NAME : SAURABH SALVI

ROLL NO : ME22B1033

BRANCH : MECHANICAL

TASK #1
ELECTRONICS

Project 1 :

Controlling the turn of the car when car comes across an obstacle.

Short Description :

In this project, we have mainly used :

- Ultrasonic Sensor
- Potentiometer

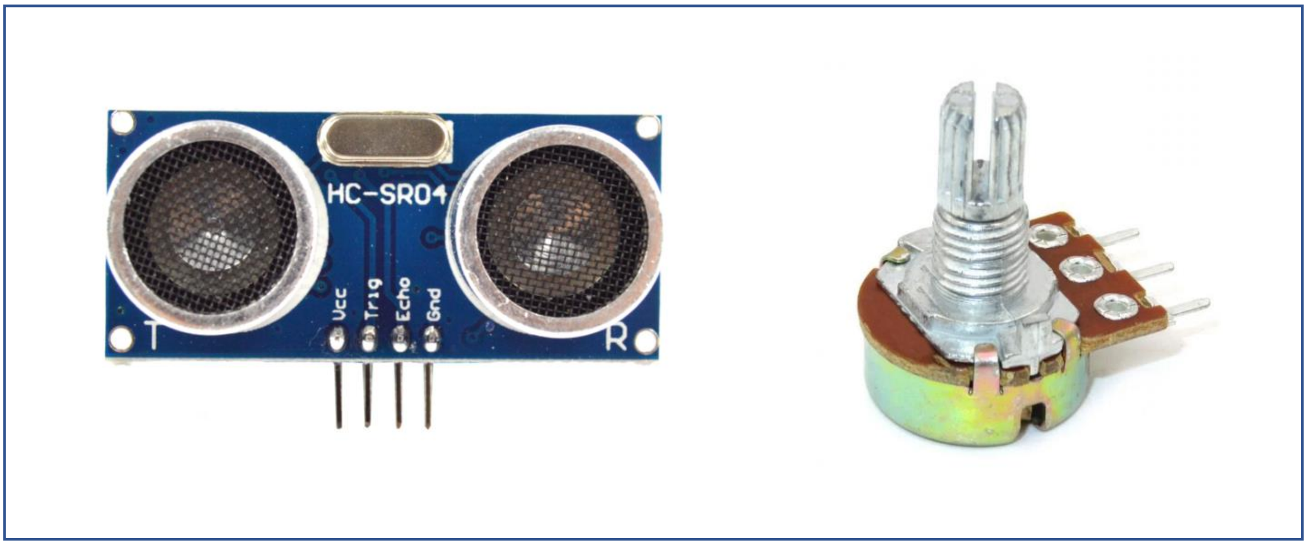
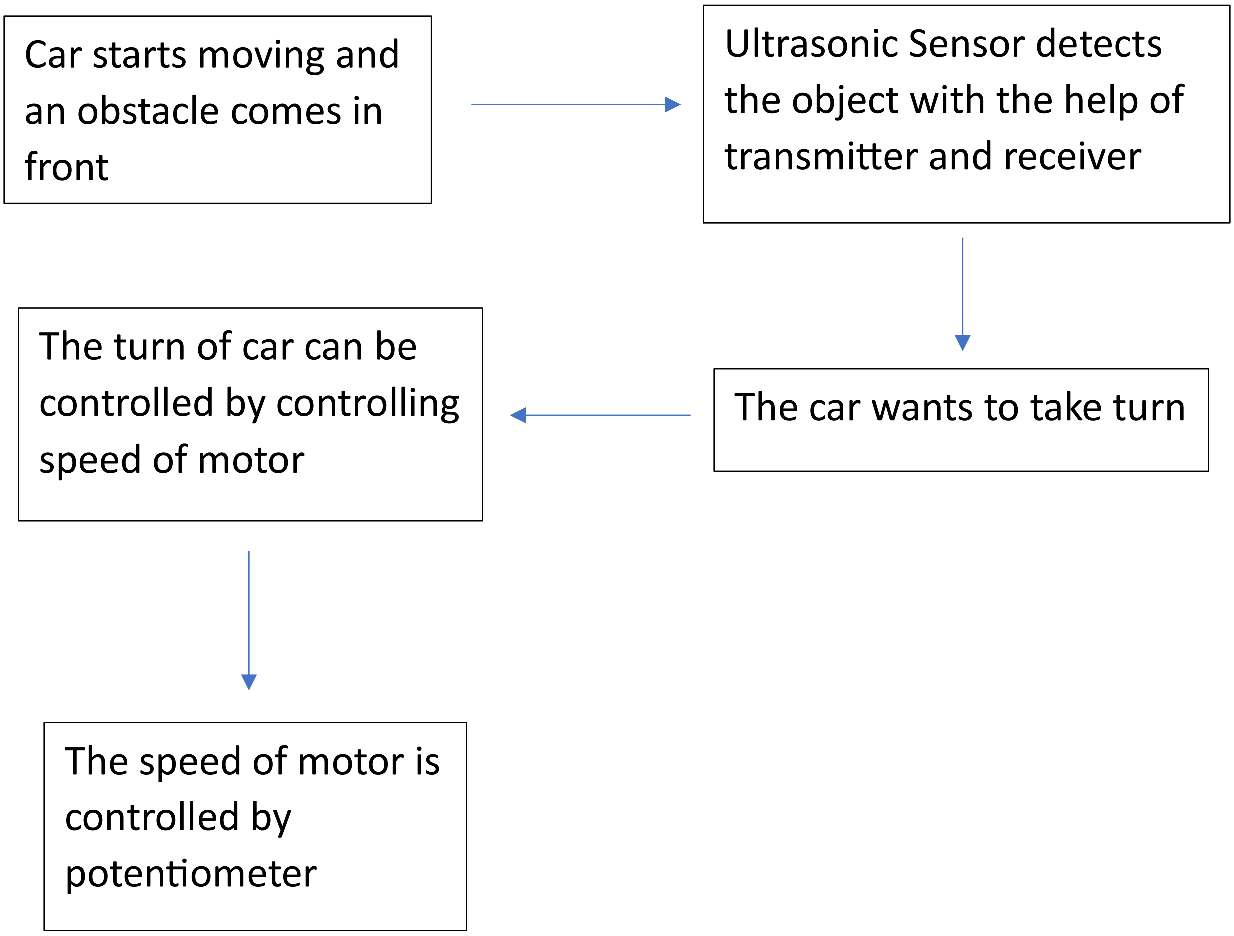


Fig. Components used in project L : Ultrasonic Sensor ; R : Potentiometer

Now, the car is given an obstacle avoiding ability by interfacing an ultrasonic sensor with an Arduino. Full process is shown by flow chart below.

Flow chart :



Car starts moving and an obstacle comes in front

Ultrasonic Sensor detects the object with the help of transmitter and receiver

The car wants to take turn

The turn of car can be controlled by controlling speed of motor

The speed of motor is controlled by potentiometer

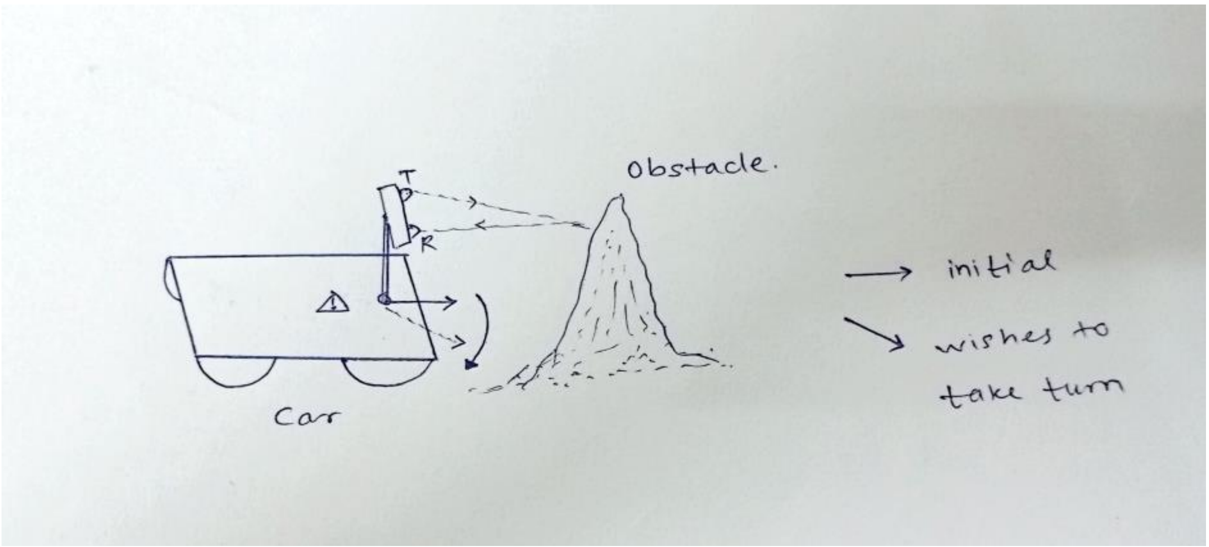


Fig. How things will work out!

The circuit schematic (using Tinkercad) :

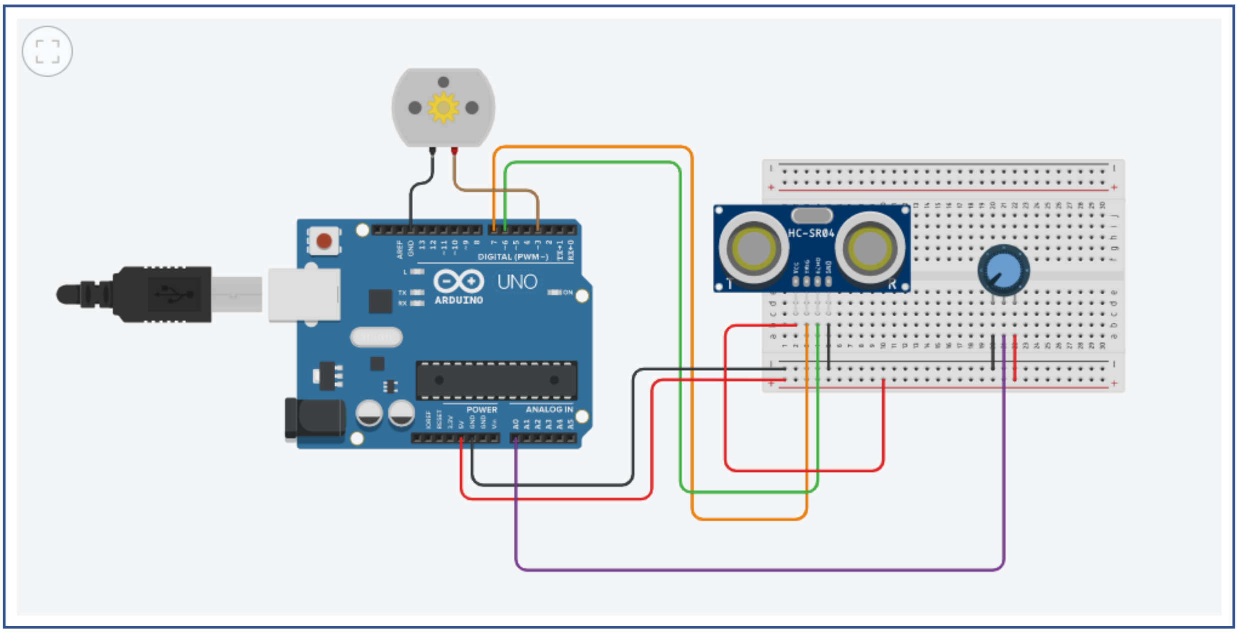


Fig. Circuit schematic using Tinkercad

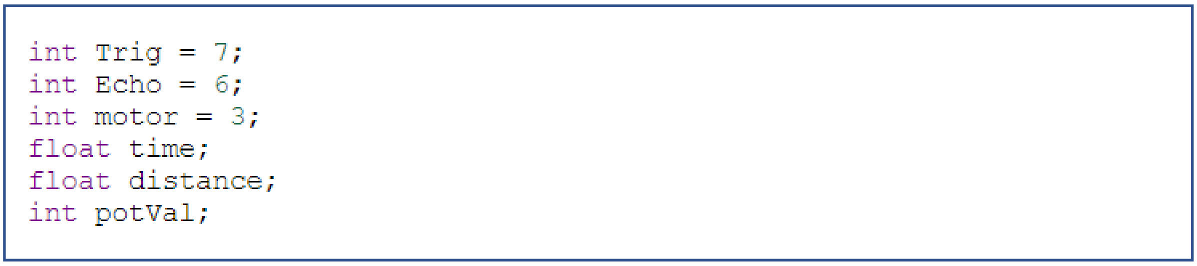
I have decided to use the breadboard instead of just direct connections in order to give ground and +5V supply to the rail because of the limited pins for ground on Arduino.

Code for the project :

For presentation purpose, I have decided to break the code into 3 parts :

- Initialization
- Void setup
- Void loop

Initialization :



```
int Trig = 7;
int Echo = 6;
int motor = 3;
float time;
float distance;
int potVal;
```

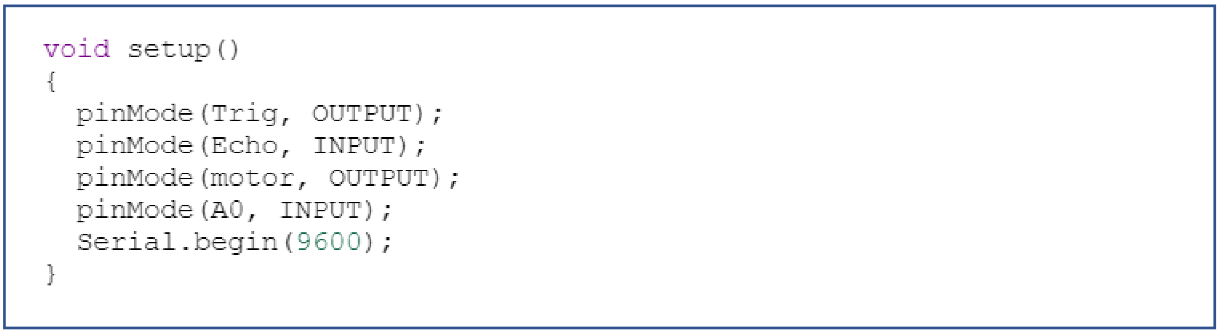
1. The 4 pins of the Ultrasonic Sensor (Vcc, Trig, Echo, Gnd) in which the end pins are connected to respective rails on breadboard while the Trig pin and Echo pin are assigned pin 7 and pin 6 respectively.

2. Also motor is connected to pin 3;

3. 'time', 'distance' are the two float values declared since it is expected that there data can be in decimals

4. The potVal is declared with 'int' datatype since the potentiometer is further connected to analog pin which returns integer value between 0 to 1023.

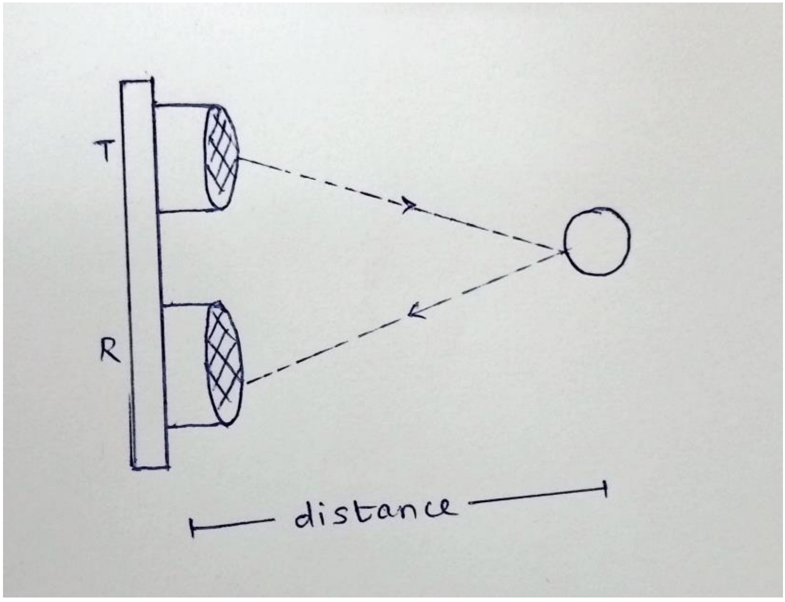
Void setup :



```
void setup()
{
  pinMode(Trig, OUTPUT);
  pinMode(Echo, INPUT);
  pinMode(motor, OUTPUT);
  pinMode(A0, INPUT);
  Serial.begin(9600);
}
```

1. The Trig pin of ultrasonic sensor is set as an OUTPUT since that is basically going to transmit ultrasonic sound waves i.e. give something ‘out’.

2. The echo pin is set as an INPUT since after reflection of the sound waves, the echo pin will receive it i.e. take in inside itself as an input.

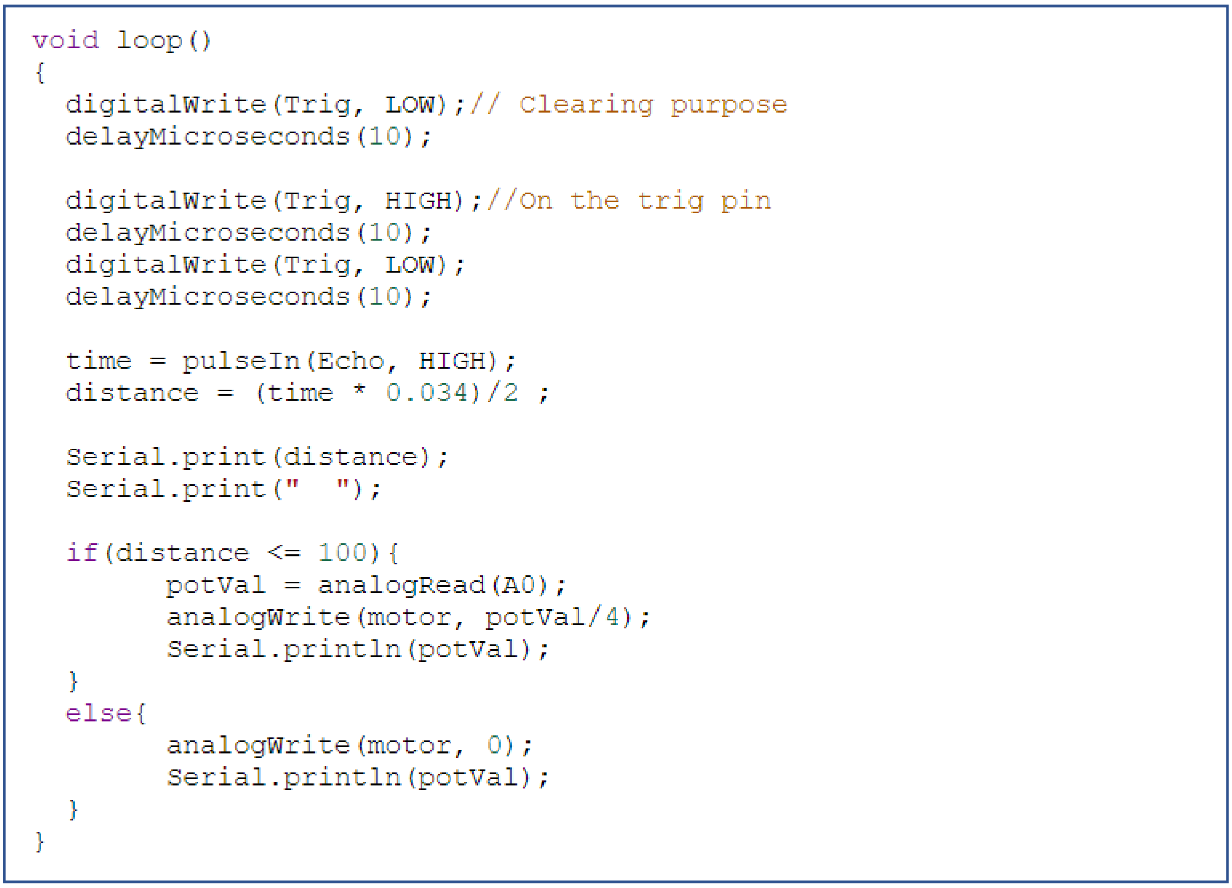


3. For The motor is connected to pin 3 as an OUTPUT component.

4. For the potentiometer, I have assigned analog pin A0 and I have ‘take’ the values from this pin, hence assigned it as INPUT in pinMode function.

5. I had to establish the communication between the Serial monitor and Arduino so that I can see the desired values.

Void loop :



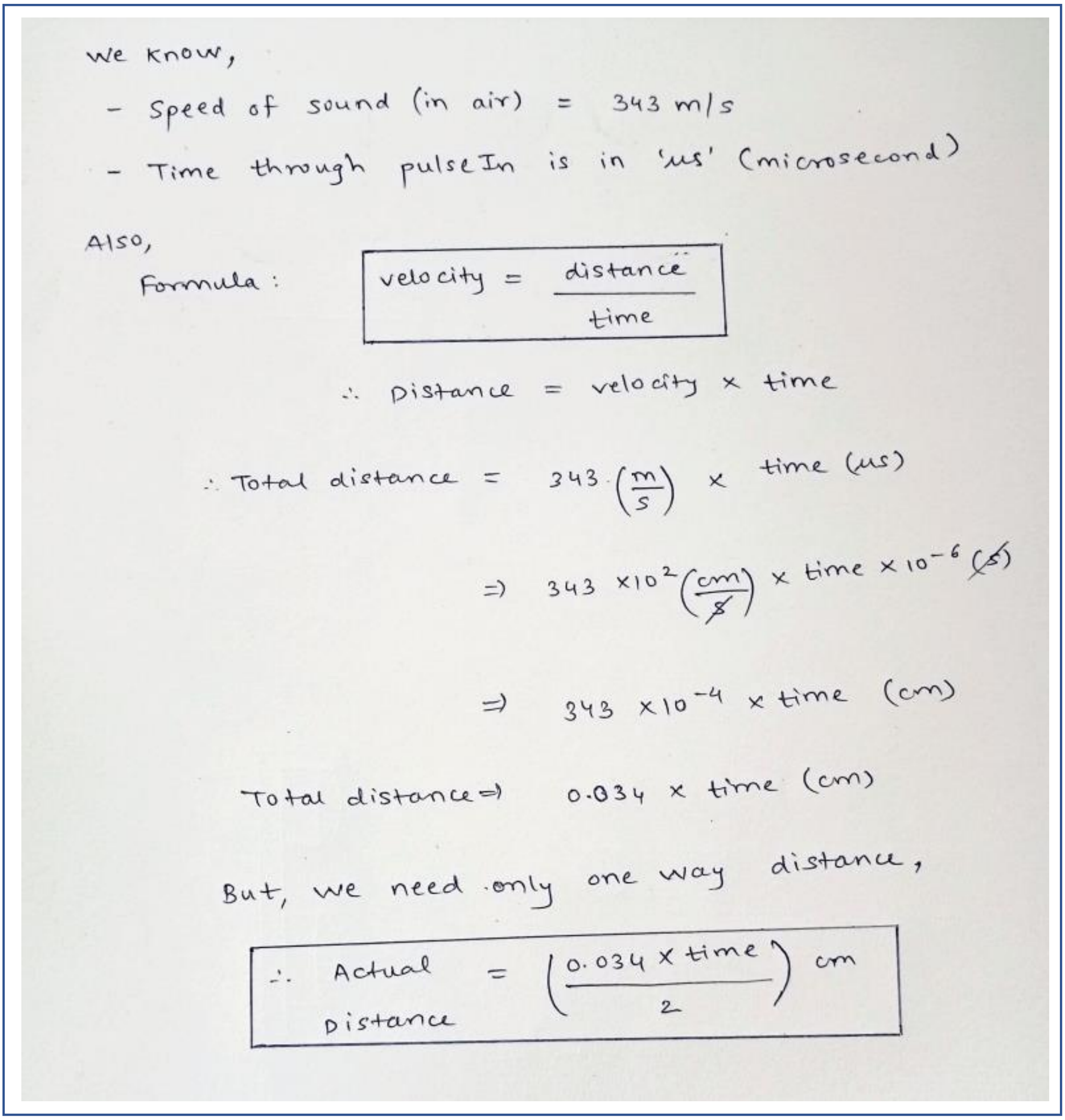
```
void loop()
{
  digitalWrite(Trig, LOW);// Clearing purpose
  delayMicroseconds(10);

  digitalWrite(Trig, HIGH);//On the trig pin
  delayMicroseconds(10);
  digitalWrite(Trig, LOW);
  delayMicroseconds(10);

  time = pulseIn(Echo, HIGH);
  distance = (time * 0.034)/2 ;

  Serial.print(distance);
  Serial.print("  ");

  if(distance <= 100){
        potVal = analogRead(A0);
        analogWrite(motor, potVal/4);
        Serial.println(potVal);
  }
  else{
        analogWrite(motor, 0);
        Serial.println(potVal);
  }
}
```

1. First of all, it’s always better to clear any previous signal (if it might be present) on a Pin. Hence, we set the Trig pin as LOW for some 10 microseconds.

2. After that, now we need to prepare for obstacle detection, hence we set Trig pin HIGH and then immediately set it LOW just to send a sound wave.

3. Now, we turn on the Echo Pin in order to receive back the sound wave which was emitted and calculate time taken for whole process. For this, we have ‘**pulseIn**’ function which returns time in ‘**microseconds**’ if HIGH.

4. Now is the actual distance calculation between sensor and object.



We know,

- Speed of sound (in air) = 343 m/s
- Time through pulseIn is in 'us' (microsecond)

Also,

Formula : $\boxed{\text{velocity} = \frac{\text{distance}}{\text{time}}}$

$\therefore$ Distance = velocity × time

$\therefore$ Total distance = $343 \left(\frac{m}{s}\right) \times \text{time (us)}$

$\Rightarrow 343 \times 10^{2} \left(\frac{cm}{\cancel{s}}\right) \times \text{time} \times 10^{-6} (\cancel{s})$

$\Rightarrow 343 \times 10^{-4} \times \text{time} \quad (cm)$

Total distance $\Rightarrow$ 0.034 × time (cm)

But, we need only one way distance,

$\boxed{\therefore \text{Actual Distance} = \left(\frac{0.034 \times \text{time}}{2}\right) cm}$

5. Now, (if the distance value is <= 100), then we will control motor, not by digital method but by analog method. For that, the motor has to be connected to a PWM(pulse width modulation) pin. Now, we get the value from A0 pin because of potentiometer.

6. We control motor by analogWrite. We give the pin 3, and (potVal/4) since the analogWrite function accepts values only between 0 to 255.

   Basically,

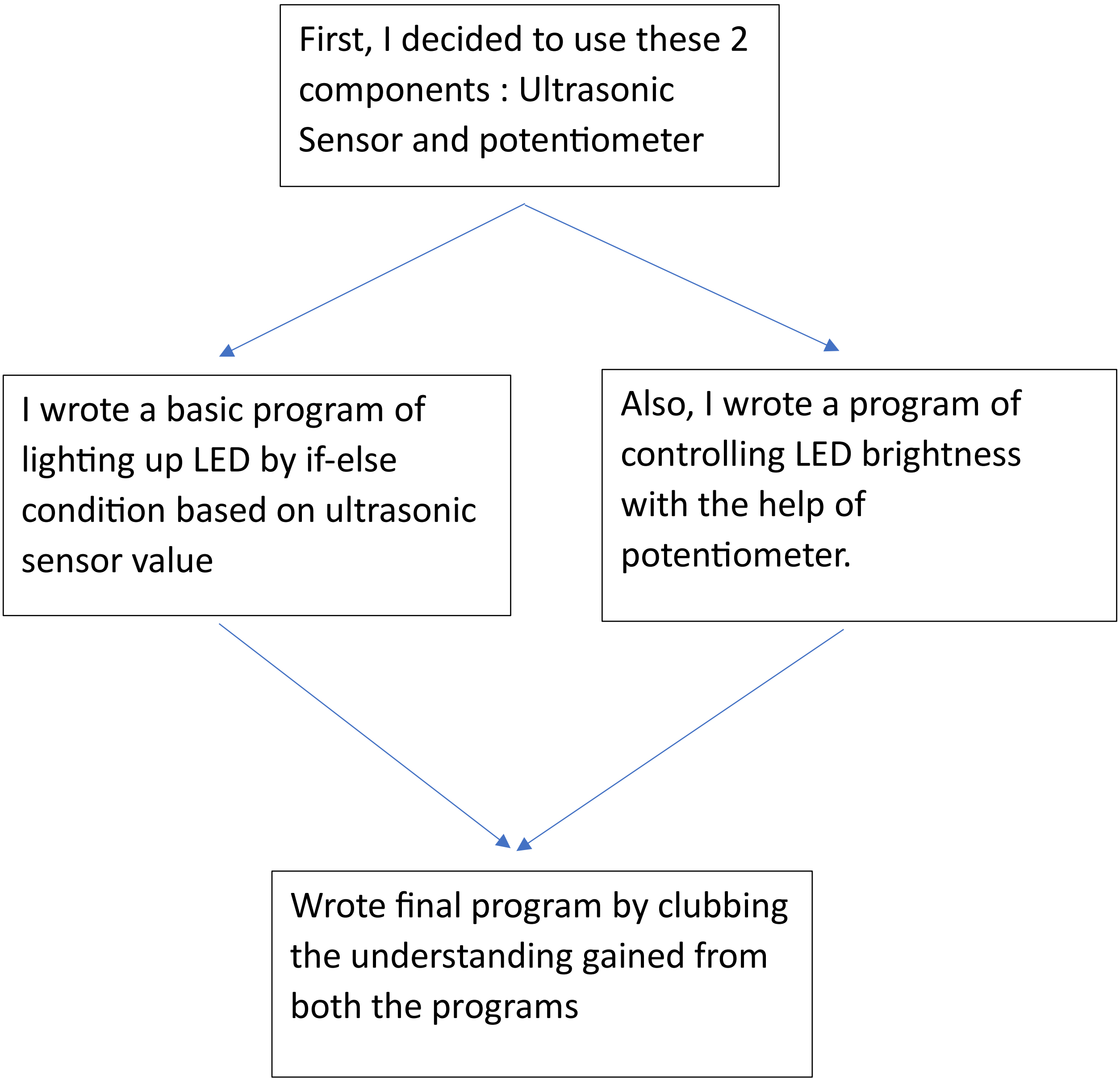
   At 1023 analog value, motor will run fastest (255)
   At 0 analog value, motor will not run (0)

7. else, we don't want to make the motor run.

Note :

This project is actually in relation with Task#0, where I have discussed steering mechanism (to turn left, right, etc) in great detail and how it can be achieved by controlling motors.

Thought process :



First, I decided to use these 2 components : Ultrasonic Sensor and potentiometer

I wrote a basic program of lighting up LED by if-else condition based on ultrasonic sensor value

Also, I wrote a program of controlling LED brightness with the help of potentiometer.

Wrote final program by clubbing the understanding gained from both the programs