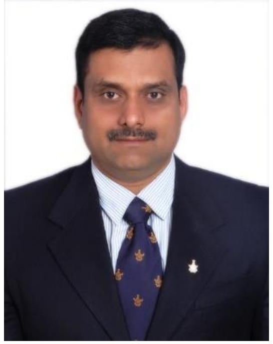
MPI Tutorial-6

8086 Arithmetic Operations ALPs

By Dr. Sanjay Vidhyadharan



Problem-1: Square root of a Number

Write an assembly language program in 8086 microprocessor to find square root of a number.

AX	BX	AX-BX	CX	CX=CX+1 and BX=BX+2 if AX ≠BX
25	1	24	1	Assume CX=0 initially
24	3	21	2	
21	5	16	3	
16	7	9	4	
9	9	0		

Problem-1: Square root of a Number

Write an assembly language program in 8086 microprocessor to find square root of a number.

AX	BX	AX-BX	CX	CX=CX+1 and BX=BX+2 if AX ≠BX
49	1	48	2	Assume CX=1 initially
48	3	45	3	
45	5	40	4	
40	7	33	5	
33	9	24	6	
24	11	13	7	
13	13	0		

Problem-1

Write an assembly language program in 8086 microprocessor to find square root of a number.

Solution:

Algorithm:

1. Move the input data in register AX
2. Move the data 0000 in CX and FFFF in BX
3. Add 0002 to the contents of BX
4. Increment the content of CX by 1
5. Subtract the contents of AX and BX
6. If Zero Flag(ZF) is not set go to step 3 else go to step 7
7. Store the data from CX to offset 600
8. Stop

❖N.B.: Will only work for perfect squares.

Problem-1

Write an assembly language program in 8086 microprocessor to find square root of a number.

Solution:

```
      MOV AX, [0500H]   //Place where the number is stored
      MOV CX, 0000
      MOV BX, FFFF
L1:   ADD BX, 02
      INC CX
      SUB AX, BX
      JNZ L1
      MOV [0600H], CX //Place where the result is stored
      HLT
```

❖N.B.: Will only work for perfect squares.

Problem-2

Write a program to find the min value in a given array in assembly 8086 microprocessor
.

Problem-2

Write a program to find the min value in a given array in assembly 8086 microprocessor

- **Solution:**

1. Assign value 500 in SI and 600 in DI
2. Move the contents of [SI] in CL and increment SI by 1
3. Assign the value 00 H to CH
4. Move the content of [SI] in AL
5. Decrease the value of CX by 1
6. Increase the value of SI by 1
7. Move the contents of [SI] in BL
8. Compare the value of BL with AL
9. Jump to step 11 if carry flag is set
10. Move the contents of BL in AL
11. Jump to step 6 until the value of CX becomes 0, and decrease CX by 1
12. Move the contents of AL in [DI]
13. Halt the program

Problem-2

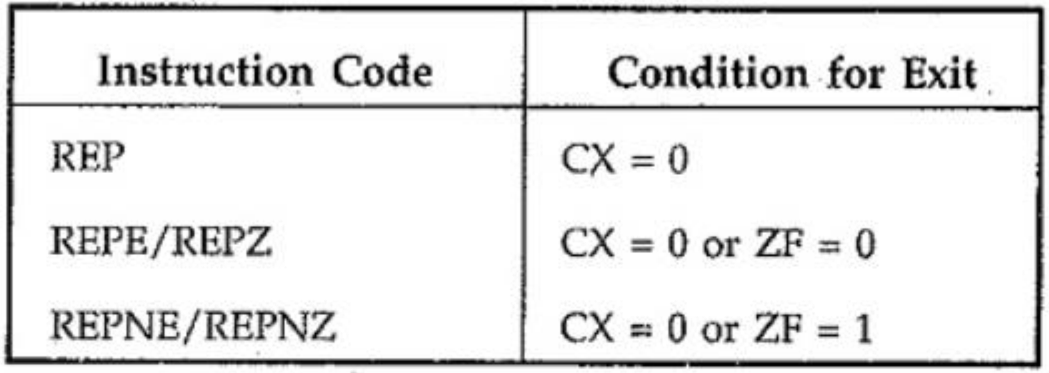
Write a program to find the min value in a given array in assembly 8086 microprocessor .

ADDRESS	MNEMONICS	COMMENTS
0400	MOV SI, 500	SI ← 500
0403	MOV DI, 600	DI ← 600
0406	MOV CL, [SI]	CL ← [SI]
0408	MOV CH, 00	CH ← 00
040A	INC SI	SI ← SI+1
040B	MOV AL, [SI]	AL ← [SI]
040D	DEC CX	CX ← CX-1
040E	INC SI	SI ← SI+1
040F	MOV BL, [SI]	BL ← [SI]
0411	CMP AL, BL	AL-BL
0413	JC 0417	Jump if carry is 1
0415	MOV AL, BL	AL ← BL
0417	LOOP 040E	Jump if CX not equal to 0
0419	MOV [DI], AL	[DI] ← AL
041B	HLT	End of the program

REPE/REPNE/REPZ/REPNZ

REPE and **REPZ** are mnemonics for the same prefix; they stand for Repeat if Equal and Repeat if Zero respectively. REPE/REPZ causes the succeeding string instruction to be repeated as long as the compared bytes or words are equal (ZF = 1) **and** CX is not yet counted down to zero.

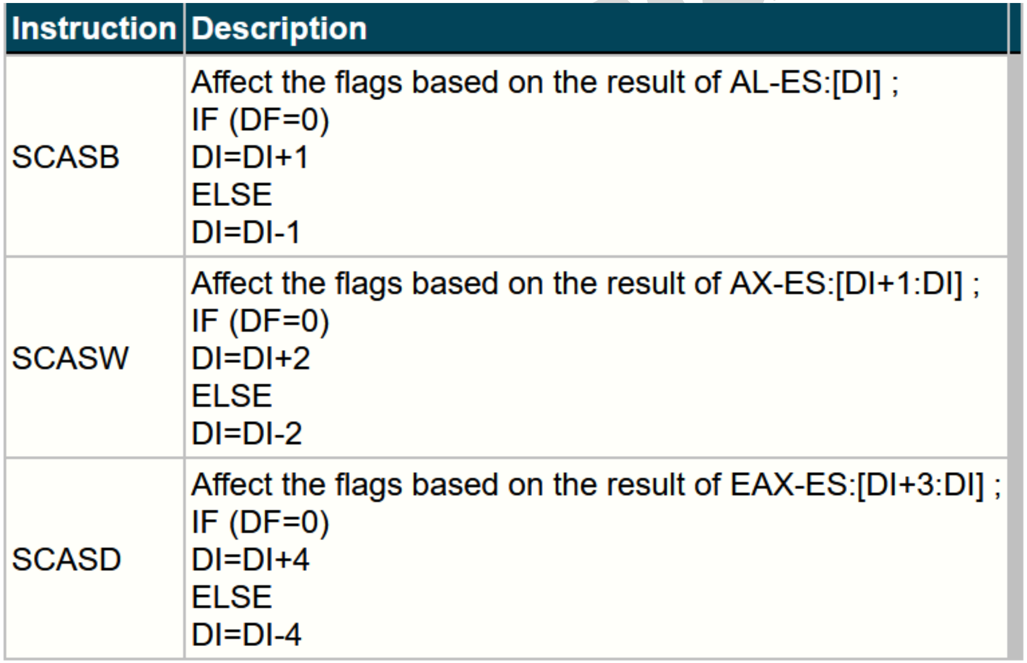
The **REPNE** and the **REPNZ** instructions stand for Repeat if Not Equal and Repeat if Not Zero respectively and cause the string instruction to be repeated until the compared bytes or words are equal (ZF = 1) **or** until CX = 0 (end of string.)



Instruction Code	Condition for Exit
REP	CX = 0
REPE/REPZ	CX = 0 or ZF = 0
REPNE/REPNZ	CX = 0 or ZF = 1

SCAS

The SCAS instruction is used for searching a particular character or set of characters in a string. The data item to be searched should be in AL (for SCASB), AX (for SCASW) or EAX (for SCASD) registers. The string to be searched should be in memory and pointed by the ES:DI (or EDI) register.



Instruction	Description

SCASB	Affect the flags based on the result of AL-ES:[DI] ;
	IF (DF=0)
	DI=DI+1
	ELSE
	DI=DI-1

SCASW	Affect the flags based on the result of AX-ES:[DI+1:DI] ;
	IF (DF=0)
	DI=DI+2
	ELSE
	DI=DI-2

SCASD	Affect the flags based on the result of EAX-ES:[DI+3:DI] ;
	IF (DF=0)
	DI=DI+4
	ELSE
	DI=DI-4

Problem-3

- Write an assembly language program in 8086 microprocessor to search a number in a string of 5 bytes, store the offset where the element is found in DX and the number of iterations used to find the number in BX.
- For e.g. if the numbers in the memory location starting from location 0600H is given as follows and we have to find the number 25:

0600	0601	0602	0603	0604
45	A5	25	78	9C

Output

0602	0002
DX	BX

Problem-3

```
MOV AX, 2000      //starting location of ES
MOV ES, AX
MOV DI, 600 //starting location of first string
MOV AL, 25  //The number to search
MOV CX, 0005      //No. of items
MOV BX, CX
CLD
REPNE SCAS B      //Repeat till ZF = 0. Scan value from [DI]
                  and compare with AL, Increment DI
DEC DI
MOV DX, DI
SUB BX, CX
DEC BX
INT 03H
```

Problem-4

Write an ALP to find 2’s complement of a string of 100 bytes.

Problem-4

Write an ALP to find 2’s complement of a string of 100 bytes.

```
2000    CLD                    : Clear direction flag
2001    MOV SI, 4000 H         : Source address put in SI
2004    MOV DI, 5000 H         : Destination address put in DI
2007    MOV CX, 0064 H         : Put the number of bytes to be 2’s
                               Complemented in CX
200A    LODSB                  : Data byte to AL and INC SI
200B    NEGAL                  : 2’s Complement of AL
200D    STOSB                  : Current AL value into DI and INC DI
200E    LOOPNZ 200A H          : Loop till CX = 0.
2010 HLT                       : Stop.
```

Thank You