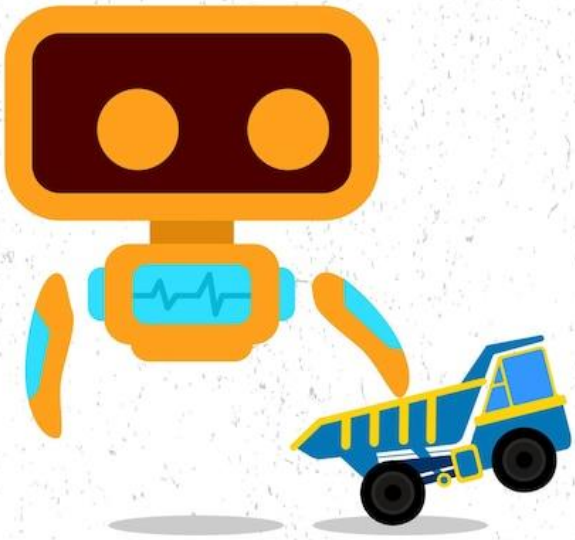
Faculty of Engineering

Alexandria University

Electronics and communication department



Obstacle Avoiding Car Using IR Sensor

Microprocessor Hardware Project

By

Names	ID
Ahmed Sameh Taha	21010083
Adham Ahmed Gamal El-din	21010220
Amira Mohamed Mokhtar	21010308
Karen Mostafa Mohamed	21010966
Rewan Gamal Abdel-Kader	21010540

Supervised by:

Dr. Nayera Sadek

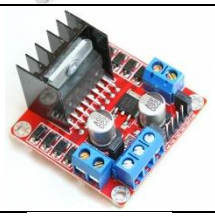
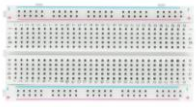
Eng. Abd-Elkareem

Index

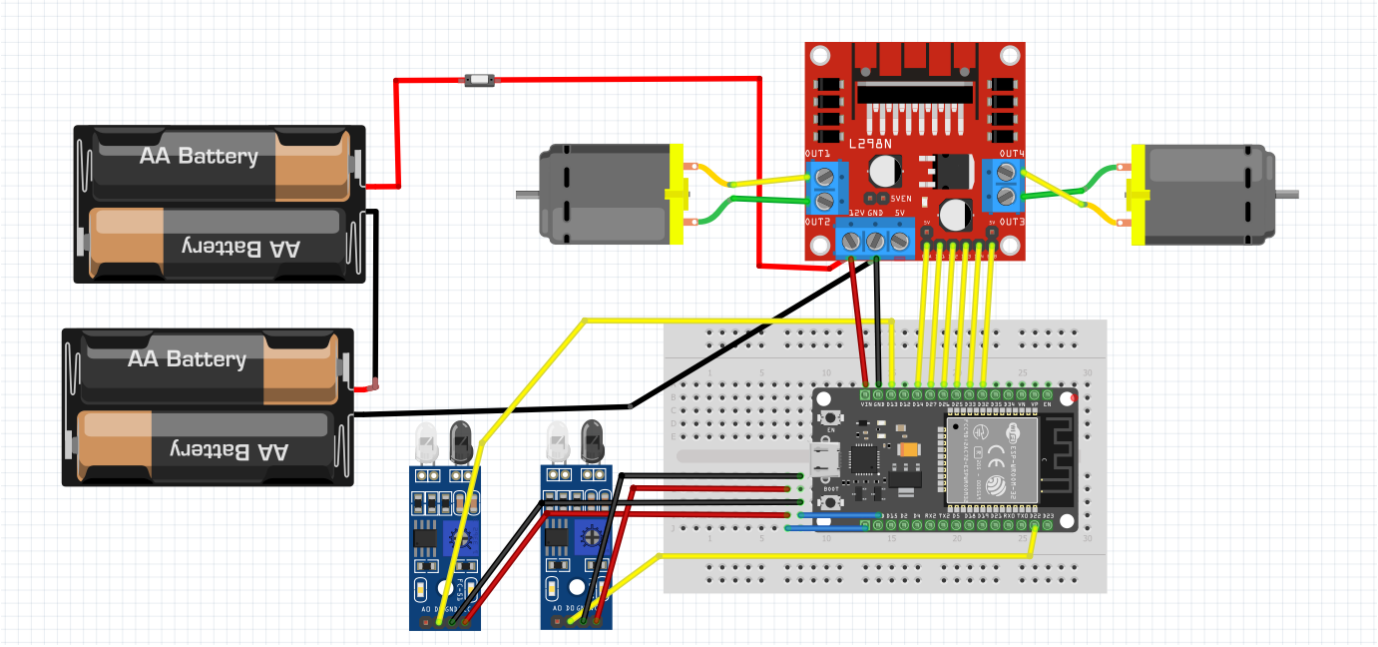
Aim of the hardware project

An obstacle avoiding car is designed to navigate through its environment while avoiding obstacles that are in its path. Infrared sensor is used in the car that allows it to detect obstacles. Then ESP32 microcontroller was used in the project and it was programmed using the Arduino Integrated Development Environment (IDE). ESP32 processes the data received from the sensor and determines the best way to navigate around the obstacles then it decides the best way to move the robot.

List of the used components

Component	Quantity	Photo
ESP32	1	
Infrared Sensor 3 Pin	2	
2WD Smart Robot Car Chassis Kit	1	
L298 Motor Driver Module	1	
AA Batteries	4	
Breadboard	1	
Jumpers	20	

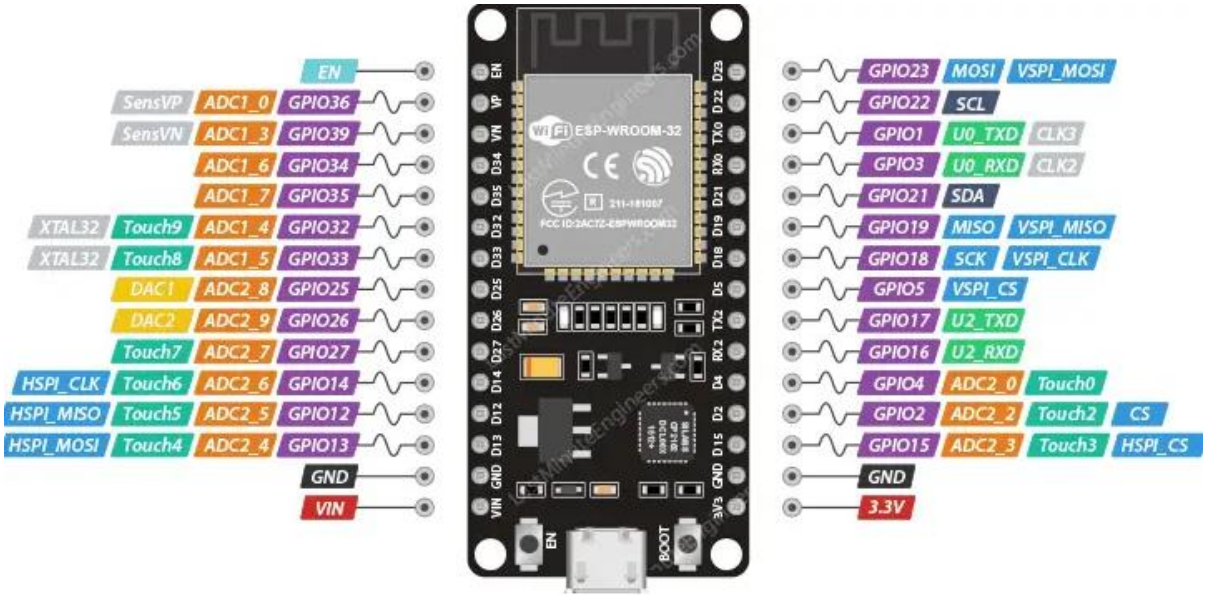
Schematic of the circuit implemented



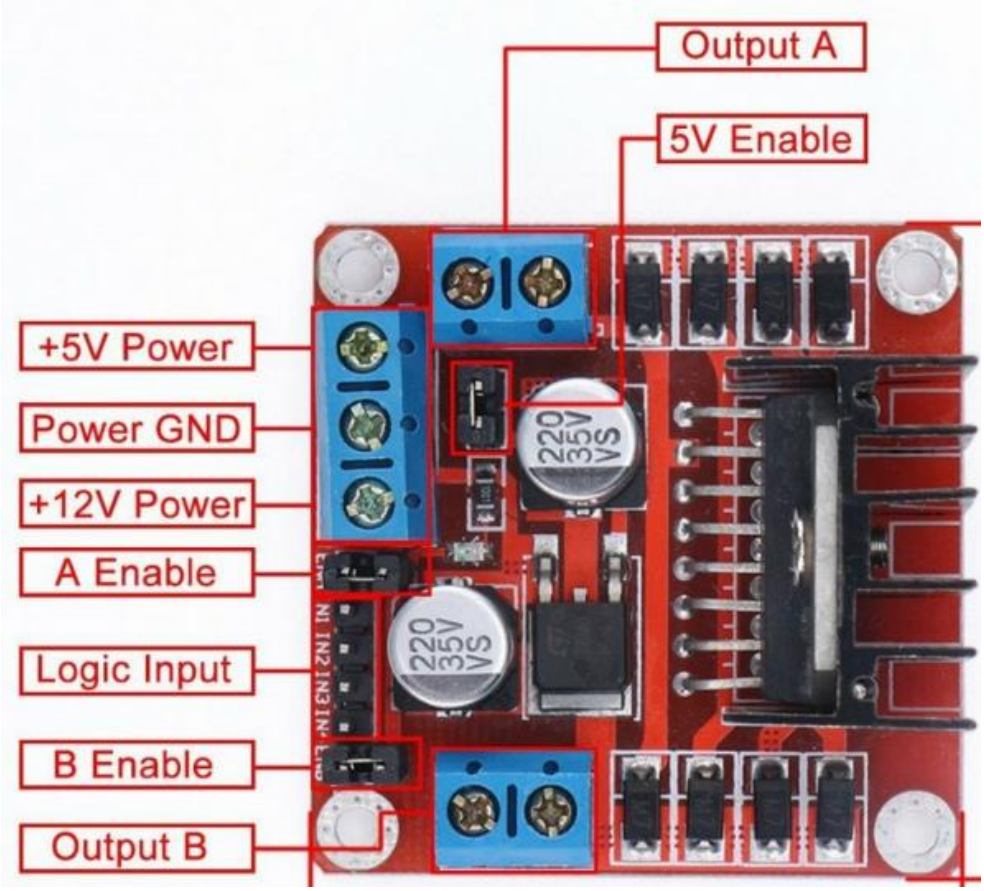
Procedure to use this circuit

Components pins data sheet:

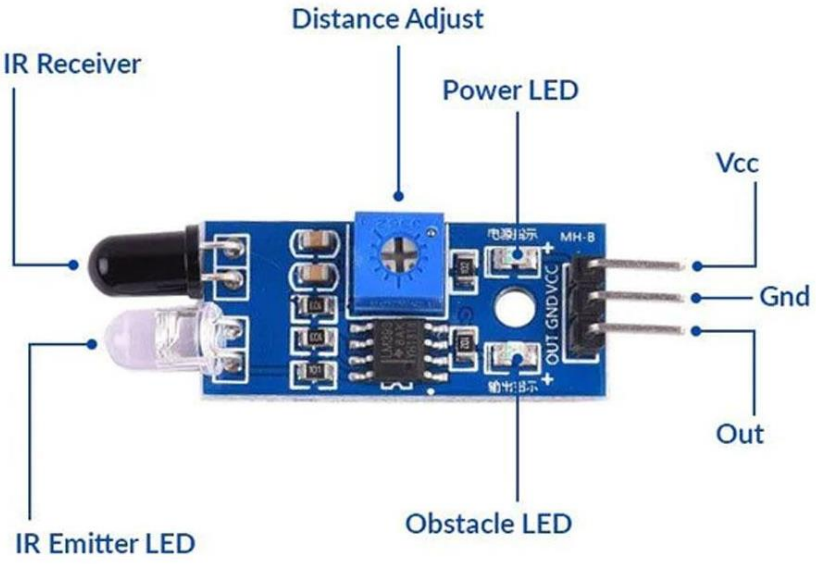
Esp32:



L298 Motor Driver Module:



IR Sensor:



Connections:

- Connect Vin pin of the esp32 to +12v pin in Driver Module.
- Connect GND pin of the esp32 to GND pin in Driver Module.
- Connect pins D14, D27, D26, D25, D33 and D32 in esp32 to ENA, In1, In2, In3, In4 and ENB in Driver Module respectively.
- Connect 3V3 pin of the esp32 to Vcc pins of the two IR Sensors.
- Connect the other GND pin in esp32 to GND pins of the two IR Sensors.
- Connect D13 and D22 pin in esp32 to Output pin of Left and right IR Sensors respectively.
- Connect Output A pins of the Driver Module to motor of the first wheel.
- Connect Output B pins of the Driver Module to motor of the second wheel.

Code:

```
//definition of motor pins
int RIR = 22 ;
int LIR = 13 ;

#define ena 14
#define in1 27
#define in2 26
#define in3 25
#define in4 33
#define enb 32

void forward (){

        digitalWrite(in1, HIGH);
        digitalWrite(in2, LOW);
        digitalWrite(in3, HIGH);
        digitalWrite(in4, LOW);
        analogWrite(ena, 200);
        analogWrite(enb, 200);

        Serial.println("Forward");


}

void backward (){

        digitalWrite(in1, LOW);
        digitalWrite(in2, HIGH);
        digitalWrite(in3, LOW);
        digitalWrite(in4, HIGH);
        analogWrite(ena, 150);
        analogWrite(enb, 150);
        delay(500);

        Serial.println("Back");


}
```

```
void leftward (){

        digitalWrite(in1, HIGH); //Right wheel moves forward
        digitalWrite(in2, LOW);
        digitalWrite(in3, LOW);  //Left wheel moves backward
        digitalWrite(in4, HIGH);
        analogWrite(ena, 200);
        analogWrite(enb, 200);
        delay(500);

        Serial.println("Left");

}

void rightward (){

        digitalWrite(in1, LOW);  //Right wheel moves backward
        digitalWrite(in2, HIGH);
        digitalWrite(in3, HIGH); //Left wheel moves forward
        digitalWrite(in4, LOW);
        analogWrite(ena, 200);
        analogWrite(enb, 200);
        delay(500);

        Serial.println("Right");

}

void stop (){

        analogWrite(ena, 0);
        analogWrite(enb, 0);
        delay(1000);

}

void setup () {
//serial monitor for checking values of ir
Serial.begin(9600);      //setting up o/p pin
pinMode (RIR , INPUT) ;  //IR sensor output is fed into ESP pin 13 as an input
pinMode (LIR , INPUT) ;

pinMode (in2, OUTPUT) ;  //motor1
pinMode (in1, OUTPUT) ;
pinMode (ena, OUTPUT) ;

pinMode (in4, OUTPUT) ;  //motor2
pinMode (in3, OUTPUT) ;
pinMode (enb, OUTPUT) ;

stop ();
}

void loop() {
    int reading_right = digitalRead(RIR);
    int reading_left = digitalRead(LIR);
    //to get readings from IR sensor
    if ( reading_right == 0 & reading_left == 1) {              //in case of right obstacle
      stop();
      backward ();
      leftward ();
    }

    else if (  reading_right == 1 & reading_left == 0 ) {       //in case of left obstacle
        stop();
        backward ();
        rightward();
    }
    else if ( reading_right == 0 & reading_left == 0) {         //in case of left and right obstacles
      stop();
      backward ();
      rightward ();
    }

    else {                                                      //in case of no obstacle

     forward();

    }
}
```

Code Explanation:

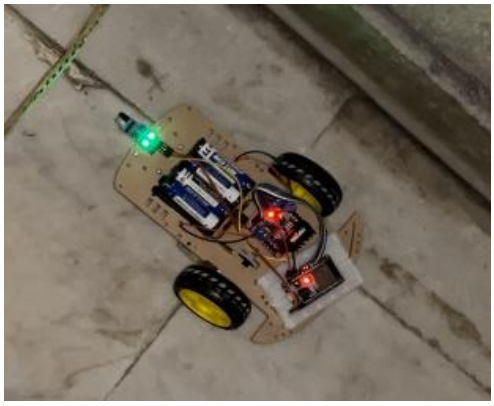
- Define the output pins of the two IR sensors and the input pins of Driver Module. by the esp32 pins.
- **Forward:** the right and the left wheel move forward when In1 and In3 are high (1) logic level, In2 and In4 low (0) logic level.
- **Backward**: the right and the left wheel move backward when In1 and In3 are low (0) logic level, In2 and In4 high (1) logic level.
- **Leftward:** the right wheel moves forward when In1 high (1) logic level and In2 low (0) logic level, the left wheel moves backward when In3 low (0) logic level and In4 high (1) logic level.
- **Right**: the left wheel moves forward when In3 high (1) logic level and In4 low (0) logic level, the right wheel moves backward when In1 low (0) logic level and In2 high (1) logic level.
- **To get the reading of IR sensor in case of right obstacle:** reading_right == 0 (obstacle) and reading_left == 1 (no obstacle) ,the car moves leftward.
- **In case of left obstacle:** reading_right == 1 (no obstacle) and reading_left == 0 (obstacle), the car moves rightward.
- **In case of left and right obstacles:** reading_right == 0 (obstacle) and reading_left == 0 (obstacle), the car stops.
- **In case of no obstacle:** otherwise the car moves forward.

Budget of the project

Component	Price
1x ESP32	400
2x Infrared Sensor 3 Pin	60
1x 2WD Smart Robot Car Chassis Kit	215
1x L298 Motor Driver Module	85
4x AA Batteries	80
Jumpers	20
Total	860

Challenges

Sensitivity of IR sensor:



At first we used only one IR sensor, the sensor doesn't see left and right obstacle, only sees the obstacle directly in front of it. Instead we used two IR sensors, but we face another challenge that it's hard to control the sensitivity of the two IR sensors by the potentiometer exactly the same.

IR sensor and Sunlight:

Infrared (IR) sensors may not work effectively in direct sunlight because sunlight contains a large amount of infrared radiation, which can interfere with the sensor's ability to accurately detect other sources of infrared radiation. This interference can make it difficult for the sensor to distinguish between the infrared radiation it is meant to detect and the ambient infrared radiation from the sunlight. Additionally, sunlight can also create variations in temperature that can affect the sensor's performance. To mitigate these issues, IR sensors may be designed with filters or shielding to reduce the impact of sunlight and other ambient sources of infrared radiation.

References

- Cameron, N. (2023). ESP32 Microcontroller. In ESP32 Formats and Communication: Application of Communication Protocols with ESP32 Microcontroller (pp. 1-54). Berkeley, CA: Apress.
- Arulananth, T. S., & Baskar, M. (2020). IR Sensor based obstacle detection and avoiding robot. Palarch's Journal of Archaeology of Egypt/Egyptology, 17(9), 3328-3336.
- Ismail, R., Omar, Z., & Suaibun, S. (2016, October). Obstacle-avoiding robot with IR and PIR motion sensors. In IOP Conference Series: Materials Science and Engineering (Vol. 152, No. 1, p. 012064). IOP Publishing.
- Dube, B., Kazi, R., Malya, A., & Gala, N. (2019, March). IR-Based Obstacle Avoiding and Self-Navigating Robot. In 2019 IEEE 5th International Conference for Convergence in Technology (I2CT) (pp. 1-4). IEEE.