ASSIGNMENT-1

SOLAMALAI COLLEGE OF ENGINEERING

NAME : BRAHADEESHWARAN M.B.S

CLASS : III year CSE

SUBJECT : SB8040 - Professional Readines

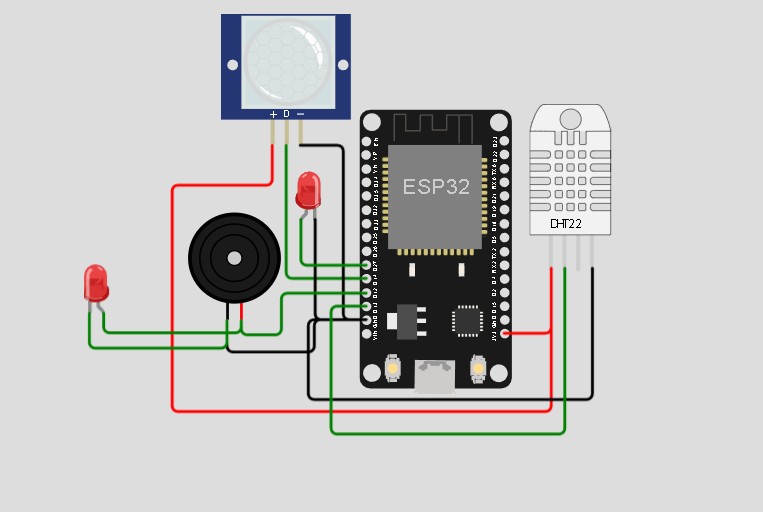
For Innovation, Employability And

Entrepreneurship.

REGISTER NO : 912220104303

Build a smart home in wokwi with minimum 2sensors,Led, buzzer

DESIGN PART



CODINGPART

sketch.ino

#include <FirebaseESP32.h> //call library firebase esp32#include<WiFi.h>//calllibrarywifi

#include"DHT.h"//calllibrarydht

#define FIREBASE\_HOST "https://smart-home-7b44f-default-rtdb.firebaseio.com" //declare variable as a firebase host#define WIFI\_SSID "Wokwi-GUEST" //declare variable as awifi\_ssid

#defineWIFI\_PASSWORD""//declarevariableasawifipassword

#define FIREBASE\_Authorization\_key"oTRPje2my9ROlP2dIhJMzEd66sd3arqmtC4WG6cv"

//declarevariableasafirebaseauthkey

FirebaseData firebaseData; //Define the FirebaseData object asfirebaseData

FirebaseJsonjson;//DefinetheFirebaseJsonasjson

intled= 27; //declare variableasa integer

String message = ""; //declare variable as a stringStringmessage1="";//declarevariableasastringString s\_pir = ""; //declare variable as a stringString b= "";//declarevaribleasa string

String c = ""; //declare variable as a stringbool a = 0; //declare variable as a boolean#definepir14//declarevariableasa pin pir

#define buzzer 12 //declare variable as a pin buzzer#define DHTPIN 13 //declare variable as a pin dht#defineDHTTYPEDHT22//declarevariableasadht11DHTdht(DHTPIN,DHTTYPE);//connectesp32todht

voidsetup(){

pinMode(led, OUTPUT); //led as an outputpinMode(pir, INPUT); //pir as an inputpinMode(buzzer,OUTPUT);//buzzerasanoutputSerial.begin(115200); //open serial connectiondht.begin();//initiatetheconnection withthedht

WiFi.begin(WIFI\_SSID,WIFI\_PASSWORD);//initiatetheconnectionwith ap

Serial.println("Connecting...");//serialprintmonitorconnecting

//if wifi not connect serial print monitor not connectedwhile (WiFi.status() != WL\_CONNECTED) {Serial.println("NOTCONNECTED");

delay(300);

}

b=WIFI\_SSID;//definevaribleblikevariablewifi\_ssidSerial.println();

Serial.print("IP Address: "); //serial print monitor ip address:Serial.println(WiFi.localIP()); //serial print monitor wifi localip

Serial.println();

Firebase.begin(FIREBASE\_HOST,FIREBASE\_Authorization\_key);//iniatitethe connectionwith firebase

}

voidloop(){

bool state\_pir = digitalRead(pir); //declare variable state\_pir asa digitalread pirsensor(boolean)

//getstringfirebasedataled

if(Firebase.getString(firebaseData,"/ESP32APP\_LED/LED"))

{

Stringmessage\_fb=firebaseData.stringData();if (message\_fb!= message){

message=message\_fb;

if(message.indexOf("LEDON")!=-1){digitalWrite(led,HIGH);}

if(message.indexOf("LEDOFF")!=-1){digitalWrite(led,LOW);}

}

}

//get string firebase data set status motion sensorif (Firebase.getString(firebaseData,"/ESP32APP\_MOTION/ALARM\_SET")){

Stringmessage1\_fb=firebaseData.stringData();if (message1\_fb != message1){

message1 = message1\_fb;if(message1.indexOf("ALARMSETON")!=-1){a=1;

s\_pir="READY";

}

if(message1.indexOf("ALARMSETOFF")!=-1){digitalWrite(buzzer,LOW);

s\_pir="OFF";a=0;

}

}

}

//condition pir sensor if detect objectif (state\_pir==1 && a==1){digitalWrite(buzzer,HIGH);

s\_pir="THIEFFFF";

}

//declare variable hum as a dht read humidity(%) andtemperature (c)

float hum = dht.readHumidity();floattemp=dht.readTemperature();

//condition dht sensor if not connect with esp32if (isnan(hum)||isnan(temp) ){

Serial.println(F("FailedtoreadfromDHTsensor!"));c= "FAILED";

}

else{

c="READY";

}

Serial.print("Temperature:");//serialprintmonitortemperature:

Serial.print(temp); //serial print monitor value temperatureSerial.print("°C");//serialprint monitor °C

Serial.print("Humidity:");//serialprintmonitorhumidity:Serial.print(hum); //serial print monitor value humiditySerial.print("%"); //serialprintmonitor%

Serial.print("");

Serial.print(message); //serial print monitor value massageSerial.print(" ");

Serial.print(message1); //serial print monitor value massage1Serial.print(" ");

Serial.print(s\_pir); //serial print monitor value s\_pirSerial.println();

Firebase.setFloat(firebaseData,"/ESP32APP\_DHT11/TEMPERATURE",temp);//firebasedataset floatvaluetemperature

Firebase.setFloat(firebaseData,"/ESP32APP\_DHT11/HUMIDITY",hum);//firebasedatasetfloatvaluehumidity

Firebase.setString(firebaseData,"/ESP32APP\_DHT11/STATUS",c);//firebasedatasetstringvaluec

Firebase.setString(firebaseData,"/ESP32APP\_MOTION/BUZZER", s\_pir); //firebase data setstring values\_pir

Firebase.setString(firebaseData,"/ESP32APP\_WIFI/STATUS",b);//firebasedatasetstringvalueb

delay(200);//delay200ms

}

#include <FirebaseESP32.h> //call library firebase esp32#include<WiFi.h>//calllibrarywifi

#include"DHT.h"//calllibrarydht

#define FIREBASE\_HOST "https://smart-home-7b44f-default-rtdb.firebaseio.com" //declare variable as a firebase host#define WIFI\_SSID "Wokwi-GUEST" //declare variable as awifi\_ssid

#defineWIFI\_PASSWORD""//declarevariableasawifipassword

#define FIREBASE\_Authorization\_key"oTRPje2my9ROlP2dIhJMzEd66sd3arqmtC4WG6cv"

//declarevariableasafirebaseauthkey

FirebaseData firebaseData; //Define the FirebaseData object asfirebaseData

FirebaseJsonjson;//DefinetheFirebaseJsonasjson

intled=27; //declare variableasainteger

String message = ""; //declare variable as a stringStringmessage1="";//declarevariableasastringString s\_pir = ""; //declare variable as a stringString b= "";//declarevaribleasa string

String c = ""; //declare variable as a stringbool a = 0; //declare variable as a boolean#definepir14//declarevariableasa pin pir

#define buzzer 12 //declare variable as a pin buzzer#define DHTPIN 13 //declare variable as a pin dht#defineDHTTYPEDHT22//declarevariableasadht11DHTdht(DHTPIN,DHTTYPE);//connectesp32todht

voidsetup(){

pinMode(led, OUTPUT); //led as an outputpinMode(pir, INPUT); //pir as an inputpinMode(buzzer,OUTPUT);//buzzerasanoutputSerial.begin(115200); //open serial connectiondht.begin();//initiatetheconnection withthedht

WiFi.begin(WIFI\_SSID,WIFI\_PASSWORD);//initiatetheconnectionwith ap

Serial.println("Connecting...");//serialprintmonitorconnecting

//ifwifinotconnectserialprintmonitornotconnectedwhile (WiFi.status() != WL\_CONNECTED) {Serial.println("NOTCONNECTED");

delay(300);

}

b=WIFI\_SSID;//definevaribleblikevariablewifi\_ssid

Serial.println();

Serial.print("IP Address: "); //serial print monitor ip address:Serial.println(WiFi.localIP()); //serial print monitor wifi localip

Serial.println();

Firebase.begin(FIREBASE\_HOST,FIREBASE\_Authorization\_key);//iniatitethe connectionwith firebase

}

voidloop(){

bool state\_pir = digitalRead(pir); //declare variable state\_pir asa digitalread pirsensor(boolean)

//getstringfirebasedataled

if(Firebase.getString(firebaseData,"/ESP32APP\_LED/LED"))

{

Stringmessage\_fb=firebaseData.stringData();if (message\_fb!= message){

message=message\_fb;

if(message.indexOf("LEDON")!=-1){digitalWrite(led,HIGH);}

if(message.indexOf("LEDOFF")!=-1){digitalWrite(led,LOW);}

}

}

//get string firebase data set status motion sensorif (Firebase.getString(firebaseData,"/ESP32APP\_MOTION/ALARM\_SET")){

String message1\_fb = firebaseData.stringData();if (message1\_fb != message1){

message1=message1\_fb;

if(message1.indexOf("ALARMSETON")!=-1){a=1;

s\_pir="READY";

}

if(message1.indexOf("ALARMSETOFF")!=-1){digitalWrite(buzzer,LOW);

s\_pir="OFF";a=0;

}

}

}

//condition pir sensor if detect objectif (state\_pir==1 && a==1){digitalWrite(buzzer,HIGH);

s\_pir="THIEFFFF";

}

//declare variable hum as a dht read humidity(%) andtemperature (c)

float hum = dht.readHumidity();floattemp=dht.readTemperature();

//condition dht sensor if not connect with esp32if (isnan(hum)||isnan(temp) ){

Serial.println(F("FailedtoreadfromDHTsensor!"));c= "FAILED";

}

else{

c="READY";

}

Serial.print("Temperature:");//serialprintmonitortemperature:

Serial.print(temp); //serial print monitor value temperatureSerial.print("°C");//serialprint monitor °C

Serial.print(" Humidity: "); //serial print monitor humidity:Serial.print(hum); //serial print monitor value humiditySerial.print("%"); //serialprintmonitor%

Serial.print("");

Serial.print(message); //serial print monitor value massageSerial.print(" ");

Serial.print(message1); //serial print monitor value massage1Serial.print(" ");

Serial.print(s\_pir); //serial print monitor value s\_pirSerial.println();

Firebase.setFloat(firebaseData,"/ESP32APP\_DHT11/TEMPERATURE",temp);//firebasedataset floatvaluetemperature

Firebase.setFloat(firebaseData,"/ESP32APP\_DHT11/HUMIDITY",hum);//firebasedatasetfloatvaluehumidity

Firebase.setString(firebaseData,"/ESP32APP\_DHT11/STATUS",c);//firebasedatasetstringvaluec

Firebase.setString(firebaseData,"/ESP32APP\_MOTION/BUZZER", s\_pir); //firebase data setstring values\_pir

Firebase.setString(firebaseData,"/ESP32APP\_WIFI/STATUS",b);//firebasedatasetstringvalueb

delay(200);//delay200ms

}

diagram.json

{

"version":1,

"author": "chandra kirana","editor":"wokwi",

"parts":[

{"type":"wokwi-esp32-devkit-v1","id":"esp","top":0,

"left":0, "attrs":{}},

{

"type":"wokwi-dht22",

"id":"dht1",

"top":-2.08,

"left":119.41,

"attrs":{"temperature":"30.1","humidity":"29"}

},

{

"type":"wokwi-led",

"id":"led1",

"top":36.62,

"left":-55.84,

"attrs":{"color":"red","flip":"1"}

},

{

"type":"wokwi-pir-motion-sensor","id":"pir1",

"top":-65.32,

"left":-97.06,

"attrs":{}

},

{

"type":"wokwi-buzzer",

"id":"bz1",

"top":65.51,

"left":-119.56,

"attrs":{"volume":"0.1"}

},

{

"type":"wokwi-led",

"id":"led2",

"top":102.36,

"left":-203.89,

"attrs":{"color":"red","flip":""}

}

],

"connections":[

["esp:TX0","$serialMonitor:RX","",[]],

["esp:RX0","$serialMonitor:TX","",[]],

["dht1:VCC","esp:3V3","red",["v0"]],

["pir1:OUT","esp:D14","green",["v0"]],

["pir1:GND","esp:GND.2","black",["h30.28","v122.32"]

],

["esp:GND.2","led1:C","black",["h0"]],

["pir1:VCC","dht1:VCC","red",["v27.99","h-70.08",

"v158.75","h261.03"]],

[ "bz1:2", "esp:D12", "green", [ "v10.06", "h28.07", "v-29.68"]],

[ "esp:GND.2", "bz1:1", "black", [ "h-36.45", "v22.63", "h-58.84"]],

[ "dht1:GND", "esp:GND.2", "black", [ "v92.14", "h-198.86", "v-55.96" ]],

["led1:A","esp:D27","green",["v0"]],

["dht1:SDA","esp:D13","green",["v116.37","h-163.6","v-91.7"]],

["bz1:1","led2:C","green", ["v19.44","h-99.3"]],

["led2:A","bz1:2","green",["v13.82","h100.85"]]

],

"dependencies":{}

}