



DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Discover. Learn. Empower.

Experiment 3

Student Name: Harsh Kumar

UID: 22BCS15754

Branch: BE - CSE

Section/Group: FL_IOT_603 'B'

Semester: 5th

Date of Performance: 31/07/2024

Subject Name: Internet of Things Architecture and its Protocol Lab

Subject Code: 22CSP-329

1. Aim- To Develop a smart traffic light management system with the help of IoT.

2. Objective:

1. Learn about interfacing.
2. Learn about IoT programming.

3. Equipment Used-

- 1 × Breadboard
- 1 × Arduino Uno R3
- 3 × LEDs (Red, Yellow, Green)
- 3 × 220Ω Resistor
- 4 × Jumper

4. Procedure-

- i. Connect LEDs on the breadboard as Red, Yellow, Green, respectively.
- ii. Connect the negative terminal of the LED and connect the 220 Ohm resistor in series. □
- iii. Connect these negative terminals to the ground. □
- iv. Connect the positive terminal of the LEDs to the pins 10, 13 and 8 respectively.
- v. Power the breadboard by using 5V and GND on the Arduino.
- vi. Open the arduino uno software and create a new sketch and upload the following code in the arduino board.



5. Code:

```
int red = 10;
int yellow = 13;
int green = 8;
void setup()
{
    pinMode(red, OUTPUT);
    pinMode(yellow, OUTPUT);
    pinMode(green, OUTPUT);
}
void loop(){
    changeLights();
    delay(5000);
}

void changeLights()
{
    // green off, yellow on for 3 seconds
    digitalWrite(green, LOW);
    digitalWrite(yellow, HIGH);
    delay(3000);

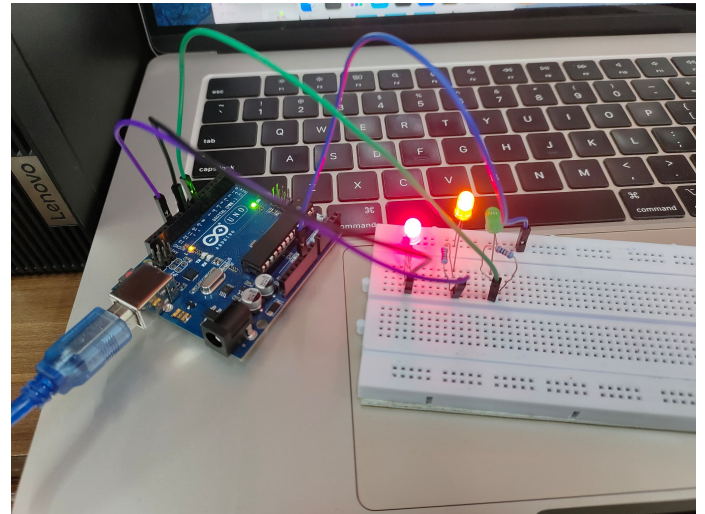
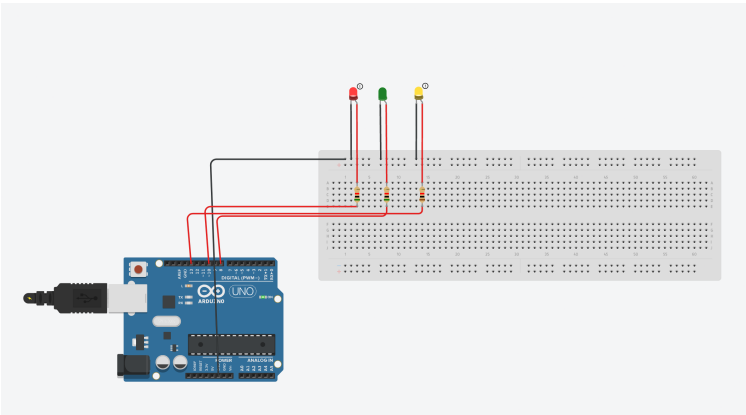
    // turn off yellow, then turn red on for 5 seconds
    digitalWrite(yellow, LOW);
    digitalWrite(red, HIGH);
    delay(3000);

    // red and yellow on for 2 seconds (red is already on though)
    digitalWrite(yellow, HIGH);
    delay(3000);

    // turn off red and yellow, then turn on green
    digitalWrite(yellow, LOW);
    digitalWrite(red, LOW);
    digitalWrite(green, HIGH);
    delay(3000);
}
```

6. Result-

You should see your LED turn on and off. If the required output is not seen, make sure you have assembled the circuit correctly, and verified and uploaded the code to your board. This traffic light controller includes a crosswalk signaling system. The traffic light controller in this system can be implemented practically, and it could be expanded.



7. Conclusion-

In this experiment, we successfully designed and implemented a smart traffic light management system using Arduino. The system utilized arduino and LEDs to manage traffic and pedestrian signals. The Arduino microcontroller effectively controlled the traffic lights based on inputs, demonstrating the potential of using microcontroller-based systems for traffic management.