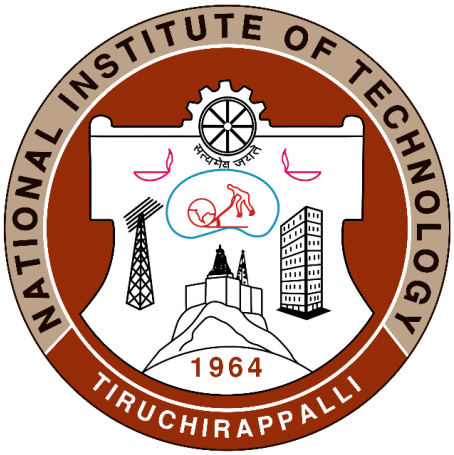
CSLR61 - Embedded Systems Lab



Lab Report

Name: **Rajneesh Pandey**

Roll No.: **106119100**

Class: **CSE-B**

Date: 02/05/2022

	on, the intensity of the sound should be increasing and the upon switching off, should make the intensity of the sound decreasing.			
14	Design a program to make ambulance sound.			
15	Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.	07-03-2022	5	23
16	Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.			
17	Blink LEDs – for all composite number below			
18	Blink LEDs – to count even numbers			
19	Interface an ultrasonic sensor and display the distance measured in a screen.	28-03-2022	6	29
20	Interface a PIR sensor with Arduino and blink a led if any movement is captured by the sensor.			
21	Interface two 7-segment display with Arduino Uno board. a)odd counter and b)even counter	11-04-2022	7	35
22	Interface the ambient light sensor with Arduino Uno board. Check the light value from the sensor and switch on/off the bulb.			
23	Interface the temperature and gas sensor with Arduino Uno board. Check the temperature and the gas value.			
24	Interface force sensor with Arduino board and display the amount of force given as input to sensor in LCD screen.	18-04-2022	8	43
25	Interface the keypad and tilt sensor with Arduino board, if the sensor is being tilted, then take input from keypad and print it in the LCD.			

Questions

Lab 1 Date: 31-01-2022

1. Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.
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. Blink LEDs – for all composite number below 15
4. Blink LEDs – to count even numbers

Lab 2 Date: 07-02-2022

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.
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.
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.

Lab 3 Date: 14-02-2022

1. Increase the intensity of one LED while decreasing the other LED's intensity at the same unit. Display the intensity value of the LEDs.
2. Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED.
3. Interface switch, 2 LEDs, potentiometer with the mbed board. If the switch is on, control the intensity of the LEDs using potentiometer (one LED should be increasing and other should be decreasing at the rate given via potentiometer). If the switch is off, blink LEDs one at increasing rate and other at decreasing rate (same rate) and loop back.

Lab 4 Date: 28-02-2022

1. Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker.
2. Design an alarm system, which makes double beep sound for every 5s using a speaker.
3. Have a switch, 2 LEDs and a speaker interfaced with mbed board. Upon switching on, the intensity of the sound should be increasing and the upon switching off, should make the intensity of the sound decreasing. Proportionally increase or decrease the intensity of two LEDs.
4. Design a program to make ambulance sound.

Lab 5 Date: 07-03-2022

1. Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.
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. Blink LEDs – for all composite number below 15

4. Blink LEDs – to count even numbers

Lab 6 Date: 28-03-2022

1. Interface an ultrasonic sensor and display the distance measured in a screen and glow red, blue, and green led if the distance is close, moderate, and far respectively. Play different tones for different cases.
2. Interface a PIR sensor with Arduino and blink a led if any movement is captured by the sensor and count the number of movements. Make a buzzer when movement detected.

Lab 7 Date: 11-04-2022

1. Interface two 7-segment display with Arduino Uno board.
2. Implement the a)odd counter and b)even counter and display the values in the seven-segment display.
3. Interface the ambient light sensor with Arduino Uno board. Check the light value from the sensor and switch on/off the bulb (based on the threshold value).
4. Interface the temperature and gas sensor with Arduino Uno board. Check the temperature and the gas value, if the limit is beyond the threshold, switch on the bulb and make alarm using buzzer.

Lab 8 Date: 18-04-2022

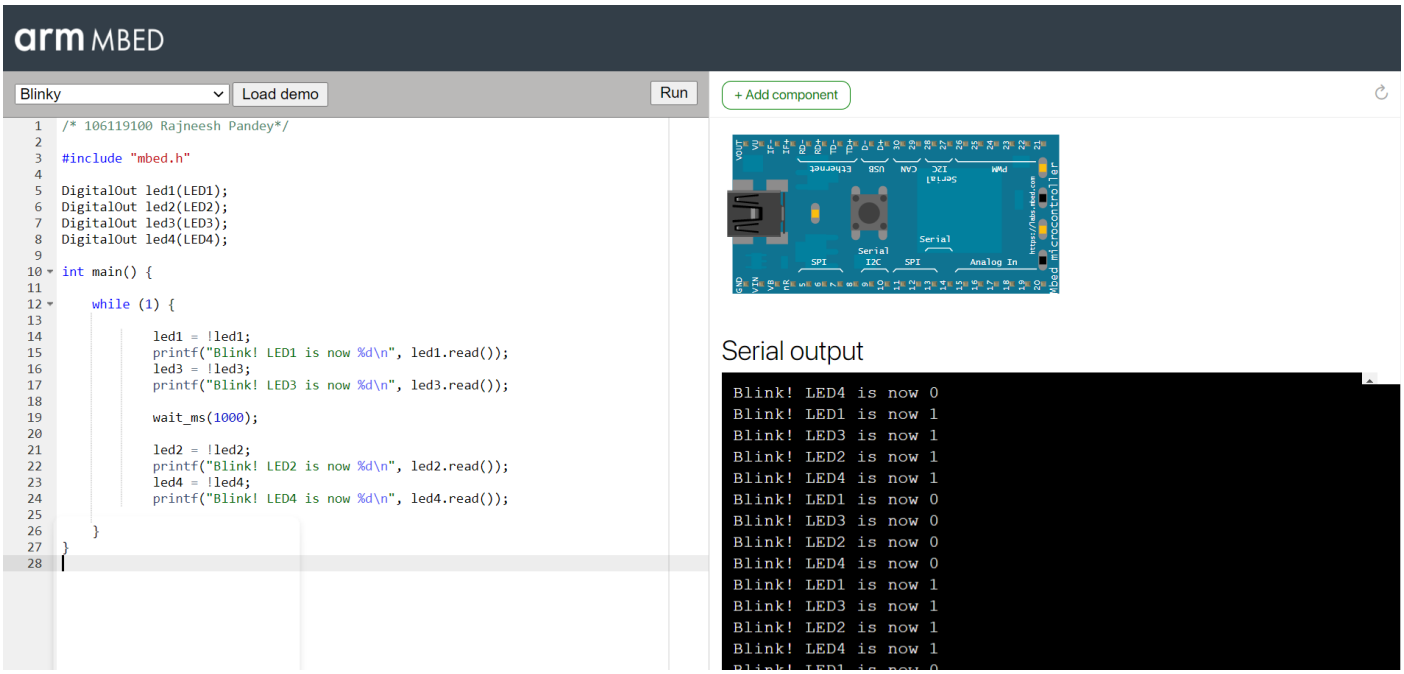
1. Interface force sensor with Arduino board and display the amount of force given as input to sensor in LCD screen. Also, interface LED RGB and change colour of the LED based on some threshold Values.
2. Interface the keypad and tilt sensor with Arduino board, if the sensor is being tilted, then take input from keypad and print it in the LCD

Lab 1

Date: 31-01-2022

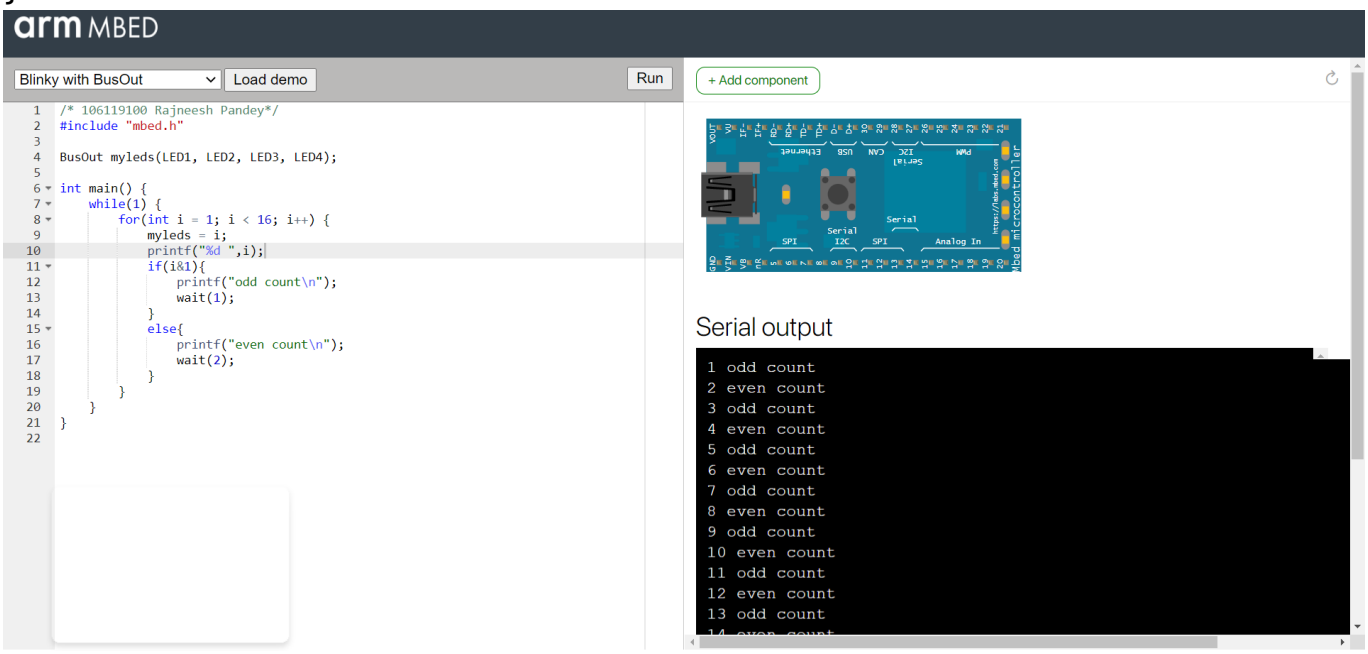
1. Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.

```
#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());
    }
}
```



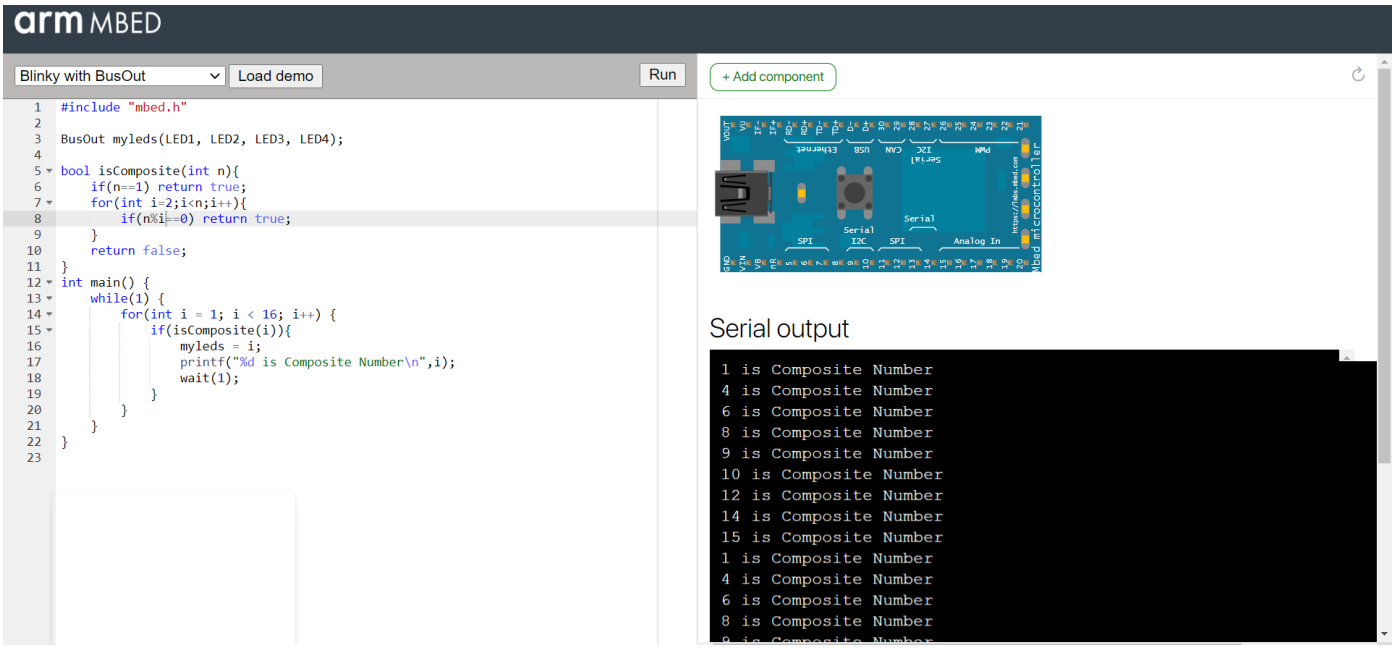
2. Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec

```
#include "mbed.h"

BusOut myleds(LED1, LED2, LED3, LED4);

int main(){
    while (1){
        for (int i = 1; i < 16; i++){
            myleds = i;
            printf("%d ", i);
            if (i & 1){
                printf("odd count\n");
                wait(1);
            }
            else{
                printf("even count\n");
                wait(2);
            }
        }
    }
}
```



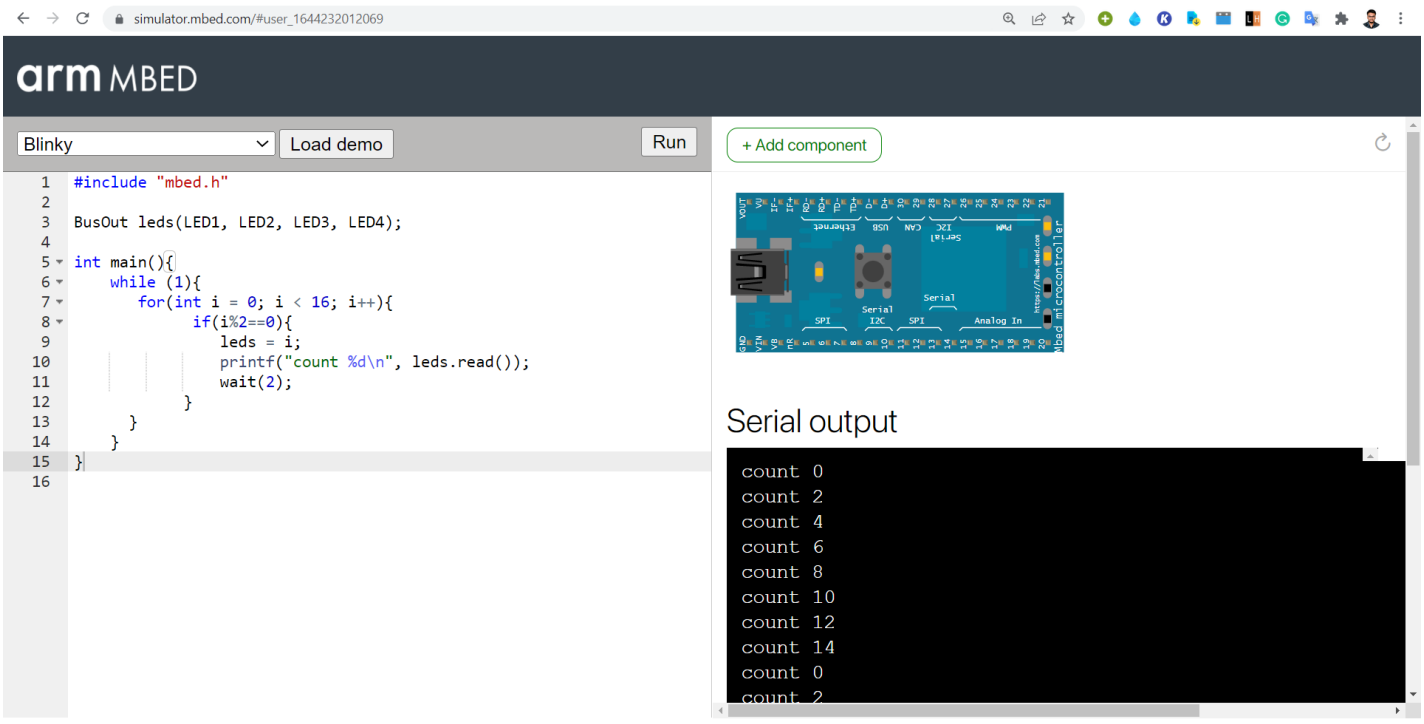
3. Blink LEDs – for all composite number below 15

```
#include "mbed.h"

BusOut myleds(LED1, LED2, LED3, LED4);

bool isComposite(int n){
    if (n == 1)
        return true;
    for (int i = 2; i < n; i++){
        if (n % i == 0)
            return true;
    }
    return false;
}
int main(){
    while (1){
        for (int i = 1; i < 16; i++){
            if (isComposite(i)){
                    myleds = i;
                    printf("%d is Composite Number\n", i);
                    wait(1);
                }
            }
        }
}
```



4. Blink LEDs – to count even numbers

```
#include "mbed.h"

BusOut leds(LED1, LED2, LED3, LED4);

int main()
{
    while (1)
    {
        for (int i = 0; i < 16; i++)
        {
            if (i % 2 == 0)
            {
                leds = i;
                printf("count %d\n", leds.read());
                wait(2);
            }
        }
    }
}
```

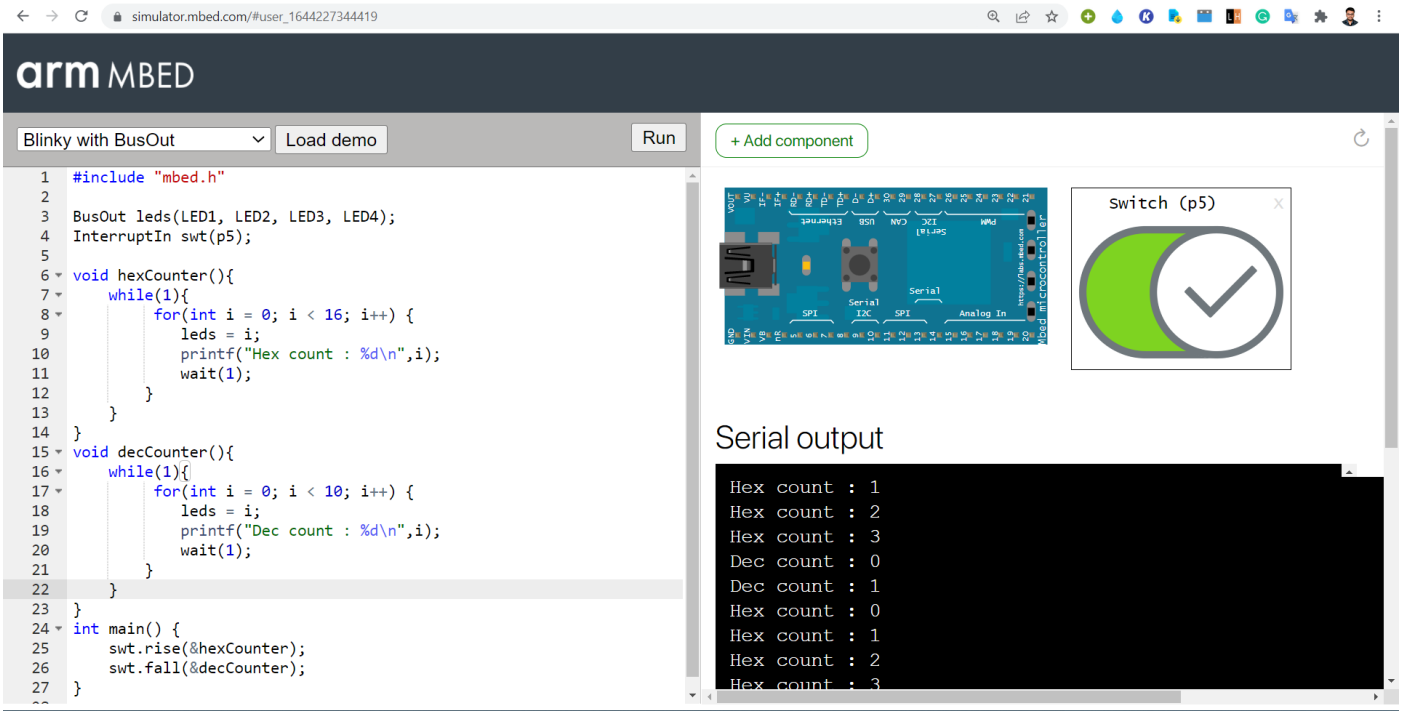


Lab 2

Date: 07-02-2022

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.

```
#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);
}
```

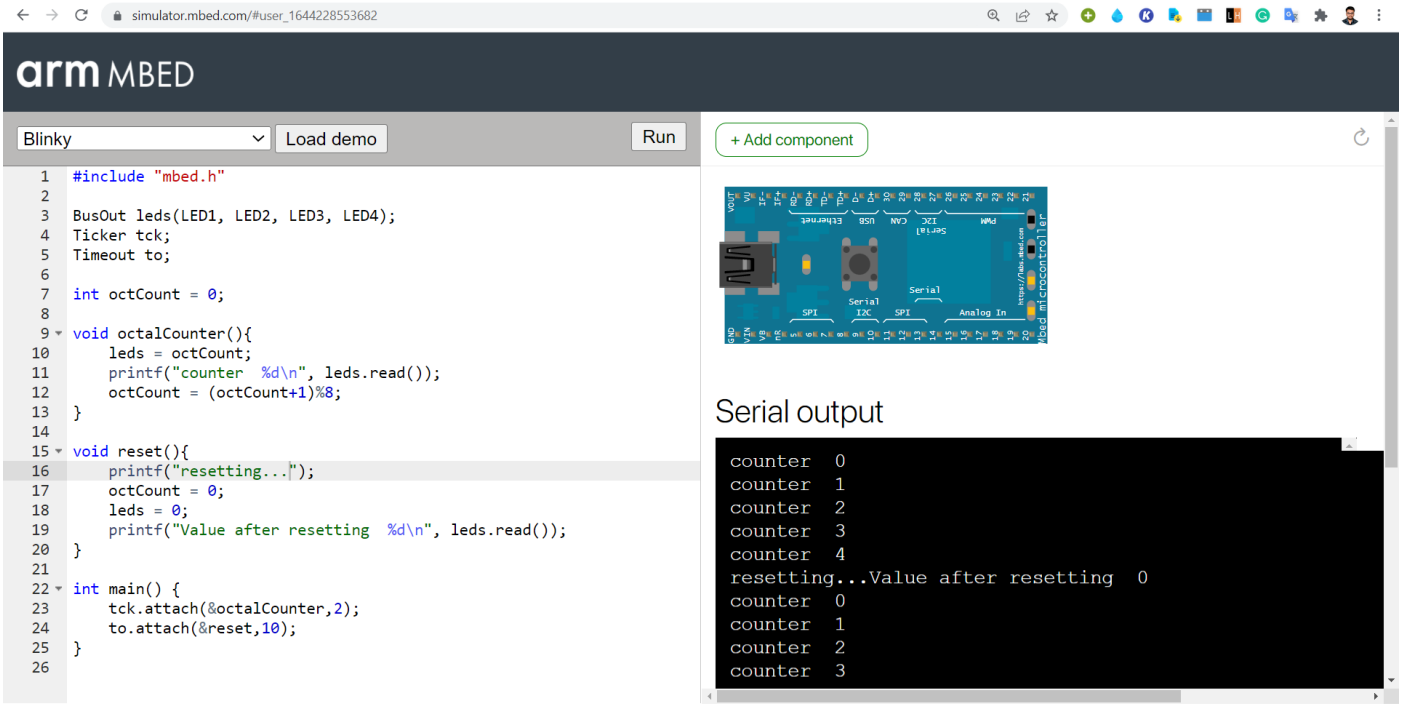



```
#include "mbed.h"

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);
}
```

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.

```
#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);
}
```



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.

```
#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);
}
```

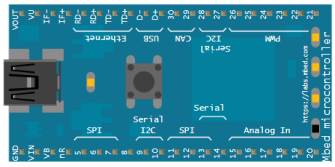
simulator.mbed.com/#user_1644231511600

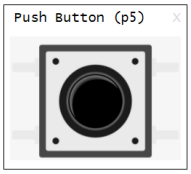
arm MBED

Blinky | Load demo | Run

```
#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;
```

+ Add component





Serial output

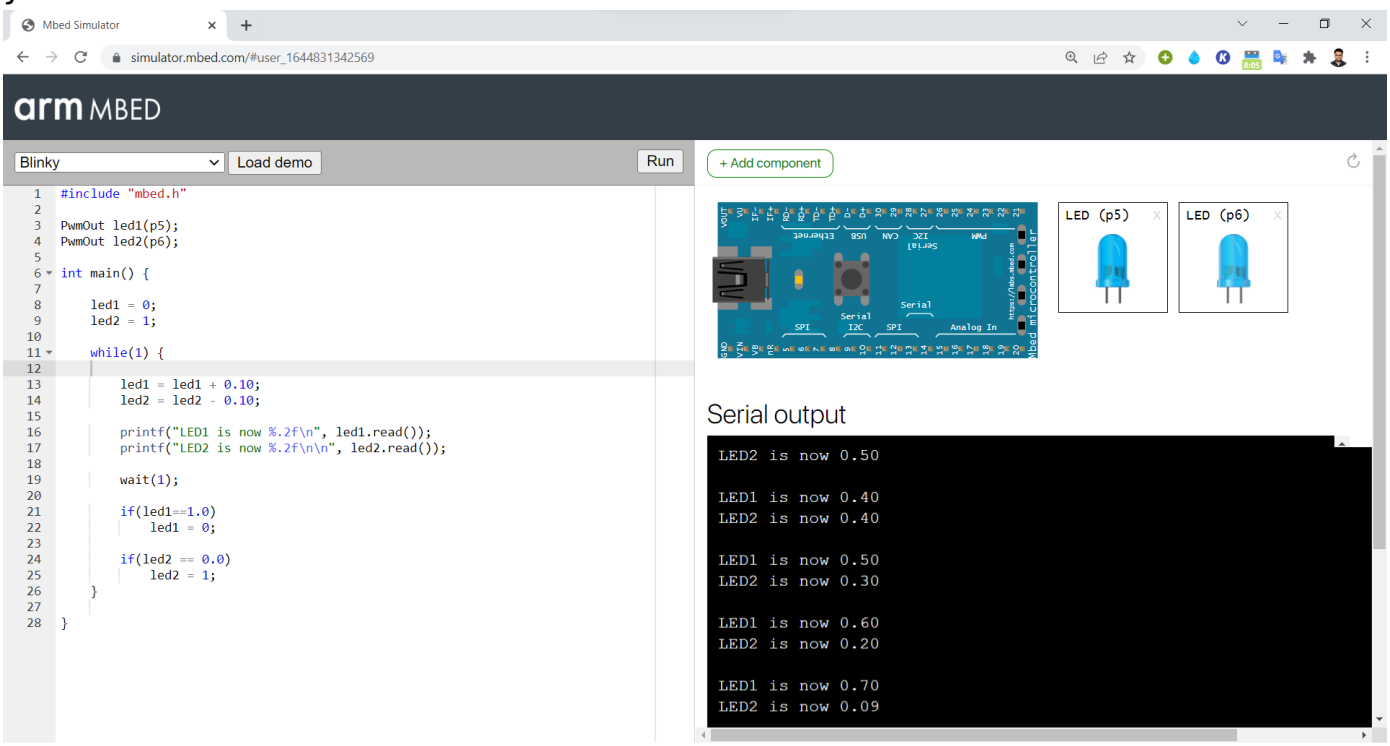
```
Odd Counter 9
resetting Odd Counter...
Starting Even Counter...
Even Counter 0
Even Counter 2
Even Counter 4
Even Counter 6
Even Counter 8
Even Counter 10
Even Counter 12
Even Counter 14
Even Counter 0
Toggle using switch...
Odd Counter 1
```

Lab 3

Date: 14-02-2022

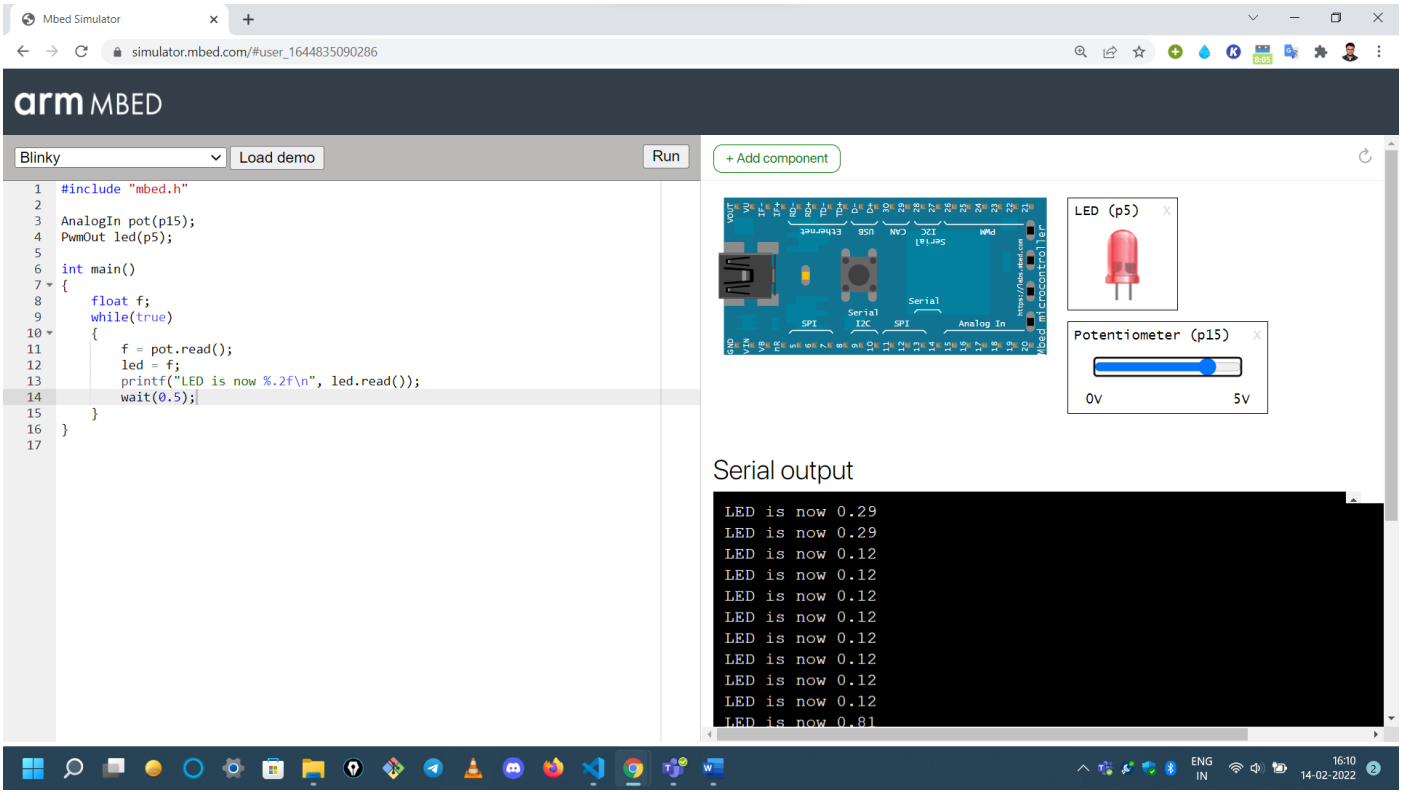
1. Increase the intensity of one LED while decreasing the other LED's intensity at the same unit. Display the intensity value of the LEDs.

```
#include "mbed.h"

PwmOut led1(p5),led2(p6);

int main(){
    led1 = 0.0;
    led2 = 1.0;
    while (true){
        led1 = led1 + 0.1;
        led2 = led2 - 0.1;
        printf("LED1 is now %.2f\n", led1.read());
        printf("LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1.0)
            led1 = 0.0;
        if (led2 == 0.0)
            led2 = 1.0;
    }
}
```



2. Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED.

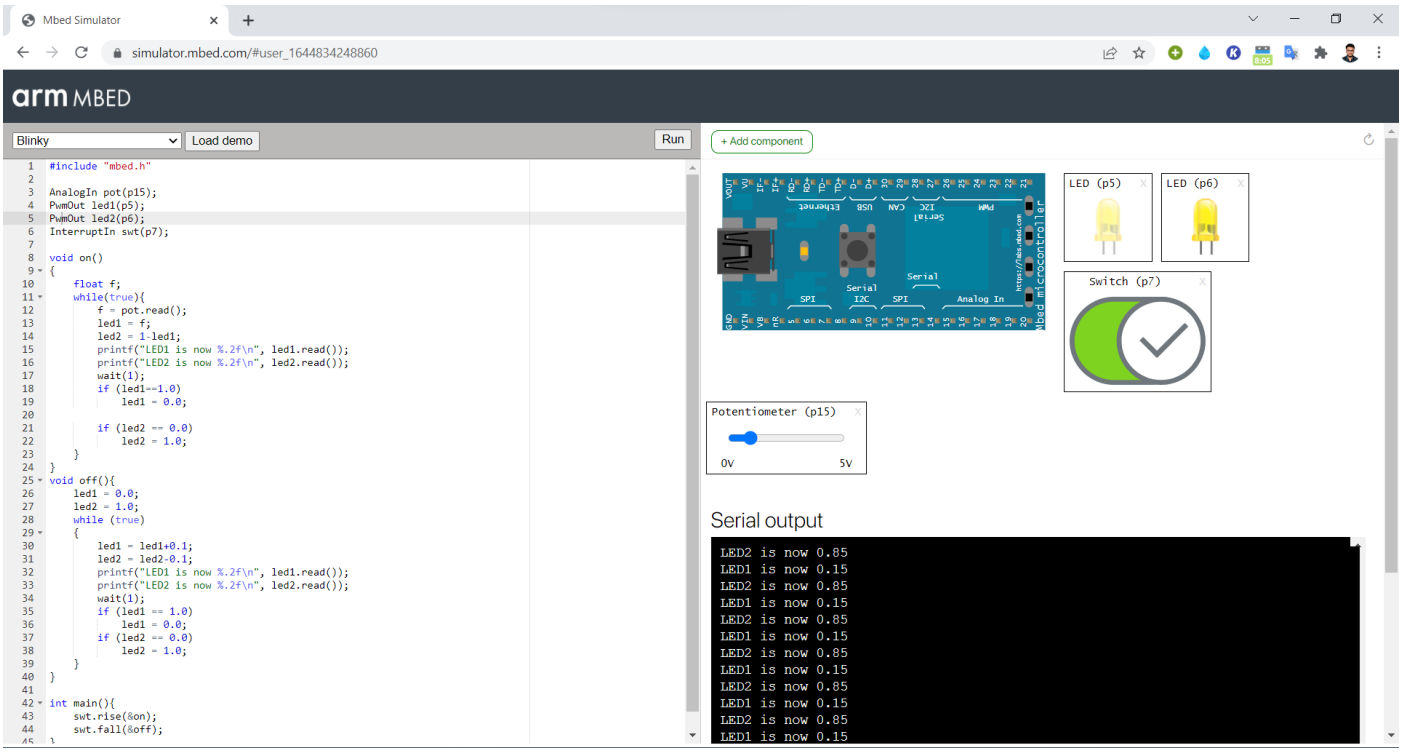
```
#include "mbed.h"

AnalogIn pot(p15);
PwmOut led(p5);

int main()
{
    float f;
    while (true)
    {
        f = pot.read();
        led = f;
        printf("LED is now %.2f\n", led.read());
        wait(0.5);
    }
}
```



3. Interface switch, 2 LEDs, potentiometer with the mbed board. If the switch is on, control the intensity of the LEDs using potentiometer (one LED should be increasing and other should be decreasing at the rate given via potentiometer). If the switch is off, blink LEDs one at increasing rate and other at decreasing rate (same rate) and loop back.

```
#include "mbed.h"

AnalogIn pot(p15);
PumOut led1(p5);
PumOut led2(p6);

InterruptIn swt(p7);

void on()
{
    float f;
    while (true)
    {
        f = pot.read();
        led1 = f;
        led2 = 1 - led1;
        printf("LED1 is now %.2f\n", led1.read());
        printf("LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1.0)
            led1 = 0.0;

        if (led2 == 0.0)
            led2 = 1.0;
    }
}
void off()
{
    led1 = 0.0;
    led2 = 1.0;
    while (true)
    {

        led1 = led1 + 0.1;
        led2 = led2 - 0.1;
        printf("LED1 is now %.2f\n", led1.read());
```

```
        printf("LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1.0)
            led1 = 0.0;
        if (led2 == 0.0)
            led2 = 1.0;
    }
}

int main()
{
    swt.rise(&on);
    swt.fall(&off);
}
```

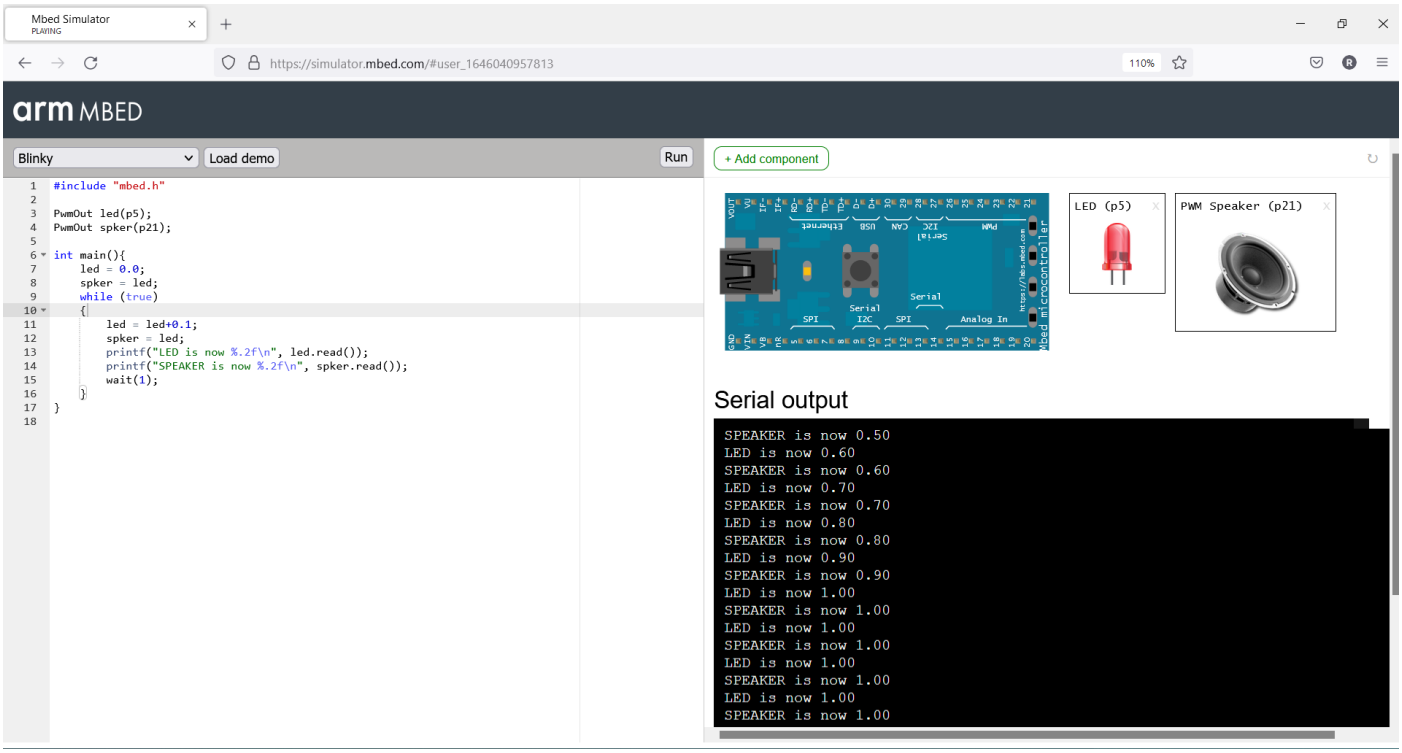


Lab 4

Date: 28-02-2022

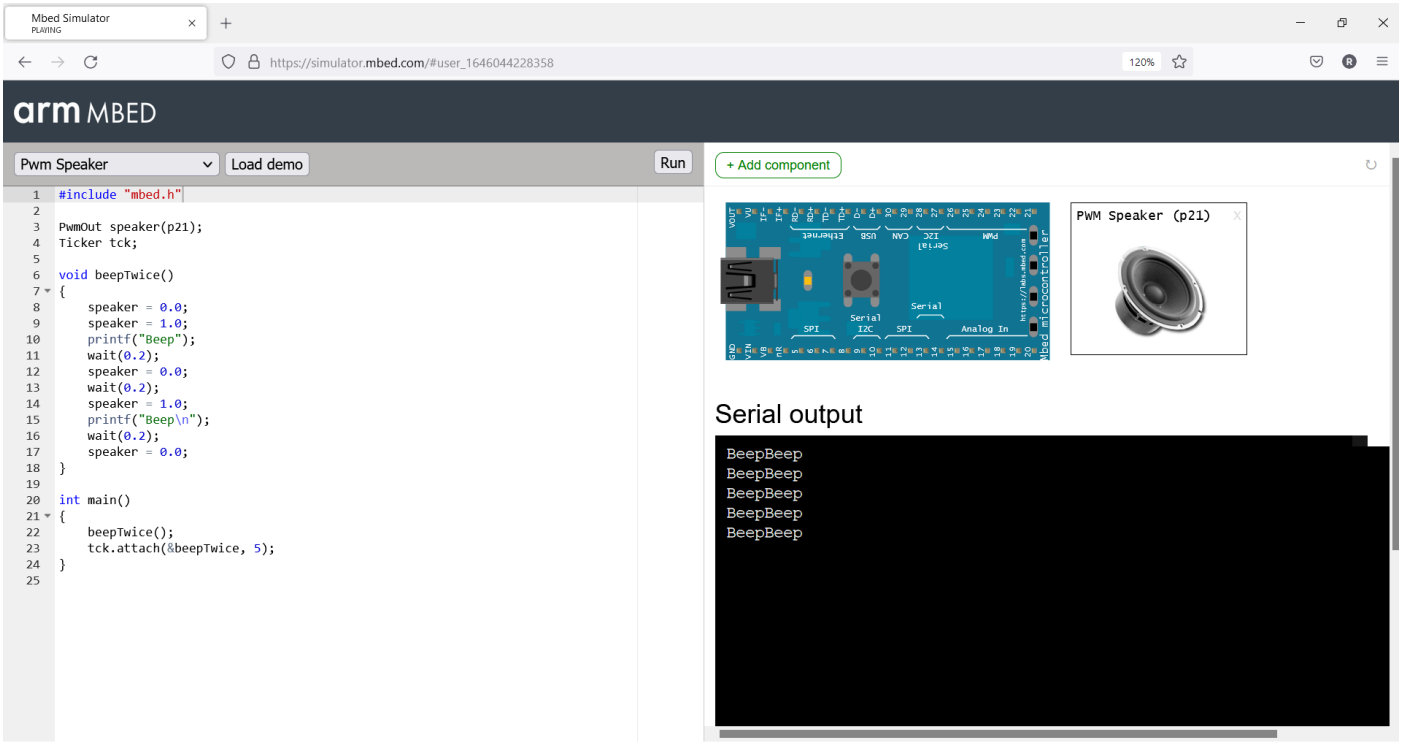
1. Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker.

```
#include "mbed.h"

PwmOut led(p5);
PwmOut spker(p6);

int main()
{
    led = 0.0;
    spker = led;
    while (true)
    {

        led = led + 0.1;
        spker = led;
        printf("LED is now %.2f\n", led.read());
        printf("SPEAKER is now %.2f\n", spker.read());
        wait(1);
    }
}
```



2. Design an alarm system, which makes double beep sound for every 5s using a speaker.

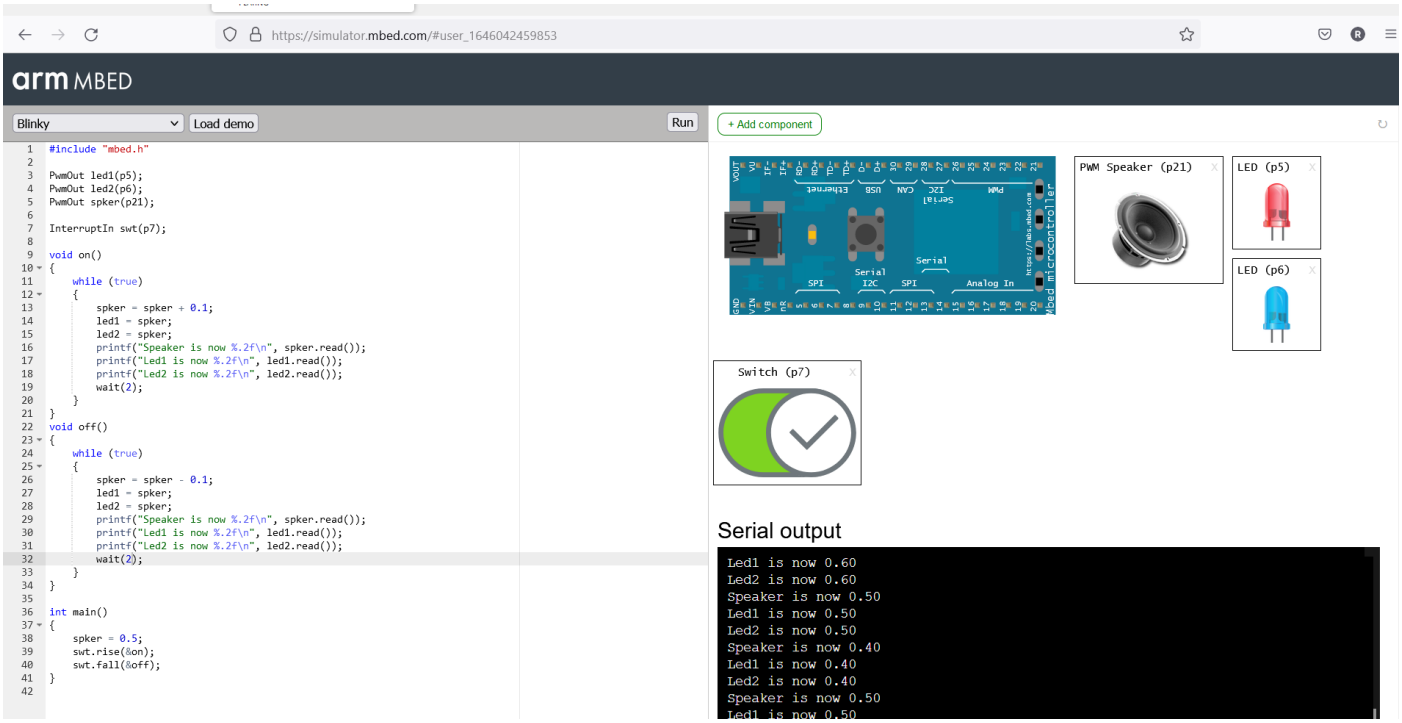
```
#include "mbed.h"

PwmOut speaker(p21);
Ticker tck;

void beepTwice(){
    speaker = 0.0;
    speaker = 1.0;
    printf("Beep");
    wait(0.2);
    speaker = 0.0;
    wait(0.2);
    speaker = 1.0;
    printf("Beep\n");
    wait(0.2);
    speaker = 0.0;
}

int main()
{
    beepTwice();
    tck.attach(&beepTwice, 5);
}
```



3. Have a switch, 2 LEDs and a speaker interfaced with mbed board. Upon switching on, the intensity of the sound should be increasing and the upon switching off, should make the intensity of the sound decreasing. Proportionally increase or decrease the intensity of two LEDs.

```
#include "mbed.h"

PwmOut led1(p5);
PwmOut led2(p6);
PwmOut spker(p21);

InterruptIn swt(p7);

void on()
{
    while (true)
    {
        spker = spker + 0.1;
        led1 = spker;
        led2 = spker;
        printf("Speaker is now %.2f\n", spker.read());
        printf("Led1 is now %.2f\n", led1.read());
        printf("Led2 is now %.2f\n", led2.read());
        wait(2);
    }
}
void off()
{
    while (true)
    {
        spker = spker - 0.1;
        led1 = spker;
        led2 = spker;
        printf("Speaker is now %.2f\n", spker.read());
        printf("Led1 is now %.2f\n", led1.read());
        printf("Led2 is now %.2f\n", led2.read());
        wait(2);
    }
}

int main()
{
```

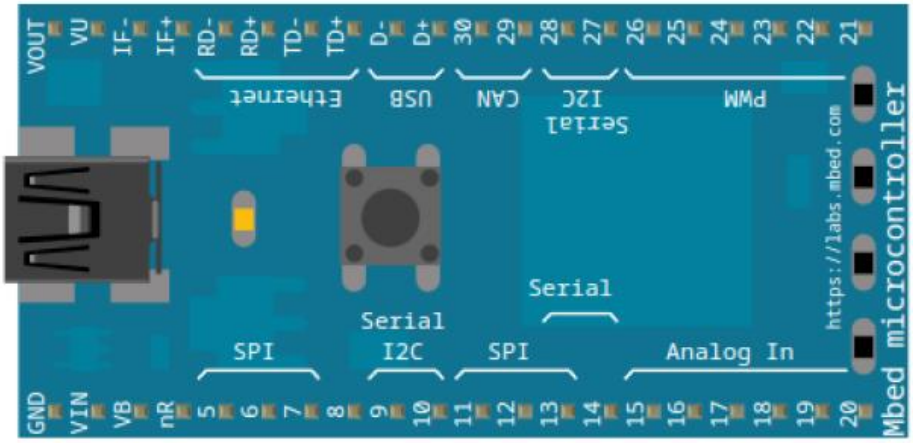
```
    spker = 0.5;
    swt.rise(&on);
    swt.fall(&off);
}
```



4. Design a program to make ambulance sound.

```
#include "mbed.h"
PwmOut speaker(p21);
int main()
{
    int i;
    while (1)
    {
        for (i = 0; i < 26; i++)
        {
            speaker.period(1.0 / 259.0);
            speaker = float(i) / 500.0;
            wait(0.5);
            speaker.period(1.0 / 340.0);
            wait(0.5);
        }
        for (i = 26; i >= 0; i--)
```

```
        {
            speaker.period(1.0 / 259.0);
            speaker = float(i) / 500.0;
            wait(0.5);
            speaker.period(1.0 / 440.0);
            wait(0.5);
        }
    }
}
```

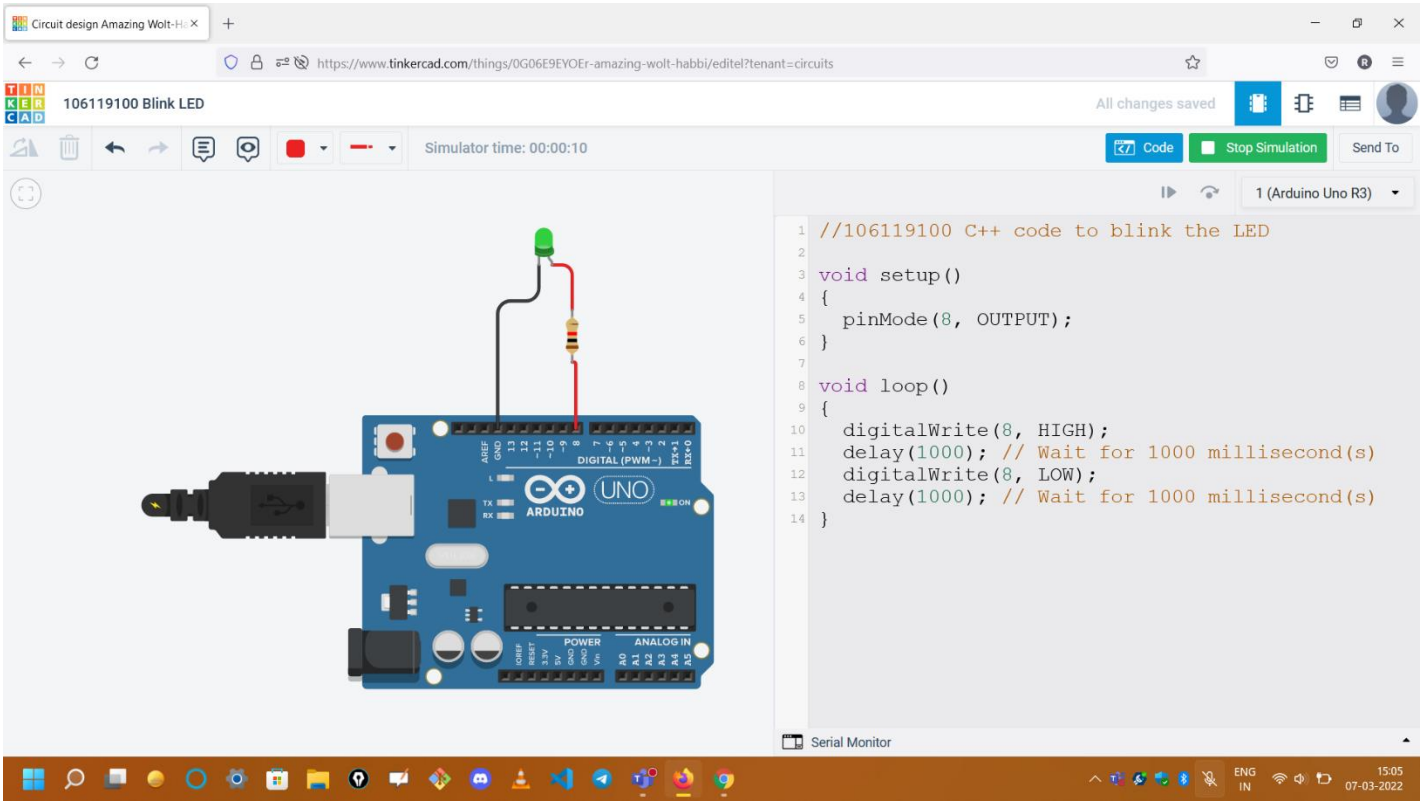


Lab 5

Date: 07-03-2022

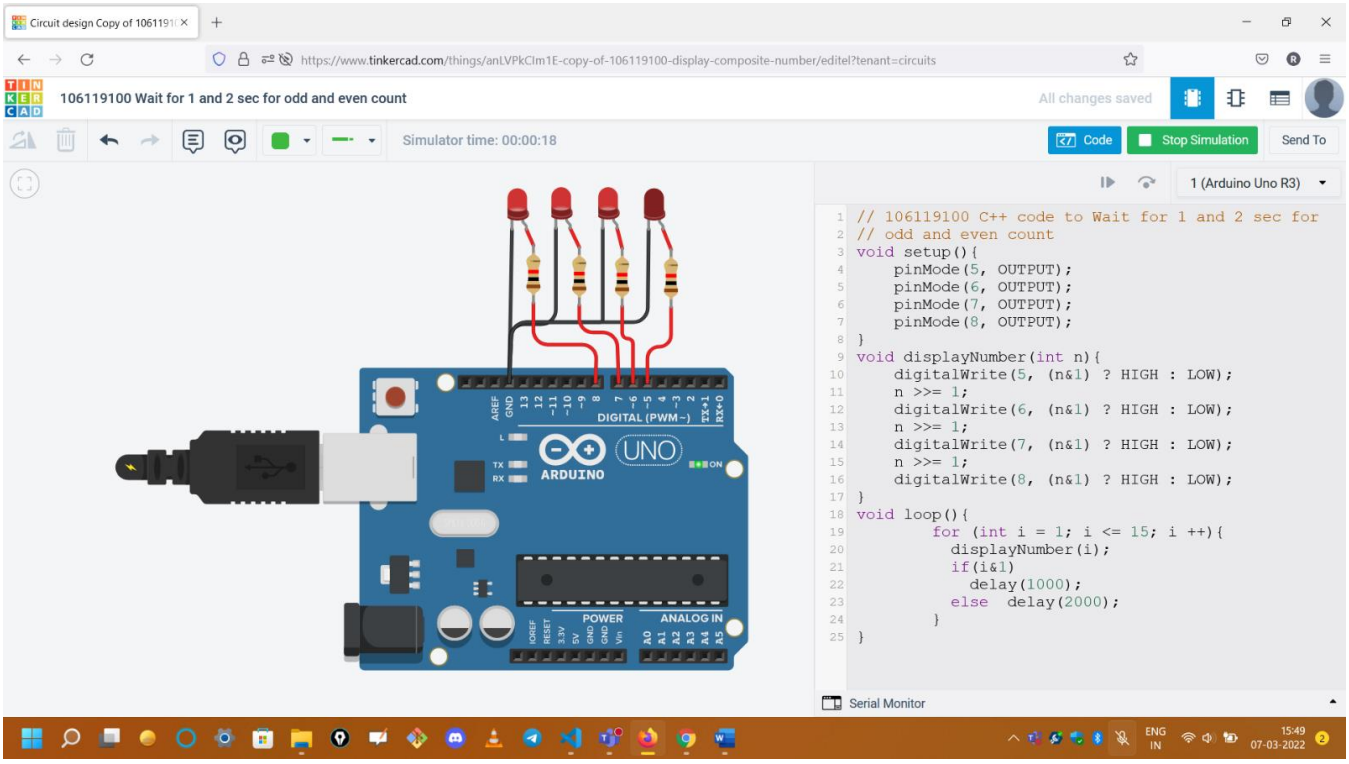
1. Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.

```
void setup()
{
    pinMode(8, OUTPUT);
}

void loop()
{
    digitalWrite(8, HIGH);
    delay(1000); // Wait for 1000 millisecond(s)
    digitalWrite(8, LOW);
    delay(1000); // Wait for 1000 millisecond(s)
}
```



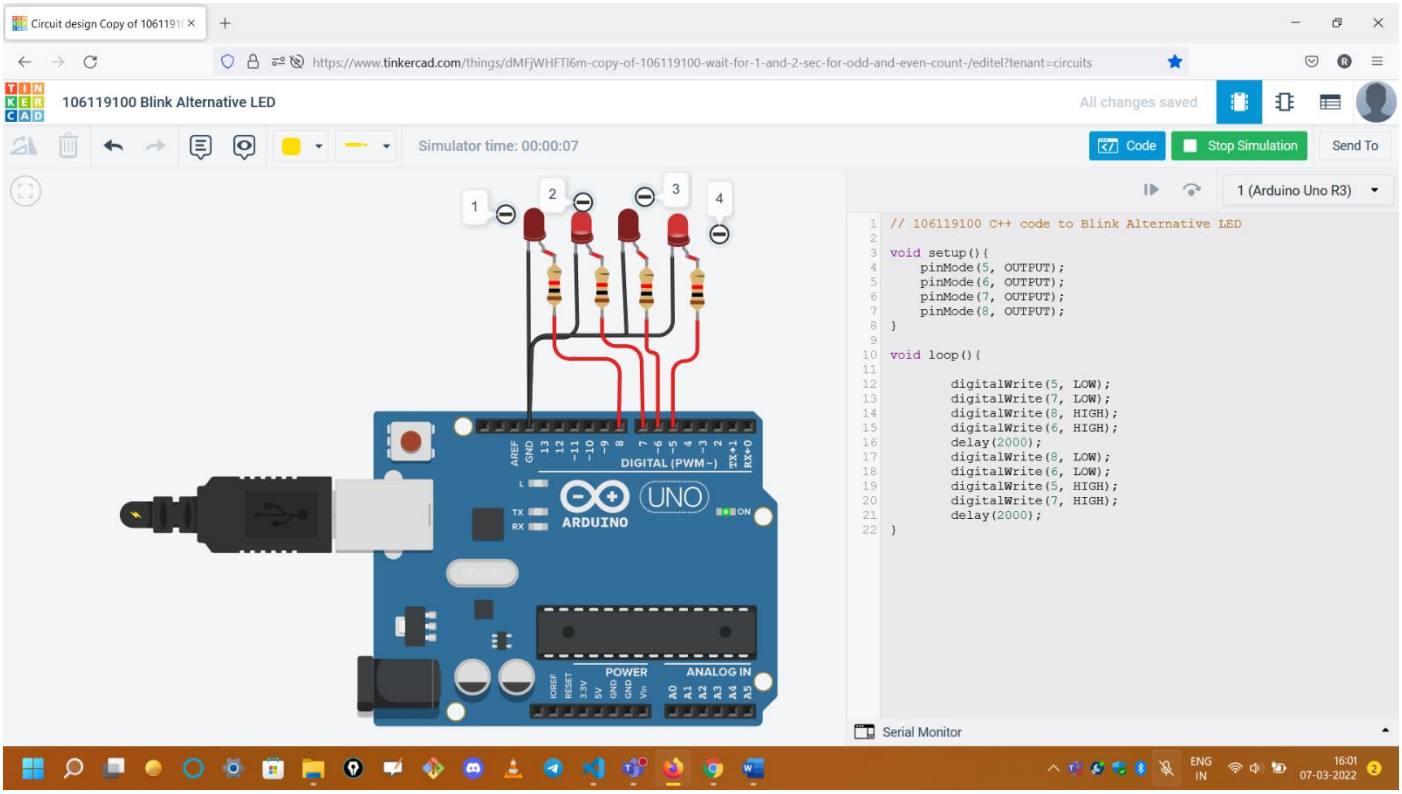
2. Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec

```
void setup(){
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
}
void displayNumber(int n){
    digitalWrite(5, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(6, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(7, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(8, (n & 1) ? HIGH : LOW);
}
void loop(){
    for (int i = 1; i <= 15; i++){
        displayNumber(i);
        if (i & 1)
            delay(1000);
        else
            delay(2000);
    }
}
```



3.Blink Alternate LEDs

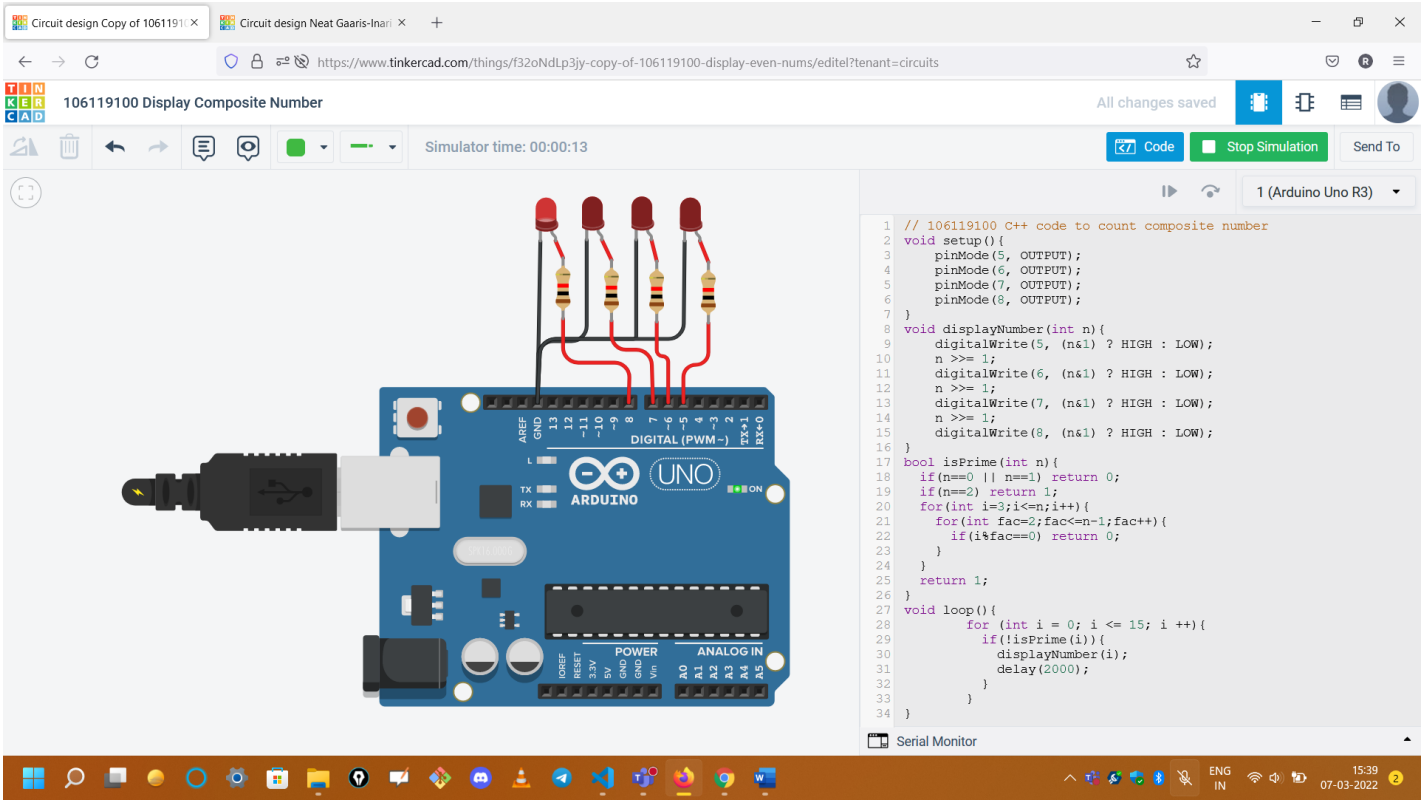
```
void setup()
{
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
}
void loop()
{
    digitalWrite(5, LOW);
    digitalWrite(7, LOW);
    digitalWrite(8, HIGH);
    digitalWrite(6, HIGH);
    delay(2000);
    digitalWrite(8, LOW);
    digitalWrite(6, LOW);
    digitalWrite(5, HIGH);
    digitalWrite(7, HIGH);
    delay(2000);
}
```



4. Blink LEDs - for all composite number below 15

```
// 106119100 C++ code to count composite number

void setup()
{
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
}
void displayNumber(int n)
{
    digitalWrite(5, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(6, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(7, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(8, (n & 1) ? HIGH : LOW);
}
bool isPrime(int n)
{
    if (n == 0 || n == 1)
        return 0;
    if (n == 2)
        return 1;
    for (int i = 3; i <= n; i++)
    {
        for (int fac = 2; fac <= n - 1; fac++)
        {
            if (i % fac == 0)
                return 0;
        }
    }
    return 1;
}
void loop()
{
    for (int i = 0; i <= 15; i++)
    {
        if (!isPrime(i))
        {
            displayNumber(i);
```

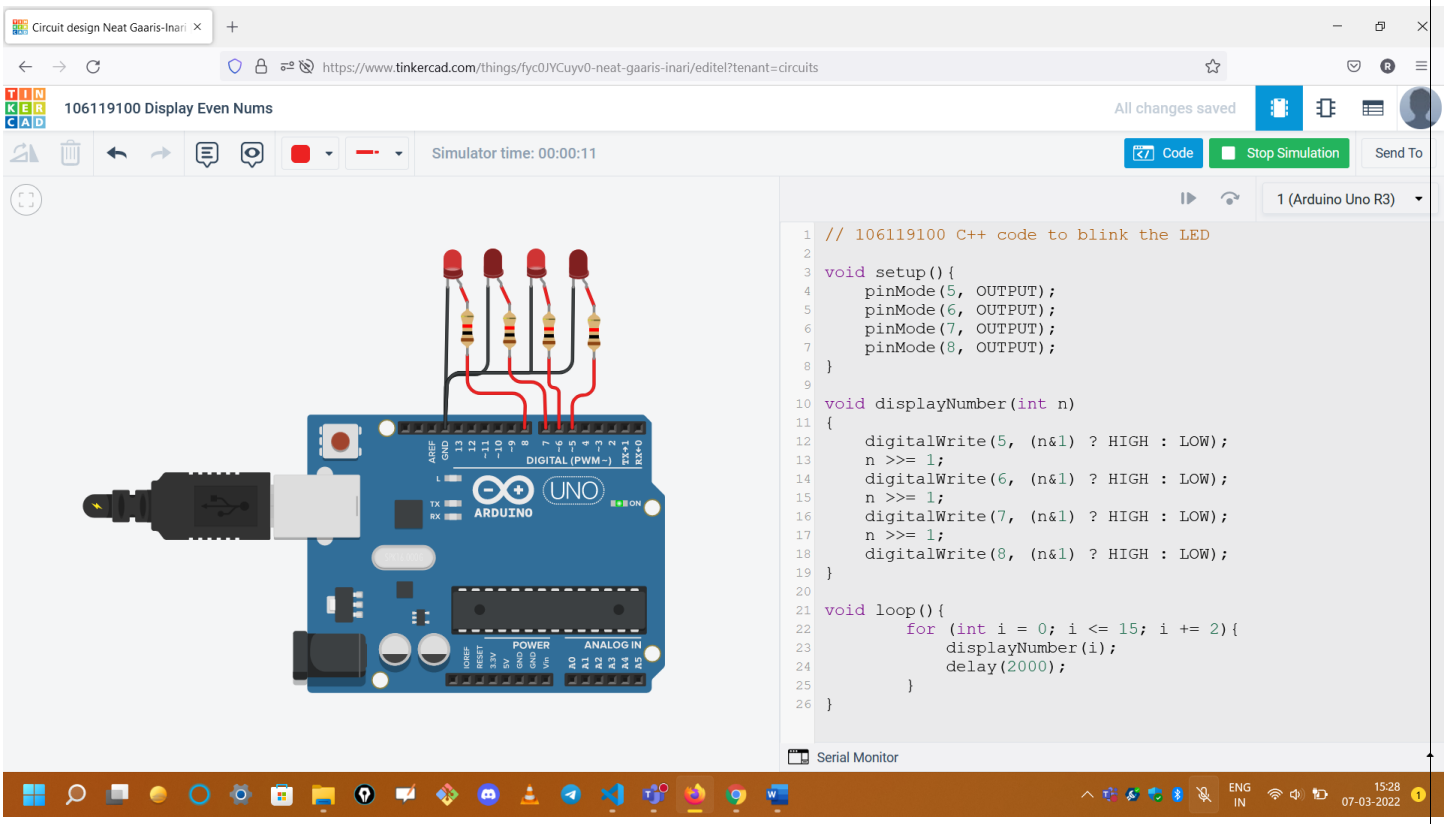
```
            delay(2000);
        }
    }
}
```



5. Blink LEDs – to count even numbers

```
// 106119100 C++ code to blink the LED

void setup()
{
    pinMode(5, OUTPUT);
    pinMode(6, OUTPUT);
    pinMode(7, OUTPUT);
    pinMode(8, OUTPUT);
}

void displayNumber(int n)
{
    digitalWrite(5, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(6, (n & 1) ? HIGH : LOW);
    n >>= 1;
```

```
    digitalWrite(7, (n & 1) ? HIGH : LOW);
    n >>= 1;
    digitalWrite(8, (n & 1) ? HIGH : LOW);
}

void loop()
{
    for (int i = 0; i <= 15; i += 2)
    {
        displayNumber(i);
        delay(2000);
    }
}
```



Lab 6

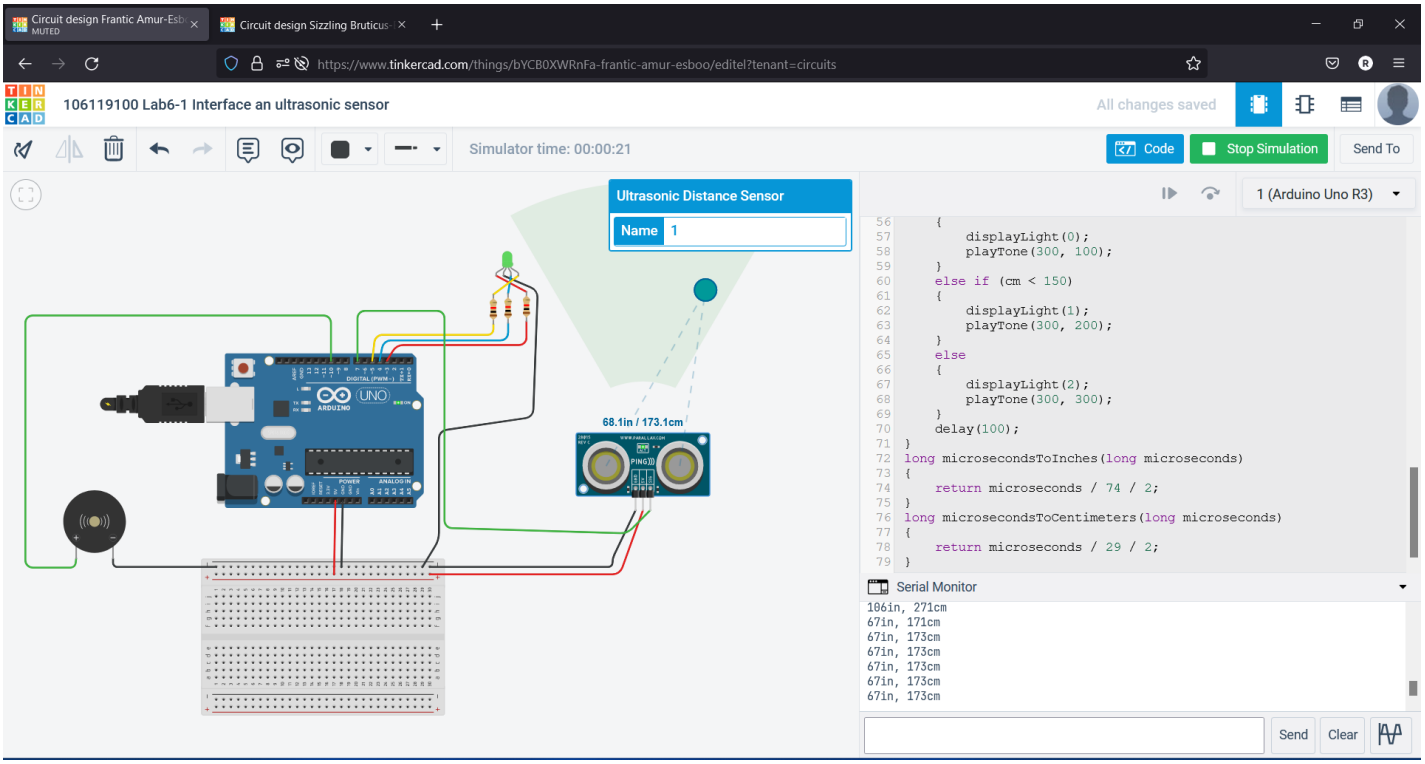
Date: 28-03-2022

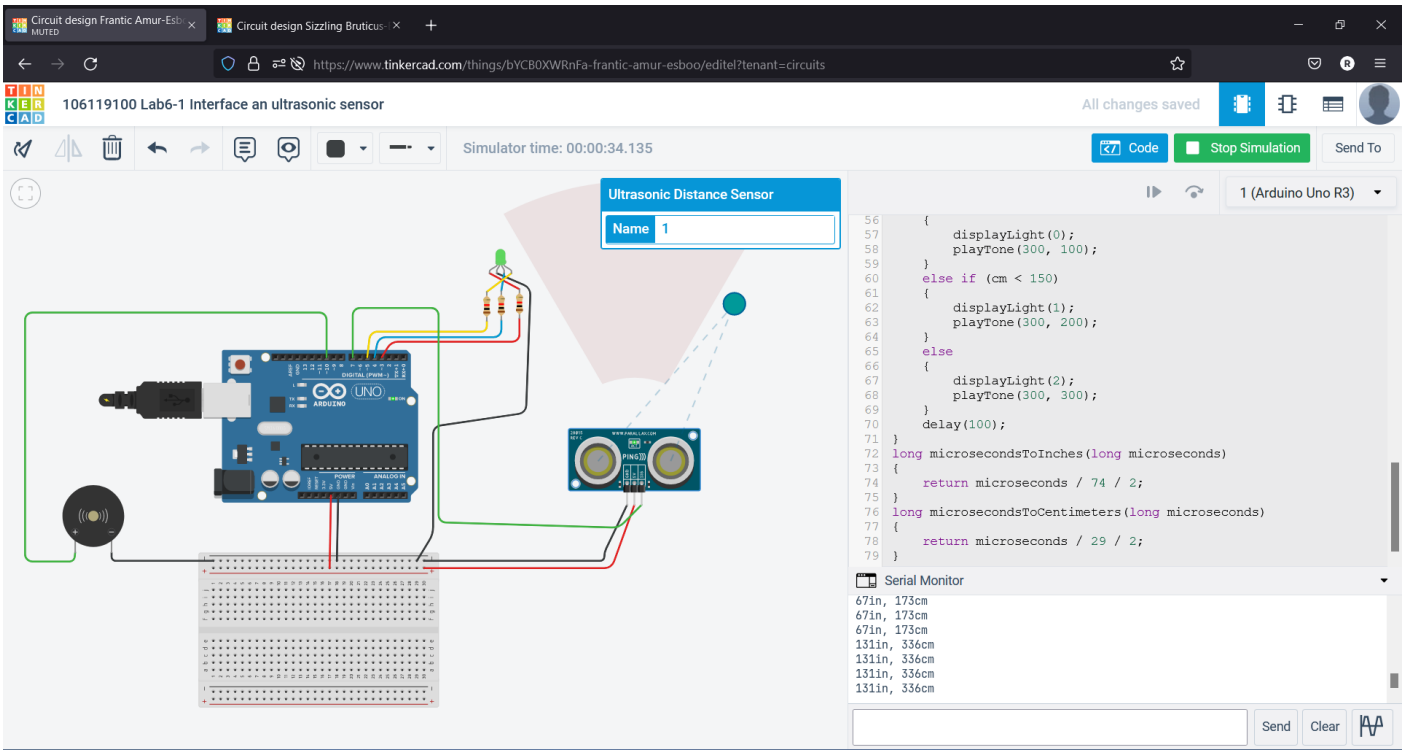
1. Interface an ultrasonic sensor and display the distance measured in a screen and glow red, blue, and green led if the distance is close, moderate, and far respectively. Play different tones for different cases.

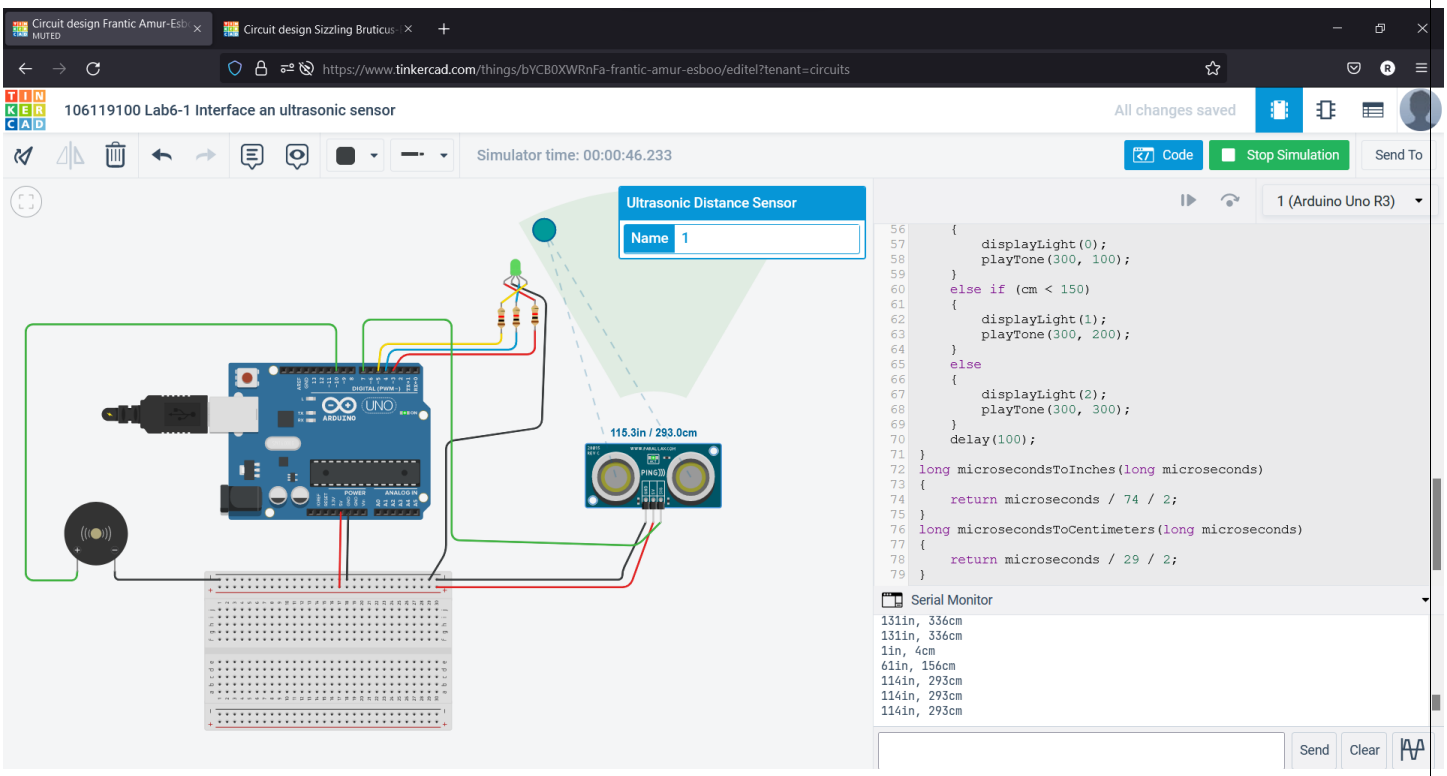
```
#define F(x) x ? HIGH : LOW
const int red = 3;
const int blue = 4;
const int green = 5;
const int speaker = 10;
const int ping = 7;

void setup()
{
    // initialize serial communication:
    Serial.begin(9600);
    pinMode(red, OUTPUT);
    pinMode(blue, OUTPUT);
    pinMode(green, OUTPUT);
    pinMode(speaker, OUTPUT);
}
void displayLight(int code)
{
    digitalWrite(red, F(code == 0));
    digitalWrite(blue, F(code == 1));
    digitalWrite(green, F(code == 2));
}
void playTone(long duration, int freq)
{
    duration *= 1000;
    int period = (1.0 / freq) * 100000;
    long elapsed_time = 0;
    while (elapsed_time < duration)
    {
        digitalWrite(speaker, HIGH);
        delayMicroseconds(period / 2);
        digitalWrite(speaker, LOW);
        delayMicroseconds(period / 2);
        elapsed_time += (period);
```

```
    }
}
void loop()
{
    long duration, inches, cm;
    pinMode(ping, OUTPUT);
    digitalWrite(ping, LOW);
    delayMicroseconds(2);
    digitalWrite(ping, HIGH);
    delayMicroseconds(5);
    digitalWrite(ping, LOW);
    pinMode(ping, INPUT);
    duration = pulseIn(ping, HIGH);
    inches = microsecondsToInches(duration);
    cm = microsecondsToCentimeters(duration);
    Serial.print(inches);
    Serial.print("in, ");
    Serial.print(cm);
    Serial.print("cm");
    Serial.println();
    if (cm < 50)
    {
        displayLight(0);
        playTone(300, 100);
    }
    else if (cm < 150)
    {
        displayLight(1);
        playTone(300, 200);
    }
    else
    {
        displayLight(2);
        playTone(300, 300);
    }
    delay(100);
}
long microsecondsToInches(long microseconds)
{
    return microseconds / 74 / 2;
}
long microsecondsToCentimeters(long microseconds)
{
    return microseconds / 29 / 2;
```

}


Circuit design Frantic Amur-Esb
MUTED
Circuit design Sizzling Bruticus
https://www.tinkercad.com/things/bYCB0XWRnFa-frantic-amur-esboo/editel?tenant=circuits
106119100 Lab6-1 Interface an ultrasonic sensor
All changes saved
Simulator time: 00:00:21
Code
Stop Simulation
Send To
Ultrasonic Distance Sensor
Name 1
68.1in / 173.1cm
1 (Arduino Uno R3)
{
displayLight(0);
playTone(300, 100);
}
else if (cm < 150)
{
displayLight(1);
playTone(300, 200);
}
else
{
displayLight(2);
playTone(300, 300);
}
delay(100);
}
long microsecondsToInches(long microseconds)
{
return microseconds / 74 / 2;
}
long microsecondsToCentimeters(long microseconds)
{
return microseconds / 29 / 2;
}
Serial Monitor
106in, 271cm
67in, 171cm
67in, 173cm
67in, 173cm
67in, 173cm
67in, 173cm
67in, 173cm
Send
Clear

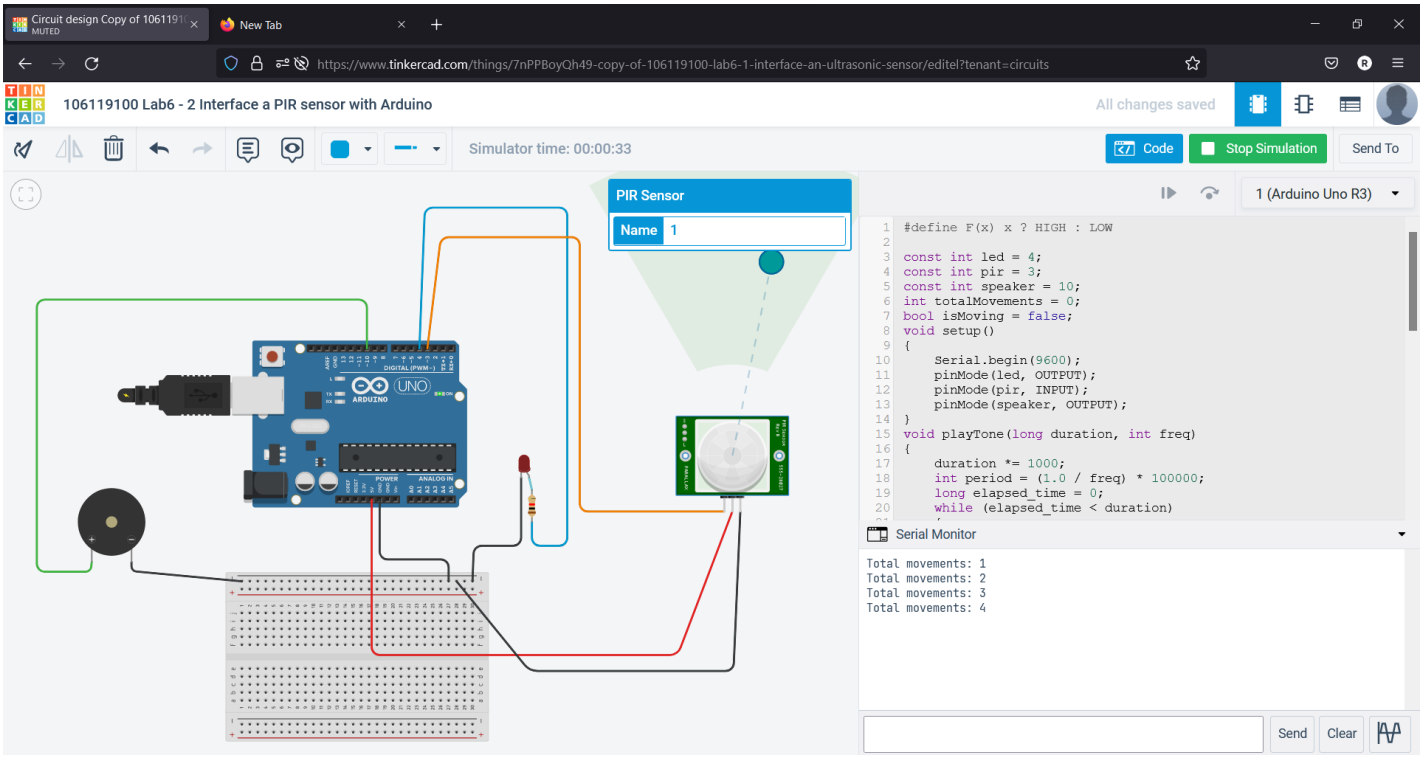

Circuit design Frantic Amur-Esb
MUTED
Circuit design Sizzling Bruticus
https://www.tinkercad.com/things/bYCB0XWRnFa-frantic-amur-esboo/editel?tenant=circuits
106119100 Lab6-1 Interface an ultrasonic sensor
All changes saved
Simulator time: 00:00:34.135
Code
Stop Simulation
Send To
Ultrasonic Distance Sensor
Name 1
1 (Arduino Uno R3)
{
displayLight(0);
playTone(300, 100);
}
else if (cm < 150)
{
displayLight(1);
playTone(300, 200);
}
else
{
displayLight(2);
playTone(300, 300);
}
delay(100);
}
long microsecondsToInches(long microseconds)
{
return microseconds / 74 / 2;
}
long microsecondsToCentimeters(long microseconds)
{
return microseconds / 29 / 2;
}
Serial Monitor
67in, 173cm
67in, 173cm
67in, 173cm
131in, 336cm
131in, 336cm
131in, 336cm
131in, 336cm
Send
Clear

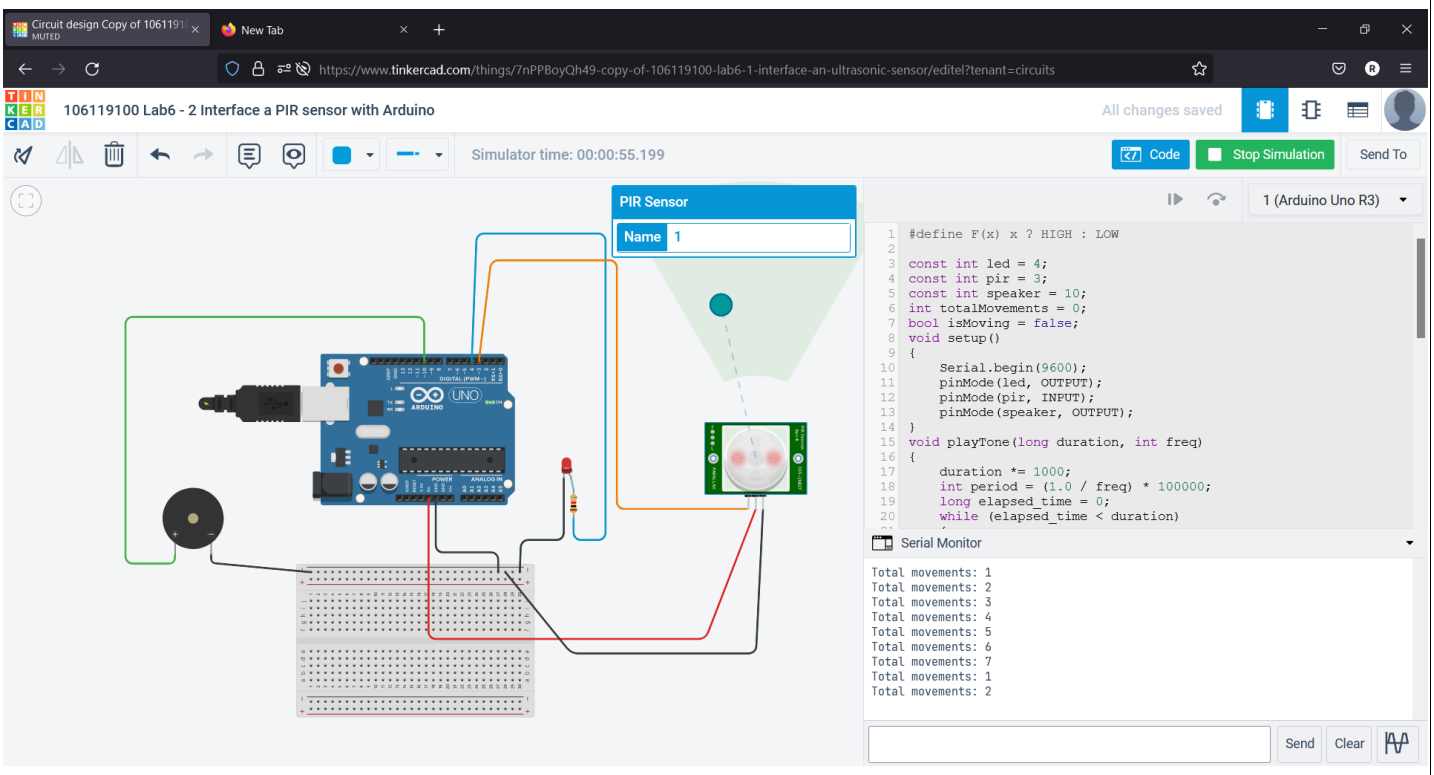


2. Interface a PIR sensor with Arduino and blink a led if any movement is captured by the sensor and count the number of movements. Make a buzzer when movement detected.

```
// 106119100 Interface a PIR sensor with Arduino
#define F(x) x ? HIGH : LOW
const int led = 4;
const int pir = 3;
const int speaker = 10;
int totalMovements = 0;
bool isMoving = false;
void setup()
{
    Serial.begin(9600);
    pinMode(led, OUTPUT);
    pinMode(pir, INPUT);
    pinMode(speaker, OUTPUT);
}
void playTone(long duration, int freq)
{
    duration *= 1000;
```

```
    int period = (1.0 / freq) * 100000;
    long elapsed_time = 0;
    while (elapsed_time < duration)
    {
        digitalWrite(speaker, HIGH);
        delayMicroseconds(period / 2);
        digitalWrite(speaker, LOW);
        delayMicroseconds(period / 2);
        elapsed_time += (period);
    }
}
void loop()
{
    int pirStat = digitalRead(pir);
    if (pirStat == HIGH)
    {
        if (!isMoving)
        {
            playTone(300, 300);
            totalMovements += 1;
            Serial.print("Total movements: ");
            Serial.print(totalMovements);
            Serial.print("\n");
            isMoving = true;
        }
        digitalWrite(led, HIGH);
    }
    else
    {
        isMoving = false;
        digitalWrite(led, LOW);
    }
}
```


106119100 Lab6 - 2 Interface a PIR sensor with Arduino
All changes saved
Simulator time: 00:00:33
Code
Stop Simulation
Send To
PIR Sensor
Name 1
1 (Arduino Uno R3)
#define F(x) x ? HIGH : LOW
const int led = 4;
const int pir = 3;
const int speaker = 10;
int totalMovements = 0;
bool isMoving = false;
void setup()
{
Serial.begin(9600);
pinMode(led, OUTPUT);
pinMode(pir, INPUT);
pinMode(speaker, OUTPUT);
}
void playTone(long duration, int freq)
{
duration *= 1000;
int period = (1.0 / freq) * 100000;
long elapsed_time = 0;
while (elapsed_time < duration)
Serial Monitor
Total movements: 1
Total movements: 2
Total movements: 3
Total movements: 4
Send
Clear


106119100 Lab6 - 2 Interface a PIR sensor with Arduino
All changes saved
Simulator time: 00:00:55.199
Code
Stop Simulation
Send To
PIR Sensor
Name 1
1 (Arduino Uno R3)
#define F(x) x ? HIGH : LOW
const int led = 4;
const int pir = 3;
const int speaker = 10;
int totalMovements = 0;
bool isMoving = false;
void setup()
{
Serial.begin(9600);
pinMode(led, OUTPUT);
pinMode(pir, INPUT);
pinMode(speaker, OUTPUT);
}
void playTone(long duration, int freq)
{
duration *= 1000;
int period = (1.0 / freq) * 100000;
long elapsed_time = 0;
while (elapsed_time < duration)
Serial Monitor
Total movements: 1
Total movements: 2
Total movements: 3
Total movements: 4
Total movements: 5
Total movements: 6
Total movements: 7
Total movements: 1
Total movements: 2
Send
Clear

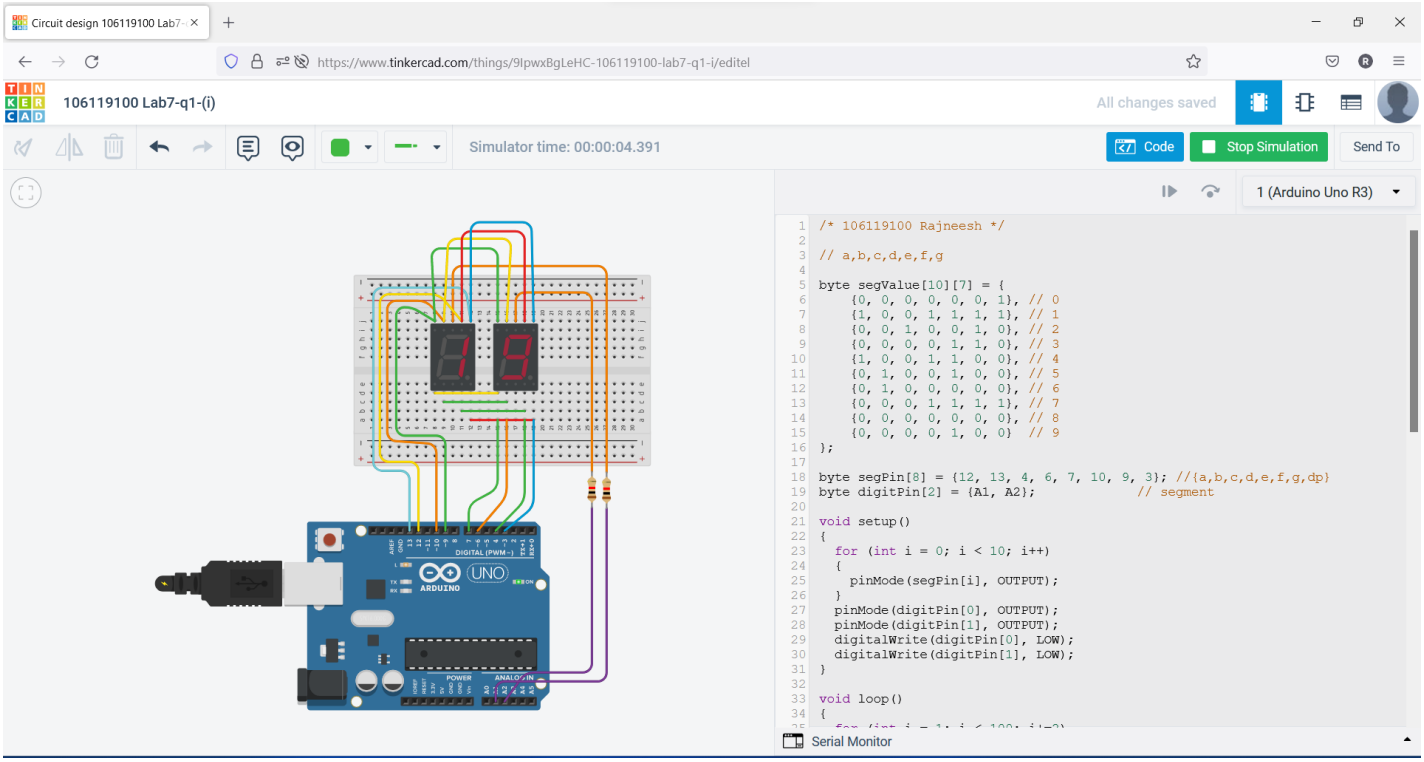
Lab 7

Date: 11-04-2022

1. Interface two 7-segment display with Arduino Uno board.

```
/* 106119100 Rajneesh */

// a,b,c,d,e,f,g
byte segValue[10][7] = {
    {0, 0, 0, 0, 0, 0, 1}, // 0
    {1, 0, 0, 1, 1, 1, 1}, // 1
    {0, 0, 1, 0, 0, 1, 0}, // 2
    {0, 0, 0, 0, 1, 1, 0}, // 3
    {1, 0, 0, 1, 1, 0, 0}, // 4
    {0, 1, 0, 0, 1, 0, 0}, // 5
    {0, 1, 0, 0, 0, 0, 0}, // 6
    {0, 0, 0, 1, 1, 1, 1}, // 7
    {0, 0, 0, 0, 0, 0, 0}, // 8
    {0, 0, 0, 0, 1, 0, 0}  // 9
};

byte segPin[8] = {12, 13, 4, 6, 7, 10, 9, 3}; //{a,b,c,d,e,f,g,dp}
byte digitPin[2] = {A1, A2};                  // segment

void setup()
{
    for (int i = 0; i < 10; i++)
    {
        pinMode(segPin[i], OUTPUT);
    }
    pinMode(digitPin[0], OUTPUT);
    pinMode(digitPin[1], OUTPUT);
    digitalWrite(digitPin[0], LOW);
    digitalWrite(digitPin[1], LOW);
}

void loop()
{
    for (int i = 1; i < 100; i += 2)
    {
        display_N(i);
        delay(1);
```

```
    }
}

void display_N(int num)
{
    int und = num % 10;
    int dec = (num % 100) / 10;
    for (int i = 0; i < 100; i++)
    {
        segOutput(1, und, 1);
        segOutput(0, dec, 1);
        delay(2);
    }

    Serial.print(dec);
    Serial.println(und);
}

// LED
void segClear()
{
    for (int i = 0; i < 8; i++)
    {
        digitalWrite(segPin[i], HIGH);
    }
}
// LED
void segOutput(int d, int Number, int dp)
{
    segClear();
    digitalWrite(digitPin[d], HIGH);
    for (int i = 0; i < 7; i++)
    {
        digitalWrite(segPin[i], segValue[Number][i]);
    }
    digitalWrite(segPin[7], dp);
    delayMicroseconds(1000);
    digitalWrite(digitPin[d], LOW);
}
```

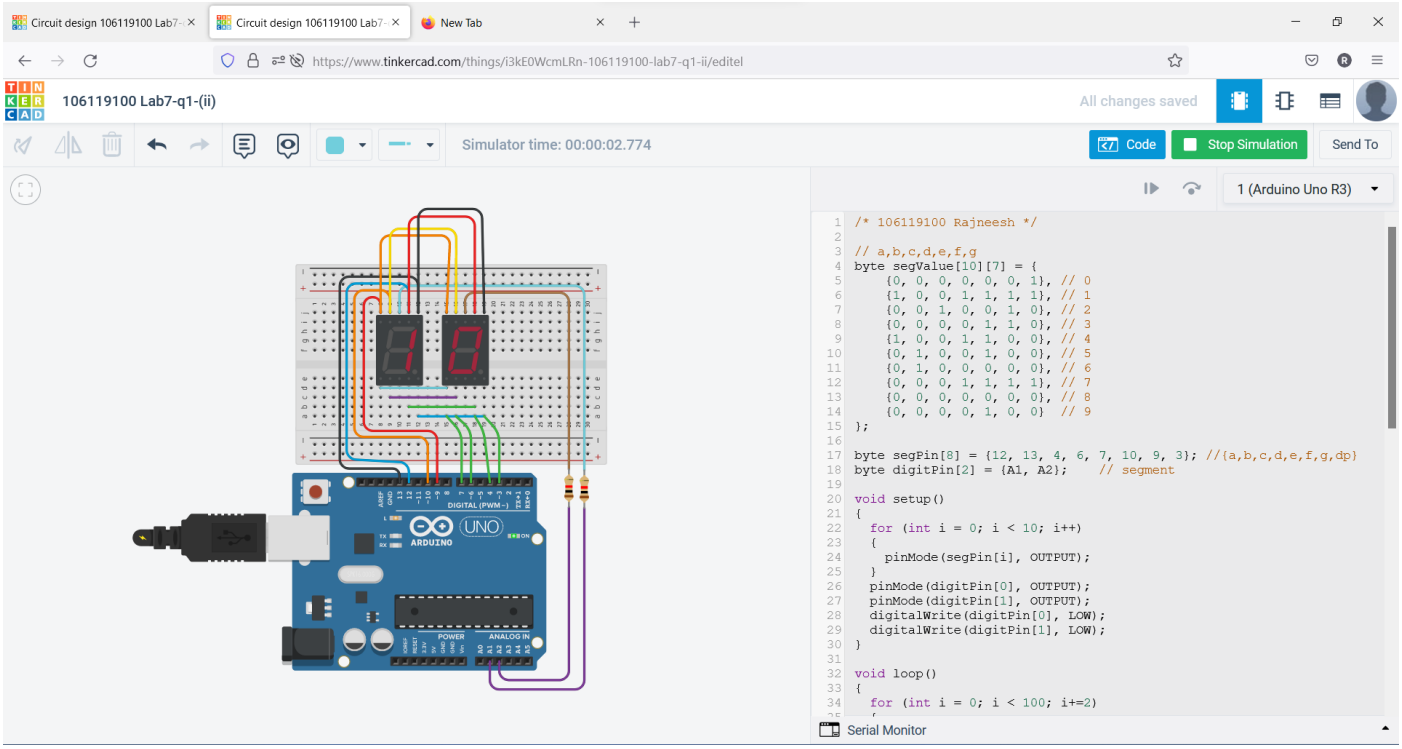


2. Implement the a)odd counter and b)even counter and display the values in the seven-segment display.

```
/* 106119100 Rajneesh */

// a,b,c,d,e,f,g
byte segValue[10][7] = {
    {0, 0, 0, 0, 0, 0, 1}, // 0
    {1, 0, 0, 1, 1, 1, 1}, // 1
    {0, 0, 1, 0, 0, 1, 0}, // 2
    {0, 0, 0, 0, 1, 1, 0}, // 3
    {1, 0, 0, 1, 1, 0, 0}, // 4
    {0, 1, 0, 0, 1, 0, 0}, // 5
    {0, 1, 0, 0, 0, 0, 0}, // 6
    {0, 0, 0, 1, 1, 1, 1}, // 7
    {0, 0, 0, 0, 0, 0, 0}, // 8
    {0, 0, 0, 0, 1, 0, 0}  // 9
};
byte segPin[8] = {12, 13, 4, 6, 7, 10, 9, 3}; //{a,b,c,d,e,f,g,dp}
byte digitPin[2] = {A1, A2};                  // segment

void setup()
{
```

```
    for (int i = 0; i < 10; i++)
    {
        pinMode(segPin[i], OUTPUT);
    }
    pinMode(digitPin[0], OUTPUT);
    pinMode(digitPin[1], OUTPUT);
    digitalWrite(digitPin[0], LOW);
    digitalWrite(digitPin[1], LOW);
}

void loop()
{
    for (int i = 0; i < 100; i += 2)
    {
        display_N(i);
        delay(1);
    }
}

void display_N(int num)
{
    int und = num % 10;
    int dec = (num % 100) / 10;
    for (int i = 0; i < 100; i++)
    {
        segOutput(1, und, 1);
        segOutput(0, dec, 1);
        delay(2);
    }

    Serial.print(dec);
    Serial.println(und);
}

// LED
void segClear()
{
    for (int i = 0; i < 8; i++)
    {
        digitalWrite(segPin[i], HIGH);
    }
}
// LED
void segOutput(int d, int Number, int dp)
```

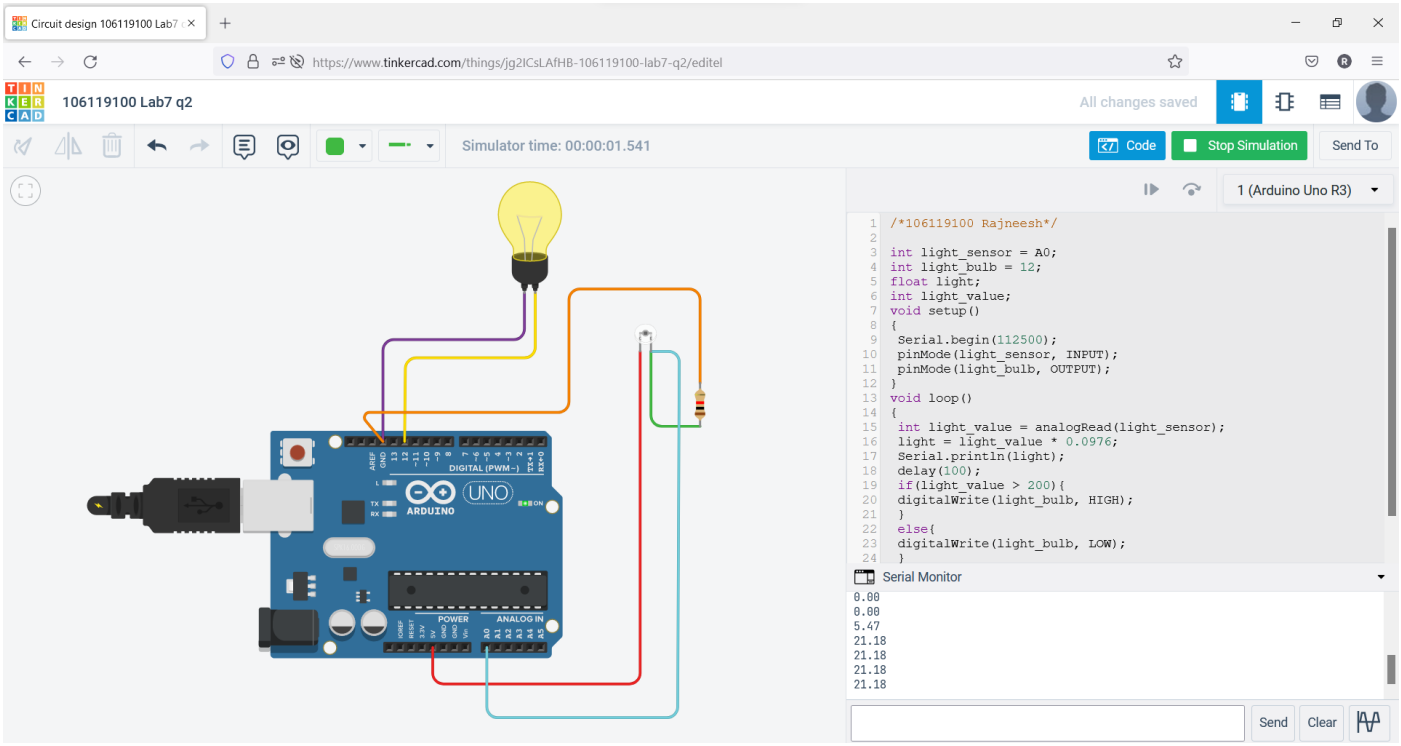
```
{
    segClear();
    digitalWrite(digitPin[d], HIGH);
    for (int i = 0; i < 7; i++)
    {
        digitalWrite(segPin[i], segValue[Number][i]);
    }
    digitalWrite(segPin[7], dp);
    delayMicroseconds(1000);
    digitalWrite(digitPin[d], LOW);
}
```



3. Interface the ambient light sensor with Arduino Uno board. Check the light value from the sensor and switch on/off the bulb (based on the threshold value).

```
/*106119100 Rajneesh*/

int light_sensor = A0;
int light_bulb = 12;
float light;
int light_value;
void setup()
{
    Serial.begin(112500);
```

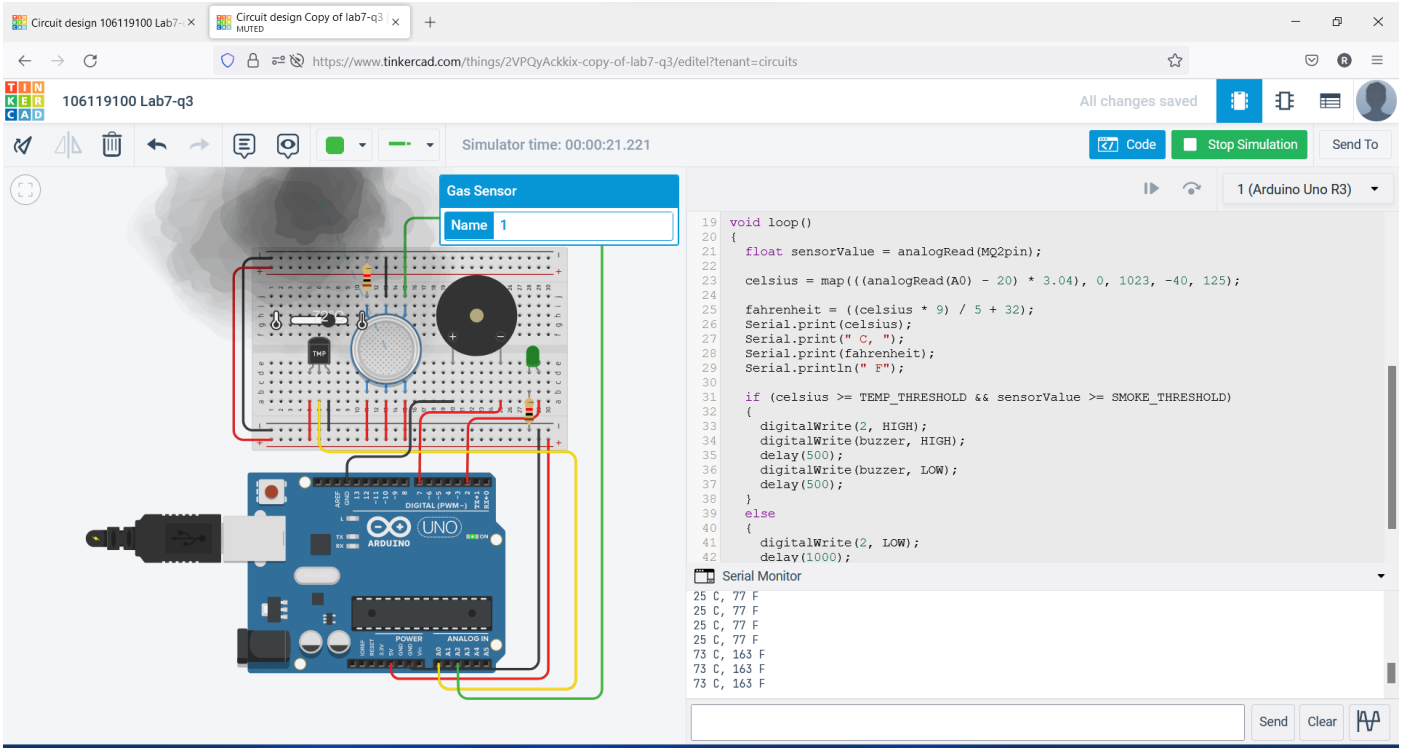
```
    pinMode(light_sensor, INPUT);
    pinMode(light_bulb, OUTPUT);
}
void loop()
{
    int light_value = analogRead(light_sensor);
    light = light_value * 0.0976;
    Serial.println(light);
    delay(100);
    if (light_value > 200)
    {
        digitalWrite(light_bulb, HIGH);
    }
    else
    {
        digitalWrite(light_bulb, LOW);
    }
}
```



4. Interface the temperature and gas sensor with Arduino Uno board. Check the temperature and the gas value, if the limit is beyond the threshold, switch on the bulb and make alarm using buzzer.

```
/*106119100 Rajneesh*/

int baselineTemp = 0;
int celsius = 0;
int fahrenheit = 0;
int buzzer = 7;
int MQ2pin = A2;
const int TEMP_THRESHOLD = 40, SMOKE_THRESHOLD = 300;

void setup()
{
    pinMode(A0, INPUT);
    Serial.begin(9600);

    pinMode(2, OUTPUT);
    pinMode(buzzer, OUTPUT);
}

void loop()
{
    float sensorValue = analogRead(MQ2pin);

    celsius = map(((analogRead(A0) - 20) * 3.04), 0, 1023, -40, 125);

    fahrenheit = ((celsius * 9) / 5 + 32);
    Serial.print(celsius);
    Serial.print(" C, ");
    Serial.print(fahrenheit);
    Serial.println(" F");

    if (celsius >= TEMP_THRESHOLD && sensorValue >= SMOKE_THRESHOLD)
    {
        digitalWrite(2, HIGH);
        digitalWrite(buzzer, HIGH);
        delay(500);
        digitalWrite(buzzer, LOW);
        delay(500);
    }
```

```
    else
    {
        digitalWrite(2, LOW);
        delay(1000);
    }
}
```



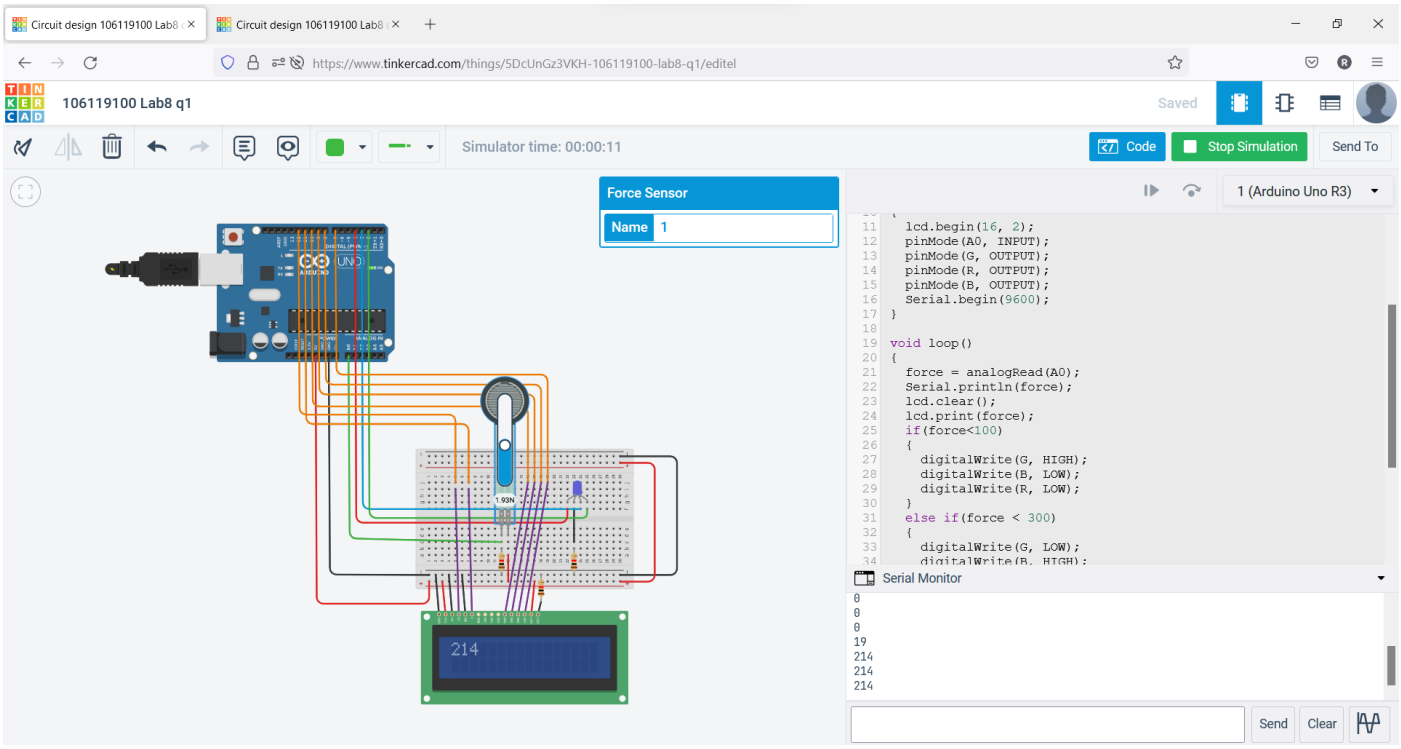
Lab 8

Date: 18-04-2022

1. Interface force sensor with Arduino board and display the amount of force given as input to sensor in LCD screen. Also, interface LED RGB and change colour of the LED based on some threshold Values.

```
#include <LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);
int force = 0;
int R = 4;
int B = 3;
int G = 2;
void setup()
{
    lcd.begin(16, 2);
    pinMode(A0, INPUT);
    pinMode(G, OUTPUT);
    pinMode(R, OUTPUT);
    pinMode(B, OUTPUT);
    Serial.begin(9600);
}
void loop()
{
    force = analogRead(A0);
    Serial.println(force);
    lcd.clear();
    lcd.print(force);
    if (force < 100)
    {
        digitalWrite(G, HIGH);
        digitalWrite(B, LOW);
        digitalWrite(R, LOW);
    }
    else if (force < 300)
    {
        digitalWrite(G, LOW);
        digitalWrite(B, HIGH);
        digitalWrite(R, LOW);
    }
    else
```

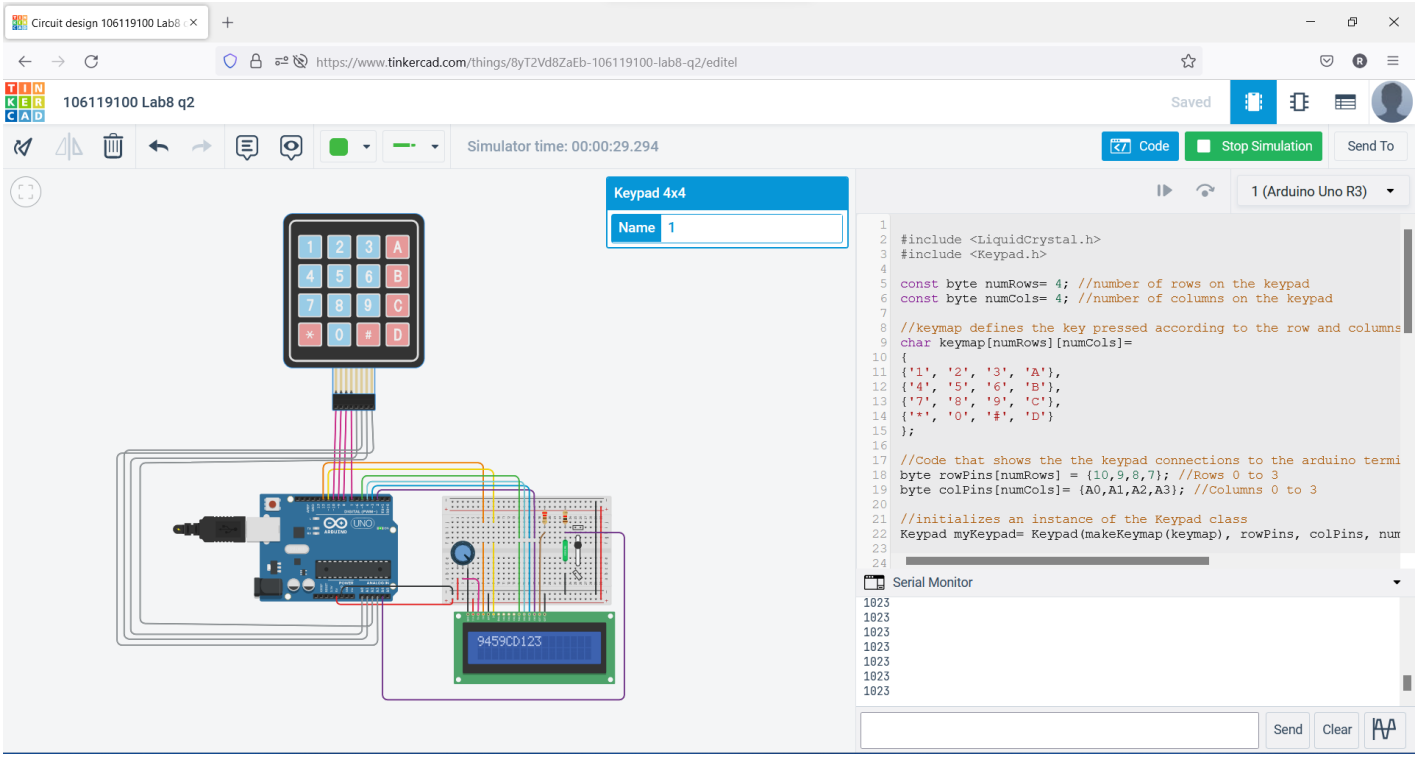
```
    {
        digitalWrite(R, HIGH);
        digitalWrite(G, LOW);
        digitalWrite(B, LOW);
    }
    delay(1000);
}
```

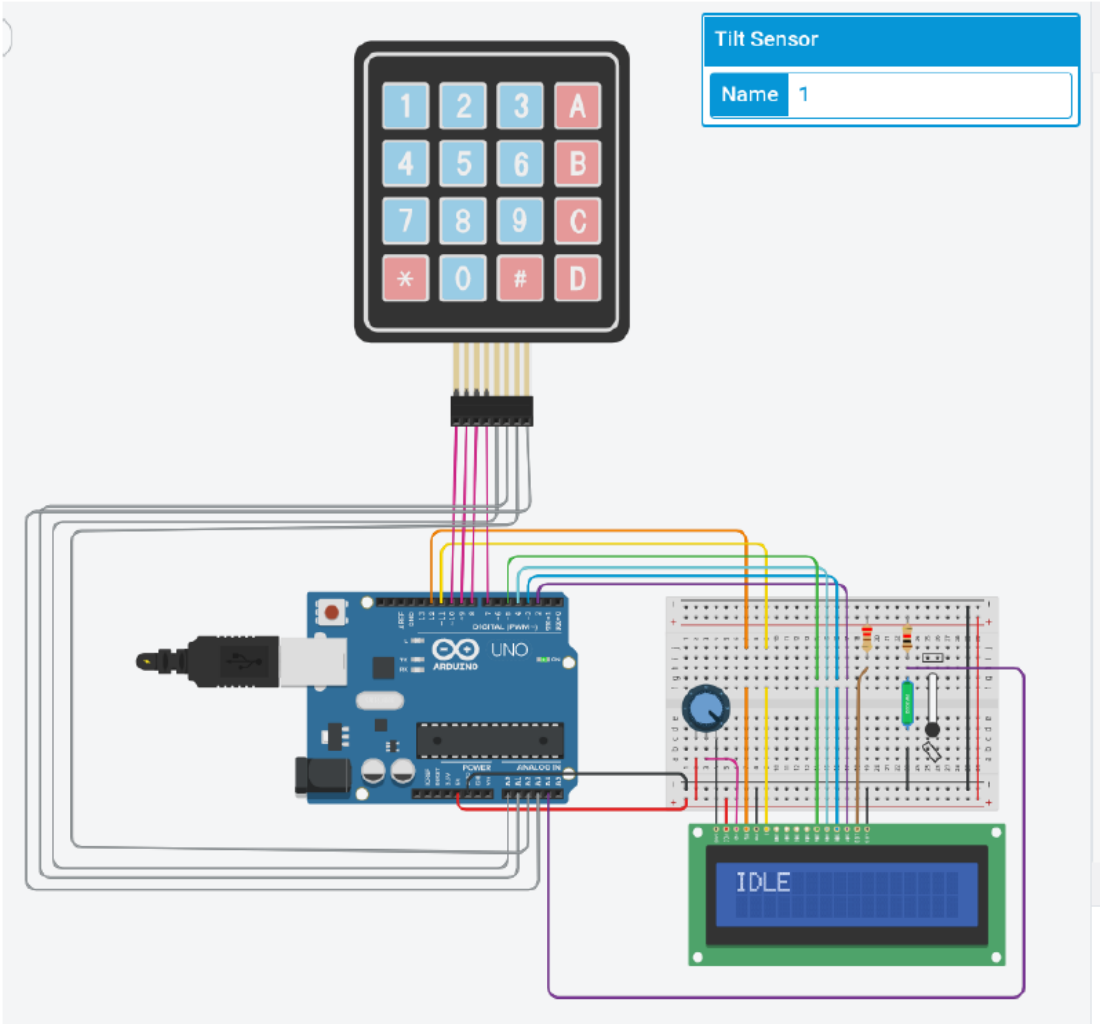


2. Interface the keypad and tilt sensor with Arduino board, if the sensor is being tilted, then take input from keypad and print it in the LCD

```
#include <LiquidCrystal.h>
#include <Keypad.h>

const byte numRows = 4; // number of rows on the keypad
const byte numCols = 4; // number of columns on the keypad

// keymap defines the key pressed according to the row and columns just as
appears on the keypad
char keymap[numRows][numCols] =
    {
        {'1', '2', '3', 'A'},
        {'4', '5', '6', 'B'},
        {'7', '8', '9', 'C'},
        {'*', '0', '#', 'D'}};
```

```
// Code that shows the the keypad connections to the arduino terminals
byte rowPins[numRows] = {10, 9, 8, 7};    // Rows 0 to 3
byte colPins[numCols] = {A0, A1, A2, A3}; // Columns 0 to 3

// initializes an instance of the Keypad class
Keypad myKeypad = Keypad(makeKeymap(keymap), rowPins, colPins, numRows,
numCols);

LiquidCrystal lcd(12, 11, 5, 4, 3, 2);
int tilt = A4;

char buf[16] = {0};
int cnt = 0;

void setup()
{
    Serial.begin(9600);
    lcd.begin(16, 2);
    pinMode(tilt, INPUT);
}

void loop()
{
    int reading = analogRead(tilt);

    char keypressed = myKeypad.getKey();

    Serial.println(reading);

    if (reading > 100)
    {

        buf[cnt] = keypressed;
        if (keypressed != 0)
            cnt++;
        cnt %= 16;
        buf[cnt] = 0;
        lcd.clear();
        lcd.print(buf);
    }
    else
    {
        lcd.clear();
        lcd.print("IDLE");
    }

    delay(50);
}
```


Circuit design 106119100 Lab8
https://www.tinkercad.com/things/8yT2Vd8ZaEb-106119100-lab8-q2/editel
106119100 Lab8 q2
Saved
Simulator time: 00:00:29.294
Code
Stop Simulation
Send To
Keypad 4x4
Name 1
1 (Arduino Uno R3)
#include <LiquidCrystal.h>
#include <Keypad.h>
const byte numRows= 4; //number of rows on the keypad
const byte numCols= 4; //number of columns on the keypad
//keymap defines the key pressed according to the row and columns
char keymap[numRows][numCols]=
{
{'1', '2', '3', 'A'},
{'4', '5', '6', 'B'},
{'7', '8', '9', 'C'},
{'*', '0', '#', 'D'}
};
//Code that shows the the keypad connections to the arduino termi
byte rowPins[numRows] = {10,9,8,7}; //Rows 0 to 3
byte colPins[numCols]= {A0,A1,A2,A3}; //Columns 0 to 3
//initializes an instance of the Keypad class
Keypad myKeypad= Keypad(makeKeymap(keymap), rowPins, colPins, num
Serial Monitor
1023
1023
1023
1023
1023
1023
1023
Send
Clear
9459CD123


Tilt Sensor
Name 1
1 2 3 A
4 5 6 B
7 8 9 C
* 0 # D
UNO
ARDUINO
IDLE