1. INTRODUCTION

1.1 OVERVIEW

SAFE DRIVING USING ALCOHOL DETECTION is a process and effort to overcome the day to day problem of drink and drive. The idea is to mount an alcohol sensor (MQ-3) on the steering wheel such that if the driver is drunk over the threshold level then the power of the car is cut of such that in that case car will not be able to move.

These roads make a vital contribution to the India's economy. According to a government report, road accidents in India killed 1,**34,000** people in 2010 (an average of **336** a day). Accidents due to drunken driving are a major problem in India.

DUI, or driving under the influence in India is a criminal offence under the Motor Vehicle Act 1988. Section 185 of the Act states that any person who is found to have alcohol exceeding 30 mg per 100 ml of blood in their system is breaking the law. This may sound like a small amount, but even with 0.03% intoxication level, a person is seven times more likely to be involved in a vehicle crash than a person who has not consumed any alcohol.

Police in India generally enforce drunk driving laws by a “selective breath checkpoint” strategy. Officers from local police stations set up fixed roadblock checkpoints that force passing vehicles to slow down, during which officers can order drivers to pull over for a brief conversation and potential breathalyzer test. If the driver is drunk (a blood alcohol concentration level of more than 30 mg of alcohol per 100 mL of blood), police confiscate the vehicle and order the driver to appear in court for sentencing (varying from monetary fines to imprisonment).

NEED FOR IMPROVEMENT IN ARDUINO

By 2030, almost two-third of the world’s population will be living in cities. This fact requires the development for automation of health Monitoring System, people’s health is a key issue.

Effectively managing the health monitoring system in developed countries is important. It will consume more time for monitoring patients individually. To make it smart and automated health monitoring system is important.

It improves the monitoring of patient’s health and his past related diseases in a record. It is easy for doctor to view the details of the patient health history and related drugs consumed by him and prescribed by other doctors.

1.2 FEATURES OF SAFE DRIVING USING ALCOHOL DETECTION

- To overcome the problem of “Drunk and Drive”.
- To provide “public and personal” safety.

1.3 ADVANTAGES OF SAFE DRIVING USING ALCOHOL DETECTION

- The aim of this idea is to reduce the inconvenience to the people by traffic jams.
- Reducing the workload of cops.
- Know the amount of alcohol is consumed by the person.

1.4 MAIN EQUIPMENTS USED IN SAFE DRIVING USING ALCOHOL DETECTION

1.4.1 ALCOHOL SENSOR

- The alcohol sensor we will use is the MQ-3 sensor. This is a sensor that is not only sensitive to alcohol, particularly ethanol, which is the type of alcohol which is found in wine, beer, and liquor.
- This type of sensor circuit can be used as a breathalyzer to check a person's blood alcohol level. Just as we exhale carbon dioxide when we breathe out, we also will breathe out some alcohol if we have alcohol in our blood. Any alcometer device can measure this alcohol content.
- The more ethanol in your blood, the more there is in the air on exhalation. This alcohol content gives a good indication for if a person is drunk and how drunk they are.
- The amount of alcohol exhaled into the air is proportional to the amount of alcohol which will be found in a person's blood. Alcometers use a built-in formula to estimate blood alcohol content from exhaled air alcohol content.

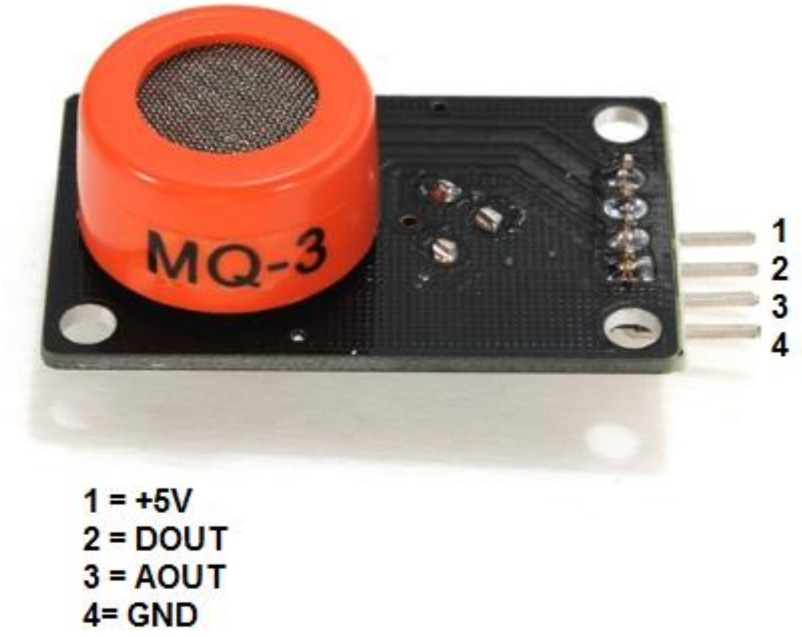


Fig. 1.4.1 Alcohol Sensor

1.4.2 ARDUINO MEGA BOARD

Arduino is a software company, project, and user community that designs and manufactures computer open-source hardware ,open-source software, and microcontroller-based kits for building digital devices and interactive objects that can sense and control physical devices .

The project is based on microcontroller board designs, produced by several vendors, using various microcontrollers. These systems provide sets of digital and analog I/O pins that can interface to various expansion boards (termed *shields*) and other circuits. The boards feature serial communication interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers. For programming the microcontrollers, the Arduino project provides an integrated development environment(IDE) based on a programming language named *Processing*, which also supports the languages C and C++.

The first Arduino was introduced in 2005, aiming to provide a low cost, easy way for novices and professionals to create devices that interact with their environment using sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.

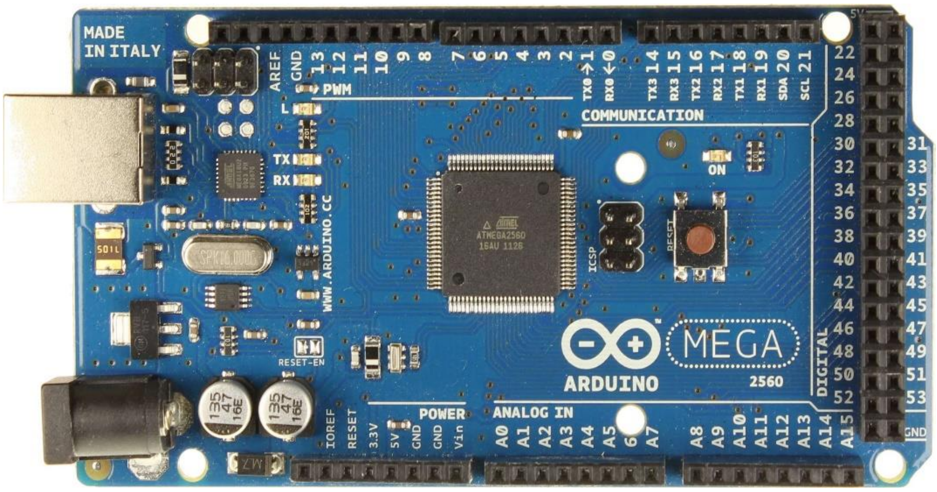


FIG 1.4.2 ARDUINO MEGA BOARD

SOFTWARE OF ARDUINO

The Arduino project provides the Arduino integrated development environment (IDE), which is a cross-platform application written in the programming language Java. It originated from the IDE for the languages Processing and Wiring. It is designed to introduce programming to artists and other newcomers unfamiliar with software development. It includes a code editor with features such as syntax highlighting, brace matching, and automatic indentation, and provides simple one-click mechanism to compile and load programs to an Arduino board[3]. A program written with the IDE for Arduino is called a "sketch".

The Arduino IDE supports the languages C and C++ using special rules to organize cod These systems provide sets of digital and analog I/O pins that can interface to various expansion boards (termed *shields*) and other circuits. The boards feature serial communication interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers. For programming the microcontrollers, the Arduino project provides an integrated development environment(IDE) based on a programming language named *Processing*, which also supports the languages C and C++.

The first Arduino was introduced in 2005, aiming to provide a low cost, easy way for novices and professionals to create devices that interact with their environment using sensors and actuators .

1.4.3 AC MOTOR

The motor that converts the alternating current into mechanical power by using an electromagnetic induction phenomenon is called an AC motor. This motor is driven by an alternating current. The stator and the rotor are the two most important parts of the AC motors. The stator is the stationary part of the motor, and the rotor is the rotating part of the motor. The AC motor may be single phase or three phases.`=

The three phase AC motors are mostly applied in the industry for bulk power conversion from electrical to mechanical. For small power conversion, A single-phase AC motors are mostly used. A single-phase AC motor is nearly small in size, and it provides a variety of services in the home, office, business concerns, factories, etc. Almost all the domestic appliances such as refrigerators, fans, washing machine, hair dryers, mixers, etc., use single phase AC motor.

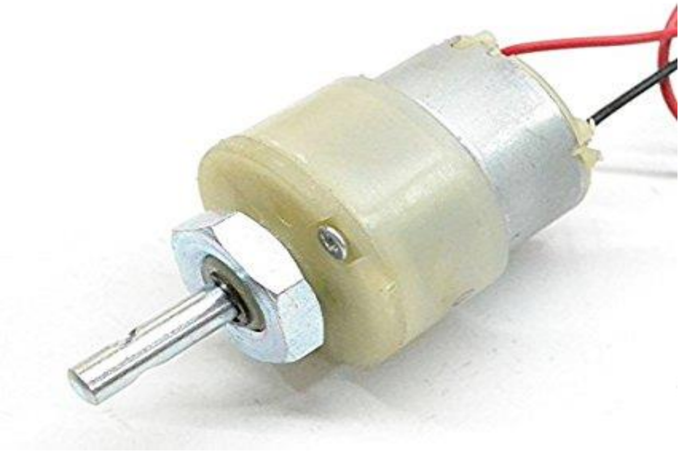


Fig 1.4.3 AC Motor

1.4.4 JUMP WIRES

Jump wires (also called jumper wires) for solderless breadboarding can be obtained in ready-to-use jump wire sets or can be manually manufactured. The latter can become tedious work for larger circuits. Ready-to-use jump wires come in different qualities, some even with tiny plugs attached to the wire ends. Jump wire material for ready-made or homemade wires should usually be 22 AWG (0.33 mm^2) solid copper, tin-plated wire - assuming no tiny plugs are to be attached to the wire ends. The wire ends should be stripped $^3/_{16}$ to $^5/_{16}$ in (4.8 to 7.9 mm). Shorter stripped wires might result in bad contact with the board's spring clips (insulation being caught in the springs). Longer stripped wires increase the likelihood of short-circuits on the board. Needle-nose pliers and tweezers are helpful when inserting or removing wires, particularly on crowded boards.

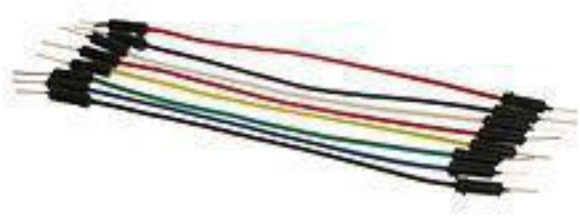


Fig. 1.4.4 JUMPER WIRES

Differently colored wires and color-coding discipline are often adhered to for consistency. However, the number of available colors is typically far fewer than the number of signal types or paths. Typically, a few wire colors are reserved for the supply voltages and ground (e.g., red, blue, black), some are reserved for main signals, and the rest are simply used where convenient.

1.4.5 CHASSIS

- In an electronic device, the chassis consists of a frame or other internal supporting structure on which the circuit boards and other electronics are mounted.
- In some designs, such as older sets, the chassis is mounted inside a heavy, rigid cabinet, while in other designs such as modern computer cases, lightweight covers or panels are attached to the chassis.
- The combination of chassis and outer covering is sometimes called an enclosure.

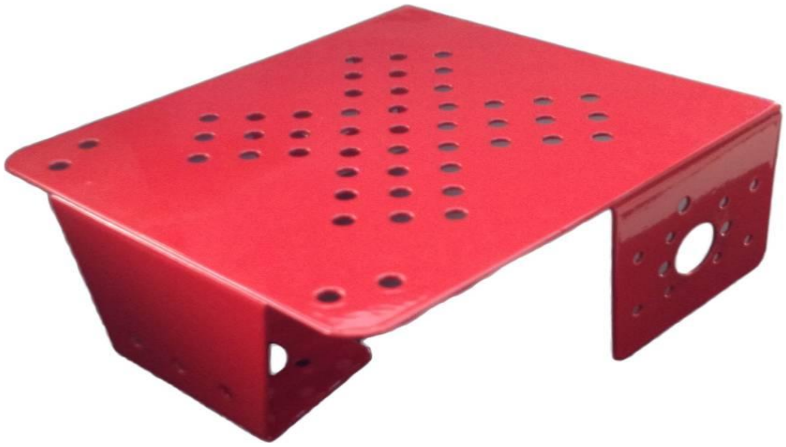


Fig. 1.4.5 CHASSIS

1.4.6 MOTOR DRIVER:

The most common method to drive DC motors in two directions under control of a computer is with an H-bridge motor driver. H-bridges can be built from scratch with bi-polar junction transistors (BJT) or with field effect transistors (FET), or can be purchased as an integrated unit in a single integrated circuit package such as the L293. The L293 is simplest and inexpensive for low current motors, For high current motors, it is less expensive to build your own H-bridge from scratch.

ITP Physical Computing has a terrific tutorial on using an Arduino and an L293 to control a bi-directional motor.

The Twin Cities Robotics Club has an *excellent* tutorial on H-bridges, and complete detail on how to build your own $5.00 H-bridge good for several amps. From the same source is a detailed tech note on PWM speed control of a motor using an H-bridge and a PIC microcontroller

The L293 is an integrated circuit motor driver that can be used for simultaneous, bi-directional control of two small motors. Small means small. The L293 is limited to 600 mA, but in reality, can only handle much small currents unless you have done some serious heat sinking to keep the case temperature down. Unsure about whether the L293 will work with your motor? Hook up the circuit and run your motor while keeping your finger on the chip. If it gets too hot to touch, you can't use it with your motor. (Note to ME2011 students: The L293 should be OK for your small motor but is not OK for your gear motor.)

The L293 comes in a standard 16-pin, dual-in line integrated circuit package. There is an L293 and an L293D part number. Pick the "D" version because it has built in fly back diodes to minimize inductive voltage spikes.

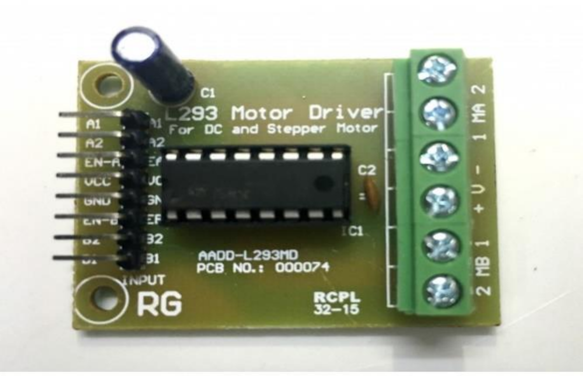


Fig. 1.4.6 MOTOR DRIVER L293

2. RELATED WORK

This method is developed in perspective for replacing the existing system which is a hand-held breath analyzer which displays the amount of alcohol consumed by the person. To display the amount of alcohol consumed it has to directly in contact with the person who is undergoing the test as he as to blow into a plastic straw which will be attached to the device.

For this to be conducted cops usually select a spot where they will set un barrel gates to let them only move from one lane where they will be conduction the test. In this process of checking they usually hold upcoming vehicles which leads to the traffic jams. Even though this much of preparations are conducted people escape it by seeing further and stopping their vehicle's in the fear of getting caught and taking rash decision where they will try to take detour or make a u turn and go against the traffic stream thus causing life threating condition foe both them and the upcoming people.

Even though cops practice this method people are mostly get caught if such scenario occur till which it could be to late so to overcome this it best to say "prevention is better than cure", So instead of catching them at particular sport why not let them start from the point it self which could prevent them from driving and coming on to the road its self.

- To overcome the problem of "Drunk and Drive".
- To reduce the work load from cops.
- To provide "public and personal" safety.

3. MOTIVATION

The motivation for this project comes when I was watching news in which at a police check point where they were testing for alcohol consumption some of the people who were drunk. Seeing this driver panicked, then rammed car for escaping in which 2 police men died and few other got injured

Problem was that dunked person will only get caught through a checkpoint where it could be already too late as he/she can get into accident before it. Instead of stooping them at particular check point it will be way better if we are able to stop them before hand at the starting point.

The Community Against Drunken Driving (CADD) said nearly 70 per cent of all fatalities are due to drunken driving, with the figure running between 44 per cent to 67 per cent in smaller cities. Despite prosecution of drunken driving having increased by about seven times in Delhi and 16 times in Mumbai since 2001, there has been no corresponding decrease in accidents and fatalities.

Prince Singhal of CADD noted that “24-hour availability of alcohol along National and State highways results in impulsive buying of alcohol and about 72 per cent of road accidents on National Highways”.

Stating that “the World Day for Remembrance of Road Accident Victims needs to be observed as a significant day especially in the Indian context as we record the highest road fatalities at 134,000 annually”, he said it should not be forgotten that “road deaths and injuries are sudden, violent, traumatic events, and their impact is long-lasting, often permanent”.

4. PROPOSED WORK

By the idea of “SAFE DRIVING USING ALCOHOL DETECTION” we are trying to reduce the number of accidents and loss of life and financial revenue. So, the idea is that to mount a (MQ3) Gas sensor on the steering wheel of the car such that when a person is drunk over the limit the electrical power is turn to low state and the car will not start up. By this we could feel a bit safer about our loved once.

ADVANTAGES

- Reduces death rate significantly.
- Reduce the traffic jams which could be caused by cops for checking.
- Reduce work load of cops.
- Able to know amount of alcohol consumed.

4.1. DESGIN METHODOLOGY

4.1.1 IoT

The Internet of things (IoT) is the network of physical devices, vehicles, home appliances, and other items embedded with electronics, software, sensors, actuators, and connectivity which enables these things to connect, collect and exchange data.

IoT involves extending Internet connectivity beyond standard devices, such as desktops, laptops, smartphones and tablets, to any range of traditionally dumb or non-internet-enabled physical devices and everyday objects. Embedded with technology, these devices can communicate and interact over the Internet, and they can be remotely monitored and controlled.

4.1.2 ARCHITECTURE

The below figure is the basic architecture of SAFE DRIVING USING ALCOHOL DETECTION. Data collects from sensors, sends to Arduino. If the value is greater than threshold value then it will display message exceeded.

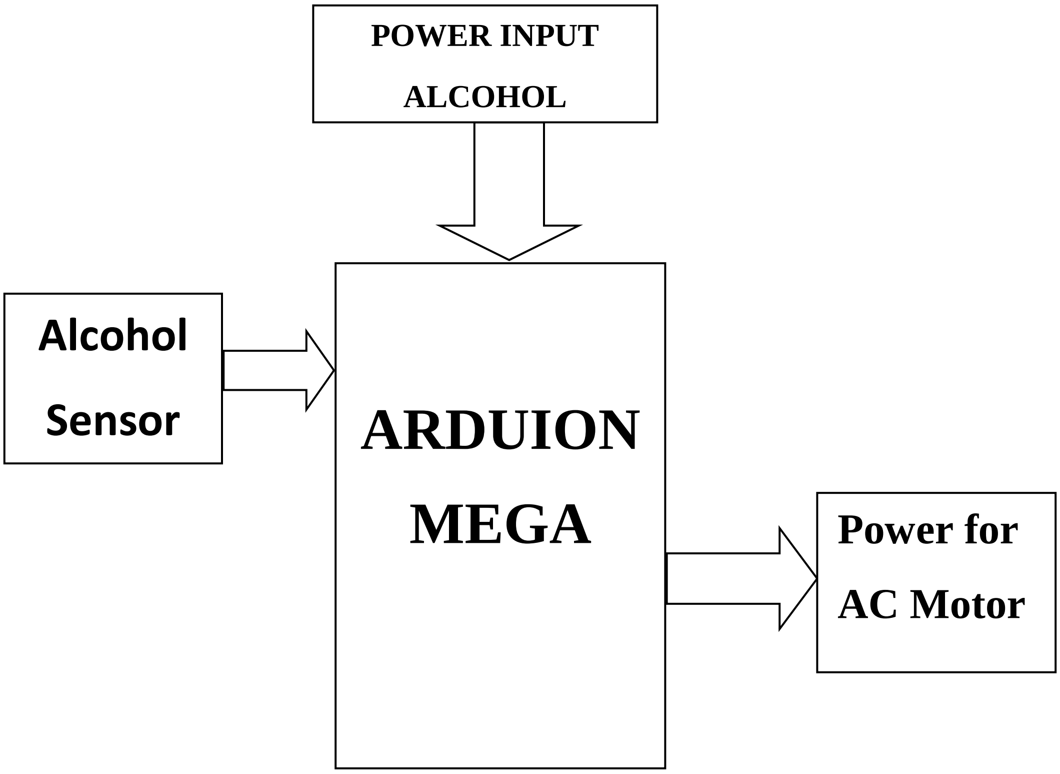
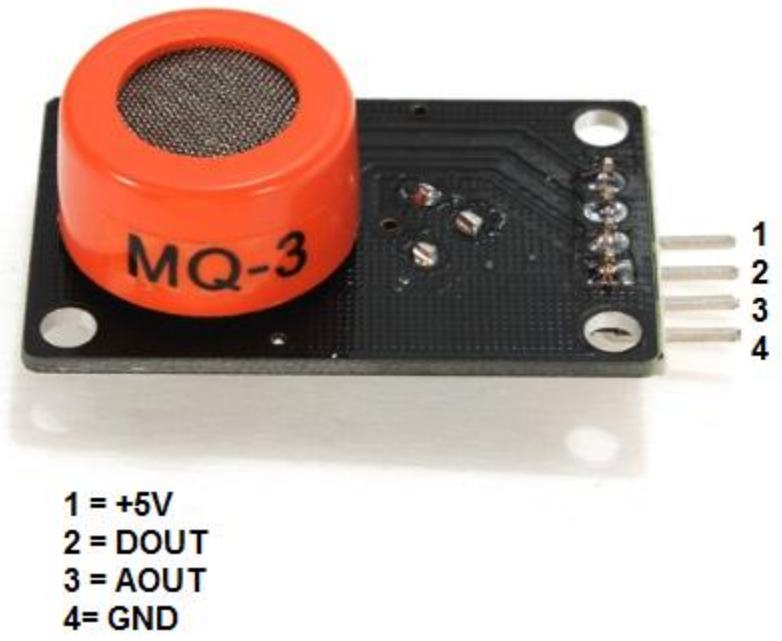


Fig. 4.1.2 System Architecture

4.2 MODULES

4.2.1 ALCOHOL SENSOR

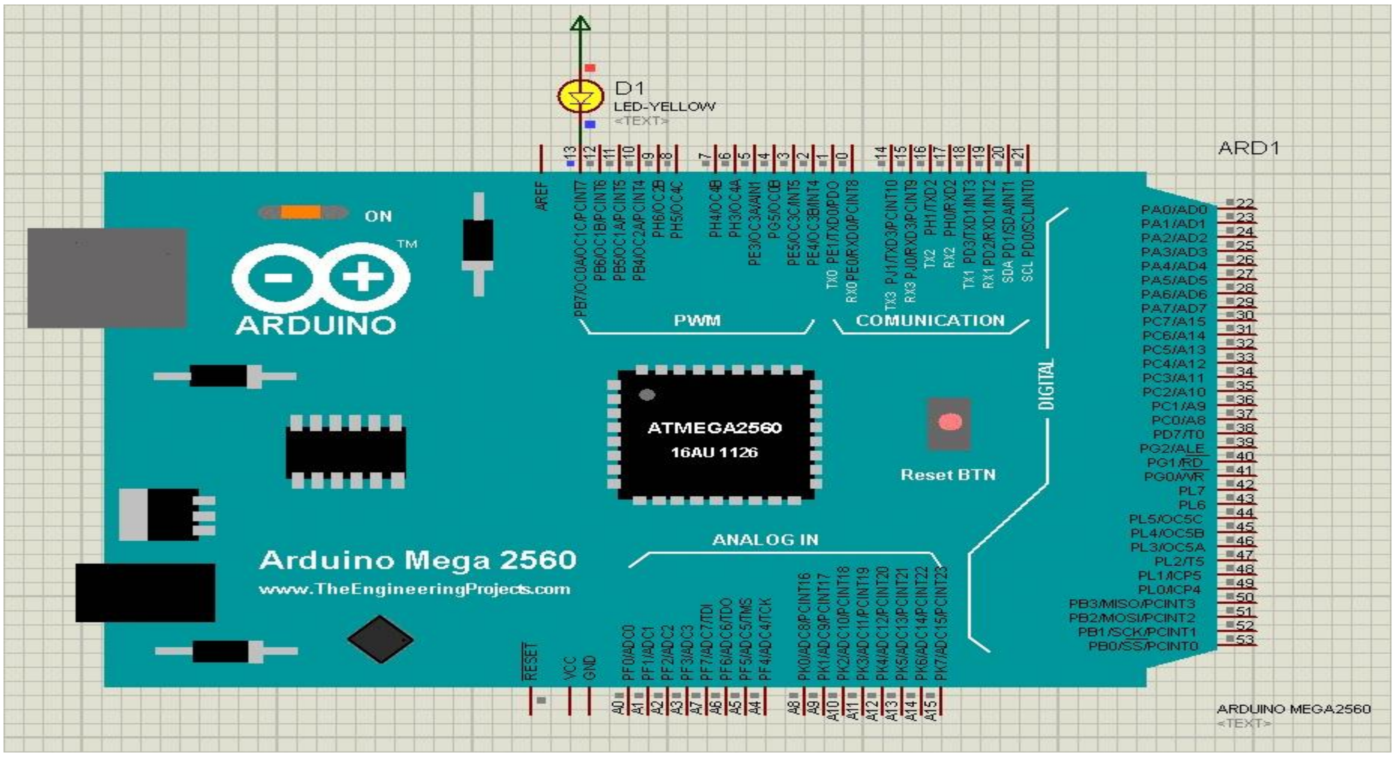
- The alcohol sensor we will use is the MQ-3 sensor. This is a sensor that is not only sensitive to alcohol, particularly ethanol, which is the type of alcohol which is found in wine, beer, and liquor.
- This type of sensor circuit can be used as a breathalyzer to check a person's blood alcohol level. Just as we exhale carbon dioxide when we breathe out, we also will breathe out some alcohol if we have alcohol in our blood. Any alcometer device can measure this alcohol content.
- The more ethanol in your blood, the more there is in the air on exhalation. This alcohol content gives a good indication for if a person is drunk and how drunk they are.
- The amount of alcohol exhaled into the air is proportional to the amount of alcohol which will be found in a person's blood. Alcometers use a built-in formula to estimate blood alcohol content from exhaled air alcohol content.



4.2.1 Alcohol Sensor

4.2.2 ARDUINO BOARD

The Arduino is a family of microcontroller boards to simplify electronic design, prototyping and experimenting for artists, hackers, hobbyists, but also many professionals. People use it as brains for their robots, to build new digital music instruments, or to build a system that lets your house plants tweet you when they're dry. Arduinos (we use the standard Arduino Mega) are built around an AT mega microcontroller — essentially a complete computer with CPU, RAM, Flash memory, and input/output pins, all on a single chip. Unlike, say, a Raspberry Pi, it's designed to attach all kinds of sensors, LEDs, small motors and speakers, servos, etc. directly to these pins, which can read in or output digital or analog voltages between 0 and 5 volts. The Arduino connects to your computer via USB, where you program it in a simple language (C/C++, similar to Java) from inside the free Arduino IDE by uploading your compiled code to the board. Once programmed, the Arduino can run with the USB link back to your computer, or stand-alone without it — no keyboard or screen needed, just power.



4.2.2 STRUCTURE OF ARDUINO MEGA BOARD

Looking at the board from the top down, this is an outline of what you will see (parts of the board you might interact with in the course of normal use are highlighted)

Digital Pins

In addition to the specific functions listed below, the digital pins on an Arduino board can be used for general purpose input and output via the pin Mode(), Digital Read(), and Digital Write() commands. Each pin has an internal pull-up resistor which can be turned on and off using digital Write() (w/ a value of HIGH or LOW, respectively) when the pin is configured as an input. The maximum current per pin is 40mA.

- **Serial: 0 (RX) and 1 (TX).** Used to receive (RX) and transmit (TX) TTL serial data. On the Arduino Decimal, these pins are connected to the corresponding pins of the FTDI USB-to-TTL Serial chip. On the Arduino BT, they are connected to the corresponding pins of the WT11 Bluetooth module. On the Arduino Mini and Lily Pad Arduino, they are intended for use with an external TTL serial module (e.g. the Mini-USB Adapter).
- **External Interrupts: 2 and 3.** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value. See the attach Interrupt() function for details.
- **PWM: 3, 5, 6, 9, 10, and 11** Provide 8-bit PWM output with the analog Write() function. On boards with an ATmega8, PWM output is available only on pins 9, 10, and 11.
- **BT Reset: 7.** (Arduino BT-only) Connected to the reset line of the Bluetooth module.
- **SPI: 10 (SS), 11 (MOSI), 12 (MISO), 13 (SCK).** These pins support SPI communication, which, although provided by the underlying hardware, is not currently included in the Arduino language.
- **LED: 13.** On the Decimal and Lily Pad, there is a built-in LED connected to digital pin 13. When the pin is HIGH value, the LED is on, when the pin is LOW, it's off.

Analog Pins

In addition to the specific functions listed below, the analog input pins support 10-bit analog-to-digital conversion (ADC) using the analog Read() function. Most of the analog inputs can also be used as digital pins: analog input 0 as digital pin 14 through analog input 5 as digital pin 19. Analog inputs 6 and 7 (present on the Mini and BT) cannot be used as digital pins.

- **I^{2}C: 4 (SDA) and 5 (SCL).** Support I^{2}C (TWI) communication using the Wire library (documentation on the Wiring website).

Power Pins

- **VIN** (sometimes labeled "9V"): The input voltage to the Arduino board when it's using an external power source (as opposed to 5 volts from the USB connection or other regulated power source). You can supply voltage through this pin, or, if supplying voltage via the power jack, access it through this pin. Also note that the Lily Pad has no VIN pin and accepts only a regulated input.
- **5V:** The regulated power supply used to power the microcontroller and other components on the board. This can come either from VIN via an on-board regulator, or be supplied by USB or another regulated 5V supply.
- **3V3** (Decimal-only) : A 3.3 volt supply generated by the on-board FTDI chip.
- **GND:** Ground pins.

Other Pins

- **AREF:** Reference voltage for the analog inputs. Used with analog Reference().
- **Reset:** (Decimal-only) Bring this line LOW to reset the microcontroller. Typically used to add a reset button to shields which block the one on the board.

ARDUINO CHARACTERISTICS

The Arduino Mega can be powered via the USB connection or with an external power supply. The power source is selected automatically. External (non-USB) power can come either from an AC-to-DC adapter (wall-wart) or battery. The adapter can be connected by plugging a 2.1mm center-positive plug into the board's power jack. Leads from a battery can be inserted in the Gnd and Vin pin headers of the POWER connector. The board can operate on an external supply of 6 to 20 volts. If supplied with less than 7V, however, the 5V pin may supply less than five volts and the board may be unstable. If using more than 12V, the voltage regulator may overheat and damage the board. The recommended range is 7 to 12 volts. The power pins are as follows:

- **IOREF.** This pin on the Arduino board provides the voltage reference with which the microcontroller operates. A properly configured shield can read the IOREF pin voltage
  and select the appropriate power source or enable voltage translators on the outputs for working with the 5V or 3.3V.

4.3 REQUIREMENT SPECIFICATION

Functional Requirements

- Level detection: The level of alcohol.

SYSTEM REQUIREMENTS

- Arduino Compiler.
- Operating System: Windows10.

HARDWARE REQUIREMENTS

- Arduino Mega
- Connecting Wires
- Alcohol sensor (MQ-3)
- AC Motor
- Chassis
- System: Intel i3,2.4GHz
- Hard disk: 40GB
- Ram: 4GB

4.4 UML DIAGRAMS

Introduction

The Unified Modelling Language is a rich visual modelling language created for architecture, design and implementation of complex software systems both structurally and behaviorally. UML consists of different types of diagrams. They describe the boundary, structure and the behavior of the system and the objects with in it.

The vocabulary of UML consists of the following:

1. Things
2. Relationships
3. Diagrams

THINGS OF UML

Things are the abstractions that are first-class citizens in a model. There are four kinds of things

- Structural Things
- Behavioral Things
- Grouping Things
- Annotational Things

Structural Things

Structural things are the nouns of the UML models. They represent elements that are physical

Behavioral Things

They are dynamic parts of UML model. Behavioral things are the verbs of a model, representing behavior over time and spaces. The different types of behavioral things are

- Interactions
- State Machine

Grouping Things

Grouping things are the organizational parts of the UML model. This includes different packages for example: a package can have name, business rules, date, etc.

An notational Things

Explanatory parts of the UML models. This includes nodes these nodes can briefly explain what it is, when it happened, and who is responsible for it etc.

RELATIONSHIPS

Different types of relationships in UML they are as follows:

Association:

It is a relationship between classifiers which is used to show that instances of a classifiers could be either linked to each other or combined logically or physically.

Aggregation:

A special form of association that specifies a whole – part relationship between the aggregate and a component part.

Composition:

It is a type of aggregation that is strong

Generalization:

Generalization is a process of extracting shared characteristics from two or more classes and combining them into generalized super class.

4.4.1 CLASS DIAGRAM FOR SAFE DRIVING USING ALCOHOL DETECTION

It is a static diagram that gives static view of an application. Class Diagram is a collection of classes, interfaces, association, collaboration, and constraints. It is also known as a structural Diagram. These are the only UML diagrams that can be mapped directly with object-oriented languages. It describes the attributes and operations of a class and also the constraints imposed on the system.

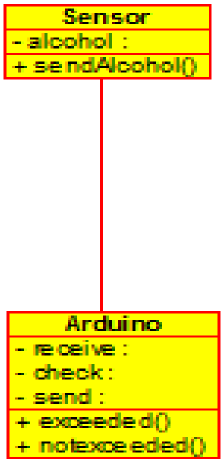


Fig: 4.4.1 Class diagram for SAFE DRIVING USING ALCOHOL DETECTION

4.4.2 SEQUENCE DIAGRAM

Sequence diagram is a type of interaction diagram that is mostly used. It mainly focuses on the message interaction between a number of lifelines.

It is a popular dynamic modelling focuses on the interaction occurring within the system. This diagram mainly focuses on the lifeline of an object and the communication between these lifelines.

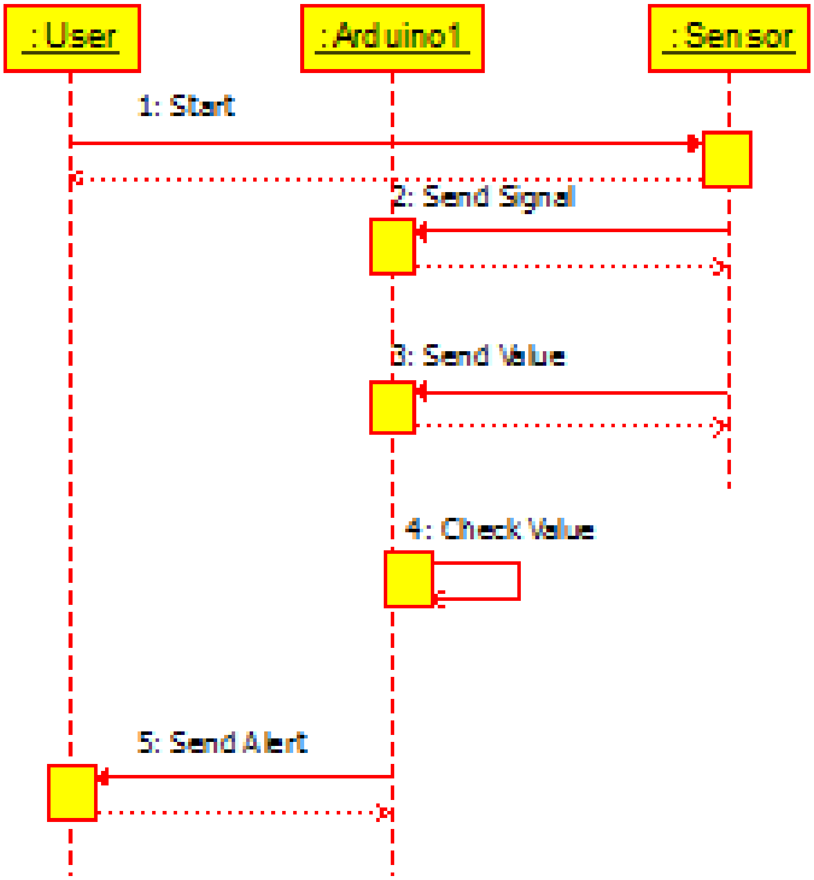


Fig: 4.4.2 Sequence diagram for SAFE DRIVING USING ALCOHOL DETECTION

5. RESULTS AND ANALYSIS

TESTING PROCESS

5.1 OVERVIEW OF TESTING

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub-assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

We can verify through testing, the various interactions, integration of components and the requirements which were implemented. It provides timely feedback to resolve the quality issues, in a timely and cost-effective manner. The ultimate goal of testing is to assess the quality of the end product. Quality assessments often consider process quality and organizational factors as well as direct product quality.

Testing is not a single activity, nor is it a phase in the project during which we assess quality. If developers are to obtain timely feedback on evolving product quality, testing must occur throughout the lifecycle: we can test the broad functionality of early prototypes: we can test the stability, coverage and performance of the architecture while there is still an opportunity to fix it; and we can test the final product to assess its readiness for delivery to customers.

Dimensions of Testing

To assess product quality, different kinds of tests, each one with a different focus, are needed. These tests can be categorized by several dimensions

Quality dimension

The major quality characteristic or attribute that is the focus of test.

Stage of testing

The point in the lifecycle at which the test, usually limited to a single quality

Type of testing

The specific test objective for an individual test, usually limited to a single quality dimension.

Stages of Testing

Testing is not a single activity, executed all at once. Testing is executed against different types of targets in different stages of the software development. Test stages progress from testing small elements of the system, such as components (unit testing), to testing completed systems (system testing). The 3 stages have the following purposes:

Unit Testing

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases. Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

Integrated Testing

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects. The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Acceptance Testing

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Types of Testing

After a test plan has been developed, system testing begins by testing program modules separately, followed by testing "bundled" modules as a unit. A program module may function perfectly in isolation but fail when interfaced with other modules. The approach is to test each entity with successively larger ones, up to the system test level.

System Testing

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration-oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

Program testing

A program represents the logical elements of system. For a program to run satisfactorily, it must compile and test data correctly and tie in properly with other programs. Achieving an error-free program is the responsibility of the programmer.

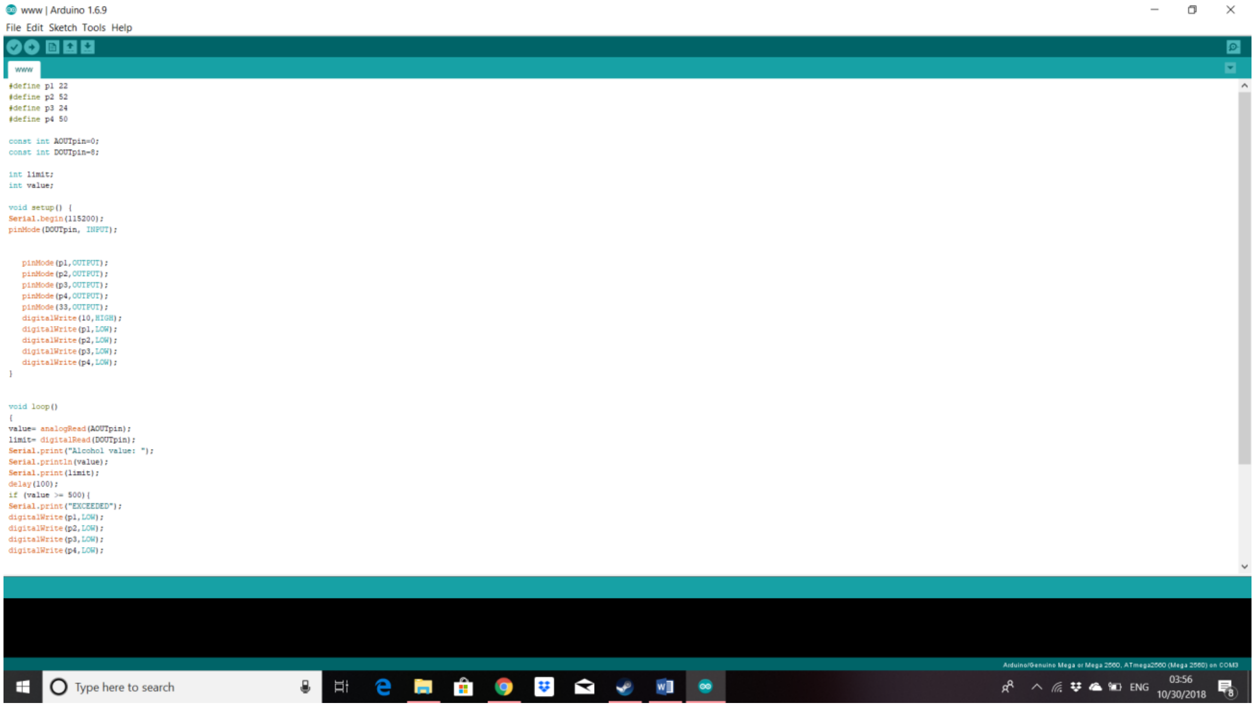
Program testing checks for two types of errors: syntax and logic. A syntax error is a program statement that violates one or more rules of the language in which it is written. A logic error, on the other hand, deals with incorrect data fields, out of range items, and invalid combinations. When a program is tested, the actual output is compared with the expected output. When there is a discrepancy, the sequence of instructions must be traced to determine the problem. The process is facilitated by breaking the program down into self-contained portions, each of which can be checked at certain key points. The idea is to compare program values against desk calculated values to isolate the problem.

String Testing:

Programs are invariably related to one another and interact in a total system. Each program is tested to see whether it conforms to related programs in the system. Each portion of the system is tested against the entire module with both test and live data before the entire system is ready to be tested.

5.2 User Acceptance Testing:

An acceptance test has the objective of selling the user on the validity and reliability of the system. It verifies that the system's procedures operate to system specifications and that the integrity of vital data is maintained. Performance of an acceptance test is actually the user's show. User motivation and knowledge are critical for the successful performance of the system. Then a comprehensive test report is prepared. The report indicates the system's tolerance, performance range, error rate, and accuracy.



```
#define p1 22
#define p2 52
#define p3 24
#define p4 50

const int AOUTpin=0;
const int DOUTpin=8;

int limit;
int value;

void setup() {
Serial.begin(115200);
pinMode(DOUTpin, INPUT);


   pinMode(p1,OUTPUT);
   pinMode(p2,OUTPUT);
   pinMode(p3,OUTPUT);
   pinMode(p4,OUTPUT);
   pinMode(33,OUTPUT);
   digitalWrite(10,HIGH);
   digitalWrite(p1,LOW);
   digitalWrite(p2,LOW);
   digitalWrite(p3,LOW);
   digitalWrite(p4,LOW);
}


void loop()
{
value= analogRead(AOUTpin);
limit= digitalRead(DOUTpin);
Serial.print("Alcohol value: ");
Serial.println(value);
Serial.print(limit);
delay(100);
if (value >= 500){
Serial.print("EXCEEDED");
digitalWrite(p1,LOW);
digitalWrite(p2,LOW);
digitalWrite(p3,LOW);
digitalWrite(p4,LOW);
```

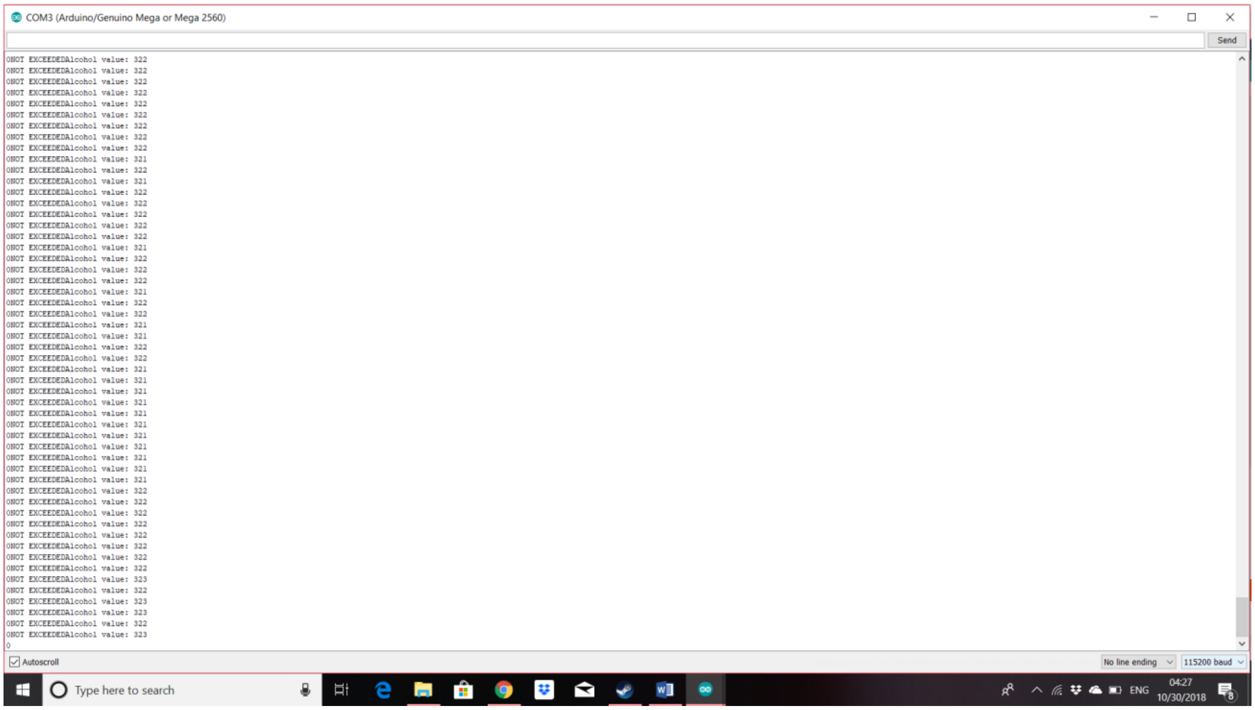
Arduino/Genuino Mega or Mega 2560, ATmega2560 (Mega 2560) on COM3
Type here to search
ENG
03:56
10/30/2018

Fig: 5.2 User Acceptance Testing

5.3 Negative Test Case Results:

S. No	Input	Expected Output	Actual Output	Results
1.	No Alcohol	Not Excided	Not Excided	Pass
2.	No Alcohol	Not Excided	Not Excided	Pass

In the below figure we can see that at the time of testing there was no alcohol so message which is displayed is “NOT EXCIEDED”. Also, displayed values are the calibration values.



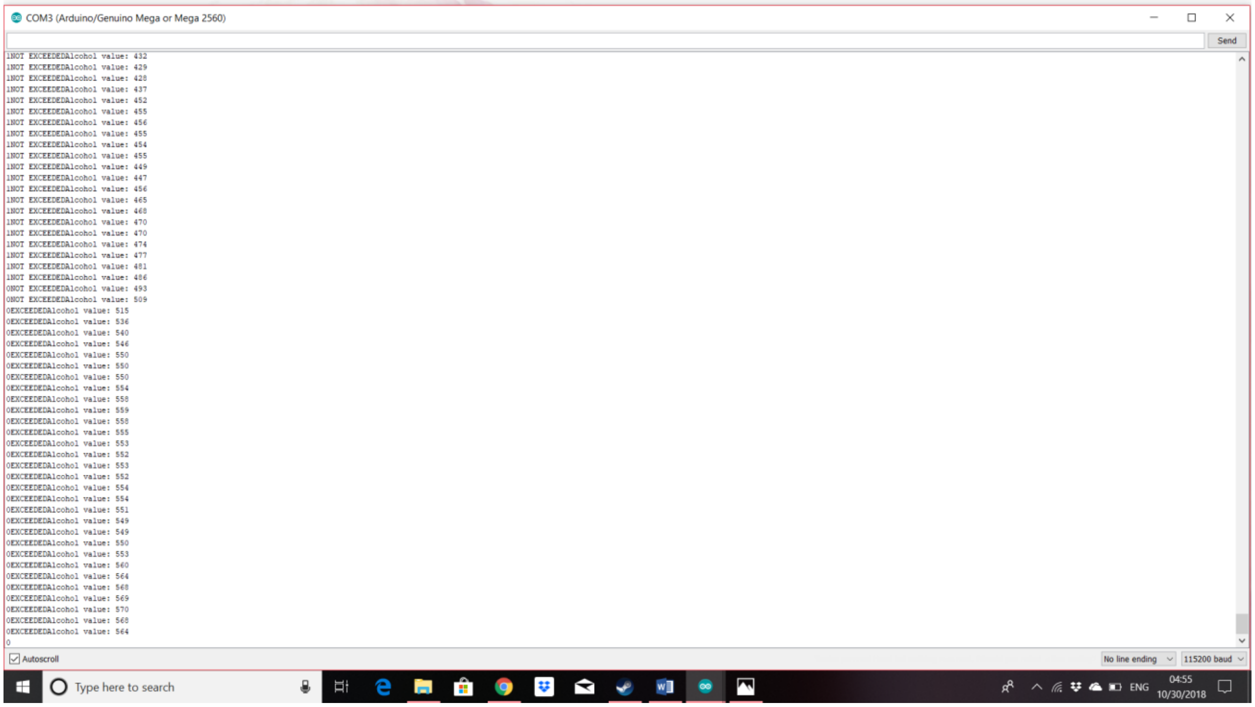
```
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 321
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 323
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 323
0NOT EXCEEDEDAlcohol value: 323
0NOT EXCEEDEDAlcohol value: 322
0NOT EXCEEDEDAlcohol value: 323
0
```

Fig: 5.3 Negative Test Case Results

5.4 Positive Test Case Results:

S. No	Input	Expected Output	Actual Output	Results
1.	Alcohol	Excided	Excided	Fail
2.	Alcohol	Excided	Excided	Fail

Now in this below figure we can see that due to the presents of alcohol the message which is displayed over is “EXCIDED” and those are the actual value od the alcohol present or consumed by driver



```
1NOT EXCEEDEDAlcohol value: 432
1NOT EXCEEDEDAlcohol value: 429
1NOT EXCEEDEDAlcohol value: 428
1NOT EXCEEDEDAlcohol value: 437
1NOT EXCEEDEDAlcohol value: 452
1NOT EXCEEDEDAlcohol value: 455
1NOT EXCEEDEDAlcohol value: 456
1NOT EXCEEDEDAlcohol value: 455
1NOT EXCEEDEDAlcohol value: 454
1NOT EXCEEDEDAlcohol value: 455
1NOT EXCEEDEDAlcohol value: 449
1NOT EXCEEDEDAlcohol value: 447
1NOT EXCEEDEDAlcohol value: 456
1NOT EXCEEDEDAlcohol value: 465
1NOT EXCEEDEDAlcohol value: 468
1NOT EXCEEDEDAlcohol value: 470
1NOT EXCEEDEDAlcohol value: 470
1NOT EXCEEDEDAlcohol value: 474
1NOT EXCEEDEDAlcohol value: 477
1NOT EXCEEDEDAlcohol value: 481
1NOT EXCEEDEDAlcohol value: 486
0NOT EXCEEDEDAlcohol value: 493
0NOT EXCEEDEDAlcohol value: 509
0EXCEEDEDAlcohol value: 515
0EXCEEDEDAlcohol value: 536
0EXCEEDEDAlcohol value: 540
0EXCEEDEDAlcohol value: 546
0EXCEEDEDAlcohol value: 550
0EXCEEDEDAlcohol value: 550
0EXCEEDEDAlcohol value: 550
0EXCEEDEDAlcohol value: 554
0EXCEEDEDAlcohol value: 558
0EXCEEDEDAlcohol value: 559
0EXCEEDEDAlcohol value: 558
0EXCEEDEDAlcohol value: 555
0EXCEEDEDAlcohol value: 553
0EXCEEDEDAlcohol value: 552
0EXCEEDEDAlcohol value: 553
0EXCEEDEDAlcohol value: 552
0EXCEEDEDAlcohol value: 554
0EXCEEDEDAlcohol value: 554
0EXCEEDEDAlcohol value: 551
0EXCEEDEDAlcohol value: 549
0EXCEEDEDAlcohol value: 549
0EXCEEDEDAlcohol value: 550
0EXCEEDEDAlcohol value: 553
0EXCEEDEDAlcohol value: 560
0EXCEEDEDAlcohol value: 564
0EXCEEDEDAlcohol value: 568
0EXCEEDEDAlcohol value: 569
0EXCEEDEDAlcohol value: 570
0EXCEEDEDAlcohol value: 568
0EXCEEDEDAlcohol value: 564
0
```

Fig: 5.4 Positive Test Case Results

6. CONCLUSION AND FUTURE SCOPE

By the help of the embedded system it is much easier to operate and the cost for making is very low. As this product are of low cost it could be easily replaced and could be experimented a lot, and this embedded system could used for the IOT (Internet Of Things). So, it could be used in more efficient and advance manner as the size of this product is very small and get fitted any place easily.

For the further scope I can be able to add the GPS and GSM modules such that when it is developed we could get the geolocation of the person and that location is send to the emergency contact via message through GPS module. Which will be helpful to the people to go and pick the drunk person.

References

[1] https://www.arduino.cc/

[2] https://www.tutorialspoint.com/

[3] https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/