Visvesvaraya Technological University, Belagavi.

PROJECT REPORT

on

Automatic speed control of vehicle and Air pollution detection using IOT

Project Report submitted in partial fulfillment of the requirement for the award of the degree of

Bachelor of Engineering

in

Electronics and Communication Engineering

For the academic year 2019-20

Submitted by

USN	Name
1CR14EC048	Hyeonjung Choi
1CR15EC011	Akashdeep Karmakar
1CR16EC426	Rashmi.V
1CR14EC009	Abhishek Chauhan

Under the guidance of

Prof. Mahesh.S.Gour
Assistant Professor
Department of ECE
CMRIT, Bangalore

Department of Electronics and Communication Engineering
CMR Institute of Technology, Bengaluru – 560 037

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

CELEBRATING 25 YEARS
CMR INSTITUTE OF TECHNOLOGY

CERTIFICATE

This is to Certify that the dissertation work **Automatic speed control of vehicle and Air pollution detection using IOT** carried out by Hyeonjung Choi (1CR14EC048), Akashdeep Karmakar (1CR15EC011), Rashimi.V (1CR16EC426) and Abishek Chauhan (1CR14EC009) bonafide students of **CMRIT** in partial fulfillment for the award of **Bachelor of Engineering** in **Electronics and Communication Engineering** of the **Visvesvaraya Technological University, Belagavi,** during the academic year **2019-20**. It is certified that all corrections/suggestions indicated for internal assessment have been incorporated in the report deposited in the departmental library. The project report has been approved as it satisfies the academic requirements in respect of Project work prescribed for the said degree.

Signature of Guide	Signature of HOD	Signature of Principal
________________	________________	________________
Prof. Mahesh.S.Gour	Dr.R. Elumalai	Dr. Sanjay Jain
Assistant Professor,	Head of the Department,	Principal,
Department of ECE,	Department of ECE,	CMRIT,
CMRIT, Bangalore.	CMRIT, Bangalore.	Bangalore.

External Viva

Name of Examiners	Signature & date
1.	
2	

ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be incomplete without the mention of people who made it possible, whose consistent guidance and encouragement crowned our efforts with success.

We consider it as our privilege to express the gratitude to all those who guided in the completion of the project.

We express our gratitude to Principal, **Dr. Sanjay Jain,** for having provided us the golden opportunity to undertake this project work in their esteemed organization.

We sincerely thank **Dr. R. Elumalai,** Head**,** Department of Electronics and Communication Engineering, CMR Institute of Technology for the immense support given to me.

We express our gratitude to our project **Prof. Mahesh.S.Gour,** Assistant Professor, for the support, guidance and suggestions throughout the project work.

Last but not the least, heartful thanks to our parents and friends for their support.

Above all, We thank the Lord Almighty for His grace on us to succeed in this endeavor.

ABSTRACT

The Main Aim of this project Automatic speed control of vehicle and air pollution detection using RF module. Nowadays in a fast moving world all the peoples are not have self-control. Such peoples are driving vehicles in a high speed in restricted areas like (school, parks, hospitals, hills area, Highways and in speed limited areas etc.) so that accidents are occurred more frequently. Because of this we lost our valuable life by making small mistakes while driving like high speed driving in restricted area, do not view the signboards which are placed by the Highway Department on the road. So in order to avoid such kind of accidents and to control their vehicle speed in restricted area like (school, parks, hospitals, hills area, Highways and in speed limited areas etc.) speed limit area, we can develop a system which can limits the speed of vehicle according to the speed limit of that particular area. So to intimate the driver about the zones and the speed limit, the project has an aim to control the speed of any vehicle automatically by means of using RF technology.

Every vehicle has its own emission of gases, but the problem is occurs when the emission of the gas beyond the standardized values. The foremost reason for this breach of emission level being the incomplete combustion of the fuel which is been supplied to the engine they occurs due to the improper servicing of vehicles on time. This emission from the vehicles which cannot be completely avoided, but in certain things we can able to control this. At this situation, in most of the countries air pollution is the major problem.

TABLE OF CONTENTS

CHAPTER 1.0

INTRODUCTION

In today's fast-moving world, as the rate of accidents and pollution is increasing day by day, speed of vehicles and pollution should be controlled as much as possible. Most of the accidents occurred in India are results of lack of speed control and violating the road rules. For this reason, different speed limits are put to decrease accidents. Unfortunately, drivers usually do not take these speed limits seriously and ignore them.Road-accidents can be prevented by adopting measures such as Traffic management, improving quality of road infrastructure and safer vehicles. To Ensure decline in accidents and to improve road safety, speed control techniques such as speed control in school and hospital zones by using RF module. Most of the manufactures has developed a laser-based control system but its cost is too high.

Every vehicle has its own emission of gases, but the problem occurs when the emission of the gas beyond the standardized values. The foremost reason for this breach of emission level being the incomplete combustion of the fuel which is been supplied to the engine they occur due to the improper servicing of vehicles on time. This emission from the vehicles which cannot be completely avoided, but in certain things we can able to control this. For this reason, we have used IOT for monitoring and detecting Air pollution.

1.1 ProblemStatement

Nowadays in a fast-moving world nobody has self-control. People are driving vehicles in a high speed in restricted areas likeschool, parks, hospitals, hills area, Highways and in speed limited areas etc. thus accidents are occurring more frequently. People do not view the signboards, sometimes they ignore it, which are placed by the Highway Department on the road. So in order to avoid such kind of accidents and to control their vehicle speed in restricted area likeschool, parks, hospitals, hills area, Highways and in speed limited areas etc. speed limit area, we have developed a system which can limit the speed of vehicle according to the speed limit of that particular area so, to intimate the driver about the zones and the speed limit, the project has an aim to control the speed of any vehicle automatically by means of RF technology. Every vehicle has its own emission of gases, but the problem occurs when the emission of the gas goes beyond the standardized values. The foremost reason for this breach of emission level being the incomplete combustion of the fuel which is been supplied to the engine they occur due to the improper servicing of vehicles on time. This emission from the vehicles which cannot be completely avoided, but in certain things we can able to control this. At this situation, in most of the countries air pollution is the major problem.

Today technology is fast-evolving, which makes it imperative that we, the students, come up with ideas that can be used in numerous fields and make the most of our product. Our project comprises a RF transmitter which is placed in the speed limit areas and a RF receiver is placed in the system which is placed inside the vehicle. RF transmitter transfers the information about the speed of the zone to the receiver which is interfaced with microcontroller. The current speed will be sensed by the proximity sensor using dc motor that also sends information to controller. The controller compares both the speeds, if speed of vehicle is greater than speed limit of the area then message is given to the driver through LCD Display to reduce the speed. And if driver does notdecreasethe speed, the control transfers automatically and after few seconds the driver can again operate it manually.

Air Pollution has now became one of the biggest problem in the World. Every vehicle has its own emission of gases, but the problem occurs when the emission of the gas beyond the standardized values. The foremost reason for this breach of emission level being the incomplete combustion of the fuel which is been supplied to the engine they occur due to the improper servicing of vehicles on time. This emission from the vehicles which cannot be completely avoided, but in certain things we can able to control this. For this reason, we have used IOT for monitoring and detecting Air pollution.

CHAPTER 2.0 LITERATURE SURVEY

PAPER 1

2.1 GSM-SMS BasedMonitoring

With the wide spread use of cellular networks, this approach is also popular when small amount of data is to be transferred through the network. Extensive work has been carried out by researchers using this approach especially in medical field.

1. Chen Peijiang and Jiang Xuehua, 2008, describe a remote monitoring system based on SMS of GSM. The system includes two parts which are the monitoring center and the remote monitoring station. The monitoring center consists of a computer and a TC35 GSM communication module. The computer and TC35 are connected by RS232. The remote monitoring station includes a TC35 GSM communication module, a MSP430F149 MCU, a display unit, various sensors, data gathering and processing unit.
2. Scanaill et al., 2006, developed a tele-monitoring system, based on short message service (SMS), to remotely monitor the long-term mobility levels of elderly people in their natural environment. Mobility is measured by an accelerometer-based portable unit, worn by each monitored subject. The portable unit houses the Analog Devices ADuC812S microcontroller board, Falcon A2D-1 GSM modem, and a battery-based power supply. Two integrated accelerometers are connected to the portable unit through the analog inputs of the microcontroller. Mobility level summaries are transmitted hourly, as an SMS message, directly from the portable unit to a remote server for long-term analysis. Each subject's mobility levels are monitored using custom-designed mobility alert software, and the appropriate medical personnel are alerted by SMS if the subject's mobility levels decrease.

3. Jiang et al., 2008, proposed a system for early diagnosis of hypertension and other chronic diseases. The proposed design consists of three main parts: a wrist Blood Pressure (BP) measurement unit, a server unit and a terminal unit. Blood Pressure is detected using data acquired by sensors intelligently using DSP microchip. The data is then transmitted to the remote server unit located at Community Healthcare Centers/Points (CHC/P) by using Short Messaging Service (SMS), and notification information is sent to the terminal unit to inform users if patient's BP isabnormal.

PAPER 2

1.2 Remote Monitoring using Wireless Sensor Networks (WSN), Bluetooth, WiFi, Zigbee technologies:

Many Wireless Technologies like RF, Wi-Fi, Bluetooth and Zigbee have been developed and remote monitoring systems using these technologies are popular due to flexibility, low operating charges, etc. Today Wireless Sensor Network are used into an increasing number of commercial solutions, aimed at implementing distributed monitoring and control system in a great number of different application areas**.**

1. Wijetunge et al., 2008, designed a general purpose controlling module designed with the capability of controlling and sensing up to five devices simultaneously. The communication between the controlling module and the remote server is done using Bluetooth technology. The server can communicate with many such modules simultaneously. The controller is based on ATMega64 microcontroller and Bluetooth communication TDK Blu2i (Class 1) module which provides a serial interface for data communication. The designed controller was deployed in a home automation application for a selected set of electrical appliances.

2. Kanma et al., 2003, proposed a home appliance control system over Bluetooth with a cellular phone, which enables remote-control, fault-diagnosisandsoftware-update for home appliances through Java applications on a cellular phone. The system consists of home appliances, a cellular phone and Bluetooth communication adapters for the appliances. The communication adapter hardware consists of a 20MHz 16bit CPU, SRAM and a Bluetooth module. The communication adapter board is connected to the home appliance and to the cellular phone through serial ports. The appliances can communicate with the cellular phone control terminal via Bluetooth SPP.
3. Sung-Nien Yu and Jen-Chieh Cheng, 2005, proposed a wireless patient monitoring system which integrates Bluetooth and WiFi wireless technologies. The system consists of the mobile unit, which is set up on the patient's side to acquire the patient's physiological signals, and the monitor units, which enable the medical personnel to monitor the patient's status remotely. The mobile unit is based on AT89C51 microprocessor. The digitized vital-sign signals are transmitted to the local monitor unit using a Bluetooth dongle. Four kinds of monitor units, namely, local monitor unit, a control center, mobile devices (personal digital assistant; PDA), and a web page were designed to communicate via the WiFi wirelesstechnology.
4. Flammini et al., 2007) suggested a novel architecture for environmental tele-monitoring that relies on GSM for sampling point delocalization, while on-field nodes implement local subnets based on the DECT technology. Local subnets contain two major blocks; Acquisition Station (AS) where sensors and actuators are located and Transmitting Module (TM), i.e., the module that handles several measurement stations and sends data to the control center (CC). Each AS acts as a data logger, storing in its internal memory device field data; communications between

ASandTMarecyclic(roundrobin),withacycletimeofabout1–10min.

PAPER 3

2.3 Internet Based Monitoring

Internet monitoring is one of the common approaches for remote monitoring. Many researchers have worked in field of Internet based remote monitoring.
(Saito et al., 2000) developed home gateway system for interconnecting home network consisting of IEEE 1394 AV network and X10 power line home automation network with Internet. This provided remote access functions from Internet for digital AV appliances like Digital Video Camera, Digital VCR connected to IEEE 1394 network and home appliances like TV, desk lamp, electric fan connected to X10controller.

1. Al-Ali and Al-Rousan, 2004, eveloped Java based home automation system via World Wide Web. The home appliances were controlled from ports of embedded system board connected to PC based server at home.
2. Alkar and Buhur, 2005, implemented Internet based wireless flexible solution where home appliances are connected to slave node. The slave nodes communicate with master node through RF and master node has serial RS232 link with PC server. The nodes are based on PIC 16F877µc. PC server is formed of a user interface component, the database and the web server components. An Internet page has been setup running on a Web server. The user interface and the Internet front end are connected to a backend data base server. The control of devices is established and their condition is monitored through theInternet.

5. Al-Khateeb et al., 2009, used X10 controller interfaced through serial port to PC server for control of devices. The Common Gateway Interface (CGI) is used to interface between the browser and the X10 protocol via http connection. The server executes CGI programs in order to satisfy a particular

request from the browser, which expresses its request using the http.

6. Peng Liu et al., 2007) developed model of web services-based email extension for remote monitoring of embedded systems which integrates web services into emails. It uses a general-purpose email messaging framework to connect devices and manipulators. This low-cost model fits for systems with low connection bandwidth, small data transportation volume and non-real time control, e.g., monitoring of home appliances and remote meter-reading.
7. (Tan and Soy, 2002) developed a system for controlling home electrical appliances over the Internet by using Bluetooth wireless technology to provide a link from the appliance to the Internet and Wireless Application Protocol (WAP) to provide a data link between the Internet and a mobile phone. However, technical details relating controller are notrevealed.

CHAPTER 3.0HARDWARE

3.1 Functional HardwareRequirements

Functional Requirement defines a function of a software system and how the system must behave when presented with specific inputs or conditions. These may include calculations, data manipulation and processing and other specificfunctionality.

The different Functional Requirement used are as follows:

- Arduino UNO
- L293DMotor Drive IC
- Node MCU ESP8266
- RF Module
- Rechargeablebattery
- Smoke Sensor
- Vibration Sensor
- Alcohol Sensor
- Temperature Sensor
- Stepper Motor

3.2 Non-FunctionalRequirements

Non-functional requirements are the requirements which are not directly concerned with the specific function delivered by the system. They specify the criteria that can be used to judge the operation of a system rather thanspecificbehaviour. Theymayrelatetoemergentsystemproperties such as reliability, response time and store occupancy. Non-functional requirements arise through the user needs, because of budget constraints, organizational policies, the need for interoperability with other software and hardware systems or because of external factors such as product Requirements. Non-functional requirements are also called the qualities of a system. These qualities can be divided into execution quality & evolution quality. Execution qualities are security & usability of the system which are observed during run time, whereas evolution quality involves testability, maintainability, extensibility or scalability.

The different Non-Functional Requirement used are as follows:

- Wheels
- Base bodyKit
- Wires
- Switches
- Buttons

3.3 HardwareComponents:

The machines, wiring, and other physical components of a computer or other electronic system are called hardware components. A hardware platform is a set of compatible hardware on which software applications can be run. Each specific hardware platform has its own machine language, and programs must be built specifically for a platform that involves a standardized type of processor and associated hardware pieces.

The components we are using hereare:

3.3.1 ARDUINO UNO

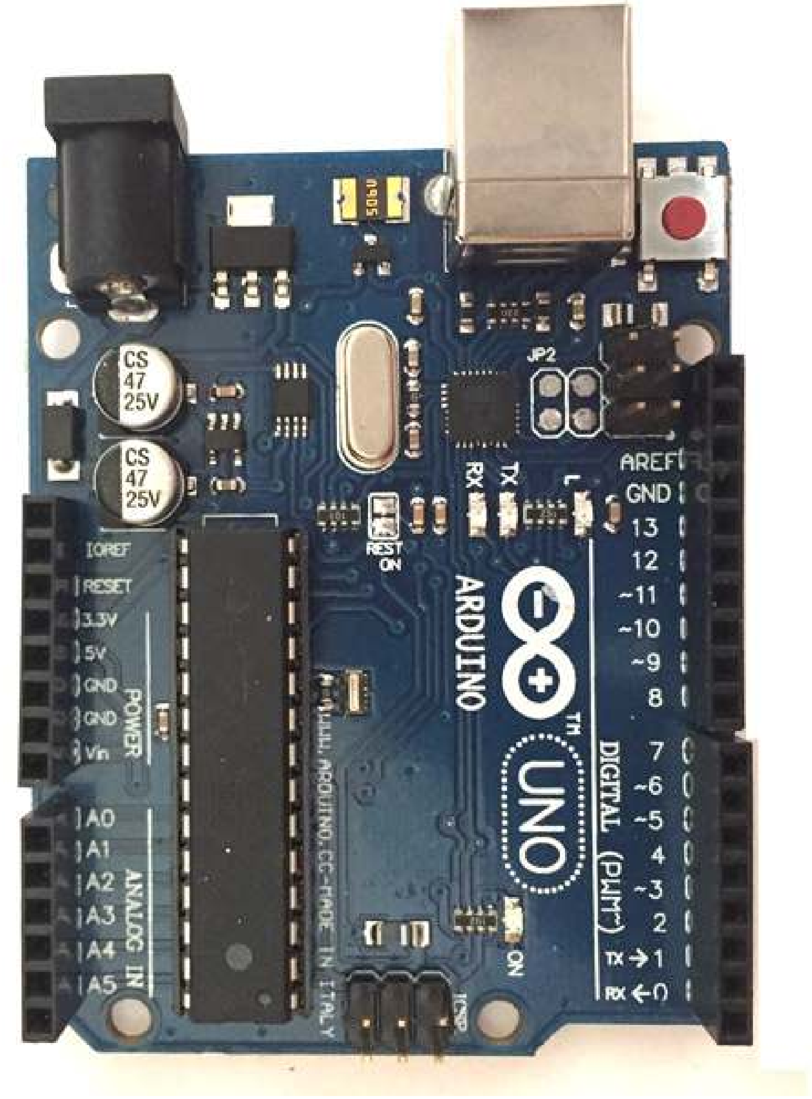


Fig. 3.1 Arduino UNO

Arduino Uno is a microcontroller board based on 8-bit ATmega328P microcontroller. Along with ATmega328P, it consists other components such as crystal oscillator, serial communication, voltage regulator, etc. to support the microcontroller. Arduino Uno has 14 digital input/output pins (out of which 6 can be used as PWM outputs), 6 analog input pins, a USB connection, A Power barrel jack, an ICSP header and a reset button.

The 14 digital input/output pins can be used as input or output pins by using pinMode(), digitalRead() and digitalWrite() functions in arduino programming. Each pin operate at 5V and can provide or receive a maximum of 40mA current, and has an internal pull-up resistor of 20-50 KOhms which are disconnected by default. Out of these 14 pins, some pins have specific functions as listed below:

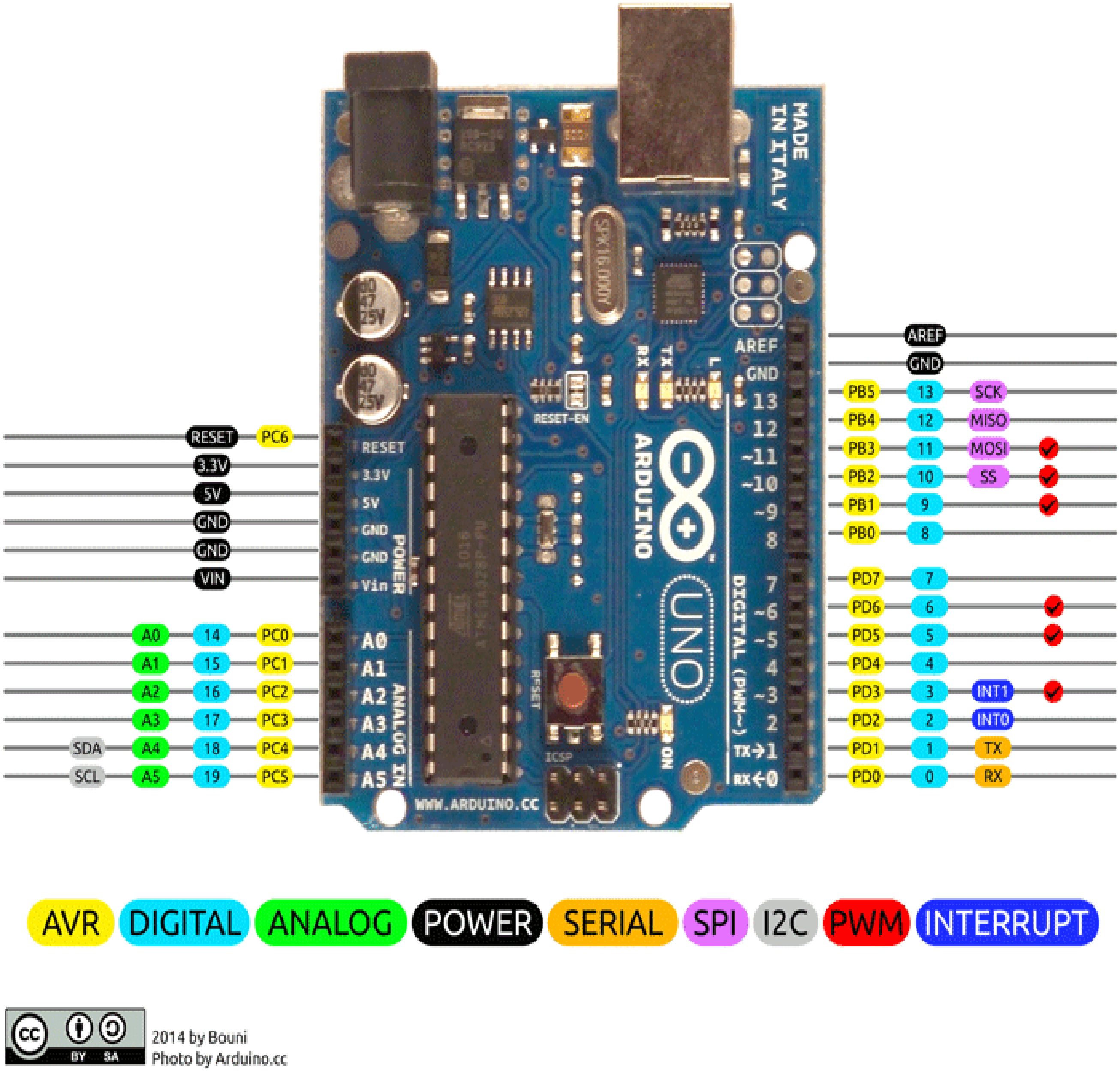
- **Serial Pins 0 (Rx) and 1 (Tx):** Rx and Tx pins are used to receive and transmit TTL serial data. They are connected with the corresponding ATmega328P USB to TTL serial chip.
- **External Interrupt Pins 2 and 3:** These pins can be configured to trigger an interrupt on a low value, a rising or falling edge, or a change in value.
- **PWM Pins 3, 5, 6, 9 and 11:** These pins provide an 8-bit PWM output by using analogWrite() function.
- **SPI Pins 10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK):** These pins are used for SPI communication.
- **In-built LED Pin 13:** This pin is connected with an built-in LED, when pin 13 is HIGH – LED is on and when pin 13 is LOW, its off.

Along with 14 Digital pins, there are 6 analog input pins, each of which provide 10 bits of resolution, i.e. 1024 different values. They measure from 0 to 5 volts but this limit can be increased by using AREF pin with analog Reference() function. Analog pin 4 (SDA) and pin 5 (SCA) also used for TWI communication using Wire library.Arduino Uno has a couple of other pins as explained below:

- **AREF:** Used to provide reference voltage for analog inputs with analogReference() function.

- **Reset Pin:** Making this pin LOW, resets the microcontroller.

3.3.1.1 PIN DIAGRAM



3.3.1.2 PIN DESCRIPTION

Vin, 3.3V, 5V, GND (POWER)

- Vin: Input voltage to Arduino when using an external power source.

- 5V: Regulated power supply used to power microcontroller and other components on the board.
- 3.3V: 3.3V supply generated by on-board voltage regulator. Maximum current draw is 50mA.
- GND: ground pins.

Reset (RESET)

Resets the microcontroller.

A0 – A5 (ANALOG PIN)

Used to provide analog input in the range of 0-5V

Digital Pins 0 – 13 (INPUT/OUTPUT PINS)

Can be used as input or output pins.

0(Rx), 1(Tx) (SERIAL)

Used to receive and transmit TTL serial data.

2, 3 (EXTERNAL INTERRUPTS)

To trigger an interrupt.

3, 5, 6, 9, 11 (PWM)

Provides 8-bit PWM output.

10 (SS), 11 (MOSI), 12 (MISO) and 13 (SCK) (SPI)

Used for SPI communication.

13 (INBUILT LED)

To turn on the inbuilt LED.

A4 (SDA), A5 (SCA) (TWI)

Used for TWI communication.

AREF (AREF)

To provide reference voltage for input voltage.

3.3.1.3 ARDRUINO UNO TECHNICAL SPECIFICATION

Microcontroller	ATmega328P – 8 bit AVR family microcontroller
Operating Voltage	5V
Recommended Input Voltage	7-12V
Input Voltage Limits	6-20V
Analog Input Pins	6 (A0 – A5)
Digital I/O Pins	14 (Out of which 6 provide PWM output)
DC Current on I/O Pins	40 mA
DC Current on 3.3V Pin	50 mA

Flash Memory	32 KB (0.5 KB is used for Bootloader)
SRAM	2 KB
EEPROM	1 KB
Frequency (Clock Speed)	16 MHz

3.3.1.4 SOFTWARE

Arduino IDE (Integrated Development Environment) is required to program the Arduino Uno board.

3.3.1.5PROGRAMMING ARDUINO

- Once arduino IDE is installed on the computer, connect the board with computer using USB cable.
- Now open the arduino IDE and choose the correct board by selecting Tools>Boards>Arduino/Genuino Uno, and choose the correct Port by selecting Tools>Port.
- Arduino Uno is programmed using Arduino programming language based on Wiring. To get it started with Arduino Uno board and blink the built-in LED, load the example code by selecting Files>Examples>Basics>Blink. Once the example code (also shown below) is loaded into your IDE, click on the 'upload' button given on the top bar.
- Once the upload is finished, you should see the Arduino's built-in LED blinking. Below is the example code for blinking:

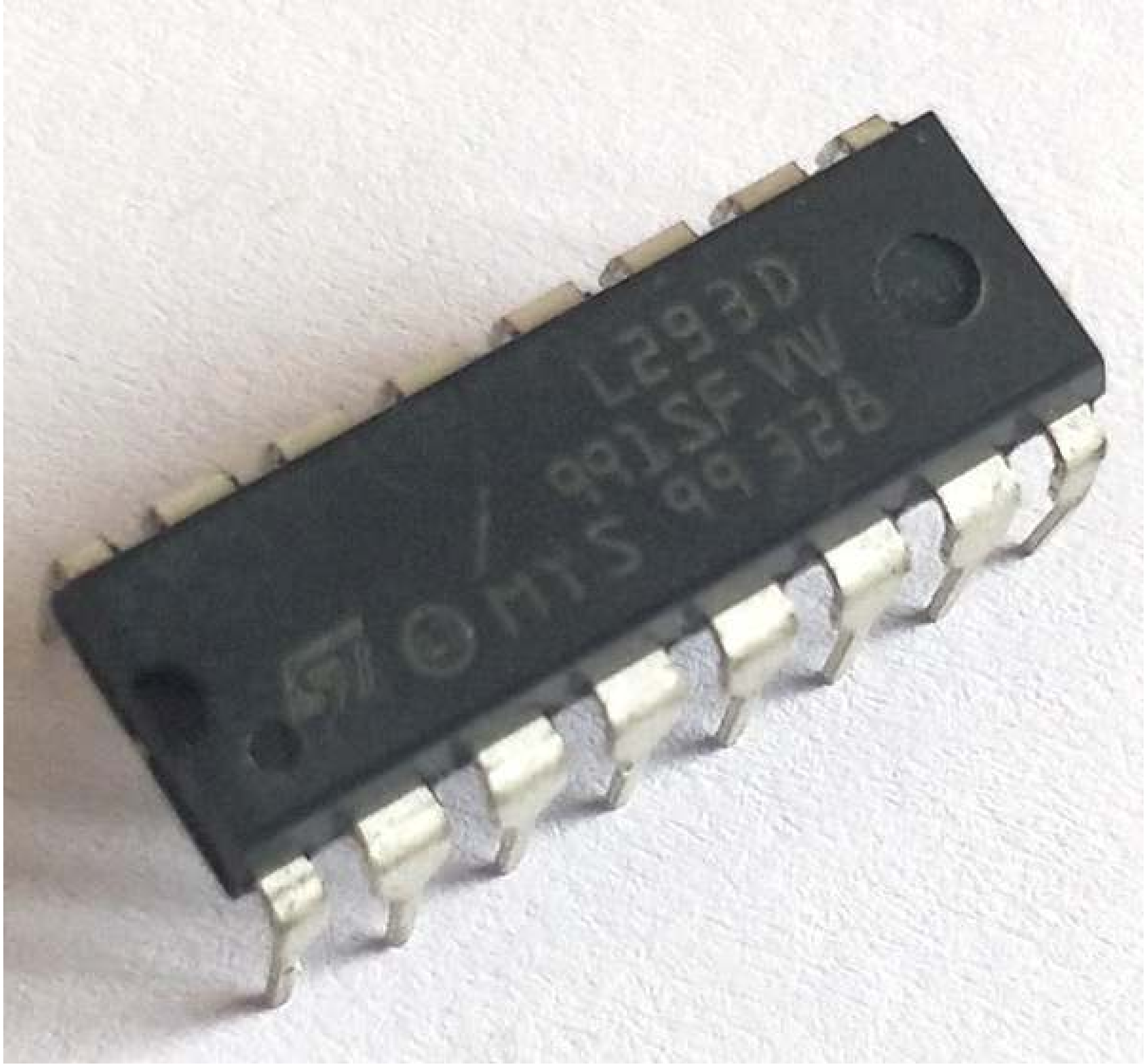
```
// the setup function runs once when you press reset or power the board
void setup() {
  // initialize digital pin LED_BUILTIN as an output.
  pinMode(LED_BUILTIN, OUTPUT);
}

// the loop function runs over and over again forever
void loop() {
  digitalWrite(LED_BUILTIN, HIGH);   // turn the LED on (HIGH is the voltage level)
  delay(1000);                       // wait for a second
  digitalWrite(LED_BUILTIN, LOW);    // turn the LED off by making the voltage LOW
  delay(1000);                       // wait for a second
}
```

3.3.1.6 APPLICATIONS

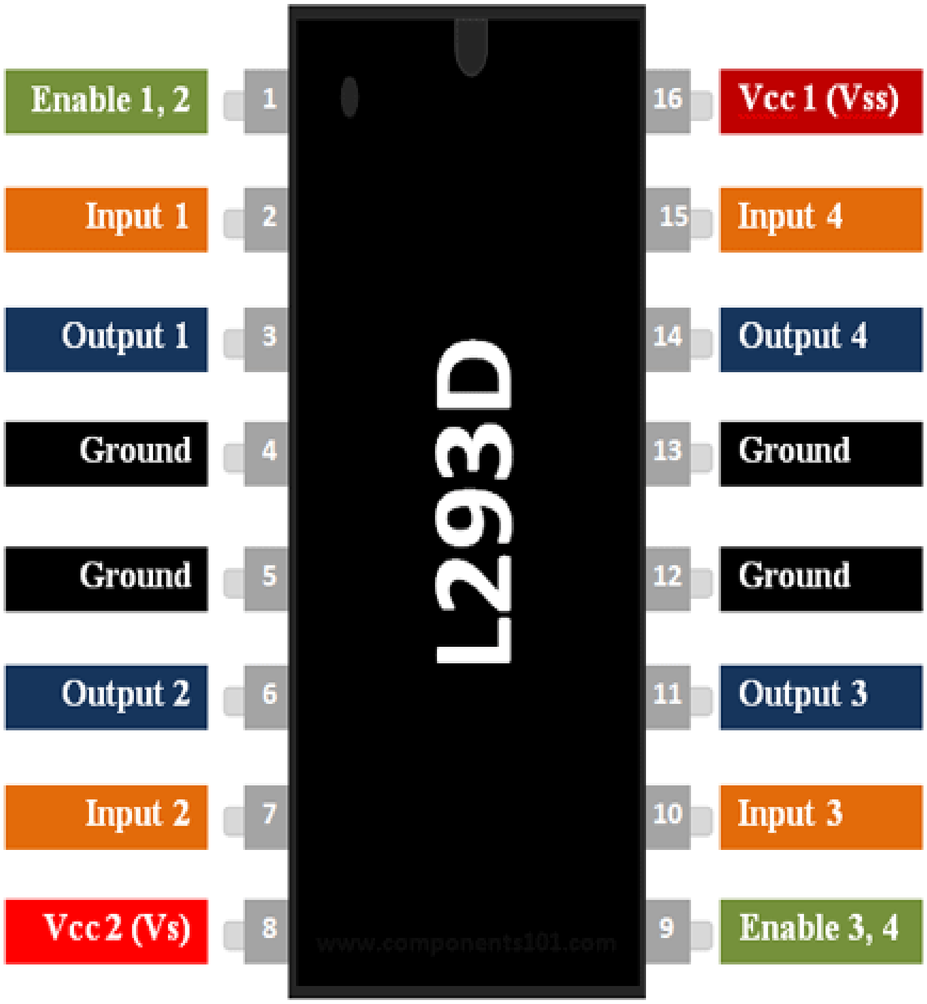
- Prototyping of Electronics Products and Systems
- Multiple DIY Projects.
- Easy to use for beginner level DIYers and makers.
- Projects requiring Multiple I/O interfaces and communications.

3.3.2 L293D MOTOR DRIVER IC



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. So if we have motors which has operating voltage less than 36V and operating current less than 600mA, which are to be controlled by digital circuits like Op-Amp, 555 timers, digital gates or even Micron rollers like Arduino, PIC, ARM etc.. this IC will be the right choice for us.

3.3.2.1 PIN DIAGRAM



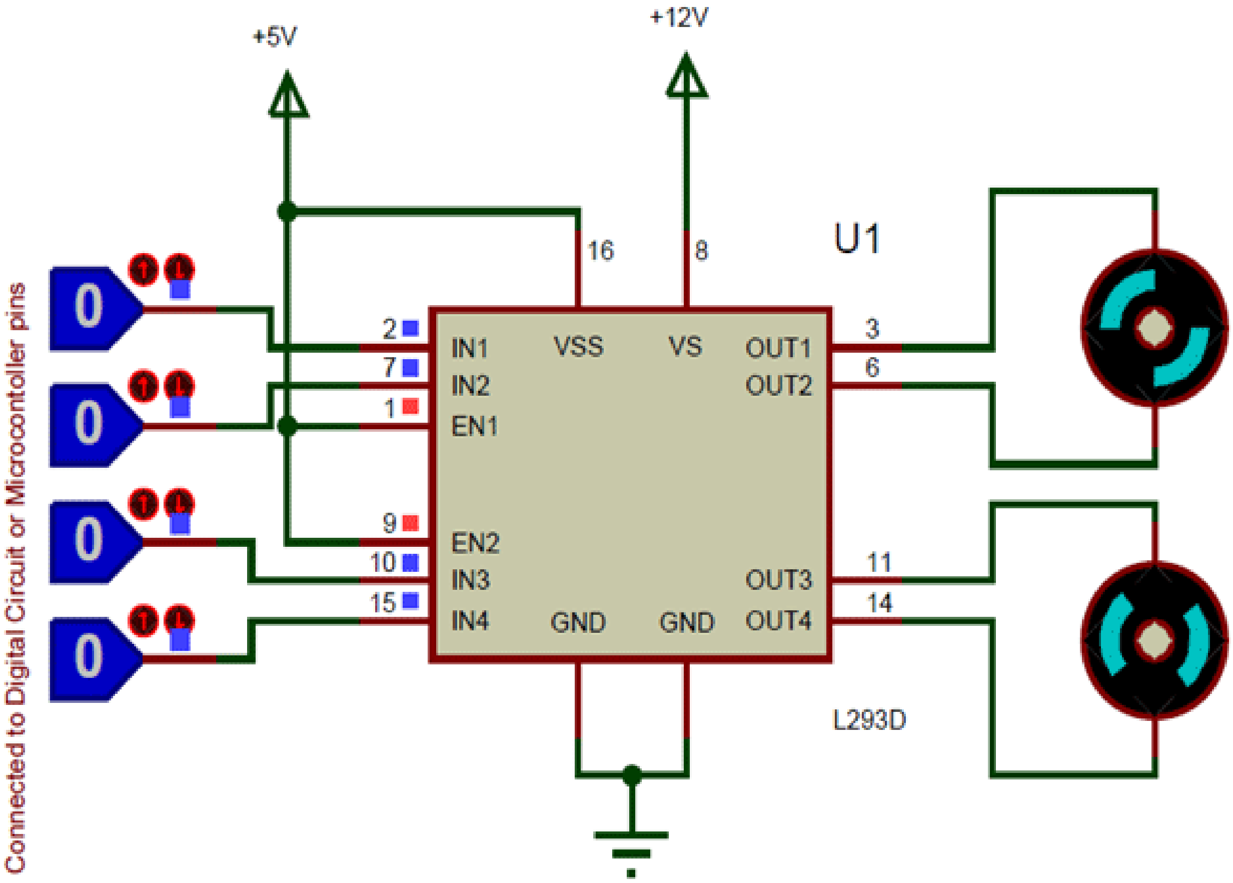
3.3.2.2 PIN DESCRIPTION

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

3.3.2.4WORKING

Using this L293D motor driver IC is very simple. The IC works on the principle of **Half H-Bridge**. H bridge is a setup which is used to run motors both in clock wise and anti-clockwise direction. 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.



- 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 microcontroller to control the speed and direction of the motor.

- We can toggle the input pins based on the following table to control our motor.

Input 1 = HIGH(5v)	Output 1 = HIGH	Motor 1 rotates in Clock wise Direction
Input 2 = LOW(0v)	Output 2 = LOW	
Input 3 = HIGH(5v)	Output 1 = HIGH	Motor 2 rotates in Clock wise Direction
Input 4 = LOW(0v)	Output 2 = LOW	

Input 1 = LOW(0v)	Output 1 = LOW	Motor 1 rotates in Anti-Clock wise Direction
Input 2 = HIGH(5v)	Output 2 = HIGH	
Input 3 = LOW(0v)	Output 1 = LOW	Motor 2 rotates in Anti -Clock wise Direction
Input 4 = HIGH(5v)	Output 2 = HIGH	

Input 1 = HIGH(5v)	Output 1 = HIGH	Motor 1 stays still
Input 2 = HIGH(5v)	Output 2 = HIGH	
Input 3 = HIGH(5v)	Output 1 = LOW	Motor 2 stays still
Input 4 = HIGH(5v)	Output 2 = HIGH	

3.3.2.5 FEATURES

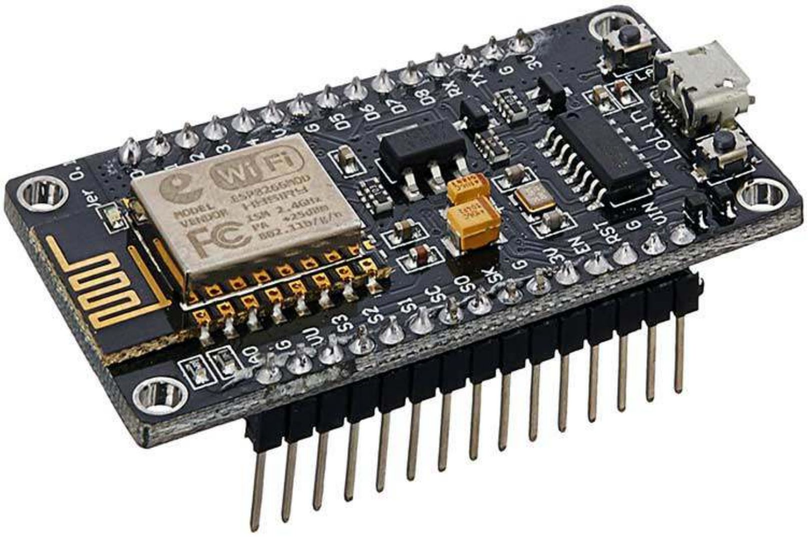
- Can be used to run Two DC motors with the same IC.
- Speed and Direction control is possible
- Motor voltage Vcc2 (Vs): 4.5V to 36V
- Maximum Peak motor current: 1.2A
- Maximum Continuous Motor Current: 600mA

- Supply Voltage to Vcc1(vss): 4.5V to 7V
- Transition time: 300ns (at 5Vand 24V)
- Automatic Thermal shutdown is available
- Available in 16-pin DIP, TSSOP, SOIC packages

3.3.2.6 APPLICATION

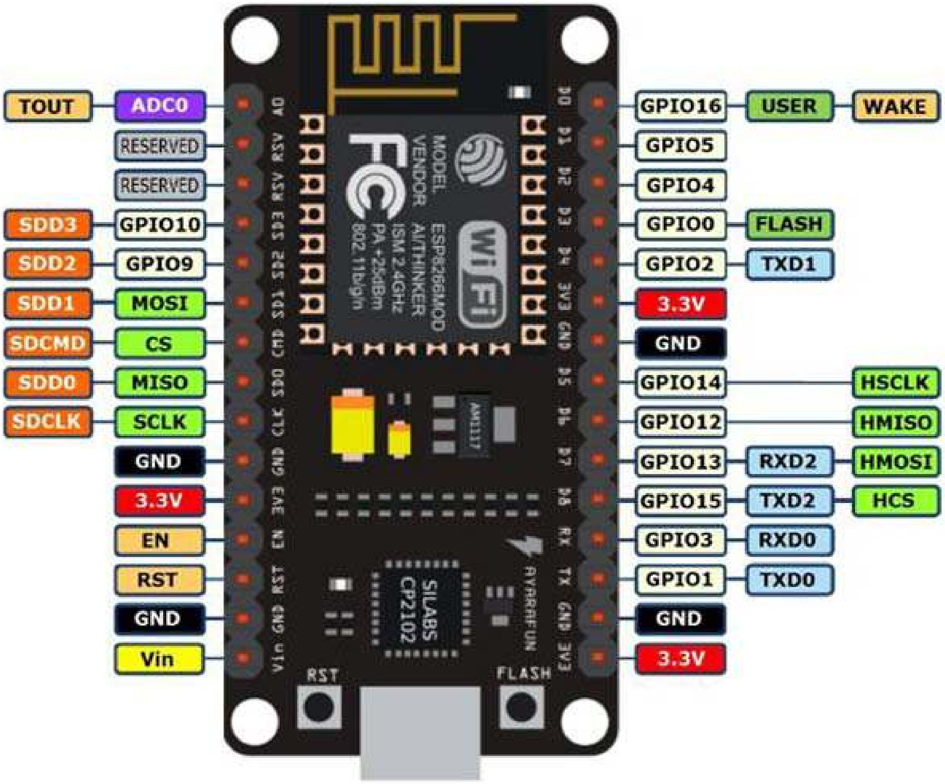
- Used to drive high current Motors using Digital Circuits
- Can be used to drive Stepper motors
- High current LED's can be driven
- Relay Driver module (Latching Relay is possible)

3.3.3 NODE MCU ESP8266



NodeMCU is an open-source Lua based firmware and development board specially targeted for IoT based Applications. It includes firmware that runs on the ESP8266 Wi-Fi SoC from Espressif Systems, and hardware which is based on the ESP-12 module.

3.3.3.1 NODE MCU ESP8266 PINOUT



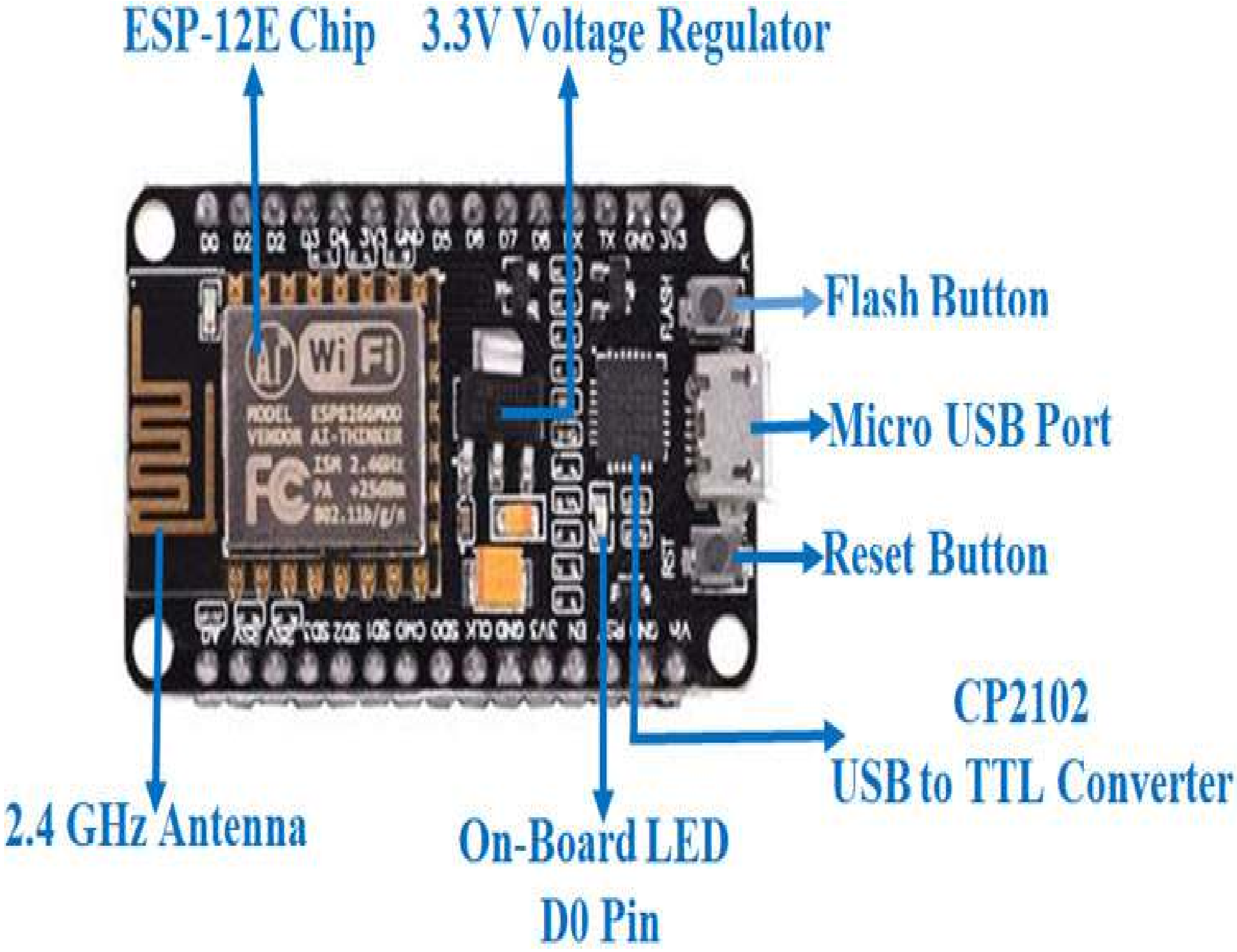
3.3.3.2 NodeMCU Development Board Pinout Configuration

Pin Category	**Name**	**Description**
Power	Micro-USB, 3.3V, GND, Vin	**Micro-USB:** NodeMCU can be powered through the USB port **3.3V:** Regulated 3.3V can be supplied to this pin to power the board **GND:** Ground pins **Vin:** External Power Supply
Control Pins	**EN, RST**	The pin and the button resets the microcontroller
Analog Pin	A0	Used to measure analog voltage in the range of 0-3.3V
GPIO Pins	GPIO1 to GPIO16	NodeMCU has 16 general purpose input-output pins on its board
SPI Pins	SD1, CMD, SD0, CLK	NodeMCU has four pins available for SPI communication.
UART Pins	TXD0, RXD0, TXD2, RXD2	NodeMCU has two UART interfaces, UART0 (RXD0 & TXD0) and UART1 (RXD1 & TXD1). UART1 is used to upload the firmware/program.
I2C Pins		NodeMCU has I2C functionality support but due to the internal functionality of these pins, you have to find which pin is I2C

3.3.3.3 BRIEF ABOUT NODE MCU ESP8266

The NodeMCU ESP8266 development board comes with the ESP-12E module containing ESP8266 chip having Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and operates at 80MHz to 160 MHz adjustable clock frequency. NodeMCU has 128 KB RAM and 4MB of Flash memory to store data and programs. Its high processing power with in-built Wi-Fi / Bluetooth and Deep Sleep Operating features make it ideal for IoT projects.

NodeMCU can be powered using Micro USB jack and VIN pin (External Supply Pin). It supports UART, SPI, and I2C interface.



3.3.3.4 Programming NodeMCU ESP8266 with Arduino IDE

The NodeMCU Development Board can be easily programmed with Arduino IDE since it is easy to use.

Programming NodeMCU with the Arduino IDE will hardly take 5-10 minutes. All you need is the Arduino IDE, a USB cable and the NodeMCU board itself.

3.3.3.5 Uploading first program

- Once Arduino IDE is installed on the computer, connect the board with the computer using the USB cable.
- Now open the Arduino IDE and choose the correct board by selecting Tools>Boards>NodeMCU1.0 (ESP-12E Module), and choose the correct Port by selecting Tools>Port.
- To get it started with the NodeMCU board and blink the built-in LED, load the example code by selecting Files>Examples>Basics>Blink.
- Once the example code is loaded into your IDE, click on the 'upload' button given on the top bar. Once the upload is finished, we should see the built-in LED of the board blinking.

3.3.3.5 NodeMCU ESP8266 Specifications & Features

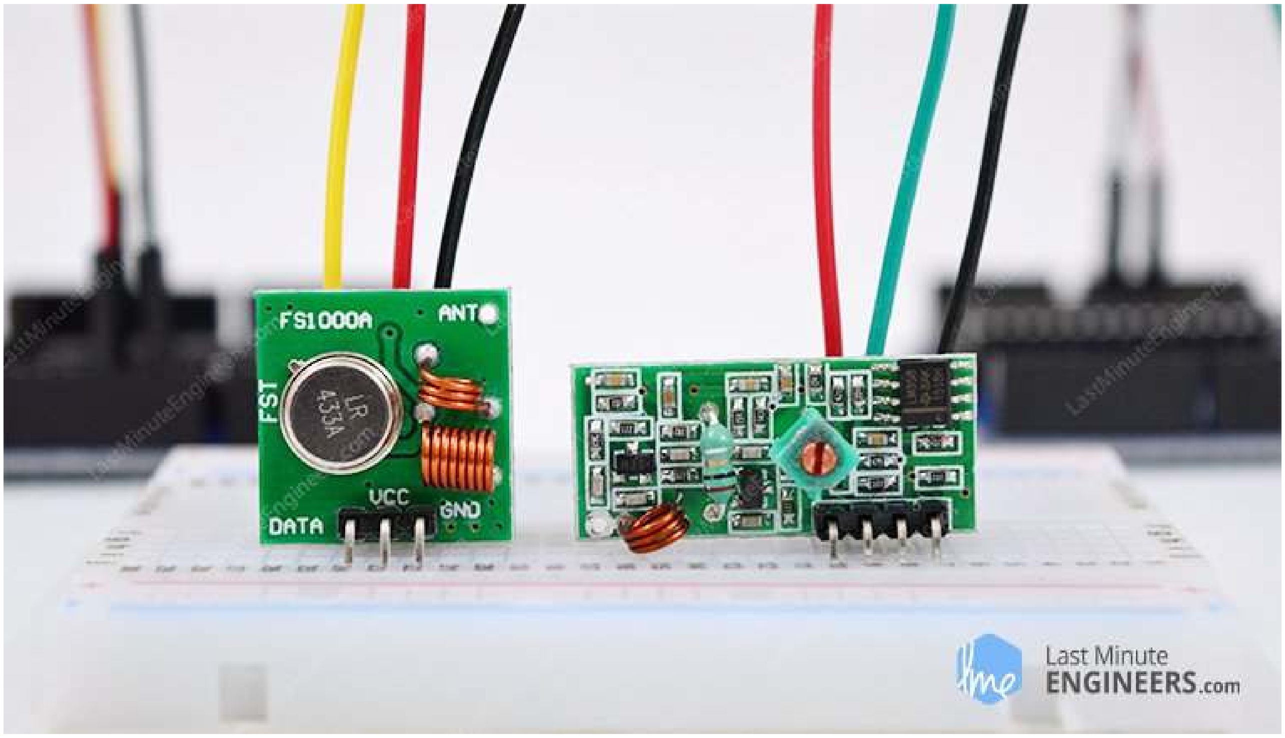
- Microcontroller: Tensilica 32-bit RISC CPU Xtensa LX106
- Operating Voltage: 3.3V
- Input Voltage: 7-12V
- Digital I/O Pins (DIO): 16
- Analog Input Pins (ADC): 1
- UARTs: 1
- SPIs: 1
- I2Cs: 1
- Flash Memory: 4 MB
- SRAM: 64 KB

- Clock Speed: 80 MHz
- USB-TTL based on CP2102 is included onboard, Enabling Plug n Play
- PCB Antenna
- Small Sized module to fit smartly inside your IoT projects

3.3.3.6 APPLICATIONS

- Prototyping of IoT devices
- Low power battery operated applications
- Network projects
- Projects requiring multiple I/O interfaces with Wi-Fi and Bluetooth functionalities

3.3.4 RF MODULE



In general, the wireless systems designer has two overriding constraints: it must operate over a certain distance and transfer a certain amount of information within a data rate. The RF modules are very small in dimension and have a wide operating voltage range i.e. 3V to 12V.

3.3.4.1 HARDWARE OVERVIEW

Basically, the RF modules are 433 MHz RF transmitter and receiver modules. The transmitter draws no power when transmitting logic zero while fully suppressing the carrier frequency thus consume significantly low power in battery operation. When logic one is sent carrier is fully on to about 4.5mA with a 3volts power supply. The data is sent serially from the transmitter which is received by the tuned receiver. Transmitter and the receiver are duly interfaced to two microcontrollers for data transfer.we use RF modules to transmitting and receive the data because it has a high volume of applications than IR. RF signals travel in the transmitter and receiver even when there is an obstruction. It operates at a specific frequency of 433MHz.

RF transmitter receives serial data and transmits to the receiver through an antenna which is connected to the 4th pin of the transmitter. When logic 0 applied to transmitter then there is no power supply in the transmitter. When logic 1 is applied to transmitter then the transmitter is ON and there is a high power supply in the range of 4.5mA with 3V voltage supply.

When a logic HIGH is applied to the DATA input, the oscillator runs producing a constant RF output carrier wave at 433.xx MHz and when the DATA input is taken to logic LOW, the oscillator stops. This technique is known as Amplitude Shift Keying.

Receiver module consists of a RF tuned circuit and a couple of OP Amps to amplify the received carrier wave from the transmitter. The amplified signal is further fed to a PLL (Phase Lock Loop) which enables the decoder to "lock" onto a stream of digital bits which gives better decoded output and noise immunity.

ASK – Amplitude Shift Keying

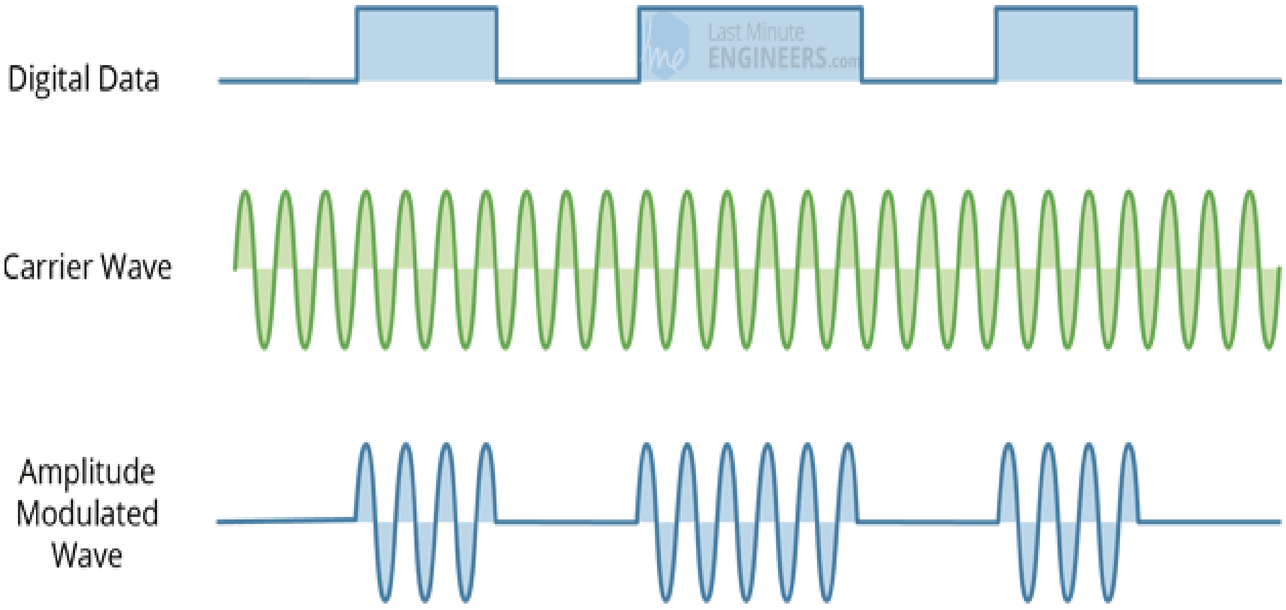
For sending the digital data over radio, these modules use a technique called Amplitude Shift Keying or ASK. In Amplitude Shift Keying the amplitude (i.e. the level) of the carrier wave (in our case it's a 433MHz signal) is changed in response to the incoming data signal.

This is very similar to the analog technique of amplitude modulation. It's sometimes called binary amplitude shift keying because there are only two levels we are concerned with. You can think of it as an ON/OFF switch.

For Digital 1 – This drives the carrier at full strength.

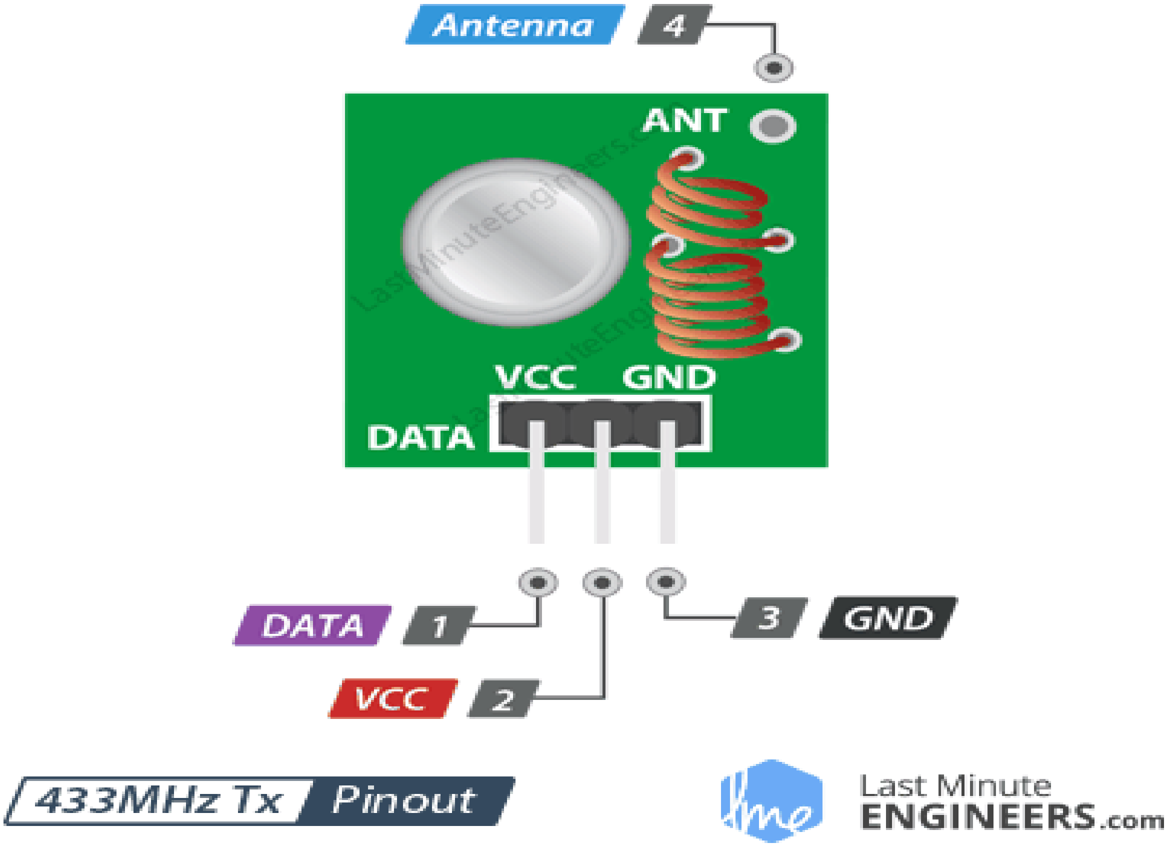
For Digital 0 – This cuts the carrier off completely.

This is how the Amplitude modulation looks like:



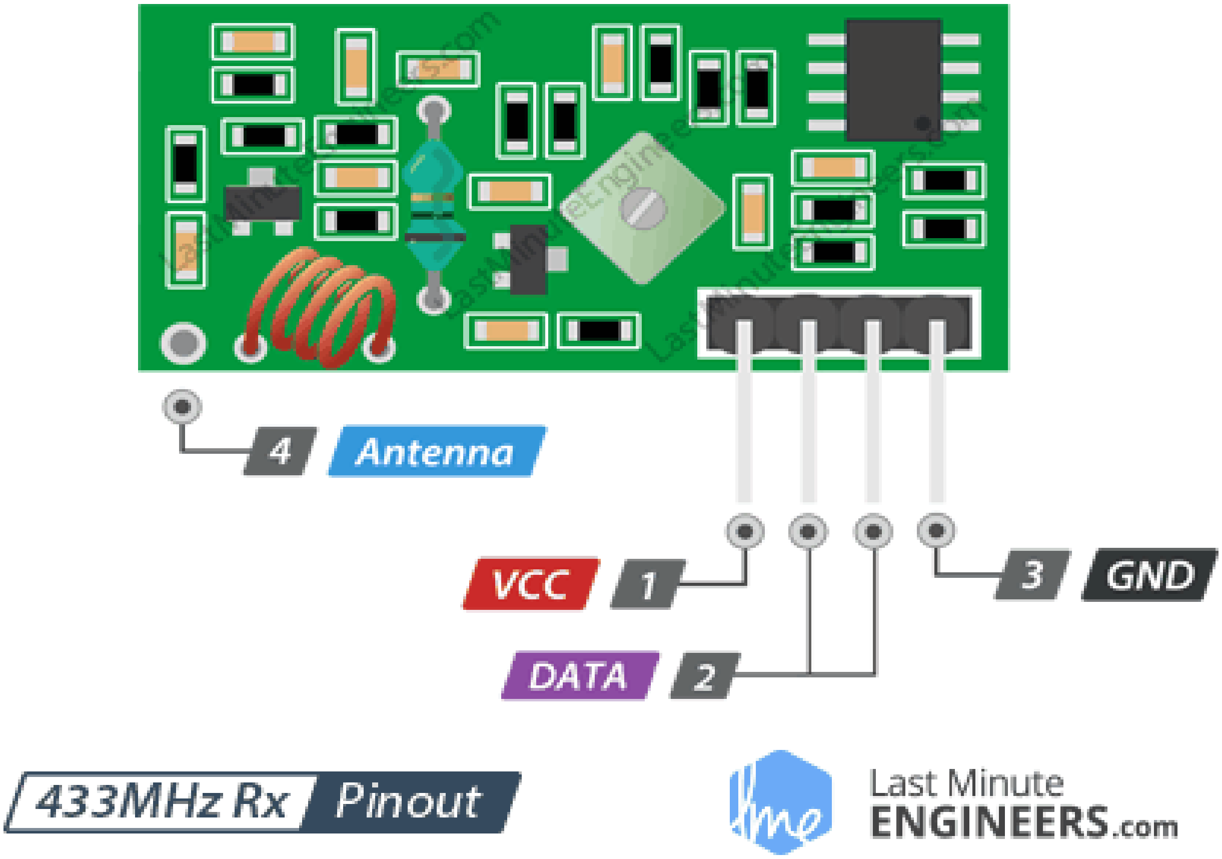
433MHz RF Transmitter Amplitude Shift Keying ASK Waveform

3.3.4.2433MHZ RF TRANSMITTER PINOUT



- **DATA** pin accepts digital data to be transmitted.
- **VCC** supplies power for the transmitter. This can be any positive DC voltage between 3.5V to 12V. Note that the RF output is proportional to the supply voltage i.e. the higher the Voltage, the greater the range will be.
- **GND** is a ground pin.
- **Antenna** is a pin for external antenna. As discussed earlier, you will want to solder a 17.3 cm piece of solid wire to this pin for the improved range.

3.3.4.3 433MHZ RF RECEIVER PINOUT



VCC supplies power for the receiver. Unlike the transmitter, supply voltage for receiver needs to be 5V.

DATA pins output the digital data received. The two center pins are internally tied together, so you can use either one for data out.

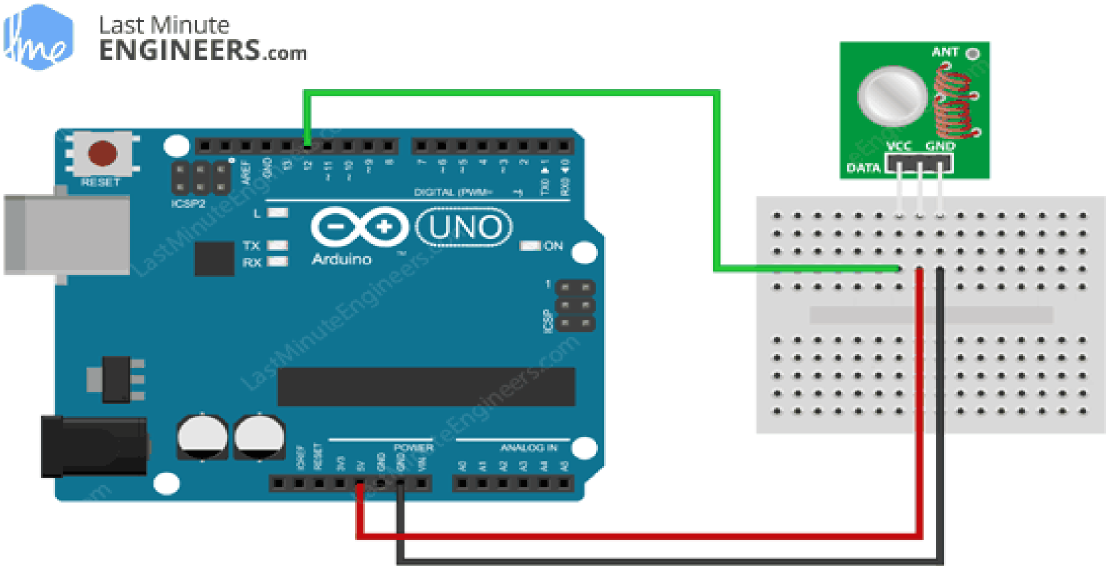
GND is a ground pin.

Antenna is a pin for external antenna which is often unmarked. It is the pad in the lower left of the module, right next to the small coil. Again, you will want to solder a 17.3 cm piece of solid wire to this pin for the improved range.

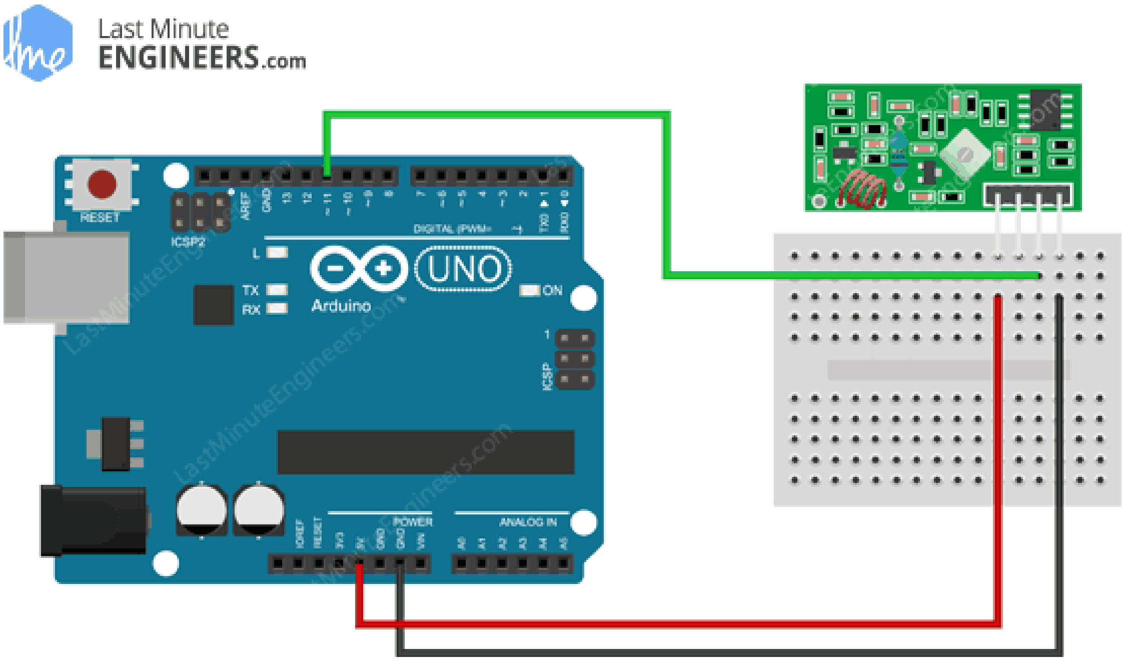
3.3.4.3 Wiring – Connecting 433MHz RF Transmitter and Receiver to Arduino UNO

We will be sending data between two Arduino boards. The wiring for the transmitter is fairly simple. It has only three connections. Connect the VCC pin to 5V pin and GND to ground on the Arduino. The Data-In pin should be connected to Arduino's digital pin #12.

The following illustration shows the wiring.

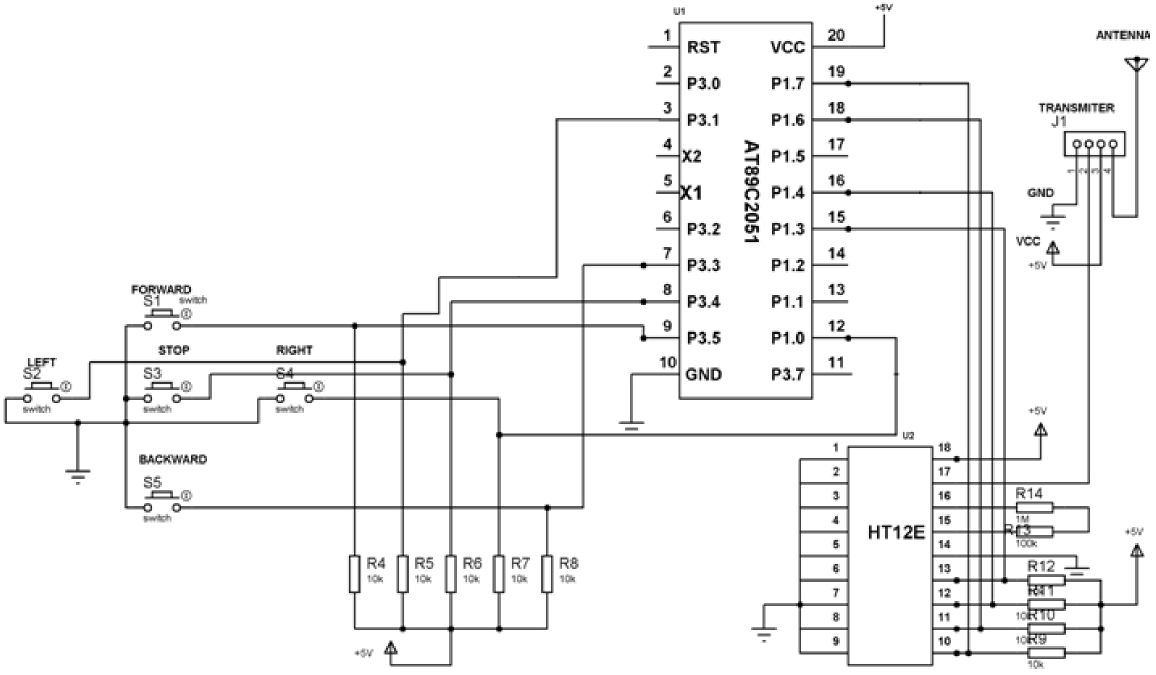


The wiring for the receiver is just as easy as the transmitter was.Once again there are only three connections to make.We have Connected the VCC pin to 5V pin and GND to ground on the Arduino. Any of the middle two Data-Out pins should be connected to digital pin #11 on the Arduino.

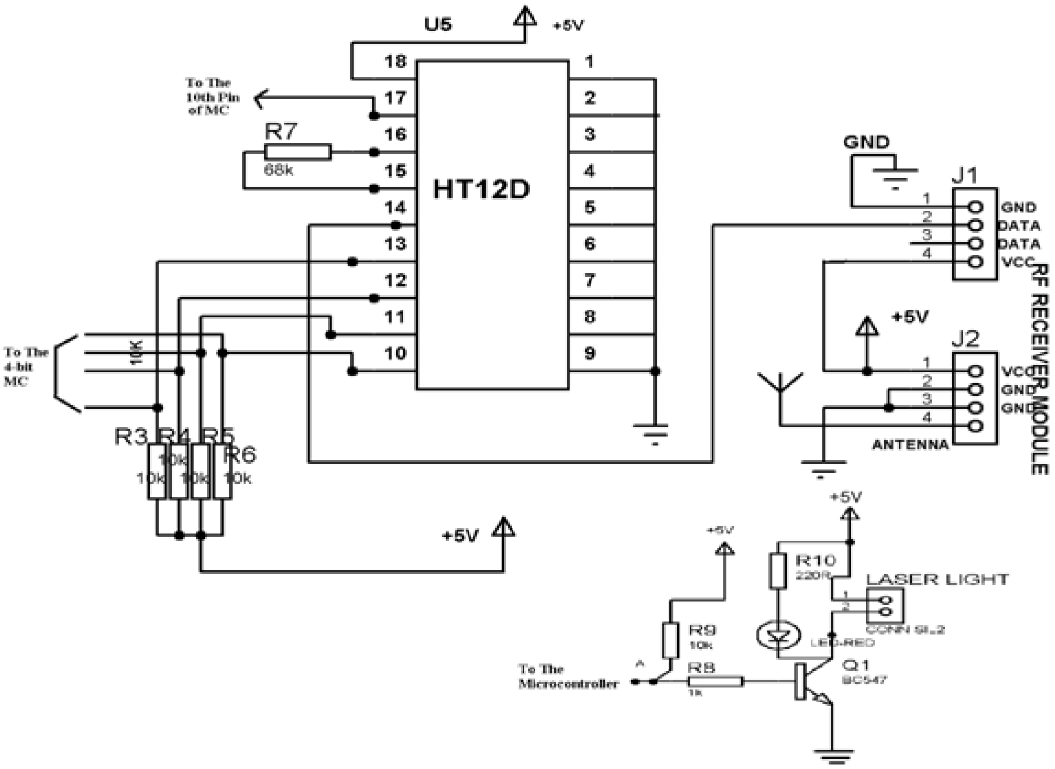


Now that both the transmitter and receiver are wired up we will need to write some code and send it to the respective Arduino boards. We will start with the transmitter. Once the code has been loaded there, we'll move on to the receiver. The Arduino to which transmitter is connected can then be powered using a power supply or battery.

3.3.4.5 RF Transmitter Circuit:



3.3.4.6 RF Receiver Circuit:



3.3.4.8 FEATURES OF RF MODULE

- Receiver frequency 433MHz
- Receiver typical frequency 105Dbm
- Receiver supply current 3.5mA
- Low power consumption
- Receiver operating voltage 5v
- Transmitter frequency range 433.92MHz
- Transmitter supply voltage 3v~6v
- Transmitter output power 4v~12v

3.3.4.8 Applications:

- Wireless security systems
- Car alarm systems
- Remote controls
- Sensor reporting
- Automation systems

CHAPTER **4.0**

SOFTWARE

The implementation phase of the project is where the detailed design is actually transformed into working code. Aim of the phase is to translate the design into the best possible solution in a suitable programming language. This chapter covers the implementation aspects of the project, giving details of the programming language and development environment used. It also gives an overview of the core modules of the project with their step by stepflow.

The implementation stage requires the following tasks.

- Carefulplanning.
- Investigation of system andconstraints.
- Design of methods to achieve thechangeover.
- Evaluation of the changeovermethod.
- Correct decisions regarding the selection of theplatform
- Appropriate selection of the language for applicationdevelopment

4.1 SoftwareUsed:

4.1.1 ARDUINO IDE

The Arduino Integrated Development Environment (IDE) is a cross-platform application (for Windows, Macos, Linux) that is written in the programming language Java. It is used to write and upload programs to Arduino board. The source code for the IDE is released under the GNU General Public License, version 2. The Arduino IDE supports the languages C and C++ using special rules of code structuring. The Arduino IDE supplies a software library from the Wiring project, which provides many common input and output procedures.

User-written code only requires two basic functions, for starting the sketch and the main program loop, that are compiled and linked with a program stub main() into an executable cyclic executive program with the GNU toolchain, also included with the IDE distribution. The Arduino IDE employs the program to convert the executable code into a text file in hexadecimal encoding that is loaded into the Arduino board by a loader program in the board's firmware.

4.1.2 BLYNK APP

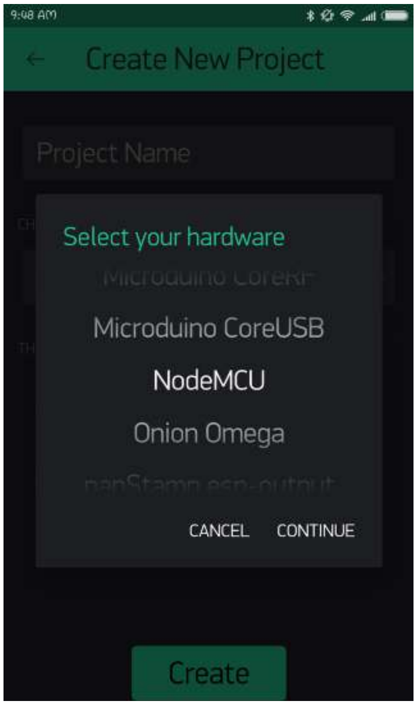
Blynk is a new platform that allows you to quickly build interfaces for controlling and monitoring your hardware projects from your iOS and Android device. After downloading the Blynk app, you can create a project dashboard and arrange buttons, sliders, graphs, and other widgets onto the screen. Using the widgets, you can turn pins on and off or display data from sensors. Whatever your project is, there are likely hundreds of tutorials that make the hardware part pretty easy, but building the software interface is still difficult. With Blynk, though, the software side is even easier than the hardware.

Currently, Blynk supports most Arduino boards, Raspberry Pi models, the ESP8266, Particle Core, and a handful of other common microcontrollers and single-board computers, and more are being added over time. Arduino Wi-Fi and Ethernet shields are supported, though you can also control devices plugged into a computer’s USB port as well.One of my favourite aspects of Blynk is that you can create a local Blynk server, allowing you to keep everything within your own home network. This is useful if you’re setting up a network in a remote location, or if you’re concerned about your traffic going through other machines in the cloud.

1.Create a Blynk Project

Click the “Create New Project” in the app to create a new Blynk app. Give it any name.

Blynk works with hundreds of hardware models and connection types. Select the Hardware type. After this, select connection type. In this project we have select WiFi connectivity.

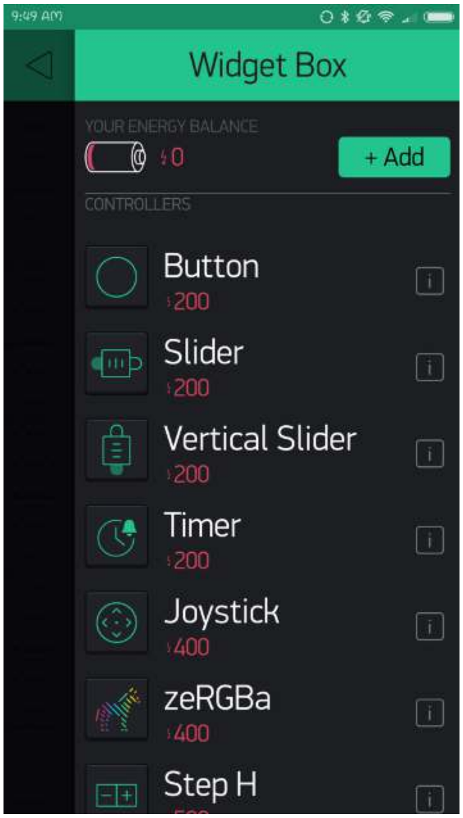


2.Add Widgets To The Project

Then you'll be presented with a blank new project. To open the widget box, click in the project window to open.

We are selecting a button to control Led connected with NodeMCU.

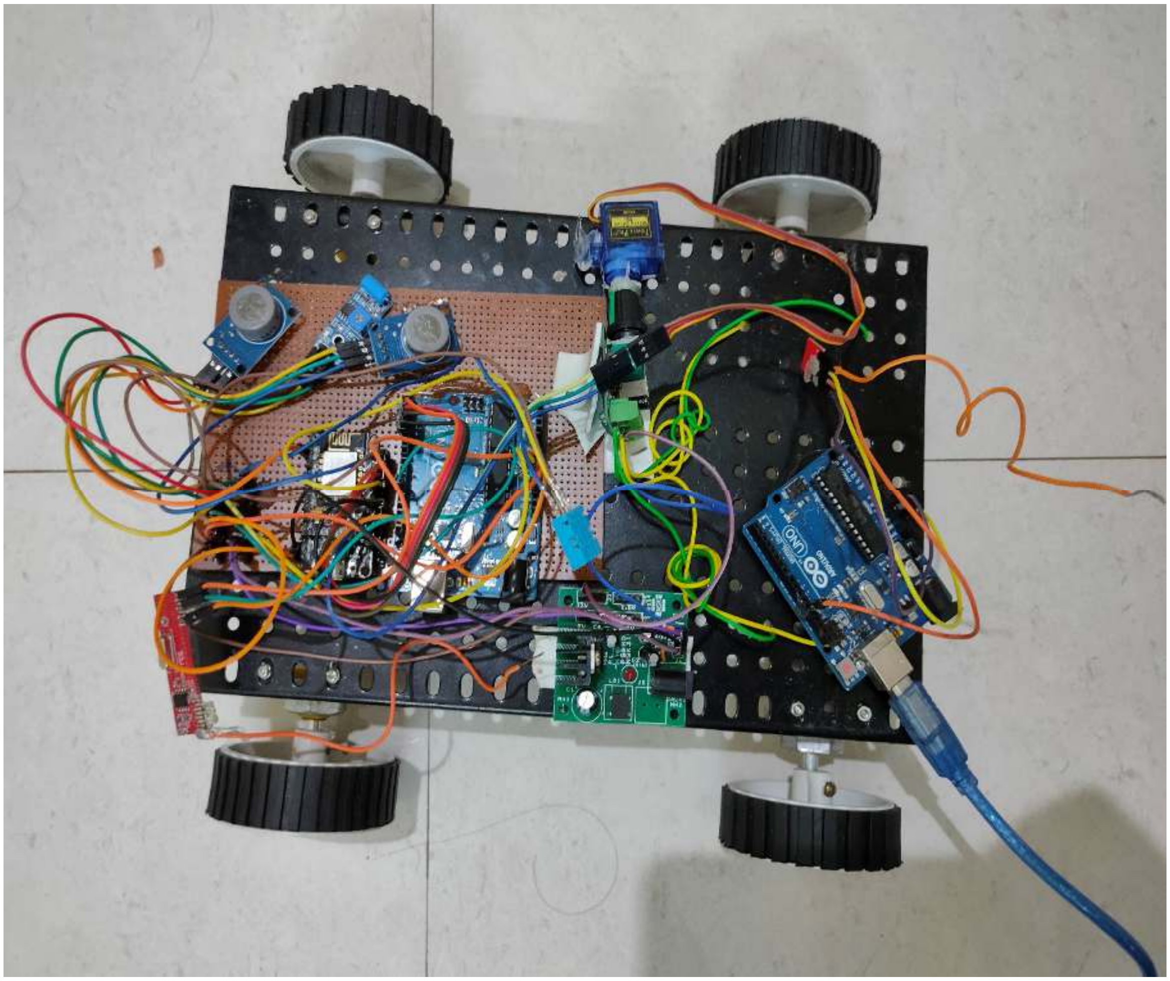
1. Click on Button.
2. Give name to Button say led.
3. Under OUTPUT tab- Click pin and select the pin to which led is connected to NodeMCU, here it is digital pin 2, hence select digital and under pin D2. And Click continue.


9:49 AM
Widget Box
YOUR ENERGY BALANCE
0
+ Add
CONTROLLERS
Button
200
Slider
200
Vertical Slider
200
Timer
200
Joystick
400
zeRGBa
400
Step H

Chapter 5.0

DESIGN AND IMPLEMENTATION

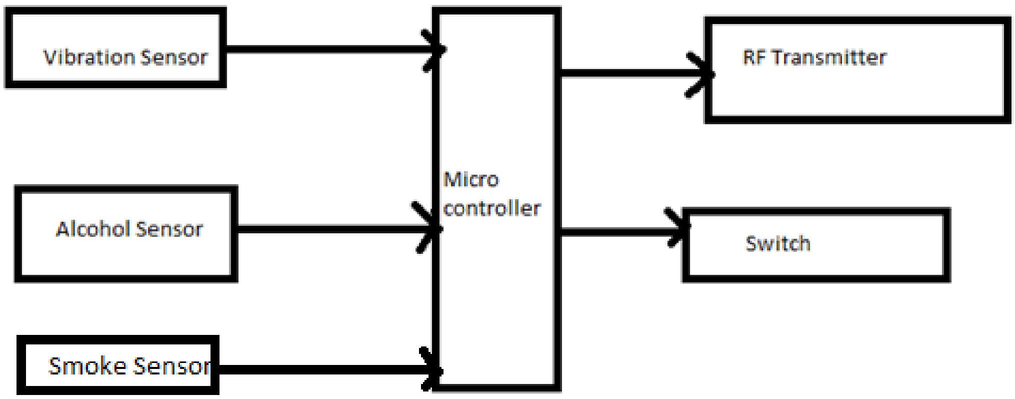
This system monitors the current pollution status andautomatic speed control of vehicles. It consists of a base with wheels and Vibration Sensor, Alcohol Sensor, Smoke Sensor, RF Transmitter, RF Receiver and Micro Controller attached on its top.



Project Design

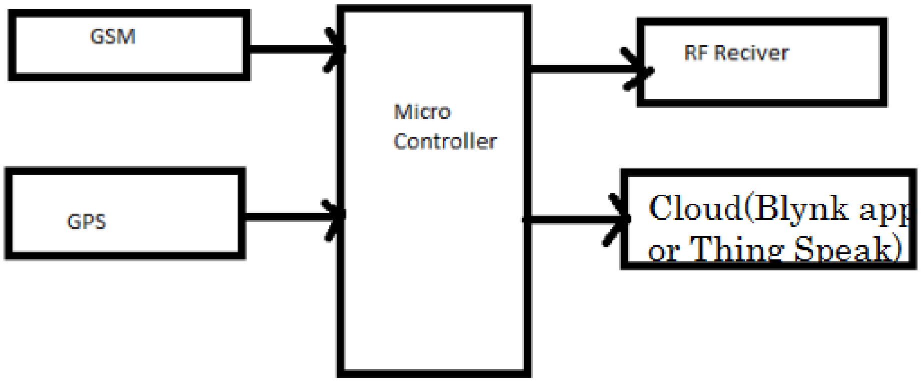
The design for each part is shown in the below sections:

RF TRANSMITTER



This Fig. shows the transmitter section of the system. The transmitter is placed in the speed limiting areas.The transmitter section consists of power supply, microcontroller and RF transmitter. This unit contains information of how much the vehicle speed inside that region.The controller is used to transmit the information through RF transmitter to an multiple receiver.

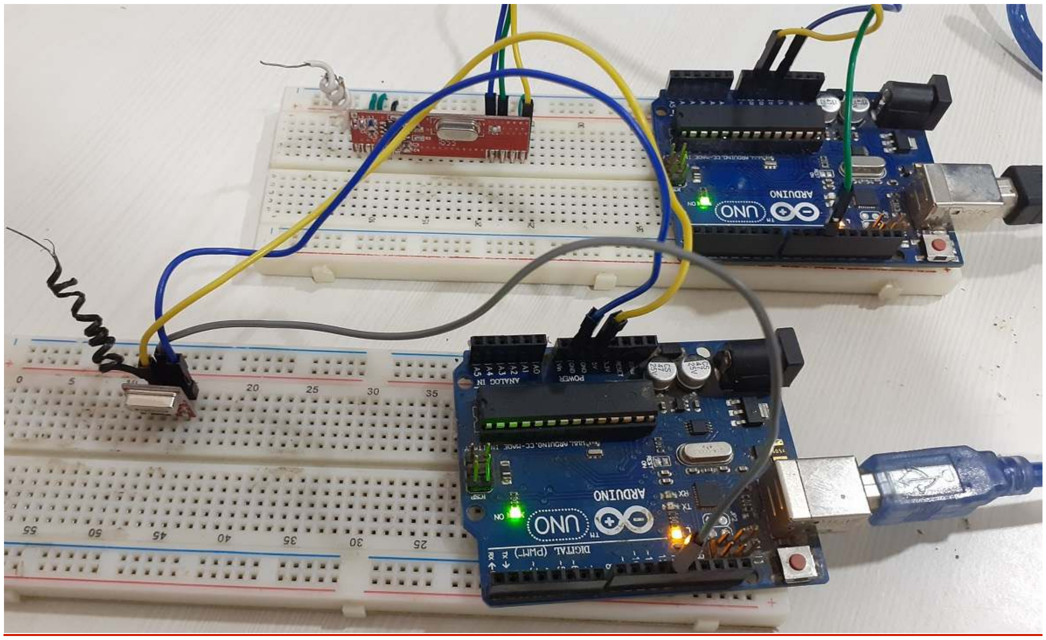
RF RECEIVER



In block diagram Sensors Send data to the micro controller and with the help of RF transmitter we can send data to the RF receiver Module.

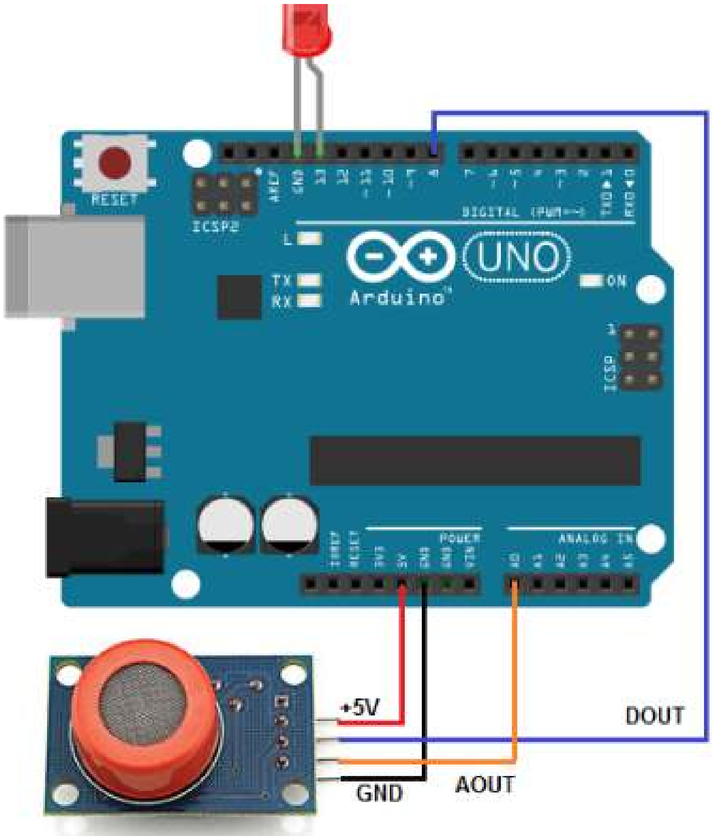
Some critical conditions we can get SMS and Location through GSM Module and GPS Module.We can send these sensor data to cloud and we will monitor through cloud.

We can see below how we have connected 433Mhz RF Module to Arduino in both the transmitter & Receiver Section.



ALCOHOL SENSOR

MQ-3 Alcohol Sensor Circuit Built with an Arduino



The alcohol sensor we will use is the MQ-3 sensor. This is a sensor that is not only sensitive to alcohol, particularly ethanol, which is the type of alcohol which is found in wine, beer, and liquor.

This type of sensor circuit can be used as a breathalyzer to check a person's blood alcohol level. Just as we exhale carbon dioxide when we breathe out, we also will breathe out some alcohol if we have alcohol in our blood. Any alcometer device can measure this alcohol content.

The more ethanol in your blood, the more there is in the air on exhalation. This alcohol content gives a good indication for if a person is drunk and how drunk they are.

The amount of alcohol exhaled into the air is proportional to the amount of alcohol which will be found in a person's blood. Alcometers use a built-in formula to estimate blood alcohol content from exhaled air alcohol content.

For different countries, the level of alcohol in the blood that defines a person as over the limit for driving varies. The range ranges from 0.01 to 0.10. Most countries have a limit of about 0.05. For example, Greece, Greenalnd, and Iceland all have limits of 0.05. Canada has a higher limit set at 0.08. In the United States, it is also 0.08. This means that if the alcometer reading measures above this, the person can receive a DUI.

For our circuit, it can function as an alcometer so that we get an estimate of a person's blood alcohol level.

SMOKE SENSOR

The MQ-2 smoke sensor is sensitive to smoke and to the following flammable gases:

LPG

Butane

Propane

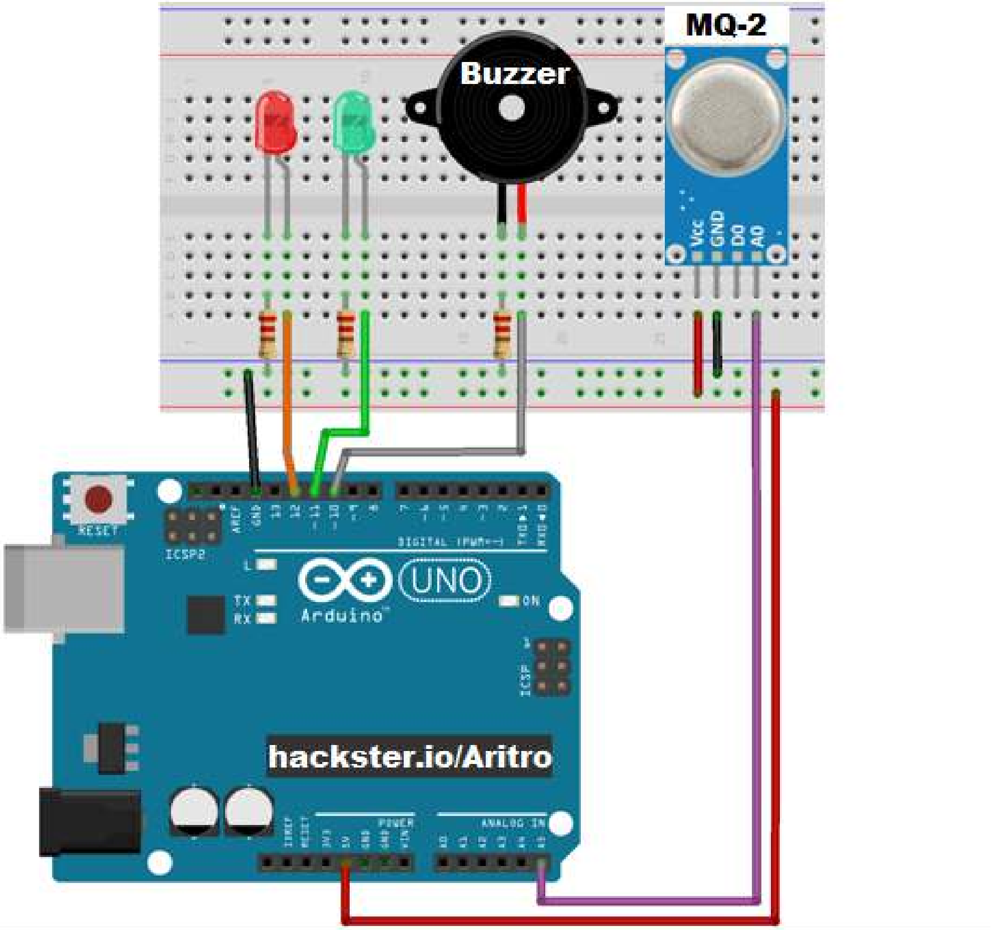
Methane

Alcohol

Hydrogen

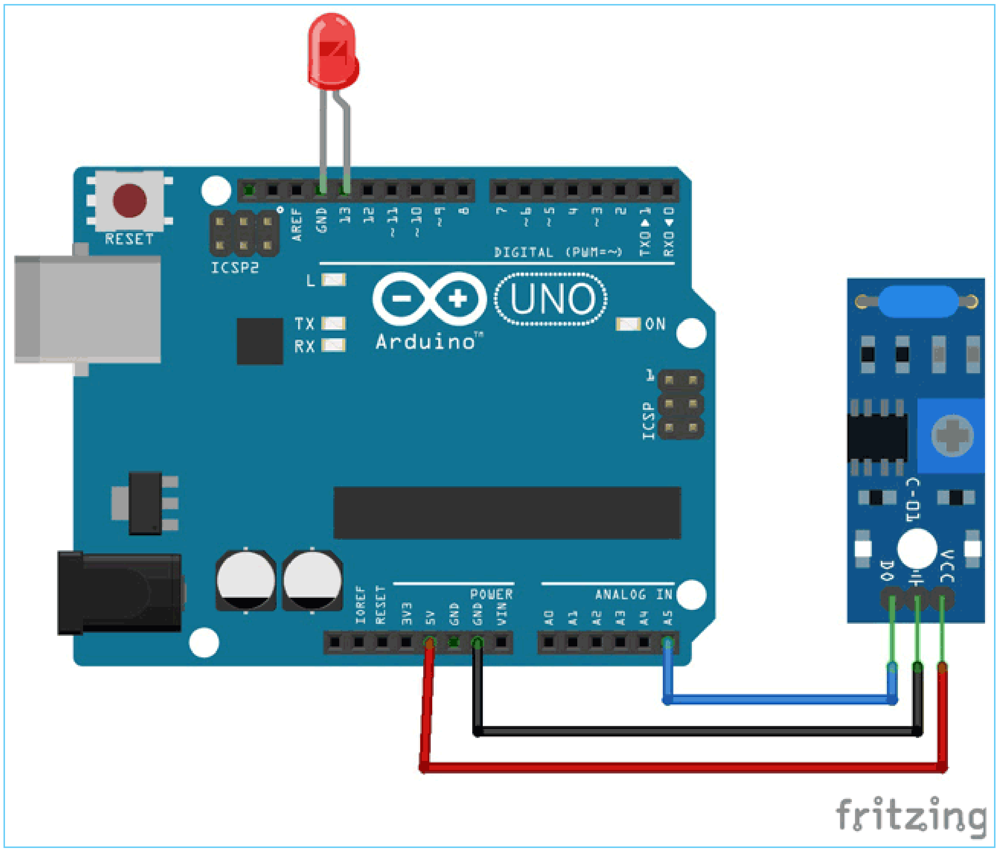
The resistance of the sensor is different depending on the type of the gas.

The smoke sensor has a built-in potentiometer that allows you to adjust the sensor sensitivity according to how accurate you want to detect gas.



Vibration Sensor:

There are several critical machines or expensive equipmentwhich suffer damages due to vibrations. In such a case, a vibration sensor is required to find out whether the machine or equipment is producing vibrations or not. Identifying the object which is continuously vibrating is not a tricky job if the proper sensor is used to detect the vibration. There are several types of vibration sensors available in the market which can detect the vibration bysensing acceleration or velocity and could provide excellent result. However, such sensors are too expensive where the accelerometer is used. Accelerometer is very sensitive and can be used to make Earthquake detector circuit. But, there are few dedicated and cheap sensors are also available to detect the vibrations only, one such vibration sensor is SW-420 which we are going to interface with Arduino Uno**.**



5.1 Implementation

CODE FOR NODE MCU ESP8266

```
#define BLYNK_PRINT Serial
#include <Servo.h>

 // create servo object to control a servo
// twelve servo objects can be created on most boards

#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
#include <SimpleTimer.h>
#include "DHT.h"
#define DHTPIN D4
#define DHTTYPE DHT11
char auth[] = "zpcHMBQH3jVaDaR3fNpO0RhHy5v9S51C";

// Your WiFi credentials.
// Set password to "" for open networks.


char ssid[] = "abcdefg";
char pass[] = "12345678";
int p=0;
DHT dht(DHTPIN, DHTTYPE);
WidgetLCD lcd(V0);
WidgetLCD lcd1(V1);
WidgetLCD lcd2(V2);
WidgetLCD lcd3(V3);
int a=0,pos=0;
int b=0,c=0;

Servo myservo;
void setup() { myservo.attach(D7);

 //D1 servo
  // put your setup code here, to run once:

  Serial.begin(9600);
    dht.begin();
    pinMode(A0,INPUT);//gass
    pinMode(D0,INPUT);//vibration
    pinMode(D1,INPUT);//ALCOHAL
     pinMode(D2,INPUT);//CONDI
      pinMode(D3,INPUT);//CONDI
       pinMode(D5,INPUT);//CONDI
        pinMode(D6,INPUT);//CONDI


    Blynk.begin(auth, ssid, pass);
    lcd.clear(); //Use it to clear the LCD Widget
    lcd1.clear();
    lcd2.clear();
}

void loop() {

 int k1= digitalRead(D2);
 int k2= digitalRead(D3);
 int k3= digitalRead(D5);
 int k4= digitalRead(D6);

if(k1==1 && k2==0 && k3==0 && k4==0)
{  Serial.println(" aaa ");

    myservo.write(90);              // tell servo to go to position in variable 'pos'
   // delay(15);
     lcd3.print(0,0 ,  "10km");
}
else if(k1==0 && k2==1 && k3==0 && k4==0)
{    Serial.println(" bb ");


    myservo.write(120);


// tell servo to go to position in variable 'pos'
  //  delay(15);

 lcd3.print(0,0 ,  "20km");
}
else if(k1==0 && k2==0 && k3==1 && k4==0)
{     Serial.println(" cc ");

    myservo.write(150);
```

```
     // tell servo to go to position in variable 'pos'
  //  delay(15);

     lcd3.print(0,0 ,  "30km");

}
else if(k1==0 && k2==0 && k3==0 && k4==1)
{
  Serial.println(" 10 ");

    myservo.write(180);

    // tell servo to go to position in variable 'pos'
   // delay(15);

     lcd3.print(0,0 ,  "40km");
}

  // put your main code here, to run repeatedly:
 float h = dht.readHumidity();

  // Read temperature as Celsius (the default)
  float t = dht.readTemperature();


  // Read temperature as Fahrenheit (isFahrenheit = true)
  float f = dht.readTemperature(true);

  // Check if any reads failed and exit early (to try again).
  if (isnan(h) || isnan(t) || isnan(f)) {
    Serial.println(F("Failed to read from DHT sensor!"));
    return;
  }

  // Compute heat index in Fahrenheit (the default)
  float hif = dht.computeHeatIndex(f, h);

  // Compute heat index in Celsius (isFahreheit = false)
  float hic = dht.computeHeatIndex(t, h, false);
 a=analogRead(A0);
 b=digitalRead(D0);
  c=digitalRead(D1);
  Serial.print(F("Humidity: "));
  Serial.print(h);
  Serial.print(F("%  Temperature: "));
  Serial.print(t);
  Serial.print(F("°C "));
  Serial.print(f);
  Serial.print(F("°F  Heat index: "));
  Serial.print(hic);
  Serial.print(F("°C "));
  Serial.print(hif);
  Serial.println(F("°F"));
   lcd.print(0,0 ,  "temp");
   lcd.print(9, 0,  hif);
   lcd.print(0,1 ,  "humidity");
   lcd.print(13, 1,  h);
   lcd1.print(0,0 ,  "CO2");
   lcd1.print(12, 0,  a);

   if(a>=500)
   {
    // myservo.write(0);
      p++;
      if(p==5)
      {
      lcd1.print(0,1 ,  "c02 detected");
      lcd1.print(12, 1,  "stop");
      }
   }
   else
   {
     lcd1.clear();
    lcd1.print(0,0 ,  "CO2");
   lcd1.print(12, 0,  a);
    lcd1.print(0,1 ,  "c02");
   lcd1.print(14, 1,  a);
   }
   if(b==0)
   {
   lcd2.print(0,1 ,  "Meet with accident");
   }else
   {
     lcd2.print(0,1 ,  "going good");
   }

   Blynk.run();
}
```

CODE FOR RF TRANSMITTER

```
#include <VirtualWire.h>

void setup()
{
  // Initialize the IO and ISR
  vw_setup(2000); // Bits per sec
  pinMode(A1, OUTPUT);   // where the motor is connected to
    pinMode(A0, OUTPUT);
}

void loop()
{
  //send("Hello, it's me");
  delay(5000);

  send("A");

   delay(5000);


   send("B");

   delay(5000);

   send("C");

   delay(5000);

   send("D");

   delay(5000);

}

void send (char *message)
{
  vw_send((uint8_t *)message, strlen(message));
  vw_wait_tx(); // Wait until the whole message is gone
}
```

CODE FOR RF RECEIVER

```
#include <VirtualWire.h>

byte message[VW_MAX_MESSAGE_LEN]; // a buffer to store the incoming
messages
byte messageLength = VW_MAX_MESSAGE_LEN; // the size of the message



void setup()
{


  pinMode(2,OUTPUT);// trigger one
  pinMode(3,OUTPUT);
  pinMode(4,OUTPUT);
  pinMode(5,OUTPUT);//triger two
  Serial.begin(9600);
  Serial.println("Device is ready");
  // Initialize the IO and ISR
  vw_setup(2000); // Bits per sec
  vw_rx_start(); // Start the receiver

}

void loop()
{

  if (vw_get_message(message, &messageLength)) // Non-blocking
  {
    Serial.print("Received: ");
    for (int i = 0; i < messageLength; i++)
    {
      Serial.write(message[i]);
      delay(100);
      if(message[i]==65)
    {
      Serial.println("speed 10km");
      digitalWrite(2,HIGH);
```

```
     digitalWrite(3,LOW);
     digitalWrite(4,LOW);
     digitalWrite(5,LOW);
   //  delay(100);

    }
   else if(message[i]==66)
    {
     Serial.println("speed 20km");
      //myservo.write(50);
        digitalWrite(2,LOW);
     digitalWrite(3,HIGH);
     digitalWrite(4,LOW);
     digitalWrite(5,LOW);
     // delay(100);

    }
   else if(message[i]==67)
    {
     Serial.println("speed 30km");
      //myservo.write(90);
        digitalWrite(2,LOW);
     digitalWrite(3,LOW);
     digitalWrite(4,HIGH);
     digitalWrite(5,LOW);
      //delay(100);

    }
    else if(message[i]==68)
    {
     Serial.println("speed 40km");
      //myservo.write(125);
        digitalWrite(2,LOW);
     digitalWrite(3,LOW);
     digitalWrite(4,LOW);
     digitalWrite(5,HIGH);
      //delay(100);

    }
    }
  // Serial.println();
  }
}
```

CHAPTER 6.0

RESULTS

The main working principle behind this is IOT which collects information from the cloud which consists of information about the pollution status which is present in our environment.

The microcontroller which is used in this device is that Arduino microcontroller which consists of 6 outputs and 6inputs so that many sensors can be clubbed together which totally sums up together as an pollution detector and monitoring using an IOT device.

This system can be applied to any kind of vehicles. The system can checks the speed of vehicle using proximity sensor and sends the message to the driver to lower down the speed, if speed is higher than speed of that area.

Hence, this project is a great life saving system in speed limit areas. Thus, it is designed in such a way that to minimize the speed of vehicle in restricted areas. By using this system, we can get the information about the speed limit of that particular area. This project is very useful for the common people to walk safely in the roads of speed restricted zones and also drivers can ride their vehicles safely.

CHAPTER 7.0

APPLICATIONS AND ADVANTAGES

This project has multiple and varied applications:

- This system can be applied to any kind of vehicles.
- It has a great importance in termination and reduction to a large extent of accidents and causalities in cramped areas.
- The system can check the speed of vehicle using proximity sensor and sends the message to the driver to lower down the speed, if speed is higher than speed of that area.
- . This project is very useful for the common people to walk safely in the roads of speed restricted zones and also drivers can ride their vehicles safely.
- we can implement sensors on this existing device so that it can automatically sense the presence of a person and act accordingly.

CHAPTER 8.0

CONCLUSION AND SCOPE FOR FUTURE WORK

8.1 CONCLUSION

This system can be applied to any kind of vehicles. The system can check the speed of vehicle using proximity sensor and sends the message to the driver to lower down the speed, if speed is higher than speed of that area.

Hence, this project is a great life-saving system in speed limit areas. Thus, it is designed in such a way that to minimize the speed of vehicle in restricted areas. By using this system, we can get the information about the speed limit of that particular area. This project is very useful for the common people to walk safely in the roads of speed restricted zones and also drivers can ride their vehicles safely.

8.1 FUTUREWORK

Most of the accidents occurred in India are results of lack of speed control and violating the road rules. For this reason, different speed limits are put to decrease accidents. Unfortunately, drivers usually do not take these speed limits seriously and ignore them.To Ensure decline in accidents and to improve road safety, speed control techniques such as speed control in school and hospital zones by using RF transceiver. In this way by adding different speed limit to different areas, we can reduce road accidents.

Our project is “Automatic Speed Control of Vehicle and Air Pollution detection using IOT”, so it has a great importance in termination and reduction to a large extent of accidents and causalities in cramped areas.

we can implement sensors on this existing device so that it can automatically sense the presence of a person and act accordingly.

With the help of IOT we can have smart windows that will dynamically respond to changing light condition of the house thus saving our energy consumption.

REFERENCES

1. REFERENCES A,Vengadesh,K.Sekar “Automatic speed control of vehiclesin restricted areas using RF and GSM” International Research Journal of Engineering and Technology(IRJET)Volume-02 Issue-09 Dec 15.

2. Ankita Mishra, JyotiSolanki “Design of RF based speed control system for vehicles”International Journal of Advanced Research in computer and communication engineering(IJARCCE)Volume-01 Issue09October2012.

3. V.ChandraPrasad,G.Pradeepkumar “Automatic Speed control in Restricted Areas” International Journal of Scientific Research (IJSR)Volume-04 Issue-04 April 2015.

4. Amarnarayan, challasaikumar,Chandra Mohan “Automatic over sed controlling of vehicle”International Journal of Combined Research and Development(IJCRD)Volume-05 Issue-05 May2016

5. P.Liu,D.Zhou, and N.wu:varied density based various spatial clustering of applications with noise in service systems and service management,2007 International conference on june2007,pp1-4

6. D.Birant and A,kut an algorithm of clustering spatial-temporal data and knowledge engineering, vol 60,no1,intelligent data mining

7. A.A.L.Kama,H.Hammadou,B.Meurisse,an d C.Papaix,’Le pic de pollution a paris du 12 au 17 mars 2014’tech.Rep.

8. K. Chakrabarty, S. S. Iyengar, H. Qi, and E. Cho, "Grid coverage for surveillance and target location in distributed sensor networks," Computers, IEEE Transactions on, vol. 51, no. 12, pp. 1448–1453,2002.