

Traffic Light Controller

Introduction

The project involves creating a traffic light controller using Arduino R3, a 7-segment display, 3 LEDs, and other electronic components. The traffic light controller will change the colors of the lights in a synchronized manner, namely Red-Yellow-Green and vice versa, at an interval of 9 seconds.

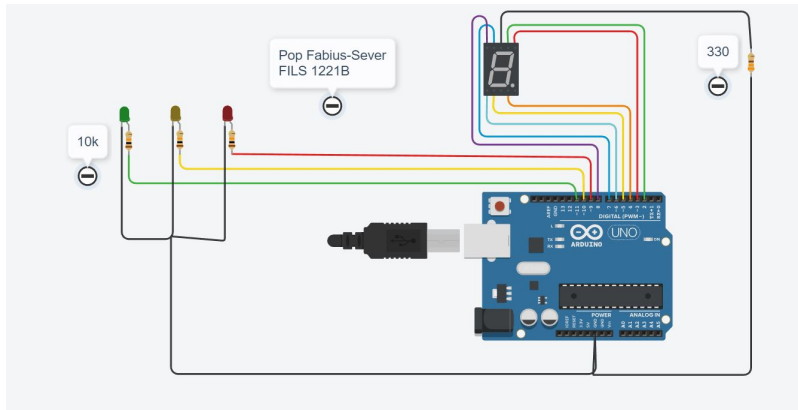
The purpose of the project is to create a useful and practical device that is capable of controlling traffic lights automatically and in a synchronized manner. The idea was to create an efficient and easy-to-use device that can be used to control traffic lights.

This project is useful for road intersections to efficiently regulate traffic.

General description

The project involves an Arduino R3 microcontroller, a breadboard, a 7-segment display, 3 LEDs, a 330 ohm resistor, and three 1k ohm resistors. Using these components, we have created a circuit that controls the LEDs to simulate the synchronized changing of the traffic light colors.

Hardware Design



List of components:

- Arduino R3 - 7-segment display (

<https://components101.com/displays/7-segment-display-pinout-working-datasheet>) - Breadboard - LEDs (3) - 330 ohm resistor (1) - 1k ohm resistors (3) - Jumper wires (Male to Male)

Software Design

Traffic Light Controller with 7 segment display Pop Fabius Sever FILS 1221B

```
#include "SevSeg.h"
```

```
SevSeg S; byte CommonPins[] = {}; byte SegPins[] = {2,3,4,5,6,7,8};
```

```
int red = 9; int yellow = 10; int green=11;
```

```
void setup() {
```

```
S.begin(COMMON_CATHODE, 1, CommonPins, SegPins, 1);
```

```
pinMode(9, OUTPUT);  
pinMode(10, OUTPUT);  
pinMode(11, OUTPUT);
```

```
digitalWrite(red, HIGH);
```

```
}
```

```
void loop() {
```

```
for(int i=9; i>=0; i--)  
{  
  display(i);  
  delay(2000);
```

```
    if(i<=3)
    {
        digitalWrite(yellow, HIGH);
        digitalWrite(red, LOW);
    }
}
```

```
digitalWrite(green, HIGH);
digitalWrite(yellow, LOW);
```

```
for(int i=9; i>=0; i--)
{
    display(i);
    delay(2000);
    if(i<=3)
    {
        digitalWrite(yellow, HIGH);
        digitalWrite(green, LOW);
    }
}
```

```
digitalWrite(red, HIGH);
digitalWrite(yellow, LOW);
digitalWrite(green, LOW);
```

```
}
```

```
void display(int num) {
```

```
    S.setNumber(num);
    S.refreshDisplay();
    delay(20);
```

```
}
```

The code was written in the Arduino IDE development environment and uses the SevSeg library to control the 7-segment display.

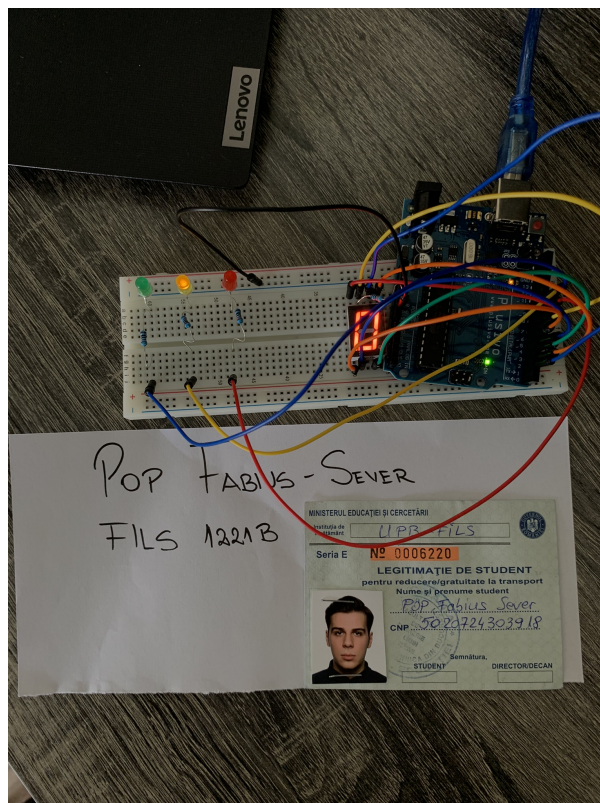
In the setup() function, the pins for the red, yellow, and green LEDs are initialized. Additionally, the 7-segment display is initialized using the SevSeg library.

In the loop() function, the steps for changing the traffic light color are implemented. In the first for loop, starting with red, the numbers from 9 to 0 are displayed on the 7-segment display with a 2-second delay between each number. If the displayed number is less than or equal to 3, the yellow LED is turned on and the red LED is turned off.

In the second for loop, the green LED is turned on and the numbers from 9 to 0 are displayed on the 7-segment display with a 2-second delay between each number. If the displayed number is less than or equal to 3, the yellow LED is turned on and the green LED is turned off.

At the end of the loop() function, all LEDs are turned off and the process starts again with red. The display function is used to display a number on the 7-segment display.

Results



I have successfully created a functional traffic light controller. The signals from the microcontroller were transmitted correctly to the LEDs, and the 7-segment display showed the corresponding number of seconds for the time interval. I have obtained a product that functions according to the specifications and can be used to control a traffic light.

Download

[pop_fabius_1221b_trafficlightcontroller.rar](#)

Bibliografie/Resurse

[bibliografie_tlc.txt](#)

[Export to PDF](#)

From:

<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:

<http://ocw.cs.pub.ro/courses/pm/prj2023/tmiu/trafficlightcontroller>



Last update: **2023/05/29 15:53**