UNIVERSITY OF INFORMATION TECHNOLOGY AND SCIENCES

Project Report -

Traffic Light Control

Submitted By –

Name: Md. Al Shahriar Labib (2125051028)
Name: Md. Riad Hossain (2125051029)
Name: Md. Abdullah Al Mamun (2125051030)
Name: Tanvir Ahmed Apu (2125051045)
Batch: 50
Section: 7A

Submitted To-

Name: Propa Punam
Designation: Lecturer of UITS
Course Title: Internet of Things Lab
Course Code: CSE 402

Submission Date: 15/12/2024

Abstract

This project aims to design and implement a traffic light control system using IoT technology. The system replicates a real-world traffic signal, providing a functional and scalable solution for traffic management. By utilizing an Arduino Uno as the microcontroller and basic electronic components, the project demonstrates the fundamental working principle of traffic light control and its potential applications in smart city development.

Objectives

1. To design a simple, functional model of a traffic light system.
2. To understand and implement the working principles of Arduino programming.
3. To simulate real-world traffic signal operations for educational and demonstrative purposes.
4. To explore the integration of IoT concepts in traffic control systems.

Equipment and Components

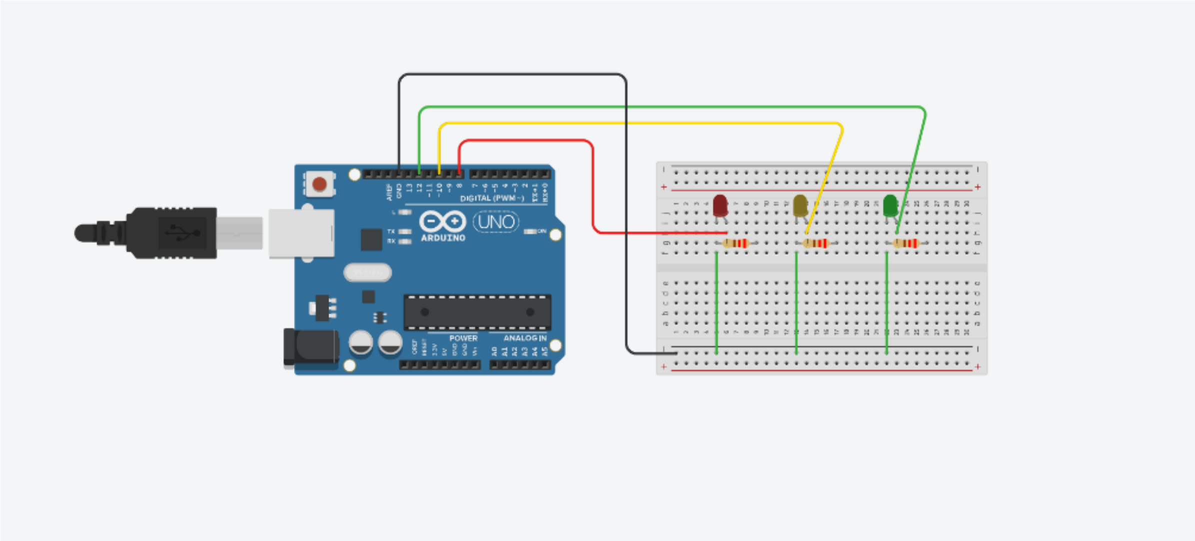
Hardware:

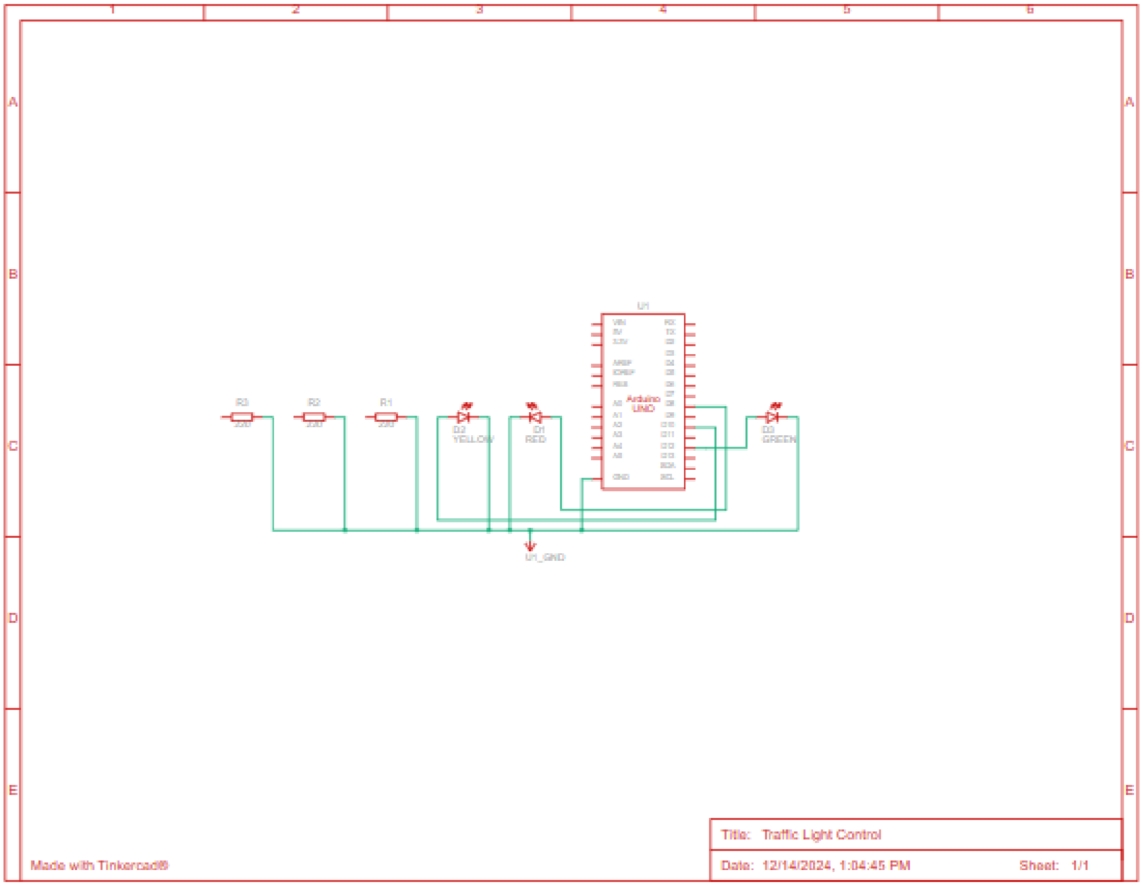
- Arduino Uno
- LED lights (Red, Yellow, Green)
- Breadboard
- Jumper wires
- Resistors (220 Ω)
- USB cable for Arduino programming
- Power source (if required for standalone operation)

Software:

- Arduino IDE (Integrated Development Environment)

Circuit Diagrams





Key Connections

1. The LED lights (Red, Yellow, Green) are connected to the Arduino Uno’s digital output pins.
2. Resistors are connected in series with each LED to limit the current and protect the LEDs from damage.
3. The Arduino Uno is connected to the computer via a USB cable for programming.
4. A common ground is established between the Arduino and the components to ensure proper circuit functioning.

Theory

Traffic lights operate on a sequential control logic to manage vehicle flow at intersections. Each light (Red, Yellow, Green) corresponds to a specific traffic command:

- Red: Stop
- Yellow: Prepare to stop
- Green: Go

Using the Arduino Uno, the sequence is implemented through timed digital signals controlling the LEDs. The resistors ensure that the LEDs operate within their safe voltage and current ratings.

Methodology

1. Designing the Circuit:

- Place the LEDs on the breadboard.
- Connect the anodes of the LEDs to the designated digital pins on the Arduino Uno.
- Connect resistors in series with each LED.
- Establish a common ground connection.

2. Programming the Arduino:

- Write the code to define the sequence and timing of the traffic lights.
- Upload the code to the Arduino Uno using the Arduino IDE.

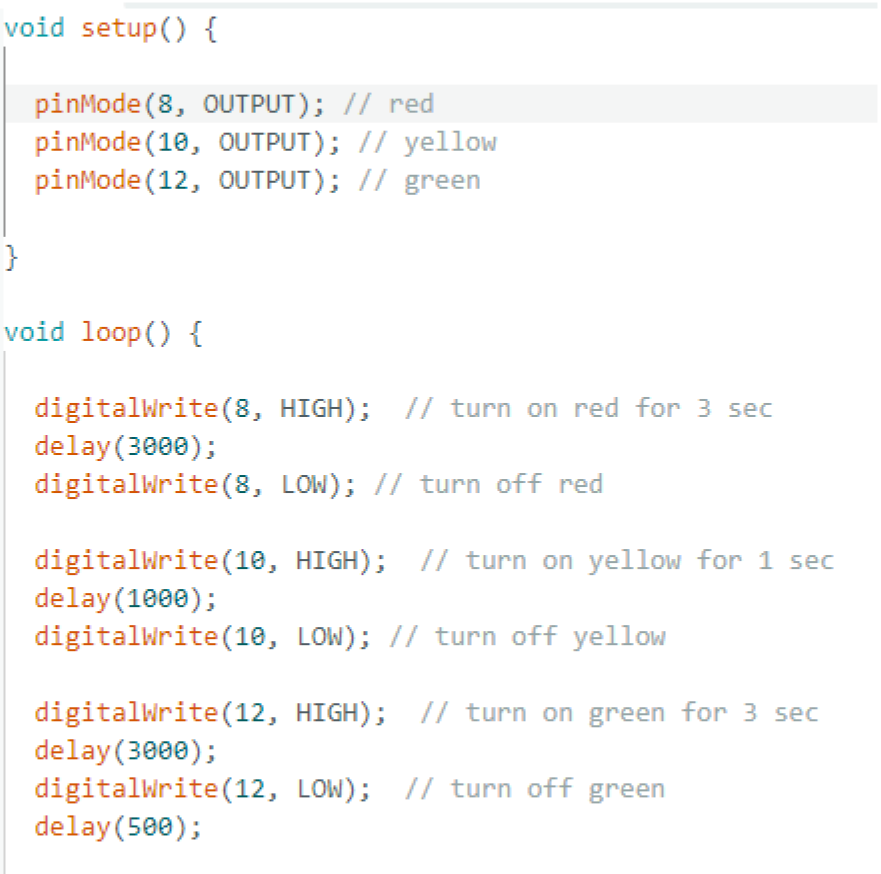
3. Testing and Debugging:

- Power the system and observe the LED sequence.
- Adjust timings and connections as necessary for accurate operation.

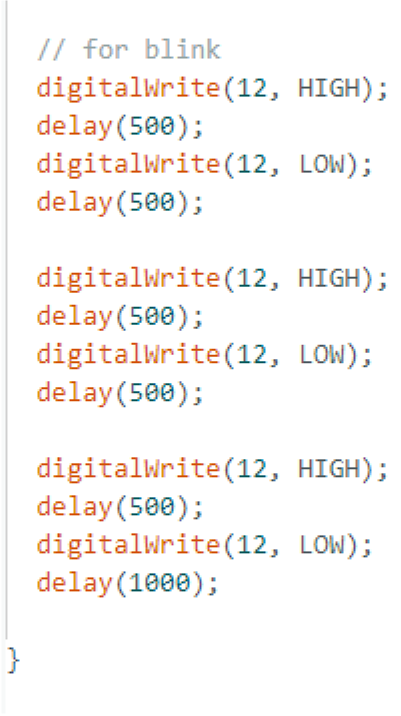
4. Final Deployment:

- Ensure stable operation of the system.
- Demonstrate the functionality as a prototype.

Code



```
void setup() {

  pinMode(8, OUTPUT); // red
  pinMode(10, OUTPUT); // yellow
  pinMode(12, OUTPUT); // green

}

void loop() {

  digitalWrite(8, HIGH);  // turn on red for 3 sec
  delay(3000);
  digitalWrite(8, LOW); // turn off red

  digitalWrite(10, HIGH);  // turn on yellow for 1 sec
  delay(1000);
  digitalWrite(10, LOW); // turn off yellow

  digitalWrite(12, HIGH);  // turn on green for 3 sec
  delay(3000);
  digitalWrite(12, LOW);  // turn off green
  delay(500);
```



```
  // for blink
  digitalWrite(12, HIGH);
  delay(500);
  digitalWrite(12, LOW);
  delay(500);

  digitalWrite(12, HIGH);
  delay(500);
  digitalWrite(12, LOW);
  delay(500);

  digitalWrite(12, HIGH);
  delay(500);
  digitalWrite(12, LOW);
  delay(1000);

}
```

Observations

1. The LEDs illuminated in the correct sequence: Red, Yellow, Green.
2. Timing intervals were consistent with the programmed delays.
3. Minor adjustments were needed to optimize resistor values and LED brightness.

Results

The traffic light control system successfully simulated the operation of a real-world traffic signal. The sequence and timing were accurate, demonstrating the feasibility of using an Arduino-based solution for traffic control applications.

Discussion and Analysis

Scalability: This project can be scaled to include sensors for adaptive traffic control based on vehicle presence.
IoT Integration: With additional modules, such as Wi-Fi or GSM, the system can be connected to a central traffic management network.
Challenges: Ensuring accurate timing and avoiding overcurrent issues required careful selection of resistor values and debugging.
Future Enhancements: Integration of pedestrian crossing signals and real-time monitoring via IoT platforms.

Conclusion

The Traffic Light Control project demonstrated a cost-effective and efficient way to replicate traffic signal operations using Arduino Uno and basic electronic components. It provided valuable insights into the principles of sequential control logic and IoT-based traffic management systems. This project serves as a foundational step for more advanced implementations in smart city technologies.

References

[1] Reints, W., 2017. 100 years of Traffic Light. [Online] Available at : https://www.siemens.com/press/en/feature/2014/infrastructure-cities/2014-08trafficlights100.php [Accessed 28 04 2019].

[2] Thaar, K. & Mays, J., 2018. DESIGN AND IMPLEMENTATION SMART TRAFFIC LIGHT USING GSM AND IR.. Iraqi Journal for Computers & Informatics Ijci, 44(2), pp. 1-5.