LIFE SAVING INTELLIGENT SAFETY HELMET WITH AUTO NOTIFICATIONS

Submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in Electronics and Communication Engineering

By

RUTH BALAJI (39130382)

SUBHARAJA CHELLAM.A (39130443)

DEVADHARSHINI (39130108)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

SCHOOL OF ELECTRICAL AND ELECTRONICS ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited with Grade "A" by NAAC

JEPPIAAR NAGAR, RAJIV GANDHI SALAI, CHENNAI- 600 119

APRIL- 2022

SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)
Accredited with Grade "A" by NAAC
JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600 119
www.sathyabama.ac.in

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATION

This is to certify that this Project Report is the bonafide work of RUTH BALAJI (39130382) , SUBHARAJA CHELLAM.A (39130443) and DEVADHARSHINI (39130108) who carried out the project entitled "**LIFE SAVING INTELLIGENT SAFETY HELMET WITH AUTO NOTIFICATIONS**" under my supervision from March 2022 to April 2022.

Internal Guide

DR.G. JEGAN

Head of the Department

DR.T.RAVI, M.E., Ph.D.

Submitted for Viva voce Examination held on_____

Internal Examiner External Examiner

DECLARATION

We, Ruth Balaji (39130382), Subharaja Chellam.A (39130443) and Devadharshini (39130108) hereby declare that the Project Report entitled **"Life Saving Intelligent Safety Helmet with Auto Notifications"** done by us under the guidance of **"Dr.G.JEGAN"** is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering/ Technology degree in **Electronics and Communication.**

DATE:

PLACE: Chennai

1.

2.

3.

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGMENT

We are pleased to acknowledge our sincere thanks to Board of Management of **SATHYABAMA** for their kind encouragement in doing this project and for completing it successfully. We are grateful to them.

We convey our thanks to **Dr.N.M.NANDHITHA.,M.E.,PH.D.** Dean, school of Electrical and Electronics Engineering and **Dr.T.RAVI., M.E., PH.D.** Head of the department. Dept. of Electronics and Communication Engineering for providing us necessary support and details at the right time during the progressive reviews.

We would like to express our sincere and deep sense of gratitude to my project guide **DR.G.JEGAN** for her valuable guidance, suggestions and constant encouragement paved way for the successful completion of our project work.

We wish to express our thanks to all Teaching and Non- teaching staff members of the Department of Electronics and Communication Engineering who were helpful in many ways for the completion of the project.

Ruth Balaji

Subharaja Chellam

Devadharshini

PROJECT DETAILS

PROJECT TOPIC: Life Saving Intelligent Safety Helmet with Auto Notifications.

PROJECT GUIDE: Dr.G.JEGAN

STUDENT 1:-

NAME OF THE STUDENT: Ruth Balaji

REGISTRATION NUMBER:- 39130382

STUDENT 2:-

NAME OF THE STUDENT: Subharaja Chellam

REGISTRATION NUMBER: 39130443

STUDENT 3:-

NAME OF THE STUDENT: Devadharshini

REGISTRATION NUMBER: 39130108

ABSTRACT

An accident is a specific, unexpected, unusual and unintended external action which occurs in a particular time and place, with no apparent and deliberate cause but with marked effects. Carelessness of the driver is the major factor of such accidents. The traffic authorities give a lot of instructions to the vehicle operators. But many of them do not obey the rules. Nowadays most of the countries are forcing the motor riders to wear the helmet and not to use the vehicles when the person is in drunken condition. But still the rules are being violated by the users. In order to overcome this, in our Proposed System we wanted to incorporate efficient safety feature for Two-Wheeler drivers which will help them save their life with less time consumption-based help system. We have a dedicated Smart switch, so only upon wearing the helmet the driver will be able to start the bike that is the first condition, second condition the helmet will automatically detect for alcohol consumption if the driver detects positive for alcohol, he will not be in a position to start the bike. As a third level Safety Feature we have a fall detection sensor which helps to detect the accident , when talking about accidents there are 2 types light accident and Major Accident if the accident is light then we have an Emergency key in our bike , upon pressing it will stop all the auto-notifications which will help you immediately ignite the bike and can use it further and in case of an Major Accident we have an auto-notification system will will send notification to Ambulance , Family Member etc but till the time they come we cannot leave the driver to be in pain or leave un attended we add an alert system specifically for accident detection , upon hearing that sound nearby people can come in rescue of the person and save his life at the earliest.

TABLE OF CONTENTS

TABLE OF FIGURES

CHAPTER 1

INTRODUCTION

1.1. INTRODUCTION TO INTERNET OF THINGS(IoT):

Internet of Things (IoT) is a network of physical objects or people called “things” that are embedded with software, electronics, network, and sensors that allows these objects to collect and exchange data. The goal of IoT is to extend to internet connectivity from standard devices like computer, mobile, tablet to relatively dumb devices like a toaster. IoT makes virtually everything “smart,” by improving aspects of our life with the power of data collection, AI algorithm, and networks. The thing in IoT can also be a person with a diabetes monitor implant, an animal with tracking devices, etc. This IoT tutorial for beginners covers all the Basics of IoT.

1.2. CHALLENGES OF IoT:

At present IoT is faced with many challenges, such as:

- Insufficient testing and updating
- Concern regarding data security and privacy
- Software complexity
- Data volumes and interpretation
- Integration with AI and automation
- Devices require a constant power supply which is difficult
- Interaction and short-range communication

1.3. ADVANTAGES AND DISADVANTAGES OF IoT:

ADVANTAGES:

- **Technical Optimization:** IoT technology helps a lot in improving technologies and making them better. Example, with IoT, a manufacturer is able to collect data from various car sensors. The manufacturer analyses them to improve its design and make them more efficient.
- **Improved Data Collection:** Traditional data collection has its limitations and its design for passive use. IoT facilitates immediate action on data.
- **Reduced Waste:** IoT offers real-time information leading to effective decision making & management of resources. For example, if a manufacturer finds an issue in multiple car engines, he can track the manufacturing plan of those engines and solves this issue with the manufacturing belt.
- **Improved Customer Engagement:** IoT allows you to improve customer experience by detecting problems and improving the process.

DISADVANTAGES:

- **Security:** IoT technology creates an ecosystem of connected devices. However, during this process, the system may offer little authentication control despite sufficient security measures.
- **Privacy:** The use of IoT, exposes a substantial amount of personal data, in extreme detail, without the user's active participation. This creates lots of privacy issues.
- **Flexibility:** There is a huge concern regarding the flexibility of an IoT system. It is mainly regarding integrating with another system as there are many diverse systems involved in the process.
- **Complexity:** The design of the IoT system is also quite complicated. Moreover, it's deployment and maintenance also not very easy.
- **Compliance:** IoT has its own set of rules and regulations. However, because of its complexity, the task of compliance is quite challenging.

1.4. REQUIREMENTS:

The key requirements for any IoT security solution are:

- Device and data security, including authentication of devices and confidentiality and integrity of data.
- Implementing and running security operations at IoT scale.
- Meeting compliance requirements and requests.
- Meeting performance requirements as per the use case.

1.5. PROGRAMMING LANGUAGES:

There are different types of programming languages used for IoT, they are:

JAVA:

As far as IoT app development is concerned, JAVA has the most prominence in the market. In the year 2019, it was the most popular programming language with an overall rating of 16.61%. The programming language alone has powered close to three billion devices. One of the major reasons why JAVA is prominent because of its Code Once Run Anywhere functionality. This means that developers can code their app once and run it on any device that is compatible with the programming

language (from cell phones to the simplest of devices). Besides, JAVA is object-oriented, which allows you to develop apps for both cloud and edge nodes. Its interoperable functionalities and availability of extensive libraries, make JAVA an ideal programming language for IoT development.

PYTHON:

Another in-demand programming language choice for IoT Product development, Python offers seamless code readability and simple syntax features. Being an interpreted language, it is highly compatible with object-oriented, structured, and functional programming. Python can be integrated with other programming languages like Java and C++, and the language also works across diverse platforms including Linux and Windows. It's comprehensive library and solid community support make Python a developer's delight. For IoT apps that require extensive data analysis, Python is the ultimate choice.

C LANGUAGE:

Launched close to two decades back, the programming language is still as relevant as it gets. Amidst all the competition from the recent programming languages, C continues to be preferred by developers to build IoT applications. The language offers flexibility to developers apart from offering features like interoperability, rich libraries, and portability. C is also super compatible with micro-controllers that are integral peripheral devices of IoT architectures. However, one hurdle developers are most likely to face is the initial difficulty in learning the programming language. The syntax is quite complicated and its architecture is layered. Once this is overcome, there is no turning back for developers developing an IoT application.

JAVA SCRIPT:

Hailed in the world of web development, JavaScript shares its libraries with other programming languages. The use of JavaScript for IoT app development makes powering devices interoperable. One of the best advantages of JavaScript is that it works across diverse environments including gateways and the cloud. The presence of an active developer community ensures that all your hiccups can be solved through direct responses, featured articles, tutorials, and more. With sensors being inevitable in IoT architecture, JavaScript's event-driven features make the environment it runs in more functional and efficient.

SWIFT:

Swift is very unique in this list. While other programming languages discussed earlier can be used to develop IoT applications for the web, smartphones, and devices across multiple platforms, Swift is a language specifically designed to develop IoT applications for Ios devices. It falls in line with Apple's design guidelines, allowing developers to build IoT apps that meet the company's benchmarking standards. It's fast, secure, and seamlessly handles errors and offers tons of programming patterns. If your IoT app development goals only revolve around Ios devices, this is

the most ideal programming language. As far as the learning curve is concerned, it's simple syntax and ease of coding make it one of the easiest programming languages to master. With increasing prominence, aspiring IoT developers should get hands on with Swift as soon as possible.

C++:

++ programming language has processing power over C. This advantage makes C++ ideal as a pre-processing catalyst for C. C++ stimulates the processing power of C, helping it to run higher-level programming languages. Although C++ is a complex language and developers can make a lot of mistakes with it, it remains programmers' favourite. This programming language shows its strength in Linux projects and the embedded programming space with its capability to abstractions and object layers. C++ also encourages the use of C#, Python, Java, and a lot more languages.

CHAPTER 2

LITERATURE SURVEY

This chapter provides a literature review on the innovation and challenges of the visitor counter system with smart power management. A summary of the past and present research was provided for this reason in three sections: Early innovations, Modern trends and Future challenges.

2.1. EARLY DEVELOPMENT:

The concept of adding sensors and intelligence to physical objects was first discussed in the 1980s, when some university students decided to modify a Coca-Cola vending machine to track its contents remotely. But the technology was bulky and progress was limited. The term 'Internet of Things' was coined in 1999 by the computer scientist Kevin Ashton. While working at Procter & Gamble, Ashton proposed putting radio-frequency identification (RFID) chips on products to track them through a supply chain. He reportedly worked the then-buzzword 'internet' into his proposal to get the executives' attention. And the phrase stuck.
Over the next decade, public interest in IoT technology began to take off, as more and more connected devices came to market. In 2000, LG announced the first smart refrigerator, in 2007 the first iPhone was launched and by 2008, the number of connected devices exceeded the number of people on the planet. In 2009, Google started testing driverless cars and in 2011, Google's Nest smart thermostat hit the market, which allowed remote control of central heating.

2.2. MODERN TRENDS:

5G:

IoT use cases that require low latency, such as connected cars, predictive maintenance, and wearable tech in healthcare, will benefit the most from 5G. 5G's ultra-reliable, low-latency communication (URLLC) capacity and support for TSN (Time-Sensitive Networking) will be very important for IoT adoption.

HEALTH TECH:

The healthcare sector has long resisted the digital revolution, lagging far behind other industries. The Covid-19 pandemic has led to the rapid adoption of medical IoT technologies such as remote patient monitoring and medical robots, which has opened up vast digitalisation opportunities for IoT solution providers.

ARTIFICIAL INTELLIGENCE OF THINGS (AioT):

Artificial intelligence (AI) technologies are often used to interpret and react to some of the human-to-machine and machine-to-machine data flows in real-time. The amalgamation of the two technologies, AI and IoT, has given rise to the concept of AioT, which involves embedding AI technology into IoT components. Combining data collected by connected sensors and actuators with AI allows for reduced latency, increased privacy, and real-time intelligence at the edge. It also means that less data needs to be sent and stored on cloud servers.

INTELLIGENCE EDGE COMPUTING:

Sensors will pop up everywhere as IoT technology grows, which will result in a huge increase in the amount of data collected. The processing power on most IoT devices, however, is limited so the data processing takes place in the cloud, in data centres that are typically far away from where the IoT devices that generate the sensor data.

Some of the data analytics functions are shifting to the edge of the network, closer to the source of data generation as IoT ecosystems become more complex to reduce latency and enable near-autonomous decision making when responding to sensor signals from IoT devices. This is especially important for time-critical use cases such as health monitoring devices or autonomous vehicles, where split-second reactions can save lives.

IoT AS A SERVICE (IoTaaS):

IoTaaS vendors provide various platforms to assist organisations with IoT deployment without the need for in-house expertise. The technology aims to make it easy for enterprises to deploy and manage their connected devices. It has become an accelerator for enterprise IoT adoption, especially in predictive maintenance, advanced automation, and condition monitoring. IoTaaS revenues are likely to grow dramatically as the world recovers from Covid-19.

DIGITAL TWINS:

Digital twins can help optimise IoT deployments for maximum efficiency and help IoT adopters figure out where things should go or how they operate before they are physically deployed. A digital twin is a software representation of physical assets and processes that allows an organisation to carry out ‘what if’ simulations. These simulations can be used to spot and avert problems proactively, help prevent downtime, and speed up the development of new products.

WEARABLES:

Covid-19 driven changes in consumer behaviour including working from home, increased digital media consumption, and the popularity of virtual fitness workouts is driving the adoption of wearables and increasing consumer IoT adoption. Wearable

technology vendors are integrating a range of health and fitness monitoring options into their devices, aided by advances in biometric sensor technologies.

NEXT-GEN CHIPS:

The emphasis in chip design has shifted from a race to place more transistors onto a square millimetre of silicon to a focus on building microprocessors as systems made up of multiple components, each of which performs a specialised task. The pressure on the semiconductor industry to develop smaller, cheaper, and faster chips ratchets up as more and more sensors and microcontrollers are packed into the connected devices. The underlying semiconductor technology embedded within IoT devices needs to be cheaper, more compact, and consume less power for IoT to be pervasive.

SOFTWARE DEFINED NETWORKING (SDN):

SDN is an emerging architecture for data networks that allows software rather than hardware to control the network path along, which data packets flow. It is still under development, but, ultimately, it may replace internet protocol (IP) networking, a hardware standard, as the main standard governing the transmission mechanisms of the internet.

SDN will have a big impact on IoT ecosystems because it fundamentally changes who controls the data centre. SDN has been slow to take off, primarily because of cybersecurity concerns. SDN's open hardware standards threaten to commoditise networking and data centre hardware, which is still largely based on proprietary systems, making it easier for internet companies to programme data networks.

2.3. FUTURE CHALLENGES:

2.3.1. TECHNOLOGY:

IoT components are implemented using divergent protocols and technologies. As a result, these components have intricate configurations and poor design. Technological challenges can be a reflection of five parameters.

- **SECURITY**:
  IoT has happened to cause major security issues that have grabbed the attention of various public and private sector companies of the world. Adding such a massive number of new hubs to the systems and the web will provide attackers with a larger platform to invade the system, particularly as many experience the ill effects of security holes. Indications suggested that the malware captured infinite number of IoT gadgets that are being used in basic applications like smart-home devices and closed-circuit cameras and deployed them against their own servers. A further critical move in security will develop from the way IoT turns out to be involved in our lives Some study proves that cameras connected to the internet will contribute 30% to security concerns. Others are being 15% on house doors, 12% on cars, 10% on TVs, 6% due to iron, 6% on heating systems, 6% on smoke systems, 5% and 5% on an oven and lightening each.

- **CONNECTIVITY:**
  The most significant challenges of the future of IoT would be to connect several devices, this communication will end up resisting the currently existing structure and the technologies associated with it. Presently, a centralized, server/client architecture is being utilized to authenticate, authorize and connect several terminals in a network . This model is appropriate only for the current situation and is not scalable to cater future needs where billions of devices will be part of a single network. This scenario will transform the current centralized system into a bottleneck. Large amount of investments and expenditure in maintaining the cloud clusters of servers are required which can deal with humongous quantity of information exchange, as unavailability of servers can lead to a total system shutdown.

- **COMPATIBILITY AND LONGEVITY:**
  IoT is developing in a widespread manner. It is incorporating many technologies and will soon advance into a convention. This will pose serious challenges and will demand setting up of additional software and hardware in order to establish communication amongst the devices. Unavailability of standardized M2M protocols, Non-unified cloud services, and varieties in firmware and operating systems among IoT devices are some of the other compatibility issues. Devices working on these technologies will become purposeless in future as these technologies are going to become outdated very soon.

- **STANDARDS:**
  Technology conventions incorporating network and communication protocols, and data-aggregation conventions, are the collection for activities that handle, process and store information obtained from several sensors. These enhance the data by increasing the scale, scope, and frequency of data available for analysis .

- **INTELLIGENT ANALYSIS AND ACTIONS**:
  The final step in the implementation of IoT is the revelation about the data for analysis. The analysis procedure is based on cognitive technologies and models. There are certain parameters that cause intelligent actions to be incorporated in IOT, some of them being lesser device cost, enhanced device functionality, the machine “influencing” human actions through behavioural-science rationale, deep learning tools, machines’ actions in unusual scenarios, information security and privacy and device interoperability.

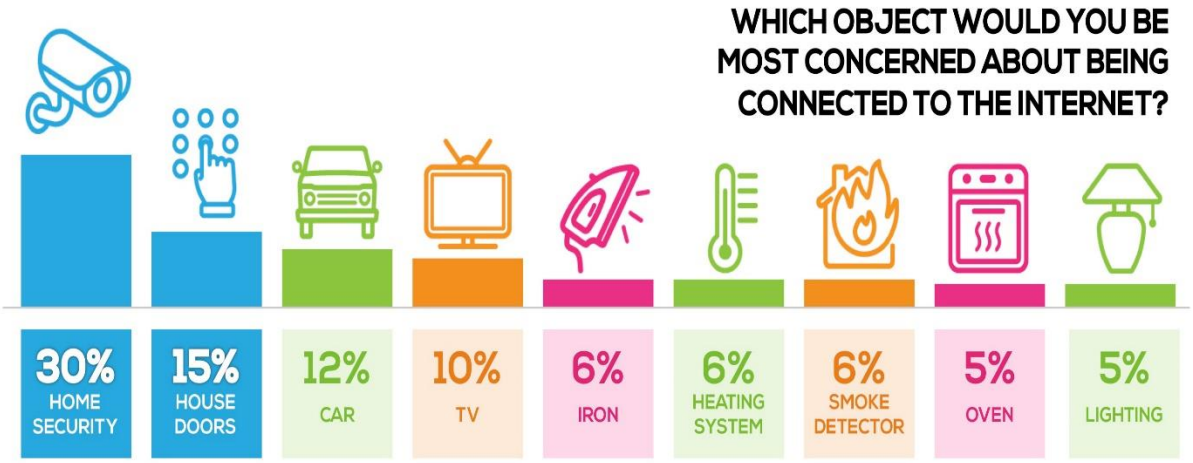


FIGURE 1: CONCERNS OF IoT

2.3.2. BUSSINESS:

The main issue is a major inspiration for beginning, putting resources into, and managing any venture, without a full proof plan of action for IoT we will have another bubble, this model should fulfil every one of the prerequisites for all kinds of e-commerce; vertical markets, horizontal markets, and consumer markets. Be that as it may, this class is always a sufferer of administrative and lawful inspection. Usage of IoT technologies plays a significant role to create a source of additional income to reduce the burden on the existing communication infrastructure.

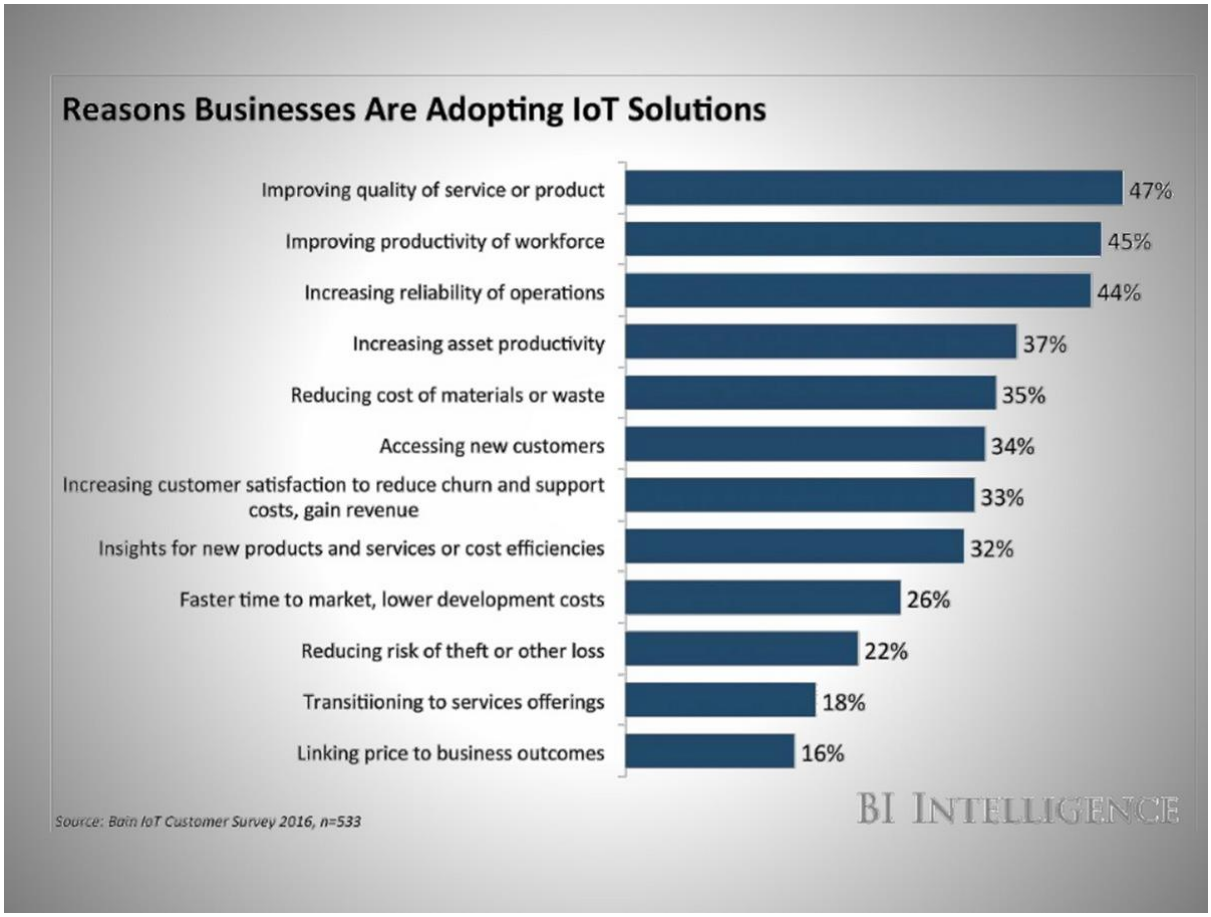


FIGURE 2: REASONS BUSINESS MAN ARE ADOPTING IoT

2.3.3. SOCIETY:

Understanding IoT from the clients and regulators point of view isn't a simple errand for the following reasons:

• Customer requests and requirements change regularly.

• New uses for devices—and also new devices—grow and develop dangerously fast.

• Inventing and reintegrating have features and capabilities that are costly and require significant investment and assets.

• The uses for an IoT technology are growing and changing—regularly in uncharted waters.

• Consumer Confidence: Each of these issues could put a dent in buyers' want to buy associated items, which would keep the IoT from satisfying its real potential.

IOT data is a very sensitive data which if leaked can give the control of the system in the attack's hands. Hence we have to have the strong and reliable technology to secure how IOT data is being used. Business policies and procedures pose some social challenges to IOT and government laws, and rules pose legal challenges to its use.

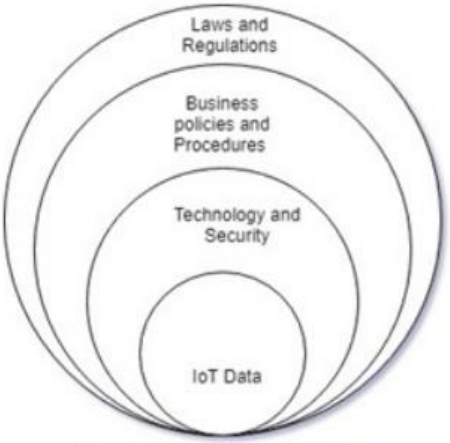


Fig. 10 Social Impact of IoT

FIGURE 3: SOCIAL IMPACT OF IOT

CHAPTER-3

AIM & SCOPE

3.1. AIM:

The main aim of the project is to make sure of the safety of the public from accidents due to alcohol consumption and to make sure the public wear their helmets while driving.

3.2. PURPOSE:

An accident is a specific, unexpected, unusual and unintended external action which occurs in a particular time and place, with no apparent and deliberate cause but with marked effects. Carelessness of the driver is the major factor of such accidents. The traffic authorities give a lot of instructions to the vehicle operators. But many of them do not obey the rules. Nowadays most of the countries are forcing the motor riders to wear the helmet and not to use the vehicles when the person is in drunken condition. But still the rules are being violated by the users. In order to overcome this, in our Proposed System we wanted to incorporate efficient safety feature for Two-Wheeler drivers which will help them save their life with less time consumption-based help system. We have a dedicated Smart switch, so only upon wearing the helmet the driver will be able to start the bike that is the first condition, second condition the helmet will automatically detect for alcohol consumption if the driver detects positive for alcohol, he will not be in a position to start the bike. As a third level Safety Feature we have a fall detection sensor which helps to detect the accident , when talking about accidents there are 2 types light accident and Major Accident if the accident is light then we have an Emergency key in our bike , upon pressing it will stop all the auto-notifications which will help you immediately ignite the bike and can use it further and in case of an Major Accident we have an auto-notification system will send notification to Ambulance , Family Member etc. but till the time they come we cannot leave the driver to be in pain or leave un attended we add an alert system specifically for accident detection , upon hearing that sound nearby people can come in rescue of the person and save his life at the earliest.

CHAPTER-4

MATERIALS AND METHOD

4.1. HARDWARE MATERIALS:

The physical components that a system requires to function:

- Arduino Uno
- Lcd Display
- Lcd Base
- Accelerometer Sensor
- Push Buttons
- Gsm
- Buzzer
- Battery

ARDUINO UNO:

Arduino UNO is a low-cost, flexible, and easy-to-use programmable open-source microcontroller board that can be integrated into a variety of electronic projects. This board can be interfaced with other Arduino boards, Arduino shields, Raspberry Pi boards and can control relays, LEDs, servos, and motors as an output. Arduino UNO features AVR microcontroller Atmega328, 6 analogue input pins, and 14 digital I/O pins out of which 6 are used as PWM output. This board contains a USB interface i.e. USB cable is used to connect the board with the computer and Arduino IDE (Integrated Development Environment) software is used to program the board. The unit comes with 32KB flash memory that is used to store the number of instructions while the SRAM is 2KB and EEPROM is 1KB. The operating voltage of the unit is 5V which projects the microcontroller on the board and its associated circuitry operates at 5V while the input voltage ranges between 6V to 20V and the recommended input voltage ranges from 7V to 12V.

The Arduino UNO board contains the following components and specifications:

ATmega328: This is the brain of the board in which the program is stored.

Ground Pin: there are several ground pins incorporated on the board.

PWM: the board contains 6 PWM pins. PWM stands for Pulse Width Modulation, using this process we can control the speed of the servo motor, DC motor, and brightness of the LED.

Digital I/O Pins: there are 14 digital (0-13) I/O pins available on the board that can be connected with external electronic components.

Analogue Pins: there are 6 analogue pins integrated on the board. These pins can read the analogue sensor and can convert it into a digital signal.

AREF: It is an Analog Reference Pin used to set an external reference voltage.

Reset Button: This button will reset the code loaded into the board. This button is useful when the board hangs up, pressing this button will take the entire board into an initial state.

USB Interface: This interface is used to connect the board with the computer and to upload the Arduino sketches (Arduino Program is called a Sketch)

DC Power Jack: This is used to power up the board with a power supply.

Power LED: This is a power LED that lights up when the board is connected with the power source.

Micro SD Card: The UNO board supports a micro SD card that allows the board to store more information.

3.3V: This pin is used to supply 3.3V power to your projects.

5V: This pin is used to supply 5V power to your projects.

VIN: It is the input voltage applied to the UNO board.

Voltage Regulator: The voltage regulator controls the voltage that goes into the board.

SPI: The SPI stands for Serial Peripheral Interface. Four Pins 10(SS), 11(MOSI), 12(MISO), 13(SCK) are used for this communication.

TX/RX: Pins TX and RX are used for serial communication. The TX is a transmit pin used to transmit the serial data while RX is a receive pin used to receive serial data.

FIGURE 4: ARDUINO UNO

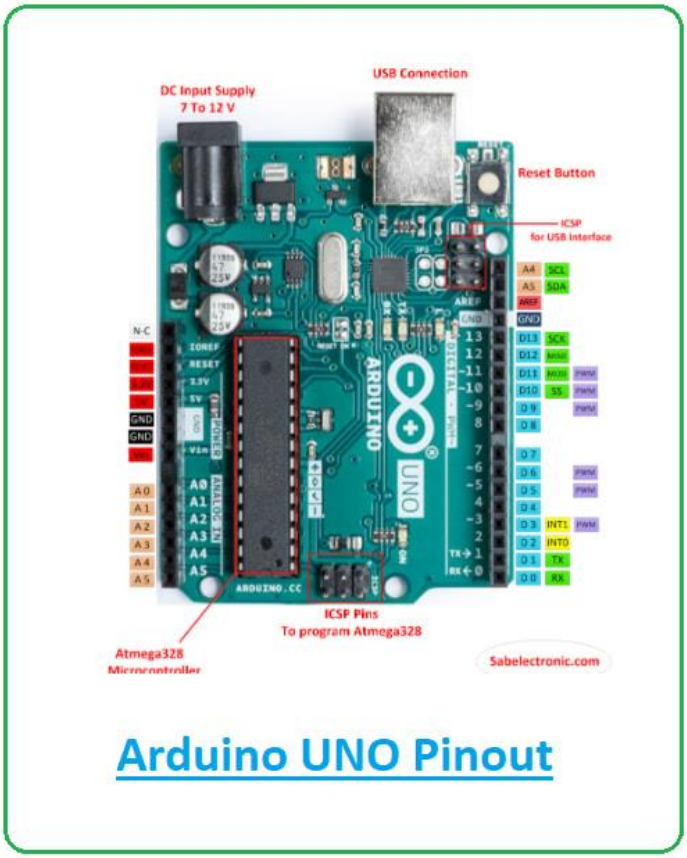


FIGURE 5: ARDUINO UNO PINOUT

LIQUID CRYSTAL DISPLAY:

Liquid crystal displays (LCDs) have materials which combine the properties of both liquids and crystals. Rather than having a melting point, they have a temperature range within which the molecules are almost as mobile as they would be in a liquid, but are grouped together in an ordered form similar to a crystal. An LCD consists of two glass panels, with the liquid crystal material sand witched in between them. The inner surface of the glass plates are coated with transparent electrodes which define the character, symbols or patterns to be displayed polymeric layers are present in between the electrodes and the liquid crystal, which makes the liquid crystal molecules to maintain a defined orientation angle. One each polarizer is pasted outside the two glass panels. These polarizers would rotate the light rays passing through them to a definite angle, in a particular direction.

When the LCD is in the off state, light rays are rotated by the two polarizers and the liquid crystal, such that the light rays come out of the LCD without any orientation, and hence the LCD appears transparent. When sufficient voltage is applied to the electrodes, the liquid crystal molecules would be aligned in a specific direction. The light rays passing through the LCD would be rotated by the polarizers, which would result in activating / highlighting the desired characters.

The LCDs are lightweight with only a few millimeters thickness. Since the LCD's consume less power, they are compatible with low power electronic circuits, and can be powered for long durations. The LCDs don't generate light and so light is needed to read the display. By using backlighting, reading is possible in the dark. The LCD's have long life and a wide operating temperature range. Changing the display size or the layout size is relatively simple which makes the LCD's more customer friendly. The LCDs used exclusively in watches, calculators and measuring instruments are the simple seven-segment displays, having a limited amount of numeric data. The recent advances in technology have resulted in better legibility, more information displaying capability and a wider temperature range. These have resulted in the LCDs being extensively used in telecommunications and entertainment electronics. The LCDs have even started replacing the cathode ray tubes (CRTs) used for the display of text and graphics, and also in small TV applications.

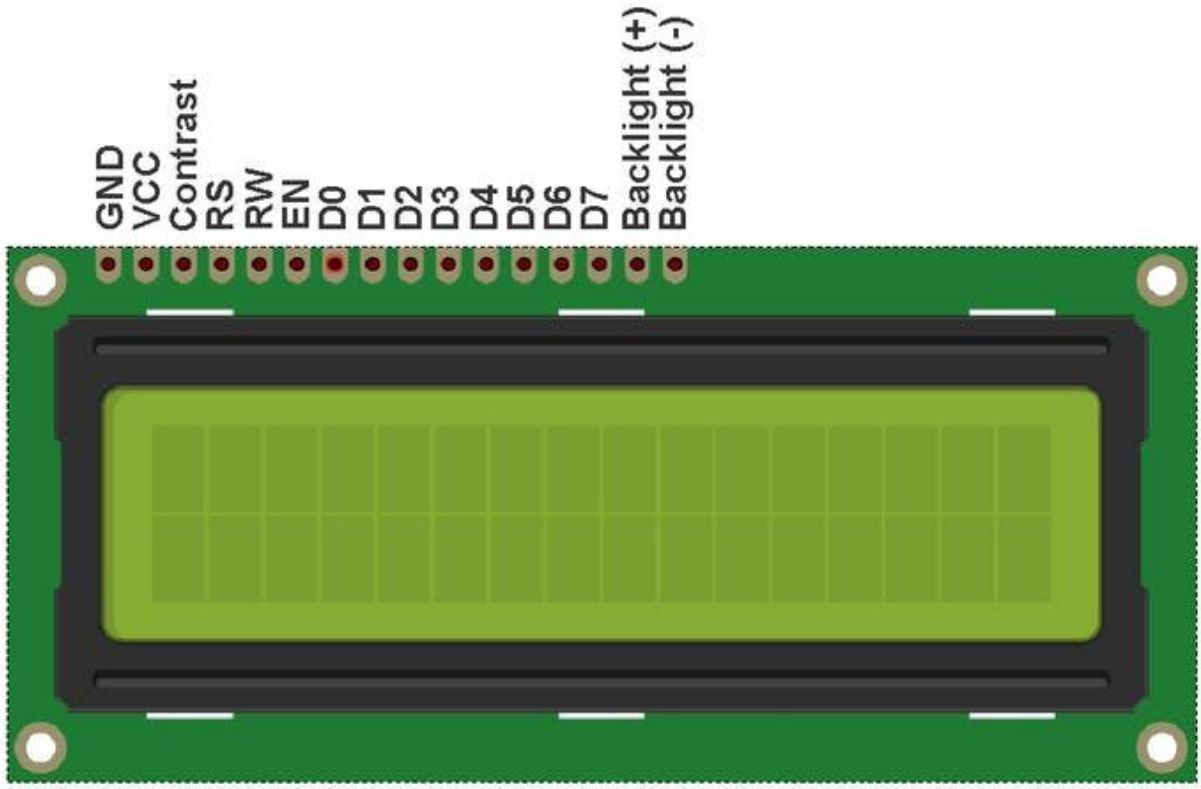


FIGURE 6: LIQUID CRYSTAL DISPLAY

ACCELEROMETER SENSOR:

An accelerometer is an electronic sensor that measures the acceleration forces acting on an object, in order to determine the object's position in space and monitor the object's movement. Acceleration, which is a vector quantity, is the rate of change of an object's velocity (velocity being the displacement of the object divided by the change in time). There are two types of acceleration forces: static forces and dynamic forces. Static forces are forces that are constantly being applied to the object (such as friction or gravity). Dynamic forces are "moving" forces applied to the object at various rates (such as vibration, or the force exerted on a cue ball in a game of pool). This is why accelerometers are used in automobile collision safety systems, for example. When a car is acted on by a powerful dynamic force, the accelerometer (sensing a rapid deceleration) sends an electronic signal to an embedded computer, which in turn deploys the airbags. There are three different types of accelerometers, and they are each designed to efficiently function in their intended environments. The three types are: piezoelectric, piezo resistance and capacitive. A piezoelectric accelerometer utilizes the piezoelectric effect (piezoelectric materials produce electricity when put under physical stress) to sense change in acceleration. Piezoelectric accelerometers are most commonly used in vibration and shock measurement. Piezo resistance accelerometers are much less sensitive than piezoelectric accelerometers, and they are better suited to vehicle crash testing. A piezo resistance accelerometer increases its resistance in proportion to the amount of pressure applied to it.

The third and most commonly used type of accelerometer is the capacitive accelerometer. Capacitive accelerometers use change in electrical capacitance to determine an object's acceleration. When the sensor undergoes acceleration, the distance between its capacitor plates changes as the diaphragm of the sensor moves.

Most accelerometers are miniscule, and they are often referred to as Micro-Electro-Mechanical Systems (MEMS) accelerometers. Because of their size and affordability, they are embedded in a myriad of hand-held electronic devices (such as phones, tablets, and video game controllers). In phones and tablets, the accelerometer is responsible for "flipping" the screen when the device is rotated. Accelerometers are also used by zoologists (to track the movement of animals in the wild), engineers (especially in collision experiments) and factories (to monitor the vibration of machinery).

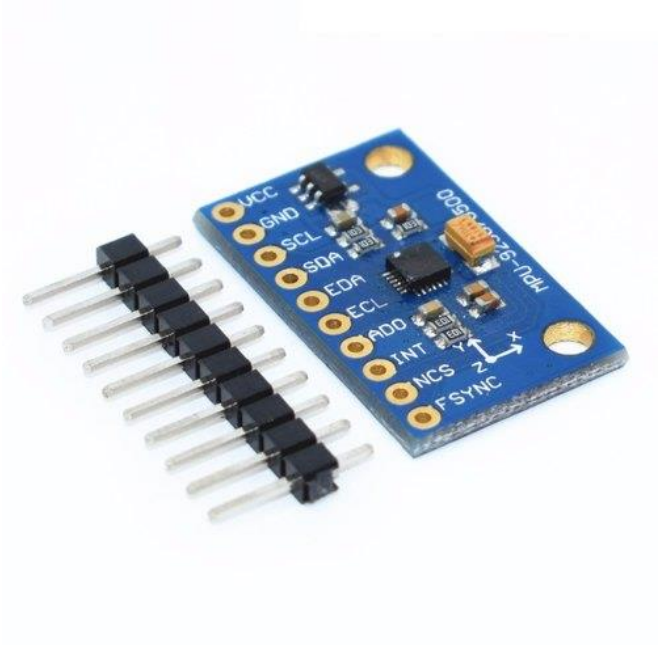


FIGURE 7: ACCELEROMETER SENSOR

PUSH BUTTONS:

A Push Button switch is a type of switch which consists of a simple electric mechanism or air switch mechanism to turn something on or off. Depending on model they could operate with momentary or latching action function. The button itself is usually constructed of a strong durable material such as metal or plastic. Push Button Switches come in a range of shapes and sizes. We have a selection of push button switches here at Herga. Push button switches are used throughout industrial and medical applications and are also recognisable in everyday life. For uses within the Industrial sector, push buttons are often part of a bigger system and

are connected through a mechanical linkage. This means that when a button is pressed it can cause another button to release.

FIGURE 8: PUSH BUTTONS

GSM:

GSM (Global System for Mobile communication) is a digital mobile network that is widely used by mobile phone users in Europe and other parts of the world. GSM uses a variation of time division multiple access (TDMA) and is the most widely used of the three digital wireless telephony technologies: TDMA, GSM and code-division multiple access (CDMA). GSM digitizes and compresses data, then sends it down a channel with two other streams of user data, each in its own time slot. It operates at either the 900 megahertz (MHz) or 1,800 MHz frequency band. GSM, together with other technologies, is part of the evolution of wireless mobile telecommunications that includes High-Speed Circuit-Switched Data (HSCSD), General Packet Radio Service (GPRS), Enhanced Data GSM Environment (EDGE) and Universal Mobile Telecommunications Service (UMTS).

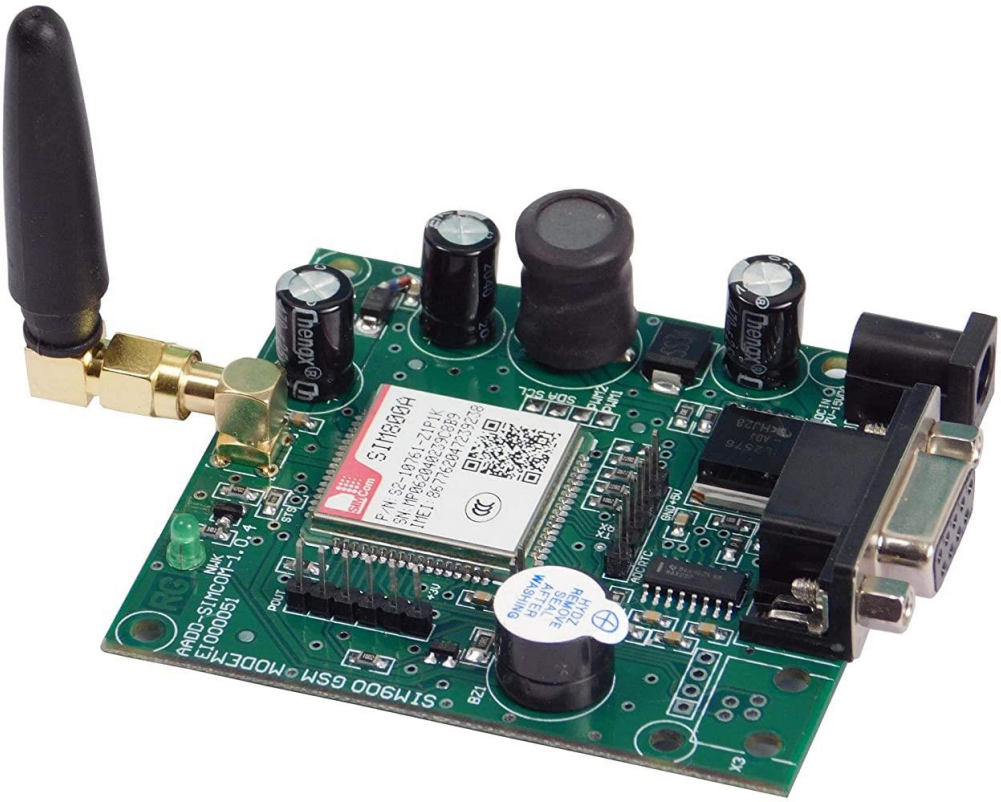


FIGURE 9: GSM

BUZZER:

An audio signalling device like a beeper or buzzer may be electromechanical or piezoelectric or mechanical type. The main function of this is to convert the signal from audio to sound. Generally, it is powered through DC voltage and used in timers, alarm devices, printers, alarms, computers, etc. Based on the various designs, it can generate different sounds like alarm, music, bell & siren.

The pin configuration of the buzzer is shown below. It includes two pins namely positive and negative. The positive terminal of this is represented with the '+' symbol or a longer terminal. This terminal is powered through 6Volts whereas the negative terminal is represented with the '-'symbol or short terminal and it is connected to the GND terminal.

FIGURE 10: BUZZER

BATTERY:

A 12 volt battery is an irregular battery used in specific electronic applications. Of all the types of batteries, the 12 volt battery is one that looks very different depending on its use. In some ways, it is one of the most diverse of all batteries. It can be large or small, heavy or light. In some cases, they may look nearly like regular AA batteries. One of the most common uses of a 12 volt battery is for transportation applications, such as in cars and boats. In these cases, the battery may be able to be recharged as current is only needed to start the vehicle. After that, the alternator takes over and runs the electrical system, if it is functioning properly. The alternator also puts a charge back into the battery. A 12 volt battery is available as a non-rechargeable alkaline battery or in rechargeable forms. This battery is often used in various outdoor applications that require greater amounts of energy in order to operate as desired. However, it may be recommended to use a rechargeable battery if the electronic item is one that is used regularly. Non-rechargeable batteries can become quite expensive if they must be purchased regularly. Many manufacturers have attempted over the years to reduce the size of the 12 volt battery to one that is easily portable, as some outdoors enthusiasts like to use them to run things like fish finders. While such electronic devices can be run using an electrical system onboard a boat, there are some anglers who do not have an onboard electrical system. Smaller craft with smaller outboard motors may have a pull start and therefore have no need of an electrical system for its main motor. Therefore, running things like fish finders and oxygenators off a separate battery is the only way to do it.

FIGURE 11: 12V BATTERY

4.2. SOFTWARE MATERIALS:

- Arduino IDE

ARDUINO IDE:

Arduino IDE is an open-source software, designed by Arduino.cc and mainly used for writing, compiling & uploading code to almost all Arduino Modules. It is an official Arduino software, making code compilation too easy that even a common person with no prior technical knowledge can get their feet wet with the learning process. It is available for all operating systems i.e. MAC, Windows, Linux and runs on the Java Platform that comes with inbuilt functions and commands that play a vital role in debugging, editing and compiling the code. A range of Arduino modules available including Arduino Uno, Arduino Mega, Arduino Leonardo, Arduino Micro and many more. Each of them contains a microcontroller on the board that is actually programmed and accepts the information in the form of code. The main code, also known as a sketch, created on the IDE platform will ultimately generate a Hex File which is then transferred and uploaded in the controller on the board. The IDE environment mainly contains two basic parts: Editor and Compiler where former is used for writing the required code and later is used for compiling and uploading the

code into the given Arduino Module. This environment supports both C and C++ languages.

4.3. PROGRAMMING LANGUAGE & CODE:

- Embedded C

EMBEDDED C PROGRAMMING:

It is an extension to the traditional C programming language, that is used in embedded systems. The embedded C programming language uses the same syntax and semantics as the C programming language. The only extension in the Embedded C language from normal C Programming Language is the I/O Hardware Addressing, fixed-point arithmetic operations, accessing address spaces, etc. The C Programming Language, developed by Dennis Ritchie in the late 60's and early 70's, is the most popular and widely used programming language. The C Programming Language provided low level memory access using an uncomplicated compiler (a software that converts programs to machine code) and achieved efficient mapping to machine instructions. The C Programming Language became so popular that it is used in a wide range of applications ranging from Embedded Systems to Super Computers. Embedded C Programming Language, which is widely used in the development of Embedded Systems, is an extension of C Program Language. The Embedded C Programming Language uses the same syntax and semantics of the C Programming Language like main function, declaration of datatypes, defining variables, loops, functions, statements, etc. The extension in Embedded C from standard C Programming Language include I/O Hardware Addressing, fixed point arithmetic operations, accessing address spaces, etc.

CODE:

```
#include<LiquidCrystal.h>
#include <SoftwareSerial.h>
#include <Wire.h>
LiquidCrystal lcd(13, 12, 11, 10, 9, 8);

SoftwareSerial gsm_g(2, 3);

#define splash splash1

#define acl A0
#define key1 A1
#define key2 A2
#define key3 A3
#define buz 4


String latout = "12.8749694", lonout = "80.2253954"; //12.8749694,80.2253954
String smsmsg, gpsval, text;
String usernum = "+919360607932";
int C_Status;
int C_tim = 10;
int Comp_tim = 60, comoff_s;
float cc_tim = C_tim;
float em_tim, en_tim;
const int ACC = 0x68; // I2C address of the MPU-6050
unsigned int AcX, AcY, AcZ;
int fStatus, f_Status;

int acl_Status, key_Status;
int acl_r, k1_r, k2_r, k3_r;
int alert_Status;


void setup() {

  // put your setup code here, to run once:
  Serial.begin(9600);
  pinMode(key1, INPUT_PULLUP);
  pinMode(key2, INPUT_PULLUP);
  pinMode(key3, INPUT_PULLUP);
  pinMode(acl, INPUT);
```

```
  pinMode(key2, INPUT_PULLUP);
  pinMode(key3, INPUT_PULLUP);
  pinMode(acl, INPUT);
  pinMode(buz, OUTPUT);


  gsm_g.begin(9600);
  Wire.begin();
  Wire.beginTransmission(0x68);
  Wire.write(0x6B);  // PWR_MGMT_1 register
  Wire.write(0);     // set to zero (wakes up the MPU-6050)
  Wire.endTransmission(true);
  LcDSet();
    SendSmS( usernum, "GSM Initialized!!!", "", "Sending SMS.....");


}
void LcDSet() {

  lcd.begin(16, 2);
  splash(0, "Two Wheeler");
  splash(1, "Safety");

  delay(2000);
  lcd.clear();

}

void(* resetFunc) (void) = 0;
void loop() {
  // put your main code here, to run repeatedly://<500>800
  f_Status = getAcc();

  acl_r = digitalRead(acl);
  k1_r = digitalRead(key1);
  k2_r = digitalRead(key2);
  k3_r = digitalRead(key3);
  Serial.println(acl_r);
  if (k1_r == 0 and k2_r == 0) {
    key_Status = 1;

    Serial.println("Engine started");
```

```
}
else {
  key_Status = 0;
}
if (!acl_r ) {
  acl_Status = 1;
} else {
  acl_Status = 0;
}
if (acl_Status == 1) {
  splash (0, "Alcohol");
  splash (1, "Detected");
}
else if (key_Status == 0 and en_tim == 0) {

  splash(0, "Please Wear");
  splash(1, "YouR Helmet");
}
else if (key_Status == 1 and en_tim == 0 ) {

  //    analogWrite(eng, 250);

  splash(0, "Engine");
  splash(1, "Started");
  delay(1000);
  splash(0, "Have A");
  splash(1, "Safe Ride");
  if (f_Status == 1) {
    splash(0, "Fall Detected");
    digitalWrite(buz, HIGH);

    splash(0, "");
    en_tim = 1;
    delay(100);

  }
}

if (k3_r == 0) {
  splash(0, "Glad You're");
  splash(1, "Safe");
  en_tim = 0;
```

```
    splash(0, "Glad You're");
    splash(1, "Safe");
    en_tim = 0;
    em_tim = 0;

    digitalWrite(buz, LOW);
    delay(1000);
    resetFunc();
  }
  if (en_tim == 1) {

    lcd.setCursor(0, 1);
    lcd.print("                ");
    lcd.setCursor(12, 1);
    lcd.print(em_tim);
    em_tim += 0.20;
    if (em_tim > C_tim) {
      alert("Accident Detected");
      em_tim = 0;
      C_tim = 60;
      alert_Status = 0;
    }

  }

  delay(100);
}

void alert(String msg) {
  if (alert_Status == 0) {

    gpsval = "  https://maps.google.com/maps?q=";
    gpsval += latout;
    gpsval += ",";
    gpsval += lonout;
    smsmsg = "Emergency ";
    smsmsg += msg;
    Serial.println(smsmsg + gpsval);
    SendSmS( usernum, smsmsg, gpsval, "Sending SMS.....");
    alert_Status = 1;
  }
}
```

```
int getAcc() {
  Wire.beginTransmission(ACC);
  Wire.write(0x3B);  // starting with register 0x3B (ACCEL_XOUT_H)
  Wire.endTransmission(false);
  Wire.requestFrom(ACC, 14, true); // request a total of 14 registers
  AcX = Wire.read() << 8 | Wire.read(); // 0x3B (ACCEL_XOUT_H) & 0x3C (ACCEL_XOUT_L)
  AcY = Wire.read() << 8 | Wire.read(); // 0x3D (ACCEL_YOUT_H) & 0x3E (ACCEL_YOUT_L)
  AcZ = Wire.read() << 8 | Wire.read(); // 0x3F (ACCEL_ZOUT_H) & 0x40 (ACCEL_ZOUT_L)




  if (AcX > 49000 && AcX < 58000)
  {

    fStatus = 1;

    delay(20);


  }

  else if (AcX > 7000 && AcX < 15000)
  {
    fStatus = 1;
    delay(20);

  }

  else if (AcY > 7000 && AcY < 15000  )
  {

    fStatus = 1;
    delay(20);

  }


  else if (AcY > 49000 && AcY < 57000 )
  {
```

```
  else if (AcY > 7000 && AcY < 15000  )
  {

    fStatus = 1;
    delay(20);

  }


  else if (AcY > 49000 && AcY < 57000 )
  {
    fStatus = 1;
    delay(20);
  }
  else {
    fStatus = 0;
    delay(20);
  }
  Serial.print("X = "); Serial.print(AcX);
  Serial.print(" | Y = "); Serial.println(AcY);

  return fStatus;
  delay(333);
}


void SendSmS( String NuM, String MsG , String gps, String LcD)
{
  lcd.setCursor(0, 0);
  lcd.print(LcD);
  lcd.setCursor(0, 1);
  lcd.print(".");
  gsm_g.println("AT");
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  gsm_g.println("AT+CMGF=1");    //Sets the gsm_g Module in Text Mode
  lcd.print(".");
  delay(1000);
  lcd.print("."); // Delay of 1000 milli seconds or 1 second
  gsm_g.println("AT+CMGS=\"" + NuM + "\"\r"); //
```

```
  lcd.setCursor(0, 0);
  lcd.print(LcD);
  lcd.setCursor(0, 1);
  lcd.print(".");
  gsm_g.println("AT");
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  gsm_g.println("AT+CMGF=1");    //Sets the gsm_g Module in Text Mode
  lcd.print(".");
  delay(1000);
  lcd.print("."); // Delay of 1000 milli seconds or 1 second
  gsm_g.println("AT+CMGS=\"" + NuM + "\"\r"); //
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  //  gsm_g.print("Hello ");
  gsm_g.print(MsG);
  gsm_g.print(gps);
  delay(1000);
  lcd.print(".");
  gsm_g.println((char)26);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  delay(1000);
  lcd.print(".");
  lcd.clear();
}
```

4.4. WORKING METHOD:

The Arduino UNO consists of 4 ports that are Analog, Digital, Transmitter & receiver and power supply port. The Analog port is connected to the Accelerometer sensor. The Digital port is connected to the LCD and buzzer. The Transmitter & Receiver port is connected to GSM because it transmits the information from the sim to the mobile number which we have fed. The power supply port is connected with all the wires that connects to the battery. The two switches which are connected with brown and black color wires are assumed as the switches set inside the helmet and are called as smart switch. There is another sensor used to detect if the person has consumed alcohol and the level of alcohol consumed. There is a button which is connected with yellow and green wire which is an emergency button used to stop the information from being transmitted from the sim to the numbers fed. In case the accident is not big then the person can press this emergency button within 1minute to stop the information from being transmitted from the GSM. We use a 12v battery which gets converted to 9V then 5V in the power supply board and finally 5V is supplied to the Arduino UNO. The accelerometer sensor detects if there is any accident taken place and then transmits the information to the GSM and the GSM transmits the information to the receiver stating the accident has taken place and the link of the location. The accelerometer will be able to identify if it is an accident or turning of the vehicle. The alcohol detection sensor detects if the person as consumed alcohol and does not allow the vehicle to move. If we wear the helmet only the motor will start or else it won't start.

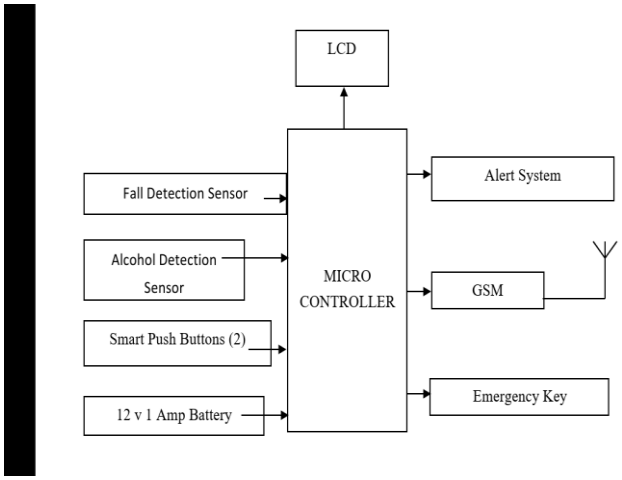
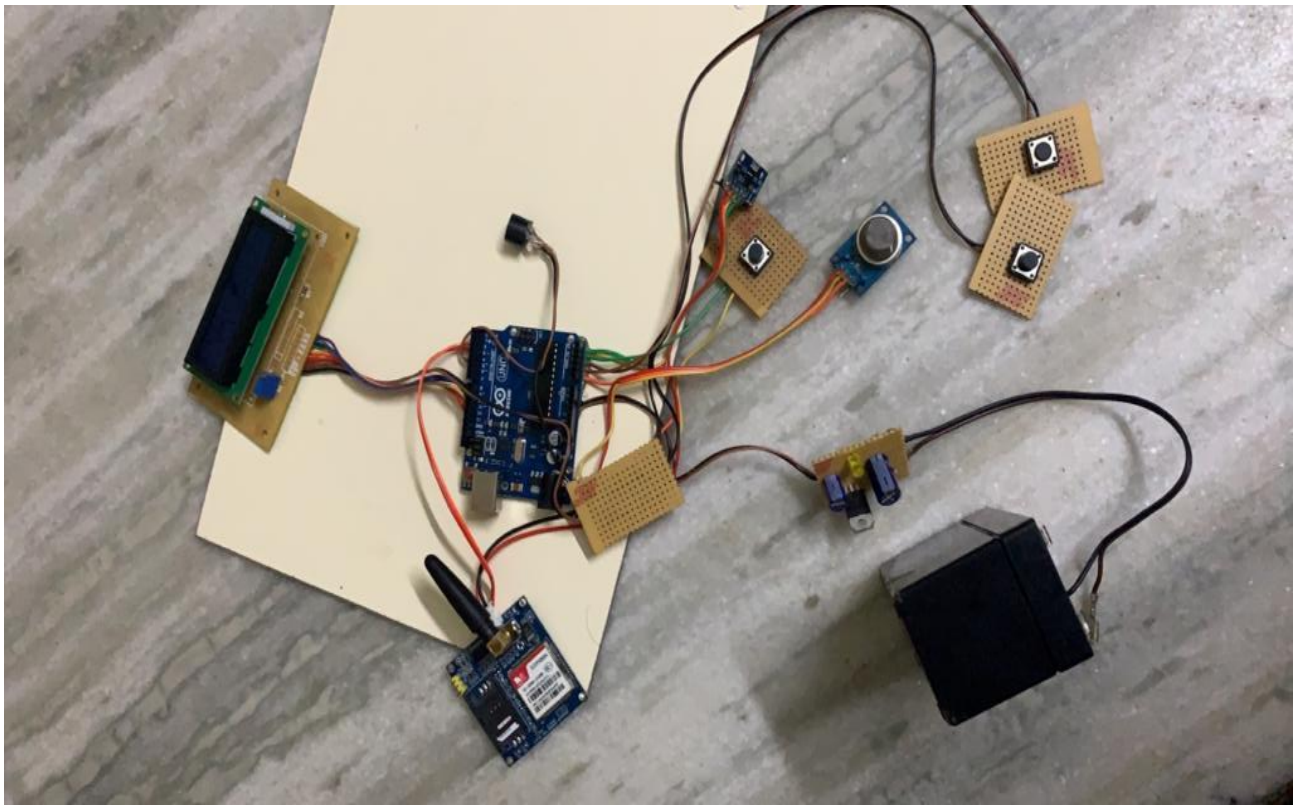


FIGURE 12: BLOCK DIAGRAM

CHAPTER-5

RESULT

- This project is very helpful for the ambulance or people to save lives of other people during an accident and being on time to the accident place.
- This is also useful in order to make sure that people under alcohol do not ride their bikes and in order to avoid accidents due to alcohol consumption.



CHAPTER-6

SUMMARY & CONCLUSION

- This project is very helpful for the ambulance or people to save lives of other people during an accident and being on time to the accident place.
- This is also useful in order to make sure that people under alcohol do not ride their bikes and in order to avoid accidents due to alcohol consumption.

REFERENCES:

- Prudhvi Raj R, Sri Krishna Kanth, BhargavAdityaBharath K, (2014), “Smart-tec Helmet” Electrical and Electronics Engineering, GITAM University,Rushikonda, Visakhapatnam, India. Advance in Electronic and Electric Engineering 4: 493-498.

- Behr, C.J., Kumar, A., Hancke, G.P “ A Smart Helmet for Air Quality and Hazardous Event Detection for the Mining Industry” Proceedings of the IEEE International Conference on Industrial Technology 2016-May,7475079, pp. 2026-2031.

- Sreenithy Chandran, Sneha Chandrasekar, N Edna Elizabeth “Konnect: An Internet of Things(IoT) based smart helmet for accident detection and notification” 2016 IEEE Annual India Conference (INDICON).

- Rashmi Vashisth, Sanchit Gupta, Aditya Jain, Sarthak Gupta, Sahil, Prashant Rana “Implementation and analysis of smart helmet” 2017 4th International Conference on Signal Processing, Computing and Control (ISPCC).

- D. Selvathi, P. Pavithra, T. Preethi “Intelligent Transportation System for Accident Prevention and Detection” 2017 International Conference on Intelligent Computing and Control Systems (ICICCS).

- Archana D, Boomija G, Manisha J, Kalaiselvi V. K. G. “Mission On! Innovations in Bike Systems to Provide a Safe Ride Based on IOT ”2017 2nd International Conference on Computing and Communications Technologies (ICCCT).

- Sayan Tapadar, Shinjini Ray; Himadri, Nath Saha; Arnab, Kumar Saha,Robin Karlose “ Accident and alcohol detection in bluetooth enabled smart helmets for motorbikes” 2018 IEEE 8th Annual Computing and Communication Workshop and Conference.