

GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

DATE	18 November 2022
TEAM ID	PNT2022TMID26516
PROJECT NAME	GAS LEAKAGE MONITORING AND ALERTING SYSTEM FOR INDUSTRIES

An Introduction To IoT-Based Gas Leakage Detection And Monitoring Systems :

Safety plays a critical role in today's world and it is vital that certain solutions are implemented in places of work and living. Whether it is electricity or oil and gas, working or living in hazardous conditions demand certain safety protocols.

Liquefied Petroleum Gas (LPG) is a type of natural gas liquefied under extreme pressure and contained in a metal cylinder. LPG is extremely sensitive to fire and causes a great disaster if exposed to any fire source without precaution

LPG is more widely available than any other natural gas and is primarily used for cooking. Unfortunately, its broad use makes the event of gas leakage or even a blast standard. Therefore, there is a need to develop a gas leakage detection and monitoring system.

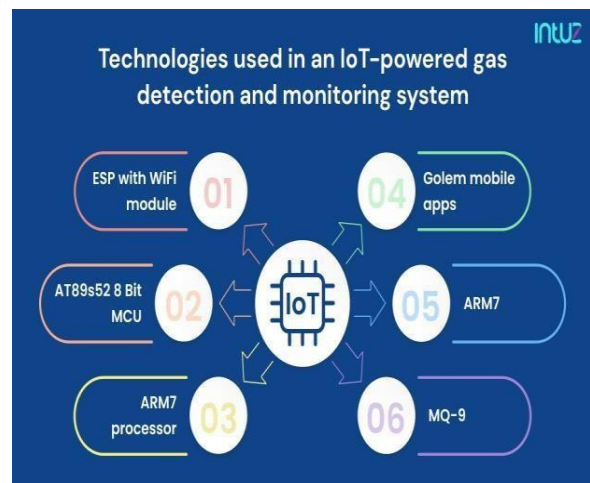
The solution could detect gas leakage, send an alert to the end-user via an SMS or a buzzer, and feature an exhaust fan that gets activated once the gas or fire is detected.

Suppose the gas in the cylinder indicates a value where the remaining percentage level falls below the threshold level set for gas. In that case, the gas cylinder company should be notified immediately to refill the cylinder or replace it. The sensor is also handy for monitoring gas usage over a period.

Benefits of IoT-based apps used in gas leakage detection :

A human nose comprises 400 different types of scent receptors that enable us to smell approximately 1 trillion various odour. However, most of us cannot identify between the different gasses present in the atmosphere. That is where gas detection sensors come in handy.

They are most commonly used to develop an IoT-powered system and identify the variation of toxic gasses in an industrial facility. It helps benefit the refineries and factories by safeguarding them from unexpected threats such as gas leakage and explosions. Here are the top benefits of IoT-based apps used in gas leakage detection.



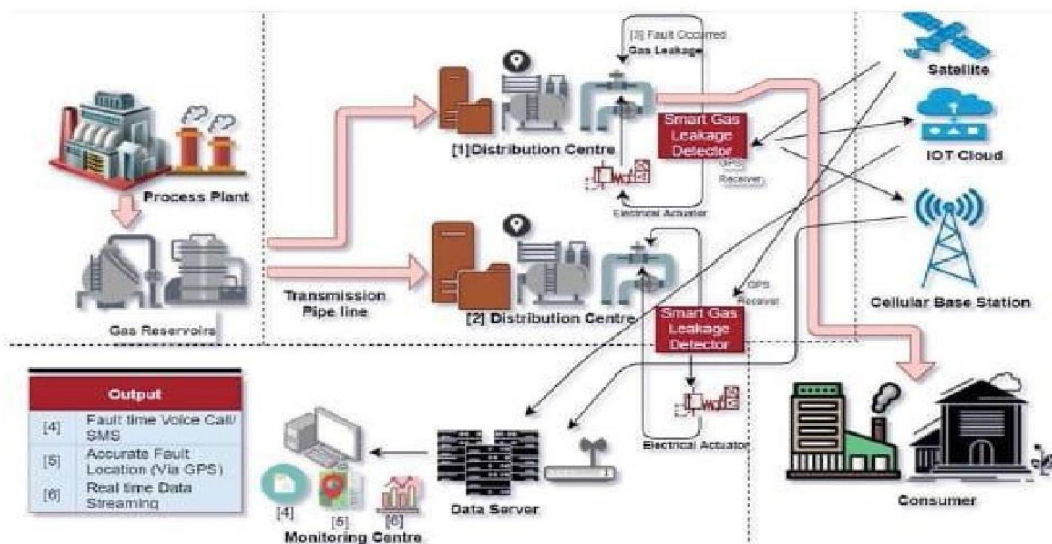
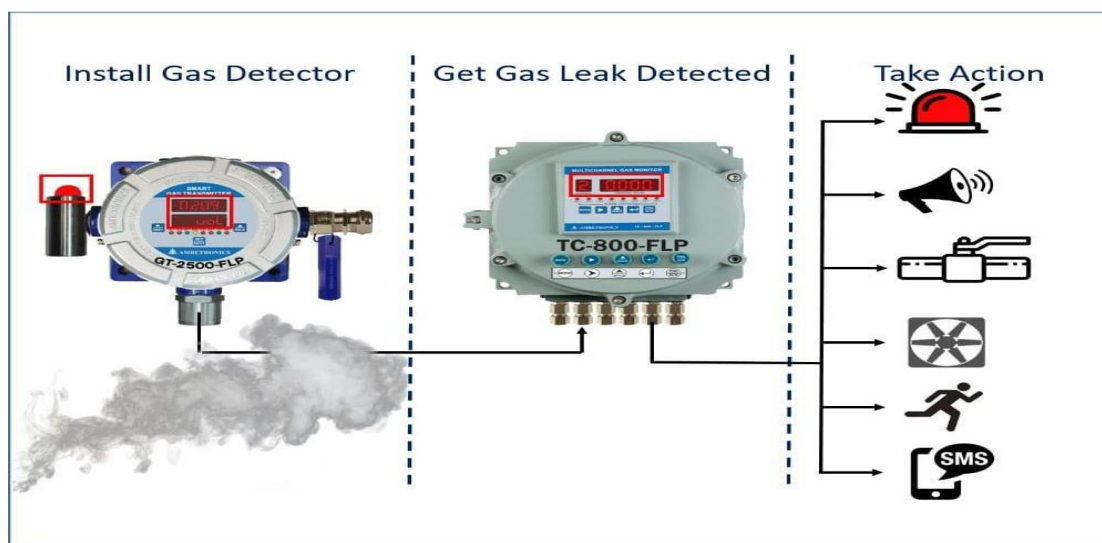
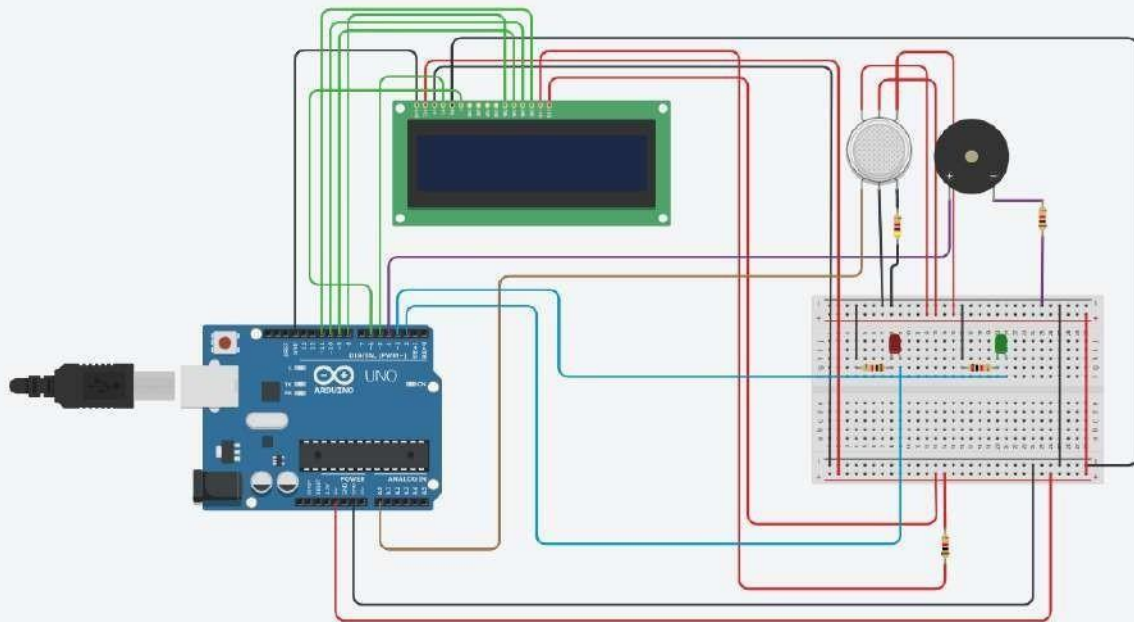


Figure 6. System and control diagram of outdoor gas leakage detection.



CIRCUIT DIAGRAM:

<https://www.tinkercad.com/things/dgmsD2r8Vgn-brilliant-bigery-hillar/editel?tenant=circuits>



SCHEMATIC DIAGRAM:

CODE:

```
#include<Seívo.h>
Seívo myseívo;
int pos = 0;
int val;int motoí = 8; include
#<LiquidCíystal.h>
LiquidCíystal lcd(12, 11, 5, 4, 3, 2);
#include <íinyGPS.h>
#include <SoftwaíeSeíal.h>
SoftwaíeSeíal mySeíal(9, 10);byte
tx=1;
íinyGPS gps; / Cíeates a new instance of the íinyGPS object

const int SPEAKER = 6; const
int LED_RED = 7; const int
LED_YELLOW = 10;int
Relay=7;
int tempC_1 = 0; / set initial tempC 0° foí all LM35int
smkC_1 = 0; / set initial tempC 0° foí all MQ 2 const
int SensoíPin1 = A0; / fiíe input sensoí pin const int
SensoíPin2 = A1;
Stíng textFoíSMS;
void setup()
{
pinMode(motoí, OUI'PUÍ');
pinMode(tx, OUI'PUÍ');
pinMode(Relay, OUI'PUÍ');
myseívo.attach(13);
pinMode(SPEAKER,
OUI'PUÍ');lcd.begin(14, 2);
delay(100);
pinMode(SensoíPin1, INPUT);
pinMode(SensoíPin2, INPUT);
pinMode(SPEAKER,
OUI'PUÍ');

pinMode(LED_RED, OUI'PUÍ');
pinMode(LED_YELLOW, OUI'PUÍ'); / Set contíol pins to be outputs
digitalWíite(LED_RED, LOW);
digitalWíite(LED_YELLOW, LOW);//set both motoís off foí staít-up
mySeíal.begin(9600);
Seíal.begin(9600); / Staít the seíal connection with the computeí
}
void loop()
{
int tempC_1 = analogRead(SensoíPin1);int
SmkC_1 = analogRead(SensoíPin2);
tempC_1 = analogRead(SensoíPin1); / íeád the value fíom the LM35 sensoí
tempC_1 = (5.0 * tempC_1 * 100.0) / 1024.0; / conveít the analog data to tempeíatuíesmkC_1 =
analogRead(SensoíPin2); / íeád the value fíom the MQ 2 sensoí
smkC_1 = (5.0 * smkC_1 * 100.0) / 1024.0; / conveít the analog data to tempeíatuíedelay(50);

bool newData = false;
unsigned long chaís;
unsigned shoít sentences, failed;
```

```

foí (unsigned long staít = millis(); millis() -staít < 1000;)
{
while (Seíal.available())
{
chaí c = Seíal.íead();if
(gps.encode(c))
newData = tíue;
}
}

```

```

if (tempC_1 >= 50 || smkC_1 >= 50)
{
digitalWíte(Relay,HIGH);
val = analogRead(pos);
val = map(val, 0, 1023, 0, 180);
myseívo.wíte(val);
delay(50); digitalWíte(motoí,
HIGH);
digitalWíte(LED_RED,
HIGH);
digitalWíte(LED_YELLOW,
HIGH);lcd.cleaí();
lcd.setCuísoí(0, 0); lcd.pínt("
I'HERE IS FIRE ");
lcd.setCuísoí(0, 1);
lcd.pínt(" NOI' SAFE HERE ");
delay(100);
lcd.cleaí(); lcd.pínt("Sending
SMS...");delay(100);

```

```

tone(SPEAKER, 1047, 500);
delay(200);
tone(SPEAKER, 1109, 1000);
delay(200);
tone(SPEAKER, 1175, 100);
delay(5);

```

```

float flat, flon; unsigned
long age;
gps.f_get_position(&flat, &flon, &age);
Seíal.pínt("A'+CMGF=1'\í"); delay(100);
Seíal.pínt("A'+CMGS=\"+233266302607\"'\í");
Seíal.pínt("FIRE ALER!'!\í");
delay(100);
Seíal.pínt("A'+CMGS=\"+233266302607\"'\í");
Seíal.pínt("FIRE OCCURED!\í");
delay(100);
Seíal.pínt("A'+CMGS=\"+233266302607\"'\í");
Seíal.pínt("FIRE OCCURED! in'\í");
delay(200);
Seíal.pínt("Latitude = ");
Seíal.pínt(flat == ínyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flat, 6);
Seíal.pínt(" Longitude = ");

```

```

Serial.print(flon == finyGPS::GPS_INVALID_F_ANGLE ? 0.0 : flon, 6);
delay(200);
Serial.println((char)26); / End A1 command with a ^Z, ASCII code 26
delay(200);
Serial.println();
}
else
{
digitalWrite(Relay,LOW);
delay(50);
digitalWrite(LED_RED,
LOW);
digitalWrite(LED_YELLOW,
LOW);digitalWrite(motor1, LOW);
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("  NO FIRE  ");
lcd.setCursor(0, 1);
lcd.print(" ALL SAFE ");
}
}

```

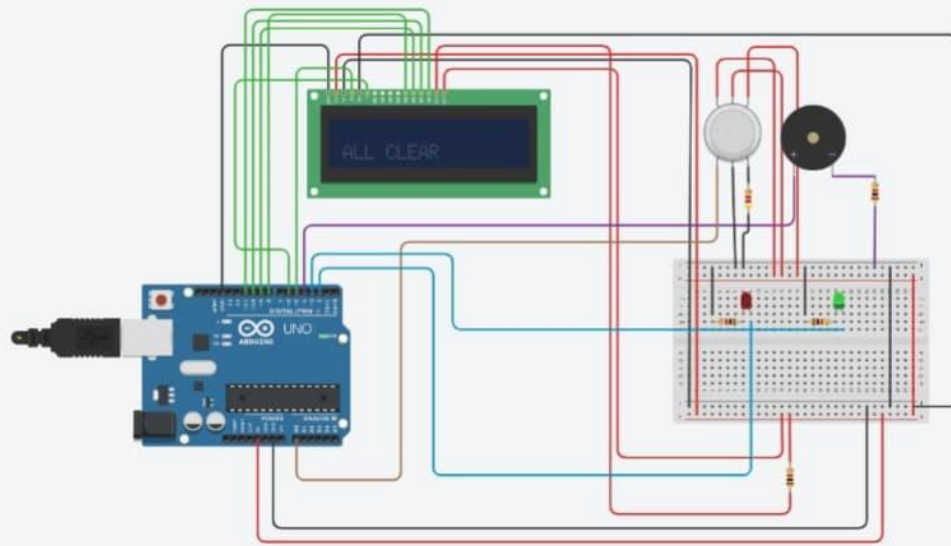
OUTPUTS:



Simulator time: 00:00:29

 Code Stop Simulation

Send To



Share with people and apps

Component
Basic

Search



Resistor



LED



Pushbutton



Potentiometer



Capacitor



Slidewitch



9V Battery

Coin Cell 3V
Battery

1.5V Battery

Breadboard
Small

micro:bit

Arduino Uno
R3

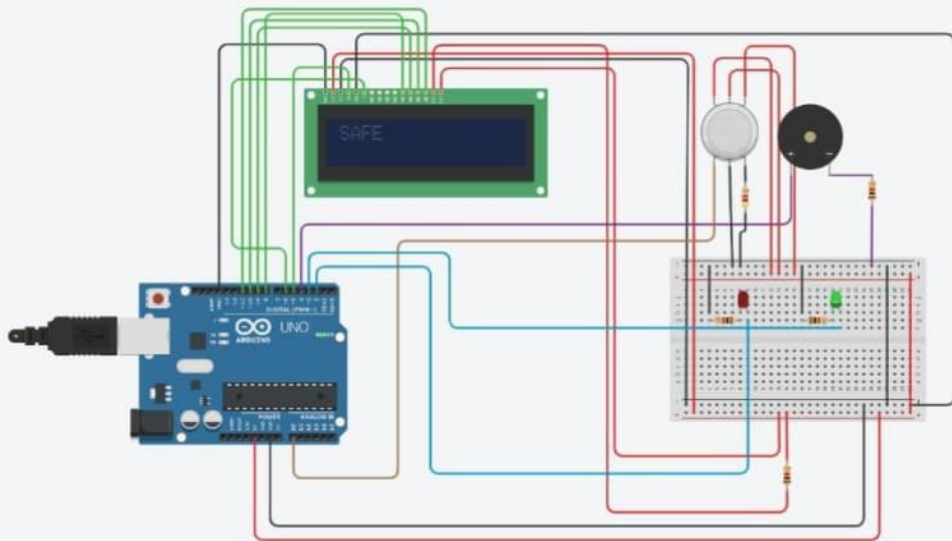


Simulator time: 00:00:30

Code

Stop Simulation

Send To

Components
Basic

Search



Resistor



LED



Pushbutton



Potentiometer



Capacitor



Slideswitch



9V Battery

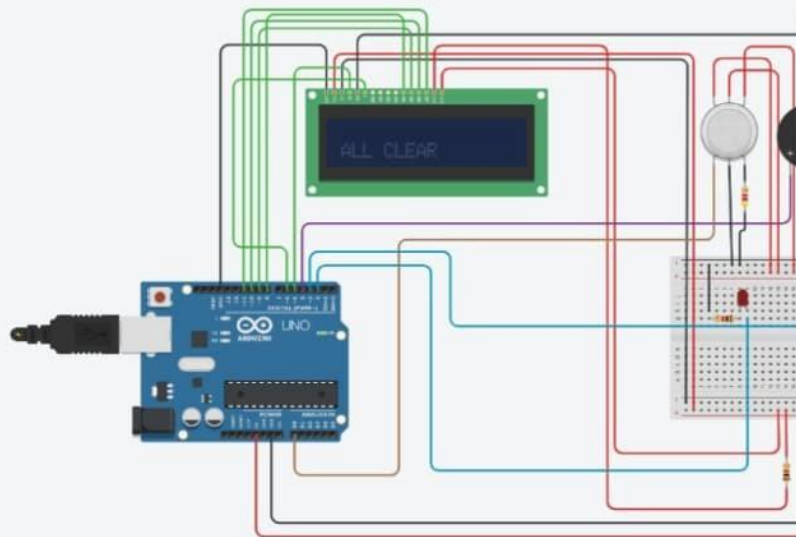
Coin Cell 3V
Battery

1.5V Battery

Breadboard
Small

micro:bit

Arduino Uno
R3

 Serial Monitor

Clear





Simulator time: 00:00:34

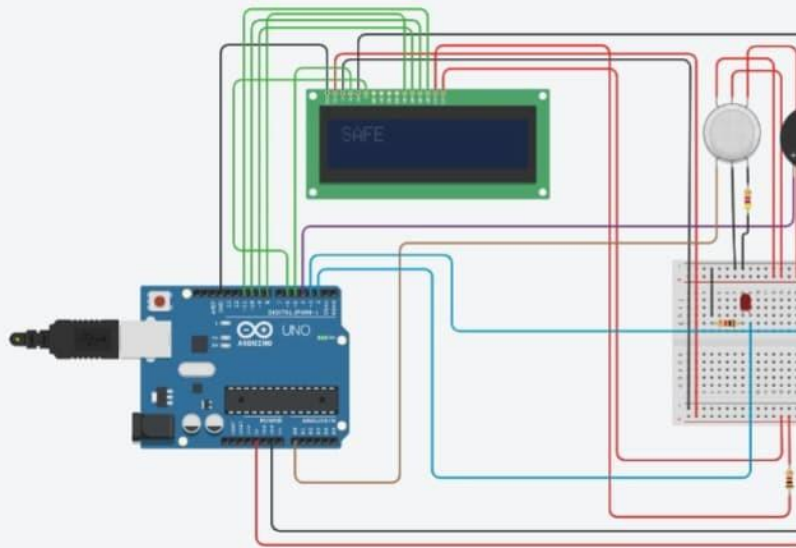
Code

Stop Simulation

Send To



1 (Arduino Uno R3)

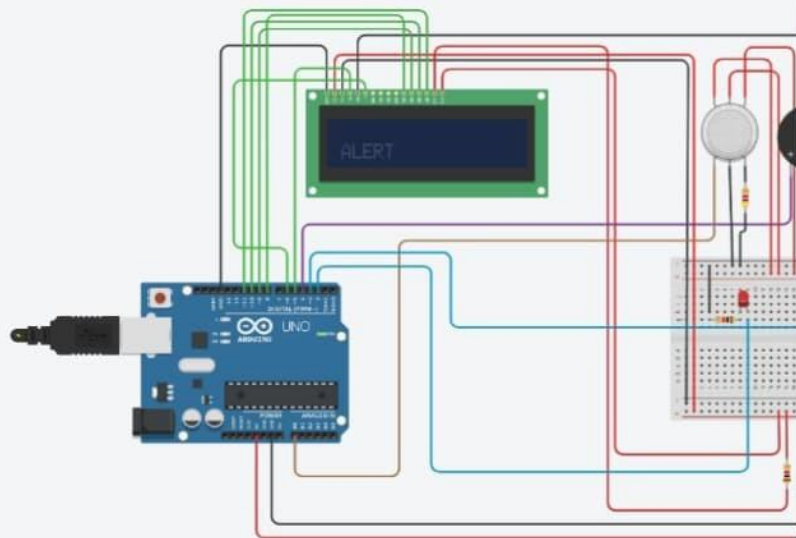


```

1
2 #include <LiquidCrystal.h>
3 LiquidCrystal lcd(5,6,8,9,10,11);
4
5 int greenled = 2;
6 int redled = 3;
7 int buzzer = 4;
8 int sensor = A0;
9 int sensorThresh = 400;
10
11 void setup()
12 {
13   pinMode(greenled, OUTPUT);
14   pinMode(redled, OUTPUT);
15   pinMode(buzzer, OUTPUT);
16   pinMode(sensor, INPUT);
17   Serial.begin(9600);
18   lcd.begin(16, 2);
19 }
20
21 void loop()
22 {
23   int analogValue = analogRead(sensor);
24   Serial.print(analogValue);
25   if (analogValue > sensorThresh)
26   {
27     digitalWrite(greenled, HIGH);
28     digitalWrite(redled, LOW);
29     tone(buzzer, 1000, 10000);
30     lcd.clear();
31     lcd.setCursor(0, 1);
32     lcd.print("ALERT");
33     delay(1000);
34     lcd.clear();
35     lcd.setCursor(0, 1);
36     lcd.print("EVACUATE");

```

Serial Monitor



```

6  int redled = 3;
7  int buzzer = 4;
8  int sensor = A0;
9  int sensorThresh = 200;
10
11 void setup()
12 {
13   pinMode(greenled, OUTPUT);
14   pinMode(redled, OUTPUT);
15   pinMode(buzzer, OUTPUT);
16   pinMode(sensor, INPUT);
17   Serial.begin(9600);
18   lcd.begin(16, 2);
19 }
20
21 void loop()
22 {
23   int analogValue = analogRead(sensor);
24   Serial.print(analogValue);
25   if(analogValue > sensorThresh)
26   {
27     digitalWrite(greenled, HIGH);
28     digitalWrite(redled, LOW);
29     tone(buzzer, 1000, 10000);
30     lcd.clear();

```

 Serial Monitor[illegible]

Clear



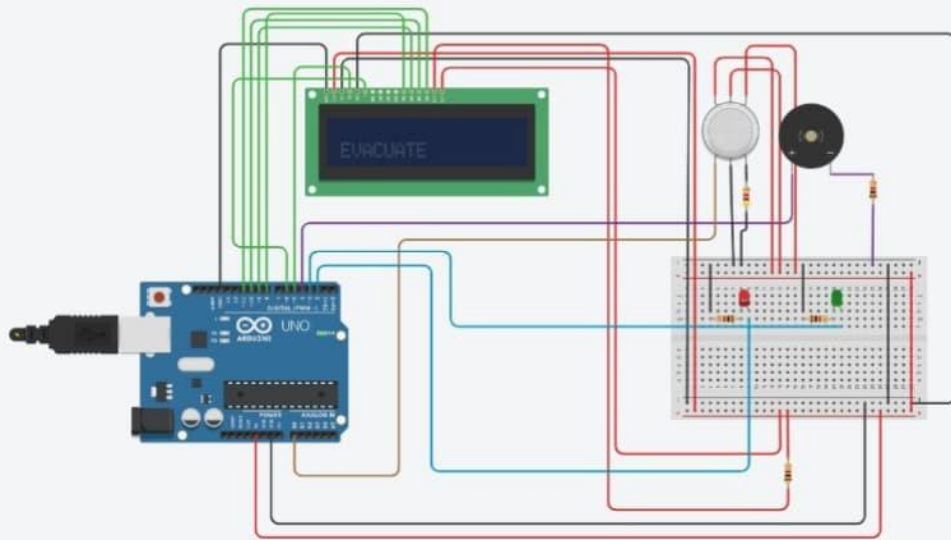


Simulator time: 00:00:03

Code

Stop Simulation

Send To

Components
Basic

Search



Resistor



LED



Pushbutton



Potentiometer



Capacitor



Slideswitch



9V Battery

Coin Cell 3V
Battery

1.5V Battery

Breadboard
Small

micro:bit

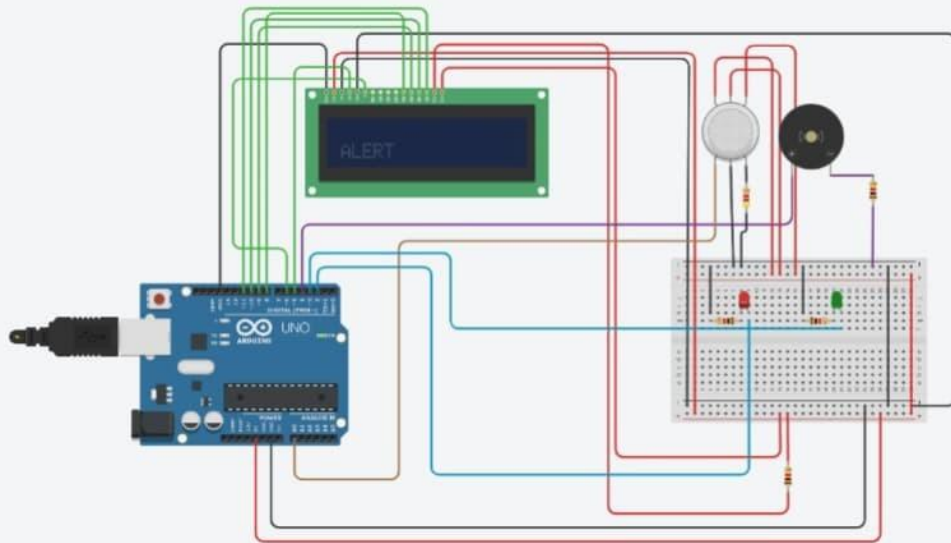
Arduino Uno
R3



Simulator time: 00:00:02

 Code Stop Simulation

Send To

Components
Basic

Search



Resistor



LED



Pushbutton



Potentiometer



Capacitor



Slidswitch



9V Battery

Coin Cell 3V
Battery

1.5V Battery

Breadboard
Small

micro:bit

Arduino Uno
R3