AI IN ROBOTICS

PROJECT REPORT

OBSTACLE AVOIDING ROBOT

SUBMITTED BY

ARINDAM SINGH(RK21PGB58)

NALAWADE NISHANT(RK21PGB57)

AMAN KUMAR VERMA(RK21PGB56)

LOVELY PROFESSIONAL UNIVERSITY,

JALANDHAR,PUNJAB

ABSTRACT

The project is design to build an obstacle avoidance robotic vehicle using ultrasonic sensors for its movement. A micro-controller is used to achieve the desired operation.

A robot is a machine that can perform task automatically or with guidance. Robotics is a combination of computational intelligence and physical machines (motors). Computational intelligence involves the programmed instructions.

This robot or a smart car is built to sense any obstacle in its path, to avoid it and resume its running involving the pre-computation of an obstacle free path. Ultrasonic sensors were adapted to implement a real-time obstacle avoidance system for wheeled robots, so that the robot can continually detect surroundings, avoid obstacles, and move toward the target area.

INTRODUCTION

An obstacle avoiding robot is an intelligent device, which can automatically sense and overcome obstacles on its path. Obstacle Avoidance is a robotic discipline with the objective of moving vehicles on the basis of the sensorial information. The use of these methods front to classic methods (path planning) is a natural alternative when the scenario is dynamic with an unpredictable behaviour. In these cases, the surroundings do not remain invariable, and thus the sensory information is used to detect the changes consequently adapting moving. It will automatically scan the surrounding for further path.

HARDWARE CONFIGURATION

Arduino Uno

Ultrasonic Range Sensor – HC – SR04

Motor Driver IC – L293D

DC Motors x 2

Power Batteries

ArduinoUno:

Arduino is a widely used open-source prototyping platform which is based on software and hardware that are easy to use. It is made up of a programmable microcontroller unit and the Arduino Integrated Development Environment (IDE) which is used to write and upload codes to the Arduino board. The Arduino board is the physical microcontroller boards used for making electronic projects. Arduino boards are capable of reading inputs and converting them into outputs. You can send instructions to the microcontroller on the Arduino board and tell it what to do using the Arduino programming language (based on Wiring).

Motor Driver IC – L293D:

Arduino board is connected with DC Motor through Motor driver board(pin10, pin11, pin12, pin13)which provide power to the actuators. Actuators are used to move robot in Forward, Backward, Left and Right directions.

DC Motors x 2 :

DC motors normally have just two leads, one positive and one negative. If you connect these two leads directly to a battery, the motor will rotate. If you switch the leads, the motor will rotate in the opposite direction.

WORKING

The obstacle avoidance robotic vehicle uses ultrasonic sensors for its movements. Ardumo Is used to achieve the desired operation. The motors are connected through motor driver IC to Ardumo. The ultrasonic sensor is attached in front of the robot.

Whenever the robot is going on the desired path the ultrasonic sensor transmits the uitrasone waves continuously from its sensor head. Whenever an obstacle comes ahead of it the ultrasonic waves are reflected back from an object and that information is passed to the arduino. The arduino controls the motors left, right, back, front, based on ultrasonic signals.

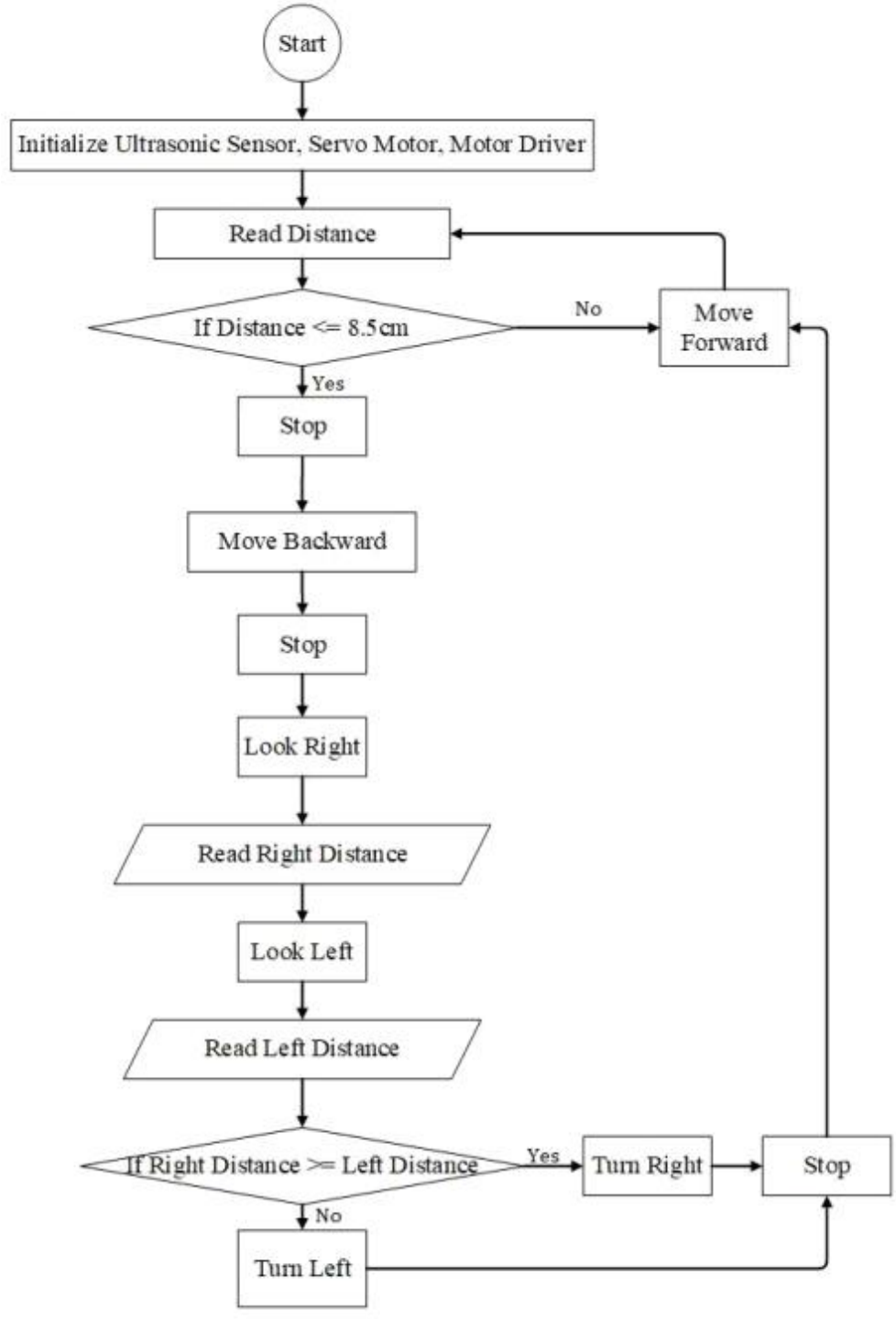
In order to control the speed of each motor pulse width moduration is used (PWM).

When ultrasonic sensor detect the object which is kept inside the path it will send the signal toward the arduino uno and according to that it will it will rotate the motor M3 & M4 in forward direction and rotate the motor MI & M2 in reverse direction such way that the car get moving in let direction.

Similarly in every time when ever an obstacle in found to be in path of car it will detect it and rotate the car in left direction to avoid the obstacle.

FLOW CHART

Flowchart of the program



Arduino Code :

```
#include <Servo.h> //Servo motor library. This is standard library
#include <NewPing.h> //Ultrasonic sensor function library. You must install this library
//our L298N control pins
const int LeftMotorForward = 7;
const int LeftMotorBackward = 6;
const int RightMotorForward = 4;
const int RightMotorBackward = 5;
//sensor pins
#define trig_pin A1 //analog input 1
#define echo_pin A2 //analog input 2
#define maximum_distance 200
boolean goesForward = false;
int distance = 100;
NewPing sonar(trig_pin, echo_pin, maximum_distance); //sensor function
Servo servo_motor; //our servo name
void setup(){
  pinMode(RightMotorForward, OUTPUT);
  pinMode(LeftMotorForward, OUTPUT);
  pinMode(LeftMotorBackward, OUTPUT);
  pinMode(RightMotorBackward, OUTPUT);
  servo_motor.attach(10); //our servo pin
  servo_motor.write(115);
  delay(2000);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
  distance = readPing();
  delay(100);
}
```

```
void loop(){
  distance = readPing();
  delay(10);
  int distanceRight = 0;
  int distanceLeft = 0;
  delay(10);
  if (distance <= 20){
    moveStop();
    delay(300);
    moveBackward();
    delay(400);
    moveStop();
    delay(300);
    distanceRight = lookRight();
    delay(300);
    distanceLeft = lookLeft();
    delay(300);
    if (distanceRight >= distanceLeft){
      turnRight();
      moveStop();
    }
    else{
      turnLeft();
      moveStop();
    }
  }
  else{
    moveForward();
  }
    distance = readPing();
}
int lookRight(){
  servo_motor.write(50);
  delay(500);
  int distance = readPing();
```

```
  delay(100);
  servo_motor.write(115);
  return distance;
}
int lookLeft(){
  servo_motor.write(170);
  delay(500);
  int distance = readPing();
  delay(100);
  servo_motor.write(115);
  return distance;
  delay(100);
}
int readPing(){
  delay(70);
  int cm;
  cm = sonar.ping_cm();
  if (cm==0){
    cm=250;
  }
  return cm;
}
void moveStop(){
  digitalWrite(RightMotorForward, LOW);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorBackward, LOW);
  digitalWrite(LeftMotorBackward, LOW);
}
void moveForward(){
  if(!goesForward){
    goesForward=true;
    digitalWrite(LeftMotorForward, HIGH);
    digitalWrite(RightMotorForward, HIGH);
    digitalWrite(LeftMotorBackward, LOW);
    digitalWrite(RightMotorBackward, LOW);
```

```
  }
}
void moveBackward(){
  goesForward=false;
  digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorBackward, HIGH);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorForward, LOW);
}
void turnRight(){
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorBackward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorForward, LOW);
  delay(500);
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorBackward, LOW);
 }
void turnLeft(){
  digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(RightMotorBackward, LOW);
  delay(500);
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorForward, HIGH);
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(RightMotorBackward, LOW);
}
```

Application of Obstacle Avoiding Robots

This device has application in surveying different landscapes and mapping them.It can also be used in commercial devices like

1)Automated lawn mover

2)Smart room cleaner hetc

3) Obstacle avoiding robots can be used in almost all mobile robot navigation systems.

4) They can also be used in dangerous environments, where human penetration could be fatal.

.

5)Unmanned vehicle driving

6)Mining Vehicle that uses Obstacle Detection.

Conclusion

The goal of My project is to create a autonomous robot which intelligently detects

the obstacle in his path and navigate according to the actions that I set for it.

This project developed an obstacle avoiding robot to detect and avoid obstacles in its path. The robot is built on the Arduino platform for data processing and its software counterpart helped to communicate with the robot to send parameters for guiding movement. For obstacle detection, three ultrasonic distance sensors were used that provided a wider field of detection. The robot is fully autonomous and after the initial loading of the code, it requires no user intervention during its operation. When placed in unknown environment with obstacles, it moved while avoiding all obstacles with considerable accuracy.