# IoT BASED VIRTUAL ROBOT

# MINOR 2 - PROJECT REPORT

विद्या तत्व ज्योतिसम:

**Department of ECE**

**Jaypee Institute of Information Technology, Noida**

**<u>Supervisor</u>: Mr. Mandeep Narula**

**<u>External</u>: Mrs. Shraddha Saxena**

**Submitted By:**

Aakash Mehta 20102004

Rishika Bahl 20102169

# CERTIFICATE

This is to certify that the work titled "**IOT BASED VIRTUAL ROBOT**" submitted by **"AAKASH MEHTA (20102004) and RISHIKA BAHL (20102169)"** in partial fulfillment for the award of degree of Bachelor of technology of Jaypee Institute of Information Technology, Noida has been carried out under my supervision. This work has not been submitted partially or wholly to any other University or Institute for the award of this or any other degree or diploma.

**Signature of Supervisor** …………………….

**Name of Supervisor** ……………………..

**Designation** ……………………..

**Date** ……………………..

# ACKNOWLEDGEMENT

It gives us immense pleasure to express our deepest sense of gratitude and sincere thanks to our highly esteemed guide, **Mr. Mandeep Narula**, Dept. Of Electronics and Communication, Jaypee Institute of Information Technology for providing us an opportunity to present our project on "**IOT Virtual Doctor Robot**" and also for his valuable guidance, encouragement and wholehearted cooperation and constructive criticism throughout the project. We with full pleasure converge our heartiest thanks to our external guide **Mrs. Shraddha Saxena**, Dept. Of Electronics and Communication, for their invaluable advice and wholehearted cooperation without which the project would not have seen the light of the day. We give heartiest thanks to all the faculties of the Department Of Electronics and Communication, our friends and seniors for their valuable advice and encouragement.


**Signature of the student** ……………………..

**Name of Student** ……………………..

**Date** ……………………..

# TABLE OF CONTENT

# ABSTRACT

Doctors are usually needed to work at every hospital and emergency center every now and then because it  is not feasible for every doctor to be available at every place at the desired time. The problem with video calling is that video calls need to be done from a PC or laptop on a desk. This limits the doctors capacity to view patients or around the operation theater at will or even move through hospital rooms as needed. The system makes use of a robotic vehicle with 4 wheel drive for easy navigation. The robot also includes a controller box for circuitry and a mounting to hold a mobile phone or tablet. The mobile or tablet is used to hold live video calls. The doctor can use an IOT based panel to control the robot. The control commands sent online are received by the robot controller. The Robot also has a contactless IR sensor for monitoring the Temperature of the Patient if needed.

| | |
|---|---|
| **Signature of Student** | **Signature of Supervisor** |
| **Name** | **Name** |
| **Date** | **Date** |

# LIST OF FIGURES

# LIST OF TABLES

# CHAPTER - 1 INTRODUCTION

## 1.1 THE IDEA

The system is equipped with a four-wheel drive robotic vehicle that can navigate easily through different environments. It also includes a controller box for circuitry and a mount for a mobile phone or tablet, which can be used to make live video calls. The robot can be controlled remotely by a doctor using an IOT-based panel, with the control commands transmitted over Wi-Fi internet. The robot responds to these commands in real time, adjusting its motor functions to achieve the desired movements. In addition, the robot has a battery status alert to remind the doctor to charge it when necessary. The Robot also has a contactless IR sensor for monitoring the Temperature of the Patient if needed.

## 1.2 CONCEPT

The system being described is a four-wheel drive robotic vehicle that has been designed to navigate through different environments. This vehicle is equipped with various features that make it useful in the medical field.

One of the primary features of the robot is its controller box, which contains the circuitry necessary to control its movements. The robot can be controlled remotely by a doctor using an IoT-based panel, which is connected to the robot via Wi-Fi internet. The doctor can send control commands to the robot using this panel, and the robot will respond in real-time, adjusting its motor functions to achieve the desired movements. This allows the robot to be operated from a distance, which can be particularly useful in situations where a doctor needs to examine a patient without being physically present.

The robot also has a mount for a mobile phone or tablet, which can be used to make live video calls. This feature can be particularly useful for remote consultations or for allowing family members to communicate with a patient who is in isolation. Another important feature of the robot is its battery status alert. This feature reminds the doctor to charge the robot when its battery is running low, ensuring that the robot is always ready for use.

Finally, the robot is equipped with a contactless IR sensor for monitoring the temperature of the patient. This feature allows the doctor to remotely monitor the patient's temperature without needing to be physically present, which can be particularly useful in situations where a patient's temperature needs to be monitored regularly.

## 1.3 MOTIVATION FOR PROJECT

The motivation behind this project is to create a robotic vehicle that can help doctors and healthcare professionals remotely monitor and communicate with patients in different environments. This is particularly relevant in situations where physical contact with patients is not possible, such as during pandemics or in isolated regions.

By equipping the robot with a four-wheel drive system, it can navigate through different environments and reach patients in areas that may be difficult for humans to access. The mount for a mobile phone or tablet enables doctors to make live video calls and remotely assess the patient's condition, while the robot's controller box and IOT-based panel allow for real-time control and adjustments of the robot's motor functions.

The contactless IR sensor for monitoring the patient's temperature is particularly useful during pandemics or other infectious diseases, as it allows doctors to remotely monitor the patient's health without having to come into contact with them.

Overall, this project aims to provide doctors and healthcare professionals with a remote tool that can assist in patient monitoring and communication, particularly in situations where physical contact is not possible or safe. It has the potential to improve access to healthcare for people in isolated regions or those who are unable to visit healthcare facilities in person.

# CHAPTER - 2 HARDWARE AND SOFTWARE COMPONENTS

## 2.1 HARDWARE COMPONENTS

### 2.1.1 NodeMCU - ESP8266 Wifi Module

A Wi-Fi module is a small electronic device that enables wireless communication over a local area network (LAN) using Wi-Fi technology.

**Features of a NodeMCU:**

- Wireless Connectivity: The module includes an antenna and radio circuitry that allows it to connect to a wireless network.
- Security: Wi-Fi modules provide security features like encryption and authentication to ensure that only authorized users can access the network.
- Low Power Consumption: Most Wi-Fi modules are designed to operate on low power to conserve battery life in battery-powered devices.
- Size: Wi-Fi modules are typically small and compact, making them suitable for use in small devices and applications with limited space.
- Easy Integration: They often come with a variety of interfaces, such as UART, SPI, I2C, and USB, making it easy to integrate into various systems.
- Range: Wi-Fi modules typically have a range of up to 100 meters in open spaces, which can be extended with the use of external antennas.
- Data Transfer Rate: Wi-Fi modules support high data transfer rates of up to several hundred Mbps, making them ideal for streaming high-quality video and audio.

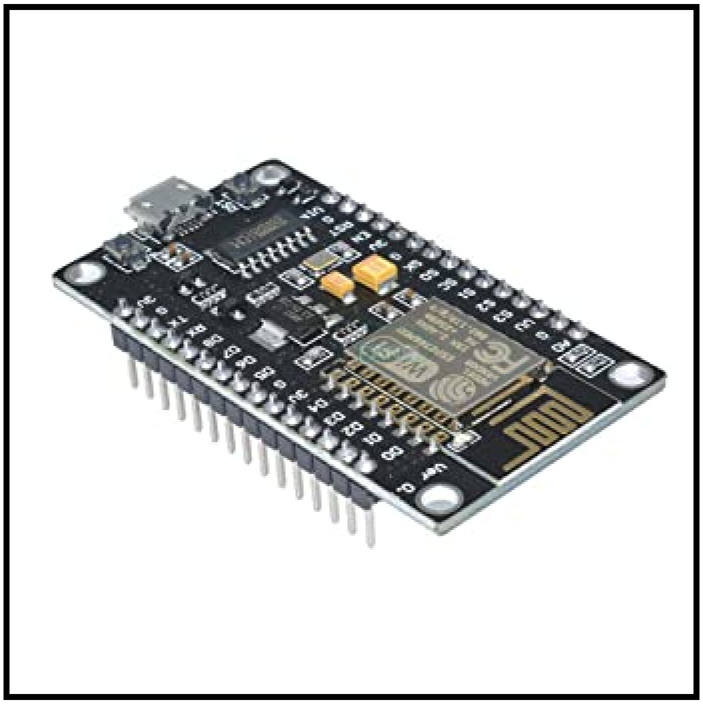


**Fig 1 - NodeMCU ESP8266**

**Pinout Configuration:**

| Pin Category | Name | Description |
|---|---|---|
| **Power** | Micro-USB, 3.3V, GND, Vin | Micro-USB: NodeMCU can be powered through a USB port.<br>3.3V: Regulated 3.3V can be supplied to this pin to power the board.<br>GND: Ground Pins<br>Vin: External power supply |
| **Control Pins** | EN, RST | The pin and button resets the microcontroller. |

| | | |
|---|---|---|
| **Analog Pins** | A0 | Used to measure analog voltage in the range of 0-3.3V |
| **GPIO Pins** | GPIO0 - 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** | TDX0, RDX0, TDX2, RDX2 | 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. |

**Table 1: NodeMCU PinOut Configuration**

Overall, a Wi-Fi module is a versatile and essential component in today's connected world, enabling devices to connect to the internet and communicate with other devices wirelessly.

### 2.1.2 MLX90614 INFRARED TEMPERATURE SENSOR

The MLX90614 is a Contactless Infrared (IR) Digital Temperature Sensor that can be used to measure the temperature of a particular object ranging from -70° C to 382.2°C. The sensor uses IR rays to measure the temperature of the object without any physical contact and communicates to the microcontroller using the I2C protocol.

**Working Principle of MLX90614**

As mentioned earlier, the MLX90614 sensor can measure the temperature of an object without any physical contact with it. This is made possible with a law called Stefan-Boltzmann Law, which states that all objects and living beings emit IR Energy and the intensity of this emitted IR energy will be directly proportional to the temperature of that object or living being. So the MLX90614 sensor calculates the temperature of an object by measuring the amount of IR energy emitted from it.

**MLX90614 Temperature Sensor Specifications**

- Operating Voltage: 3.6V to 5V (available in 3V and 5V version)
- Supply Current: 1.5mA
- Object Temperature Range: -70° C to 382.2°C
- Ambient Temperature Range: -40° C to 125°C
- Accuracy: 0.02°C
- Field of View: 80°
- Distance between object and sensor: 2cm-5cm (approx.)

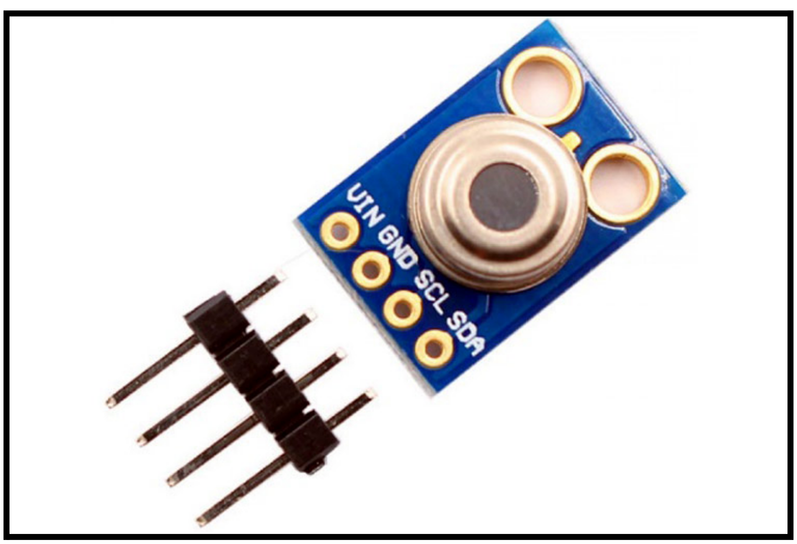


**Fig 2 - MLX90614 IR Temperature Sensor**

**MLX90614 Pinout Configuration:**

| **Pin No.** | **Pin Name** | **Description** |
|---|---|---|
| **1.** | VDD | Vdd can be used to power the sensor, typically using 5V. |
| **2.** | GND | The metal can also act as a ground. |
| **3.** | SDA - Serial Data | Serial Data pin used for I2C communication. |
| **4.** | SCL - Serial Clock | Serial Clock pin used for I2C communication. |

**Table 2 - MLX90614 PinOut Configuration**

**Advantage of using MLX90614**

- Non-contact temperature sensing: The MLX90614 can measure temperature without making physical contact with the object being measured, which is especially useful in applications where contact temperature sensing is not practical or safe.
- High accuracy: The MLX90614 has a high accuracy of ±0.5°C at room temperature, making it suitable for precise temperature measurement in a variety of applications.
- Wide temperature range: The MLX90614 can measure temperatures ranging from -70°C to +380°C, making it useful in a wide range of temperature measurement applications.
- Low power consumption: The MLX90614 consumes very little power, making it ideal for battery-powered devices.
- Easy to use: The MLX90614 is easy to use, with a simple I2C interface that can be easily integrated into microcontroller-based projects.

### 2.1.3 L298N Motor Driver

L298N Motor Driver Module is a high power motor driver module for driving DC and Stepper Motors. This module consists of an L298 motor driver IC and a 78M05 5V regulator. L298N Module can control up to 4 DC motors, or 2 DC motors with directional and speed control. 78M05 Voltage regulator will be enabled only when the jumper is placed. When the power supply is less than or equal to 12V, then the internal circuitry will be powered by the voltage regulator and the 5V pin can be used as an output pin to power the microcontroller. The jumper should not be placed when the power supply is greater than 12V and separate 5V should be given through a 5V terminal to power the internal circuitry. ENA & ENB pins are speed control pins for Motor A and Motor B while IN1& IN2 and IN3 & IN4 are direction control pins for Motor A and Motor B.

**L298N Pinout Configuration:**

| **Pin No.** | **Pin Name** | **Description** |
|---|---|---|
| **1.** | IN1 & IN2 | Motor A input pins. Used to control the spinning direction of Motor A |
| **2.** | IN3 & IN4 | Motor B input pins. Used to control the spinning direction of Motor B |
| **3.** | ENA | Enables PWM signal for Motor A |
| **4.** | ENB | Enables PWM signal for Motor B |
| **5.** | OUT1 & OUT2 | Output pins of Motor A |
| **6.** | OUT3 & OUT4 | Output pins of Motor B |

| | | |
|---|---|---|
| **7.** | 12V | 12V input from DC power Source |
| **8.** | 5V | Supplies power for the switching logic circuitry inside L298N IC |
| **9.** | GND | Ground Pin |

**Table 3 - Pin Configuration of L298N Motor Driver**

**Features and Specifications:**

- Driver Model: L298N 2A
- Driver Chip: Double H Bridge L298N
- Motor Supply Voltage (Maximum): 46V
- Motor Supply Current (Maximum): 2A
- Logic Voltage: 5V
- Driver Voltage: 5-35V
- Driver Current:2A
- Logical Current:0-36mA
- Maximum Power (W): 25W
- Current Sense for each motor
- Heatsink for better performance
- Power-On LED indicator

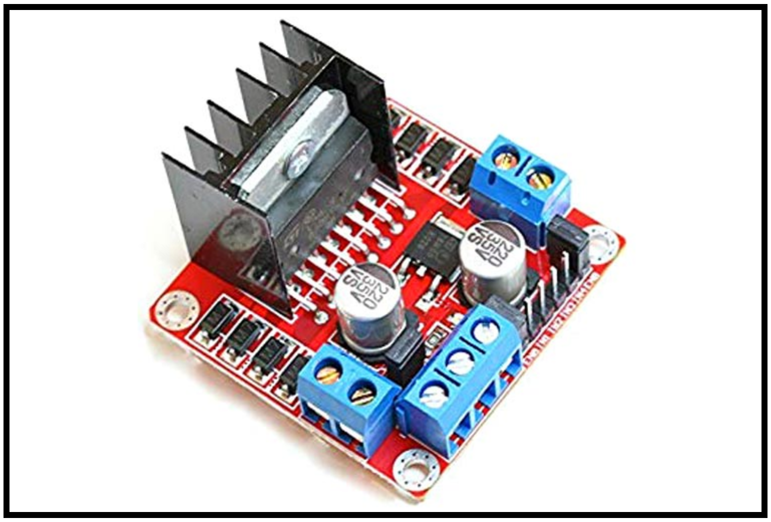


**Figure 3 - L298N Motor Driver**

### 2.2.4 Breadboard

A breadboard, solderless breadboard, or protoboard is a construction base used to build semi-permanent prototypes of electronic circuits. Unlike a perfboard or stripboard, breadboards do not require soldering or destruction of tracks and are hence reusable. For this reason, breadboards are also popular with students and in technological education.

A variety of electronic systems may be prototyped by using breadboards, from small analog and digital circuits to complete central processing units (CPUs).

Compared to more permanent circuit connection methods, modern breadboards have high parasitic capacitance, relatively high resistance, and less reliable connections, which are subject to jostle and physical degradation. Signaling is limited to about 10 MHz, and not everything works properly even well below that frequency.

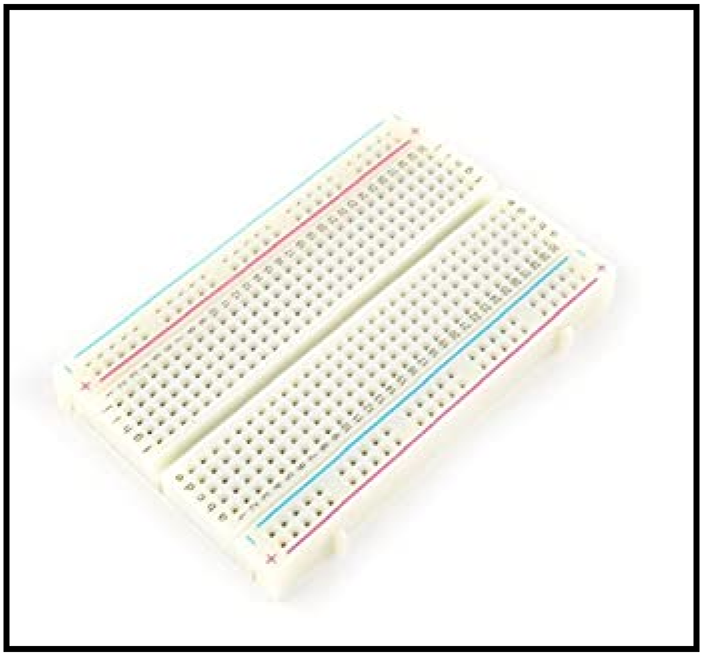


**Figure 4 - Breadboard**

### 2.2.5 Dc Motor

A DC motor is any of a class of rotary electrical motors that converts direct current (DC) electrical energy into mechanical energy. The most common types rely on the forces produced by induced magnetic fields due to flowing current in the coil. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor.

DC motors were the first form of motors widely used, as they could be powered from existing direct-current lighting power distribution systems. A DC motor's speed can be controlled over a wide range, using either a variable supply voltage or by changing the strength of current in its field windings. Small DC motors are used in tools, toys, and appliances. The universal motor, a lightweight brushed motor used for portable power tools and appliances can operate on direct current and alternating current. Larger DC motors are currently used in propulsion of electric vehicles, elevators and hoists, and in drives for steel rolling mills. The advent of power electronics has made replacement of DC motors with AC motors possible in many applications.

**Figure 5 - DC Motor**

### 2.2.6 Li-ion Battery

The 3.7v lithium battery is a lithium battery with a nominal voltage of 3.7v and a full-charge voltage of 4.2v. Its capacity ranges from several hundred to several thousand mAh. It is generally used in various instruments and meters, testing instruments, medical instruments, POS machines, notebook computers, and other products. About the 3.7V lithium battery capacity, the larger the volume of a single lithium battery, the greater the capacity, or we can say that the more the number of lithium batteries in parallel, the greater the capacity. Generally, a 3.7v lithium battery needs a "protection board" for over-charging discharging. The battery without a protection board can only be charged with 4.2V voltage, because the ideal full charge voltage of a lithium battery is 4.2v, once the voltage exceeds 4.2v, the battery may be damaged. Charging in this way requires someone to monitor the condition of the battery at all times. On the contrary, the battery with a protection board can be charged with 5V (range from 4.8V to 5.2V). As we know, in most cases, a 5V charger can be used for USB of computers and mobile phones. The charging cut-off voltage of the 3.7V battery is 4.2V and the discharge cut-off voltage is 3.0V. Therefore, when the open-circuit voltage of the battery is lower than 3.6V, it should be able to charge. It is better to use the 4.2V constant voltage charging mode, so you don't need to pay attention to the charging time. If 5V charging is used, overcharging is easy to happen.


**Figure 6 - Li-ion Battery**

## 2.2 SOFTWARE USED

### 2.2.1 Arduino IDE

Arduino is an open-source hardware and software platform that allows users to create interactive electronic projects. The Arduino software, also known as the Arduino Integrated Development

Environment (IDE), is a software application that is used to write and upload code to Arduino boards.

- The Arduino IDE is a cross-platform application that can be used on Windows, Mac OS X, and Linux operating systems. It is designed to be user-friendly and easy to use, even for beginners with no programming experience.
- The IDE comes with a code editor, a compiler, and a uploader. Users can write their code in the editor, compile it into machine language that the Arduino board can understand, and then upload it to the board using a USB cable.
- The Arduino software also comes with a vast library of pre-written code called "sketches" that users can modify and adapt to their own projects. These sketches are available for a wide range of applications, from controlling LEDs to reading sensor data.

Overall, the Arduino software is an essential tool for anyone interested in creating interactive electronic projects, whether they are beginners or experienced programmers. Its ease of use, cross-platform compatibility, and large community make it a popular choice among hobbyists, students, and professionals alike.

### 2.2.2 Blynk IOT

Blynk is an Internet of Things (IoT) platform that allows developers to easily build applications for connected devices. The platform provides a suite of tools and APIs that enable developers to connect their devices to the internet, collect and analyze data, and control them remotely.

- One of the key features of Blynk is its drag-and-drop interface, which allows developers to quickly build custom user interfaces for their connected devices. The platform also provides a range of widgets, such as buttons, sliders, and graphs, that can be easily added to these interfaces.

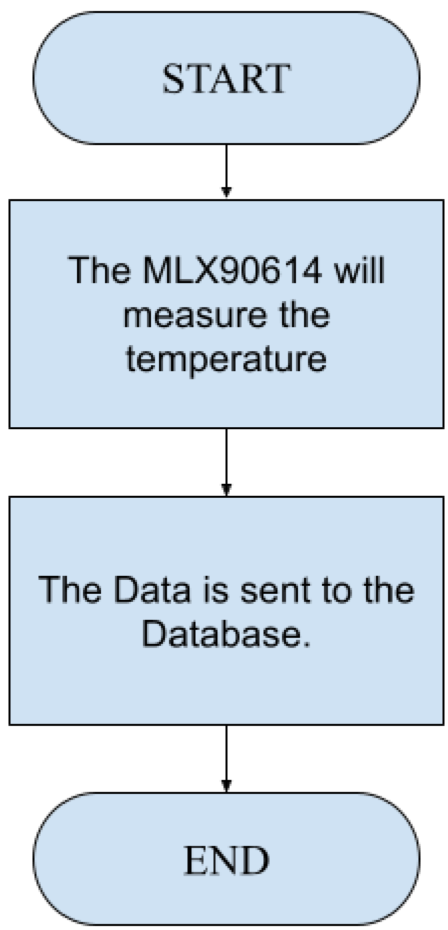
- Blynk supports a wide range of hardware platforms, including Arduino, Raspberry Pi, ESP8266, and more. The platform also integrates with a range of popular IoT services and platforms, such as IFTTT, Amazon Alexa, and Google Home.

Overall, Blynk is a powerful and flexible platform that makes it easy for developers to build IoT applications and connect their devices to the internet.

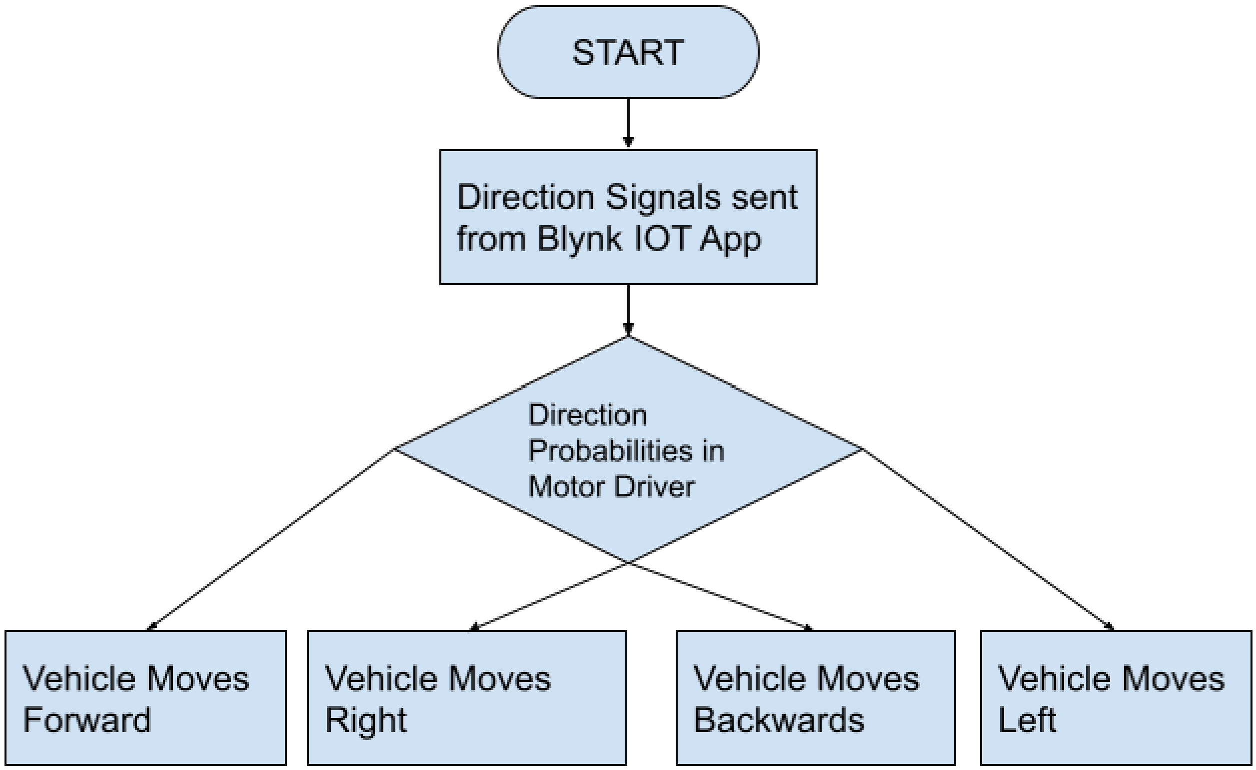
# CHAPTER - 3 FLOW OF PROJECT

## 3.1 FLOWCHART

- **Temperature Sensor**



- **4 wheel Drive Robot**




**3.2 WORKING OF PROJECT**

The project involves a robotic vehicle with 4-wheel drive for easy navigation. The vehicle is equipped with a controller box that contains the circuitry needed to operate the robot. The robot also has a mounting to hold a mobile phone or tablet, which is used to hold live video calls.

The doctor can use an IoT-based panel to control the robot. The control commands sent online are received by the robot controller, which operates over Wi-Fi internet. The received commands are processed in real-time, and the robot motors are operated to achieve the desired movement commands.

**Movement Commands:**

| **Left Side Wheels** | **Right Side Wheels** | **Direction** |
|---|---|---|

| | | |
|---|---|---|
| Forward | Forward | Front |
| Forward | Backward | Right |
| Backward | Forward | Left |
| Backward | Backward | Back |

**Table 4 - Wheel Direction Combination Pattern**

To control the robot, the doctor can use a user-friendly interface provided by the IoT-based panel. The interface allows the doctor to move the robot in any direction using a joystick or a set of arrow keys. The doctor can also adjust the speed of the robot using the same interface.

The mobile or tablet mounted on the robot is used to hold live video calls. The doctor can communicate with the patient through the video call and can see the patient's condition in real-time. The doctor can also use the video call to guide the patient in case the patient needs any assistance.

The robot also has an IR sensor attached to it which helps to measure the temperature of the patient if needed and that too without any contact. The Infrared Sensor sends the data to the Database which can be accessible by the doctor.

In summary, the project uses a robotic vehicle with 4-wheel drive, a controller box for circuitry, and a mobile or tablet for live video calls. The doctor can control the robot using an IoT-based panel, and the received commands are processed in real-time to operate the robot motors.

### 3.3 CIRCUIT DIAGRAM AND CONNECTIONS

- The below table shows the circuit pin connections between the MLX90614 IR Temperature sensor, piezo buzzer and the NODE-MCU ESP8266 board.

| MLX90614 IR Temperature Sensor | NodeMCU ESP8266 Pins |
|---|---|
| SDA | D2 |
| SCL | D1 |
| Vcc | 3.3V |
| GND | GND |

**Table 5 - Circuit Pin Connections of Temperature Sensor**

- The below figure shows the circuit pin connections between the L298N motor driver, Dc motors and Esp8266 and the NODE-MCU ESP8266 board.

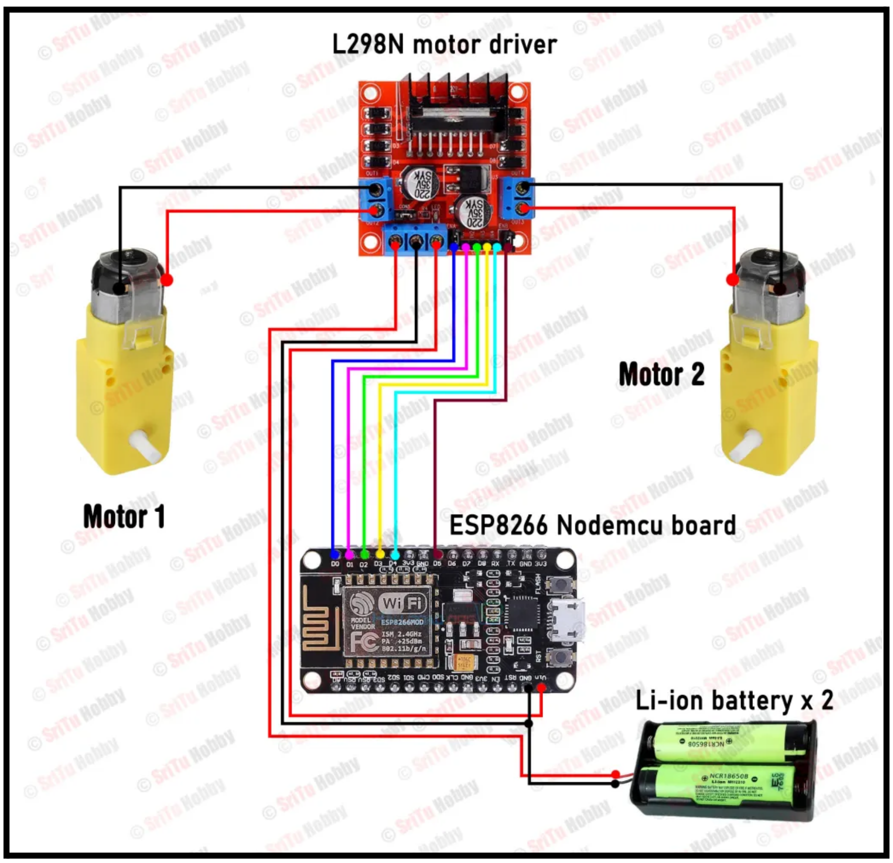



**Figure 7 - Connection of IOT Robot**

# CHAPTER - 4 RESULTS AND CONCLUSION

## 4.1 RESULTS

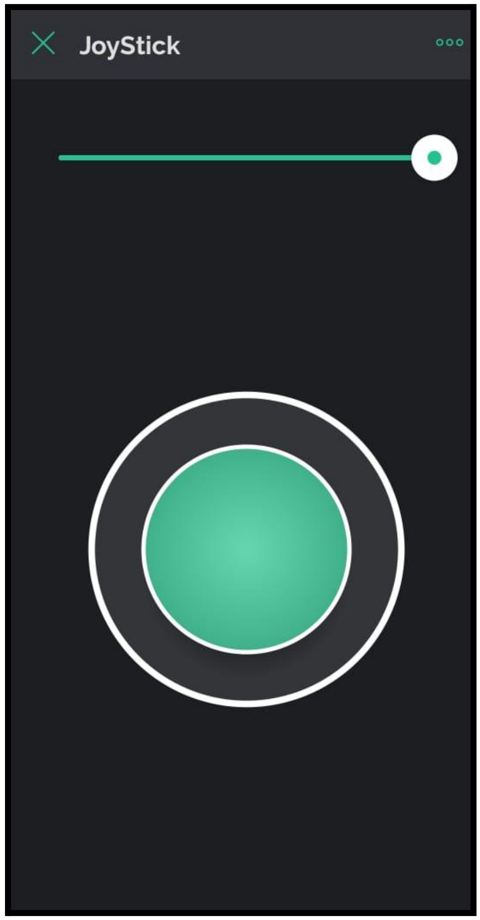



**Figure 8 - Blynk IOT Robot Controller**

**Figure 9 - Temperature Sensor Connections**

**Figure 10 - Video Call feature on Shaft**

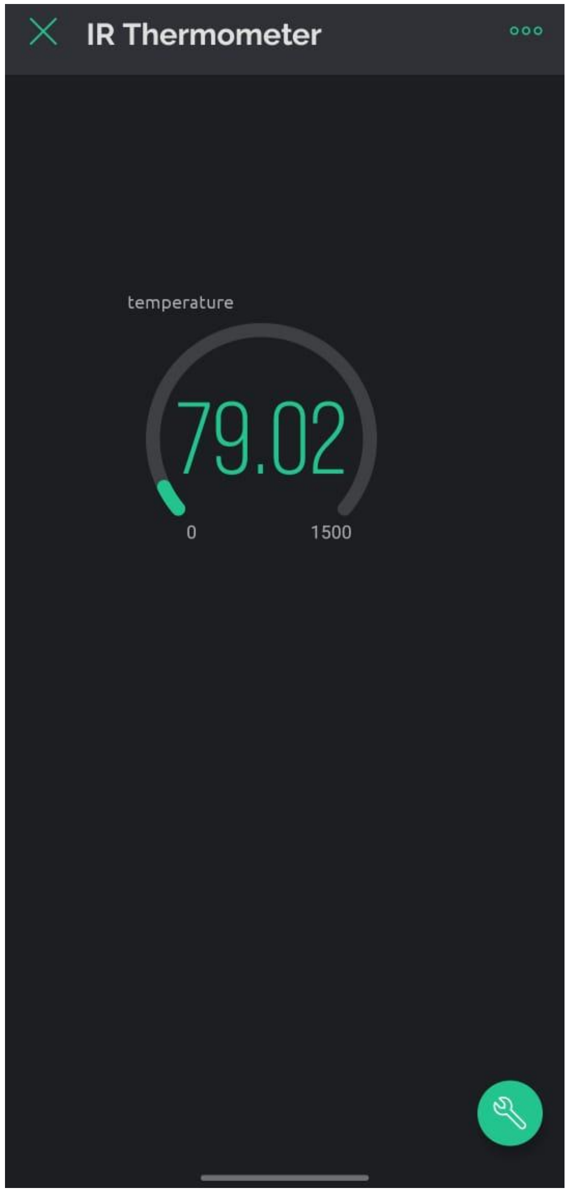



**Figure 11 - Blynk IOT Temperature Sensor**

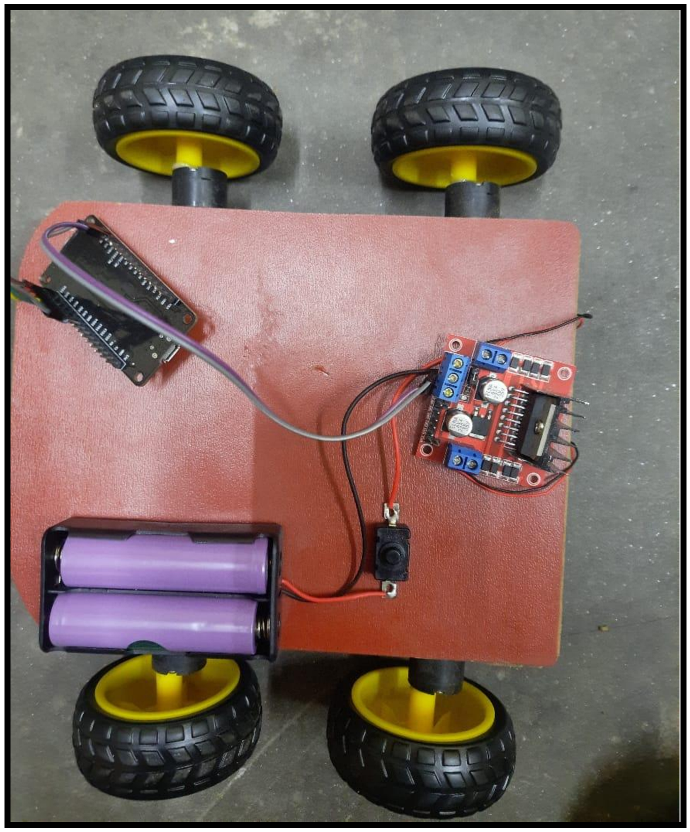


**Figure 12 - Four wheel drive connections**

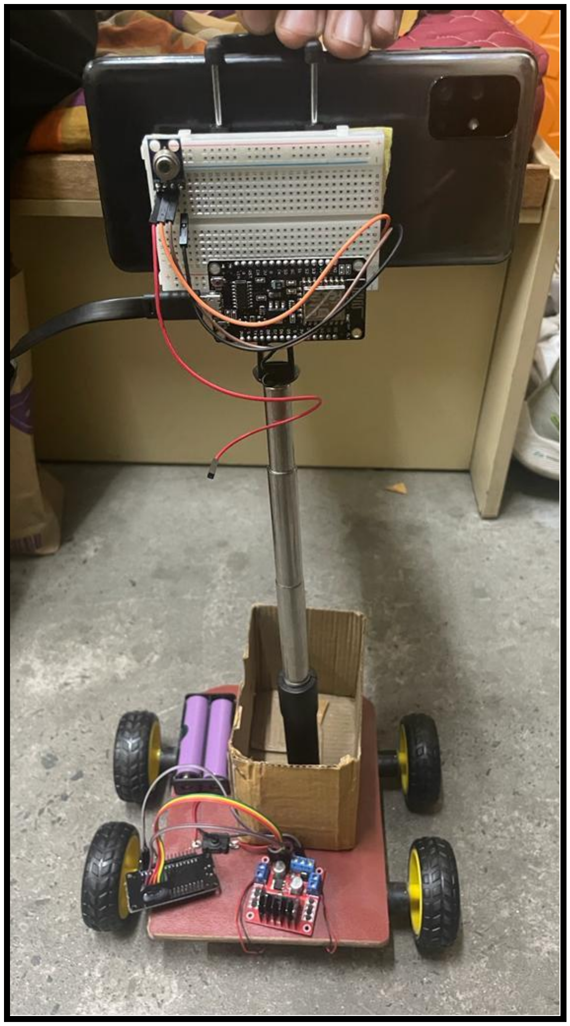


**Figure 13 - Final Display (BackView)**

**Figure 14 - Final Display (FrontView)**

## 4.2 FEATURES

- Robotic vehicle with 4 wheel drive: The robotic vehicle is equipped with four-wheel drive, which makes it easier to navigate through different terrain types. This feature enables the robot to move efficiently and effectively in a variety of environments.
- Controller box for circuitry: The robot includes a controller box that houses the circuitry needed to operate the robot. This feature allows for easy access to the robot's electronics, making it easier to diagnose and repair any issues that may arise.
- Mounting for mobile phone or tablet: The robot includes a mounting system for a mobile phone or tablet. This feature enables live video calls to be made, allowing doctors to remotely communicate with patients and conduct virtual examinations.
- IOT based panel for remote control: An IOT based panel is used to remotely control the robot. This feature allows doctors to operate the robot from a remote location, enabling them to reach patients who are not able to travel to the hospital or clinic.
- Wi-Fi internet connectivity: The robot controller operates over Wi-Fi internet. This feature allows for real-time communication between the robot and the doctor, enabling remote control of the robot and real-time video communication.
- Real-time movement commands: The received commands are received in real time, and the robot's motors are operated to achieve the desired movement commands. This feature enables the doctor to remotely control the robot's movements in real time, making it easier to navigate through different environments and reach patients who are not able to travel.

- Temperature Sensor: The IR Sensor attached to it measures the temperature of the patient if needed and sends it to the Database and the Blynk IOT application which is being controlled by the Doctor or the handler.

Overall, these features make the telemedicine robot an effective tool for remote healthcare delivery, enabling doctors to remotely examine and treat patients in real time.

**4.3 FUTURE SCOPE**

The telemedicine robot project has several future scopes that can be explored and expanded upon. Some of the potential future scopes of the project include:

- Autonomous Navigation: One of the future scopes of the project is to develop autonomous navigation capabilities for the robot. This feature would enable the robot to navigate through different environments and obstacles without the need for remote control.
- AI-based Diagnosis: The robot could be integrated with Artificial Intelligence (AI) algorithms to enable it to perform basic diagnosis of patients' conditions. This feature would enable the robot to provide a preliminary diagnosis to the doctor, helping to save time and improve patient outcomes.
- Augmented Reality (AR) Integration: Another future scope of the project is to integrate augmented reality technology into the robot. This feature would enable doctors to see the patient's medical data and vital signs in real-time, overlaid on the live video stream, providing doctors with additional information to make more informed decisions.
- Multi-Patient Monitoring: The robot could be used to remotely monitor multiple patients simultaneously. This feature would enable doctors to manage and monitor patients in real

time from a remote location, improving patient outcomes and reducing the need for physical hospital visits.

- Telemedicine Platform Integration: The telemedicine robot project could be integrated with a telemedicine platform, enabling doctors to remotely consult with patients in real time using a variety of tools and technologies. This feature would provide a complete telemedicine solution for healthcare providers, enabling them to reach patients who are not able to travel to the hospital or clinic.

Overall, the telemedicine robot project has significant potential for future development and expansion, enabling remote healthcare delivery and improving patient outcomes. As technology continues to evolve, new features and capabilities will become available, further enhancing the telemedicine robot's usefulness and effectiveness.

# REFERENCES


- Maria Nicola. “The socio-economic implications of the coronavirus pandemic (COVID-19): A review”. Elsevier Public Health Emergency Collection (2020).
- Melexis, MLX90614 family, Datasheet Single and Dual Zone InfraRed Temperature, (Revision 13 September 2019)
- https://www.wikipedia.org/
- https://techatronic.com/how-to-make-metal-detector-metal-detector-using-arduino-techatronic/
- https://circuitdigest.com/microcontroller-projects/arduino-metal-detector
- https://srituhobby.com/how-to-make-a-wifi-control-car-with-the-new-blynk-app-step-by-step/

# APPENDIX : Codes

1. **Temperature Sensor**

```
#include <Wire.h>

#include <Adafruit_MLX90614.h>

#include <Adafruit_GFX.h>

#include <Adafruit_SSD1306.h>

#define BLYNK_PRINT Serial

#include <Blynk.h>

#include <ESP8266WiFi.h>

#include <BlynkSimpleEsp8266.h>


Adafruit_MLX90614 mlx = Adafruit_MLX90614();


double temp_amb;

double temp_obj;

double calibration = 2.36;


char auth[] = "teccEkrhvziuYpT3vnETyUejty1NMMLI";     // You should get Auth Token in the
Blynk App.

char ssid[] = "esp";                       // Your WiFi credentials.

char pass[] = "12345678";


void setup()
```

```
{
  Serial.begin(9600);
  mlx.begin();         //Initialize MLX90614
    //display.begin(SSD1306_SWITCHCAPVCC, 0x3C); //initialize with the I2C addr 0x3C
(128x64)
  Blynk.begin(auth, ssid, pass);

  Serial.println("Temperature Sensor MLX90614");

  delay(2500);
}

void loop()
{
  //Reading room temperature and object temp
  //for reading Fahrenheit values, use
  //mlx.readAmbientTempF() , mlx.readObjectTempF() )
  Blynk.run();
  temp_obj = mlx.readObjectTempF();

  //Serial Monitor

  Serial.print("Object temp = ");
```

```
  Serial.println(temp_obj);

  Blynk.virtualWrite(V2, (temp_obj + calibration));

  delay(1000);
}
```

2. **4 - Wheel Drive Robot**

```
#include <dummy.h> // Include the library files
#define BLYNK_PRINT Serial
#include <ESP8266WiFi.h>
#include <BlynkSimpleEsp8266.h>
// Define the motor pins
#define ENA D0
#define IN1 D1
#define IN2 D2
#define IN3 D3
#define IN4 D4
#define ENB D5

// Variables for the Blynk widget values
int x = 50;
int y = 50;
```

```
int Speed;

char auth[] = "_alsZTFxQqKiaK7G4-L_1LG3Pxj3VRkW"; //Enter your Blynk auth token
char ssid[] = "esp"; //Enter your WIFI name
char pass[] = "12345678"; //Enter your WIFI passowrd

void setup() {
  Serial.begin(9600);  //Set the motor pins as output pins
  pinMode(ENA, OUTPUT);
  pinMode(IN1, OUTPUT);
  pinMode(IN2, OUTPUT);
  pinMode(IN3, OUTPUT);
  pinMode(IN4, OUTPUT);
  pinMode(ENB, OUTPUT);

  // Initialize the Blynk library
  Blynk.begin(auth, ssid, pass, "blynk.cloud", 80);
}

// Get the joystick values
BLYNK_WRITE(V0) {
  x = param[0].asInt();
}
```

```
// Get the joystick values
BLYNK_WRITE(V1) {
  y = param[0].asInt();
}
//Get the slider values
BLYNK_WRITE(V2) {
  Speed = param.asInt();
}


// Check these values using the IF condition
void smartcar() {
  if (y > 70)
{
    carForward();
    Serial.println("carForward");
  }
else if (y < 30) {
    carBackward();
    Serial.println("carBackward");
  }
else if (x < 30) {
    carLeft();
    Serial.println("carLeft");
```

```
  }
else if (x > 70) {
    carRight();
    Serial.println("carRight");
  }
else if (x < 70 && x > 30 && y < 70 && y > 30) {
    carStop();
    Serial.println("carstop");
  }
}
void loop() {
  Blynk.run();// Run the blynk function
  smartcar();// Call the main function
}

/**************Motor movement functions*****************/
void carForward() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
```

```
}
void carBackward() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, HIGH);
}
void carLeft() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, HIGH);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, HIGH);
  digitalWrite(IN4, LOW);
}
void carRight() {
  analogWrite(ENA, Speed);
  analogWrite(ENB, Speed);
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, HIGH);
  digitalWrite(IN3, LOW);
```

```
  digitalWrite(IN4, HIGH);
}
void carStop() {
  digitalWrite(IN1, LOW);
  digitalWrite(IN2, LOW);
  digitalWrite(IN3, LOW);
  digitalWrite(IN4, LOW);
}
```