TRAFFIC SIGNAL CONTROL USING IoT

A traffic signal is one of those items that seems to be a part of everyday life for every passenger. Wherever there are a large number of cars, one of these systems is likely to be present. These days, traffic lights are highly sophisticated inside, with a variety of sensors, timers, and traffic monitoring systems that aid in traffic control.

Nowadays, personal vehicles are very common, so the number of vehicles on the road is rising exponentially. Accidents and traffic congestion can result from roads that lack guidance or traffic lights.

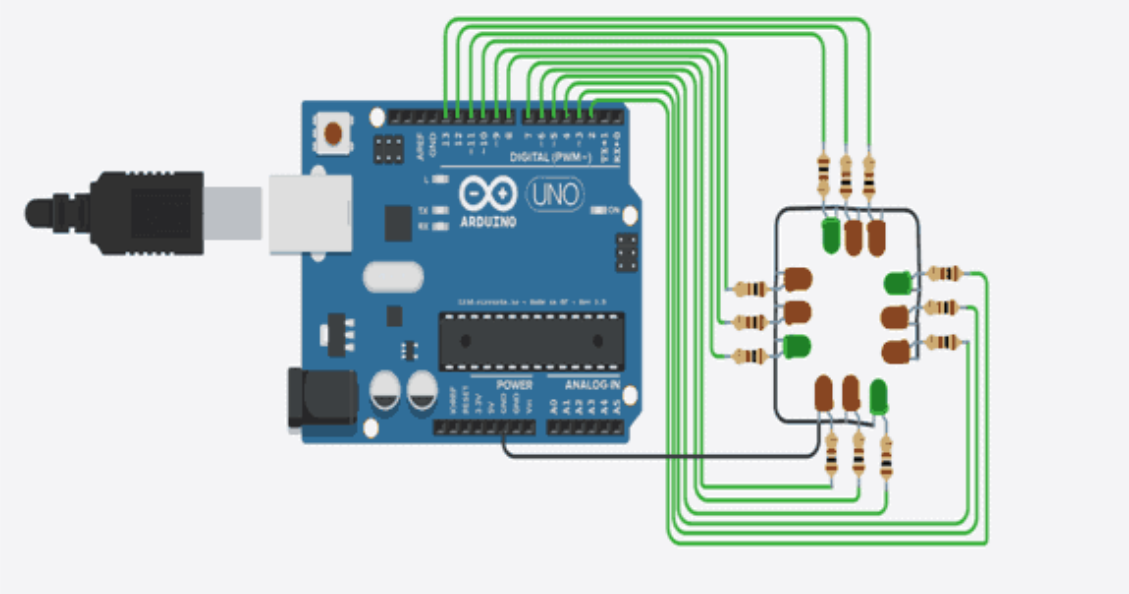
By displaying lights of a standard color on four cross streets, the traffic light system gives drivers and pedestrians directions. Red, yellow, and green are the colors used at traffic lights to indicate stop, slow, and go, respectively. In order to ensure that vehicle traffic moves smoothly and safely, the system is programmed to control the traffic lights. The system is made up of electromechanical controllers and cutting-edge solid-state computerized systems that are simple to set up and keep running. The Traffic Light system in this project is based on an Arduino board. A simple system but can be

extended to a real-time system with programmable timings, pedestrian lighting, etc.

Project requirement:

- 4 Green LED
- 4 Red LED
- 4 Yellow LED
- Connecting wires
- Breadboard
- Arduino broad

Circuit Diagram :



Working:

In this project, a simple traffic light system for a 4 way intersection is implemented using Arduino UNO development board. Although it is not the practical implementation for real life scenarios, it gives a general idea of the process behind the traffic light control system

Simple traffic light controller is made using Arduino UNO, where the traffic is controlled in based on programmed timings.

For better understand let's consider time delays as below

Green – 20 Sec.

Yellow – 5 Sec.

Red – 75 Sec.

Each lane gets time duration of 25 seconds including the yellow light time to move.

The yellow light turns ON for short duration after green light, indicating vehicles to slow down before the red light appears to avoid sudden stoppage.

Vehicles in every lane have to wait for 75 Sec. during red signal.

	Lane 1	Lane 2	Lane 3	Lane 4
Lane 1 Green	20G	25R	50R	75R
	5Y	5R	30R	55R
Lane 2 Green	75R	20G	25R	50R
	55R	5Y	5R	30R
Lane 3 Green	50R	75R	20G	25R
	30R	55R	5Y	5R
Lane 4 Green	25R	50R	75R	20G
	5R	30R	55R	5Y

Code :

```
int Lane1[] = {13,12,11}; // Lane 1 Red, Yellow and Green

int Lane2[] = {10,9,8};     // Lane 2 Red, Yellow and Green
```

```
int Lane3[] = {7,6,5};       // Lane 3 Red, Yellow and Green
int Lane4[] = {4,3,2};       // Lane 4 Red, Yellow and Green
void setup()
{
for (int i = 0; i < 3; i++)
{
pinMode(Lane1[i], OUTPUT);
pinMode(Lane2[i], OUTPUT);
pinMode(Lane3[i], OUTPUT);
pinMode(Lane4[i], OUTPUT);
}
for (int i = 0; i < 3; i++)
{
digitalWrite(Lane1[i], LOW);
digitalWrite(Lane2[i], LOW);
digitalWrite(Lane3[i], LOW);
digitalWrite(Lane4[i], LOW);
}


}


void loop()
```

```
{
digitalWrite(Lane1[2], HIGH);
digitalWrite(Lane3[0], HIGH);
digitalWrite(Lane4[0], HIGH);
digitalWrite(Lane2[0], HIGH);
delay(7000);
digitalWrite(Lane1[2], LOW);
digitalWrite(Lane3[0], LOW);
digitalWrite(Lane1[1], HIGH);
digitalWrite(Lane3[1], HIGH);
delay(3000);
digitalWrite(Lane1[1], LOW);
digitalWrite(Lane3[1], LOW);
digitalWrite(Lane1[0], HIGH);
digitalWrite(Lane3[2], HIGH);
delay(7000);
digitalWrite(Lane3[2], LOW);
digitalWrite(Lane4[0], LOW);
digitalWrite(Lane3[1], HIGH);
digitalWrite(Lane4[1], HIGH);
delay(3000);
digitalWrite(Lane3[1], LOW);
```

```
digitalWrite(Lane4[1], LOW);
digitalWrite(Lane3[0], HIGH);
digitalWrite(Lane4[2], HIGH);
delay(7000);
digitalWrite(Lane4[2], LOW);
digitalWrite(Lane2[0], LOW);
digitalWrite(Lane4[1], HIGH);
digitalWrite(Lane2[1], HIGH);
delay(3000);
digitalWrite(Lane4[1], LOW);
digitalWrite(Lane2[1], LOW);
digitalWrite(Lane4[0], HIGH);
digitalWrite(Lane2[2], HIGH);
delay(7000);
digitalWrite(Lane1[0], LOW);
digitalWrite(Lane2[2], LOW);
digitalWrite(Lane1[1], HIGH);
digitalWrite(Lane2[1], HIGH);
delay(3000);
digitalWrite(Lane2[1], LOW);
digitalWrite(Lane1[1], LOW);
}
```