ASSIGNMENT-04

Date	**27 October 2022**
Team ID	**PNT2022TMID42308**
Project Name	**Project -**Real time river water quality monitoring and Control System
Maximum Marks	**4 Marks**

Project Title**:** Real Time River water quality monitoring and Control system

Team ID: PNT2022TMID42308

Team Members:

- **Vikash V V - Team Leader**
- **Sharmila R -Team Member**
- **Ishwarya J - Team Member**
- **Jeevan Kumar M – Team Member**

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 devicerecent events.

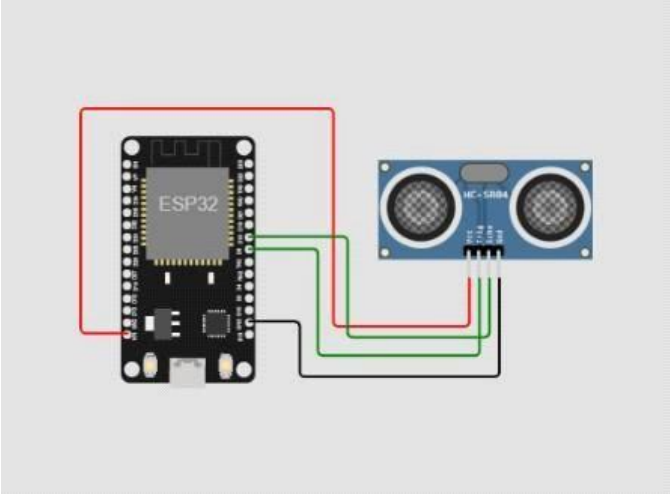
CODE:

```
#include
<WiFi.h>
#include
<PubSubClient.h>
void    callback(char*    subscribetopic,    byte*    payload,
        unsigned          intpayloadLength);
//-------credentials of IBM Accounts------
#define ORG "Ashfaq1824"//IBM ORGANITION ID
#define DEVICE_TYPE "ESP32"//Device type mentioned in ibm watson IOT
Platform#define DEVICE_ID "12345"//Device ID mentioned in ibm watson IOT
Platform #define TOKEN "12345678" //Token
String data3;
char                server[]                =                ORG
".messaging.internetofthings.ibmcloud.com";                charpublishTopic[] = "iot-2/evt/Data/fmt/json"; char subs
char token[] = TOKEN;
char clientId[] = "d:" ORG ":" DEVICE_TYPE ":" DEVICE_ID;
WiFiClient wifiClient;
```

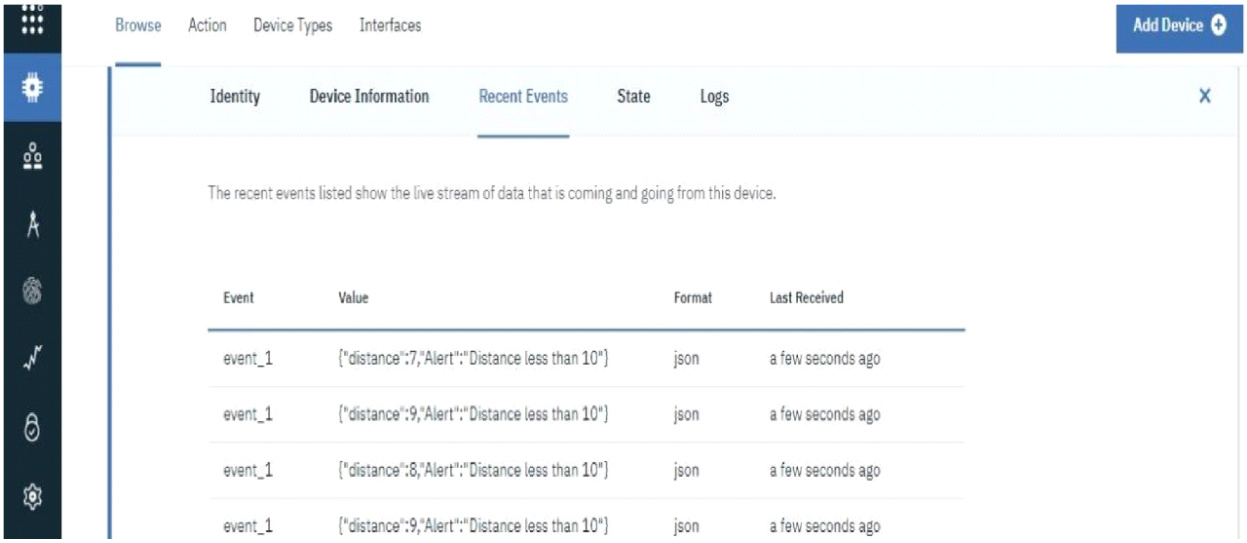
```
PubSubClient  client(server, 1883, callback
,wifiClient);const int trigPin = 5;
const int echoPin = 18;

#define         SOUND_SPEED
0.034   long    duration;   floatdistance;        void     setup() {Serial.begin(115200); pinMode(trigPin, OUTPUT);p
}       void        loop()       {digitalWrite(trigPin,              LOW);delayMicroseconds(2); digitalWrite(trigPin,
Serial.print("Distance
(cm): ");
Serial.println(distance);
if(distance<100)
{
Serial.println("ALERT!!"
); delay(1000);
PublishData(dist
ance);
delay(1000); if
(!client.loop()) {
mqttconnect();
} }
delay(1000
); }
void       PublishData(float       dist)       {mqttconnect();
String    payload    =
          "{\"Distance\":";payload
+= dist;
payload += ",\"ALERT!!\":""\"Distance less than
100cms\"";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");
}   }   void   mqttconnect()   {   if
(!client.connected())                    {Serial.print("Reconnecting client to");
Serial.println(server);
while (!!!client.connect(clientId, authMethod, token)) {
Serial.prin
t(".");
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.print(".");
}
Serial.println("");
Serial.println("WiFi
connected");
Serial.println("IP address:
");
Serial.println(WiFi.localIP(
));
}
void initManagedDevice() { if (client.subscribe(subscribetopic)) { Serial.println((subscribetopic)); Serial.print
} else {
Serial.println("subscribe to cmd FAILED");
} }
void callback(char* subscribetopic, byte* payload,
unsigned intpayloadLength) {
Serial.print("callback invoked for
topic: ");
Serial.println(subscribetopic); for
(int i = 0; i < payloadLength; i++)
{
//Serial.print((char)payload[i]);
data3 += (char)payload[i];
}
Serial.println("data: "+
data3);
data3="";
}
```

SCHEMATIC/CIRCUIT DIAGRAM:



IBM CLOUD OUTPUT:



WOKWI LINK:
https://wokwi.com/projects/346419220039336530