



**FAKULTA
ELEKTROTECHNICKÁ
ČVUT V PRAZE**



P02

ELEKTRONIKA A KOMUNIKACE PRAKTICKY

Komunikační možnosti ESP32

- Drátové (wired) – SPI, UART, I2C, CAN, ...
- Bezdrátové (wireless) – Bluetooth, WiFi, LoRa, GSM

ESP32 Communications

ESP32 offers multiple communications options

Wireless	Wired
Wifi	3 x SPI (Serial Peripheral Interface)
Bluetooth	2 x I ² C
	2 x I ² S
	3 x UART
	Ethernet MAC interface
	CAN 2.0
	IR (TX/RX)

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

Komunikační možnosti ESP32

ESP32 Communications

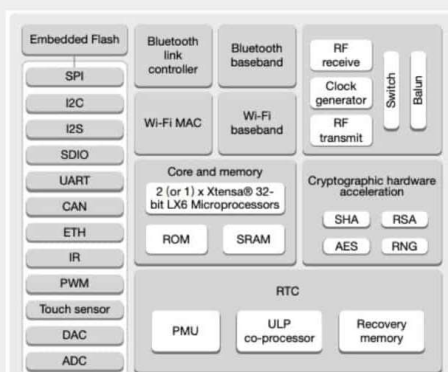


Figure 1: Functional Block Diagram

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / Wi-Fi s ESP32

Komunikační možnosti ESP32

Integrované Wi-Fi = anténní obvod se zesilovačem, různými filtry a řízením spotřeby. Podpora 802.11 b, g, n @ 150 Mbit.

Bluetooth 4.2 a BLE (Bluetooth Low Energy).

3x SPI kanály @ max. 80 MHz

2x I2C master nebo slave, standardní nebo rychlý režim.

2x I2S pro zvukové aplikace

1x CAN pro automotive aplikace

IR infračervená komunikační schopnost přijímat a vysílat

Podpora Ethernet MAC pro připojení k síti

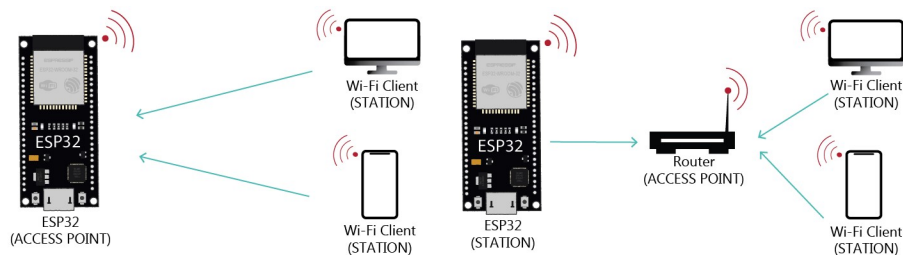
ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / Wi-Fi s ESP32

WiFi s ESP32 – AP vs. klient

ACCESS POINT (AP)= základnová stanice = poskytuje síť pro klienty

STATION = klient = připojuje se k základnové stanici

ESP umí samostatně nebo i současně oba módy



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

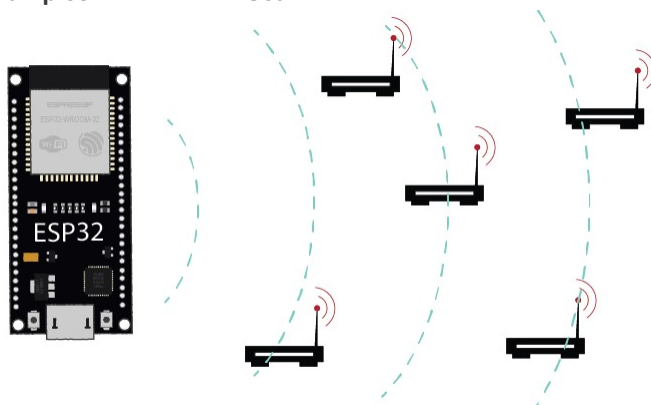
WiFi s ESP32 – možné činnosti

- skenování dostupných sítí Wi-Fi
- připojení k existující síti Wi-Fi
- získání síly připojení Wi-Fi
- kontrola stavu připojení
- opětovné připojení k síti po ztrátě připojení
- stav Wi-Fi
- režimy Wi-Fi
- získání IP adresy ESP32
- nastavení pevné IP adresy

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – skenování dostupných sítí Wi-Fi

ESP32 může skenovat blízké sítě Wi-Fi v dosahu Wi-Fi.
File > Examples > WiFi > WiFiScan.



Zdroj: <https://randomnerdtutorials.com/esp32-useful-wi-fi-functions-arduino/>

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – skenování dostupných sítí Wi-Fi

Jak zkontrolovat dostupné sítě a jejich parametry?

Funkce `WiFi.scanNetworks()` vrátí počet nalezených sítí.

```
int n = WiFi.scanNetworks();
```

Po skenování můžeme přistupovat k parametrům každé sítě.

`WiFi.SSID()` vypíše SSID pro konkrétní síť:

```
Serial.print(WiFi.SSID(i));
```

`WiFi.RSSI()` vrátí RSSI dané sítě.

RSSI je zkratka pro Indikátor síly přijatého signálu. Jedná se o odhadovanou míru úrovně výkonu, kterou zařízení RF klienta přijímá z přístupového bodu nebo směrovače.

```
Serial.print(WiFi.RSSI(i));
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – skenování dostupných sítí Wi-Fi

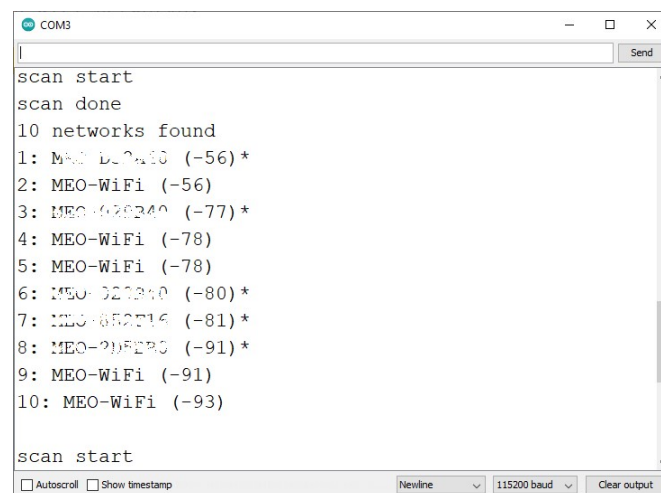
Jak zkontrolovat dostupné sítě a jejich parametry?

`WiFi.encryptionType()` vrátí typ šifrování sítě. Tento konkrétní příklad uvádí * v případě otevřených sítí. Tato funkce však může vrátit jednu z následujících možností (nejen otevřené sítě):

```
WIFI_AUTH_OPEN
WIFI_AUTH_WEP
WIFI_AUTH_WPA_PSK
WIFI_AUTH_WPA2_PSK
WIFI_AUTH_WPA_WPA2_PSK
WIFI_AUTH_WPA2_ENTERPRISE
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – skenování dostupných sítí Wi-Fi



```
COM3
scan start
scan done
10 networks found
1: MEO-2005F2B0 (-56) *
2: MEO-WiFi (-56)
3: MEO-2005F2B0 (-77) *
4: MEO-WiFi (-78)
5: MEO-WiFi (-78)
6: MEO-2005F2B0 (-80) *
7: MEO-2005F2B0 (-81) *
8: MEO-2005F2B0 (-91) *
9: MEO-WiFi (-91)
10: MEO-WiFi (-93)

scan start
```

Autoscroll ☐ Show timestamp Newline 115200 baud Clear output

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – připojení k síti

Chcete-li připojit ESP32 ke konkrétní síti Wi-Fi, musíte znát její SSID a heslo. Kromě toho musí být tato síť v dosahu ESP32 Wi-Fi.

Pro připojení ESP32 k síti Wi-Fi můžeme použít funkci `initWiFi()` :

```
void initWiFi() {
  WiFi.mode(WIFI_STA);
  WiFi.begin(ssid, password);
  Serial.print("Connecting to WiFi ..");
  while (WiFi.status() != WL_CONNECTED) {
    Serial.print('.');
    delay(1000);
  }
  Serial.println(WiFi.localIP());
}
```

Proměnné `ssid` a `password` obsahují SSID a heslo sítě, ke které se chcete připojit.

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – stav připojení

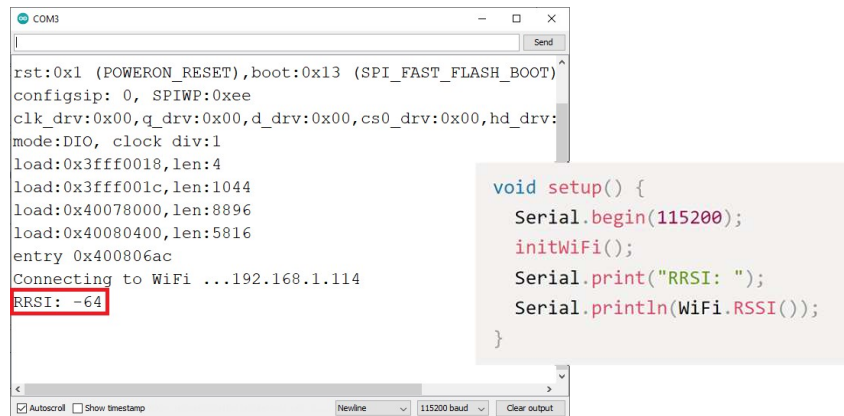
Chcete-li zjistit stav připojení Wi-Fi, můžete použít `WiFi.status()`. Vráťí jednu z následujících hodnot, které odpovídají konstantám v tabulce:

Hodnota	Konstanta	Význam
0	WL_IDLE_STATUS	dočasný stav přiřazený při volání <code>WiFi.begin()</code>
1	WL_NO_SSID_AVAIL	není k dispozici žádný identifikátor SSID
2	WL_SCAN_COMPLETED	prohledávání sítě je dokončeno
3	WL_CONNECTED	při připojení k síti Wi-Fi
4	WL_CONNECT_FAILED	pokud se připojení nezdaří
5	WL_CONNECTION_LOST	při ztrátě připojení
6	WL_DISCONNECTED	při odpojení od sítě

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – síla signálu

Chceme-li získat sílu připojení WiFi, můžeme jednoduše zavolat `WiFi.RSSI()` po připojení k WiFi.



The screenshot shows a serial monitor window titled 'COM3' with a 'Send' button. The output text includes boot parameters and a red box highlighting 'RSSI: -64'. To the right, a code snippet is shown in a light gray box.

```

rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
config: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:
mode:DIO, clock div:1
load:0x3fff0018,len:4
load:0x3fff001c,len:1044
load:0x40078000,len:8896
load:0x40080400,len:5816
entry 0x400806ac
Connecting to WiFi ...192.168.1.114
RSSI: -64
  
```

```

void setup() {
  Serial.begin(115200);
  initWiFi();
  Serial.print("RSSI: ");
  Serial.println(WiFi.RSSI());
}
  
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – IP adresa

Když je ESP32 nastaven jako Wi-Fi stanice, může se připojit k jiným sítím (například k routeru). V tomto scénáři směrovač přiřadí jedinečnou adresu IP desce ESP32. Chceme-li získat tuto IP adresu, musíme po navázání spojení se sítí volat:

```
Serial.println(WiFi.localIP());
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – IP adresa

Nastavení pevné IP adresy:

```
// Set your Static IP address
IPAddress local_IP(192, 168, 1, 184);
// Set your Gateway IP address
IPAddress gateway(192, 168, 1, 1);

IPAddress subnet(255, 255, 0, 0);
IPAddress primaryDNS(8, 8, 8, 8); // optional
IPAddress secondaryDNS(8, 8, 4, 4); // optional
```

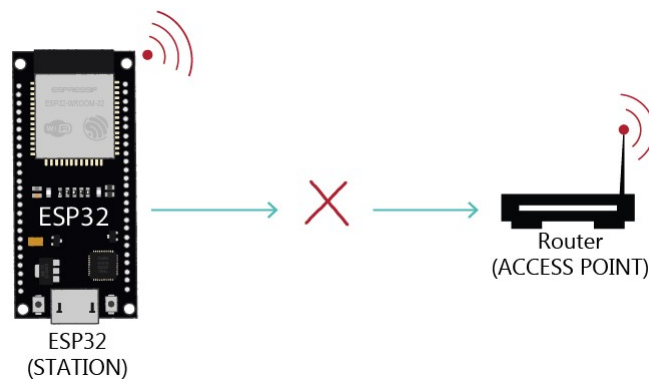
Poté v `setup()` voláme `WiFi.config()` pro přiřazení konfigurací k ESP32.

```
// Configures static IP address
if (!WiFi.config(local_IP, gateway, subnet, primaryDNS, secondaryDNS)) {
  Serial.println("STA Failed to configure");
}
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – odpojení od sítě

Chcete-li se odpojit od dříve připojené sítě Wi-Fi, použijeme `WiFi.disconnect()`



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – znovupřipojení k síti

Chceme-li se po ztrátě připojení znovu připojit k síti Wi-Fi, můžeme se pomocí funkce `WiFi.reconnect()` pokusit znovu připojit k dříve připojenému přístupovému bodu.

Nebo můžeme zavolat

```
WiFi.disconnect();
WiFi.begin(ssid, password);
```

Případně se můžeme také pokusit restartovat ESP32 pomocí `ESP.restart()` při ztrátě připojení.

```
unsigned long currentMillis = millis();
// if WiFi is down, try reconnecting
if ((WiFi.status() != WL_CONNECTED) && (currentMillis - previousMillis >= interval)) {
  Serial.print(millis());
  Serial.println("Reconnecting to WiFi...");
  WiFi.disconnect();
  WiFi.reconnect();
  previousMillis = currentMillis;
}
```

```
unsigned long previousMillis = 0;
unsigned long interval = 30000;
```

Deklarace proměnné *Millis* a *interval*. Interval odpovídá době mezi jednotlivými kontrolami v milisekundách (například 30 sekund).

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – Multi SSID

ESP32 WiFiMulti umožňuje registrovat více sítí (kombinace SSID/hesla). ESP32 se připojí k síti Wi-Fi s nejsilnějším signálem (RSSI). Pokud dojde ke ztrátě připojení, připojí se k další síti v seznamu. To vyžaduje zahrnutí knihovny WiFiMulti.h (ve výchozím nastavení je dodávána s balíčkem ESP32).

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – Název WiFi

Pro nastavení vlastního názvu sítě WiFi využijeme

`WiFi.setHostname` (`VASE_NOVE_JMENO_SITE`) před voláním `WiFi.begin()`

Výchozí název sítě ESP32 je **espressif**.



Nejprve začneme definováním nového názvu.

```
String hostname = "ESP32 Node Temperature";
```

Potom zavolejte funkci `WiFi.setHostname()` před voláním `WiFi.begin()`.

```
WiFi.config(INADDR_NONE, INADDR_NONE, INADDR_NONE, INADDR_NONE);
WiFi.setHostname(hostname.c_str()); //define hostname
```

Po restartu routeru:



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32

...klasika nebo

ESP NOW

ESP-NOW je "protokol vyvinutý společností Espressif," který umožňuje více zařízením komunikovat mezi sebou bez použití Wi-Fi infrastruktury. Protokol je podobný bezdrátovému připojení 2,4 GHz s nízkou spotřebou (...). Párování mezi zařízeními je nutné před jejich komunikací. Po dokončení párování je připojení bezpečné a peer-to-peer.

To znamená, že po vzájemném spárování zařízení je připojení trvalé. Jinými slovy, pokud náhle jedna z vašich desek ztratí napájení nebo se resetuje, po restartování se automaticky připojí ke svému vrstevníkovi a bude pokračovat v komunikaci.



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

Zdroj: <https://randomnerdtutorials.com/esp-now-esp32-arduino-ide/>

WiFi s ESP32



ESP-NOW podporuje následující funkce:

- Šifrovaná a nešifrovaná jednosměrná komunikace
- Smíšená šifrovaná a nešifrovaná komunikace
- Lze posílat až 250-bajtové zprávy;
- Funkce odesílání zpětného volání, kterou lze nastavit tak, aby informovala aplikační vrstvu o úspěchu nebo selhání přenosu.

Technologie ESP-NOW má také následující omezení:

- V režimu stanice podporováno maximálně 10 šifrovaných partnerských zařízení; 6 maximálně v režimu SoftAP nebo SoftAP + Station
- Max. 20 nešifrovaných partnerů v síti

Jednoduše řečeno, ESP-NOW je rychlý komunikační protokol, který lze použít k výměně malých zpráv (až 250 bajtů) mezi deskami ESP32.

ESP-NOW je velmi univerzální, s jednosměrnou nebo obousměrnou komunikací v různých nastaveních.

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Komunikace v jednom směru

Odesílání dat z jedné desky na druhou, jako jsou odečty senzorů nebo příkazy ON a OFF pro řízení GPIO.



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Komunikace od jednoho zdroje „master“ k mnoha cílům „slaves“

Jedno ESP32 odesílá stejné nebo různé příkazy na různá ESP32. Tato konfigurace je ideální pro vytvoření např. dálkového ovládání = několik ESP32 ovládaných jediným ESP32 kitem.



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Vícezdrojová komunikace k jednomu cíli

Tato konfigurace je ideální, pokud chcete shromažďovat data z několika uzlů senzorů do jednoho ESP32. To lze nakonfigurovat jako webový server pro zobrazení dat ze všech ostatních zdrojů.



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Obousměrná komunikace

S ESP-NOW může být každé ESP32 současně odesílatelem a příjemcem. Můžete tedy navázat obousměrnou komunikaci mezi kity.



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Obousměrná komunikace

S ESP-NOW může být každé ESP32 současně odesílatelem a příjemcem. Můžete tedy navázat obousměrnou komunikaci mezi kity. Do této konfigurace můžete přidat další ESP32 a mít něco, co vypadá jako síť (všechny ESP32 spolu komunikují).



Stručně řečeno, ESP-NOW je ideální pro vybudování sítě, ve které můžete mít několik ESP32, která si navzájem vyměňují data.

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32

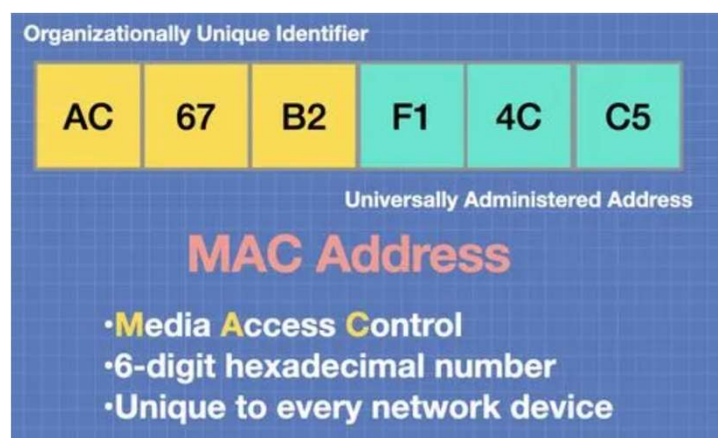


ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



MAC adresa



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



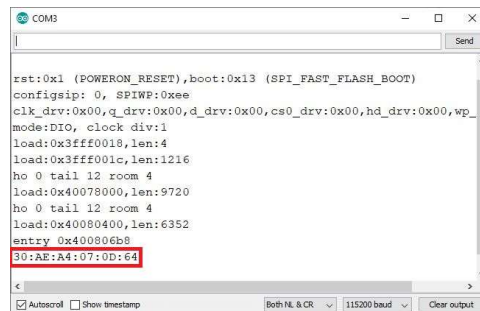
Získání MAC adresy

Pro komunikaci prostřednictvím ESP-NOW musíme znát MAC adresu přijímače ESP32. Každé ESP32 má jedinečnou MAC adresu a tak identifikujeme každou desku, abychom na ni mohli odesílat data.

```
#include "WiFi.h"

void setup(){
  Serial.begin(115200);
  WiFi.mode(WIFI_MODE_STA);
  Serial.println(WiFi.macAddress());
}

void loop(){
}
```



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Jedno ESP32 bude "odesílatelem" a druhé ESP32 bude "příjemcem".



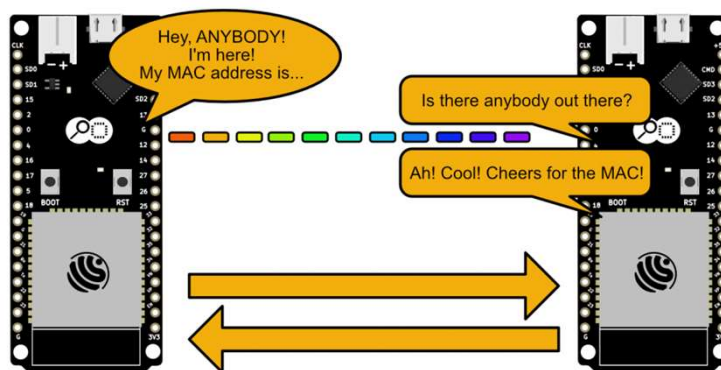
ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Jedno ESP32 bude "odesílatelem" a druhé ESP32 bude "příjemcem".



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Jedno ESP32 bude "odesílatelem" a druhé ESP32 bude "příjemcem".

Odešleme strukturu, která obsahuje proměnnou typu *char*, *int*, *float* a *boolean*. Poté můžete upravit strukturu tak, aby odesílala libovolné typy proměnných, které jsou vhodné pro náš projekt (například odečty senzorů nebo logické proměnné pro zapnutí nebo vypnutí něčeho).

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Pozn. "odesílatel" ESP32 #1 a "příjemce" je ESP32 #2.

V programu odesílatele budeme provádět následující kroky:

- Inicializace ESP-NOW
- Vytvoření funkce OnDataSent = bude spuštěna při odesílání zprávy. Vrací informaci o doručení nebo nedoručení zprávy.
- Přidání příjemce. K tomu potřebujeme znát jeho MAC adresu.
- Odešleme zprávu příjemci.

Na straně příjemce by měl program obsahovat kroky:

- Inicializace ESP-NOW
- Vytvoření funkce OnDataRecv = bude spuštěna po přijetí zprávy. Vrací potvrzení o přijetí zprávy.

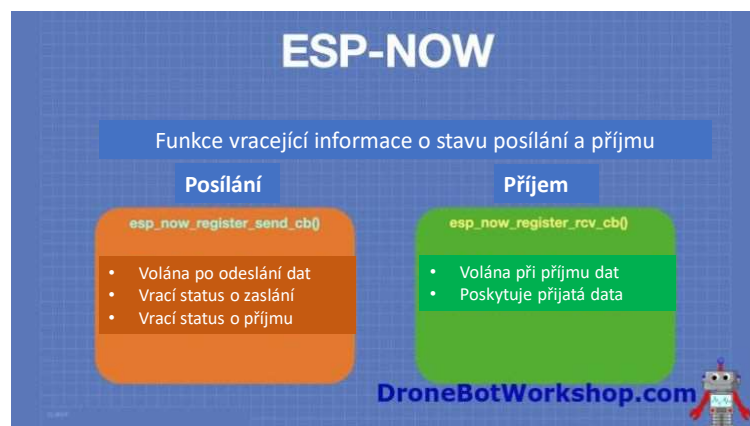
ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Funkce pro ověření, zda zpráva byla odeslána a přijata v pořádku



Zdroj: <https://dronebotworkshop.com/esp-now/>

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Odesílání zprávy



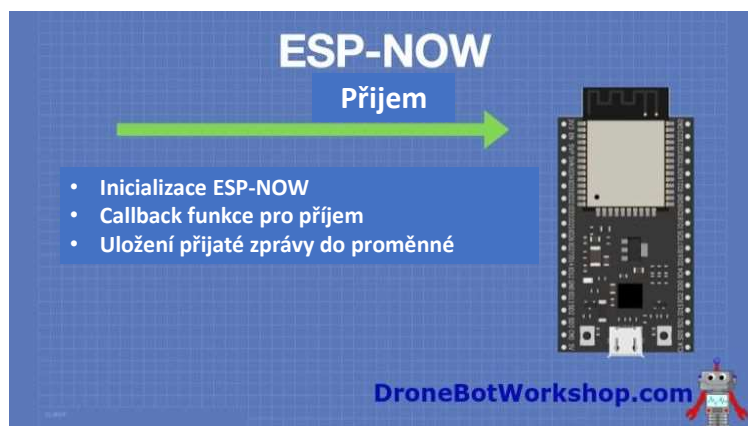
ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Jednosměrná komunikace P2P (Point-to-point)

Příjem zprávy



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32



Název a popis funkce

`esp_now_init()` Inicializuje ESP-NOW. Před inicializací ESP-NOW je nutné inicializovat Wi-Fi.

`esp_now_add_peer()` Zavolejte tuto funkci pro spárování zařízení a předejte jako argument adresu MAC.

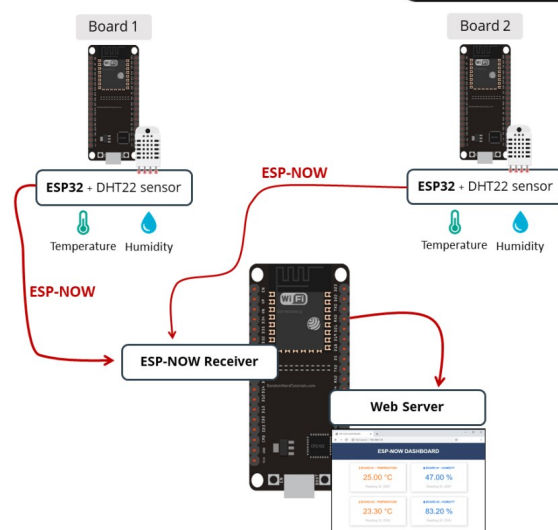
`esp_now_send()` Odesíláte data pomocí ESP-NOW.

`esp_now_register_send_cb()` Zaregistrujte funkci zpětného volání, která se spustí při odeslání dat. Při odeslání zprávy je volána funkce – tato funkce vrátí, zda bylo doručení úspěšné nebo ne.

`esp_now_register_rcv_cb()` Zaregistrujte funkci zpětného volání, která se spustí při příjmu dat. Při příjmu dat prostřednictvím ESP-NOW je volána funkce.

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

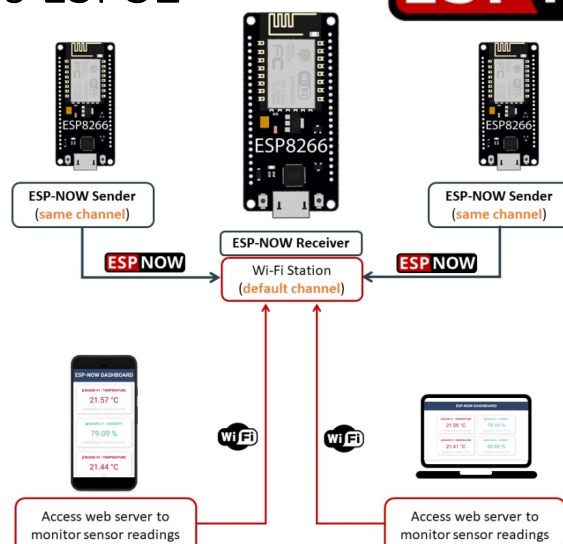
WiFi s ESP32



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32

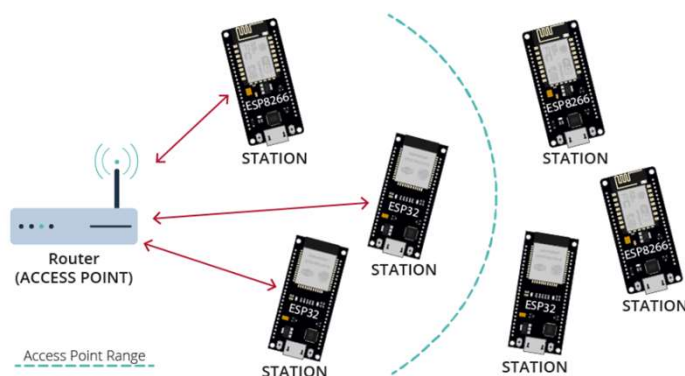
ESP NOW



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – „klasika“

V tradiční architektuře Wi-Fi sítě je jeden uzel (přístupový bod – obvykle router) připojen ke všem ostatním uzlům (stanicím). Každý uzel může vzájemně komunikovat pomocí přístupového bodu. **To je však omezeno na pokrytí wi-fi přístupového bodu. Každá stanice musí být v dosahu, aby se připojila přímo k přístupovému bodu.**

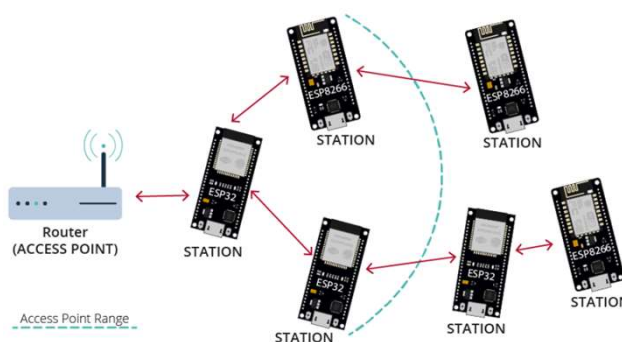


ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

<https://randomerdutorials.com/esp-mesh-esp32-esp8266-painlessmesh/>

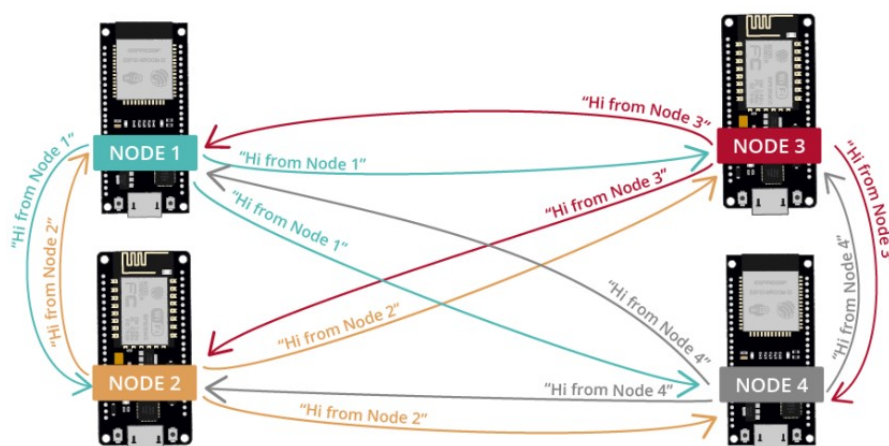
WiFi s ESP32 – ESP MESH

S ESP-MESH se uzly nemusí připojovat k centrálnímu uzlu. Uzly jsou zodpovědné za vzájemné předávání přenosů. To umožňuje více zařízením rozložit se na velkou fyzickou plochu. Uzly se mohou samy organizovat a dynamicky spolu komunikovat, aby zajistily, že paket dosáhne svého konečného cíle uzlu. Pokud je některý uzel odstraněn ze sítě, je schopen se sám organizovat, aby se ujistil, že pakety dosáhnou svého cíle.



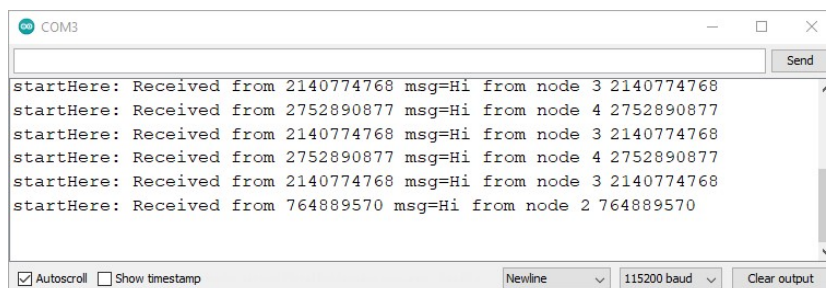
ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH



```

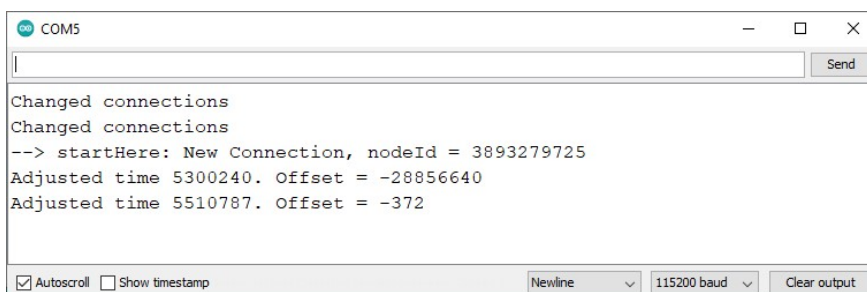
startHere: Received from 2140774768 msg=Hi from node 3 2140774768
startHere: Received from 2752890877 msg=Hi from node 4 2752890877
startHere: Received from 2140774768 msg=Hi from node 3 2140774768
startHere: Received from 2752890877 msg=Hi from node 4 2752890877
startHere: Received from 2140774768 msg=Hi from node 3 2140774768
startHere: Received from 764889570 msg=Hi from node 2 764889570
  
```

Výpis sériového monitoru z Node 1

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH

Když dojde ke změnám v síti, např. když jedna deska s ESP32 opustí nebo se připojí k síti.



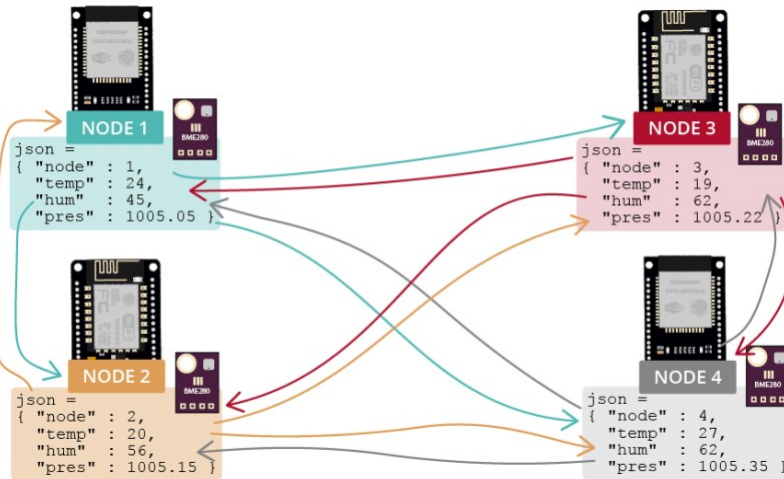
```

Changed connections
Changed connections
--> startHere: New Connection, nodeId = 3893279725
Adjusted time 5300240. Offset = -28856640
Adjusted time 5510787. Offset = -372
  
```

ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH

Senzorová síť s BME280



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH

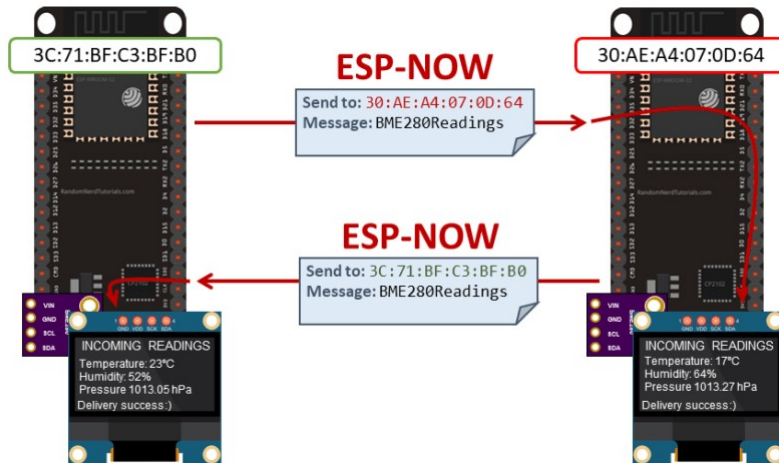
Senzorová síť s BME280

```

COM3
Node: 4
Temperature: 21.01 C
Humidity: 95.06 %
Pressure:1013.54hpa
Changed connections
Received from 2140774768 msg={ "node":3,"temp":20.389999389648438,"hum":71.609375,"pres":1013.549072265625}
Node: 3
Temperature: 20.39 C
Humidity: 71.61 %
Pressure:1013.54hpa
Received from 764889570 msg={ "node":2,"temp":19.700000762939453,"hum":65.2353515625,"pres":1012.779541015625}
Node: 2
Temperature: 19.70 C
Humidity: 65.24 %
Pressure:1012.77hpa
Received from 2752890877 msg={ "node":4,"temp":21,"hum":94.5439453125,"pres":1013.8912353515625}
Node: 4
Temperature: 21.00 C
Humidity: 94.54 %
Pressure:1013.89hpa
Received from 2140774768 msg={ "node":3,"temp":20.360000610351562,"hum":72.0869140625,"pres":1013.5477905273438}
Node: 3
Temperature: 20.36 C
Humidity: 72.09 %
Pressure:1013.54hpa
Autoscroll Show timestamp
Newline 115200 baud Clear output
  
```


WiFi s ESP32 – ESP MESH

Senzorová síť s BME280



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

Zdroj: <https://randomnerdtutorials.com/esp-now-two-way-communication-esp32/>

WiFi s ESP32 – ESP MESH

Senzorová síť s BME280

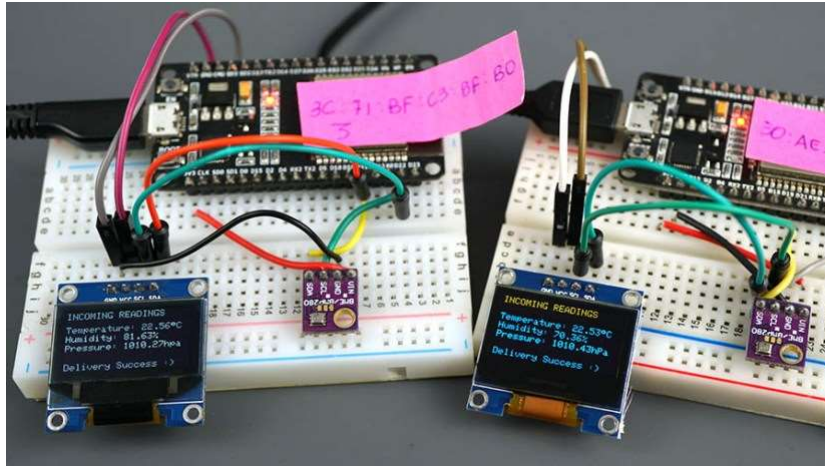
```
// Callback when data is sent
void OnDataSent(const uint8_t *mac_addr, esp_now_send_status_t status) {
    Serial.print("\r\nLast Packet Send Status:\t");
    Serial.println(status == ESP_NOW_SEND_SUCCESS ? "Delivery Success" : "Delivery Fail");
    if (status == 0) {
        success = "Delivery Success :>";
    }
    else {
        success = "Delivery Fail :(";
    }
}

// Callback when data is received
void OnDataRecv(const uint8_t * mac, const uint8_t *incomingData, int len) {
    memcpy(&incomingReadings, incomingData, sizeof(incomingReadings));
    Serial.print("Bytes received: ");
    Serial.println(len);
    incomingTemp = incomingReadings.temp;
    incomingHum = incomingReadings.hum;
    incomingPres = incomingReadings.pres;
}
```

Zdroj: <https://randomnerdtutorials.com/esp-now-two-way-communication-esp32/>

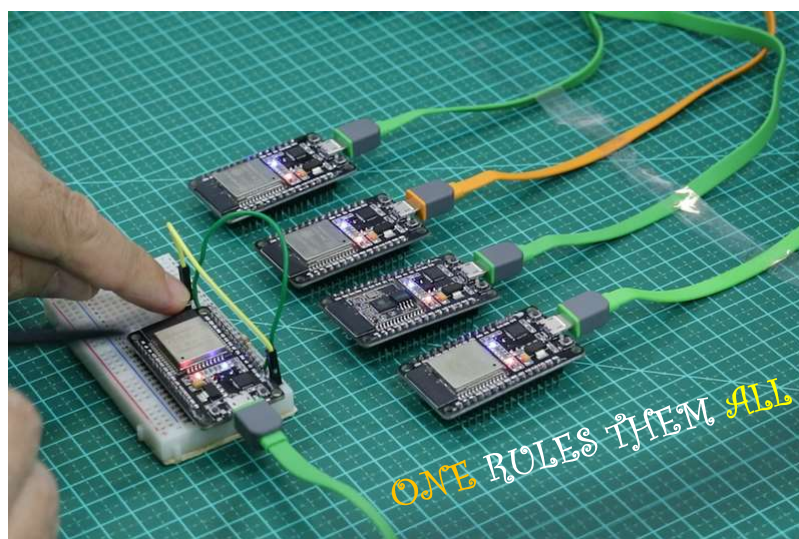
WiFi s ESP32 – ESP MESH

Senzorová síť s BME280



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32

WiFi s ESP32 – ESP MESH



ELEKTRONIKA A KOMUNIKACE PRAKTICKY / P02 / WiFi s ESP32