

```
//constant variable will not change

const int Trigpin = 31; // trigger pin (P5.6)

const int Ecopin = 32; // eco pin (P3.7)

const int PWMpin = 38; // PWM signal generated (P2.4)

const int DIRpin = 37; // motor direction (P2.7)

int flag = 0;

int duration;

float distance;


void setup() {
    // put your setup code here, to run once:
    Serial.begin(9600); //initialize serial communication at 9600 bits per second
    pinMode(PWMpin, OUTPUT); // set pin P2.4 as PWM output
    pinMode(DIRpin, OUTPUT); // set pin P2.7 as outout for motor direction
    pinMode(Trigpin, OUTPUT); // set pin P5.6 as trigger output (to generate pulse)
    pinMode(Ecopin, INPUT); // set pin P3.7 as input to receive signal from sensor
}


void loop()
{
    digitalWrite(Trigpin, LOW);
    delay(60);
    digitalWrite(Trigpin, HIGH);
    delay(60); // 60ms time delay
    digitalWrite(Trigpin, LOW);
    duration = pulseIn(Ecopin, HIGH);
    distance = duration / 58.2; // Centimeter unit
    Serial.println(distance);
}
```

```
if (flag == 0)
{
    if (distance >= 10 && distance <= 50) // total length of the board, the distance you want to travel
    {
        digitalWrite(DIRpin, HIGH); //P2.7 = low (0) = anticlockwise
        analogWrite(PWMPin, 229); //Motor start
    }
}
```

```
if (distance > 50)
{
    analogWrite(PWMPin, 0);
    delay(5000);
    flag = 1;
}
}
```

```
if (flag == 1)
{
    if (distance > 20 && distance < 55) // total length + skid (example 5cm skid but 2 cm error tolerance)
    {
        digitalWrite(DIRpin, LOW); //P2.7 = low (0) = anticlockwise
        analogWrite(PWMPin, 229); //Motor start
    }
}
```

```
if (distance < 23) // min length + skid
{
    analogWrite(PWMPin, 0);
    delay(5000);
    flag = 0;
}
```

}

}

}