

Lab 1.

procedure steps :-

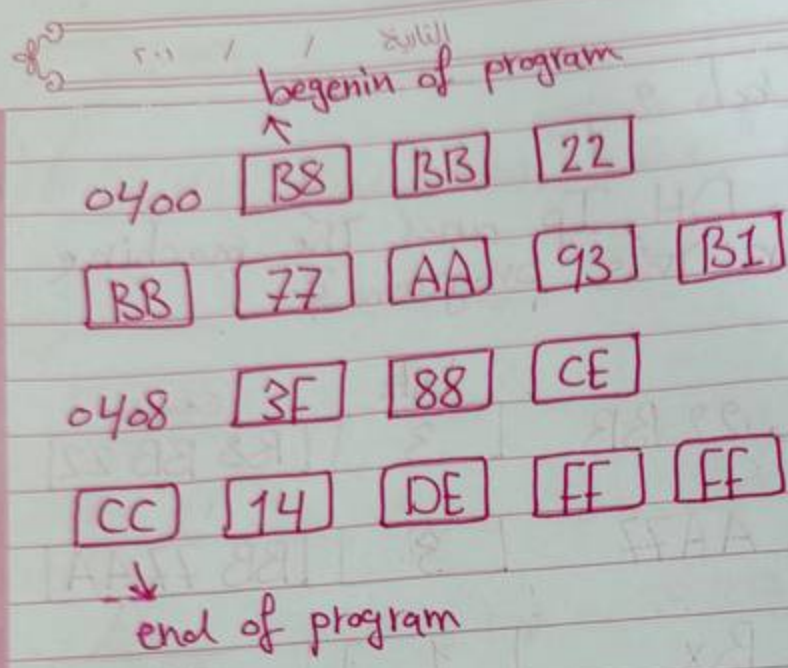
Button

- ① after turn ups on press reset push Button.
- ② press Enter in Keyboard.
- ③ The following menu will display on LCD.
A. D. F. G. I. M. P. T. U
- ④ To used The Assemble command press A Then press Enter The LCD Display as a shown below:-

0400

- ⑤ Insert basic Instruction
- ⑥ To End The program Int ↵ 3
- ⑦ To Execute The program and find the result
F7 → G ↵ → F7 → R

add	program	byte
0400	MOV _{space} Ax, 3344 ↵	3
0403	MOV Bx, 55BB	3
0406	MOV CL, 77	2



Discussion 3:
 what is difference between these two INS
 Mov AX, BX
 Xchg AX, BX

Sol Mov is mine copy the code for AX or BX and but the code in cide it the xchg mine take the code from AX and BX and but it incide AX

o/Sol The defference between the two Ins is
 (Mov AX, BX) takes 3b and the (Xch
 AX, BX) takes 1b.

- Lab 2 -

Find Ax, Bx, cL, DH, Ip and The machine Language code for This program?

add		byte	codes
0400	MOV AX, 22BB	3	B8 BB 22
0403	MOV BX, AA77	3	BB 77 AA
0406	Xchg Ax, Bx	1	93
0407	MOV CL, 3F	2	B1 3F
0409	MOV DH, CL	2	88 CE
040B	INT 3 ↴	1	CC
040C			

F7 → G ↴ → F7 → R

AX = AA77, BX = 22BB, CX = 3F, DH = 3F
Ip = 040B

To find The machine code ↴

F7 → D → 0400 ↴

0408 Mov [1000], AX 4

040C Int 3 ↓ 1

040D

Find AX, BX, CL, IP?

AX = 3344, BX = 55BB, CX = 0077

IP = 040C

Discussion :-

is this INS True? why?

Mov DL, 6789

Sol No. is NOT True Because DL is take
~~2~~ 1 Byte NOT one byte

- Lab 4 -

Ex write a program to find The average of these Two No's 06, 04 Store the First No off Set address 2000 The Second No off Set add 2001 and the result of average No in off Set add 2002 Find

AL=05 BL=04 CL=02 IP=0416 FL=F093
MLC?

add		M.L.C
0400	MOV AL, 06	B0 06
0402	MOV [2000], AL	88 06 00 20
0406	MOV BL, 04	B3 04
0408	MOV [2001] BL	88 1E 01 20
040C	ADD AL, BL	00 D8
040E	MOV CL, 02	B1 02
0410	DIV CL	F6 F1
0412	MOV [2002], AL	88 06 02 20
0416	INT 3 ↓	CC
0417		

add

0400

MOV AX, 82A5

B8 A5 82

0403

MOV BX, 526F

BB 6F 52

0406

MOV DX, BX

89 DA

0408

MOV CX, AX

89 C1

040A

ADD CX, BX

01 D9

040C

SUB DX, AX

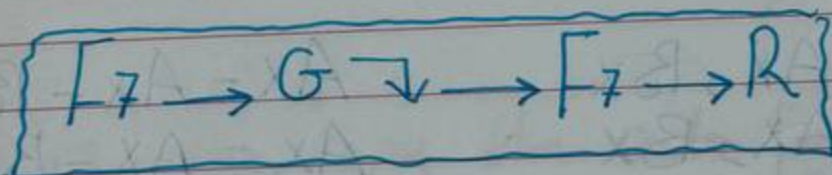
29 C2

040E

INT 3 ↓

CC

040F



Find 30

AX = 82A5 BX = 526F CX = D514

DX = CFCA IP = 0406 FL = F887

M.L.C

Discussion 30

Find 30

$$\begin{array}{r}
 82A5 \\
 + 526F \\
 \hline
 D514
 \end{array}$$

- Lab 3 -

Arithmetic Instructions :-
Additions :-

ADD, ADC, INC

Ex. ADD AX, BX
ADC AX, BX
INC AX

$$BX = BX + AX$$
$$AX = AX + BX$$

$$AX = AX + BX + CF$$
$$AX = AX + 1$$

Subtraction :-
SUB, SBB, DEC

Ex. SUB AX, BX
SBB AX, BX
DEC AX

$$AX = AX - BX$$
$$AX = AX - BX - CF$$
$$AX = AX - 1$$

Multiplication and Division :-

MUL, DIV

Ex. MUL BL
DIV BL

$$AX = AL - BL$$
$$AX = \frac{AX}{BL}$$

Find step by step the results of program?

Ax = Bx = Ip = FL =

* to find the results step by step as follows

T $\equiv$ Trace

تكرار

F7 $\rightarrow$ T $\rightarrow$ 01 $\downarrow$ $\rightarrow$ [F7 $\rightarrow$ G $\downarrow$ $\rightarrow$ F7 $\rightarrow$ R]

Discussion 3,

* Find 32BA OR 45F2

32BA $\rightarrow$	0011	0010	1011	1010
45F2 $\rightarrow$	0100	0101	1111	0010
	<u>0111</u>	<u>0111</u>	<u>1111</u>	<u>1010</u>
	7	7	F	A

* Find NOT 13513

FFFF
<u>13513</u>
4AEC

- Lab 5 -

Logical Instruction :- (AND, OR, XOR, NOT)

AND :- any (0) = 0

OR :- any (1) = 1

XOR :- Different (1) and the Same (0)

NOT :- Subtract AX from FFFF.

program :-

0400 MOV AX, 32BA

0403 MOV BX, 45F2

0406 OR AX, BX

0408 AND AX, 1234

040B XOR AX, A723

040E NOT AX

0410 INT 3 ↓

0411

② Show what this INS means?
MOV [2000], AL

Sol It means store the contents of AL in address [2000].

* where is the result of division process stored?

Insert The program line below:

MOV AX, 31A4

NOT AX

XOR AX, 2032

(4) ROR AX, 1

OR AX, 1234

CLC

(7) RCR AX, 1

INT 3

Find The result of this program step by step with FL?

Discussion:

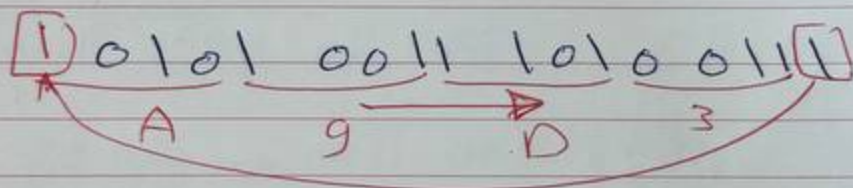
Find theoretically the result in step 4 and step 7?

Lab: 6

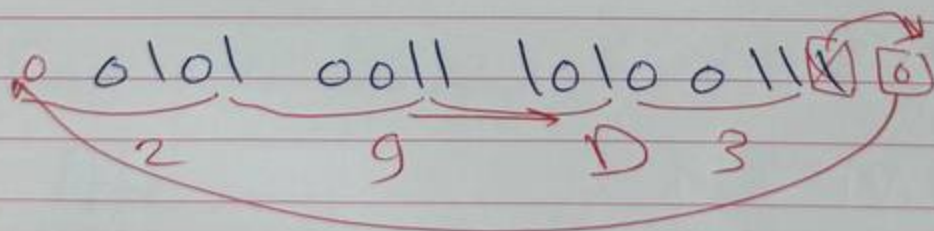
Rotate Instruction:

- * ROR (Rotate Right)
- * ROL (Rotate Left)
- * RCR (Rotate Right with Carry)
- * RCL (Rotate Left with Carry)
- * CLC clear Carry CF=0
- * STC Set Carry CF=1

Ex: MOV AX, 53A7
 ROR AX, 1



Ex: MOV AX, 53A7.
 CLC
 RCR AX, 1



program?

L =

p as

6

F7 → R

$$AX_1 = 32BA \quad BX_1 = 05C2 \quad Ip_1 = 0403 \quad FL_1 = F102$$

$$AX_2 = 32BA \quad BX_2 = 45F2 \quad Ip_2 = 0406 \quad FL_2 = F102$$

$$AX_3 = 77FA \quad BX_3 = 45F2 \quad Ip_3 = 0408 \quad FL_3 = F106$$

$$AX_4 = 2230 \quad BX_4 = 45F2 \quad Ip_4 = 040B \quad FL_4 = F106$$

$$AX_5 = 45F2 \quad BX_5 = 45F2 \quad Ip_5 = 040E \quad FL_5 = F182$$

$$AX_6 = 4AEC \quad BX_6 = 45F2 \quad Ip_6 = 0410 \quad FL_6 = F182$$

$\overbrace{0011}^6 \quad \overbrace{0010}^4 \quad \overbrace{0111}^4 \quad \overbrace{1010}^5 \quad 1$

0010	0001	0011	0011
1110	0100	1111	0101
0100	1010	1100	0111
4	5	c	6

$\overbrace{1000}^8 \quad \overbrace{0101}^B \quad \overbrace{1100}^8 \quad \overbrace{0110}^c \quad 0$

0400 MOV AX, 327A
 STC
 RCL AX, 1
 XOR AX, 2133
 ROL AX, 1
 INT 3

MOV AX, 327A
 STC
 RCL AX, 1
 XOR AX, 2133
 MOV CL, 04
 ROL AX, CL
 INT 3

Discussion :

Find Theoretically the Result of each step?

F₇ → T → 01 → F₇ → G → F₇ → R

① AX = 327A
 FL = F102

② AX = 327A
 FL = F103

③ AX = 64F5
 FL = F102

④ AX = 45C6
 FL = F406

⑤ AX = 8B8C
 FL = F906

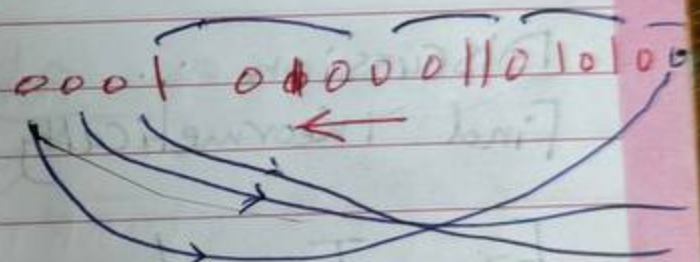
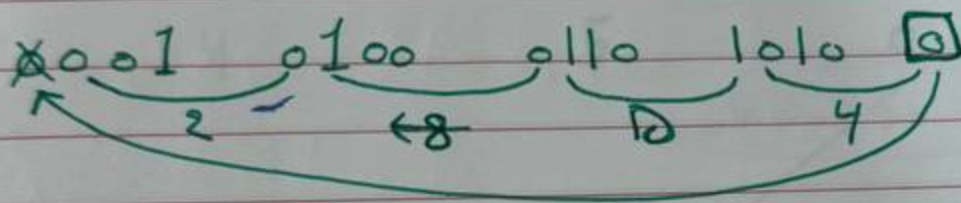
Lab 7

Rotate Instruction ..

Rol, Rcl

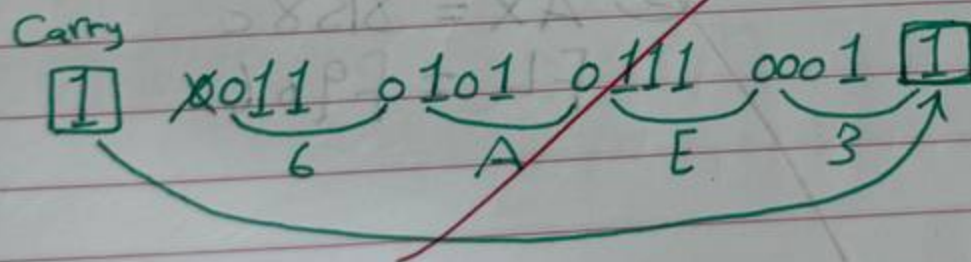
Ex Mov Ax, 146A
Rcl Ax, 1

Mov Ax, 146A
Mov Cl, 03
~~Rol~~ Rcl Ax, Cl



Ex Mov Ax, 3571
STC cf = 1
Rcl Ax, 1

A 350



Find step by step The result of this program Ax
FL?

Lab -9-

Write a program to find the maximum No. in a given string (4 byte) and store it in Last Location
The Data are :-

0600 22
0601 37
0602 1A
0603 B3

Find in each Loop
the result :
SI, CX, AH, IP, FL

```

0400 MOV SI, 0600
0403 MOV CX, 04          T=05
0406 MOV AH, 00
0408 CMP AH, [SI]
040A JNB 040E
040C MOV AH, [SI]
040E INC SI
040F LoopNE 0408 < DEC CX
0411 MOV [SI], AH      JNZ 0408
0413 INT 3
    
```

To fill the Data in memory as follow

F7 → F

① BL = C2

② BL = 43

③ BL = 43

④ BL = 43

⑤ BL = 00

⑥ BL = 00

FL = F102

FL = F102

FL = F193

FL = F193

FL = F143

FL = F143

Discussion

* write The procedure How to find the add. if we use Jump Inst.?

JSI
* We replace the address with Imaginary add

* What is mean of this Inst.?
CMP CX, AX.

Lab 8

Compare Instruction:

CMP

```

EX1  MOV AX, 2000
      MOV BX, 2500
      CMP AX, BX

```

D-S

JZ **

Jump Instructions:

* unconditional jump. **JMP**

* **Conditional jump:**

JE/JZ jump if equal / jump if zero

JNE/JNZ jump if not / jump if not zero

JNAE/JB jump if not above

JAE/JNB jump if above or equal / jump if

* Find the value of BL, FL not below.
for this program step by step?

0400 MOV AL, 49

0402 MOV BL, 43

0404 CMP BL, AL

0406 JNZ (A33) add.

0408 MOV BL, 52

040A (A2) INT 3

040B (A33) MOV BL, 100

JMP (A2) add.

A2 = 040A

A33 = 040B

- Lab 10 -

Write a program to find the minimum No. in a given string (5 byte) The Data are:

0800	B7
0801	35
0802	37
0803	72
0408	55

Find the result in each loop AH, CX, DI, IP, FI

0400	MOV DI,	
0403	MOV CX,	
0406	MOV AH, FF	
0408	CMP AH, [DI]	
040A	JB 040E	
040C	MOV AH, [DI]	
040E	INC DI	
040F	LOOPNE 0408	
0411	INT 3	
0412		



SI	CX	AH	Ip	FL
0601	04	32	040C	F1a7
0602	03	37	040E	F46
0603	02	33	040C	F907
0604	01	B3	0413	F103
0604	00	B3	0413	F103

agiven
on

0000

0400

0400

FF

0600

0600

22 ↓

0601

0601

37 ↓

0602

0602

1A ↓

0603

0603

1B3 ↓

Discussion :-

* what is means of this INS.

LoopNE ****

B
2/22

* why take $T=05$

~~Sol~~ check cx if Not Zero Jump to INS that we write it and multie him in step

* because the ~~IPs~~ between the beging of the loop and the end of the loop fine steep

0409
Loop

Test and
ans

0409

② F7A

③ F7A

0409

Lab 11

agiven

write a program to find the even No. in agiven string (6 byte) and store it in offset address (0800) the Data are : Them

beginning

Find the result in each
Loop AL, SI, DI, CX
[DI], IP, FI

0700	31
0701	76
0702	55
0703	A8
0705	3A

```

0400 MOV SI, 0700
0403 MOV DI, 0800
0406 MOV CX, 06
0409 MOV AL, [SI]
040B TEST AL, 01
040E JNZ 0403
0410 MOV [DI], AL
0412 INC DI
0413 INC SI
0414 LOOPNE 0409
0416 INT 3
  
```

Dec CX
JNZ 409

Test	0000	0001	01
(AND)	0011	0001	31
	0000	0001	odd

Sol 1 - CX

2 - because we put information in the reg

3 - if the value of AH Low Down of Value DI then Jump to address.

$$1_{\text{...}} + \frac{V_{\text{...}}}{\text{...}} + \frac{V_{\text{...}}}{\text{...}} + \frac{100}{\text{...}}$$

Discussion :

- * which reg. Deal with the LoopNE INstruction?
- * why use DI reg in the program?
- * what means of JB INstruction?

CF
24

AH	CX	DI	IP	FI
FF	05	0800	040C	F106
B7	04	0801	040C	F186
35	03	0802	040C	F102
22	02	0403	040E	F183
22	01	0405	040F	F107
22	00	0805	0411	F107

لايجاد البيانات في عنوان معين نتبع ما يلي :-

عنوان
F7 → D → add

AL	CX	SI	DI	IP	FL	DI
31	06	0700	0800	0413	F102	
76	05	0701	0800	0412	F140	
55	04	0702	0801	0413	F102	
A8	03	0703	0801	0412	F146	
27	02	0704	0802	0413	F102	
3A	01	0705	0802	0412	F142	

[DI]

76 A8 3A

A

4/5