

VEHICLE CRASH ALERTING SYSTEM

## A PROJECT REPORT

***Submitted by***

|  |  |  |
| --- | --- | --- |
|  |  |  |
| **GOWRISANKAR S** | **-** | **611221106032** |
| **THRAYAMBIKA P** | **-** | **611221106113** |

***In partial fulfilment for the award of the degree***

***Of***

***BACHELOR OF ENGINEERING***

***In***

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**KNOWLEDGE INSTITUTE OF TECHNOLOGY**

**KAKAPALAYAM , SALEM-637504**

**TABLE OF CONTENTS**

**S.NO TITLE PAGE NO**

**1 ABSTARCT 04**

**2 INTRODUCTION 05**

**3 OBJECTIVE 06**

**4 PROBLEM STATEMENT 06**

**5 SCOPE OF THE PROJECT 07**

**6 EXISTED SOLUTION 07**

**7 PROPOSED SYSTEM ANALYSIS 08**

**8 APPLICATIONS 09**

**9 ADVANTAGES 09**

**10 BLOCK DIAGRAM 10**

**11 COMPONENTS 11**

11.1 ARDUNIO NANO 11

11.2 ACCELEROMETER 12

11.3 GPS 13

11.4 GSM 14

11.5 LCD 15

11.6 BATTERIES 16

**12 FLOW CHART 17**

**13 HARDWARE 18**

**14 SOFTWARE 18**

**15 CIRCUIT DIAGRAM 19**

**16 SOURCE CODE 20**

**17 FUTURE SCOPE 31**

**18 CONCLUSION 32**

**19 SUMMARY 33**

**20 REFERENCE 33**

# ABSTRACT

Road accidents continue to be a major global concern, with millions of lives lost each year due to collisions and crashes. As automotive technology advances, integrating smart safety systems into vehicles becomes imperative to minimize the impact of accidents. In this context, a Vehicle Crash Alert System (VCAS) that utilizes an accelerometer, GPS, and GSM technology has emerged as a promising solution to enhance road safety. The Vehicle Crash Alert System is designed to detect and respond to crashes promptly by utilizing a combination of sensor technologies. The core components of this system include an accelerometer, a GPS module, and a GSM communication module. This paper aims to explore the design, working Road accidents continue to be a major global concern, with millions of lives lost each year due to collisions and crashes. As automotive technology advances, integrating smart safety systems into vehicles becomes imperative to minimize the impact of accidents. In this context, a Vehicle Crash Alert System (VCAS) that utilizes an accelerometer, GPS, and GSM technology has emerged as a promising solution to enhance road safety. working principles, and benefits of the VCAS in enhancing automotive safety.The benefits of a Vehicle Crash Alert System using an accelerometer, GPS, and GSM technology are manifold. Firstly, it provides a proactive safety measure, allowing for immediate assistance and potentially saving lives in critical situations. Secondly, the system enables efficient emergency response and medical care, reducing the severity of injuries and improving the overall outcome of accidents. Thirdly, the collected crash data can be utilized to conduct in-depth accident analysis, leading to better safety regulations and guidelines. In conclusion, the Vehicle Crash Alert System utilizing accelerometer, GPS, and GSM technology is a promising approach to enhance road safety and reduce the devastating impact of accidents. By combining real-time crash detection, precise location tracking, and rapid communication with emergency services, the VCAS can play a crucial role in improving automotive safety standards and protecting the lives of drivers, passengers, and pedestrians alike. As technology continues to advance, further research and development in this field are vital to continually improve the effectiveness and reliability of the VCAS.

**INTRODUCTION**

The modern world's growing reliance on automobiles has unfortunately been accompanied by a rise in road accidents, leading to a significant toll on human lives and property. As a response to this pressing issue, the development of advanced safety systems has become imperative to mitigate the impact of collisions and enhance road safety. The Vehicle Crash Alert System (VCAS) emerges as a crucial solution that leverages cutting-edge sensor technologies, including the accelerometer, GPS, and GSM, to detect crashes promptly and facilitate timely emergency responses, potentially saving lives and reducing the severity of injuries. In recent years, the acceleration in technological advancements has paved the way for sophisticated safety systems to revolutionize vehicle safety. The accelerometer, a core component of the VCAS, plays a pivotal role in this endeavour. It continuously monitors the vehicle's acceleration patterns during regular driving. However, in the event of a crash or collision, the accelerometer detects sudden and significant changes in acceleration, acting as an immediate trigger for the crash detection process. In conjunction with the accelerometer, the Vehicle Crash Alert System incorporates a Global Positioning System (GPS) module. GPS technology offers precise and real-time tracking of the vehicle's location. Moreover, the GPS data facilitates post-crash analysis and supports the development of effective safety measures to prevent similar incidents in the future. To ensure effective communication between the Vehicle Crash Alert System and external entities, such as emergency services and designated contacts, the system integrates Global System for Mobile Communications (GSM) technology. When a crash is detected by the accelerometer, the VCAS utilizes the GSM module to send real-time alerts to pre-programmed emergency contact numbers or a central monitoring station. The alerts contain vital information, including the vehicle's GPS coordinates, providing emergency responders with immediate access to critical data for rapid response and assistance.

**OBJECTIVE AND PROBLEM STATEMENT OF THE PROJECT:**

## OBJECTIVE:

* To enhance road safety by providing a real-time crash detection and notification mechanism.
* Minimize emergency response times and enhance medical assistance by providing immediate notifications to relevant authorities.
* Increase driver and passenger safety by proactively alerting emergency services in critical situations, potentially reducing the severity of injuries and fatalities

## PROBLEM STATEMENT:

* The problem is to design and implement an accurate and real-time

Vehicle crash detection system to enhance road safety.

* The goal of the project is to detect accidents and alert the rescue team

In time.

* The number of accidents is increasing due to population growth, the

High speed of the new advanced cars and the use of mobile

Telephones while driving for calling or text.

* Another scenario where the person who has the accident is unable to

Reach out for help or call the ambulance because the persons was

Injured , either the person is not awake and helpless or the accident

Occurred on an inaccessible road or away from the neighborhoods.

# SCOPE OF THE PROJECT:

* The project aims to develop an advanced vehicle crash alert system that combines accelerometer, GPS, and GSM technologies.
* The system will be designed to detect sudden changes in acceleration indicative of a crash and trigger real-time alerts.
* It will utilize GPS to accurately pinpoint the vehicle's location during emergencies.
* The GSM module will establish seamless communication with emergency services, developing an efficient crash detection algorithm, integrating hardware components, creating a user-friendly interface, and conducting extensive testing to ensure reliability and effectiveness.
* The system's focus is on enhancing road safety, minimizing response times, and potentially saving lives.

# EXISTED SOLUTION:

* The existing vehicle crash alert system typically involves the integration of various sensors and technologies within the vehicle to detect potential accidents and alert drivers and emergency services. These systems commonly use a combination of radar, lidar , cameras, and onboard computers.
* When a potential crash scenario is detected, the system evaluates the data collected from these sensors and employs sophisticated algorithms to determine if a collision is imminent. If a significant risk is identified, the system triggers immediate alerts to the driver, such as audible warnings, visual notifications on the dashboard, or haptic feedback through the steering wheel or seat.
* Additionally, some crash alert systems are designed to automatically engage certain safety mechanisms, such as pre-tensioning seat belts, adjusting the airbags, or applying emergency braking if the driver does not respond to the warnings.
* Moreover, several modern vehicles are equipped with communication technologies, like cellular or satellite connections, allowing them to transmit crash data and location information to emergency services, enabling faster response times and potentially saving lives.
* It is crucial to note that technology in this domain is continually evolving, and advancements beyond my knowledge cutoff date might have occurred.

# PROPOSED SYSTEM ANALYSIS

* The proposed Vehicle Crash Alert System is an innovative solution that

utilizes accelerometer, GPS, and GSM technologies to enhance road safety

and provide timely assistance in the event of a crash.

* The system aims to proactively detect collisions and alert emergency services, potentially reducing the severity of injuries and saving lives.
* The core of the system is the accelerometer, which continuously monitors

the vehicle's acceleration patterns. In the event of a crash, the

accelerometer promptly detects the sudden changes in acceleration,

indicating a collision has occurred.

* This triggers the crash detection process .
* To accurately determine the crashed vehicle's location, the system integrates a GPS module.
* The GPS technology provides real-time and precise coordinates, allowing emergency services to swiftly reach the accident site, even in remote or unfamiliar areas.
* The location data also aids in post-crash analysis, helping authorities understand the accident dynamics for better safety improvements.
* The GSM communication module is an essential component of the

proposed system, enabling seamless communication with emergency

services and designated contacts.

* When a crash is detected, the system uses GSM technology to send

immediate alerts to predefined emergency contact numbers or a central

monitoring station.

* The alerts include crucial crash data and GPS coordinates, providing

emergency responders with critical information to respond promptly.

# APPLICATIONS

1. Remote Location Tracking
2. Enhance Remote Location Tracking
3. Timely Medical Assistance
4. Post-Crash Analysis
5. Seamless Integration
6. Remote Location Tracking

# ADVANTAGES

# Immediate Emergency Alert

# Low maintenance

# Enhanced safety

# Real time crash detection

# Location tracking

# BLOCK DIAGRAM

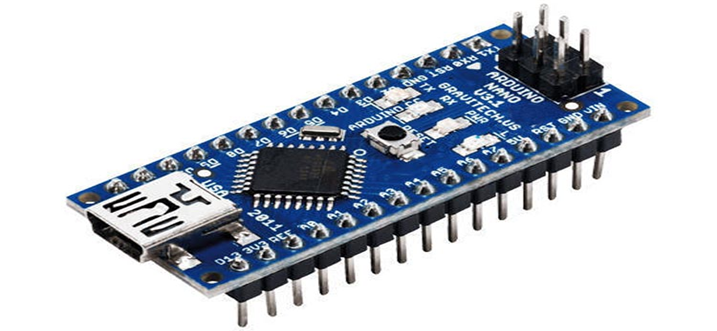
# FIGURE 1: Block diagram

# COMPONENTS:

# ARDUNIO NANO:

# 

The Arduino Nano is a popular and versatile microcontroller board designed for easy prototyping and DIY electronics projects. It is a compact version of the standard Arduino board, equipped with the same Atmel ATmega328P microcontroller, but with a smaller footprint. The Nano comes with a variety of digital and analog input/output pins, enabling it to interact with sensors, actuators, and other electronic components. Its small size makes it suitable for projects with limited space, and it can be easily integrated into wearable devices, robotics, home automation, and more. The Arduino Nano is programmable using the Arduino IDE, which provides a user-friendly environment for writing and uploading code. Additionally, it supports a broad range of libraries and community-contributed code, making it accessible to both beginners and experienced developers. Due to its versatility, ease of use, and cost effectiveness, the Arduino Nano has become a favorite among hobbyists, students, and professionals alike for bringing creative electronic projects to life.

****

**FIGURE 2: Arduino Nano**

**ACCELEROMETER ADXL345**

The ADXL345 is a widely used three-axis digital accelerometer developed

by Analog Devices. It is a versatile and precise sensor capable of measuring

acceleration in various applications. The ADXL345 uses

microelectromechanical systems (MEMS) technology to sense changes in

acceleration along three perpendicular axes (X, Y, and Z). It can measure

both static acceleration (like gravity) and dynamic acceleration (due to

motion or vibration). The sensor provides a digital output that can be easily

interfaced with microcontrollers or other digital systems. The ADXL345

offers adjustable sensitivity ranges, enabling it to detect subtle movements

or handle high-impact scenarios. Its low power consumption and compact

size make it suitable for battery-operated and portable devices. The sensor

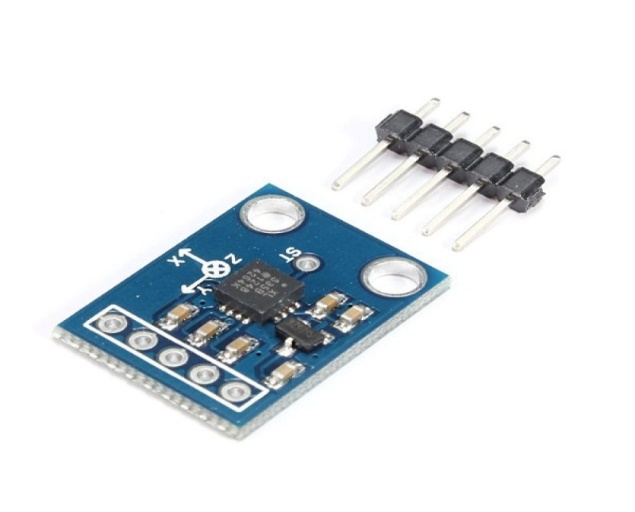
finds applications in various fields, including robotics, gaming, wearable

technology, and motion detection systems. Whether used to measure tilt,

inclination, or acceleration, the ADXL345 plays a crucial role in providing

accurate motion-related data to enhance the capabilities of a wide range of

electronic devices and systems.



# FIGURE 3: Accelerometer (ADXL345)

# GPS MODULE

The NEO-6M GPS module is a compact and highly reliable global positioning system (GPS) device widely used in various applications, from navigation and tracking systems to geotagging and location-based projects. Developed by u-blox, the NEO-6M module boasts impressive accuracy and sensitivity, capable of acquiring and tracking multiple satellite signals simultaneously. Its small form factor, low power consumption, and straightforward communication interface (usually through UART) make it popular among hobbyists and professionals alike. The module's built-in onboard flash memory allows for firmware updates, ensuring adaptability to evolving GPS technologies. Its fast time-to-first-fix (TTFF) and ability to operate in challenging environments, such as urban canyons or dense foliage, further enhance its performance. With its affordable cost and ease of integration, the NEO-6M GPS module continues to be a favored choice for numerous location-aware applications across various industries.



# FIGURE 4: GPS MODULE

# GSM MODULE

The SIM800L GSM module is a compact and versatile communication device widely used in the field of Internet of Things (IOT) and mobile communication projects. Manufactured by SIMCom, this module provides seamless global connectivity, supporting quad-band GSM/GPRS frequencies, making it suitable for use in various countries and regions. With its small form factor and low power consumption, the SIM800L is well-suited for applications where space and energy efficiency are crucial. It integrates essential features like voice communication, SMS, and GPRS data transmission, enabling devices to send and receive information over the cellular network. Additionally, it supports

embedded TCP/UDP protocols, enabling direct communication with

web servers, making it an ideal choice for IOT applications. The

module is designed with numerous GPIO pins and interfaces,

allowing seamless integration with microcontrollers and other

peripheral devices. Its ease of use and extensive documentation

make it popular among hobbyists, students, and developers looking

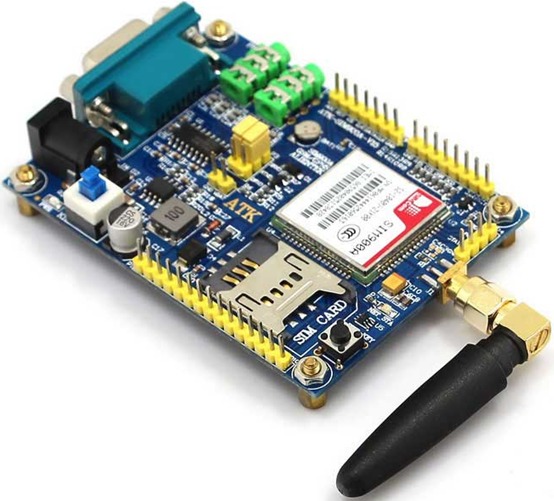
to incorporate cellular communication capabilities into their projects

with minimal complexity. Whether it's tracking devices, remote

monitoring systems, or other IOT applications, the SIM800L GSM

module provides a reliable and cost-effective solution for enabling

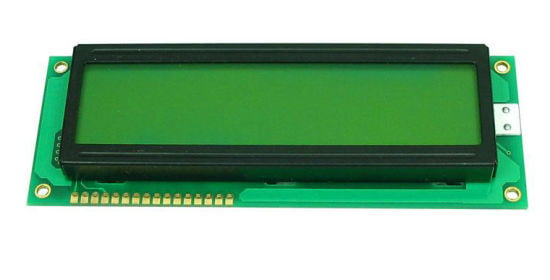
cellular connectivity.



# FIGURE 5: GSM Module

# LCD (LIQUID CRYSTAL DISPLAY)

An LCD (Liquid Crystal Display) is a widely used flat-panel technology in electronic devices, renowned for its thin and energy-efficient design. It employs a complex system of liquid crystals sandwiched between two glass layers, manipulated by electric currents to generate images. These crystals can twist and untwist to control the passage of light through the display, producing vibrant colors and sharp visuals. LCD displays are prevalent in various applications, from smartphones, tablets, and computer monitors to televisions and digital signage. Their widespread adoption is attributed to their ability to render high-definition content with exceptional clarity and reduced power consumption. Moreover, LCDs offer wider viewing angles compared to older technologies like CRT displays. As technology advances, variations like LED-backlit LCDs and OLEDs continue to improve the display quality and propel the development of next-generation visual experiences.



**FIGURE 6:** **Liquid Crystal Display**

# POWER SUPPLY

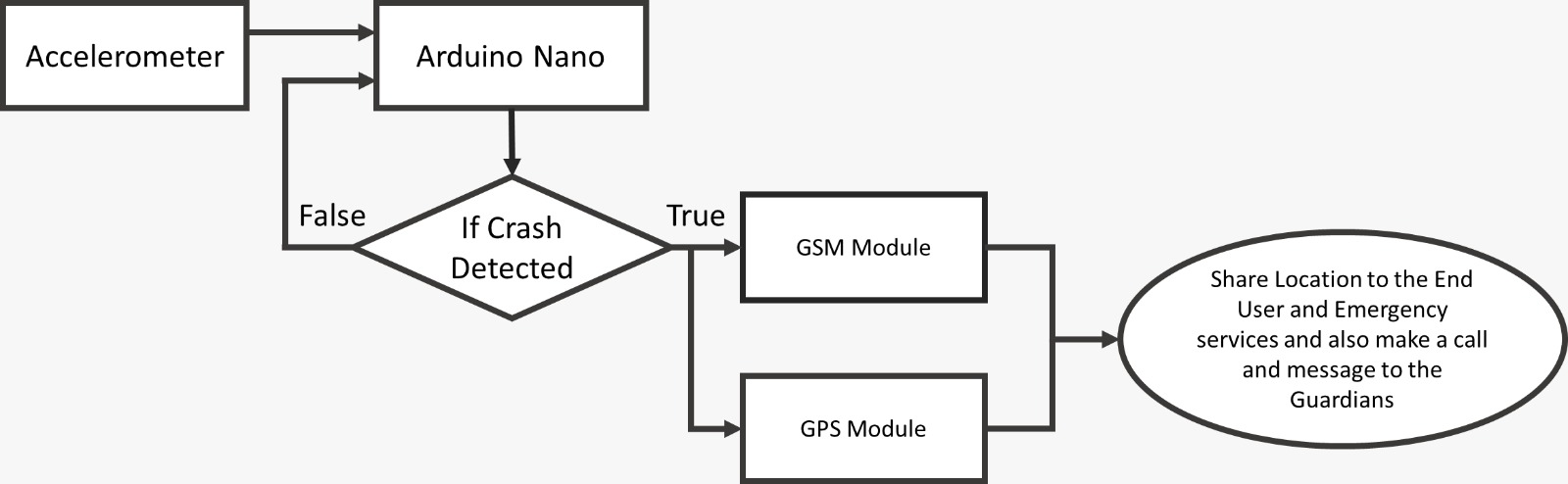
# A power supply is a fundamental cosmponent in various electronic systems, responsible for converting electrical energy from a source (such as mains power or a battery) into the appropriate voltage, current, and frequency required to power electronic devices. It ensures a stable and reliable flow of electricity, providing the necessary energy to make devices and equipment function correctly. The design of a power supply must consider factors like efficiency, voltage regulation, current capacity, and safety measures to prevent damage to connected devices. Switching power supplies are common in many modern applications due to their high efficiency and compact size. In contrast, linear power supplies are still used in certain specialized applications where noise and ripple are critical concerns. From small electronic gadgets to complex industrial machinery, power supply solutions cater to a wide range of applications, optimizing energy usage and enhancing overall performance. Ensuring a well-designed and efficient power supply is crucial for the proper functioning and longevity of electronic equipment, playing a vital role in powering our technology-driven lifestyles.

# 

# FIGURE 7: Batteries

# IMPLEMENTATIONS

The steps taken to construct a system include defining steps for requirement gathering, design and architecture, development and coding, quality control and software testing, implementation, and maintenance and support. This research's product, known as the Vehicle Crash Alert System (VCAS), was cited. This reflected the stages of the created system's development. The report described the created system and listed any potential issues that may have arisen during its creation. Sensibly following the working plan allowed for the system's development. Multiple tests and additional evaluations were conducted in order to establish whether the system is meeting the predetermined objectives and to make any necessary adjustments.



# FIGURE 8: Flow chart

# HARDWARE

We have used various components in this project

The components are as follows:

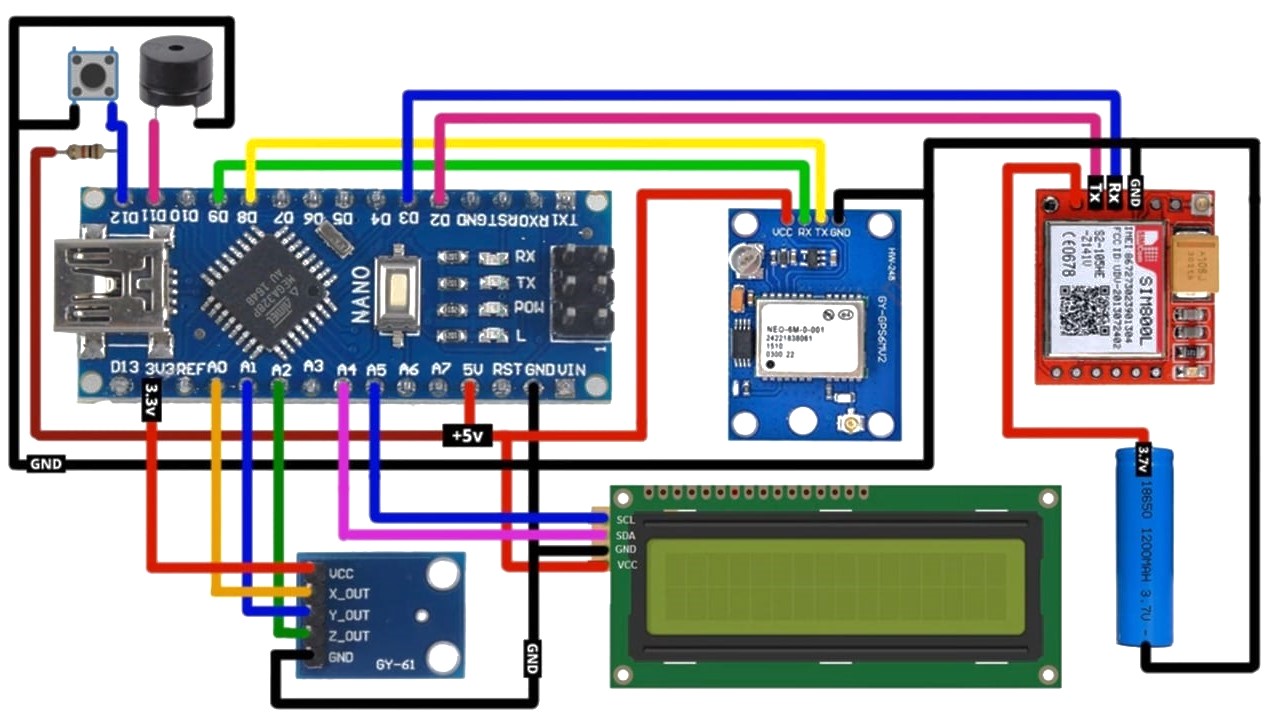
1. Arduino NANO
2. Accelerometer ADXL345
3. LCD display
4. SIM800L GSM Module
5. Neo6mGPS
6. Power supply

# SOFTWARE

The Arduino Uno is programmed using the Arduino Software (IDE), which is an Integrated Development Environment common to all Arduino boards and can be run both online and offline. C or Embedded C is used to write the program.

1. We have used two different software’s for the development of this project.
2. It is required to have a laptop in order to develop this project.
3. We have used third-party tools like Thingspeak and proteus
4. One has to know how to work with Arduino ide software.
5. One should have knowledge of the understanding of android studio and how to write and build C or Embedded C.

# CIRCUIT DIAGRAM



**FIGURE 9: Circuit diagram**

# SOURCE CODE

# #include<LiquidCrystal\_I2C.h>

# #include <AltSoftSerial.h>

# #include <TinyGPS++.h>

# #include <SoftwareSerial.h>

# #include <math.h>

# #include<Wire.h>

# //must add i2c lcd address use i2c-scanner.ino file

# LiquidCrystal\_I2C lcd(0x27, 16, 2);

# //--------------------------------------------------------------

# //emergency phone number with country code

# const String EMERGENCY\_PHONE = "ENTER\_EMERGENCY\_PHONE\_NUMBER";

# //--------------------------------------------------------------

# //GSM Module RX pin to Arduino 3

# //GSM Module TX pin to Arduino 2

# #define rxPin 2

# #define txPin 3

# SoftwareSerial sim800(rxPin,txPin);

# //--------------------------------------------------------------

# //GPS Module RX pin to Arduino 9

# //GPS Module TX pin to Arduino 8

# AltSoftSerial neogps;

# TinyGPSPlus gps;

# //--------------------------------------------------------------

# String sms\_status,sender\_number,received\_date,msg;

# String latitude, longitude;

# //--------------------------------------------------------------

# #define BUZZER 12

# #define BUTTON 11

# //--------------------------------------------------------------

# #define xPin A1

# #define yPin A2

# #define zPin A3

# //--------------------------------------------------------------

# byte updateflag;

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

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

# int vibration = 2, devibrate = 75;

# int magnitude = 0;

# int sensitivity = 20;

# double angle;

# boolean impact\_detected = false;

# //Used to run impact routine every 2mS.

# unsigned long time1;

# unsigned long impact\_time;

# unsigned long alert\_delay = 30000; //30 seconds

# //--------------------------------------------------------------

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* setup() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void setup()

# {

# //--------------------------------------------------------------

# //Serial.println("Arduino serial initialize");

# Serial.begin(9600);

# //--------------------------------------------------------------

# //Serial.println("SIM800L serial initialize");

# sim800.begin(9600);

# //--------------------------------------------------------------

# //Serial.println("NEO6M serial initialize");

# neogps.begin(9600);

# //--------------------------------------------------------------

# pinMode(BUZZER, OUTPUT);

# pinMode(BUTTON, INPUT\_PULLUP);

# //--------------------------------------------------------------

# //initialize lcd screen

# lcd.begin();

# // turn on the backlight

# lcd.backlight();

# lcd.clear();

# //--------------------------------------------------------------

# sms\_status = "";

# sender\_number="";

# received\_date="";

# msg="";

# //--------------------------------------------------------------

# sim800.println("AT"); //Check GSM Module

# delay(1000);

# //SendAT("AT", "OK", 2000); //Check GSM Module

# sim800.println("ATE1"); //Echo ON

# delay(1000);

# //SendAT("ATE1", "OK", 2000); //Echo ON

# sim800.println("AT+CPIN?"); //Check SIM ready

# delay(1000);

# //SendAT("AT+CPIN?", "READY", 2000); //Check SIM ready

# sim800.println("AT+CMGF=1"); //SMS text mode

# delay(1000);

# //SendAT("AT+CMGF=1", "OK", 2000); //SMS text mode

# sim800.println("AT+CNMI=1,1,0,0,0"); /// Decides how newly arrived SMS should be handled

# delay(1000);

# //SendAT("AT+CNMI=1,1,0,0,0", "OK", 2000); //set sms received format

# //AT +CNMI = 2,1,0,0,0 - AT +CNMI = 2,2,0,0,0 (both are same)

# //--------------------------------------------------------------

# time1 = micros();

# //Serial.print("time1 = "); Serial.println(time1);

# //--------------------------------------------------------------

# //read calibrated values. otherwise false impact will trigger

# //when you reset your Arduino. (By pressing reset button)

# xaxis = analogRead(xPin);

# yaxis = analogRead(yPin);

# zaxis = analogRead(zPin);

# //--------------------------------------------------------------

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* loop() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void loop()

# {

# //--------------------------------------------------------------

# //call impact routine every 2mS

# if (micros() - time1 > 1999) Impact();

# //--------------------------------------------------------------

# if(updateflag > 0)

# {

# updateflag=0;

# Serial.println("Impact detected!!");

# Serial.print("Magnitude:"); Serial.println(magnitude);

# getGps();

# digitalWrite(BUZZER, HIGH);

# impact\_detected = true;

# impact\_time = millis();

# 

# lcd.clear();

# lcd.setCursor(0,0); //col=0 row=0

# lcd.print("Crash Detected");

# lcd.setCursor(0,1); //col=0 row=1

# lcd.print("Magnitude:"+String(magnitude));

# }

# //--------------------------------------------------------------

# if(impact\_detected == true)

# {

# if(millis() - impact\_time >= alert\_delay) {

# digitalWrite(BUZZER, LOW);

# makeCall();

# delay(1000);

# sendAlert();

# impact\_detected = false;

# impact\_time = 0;

# }

# }

# 

# if(digitalRead(BUTTON) == LOW){

# delay(200);

# digitalWrite(BUZZER, LOW);

# impact\_detected = false;

# impact\_time = 0;

# }

# //--------------------------------------------------------------

# while(sim800.available()){

# parseData(sim800.readString());

# }

# //--------------------------------------------------------------

# while(Serial.available()) {

# sim800.println(Serial.readString());

# }

# //--------------------------------------------------------------

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* Impact() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void Impact()

# {

# //--------------------------------------------------------------

# time1 = micros(); // resets time value

# //--------------------------------------------------------------

# int oldx = xaxis; //store previous axis readings for comparison

# int oldy = yaxis;

# int oldz = zaxis;

# xaxis = analogRead(xPin);

# yaxis = analogRead(yPin);

# zaxis = analogRead(zPin);

# 

# //--------------------------------------------------------------

# //loop counter prevents false triggering. Vibration resets if there is an impact. Don't detect new changes until that "time" has passed.

# vibration--;

# //Serial.print("Vibration = "); Serial.println(vibration);

# if(vibration < 0) vibration = 0;

# //Serial.println("Vibration Reset!");

# 

# if(vibration > 0) return;

# //--------------------------------------------------------------

# deltx = xaxis - oldx;

# delty = yaxis - oldy;

# deltz = zaxis - oldz;

# 

# //Magnitude to calculate force of impact.

# magnitude = sqrt(sq(deltx) + sq(delty) + sq(deltz));

# 

# //N

# if (magnitude >= sensitivity) //impact detected

# {

# updateflag=1;

# // reset anti-vibration counter

# vibration = devibrate;

# }

# 

# else

# {

# //if (magnitude > 15)

# //Serial.println(magnitude);

# //reset magnitude of impact to 0

# magnitude=0;

# }

# 

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* parseData() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void parseData(String buff){

# Serial.println(buff);

# unsigned int len, index;

# //--------------------------------------------------------------

# //Remove sent "AT Command" from the response string.

# 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);

# //Serial.println(buff);

# //--------------------------------------------------------------

# if(cmd == "+CMTI"){

# //get newly arrived memory location and store it in temp

# //temp = 4

# index = buff.indexOf(",");

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

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

# //AT+CMGR=4 i.e. get message stored at memory location 4

# sim800.println(temp);

# }

# //--------------------------------------------------------------

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

# //extractSms(buff);

# //Serial.println(buff.indexOf(EMERGENCY\_PHONE));

# if(buff.indexOf(EMERGENCY\_PHONE) > 1){

# buff.toLowerCase();

# //Serial.println(buff.indexOf("get gps"));

# 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{

# //The result of AT Command is "OK"

# }

# 

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* getGps() Function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void getGps()

# {

# // Can take up to 60 seconds

# boolean newData = false;

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

# while (neogps.available()){

# if (gps.encode(neogps.read())){

# newData = true;

# break;

# }

# }

# }

# 

# if (newData) //If newData is true

# {

# 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("Lngitude= "); Serial.println(longitude);

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* sendAlert() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void sendAlert()

# {

# String sms\_data;

# sms\_data = "Accident Alert!!\r";

# sms\_data += "http://maps.google.com/maps?q=loc:";

# sms\_data += latitude + "," + longitude;

# sendSms(sms\_data);

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* makeCall() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# void makeCall()

# {

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

# sim800.println("ATD"+EMERGENCY\_PHONE+";");

# delay(20000); //20 sec delay

# sim800.println("ATH");

# delay(1000); //1 sec delay

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* sendSms() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# 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); //ascii code for ctrl-26 //sim800.println((char)26); //ascii code for ctrl-26

# delay(1000);

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

# }

# /\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# \* SendAT() function

# \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

# boolean SendAT(String at\_command, String expected\_answer, unsigned int timeout){

# uint8\_t x=0;

# boolean answer=0;

# String response;

# unsigned long previous;

# 

# //Clean the input buffer

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

# sim800.println(at\_command);

# 

# x = 0;

# previous = millis();

# //this loop waits for the answer with time out

# do{

# //if there are data in the UART input buffer, reads it and checks for the asnwer

# if(sim800.available() != 0){

# response += sim800.read();

# x++;

# // check if the desired answer (OK) is in the response of the module

# if(response.indexOf(expected\_answer) > 0){

# answer = 1;

# break;

# }

# }

# }while((answer == 0) && ((millis() - previous) < timeout));

# Serial.println(response);

# return answer;

# }

# FUTURE SCOPE

The future scope of a vehicle crash alert system is poised to revolutionize road safety and reduce the number of fatalities and injuries on the roads. As technology continues to advance rapidly, so will the capabilities of these systems. In the near future, vehicle crash alert systems will be seamlessly integrated with autonomous driving technologies, creating a comprehensive safety net for passengers and pedestrians alike.

One of the key advancements will be the utilization of artificial intelligence and machine learning algorithms to predict and prevent accidents before they occur. These systems will be able to analyze real-time data from various sensors, including cameras, LiDAR, radar, and vehicle-to-vehicle communication, to identify potential collision risks and alert drivers in a split second. Moreover, the integration of V2X (vehicle-to-everything) communication will enable vehicles to share vital information with the surrounding infrastructure, leading to collective hazard awareness and better traffic management.

As the technology becomes more sophisticated, crash alert systems will not only notify drivers but also autonomously take evasive actions, such as applying brakes or steering away from imminent danger. Additionally, vehicle-to-pedestrian communication will play a crucial role in safeguarding vulnerable road users, as pedestrians and cyclists can receive alerts on their devices when they're in a hazardous position near vehicles.

The future will witness increased collaboration between automotive manufacturers, tech companies, and governments to establish standardized protocols and regulations for these systems, ensuring seamless integration across various vehicle types and making them a fundamental aspect of road safety. Ultimately, the future of vehicle crash alert systems holds the promise of significantly reducing accidents and making roads safer for everyone.

# CONCLUSION

The vehicle crash detection system represents a remarkable advancement in automotive safety, providing an intelligent and reliable solution to mitigate the consequences of accidents. Through the integration of cutting-edge technologies, including advanced sensors, artificial intelligence, and real-time data processing, the system exhibits exceptional proficiency in identifying and responding to collision events. Upon detecting a potential crash, the system swiftly analyzes various parameters, such as impact force, vehicle speed, and trajectory, to accurately determine the severity of the collision. Depending on the situation, the system activates an array of safety measures, including automatic emergency braking, seatbelt pretensions, and airbag deployment, all within milliseconds to protect occupants from harm.

Moreover, the crash detection system incorporates vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication capabilities, enabling it to share critical crash data with nearby vehicles and relevant authorities. This collaborative approach enhances overall road safety by warning nearby drivers, optimizing traffic flow, and streamlining emergency response efforts. Additionally, the system's data logging and analysis features offer valuable insights for post-crash investigations and vehicle design improvements.

In conclusion, the vehicle crash detection system sets a new standard for automotive safety, demonstrating its potential to save countless lives, prevent injuries, and revolutionize the way we approach road safety. As technology continues to evolve, further refinements and widespread adoption of this system hold the promise of significantly reducing road accidents and creating a safer driving environment for all.

# SUMMARY

After the study is over, the objectives have been developed as the basis for working. The "Vehicle Crash Alert System" system, which was intended to help the community gradually lower the number of fatal accident deaths, achieved the researchers' overall goal. The society will undergo a significant transformation as a result.

# REFERENCE

**[1]** M. Schoeberl, “A java processor architecture for embedded real-

timesystems,” Journal of Systems Architecture, vol. 54, no. 1-2, pp. 265–

286, 2008.

**[2]** P. Garcia, K. Compton, M. Schulte, E. Blem, and W. Fu, “An overviewof reconﬁgurable hardware in embedded systems,” EURASIP Journalon Embedded Systems, vol. 2006, no. 1, p. 056320, 200

**[3]** Hasnat G.N.T., Kabir M.A., Hossain M.A. (2018) Major Environmental Issues and Problems of South Asia, Particularly Bangladesh. In: Hussain C. (eds) Handbook of Environmental Materials Management. Springer, Cham

[**4**] M. Hasan, P. Biswas, M. T. I. Bilash and M. A. Z. Dipto, "Smart Home Systems: Overview and Comparative Analysis," 2018 Fourth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Kolkata, India, 2018, pp. 264-268.

**[5]** A. H. Siddique, B. Barkat and M. Poshtan, "Experimental setup of wide area monitoring using Zigbee IEEE 802.15.4 technology and RF FM technique," World Congress on Sustainable Technologies (WCST-2012), London, 2012, pp. 44-48.

**[6]** M. Hasan, A. Z. Dipto, M. S. Islam, A. Sorwar, S. Islam, “A Smart Semi-Automated Multifarious Surveillance Bot for Outdoor Security Using Thermal Image Processing,” Advances in Networks, vol. 7, no. 2, pp. 21-28, 2019.

**[7]** S. Paul, P. D. Nath, N. M. Abdus Sattar and H. U. Zaman, "rTraffic - a realtime web application for traffic status update in the streets of Bangladesh," 2017 International Conference on Research and Innovation in Information Systems (ICRIIS), Langkawi, 2017, pp. 1-6.

**[8]** H. U. Zaman, J. Khisha, N. Zerin and M. H. Jamal, "Speech responsive mobile robot for transporting objects of different weight categories," 2017 18th International Conference on Advanced Robotics (ICAR), Hong Kong, 2017, pp. 3