Electrical Circuit Construction

Electronics Engineering

[Task 1]

Mohamad Abdulmoula,R. Rama

ramamohameed9@gmail.com

Department of Software Engineer

Under the Supervision Of:

Eng Asmaa Duramae

July 5, 2025

Contents

Glossary of Terms

Term	Definition
Arduino UNO R3	A microcontroller board based on the ATmega328P, used for building electronic projects.
LED	Light Emitting Diode; a semiconductor device that emits light when current passes through it.
Push Button	A momentary switch used to allow user input by completing a circuit when pressed.
Breadboard	A board used for constructing and testing electronic circuits without soldering.
Resistor	An electronic component that limits the flow of current in a circuit.
Jumper Wires	Wires used to make electrical connections between components on a breadboard or Arduino.
INPUT_PULLUP	Arduino pin mode that uses an internal resistor to keep the input HIGH until pulled LOW.
Tinkercad	A web-based simulation tool used for 3D modeling and electronics prototyping.

Project Overview

In this project, we developed a simple interactive electronic circuit using Arduino UNO R3 and a pushbutton. The system was designed and simulated in Tinkercad to make the LED light up only while the button is being pressed, and turn off when released. The development process followed standard phases including Planning, Requirements Analysis, Design, Coding, and Testing.

1. Planning

We planned to build an interactive electrical circuit using a push button to control an LED, with Arduino UNO R3 serving as the processing unit. Before physical implementation, we chose to simulate the entire system using Tinkercad to verify its functionality and ensure correct operation.

2. Requirements Analysis

Hardware Components

We added the following components to build the circuit:

a) Arduino UNO R3
b) LED
c) 220Ω Resistor
d) Push Button
e) Small Breadboard
f) Jumper Wires

Functional Requirements

a) The LED should light up only while the button is pressed.
b) The LED should turn off immediately when the button is released.
c) The system should work in real-time, continuously.

3. Design

Circuit Diagram Description

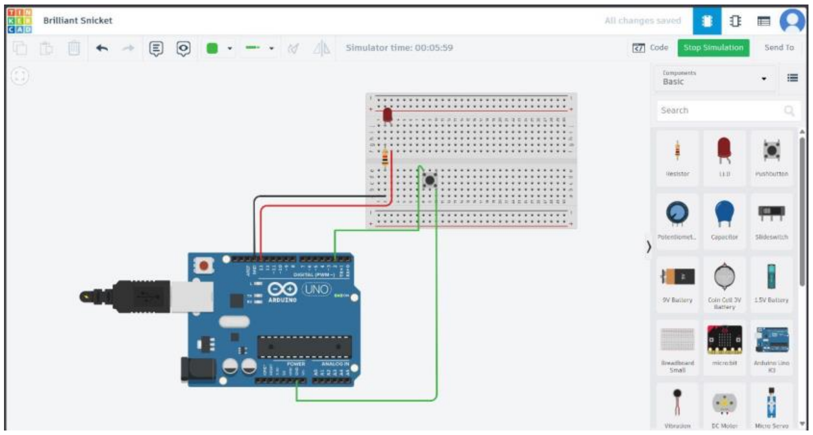


Figure 1 Button Not Pressed

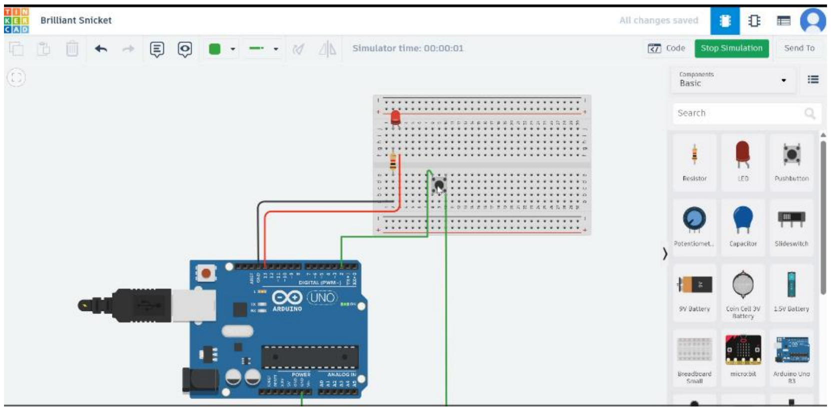


Figure 2 Button Pressed

We designed the circuit as follows:

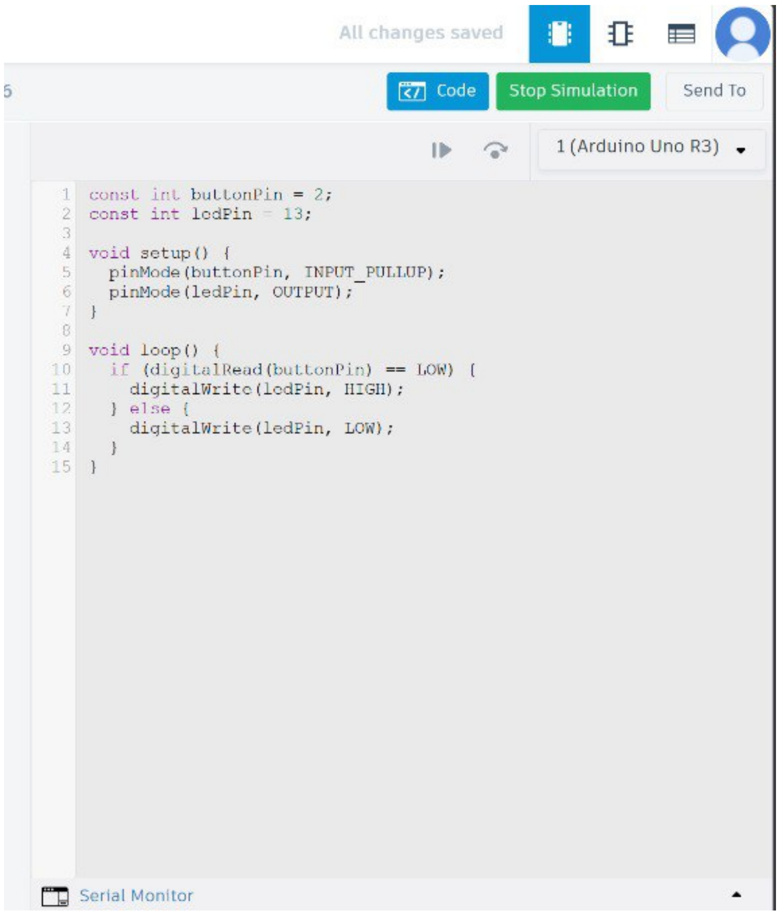
The anode (long leg) of the LED was connected to pin 13 on the Arduino, while the cathode was connected to GND through a 220Ω resistor. One leg of the push button was connected to digital pin 2, and the other leg was connected to GND through a 10kΩ pull-down resistor. That same side of the button was also connected to 5V, allowing the pin to read HIGH when pressed.

Figure 1 shows the circuit in its default state, where the button is not pressed and the LED remains OFF.

Figure 2 illustrates the active state, where pressing the button causes the Arduino to detect a LOW signal and turn the LED ON.

4. Coding

Here is the code we uploaded to the Arduino:



```
const int buttonPin = 2;
const int ledPin = 13;

void setup() {
  pinMode(buttonPin, INPUT_PULLUP);
  pinMode(ledPin, OUTPUT);
}

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

Figure 3 Arduino code for controlling an LED using a push button

In this code, we control an LED using a push button connected to an Arduino. The button is connected to pin 2 and the LED to pin 13. We use INPUT_PULLUP for the button, which means the pin normally reads HIGH and goes LOW when the button is pressed. In the loop(), the Arduino checks if the button is pressed (reads LOW). If it is, the LED is turned ON by sending HIGH to pin 13. If the button is not pressed, the LED is turned OFF by sending LOW to the same pin.

5. Testing

We tested the circuit and observed the following:

a) When we pressed the button, the LED turned ON.
b) When we released the button, the LED turned OFF.
c) The response was instant and stable in the Tinkercad simulation.
d) No toggle behavior was observed—LED only lights during button press, as intended.

Note: ***A recorded test demonstration is provided below to validate the correct operation of the circuit as designed.***

Testing.wmv

Conclusion

In this project, we successfully:

a) Built a real-time LED control system using Arduino UNO and a push button.
b) Used basic components: LED, resistor, breadboard, and jumper wires.
c) Ensured the LED only lights when the button is pressed.
d) Followed a full project lifecycle including planning, design, simulation, and testing using Tinkercad.

References

1. Arduino Official Documentation – https://www.arduino.cc

2. Tinkercad by Autodesk – https://www.tinkercad.com

3. Arduino Reference: https://www.arduino.cc/reference/en/

4. Electronics Tutorials – https://www.electronics-tutorials.ws