IoT BASED AUTOMATIC VEHICLE CONTROL SYSTEM



SONARGAON UNIVERSITY (SU)

Supervised By

Md. Rais Uddin Mollah
Assistant Professor
Department of EEE

Submitted By

NAME	ID NO
S.M. ZONAID	EEE2002020056
MD. FAZLE RABBI	EEE2002020049
ARIFUR RAHMAN	EEE2002020062
MD SHAMIM AHAMED	EEE2002020013
FOYSAL HASAN	EEE1601007006

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING
SONARGAON UNIVERSITY (SU)
147/I, PANTHAPATH, DHAKA–1215, BANGLADESH

DATE OF SUBMISSION: SEPTEMBER-2023

Declaration

It is declared hereby that this project paper or any part of it has not been submitted to anywhere else for the award of any degree.

S.M. ZONAID

MD. FAZLE RABBI

ARIFUR RAHMAN

MD SHAMIM AHMED

FOYSAL HASAN

Under Supervision of

…….

Md. Rais Uddin Mollah
Assistant Professor
Department of EEE
Sonargaon University (SU)

Certification

This is to certify that this project entitled " IoT based automatic vehicle control system" is done by the following students under my direct supervision This project work has been carried out by them in the laboratories of the Department of Electrical and Electronics Engineering under the Faculty of Engineering. Sonargaon University (SU) in partial fulfillment of the requirements for the degree of Bachelor of Science in Electrical and Electronics Engineering.

Under Supervision of

…….

Md. Rais Uddin Mollah
Assistant Professor
Department of EEE
Sonargaon University (SU)

ACKNOWLEDGEMENT

The report titled as on monitoring and Alert system of IoT based vehicle automation system has been prepared to fulfill the requirement of our practicum program. In the process of doing and preparing our practicum report, we would like to pay our grade to some person for their enormous help and vast co-operation. At first, we would like to show our gratitude to the University authority to permit us to do our practicum. Specially, we would like to thank to our honorable supervisor Md. Rais Uddin Mollah, Assistant Professor Department of Electrical & Electronics Engineering Sonargaon University. Dhaka for his valuable and patient advice, sympathetic assistance, co-operation. contribution of new Deep theoretical and hardware knowledge & keen interest of our supervisor in this field influenced us to carry out this project. His endless patience. scholarly guidance, continual encouragement, constant and energetic supervision. constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

Finally, we would like to thanks again to the respected Vice-Chancellor of SU, also thanks to Head of Department of SU Electrical & Electronics Engineering, because they are designated such an environment for learning through which weight the opportunity to acquire knowledge under him in EEE program, and that will be very helpful for our prospective career

We are, indeed, grateful to all those from whom we got sincere cooperation and help for the preparation of this report.

Abstract

The project titled "IoT-Based Automatic Vehicle Control system," is a cutting-edge endeavor that seamlessly integrates modern technology to enhance the safety and control of automobiles. At its heart, this project utilizes a card-based ignition system, a mobile application interface, obstacle avoidance sensors, and fire detection sensors to create a sophisticated automotive experience. The card-based ignition system adds an extra layer of security and user authorization to vehicle start-up. This feature ensures that only authorized users with the corresponding card can initiate the vehicle, enhancing theft prevention and access control. Moreover, the mobile application acts as a powerful control center, allowing users to regulate the car's maximum speed remotely. This real-time speed control feature enhances safety by preventing over-speeding, making it particularly valuable for parents or fleet managers. The integration of obstacle avoidance sensors further elevates the safety quotient. These sensors constantly scan the surroundings and provide alerts to the driver via the mobile app, helping to avert potential collisions and accidents. In addition to safety, the project incorporates fire detection sensors to promptly identify and respond to fire incidents within the vehicle. This feature is invaluable in mitigating fire-related risks and ensuring passenger safety. The culmination of these components results in an intelligent and secure vehicle automation system. Users can monitor critical data, such as speed, obstacle detection, and fire status, through the mobile app's intuitive interface, providing peace of mind and control while on the road. "IoT-Based Vehicle Automation" redefines the driving experience, prioritizing safety, security, and real-time control through the seamless integration of Internet of Things (IoT) technologies.

TABLE OF CONTENTS

Topics		Page No
FRONT PAGE		**i**
DICLARATION		ii
CONTENT		iii
ACKNOWLEDGEMENT		iv
ABSTRACT		v
TABL OF CONTENTS		vi
CHAPTER-1: INTRODUCTION		
1.1	Introduction of the system	1
1.2	Background of study	3
1.3	Objectives of Project	4
1.4	Project outline	4
CHAPTER-2: PROJECT BACKGROUND		
2.1	Introduction	5
2.2	Literature survey	6
2.2.1	Internet of Things-concept & definition	7
2.2.2	Internet of Things-concept & definition	9
2.2.3	IoT enabling technologies	11
2.2.4	Required Equipment	14
2.2.5	Micro controller	14
2.2.6	Flame Sensor	15
2.2.7	Potentiometer	16
2.2.8	Capacitor	16
2.2.9	Printed circuit Board (PCB)	17
2.2.10	LIQUID-CRYSTAL DISPLAY(LCD-16X2)	19
2.2.11	Switched-Mode power supply (SMPS)	22
2.2.12	Obstacle sensor (IR)	24
2.3	Conclusion	25

CHAPTER-3: HARDWARE IMPLEMENTATION OF THE PROPOSED PROJECT		
3.1	Introduction	26
3.2	Working Process	26
3.3	Block Diagram of the Proposed Project	27
3.4	Flow Chart of the Proposed Project	28
3.5	Circuit Diagram of the Proposed Project	29
3.6	Final View of the Project	30
CHAPTER-4: RESULT AND DISCUSSION		
4.1	Introduction	31
4.2	Result	32
4.3	Future Scope	33
4.4	Discussion	34
CHAPTER-5: CONCLUSION & RECOMMENDATION		
5.1	Introduction	35
5.2	Advantages	36
5.3	Future Modification	38
5.4	Conclusion	39
	REFERENCE:	40
	Appendix:	43

CHAPTER-1

INTRODUCTION

1.1 Introduction to the system.

Smart technologies such as Artificial Intelligence, Internet of Things, Machine Learning, Augmented Reality and Virtual Reality have revolutionized manufacturing and engineering sectors [1]. Smart technologies enable us to maintain trust among the business entities by increase the transparency of the data [2]. With the use of smart technologies like block chain and the internet of things, the process of communication and transparency can be enhanced [3]. Tourism is the fastest-growing industry. In smart cities, migrants hire a taxi or driver less car to visit different places in the city. Almost all leading rental service providers are using mobile applications from the booking of the car to the payment. In order to make these facilities more efficient and increase the trust among both parties, various researchers have come up with different solutions based on the Internet of Things

(IoT), roadside unit, Vehicular Adhoc Network (VANET) and Radio Frequency Identification (RFID) tag [4]. Nowadays, fifth-generation (5G)–based Internet of things (IoTs) is gaining fame due to its capacity to vastly incorporate communication at mobile systems for massive multiple inputs and multiple outputs. The mobile communication using 5G technology will make a massive multiple data flow to work parallel in producing real-time analytic and decisions. Whereas, the IoT provides services for linking physical objects, buildings, vehicle and electronic components, software, sensors, actuators, and other integrated items. The linking equipment within the city is via a network connection and data exchange that allows those objects or things to gather and exchange data. The idea of IoT is not new, as initial became

famous in 1999. Furthermore, it goes through great revolution, as since 2016, the visualization of Internet objects has evolved because of the convergence of various network technologies. Network technologies involve communications as wireless everywhere, analysis of real-time devices, commodity sensors, machine learning, and integrated framework.

In IoT, interaction between the physical and digital worlds using sensors and actuators are carried out. A sensor or a network of sensors are used to sense the physical parameters or the respective environment. These processed sensor outputs are then sent to the main server or cloud with the help of various network devices. The data can be accessed over internet from anywhere around the world. Monitoring and controlling form the basic objective of

IoT technology. Hence IoT based monitoring is preferred more than manual monitoring. For the most part, the vehicle leaves the assembly line with much of the capability that is required to gather data for an IoT solution. Each sensor, module and control system is able to report its status via an on-board connector, such as the ODBII connector on most passenger vehicles. The first addition to enable communication is to be able to take the data that is produced and bring it into your IoT application. In some cases, vehicle manufacturers have installed cellular components that allow this data to be sent without the need for a cellular gateway to be added. In most cases, a cellular gateway connects to the vehicles diagnostic port to gather data. In some cases, a user/manufacturer may wish to gather other data, such as the opening of a door or the temperature of a storage area. In those scenarios, sensors are used to provide that data in real-time. Here we proposed a system which is a real

time monitoring and control of vehicle parameters such as speed, speed limit, obstacle and fire detection.

1.2 Background of Study:

The evolution of transportation technology has been a driving force in shaping the modern world. As society continues to rely heavily on vehicular transportation, it becomes imperative to continually innovate in order to address critical challenges related to safety, efficiency, and sustainability.

In recent years, the proliferation of the Internet of Things (IoT) has opened up unprecedented possibilities in the realm of automotive technology. IoT, with its interconnected network of devices and sensors, has the potential to revolutionize the way vehicles are controlled, monitored, and interacted with.

The IoT-based vehicle control system project emerges against the backdrop of this transformative landscape. Traditional vehicle control systems are being augmented and, in some cases, replaced by smart, connected technologies. The project is driven by the recognition that the future of transportation lies in integrating IoT capabilities into vehicles, paving the way for safer, more intelligent, and environmentally responsible mobility.

This background underscores the urgency and relevance of the project, as it seeks to harness the power of IoT to address pressing issues such as road safety, personalized vehicle experiences, and efficient traffic management. By exploring the integration of IoT-based components like the Smart Card System, Automatic Braking System, Fire Detection and Alarm System, and Software-Based Speed Control System, this study aims to contribute to the ongoing transformation of the automotive industry and the broader field of transportation.

1.3 Objectives of Project

- To use smart card for startup.
- To enable speed monitoring.
- To enable Real Time obstacle Monitoring.
- To enable Fire Detection System with Alarm.
- To enable setting a speed limit to avoid accidents.

- To enable Mobile Apps based Monitoring System.

1.4 Project Outline

Chapter 1: Contains overview, objectives of the project.

Chapter 2: Discuss about background and equipment of the project.

Chapter 3: Discuss about hardware implementation and experimental results of the project.

Chapter 4: Discussion and conclusion of the project.

CHAPTER-2

PROJECT BACKGROUND

2.1 Introduction

In this modern era, so much equipment is discovered to get different information. In this project, we used several types of equipment, each equipment having a different function and figure. In this chapter, the different functions of this equipment are described with figures.

2.2 Literature Survey

As a vital issue, various study has been proposed earlier. Such as, a paper titled "Modeling Quality of IoT Experience in Autonomous Vehicles" [1] introduce the term Quality of IoT experience (QoIoT) within the context of AV, where the quality evaluation, besides end users, considers quantifying the perspectives of intelligent machines with objective metrics. Hence, they propose a novel architecture that considers Quality of Data (QoD), Quality of Network (QoN), and Quality of Context (QoC) to determine the overall QoIoT in the context of AVs. Finally, we present a case study to illustrate the use of QoIoT. Then

X. Krasniqi and E. Hajrizi [2] examines the market and technical trends towards Autonomous Vehicles, evolution stages from early cars to fully autonomous, the importance of IoT in driving this industry ecosystem, advantages and disadvantages of Autonomous Vehicles, key issues and challenges faced by the industry, standards activities around this industry and finally the deployment use cases. The focus of this paper is more based on an industrial push to identify issues and challenges of Autonomous Vehicles and less on any academic research activity.

A study [15] on a smart traffic control scenario-a real-time IoT application, where a group of autonomous vehicles independently decide on their lane velocities, in collaboration with road-side units to efficiently utilize intersections with minimal environmental impact. They decompose this problem as an unconstrained network utility maximization problem. A consensus-based, constant step-size gradient descent algorithm is proposed to obtain a near-optimal solution. Then they analyze the delay-accuracy tradeoff in reaching a near optimal velocity.

Then [3] implements(THMS) Transformer health monitoring system using GSM module and single chip microcontroller which installed at distribution transformer and output of component are stored in memory for further requirement and they also applied protection system which followed predefined instruction in case of abnormality. To check for abnormal situations, the system is configured with specific predetermined instructions. If there is a problem with the system, the GSM module will send SMS (Short Message Service) messages to specified mobile phones with information about the problem, as per the predefined instructions. This transportable technology will assist utilities in making the best use of transformers and detecting problems before they become disastrous. By reducing human dependency, this technology will be a step forward in automation. Because it is a wireless communication system, there is no need for expensive big cables. As a result, the THMS provides a better service.

Subsequently, the goal of the paper [4] is to build and implement a mobile embedded system that can monitor and record critical distribution transformer data such as load currents, oil level, and ambient temperature. The idea behind the online monitoring system is to combine a GSM modem with a standalone Arduino and several sensors. It is located at the distribution transformer location, and the above parameters are

captured using the embedded system's analog to digital converter (ADC). The collected data is processed and saved in the system memory. If an abnormality or an emergency scenario happens, the system sends SMS (short messaging service) messages to mobile phones with information about the abnormality, based on predefined instructions encoded in the microcontroller.

Paper [16] examine how the latest approaches in AI and IoT can assist in the search for the Autonomous Vehicles. It has been shown that human errors are the source of 90% of automotive crashes, and the safest drivers drive ten times better than the average (Wu et al. Accident Analysis and Prevention, 117, 21–31, 2018). The automated vehicle safety is significant, and users are requiring 1000 times smaller acceptable risk level. Some of the incredible benefits of AVs are: (1) increasing vehicle safety, (2) reduction of accidents, (3) reduction of fuel consumption, (4) releasing of driver time and business opportunities, (5) new potential market opportunities, and (6) reduced emissions and dust particles. However, AVs must use large-scale data/information from their sensors and devices. Besides that, there are quite a few other studies that have been done earlier in this scope.

2.2.1 IoT CONCEPT & DEFINITION:

Internet of Things (IoT) [5], [6] consists of two Words-Internet and Things, the term "Things" in loT refers to various loT devices having unique identities and having capabilities to perform remote sensing, actuating, and live to monitor certain sorts of data. IoT devices are also enabled to have a live exchange of data with other connected devices and applications either directly or indirectly, or collect data from other devices and process the data and send the data to various servers, the other term "Internet" is defined as Global Communication network connecting millions of

computers across the planet enabling sharing of information. As forecasted by various researchers, 50 billion devices based on loT would be connected all across the planet by the year 2020.

The Internet of Things (IoT) has been defined as a dynamic global network infrastructure with self-configuring capabilities based on standard and interoperable communication protocols where physical and virtual “Things" have identities, physical attributes, and virtual personalities and use intelligent interfaces and are seamlessly integrated into the information. network, often communicate data associated with users and their environments", An ideal IoT device consists of various interfaces for making connectivity to other devices which can either be wired or wireless.

Any loT based device consists of the following components; I/O interface for Sensors, Interface for connecting to the Internet, Interface for Memory and Storage, and Interface for Audio/Video, IoT devices can be of various forms like wearable sensors, smartwatches, IoT smart home monitoring, IoT intelligent transport systems, IoT smart health devices etc [7]–[9].

2.2.2 IoT ENABLING TECHNOLOGIES:

Internet of Things has a strong backbone of various enabling technologies. Wireless Sensor Networks, Cloud Computing, Big Data, Embedded Systems, Security Protocols and Architectures, Protocols enabling communication, web services, Internet and Search Engines [6], [10]. Wireless Sensor Network (WSN): It consists of various sensors/nodes which are integrated to monitor various sorts of data, Cloud Computing: Cloud Computing, also known as on-demand computing, is a type of Internet-based computing that provides shared processing resources and data to computers and other devices on demand. It is a sort of computer system which consists of both hardware and software to perform specific tasks. It includes a

microprocessor/microcontroller, RAM/ROM, networking components, 1/O units, and storage devices.

2.3 Required Equipment

2.2.3 Microcontroller- Esp32

ESP32 [11, p. 32] is powerful SoC (System on Chip) microcontroller with integrated WiFi 802.11 b/g/n, dual mode Bluetooth version 4.2 and variety of peripherals shown in fig 2.1. It is an advanced successor of the 8266 chip primarily in the implementation of two cores clocked in different version up to 240 MHz. Compared to its predecessor, except these features, it also extends the number of GPIO pins from 17 to 36, the number of PWM channels per 16 and is equipped with 4MB of flash memory.

The ESP32 chip has been developed by the Express if Systems company, which currently offers several ESP32 versions of the SoC in the form of ESP32 Developer Kit, the ESP32 Wrover Kit, which also includes an SD card and 3.2” LCD display and last but not least the ESP32 Azure IoT kit with USB Bridge and other built-in sensors. In addition to Express if Systems, other producers are devoted to these chips - SparkFun with ESP32 Thing DB, WeMoS with its TTGO, D1, Lolin32 and Lolin D32, Adafruid (with Huzzah32), DF Robot (ESP32 Fire Beeatle) and many other manufacturers sometimes offer good and sometimes bad clones.

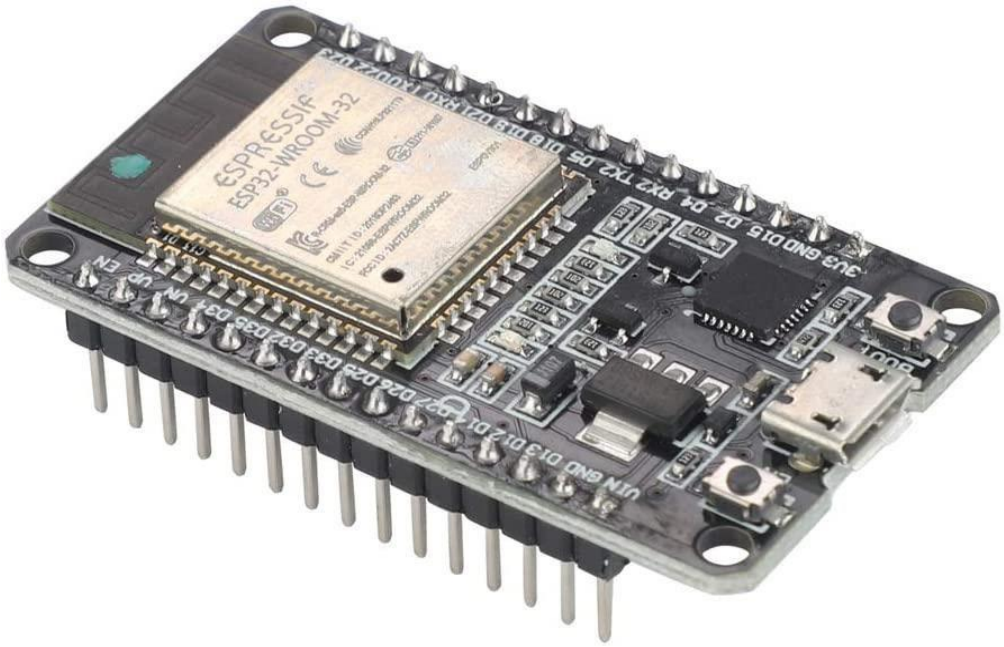


Fig 2.1: ESP32 microcontroller

ESP32 includes two core (Xtensa LX6 processor made with 40 nm technology). CPU cores can be individually controlled. There is 520 KB of on-chip SRAM for data and instructions available. Some SoC modules such as ESP32-Wrover features 4 MB of external SPI flash and an additional 8 MB of SPI PSRAM (Pseudo static RAM). We have the possibility to use SPI, I2S, I2C, CAN, UART, Ethernet MAC, and IR in various quantities, depending on the type of the board. Standard equipment also includes Hall Effect sensor, temperature sensor and touch sensor, other built-in sensors are implemented in Azure IoT and

Developer kit. SoC also provides Cryptographic hardware acceleration: AES, SHA-2, RSA, and Elliptic Curve Cryptography (ECC) and random number generator (RNG).

Figure 1 shows the ESP32 microcontroller variants with which we are working at the Department of Control System and Instrumentation.

ESP32 boards are produced in prototype designs that can be used in smart home applications, automation, wearables, audio applications, cloud-based IoT applications,

and more. It is possible to choose a specific development kit or to design a custom embedded system built on the ESP32 microcontroller.

2.2.4 Flame Sensor:

A flame sensor is an essential component in fire detection and safety systems. It works in conjunction with an Arduino micro-controller to detect the presence of a flame or fire. This sensor plays a crucial role in preventing and mitigating potential fire hazards by providing an early warning system. The flame sensor operates on the principle of infrared radiation detection. It consists of an infrared-sensitive photo diode and a specialized optical filter that allows it to selectively detect flame signatures. When exposed to a flame, the sensor detects the infrared radiation emitted by the fire and generates an electrical signal. The Arduino micro-controller acts as the brain of the flame sensor system. It receives the electrical signal from the flame sensor and processes it to determine the presence or absence of a flame. Arduino boards are popular among hobbyists and professionals due to their versatility and ease of use. They provide a user-friendly platform for connecting and controlling various electronic components. To utilize the flame sensor with Arduino, a simple circuit needs to be set up. The flame sensor is connected to the appropriate pins of the Arduino board, ensuring the correct wiring and connections. The Arduino code is then programmed to read the signal from the flame sensor and trigger appropriate actions based on the detected flame. The Arduino code for the flame sensor typically involves configuring the necessary input/output pins and implementing the logic for flame detection.

When a flame is detected, the Arduino can activate various output devices such as alarms, LED indicators, or relay modules to trigger an emergency shutdown or fire

suppression system. Additionally, the Arduino can communicate with other devices or send

notifications via Wi-Fi or GSM modules to alert users or authorities about the fire incident. The flame sensor's sensitivity can be adjusted to optimize its performance for different fire detection scenarios. This allows users to set the detection threshold based on their specific requirements. For instance, in industrial settings, a higher sensitivity level may be desired to detect even small flames, while in residential applications, a moderate sensitivity level may suffice. One of the advantages of using a flame sensor with Arduino is the ability to integrate it into larger automation and control systems.

Arduino’s flexibility allows for seamless integration with other sensors, actuators, and communication modules. This enables the creation of intelligent fire detection and suppression systems that can automatically trigger actions based on the detected flame.

Flame-sensors are classified into four types

- IR single frequency
- IR multi-spectrum
- UV flame detectors
- UV/ IR flame detectors

The flame sensor and Arduino combination finds applications in a wide range of industries, including residential, commercial, and industrial sectors. It is commonly used in fire alarm systems, flame-powered appliance safety systems, industrial furnaces, and even robotics. The compact size and low power consumption of Arduino make it suitable for portable and battery-operated fire safety devices as well. In conclusion, the flame sensor that works with Arduino provides an efficient and

cost-effective solution for fire detection and safety applications. By combining the capabilities of the flame sensor with the versatility of

Arduino, users can create robust fire detection systems that help protect lives and property. Whether it is for residential, commercial, or industrial use, the flame sensor and Arduino combination offers a reliable and customizable solution for fire detection and prevention.



Fig 2.2: Flame Sensor

- Output Channel: 1
- Power Supply: 3.3/5V
- Built in a potentiometer for sensitivity control.
- Onboard signal output indication, output effective signal is high, at the same time the indicator light up, the output signal can directly connect to microcontroller IO.
- Detection angle about 60 degrees, the flame spectrum especially sensitive.
- The flame of the most sensitive sensors flame, the regular light is also a response, generally used for fire alarm purposes.

General Specification	
Output Channel	**1**
Operating Voltage (VDC)	**3.3 ~ 5**
Mounting Hole(mm)	**3**
Length (mm)	**32**
Width (mm)	**14**
Weight (gm)	**3**
Shipment Weight	**0.005 kg**
Shipment Dimensions	**4 × 2.5 × 2 cm**

Table: 2.3: Specification of Flame sensor

Configuration:

- Pin1 (VCC pin): Voltage supply rages from 3.3V to 5.3V
- Pin2 (GND): This is a ground pin
- Pin3 (AOUT): This is an analog output pin (MCU.IO)
- Pin4 (DOUT): This is a digital output pin (MCU.IO)

2.2.5 Potentiometer:

A potentiometer is a variable resistor that is commonly used in electronic circuits to control voltage levels or adjust the resistance in a circuit. When paired with an Arduino microcontroller, a potentiometer becomes a valuable input device that allows users to interact with the system by adjusting its analog value. The basic construction of a potentiometer consists of a resistive element with a sliding contact, or wiper, that moves along the resistor track. By adjusting the position of the wiper, the resistance value between the wiper and the ends of the resistor track can be varied. When using a potentiometer with an Arduino, the potentiometer is connected to the analog input pins of the Arduino board. The Arduino's analog-to-digital converter (ADC) reads the

voltage level provided by the potentiometer and converts it into a digital value that can be processed by the microcontroller. The Arduino code for reading the potentiometer value typically involves configuring the appropriate pins as analog inputs and using the analog Read() function to obtain the digital value representing the potentiometer position. This digital value can then be used for various purposes within the Arduino program.

One common application of a potentiometer with Arduino is to control the brightness of an LED. By connecting the potentiometer to an analog input pin and an LED to a digital output pin of the Arduino, the user can adjust the resistance of the potentiometer to vary the LED's brightness. As the potentiometer's wiper position changes, the voltage level at the analog input pin changes, resulting in different analog values read by the Arduino. The Arduino code can map these values to appropriate brightness levels for the LED, providing a user-controlled lighting effect.

Another use of a potentiometer with Arduino is in analog input-based systems. The potentiometer can act as a user interface element, allowing users to control various parameters of a system. For example, in a temperature control system, the potentiometer can be used to adjust the desired temperature set point. The Arduino reads the potentiometer value and adjusts the system's behavior accordingly. Potentiometers can also be used in robotic applications.

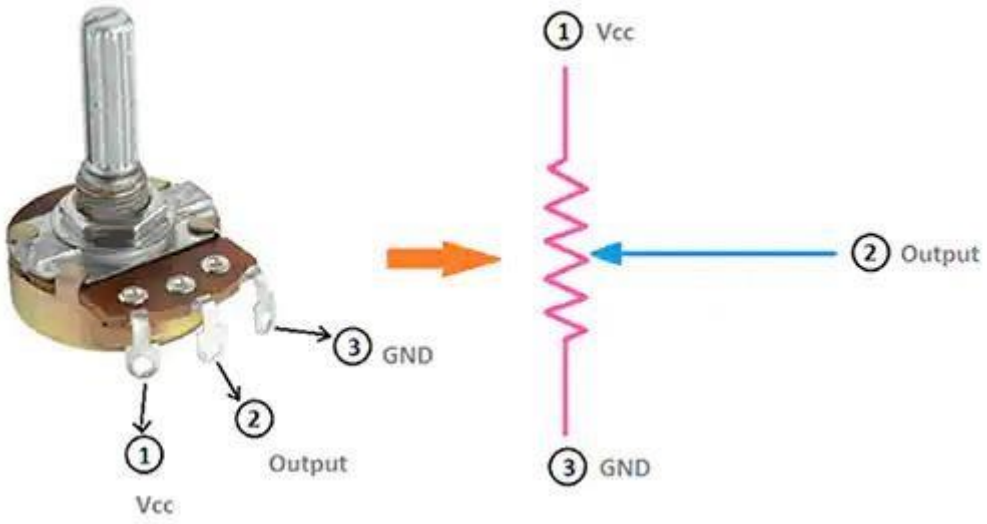


Fig 2.3 Potentiometer

By connecting a potentiometer to a servo motor and an Arduino, the potentiometer can serve as a position control input for the servo motor. As the user adjusts the potentiometer, the Arduino reads the corresponding analog value and positions the

servo motor accordingly, allowing precise control over the motor's position. Furthermore, potentiometers can be integrated into Arduino-based data acquisition systems. For instance, by connecting a potentiometer to an Arduino along with other sensors such as temperature or light sensors, users can collect data from multiple sources simultaneously. This data can then be processed, logged, or transmitted to external devices for further analysis. In summary, the combination of a potentiometer and Arduino provides a versatile means of user input and control in electronic systems. Whether used for adjusting analog values, controlling LED brightness, or interfacing with other components, the potentiometer offers a straightforward and intuitive way for users to interact with Arduino based projects. Its simplicity, affordability, and broad range of applications make the potentiometer a valuable component in the world of Arduino-based electronics.

2.2.8 Capacitor

A capacitor is a device that stores electrical energy in an electric field. It is a passive electronic component with two terminals. The effect of a capacitor is known as capacitance. While some capacitance exists between any two electrical conductors in proximity in a circuit, a capacitor is a component designed to add capacitance to a circuit. The capacitor was originally known as a condenser or condensate.

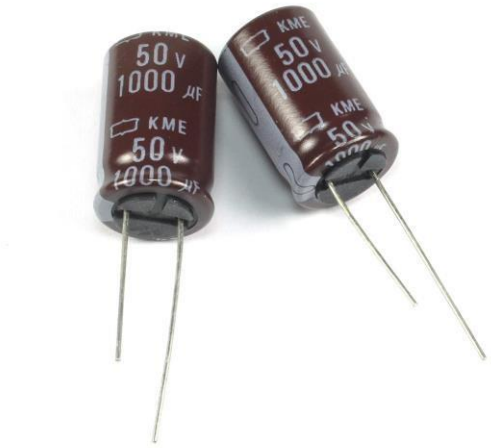


Fig 2.4.1: Capacitor 1000uF/50v

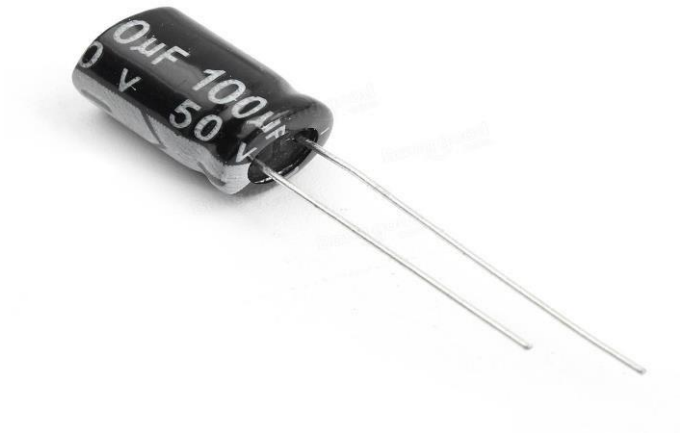


Fig 2.4.2: Capacior 100uF/50v

2.2.9 Printed Circuit Board (PCB):

A printed circuit board (PCB) mechanically supports and electrically connects electronic or electrical components using conductive tracks, pads, and other features etched from one or more sheet layers of copper laminated onto and between sheet layers of a non-conductive substrate. Components are generally soldered onto the PCB to both electrically connect and mechanically fasten them to it. Printed circuit boards are used in all but the most straightforward electronic products. They are also used in some electrical products, such as passive switch boxes. Alternatives to PCBs include wire wrap and point-to-point construction, both once famous but now rarely used. PCBs require additional design effort to lay out the circuit, but manufacturing and assembly can be automated. Specialized CAD software is available to do much of the work of layout. Mass-producing circuits with PCBs is cheaper and faster than with other wiring methods, as components are mounted and wired in one operation. Large numbers of PCBs can be fabricated at the same time, and the layout only has to be done once. PCBs can also be made manually in small quantities, with reduced benefits. PCBs can be single-sided (one copper layer), double-sided (two copper layers on both sides of one substrate layer), or multilayer (outer and inner layers of

copper, alternating with layers of the substrate). Multi-layer PCBs allow for much higher component density because circuit traces on the inner layers would otherwise take up surface space between components. The rise in popularity of multilayer PCBs with more than two, and especially with more than four, copper planes was concurrent with the adoption of surface mount technology. However, multilayer PCBs make the repair, analysis, and field modification of circuits more difficult and usually impractical. Due to the lack of proper support and time we had to use ready-made type PCB known as vero board, shown in fig-5.

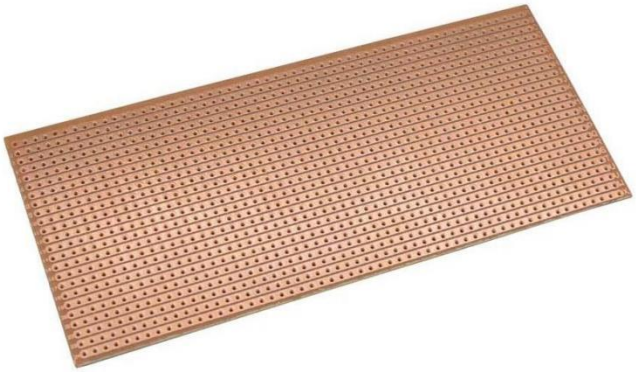


Fig 2.5: Printed Circuit Board (PCB)

2.2.10 LIQUID-CRYSTAL DISPLAY(LCD-16X2):

The term LCD stands for liquid crystal display. It is one kind of electronic display module used in an extensive range of applications like various circuits & devices like mobile phones, calculators, computers, TV sets, etc. These displays are mainly preferred for multi segment light-emitting diodes and seven segments. The main benefits of using this module are inexpensive; simply programmable, animations, and there are no limitations for displaying custom characters, special and even animations, etc.

The 16×2 LCD pinout is shown below.

- Pin1 (Ground/Source Pin): This is a GND pin of display, used to connect the GND terminal of the microcontroller unit or power source.
- Pin2 (VCC/Source Pin): This is the voltage supply pin of the display, used to connect the supply pin of the power source.
- Pin3 (V0/VEE/Control Pin): This pin regulates the difference of the display, used to connect a changeable POT that can supply 0 to 5V.
- Pin4 (Register Select/Control Pin): This pin toggles among command or data register, used to connect a microcontroller unit pin and obtains either 0 or 1(0 = data mode, and 1 = command mode).
- Pin5 (Read/Write/Control Pin): This pin toggles the display among the read or writes operation, and it is connected to a microcontroller unit pin to get either 0 or 1 (0 = Write Operation, and 1 = Read Operation).
- Pin 6 (Enable/Control Pin): This pin should be held high to execute Read/Write process, and it is connected to the microcontroller unit & constantly held high.
- Pins 7-14 (Data Pins): These pins are used to send data to the display. These pins are connected in two-wire modes like 4-wire mode and 8-wire mode. In 4-wire mode, only four pins are connected to the microcontroller unit like 0 to 3, whereas in 8-wire mode, 8-pins are connected to microcontroller unit like 0 to 7.
- Pin15 (+ve pin of the LED): This pin is connected to +5V
- Pin 16 (-ve pin of the LED): This pin is connected to GND.

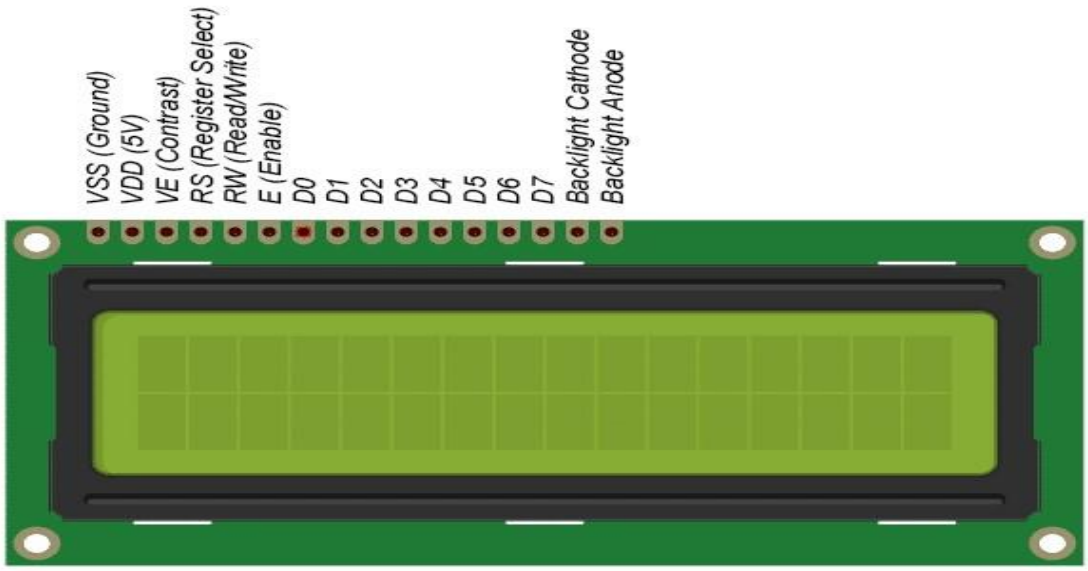


Fig 2.6: LCD 16x2

The features of this LCD mainly include the following.

- The operating voltage of this LCD is 4.7V-5.3V
- It includes two rows where each row can produce 16-characters.
- The utilization of current is 1mA with no backlight
- Every character can be built with a 5×8-pixel box
- The alphanumeric LCDs alphabets & numbers
- Is display can work on two modes like 4-bit & 8-bit
- These are obtainable in Blue & Green Backlight
- It displays a few custom generated characters

2.2.11 SWITCHED-MODE POWER SUPPLY (SMPS)

A switched-mode power supply (SMPS) is an electronic circuit that converts power using switching devices that are turned on and off at high frequencies, and storage components such as instructors or capacitors to supply power when the switching device is in its non-conduction state. Switching power supplies have high efficiency and are widely used in a variety of electronic equipment, including computers and other sensitive equipment requiring stable and efficient power supply. The SMPS is

mostly used where switching of voltages is not at all a problem and where efficiency of the system really matters. There are few points which are to be noted regarding SMPS. They are:

- SMPS circuit is operated by switching and hence the voltages vary continuously.
- The switching device is operated in saturation or cut off mode.
- The output voltage is controlled by the switching time of the feedback circuitry.
- Switching time is adjusted by adjusting the duty cycle.
- The efficiency of SMPS is high because, instead of dissipating excess power as heat, it continuously switches its input to control the output.



Fig 2.7: SMPS Device

Disadvantages:

- The noise is present due to high frequency switching.
- The circuit is complex.
- It produces electromagnetic interference.

Advantages:

- The efficiency is as high as 80 to 90%
- Less heat generation; less power wastage.
- Reduced harmonic feedback into the supply mains.
- The device is compact and small in size.
- The manufacturing cost is reduced.
- Provision for providing the required number of voltages.

2.2.12 Obstacle Sensor (IR)

The sensor module has a pair of infrared transmitting and receiving tube, when detecting direction meets with obstacles (reflecting surface), reflected infrared receiving by the tube. After the comparator circuit processing, green indicator will light up, at the same time the signal output interface output a digital signal (a low-level signal), can be adjust the detection distance through the potentiometer knob, effective distance range 2 ~ 80 cm, easy to assemble and use. So, it can be widely used in robot obstacle avoidance, obstacle avoidance car, line count, and so on many occasions. When this module works with a microcontroller unit then the pins are:

- ✦ Pin1 (VCC pin): Voltage supply rages from 3.3V to 5.3V
- ✦ Pin2 (GND): This is a ground pin
- ✦ Pin3 (DOUT): This is a digital output pin (MCU.IO)

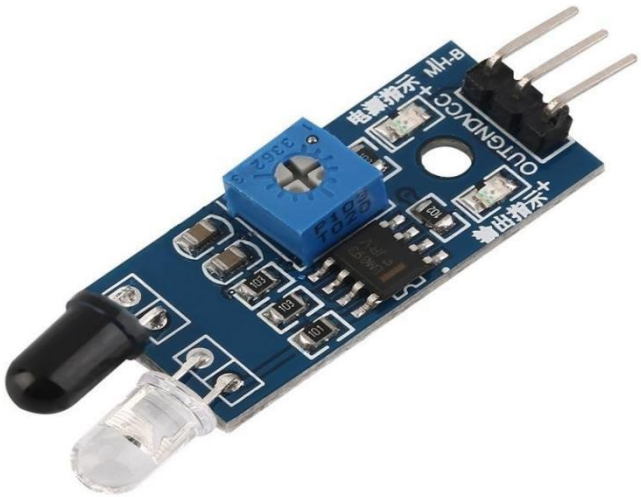


Fig 2.8: Obstacle Sensor (IR)

Features & Specifications: The features of this sensor include the following.

- If there is an obstacle, the green indicator light on the circuit board.
- Detection distance: 2 ~ 60cm
- Detection angle: 35 °
- Comparator chip: LM393
- 3mm screw holes for easy mounting
- Dimensions: 3.1 x 1.5cm
- Working Voltage:3.3 V to 5 V.

Performance:

- When there is no obstacle signal in front of the module, the OUT port will output high level (5V) continuously. When the module detects the front obstacle signal, the green indicator on the circuit board will light and the OUT port will continue to output low (0V) signal, the module detects the distance of 2 ~ 30cm, detection angle of 35 °, the detection distance can be adjusted by

potentiometer, clockwise potentiometer, the detection distance increases; counterclockwise potentiometer to reduce the detection distance.

- Sensor active infrared reflection detection, so the target's reflectivity and shape is the key to detecting the distance. One black detection ranges of the smallest, white maximum; small area of object distance, large area distance.
- Sensor module output port OUT can be directly connected with the micro controller IO port can also directly drive a 5V relay; Connection: VCC-VCC; GND-GND; OUT-IO
- The comparative with LM393, stable.
- Can be used 3.3V-5V DC power supply module. When the power is turned on, the red power indicator light.

Instructions for use:

- VCC external 3.3V-5V voltage (can be directly connected with the 5v micro controller or 3.3v single-chip)
- GND external GND
- OUT small board digital output interface (0 and 1)
- Can be used 3.3V-5V DC power supply module. When the power is turned on, the red power indicator light.

Instructions for use:

- VCC external 3.3V-5V voltage (can be directly connected with the 5v micro-controller or 3.3v single-chip)
- GND external GND
- OUT small board digital output interface (0 and 1)

2.3 Conclusion

This chapter describes the different types of equipment used in this project with their proper figure. Also discussed the literature survey of this project. Finally, we can say that from this chapter, the using equipment of this project is reliable, cheap, easily available in any

place. If we want, we can add more equipment to get extra benefit.

CHAPTER-3

HARDWARE IMPLEMENTATION OF THE PROPOSED PROJECT

3.1 Introduction

This project is a combination of flame sensors, temperature sensor, flame sensor power supply etc. which have different working principles and characteristic. In this chapter, we will explain how all of this equipment work and their characteristic with diagram.

3.2 Working Process

The proposed system uses an esp32 microcontroller to get the biometric information. Then start the system if the information matches. Then fuel info, alcohol info and speed are collected from sensors and processed in esp. These parameters can be continuously displayed on a mobile apps using IoT technology, which is a high-performance technology that integrates database recording and simulation on a single platform. The monitored parameters are compared to the rated values, and the microcontroller is programmed to take corrective action if the monitored values surpass the rated values, besides displaying the value on the app The microcontroller is programmed to constantly scan the area and send the parameters at fixed intervals to the cloud database.

3.3 Block Diagram of the Proposed Project:

Block diagram represents the whole project in small blocks. Here in Fig 3.1.1, we have shown the block diagram of our project. Driver Section is consisting of three different blocks Display, Identity Verification and Alcohol Sensor. Then all others parts such as Fuel level unit, Obstacle Detection unit data is collected through micro-controller. Then the output from the micro controller is sends to the Breaking and speed control unit which controls the wheel section. Speed of the car is measured from the wheel through the speed measure unit and sends to the microcontroller.

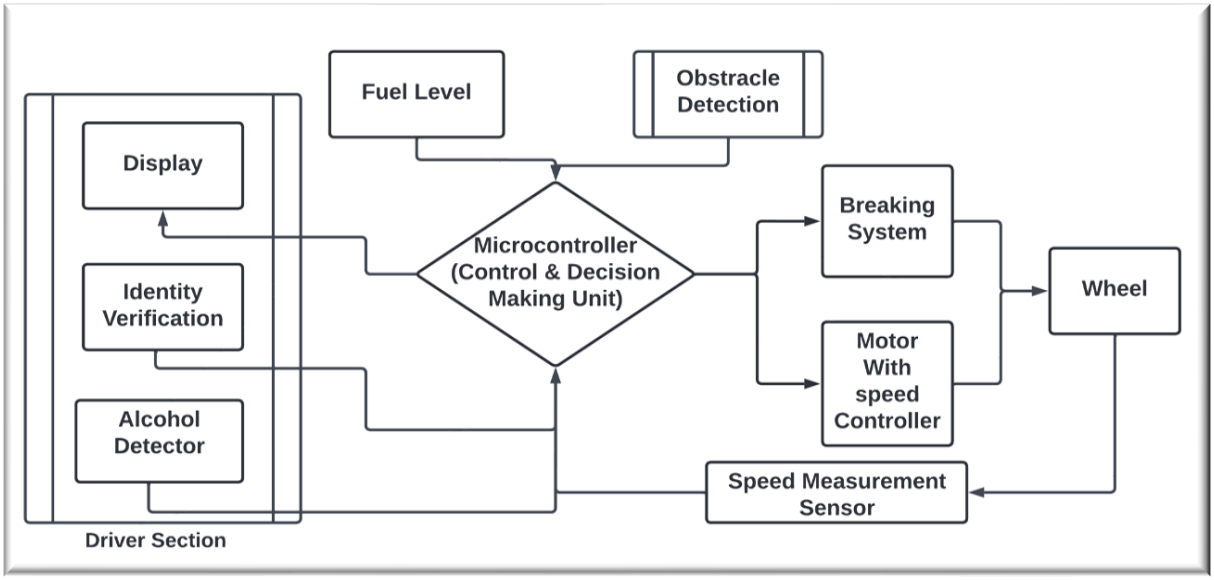


Fig 3.1.1: Overview of block diagram of the proposed project

3.4 Flow Chart of the Proposed Project:

A flowchart is a type of diagram that represents a work flow or process. A flowchart can also be defined as a diagrammatic representation of an algorithm, a step-by-step approach to solving a task.

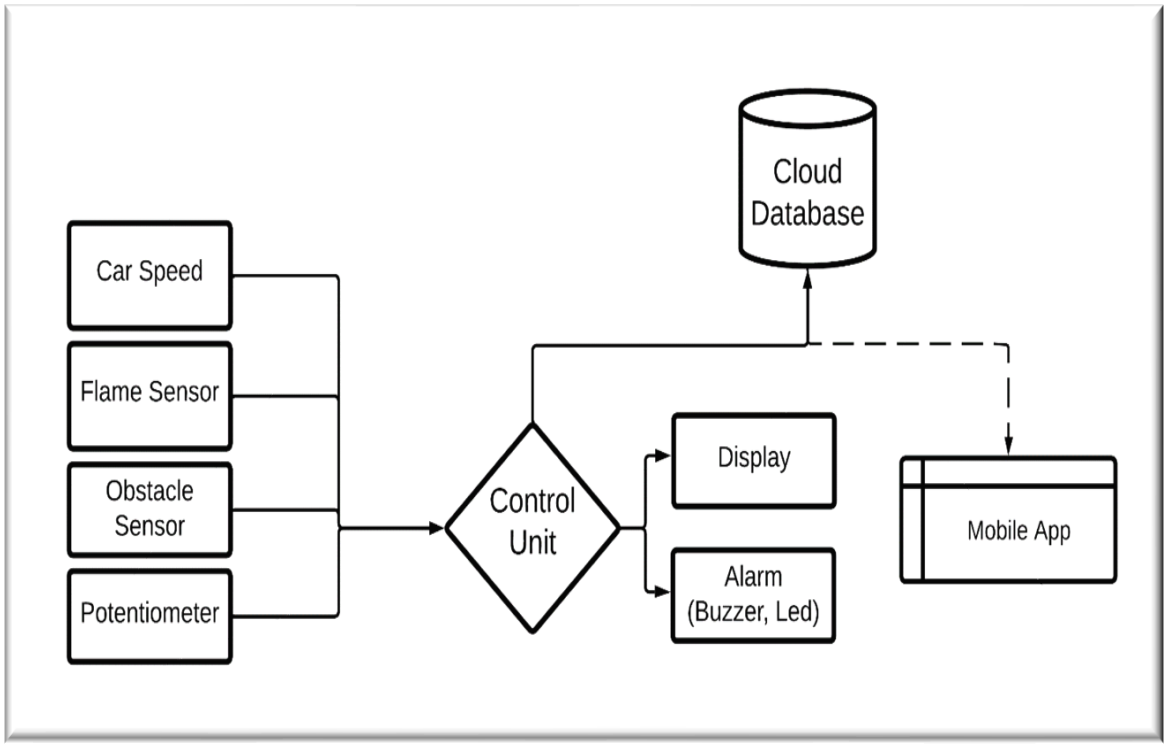


Fig 3.1.1: Flow Chart of the proposed project

In this flow chart we can see car speed, flame sensor, obstacle sensor, potentiometer, data came to control unit. Similarly control unit collect data and give it to display and alarm unit also send to data cloud data base. From cloud data base we can collect data by mobile apps.

3.5 Circuit Diagram of the Proposed Project:

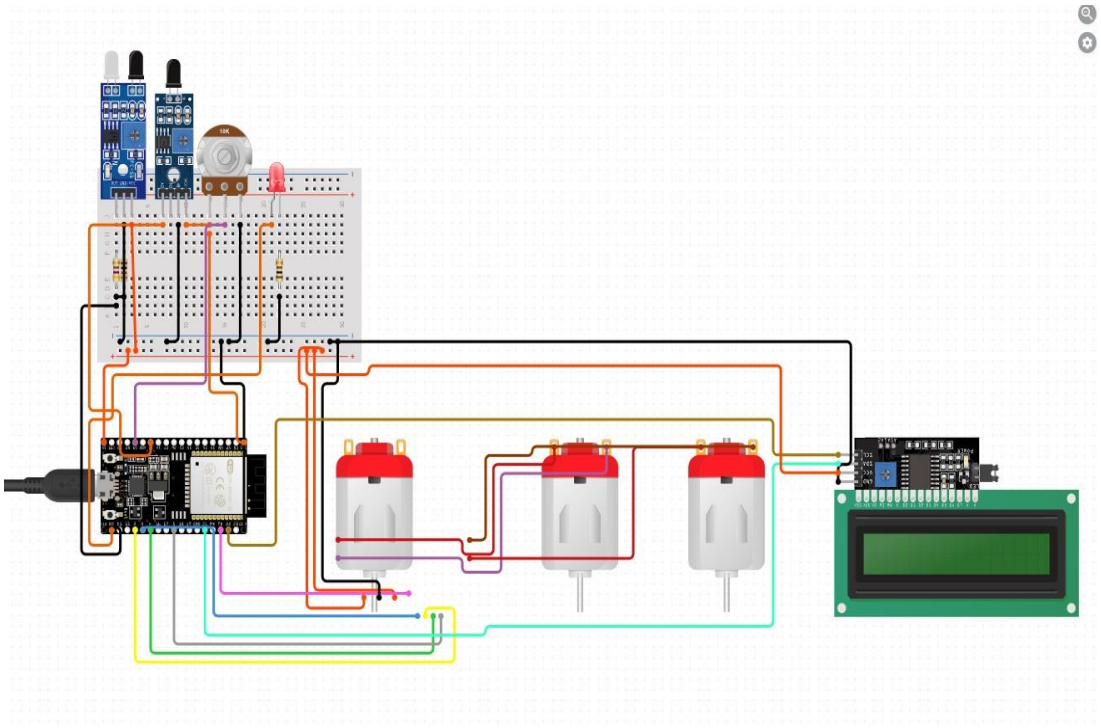


Fig 3.1.1: Circuit diagram of the proposed project

In the provided circuit diagram, there is a printed circuit board (PCB) serving as the central hub for interconnecting various components. An Arduino microcontroller is employed, programmed with specific code to execute control functions. This microcontroller interacts with multiple sensors integrated into the PCB, collecting data and, in response, instructing actions such as motor control and data display. If you require further details or professional assistance regarding this circuit, please feel free to inquire.

3.6 Final View of the Project:

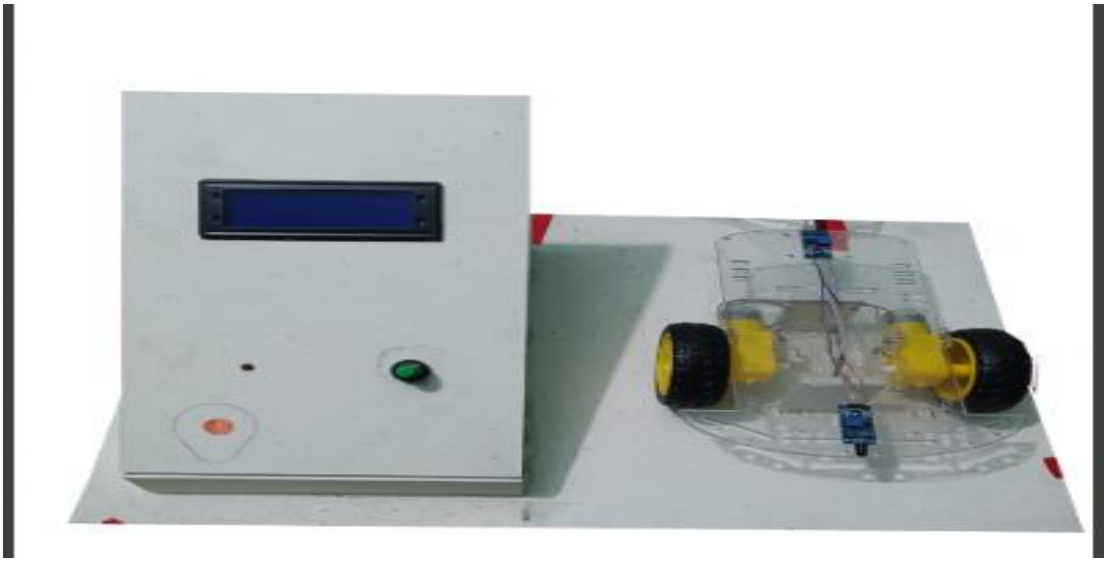


Fig 3.5.1: Final Project view



Fig 3.5.2: Mobile app view

In this picture which is collect from mobile apps we can see first left side have current speed. Right side have obstacle found detection. Under this left side have speed limit and right side have fire detection alarm unit.

CHAPTER-4

RESULT AND DISCUSSION

4.1 Introduction

This project is a combination of flame sensors, obstacle sensor, potentiometer, power supply etc. which have different working principles and characteristic. In this chapter we will discuss about the result and the benefit as well as demerits of the project.

4.2 Result

Mobile apps are used in this project for monitoring the data and set the speed limit. The micro-controller collects and retrieve data from a fire base cloud storage. The system usage a very small amount of data and bandwidth as show in Figure 4.1.

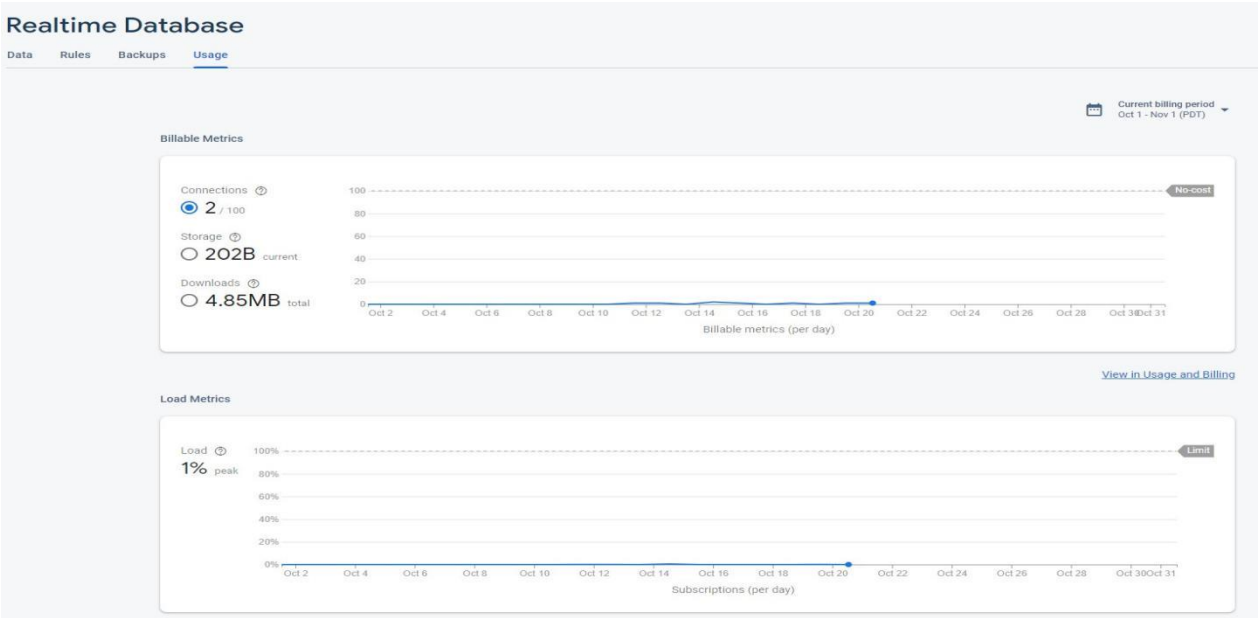


Fig 4.1: Data and bandwidth usage of the system

With the mobile apps we can monitor all kind of parameter and set the speed limit, shown in Fig 4.2

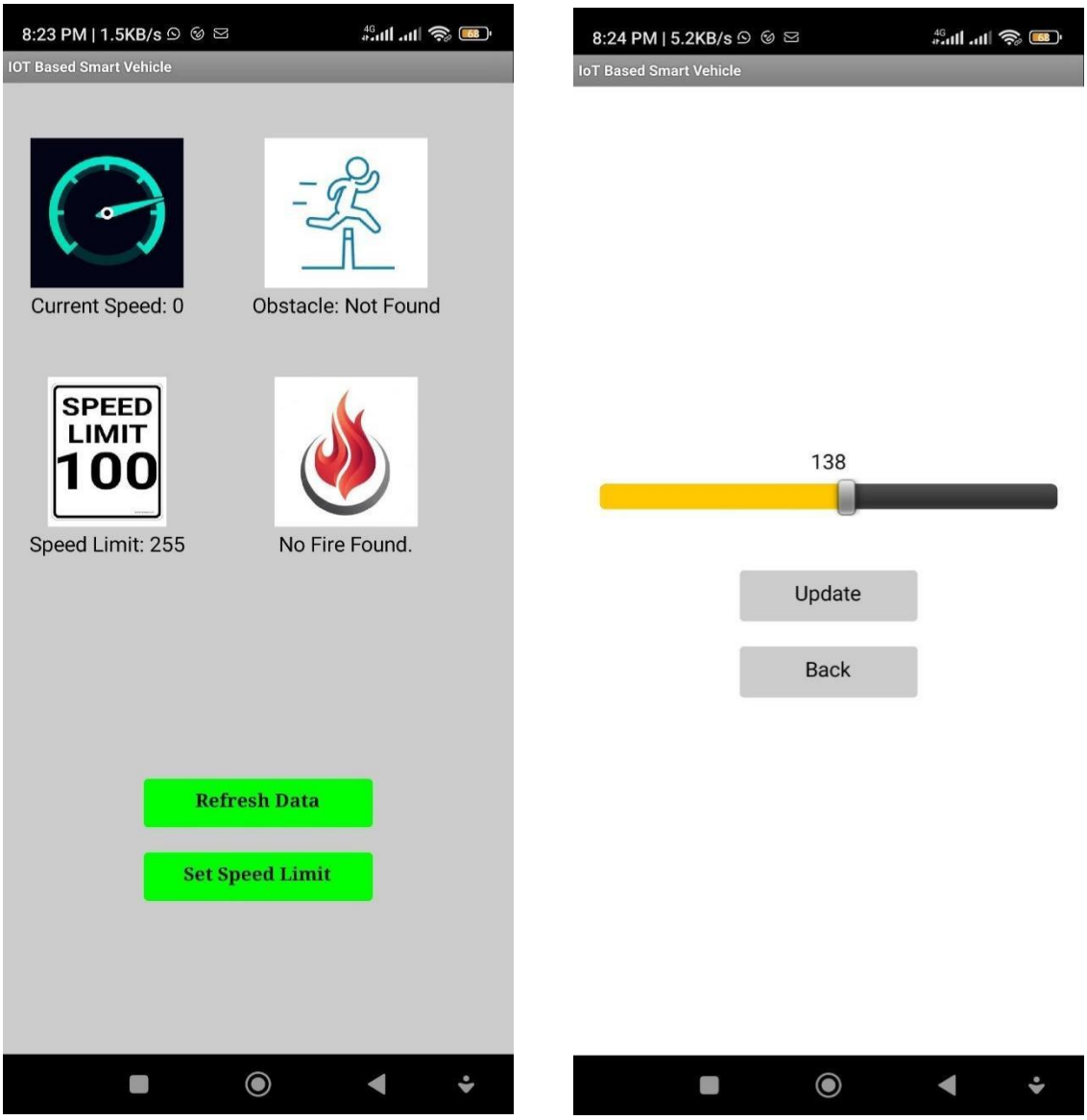


Fig 4.2: View and speed settings in the mobile apps

In the first picture, displayed through a mobile app, we observe the following elements:

1. *Left Side: * Current speed information.

2. *Right Side: * Obstacle detection status.

- Underneath the obstacle detection status, there's:

  - *Left Side: * Speed limit display.

  - *Right Side: * Fire detection alarm unit status.

In the second picture, it appears to demonstrate how to update the speed limit using a mobile app. If you have any specific questions or need further details, please let me know, and I'll be happy to assist.

4.3 Future Scope

This project has following advantages:

- Low cost IoT based system.
- Mobile app based real-time monitoring system.
- Speed control to avoid accident.
- Easy to implantable system.

Rather than that the project can be improvised by using more hardware and networking equipment's. Long range radio frequency could be used so that the wi-fi dependency could be minimized. Besides that, internal parameter such as voltage, current reading could be added into the monitoring unit so that better understanding of the current system environment could be achieved.

4.4 Discussion

Developed system is tested under various conditions. The RFID card reader sensor test for all the conditions and results are interpreted successfully. The obstacle sensor reading at different conditions is taken and updated. The wireless transmission was achieved using Wi- Fi. The data is stored in cloud server; the data is then retrieved successfully from mobile app which is used for monitoring purpose. The flame and obstacle sensor works well for detecting fire and sending emergency notification to

the user. The transportation system plays a vital role in our society. Therefore, the monitoring and protection of vehicle is very crucial. This system introduces a new and improved method of vehicle health and condition parameter monitoring using IoT. Further study, simulation and Real-time application is required. After that this could be implemented in an attractive price in Bangladesh.

1. IoT and Autonomous Vehicles Synergy: The project rightly recognizes the synergy between IoT and autonomous vehicles. IoT provides the necessary infrastructure for collecting and transmitting data in real-time, which is essential for the operation of autonomous vehicles. This symbiotic relationship is pivotal for the success of autonomous transportation.

2. Market Trends and Transformation: Understanding market trends is crucial, especially as they indicate the direction of technological progress. The projected transformation of the automotive industry is significant, with potential disruptions to traditional manufacturing, supply chains, and service models. It's essential for stakeholders to adapt to these changes proactively.

3. Adoption Timeline: Setting a target adoption timeline for autonomous vehicles, such as 2025, provides a sense of urgency. However, achieving widespread adoption involves overcoming numerous challenges, including regulatory, safety, and infrastructure hurdles. It's important to monitor progress toward this goal and be flexible in adjusting expectations.

4. Cost-Effective IoT Solutions: Developing cost-effective IoT systems for autonomous vehicles can democratize access to this technology. This is particularly important in regions with limited resources, as it can lead to improved mobility, reduced traffic congestion, and economic growth.

CHAPTER-5

CONCLUSION & RECOMMENDATION

5.1 Introduction

As a complex engineering project, this project has its own pros and cons. In this chapter we will discuss about the advantage, disadvantage and conclude the study by discussing the overall methods and future scopes.

5.2 Advantages

This project is focused to develop a low cost IoT based solution of smart vehicle. Among different advantages few main key points are as follow:

- Low cost IoT based system.
- Mobile app based real-time monitoring system.
- Speed control to avoid accident.
- Easy to implantable system.

5.3 Dis-advantages

Every project and research have some disadvantages. Few important shot comings of this project are as follow:

Need to apply and test in real time before final decision making.

Long term testing and simulation has not been conducted due to time and resource limitation.

Requires continuous internet connectivity for monitoring

5.4 Future Modification

The future modification of a project like the one described, which involves IoT technology for autonomous vehicles and its implementation in a country like Bangladesh, can be driven by technological advancements, regulatory changes, and evolving market dynamics. Here are potential areas for future modifications and improvements:

1. Technological Advancements:
   - Incorporating advancements in IoT technology, sensor technology, and artificial intelligence to enhance the safety and efficiency of autonomous vehicles.
   - Integrating 5G or future wireless communication standards to improve data transmission speeds and reduce latency.
2. Autonomous Vehicle Features:
   - Expanding the capabilities of autonomous vehicles, such as adding support for more complex urban environments, adverse weather conditions, or specialized use cases (e.g., logistics, agriculture).
   - Implementing vehicle-to-everything (V2X) communication for improved vehicle-to-vehicle and vehicle-to-infrastructure interactions.
3. Data Security and Privacy:
   - Enhancing cybersecurity measures to protect autonomous vehicles from potential threats and ensuring data privacy for vehicle occupants.
   - Complying with evolving data protection regulations and standards.
4. *Regulatory Changes: *
   - Adapting to changes in local and international regulations related to autonomous vehicles and IoT technology.
   - Collaborating with regulatory bodies to establish safety standards and guidelines.
5. Environmental Considerations:
   - Developing more environmentally friendly autonomous vehicles, such as electric or hydrogen-powered vehicles, to reduce emissions.
   - Incorporating sustainability practices into manufacturing and operations.
6. User Experience and Accessibility:
   - Improving the user interface and user experience within autonomous vehicles to make them more user-friendly and accessible to a broader range of passengers, including people with disabilities.

- Enhancing vehicle-to-passenger communication.

7. Scalability and Expansion:

- Scaling up the project to cover a broader geographical area within Bangladesh and potentially expanding to neighboring regions.
- Evaluating the feasibility of autonomous public transportation systems.

8. Economic and Social Impact:

- Continuously assessing the economic and social impact of the project, including its contribution to job creation, economic growth, and reduced traffic accidents.

9. Community Engagement:

- Engaging with local communities to address concerns, gather feedback, and ensure that the project aligns with the needs and values of the population.

10. Partnerships and Collaboration:

- Establishing new partnerships with technology companies, automotive manufacturers, research institutions, and government agencies to leverage expertise and resources.

11. Data Analytics and Insights:

- Utilizing data collected from the IoT system to gain insights into traffic patterns, transportation efficiency, and user behavior, which can inform future improvements.

12. Cost Efficiency and Affordability:

- Ongoing efforts to reduce the cost of autonomous vehicle technology to make it even more affordable for a wider range of users.

13. Alternative Mobility Solutions:

- Exploring innovative mobility solutions beyond traditional autonomous cars, such as autonomous drones or micro-mobility options.

Future modifications and enhancements to the project should be driven by a combination of technological innovation, regulatory compliance, public feedback, and the evolving needs of the region in which it operates. Flexibility and adaptability will be key to ensuring the project's long-term success and relevance.

5.4 Conclusion

This paper analyzed and looked at the IoT technology as a driver for Autonomous Cars with the focus on market trends of this technology and the trends of Autonomous Vehicles ecosystem. It was pointed out that the potential of this ecosystem is huge and that this will transform the automotive industry immensely. From the adoption perspective of this ecosystem, 2025 is seen as the target time frame by which time we will start seeing autonomous cars on our roads. For that purpose, a cost effective and easy to use system based on IoT is presented in this study. This project has a great possibility to open a path for lower income country like Bangladesh to adopt new autonomous vehicle technology easily and in a quick time to improve the national growth and stability. However, further study, simulation and Real time application is required. After that this could be implemented in an attractive price in Bangladesh.

REFERENCE:

[1] D. Minovski, C. Åhlund and K. Mitra, "Modeling Quality of IoT Experience in Autonomous Vehicles," in IEEE Internet of Things Journal, vol. 7, no. 5, pp. 3833-3849, May 2020, doi: 10.1109/JIOT.2020.2975418..

[2] X. Krasniqi and E. Hajrizi, “Use of IoT Technology to Drive the Automotive Industry from Connected to Full Autonomous Vehicles,” 17th IFAC Conference on International Stability, Technology and Culture TECIS 2016, vol. 8, pp. 1–10, Jan. 2019, doi: 10.11648/j.epes.20190801.11.

[3] S. Rahman, S. K. Dey, B. K. Bhawmick, and N. K. Das, “Design and implementation of real time transformer health monitoring system using GSM technology,” in 2017 International Conference on Electrical, Computer and Communication Engineering (ECCE), Feb. 2017, pp. 258–261. doi: 10.1109/ECACE.2017.7912915.

[4] “IJEEDC Transformer Health Monitoring and Control Through Arduino.” http://iraj.in/journal/IJEEDC/paper_detail.php?paper_id=6896&name=Transformer_Health_Monitoring_and_Control_Through_Arduino (accessed May 19, 2022).

[5] “Internet of Things (IoT): A Literature Review.” https://www.scirp.org/html/56616_56616.htm?pagespeed=noscript (accessed May 19, 2022).

[6] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): A vision, architectural elements, and future directions,” Future Generation Computer Systems, vol. 29, no. 7, pp. 1645–1660, Sep. 2013, doi: 10.1016/j.future.2013.01.010.

[7] N. Hossein Motlagh, M. Mohammadrezaei, J. Hunt, and B. Zakeri, “Internet of Things (IoT) and the Energy Sector,” Energies, vol. 13, no. 2, Art. no. 2, Jan. 2020, doi: 10.3390/en13020494.

[8] C. R. Srinivasan, B. Rajesh, P. Saikalyan, K. Premsagar, and E. S. Yadav, “A review on the different types of internet of things (IoT),” Journal of Advanced Research in Dynamical and Control Systems, vol. 11, no. 1, pp. 154–158, Jan. 2019.

[9] I. Lee and K. Lee, “The Internet of Things (IoT): Applications, investments, and challenges for enterprises,” Business Horizons, vol. 58, no. 4, pp. 431–440, Jul. 2015, doi: 10.1016/j.bushor.2015.03.008.

[10] S. Krčo, B. Pokrić, and F. Carrez, “Designing IoT architecture(s): A European perspective,” in 2014 IEEE World Forum on Internet of Things (WF-IoT), Mar. 2014, pp. 79–84. doi: 10.1109/WF-IoT.2014.6803124.

[11] M. Babiuch, P. Foltýnek, and P. Smutný, “Using the ESP32 Microcontroller for Data Processing,” in 2019 20th International Carpathian Control Conference
(ICCC), May 2019, pp. 1–6. doi: 10.1109/CarpathianCC.2019.8765944.

[12] M. S. Kumar, T. R. Chandra, D. P. Kumar, and M. S. Manikandan, “Monitoring moisture of soil using low cost homemade Soil moisture sensor and Arduino UNO,” in 2016 3rd International Conference on Advanced Computing and Communication Systems (ICACCS), Jan. 2016, vol. 01, pp. 1–4. doi: 10.1109/ICACCS.2016.7586312.

[13] G. M. Debele and X. Qian, “Automatic Room Temperature Control System Using Arduino UNO R3 and DHT11 Sensor,” in 2020 17th International Computer Conference on Wavelet Active Media Technology and Information Processing (ICCWAMTIP), Dec. 2020, pp. 428–432. doi: 10.1109/ICCWAMTIP51612.2020.9317307.

[14] A. Amin, “MONITORING WATER LEVEL CONTROL BERBASIS ARDUINO UNO MENGGUNAKAN LCD LM016L,” EEICT (Electric, Electronic, Instrumentation, Control, Telecommunication), vol. 1, no. 1, Art. no. 1, Apr. 2018, doi: 10.31602/eeict.v1i1.1421.

[15] B. V. Philip, T. Alpcan, J. Jin and M. Palaniswami, "Distributed Real-Time IoT for Autonomous Vehicles," in IEEE Transactions on Industrial Informatics, vol. 15, no. 2, pp. 1131-1140, Feb. 2019, doi: 10.1109/TII.2018.2877217.

[16] Khayyam, H., Javadi, B., Jalili, M., Jazar, R.N. (2020). Artificial Intelligence and Internet of Things for Autonomous Vehicles. In: Jazar, R., Dai, L. (eds) Nonlinear Approaches in Engineering Applications. Springer,Cham. https://doi.org/10.1007/978-3-030-18963-1_2

Appendix:

```
#define obsPin 25
#define firePin 33
#define pot 35
#define motor 26
#define led 27
int maxSpeed = 255, maxSpeedPV = 0;
#include <Wire.h>
#include <LiquidCrystal_I2C.h>
#include <analogWrite.h>
LiquidCrystal_I2C lcd(0x3F, 20, 4);
int speeds, obs, fire, card, spdPV = 500, firePV = 11, obsPV = 11, spdPVV = 500,
firePVV = 11, obsPVV = 11;

#include <SPI.h>
#include <MFRC522.h>

#define RST_PIN  34   // Configurable, see typical pin layout above
#define SS_PIN     32  // Configurable, see typical pin layout above

MFRC522 mfrc522(SS_PIN, RST_PIN);   // Create MFRC522 instance.

// Number of known default keys (hard-coded)
// NOTE: Synchronize the NR_KNOWN_KEYS define with the defaultKeys[] array
#define NR_KNOWN_KEYS   8
// Known keys, see: https://code.google.com/p/mfcuk/wiki/MifareClassicDefaultKeys
byte knownKeys[NR_KNOWN_KEYS][MFRC522::MF_KEY_SIZE] =  {
  {0xff, 0xff, 0xff, 0xff, 0xff, 0xff}, // FF FF FF FF FF FF = factory default
  {0xa0, 0xa1, 0xa2, 0xa3, 0xa4, 0xa5}, // A0 A1 A2 A3 A4 A5
  {0xb0, 0xb1, 0xb2, 0xb3, 0xb4, 0xb5}, // B0 B1 B2 B3 B4 B5
  {0x4d, 0x3a, 0x99, 0xc3, 0x51, 0xdd}, // 4D 3A 99 C3 51 DD
  {0x1a, 0x98, 0x2c, 0x7e, 0x45, 0x9a}, // 1A 98 2C 7E 45 9A
  {0xd3, 0xf7, 0xd3, 0xf7, 0xd3, 0xf7}, // D3 F7 D3 F7 D3 F7
  {0xaa, 0xbb, 0xcc, 0xdd, 0xee, 0xff}, // AA BB CC DD EE FF
  {0x00, 0x00, 0x00, 0x00, 0x00, 0x00}  // 00 00 00 00 00 00
};

void dump_byte_array(byte *buffer, byte bufferSize) {
  for (byte i = 0; i < bufferSize; i++) {
    Serial.print(buffer[i] < 0x10 ? " 0" : " ");
    Serial.print(buffer[i], HEX);
  }
```

```
}
bool try_key(MFRC522::MIFARE_Key *key)
{
 bool result = false;
 byte buffer[18];
 byte block = 0;
 MFRC522::StatusCode status;

 // Serial.println(F("Authenticating using key A..."));
 status = mfrc522.PCD_Authenticate(MFRC522::PICC_CMD_MF_AUTH_KEY_A,
block, key, &(mfrc522.uid));
 if (status != MFRC522::STATUS_OK) {
  // Serial.print(F("PCD_Authenticate() failed: "));
  // Serial.println(mfrc522.GetStatusCodeName(status));
  return false;
 }

 // Read block
 byte byteCount = sizeof(buffer);
 status = mfrc522.MIFARE_Read(block, buffer, &byteCount);
 if (status != MFRC522::STATUS_OK) {
  // Serial.print(F("MIFARE_Read() failed: "));
  // Serial.println(mfrc522.GetStatusCodeName(status));
 }
 else {
  // Successful read
  result = true;
  Serial.print(F("Success with key:"));
  dump_byte_array((*key).keyByte, MFRC522::MF_KEY_SIZE);
  Serial.println();
  // Dump block data
  Serial.print(F("Block ")); Serial.print(block); Serial.print(F(":"));
  dump_byte_array(buffer, 16);
  Serial.println();
 }
 Serial.println();

 mfrc522.PICC_HaltA();       // Halt PICC
 mfrc522.PCD_StopCrypto1();  // Stop encryption on PCD
 return result;
}

unsigned long getID() {
 if ( ! mfrc522.PICC_ReadCardSerial()) { //Since a PICC placed get Serial and
continue
```

```
  return 0;
 }
 unsigned long hex_num;
 hex_num =  mfrc522.uid.uidByte[0] << 24;
 hex_num += mfrc522.uid.uidByte[1] << 16;
 hex_num += mfrc522.uid.uidByte[2] <<  8;
 hex_num += mfrc522.uid.uidByte[3];
 mfrc522.PICC_HaltA(); // Stop reading
 return hex_num;
}

unsigned long key = 1227299763, uid = 0;
//------------------------------------------------------------------------------------------------------
------
#if defined(ESP32)
#include <WiFi.h>
#elif defined(ESP8266)
#include <ESP8266WiFi.h>
#endif
#include <Firebase_ESP_Client.h>

// Provide the token generation process info.
#include "addons/TokenHelper.h"
// Provide the RTDB payload printing info and other helper functions.
#include "addons/RTDBHelper.h"

// Insert your network credentials
//#define WIFI_SSID "DRC_WiFi"
//#define WIFI_PASSWORD "%drc404#"
#define WIFI_SSID "test"
#define WIFI_PASSWORD "1234567890"

// Insert Firebase project API Key
#define API_KEY "AIzaSyA1JpIp7pysvaNjXIEh3kUjv4Q1hEqYq6E"

// Insert RTDB URLefine the RTDB URL */
#define DATABASE_URL "https://jan2023-649a3-default-rtdb.firebaseio.com/"
// Define Firebase Data object
FirebaseData fbdo;

FirebaseAuth auth;
FirebaseConfig config;

unsigned long sendDataPrevMillis = 0, tDelay = 10000;
int count = 0;
```

```
bool signupOK = false;

//-------------------------------------------------------------------------------------------------
------
void setup()
{
 Serial.begin(115200);
 Serial.println("ESP Start...");
 lcd.init();
 lcd.backlight();
 lcd.setCursor(0, 2);
 lcd.print("System Starting...");
 pinMode(obsPin, INPUT);
 pinMode(firePin, INPUT);
 pinMode(pot, INPUT);
 pinMode(motor, OUTPUT);
 pinMode(led, OUTPUT);
 pinMode(buz, OUTPUT);
 //-------------------------------------------------------------------------------------
 WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
 Serial.print("Connecting to Wi-Fi");
 int ff = 100;
 while (WiFi.status() != WL_CONNECTED && ff > 0)
 {
  Serial.print(".");
  ff--;
  delay(300);
 }
 if (WiFi.status() != WL_CONNECTED)
 {
  lcd.clear();
  lcd.setCursor(3, 0);
  lcd.print("Wi-Fi Not Connected!!!");
  delay(1000);
 }
 Serial.println();
 Serial.print("Connected with IP: ");
 Serial.println(WiFi.localIP());
 Serial.println();

 /* Assign the api key (required) */
 config.api_key = API_KEY;

 /* Assign the RTDB URL (required) */
 config.database_url = DATABASE_URL;
```

```
/* Sign up */
if (Firebase.signUp(&config, &auth, "", ""))
{
  Serial.println("ok");
  signupOK = true;
}
else
{
  Serial.printf("%s\n", config.signer.signupError.message.c_str());
}

/* Assign the callback function for the long running token generation task */
config.token_status_callback = tokenStatusCallback; // see addons/TokenHelper.h

Firebase.begin(&config, &auth);
Firebase.reconnectWiFi(true);
//-------------------------------------------------------------------------------------

lcd.setCursor(0, 2);
lcd.print("     Starting...");

SPI.begin();
mfrc522.PCD_Init();

lcd.clear();
lcd.setCursor(0, 2);
lcd.print("Enter Valid Card..");

while (uid != key)
{
  if (mfrc522.PICC_IsNewCardPresent()) {
    uid = getID();
    if (uid != 0 && uid != key) {
      Serial.print("Card detected, UID: "); Serial.println(uid);
      lcd.clear();
      lcd.setCursor(2, 2);
      lcd.print("Wrong Card!!!");
      digitalWrite(buz, 1);
      delay(500);
      digitalWrite(buz, 0);
      delay(500);
      lcd.clear();
      lcd.setCursor(0, 2);
      lcd.print("Enter Valid Card..");
```

```
    }
   }
  }
  downData();
  lcd.clear();
  lcd.setCursor(2, 2);
  lcd.print("Welcome!");
  delay(500);
  lcd.clear();
}

void upData()
{
  if (fire != firePVV)
  {
   if (Firebase.RTDB.setInt(&fbdo, "carCard/fire", fire))
   {
    //    Serial.println("PASSED");
    //    Serial.println("PATH: " + fbdo.dataPath());
    //    Serial.println("TYPE: " + fbdo.dataType());
    firePVV = fire;
   }
   //  else
   //  {
   //    Serial.println("FAILED");
   //    Serial.println("REASON: " + fbdo.errorReason());
   //    return;
   //  }

  }

  if (obs != obsPVV)
  {
   if (Firebase.RTDB.setInt(&fbdo, "carCard/obs", obs))
   {
    obsPVV = obs;
   }
  }
  if (abs(spdPVV - speeds) > 5)
  {
   if (Firebase.RTDB.setInt(&fbdo, "carCard/spd", speeds))
   {
    spdPVV = speeds;
   }
  }
```

```
}

void downData()
{
  if ((millis() - sendDataPrevMillis > tDelay || sendDataPrevMillis == 0)) //tDelay
  {
    sendDataPrevMillis = millis();
    if (Firebase.RTDB.getString(&fbdo, "carCard/ spdlimit"))
    {
      if (fbdo.dataType() == "string")
      {
        maxSpeed = fbdo.stringData().toInt();
      }
    }
    if (maxSpeed != maxSpeedPV)
    {
//      lcd.setCursor(15, 0);
//      lcd.print("/    ");
//      lcd.setCursor(16, 0);
//      lcd.print(map(maxSpeed, 0, 255, 0, 100));
//      lcd.print("%");

      maxSpeedPV = maxSpeed;
    }

    String s = "Speed Limit:" + String(maxSpeed);
    Serial.println(s);
  }
}
void loop()
{
  speeds = analogRead(pot);
  fire = !digitalRead(firePin);
  obs = !digitalRead(obsPin);

  speeds = map(speeds, 0, 4096, 255, 0);
  if (speeds > maxSpeed)
  {
    speeds = maxSpeed;
  }
  Serial.println("speed=" + String(speeds));
  Serial.println("fire=" + String(fire));
  Serial.println("obs=" + String(obs));
  if (speeds < 50)
```

```
  speeds = 0;
else if (speeds > 255)
  speeds = 255;

if (obs != obsPV || fire != firePV)
{
  digitalWrite(buz, 1);
  delay(100);
  digitalWrite(buz, 0);
  delay(50);
  if ( !fire && !obs)
  {
    lcd.setCursor(0, 3);
    lcd.print("               ");
    lcd.setCursor(0, 3);
    lcd.print("    ALL OK        ");
    digitalWrite(led, 0);
    analogWrite(motor, speeds);

  }
  else if (obs)
  {
    lcd.setCursor(0, 3);
    digitalWrite(led, 1);
    lcd.print("  Prob: Obstacle!");
    analogWrite(motor, 0);
  }
  else if (fire)
  {
    lcd.setCursor(0, 3);
    digitalWrite(led, 1);
    lcd.print("  Prob: Fire!!!");
    analogWrite(motor, 0);
  }

}

if (abs(speeds - spdPV) > 3)
{

  lcd.setCursor(2, 0);
  lcd.print("Speed=            ");
  lcd.setCursor(10, 0);
  lcd.print(map(speeds, 0, 255, 0, 100));
  lcd.print("% /");
```

```
    lcd.print(map(maxSpeed, 0, 255, 0, 100));
    lcd.print("%");
    spdPV = speeds;
    if (!obs)
      analogWrite(motor, speeds);
  }
  if (fire != firePV)
  {
    lcd.setCursor(0, 1);
    if (fire)
    {
      lcd.print("Fire! Fire!!");
    }
    else
    {
      lcd.setCursor(0, 1);
      lcd.print("              ");
      lcd.setCursor(4, 1);
      lcd.print("No Fire.");
    }
    firePV = fire;
  }

  if (obs != obsPV)
  {
    lcd.setCursor(0, 2);
    if (obs)
    {
      lcd.print("Obstacle Ahead!");
      analogWrite(motor, 0);
    }
    else
    {
      lcd.setCursor(0, 2);
      lcd.print("              ");
      lcd.setCursor(0, 2);
      lcd.print("    No Obstacle");
      analogWrite(motor, speeds);
    }
    obsPV = obs;
  }
  upData();
  downData();

}
```