VISION & MISSION OF THE INSTITUTE

VISION

To induce higher planes of learning by imparting technical education with

- International Standards
- Applied Research
- Creative Ability
- Value based instruction and to merge as a premier institute

MISSION

Achieving academic excellence by providing globally acceptable technical education by fore casting technology through

- Innovative Research And development
- Industry Institute Interaction
- Empowered Manpower

VISION & MISSION OF THE DEPARTMENT

VISION

To be a center of excellence and renowned for Electronics and Communication Engineering and research

MISSION

M1: Enlighten the graduates in the basic concepts under laying the principles of analog and digital electronics, communication systems and advanced technologies.

M2: Provide state of the art infrastructure and research facilities
M3: Organizing industrial programs and social activities in collaboration with industries, NSS to disseminate knowledge.

A Project Report

on

DESIGN AND DEVELOPMENT OF ROBOTIC VEHICLE FOR ISOLATION WARD

Submitted in partial fulfillment of the requirements for the award of the degree of

BACHELOR OF TECHNOLOGY IN

ELECTRONICS & COMMUNICATION ENGINEERING

By

MADDURI MOUNIKA	20MH1A04F0
KAYALA SAI SUDHA	20MH1A04F2
JOKA RAMYA	20MH1A04E6
KARNIDI VEERENDRA	21MH5A0464

Under the Esteemed Guidance of

Mr.P.BHUPA REDDY M.Tech(Ph.D)

Assistant Professor

Department of Electronics & Communication Engineering

ADITYA COLLEGE OF ENGINEERING(A)

(Approved by AICTE, New Delhi & Permanently Affiliated to JNTUK, Kakinada, accredited by NBA & NAAC)

Recognized by UGC under sections 2(f) & 12(b) of UGC Act, 1956

Aditya Nagar, ADB road, Surampalem, E.G.Dt, A.P-533437

2020-2024

ADITYA COLLEGE OF ENGINEERING(A)

DEPARTMENT OF ELECTRONICS AND COMMUNICATION ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that the Project Report entitled

DESIGN AND DEVELOPMENT OF ROBOTIC VEHICLE FOR ISOLATION WARD

being submitted by

MADDURI MOUNIKA	20MH1A04F0
KAYALA SAI SUDHA	20MH1A04F2
JOKA RAMYA	20MH1A04E6
KARNIDI VEERENDRA	21MH5A0464

In partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Electronics and Communication Engineering at Aditya College of Engineering is a bonafide work carried out by them under my guidance and supervision.The result embodied in this report has not been submitted to any other University or Institution for the award of any degree or diploma.

Project Guide
Mr. P.BHUPA REDDY M.Tech(Ph.D).
Assistant Professor
Department of ECE
Aditya College of Engineering(A)

Head of the Department
CH. JANAKI DEVI, M.Tech(Ph.D).
Associate Professor
Department of ECE
Aditya college of Engineering(A)

EXTERNAL EXAMINER

DECLARATION

We hereby declare that the entire project work embodied in this dissertation entitled **“Design and development of robotic vehicle for isolation ward”** has beenindependently carried out by us. As per our knowledge,no part of this work has submittedfor any degree in any institution, university and organization previously. We here by boldly state that to the best of our knowledge our work is free from plagiarism.

Yours Sincerely,

MADDURI MOUNIKA	**20MH1A04F0**
KAYALA SAI SUDHA	**20MH1A04F2**
JOKA RAMYA	**20MH1A04E6**
KARNIDI VEERENDRA	**21MH5A0464**

ACKNOWLEDGEMENT

We express our sincere gratitude and heartful thanks to the under stated person for the successful completion of our final project on **"Design and development of robotic vehicle for isolation ward".**

First and foremost we wish to thank our beloved guide **Mr.P.BHUPA REDDY** for his kind guidance, valuable advises and utmost care at every stage of our final project.

We would like to thank Prof **CH. JANAKI DEVI,** HOD of E.C.E who have provided vital information, which was necessary for success of the project.

The credit in successful completion of our final project goes to highly esteemed **Dr. A.RAMESH** sir, Principal of **ADITYA COLLEGE OF ENGINEERING** for his kind guidance, commendable and noteworthy advises and even memorable encouragement to the same.

Yours Sincerely,

MADDURI MOUNIKA	**20MH1A04F0**
KAYALA SAI SUDHA	**20MH1A04F2**
JOKA RAMYA	**20MH1A04E6**
KARNIDI VEERENDRA	**21MH5A0464**

ABSTRACT

The COVID-19 pandemic has affected the health systems severely. It is very difficult task for the all countries of the entire world to make isolated wards and treat the patients in this pandemic situation. Although majority of people of the world have taken vaccine, third wave is started in the majority part of the world. Vaccines are thought to be the best available solution for controlling the ongoing SARS-CoV-2 pandemic. However, the emergence of vaccine-resistant strains may come too rapidly for current vaccine developments to alleviate the health, economic and social consequences of the pandemic. Indian Health Ministry designated Delta Plus a Variant of Concern (VOC) on 22nd June, 2021 citing its perceived increased transmissibility, ability to bind more strongly to receptors on lung cells, and potential to evade an antibody response. Corona virus strain ‘Omicron’ was declared a variant of concern in November 2021 by WHO. So it is clear that several more waves will come in coming years. For health workers and medical staff it is very difficult to sanitize themselves frequently and do the work in an isolated ward while wearing a PPE kit. The work presented here deals with a smart robotic vehicle that can supply a medicine, food, etc. in COVID-19 isolation ward by moving in the entire space. For hospital staff, robotic vehicle can significantly reduce the risk of infectious disease and make possible to monitor and treat patients from a safe distance.

INDEX

CONTENTS **PAGENO**

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

PREAMBLE

1.1 INTRODUCTION

A robot is an electro mechanical machine that is controlled by computer program for various operations. Advancement in robotic vehicles using different techniques is going on in the world. For durability and long lasting performance, EV battery design technique can be used for robotic vehicles. Some advanced and intelligent robotic vehicles are also being designed in the world and such robotic vehicle is a base of this work. Various modern tools along with robotics are used today to fight covid-19. Robotics and AI can be used to measure temperature and blood pressure, to serve food and drug to sanitize the isolation wards. According to data received from Government of India, the daily cases of Covid-19 were 306064 on 24th January 2022. This number of cases may reach up to 4 lakhs per day during peak of third wave in the end of January 2022. Although after fully vaccinated, peoples are get infected in third wave. Health workers and doctors are also getting infected rapidly in third wave. It is in priority for every nation to protect health workers and doctors from Covid-19 (and any other infectious diseases) so that healthcare system can work efficiently in current pandemic situation. The vehicle can be controlled from a safe distance to avoid contact of the patient. If the prototype will become successful, then it will be made in more compact size, so that not only hospitals, but any individual can use it in his premises. Robot is a re-programmable, multifunctional device which is primarily designed to do work like a human such as pick and place, loading and unloading, surveillance, health care, industrial, aerospace application. Aim of this work is to design and control the motion of robot using Bluetooth of an android phone in isolation ward. As per the commands received from android application the robot motion can be controlled. The motion of a robotic vehicle presented here is very accurate. A wireless camera can be mounted on the robot vehicle for spying and surveillance purpose even in night time

1.2 EXISTING SYSTEM

Robots are becoming frequent in almost every industry, from healthcare and manufacturing to defense and education. Also, it is the most promising evaluation to the upcoming world. All set to play an important role in this upcoming society. By and large, robots are customized to perform explicit assignments. This restricts the utilization of these robots. To build the utilization of robots where conditions are unsure, for example, firefighting or salvage activity we can make robots which adhere to the directions of a human administrator and play out the undertaking. Along these lines' choices are taken by the working conditions by the administrator and undertaking is performed by the robots in this manner we can utilize these robots to play out those errands that might be unsafe to people. Likewise, this framework is introduced for the society at minimum cost. Which can be used in any type of pandemic situation in hospitals for the safety of doctors and other staffs in the hospitals. Also, it will help in all types of hospital usage which will surely human effort and interference with the critical patients. As it can be modified as per required and all necessary changes can be made.

1.3 PROPOSED SYSTEM

The work presented here deals with a smart robotic vehicle that can supply a medicine, food, etc. in COVID-19 isolation ward by moving in the entire space. For hospital staff, robotic vehicle can significantly reduce the risk of infectious disease and make possible to monitor and treat patients from a safe distance.

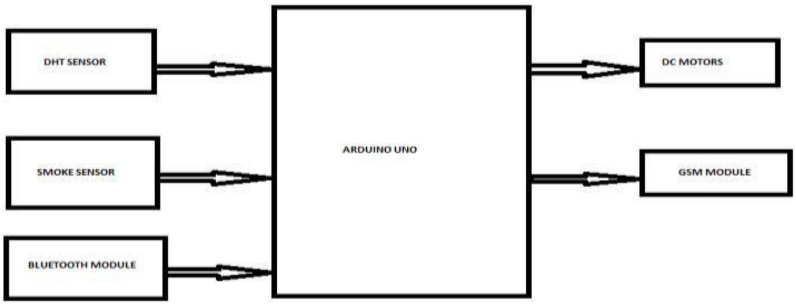


Fig 1.1: Block Diagram Of Proposed System

CHAPTER 2

LITERATURE SURVEY

A group of vehicles is tasked to reach a number of target points at the same time (simultaneous arrival problem) without colliding among themselves and with obstacles, subject to the requirement that the overall energy required for vehicle motion be minimized. With the theoretical setup adopted, the vehicle dynamics are explicitly taken into account at the planning level. This paper formulates the problem of multiple-vehicle motion planning in a rigorous mathematical setting, describes the optimization algorithm used to solve it, and discusses the key implementation details. The efficacy of the method is illustrated through numerical examples for the simultaneous arrival problem. The initial guess to start the optimization procedure is obtained from simple geometrical considerations, e.g., by joining the desired initial and final positions of the vehicles via straight lines. Even though the initial trajectories thus obtained may result in intervehicle and vehicle/obstacle collisions, we show that the optimization procedure that we employ in this paper will generate collision-free trajectories that also minimize the overall energy spent by each vehicle and meet the required temporal and spatial constraints. The method developed applies to a very general class of vehicles; however, for clarity of exposition, we adopt as an illustrative example the case of wheeled robots.**[1]**

Consumer demand for Electric Vehicles (EVs) is increasing due to improving performance and affordability. However, EV manufacturers are struggling to meet this rise in demand. A key bottleneck is supply from a nascent EV battery supply chain that is new and developing. In this paper, we propose robotic work cell design for fast and reliable assembly of EV battery modules, at scale, to reduce this demand-supply gap. **[2]**

The COVID-19 pandemic has hit the global at a colossal scale. With worldwide reported cases of 5.34 million it has led to severe impact on humanity. Being a highly contagious disease, it has given global health services their most severe challenge. Various countries are fighting to minimize the losses due to the outbreak, however a common trait is enforcing lockdown, which has become the main defence mechanism. Researchers are working around the clock to find a breakthrough in the diagnostics and treatment of the pandemic. AI technology is useful for fast drug development and treatment. In the starting phase of COVID-19 pandemic, the medical fraternity in China diagnosed the virus using computed tomography (CT) and X-ray images due to the limitation of testing kits. Deep learning neural network model have also been used for COVID-19 diagnosis. AI assisted intelligent humanoid robots can be used to reduce the human contact and spread of COVID-19. In Italy robots have been used for measuring blood pressure, oxygen saturation and temperature of patients. Robots have also found applications in disinfecting and sterilizing of public places, COVID-19 testing, food and medicine delivery as well as entertaining patients in hospitals and quarantine centers, thereby reducing the workload of doctors and nurses. Prediction of the spread of virus and providing the guidelines or prevention measures is another AI application in COVID-19. Kaggle and GitHub are the two websites where the real-time data of COVID-19 is aggregated. This includes confirmed cases, active cases, cured cases and deaths in each country. This data set can be used for predicting the active cases across different regions of the world so that appropriate amount of health infrastructure can be made available to these places.[**3]**

CHAPTER 3

SOFTWARE REQUIREMENT SPECIFICATIONS

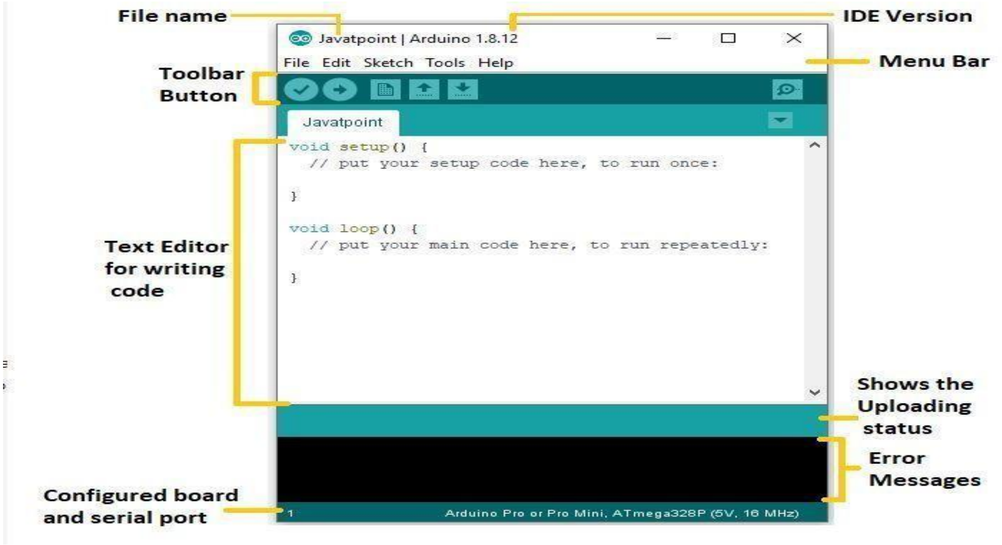
3.1 INTRODUCTION

The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

3.2 SOFTWARE REQUIREMENTS

The program or code written in the Arduino IDE is often called as sketching. We need to connect the Genuino and Arduino board with the IDE to upload the sketch written in the Arduino IDE software. The sketch is saved with the extension '.ino.'

The Arduino IDE will appear as:


File name
IDE Version
Toolbar Button
Menu Bar
Text Editor for writing code
Shows the Uploading status
Error Messages
Configured board and serial port
Javatpoint | Arduino 1.8.12
File Edit Sketch Tools Help
Javatpoint

```
void setup() {
  // put your setup code here, to run once:

}

void loop() {
  // put your main code here, to run repeatedly:

}
```

1 Arduino Pro or Pro Mini, ATmega328P (5V, 16 MHz)

Fig 3.1: Arduino IDE Home Page

Let's discuss each section of the Arduino IDE display in detail.

Toolbar Button

The icons displayed on the toolbar are New, Open, Save, Upload, and Verify**.**

It is shown below:

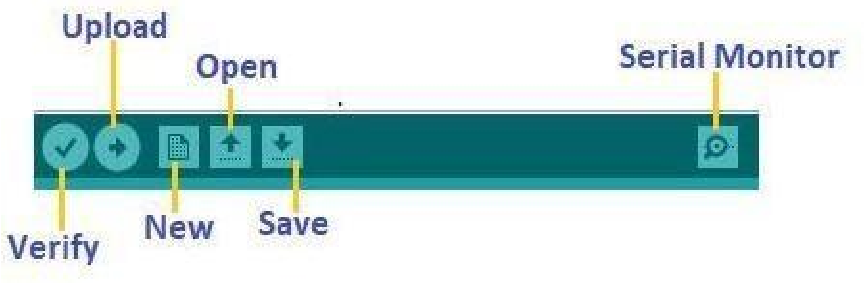


Fig 3.2: Toolbar

Upload

The Upload button compiles and runs our code written on the screen. It further uploads the code to the connected board. Before uploading the sketch, we need to make sure that the correct board and ports are selected.

We also need a USB connection to connect the board and the computer. Once all the above measures are done, click on the Upload button present on the toolbar.

The latest Arduino boards can be reset automatically before beginning with Upload. In the older boards, we need to press the Reset button present on it. As soon as the uploading is done successfully, we can notice the blink of the Tx and Rx LED.

If the uploading is failed, it will display the message in the error window.We do not require any additional hardware to upload our sketch using the Arduino Bootloader. A **Bootloader** is defined as a small program, which is loaded in the microcontroller present on the board. The LED will blink on PIN 13.

Open

The Open button is used to open the already created file. The selected file will be opened in the current window.

Save

The save button is used to save the current sketch or code.

New

It is used to create a new sketch or opens a new window.

Verify

The Verify button is used to check the compilation error of the sketch or the written code.

Serial Monitor

The serial monitor button is present on the right corner of the toolbar. It opens the serial monitor.

When we connect the serial monitor, the board will reset on the operating system Windows, Linux, and Mac OS X. If we want to process the control characters in our sketch, we need to use an external terminal program. The terminal program should be connected to the COM port, which will be assigned when we connect the board to the computer.

Menu Bar

File

When we click on the File button on the Menu bar, a drop-down list will appear.It is shown below:

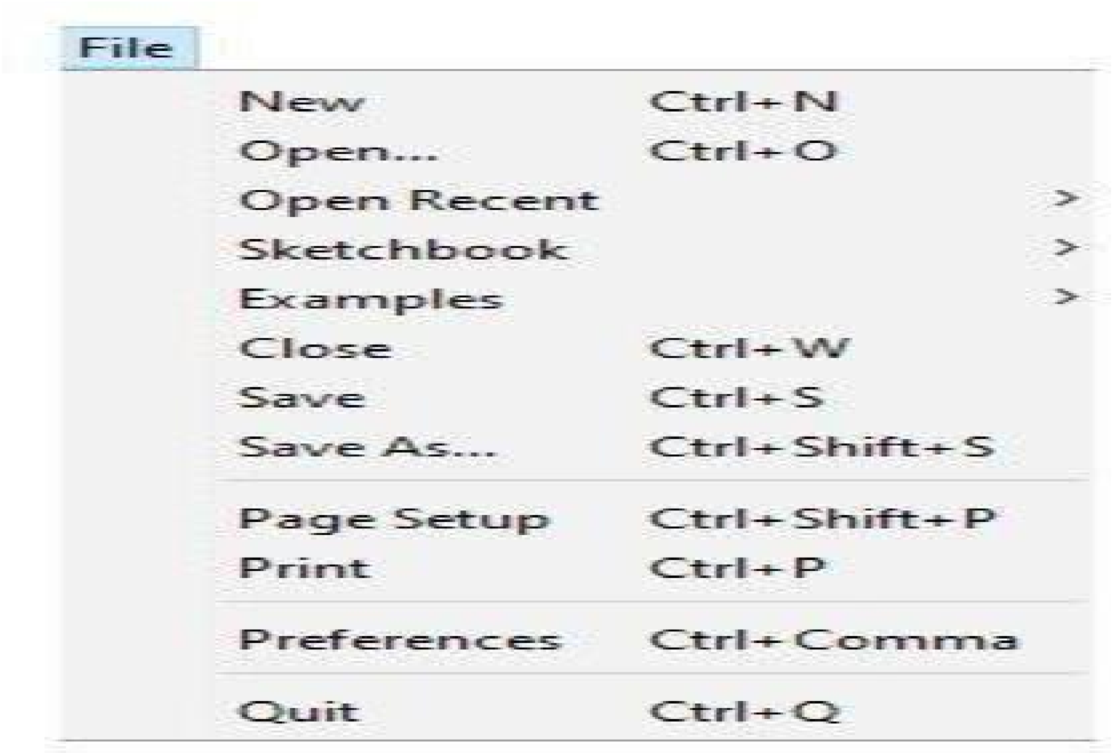


Fig 3.3: File Button

Let's discuss each option in detail.

New

The New button opens the new window. It does not remove the sketch which is already present.

Open

It allows opening the sketch, which can be browsed from the folders and computer drivers.

Open Recent

The Open Recent button contains the list of the recent sketches.

Sketchbook

It stores the current sketches created in the Arduino IDE software. It opens the

Close

The Close button closes the window from which the button is clicked.

Save

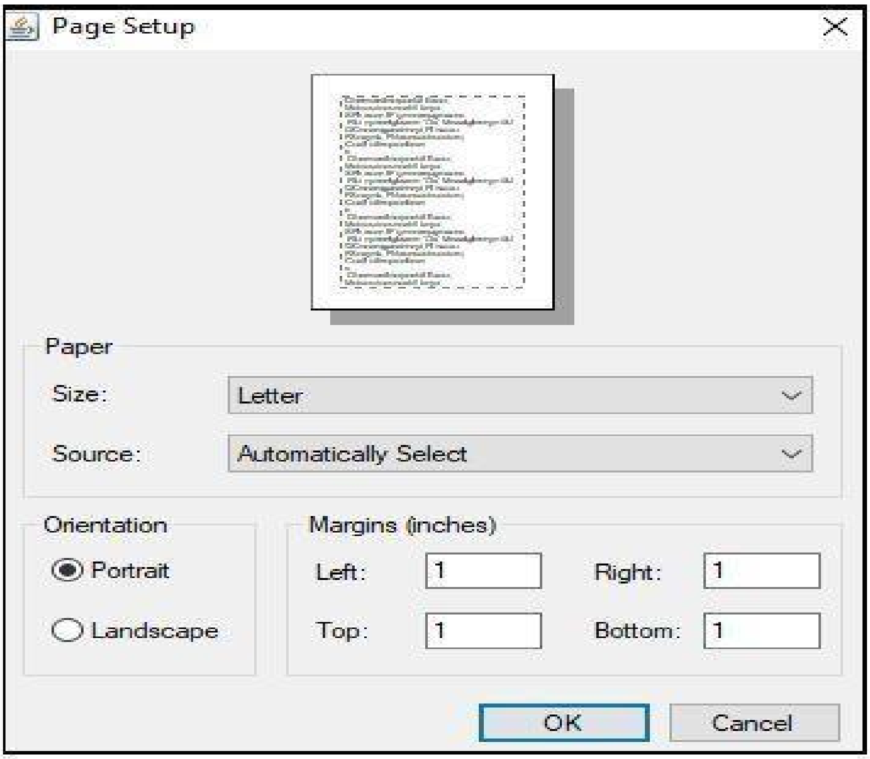
The save button is used to save the current sketch. It also saves the changes made to the current sketch. If we have not specified the name of the file, it will open the 'Save As...' window.

Save As...

We can save the sketch with a different name using the 'Save As...' button. We can also change the name accordingly.

Page Setup

It allows setting the page margins, orientation, and size for printing.



Page Setup

Paper

Size: Letter

Source: Automatically Select

Orientation

● Portrait

○ Landscape

Margins (inches)

Left: 1 Right: 1

Top: 1 Bottom: 1

Fig 3.4: Page Set Up

Print

According to the settings specified in the 'Page Setup', it prepares the current sketch for printing.

Preferences

It allows the customization settings of the Arduino IDE.

Quit

The Quit button is used to close all the IDE windows. The same closed sketch will be reopened when we will open the Arduino IDE.

Edit

When we click on the Edit button on the Menu bar, a drop-down list appears. It is shown below:

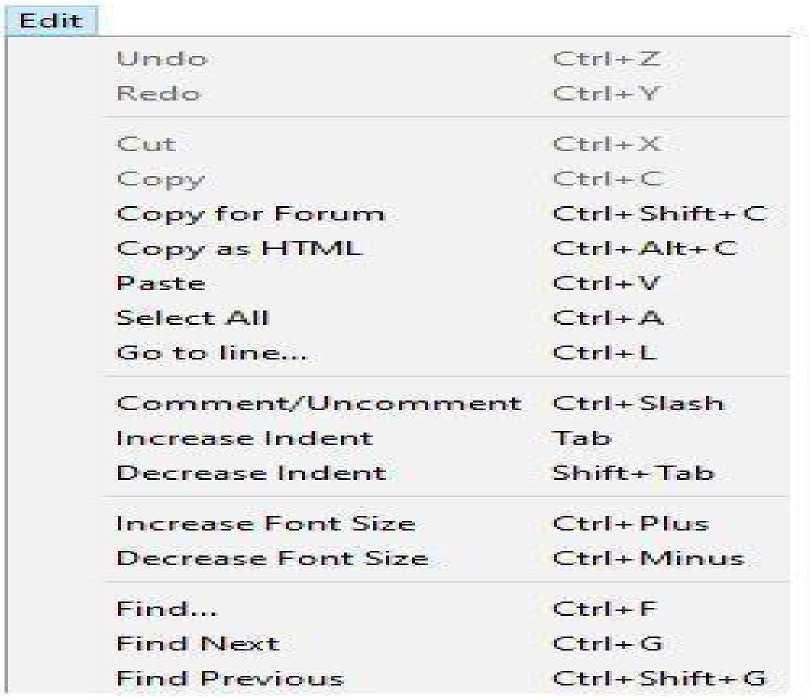


Fig 3.5: Edit Button

Let's discuss each option in detail.

Undo

The Undo button is used to reverse the last modification done to the sketch while editing.

Redo

The Redo button is used to repeat the last modification done to the sketch whileediting.

Cut

It allows us to remove the selected text from the written code. The text is further placed to the clipboard. We can also paste that text anywhere in our sketch.

Copy

It creates a duplicate copy of the selected text. The text is further placed on the clipboard. The 'Copy for Forum' button is used to copy the selected text to the clipboard,which is also suitable for posting to the forum.

Copy as HTML

The 'Copy for Forum' button is used to copy the selected text as HTML to the clipboard. It is desirable for embedding in web pages.

Paste

The Paste button is used to paste the selected text of the clipboard to the specified position of the cursor.

Select All

It selects all the text of the sketch.

Go to line

It moves the cursor to the specified line number.

The window will appear as:

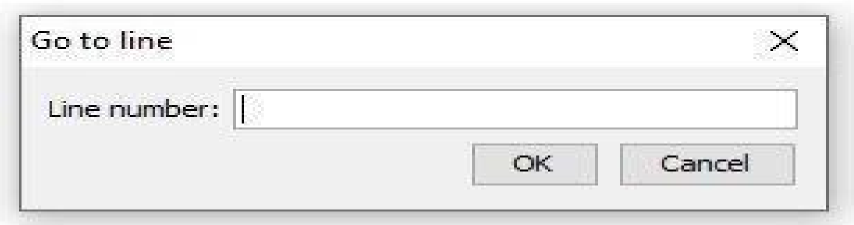


Fig 3.6: Go To Line

Comment / Decomment

The Comment/ Decomment button is used to put or remove the comment mark (//) at the beginning of the specified line.

Increase Indent

It is used to add the space at the starting of the specified line. The spacing moves the text towards the right.

Decrease Indent

It is used to subtract or remove the space at the starting of the specified line. The spacing moves the text towards the left.

Increase Font Size

It increases the font size of the written text.

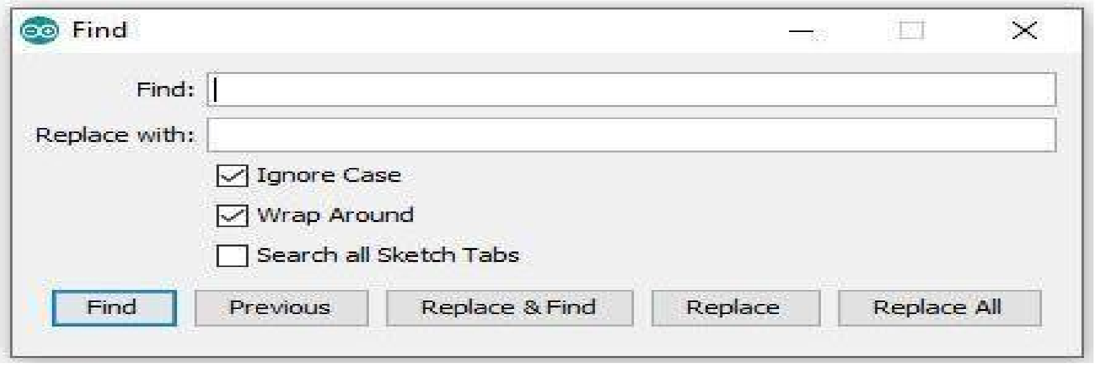
Decrease Font Size

It decreases the font size of the written text.

Find

It is used to find the specified text. We can also replace the text. It highlights the text in the sketch.

The window will appear as:



☑ Ignore Case
☑ Wrap Around
□ Search all Sketch Tabs

Fig 3.7: Find Button

Find Next

It highlights the next word, which has specified in the 'Find...' window. If there is no such word, it will not show any highlighted text.

Find Previous

It highlights the previous word, which has specified in the 'Find...' window. If there is no such word, it will not show any highlighted text.

Sketch

When we click on the Sketch button on the Menu bar, a drop-down list appears. It is shown below:

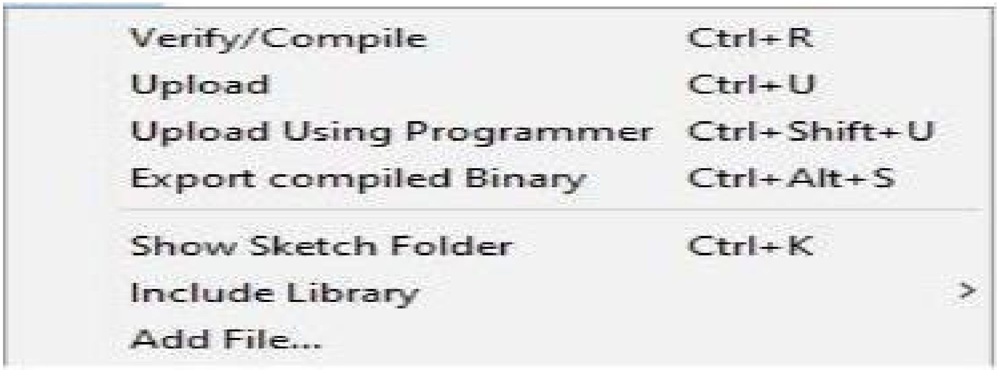

Verify/Compile Ctrl+R
Upload Ctrl+U
Upload Using Programmer Ctrl+Shift+U
Export compiled Binary Ctrl+Alt+S
Show Sketch Folder Ctrl+K
Include Library >
Add File...

Fig 3.8: Sketch Button.

Verify/Compile

It will check for the errors in the code while compiling. The memory in the console area is also reported by the IDE.

Upload

The Upload button is used to configure the code to the specified board through the port.

Upload Using Programmer

It is used to override the Bootloader that is present on the board. We can utilize the full capacity of the Flash memory using the '**Upload Using Programmer**' option. To implement this, we need to restore the Bootloader using the Tools-> Burn Bootloader option to upload it to the USB serial port.

Export compiled Binary

It allows saving a **.hex** file and can be kept archived. Using other tools, .hex file can also be sent to the board.

Show Sketch Folder

It opens the folder of the current code written or sketch.

Include Library

Include Library includes various Arduino libraries. The libraries are inserted into our code at the beginning of the code starting with the #. We can also import the libraries from .zip file.

Add File

The Add File... button is used to add the created file in a new tab on the existing file.For example, let's add 'Blink' file to the 'Javapoint' file. The tab will now appear as:

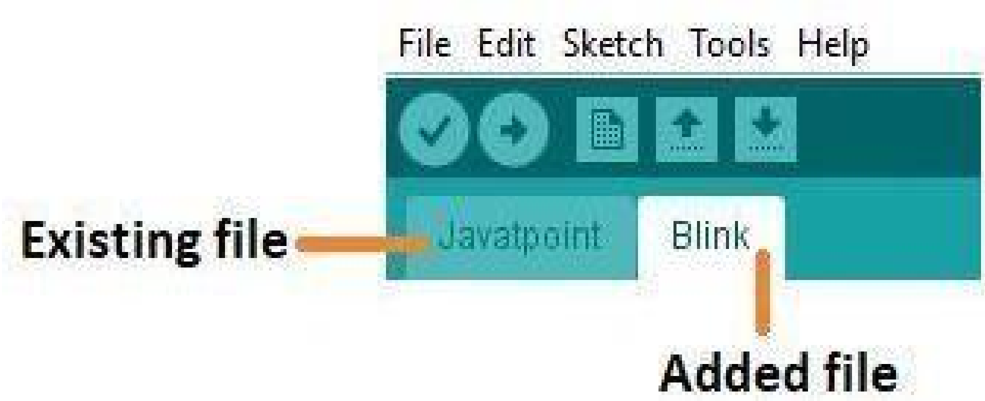
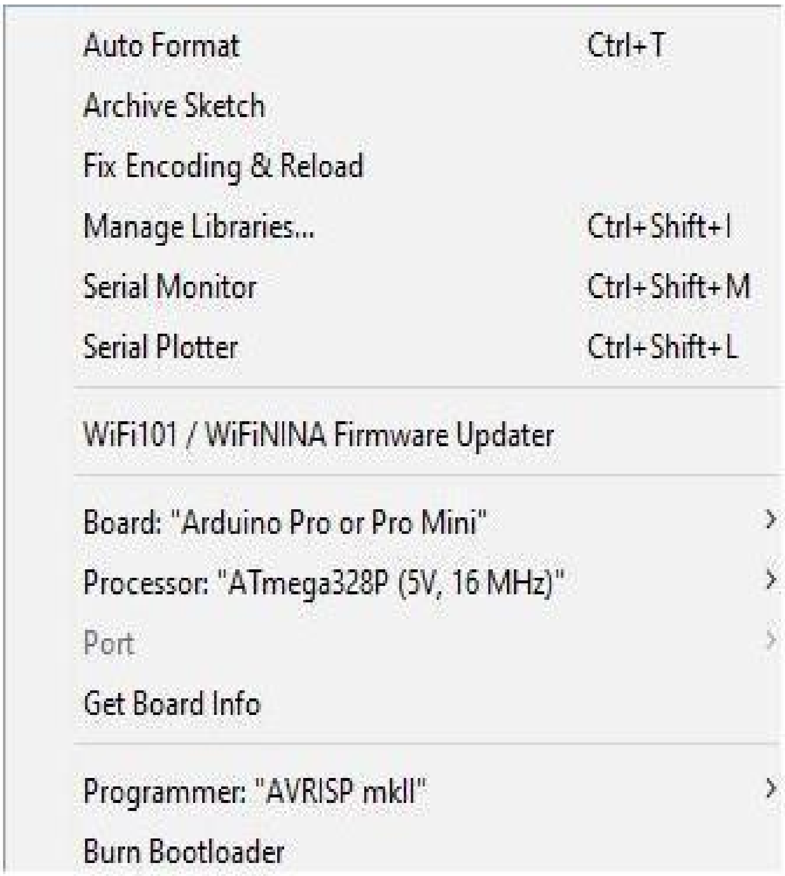


Fig 3.9: Add file Button

We can also delete the corresponding file from the tab by clicking on **the** small triangle -> Delete option.

Tools

When we click on the Tools button on the Menu bar, a drop-down list appears. It is shown below:



Board: "Arduino Pro or Pro Mini"

Processor: "ATmega328P (5V, 16 MHz)"

Programmer: "AVRISP mkII"

Fig 3.10: Tools Button

Let's discuss each option in detail.

Auto Format

The Auto Format button is used to format the written code. For example, lining the open and closed curly brackets in the code.

Archive Sketch

The copy of the current sketch or code is archived in the .zip format. The directory of the archived is same as the sketch.

Fix Encoding and Reload

This button is used to fix the inconsistency between the operating system char maps and editor char map encoding.

Manage Libraries

It shows the updated list of all the installed libraries. We can also use this option to install a new library into the Arduino IDE.

Serial Monitor

It allows the exchange of data with the connected board on the port.

Serial Plotter

The Serial Plotter button is used to display the serial data in a plot. It comes preinstalled in the Arduino IDE.

WiFi101/WiFiNINA Firmware Updater

It is used to check and update the Wi-Fi Firmware of the connected board.

Board

We are required to select the board from the list of boards. The selected board must be similar to the board connected to the computer.

Processor

It displays the processor according to the selected board. It refreshes every time during the selection of the board.

Port

It consists of the virtual and real serial devices present on our machine.

Get Board Info

It gives the information about the selected board. We need to select the appropriate port before getting information about the board.

Programmer

We need to select the hardware programmer while programming the board. It is required when we are not using the onboard USB serial connection. It is also required during the burning of the Boot loader.

Burn Boot loader

The Boot loader is present on the board onto the micro controller. The option is useful when we have purchased the micro controller without the boot loader. Before burning the boot loader, we need to make sure about the correct selected board and port.

Help

When we click on the Help button on the Menu bar, a drop-down list will appear. It is shown below:

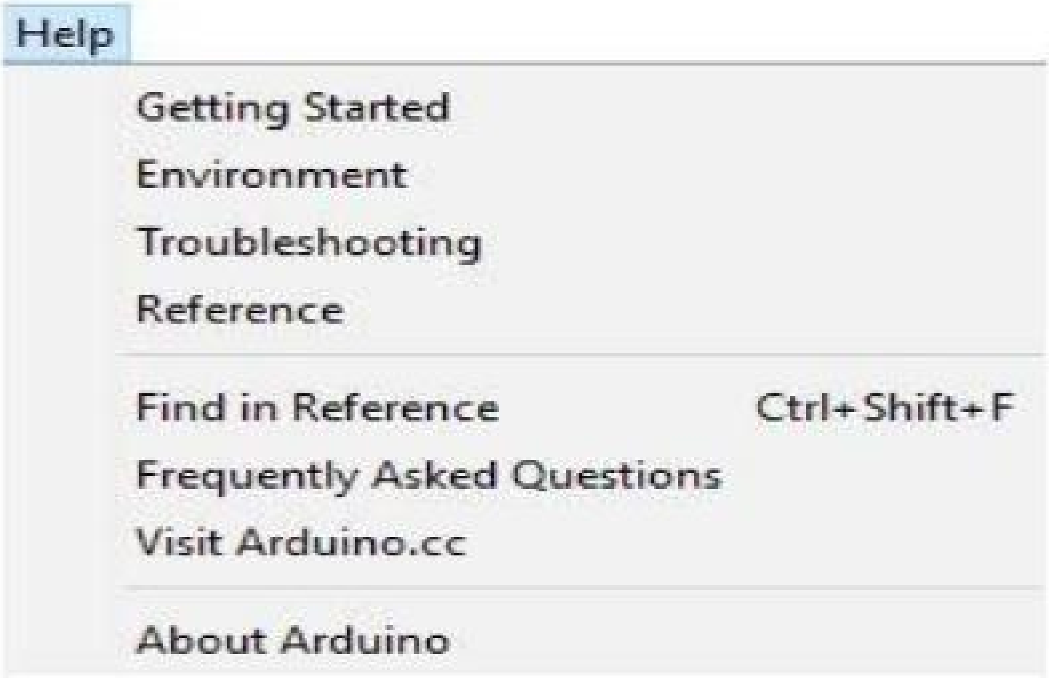


Fig 3.11: Help Button

The Help section includes several documents that are easy to access, which comes along with the Arduino IDE. It consists of the number of options such as Getting Started, Environment, Troubleshooting, Reference, etc. We can also consider the image shown above, which includes all the options under the Help section.

CHAPTER 4

HARDWARE REQUIREMENT SPECIFICATIONS

4.1 ARDUINO UNO BOARD

Arduino is an open-source electronics platform based on easy-to-use hardware and software. Arduino boards are able to read inputs - light on a sensor, a finger ona button, or a Twitter message - and turn it into an output - activating a motor, turning on an LED, publishing something online. You can tell your board what to do by sending a set of instructions to the microcontroller on the board. To do so you use the Arduino programming language (based on Wiring), and the Arduino Software(IDE), based on Processing.

Over the years Arduino has been the brain of thousands of projects, from every- day objects to complex scientific instruments. A worldwide community of makers - students, hobbyists, artists, programmers, and professionals - has gathered around this open-source platform, their contributions have added up to an incredible amountof accessible knowledge that can be of great help to novices and experts alike.

Arduino was born at the Ivrea Interaction Design Institute as an easy tool for fast prototyping, aimed at students without a background in electronics and program- ming. As soon as it reached a wider community, the Arduino board started changing to adapt to new needs and challenges, differentiating its offer from simple 8-bit boards to products for IoT applications, wearable, 3D printing, and embedded environments. All Arduino boards are completely open-source, empowering users to build them in- dependently and eventually adapt them to their particular needs.

Arduino is a computer hardware and software company, project, and user community that designs and manufactures micro controller kits for building digital devices and interactive objects that can sense and control objects in the physical world. The project’s products are distributed as open-source hardware and software, which are licensed under the GNU Lesser General Public License (LGPL) or the GNU General Public License (GPL),[1] permitting the manufacture of Arduino boards and software distribution by anyone. Arduino boards are available commercially in resembled form, or as do-it-yourself kits.

The project’s board designs use a variety of microprocessors and controllers. These systems provide sets of digital and analog input/output (I/O) pins that may be inter- faced to various expansion boards (“shields”) and other circuits. The boards feature serial communications interfaces, including Universal Serial Bus (USB) on some models, for loading programs from personal computers. The microcontrollers are mainly programmed using a dialect of features from the programming languages C and C++.In addition to using traditional compiler tool chains, the Arduino project pro- vides an integrated development environment (IDE) based on the Processing languageproject.

The Arduino project started in 2005 as a program for students at the Interaction Design Institute Ivrea in Ivrea, Italy,[2] aiming to provide a low-cost and easy way for novices and professionals to create devices that interact with their environmentusing sensors and actuators. Common examples of such devices intended for beginner hobbyists include simple robots, thermostats, and motion detectors.

Arduino/Genuino Uno is a microcontroller board based on the ATmega328P (datasheet). It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz quartz crystal, a USB connection, a power jack,an ICSP header and a reset button. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with aAC-to-DC adapter or battery to get started.You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

“Uno” means one in Italian and was chosen to mark the release of Arduino Software (IDE) 1.0. The Uno board and version 1.0 of Arduino Software (IDE) were the reference versions of Arduino, now evolved to newer releases.

4.2 WHY ARE WE USING ARDUINO

Thanks to its simple and accessible user experience, Arduino has been used in thou- sands of different projects and applications. The Arduino software is easy-to-use for beginners, yet flexible enough for advanced users. It runs on Mac, Windows, and Linux. Teachers and students use it to build low cost scientific instruments, to prove chemistry and physics principles, or to get started with programming and robotics. Designers and architects build interactive prototypes, musicians and artists use it for installations and to experiment with new musical instruments. Makers, of course, use it to build many of the projects exhibited at the Maker Fair, for example. Arduino is a key tool to learn new things.

Anyone children, hobbyists, artists, programmers can start tinkering just following the step by step instructions of a kit, or sharing ideas online with other members of the Arduino community.

There are many other micro controllers and micro controller platforms available for physical computing. Parallax Basic Stamp, Net media’s BX-24, Phidgets, MIT’s Handy board, and many others offer similar functionality. All of these tools take the messy details of micro controller programming and wrap it up in an easy-to-use package. Arduino also simplifies the process of working with micro controllers, but it offers some advantage for teachers, students, and interested amateurs over other systems

Inexpensive

Arduino boards are relatively inexpensive compared to other micro controller platforms. The least expensive version of the Arduino module can be assembled by hand, and even the pre-assembled Arduino modules cost less than 50 dollars

Cross-platform

The Arduino Software (IDE) runs on Windows, Macintosh OSX, and Linux operating systems. Most micro controller systems are limited to windows

Simple, clear programming software

The Arduino Software (IDE) is easy- to-use for beginners, yet flexible enough for advanced users to take advantage of as well. For teachers, it's conveniently based on the Processing programming environment, so students learning to program in that environment will be familiar with how the Arduino IDE works.

Open source and extensible software

The Arduino software is published as open source tools, available for extension by experienced programmers. The language can be expanded through C++ libraries, and people wanting to understand the technical details can make the leap from Arduino to the AVR C programming language on which it's based. Similarly, you can add AVR-C code directly into your Arduino programs if you want to the plans of the Arduino boards are published under a Creative Common license, so experienced circuit designers can make their own version of the module, extending it and improving it. Even relatively inexperienced users can build the breadboard version of the module in order to understand how it works and save money.

Table 4.1: Technical Specification

Micro controller	ATMega328P
Operating Voltage	5V
Input Voltage(recommended)	7-12V
Input Voltage(limit)	6-20V
Digital I/O Pins	14(of which 6 provide PWN output)
PWM Digital I/O Pins	6
Analog Input Pins	6
DC Current per I/O Pins	20 mA
Flash Memory	32 KB (ATmega328P) of which 0.5 KB used by boot-loader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED BUILTIN	13
Length	68.6 mm
Width	53.4 mm
Weight	25 g

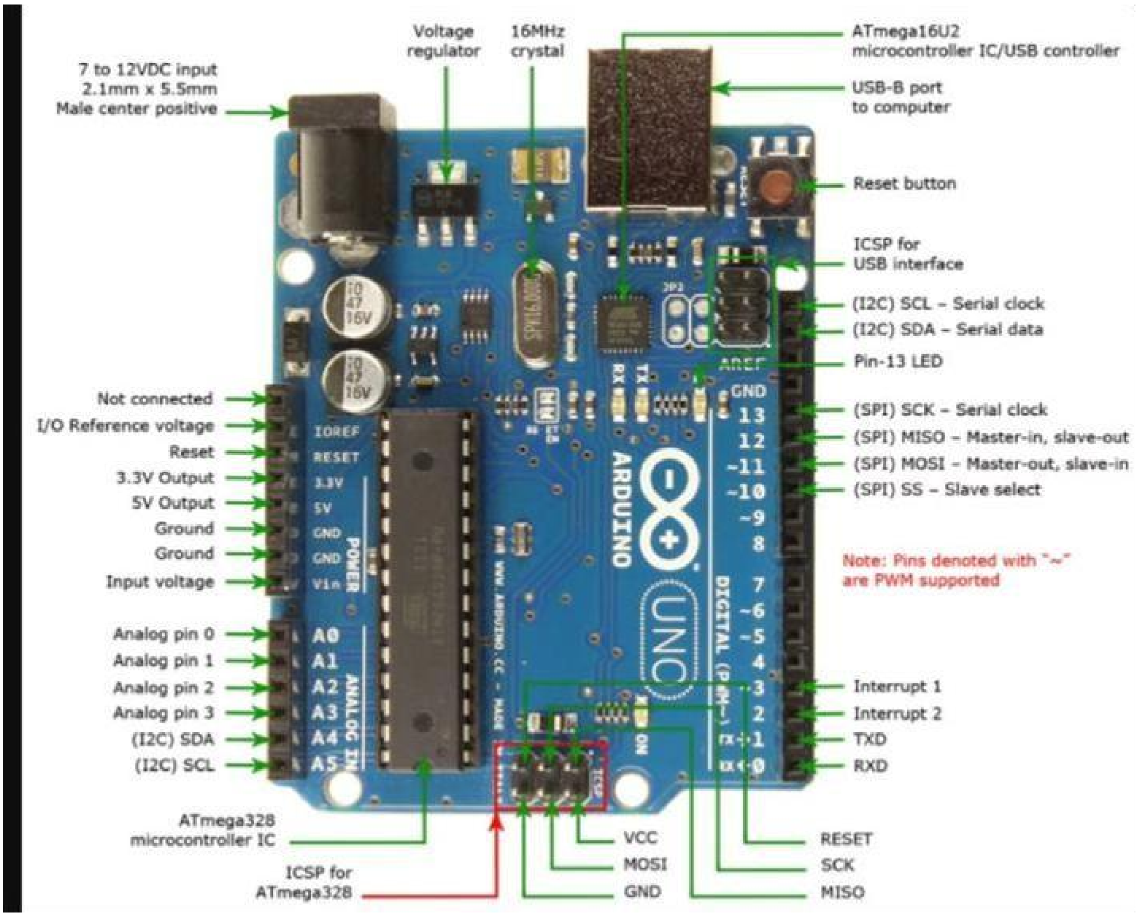


Fig 4.1: ATMega328P

4.3 BLUETOOTH MODULE HC-05

HC-05 module is an easy touse Bluetooth SPP (Serial Port Protocol) module, designed for transparent wireless serial connection setup. Serial port Bluetooth module is fully qualified Bluetooth V2.0+EDR (Enhanced Data Rate) 3Mbps Modulation with complete 2.4GHz radio transceiver and baseband. It uses CSR Blue core 04-External single chip Bluetooth system with CMOS technology and with AFH (Adaptive Frequency Hopping Feature). It has the footprint as small as 12.7mmx27mm.

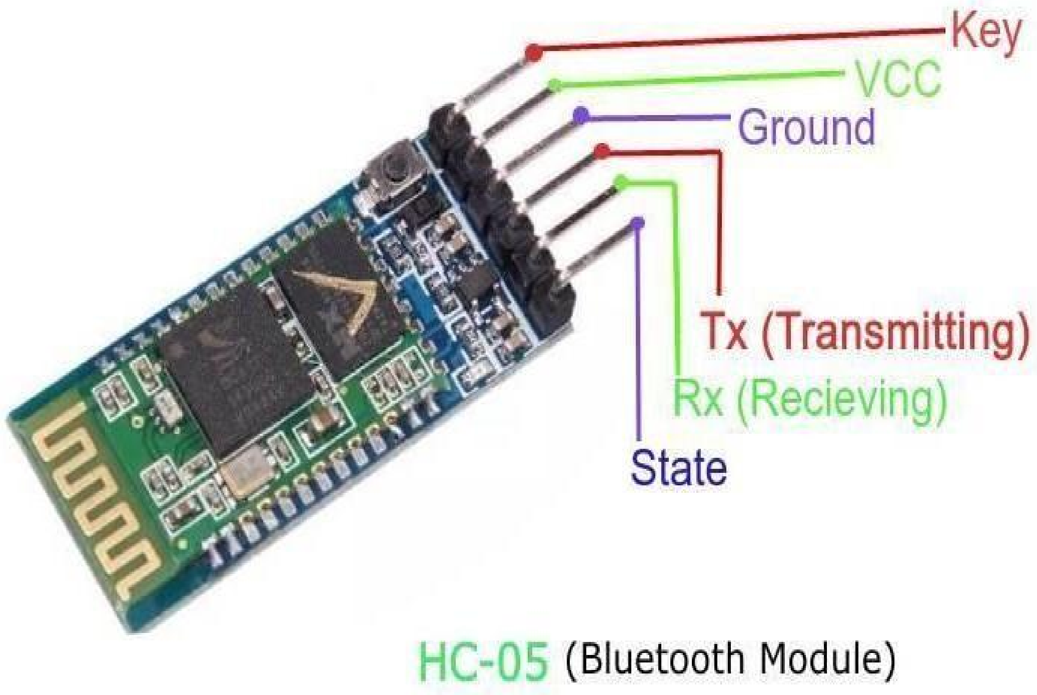


Fig 4.2: Bluetooth Module HC-05

Specifications

- Hardware features
- Typical -80dBm sensitivity
- Up to +4dBm RF transmit power
- Low Power 1.8V Operation ,1.8 to 3.6V I/O
- PIO control
- UART interface with programmable baud rate
- With integrated antenna
- With edge connector

Software features

- Default Baud rate: 38400, Data bits:8, Stop bit:1, Parity: No parity, Data control has Supported baudrate: 9600,19200,38400,57600,115200,230400,460800.
- Given a rising pulse in PIO0, device will be disconnected.
- Status instruction port PIO1: low-disconnected, high-connected;
- PIO10 and PIO11 can be connected to red and blue led separately. When master and slave are paired, red and blue led blinks 1time/2s in interval, while disconnected only blue led blinks 2times/s. Auto-connect to the last device on power as default.
- Permit pairing device to connect as default.
- Auto-pairing PINCODE:"0000" as default
- Auto-reconnect in 30 min when disconnected as a result of beyond the range of connection.

4.4 DHT11 SENSOR

DHT11 Temperature & Humidity Sensor features a temperature & humidity sensor complex with a calibrated digital signal output. By using the exclusive digital-signal-acquisition technique and temperature & humidity sensing technology, it ensures high reliability and excellent long-term stability.

This sensor includes a resistive-type humidity measurement component and an NTC temperature measurement component, and connects to a high performance 8-bit micro controller, offering excellent quality, fast response, anti-interference ability and costeffectiveness.

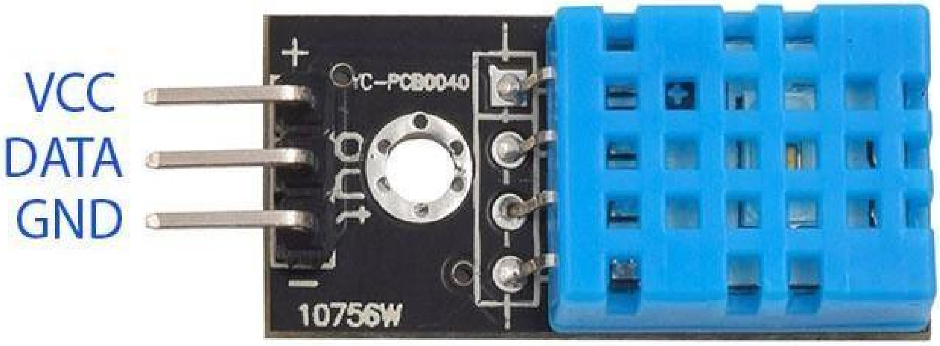
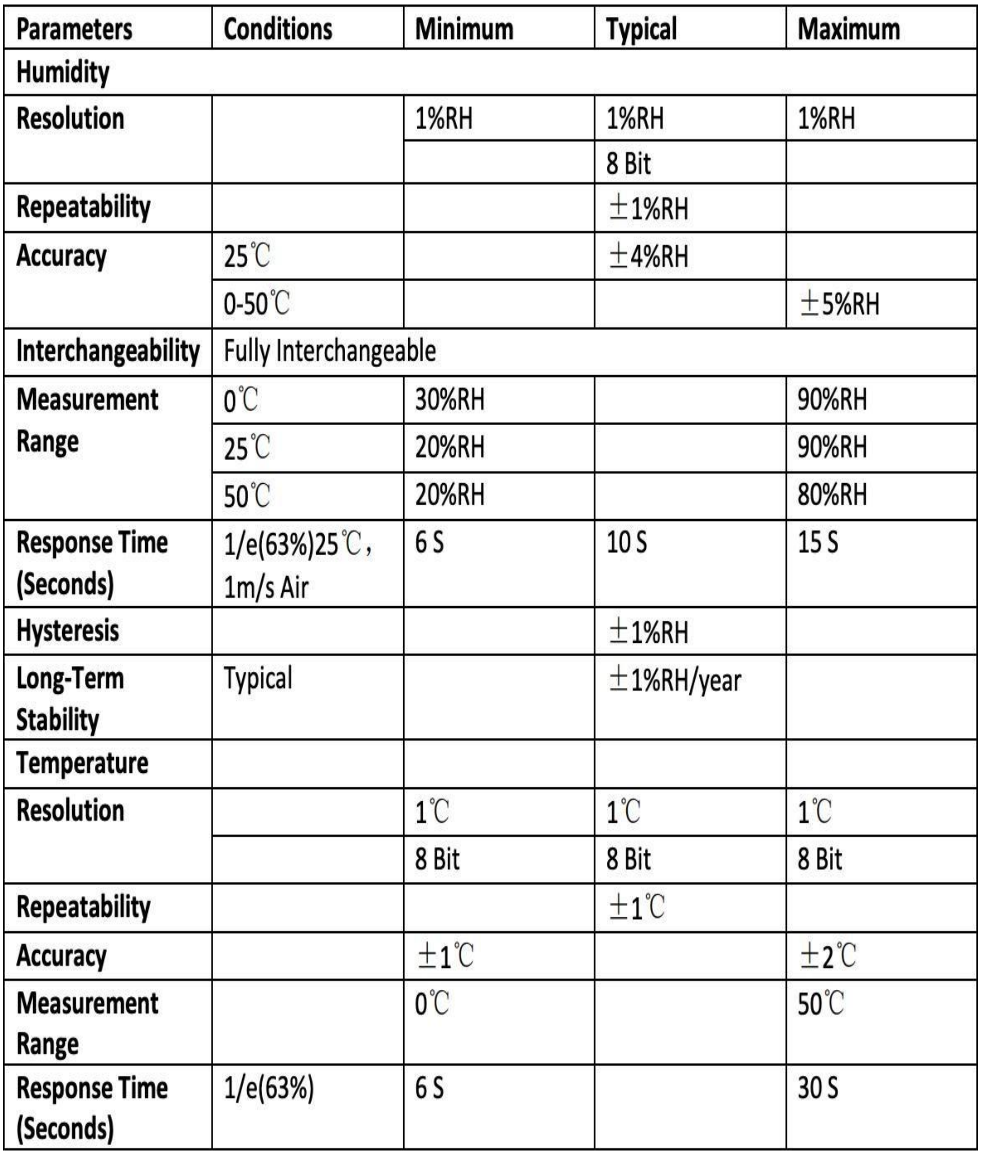


Fig 4.3: DHT 11 Sensor

Each DHT11 element is strictly calibrated in the laboratory that is extremely accurate on humidity calibration. The calibration coefficients are stored as programs in the OTP memory, which are used by the sensor's internal signal detecting process. The single-wire serial interface makes system integration quick and easy. Its small size, low power consumption and up-to-20meter signal transmission making it the best choice for various applications, including those most demanding ones. The component is 4-pin single row pin package. It is convenient to connect and special packages can be provided according to users' request.

The DHT11 provides digital output for both temperature and humidity readings, making it simple to integrate into projects without the need for analog-to-digital conversion. However, it has limitations in terms of accuracy and response time compared to more advanced sensors like the DHT22 or SHT series.

Table 4.2: Specification of Arduino



Parameters	Conditions	Minimum	Typical	Maximum
Humidity				
Resolution		1%RH	1%RH	1%RH
			8 Bit	
Repeatability			±1%RH	
Accuracy	25℃		±4%RH	
	0-50℃			±5%RH
Interchangeability	Fully Interchangeable			
Measurement Range	0℃	30%RH		90%RH
	25℃	20%RH		90%RH
	50℃	20%RH		80%RH
Response Time (Seconds)	1/e(63%)25℃，1m/s Air	6 S	10 S	15 S
Hysteresis			±1%RH	
Long-Term Stability	Typical		±1%RH/year	
Temperature				
Resolution		1℃	1℃	1℃
		8 Bit	8 Bit	8 Bit
Repeatability			±1℃	
Accuracy		±1℃		±2℃
Measurement Range		0℃		50℃
Response Time (Seconds)	1/e(63%)	6 S		30 S

4.5 MQ 2 SENSOR

Sensitive material of MQ-2 gas sensor is SnO2, which with lower conductivity in clean air. When the target combustible gas exist, The sensor’s conductivity is more higher along with the gas concentration rising. Please use simple electro circuit, Convert change of conductivity to correspond output signal of gas concentration.

MQ-2 gas sensor has high sensitivity to LPG, Propane and Hydrogen, also could be used to Methane and other combustible steam, it is with low cost and suitable for different application.

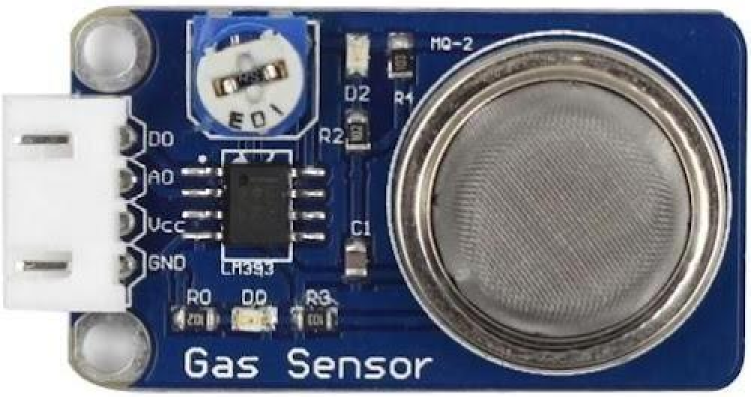


Fig 4.4: MQ 2 Sensor

Character

* Good sensitivity to Combustible gas in wide range.

* High sensitivity to LPG, Propane and Hydrogen.

* Long life and low cost.

* Simple drive circuit.

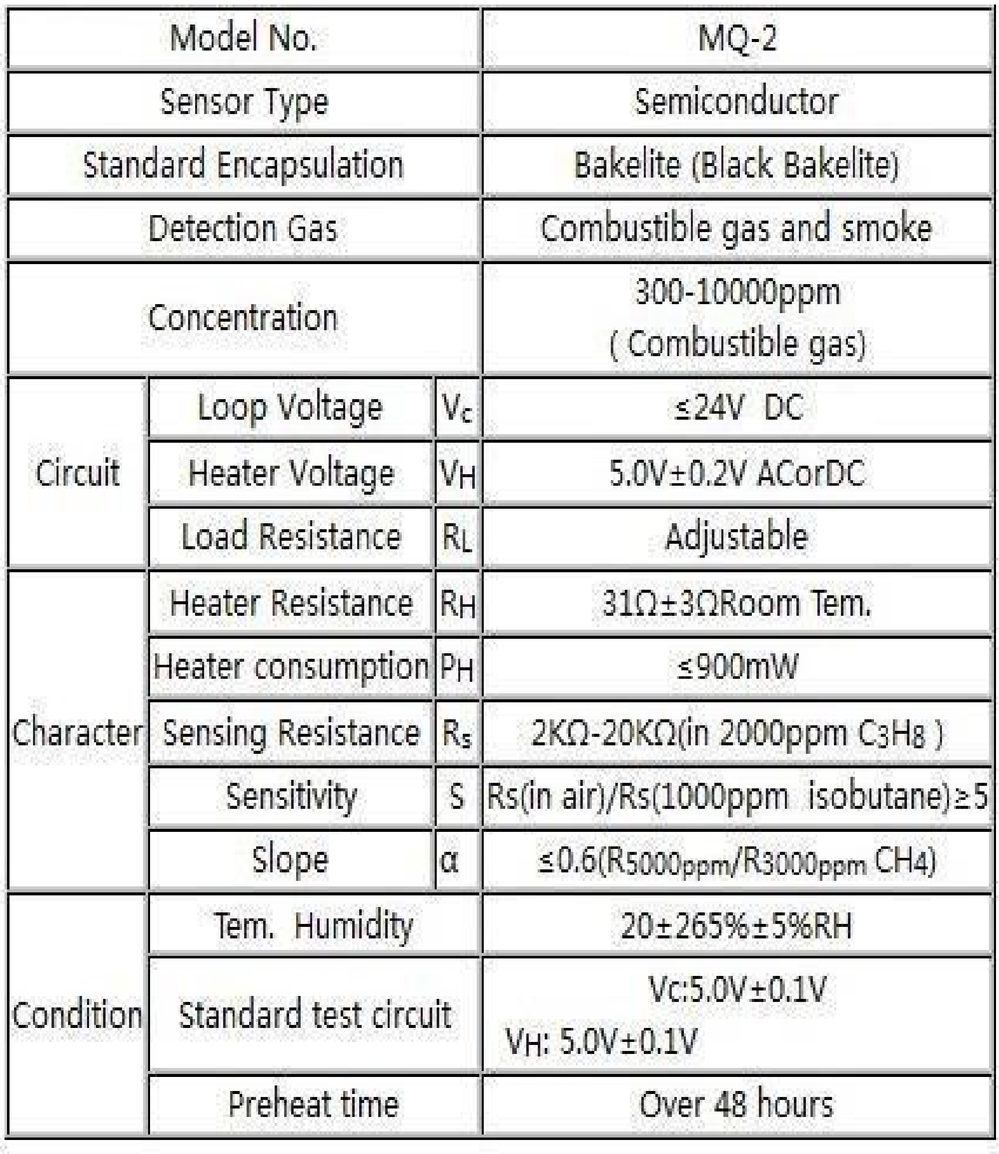
Application

* Domestic gas leakage detector.

* Industrial Combustible gas detector.

* Portable gas detector.

Table 4.3: Technical Data



Model No.			MQ-2
Sensor Type			Semiconductor
Standard Encapsulation			Bakelite (Black Bakelite)
Detection Gas			Combustible gas and smoke
Concentration			300-10000ppm (Combustible gas)
Circuit	Loop Voltage	V_c	≤24V DC
	Heater Voltage	V_H	5.0V±0.2V ACorDC
	Load Resistance	R_L	Adjustable
Character	Heater Resistance	R_H	31Ω±3ΩRoom Tem.
	Heater consumption	P_H	≤900mW
	Sensing Resistance	R_s	2KΩ-20KΩ(in 2000ppm C_3H_8)
	Sensitivity	S	Rs(in air)/Rs(1000ppm isobutane)≥5
	Slope	α	≤0.6($R_{5000ppm}/R_{3000ppm}$ CH_4)
Condition	Tem. Humidity		20±265%±5%RH
	Standard test circuit		V_c:5.0V±0.1V V_H: 5.0V±0.1V
	Preheat time		Over 48 hours

4.6 GSM SIM 800 L

SIM Com offers this information as a service to its customers, to support application and engineering efforts that use the products designed by SIM Com. The information provided is based upon requirements specifically provided to SIM Com by the customers.

SIM Com has not undertaken any independent search for additional relevant information, including any information that may be in the customer's possession.

Furthermore, system validation of this product designed by SIM Com within a larger electronic system remains the responsibility of the customer or the customer's system integrator. All specifications supplied herein are subject to change.

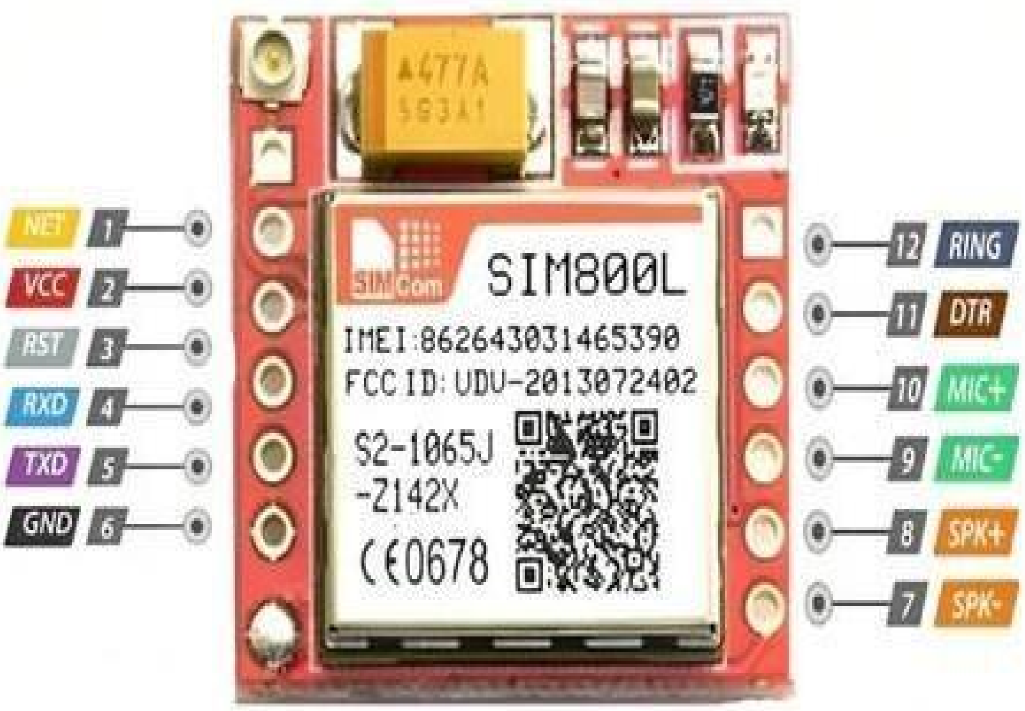


Fig 4.5: GSM SIM 800 L

SIM800H&SIM800L is a quad-band GSM/GPRS module, that works on frequencies GSM850MHz, EGSM900MHz, DCS1800MHz and PCS1900MHz. SIM800H&SIM800L features GPRS multi-slot class 12/ class 10 (optional) and supports the GPRS coding schemes CS-1, CS-2, CS-3 and CS-4. With a tiny configuration of 17.8*15.8*2.4mm, SIM800H&SIM800L can meet almost all the space requirements in customer applications, such as smart phone, PDA and other mobile devices.

Table 4.4 : Specifications of GSM

Operating frequency	GSM 850MHz, EGSM 900MHz, DCS 1800MHz and PCS 1900MHz
Operating Voltage rating	3.2V - 4.8V dc
Output pin voltage	5V dc
Output pin current	25mA
Communication mode	UART interface, configured for full-duplex asynchronous mode.
Baud rate	Supports auto bauding,9.6kb/s used.

SIM800H&SIM800L is a LGA package with 88 pads, and provides all hardware interfaces between the module and customers' boards.

- Support 5*5*2 keypads
- One full modem serial port, customer can configure two serial ports
- One USB, the USB interfaces can debug, download software
- Audio channel which includes two microphone input; a receiver output and a speaker output
- Programmable general purpose input and output.
- A SIM card interface
- Support Blue tooth(only SIM800H)
- Support FM
- Support one PWM SIM800H&SIM800L is designed with power saving technique so that the current consumption is as low as 1.04mA in sleep mode.

4.7 12V DC MOTOR

This miniature gearbox is of steel and brass construction with brass gears and is mounted on a 1mm thickness steel bracket. It incorporates high quality three pole motor with sleeved bearings. The design and construction of the unit make it suitable for a host of model and light industrial applications

*

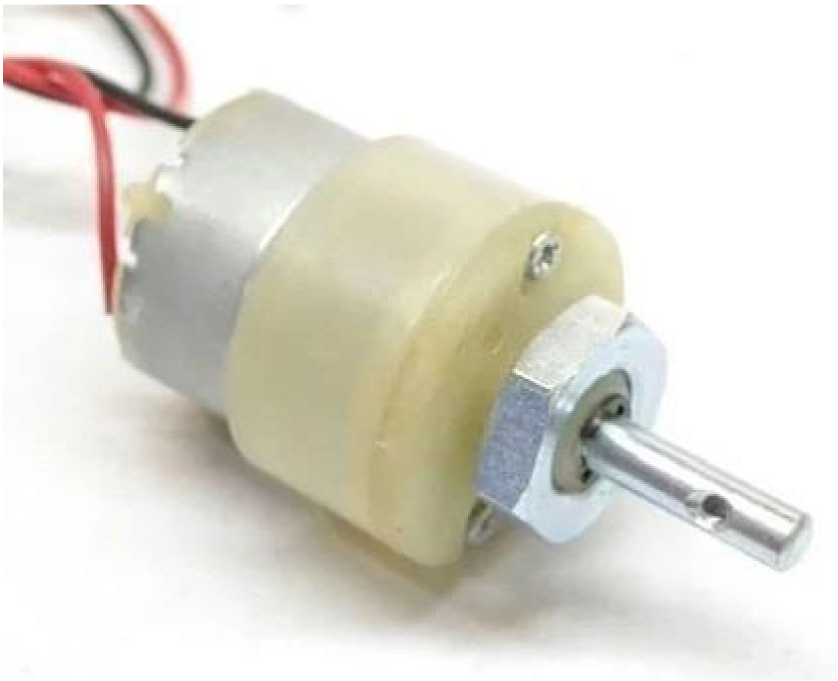


Figure 4.6: 12V DC MOTOR

When kept in a magnetic field, a current-carrying conductor gains torque and develops a tendency to move.In short, when electric fields and magnetic fields interact, a mechanical force arises. This is the principle on which the DC motors work.

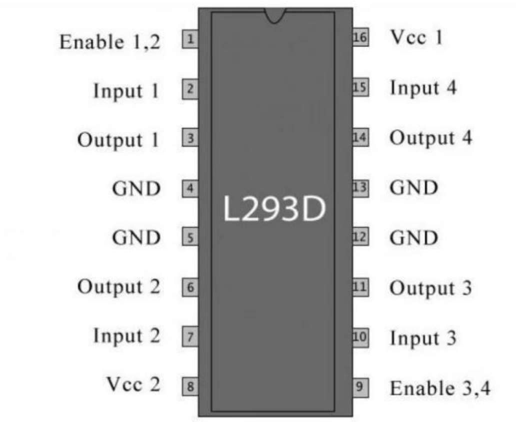
Table 4.5 : Specifications of 12V DC motor

Descriptions	Specification
Voltage	12VDC
Torque	1200mN.m
Speed	24 RPM
Current	0.9 A
Gear ratio	264:1
Encoder Output	1848 pulses per rotation, single channel output

4.8 L293D MOTOR DRIVER

The L293 and L293D devices are quadruple high current half-H drivers. The L293 is designed to provide bidirectional drive currents of up to 1 A at voltages from 4.5 V to 36 V. The L293D is designed to provide bidirectional drive currents of up to 600-mA at voltages from 4.5 V to 36 V. Both devices are designed to drive inductive loads such as relays, solenoids, DC and bipolar stepping motors, as well as other high-current/high-voltage loads in positive supply applications.

Fig 4.7: L293D Motor Driver



Each output is a complete totem-pole drive circuit ,with a Darling ton transistor sink and a pseudo Darling ton source. Drivers are enabled in pairs, with drivers 1 and 2 enabled by 1,2EN and drivers 3 and 4 enabled by 3,4EN.

4.9 REVOLVONG WHEEL

Incorporating a revolving wheel into a robotic vehicle's design serves multifaceted purposes, enhancing its mobility, adaptability, and performance. By enabling omnidirectional movement, the vehicle gains unparalleled in tight spaces or crowded environments without the need to adjust its orientation. Additionally, specialized treads or surfaces on the revolving wheels improve traction, allowing the vehicle to navigate challenging terrains such as rough terrain or gravel with greater ease. Furthermore, the rotating motion of the wheels contributes to the vehicle's stability and balance, particularly on uneven surfaces or when traversing obstacles, minimizing the risk of tipping over. Moreover, efficient energy utilization is facilitated through optimized motion efficiency, potentially reducing energy consumption and enabling features like regenerative braking.

Revolving wheel: Specification

- Diameter:40mm.
- Size:1.5Inch.
- Width: 20mm.

Four similar wheels are used in this robotic vehicle.

Fig :4.8 Revolving Wheel

CHAPTER 5

IMPLEMENTATION

Implementation is a stage of the project where the theoretical design is turned into a working system. This phase involves the actual materialization of the ideas, which are expressed in the analysis document and developed in the design phase. Implementation should be perfect mapping of the design document in a suitable programming language in order to achieve the necessary final product. If the implementation is not carefully planned and controlled, it can cause chaos and confusion.

The implementation stage requires the following tasks:

- Careful Planning.
- Investigation of system and constraints.
- Design of methods to achieve the changeover.
- Evaluation of the changeover method.
- Correct decisions regarding selection of the platform.
- Appropriate selection of the language for application development.

5.1 WORKING PRINCIPLE

Bluetooth is used for moving robot in four directions and to open the medicine and food box at the patients bed. The DHT sensor and MQ2 sensor will sense if there is any abnormal condition in ICU. If the sensors give data then Arduino will send the alertingSMS to doctor regarding the Humidity, Temperature and Smoke present in the ICU.

CHAPTER 6

RESULTS

Testing is an important phase in the development life cycle of the product this was the phase where the error remaining from all the phases was detected. Hence testing performs a very critical role for quality assurance and ensuring the reliability of the software. Once the implementation is done, a test plan should be developed and run on a given set of test data. Each test has a different purpose, all work to verify that all the system elements have been properly integrated and perform allocated functions. The testing process is actually carried out to make sure that the product exactly does the same thing what is suppose to do. Testing is the final verification and validation activity within the organization itself. In the testing stage following goals are tried to achieve

- To affirm the quality of the project
- To find and eliminate any residual errors from previous stages.
- To validate the software as the solution to the original problem.
- To provide operational reliability of the system.

During testing the major activities are concentrated on the examination and modification of the source code.

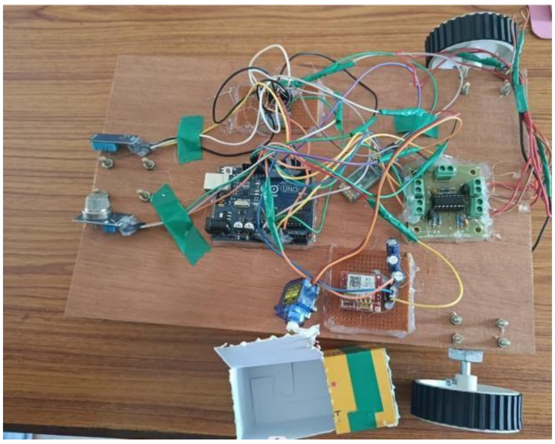


Fig 6.1: Prototype

CHAPTER 7

CONCLUSION AND FUTURE SCOPE

7.1 CONCLUSION

Third wave of Covid-19 is started in the world and in this wave corona cases rise around five times faster than the second wave. World will see some more waves in the coming years. We can’t say clearly at this stage when this pandemic will end. In such situations sometimes the healthcare system works under stress. So automation is very much helpful in this pandemic. For this purpose one simple low cost robotic vehicle is designed and presented here. Using this robotic vehicle we can sanitize the ward or part of the ward of the hospital. We can also mount a camera on the vehicle for surveillance purposes. We can do many types of modification as per requirement.

7.2 FUTURE SCOPE

Future work, image capturing of patients can be employed for better treatment The future of robotic vehicles in isolation wards holds great promise for advancing patient care and healthcare delivery. By focusing on improving navigation, enhancing interaction capabilities, integrating advanced sensing technologies, and incorporating tele medicine features, these robotic vehicles can play a crucial role in providing efficient and effective medical assistance. Moreover, advancements in autonomous task execution, safety, and reliability will ensure that these robots operate seamlessly in dynamic healthcare environments while minimizing risks to patients and staff. Additionally, efforts to make these solutions more scalable and cost-effective will enable their deployment in a wide range of healthcare settings, from large hospitals to smaller clinics and temporary facilities. Through interdisciplinary collaboration and innovation, robotic vehicles in isolation wards can significantly contribute to mitigating the spread of infectious diseases, enhancing patient outcomes, and revolutionizing healthcare delivery in the future

CHAPTER 8

REFERENCES

1. A. J. Häusler, A. Saccon, A. P. Aguiar, J. Hauser and A. M. Pascoal, "Energy-OptimalMotion
Planning for Multiple Robotic Vehicles With Collision Avoidance," in IEEE Transactions onControl Systems Technology, vol. 24, no. 3, pp. 867-883, May 2016

1.A. Sharma, P. Zanotti and L. P. Musunur, "Enabling the Electric Future of Mobility: Robotic Automation for Electric Vehicle Battery Assembly," in IEEE Access, vol. 7, pp. 170961-
170991, 2019

3.D. Kuan, G. Phipps and A.C. Hsueh, "Autonomous robotic vehicle road following," in IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 10, no. 5, pp. 648-658, Sept.
1988

CHAPTER 9

APPENDIX CODE

9.1 APPENDIX CODE

```
#include <DHT.h>
#include <DHT_U.h>
#include <servo.h>
#include<SoftwareSerial.h>
servo myservo;
SoftwareSerial Serial1(2,3); //make RX arduino line is pin 2, make TX arduino line is
pin 3.
SoftwareSerial Serial2(7,8); //bluetooth. make RX arduino line is pin 2, make TX
arduino line is pin 3.
//#include <DHT.h>
int pos = 0;
#define DHTPIN 4 // what pin we're connected to
#define DHTTYPE DHT11 // DHT 11 (AM2302)
DHT dht(DHTPIN, DHTTYPE);
int sen = 5;
int v1 = 6;
int rm1 = 9;
int rm2 = 10;
int lm1 = 11;
int lm2 = 12;
int ob;
char t;
int chk;
```

```
float hum; //Stores humidity value
float temp; //Stores temperature value
  void setup ()
{
 Serial1.begin(9600);
 Serial2.begin(9600);
myservo.attach(13);
Serial.begin(9600);
pinMode(v1, OUTPUT);
pinMode(rm1, OUTPUT); //left motors forward
pinMode(rm2, OUTPUT); //left motors reverse
pinMode(lm1, OUTPUT);  //right motors forward
pinMode(lm2, OUTPUT);  //right motors reverse
   pinMode(sen, INPUT);
   dht.begin();
Serial.println("ICU Monitoring");
}
    void loop ()
{
  digitalWrite(v1, LOW);

   hum = dht.readHumidity();
   temp= dht.readTemperature();
   Serial.print("Humidity: ");
   Serial.print(hum);
   Serial.print(" %, Temp: ");
   Serial.print(temp);
Serial.println(" Celsius");
    ob = analogRead(A0);
    Serial.print("Smoke: ");
    Serial.print(ob);
        Delay (2000); //Delay 2 sec.
    If(Serial2.available()){
  t = Serial2.read();
  Serial.println(t);
}
    if(t == '1'){            //move forward(all motors rotate in forward direction)
  digitalWrite(rm1, HIGH);
  digitalWrite(rm2,LOW);
  digitalWrite(lm1,HIGH);
  digitalWrite(lm2,LOW);
}
 else if(t == '2'){ //move reverse (all motors rotate in reverse direction)
  digitalWrite(rm1,LOW);
  digitalWrite(rm2,HIGH);
  digitalWrite(lm1,LOW);
  digitalWrite(lm2,HIGH);
}
  else if(t == '3'){      //turn right (left side motors rotate in forward direction, right side
doesn't rotate)
  digitalWrite(rm1,LOW);
  digitalWrite(rm2,HIGH);
  digitalWrite(lm1,HIGH);
  digitalWrite(lm2,LOW);
}
```

```
else if(t == '4'){ //turn left (right side motors rotate in forward direction, left side motors doesn't rotate)
  digitalWrite(rm1,HIGH);
  digitalWrite(rm2,LOW);
  digitalWrite(lm1,LOW);
  digitalWrite(lm2,HIGH);
}
 else if(t == '5')
{      //STOP (all motors stop)
digitalWrite(rm1,LOW);
digitalWrite(rm2,LOW);
digitalWrite(lm1,LOW);
digitalWrite(lm2,LOW);
}
else if(t == '6')
{
    myservo.write(180);
}
else if(t == '7')
{
myservo.write(-180);
}
delay(100);
if (ob>500) // pls change smoke value here
//if (ob == LOW)
{
  Serial.println("fire occuring");
  Serial.println("Sending SMS");
  Serial1.println("AT"); delay(500);
  Serial1.println("AT+CMGF=1");
      delay(500);
      Serial1.print("AT+CMGS=");
      Serial1.print("");
      Serial1.print("8179134842");     //mobile no. for SMS alert
      Serial1.println("");
  delay(500);
      Serial1.print("Fire Occuring");
  delay(500);
      Serial1.write(26);
      delay(500);
   Serial.println("SMS Sent");
   digitalWrite(v1, HIGH);
   delay(500);
Serial.println("fire occuring");
Serial.println("Sending SMS");
Serial1.println("AT"); delay(500);
Serial1.println("AT+CMGF=1");
delay(500);
Serial1.print("AT+CMGS=");
```

```
Serial1.print("");
Serial1.print("9381682379");     //mobile no. for SMS alert
 Serial1.println("");
delay(500);
Serial1.print("Fire Occuring");
delay(500);
Serial1.write(26);
delay(500);
Serial.println("SMS Sent");
digitalWrite(v1, HIGH);
delay(500);
}
if (hum>75) // pls change humidity threshold here
{
 Serial.println("High humidity");
 Serial.println("Sending SMS");
 Serial1.println("AT"); delay(500);
 Serial1.println("AT+CMGF=1");
 delay(500);
 Serial1.print("AT+CMGS=");
 Serial1.print("");
  Serial1.print("8179134842");     //mobile no. for SMS alert
  Serial1.println("");
   delay(500);
   Serial1.print("very high Humidity is observed");
    delay(500);
    Serial1.write(26);
    delay(500);
    Serial.println("SMS Sent");
    digitalWrite(v1, HIGH);
    delay(500);
    Serial.println("High humidity");
    Serial.println("Sending SMS");
    Serial1.println("AT"); delay(500);
    Serial1.println("AT+CMGF=1");
    delay(500);
        Serial1.print("AT+CMGS=");
        Serial1.print("");
        Serial1.print("9381682379");     //mobile no. for SMS alert
        Serial1.println("");
       delay(500);
        Serial1.print("very high Humidity is observed");
    delay(500);
    Serial1.write(26);
    delay(500);
    Serial.println("SMS Sent")
    digitalWrite(v1, HIGH);
    delay(500);
    }
     if (temp>30) // pls change temperature threshold here
    {
     Serial.println("High temperature");
     Serial.println("Sending SMS");
Serial1.println("AT");
     delay(500);
   Serial1.println("AT+CMGF=1");
   delay(500);
   Serial1.print("AT+CMGS=");
```

```
Serial1.print("");
Serial1.print("8179134842");     //mobile no. for SMS alert
Serial1.println("");
delay(500);
Serial1.print("very high temperature is observed");
delay(500);
Serial1.write(26);
 delay(500);
 Serial.println("SMS Sent");
 digitalWrite(v1, HIGH);
 delay(500);
 Serial.println("High temperature");
 Serial.println("Sending SMS");
 Serial1.println("AT");
 delay(500);
 Serial1.println("AT+CMGF=1");
 delay(500);
 Serial1.print("AT+CMGS=");
 Serial1.print("");
 Serial1.print("9381682379");     //mobile no. for SMS alert
 Serial1.println("");
  delay(500);
  Serial1.print("very high temperature is observed");
  delay(500);
  Serial1.write(26);
  delay(500);
  Serial.println("SMS Sent");
  digitalWrite(v1, HIGH);
  delay(500);
    }
    }
```