



# Inteligentné relé s WiFi konektivitou do siete eduroam

Bakalárska práca

Martin Chlebovec

Vedúci práce: prof. Ing. Miloš Drutarovský, CSc.

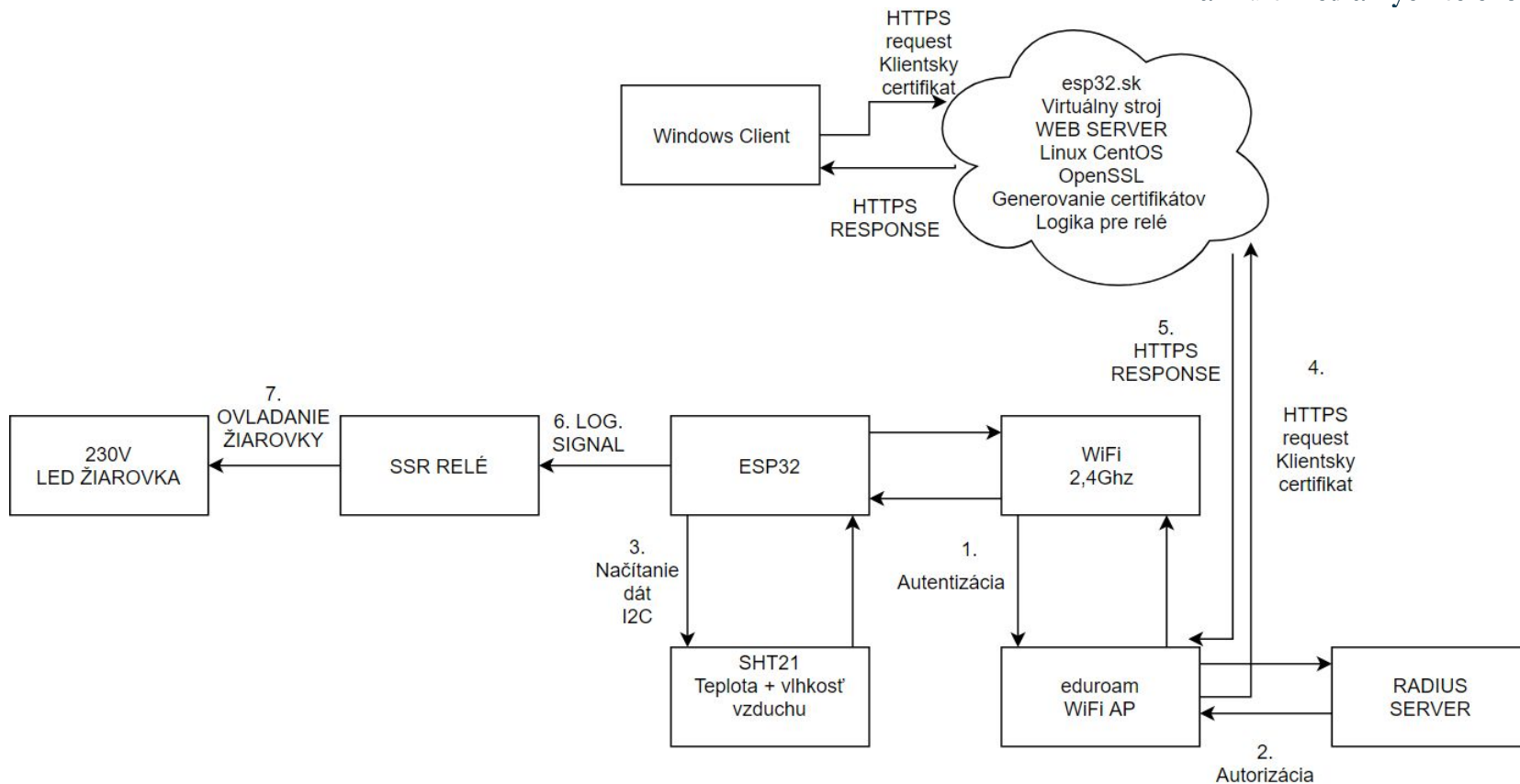
KEMT FEI TU v Košiciach

# Zadanie práce a riešená problematika

Na báze mikropočítača ESP32 navrhnete a otestujete inteligentné relé umožňujúce jeho bezpečné ovládanie prostredníctvom Internetu. Bezpečné ovládanie realizujete použitím vhodného šifrovania komunikácie s inteligentným relé. Programové riešenie vytvorte v jazyku C s využitím platformy Arduino. Navrhnete vhodné využitie architektúry klient-server a demonštrujete jej využitie na typických príkladoch (scenároch) použitia. V rámci experimentov otestujete aj funkčnosť navrhnutého riešenia v školskej sieti eduroam.

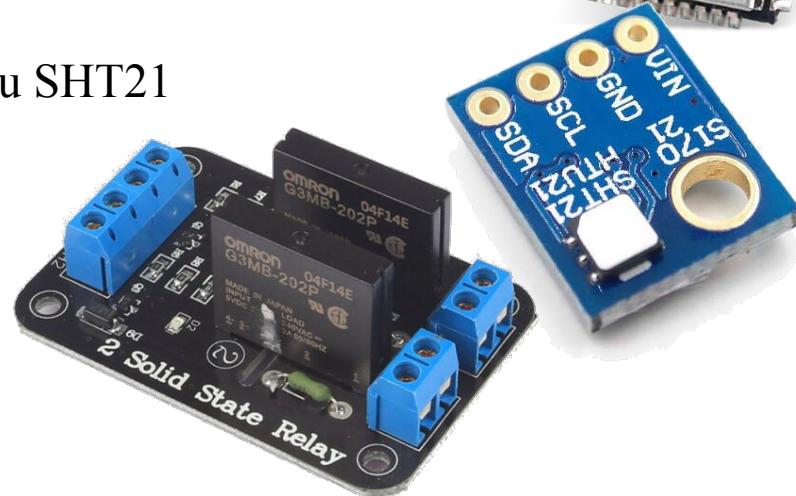
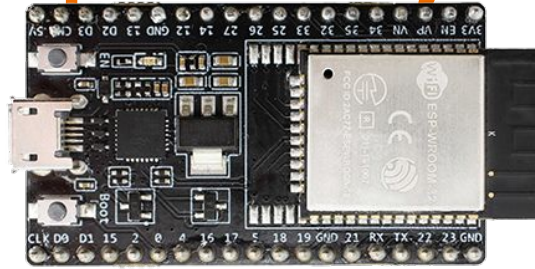
- Pripojenie vývojovej dosky s čipom ESP32 do siete eduroam
- Tvorba webového rozhrania pre Inteligentné relé s programovou logikou
- Návrh a realizácia zabezpečenej klient-server architektúry (HTTPS protokol)
- Generovanie certifikátov algoritmom RSA nástrojom OpenSSL
- Návrh minimálnej schémy s čipom ESP32-WROOM-32

# Blokový návrh riešenia



# Použité hardvérové prostriedky

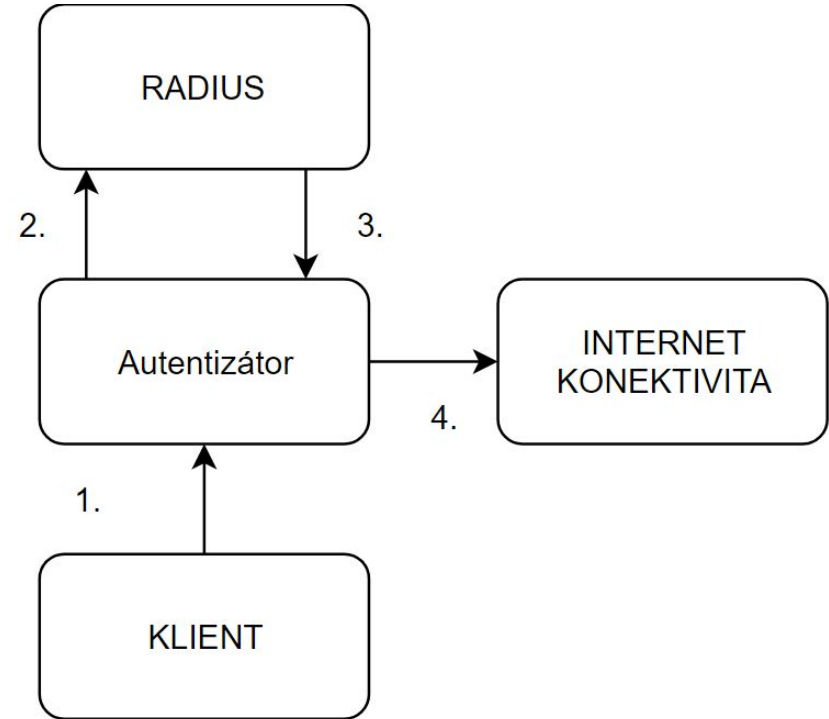
- ESP32 DevKitC V4
- ESP32-WROOM-32
- I2C modul teploty a vlhkosti vzduchu SHT21
- SSR relé modul OMRON G3MB



# Použité softvérové prostriedky

- Arduino IDE (vývojové prostredie pre jazyk Wiring)
- Arduino Core (sada knižníc čipu ESP32 pre jazyk Wiring)
- OpenSSL (kryptografický nástroj a knižnica (aj) pre generovanie certifikátov)
- Značkovací jazyk HTML, PHP, knižnica jQuery pre tvorbu webového rozhrania
- Práca s operačným systémom Windows, nastavenie certifikátov, PacketSender
- Práca s operačným systémom Linux Cent OS, použitie nástroja OpenSSL, nastavenie certifikátov, HTTPS spojenia
- Autodesk Eagle pre návrh elektrotechnickej schémy prototypu

- Navrhnutá pre akademickú a výskumnú sféru
- Prosebník (Supplikant) pre klienta
- Komunikacia cez Autentizátor (AP, switch)
- Overovací prvok - RADIUS server
- Autentizácia, autorizácia, tarifikácia
- Identita, realm používateľov
- Poskytovateľ identity, služby





# Webové rozhranie pre inteligentné relé


- Programová implementácia v jazyku PHP, funkciami jQuery knižnice
- Aktuálny záznam teploty a vlhkosti vzduchu, systémových dát
- Softvérové riadenie spínania relé
- Automatický režim - spínanie na základe cieľovej, nameranej teploty a hysterézy
- Manuálny režim - zapnutie/vypnutie relé na dobu neobmedzenú
- Vzdialený reštart ESP32
- Indikátor konektivity


# Webové rozhranie pre inteligentné relé


<https://esp32.sk> (Prístupné po preukázaní sa klientským certifikátom)

 Prehľad

 Ovládanie

 Reštart

 Zdrojový kód

 Wiring

**Uptime:**  
0 hodín  
2 minút  
44 sekúnd

 **Teplota**

30.42 °C

 **Vlhkosť**

49.79 %


## Výstupy

Výstup	Režim	Ref. teplota	Teplota	Hysteréza	Stav
 <b>Relé</b>	Manual	28 °C	30.42 °C	+ 0.5 °C	VYP

## Wifi

**SSID**

**Sila signálu**

 eduroam

-82 dBm

## Mikrokontrolér

Názov	Teplota	Hall	Konektivita
 <b>ESP32 Devkit</b>	49.44 °C	23 m <sup>3</sup> A <sup>-1</sup> s <sup>-1</sup>	



# Prenos dát na web server z ESP32

- Zapuzdrenie dát do dopytu na cieľový .php súbor GET metódou
- subor.php?teplota=20.35&vlhkost=60.89
- ? - znak pre požiadavku (request)
- & - oddeľovač viacerých hodnôt
- Spracovanie dát serverovým jazykom PHP
- Uloženie dát
- Vykonanie logiky inteligentného relé
- Vyčítanie stavu relé prostredníctvom prečítania .txt súboru
- Orezanie HTTP hlavičky
- Prečítanie a aplikovanie stavu premennou

```
HTTP/1.1 200 OK
Date: Sun, 16 Jun 2019 02:11
Server: Apache/2.2.15 (CentO
Strict-Transport-Security: m
X-Frame-Options: DENY
X-Content-Type-Options: nosn
Last-Modified: Wed, 12 Jun 2
ETag: "5f678-3-58b1eaf74ce93
Accept-Ranges: bytes
Content-Length: 3
Connection: close
Content-Type: text/plain; ch

[V][ssl_client.cpp:243] stop
VYP
```

# Generovanie certifikátov - OpenSSL

- Použitie kryptografického nástroja OpenSSL
- Využitie algoritmu RSA pri generovaní certifikátov
- Obojstranná autentizácia (server-klient, klient-server)

## Generované typy certifikátov

- Certifikát certifikačnej autority
- Certifikát webového servera
- Certifikát pre klienta

```
[E][ssl_client.cpp:32] handle_error(): X509 - Certificate verification failed, e.g. CRL, CA or  
[E][ssl_client.cpp:34] handle_error(): MbedTLS message code: -9984  
[E][WiFiClientSecure.cpp:108] connect(): lwip_connect_r: 11  
Nepodarilo sa odoslat data  
Pripajam sa na: esp32.sk  
[E][ssl_client.cpp:32] handle_error(): X509 - Certificate verification failed, e.g. CRL, CA or  
[E][ssl_client.cpp:34] handle_error(): MbedTLS message code: -9984  
[E][WiFiClientSecure.cpp:108] connect(): lwip_connect_r: 11  
Nepodarilo sa nacistat stav rele
```

## Problémy pri realizácii bakalárskej práce:

- Problémová implementácia certifikátov generovaných z 1024 bitového kľúča pre ESP32
- Riešenie v podobe generovania kľúčov s minimálnou dĺžkou 2048 bitov.

# Implementácia certifikátov

Windows:

- Inštalácia certifikátu certifikačnej autority do Dôveryhodných koreňových certifikačných autorít
- Inštalácia klientského certifikátu do Osobných certifikátov

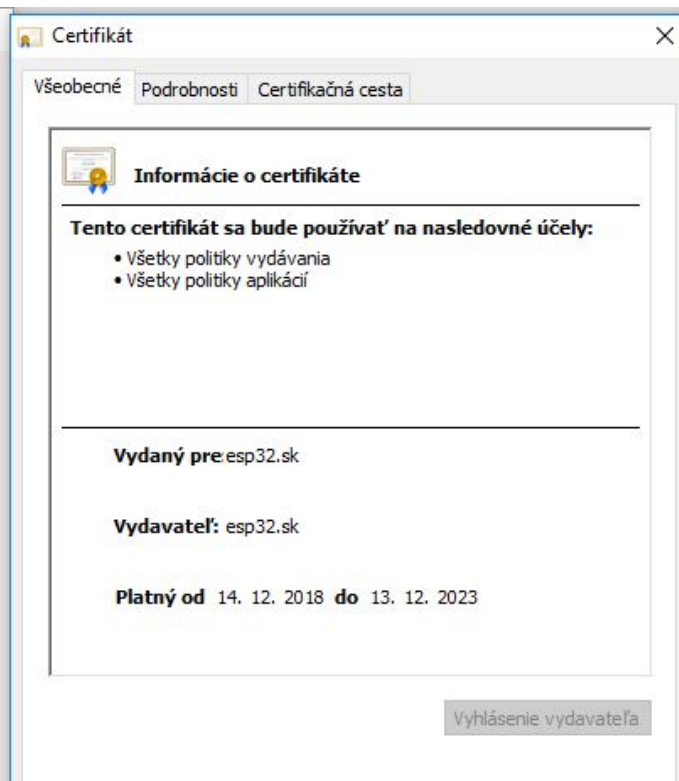
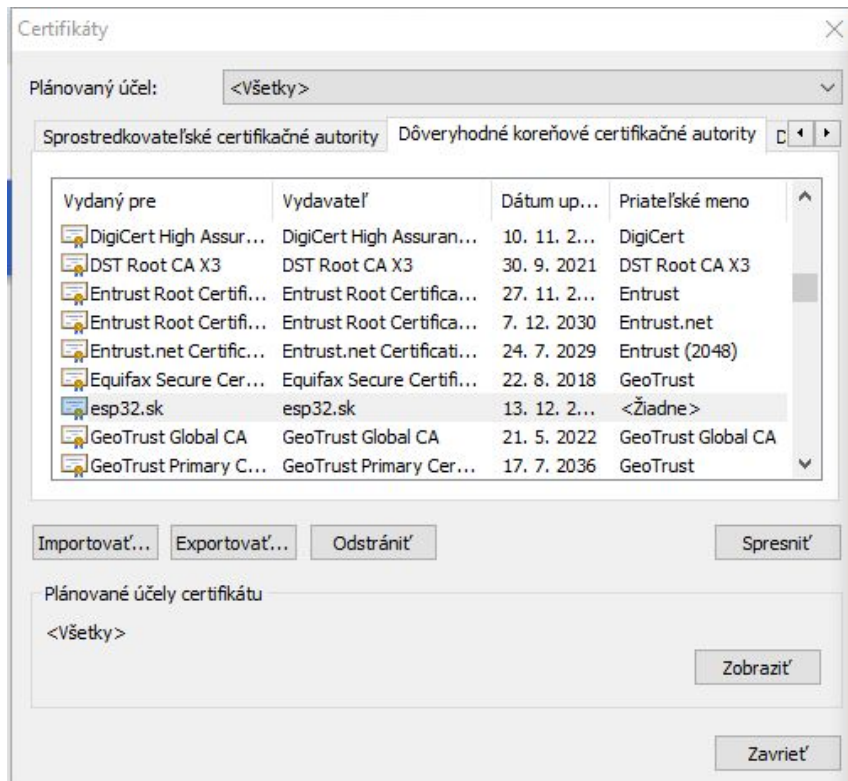
Linux Cent OS:

- Nastavenie umiestnení certifikátov do konfiguračného súboru balíka httpd

ESP32:

- Implementácia certifikátov do programu a použitie vo funkciách

# Implementácia certifikátov - prakticky



# Overenie klienta v praxi - ESP32

```
[V][ssl_client.cpp:53] start_ssl_client(): Free internal heap before TLS 284472
[V][ssl_client.cpp:55] start_ssl_client(): Starting socket
[V][ssl_client.cpp:88] start_ssl_client(): Seeding the random number generator
[V][ssl_client.cpp:97] start_ssl_client(): Setting up the SSL/TLS structure...
[V][ssl_client.cpp:110] start_ssl_client(): Loading CA cert
[V][ssl_client.cpp:158] start_ssl_client(): Loading CRT cert
[V][ssl_client.cpp:165] start_ssl_client(): Loading private key
[V][ssl_client.cpp:175] start_ssl_client(): Setting hostname for TLS session...
[V][ssl_client.cpp:190] start_ssl_client(): Performing the SSL/TLS handshake...
[D][ssl_client.cpp:203] start_ssl_client(): Protocol is TLSv1.2 Ciphersuite is TLS-ECDHE-RSA-WITH-AES-256-GCM-SHA384
[D][ssl_client.cpp:205] start_ssl_client(): Record expansion is 29
[V][ssl_client.cpp:211] start_ssl_client(): Verifying peer X.509 certificate...
[V][ssl_client.cpp:220] start_ssl_client(): Certificate verified.
[V][ssl_client.cpp:235] start_ssl_client(): Free internal heap after TLS 242528
[V][ssl_client.cpp:274] send_ssl_data(): Writing HTTP request...
[V][ssl_client.cpp:243] stop_ssl_socket(): Cleaning SSL connection.
```

# Minimálna schéma, prototyp s ESP32

- Návrh minimálnej schémy pre samostatný čip ESP32 s napájacou sústavou v prostredí Autodesk Eagle
- Implementácia (prepojenie) s komponentami inteligentného relé
- Zostavenie prototypu s dokumentáciou (Ing. Slovák PhD. z katedry KEMT)
- Využitie samostatného USB-UART FTDI prevodníka pre programovanie

Problémy pri ESP32 DevKitC V4:

- Po pripojení napájania čip nenabootuje z dôvodu rovnakých časových konštánt nábehu logických signálov pre BOOT a EN vývod (DOWNLOAD mód)

Riešenie:

- Zvýšenie časovej koštanty kondenzátora
- Vlastná sekvencia riadiacich signálov



# Prototyp s ESP32

**SVETELNÝ ZDROJ  
LED ŽIAROVKA**

**SHT21**

**PROTOTYP S ESP32-WROOM-32**

**SSR RELÉ  
OMRON  
G3MB-202**

**15/17**

- Overená funkčnosť pripojenia do siete eduroam
- Navrhnutý a odtestovaný systém inteligentného relé s webovým rozhraním spoločne
- Generované certifikáty s úspešnou implementáciou pre rôzne operačné systémy a mikrokontroler ESP32
- Navrhnutá schéma pre využitie samostatného čipu ESP32
- Overená funkčnosť minimálneho zapojenia s prototypom



# Ďakujem za pozornosť

Na záver sa chcem poďakovať vedúcemu práce prof. Ing. Milošovi Drutarovskému, CSc. za odborné rady a vedenie bakalárskej práce a taktiež odbornému asistentovi Ing. Slovákovi PhD. za konzultáciu v oblasti návrhu elektrotechnických schém a vyhotoveniu prototypu spoločne s projektovou dokumentáciou výrobu plošného spoja a osadením samostatného čipu ESP32.

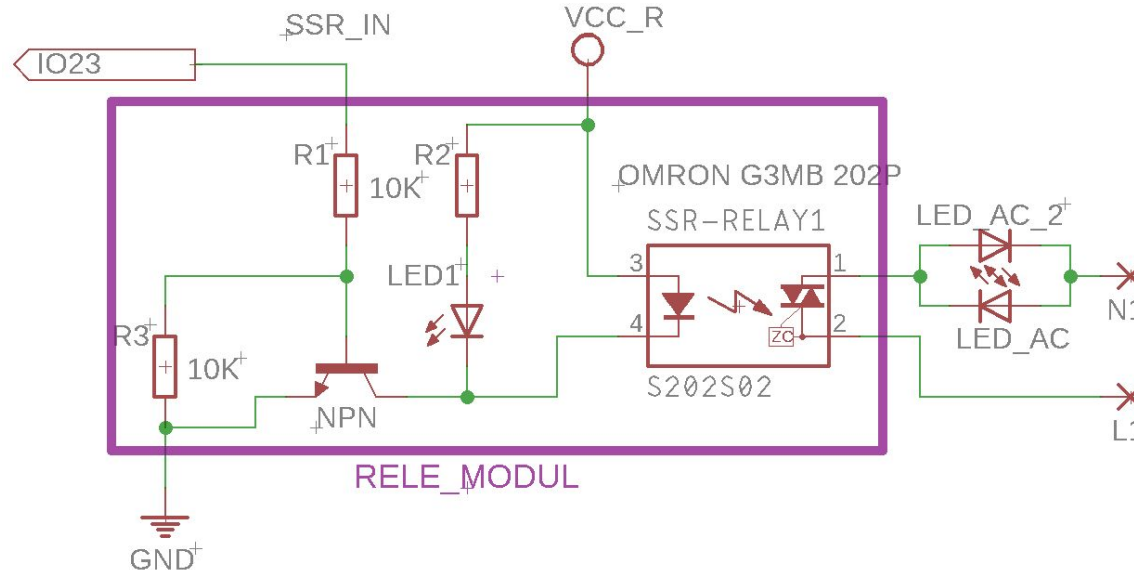
Martin Chlebovec

Inteligentné relé s WiFi konektivitou do siete eduroam

# Otázka 1

Prečo je v schéme BP (obr. 15) uvedený len jeden kanál s SSR relé?

- Minimálna schéma obsahuje minimálne zapojenie, ktoré sa v BP používa. Obsahuje teda zakreslený iba jeden ovládaný kanál pre SSR relé. Druhý kanál nemá žiadnu funkcionality, slúži iba ako možná náhrada v prípade poruchy.

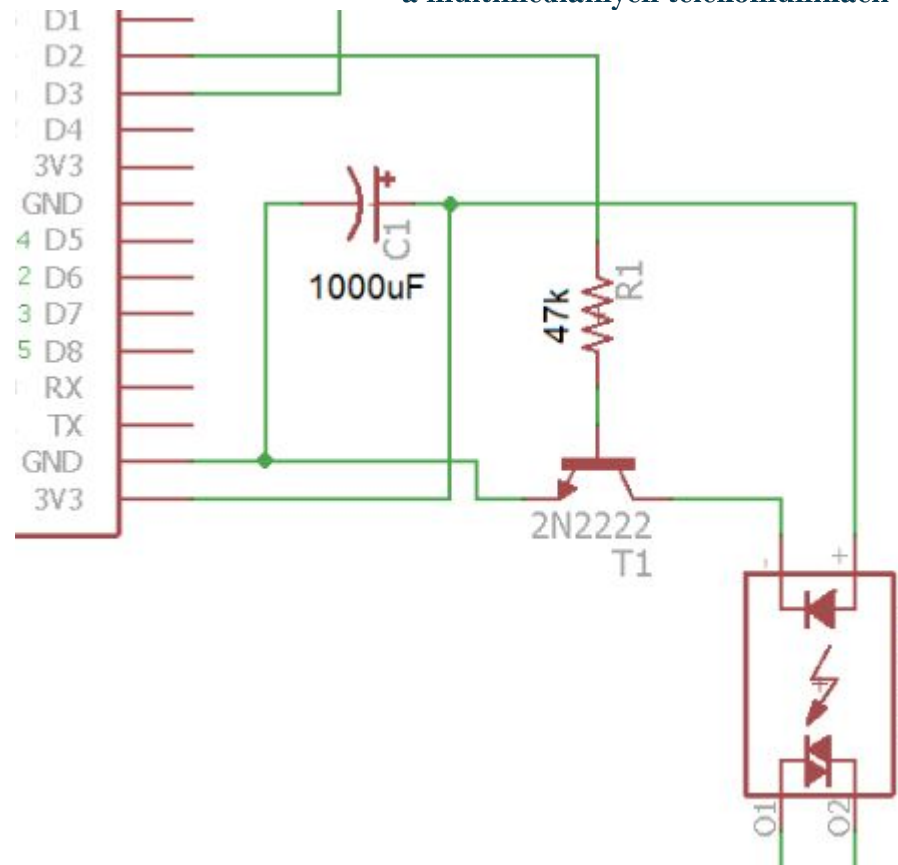
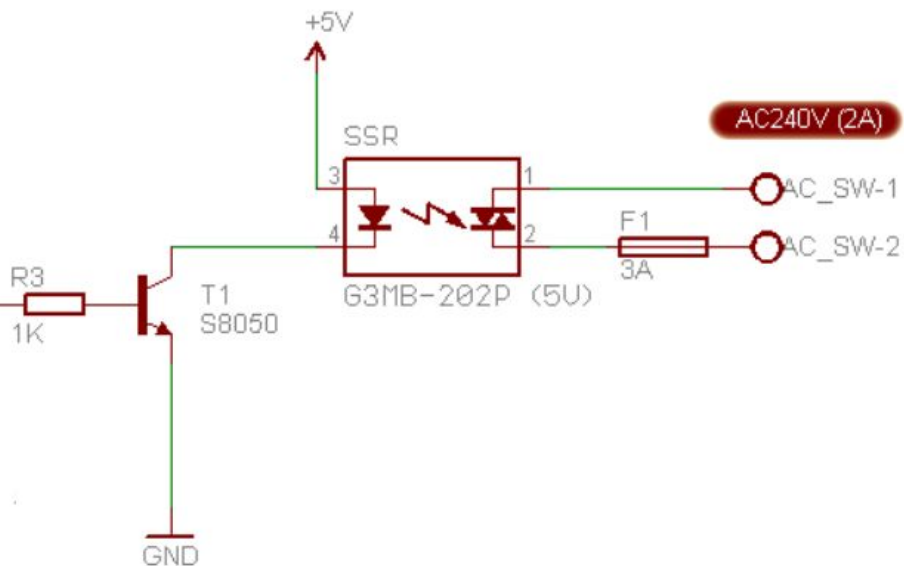


# Otázka 1

Nech bakalár pri obhajobe vysvetlí činnosť SSR relé (str. 29, 2. odstavce zhora) pri ovládaní riadiacim signálom, z modulu mikropočítača ESP32.

- „Obvod je zopnutý ak je na tranzistor privedený active-low signál, t.j. log 0, čo vo výsledku aktivuje triak privedením 5V na riadiacu elektródu prostredníctvom tranzistora. Pri log 1 je obvod rozopnutý v dôsledku nepriechodnosti triaku.“
- Nedostupná schéma
- Zakreslenie z dostupných schém

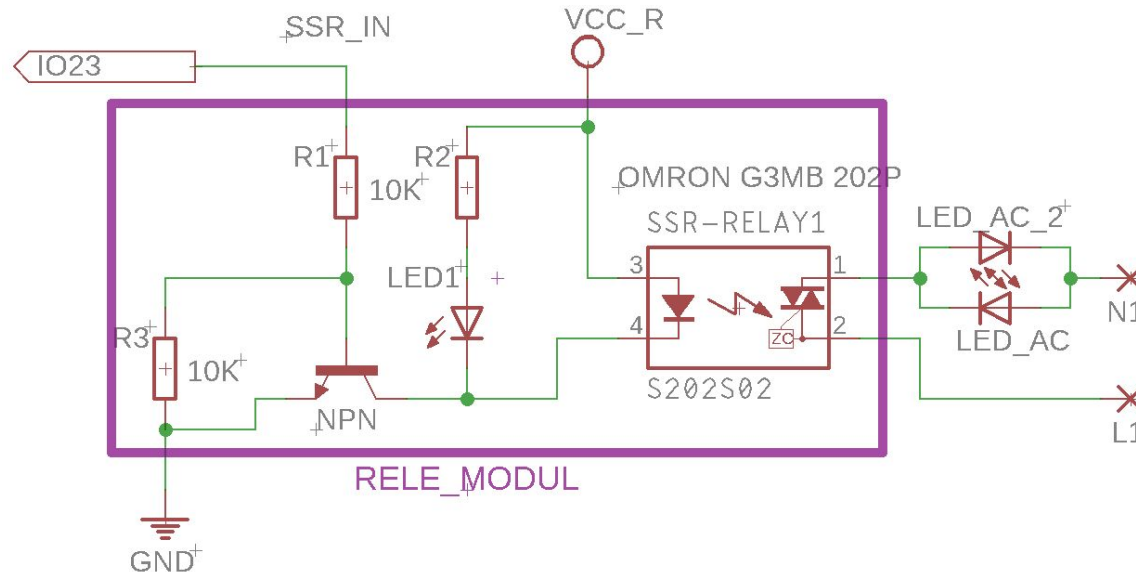
# Otázka 1



# Otázka 1

Aká je funkcia diód LED AC\_2 a LED\_AC?

- LED\_AC\_2 a LED\_AC reprezentujú zjednodušené zapojenie svetelného zdroja. Napájacie napätie je 230V striedavých. Pripojenie polarity na vývody svetelného zdroja je votelitľné.



# Otázka 2

Bolo v prípade offline verzie ovládania inteligentného relé, t.j. prostredníctvom programu Packet Sender, úspešne otestované uvádzané obojsmerné (vzájomné, str. 24) overenie certifikátov?

- V prípade offline verzie ovládania inteligentného relé nebolo otestované overenie certifikátmi.
- Program som využíval iba na posielanie jednoduchých UDP správ s informáciou pre ovládanie výstupu ESP32
- Knižnica AsyncUDP pre ESP32 v súčasnosti nedisponuje funkciami pre implementáciu takéhoto riešenia

# Otázka 3

Ošetril bakalár a ak áno ako, prípady keď dôjde k nečakanému výpadku napájania inteligentného senzorového uzla na báze ESP32, alebo keď nedôjde ku nadviazaniu spojenia ako na strane senzorového uzla, tak aj na strane klienta?

- V prípade vypadnutia spojenia s prístupovým bodom sa inkrementuje programová premenná, pri určitej hodnote sa vykoná softvérový reštart ESP32 a výstup sa vypne až do nadviazania spojenia, kedy sa opäť synchronizuje s webom a je umožnená jeho zmena
- V prípade nedostupnosti webového rozhrania s dostupnosťou konektivity do WiFi nie je táto situácia programovo ošetrovaná. Relé zotrváva v poslednom známom stave.
- V prípade výpadku napájania pre ESP32 sa výstup rozopne a spotrebič je vypnutý.

# Otázka 3

Vysvetliť riadky: 156.`while (client.connected()){..}` a 166.`String`  
`premenna=client.readStringUntil('\n');` Keď cyklus skončí na „client not connected“, čo  
sa bude diať v nasledujúcom riadku `client.readString`?



```
if (client.connect(host, 443)) {
    Serial.println("Pripojenie pre stav rele uspesne");
    String url = "/values/stav.txt";
    client.print(String("GET ") + url + " HTTP/1.1\r\n")
    while (client.connected()) {
        String hlavicka = client.readStringUntil('\n');
        Serial.println(hlavicka);
        if (hlavicka == "\r") {
            break;
        }
    }
    String premenna = client.readStringUntil('\n');
    Serial.println(premenna);
    if (premenna == "ZAP") { //rozhodovanie medzi ZAP
        Serial.println("ZAPINAM RELE");
        digitalWrite(rele, LOW);
        // digitalWrite(LED_BUILTIN, HIGH);
    } else if (premenna == "VYP") {
        Serial.println("VYPINAM RELE");
        digitalWrite(rele, HIGH);
        // digitalWrite(LED_BUILTIN, LOW);
    }
} else {
    Serial.println("Nepodarilo sa nacistat stav rele");
}

client.stop(); //ukonc spojenia
```

```
HTTP/1.1 200 OK
Date: Sun, 16 Jun 2019 12:04:14 GMT
Server: Apache/2.2.15 (CentOS)
Strict-Transport-Security: max-age=63072000; includeSubDomains
X-Frame-Options: DENY
X-Content-Type-Options: nosniff
Last-Modified: Wed, 12 Jun 2019 11:26:32 GMT
ETag: "5f678-3-58b1eaf74ce93"
Accept-Ranges: bytes
Content-Length: 3
Connection: close
Content-Type: text/plain; charset=UTF-8

[V][ssl_client.cpp:243] stop_ssl_socket(): Clean
VYP
VYPINAM RELE
```

# Otázka 4

Nech bakalár vysvetlí dôvod prečo sa po každej požiadavke pripája a odpája od servera?

- Problémy knižnice WifiClientSecure
- Chybovosť, nesprávne návratové kódy pri viacnásobných spojeniach pri jednom pripojení. Z tohto dôvodu je z hľadiska spoľahlivosti lepšie použiť nové pripojenie pri každej požiadavke.