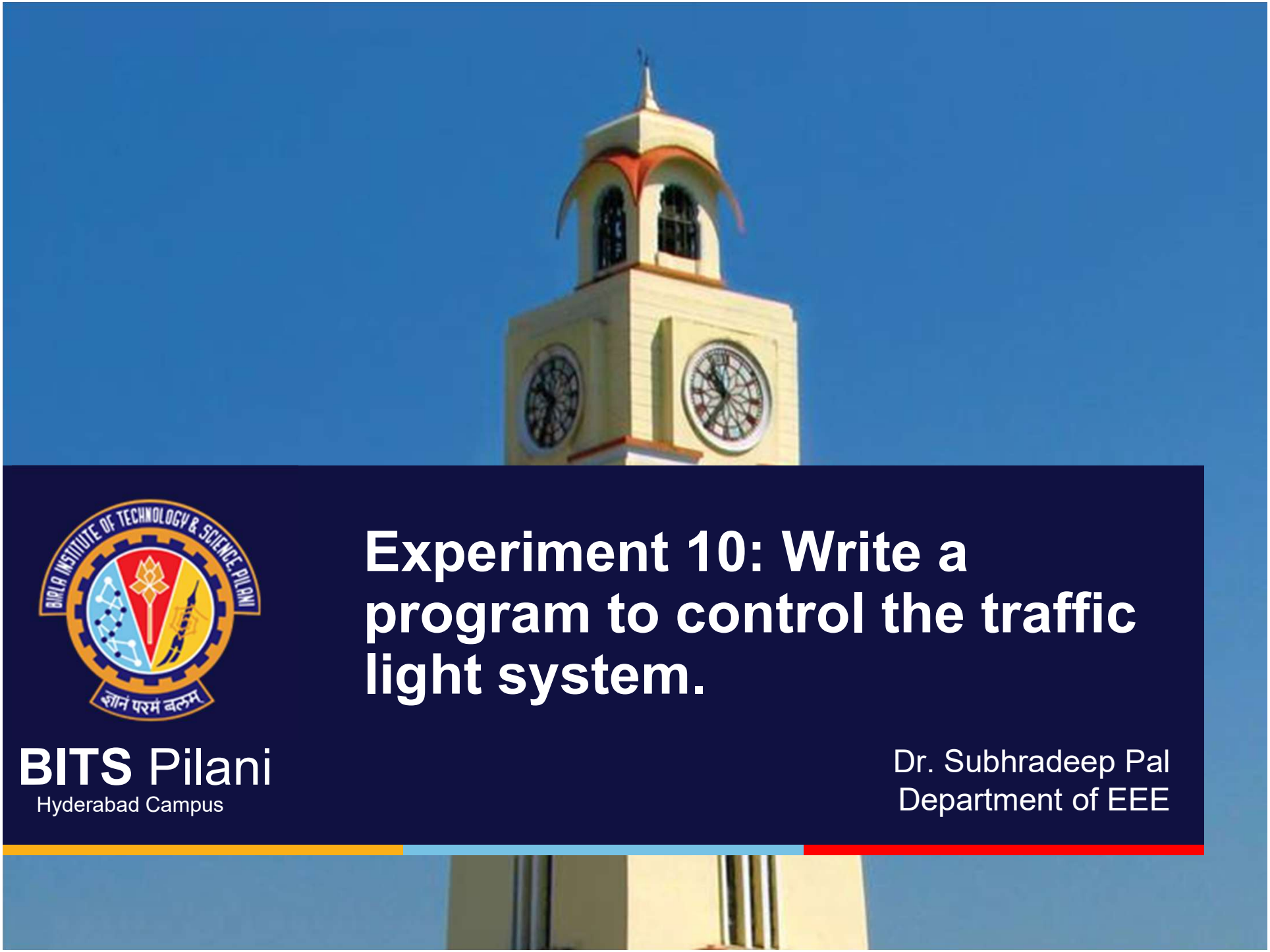


Experiment 10: Write a program to control the traffic light system.

BITS Pilani
Hyderabad Campus

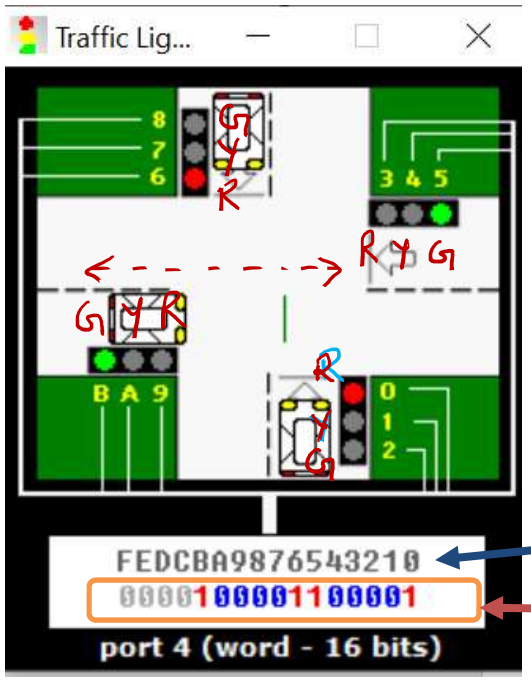
Dr. Subhradeep Pal
Department of EEE

ALPs to be completed

1. Write a program to control the traffic light system using 8086.

Traffic Light in Emulator

- A virtual traffic light system is available in EMU8086 with port address 4.
- It consists of 12 LEDs with an animation of car moving in direction where green LED is activated.
- Control word can be provided using 8086 to change the signal color.



State of each LED

1: LED is turned on
0: LED is turned off

Control Word Format

Current Control Word

Traffic Light in Emulator

- Port Address: 04H
- Instruction to include stepper motor in emulator:
  #start=traffic_lights.exe#
- How to load the control word in traffic light?

  Use OUT instruction. For out instruction select port 4. Use AX register to store control word intermediately.

- How to maintain time gap between two control words?

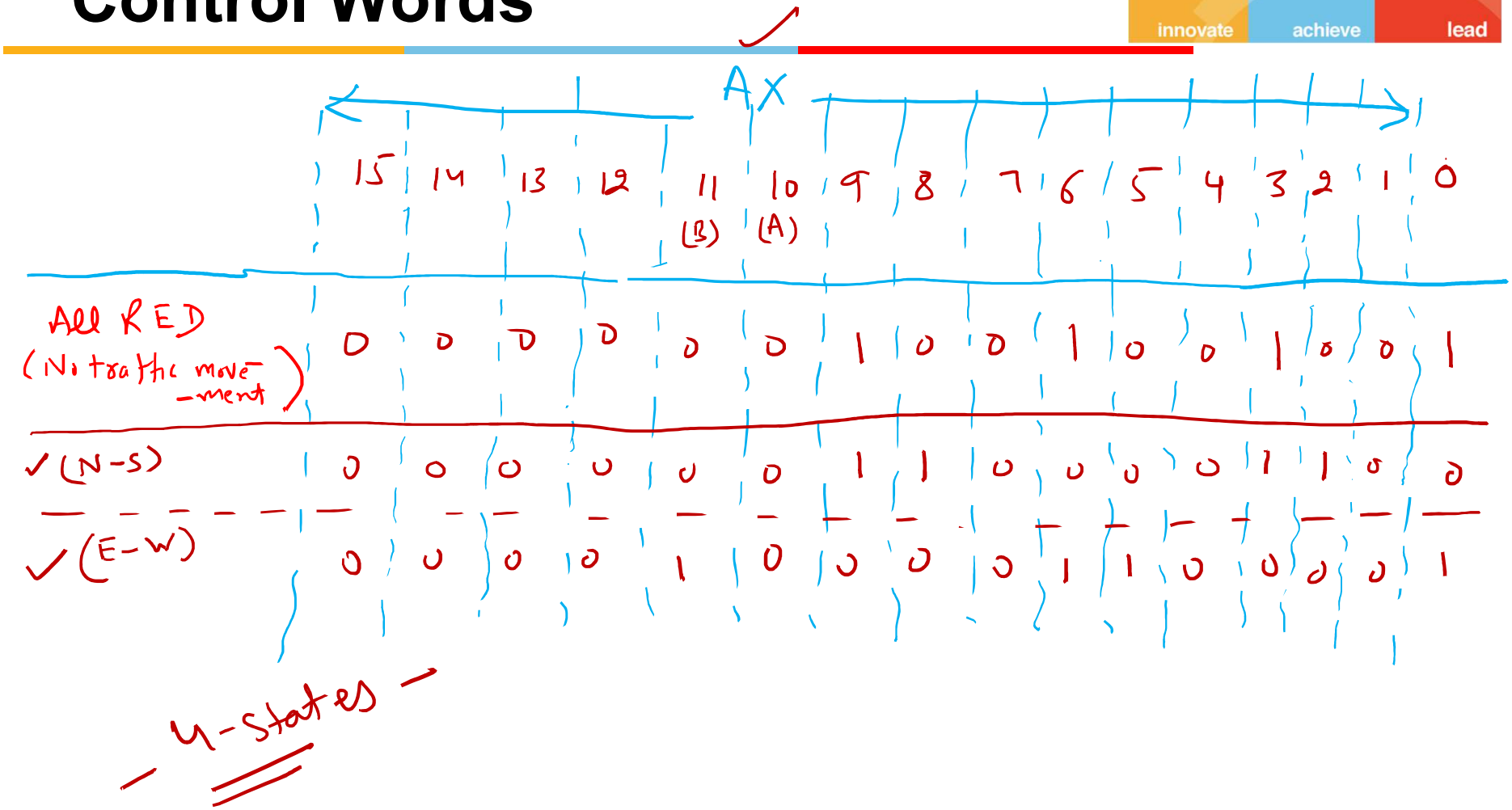
  Use CX-DX registers for wait time. Note: Unit of time for 8086 is μs.

Control Words

F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0	
×	×	×	×	1	0	0	0	0	1	1	0	0	0	0	1	Default
×	×	×	×	0	0	1	0	0	1	0	0	1	0	0	1	All red
×	×	×	×	0	0	1	1	0	0	0	0	1	1	0	0	
×	×	×	×	0	1	1	0	1	0	0	1	1	0	1	0	
×	×	×	×	1	0	0	0	0	1	1	0	0	0	0	1	
×	×	×	×	1	0	0	0	0	1	1	0	0	0	0	1	
×	×	×	×	0	1	0	0	1	1	0	1	0	0	1	1	

×: Not used bits (should be replaced by 0).

Control Words



ALP for Traffic Signal

```
#start=Traffic_Lights.exe#

name "traffic"

mov ax, all_red
out 4, ax

mov si, offset s1
next:
      mov ax, [si]
      out 4, ax

mov     cx, xxxxh
mov     dx, xxxxh

mov     ah, 86h
int     15h

add si, 2
cmp si, sit_end
jb  next
mov si, offset s1
jmp next

s1          dw      xxxx_xxxx_xxxx_xxxxb
s2          dw      xxxx_xxxx_xxxx_xxxxb
s3          dw      xxxx_xxxx_xxxx_xxxxb
s4          dw      xxxx_xxxx_xxxx_xxxxb
sit_end = $
all_red     equ     0000_0010_0100_1001b
```

Initializes the virtual traffic signal

Closes all traffic

Offset calculation of CW and load in SI.

Loads the first data in AX and send it IO

Use CX-DX register to provide wait instruction for 5s.

BIOS Delay Function. Unit in µs. Content Format: CX-DX

Increase SI for next data

Check all situational data are emulated or not. If yes, then restart else complete all the situational control words.

Control world (16-bit format)

ALP for Traffic Signal

```
#start=Traffic_Lights.exe#

name "traffic"

mov ax, all_red
out 4, ax

mov si, offset s1
next:
    mov ax, [si]
    out 4, ax

    mov   cx, xxxxh
    mov   dx, xxxxh

    mov   ah, 86h
    int   15h

    add si, 2
    cmp si, sit_end
    jb  next
    mov si, offset s1
    jmp next

s1          dw    xxxx_xxxx_xxxx_xxxxb
s2          dw    xxxx_xxxx_xxxx_xxxxb
s3          dw    xxxx_xxxx_xxxx_xxxxb
s4          dw    xxxx_xxxx_xxxx_xxxxb
sit_end = $
all_red     equ   0000_0010_0100_1001b
```

Wait Time = 5 s = 5 × 10^6 µs
Hexadecimal Value = 4C4B40
CX = 004Ch
DX = 4B40h

```
s1    dw    0000_0011_0000_1100b
s2    dw    0000_0110_1001_1010b
s3    dw    0000_1000_0110_0001b
s4    dw    0000_0100_1101_0011b
```

Review Questions → Assignment

- What should be the value of CX-DX register for wait time of 2s between two control instructions? Run the program in emulator and demonstrate.