Design and Implementation of a Traffic Control System Based on Congestion

Ahmed J. Abougarair*, Nasar Aldian A. Shashoa, Widd E. Ben Guma

Electrical and Electronics Engineering University of Tripoli, Libya

Abstract: The traffic issues have garnered more and more attention on a global scale as the number of cars has grown. One of the biggest problems is the traffic congestion, also the fixed-time settings are still used by the majority of traffic systems today. These technologies are unable to dynamically alter the timing of traffic lights in response to heavy traffic. Thanks to technological advancements, sensors or cameras can now collect data on traffic volume and wait times. This study provided an illustration of a traffic light control system that can manage traffic according to the number of vehicles in each road. Additionally, it showed how the system was designed using the Proteus design suite software and how a prototype of the system was implemented using an Arduino Mega 2560 and an infrared sensor. Through the results obtained, the efficiency of the proposed system is clear by comparing it with the system that depends on the fixed time of traffic signals.

Keywords: Congestion, Traffic light, Vehicle sensing, Radar sensors, Arduino.

1. INTRODUCTION

Since traffic light system timers only have a certain amount of time to move traffic between multiple routes during the day, the current traffic control system is ineffective. Some vehicles may have to wait for a long time period as a result of this. A situation like this results in high traffic on one side of the road and no traffic on the other. To overcome this problem, it is required to design a traffic light system in which has a dynamic time for traffic lights based on traffic volume on each route. The purpose of this study is to improve traffic light systems' efficiency by tracking and dynamically adjusting traffic light waiting times in response to traffic volume. Moreover, Reduce the energy consumption when the road intersection has no vehicle traveling on any of the roads [1]. Signal timing is updated and adjusted using a control algorithm by monitoring traffic density and dynamically altering the waiting time at traffic signals. These unproductive traffic bottlenecks are anticipated to be avoided or reduced by the suggested solution.

The broad issue of spotting traffic signs has been the subject of extensive investigation. Adaptive cruise control is a proven technology that is now being manufactured. As part of efforts to develop intelligent highways, numerous concepts for intelligent transportation systems (ITS) were thoroughly researched in the 1990s in the United States, Japan, and Europe [2]. Most research on the subject of signal-timing optimization (also known as signal synchronization) has been done at intersections [3-5]. Dresner *et al*. [6, 7] have recently suggested replacing traffic lights and stop signs with intelligent lights for futuristic autonomous vehicles. Using a two-way communication protocol, the autonomous vehicles call the intersection in front of them to reserve a time-space slot to pass, which, among other things, can help improve the fuel economy. There are also a ton of similar articles that employ fuzzy logic and rules to intelligently control traffic signals in the literature. Generic algorithms, neural networks with learning, certain hybrid systems combining fuzzy logic and a generic algorithm, and running an isolated intersection traffic signal are also used [8-10]. The following is a breakdown of the paper's structure: The vehicle sensing explained in Section two. Section three presents the density-based traffic light control system. Component Description is demonstration in section four. System design is demonstrated in section five. Result and diction are presented in section six is and the conclusion will be presented in seven.

2. VEHICLE SENSING

Networked systems called sensing systems are based on both infrastructure and vehicles. Infrastructure sensors, such as in-road reflector devices that are installed, embedded, or located at road sides, are indestructible. For identifying communications, vehicle-sensing systems deploy electronic beacons from the infrastructure to the vehicle and from the vehicle to the infrastructure. When determining the best technology for a given application's size of target, sensing range, mounting of the sensor, and whether it is primarily indoor or

*Address correspondence to this author at the Electrical and Electronics Engineering, University of Tripoli, Tripoli, Libya; Mobil: +218925385942;
E-mail: a.abougarair@uot.edu.ly

E-ISSN: 2409-9694/22

outdoor, the ability to reliably detect vehicles offers significant benefits for asset management, resource allocation, site safety, and traffic control [11].

Some technologies used for vehicle detection with a brief overview:

- ***Magnetometer:*** A magnetometer detects large ferrous objects (such as a truck, car, or rail car) utilizing passive sensing technology by observing changes in the ambient magnetic field. The sensor notices changes when a vehicle modifies that magnetic field. The magnetometer's range will vary depending on the target, much like with other sensors. Numerous indoor and outdoor applications are possible for it. A drive-through application is one in which a magnetometer can accurately identify automobiles at drive-through systems or other drive-up kiosks. Personnel may be alerted to the presence of a vehicle upon detection of one [12].

- ***Optical Sensor (Infrared):*** Opposed-mode optical sensors (infrared) use light beam disruption between an emitter and receiver to detect things. For instance, a car passing between the transmitter and receiver can cause the light beam to be broken. However, due to their vulnerability to a variety of conditions, optical sensors are not as frequently utilized for vehicle detection in outdoor applications. They may, nevertheless, be a smart choice in some situations [12].

- Radar Sensors: Since they are unaffected by weather factors including wind, rain, fog, light, humidity, and air temperature, radar sensors use frequency-modulated continuous-wave (FMCW) radar to accurately identify moving or stationary targets, such as vehicles, trains, trucks, and cargo, outside in severe weather. The selection of a technique depends on the application to be used and the necessary sensor characteristics. The following features were taken into account: accuracy, range, calibration, resolution, and affordability. There are numerous other approaches that cannot be covered in full here. Despite the fact that infrared (IR) sensors are typically disturbed by background noise like radiation, ambient light, etc., they were chosen for this design because they are affordable, easily accessible, and quickly interfaceable.

3. DENSITY-BASED TRAFFIC LIGHT CONTROL SYSTEM

The system can be divided into a group of units; each unit has a specific function as follows:

The Display unit: It is the basic traffic signal display which the vehicle driver or the commuter can see. Specifically, three light emitting diodes, ‘RED’, ‘YELLOW’ and ‘Green’, each having their usual meaning of ‘STOP’, ‘READY’ and ‘GO’ respectively as shown in Figure **1**. They are controlled by the control buses of the microcontroller depending on the logical decisions taken by the controller to control the roads of traffic according to their densities.

Figure 1: Traffic light signal display.

The Detector unit: It is the component that recognizes the presence of cars and transmits this data to the controller for processing by IR sensors. On the sidewalk, these sensors are mounted.

The controller unit: a control system It could be a simple microcontroller or a computer with a microprocessor [13]. The component that receives detector (sensor) output provides an indicator of the presence of cars, calculates traffic density as a result, and controls the display unit as necessary.

4. COMPONENT DESCRIPTION

The proposed traffic light system (TLS) consists of:

- The controller (Arduino MEGA 2560) [14,15].
- IR sensor.
- Traffic Light Module.
- Arduino software.
- Connection wires.
- Bread board.
- Printed foam board.

Arduino MEGA 2560: Arduino Mega 2560 as shown in Figure 2 is an Arduino microcontroller board based on the ATmega2560. It has 54 digital input/output pins (of which 15 can be used as PWM outputs), 16 analog inputs, 4 UARTs (hardware serial ports), a 16 MHz crystal oscillator, a USB connection, a power jack, an ICSP header, and a reset button [16].

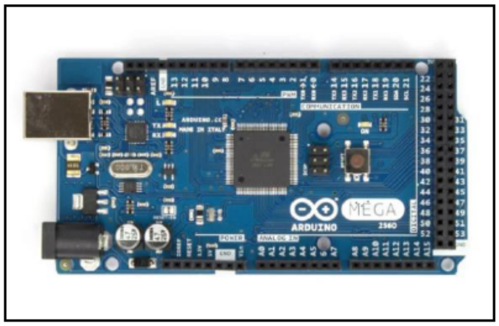

Figure 2: Arduino MEGA 2560.

IR sensor: An electronic gadget known as an IR sensor as shown if Figure 3 emits light in order to detect nearby objects. An IR sensor can monitor an object's heat while also spotting movement. Typically, all items emit some kind of thermal radiation in the infrared range. Although these kinds of radiations are invisible to the human eye, infrared sensors may pick them up [17]. The sensor can detect objects up to 30 cm away. The potentiometer on the sensor may be changed to alter the detecting distance [18]. Infrared transmitters come in a variety of designs based on the wavelengths, output power, and response time they have. A photocoupler, also known as an optocoupler, or photo LED and photo diode, are the two components that make up an infrared sensor [19].

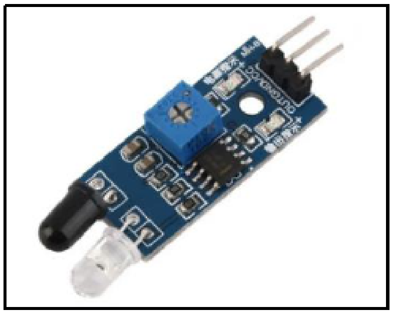

Figure 3: IR sensor.

- ***IR Transmitter or IR LED (White part):*** An infrared transmitter, often known as an IR LED, is a light-emitting diode (LED) that emits infrared radiation.

- ***IR Receiver or Photodiode (Black part):*** Infrared radiation from an IR transmitter is picked up by infrared receivers or infrared sensors. Photodiodes and phototransistors are used as IR receivers. As opposed to regular photo diodes, infrared photodiodes only pick up on infrared radiation [20]. As seen in Figure **4**, some of the radiation emitted by the IR transmitter by the time it reaches the item bounces back to the IR receiver. The output of the sensor specifies [17] based on the strength of the IR receiver's reception.

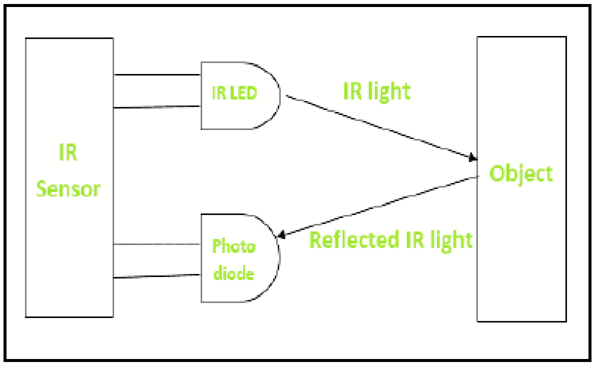

Figure 4: Working principle of IR sensor.

Traffic Light Module: Traffic Light Module is a light-emitting diode module for sending a digital/analog signal output. It includes a large 8mm Red, Yellow and Green LEDs arranged like a typical traffic light that are useful for experimenting with traffic signals. There modules have current limiting resistors built-in, so they can be connected directly to the pins of a microcontroller unit [19].

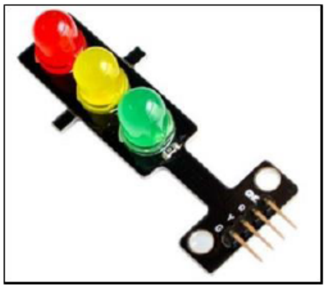

Figure 5: Traffic Light Module.

Arduino software (IDE): It is simple to write code and upload it to the board using the free and open-source Arduino Software (IDE). Any programming language may be used to create the environment, and any Arduino board may be used with the software [21]. The program was created in the C++ language for this paper and presented in the appendix.

5. SYSTEM DESIGN

5.1. Road layout and system design using Proteus software

This traffic light system designed to control the traffic light signals in a 4-side junction and a roundabout in the center with traffic flow in each side in both ways as shown in Figure **6**. Each lane has two appropriate sensors in order to vary the density of the traffic. The system is designed in Proteus software in order to represent the connections for each road with the controller. The number of IR sensors can be increased for better traffic volume predictions, which in turn leads to system improvement [22, 23].

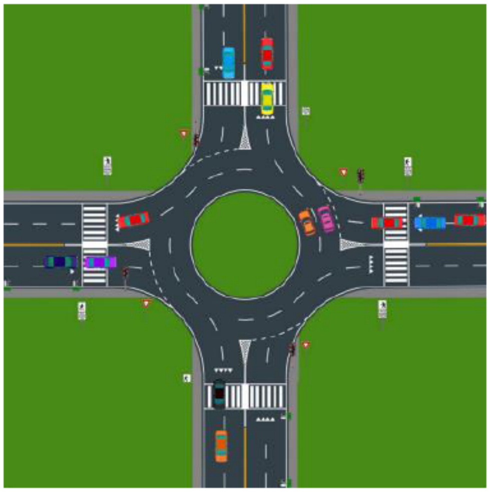

Figure 6: Road layout of the Project.

The four roads from Figure 7 are labeled from A to D in order to manage the priority of them which is used by the controller to make a decision of which route must opened when their traffic density is equal.

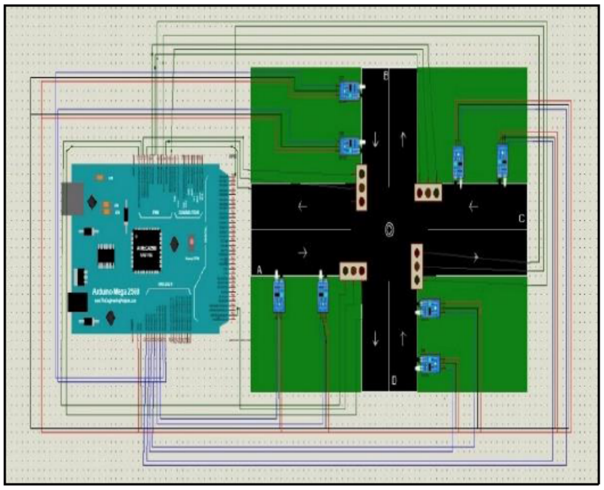

Figure 7: System circuit design using Proteus software.

5.2. Simple prototype of Density-based Traffic Light control system using IR Sensors

A prototype of traffic signal control system can be made using IR sensors along with Microcontroller and LEDs as shown in Figure **7** above, which can prove a worth for the real time application of controlling traffic signals based on the density of traffic. The system consists of the following three main components [24-28]:

Display Unit: It consists of 3 LEDs- (Green, Red and Yellow) as shown in Figure **8** in each side of the junction, with an all total of 12 LEDs.

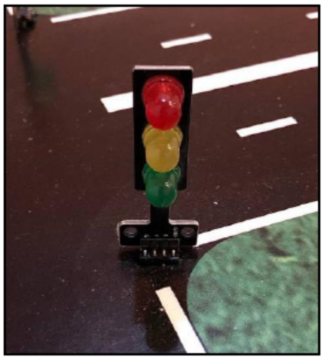

Figure 8: Display unit.

Detector Unit: It consists of an arrangement of IR LEDs at each junction which detects presence of vehicles as displayed in Figure **9**.

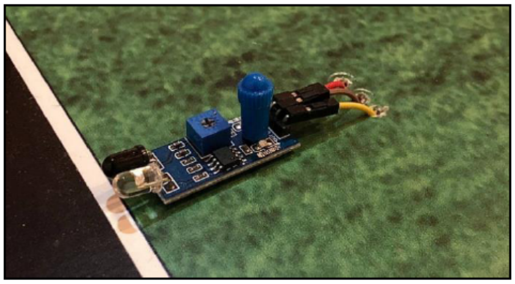

Figure 9: Detector unit.

Controller Unit: It consists of an Arduino Mega 2560 controller as shown in Figure **10** which receives

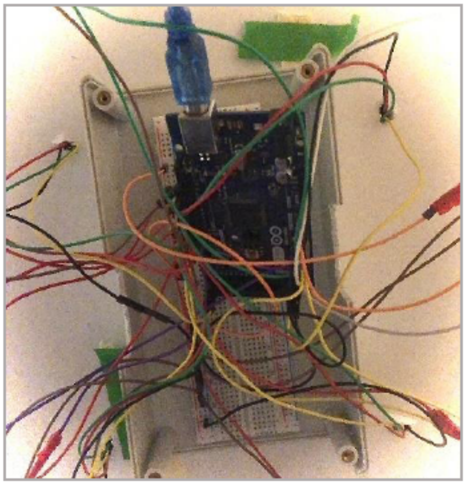

Figure 10: Controller unit.

the IR sensor output and accordingly controls the glowing of LEDs.

5.3. Working Principle of the System

For simplification each road in this model has two IR sensors which used to detect the vehicles; but in larger models, the distance between the sensors can be increased so the sensors used to estimate the number of vehicles (traffic density) and not count them. When infrared LED emits Infrared signals at certain frequency and a vehicle appears on the line of infrared light, it is reflected back by the vehicle which is sensed by the receiver. And so, the sensor detects this vehicle, then the LED indicator lights up, giving a low-level output signal in the OUT pin. Figure **10** shows the previous discussion. The microcontroller makes a decision on which route has to be opened based on the output signals from all infrared sensors (IR sensors) in the system. The route selection is taken as:

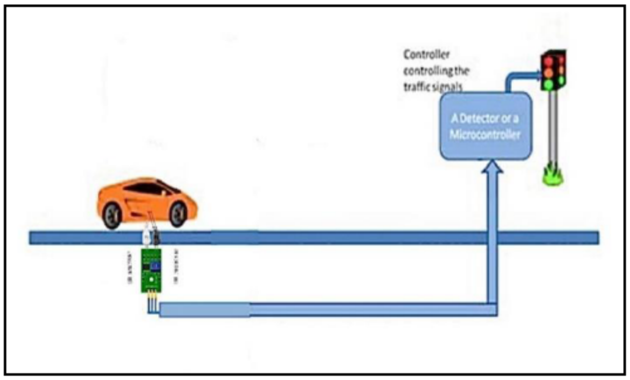
Controller controlling the traffic signals
A Detector or a Microcontroller

Figure 11: Vehicle detection using IR sensor.

1- In each route, output signals from two IR sensors are sent to the microcontroller.

2- The microcontroller then selects the route that has the higher traffic volume to be opened.

3- When two roads or more have the same traffic conditions then the microcontroller decision depends on the priority of the roads. In other words, the road that has the higher priority must open before. Route 'B' has the highest priority, then route 'D' comes in the second place, next route 'A' and finally route 'C' has the least priority.

4- When all the roads are empty the traffic lights are all OFF in order to reduce the consumption of the power. The order of the labeled roads is referenced in Figure **12**.

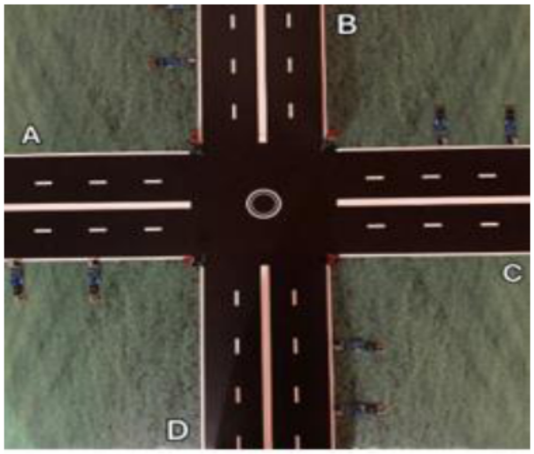

Figure 12: Prototype of Density- based traffic light control.

6. RESULTS AND DISCUSSION

The system was tested with different traffic conditions to approve the validity of the design. This chapter reveals some of traffic condition cases and the response of the system in each case.

1. Different traffic density in roads:

With different traffic density in the roads, the traffic light system must open the route that has the more traffic volume. Figure **14** show the response of the system if route 'A' has the highest traffic density.

2. Equal traffic density in roads:

If the traffic density in some roads is equal, then the traffic light system decision is made according to the priority of the routes. Figure **15** present the response of the system with equal traffic volume in roads. Road 'D' is opened in this figure since it has a higher traffic volume over route 'B' and it has the highest priority over routes 'A' and 'C'. In the same way, Road 'B' is opened as it has the highest priority over all other routes.

3. Roads with no traffic:

If all the roads are empty, the display units (traffic lights) are all OFF as shown in Figure **16** in order to minimize the power usage.

Figure **13** show the flowchart of Density-based Traffic Light system Suppose density for each road has 3 levels 2,1 and 0. Road 'X' has level 2 density when both IR sensors x1 and x2 are 1,1. In the same way, it has level 1 density when x1 and x2 are 1,0 respectively. Lastly, the level is 0 when x1 and x2 are 0,0.

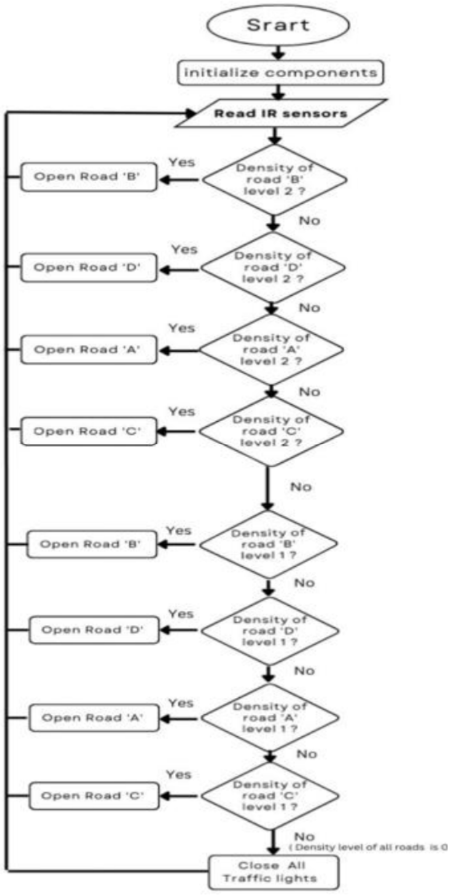
Srart
initialize components
Read IR sensors
Density of road 'B' level 2 ?
Yes
Open Road 'B'
No
Density of road 'D' level 2 ?
Yes
Open Road 'D'
No
Density of road 'A' level 2 ?
Yes
Open Road 'A'
No
Density of road 'C' level 2 ?
Yes
Open Road 'C'
No
Density of road 'B' level 1 ?
Yes
Open Road 'B'
No
Density of road 'D' level 1 ?
Yes
Open Road 'D'
No
Density of road 'A' level 1 ?
Yes
Open Road 'A'
No
Density of road 'C' level 1 ?
Yes
Open Road 'C'
No
(Density level of all roads is 0
Close All Traffic lights

Figure 13: Flowchart of density-based traffic light system mechanism.

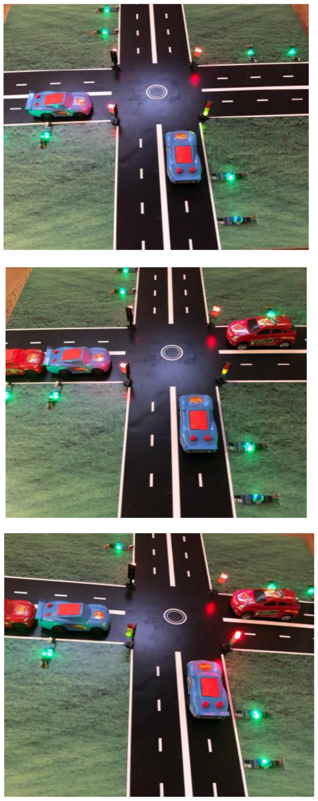

Figure 14: Density-based Traffic Light system response due to different traffic density in roads.

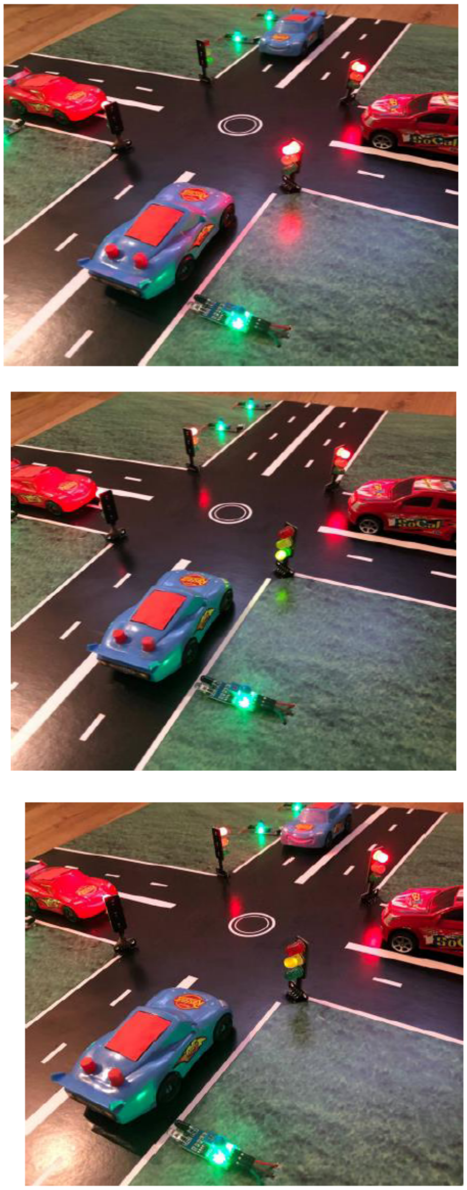

Figure 15: Density-based Traffic Light system response due to equal traffic density in roads.

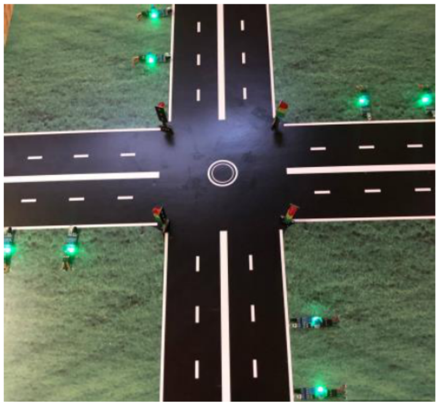

Figure 16: Density-based Traffic Light system response with no traffic in roads.

7. COMPARISON BETWEEN FIXED AND DENSITY-BASED TRAFFIC LIGHT SYSTEM (TLS)

The delay time for the following case studies was measured for fixed (old) TLS in different routes opening order and dynamic (proposed) TLS that has a dynamic routes opening order depends on the density of the traffic and/or the priority of the roads. The obtained results are shown in the following tables. (For fixed TLS each road opened for 20sec)

Case Study 1

The first case study is the red car in the left (located at road ‘A’) in second image of Figure **14**. Table **1** indicates the delay time (waiting time) that this car takes before it can GO. (Measurements started when Green led is ON at route ‘D’).

Case Study 2

The second case study is the red car in the right (located at road ‘C’) represented in third image of Figure **15**. Table **2** obtains the delay time that this car takes before it can GO. (Measurements started when the Green led is ON at route ‘B’).

Figure 16 shows a chart that compares between the results indicated in Table **1** and Table **2** which proves the improvement in traffic light system performance when using the proposed Density- based TLS. The improvements can be summarized in waiting time minimization so the vehicles haven’t need to wait for a long time period and avoid high traffic on one side of the road and no traffic on the other.

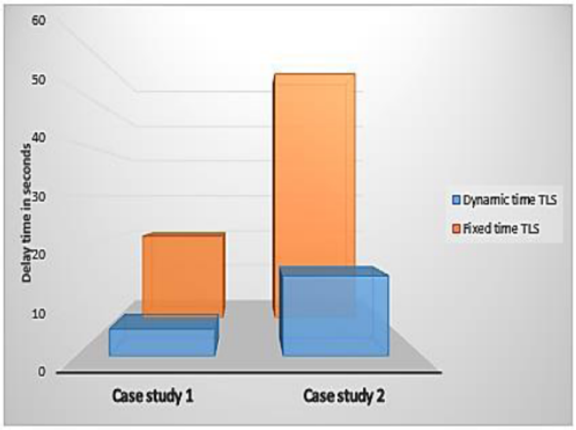

Figure 16: Comparison between delay time in fixed time TLS and dynamic TLS for both case studies.

8. CONCLUSION

In this work, a density-based traffic light control system for traffic control at road intersections was designed to decrease needless time wastage and reduce energy consumption when the road is empty—goals that the current conventional traffic light control system has failed to fulfill. Using infrared sensors and a controller chip, it has demonstrated a method of managing traffic at road intersections. In particular, it shows how a functional software program may control traffic based on how much traffic is present on each route at the intersection. Additionally, it offers an

Table 1: Comparison between Waiting Time Traditional and Dynamic TLS for the First Case Study

Road Pattern	$D \rightarrow A$ **Best Case**	$D \rightarrow (B or C) \rightarrow A$	$D \rightarrow ,B or C. \rightarrow ,C or B. \rightarrow A$ ***Worst case***
Delay time in seconds (Fixed time system)	20	40	60
Delay time in seconds (Dynamic time system)	5	-	--

Table 2: Comparison between Waiting Time Traditional and Dynamic TLS for the Second Case Study

Road Pattern	$B \rightarrow C$ **Best Case**	$B \rightarrow (A or D) \rightarrow C$	$B \rightarrow ,A or D. \rightarrow ,D or A. \rightarrow C$ **Worst case**
Delay time in seconds (Fixed time system)	20	40	60
Delay time in seconds (Dynamic time system)	--	-	15

alternative to the traditional traffic signal, which regulates traffic flow according to the density of each lane regardless of the number of vehicles using it. Additionally, the system. The sensor utilized in the design (IR sensor), which is regularly accessible in the market, is inexpensive, easy to interface with, and typically disturbed by noise in the surroundings such as radiations, ambient light, etc.

The design of this paper can be changed to regulate more than four lanes of traffic, which would be an improvement. Additionally, a different kind of sensing can be used in place of the existing sensing methodology. For instance, since a large percentage of automobiles today include one or more cellphones that periodically communicate their presence information to the cellular networks, cellphone networks can be used to estimate traffic density. The benefit of this approach is that since the current mobile networks are utilized, no extra infrastructure along the route is required. The technique could be challenging in regions where a single cellular tower supports two or more parallel pathways.

APPENDIX

```
#define ledA1 12 // green led for road ‘A’
 #define ledA2 52 //yellow led for road ‘A’
#define ledA3 13 // red led for road ‘A’
#define ledB1 11
#define ledB2 4
#define ledB3 10
#define ledC1 9
#define ledC2 2
#define ledC3 8
#define ledD1 7
#define ledD2 22
#define ledD3 6
int a1, a2, b1, b2, c1, c2, d1, d2,G;
void setup() {
 pinMode(ledA1, OUTPUT);
 pinMode(ledA2, OUTPUT);
 pinMode(ledA3, OUTPUT);
 pinMode(ledB1, OUTPUT);
 pinMode(ledB2, OUTPUT);
 pinMode(ledB3, OUTPUT);
 pinMode(ledC1, OUTPUT);
 pinMode(ledC2, OUTPUT);
 pinMode(ledC3, OUTPUT);
 pinMode(ledD1, OUTPUT);
 pinMode(ledD2, OUTPUT);
 pinMode(ledD3, OUTPUT);
 }
void loop() {
 readSensor(); //check which road has highest density:
 if (b1 == 1 && b2 == 1) // highest priority for road ‘B’
 { roadBopen(); }
 else if (d1 == 1 && d2 == 1 && (b1 == 0 || b2 == 0))
 { roadDopen();
 if (b1 == 1 && b2 == 1)
 {
 roadBopen(); }
 }
 else if (a1 == 1 && a2 == 1 && ((d2 == 0 || b2 == 0) || (d1 ==
0 || b1 == 0)))

{ roadAopen();
 if (b1 == 1 && b2 == 1)
 { roadBopen();
 }
 else if (d1 == 1 && d2 == 1 && (b1 == 0 || b2 == 0))
 { roadDopen(); }
 }
 else if (c1 == 1 && c2 == 1 && ((d2 == 0 || b2 == 0 || a2 == 0)
|| (d1 == 0 || b1 == 0 || a1 == 0))) /*least priority for road ‘C’ */
 { roadCopen();
 if (b1 == 1 && b2 == 1)
 { roadBopen();
 }
 else if (d1 == 1 && d2 == 1 && (b1 == 0 || b2 == 0))
 { roadDopen();
 } }
 else if ((b1 == 1 && b2 == 0) && (c1 == 1 || d1 == 1 || b1 ==
1) && (c2 == 0 && d2 == 0 && b2 == 0))
 { roadBopen();
 }
 else if ((d1 == 1 && d2 == 0) && (c1 == 1 || a1 == 1) && (b1
== 0 && b2 == 0) && (c2 == 0 && a2 == 0))
 { roadDopen();
 }
 else if ((a1 == 1 && a2 == 0) && (c1 == 1 && c2 == 0) && (d1
== 0 && d2 == 0) && (b1 == 0 && b2 == 0))
 { roadAopen();
 }
 else if ((c1 == 1 && c2 == 0) && (b1 == 0 || b2 == 0) && (d1
== 0 || d2 == 0) && (a1 == 0 || a2 == 0))
 { roadCopen();
 }
 else if ((b1 == 1 && b2 == 0) && (c1 == 0 || c2 == 0) && (d1
== 0 || d2 == 0) && (a1 == 0 || a2 == 0))
 { roadBopen();
 }
 else if ((d1 == 1 && d2 == 0) && (c1 == 0 || c2 == 0) && (b1
== 0 || b2 == 0) && (a1 == 0 || a2 == 0))
 { roadDopen();
 }
 else if ((a1 == 1 && a2 == 0) && (c1 == 0 || c2 == 0) && (d1
== 0 || d2 == 0) && (b1 == 0 || b2 == 0))
 { roadAopen();
 }
 else if (a1 == 0 && b1 == 0 && c1 == 0 && d1 == 0) // if all
roads are empty
 {
 digitalWrite(ledA3,LOW);
 digitalWrite(ledB3, LOW);
 digitalWrite(ledC3, LOW);
 digitalWrite(ledD3, LOW);
 digitalWrite(ledA1, LOW);
 digitalWrite(ledB1, LOW);
 digitalWrite(ledC1, LOW);
 digitalWrite(ledD1, LOW);
 G=0;
 readSensor();
 }
 }
void readSensor() // IR read function
```

```
{
 a1 = analogRead(A7);

 a2 = analogRead(A6);
 b1 = analogRead(A4);
 b2 = analogRead(A5);
 c1 = analogRead(A1);
 c2 = analogRead(A0);
 d1 = analogRead(A3);
 d2 = analogRead(A2);
 if (a1 < 400) a1 = 1; else a1 = 0; if (a2 < 400) a2 = 1; else a2
= 0;
if (b1 < 400) b1 = 1; else b1 = 0; if (b2 < 400) b2 = 1; else b2
= 0;
if (c1 < 400) c1 = 1; else c1 = 0; if (c2 < 400) c2 = 1; else c2 =
0;
if (d1 < 400) d1 = 1; else d1 = 0; if (d2 < 400) d2 = 1; else d2
= 0;
}
// Open road 'X' function:
void roadAopen()
{
 // G used to save yellow led pin for the last opened road
 if (G != ledA2) /*if A is the last opened road then the yellow
LED still LOW because 'A' now opened AGAIN*/
 { digitalWrite(ledB1, LOW);
 digitalWrite(ledC1, LOW);
 digitalWrite(ledD1, LOW);
 digitalWrite(G, HIGH); /*yellow led is HIGH for the last
opened road only Green → Yellow*/
 delay(3000); // open yellow led for 3 sec
 digitalWrite(G, LOW);
 }
 digitalWrite(ledA3, LOW);
 digitalWrite(ledA1, HIGH); //Green led road 'A'
 digitalWrite(ledB3, HIGH);
 digitalWrite(ledC3, HIGH);
 digitalWrite(ledD3, HIGH);
 delay(2000); /*keep road 'A' open for 2sec and check density
again */
 G = ledA2; /*save last opened road G= road 'A' yellow led
pin */
 readSensor();
} //Repreat concept for other roads :
void roadBopen()
{ if (G != ledB2)
{ digitalWrite(ledA1, LOW);
 digitalWrite(ledC1, LOW);
 digitalWrite(ledD1, LOW);
 digitalWrite(G, HIGH);
 delay(3000);
 digitalWrite(G, LOW); }
 digitalWrite(ledB3, LOW);
 digitalWrite(ledA3, HIGH);
 digitalWrite(ledB1, HIGH);
 digitalWrite(ledC3, HIGH);
 digitalWrite(ledD3, HIGH);
 delay(2000);
 G = ledB2;
 readSensor();
}
void roadCopen()
{ if (G != ledC2)
{ digitalWrite(ledA1, LOW);
 digitalWrite(ledB1, LOW);
 digitalWrite(ledD1, LOW);
 digitalWrite(G, HIGH);
 delay(3000);
 digitalWrite(G, LOW);
 }
 digitalWrite(ledC3, LOW);
 digitalWrite(ledA3, HIGH);
 digitalWrite(ledB3, HIGH);
 digitalWrite(ledC1, HIGH);
 digitalWrite(ledD3, HIGH);
 delay(2000);
 G= ledC2;
 readSensor();
}
void roadDopen()
{ if (G != ledD2)
{ digitalWrite(ledA1, LOW);
 digitalWrite(ledB1, LOW);
 digitalWrite(ledC1, LOW);
 digitalWrite(G, HIGH);
 delay(3000);
 digitalWrite(G, LOW);
}
digitalWrite(ledD3, LOW);
 digitalWrite(ledA3, HIGH);
 digitalWrite(ledB3, HIGH);
 digitalWrite(ledC3, HIGH);
 digitalWrite(ledD1, HIGH);
 delay(2000);
 G= ledD2;
 readSensor();
```

REFERENCES

[1] Morten Jensen, Mark Philipsen, Andreas Møgelmose, Thomas Moeslund, and Mohan Trivedi,Vision for Looking at Traffic Lights: Issues, Survey, and Perspectives. IEEE transactions on intelligent transportation systems, 2016; 17(7). https://doi.org/10.1109/TITS.2015.2509509

[2] A. Vahidi and A. Eskandarian, "Research advances in intelligent collision avoidance and adaptive cruise control," IEEE Trans. Intel. Transport. Syst. 2003; 4(3): 143-153. https://doi.org/10.1109/TITS.2003.821292

[3] E. Brockfeld, R. Barlovic, A. Schadschneider, and M. Schreckenberg, "Optimizing traffic lights in a cellular automaton model for city traffic," Phys. Rev. E, 2001; 64(5). https://doi.org/10.1103/PhysRevE.64.056132

[4] DW. Huang and WN. Huang, "Traffic signal synchronization," Phys. Rev. E, 2003; 67(5). https://doi.org/10.1103/PhysRevE.67.056124

[5] C. Gershenson, "Self-organizing traffic lights," Complex Syst. 2005; 16(1): 29-53.

[6] K. Dresner and P. Stone, "A multiagent approach to autonomous intersection management," J. Artificial Intel. Res. 2008; 31: 591-656. https://doi.org/10.1613/jair.2502

[7] M. Van Middles worth, K. Dresner, and P. Stone, "Replacing the stop sign: unmanaged intersection control for autonomous vehicles," in Proc. 5th Workshop Agents Traffic Transport. Multiagent Syst., Estroil, Portugal, 2008; pp. 94-101.

[8] MM. Edardar and Ahmed J. Abougarair, Adaptive Traffic Light Dynamic Control Based on Road Traffic Signal from Google Maps, ICEMIS'21, October 11-13, 2021, Almaty, Kazakhsta. https://doi.org/10.1145/3492547.3492584

[9] He Z, Guan W, Ma S. A traffic-condition-based route guidance strategy for a single destination road network. Transp. Res. Part C Emerg. Technol, 2013; 32: 89-102. https://doi.org/10.1016/j.trc.2013.04.005

[10] Mekki T, Jabri I, Rachedi A, ben Jemaa M. Vehicular cloud networks: Challenges, architectures, and future directions. Veh. Comm., 2017; pp. 268-280. https://doi.org/10.1016/j.vehcom.2016.11.009

[11] Michel Ferreira and Pedro Orey.2012. On the Impact of Virtual Traffic Lights on Carbon Emissions Mitigation. IEEE transactions on intelligent transportation systems, 2012; 13(1). https://doi.org/10.1109/TITS.2011.2169791

[12] Ahmed J. Abougarair Design and implementation of smart voice assistant and recognizing academic words, International Robotics & Automation Journal 2022; 8(1): 27-32.

[13] Mohamed Edardar, Ahmed J. Abougarair, Lyapunov Redesign of Piezo-Actuator for Positioning Control, 2021 9th International Conference on Systems and Control (ICSC), PP. 499-503, France, 2021. https://doi.org/10.1109/ICSC50472.2021.9666594

[14] E. Brockfeld, R. Barlovic, A. Schadschneider, and M. Schreckenberg. Optimizing traffic lights in a cellular automaton model for city traffic. Phys. Rev. E, 2001; 64(5). https://doi.org/10.1103/PhysRevE.64.056132

[15] D. Huang and W. Huang.2003. Traffic signal synchronization. Phys. Rev. E, vol. 67, no. 5, 2003. https://doi.org/10.1103/PhysRevE.67.056124

[16] Nagmden M. Nasser and Ahmed J. Abougarair, Traffic Sign Classification Using Deep Learning Based Convolutional Neural Networks, Azzaytuna University Journal, 2022; 3(4): 2, PP. 303-326.

[17] K. Dresner and P. Stone, A multiagent approach to autonomous intersection management. J. Artificial Intel. Res., vol. 31, pp. 591-656, 2008. https://doi.org/10.1613/jair.2502

[18] M. worth, K. Dresner, and P. Stone, Replacing the stop sign: unmanaged intersection control for autonomous vehicles. in Proc. 5th Workshop Agents Traffic Transport. Multiagent Syst., Estoril, Portugal, pp. 94-101, May 2008.

[19] U. S. Department of Transportation, Washington, DC. Urban drive control. [Online]. http://www.itsbenefits.its.dot.gov/its/benecost.nsf/ID/876D690DAB8FCA6B8525733A006D5BCC?OpenDocument&Query=BApp.

[20] Mohamed K. Aburakhis, Ahmed J Abougarair, Mohammed Elmaryu, Real time traffic sign detection and recognition for autonomous vehicle, Vol. 8, Issue.2, PP. 82-87, International Robotics & Automation Journal.

[21] H. Guan.2013. A traffic-condition-based route guidance strategy for a single destination road network. Transp. Res. Part C Emerge. Technol, 32, pp. 89-102. https://doi.org/10.1016/j.trc.2013.04.005

[22] T. Mekki and T. Jabri. 2017. Vehicular cloud networks: Challenges, architectures, and future directions. Veh. Comm., pp. 268-280. https://doi.org/10.1016/j.vehcom.2016.11.009

[23] K. Hewage and J. Ruwanpura.2004. Optimization of traffic signal light timing using simulation". In WSC '04: Proceedings of the 36th conference on Winter simulation, pp. 1428-1436.

[24] L. Klein. 2001. Sensor Technologies and Data Requirements for ITS. Artech House Publishers: Norwood, MA, USA.

[25] Ahmed J. Abougarair, Intelligent control designs for linear model of active suspension system International Conference on Microelectronics (ICM), 17-20, 2018. https://doi.org/10.1109/ICM.2018.8703995

[26] Ahmed J Abougarair, Position and Orientation Control of A Mobile Robot Using Intelligent Algorithms Based Hybrid Control Strategies, Journal of Engineering Research, 2022; 34: PP. 67-86.

[27] Ahmed J. Abougarair. Model Reference Adaptive Control and Fuzzy Optimal Controller for Mobile Robot. Journal of Multidisciplinary Engineering Science and Technology (JMEST), 2019; 6(3): 9722-9728.

[28] A. Nicoli, M. Deflorio, F. Chiara and B. Spagnolini. Wireless sensor networks for traffic management and road safety. J. Intel. Transp. Syst. 2012; pp. 67-77, IET 2012. https://doi.org/10.1049/iet-its.2010.0129

Received on 07-11-2022 Accepted on 25-11-2022 Published on 05-12-2022

DOI: https://doi.org/10.31875/2409-9694.2022.09.09