

Embedded Systems Lab Report

*Embedded Systems Lab
(CSLR 61)*

Dipesh Kafle

Roll No: 106119029
Class: CSE A

30 April 2022

Contents

Lab 1	Date: 31-01-2022	5
1 Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.	5	
2 Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.	6	
3 Blink LEDs – for all composite number below 15.	7	
4 Blink LEDs – to count even numbers	8	
Lab 2	Date: 07-02-2022	10
5 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.	10	
6 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.	11	
7 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.	12	
Lab 3	Date: 14-02-2022	15
8 Increase the intensity of one LED while decreasing the other LED's intensity at the same unit. Display the intensity value of the LEDs.	15	
9 Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED.	16	
10 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.	18	
Lab 4	Date: 28-02-2022	20
11 Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker	20	
12 Design an alarm system, which makes double beep sound for every 5s using a speaker	21	
13 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.	23	
14 Design a program to make ambulance sound.	24	
Lab 5	Date: 07-03-2022	25
15 Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.	25	
16 Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.	27	
17 Blink LEDs – for all composite number below 15.	31	
18 Blink LEDs – to count even numbers	36	
Lab 6	Date: 28-03-2022	42
19 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.	42	
20 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.	45	
Lab 7	Date: 11-04-2022	49
21 Interface two 7-segment display with Arduino Uno board. Implement the a)odd counter and b)even counter and display the values in the seven segment display.	49	
22 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).	53	
23 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.	55	
Lab 8	Date: 18-04-2022	59

24 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 color of the LED based on the some threshold values.	59
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.	62

S.N	Topic	Date	Lab
1	Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.	31-01-2022	1
2	Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.	31-01-2022	1
3	Blink LEDs – for all composite number below 15.	31-01-2022	1
4	Blink LEDs – to count even numbers	31-01-2022	1
5	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	07-02-2022	2
6	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	07-02-2022	2
7	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	07-02-2022	2
8	Increase the intensity of one LED while decreasing the other LED's intensity at the same unit. Display the intensity value of the LEDs. Libraries Used: pwmout	14-02-2022	3
9	Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED. Libraries To Be Used: pwmout, analogin	14-02-2022	3
10	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.	14-02-2022	3
11	Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker. Libraries Used: pwmout	28-02-2022	4
12	Design an alarm system, which makes double beep sound for every 5s using a speaker. Libraries To Be Used: pwmout, analogin	28-02-2022	4
13	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.	28-02-2022	4
14	Design a program to make ambulance sound.	28-02-2022	4
15	Blink LEDs in alternate order – 1 and 3 together and 2 and 4 together.	07-03-2022	5
16	Blink LEDs – count from 1 to 15; if the board is counting odd value, wait for 1 sec, else wait for 2 sec.	07-03-2022	5
17	Blink LEDs – for all composite number below 15.	07-03-2022	5
18	Blink LEDs – to count even numbers	07-03-2022	5
19	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.	28-03-2022	6
20	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	28-03-2022	6
21	Interface two 7-segment display with Arduino Uno board. Implement the a)odd counter and b)even counter and display the values in the seven segment display.	11-04-2022	7
22	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).	11-04-2022	7
23	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.	11-04-2022	7
24	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 color of the LED based on the some threshold values.	18-04-2022	8

S.N	Topic	Date	Lab
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.	18-04-2022	8

Lab 1

Date: 31-01-2022

Questions

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

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

Question

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

Aim

To blink LEDs in alternate order – 1 and 3 should blink together and 2 and 4 should blink together.

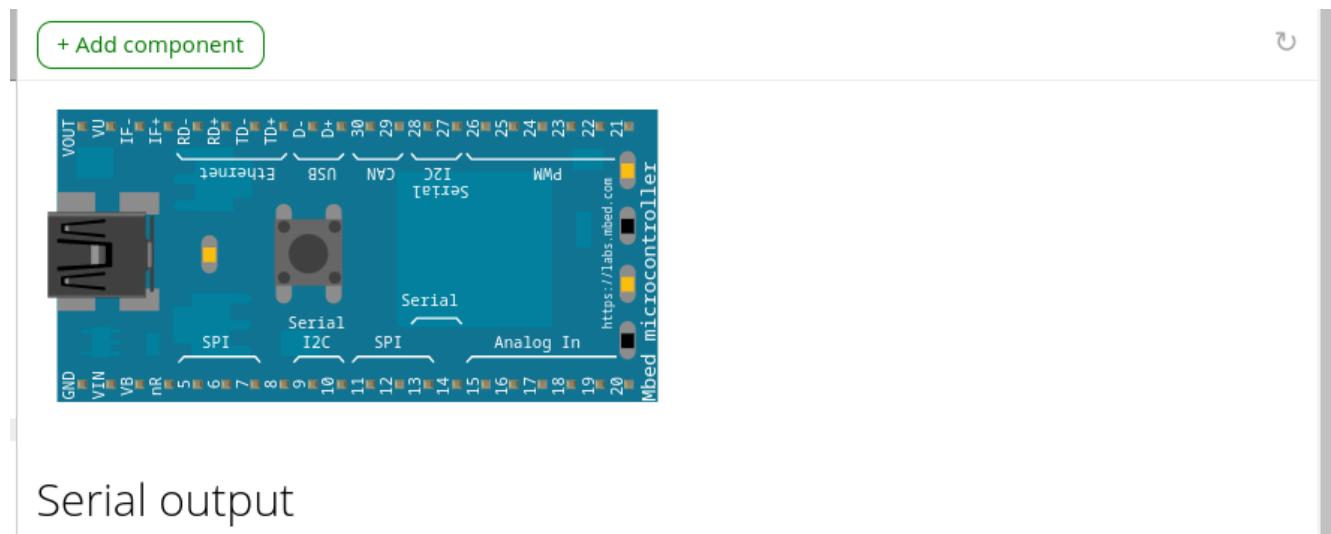
Code

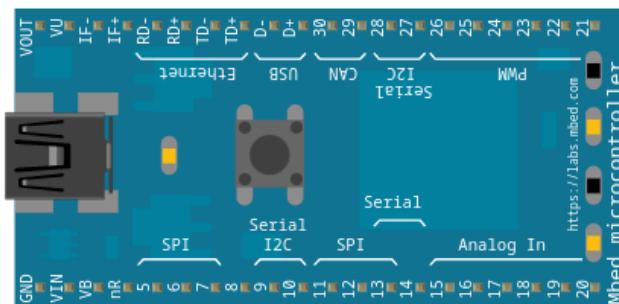
```
#include "mbed.h"

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

int main() {
    while (1) {
        // 1010
        // 0101
        int arr[] = {10, 5};
        for (int i = 0; i < sizeof(arr) / sizeof(int); i++) {
            myleds = arr[i];
            wait(1);
        }
    }
}
```

Output





Serial output

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

Question

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

Aim

To blink LEDs and count from 1 to 15. If the board is counting odd value, it should wait for 1 sec, else it should wait for 2 sec.

Code

```
#include "mbed.h"

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

int main() {
    while (1) {
        for (int i = 0; i < 16; i++) {
            myleds = i;
            if (i % 2 == 1) {
                printf("Waiting 1s for %d\n", i);
                wait(1);
            } else {
                printf("Waiting 2s for %d\n", i);
                wait(2);
            }
        }
    }
}
```

Output

The screenshot shows the arm MBED IDE interface. On the left, the code editor displays the 'Blinky' demo code. The code includes a loop that iterates from 0 to 15, printing the current value and waiting 2 seconds if it's a composite number. On the right, the component palette shows various pins and peripherals like SPI, I2C, Serial, and Analog In. Below the palette, a 'Serial output' window shows the printed text.

```

1 #include "mbed.h"
2
3 BusOut myleds(LED1, LED2, LED3, LED4);
4
5 int main() {
6     while (1) {
7         for (int i = 0; i < 16; i++) {
8             myleds(i);
9             if ((i % 2 == 1) {
10                 printf("Waiting %d for %d\n", i);
11                 wait(2);
12             } else {
13                 printf("Waiting 2s for %d\n", i);
14                 wait(2);
15             }
16         }
17     }
18 }

```

Serial output

```

Waiting 2s for 8
Waiting 1s for 9
Waiting 2s for 10
Waiting 1s for 11
Waiting 2s for 12
Waiting 1s for 13
Waiting 2s for 14
Waiting 1s for 15
Waiting 2s for 0
Waiting 1s for 1
Waiting 2s for 2
Waiting 1s for 3
Waiting 2s for 4
Waiting 1s for 5
Waiting 2s for 6
Waiting 1s for 7
Waiting 2s for 8
Waiting 1s for 9
Waiting 2s for 10
Waiting 1s for 11
Waiting 2s for 12
Waiting 1s for 13
Waiting 2s for 14

```

3 Blink LEDs – for all composite number below 15.

Question

Blink LEDs – for all composite number below 15.

Aim

To blink LEDs for all composite numbers below 15.

Code

```

#include "mbed.h"

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

bool isComposite(int n) {
    if (n == 1)
        return false;
    for (int i = 2; i * 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)) {
        printf("%d\n", i);
        myleds = i;
        wait(1);
    }
}
```

Output

4 Blink LEDs – to count even numbers

Question

Blink LEDs – to count even numbers

Aim

To blink LEDs to count even numbers

Code

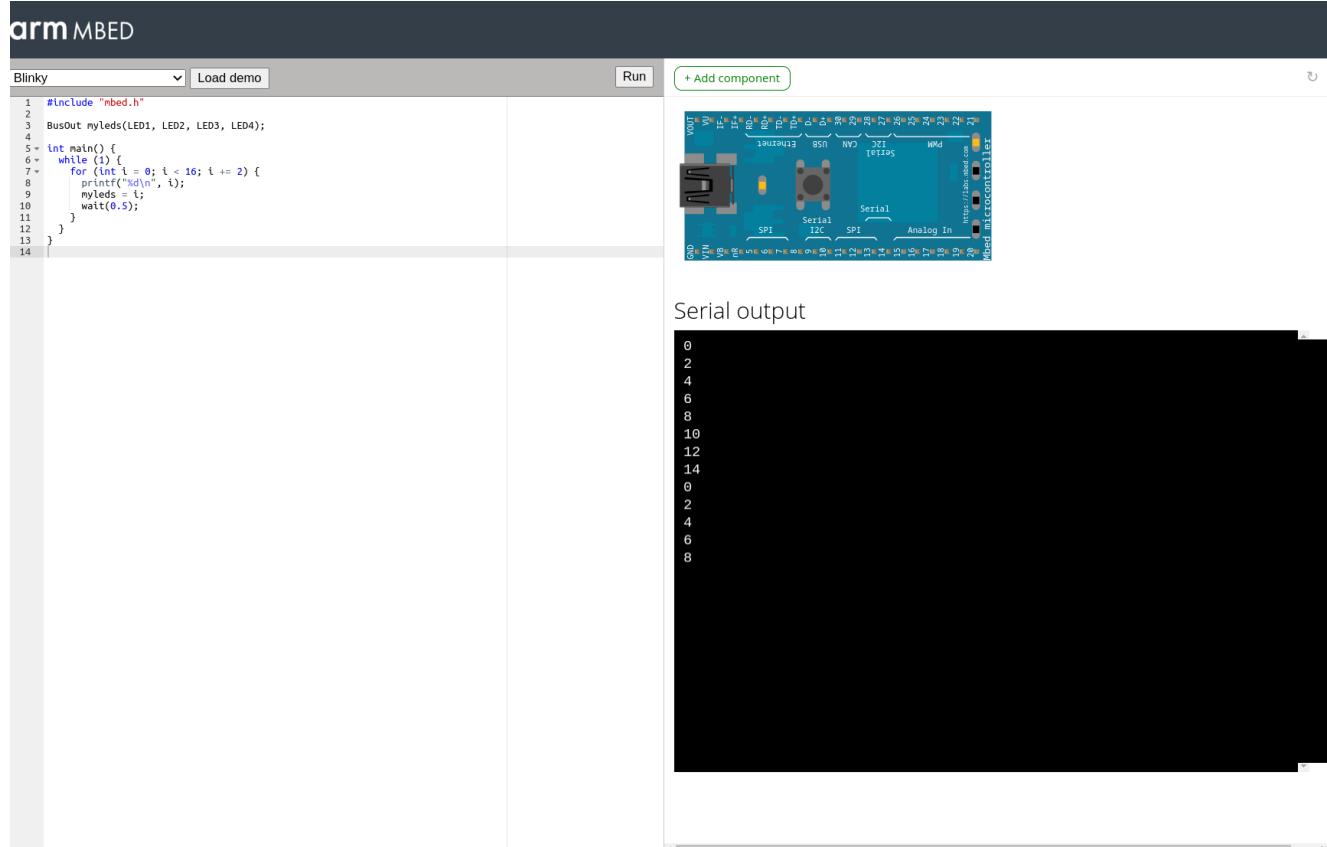
```
#include "mbed.h"
```

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

```
int main() {  
    while (1) {
```

```
for (int i = 0; i < 16; i += 2) {  
    printf("%d\n", i);  
    myleds = i;  
    wait(0.5);  
}  
}  
}
```

Output



Lab 2

Date: 07-02-2022

Questions

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
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
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

5 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.

Question

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

Aim

To switch between hex counter and decade counter using switches. It must display the current value of the counter with the help of 4 LEDs and the screen.

Code

```
#include "mbed.h"
BusOut myleds(LED1, LED2, LED3, LED4);
InterruptIn button(p5);
void hex_counter() {
    printf("button Raised");
    while (1) {
        for (int i = 0; i < 16; i++) {
            myleds = i;
            printf("Hex counter: %d\n", i);
            wait(0.5);
        }
    }
}
void dec_counter() {
    printf("button Fallen");
    while (1) {
        for (int i = 0; i < 10; i++) {
            myleds = i;
            printf("Decade counter: %d\n", i);
            wait(0.5);
        }
    }
}
int main() {
    button.rise(&hex_counter);
    button.fall(&dec_counter);
}
```

Output

The screenshot shows the mbed IDE interface. On the left, the code for 'Blinky' is displayed:

```

1 #include "mbed.h"
2 BusOut myleds(LED1, LED2, LED3, LED4);
3 InterruptIn button(p5);
4 void hex_counter() {
5     printf("button Raised");
6     while (1) {
7         for (int i = 0; i < 16; i++) {
8             myleds = i;
9             printf("Hex counter: %d\n", i);
10            wait(0.5);
11        }
12    }
13 }
14 void dec_counter() {
15     printf("button Fallen");
16     while (1) {
17         for (int i = 0; i < 10; i++) {
18             myleds = i;
19             printf("Decade counter: %d\n", i);
20             wait(0.5);
21         }
22     }
23 }
24 int main() {
25     button.rise(&hex_counter);
26     button.fall(&dec_counter);
27 }

```

On the right, the component palette shows a 'Switch (p5)' component, which is a digital input component labeled 'p5'.

The 'Serial output' window displays the following text:

```

button FallenDecade counter: 0
Decade counter: 1
Decade counter: 2
Decade counter: 3
Decade counter: 4
Decade counter: 5
Decade counter: 6
Decade counter: 7
Decade counter: 8
Decade counter: 9
button RaisedHex counter: 0
Hex counter: 1
Hex counter: 2
Hex counter: 3
Hex counter: 4
Hex counter: 5
Hex counter: 6
Hex counter: 7
Hex counter: 8
Hex counter: 9
Hex counter: 10

```

6 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.

Question

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

Aim

To display the octal counter using the onboard LEDs using Ticker object. After 10 second, it must reset all the values of the onboard LED using timeout Object.

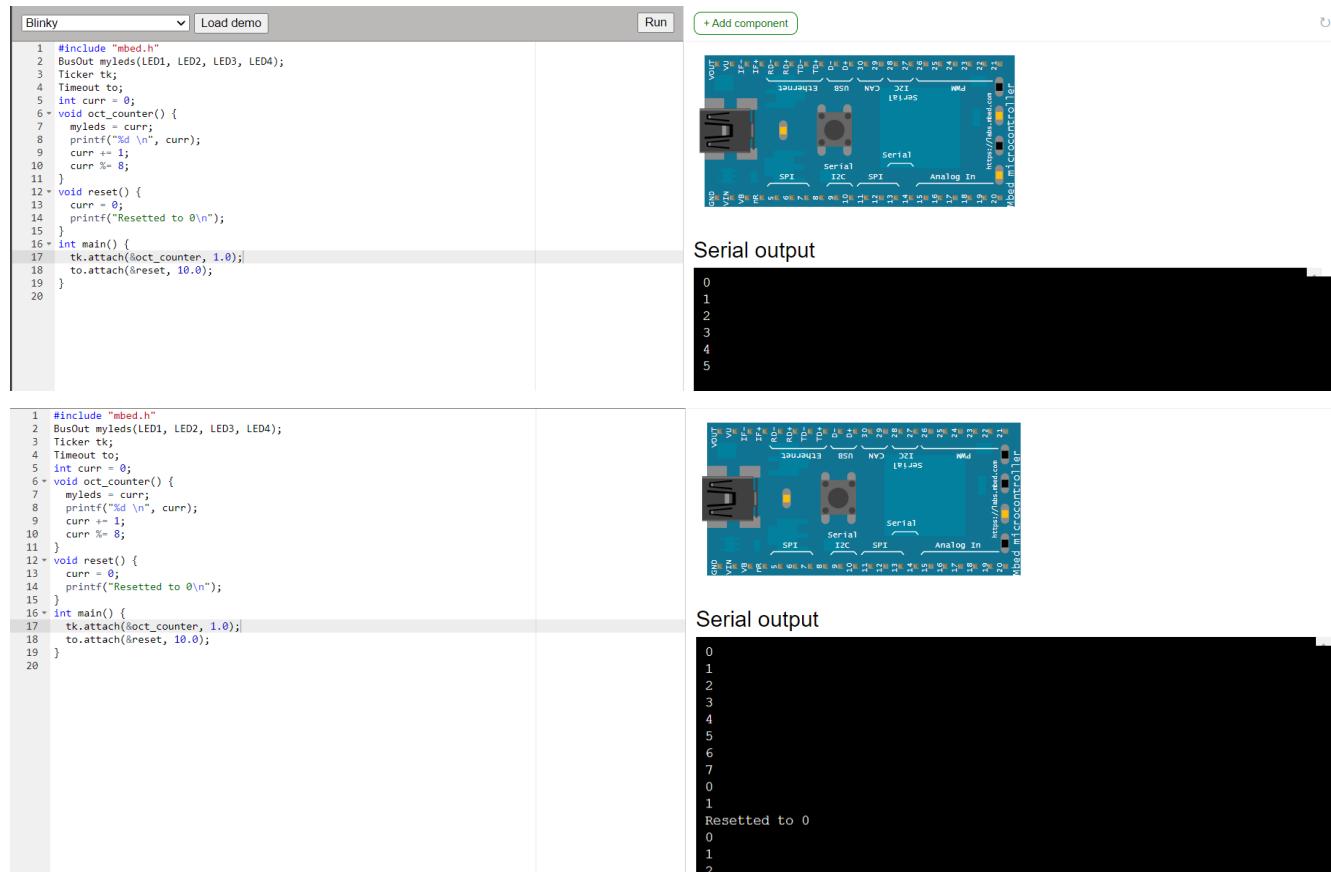
Code

```

#include "mbed.h"
BusOut myleds(LED1, LED2, LED3, LED4);
Ticker tk;
Timeout to;
int curr = 0;
void oct_counter() {
    myleds = curr;
    printf("%d \n", curr);
    curr += 1;
    curr %= 8;
}
void reset() {
    curr = 0;
    printf("Resetted to 0\n");
}
int main() {
    tk.attach(&oct_counter, 1.0);
    to.attach(&reset, 10.0);
}

```

Output



7 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.

Question

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

Aim

To display the odd counter using the onboard LEDs and screen. After 10 seconds, it reset all the values of the onboard LED and display even counter using Timeout Object. Switch should be there to manually toggle between two counters as well.

Code

```
#include "mbed.h"
BusOut myleds(LED1, LED2, LED3, LED4);
InterruptIn btn(p5);
Ticker tk;
Timeout to;
int odd = 1;
const char *cntr_type = "Odd";
int val = 1;
void counter() {
```

```

myleds = val;
printf("%s: %d \n", cntr_type, val);
val += 2;
val %= 16;
}
void change_type() {
printf("changed type using button \n");
if (odd) {
    odd = 0;
    val = 0;
    cntr_type = "Even";
} else {
    odd = 1;
    val = 1;
    cntr_type = "Odd";
}
}
void reset() {
printf("reseting the values after timeout \n");
odd = 0;
val = 0;
cntr_type = "Even";
}
int main() {
tk.attach(&counter, 1.0);
btn.rise(&change_type);
btn.fall(&change_type);
to.attach(&reset, 10.0);
}

```

Output

The screenshot shows the Microcontroller IDE interface. On the left, the code editor displays the provided C++ code for a microcontroller. On the right, the component palette lists various hardware components like BusOut, InterruptIn, Ticker, Timeout, and Switch (p5). Below the palette, the serial output window shows the following text:

```

Serial output
Odd: 1
Odd: 3
Odd: 5
Odd: 7
Odd: 9

```

Blinky Load demo Run + Add component

```

1 #include "mbed.h"
2 BusOut myleds(LED1, LED2, LED3, LED4);
3 InterruptIn btn(p5);
4 Ticker tk;
5 Timeout to;
6 int odd = 1;
7 const char *cntn_type = "Odd";
8 int val = 1;
9 void counter() {
10     myleds = val;
11     printf("%s: %d \n", cntn_type, val);
12     val += 2;
13     val %= 16;
14 }
15 void change_type() {
16     printf("changed type using button \n");
17     if (odd) {
18         odd = 0;
19         val = 0;
20         cntn_type = "Even";
21     } else {
22         odd = 1;
23         val = 1;
24         cntn_type = "Odd";
25     }
26 }
27 void reset() {
28     printf("resetting the values after timeout \n");
29     odd = 0;
30     val = 0;
31     cntn_type = "Even";
32 }
33 int main() {
34     tk.attach(&counter, 1.0);
35     btn.rise(&change_type);
36     btn.fall(&change_type);
37     to.attach(&reset, 10.0);
38 }
39

```

Serial output

```

Odd: 1
Odd: 3
Odd: 5
Odd: 7
Odd: 9
Odd: 11
Odd: 13
Odd: 15
Odd: 1
Odd: 3
resetting the values after timeout
Even: 0
Even: 2
Even: 4
Even: 6
Even: 8
Even: 10
Even: 12

```

Blinky Load demo Run + Add component

```

1 #include "mbed.h"
2 BusOut myleds(LED1, LED2, LED3, LED4);
3 InterruptIn btn(p5);
4 Ticker tk;
5 Timeout to;
6 int odd = 1;
7 const char *cntn_type = "Odd";
8 int val = 1;
9 void counter() {
10     myleds = val;
11     printf("%s: %d \n", cntn_type, val);
12     val += 2;
13     val %= 16;
14 }
15 void change_type() {
16     printf("changed type using button \n");
17     if (odd) {
18         odd = 0;
19         val = 0;
20         cntn_type = "Even";
21     } else {
22         odd = 1;
23         val = 1;
24         cntn_type = "Odd";
25     }
26 }
27 void reset() {
28     printf("resetting the values after timeout \n");
29     odd = 0;
30     val = 0;
31     cntn_type = "Even";
32 }
33 int main() {
34     tk.attach(&counter, 1.0);
35     btn.rise(&change_type);
36     btn.fall(&change_type);
37     to.attach(&reset, 10.0);
38 }
39

```

Serial output

```

Even: 8
Even: 10
Even: 12
Even: 14
Even: 0
Even: 2
Even: 4
Even: 6
changed type using button
Odd: 1
Odd: 3
changed type using button
Even: 0
Even: 2
Even: 4
Even: 6
Even: 8
Even: 10
Even: 12

```

Lab 3

Date: 14-02-2022

Questions

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. Libraries Used: pwmout
2. Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED. Libraries To Be Used: pwmout, analogin
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.

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

Question

Increase the intensity of one LED while decreasing the other LED's intensity at the same unit. Display the intensity value of the LEDs. Libraries Used: pwmout

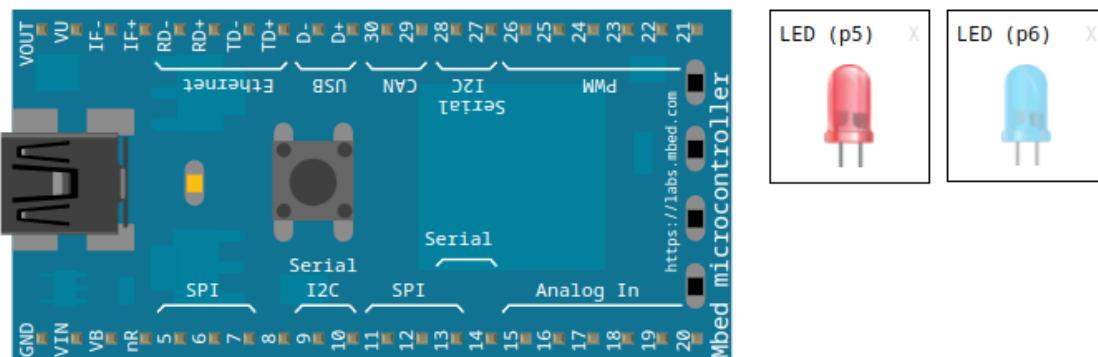
Aim

To increase the intensity of one LED while decreasing the other LED's intensity at the same unit. We must also display the intensity value of the LEDs.

Code

```
#include "mbed.h"
PwmOut led1(p5);
PwmOut led2(p6);
int main() {
    led2 = 1;
    led1 = 0;
    while (1) {
        led1 = led1 + 0.10;
        led2 = led2 - 0.10;
        printf("LED1 is now %.2f\n", led1.read());
        printf("LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1.0) {
            led1 = 0;
        }
        if (led2 == 0) {
            led2 = 1;
        }
    }
}
```

Output



Serial output

```
LED2 is now 0.30
LED1 is now 0.80
LED2 is now 0.20
LED1 is now 0.90
LED2 is now 0.09
LED1 is now 1.00
LED2 is now 0.00
LED1 is now 1.00
LED2 is now 0.90
LED1 is now 0.10
LED2 is now 0.80
LED1 is now 0.20
LED2 is now 0.70
LED1 is now 0.30
LED2 is now 0.60
LED1 is now 0.40
LED2 is now 0.50
LED1 is now 0.50
LED2 is now 0.40
LED1 is now 0.60
LED2 is now 0.30
LED1 is now 0.70
LED2 is now 0.20
```

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

Question

Interface potentiometer with mbed board and based on the value of potentiometer, adjust the intensity of an external LED. Libraries To Be Used: pwmout, analogin

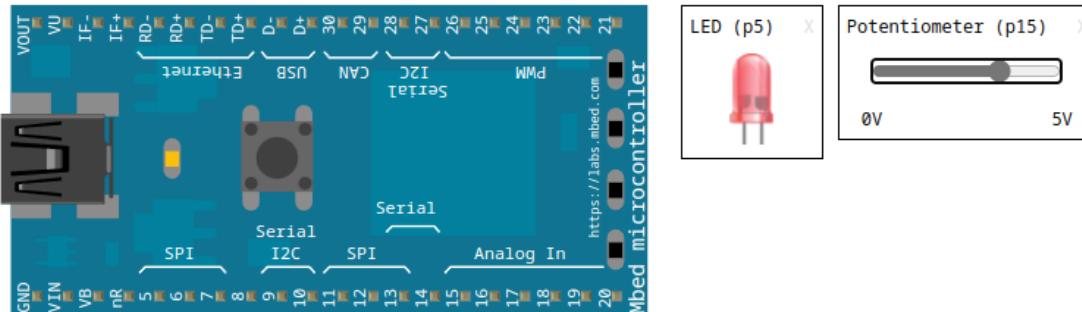
Aim

To interface potentiometer with mbed board and based on the value of potentiometer and adjust the intensity of an external LED.

Code

```
#include "mbed.h"
PwmOut led(p5);
AnalogIn p(p15);
int main() {
    while (1) {
        led = p.read();
        printf("LED is now %.2f\n", led.read());
        wait(0.5);
    }
}
```

Output



Serial output

```
LED is now 0.00
LED is now 0.69
LED is now 0.69
LED is now 0.69
LED is now 0.69
```

10 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.

Question

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.

Aim

To interface switch, 2 LEDs, potentiometer with the mbed board. If the switch is on, we must 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, we must blink LEDs one at increasing rate and other at decreasing rate (same rate) and loop back.

Code

```
#include "mbed.h"
PwmOut led1(p5);
PwmOut led2(p6);
InterruptIn btn(p7);
AnalogIn p(p15);
#define MAX(x, y) x > y ? x : y
#define MIN(x, y) x > y ? y : x
void inc_dec_by_potentio() {
    led1 = 0;
    led2 = 1;
    while (1) {
        float rate = p.read();
        led1 = MIN(led1 + rate, 1);
        led2 = MAX(led2 - rate, 0);
        printf("Potentiometer: LED1 is now %.2f\n", led1.read());
        printf("Potentiometer: LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1) {
            led1 = 0;
        }
        if (led2 == 0) {
            led2 = 1;
        }
    }
}
void inc_dec() {
    led1 = 0;
    led2 = 1;
    while (1) {
        led1 = led1 + 0.10;
        led2 = led2 - 0.10;
        printf("LED1 is now %.2f\n", led1.read());
        printf("LED2 is now %.2f\n", led2.read());
        wait(1);
        if (led1 == 1.0) {
```

```

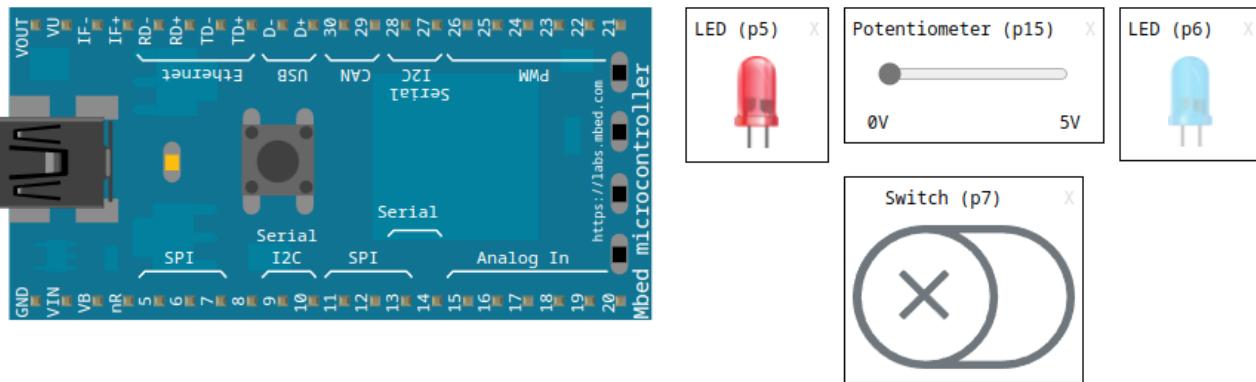
    led1 = 0;
}
if (led2 == 0) {
    led2 = 1;
}
}
}

int main() {
    btn.rise(inc_dec_by_potentio);
    btn.fall(inc_dec);
}

// 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.

```

Output



Serial output

```

LED1 is now 0.10
LED2 is now 0.90
LED1 is now 0.20
LED2 is now 0.80
LED1 is now 0.30
LED2 is now 0.70
LED1 is now 0.40
LED2 is now 0.60
LED1 is now 0.50
LED2 is now 0.50
LED1 is now 0.60
LED2 is now 0.40
LED1 is now 0.70
LED2 is now 0.30
LED1 is now 0.80
LED2 is now 0.20
LED1 is now 0.90
LED2 is now 0.09

```

Lab 4**Date: 28-02-2022****Questions**

1. Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker. Libraries Used: pwmout
2. Design an alarm system, which makes double beep sound for every 5s using a speaker. Libraries To Be Used: pwmout, analogin
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.

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

Question

Write a program to increase the intensity of external LED. At the same rate, increase the sound intensity of speaker. Libraries Used: pwmout

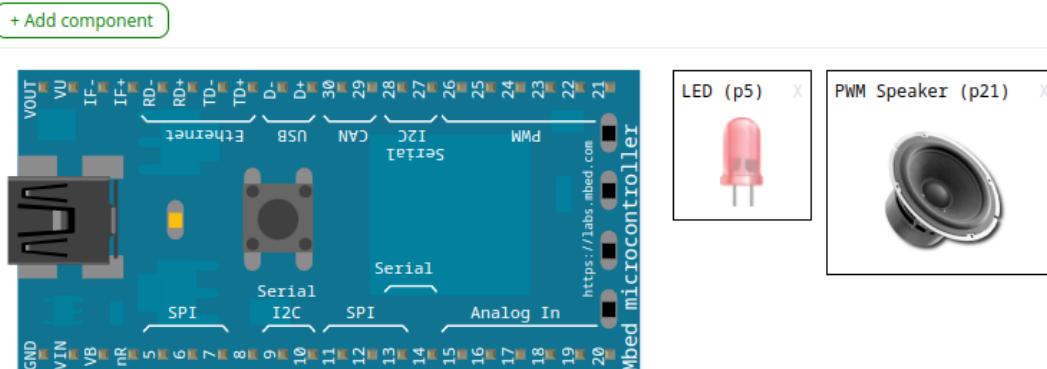
Aim

To write a program to increase the intensity of external LED. At the same rate, we must increase the sound intensity of speaker.

Code

```
#include "mbed.h"
PwmOut led(p5);
PwmOut speaker(p21);
int main() {
    int inc = 1;
    while (1) {
        led = led + inc * 0.10;
        speaker = speaker + inc * 0.10;
        printf("LEDs are now %.2f, and speaker is %.2f\n", led.read(),
               speaker.read());
        wait(0.5);
        if (led == 1.0) {
            inc = -1;
        }
        if (led == 0) {
            inc = 1;
        }
    }
}
```

Output



Serial output

```
LEDs are now 0.10, and speaker is 0.10
LEDs are now 0.20, and speaker is 0.20
LEDs are now 0.30, and speaker is 0.30
LEDs are now 0.40, and speaker is 0.40
LEDs are now 0.50, and speaker is 0.50
LEDs are now 0.60, and speaker is 0.60
LEDs are now 0.70, and speaker is 0.70
LEDs are now 0.80, and speaker is 0.80
LEDs are now 0.90, and speaker is 0.90
LEDs are now 1.00, and speaker is 1.00
LEDs are now 1.00, and speaker is 1.00
LEDs are now 0.90, and speaker is 0.90
LEDs are now 0.80, and speaker is 0.80
LEDs are now 0.70, and speaker is 0.70
LEDs are now 0.60, and speaker is 0.60
LEDs are now 0.50, and speaker is 0.50
LEDs are now 0.40, and speaker is 0.40
```

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

Question

Design an alarm system, which makes double beep sound for every 5s using a speaker. Libraries To Be Used: pwmout, analogin

Aim

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

Code

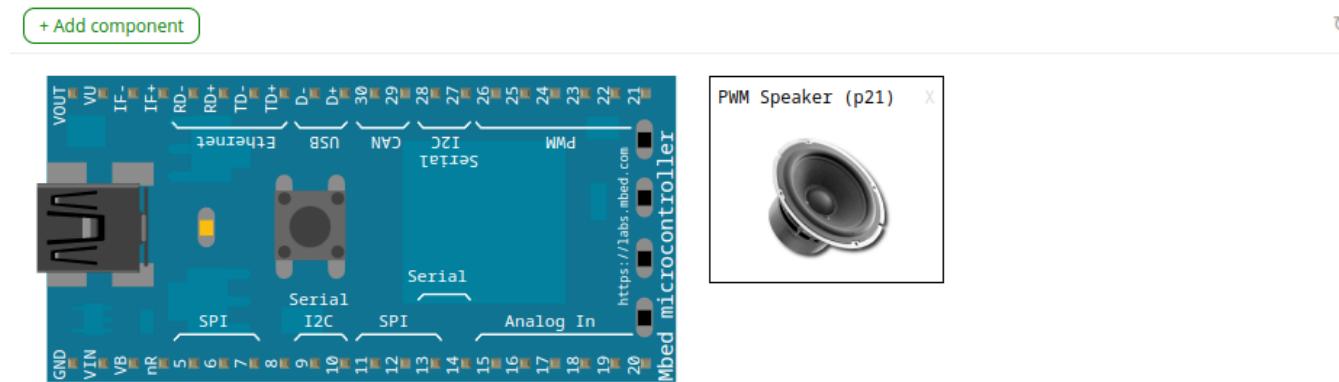
```
#include "mbed.h"
PwmOut speaker(p21);
int main() {
```

```

while (1) {
    // Can use ticker as well
    wait_ms(5000);
    printf("Speaker will beep twice now \n\n");
    speaker = 0.5;
    wait_ms(50);
    printf("Speaker: %.2f\n", speaker.read());
    speaker = 0;
    wait_ms(50);
    speaker = 0.5;
    wait_ms(50);
    printf("Speaker: %.2f\n", speaker.read());
    speaker = 0;
}
}

```

Output



Serial output

```

Speaker will beep twice now
Speaker: 0.50
Speaker: 0.50
Speaker will beep twice now

Speaker: 0.50
Speaker: 0.50
Speaker will beep twice now

Speaker: 0.50
Speaker: 0.50

```

13 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.

Question

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.

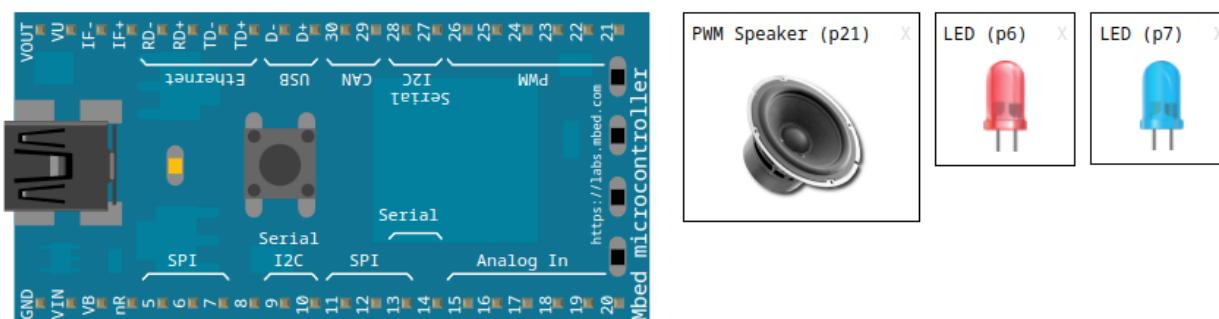
Aim

To interface a switch, 2 LEDs and a speaker with mbed board. Upon switching on, the intensity of the sound is to be increased and upon switching off, the intensity of the sound is to be decreased. The intensity of two LEDs is to be proportionally increased or decreased.

Code

```
#include "mbed.h"
PwmOut speaker(p21);
PwmOut led1(p6);
PwmOut led2(p7);
float delta = -0.1f;
InterruptIn btn(p5);
void pos() { delta = 0.1f; }
void neg() { delta = -0.1f; }
int main() {
    btn.rise(&pos);
    btn.fall(&neg);
    speaker = 0;
    while (1) {
        if (delta > 0 && speaker >= 1)
            speaker = 0;
        if (delta < 0 && speaker <= 0)
            speaker = 1;
        speaker = speaker + delta;
        led1 = speaker;
        led2 = speaker;
        wait(0.5);
    }
}
```

Output



14 Design a program to make ambulance sound.

Question

Design a program to make ambulance sound.

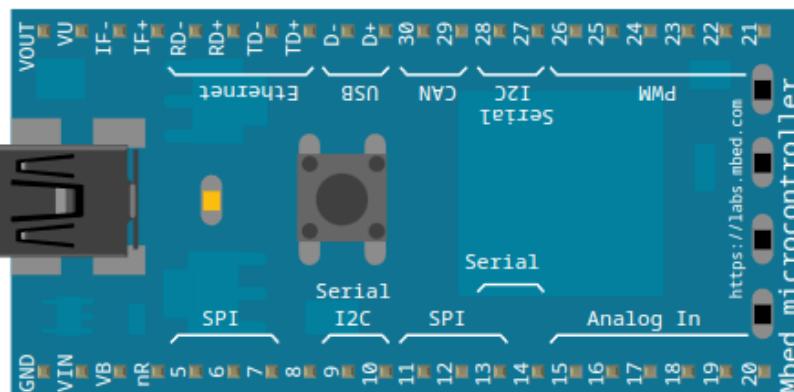
Aim

To design a program which will output ambulance sound.

Code

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

Output



Lab 5**Date: 07-03-2022****Questions**

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

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

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

Aim

To blink LEDs in alternate order – 1 and 3 should blink together and 2 and 4 should blink together.

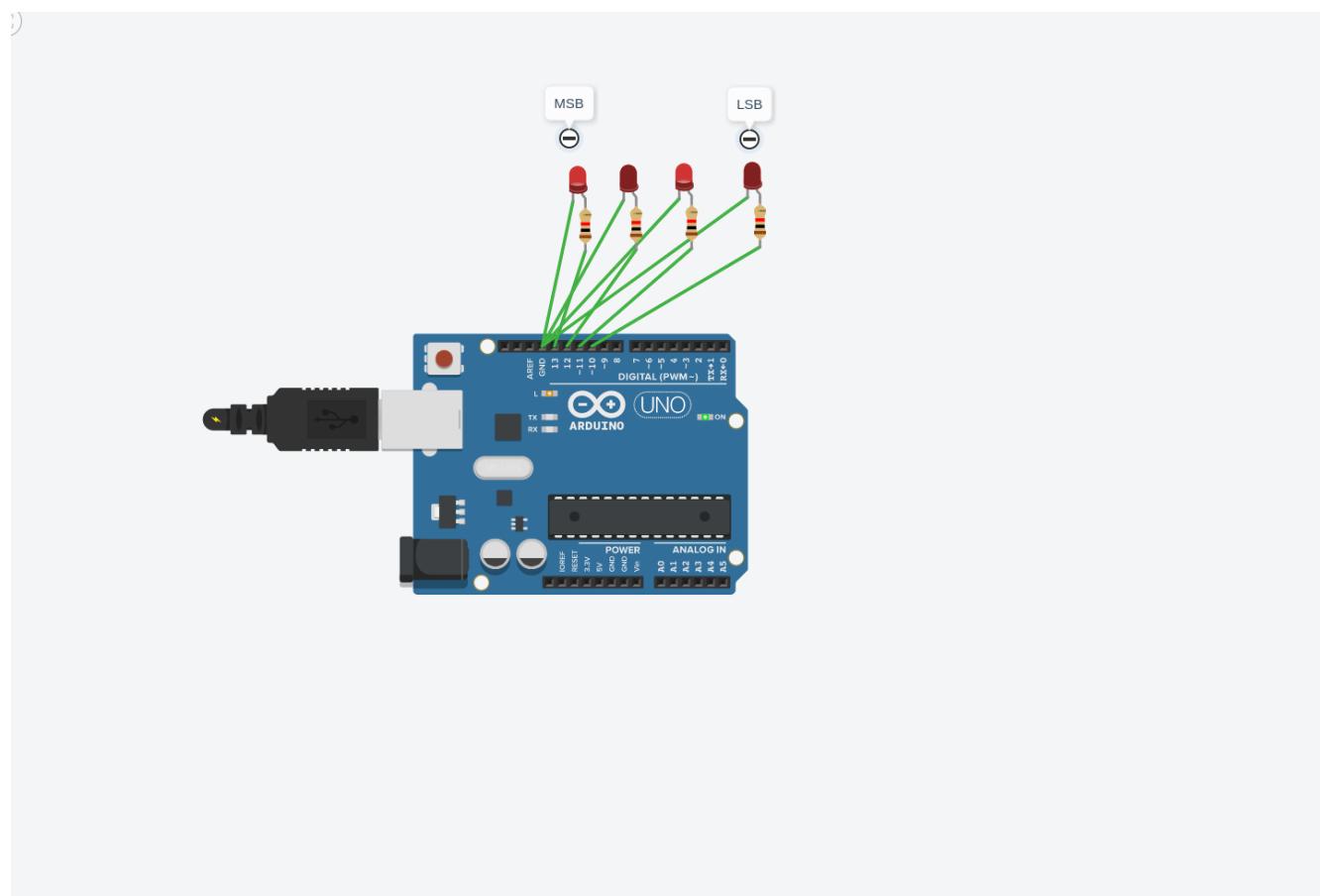
Code

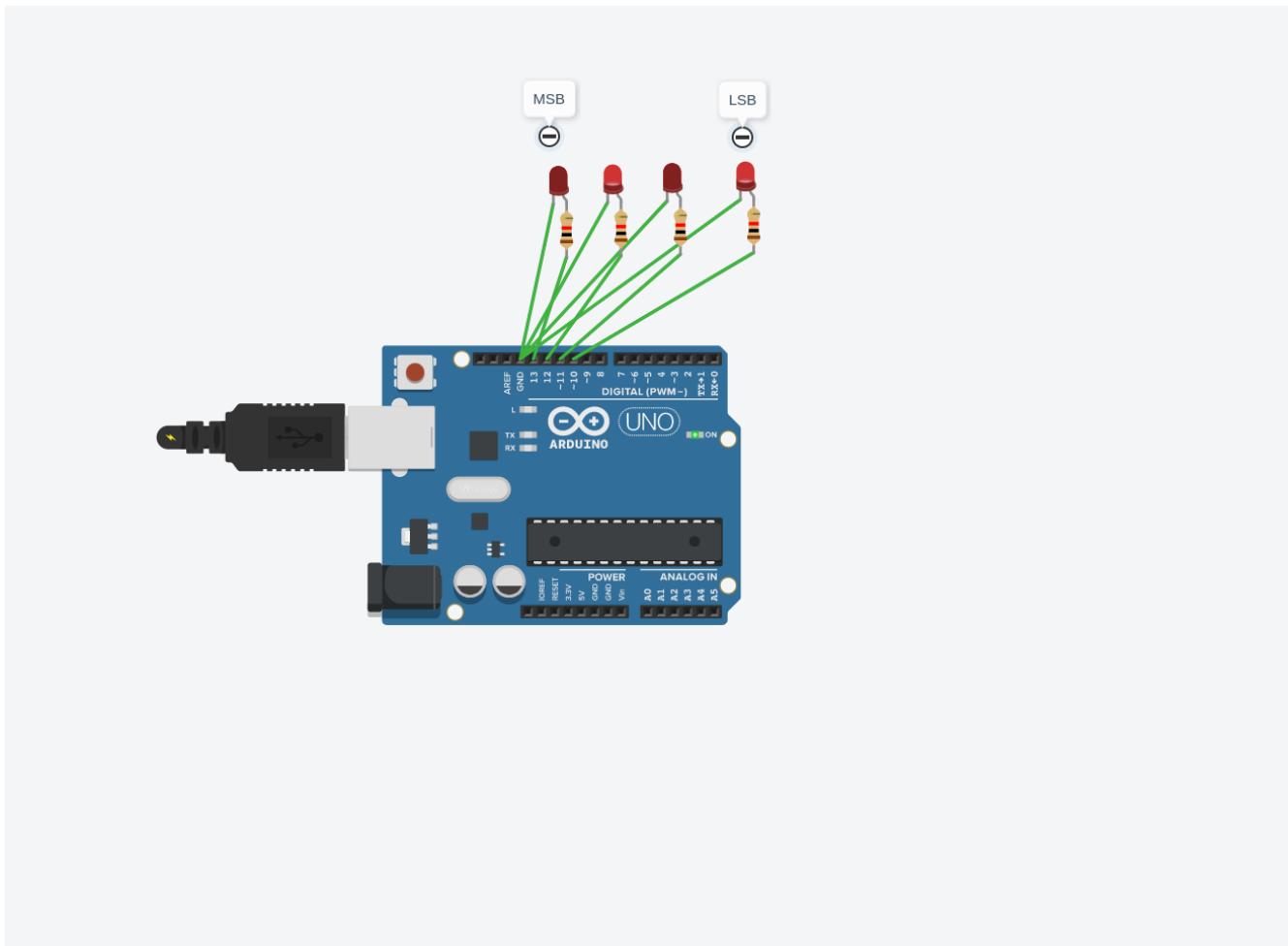
```
// C++ code
//
#define F(x) x? HIGH: LOW
int pins[] = {13,12,11,10};
void setup()
{
    for(int i=0;i<4;i++)
        pinMode(pins[i], OUTPUT);
}
void show(int x){
    int vals[] = {F(x&8) ,F(x&4) , F(x&2) , F(x&1)} ;
    for(int i=0;i<4;i++){
        digitalWrite(pins[i],vals[i]);
    }
}

void loop()
{
    show(0b1010);
    delay(1000);
    show(0b0101);
    delay(1000);

}
```

Output





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

Question

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

Aim

To blink LEDs and count from 1 to 15. If the board is counting odd value, it should wait for 1 sec, else it should wait for 2 sec.

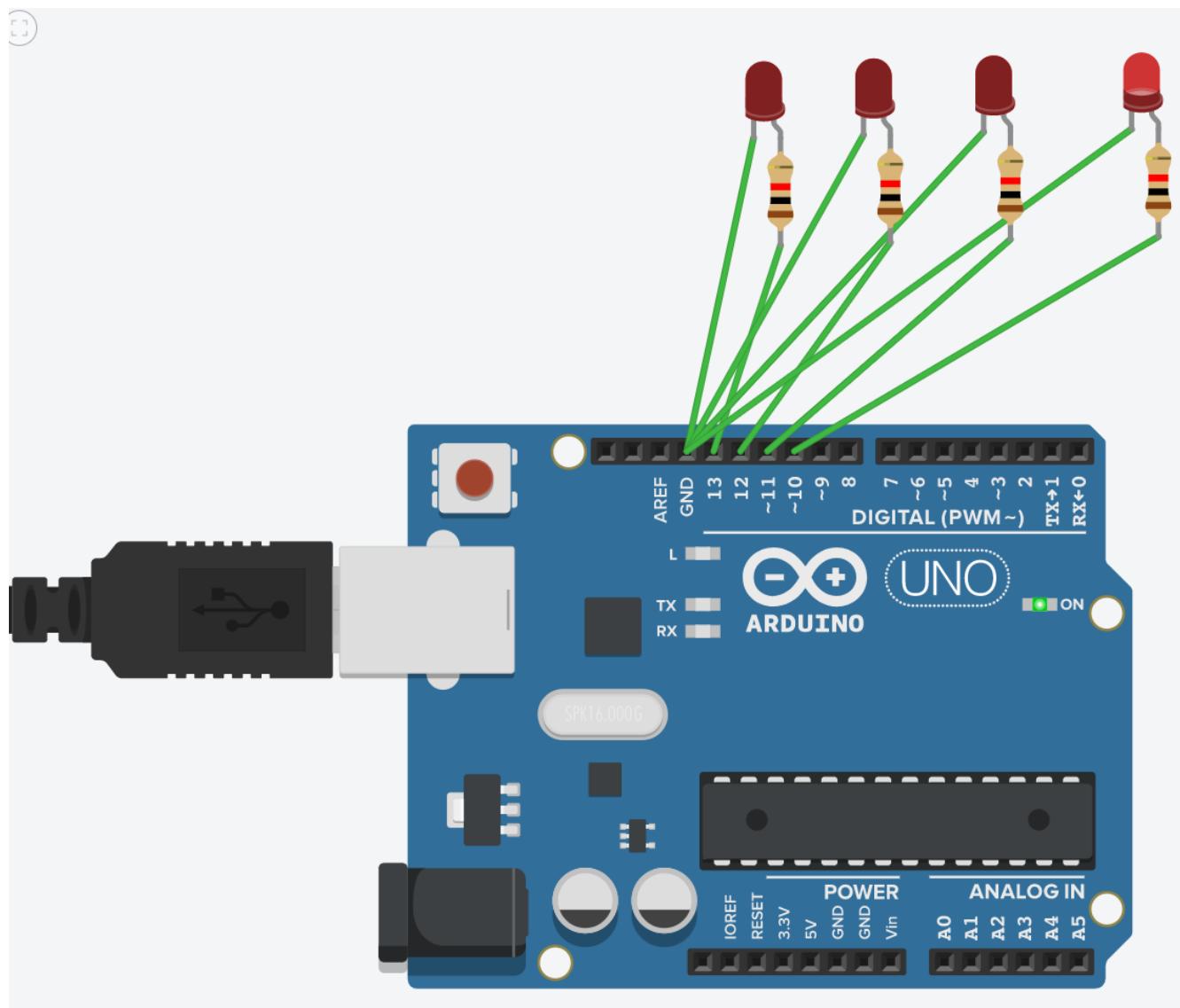
Code

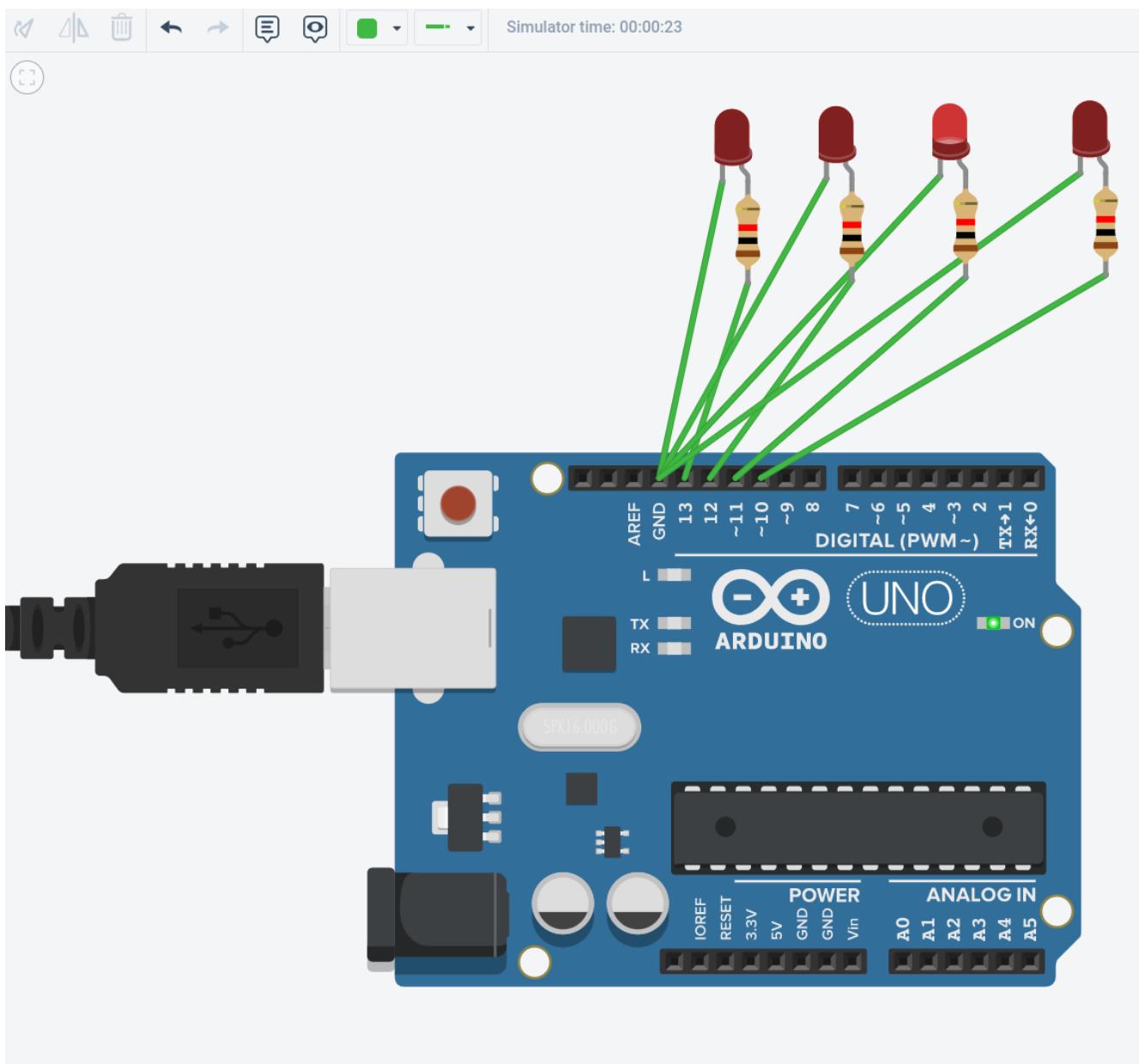
```
// C++ code
//
#define F(x) x? HIGH: LOW
int pins[] = {13,12,11,10};
void setup()
{
    for(int i=0;i<4;i++)
        pinMode(pins[i], OUTPUT);
}
void show(int x){
    int vals[] = {F(x&8),F(x&4), F(x&2), F(x&1)} ;
    for(int i=0;i<4;i++){

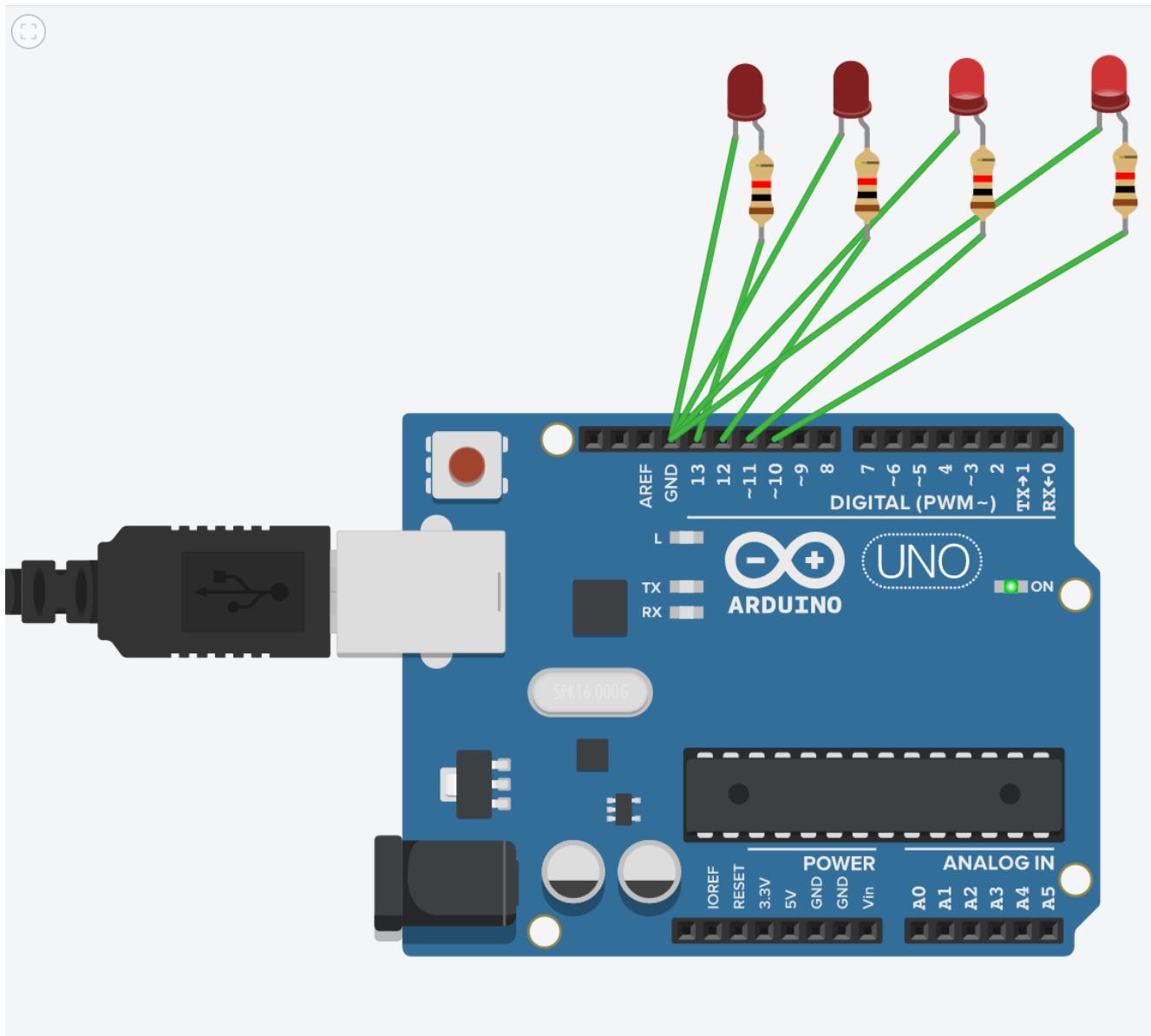
```

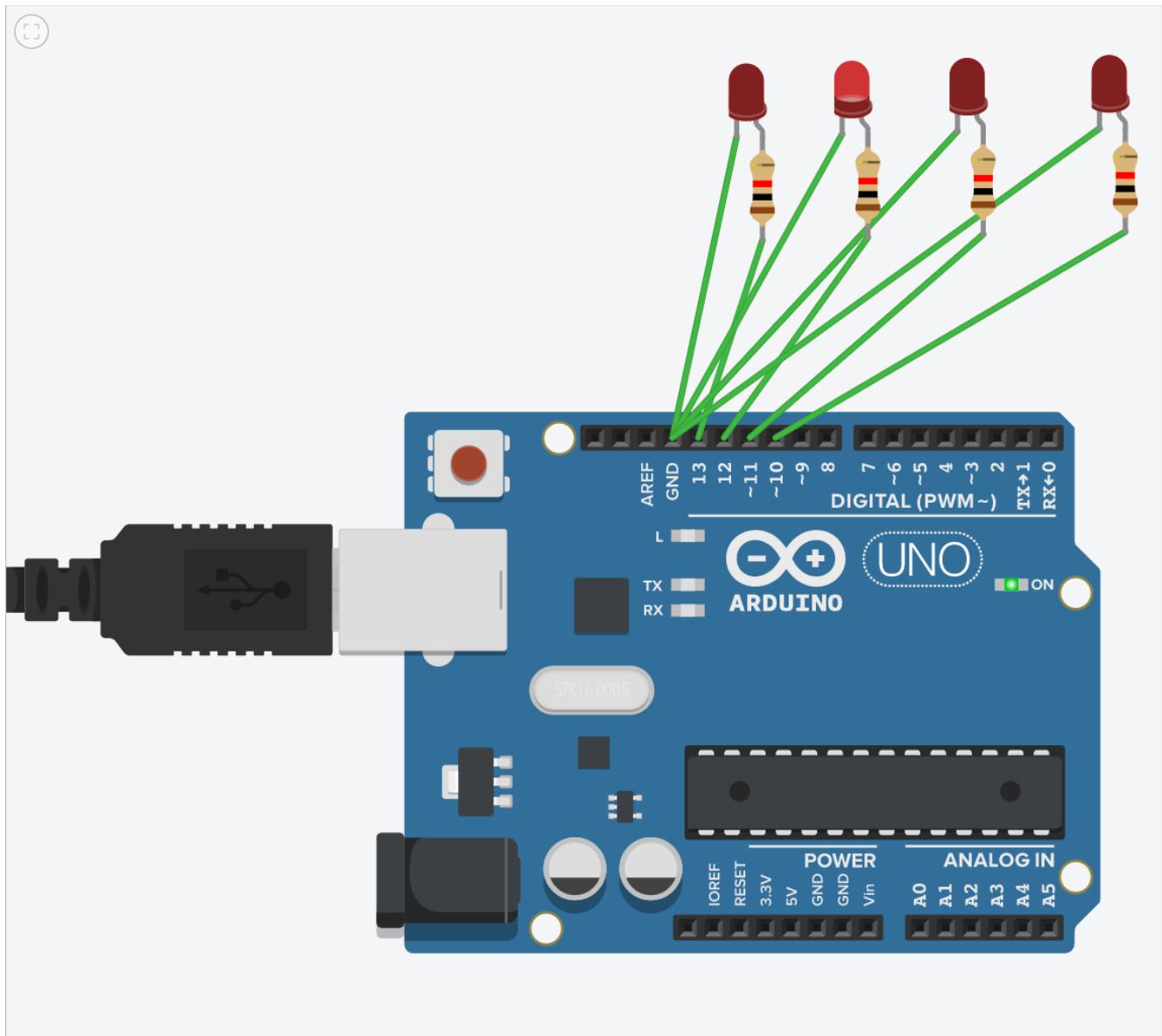
```
    digitalWrite(pins[i],vals[i]);  
}  
}  
  
void loop()  
{  
    for(int i=1;i<16;i++){  
        show(i);  
        if(i%2==1)  
            delay(1000);  
        else  
            delay(2000);  
    }  
}
```

Output









17 Blink LEDs – for all composite number below 15.

Question

Blink LEDs – for all composite number below 15.

Aim

To blink LEDs for all composite numbers below 15.

Code

```
// C++ code
//
#define F(x) x? HIGH: LOW

int pins[] = {13,12,11,10};

void setup()
{
```

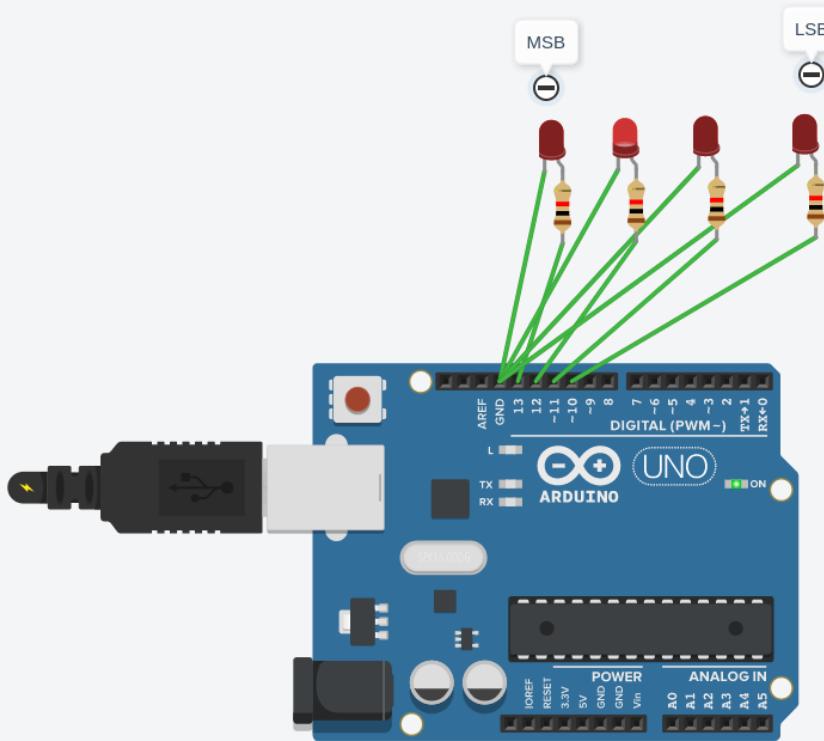
```
for(int i=0;i<4;i++)
    pinMode(pins[i], OUTPUT);
}

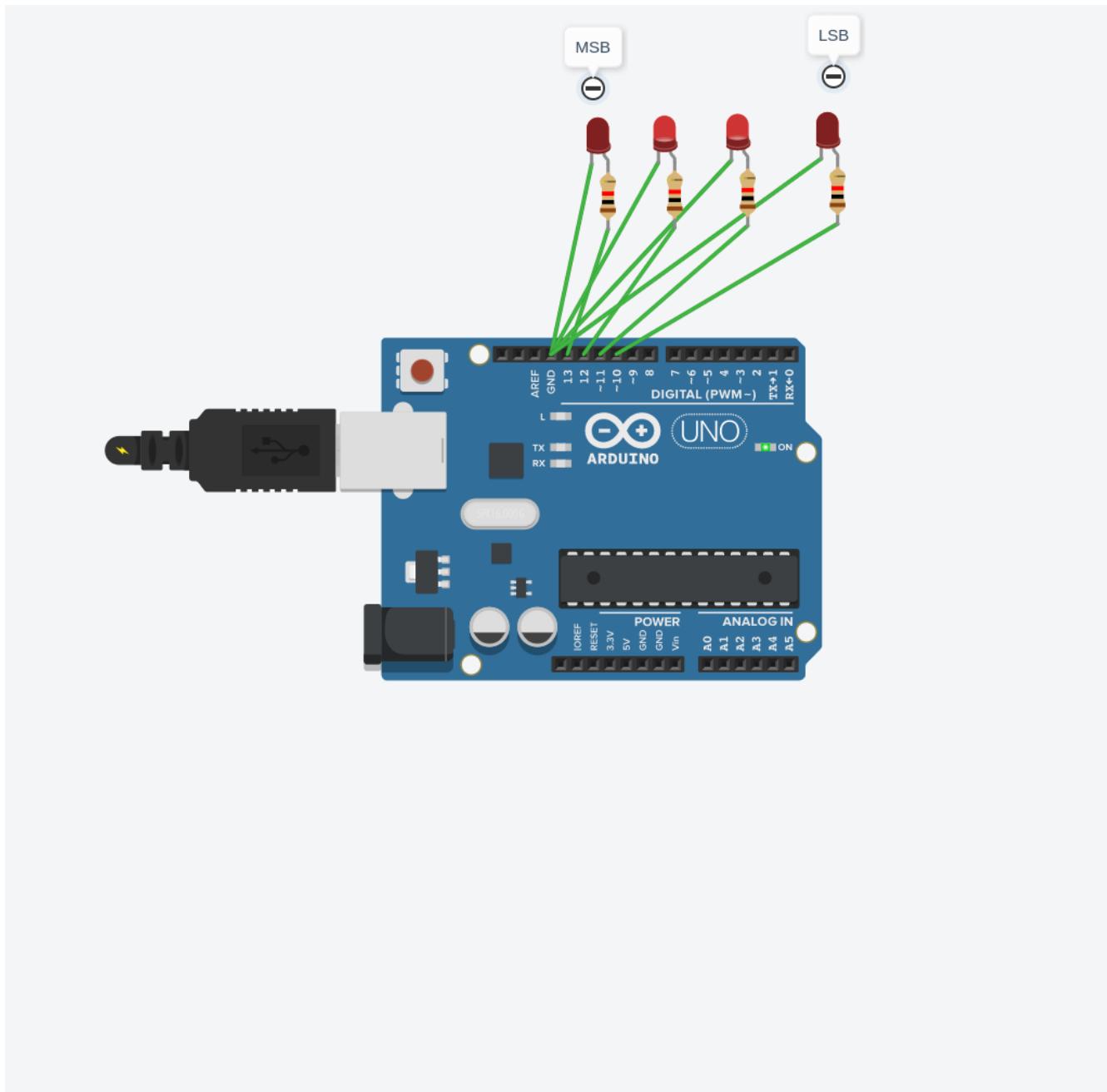
void show(int x){
    int vals[] = {F(x&8) ,F(x&4) , F(x&2) , F(x&1)} ;
    for(int i=0;i<4;i++){
        digitalWrite(pins[i],vals[i]);
    }
}

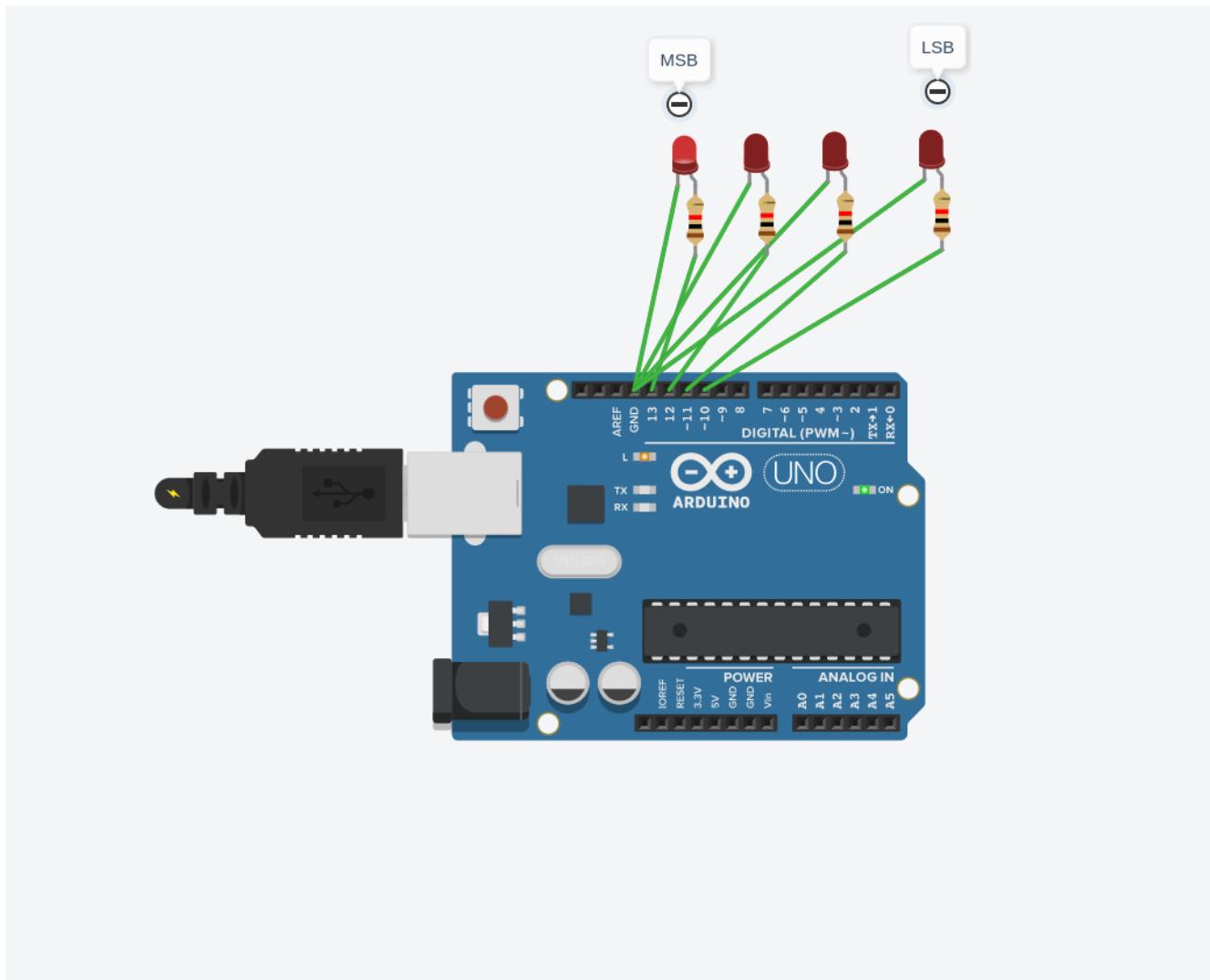
bool is_prime(int x){
    for(int i=2;i<x;i++){
        if(x%i==0)
            return false;
    }
    return true;
}

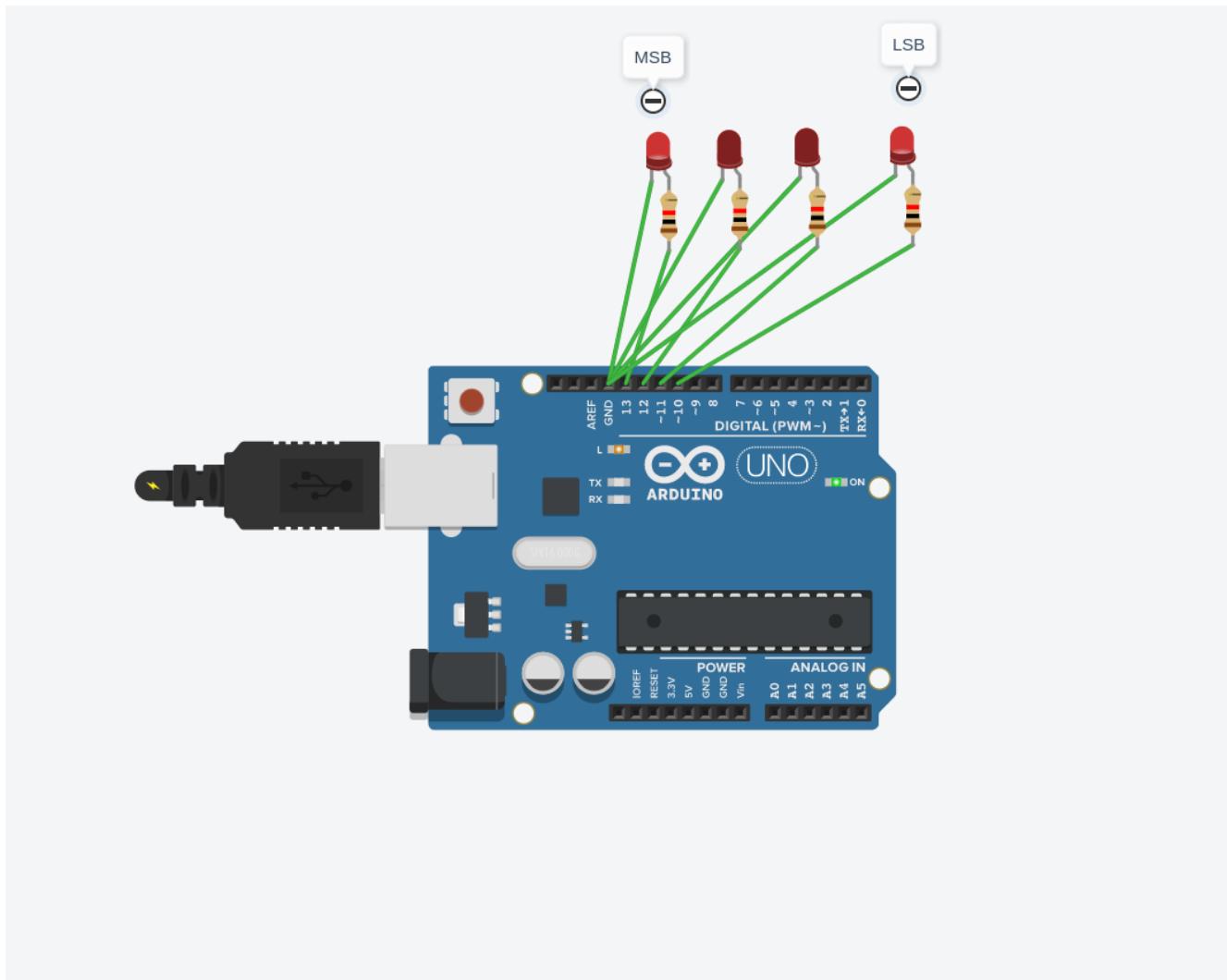
void loop()
{
    for(int i=2;i<16;i+=1){
        if(!is_prime(i)){
            show(i);
            delay(1000);
        }
    }
}
```

Output









18 Blink LEDs – to count even numbers

Question

Blink LEDs – to count even numbers

Aim

To blink LEDs to count even numbers

Code

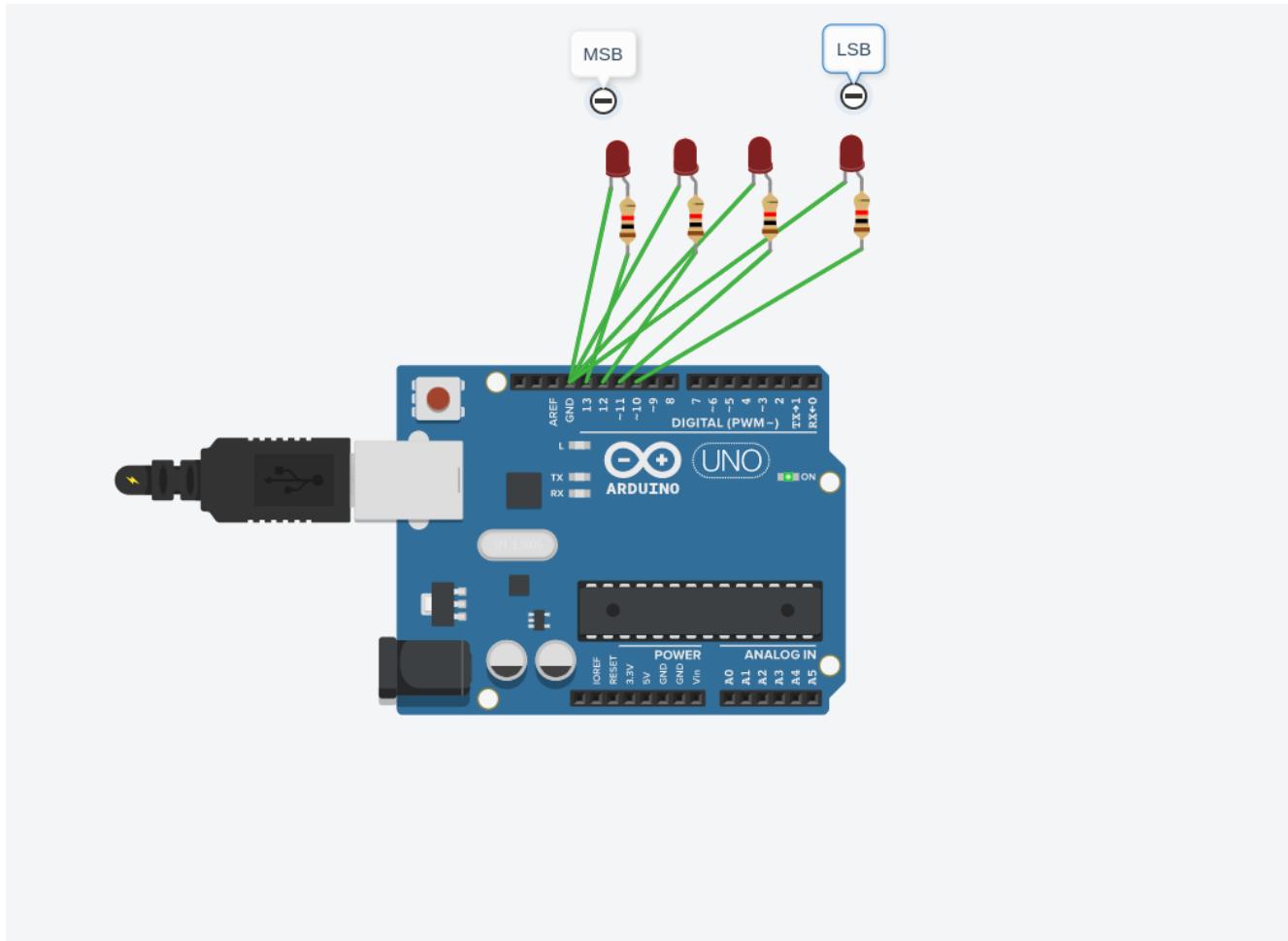
```
// C++ code
//
#define F(x) x? HIGH: LOW

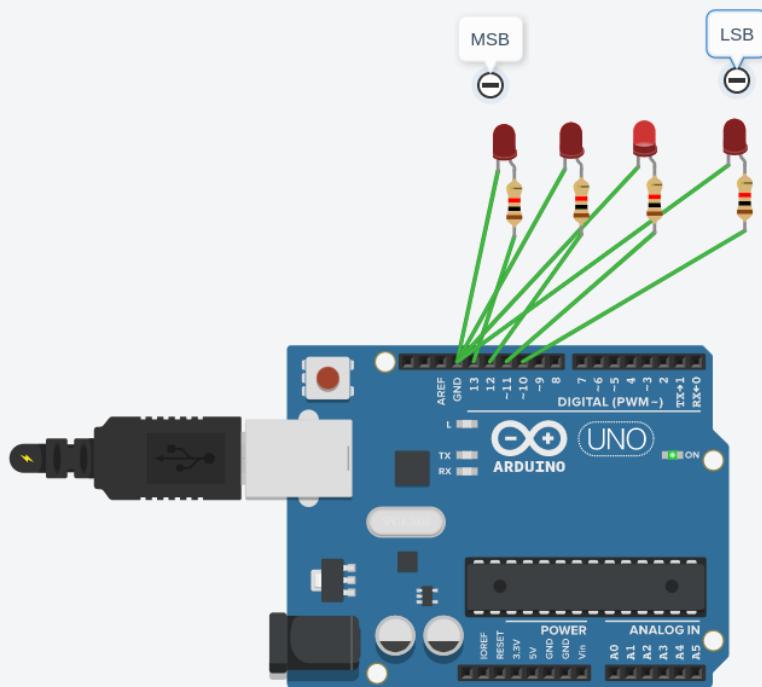
int pins[] = {13,12,11,10};

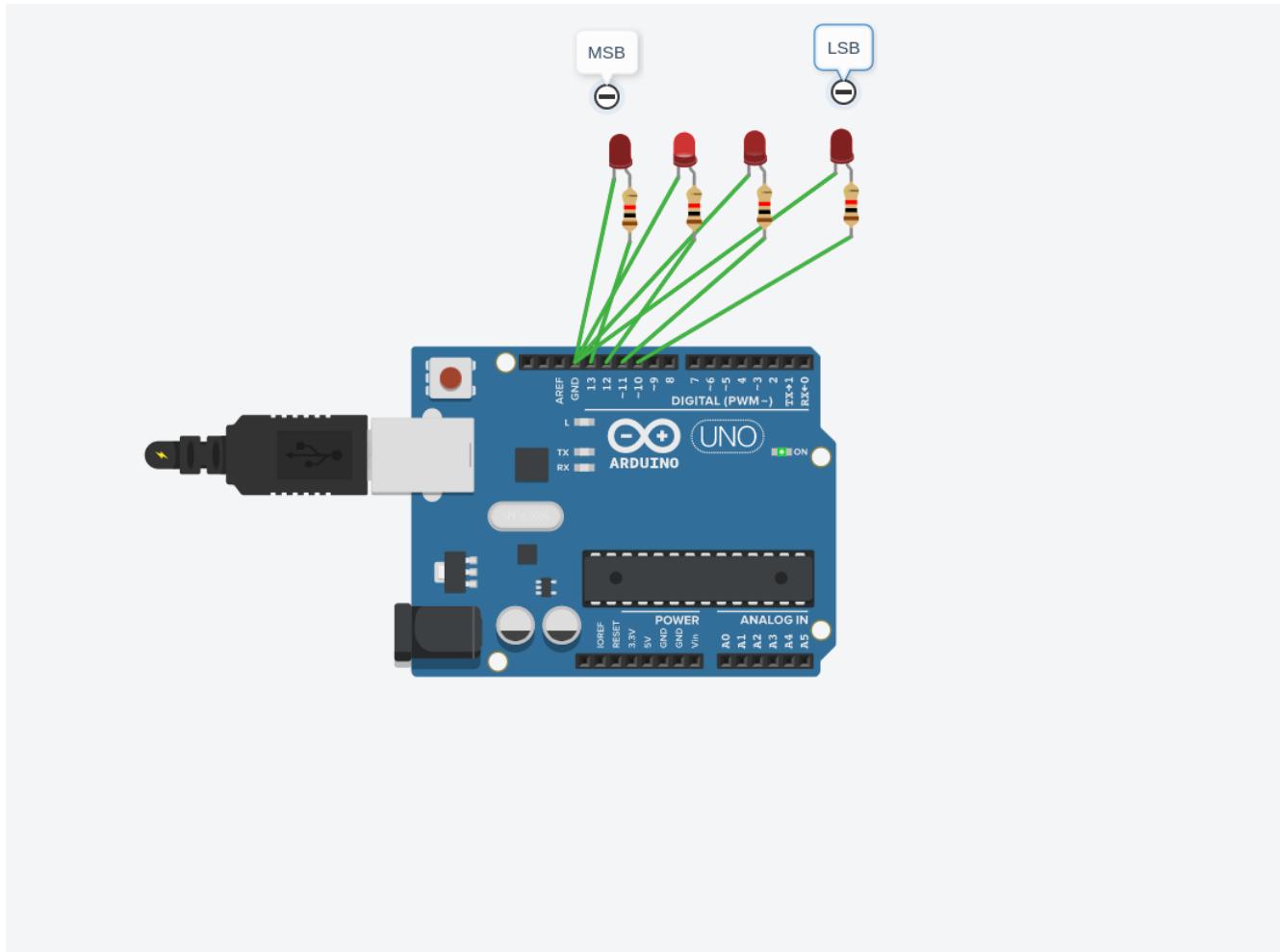
void setup()
{
    for(int i=0;i<4;i++)
        pinMode(pins[i], OUTPUT);
}
```

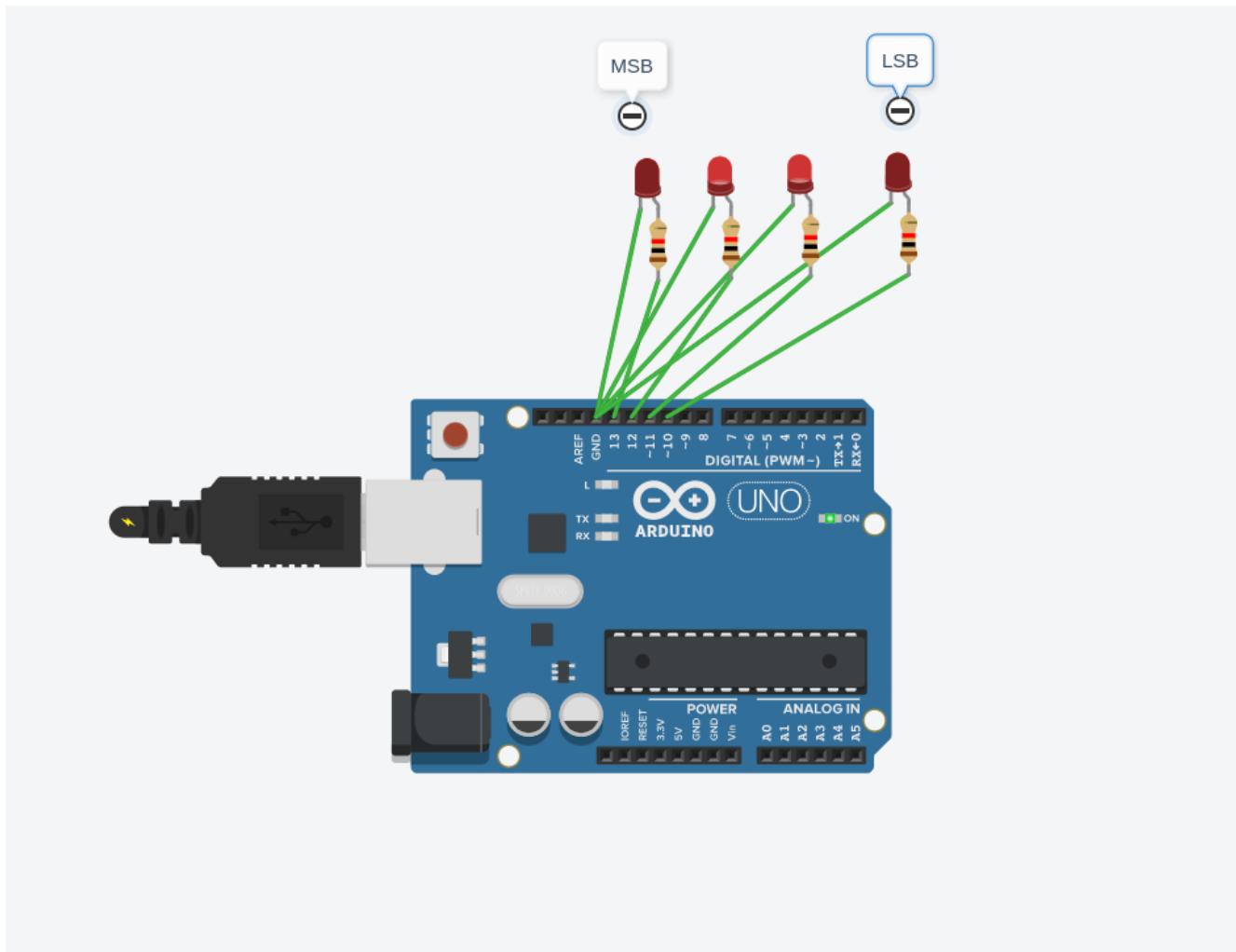
```
void show(int x){  
    int vals[] = {F(x&8) ,F(x&4) , F(x&2) , F(x&1)} ;  
    for(int i=0;i<4;i++){  
        digitalWrite(pins[i],vals[i]);  
    }  
}  
  
void loop()  
{  
    for(int i=0;i<16;i+=2){  
        show(i);  
        delay(1000);  
    }  
}
```

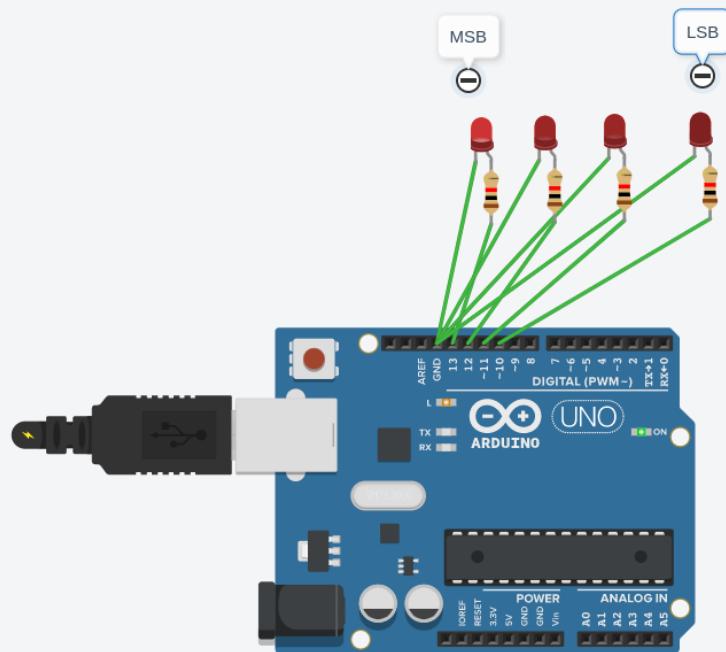
Output











Lab 6

Date: 28-03-2022

Questions

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

19 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.

Question

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.

Aim

To 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. We must play different tones for different cases.

Code

```
// C++ code
//
#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();

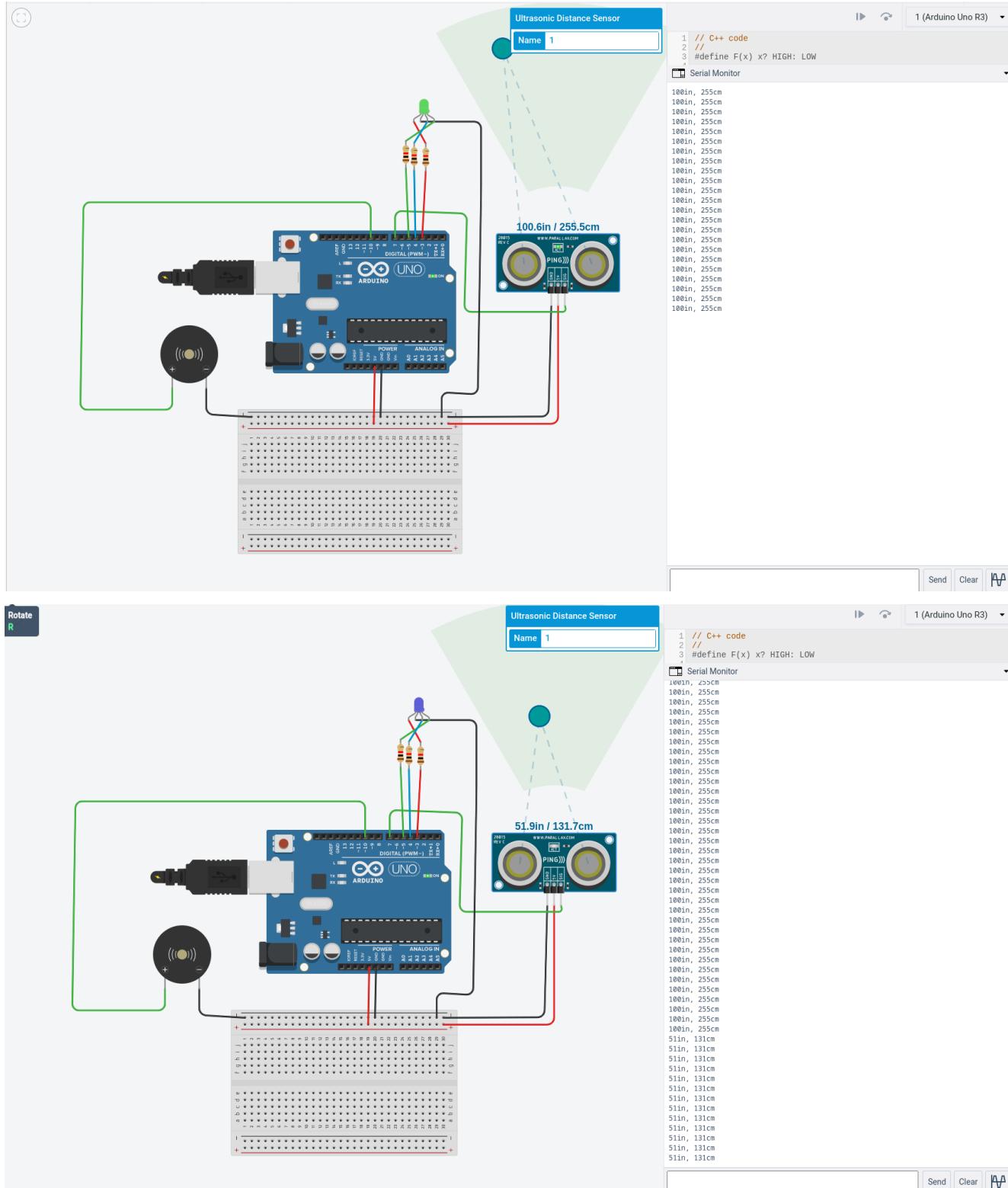
if (cm < 50) {
  displayLight(0);
  playTone(300, 100);
} else if (cm < 150) {
  displayLight(1);
  playTone(300, 200);
} else {
  displayLight(2);
  playTone(300, 300);
}

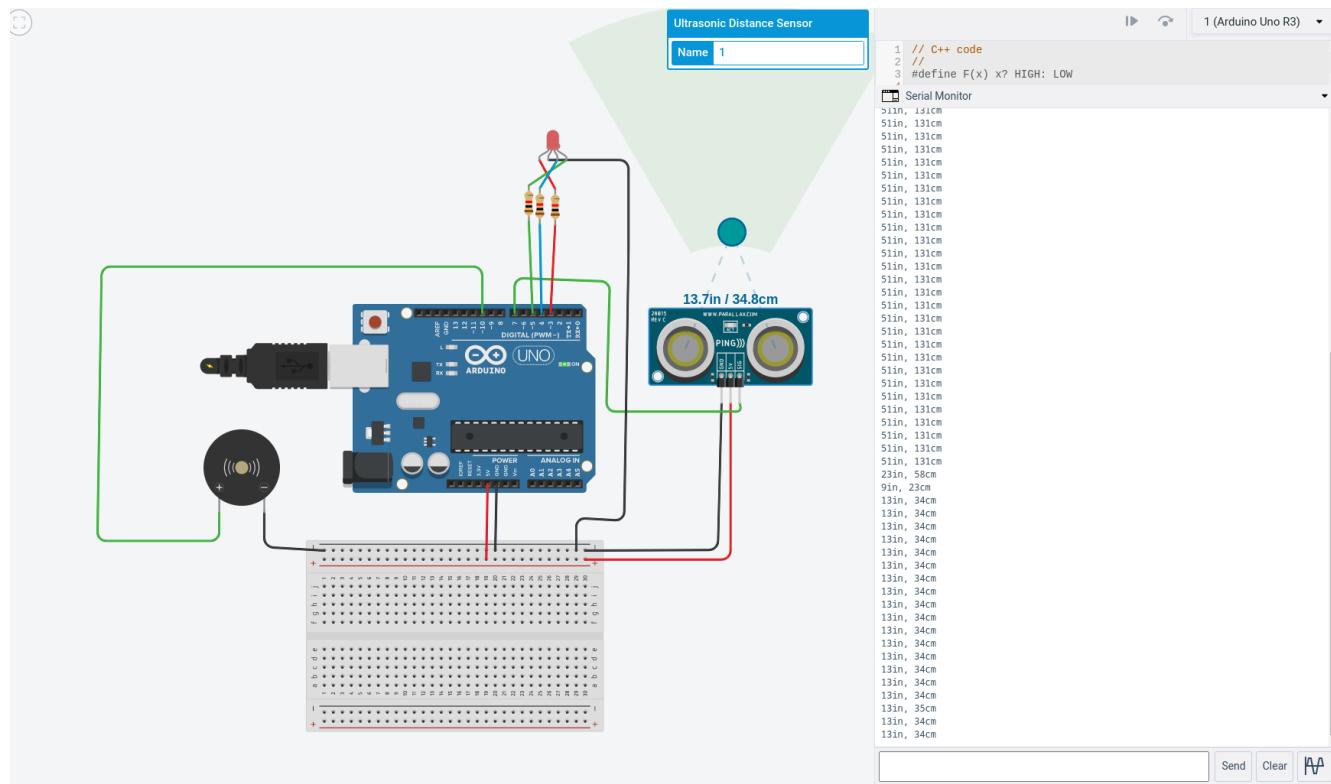
delay(100);
}

long microsecondsToInches(long microseconds) {
  return microseconds / 74 / 2;
}

long microsecondsToCentimeters(long microseconds) {
  return microseconds / 29 / 2;
}
```

Output





20 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.

Question

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

Aim

To interface a PIR sensor with Arduino and blink a led if any movement is captured by the sensor and count the number of movements. We must make a buzzer when movement detected.

Code

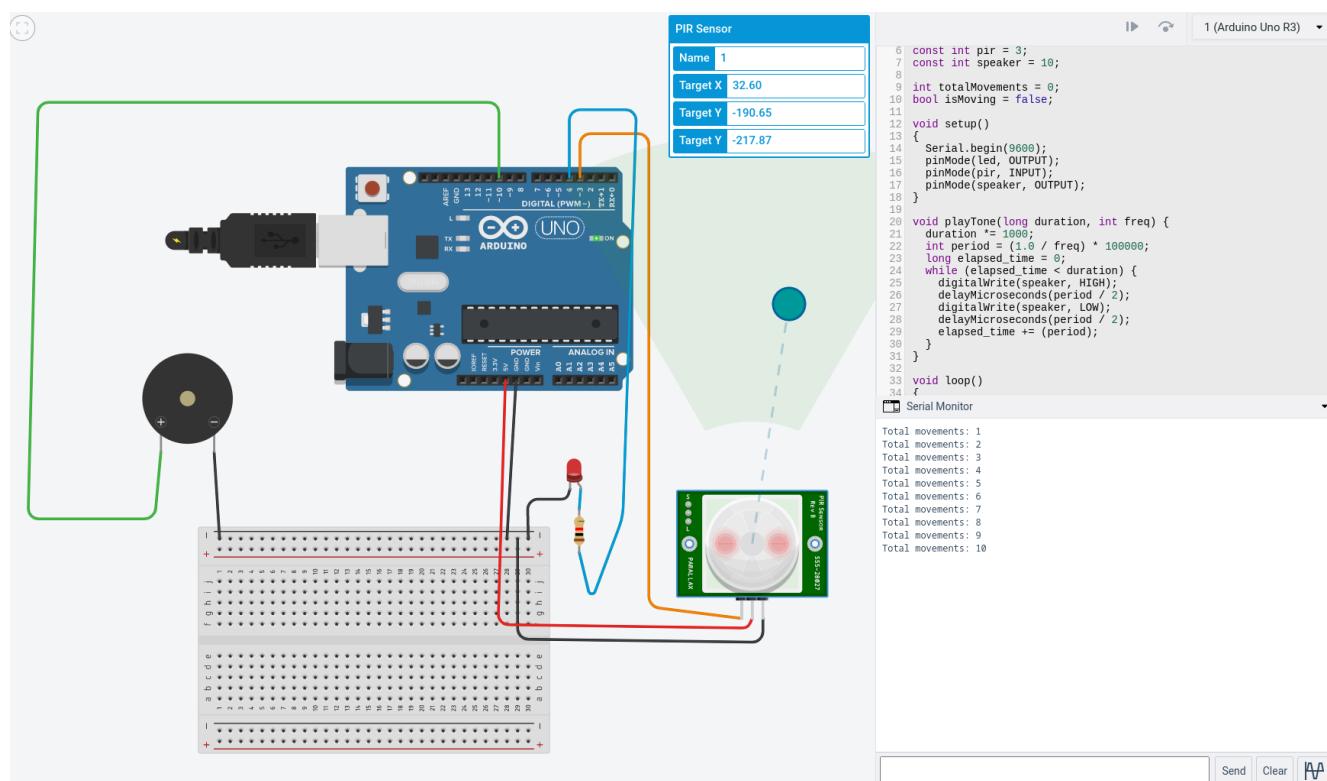
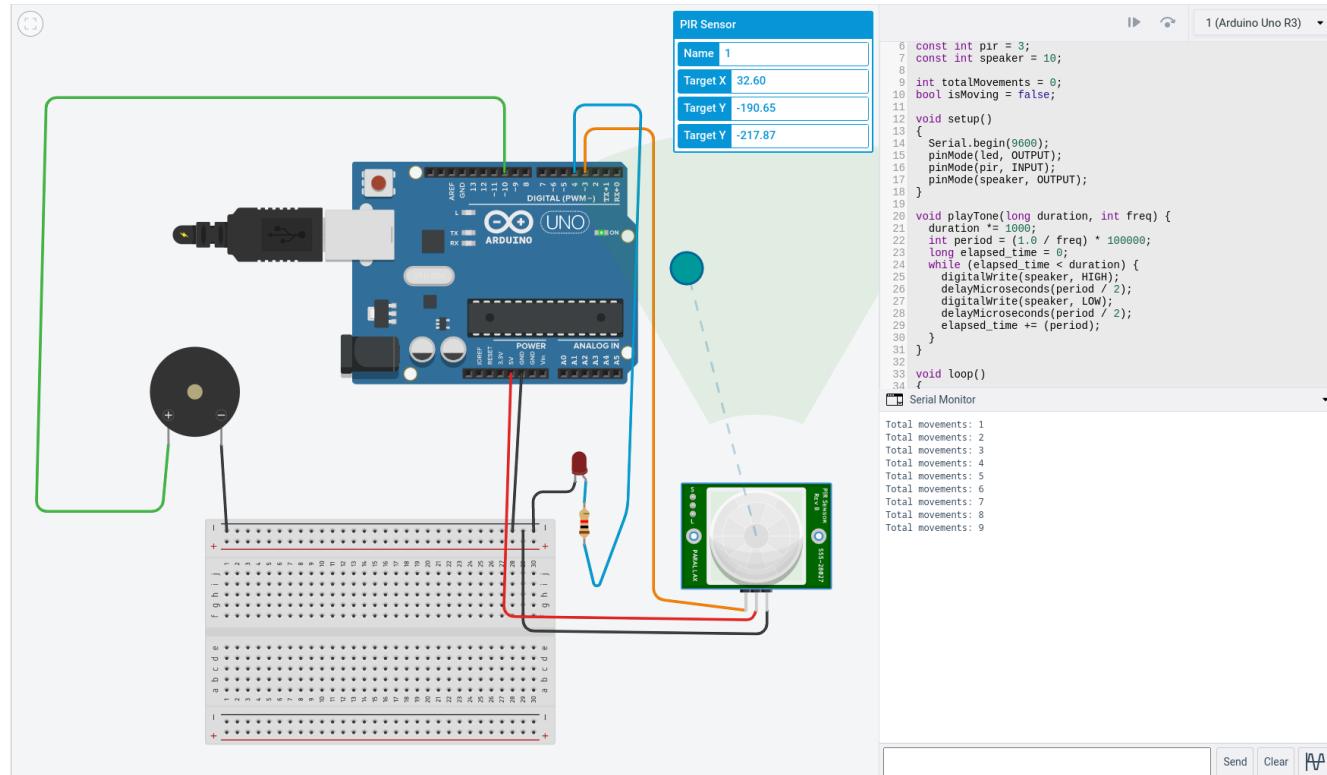
```
// C++ code
//  
#define F(x) x? HIGH: LOW  
  
const int led = 4;  
const int pir = 3;  
const int speaker = 10;  
  
int totalMovements = 0;  
bool isMoving = false;  
  
void setup()  
{  
    Serial.begin(9600);  
    pinMode(led, OUTPUT);  
    pinMode(pir, INPUT);
```

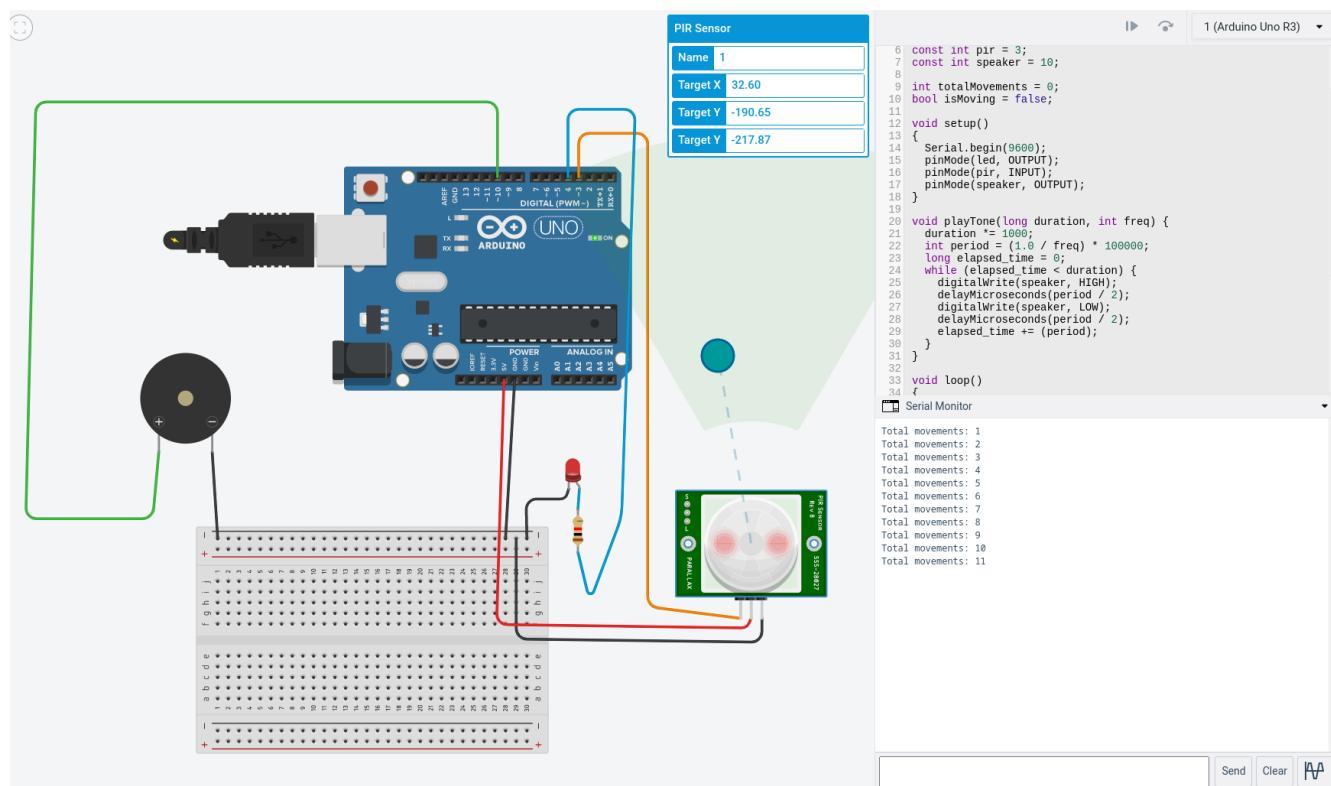
```
pinMode(speaker, OUTPUT);
}

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);
    }
}
```

Output





Lab 7**Date: 11-04-2022****Questions**

1. Interface two 7-segment display with Arduino Uno board. Implement the a)odd counter and b)even counter and display the values in the seven segment display.
2. 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).
3. 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.

21 Interface two 7-segment display with Arduino Uno board. Implement the a)odd counter and b)even counter and display the values in the seven segment display.**Question**

Interface two 7-segment display with Arduino Uno board. Implement the a)odd counter and b)even counter and display the values in the seven segment display.

Aim

To interface two 7-segment display with Arduino Uno board. We must implement a)odd counter b)even counter and display the values in the seven segment display.

Code

```
unsigned const int A = 13;
unsigned const int B = 12;
unsigned const int C = 11;
unsigned const int D = 10;
unsigned const int E = 9;
unsigned const int F = 8;
unsigned const int G = 7;
unsigned const int H = 6;
unsigned const int btn = 4;

void setup(void)
{
    pinMode(A, OUTPUT);
    pinMode(B, OUTPUT);
    pinMode(C, OUTPUT);
    pinMode(D, OUTPUT);
    pinMode(E, OUTPUT);
    pinMode(F, OUTPUT);
    pinMode(G, OUTPUT);
    pinMode(H, OUTPUT);
    pinMode(btn, INPUT);
}
int pins[] = {A,B,C,D,E,F,G,H};

int zero = 0b01111110;
int one = 0b00010010;
int two = 0b10111100;
int three= 0b10110110;
int four = 0b11010010;
int five = 0b11100110;
int six = 0b11101110;
int seven= 0b00110010;
```

```
int eight= 0b1111110;
int nine = 0b11110110;

//void (*nums[])() ={zero,one,two,three, four, five,six,seven,eight,nine};
int numbers[] = {zero,one,two,three, four, five,six,seven,eight,nine};

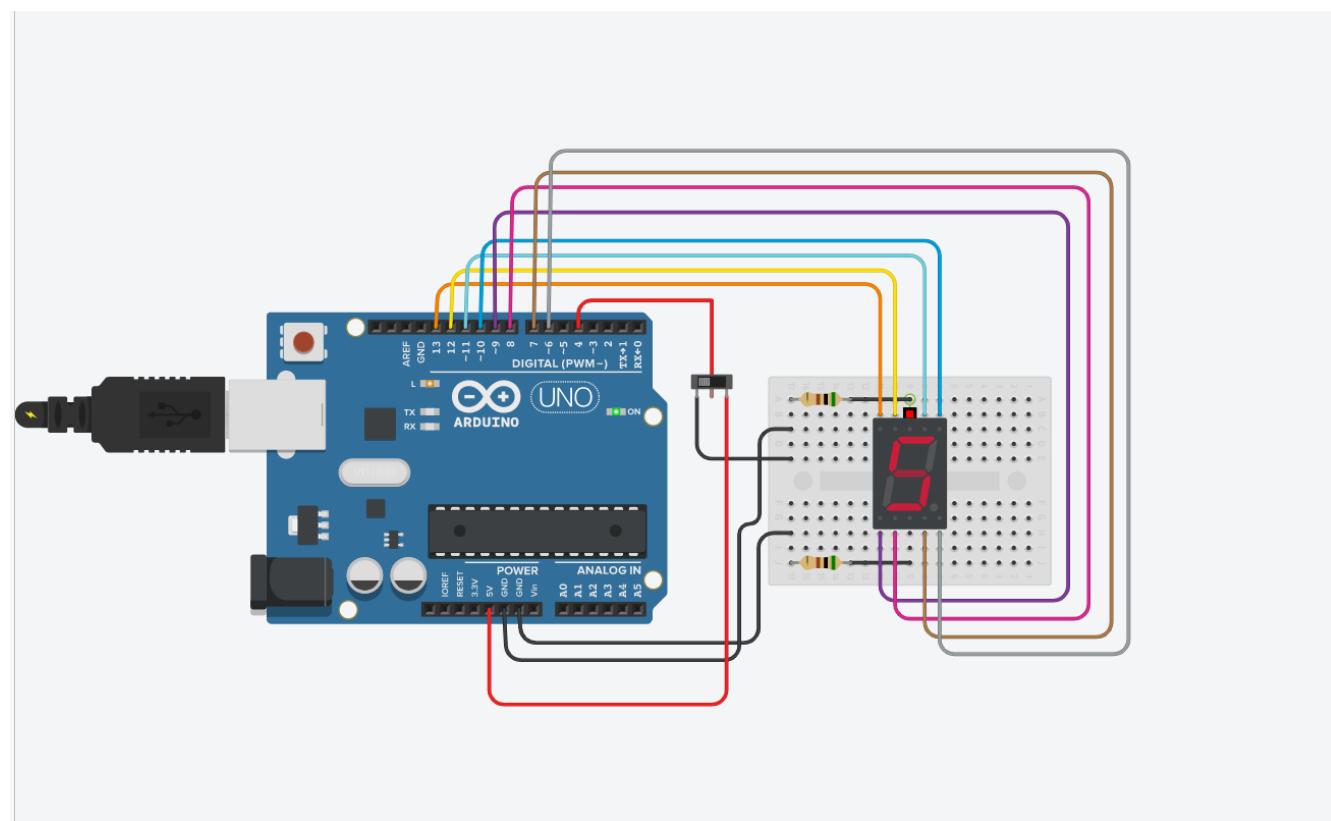
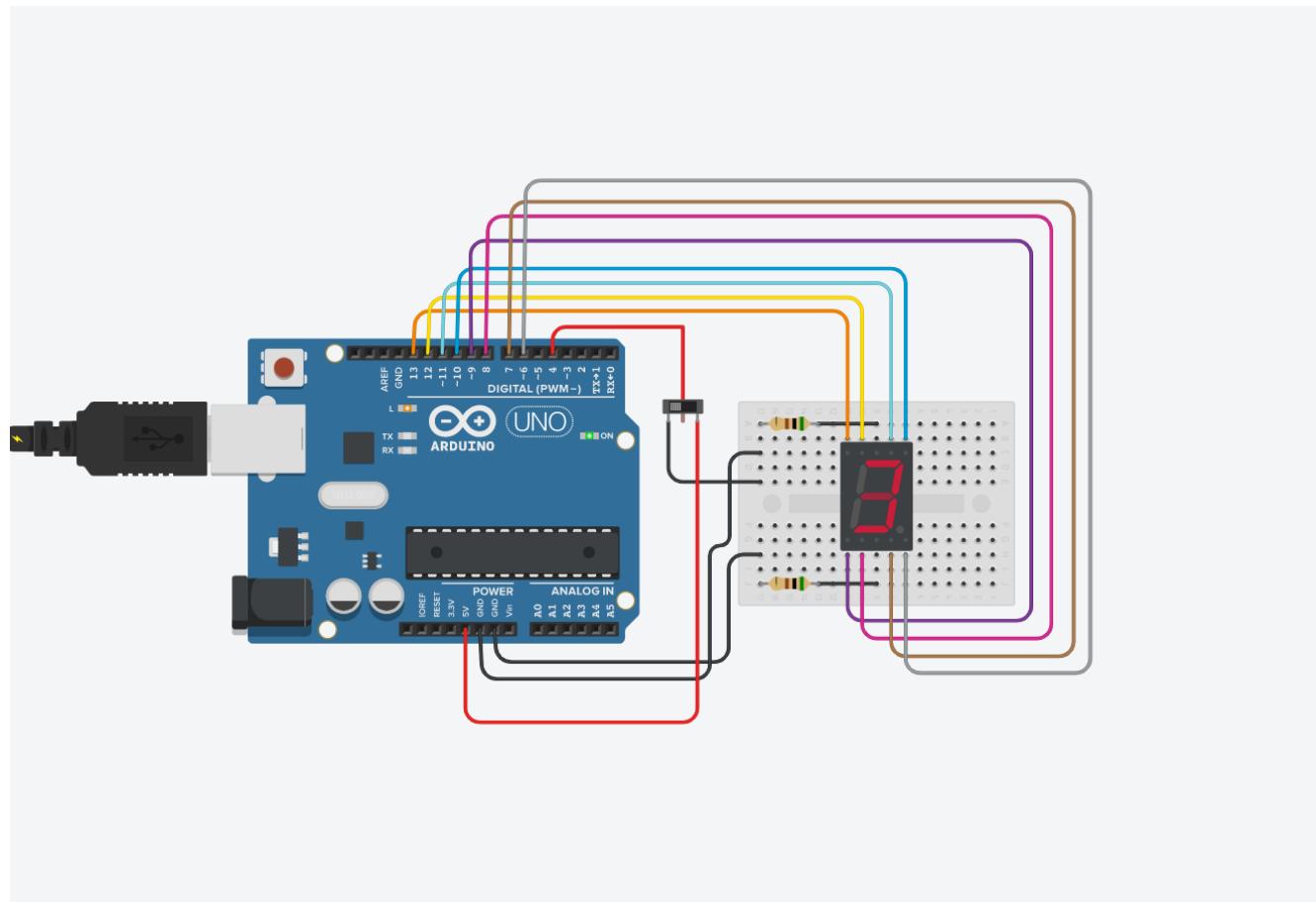
void show(int num){
    digitalWrite(H, num&1);
    digitalWrite(G, num&2);
    digitalWrite(F, num&4);
    digitalWrite(E, num&8);
    digitalWrite(D, num&16);
    digitalWrite(C, num&32);
    digitalWrite(B, num&64);
    digitalWrite(A, num&128);
}

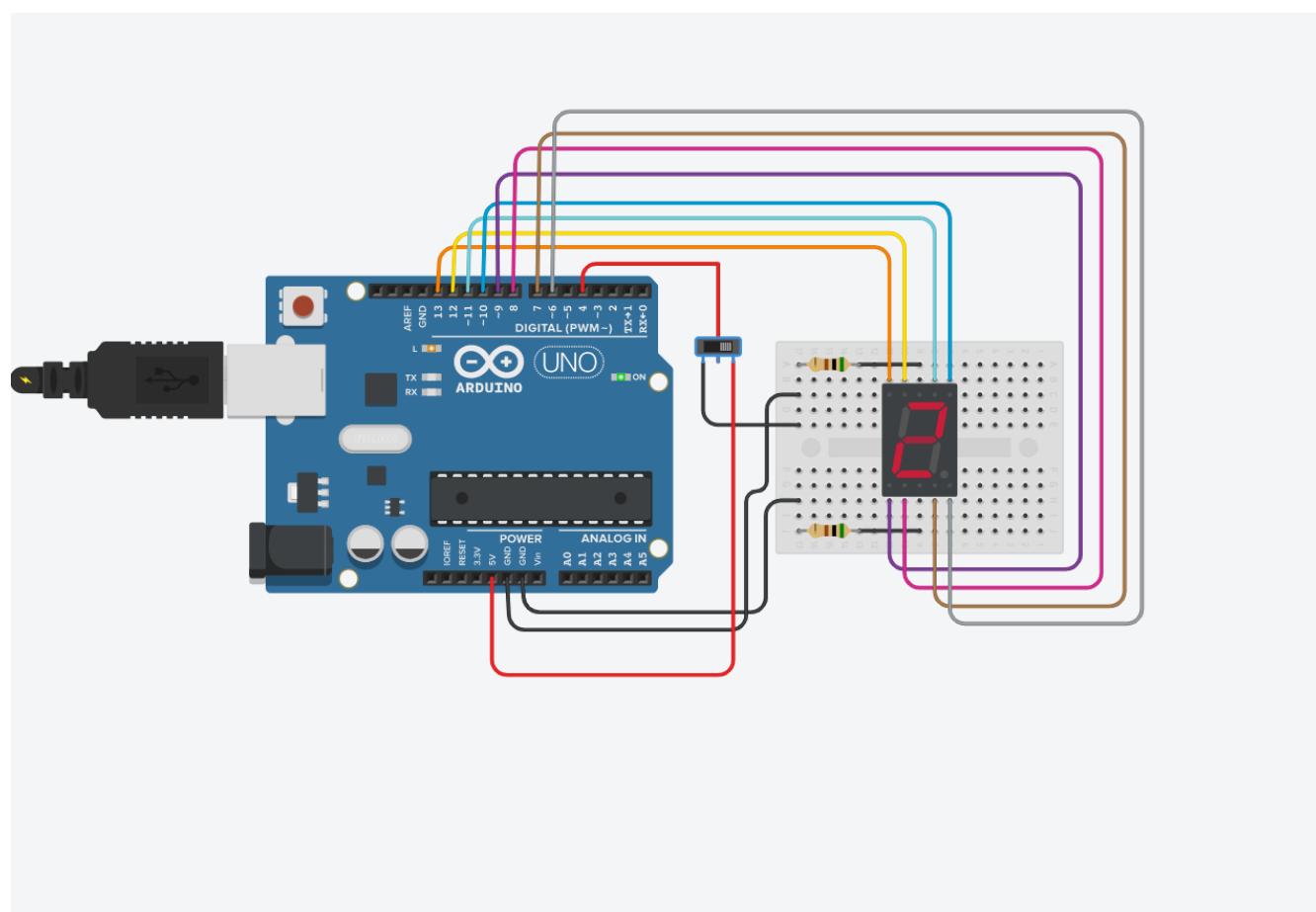
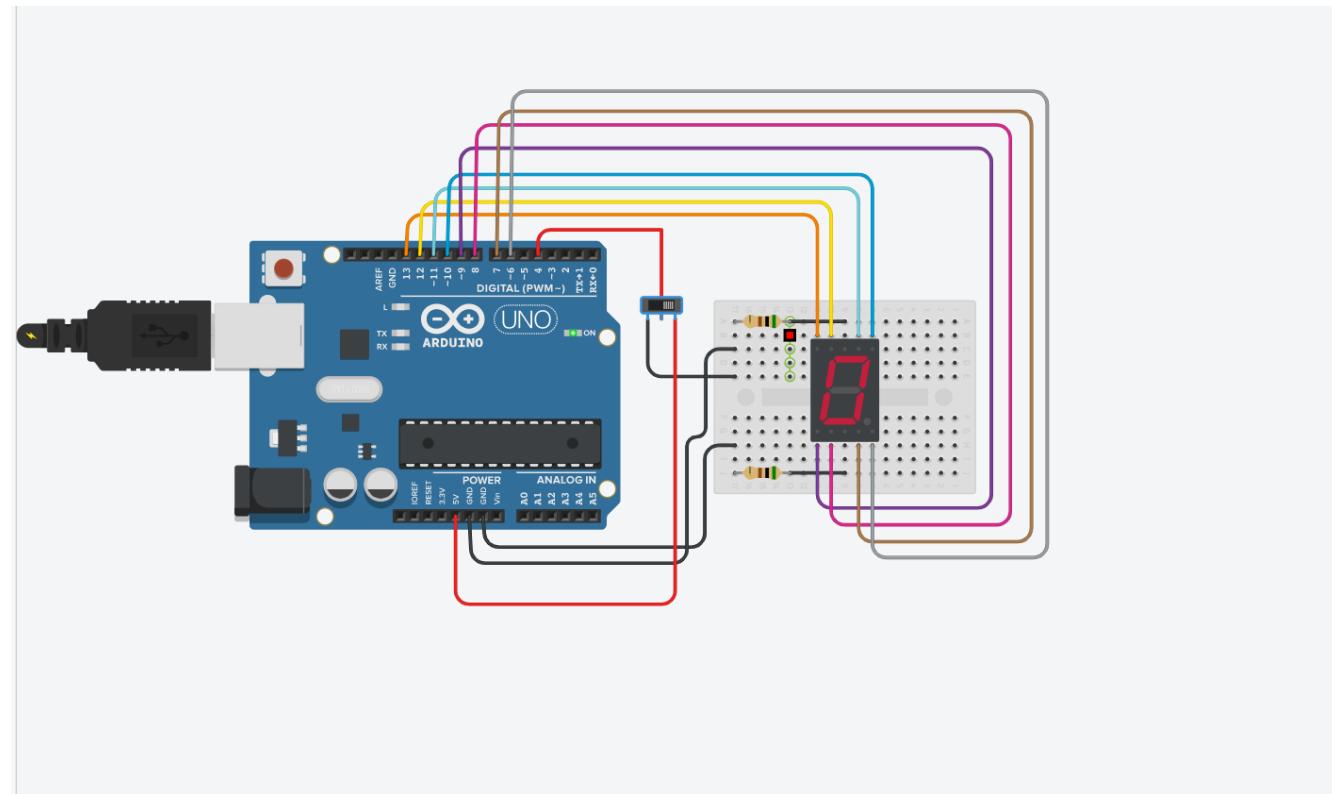
// Start
void loop(void)
{
    if(digitalRead(btn)==HIGH){
        for(int i=0;i<10;i+=2){
            show(numbers[i]);
            delay(1000);
        }
    }else{
        for(int i=1;i<10;i+=2){
            show(numbers[i]);
            delay(1000);
        }
    }
}
```

Output

- Odd and Even decided based on switch

Odd Output



Even Output

22 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).

Question

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).

Aim

To interface the ambient light sensor with Arduino Uno board. We must check the light value from the sensor, and switch on/off the bulb (based on some threshold value).

Code

```
int LED = 12;
int MQ2pin = A0;

void setup() {
    Serial.begin(9600);
}

void loop() {
    float sensorValue;
    sensorValue = analogRead(MQ2pin); // read analog input pin 0

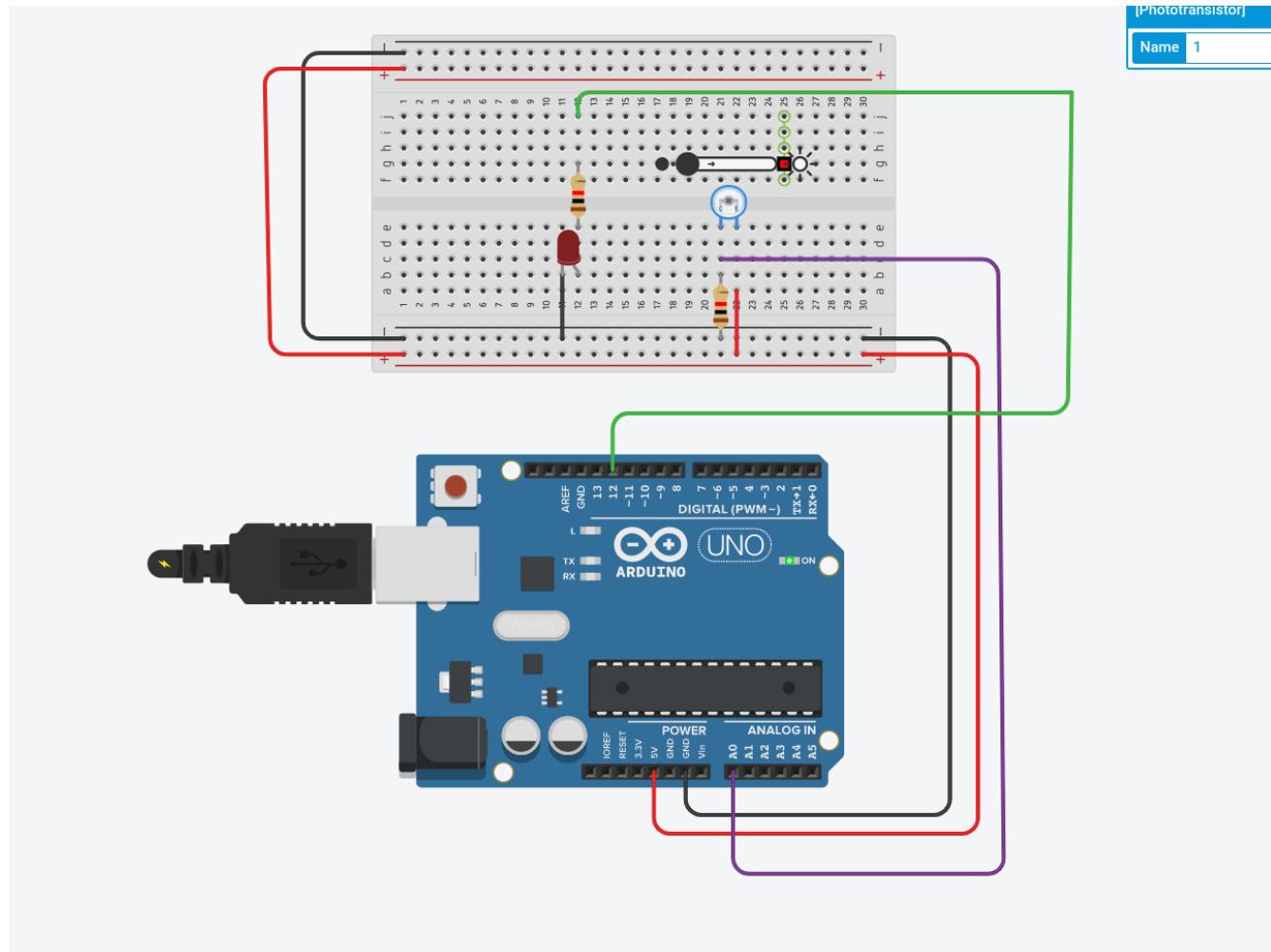
    if(sensorValue >= 10){
        digitalWrite(LED,HIGH);

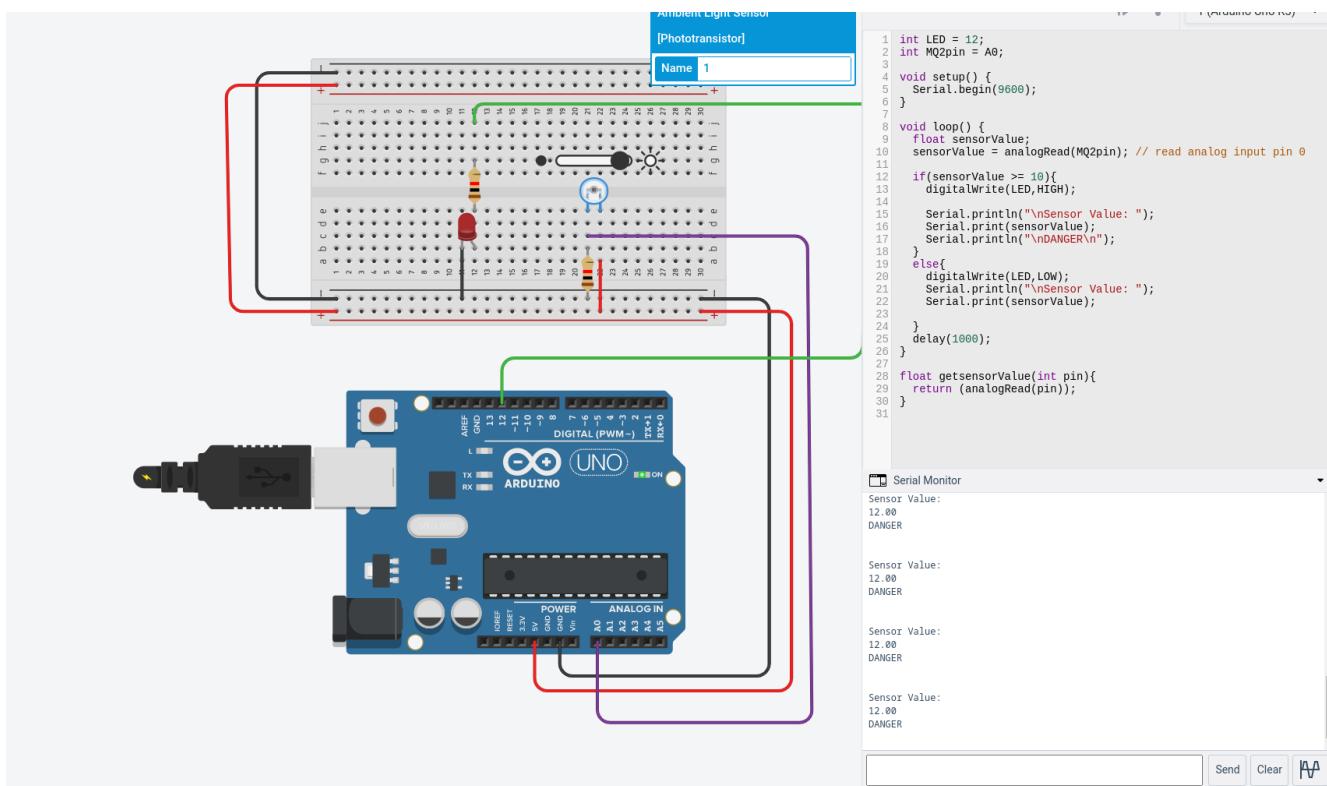
        Serial.println("\nSensor Value: ");
        Serial.print(sensorValue);
        Serial.println("\nDANGER\n");
    }
    else{
        digitalWrite(LED,LOW);
        Serial.println("\nSensor Value: ");
        Serial.print(sensorValue);

    }
    delay(1000);
}

float getsensorValue(int pin){
    return (analogRead(pin));
}
```

Output





23 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.

Question

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.

Aim

To interface the temperature and gas sensor with Arduino Uno board. We must check the temperature and the gas value, if the limit is beyond the threshold, switch on the bulb and make alarm using buzzer.

Code

```

int LED = 12;
int gas = A0;
int tmppin= A2;
int piezo = 13;

void setup() {
  Serial.begin(9600);
  pinMode(LED, OUTPUT);
  pinMode(piezo, OUTPUT);
  pinMode(gas, INPUT);
  pinMode(tmppin, INPUT);
}

void loop() {
  float sensorValue, tempSensor;

```

```
sensorValue = analogRead(gas); // read analog input pin 0

tempSensor= analogRead(tmpPin);

bool smoke= sensorValue >= 300;
bool temp= tempSensor >100;

if(smoke&&temp){
    digitalWrite(LED,HIGH);
    digitalWrite(piezo, HIGH);

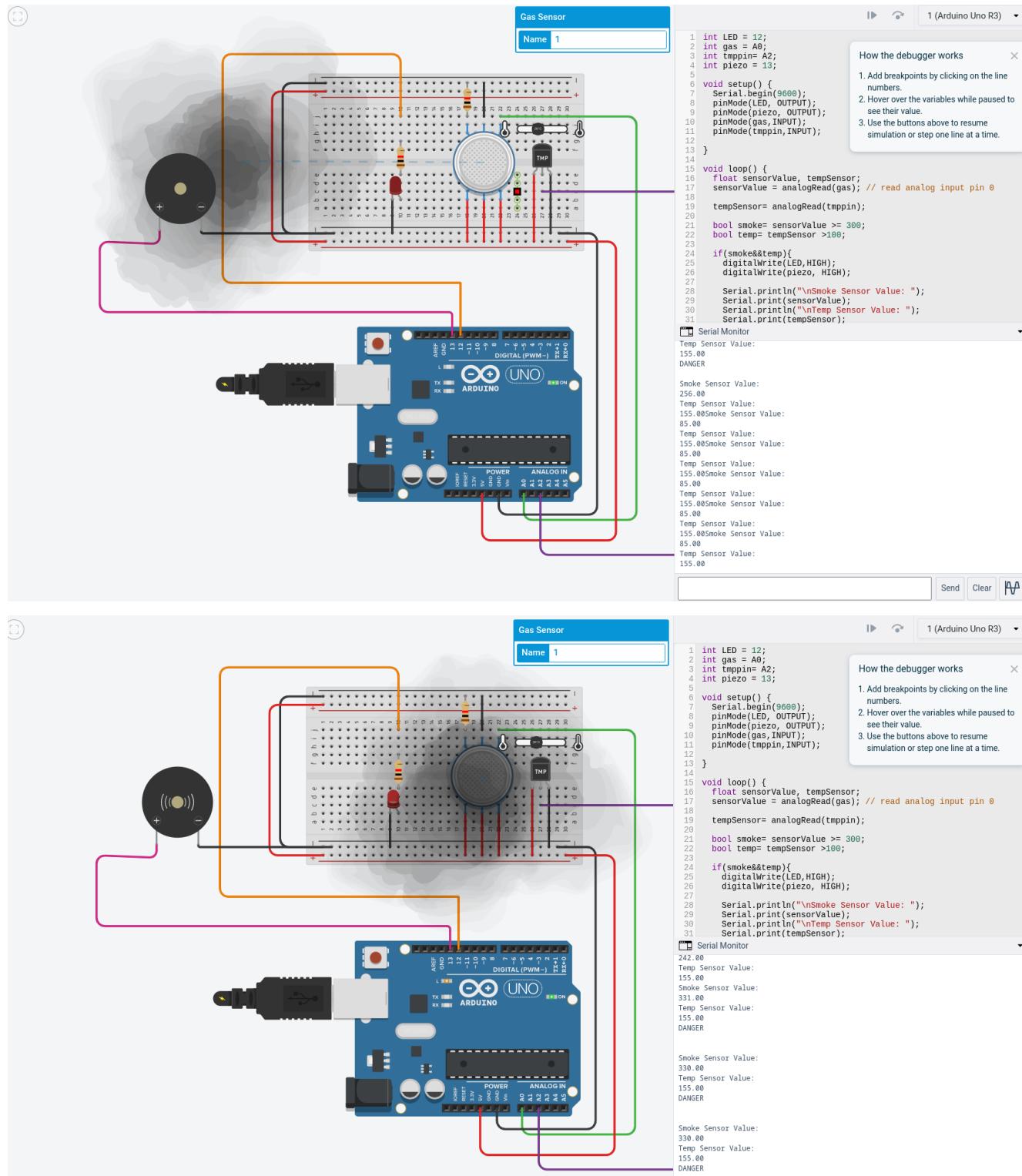
    Serial.println("\nSmoke Sensor Value: ");
    Serial.print(sensorValue);
    Serial.println("\nTemp Sensor Value: ");
    Serial.print(tempSensor);
    Serial.println(" \nDANGER\n");
}

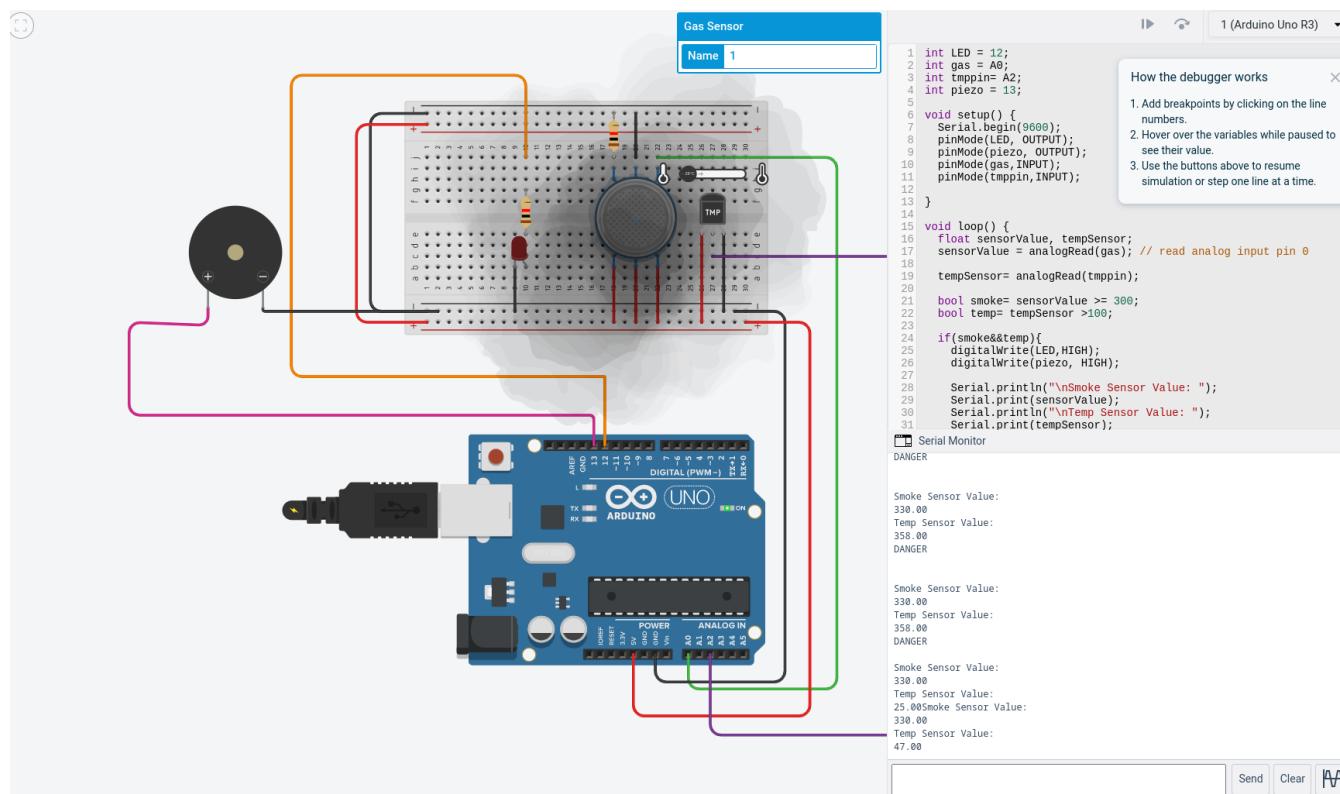
else{
    digitalWrite(LED,LOW);
    digitalWrite(piezo, LOW);
    Serial.println("Smoke Sensor Value: ");
    Serial.print(sensorValue);
    Serial.println("\nTemp Sensor Value: ");
    Serial.print(tempSensor);
}

delay(1000);
}

float getsensorValue(int pin){
    return (analogRead(pin));
}
```

Output





Lab 8**Date: 18-04-2022****Questions**

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 color of the LED based on the 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.

24 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 color of the LED based on the some threshold values.

Question

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 color of the LED based on the some threshold values.

Aim

To interface force sensor with Arduino board and display the amount of force given as input to sensor in LCD screen. We must also interface LED RGB and change color of the LED based on the some threshold values.

Code

```
// C++ code
//
#include<LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10, 9, 8, 7);

int force = 0;
int red = 4;
int blue = 3;
int green = 2;
void setup()
{
    lcd.begin(16, 2);
    pinMode(A0, INPUT);
    pinMode(green, OUTPUT);
    pinMode(red, OUTPUT);
    pinMode(blue, OUTPUT);
    Serial.begin(9600);
}

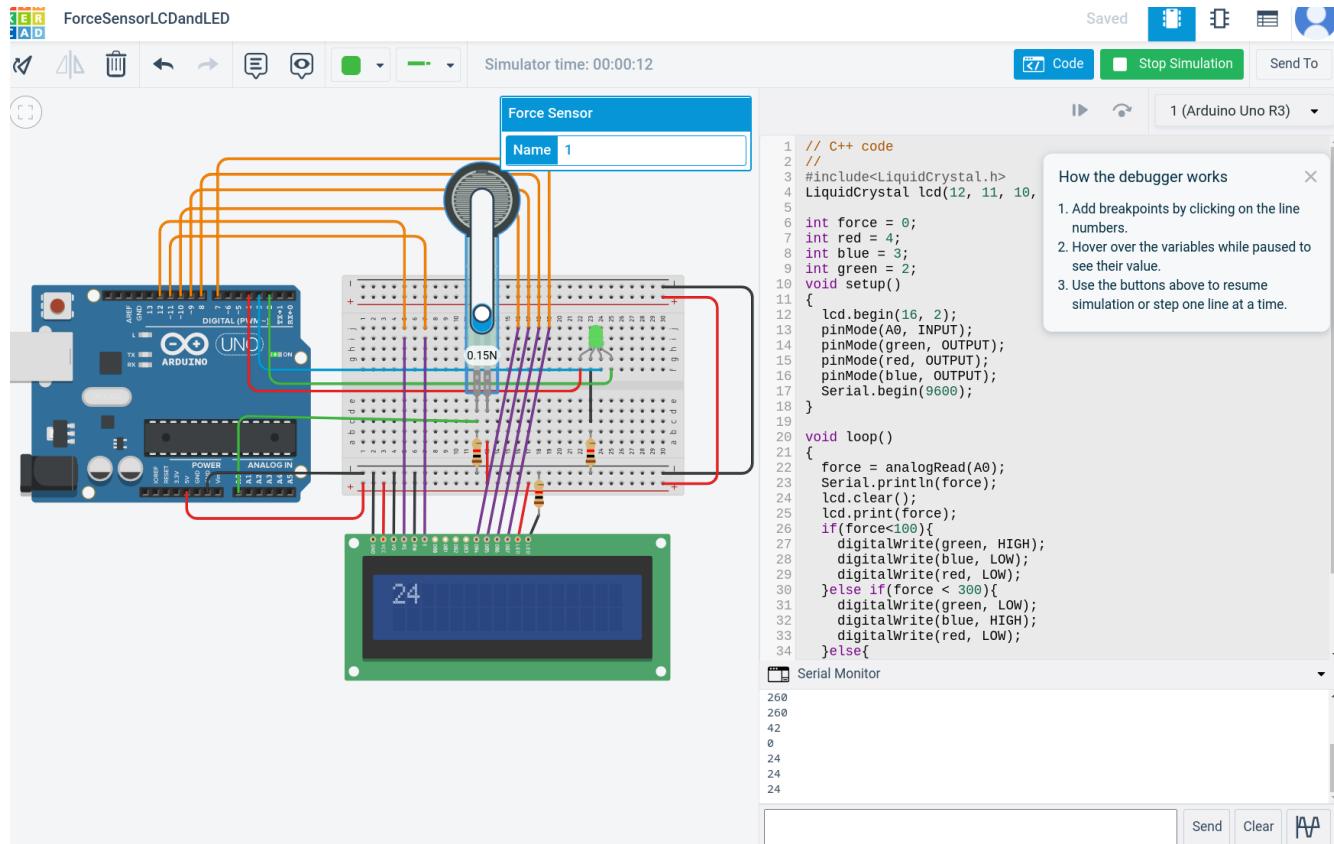
void loop()
{
    force = analogRead(A0);
    Serial.println(force);
    lcd.clear();
    lcd.print(force);
    if(force<100){
        digitalWrite(green, HIGH);
        digitalWrite(blue, LOW);
        digitalWrite(red, LOW);
    }else if(force < 300){
        digitalWrite(green, LOW);
        digitalWrite(blue, HIGH);
        digitalWrite(red, LOW);
    }
}
```

```

}else{
    digitalWrite(green, LOW);
    digitalWrite(blue, LOW);
    digitalWrite(red, HIGH);
}
delay(1000);
}

```

Output



The screenshot shows a Tinkercad simulation environment. On the left, an Arduino Uno board is connected to a breadboard. A force sensor is attached to the breadboard, with its output connected to pin A0. The breadboard also features a 16x2 LCD display showing the number '240'. Various components like resistors and capacitors are placed on the breadboard. The right side of the screen displays the C++ code for the Arduino sketch, which reads the analog input from the force sensor and prints it to the serial monitor. A sidebar on the right provides instructions on how the debugger works.

```
// C++ code
//
#include<LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10,
5, 4, 3, 2);
int force = 0;
int red = 4;
int blue = 3;
int green = 2;
void setup()
{
  lcd.begin(16, 2);
  pinMode(A0, INPUT);
  pinMode(green, OUTPUT);
  pinMode(red, OUTPUT);
  pinMode(blue, OUTPUT);
  Serial.begin(9600);
}
void loop()
{
  force = analogRead(A0);
  Serial.println(force);
  lcd.clear();
  lcd.print(force);
  if(force<100){
    digitalWrite(green, HIGH);
    digitalWrite(blue, LOW);
    digitalWrite(red, LOW);
  }else if(force > 300){
    digitalWrite(green, LOW);
    digitalWrite(blue, HIGH);
    digitalWrite(red, LOW);
  }else{

```

The screenshot shows a breadboard setup connected to an Arduino Uno. The analog force sensor is connected to pin A0. The digital force sensor is connected to a breadboard power rail. An LCD display is connected to pins 12, 11, and 10. A digital potentiometer is also connected to the breadboard. The code in the editor reads the analog input from A0 and prints the value to the serial monitor.

```
// C++ code
//
#include<LiquidCrystal.h>
LiquidCrystal lcd(12, 11, 10,
5
6 int force = 0;
7 int red = 4;
8 int blue = 3;
9 int green = 2;
void setup()
11 {
12 lcd.begin(16, 2);
13 pinMode(A0, INPUT);
14 pinMode(green, OUTPUT);
15 pinMode(red, OUTPUT);
16 pinMode(blue, OUTPUT);
17 Serial.begin(9600);
18 }
19
void loop()
20 {
21 force = analogRead(A0);
22 Serial.println(force);
23 lcd.clear();
24 lcd.print(force);
25 if(force<100){
26 digitalWrite(green, HIGH);
27 digitalWrite(blue, LOW);
28 digitalWrite(red, LOW);
29 }else if(force < 300){
30 digitalWrite(green, LOW);
31 digitalWrite(blue, HIGH);
32 digitalWrite(red, LOW);
33 }else{
34 }
35 }
```

Serial Monitor

240
240
240
393
436
436
448

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.

Question

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.

Aim

To interface the keypad and tilt sensor with Arduino board. If the sensor is being tilted, we take input from keypad and print it in the LCD.

Code

```
#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 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;</pre>  
    buf[cnt]=0;  
    lcd.clear();  
    lcd.print(buf);  
  
}else{  
    lcd.clear();  
    lcd.print("IDLE");  
}  
  
delay(50);  
}
```

Output

