Real-Time River Water Quality Monitoring and Control Systems

Assignment - 4

submitted by: ***DHEVAK P***

312319106037

Question-1:

Write a code and connections in wokwi for the ultrasonic sensor. Whenever the distance is less than 100cms send an "Alert" to IBM cloud and display in the device recent events.

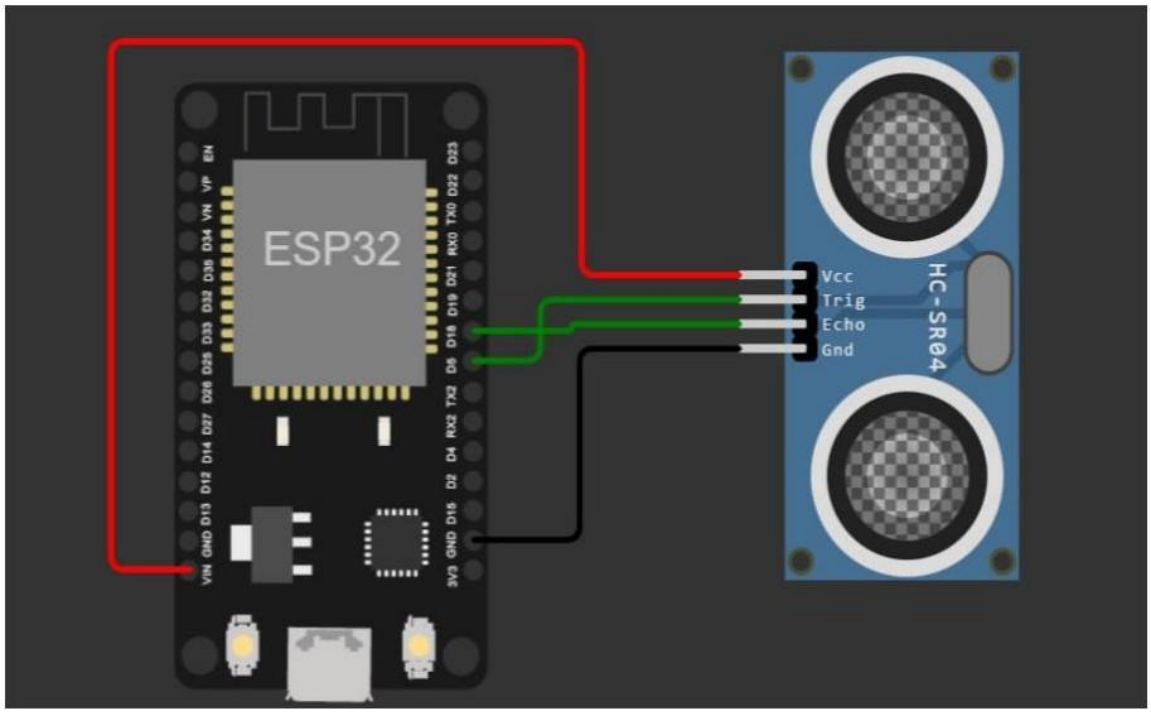
Code:

```
#i
nc
lu
de
<W
iF
i.
h>
#i
nc
lu
de
<P
ub
Su
bC
li
en
t.
h>
void callback(char* subscribetopic, byte* payload,
```

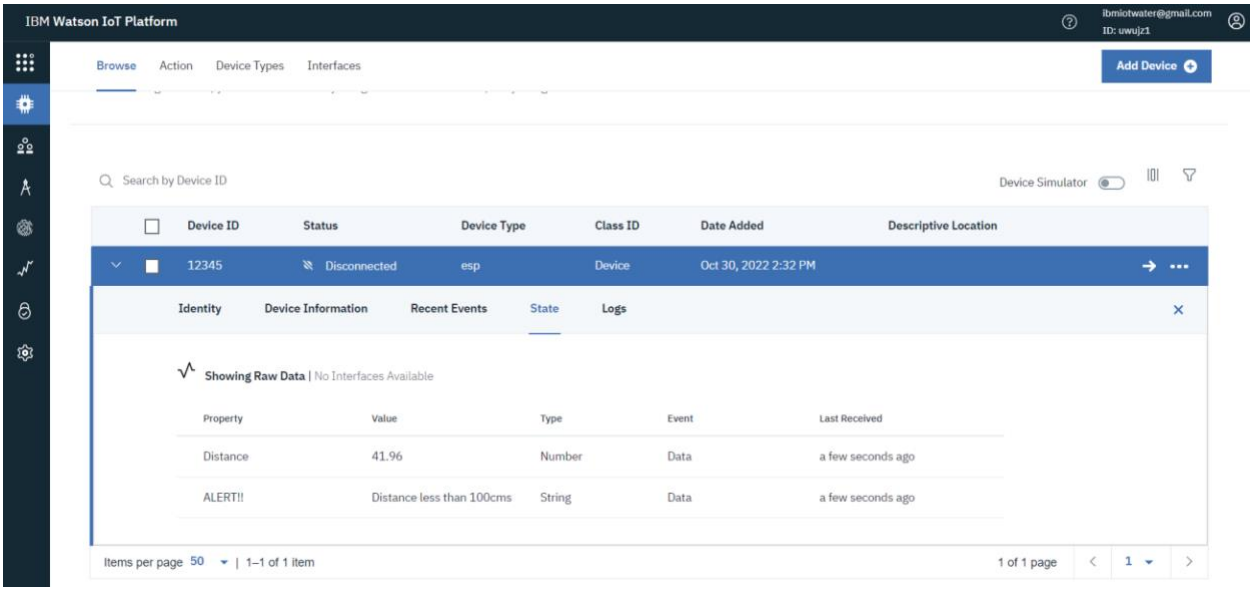
```
mqttconnect();
}
void loop()
{
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = duration * SOUND_SPEED/2;
Serial.print("Distance (cm): ");
Serial.println(distance);
if(distance<100)
{
Serial.println("ALERT!!");
delay(1000);
PublishData(distance);
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.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.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.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++) {
//Serial.print((char)payload[i]);
data3 += (char)payload[i];
}
Serial.println("data: "+ data3);
data3="";
}
```

Circuit Diagram:



IBM Cloud Output:



Message	Timestamp
Closed connection. The client ID was reused.	Oct 30, 2022 2:45 PM
Closed connection. The client ID was reused.	Oct 30, 2022 2:45 PM
Token auth succeeded: ClientID='d:uwujz1:esp:...	Oct 30, 2022 2:45 PM
Token auth succeeded: ClientID='d:uwujz1:esp:...	Oct 30, 2022 2:45 PM
Closed connection. The connection timed out	Oct 30, 2022 2:43 PM
Token auth succeeded: ClientID='d:uwujz1:esp:...	Oct 30, 2022 2:42 PM
Token auth succeeded: ClientID='d:uwujz1:esp:...	Oct 30, 2022 2:41 PM

Output:

```
Connecting to ...
WiFi connected
IP address:
10.10.0.2
Reconnecting client to uwujz1.messaging.internetofthings.ibmcloud.com
iot-2/cmd/test/fmt/String
subscribe to cmd OK

Distance (cm): 41.97
ALERT!!
Sending payload: {"Distance":41.97,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 41.96
ALERT!!
Sending payload: {"Distance":41.96,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 41.96
ALERT!!
Sending payload: {"Distance":41.96,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 41.96
ALERT!!
```