#include <LiquidCrystal.h>

#define trig 2 //attach pin D2 Arduino to

pin Trig of HC-SR04

#define echo A3 // attach pin A3 Arduino

to pin Echo of HC-SR04

#define GREEN\_LED 11 //attach pin D11

Arduino to GREEN LED

#define YELLOW\_LED 10 //attach pin D10

Arduino to YELLOW LED

#define ORANGE\_LED 9 //attach pin D9

Arduino to ORANGE LED

#define RED\_LED 8 //attach pin D8 Arduino

to RED LED

#define rs 13 //attach pins of Arduino to

LCD pins

#define en 12

#define d4 4

#define d5 5

#define d6 6

#define d7 7

LiquidCrystal lcd (rs,en,d4,d5,d6,d7);

float duration; // variable for the duration of

sound wave travel

float Distance; // variable for the distance

measurement in cm

int precentage; // variable for the percentage of

rubbish level in %

float rubbish\_height;

float CheckX;

float Checkingmills;

float TimeXwasTrue;

void setup() {

Serial.begin(9600); // Serial

Communication is starting with 9600 of

baudrate speed

pinMode(trig, OUTPUT); // Sets the

trigPin as an OUTPUT

pinMode(echo, INPUT); // Sets the

echoPin as an INPUT

pinMode(GREEN\_LED, OUTPUT); // Sets

the green led Pin as an OUTPUT

pinMode(YELLOW\_LED, OUTPUT); // Sets

the yellow led Pin as an OUTPUT

pinMode(ORANGE\_LED, OUTPUT); // Sets

the orange led Pin as an OUTPUT

pinMode(RED\_LED, OUTPUT); // Sets the

red led Pin as an OUTPUT

lcd.begin(16,2);}

void loop() {

digitalWrite(trig, LOW); // Clears the trigPin

condition

delay(2);

digitalWrite(trig, HIGH);

delay(15); // Sets the trigPin HIGH (ACTIVE)

for 15 microseconds

digitalWrite(trig, LOW);

duration = pulseIn(echo, HIGH); // Reads the

echoPin, returns the sound wave travel time in

microseconds

// Calculating the distance in cm ,, the velocity

of sound =340 m/s = 0.034 cm/us

Distance = (duration\*0.034)/ 2; // Speed of

sound wave divided by 2 (go and back)

percentage = 100-(Distance\*100)/120;

rubbish\_height = 120 - Distance ;

if (Distance < 120 ) {

digitalWrite(GREEN\_LED, HIGH);

}

else {

digitalWrite(GREEN\_LED, LOW);

}

if (Distance < 90) {

digitalWrite(YELLOW\_LED, HIGH);

}

else {

digitalWrite(YELLOW\_LED, LOW);

}

if (Distance < 60 ) {

digitalWrite(ORANGE\_LED, HIGH);

}

else {

digitalWrite(ORANGE\_LED, LOW);

}

if ( Distance < 20) {

digitalWrite(RED\_LED, HIGH);

}

else {

digitalWrite(RED\_LED, LOW);}

Serial.print("Rubbish height:");

Serial.println("cm");

Serial.print("DustbinLevel:");

Serial.print(percentage);

Serial.println("%");

lcd.clear(); // print water level in LCD in %

lcd.setCursor(0,0);

lcd.print("DustbinLevel:");

lcd.print(percentage);

lcd.println("%");

delay(500);

lcd.print("Rubbishheight: ");

lcd.print(rubbish\_height);

if (percentage > 80){

CheckX == false;

// if rubbish level >80% shown on lcd

if (percentage > 80 && CheckX == false ){

CheckX = true;

TimeXwasTrue = millis();

}

Checkingmills = millis();

if ( CheckX && Checkingmills - TimeXwasTrue

>= 5000 ) {

Serial.println("Clean the Dustbin");

lcd.setCursor(0,1);

lcd.print("Clean the Dustbin");

delay(1000);

}}

if (percentage <80){

lcd.print("");

}

}

[SEEM\_S2\_G11\_InstrumentationLab\_Group Report.pdf](https://github.com/OmarBatis/Smart-Dustbin/files/8867661/SEEM\_S2\_G11\_InstrumentationLab\_Group.Report.pdf)

https://www.tinkercad.com/things/eQnHSMiG9wz-copy-of-water-level-monitoring/editel?sharecode=y5YqmozRuTmKy-s\_u5r16v6paPEVbLBO2OogtHcH6Pw