ASSIGNMENT-4

Assignment Date	24 October 22
Student Name	Harish J
Team ID	PNT2022TMID35177
Project Name	Smart Solution for Railways

QUESTION:

Write code and connections in wokwi for ultrasonic sensor. Whenever distance is less than 100 cm, send “alert” to IBM cloud and display in device recent events.

WOKWI SIMULATION LINK:

https://wokwi.com/projects/347429012326318674

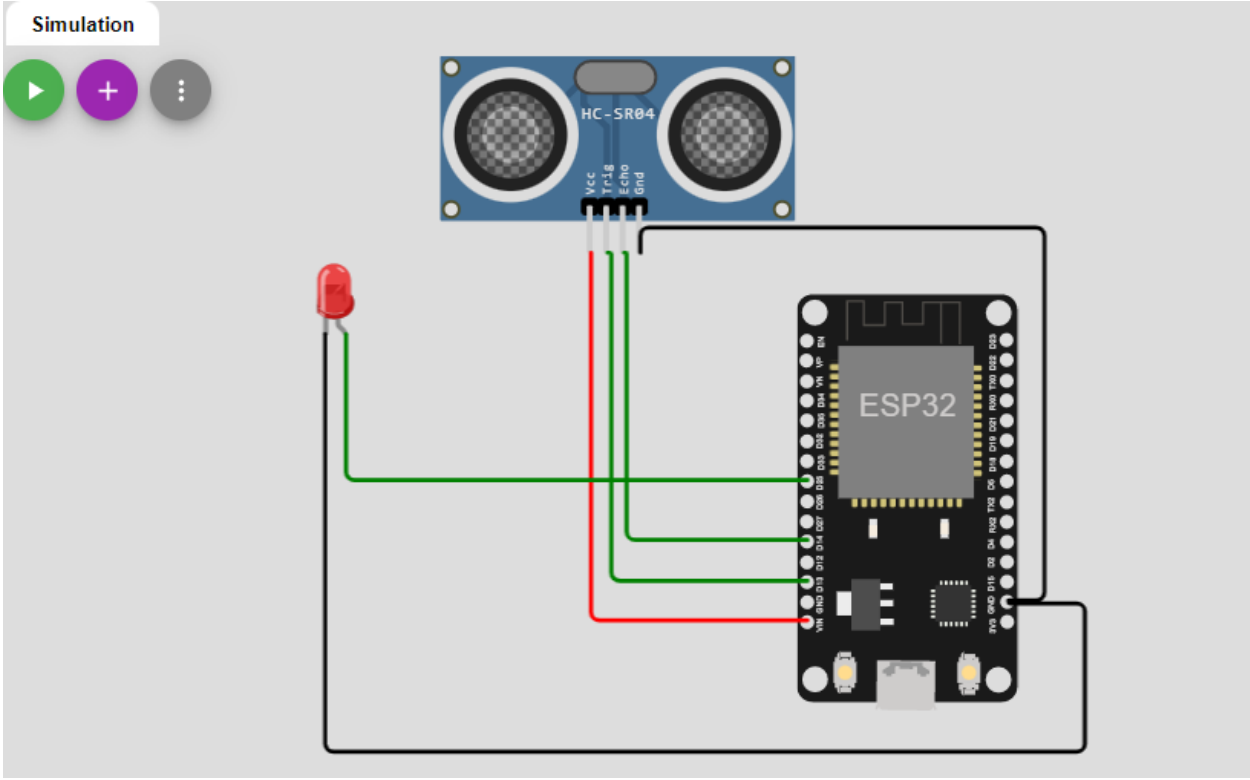
CODE:

```
#include <WiFi.h>
#include <PubSubClient.h>
#define ORG "xzgfzr"
#define DEVICE_TYPE "ESP-Ultrasonic"
#define DEVICE_ID "3524"
#define ECHO_PIN 14
#define TRIG_PIN 13
#define LED 25
#define PORT 1883
char device[] = "d:"ORG":"DEVICE_TYPE":"DEVICE_ID;
char username[] = "use-token-auth";
char password[] = "TWLCe1GCKZg8&O--xn";
char server[] = ORG".messaging.internetofthings.ibmcloud.com";
char publishTopic[]="iot-2/evt/Distance/fmt/json";
char subscribeTopic[] ="iot-2/cmd/Sub/fmt/String";
WiFiClient wifiClient;
PubSubClient client(
  server,
  PORT,
  callback,
  wifiClient
);
```

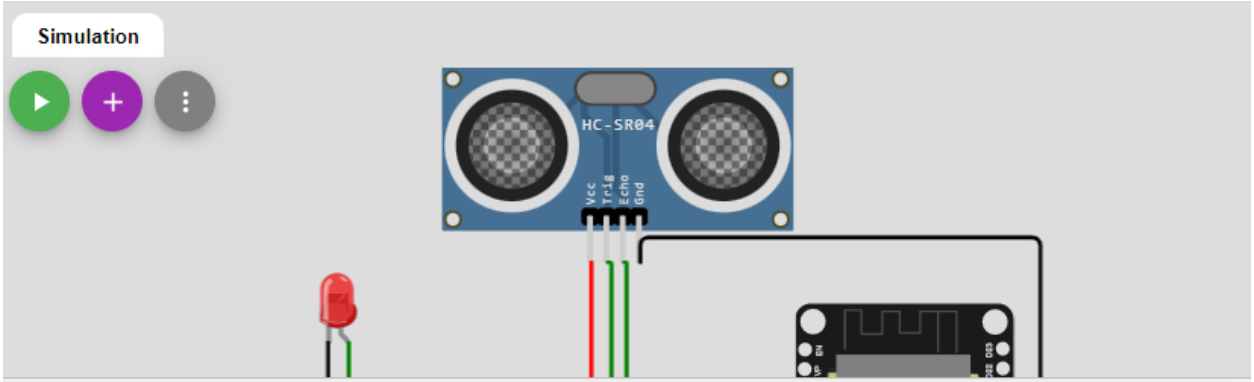
```
void setup() {
  Serial.begin(115200);
  pinMode(TRIG_PIN, OUTPUT);
  pinMode(ECHO_PIN, INPUT);
  pinMode(LED, OUTPUT);
  connectWifi();
  connectMQTT();
}
void loop() {
  float distance = getDistance();
  bool isNearby = distance < 100;
  digitalWrite(LED, isNearby);
  Serial.print("Measured distance: ");
  Serial.println(distance);
  if (distance < 100) {
    Serial.println("DISTANCE < 100 !!!!!");
    publishData(distance);
  }
  delay(5000);
   if(!client.loop())connectMQTT();
}
float getDistance() {
  digitalWrite(TRIG_PIN, LOW);
  delayMicroseconds(2);
  digitalWrite(TRIG_PIN, HIGH);
  delayMicroseconds(10);
  digitalWrite(TRIG_PIN, LOW);
  int duration = pulseIn(ECHO_PIN, HIGH);
  return duration * 0.034 / 2;
}
```

```
void connectWifi(){
  Serial.println();
  Serial.print("Connecting to");
  WiFi.begin("Wokwi-GUEST","",6);
  while(WiFi.status()!=WL_CONNECTED)Serial.print(".");
  Serial.println("");
  Serial.println("WIFI CONNECTED");
  Serial.println("IP address:");
  Serial.println(WiFi.localIP());
}
void connectMQTT(){
 while (!client.connect(device, username, password))Serial.print(".");
 Serial.println("\nConnected to IBM Watson!");
 if (client.subscribe(subscribeTopic))Serial.println("Subscribed to CMD");
 else Serial.println("Subscribe FAILED");
}
void publishData(float distance) {
  if(!client.loop())connectMQTT();
  String payload = "{";
  payload += "\"distance\": ";
  payload += distance;
  payload += "}";
  if (client.publish(publishTopic, (char*)payload.c_str())) {
    Serial.println("Publish OK");
    Serial.println("--------------------");
  }
  else Serial.println("Publish FAILED");
}
void callback(char *subscribeTopic,byte*payload,unsigned int length){
   Serial.println("Callback Invoked!");
    for (int i = 0; i < length; ++i)Serial.print((char)payload[i]);
}
```

CIRCUIT DIAGRAM:



OUTPUT:




```
Connecting to..
WIFI CONNECTED
IP address:
10.10.0.2

Connected to IBM Watson!
Subscribed to CMD
Measured distance: 68.95
DISTANCE < 100 !!!!!
Publish OK
--------------------
Measured distance: 68.99
DISTANCE < 100 !!!!!
Publish OK
```

IBM WATSON IOT PLATFORM: