

# **CSLR61 : EMBEDDED SYSTEMS**

## **LAB-1**

**Roll no. : 106119100**

**Name : Rajneesh Pandey**

**Section : CSE-B**

1. Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.

```
/* 106119100 Rajneesh Pandey */
#include "mbed.h"

DigitalOut led1(LED1);
DigitalOut led2(LED2);
DigitalOut led3(LED3);
DigitalOut led4(LED4);

int main(){
    while (1){
        led1 = !led1;
        printf("Blink! LED1 is now %d\n", led1.read());
        led3 = !led3;
        printf("Blink! LED3 is now %d\n", led3.read());
        wait_ms(1000);
        led2 = !led2;
        printf("Blink! LED2 is now %d\n", led2.read());
        led4 = !led4;
        printf("Blink! LED4 is now %d\n", led4.read());
    }
}
```

arm MBED

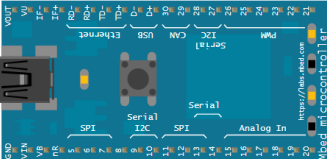
Blinky

Load demo

Run

+ Add component

```
1 /* 106119100 Rajneesh Pandey */
2
3 #include "mbed.h"
4
5 DigitalOut led1(LED1);
6 DigitalOut led2(LED2);
7 DigitalOut led3(LED3);
8 DigitalOut led4(LED4);
9
10 int main() {
11     while (1) {
12         led1 = !led1;
13         printf("Blink! LED1 is now %d\n", led1.read());
14         led3 = !led3;
15         printf("Blink! LED3 is now %d\n", led3.read());
16         wait_ms(1000);
17         led2 = !led2;
18         printf("Blink! LED2 is now %d\n", led2.read());
19         led4 = !led4;
20         printf("Blink! LED4 is now %d\n", led4.read());
21     }
22 }
```



Serial output

```
Blink! LED4 is now 0
Blink! LED1 is now 1
Blink! LED3 is now 1
Blink! LED2 is now 1
Blink! LED4 is now 1
Blink! LED1 is now 0
Blink! LED3 is now 0
Blink! LED2 is now 0
Blink! LED4 is now 0
Blink! LED1 is now 1
Blink! LED3 is now 1
Blink! LED2 is now 1
Blink! LED4 is now 1
Blink! LED1 is now 0
```

2. Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.

```
3. /* 106119100 Rajneesh Pandey*/
4. #include "mbed.h"
5.
6. BusOut myleds(LED1, LED2, LED3, LED4);
7.
8. int main(){
9.     while (1){
10.         for (int i = 1; i < 16; i++){
11.             myleds = i;
12.             printf("%d ", i);
13.             if (i & 1){
14.                 printf("odd count\n");
15.                 wait(1);
16.             }
17.             else{
18.                 printf("even count\n");
19.                 wait(2);
20.             }
21.         }
22.     }
23. }
```

arm MBED

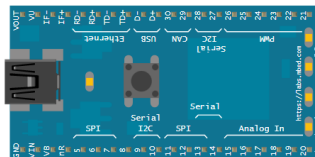
Blinky with BusOut

Load demo

Run

+ Add component

```
1  /* 106119100 Rajneesh Pandey*/
2  #include "mbed.h"
3
4  BusOut myleds(LED1, LED2, LED3, LED4);
5
6  int main(){
7      while(1){
8          for(int i = 1; i < 16; i++){
9              myleds = i;
10             printf("%d ", i);
11             if(i&1){
12                 printf("odd count\n");
13                 wait(1);
14             }
15             else{
16                 printf("even count\n");
17                 wait(2);
18             }
19         }
20     }
21 }
22
```



Serial output

```
1 odd count
2 even count
3 odd count
4 even count
5 odd count
6 even count
7 odd count
8 even count
9 odd count
10 even count
11 odd count
12 even count
13 odd count
14 even count
```

### 3. Blink LEDs – for all composite number below 15.

```
4. /* 106119100 Rajneesh Pandey */
5. #include "mbed.h"
6.
7. BusOut myleds(LED1, LED2, LED3, LED4);
8.
9. bool isComposite(int n)
10. {
11.     if (n == 1)
12.         return true;
13.     for (int i = 2; i < n; i++)
14.     {
15.         if (n % i == 0)
16.             return true;
17.     }
18.     return false;
19. }
20. int main()
21. {
22.     while (1)
23.     {
24.         for (int i = 1; i < 16; i++)
25.         {
26.             if (isComposite(i))
27.             {
28.                 myleds = i;
29.                 printf("%d is Composite Number\n", i);
30.                 wait(1);
31.             }
32.         }
33.     }
34. }
```

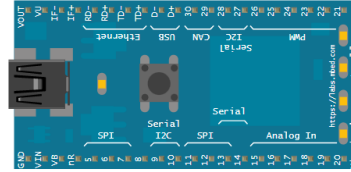
Blinky with BusOut

Load demo

Run

+ Add component

```
1 #include "mbed.h"
2
3 BusOut myleds(LED1, LED2, LED3, LED4);
4
5 bool isComposite(int n){
6     if(n==1) return true;
7     for(int i=2;i<n;i++){
8         if(n%i==0) return true;
9     }
10    return false;
11 }
12
13 int main() {
14     while(1) {
15         for(int i = 1; i < 16; i++) {
16             if(isComposite(i)){
17                 myleds = i;
18                 printf("%d is Composite Number\n",i);
19                 wait(1);
20             }
21         }
22     }
23 }
```



## Serial output

```
1 is Composite Number
4 is Composite Number
6 is Composite Number
8 is Composite Number
9 is Composite Number
10 is Composite Number
12 is Composite Number
14 is Composite Number
15 is Composite Number
1 is Composite Number
4 is Composite Number
6 is Composite Number
8 is Composite Number
9 is Composite Number
```

## 4. Blink LEDs – to count even numbers

```
5. #include "mbed.h"
6.
7. BusOut leds(LED1, LED2, LED3, LED4);
8.
9. int main(){
10.     while (1){
11.         for(int i = 0; i < 16; i++){
12.             if(i%2==0){
13.                 leds = i;
14.                 printf("count %d\n", leds.read());
15.                 wait(2);
16.             }
17.         }
18.     }
19. }
20.
```

The screenshot displays the ARM Mbed simulator interface. The top bar shows the URL `simulator.mbed.com/#user_1644232012069`. The main area is divided into three sections:

- Left Panel:** Contains the source code for a program named "Blinky". The code is identical to the one shown in the previous block, with line numbers 1 through 16.
- Right Panel (Top):** Shows a schematic diagram of an mbed microcontroller board with various components like LEDs, buttons, and connectors labeled.
- Right Panel (Bottom):** A "Serial output" window showing the execution results. It displays a list of "count" values: 0, 2, 4, 6, 8, 10, 12, 14, 0, 2. The values are printed on separate lines, with the last line being "count 2".