**NAAN MUDHALVAN – PROFESSIONAL READINESS FOR**

**INNOVATION, EMPLOYMENT AND ENTERPRENEURSHIP**

**ASSIGNMENT – 1**

|  |  |
| --- | --- |
| **STUDENT NAME** | **NITHISHNA** |
| **STUDENT ROLL NO** | **814720104031** |

**QUESTION:**

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

* Example: pir sensor for home security, servo motor for door lock system.
* Hint: replicate tinkercad code and connections in wokwi and integrate both codes to a single code.

**LINK:**

https://wokwi.com/projects/364535604848980993

**CODE:**

#define BLYNK\_TEMPLATE\_ID "TMPLgCeV0y1b"

#define BLYNK\_DEVICE\_NAME "Home"

#define BLYNK\_AUTH\_TOKEN "93h-1b23ewIQooDTdB2y2COGacfYkbdO"

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27, 20, 4);

#define BLYNK\_PRINT **Serial**

#include <WiFi.h>

#include <WiFiClient.h>

#include <BlynkSimpleEsp32.h>

#include "DHTesp.h"

BlynkTimer timer;

char auth[] = BLYNK\_AUTH\_TOKEN;

char ssid[] = "Wokwi-GUEST"; char pass[] = ""; int val = 0, va1,va2,va3,va4,va5,ge, t =15 ; float tmp,hum = 0;

int ledPin = 33; int inputPin = 27; int pirState,k; int v = 0;

//temp symbol

byte t1[8]={B00000, B00001, B00010, B00100, B00100, B00100, B00100, B00111,}; byte t2[8]={B00111, B00111, B00111, B01111,B11111, B11111, B01111, B00011,}; byte t3[8]={B00000, B10000, B01011, B00100, B00111, B00100, B00111, B11100,}; byte t4[8]={B11111, B11100, B11100, B11110,B11111, B11111, B11110, B11000,};

//humidity symbol byte hum1[8]={B00000, B00001, B00011, B00011,B00111, B01111, B01111, B11111,}; byte hum2[8]={B11111, B11111, B11111, B01111,B00011, B00000, B00000, B00000,}; byte hum3[8]={B00000, B10000, B11000, B11000, B11100, B11110, B11110, B11111,};

byte hum4[8]={B11111, B11111, B11111, B11110, B11100, B00000, B00000,

B00000,};

//Home Symbol

byte house1[8]={B00000, B00001, B00011, B00011, B00111, B01111, B01111, B11111,};

byte house2[8]={B11111, B11111, B11100, B11100, B11100, B11100, B11100, B11100,};

byte house3[8]={B00000, B10010, B11010, B11010, B11110, B11110, B11110, B11111,};

byte house4[8]={B11111, B11111, B11111, B10001, B10001, B10001, B11111, B11111,}; byte d[8] = { 0b00011,0b00011,0b00000,0b00000,0b00000,0b00000,0b00000,0b00000 }; byte Lck[] = { B01110, B10001, B10001, B11111, B11011, B11011, B11111, B00000 };

DHTesp temps;

BLYNK\_WRITE(V0){ va1 = param.asInt(); digitalWrite(5, va1);

}

BLYNK\_WRITE(V1){ va2 = param.asInt(); digitalWrite(18, va2);

}

BLYNK\_WRITE(V2){ va3 = param.asInt(); digitalWrite(19, va3);

}

BLYNK\_WRITE(V3){ va4 = param.asInt(); digitalWrite(4, va4);

}

BLYNK\_WRITE(V4){ va5 = param.asInt(); digitalWrite(2, va5);

}

BLYNK\_WRITE(V7) { pirState = param.asInt(); if(pirState == 0){ digitalWrite(ledPin, LOW); k = 1;

ge = 0;

} else {

digitalWrite(ledPin, HIGH); k= 0; ge = 1; }

} void myTimer()

{

Blynk.virtualWrite(V5,tmp);

Blynk.virtualWrite(V6,hum);

}

void setup()

{

**Serial**.begin(115200);

Blynk.begin(auth, ssid, pass);

pinMode(5, OUTPUT); pinMode(18, OUTPUT); pinMode(19, OUTPUT); pinMode(4, OUTPUT); pinMode(23,INPUT); pinMode(2,OUTPUT); temps.setup(t, DHTesp::DHT22); pinMode(ledPin, OUTPUT); pinMode(inputPin, INPUT\_PULLUP);

lcd.init(); lcd.backlight(); digitalWrite(5, LOW); digitalWrite(18, LOW); digitalWrite(19, LOW); digitalWrite(21, LOW);

lcd.setCursor(0,0); lcd.print("CircuitDesignContest"); lcd.setCursor(8,1); lcd.print("2022"); lcd.setCursor(0,2); lcd.print("--------------------"); lcd.setCursor(9,3); lcd.print("- eDiYLaBs"); delay(3000); lcd.clear(); lcd.createChar(6, Lck); lcd.createChar(1,house1); lcd.createChar(2,house2); lcd.createChar(3,house3); lcd.createChar(4,house4); lcd.setCursor(1,2); lcd.write(1); lcd.setCursor(1,3); lcd.write(2); lcd.setCursor(2,2); lcd.write(3); lcd.setCursor(2,3); lcd.write(4); lcd.setCursor(17,2); lcd.write(1); lcd.setCursor(17,3); lcd.write(2); lcd.setCursor(18,2); lcd.write(3); lcd.setCursor(18,3); lcd.write(4);

lcd.setCursor(19,0); lcd.write(6); lcd.setCursor(9,0); lcd.print("connected-"); lcd.setCursor(2,1); lcd.print("HOME AUTOMATION"); lcd.setCursor(6,2); lcd.print("USING IOT"); delay(3000);

Blynk.virtualWrite(V7, pirState); timer.setInterval(1000L, myTimer);

}

void loop() { Blynk.run(); timer.run(); val = digitalRead(23); if(val == 1)

{

digitalWrite(2,va5); }

else{ digitalWrite(2,LOW);

}

TempAndHumidity x = temps.getTempAndHumidity(); tmp = x.temperature ; hum = x.humidity ;

v = digitalRead(inputPin); if (v == HIGH) { if (k == 1) { digitalWrite(ledPin, LOW); k = 0 ; ge = 0; } else if (k == 0) { digitalWrite(ledPin, HIGH); k = 1; ge = 1;

}

}

if (va1 == 1){ lcd.clear(); lcd.setCursor(19,0); lcd.write(6); lcd.setCursor(0, 1); lcd.print("SW\_1= "); lcd.print("ON "); } else{ lcd.clear(); lcd.setCursor(19,0); lcd.write(6); lcd.setCursor(0, 1); lcd.print("SW\_1= "); lcd.print("OFF"); } if (va2 == 1){

lcd.setCursor(11, 1); lcd.print("SW\_2= "); lcd.print("ON ");

}

else{ lcd.setCursor(11, 1); lcd.print("SW\_2= "); lcd.print("OFF"); } if (va3 == 1){ lcd.setCursor(0, 2); lcd.print("SW\_3= "); lcd.print("ON "); } else{ lcd.setCursor(0, 2); lcd.print("SW\_3= "); lcd.print("OFF"); } if (va4 == 1){ lcd.setCursor(11, 2); lcd.print("SW\_4= "); lcd.print("ON "); } else{ lcd.setCursor(11, 2); lcd.print("SW\_4= "); lcd.print("OFF"); } if (va5 == 1){ lcd.setCursor(0, 3); lcd.print("OD\_L= "); lcd.print("ON "); } else{ lcd.setCursor(0, 3); lcd.print("OD\_L= "); lcd.print("OFF"); } if (ge == 1){ lcd.setCursor(11, 3); lcd.print("WR\_L= "); lcd.print("ON "); } else{

lcd.setCursor(11, 3); lcd.print("WR\_L= "); lcd.print("OFF"); } delay(1500); lcd.clear(); lcd.createChar(1,t1); lcd.createChar(2,t2); lcd.createChar(3,t3); lcd.createChar(4,t4); lcd.createChar(5, d); lcd.createChar(6, Lck); lcd.setCursor(19,0); lcd.write(6); lcd.setCursor(1,1); lcd.write(1); lcd.setCursor(1,2); lcd.write(2); lcd.setCursor(2,1); lcd.write(3); lcd.setCursor(2,2); lcd.write(4); lcd.setCursor(4,1); lcd.print("Temperature :"); lcd.setCursor(7,2); lcd.print(tmp); lcd.setCursor(11,2); lcd.write(5); lcd.setCursor(12,2); lcd.print("C"); delay(750); lcd.clear(); lcd.createChar(1,hum1); lcd.createChar(2,hum2); lcd.createChar(3,hum3); lcd.createChar(4,hum4); lcd.setCursor(19,0); lcd.write(6); lcd.setCursor(3,1); lcd.write(1); lcd.setCursor(3,2); lcd.write(2); lcd.setCursor(4,1); lcd.write(3); lcd.setCursor(4,2); lcd.write(4); lcd.setCursor(6,1); lcd.print("Humidity :");

lcd.setCursor(7,2); lcd.print(hum); lcd.setCursor(12,2); lcd.print("%"); delay(750);

}

**CIRCUIT DIAGRAM:**

