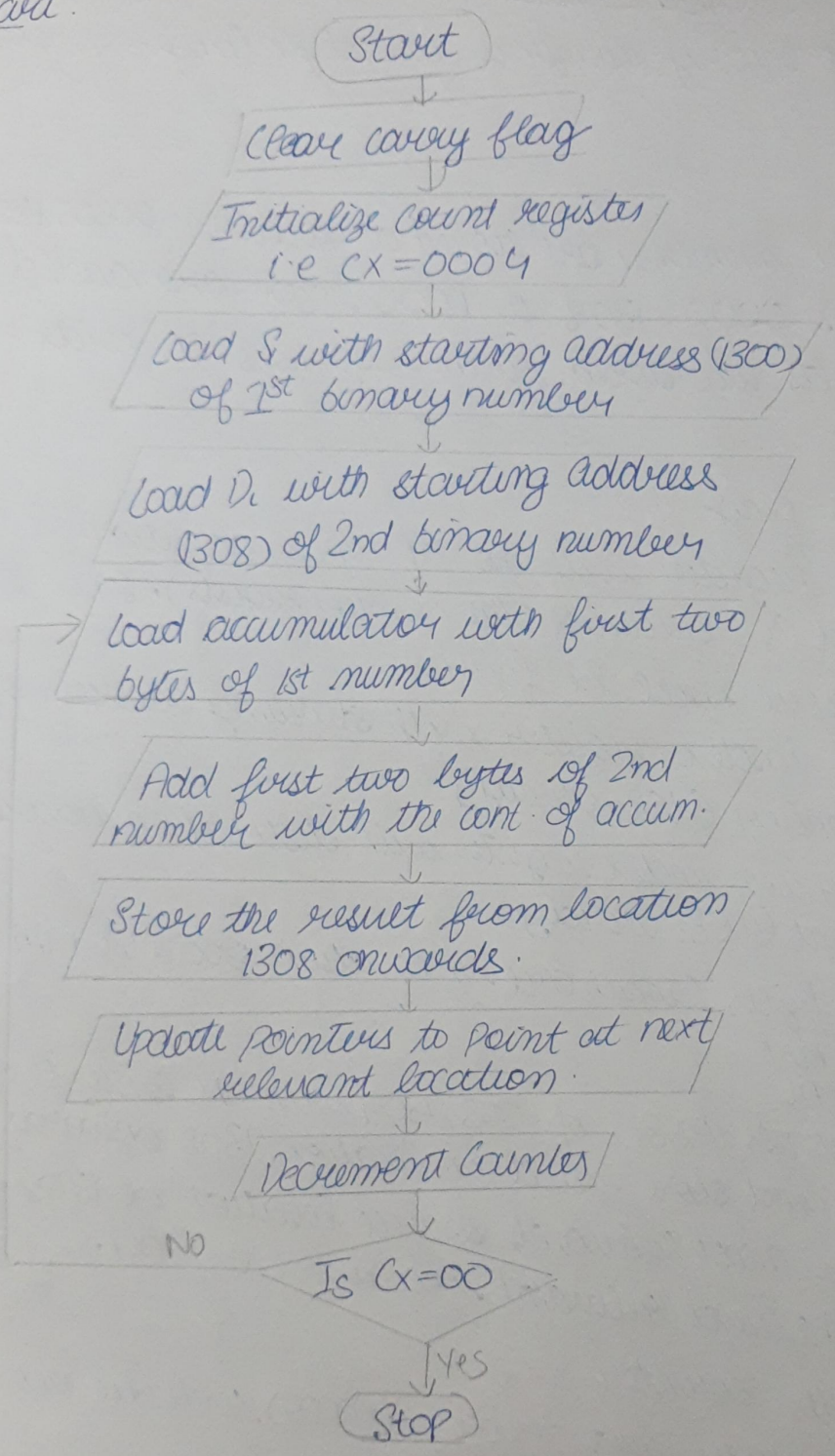
Theory:

Two eight byte numbers are stored at location 0000:1300 to 0000:1307 & 0000:1308 to 0000:130F are added and the results are stored from 0000:1308 onwards

Algorithm:

i) Clear carry flag
2) Load counter register with no. of times addition is to be performed (i.e initialize the counter register). Since 8086 is 16 bit, value will be four.
3) Load source index register with starting address of 1st binary number (LSB's array)
4) Load destination index register with destination address' starting point (MSBs array)
5) Load data bytes (which are in location 1300 & 1301 in 16 bit A)
   i.e 1300 - AH
   1301 - AL
6) Add the contents (MSBs) of 1308, 1309 with the contents of 1300+1301 (i.e. LSBs) and store result in location 1308 onwards.
7) Point at the next relevant source location i.e 1302.
8) Point at the next relevant location (i.e 1304.
9) Decrement the counter.
10) If count register not zero (i.e X≠0000), continue addition
11) Else Halt.

Flowchart:


Start
Clear carry flag
Initialize count register i.e CX=0004
Load SI with starting address (1300) of 1st binary number
Load DI with starting address (1308) of 2nd binary number
Load accumulator with first two bytes of 1st number
Add first two bytes of 2nd number with the cont. of accum.
Store the result from location 1308 onwards.
Update pointers to point at next relevant location.
Decrement Counter
Is CX=00
No
Yes
Stop

Program

LABEL	ADDRESS	MNEMONICS	OPERAND	COMMENT.
	0200	CLC		Clear the carry flag
	0201	MOV CX	0004	Initialize counter register with no of times addition will take place (16×4 = 64 i.e 4 time)
	0204	MOV SI	1300	Load SI register with LSBs array's starting address
	0207	MOV DI	1308	Load destination register with starting add of MSB
LOOP	020A	MOV AX, [SI]		Load the data bytes from SI to accumulator
	020C	ADC [DI], AX		Add accumulator with DI
	020E	INC SI		Increment SI register
	020F	INC SI		Increment SI register
	0210	INC DI		Increment DI register
	0211	INC DI		Increment DI register
	0212	DEC CX		Decrement count register
	0213	JNZ LOOP		Jump to loop (020A) if count register is not zero
	0215	HLT		HALT

	BEFORE	AFTER
0000:1300	01	
0000:1301	02	
0000:1302	03	
0000:1303	04	
0000:1304	05	
0000:1305	06	
0000:1306	07	
0000:1307	08	
0000:1308	0A	0B
0000:1309	0B	0D
0000:130A	0C	0F
0000:130B	0E	12
0000:130C	0F	14
0000:130D	10	16
0000:130E	11	18
0000:130F	12	1A

RESULT: Two 8byte long binary numbers were added & their outputs was stored