Ultrasonic sensor simulation in Wokwi

TEAM ID	NM2023TMID10876
PROJECT NAME	Industrial Workers Health and Safety System Based on Internet of Things
Date	12 MAY 2023

Problem statement:

Build wowki product, use ultrasonic sensor and detect the distance from the object. Whenever distance is less than 100cms upload the value to the ibm cloud.in recent device events upload the data from wokwi.

Wokwi link: https://wokwi.com/projects/364770802247914497

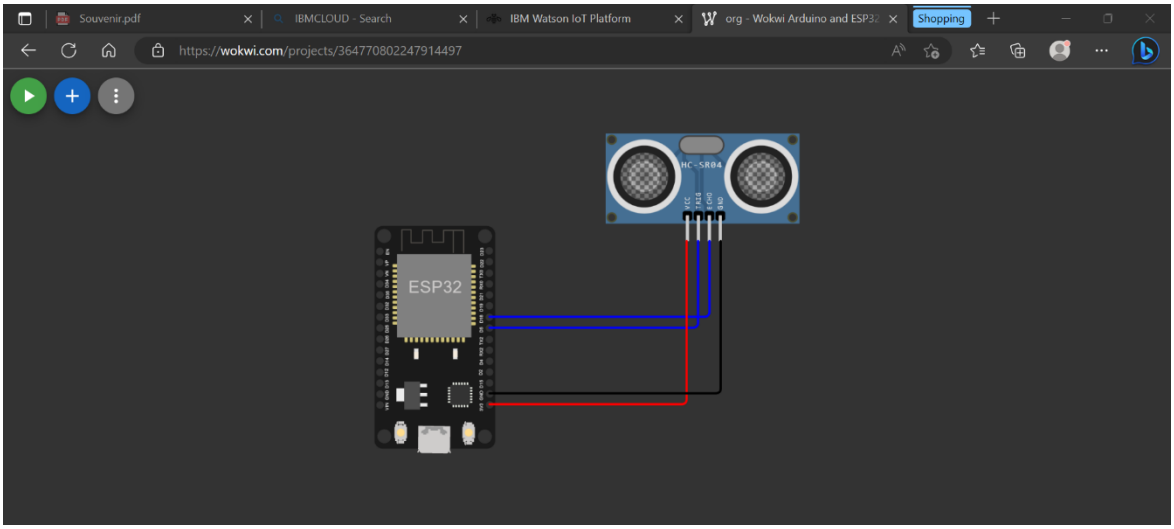
Code:

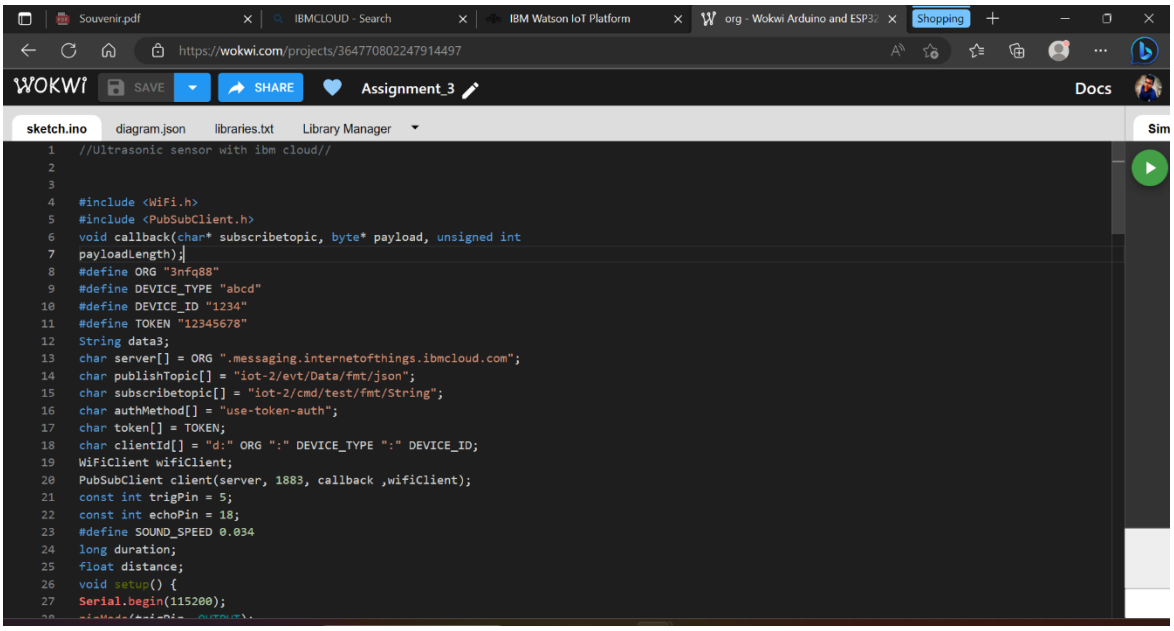
```
//Ultrasonic sensor with ibm cloud//

#include <WiFi.h>
#include <PubSubClient.h>
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength);
#define ORG "3nfq88"
#define DEVICE_TYPE "abcd"
#define DEVICE_ID "1234"
#define TOKEN "12345678"
String data3;
char server[] = ORG ".messaging.internetofthings.ibmcloud.com";
char publishTopic[] = "iot-2/evt/Data/fmt/json";
char subscribetopic[] = "iot-2/cmd/test/fmt/String";
char authMethod[] = "use-token-auth";
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;
float distance;
void setup() {
Serial.begin(115200);
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
wificonnect();
mqttconnect();
}
void loop()
{
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
digitalWrite(trigPin, HIGH);
delayMicroseconds(10);
digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
```

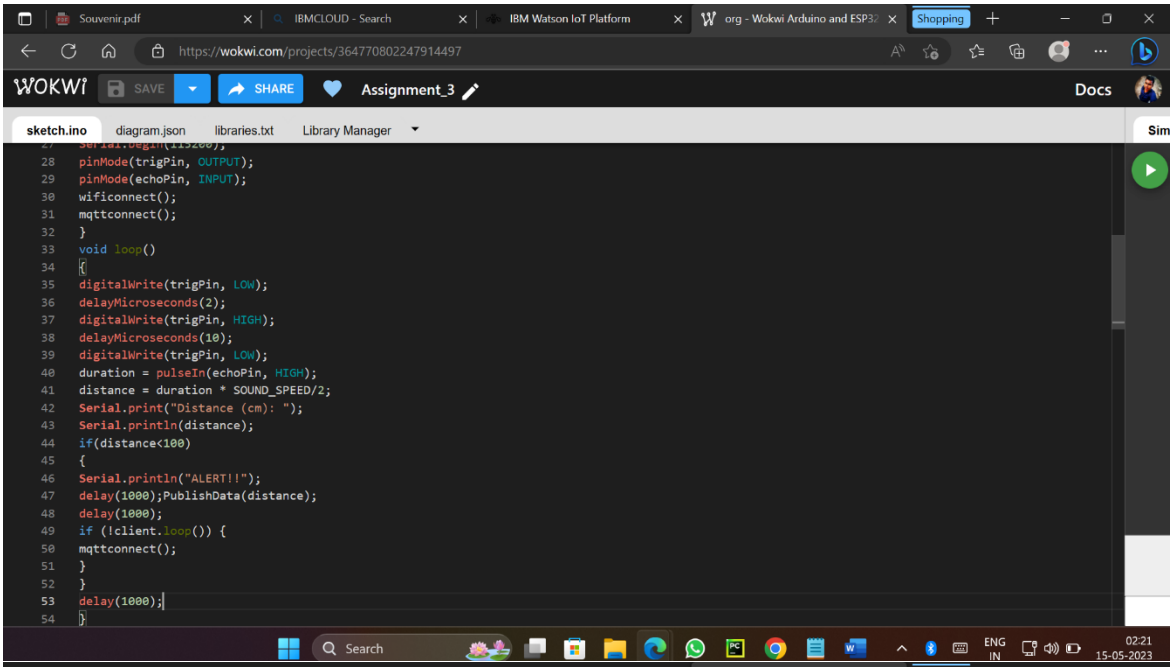
```
distance = duration * SOUND_SPEED/2;
Serial.print("Distance (cm): ");
Serial.println(distance);
if(distance<100)
{
Serial.println("ALERT!!");
delay(1000);PublishData(distance);
delay(1000);
if (!client.loop()) {
mqttconnect();
}
}
delay(1000);
}
void PublishData(float dist) {
mqttconnect();
String 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 {
```

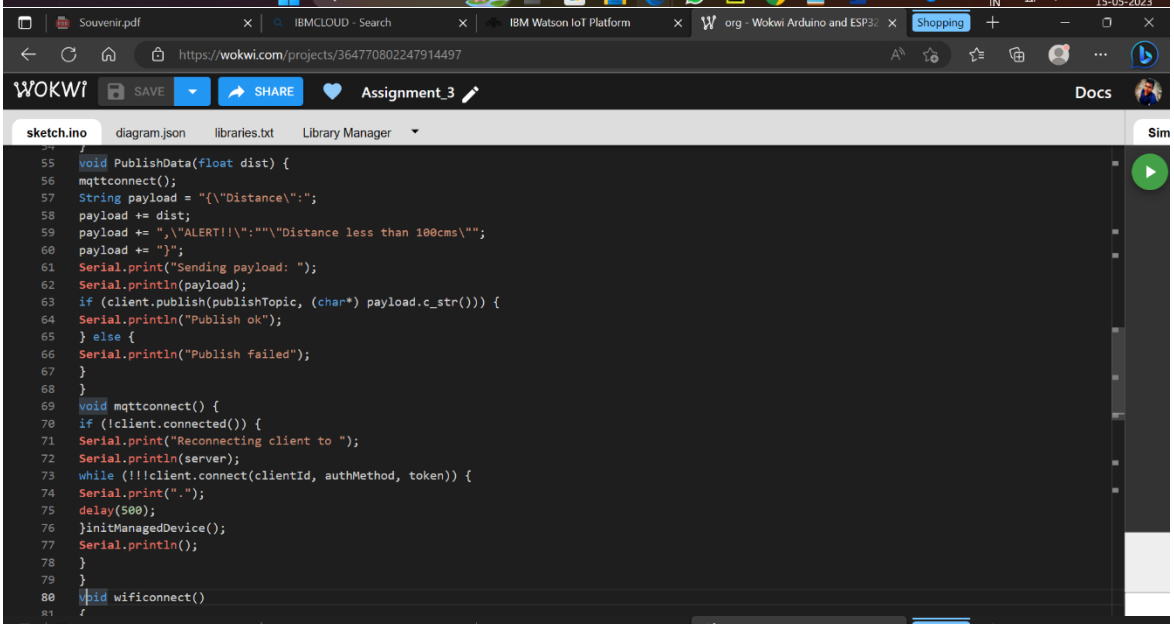
```
Serial.println("subscribe to cmd FAILED");
}
}
void callback(char* subscribetopic, byte* payload, unsigned int
payloadLength)
{
Serial.print("callback invoked for topic: ");
Serial.println(subscribetopic);
for (int i = 0; i < payloadLength; i++)
{//Serial.print((char)payload[i]);
data3 += (char)payload[i];
}
Serial.println("data: "+ data3);
data3="";
}
```

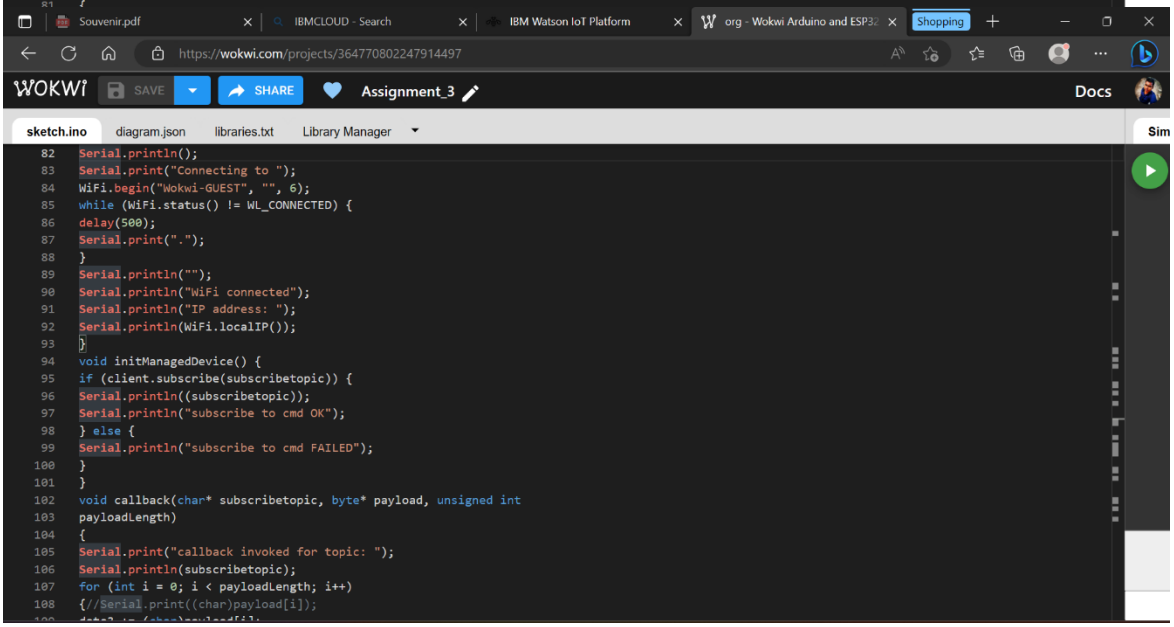
Schematics :

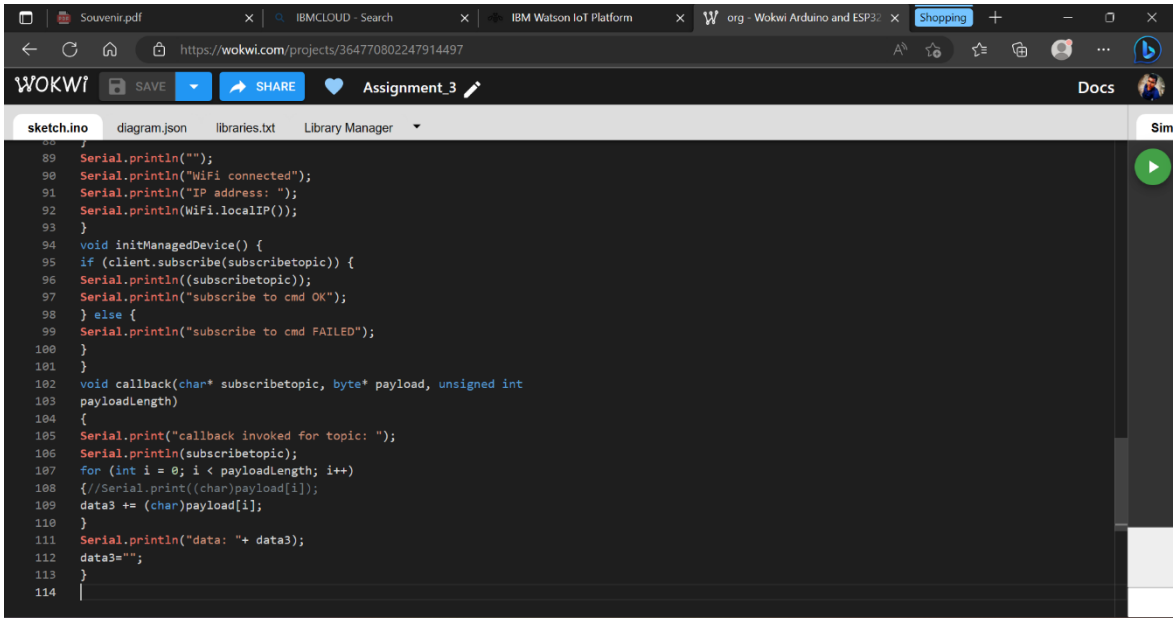


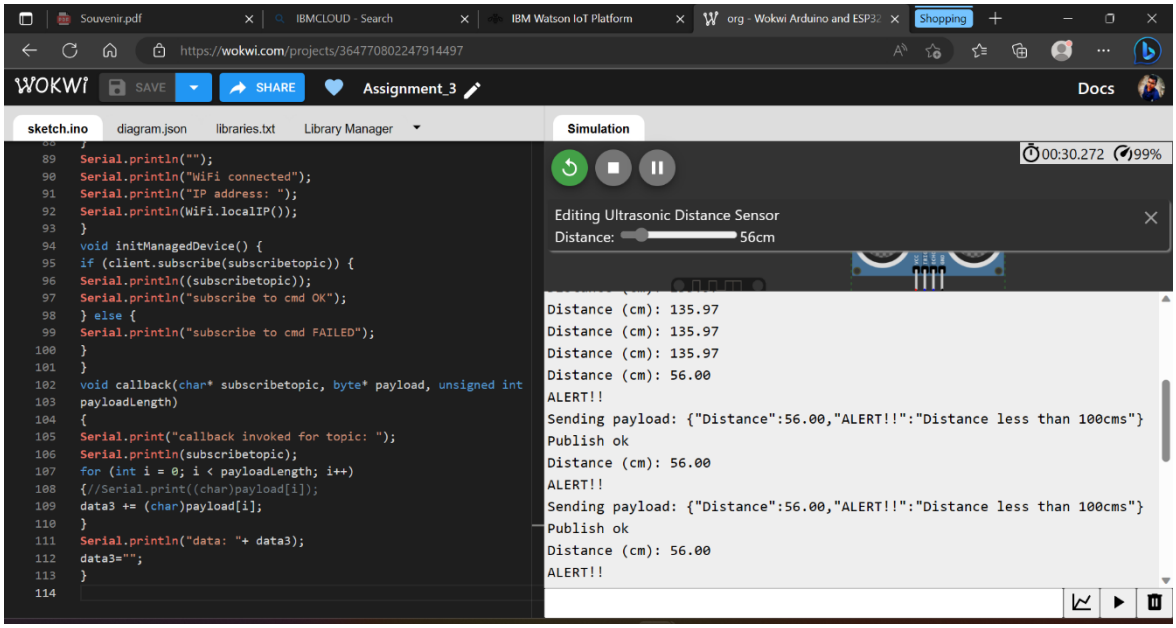


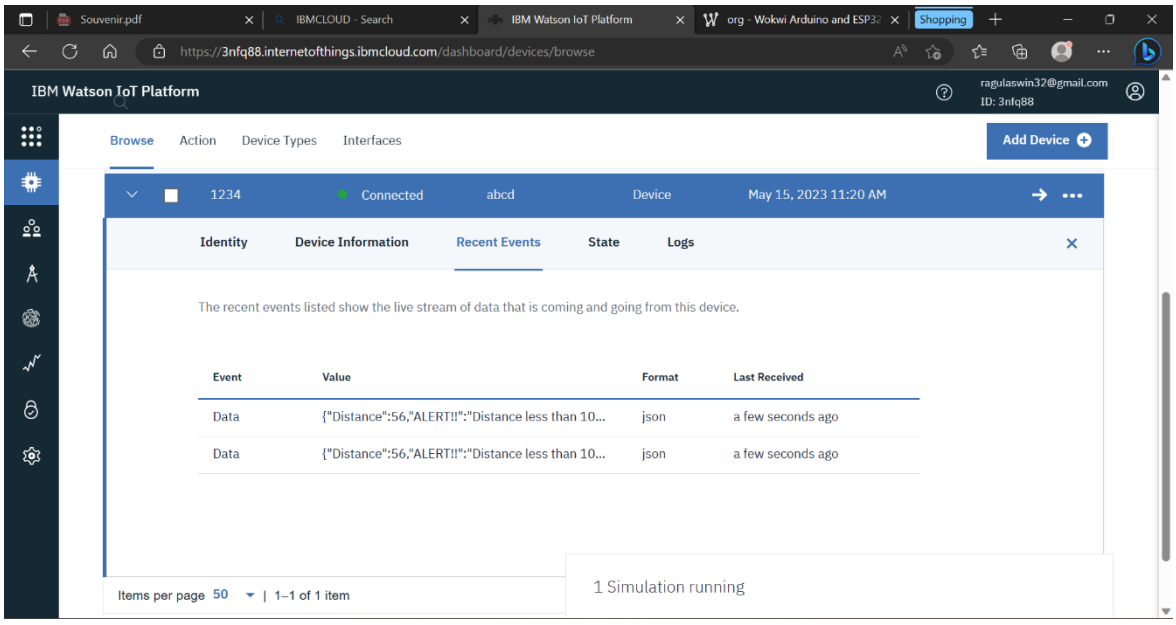

Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://wokwi.com/projects/364770802247914497
WOKWI
SAVE
SHARE
Assignment_3
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
Sim
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
wificonnect();
mqttconnect();
}
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);
}
Search
ENG
IN
02:21
15-05-2023


Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://wokwi.com/projects/364770802247914497
WOKWI
SAVE
SHARE
Assignment_3
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
Sim
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()


Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://wokwi.com/projects/364770802247914497
WOKWI
SAVE
SHARE
Assignment_3
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
Sim
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]);


Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://wokwi.com/projects/364770802247914497
WOKWI
SAVE
SHARE
Assignment_3
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
Sim
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="";
}


Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://wokwi.com/projects/364770802247914497
WOKWI
SAVE
SHARE
Assignment_3
Docs
sketch.ino
diagram.json
libraries.txt
Library Manager
Simulation
00:30.272
99%
Editing Ultrasonic Distance Sensor
Distance:
56cm
Distance (cm): 135.97
Distance (cm): 135.97
Distance (cm): 135.97
Distance (cm): 56.00
ALERT!!
Sending payload: {"Distance":56.00,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 56.00
ALERT!!
Sending payload: {"Distance":56.00,"ALERT!!":"Distance less than 100cms"}
Publish ok
Distance (cm): 56.00
ALERT!!


Souvenir.pdf
IBMCLOUD - Search
IBM Watson IoT Platform
org - Wokwi Arduino and ESP32
Shopping
https://3nfq88.internetofthings.ibmcloud.com/dashboard/devices/browse
IBM Watson IoT Platform
ragulaswin32@gmail.com
ID: 3nfq88
Browse
Action
Device Types
Interfaces
Add Device
1234
Connected
abcd
Device
May 15, 2023 11:20 AM
Identity
Device Information
Recent Events
State
Logs
The recent events listed show the live stream of data that is coming and going from this device.
Event
Value
Format
Last Received
Data
{"Distance":56,"ALERT!!":"Distance less than 10...
json
a few seconds ago
Data
{"Distance":56,"ALERT!!":"Distance less than 10...
json
a few seconds ago
Items per page 50
1–1 of 1 item
1 Simulation running