



F3

**Fakulta elektrotechnická
Katedra měření**

Bakalářská práce

Monitorovací stanice spotřeby elektrické energie

Petr Procházka

Otevřená Informatika - Internet věcí

Leden 2023

Vedoucí práce: Ing. Vladimír Janíček, Ph.D.

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Procházka** Jméno: **Petr** Osobní číslo: **498964**
Fakulta/ústav: **Fakulta elektrotechnická**
Zadávající katedra/ústav: **Katedra měření**
Studijní program: **Otevřená informatika**
Specializace: **Internet věci**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Monitorovací stanice spotřeby elektrické energie

Název bakalářské práce anglicky:

Monitoring Station of Electricity Consumption

Pokyny pro vypracování:

- 1) Proveďte analýzu trhu se zařízeními na monitoring spotřeby el. energie.
- 2) Navrhněte vlastní koncept zařízení měřící spotřebu. Využijte vhodné moduly.
- 3) Obslužný software bude umožňovat dálkový monitoring provozních parametrů, správu, statistickou analýzu spotřeby a nákladů, vizualizaci.
- 4) Zařízení bude schopné komunikovat přes MQTT protokol a bude integrovatelné do systémů chytré domácnosti.
- 5) Sestrojte funkční prototyp. Ověřte jeho funkce.
- 6) Porovnejte dosažené parametry s několika komerčními výrobky z pohledu dostupných funkcí.

Seznam doporučené literatury:

- 1) Electronics Projects with the ESP8266 and ESP32: Building Web Pages, Applications, and WiFi Enabled Devices, Cameron N., ISBN 1484263359
- 2) Developing IoT Projects with ESP32: Automate your home or business with inexpensive Wi-Fi devices, Oner V.O., ISBN 1838641165
- 3) Energy Management in Buildings, Thorpe D., ISBN 9780367787448

Jméno a pracoviště vedoucí(ho) bakalářské práce:

Ing. Vladimír Janíček, Ph.D. katedra mikroelektroniky FEL

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) bakalářské práce:

Datum zadání bakalářské práce: **14.02.2023**

Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce:

do konce letního semestru 2023/2024

Ing. Vladimír Janíček, Ph.D.
podpis vedoucí(ho) práce

podpis vedoucí(ho) ústavu/katedry

prof. Mgr. Petr Páta, Ph.D.
podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

Student bere na vědomí, že je povinen vypracovat bakalářskou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v bakalářské práci.

Datum převzetí zadání

Podpis studenta

Poděkování / Prohlášení

Chtěl bych poděkovat svému vedoucímu Ing. Vladimíru Janíčkoví, Ph.D. za odborné vedení, za pomoc a rady při zpracování této práce. Rovněž bych chtěl poděkovat své rodině a všem přátelům, kteří mě podporovali nejen během psaní této práce, ale hlavně po celou dobu studia.

Prohlašuji, že jsem předloženou práci vypracoval samostatně a že jsem uvedl veškeré použité informační zdroje v souladu s Metodickým pokynem o dodržování etických principů při přípravě vysokoškolských závěrečných prací.

V Praze dne 22. 5. 2023

.....

Abstrakt / Abstract

Předmětem této bakalářské práce je návrh možného zařízení pro měření spotřeby elektrické energie v domácnosti. V úvodní části jsou představeny veličiny a jednotky pro jednofázové sítě. Následně je provedena analýza trhu zařízení měřících spotřebu a na základě získaných poznatků je popsána vize vlastního prototypu. V další části je vybrána vhodná hardwarová platforma a způsob zpracování naměřených dat. Poté následuje praktická část práce, kde je podrobně popsán proces návrhu prototypu včetně vlastní desky plošných spojů a pouzdra vytištěného na 3D tiskárně. Rovněž je popsán vývoj firmwaru pro monitorovací stanici. V poslední části je popsáno testování prototypu a porovnání dosažených výsledků s komerčními zařízeními.

Klíčová slova: ESP32, internet věcí, spotřeba elektrické energie, vizualizace dat, měření.

The subject of this bachelor's thesis is the design of a possible device for measuring household electricity consumption. In the introductory part, quantities related to consumption measurement are presented. Subsequently, an analysis of the consumption measuring devices market is carried out and, based on the knowledge gained, the vision of the own prototype is described. In the next part, a suitable hardware platform and the method of processing the measured data are selected. This is followed by the practical part of the work, where the prototype design process is described in detail, including its own printed circuit board and a case printed on a 3D printer. The firmware development for the monitoring station is also described. The last part describes the testing of the prototype and the comparison of the achieved results with commercial devices.

Keywords: ESP32, Internet of Things, power consumption, data visualization, measurement.

Title translation: Monitoring Station of Electricity Consumption

Obsah /

| | | | |
|--|-----------|--|--|
| 1 Úvod | 1 | | |
| 2 Měření spotřeby - veličiny a jednotky | 2 | | |
| 2.1 Jednofázový harmonický průběh | 2 | | |
| 2.1.1 Okamžitý výkon | 2 | | |
| 2.1.2 Činný výkon | 3 | | |
| 2.1.3 Imaginární (jalový) výkon | 3 | | |
| 2.1.4 Zdánlivý výkon | 3 | | |
| 2.1.5 Účinník | 4 | | |
| 2.1.6 Energie | 4 | | |
| 2.2 Spotřebiče a jejich charakteristika | 5 | | |
| 3 Dostupná zařízení na trhu | 6 | | |
| 3.1 Analýza zařízení do zásuvky | 6 | | |
| 3.1.1 Porovnání parametrů a dostupných funkcí | 7 | | |
| 3.1.2 Zatížení a měření spotřeby | 7 | | |
| 3.1.3 Konektivita a vzdálené ovládání | 7 | | |
| 3.1.4 Vlastní aplikace a systémy chytré domácnosti | 7 | | |
| 3.1.5 Kalkulace nákladů | 8 | | |
| 3.1.6 Otevřené API | 8 | | |
| 3.2 Analýza zařízení do rozvodné skříně | 8 | | |
| 3.2.1 Porovnání parametrů a dostupných funkcí | 9 | | |
| 3.2.2 Zatížení a měření spotřeby | 9 | | |
| 3.2.3 Konektivita a vzdálené spínání | 9 | | |
| 3.2.4 Vlastní aplikace a systémy chytré domácnosti | 9 | | |
| 3.2.5 Kalkulace nákladů | 10 | | |
| 3.2.6 Otevřené API | 10 | | |
| 3.3 Shrnutí dostupných produktů | 10 | | |
| 3.4 Vize monitorovací stanice | 10 | | |
| 4 HW platforma | 11 | | |
| 4.1 Výběr vhodného MCU | 11 | | |
| 4.2 ESP32 | 11 | | |
| 4.2.1 Periferie ESP32 | 12 | | |
| 4.3 Výběr platformy pro ukládání a vizualizaci dat | 12 | | |
| 4.4 Raspberry Pi 4 | 13 | | |
| 4.5 Měřicí modul spotřeby | 14 | | |
| 4.5.1 Napětí a proud | 14 | | |
| 4.5.2 Činný výkon a energie | 14 | | |
| 4.5.3 Účinník a frekvence | 15 | | |
| 4.5.4 Komunikační protokol | 15 | | |
| 4.5.5 Rozdíl napětí logických úrovní | 15 | | |
| 5 Zpracování a přenos dat | 16 | | |
| 5.1 Internet věcí | 16 | | |
| 5.2 IoT jako služba | 16 | | |
| 5.3 IoT vs. TCP/IP | 16 | | |
| 5.4 Datová pipeline | 17 | | |
| 5.4.1 Sběr dat | 18 | | |
| 5.4.2 Přenos dat | 18 | | |
| 5.4.3 Uložení dat | 18 | | |
| 5.4.4 Vizualizace dat | 19 | | |
| 5.5 Zvolená forma komunikace | 19 | | |
| 5.5.1 Volba protokolu pro přenos zpráv | 19 | | |
| 5.6 MQTT | 20 | | |
| 6 Realizace monitorovací stanice | 21 | | |
| 6.1 Popis simulovaného prostředí | 21 | | |
| 6.2 Vývojová deska | 22 | | |
| 6.3 Hardwarové komponenty | 22 | | |
| 6.3.1 Měřicí modul PZEM-004T V3 | 23 | | |
| 6.3.2 Obousměrný převodník logických úrovní | 23 | | |
| 6.3.3 Napájecí zdroj | 23 | | |
| 6.4 Schéma zapojení | 23 | | |
| 6.5 Návrh desky plošných spojů | 24 | | |
| 6.5.1 Schéma obvodového zapojení | 24 | | |
| 6.5.2 Schéma desky plošných spojů | 25 | | |
| 6.6 Ochranné pouzdro jednotky | 25 | | |
| 6.6.1 3D návrh pouzdra | 26 | | |
| 6.6.2 3D tisk | 26 | | |
| 6.7 Osazení desky a montáž | 27 | | |
| 7 Programování jednotky a nastavení serveru | 28 | | |
| 7.1 Program pro monitorovací stanici | 28 | | |
| 7.1.1 Konfigurace | 28 | | |
| 7.1.2 Připojení k Wi-Fi a MQTT serveru | 29 | | |
| 7.1.3 Hlavní smyčka | 30 | | |

| | | |
|-----------|---|-----------|
| 7.1.4 | Měření veličin a ode- slání dat | 31 |
| 7.1.5 | Příjem konfiguračních příkazů | 32 |
| 7.1.6 | Nastavení zúčtovacího období | 32 |
| 7.1.7 | Odhad spotřeby | 32 |
| 7.2 | Nastavení serveru chytré domácnosti | 32 |
| 7.2.1 | Použitý HW a instalace . . | 33 |
| 7.2.2 | Nastavení databáze In- fluxDB | 33 |
| 7.2.3 | Nastavení MQTT brokera . | 33 |
| 7.2.4 | Nastavení nástroje Grafana | 34 |
| 8 | Testování prototypu | 35 |
| 8.1 | Spuštění monitorovací stanice . | 35 |
| 8.2 | Vytvoření přehledu v ná- stroji Grafana | 35 |
| 8.3 | Export nástěnky na hlavní přehled | 36 |
| 8.4 | Testovací provoz | 37 |
| 9 | Zhodnocení dosažených výsledků | 38 |
| 9.1 | Náklady na výrobu prototypu . | 38 |
| 9.2 | Porovnání s komerčně do- stupnými produkty | 38 |
| 9.2.1 | Srovnání z hlediska měřených veličin | 38 |
| 9.2.2 | Srovnání z hlediska do- stupných funkcí | 39 |
| 9.3 | Celkové zhodnocení | 39 |
| 10 | Závěr | 40 |
| | Literatura | 41 |
| | A Fotografie prototypu | 45 |
| | B Seznam zkratk | 47 |

Tabulky / Obrázky

| | |
|--|----|
| 3.1 Chytré zásuvky s měřením spotřeby | 6 |
| 3.2 Srovnání chytrých zásuvek..... | 7 |
| 3.3 Chytré senzory spotřeby do rozvaděče | 8 |
| 3.4 Srovnání senzorů do rozvaděče .. | 9 |
| 9.1 Cena použitých komponent. ... | 38 |
| 2.1 Zdánlivý výkon S a vztah mezi P a Q..... | 4 |
| 2.2 Okamžitý výkon a jeho rozklad na činný a jalový výkon. ... | 5 |
| 3.1 Chytrá zásuvka..... | 6 |
| 3.2 Shelly Pro 3EM..... | 8 |
| 4.1 ESP32-WROOM moduly..... | 12 |
| 4.2 Raspberry Pi 4. | 13 |
| 4.3 Blokový diagram modulu PZEM-004T při použití proudové sondy. | 14 |
| 5.1 TCP/IP model a protokoly IoT. | 17 |
| 5.2 Základní datová pipeline IoT zařízení..... | 18 |
| 5.3 Model komunikace publish-subscribe v MQTT. | 20 |
| 6.1 Schéma zapojení simulovaného prostředí..... | 21 |
| 6.2 Popis pinů modulu NodeMCU. | 22 |
| 6.3 Schéma zapojení převodníku logických úrovní. | 23 |
| 6.4 Schéma zapojení prototypu. ... | 24 |
| 6.5 Schéma obvodového zapojení. . | 25 |
| 6.6 Schéma desky plošných spojů. . | 25 |
| 6.7 3D model pouzdra..... | 26 |
| 7.1 Funkce setup()..... | 29 |
| 7.2 Konfigurační portál pro připojení k Wi-Fi. | 30 |
| 7.3 Hlavní řídicí smyčka..... | 31 |
| 7.4 Funkce pro měření veličin. | 31 |
| 7.5 Funkce pro odhad spotřeby. ... | 32 |
| 7.6 Konfigurace InfluxDB. | 33 |
| 7.7 Konfigurace MQTT. | 33 |
| 7.8 Konfigurace ukládání entit do InfluxDB. | 34 |
| 8.1 Vytvoření databázového dotazu. | 35 |
| 8.2 Ukázka výsledného přehledu v nástroji Grafana. | 36 |
| 8.3 Ukázka vizualizace entity pomocí Home Assistant karty. ... | 37 |

Kapitola 1

Úvod

Bakalářská práce je vypracována v období války na Ukrajině a s tím související vysoké inflace a krize na energetickém trhu, která způsobila rapidní růst cen energií. Společně s tím se v posledních letech Evropská unie zaměřuje na snižování emisí skleníkových plynů a celkově šetrný přístup k životnímu prostředí. To vše jsou důvody, ať už ekonomické nebo ideologické, proč bychom se měli pokusit optimalizovat spotřebu elektrické energie. K tomu je vhodné znát její rozložení v čase a také vědět, bez kterého spotřebiče se v daný čas lze obejít, a naopak který je nezbytný. S nástupem obnovitelných zdrojů energie je v dnešní době důležitější než kdy jindy umět plánovat spotřebu v čase. Zjednodušeně řečeno chceme odložit spotřebu, která není časově kritická, na čas, kdy je v síti přebytek energie. V menším měřítku chceme například odložit spuštění pračky na čas, kdy je podle spotových cen levnější elektřina.

K tomu nám může pomoci zařízení, které spotřebu měří, data odesílá a následně jsou prezentována uživateli. V ideálním případě zařízení, které je otevřené a je navrženo pomocí časem osvědčených protokolů a hardwarových platforem. Tento přístup má několik výhod. Zařízení používající otevřené a ověřené protokoly umožňují větší flexibilitu a interoperabilitu mezi sebou a ostatními systémy. Díky tomu mohou uživatelé snadno propojit svá zařízení a využívat je v různých kombinacích, což zvyšuje jejich využitelnost a uživatelskou hodnotu. Další výhodou otevřených protokolů a hardwarových platforem je větší transparentnost a důvěryhodnost. Otevřené protokoly jsou veřejně dostupné a mohou být přezkoumány a ověřeny nezávislými vývojáři. To umožňuje lepší kontrolu nad provozem a bezpečností zařízení.

Naproti tomu proprietární řešení mohou být problematická z hlediska kompatibility a bezpečnosti. Uživatelé mohou být nuceni používat pouze zařízení od jednoho výrobce, což je omezující z hlediska výběru a flexibility. Proprietární protokoly a hardwarové platformy mohou být navíc závislé na určitém softwaru, což může být problémem při migraci na jiný systém. Z těchto důvodů je patrné, proč má smysl představit koncept zařízení pro měření spotřeby, u kterého je již při návrhu myšleno na použití osvědčených protokolů, hardwarových platforem a snadnou integraci s různými systémy.

Kapitola 2

Měření spotřeby - veličiny a jednotky

Měření spotřeby elektrické energie v domácnosti lze realizovat na 2 místech. Prvním, a z dlouhodobého hlediska vhodnějším místem je měření přímo v rozvodné skříni, kde lze měřit spotřebu celé domácnosti a všechnu elektroniku umístit na standardizovanou DIN lištu.[1] Podle konkrétního zapojení lze měřit spotřebu buď na jedné fázi, nebo případně i na všech 3 fázích. Je zapotřebí na to mít dostatek prostoru, což se ale často ukáže jako problém, protože rozvodné skříně nejsou dimenzovány na přidání dalších zařízení. Druhým místem je měření elektrické spotřeby pro konkrétní elektrickou zásuvku, a tedy měření jen na 1 fázi. Výhodou tohoto řešení je, že zařízení není omezeno volným prostorem, jako je tomu v rozvodné skříni, a je tudíž možné měřit odděleně spotřebu konkrétního spotřebiče nebo celé skupiny.

Tento koncept se zaměřuje na měření elektrické spotřeby na 1 fázi. V následujících podkapitolách je popsán stručný úvod do problematiky měření elektrické spotřeby v jednofázové síti se střídavým proudem. Jsou popsány základní veličiny a jevy týkající se tohoto tématu.

2.1 Jednofázový harmonický průběh

Harmonický zdroj napětí

$$v = \sqrt{2}V\sin(\omega t)$$

napájející lineární zátěž bude produkovat harmonický proud (předpokládá se fázový posun za napětím) o velikosti

$$i = \sqrt{2}I\sin(\omega t - \phi)$$

, kde V je efektivní hodnota napětí (V) a I je efektivní hodnota proudu (A).

Základní veličiny použité ve vztazích:

ω je úhlová frekvence $2\pi f$ (rad/s)

f je frekvence napájecí soustavy (Hz)

ϕ je fázový posun mezi napětím a proudem (rad)

t je čas (s)

2.1.1 Okamžitý výkon

Okamžitý výkon p , jednotka [W], je dán vztahem

$$p = vi$$

$$p = p_a + p_q$$

, kde

$$p_a = P[1 - \cos(2\omega t)]; \quad P = VI \cos \phi$$

$$p_q = -Q \sin(2\omega t); \quad Q = VI \sin \phi$$

Složka p_a je okamžitý činný výkon. Je tvořen činnou složkou proudu (tj. složkou, která je ve fázi s napětím). Okamžitý činný výkon p_a je míra toku energie

$$w_a = \int_{t_0}^t p_a dt$$

Tato energie jednosměrně proudí ze zdroje do zátěže a je nezáporná, $p_a \geq 0$.

Složka p_q je okamžitý jalový výkon, který vzniká z jalové složky proudu. Okamžitý jalový výkon p_q je míra toku energie

$$w_q = \int_{t_0}^t p_q dt$$

Tato energie osciluje mezi zdrojem a obvodem ve formě magnetického pole kolem vodičů, elektrického pole kondenzátorů a také ve formě mechanické energie v rotorech motorů. Průměrná hodnota je rovna 0 a čistý přenos energie do zátěže je také nulový. Přesto způsobuje ztráty energie (Jouleovo teplo a vířivé proudy) ve vodičích.[2]

■ 2.1.2 Činný výkon

Činný výkon lze popsat jako výkon, který byl využit spotřebičem ke konání skutečné práce. U výpočtu elektrické spotřeby je nejdůležitější, protože podle něj jsou domácnosti účtovány dodavatelem.

Činný výkon P , jednotka [W], je průměrná hodnota okamžitého výkonu během časového intervalu τ až $\tau + kT$. [2]

$$P = \frac{1}{kT} \int_{\tau}^{\tau+kT} p dt$$

, kde T je perioda, k je \mathbb{N} a τ je začátek intervalu. P se také rovná průměru p_a v intervalu τ až $\tau + kT$, protože průměr p_q je roven nule.

■ 2.1.3 Imaginární (jalový) výkon

Jalový výkon Q , jednotka [var] (voltampér reaktanční), působí ztráty činného výkonu a je snahou ho minimalizovat pomocí tzv. kompenzátorů účinnosti. Součtem činného a jalového výkonu získáme zdánlivý výkon. Velikost jalového výkonu Q se rovná amplitudě kmitajícího okamžitého jalového výkonu p_q a je dána vztahem

$$Q = VI \sin \phi$$

Pokud je zátěž induktivní, pak $Q > 0$. Pokud je zátěž kapacitní, pak $Q < 0$. To znamená, že když je proud fázově posunut za napětím, je $\phi > 0$ a naopak.

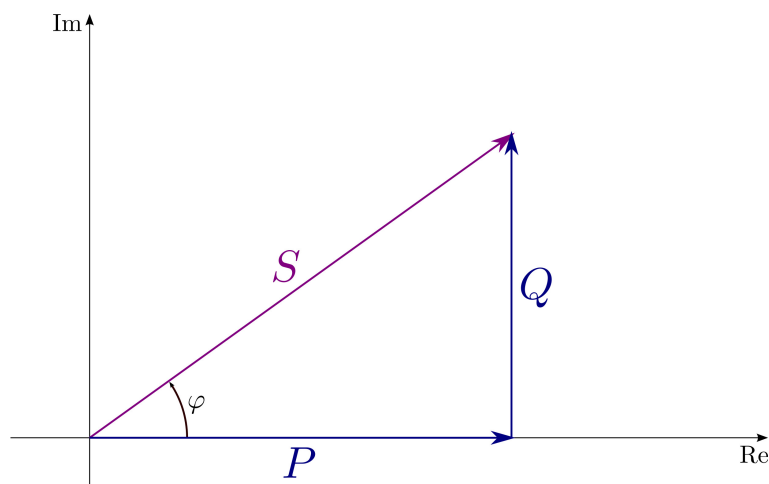
■ 2.1.4 Zdánlivý výkon

Zdánlivý výkon S , jednotka [VA] (voltampér) je součinem efektivní hodnoty napětí a efektivní hodnoty proudu.

$$S = VI$$

$$S = \sqrt{P^2 + Q^2}$$

Zdánlivý výkon jednofázové zátěže lze interpretovat jako maximální činný výkon, který lze přenášet stejným vedením při zachování konstantního efektivního napětí zátěže V a úbytku výkonu napájecího vedení konstantní (tj. konstantní efektivní proud I).



Obrázek 2.1. Zdánlivý výkon S , ϕ značí fázový posun. [3]

■ 2.1.5 Účinník

Účinník, značen $\cos \phi$ lze interpretovat jako poměr mezi energií přenášenou do zátěže a maximální energií, která by mohla být přenesena při zachování stejných ztrát ve vedení, tj. poměr činného a zdánlivého výkonu.

$$\cos \phi = \frac{P}{S}$$

Hodnota účinníku je v rozmezí od nuly do jedné, při účinníku rovném jedné je zdánlivý výkon roven činnému výkonu (fázový posun je nulový). Při nulovém účinníku je zdánlivý výkon roven jalovému výkonu, zátěž je buď čistě induktivní (fázový posun je $\phi = +\frac{\pi}{2}$, tj. proud se zpožďuje za napětím) nebo čistě kapacitní (fázový posun je $\phi = -\frac{\pi}{2}$, tj. napětí se zpožďuje za proudem).

■ 2.1.6 Energie

Elektrická energie je forma energie, kterou lze charakterizovat jako schopnost elektromagnetického pole konat elektrickou práci. Maxwellovy rovnice jsou klíčové pro popis elektromagnetického pole a tedy toho, jak elektromagnetické pole přenáší elektrickou energii.

Spotřebovaná elektrická energie E , jednotka [J] nebo ekvivalentní [Ws] (v praxi násobná jednotka kWh), je dána vztahem

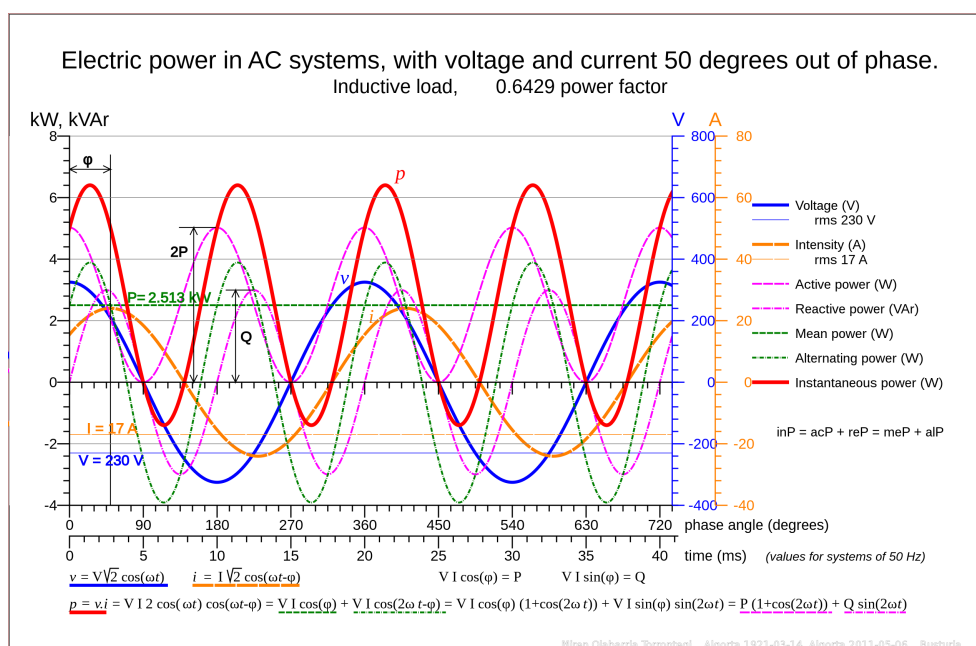
$$E = Pt$$

, kde P je činný výkon t je čas. Energie se zpravidla přeměňuje na energii mechanickou, tepelnou nebo světelnou.

2.2 Spotřebiče a jejich charakteristika

Domácí spotřebiče mohou mít různou závislost na proudu a napětí – VA charakteristiku. Nejzákladnějším typem je odporová charakteristika. Ta převládá v případech, kdy se spotřebič chová jako rezistor, tedy s rostoucím napětím lineárně roste odebíraný proud. Jedná se hlavně o spotřebiče s topným tělesem, jako je například rychlovarná konvice, žehlička, ohřívač vody, elektrický sporák nebo klasická žárovka. Okamžitý výkon takového spotřebiče je v každém okamžiku roven součinu okamžitého napětí a okamžitého proudu. U těchto typů spotřebičů nedochází k fázovému posunu mezi napětím a proudem a okamžitý výkon je až na výjimky kladný a odpovídá činnému výkonu, tedy účinník je roven 1.

Na druhé straně máme spotřebiče s netriviálním průběhem proudu. Ať už jsou to spotřebiče se spínaným zdrojem s impulzní proudovou charakteristikou jako jsou například počítače, nebo spotřebiče s fázově posunutým proudem vzhledem k napětí. Zde jsou příkladem spotřebiče s induktivní zátěží, kde je proud opožděn za napětím. Jedná se převážně o spotřebiče s elektrickým motorem – pračky, ventilátory. Opačným případem je spotřebič s kapacitní zátěží, kde je napětí opožděno za proudem.



Obrázek 2.2. Okamžitý výkon a jeho rozklad na činný a jalový výkon; když proud zaostává za napětím o 50 stupňů. [4]

Pro nejpřesnější hodnotu skutečné spotřeby je tedy nutné brát v potaz všechny náležitosti, které jsou pro spotřebiče střídavé sítě specifické, a měřit přesně činný výkon.

Kapitola 3

Dostupná zařízení na trhu

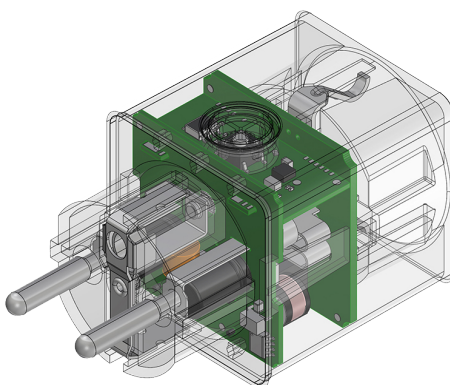
V této kapitole je shrnut průzkum trhu v oblasti zařízení měřících elektrickou spotřebu. Jak již bylo naznačeno v předchozí kapitole, měření spotřeby lze realizovat na dvou místech. Buď přímo z jedné elektrické zásuvky, nebo měřením konkrétních okruhů v rozvodné skříni. V první části jsou představena zařízení do zásuvky a zhodnoceny jejich parametry. V druhé části jsou zhodnoceny zařízení na DIN lištu v rozvodné skříni. V závěru je shrnutí průzkumu této oblasti.

3.1 Analýza zařízení do zásuvky

Chytrých zásuvek s měřením elektrické spotřeby je na trhu velké množství. Jejich počet jde do desítek variant od mnoha výrobců. Jsou také cenově nejdostupnější variantou. Jejich cena začíná na zhruba 300 Kč a lepší modely dosahují ceny až 1200 Kč za kus. V tabulce 3.1 je výčet analyzovaných zařízení společně s jejich maloobchodní cenou.

| Název zařízení | Výrobce | Cena [Kč] | Dále pod názvem |
|------------------------------|-------------|-----------|-----------------|
| Immax NEO LITE zásuvka v2[5] | Immax | 299 | Immax NEO LITE |
| Tesla Smart Plug[6] | TESLA Solar | 349 | Tesla Plug |
| TP-LINK Tapo P115[7] | TP-LINK | 484 | TP-LINK Tapo |
| Eve Energy[8] | Eve Systems | 1089 | Eve Energy |

Tabulka 3.1. Chytré zásuvky s měřením spotřeby.



Obrázek 3.1. Chytrá zásuvka [9]

| Funkce | Immax NEO LITE | Tesla Plug | TP-LINK Tapo | Eve Energy |
|------------------------------|----------------|--------------|--------------|------------|
| Měření spotřeby | Ano | Ano | Ano | Ano |
| Maximální zatížení [W] | 3680 | 3680 | 3680 | 2500 |
| Vzdálené spínání a plánování | Ano | Ano | Ano | Ano |
| Konektivita | Wi-Fi, BT | Wi-Fi, BT | Wi-Fi, BT | BT, Thread |
| Vlastní aplikace | Android, iOS | Android, iOS | Android, iOS | iOS |
| Samsung SmartThings | Ano | Ne | Ano | Ano |
| Google Home | Ne | Ne | Ano | Ano |
| Apple Home | Ne | Ne | Ne | Ano |
| Kalkulace nákladů | Ne | Ne | Ano | Ano |
| Otevřené API(MQTT, apod.) | Ne | Ne | Ne | Ne |

Tabulka 3.2. Chytré zásuvky s měřením spotřeby.

■ 3.1.1 Porovnání parametrů a dostupných funkcí

Chytré zásuvky se liší v mnoha aspektech, ať už je to způsob komunikace, kompatibilita s domácími asistenty nebo maximální možný výkon. V tabulce 3.2 jsou srovnány dostupné funkce chytrých zásuvek a jejich klíčové parametry.

■ 3.1.2 Zatížení a měření spotřeby

Maximální zatížení všech zásuvek kromě Eve Energy činí 3680 W, respektive 16 A při 230 V. Eve Energy je omezena na 2500 W. Všechna zmíněná zařízení umožňují měření spotřeby elektrické energie a aktuální výkon. Žádný výrobce neuvádí přesnost měření. Měření dalších veličin, jako je napětí, proud nebo účinník, není u žádného produktu k dispozici.

■ 3.1.3 Konektivita a vzdálené ovládání

Všechna zařízení kromě Eve Energy disponují technologií Wi-Fi, u Tesla Plug a TP-LINK Tapo je to verze 802.11b/g/n - respektive Wi-Fi 4. U Immax NEO LITE není verze specifikována. Bluetooth používají zařízení k prvnímu připojení a konfiguraci. Zařízení Eve Energy se liší v tom, že používá Bluetooth i jako protokol pro přenos dat a ovládání zařízení. Eve Energy oproti ostatním zařízením, které používají Wi-Fi, využívá protokol Thread.

Thread je otevřený, energeticky úsporný, bezdrátový síťový protokol, který je navržen pro spojení zařízení Internetu věcí způsobem, který je zabezpečený, spolehlivý a škálovatelný. Je založený na 6LoWPAN - protokol IPv6 pro low-power komunikaci v osobních sítích[10], standard IEEE 802.15.4. Thread používá decentralizovanou architekturu, což znamená, že každé zařízení v síti může komunikovat s ostatními zařízeními přímo, bez nutnosti spoléhat na centrální uzel nebo bránu. To činí síť Thread odolnější proti výpadkům než jiné IoT protokoly.

Eve Energy je tedy lépe propojitelné se zařízeními od ostatních výrobců, které používají protokol Thread. Není nutná kompatibilita všech zařízení s jedním centrálním bodem. Všechna zařízení lze vzdáleně ovládat a sledovat. To je možné přes konkrétní mobilní aplikaci pro dané zařízení nebo v aplikacích chytré domácnosti.

■ 3.1.4 Vlastní aplikace a systémy chytré domácnosti

Všechna zařízení disponují vlastní aplikací. Ta se používá pro první připojení, ovládání a vizualizaci spotřeby. Eve Energy má aplikaci pouze pro iOS, ostatní zařízení i na operační systém Android. Pokud není zařízení kompatibilní s některým ze systémů chytré domácnosti, je jeho integrovatelnost velmi omezená. Na trhu zařízení chytré domácnosti existuje široká škála výrobců a jejich vlastních aplikací, z tohoto důvodu je často obtížné spolu zařízení propojovat a automatizovat.

Integraci do některého ze systémů chytré domácnosti zmíněného v tabulce 3.2 podporuje kromě Tesla Plug každé zařízení. Immax NEO LITE je kompatibilní pouze se Samsung SmartThings, TP-LINK Tapo je kromě Apple Home kompatibilní se všemi. Eve Energy lze integrovat do každého výše zmíněného systému.

■ 3.1.5 Kalkulace nákladů

Výpočet nákladů za spotřebovanou elektřinu umožňuje pouze TP-LINK Tapo a Eve Energy. Zobrazují velikost spotřebované energie za zvolené časové období a umožňují základní kalkulace cen za spotřebovanou elektřinu. Na základě současné průměrné spotřeby také umí vypočítat odhadovanou roční spotřebu.

■ 3.1.6 Otevřené API

Žádné výše zmíněné zařízení nemá otevřené API pro integraci do vlastních, nebo open-source řešení chytré domácnosti. Tento nedostatek lze obejít pomocí uživatelsky programovaných knihoven na zařízení konkrétního výrobce. Příkladem je knihovna pro integraci TP-LINK Tapo do platformy Home Assistant.¹

■ 3.2 Analýza zařízení do rozvodné skříně

Na rozdíl od chytrých zásuvek je množství zařízení pro měření spotřeby v rozvodné skříni značně omezené. Filozofie těchto produktů se liší od chytrých zásuvek. Cílí na měření domácnosti jako celku, měření řádově vyšších výkonů či nasazení v průmyslu. Pohybuje se od 1500 Kč za téměř nekompatibilní zařízení s moderními domácími asistenty po 3500 Kč za kvalitní řešení. Hlavním hráčem v tomto segmentu je firma Shelly. V tabulce 3.3 je výčet dále zkoumaných zařízení.

| Název zařízení | Výrobce | Cena [Kč] | Dále pod názvem |
|-----------------------------------|---------|-----------|-----------------|
| Shelly Plus 1PM[11] | Shelly | 480 | Shelly Plus 1PM |
| AEOTEC Home Energy Meter Gen5[12] | Aeotec | 2500 | Aeotec Gen5 |
| Zamel MEW-01 Lite[13] | Zamel | 3300 | Zamel Lite |
| Shelly Pro 3EM[14] | Shelly | 3400 | Shelly Pro 3EM |

Tabulka 3.3. Chytré senzory spotřeby do rozvaděče.



Obrázek 3.2. Shelly Pro 3EM. [15]

¹ <https://github.com/petretiandrea/home-assistant-tapo-p100>

■ 3.2.1 Porovnání parametrů a dostupných funkcí

Zařízení do rozvodné skříně mají schopnost měřit řádově větší výkon než zařízení do zásuvky. Je to způsobeno tím, že k měření používají neinvazivní proudové sondy, které jsou schopné měřit proudy o velikosti desítek ampér. Oproti chytrým zásuvkám neposkytují tolik chytrých funkcí, jakou je například funkce spínače, nebo plán spínání. V tabulce 3.4 jsou srovnány dostupné funkce senzorů spotřeby do rozvaděče a jejich klíčové parametry.

| Funkce | Shelly Plus 1PM ² | Aeotec Gen5 | Zamel Lite | Shelly Pro 3EM |
|-------------------------------|------------------------------|--------------|--------------|----------------|
| Měření spotřeby | Ano | Ano | Ano | Ano |
| Maximální měřitelný proud [A] | 16 | 60 | 100 | 120 |
| Vzdálené spínání a plánování | Ano | Ne | Ne | Ne |
| Konektivita | Wi-Fi, BT | 868 MHz | Wi-Fi | Wi-Fi, LAN, BT |
| Vlastní aplikace | Android, iOS | Android, iOS | Android, iOS | Android, iOS |
| Samsung SmartThings | Ano | Ano | Ne | Ano |
| Google Home | Ano | Ne | Ne | Ano |
| Apple Home | Ne | Ne | Ne | Ne |
| Kalkulace nákladů | Ne | Ne | Ano | Ne |
| Otevřené API(MQTT, apod.) | Ano | Ne | Ne | Ano |

Tabulka 3.4. Srovnání senzorů do rozvaděče.

■ 3.2.2 Zatížení a měření spotřeby

Maximální měřitelný výkon všech zařízení kromě Shelly Plus 1PM je dán typem proudové sondy. Shelly Plus 1PM se od ostatních zařízení v této kategorii liší tím, že je funkčně podobné s chytrou zásuvkou, měří proud protékající skrz, ale je navrženo pro umístění za zásuvku, nebo přímo do rozvaděče. Všechna zmíněná zařízení umožňují měření spotřeby elektrické energie a aktuální výkon. Pouze Shelly Pro 3EM a Zamel Lite dle výrobců měří i další veličiny, jako je účinník, zdánlivý výkon atd. Přesnost měření u zařízení Aeotec Gen5 uvádí výrobce více než 99 %, pokud je proud větší než 0.05A[16]. U zařízení Zamel Lite udává výrobce přesnost $\pm 2\%$ [17]. Shelly uvádí přesnost pouze u varianty Pro 3EM. Pro voltmetr $\pm 1\%$ a pro ampérmetr $\pm 1\%$ pro rozsah měření 2 - 120 A, $\pm 2\%$ pro rozsah 1 - 2 A, $\pm 5\%$ pro rozsah 0 - 1A[18].

■ 3.2.3 Konektivita a vzdálené spínání

Všechna zařízení kromě Aeotec Gen5 používají jako primární technologii Wi-Fi. Shelly Plus 1PM navíc Bluetooth a Shelly Pro 3EM Bluetooth a ethernetové připojení pomocí LAN portu. Aeotec 5G používá protokol Z-Wave, který umožňuje zařízením Internetu věci komunikovat mezi sebou a zároveň s centrálním uzlem pomocí rádiové komunikace, v Evropě na frekvenci 868.42 MHz[19]. Z-Wave umožňuje násobně větší dosah, v řádu desítek metrů oproti technologii Wi-Fi, za cenu menší datové propustnosti.

Vzdáleně spínat a automatizovat lze pouze pomocí Shelly Plus 1PM. Je to dáno tím, že účelem ostatních zařízení v tabulce 3.4 je měření spotřeby pomocí neinvazivních sond a není možné pomocí nich spínat přívod elektrické energie.

■ 3.2.4 Vlastní aplikace a systémy chytré domácnosti

Rovněž v této kategorii má každé zařízení svou vlastní aplikaci. Problém kompatibility a kvantity aplikací od různých výrobců je zde stejný. Shelly a Aeotec vyrábí širší spektrum produktů, proto použití jejich aplikací dává smysl v těch případech, kdy uživatel vlastní více produktů od stejné značky.

² Shelly Plus 1PM má blíže k chytré zásuvce, je ale koncipováno jako zařízení za zásuvku nebo do rozvaděče, proto je pro porovnání zařazeno sem.

Kapitola 4

HW platforma

V této kapitole je navržena vhodná hardwarová platforma a potřebné periferie k měření elektrické spotřeby. V úvodu je představen MCU ESP32, který je řídicí jednotkou celého zařízení měřícího spotřebu. Poté je představeno zařízení Raspberry Pi 4, na kterém je provozován server chytré domácnosti, se kterým ESP32 komunikuje. Použití Raspberry Pi není nutnou podmínkou, data lze ukládat a vizualizovat za splnění určitých podmínek bezplatně na cloudové infrastruktuře poskytovatelů třetích stran. Na konec je představen senzor měření spotřeby.

4.1 Výběr vhodného MCU

Klíčové hledané parametry vhodného MCU pro monitorovací stanici elektrické spotřeby jsou: bezdrátová konektivita, snadná programovatelnost, cenová dostupnost, komunitní podpora. V tomto konceptu je vybírán MCU s podporou Wi-Fi, protože pro použití v domácnosti je to vhodný protokol z důvodu minimální konfigurace a potřeby dalšího hardwaru.

Na trhu mikrokontrolérů s Wi-Fi figuruje celá řada výrobců. Za zmínku stojí řada STM32Wx od STMicroelectronics, ta se primárně zaměřuje na krátkodosahové moduly s podporou BLE (Bluetooth Low Energy), Thread, ZigBee, nebo moduly na delší vzdálenosti s podporou protokolů LoRaWAN a Sigfox. Dalším výrobcem je NXP Semiconductors s podobným zaměřením jako STMicroelectronics. Od firmy Microchip Technology je k dispozici řada mikrokontrolérů PIC32MZ-W1. V neposlední řadě je na trhu produkt ESP32 od firmy Espressif Systems.

Toto je jen několik příkladů mikrokontrolérů s Wi-Fi, které se běžně používají v různých aplikacích. Na trhu je dnes k dispozici mnoho dalších MCU s podobnými funkcemi.

Pro návrh konceptu monitorovací stanice spotřeby elektrické energie byla zvolena platforma ESP32 od firmy Espressif Systems. Mezi hlavní nevýhody ostatních produktů patří: vysoká cena, absence vývojových sad, komplexnost. V seznamu jsou shrnuty hlavní argumenty, které zdůvodňují výběr této platformy.

- cenová dostupnost oproti ostatním výrobcům
- široká škála kompaktních vývojových sad, pro testování prototypů
- mnoho periférií
- jednoduché naprogramování. (ArduinoIDE, PlatformIO, MicroPython, FreeRTOS a další)
- početná uživatelská základna - komunitní podpora
- dostatečný výkon

4.2 ESP32

ESP32 je mikrokontrolér vyráběný společností Espressif Systems od roku 2016. Jedná se o levný systém na čipu (SoC) s nízkou spotřebou, dostupnou cenou a značným výkonem. Výrobcem čipu je tchajwanská společnost TSMC.

ESP32 je nástupcem mikrokontroléru ESP8266. Jednou z klíčových vlastností ESP32 je dvoujádrový procesor, který umožňuje efektivní multitasking a vysokou úroveň výkonu. Procesor je založen na 32bitové architektuře Xtensa LX6 a může pracovat s taktem až 240 MHz. Tento mikrokontrolér je dostupný v mnoha variantách s drobně se lišícími parametry, jako je například velikost FLASH paměti.

ESP32 má také vestavěné funkce Wi-Fi a Bluetooth, což usnadňuje připojení k bezdrátovým sítím a dalším zařízením. Modul Wi-Fi podporuje pásma 2,4 GHz a je kompatibilní se standardy 802.11 b/g/n. Modul Bluetooth podporuje klasické protokoly Bluetooth a Bluetooth Low Energy (BLE), což umožňuje širokou škálu aplikací, například domácí automatizaci nebo zařízení IoT. [21]



Obrázek 4.1. Vlevo ESP32 s PCB anténou, vpravo ESP32 s konektorem pro externí anténu.[21]

4.2.1 Periferie ESP32

Pro SoC ESP-WROOM-32 jsou dostupné následující periferie:

- 32 univerzálních vstupně výstupních pinů (GPIO)
- 4 rozhraní SPI
- 3 rozhraní UART
- 2 rozhraní I2C
- 8 výstupů pro pulzně-šířkovou modulaci PWM
- 18 AD převodníků s rozlišením 12 bitů
- 2 DA převodníky s rozlišením 8 bitů

4.3 Výběr platformy pro ukládání a vizualizaci dat

Monitorovací stanice odesílá data na zařízení, které je ukládá a následně na nich provádí další úkony. Mezi tyto úkony patří vizualizace, obohacení dat o další informace, detekce anomálií a další. Toto zařízení nemusí být v lokální síti monitorovací stanice a je možné ho nahradit využitím cloudové infrastruktury třetích stran. Využití cloudové infrastruktury je u řady poskytovatelů bezplatné za předpokladu, že jsou dodrženy určité limity.

Limitem může být maximální velikost uložených dat, rychlost toku dat do databáze nebo maximální počet dní, po které jsou data uchována.

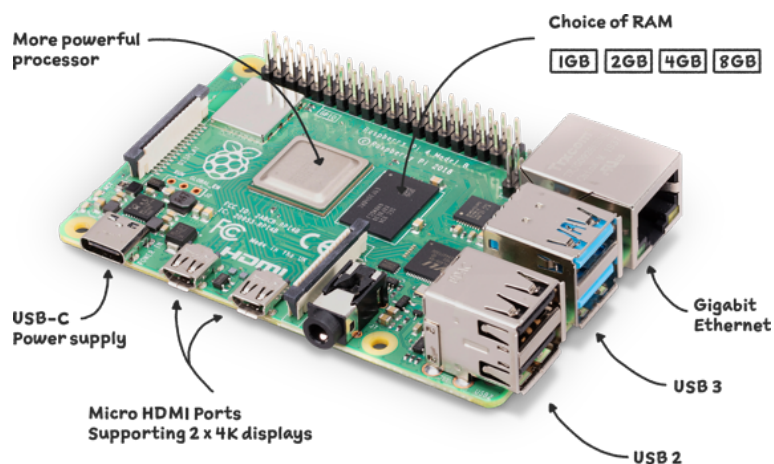
Použití vlastního zařízení je způsob, jak se těmto limitům vyhnout. Velikost uchovaných dat je dána velikostí úložiště zařízení. Obě zařízení jsou v jedné lokální síti, a nemusí tedy spoléhat na třetí stranu.

Jako zařízení určené pro zpracování dat a běh potřebného softwaru byl zvolen mini-počítač Raspberry Pi 4, mezi uživateli velmi populární zařízení pro provoz domácích serverů. V seznamu jsou shrnuty hlavní argumenty, které zdůvodňují výběr této platformy:

- malá velikost
- nízká spotřeba
- dostatečný výkon
- početná uživatelská základna - komunitní podpora
- přijatelná cena
- bez ventilátorů - tiché

4.4 Raspberry Pi 4

Raspberry Pi 4 je miniaturní jednodeskový počítač, který byl uveden na trh v roce 2019 jako nástupce úspěšných mini počítačů Raspberry Pi 2 a 3. Je vyvíjen Raspberry Pi Foundation.



Obrázek 4.2. Raspberry Pi 4.[22]

Předností Raspberry Pi 4 je 64bitový čtyřjádrový procesor BCM2711 o taktu 1,8 GHz a LPDDR4 SDRAM o velikosti 1 - 8 GB. To umožňuje provozovat energeticky úsporný domácí server, na kterém lze vybudovat infrastrukturu potřebnou k vizualizaci dat ze zařízení chytré domácnosti. Raspberry Pi 4 dále disponuje Wi-Fi 5(802.11ac), Gigabitovým Ethernetem a Bluetooth 5.0. [22]

Další přednosti Raspberry Pi 4:

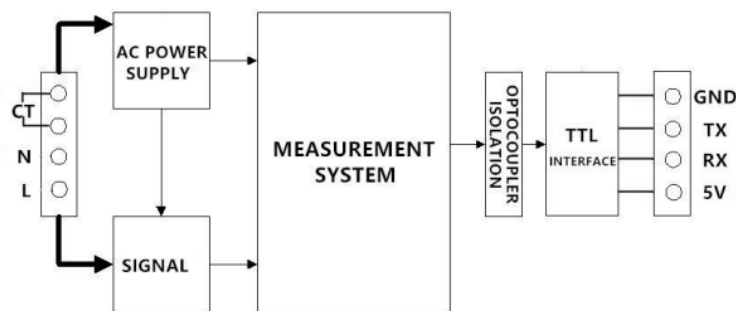
- 2 porty USB 3.0, 2 porty USB 2.0
- 40 pinový **GPIO** header
- 2 rozhraní **micro-HDMI**
- grafický standard **OpenGL ES 3.0**
- **MicroSDHC** slot pro operační systém a uchování dat
- **USB-C** napájecí port

4.5 Měřicí modul spotřeby

Pro přesné měření spotřeby je nutné měřit okamžité napětí a okamžitý proud. Na trhu existuje mnoho zařízení pro měření jak napětí, tak pro měření proudu, které jsou pomocí standardních sběrnic kompatibilní se zvoleným MCU ESP32. Nevýhodou tohoto řešení je potřebná kalibrace senzorů a implementace softwaru, který počítá požadované veličiny z naměřených hodnot. Pravdou je, že pro řadu senzorů jsou již veřejně dostupné uživatelské knihovny, které tento problém řeší.

Alternativou k použití dvou senzorů pro měření zvláště napětí a zvláště proudu je použití multifunkčního modulu. Jelikož návrh vlastního hardware není předmětem této práce a pro obecnou jednoduchost je zvolen cenově dostupný modul PZEM-004T V3 od firmy Peacefair. Ten umožňuje měření 230 V síťového napětí, proudu, účinníku, činného výkonu a frekvence.

Modul pracuje ve 2 režimech. V prvním proud protéká přímo skrz a je možné ho měřit v rozsahu 0 - 10A. V druhém režimu je k modulu připojena externí proudová sonda. Ta umožňuje měření v rozsahu 0 - 100 A. [23]



Obrázek 4.3. Blokový diagram modulu PZEM-004T při použití proudové sondy.

4.5.1 Napětí a proud

Napětí je měřeno v rozsahu 80 - 260V. Rozlišení je 0.1 V a přesnost je 0.5 %. Bez použití externí sondy proudu zmíněné výše je proud měřen od 0.01 A. S použitím sondy je proud měřen od 0.02 A. Rozlišení je 0.001 A a přesnost měření je 0.5 %.

4.5.2 Činný výkon a energie

Bez použití externí sondy proudu je měřen v rozsahu 0 - 2300 W. S použitím sondy je měřen v rozsahu 0 - 23 kW. Začíná měřit při 0.4 W s rozlišením 0.1 W. Přesnost měření je 0.5 %. Od 1000 W a výše vrací pouze celé číslo. Spotřebovanou elektrickou energii měří modul do výše 9999.99 kWh. Rozlišení je 1 Wh a přesnost 0.5 %.

■ 4.5.3 Účinnost a frekvence

Účinnost je měřena v rozsahu 0 - 1 s rozlišením 0.01 a přesností měření 1 %. Frekvence je měřena v rozsahu 45 - 65 Hz s rozlišením 0.1 Hz a přesností 0.5 %.

■ 4.5.4 Komunikační protokol

Modul na fyzické vrstvě používá převodník UART na RS485. Přenosová rychlost je 9600 znaků za sekundu. Na aplikační vrstvě používá modul protokol Modbus. Pomocí tohoto protokolu je možné změnit adresu konkrétního modulu, číst hodnoty z registrů nebo vynulovat aktuální měření spotřebované energie.

■ 4.5.5 Rozdíl napětí logických úrovní

Rozhraní TTL modulu PZEM-004T vyžaduje externí napájení o velikosti 5 V. Piny MCU ESP32 ale nejsou 5V tolerantní. Pro správnou funkčnost je nutné použít logický převodník 3V3 na 5V.

Kapitola 5

Zpracování a přenos dat

Tato kapitola nejprve definuje pojem Internet věcí jako takový, následně popisuje, jakým způsobem jsou u zařízení IoT zpracovávána data a jaké k tomu lze použít nástroje a služby. Jsou popsány části pipeline zpracování dat. V závěru je popsána zvolená forma komunikace pro hlavní zaměření této práce - senzor spotřeby.

5.1 Internet věcí

Nejprve je dobré si definovat, co je to Internet věcí (IoT). Existuje mnoho definic, ale ne každá obsahuje všechny aspekty nebo je dostatečně stručná. Výzkumná firma Gartner definuje Internet věcí následujícím způsobem.[24]

"Síť fyzických objektů, které obsahují vestavěnou technologii pro komunikaci a snímání nebo interakci s jejich vnitřními stavy nebo vnějším prostředím."

5.2 IoT jako služba

Využití IoT jako služby znamená využití infrastruktury, softwaru a platformy třetí strany, která umožňuje propojení a komunikaci mezi zařízeními IoT, ukládání dat a jejich analýzu.

Tento přístup umožňuje využívat výhody Internetu věcí bez nutnosti počátečních investic do vývoje a údržby infrastruktury. Poskytovatelé Internetu věcí jako služby obvykle nabízejí řadu možností, včetně připojení zařízení, správy a ukládání dat, analýzy dat v reálném čase a nástrojů pro vizualizaci. Pomocí těchto služeb mohou zákazníci rychle a snadno nasadit řešení IoT, která jsou přizpůsobena jejich specifickým potřebám a požadavkům.

Internet věcí jako služba zahrnuje následující části:

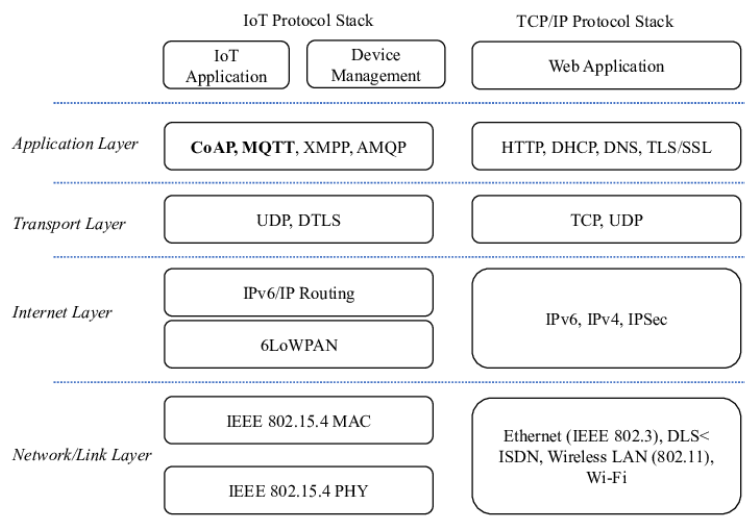
- **Hardware** – fyzická zařízení, senzory, akční členy
- **Konektivita** - technologie, která umožňuje zařízením připojení k dané síti a komunikovat v ní (internet nebo jiné např. proprietární sítě - SigFox)
- **Platforma** – platforma slouží nejen k ukládání, analýze, vizualizaci dat, ale také k ovládání zařízení a automatizaci některých procesů
- **Aplikace** – nástroje postavené nad platformou poskytující rozšířené funkce na sesbíraných datech, např. predikce událostí, složitější výpočty nad daty, vzdálené monitorování atd.

5.3 IoT vs. TCP/IP

Kromě TCP/IP modelu i pro Internet věcí existují rodiny protokolů, které definují komunikaci na jednotlivých vrstvách mezi zařízeními v síti.

Rodina IoT protokolů se zaměřuje zejména na specifické požadavky zařízení Internetu věcí, jako je nízká spotřeba energie, omezená datová propustnost a výpočetní výkon nebo časté výpadky konektivity. Naproti tomu je model TCP/IP univerzální, a používá se pro širokou škálu aplikací. Model TCP/IP je stavěn na velké datové toky v reálném čase, dostatek výpočetního výkonu a až na výjimky kontinuální konektivitu.

Na obrázku 5.1 jsou pro jednotlivé vrstvy TCP/IP modelu zobrazeny ekvivalenty na straně IoT. Pro každou vrstvu existuje v současnosti mnoho protokolů a další vznikají.

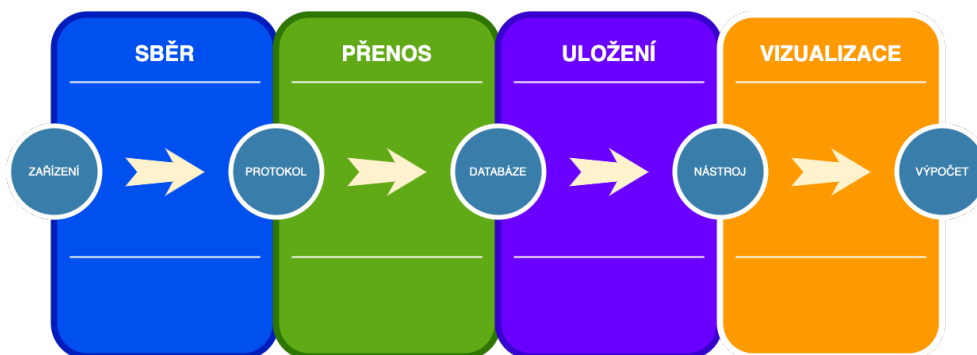


Obrázek 5.1. TCP/IP model a protokoly IoT.[25]

Mezi populární standardy lze v současnosti zařadit LoRaWAN, SigFox, ZigBee nebo NB-IoT. Tyto protokoly nejsou specifické pro pouze jednu vrstvu výše zobrazeného modelu, ale jako celistvé řešení od fyzické vrstvy po vrstvu aplikační.

5.4 Datová pipeline

Datová pipeline je abstrakční model, který popisuje jednotlivé fáze toku dat z IoT zařízení. Základní model pipeline má 4 části. Jak již bylo zmíněno výše, je to sběr dat, přenesení dat, uložení dat, a nakonec jejich vizualizace. Na obrázku 5.2 jsou zobrazeny jednotlivé fáze, které jsou v dalších podkapitolách popsány blíže.



Obrázek 5.2. Základní datová pipeline IoT zařízení.

■ 5.4.1 Sběr dat

Sběr dat spočívá v zaznamenání vnitřních stavů IoT zařízení a stavu vnějšího prostředí. Mezi vnitřní stavy se řadí například vytížení CPU, zbývající volná paměť RAM, teplota zařízení a další. Sběr dat popisující vnitřní stavy zařízení může odhalit trendy, které by mohly vést k pádu běžícího programu, nebo jiné nežádoucí události. Mezi vnější stavy lze zařadit informace o vnějším prostředí a stavech akčních členů IoT zařízení. Jedná se o naměřená data ze senzorů, kupříkladu teplota, vlhkost, tlak, napětí, úhel otočení motorů atd.

■ 5.4.2 Přenos dat

K přenosu dat dochází mezi IoT zařízeními a databází, kde jsou data uchovávána. Přenášena mohou být buď nestrukturovaná data nebo data předem upravená.

Přenos nestrukturovaných dat je výhodný u velkých infrastruktur, kde dochází k přesunu velkého množství dat od různých typů zařízení, při různých rychlostech a v jiném formátu. Tato data jsou strukturována až v další fázi. Populárním open-source nástrojem pro řízení datových toků v IoT infrastruktuře je Apache Kafka. Je to distribuovaný systém odolný proti chybám, který má vysokou datovou propustnost, škálovatelnost a je používán v kritických aplikacích.[26]

Data mohou být přenášena v reálném čase nebo ukládána do bloků, které jsou přenášeny ze zařízení v pravidelných intervalech. Přenos dat v pravidelných intervalech se používá u bateriových zařízení, kde je kladen důraz na nízkou spotřebu. Bezdrátový modul je aktivní jen po minimální nutnou dobu k přenesení naměřených dat. Na druhé straně data ze serverů, která obsahují informace o vytížení, síťovém provozu atp., jsou přenášena v reálném čase. Výhodou tohoto přístupu je možnost okamžitě reagovat na neočekávané stavy.

■ 5.4.3 Uložení dat

Data jsou přenesena do databáze, ke které se zařízení autentizuje pomocí svého unikátního tokenu. Vhodným typem databáze pro data z IoT senzoru je databáze časových řad, značena TSDB (time series database). Mezi populární databáze časových řad patří: InfluxDB, TimescaleDB, Prometheus.

Mezi hlavní výhody patří:

- **Efektivní ukládání dat** - Databáze časových řad jsou optimalizovány pro ukládání a dotazování se na velké množství datových bodů. To znamená, že jsou efektivní pro ukládání dat, která se časem mění, jako jsou data ze senzorů, systémové logy, webová aktivita atd.

- **Rychlé dotazování** - Databáze časových řad jsou optimalizovány pro rychlé dotazování na data v určitém časovém rozmezí
- **Snadná analýza dat** - Databáze časových řad mají vestavěné funkce pro analýzu dat, jako jsou agregace, průměry, lineární regrese atd.
- **Snadná integrace s jinými nástroji** - Databáze časových řad mají obvykle jednoduchá API rozhraní a podporují mnoho formátů dat, což umožňuje snadnou integraci s dalšími nástroji pro analýzu a vizualizaci dat.

■ 5.4.4 Vizualizace dat

Vizualizace dat z IoT senzoru je důležitá pro monitorování a analýzu chování zařízení. Vytváření interaktivních nástěnek umožňuje zobrazovat data z různých zdrojů na jednom místě, sledovat chování senzorů v reálném čase a vyhodnocovat data v závislosti na různých faktorech. Díky tomu je možné rychle reagovat na problémy a minimalizovat potenciální škody. Pomocí vizualizace časových řad je možné v datech hledat trendy a lépe predikovat jinak neočekávané události a reagovat na ně.

Data uložená v databázi lze vizualizovat pomocí populárních nástrojů, jako je Grafana nebo Kibana.

■ 5.5 Zvolená forma komunikace

Pro přenos dat z monitorovací stanice je zvolena technologie Wi-Fi. Pro integraci do lokální chytré domácnosti je to nejjednodušší řešení, které nevyžaduje další hardware a jeho konfiguraci. Využívá již zavedené lokální sítě a pro tuto implementaci je to nejvhodnější forma komunikace, kterou MCU ESP32 podporuje.

Wi-Fi je bezdrátová technologie, která pomocí rádiové komunikace v nelicencovaném pásmu (2,4 nebo 5 GHz) umožňuje zařízením se připojit k lokálnímu přístupovému bodu (AP). Přístupovým bodem je v domácnosti obecně směrovač (router). Organizace Wi-Fi Alliance, která vlastní ochrannou známku, definuje Wi-Fi jako jakýkoli bezdrátový lokální síťový produkt založený na standardu IEEE 802.11.

Jak již bylo zmíněno výše, data by následně mohla být odesílána na endpoint vzdálené databáze pomocí protokolu HTTP a tím by byla část na straně domácnosti vyřešena. V tomto případě je server, který zpracovává a vizualizuje data, rovněž umístěn v lokální síti.

■ 5.5.1 Volba protokolu pro přenos zpráv

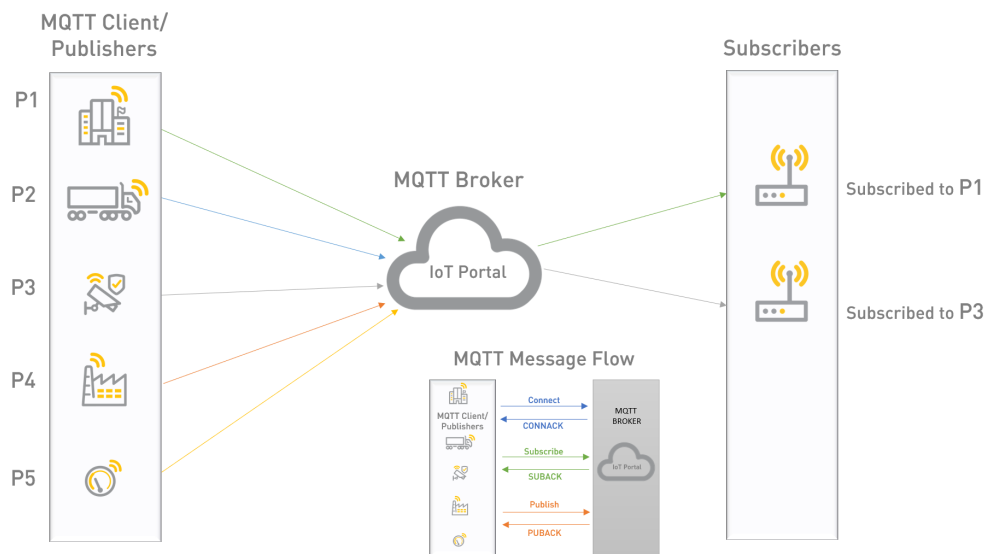
Pro komunikaci server-server nebo aplikace-server v internetu je standardem protokol HTTP. Je na něm postaveno fungování webových prohlížečů. Funguje na principu požadavku od klienta a odpovědi od serveru. Mezi jeho výhody patří univerzálnost a množství již dostupných nástrojů, frameworků a návodů. Při použití platformy jako služby v IoT ho poskytovatelé cloudových infrastruktur používají pro svá API. Na druhé straně mezi jeho nevýhody patří velká režie, která v IoT aplikacích, kdy dochází k pravidelnému přenosu malého objemu dat, není žádoucí[27].

Jako protokol pro zasílání zpráv mezi monitorovací stanicí a serverem je zvolen protokol MQTT. Tento protokol je v IoT velmi populární z několika důvodů:

- **Snadná implementace** - protokol byl historicky navržen pro MCU s malým výkonem
- **Minimální režie** - kombinace malého výkonu s nestabilním připojením je důvod, proč byl protokol navržen s minimální nutnou režii
- **Vysoká spolehlivost** - protokol využívá mechanismu QoS
- **Jednoduchá obousměrná komunikace mezi zařízeními**

5.6 MQTT

MQTT (Message Queuing Telemetry Transport) je otevřený, efektivní a spolehlivý protokol pro komunikaci mezi zařízeními. Protokol MQTT byl vytvořen pro efektivní přenos dat v prostředí omezených sítí a hardwarových prostředků, jako je komunikace v kontextu M2M (Machine to Machine) a IoT[28]. Protokol MQTT používá publish-subscribe model, což znamená, že zprávy jsou posílány na tzv. témata (topics) a odběratelé (subscribers) si zvolí, na kterých tématech přijímají zprávy. Klienti jsou mezi sebou vzájemně odděleni a komunikaci zprostředkovává tzv. MQTT broker. Broker filtruje příchozí zprávy a přeposílá je na konkrétní témata k odběratelům. Princip komunikace publish-subscribe je zobrazen na obrázku 5.3.



Obrázek 5.3. Znázornění komunikace publish-subscribe mezi zařízeními.[29]

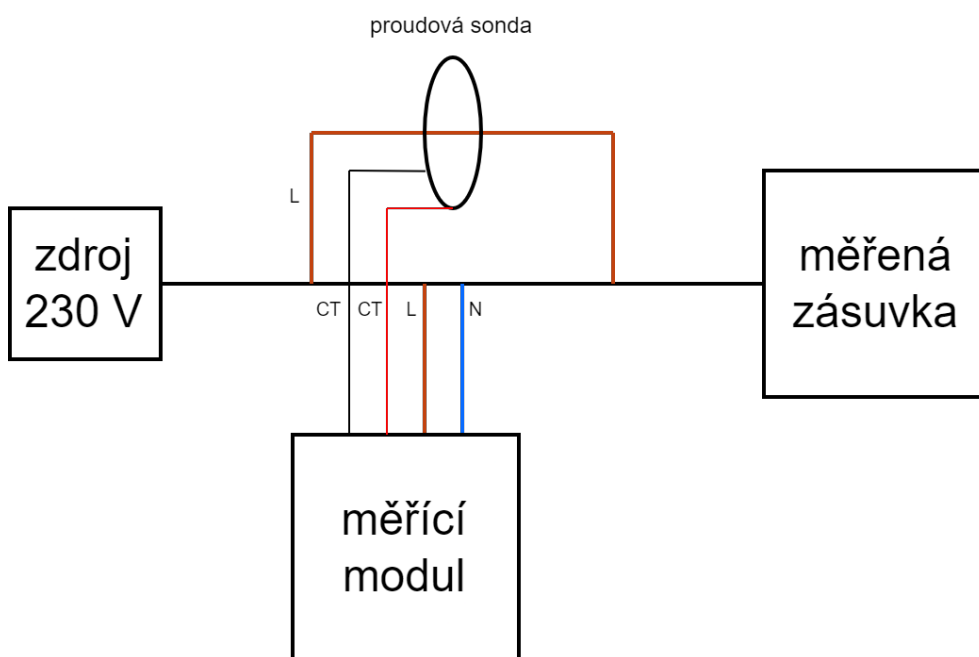
Kapitola 6

Realizace monitorovací stanice

Tato kapitola se zabývá realizací monitorovací stanice spotřeby elektrické energie. Popisuje proces od prvotního konceptu po funkční zařízení v ochranném zapouzdření. Dále jsou ukázány použité postupy a technologie společně s popisem použitých komponent a materiálů.

6.1 Popis simulovaného prostředí

K otestování funkčnosti prototypu monitorovací stanice nebude měřena spotřeba celé domácnosti, ale jen určité skupiny spotřebičů. Toho je dosaženo upravením prodlužovacího kabelu, na který je měřená skupina spotřebičů připojena. Díky tomuto přístupu není nutné zasahovat do rozvodné skříňe a zároveň je možné otestovat různé spotřebiče pouhým přidáním upraveného prodlužovacího kabelu před daný spotřebič. Schéma zapojení simulovaného prostředí je ukázáno na obrázku 6.1.



Obrázek 6.1. Schéma zapojení simulovaného prostředí.

Fázový vodič (L) a nulový vodič (N) z přívodního kabelu jsou rozděleny pomocí Wago svorek a zapojeny do měřicího modulu, aby bylo možné měřit napětí. K měření proudu je použita proudová sonda. Ta musí obepínat pouze jeden vodič, L nebo N, na který je připojena zátěž. Výstupy ze sondy jsou rovněž zapojeny do měřicího modulu. Ochranný vodič je nepřerušen a vede do měřené zásuvky.

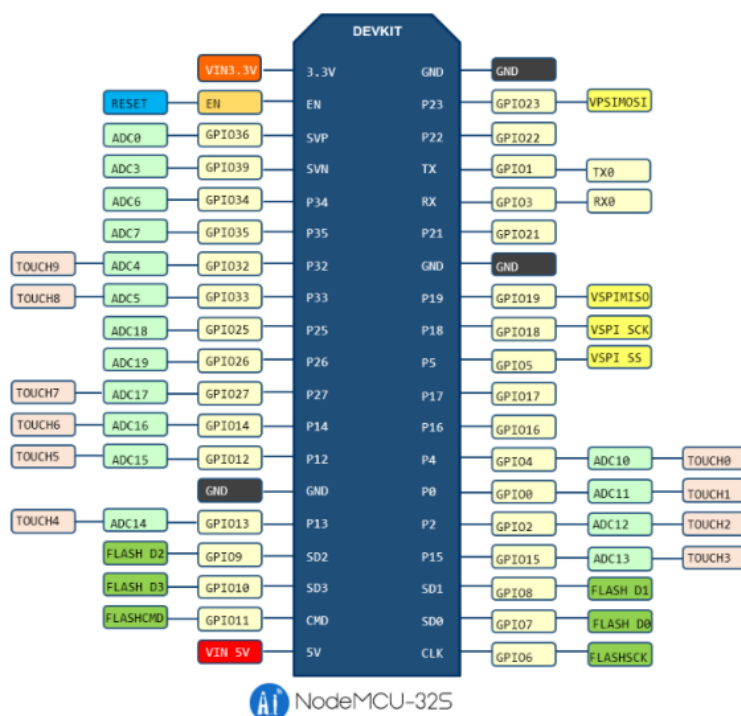
6.2 Vývojová deska

Řídící jednotkou monitorovací stanice je vývojová deska NodeMCU-32S. Jedná se o finančně dostupnou, uživatelsky podporovanou vývojovou desku s čipem ESP32, který byl vybrán ve čtvrté kapitole. Deska obsahuje SoC (System on Chip) ESP32-Wroom-32s a má 38 pinů, jejichž funkce jsou znázorněny na obrázku 6.2.

Deska obsahuje převodník CP2102 z UART na USB od firmy Silicon Labs. Díky tomuto převodníku lze desku jednoduše programovat z počítače, ke kterému je připojena pomocí micro-USB kabelu. Desku lze napájet několika způsoby. Prvním z nich je přivedením 3.3 V na VIN3.3V, druhým z nich je přivedení 5 V na VIN 5V a poslední možností je přivedení 5 V pomocí již zmíněného micro-USB konektoru. Pokud je zvolena varianta napájení pomocí 5 V, dojde ke snížení napětí pomocí lineárního napěťového regulátoru AMS1117 na 3.3 V. Tento krok je nutný, protože čip ESP32 pracuje na napětí 3.3 V a pokud by k regulaci nedošlo, mohlo by dojít k poškození periférií čipu.[30]

Dalším přidaným prvkem na vývojové desce k modulu ESP32 jsou 2 LED. První dioda je červená a signalizuje, zda je deska připojena k napájení. Je natvrdo připojena v napájecí cestě a nelze ji ovládat pomocí programu. Druhá dioda je modrá a je možné ji pomocí programu ovládat.

Na závěr vývojová deska obsahuje 2 tlačítka. Tlačítko BOOT, jehož podržením se deska přepne do programovacího režimu a tlačítko EN, kterým lze desku restartovat.



Obrázek 6.2. Popis pinů modulu NodeMCU-32S.[30]

6.3 Hardwarové komponenty

V následujících podkapitolách jsou popsány hardwarové komponenty, které byly použity při testování funkčnosti prototypu monitorovací stanice společně s vývojovou deskou NodeMCU-32S.

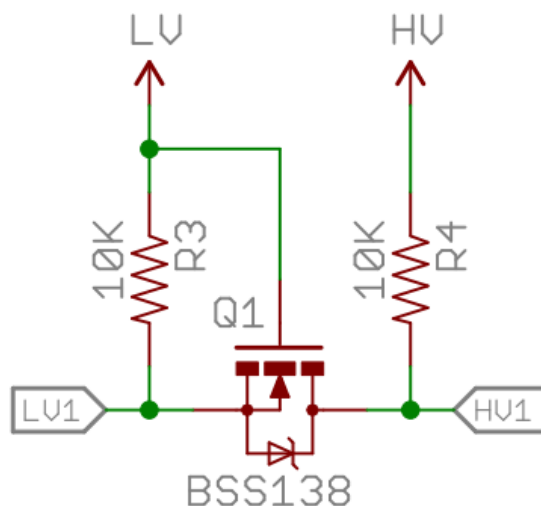
6.3.1 Měřicí modul PZEM-004T V3

Je použit měřicí modul PZEM-004T V3, který byl vybrán ve čtvrté kapitole. Ten lze zakoupit v ochranném plastovém pouzdře, který snižuje riziko úrazu elektrickým proudem. Dále je s tímto modulem použita proudová sonda ve variantě, kdy je možné sondu rozepínat a je tedy snadné ji odebrat nebo přidat bez nutnosti zasahovat do obvodu.

Měřicí modul komunikuje s MCU přes pasivní UART rozhraní a je nutné k modulu přivést napájení 5 V. Totéž napětí používá modul při komunikaci a proto je nutné mezi modulem a MCU použít převodník logických úrovní.

6.3.2 Obousměrný převodník logických úrovní

Pro obousměrný převod logických úrovní mezi 3.3 V a 5 V je na každém kanálu použit obvod s unipolárním N-MOSFET tranzistorem BSS138. Schéma zapojení pro jeden kanál je na obrázku 6.3.



Obrázek 6.3. Schéma zapojení převodníku logických úrovní pro jeden kanál.[31]

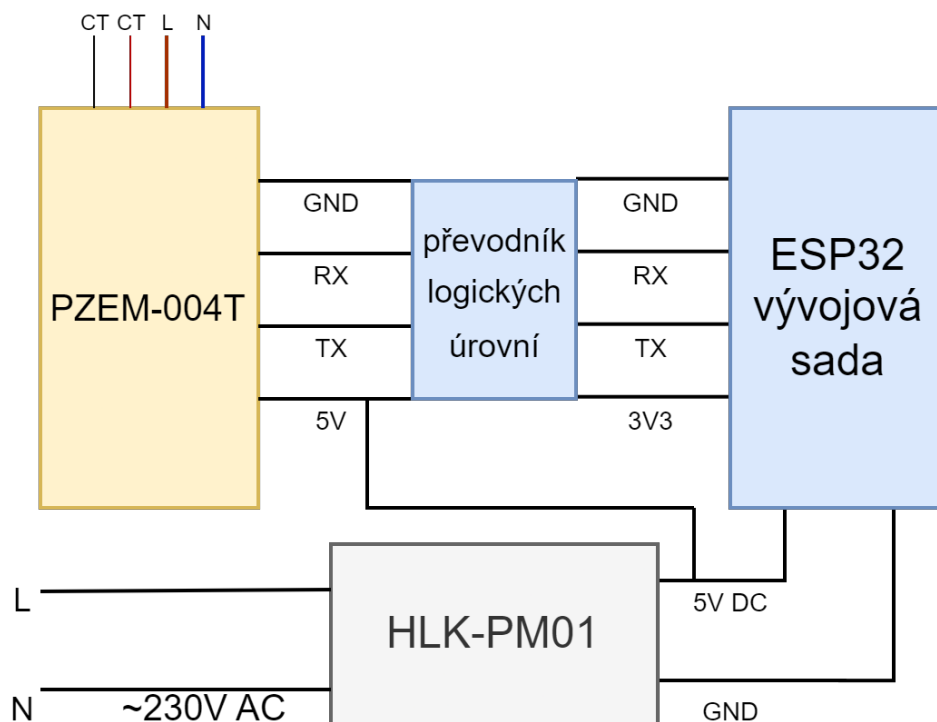
6.3.3 Napájecí zdroj

K napájení monitorovací stanice je použit spínaný zdroj HLK-PM01 od firmy Hi-Link. Je to zdroj o výkonu 3 W s širokým rozsahem vstupního napětí od 10 V do 240 V AC. Výstupní napětí je 5 V. Zdroj je v ochranném obalu a disponuje CE certifikací.

Výrobek tedy splňuje bezpečnostní, zdravotní i environmentální požadavky EU. Označení CE se vyžaduje u výrobků, které jsou uváděny na trh EU, ať už byly vyrobeny kdekoli na světě. [32]

6.4 Schéma zapojení

Na obrázku 6.4 je znázorněno schéma propojení jednotlivých komponent. Spínaný zdroj HLK-PM01 je připojen k střídavému napětí 230 V, které převádí na stejnosměrné napětí 5 V. Tím je napájena vývojová sada NodeMCU-32S. UART rozhraní na modulu PZEM-004T je pasivní a je tedy nutné i měřicí modul připojit k 5 V od spínaného zdroje. Samotný modul ESP32 používá ke komunikaci piny GPIO16 a GPIO17.



Obrázek 6.4. Schéma zapojení prototypu.

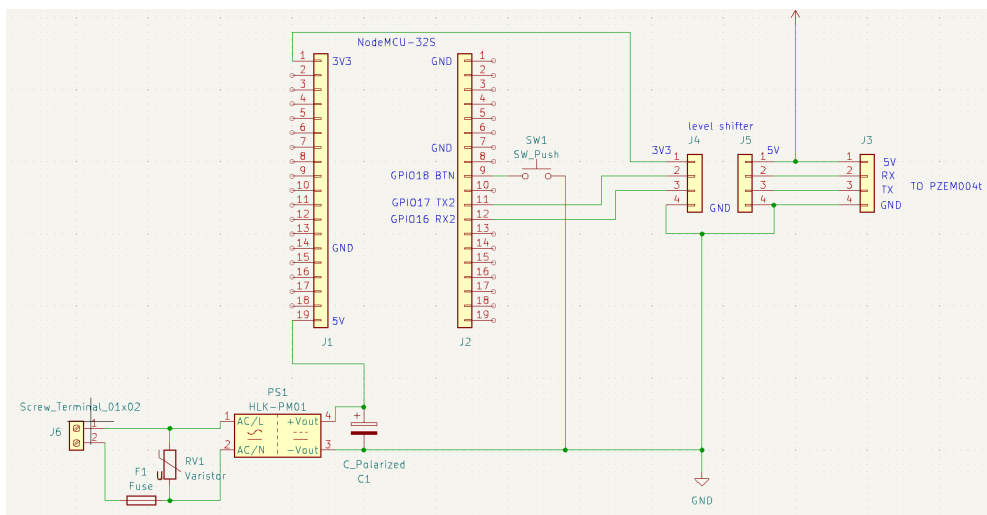
V další fázi byla ověřena funkčnost tohoto zapojení. Pro ověření byl použit jednoduchý program, který naměřené hodnoty periodicky vypisoval na sériový monitor.

6.5 Návrh desky plošných spojů

Prototyp byl otestován na nepájivém poli a následně byl přenesen na prototypovou desku. K ní byly jednotlivé komponenty připájeny. Řešení se nezdálo být dosti robustní a trvalé. Bylo přistoupeno k návrhu vlastní desky plošných spojů (DPS). Schematické zapojení a následný návrh DPS byl proveden v open-source programu KiCad.

6.5.1 Schéma obvodového zapojení

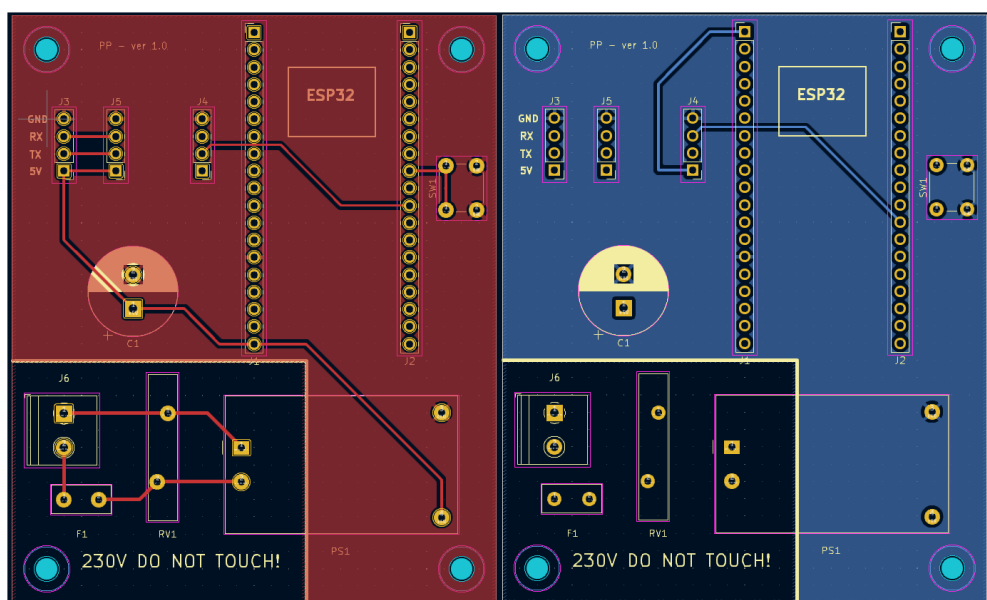
Všechny prvky byly vybrány z KiCad databáze. Některé prvky byly zrcadlově otočeny nebo jinak upraveny, aby odpovídaly finálnímu provedení. Schéma je zachyceno na obrázku 6.5. Pro větší bezpečnost byla do obvodu před spínaný zdroj umístěna pojistka KLS5-102-5ET-0250H-R na 250 mA, která by v případě příliš velkého proudu rozepnula obvod. Rovněž byl přidán varistor HITANO VCR20D431KAR, který by v případě přepětí zkratoval obvod. Ve schématu je ještě za spínaným zdrojem přidán kondenzátor pro stabilizaci zdroje. Ten ale nakonec nebyl osazen, protože obvod fungoval správně. Na pin GPIO18 bylo přidáno uživatelské tlačítko. To používá vnitřní pull-up rezistor a je připojeno k zemi.



Obrázek 6.5. Schéma obvodového zapojení.

6.5.2 Schéma desky plošných spojů

Deska plošných spojů může obsahovat mnoho vrstev v závislosti na tom, kolik propojů deska obsahuje. Na jedné vrstvě není možné překřížit dva vodiče, aniž by při tom došlo k jejich spojení. V tomto případě stačilo použít 2 vrstvy, horní a spodní. Schéma desky plošných spojů je zachyceno na obrázku 6.6. Šířka cest je pro výkonové propoje 0,5 mm a pro ostatní 0,3 mm. Kromě části na desce, kde jsou vodiče 230 V, byla vytvořena oblast mědi připojená k zemi, což zlepšuje impedanční vlastnosti desky, ale také může fungovat jako pasivní chladič. Rozměry desky jsou 86,4 mm x 71.1 mm. V rozích byly vytvořeny otvory pro šrouby velikosti M3. Deska byla vyrobena u firmy JLCPCB.



Obrázek 6.6. Schéma desky plošných spojů, nalevo horní vrstva, napravo spodní vrstva.

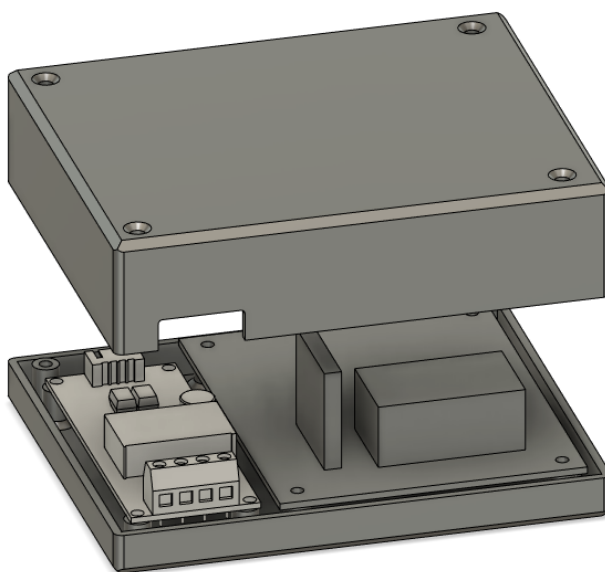
6.6 Ochranné pouzdro jednotky

Pro ochranné pouzdro byla zvolena metoda 3D tisku. 3D tisk je způsob aditivní výroby, který umožňuje vytvářet sofistikovaně tvarovaná tělesa při relativně nízkých nákladech

a v krátkém čase. Je to metoda užívaná též ve strojírenství pro rychlé prototypování. Nejběžnější způsob 3D tisku je metoda Fused Deposition Modeling (FDM)[33]. Principem FDM je natavování přídatného materiálu (filamentu) v podobě drátu a jeho vytlačování pomocí extruderu malou tryskou, která je nahřáta na požadovanou teplotu pro daný materiál.

6.6.1 3D návrh pouzdra

Pro návrh modelu byl použit program Autodesk Fusion 360[34]. Ten je možné použít zcela zdarma ve verzi pro studenty. 3D model je znázorněn na obrázku 6.7. Pouzdro se skládá z horní a spodní poloviny. Ve spodní polovině jsou umístěny sloupky pro uchycení měřicího modulu PZEM-004T a desku plošných spojů s modulem ESP32. Sloupky jsou navrženy pro velikost šroubů M3, což odpovídá velikostem otvorů v uchycených deskách. Pro připojení vodičů k měřicímu modulu je v horní polovině pouzdra navržena otvor.



Obrázek 6.7. 3D model pouzdra pro monitorovací stanici.

6.6.2 3D tisk

Z programu Autodesk Fusion360 byl exportován soubor STL, který obsahuje informace od 3D modelu. Ten byl importován do programu PrusaSlicer[35]. Ten následně soubor převede do formátu, který je pro 3D tiskárnu pochopitelný a obsahuje pokyny pro pohyby motorů společně s dalšími informacemi (čas tisku, teplota v jednotlivých fázích tisku, pauzy pro změnu filamentu). Tento typ souboru se nazývá G-code a běžně se používá v průmyslu u CNC strojů.

K tisku byla použita tiskárna Original Prusa MINI+ od české firmy Prusa Research[36], která byla v článku New York Times ohodnocena jako nejlepší 3D tiskárna za dostupnou cenu[37]. Prusa Research je globálně významná firma v oblasti 3D tisku a je výhercem mnoha prestižních ocenění.

Pro tisk byl použit materiál PETG (Glykolem modifikovaný PolyEthylenTereftalát). Je houževnatý, s dobrou tepelnou odolností. Využití pro PETG se najde hlavně při tisku různých mechanických dílů, držáků, svěráků a vodotěsných dílů (díky dobrému propojení vrstev).

6.7 Osazení desky a montáž

Vývojová sada NodeMCU-32S a převodník logických úrovní jsou umístěny do dutinkových lišt. Toto řešení je flexibilní a umožňuje vývojovou sadu vyjmout a naprogramovat nebo součástky použít v dalších projektech. Ostatní součástky jsou na desku osazeny přímo. Měřicí modul a vývojová sada s ESP32 jsou navzájem propojeny pomocí DuPont kabelů. DPS je napájena pomocí dvou vodičů, které jsou připájeny ke dvěma pinům, které slouží k měření napětí na modulu PZEM-004T. Ty jsou na DPS připojeny do šroubovací svorkovnice.

Deska plošných spojů je k spodní části pouzdra upevněna pomocí dvou M3 šroubů vlevo a vpravo pomocí dlouhých M3 šroubů se zapuštěnou hlavou, které připojují horní část pouzdra ke spodní. Měřicí modul PZEM-004T je připojen pomocí tří M3 šroubů a v levém dolním rohu je rovněž připojen dlouhým M3 šroubem se zapuštěnou hlavou k horní části pouzdra. Čtvrtý dlouhý M3 šroub s zapuštěnou hlavou má ve spodním pouzdře samostatný sloupek v levém horním rohu, protože měřicí modul PZEM-004T je na výšce kratší než DPS a otvory v horním pouzdru by nebyly symetricky umístěny.

Kapitola 7

Programování jednotky a nastavení serveru

V této kapitole je popsán vývoj programu pro monitorovací stanici a nastavení serveru chytré domácnosti pro vizualizaci a agregaci naměřených dat. Firmware pro monitorovací stanici byl vyvinut v nástroji PlatformIO[38]. Zvoleným řešením pro server je open-source systém Home Assistant[39].

PlatformIO je platforma kombinující vývojové prostředí, správce knihoven a nástroje pro kompilaci a nahrání výsledného kódu do MCU. Podporuje více než 1500 vývojových desek. Programovacím jazykem je C/C++.

Home Assistant je open-source domácí automatizační platforma, do které je možné snadno integrovat chytrá zařízení od různých výrobců a navzájem je propojovat. Home Assistant také umožňuje instalovat další rozšíření a nástroje, které jsou spuštěny paralelně a přidávají další funkce.

7.1 Program pro monitorovací stanici

Program pro monitorovací stanici je rozdělen do několika souborů. Hlavní program je v souboru *main.cpp*. Funkce týkající se měření veličin a kalkulace s daty jsou v souborech *measurement.cpp* a *measurement.h*. Funkce týkající se připojení k Wi-Fi a komunikace se serverem jsou v souborech *network.cpp* a *network.h*. V souboru *main.cpp* je kromě funkce *setup()*, kde dochází ke konfiguraci, a *loop()*, kde je hlavní smyčka programu, také inicializace globálních instancí pro připojení k Wi-Fi, odečet dat z monitorovací stanice a MQTT komunikaci. V následujících podkapitolách jsou popsány jednotlivé části programu a konkrétní funkce.

7.1.1 Konfigurace

Ke konfiguraci monitorovací stanice dochází ve funkci *setup()* v souboru *main.cpp*. Je zde zahájena sériová komunikace pro výpis ladících informací a nastavení modré LED jako výstupu a uživatelského tlačítka jako vstupu s použitím vnitřního pull-up rezistoru. Pro případné obnovení uložených údajů k Wi-Fi je čten stav uživatelského tlačítka okamžitě po startu programu. V další fázi je zavolána funkce *setupwifi()*, která zajišťuje připojení k Wi-Fi a inicializaci MQTT komunikace. Na závěr se modul připojí k NTP serveru pro zajištění přesného času a vypíše stav zápisu zúčtovacího období v paměti ESP32.

```

1 void setup()
2 {
3   Serial.begin(115200);
4   // Configure GPIO for LED and user BUTTON
5   pinMode(LED_PIN, OUTPUT);
6   pinMode(BUTTON, INPUT_PULLUP);
7   // If the user button is pressed immediately after boot,
8   // reset Wi-Fi configuration
9   if (digitalRead(BUTTON) == LOW) {
10    wm.resetSettings();
11  }
12
13  // Connect to Wi-Fi and configure MQTT connection
14  setup_wifi();
15
16  // Init and get time from NTP
17  configTime(3600, 3600, ntp_server);
18  // Print billing period from flash memory
19  print_date("datefrom");
20  print_date("billdate");
21 }

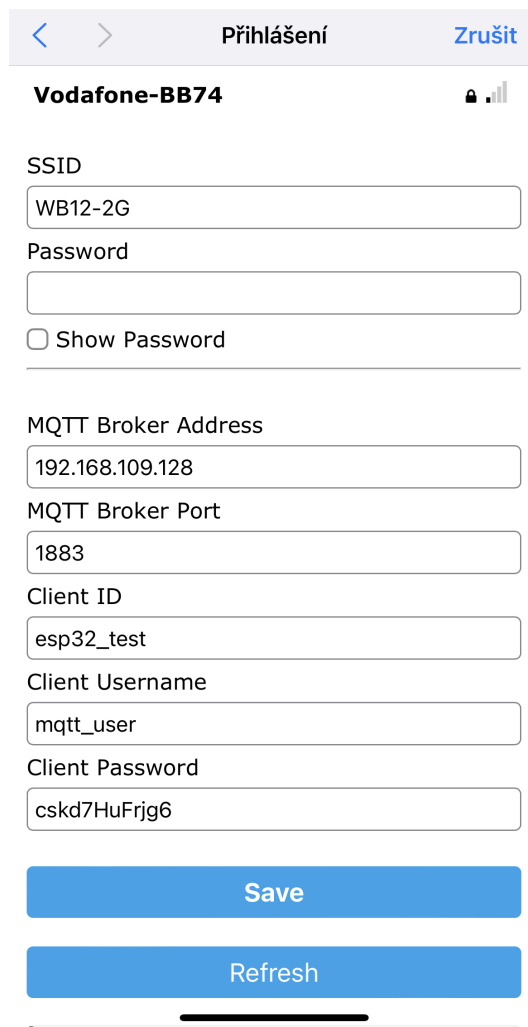
```

Obrázek 7.1. Úvodní konfigurace ve funkci setup().

■ 7.1.2 Připojení k Wi-Fi a MQTT serveru

Pro připojení k Wi-Fi byla použita knihovna *WiFiManager* od autora tzapu[40]. Ta umožňuje snadné připojení k Wi-Fi a zadání potřebných parametrů pro MQTT komunikaci bez nutnosti zasahovat do zdrojového kódu. ESP32 se po zapnutí přepne do režimu přístupového bodu (AP), ke kterému se uživatel pomocí mobilního telefonu nebo počítače připojí. Následně do konfiguračního portálu vyplní heslo konkrétního přístupového bodu. Tyto údaje jsou poté uloženy a při dalším spuštění se již ESP32 automaticky připojí bez nutnosti opětovného zadávání.

Funkce `setupwifi()` se pokusí připojit k poslední uložené Wi-Fi. Pokud taková Wi-Fi neexistuje nebo pokus o připojení selže, vytvoří se konfigurační portál, který navíc obsahuje formulář pro zadání MQTT parametrů. Pokud jsou již některé MQTT parametry uloženy, jsou ve formuláři předvyplněné. Po úspěšném připojení se nové údaje opět uloží do paměti. Ukázka portálu po zvolení konkrétní Wi-Fi ze seznamu je na obrázku 7.2.



The screenshot shows a mobile application interface for configuration. At the top, there is a header with a back arrow, a right arrow, the title "Přihlášení", and a "Zrušit" button. Below the header, the network name "Vodafone-BB74" is displayed with a lock icon and signal strength indicator. The form contains several input fields: "SSID" with the value "WB12-2G", "Password" (empty), "MQTT Broker Address" with "192.168.109.128", "MQTT Broker Port" with "1883", "Client ID" with "esp32_test", "Client Username" with "mqtt_user", and "Client Password" with "cskd7HuFrjg6". There is a "Show Password" checkbox which is unchecked. At the bottom, there are two blue buttons: "Save" and "Refresh".

Obrázek 7.2. Konfigurační portál pro připojení k Wi-Fi.

Pro vytvoření MQTT klienta a následnou komunikaci byla použita knihovna *Pub-SubClient* od autora Nick O’Leary[41]. Klient se pomocí uložených přístupových údajů a unikátního identifikátoru připojí k MQTT brokerovi. Dále začne odebírat topic, na který budou přicházet konfigurační příkazy od brokera. Při příchodu zprávy na odebíraný topic je zavolána callback funkce, ve které je příchozí zpráva zpracována.

■ 7.1.3 Hlavní smyčka

V hlavní smyčce je periodicky kontrolován stav připojení k Wi-Fi a k MQTT brokerovi. Dále je kontrolován stisk uživatelského tlačítka pro obnovení Wi-Fi. Každých 10 sekund je provedeno měření a každou minutu je vypočítán odhad spotřeby na konci zúčtovacího období.

```

1 void loop()
2 {
3   if (WiFi.status() != WL_CONNECTED) {
4     ESP.restart();
5   }
6   if (!client.connected()) {
7     reconnect();
8   }
9   if (digitalRead(BUTTON) == LOW) {
10    reset_wifi_button();
11  }
12  client.loop();
13
14  long now = millis();
15  // do measurement every 10 seconds
16  if (now - lastMsg > 10000) {
17    lastMsg = now;
18    cycles++;
19
20    String str_payload = measure_data();
21
22    if (cycles > 6) { // every 1 minute estimate energy
23      cycles = 0;
24      project_energy();
25    }
26
27    Serial.print("Sending data -> ");
28    Serial.println(str_payload);
29    client.publish(mqtt_publish_topic, str_payload.c_str());
30  }
31 }

```

Obrázek 7.3. Hlavní řídicí smyčka.

7.1.4 Měření veličin a odeslání dat

Pro měření s modulem PZEM-004T je využita knihovna od autora Jakub Mandula[42]. Naměřená data jsou následně převedena do formátu JSON. Pro každou veličinu je stanoveno zaokrouhlení na daný počet desetinných míst.

```

1 String measure_data(void)
2 {
3   // Read the data from the sensor
4   float voltage = pzem.voltage();
5   float current = pzem.current();
6   float power = pzem.power();
7   float energy = pzem.energy();
8   float frequency = pzem.frequency();
9   float pf = pzem.pf();
10
11   StaticJsonDocument<800> payload;
12   payload["voltage"] = String(voltage, 1);
13   payload["current"] = String(current, 3);
14   payload["power"] = String(power, 3);
15   payload["energy"] = String(energy, 2);
16   payload["projection"] = String(projection);
17   payload["frequency"] = String(frequency, 1);
18   payload["pfactor"] = String(pf, 2);
19
20   String str_payload;
21   serializeJson(payload, str_payload);
22   return str_payload;
23 }

```

Obrázek 7.4. Funkce pro měření veličin.

JSON je následně převeden na String objekt a pomocí funkce publish() jsou data odeslána na topic *homeenergy/sensorx*, kde x je číslo konkrétní monitorovací stanice.

7.1.5 Příjem konfiguračních příkazů

Pokud MQTT broker předá data na topic, který je monitorovací stanicí odebíráán, dojde k zavolání callback() funkce. Odebíraný topic monitorovací stanicí je `esp32/sensorx`, kde `x` je číslo konkrétní stanice. Seznam příkazů, které callback funkce zpracovává:

- `on, off` pro ovládání uživatelské LED, slouží k testování spojení
- `resetEnergy` vynuluje počítadlo spotřebované energie
- `resetWiFi` vymaže údaje o Wi-Fi a restartuje ESP32
- `restartESP` restartuje ESP32
- `datefrom-<dd/mm/yy>` nastaví počátek zúčtovacího období
- `billdate-<dd/mm/yy>` nastaví konec zúčtovacího období

7.1.6 Nastavení zúčtovacího období

Pro práci s daty program využívá knihovnu `time.h`. Datum, který stanice přijme pomocí MQTT příkazu, převede na unixový čas. Ten je definován jako počet sekund od 1. ledna 1970. Ten uloží do 32 bitového integeru a zapíše do paměti flash. Při zavolání funkce `resetEnergy` je pomocí NTP zjištěno aktuální datum a je uložen nový počátek zúčtovacího období.

7.1.7 Odhad spotřeby

Odhad spotřeby je počítán jednou za minutu. Funkce pro odhad spotřeby zjistí pomocí NTP protokolu aktuální čas a přečte z paměti počátek a konec zúčtovacího období. Následně pomocí jednoduchého vzorce odhadne stav spotřeby na konci zúčtovacího období. Funkce je ukázána na obrázku 7.5.

```

1 // Calculates estimated consumption
2 void project_energy(void)
3 {
4     int energy = pzem.energy() * 1000;
5     time_t date_from = read_date("datefrom");
6     time_t bill_date = read_date("billdate");
7     if (date_from != 0 && bill_date != 0) {
8         struct tm timeinfo;
9         if (!getLocalTime(&timeinfo)) {
10            Serial.println("ERROR: Failed to obtain time");
11        } else {
12            double now = mktime(&timeinfo);
13            double remaining = bill_date - now;
14            double elapsed = now - date_from;
15            Serial.print("remaining hours: ");
16            Serial.println(remaining / (3600));
17            Serial.print("elapsed hours: ");
18            Serial.println(elapsed / (3600));
19            projection = energy * (1.0 + (remaining / elapsed));
20            Serial.print("Projection in Wh: ");
21            Serial.println(projection);
22        }
23    }
24 }

```

Obrázek 7.5. Funkce pro odhad spotřeby.

7.2 Nastavení serveru chytré domácnosti

V následujících podkapitolách je popsána konfigurace použitého serveru chytré domácnosti - Home Assistant. Ten pomocí MQTT přijímá data z monitorovací stanice a následně je uchovává a vizualizuje.

7.2.1 Použitý HW a instalace

K provozu serveru chytré domácnosti je zvolen mini počítač RaspberryPi 4 ve variantě s pamětí 2 GB RAM. Na MicroSDXC kartu o velikosti 64 GB je nahrán image operačního systému a jsou následovány další kroky v návodu[43]. Po spuštění Raspberry s vloženou SD kartou je v lokální síti server dostupný na adrese *homeassistant.local:8123*.

7.2.2 Nastavení databáze InfluxDB

Databáze InfluxDB je nainstalována skrze nabídku rozšíření (Add-on Store). V rozhraní InfluxDB je vytvořena nová databáze a nový uživatel, který do ní má přístup. Při vytvoření databáze je možné zvolit, jestli mají být data po určitém čase mazána. Na závěr je potřeba do souboru *configuration.yaml* v Home Assistant přidat několik řádků, které jsou na obrázku 7.6. Je v nich obsažena lokální IP adresa serveru, jméno vytvořené databáze a přístupové údaje vytvořeného uživatele.

```

1 # InfluxDB
2 influxdb:
3   host: <IP address>
4   port: 8086
5   database: <name>
6   username: <username>
7   password: <password>
8   max_retries: 3
9   default_measurement: state

```

Obrázek 7.6. Konfigurace InfluxDB v souboru *configuration.yaml*.

7.2.3 Nastavení MQTT brokeru

Pro komunikaci mezi serverem a monitorovací stanicí je nutné nastavit MQTT broker. Ten je rovněž nainstalován skrze nabídku rozšíření a je dostupný pod jménem Mosquitto broker. Poté je v nastavení Home Assistant vytvořen nový uživatel (bez administrátorských pravomocí), který bude autorizován pro připojení k brokerovi. Nakonec je server restartován a při opětovném spuštění je uživatel vyzván ke konfiguraci MQTT brokeru, kde jsou vyplněny přístupové údaje nově vytvořeného uživatele. Tím je konfigurace MQTT brokeru hotová, ale v tomto případě by nedocházelo k uchování přijatých dat. Do souboru *configuration.yaml* je nutné přidat několik řádků, viz obrázek 7.7.

```

1 mqtt:
2   sensor:
3     - name: "energymonitor_power"
4       state_topic: "homeenergy/sensor1"
5       suggested_display_precision: 1
6       unit_of_measurement: "W"
7       value_template: "{{ value_json.power }}"
8
9     - name: "energymonitor_voltage"
10      state_topic: "homeenergy/sensor1"
11      suggested_display_precision: 1
12      unit_of_measurement: "V"
13      value_template: "{{ value_json.voltage }}"

```

Obrázek 7.7. Konfigurace MQTT v souboru *configuration.yaml*.

Je nastaven název entity, která nově vznikne a kterou je možné v samotném rozhraní Home Assistant použít k automatizaci, nebo její hodnotu zobrazit na hlavní nástěnce. Dále je vyplněn topic, na který monitorovací stanice odesílá data a označení, pod kterým je ve formátu JSON uložena konkrétní veličina. Na obrázku 7.7 je uveden příklad pro výkon a napětí. Pro další veličiny je to nastaveno analogicky. Zbývá upravit konfiguraci InfluxDB tak, aby ukládala nově vytvořené entity, viz obrázek 7.8.

```
1 # InfluxDB
2 influxdb:
3   host: <IP address>
4   port: 8086
5   database: <name>
6   username: <username>
7   password: <password>
8   max_retries: 3
9   default_measurement: state
10  include:
11    entities:
12      - sensor.energymonitor_power
13      - sensor.energymonitor_voltage
14      - etc...
```

Obrázek 7.8. Konfigurace ukládání entit do InfluxDB v souboru configuration.yaml.

■ 7.2.4 Nastavení nástroje Grafana

Posledním krokem zpracování dat je vizualizace. Grafana je opět nainstalována skrze nabídku rozšíření. Po instalaci je v nastavení Grafany zvolena sekce Data Sources a přidán nový zdroj v podobě InfluxDB. V konfiguračním okně je vyplněna IP adresa, port a přístupové údaje databáze zmíněné v podkapitole o nastavení InfluxDB. Nyní je možné vytvářet přehledy z naměřených dat.

Kapitola 8

Testování prototypu

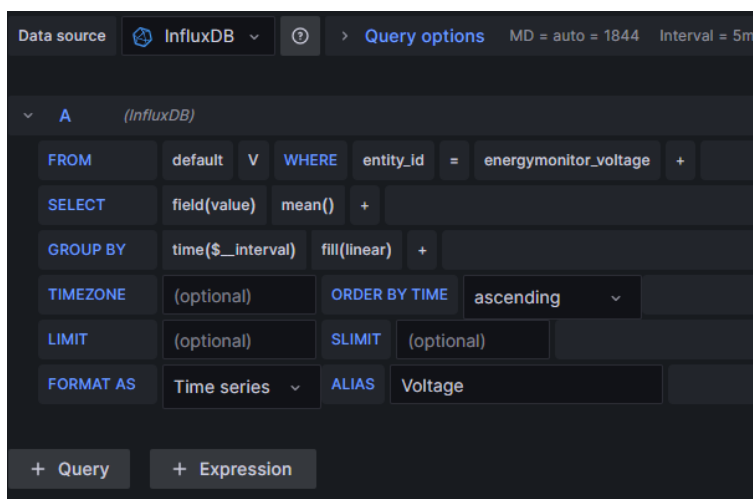
Tato kapitola popisuje testovací provoz prototypu. Jsou zde ukázány zvolené formy vizualizace konkrétních metrik a je vysvětleno, jaké byly použity dotazy na databázi. Dále je představena možnost vložení panelů z Grafany na hlavní přehled Home Assistant. Na konec je shrnut testovací provoz.

8.1 Spuštění monitorovací stanice

Po připojení monitorovací stanice k napájení se okamžitě vytvořil konfigurační portál. Do něj byla z mobilního telefonu vyplněna požadovaná data a stanice se připojila k Wi-Fi. Po připojení k Wi-Fi navázala spojení s MQTT brokerem a okamžitě začala odesílat naměřená data do databáze.

8.2 Vytvoření přehledu v nástroji Grafana

Na jedné nástěnce v Grafaně může být několik panelů a každý panel je tvořen jedním nebo více dotazy na databázi InfluxDB. Dotaz na databázi se vytváří pomocí jednoduchého UI rozhraní, přičemž Grafana sama nabízí, které entity je možné zvolit. Příklad dotazu na hodnotu napětí je na obrázku 8.1.

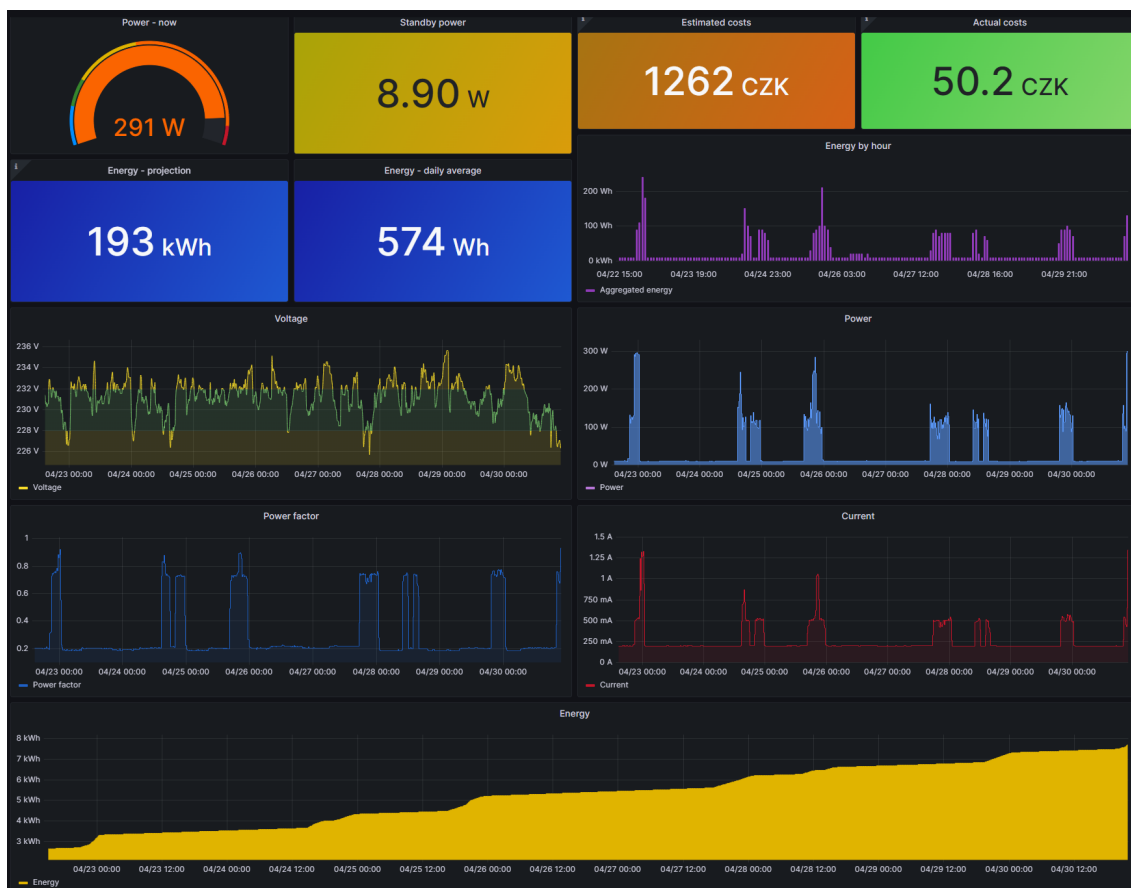


Obrázek 8.1. Vytvoření databázového dotazu v Grafaně.

Dotazy na databázi je možné různě kombinovat a s výsledky lze provádět další výpočty. Na obrázku 8.2 je vidět výsledný přehled, který byl pro testování prototypu vytvořen. V horní části je vytvořeno několik barevných dlaždic. Obsahují aktuální výkon a nejčastější výkon (výkon, který je v daném časovém intervalu nejčtenější). Dále je zobrazen odhad ceny, který se počítá z odhadu spotřeby v databázi a ceny za kWh, která se vkládá do proměnné v horní části nástěnky. Společně s odhadem ceny je zde

také kalkulace ceny za aktuálně spotřebovanou elektřinu. Poslední dvě dlaždice obsahují odhad spotřebované energie na konci zúčtovacího období a denní průměr spotřebované energie.

Pod barevnými dlaždicemi jsou dále zobrazeny grafy pro jednotlivé veličiny. Je zobrazen graf pro napětí, spotřebu agregovanou po hodinách, výkon, účinník, proud a kumulativní graf energie. Všechny tyto grafy jsou zobrazeny pro konkrétní časový interval, který je v rozhraní Grafany nastaven.

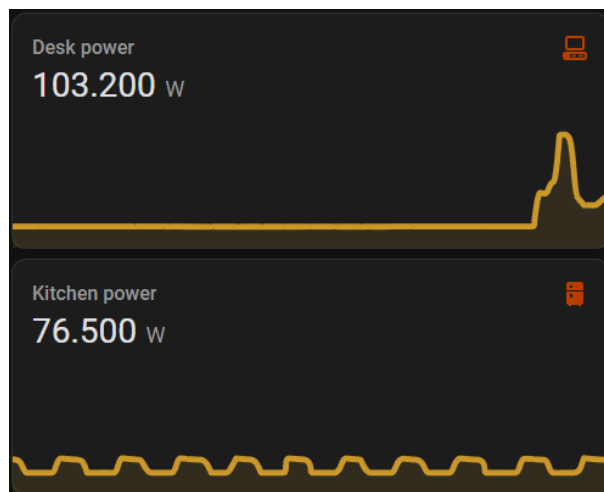


Obrázek 8.2. Ukázka výsledného přehledu v nástroji Grafana.

8.3 Export nástěnky na hlavní přehled

Celou nástěnku nebo konkrétní panel je možné exportovat na hlavní přehled Home Assistant jako iframe, což je v podstatě vložený kus HTML kódu. U konkrétního panelu je zvolena možnost sdílet (share) a následně možnost embed, kde je zkopírována src adresa. Poté je v editaci nástěnky Home Assistant zvolena možnost přidat kartu (add card) a v nabídce je vybrána možnost webpage. Do kolonky URL je vložena src adresa zkopírována dříve. Nyní jsou výstupy z Grafany dostupné i na hlavním přehledu.

Alternativou je vizualizace entit bez použití nástroje Grafana. K tomu stačí na nástěnku přidat novou kartu, například senzor, a vybrat danou entitu. Příklad zobrazení je na obrázku 8.3. Tento přístup na rozdíl od kombinace InfluxDB a Grafana neumožňuje na datech provádět složitější dotazy.



Obrázek 8.3. Ukázka vizualizace entity pomocí Home Assistant karty na nástěnce.

8.4 Testovací provoz

Během testování byla ověřena funkčnost konceptu monitorovací stanice. Zařízení je v téměř nepřetržitém provozu přes 60 dní. Vyzkoušeny byly všechny příkazy, které lze na zařízení odeslat. Zařízení spolehlivě komunikuje a úspěšně reaguje na neplánované incidenty. Bylo realizováno několik variant simulovaných incidentů, mezi které patřil výpadek napájení, výpadek konektivity a výpadek MQTT brokera. Zařízení se ze všech těchto stavů bylo schopno opět zprovoznit. Server pro zpracování dat rovněž fungoval bez jakýchkoli problémů.

Pro ověření snadné integrace byl sestaven druhý testovací prototyp, který se po vyplnění připojovacích parametrů z mobilního telefonu a konfiguraci na straně Home Assistant automaticky připojil a začal fungovat, jak bylo očekáváno.

Kapitola 9

Zhodnocení dosažených výsledků

Tato kapitola shrnuje dosažené výsledky a porovnává je s komerčně dostupnými produkty. Zároveň jsou vyčísleny náklady za sestavení prototypu monitorovací stanice.

9.1 Náklady na výrobu prototypu

V tabulce 9.1 jsou shrnuty náklady na jednotlivé položky. Celková cena vývojového prototypu je 384,40 Kč. Cenu by bylo možné ještě více snížit, pokud by místo vývojové sady s ESP32 byl použit samotný ESP32 modul, který by byl osazen na desce plošných spojů přímo. V celkové ceně nejsou zahrnuty náklady na poštovné ani na 3D tisk. V nákladech není zahrnuta ani lidská práce, která by celkovou cenu výrazně navýšila. Důvodem je to, že výroba prototypů je poměrně časově náročná.

| Položka | Prodejce | Cena [Kč] |
|-------------------------------|---------------|-----------|
| Vývojová deska NodeMCU-32S | Aliexpress | 130,00 |
| Měřicí modul PZEM-004T | Aliexpress | 120,00 |
| Spínaný zdroj HLK-PM01 | Aliexpress | 54,00 |
| Převodník logických úrovní | Laskakit | 16,00 |
| Dutinkové a kolíkové lišty | Laskakit | 15,00 |
| Varistor VCR20D431KAR | GM electronic | 11,00 |
| Pojistka radiální 250 mA | GM electronic | 10,00 |
| Spojovací materiál | Šrouby Soukup | 10,00 |
| Deska plošných spojů | JLCPCB | 8,80 |
| Svorkovnice 2 pól | GM electronic | 4,60 |
| Tlačítko, DuPont kabel, pájka | vlastní | 5,00 |
| | Celkem | 384,40 Kč |

Tabulka 9.1. Cena použitých komponent.

9.2 Porovnání s komerčně dostupnými produkty

V následující části je prototyp monitorovací stanice srovnán s komerčně dostupnými produkty. Jsou srovnány jednotlivé aspekty a shrnuty výhody a nevýhody testovacího prototypu.

9.2.1 Srovnání z hlediska měřených veličin

Stejně jako komerční produkty analyzované v úvodu je testovací prototyp schopen měřit aktuální výkon a celkovou spotřebu. V porovnání s chytrými zásuvkami je testovací prototyp schopen měřit více veličin než jen aktuální výkon a spotřebu. Na druhou

stranu není prototyp koncipován jako chytrá zásuvka, a nemůže tedy spínat napájení. S tímto faktem je spojené to, že testovací prototyp umožňuje měřit několikanásobně větší výkon, a to díky oddělené proudové sondě, která je schopna měřit proud až do velikosti 100 A.

V porovnání se zařízeními do rozvodné skříně je jeho hlavní výhodou kombinace výkonného MCU, otevřeného API a možnosti kalkulace spotřeby. Podobně jako u komerčních zařízení je testovací prototyp schopen měřit velké proudy díky proudové sondě. Pro účely snadného testování a maximální bezpečnosti nebylo pouzdro testovacího prototypu navrženo pro umístění na DIN lištu.

■ 9.2.2 Srovnání z hlediska dostupných funkcí

Testovací prototyp je vybudován na uživatelsky populární platformě ESP32, která disponuje výkonným procesorem a možností komunikace pomocí Wi-Fi a Bluetooth. Díky tomu je možné přímo na zařízení provádět náročnější operace a postupem času přidávat další funkce. Zařízení je oproti komerčně dostupným produktům od počátku navrženo jako otevřené a je tedy snadné ho integrovat do systémů chytré domácnosti. K tomu byl použit populární protokol MQTT, který slouží ke komunikaci mezi zařízením a MQTT brokerem. Díky použití protokolu MQTT je možné zařízení ovládat pomocí mobilního telefonu. Všechny tyto vlastnosti splňují řádově dražší komerční zařízení, jako je například analyzované zařízení Shelly Pro 3EM.

■ 9.3 Celkové zhodnocení

Návrh prototypu monitorovací stanice vycházel ze současné situace na trhu chytrých senzorů spotřeby. Návrh prototypu vycházel z několika předpokladů. Prvním z nich byla otevřenost zařízení, protože na trhu komerčně dostupných produktů je běžné mít pro jedno zařízení jednu specifickou aplikaci a je složité mezi sebou různá zařízení propojovat. Dalším předpokladem bylo vybudovat prototyp zařízení na populární platformě, která je zároveň cenově dostupná. K tomu byla vybrána platforma ESP32, která disponuje Wi-Fi konektivitou a výkonným procesorem. Rovněž bylo předpokladem, aby bylo zařízení schopné měřit spotřebu konkrétního okruhu v rozvodné skříně. K tomu byl použit monitorovací modul s externí proudovou sondou, která umožňuje měřit proud až do velikosti 100 A. Na závěr byl představen způsob, jak naměřená data uchovat a vizualizovat pomocí nejpopulárnějšího otevřeného systému chytré domácnosti - Home Assistant. K vizualizaci byl použit nástroj Grafana, který jako zdroj používal databázi časových řad InfluxDB.

Náklady na testovací prototyp činí 384,40 Kč, což je srovnatelná částka, za kterou je možné pořídit levnější chytré zásuvky. Oproti chytrým zásuvkám ale testovací prototyp umožňuje kalkulaci nákladů, větší počet měřených veličin a otevřené API pro integraci do jiných systémů. Srovnatelné vlastnosti má pouze zařízení Shelly Pro 3EM, jehož cena přesahuje 3000 Kč.

Další výhodou je volba výkonného mikrokontroléru. To umožňuje zařízení do budoucna vylepšovat. Mezi tato vylepšení by mohla patřit schopnost měřit více fází pomocí dalších měřicích modulů, komplexnější analýza měřených spotřebičů a přesnější odhad celkové spotřeby, lepší systém pro kalkulaci nákladů, který počítá s pohyblivými cenami za elektřinu, nebo využití detekce anomálií pro včasné odhalení poruch a jiných nežádoucích incidentů. Dalším vylepšením by mohlo být přidání kompatibility s komerčními systémy chytré domácnosti, jako je Google Home nebo Samsung SmartThings.

Kapitola 10

Závěr

Cílem bakalářské práce bylo provést analýzu trhu se zařízeními na monitoring spotřeby elektrické energie a získané poznatky použít k návrhu vlastního konceptu zařízení měřícího spotřebu. Bylo požadováno vytvořit prototyp monitorovací stanice, který umožňuje komunikaci pomocí protokolu MQTT a následně nakonfigurovat obslužný software tak, aby bylo možné na dálku monitorovat měřené parametry a provést základní kalkulaci nákladů za spotřebovanou elektrickou energii a statistickou analýzu. Nakonec měl být prototyp otestován a porovnán s komerčními zařízeními.

Nejprve byly popsány základní veličiny týkající se měření spotřeby elektrické energie v jednofázové síti s harmonickým průběhem. V analýze trhu bylo zkoumáno celkem 8 zařízení, z nichž polovina byla určena jako zařízení do zásuvky a polovina jako zařízení do rozvodné skříně. Byly porovnány jednotlivé parametry a ukázalo se, že společným jmenovatelem komerčních zařízení je uzavřenost jejich systémů a velké množství navzájem nekompatibilních aplikací.

V návrhu prototypu byl vybrán cenově dostupný a uživatelsky populární mikrokontrolér ESP32, který disponuje Wi-Fi konektivitou a dostatečným výkonem. Dále byl vybrán vhodný měřicí modul a další potřebné prvky pro funkční testovací prototyp. Byl představen vhodný proces zpracování a vizualizace dat. Zařízení periodicky monitoruje spotřebu a pomocí protokolu MQTT data odesílá do otevřeného systému chytré domácnosti Home Assistant. Tam jsou následně uchována v databázi časových řad InfluxDB a vizualizována pomocí nástroje Grafana. Monitorovací stanici je možné pomocí MQTT také ovládat a nastavovat zúčtovací periodu, která slouží ke kalkulaci odhadovaných nákladů. Pro testovací prototyp byla navržena deska plošných spojů a 3D pouzdro, které bylo vyrobeno pomocí 3D tiskárny. Náklady na výrobu prototypu byly necelých 400 Kč.

Prototyp byl testován v souvislém provozu po dobu přesahující 60 dní. Jednotlivé funkce byly používány a byla ověřena jejich správná funkčnost. V porovnání s komerčními zařízeními má prototyp několik výhod mezi které patří otevřenost díky použití protokolu MQTT, kalkulace nákladů za spotřebovanou elektřinu, nízká cena prototypu, výkonný čip ESP32 a možnost měřit velké proudy díky použití proudové sondy.

Použití výkonného mikrokontroléru ESP32 umožňuje budoucí vylepšování monitorovací stanice o další funkce a složitější výpočty. Mezi budoucí rozšíření by mohla patřit schopnost měřit více fází, komplexnější analýza spotřebičů nebo všestrannější kalkulace nákladů.

Literatura

- [1] *Rozměry spínacích a řídicích zařízení nízkého napětí - Normalizované montážní lišty pro mechanické upevnění spínacích a řídicích zařízení a příslušenství*. ČSN EN 60715 ED.2. vyd.. Praha: Česká agentura pro standardizaci, 2018.
- [2] IEEE. IEEE Standard Definitions for the Measurement of Electric Power Quantities Under Sinusoidal, Nonsinusoidal, Balanced, or Unbalanced Conditions. *IEEE Standards*. 2010, STD 1459-2000 1-52. DOI 10.1109/IEEESTD.2010.5439063.
- [3] Eli Osherovich. *Complex power triangle*. 2011.
<https://upload.wikimedia.org/wikipedia/commons/d/d8/Cmplxpower.svg>.
Přistoupeno: 2023-04-07.
- [4] Jon Peli Oleaga. *Image showing the instantaneous electric power in AC systems and its decomposition into active and reactive power; when the current lags the voltage 50 degrees*. 2022.
<https://upload.wikimedia.org/wikipedia/commons/4/4d/Active-and-reactive-power-064pf-en.svg>. Přistoupeno: 2023-04-07.
- [5] *Immax NEO LITE ZÁSUVKA Smart v2*.
<https://www.immax.cz/immax-neo-lite-zasuvka-smart-vnitri-zasuvka-v2-s-kolikem-typ-e-wifi-p15497/>. Přistoupeno: 2023-04-07.
- [6] *Tesla Smart Plug*.
<https://www.teslasmart.com/cz/tesla-smart-plug>. Přistoupeno: 2023-04-07.
- [7] *TP-LINK Tapo P115*.
<https://www.tp-link.com/cz/home-networking/smart-plug/tapo-p115/>.
Přistoupeno: 2023-04-07.
- [8] *Eve Energy*.
<https://www.evehome.com/en/eve-energy>. Přistoupeno: 2023-04-07.
- [9] *Obrázek zásuvky Oblo*.
<https://www.obloliving.com/wp-content/uploads/2020/12/smart-plug-tech-spec.png>. Přistoupeno: 2023-04-08.
- [10] Zach a Carsten BORMANN SHELBY. *6LoWPAN: the wireless embedded internet*. Edition 1 vyd.. Chichester: John Wiley & Sons, 2009. ISBN 978-0-470-74799-5.
- [11] *Shelly Plus 1PM*.
<https://www.shelly.cloud/en-cz/products/product-overview/shelly-plus-1-pm-2-pack/shelly-plus-1-pm>. Přistoupeno: 2023-04-08.
- [12] *AEOTEC Home Energy Meter Gen5*.
<https://aeotec.com/products/aeotec-home-energy-meter/>. Přistoupeno: 2023-04-08.
- [13] *Zamel MEW-01 Lite*.
<https://supla.zamel.com/en/product/mew-01-lite/>. Přistoupeno: 2023-04-08.

- [14] *Shelly Pro 3EM*.
<https://www.shelly.cloud/en-cz/products/product-overview/pro-3-em>.
Přistoupeno: 2023-04-08.
- [15] *Obrázek Shelly Pro 3EM*.
<https://www.shelly.cloud/en-cz/products/product-overview/pro-3-em>.
Přistoupeno: 2023-04-08.
- [16] *Přesnost měření Aeotec Gen5*.
<https://aeotec.freshdesk.com/support/solutions/articles/6000168072-home-energy-meter-gen5-technical-specifications->. Přistoupeno: 2023-04-10.
- [17] *Přesnost měření Zamel MEW-01 Lite*.
<https://supla.zamel.com/pdfdocs/mew-01-manual.pdf>. Přistoupeno: 2023-04-10.
- [18] *Přesnost měření Shelly Pro 3EM*.
<https://www.shelly.cloud/en-cz/products/product-overview/pro-3-em>.
Přistoupeno: 2023-04-10.
- [19] *Z-Wave frequencies by country*.
<https://atrim.co/zwave/z-wave-frequencies-countries.html>. Přistoupeno: 2023-04-10.
- [20] J. P. Ross a Alan Meier. *Whole-House Measurements of Standby Power Consumption*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2001. ISBN 978-3-642-56531-1.
- [21] *ESP32 WROOM 32 datasheet*.
https://www.espressif.com/sites/default/files/documentation/esp32-wroom-32_datasheet_en.pdf. Přistoupeno: 2023-04-16.
- [22] *Raspberry Pi 4 product brief*.
<https://datasheets.raspberrypi.com/rpi4/raspberry-pi-4-product-brief.pdf>. Přistoupeno: 2023-04-16.
- [23] *PZEM-004T V3.0 User Manual*.
<https://innovators-guru.com/wp-content/uploads/2019/06/PZEM-004T-V3.0-Datasheet-User-Manual.pdf>. Přistoupeno: 2023-04-16.
- [24] *Definition of Internet of Things*.
<https://www.gartner.com/en/information-technology/glossary/internet-of-things>. Přistoupeno: 2023-01-14.
- [25] *IoT vs TCP/IP protocol stack*.
https://www.researchgate.net/figure/Typical-IoT-vs-TCP-IP-Protocol-Stack_fig3_336179532. Přistoupeno: 2023-04-20.
- [26] *Apache Kafka*.
<https://kafka.apache.org/>. Přistoupeno: 2023-04-30.
- [27] Jasenka Dizdarevic, Francisco Carpio, Admela Jukan a Xavi Masip-Bruin. A Survey of Communication Protocols for Internet of Things and Related Challenges of Fog and Cloud Computing Integration. 2019, 51 (6), DOI 10.1145/3292674.
- [28] *MQTT Version 5.0 — docs.oasis-open.org*. <https://docs.oasis-open.org/mqtt/mqtt/v5.0/cs01/mqtt-v5.0-cs01.html>. Přistoupeno: 2023-05-07.

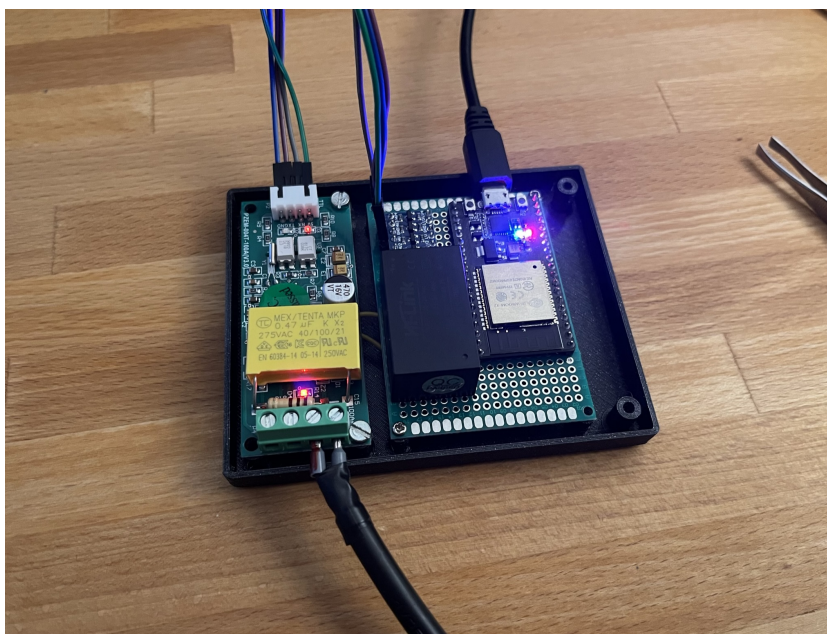
- [29] *MQTT publish-subscribe model*.
<https://docs.devicewise.com/Content/Resources/Images/MQTT.png>. Přistoupeno: 2023-05-07.
- [30] *Datasheet Nodemcu-32S*.
https://docs.ai-thinker.com/_media/esp32/docs/nodemcu-32s_product_specification.pdf. Přistoupeno: 2023-05-08.
- [31] *Logic bi-directional level shifter schematic*.
https://cdn.sparkfun.com/datasheets/BreakoutBoards/Logic_Level_Bidirectional.pdf. Přistoupeno: 2023-05-09.
- [32] *Označení CE*.
https://europa.eu/youreurope/business/product-requirements/labels-markings/ce-marking/index_cs.htm. Přistoupeno: 2023-05-09.
- [33] Ksawery Szykiedans, Wojciech Credo a Dymitr Osiński. Selected Mechanical Properties of PETG 3-D Prints. *Procedia Engineering*. 2017, 177 455-461. DOI 10.1016/j.proeng.2017.02.245.
- [34] *Autodesk Fusion360*.
<https://www.autodesk.com/products/fusion-360>. Přistoupeno: 2023-05-12.
- [35] *PrusaSlicer*.
https://www.prusa3d.com/page/prusaslicer_424/. Přistoupeno: 2023-05-12.
- [36] *Original Prusa Mini+*.
<https://www.prusa3d.com/cs/produkt/castecne-sestavena-3d-tiskarna-original-prusa-mini-4/>. Přistoupeno: 2023-05-12.
- [37] *The Best 3D Printer*.
<https://www.nytimes.com/wirecutter/reviews/best-home-3d-printer/>. Přistoupeno: 2023-05-12.
- [38] *Platformio*.
<https://platformio.org/>. Přistoupeno: 2023-05-15.
- [39] *Home Assistant*.
<https://www.home-assistant.io/>. Přistoupeno: 2023-05-15.
- [40] tzapu. *WiFiManager Library*.
<https://github.com/tzapu/WiFiManager/tree/master>. Přistoupeno: 2023-05-15.
- [41] Nick O'Leary. *PubSubClient Library*.
<https://pubsubclient.knolleary.net/api>. Přistoupeno: 2023-05-15.
- [42] Jakub Mandula. *Arduino library for Peacefair PZEM-004T*.
<https://github.com/mandulaj/PZEM-004T-v30>. Přistoupeno: 2023-05-15.
- [43] *Home Assistant installation Raspberry Pi*.
<https://www.home-assistant.io/installation/raspberrypi>. Přistoupeno: 2023-05-15.

Příloha A

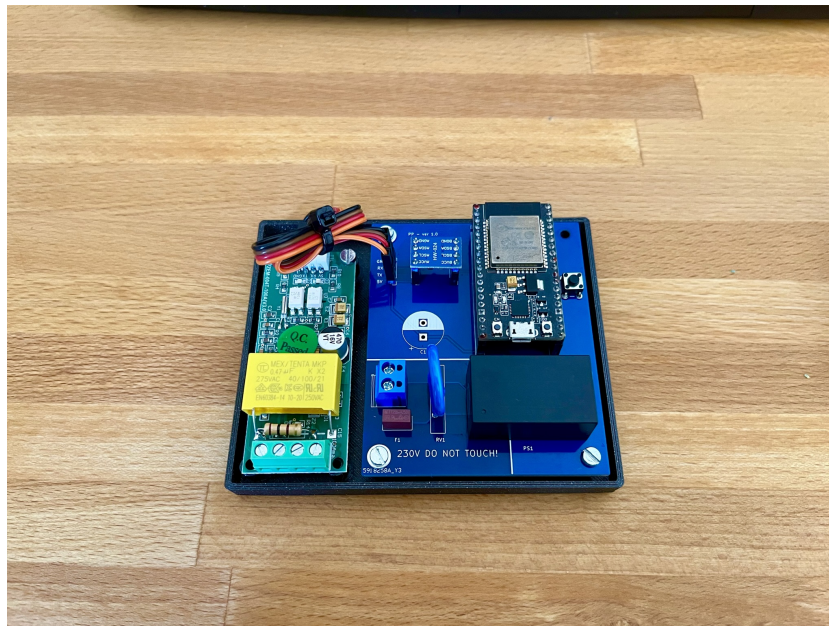
Fotografie prototypu



Obrázek A.1. Montáž prvního prototypu.



Obrázek A.2. Funkční prototyp.



Obrázek A.3. Finální verze na vlastní desce plošných spojů.

Příloha B

Seznam zkratk

| | | |
|------|---|---|
| API | ■ | Application Programming Interface |
| BLE | ■ | Bluetooth Low Energy |
| CNC | ■ | Computerized Numerical Control |
| FDM | ■ | Fused Deposition Modeling |
| IDE | ■ | Integrated Development Environment |
| IoT | ■ | Internet of things |
| LED | ■ | light-emitting diode |
| MCU | ■ | Microcontroller unit |
| MQTT | ■ | Message Queuing Telemetry Transport |
| M2M | ■ | Machine to Machine |
| QoS | ■ | Quality of Service |
| REST | ■ | Representational State Transfer |
| SoC | ■ | System on Chip |
| UART | ■ | Universal asynchronous receiver-transmitter |
| USB | ■ | Universal serial bus |