

Pripojenie na sieť INTRAK s doskou ESP32

Pripojenie na sieť INTRAK prostredníctvom vývojovej dosky s čipom ESP32

Inštalácia Arduino core pre ESP32 (vyžaduje inštaláciu GIT-.u):

Win: <https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/windows.md>

MAC: <https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/mac.md>

Linux: https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/debian_ubuntu.md

Fedora: <https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/fedora.md>

OpenSUSE: <https://github.com/espressif/arduino-esp32/blob/master/docs/arduino-ide/opensuse.md>

Podľa typu prevodníka nainštalujeme jeho ovládač, najčastejšie CP2102:

<https://www.silabs.com/products/development-tools/software/usb-to-uart-bridge-vcp-drivers>

alebo CH340:

<https://sparks.gogo.co.nz/ch340.html>

Po úspešnej inštalácii Arduino Core pokračujeme podľa krokov.

V prostredí ArduinoIDE v záložke **Nástroje** vyberieme našu vývojovú **dosku** a **COM** port.

Program pre zistenie MAC adresy zariadenia:

```
#include <WiFi.h>
```

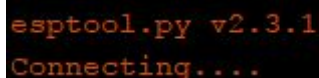
```
byte mac[6];
```

```
void setup(){
  Serial.begin(115200);
  delay(1000);
  WiFi.mode(WIFI_STA);
  WiFi.begin();
  Serial.println("WiFi spustene");
  WiFi.macAddress(mac);
  Serial.print("MAC adresa: ");
  Serial.print(mac[0],HEX);
  Serial.print(":");
  Serial.print(mac[1],HEX);
  Serial.print(":");
  Serial.print(mac[2],HEX);
  Serial.print(":");
  Serial.print(mac[3],HEX);
  Serial.print(":");
  Serial.print(mac[4],HEX);
  Serial.print(":");
  Serial.println(mac[5],HEX);
}
```

```
void loop(){
}
```

Program nahráme do dosky ESP32 cez USB port.

Počas nahrávania je nutné držať tlačidlo BOOT pri pripájaní esptoolu:



```
esptool.py v2.3.1
Connecting...
```

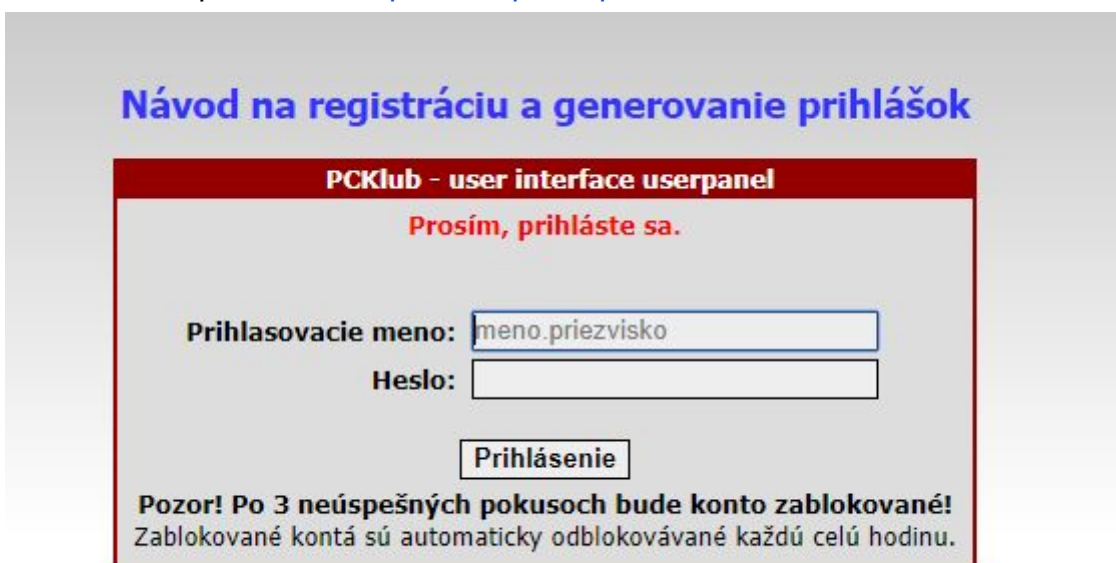
Po nahratí programu otvoríme Serial monitor na rýchlosti 115200 z ArduinoIDE, alebo cez externý terminál (Putty, Hyperterminal na príslušnom COM porte). Ak doska nevypíše nič (je stále v boot móde), je nutné ju reštartovať cez EN tlačidlo na doske. Doska vypíše MAC adresu v HEX tvare.

```
WiFi spustene
MAC adresa: 30:AE:A4: [redacted]
```

Automatické rolovanie Riadok nemá ukončovací znak 115200 baud Vymazať výstup

MAC adresu ESP dosky vpíšeme do userpanelu.

Prihlásenie do userpanelu: → <https://userpanel.pcklub.sk>



Získanú MAC adresu zapíšeme do wifi pripojenia a počkáme na jej schválenie PC klubom



Po schválení sa už môžeme pripojiť do siete INTRAK pod protokolom 802.1x.

Po overení používateľa jeho menom a heslo a následnej autorizácii RADIUS serverom prideli DHCP server adresu a umožní pripojenie do internetu. Testovací program vykoná HTTP spojenie na web <http://intrak.sk> každých 30 sekúnd.

V programe je nutné zmeniť údaje na vaše v časti identita a heslo z userpanelu:

```
#define EAP_IDENTITY "meno.priezvisko"
```

```
#define EAP_PASSWORD "heslo"
```

Program:

```
#include <WiFi.h> //Wifi library
```

```
#include "esp_wpa2.h" //kniznica s crypto funkciami pre Enterprise spojenie
```

```
const char* ssid = "INTRAK"; // SSID
```

```

const char* host = "intrak.sk"; //externy server pre HTTP spojenie po autentizacii a autorizacii
int counter = 0; //pocitadlo restartu
#define EAP_IDENTITY "meno.priezvisko" //meno.priezvisko je identita a aj meno do siete INTRAK
#define EAP_PASSWORD "heslo" //heslo
void setup() {
  Serial.begin(115200);
  delay(10);
  Serial.println();
  Serial.print("Pripojenie na siet: ");
  Serial.println(ssid);
  WiFi.disconnect(true); //odpoj wifi pred novym nastavenim wifi
  WiFi.mode(WIFI_STA); //init wifi modu - vyzadovane od Juna 2018 v Arduino core
  esp_wifi_sta_wpa2_ent_set_identity((uint8_t *)EAP_IDENTITY, strlen(EAP_IDENTITY)); //preukazanie
identity
  esp_wifi_sta_wpa2_ent_set_username((uint8_t *)EAP_IDENTITY, strlen(EAP_IDENTITY)); //preukazanie
mena
  esp_wifi_sta_wpa2_ent_set_password((uint8_t *)EAP_PASSWORD, strlen(EAP_PASSWORD));
//preukazanie hesla
  esp_wpa2_config_t config = WPA2_CONFIG_INIT_DEFAULT(); //nastav pociatocne parametre
  esp_wifi_sta_wpa2_ent_enable(&config);
  WiFi.begin(ssid); //pripoj sa na wifi siet
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
    counter++;
    if(counter>=60){ //po 30 sekundach restartuj dosku
      ESP.restart();
    }
  }
  Serial.println("");
  Serial.println("WiFi pripojene");
  Serial.println("IP adresa nastavena: ");
  Serial.println(WiFi.localIP()); //vypis LAN IP
}
void loop() {
  if (WiFi.status() == WL_CONNECTED) { //ak je doska pripojena
    counter = 0; //vynuluj pocitadlo
    Serial.println("Wifi je stale pripojene s IP: ");
    Serial.println(WiFi.localIP());
  }else if (WiFi.status() != WL_CONNECTED) { //ak strati spojenie, pripoj sa
    WiFi.begin(ssid);
  }
  while (WiFi.status() != WL_CONNECTED) { //pocas strateneho spojenia posielaj .
    delay(500);
    Serial.print(".");
    counter++;
    if(counter>=60){ //reset po 30 sekundach
      ESP.restart();
    }
  }
  Serial.print("Pripajanie na web: ");
  Serial.println(host);
  WiFiClient client;

```

```

if (client.connect(host, 80)) {
  String url = "/";
  client.print(String("GET ") + url + " HTTP/1.1\r\n" + "Host: " + host + "\r\n" + "User-Agent: ESP32\r\n" +
"Connection: close\r\n\r\n");

  while (client.connected()) {
    String line = client.readStringUntil('\n');
    if (line == "\r") {
      break;
    }
  }
  while (client.available()) {
    char c = client.read();
    Serial.write(c); //vypisuj po znaku
  }
}else{
  Serial.println("Pripojenie neuspesne");
}
  delay(30000); //pockaj 30 sekund pred novym pripojenim
}

```

Po pripojení doska vypíše IP adresu a zdrojový kód stránky ako demonštráciu úspešného pripojenia do wifi siete. Program je možné upraviť pre HTTPS spojenia, MQTT protokol, či na tvorbu vlastnej webaplikácie.

```

Pripojenie na siet: INTRAK
.....
WiFi pripojene
IP adresa nastavena:
192.168. [REDACTED]
Wifi je stale pripojene s IP:
192.168 [REDACTED]
Pripajanie na web: intrak.sk
7976
<!DOCTYPE html>
<html>
  <head>
    <title>Jedlikova 9 - Domov</title>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <meta name="viewport" content="width=device-width" />

    <link rel="shortcut icon" href="http://intrak.sk/assets/img/icons/faviconJ9.ico" type="image/x-icon">
    <link rel="icon" href="/favicon.ico" type="image/x-icon">
    <link rel="stylesheet" href="http://intrak.sk/assets/css/basic.css">
    <link rel="stylesheet" href="http://intrak.sk/assets/css/fonts-style.css">
    <link rel="stylesheet" href="http://intrak.sk/assets/css/bootstrap.min.css">

```

Pre debug výpis počas pripájania do wifi siete je možné použiť v záložke Nástroje možnosť Core Debug Level na možnosť "DEBUG", čo umožní doplnkový výpis do Serial monitoru a pomôže identifikovať chybu pri procese autentizácie a autorizácie. (Nepodporujú niektoré ESP32 dosky)

```

Pripajam sa na wifi siet: INTRAK
[D][WiFiGeneric.cpp:304] _eventCallback(): Event: 2 - STA_START
..... [D][WiFiGeneric.cpp:304] _eventCallback(): Event: 4 - STA_CONNECTED
. [D][WiFiGeneric.cpp:304] _eventCallback(): Event: 7 - STA_GOT_IP

```

CORE DEBUG LEVEL: "DEBUG"

Testované dosky:

DOIT ESP32 Devkit v1



ESP32 Devkitc v4



Testovaná verzia Arduino core pre ESP32:
Stable release June 2018

Riešenie je možné aplikovať aj na Eduroam sieť, alebo inú wifi sieť na WPA/WPA2 Enterprise štandarde s overením PEAP + MsCHAPv2, prípadne EAP-TTLS.