

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | |
|  | CSLR61 : EMBEDDED SYSTEMS  **LAB-2** | | | | | |  |
|  |  | | | | |  | |
|  | | |  |  | | | |
|  | | | Roll no. : 106119100Name : Rajneesh PandeySection : 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 :

Graphical user interface

Description automatically generated

1. 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);

}

Graphical user interface, text

Description automatically generated

1. 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);

}

Graphical user interface, text

Description automatically generated