TABLE OF CONTENTS

LIST OF FIGURES

CHAPTER-1

INTRODUCTION

Robotics is a branch of engineering and science that includes electronics engineering, mechanical engineering and computer science and so on. This branch deals with the design, construction, use to control robots, sensory feedback and information processing. These are some technologies which will replace humans and human activities in coming years. These robots are designed to be used for any purpose but these are using in sensitive environments like bomb detection, deactivation of various bombs etc.

Robotics is that branch of engineering that deals with conception, design, operation, and manufacturing of robots. There was an author named Issac Asimov, he said that he was the first person to give robotics name in a short story composed in 1940's. In that story, Issac suggested three principles about how to guide these types of robotic machines. Later on, these three principals were given the name of Issac's three laws of Robotics. These three laws state that:

- Robots will never harm human beings.
- Robots will follow instructions given by humans with breaking law one.
- Robots will protect themselves without breaking other rules.

CHAPTER -2
HARDWARE

2,1: Components Required for the project:

1. Arduino UNO

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 on a 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.

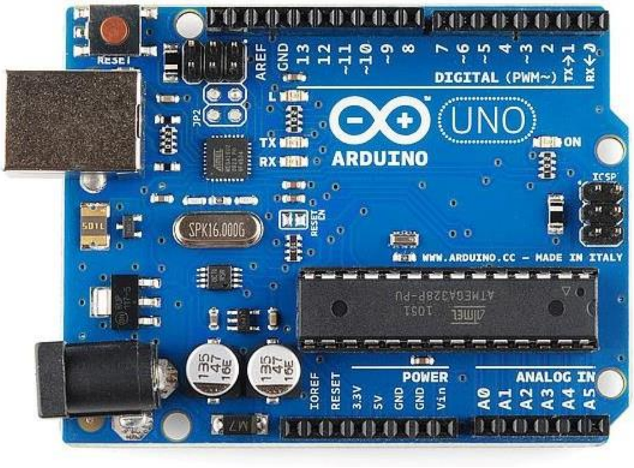


Fig 2.1: Arduino uno

2. Bluetooth HC-05 Module

Wireless communication is swiftly replacing the wired connection when it comes to electronics and communication. Designed to replace cable connections HC-05 uses serial communication to communicate with the electronics. Usually, it is used to connect small devices like mobile phones using a short-range wireless connection to exchange files. It uses the 2.45GHz frequency band. The transfer rate of the data can vary up to 1Mbps and is in range of 10 meters.
The HC-05 module can be operated within 4-6V of power supply. It supports baud rate of 9600, 19200, 38400, 57600, etc.

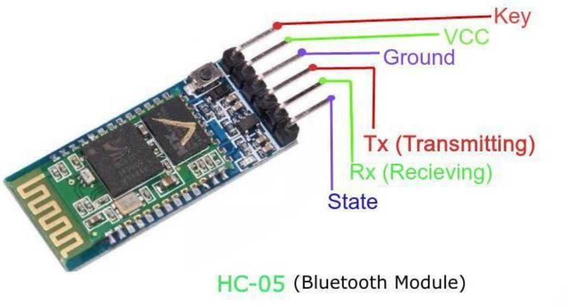


Fig 2.2: HC-05 (Bluetooth module)

3. Motor Driver L298

The L298 is an integrated monolithic circuit in a 15-lead Multiwatt and PowerSO20 packages. It is a high voltage, high current dual full-bridge driver designed to accept standard TTL logic levels and drive inductive loads such as relays, solenoids, DC and stepping motors.

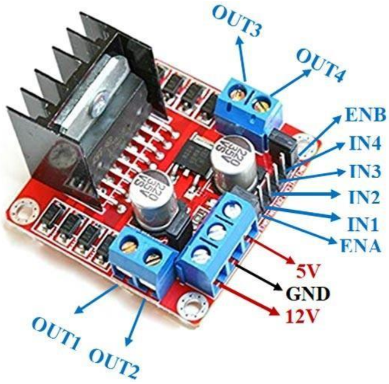


Fig 2.3: Motor driver L298

4. Jumper Wires

Jumper wires are electrical wires with connector pins at each end. They are used to connect two points in a circuit without soldering. You can use jumper wires to modify a circuit or diagnose problems in a circuit.

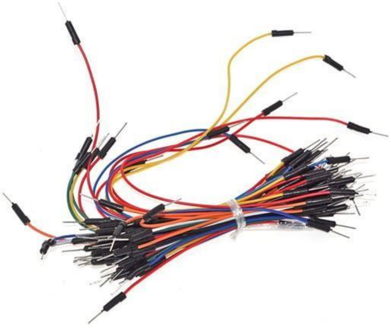


Fig 2.4: Jumper cables

5. Gear Motors (x4)

A geared motor is a component whose mechanism adjusts the speed of the motor, leading them to operate at a certain speed. geared motor has the ability to deliver high torque at low speeds, as the gearhead functions as a torque multiplier and can allow small motors to generate higher speed.

Fig 2.5: Gear Motors

6. 12 V battery (x3 Lithium-ion cells)

A battery is a device that converts chemical energy contained within its active materials directly into electric energy by means of an electrochemical oxidation-reduction (redox) reaction. This type of reaction involves the transfer of electrons from one material to another via an electric circuit.

Fig 2.6: Battery

7. Motor Wheels

A wheel containing the motor and all the required gearing so that it is an independent drive unit.

Fig 2.7: Motor wheels

CHAPTER-3
SOFTWARE

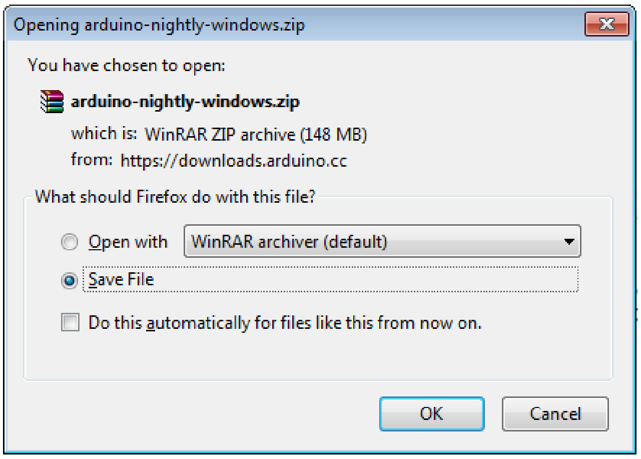
3.1: ARDUINO INSTALLATION

Step 1 − First you must have your Arduino board (you can choose your favorite board) and a USB cable. In case you use Arduino UNO, Arduino Duemilanove, Nano, Arduino Mega 2560, or Diecimila, you will need a standard USB cable (A plug to B plug), the kind you would connect to a USB printer as shown in the following image.

Fig 3.1: USB cable

Step 2 − Download Arduino IDE Software.

You can get different versions of Arduino IDE from the Download page on the Arduino Official website. You must select your software, which is compatible with your operating system (Windows, IOS, or Linux). After your file download is complete, unzip the file.



Opening arduino-nightly-windows.zip

You have chosen to open:

arduino-nightly-windows.zip

which is: WinRAR ZIP archive (148 MB)

from: https://downloads.arduino.cc

What should Firefox do with this file?

○ Open with WinRAR archiver (default)

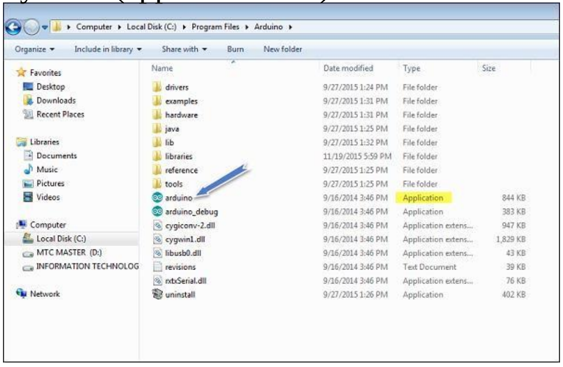
● Save File

□ Do this automatically for files like this from now on.

Fig 3.2: Unzipping

Step 3 − Launch Arduino IDE.

After your Arduino IDE software is downloaded, you need to unzip the folder. Inside the folder, you can find the application icon with an infinity label (application.exe). Double-click the icon to start the IDE.



Computer ▸ Local Disk (C:) ▸ Program Files ▸ Arduino ▸

Name	Date modified	Type	Size
drivers	9/27/2015 1:24 PM	File folder	
examples	9/27/2015 1:31 PM	File folder	
hardware	9/27/2015 1:31 PM	File folder	
java	9/27/2015 1:25 PM	File folder	
lib	9/27/2015 1:32 PM	File folder	
libraries	11/19/2015 5:59 PM	File folder	
reference	9/27/2015 1:25 PM	File folder	
tools	9/27/2015 1:25 PM	File folder	
arduino	9/16/2014 3:46 PM	Application	844 KB
arduino_debug	9/16/2014 3:46 PM	Application	383 KB
cygiconv-2.dll	9/16/2014 3:46 PM	Application extens...	947 KB
cygwin1.dll	9/16/2014 3:46 PM	Application extens...	1,829 KB
libusb0.dll	9/16/2014 3:46 PM	Application extens...	43 KB
revisions	9/16/2014 3:46 PM	Text Document	39 KB
rxtxSerial.dll	9/16/2014 3:46 PM	Application extens...	76 KB
uninstall	9/27/2015 1:26 PM	Application	402 KB

Fig 3.3:Opening Ardunio

Step 5 − Open your first project.

Once the software starts, you have two options −

- Create a new project.
- Open an existing project example.

To create a new project, select File → **New**.

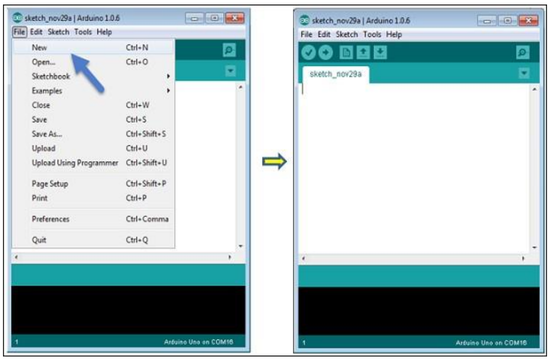


Fig 3.4: Opening new file.

To open an existing project example, select File → Example → Basics → Blink.

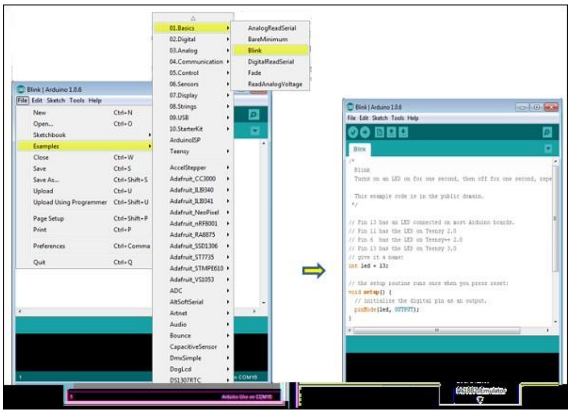


Fig 3.5: Example

Here, we are selecting just one of the examples with the name **Blink**. It turns the LED on and off with some time delay. You can select any other example from the list.

Step 6 − Select your Arduino board.

To avoid any error while uploading your program to the board, you must select the correct Arduino board name, which matches with the board connected to your computer.

Go to Tools → Board and select your board.

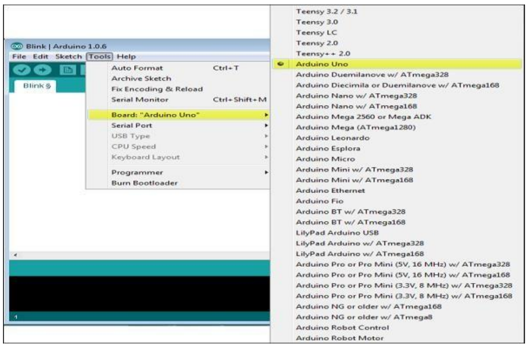

Blink | Arduino 1.0.6
File Edit Sketch Tools Help
Auto Format Ctrl+T
Archive Sketch
Fix Encoding & Reload
Serial Monitor Ctrl+Shift+M
Board: "Arduino Uno"
Serial Port
USB Type
CPU Speed
Keyboard Layout
Programmer
Burn Bootloader
Teensy 3.2 / 3.1
Teensy 3.0
Teensy LC
Teensy 2.0
Teensy++ 2.0
Arduino Uno
Arduino Duemilanove w/ ATmega328
Arduino Diecimila or Duemilanove w/ ATmega168
Arduino Nano w/ ATmega328
Arduino Nano w/ ATmega168
Arduino Mega 2560 or Mega ADK
Arduino Mega (ATmega1280)
Arduino Leonardo
Arduino Esplora
Arduino Micro
Arduino Mini w/ ATmega328
Arduino Mini w/ ATmega168
Arduino Ethernet
Arduino Fio
Arduino BT w/ ATmega328
Arduino BT w/ ATmega168
LilyPad Arduino USB
LilyPad Arduino w/ ATmega328
LilyPad Arduino w/ ATmega168
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega328
Arduino Pro or Pro Mini (5V, 16 MHz) w/ ATmega168
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega328
Arduino Pro or Pro Mini (3.3V, 8 MHz) w/ ATmega168
Arduino NG or older w/ ATmega168
Arduino NG or older w/ ATmega8
Arduino Robot Control
Arduino Robot Motor

Fig 3.6: Selection of Arduino board

Here, we have selected Arduino Uno board according to our tutorial, but you must select the name matching the board that you are using.

Step 7 − Select your serial port.

Select the serial device of the Arduino board. Go to **Tools → Serial Port** menu. This is likely to be COM3 or higher (COM1 and COM2 are usually reserved for hardware serial ports). To find out, you can disconnect your Arduino board and re-open the menu, the entry that disappears should be of the Arduino board. Reconnect the board and select that serial port.

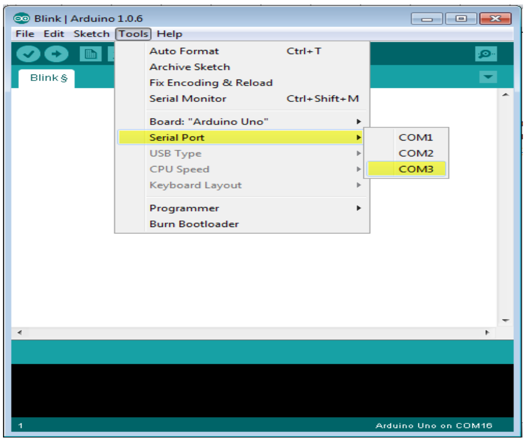


Fig 3.7: Port selection

Step 8 − Upload the program to your board.

Before explaining how we can upload our program to the board, we must demonstrate the function of each symbol appearing in the Arduino IDE toolbar.

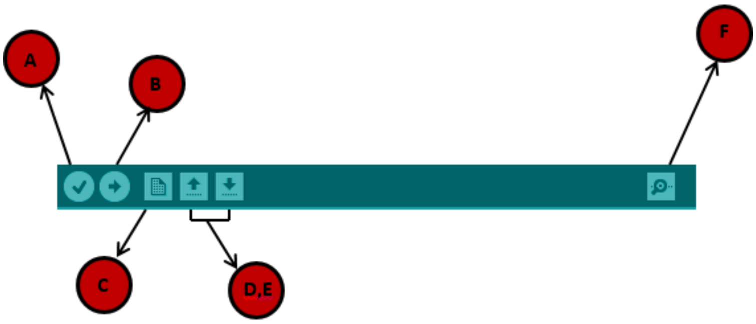


Fig 3.8: Toolbar

A − Used to check if there is any compilation error.

B − Used to upload a program to the Arduino board.

C − Shortcut used to create a new sketch.

D − Used to directly open one of the example sketches.

E − Used to save your sketch.

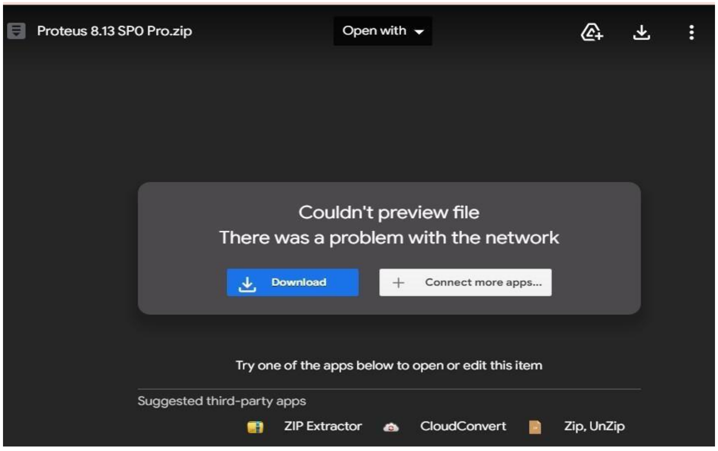
F − Serial monitor used to receive serial data from the board and send the serial data to the board.

Now, simply click the "Upload" button in the environment. Wait a few seconds; you will see the RX and TX LEDs on the board, flashing. If the upload is successful, the message "Done uploading" will appear in the status bar.

3.2: PROTEUS INSTALLATION

Steps to Install Proteus Software in Windows 10

Step 1: First we need to Download Proteus. Here we are going to install Proteus 8.13. Let's For Downloading Click on Download. It will show the following interface now click on Download.

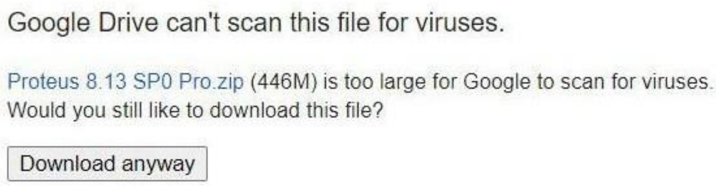


Couldn't preview file
There was a problem with the network

Try one of the apps below to open or edit this item

Fig 3.9: Proteus File Link

Step 2: In the case of google drive it may show the following interface just click on Download anyway.

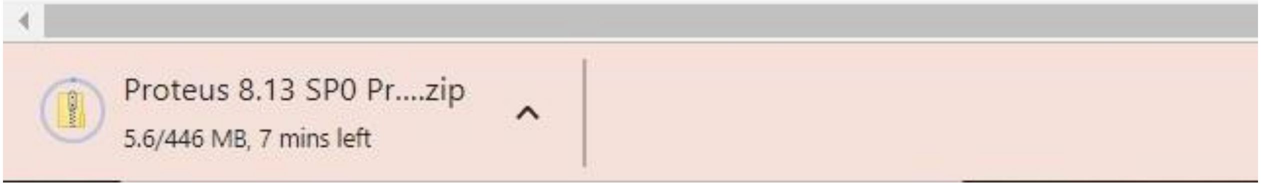


Google Drive can't scan this file for viruses.

Proteus 8.13 SP0 Pro.zip (446M) is too large for Google to scan for viruses. Would you still like to download this file?

Fig 3.10: Download Option

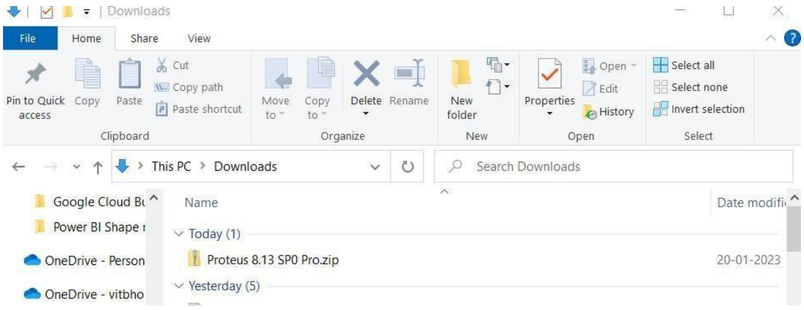
Step 3: It will start to download the zip file. Based on your internet speed it will take some time. Wait until the download process is completed.



Proteus 8.13 SP0 Pr....zip
5.6/446 MB, 7 mins left

Fig 3.11: ZIP File

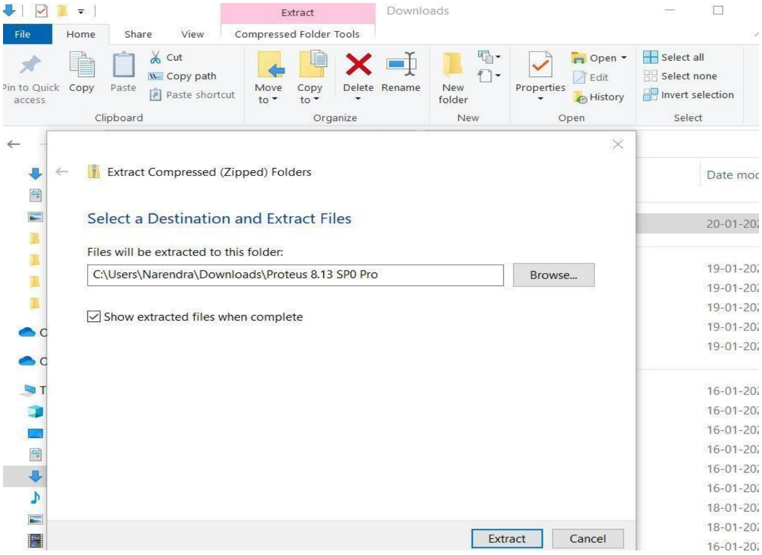
Step 4: When downloading is completed and then going to the download folder it will show the zip file.



Today (1)
Proteus 8.13 SP0 Pro.zip 20-01-2023
Yesterday (5)

Fig 3.12: Find ZIP File

Step 5: Now on right-click the file and extract this using whatever software you have to extract the zip file and select the destination. Here, browses the location where you want to save.



Extract Compressed (Zipped) Folders

Select a Destination and Extract Files

Files will be extracted to this folder:

C:\Users\Narendra\Downloads\Proteus 8.13 SP0 Pro

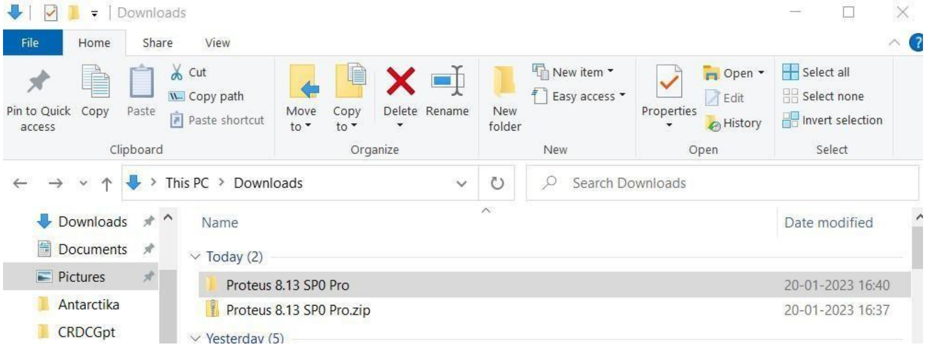
Browse...

☑ Show extracted files when complete

Extract Cancel

Fig 3.13: Extraction of files

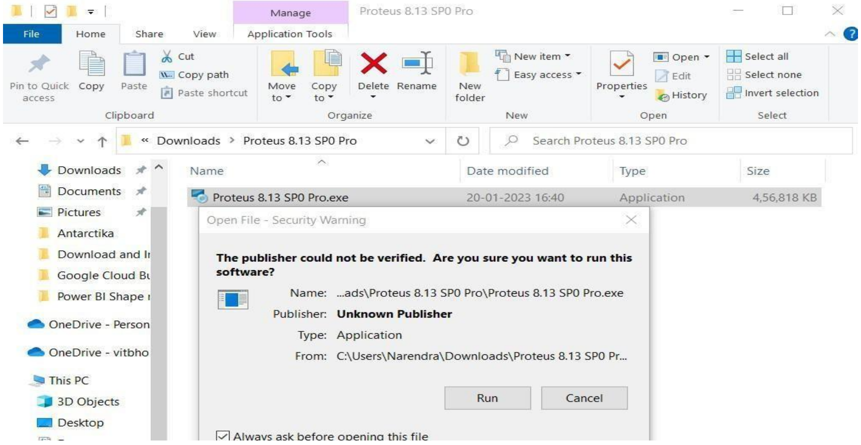
Step 6: Extracted folder will look like this.



Name	Date modified
Today (2)	
Proteus 8.13 SP0 Pro	20-01-2023 16:40
Proteus 8.13 SP0 Pro.zip	20-01-2023 16:37
Yesterday (5)	

Fig 3.14: Extracted Folder

Step 7: Now open Proteus 8.13 SP0 Pro Folder and click on the .exe file. It will ask to run. Click on Run.



Name	Date modified	Type	Size
Proteus 8.13 SP0 Pro.exe	20-01-2023 16:40	Application	4,56,818 KB

Open File - Security Warning

The publisher could not be verified. Are you sure you want to run this software?

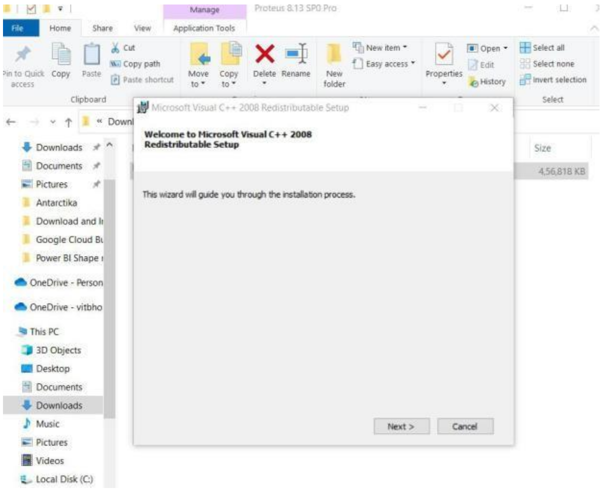
Name: ...ads\Proteus 8.13 SP0 Pro\Proteus 8.13 SP0 Pro.exe
Publisher: **Unknown Publisher**
Type: Application
From: C:\Users\Narendra\Downloads\Proteus 8.13 SP0 Pr...

Run Cancel

☑ Always ask before opening this file

Fig 3.15: Running Proteus Software

Step 8: In case select you don't have an installation wizard then first it will ask the Click an install wizard. Click next and mark check on Agree. Then it will install the wizard, it is an optional process, if you already have this wizard then it will go to the next step directly.



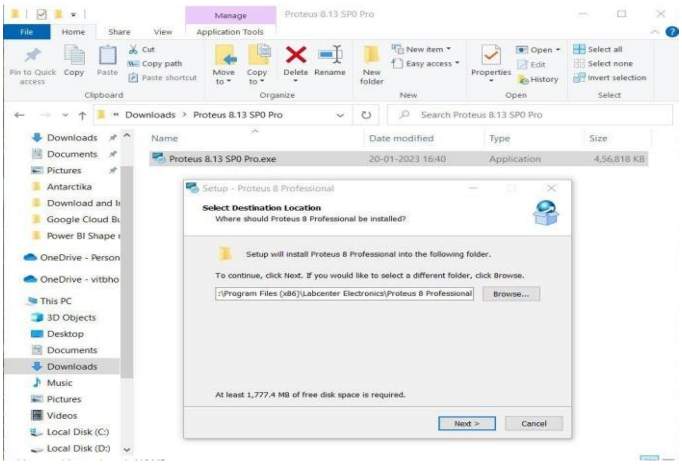
Microsoft Visual C++ 2008 Redistributable Setup

Welcome to Microsoft Visual C++ 2008 Redistributable Setup

This wizard will guide you through the installation process.

Fig 3.16: Installation of Wizard

Step 9: Next it will show the following interface and we need to select the location where we want to install this software. Click on Next.



Setup - Proteus 8 Professional

Select Destination Location

Where should Proteus 8 Professional be installed?

Setup will install Proteus 8 Professional into the following folder.

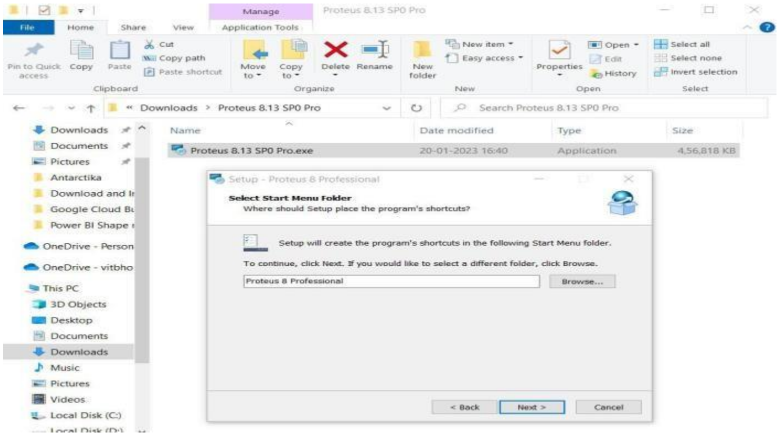
To continue, click Next. If you would like to select a different folder, click Browse.

:\Program Files (x86)\Labcenter Electronics\Proteus 8 Professional

At least 1,777.4 MB of free disk space is required.

Fig 3.17: Selecting Location

Step 10: Now it will show the following interface. Simply click on Next.



Setup - Proteus 8 Professional

Select Start Menu Folder

Where should Setup place the program's shortcuts?

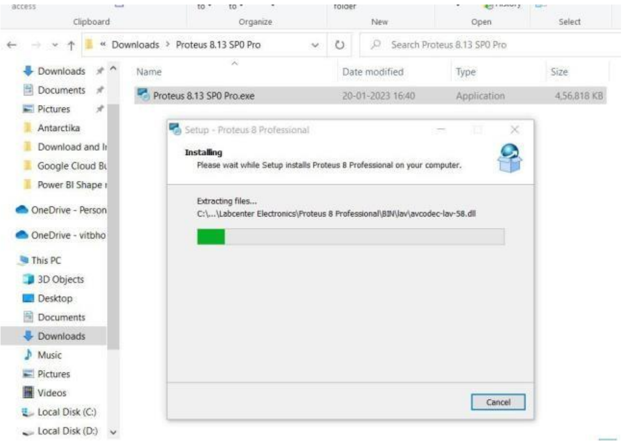
Setup will create the program's shortcuts in the following Start Menu folder.

To continue, click Next. If you would like to select a different folder, click Browse.

Proteus 8 Professional

Fig 3.18: Allowing Permissions

Step 11: Now it will start to install. It will take some time.



Downloads › Proteus 8.13 SP0 Pro
Search Proteus 8.13 SP0 Pro
Proteus 8.13 SP0 Pro.exe
20-01-2023 16:40
Application
4,56,818 KB
Setup - Proteus 8 Professional
Cancel

Installing

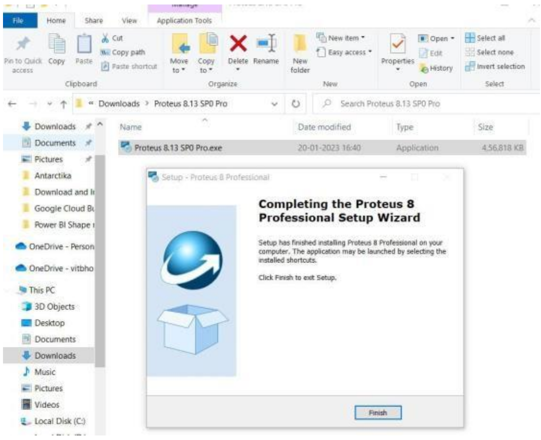
Please wait while Setup installs Proteus 8 Professional on your computer.

Extracting files...

C:\...\Labcenter Electronics\Proteus 8 Professional\BIN\lav\avcodec-lav-58.dll

Fig 3.19: Final Installation

Step 12: After Completion of Installation, it will show the following interface. Simply Click onFinish. Our Proteus 8.13 is installed in our Windows 10.



Completing the Proteus 8 Professional Setup Wizard

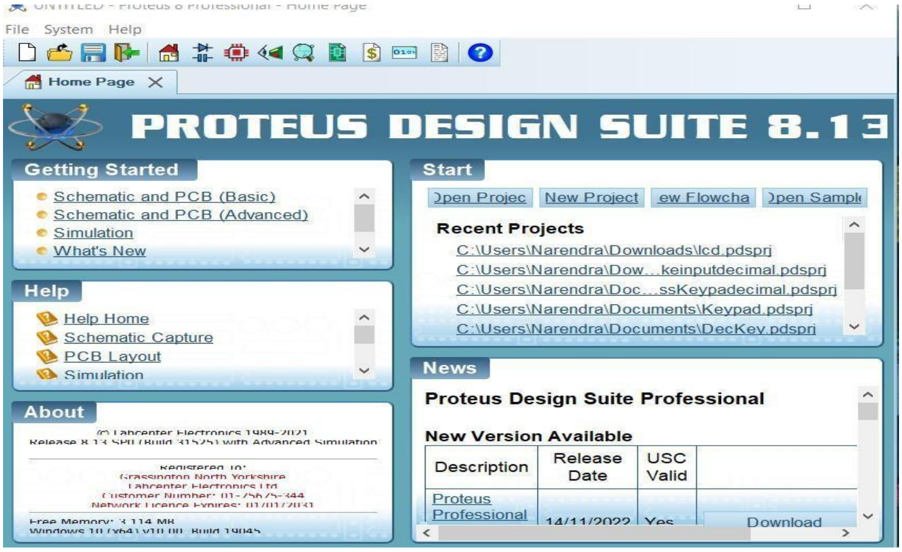
Setup has finished installing Proteus 8 Professional on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

Fig 3.20: Installation Complete

Step 13: In the desktop icon of this software will show like this. For checking whether it is installed perfectly or not, click on Proteus Professional Icon.

Step 14: It is opening correctly and now we can start to work with this Proteus.



C:\Users\Narendra\Downloads\lcd.pdsprj

C:\Users\Narendra\Dow…keinputdecimal.pdsprj

C:\Users\Narendra\Doc…ssKeypadecimal.pdsprj

C:\Users\Narendra\Documents\Keypad.pdsprj

C:\Users\Narendra\Documents\DecKey.pdsprj

(C) Labcenter Electronics 1989-2021

Release 8.13 SP0 (Build 31525) with Advanced Simulation

Registered To:

Grassington North Yorkshire

Labcenter Electronics Ltd

Customer Number: 01-75675-344

Network Licence Expires: 01/01/2031

Free Memory: 3 114 MB

Windows 10 (x64) v10.00, Build 19045

Proteus Design Suite Professional

New Version Available

Description	Release Date	USC Valid
Proteus Professional	14/11/2022	Yes

Fig 3.21: Proteus Interface

This was the Installation process of Proteus 8.13 in Windows 10.

3.3: ARDUINO BLUETOOTH CONTROL

- Arduino Bluetooth Control is an application that allows you to control your Arduino board (and similar boards) via Bluetooth, and so to create awesome and fully customized projects, with the new features available within the app.
- The settings section allows you to adapt the application to your needs, through a very simple and intuitive interface.
- The application also smartly remembers your Bluetooth module and tries to connect automatically to the latest one you have used, so you won't have to select it every time you use it.

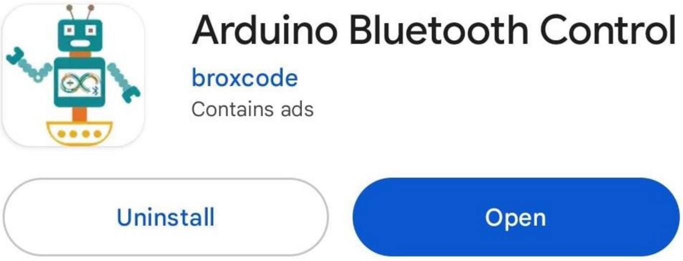


Fig 3.22 : Arduino Bluetooth Control

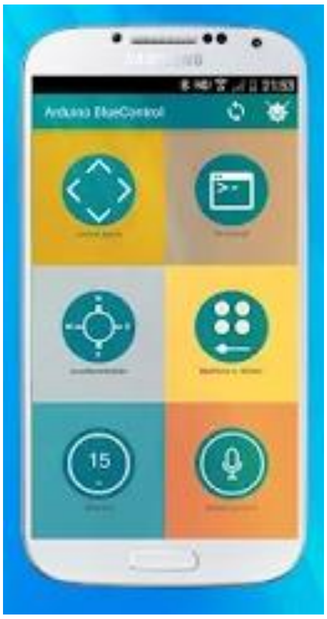


Fig 3.23 : Application Interface

Features:

There are multiple options to control Arduino Board through this application. Few of them are as shown below:

- **ARROW KEYS –** We can control the Arduino with arrow keys by assigning particular commands to keys.

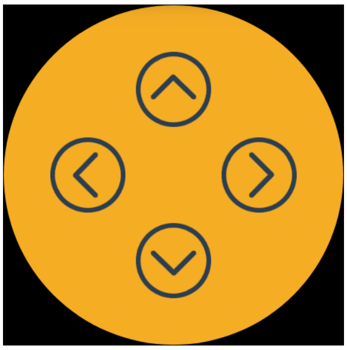
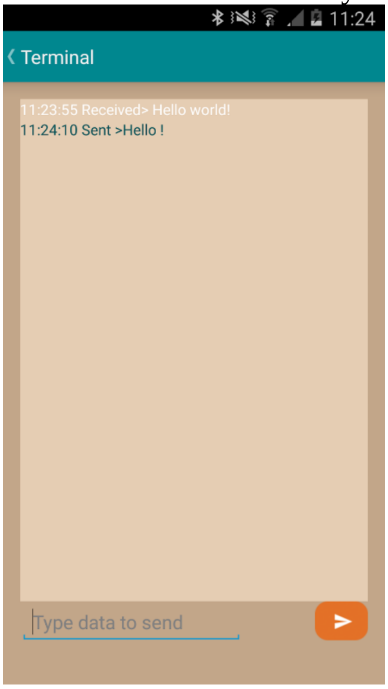


Fig 3.24 : Arrow Keys

- **TERMINAL –** We can give serial data to the Arduino Board by using this terminal option.



11:23:55 Received> Hello world!
11:24:10 Sent >Hello !

Fig 3.25 : Terminal

- **ACCELEROMETER –** We can access the Arduino Board by assigning commands to the position of accelerometer.
- **BUTTONS AND SLIDER** – We can give commands to the Arduino Board by using buttons and slider which are having predefined commands.

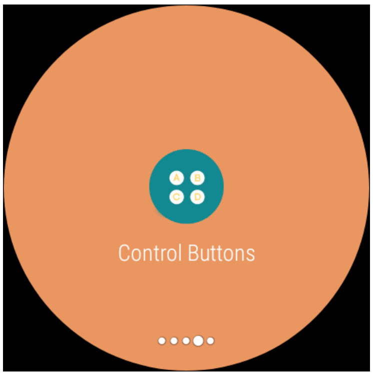


Fig 3.26 : Buttons and Slider

- **VOICE CONTROL –** We can give voice commands to Arduino Board that had predefined meaning.

Fig 3.27 : Voice Control

CHAPTER - 4

PROGRAM

```
void setup() {
pinMode(13,OUTPUT); //left motors forward
pinMode(12,OUTPUT); //left motors reverse
pinMode(11,OUTPUT); //right motors forward
pinMode(10,OUTPUT); //right motors reverse
pinMode(9,OUTPUT);  //Led
Serial.begin(9600);
}
void loop() {
if(Serial.available()){
t = Serial.read();
Serial.println(t);
}
if(t == 'F'){            //move forward(all motors rotate in forward direction)
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
}
else if(t == 'B'){      //move reverse (all motors rotate in reverse direction)
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
}
else if(t == 'R'){      //turn right (left side motors rotate in forward direction, right side motors doesn't rotate)
  digitalWrite(11,HIGH);
}
else if(t == 'L'){      //turn left (right side motors rotate in forward direction, left side motors doesn't rotate)
  digitalWrite(13,HIGH);
}
else if(t == 'M'){    //turn led on or off)
  digitalWrite(9,HIGH);
```

```
}
else if(t == 'm'){
  digitalWrite(9,LOW);
}


else if(t == 'S'){      //STOP (all motors stop)
  digitalWrite(13,LOW);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
else if(t == 'Q'){
  digitalWrite(13,HIGH);
  delay(2000);
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
  delay(2000) ;
  digitalWrite(13,LOW);
  digitalWrite(11,LOW);
}
 else if(t == 'E'){
  digitalWrite(11,HIGH);
  delay(2000);
  digitalWrite(13,HIGH);
  digitalWrite(11,HIGH);
  delay(2000);
  digitalWrite(13,LOW);
  digitalWrite(11,LOW);
}
 else if(t == 'Z'){
  digitalWrite(13,HIGH);
  delay(2000);
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
  delay(2000) ;
  digitalWrite(12,LOW);
  digitalWrite(10,LOW);
  digitalWrite(13,LOW);
}
 else if(t == 'C'){
  digitalWrite(11,HIGH);
  delay(2000);
  digitalWrite(12,HIGH);
  digitalWrite(10,HIGH);
  delay(2000);
  digitalWrite(12,LOW);
  digitalWrite(11,LOW);
  digitalWrite(10,LOW);
}
}
```

CHAPTER-5
METHODOLOGY

5.1: ARDUINO CODING

1. Open the Arduino IDE.
2. Enter the code in a new file.
3. Compile the code for errors.
4. Rectify the errors if any.

5.2: SIMULATION

1. Open proteus software.
2. Create new file.
3. Select the components.
4. Connect the components as per project model.
5. Convert Arduino code to binary file.
6. Now dump the binary file into the Arduino in proteus file.
7. Simulate the circuit.

5.3: HARDWARE

1. Collect the components required for the project.
2. Connect the components as per project model we had simulated.
3. Reset the Arduino board.
4. Connect the data cable between system and the Arduino Board.
5. Now open the Arduino software in the system.
6. Open the project file.
7. Select the Arduino Board type (Arduino UNO) in the board option.
8. Select the port to transfer the code.
9. Upload the code by clicking on upload.

5.4: WORKING

1. Open the Arduino Bluetooth Control application in mobile.
2. Connect the Bluetooth module via mobile Bluetooth.
3. Now define the commands.
4. Open the terminal mode.
5. Enter
    - F – Forward
    - B – Backward
    - R – Right
    - L – Left
    - S – Stop
6. By using above commands we can control the Robotic Car.

CHAPTER-6

RESULT

OUTPUT:

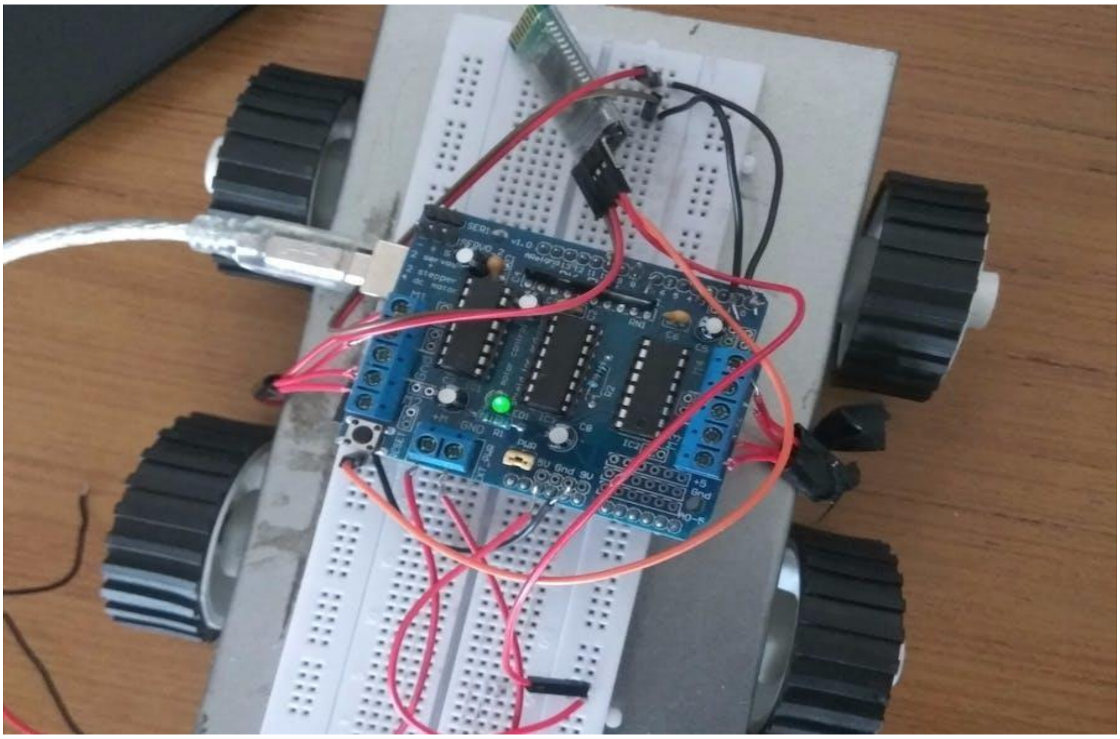


Fig 6.1: Outcome