

MIT LAB 8

SAHIL BONDRE: U18CO021

1. Write 8086 ALP for addition of two 8-bit numbers.

```
model small

.8086
.data
sum db ?
x db 20h
y db 30h

.code
mov ax,@data
mov ds,ax
mov al,x
add al,y
mov sum,al
mov ax,4c00h
int 21h
end
```

```
C:\SOURCE\TASM>DEBUG.EXE Q1.EXE
```

```
-u
```

076A:0000	B8B07	MOV	AX,076B
076A:0003	8ED8	MOV	DS,AX
076A:0005	A00500	MOV	AL,[0005]
076A:0008	02060600	ADD	AL,[0006]
076A:000C	A20400	MOV	[0004],AL
076A:000F	B8004C	MOV	AX,4C00
076A:0012	CD21	INT	21
076A:0014	0020	ADD	[BX+SI],AH
076A:0016	30833589	XOR	[BP+DI+8935],AL
076A:001A	1E	PUSH	DS
076A:001B	5A	POP	DX
076A:001C	0472	ADD	AL,72
076A:001E	05830E	ADD	AX,0E83

```
-g
```

```
Program terminated normally
```

Program terminated normally

-d 076B:0000

076B:0000	00 4C CD 21 50 20 30 83-35 89 1E 5A 04 72 05 83	.L.!P 0.5..Z.r..
076B:0010	0E 7A 04 02 BE 36 58 B4-00 8C 0E 50 34 E8 6C 35	.z...6X....P4.15
076B:0020	89 1E 5C 04 72 05 83 0E-7A 04 04 BE 42 58 B4 00	..\r...z...BX..
076B:0030	8C 0E 50 34 E8 55 35 89-1E 5E 04 72 05 83 0E 7A	..P4.U5..^..r...z
076B:0040	04 08 83 3E 7A 04 00 75-03 E9 AE 00 53 8B 1E 38	...>z..u....S..8
076B:0050	07 C4 7F 02 83 C7 18 3B-3F 76 09 E8 2B F8 5B 73;?o...+.ls
076B:0060	07 E8 EC F7 87 7F 02 5B-8B D7 A1 56 04 AB A1 74l...U...t
076B:0070	07 AB A1 84 07 AB A1 94-07 AB A1 AC 07 AB B8 01

2. Write 8086 ALP for subtraction of two 8-bit numbers

```
model small
.8086
.data
sum db ?
x db 54h
y db 14h

.code
mov ax, @data
mov ds, ax
mov al, x
sub al, y
mov sum, al
mov ax, 4c00h
int 21h
end
```

```
C:\SOURCE\TASM>DEBUG.EXE Q2.EXE
```

```
-u
```

```
076A:0000 B86B07      MOV     AX,076B
076A:0003 8ED8        MOV     DS,AX
076A:0005 A00500      MOV     AL,[0005]
076A:0008 2A060600    SUB     AL,[0006]
076A:000C A20400      MOV     [0004],AL
076A:000F B8004C      MOV     AX,4C00
076A:0012 CD21        INT     21
076A:0014 005414      ADD     [SI+14],DL
076A:0017 833589      XOR     WORD PTR [DI],-77
076A:001A 1E          PUSH    DS
076A:001B 5A          POP     DX
076A:001C 0472        ADD     AL,72
076A:001E 05830E      ADD     AX,0E83
```

```
Program terminated normally
```

```
-d 076B:0000
```

```
076B:0000 00 4C CD 21 40 54 14 83-35 89 1E 5A 04 72 05 83 .L.!@T..5..Z.r..
076B:0010 0E 7A 04 02 BE 36 58 B4-00 8C 0E 50 34 E8 6C 35 .z...6X....P4.15
076B:0020 89 1E 5C 04 72 05 83 0E-7A 04 04 BE 42 58 B4 00 ..\..r...z...BX..
076B:0030 8C 0E 50 34 E8 55 35 89-1E 5E 04 72 05 83 0E 7A ..P4.U5...^..r...z
076B:0040 04 08 83 3E 7A 04 00 75-03 E9 AE 00 53 8B 1E 38 ...>z..u....S..8
076B:0050 07 C4 7F 02 83 C7 18 3B-3F 76 09 E8 2B F8 5B 73 .....;?v...+.[s
076B:0060 07 E8 EC F7 87 7F 02 5B-8B D7 A1 56 04 AB A1 74 .....[...U...t
076B:0070 07 AB A1 84 07 AB A1 94-07 AB A1 AC 07 AB B8 01 .....

```

3. Write 8086 ALP for addition of two 16-bit numbers

```
model small
.8086
.data
sum db ?
sum1 db ?

.code
mov ax,@data
mov ds,ax
mov ah,12h
mov al,34h
mov bh,11h
mov bl,11h
add al,bl
adc ah,bh
mov sum,al
mov sum1,ah
mov ax,4c00h
int 21h
end
```

```

C:\SOURCE\TASM>DEBUG.EXE Q3.EXE
-u
076A:0000 B86B07      MOV     AX,076B
076A:0003 8ED8        MOV     DS,AX
076A:0005 B412        MOV     AH,12
076A:0007 B034        MOV     AL,34
076A:0009 B711        MOV     BH,11
076A:000B B311        MOV     BL,11
076A:000D 02C3        ADD     AL,BL
076A:000F 12E7        ADC     AH,BH
076A:0011 A20E00      MOV     [000E],AL
076A:0014 88260F00    MOV     [000F],AH
076A:0018 B8004C      MOV     AX,4C00
076A:001B CD21        INT     21
076A:001D 7205        JB      0024
076A:001F 830E7A0402    OR      WORD PTR [047A],+02
-g

```

```

Program terminated normally
-d 076B:0000
076B:0000 E7 A2 0E 00 88 26 0F 00-B8 00 4C CD 21 72 45 23  ....&....L.†rE#
076B:0010 0E 7A 04 02 BE 36 58 B4-00 8C 0E 50 34 E8 6C 35  .z...6X....P4.15
076B:0020 89 1E 5C 04 72 05 83 0E-7A 04 04 BE 42 58 B4 00  ..\..r...z...BX..
076B:0030 8C 0E 50 34 E8 55 35 89-1E 5E 04 72 05 83 0E 7A  ..P4.U5...^..r...z
076B:0040 04 08 83 3E 7A 04 00 75-03 E9 AE 00 53 8B 1E 38  ...>z..u....S..8
076B:0050 07 C4 7F 02 83 C7 18 3B-3F 76 09 E8 2B F8 5B 73  .......;?v...+.[s
076B:0060 07 E8 EC F7 87 7F 02 5B-8B D7 A1 56 04 AB A1 74  .......[...U...t
076B:0070 07 AB A1 84 07 AB A1 94-07 AB A1 AC 07 AB B8 01  ....

```

Answer in 000E and 000F

4. Write 8086 ALP for subtraction of two 16-bit numbers

```

model small
.8086
.data
sum db ?
sum1 db ?

.code
mov ax, @data
mov ds, ax
mov ah, 12h
mov al, 34h
mov bh, 11h
mov bl, 11h
sub al, bl
sbb ah, bh
mov sum, al
mov sum1, ah

```

```

mov ax, 4c00h
int 21h
end

```

```

C:\SOURCE\TASM>DEBUG.EXE Q4.EXE
-u
076A:0000 B86B07      MOV     AX,076B
076A:0003 8ED8             MOV     DS,AX
076A:0005 B412             MOV     AH,12
076A:0007 B034             MOV     AL,34
076A:0009 B711             MOV     BH,11
076A:000B B311             MOV     BL,11
076A:000D 2AC3             SUB     AL,BL
076A:000F 1AE7             SBB     AH,BH
076A:0011 A20E00          MOV     [000E],AL
076A:0014 88260F00          MOV     [000F],AH
076A:0018 B8004C          MOV     AX,4C00
076A:001B CD21             INT     21
076A:001D 7245             JB      0064
076A:001F 230E7A04          AND     CX,[047A]

```

```

-d 076B:0000
076B:0000 E7 A2 0E 00 88 26 0F 00-B8 00 4C CD 21 72 23 01  ....&....L.!r#.
076B:0010 0E 7A 04 02 BE 36 58 B4-00 8C 0E 50 34 E8 6C 35  .z...6X....P4.15
076B:0020 89 1E 5C 04 72 05 83 0E-7A 04 04 BE 42 58 B4 00  ..\..r...z...BX..
076B:0030 8C 0E 50 34 E8 55 35 89-1E 5E 04 72 05 83 0E 7A  ..P4.U5...^..r...z
076B:0040 04 08 83 3E 7A 04 00 75-03 E9 AE 00 53 8B 1E 38  ...>z..u....S..8
076B:0050 07 C4 7F 02 83 C7 18 3B-3F 76 09 E8 2B F8 5B 73  .......;?v...+.ls
076B:0060 07 E8 EC F7 87 7F 02 5B-8B D7 A1 56 04 AB A1 74  .......[...U...t
076B:0070 07 AB A1 84 07 AB A1 94-07 AB A1 AC 07 AB B8 01  ....

```

Answer in 000E and 000F