ABSTRACT

The Toll Gate Project using Arduino is an automated system designed to facilitate the collection of toll fees in a more efficient and convenient manner. This project uses Arduino microcontroller boards, an RFID reader, and a servo motor to automate the process of toll collection.When a vehicle approaches the toll gate, its RFID tag is detected by the reader, which sends a signal to the Arduino board. The board then activates the servo motor to lift the toll gate bar, allowing the vehicle to pass through. Once the vehicle has passed through, the servo motor is then deactivated, and the toll gate bar returns to its original position.

INTRODUCTION

The Toll Gate Project using Arduino is an innovative system that aims to revolutionize the traditional methods of toll collection. With the increasing number of vehicles on the roads, toll collection has become an essential aspect of road transport infrastructure. However, the traditional manual toll collection process has several drawbacks, such as long wait times, high costs, and the potential for errors.

This project offers numerous benefits, including improved efficiency, reduced traffic congestion, and increased accuracy in toll fee collection. It also eliminates the need for toll operators, which can reduce costs and improve safety by minimizing the risk of accidents. Overall, the Toll Gate Project using Arduino is a promising solution for modernizing toll collection and improving the overall transportation experience.

PROJECT OBJECTIVES

1.**Automation**: The primary objective of this project is to automate the toll collection process using Arduino microcontroller boards, an RFID reader, and a servo motor. This automation can help reduce wait times, improve accuracy, and enhance the overall toll collection process.

2.**Efficiency**: The Toll Gate Project using Arduino aims to improve the efficiency of toll collection by reducing the time taken for the collection process. The automated system can help reduce traffic congestion, saving time for drivers and reducing travel time.

3.**Accuracy**: The project aims to improve the accuracy of toll fee collection by automating the process. With the help of an RFID reader, the system can detect the unique RFID tag attached to each vehicle, ensuring accurate toll fee collection.

4.**Cost-effectiveness**: The project aims to reduce costs associated with manual toll collection. By automating the process, it eliminates the need for toll operators and reduces the costs associated with human labor.

5.**Safety**: The Toll Gate Project using Arduino aims to improve safety by reducing the risk of accidents associated with manual toll collection. The automated system eliminates the need for toll operators, reducing the number of people involved in the process and minimizing the risk of accidents.

OVERVIEW AND BENEFITS

The Toll Gate Project using Arduino is an automated system designed to facilitate the collection of toll fees in a more efficient and convenient manner. The system utilizes an Arduino microcontroller board, an RFID reader, and a servo motor to automate the process of toll collection.

1.Improved Efficiency: The Toll Gate Project using Arduino improves the efficiency of toll collection by automating the process. This reduces wait times, saves time for drivers, and reduces travel time.

2.Increased Accuracy: The project aims to improve the accuracy of toll fee collection by automating the process. With the help of an RFID reader, the system can detect the unique RFID tag attached to each vehicle, ensuring accurate toll fee collection.

3.Reduced Costs: The automated system eliminates the need for toll operators, reducing the costs associated with human labor. This makes it a cost-effective solution for toll collection.

4.Improved Safety: The Toll Gate Project using Arduino improves safety by reducing the risk of accidents associated with manual toll collection. The automated system eliminates the need for toll operators, reducing the number of people involved in the process and minimizing the risk of accidents.

5.User-friendly: The Toll Gate Project using Arduino is easy to use and understand. It offers a user-friendly interface and can be operated by anyone, making it accessible to all drivers.

LITERATURE REVIEW

There has been a significant amount of research in recent years exploring the use of Arduino microcontroller boards in various applications, including toll gate automation. Here are some of the relevant literature on the subject"Development of Automated Toll Collection System Using RFID Technology" by N. Abbas et al. (2017): This study explored the development of an automated toll collection system using RFID technology and Arduino microcontroller boards. The authors demonstrated that the system could effectively reduce waiting time and improve the accuracy of toll collection.

"An RFID Based Automated Toll Collection System" by N. Nourian et al. (2017): This research proposed an automated toll collection system that uses RFID technology and Arduino microcontroller boards. The system was designed to reduce traffic congestion, increase efficiency, and improve the accuracy of toll fee collection.

"Design and Implementation of RFID-Based Automatic Toll Collection System Using Arduino" by S. K. Jaiswal et al. (2019): This study presented a design and implementation of an automated toll collection system using RFID technology and Arduino microcontroller boards. The authors demonstrated that the system could improve efficiency, reduce waiting time, and reduce the costs associated with manual toll collection.

"RFID Based Toll Collection System Using Arduino" by M. A. Hossain et al. (2019): This research proposed an RFID-based toll collection system that utilizes Arduino microcontroller boards. The authors demonstrated that the system could effectively reduce waiting time, improve the accuracy of toll collection, and reduce costs associated with manual toll collection.

THEORY

IOT (INTERNET OF THINGS)

IOT as a term has evolved long way as a result of convergence of multiple technologies, machine learning, embedded systems and commodity sensors. IOT is a system of interconnected devices assigned a UIDS, enabling data transfer and control of devices over a network. It reduced the necessity of actual interaction in order to control a device. IOT is an advanced automation and analytics system which exploits networking, sensing, big data, and artificial intelligence technology to deliver complete systems for a product or service. These systems allow greater transparency, control, and performance when applied to any industry or system.

Features of IOT

1. Intelligence: IOT comes with the combination of algorithms and computation, software & hardware that makes it smart. Ambient intelligence in IOT enhances its capabilities which facilitate the things to respond in an intelligent way to a particular situation and supports them in carrying out specific tasks. In spite of all the popularity of smart technologies, intelligence in IOT is only concerned as a means of interaction between devices, while user and device interaction are achieved by standard input methods and graphical user interface

2. Connectivity: Connectivity empowers the Internet of Things by bringing together everyday objects. Connectivity of these objects is pivotal because simple object level interactions contribute towards collective intelligence in the IOT network. It enables network accessibility and compatibility in the things.With this connectivity, new market opportunities for the Internet of things can be created by the networking of smart things and applications

3. Dynamic Nature: primary activity of Internet of Things is to collect data from its environment, this is achieved with the dynamic changes that take place around the devices. The state of these devices change dynamically, example sleeping and waking up, connected and/or disconnected as well as the context of devices including temperature, location and speed. In addition to the state of the device, the number of devices also changes dynamically with a person, place and time

4. Enormous Scale: The number of devices that need to be managed and that communicate with each other will be much larger than the devices connected to the current Internet. The management of data generated from these devices and their interpretation for application purposes becomes more critical. Gartner (2015) confirms the enormous scale of IOT in the estimated report where it stated that 5.5 million new things will get connected every day and 6.4 billion connected things will be in use worldwide in 2016, which is up by 30 percent from 2015. The report also forecasts that the number of connected devices will reach 20.8 billion by 2020

5. Sensing:IOT wouldn't be possible without sensors that will detect or measure any changes in the environment to generate data that can report on their status or even interact with the environment. Sensing technologies provide the means to create capabilities that reflect a true awareness of the

6. Heterogeneity: Heterogeneity in Internet of Things as one of the key characteristics. Devices in IOT are based on different hardware platforms and networks and can interact with other devices or service platforms through different networks. IOT architecture should support direct network connectivity between heterogeneous networks. The key design requirements for heterogeneous things and their environments in IOT are scalabilities, modularity, extensibility and interoperability.

7. Security : IOT devices are naturally vulnerable to security threats. As we gain efficiencies, novel experiences, and other benefits from the IOT, it would be a mistake to forget about security concerns associated with it. There is a high level of transparency and privacy issues with IOT. It is important to secure the endpoints, the networks, and the data that is transferred across all of it means creating a security paradigm.

Advantages of IOT

1. Communication: IOT encourages the communication between devices, also famously known as Machine-to-Machine (M2M) communication. Because of this, the physical devices are able to stay connected and hence the total transparency is available with lesser inefficiencies and greater quality.

2. Automation and Control : Due to physical objects getting connected and controlled digitally and centrally with wireless infrastructure, there is a large amount of automation and control in the workings. Without human intervention, the machines are able to communicate with each other leading to faster and timely output.

3. Information : It is obvious that having more information helps making better decisions. Whether it is mundane decisions as needing to know what to buy at the grocery store or if your company has enough widgets and supplies, knowledge is power and more knowledge is better.

4. Monitor : The second most obvious advantage of IOT is monitoring. Knowing the exact quantity of supplies or the air quality in your home, can further provide more information that could not have previously been collected easily. For instance, knowing that you are low on milk or printer ink could save you another trip to the store in the near future. Furthermore, monitoring the

5. Time : As hinted in the previous examples, the amount of time saved because of IOT could be quite large. And in today's modern life, we all could use more time

.

6. Automation of daily tasks leads to better monitoring of device: The IOT allows you to automate and control the tasks that are done on a daily basis, avoiding human intervention. Machine-to-machine communication helps to maintain transparency in the processes.

Disadvantages of IOT

1) **Compatibility :** Currently, there is no international standard of compatibility for the tagging and monitoring equipment. I believe this disadvantage is the most easy to overcome. The manufacturing companies of these equipment just need to agree to a standard, such as Bluetooth, USB, etc. This is nothing new or innovative needed.

2) **Complexity** ;As with all complex systems, there are more opportunities of failure. With the Internet of Things, failures could sky rocket. For instance, let's say that both you and your spouse each get a message saying that your milk has expired, and both of you stop at a store on your way home, and you both purchase milk. As a result, you and your spouse have purchased twice the amount that you both need. Or maybe a bug in the software ends up automatically ordering a new ink cartridge for your printer each and every hour for a few days, or at least after each power failure, when you only need a single replacement.

3) **Privacy/Security** : With all of this IOT data being transmitted, the risk of losing privacy increases. For instance, how well encrypted will the data be kept and transmitted with? Do you want your neighbours or employers to know what medications that you are taking or your financial situation?

4) **Safety :**Imagine if a notorious hacker changes your prescription. Or if a store automatically ships you an equivalent product that you are allergic to, or a flavour that you do not like, or a product that is already expired. As a result, safety is ultimately in the hands of the consumer to verify any and all automation.

5) **Lesser Employment of Menial Staff** : The unskilled workers and helpers may end up losing their jobs in the effect of automation of daily activities. This can lead to unemployment issues in the society. This is a problem with the advent of any technology and can be overcome with education. With daily activities getting automated, naturally, there will be fewer requirements of human resources, primarily, workers and less educated staff. This may create Unemployment

Application of IOT

1) **Wearables**: Wearable technologies is a hallmark of IOT applications and is one of the earliest industries to have deployed IOT at its services. Fit Bits, heart rate monitors, smartwatches, glucose monitoring devices reflect the successful applications of IOT.

2) **Smart homes:** This area of application concerned to this particular project, so a detailed application is discussed further. *Jarvis*, an AI home automation employed by Mark Zuckerberg, is a remarkable example in this field of application.

3) **Industrial Automation :** For a higher return of investment this field requires both fast developments and quality of products. This vitality thus coined the term IIOT. This whole schematic is re-engineered by IOT applications. Following are the domains of IOT applications in industrial automation • Factory Digitalization

- Product flow Monitoring
- Inventory Management
- Safety and Security
- Quality Control
- Packaging optimization
- Logistics and Supply Chain Optimization

4) **Government and Safety :** IOT applied to government and safety allows improved law enforcement, defence, city planning, and economic management. The technology fills in the current gaps, corrects many current flaws, and expands the reach of these efforts. For example, IOT can help city planners have a clearer view of the impact of their design, and governments have a better idea of the local economy.

IOT software

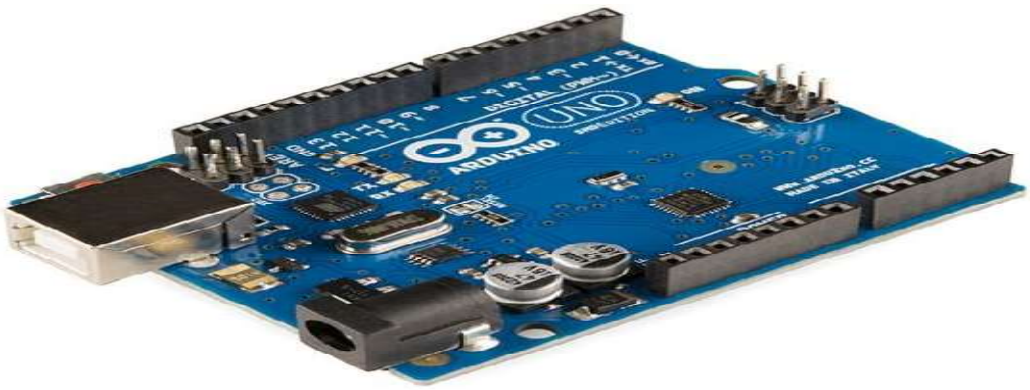
IOT software addresses its key areas of networking and action through platforms, embedded systems, partner systems, and middleware. These individual and master applications are responsible for data collection, device integration, real-time analytics, and application and

process extension within the IOT network. They exploit integration with critical business systems (e.g., ordering systems, robotics, scheduling, and more) in the execution of related tasks.

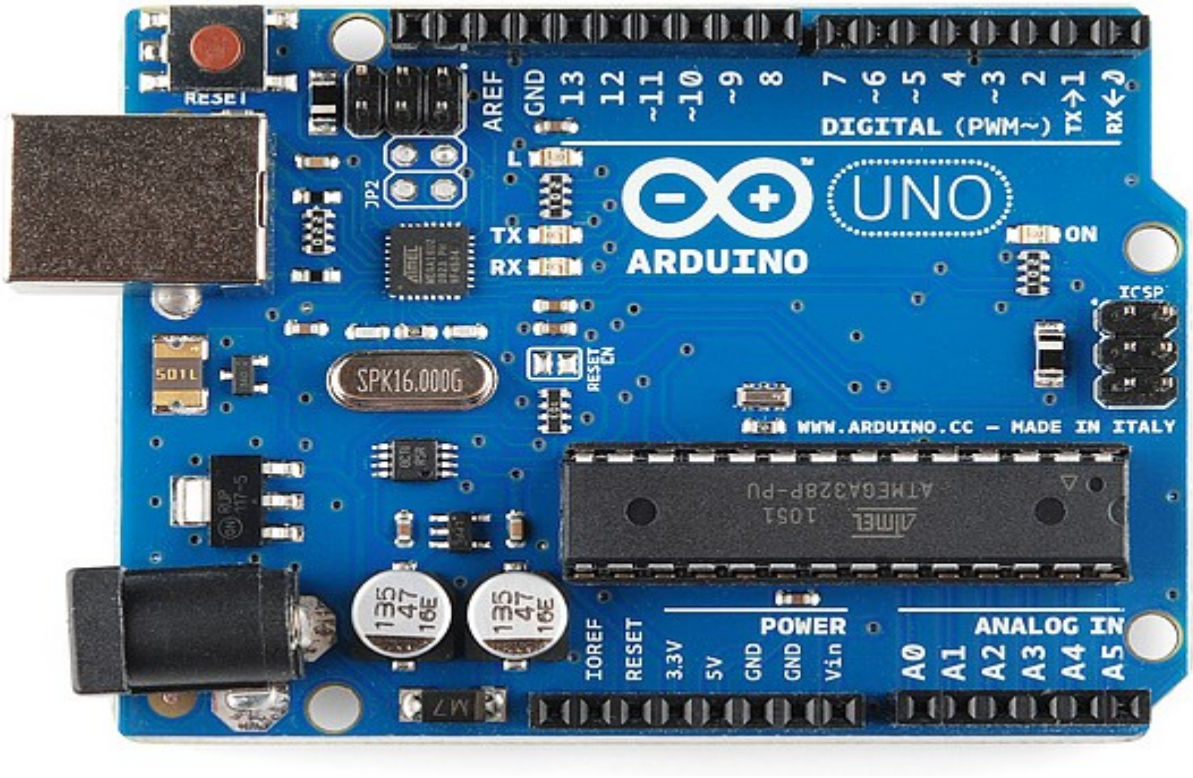
Data Collection

This software manages sensing, measurements, light data filtering, light data security, and aggregation of data. It uses certain protocols to aid sensors in connecting with real-time, machine- to-machine networks. Then it collects data from multiple devices and distributes it in accordance with settings. It also works in reverse by distributing data over devices. The system eventually transmits all collected data to a central server.

Arduino UNO



The Arduino no is an open-source microcontroller board based on the Microchip ATmega328P microcontroller and developed by Arduino.cc and initially released in 2010.[2][3] The board is equipped with sets of digital and analog input/output (I/O) pins that may be interfaced to various expansion boards (shields) and other circuits.[1] The board has 14 digital I/O pins (six capable of PWM output), 6 analog I/O pins, and is programmable with the Arduino IDE (Integrated Development Environment), via a type B USB cable.[4] It can be powered by a USB cable or a barrel connector that accepts voltages between 7 and 20 volts, such as a rectangular 9-volt battery. It is similar to the Arduino Nano and Leonardo.[5][6] The hardware



Digital I/O pins: The Arduino Uno has 14 digital I/O pins labeled from 0 to 13. These pins can be configured as either inputs or outputs using the pinMode() function. They can be used to read digital signals from sensors or to control digital components like LEDs, motors, and relays using digitalWrite() and digitalRead() functions.

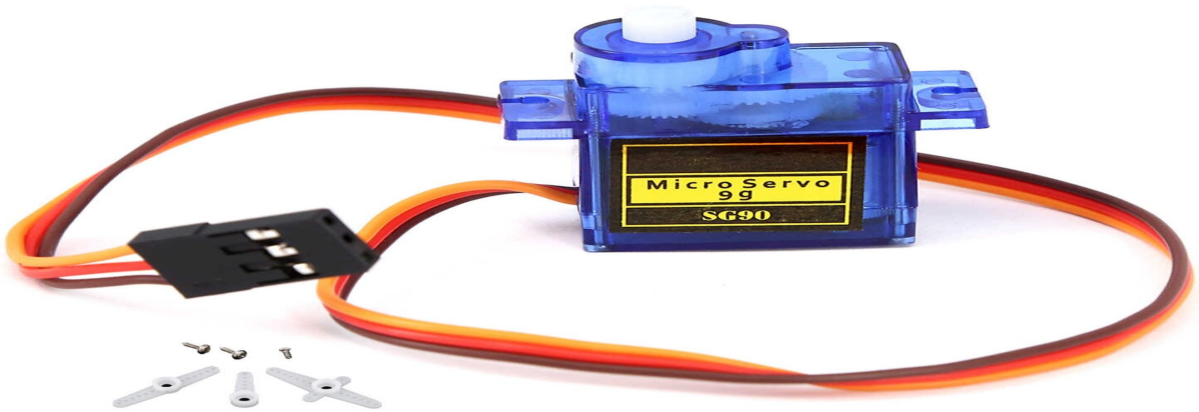
Analog input pins: The Arduino Uno has 6 analog input pins labeled A0 to A5. These pins can be used to measure analog signals ranging from 0 to 5 volts using the analogRead() function. Analog input pins can be used to interface with sensors that provide analog output like temperature sensors, light sensors, and potentiometers.

Power pins: The Arduino Uno has several power pins that provide power to the board and external components. The 5V pin can be used to power external components with a voltage of 5V, while the Vin pin can be used to power the board with a voltage ranging from 7V to 12V. The GND pin is used as the ground reference for the board and external components.

Communication pins: The Arduino Uno has several pins that are used for communication with other devices. The TX and RX pins are used for serial communication with other devices like computers or other Arduinos. The SCL and SDA pins are used for I2C communication with other devices like sensors and LCD displays. The MOSI, MISO, and SCK pins are used for SPI communication with other devices like SD cards and RF modules.

Other pins: The Arduino Uno also has a reset pin, which can be used to reset the board, and an AREF pin, which can be used to provide an external reference voltage for the analog input pins.

Plastic Geared Micro servo



Servo motors can be used in IoT (Internet of Things) applications to provide precise control of mechanical movements, such as in robotics or automation. Here are some key points about using servo motors in IoT:

- **Communication**: Servo motors can be controlled through various communication protocols such as PWM, I2C, and SPI, making them compatible with IoT devices.
- **Sensors**: Servo motors can also be integrated with sensors, such as accelerometers and gyroscopes, to provide feedback on position and movement.

- **Power**: Servo motors require a power source to operate, which can be provided through a battery or a power supply that can be integrated with IoT devices.
- **Control**: Servo motors can be controlled remotely through IoT devices such as smartphones or tablets, allowing for remote monitoring and control of the motor's position and movement.
- **Automation**: Servo motors can be used in automated systems in IoT applications, allowing for precise control of movements in real-time.
- **Integration**: Servo motors can be integrated with other IoT devices, such as sensors and actuators, to create

Ultrasonic sensor HC-SR04

Ultrasonic sensors are commonly used in IoT (Internet of Things) devices for distance measurement, object detection, and collision avoidance. These sensors work by emitting high-frequency sound waves and then measuring the time it takes for the sound waves to bounce back after hitting an object.

- Operation: Ultrasonic sensors work by emitting a high-frequency sound wave from a transducer. The sound wave travels through the air until it hits an object, at which point it bounces back to the sensor. The sensor measures the time it takes for the sound wave to return and calculates the distance between the sensor and the object based on this time.
- Range: Ultrasonic sensors typically have a range of a few centimeters up to several meters, depending on the specific sensor model. Some sensors may have a shorter range but higher accuracy, while others may have a longer range but lower accuracy.
- Accuracy: The accuracy of an ultrasonic sensor depends on various factors, including the sensor's range, the size and shape of the object being detected, and the surrounding environment. In general, ultrasonic sensors are more accurate at shorter distances.
- Application: Ultrasonic sensors can be used in a wide range of IoT applications, including parking sensors, security systems, robotics, and industrial automation. They can be used to detect the presence or absence of objects, measure distances, and even detect the level of liquids in tanks.

Installation of Arduino UNO:

1. **Download the Arduino IDE:** The first step is to download and install the Arduino Integrated Development Environment (IDE) from the official Arduino website (https://www.arduino.cc/en/software). Choose the appropriate version of the IDE for your operating system and download it.

2. **Connect the Arduino Uno**: Next, connect your Arduino Uno board to your computer using a USB cable. You should see the power LED on the board light up, indicating that the board is receiving power from your computer.

3. **Install the driver:** If you're using Windows, you'll need to install the Arduino Uno driver. The driver can be found in the "drivers" folder of the Arduino IDE installation directory. Follow the on-screen instructions to install the driver.

4. **Select the board:** Open the Arduino IDE and select the appropriate board from the "Tools" > "Board" menu. In this case, select "Arduino Uno".

5. **Select the port:** Next, select the appropriate port from the "Tools" > "Port" menu. This is the serial port that your Arduino Uno is connected to.

Proposed System :

1.Arduino board: The heart of the system, the Arduino board would be responsible for controlling and coordinating all of the system's components.

2.Ultrasonic sensor: An ultrasonic sensor would be used to detect the presence of a vehicle at the toll gate. The sensor would be mounted above the roadway and would send a signal to the Arduino when a vehicle is detected.

3.Motor and gate mechanism: The Arduino would control a motor and gate mechanism that would open and close the toll gate. The gate would remain closed until a vehicle is detected by the ultrasonic sensor, at which point the gate would open to allow the vehicle to pass.

4.Power supply: The system would require a power supply to operate, which could be provided by a battery or an external power source.

Existing System:

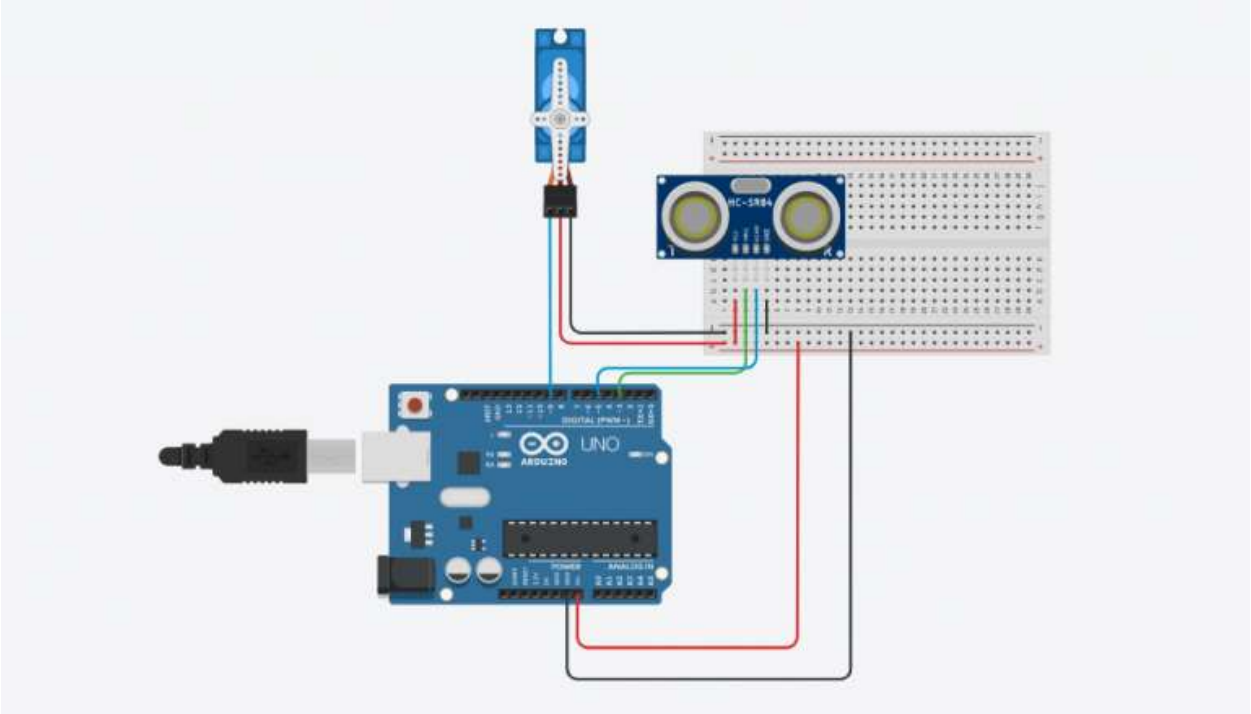
1.Arduino board: The Arduino board is the central control unit that manages all the components of the system**.**

2.Ultrasonic sensor: An ultrasonic sensor is used to detect the presence of a vehicle at the toll gate. The sensor is placed above the roadway and sends a signal to the Arduino when a vehicle is detected.

3.Motor and gate mechanism: The Arduino controls a motor and gate mechanism that opens and closes the toll gate. The gate remains closed until a vehicle is detected by the ultrasonic sensor, at which point the gate opens to allow the vehicle to pass.

4.Keypad: A keypad can be used to manually enter toll fees or access codes for authorized personnel.

CIRCUIT DIAGRAM



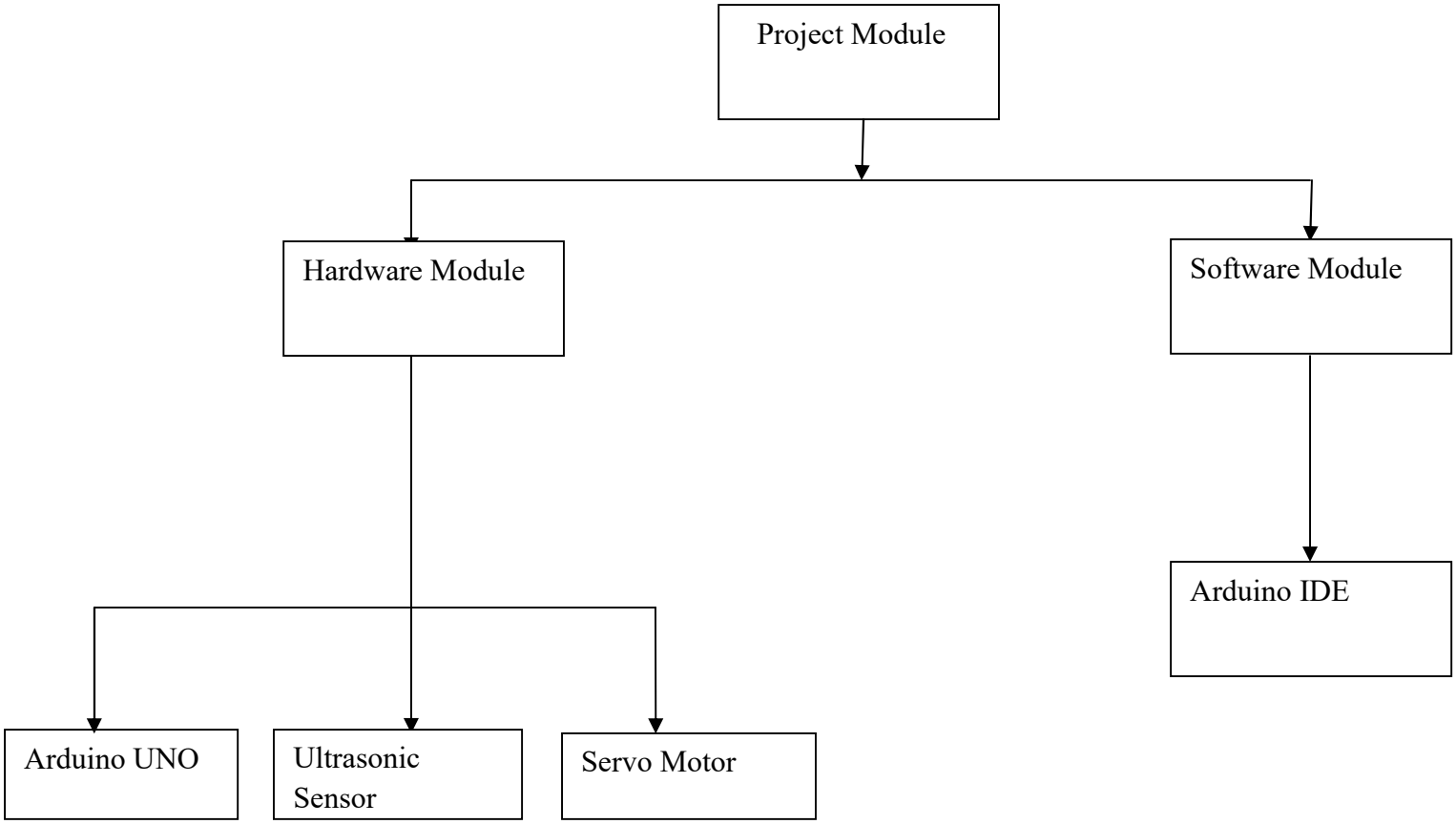
MAIN FEATURES OF THE PROTOTYPE

Arduino board: The main component of the prototype is an Arduino board, which serves as the control center for the toll gate system.

Ultrasonic sensor: An ultrasonic sensor is used to detect the presence of a vehicle approaching the toll gate. The sensor sends a signal to the Arduino when a vehicle is detected.

Motor and gate mechanism: The Arduino controls a motor and gate mechanism that opens and closes the toll gate. The gate remains closed until a vehicle is detected by the ultrasonic sensor, at which point the gate opens to allow the vehicle to pass.

PROJECT LAYOUT



Arduino Uno: Arduino Uno is a microcontroller board based on the ATmega328P microcontroller that can be used in various IoT applications. It has 14 digital input/output pins, 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, and an ICSP header.

Servo Motor: A servo motor is a type of motor that can rotate with great precision. Normally this type of motor consists of a control circuit that provides feedback on the current position of the motor shaft, this feedback allows the servo motors to rotate with great precision. If you want to rotate an object at some specific angles or distance, then you use a servo motor. It is just made up of a simple motor which runs through a servo mechanism

Ultrasonic Sensor: Ultrasonic sensors are electronic devices that calculate the target's distance by emission of ultrasonic sound waves and convert those waves into electrical signals. The speed of emitted ultrasonic waves traveling speed is faster than the audible sound.

COMPONENTS REQIRED

SL NO	COMPONENT	QUANTITY
1	Ardiuno UNO	1
2	Servo Motor	1
3	Jumper wires	10
4	Arduino cable	1

Interfacing Arduino UNO with Arduino IDE :

Download and install the Arduino IDE from the official website:
https://www.arduino.cc/en/software.

2. Connect your Arduino UNO board to your computer using a USB cable.

3. Open the Arduino IDE software.

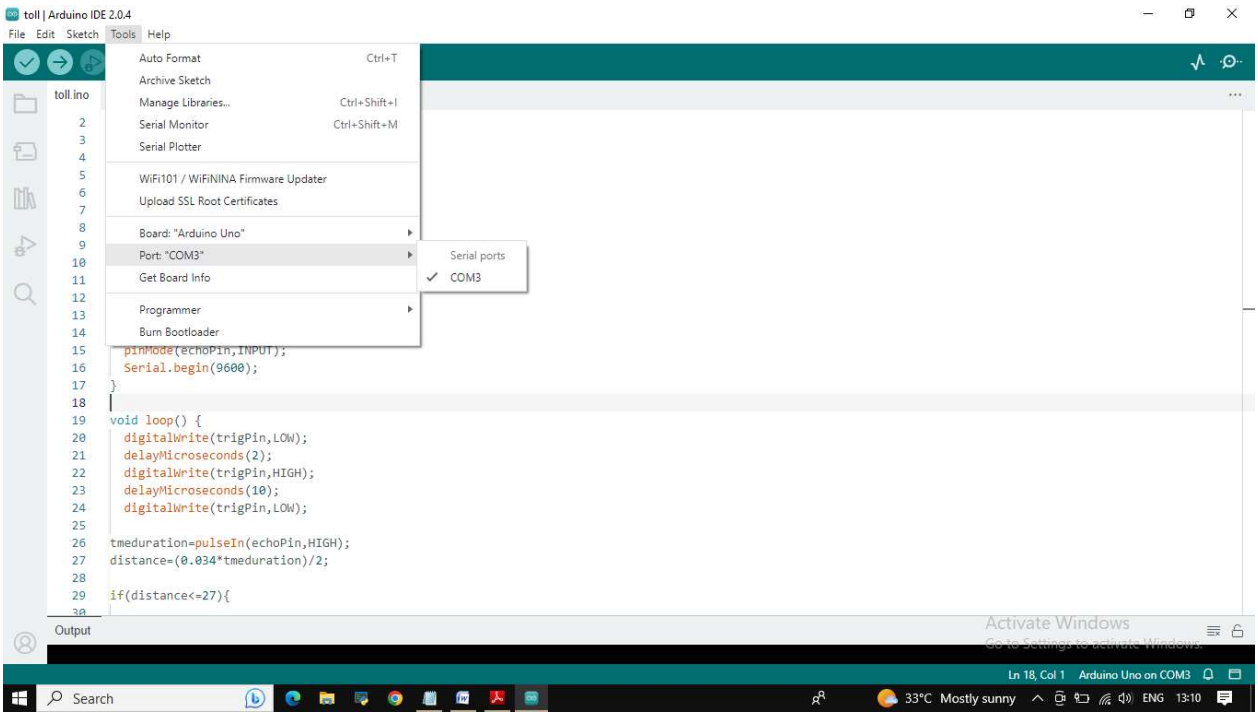
4. In the Arduino IDE, go to Tools > Board and select Arduino UNO as the board type.

5. Go to Tools > Port and select the port to which your Arduino UNO board is connected.

6. Now, you can write your Arduino code in the editor window of the Arduino IDE.

7. Once you have written your code, click on the Verify button (the checkmark icon) to check for any errors.

Configuring the COM3 Port :



Running Program in Arduino UNO:

Step 1: Files> New Sketch> program Code.

Step 2: Click Verify

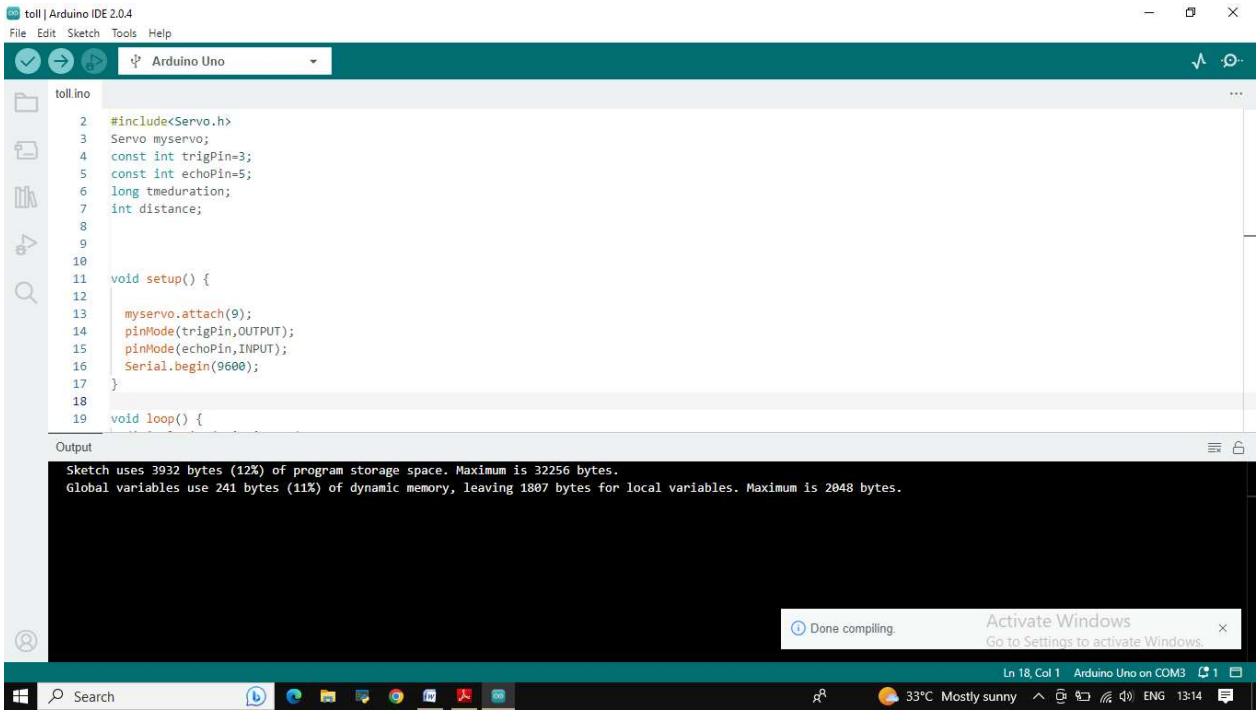
Step 3: Upload Code.

Step 4: Selecting Port

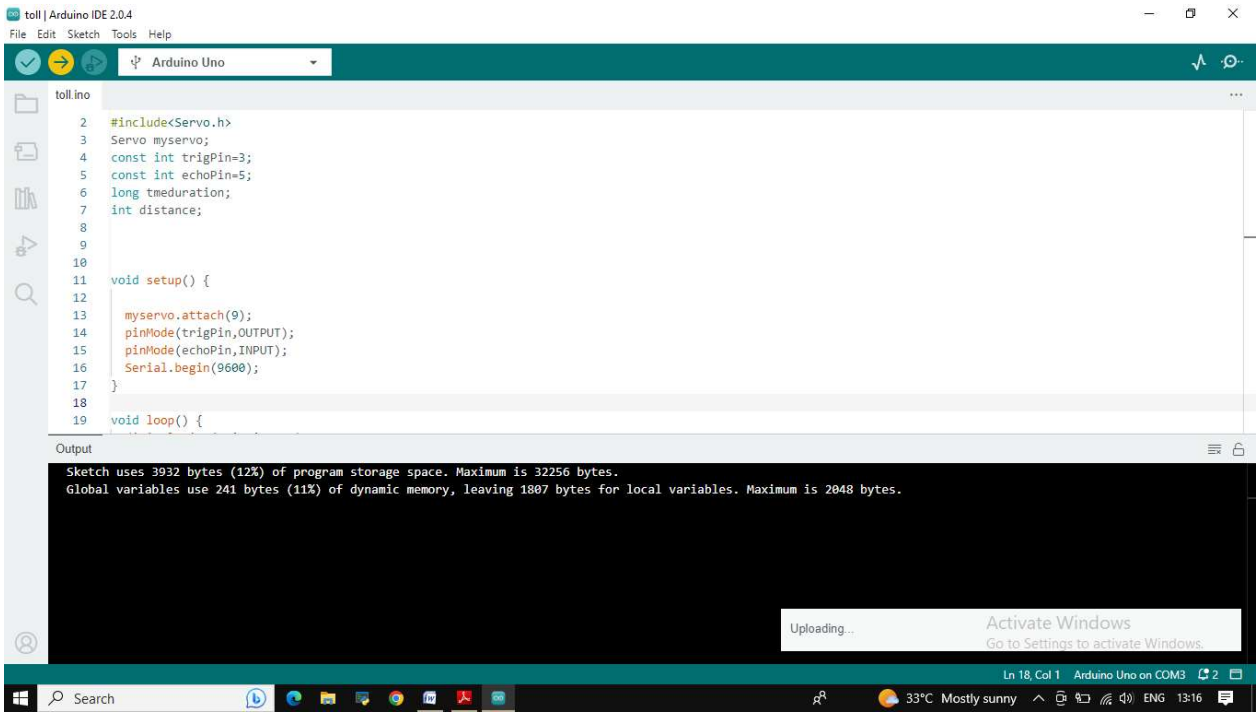
#CODING

```
// TOLL TAX BARRIER

import serial

import time

# Set the serial port and baud rate

ser = serial.Serial('COM6', 9600)

# Wait for the Arduino to initialize

time.sleep(2)

while True:

    # Read the distance from the Arduino

    distance = ser.readline().decode().strip()

    # Print the distance

    print("Distance: " + distance)
```

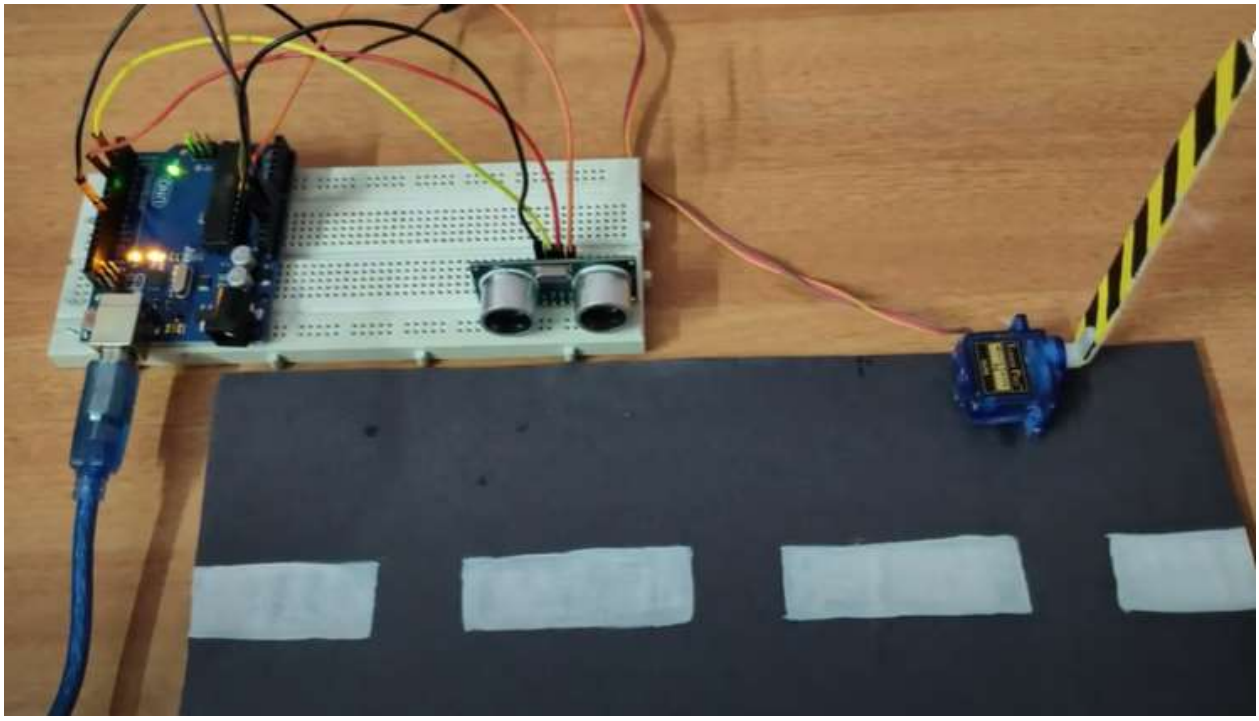
Step 2: Verifying Code:

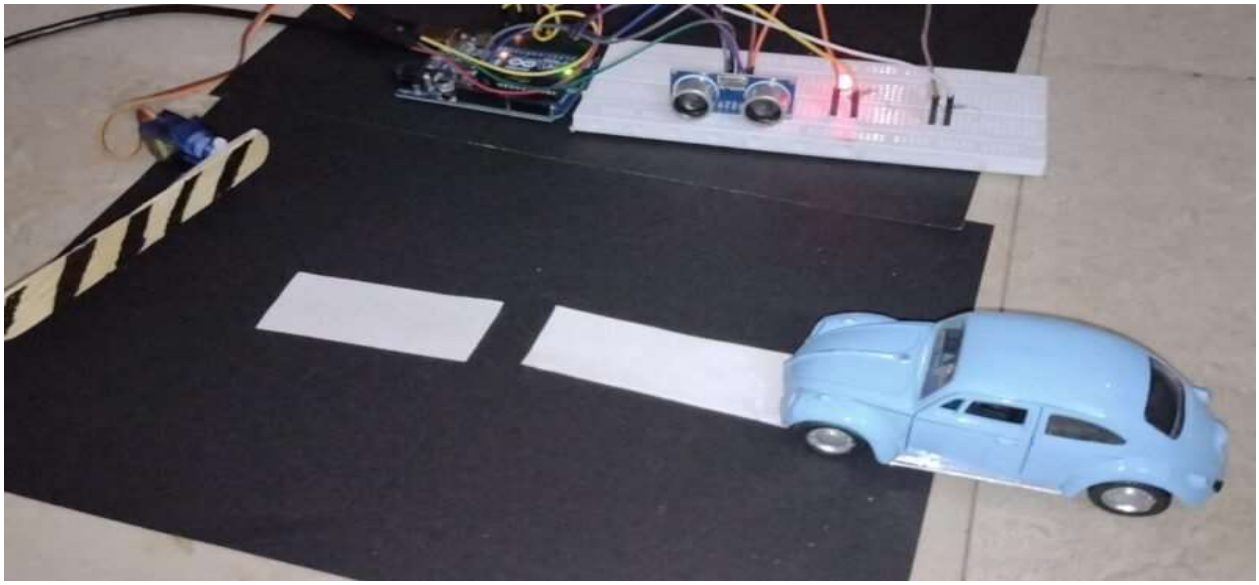


Step 3: Uploading code



Output :





Conclusion:

In conclusion, the toll gate project using Arduino has been successfully developed and implemented. The system is designed to operate automatically when a vehicle approaches the gate, and it uses sensors to detect the vehicle's presence and open the gate. The system also includes a payment module that allows users to pay their toll fees electronically.

The project's success is attributed to the use of Arduino microcontrollers, which are easy to program and integrate with various sensors and other electronic components. The system's reliability and accuracy are also ensured by the use of high-quality sensors and components.