

Embedded System PROJECT

Interfacing Traffic Light USING ULTRASONIC SENSOR

SUBMITTED TO

**Department of Computer Science & Engineering**

Team Members(CSE B):

Ananya Paleti – 220905081(11)

Shubhangi Shree – 220905114(16)

Manjyot Saini – 220905135(20)

Yash Thandani – 220905203(28)

Tarushi Tiwary – 220905171(24)

Introduction:

In urban environments, efficient traffic management is crucial for ensuring safety and smooth flow of vehicles and pedestrians. Interfacing traffic lights with advanced sensing technologies, such as ultrasonic sensors, offers a solution to optimize traffic control systems. In this project, we explore the integration of an ultrasonic sensor with an Arduino microcontroller to regulate traffic light signals based on real-time vehicle detection. By leveraging this technology, we aim to enhance traffic efficiency, reduce congestion, and improve overall road safety.

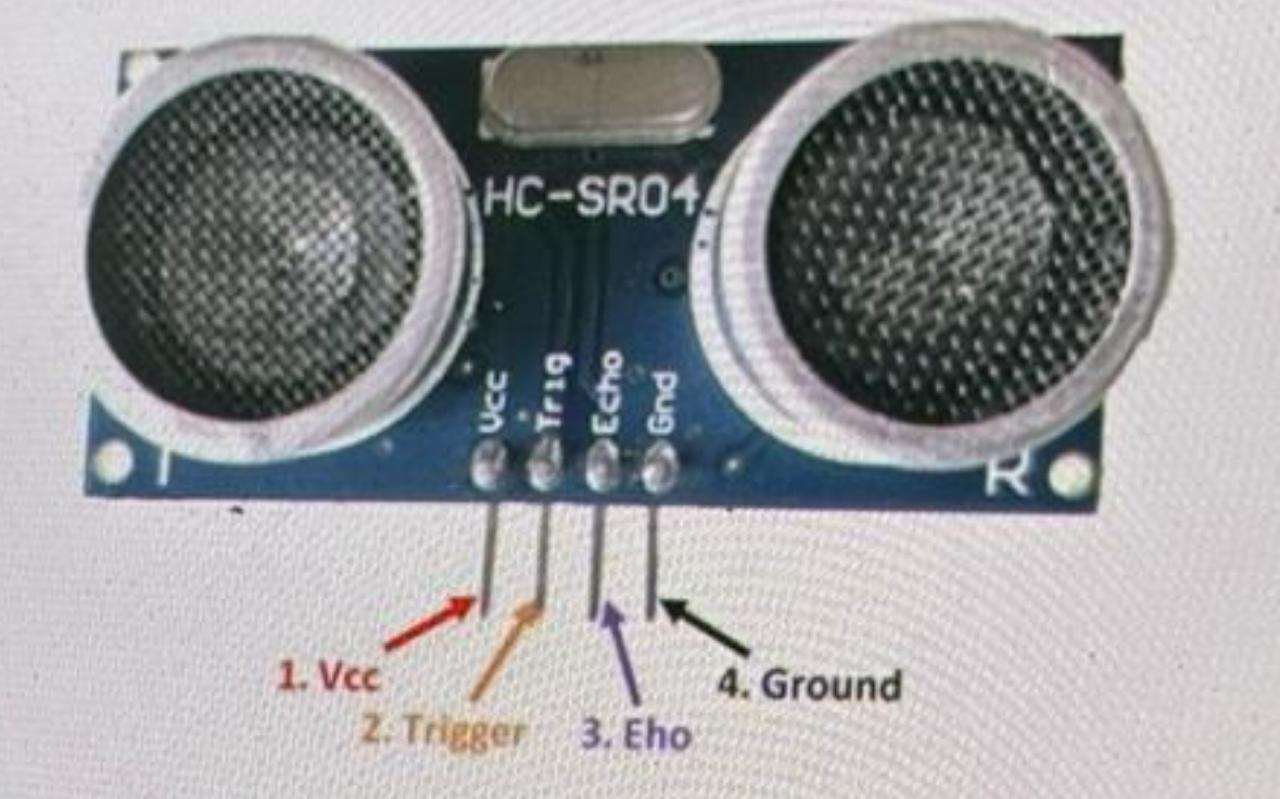
Components/Tools Used:

**ULTASONIC SENSOR**

-An ultrasonic sensor is an instrument that measures the distance to an object using ultrasonic sound waves.

-An ultrasonic sensor uses a transducer to send and receive ultrasonic pulses that relay back information about an object's proximity.

Pin out of Ultasonic sensor:



**VCC**: The power supply pin of the sensor that mainly operates at 5V DC.

Trig pin: This pin triggers the transmitter to send ultrasonic wave.

**Echo pin**: This pin gives signal in pulse form when the receiver receives the reflected wave from the object

**GND**: The power supply of the sensor that is connect to ground or

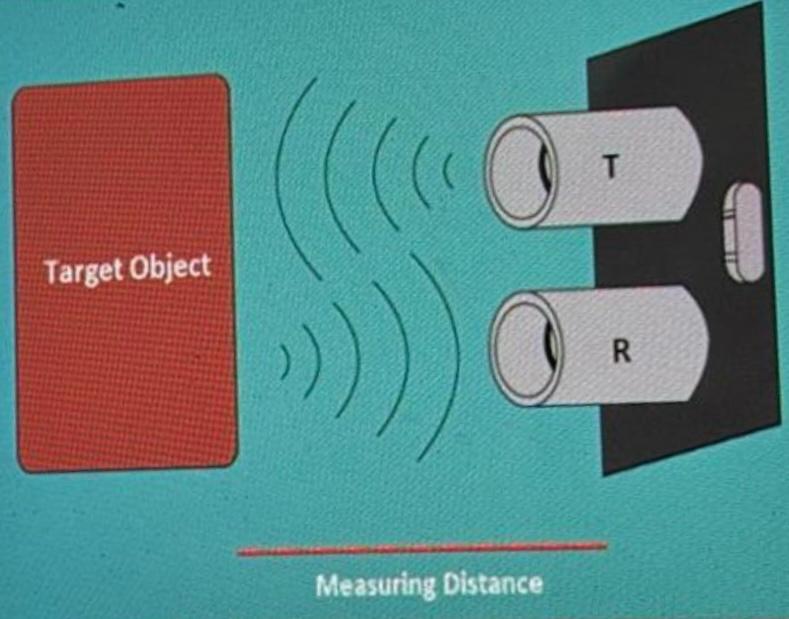
**Working of ultrasonic Sensor:**

As we connect the ultrasonic module to 5V.

initialize the input pin, it starts transmitting the sound waves which then travel through the airand hit the required object. These waves hit and bounce back from the objectthen collected by the receiver of the module.

Distance is directly proportional to the time these waves require to come back at the receiving end.

The more time is taken, the more the distance will be.



**Distance Calculation:**

The following formula is used to calculate the distance of the object :

**Distance =Time x Speed of Sound in Air (343m/s)/2**

• Sound waves take to come back after hitting the object.

• We need to divide the value by 2 because time will be doubled as the waves travel and bounce back from the initial point.

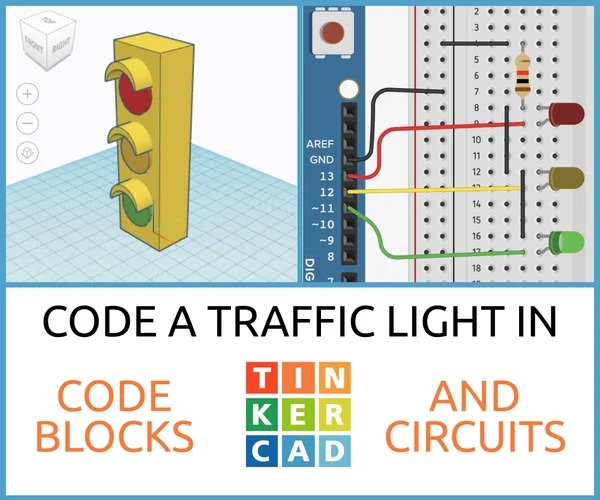
* Dividing it by 2 will give the actual distance of the target object.
* This project has been made on **ARDUINO** :

Arduino is a popular open-source platform used for building electronics projects. It consists of both hardware and software components, including a microcontroller board and an integrated development environment (IDE). Arduino is commonly used for prototyping, creating interactive electronic devices, controlling sensors and actuators, and developing IoT (Internet of Things) applications. It's widely utilized in fields such as hobbyist electronics, education, prototyping, home automation, and even in industrial settings for rapid prototyping and experimentation.

* The online simulations for the given code were implemented using an online simulator called **TINKERCAD**:

Tinkercad is a free web app for 3D design, electronics, and coding.

It is a web-based simulation tool which will help you to test your hardware as well as software without making any physical connection or even without buying any hardware.



|  |  |
| --- | --- |
| Name of Component | Quantity |
| Arduino Uno | 1 |
| Arduino Uno USB Cable | 1 |
| Breadboard | 1 |
| Resistor(220 Ohms) | 3 |
| LED | 3 |
| Ultrasonic | 1 |

Working:

• In this activity ultrasonic sensor is interfaced with 3 LEDs.

• This 3 LEDs will act as indicator at different range levels.

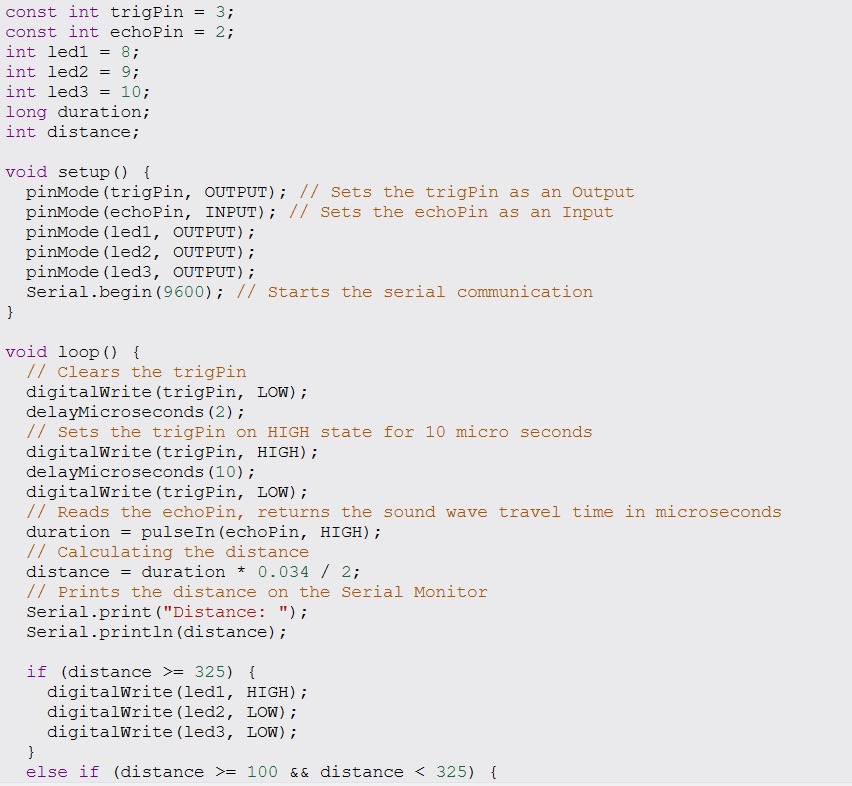
1.if Distance of Object is Greater than 325 then yellow Led will Glow.

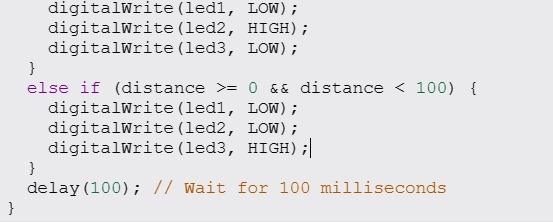
2.If Distance of Object is in between 100 to 325 then Green LED will Glow.

3. If Distance of Object is in less than 100 then Red LED will Glow.

Note: Change the distance range according to the requirement.

**CODE:**





The code is for an Arduino project that uses an ultrasonic sensor to measure distance and control three LEDs based on that distance. Here's a breakdown:

**1. \*Constants Declaration\*:**

- trigPin and echoPin are defined as the digital pins connected to the ultrasonic sensor for triggering and receiving signals.

- led1, led2, and led3 are defined as the digital pins connected to three LEDs.

**2. \*Global Variables Declaration\*:**

- duration stores the time taken for the ultrasonic pulse to travel.

- distance stores the calculated distance based on the duration.

**3. \*Setup Function\*:**

- pinMode() is used to set the mode of the pins: trigPin as output and echoPin as input.

- Also, the mode of led1, led2, and led3 pins are set as outputs.

- Serial.begin(9600) initializes serial communication at a baud rate of 9600.

**4. \*Loop Function\*:**

- The loop continuously executes the code inside.

- It starts by sending a short pulse to the ultrasonic sensor and measures the time taken for the pulse to bounce back.

- Using the formula distance = duration \* 0.034 / 2, it calculates the distance based on the time taken.

- The calculated distance is then printed to the Serial Monitor.

- Based on the distance measured, the code controls the LEDs:

- If the distance is greater than or equal to 325 units, only led1 is turned on.

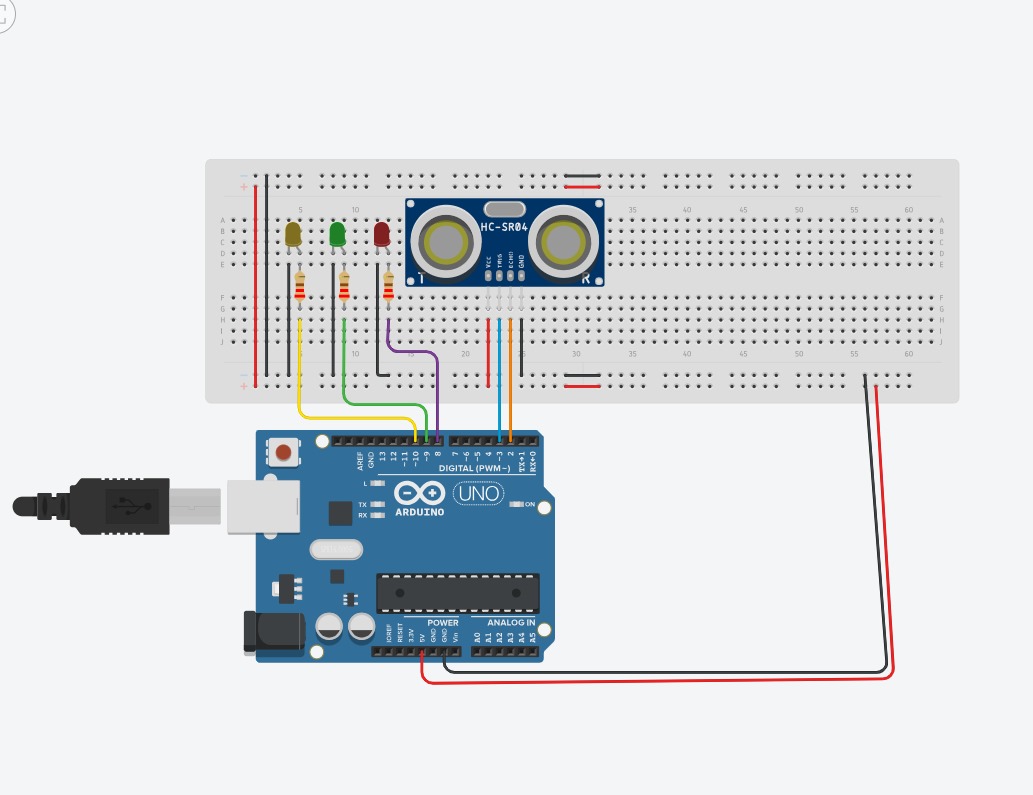
- If the distance is between 100 and 325 units, only led2 is turned on.

- If the distance is less than 100 units, only led3 is turned on.

- Finally, a delay of 100 milliseconds is added before repeating the process.

This code essentially creates a distance measurement system using an ultrasonic sensor and provides visual feedback using LEDs based on the measured distance.

RESULT :



It is observed that by altering the distance of the object infront of the ultrasonic distance sensor , the LED’s get turned on based on the distance of the object.

Advantages

•This system’s traffic light controller can be used in practice, and it can be expanded further by you.

•A crosswalk signalling mechanism is included in this traffic light controller.

•External memory can be interfaced with the central controller, allowing the timings to be programmed during operation rather than during programming.

Disadvantages

•The project is not intended for actual implementation, but more as a demonstration of the system’s process.

•The project can be run manually or with the help of pre-programmed activities. It is unable to function in both directions.

•The operator of a real-time traffic control system has the ability to adjust the timings and intensity of each lane’s traffic signal.

FUTURE SCOPE

•In the future, the model ambulance can be able to communicate with all base stations to get an easy free lane, to rush up reaching the hospital on time for needy people.

•An IOT-based real-time traffic monitoring system is proposed for the dynamic handling of traffic signals based on traffic density.

• Provides a real-time dashboard to monitor the traffic updates.

•This can save their time expansion for reaching the proposed destination and can prevent the loss of human life up to the great extent

REFERANCES:

1.https://docs.arduino.cc/hardware/uno-rev3/

2.<https://www.mathaelectronics.com/traffic-light-controller-using-arduino%EF%BF%BC/#>

3.<https://robu.in/traffic-light-controller-using-arduino/>

4. https://www.geeksforgeeks.org/traffic-light-control-project-using-arduino/