SURVEILLANCE ROBOTIC CAR USING (IOT)

A PROJECT REPORT

Submitted by

GOMATHI. P	**-**	**420320106001**
ASHOK BHARATHI. E	**-**	**420320106302**
PRIYADHARSHINI. S	**-**	**420320106304**

in partial fulfilment for the award of the degree

of

BACHELOR OF ENGINEERING

in

ELECTRONICS AND COMMUNICATION ENGINEERING.

A.R ENGINEERING COLLEGE, VILLUPURAM

ANNA UNIVERSITY, CHENNAI. 600025

MAY 2024

ANNA UNIVERSITY: CHENNAI. 600 025

BONAFIDE CERTIFICATE

Certified that this project report "**SURVEILLANCE ROBOTIC CAR USING (IOT)**" is the bonafide work of **"**GOMATHI.P (420320106001), ASHOK BHARATHI.E (420320106302), PRIYADHARSHINI.S(420320106304)**"** who carried out the project work under my supervision

SIGNATURE
Mr. P. SARAVANAN, M.TECH
HEAD OF THE DEPARTMENT
ASSISTANT PROFESSOR

Department of Electronics and Communication Engineering.
A.R Engineering College,
Vadakuchipalayam,
Kappiyampuliyur post,
Villupuram-605 602

SIGNATURE
Mrs. R. NISHA, M.TECH
SUPERVISOR
ASSISTANT PROFESSOR

Department of Electronics and Communication Engineering.
A.R Engineering College,
Vadakuchipalayam,
Kappiyampuliyur post,
Villupuram-605 602

submitted for the project viva-voice examination held on…….

INTERNAL EXAMINER EXTERNAL EXAMINAR

ACKNOWLEDGEMENT

We would forever remain grateful to our benevolent founder Mr. G.MAHADEVAN Chairman for providing us with facilities for our project to be completed successfully.

We render our special thanks to Mr. M. KUBERAN, M.Tech., Vice Chairman, A.R. Engineering College, for his kind and great patronage

This is the time to express foremost thanks and gratitude to the honorable Mr. M. PRABU, B.Tech., secretary of A.R ENGINEERING COLLAGE for providing us with facilities for our project to be completed successfully.

We express our sincere thanks to Dr. R. PANNEERDHASS, M.E., MBA., Ph.,D., Principal A.R. Engineering College, for all help during this course of study.

This project would not be a success if we do not thank our beloved Head of The Department (HOD), Mr. P. SARAVANAN, M. TECH for providing us invaluable inputs and direction to our approach for this project

We express heartily gratitude to our internal guide Mrs. R. NISHA, M.Tech., Assistant Professor, A.R. Engineering College, for her exceptional training and guidance during the entire tenure of the project

We also thank all the faculty of ECE department for their kind advice, support and encouragement.

It would be gross injustice on our part if we don't mention the constant drive and support that our parents and friends have given us at all times. Last but not least, we would like to thank God for his blessings in making our project successful. We once again thanks to all

ABSTRACT

Robotics is an interesting and fast growing field. The concept of Mobile robot is fast evolving and the number of mobile robots and their complexities are increasing with different applications. Nowadays robots are in wide usage due to their high level of performance and reliability and which is great for human beings. The Surveillance Robot is a vehicle robotics is used for live viewing or monitoring the required area and follows the instructions given by us to move. The Surveillance robotic vehicle is designed in such a way to first track instructions given by us and follows them. The design of the Surveillance robot requires the integration of many sensors according to their task. Here live viewing is the primary requirement. The robot gets the information from the device that we give instruction to it use daily. Some sensing devices like bump sensors, infrared sensors, and ultrasonic sensors are used for obstacle detection. The ultrasonic sensor is most suitable for obstacle detection and is of low cost and has a high ranging capability.

Sensors coupled with motors, microprocessor and few other components combined gives the vehicle an intelligent live viewing with movement. Surveillance robots can be used in almost all mobile robot navigation systems. They can also be used in dangerous environments, where human penetration could be fatal. The human cannot record video safely in critical conditions and environments. These conditions and environments may be buildings where the fire breaks out, Areas with poisonous gases or harmful radiation and the places where there is an exchange of fire such as battlefield.

TABLE OF CONTENT

List of Abbrevations

ESP 32 CAM	-	Espressif Systems P32 Camara
WI-FI	-	Wireless Fidelity
L293N	-	L293n Motor Drive Shield
TF CARD	-	Trans Flash Card
SD CARD	-	Secure Digital Card
DPST Switch	-	Double Pole, Single Throw Switch

List of Figures

Figures

List of Tables

CHAPTER – 1

INTRODUCTION

The device Surveillance Robot is designed in a way to control robot through instruction given by us and follow according via any browser and also in a particular application.

The connection between the mobile and the vehicle is facilitated with internet technology. The aim of the device is to perform required tasks by receiving the commands of the user. The controlling can be done by any smart phone, tab, ios device or laptops etc on any internet enabled devices which we use in our day to day life.

The commands from the mobile are transmitted through the internet from the input given by mobile device from any range the commands forward, backward, right, left and stop are used to control the device. After receiving the commands given by user the microcontroller then operates the motors to move using the motor driver.

The movement of the robot is facilitated by two 100 RPM motors connected to the motor driver.

Also there will be space available to store the things or objects and will be having a door like design controlled using servo motor.

CHAPTER – 2

LITERATURE SURVEY

1. **Wireless Control Surveillance Robot, done by Shaikh Shoeb, Mir Ibrahim, Ansari Asagr Ali published in International Journal of Advance Foundation and Research in Science and Engineering, 2015.**

This project describes a new economical solution of robot control systems. In general; the Robots are controlled through wired network.

The programming of the robot takes time if there is any change in the project the reprogramming has to be done. Thus they are not user friendly and worked along with the user preferences. To make a robot user-friendly and to get the multimedia tone in the control of the robot, they are designed to make user commanded work. The modern technology has to be implemented to do this.

2. **Surveillance robot using Arduino, done buy N. Pugazhenthi, K.Vinu Lakshmi, V. Preneeth published in International Journal of Innovative Technology and Exploring Engineering, 2019.**

Surveillance is the method for observing a location, an area or a person for protection and security purpose. This activity always happens in a military, police, public places and even in houses nowadays for monitoring and to control the illegal activities. Especially, the surveillance activity is used mainly for human because the people were doing all illegal work against the government and at the same time to protect them from those activities. The advent of technology has brought a revolutionary change in the field of robotics, especially in the automation sector. The usage of robotics is increasing day by day, which reduces the human work.

3. **Surveillance Robot using IoT, done by T.Akilan, Princi Kumar, Satyam Chaudhary published in International Journal of Research in Engineering, Science and Management, 2020.**

Surveillance of human activities or any suspicious activities in war field and border lines with the help of a robot based on IoT technology as human access is not possible everywhere and if possible can lead to risk of losing life. Surveillance takes place through wireless camera interfacing with Arduino and various sensors.

WIFI/Bluetooth module is used for communication that is controlled from a distance by Smartphone or a PC. Wireless camera sends the real time video signals. Robot also collects data from various sensors send it to micro-controller. The movement of robot is controlled by the user through a Smartphone or PC. The robot is fully capable to work as required in defence areas as it can be controlled automatically and manually both. According to security perspective this robot is very useful not only in defence but also in domestic areas too.

4. **Border Surveillance Robot, done by Sabiya Sultana, D. Shalini, Prashanth Varma, V.SriCharan published in International Journal of Advance Research, Ideas and Innovations in Technology, 2020.**

Border security provides regional monitoring, immediate warning and border patrolling management. For a long time, this has been a major problem to protect the country's boundaries against terrorists, illegal immigrants, illegal trades etc. Currently, most of the military monitoring services lack the required standard's which is not up to expectation, leading to border soldiers lives being put in constant danger. So as to decrease the soldier lives being lost and to improve the surveillance standard's there is need for a system which can effectively monitor the border with locomotion and surveillance capabilities

5. **Military Surveillance Robot, done by Abhijeet Dhule, Neha Sangle, Supriya Nagarkar, Asmita Namjoshi published in International Research Journal of Engineering and Technology, 2020.**

Surveillance plays an important role in border areas to keep eye on enemies. In such situations it is difficult to allow duty of surveillance to a soldier, which may cause dangerous to the life on one. Rather we can use an robot to keep eye on border areas. So in such cases this kind of robots are very useful they are small in size and provided with many abilities so they can perform the duty of surveillance and spying perfectly. In case if they found by the combatant, they have no identity to whom they belong. Military on border area are facing many problems so this kind of technology help them to aware about the opponent activities, so they can take further decisions.

CHAPTER – 3

AIM OF PROJECT

The main objectives of the project are comprehended as follows:-

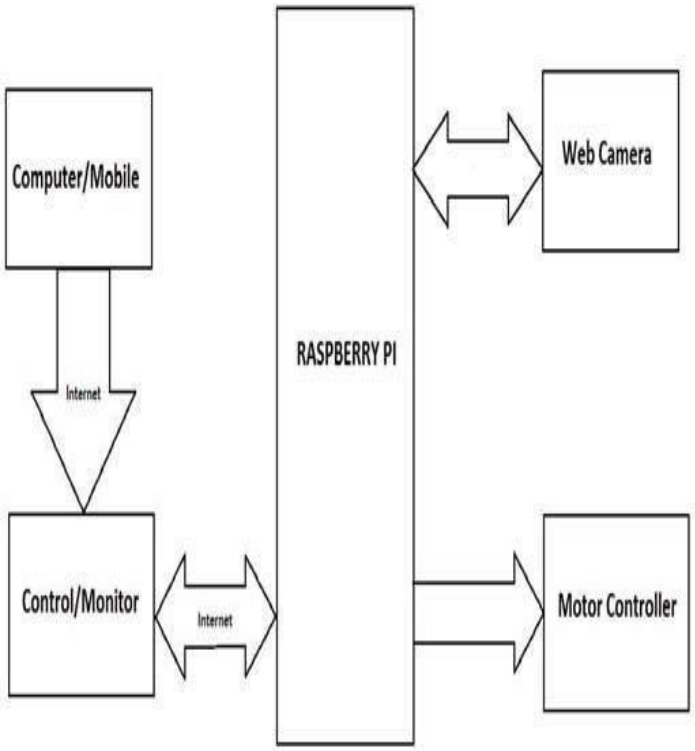
- The features described above suggest a versatile and efficient surveillance robot that could be useful in a variety of applications. Its ability to move around in an unknown environment and show its surroundings using a live camera feed is particularly valuable in surveillance and security contexts.
- The capacity to store and deliver objects to desired locations remotely could also be useful in a variety of settings, such as logistics or healthcare. Additionally, the robot's ability to operate without external control and in unknown environments makes it flexible and adaptable to a wide range of situations.
- The inclusion of a servo-controlled door system and a flash for dark conditions suggests that the designers have considered a range of practical issues that could arise during the robot's operation. The use of an internet device application or browser to view the robot's camera feed also suggests that the robot could be easily controlled and monitored remotely.
- Overall, the described surveillance robot has the potential to be a valuable tool in a range of applications where remote surveillance, delivery, or exploration is required.

CHAPTER – 4

EXISTING METHODOLOGY

4.1. SYSTEM DESCRIPTION:-

Security is very important in day to day life and with growth in technology; a security system can be made with features in accordance with the user's necessities. This system is very user friendly and is very efficient. A wide area can be captured due to the motor control set up in the project. The system has two individual circuits which provide automatic and manual operation of the device, different sensors are used to make the robot automatic and the robot can be manually controlled using raspberry pi set up.



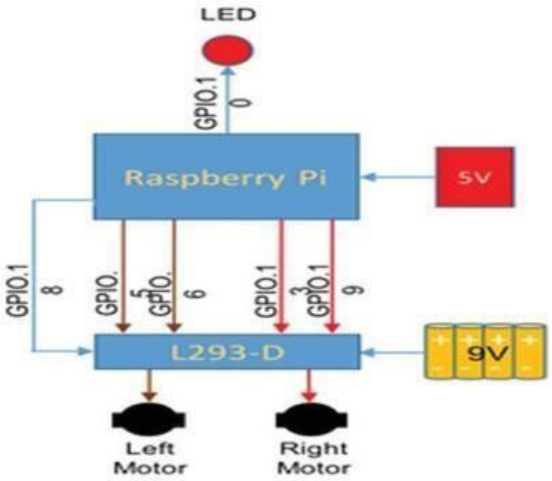


Fig: 4.1.1

Manual Operation Schematic

Fig: 4.1.2

Automatic Operation Schematic

4.2. DESIGN AND IMPLEMENTATION :

This System of the security surveillance robot can be operated in two ways i.e. the robot can be used in automatic mode or it can also be controlled manually through the web. This facility helps the user to get a video feed at all times which can be seen live or through recording. Both the operations are implemented using separate circuits, the annual operation is done using a circuit that uses a raspberry pi as its core and the automatic operation is done by using Arduino as its core. But both modes cannot be used simultaneously they have to be preset do working a particular operating mode. The circuits are constructed considering the drawback; each circuit has its own drawbacks if taken individually but if used together they eliminate most of the drawbacks.

4.3.PROPOSED METHODOLOGY

4.3.1.System Description:-

Live streaming method used in camera for the visibility of the area required using the internet.

App or Browser for viewing live area view of the required area where the internet is available for the device and the controlling system, so that it can be connected and the information will be telecasted from the device to the required monitor or screen.

Esp-32 for controlling the device this is the heart of the device where the whole information will be send and received and this will be sending to which sensor the information is to be send.

Servo motor to close and open the door of the storage area whenever required to the delivery to the desired person.

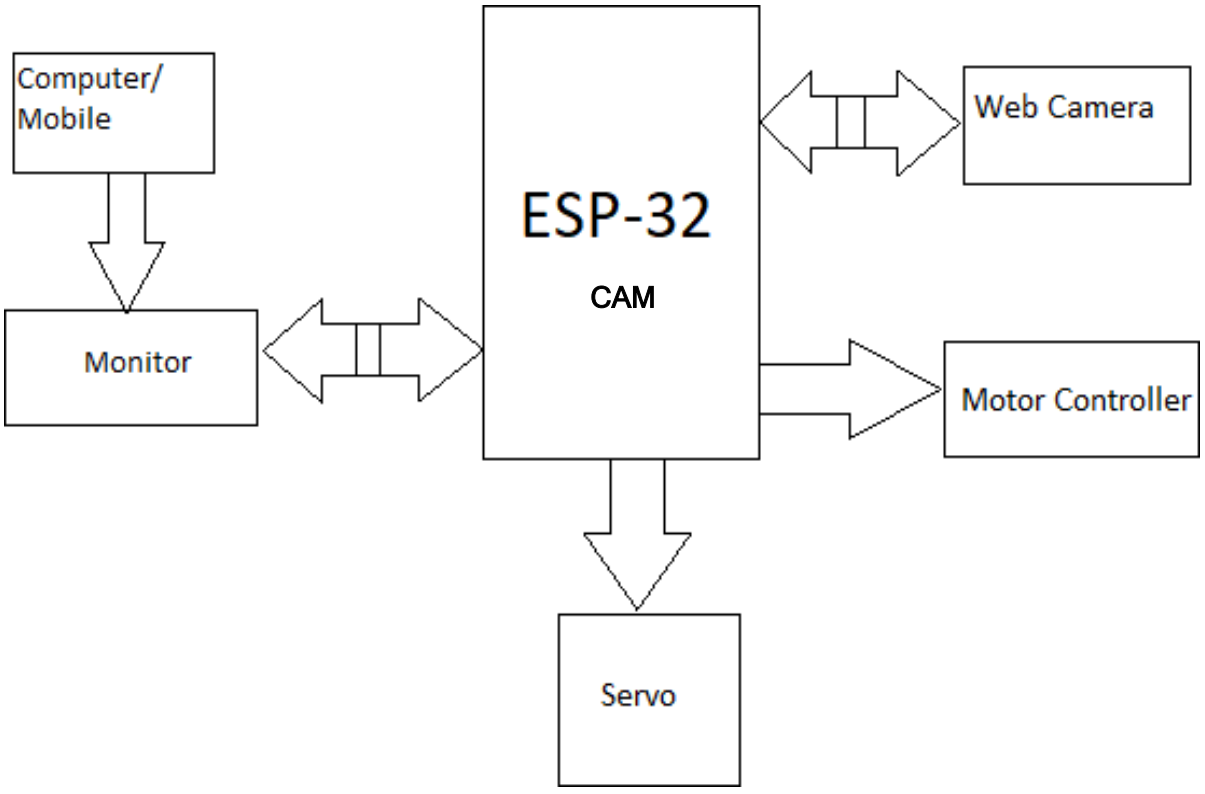


Fig: 4.2.1

Manual Operation Schematic

4.3.2.Design and Implementation:

This System of the security surveillance robot can be operated in two ways i.e. the robot can be used in automatic mode or it can also be controlled manually through the web. This facility helps the user to get a video – feed at all times which can be seen live or through recording. Both the operations are implemented using separate circuits, the manual operation is done using a circuit that uses ESP-32 as its core and the automatic operation is done by using ESP-32 as its core. But both modes cannot be used simultaneously they have to be preset do work in a particular operating mode. The circuits are constructed considering the drawback, each circuit has its own drawbacks if taken individually but if used together they eliminate most of the drawbacks.

CHAPTER - 5

BLOCK DIAGRAM

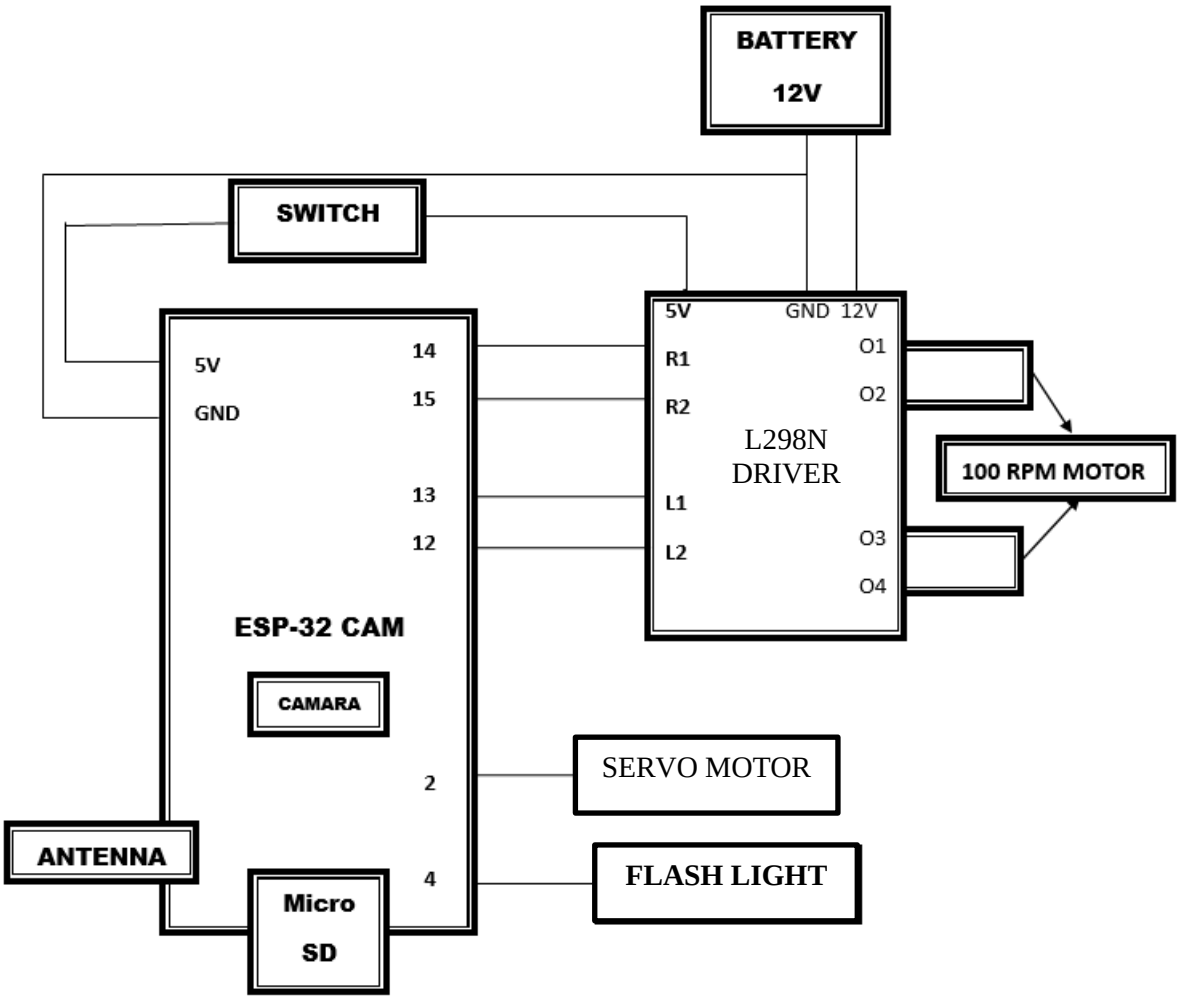


Fig: 5.1

Block Diagram of Surveillance Robot

CHAPTER - 6

COMPONENTS USED

6.1. HARDWARE COMPONENTS

- ESP-32 Camera Module
- L293N Motor Driver IC
- SG-90 Servo Motor
- Micro SD card
- Antenna
- Rechargeable Batteries (12V)
- 100Rpm Motors
- Mountain Wheels
- Switch
- Jumper wire and connecting wire

6.2. ESP-32 CAMERA MODULE

The ESP32-CAM is a small size, low power consumption camera module based on ESP32. It comes with an OV2640 camera and provides on board TF card slot. The ESP32-CAM can be widely used in intelligent IoT applications such as wireless video monitoring, Wi-Fi image upload, QR identification, and so on.

6.2.1.Features

On-board ESP32-S module, supports Wi-Fi + Bluetooth; OV2640 camera with flash

On-board TF card slot, supports up to 4G TF card for data storage Supports Wi-Fi video monitoring and Wi-Fi image upload Supports multi sleep modes, deep sleep current as low as 6mA

Control interface is accessible via pin header, easy to be integrated and embedded into products

6.2.2.Specifications:

WIFI module: ESP-32S; ***Processor***: ESP32-D0WD; ***Built-in Flash***: 32Mbit

RAM: Internal 512KB + External 4M PSRAM; ***Antenna***: On-board PCB antenna

Wi-Fi protocol: IEEE 802.11 b/g/n/e/I; ***Bluetooth***: Bluetooth 4.2 BR/EDR and BLE

WIFI mode: Station / Soft AP / Soft AP + Station Security: WPA/WPA2/WPA2-Enterprise/WPS

Output image format: JPEG (OV2640 support only), BMP, GRAYSCALE

Supported TF card: up to 4G

Peripheral interface: UART/SPI/I2C/PWM

IO port: 9; UART baud rate: default 115200bps

Power supply: 5V

6.2.3Transmitting power:

802.11b:17±2dBm (@11Mbps); 802.11g: 14±2dBm (@54Mbps);

802.11n:13±2dBm (@HT20, MCS7)

6.2.4Receiving sensitivity:

CCK, 1Mbps: -90 dBm; CCK, 11Mbps: -85 dBm; 6Mbps (1/2 BPSK): -88 dBm; 54Mbps (3/4 64-QAM): -70 dBm; HT20, MCS7 (65Mbps, 72.2Mbps): -67 dB

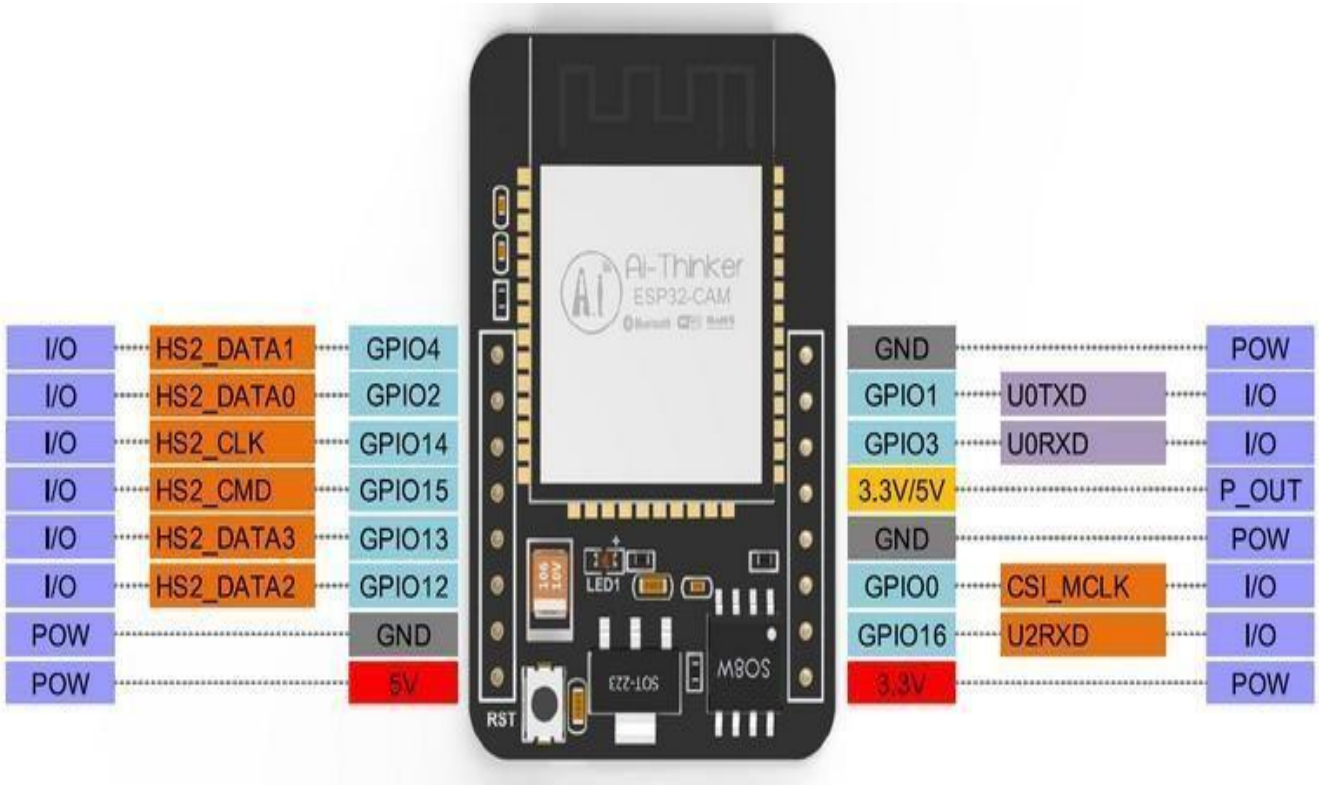


Fig:6.1.1

Pin Description of ESP-32

6.2.5.Applications:

- ✓ Smart home devices image upload
- ✓ Wireless monitoring Intelligent agriculture
- ✓ QR wireless identificationFacial recognition

6.3. L293N MOTOR DRIVE SHIELD

This 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. The driver greatly simplifies and increases the ease with which you may control motors, relays, etc. from microcontrollers.

It can drive motors up to 36V with a total DC current of up to 600mA. You can connect the two channels in parallel to double the maximum current or in series to double the maximum input voltage.

This motor driver uses screw terminals for easy connections, mounting holes for easy mounting, back EMF protection circuit, on board heat sink for better heat dissipation and more efficient performance.

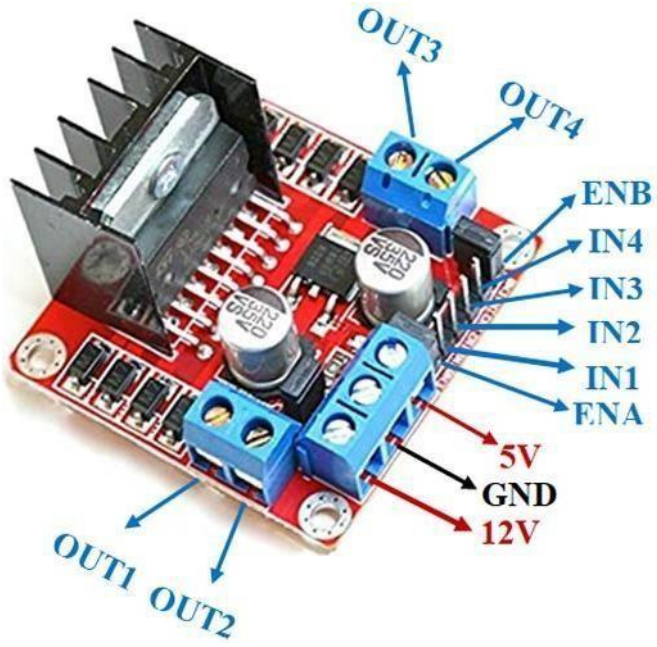


Fig: 6.2.1

L293N Motor Driver Shield

This motor driver is perfect for robotics and mechatronics projects for controlling motors from microcontrollers, switches, relays, etc. Perfect for

You may use this driver with our development boards to control DC and Stepper Motors. To use this driver with our development board, just connect the power pins

On the driver to the power pins on the development board, and connect the drivers input pins to suitable IO pin on the development board. You may use our Female to Female - Jumper Wires to make the connection

Pin Name	Description
IN1 & IN2	Motor A input pins. Used to control the spinning direction of Motor A
IN3 & IN4	Motor B input pins. Used to control the spinning direction of Motor B
ENA	Enables PWM signal for Motor A
ENB	Enables PWM signal for Motor B
OUT1 & OUT2	Output pins of Motor A
OUT3 & OUT4	Output pins of Motor B
12V	12V input from DC power Source
5V	Supplies power for the switching logic circuitry inside L298N IC
GND	Ground pin

Table 6.1 Pin description

6.3.1.Features:

Driver Model: L298N 2A; Driver Chip: Double H Bridge L298N

Motor Supply Voltage (Maximum): 46V; Motor Supply Current (Maximum): 2A

Logic Voltage: 5V; Logical Current: 0-36mA

Driver Voltage: 5-35V; Driver Current: 2A

Maximum Power (W): 25W; Current Sense for each motor; On-board Heat sink for better performanceThe driver two holes of 3mm diameter for easy mounting.

6.3.2.Applications:

Drive DC motors Drive stepping motors In Robotics

6.4.SG-90 SERVO MOTOR

A servo motor is an electrical device which can push or rotate an object with great precision. If we want to rotate and object at some specific angles or distance, then we use servo motor. It is just made up of simple motor which run through servo mechanism. If motor is used is DC powered then it is called DC servo motor, and if it is AC powered motor then it is called AC servo motor.

Fig: 6.3.1

Servo Motor

We can get a very high torque servo motor in a small and light weight packages. Doe to these features they are being used in many applications like toy car, RC helicopters and planes, Robotics, Machine etc. The position of a servo motor is decided by electrical pulse and its circuitry is placed beside the motor. Now day's servo system has huge industrial applications. Servo motor applications are also commonly seen in remote controlled toy cars for controlling direction of motion and it is also very commonly used as the motor which moves the tray of a CD or DVD player. Beside these there are other hundreds of servo motor applications we see in our daily life. The main reason behind using a servo is that it provides angular precision, i.e. it will only rotate as much we want and then stop and wait for next signal to take further action. This is unlike a normal electrical motor which starts rotating as and when power is applied to it and the rotation continues until we switch off the power.

6.4.1.WORKING PRINCIPLE OF SERVO MOTORS:

A servo consists of a Motor (DC or AC), a potentiometer, gear assembly and a controlling circuit. First of all, we use gear assembly to reduce RPM and to increase torque of motor. Say at initial position of servo motor shaft, the position of the potentiometer knob is such that there is no electrical signal generated at the output port of the potentiometer. Now an electrical signal is given to another input terminal of the error detector amplifier. Now difference between these two signals, one comes from potentiometer and another comes from other source, will be processed in feedback mechanism and output will be provided in term of error signal. This error signal acts as the input for motor and motor starts rotating. Now motor shaft is connected with potentiometer and as motor rotates so the potentiometer and it will generate a signal. So as the potentiometer's angular position changes, its output feedback signal changes. After sometime the position of potentiometer reaches at a position that the output of potentiometer is same as external signal provided. At this condition, there will be no output signal from the amplifier to the motor input as there is no difference between external applied signal and the signal generated at potentiometer, and in this situation motor stops rotatin

6.4.2.Applications:

1.Robotics 2.Animatronics

3.Radio Control Cars/Boats/Planes

6.4.3.Advantages:

1. Low cost - (RC Servos) Smaller sized servos can be purchased for just a few dollars.
2. Variety - There is a wide range of sizes and torque ratings.
3. Simple to control - using logic level pulses from a microcontroller or a dedicated servo controller.

6.5. MICRO SD CARD

Memory cards act as electronic storage for your devices by storing digital media such as photos and videos. If you have a camera, camcorder, drone or mobile device, chances are you use a memory card.

The SD and micro SD variants are most commonly used in smartphones and digital cameras. They fit into everything from your DSLR to your Nintendo Switch. However, not all cards are created equal — different devices require a different type of memory card.

The first thing to consider when choosing a memory card is figuring out what type of card your device requires, which can usually be found in the instruction manual or on the manufacturer's website. This will usually indicate what SD standard the device requires. Both SD and micro SD cards share the same standards: SD, SDHC, SDXC and SDUC, and micro SD, microSDHC, microSDXC and microSDUC.

The two standards that are more popular today for both SD and microSD cards are SDHC and SDXC. The main difference between the SD standards is storage capacity. If you shoot 4K video, you'll most likely need an SDXC card since it has a maximum capacity of 2TB, which is plenty for optimal video recording performance. Additionally, SDXC cards use the exFAT file system to support the large files that can be produced when recording video at high bitrates with cameras

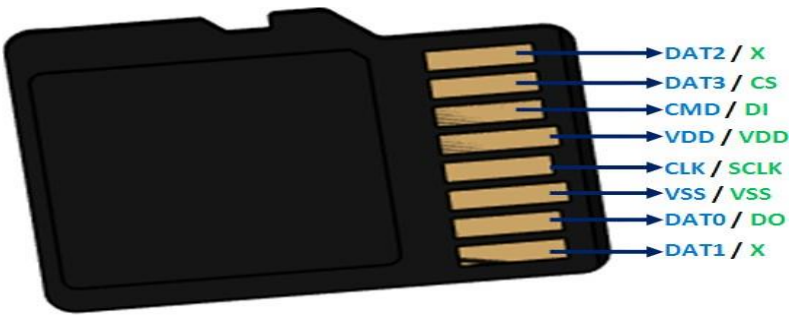


Fig: 6.4.1

Pin Description of Micro SD card

MicroSD cards share the same four SD standards as SD cards. The four SD standards for microSD cards are:

MicroSD - 2GB and under

MicroSDHC -More than 2GB, up to 32GB MicroSDXC -More than 32GB, up to 2TB MicroSDUC -More than 2TB, up to 128TB

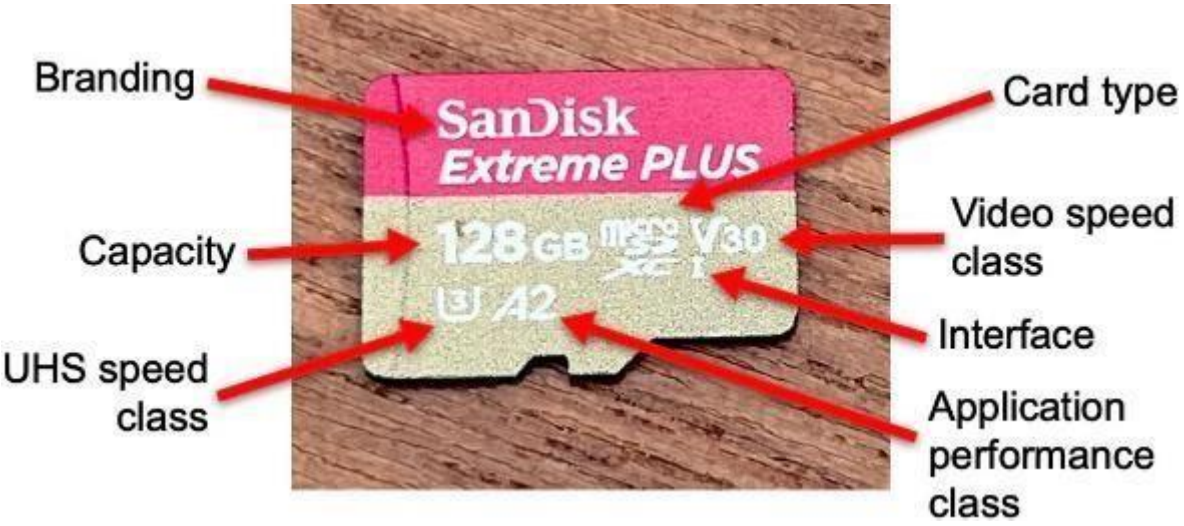


Fig: 6.4.2

Micro SD Card

MicroSD cards are the smaller-sized version of SD cards and the biggest difference between the two is the form factor. They're also more versatile since they're often available with an SD adaptor that allows you to use microSD cards in hardware devices that only support SD cards. MicroSD cards are more commonly used to expand the storage system of smartphones, drones, gaming devices and cameras.

Hardware devices are also backward compatible with microSD cards, just like full-sized SD cards. The same rules apply for microSD cards as they do for SD cards. A microSDXC-compatible device will work with microSDXC, microSDHC and microSD cards. A microSDHC-compatible device will work with microSDHC and microSD cards. A microSD-compatible device will only work with a microSD card. Again, hardware devices that support newer standards are backward compatible with older standard microSD cards.

6.6. ANTENNA

The ESP32-CAM comes with an on-board Wi-Fi antenna, but it also has an IPEX connector if you want to use an external antenna. Using an external antenna can solve problems related with slow video streaming web servers and other connectivity problems. This tutorial shows how to use an external antenna with the ESP32-CAM.

The ESP32-CAM has the option to use either the built-in PCB antenna or an external antenna.

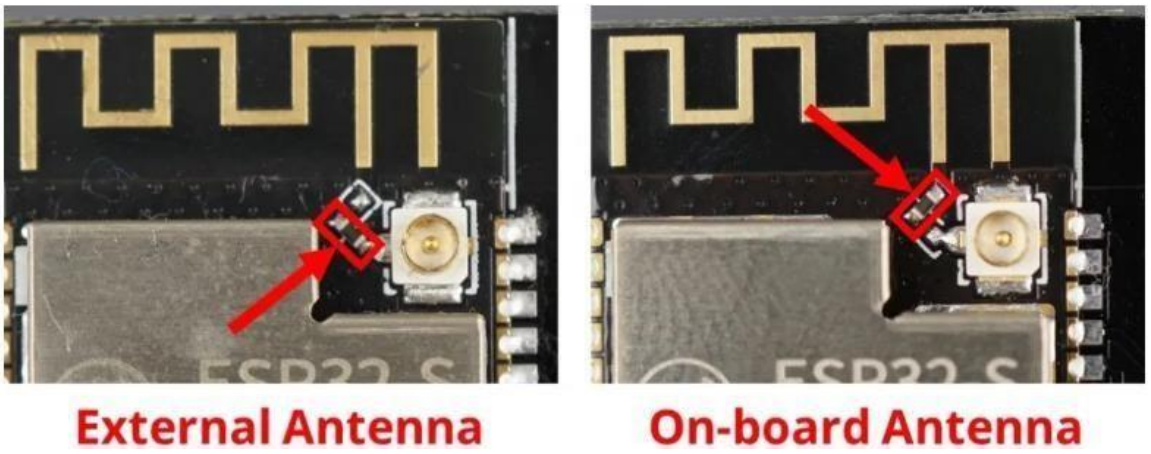


Fig: 6.5.1

Difference between External Antenna and on Board Antenna

An external antenna and an on-board antenna are two different types of antennas used in electronic devices. The main difference between these two types of antennas is their physical location and design.

An external antenna is a separate component that is attached to the device using a cable or connector. It is mounted on the exterior of the device and is often designed to be adjustable or removable. External antennas can be used to improve the range and quality of wireless signals, such as Wi-Fi, Bluetooth, or cellular signals. They are commonly found in devices such as routers, modems, and mobile phones.

On the other hand, an on-board antenna is a built-in antenna that is integrated into the electronic device itself.

It is typically smaller and more compact than an external antenna and does not require a cable or connector to connect to the device. On-board antennas are commonly used in devices such as laptops, tablets, and smartphones, where space is limited and a compact design is required.

The main advantage of an on-board antenna is its compact size and integrated design, which makes it more convenient and aesthetically pleasing. However, its performance may be affected by the proximity of other components and the materials used in the device's construction. In contrast, an external antenna can be positioned for optimal signal reception and is less susceptible to interference from other components in the device.

In summary, the main difference between an external antenna and an on-board antenna is their physical location and design. External antennas are separate components that are mounted on the exterior of the device, while on-board antennas are built-in and integrated into the device itself. Each type of antenna has its own advantages and disadvantages, depending on the specific application and requirements.Next to the IPEX connector there are three little white squares laid out like a "<" with the middle position being common. There is a resistor selecting the desired antenna. Here are the two configurations:

To use the IPEX connector with an external antenna, the resistor must be on the bottom position, like this "\". See illustration below;

To use the PCB antenna (on-board antenna), the resistor must be on the top position, like this "/".

Take a look at your board to see if it is set to use the on-board antenna or the IPEX connector. Using the on-board antenna works well if you are close to your router. We recommend using the IPEX connector with an external antenna for better results

6.7. BATTRIES

Rechargeable batteries, also known as secondary batteries, are batteries that can be recharged and reused multiple times. They are a more environmentally friendly and cost-effective alternative to disposable batteries, which are designed to be used once and then discarded.

Rechargeable batteries can be made of various chemistries, including nickel-cadmium (Ni-Cd), nickel-metal hydride (Ni-MH), and lithium-ion (Li-ion). Each type of battery has its own advantages and disadvantages, depending on the specific application and requirements.

One of the main advantages of rechargeable batteries is their ability to be recharged and reused multiple times, which reduces the amount of waste generated by disposable batteries. Rechargeable batteries also tend to have a higher energy density than disposable batteries, which means they can store more energy in a smaller size.

Model: 12V 6Ah Battery Model: 18650 Cell type: Lithium-ion Rated voltage: 11.1V- 12.6V

Nominal capacity: 6000mAh Charge method: CC-CV Continuous discharge current: 5A

Instantaneous discharge current: 10A Maximum working current: 3A Maximum instantaneous current: 5A Charge temperature: 0-45 °C Discharge temperature: -10-60? Storage temperature: -20-45 °C Cycle life: 80%

DOD after 500 cycles

Weight: 300g Size: 110 * 70 * 19MM

Input: DC 5.5 * 2.1 plug wire With protection circuit: overcharge, over discharge, overcurrent, short circuit protection functions

Scope of application: various power models, lighting tools, monitoring systems, electric tools, electric toys, etc. All 12 V electrical equipment, washing equipment, coal, hernia lamps, not 12 V batteries, standby power supply, LED lamp, instrumentation, amplifiers, monitors, electrical types, various 12 V lamps. The advantages of lithium batteries: small size, light weight, large capacity, environmental protection, no memory effect.

Fig: 6.6.1

12V Rechargeable Batteries

Model Name: 12V 6Ah 18650 rechargeable battery 6000mAh Li-ion battery pack LED light backup power audio battery + 12.6V 1A charger

Type: Travel Adaptor Colour: Black

Power Requirement: 12V Style Code: HE0517 Colour Code: Black

6.8.100RPM MOTOR

DC Gear motor is also called DC Geared Motor, Geared Dc Motor and gearhead motor or gearbox motor. It consists of a electric DC motor and a gearbox or gearhead; these gearheads are used to reduce the DC motor speed, while increase the DC motor torque. Therefore user can get lower speed and higher torque from gear motor.

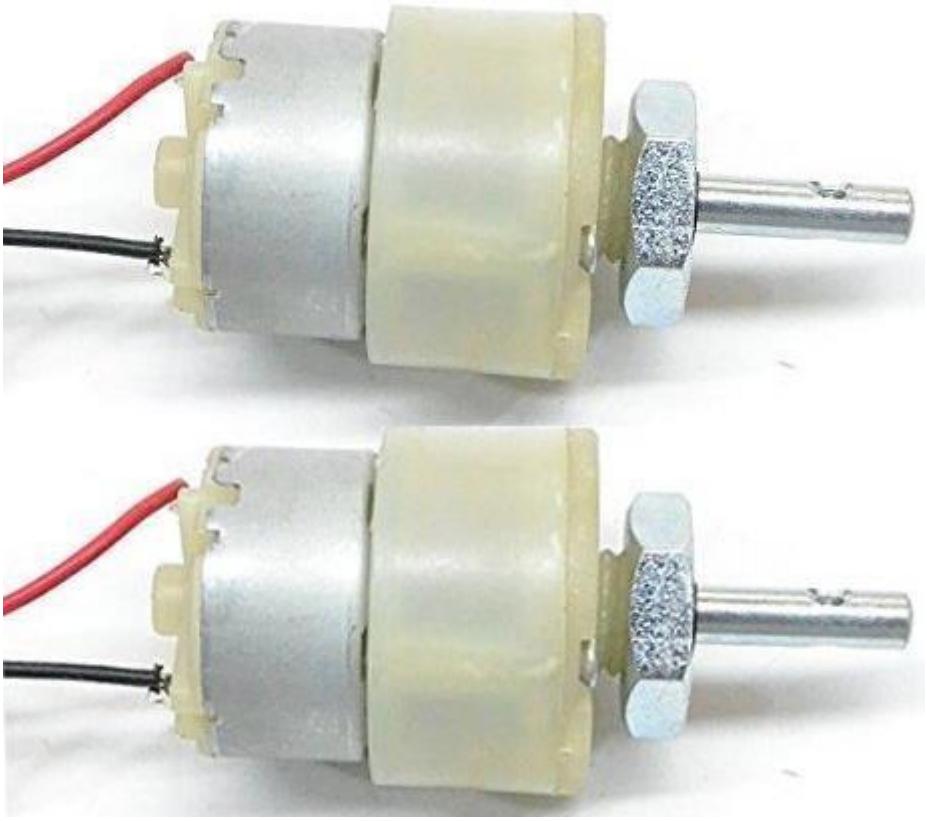


Fig: 6.7.1

100Rpm DC gear motors

There are different types of 100RPM motors available, including DC (direct current) motors and AC (alternating current) motors. DC motors are typically powered by batteries or power supplies and are known for their simplicity, high torque, and precise speed control. AC motors, on the other hand, are typically powered by the mains electricity supply and are known for their reliability, efficiency, and low maintenance requirements.

The performance of a 100RPM motor depends on a variety of factors, including its power rating, voltage, and torque. Higher power ratings generally result in higher speed and torque, while higher voltages may be required to achieve the desired speed. The torque of the motor also affects its performance, with higher torque motors being able to handle heavier loads.

When selecting a 100RPM motor for a specific application, it is important to consider the requirements of the application, such as the load, power source, and speed control requirements. It is also important to consider the motor's durability, noise level, and overall efficiency to ensure that it is suitable for the intended use.

6.8.1Advantages:

- ✓ DC Voltage: from 1VDC to 30VDC Low Speed: from 0.1rpm to 2000rpm Torque: 0.1kg.cm to 400kg.cm
- ✓ Mini size: diameter from Φ6mm to Φ50mm Low noise
- ✓ Long lifetime

6.9. OTHER SMALLER COMPONENTSWHEELS

The design and construction of wheels vary depending on the application, but they all share common features, including the hub, the spokes, and the rim. The hub is the central part of the wheel that attaches it to the axle. The spokes connect the hub to the rim and provide support and stability to the wheel. The rim is the outer edge of the wheel that provides the surface on which the tire is mounted.

Fig: 6.8.1

Wheels

Wheels are made from a variety of materials, including steel, aluminum, carbon fiber, and plastic. The choice of material depends on the specific application and the desired properties, such as strength, durability, and weight.

In summary, wheels are a fundamental component of transportation and movement and have played a critical role in human history. They come in various designs and materials and are used in a wide range of applications, from cars to conveyor belts.

6.10.JUMPER WIRES

Jumper wires, also known as jumper cables or jumpers, are electrical wires that are used to make temporary connections between two points in an electronic circuit. They are often used for prototyping and testing circuits, as well as for repairing and modifying electronic devices.

Fig: 6.8.2

Jumper Wires

Jumper wires are typically made of insulated wire with a small metal pin or clip at each end. The pins or clips are used to connect the jumper wire to the circuit components or to a breadboard, which is a tool used for prototyping circuits.

Jumper wires are available in various lengths, colors, and gauges (thicknesses), depending on the specific application. They may also come in pre-made sets, which include a variety of lengths and colors for easy organization and identification.

Jumper wires are an essential tool for anyone working with electronics, as they allow for quick and easy testing and modification of circuits. They are commonly used in educational settings, hobbyist projects, and professional electronics work.

6.11.DPST Switch

DPST stands for Double Pole Single Throw, and it refers to a type of electrical switch that has two independent circuits controlled by a single actuator.

The DPST switch has two terminals, which are connected to the two independent circuits. The actuator of the switch is used to change the position of the switch, which connects one circuit and disconnects the other. The switch has two possible positions: on and off.

Fig: 6.8.3
DPST Switch

The term "double pole" refers to the fact that the switch has two separate circuits, and "single throw" means that the switch can only be in one of two positions, either on or off.

DPST switches are commonly used in a variety of electrical applications, such as controlling power to two different devices, reversing the direction of a motor, or selecting between two power sources.

They are also used as safety switches, where the switch controls the power to both poles of a circuit, ensuring that the circuit is completely disconnected when the switch is in the off position.

DPST switches come in a variety of shapes, sizes, and configurations, and can be found in a range of electrical equipment and devices, from industrial machinery to household appliances.

CHAPTER - 7

CONNECTIONS

7.1.Pin Connections:

From ESP-32 to L293N Motor Driver

- Pin 14 – R1
- Pin 15 – R2
- Pin 13 – L1
- Pin 12 – L2
- 5V -- 5V
- Grd -- Grd

From L293N Motor Driver to Motors

- O1 – Motor 1(side 1)
- O2 – Motor 1(side 2)
- O3 – Motor 2(side 1)
- O4 – Motor 2(side 2)

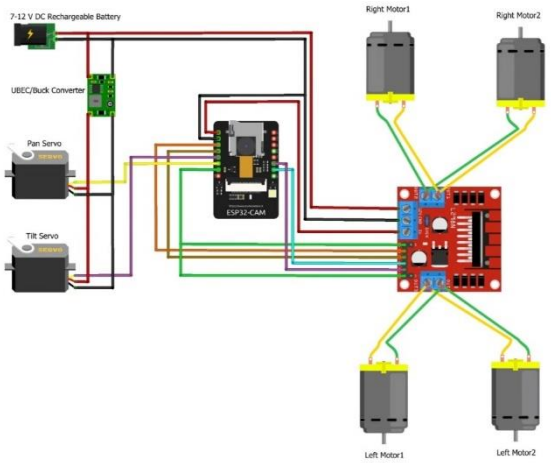


Fig: 7.1

Connections of Surveillance Robot

From ESP-32 to Other Components:

- Pin 2 – Servo Motor
- Pin 4 – Flash
- Memory Card in slot holder
- Camera in-built
- Antenna in extension

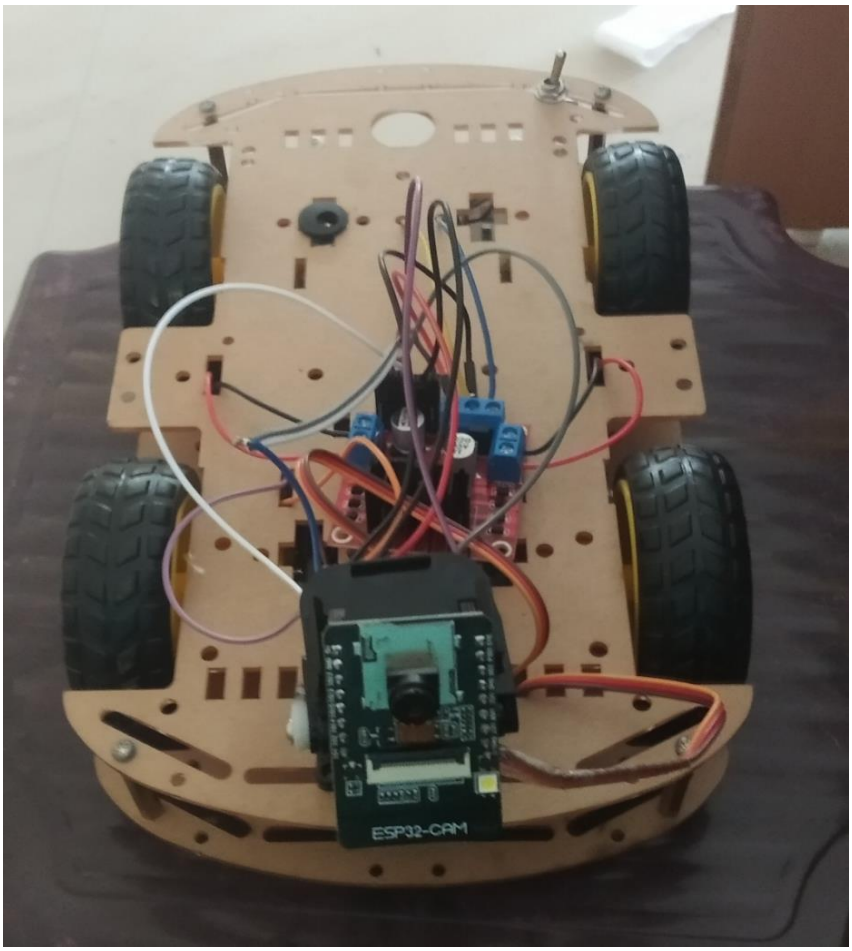
CHAPTER – 8

WORKING METHOD

1. The robot is switched on by using 12V DC batteries as the power supply from external source, then the ESP-32 and L293N Motor driver shield gets power to get started. With this the robot gets moving.
2. The Camera starts viewing and will be used for live viewing on the desired device on which you are given with the IP address or on an application that are built for it.
3. The motors are used for the movement of the robot by taking the instruction given by us.
4. The ESP-32 will be taking all the information from us and giving commands to all the robot and this is also called heart of the robot and this will be connecting to the internet where communication between the devices will be taken here from monitor to the device.
5. The controlling can be done by the IP entered device browser or the application built for it.
6. The controls will be forward, backward, right, left, stop, flash, servo and speed of motor.
7. The device can be controlled from anywhere in the world with any internet enabled device.
8. The live viewing can be seen in the monitor of the controlling device.

Fig: 8.1

Surveillance Robot



CHAPTER - 9

APPLICATIONS

- The modification of this logic code is used in vacuum cleaners.
- This robot can be used for avoiding concealed paths and monitor the area, such as an industrial robot in a factory is expected to avoid workers so that it won't hurt those.
- It will be very useful in parking system.
- It can also be used in assembling automobiles and in chemical industries.
- They have great importance in scientific exploration and emergency rescue, there may be places that are dangerous for humans or even impossible for humans to reach directly, then we should use robots to help us gather information to about their surrounding challenging environments.
- This is also secured as to connect it the IP address is required so it can't be accessed by everyone and can easy to connect.

CHAPTER – 10

FUTURE ENHANCEMENT

- This technique can also be used as a vision belt for blind people by adding a kinetic sensor, which is a type of microwave sensor whose sensing range is very high and the output of this sensor vary in according to the object position changes. This technique enables blind people to navigate obstacles easily by placing three vibratos in left, right and the center of a belt named as Vision Belt.
- On top of Surveillance robot temperature/ pressure sensors can be added to monitor the atmospheric conditions around. This is useful in places where the environment is not suitable for humans. Same technology can be used in various applications by modifying the program for example Line / Path finder robot, automatic vacuum cleaner etc.
- They can be used as services robots, for the low cost Obstacle Avoidance Robot, purpose of household work and so many other indoor applications.
- Use as a firefighting robot: By adding temperature sensor, water tank and making some changes in programming we can use this robot as firefighting robot.
- We can extend this project with wireless technology by IR (or) RF (or) ZIGBEE.
- We can use the DTMF receiver by using the mobile phone.
- This robot can be used for pick and place the required object by giving directions to the robot but ultrasonic sensor should be replaced depending upon the application.

CHAPTER – 11

RESULT

- Today we are in the world of robotics. Knowingly or unknowingly, we have been using different types of robots in our daily life.
- The project is “ESP-32 based Surveillance Robot for Millitary Applications” is practically proved by using the Camera for live telecasting Motor Shield Driver for the driving the dc motors, dc motor is used for the movement of the robot with the help of the Esp-32 Microcontroller.
- A lot of factors determined the accuracy of the robot we designed. These factors were the environmental phenomenon in which the robot was tested, the number of obstacles present making the test space crowded or relatively less crowded the type and shape of the obstacle (the robot is designed for a uniform shaped obstacle).These factors majorly affected the sensors.
- The accuracy of the robot is dependent on the sensors used. Thus, the nature of the sensor and its accuracy defined the accuracy of my robot.

CHAPTER – 12

CONCLUSION

By way of this project, it can be concluded that the practicality and scope of this robotic unit is quite extensive, and can play a huge role in the surveillance and safety industries in best interest of its stake holders and users. This unit can also be used in the warehouses of large companies, where the importance of monitoring in case an error of misplacement has taken, is to be relayed back to the connected controlling person's device. The main components of this entire unit, namely the ESP32 Cam Module and the NodeMCU ESP8266 Module, hold the majority of the importance and relevance in terms of the working of this robot. As has been explained earlier, the ESP32 Cam Module is meant for collecting the visual data and through the Wi-Fi connection relays it back to the concerned device, while the ESP8266 Module is the basic controlling module using which the robot can be moved in either of the directions, using another or the same Wi-Fi connected device.

PROGRAMME

```
#include "esp_camera.h"
#include <Arduino.h>
#include <WiFi.h>
#include <AsyncTCP.h>
#include <ESPAsyncWebServer.h>
#include <iostream>
#include <sstream>
#include <ESP32Servo.h>

#define PAN_PIN 14
#define TILT_PIN 15

Servo panServo;
Servo tiltServo;

struct MOTOR_PINS
{
  int pinEn;
  int pinIN1;
  int pinIN2;
};

std::vector<MOTOR_PINS> motorPins =
{
  {2, 12, 13}, //RIGHT_MOTOR Pins (EnA, IN1, IN2)
  {2, 1, 3},  //LEFT_MOTOR  Pins (EnB, IN3, IN4)
};
#define LIGHT_PIN 4

#define UP 1
#define DOWN 2
#define LEFT 3
#define RIGHT 4
#define STOP 0

#define RIGHT_MOTOR 0
#define LEFT_MOTOR 1

#define FORWARD 1
#define BACKWARD -1

const int PWMFreq = 1000; /* 1 KHz */
```

```
const int PWMResolution = 8;
const int PWMSpeedChannel = 2;
const int PWMLightChannel = 3;

//Camera related constants
#define PWDN_GPIO_NUM     32
#define RESET_GPIO_NUM    -1
#define XCLK_GPIO_NUM      0
#define SIOD_GPIO_NUM     26
#define SIOC_GPIO_NUM     27
#define Y9_GPIO_NUM       35
#define Y8_GPIO_NUM       34
#define Y7_GPIO_NUM       39
#define Y6_GPIO_NUM       36
#define Y5_GPIO_NUM       21
#define Y4_GPIO_NUM       19
#define Y3_GPIO_NUM       18
#define Y2_GPIO_NUM        5
#define VSYNC_GPIO_NUM    25
#define HREF_GPIO_NUM     23
#define PCLK_GPIO_NUM     22

const char* ssid     = "For the love of PHYSICS";
const char* password = "12345678";

AsyncWebServer server(80);
AsyncWebSocket wsCamera("/Camera");
AsyncWebSocket wsCarInput("/CarInput");
uint32_t cameraClientId = 0;

const char* htmlHomePage PROGMEM = R"HTMLHOMEPAGE(
<!DOCTYPE html>
<html>
  <head>
  <meta name="viewport" content="width=device-width, initial-scale=1, maximum-
scale=1, user-scalable=no">
    <style>
    .arrows {
      font-size:30px;
      color:red;
    }
    td.button {
      background-color:black;
      border-radius:25%;
```

```
  box-shadow: 5px 5px #888888;
}
td.button:active {
  transform: translate(5px,5px);
  box-shadow: none;
}

.noselect {
  -webkit-touch-callout: none; /* iOS Safari */
    -webkit-user-select: none; /* Safari */
     -khtml-user-select: none; /* Konqueror HTML */
       -moz-user-select: none; /* Firefox */
        -ms-user-select: none; /* Internet Explorer/Edge */
            user-select: none; /* Non-prefixed version, currently
                                  supported by Chrome and Opera */
}

.slidecontainer {
  width: 100%;
}

.slider {
  -webkit-appearance: none;
  width: 100%;
  height: 15px;
  border-radius: 5px;
  background: #d3d3d3;
  outline: none;
  opacity: 0.7;
  -webkit-transition: .2s;
  transition: opacity .2s;
}

.slider:hover {
  opacity: 1;
}

.slider::-webkit-slider-thumb {
  -webkit-appearance: none;
  appearance: none;
  width: 25px;
  height: 25px;
  border-radius: 50%;
  background: red;
```

```
    cursor: pointer;
   }

   .slider::-moz-range-thumb {
    width: 25px;
    height: 25px;
    border-radius: 50%;
    background: red;
    cursor: pointer;
   }

   </style>

  </head>
  <body class="noselect" align="center" style="background-color:white">
   <table id="mainTable" style="width:400px;margin:auto;table-layout:fixed"
CELLSPACING=10>
    <tr>
     <img id="cameraImage" src="" style="width:400px;height:250px"></td>
    </tr>
    <tr>
     <td></td>
     <td class="button" ontouchstart='sendButtonInput("MoveCar","1")'
ontouchend='sendButtonInput("MoveCar","0")'><span class="arrows"
>&#8679;</span></td>
     <td></td>
    </tr>
    <tr>
     <td class="button" ontouchstart='sendButtonInput("MoveCar","3")'
ontouchend='sendButtonInput("MoveCar","0")'><span class="arrows"
>&#8678;</span></td>
     <td class="button"></td>
     <td class="button" ontouchstart='sendButtonInput("MoveCar","4")'
ontouchend='sendButtonInput("MoveCar","0")'><span class="arrows"
>&#8680;</span></td>
    </tr>
    <tr>
     <td></td>
     <td class="button" ontouchstart='sendButtonInput("MoveCar","2")'
ontouchend='sendButtonInput("MoveCar","0")'><span class="arrows"
>&#8681;</span></td>
     <td></td>
    </tr>
    <tr/><tr/>
```

```
    <tr>
      <td style="text-align:left"><b>Speed:</b></td>
      <td colspan=2>
       <div class="slidecontainer">
         <input type="range" min="0" max="255" value="150" class="slider"
id="Speed" oninput='sendButtonInput("Speed",value)'>
       </div>
      </td>
    </tr>
    <tr>
      <td style="text-align:left"><b>Light:</b></td>
      <td colspan=2>
       <div class="slidecontainer">
         <input type="range" min="0" max="255" value="0" class="slider" id="Light"
oninput='sendButtonInput("Light",value)'>
       </div>
      </td>
    </tr>
    <tr>
      <td style="text-align:left"><b>Pan:</b></td>
      <td colspan=2>
       <div class="slidecontainer">
         <input type="range" min="0" max="180" value="90" class="slider" id="Pan"
oninput='sendButtonInput("Pan",value)'>
       </div>
      </td>
    </tr>
    <tr>
      <td style="text-align:left"><b>Tilt:</b></td>
      <td colspan=2>
       <div class="slidecontainer">
         <input type="range" min="0" max="180" value="90" class="slider" id="Tilt"
oninput='sendButtonInput("Tilt",value)'>
       </div>
      </td>
    </tr>
  </table>

  <script>
    var webSocketCameraUrl = "ws:\/\/" + window.location.hostname + "/Camera";
    var webSocketCarInputUrl = "ws:\/\/" + window.location.hostname + "/CarInput";
    var websocketCamera;
    var websocketCarInput;
```

```
    function initCameraWebSocket()
    {
      websocketCamera = new WebSocket(webSocketCameraUrl);
      websocketCamera.binaryType = 'blob';
      websocketCamera.onopen    = function(event){};
      websocketCamera.onclose   = function(event){setTimeout(initCameraWebSocket,
2000);};
      websocketCamera.onmessage = function(event)
      {
        var imageId = document.getElementById("cameraImage");
        imageId.src = URL.createObjectURL(event.data);
      };
    }

    function initCarInputWebSocket()
    {
      websocketCarInput = new WebSocket(webSocketCarInputUrl);
      websocketCarInput.onopen    = function(event)
      {
        sendButtonInput("Speed", document.getElementById("Speed").value);
        sendButtonInput("Light", document.getElementById("Light").value);
        sendButtonInput("Pan", document.getElementById("Pan").value);
        sendButtonInput("Tilt", document.getElementById("Tilt").value);
      };
      websocketCarInput.onclose   =
function(event){setTimeout(initCarInputWebSocket, 2000);};
      websocketCarInput.onmessage = function(event){};
    }

    function initWebSocket()
    {
      initCameraWebSocket ();
      initCarInputWebSocket();
    }

    function sendButtonInput(key, value)
    {
      var data = key + "," + value;
      websocketCarInput.send(data);
    }

    window.onload = initWebSocket;
    document.getElementById("mainTable").addEventListener("touchend",
function(event){
```

```
      event.preventDefault()
    });
   </script>
  </body>
</html>
)HTMLHOMEPAGE";


void rotateMotor(int motorNumber, int motorDirection)
{
  if (motorDirection == FORWARD)
  {
    digitalWrite(motorPins[motorNumber].pinIN1, HIGH);
    digitalWrite(motorPins[motorNumber].pinIN2, LOW);
  }
  else if (motorDirection == BACKWARD)
  {
    digitalWrite(motorPins[motorNumber].pinIN1, LOW);
    digitalWrite(motorPins[motorNumber].pinIN2, HIGH);
  }
  else
  {
    digitalWrite(motorPins[motorNumber].pinIN1, LOW);
    digitalWrite(motorPins[motorNumber].pinIN2, LOW);
  }
}

void moveCar(int inputValue)
{
  Serial.printf("Got value as %d\n", inputValue);
  switch(inputValue)
  {

    case UP:
      rotateMotor(RIGHT_MOTOR, FORWARD);
      rotateMotor(LEFT_MOTOR, FORWARD);
      break;

    case DOWN:
      rotateMotor(RIGHT_MOTOR, BACKWARD);
      rotateMotor(LEFT_MOTOR, BACKWARD);
      break;

    case LEFT:
```

```
      rotateMotor(RIGHT_MOTOR, FORWARD);
      rotateMotor(LEFT_MOTOR, BACKWARD);
      break;

    case RIGHT:
      rotateMotor(RIGHT_MOTOR, BACKWARD);
      rotateMotor(LEFT_MOTOR, FORWARD);
      break;

    case STOP:
      rotateMotor(RIGHT_MOTOR, STOP);
      rotateMotor(LEFT_MOTOR, STOP);
      break;

    default:
      rotateMotor(RIGHT_MOTOR, STOP);
      rotateMotor(LEFT_MOTOR, STOP);
      break;
  }
}

void handleRoot(AsyncWebServerRequest *request)
{
  request->send_P(200, "text/html", htmlHomePage);
}

void handleNotFound(AsyncWebServerRequest *request)
{
    request->send(404, "text/plain", "File Not Found");
}

void onCarInputWebSocketEvent(AsyncWebSocket *server,
                      AsyncWebSocketClient *client,
                      AwsEventType type,
                      void *arg,
                      uint8_t *data,
                      size_t len)
{
  switch (type)
  {
    case WS_EVT_CONNECT:
      Serial.printf("WebSocket client #%u connected from %s\n", client->id(), client-
>remoteIP().toString().c_str());
      break;
```

```
    case WS_EVT_DISCONNECT:
      Serial.printf("WebSocket client #%u disconnected\n", client->id());
      moveCar(0);
      ledcWrite(PWMLightChannel, 0);
      panServo.write(90);
      tiltServo.write(90);
      break;
    case WS_EVT_DATA:
      AwsFrameInfo *info;
      info = (AwsFrameInfo*)arg;
      if (info->final && info->index == 0 && info->len == len && info->opcode ==
WS_TEXT)
      {
        std::string myData = "";
        myData.assign((char *)data, len);
        std::istringstream ss(myData);
        std::string key, value;
        std::getline(ss, key, ',');
        std::getline(ss, value, ',');
        Serial.printf("Key [%s] Value[%s]\n", key.c_str(), value.c_str());
        int valueInt = atoi(value.c_str());
        if (key == "MoveCar")
        {
          moveCar(valueInt);
        }
        else if (key == "Speed")
        {
          ledcWrite(PWMSpeedChannel, valueInt);
        }
        else if (key == "Light")
        {
          ledcWrite(PWMLightChannel, valueInt);
        }
        else if (key == "Pan")
        {
          panServo.write(valueInt);
        }
        else if (key == "Tilt")
        {
          tiltServo.write(valueInt);
        }
      }
      break;
    case WS_EVT_PONG:
```

```
    case WS_EVT_ERROR:
      break;
    default:
      break;
  }
}

void onCameraWebSocketEvent(AsyncWebSocket *server,
                            AsyncWebSocketClient *client,
                            AwsEventType type,
                            void *arg,
                            uint8_t *data,
                            size_t len)
{
  switch (type)
  {
    case WS_EVT_CONNECT:
      Serial.printf("WebSocket client #%u connected from %s\n", client->id(), client-
>remoteIP().toString().c_str());
      cameraClientId = client->id();
      break;
    case WS_EVT_DISCONNECT:
      Serial.printf("WebSocket client #%u disconnected\n", client->id());
      cameraClientId = 0;
      break;
    case WS_EVT_DATA:
      break;
    case WS_EVT_PONG:
    case WS_EVT_ERROR:
      break;
    default:
      break;
  }
}

void setupCamera()
{
  camera_config_t config;
  config.ledc_channel = LEDC_CHANNEL_4;
  config.ledc_timer = LEDC_TIMER_2;
  config.pin_d0 = Y2_GPIO_NUM;
  config.pin_d1 = Y3_GPIO_NUM;
  config.pin_d2 = Y4_GPIO_NUM;
  config.pin_d3 = Y5_GPIO_NUM;
```

```
  config.pin_d4 = Y6_GPIO_NUM;
  config.pin_d5 = Y7_GPIO_NUM;
  config.pin_d6 = Y8_GPIO_NUM;
  config.pin_d7 = Y9_GPIO_NUM;
  config.pin_xclk = XCLK_GPIO_NUM;
  config.pin_pclk = PCLK_GPIO_NUM;
  config.pin_vsync = VSYNC_GPIO_NUM;
  config.pin_href = HREF_GPIO_NUM;
  config.pin_sscb_sda = SIOD_GPIO_NUM;
  config.pin_sscb_scl = SIOC_GPIO_NUM;
  config.pin_pwdn = PWDN_GPIO_NUM;
  config.pin_reset = RESET_GPIO_NUM;
  config.xclk_freq_hz = 20000000;
  config.pixel_format = PIXFORMAT_JPEG;

  config.frame_size = FRAMESIZE_VGA;
  config.jpeg_quality = 10;
  config.fb_count = 1;

  // camera init
  esp_err_t err = esp_camera_init(&config);
  if (err != ESP_OK)
  {
    Serial.printf("Camera init failed with error 0x%x", err);
    return;
  }

  if (psramFound())
  {
    heap_caps_malloc_extmem_enable(20000);
    Serial.printf("PSRAM initialized. malloc to take memory from psram above this
size");
  }
}

void sendCameraPicture()
{
  if (cameraClientId == 0)
  {
    return;
  }
  unsigned long  startTime1 = millis();
  //capture a frame
  camera_fb_t * fb = esp_camera_fb_get();
```

```
  if (!fb)
  {
     Serial.println("Frame buffer could not be acquired");
     return;
  }

  unsigned long  startTime2 = millis();
  wsCamera.binary(cameraClientId, fb->buf, fb->len);
  esp_camera_fb_return(fb);

  //Wait for message to be delivered
  while (true)
  {
    AsyncWebSocketClient * clientPointer = wsCamera.client(cameraClientId);
    if (!clientPointer || !(clientPointer->queueIsFull()))
    {
      break;
    }
    delay(1);
  }

  unsigned long  startTime3 = millis();
  Serial.printf("Time taken Total: %d|%d|%d\n",startTime3 - startTime1, startTime2 -
startTime1, startTime3-startTime2 );
}

void setUpPinModes()
{
  panServo.attach(PAN_PIN);
  tiltServo.attach(TILT_PIN);

  //Set up PWM
  ledcSetup(PWMSpeedChannel, PWMFreq, PWMResolution);
  ledcSetup(PWMLightChannel, PWMFreq, PWMResolution);

  for (int i = 0; i < motorPins.size(); i++)
  {
    pinMode(motorPins[i].pinEn, OUTPUT);
    pinMode(motorPins[i].pinIN1, OUTPUT);
    pinMode(motorPins[i].pinIN2, OUTPUT);
    /* Attach the PWM Channel to the motor enb Pin */
    ledcAttachPin(motorPins[i].pinEn, PWMSpeedChannel);
  }
  moveCar(STOP);
```

```
  pinMode(LIGHT_PIN, OUTPUT);
  ledcAttachPin(LIGHT_PIN, PWMLightChannel);
}


void setup(void)
{
  setUpPinModes();
  //Serial.begin(115200);

  WiFi.softAP(ssid, password);
  IPAddress IP = WiFi.softAPIP();
  Serial.print("AP IP address: ");
  Serial.println(IP);

  server.on("/", HTTP_GET, handleRoot);
  server.onNotFound(handleNotFound);

  wsCamera.onEvent(onCameraWebSocketEvent);
  server.addHandler(&wsCamera);

  wsCarInput.onEvent(onCarInputWebSocketEvent);
  server.addHandler(&wsCarInput);

  server.begin();
  Serial.println("HTTP server started");

  setupCamera();
}


void loop()
{
  wsCamera.cleanupClients();
  wsCarInput.cleanupClients();
  sendCameraPicture();
  Serial.printf("SPIRam Total heap %d, SPIRam Free Heap %d\n", ESP.getPsramSize(),
ESP.getFreePsram());
}
```

CHAPTER – 13

REFERENCES

- F. Tang, Y. Ying, J. Wang and Q. Peng, "A novel texture synthesis based algorithm for object removal in photographs",
  . LNCS, vol. 3321, ASIAN2004 ASIAN2004 pp. 248-258.
- Haritaoglu, D. Harwood and L.S. Davis, "W4: “Real-Time Surveillance of People and their Activities", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2020.vol. 22, pp. 809-830.
- J. Schiff, M. Meingast, D.K. Mulligan, S. Sastry and K Goldberg, "Respectful cameras: Detecting visual markers in real-time to address privacy concerns", Protecting Privacy in Video Surveillance, 2019, pp. 65-89.
- M. Md Athiq UR Raza Ahamed and Wajid Ahamed, "A Domestic Robot for Security Systems by Video Surveillance using Zigbee Technology",

  international Journal of Scientific Engineering and Technology, May 2021, vol. 2, no. 5, pp. 448- 453.
- P Kumar, A. Singhal, S. Mehta and A. Mittal, "Real-time moving object detection algorithm on high resolution using GPUs", Journal of Real-Time Image Processing, International journal of computer application, Mar. 2019, vol. 11, no. 1, pp. 93-109, 2020. vol. 117.
- P. Vamsi krishna, S.R. Hussain, N. Ramu, P.M. Rao, G. Rohan and B.D.S Teja, "Advanced Raspberry Pi Surveillance (ARS) system",

  Proceedings of the 2015 Global Conference on Communication Technologies (GCCT), 23–24 April 2019, pp. 860-862.

- Tasleem Mandrupkar and Manisha KumariRupali Mane, "Smart Video Security Surveillance with Mobile Remote Control", International journal of Advanced Research in Computer Science and Software Engineering, 2021, vol. 3, no. 3.
- V Meenakshi, Ch. Lakshmi Saketh and K. Kalyan Kumar, "Secured Spy IP Control Robot Using Raspberry Pi", International Journal of Emerging Technology and Advanced Engineering, 2018, vol. 5, no. 2, pp. 513-518.
- Zinkevich, A.V. (2021). ESP8266 Microcontroller Application in Wireless Synchronization Tasks, 2021 Intrenational Conference on Industrial Engineering, Applications and Manufacturing(ICIEAM)
- Ajay Sudhir Bale, N.Vinay, Subhashish Tiwari “Exploring Speciality Optical fiber for AStrophotonics Applications”
- Naveen Ghorpade, Muhammad Furqaan Hashim, Jatin Vasihnav “A Comprehensive Study on Metaverse and it’s impact on humans”