**Assignment -1**

Problem Statement :

lo T-Based Industry - Real-Time River Water Quality Monitoring and Control

System Domain :

Internet of Things

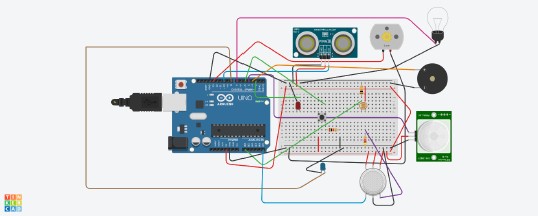
|  |  |
| --- | --- |
| Assignment Date | 28 October 2022 |
| Student Name | R.RAKSHANA |
| Student Roll Number | 623019104013 |
| Maximum Marks | 2 Marks |

# Assignment 1 : Circuit design Home automation system in TinkerCad.

**https://**[**www.tinkercad.com/things/9ylzaArYyOz-Home-Automation**](http://www.tinkercad.com/things/9ylzaArYyOz-Home-Automation)

# / editel?tenant=circuits

**Circuit diagram :**



Program:

Arduino Uno Code : const int pingPin = 10;

const int ledUS = 2; const int light = 7;

const int pir = 4; #define photoSensor AO #define buzzer 3 int const PINO\_SGAS = A5;

int const ledGas = 8; int const button = 5; int const motor

= 13;

void setup()

pinMode(ledUS, OUTPUT); pinMode(light, OUTPUT);

pinMode(buzzer, OUTPUT); pinMode(ledGas, OUTPUT); pinMode(motor, OUTPUT);

pinMode(pir, INPUT); pinMode(button,

INPUT);

pinMode(photoSensor, INPUT);

Serial.begin(9600); void

loop()

long duration, cm; int valLight = analogRead(photoSensor); int

valPIR= digitalRead(pir); int valGAS = analogRead(PINO\_SGAS); valGAS =

map(valGAS, 300, 750, 0, 100); int valBt = digitalRead(button);

pinMode(pingPin, OUTPUT);

digitalWrite(pingPin, LOW); delayMicroseconds(2);

digitalWrite(ping Pin, HIGH); delayMicroseconds(5);

digitalWrite(ping Pin, LOW);

pinMode(pingPin, INPUT); duration

= pulseln(pingPin, HIGH); cm =

microsecondsToCentimeters(duratio n); if(cm <336){

digitalWrite(ledUS, HIGH); }else{

digitalWrite(ledUS, LOW);

if(valLight < 890){

digitalWrite(light, HIGH);

}else{

digitalWrite(light, LOW);

if(valPIR == 1){

digitalWrite(buzzer, HIGH);

}else{

digitalWrite(buzzer, LOW); if(valBt ==

1){

digitalWrite(motor motor,

HIGH); }else{

digitalWrite(motor, LOW);

if(valGAS > 20){

digitalWrite(ledGas, HIGH); }else{

digitalWrite(ledGas ledGas,

LOW);

Serial.print(val PIR);

Serial.println();

long microsecondsToCentimeters(long microseconds) {

return microseconds / 29 / 2;

}