END SEMESTER ASSESSMENT (ESA)

B.TECH. (CSE)

VII SEMESTER

UE19CS400SC – Design of IOT Solutions

PROJECT REPORT

ON

Sensor Driven Utility Vehicle

SUBMITTED BY

NAME	SRN
1) Ria Singh	PES2UG19CS326
2) Harsh Pooniwala	PES2UG19CS143
3) Shatakshi Mohan	PES2UG19CS379

AUGUST – DECEMBER 2022

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

ELECTRONIC CITY CAMPUS,

BENGALURU – 560100, KARNATAKA, INDIA

TABLE OF CONTENTS

Sl.No	TOPIC	PAGE NO.
1.	ABSTRACT	3
2.	PRINCIPLE	3
3.	TECH STACK	4
4.	COMPONENTS	5
5.	CIRCUIT DIAGRAM	6
6.	ARDUINO CODE	6-7
7.	IMAGES	8
8.	REFERENCES	8

ABSTRACT:

A significant technology shift is happening in our world, and it is centered around the Internet of Things (IoT). The IoT is all about connecting the unconnected. More and more objects are getting connected to the internet day by day. In this project, we have tried to use IoT to design a utility vehicle that will be of immense use in the near future in this fast-changing world. Using commonly available items and sensors, we have designed a low-cost sensor-driven utility robot that can assist humans in numerous areas. The potential for such a low-cost utility vehicle connected to the internet is limitless.

PRINCIPLE:

The basis for object detection and monitoring is the data that both sensors have accepted. When an object is present in front of the ultrasonic sensor within a given range—in this example, between 10 and 30 centimeters—it can be detected. All four engines are at rest if there isn't anything in this area, like our hand. When an item enters this area, the infrared sensor data is received, and depending on the information collected, orders are provided to the motors, causing the robot to move in the intended direction. The setting for the ultrasonic sensor's minimum response distance, which in our instance is greater than 10 cm, should be used to change this distance so that it is just a little bit greater.

HARDWARE TECH STACK:

Ultrasonic Motion Sensor

An ultrasonic sensor is a device that uses ultrasonic sound waves to detect a target object's distance and then turns the sound that is reflected back into an electrical signal. The speed of audible sound is greater than the speed of ultrasonic waves (i.e. the sound that humans can hear). The transmitter (which generates sound using piezoelectric crystals) and the receiver are the two major parts of an ultrasonic sensor (which encounters the sound after it has traveled to and from the target).

Infrared Sensor

An electrical device that monitors and detects infrared radiation in its environment is called an infrared (IR) sensor. Since IR's wavelength is longer than that of visible light, it is not visible to the human eye (though it is still on the same electromagnetic spectrum). Infrared radiation is produced by everything that emits heat (i.e., anything that is warmer than five degrees Kelvin).

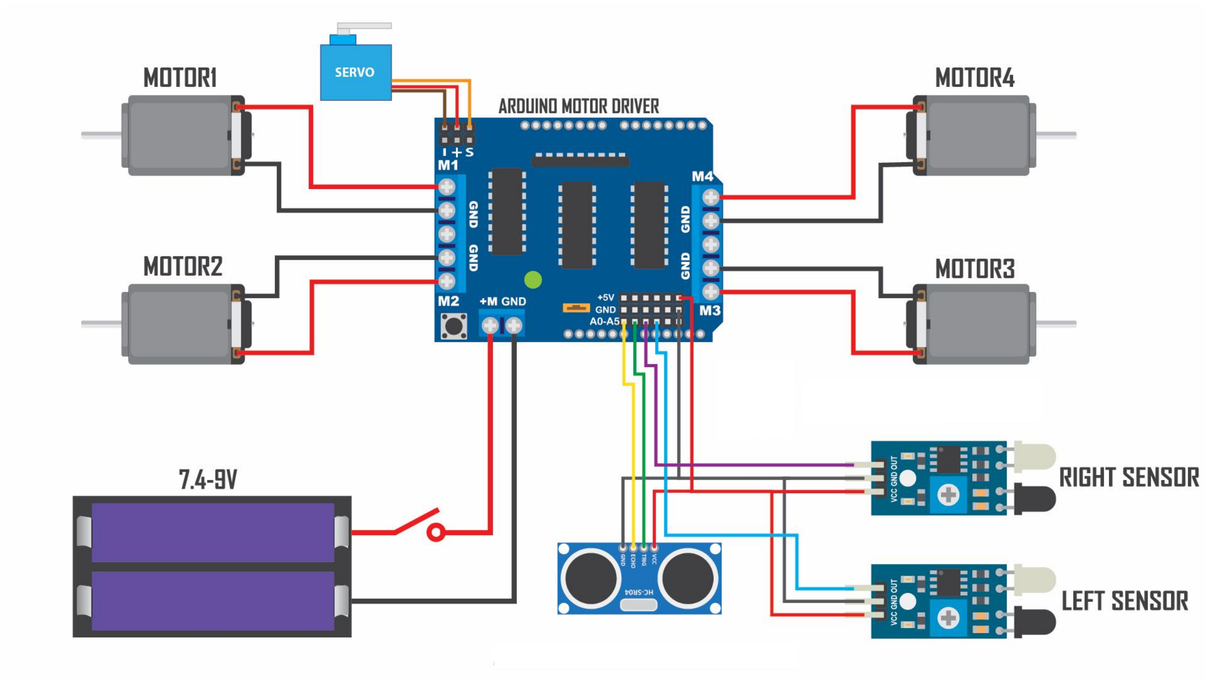
L293D Motor Driver

The L293D is a 16-pin Motor Driver IC which can control a set of two DC motors simultaneously in any direction. The L293D is designed to provide bidirectional drive currents of up to 600 mA (per channel) at voltages from 4.5 V to 36 V. We use it to control small dc motors which are attached to the wheels.

COMPONENT LIST:

Component Name	Quantity
Arduino Uno	1
Motor Driver Shield	1
DC Motor Wheels (4x)	4
TT Gear Motor	1
Servo Motor	1
Ultrasonic Sensor	1
Infrared Sensor	2
18650 Li-on Battery	2
18650 Battery Holder	1
Male and Female Jumper wire	multiple
DC Power Switch	1

CIRCUIT DIAGRAM:



ARDUINO CODE:

```
//Arduino Human Following Robot
Sketchbook
//include the library code:
#include<NewPing.h>
#include<Servo.h>
#include<AFMotor.h>

#define RIGHT A2              // Right IR sensor connected to analog pin A2 of Arduino Uno:
#define LEFT A3               // Left IR sensor connected to analog pin A3 of Arduino Uno:
#define TRIGGER_PIN A1        // Trigger pin connected to analog pin A1 of Arduino Uno:
#define ECHO_PIN A0           // Echo pin connected to analog pin A0 of Arduino Uno:
#define MAX_DISTANCE 200      // Maximum ping distance:

unsigned int distance = 0;    //Variable to store ultrasonic sensor distance:
unsigned int Right_Value = 0; //Variable to store Right IR sensor value:
unsigned int Left_Value = 0;  //Variable to store Left IR sensor value:


NewPing sonar(TRIGGER_PIN, ECHO_PIN, MAX_DISTANCE);  //NewPing setup of pins and maximum distance:

//create motor objects
AF_DCMotor Motor1(1,MOTOR12_1KHZ);
AF_DCMotor Motor2(2,MOTOR12_1KHZ);
AF_DCMotor Motor3(3,MOTOR34_1KHZ);
AF_DCMotor Motor4(4,MOTOR34_1KHZ);

 Servo myservo; //create servo object to control the servo:
 int pos=0;     //variable to store the servo position:

void setup() { // the setup function runs only once when power on the board or reset the board:

   Serial.begin(9600); //initailize serial communication at 9600 bits per second:
   myservo.attach(10); // servo attached to pin 10 of Arduino UNO
{
for(pos = 90; pos <= 180; pos += 1){    // goes from 90 degrees to 180 degrees:
  myservo.write(pos);                   //tell servo to move according to the value of 'pos' variable:
```

```
    delay(15);                          //wait 15ms for the servo to reach the position:
  }
  for(pos = 180; pos >= 0; pos-= 1) {   // goes from 180 degrees to 0 degrees:
    myservo.write(pos);                 //tell servo to move according to the value of 'pos' variable:
    delay(15);                          //wait 15ms for the servo to reach the position:
  }
  for(pos = 0; pos<=90; pos += 1) {     //goes from 180 degrees to 0 degrees:
    myservo.write(pos);                 //tell servo to move according to the value of 'pos' variable:
    delay(15);                          //wait 15ms for the servo to reach the position:
  }
}
   pinMode(RIGHT, INPUT); //set analog pin RIGHT as an input:
   pinMode(LEFT, INPUT);  //set analog pin RIGHT as an input:
}

// the lope function runs forever
void loop() {

delay(50);                                        //wait 50ms between pings:
distance = sonar.ping_cm();                       //send ping, get distance in cm and store it in 'distance' variable:
Serial.print("distance");
Serial.println(distance);                         // print the distance in serial monitor:


    Right_Value = digitalRead(RIGHT);             // read the value from Right IR sensor:
    Left_Value = digitalRead(LEFT);               // read the value from Left IR sensor:

Serial.print("RIGHT");
Serial.println(Right_Value);                      // print the right IR sensor value in serial monitor:
Serial.print("LEFT");
Serial.println(Left_Value);                       //print the left IR sensor value in serial monitor:

if((distance > 1) && (distance < 15)){            //check wheather the ultrasonic sensor's value stays between 1 to 15.
                                                  //If the condition is 'true' then the statement below will execute:
  //Move Forward:
  Motor1.setSpeed(130);  //define motor1 speed:
```

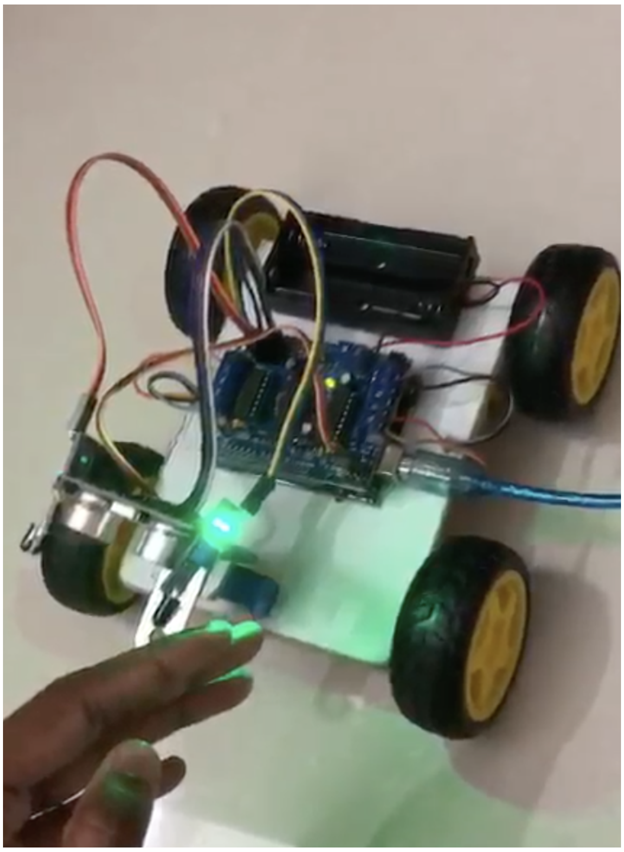
```
}else if((Right_Value==0) && (Left_Value==1)) {   //If the condition is 'true' then the statement below will execute:

  //Turn Left
  Motor1.setSpeed(150);  //define motor1 speed:
  Motor1.run(FORWARD);   //rotate motor1 cloclwise:
  Motor2.setSpeed(150);  //define motor2 speed:
  Motor2.run(FORWARD);   //rotate motor2 clockwise:
  Motor3.setSpeed(150);  //define motor3 speed:
  Motor3.run(BACKWARD);  //rotate motor3 anticlockwise:
  Motor4.setSpeed(150);  //define motor4 speed:
  Motor4.run(BACKWARD);  //rotate motor4 anticlockwise:
  delay(150);

}else if((Right_Value==1)&&(Left_Value==0)) {     //If the condition is 'true' then the statement below will execute:

  //Turn Right
  Motor1.setSpeed(150);  //define motor1 speed:
  Motor1.run(BACKWARD);  //rotate motor1 anticlockwise:
  Motor2.setSpeed(150);  //define motor2 speed:
  Motor2.run(BACKWARD);  //rotate motor2 anticlockwise:
  Motor3.setSpeed(150);  //define motor3 speed:
  Motor3.run(FORWARD);   //rotate motor3 clockwise:
  Motor4.setSpeed(150);  //define motor4 speed:
  Motor4.run(FORWARD);   //rotate motor4 clockwise:
  delay(150);
```

OUTPUT:



REFERENCES:

[1] H. Takemura, N. Zentaro, and H. Mizoguchi, “Development of vision-based person following module for mobile robots in/outdoor environment,” in 2009 IEEE International Conference on Robotics and Biomimetics (ROBIO), 2009.

[2] K. Morioka, J.-H. Lee, and H. Hashimoto, “Human-following mobile robot in a distributed intelligent sensor network,” IEEE Trans. Ind. Electron., vol. 51, no. 1, pp. 229–237, Feb. 2004.

[3] N. Bellotto and H. Hu, “Multisensor integration for human-robot interaction,” IEEE J. Intell. Cybern.Syst., vol.1, no. 1, p. 1, 2005.

[4] Y. Matsumoto and A. Zelinsky, “Real-time face tracking system for human-robot interaction,” in 1999 IEEE International Conference on Systems, Man, and Cybernetics, 1999. IEEE SMC ’99 Conference Proceedings, 1999, vol. 2, pp. 830– 835 vol.2.