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:
Ibanez, John Nicole F.
Isidro, Jerome H.
Laurente, Queenie D.
Malabago, Antonio Emmanuel C.
S 1:00-7:00PM / CPE 0412-2

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 and TinkerCAD Interface. Below are the specific objectives of this activity:

1. To familiarize with the Arduino and TinkerCAD interface and learn how to navigate and utilize their features effectively.
2. To understand the concept of a ring counter display, and its implementation using various LEDs.
3. To design a circuit diagram for a ring counter display by using multiple LEDs for visually displaying the output.
4. To integrate the appropriate LED connection and explain the significance of resistors in the circuit.

II. Methodology

The following components were used for the implementation of a ring counter display:

1. Arduino Uno R3 (1)
2. Breadboard (1)
3. 1kΩ resistors (8)
4. LEDs (8)
5. Connecting wires

To accomplish the objectives of the laboratory, the detailed procedures for designing and implementing a ring counter display are enumerated below:

1. Prepare the Arduino UNO R3, breadboard, and LEDs to be used.
2. Develop logic algorithms for the ring counter display and program the corresponding code.
3. Simulate the hardware architecture and design in the TinkerCAD interface.
4. Replicate the layout into physical hardware and upload the code into Arduino Uno R3**.**
5. Document results, observations, and outcomes.

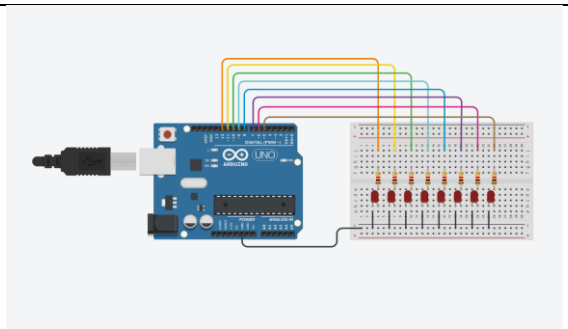

Figure 1. Setup of the ring counter display in simulator.

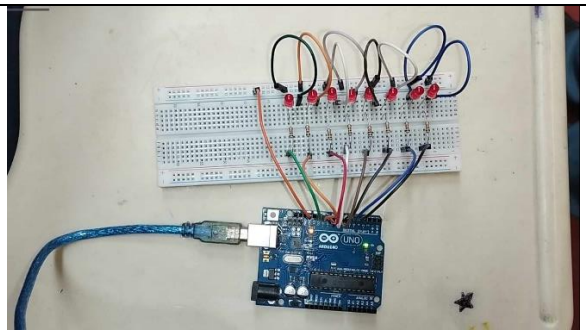

Figure 2. Setup of the ring counter display in actual.

As shown in the figure above, the set-up connection of the ring counter display was implemented in both the TinkerCAD simulator and the actual breadboard. The eight (8) LEDs

were interfaced linearly in the breadboard. Every cathode of LED (longer leg) was connected to digital pins of the Arduino Uno R3, regulated by a 1kΩ resistor. On the other hand, their anode (shorter leg) was connected directly to the ground. Each digital pin has a voltage capacity of 5V, which significantly exceeds the forward voltage rating of the red LED (1.8V).

```
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");
}
```

Figure 3. Code for the ring counter display

Figure 3 illustrates the coding implementation for the ring counter display with C++ programming language. Several functions were created to perform modularized functionalities, including a setup function that initializes the serial monitor and LED digital pins, and a loop function that iteratively executes the instruction within them. The latter function will be executed continuously until the simulator is manually stopped.

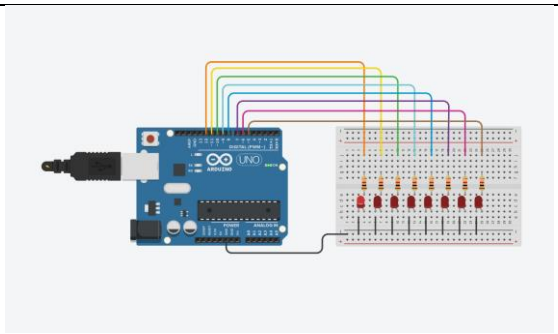

Figure 1. Circuit simulation in TinkerCAD.

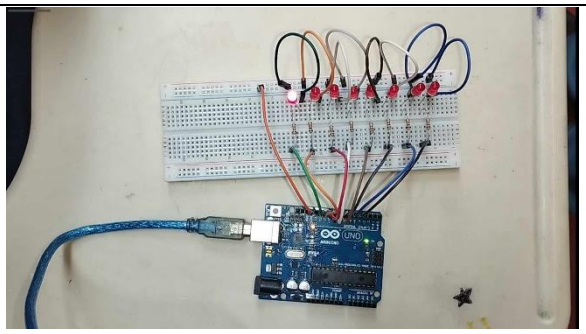

Figure 2. Circuit implementation in actual breadboard

The figure above shows the ring counter display simulation in the TinkerCAD interface and its implementation in the actual breadboard. Each LED turns on when the assigned state to its digital pin is HIGH. However, the LED turns off when a LOW value is assigned to the digital pin. Also, a delay of 500ms was programmed in between their HIGH and LOW values. This results in successive blinking of LEDs from left to right. Lastly, a string text is also printed on the serial monitor to inform the user of the recent state of the ring counter display.

III. Results

The implementation of ring counter display was successful without any encountered errors and bugs. All objectives provided above were met. The links below provide a video of the operation of the circuit in both the simulation and the actual breadboard.

1. To simulate the program, please follow this link: https://www.tinkercad.com/things/2QW0wBOGgw1-ingenious-lahdi/editel?sharecode=P42s12mIqV9z3NjNKSsY8BAOZozw2y2SwIYGYeUEuII
2. To view the actual implementation of the program, please follow this link: https://plmedu-my.sharepoint.com/:v:/g/personal/aecmalabago2020_plm_edu_ph/Ec2eNY0S6VRNnpmIxzqYfMMBiLUacTKPdfPzsGyJ5YFD3A?nav=eyJyZWZlcnJhbEluZm8iOnsicmVmZXJyYWxBcHAiOiJPbmVEcml2ZUZvckJ1c2luZXNzIiwicmVmZXJyYWxBcHBQbGF0Zm9ybSI6IldlYiIsInJlZmVycmFsTW9kZSI6InZpZXciLCJyZWZlcnJhbFZpZXciOiJNeUZpbGVzTGlua0RpcmVjdCJ9fQ&e=hi7nGI

IV. Conclusion

This laboratory activity introduces the interface of Arduino and TinkerCAD through the design of a ring counter display using Arduino Uno R3, LEDs, 1kΩ resistors, and a breadboard. In hardware design, a ring counter is a type of counter where flip-flops are arranged and connected to form a shift register [1]. The output of the last flip-flop is sent as an input to the next flip-flop, forming a circular loop structure. Ring counters can be categorized as straight or twisted rings. The straight-ring counter connects the output of the previous last register to the input of the first shift register, circulating the same bit around the ring. Meanwhile, the twisted ring counter complements the output of the last register and then connects it to the input of the first shift register, circulating an alternative stream of bits around the ring [2]. For this activity, a straight-ring counter display is required.

To construct the circuit, the cathode side of the LEDs (the negative or shorter leg) is placed to the ground while its anode side (the positive or longer leg) is connected to the digital pins of the Arduino Uno 3 [3]. Based on the observation, a 1kΩ resistor should be coupled to every LED to restrict the flow of excessive voltage, which could burn out the filament inside the LEDs [4]. Afterward, the action of each LED is programmed in the code editor of TinkerCAD. In the setup function, the digital pin mode for each LED was initialized as output, and the serial monitor was included to display the recent state of the ring counter in the serial monitor. Meanwhile, all commands that require repetition are placed in the loop function. Each LED was turned on or off, and a delay of 500ms was included. This process was repeated for each LED from left to right, creating an array of blinking LEDs that mimic the behavior of the ring counter display.

V. References

[1] GeeksforGeeks, “Ring counter in digital logic,” *GeeksforGeeks*, Jan. 2023, [Online]. Available: https://www.geeksforgeeks.org/ring-counter-in-digital-logic/
[2] “Ring counter in digital Electronics - Javatpoint,” *www.javatpoint.com*. https://www.javatpoint.com/ring-counter-in-digital-electronics
[3] “Blink,” *Arduino*. https://www.arduino.cc/en/Tutorial/BuiltInExamples/Blink
[4] R. Keim, “Resistors for LED circuits,” *Resistor Applications | Resistor Guide*, [Online]. Available: https://eepower.com/resistor-guide/resistor-applications/resistor-for-led/