

ACCIDENT ALERT SYSTEM USING ARDUINO UNO

PROJECT REPORT

Submitted by

P.S.SHILPA

Reg No: AB20PHY022

Under the guidance of

Dr. SUNSU KURIAN, ASSISTANT PROFESSOR

Department of Physics, St. Teresa's College (Autonomous), Ernakulam

Kochi, 682011

Submitted to,

Mahatma Gandhi University, Kottayam

In partial fulfillment of the requirement for the award of

BACHELOR'S DEGREE OF SCIENCE IN PHYSICS



St. TERESA'S COLLEGE (AUTONOMOUS)

ERNAKULAM

ST.TERESA'S COLLEGE (AUTONOMOUS)

ERNAKULAM



CERTIFICATE

This is to certify that the project entitled “**ACCIDENT ALERT SYSTEM USING ARDUINO UNO**” is an bonafide work done by P.S.Shilpa of St. Teresa’s college (Autonomous),Ernakulam, under my supervision at the Department of Physics, St. Teresa’s college Ernakulam for the partial fulfillment of the award of degree of bachelor of science in physics during the academic year 2022-23.The work presented in this dissertation has not been submitted for any other degree in this or any other university.

Supervising guide

Dr. Sunsu Kurian

Assistant Professor

Head of the department

Dr. Priya Parvathy Ameena Jose

Assistant professor

Place:Ernakulam

Date : 25/04/2023

ST.TERESA'S COLLEGE (AUTONOMOUS), ERNAKULAM



B.Sc. Physics


PROJECT REPORT

Name: P.S.SHILPA

Register Number: AB20PHY022

Year of work: 2022-'23

This is to certify that this project "ACCIDENT ALERT SYSTEM USING ARDUINO UNO " is the work done by P.S.SHILPA.

Staff member in charge 
Dr. SUNSU KURIAN

Head of the department 
Dr .PRIYA PARVATHI AMEENA JOSE

Submitted for the university examination held in St. Teresa's College (Autonomous), Ernakulam.

Date 25/04/2023

Examiners: 






DECLARATION

I, **P.S.SHILPA**(REG NO:AB20PHY022),final year B.Sc Physics student ,Department of Physics ,St. Teresa's College (Autonomous)Ernakulam do hereby declare that the project work entitled "**ACCIDENT ALERT SYSTEM USING ARDUINO UNO**" has been originally carried out under the guidance of Dr.Sunsu Kurian ,Assistant Professor, Department of Physics, St. Teresa's College (Autonomous) Ernakulam in partial fulfilment for the award of the degree of bachelor's of Physics. We further declare that this project is not partially or wholly submitted for any other purpose and the data included in the project is collected from various sources and are true to the best of my knowledge.

Place:Ernakulam

Date: 19/4/23

P.S SHILPA


ACKNOWLEDGEMENT

We express our sincere gratitude to everyone who has helped us in this project .First of all, we express our gratitude to god almighty, for having strengthened us and for having made me capable of doing. We wish to convey our deep gratitude to our beloved and respected Dr.Sunsu Kurian ,the project in charge ,for guiding us throughout the journey of project work.

We are indebted to all the faculty members of the physics department for their valuable suggestions and corrections .And also, we owe our heartfelt gratitude to our parents, friends for their deep support and help which made the completion of this project possible.

CONTENTS

| Chapter | Title | Page no |
|---------|------------------------------|---------|
| | Abstract | |
| 1 | Introduction | 7-8 |
| | 1.1 Relevance of project | 9 |
| 2 | Over view of Components | 10-18 |
| 3 | Methodology | 19-22 |
| 4 | Working | 23-25 |
| | 4.1 Observations and Results | 26-28 |
| 5 | Advantages and Disadvantages | 29 |
| 6 | Conclusion and Future Scope | |
| | 6.1 Conclusion | 30 |
| | 6.2 Future scope | 31 |
| 7 | Appendix | 32-47 |
| 8 | References | |

ABSTRACT

In day to day life a large number of people die from traffic accidents. Road accidents is the most unwanted thing happen to a road user ,though they happen quite often .Many accidents result from more than one cause, so several people and entities may be involved in the case.Speed is one of the reasons for vehicle accidents. Many lives could have been saved if emergency service could get accident information and reach in time .Accident alert systems help to reduce fatalities stemming from accidents by decreasing the response time of emergency responders .Smartphones and their onboard sensors(such as GPS receivers and accelerometers)are promising platforms for constructing such systems. Nowadays ,GPS has became an integral part of a vehicle system .This project deals with an Arduino based accident alert system .Here an inexpensive ,small sized ,minimum cost sensor(Accelerometer) is incorporated to an Arduino microcontroller unit. With signals from an accelerometer, a severe collision is recognized and the vibration sensor will send a signal to microcontroller. This system aims to alert the nearby medical center about the accident to provide immediate medical aid . The attached accelerometer in the vehicle senses the accident and thus system will make the decision and sends the information to the smartphone ,connected to the accelerometer through GSM and GPS modules. In addition to this ,system shares the exact location of the accident and it can save time. This will help to reach the rescue service in time and save the valuable human life .The advantages of the detector are rapid response ,recovery, reliability, stability ,long life requirement.

CHAPTER 1

INTRODUCTION

Nowadays ,population of the world is increasing considerably. The development of transportation system has been a vital part for the human existence .We utilize it to go to our work place, keep in touch with our friends and family, and deliver our goods .But it can also bring disaster to us and even can kill us through accidents .Life of the people is under high risk. Road accidents constitute the major part of the accident deaths all over the world. It not only affects the severity of a crash but also increases risk of being involved in a crash. The main cause of the accident death is “delay of rescue ”.Despite many efforts taken by different governmental and non-governmental organizations all over the world by various programs to aware against careless driving, yet accidents are taking place every now and then.

Most unfortunate thing is that we don't learn from our mistakes on the road. Most of the road users are quite well aware of the general rules and safety measures while using the roads .But it is only the laxity on the part of road users, which causes accidents and crashes .The main cause of accidents and crashes are due to human errors .We are elaborating some of the common behaviours of humans, which results in accidents.

- 1.Over speeding
- 2.Drunken driving
- 3.Distractions to driver
- 4.Red light jumping
- 5.Avoiding safety gears like seat belts and helmets.
- 6.Non-adherence to lane driving and overtaking in a wrong manner.

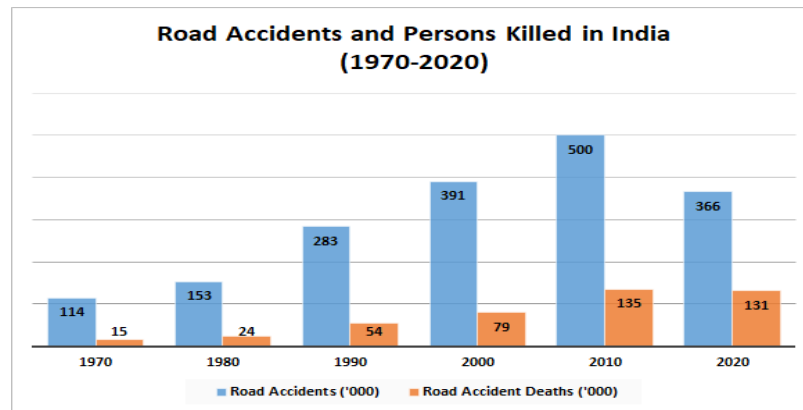


Fig 1: Histogram of accidents and death rate.

Whenever a road accident happens, the people nearby have to manually call the ambulance which causes a waste of time. Hence there is a delay for emergency services to arrive at the location of the accident.

The problem is difficult to know that an accident has occurred and to locate the position where it has happened. However, many lives could have been saved if the emergency service could get the crash information in time. As such, efficient automatic accident detection with an automatic notification to the emergency service with the accident location is a prime need to save the precious human life.

In order to fix this problem, we are going to build a system that will provide the emergency facilities for the victims in the shortest time possible. It incorporates an embedded system that contains GPS and GSM modules connected with an Arduino UNO. The entire set-up is installed at the front end of the vehicle. This application provides the optimum solution to poor emergency facilities provided for road accidents in the most feasible way.

Global Positioning System (GPS) is used to identify the location of the vehicle. GSM is used to inform the exact vehicular location in the form of the longitude and latitude coordinates to the precoded numbers using an SMS. GSM module provides a two-way communication by using a SIM card. The accelerometer detects the sudden change in the axes of the vehicle and the GSM module sends the alert message on your Mobile Phone with the location of the accident. The location of the accident is sent in the form of the Google Map link, derived from the latitude and longitude from the GPS module.

1.1 Relevance of ARDUINO based projects

Science and technology have always played a significant role in expanding human knowledge, enhancing education, and making life more useful. In other words, science is one of the most important channels of knowledge. It has specific role as well as a variety of functions for the benefit of our society.

In order to save human lives and make this a better and safer place to live, we have devised this project. Additionally, we firmly think that the advantages of science and technology must be applied to enhance human life. As a result, we are presenting a method for detecting accidents using an Arduino board, which is crucial to robotics because it fosters creativity and problem-solving.

The most popular and often used development board is the Arduino UNO. Since the hardware used by Arduino is open source, anyone can modify the design to make a new item.

There are many other microcontrollers and development boards, but arduinos accessibility, easy to understand hardware design and simple software make it appealing to different types of users and it is affordable ,it can be programmed using a flexible platform, the arduino IDE which can run on 3 operating system-LINUX ,windows and MAC. Thus makes it friendly to students.

LITERATURE SURVEY

Arduino based other projects:

1. Accident detection system

In this project, accelerometer detects the inclination of the vehicle. It warns the driver by providing a buzzer sound with blinking of LED when the vehicle exceeds to a particular angle.

2. . Smart GPS Tracker Using Arduino

In this project, a smart tracker that can keep track of a child. Besides this, the device can also be used to track your vehicle location and other objects .After coding,, create the setup function where the baud rate of SIM800L and GPS modules for serial communication are set to 4800 and 9600 respectively. create a loop function to check whether the incoming message has been received by SIM800L or not. If the message has been received, then convert it into a readable

string. It will check the message for a 'get location' command, after which the collected data (with respect to location) will be sent back to the original device..Within the same loop function, a function named get location() will be called .This function will keep updating the GPS data and return the results to the loop function

3.Arduno GPS clock

There are many GPS satellites around the Earth which are used to provide the exact location of any place. Along with the location coordinates (Latitude and Longitude), it also provide other data like time, date, altitude, direction tracking angle etc. So here made a GPS clock using the 'Time and Date' data from the GPS satellite. GPS Updated Clock is very accurate and provides the real time data with precision of milliseconds.

CHAPTER -2

OVERVIEW OF THE COMPONENTS

1. ARDUINO UNO

2. JUMPER WIRES

3. ACCELEROMETER

4.GSM SIM800L

5.GPS NEO-6M

6.LM2596 STEP CONVERTOR

7.ZERO PCB

1. ARDUINO UNO

It is a microcontroller board based on the ATmega328p. It has 14 digital input/ output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack, an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with ac to dc adapter or battery to get started. “UNO” means one in Italian and was chosen to mark the release of Arduino Software 1.0. The Uno board and version 1.0 of Arduino Software were the reference versions of Arduino, now evolved to newer releases. The Uno board is the first in a series of USB Arduino boards, and the reference model for the Arduino platform.



Fig 2: Arduino-UNO

2.JUMPER WIRES

A jumper wire or jump wire is an electrical wire, or a group of them in a cable, with a connector or pin at each end, which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering. There are three types of jumper wires: female to female, male to male and male to female.



Fig 3:Jumper Wires

3. ACCELEROMETER

Nowadays different types of accelerometers are available in market which are used for various purposes. Some of them works on the principle of MEMS(micro electro mechanical sensor) working. Which consists of a small mass which is engraved into silicon surface and then integrated into a small circuit. When force is applied on this mass then it covers some displacement, so acceleration is produced in this mass according to newton second law of motion $F= ma$ which is sensed by sensor. Similarly, if we consider analog accelerometers then they work on two principles like capacitive sensing and piezo electric sensing. Apparently, ADXL335 accelerometer is an analog accelerometer therefore it works on the principle of capacitive sensing. In capacitive sensing accelerometer, when it is moved in any direction then its capacitance is varied. When this capacitance is changed then its analog voltages are changed too which is sensed by its interfacing controller.

GY-61 DXL335 3-Axis Accelerometer Module is a three axis accelerometer sensor module based on ADXL335 integrated circuit. The ADXL335 is a triple axis accelerometer with extremely low noise and power consumption. The sensor has a full sensing range of +/-3g. It can measure the static acceleration of gravity in tilt-sensing applications, as well as dynamic acceleration resulting from motion, shock, or vibration. There is an on-board 3.3V voltage regulator to power the ADXL335 so power provided should be between 3.3V and 6V DC.

Model : GY-61

Three-axis magnetic field accelerometer module

Compact size, low power supply used for game systems, mobile devices, etc .

General Specifications

- ADXL335 3-axis Accelerometer
- On-board 3.3V Voltage Regulator
- Analog voltage output centered at 1.65V
- Suitable for connection to 5V and 3.3V systems.



Fig 4: Accelerometer adxl335

Technical Specifications

- Sensor Chip : ADXL335

- Operating Voltage Range : 3V ~ 5V
- Supply Current : 400uA
- Interface : Analog quantity output
- Full scale range : +/-3g
- Operating Temperature : -40'C~ +85'C
- Sensitivity : 300mv /g;
- Sensitivity of accuracy (%) : +/- 10
- Application : Various electronic products or DIY project
- Material : PCB + Brass
- Dimensions : 21 x 16 x 10 mm / 0.83 x 0.63 x 0.39 inch
- Weight : 2 g / 0.07 oz
- Color : Blue

DIMENSIONS

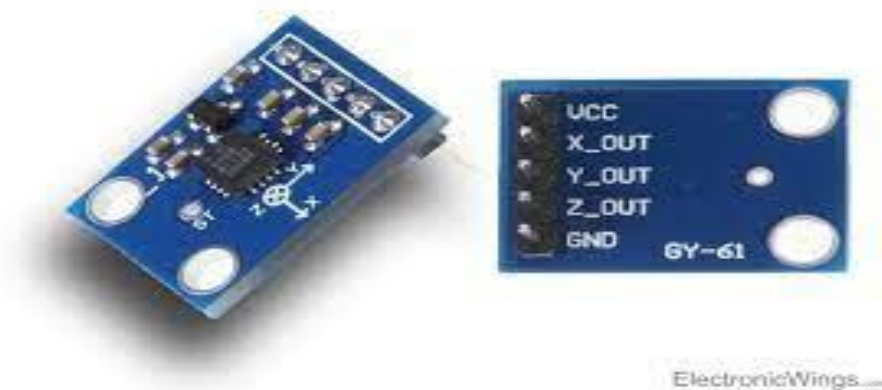


Fig5: Accelerometer

PIN DEFINITIONS:

1. VCC: 3.3V or 5V
2. X_OUT: Analog Output
3. Y_OUT: Analog Output
4. Z_OUT: Analog Output
5. GND: Ground

4.GSM SIM800I

SIM800L GSM/GPRS module is a miniature cellular GSM modem from simcom ,which can easily interface with any microcontroller to give the microcontroller GSM functionality ,and allows for GPRS transmission. This module connects the microcontroller to the mobile network to make or receive phone calls send or receive sms (text messages),and connect to the internet using GPRS,TCP,IP . Another advantage is ,it supports quadband GSM/GPRS network ,which means it can work any where in the world.

The SIM800L GSM/GPRS module consists of four key components, which take important roles in the work of the module. These key components are SIM800L GSM cellular chip, LED Status Indicators, Antennas, and Micro-SIM socket. On the top surface of the GSM module, we can see a chip is mounted on the module board. This is a Quad-band SIM800L GSM/GPRS cellular chip from SimCom in SMT type. SIM800L supports Quad-band frequency its works on frequencies 850MHz, 900MHz, 1800MHz, and 1900MHz, it can transmit and receive voice, SMS, and data information with low power consumption. The operating voltage of this chip is from 3.4V to 4.4V which makes it ideal to operate by a LiPo battery supply.

LED Status Indicators

On the topmost right corner side of the SIM800L Module, we can see an LED that indicates the status of your cellular network. After applying the power supply to the module the LED will blink at three different ratios, which shows three different statuses of your cellular network.

Blink every 1s:

When the LED Blinking with a delay of 1s, then it indicates that the GSM module is running but it hasn't made the connection to the cellular network yet.

Blink every 2s:

When the LED Blinking with a delay of 2s, then it indicates that The GPRS data connection you requested is active.

Blink every 3s:

When the LED Blinking with a delay of 2s, then it indicates that the module has made contact with the cellular network and it is ready to transmit/receive voice and SMS.



Fig 6: GSM SIM8001

5.GPS NEO -6m

The NEO-6m GPS module is a well performing complete GPS receiver with a built in 25x25x4mm ceramic antenna, which provides a strong satellite search capability .With the power and signal indicators, you can monitor the status of the module .Module can save the data when the main power is shut down accidentally. It can track up to 22 satellites over 50 channels while consuming only 45 mA of current.

- High power saving mode
- Power supply 3-5v.Default baud rate 9600 bps

TEST

In this test,we will send the positioning data collected by the NEO-6M GPS module to the software on the pc and compare the data with that of a GPS device thus we can know this module works or not.



Fig 7: GPS NEO6m

6. LM2596 STEP CONVERTOR

The LM2596 regulator is monolithic integrated circuit ideally suited for easy and convenient design of a step-down switching regulator (buck converter). It is capable of driving a 3.0 A load with excellent line and load regulation. This device is available in adjustable output version and it is internally compensated to minimize the number of external components to simplify the power supply design. Since LM2596 converter is a switch-mode power supply, its efficiency is significantly higher in comparison with popular three-terminal linear regulators, especially with higher input voltages. The LM2596 operates at a switching frequency of 150 kHz thus allowing smaller sized filter components than what would be needed with lower frequency switching regulators.

APPLICATIONS

- Simple High-Efficiency Step-Down (Buck) Regulator
- Efficient Pre-Regulator for Linear Regulators
- On-Card Switching Regulators
- Positive to Negative Converter (Buck-Boost)
- Negative Step-Up Converters
- Power Supply for Battery Chargers



Fig 8:LM2596 step convertor

7.ZERO PCB

Zero PCB is basically a general purpose printed circuit board(PCB) also known as perfboard or DOT PCB.It is a thin rigid copper sheet with holes predrilled at standard intervals across a grid with 2.54mm spacing between holes .The PCB's are generally used to provide electrical connection and mechanical support to the electrical component of a circuit .They are prevalent in electronic devices and can be easily identified as the green coloured board in most cases.



Fig 9:zero PCB

CHAPTER -3

METHODOLOGY

The project is based on accident alert system using Arduino,GPS,and GSM.Now I will explain the pins of Arduino where I connected all the components.ADXL 335 sensor is connected to the pin number :-

- Vcc - 5v
- GND -GND
- X-axis- A0
- Y-axis- A1
- Z-axis -A2

GSM SIM8001 module will be connected to the pin number:-

- Vcc -3.3v
- GND-GND
- Tx - D2
- Rx - D3

GPS Neo-6m module will be connected to the pin number:-

- Vcc-5v
- GND-GND
- Tx-D10
- Rx-D11

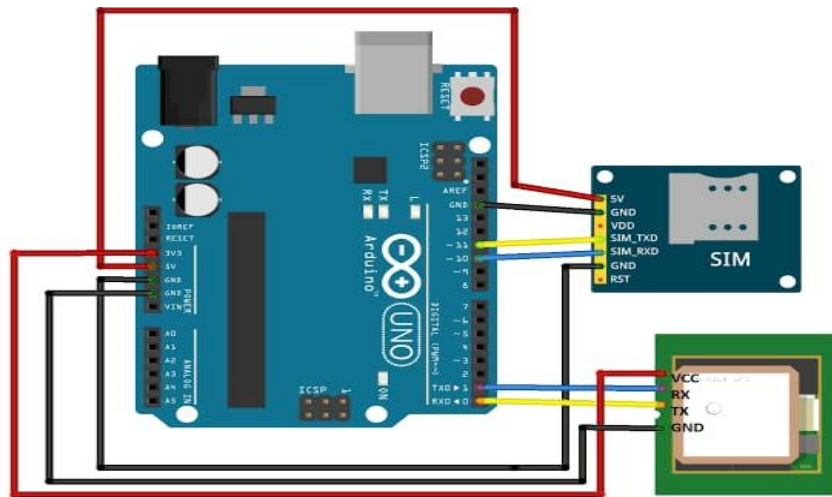
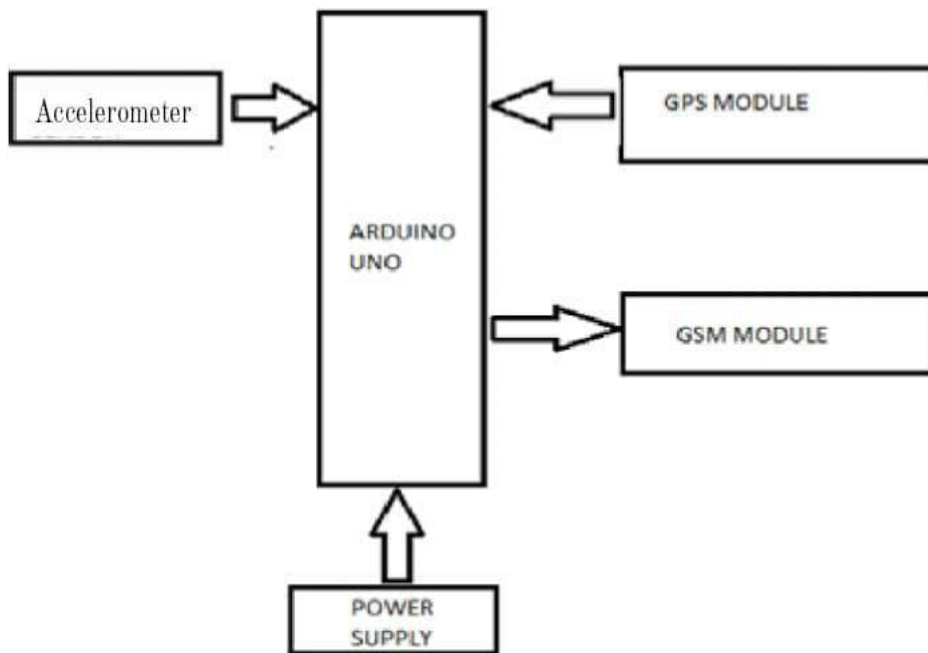


Fig10: circuit diagram of accident alert system.

If any movement is detected by the ADXL 335 sensor ,GSM send the message.

BLOCK DIAGRAM



Step converter is connected to the GND of Arduino board and vcc of GSM module.

3.1 Software Specification

The software used to implement the accident alert system is Arduino IDE (integrated development environment)-win 7 and newer.The software is installed on the laptop.This is an open source software that makes it easy to write code to Arduino uno board .The programs written in the Arduino ide are called as sketches. The software is well-suited with all kinds of operation systems like linux, windows ,and macintosh etc.Some of the basic functions of Arduino technology are,

- Void setup()-it is a function that you create at the top of each program .Inside the curly brackets is the code that you want to run one time as soon as the program starts running.
- Void loop()- This is where the bulk of Arduino sketches are executed.
- digitalRead()-it reads the digital value of the given pin.
- digitalWrite()-it is used to write the digital value to the given pin.
- analogRead()-it reads the analog pin and returns the value.
- analogWrite()-it is used to write the analog value to the given pin.
- pinMode()-it is used to set the pin to I/O mode.
- Serial.begin(9600)-it is used to set the beginning of serial communication by setting the rate of bit.
- Serial.println()-it is used to print any message ,any value of a variable or the values returned by the functions on the serial monitor, which is a screen that displays such information for the user.

How to upload a sketch to the Arduino board:-

Step1- enter the code in Arduino Arduino IDE.

Step2- compile the program and correct the syntax errors.

Step3- connect the Arduino using the USB cable.

Step4- select the port.

Step5- choose upload option to upload the sketch to the Arduino UNO.



Fig11:Arduino IDE icon

CHAPTER -4

Working principle

The working of the project Accident Detection and Alert System using Arduino can be summarized in 3 points below:

1. When an accident has occurred, the location details of the vehicle/object are collected by the GPS module from the satellite, this information is in the form of latitude and longitude scales.
2. Thus, collected information is then fed to Arduino uno. Necessary processing is done and the information is passed to the LCD and GSM modem.
3. The GSM modem collects the information for Arduino uno and then transfers it to the mobile phone through the SMS which is in text format.

Connections are made as mentioned above. Plug in Arduino using a USB cable. Arduino acts as the controller of all modules used here. Then, turn on GSM module by applying a 12V input. GSM will send the received coordinates to the user via SMS. But, the GSM module that used here accepts only voltages in the range of 3.6-4.4V. A step converter to convert voltage from high to low (passes only 4.1V to the GSM module). Voltages across step converter and GSM module can be measured using a digital multimeter. Above all, make sure the GSM catches signal after inserting a SIM. It can be determined by the period of blinking on the GSM module. Module receives signal only if it shows blinking in the interval of 3sec. Moreover, GPS module is used to receive the coordinates of the vehicle. After the connections, check whether the GSM module catches signal properly. Upload the program and observe the serial monitor whether all AT commands give us an ok reply, then the module is ready to catch the signal. Screenshots mentioned below show the results of our project.

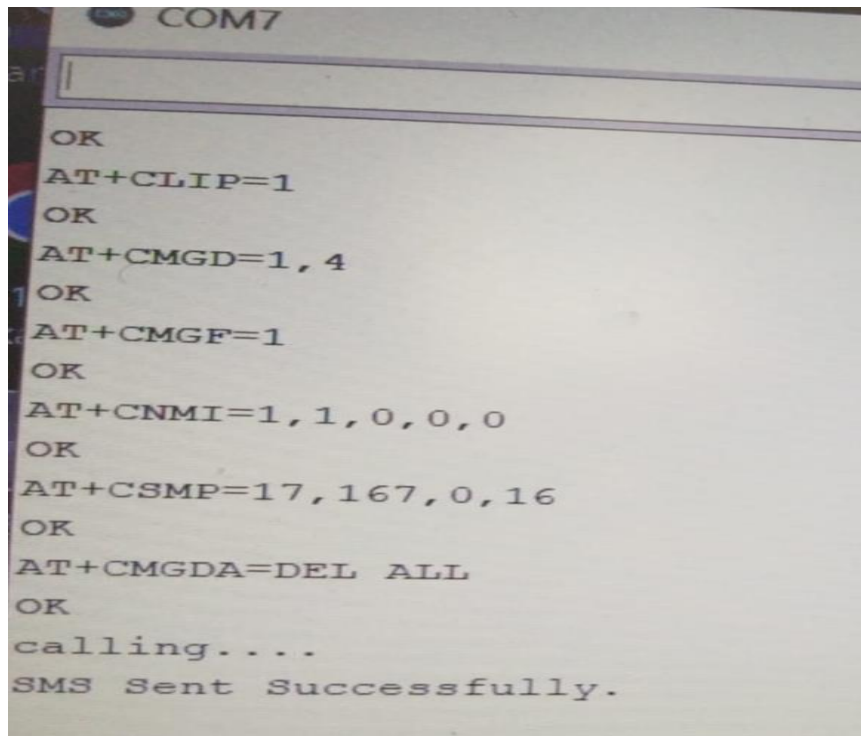


Fig12 :Display on the serial monitor

Here comes the prominent part of the project ,uploading the main software.After uploading the program,you will see ‘collissioncheck!!!’ on the serial monitor. Correspondingly,trigger the accelerometer by tapping on it which demonstrates the accident. while, you would see ‘accident!!!’appearing on the serial monitor. Immediately, serial monitor will show ‘calling’ thereby, call will be send to the phone number which is given as the emergency number in the code. Call will persists for 20s and then cut off.Just after that,serial monitor will show ‘send alert’.Automatically,the exact location of the accident would be send to the specified phone number as a text message.

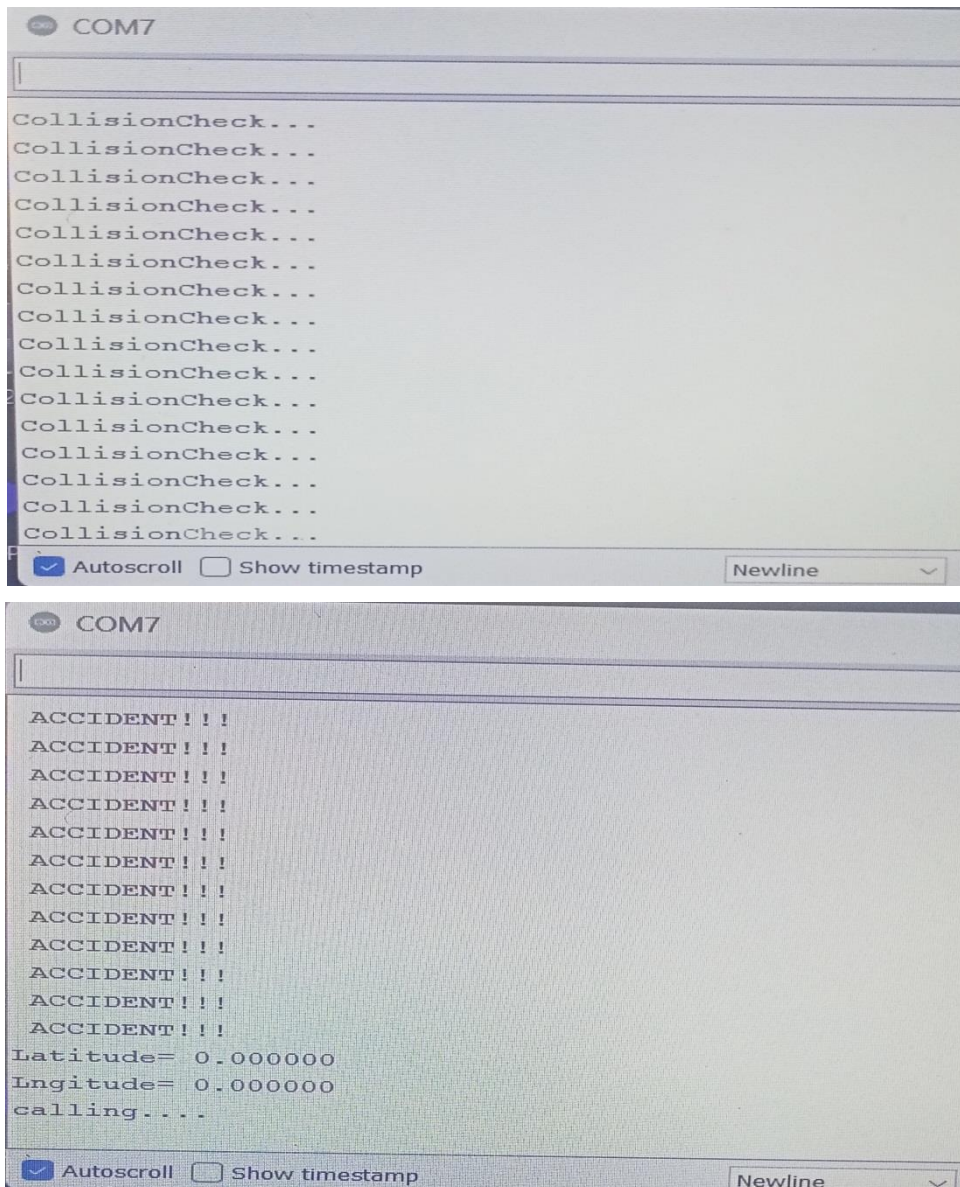


Fig13: display on serial monitor

We can reuse it by restarting the Arduino board only if it is not destroyed by the accident.

Here The accelerometer senses a sudden shift in the vehicle's axles. It will be tested by Arduino. The Arduino sends the warning message via the GSM module to the police control room or a rescue team, including the location. So, after receiving the information, it is possible to automatically track the location via the GPS module. Then, the appropriate action will be taken after verifying the venue.

4.1 OBSERVATIONS

It is also possible to connect the accelerometer sensor to Arduino UNO and reading the corresponding orientations. The accelerometer can measure gravitational acceleration along the 3 axes. So we can measure tilt angle using accelerometer and Arduino.

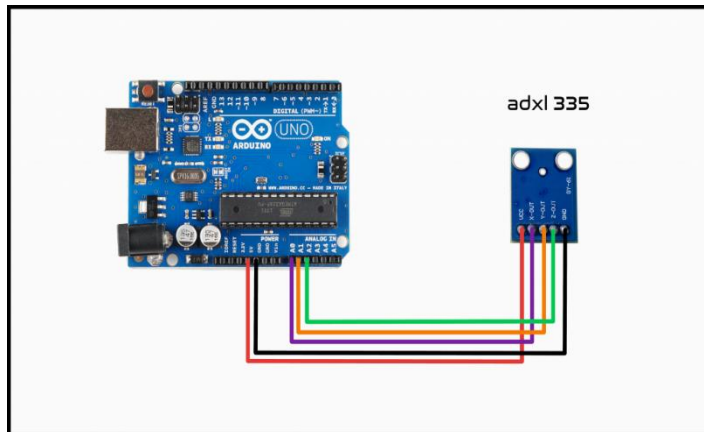


Fig14: Connection Diagram

Running the program will display the default sensor values.

The XYZ coordinates returned by the accelerometer is as shown in table 1-table 3

| Coordinates | ADC values | | | | | Average value |
|-------------|------------|-----|-----|-----|-----|---------------|
| | X | 417 | 416 | 416 | 416 | |
| Y | 348 | 348 | 348 | 348 | 348 | 348 |
| Z | 393 | 392 | 392 | 392 | 392 | 392.2 |

Table 1: XYZ coordinates returned by the accelerometer

| Coordinates | ADC values | | | | | Average value |
|-------------|------------|-----|-----|-----|-----|---------------|
| X | 396 | 396 | 396 | 396 | 396 | 396 |
| Y | 348 | 347 | 348 | 347 | 348 | 347.6 |
| Z | 417 | 417 | 417 | 417 | 417 | 417 |

Table 2. XYZ coordinates returned by the accelerometer

| Coordinates | ADC values | | | | | Average value |
|-------------|------------|-----|-----|-----|-----|---------------|
| X | 299 | 299 | 299 | 299 | 299 | 299 |
| Y | 353 | 353 | 353 | 353 | 353 | 353 |
| Z | 406 | 406 | 405 | 405 | 405 | 405.5 |

Table 3:XYZ coordinates returned by the accelerometer

As a first trial consider a flat desk position(in other words, 0 degrees of tilt). While stable, the X axis shows 357, the Y axis shows 356, and the Z axis returns a value of 431.

To describe the orientation of the sensor , by determining the amount of gravity acting on each part of the sensor, it becomes possible to determine the orientation. If you want to determine

the orientation, we can take the value of X from the XYZ axis display and determine the orientation.

We can see values from 299 (-90 degrees) to 417 (90 degrees). The normal position is around 390, so we can see that the sensor values are 299(-90 degrees), 390(0 degrees), and 417 (90 degrees). We will use a proportional expression to determine the angle.

CALCULATIONS

Determining the correlation coefficient between sensor values and angle

1 degree of sensor value = (largest value – smallest value)/180

$(417 - 299)/180 = 0.65$ ← **approximate value of 1 degree**

Angle = (current sensor value – smallest value)/0.65 – 90

The preceding formula is used to convert the values that we retrieve from Arduino into an angle.

| CURRENT SENSOR VALUE | SMALLEST VALUE | ANGLE |
|----------------------|----------------|-------|
| 403 | 299 | 70 |
| 399 | 299 | 63.8 |
| 396 | 299 | 59.2 |

Table 5: Angle from the accelerometer

CHAPTER -5

ADVANTAGES OF THE SYSTEM

- 1) They are cost effective in implementation.
- 2) They have high tendency to assure safety.
- 3) The victim's life can be saved quickly involved in the accident.
- 4) It has low power consumption.
- 5) It has better accuracy.
- 6) There is efficient time consumption.
- 7) It reduces the chances of human error.
- 8) Small in size.

DISADVANTAGES OF THE SYSTEM

- 1)The system won't work if the phone is disconnected or damaged.
- 2)Possibility of false reporting of the accident.
- 3)Damage to the sensor cannot be detected.
- 4)It may find difficulty in communication in places where there is no provision of GSM network.
- 5)The application does not have an offline mode.
- 6)Blockage of signal transmission by mountains,hugh buildings.

CHAPTER-6

CONCLUSION AND FUTURE SCOPE OF THE PROJECT

The proposed system deals with the detection of accidents.. This device offers the most realistic alternative to the inadequate emergency services given to victims of road accidents. With the aid of this technology, when an accident happens, prompt action can be taken by alerting the appropriate individuals by sending a message. By increasing the technology we can also avoid accidents by providing warnings that can stop the vehicle to overcome the accidents

In the future,we can upgrade this with different sensors such as alcohol detector,drowsiness detector,heart rate detector etc.By these technologies ,we can really prevent an accident and save lives.This can also modified by installing a controller module that takes the photographs of the accident spot that makes the tracking easier.

CHAPTER-7

APPENDIX

The code used to check the software:-

```
#include <SoftwareSerial.h>

const String EMERGENCY_PHONE = "+919778005211

#define rxPin 2

#define txPin 3

SoftwareSerial sim800(rxPin, txPin);

void setup() {

  Serial.begin(9600);

  sim800.begin(9600);

  sim800.println("AT");

  updateSerial();

  //delay(1000);

  sim800.println("AT+CLIP=1");

  updateSerial();

  //delay(1000);

  updateSerial();

  //delay(1000);

  sim800.println("AT+CMGF=1");

  updateSerial();

  //delay(1000);

  sim800.println("AT+CNMI=1,1,0,0,0");
```

```

updateSerial();

sim800.println("AT+CSMP=17,167,0,0");

updateSerial();

sim800.println("AT+CMGDA=DEL ALL");

updateSerial();

//delay(1000);

//sendAlert();

makeCall();

//sendSms();

}

void loop() { }

void sendAlert()

{

String sms_data;

sms_data = "Accident Alert!!\r";

//sms_data += "http://maps.google.com/maps?q=loc:";

//sms_data += latitude + "," + longitude;

sendSms();

}

void makeCall()

{

Serial.println("calling...");

```

```

sim800.println("ATD" + EMERGENCY_PHONE + ";");

delay(20000); //20 sec delay

sim800.println("ATH");

delay(1000); //1 sec delay

}

//void sendSms(String text)

void sendSms()

{

//return;

//sim800.print("AT+CMGF=1\r");

//delay(1000);

//sim800.print("AT+CMGS=\"" + EMERGENCY_PHONE + "\"\r");

//delay(1000);

String text="sms test\r";

sim800.print(text);

//delay(10);

sim800.write(0x1A);

delay(1000);

Serial.println("SMS Sent Successfully.");

}

void updateSerial()

{

//delay(5000);

delay(2000);

```

```

//delay(1000);

//delay(300);

while(Serial.available())//forward what Serial rcvd to softwareSerialport
sim800.write(Serial.read());

}

while(sim800.available())//forward what softwareSerial rcvd toSerial port
{

Serial.write(sim800.read());

}}

```

Code used for the accident alert:-

```

#include <TinyGPS++.h>

#include <TinyGPSPlus.h>

//#include <AltSoftSerial.h>

//#include <TinyGPS++.h>

#include<Wire.h>

#include <math.h>

#include <SoftwareSerial.h>

//const String EMERGENCY_PHONE = "+919745742608";

const String EMERGENCY_PHONE = "+919778005211";

#define rxSim 2

#define txSim 3

#define rxGPS 5

#define txGPS 6

SoftwareSerial sim800(rxSim, txSim);

```

```
//AltSoftSerial neogps;

TinyGPSPlus gps;

//gps(9,8)

SoftwareSerial gpsSerial(rxGPS, txGPS);

//String sms_status, sender_number, received_date, msg;

String latitude, longitude;

#define BUZZER 10 // LED+ BUZZER

#define xPin A0

#define yPin A1

#define zPin A2

byte updateflag;

int xaxis = 0, yaxis = 0, zaxis = 0;

int deltx = 0, delty = 0, deltz = 0;

int vibration = 2;

int devibrate = 75;

int magnitude = 0;

//int sensitivity = 20;

int sensitivity = 50;

double angle;

boolean impact_detected = false;

unsigned long time1;

unsigned long impact_time;

unsigned long alert_delay = 30000;

void setup()
```

```
{  
  
  Serial.begin(9600);  
  
  sim800.begin(9600);  
  
  //neogps.begin(9600);  
  
  //gpsSerial.begin(9600);  
  
  pinMode(BUZZER, OUTPUT);// LED  
  
  Serial.println("in setup");  
  
  gpsSerial.begin(9600);  
  
  //sms_status = "";  
  
  //sender_number = "";  
  
  //received_date = "";  
  
  //msg = "";  
  
  sim800.println("AT");  
  
  updateSerial();  
  
  //delay(1000);  
  
  sim800.println("AT+CLIP=1");  
  
  updateSerial();  
  
  //delay(1000);  
  
  sim800.println("AT+CMGD=1,4");  
  
  updateSerial();  
  
  //delay(1000);  
  
  sim800.println("AT+CMGF=1");  
  
  updateSerial();  
  
  //delay(1000);
```

```
sim800.println("AT+CNMI=1,1,0,0,0");  
updateSerial();  
sim800.println("AT+CSMP=17,167,0,0");  
updateSerial();  
sim800.println("AT+CMGDA=DEL ALL");  
updateSerial();  
/*  
sim800.println("AT");  
delay(1000);  
sim800.println("ATE1");  
delay(1000);  
sim800.println("AT+CPIN?");  
delay(1000);  
sim800.println("AT+CMGF=1");  
delay(1000);  
sim800.println("AT+CNMI=1,1,0,0,0");  
delay(1000);  
*/  
time1 = micros();  
xaxis = analogRead(xPin);  
yaxis = analogRead(yPin);  
zaxis = analogRead(zPin);  
Serial.println("Exit setup");  
}
```



```

void loop()
{
  if (micros() - time1 > 1999)
    Impact();
  if (updateflag > 0)
  {
    updateflag = 0;
    Serial.println("IMPACT !!");
    //Serial.print("Magnitude:");
    Serial.println(magnitude);
    getGps();
    //digitalWrite(BUZZER, HIGH);
    //delay(3000);
    impact_detected = true;
    impact_time = millis();
  }
  if (impact_detected == true)
  {
    //if (millis() - impact_time >= alert_delay) { //digitalWrite(BUZZER, LOW);
    //Serial.print("make call:")
    makeCall();
    delay(1000);
    Serial.print("send alert:");
    sendAlert();
  }
}

```

```

//delay(1000);//04/02

//delay(500);

delay(2000);

impact_detected = false;

impact_time = 0;

//}

while (1){

  Serial.println("CRASH!!!!");

  }

}

/*

if (digitalRead(BUTTON) == LOW) {

  delay(200);

  //digitalWrite(BUZZER, LOW);

  impact_detected = false;

  impact_time = 0;

}

while (sim800.available()) {

  parseData(sim800.readString());//FUNCTION WE WERE SEARCHING FOR

}

while (Serial.available()) {

  sim800.println(Serial.readString());

}

*/

```

```

} // LOOP END

void Impact()
{
    time1 = micros();

    int oldx = xaxis;

    int oldy = yaxis;

    int oldz = zaxis;

    xaxis = analogRead(xPin);

    yaxis = analogRead(yPin);

    zaxis = analogRead(zPin);

    vibration--;

    Serial.print("CollisionCheck...");

    Serial.println();

    //Serial.println(vibration);

    if (vibration < 0) vibration = 0;

    if (vibration > 0) return;

    deltax = xaxis - oldx;

    delty = yaxis - oldy;

    deltz = zaxis - oldz;

    magnitude = sqrt(sq(deltax) + sq(delty) + sq(deltz));

    if (magnitude >= sensitivity) //impact detected
    {
        updateflag = 1;

        vibration = devibrate;
    }
}

```

```

}
else
{
    magnitude = 0;
}}

void parseData(String buff) { // LINE 106

    Serial.println(buff);

    unsigned int len, index;

    index = buff.indexOf("\r");

    buff.remove(0, index + 2);

    buff.trim();

    if (buff != "OK") {

        index = buff.indexOf(":");

        String cmd = buff.substring(0, index);

        cmd.trim();

        buff.remove(0, index + 2);

        if (cmd == "+CMTI") {

            index = buff.indexOf(",");

            String temp = buff.substring(index + 1, buff.length());

            temp = "AT+CMGR=" + temp + "\r";

            sim800.println(temp);

        }

        else if (cmd == "+CMGR") {

            if (buff.indexOf(EMERGENCY_PHONE) > 1) {

```

```

buff.toLowerCase();

if (buff.indexOf("get gps") > 1) {

    getGps();

    String sms_data;

    sms_data = "GPS Location Data\r";

    sms_data += "http://maps.google.com/maps?q=loc:";

    sms_data += latitude + "," + longitude

    sendSms(sms_data);

}

}

}

}

else {

}

}

void getGps()

{

// Can take up to 60 seconds

boolean newData = false;

for (unsigned long start = millis(); millis() - start < 2000;start++) {

    Serial.println(" ACCIDENT!!! ");

    while (gpsSerial.available()) {

        if (gps.encode(gpsSerial.read())) {

            newData = true;

```

```

        break;
    }} }

if (newData)
{
    //latitude = String(gps.location.lat(), 6);
    //longitude = String(gps.location.lng(), 6);
    latitude = String(gps.location.lat(), 6);
    longitude = String(gps.location.lng(), 6);
    newData = false;
}
else {
    Serial.println("No GPS data is available");
    latitude = "";
    longitude = "";
}

Serial.print("Latitude= "); Serial.println(latitude);
Serial.print("Longitude= "); Serial.println(longitude);
}

void sendAlert()
{
    String sms_data;

    sms_data = "Accident Alert\r";

    sms_data += "http://maps.google.com/maps?q=loc:\r";

    //sms_data += latitude + "," + longitude;

```

```

    sendSms(sms_data);
}

void makeCall()
{
    Serial.println("calling....");

    sim800.println("ATD" + EMERGENCY_PHONE + ";");

    delay(20000); //20 sec delay

    //delay(15000);

    sim800.println("ATH");

    delay(1000); //1 sec delay
}

void sendSms(String text)
{
    //return;

    sim800.print("AT+CMGF=1\r");

    delay(1000);

    sim800.print("AT+CMGS=\"" + EMERGENCY_PHONE + "\"\r");

    delay(1000);

    sim800.print(text);

    delay(100);

    sim800.write(0x1A);

    delay(1000);

    Serial.println("SMS Sent Successfully.");
}

```

```

void updateSerial()
{
    //delay(5000);

    delay(1000);

    //delay(2000);

    while(Serial.available())//forward what Serial rcvd to softwareSerial port
    {

        sim800.write(Serial.read());

    }

    while(sim800.available())//forward what softwareSerial rcvd to Serial port
    {

        Serial.write(sim800.read());

    }

}

boolean SendAT(String at_command, String expected_answer, unsigned int timeout) {

    uint8_t x = 0;

    boolean answer = 0;

    String response;

    unsigned long previous;

    while ( sim800.available() > 0) sim800.read();

    sim800.println(at_command);

    x = 0;

    previous = millis();

    do {

```



```
if (sim800.available() != 0) {  
    response += sim800.read();  
  
    x++;  
  
    if (response.indexOf(expected_answer) > 0) {  
        answer = 1;  
  
        break;  
    }  
}  
  
}  
  
while ((answer == 0) && ((millis() - previous) < timeout));  
  
Serial.println(response);  
  
return answer;
```

CHAPTER-8

REFERENCES

- 1) Prabha, C., Sunitha, R., & Anitha, R. (2014). Automatic vehicle accident detection and messaging system using GSM and GPS modem. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 3(7), 10723-10727..
- 2) Faiz, A. B., Imteaj, A., & Chowdhury, M. (2015, November). Smart vehicle accident detection and alarming system using a smartphone. In *2015 international conference on computer and information engineering (ICCIE)* (pp. 66-69). IEEE.
- 3) Amin, M. S., Bhuiyan, M. A. S., Reaz, M. B. I., & Nasir, S. S. (2013, December). GPS and Map matching based vehicle accident detection system. In *2013 IEEE Student Conference on Research and Development* (pp. 520-523). IEEE
- 4) Aswin M,Sujitha E,Archunan P,Sandhya devi RS(2021,March), Accident prevention and alert system using Arduino international research journal of engineering and technology(p-issn :2395-0072).IRJET
- 5) Shindalkar, P., Shaikh, A. F., & Mate, C. (2017). Arduino Based Vehicle Accident Detection System. *International Journal of Innovative Research in Computer and Communication Engineering*, 4(5).
- 6) <https://www.arduino.cc/reference/en/>
- 7) <https://www.arduino.cc/glossary/en/>