

Zadání maturitní práce s obhajobou

**Číslo a název**:13 - Detekce přítomnosti

**Jméno žáka**: Gabriel-Max Hlásný

**Konzultant**: Ing. Luděk Fedurca

**Oponent**: Bc. Jakub Malý

**Datum zadání**:2. prosinec 2022

**Datum odevzdání**:1. dubna 2023

**Doba obhajoby**:15 minut

**Zadání:**

Implementujte detektor přítomnosti osob na základě dostupnosti mobilního telefonu/nositelné elektroniky připojené k Wi-Fi síti s pevně definovanou IP adresou vázanou na MAC adresu zařízení.  
Detekce budou ukládány do externího systému dostupného přes webové rozhraní.

**Způsob zpracování:**

* **Tištěná forma**: rozsah dokumentace 15 – 20 stran textu; v obálce s chlopněmi nebo pevná vazba; součástí práce bude úvodní obálka, zadání práce, harmonogram a prohlášení o souhlasu se zadáním práce, samostatnosti zpracování práce a použitím legálního software
* **Digitální forma**: kopie práce a pracovní soubory, dokumentace, prezentace na přiloženém CD nebo DVD v papírové obálce s jednoduchým HTML rozcestníkem; soubory v alternativních formátech
* **Model projektu**: vytvořte funkční model, který bude simulovat zadání

**Počet vyhotovení:** 1

**Formální úprava práce:**  
 **Písmo:** velikost 12  
 **Font:** Calibri, Arial nebo Times New Roman (zvolený font dodržte v celé práci)  
 **Řádkování:** 1,5  
 **Vzdálenost mezi odstavci:** 6 b.  
 **Okraje:** horní a dolní 25 mm, levý (vnitřní) 40 mm, pravý (vnější) 20 mm  
 **Zarovnání odstavce:** do bloku  
 **Číslování stránek:** vpravo dolu  
 **Začátek hlavní kapitoly:** vždy na nové straně  
 Dodržení typografických pravidel hladké sazby

**Hodnocení:**

1. Splnění zadání
2. Plnění plánu práce a účast na konzultacích
3. Aktuálnost a přínosnost tématu
4. Odborná úroveň práce, kvalita zpracování práce, použité prostředky
5. Zpracování dokumentace – typografie, zdroje, struktura, rozsah…
6. Dodržení ČSN ISO 690 a ČSN ISO 690-2 – bibliografické citace dokumentů
7. Hodnocení modelu a prezentace

**Podpis žáka**  …………………………………………………………………………….

**Podpis konzultanta** …………………………………………………………………………….

**Podpis ředitele školy** …………………………………………………………………………….

| **Harmonogram práce MZ** |
| --- |

| **Třída:** 4. I |
| --- |
| **Studijní obor:** 18-20-M/01 Informační technologie |
| **Jméno studenta:** Gabriel-Max Hlásný |
| **Konzultant:** Ing. Luděk Fedurca |
| **Číslo a název úlohy:** 13 - Detekce přítomnosti |

| **Plán práce** | |
| --- | --- |
| **Týden** | **Práce** |
| 16.1.-21.1 2023 | osnova závěrečné práce, náplň kapitol prezentace prototypu |
| 6.2.-10.2 2023 | prezentace funkčnosti minimálního prototypu, 50% dokumentace |
| 27.2.-3.3. 2023 | cvičná obhajoba, finální text k revizi |

| **Ve Štětí dne ………………………………………** |
| --- |
| **Podpis ………………………………………** |
|  |

| **Kontrola plnění plánu práce** | | |
| --- | --- | --- |
| **Datum** | **Poznámky** | **Podpis** |
|  |  |  |
|  |  |  |
|  |  |  |

**Prohlášení**

**Třída:** 4. I

**Studijní obor:** 18-20-M/01 Informační technologie

**Jméno žáka:** Gabriel-Max Hlásný

**Konzultant:** Ing. Luděk Fedurca

**Číslo a název úlohy:** 13 - Detekce přítomnosti

***Čestné prohlášení o souhlasu se zadáním maturitní práce***

Prohlašuji, že jsem se seznámil s obsahem zadání maturitní práce. Souhlasím se zadaným tématem.[[1]](#footnote-0)

Ve Štětí dne …………………………………… Podpis: ………………………………

***Čestné prohlášení o samostatnosti zpracování maturitní práce***

Prohlašuji, že jsem odevzdanou maturitní práci vypracoval samostatně a uvedl jsem všechny použité zdroje. Uvědomuji si, že prokáže-li se opak, může být má práce hodnocena jako nedostatečná.

Ve Štětí dne …………………………………… Podpis: ………………………………

***Čestné prohlášení o použití legálního softwarového vybavení***

Prohlašuji, že veškeré programové vybavení, které bylo použito při řešení této maturitní práce, bylo užito v souladu s jeho licencí.

Ve Štětí dne …………………………………… Podpis: ……………………………………

[**Anotace 6**](#_j7kt9n3hy9rw)

[**Konfigurace sítě 8**](#_scai1wk9lbhi)

[**Topologie sítí 8**](#_ft9cbfm6j2cs)

[**Rozsah privátních adres 9**](#_h3n1125ghtn2)

[**Network Address Translation (NAT) 10**](#_4h86id6qmk7a)

[**Server DHCP (Dynamic Host Configuration Protocol server) 10**](#_bzdp2mei9uvb)

[Alternativní možnosti 10](#_af435cxhd71s)

[**Prezentace kódu 12**](#_k355fum8mxoc)

[**Co je to M5StickCPlus? 12**](#_8xcbbisecma)

[**O knihovně M5StickCPlus 12**](#_x03sfxges6ie)

[Knihovny 12](#_15qp27sxq6gm)

[Připojení WiFi a propojení s ThingSpeak 13](#_yt18lfno3kzo)

[Co je to ThingSpeak? 13](#_17jmqxjdw90a)

[Propojení s ThingSpeakem 14](#_7rsmth89j4x9)

[**Kód pro pingování zařízení 19**](#_aebopmm17sqw)

[**Zdroje 24**](#_iq3o9u5f4xj7)

Implementujte detektor přítomnosti osob na základě dostupnosti mobilního telefonu/nositelné elektroniky připojené k Wi-Fi síti s pevně definovanou IP adresou vázanou na MAC adresu zařízení.  
Detekce budou ukládány do externího systému dostupného přes webové rozhraní.  
  
AnotaceLF

Projekt detektoru přítomnosti využívá k fungování domácí Wi-Fi síť a dedikované zařízení jako monitorovací sondu. Hardwarová část se skládá z programovatelného mikrořadiče s podporou Wi-Fi a domácího routeru s patřičně upravenou konfigurací DHCP serveru (MAC static lease).  
 Detektor testuje přítomnost zařízení v síti periodickým vysíláním ICMP pingu za adresu monitorovaného zařízení a výsledek testu ukládá do cloudové služby ThingSpeak.

# Anotace

Projekt detektoru přítomnosti využívá k fungování domácí Wi-Fi síť a dedikované zařízení jako monitorovací sondu. Hardwarová část se skládá z programovatelného mikrořadiče s podporou Wi-Fi a domácího routeru s patřičně upravenou konfigurací DHCP serveru (MAC static lease).  
 Detektor testuje přítomnost zařízení v síti periodickým vysíláním ICMP pingu za adresu monitorovaného zařízení a výsledek testu ukládá do cloudové služby ThingSpeak.

Kód je napsán v jazyce C++ pro platformu Arduino, konkrétně pro zařízení M5StickCPlus. Zařízení M5StickCPlus se připojí k síti Wi-Fi pomocí SSID a hesla zadaného v kódu a poté pomocí knihovny ESP32Ping pinguje vzdálenou IP adresu konkrétního mobilního zařízení (192.168.0.98) a zjišťuje, zda je k dispozici. Pokud je ping úspěšný, na obrazovce zařízení se vypíše "přítomen" nebo "nepřítomen" podle toho, zda je mobilní zařízení ve stejné síti. Kód také na obrazovku vypíše místní IP a MAC adresu zařízení. Funkce smyčky se opakuje s pravidelnou prodlevou mezi iteracemi.

Dále zařízení M5StickCPlus založené na systému ESP32 se připojuje k síti Wi-Fi a používá jej k pravidelnému odesílání údajů o síle signálu Wi-Fi do ThingSpeaku a cloudové platformy pro zařízení internetu věcí. Kód měří sílu signálu Wi-Fi tak, že provede několik měření a zprůměruje je, a poté odešle data do ThingSpeak prostřednictvím požadavků HTTP POST. LED dioda se rozbliká, která signalizuje úspěšné stisknutí tlačítka a HTTP POST. Program také obsahuje funkci pro připojení k síti Wi-Fi, funkci pro odeslání požadavku HTTP do ThingSpeak a funkci pro blikání LED.

# Konfigurace sítě

## Topologie sítí

Topologie sítí se zabývá zapojením různých prvků do počítačových sítí a zachycením jejich skutečné (reálné) a logické (virtuální) podoby (datové linky, síťové uzly).  
  
Fyzická topologie popisuje reálnou konstrukci sítě, jednotlivé uzly a fyzicky zapojená zařízení a jejich umístění včetně instalovaných kabelů, přesného umístění uzlů a přípojek mezi nimi (např. UTP).

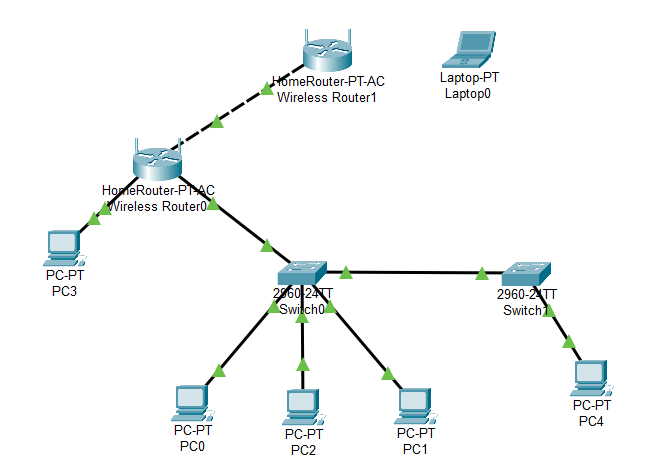
Logická topologie se vztahuje k tomu, jak jsou data v síti přenášena a kudy protékají z jednoho zařízení do druhého. Nemusí nutně kopírovat fyzické schéma sítě.  
  
Běžné topologie a jejich dělení

Fyzická topologie

dvoubodové spoje:

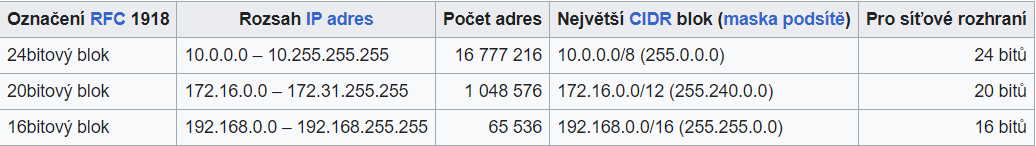
* kruh
* hvězda
* strom

V projektu využívám Hvězdicovou topologii  
Příklad zde:



## Rozsah privátních adres

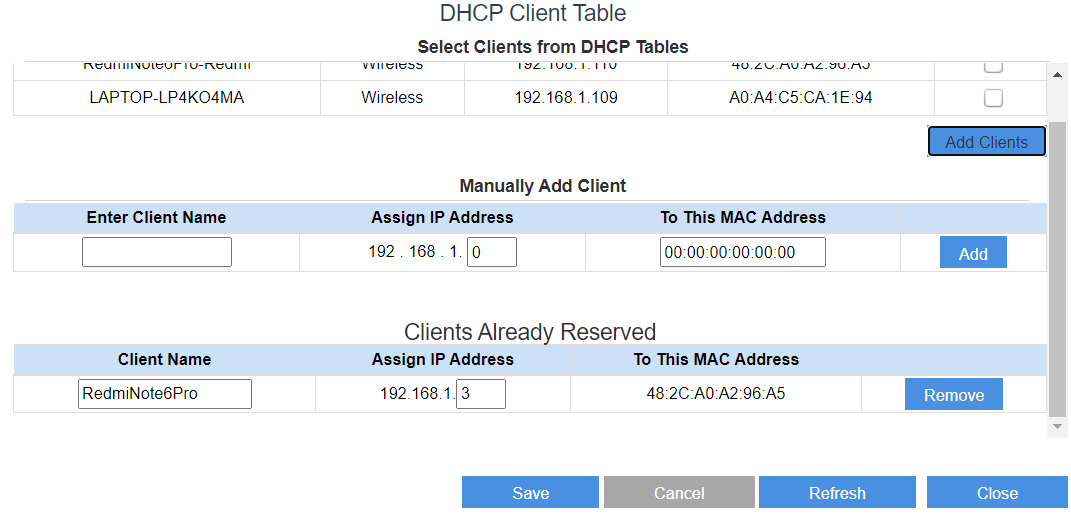
Internet Engineering Task Force (IETF) přikázala IANA zachování následujících rozsahů IPv4 adres pro  
soukromé sítě



Tyto IP adresy jsou určeny pro použití v uzavřených sítích, jako jsou domácí nebo firemní sítě, a nelze je směrovat přes Internet. K tomu potřebujeme znát NAT a jak funguje.

Router na kterém je projekt připojen používá 16 bitový blok a je v podsíti 192.168.0.0/24

Zařízení má přidělenou statickou ip adresu níže v obrázku



## Network Address Translation (NAT)

Network Address Translation (NAT) je technika používaná k mapování více zařízení v soukromé síti na jednu veřejnou IP adresu, která se používá na internetu. To umožňuje zařízením v soukromé síti sdílet jedno připojení k internetu, aniž by každé zařízení muselo mít vlastní veřejnou IP adresu. NAT funguje tak, že překládá soukromé IP adresy zařízení v místní síti na veřejnou IP adresu přidělenou směrovači, který připojuje místní síť k internetu. Když zařízení v místní síti odešle požadavek na Internet, je tento požadavek odeslán na veřejnou IP adresu směrovače, který pak požadavek předá příslušnému zařízení v místní síti na základě informací v překladové tabulce. NAT se běžně používá v domácích a malých kancelářských sítích.

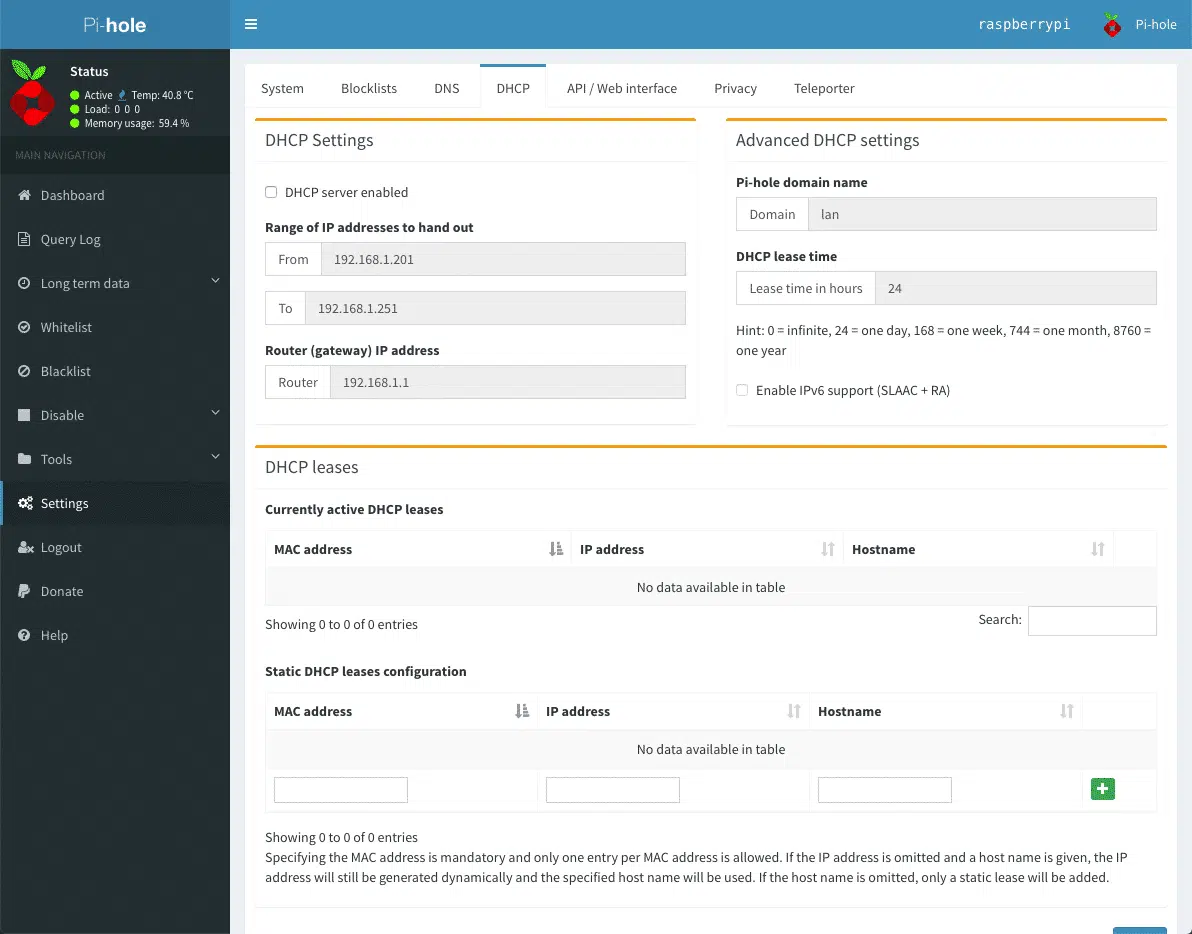
## Server DHCP (Dynamic Host Configuration Protocol server)

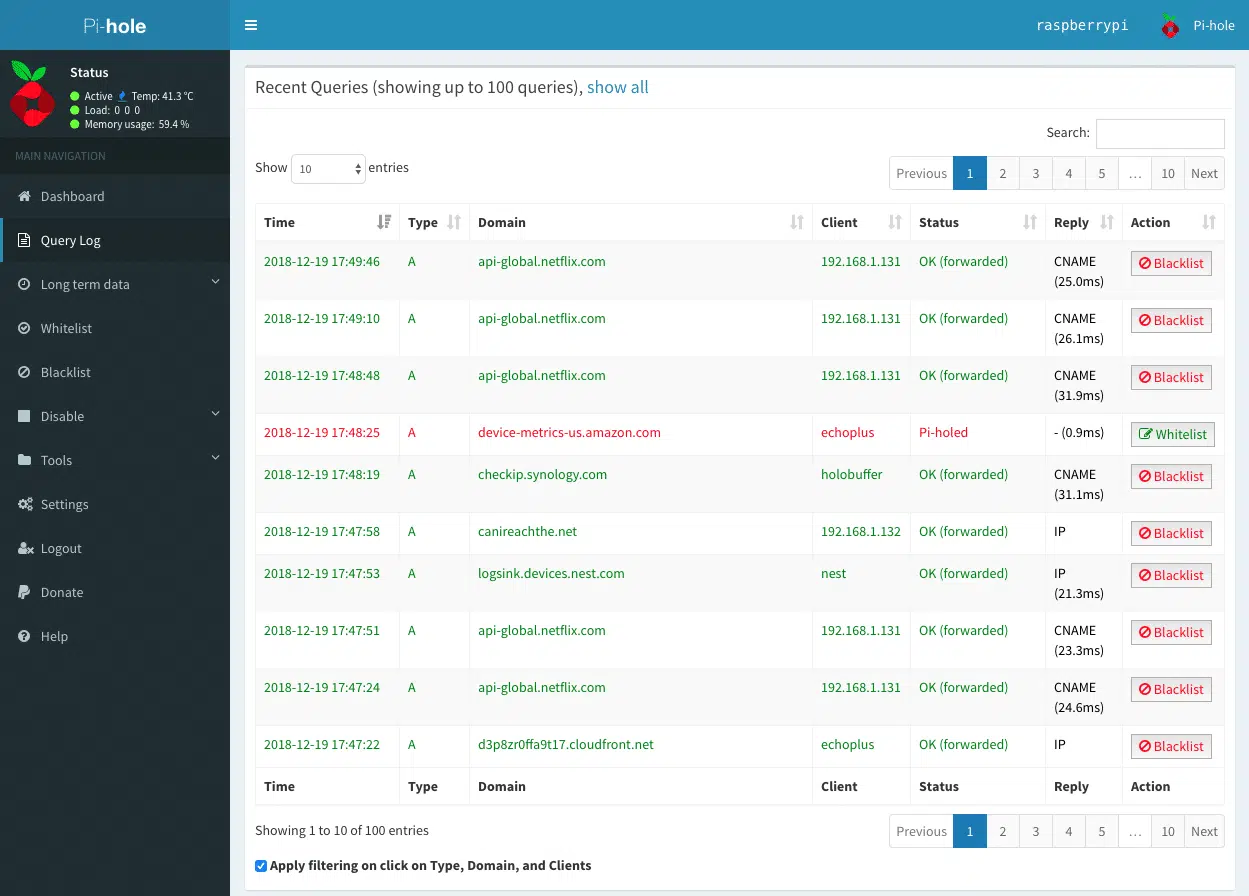
Server DHCP (Dynamic Host Configuration Protocol server) je síťový server, který automaticky přiděluje a spravuje IP adresy a další informace o konfiguraci sítě zařízením v síti. Server DHCP odstraňuje potřebu správců sítě ručně přidělovat IP adresy každému zařízení v síti a umožňuje zařízením snadné připojení a vzájemnou komunikaci. Servery DHCP lze implementovat na různých hardwarových zařízeních a softwarových platformách.

V mém případě používám Statickou ip adresu (nastavenou manuálně).

## Alternativní možnosti

Pi-hole je bezplatná aplikace s otevřeným zdrojovým kódem pro blokování reklam a internetových trackerů na síťové úrovni. Funguje jako "díra" v systému DNS (Domain Name System) tím, že zařazuje na Blacklist a blokuje požadavky na konkrétní domény, čímž zabraňuje načítání reklam a sledovacích služeb. Pi-hole se obvykle instaluje na Raspberry Pi nebo jiný počítač se systémem Linux a lze jej použít k blokování reklam a sledovacích služeb pro všechna zařízení v síti, včetně počítačů, chytrých telefonů a chytrých televizorů.





# Prezentace kódu

## Co je to M5StickCPlus?

M5StickCPlus je malé zařízení založené na mikrokontroléru ESP32 od společnosti M5Stack. Je velké jen několik centimetrů a obsahuje řadu senzorů a periferií, jako jsou akcelerometr, gyroskop, barometr, teploměr, světelný senzor a tlačítka, ke kterým lze připojit další zařízení pro měření. Je kompatibilní s Arduino IDE a dalšími frameworky pro vývoj projektů IoT. M5StickCPlus je vhodný pro vývoj a prototypování různých projektů, jako jsou zařízení chytré domácnosti, bezdrátové senzory, chytré náramky a další.

## O knihovně M5StickCPlus

Knihovna M5StickCPlus je speciální knihovna pro vývojovou desku M5StickC Plus, která poskytuje řadu funkcí a metod pro ovládání všech periferií desky, včetně displeje, tlačítek, senzorů a modulu WiFi. Knihovna obsahuje mnoho příkladů kódu, které demonstrují různé funkce desky a umožňují vývojářům snadno začít s programováním desky. M5StickC Plus je malá a přenosná vývojová deska založená na čipu ESP32, která je ideální pro vývoj různých projektů internetu věcí a elektroniky.

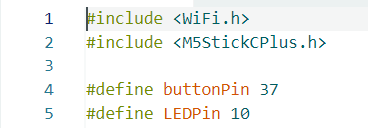
## Knihovny

Do kódu musíme zahrnout potřebné knihovny (soubory) které využívá pro funkci kódu.

Knihovny:

* WiFi
* M5StickCPlus
* secret.h (soubor obsahující data které by neměli být viděny veřejně např. hesla)

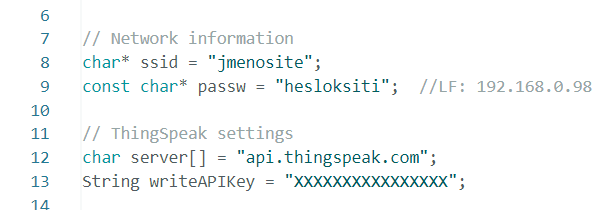
Také musíme definovat tlačítka a LED žárovku na M5StickCPlus do určitých pinnů



## Připojení WiFi a propojení s ThingSpeak

Zařízení se musí připojit k síti WiFi v tomhle případě ssid = název sítě

a heslo k síti



## Co je to ThingSpeak?

ThingSpeak je open-source platforma IoT, která uživatelům umožňuje shromažďovat, analyzovat a vizualizovat data ze senzorů v reálném čase. Poskytuje snadno použitelné rozhraní API pro odesílání a přijímání dat do a ze zařízení IoT. ThingSpeak nabízí vestavěné integrace s různými dalšími platformami a službami IoT, včetně MATLAB Analytics, IFTTT a Twitter. Uživatelé mohou také vytvořit vlastní webové aplikace pro zobrazení a analýzu svých dat a mohou svá data sdílet s ostatními. ThingSpeak lze použít pro širokou škálu aplikací, například pro sledování povětrnostních podmínek, sledování kvality ovzduší a ovládání systémů domácí automatizace.

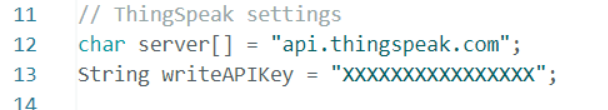
## Propojení s ThingSpeakem

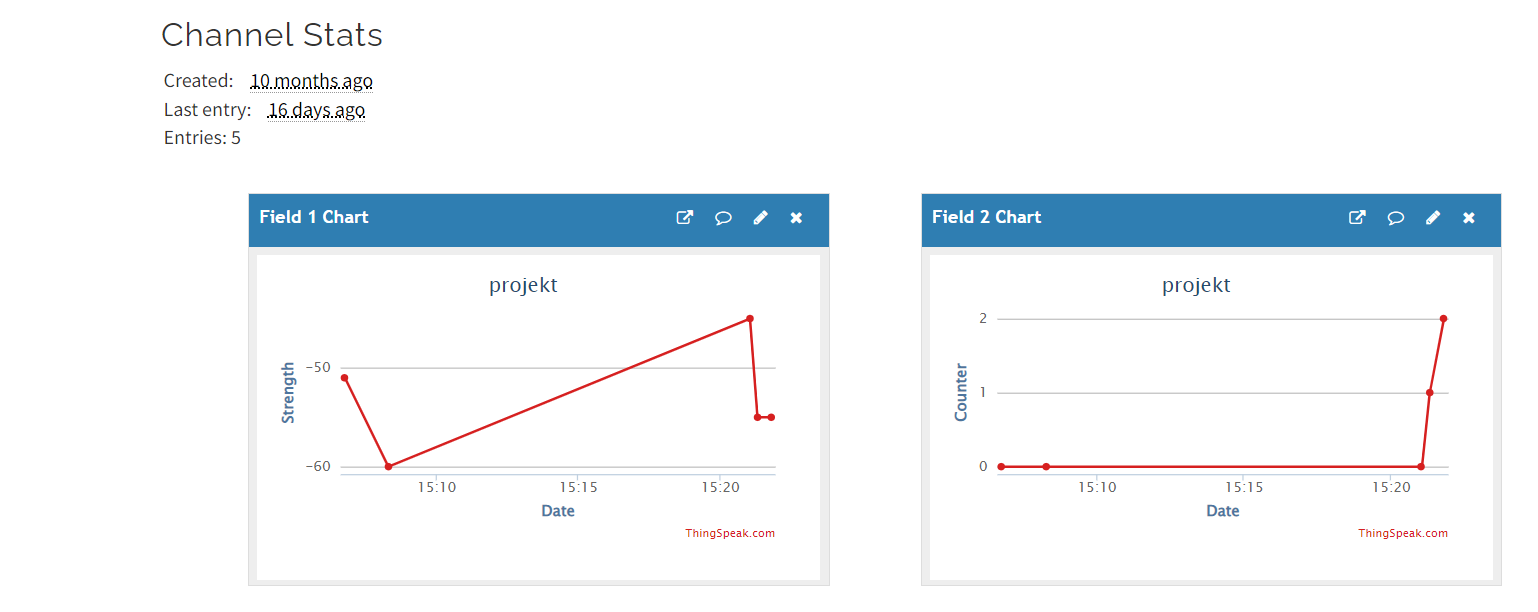
Kód vidí k jaké doméně se má připojit a podle zadaného klíče rozpozná kam má

posílat a ukládat naměřené hodnoty

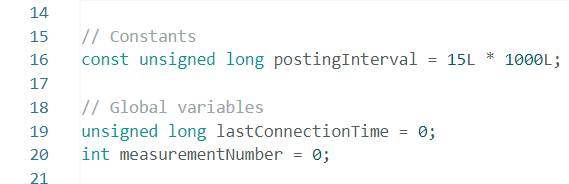
writeAPIKey je unikátní a neexistují že by byly dvě stejné

za “XXXXXXXXXXXXXXXX” vložím svůj unikátní klíč aby hodnoty se zapisovali do mých tabulek



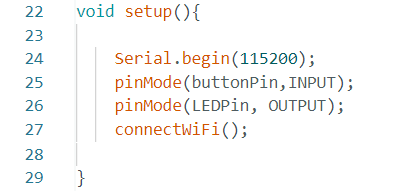
Příklad zde:

Dodatečné konstanty které kód využívá:



Zde vidíme při startu zařízení M5StickCPlus si určí tlačítka jako vstup a LED žárovku jako výstup

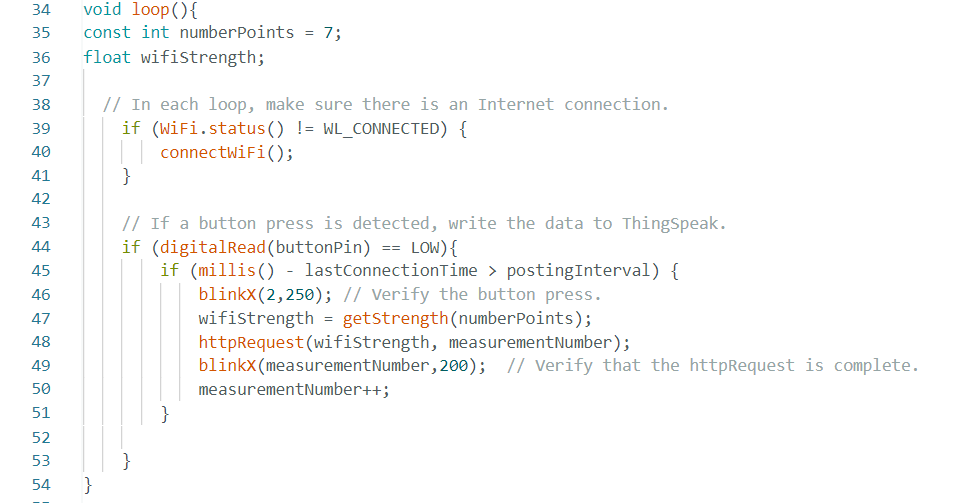
Dále se zařízení připojí k dané wifi



Začátek Loopu(pravidelné opakování)

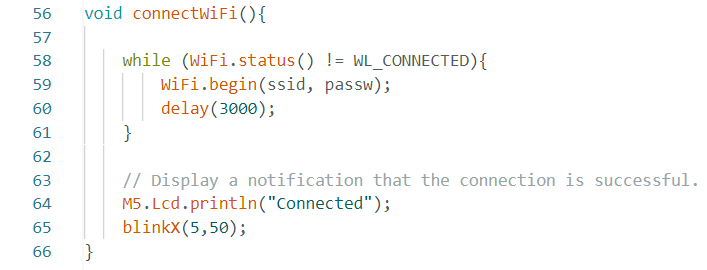
v každém loopu si ověříme zda jsme stále připojení k síti

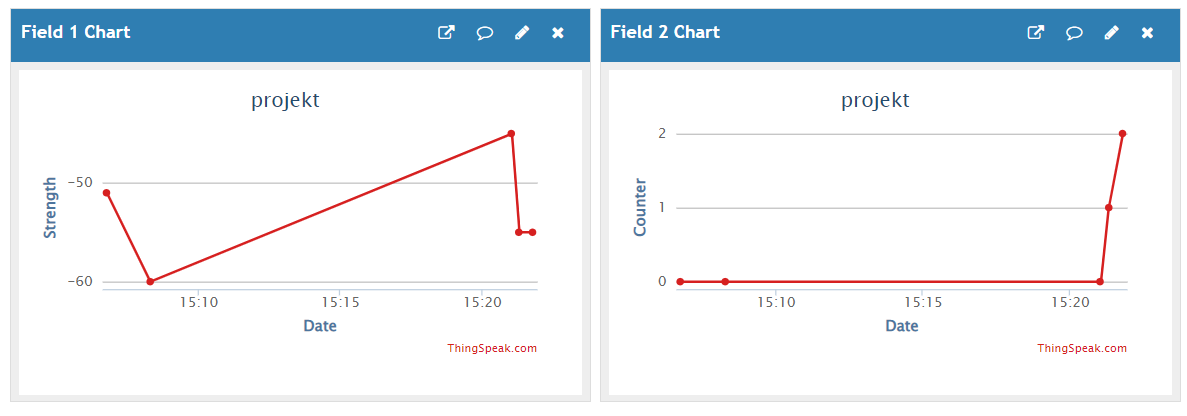
po kontrole když zmáčkneme tlačítko rozbliká se nám LED žárovka  
to nám řekne, že kód právě měří sílu wi-fi signálu daného zařízení



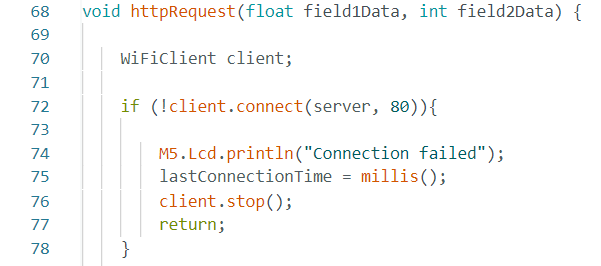
Po úspěšném měření se kód posune dál, pokud jsme stále připojení k síti,

tak na display zařízení se nám vypíše “připojeni”



Poté se zařízení pokusí spojit se stránkou ThingSpeak.com a požádá o přístup k datům našich grafů

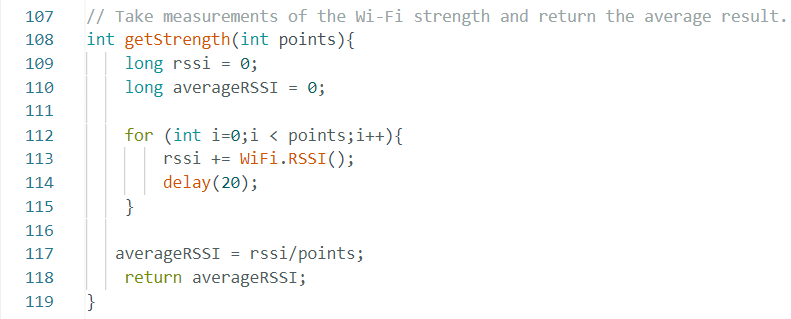
Pokud ke spojení nedojde kód vypíše na display zařízení chybu že “spojení selhalo”  
a zastaví se.



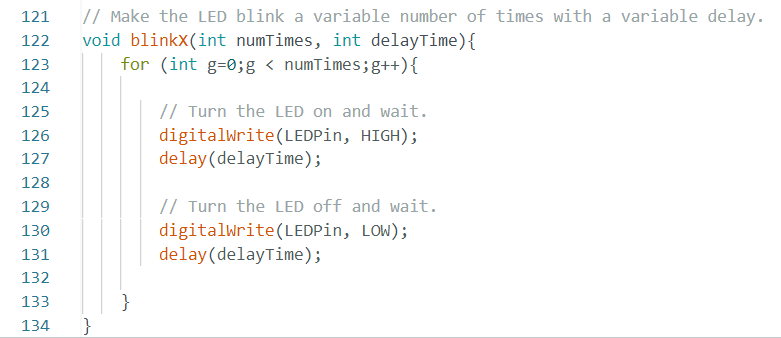
Jestliže spojení proběhne v pořádku tak do našich grafů se začnou zapisovat naše naměřené hodnoty díky našemu zapisnýmu klíči určíme do jakých grafů se mají hodnoty zapisovat.



měřené hodnoty se zprůměrují.



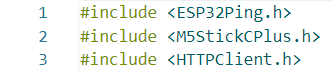
Pro naši kontrolu se LED žárovka zařízení rozbliká tím nám poví že zapsání proběhlo úspěšně



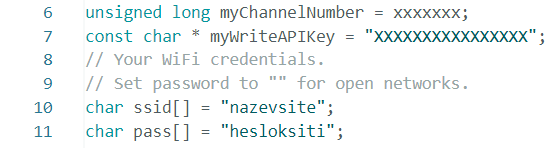
Tím se kód vrací na začátek loopu.

# Kód pro pingování zařízení

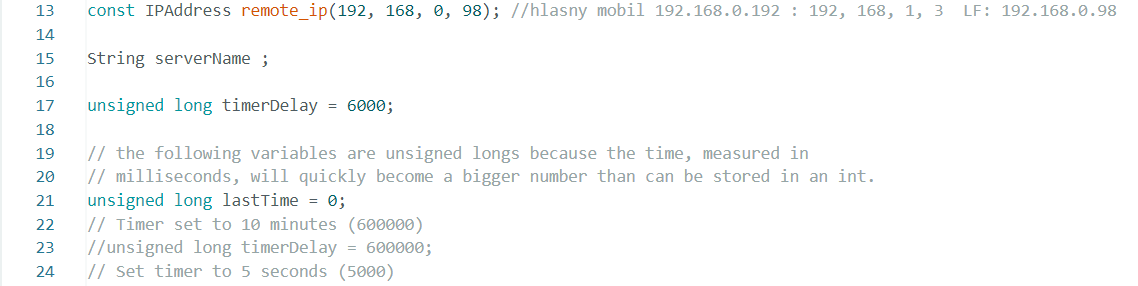
Potřebné knihovny:



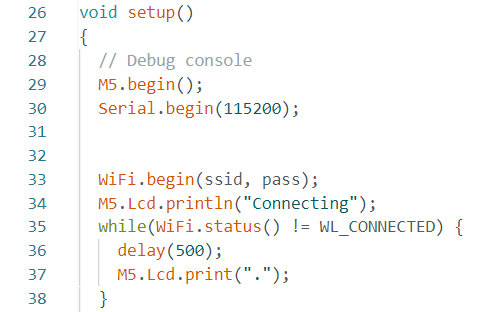
údaje k připojení wifi a připojení k ThingSpeak:



IP adresa kterou chceme pingovat a delay jak často ping chceme opakovat



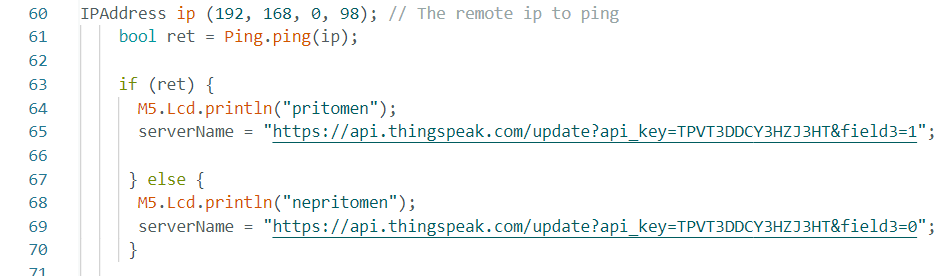
spuštění zařízení, připojení do sítě a následně na display vypíše, že se připojuje.

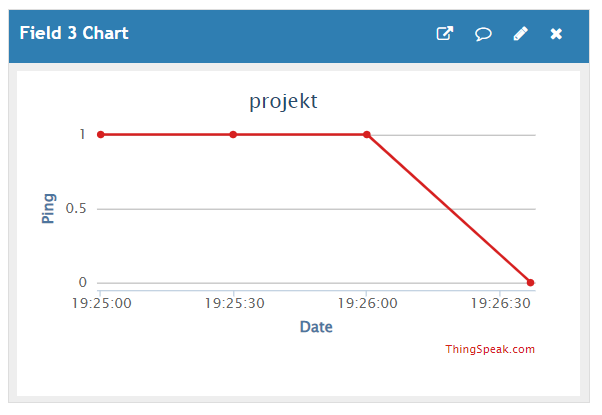


Dále v loopu se vypíšou informace kdy se bude loop opakovat

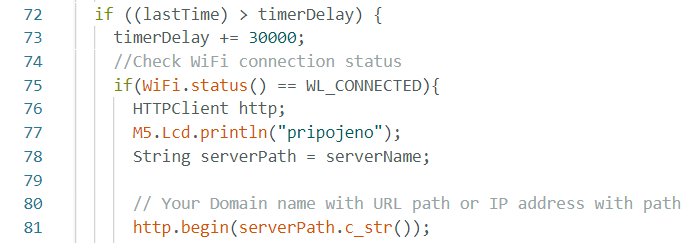


M5Stick pinguje ip adresu zařízení v tomto případě mobilní telefon.  
Na základě toho zda se na síti zařízení nachází vypisuje jestli je přítomno nebo nepřítomno.  
Tyto data pošle a zapíše do grafu na ThingSpeak



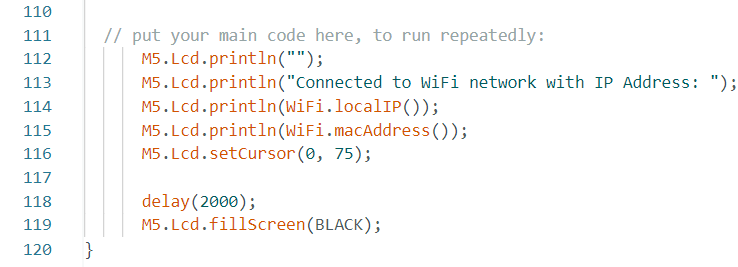


Každých 30 vteřin se ping z opakuje a zjistí zda pokaždé má připojení k Wi-fi.  
Zjistí jestli má spojení se serverem kam chceme data posílat





Poté opakovaně se na display bude vypisovat k jaké síti je připojeno s danou ip adresou a mac adresou potom se obrazovka resetuje.



Kód je nahráván do M5StickCPlus přes usb-c kabel

# Zdroje

Můj Kód + knihovny: <https://github.com/gabrielmaxh/Maturita>

Grafy ke kódu: <https://thingspeak.com/channels/1695691>

[www1] Wi-Fi  
[online]  
dostupné z : <https://cs.wikipedia.org/wiki/Wi-Fi> [cit. 14.3.2023]

[www2] Network address translation  
[online]  
dostupné z : <https://cs.wikipedia.org/wiki/Network_address_translation> [cit. 14.3.2023]

[www3] Dynamic Host Configuration Protocol  
[online]  
dostupné z : <https://en.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol> [cit. 14.3.2023]

[www4] Wireless Network Signal Strength with ESP32 with Arduino IDE

[online]  
dostupné z : <https://www.mathworks.com/help/thingspeak/measure-arduino-wifi-signal-strength-with-esp32.html> [cit. 14.3.2023]

[www5] ThingSpeak ReadDataFromChannel  
[online]  
dostupné z : <https://www.mathworks.com/help/thingspeak/read-data-from-channel.html> [cit. 14.3.2023]

[www6] ThingSpeak ListYourChannels  
[online]  
dostupné z : <https://www.mathworks.com/help/thingspeak/listyourchannels.html> [cit. 14.3.2023]

[www7] ThingSpeak Examples  
[online]  
dostupné z : <https://www.mathworks.com/help/thingspeak/examples.html> [cit. 14.3.2023]

[www8] ThingSpeak WriteSingleField GitHub  
[online]  
dostupné z : <https://github.com/mathworks/thingspeak-arduino/blob/master/examples/ESP32/WriteSingleField/WriteSingleField.ino> [cit. 14.3.2023]

[www9] ESP32 HTTP GET and HTTP POST with Arduino IDE (JSON, URL Encoded, Text)  
[online]  
dostupné z : <https://randomnerdtutorials.com/esp32-http-get-post-arduino/> [cit. 14.3.2023]

[www10] Pi-hole dokumentace  
[online]  
dostupné z : <https://pi-hole.net/> [cit. 14.3.2023]

[www11] Topologie sítí dokumentace  
[online]  
dostupné z : <https://cs.wikipedia.org/wiki/Topologie_s%C3%ADt%C3%AD> [cit. 14.3.2023]

[www12] Privátní sítě dokumentace  
[online]  
dostupné z : <https://cs.wikipedia.org/wiki/Priv%C3%A1tn%C3%AD_s%C3%AD%C5%A5> [cit. 14.3.2023]

[www13] M5Stack M5StIckC PLUS ESP32-PICO Mini kit  
[online]  
dostupné z : <https://rpishop.cz/m5stack/4175-m5stickc-plus-esp32-pico-mini-iot-vyvojovy-kit.html> [cit. 14.3.2023]

1. V případě nesouhlasu se zadáním maturitní práce se v den zadávání žák písemně obrátí na ředitele   
   VOŠ, SPŠ, SOŠS a CR s odůvodněním svého nesouhlasu. [↑](#footnote-ref-0)