International Journal of Soft Computing and Engineering (IJSCE)
ISSN: 2231-2307, Volume-6 Issue-1, March 2016

Improved Design of Obstacle Avoidance Robot using three Ultrasonic Sensors and ATMEGA328P Microcontroller

Mukesh Kumar Dey

Abstract— This paper proposes an alternative design for a cost effective and simplified version of the obstacle avoidance robot using three ultrasonic sensors. It also provides a dynamic algorithm which directs the robot to navigate smoothly in different environments, avoiding obstacles. The main objective of this paper is to improve the accuracy of the robot in detection and avoidance of obstacles at various angles. The paper talks about the design of the robot which is followed by the working principle of sensors and algorithm for the microcontroller. The paper finally makes a detailed explanation of the code for the obstacle avoidance strategy and draws a conclusion based on the designed robot.

Index Terms— ATMEGA328P, Binary Logic, HC-SR04, Dynamic algorithm.

I. INTRODUCTION

A number of methods for robot automation and navigation has been developed in recent years such as wall-following, edge-following, human-following and the obstacle avoidance robot. Some wall-following robots have been employed for a number of tasks such as floor cleaning for long hallways [1], others such as edge detection robots have been implemented for various solving real-time maze games puzzles [2], [3]. One drawback of such methods of detection is the requirement of an obstacle or source such as a wall or a human for navigation. This makes the navigation of the robot limited. Also, the robot needs to stop in front of the obstacle in order to detect it, and then take adequate actions. Though some robots have already been developed to reduce the need of it to stop in front of an obstacle, those includes sophisticated algorithms and complex structures to function. The more dynamic version of robot auto-navigation developed is the obstacle avoidance robot [4]. It dynamically navigates through different environments without the existence of a source, and changes its course on detection of an obstacle. Though this robot has managed to reduce the complexity and sophistication of obstacle avoidance algorithms, its accuracy of detection and collision avoidance at various angles of sensing for real-time application is still limited. This paper proposes a modified version of the obstacle avoidance algorithm and design of the robot based on more accurate sensing and measurements. Thus, the design may solve some of the issues faced with the earlier algorithms for better navigation and obstacle avoidance. The design proposed can be further improved depending upon the required action.

Revised Version Manuscript Received on February 18, 2016.

Mukesh Kumar Dey, School of Electronics and Communication Engineering (SENSE), VIT University, Vellore, India.

II. BASIC DESIGN OF ROBOT

The brain of the obstacle avoidance robot is the ATMEGA328P microcontroller development board. The robot consists of two 200rpm DC Motors interfaced to the microcontroller with the L293D Motor Shield. The positive pins of the right and left motors are connected to M1+ and M2+ pins of the Motor Shields respectively, while the negative pins are connected to the M1- and M2- pins. Three Ultrasonic Sensors have been placed for obstacle detection- Sensor 1 (0°) is taken as reference, Sensor 2 is placed at -30° and Sensor 3 is placed at +30° with respect to Sensor 1.

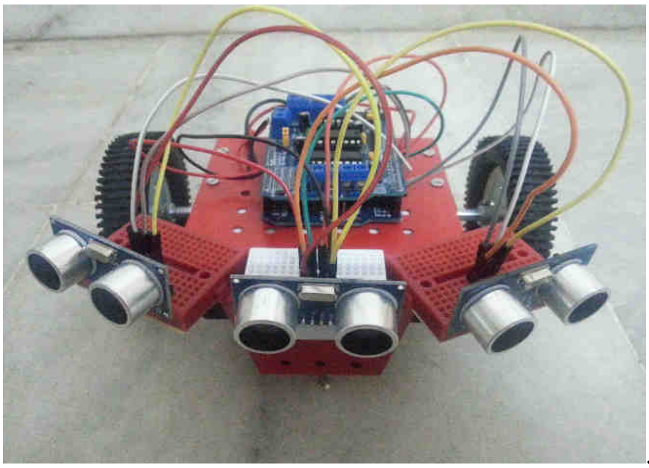

Fig. 1: Position of Ultrasonic Sensors

The Detection range of sensors is set to 20 cm which can be varied as per the requirements.

III. SENSORS FOR OBSTACLE AVOIDANCE

A number of sensors are available for obstacle detection. Some are Ultrasonic Sensors, Infrared Sensors, Cameras, LIDAR which measures obstacles over thousands of points in its field of view [5]. Since accuracy with reduced cost is the objective of the current study, the HC-SR04 Ultrasonic Sensor was selected for the design. The HC-SR04 Ultrasonic Sensor occupy reasonable ground in terms of cost as compared to other sensors. Another advantage of Ultrasonic Sensor is its ability to sense in high light intensity environments, which is limited for low cost devices such as IR Sensors. As accuracy is the key purpose of the design, the use of Ultrasonic Sensor serves well to meet the requirements. The HC-SR04 Ultrasonic Sensor is equipped to detect objects from a range of 2-400cm [6]. The module works at 5V DC Power Supply and a working current of 15mA. It requires a 10µS trigger pulse to initiate the triggering of waves for detection. Upon triggering, the transmitter directs eight cycles of ultrasound waves at 40 kHz frequency. If an obstacle

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

exist(s) in its course, the sound returns backwards to be received by the receiver of the module. The distance is then calculated as:

Obstacle Distance = [Time between transmitting and Receiving of Signal*Velocity of Sound (340m/s)]/2

IV. ALGORITHM - WORKING PRINCIPLE

The working of the robot involves the transmission of the sensed signal to the microcontroller to control the DC Motors for obstacle avoidance. As we have used three different sensors, let us name Sensor 1 (Forward Sensor, Reference) as S1, Sensor 2 (Left Sensor) as S2 and Sensor 3 (Right Sensor) as S3. For the DC Motors, let us name LMF if the motor moves in the forward (clockwise) direction, LMB if the motor moves in the backward (anti-clockwise) direction. Similarly, RMF and RMB for forward and backward directions for the right motor. Using the convention of Binary Logic, let us assume the detection of an obstacle by a sensor as logical ‘1’ and when no obstacle is detected, the sensor’s output is logical ‘0’. For the DC Motors, the output from the microcontroller to the terminals of the motors can be taken as ‘1’. The H-Bridge L293D Motor Shield controls the direction of the motors to move in either clockwise or anti-clockwise directions as per the instructions provided by the ATMEGA328P Microcontroller [7]. Initially, the robot starts moving in the forward direction and all the three sensors simultaneously starts sensing the obstacles in its path. When no obstacle is detected (S1=S2=S3=0), the robot continues to move in the forward direction (LMF and RMF). When an obstacle is detected in the forward direction (S1=1 and S2=S3=0), the robot turns right to avoid collision (LMF and RMB). Similarly, when an obstacle is detected in the forward as well as left direction (S1=S2=1 and S3=0), the robot again turns right (LMF and RMB). In case, the robot comes to an environment where it senses obstacles in the forward as well as right direction (S1=S3=1 and S2=0), the robot turns left (LMB and RMF) until either S1 and S3 becomes ‘0’. The process is repeated in an infinite loop by the microcontroller so as to avoid collision at any instance of time. The algorithm presents a continuous movement of the robot without any need to reach a specific destination. The algorithm can be summarized in the flow chart as shown in the fig.3.

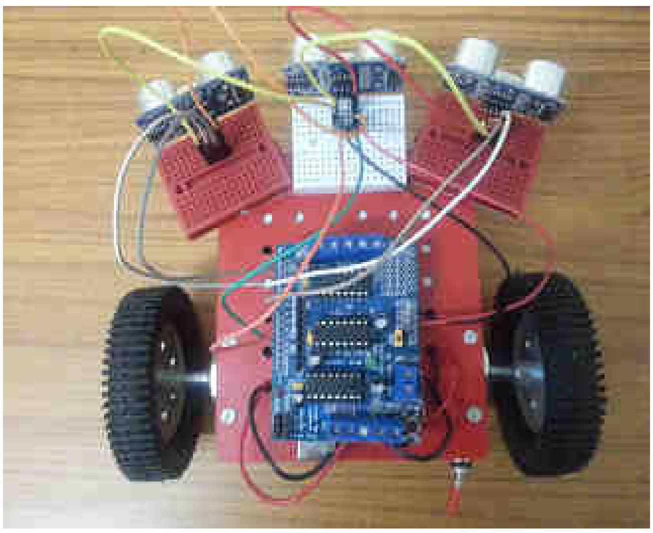

Fig. 2: Basic Connections of the Robot.

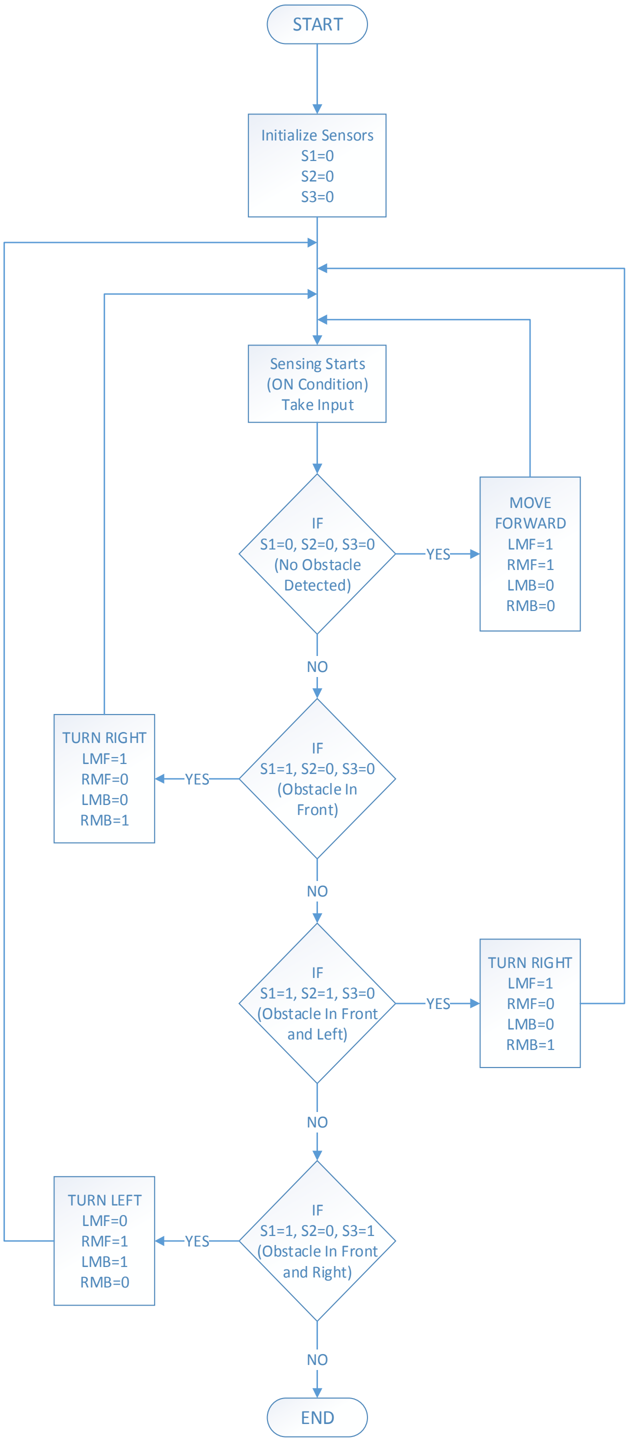

Fig. 3: Obstacle Avoidance Flow Chart

V. IMPLEMENTATION

The implementation of the algorithm involves coding and uploading of the code to the microcontroller. This is achieved by using software Arduino (Version 1.6.7). Once the instructions are coded on the PC, it is then uploaded to the microcontroller using the common USB 2.0 A-B (Male-Male) peripheral connector. After the code has been uploaded to the microcontroller, the robot is then switched on and it can now roam around. It continuously takes the input from the HC-SR04 Ultrasonic Sensors and their outputs are constantly fed to the microcontroller. The Microcontroller then takes decisions based upon the program and provides output for controlling the motors. The general block diagram for the complete execution is as follows:

Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

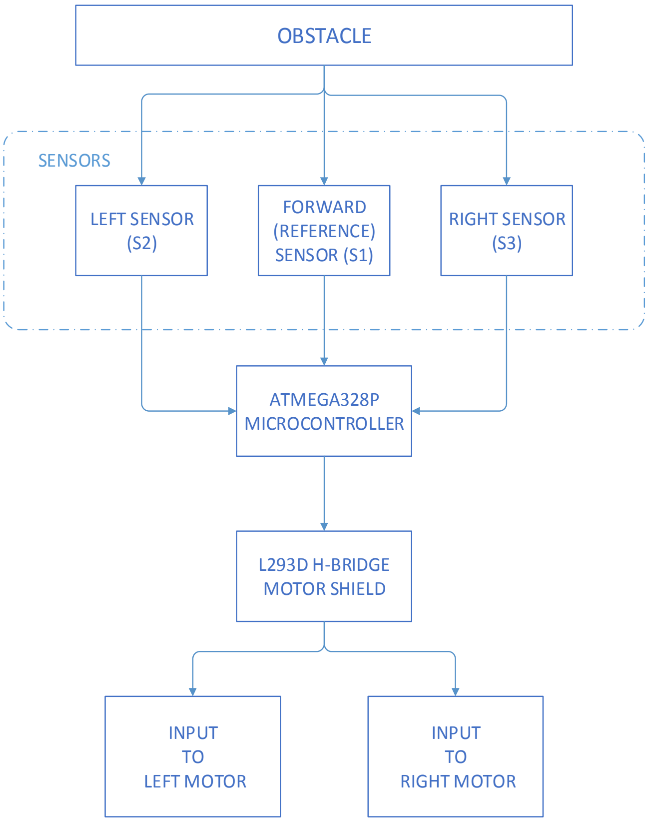

Fig. 4: Block Diagram of Obstacle Avoidance Robot

VI. SOFTWARE IMPLEMENTATION

The code for the ATMEGA328P microcontroller for dynamic obstacle avoidance robot are as follows:

```
#include <AFMotor.h> //Importing the motor shield library
#define trigPin1 13 //Set Pin 13 as Trig Pin for Sensor 1 (Front Sensor)
#define trigPin2 2  //Set Pin 2 as Trig Pin for Sensor 2 (Left Sensor)
#define trigPin3 5  //Set Pin 5 as Trig Pin for Sensor 3 (Right Sensor)
#define echoPin1 12 //Set Pin 12 as Echo Pin for Sensor 1
#define echoPin2 4 //Set Pin 4 as Echo Pin for Sensor 2
#define echoPin3 6 //Set Pin 6 as Echo Pin for Sensor 3

AF_DCMotor motor1(1,MOTOR12_64KHZ); //Set up motor 1 for Motor Shield (Port 1 at 64KHz Frequency)
AF_DCMotor motor2(2, MOTOR12_8KHZ); //Set up motor 2 for Motor Shield (Port 2 at 8KHz Frequency)

long duration, distance;
void setup() {
Serial.begin(9600); // Communication begins
Serial.println("Motor test!");
pinMode(trigPin1, OUTPUT); //Setting the trig pin for Sensor 1 to output (Send sound waves)
pinMode(echoPin1, INPUT); //Setting the echo pin for Sensor 1 to input (recieve sound waves)
pinMode(trigPin2, OUTPUT); //Setting the trig pin for Sensor 2 to output (Send sound waves)
pinMode(echoPin2, INPUT); //Setting the echo pin for Sensor 2 to input (recieve sound waves)
pinMode(trigPin3, OUTPUT); //Setting the trig pin for Sensor 3 to output (Send sound waves)
pinMode(echoPin3, INPUT); //Setting the echo pin for Sensor 3 to input (recieve sound waves)
motor1.setSpeed(50); //Setting the speed of the motor 1 to 200rpm
motor2.setSpeed(50); //Setting the speed of the motor 2 to 200rpm
}

 void loop() {
SonarSensor(trigPin1,echoPin1); //Triggering the Front Sensor for Distance measurement
SonarSensor(trigPin2,echoPin2); //Triggering the Left Sensor for Distance measurement
RightSensor(trigPin3,echoPin3); //Triggering the Right Sensor for Distance measurement
}

void SonarSensor(int trigPin,int echoPin)/* Setting function for Sensor8*/{
digitalWrite(trigPin, LOW); //Setting the trig pin to logical '0'
delayMicroseconds(2); // Setting Delay
digitalWrite(trigPin, HIGH); //Setting the trig pin to logical '1'
delayMicroseconds(10); // Setting Delay
digitalWrite(trigPin, LOW); //Setting the trig pin to logical '0'
duration = pulseIn(echoPin, HIGH); //Assigning the time required for the echo pin to turn 'HIGH' after triggering to duration
distance = (duration/2)/29.1; //Converting duration of the pulse to distance of the robot to the obstacle

if (distance < 20)/*If the sensor detects an obstacle 20 cm at its field of sensing, then do the following: */ {
Serial.println ("Close Obstacle detected at: "); //Printing the value of the distance
Serial.print("Distance = ");
Serial.print(distance);
Serial.print("cm ");
motor1.run(FORWARD); //Left Motor moves forward
motor2.run(BACKWARD); //Right Motor moves backward
}
else {
Serial.println ("No obstacle detected, going forward"); //If no obstacle is detected:
delay (15);
motor1.run(FORWARD); //Left Motor moves forward
motor2.run(FORWARD); //Right Motor moves backward
}
}
void RightSensor(int trigPin,int echoPin)/* Setting function for Sensor8*/{
digitalWrite(trigPin, LOW); //Setting the trig pin to logical '0'
delayMicroseconds(2); //Setting Delay
digitalWrite(trigPin, HIGH); //Setting the trig pin to logical '1'
delayMicroseconds(10); //Setting Delay
digitalWrite(trigPin, LOW); //Setting the trig pin to logical '0'
duration = pulseIn(echoPin, HIGH); //Assigning the time required for the echo pin to turn 'HIGH' after triggering to duration
distance = (duration/2)/29.1; //Converting duration of the pulse to distance from the robot to the obstacle
if (distance < 20)/*If the sensor detects an obstacle 20 cm at its field of sensing, then do the following: */ {
Serial.println ("Close Obstacle detected at: "); //Printing the value of the distance
Serial.print("Distance = ");
Serial.print(distance);
Serial.print("cm ");
```


Published By:
Blue Eyes Intelligence Engineering
& Sciences Publication Pvt. Ltd.

```
motor1.run(BACKWARD);//Left Motor moves backward
motor2.run(FORWARD); //Right Motor moves forward
}
else {
Serial.println ("No obstacle detected, going forward"); //If no
obstacle is detected:
delay (15);
motor1.run(FORWARD); //Left Motor moves forward
motor2.run(FORWARD); //Right Motor moves backward
}
}
```

VII. LIMITATION TO USE AND FURTHER IMPROVEMENTS

Though the robot has been successful in avoiding many obstacles and navigate its course smartly, there are certain limitations as well. Firstly, further improvement can be achieved by the use of only one servo-controlled ultrasonic sensor instead of three sensors. Secondly, a design can also be implemented with the use of other costly and sophisticated sensors for high accuracy applications [8]. Also, use of Computer Vision with Camera features can be implemented for monitoring applications. The design of such a robot is very flexible and various methods can be adapted for its implementation and a comparative analysis of all can be studied.

VIII. CONCLUSION

Avoidance of Obstacles is an essential requirement of any real-time commercial robots and its accuracy is certainly a desirable factor. Their importance in scientific explorations and experiments are vast and especially, for those areas where human reach is limited or dangerous. Some real time applications of robots that are highly researched today involves their capabilities of navigating in a crowded environment such as factories and shopping malls without any collision with obstacles or humans. In conclusion, obstacle avoidance is the foremost requirement for any efficient robot and is widely researched and applied in the field of automated navigation and robotics.

REFERENCES

1. Engelberger, J., Transitions Research Corporation, private communication, 1986.
2. Borenstein, J. and Koren, Y., "Obstacle Avoidance With Ultrasonic Sensors." IEEE Journal of Robotics and Automation. Vol. RA-4, No. 2, 1988, pp. 213-218.
3. Borenstein, J., "The Nursing Robot System." Ph. D. Thesis, Technion, Haifa, Israel, 1987.
4. Vivek Hanumante et al., “Low Cost Obstacle Avoidance Robot”, IJSCE, ISSN: 2231-2307, Volume-3, Issue-4, September 2013.
5. Paul Kinsky and Quan Zhou, “Obstacle Avoidance Robot.” Major Qualifying Project Report, WORCESTER POLYTECHNIC INSTITUTE, Project Number: YR-11E1.
6. Elec Freaks, Ultrasonic Ranging Module HC - SR04. Available: http://www.micropik.com/PDF/**HCSRO4.**pdf (Last Accessed on 16th February, 2016).
7. Adafruit Motor Shield – Arduino Playground. Available: http://www.playground.arduino.cc/Main/Adafruit**MotorShield** (Last Accessed on 31st January, 2016).
8. Jnaneshwar Das, Mrinal Kalakrishnan and Sharad Nagappa, “Obstacle Avoidance for a Mobile Exploration Robot with Onboard Embedded Ultrasonic Range Sensor”, PES Institute of Technology, Bangalore.

Mukesh Kumar Dey is currently pursuing B.Tech degree in the School of Electronics & Communication Engineering (SENSE) at VIT University, Vellore and would be graduating in summer 2017. He has undertaken various microcontroller and VLSI based projects. His field of interest includes Embedded System & VLSI Design, Digital Signal Processing and Microwave. He was awarded ‘The star performer’ in the Winter Training Program-2015 for VLSI & VHDL Course held by Hewlett Packard Enterprise and Ritusha Consultants Pvt. Ltd., Bengaluru.