PAMANTASAN NG LUNGSOD NG MAYNILA

(University of the City of Manila)
Intramuros, Manila

Microprocessor Lab

Laboratory Activity No. 2

Arduino and Tinkercad Interface

□

Score

Submitted by:
Castillo, Franchesca Nicole A.
Saturday 7AM – 1PM / BS CpE 4-1

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.

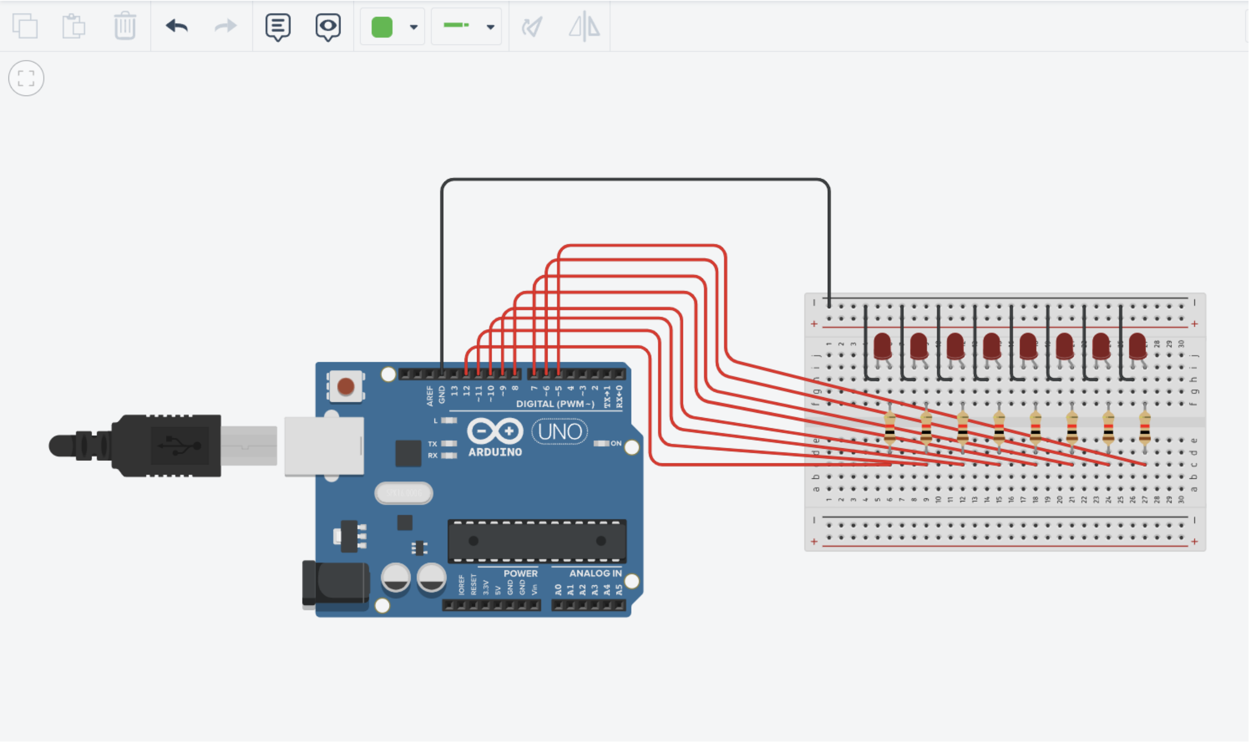
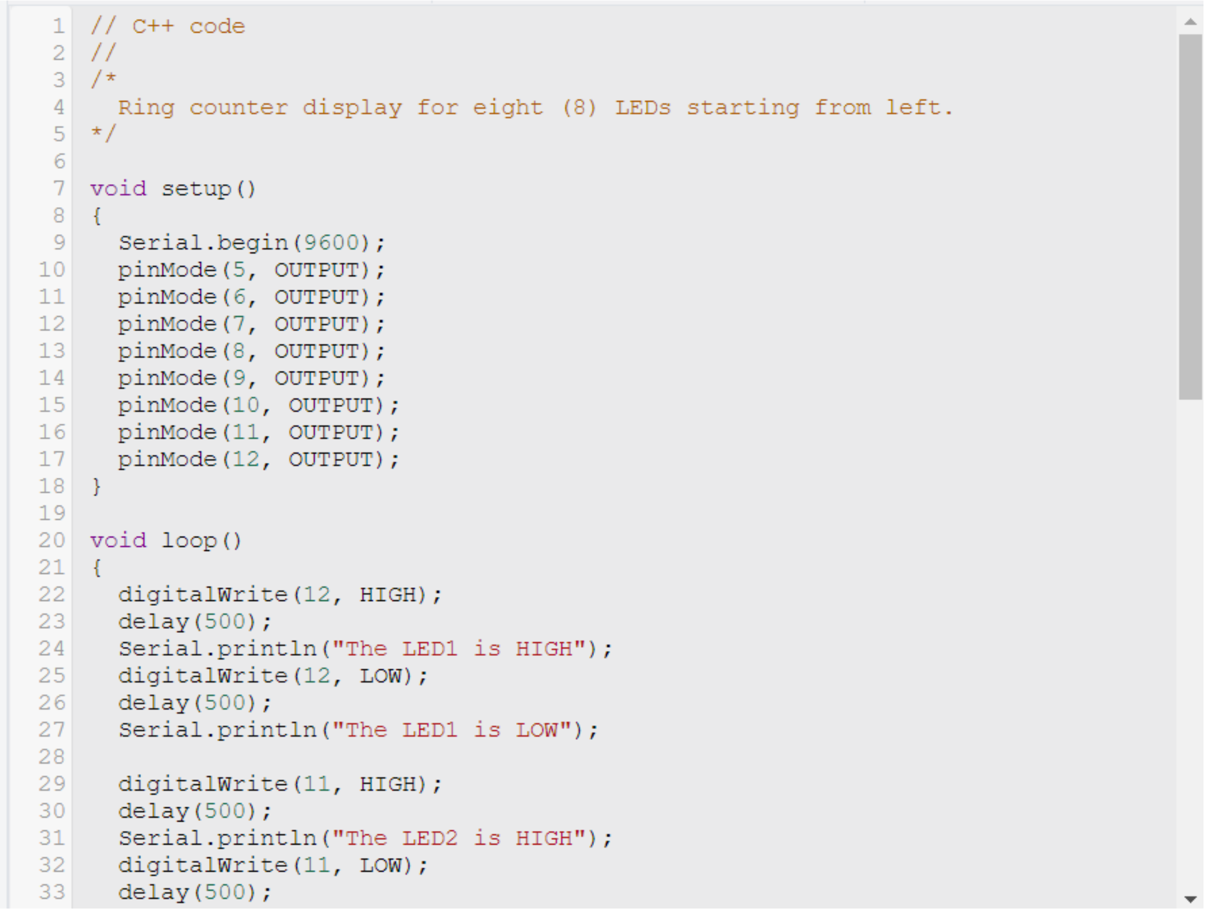


Figure No.1 Ring Counter Display Circuit Diagram

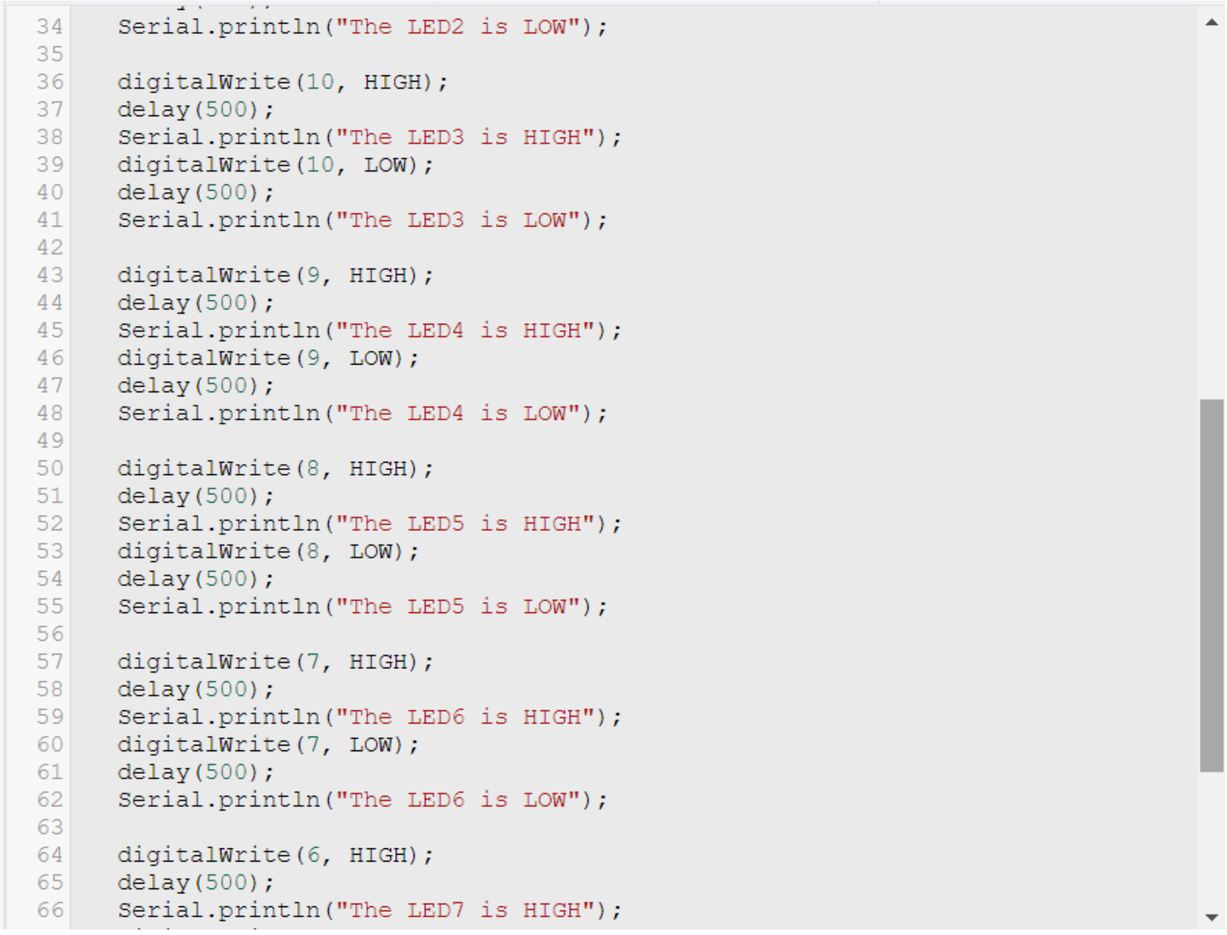
Components Used

1. 8 LEDs
2. Resistor
3. Breadboard

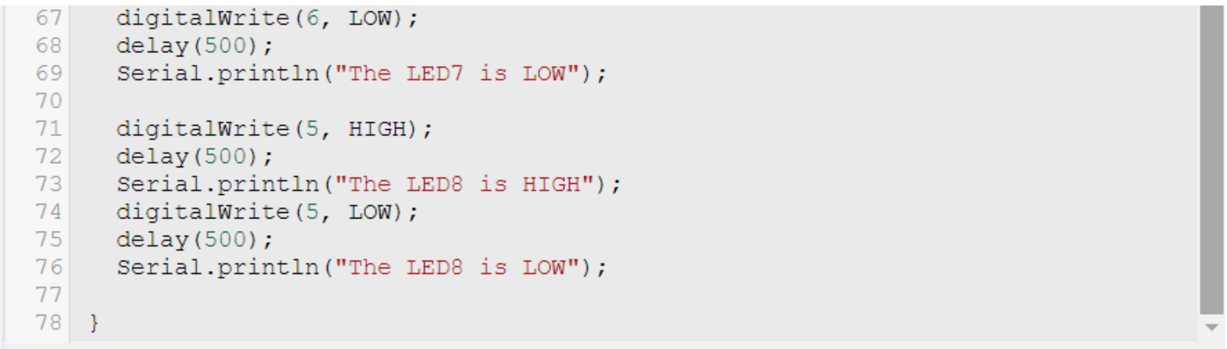
CODE:



```
// 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);
```

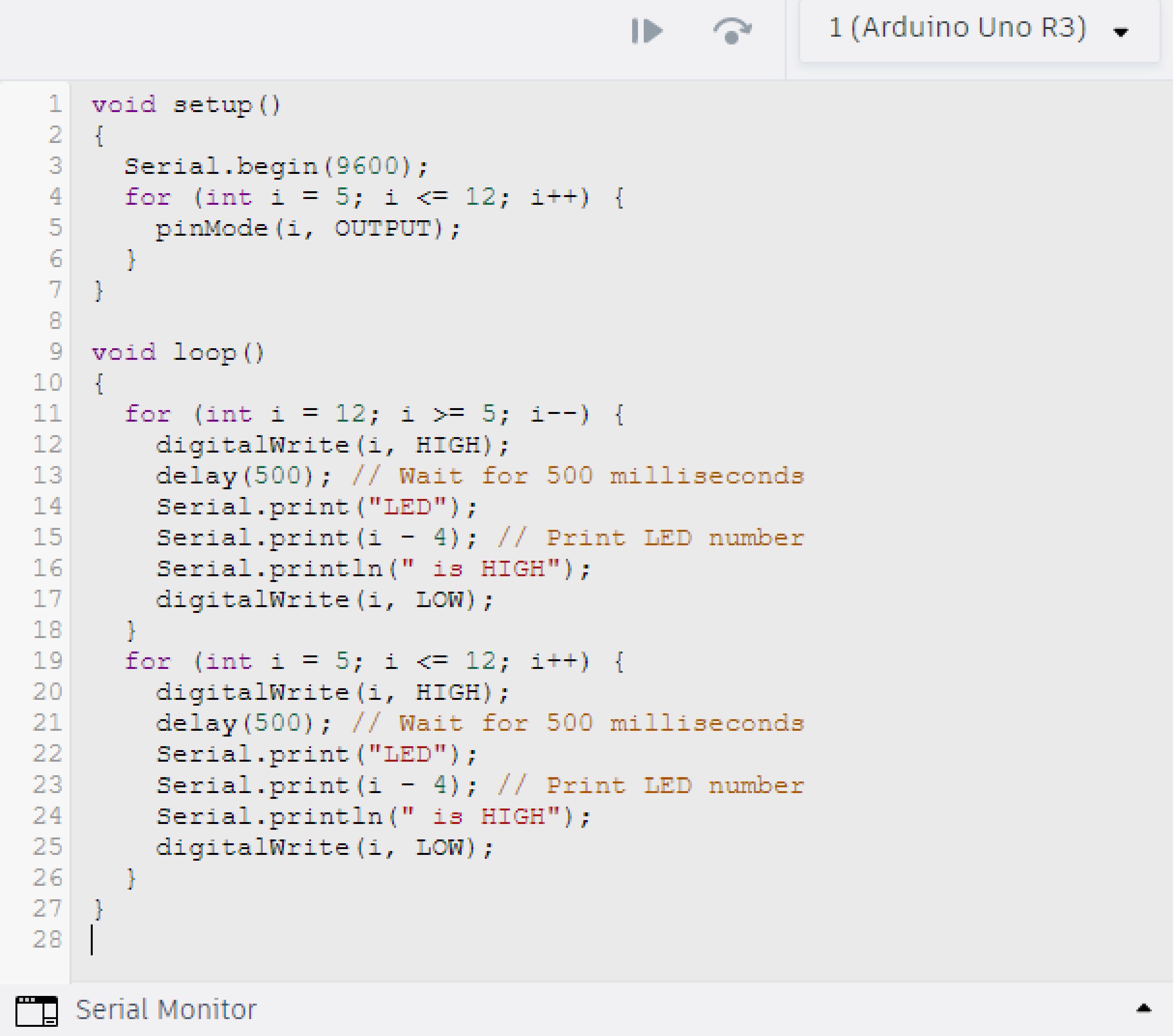


```
  Serial.println("The LED2 is LOW");

  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");

}
```

Other possible program:



```
void setup()
{
  Serial.begin(9600);
  for (int i = 5; i <= 12; i++) {
    pinMode(i, OUTPUT);
  }
}

void loop()
{
  for (int i = 12; i >= 5; i--) {
    digitalWrite(i, HIGH);
    delay(500); // Wait for 500 milliseconds
    Serial.print("LED");
    Serial.print(i - 4); // Print LED number
    Serial.println(" is HIGH");
    digitalWrite(i, LOW);
  }
  for (int i = 5; i <= 12; i++) {
    digitalWrite(i, HIGH);
    delay(500); // Wait for 500 milliseconds
    Serial.print("LED");
    Serial.print(i - 4); // Print LED number
    Serial.println(" is HIGH");
    digitalWrite(i, LOW);
  }
}
```

In the latest code, the LEDs are initially illuminated in a sequential manner, commencing from the leftmost LED (pin 12) and progressing towards the rightmost LED (pin 5). Subsequently, the LEDs are illuminated in a reverse sequence, commencing from the rightmost LED (pin 5) and progressing towards the leftmost LED (pin 12). This sequential illumination pattern is executed in a continuous loop. This configuration results in the manifestation of an 8-LED ring counter that undergoes a cyclic progression from the leftmost position to the rightmost position, followed by a reversal from the rightmost position back to the leftmost position.

The code provided demonstrates traits that resemble those of a ring counter, as it sequentially activates LEDs in a loop, resulting in a perpetual cycling effect. Nevertheless, it shifts from the conventional notion of a ring counter in two essential manners. In constructing a standard ring counter, flip-flops are interconnected in a circular or ring-like configuration, where the output of each flip-flop is connected to the input of the subsequent flip-flop. On the other hand, this code utilizes an array and software logic to imitate the sequential activation of LEDs, without the presence of physical flip-flops. Furthermore, it is important to note that although a ring counter is typically employed for tasks such as pulse counting or clock edge detection, the code in question largely concentrates on the sequential illumination of LEDs and does not encompass these aforementioned counting activities. Notwithstanding these delineations, the code functions as a streamlined simulation of the operational characteristics of a ring counter, providing a foundational framework for the creation of LED animations and visual effects.

IV. Conclusion

The concept of ring counters is that they are a type of sequential circuit that can be used to count pulses. They are made up of flip-flops that are connected in a ring. The output of one flip-flop is connected to the input of the next flip-flop. This creates a loop that allows the counter to count continuously. Ring counters can be used in a variety of applications, such as frequency counters, digital clocks, and timers [1].

In conclusion, the provided code offers a simplified depiction of the attributes of a ring counter. However, it deviates from conventional ring counters by utilizing software-based logic and omitting certain inherent functions. While it may not fully embody the traditional concept of a ring counter with interconnected flip-flops, it serves as a practical starting point for creating LED animations and visual effects in the context of Arduino programming.

Furthermore, it's worth noting that tools like Tinkercad have greatly facilitated hands-on learning experiences, such as the laboratory activity described. Tinkercad, a free web-based 3D modeling program, empowers users, including students and educators, to design and prototype objects easily. Its versatility in creating 3D designs, circuits, and coding within a single platform makes it an invaluable resource for teaching and learning about electronics and digital design [2].

References

[1]T. Agarwal, “Ring Counter : Working, Classification and Its Applications,” *ElProCus - Electronic Projects for Engineering Students,* Mar. 07, 2020. [Online]. Available: https://www.elprocus.com/ring-counter-in-process/

[2]K. Kennedy and K. Kennedy, “What is Tinkercad? - Product Design Online,” *Product Design Online - Learn Fusion 360 faster with concise step-by-step tutorials.*, Feb. 18, 2020. [Online]. Available: https://productdesignonline.com/tips-and-tricks/what-is-tinkercad/