

# AUTOMATION AND CONTROLLING OF AUTOMATIC FLOOR CLEANER

*A thesis Submitted in partial fulfilment of the requirements for  
the award of the degree of*

Bachelor of Technology  
In  
Mechanical Engineering  
By

**Rajaranjan Senapati  
(111ME0346)**

Under The Guidance of

**Prof. S.K. Behera**



**Department of Mechanical Engineering  
National Institute of Technology Rourkela  
Orissa -769008, India**

**2014-15**

**National Institute of Technology Rourkela**

# CERTIFICATE

This is to certify that the thesis entitled — “**Automation and Controlling of Automatic Floor Cleaner**” submitted to the National Institute of Technology, Rourkela (Deemed University) by **RAJARANJAN SENAPATI, Roll No. 111ME0346** for the award of the Degree of **Bachelor in Technology** in Mechanical Engineering is a record of research work carried out by him under my supervision and guidance. The results presented in this thesis has not been, to the best of my knowledge, submitted to any other University or Institute for the award of any degree or diploma. The thesis, in my opinion, has reached the standards fulfilling the requirement for the award of the degree of Bachelor of Technology in accordance with regulations of the Institute.

Place: Rourkela

Prof. S.K. Behera

Date

Professor

Department of Mechanical Engineering

National Institute of technology,

Rourkela- 769008

# *Acknowledgement*

I express my deep sense of gratitude and indebtedness to my thesis supervisor Prof. Suraj Kumar Behera, Professor of Mechanical Engineering for providing precious guidance, inspiring discussions and constant supervision throughout the course of this work. His timely help, constructive criticism, and conscientious efforts made it possible to present the work contained in this thesis.

I express my sincere thanks to Mr. Swarup Raj Jena, Btech student for their support during the project work. I am also thankful to all the staff members of the department of Mechanical Engineering and to all my well-wishers for their inspiration and help. I am also thankful to my classmates for their help during my project work.

I feel pleased and privileged to fulfil my parent's ambition and I am greatly indebted to them for bearing the inconvenience during my B Tech. course.

Date :

Rajaranjan Senapati

Roll No: 111ME0346

<b>DESCRIPTION</b>	<b>PAGE NO</b>
1. ABSTRACT	1
2. INTRODUCTION	2
2.1 ROBOT	2
2.2 SENSORS	3
2.3 HELP UPON MANKIND	3
3. LITERATURE REVIEW	5
3.1 BODY	5
3.2 INDOOR NAVIGATION	6
3.2.1 MODELLING THE SPIRAL PATH	6
3.2.2 MODELLING OF RANDOM PATH	6
4. DESIGN OF BOT	7
4.1 DRY VACUUM PUMP	7
4.2 WATER SPRINKLER	7
4.3 SCRUBBER SYSTEM	8
4.4 SUCKING OF WET DEBRIS	8
5. AUTOMATION OF MOBILE ROBOT	11
5.1 AUTOMATION	11
5.1.1 SENSORS	11
5.1.2 ARDUINO BOARD	13
5.1.3 MOTOR DRIVER	14
5.1.3.1 CODE FOR MOTOR DRIVER	16
5.1.4 ALGORITHM	18
5.1.4.1 RANDOM STRAIGHT PATH ALGORITHM	18
5.1.4.2 SPIRAL PATH ALGORITHM	19
5.1.5 FINAL PROGRAM	20
6. MAKING ULTRASONIC SENSOR	27
6.1 TRANSMITTER CIRCUIT	28
6.2 RECIEVER CIRCUIT	30
6.3 RANGE CALCULATION	31

7. POSSIBILITY OF COMMERCIALISATION	34
8. CONCLUSION	35
REFERENCE	36

## ***CHAPTER -1***

### **ABSTRACT**

Automatic floor cleaner is a compact robotics system which provides floor cleaning service in room and big offices reducing human labor. Basically as a robot it eliminates human error and provide cleaning activity with much more efficiency. If we clean the floor manually then there is a possibility that the operator will leave some portion of the floor. Also due to manual labor involved this is time consuming and irritating to clean the floor. Also in big offices floor area is very huge and the people involved there for cleaning purpose cannot clean it much more efficiently. This is where the robot comes as an advantage. Also the robot is small and compact in size. So we can carry it and place it wherever we can on the house. Also in industries the robot is very cost effective as compared to manual labor involved. The flexibility, time saving and efficiency make the robot a clean choice for cleaning the floor.

## ***CHAPTER-2***

### **INTRODUCTION**

#### **2.1 ROBOT**

Robot is an intelligent device having its own brain fed with computer logic so that it can do the work according to the algorithm designed. Autonomous movement of vehicle is guided by the logic controller designed. Robots play an important role in each every field of life. It is used in industries, in households and in institutes. The robots are just becoming as intelligent as human now a days. Mostly an average human uses 2-3 robots per day in his day to day life.

Various robotics parts are:-

- Pneumatic devices
- Actuators
- Sensors
- Mechanical control devices like valve
- Microcontroller – Controlling unit

Mechanical control devices are used to control the flow or movement of materials or any other parts present in the device. Actuators are used for controlling a mechanism which ultimately controls a part of the device.

Sensors are the sensing devices which transmit a signal and receive the signal and accordingly used to accumulate the various environment information which is ultimately fed to microcontroller for deciding the working of machines.

Microcontroller is the brain of robot where program is written and sensors are connected as input and actuators as output. The controlling of the robot is governed by various algorithms like fuzzy controller, machine learning based practices and artificial neural network based algorithms. Depending upon the environment value received to the controller it eliminates the error and transits from one state to another. Basically there are two types of controllers, one is continuous controller and another is PID based controller. Continuous controller is more direct and less effective while PID controller is more advanced and varies according to the current state and gives efficient result.

## **2.2 SENSORS**

Sensor plays a very important role in every type of robot. Various types of sensors are present. A sensor is a transducer whose intention is to sense (that is, to distinguish) some normal for its environs. It identifies occasions or changes in amounts and gives a relating yield, by and large as an electrical or optical sign; for instance, a thermocouple changes over temperature to a yield voltage. In any case, a mercury-in-glass thermometer is additionally a sensor; it changes over the deliberate temperature into development and construction of a fluid which can be perused on an adjusted glass tube.

Sensors are utilized as a part of ordinary questions, for example, touch-touchy lift catches (material sensor) and lights which diminish or light up by touching the base, other than endless utilizations of which the vast majority are never mindful. With advances in micro machinery and simple to-utilize microcontroller stages, the employments of sensors have extended past the more customary fields of temperature, weight or stream measurement, for instance into MARG sensors. Besides, simple sensors, for example, potentiometers and power detecting resistors are still generally utilized. Applications incorporate assembling and apparatus, planes and aviation, autos, solution and apply autonomy.

A sensor's affectability shows how much the sensor's yield changes when the data amount being measured changes. Sensor is mainly consists of transducers, operational amplifiers. Transducer converts any physical signal to electrical signal. Basically the electrical signal present in very small range. So we have to use amplifiers which provides very high gain and increases the signal to volt range. Among various types of sensor major sensors are ultrasonic distance measurement sensor, sound sensor, pressure sensor, touch sensor, temperature sensor, and field sensor.

## **2.3 HELP TO MANKIND**

Controlling provides various advantages over human powered work. Following are some of them:-

- It gives accurate results and eliminates possibility of manual error.

- It is very first and efficient and the control system used in industries are 100 times efficient than human work.
- In some part of the work areas it lessens the human efforts. Washing machine comes under this category.
- It also plays the great role in bringing entertainment in human life in different work. Television is the live example of these type of robots.

## **CHAPTER-3**

### **LITERATURE REVIEW**

Traditionally floor is cleaned with the help of dry mop or wet mop using the hand as a potential tool. They have to scrub hard on the surface. The cleaning includes cleaning of various surfaces basically cement floors, highly polished wooden or marble floors. Among these floors the rough surface floor such as cement floor, mostly present in semi urban areas are covered with so much dust.

From early human civilization human is increasingly dependent on the machines. Human is trying to reduce the workload upon himself. By the help of machines also we can get huge efficiency because there is no chance of human error there. Now -a -days from 30 years intelligence and robotics growing with a vast pace. Every human is using 2-3 robot at least per day. If we look at past 30 years we will see robotics from large structure going to small and smaller in Nano range. Very complicated sensors have been designed to help the robot in various works .Complicated pneumatic and actuating systems have been designed. One of the best examples is the mobile phone. If we look at the floor cleaning robot we can see iRobot is dominating the market with its 90 sq. cm robot having indoor navigation as its principal controlling system. Also for many in-house mobile robots indoor navigation is a big issue. Also currently indoor GPS is evolving which uses unsupervised learning and determining its path in its first run. Since now indoor navigation has not been solved completely. Also now-a-days complicated artificial intelligence algorithms like unsupervised and supervised learning, Swarm optimization, ANT algorithm, natural heuristic search are playing a major role in designing control system of the most of the mobile robot [10].

### **3.1 BODY**

The body of the robot has many small components. Like all robots it has sensors, microcontrollers and actuators and other components. It has 2 vacuum pumps connected in backside as well as front side of the robot. A 300 rpm DC motor is connected in the middle of the robot with the scrubber. A bearing is attached to the axle of the scrubber. 2 DC motors of 100 rpm are connected to the wheels. One microcontroller with 4 ultrasonic sensors is attached to it. This has 2 bread boards for circuit connection which ultimately can be replaced after welding.

For scrubbing we are using the brushes instead of cloths .The scrubber rotates at very high speed which performs very good mopping action.

### **3.2 NAVIGATION SYSTEM**

Navigation system of the robot is basically dependent on the sensors and microcontroller and algorithm fed to it. Basically the data acquisition system (here sensor) first collects the data from the environment and feeds to microcontroller .The microcontroller uses 2 algorithms. The 2 algorithms are:-

- Spiral motion algorithm
- Random Straight path following

#### **3.2.1 SPIRAL MOTION**

Basically after sensing the obstacle distance from outside environment, if the robot has sufficient space on its 4 sides it will move in spiral path at first half of its running. The spiral path can be anti-clockwise and clockwise .The spiral path can be generated by the decreasing ratio of left motor encoder and right motor encoder [1].

#### **3.2.2 RANDOM STRAIGHT PATH**

Basically random straight path searches from one node to another by the help of natural heuristic search. After the spiral motion the robot if detects a collision then it follows the edge of the wall until it gets enough free space for spiral motion again. After some moment if it doesn't get any specific clear area for spiral motion then it will move in random path for some time and the obstacle detection and avoidance system will be carried out by the help of ultrasonic sensors [2]. After that robots stop rotating if the timer is over. In this process we can divide a particular area in the floor as grids and move accordingly so that it will have very confine control over the robot. So it will have grid based search over the floor for movement [4].

Finally we implemented computer vision by the help of ultrasonic imaging and analyzing the image for the dust particles by the help of supervised learning and clustering the data [5]. We have implemented here A\* search algorithm for motion planning [7]. The breadth first search implemented here is very effective and provides efficient result for moving [9].

### **DESIGN OF THE BOT**

The mechanical parts include the systems that if they work in proper sequence the floor will be cleaned. This is achieved by 4 processes.

1. Dry vacuum cleaning
2. Sprinkle of water or cleansing liquid on the surface.
3. Scrubbing of surface.
4. Sucking of wet debris

#### **4.1. Dry vacuum pump**

This is the process to clean the dust particles from the surface so that the load will be lessen for the purpose of other operation. If we remove this part there will be unnecessary load on scrubbing and wet sucking. This process is achieved by using a 12v DC vacuum pump. The inlet is divided into a number of holes so that dust all over the width can be sucked.

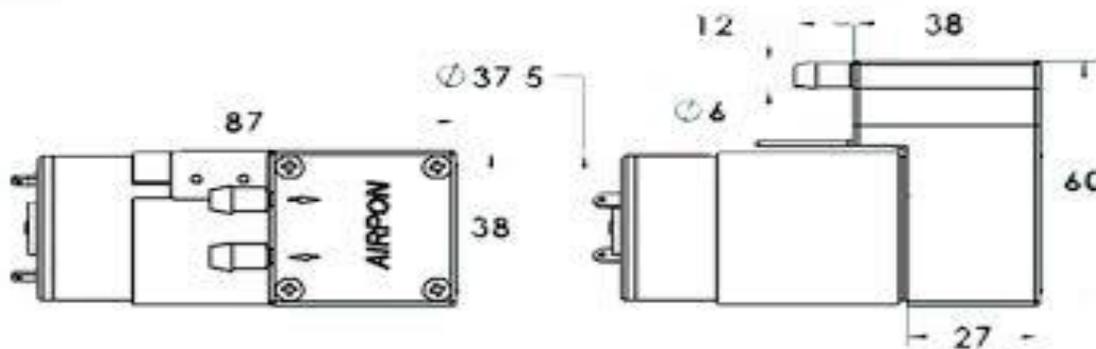


Figure 4.1 Vacuum Pump front view and side view

The outlet is connected to a chamber that collects dry debris for disposal on a later case.

#### **4.2. Sprinkle of water or cleansing liquid on the surface:**

The challenge here is to spray liquid across the width of the machine with proper amount. The amount should vary according to the rotation or movement of the machine. While turning the spray should be less. This is achieved by using a motor in the water chamber. This motor controls the amount of water to be sprayed. The flow all over the width is achieved by a sprinkler mechanism.

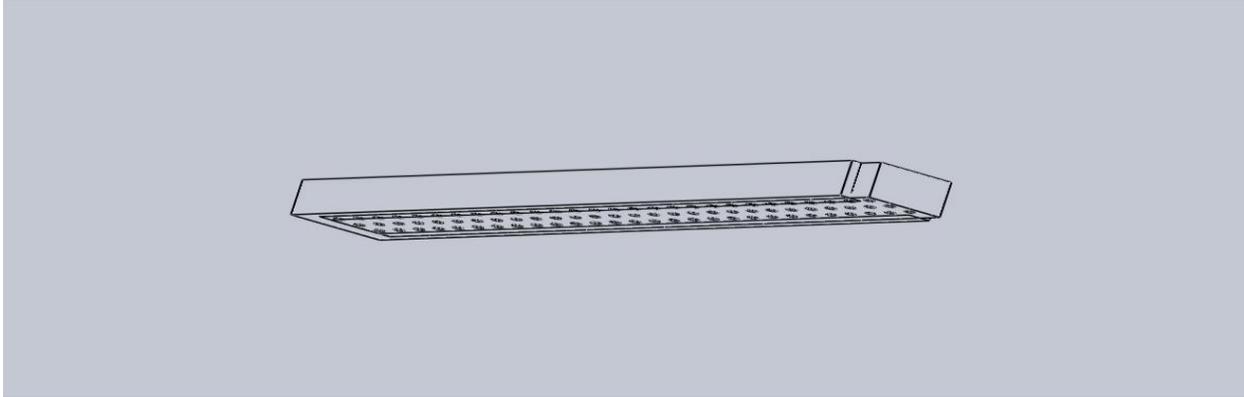


Figure 4.2 Sprinkler

### 4.3 Scrubber System

This is a mechanism to rub the floor repeatedly to clean all types of debris which may be oily or any sticky type.

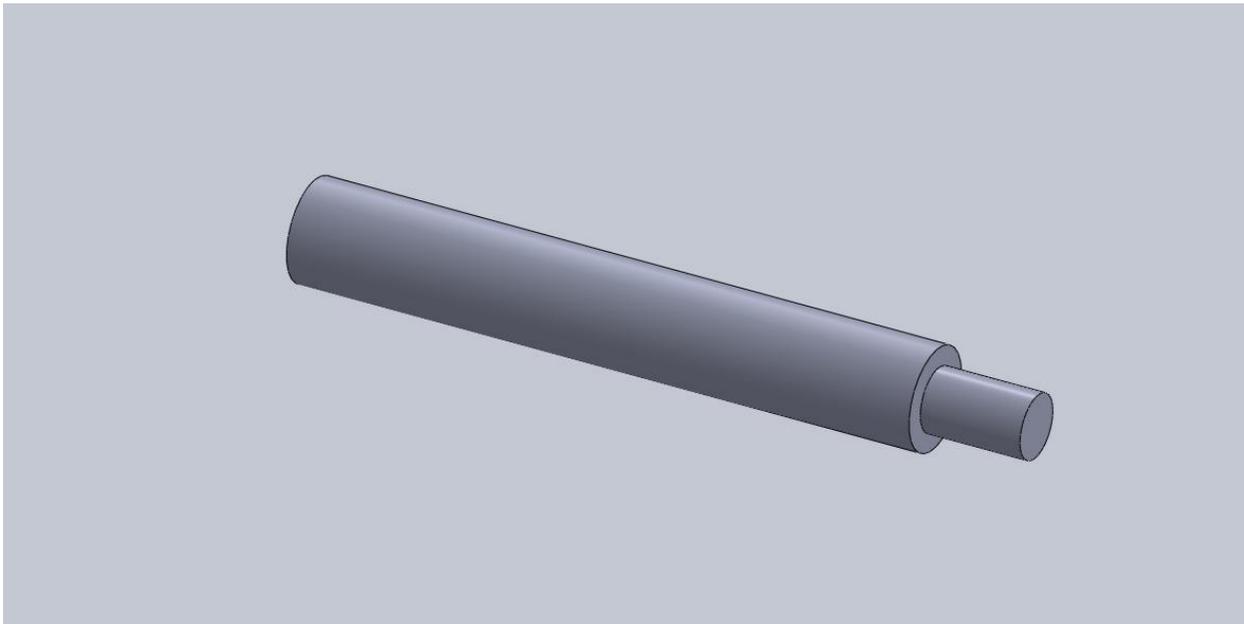


Figure 4.3 Scrubber system

In this case one side of the scrubber is fixed to a motor via a hub which gives it a rotational motion leading to cleaning of surface. The other side of the scrubber is fixed to a ball bearing which gives support.

### 4.4 Sucking of wet debris

This is the last step on this process. Here another vacuum pump is used to suck wet debris so that the floor will be dry. In this case also the vacuum is equally distributed over the width.

The power of the pump is 12v dc. The face of the pipe is towards the front to increase efficiency of suction.

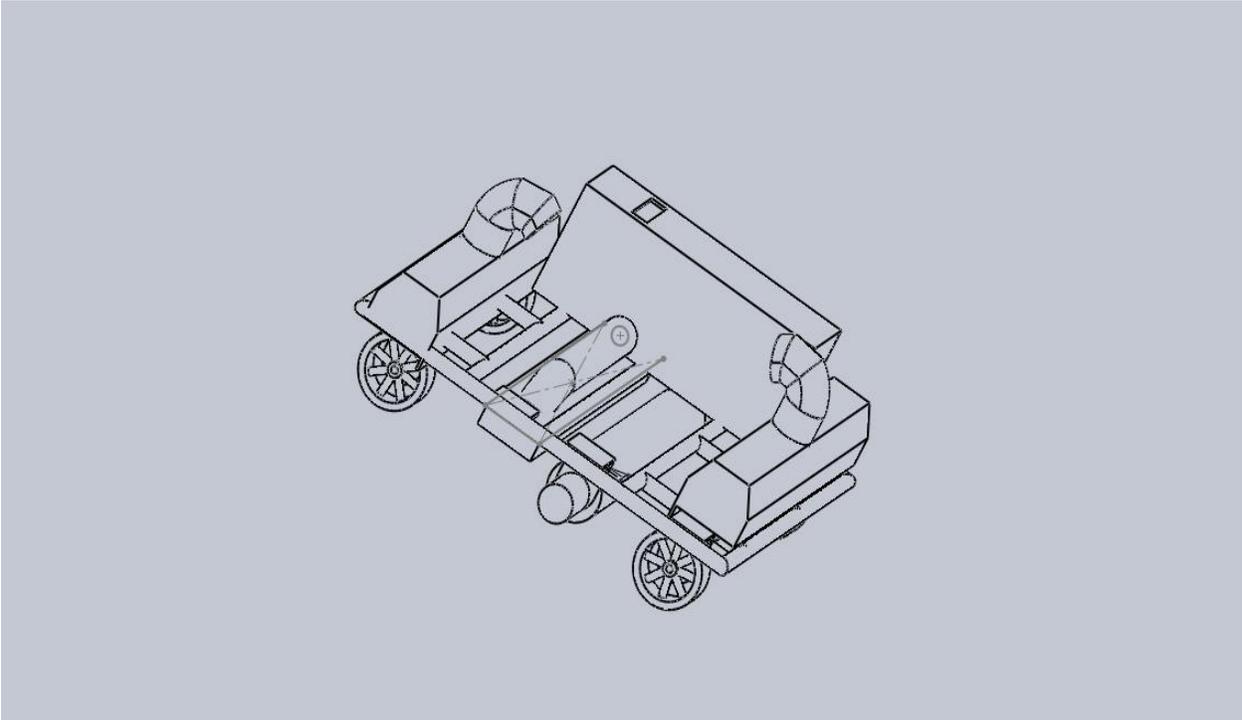


Figure 4.4 Assembly File Isometric view

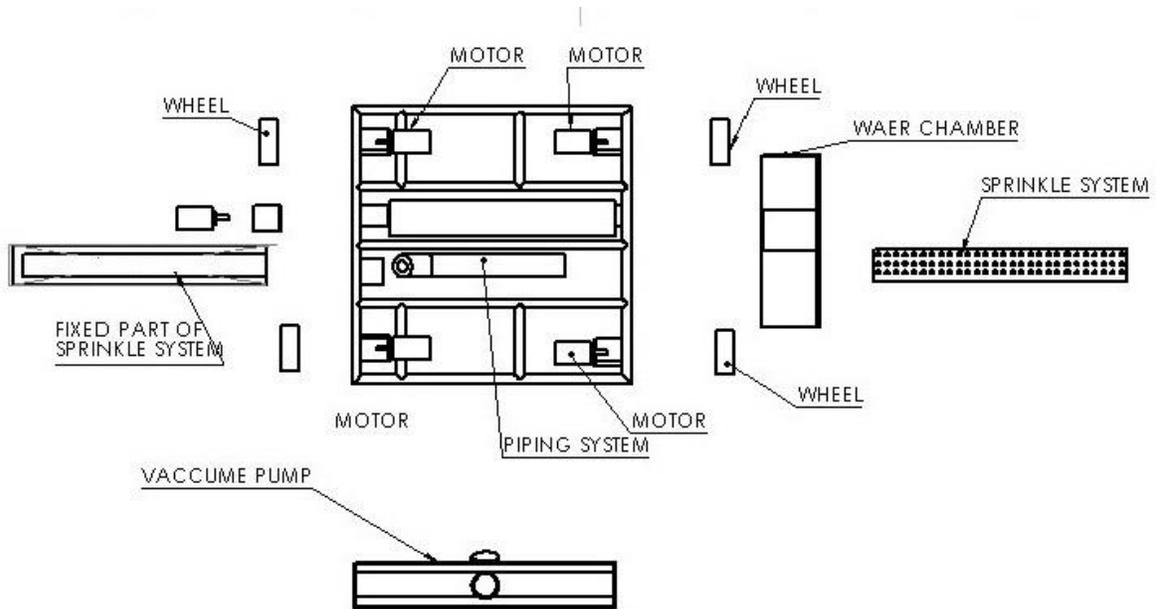


Figure 4.5 Assembly File top view

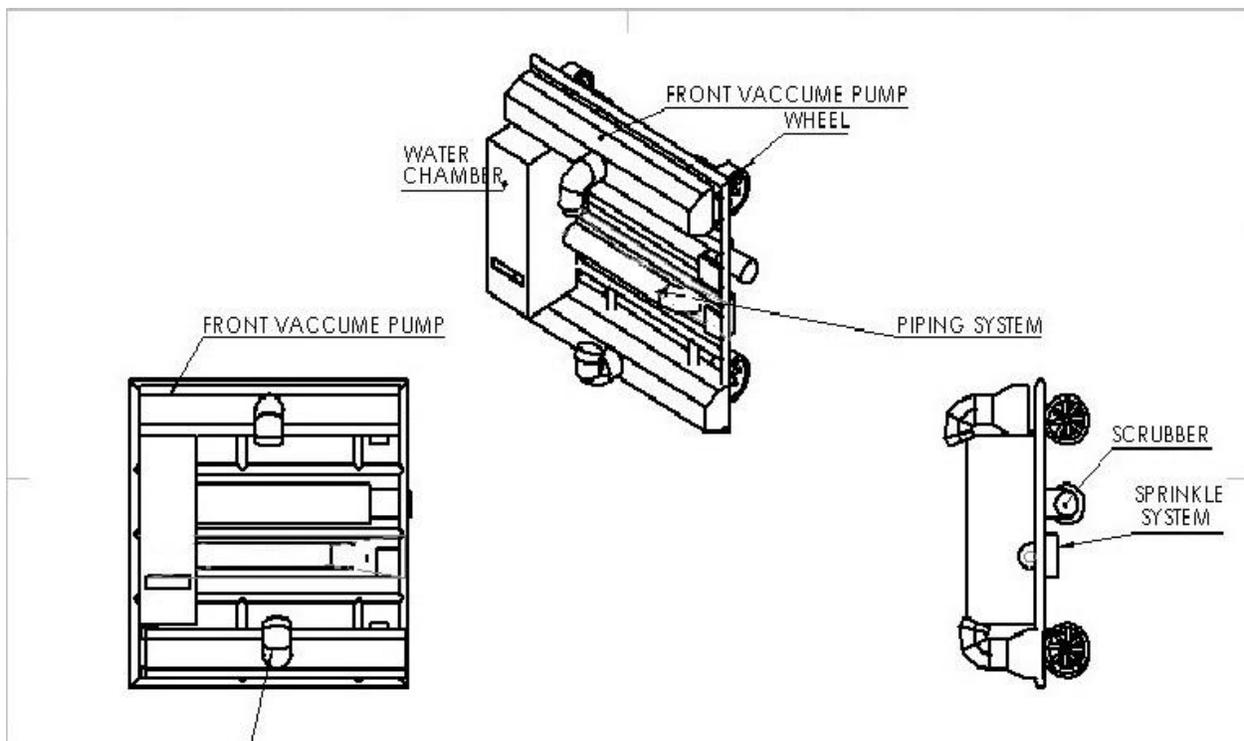


Figure 4.6 Assembly File Isometric view, top view, left side view

## CHAPTER-5

### AUTOMATION AND CONTROL OF MOBILE ROBOT

#### 5.1 AUTOMATION

We have to automate the robot so that it will roam freely on the floor avoiding all the obstacles. We have to also provide a microcontroller in which we have to feed the code so that it will work as a brain of the robot. Also we have to give a proper power source and proper motor for regulating the sprinkling of the robot and motor driver for controlling the direction and speed of motor connected to wheel.

##### 5.1.1 SENSORS



Figure 5.1 Ultrasonic sensor  
Company name – HCSR04

This is an ultrasonic sensor of 40 kHz frequency specification. It requires power supply of 5 volt with working current specification as 15 mA .It detects around 13ft of distance. It triggers the pulses in the interval of 10 us. It has 4 terminals namely  $V_{cc}$ ,trig,echo and GND pin .

$V_{cc} = 5v$

Trig pin =connected to Arduino board PWM pin

Echo pin = connected to Arduino

GND = negative terminal

Through trig pin we set the pulses and emit it to the environment and after emitting square waves the signals are received from the echo pin of this sensor module which ultimately fed to the Arduino board.

### Code for Arduino

```
const int trig_pin = 12;
const int echo_pin = 10;
void setup()
{
  Serial.begin(9600);
}
void loop()
{
  log time,cm;
  pinMode(trig_Pin,OUTPUT);
  digitalWrite(trig_Pin,LOW);
  delaymicroseconds(3);
  digitalWrite(trig_pin,HIGH);
  delayMicroseconds(10);
  digitalWrite(trig_Pin,LOW); // A full square wave is generated with T(on)= 3 and T(off)=10
  pinMode(echopin,OUTPUT);
  time = pulseIn(echo_Pin,OUTPUT);
  cm = microseconds_to_Centemeter(time);
  Serial.print(cm);
  Serial.print("cm");
  Serial.print();
  delay(1000);
}
long microseconds_to_centimeter(long micro_sec)
{
  return micro_sec/29/2;
}
```

Here above trig pin of sensor is connected to 12 of Arduino and echo pin is connected to 10 of Arduino.

Connected circuit for above:-

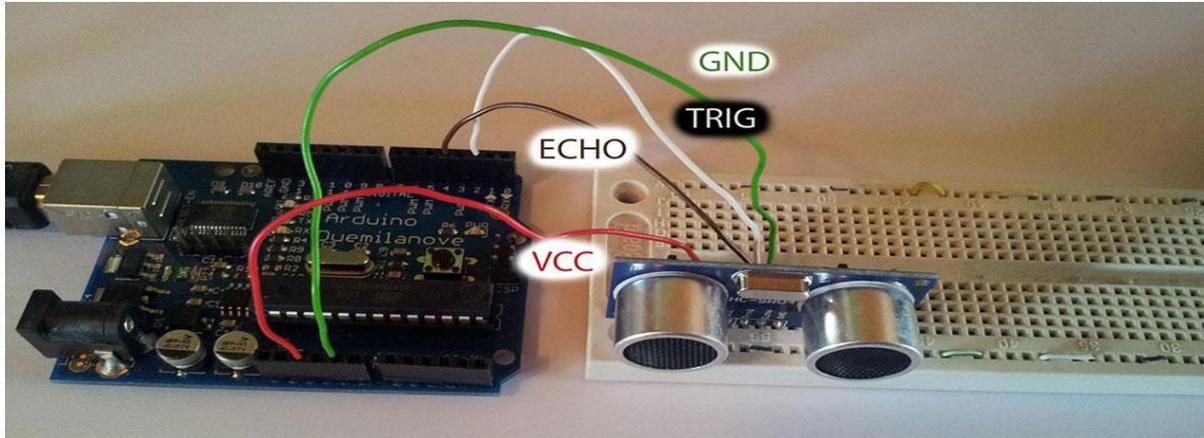


Figure 5.2 Ultrasonic sensor connected with Arduino

### 5.1.2 ARDUINO BOARD:-

For this experiment purpose we are using Arduino due board.

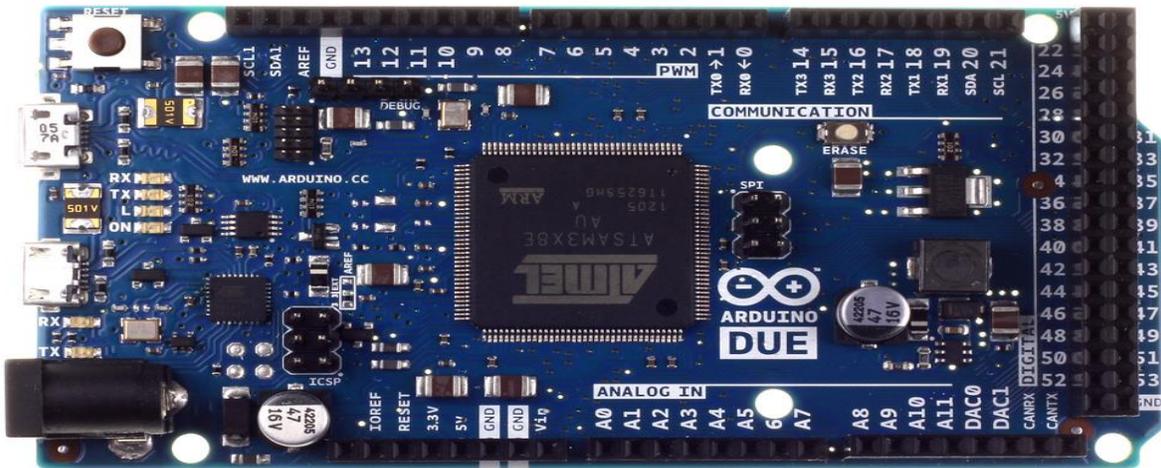


Figure 5.3 Arduino Due Microcontroller | Link – <http://www.arduino.cc>

Arduino due board requires at 3.3. Volt. If we give more than 3.3, i.e. 5 volt to i/o pins it will damage the board. It is 32-bit ARM Core. CPU clock rate of this board is 84 MHz. It has SRAM of 92Kb and flash memory of 512 Kb. The microcontroller version of this board is AT91SAM3X8E.

It has 54 digital I/O pins from which 12 has PWM output. It has 12 analog output pins from which 2 DAC present. We can give power to it either from the USB or through power jack from batteries.

Pin 2 – 13 are 8 bit PWM output. Pins 0 and 1 are connected to the corresponding pins of the ATmega16U2 USB-to-TTL Serial chip.

2 DAC pins provide analog outputs of 12 bit resolution through analogWrite() function. Reset button on the board is provided for resting the microcontroller board. There are 2 USB ports. one is native USB port and other one is programming USB port. We feed the program written in Arduino IDE to Arduino board through programming usb port through USB cable.

The program written in Arduino IDE consists of two functions basically .One is loop() and other one is setup().

### 5.1.3 MOTOR DRIVER:-

For this experiment purpose we used L293D motor driver. The schema of L293D is as following.

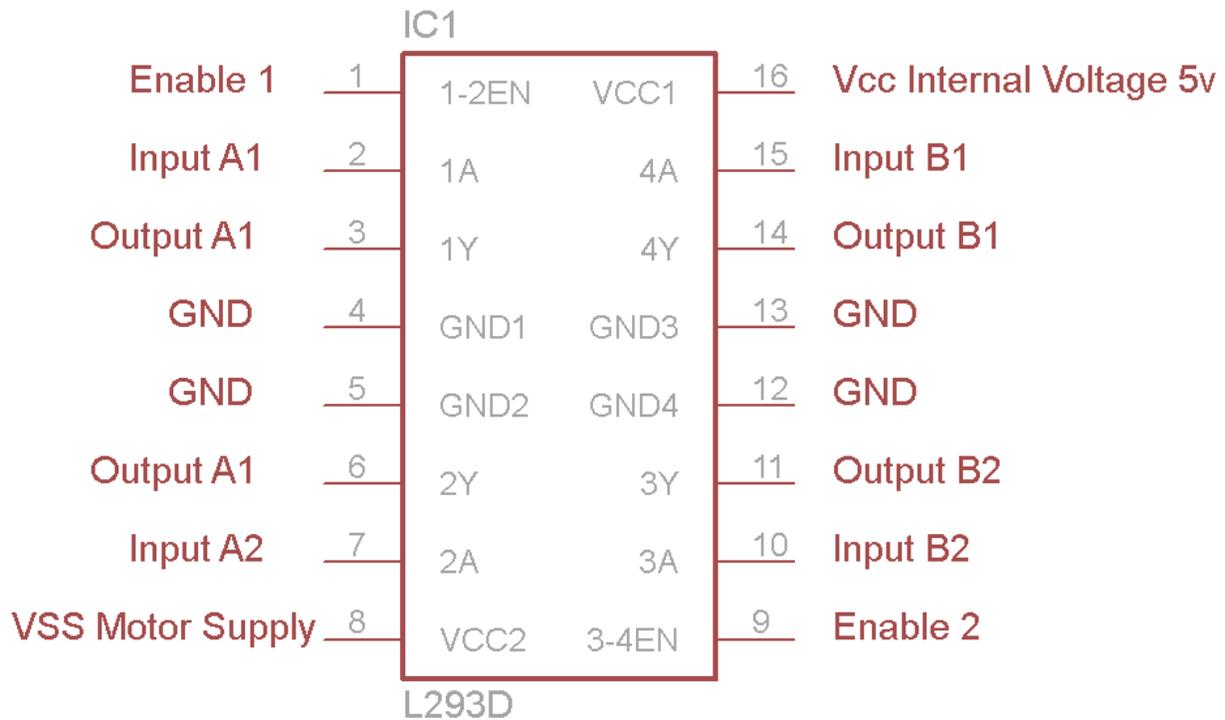


Figure 5.4 L293D pin configuration

It is 16 pin structure based IC. This L293D motor driver IC has 4 input pins and 4 output pins and 2 enable pins and motor power supply of 7 volt.

$V_{cc} = 5$  volt

$V_{ss} = 7$  volt

The positive terminal of motor is connected to pin 3 and –ve terminal to pin 6. Following is the criterion for motor rotation:-

- Pin 2 = Logic HIGH and Pin 7 = Logic LOW| Clockwise Direction

- Pin 2 = Logic LOW and Pin 7 = Logic HIGH | Anticlockwise Direction
- Pin 2 = Logic LOW and Pin 7 = Logic LOW | Idle [No rotation] [Hi-Impedance state]
- Pin 2 = Logic HIGH and Pin 7 = Logic LOW | Idle [No rotation]

By this IC we can control 2 dc motors .This can provide maximum current of 600mA.  $V_{cc}$  can be extended upto 36 volt and beyond that it will hamper the board.

### Connection of arduino with l293d and motor:-

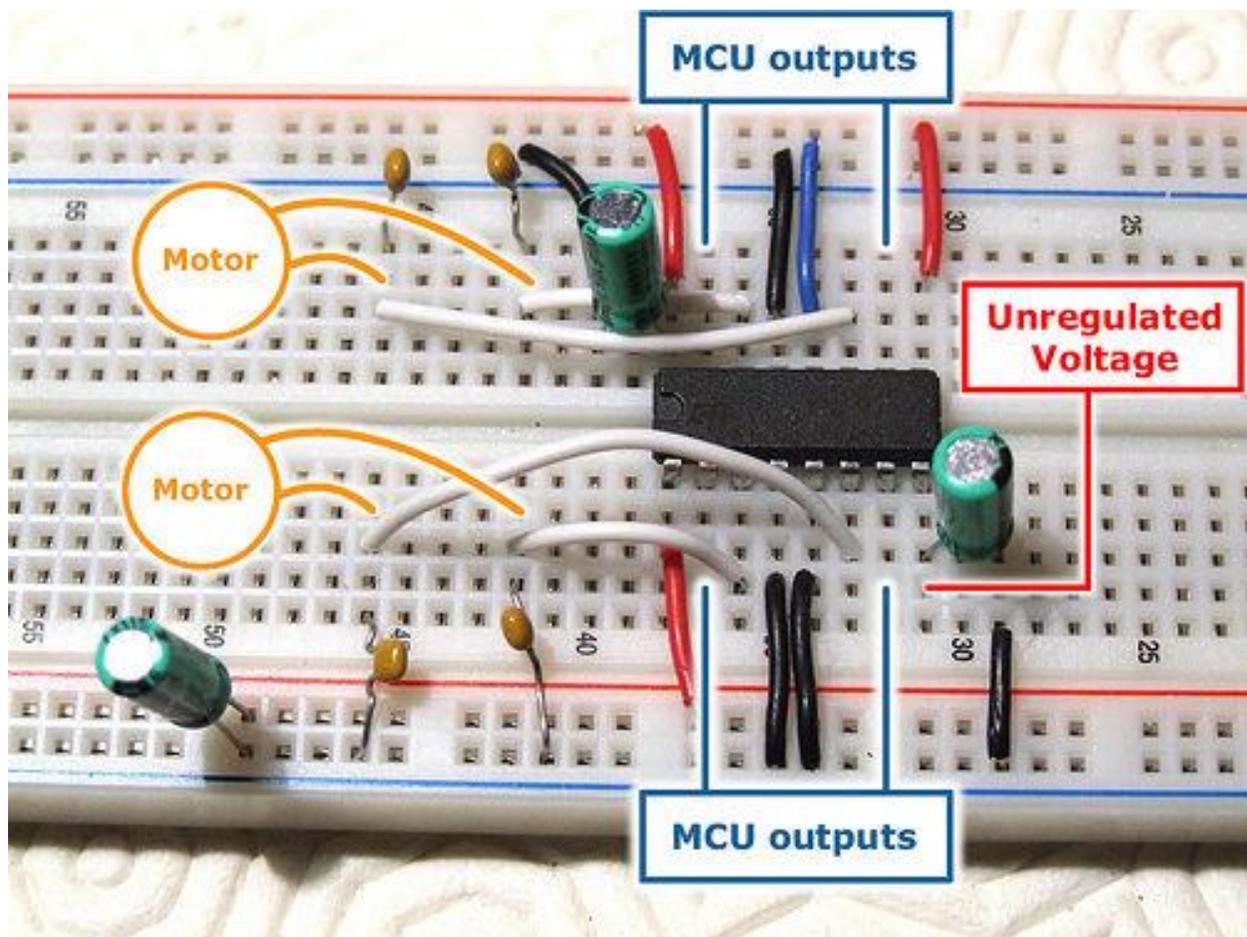


Figure 5.5 L293D Circuit connection

### 5.1.3.1 Arduino code for DC motor control

```
// ---- Motors' variable
int motorLeft[] = {2, 3};
int motorRight[] = {7, 8};

// ----Setup function
void setup() {
  Serial.begin(9600);

  // Setup motors and select the pins
  int i;
  for(i = 0; i < 2; i++){
    pinMode(motorLeft[i], OUTPUT);
    pinMode(motorRight[i], OUTPUT);
  }
}

// ---- Loop function
void loop() {

  driveForward();
  delay(1000);
  motorStop();
  Serial.println("1");

  driveBackward();
  delay(1000);
  motorStop();
  Serial.println("2");

  turnLeft();
  delay(1000);
  motorStop();
  Serial.println("3");

  turnRight();
  delay(1000);
  motorStop();
  Serial.println("4");

  motorStop();
  delay(1000);
  motorStop();
}
```

```
// -- Driving the bot backward and forward, left and right
```

```
void motorStop(){  
digitalWrite(motorLeft[0], LOW);  
digitalWrite(motorLeft[1], LOW);  
  
digitalWrite(motorRight[0], LOW);  
digitalWrite(motorRight[1], LOW);  
delay(25);  
}
```

```
void driveForward(){  
digitalWrite(motorLeft[0], HIGH);  
digitalWrite(motorLeft[1], LOW);  
  
digitalWrite(motorRight[0], HIGH);  
digitalWrite(motorRight[1], LOW);  
}
```

```
void driveBackward(){  
digitalWrite(motorLeft[0], LOW);  
digitalWrite(motorLeft[1], HIGH);  
  
digitalWrite(motorRight[0], LOW);  
digitalWrite(motorRight[1], HIGH);  
}
```

```
void turnLeft(){  
digitalWrite(motorLeft[0], LOW);  
digitalWrite(motorLeft[1], HIGH);  
  
digitalWrite(motorRight[0], HIGH);  
digitalWrite(motorRight[1], LOW);  
}
```

```
void turnRight(){  
digitalWrite(motorLeft[0], HIGH);  
digitalWrite(motorLeft[1], LOW);  
  
digitalWrite(motorRight[0], LOW);  
digitalWrite(motorRight[1], HIGH);  
}
```

## 5.1.4 ALGORITHM

In this bot we are using two algorithms. One is:

- Spiral path following algorithm
- Random Straight motion algorithm

### 5.1.4.1 Random Straight Path Flow chart

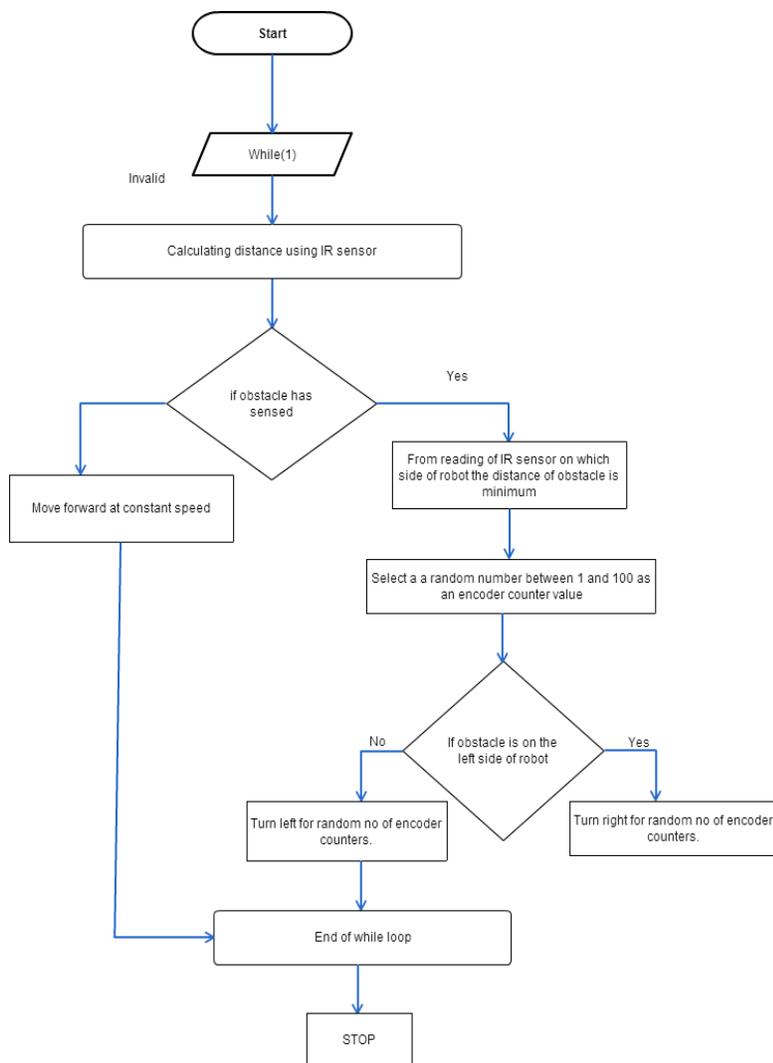


Figure 5.6 Random Straight Path Flow chart

### 5.1.4.2 Spiral Path following flow chart

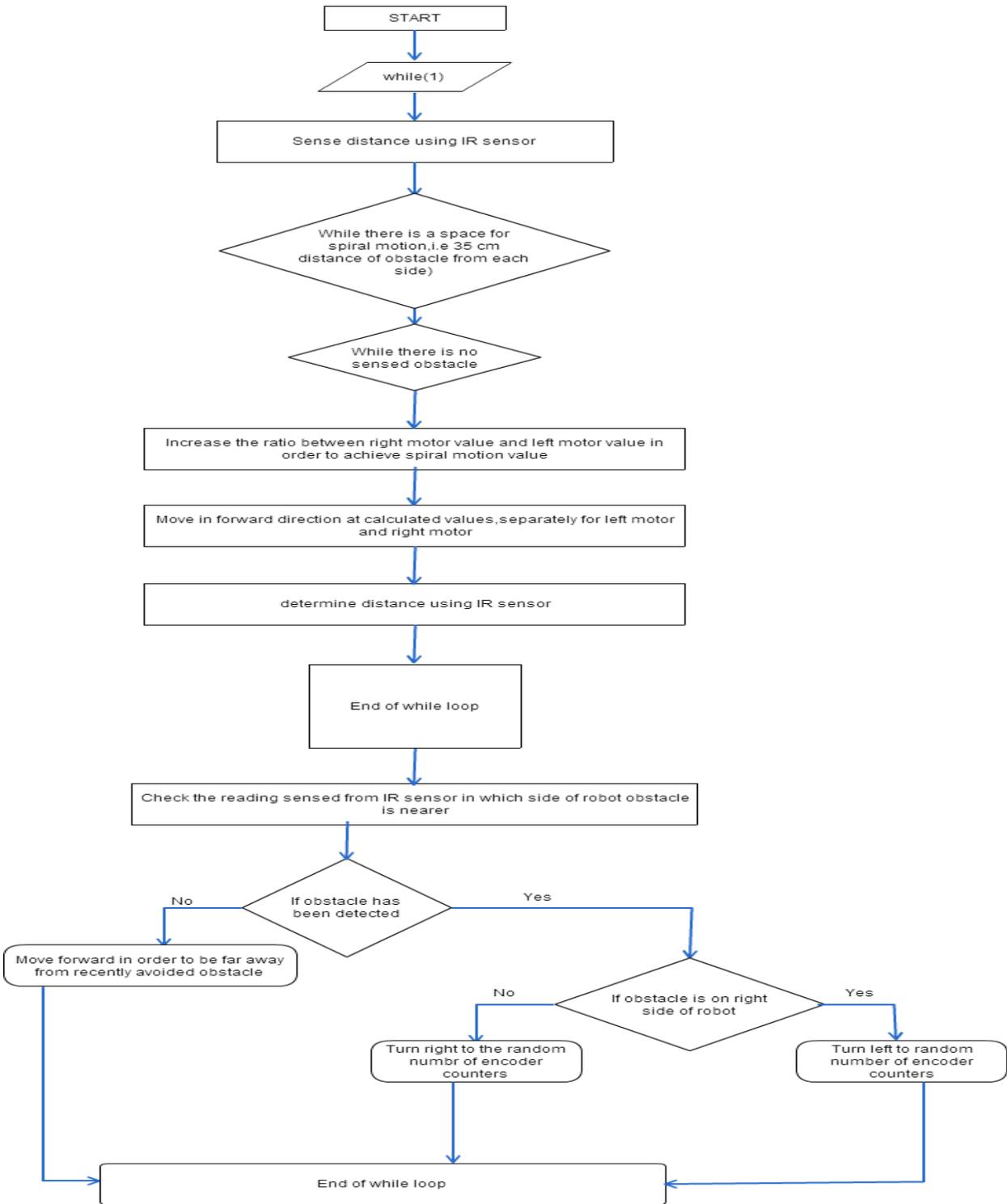


Figure 7 Spiral Path Flow chart

### **5.1.5 Final Program**

**(For automatic floor cleaner with full control)**

```
const int trigpin_left=13,trigpin_right=12,trigpin_front=11;
const int echopin_left=7,echopin_right=6,echopin_front=5;
const int dist = 18;
const int
LeftMotorBackward=22,LeftMotorForward=24,RightMotorBackward=26,RightMotorForward
=28;
const long duration_left = 0,duration_right = 0,duration_front = 0;
const int dist_right = 0,dist_left = 0,dist_front = 0;
void setup()
{
  Serial.begin();
  pinMode (trigpin_left, OUTPUT);
  pinMode (echopin_left, INPUT );
  pinMode (trigpin_right, OUTPUT);
  pinMode (echopin_right, INPUT );
  pinMode (trigpin_front, OUTPUT);
  pinMode (echopin_front, INPUT );
  pinMode (LeftMotorBackward, OUTPUT);
  pinMode (LeftMotorForward, OUTPUT);
  pinMode (RightMotorBackward, OUTPUT);
  pinMode (RightMotorBackward, OUTPUT);
}
void loop()
{
  /*For left sensor*/
```

```
digitalWrite(trigpin_left,LOW);
delayMicroseconds(3);

digitalWrite(trigpin_left,HIGH);
delayMicroseconds(10);

duration_left=pulseIn (echopin_left,HIGH);
dist_left =microsecondsToCentimeter(duration_left);

/*For right sensor*/
digitalWrite(trigpin_right,LOW);
delayMicroseconds(3);

digitalWrite(trigpin_right,HIGH);
delayMicroseconds(10);

duration_right=pulseIn (echopin_right,HIGH);
dist_right =microsecondsToCentimeter(duration_right);

/*For front sensor*/
digitalWrite(trigpin_front,LOW);
delayMicroseconds(3);

digitalWrite(trigpin_front,HIGH);
delayMicroseconds(10);

duration_front=pulseIn (echopin_front,HIGH);
dist_front = microsecondsToCentimeter(duration_front);
```

```

if((dist_front > dist) && (dist_left > dist) && (dist_right > dist))
{
    spiral_clock();
}
if(abs(dist_right - dist_left) < 5)
{
    move_Backward();           //Go to the moveBackward function
    delay(200);                //Pause the program for 200 milliseconds to let the robot
reverse
    move_Right();              //Go to the moveRight function
    delay(100);                //Pause the program for 200 milliseconds to let the robot
turn right
}
else if(dist_right < dist_left) //If the distance on the right is less than that on the
left then...
{
    move_Left();               //Go to the moveLeft function
    delay(100);                //Pause the program for half a second to let the robot
turn
}
else if(dist_left < dist_right) //Else if the distance on the left is less than that on the
right then...
{
    move_Right();              //Go to the moveRight function
    delay(100);                //Pause the program for half a second to let the robot
turn
}
}

```

```

/*LOGIC STARTAS HERE*/
void move_Forward()           //This function says the robot to go forward
{
  Serial.println("");
  Serial.println("Moving forward");
  digitalWrite(LeftMotorBackward, LOW);
  digitalWrite(LeftMotorForward, HIGH);
  digitalWrite(RightMotorBackward, LOW);
  digitalWrite(RightMotorForward, HIGH);
}

void move_Backward()         //This function says the robot to move backward
{
  Serial.println("");
  Serial.println("Moving backward");
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorForward, LOW);
  digitalWrite(RightMotorBackward, HIGH);
}

void move_Left()             //This function indicates the robot to turn left
{
  Serial.println("");
  Serial.println("Moving left");
  digitalWrite(LeftMotorForward, LOW);
  digitalWrite(LeftMotorBackward, HIGH);
  digitalWrite(RightMotorBackward, LOW);
  digitalWrite(RightMotorForward, HIGH);
}

```

```
}
```

```
void move_Right()           //This function indicates the robot to turn right
```

```
{
```

```
  Serial.println("");
```

```
  Serial.println("Moving right");
```

```
  digitalWrite(LeftMotorBackward, LOW);
```

```
  digitalWrite(LeftMotorForward, HIGH);
```

```
  digitalWrite(RightMotorForward, LOW);
```

```
  digitalWrite(RightMotorBackward, HIGH);
```

```
}
```

```
void move_Stop()           //This function indicates the robot to stop moving
```

```
{
```

```
  Serial.println("");
```

```
  Serial.println("Stopping");
```

```
  digitalWrite(LeftMotorBackward, LOW);
```

```
  digitalWrite(LeftMotorForward, LOW);
```

```
  digitalWrite(RightMotorForward, LOW);
```

```
  digitalWrite(RightMotorBackward, LOW);
```

```
}
```

```
void spiral_clock()
```

```
{
```

```
  Serial.println("");
```

```
  Serial.println("Moving clockwise");
```

```
  digitalWrite(LeftMotorBackward, 0); //Giving value to motor driver and decreasing the  
  ration for turning clockwise */
```

```

digitalWrite(LeftMotorForward, 110);
digitalWrite(RightMotorForward, 0);
digitalWrite(RightMotorBackward, 150);

    digitalWrite(LeftMotorBackward,0);
digitalWrite(LeftMotorForward, 150);
digitalWrite(RightMotorForward, 0);
digitalWrite(RightMotorBackward, 180);

digitalWrite(LeftMotorBackward, 0);
digitalWrite(LeftMotorForward, 180);
digitalWrite(RightMotorForward, 0);
digitalWrite(RightMotorBackward, 200);
}
/*Moving anticlockwise spiral motion*/
void spiral_antii()
{
    Serial.println("");
    Serial.println("Moving anticlockwise");
    digitalWrite(LeftMotorForward, 0);
    digitalWrite(LeftMotorBackward, 150); // Giving value to motor driver and decreasing the
    digitalWrite(RightMotorBackward, 0);      ratio for turning anticlockwise
    digitalWrite(RightMotorForward, 110);

    digitalWrite(LeftMotorForward, 0);
    digitalWrite(LeftMotorBackward, 180);
    digitalWrite(RightMotorBackward, LOW);
    digitalWrite(RightMotorForward, 150);

```

```
digitalWrite(LeftMotorForward, LOW);  
digitalWrite(LeftMotorBackward, 200);  
digitalWrite(RightMotorBackward, LOW);  
digitalWrite(RightMotorForward, 180);  
}
```

```
/* duration to centimeter conversion*/  
long microsecondsToCentimeter(long microseconds)  
{  
  return microseconds/29/2;  
}
```

# Chapter-6

## Construction of ultrasonic sensor from transducer pairs

Ultrasonic range finder is a circuit for measuring distance by the help of ultrasonic sound. First the ultrasonic burst is transmitted from the transmitter and then receiver receives the ultrasonic burst. The ultrasonic sound velocity is known in the medium as

$$c = 331,3 \text{ (m/s)} * (1+T/273)^{1/2}$$

Actually the speed of sonic wave varies with respect to temperature of medium. Overall attenuation in air depends upon: geometric spreading, molecular relaxation, boundaries, and refraction by non-homogeneous atmosphere, and diffraction by turbulence, conduction and shear viscosity losses. Attenuation also varies with respect to distance travelled by the sonic wave. In this experiment we are using two transducers of 40 kHz. Transducer is a device which converts one form of energy to electrical energy and electrical energy to another form of energy. It carries a piezoelectric material which does these energy conversion.

TRANSDUCER PAIR:-

Between the two transducer pair one is transmitter and other is receiver.

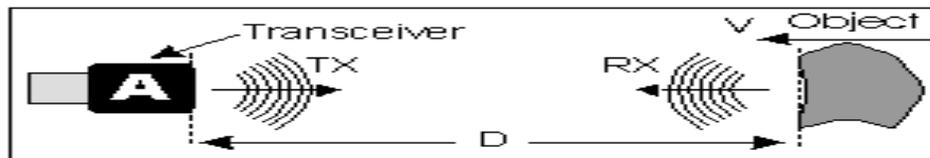


Figure 6.1 Transducer receiver and transmitter

$$D_i = 0.5 * C * ( T_{\text{initial}} - T_{\text{final}} )$$

(Where:

$D_i$  = Distance to Object

$C$  = Speed of Sound

$T_{\text{initial}}$  = Time at which sonic wave is transmitted

$T_{\text{final}}$  = Time at which sonic wave is received)

The main characteristic from which we have to choose ultrasonic transducer from are:-

- Resonant frequency
- Radiation pattern

- Sensitivity

The more the resonant frequency the lesser will be the wavelength of transmitted radiation and it will provide good surrounding condition. The more directional the sonic wave the more resolution in the measurement come. Sensitivity helps in decreasing signal to noise ratio.

Here we used two transducers of 40khz.as shown below:



Figure 6.2 Receiver and transmitter transducer pairs  
HCSR - 04

### 6.1 Transmitter circuit

Here the transmitter is connected in the bridged output mode. It is getting power from atmega 16 from Arduino board. In order to get the maximum output from power we are using bridged output. The transistor we are using as BC557.

Here  $V_{cc} = +5$  volt

The transmitter and receiver transducer is placed on the board on an inch distance. We can also use decoupling capacitors for reducing the noise from signal. The other terminal of end h bridge is connected to Arduino digital pin 10. Now we have to generate square wave from the atmega 16 powered in Arduino. We have to control the duty cycle of the square wave.

$$duty\ cycle = T_{on} / (T_{on} + T_{off})$$

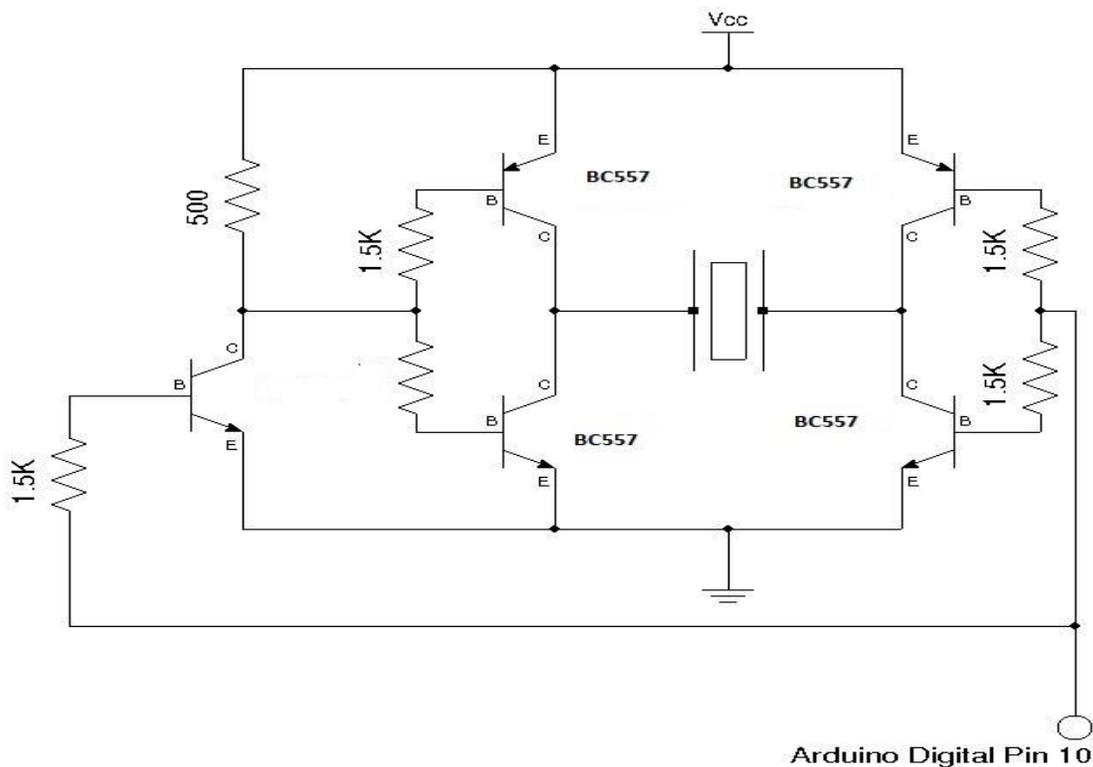


Figure 6.3 Ultrasonic Transmitter circuit

### Code to generate square wave in Arduino due:

```

void start_Transducer(float freq, float duty_Cycle)
{
  if (duty_Cycle > 0.5) duty_Cycle = 0.5;
  else if (duty_Cycle < 0) duty_Cycle = 0;

  cli();
  TCCR1B = _BV(WGM13) | _BV(CS10) | _BV(ICNC1);
  //f0 = fclk / (2 * N * Top)
  long topv = (long) ((float) F_CPU / (freq * 2.0 * 1.0));
  ICR1 = topv;

  OCR1A = (int) ((float) topv * duty_Cycle);
  OCR1B = (int) ((float) topv * (1 - duty_Cycle));
  DDRB |= _BV(PORTB1) | _BV(PORTB2);
  TCCR1A = _BV(COM1A1) | _BV(COM1B1);
  sei();
}

void stop_Transducer()
{
  cli();
  TCCR1B = 0;
  sei();
  digitalWrite(9, LOW);
  digitalWrite(10, LOW);
}

```

The range of transducer will be around 20 – 30 feet.

## 6.2 RECIEVER CIRCUIT

Performance of the transmitter is largely influenced by the sensitivity of the receiving signal. Generally the signal reaching on the receiver is very weak (in some millivolt range). So we have to amplify the signal and reduce the noise for optimal performance.

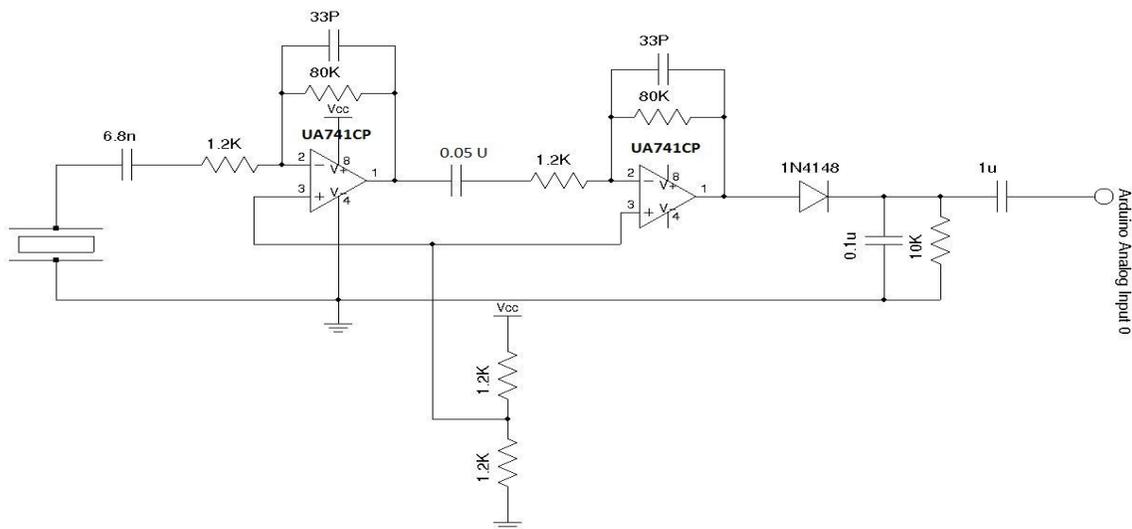


Figure 6.4 Receiver Circuit of ultrasonic sensor

IN4148 – Diode

UA741CP – opamp model

Gain in first stage of amplifier (approx.) =  $-20 \log (R_2/R_1) = -20 \log (80/1.2) = -36.92\text{db}$

(Here  $R_2$  value taken as 80 k $\Omega$  approximately whereas real value of  $R_2$  is less as 80K $\Omega$  is connected to capacitor of value 33pF. So gain will be less. But it will provide frequency stability.)

So approximate gain from two stage =  $36.92 + 36.92 = 72 \text{ dB}$

The 0.1 coulomb capacitor passes the signal having medium and high frequency signals.

Here 2 inverting opamps are used and combined stage has a voltage gain of 67 db. So after 2 successive inversion the phase of signal doesn't change. Here band pass filters are used with RC circuit where we pass high frequency signal into the circuit input. Band pass filter is centered on the frequency 40 kHz. The output voltage of the opamp is around 2.5 volt ( $V_{cc}/2$ ).

For easier processing of the echo, a diode (IN4148), capacitor of 0.05uF and resistor of 10k ohm are used to demodulate the signal. A coupling capacitor (1uF) is used. This helps to get rid of demodulated signal of the DC component.

### **6.3 RANGE CALCULATION**

For finding the range we calculated the time elapsed between the transmitted signal and received signal for range calculation. Here in the below code we specified the initial time as t\_start and the final time as t\_peak.

#### **CODE to calculate range in Arduino due:-**

```
byte a = 0;
unsigned long t_start = 0;
unsigned long t_peak = 0;
unsigned long t = 0;
byte v_peak = 0;
const float SPEED_OF_SOUND_20C = 0.0003432; //meters per micro-second
float d = 0;

void loop()
{
  startTransducer(24000.0, 0.5);
  delayMicroseconds(300);
  stopTransducer();

  v_peak = 0;
  t_start = micros();
  t_peak = t_start;
  delayMilliseconds(1);

  for (int i = 0; i < 256; i++) {
    a = analogRead(0);
    t = micros();

    if (a > v_peak) {
      t_peak = t;
      v_peak = a;
    }
  }

  t = t_peak - t_start;
  d = (float) t * SPEED_OF_SOUND_20C / 2.0;
  Serial.println(d , 2);
}
```

Complete circuit is done combining above two circuits. The duty cycle is set to 0.5 in both the circuit we can increase the duty cycle little bit to get the efficient result.

## **COMPLETE CODE**

**(For making ultrasonic transducer and receiver pair)**

```
void delay_Milliseconds(int ms) {
  for (int k = 0; k < ms; k++) {
    delayMicroseconds(1100);
  }
}

void stop_Transducer()
{
  cli();
  TCCR1B = 0;
  sei();
  digitalWrite(9,LOW);
  digitalWrite(10,LOW);
}

void start_Transducer(float freq, float duty_Cycle)
{
  if (duty_Cycle > 0.5) duty_Cycle = 0.5;
  else if (duty_Cycle < 0) duty_Cycle = 0;

  cli();
  TCCR1B = _BV(WGM13) | _BV(CS10) | _BV(ICNC1);
  //f0 = fclk / (2 * N * Top)
  long topv = (long) ((float) F_CPU / (freq * 2.0 * 1.0));
  ICR1 = topv;

  OCR1A = (int) ((float) topv * duty_Cycle);
  OCR1B = (int) ((float) topv * (1 - duty_Cycle));
  DDRB |= _BV(PORTB1) | _BV(PORTB2);
  TCCR1A = _BV(COM1A1) | _BV(COM1B1);
  sei();
}

void setup()
```

```

{
  Serial.begin(9600);

  pinMode(9, OUTPUT);
  pinMode(10, OUTPUT);
}

byte a = 0;
unsigned long t_start = 0;
unsigned long t_peak = 0;
unsigned long t = 0;
byte v_peak = 0;
const float SPEED_OF_SOUND_20C = 0.0003432; //per micro-second
float d = 0;

void loop()
{
  start_Transducer(24000.0, 0.5);
  delayMicroseconds(300);
  stop_Transducer();

  v_peak = 0;
  t_start = micros();
  t_peak = t_start;
  delay_Milliseconds(1);

  for (int i = 0; i < 256; i++) {
    a = analogRead(0);
    t = micros();

    if (a > v_peak) {
      t_peak = t;
      v_peak = a;
    }
  }

  t = t_peak - t_start;
  d = (float) t * SPEED_OF_SOUND_20C / 2.0;
  Serial.println(d, 2);
}

```

## Chapter-7

### **7. Commercialization possibility**

Now in the automatic floor cleaner market iRobot and Scooba are playing major roles. They hold around 80% of the market. Their costs are around 25000 to 35000. Also the algorithms used by them are not most effective. They are using algorithms which approximately provides 70% accuracy. They are not using any image processing algorithms to run their robot. But the robot designed by us is cost efficient which will cost around 15000. Also we can use camera lens for small dust particle detection, so that it will give more efficient decision in governing the motion of the particle which ultimately save considerable amount of power and reduce the timing with better efficiency and sensitivity. This will act like a pheromone like in ant algorithm.. In ant algorithm when pheromone density of ants in particular direction is denser all other ants follow that direction. Similarly when the robot will find the particular dust size on floor on one side of it and there are less on other 3 sides, it will head towards dusty area if obstacle is not present.

Time redundancy and power saving with low cost provides the best opportunity for marketing this consumer product.

## Chapter- 8

### **8. CONCLUSION**

The Product developed is definitely a very important product in robotics and floor cleaning area .The robots developed uses 2 vacuum pump which ultimately provides lots of vibration and power loss in the system. Also the algorithm implemented is not very effective. So there is definitely current scope for improvement and optimization till the most effective product is being developed. After optimizing the algorithm and taking it to the heuristic based search like bee algorithm it will be a great product and can revolutionize this industry. Definitely it has very huge potential. Also we can use 1 vacuum pump instead two so that it will be cost effective and very energy saving product with less vibration and much control over the robot. The robot having 33\*30\*8 cm in dimension is very compact in nature and can go beneath any furniture and bed. This is also very handy in portability. The scrubber of the robot now consists of small plastic fibers .But it can be further improved so that the surface area of the scrubber will come 90% in contact with the floor.

## **REFERENCE**

- [1] **Razvan Solea, Adrian Filipescu and Grigore Stamatescu** "Sliding-mode real-time mobile platform control in the presence of uncertainties ", *Decision and Control*(2009) 32 16-18
- [2] **T. Palleja, M. Tresanchez, M. Teixido, J. Palacin** " Modeling floor-cleaning coverage performances of some domestic mobile robots in a reduced scenario", *Robotics and Autonomous Systems*(2010) 58 37-45.
- [3] **M.R.B. Bahara, A.R. Ghiasib, H.B. Bahara**, "Grid roadmap based ANN corridor search for collision free, path planning ", *Scientia Iranica* (2012) 19 1850-1855.
- [4] **Ayoub Bahmanikashkoolia , Majid Zareb, Bahman Safarpourc, Mostafa Safarpourd** " Application of Particle Swarm Optimization Algorithm for Computing Critical Depth of Horseshoe Cross Section Tunnel " *APCBEE Procedia*( 2014)9 207–211
- [5] **Spyros G. Tzafestas**"9 – Mobile Robot Control V: Vision-Based Methods", *Introduction to Mobile Robot Control*(2014) 319–384
- [6] **Spyros G. Tzafestas**"11 – Mobile Robot Path, Motion, and Task Planning", *Introduction to Mobile Robot Control* (2014) 429–478
- [7] **Masoud Nosrati , Ronak Karimi , Hojat Allah Hasanvand** "Investigation of the \* (Star) Search Algorithms: Characteristics, Methods and Approaches" *Applied Programming*(2012) 2 251-256
- [8] **Dr. R.Anbuselvi** "PATH FINDING SOLUTIONS FOR GRID BASED GRAPH" *Advanced Computing*(2013) 4
- [9] **Rina Dechter, Judea Pearl** "Generalized best-first search strategies and the optimality of A\*", *Journal of the association of Computing Machinery*(1985) 32 505-536
- [10] **Ashraf A. Kassim, , B.V.K. Vijaya Kumar**"Path planners based on the wave expansion neural network", *Robotics and Autonomous Systems*(1999) 26 1–22