DECEMBER 4, 2021

TEAM : LOML MXD

PROJECT REPORT

TEAM MEMBERS :->

TIRTH MOTKA(2020101036)

JAY GHEVARIYA(2020101070)

PRIYANSH(2020101056)

ABHINAV ANAND(2020101054)

Motivation and idea

SMART ALARM CLOCK

The main motivation for this project was that we always use alarm clocks especially students having exams and deadlines . But we don't want to get up and stop it or we don't want to do any work to snooze it . So we thought we might be helpful to those people wanting to switch off the alarm or snooze it from the same place they are without touching the alarm clock .

The main idea now to develop such alarm clock is to use our hand and the distance between hand and alarm clock to snooze or stop the clock .

WORKING

Point by point —>

1. At first, we record the distance between the nearest thing to the alarm clock in the direction of USS(ultrasonic sensor) . In the case for stoping or snoozing there would be hand as the nearest thing .
2. Then we check that distance with our predefined constraints for it being in the range for stopping or snoozing or not doing anything .
3. We check for that distance and make our alarm clock work accordingly .
4. We can control the snoozing time or constraint distance or range .

COMPONENTS USED

1. ESP32
2. Breadboard
3. Buzzer
4. Ultrasonic sensor
5. Jumper wires (both male-male, male-female)
6. LED lights
7. Arduino software

8. OneM2M software

9. Serial bluetooth app

CODE

```
#include "BluetoothSerial.h"

#include "PubSubClient.h"

#include "WiFi.h"

#include "ThingSpeak.h"

const int trigPin = 12;

const int echoPin = 13;

const char* ssid = "iQOO";

const char* password = "12991300";

char* topic = "channels/1581431/publish/3C6LR8M0BMZEL61A";

const char* server = "mqtt.thingspeak.com";

int hh=0;



int check = 0;

int thre=0;

WiFiClient wifiClient;
```

```
PubSubClient client(server, 1883, wifiClient);


void callback(char* topic, byte* payload, unsigned int length) {
//nothing there
}


BluetoothSerial SerialBT;
byte BTData;


void setup() {
  Serial.begin(115200);
  pinMode(trigPin, OUTPUT);
  pinMode(echoPin, INPUT);
   pinMode(27, OUTPUT);
  digitalWrite(27,LOW);


  delay(10);
  Serial.println();
  Serial.print("Connecting to ");

  Serial.print("ssid ");



  WiFi.begin(ssid, password);



  while (WiFi.status() != WL_CONNECTED) {

    delay(1000);

    Serial.print(".");

  }

  Serial.println("");

  Serial.println("WiFi connected");

  Serial.println("IP address: ");

  Serial.println(WiFi.localIP());



  String clientName = "ESP-Thingspeak";

  Serial.print("Connecting to ");

  Serial.print(server);

  Serial.print(" as ");
  Serial.println(clientName);


  if (client.connect((char*) clientName.c_str())) {

    Serial.println("Connected to MQTT broker");

    Serial.print("Topic is: ");

    Serial.println(topic);


    if (client.publish(topic, "hello from ESP8266")) {

      Serial.println("Publish ok");

    }

    else {

      Serial.println("Publish failed");

    }

  }

  else {

    Serial.println("MQTT connect failed");

    Serial.println("Will reset and try again...");

    abort();
  }
}

void loop() {

  digitalWrite(trigPin, LOW);
  delayMicroseconds(2);

  digitalWrite(trigPin, HIGH);
  delayMicroseconds(10);
  digitalWrite(trigPin, LOW);


  float duration = pulseIn(echoPin, HIGH);


  float distanceCm = (duration/2) * 0.034;
float distanceInch = distanceCm / 2.5;


 Serial.print("Distance (cm): ");

 Serial.println(distanceCm);


 if(distanceCm>50 && hh == 0){

   digitalWrite(27,HIGH);

 }


 if(distanceCm<=50 && distanceCm>=10 && hh == 0){

   digitalWrite(27,LOW);

   delay(5000);

   digitalWrite(27,HIGH);


 }
```

```
if(distanceCm<10){

  digitalWrite(27,LOW);

  hh = 1;



}



String payload="field1=";

payload+=distanceCm;

payload+="&status=MQTTPUBLISH";



if (client.connected()){

  Serial.print("Sending payload: ");

  Serial.println(payload);



  if (client.publish(topic, (char*) payload.c_str())) {

    Serial.println("Publish ok");

  }
```

```
    else {
      Serial.println("Publish failed");
    }
  }
  delay(1000);
}
```