UGC & Govt. Approved

Sonargaon University (SU)

সোনারগাঁও ইউনিভার্সিটি (এসইউ)

"GSM (Global System for Mobile Communications) Based Accident Alerting System"

A Thesis

By

Rakibul Islam Rony	ID: EEE2001019130
Mubin Uddin Antu	ID: EEE2001019190
Mohammad Mosla Uddin Imon	ID: EEE2001019060
Sourav Sikder	ID: EEE2001019026
Jaber Al Sayef	ID: EEE2001019044
Md. Amit Hasan	ID: EEE2001019021

Supervisor

Shithi Dev

Lecturer

Department of Electrical and Electronic Engineering

Sonargaon University (SU)

Dhaka-1215, Bangladesh

January, 2024

“GSM (Global System for Mobile Communications) Based Accident Alerting System”

A report submitted to the Department of Electrical and Electronic Engineering, Sonargaon University of Bangladesh in partial fulfillment of the requirements for the Award of Degree of Bachelor of Science in Electrical and Electronic Engineering.

Submitted by

Rakibul Islam Rony	ID: EEE2001019130
Mubin Uddin Antu	ID: EEE2001019190
Mohammad Mosla Uddin Imon	ID: EEE2001019060
Sourav Sikder	ID: EEE2001019026
Jaber Al Sayef	ID: EEE2001019044
Md. Amit Hasan	ID: EEE2001019021

Supervised by

Shithi Dev

Lecturer

Department of Electrical and Electronic Engineering

Sonargaon University (SU)

Dhaka-1215, Bangladesh

January, 2024

DECLARATION

We do hereby solemnly declare that, the work presented here in this project report has been carried out by us and has not been previously submitted to any University/ Organization for award of any degree or certificate

We hereby ensure that the works that has been prevented here does not breach any existing copyright.

We further undertake to indemnify the university against any loss or damage arising from breach of the foregoing obligation.

Rakibul Islam Rony
EEE2001019130

Sourav Sikder
EEE2001019026

Mubin Uddin Antu
EEE2001019190

Jaber Al Sayef
EEE2001019044

Mohammad Mosla Uddin Imon
EEE2001019060

Md. Amit Hasan
ID: EEE2001019021

ACKNOWLEDGEMENT

First, we started in the name of almighty Allah. This thesis is accomplished under the supervision of Shithi Dev, Lecturer, Department of Electrical and Electronic Engineering, Sonargaon University. It is a great pleasure to acknowledge our profound gratitude and respect to our supervisor for this consistent guidance, encouragement, helpful suggestion, constructive criticism and endless patience through the progress of this work. The successful completion of this thesis would not have been possible without his persistent motivation and continuous guidance.

The authors are also grateful to **Name**, Head of the Department of Electrical and Electronic Engineering and all respect teachers of the Mechanical Engineering Department for their co-operation and significant help for completing the thesis work successfully.

ABSTARCT

The GSM-Based Accident Sensing System with GPS Integration is an innovative approach to enhance road safety by deploying an intelligent and responsive system in vehicles. This system utilizes a combination of sensors and communication modules, including Node MCU, GSM, and GPS, to detect and report accidents promptly. In the event of a sudden impact or unusual vehicle behavior, the system triggers an immediate response mechanism. The project employs accelerometer or gyroscopes to detect abrupt changes in acceleration, indicative of a potential accident. The Node MCU, functioning as the central controller, processes this sensor data and, in conjunction with the GPS module, determines the precise location of the vehicle. In the case of an accident, the system activates the GSM module to send a detailed SMS alert to a predefined emergency contact number. This alert includes critical information about the accident and the current GPS coordinates, enabling swift emergency response. A buzzer is integrated into the system to audibly signal the accident occurrence, drawing attention to the site and potentially expediting assistance. The combination of GPS technology ensures that emergency services can quickly locate and respond to the incident, improving overall response times in critical situations.

TABLE OF CONTENTS

Letter of Transmittal		ii
Declaration		iii
Acknowledgement		iv
Abstract		v
Table of Contents		vi
Table of Figure		viii
CHAPTER-1	**INTRODUCTION**	
1.1 Introduction		1
1.2 Scope of use		1
1.3 Motivation		2
1.4 Objective		2
1.4 Structure of the Projects		2
CHAPTER 2	**LITERATURE REVIEW**	
2.1 Introduction		4
2.2 Literature Review		4
2.3 Summary		5
CHAPTER-3	**HARDWARE AND SOFTWARE ANALYSIS**	
3.1 Introduction		06
3.2 Hardware List		06
3.3 Arduino Nano		06
3.4 Node MCU		10
3.5 Buck Converter		12
3.6 Gear Motor		14
3.7 Motor Driver		15
3.8 Resistor		18
3.9 GPS Module		19
3.10 GSM Module		21

3.11 Wheel 23
3.12 Arduino Software 24
3.13 Easy EDA Software 25
3.14 Remote XY 26

CHAPTER-4 METHODOLOGY
4.1 Methodology 30
4.2 Working Step Chart 30
4.3 Block Diagram 31
4.4 Schematic Diagram 31
4.5 Working Principle 32
4.6 Our Final Project View 32

CHAPTER-5 RESULT AND DISCUSSION
5.1 Result 33
5.2 Discussion 33
5.3 Advantages 33
5.4 Application 34
5.5 Limitations 34

CHAPTER-6 CONCLUSION
6.1 Conclusion 35
6.2 Future Scope 35

Reference 36
Appendix 38

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO
3.1	Arduino Nano	07
3.2	Arduino Nano Schematic Diagram	07
3.3	How Arduino Nano Looks like	08
3.4	Micro-controller IC AT mega 328p	09
3.5	Node MCU	10
3.6	Node MCU Schematic Diagram	11
3.7	Node MCU Pin Out	11
3.8	DC-DC Buck Converter	13
3.9	Gear Motor	14
3.10	Motor Driver IC L293D	15
3.11	L293D Circuit Diagram	17
3.12	Resistor	18
3.13	Schematic Diagram of GPS	19
3.14	NEO-6m GPS	20
3.15	GSM Module	21
3.16	GSM Module Introducing	21
3.17	GSM Module Pin out	22
3.18	Arduino Software Interface IDE	24
3.19	Easy EDA Software Interface	26
4.1	Working Step Chart	30
4.2	Block Diagram	31
4.3	Schematic Diagram	31
4.4	Our Final System Overview	32

CHAPTER 1

INTRODUCTION

1.1 Introduction

As far as we know, traffic accidents are on the rise and it has become one of the main problems in cities. This is because of the high use of cars like this, the increase in bicycle accidents, as well as the speeding of drivers. And the reason is that in the absence of advanced technology, the severity of accidents is declining. Appropriate, efficient, and standardized solutions need to be put in place to reduce the incidence of accidents in the country. At present, there is no advanced, efficient technology to reduce and detect accidents.

It also increases the likelihood of the victim's death by delaying the arrival of the ambulance to the scene of the accident, as well as delaying the delivery of information about the health of the victim to the hospital. The existing system can detect accidents and track the car, and it will take more time to reach the hospital, so the system will delay the treatment of the victim. In the proposed system, the time between the accident and the provision of necessary facilities is delayed and the victim is not treated with the necessary medical facilities.

Accidents are common in urban areas, and many of these types of accidents can be easily handled, but some accidents occur when the visibility is low at night, making it difficult for the ambulance driver to locate the scene. Citizen phone calls help. If the driver knows the exact location of the accident, the time between the scene and the hospital will be significantly reduced. The main purpose of this article is to help reduce the time factor in the event of an accident. There are many cases of people who have an accident at night and have lost consciousness, so it takes hours for someone to find out and report it to the government. So saving such precious time really saves lives.

1.2 Scopes of use

- In rescue accident car
- Detect car location in emergency situation

1.3 Motivation

In the era of development, advancement and pace we attained efficiency, mobility, flexibility and comfort but we need safe guard as well as. We, as students of electrical side has thus got motivation to design such a system that ensures protection of different vehicle and aware us even if we are distant from this vehicle. Knowing the location of the vehicle after the accident, if rescue workers are sent there soon, people's lives will be saved. This project physically gives us little hope to advance a bit for protection of accident vehicle.

1.4 Objective

The main objective of this project is to develop a smart accident detection and rescue system, the vehicle accident detection, GPS tracking and automatic notification to rescue team. Our objectives are pointed out below:

- To study of the **GSM (Global System for Mobile Communications) Based Accident Alerting System.**
- To design and construct of a GSM Based **Accident Alerting System** with Real Time GSM based Vehicle Accident Alert with GPS Location Tracking System.
- To implement and study of take proper step to reduce the accident.
- To take necessary notes from the project for future improvements.

1.5 Structure of the Project

This Project is organized as follows:

Chapter 1 Introduction: The first chapter contains the statement of the introduction, our background study for the project, objectives of the study, methodology used in the project and the project outline.

Chapter 2 Literature Review: The chapter two contains our literature review part.

Chapter 3 Hardware and Software Analysis: Chapter three describes the theoretical model. Here we mainly discuss about Hardware and software development of our project etc.

Chapter 4 Methodology: Here we mainly discuss about proposed system architecture in details with having block diagram, circuit diagram, project working principle, complete project image of our project etc.

Chapter 5 Results and Discussion: Chapter five deals with the result, advantages, application and discussion about our project advantages and application.

Chapter 6 Conclusion: all about our project conclusion, limitations and future scope.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this section topics related to **GSM (Global System for Mobile Communications) Based Accident Alerting System** are included. These provide a sampling of problems appropriate for application of GSM (Global System for Mobile Communications) Based Accident Sensing System. The references are summarized below.

2.2 Literature Review

Hari Sankar et al., [1] recommend a full response to each collision with a vehicle identification and rescue vehicle. When a car collides with a car identification device in a car, the entire server sends a nearby ambulance with the car to the collision. An Android device using an ambulance will help drivers quickly and securely collect area details. Automation of collision by recognizing and sending the vehicle, along with training the ambulance driver.

Nicky Kattukkaran and others. The relevant accelerator in the car recognizes the inclination of the car and the heartbeat sensor in the user's physics recognizes the abnormality of the heartbeat and understands the importance of the car. Thus, the devices select the selector via Bluetooth and send the laptop to a smartphone connected to the accelerator and heart rate sensor. The Android device on the mobile phone sends text content material text messages to nearby clinics and friends. In addition, the application shares the actual location of the time-saving hospital.

Adnan bin Fayez et al., [3] have developed a fully-fledged Android system that can detect the situation unexpectedly and send emergency warning messages to nearby police stations and fitness centers. The device is equipped with an external pressure sensor that removes the external pressure of the body. It measures speed and switching points with GPS and accelerator sensors, respectively, on Android phones. By checking the conditions, this software program is helpful in reducing false alarm fees.

D Selvathi et al. The prevention area includes Smart Helmet. If these two preconditions are not met, the rails will no longer be on the engine. The microcontroller controls the transmission capability and fires as a result of the retreat. In addition, the component can detect the collision with a car in any area and assess the collision of the car with the predefined numbers with the GSM module. The micro-controller continuously archives all the parameters of the car to prevent and acknowledge the collision of the car.

Chunxiao Liao, et. al introduced the "Mobile Shore-Based Smart Traffic Accident Detection System" in 2017. The article presents a framework for smart car crashes based on the nearby mobile Edge computer-based, idle and recyclable and car-certified proof. Our framework uses basic mobile phones to increase speed and speed, and to identify images that display error-prone scenes, to recognize computerized error-detection if false positives occur, and to advise environmental factors and divisions, such as clinics and transportation departments.

Sanjana. K.R, et. In 2015, the Emergency Services Center introduced the "Smart Traffic Lights with Automatic Rescue System." They used a sensor to create a frame that automatically identifies street accidents, advising them to use GSM to manage crises and close them by family members. It is fully computerized, using Google's guides to locate the fault, control traffic lights and reach the emergency center in a timely manner. This framework will be viable in countries with a large population, such as India.

2.3 Summary

The above has been discussed in detail in the past few literature's which has given us a lot of motivation to do this project.

CHAPTER 3

HARDWARE AND SOFTWARE ANALYSIS

3.1 Introduction

In this section, we will discuss elaborately about our hardware design of "**GSM (Global System for Mobile Communications) Based Accident Alerting System**" and the component description, features, working procedure and discription of our all equipment. The system hardware fabricates composed of micro-controller unit, GPS Module, GSM Module, Gear Motor, Battery, Buck Converter, Motor Driver and many more related components.

3.2 Hardware List

- Arduino Nano
- Node MCU
- GPS Module
- GSM Module
- Gear Motor
- Motor Driver
- Wheel
- Battery
- Buck Converter
- Capacitor
- Resistor

3.3 Arduino Nano

Arduino is open-source electronics prototyping platform based on flexible, easy-to-use hardware and software. It's intended for artists, designers, hobbyists, and anyone interested in creating interactive objects or environments. Arduino can sense the environment by receiving input from a variety of sensors and can affect its surroundings by controlling Lights, motors, and other actuators.

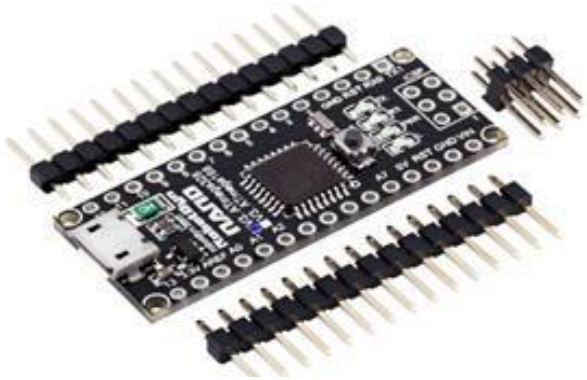


Figure 3.1: Arduino Nano

The micro-controller on the board is programmed using the Arduino programming language (based on Wiring) and the Arduino development environment (based on Processing). Arduino projects can be stand-alone or they can communicate with software on running on a computer (e.g. Flash, Processing, Maxims'). Arduino Nano is a surface mount breadboard embedded version with integrated USB. It is a small, complete, and breadboard friendly component. It has everything that Decimal/ Duemilanove has (electrically) with more analog input pins and onboard +5V AREF jumper. Physically, it is missing power jack. The Nano can automatically sense and switch to the higher potential source of power.

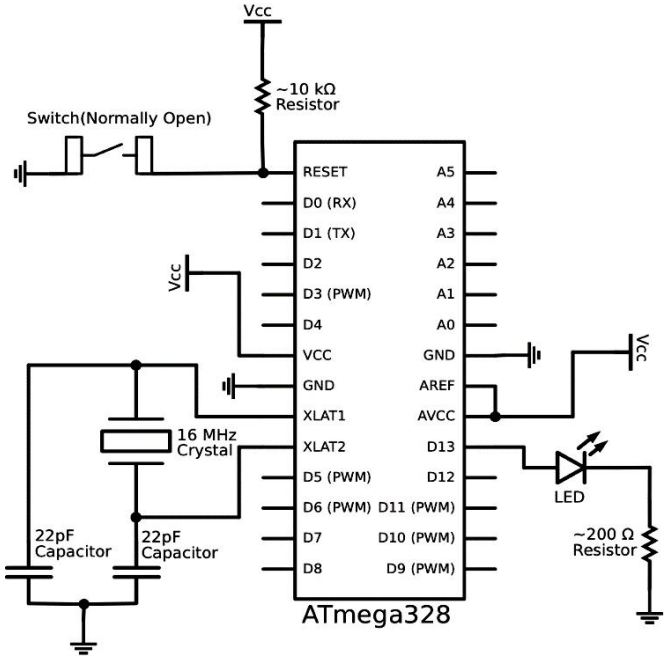


Figure 3.2: Arduino Nano Schematic Diagram

Nano's got the breadboard-ability of the Boarding and the Minibus with smaller footprint than either, so users have more breadboard space. It's got a pin layout that works well with the Mini or the Basic Stamp (TX, RX, ATN, and GND on one top, power and ground on the other). This new version 3.0 comes with ATMEGA328 which offer more programming and data memory space. It has two layers. That make it easier to hack and more affordable. One of the best features of Arduino Nano is, it's easy to use, compact and also small.

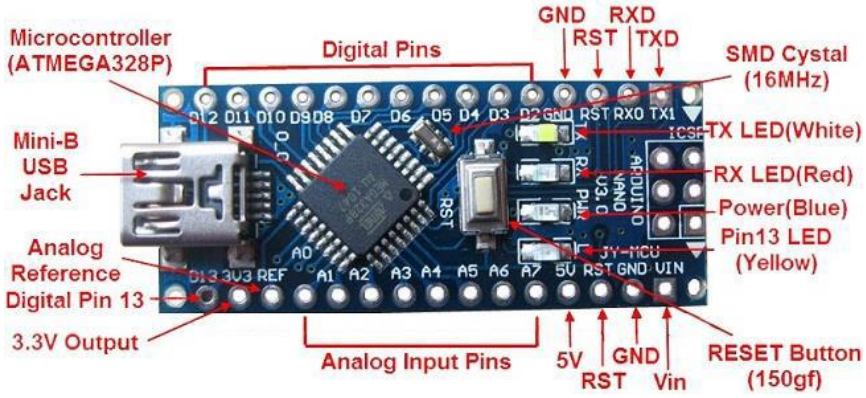


Figure 3.3: How Arduino Nano looks like

Specifications:

- Micro-controller: Atmel ATmega328
- Operating Voltage (logic level): 5 V
- Input Voltage (recommended): 7-12 V
- Input Voltage (limits): 6-20 V
- Digital I/O Pins: 14 (of which 6 provide PWM output)
- Analog Input Pins: 8
- DC Current per I/O Pin: 40 mA
- Flash Memory: 32 KB (of which 2KB used by boot loader)
- SRAM : 2 KB
- EEPROM: 1 KB
- Clock Speed: 16 MHz
- Dimensions: 0.70" x 1.70"

Features:

- Automatic reset during program download
- Power OK blue LED
- Green (TX), red (RX) and orange (L) LED
- Auto sensing/switching power input
- Small mini-B USB for programming and serial monitor
- ICSP header for direct program download
- Manual reset switch

Micro-controller IC ATmega328p

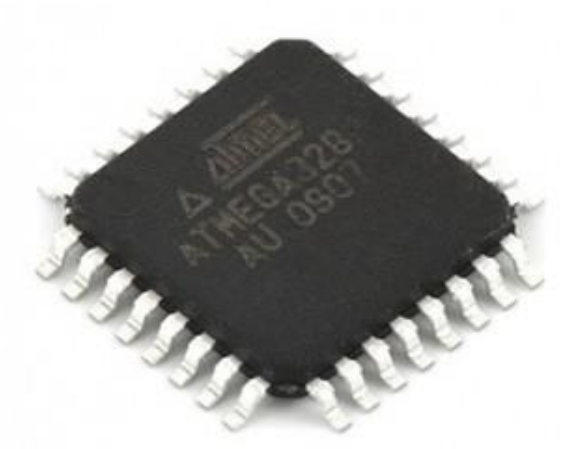


Figure 3.4: Micro-controller IC AT mega 328p

The high-performance Microchip Pico Power 8-bit AVR RISC-based micro-controller combines 32KB ISP flash memory with read-while-write capabilities, 1024B EEPROM, 2KB SRAM, 23 general purpose I/O lines, 32 general purpose working registers, three flexible timer/counters with compare modes, internal and external interrupts, serial programmable USART, a byte-oriented 2-wire serial interface, SPI serial port, a 6-channel 10-bit A/D converter (8-channels in TQFP and QFN/MLF packages), programmable watchdog timer with internal oscillator, and five software selectable power saving modes. The device operates between 1.8-5.5 volts. By executing powerful instructions in a single clock cycle, the device achieves throughput's approaching 1 MIPS per MHz, balancing power consumption and processing speed.

3.4 Node MCU

Node MCU is an open source firmware for which open source prototyping board designs are available. The name "Node MCU" combines "node" and "MCU" (micro-controller unit). The term "Node MCU" strictly speaking refers to the firmware rather than the associated development kits. Both the firmware and prototyping board designs are open source.

The firmware uses the Lua scripting language. The firmware is based on the eLua project, and built on the Espressif Non-OS SDK for ESP8266. It uses many open source projects, such as lua-cjson and SPIFFS. Due to resource constraints, users need to select the modules relevant for their project and build a firmware tailored to their needs. Support for the 32-bit ESP32 has also been implemented.

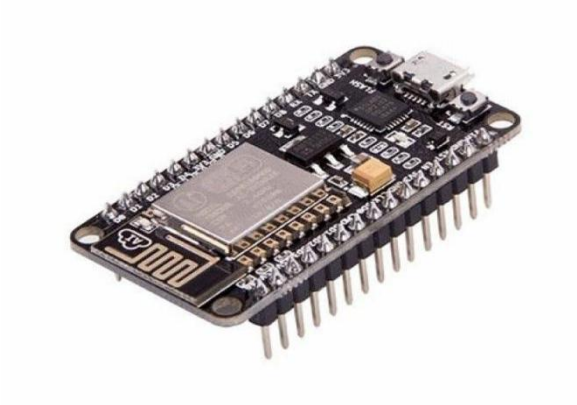


Figure 3.5: Node MCU

The prototyping hardware typically used is a circuit board functioning as a dual in-line package (DIP) which integrates a USB controller with a smaller surface-mounted board containing the MCU and antenna. The choice of the DIP format allows for easy prototyping on breadboards. The design was initially was based on the ESP-12 module of the ESP8266, which is a Wi-Fi SoC integrated with a Tensilica Xtensa LX106 core, widely used in IoT applications. This an open source IoT platform. It includes firmware which runs on the ESP8266 Wi-Fi SoC from Express if Systems, and hardware which is based on the ESP-12 module. The term "Node MCU" by default refers to the firmware rather than the development kits. The firmware uses the Luascripting language. It is based on the eLua project, and built on the Es press if Non-OS SDK for ESP8266. Node MCU was created shortly after the ESP8266 came out.

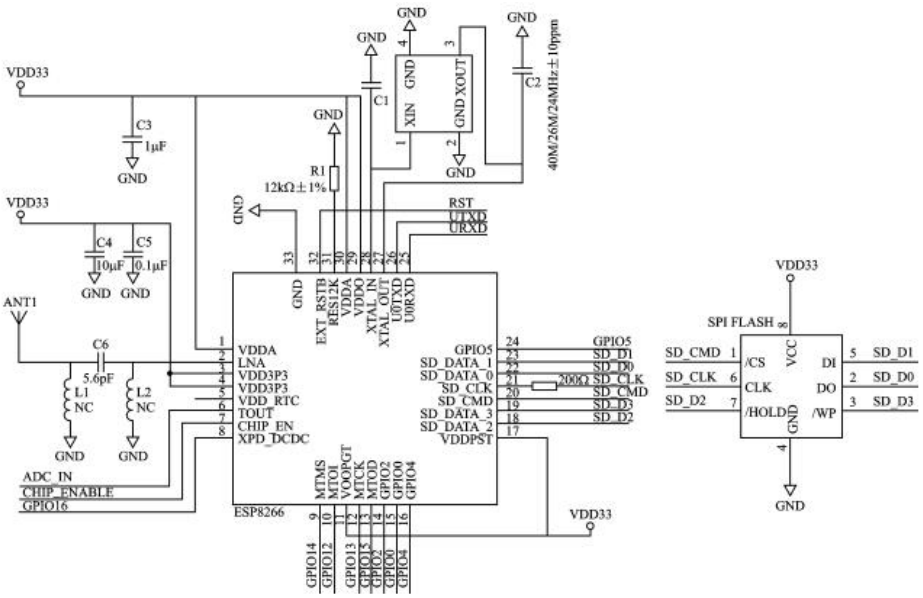


Figure 3.6: Node MCU Schematic Diagram

Another important update was made on 30 Jan 2015, when Devsaurus ported the u8glibto Node MCU project, enabling Node MCU to easily drive LCD, Screen, OLED, even VGA displays. In summer 2015 the creators abandoned the firmware project and a group of independent contributors took over. By summer 2016 the Node MCU included more than 40 different modules. Due to resource constraints users need to select the modules relevant for their project and build a firmware tailored to their needs.

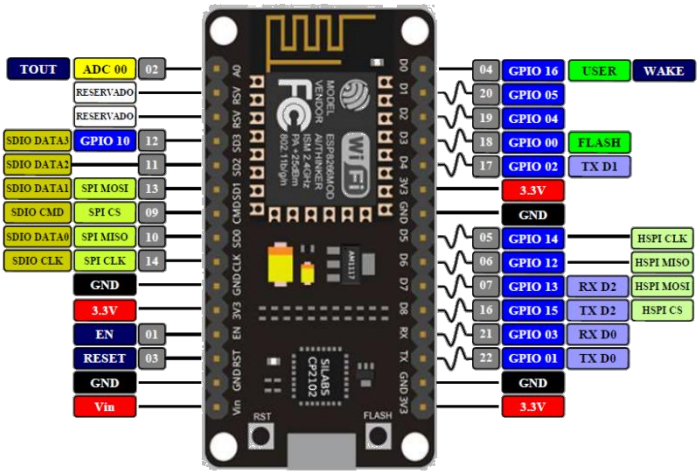


Figure 3.7: Node MCU Pin Out

Node MCU V3 ESP8266 ESP-12E is Wi-Fi development board that helps you to prototype your IoT product with few Lua script lines, or through Arduino IDE. The board is based on ESP8266 ESP-12E variant, unlike other ESP-12E, you won't need to buy a separate breakout board, usb to serial adapter, or even solder it to a PCB to get started, you will only need a usb cable (Micro USB).

Features

1. Communication interface voltage: 3.3V.
2. Antenna type: Built-in PCB antenna is available.
3. Wireless 802.11 b/g/n standard
4. WiFi at 2.4GHz, support WPA / WPA2 security mode
5. Support STA/AP/STA + AP three operating modes
6. Built-in TCP/IP protocol stack to support multiple TCP Client connections (5 MAX)
7. D0 ~ D8, SD1 ~ SD3: used as GPIO, PWM, IIC, etc., port driver capability 15mA
8. AD0: 1 channel ADC
9. Power input: 4.5V ~ 9V (10VMAX), USB-powered
10. Current: continuous transmission: ≈70mA (200mA MAX), Standby: <200uA
11. Transfer rate: 110-460800bps
12. Support UART / GPIO data communication interface
13. Remote firmware upgrade (OTA)

3.5 Buck Converter

A **buck converter** (**step-down converter**) is a DC-to-DC power converter which steps down voltage (while drawing less average current) from its input (supply) to its output (load). It is a class of switched-mode power supply (SMPS) typically containing at least two semiconductors (a diode and a transistor, although modern buck converters frequently replace the diode with a second transistor used for synchronous rectification) and at least one energy storage element, a capacitor, inductor, or the two in combination. To reduce voltage ripple, filters made of capacitors (sometimes in combination with inductors) are normally added to such a converter's output (load-side filter) and input (supply-side

filter). It is called a buck converter because the voltage across the inductor “bucks” or opposes the supply voltage.



Figure 3.8: DC -DC Buck Converter

DC-DC Buck Converter Step Down Module LM2596 Power Supply is a step-down(buck) switching regulator, capable of driving a 3-A load with excellent line and load regulation. These devices are available in fixed output voltages of 3.3 V, 5 V, 12 V, and an adjustable output version. The LM2596 series operates at a switching frequency of 150kHz, thus allowing smaller sized filter components than what would be required with lower frequency switching regulators.

Specifications of DC-DC Buck Converter Step Down Module LM2596 Power Supply :

- Conversion efficiency: 92%(highest)
- Switching frequency: 150KHz
- Output ripple: 30mA9maximum)
- Load Regulation: ± 0.5%
- Voltage Regulation: ± 0.5%
- Dynamic Response speed: 5% 200uS
- Input voltage:4.75-35V
- Output voltage:1.25-26V(Adjustable)
- Output current: Rated current is 2A, maximum 3A (Additional heat sink is required)
- Conversion Efficiency: Up to 92% (output voltage higher, the higher the efficiency)
- Switching Frequency: 150KHz
- Rectifier: Non-Synchronous Rectification

- Module Properties: Non-isolated step-down module (buck)
- Short Circuit Protection: Current limiting, since the recovery
- Operating Temperature: Industrial grade (-40 to +85) (output power 10W or less)

3.6 Gear Motor

Gear motors are mechanisms that adjust the speed of electric motors, leading them to operate at a certain speed. They are composed of a series of gears that make up a kinematic chain, working on a set of rotary parts. Their main purpose is to allow the reduction from an initial high speed to a lower one without negatively affecting the mechanism. In addition to this adjustment, a gear motor is in charge of adjusting the mechanical power of a system.



Figure 3.9: Gear Motor

Mechanical components that complement gear motors

Gear motors are composed of an electric motor and gears, which form the kinematic chain – the fundamental component of the gear ratio.

Kinematic chain

A motor's speed reducer is composed of a speed reducer and its gears. This speed reducer is basically a variable speed drive that allows for the speed to be reduced and increased at the output shaft.

Gears

Gears are toothed wheels made of metal or plastic (and new materials with each passing day) that transmit motion when meshing with each other. They are defined by their number of teeth and their size. In addition, they may have straight-cut or helical teeth.

Motors

The five types of motors that see the most use in gear motors are:

- Brushed motors, with brushes normally made out of carbon. They are bidirectional and may be used with DC or AC. They have a service life of about 3000 hours.
- Asynchronous motors, which are brush less single-direction motors. They are highly limited.
- Synchronous brush less motors, which may be single-direction or bidirectional. They have a constant speed if the frequency of the power source is stable.
- Brush less DC motors that use a driver and can attain high speeds.
- Stepper DC brush less motors. They can be positioned with an average precision of 7.5º.

3.7 Motor Driver

The **L293D** is a popular 16-Pin **Motor Driver** IC. As the name suggests it is mainly used to drive **motors**. A single **L293D** IC is capable of running two DC **motors** at the same time; also the direction of these two **motors** can be controlled independently.

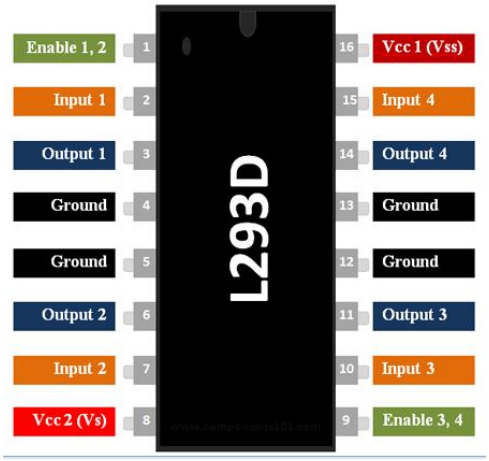


Figure 3.10: Motor driver IC L293D

Working Process:

L293D IC is a typical **Motor Driver** IC which allows the DC **motor** to drive on any direction. This IC consists of 16-pins which are used to control a set of two DC **motors** instantaneously in any direction. It means, by using a **L293D** IC we can control two DC **motors**.

Features

- Can be used to run Two DC motors with the same IC.
- Speed and Direction control is possible
- Motor voltage Vcc2 (Vs): 4.5V to 36V
- Maximum Peak motor current: 1.2A
- Maximum Continuous Motor Current: 600mA
- Supply Voltage to Vcc1(vss): 4.5V to 7V
- Transition time: 300ns (at 5Vand 24V)
- Automatic Thermal shutdown is available

L293D Pin Configuration

Pin Number	Pin Name	Description
1	Enable 1,2	This pin enables the input pin Input 1(2) and Input 2(7)
2	Input 1	Directly controls the Output 1 pin. Controlled by digital circuits
3	Output 1	Connected to one end of Motor 1
4	Ground	Ground pins are connected to ground of circuit (0V)
5	Ground	Ground pins are connected to ground of circuit (0V)
6	Output 2	Connected to another end of Motor 1
7	Input 2	Directly controls the Output 2 pin. Controlled by digital circuits
8	Vcc2 (Vs)	Connected to Voltage pin for running motors (4.5V to 36V)
9	Enable 3,4	This pin enables the input pin Input 3(10) and Input 4(15)
10	Input 3	Directly controls the Output 3 pin. Controlled by digital circuits
11	Output 3	Connected to one end of Motor 2
12	Ground	Ground pins are connected to ground of circuit (0V)

13	Ground	Ground pins are connected to ground of circuit (0V)
14	Output 4	Connected to another end of Motor 2
15	Input 4	Directly controls the Output 4 pin. Controlled by digital circuits
16	Vcc2 (Vss)	Connected to +5V to enable IC function

Table 03: Motor Driver Pinout

Use of a L293D Motor Driver IC:

Using this L293D motor driver IC is very simple. The IC works on the principle of **Half H-Bridge**, let us not go too deep into what H-Bridge means, but for now just know that H bridge is a set up which is used to run motors both in clock wise and anti clockwise direction. As said earlier this IC is capable of running two motors at the any direction at the same time, the circuit to achieve the same is shown below.

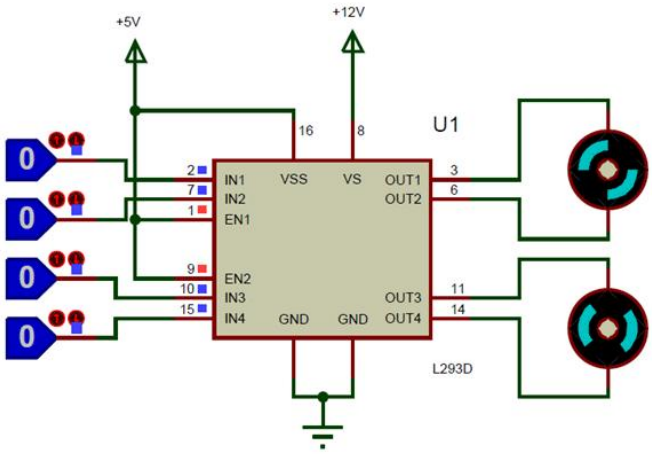


Figure 3.11: L293D circuit Diagram

All the Ground pins should be grounded. There are two power pins for this IC, one is the Vss(Vcc1) which provides the voltage for the IC to work, this must be connected to +5V. The other is Vs(Vcc2) which provides voltage for the motors to run, based on the specification of your motor you can connect this pin to anywhere between 4.5V to 36V, here I have connected to +12V.

The Enable pins (Enable 1,2 and Enable 3,4) are used to Enable Input pins for Motor 1 and Motor 2 respectively. Since in most cases we will be using both the motors both the pins are held high by default by connecting to +5V supply. The input pins Input 1,2 are used to control the motor 1 and Input pins 3,4 are used to control the Motor 2. The input pins are connected to the any Digital circuit or micro-controller to control the speed and direction of the motor.

Applications

- Used to drive high current Motors using Digital Circuits
- Can be used to drive Stepper motors
- High current LED's can be driven
- Relay Driver module (Latching Relay is possible)

3.8 Resistor

A resistor is a passive two-terminal electrical component that implements electrical resistance as a circuit element. Resistors act to reduce current flow, and, at the sometime, act to lower voltage levels within circuits. Resistors may have fixed resistances or variable resistances, such as those founding thermostats, visitors, trimmers, photo resistors, hamsters and potentiometers. The current through a resistor is in direct proportion to the voltage across the resistor's terminals. This relationship is represented by Ohm's law.



Figure 3.12: Resistor

Theory of operation

The behavior of an ideal resistor is dictated by the relationship specified by Ohm 'slaw:

$$V = I.R$$

Ohm's law states that the voltage (V) across a resistor is proportional to the current (I), where the constant of proportionality is the resistance (R).

Equivalently, Ohm's law can be stated:

$$I = V/R$$

This formulation states that the current (I) is proportional to the voltage (V) and inversely proportional to the resistance (R). This is directly used in practical computations. For example, if a 300-ohm resistor is attached across the terminals of a12 volt battery, then a current of12 / 300 = 0.04 amperes flows through that resistor.

3.9 GPS Module

The NEO-6M GPS module is a well-performing complete GPS receiver with a built-in 25 x 25 x 4mm ceramic antenna, which provides a strong satellite search capability. With the power and signal indicators, you can monitor the status of the module. Thanks to the data backup battery, the module can save the data when the main power is shut down accidentally. Its 3mm mounting holes can ensure easy assembly on your aircraft, which thus can fly steadily at a fixed position, return to Home automatically, and automatic waypoint flying, etc. Or you can apply it on your smart robot car for automatic returning or heading to a certain destination, making it a real "smart" bot! The schematic diagram of the module is shown as below:

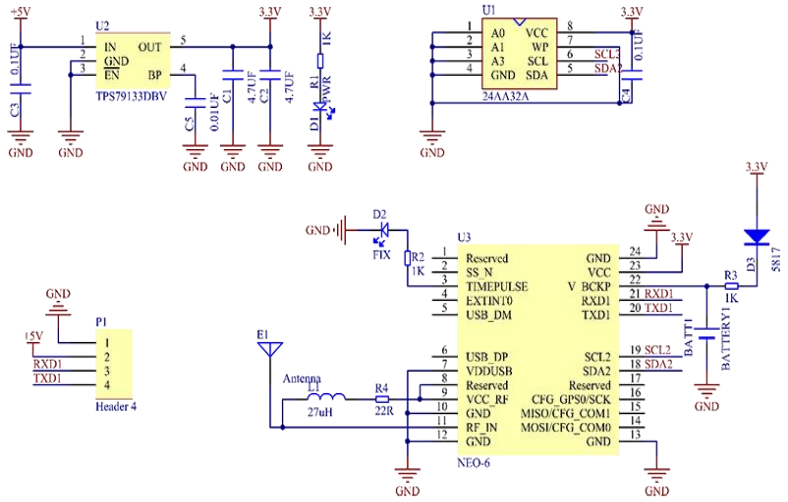


Figure 3.13: Schematic Diagram Of GPS

This is a complete GPS module that is based on the Ublox NEO 6M GPS. This unit uses the latest technology from Ublox to give the best possible positioning information and includes a larger built-in 25 x 25mm active GPS antenna with a UART TTL socket. A battery is also included so that you can obtain a GPS lock faster. This is an updated GPS module that can be used with ardupilot mega v2. This GPS module gives the best possible

position information, allowing for better performance with your Ardupilot or other Multi rotor control platform.

Features:

1. 5Hz position update rate
2. Operating temperature range: -40 TO 85°CUART TTL socket
3. EEPROM to save configuration settings
4. Rechargeable battery for Backup
5. The cold start time of 38 s and Hot start time of 1 s
6. Supply voltage: 3.3 V
7. Configurable from 4800 Baud to 115200 Baud rates. (default 9600)
8. Super Sense ® Indoor GPS: -162 dBm tracking sensitivity
9. Support SBAS (WAAS, EGNOS, MSAS, GAGAN)
10. Separated 18X18mm GPS antenna

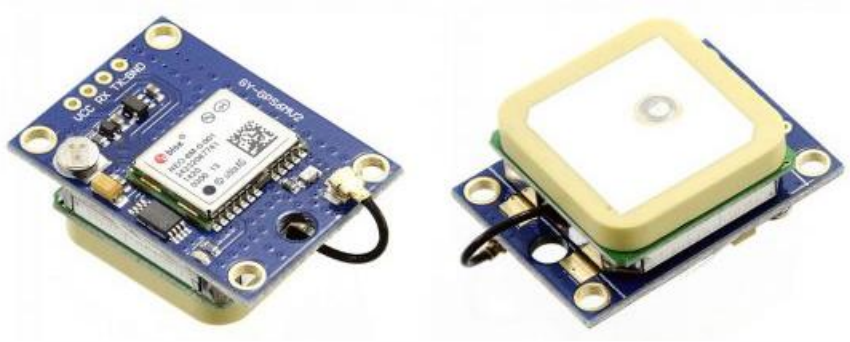


Figure 3.14: NEO-6m GPS

3.10 GSM Module

At the heart of the module is a SIM800L GSM cellular chip from Sim Com. The operating voltage of the chip is from **3.4V to 4.4V**, which makes it an ideal candidate for direct LiPo battery supply. This makes it a good choice for embedding into projects without a lot of space.



Figure 3.15: GSM Module

All the necessary data pins of SIM800L GSM chip are broken out to a 0.1″ pitch headers. This includes pins required for communication with a microcontroller over **UART**. The module supports baud rate from **1200bps** to **115200bps** with Auto-Baud detection.

The module needs an external antenna to connect to a network. The module usually comes with a **Helical Antenna** and solders directly to NET pin on PCB. The board also has a U.FL connector facility in case you want to keep the antenna away from the board.

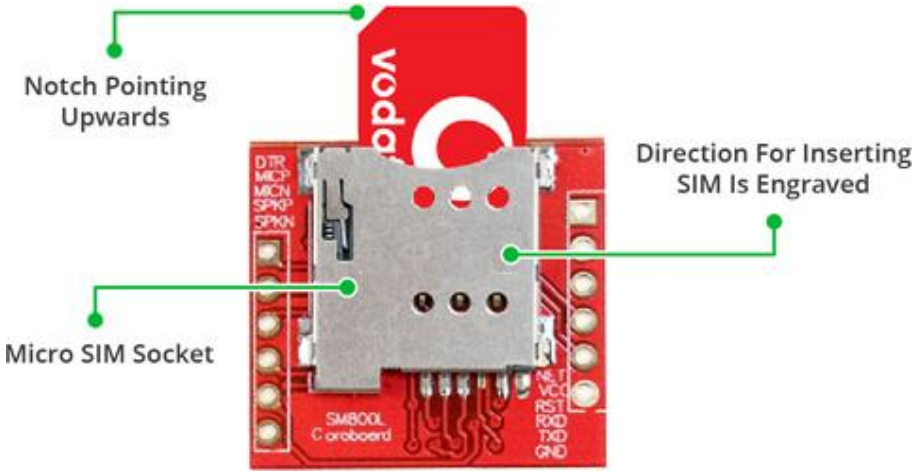


Figure 3.16: GSM Module Introducing

There's a SIM socket on the back! Any activated, **2G micro SIM card** would work perfectly. Correct direction for inserting SIM card is normally engraved on the surface of the SIM socket. This module measures only 1 inch² but packs a surprising amount of features into its little frame. Some of them are listed below:

- Supports Quad-band: GSM850, EGSM900, DCS1800 and PCS1900
- Connect onto any global GSM network with any 2G SIM

- Make and receive voice calls using an external 8Ω speaker & electret microphone
- Send and receive SMS messages
- Send and receive GPRS data (TCP/IP, HTTP, etc.)
- Scan and receive FM radio broadcasts
- Transmit Power:
    - Class 4 (2W) for GSM850
    - Class 1 (1W) for DCS1800
- Serial-based AT Command Set
- FL connectors for cell antennae
- Accepts Micro SIM Card

SIM800L GSM Module Pinout

The SIM800L module has total 12 pins that interface it to the outside world. The connections are as follows:



Figure 3.17: GSM Module Pinout

NET is a pin where you can solder Helical Antenna provided along with the module.

VCC supplies power for the module. This can be anywhere from 3.4V to 4.4 volts. Remember connecting it to 5V pin will likely destroy your module! It doesn't even run on 3.3 V! An external power source like Li-Po battery or DC-DC buck converters rated 3.7V 2A would work.

RST (Reset) is a hard reset pin. If you absolutely got the module in a bad space, pull this pin low for 100ms to perform a hard reset.

RxD (Receiver) pin is used for serial communication.

TxD (Transmitter) pin is used for serial communication.

GND is the Ground Pin and needs to be connected to GND pin on the Arduino.
RING pin acts as a Ring Indicator. It is basically the 'interrupt' out pin from the module. It is by default high and will pulse low for 120ms when a call is received. It can also be configured to pulse when an SMS is received.
DTR pin activates/deactivates sleep mode. Pulling it HIGH will put module in sleep mode, disabling serial communication. Pulling it LOW will wake the module up.
MIC± is a differential microphone input. The two microphone pins can be connected directly to these pins.
SPK± is a differential speaker interface. The two pins of a speaker can be tied directly to these two pins.

3.11 Wheel

65mm Robot Wheel for BO Motors is a circular component that is intended to rotate on an axial bearing. The wheel is one of the main components of the wheel and axle which is one of the six simple machines. Wheels, in conjunction with axles, allow heavy objects to be moved easily facilitating movement or transportation while supporting a load, or performing labor in machines.

Features:

- Car hub is reinforced nylon, very sturdy.
- The tire is made of a soft rubber for non-slip effect.
- Good load capacity and greater ability to grip
- Robot Wheel Material: Plastic+Rubber
- New design wheel for better combination with the motor

Specifications:

Loading Capacity	2.5 Kg
D hole For BO Motor	6 mm
Wheel Diameter	65 mm
Color	Black (Tyre), Yellow (Rim)
Wheel Width	28 mm

Body Material	Plastic
Grip Material	Rubber
Center Shaft Hole Diameter	5.4mm-(round side) and 3.5mm-(flat side)
Weight	34 gm

3.12 Arduino Software

An intelligent micro-controller unit called the Arduino Uno can be programmed with the Arduino software. There is no need to install any software other than the Arduino first. From the Tools Board menu (according to your on-board micro-controller) select "Arduino Unio A" using the hardware programmer.

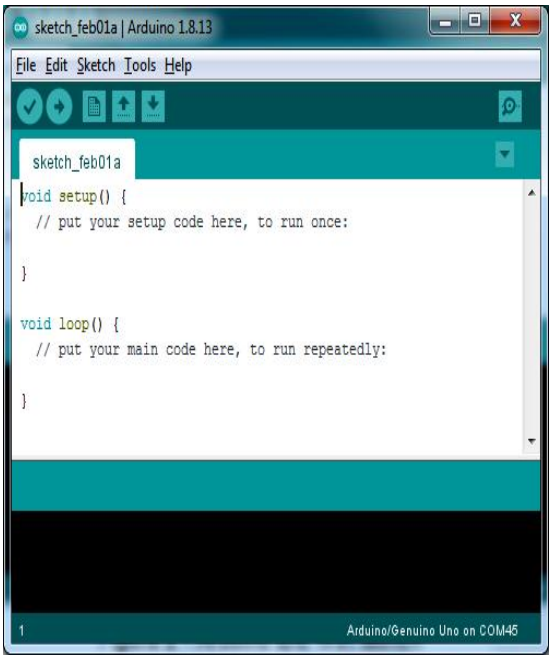


Figure 3.18: Arduino Software Interface IDE

The communication uses the native STK 500 protocol (reference C header file). We can bypass the boot loader and program the micro-controller under the ICSP (In Circuit Serial Programming) heading. Atmeg16U2 firmware source code (or 8U2 on The Rev1 and Rev2) Atmega 16U2/8U2 boards are loaded with a DFU boot loader, which can be enabled at: On the Rev 1: board, attach a solder jumper to the back of the board. Rev 2 or later: There is a resistor that pulls the 8U2/16U2 HWB wires to ground, making it easy to

put in DFU mode. The Arduino Uno is the latest smart micro-controller unit and has Several advantages for communicating with other Arduino computers or other micro-controllers.

The ATMA 328 provides URT TTL with Communication Protocol (5V), available in digital PIN 0 - (RX) for data retrieved with PIN No. 1 (TX) for data transmission. The system port communicates via USB Atimega 16U2 in the port and comes out as a low-end port with computer software. '16U2 firmware uses standard USB CMM driver and no external drivers required. Anyway, Windows needs an .inf file.The Arduino file system contains a serial viewer that allows sending easy text data to and from Arduino boards. The RX and TX connectors will flash on the board when transferring data via USB-to-serial connector and USB connection to the computer (but not for serial communication on pins 0 and 1).

The software system library allows seamless communication within any digital PIN. The Atmeg 328 supports I2C information communication (TWI) and SPI. The Arduino software includes a wire library to support the use of the I2C car. Arduino programs are written in C or C ++ and the program code written for Arduino is called Sketch. Arduino IDE uses GNU and AVR LBC toolkit to compile programs, as well as upload programs using Android. For example, software for Arduino can be developed using the Arduino platform using Atmel micro-controllers, around the development of Atmel, AVR Studio or even the new Atmel Studio.

3.13 Easy EDA Software

Easy EDA is a web-based EDA tool suite that enables hardware engineers to design, simulate, share - publicly and privately - and discuss schematics, simulations and printed circuit boards. Other features include the creation of a bill of materials, Gerber files and pick and place files and documentary outputs in PDF, PNG and SVG formats. Easy EDA allows the creation and editing of schematic diagrams, SPICE simulation of mixed analogue and digital circuits and the creation and editing of printed circuit board layouts and, optionally, the manufacture of printed circuit boards.

Subscription-free membership is offered for public plus a limited number of private projects. The number of private projects can be increased by contributing high quality

public projects, schematic symbols, and PCB footprints and/or by paying a monthly subscription.Registered users can download Gerber files from the tool free of charge; but for a fee, Easy EDA offers a PCB fabrication service. This service is also able to accept Gerber file inputs from third party tools.

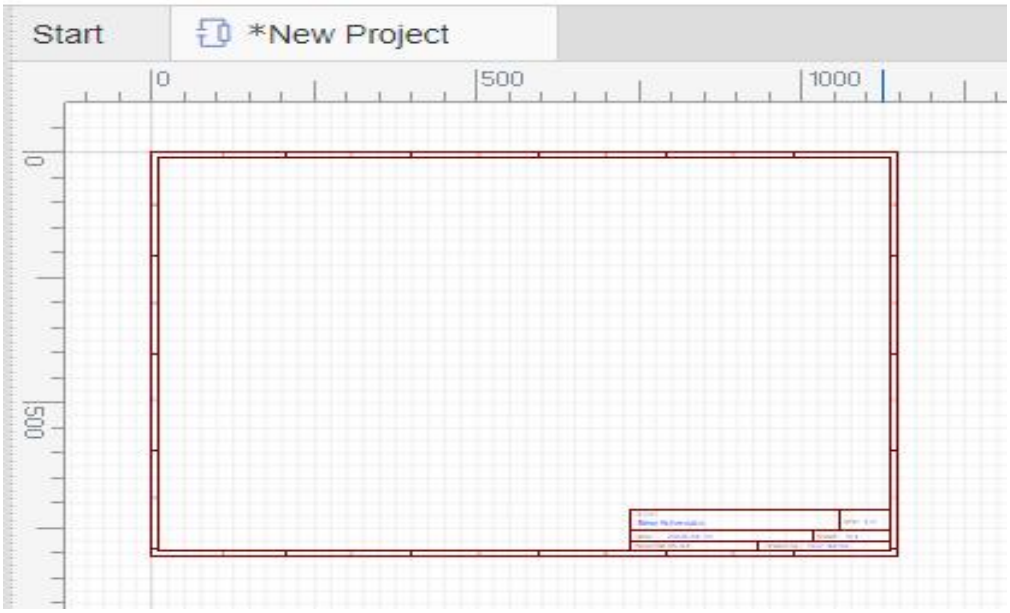


Figure 3.19: Easy EDA Software Interface

3.14 Remote XY

Remote XY is easy way to make and use a mobile graphical user interface for controller boards to control via smartphone or tablet. The system includes:

- Editor of mobile graphical interfaces for controller boards, located on the site remotexy.com
- Mobile app Remote XY that allows to connect to the controller and control it via graphical interface. Download app.

Distinctive features:

- The interface structure is stored in the controller. When connected, there is no interaction with servers to download the interface. The interface structure is downloaded to the mobile application from the controller.

- One mobile application can manage all your devices. The number of devices is not limited.

Supported connection methods:

- Internet over Cloud Server;
- Wi-Fi client and access point;
- Bluetooth;
- Ethernet by IP or URL;
- USB OTG (Android only that support USB OTG);

Supported boards:

- Arduino UNO, MEGA, Leonardo, Pro Mini, Nano, MICRO and compatible AVR boards;
- ESP8266 boards;
- ESP32 boards;
- Chip KIT UNO32, Chip KIT uC32, Chip KIT Max32;

Supported communication modules:

- Bluetooth HC-05, HC-06 or compatible;
- Bluetooth BLE HM-10 or compatible;
- ESP8266 as modem;
- Ethernet W5100;

Supported IDE:

- Arduino IDE;
- FLProg IDE;
- MPIDE;

Supported mobile OS:

- Android;
- iOS;

Remote XY is easy way to make a unique graphical interface to control micro-controller device via mobile application, Arduino for example.

Remote XY allows:

- To develop any graphical management interface, using the control, display and decoration elements any combination thereof. You can develop the graphical interface for any task, placing the elements on the screen using the online editor. Online editor posted on the website remotexy.com.
- After the development of the graphical interface, you get the source code for the micro-controller that implements your interface. The source code provides a structure for interaction between your program with the controls and display. Thus you can easily integrate the control system into your task for which you are developing the device.
- To manage micro-controller device using your smartphone or tablet with the graphical interface. For manage used mobile application Remote XY.
- Using one mobile application, you can manage a large number of devices with different graphical management interfaces. As the interface description is stored on board the micro-controller device.

The source code for the micro-controller, for example a sketch for the Arduino, you opened it in the IDE. In the simplest case, you can download the sketch in Arduino board and test your interface immediately, connecting via a smartphone. But often you need to integrate the developed graphical interface with your problem you want to solve it using the Arduino. Next, we will show how it is easy to do.

The generated source code have a part that is starts with tag **Remote XY include library** and ends with tag **END Remote XY include**. This part of the code defines the structure of the graphical interface that you have created. In turn, this structure is also divided into parts. Consider the order.

Connection settings

This part of the code in more detail defines the characteristics of the selected connection. For example, if you are using a connection via Software Serial, you can determine which pins micro-controller will be connected to the Bluetooth module. You can also determine

the rate of exchange with the Bluetooth module, if it is different from the default 9600 baud. All definitions are set by default, but you can change them as you need. This is the only section of code in the included section of Remote XY that you can change.

CHAPTER 4

METHODOLOGY

4.1 Our methodologies for the project

Our methodologies for the project:

- Creating an idea for **GSM Based Accident Alerting System**. And designing a block diagram & circuit diagram to know which components we need to construct it.
- Collecting all the components of our system.
- Setting up all the components in a PCB board & then soldering. Lastly, assembling all the blocks in a board and to run the system & for checking purposes.

4.2 Working Step Chart

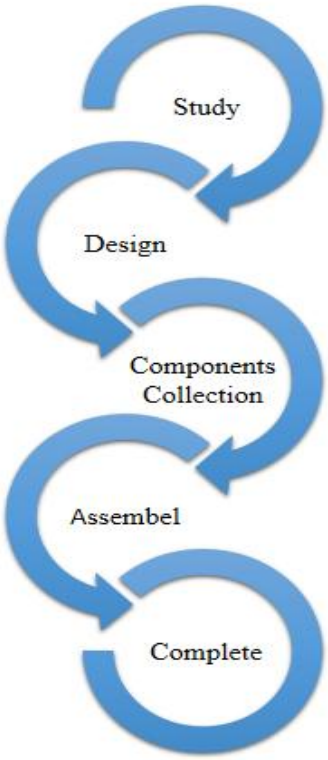


Figure 4.1: Working Step Chart

4.3 Block Diagram

The way of whole project works is that we take DC power from the supply voltage and then feed it to a Buck Converter. The Buck Converter simply converts the 5V DC. We will use this 5V DC output from the Buck Converter to run our controller, motor and other units.

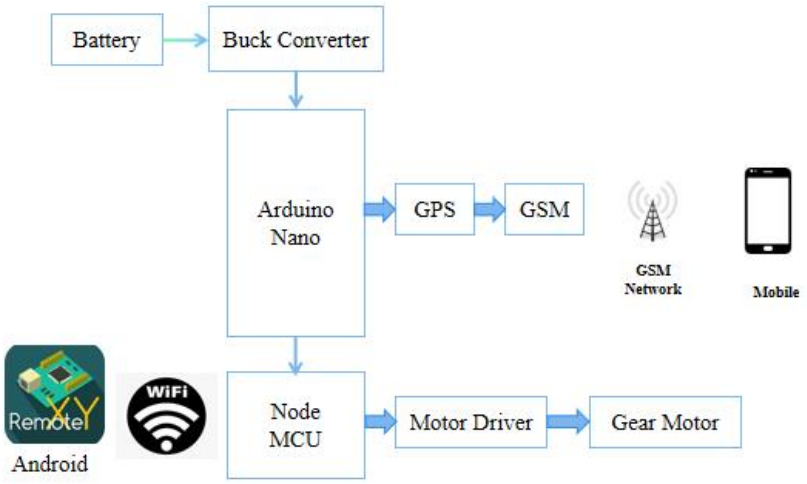


Figure 4.2: Block Diagram of Our System

4.4 Schematic Diagram

The schematic diagram here is representing the electrical circuit and the components of the project. Here we have used standardized symbols and lines.

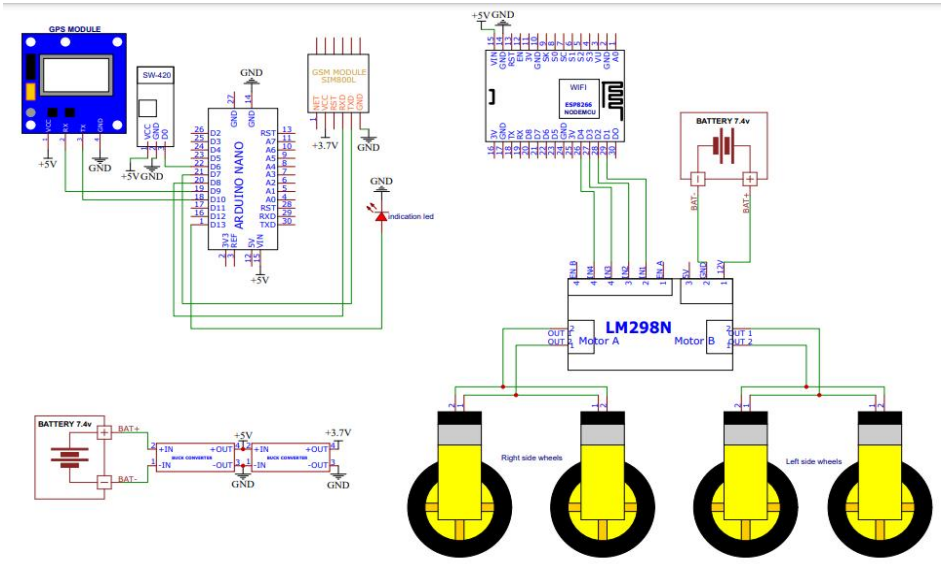


Figure 4.3: Schematic Diagram of the Project

4.5 Working Principle

In our project we have a power source that is battery. In this project we have used an Node MCU for operate this system. Here we use GPS sensor who sense the vehicle location, GSM module will help the send a text in rescue team. Once the system is activated then if the car has an accident somewhere, it will immediately track the location of the place via GPS and send a text (Location Link) to the phone via GSM. Android phone will be control the car. This will save the lives of the passengers in the car by rescuing them very quickly. This is the main purpose of our system.

4.6 Our Final System View

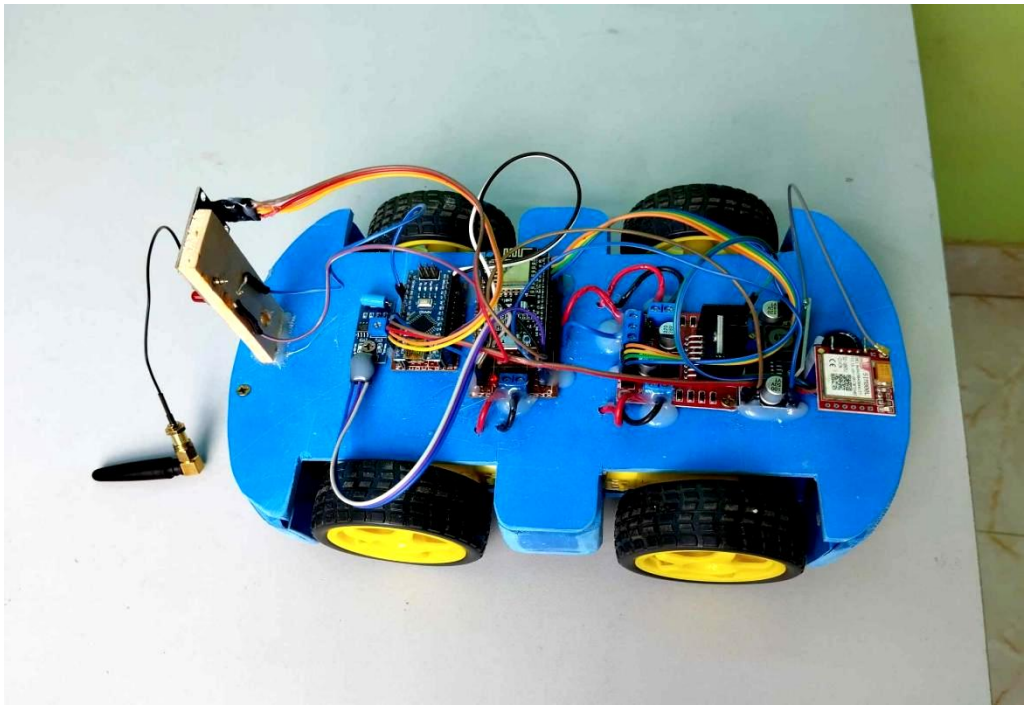


Figure 4.4: Our Final System Overview

CHAPTER 5
RESULT AND DISCUSSION

5.1 Result

This chapter contains the results obtained and discussion about the full project. Our project is smoke and gas detection system. In our project making we used PVC boards for total hardware making. After finally completing this project, we ran it & we observed the output of this project. We can see that it is working well as expected. After making our project we observe it very careful. It works as we desire. Our project give output perfectly and all equipment are work perfectly. We check how much it works and we get perfect output from this project.

- Finally, we have completed our project successfully & check our project its run accurately according to our objective.
- At first, we start our system.
- If the car has an accident, it will track the location of the car via GPS and send a text to the rescue team's phone.
- The GSM module will help send this text.
- This will make it easier to get an accident vehicle out, and rescue teams will be able to get there quickly.

5.2 Discussion

While working on our project, we did face some difficulties as it is a very complex system but the end results, we came up with were quite satisfactory. We have put the whole system through several tasks to validate our work and also have taken necessary notes for future improvements. Some future recommendations that we have involves improvement in system design and wiring, adding features for more efficient.

5.3 Advantage

There are many advantages of our project because of its accuracy. Some of the advantages are pointed out below:

- Accident location detected automatically.

- Automatic text sends to rescue team.
- Reduce the searching time to find accident car.
- This project is easy to use.
- Cost effective.
- User-friendly.
- Automatic and requires less human interaction.
- Can be operated virtually from anywhere in the world.
- The whole system consumes very little energy.

5.4 Application

Some of the application areas of the project have been pointed out below:

- It can be used in any kind of vehicle.
- Vehicle Safety and Accident Detection
- Location Tracking and Recovery
- Emergency Response System
- Smart Transportation Systems
- Driver Assistance and Safety Features

5.5 Limitations

Though this project has many advantages but it has some limitations as well and they are listed below:

- Here we used Chinese products, for the prototype so there's some processing delay present in the circuit.
- In GSM Module sim must be have some money to send the text.
- This project can now be only used for small scale purposes.
- Battery will discharge quickly.

CHAPTER 6
CONCLUSION

6.1 Conclusion

The completed module is successfully detecting accidents using this module is connected to GSM and GPS to send the data via message or notification. This module is successfully transmitting the location coordinates of accident spots to mobile via the GSM Module. This system will assure the accident spot. The ambulance will get the direction through map. So that, the ambulance reaches the accident place without facing any trouble and the victim will reach the hospital on timely. This system will show an alternative road without that accident road. So, other user can reach their destination easily and save their time. Hence, it will reduce life risk of the victim. After all, we may hope for a better solution of smart accident detection and rescue system.

6.2 Future Scope

We are thinking about adding many features to our project in the future to get more desirable outcomes. Some of the steps that we are thinking about taking are given below:

- In future development this project can be develop by more sensor & alarm system.
- In future we will add a huge battery source.

REFERENCE

[1]. Hari Sankar S, Jayadev K, Suraj B and Aparna P, "A Comprehensive Solution to Road Traffic Collision with Vehicle Recognition Hospital wagon Management", *International Conference on Advances in Electrical, Electronic and System Engineering*, November 2016.

[2]. Nicky Kattuk karan, Arun George, Mithun Haridas T.P, "Intelligent Collision with vehicle Recognition and Warning System for Urgent Situation Medical Assistance", *International Conference on Computer Communication and Informatics (ICCCI), India*, January 2017.

[3]. Adnan Bin Faiz, Ahmed Imteaj, Mahfuzul hoq Chowdhury, "Smart Vehicle Collision with vehicle Recognition and Alarming System Using a Smartphone", *International Conference on Computer & Information Engineering, Bangladesh*, November2015.

[4]. D.Selvathi, P.Pavithra, T.Preethi, "Intelligent Transportation System for Collision with Vehicle Prevention and Recognition", *International Conference on Intelligent Computing and Control Systems(ICICCS),* 2017.

[5]. Kiran Sawant, Imran Bhole, Prashant Kokane, PirajiDoiphode, Prof. Yogesh Thorat, "Collision with vehicle Warning and Vehicle Tracking System", *International Journal of Innovative Research in Computer and Communication Engineering*, Volume 4, Issue 5, May 2016.

[6]. Manasi Patil, Aanchal Rawat, Prateek Singh, Srishtie Dixit, "Collision with vehicle Recognition and Hospital wagon Control using Intelligent Traffic Control System", *International Journal of Engineering Trends and Technology (IJETT)*, Volume 34, Number 8, April 2016.

[7]. V.Sagar Reddy, L.Padma Sree, V. Naveen Kumar, "Design and Development of accelerometer based System for driver safety", *International Journal of Science,*

Engineering and Technology Research (IJSETR), Volume 3, Issue 12, December 2014.

[8]. Sri Krishna Chaitanya Varma, Poornesh, Tarun Varma, Harsha, "Automatic Vehicle Collision with vehicle Recognition and Messaging System Using GPS and GSM Modems", *International Journal of Scientific & Engineering Research*, Volume 4, Issue 8, August 2013.

[9]. Apurva Mane, Jaideep Rana, "Vehicle Collision recognition and Remote Alarm Device using Arduino", *International Journal of Current Engineering and Technology*, Volume 4, Number 3, June 2014.

APPENDIX

Programming Code:

Car Controller:

```
#define REMOTEXY_MODE__ESP8266WIFI_LIB_CLOUD
#include <ESP8266WiFi.h>

#include <RemoteXY.h>

// RemoteXY connection settings
#define REMOTEXY_WIFI_SSID "abcde"
#define REMOTEXY_WIFI_PASSWORD "123456789"
#define REMOTEXY_CLOUD_SERVER "cloud.remotexy.com"
#define REMOTEXY_CLOUD_PORT 6376
#define REMOTEXY_CLOUD_TOKEN "24d85257a4381c95d4899dcb6d14069c"


// RemoteXY configurate
#pragma pack(push, 1)
uint8_t RemoteXY_CONF[] =   // 109 bytes
  { 255,4,0,0,0,102,0,16,24,2,10,48,43,4,15,15,25,8,15,15,
  4,26,31,70,111,114,119,97,114,100,0,31,79,70,70,0,10,48,5,24,
  15,15,5,40,15,15,4,26,31,76,101,102,116,0,31,79,70,70,0,10,
  48,81,24,15,15,44,40,15,15,4,26,31,82,105,103,104,116,0,31,79,
  70,70,0,10,52,44,43,15,15,25,70,15,15,4,26,31,82,101,118,101,
  114,115,101,0,31,79,70,70,0 };

// this structure defines all the variables and events of your control interface
struct {

    // input variables
  uint8_t pushSwitch_1; // =1 if state is ON, else =0
  uint8_t pushSwitch_2; // =1 if state is ON, else =0
  uint8_t pushSwitch_3; // =1 if state is ON, else =0
  uint8_t pushSwitch_4; // =1 if state is ON, else =0

    // other variable
  uint8_t connect_flag;  // =1 if wire connected, else =0

} RemoteXY;
#pragma pack(pop)

/////////////////////////////////////////////
//          END RemoteXY include          //
/////////////////////////////////////////////
int in1 = D1;
int in2 = D2;
```

```
int in3 = D3;
int in4 = D4;

void setup()
{
 RemoteXY_Init ();
 Serial.begin(9600);
 pinMode(in1, OUTPUT);
 pinMode(in2, OUTPUT);
 pinMode(in3, OUTPUT);
 pinMode(in4, OUTPUT);
}

void loop()
{
 RemoteXY_Handler();

 // Check the state of buttons and control the car accordingly
 if (RemoteXY.pushSwitch_1 == 1)
 {
  // Motor control logic for forward motion
  analogWrite(in1, 140);
  analogWrite(in2, 0);
  analogWrite(in3, 180);
  analogWrite(in4, 0);
 }
 else if (RemoteXY.pushSwitch_4 == 1)
 {
  // Motor control logic for reverse motion
  analogWrite(in1, 0);
  analogWrite(in2, 140);
  analogWrite(in3, 0);
  analogWrite(in4, 180);
 }
 else if (RemoteXY.pushSwitch_3 == 1)
 {
  // Motor control logic for left turn
  analogWrite(in1, 255);
  analogWrite(in2, 0);
  analogWrite(in3, 0);
  analogWrite(in4, 255);
 }
 else if (RemoteXY.pushSwitch_2 == 1)
 {
  // Motor control logic for right turn
  analogWrite(in1, 0);
  analogWrite(in2, 255);
  analogWrite(in3, 255);
  analogWrite(in4, 0);
 }
```

```
  else
  {
    // Stop the motors if no button is pressed
    analogWrite(in1, 0);
    analogWrite(in2, 0);
    analogWrite(in3, 0);
    analogWrite(in4, 0);
  }

  // Add a delay if necessary to prevent flooding the RemoteXY server
  RemoteXY_delay(50);
}
```

GSM Based:

```
#include <SoftwareSerial.h>
#include <Wire.h>
#include <TinyGPS++.h>

SoftwareSerial SIM900(7, 8);
SoftwareSerial gpsSerial(9, 10);
String textMessage;
char msg;
char phone_no[] = "+01866745431"; // Include the country code and remove leading zero
01770888106
String message;
char str[15];
int vib = 6;
int led = 13;
float latitude = 0.0;
float longitude = 0.0;
bool getGPS(TinyGPSPlus &gps);

TinyGPSPlus gps;

void setup() {


  Serial.begin(9600);
  pinMode(vib, INPUT);
  pinMode(led, OUTPUT);
  digitalWrite(led, LOW);
  SIM900.begin(9600);
  delay(200);
  SIM900.print("AT+CMGF=1\r");
  delay(100);
  SIM900.print("AT+CNMI=2,2,0,0,0\r");
  delay(100);
  SIM900.println("AT+CNMI=2,2,0,0,0");// AT Command to receive a live SMS
```

```
  Serial.println("AT+CNMI=2,2,0,0,0");
  Serial.print("SIM900 ready...");
  gpsSerial.begin(9600);

}


void loop() {
  int vibration = digitalRead(vib);
  Serial.print(vibration);


  if (vibration == 1) {
    digitalWrite(led, HIGH);
    SendMessage();
    delay(5000);
    digitalWrite(led, LOW);

  }
  while (1)
  {
    while (gpsSerial.available() > 0)
    {
      gps.encode(gpsSerial.read());
    }

    if (gps.location.isUpdated())
    {
      Serial.print("LAT=");  Serial.println(gps.location.lat(), 6);
      Serial.print("LONG="); Serial.println(gps.location.lng(), 6);
      latitude = gps.location.lat();
      longitude = gps.location.lng();
      break;
    }
  }
  delay(500);
}

void SendMessage()
{
  SIM900.print("AT+CMGF=1\r");
  delay(20);
  SIM900.print("AT+CMGS=\"");
  SIM900.print(phone_no);
  SIM900.println("\"");
  delay(20);
  SIM900.print("Accident detected!!!!    ");
  delay(20);
  SIM900.print("http://maps.google.com/maps?q=loc:");
  SIM900.print(latitude, 6);
```

```
  SIM900.print(",");
  SIM900.print(longitude, 6);
  delay(20);
  SIM900.println((char)26);
  delay(20);
  SIM900.println();
  delay(3000);
}
```