IOP Conference Series: Earth and Environmental Science

PAPER • OPEN ACCESS

Design and Construction of Density Based Traffic Control System

To cite this article: M. R. Usikalu *et al* 2019 *IOP Conf. Ser.: Earth Environ. Sci.* **331** 012047

Design and Construction of Density Based Traffic Control System

M. R. Usikalu, A. Okere, O. Ayanbisi, T. A. Adagunodo and I. O. Babarimisa
Department of Physics, Covenant University, P.M.B 1023, Ota, Ogun State, Nigeria

Correspondence Email: moji.usikalu@covenantuniversity.edu.ng

Abstract. Congestion is a serious issue due to vechicular traffic. One of the known causes of traffic congestion is the amount of time spend waiting for the red light to change to green. The changing of traffic light is hard-coded and it is not reliant on traffic volume. There is therefore need to simulate and optimize traffic control to better accommodate density based traffic rather than time based. This system attempts to lessen possibilities of traffic jams brought about by traffic lights to a reasonable degree. This project, a density based traffic control system is been implemented to solve this problem. The system entails programming an Arduino using Arduino integrated development environment (IDE) to enable traffic lights give the right of access to the road by selecting the lane with the high number of cars. The traffic lights are modified to chip away at an auspicious premise until there is a signal identified by the infrared sensors. The sensor identifies an object (i.e. a vehicle, a motorcycle etc) and signals the Arduino to control the traffic lights for its individual path. Once there is no sign identified by any of the four sensors the traffic lights keep on dealing with an auspicious premise. The mean response time of the sensor was found to be 0.39 seconds. Further research is recommended to produce the device on a large scale to be deployed to all roads in the country.

Keywords: Traffic, Density, Control, IR sensors.

1. Introduction

One of such over-stretched infrastructure is the road, a circumstance which has resulted to increase in traffic. In spite of the fact that traffic lights have dependably been utilized for controlling the movement of traffic (pedestrian or automobile), traffic management in many real urban areas around the globe has kept on being a subject of concern [1]. A traffic light is a signalling device which controls a traffic flow at road intersections. It consists of three basic lights which include red, yellow and green. Red signal is used to halt traffic from proceeding; yellow signal alerts vehicles for brief stop, while green signal alerts vehicles for procession in the indicated directions [2]. Traffic congestion is a situation that happens on road networks in which vehicles travel slower than usual due to the increased physical use of vehicles (traffic) on the road at that moment. Also known as traffic jam, traffic congestions may result from roads being blocked, bad roads, accidents on the road, lack of proper traffic light system to control vehicles, inappropriate driving by road users etc. This would make the journey longer due to the slow movement of the traffic and increased queuing of vehicles [3]. Urban communities started to make traffic tenets to minimize crashes, while traffic flags and cops were utilized to coordinate option to proceed at major urban convergences. Traffic control development in urban streets has aided easy movement and use of automobile in mega cities [4]. Most of the major roads have effective traffic control system [5], which has facilitated easy flow of vehicles in urban regions. Railroads were invented serving as a temporary solution to the developing issue of street traffic control. However, this creates congestion at terminals inside urban communities.

Therefore, the aim of this study is to design a density based traffic control using Arduino in order to solve the problem of traffic congestion.

2. Methodology and System Design

In the design and construction of this system all components work simultaneously. The system works just as a typical traffic light system. The uniqueness of this is in the event that the thickness of vehicles in a specific path of the road is high. At that point, the sensor in that specific path turns out to be low else it is read as a high signal. The signal from the IR (Infrared) is used by the system to control the traffic jamming of the lane. In the event that we get a low signal from any of these sensors, at that point the green LED shines to that specific way and gives a red to every other way. The Arduino IDE is programmed using C language. The block diagram of the density based traffic control system is shown in Figure.1. It is divided into four different sections with each section representing a lane.

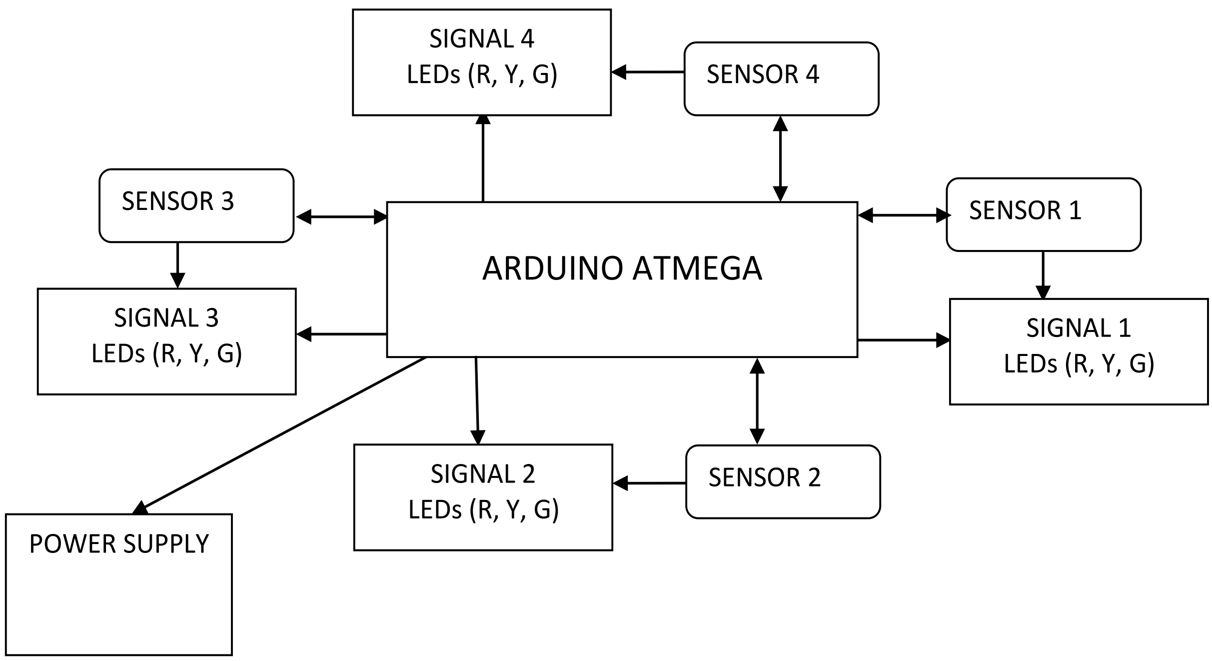

Figure 1: Block diagram of a density based traffic control system.

2.1 Design of the Project

The construction of the system is divided into three stages as stated: power supply stage, sensor switching stage, and Arduino connection stage. The power supply for this system is 9 V DC battery as shown in Figure 2. It has a nominal voltage of 9 V, a discharge resistance of 620 Ω s and a cut off voltage of 5.4 V. According to the power needed for the components of the thickness based traffic light control framework, supply of +5 V regarding GND is created. The total hardware worked with TTL (Transistor-Transistor Logic) rationale dimension of 0 V to 5 V. Power applied to the V_{in} pin is stepped down to 5 V by the on-board regulator on the Mega. The minimum voltage is about 6.2 V because of the regulator dropout (that is, the regulator needs at least 1.2 V above its 5 V output in order to operate) 9 -12 V is recommended. Applying a higher voltage to the system will not provide any more power to the Mega and its peripherals/shields. Instead, the excess power is dissipated in the regulator as heat.

IOP Conf. Series: Earth and Environmental Science **331** (2019) 012047 doi:10.1088/1755-1315/331/1/012047

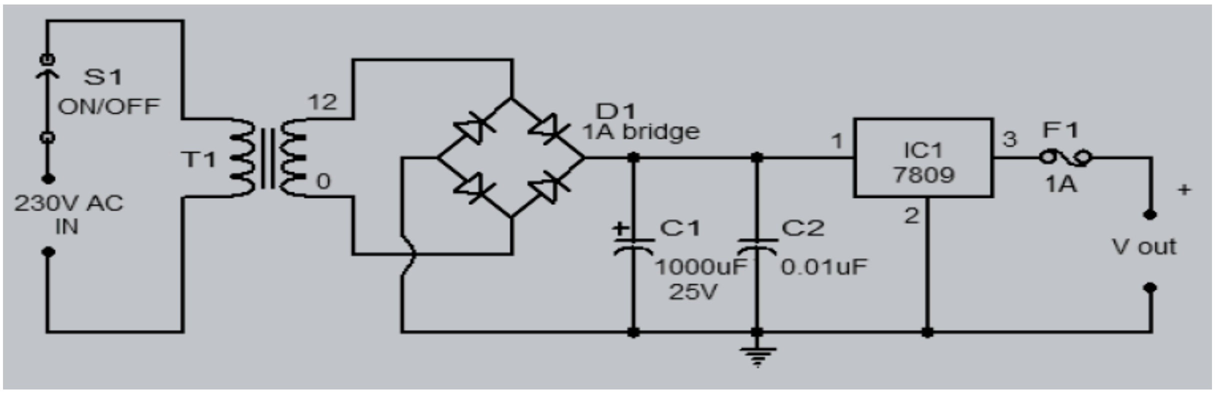

Figure 2: 9V power supply for the Arduino Atmega (retrieved from [11])

The infrared sensors (IR) are the most important components of this project. The sensors act like a switch as it controls the switching of the LEDs. The IR sensors have been applied to several traffic systems [6-9]. Other uses of electronic device for environmental factors have been listed in [10]. The IR system is designed in such a way that its receiver and transmitter are mounted on either side of the road such that it gets activated whenever automobile passes between the two sensors. The sensors enable this system to be automated unless it is a regular traffic control system which has been rendered ineffective in densely populated areas. The infrared sensors have a detection range of 2 cm – 30 cm. They determine if there is a heavy traffic on one lane and allow the flow of traffic in preference to other less dense lanes. Figure 3 shows the connection of the sensor and Arduino.

The sensor detects a minimum of 3.3 V and a maximum of 5 V to the Arduino board. The board then reads the signal in zeros and ones. When the board processes the signal from the sensor as high, it performs the function that it is programmed to do. After the sensor senses signal at one lane all other lanes will be at a stop, but if it is still sensing at one lane and another sensor is triggered it would not change until the traffic at the lane that was called first is done and sends a low signal to the Arduino board for it to switch to the lane of the second command. This process continues until no signal is detected at the sensors anymore and returns to its timely controlled state.

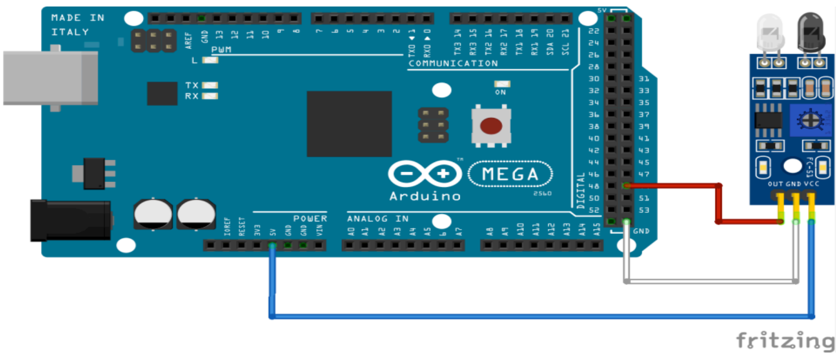

Figure 3 Infrared Sensor interface with Arduino mega

Brief description of the board

Microcontroller	Atmega2560
Operating Voltage	5 V
Input Voltage (recommended)	7 V-12 V
Input Voltage (limit)	6 V-20 V

Digital I/O Pins	54 (of which 15 provide PWM output)
Analog Input Pins	16

The Arduino mega board can be powered via USB connection or with an external power supply. The power supply gives 9 V used to power the board. This power can be by an adapter or battery of required voltage. The board can operate on an external supply of 7 V – 12 V. If there is a voltage supply less than 7 V, the board will become unstable and will not be able to power the components connected to it. Voltage above 12 V may damage the board due to overheating. There are 54 digital input output pins which out of this 16 are used in this project. The LEDs are connected to 12 of these pins i.e. 3 LEDs for each lane and 4 are for the sensors on each of the lanes. The Arduino is programmed using the Arduino IDE.

```
#include<TimerOne.h>

int signal1[] = {23, 25, 27};
int signal2[] = {48, 50, 34};
int signal3[] = {10, 9, 8};
int signal4[] = {6, 5, 4};

bool Sensor1 = true;
bool Sensor2 = false;
bool Sensor3 = false;
bool Sensor4 = false;

bool sen1 = true;
bool sen2 = true;
bool sen3 = true;
bool sen4 = true;

bool found = false;

uint8_t current;


const int triggerpin1 = 27;
const int ProxSensor1 = 29;
const int triggerpin2 = 34;
const int ProxSensor2 = 52;
const int triggerpin3 = 8;
const int ProxSensor3 = 11;
const int triggerpin4 = 4;
```

This part of the code declares the variables to a name that is used throughout the program.The variables declared correspond to the components used in the physical project design. By declaring them as variables, they can be called any time in the program when needed.

```
void setup() {

  Serial.begin(9600);
  for (int i = 0 ; i < 3; ++i) {
    pinMode(signal1[i], OUTPUT);
    pinMode(signal2[i], OUTPUT);
    pinMode(signal3[i], OUTPUT);
    pinMode(signal4[i], OUTPUT);
  }




  // Declaring Proximity sensor pins as output
  pinMode(triggerpin1, OUTPUT);
  pinMode(ProxSensor1, INPUT);
  pinMode(triggerpin2, OUTPUT);
  pinMode(ProxSensor2, INPUT);
  pinMode(triggerpin3, OUTPUT);
  pinMode(ProxSensor3, INPUT);
  pinMode(triggerpin4, OUTPUT);
  pinMode(ProxSensor4, INPUT);

```

This part is the initial process of the circuit, this states that the LEDs are declared as output and the sensors are declared as the input to control the LEDs. The “pinMode” identifies the connections on the digital input output parts of the Arduino

```
if (Sensor1) {

  current = 1;

  digitalWrite(signal2[0] , HIGH);
  digitalWrite(signal3[0] , HIGH);
  digitalWrite(signal4[0] , HIGH);

  while (countt != 4) {

    digitalWrite(signal1[a] , HIGH);
    digitalWrite(signal2[b] , HIGH);
    if ( (millis() - tim) >= 3000) {
      digitalWrite(signal1[a] , LOW);
      digitalWrite(signal2[b] , LOW);
      tim = millis();
      --a;
      ++b;
      ++countt;

    }
    Sensor1 = false;
    Sensor2 = true;
    rst(countt , a , b , tim);
    if (found) {
      found = false;
      break;
    }
```

This part of the code controls the sensor. Once the sensor has detected an obstacle within its observation radius it will send a signal to Arduino to command the traffic to flow for the lane to which it corresponds. Not until the obstacle has left the radius on the sensor before it continues sequentially on a timely basis. However, the sensors at any other lane are being triggered to control traffic on the other lanes. The code can be briefly explained using the flow chart in Figure 4.

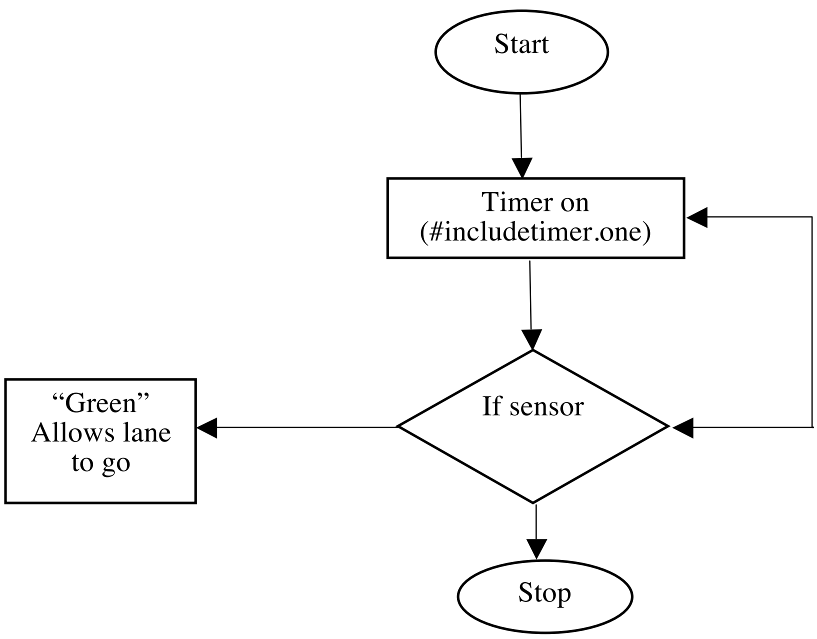

Figure 4: Flowchart of coding process

3. Results and Discussion

The construction of the project was done firstly on the breadboard before being transferred to the vero-board. The LEDs which are red (5 mm), yellow (5 mm) and green (5 mm) are connected in series with resistors of 220 Ω each connected to the negative legs of the LEDs. Figure 5 shows the connections of the LEDs and resistors. The connections of the LEDs and the resistors are created to represent traffic lights for each lane i.e. there will be four of these for each of the lanes. The Arduino is placed at the middle as the controlling system that will send the information for this operation to be effective. Figure 6 shows the arrangement of the LEDs on the breadboard for each lane. The legs of each of the components are then wired to the digital input and output pins of the Arduino board. The LEDs at the lanes will be connected to the pins on the board between pins 3 and 53. The legs of the IR sensors are 3 in number. The V_{cc} leg is connected to the 5 V pin on the arduino board, the GND leg is connected to the GND pin on the board and the legs of the resistors simultaneously. The OUT leg is connected to one of the digital input and output pins corresponding to the traffic light it is to control. Figure 7 shows the connections of the components to the arduino board. The power is connected using an Arduino power cord and a 9 V battery. Although the alternative power supply used is the USB cable to be able to send the codes and power to the board simultaneously. The Arduino is then programmed to enable the traffic flow on a timely basis. Figure 8 shows a lane going and the rest are stopped. So also when the sensor at lane 2 is being signalled, this will turn the traffic lights to green i.e. a GO on that lane and making the other lanes to stop for lane 2 to have the right of way. The sensor at lane 3 also detecting an obstacle turns the traffic light on that lane to green for go and stops all other lanes. When two sensors are detected at the same time as shown in Figure 9 the sensor that detected the signal first will be given the right of way and it will switch to the next sensor once the first one stops detecting the signal.

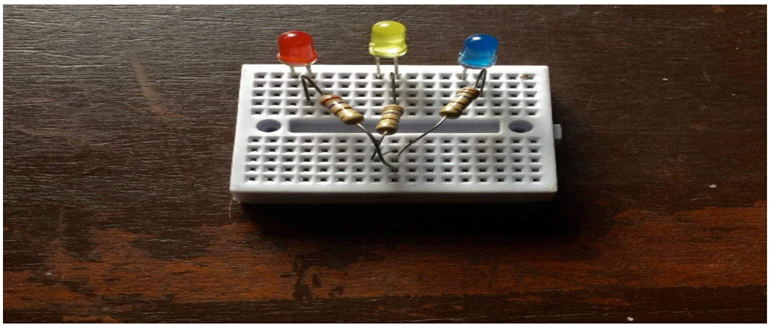

Figure 5: LEDs and resistor connection on the breadboard

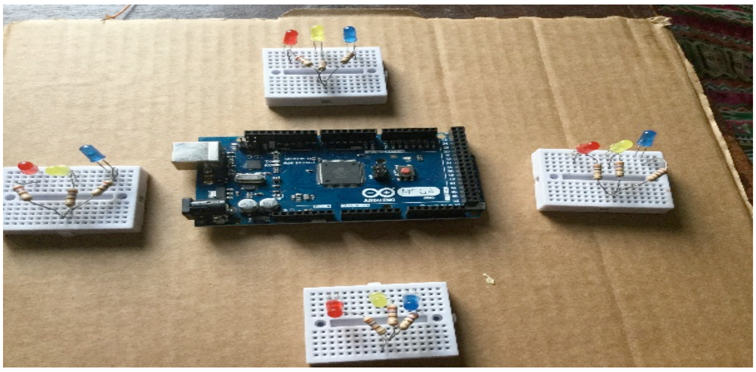

Figure 6: Arrangement of LEDs for each lane

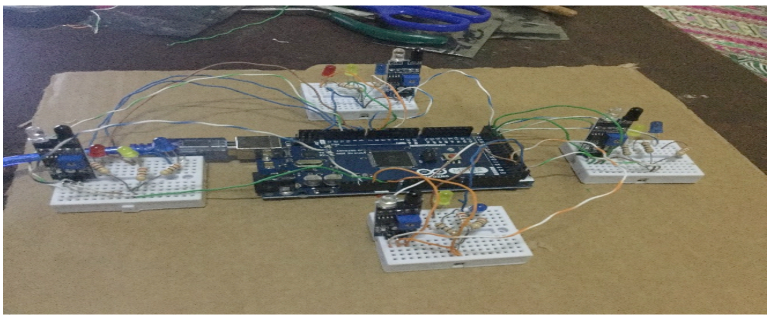

Figure 7: Connection of components to the Arduino board

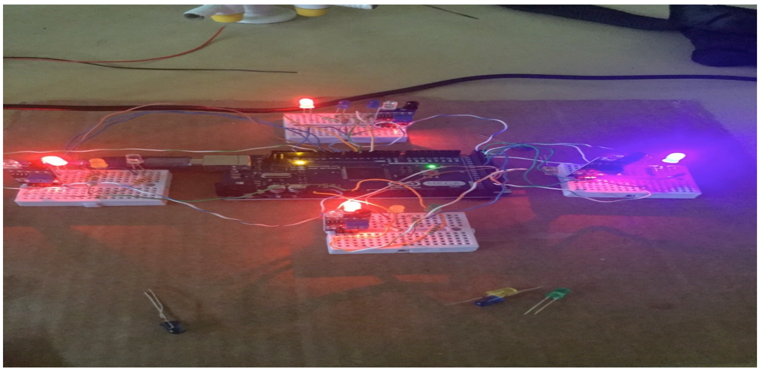

Figure 8: Traffic flow at signal 1

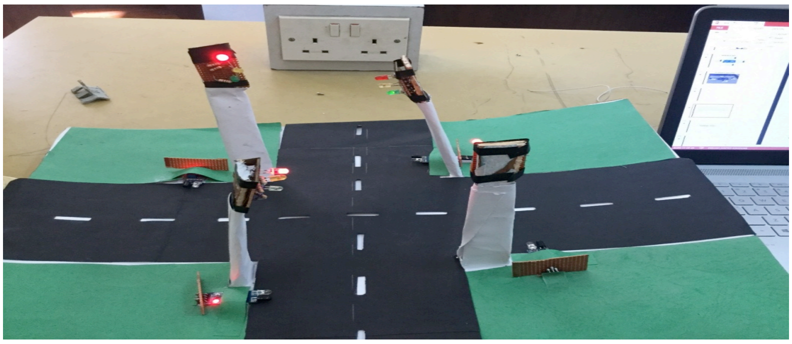

Figure 9: Signalling sensors and response

The response time of the sensor is the time taken for the sensor to produce an output. This is the time it takes once it detects an obstacle and sends a signal to the Arduino and changes the traffic light colour for execution. The result obtained here is in agreement with the report of [9]. Table 1 shows the average response time and Figure 10 is the pictorial chart of the time.

Table 1: Response time of the device

Number of Attempts	Time (s)
1	0.32
2	0.41
3	0.44
4	0.31
5	0.31
6	0.32
7	0.54
8	0.54
9	0.29
10	0.39
Mean	0.387

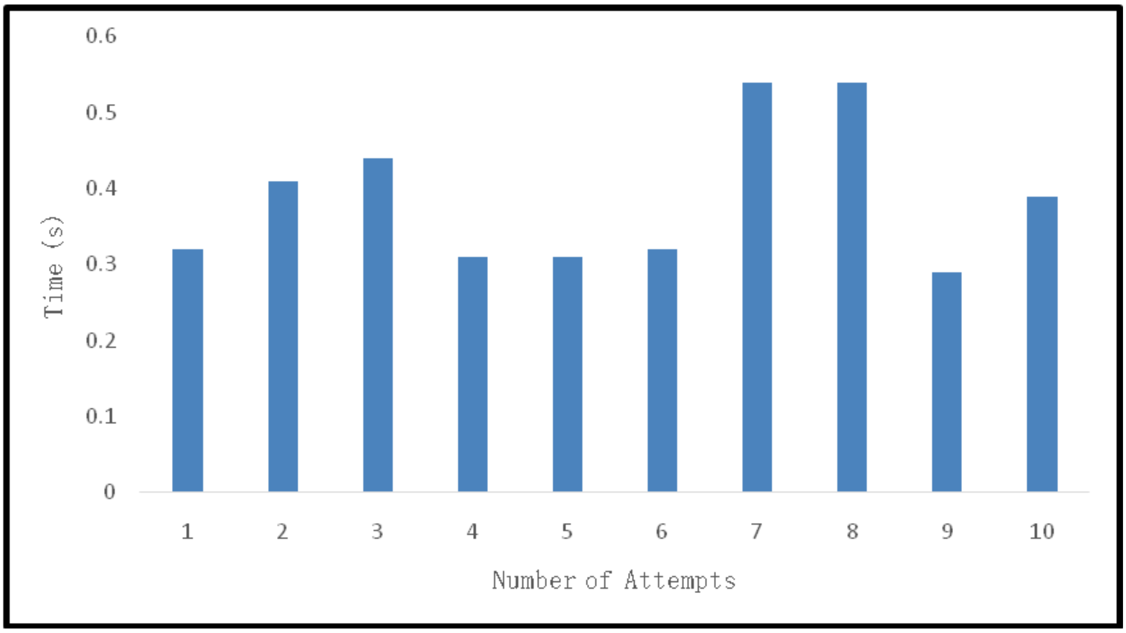

Figure 10: Response time of the device

4. Conclusion

The density based traffic control system has been designed, constructed and tested to ensure validation of its function and operations. In this research, we have succeeded in minimizing the traffic congestions created by the fixed time based traffic light system. The system is effective and the cost of production is very low. Future work is recommended in order to produce the device on a large scale and deploy to all roads in order to reduce traffic congestion in places like Lagos where traffic congestion has become a big issue.

Acknowledgments

Authors acknowledged the publication support received from Covenant University.

References

[1] Udoakah1 Y. N. and Okure I. G. (2017) Design and implementation of a density-based traffic light control with surveillance system, *Nigerian Journal of Technology (NIJOTECH)*, 36(4): 1239 – 1248
[2] Kham N and New (2014) Implementation of modern traffic light control system", *International Journal of Scientific and Research Publications*, 4(6): 1-6
[3] Traffic Control Systems Handbook (2005). Prepared for federal highway administration by Dunn Engineering Associates in association with Siemens Intelligent Transportation Systems
[4] Kell J. H. and Fullerton I. J (1998) Manual of Traffic Signal Design Institute of Transportation Engineers, Prentice-Hall, Inc., 138pp
[5] Sinhmar P (2012), "Intelligent traffic light and density control using IR sensors and microcontroller", *International Journal of Advanced Technology and Engineering Research (IJATER)*, 2(2): 30-35.
[6] Nwoye C. D., Usikalu M. R., Babarimisa I. O, Achuka J. A and Ayara W. A. (2017) Construction of An Automatic Power Switch using Infrared Motion Sensor, *Journal of Informatics and Mathematical Sciences,* 9(2): 331–337
[7] Ayara W. A, Omotosho T. V, Usikalu M. R, Singh M. S and Suparta W. (2017) Development of a solar charged laboratory bench power supply, *Journal of Physics: Conference Series*, 852(1): 012044
[8] Usikalu M. R, Shittu A. H and Obafemi L. N (2018) Construction of an intelligent and efficient light control system, *International Journal of Mechanical and Production Engineering Research and Development (IJMPERD)*, 8(4): 1057-1066

IOP Conf. Series: Earth and Environmental Science **331** (2019) 012047 doi:10.1088/1755-1315/331/1/012047

[9] A. Dakhole and M. Moon (2013) Design of intelligent traffic control system based on ARM, *International Journal of Advance Research in Computer Science and Management Studies*, 1(6): 76-80.
[10] Adagunodo T.A., Ajigbotosho J.J., Obafemi L.N., Usikalu M.R., Akinwumi S.A., Ayara W.A. (2018) Construction of an in-situ Smart Device that Measures some Basic Environmental Factors for Agricultural Monitoring. IOP Conference Series: Earth and Environmental Science, 173: 012023. https://doi.org/10.1088/1755-1315/173/1/012023.
[11] Retrieved from www.circuitdiagram.org