ASSIGNMENT -4

DISTANCE DETECTION USING ULTRASONIC SENSOR

DATE	**07/11/2022**
TEAM ID	**PNT2022TMID42536**
STUDENT NAME	**Anitha M**
STUDENT ROLL NUMBER	**711019106001**
MAXIMUM MARKS	**2 Marks**

QUESTION:

- Write code and connections in wokwi for the ultrasonic sensor.
- Whenever the distance is less than 100 cms send an "alert" to the IBM cloud and display in the device recent events.
- Upload document with wokwi share link and images of IBM cloud

WOKWI LINK:

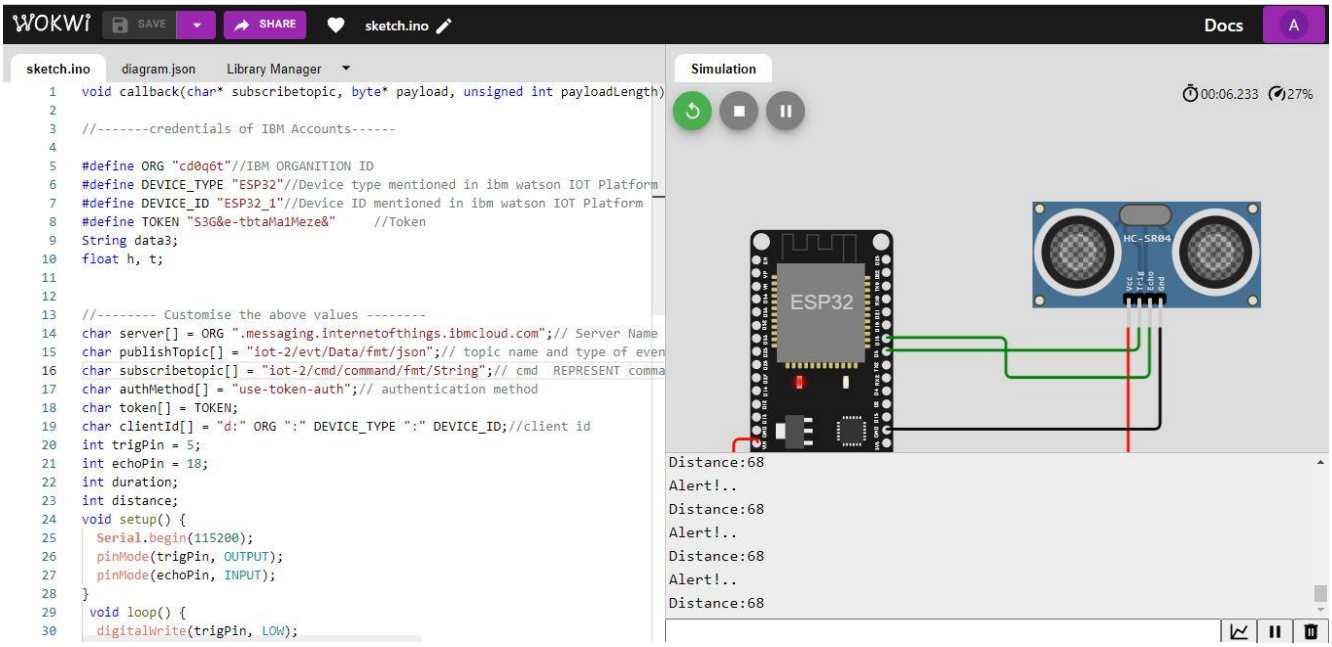
https://wokwi.com/projects/347670701615874642

CODE:

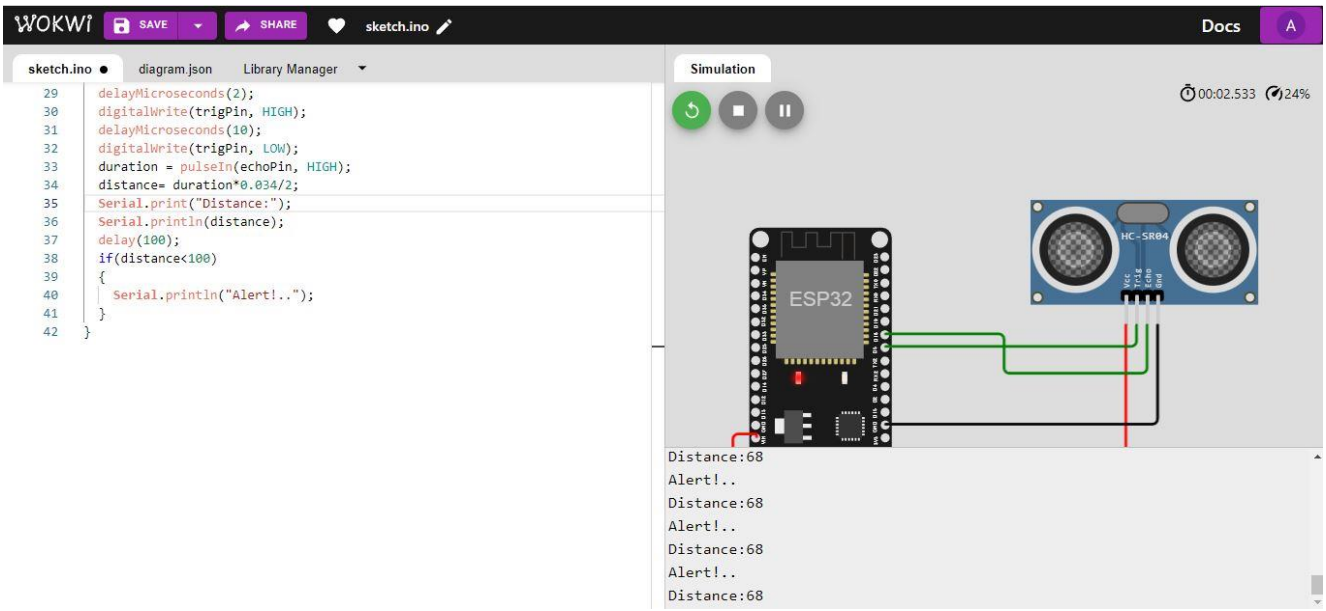
```
void callback(char* subscribetopic, byte* payload, unsigned int payloadLength);

//-------credentials of IBM Accounts------

#define ORG "cd0q6t"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "ESP32_1"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "S3G&e-tbtaMa1Meze&"     //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of event
perform and format in which data to be send
```

```
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT command
type AND COMMAND IS TEST OF FORMAT STRING
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
int trigPin = 5;
int echoPin = 18;
int duration;
int distance;
void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}
 void loop() {
  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance= duration*0.034/2;
  Serial.println(“Distance:”);
  Serial.println(distance);
  delay(100);
  if(distance<100)
  {
    Serial.println("Alert!..");
  }
}
```

DESIGN:




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

//-------credentials of IBM Accounts------

#define ORG "cd0q6t"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT Platform
#define DEVICE_ID "ESP32_1"//Device ID mentioned in ibm watson IOT Platform
#define TOKEN "S3G&e-tbtaMa1Meze&"      //Token
String data3;
float h, t;


//-------- Customise the above values --------
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";// Server Name
char publishTopic[] = "iot-2/evt/Data/fmt/json";// topic name and type of even
char subscribetopic[] = "iot-2/cmd/command/fmt/String";// cmd  REPRESENT comma
char authMethod[] = "use-token-auth";// authentication method
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;//client id
int trigPin = 5;
int echoPin = 18;
int duration;
int distance;
void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
}
 void loop() {
  digitalWrite(trigPin, LOW);
```




```
  delayMicroseconds(2);
  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);
  duration = pulseIn(echoPin, HIGH);
  distance= duration*0.034/2;
  Serial.print("Distance:");
  Serial.println(distance);
  delay(100);
  if(distance<100)
  {
    Serial.println("Alert!..");
  }
}
```

IMAGES OF IBM CLOUD: