**Meranie teploty a vlhkosti, senzor na kontrolu otvoreného okna a zasielanie informácie stave okna na e-mail**

**Vypracovala:** Mgr. Jana Šimonová

**Rok:** 2024/2025

**Predmet:** Internet veci

**Ročník:** 2. RŠI

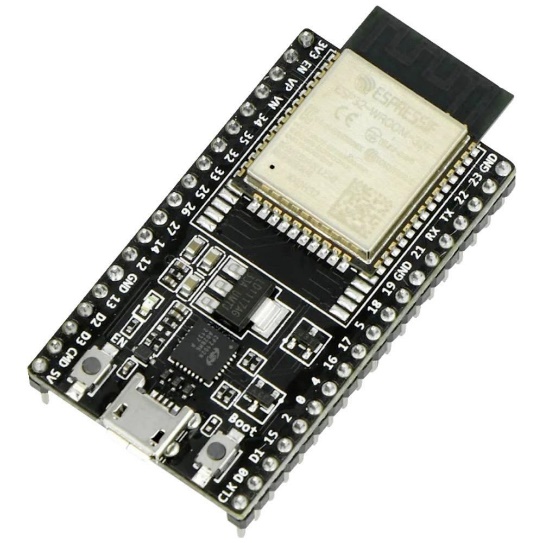
Odkaz na stiahnutie:

**Cieľ projektu**: Cieľom projektu je vytvorenie projektu prostredníctvom ESP32 a senzora DHT22, OLED displeja 0.66 shield, dverového senzora MC-51 na zaznamenávanie teploty a vlhkosti a zaznamenávanie stavu okna (otvorené/zatvorené). Informáciu o otvorenom prípadne zatvorenom okne mikrokontrolér zašle na e-mail.

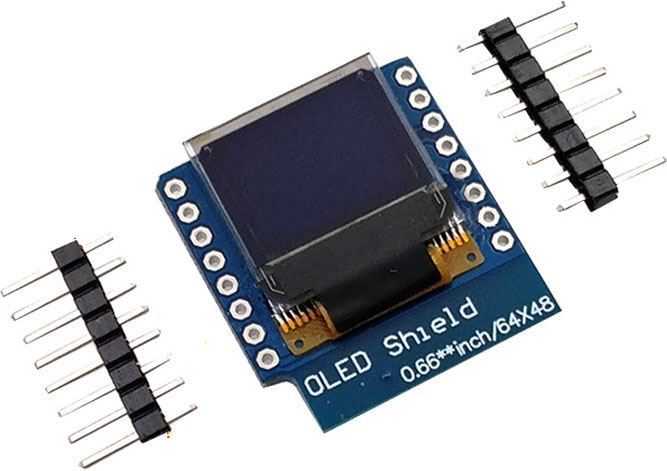
**Použitie v praxi**: Projekt môže byť využitý pre bežných spotrebiteľov v domácnostiach, firmách. Je možné v tomto projekte pridať ku kontaktnému poľu napr. alarm.

**Použité komponenty**:

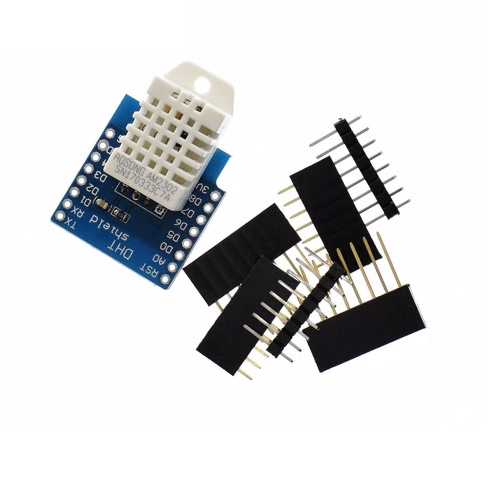
1. Mikrokontrolér ESP32 WROOM-32: najdôležitejšia súčiastka projektu, ktorý spracováva údaje so senzora a prostredníctvom WIFI komunikuje s Adafruit



2. Wemos mini OLED 0.66" shield displej



3. Wemos mini shield senzor teploty a vlhkosti DHT22



4. MC-51 dverový senzor



5. Ďalšie komponenty: kontaktná doska SAB-120, prepojovacie káble rôznej farby /11/

**Umiestnenie pinov na ESP32 WROOM32**

Obrázok, na ktorom je text, snímka obrazovky, okruh

Automaticky generovaný popis

**Zapojenie komponentov**:

1. nainštalovanie vývojového prostredia Arduiono IDE cez www.arduiono.cc

1. zapojenie mikrokotroléra ESP32 do dosky pomocou micro USB

2. bolo potrebné vytvoriť virtuálneho COM PORT pre komunikáciu s ESP32 pomocou ovládača CP210x

3. zapojenie jednotlivých komponentov:

DHT22 - ESP32 – na 3,3V cez červenú zbernicu - červený kábel, GND cez modrú zbernicu – žltý kábel, D4-D4 - modrý kábel /GPIO4/

DIPLEJ model oled 0.66 WEMOS mini – pripojené na červenú zbernicu 3,3V fialový kábel a modrú zbernicu /GND/ sivý kábel, a piny D21/ESP/ - D2 /SDA/-biely a D22/ESP/ - D1/SCL/ čierny kábel.

Dverový senzor – GND mínus – čierny kábel, D26 červený kábel /GPIO26/

4. nahratie knižníc: include <DHT.h>  // kniznica potrebna pre DHT22 snimac

#include <DHT\_U.h>

#include <WiFi.h> //zahrnie knižnicu na prácu s WiFi modulom

#include <ESP\_Mail\_Client.h> //zahrnie knižnicu na prácu s e-mailom

#include <Adafruit\_Sensor.h>  // kniznica pre senzor

#include <Wire.h>    //kniznica pre komunikaciu i2c komunikácia s displejom + DHT senzor

#include <Adafruit\_GFX.h>  // kniznica pre displej

#include <Adafruit\_SSD1306.h>  // knižnica pre displej mini OLED 0.66 shield

**Kód Arduino IDE:**

#include <DHT.h>  // kniznica potrebna pre DHT22 snimac

#include <DHT\_U.h>

#include <WiFi.h> //zahrnie knižnicu na prácu s WiFi modulom

#include <ESP\_Mail\_Client.h> //zahrnie knižnicu na prácu s e-mailom

#include <Adafruit\_Sensor.h>  // kniznica pre senzor

#include <Wire.h>    //kniznica pre komunikaciu i2c komunikácia s displejom + DHT senzor

#include <Adafruit\_GFX.h>  // kniznica pre displej

#include <Adafruit\_SSD1306.h>  // pozor kniznica pre MiniOled  Adafruit\_SSD1306\_Wemos\_OLED nie klasicka Adafruit\_SSD1306

//NASTAVENIE WIFI PRIPOJENIA

#define WIFI\_NAZOV "nazov\_wifi" //definuje premennú WIFI\_NAZOV

#define WIFI\_HESLO "xxxxxxxxx" //definuje premennú WIFI\_HESLO

//NASTAVENIE SMTP PROTOKLU

//špecifikujeme Simple Mail Transfer Protocol

#define SMTP\_HOST "mail.webhouse.sk" //definuje premennú SMTP\_HOST (posielame z účtu Gmail)

#define SMTP\_PORT 587 //definuje premennú SMTP\_PORT (predvolený port serveru)

//špecifikujeme odosielateľa a príjemcu

#define ODOSIELATEL\_EMAIL "esp32test@xxxxx.sk" //definuj e-mail odosielateľa

#define ODOSIELATEL\_HESLO "xxxxxxxxx" //definuje heslo do e-mailu odosielateľa

#define PRIJEMCA\_EMAIL "xxxxxx@xxxxxx.sk" //definuje e-mail príjemcu

SMTPSession smtp; //slúži na odosielanie e-mailov

//DEFIN reset displ none

#define OLED\_RESET -1  // displej sa neda resetovať

Adafruit\_SSD1306 display(OLED\_RESET);

// uvodneho loga

#define NUMFLAKES 10

#define XPOS 0

#define YPOS 1

#define DELTAY 2

#define LOGO16\_GLCD\_HEIGHT 16

#define LOGO16\_GLCD\_WIDTH  16

static const unsigned char PROGMEM logo16\_glcd\_bmp[] =

{ B00000000, B11000000,

  B00000001, B11000000,

  B00000001, B11000000,

  B00000011, B11100000,

  B11110011, B11100000,

  B11111110, B11111000,

  B01111110, B11111111,

  B00110011, B10011111,

  B00011111, B11111100,

  B00001101, B01110000,

  B00011011, B10100000,

  B00111111, B11100000,

  B00111111, B11110000,

  B01111100, B11110000,

  B01110000, B01110000,

  B00000000, B00110000 };

#if (SSD1306\_LCDHEIGHT != 48)

#error("Height incorrect, please fix Adafruit\_SSD1306.h!");

#endif

// Uncomment the type of sensor in use:

//#define DHTTYPE    DHT11     // DHT 11

#define DHTTYPE    DHT22     // DHT 22 (AM2302)  - POUZITY SENZOR

//#define DHTTYPE    DHT21     // DHT 21 (AM2301)

#define DHTPIN 4 // Digital pin connected to the DHT sensor  POUZITE GPIO 04

DHT dht(DHTPIN, DHTTYPE);

//Def premenen - OKNO senzor otvorenia

int OKNO\_PIN = 26;                          //OKNO kontakt  POUZITE GPIO 26  D26

int OKNO\_StavSucasny = LOW;               // premenna sucasny stav dverneho senzora

int OKNO\_StavMinuly = LOW;                // premenna minuly stav dverneho senzora

void setup() {

 Serial.begin(9600);

 Serial.println();

 Serial.print("Pripájam k sieti...");

 WiFi.begin(WIFI\_NAZOV, WIFI\_HESLO); //spustí WiFi modul

 while (WiFi.status()!= WL\_CONNECTED){ //pokiaľ nie je WiFi pripojená, vypisuje bodky, toto je podmienka aby to vedelo odslat email, ak nie je pripojen k wifi ani nezacne zobrazovat udaje

    Serial.print(".");

     delay(2000);

  }

  Serial.println("");

  Serial.print("Pripojené na WiFi sieť ");

  Serial.println(WIFI\_NAZOV);

  Serial.print("IP adresa mikrokontroléra je: ");

  Serial.println(WiFi.localIP()); //vypíše IP adresu ESP32

  Serial.println();

ESP\_Mail\_Session session;

  session.server.host\_name = SMTP\_HOST ;

  session.server.port = SMTP\_PORT;

  session.login.email = ODOSIELATEL\_EMAIL;

  session.login.password = ODOSIELATEL\_HESLO;

  session.login.user\_domain = "";

 SMTP\_Message message;

  message.sender.name = "ESP32";

  message.sender.email = ODOSIELATEL\_EMAIL;

  message.subject = "ESP32 is UP";

  message.addRecipient("TESTER",PRIJEMCA\_EMAIL);

 //Send HTML message

  String htmlMsg = "<div style=\"color:#FF0000;\"><h1>ESP sa ZOBUDILO</h1><p>Sent by ESP</p></div>";

  message.html.content = htmlMsg.c\_str();

  message.html.content = htmlMsg.c\_str();

  message.text.charSet = "us-ascii";

  message.html.transfer\_encoding = Content\_Transfer\_Encoding::enc\_7bit;

 if (!smtp.connect(&session)) //ak sa neporadí spojiť so serverom,

    return; //vráti sa späť na začiatok

 else

 Serial.println("SMTP-pripojene");

if (!MailClient.sendMail(&smtp, &message))

  Serial.println("Error sending Email, " + smtp.errorReason());

else

 Serial.println("Odoslane info o pripojeni");

  pinMode(OKNO\_PIN, INPUT\_PULLUP);  //nastavenie rezimu INPUT pre OKNO\_PIN - treba nsatavit rezim PULLUP lebo ma to nejaky odpor aj ked je zopnute

  dht.begin();

  display.begin(SSD1306\_SWITCHCAPVCC, 0x3C);  // initialize with the I2C addr 0x3C (for the 64x48)

  // Show image buffer on the display hardware.

  // Since the buffer is intialized with an Adafruit splashscreen

  // internally, this will display the splashscreen.

  //zobrazenie uvodneho  logo

  display.display();

  delay(2000);

  // Clear the buffer.

  display.clearDisplay();

  }

void zobraz\_na\_disp\_stav\_okna() {

   // display OKNO\_STAV

  display.setTextSize(1);

  display.setCursor(0, 25);

  display.print("OKNO: ");

  display.setTextSize(1,2);

  display.setCursor(10,35);

  if (OKNO\_StavSucasny == LOW)

     {display.print("ZATVORENE");

     }

   else

    {display.print("OTVORENE");

    }

  display.display(); //zobrazim

  delay(2000); //pockam

  display.clearDisplay(); // vymazem

}

void loop() {

  delay(1000);

  OKNO\_StavMinuly = OKNO\_StavSucasny;                   // ulozime stary stav dverneho senzora

  OKNO\_StavSucasny = digitalRead(OKNO\_PIN);            // nacitame novy stav

 if (OKNO\_StavMinuly == LOW && OKNO\_StavSucasny == HIGH)            // zmena: LOW -> HIGH

  {  Serial.println("OKNO OTVORENE!");

     zobraz\_na\_disp\_stav\_okna();

          // do tejto časti je možné pridať alarm, notifikaciu...

ESP\_Mail\_Session session;

  session.server.host\_name = SMTP\_HOST ;

  session.server.port = SMTP\_PORT;

  session.login.email = ODOSIELATEL\_EMAIL;

  session.login.password = ODOSIELATEL\_HESLO;

  session.login.user\_domain = "";

 SMTP\_Message message;

  message.sender.name = "ESP32";

  message.sender.email = ODOSIELATEL\_EMAIL;

  message.subject = "ESP32 is UP";

  message.addRecipient("TESTER",PRIJEMCA\_EMAIL);

           String htmlMsg = "<div style=\"color:#FF0000;\"><h1>OKNO OTVORENE</p></div>";

      message.html.content = htmlMsg.c\_str();

      message.html.content = htmlMsg.c\_str();

      message.text.charSet = "us-ascii";

      message.html.transfer\_encoding = Content\_Transfer\_Encoding::enc\_7bit;

 if (!smtp.connect(&session)) //ak sa neporadí spojiť so serverom,

    return; //vráti sa späť na začiatok

 else

 Serial.println("SMTP-pripojene");

if (!MailClient.sendMail(&smtp, &message))

  Serial.println("Error sending Email, " + smtp.errorReason());

else

 Serial.println("Odoslane info o pripojeni");

  }

 else

   if (OKNO\_StavMinuly == HIGH && OKNO\_StavSucasny == LOW)          // zmena: HIGH -> LOW

     {

        Serial.println("OKNO ZATVORENE!");

        zobraz\_na\_disp\_stav\_okna();

ESP\_Mail\_Session session;

  session.server.host\_name = SMTP\_HOST ;

  session.server.port = SMTP\_PORT;

  session.login.email = ODOSIELATEL\_EMAIL;

  session.login.password = ODOSIELATEL\_HESLO;

  session.login.user\_domain = "";

 SMTP\_Message message;

  message.sender.name = "ESP32";

  message.sender.email = ODOSIELATEL\_EMAIL;

  message.subject = "ESP32 is UP";

  message.addRecipient("TESTER",PRIJEMCA\_EMAIL);

         String htmlMsg = "<div style=\"color:#FF0000;\"><h1>OKNO ZATVORENE</p></div>";

         message.html.content = htmlMsg.c\_str();

        message.html.content = htmlMsg.c\_str();

        message.text.charSet = "us-ascii";

       message.html.transfer\_encoding = Content\_Transfer\_Encoding::enc\_7bit;

 if (!smtp.connect(&session)) //ak sa neporadí spojiť so serverom,

    return; //vráti sa späť na začiatok

 else

 Serial.println("SMTP-pripojene");

if (!MailClient.sendMail(&smtp, &message))

  Serial.println("Error sending Email, " + smtp.errorReason());

else

 Serial.println("Odoslane info o pripojeni");

        }

  //Teraz citame teplotu a vlhkost

  float t = dht.readTemperature();   // do float premenej t - nacitavam teplotu

  float h = dht.readHumidity();  // h - vlhkost

  // kontrola ci vieme citat data z snimaca DHT22 ak nie vypise hlasku

  if (isnan(h) || isnan(t)) {

    Serial.println("Chyba citania z teplotneho senzora!");

  }

  // clear display

  display.clearDisplay();

  // Print to serial monitor  -  pre testovanie DHT22 vypis na ser.konzolu

  Serial.print(F("Vlhkost: ")); // vypisuje na konzolu

  Serial.print(h);

  Serial.print(F(" %"));

  Serial.println(F(" "));

  Serial.print(F("Teplota: "));

  Serial.print(t);

  Serial.print(F(" °C"));

  Serial.println(F(" "));

  Serial.print(F("STAV OKNA: "));

  Serial.println(OKNO\_StavSucasny);

  //Serial.println();

  //Serial.println(F("Nacitavam ..... "));

  // display temperature  na dsiplej

  display.setTextSize(1);  // velkost text

  display.setTextColor(WHITE);

  display.setCursor(0,0); // pozicia

  display.print("Teplota: ");

  display.setTextSize(1,2);

  display.setCursor(10,10);

  display.print(t);

   display.print(" C");

  // display humidity

  display.setTextSize(1);

  display.setCursor(0, 25);

  display.print("Vlhkost: ");

  display.setTextSize(1,2);

  display.setCursor(10,35);

  display.print(h);

  display.print(" %");

  display.display(); //zobrazim

  delay(2000); //pockam

  display.clearDisplay(); // vymazem

   //

   display.setTextSize(1);

   display.setCursor(0, 0);

  display.print("Nacitavam...........");

  display.display(); //zobrazim

  display.display();

}