IOT SOCIAL DISTANCING AND MONITORING ROBOT FOR QUEUE

Project report submitted in partial fulfillment of the requirement for the degree of

BACHELOR OF TECHNOLOGY

IN

ELECTRONICS AND COMMUNICATION ENGINEERING

By

Surbhi Kumari (181010)
Kirti Parshionikar (181024)

UNDER THE GUIDANCE OF

Dr. Vikas Baghel

JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY, WAKNAGHAT

May 2022

TABLE OF CONTENTS

DECLARATION

We hereby declare that the work reported in the B.Tech Project Report entitled **"IOT Social Distancing & Monitoring Robot For Queue"** submitted at **Jaypee University of Information Technology, Waknaghat, India** is an authentic record of our work carried out under the supervision of **Dr. Vikas Baghel.** We have not submitted this work elsewhere for any other degree or diploma.

Surbhi Kumari	**Kirti Parshionikar**
(181010)	**(181024)**

CERTIFICATE

This is to certify that the content of this project entitled **"IOT Social Distancing And Monitoring Robot For Queue"** by Surbhi Kumari and Kirti Parshionikar is their project work carried out under my supervision, of the Electronics and Communication branch at Jaypee University Of Information Technology, Solan, HP in the academic year 2021-22 for the fulfillment of the requirements of B. Tech Major Project.

This work has not been submitted partially or wholly to any other university or institute of award of this or any other degree or diploma.

Signature of the Supervisor

Dr. Vikas Baghel
Department of Electronics and Communication Engineering,
Jaypee University Of Information Technology,
Waknaghat,Solan,H.P.
INDIA

ACKNOWLEDGEMENT

We would like to express our heartfelt gratitude to everyone who has inspired and driven us to see this project through to completion.

We would like to express our gratitude to our supervisor, **Dr.Vikas Baghel,** for providing a conducive environment for the initiative to continue. We would like to use this opportunity to express our gratitude to **Dr. Rajiv Kumar**, Head of the **Department of Electronics and Communication** for his invaluable assistance in the development of this project. We are also grateful to the rest of the **Department of Electronics and Communication** team for their unwavering support.

Our heartfelt gratitude to all of the well-wishers and friends who have supported us during the project development and contributed to make it a huge success.

At last, we would also like to thank **Jaypee University Of Information Technology** which provides all the necessary facilities to ensure a good environment for studies.

Thank you for your support.

LIST OF ACRONYMS AND ABBREVIATIONS

IoT	Internet of Things
PIR	Passive infrared
LED	Light Emitting Diode
USB	Universal Serial Bus
MCU	Micro controller unit

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

Since the COVID-19 epidemic has spread around the globe, it has become a major health as well as a communities safety issue and the rise of technology has seen a major surge, the plan to use robots to reduce the burden on the systems and improve it. We have to co-exist with this virus for quite a long time. Among many precautions which need to be taken care of, the most effective and important measure is to control the maintenance of Social distancing, it stops the unfold of the infectious diseases and minimizing close physical contact, minimizes the contracting and spreading of the virus. Some people who do not understand the seriousness tend to break the rule if they think someone is not watching them over. This project aims to solve the problem by mechanically observing a queue in spaces like malls or hospital. If the person is not maintaining two metres distance then it works by alarming them and the a picture is taken and is sended over Wifi with the help of IoT to the respective authority. It makes sure to keep in check for the places and observe people.

CHAPTER 1
INTRODUCTION

1.1 Introduction

COVID19 has become a major health concern and safety issue for communities and healthcare organisations all over the globe since its emergence. Minimizing intimate physical contact is the most effective important step in reducing viral spread. Banks, theatres, schools, and government buildings, for example, cannot be secured 24 hours a day, seven days a week by security personnel. Taking into account the social distance convention, our proposal intends to create a one-of-a-kind solution for mechanically stationing people in lines. The robot is made out of a frame structure with four wheels that allows it to move around using the trail tracking principle. Two ultrasonic sensors are installed to detect obstacles in its path and one to assess the distance between two people.If the distance is less than 2m, the robot will send the camera photos to the person in charge of the workplace through WiFi module via Internet of Things.

1.1.1 Iot and its applications

IoT simply refers to a network of things that are connected to the internet; these devices range from consumer electronics to high-tech sensors.

IoT focuses on action and interaction behaviour, and combining the two adds significant value.

These cognitive components are found in IoT paired with Robots:

The robot begins by gathering information from sensors and other sources.

Second, it analyses and monitors data before processing it on the edge, eliminating the need to send large amounts of data to the cloud.

These two components assist the robot in determining which action to take, resulting in physical item manipulation and completion of the task at hand. These interactions lead us closer to bigger ideals of machine-human collaboration.

1.1.2 Objectives

Our goal is to use Internet of Things technology to automate observe persons in lines who do not adhere to social distancing standards and constraints. Over the internet/WiFi, the robot can be controlled by devices from anywhere at any time. It effectively assures that no touch is made during the operation.

• Collect the most important system needs

• Research the system's platform requirements.

• Determine which development languages, technologies, and tools are appropriate.

• Evaluate the interface methods

• Set up the Raspberry Pi

• A dc motor interface board

• Test and evaluate the system

• Maintaining the system

1.2 Literature Survey

These papers are based on a comprehensive literature analysis and experimentation, and they indicate some of the most efficient feasible technologies and algorithms.

[1]NOVEL ECONOMICAL SOCIAL DISTANCING SMART DEVICE FOR COVID19 by Rahul Reddy Nadikattu,Sikender Mohsienuddin Mohammad and Dr.Pawan Whig.

This research study is built on the PIR sensor principle, and it proposes an unique localization approach for tracking humans' position in an outdoor setting using sensors. The thermal conductivity of the infrared received is calibrated, and the body temperature is computed. The device will inform you if someone is within six feet of another person, thanks to artificial intelligence. The strategy is fairly precise and can be extremely helpful in preserving social distance.

[2]IoT SOCIAL DISTANCING & MONITORING ROBOT FOR QUEUE by Aditi Vijay, Ashutosh Gupta, Ashwani Pal, B. Sriswathi, Geetika Mathur, Satish Alaria

To detect impediments in the vehicle path, the robot consist two ultrasonic sensor for obstacle detection in this paper. It determines the distance between two persons using an ultrasonic sensor. The robot beeps and warns of injuries when the space between individuals is less than two metres. It also delivers these violation notices and video images to higher departments/key workplaces through WiFi via the IoT if there are indications that there have been violations

[3]MONITORING SOCIAL DISTANCING CONSTRAINTS IN CROWDED SCENARIOS by Adarsh Jagan Sathyamoorthy, Utsav Patel, Yash Ajay Savle, Moumita Paul and Dinesh Manocha

It employs a mobile robot with sensors, a camera, and a 2-D lidar to navigate through a crowd without colliding and calculate the distance between all observed humans in the camera's field of view. It also included thermal camera on the robot, which wirelessly feeds thermal images to a security/healthcare professional who watches the situation.

[4]SOCIAL DISTANCE MONITORING APPROACH USING WEARABLE SMART TAGS by Tareq Alhmiedat and Maed Aborokbah

Based of human detection and proximity distance function the proposed smart monitoring system consist of a new smart wearable prototype of minimal cost device that estimate social distance between people and issues notification when the social distance is less than a pre-defined threshold. The planned system was to put to test and found to have 96.1 percent acceptance rate and a 6 metre localization error.

CHAPTER 2
REQUIREMENT SPECIFICATION

2.1 Functional

The current system's functions and services are the focus of functional requirements. It contains information on the system.

• Measure distance

• Detect obstacles

• Identify violators

• Calculate data

• Send notifications

2.2 Non fuctional

The project's non-functional needs are as follows:

• The proposed system is simple to use and may be accessed at any time. It has the ability to offer data at any time.

• The system will always produce precise measurements, and its lifespan will be extended.

• The proposed system is simple to maintain, and we can easily add new functionality by merging components.

• Because this system is user-friendly, no prior expertise is required. All we need to do now is learn how to use it.

• The most significant aspect is that it is inexpensive. We can start with a small budget.

2.3 Hardware Requirements

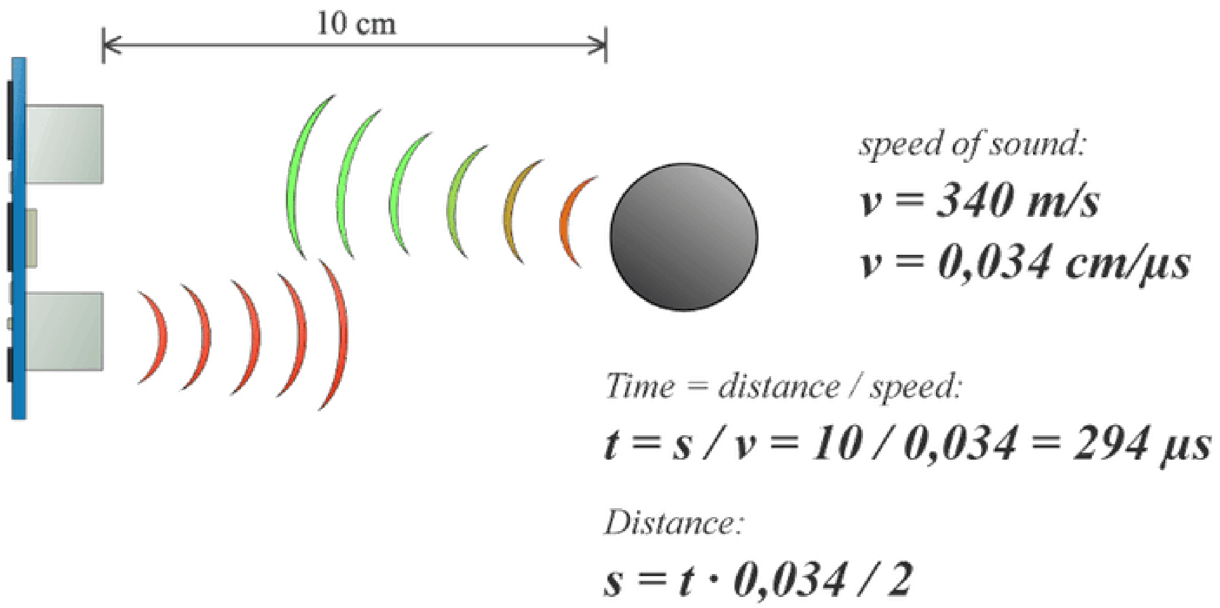
For Developing hardware we require following components:-

2.3.1 Ultrasonic Sensor:

Ultrasonic sensor is an electrical device that measures the gap or distance between two objects using ultrasonic sound waves. Ultrasonic sound waves are significantly higher than human hearing, which ranges from 20kHz to 40kHz.

It comprises of a transmitter that transmits a 40kHz ultrasonic pulse through the air, which detects an object or obstruction and bounces back to the receiver, which calculates the distance using the speed of sound and the trip time.

They're mostly employed as proximity sensors because they move faster than audible sounds.



It is simply based on the ultrasonic sensor formula.

However, while the formula is the same, the readings are not, as we will discover later in the project. When the object passes through the sensors, the programme calculates the speed, and approximately 8 inches distance between the two sensors. Therefore, in the programme, we will use (fps=666666/12) to calculate frames per second. The inches of a foot are 12 inches, and 8/12 is 666666.

Both sensors are active at the same time, calculating gap. If the gap between two is less than 100 metres, then the sensor with the lowest value illuminates and a notification is sent to the violator's phone as well as the person's image is recorded in rpi camera module.

The sensor has following specifications:

Operating Voltage	DC-5V
Operating Current	15mA
Operating Frequency	40KHZ
Farthest Range	4m
Nearest Range	2cm
Measuring Angle	15 Degree
Input Trigger Signal	10us TTL pulse
Output Echo Signal	Output TTL level signal, proportional with range
Dimensions	45*20*15mm

Table 2.1 Ultrasonic Sensors specifications

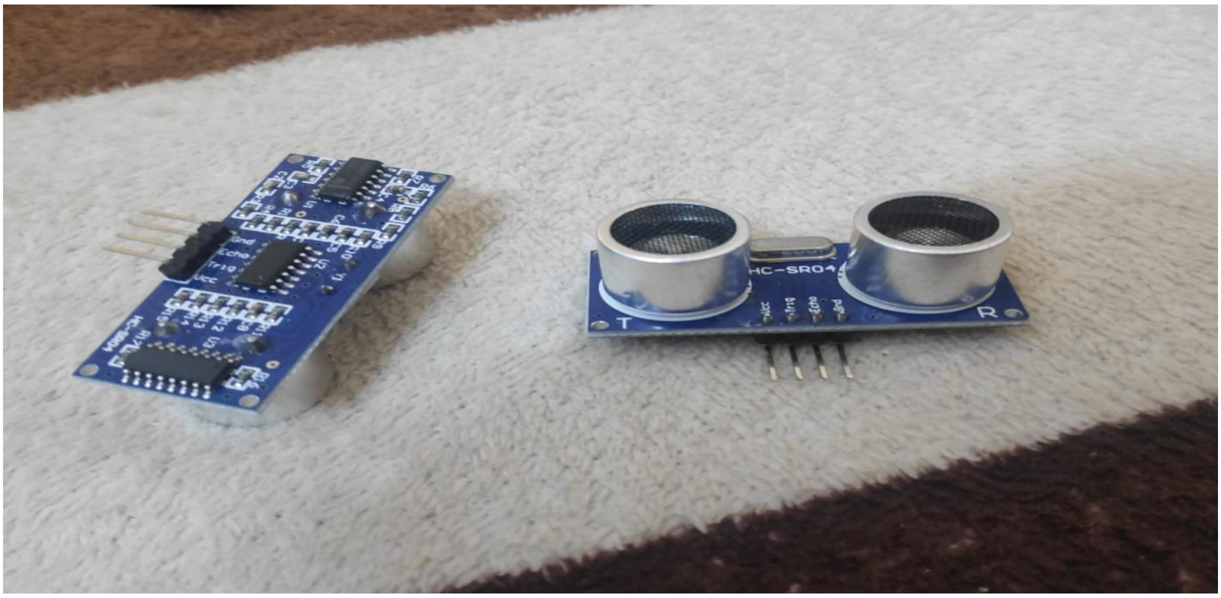


Figure 2.1: Ultrasonic Sensors

2.3.2 Raspberry Pi 3:

In the United Kingdom the Raspberry Pi Foundation collaborated with Broadcom to create a line f small single-board computer. It’s a credit card size computer with a standard keyboard and mouse that conncets to a computer monitor or television. It’s a competent tiny device that enables people of all ages to learn about computing and programming with scratch and python. It can browse the internet and stream high definition video, as well as spreadsheets, word processing and gaming just like a desktop computer.

Specifications	Raspberry Pi 3 Model B
Processor Chipset	1.2GHz 64-bit quad core ARMv8
RAM	1GB
Storage	Micro SD
USB 2.0	4 Ports
Ethernet Port	Yes
Wireless LAN	802.11n
Bluetooth	Yes
GPIO	40 pins
Power Draw/Voltage	1.8A at 5V

Table 2.2 Raspberry Pi specifications

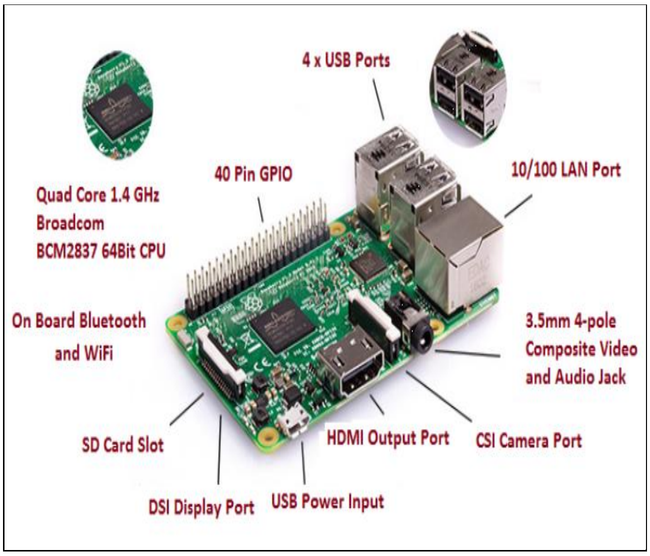


Figure 2.2: Raspberry Pi

Pin configuration check in appendix (figure A6)

2.3.3 ESP8266 NodeMcu:

On the NodeMCU ESP8266 development board, the ESP-12E module has an ESP8266 chip with a Tensilica Xtensa 32-bit LX106 RISC microprocessor. This microprocessor supports RTOS and runs at a customizable clock frequency of 80MHz to 160MHz. The NodeMCU has 128 KB of RAM and 4MB of Flash memory for storing data and programmes. Its high processing power, built-in Wi-Fi / Bluetooth, and Deep Sleep Operating characteristics make it ideal for IoT projects.

- **Specifications and features of the NodeMCU ESP8266**

Input Voltage: 7-12V

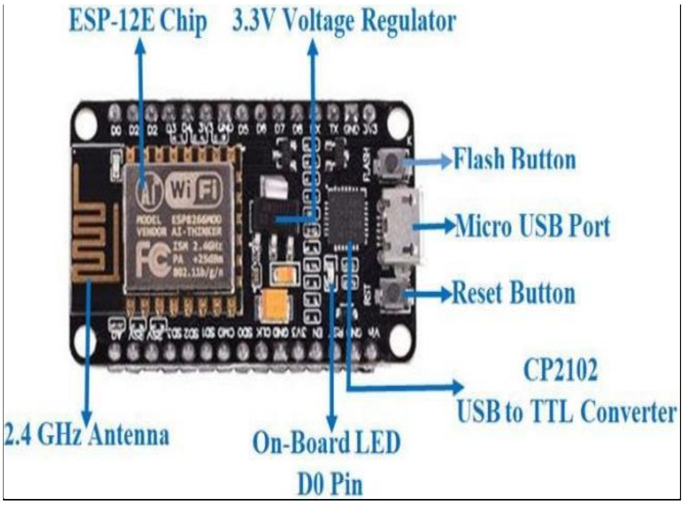
Operating Voltage: 3.3V

Analog Input Pins (ADC): 1

Digital I/O Pins (DIO): 16

SRAM: 64 KB

Flash Memory: 4 MB



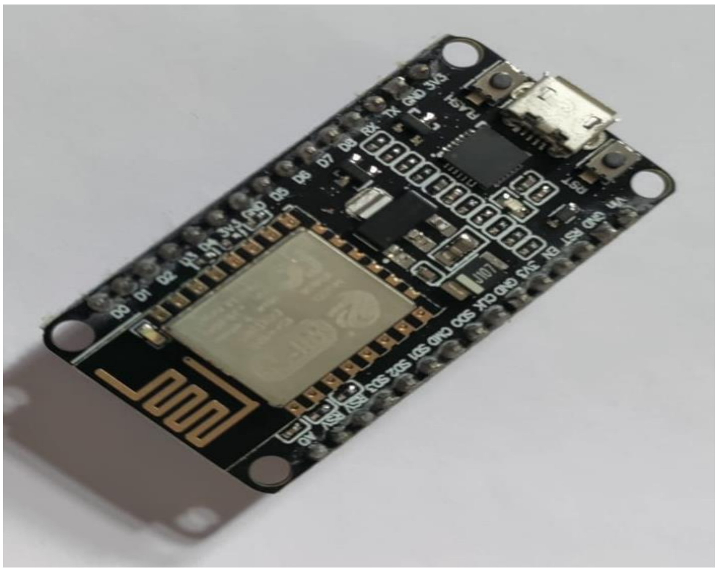


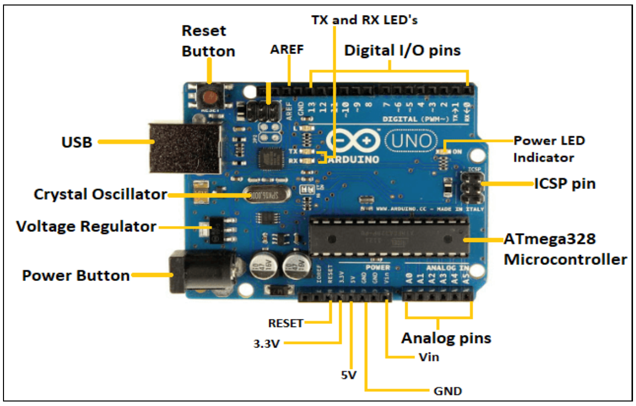
Figure 2.3: NodeMcu

2.3.4 Arduino Uno:

The Arduino UNO uses the ATmega328P microprocessor. It is simple to use in comparison to other boards such as the Arduino Mega. The board is made up of digital and analogue input/output pins (I/O), shields, and other electronics. There are six analogue pin inputs, fourteen digital pins, a USB connector, a power jack, and an ICSP connector (In-Circuit Serial Programming) header are all included in the Arduino UNO. It is written in the IDE (Integrated Development Environment) programming language. It is available for both online and offline use.

Microcontroller	ATmega328
Clock Speed	16MHz
Operating Voltage	5V
Maximum supply Voltage (not recommended)	20V
Supply Voltage (recommended)	7-12V
Analog Input Pins	6
Digital Input/Output Pins	14
DC Current per Input/Output Pin	40mA
DC Current in 3.3V Pin	50mA
SRAM	2KB
EEPROM	1KB
Flash Memory	32KB of which 0.5KB used by boot loader

Table 2.3 Arduino specifications



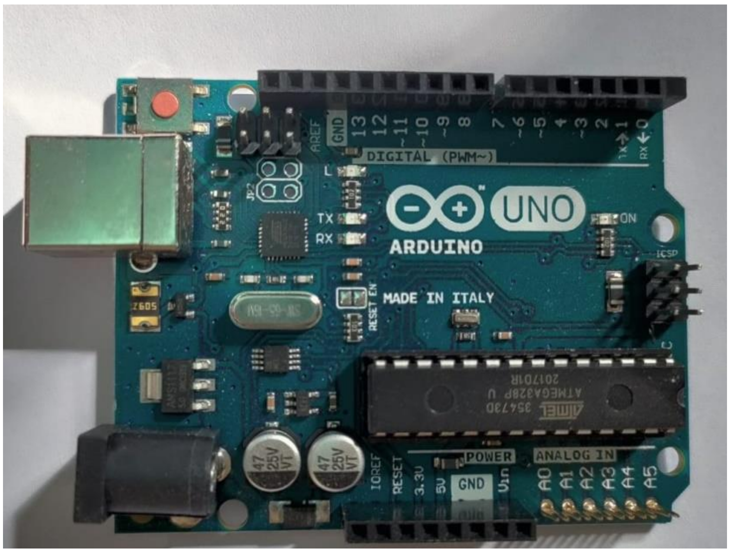
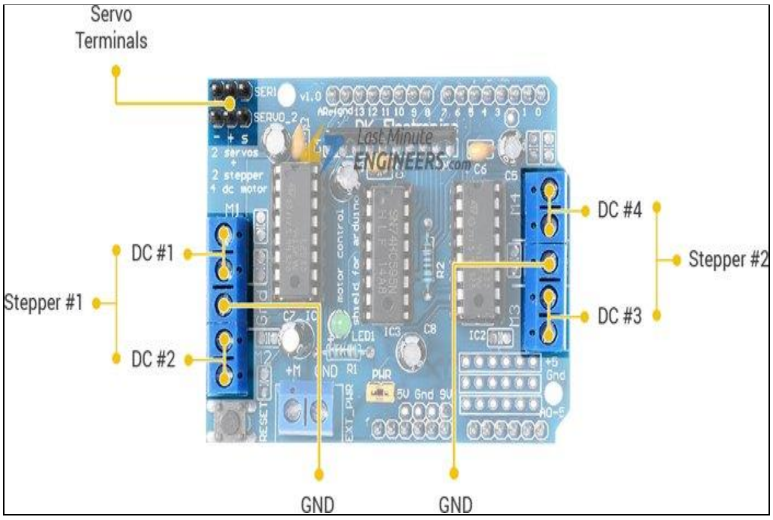


Figure 2.4: Arduino Uno

Pin configuration check in appendix (figure A7)

2.3.5 Arduino motor shield:

The Arduino Motor Shield lets you control loads that aren't controlled by a typical Arduino pin. The motor shield includes a number of useful features, including current measurement and the ability to control a single stepper motor. The L298P dual full bridge driver, which can withstand up to 3 amps for extremely brief periods of time or 2 amps constantly per channel, lies at the heart of this shield. The included zipped folder contains comprehensive example code for all parts. It helps drive DC motor with arduino board and control the speed as well as direction of both independently.



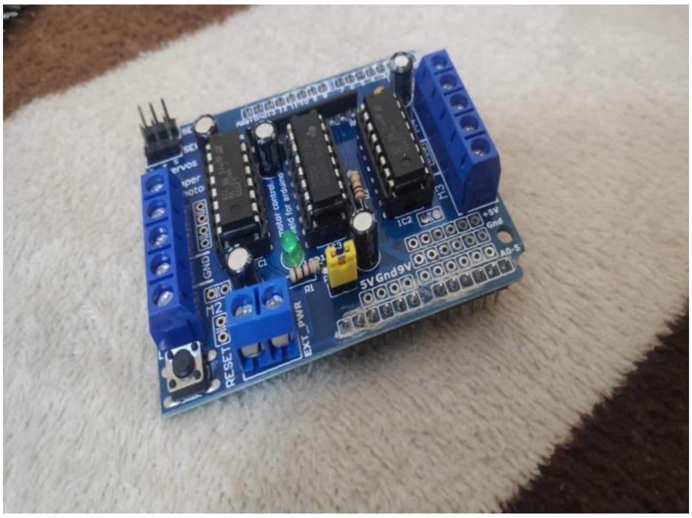


Figure 2.5: Arduino Motor Shield

Pin configuration check in appendix (figure A6)

2.3.6 Raspberry pi Cam:

The Raspberry Pi is a hardware expansion module.

The Raspberry Pi Camera Board is a Raspberry Pi add-on module that was built specifically for the Raspberry Pi. A unique CSI interface links it to the Raspberry Pi. The sensor's native resolution is 5 megapixels when used in still capture mode. In video mode, it can capture 1080p video at 30 frames per second.

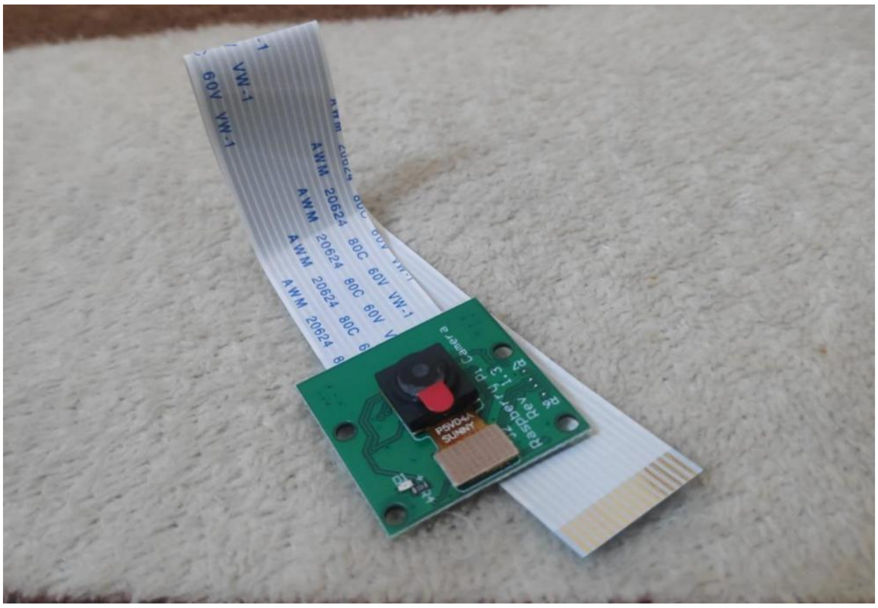


Figure 2.6: Raspberry Pi cam module

2.3.7 LED's:-

When current passes through a crystal rectifier, light is emitted. It's a light source made of semiconductors. If current flows through an LED, electrons and holes combine together and light is produced.

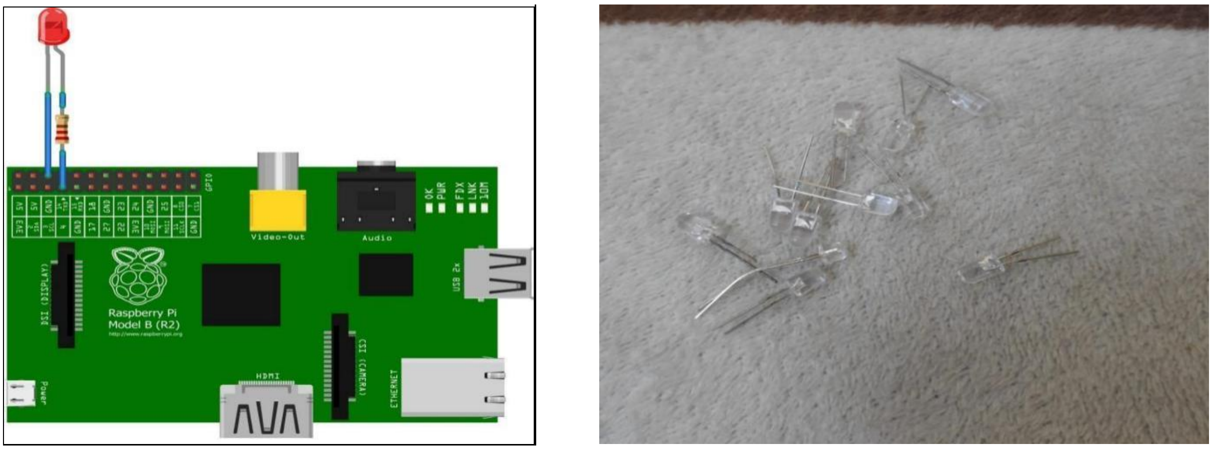


Figure 2.7: LED

2.4 Software

This Social Distancing Bot is created with the Raspberry Pi, and it requires the Arduino IDE platform as well as Python to operate. IDE is a piece of software that lets you write the code and upload it to the board. It uses C and C++ to install and operate on Windows.

Python app
pyqt5 library for api connections are required.

CHAPTER 3
DESIGN OF THE SYSTEM

3.1 Hardware Designing

The robot is made up of a four-wheeled frame system that is utilised to run the robotic vehicle. It employs the tail tracking approach to continuously check the distance and track social distance-violating behaviours. For detecting obstructions in the vehicle path, the robot is now outfitted with ultrasonic sensor for obstacle detection. Another ultrasonic sensor is used by the robotic vehicle to assess the distance between two persons. The robot warns of injuries when the space between individuals is less than two metres and along with that it also sends notification to the security personel along with image.

3.1.1 Block diagram

The Block diagram is as follows:

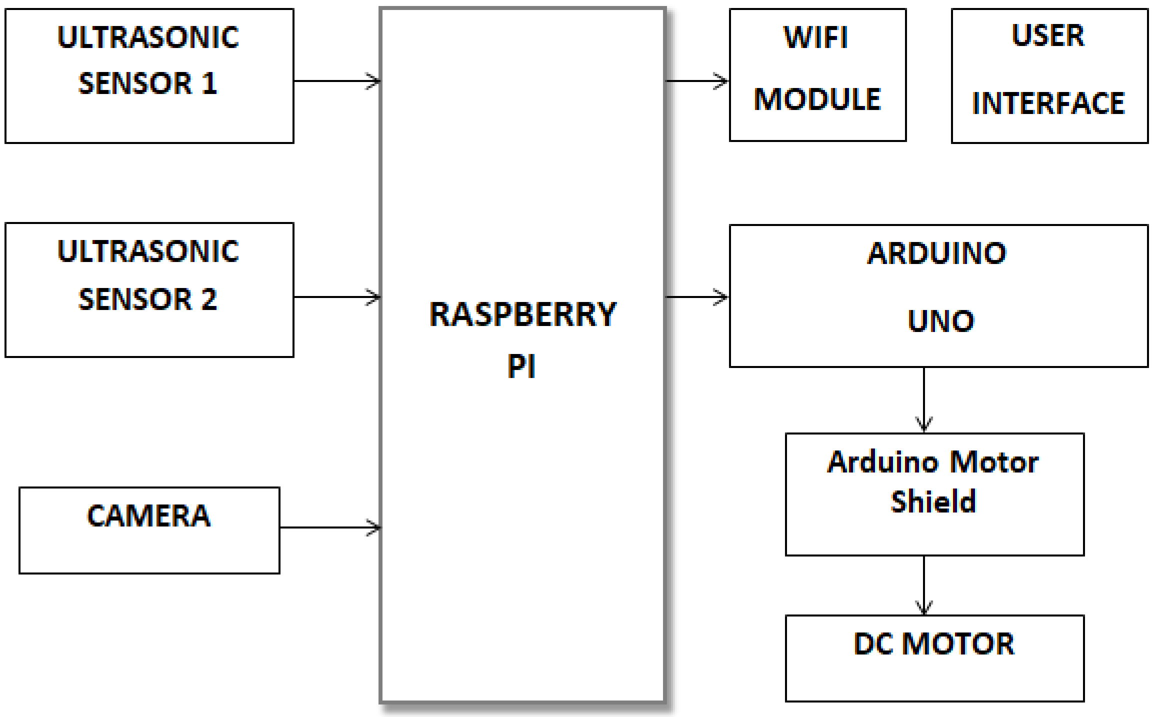


Figure 3.1: IoT Architecture

3.1.2 Circuit Design

The circuit diagram is as follows:

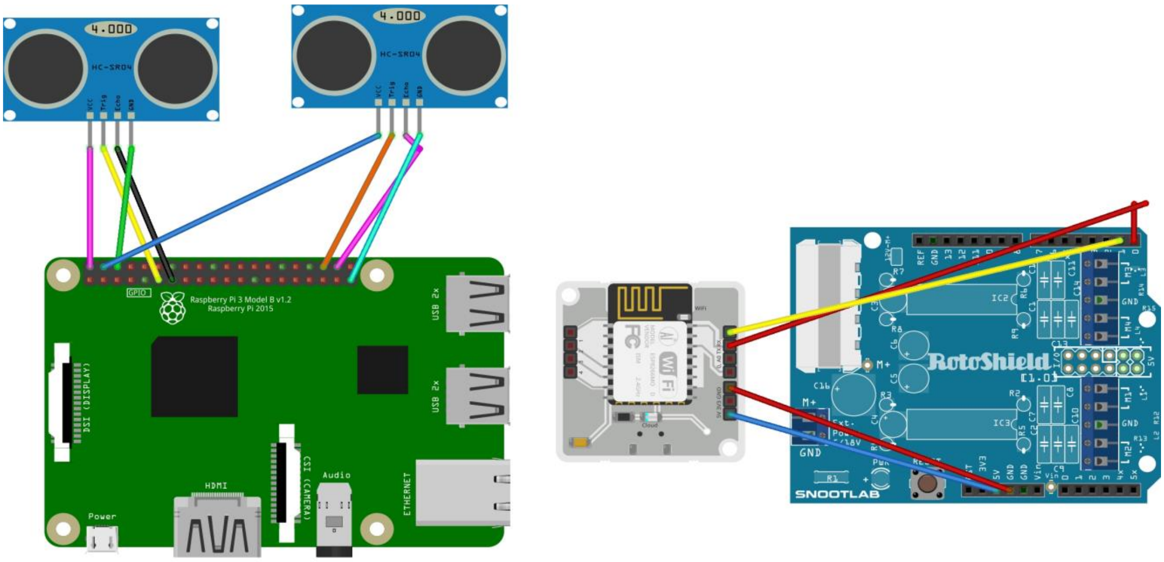


Figure 3.2: Circuit Diagram

3.2 Interfacing

1.) Use a USB cord to connect the Raspberry Pi to the Arduino and use the les/devttyACM0 port for serial communication between the Raspberry Pi and the Arduino.

2)Wifi rx with Arduino tx and vice versa

3)For the ultrasonic sensor in front:

GPIO26 TRIG PIN

GPIO19 ECHO PIN

4)For ultrasonic backward sensor:

GPIO17 TRIG PIN

GPIO27 ECHO PIN

5)Connect the arduino shields M1, M2, M3, M4 to the appropriate bo motors.

6)Connect the forward and backward leds to the appropriate ultrasonic sensor.

(LEDs are not required)

7)Connect the arduino shield's vcc and gnd to the wifi module.

8) Raspberry Pi camera moudle

9)Power supply:

a)10000mAh power bank

b)12v lipo battery

CHAPTER 4

Implementation

4.1 CODES

- **SETUP THE ARDUINO:**

```
#include <AFMotor.h>

AF_DCMotor motor1(1);
AF_DCMotor motor2(2);
AF_DCMotor motor3(3);
AF_DCMotor motor4(4);

void setup() {
  Serial.begin(9600); // set baud rate
  Serial.setTimeout(50);
  Serial.println("WELCOME++++");
  motor1.setSpeed(255);
  motor2.setSpeed(255);
  motor3.setSpeed(255);
  motor4.setSpeed(255);

  motor1.run(RELEASE);
  motor2.run(RELEASE);
  motor3.run(RELEASE);
  motor4.run(RELEASE);

}

void loop() {
String ch;
 if (Serial.available() > 0) {
    ch= Serial.readString();
    Serial.println(ch);
```

```
 if (ch=="F") {
 motor1.run(FORWARD);
 motor2.run(FORWARD);
 motor3.run(FORWARD);
 motor4.run(FORWARD);
}
if(ch=="B"){
 motor1.run(BACKWARD);
 motor2.run(BACKWARD);
 motor3.run(BACKWARD);
 motor4.run(BACKWARD);
}
 if(ch=="L"){
 motor1.run(FORWARD);
 motor2.run(BACKWARD);
 motor3.run(FORWARD);
 motor4.run(BACKWARD);
}
 if(ch=="S"){
 motor1.run(RELEASE);
 motor2.run(RELEASE);
 motor3.run(RELEASE);
 motor4.run(RELEASE);
}
 if(ch=="LF"){
 motor1.run(RELEASE);
 motor2.run(FORWARD);
 motor3.run(RELEASE);
```

```
 motor4.run(FORWARD);
}
 if(ch=="RF"){
 motor1.run(FORWARD);
 motor2.run(RELEASE);
 motor3.run(FORWARD);
 motor4.run(RELEASE);
}
  if(ch=="LB"){
 motor1.run(RELEASE);
 motor2.run(BACKWARD);
 motor3.run(RELEASE);
 motor4.run(BACKWARD);
}
  if(ch=="RB"){
 motor1.run(FORWARD);
 motor2.run(RELEASE);
 motor3.run(FORWARD);
 motor4.run(RELEASE);
}
  if(ch=="R"){
 motor1.run(BACKWARD);
 motor2.run(FORWARD);
 motor3.run(BACKWARD);
 motor4.run(FORWARD);
}
}
}
```

- **Setup the raspberry pi:**

For ultrasonic forward:

main.py

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)

TRIG=19
ECHO=26

GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG, False)
time.sleep(2)
def valb():
    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)
    while GPIO.input(ECHO)==0:
        pulse_start = time.time()
    while GPIO.input(ECHO)==1:
        pulse_end = time.time()
    pulse_duration = pulse_end - pulse_start
    distancem = pulse_duration*17150
    distance = round(distancem, 2)
    return distance
```

For ultrasonic back:

```
import RPi.GPIO as GPIO
import time
GPIO.setmode(GPIO.BCM)

TRIG=19
ECHO=26

GPIO.setup(TRIG,GPIO.OUT)
GPIO.setup(ECHO,GPIO.IN)

GPIO.output(TRIG, False)
time.sleep(2)
def valb():
    GPIO.output(TRIG, True)
    time.sleep(0.00001)
    GPIO.output(TRIG, False)
    while GPIO.input(ECHO)==0:
        pulse_start = time.time()
    while GPIO.input(ECHO)==1:
        pulse_end = time.time()
    pulse_duration = pulse_end - pulse_start
    distancem = pulse_duration*17150
    distance = round(distancem, 2)
    return distance
```

- **Main:**

```python
from ultrab import valb
from ultraf import valf
import RPi.GPIO as GPIO
import time
from picamera import PiCamera
from servercode import sendit
GPIO.setmode(GPIO.BCM)
ledb=21
ledf=22
GPIO.setup(ledb,GPIO.OUT)
GPIO.setup(ledf,GPIO.OUT)
while True:
    try:
        x=valb()
        print(x)
        y=valf()
        print(y)
        if x<=100 or y<=100:
            print("got it")
            camera=PiCamera()
            camera.capture("pritam.jpg")
            print("captured")
            if x>=y:
                GPIO.output(ledb, True)
                time.sleep(5)
                GPIO.output(ledb, False)
            if y>=x:
                GPIO.output(ledf, True)
                time.sleep(5)
```

CHAPTER 5

RESULT AND FUTURE WORK

5.1 Result

During the epidemic, the employment of robots has increased dramatically over the world in order to preserve contact-free operations and lessen the burden on communities. We might have to live with this infection for a long time. As a result, the IoT social distancing robot represents an approach to automation that is needed in a variety of industries. Taking into account all of the variables, this approach aids in the effective execution of the social distance norm, removing any hazards. As a consequence of the results, the robot informs persons who do not follow the rules, and an image of that person is submitted to the authorities. With the help of the internet, IoT connects all of the sensors. It will be easier for places to keep track of things with the help of this robot.

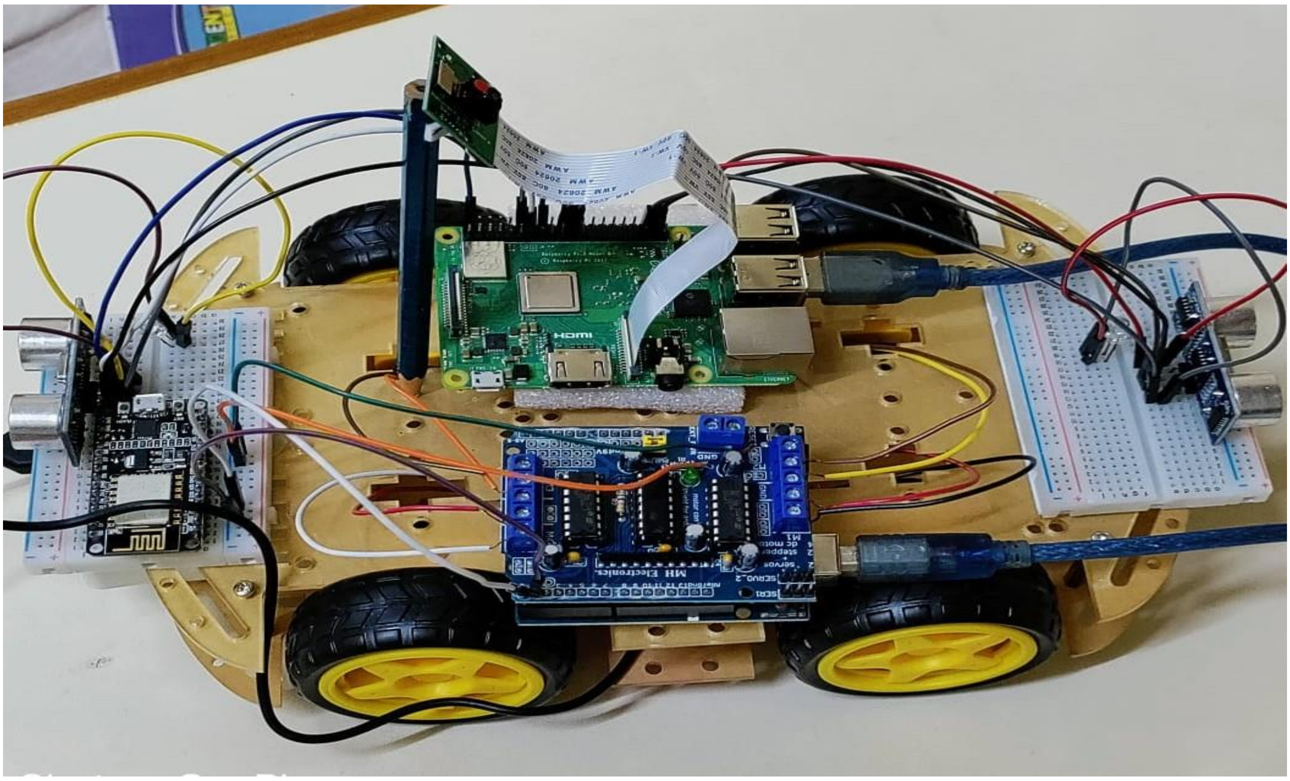


Figure 5.1: Robot

5.2 Future work

- During the epidemic, the usage of robots has increased significantly around There are numerous things to add to this project for future work:

  To detect faces, we can combine machine learning techniques with IoT.
  Adding a user interface for a mobile app.
  Although this bot is limited to queues, we can make it operate in congested situations.

REFERENCES

[1] Aditi Vijay, Ashutosh Gupta , Ashwani Pal , B. Sriswathi , Geetika Mathur, Satish Kumar Alaria "IoT SOCIAL DISTANCING & MONITORING ROBOT FOR QUEUE" International Journal of Engineering Trends and Applications (IJETA) – Volume 8 Issue 4, Jul-Aug 2021.

[2] Rahul Reddy Nadikattu,Sikender Mohsienuddin Mohammad,Pawan Whig" NOVEL ECONOMICAL SOCIAL DISTANCING SMART DEVICE FOR COVID19" International Journal of Electrical Engineering & Texhnology , June 2020.

[3] Adarsh Jagan Sathyamoorthy, Utsav Patel, Yash Ajay Savla, Moumita Paul and Dinesh Manocha "COVID-Robot: Monitoring Social Distancing Constraints in Crowded Scenarios", 21 August 2020.

[4] Siddiqui, Shama, Muhammad Zeeshan Shakir, Anwar Ahmed Khan, and Indrakshi Dey. "Internet of Things (IoT) Enabled Architecture for Social Distancing During Pandemic." Frontiers in Communications and Networks, 2021.

[5] Kumar, Adarsh, Kriti Sharma, Harvinder Singh, P. Srikanth, Rajalakshmi Krishnamurthi, and Anand Nayyar. "Drone-Based Social Distancing,Sanitization, Inspection, Monitoring, and Control Room for COVID-19." Artificial Intelligence and Machine Learning for COVID-19, 2021

[6] Fu, Shuangquan, and Pritesh Chandrashekhar Bhavsar. "Robotic arm control based on internet of things." In 2019 IEEE Long Island Systems, Applications and Technology Conference (LISAT), pp. 1-6. IEEE, 2019.

[7] Priyanka, P. "Iot Based Covid-19 Patient Health Monitoring System" Turkish Journal of Computer and Mathematics Education (TURCOMAT) 12, no. 12, pp. 4599-4604, 2021.

[8] Pooja Saini, Manish Mukhija, Prakash Dangi, "Design simulation of Efficient Character Recognition System Using Improved PreProcessing and Feature Extraction Process",Journal of The Gujrat Research Society,Vol.-21, Issue 16, pp. 1883-1889, ISSN: 0374-8588, December 2019.

[9] Ennafiri, M., and T. Mazri. "Internet of Things for Smart Healthcare: A Review on a Potential IOT Based System and Technologies to Control COVID-19 Pandemic." Innovations in Smart Cities Applications, Volume 4, 2021.

[10] Faang, Ho Yun, NaimahYaakob, Mohamed Elobaid Elshaikh, Azami Khairul Sidek, Ong Bi Lynn, and Mahathir Almashor. "IoT-Based Automated and Contactless Shopping Cart During Pandemic Diseases Outbreak." In Journal of Physics: Conference Series, vol. 1962, no. 1, p. 012051. IOP Publishing, 2021.

APPENDIX

Figure of Components used in this project:

Jumper wires:

1. Female/male

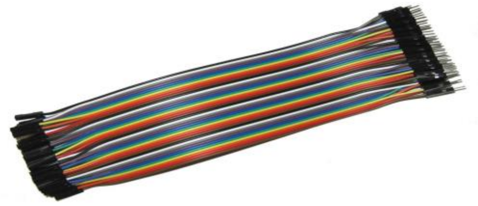


Figure A1

2. Male to male

Figure A2

3. Female to female

Figure A3

4. **USB- B cable**

Figure A4

5. **Mini USB cable for NodeMcu**

Figure A5

6. **Raspberry Pi Pin configuration**

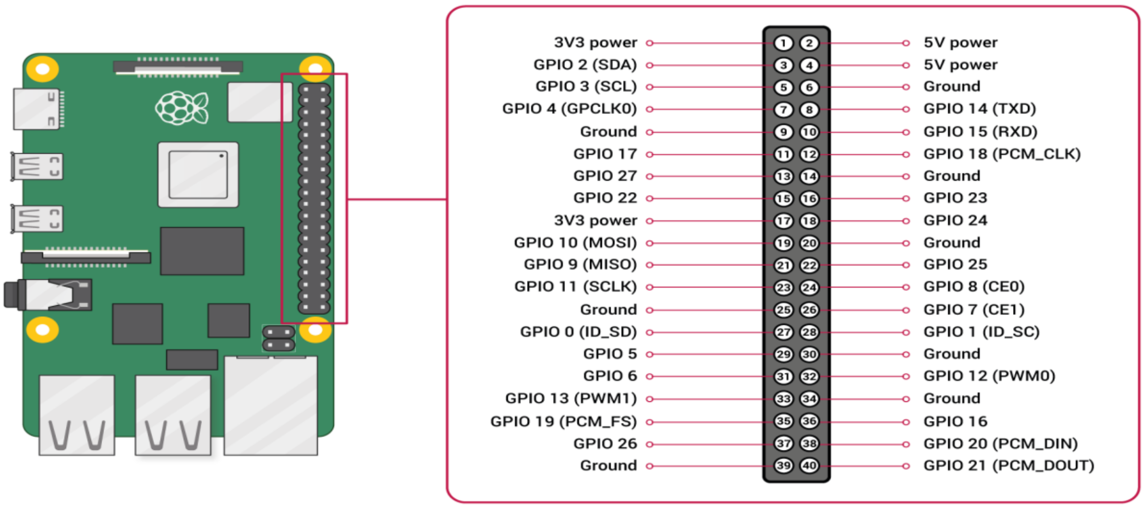


Figure A6

7. Arduino pin configuration

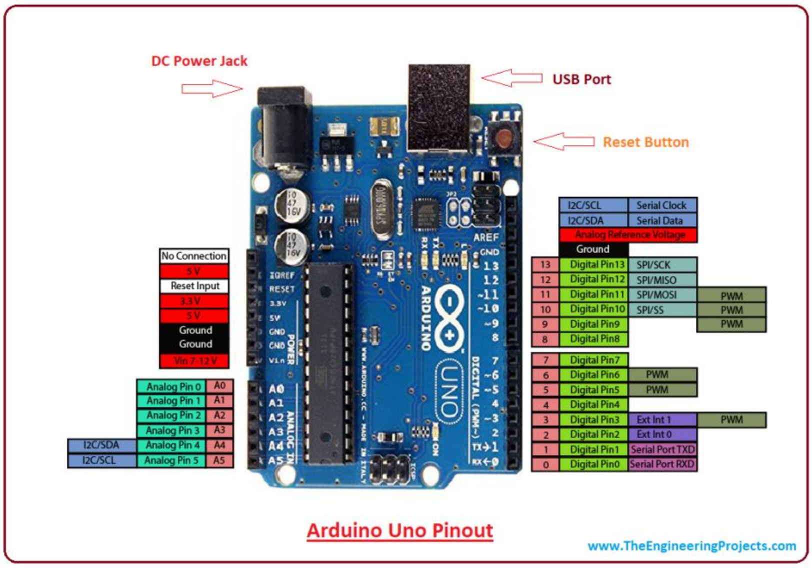


Figure A7

8. Arduino motor shield Pin configuration

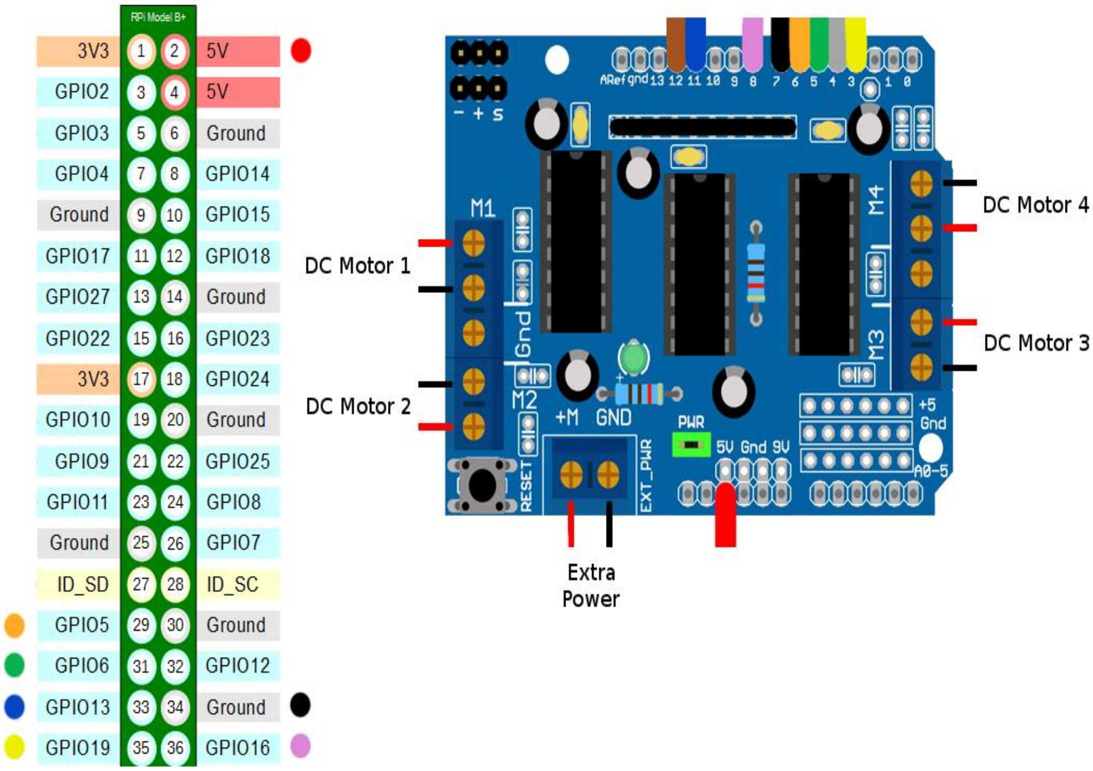


Figure A8

9. Node MCU pin configuration

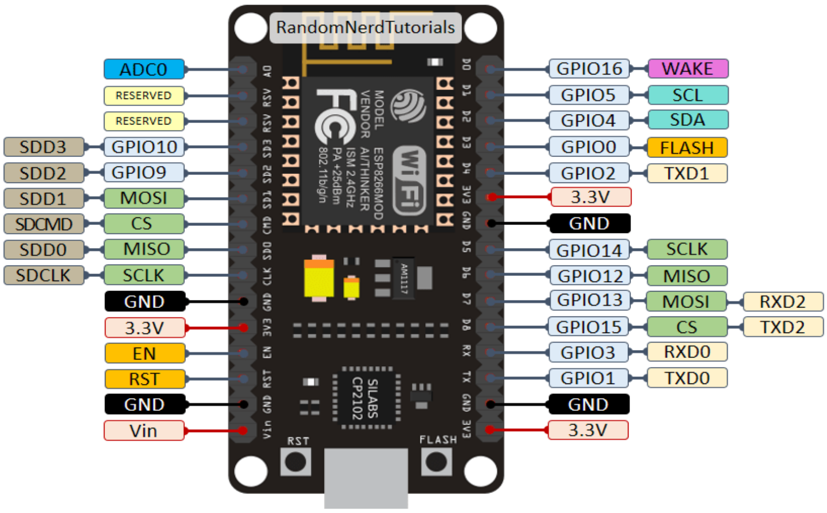


Figure A9