**Microprocessor Lab**

Laboratory Activity No. 2

**Arduino and Tinkercad Interface**

|  |
| --- |
|  |

Score

*Submitted by:*

**Gomez, Ericka Mae S.**

**<Saturday 1:00pm-7:00pm> / <CPE0412.1-2>**

*Date Submitted*

**30-09-2023**

*Submitted to:*

**Engr. Maria Rizette H. Sayo**

I. Objectives

This laboratory activity aims to implement the principles and techniques of hardware programming using Arduino through:

- creating an Arduino programming and circuit diagram.

II. Method/s

- Perform a task problem given in the presentation.

- Write a code and perform an Arduino circuit diagram of a ring counter that display eight (8)LEDs starting from left.

III. Results

**TinkerCad**

**Exercise 1: Write a code that does a ring counter display for eight (8) LEDs starting from left.**

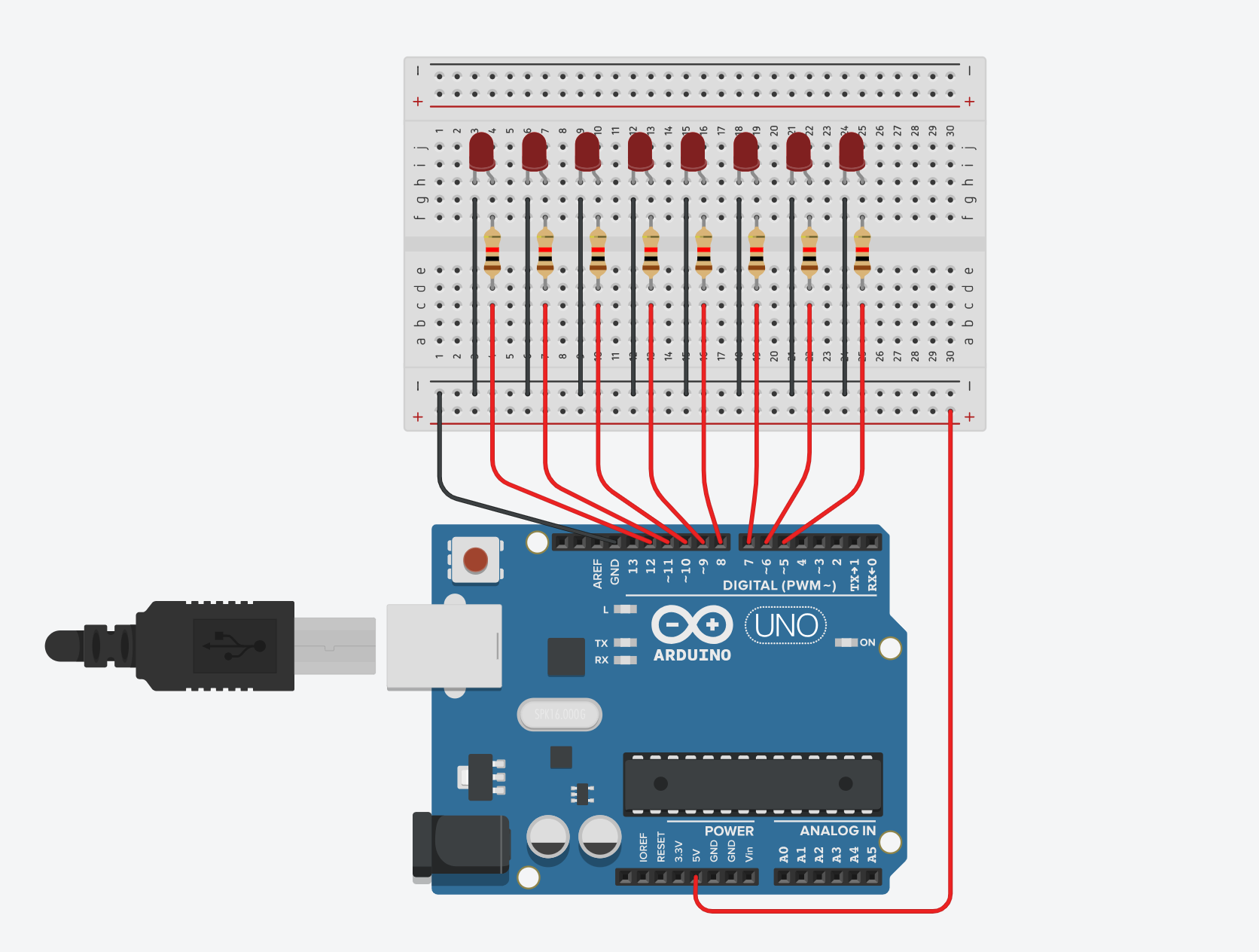


Figure No.1 Ring Counter Display Circuit Diagram

**Components Used**

1. 8 LEDs
2. Resistor
3. Breadboard

**CODE:**

// C++ code

//

/\*

Ring counter display for eight (8) LEDs starting from left.

\*/

void setup()

{

Serial.begin(9600);

pinMode(5, OUTPUT);

pinMode(6, OUTPUT);

pinMode(7, OUTPUT);

pinMode(8, OUTPUT);

pinMode(9, OUTPUT);

pinMode(10, OUTPUT);

pinMode(11, OUTPUT);

pinMode(12, OUTPUT);

}

void loop()

{

digitalWrite(12, HIGH);

delay(500);

Serial.println("The LED1 is HIGH");

digitalWrite(12, LOW);

delay(500);

Serial.println("The LED1 is LOW");

digitalWrite(11, HIGH);

delay(500);

Serial.println("The LED2 is HIGH");

digitalWrite(11, LOW);

delay(500);

Serial.println("The LED2 is LOW");

digitalWrite(10, HIGH);

delay(500);

Serial.println("The LED3 is HIGH");

digitalWrite(10, LOW);

delay(500);

Serial.println("The LED3 is LOW");

digitalWrite(9, HIGH);

delay(500);

Serial.println("The LED4 is HIGH");

digitalWrite(9, LOW);

delay(500);

Serial.println("The LED4 is LOW");

digitalWrite(8, HIGH);

delay(500);

Serial.println("The LED5 is HIGH");

digitalWrite(8, LOW);

delay(500);

Serial.println("The LED5 is LOW");

digitalWrite(7, HIGH);

delay(500);

Serial.println("The LED6 is HIGH");

digitalWrite(7, LOW);

delay(500);

Serial.println("The LED6 is LOW");

digitalWrite(6, HIGH);

delay(500);

Serial.println("The LED7 is HIGH");

digitalWrite(6, LOW);

delay(500);

Serial.println("The LED7 is LOW");

digitalWrite(5, HIGH);

delay(500);

Serial.println("The LED8 is HIGH");

digitalWrite(5, LOW);

delay(500);

Serial.println("The LED8 is LOW");

}

IV. Conclusion

The provided C++ code is an Arduino sketch that creates a ring counter display using eight LEDs, starting from the leftmost LED and cycling through them in a circular fashion. The code uses a simple loop to turn each LED on, wait for a brief period, turn it off, and then move to the next LED.

The code begins with the setup function. In Arduino, this function runs once when the board is powered on or reset. Its purpose is to perform any necessary initializations. the code initializes the serial communication with a baud rate of 9600, which is commonly used for debugging purposes. It also sets each of the eight pins (from 5 to 12) as OUTPUT, indicating that these pins will be used to control the LEDs. Following the setup function, we have the loop function. In Arduino, the loop function runs continuously after the setup is completed. This is where the core logic of the code is implemented. In this loop, the code sequentially turns each LED on by setting its respective pin to HIGH, waits for 500 milliseconds using the delay function, and then turns the LED off by setting the pin to LOW. After each LED is turned on and off, it also prints a message to the serial monitor indicating the current state of the LED (HIGH or LOW). This process is repeated for each of the eight LEDs, creating a ring counter effect. The LEDs light up one after the other in a circular manner, and the messages in the serial monitor provide information about which LED is currently active.

In summary, the provided Arduino sketch demonstrates a ring counter display using eight LEDs. It utilizes a loop to control the LEDs, turning them on and off sequentially with a delay between each step. The code also provides informative messages through the serial monitor to indicate the state of each LED. This simple yet effective program can be used for various visual display applications, such as decorative lighting or visual feedback in Arduino projects.

**References**

Please refer to this link:

[Circuit design Lab-Activity-No.2 | Tinkercad](https://www.tinkercad.com/things/6SptePksbIR-brave-gogo/editel?tenant=circuits)