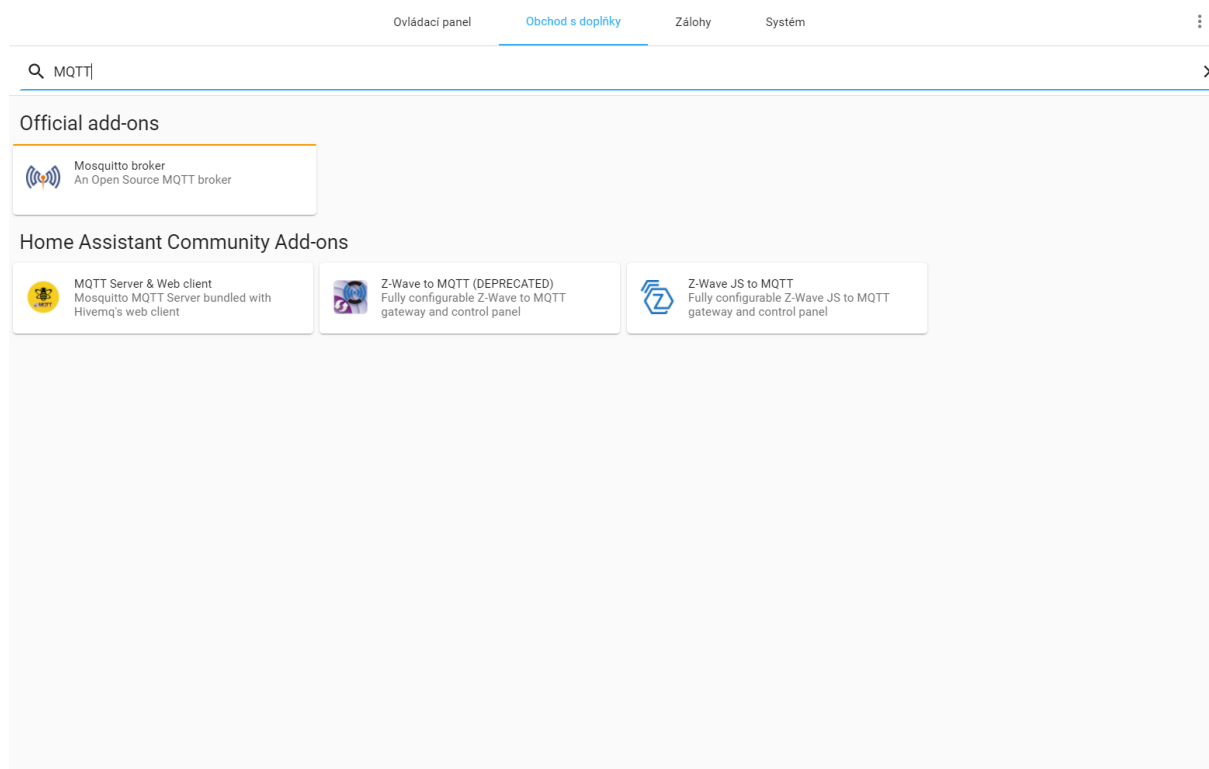


# Příloha č.2 – Uživatelská Dokumentace

Tato příloha se dělí na čtyři hlavní části, první je část zaměřená na dokumentaci Chytré hlavice, druhá se zaměřuje na čidlo spotřeby, třetí je zaměřená na detektor zatopení sklepa a poslední čtvrtá část popisuje instalaci služby pro vzdálené ovládání. Jednotlivé části prvních třech kapitol jsou dále rozdělené do dalších podčástí, se zaměřením na dokumentaci tvorby, úpravu softwaru pro vlastní užití, a nakonec integrace do Home Assistantu.

## Nutné prerekvizity

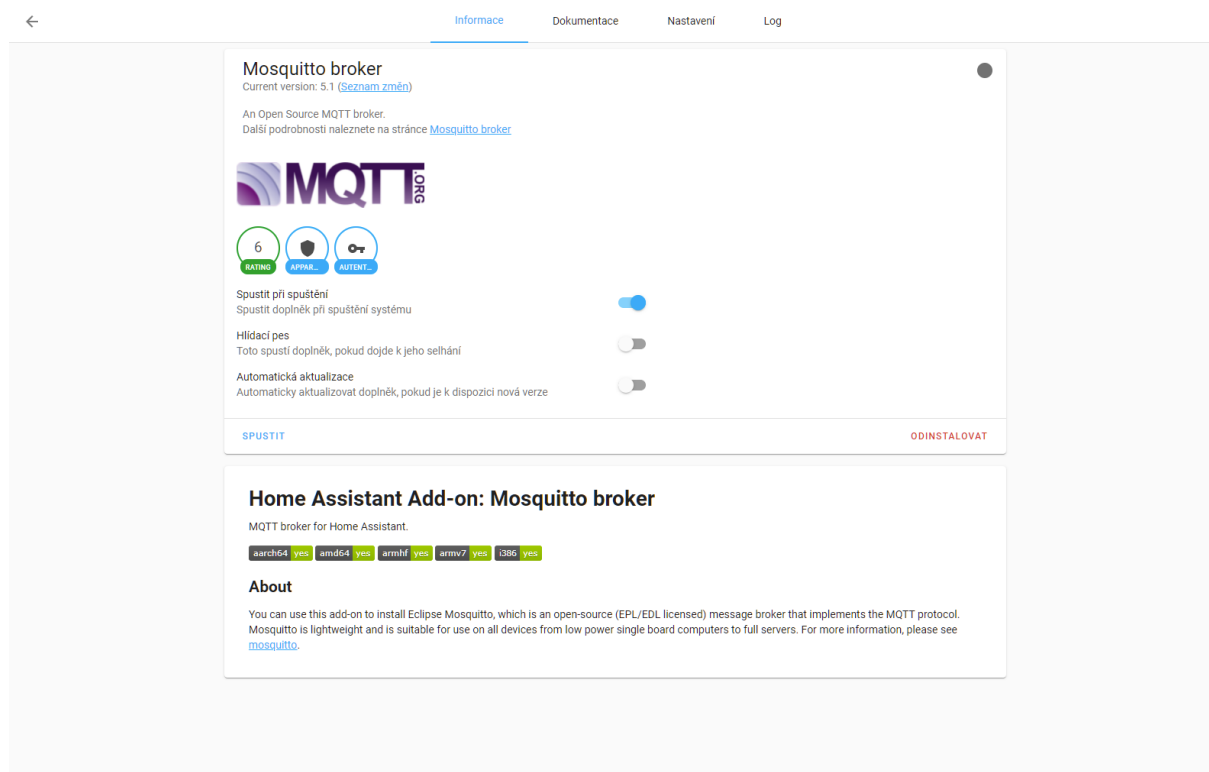
Všechna chytrá zařízení využívají pro komunikaci protokol MQTT. Pro správné fungování je tedy zapotřebí mít v Home Assistantu nainstalovaný MQTT Broker, který lze doinstalovat přímo v záložce doplňky v nastavení Home Assistantu = Supervisor -> obchod s doplňky. Jako doplněk k instalaci vybereme „mosquitto broker“.



**Obrázek 1 - Obchod pro přidání doplňků**

Po instalaci MQTT Brokeru je třeba vytvořit uživatele pro budoucí chytrá zařízení. To lze provést v nastavení Home Assistantu = Nastavení -> Uživatelé -> Přidat uživatele. Jméno a heslo nového uživatele si zaznamenáme pro užití v budoucích krocích.

Nyní se vrátíme do nastavení supervisor a otevřeme nastavení MQTT Brokeru, kde musíme samotný mosquitto broker zapnout. Důležité je také mít zapnuté automatické spouštění.



**Obrázek 2 - Konfigurace MQTT Brokeru**

Po spuštění je MQTT Broker připraven pro užití ve vnitřní domácí síti. Jedná se o základní konfiguraci, kterou lze dále rozšířit o zabezpečení pomocí certifikátu, nebo přísnější pravidla pro přístup.

## Chytrá hlavice

V této části čtenář získá návod na sestrojení hlavice, úpravu kódu a také jak následně hlavici integrovat do Home Assistantu. **Chytrou hlavici lze v současném stavu sestroit pouze s užitím 3D tiskárny, vzhledem k nutnosti vytisknout některé mechanické součástky.**

### Co je potřeba

- Notebook/počítač s nainstalovaným Arduino IDE
- Přístup k 3D tiskárně
- Páječka
- Nerezový šroub M5x20 s maticí bez brzdy
- ESP8266-Pro, nebo modifikované ESP 8266-01 s vyvedeným pinem pro deepsleep a ADC, nebo jiná alternativa desky ESP s potřebnými piny.

- Univerzální plošný spoj
- Rezistory
- DS18B20 teplotní čidlo
- L9110S H-můstek
- DC motor 6-12V 8-16RPM s převodovkou a dlouhou hřídelí (180 mA 12VDC)
- Pouzdro pro dvě AA baterie
- Tlačítko
- Kvalitní vteřinové lepidlo

### Část první – sestrojení (USB napájená hlavice)

Nejprve je nutné na 3D tiskárně vytisknout mechanické součástky. Jedná se o součástky v příloze „Příloha č3 – Zdrojová data 3D modelů (CAD)“ této práce ve složce hlavice. Pro vytištění se hodí materiál PET-G pro svou dobrou cenu a dobrou mechanickou i tepelnou odolnost. Naopak často využívaný materiál PLA se pro tuto konkrétní aplikaci nehodí z důvodu nižší tepelné odolnosti a po čase ztráty pevnosti ve stlačovaných místech. Po vytištění máme k dispozici 4 hlavní mechanické součástky. Na obrázku lze vidět součástky spolu s popisem.

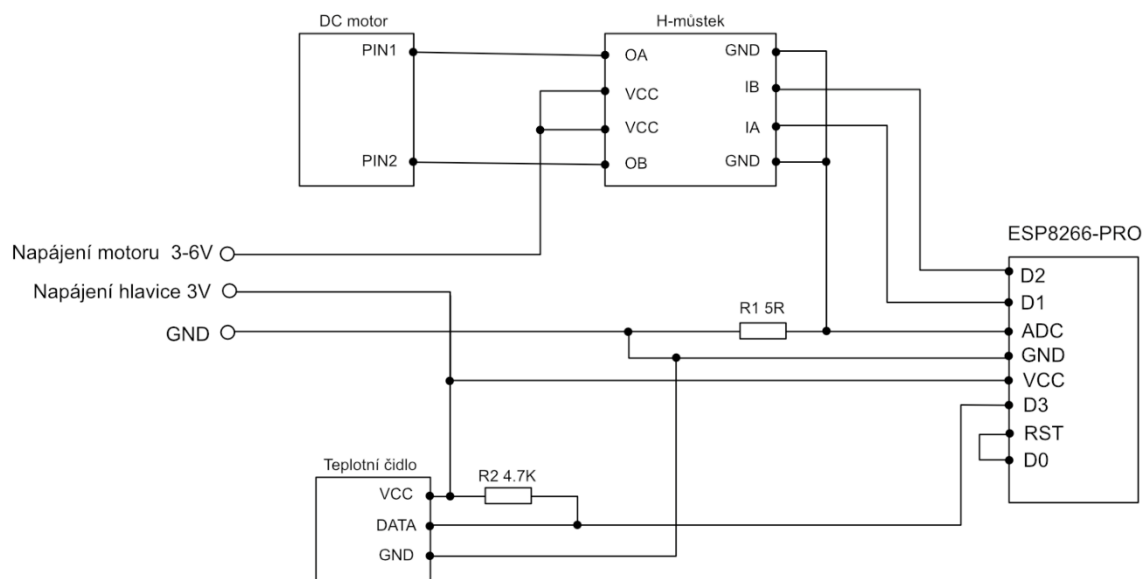


Obrázek 3 – Mechanické součástky s popisem

Hlavice je navržena pro kombinaci s nerezovým šroubem M5 s maticí bez brzdy. Matice s brzdou by zde byla kontraproduktivní, zvyšovala by pouze tření. Hlavu šroubu vlepíme do připraveného otvoru v součástce č.3.

1. Nerezovou matici M5 vložíme do připraveného otvoru v díle č.1. Matice by zde měla zapadnout a sedět bez užití lepidla, ale pro delší výdrž je lepší matici zabezpečit několika kapkami lepidla. Je zde zapotřebí dávat pozor, aby se lepidlo v žádném případě nedostalo do závitu.
2. Nyní přichází na řadu krácení šroubu M5. Součástku č.1 přišroubujeme na ventil topení. Následně šroubujeme součástku č.3 s již přilepeným šroubem až do úplného uzavření ventilu topení. Poté si poznačíme vzdálenost, o kterou byl šroub posunut a následně šroub vyšroubujeme a zkrátíme na potřebnou délku.
3. Nyní je třeba upravit osu motoru. Zde je třeba osu zkrátit na cca 1 cm a vyříznout hranu, aby osa zapadla do otvoru v součástce č.4. Po nasazení zakápneme lepidlem.
4. Nyní nejprve přišroubujeme motor na součástku č.2, poté otočíme součástku č.1 a slepíme ji se součástkou č.2. Zde je ovšem zapotřebí vzdálenost vyzkoušet z důvodu různých odchylek při 3D tisku, aby se ozubená kola (součástky 3 a 4) dotýkala a vytvářela převod. Alternativou k lepení může být sešroubování, které provedeme tak, že páječkou vytvoříme skrze dva díly díрку, do které poté zašroubujeme o něco větší vrut, než je samotná dířka vytvořená páječkou.
5. V případě vypadávání matice M5 je možné vytisknout podložku kterou lze vložit mezi hlavici a ventil. (V příloze součástka č.0)

Po sestavení mechanické části je zapotřebí sestavit i elektroniku, k tomu slouží přiložené schéma. Je zde řešeno připojení motoru k ESP pomocí H-můstku, zapojení teplotního čidla ale také detekce odběru motoru. Celý obvod lze poté poskládat na univerzální plošný spoj, nebo využít přiložené schéma pro výrobu plošného spoje, které lze nalézt v příloze „Příloha č4 – Schémata elektroniky“



**Obrázek 4 - Schéma zapojení součástek hlavice**

### Část druhá – úprava kódu

Po sestavení je potřeba nahrát do ESP software. Kompletní software je přílohou této práce, ale je zapotřebí udělat změny pro funkčnost v konkrétní domácnosti. Zde je zapotřebí definovat SSID domácí wifi sítě a heslo, napsat IP adresu Home Assistantu a vypsát jméno a heslo pro uživatele MQTT brokeru. Další nutností je definování jména hlavice a cest pro ukládání hodnot pro Home Assistant, cesty pro nastavování teplot a také cestu pro změnu intervalu obnovení. Veškerá konfigurace je k nalezení i s popiskem v úvodní části kódu.

### Část třetí – Integrace do Home Assistantu

Pro integraci chytré hlavice do Home Assistantu použijeme konfigurační soubor „*configuration.yaml*“ kde vložíme novou konfiguraci MQTT, konkrétně *mqtt climate*.

```
climate:
  - platform: mqtt
    name: "Kancelar"
    send_if_off: true
    retain: true
```

```
availability_topic: "home/topeni/kancelar/stav"
min_temp: 8
max_temp: 27
current_temperature_topic: "home/topeni/kancelar/teplota"
temperature_command_topic: "home/topeni/kancelar/nastav"
temperature_state_topic: "home/topeni/kancelar/aktualne"
action_topic: "home/topeni/kancelar/rezim"
modes:
  - "off"
  - "heat"
```

Zde jsou vidět konfigurační proměnné entity, jako je například jméno, rozsah možných teplot pro nastavení a cesty pro komunikaci. Názvy jsou totožné s cestami v programu v ESP, tedy i jejich popis je totožný a lze nahlédnout do kódu.

Pro využití funkce prodloužení intervalu obnovy, lze využít automatizaci v Home Assistantu, kdy pro změnu intervalu obnovy u hlavičky je zapotřebí odeslat MQTT zprávu do tématu definovaném v kódu jako „refresh\_topic“. Zde je možnost 3, 5, 10, 15, nebo 30 minut.

## Měřič spotřeby

V této části čtenář získá návod na sestavení jednofázového měřiče spotřeby s rozsahem až do 100A, úpravu kódu a také jak následně měřič integrovat do Home Assistantu.

**VAROVÁNÍ** – Při instalaci přijde uživatel do kontaktu s vodiči střídavého napětí 230V, kdy hrozí reálné riziko smrti při neodborné manipulaci. Instalace by měla probíhat vždy odborným pracovníkem, nebo pod odborným dohledem!

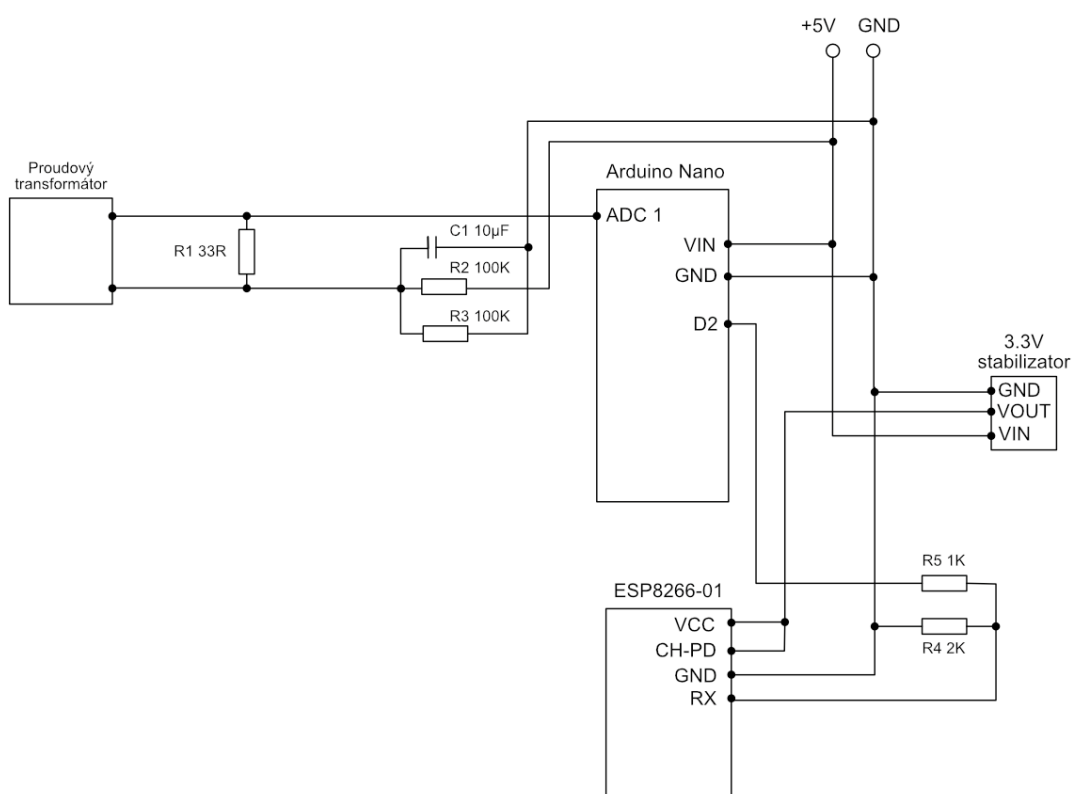
### Co je potřeba

- Notebook/počítač s nainstalovaným Arduino IDE
- Páječka
- Arduino nano
- ESP8266-01
- Proudový transformátor SCT-013-000
- Stabilizátor napětí AMS1117 3.3V
- Univerzální plošný spoj

- Odpor
- Kondenzátor 10 $\mu$ F

### Část první – sestrojení

Pro sestavení celého zařízení je potřeba znát samotné zapojení, k tomu slouží schéma níže. Je zde řešeno připojení proudového transformátoru a úprava jeho výstupu, zapojení Arduina, napájení modulu ESP, a nakonec jeho propojení s Arduinem. Pro napájení ESP je třeba zajistit stabilizaci z 5V na 3.3V a zároveň snížit komunikační napětí mezi Arduinem a ESP na logickou úroveň 3.3V pomocí děliče za užití odporů.



**Obrázek 5 - Schéma zapojení součástek pro měřič spotřeby**

Celý obvod lze poté poskládat na univerzální plošný spoj, nebo využít přiložené schéma pro výrobu plošného spoje, které lze nalézt v příloze „Příloha č4 – Schémata elektroniky“

Až po kompletním sestavení je možné proudový transformátor připojit na vodič. Je nutné proudový transformátor připojit pouze na jeden vodič, transformátor se nepřipojuje na svazek vodičů, nebo kabel.

### **Část druhá – úprava kódu**

Po sestavení je potřeba nahrát do Arduina a ESP software. Kompletní software je přílohou této práce, ale je zapotřebí udělat změny pro funkčnost v konkrétní domácnosti. V Arduinu je třeba upravit proměnnou „CURRENT\_CAL1“ v závislosti na naměřené hodnotě oproti reálnému výkonu spotřebiče. Druhou úpravou je nutnost upravit software pro ESP. Zde je zapotřebí definovat SSID domácí wifi sítě a heslo, napsat IP adresu Home Assistantu a vypsat jméno a heslo pro uživatele MQTT brokeru. Další nutností je definování jména čidla a cest pro ukládání hodnot pro Home Assistant. Veškerá konfigurace je k nalezení i s popiskem v úvodní části kódu.

### **Část třetí – Integrace do Home Assistantu**

Pro integraci do Home Assistantu použijeme konfigurační soubor „*configuration.yaml*“ kde vložíme novou konfiguraci MQTT, konkrétně mqtt senzoru.

```
- platform: mqtt
  name: "tepelne_cerpadlo_wh"
  unit_of_measurement: "wh"
  state_topic: "home/energy/cerpadlo/Wh"
```

a dále:

```
- platform: mqtt
  name: "tepelne_cerpadlo_A"
  unit_of_measurement: "A"
  state_topic: "home/energy/cerpadlo/current"
```

Kde „name“ je jméno entity v Home Assistantu, „unit of measurement“ je jednotka a „state topic“ je již zmíněná cesta, která je nakonfigurovaná v ESP.

Po úspěšné integraci senzorů je možné vytvořit integraci Utility meter. To je integrace přímo určená pro měření spotřeby, kdy je možné nakonfigurovat cyklus obnovy, tarify a další. Pro příklad si můžeme uvést konfiguraci měřící Wh s resetem hodnot každý měsíc.

mesicni\_spotreba\_cerpadlo\_wh:

```
source: sensor.tepelne_cerpadlo_wh
```

cycle: monthly

Po úspěšné konfiguraci je možné v rozhraní lovelace využívat entity a přidat si například grafy nebo jiné vizualizační doplňky Home Assistantu.

## **Detektor zatopení sklepa**

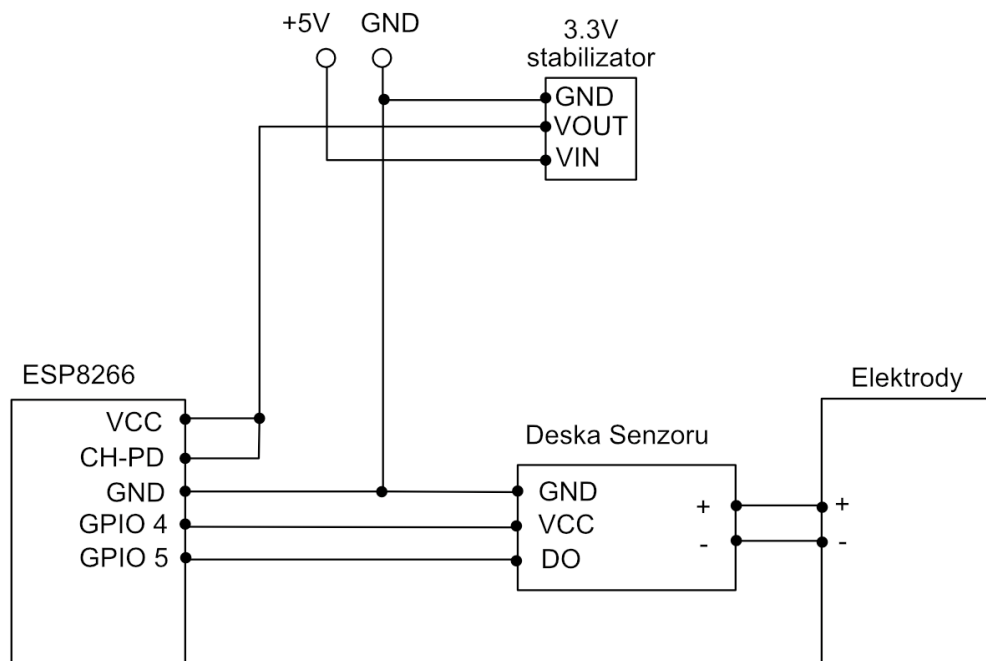
V této části čtenář získá návod na sestavení jednoduchého čidla detekující zatopení sklepa, úpravu kódu a také jak následně čidlo integrovat do Home Assistantu.

### **Co je potřeba**

- Notebook/počítač s nainstalovaným Arduino IDE
- Páječka
- ESP8266
- Čidlo vlhkosti
- Stabilizátor napětí AMS1117 3.3V

### **Část první – sestavení**

Pro sestavení celého zařízení je potřeba pouze několik součástek. Jedná se o nejjednodušší zařízení z těchto třech. Celé zařízení je sestavené podle schématu níže popsáném. Hlavní části jsou samotné elektrody pro měření, které jsou v místě detekce, druhou částí je zesilovač signálu, který je napojený na ESP8266, které je napájené stabilizátorem na 3.3V stabilizující z 5V ze zdroje.



**Obrázek 6 - Schéma zapojení součástek pro detektor zatopení**

### Část druhá – úprava kódu

Po sestavení je potřeba nahrát do ESP software. Kompletní software je přílohou této práce, ale je zapotřebí udělat změny pro funkčnost v konkrétní domácnosti. Je třeba definovat SSID domácí wifi sítě a heslo, napsat IP adresu Home Assistantu a vypsát jméno a heslo pro uživatele MQTT brokeru. Další nutností je definování jména čidla a cesty pro ukládání hodnoty pro Home Assistant. Veškerá konfigurace je k nalezení i s popisem v úvodní části kódu.

### Část třetí – Integrace do Home Assistantu

Pro integraci do Home Assistantu použijeme konfigurační soubor „*configuration.yaml*“ kde vložíme novou konfiguraci MQTT, konkrétně mqtt senzoru.

```
- platform: mqtt
  name: "sklep_zatopeni"
  state_topic: "home/zatopeni/sklep/stav "
```

Kde „name“ je jméno entity v Home Assistantu a state\_topic“ je již zmíněná cesta, která je nakonfigurovaná v ESP. Po úspěšné konfiguraci je možné v rozhraní lovelace využívat entitu s názvem *sklep\_zatopeni* a vizualizovat stav.

## Instalace vzdáleného ovládání na lokální instanci HA

Služba vzdáleného ovládání musí běžet ve stejné lokální síti jako instance Home Assistantu. V ideálním případě tato služba běží přímo v systému Home Assistant, avšak lze mít službu spuštěnou na jiném zařízení a pouze směřovat na IP lokální instalace Home Assistantu.

Pro vlastní běh v systému Home Assistant využívá tento program doplněk s názvem AppDaemon 4. Jedná se modul, který lze nainstalovat z obchodu s doplňky přímo ve správě Home Assistantu. Tento doplněk umožňuje vytvářet pokročilé programy, spouštět programy jako služby, nebo vytvářet kompletně nové uživatelské rozhraní. Pro účel instalace služby vzdáleného ovládání využijeme funkci automatického spouštění python programů.

Prvním krokem je tedy samotná instalace doplňku AppDaemon 4 z obchodu s doplňky v Home Assistantu. Po instalaci je zapotřebí, pokud již není, zakliknout možnost automatického spouštění doplňku při startu systému a také povolit tzv. hlídacího psa, tedy služba kontrolující správný běh aplikace a v případě chyby doplněk restartuje. Nyní je potřeba doplněk poprvé spustit.

Po prvním spuštění jsou vygenerovány nové konfigurační soubory v cestě /config/appdaemon. Pro přidání nové aplikace je třeba přesunout danou aplikaci, službu vzdáleného přístupu, do složky /config/appdaemon/apps. Druhým krokem je konfigurace AppDaemonu souborem /config/appdaemon/apps/apps.yaml. Do souboru je třeba napsat konfiguraci níže.

```
remote:  
  module: remote  
  class: Remote
```

Po správném nakonfigurování AppDaemonu je třeba nakonfigurovat samotnou službu vzdáleného přístupu. Ovšem nejprve před samotnou konfigurací je třeba vygenerovat tzv. Long Life token v instalaci Home Assistantu. Tento token povolí přístup služby vzdáleného přístupu k ovládání. Ve správě Home Assistantu stačí kliknout na svého uživatele a následně srolovat na konec stránky a vytvořit nový token s dlouhou životností. V tomto kroku je důležité si token uložit pro pozdější použití!

Nyní přichází na řadu samotná konfigurace služby. To probíhá přímo v kódu, v první části označené jako konfigurace. Níže jsou popsány parametry nutné ke konfiguraci:

| Parametr         | Funkce                                                             |
|------------------|--------------------------------------------------------------------|
| HA_ip            | Lokální adresa Home Assistantu                                     |
| HA_Friendly_name | Jméno užívané v adrese při ovládání                                |
| token            | Zde je potřeba vložit vygenerovaný token s dlouhou životností      |
| broker_address   | Adresa MQTT Brokeru, tedy cloudového serveru s veřejnou IP adresou |

Poté lze službu již spustit spuštěním doplňku AppDaemon.