PAMANTASAN NG LUNGSOD NG MAYNILA
(University of the City of Manila)
Intramuros, Manila

Microprocessor Lab

Laboratory Activity No. 3

Binary Representation of 8 LEDs in TinkerCad and Arduino Programming

Score

Submitted by:
Bergola, Khryx Rhoien D.
Saturday 10:00 AM – 1:00 PM / CPE 0412.1-1 Microprocessors

Date Submitted
8-10-2023

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 circuit of Binary representation (decimal 0-255 using 8 LEDs)

II. Method/s

Write a code and perform an Arduino circuit diagram of a binary counter that displays using 8 LEDs with decimals equivalent from 0-255.

III. Results

Write a code that does a binary counter display from 0-255 using eight (8) LEDs.

TinkerCad Setup:

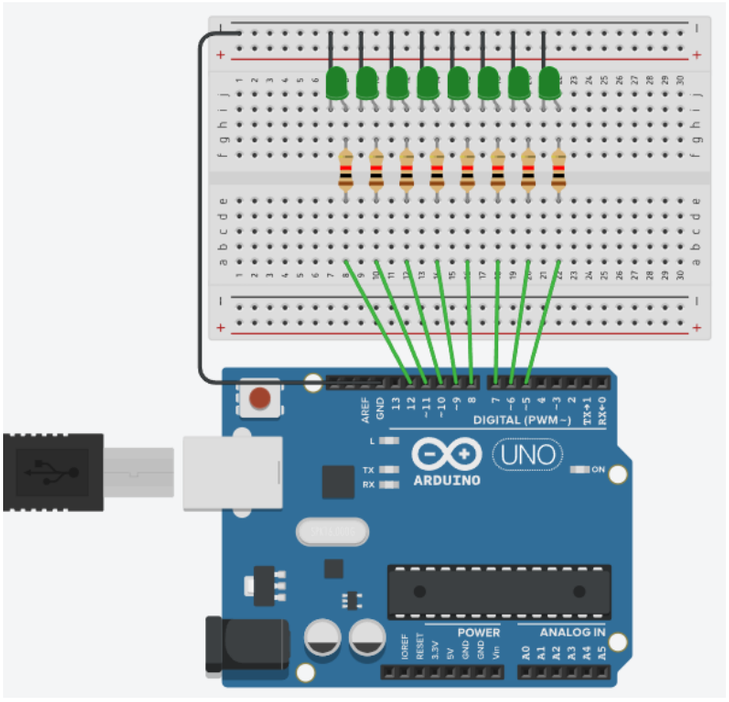


Figure 1. Binary Counter Display Circuit Diagram

Components Used

1. 8 LEDs
2. 1k Resistor
3. Breadboard
4. Arduino UNO

CODE:

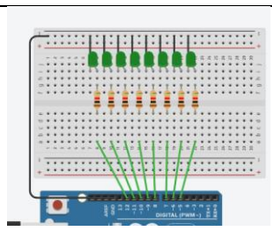
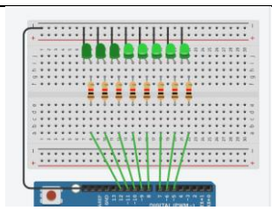
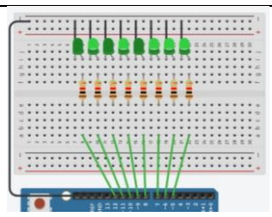
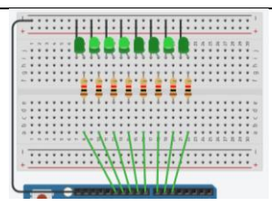
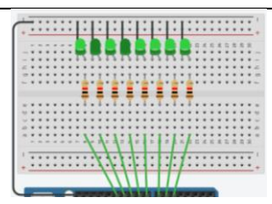
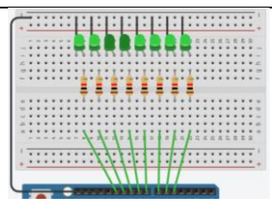
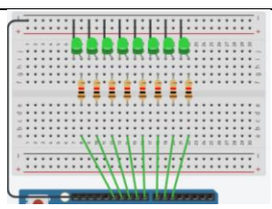
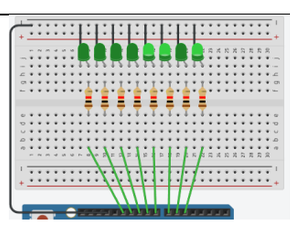
```
const int led[] = {5, 6, 7, 8, 9, 10, 11, 12};

void setup()
{
  Serial.begin(9600);
  for (int i = 0; i < 8; i++) {
    pinMode(led[i], OUTPUT);
  }
}

void loop()
{
  int count = 0;
  for (int dec = 0; dec < 256; dec++) {
    int bin = dec;
    count++;
    // Turn on the LEDs corresponding to the binary value
    for (int i = 0; i < 8; i++) {
      digitalWrite(led[i], (bin & (1 << i)) != 0);

    }
    Serial.print("Decimal value = ");
    Serial.print(dec);
    Serial.print("\n");
    delay(500);
  }
}
```

Result:

Decimal Value	Setup	Serial Monitor
0		Serial Monitor Decimal value = 0
31		Serial Monitor Decimal value = 24 Decimal value = 25 Decimal value = 26 Decimal value = 27 Decimal value = 28 Decimal value = 29 Decimal value = 30 Decimal value = 31
87		Serial Monitor Decimal value = 80 Decimal value = 81 Decimal value = 82 Decimal value = 83 Decimal value = 84 Decimal value = 85 Decimal value = 86 Decimal value = 87
114		Serial Monitor Decimal value = 107 Decimal value = 108 Decimal value = 109 Decimal value = 110 Decimal value = 111 Decimal value = 112 Decimal value = 113 Decimal value = 114
175		Serial Monitor Decimal value = 168 Decimal value = 169 Decimal value = 170 Decimal value = 171 Decimal value = 172 Decimal value = 173 Decimal value = 174 Decimal value = 175
207		Serial Monitor Decimal value = 200 Decimal value = 201 Decimal value = 202 Decimal value = 203 Decimal value = 204 Decimal value = 205 Decimal value = 206 Decimal value = 207
255		Serial Monitor Decimal value = 248 Decimal value = 249 Decimal value = 250 Decimal value = 251 Decimal value = 252 Decimal value = 253 Decimal value = 254 Decimal value = 255
Reset		Serial Monitor Decimal value = 248 Decimal value = 249 Decimal value = 250 Decimal value = 251 Decimal value = 252 Decimal value = 253 Decimal value = 254 Decimal value = 255 Decimal value = 0 Decimal value = 1 Decimal value = 2 Decimal value = 3 Decimal value = 4 Decimal value = 5 Decimal value = 6 Decimal value = 7 Decimal value = 8 Decimal value = 9 Decimal value = 10 Decimal value = 11 Decimal value = 12 Decimal value = 13

IV. Conclusion

This experiment shows the circuit of a Binary Counter with equivalent decimal values of 0 – 255 using 8 LEDs on a breadboard. The binary numbers can be easily solved by the power of two (2^{n-1}) where n is the number of LEDs on the board.[1] This circuit has 8 LEDs which means it has the value of 2^7 which is equal to 128, which is the highest single lighted LED. Adding all values of LEDs will equate to a maximum value of 255.

The use of shift (<<) is important in showing the process of Binary display. The left shift operator (<<) causes bits to be shifted to the left specified by the right operand.[2] This imitates the process of adding ones (1) in the binary such as 1 + 1 = 10. Setting-up the serial monitor is also important by writing Serial.begin(9600) in the set-up to show the current decimal value.

References

[1] *Powers of Two*. Powers of two. (n.d.). https://babbage.cs.qc.cuny.edu/courses/cs341/Powers_of_Two.html

[2] << - Arduino Reference. (n.d.). https://www.arduino.cc/reference/tr/language/structure/bitwise-operators/bitshiftleft/