

CSLR61 : EMBEDDED SYSTEMS

LAB-2

Roll no. : 106119100

Name : Rajneesh Pandey

Section : CSE-B

1. Switch between hex counter and decade counter using switches. Display the current value of the counter with the help of 4 LEDs and the screen.

•Libraries Used: InterruptIn, BusOut

```
#include "mbed.h"
BusOut leds(LED1, LED2, LED3, LED4);
InterruptIn swt(p5);

void hexCounter()
{
    while (1)
    {
        for (int i = 0; i < 16; i++)
        {
            leds = i;
            printf("Hex count : %d\n", i);
            wait(1);
        }
    }
}

void decCounter()
{
    while (1)
    {
        for (int i = 0; i < 10; i++)
        {
            leds = i;
            printf("Dec count : %d\n", i);
            wait(1);
        }
    }
}

int main()
{
    swt.rise(&hexCounter);
    swt.fall(&decCounter);
}
```

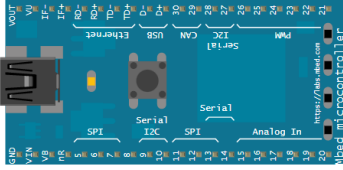
Output :

simulator.mbed.com/#user_1644227344419

arm MBED

Blinky with BusOut Load demo Run + Add component

```
1 #include "mbed.h"
2 BusOut leds(LED1, LED2, LED3, LED4);
3 InterruptIn swt(p5);
4
5
6 void hexCounter(){
7     while(1){
8         for(int i = 0; i < 16; i++) {
9             leds = i;
10            printf("Hex count : %d\n",i);
11            wait(1);
12        }
13    }
14 }
15 void decCounter(){
16     while(1){
17         for(int i = 0; i < 10; i++) {
18             leds = i;
19             printf("Dec count : %d\n",i);
20             wait(1);
21         }
22     }
23 }
24 int main() {
25     swt.rise(&hexCounter);
26     swt.fall(&decCounter);
27 }
```



Switch (p5)

Serial output

```
Hex count : 1
Hex count : 2
Hex count : 3
Dec count : 0
Dec count : 1
Hex count : 0
Hex count : 1
Hex count : 2
Hex count : 3
```

2. Display the octal counter using the onboard LEDs using Ticker Object. After 10 second, reset all the values of the onboard LED using Timeout Object.

•Libraries Used: TimeOut, Ticker

```
#include "mbed.h"

BusOut leds(LED1, LED2, LED3, LED4);
Ticker tck;
Timeout to;

int octCount = 0;

void octalCounter()
{
    leds = octCount;
    printf("counter  %d\n", leds.read());
    octCount = (octCount + 1) % 8;
}

void reset()
{
    printf("resetting...");
    octCount = 0;
    leds = 0;
    printf("Value after resetting  %d\n", leds.read());
}
```

```

int main()
{
    tck.attach(&octalCounter, 2);
    to.attach(&reset, 10);
}

```

simulator.mbed.com/#user_1644228553682

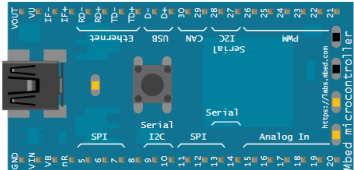
arm MBED

Blinky Load demo Run + Add component

```

1 #include "mbed.h"
2
3 BusOut leds(LED1, LED2, LED3, LED4);
4 Ticker tck;
5 Timeout to;
6
7 int octCount = 0;
8
9 void octalCounter(){
10     leds = octCount;
11     printf("counter %d\n", leds.read());
12     octCount = (octCount+1)%8;
13 }
14
15 void reset(){
16     printf("resetting...\n");
17     octCount = 0;
18     leds = 0;
19     printf("Value after resetting %d\n", leds.read());
20 }
21
22 int main() {
23     tck.attach(&octalCounter,2);
24     to.attach(&reset,10);
25 }
26

```



Serial output

```

counter 0
counter 1
counter 2
counter 3
counter 4
resetting...Value after resetting 0
counter 0
counter 1
counter 2
counter 3

```

3. Display the odd counter using the onboard LEDs and screen; After 10 seconds, reset all the values of the onboard LED and display even counter using Timeout Object. Give switch option to manually toggle between two counters.

Libraries Used: InterruptIn, Ticker, TimeOut, BusOut

```
#include "mbed.h"

BusOut leds(LED1, LED2, LED3, LED4);
Ticker tck;
Timeout to;
InterruptIn swt(p5);

int state = 1;
int odd = 1;

void Counter(){
    while(1){
        for(int i = 0; i < 16; i++){
            if (i % 2 == state){
                leds = i;
                if (state)
                    printf("Odd Counter %d\n", leds.read());
                else{
                    printf("Even Counter %d\n", leds.read());
                }
                wait(1);
            }
        }
    }
}
```

```

void Toggle(){
    state = !state;
    printf("Toggle using switch...\n");
    Counter();
}

void reset(){
    printf("resetting Odd Counter...\n");
    state = 0;
    leds = 0;
    tck.detach();
    printf("Starting Even Counter...\n");
    Counter();
}

void oddCounter(){
    leds = odd;
    odd = (odd+2)%16;
    printf("Odd Counter %d\n", leds.read());
}

int main()
{
    tck.attach(&oddCounter, 2.0);
    to.attach(&reset, 10);
    swt.rise(&Toggle);
}

```

Blinky ▼ [Load demo](#)

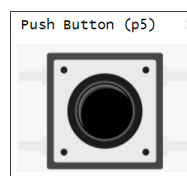
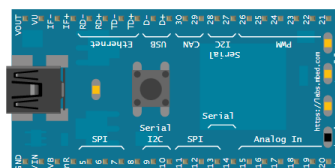
Run

+ Add component

```

1  #include "mbed.h"
2
3  BusOut leds(LED1, LED2, LED3, LED4);
4  Ticker tck;
5  Timeout to;
6  InterruptIn swt(p5);
7
8  int state = 1;
9  int odd = 1;
10
11 void Counter(){
12     while(1){
13         for(int i = 0; i < 16; i++){
14             if (i % 2 == state){
15                 leds = i;
16                 if (state)
17                     printf("Odd Counter %d\n", leds.read());
18                 else
19                     printf("Even Counter %d\n", leds.read());
20             }
21             wait(1);
22         }
23     }
24 }
25
26
27 void Toggle(){
28     state = !state;
29     printf("Toggle using switch...\n");
30     Counter();
31 }
32
33 void reset(){
34     printf("resetting Odd Counter...\n");
35     state = 0;

```



Serial output

```
Odd Counter 9
resetting Odd Counter...
Starting Even Counter...
Even Counter 0
Even Counter 2
Even Counter 4
Even Counter 6
Even Counter 8
Even Counter 10
Even Counter 12
Even Counter 14
Even Counter 0
Toggle using switch...
Odd Counter 1
```