



Jose Rizal University  
College of Computer Studies Engineering  
Computer Engineering Department

# Lab Experiment Submission No. 2

**CPE C312 – EMBEDDED SYSTEMS**

**Submitted by:**

Justine Angela C. Cristobal

Louie Jay Layderos

Exiquiel John A. Pines

Rom Jordan Resurreccion

**Submitted to:**

Engr. Nastaran Reza Nazar Zadeh

**Date Submitted:**

September 12, 2023

## **I. OBJECTIVES**

Upon completion of this laboratory experiment, students should be able to:

- Understand the basic of input/output operation of the Arduino Uno board.
- Program to read inputs and to produce outputs in response to that input

## **II. MATERIALS & TOOLS**

1. 1 x Arduino Uno board and USB cable
2. 4 x 5mm/3mm Red LEDs
3. 4 x 470 $\Omega$  resistors (for LEDs) and 1 x 10k $\Omega$  resistor (for tact switch)
4. 1 x breadboard
5. 1 x tact switch or toggle switch (whichever is available or simply a pull-up wire)

## **III. BACKGROUND INFORMATION**

1. Open the previous sketch. To do this, select File > Open... or the button located in the Arduino program. Navigate to where you saved the .pde file, select it and click Open.
2. Create a copy of this sketch and rename it to correspond to this new exercise by selecting File > Save As.... By doing this you should now have a new .pde file that contains the code you made during the previous exercise.

## **IV. PROCEDURE**

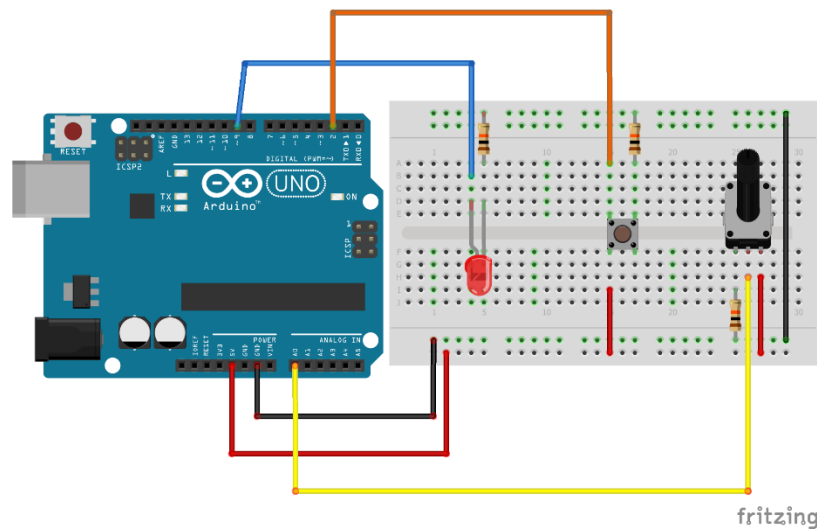
1. Modify the existing code so that instead of only 1 pin, 4 pins are configured to accommodate the 4 LEDs. Note that you have two main functions so choose wisely where you put the initialization and working program. Add another configuration for a pin to be connected to the tact/toggle switch.
2. Create the part of the program that will read the switch's state. Be sure to understand which nodes of the tact/toggle switch are already shorted and not.
3. Make the program that would perform the dancing LED's when the program recognizes the button press. The program must not execute the light effect until the user releases the button. The dancing LED is composed of 4 LED's that will light up 1 at a time starting from LED1 to LED4 and afterwards from LED4 back to LED1 Repeatedly. Code the program efficiently by using looping functions.

4. Once you are done writing the code, save your file and compile it.

## IV. PROCEDURES

Test your program

1. Construct the circuit using the Arduino Uno, Breadboard, LEDs, resistors and tact/toggle switch.
2. Connect the board to the PC using the USB cable. Upload the sketch saved earlier and check if the program works. The LEDs should behave as previously described in Step 3 of the instructions in programming the dancing LED's. Note that the board should not perform the blinking LED exercise anymore since it was overwritten; otherwise, the new program wasn't uploaded properly.



```
int LED = 13;
int BUTTON = 2;

void setup() {
  pinMode(LED,OUTPUT);
  pinMode(BUTTON,INPUT);
}

void loop(){
  if (digitalRead(BUTTON) == HIGH) {
    digitalWrite(LED,HIGH);
  } else {
    digitalWrite(LED,LOW);
  }
}
```

## V. REPORT

Write a report that summarizes this experiment. Your report brief must include:

- **Objective** - The objective of this experiment is to program an Arduino Uno board to control four LEDs in a dancing pattern. The LEDs will light up one by one, starting from LED1 to LED4, and then back from LED4 to LED1. The pattern will repeat until the user presses a button.
- **Procedure** - The experiment was conducted by programming an Arduino Uno board to control four LEDs in a dancing pattern. The LEDs were connected to the Arduino board in a specific order, and the Arduino code was written to turn on the LEDs one by one, starting from LED1 to LED4, and then back from LED4 to LED1. The pattern was repeated until the user pressed a button.
- **Discussion** - The results of the experiment showed that it was possible to program an Arduino Uno board to control four LEDs in a dancing pattern. The LEDs lit up one by one, starting from LED1 to LED4, and then back from LED4 to LED1. The pattern repeated until the user pressed a button.

The principles involved in the experiment include:

- The use of digital logic to control the LEDs.
- The use of loops to repeat the dancing pattern.
- The use of conditional statements to check for the button press.

The experiment was successful in demonstrating the basic principles of Arduino programming. It also showed that it is possible to create simple but visually appealing LED animations using Arduino.

### Code:

```
int button = 2;
int LED1 = 4;
int LED2 = 5;
int LED3 = 6;
int LED4 = 7;
int count = 0;

void setup() {
  pinMode(button, INPUT);

  pinMode(LED1, OUTPUT);
  pinMode(LED2, OUTPUT);
  pinMode(LED3, OUTPUT);
  pinMode(LED4, OUTPUT);
}

void loop() {
  while(true) {
    if(digitalRead(button) == HIGH) {
      count += 1;
      break;
    }
  }
}
```

```

    }
}

switch(count) {
    case 1:
        digitalWrite(LED1, HIGH);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, LOW);
        break;

    case 2:
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, HIGH);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, LOW);
        break;

    case 3:
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, HIGH);
        digitalWrite(LED4, LOW);
        break;

    case 4:
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, HIGH);
        break;

    case 5:
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, HIGH);
        digitalWrite(LED4, LOW);
        break;

    case 6:
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, HIGH);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, LOW);
        break;

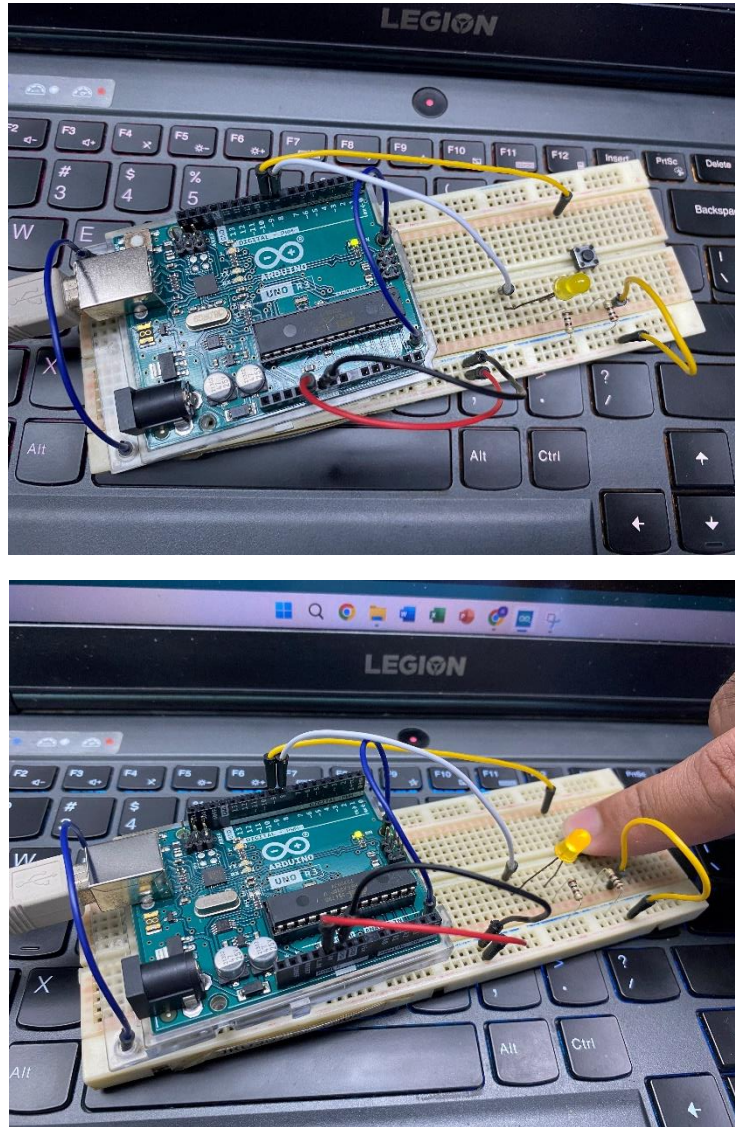
    case 7:
        digitalWrite(LED1, HIGH);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, LOW);
        break;

    default:
        count = 0;
        digitalWrite(LED1, LOW);
        digitalWrite(LED2, LOW);
        digitalWrite(LED3, LOW);
        digitalWrite(LED4, LOW);
        break;
}

delay(1000);
}

```

## Actual Pictures:



## VI. CONCLUSION

In conclusion, we have learned that it is possible to program an Arduino Uno board to control four LEDs in a dancing pattern. The experiment demonstrated the basic principles of Arduino programming, such as the use of digital logic, loops, and conditional statements. It also showed that it is possible to create simple but visually appealing LED animations using Arduino. We also learned that it is important to be careful when writing Arduino code, as even small errors can cause the program to malfunction.