

ASSIGNMENT - 4

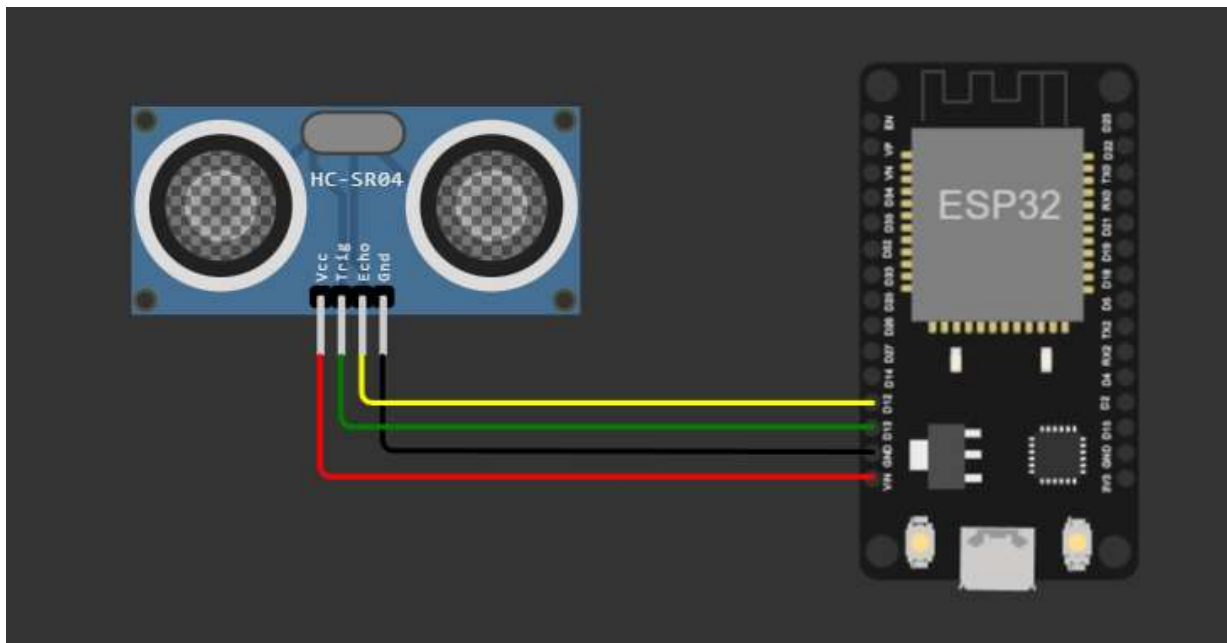
Date	19 September 2022
Team ID	PNT2022TMID21342
Project Name	SmartFarmer - IoT Enabled Smart Farming Application
Maximum Marks	2 Marks

Objective:

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.

Circuit Diagram:

Link: <https://wokwi.com/projects/346824939389583954>



OUTPUT:

WOKWI

Sketch: 100% 100% 100%

Serial: 115200

```
1 #include <ESP32.h> //library for esp32
2 #include <PubSubClient.h> //library for mqtt
3 #define PIN_POW 15
4 #define PIN_ECHO 17
5
6
7
8 void callback(char* topic, byte* payload, unsigned int payloadLength) {
9   //... your code to process the message ...
10
11   //... publish the message ...
12   const char* msg = "Distance: 67.95,MESSAGE: 'ALERT'";
13   const char* topic = "test/esp32/mqtt";
14   const char* device_id = "ESP32";
15   const char* token = "token";
16
17   //... configure the mqtt client ...
18   const char* url = "mqtt://test:token@192.168.1.1:1883";
19   PubSubClient client(url, 1883, callback, device_id, token);
20
21   //... connect to the mqtt server ...
22   client.connect();
23   if (client.connected()) {
24     //... publish the message ...
25     client.publish(topic, msg);
26   }
27 }
28
29 void setup() {
30   //... configure the pins ...
31   pinMode(PIN_POW, OUTPUT);
32   pinMode(PIN_ECHO, OUTPUT);
33   digitalWrite(PIN_POW, LOW);
34   digitalWrite(PIN_ECHO, LOW);
35 }
36
37 void loop() {
38   //... read the sensor ...
39   int distance = 0;
40   while (true) {
41     distance = ultrasonic.read();
42     if (distance > 0) {
43       //... call the callback function ...
44       callback(topic, payload, payloadLength);
45     }
46   }
47 }
```

Simulation

ESP32

Ultrasonic sensor

Distance: 67.95 cm

MQTT

Distance: 67.95,MESSAGE: 'ALERT'

Distance: 67.95,MESSAGE: 'ALERT'

Distance: 67.95,MESSAGE: 'ALERT'

Distance: 67.95,MESSAGE: 'ALERT'

Distance: 67.95,MESSAGE: 'ALERT'

WOKWI

Sketch: 100% 100% 100%

Serial: 115200

```
1 #include <ESP32.h> //library for esp32
2 #include <PubSubClient.h> //library for mqtt
3 #define PIN_POW 15
4 #define PIN_ECHO 17
5
6
7
8 void callback(char* topic, byte* payload, unsigned int payloadLength) {
9   //... your code to process the message ...
10
11   //... publish the message ...
12   const char* msg = "Distance: 146.95,MESSAGE: 'SAFE'";
13   const char* topic = "test/esp32/mqtt";
14   const char* device_id = "ESP32";
15   const char* token = "token";
16
17   //... configure the mqtt client ...
18   const char* url = "mqtt://test:token@192.168.1.1:1883";
19   PubSubClient client(url, 1883, callback, device_id, token);
20
21   //... connect to the mqtt server ...
22   client.connect();
23   if (client.connected()) {
24     //... publish the message ...
25     client.publish(topic, msg);
26   }
27 }
28
29 void setup() {
30   //... configure the pins ...
31   pinMode(PIN_POW, OUTPUT);
32   pinMode(PIN_ECHO, OUTPUT);
33   digitalWrite(PIN_POW, LOW);
34   digitalWrite(PIN_ECHO, LOW);
35 }
36
37 void loop() {
38   //... read the sensor ...
39   int distance = 0;
40   while (true) {
41     distance = ultrasonic.read();
42     if (distance > 0) {
43       //... call the callback function ...
44       callback(topic, payload, payloadLength);
45     }
46   }
47 }
```

Simulation

ESP32

Ultrasonic sensor

Distance: 146.95 cm

MQTT

Distance: 146.95,MESSAGE: 'SAFE'

Distance: 146.95,MESSAGE: 'SAFE'

Distance: 146.95,MESSAGE: 'SAFE'

Distance: 146.95,MESSAGE: 'SAFE'

Distance: 146.95,MESSAGE: 'SAFE'

