

Microprocessor and Interfacing (Assignment No. 1)

Question-1:

Execute the assembly program for the following two arrays, add first 4 numbers of array1 with array2 and subtract 5th to 8th elements and place all data in array3 Array1: 1 2 5 9 6 8 7 3 Array2: 2 3 9 2 5 7 9 1 Array3: 0 0 0 0 0 0 0 0

Solution

SOURCE CODE:

```
org 100h

MOV BP, offset Arr1
MOV SI , offset Arr2
MOV DI, offset Arr3
MOV CX,4

L1:
MOV BL,[BP]
MOV AL,[SI]
ADD BL,AL
MOV [DI],BL
INC SI
INC DI
INC BP
loop L1
MOV CX,4

L2:
MOV BL,[BP]
```

MOV AL,[SI]

SUB BL,AL

MOV [DI],BL

INC SI

INC DI

INC B

loop L2

Arr1 DB 1,2,5,9,6,8,7,3

Arr2 DB 2,3,9,2,5,7,9,1

Arr3 DB 0,0,0,0,0,0,0,0

Ret

OUTPUT:

The screenshot shows a DOS emulator window titled "emulator: noname.com_". The top panel displays memory at address 0700:012B, showing a table of values. The bottom panel shows the emulator interface with registers, memory, and source code.

Registers:

Register	H	L
AX	00	00
BX	00	00
CX	00	04
DX	00	00
CS	0700	
IP	0106	
SS	0700	
SP	FFFE	
BP	0000	
SI	012B	
DI	0000	
DS	0700	
ES	0700	

Memory (0700:0106):

Address	Value
07100	B9 185
07101	04 004
07102	00 000 NULL
07103	BE 190
07104	2B 043
07105	01 001
07106	BF 191
07107	33 051
07108	01 001
07109	BD 189
0710A	3B 059
0710B	01 001
0710C	8A 138
0710D	04 004
0710E	8A 138
0710F	1D 029
07110	02 002
07111	C3 195
07112	88 136
07113	46 070 F
07114	00 000 NULL
07115	46 070 F
07116	47 071 G
07117	45 069 E
07118	E2 226

Source Code:

```
01 ; You may customize this and
02 ; The location of this templ
03
04
05 org 100h
06
07 MOV CX,4
08
09 MOV SI,offset Array1
10 MOV DI,offset Array2
11 MOV BP,offset Array3
12
13 L1:
14 MOV AL,[SI]
15 MOV BL,[DI]
16 ADD AL,BL
17 MOV [BP],AL
18 INC SI
19 INC DI
20 INC BP
21 loop L1
22
23 MOV CX,4
24
25 L2:
26 MOV AL,[SI]
27 MOV BL,[DI]
28 SUB AL,BL
29 MOV [BP],AL
30 INC SI
31 INC DI
32 INC BP
33 loop L2
34
```


0700:013B
update
table
list

0700:013B	03 05 0E 0B 01 01 FE 02-C3 90 90 90 90 90 90 90 90	00000000-00000000
0700:014B	90 90 90 90 90 90 90 90 90 90 90 90 90 90	00000000-00000000
0700:015B	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000
0700:016B	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000
0700:017B	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000
0700:018B	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000
0700:019B	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000
0700:01AD	00 00 00 00 00 00 00 00 00 00 00 00 00 00	00000000-00000000

emulator: noname.com_

file math debug view external virtual devices virtual drive help

Load
reload
step back
single step
run

step delay ms: 0

registers

	H	L
AX	00	02
BX	00	01
CX	00	00
DX	00	00
CS	0700	
IP	012B	
SP	0700	
SS	FFFE	
BP	0143	
SI	0133	
DI	013B	
DS	0700	
ES	0700	

0700:012B
0700:012B

07118:	E2	226	↑
07119:	F2	242	↑
0711A:	B9	185	↑
0711B:	04	004	↑
0711C:	00	000	NULL
0711D:	8A	138	↑
0711E:	04	004	↑
0711F:	8A	138	↑
07120:	1D	029	↑
07121:	2A	042	↑
07122:	C3	195	↑
07123:	88	136	↑
07124:	46	070	F
07125:	00	000	NULL
07126:	46	070	F
07127:	47	071	G
07128:	45	069	E
07129:	E2	226	↑
0712A:	F2	242	↑
0712B:	01	001	0
0712C:	02	002	0
0712D:	05	005	0
0712E:	09	009	TAB
0712F:	06	006	↑
07130:	08	008	BACK

MOV CX, 00004h
MOV SI, 0012Bh
MOV DI, 00133h
MOV BP, 0013Bh
MOV AL, [SI]
MOV BL, [DI]
ADD AL, BL
MOV [BP] + 00h, AL
INC SI
INC DI
INC BP
LOOP 010Ch
MOV CX, 00004h
MOV AL, [SI]
MOV BL, [DI]
SUB AL, BL
MOV [BP] + 00h, AL
INC SI
INC DI
INC BP
LOOP 011Dh
ADD [BP + SI], AX
ADD AX, 00609h
OR [BX], AL
...

original source code

```

02 ; You may customize this and
03 ; The location of this templ
04
05 org 100h
06
07 MOV CX,4
08
09 MOV SI,offset Array1
10 MOV DI,offset Array2
11 MOV BP,offset Array3
12
13 L1:
14 MOV AL,[SI]
15 MOV BL,[DI]
16 ADD AL,BL
17 MOV [BP],AL
18 INC SI
19 INC DI
20 INC BP
21 loop L1
22
23 MOV CX,4
24
25 L2:
26 MOV AL,[SI]
27 MOV BL,[DI]
28 SUB AL,BL
29 MOV [BP],AL
30 INC SI
31 INC DI
32 INC BP
33 loop L2
34
35 
```