IJCRT.ORG ISSN : 2320-2882

INTERNATIONAL JOURNAL OF CREATIVE RESEARCH THOUGHTS (IJCRT)
An International Open Access, Peer-reviewed, Refereed Journal

DENSITY BASED TRAFFIC CONTROL AND EMERGENCY VECHICLES ALERT

1Mrs. P S S N Mowlika,2Mrs.S .Swathi, K. V G Kasi Viswanath, K. Dharma Teja, M. Pradeep Kumar, G. Anand Vamsi Reddy

[1,2]Assistant Professor, [2345] UG Scholars

Department of electronics and Communications Engineering,

Sagi Rama Krishnam Raju Engineering College, Bhimavaram, Andhra Pradesh, India

Abstract: This project presents a novel approach to traffic management through a Density-Based Traffic Control System (DBTCS) enhanced with emergency vehicle priority detection. The system utilizes infrared (IR) sensors to monitor traffic density in each lane, allowing for real-time adjustment of traffic signal timings based on traffic density levels. By dynamically adapting signal timings, the system aims to optimize traffic flow, reduce congestion, and enhance overall efficiency of urban road networks.

In addition to density monitoring, the system integrates a sound sensor to detect approaching emergency vehicles. Upon detection, the system promptly identifies the lane of the emergency vehicle and grants priority by adjusting signal timings to facilitate its passage. This prioritization of emergency vehicles enhances response times for first responders, potentially saving lives in critical situations.

The proposed system offers a flexible and adaptive solution to traffic management challenges, capable of responding to changing traffic conditions in real-time while ensuring efficient prioritization of emergency vehicles. Through simulation and experimental validation, the effectiveness and feasibility of the DBTCS with emergency vehicle priority detection are demonstrated, highlighting its potential for implementation in urban traffic infrastructure.

Index Terms – traffic density control, infrared sensor, emergency vehicle detection, sound sensor.

I. INTRODUCTION

Major cities worldwide face a significant challenge in urban traffic congestion, impacting both times spent in traffic and carbon dioxide emissions. Prolonged idling of engines during traffic exacerbates environmental concerns. Addressing this issue not only enhances urban residents' lives but also contributes positively to the climate. The current traffic light system operates as an open-loop control system, with a typical arrangement featuring red, yellow, and green lights functioning on fixed time intervals, irrespective of traffic density. This fixed timing approach leads to operational challenges, such as traffic light mismanagement, signal jumping by drivers, increased wait times, and petroleum loss. Numerous proposed solutions advocate for automatic traffic signals responsive to traffic density rather than fixed timings. These solutions leverage technologies like PIC microcontrollers, IR sensors, X bee communication, innovative algorithms (fuzzy and genetic), IoT devices, and various methodologies. This article introduces a self-adaptive traffic light system that detects traffic density and allocates a green signal to the lane with the highest density using infrared sensors (IR). The prototype employs two IR sensors per route, adaptable for real-life scenarios with multiple sensors per route. Density measurement is determined by the number of IR sensors detecting an object. For instance, a lane with two IR sensors detecting traffic receives a green signal for a specified time interval from the Arduino microcontroller board. Subsequently, the first lane receives a green signal after the set time elapses. When no vehicles are present, green signals cycle through each lane sequentially for a specified duration until an IR sensor detects traffic. This paper measures IR sensor output based on object detection, controlling traffic lights through control signals from the Arduino microcontroller board. Components and their Description: The essential components employed in crafting this prototype encompass the Arduino UNO ATmega328P microcontroller board, IR sensors, red light-emitting diodes (LEDs), yellow LEDs, green LEDs, connecting wires, Arduino power cable, and resistors.

1.1 BLOCK DIAGRAM

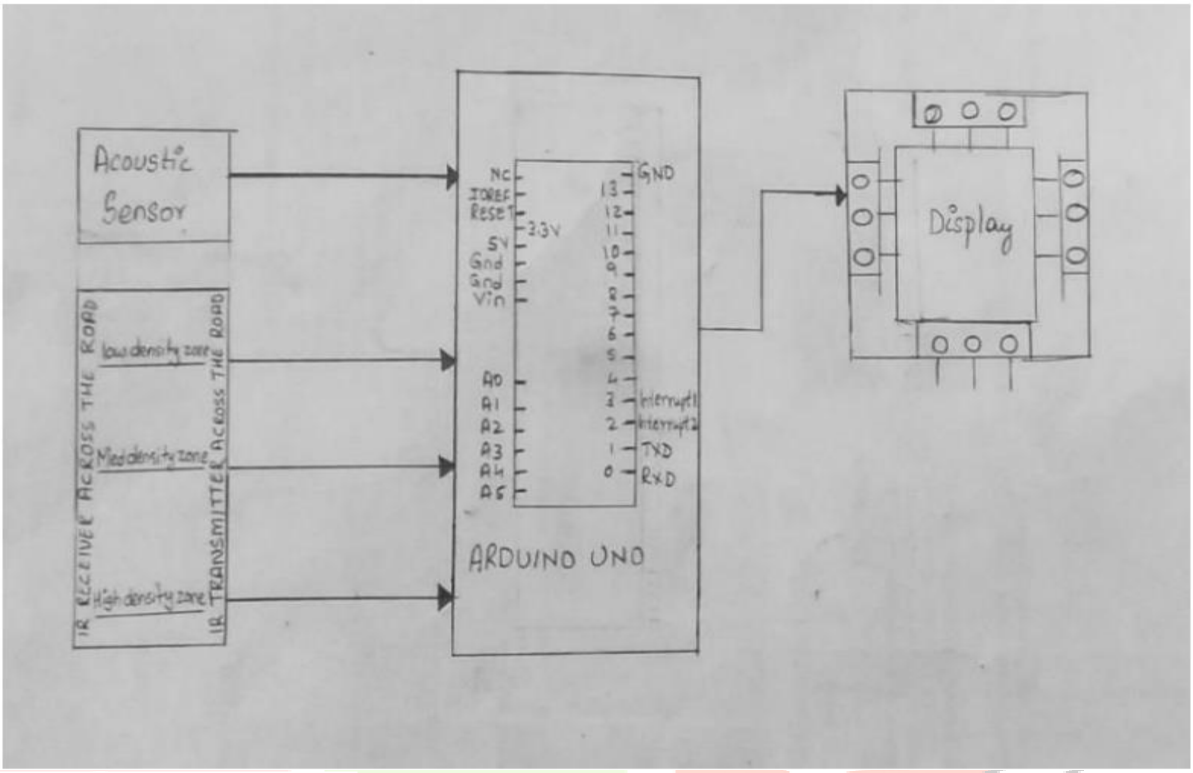

Figure -1: Block diagram of the proposed system

The basic block diagram of the density-based traffic control system is as shown in figure 1. It consists of four roads, IR Sensors, Acoustic Sensor, a microcontroller unit and signal lights (LED's).

The components required are as follows:

1. IR Sensor
2. Light emitting diode
3. Arduino UNO
4. Sound Sensor

1.1.1 IR Sensor

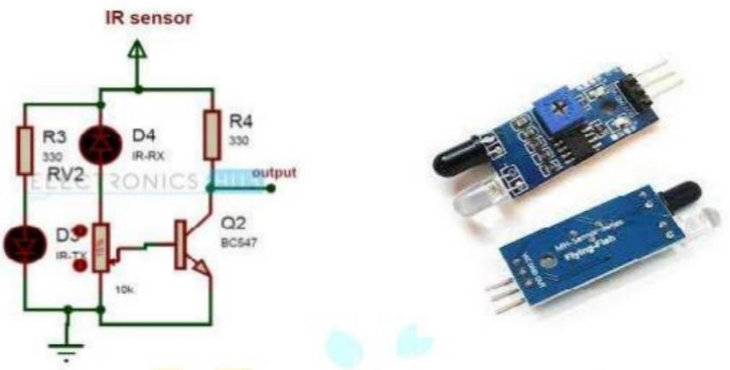

Figure -2: IR Sensor

An IR sensor comprises key elements such as an operational amplifier (Op-amp), a variable resistor, an LED for output indication, and both an IR transmitter and receiver.

The IR sensor is equipped with three pins: VCC, GND, and O/P. VCC supplies power to the sensor, GND is linked to the Arduino's ground, and O/P delivers the sensor's output signal to the Arduino UNO.

1. IR Transmitter: This component, an IR LED, emits light in the infrared frequency, which is invisible to the naked eye due to its higher wavelength than visible light. The IR LED is designed to be white or transparent to maximize light emission.

2. IR Receiver (Photodiode Receiver): Functioning as an IR receiver, a photodiode conducts electricity when exposed to light. The photodiode within the IR receiver consists of a reverse-biased P-N junction semiconductor. When illuminated, it conducts current in the reverse direction. Covered with a black surface to enhance light absorption, the photodiode, akin to an LED, produces current flow proportional to the absorbed light.

1.1.2 Light Emitting Diode

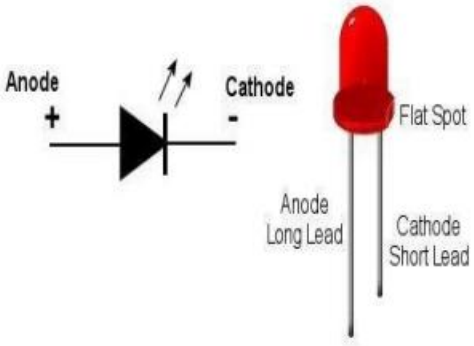

Figure -3: Symbol of LED

The term LED refers to a light-emitting diode, a semiconductor device similar to a diode but distinguished by its emission of light when a voltage of the correct polarity is applied across its two terminals—one being the anode and the other the cathode. The colour of the LED is determined by the semiconductor material employed in its production. Red, yellow, and green LEDs are specifically chosen for their appearance in a traffic light configuration.

1.1.3Arduino UNO

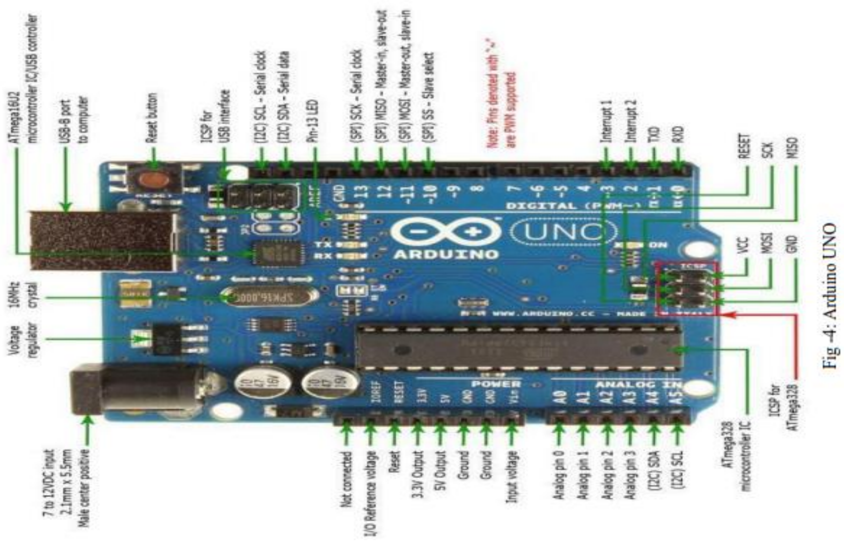

Figure -4: Arduino UNO

The Arduino UNO is a development board featuring an ATmega328P microcontroller, designed for project prototyping. With its expansive capabilities, the board offers 54 digital input-output pins, 15 of which support pulse width modulation (PWM). Additionally, it includes 16 Analog input pins, 4 serial ports (UART), USB connectivity, a Power Jack, a 16 MHz crystal oscillator, and a reset button.

The programming of the Arduino board is accomplished using the Arduino IDE freeware, which is based on the C++ programming language. The code is transferred to the Arduino UNO board through the USB D-type port. In the creation of the traffic control prototype, a total of 20 digital input-output pins, GND, 5V pin, and the USB port (for code uploading and power supply from the laptop) have been employed.

1. Digital Input-Output Pins: These pins receive digital signals from sensors, which output digital signals. They can also function for digital output, representing 1 as 3.5V - 5V and 0 as 0 - 2V.

2. Analog Pins: Utilized for both Analog input and output within the range of 0-5V.

3. Reset: This feature resets the microcontroller board.

4. Serial Pins (Rx, Tx): These pins facilitate serial communication, with Rx for receiving and Tx for transmitting TTL serial data.

5. PWM Pins: Offering pulse-width modulation output, these pins enable Analog output for coordinating switching operations among multiple switches. The programming command "Analog Write" (PWM pin, value) is employed for this purpose.

6. Vin: Serving to supply input power to the Arduino board if an external 5V source is available.

7. 5V Pin: Used to provide 5V power to both onboard and external components when powered through the Arduino board.

8. 3.3V: This regulated supply, generated by a voltage regulator on the Arduino UNO board, is available to supply external components if necessary.

9. GND Pins: These pins provide ground to external components.

1.1.4 Sound Sensor

Figure -5: Sound Sensor

The sound detector is one type of module used to notice the sound. Generally, this module is used to descry the intensity of sound. The operations of this module substantially include switch, security, as well as covering detector

II.THE TRAFFIC LIGHT SYSTEM

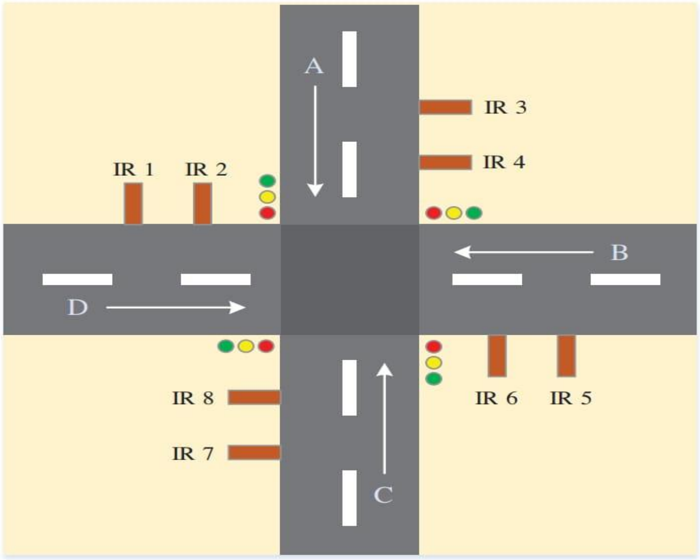

Figure -6: Architecture of four-way road intersection with the IR sensor module

The schematic of the four-way road intersection incorporating the IR sensor and sound sensor module. In this representation, A, B, C, and D denote the different roads, and directional arrows within each lane indicate the flow of traffic on that particular road. The IR sensors, labelled IR 1 and IR 2, are oriented towards road D, where the IR transmitter and photodiode receiver face this road. Similarly, IR 3 and IR 4 are designated for road A, while IR 5, IR 6, IR 7, and IR 8 are allocated for road B and C, respectively.

When a vehicle approaches an IR sensor, the output signal serves as an indicator of traffic density. The higher the number of IR sensors detecting traffic in a specific lane, the greater the traffic density on that road. To determine the allocation of green signals to roads, the following rules are implemented:

1. When no IR sensor detects a signal, green signals are sequentially given to each road in the order B, D, C, A for a set duration.

2. Traffic congestion near inner IR sensors (IR 2, IR 4, IR 6, IR 8) is prioritized over outer IR sensors (IR 1, IR 3, IR 5, IR 7). If only an inner IR sensor on a road detects a signal, that road receives a green signal until traffic conditions change.

3. If only outer IR sensors detect traffic, Rule 1 is executed, assuming that traffic does not intend to move forward.

4. If internal IR sensors on two or more roads detect traffic, one road opens based on priority in the programming code, receiving a green signal until traffic density decreases. Subsequent roads are given green signals if traffic parameters for the other roads remain unchanged.

5. If both IR sensors on one road detect traffic, that road is given a green signal until traffic subsides.

6. If both sensors on two or more roads detect traffic, one road opens based on code priority and receives a green signal until traffic density decreases. The next road gets a green signal if traffic parameters for the other roads remain consistent.

7. If both sensors on a road detect traffic and in remaining roads, only one sensor detects traffic, the road with both sensors receives a green signal.

8. If both sensors on two or more roads detect traffic, and in the remaining roads, either one or no sensor detects traffic, the road with two sensors detecting traffic gets a green signal based on code priority.

III. AMBULANCE DETECTION

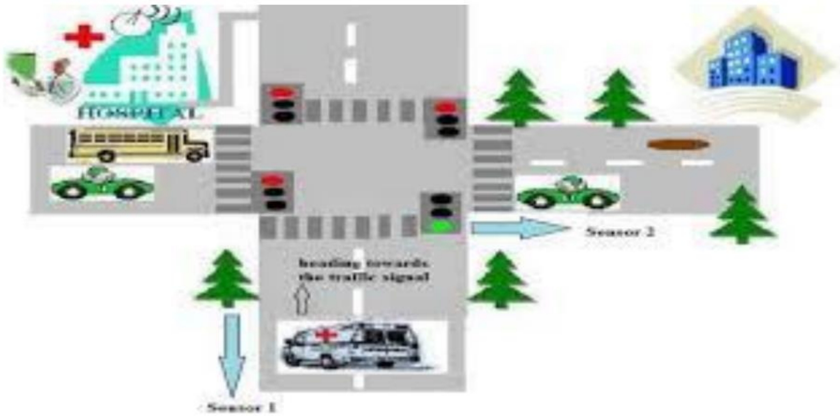

Figure-7: Ambulance detection with acoustic sensors.

Let’s consider a scenario with a four-lane intersection, where each lane has a dedicated sound sensor to detect emergency vehicles. Here's how the sound sensor would work in various cases:

1.Single Emergency Vehicle in One Lane:
- Case 1: If only one sound sensor detects an emergency vehicle, the lane associated with that sensor receives immediate priority.
- Example: An ambulance approaches from lane A, triggering the sound sensor in lane A. The traffic light for lane A turns green while lights for other lanes remain unchanged.

2. *Multiple Emergency Vehicles in Different Lanes: *
- Case 2: If multiple sound sensors detect emergency vehicles in different lanes simultaneously, priority is determined based on the urgency of the situations.
- Example: An ambulance approaches from lane A, while a fire truck approaches from lane B. Both sound sensors detect the emergency vehicles simultaneously. The traffic management system prioritizes the lane with the more urgent situation, perhaps giving priority to the fire truck in lane B over the ambulance in lane A.

3. *Multiple Emergency Vehicles in the Same Lane: *
- Case 3: If multiple emergency vehicles are detected by the same sound sensor in one lane, the system prioritizes the most urgent vehicle.
- Example: Two ambulances approach from lane C, triggering the sound sensor in lane C. The traffic management system evaluates the urgency of each situation (e.g., critical patients versus non-critical patients) and prioritizes the passage of the ambulance with the most critical situation.

4. *Combination of IR and Sound Sensor Inputs: *
- Case 4: If both IR sensors and sound sensors detect traffic and emergency vehicles simultaneously, the system intelligently prioritizes the most urgent situation while managing traffic flow efficiently.
- Example: A traffic jam occurs in lane D while a police car approaches from lane B with its siren on. Both IR sensors in lane D and the sound sensor in lane B detect the respective situations. The system gives priority to the police car in lane B to clear the way for emergency response while managing traffic in lane D to alleviate congestion.

In each case, the sound sensors work in conjunction with the IR sensors to ensure the swift passage of emergency vehicles while efficiently managing traffic flow at the intersection.

IV.HARDWARE IMPLEMENTATION

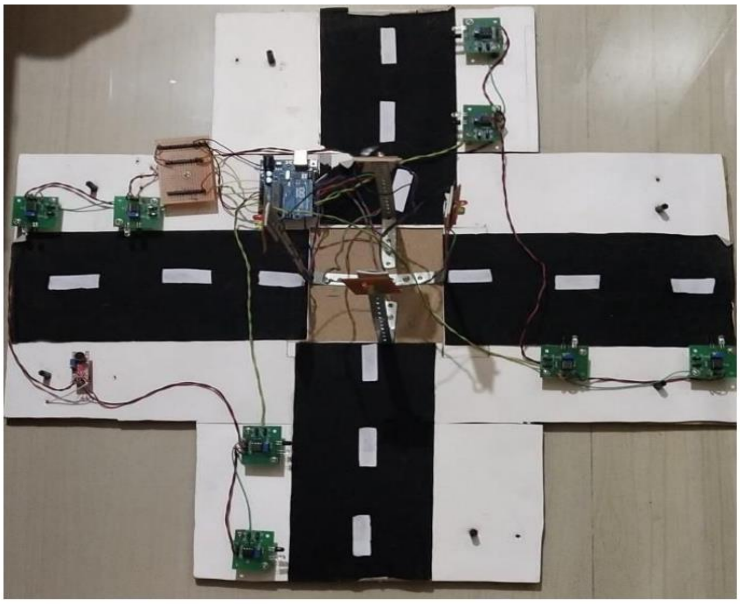

Figure-8: Demonstration Board

In this section, a comprehensive step-by-step guide is provided for connecting various components in a manner that allows anyone to create the prototype by following the outlined process. Due to the complexity of illustrating all connections in a single figure, the connections for a single road are demonstrated in Figure 9. The same approach can be applied to connect components for other roads. For a single road, two IR sensors, three LEDs (red, yellow, and green), and some resistors are required. The sequential process for connecting these components to the Arduino Mega microcontroller board is as follows:

1. Arrange an Arduino Mega, two IR sensors, three LEDs, and three resistors according to Figure 9. Note that the longer leg of the LEDs is the bent leg.

2. Establish a common ground bus (Bus GND) and a common positive bus (Bus +Ve). Connect Arduino GND to Bus GND and its 5V to Bus +ve.

3.Connect the GND of the first IR sensor to Bus GND. Connect its VCC to Bus +Ve, and its output to Arduino pin 5.

4. Connect the GND of the second IR sensor to Bus GND. Connect its VCC to Bus +Ve, and its output to Arduino pin 6.

5. Connect the Vin of the yellow LED to Arduino pin 4. The Vin is identified by the bent leg of the LED. Connect the resistor of the yellow LED to Bus GND.

6.Connect the Vin of the green LED to Arduino pin 6 and connect its resistor to Bus GND.

7.Connect the Vin of the red LED to Arduino pin 3 and connect its resistor to Bus GND.

8.Connect the Arduino board to the computer using a USB string and upload the law algorithm.

9. Open the serial monitor in Arduino IDE and adjust the position of the onboard trimmer to modify the object detection distance by the IR sensor.

10.Verify all rules by following the above steps. If any errors occur, troubleshoot accordingly. Following the steps outlined above, connections have been established for all four roads in the self-adaptive traffic control system. The completed prototype is depicted in Figure 10, illustrating the hardware connections of all components with the Arduino board. The sensors are uniformly spaced at a fixed gap of 1.5cm, and the length of each road is maintained consistently at 10.5cm.

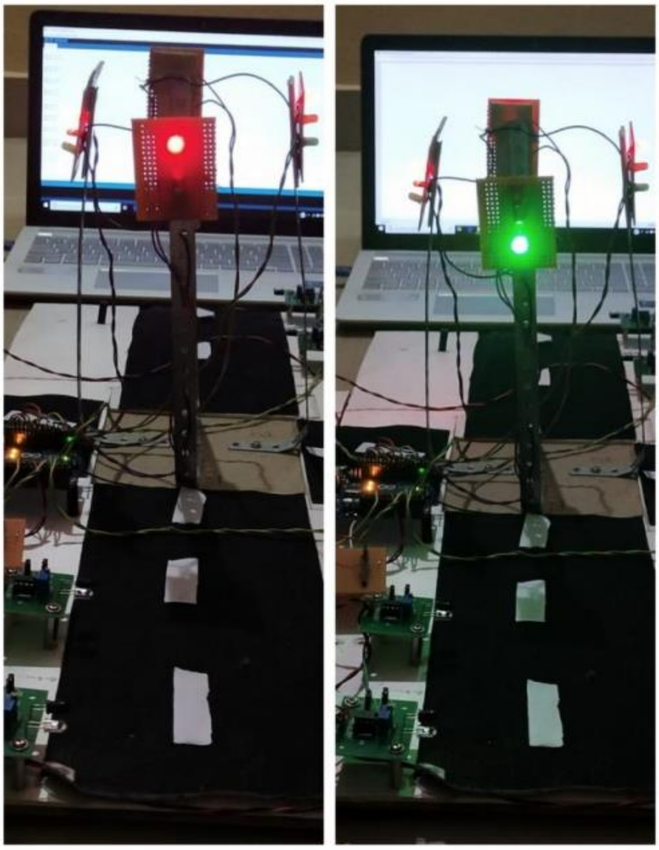

Figure-10: Result of the Density Based Traffic Control System

To implement a sound sensor in the self-adaptive traffic control system, follow these additional steps:

11.Obtain a sound sensor module compatible with Arduino.

12. Connect the sound sensor's GND pin to Bus GND, VCC pin to Bus +Ve, and its Analog output pin to any available Analog pin on the Arduino Mega (e.g., A0).

13. Upload the appropriate code to the Arduino Mega microcontroller board. This code should include instructions to read the Analog input from the sound sensor.

14. Open the serial monitor in the Arduino IDE to observe the readings from the sound sensor. Adjust any necessary parameters or thresholds in the code to detect sound levels indicative of traffic or other relevant events.

15. Integrate the sound sensor readings into the existing algorithm for traffic control. For example, you can use sound levels to prioritize certain roads or adjust traffic light timings based on ambient noise levels.

16. Test the entire system, including the newly added sound sensor, to ensure proper functionality. Make any necessary adaptations or troubleshoot any issues that arise.

V CONCLUSION AND FUTURE WORK

The implementation of the self-adaptive traffic light system is straightforward and can be applied in real-life scenarios, potentially integrated with a manual control system to enhance reliability. This model is particularly suitable for intersections where traffic flows in a single direction on each road.

In practical implementation, a total of eight IR sensors were strategically placed—two sensors for each road—to effectively detect traffic density. Priority is given to internal sensors, meaning that if only the internal sensor detects traffic, the corresponding road is granted a green signal. Conversely, if only external sensors detect traffic, the road does not receive a green signal. While the practical implementation utilized two sensors per road, more sensors may be employed for increased accuracy.

The sensing time for detecting traffic in hardware implementation is set to 10 seconds. During this time, the road with the highest traffic density is given a green signal, and this cycle repeats until the sensor detects traffic density again.

The prototype model can be adapted for use at intersections with more than four roads through simple modifications in the code. To enhance emergency vehicle responsiveness, a sound detection system could be integrated, ensuring that emergency vehicles can navigate through traffic without delays.

Furthermore, instead of the Arduino UNO microcontroller board, a more advanced Raspberry Pi board could be employed. The Raspberry Pi is an IoT-based device capable of seamlessly running multiple programs concurrently, unlike Arduino, which can execute only one program at a time. The Raspberry Pi's wireless communication feature eliminates the need for extensive wiring, which is particularly advantageous when traffic signals are spaced over a considerable distance at an intersection.

This prototype is well-suited for one or two-lane roads. Extending its applicability to accommodate more lanes per road could be a valuable future extension. Handling additional lanes would necessitate the integration of more IR sensors for traffic detection. Alternatively, employing cameras for a broader traffic overview is an option, requiring image processing tools and increased computational capabilities. While this approach might incur higher costs, the enhanced results could justify the investment.