|  |  |
| --- | --- |
|  | Project Report |
|  |  |
|  | Computer Peripheral and Interfacing  [Course title]  31/1/17 |

**Project Title**: Obstacle avoiding Robot.

**Project Description:**

Our project is a smart robot which can detect obstacles and avoid them when detected. The robot moves in an auto pilot mode.

When the robot is powered on, both the motors of the robot will run normally and the robot moves forward. During this time, the ultrasonic sensor continuously calculates the distance between the robot and the reflective surface.

This information is processed by the Arduino. If the distance between the robot and the obstacle is less than 5cm, the left wheel motor is stopped and the right wheel motor is operated normally.

This will rotate the robot towards right. This rotation continues until the distance between the robot and any obstacle is greater than 5cm. The process continues forever and the robot keeps on moving without hitting any obstacle.

If it senses an obstacle in front of it then it senses its left. If no obstacle is detected, then turns left and moves. Otherwise it senses its right and turns right is no obstacle is detected. If it detects obstacle in its front, left and right, then it moves backward until obstacle is not detected.

**Hardware Required:**

* Arduino Uno
* Bread Board
* HC-SR04 ultrasonic sensor
* DC motors
* Wheels
* LED
* 220 Ohm Resistor
* 9 V Battery
* Jumper wires, bread board wires, nuts and screws
* Robot chassis

**Tools:**

* Drill machine
* Super glue
* Screw driver
* Hot gun glue

**Design of our robot:**

Arduino is the main processing unit of the robot. Out of the 14 available digital I/O pins, 10 pins are used in this project design.

Three ultrasonic sensor are used. The ultrasonic sensor has 4 pins: Vcc, Trig, Echo and Gnd. Vcc and Gnd are connected to the supply pins of the Arduino. Trig is connected to the 11th pin and Echo is connected to 10th pin of the Arduino.

L293D is a 16 pin IC. Pins 1 and 9 are enable pins. They are connected to Vcc. Pins 2 and 7 are control inputs from microcontroller for first motor. They are connected to pins 9 and 8 of Arduino respectively.

Similarly, pins 10 and 15 are control inputs from microcontroller for second motor. They are connected to pins 4 and 3 of Arduino. Pins 4, 5, 12 and 13 of L293D are ground pins and are connected to Gnd.

First motor (consider this as the motor for left wheel) is connected across the pins 3 and 6 of L293D. The second motor, which acts as the right wheel motor, is connected to 11 and 14 pins of L293D.

The 16th pin of L293D is Vcc1. This is connected to 5V Vcc. The 8th pins is Vcc2. This is the motor supply voltage. This can be connected anywhere between 4.7V and 36V. In this project, pin 8 if L293D is connected to 9V supply.

**Future Plan:**

We have a plan to make the robot smarter using artificial intelligence and by adding some other sensors. We have also a plan to add an IP camera for taking photos and videos so that we can use this in security purpose. We are trying to add the feature of controlling by android.

**Project goals:**

* Obstacle avoiding robots can be used in almost all mobile robot navigation systems.
* Can be used for household work like automatic vacuum cleaning.
* Can also be used in dangerous environments, where human penetration could be fatal.

**Project Sketch:**

*#define ledPin 8 //this is the led pin*

*#define ledPin2 9*

*#define leftPin 6 //not working*

*#define rightPin 7*

*class Sonar{*

*public:*

*int triggerPin;*

*int echoPin;*

*long distance, duration;*

*void sonar(int trig, int echo){ //custom initializer*

*triggerPin = trig;*

*echoPin = echo;*

*pinMode(triggerPin, OUTPUT);*

*pinMode(echoPin, INPUT);*

*}*

*};*

*class Testing{*

*public:*

*static bool isObstaclePresent(Sonar s){ //checking an obstacle is within less than or equal five centimeters*

*digitalWrite(s.triggerPin, LOW);*

*delayMicroseconds(2);*

*digitalWrite(s.triggerPin, HIGH);*

*delayMicroseconds(10);*

*digitalWrite(s.triggerPin, LOW);*

*s.duration = pulseIn(s.echoPin, HIGH);*

*s.duration /= 29; //for converting miliseconds to centimeters*

*s.duration /= 2; //for getting the actual distance*

*s.distance = s.duration;*

*if(s.distance <= 20){ //checking if the distance is less or equal five*

*Serial.print("this is for sonar: ");*

*Serial.println(s.distance);*

*// delay(40);*

*return true;*

*}*

*else{*

*Serial.print("this is for sonar: ");*

*Serial.println(s.distance);*

*// delay(40);*

*return false;*

*}*

*}*

*};*

*Sonar sonar1;*

*Sonar sonar2;*

*//Sonar sonar3;*

*void setup() {*

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

*sonar1.sonar(2, 3); // sonar(trig, eho);*

*sonar2.sonar(4, 5);*

*//sonar3.sonar(10, 11);*

*Serial.begin(9600);*

*pinMode(leftPin, OUTPUT);*

*pinMode(rightPin, OUTPUT);*

*pinMode(ledPin2,OUTPUT);*

*pinMode(ledPin,OUTPUT);*

*}*

*void loop() {*

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

*if((Testing::isObstaclePresent(sonar1) == true) ){ //checking the obstacle for sonar1*

*digitalWrite(ledPin,HIGH);*

*digitalWrite(leftPin, LOW);*

*delay(40);*

*// Serial.println(sonar1.distance);*

*//Serial.println(sonar2.distance);*

*// Serial.println(sonar3.distance);*

*}*

*else{*

*digitalWrite(ledPin, LOW);*

*digitalWrite(leftPin, HIGH);*

*}*

*if((Testing::isObstaclePresent(sonar2) == true)){*

*digitalWrite(ledPin2,HIGH);*

*digitalWrite(rightPin, LOW);*

*delay(40);*

*}*

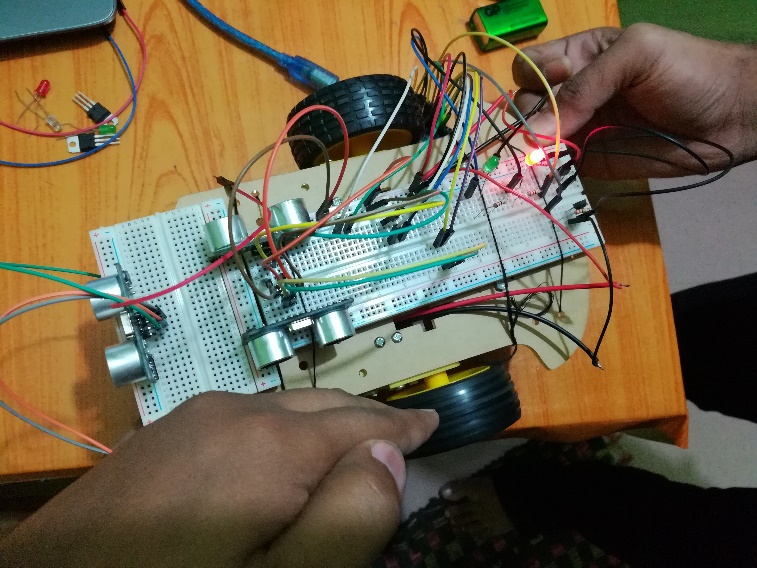
*else {*

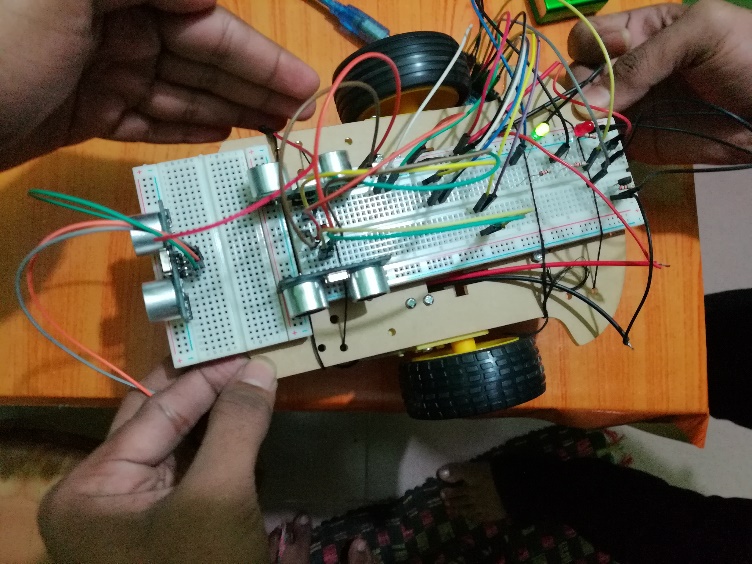
*digitalWrite(ledPin2,LOW);*

*digitalWrite(rightPin, HIGH);*

*}*

*}*

****Output:**

**

