



VIT[®]
Vellore Institute of Technology
(Deemed to be University under section 3 of UGC Act, 1956)

Computer Science & Engineering

CSE2006

Microprocessor and Interfacing

LAB ASSIGNMENT 5

Submitted to **Prof. SANJAY R**

TOPIC: ASSEMBLY LANGUAGE PROGRAMMING

NAME: PUNIT MIDDHA

REG.NO: 19BCE2060

SLOT: L43+L44

DATE: 24/11/2021

1. Traffic Light Control System

Aim:

To load different control words in traffic light system in emu8086

Handwritten Program:

1.

```
; NAME: PUNIT MIDDHA
```

```
; REGNo: 19 BCE2060
```

```
# start = Traffic_Lights.exe#
```

```
name "traffic"
```

```
mov ax, all_red
```

```
out 4, ax
```

```
mov si, offset situation
```

```
next:
```

```
mov ax, [si]
```

```
out 4, ax
```

```
; wait 5 seconds < 5 million microseconds >
```

```
mov cx, 4ch ; 004C4B40h = 5,000,000
```

```
mov dx, 4B40h
```

```
mov ah, 86h
```

```
int 15h
```

```
add si, 2 ; Next situation
```

```
cmp si, sit_end
```

```
jb next
```

```
mov si, offset situation
```

```
jmp next
```

```
;
```

```
Situation dw
```

```
S1 dw
```

```
S2 dw
```

```
S3 dw
```

```
S4 dw
```

```
sit_end = $
```

```
all_red equ
```

```
FEDC-BAG8-7654-3210
0000-1010-0010-0101b
0000-0000-1000-0011b
0000-0010-1011-0101b
0000-1010-0001-1001b
0000-0000-0111-1100b
```



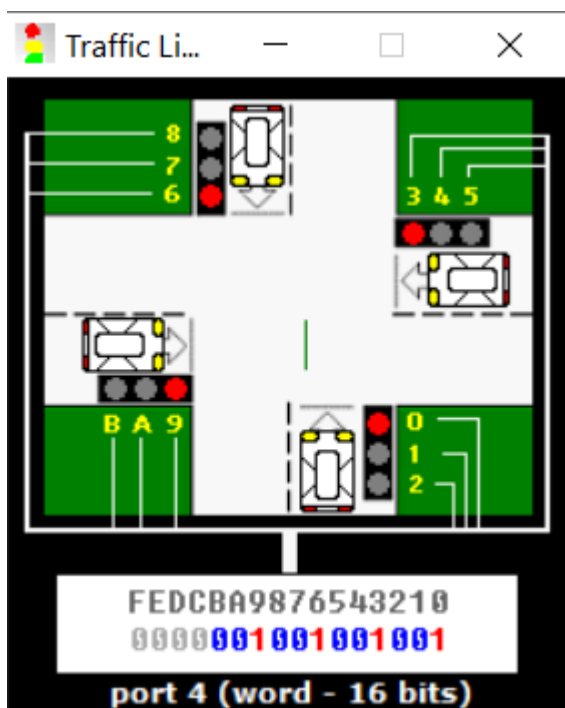
Snapshots of typed program and Output:

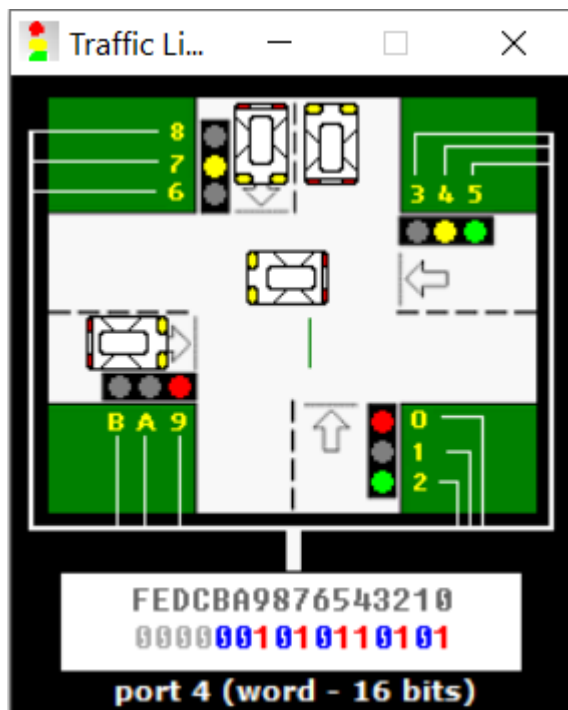
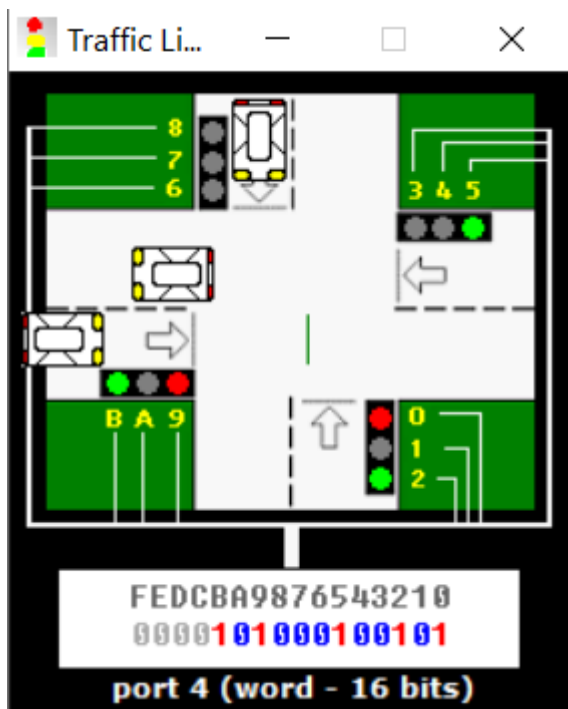
edit: C:\emu8086\examples\traffic_lights.asm

file edit bookmarks assembler emulator math ascii codes help

```
new open examples save compile emulate calculator convertor options help about

01 ; controlling external device with 8086 microprocessor.
02 ; realistic test for c:\emu8086\devices\Traffic_Lights.exe
03
04 ; NAME: PUNIT MIDDHA
05 ; REGNO: 19BCE2060
06
07 #start=Traffic_Lights.exe#
08
09 name "traffic"
10
11
12 mov ax, all_red
13 out 4, ax
14
15
16 mov si, offset situation
17
18
19 next:
20 mov ax, [si]
21 out 4, ax
22
23 ; wait 5 seconds (5 million microseconds)
24 mov cx, 4Ch ; 004C4B40h = 5,000,000
25 mov dx, 4B40h
26 mov ah, 86h
27 int 15h
28
29
30 add si, 2 ; next situation
31 cmp si, sit_end
32 jb next
33 mov si, offset situation
34 jmp next
35
36
37 ;
38 situation dw FEDC_BA98_7654_3210
39 s1 dw 0000_1010_0010_0101b
40 s2 dw 0000_0000_1000_0011b
41 s3 dw 0000_0010_1011_0101b
42 s4 dw 0000_1010_0001_1001b
43 sit_end = $
44
45
46 all_red equ 0000_0010_0100_1001b
```





Inference:

- Output 1:
In the first output i.e., all red, the control word is:
0000001001001001
001 means red light.

So, we can see in the output red light on in all traffic lights.

- Output 2:

In the second output the control word is:

0000101000100101

101 means red and green light are on

000 means no light is on

100 means red light is on

101 means red and green lights are on

This can be seen in the output.

- Output 3:

In the last output the control word is:

0000001010110101

001 means red light is on

010 means yellow light is on

110 means green and yellow lights are on

101 means red and green lights are on

This can be seen in the output.