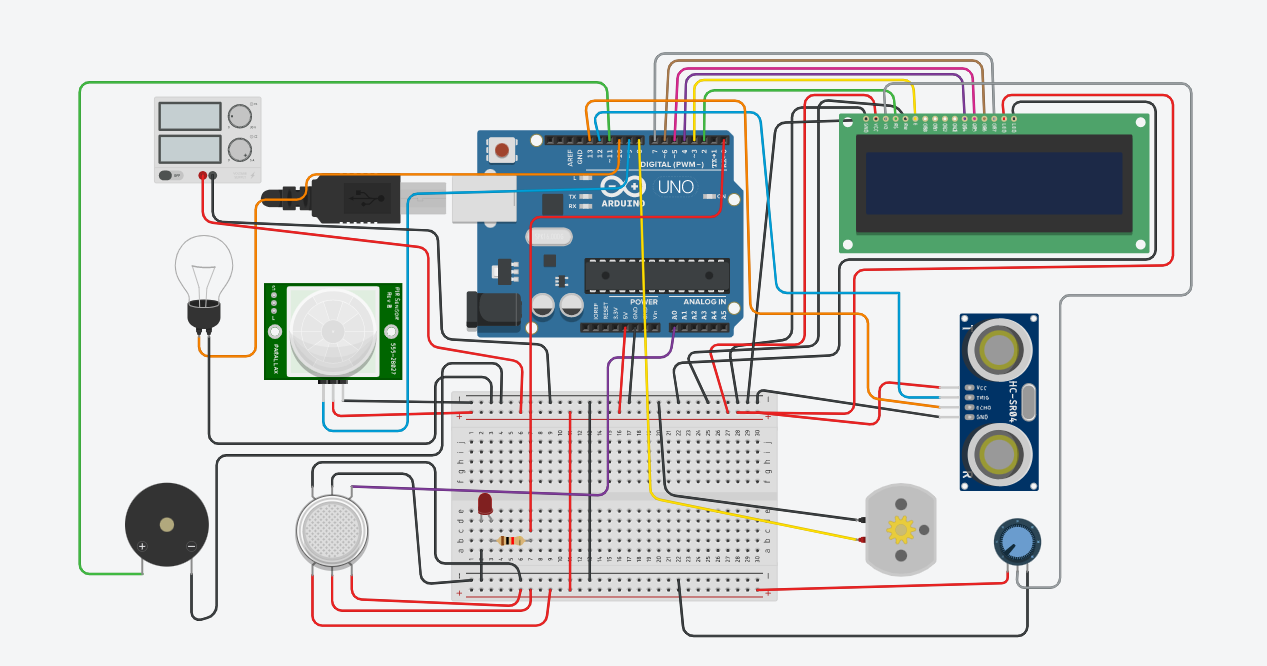
### IoT Based Safety Gadget for Child Safety Monitoring and Notification

Team ID: PNT2022TMID18238 Submitted by: Saran S

**ASSIGNMENT 1**

**SMART HOME IN TINKERCAD USING 2+ SENSORS**

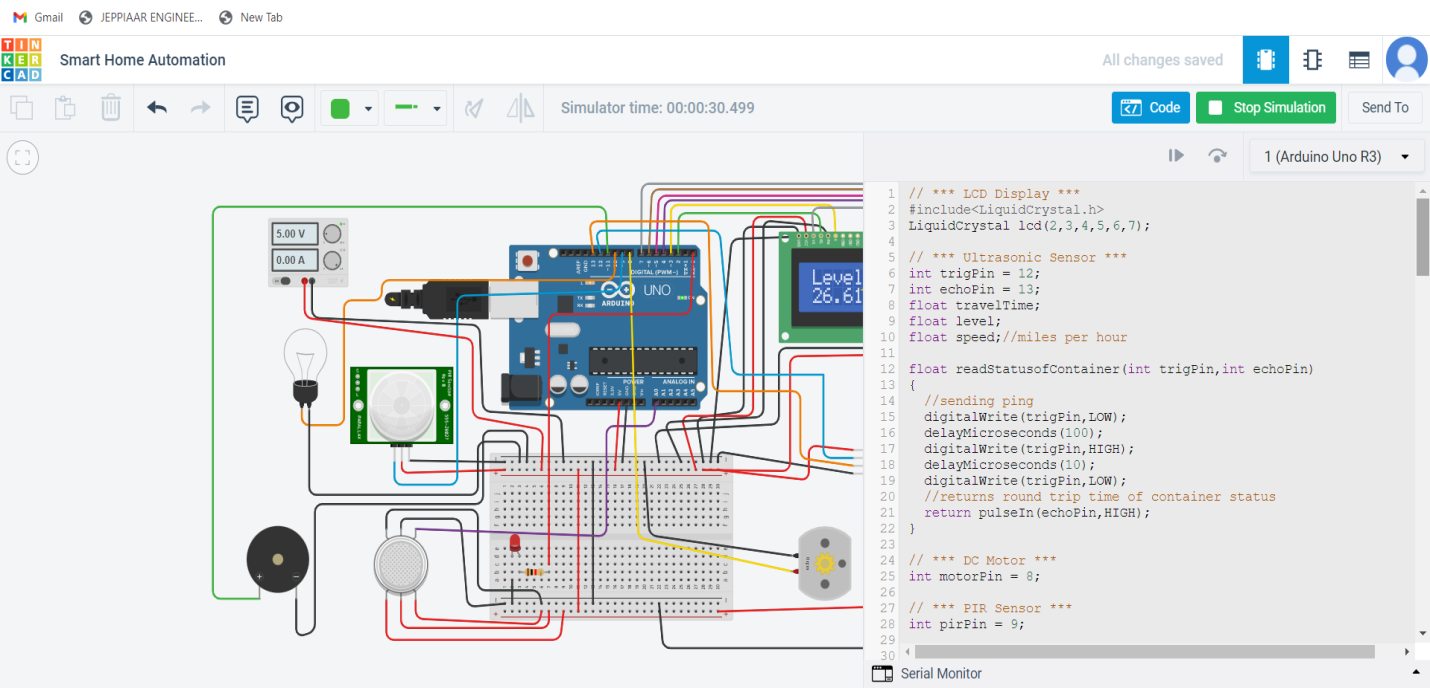
* **Circuit Diagram**



* **Components Required:**

|  |  |  |
| --- | --- | --- |
| **Name** | **Quantity** | **Component** |
| U1 | 1 | Arduino Uno R3 |
| DIST1 | 1 | Ultrasonic Distance Sensor |
| M1 | 1 | DC Motor |
| U2 | 1 | LCD 16 x 2 |
| P1 | 1 | 21 , 5 Power Supply |
| PIR1 | 1 | 20.722593638638443 , -237.58262756842853 , -197.9092690878765 , -253.1130883508938 PIR Sensor |
| L2 | 1 | Light bulb |
| GAS1 | 1 | Gas Sensor |
| PIEZO1 | 1 | Piezo |
| R1 | 1 | 10 kâ„¦ Resistor |
| Rpot1 | 1 | 250 kâ„¦ Potentiometer |
| D1 | 1 | Red LED |
| R2 | 1 | 330 â„¦ Resistor |

* **Running Simulation in Tinkercad**



* **Code:**

// \*\*\* LCD Display \*\*\*

#include<LiquidCrystal.h>

LiquidCrystal lcd(2,3,4,5,6,7);

// \*\*\* Ultrasonic Sensor \*\*\*

int trigPin = 12;

int echoPin = 13;

float travelTime;

float level;

float speed;//miles per hour

float readStatusofContainer(int trigPin,int echoPin)

{

//sending ping

digitalWrite(trigPin,LOW);

delayMicroseconds(100);

digitalWrite(trigPin,HIGH);

delayMicroseconds(10);

digitalWrite(trigPin,LOW);

//returns round trip time of container status

return pulseIn(echoPin,HIGH);

}

// \*\*\* DC Motor \*\*\*

int motorPin = 8;

// \*\*\* PIR Sensor \*\*\*

int pirPin = 9;

// \*\*\* Light \*\*\*

int lightPin = 10;

// \*\*\* Gas Sensor \*\*\*

int gasPin = A0;

int threshold = 400;

// \*\*\* Piezo \*\*\*

int buzzPin = 11;

// \*\*\* LED \*\*\*

int ledPin = 0;

void setup()

{

Serial.begin(9600);

// \*\*\* LCD Display \*\*\*

lcd.begin(16,2);

// \*\*\* Ultrasonic Sensor \*\*\*

pinMode(trigPin,OUTPUT);

pinMode(echoPin,INPUT);

// \*\*\* DC Motor \*\*\*

pinMode(motorPin,OUTPUT);

// \*\*\* PIR Sensor \*\*\*

pinMode(pirPin,INPUT);

// \*\*\* Light \*\*\*

pinMode(lightPin,OUTPUT);

// \*\*\* Gas Sensor \*\*\*

pinMode(gasPin,INPUT);

// \*\*\* Piezo \*\*\*

pinMode(buzzPin, OUTPUT);

// \*\*\* LED \*\*\*

pinMode(ledPin,OUTPUT);

}

void loop()

{

// \*\*\* Trash can monitoring \*\*\*

// Trash can height 5 inches

travelTime = readStatusofContainer(trigPin,echoPin);//microseconds

travelTime = travelTime/1000000;//seconds

travelTime = travelTime/3600;//hours

speed = 60.0;//miles per hour(86.4 for 5 inches)

level = speed \* travelTime;//miles

level = level/2;//because travelTime is round trip time

level = level \* 63360;//inch

if(level <= 4.5)

{

//dispaly status

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Trash Level:");

lcd.setCursor(0,1);

lcd.print(level);

lcd.print(" inches");

delay(100);

}

else

{

//dispaly status

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Trash is full");

lcd.setCursor(0,1);

lcd.print(level);

lcd.print(" inches away");

delay(100);

}

// \*\*\* Water level monitoring \*\*\*

// Water tank height 20 inches

travelTime = readStatusofContainer(trigPin,echoPin);//microseconds

travelTime = travelTime/1000000;//seconds

travelTime = travelTime/3600;//hours

speed = 240.1;//miles per hour(345.3 for 20 inches)

level = speed \* travelTime;//miles

level = level/2;//because travelTime is round trip time

level = level \* 63360;//inch

if(level <= 19.0)

{

//dispaly status and Turn on motor

digitalWrite(motorPin,HIGH);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Level: Motor");

lcd.setCursor(0,1);

lcd.print(level);

lcd.print(" in On");

delay(100);

}

else

{

//dispaly status and Turn off motor

digitalWrite(motorPin,0);

lcd.clear();

lcd.setCursor(0,0);

lcd.print("Level: Motor");

lcd.setCursor(0,1);

lcd.print(level);

lcd.print(" in Off");

delay(100);

}

// \*\*\* Motion Detection

if(digitalRead(pirPin)==HIGH)

digitalWrite(lightPin, HIGH);

else

digitalWrite(lightPin, LOW);

delay(100);

// \*\*\* Detects flammable gases \*\*\*

if(analogRead(gasPin) >= threshold)

{

digitalWrite(ledPin,HIGH);

digitalWrite(buzzPin,HIGH);

}

else

{

digitalWrite(ledPin,LOW);

digitalWrite(buzzPin,LOW);

}

delay(100);

}

* **Output:**

