NAME: MURAGWABUGABO GILBERT

REG: 21RP10459

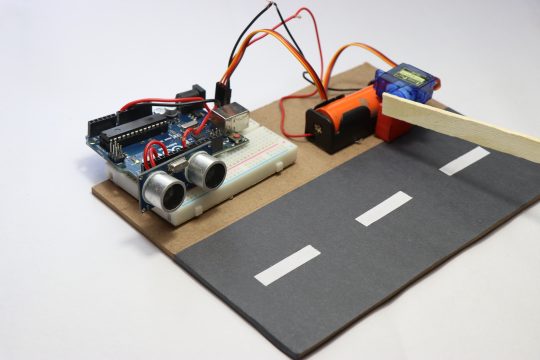
TASK: ASSIGNMENT

MODULE: INDUSTRIAL SENSOR

LOCATION: RP IPRC MUSANCE

HOW TO MAKE AUTOMATIC smart car barrier USING ARDUINO AND ULTRASONIC SENSOR.

Learn How to make Automatic Smart Car Barrier System with Arduino | Arduino Project. Here we will make a Smart Car Barrier System which are used in toll tax on highways, smart car parking systems, automatic gates by using Arduino, a servo motor and ultrasonic sensor.



INTRODUCTION

In this article we are making automated Toll tax system using Arduino Uno In very simple way, I have given step by step instructions along with detailed description about codes and circuit diagram. If you are thinking of project for science fair this is a wonderful option.

I have been working on intermediate and advanced level projects, I have been asked to start building some mini projects. So, here I have come up with this simple idea. You know the Ultrasonic sensor and Servo are quite popular and I know most of the beginners love to use these devices.

From this project, you will learn the most basic things, how to detect a car using ultrasonic sensor and then how to use the Servo to open the barrier.

Before going into our project building phase let us know few things about toll tax system or few call as toll plaza halting system.

Do you remember when you first visited toll plaza? And the wonderful stopping mechanism they used to stop the vehicles passing by to collect funds? Yes we are making the exact same project here but in a mini scale.

In this project, we are making an exact replica found in toll plaza centers that is called as stopping system, The idea for this project was inspired from actual system, in actual toll they stop the vehicles using a stopper that is completely automated and it is activated when any vehicle passes in front the sensor, or some time it’s activated through a button.

In our case, we are using an HC-SR04 or called as ultrasonic distance sensor to detect an obstacle (vehicle) and later to lift the barrier we are using micro servo, that is the mechanism involved in this project, now let us dive into the building stage

**Materials required to build automated toll system**

* Arduino Uno with atmega328P
* Ultrasonic sensor HC-SR04
* Plastic geared Micro servo
* Single strand wire instead of jumper wires
* Mini Breadboard
* Lithium ion battery with case
* Few drops of superglue
* Arduino programming cable
* Arduino IDE
* A Piece of popsicle stick

## ****Amazon Links:****

[Arduino Uno](https://amzn.to/3da4HDa)

[HC-SR04 Ultrasonic Sensor:](https://amzn.to/39o2P8t)

[Micro Servo Motors](https://amzn.to/3flYwhK)

**Other Tools and Components:**

[Top Arduino Sensors:](https://amzn.to/3vZbnfM)

[Super Starter kit for Beginners](https://amzn.to/3cGmLEv)

[Digital Oscilloscopes](https://amzn.to/3bSarSk)

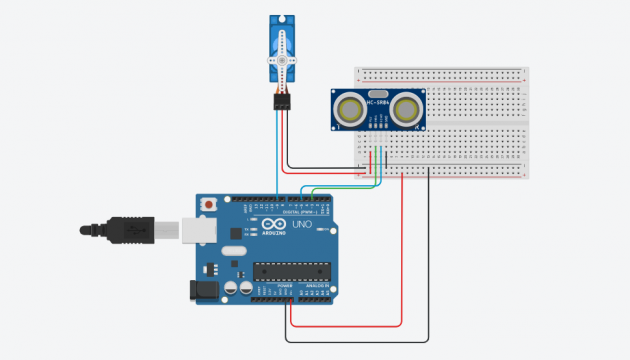
[Variable Supply](https://amzn.to/2NwnfVm)

[Digital Multimeter](https://amzn.to/3lmU4k5)

[Soldering iron kits](https://amzn.to/3s1LcTy)

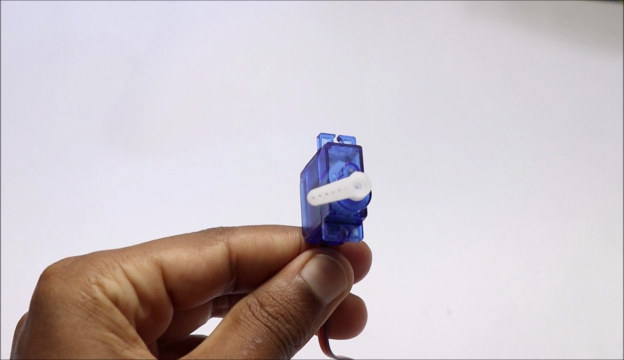
[PCB small portable drill machines](https://amzn.to/3loq478)

## ****Circuit Diagram of toll system:****



The circuit diagram of the Toll System is really simple. You can clearly see all the connections.

## ****Micro servo connections:****



Those who are new here have a look at the plastic geared micro servo as shown in the visual above, it has plastic gears and the servo horns carry a holder/barrier made from Popsicle stick.

Coming to the connections we are using D9 pin on Arduino Uno board which is PWM pin elaborated as pulse width modulations meaning the signals for micro servo goes in the form of pulses from D9 pin.

As we have connected D9 pin with signal input pin of micro servo we will be using the other 2 pins called as Ground and Positive to + and – pin of Breadboard power rails.

These completes servo connections now we will see the basic functions and connections of ultrasonic sensor.

The HC-SR04 or generally called by the name ultrasonic sensor or ultrasonic distance sensor has 4 pins.

These pins are called as VCC, Trig, Echo and Ground here as usual we will be connecting VCC and Ground pins with + and – rails of breadboard whereas Trig and Echo are connected to D5 and D3 pins of Arduino Board.

## ****Functions of Ultrasonic sensor****

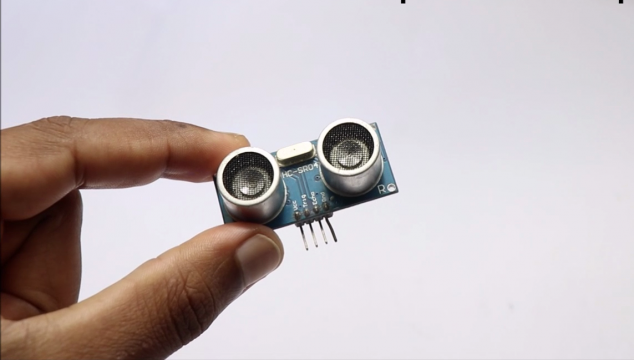
The actual image of HC-SR04 sensor is as shown in image below from the visual we can see 2 circle shaped structures.

These circles are nothing but transmitter and receiver parts, here when it is supplied with power, one end transmits ultrasonic signals and this travels up to accurate distance of 15cms in the travel path if it detects any obstacles the signals hit the object and travel back to the receiver end.

There are numerous applications of this small module and one of the application is in our project!

So coming to our project we are using vehicles as obstacle, when this sensor picks up signals from obstacle it sends signals to micro servo to lift for a specific time(as given in the code) and come to default state.

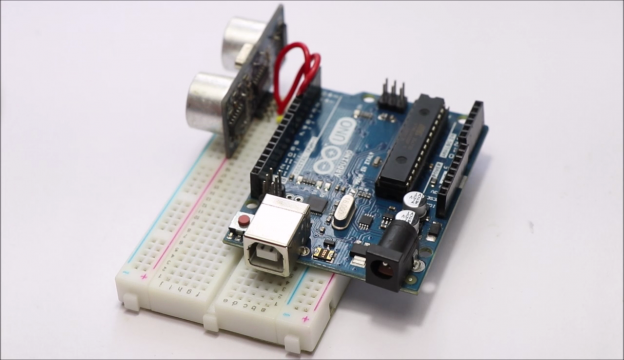
If you observe carefully you can see letters T and R at the corners of sensors these are Transmitter and receiver part.



This completes our circuit for this project.

**Tip:** To make the circuit and project look a lot cleaner I recommend to use single strand wires instead of jumpers as this avoids extra pieces of wires.

Below is the visual after assembling ultrasonic sensor.



You can see that almost the wires are hidden from outer area. For the ultrasonic power input the wires are running behind the Arduino Uno. To attach Arduino Uno to breadboard I used pieces of double sided adhesives/tape.

Once you add micro servo to this circuit we will have this setup as in below visual.

Note, as we are not using power supply from USB we need a lithium ion battery which delivers 3.7v dc supply to our project.

I used a battery holder which already had 2 wires for connection, to make it easier I added ends of jumper wires so that we can attach and detach power supply to breadboard power rails.

Don’t supply power to your project unless you add some code to the board.

## ****Code for Arduino automatic barrier for toll****

|  |  |
| --- | --- |
|  |  |

Some basics about codes used in our project.

#include<Servo.h>

Servo myservo;

const int trigPin=3;

const int echoPin=5;

long tmeduration;

int distance;

void setup() {

myservo.attach(9);

pinMode(trigPin,OUTPUT);

pinMode(echoPin,INPUT);

Serial.begin(9600);

}

void loop() {

digitalWrite(trigPin, LOW);

delayMicroseconds(2);

digitalWrite(trigPin, HIGH);

delayMicroseconds(10);

digitalWrite(trigPin, LOW);

tmeduration=pulseIn(echoPin,HIGH);

distance=(0.034\*tmeduration)/2;

if(distance<=50){

myservo.write(180);

}

else{

myservo.write(0);}

Serial.print("distance");

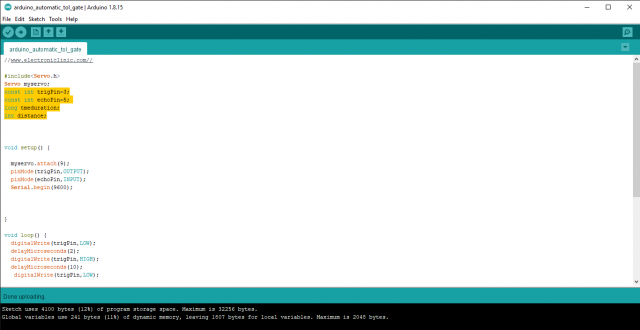
Serial.println(distance);

delay(2000);

}

To make it easier for beginner to understand and customize this code(if needed) I have split the codes into 3 parts.

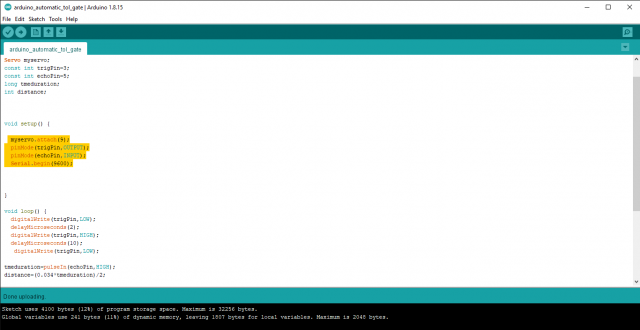
These 3 parts helps you to understand what these contribute to final working.



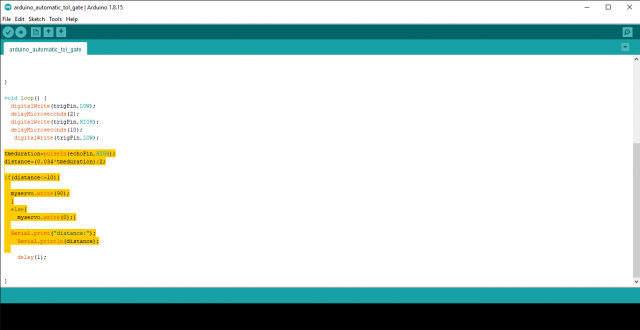
This is first part also called as declaration or defining, here we are assigning pin numbers to send/receive signals from modules.

Also at the beginning we are calling out specific modules from the library in our case servo.

You can see we are assigning pins to echo and Trig terminals and calling the time duration and distance parameters that will be used in upcoming lines which is shown in this second part.



In this part, we are giving output pin declaration in our case as we are using micro servo we are assigning pin number 9(D9) and giving command as which are the Input pins and what are the output pins for ultrasonic sensor.



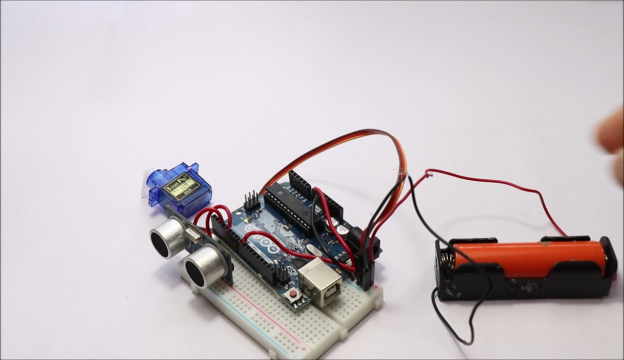
In the final part, we are setting up servo horn lifting angle and the response time, if you want any changes in the angle of rotation or delay at which the servo should work you can modify myservo.write or delay values, if you have the requirement as in this project I recommend not to make any changes

Note: we have 2 modes here, mode 1 makes the servo horn move for single angle when we bring obstacle for a moment and remove, where as in mode 2 we have 2 step rotations that can be activated by placing obstacle for more than 2 seconds in front of sensor

**Steps to upload code into Arduino Uno**

1. Connect Uno board with programming cable and connect USB side to computer.
2. Open Arduino IDE and Copy paste the above code.
3. Before clicking on upload check if proper board is selected.
4. You can click on tools then Board option later select Arduino Uno.
5. Now cross check for port number and you can select this from tools and port.
6. Once you are done with this you can hit on that upload button.
7. Once you have finished uploading successfully disconnect USB cable.
8. Now you can power Arduino Uno using external battery.
9. As I’ am using 3.7v battery I did not use any resistor, if you are using higher volt battery you may need resistor as per the voltage values.

At the very end, we have our fully working circuit ready that can be used, but using this without any additional body is not much appealing so I used a piece of cardboard and color papers to look like a road.



So we can place all the components on top of this, below is the visual of finished project that is ready to collect some funds!

I have placed all the components in such an order so that nothing comes in the way of moving vehicle/obstacle. If there are any components that comes in way for instance the servo starts to move unnecessarily, so follow this order for effective placement.

That completes our project, as usual if you have any queries let me know in the comments box.

CONCLUTION