ASSIGNMENT-4

DISTANCE DETECTION USING ULTRASONIC SENSOR

Date	28 October 2022
Team ID	PNT2022TMID46387
Student Register Number	820319104007
Student Names	ARCHANA B
Project Name	Personal Assistance for Seniors Who Are Self Reliant
Maximum Marks	2 Marks

Question :

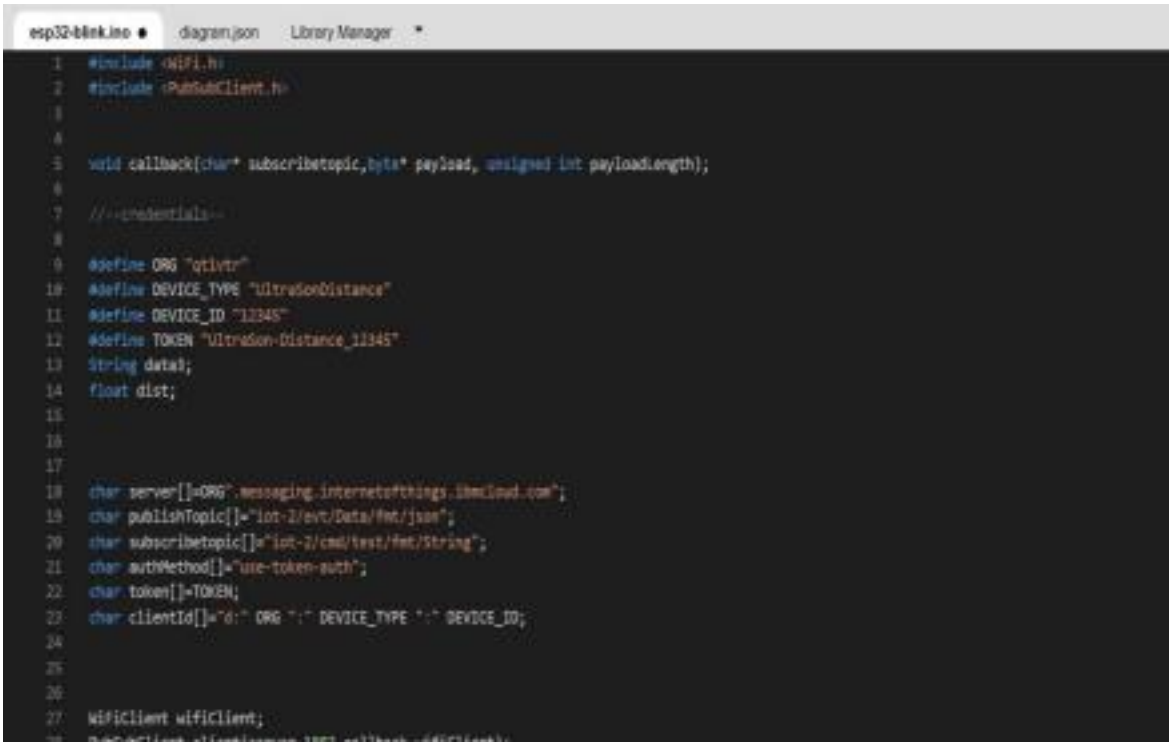
• Write code and connections in wokwi for ultrasonic sensor.

• Whenever distance is less than 100 cms send "alert" to ibm cloud and display in device recent events.

WOKWI LINK :

https://wokwi.com/projects/305566932847821378

CODE :



esp32-blink.ino ● diagram.json Library Manager

```
WiFiClient wifiClient;
PubSubClient client(server,1883,callback,wifiClient);

int LED=4;
int trig=5;
int echo=18;
void setup()
{
Serial.begin(115200);
pinMode(trig, OUTPUT);
pinMode(echo, INPUT);
pinMode(LED, OUTPUT);
delay(10);
wificonnect();
mqttconnect();
}
void loop()
{

  digitalWrite(trig,LOW);
   digitalWrite(trig,HIGH);
   delayMicroseconds(10);
   digitalWrite(trig, LOW);
   float dur=pulseIn(echo,HIGH);
   float dist=(dur*0.034)/2;
   Serial.print("Distance in cm");
   Serial.println(dist);
```

esp32-blink.ino ● diagram.json Library Manager

```
   Serial.println(dist);


   PublishData(dist);
   delay(1000);
   if (!client.loop()){
     mqttconnect();
   }
}



/* dv*/

void PublishData(float dist){
  mqttconnect();



  String object;
  if(dist<100){
    digitalWrite(LED, HIGH);
    Serial.println("Object is Near");
    object="Near";
  }
  else{
    digitalWrite(LED, LOW);
```

esp32-blink.ino ● diagram.json Library Manager

```
    digitalWrite(LED, LOW);
    Serial.println("NO Object Found");
    object="No";
  }
  String payload="{\"distance\":";
  payload += dist;
  payload +="," "\"object\":\"";
  payload += object;
  payload +="\"}";
  Serial.print("Sending Payload:");
  Serial.println(payload);
  if(client.publish(publishTopic,(char*) payload.c_str())){
    Serial.println("Publish Ok");
  }else{
    Serial.println("publish failed");
  }
}
```

esp32-blink.ino ● diagram.json Library Manager

```
}
void mqttconnect(){
  if (!client.connected()){
    Serial.print("Reconnecting client to");
    Serial.println(server);
    while(!!!client.connect(clientId,authMethod,token)) {
      Serial.print(".");
      delay(500);
    }

    initManagedDevice();
    Serial.println();
  }
}
void wificonnect()
{
  Serial.println();
  Serial.print("Connecting To");

  WiFi.begin("Wokwi-GUEST","",6);
  while(WiFi.status() !=WL_CONNECTED){
    delay(500);
    Serial.println(".");
  }
  Serial.println("");
  Serial.print("WiFi CONNECTED");
  Serial.println("IP ADDRESS:");
```

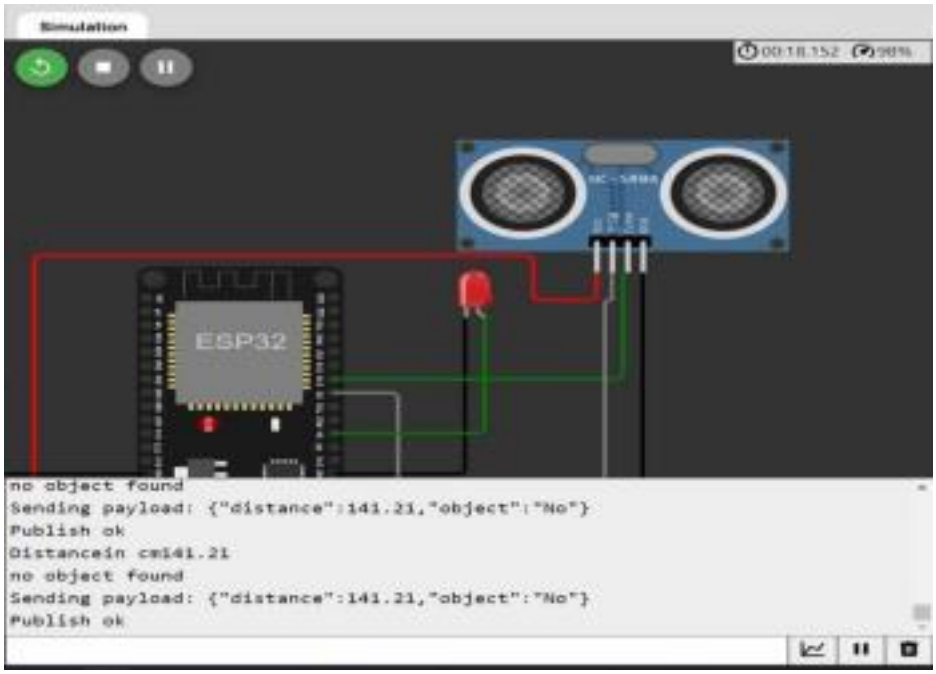
esp32-blink.ino ● diagram.json Library Manager

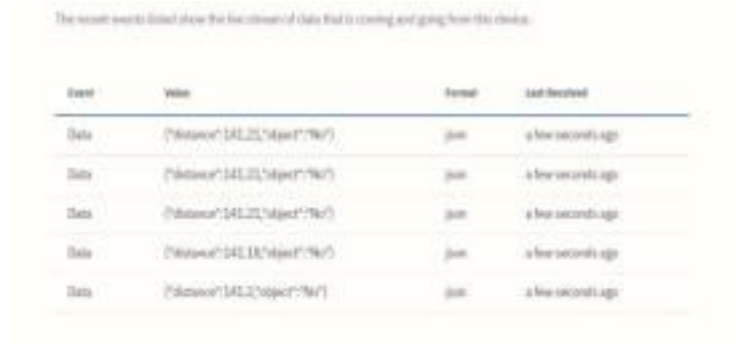
```
void wificonnect()
{
  Serial.println();
  Serial.print("Connecting To");

  WiFi.begin("Wokwi-GUEST","",6);
  while(WiFi.status() !=WL_CONNECTED){
    delay(500);
    Serial.println(".");
  }
  Serial.println("");
  Serial.print("WiFi CONNECTED");
  Serial.println("IP ADDRESS:");
  Serial.println(WiFi.localIP());
}

void initManagedDevice(){
  if(client.subscribe(subscribetopic)){
    Serial.println((subscribetopic));
    Serial.println("SUBSCRIBE TO CMD OK");
  }else{
    Serial.println("SUBSCRIBE TO CMD FAILED");
  }
}

void callback(char* subscribetopic,byte*payload,unsigned int payloadength)
{
```

```
  Serial.println("IP ADDRESS:");
  Serial.println(WiFi.localIP());
}

void initManagedDevice(){
  if(client.subscribe(subscribetopic)){
    Serial.println((subscribetopic));
    Serial.println("SUBSCRIBE TO CMD OK");
  }else{
    Serial.println("SUBSCRIBE TO CMD FAILED");
  }
}

void callback(char* subscribetopic,byte*payload,unsigned int payloadLength)
{

  Serial.print("CALLBACK INVOKED FOR TOPIC:");
  Serial.println(subscribetopic);
  for(int i=0;i<payloadLength;i++){

    data3 +=(char)payload[i];
  }
 data3="";
}
```

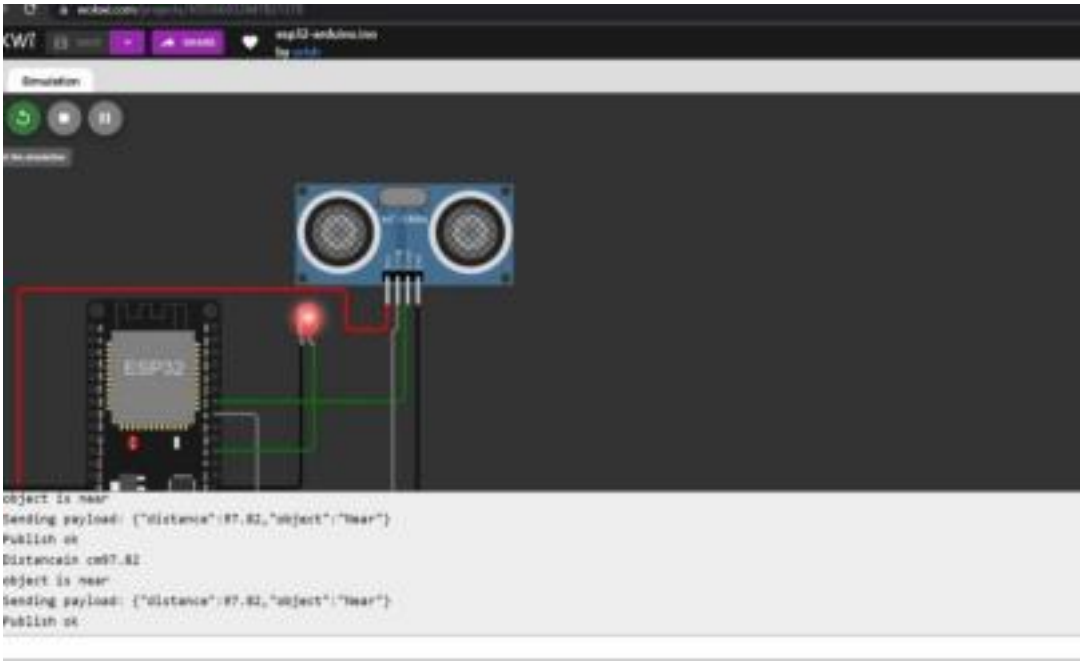
OUTPUT:

OUTPUT ON WOKWI SITE





OUTPUT ON WOKWI SITE



IBM CLOUD DATA GENEREATION