

ROBOTICS BOOT CAMP

ELECTRONIC COMMUNICATION AND ENGINEERING(ECE)

THE ROCK STARS

## Team Leader: N. Naga Madhuri (23KE1A0478)

## Team Members:

## V. Sravani (23KE1A04C5)

## V. Jahnavi (23KE1A04C7)

## N. Yusha Sri (23KE1A0479)

## E. Hima Bindu (24KE5A0406)

## CH. Alekhya Gayatri(24KE5A0404)

## From:- ECE



# OBSTACLE ROBOT

## INTRODUCTION OF OBSTACLE ROBOT:-

* An obstacle avoidance robot is a type of robot that can detect and navigate around obstacles in its environment. These robots use sensors and algorithms to:
* Detect obstacles: Identify objects or barriers in their path
* Map the environment: Create a mental map of their surroundings
* Plan a path: Determine the best route to avoid obstacles
* Navigate: Move around obstacles and reach their destination

## REQUIREMENTS OF OBSTACLE ROBOT:-

* CHASSIS
* WHEELS
* MOTOR DRIVER(L298)
* MOTOR (ACUTATORS)
* JUMPER WIRES
* BATTERY
* ESP32 CHIP
* ULTRA SONIC SENSOR
* GENERAL PCB (PRINTED CIRCUIT BOARD)
* SERVO MOTOR

### METALLIC CHASSIS:-

* A robot chassis, also known as it’s frame, is a structural component of the robot that’s provides a foundation for the robot and other components allows it’s to move the robot.



### MOTORS:-

* A motor is an electrical device that converts electrical energy into mechanical energy, producing motion or torque.



### MOTOR DRIVER(L298N):-

A motor driver(L298N) is an electronic device,which can amplify‘s the current’s upto 2A. Motor drivers act as an interface between a micro controller and the motor.

Key functions of a motor driver:

* Direction control: Switches motor rotation between clockwise and counterclockwise.
* Speed control: Regulates motor speed, often using Pulse Width Modulation (PWM).
* Current limiting: Prevents excessive current draw, protecting the motor and driver.
* Overheat protection: Shuts down the driver if it overheats.



### WHEELS:-

Wheels play a crucial role in robots, enabling mobility, navigate around the ground, and interaction with the environment.

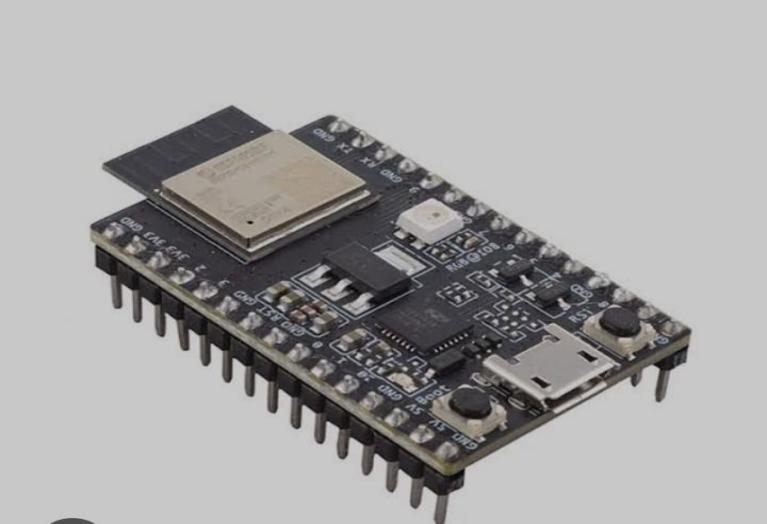


### ESP32 CHIP:-

The ESP32 chip can be used as the control unit in robotics project to provide wireless control and communication capabilities to robot. ESP32 is a micro controller with integrated Wi-Fi and Bluetooth capabilities. It's widely used in robotics for its versatility, ease of use, and connectivity features.

Here are some examples of how the ESP32 can be used for wireless robot:

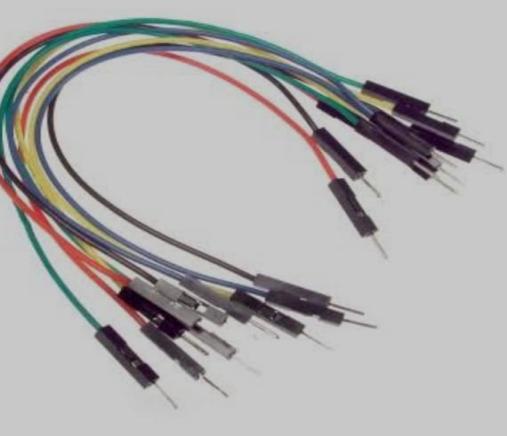
* Bluetooth game pad controller
* Wi-Fi remote controller
* PS3 controller



* JUMPER WIRES:-

Jumper wires are used for these connections:

* Connect sensors and actuators to micro controllers.
* Link modules and components in robotic systems.
* Prototype and test robotic circuits.
* Debug and troubleshoot robotic systems.



### BATTERY(POWER SOURCE):-

Power sources in robots provide the necessary energy

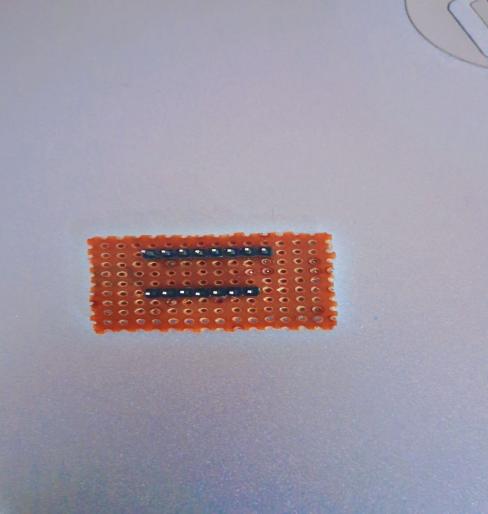
for operation. Common power sources includes like Batteries,Solar panels,. Pneumatic or hydraulic systems.

In the IOT robot we are using power source like battery of 12v-1A-speed-300RPS



### GENARAL PCB(PRINTED CIRCUIT BOARD):-

The common terminal of +5V and GND terminal are connected in general PCB



## ULTRA SONIC SENSOR:-

An ultrasonic sensor is a device that uses high-frequency sound waves to measure the distance, proximity, or presence of an object. It works on the principle of echolocation.

* Transmitter: The sensor emits high-frequency ultrasonic sound waves (typically above 20 kHz).
* Propagation: The sound waves travel through the air until they hit an object.
* Reflection: The sound waves bounce back from the object and return to the sensor.
* Receiver: The sensor receives the reflected sound waves and measures the time-of-flight (ToF).
* Calculation: The sensor calculates the distance based on the ToF and the speed of sound.
* In ultra sonic sensor it measures the distance in the formula of
* DISTANCE=0.034cm/micro sec\*T/2



### SERVO MOTOR:-

A servo motor is a type of electric motor that uses a feedback mechanism to precisely control its rotation or position. It's commonly used in applications where accurate positioning and control are required.

Key characteristics:

* Precise control: Servo motors can rotate to a specific angle or position.
* Feedback mechanism: A sensor monitors the motor's position and sends feedback to the controller.
* High torque: Servo motors produce high torque relative to their size.
* Low speed: Servo motors typically operate at low speeds.



### PROCEDURE:-

* Design the chassis of your robot whether it will plastic or metallic chassis.
* First check positive and negative terminals of the motors, by connecting the motors to the battery positive to positive and negative to negative to the battery then motor moves clock wise direction and also fix the motors left and right to the chassis.
* Motor is a polarized component then we have connect the left motors and right motors in parallel connection between positive and negative terminals.
* In Parallel connection the voltage passes equally to the motors of left and right then the all motors moves equally.
* Where the equation is

V DIRECTLY PROPORTIONAL TO S

* Connect the four wheels to the four motors
* After that we have connect the connections between motors,L298N(motor driver),ESP32 chip then the connections are
* Then connect the motor left(ML) positive to L298N(out1 pin) and negative terminals to L298N(out2 pin).
* After that connect motor right(MR)positive to L298N (out3 pin) and negative terminals to L298N (out4 pin).
* In L298N (Vcc pin) is for positive terminal of a battery.
* In L298N two wires across the (GND pin) one wire is for negative terminal of battery and another wire of female end of jumper wire is connected to the one side of the general PCB.
* L298N (+5V pin) is connected to another side of the general PCB it is considered as +5V
* ESP32(Vin pin) is connected to the common +5V of the general PCB
* ESP32(GND pin) is connected to the common GND of the general PCB
* The ultra sonic sensor is connected to the top of the servo motor
* The ultra sonic sensor(VCC pin) is connected to the common +5V pin of the general PCB
* The ultra sonic sensor (trig pin) is connected to the ESP32(D13 pin)
* The ultra sonic sensor (echo pin) is connected to the ESP32(D12 pin)
* The ultra sonic sensor (GND pin ) is connected to the common GND of the general PCB
* The servo motor(RED wire ) is connected to the common +5V of the general PCB
* The servo motor (BROWN wire) is connected to the common GND of the general PCB
* The servo motor (ORANGE wire) is connected to the ESP32(D15 pin)
* L298N (IN1 pin) is connected to the ESP32 (D14 pin).
* L298N (IN2 pin ) is connected to the ESP32(D27 pin).
* L298N (IN3 pin) is connected to the ESP32 (D26 pin).
* L298N (IN4 pin) is connected to the ESP32 (D25 pin).
* L298N (ena pin) is connected to the ESP32 (D33 pin).
* L298N (enb pin)is connected to the ESP32 (D32 pin).
* We have to write a program for line follower robot in arduino software in the laptop. Before writing the program we have to install ESP32 servo library -vestion-3.0.5

### Program:-

#include<ESP32Servo.h>

Servo myservo;

int in1=14;

int in2=27;

int in3=26;

int in4=25;

int ena=33;

int enb=32;

int trig=13;

int echo=12;

int servo=15;

int normalSpeed=180;

int highSpeed=255;

int leftDistance,rightDistance;

int threshold=100;

void setup() {

// put your setup code here, to run once:

pinMode(in1,OUTPUT);

pinMode(in2,OUTPUT);

pinMode(in3,OUTPUT);

pinMode(in4,OUTPUT);

pinMode(ena,OUTPUT);

pinMode(enb,OUTPUT);

pinMode(trig,OUTPUT);

pinMode(echo,INPUT);

myservo.attach(servo);

myservo.write(90);

Serial.begin(9600);

}

void loop() {

// put your main code here, to run repeatedly:

int distanceFront=getDistance(90);

if(distanceFront<threshold){

robotStop();

delay(1000);

biscuit:

checkSurroundings();//40cm,120cm

int choice=(leftDistance>rightDistance)?0:180;

if((choice==0 && leftDistance>threshold)||(choice==180 && rightDistance>threshold)){

if(choice==0)

robotLeft();

else if(choice==180)

robotRight();

} else{

robotBack();

delay(1000);

goto biscuit;

}

}else{

robotFront();

}

}

void robotControl(int i1,int i2,int i3,int i4,int ea,int eb){

digitalWrite(in1,i1);

digitalWrite(in2,i2);

digitalWrite(in3,i3);

digitalWrite(in4,i4);

analogWrite(ena,ea);

analogWrite(enb,eb);

}

void robotBack(){

robotControl(1,0,1,0,normalSpeed,normalSpeed);

}

void robotFront(){

robotControl(0,1,0,1,normalSpeed,normalSpeed);

}

void robotRight(){

robotControl(0,0,1,0,0,highSpeed);

}

void robotLeft(){

robotControl(1,0,0,0,highSpeed,0);

}

void robotStop(){

robotControl(0,0,0,0,0,0);

}

int getDistance(int angle){

myservo.write(angle);

delay(300);

int validDistance=0;

for(int i=0;i<3;i++){

digitalWrite(trig,0);

delayMicroseconds(2);

digitalWrite(trig,0);

delayMicroseconds(10);

digitalWrite(trig,0);

float duration=pulseIn(echo,1);

duration=duration/2;

validDistance=(0.034\*duration);

if(validDistance>0)

break;

}

Serial.print("Distance:");

Serial.print(validDistance);

Serial.println("cm");

return validDistance;

}

void checkSurroundings(){

leftDistance=getDistance(0);

rightDistance=getDistance(180);

}

* After writing the program we have to compile the program .
* Then upload the program in ESP32 chip after the uploading then the IOT robot is connected to battery.

### RESULT:-

