



ZADÁNÍ BAKALÁ SKÉ PRÁCE

Název:	Návrh a realizace wattrouteru
Student:	Ji í Cvr ek
Vedoucí:	Ing. Stanislav Je ábek
Studijní program:	Informatika
Studijní obor:	Po íta ové inženýrství
Katedra:	Katedra íslicového návrhu
Platnost zadání:	Do konce letního semestru 2017/18

Pokyny pro vypracování

Prove te analýzu existujících ešení pro ízení elektrických spot ebi v závislosti na výrob fotovoltaické (FVE) i v trné elektrárny (VTE), pro které se používá název wattrouter. Zhodno e pozitiva a negativa jednotlivých ešení a navrhn te specifikaci ideálního wattrouteru.

Prove te analýzu dostupného hardwaru, na kterém lze založit návrh wattrouteru s vlastnostmi co nejbližšími navržené specifikaci. Zhodno te náro nost naprogramování, možnosti p ípojení periferií a cenu hardwaru (p edpokládáme, že p jde o použití Arduina).

Navrhn te wattrouter, který bude umož ůvat komunikaci a konfiguraci pomocí p íkaz p edávaných po sériové lince nebo po síti. Realizujte i jednoduché ovládání pomocí tla ítek a displeje, který bude zobrazovat aktuální stav výroby a spot eby elektrické energie.

Seznam odborné literatury

Dodá vedoucí práce.

doc. Ing. Hana Kubátová, CSc.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
d kan

V Praze dne 20. ledna 2017

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
KATEDRA ČÍSLICOVÉHO NÁVRHU



Bakalářská práce

Návrh a realizace wattrouteru

Jiří Cvrček

Vedoucí práce: Ing. Stanislav Jeřábek

15. května 2017

Poděkování

Chtěl bych poděkovat panu Ing. Stanislavu Jeřábkovi za vedení této práce, cenné rady a doporučení, které mi pomohly tuto práci zkompletovat. Dále bych chtěl poděkovat panu Dr-Ing. Martinu Novotnému za odborné rady, které mi pomohly vyřešit některé problémy. Nemohu vynechat své rodiče, kterým chci poděkovat za obrovskou podporu při psaní práce i během celého studia.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracoval(a) samostatně a že jsem uvedl(a) veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů. V souladu s ust. § 46 odst. 6 tohoto zákona tímto uděluji nevýhradní oprávnění (licenci) k užití této mojí práce, a to včetně všech počítačových programů, jež jsou její součástí či přílohou, a veškeré jejich dokumentace (dále souhrnně jen „Dílo“), a to všem osobám, které si přejí Dílo užít. Tyto osoby jsou oprávněny Dílo užít jakýmkoli způsobem, který nesnižuje hodnotu Díla, a za jakýmkoli účelem (včetně užití k výdělečným účelům). Toto oprávnění je časově, teritoriálně i množstevně neomezené. Každá osoba, která využije výše uvedenou licenci, se však zavazuje udělit ke každému dílu, které vznikne (byť jen zčásti) na základě Díla, úpravou Díla, spojením Díla s jiným dílem, zařazením Díla do díla souborného či zpracováním Díla (včetně překladu), licenci alespoň ve výše uvedeném rozsahu a zároveň zpřístupnit zdrojový kód takového díla alespoň srovnatelným způsobem a ve srovnatelném rozsahu, jako je zpřístupněn zdrojový kód Díla.

V Praze dne 15. května 2017

.....

České vysoké učení technické v Praze

Fakulta informačních technologií

© 2017 Jiří Cvrček. Všechna práva vyhrazena.

Tato práce vznikla jako školní dílo na Českém vysokém učení technickém v Praze, Fakultě informačních technologií. Práce je chráněna právními předpisy a mezinárodními úmluvami o právu autorském a právech souvisejících s právem autorským. K jejímu užití, s výjimkou bezúplatných zákonných licencí, je nezbytný souhlas autora.

Odkaz na tuto práci

Cvrček, Jiří. *Návrh a realizace wattrouteru*. Bakalářská práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2017.

Abstrakt

Tato bakalářská práce se zabývá řešením problému řízeného využívání energie vyrobené fotovoltaickou elektrárnou místo její distribuce do rozvodné sítě. Cílem práce je navrhnout zařízení, které bude řídit spotřebu vyrobené energie v připojených spotřebičích na základě zadaných pravidel. v rámci práce se podařilo navrhnout zařízení, které sleduje aktuální výrobu fotovoltaické elektrárny, spotřebu domácnosti a porovnává je se zadanými pravidly. Spotřebiče se zapínají za předpokladu, že výroba převyšuje spotřebu a jsou splněna nastavená pravidla pro sepnutí daného spotřebiče. v příloze práce lze nalézt schéma a popis zapojení zařízení.

Klíčová slova zařízení, vestavný systém, automatizace, optimalizace, energetika, fotovoltaika

Abstract

This bachelor thesis is about solving the problem of using the produced energy by a photovoltaic power plant instead of distributing to electric power transmission. The aim of the thesis is to design a device that will control the consumption of energy produced using connected devices based on specified rules. During my work, I designed that monitors the current production of a

photovoltaic power plant, household consumption and compares with the set rules. The appliance switches on if the production exceeds the consumption and the set rules are met. The attachment contains a block diagram and a description of the device connection.

Keywords device, embedded system, automatization, optimalization, electrical energy, photovoltaics

Obsah

Úvod	1
1 Cíl práce	3
2 Analýza a návrh	5
2.1 Dostupná zařízení	5
2.2 Souhrn výhod a nevýhod dostupných zařízení	5
2.3 Návrh zařízení na základě analýzy dostupných řešení	7
2.4 Dostupné platformy mikropočítačů	7
2.5 Výběr konkrétní platformy	8
3 Realizace	9
3.1 Měření odebírané energie z rozvodné sítě	9
3.2 Nadvýroba a její využití	11
3.3 Nízký tarif	12
3.4 Spínání spotřebičů	13
3.5 Kontrola zapnutého spotřebiče	14
3.6 Ovládací prvky a periferie	14
3.7 Konfigurační a dotazovací příkazy	19
4 Testování	27
4.1 Měření napětí a výpočet efektivní hodnoty	27
4.2 Měření proudu a výpočet efektivní hodnoty	27
4.3 Detekce směru toku proudu	28
4.4 Připojení a testování periferií	31
4.5 Testování příkazů wattrouteru	32
5 Budoucí práce	35
5.1 Aplikace pro mobilní zařízení a PC	35
5.2 Rozšíření efektivního řízení spotřeby	35

Závěr	37
Literatura	39
A Seznam použitých zkratk	41
B Obsah přiloženého CD	43

Seznam obrázků

2.1	Wattrouter M SSR od firmy SolarControls s.r.o. [1]	6
2.2	Wattrouter M SSR od firmy SolarControls s.r.o. [2]	8
3.1	Zapojení měřícího modulu napětí s oddělovacím transformátorem. Vytvořeno v programu Eagle. [3]	12
3.2	Zapojení spínacího relé. Vytvořeno v programu Eagle. [3]	14
3.3	Ukázka zobrazení denní spotřeby.	15
3.4	Ukázka konfiguračního souboru.	17
3.5	Zapojení modulu s tlačítky. Vytvořeno v programu Eagle. [3]	18
3.6	Připojení modulu čtečky SD karet pomocí odporových děličů. Vy- tvořeno v programu Eagle. [3]	18
3.7	Blokové schéma připojení periférií. Vytvořeno v programu Fritzing. [4]	25
4.1	Ukázka nepřenosti měření bez zátěže. (žlutá: napětí v síti, zelená: protékající proud)	29
4.2	Ukázka zámků napájecího napětí wattrouteru se způsobenou ne- přeností měření. (žlutá: napětí v síti, zelená: protékající proud, modrá: napájecí napětí)	29
4.3	Ukázka průběhu proudu a napětí při spotřebě. (žlutá: napětí v síti, zelená: protékající proud)	30
4.4	Ukázka průběhu proudu a napětí při výrobě. (žlutá: napětí v síti, zelená: protékající proud)	30

Úvod

Tématem této bakalářské práce je řízení elektrických spotřebičů v závislosti na vlastní výrobě elektrické energie. v práci se budu zabývat optimalizací spotřeby vyrobené elektrické energie. Tu lze prodávat do sítě, ale ne vždy se to musí cenově vyplatit a může být výhodnější ji spotřebovat. Spotřebu lze realizovat libovolným elektrickým spotřebičem.

Zařízení pro optimalizaci spotřeby přebytků elektrické energie z nestálých zdrojů by ocenil každý majitel budovy s vlastní fotovoltaickou elektrárnou.

Výsledek práce bude určen všem, kdo mají k dispozici libovolný nestálý zdroj elektrické energie a chtějí ho automatizovaně plně využívat. Navrhované zařízení bude schopné zapínat a vypínat připojené spotřebiče podle vlastního nastavení.

Podobná zařízení pro řízení spotřeby již existují, ale obvykle poskytují pouze omezené možnosti v nastavení spínání spotřebičů. Chybí například fyzické ovládací prvky pro okamžité zapnutí či vypnutí spotřebiče bez nutnosti změny nastavení prostřednictvím osobního počítače (PC).

Cíl práce

Cílem rešerše je získat přehled o stávajících zařízeních pro řízení spotřeby elektrických spotřebičů dostupných na trhu a seznámit se základními principy funkčností a zapojením těchto zařízení.

Dalším cílem je provést analýzu nedostatků současných zařízení a veřejně dostupných požadavků uživatelů na vylepšení. Současně je potřeba nastudovat, na jakém základu lze toto zařízení realizovat, porovnat možnosti a některou z nich vybrat.

Cílem praktické části práce je návrh a realizace zařízení pro řízení elektrických spotřebičů. Zařízení by mělo spínat spotřebiče v závislosti na požadavcích uživatele a výrobě fotovoltaické elektrárny (FVE). Mezi požadavky uživatele může být například maximální doba provozu spotřebiče a sepnutí spotřebiče na nízký či vysoký tarif za předpokladu nenaplnění doby provozu. Požadavky bude možné nastavit prostřednictvím osobního počítače.

Analýza a návrh

Zařízení, které řídí rozložení zátěže elektrických spotřebičů v domě a jejich řízení v závislosti na dodávce energie z nestálých zdrojů se říká wattrouter.

Wattrouter je zařízení, které sleduje okamžitý odběr elektrické energie v domě. Zároveň sleduje výrobu elektrické energie FVE a ve chvíli, kdy výroba přesáhne spotřebu zapíná předvolené spotřebiče, které přebytečnou energií využijí, místo jejího prodeje do rozvodné sítě. [5]

2.1 Dostupná zařízení

Provedl jsem analýzu dostupných komerčních řešení regulace spotřeby energie dodané FVE. Zjistil jsem, že existuje pouze jeden výrobce wattrouteru na českém trhu. Firma SolarControls s.r.o. jako jediná dodává na českém trhu wattroutery [1]. A to hned v několika variantách. K dispozici jsou:

- obyčejné wattroutery, které pouze sledují výrobu a spotřebu a spínají připojené spotřebiče, nastavení je možné provádět z PC po propojení s wattrouterem pomocí USB kabelu,
- dále jsou k dispozici wattroutery, které podporují připojení více spotřebičů,
- v dražší cenové kategorii jsou zařízení, která lze připojit do počítačové sítě a konfiguraci provádět z PC. U těchto zařízení lze sledovat statistiky výroby a odběru.

2.2 Souhrn výhod a nevýhod dostupných zařízení

Všechny wattroutery od firmy SolarControls s.r.o. mají možnost nastavit sledování spotřeby elektrické energie na jednotlivých fázích zvlášť či společně.

2. ANALÝZA A NÁVRH



Obrázek 2.1: Wattrouter M SSR od firmy SolarControls s.r.o. [1]

U všech také najdeme možnost nastavení denního limitu odběru proudu, možnost běhu na nízký tarif do naplnění limitu či volbu priorit pro nastavení pořadí spínání spotřebičů. K některým modelům wattrouterů lze dokoupit moduly pro připojení do počítačové sítě.

Dražší wattroutery disponují moduly reálného času. Ty potom lépe poznají, kdy je třeba nulovat denní limity. Dále mají možnost záznamu naměřených dat. Bohužel žádný z wattrouterů nedisponuje displejem a tlačítky pro rychlou konfiguraci, například okamžité zapnutí či vypnutí elektrického spotřebiče. Zároveň by uživatel ocenil, kdyby se rychle mohl dozvědět, kolik daný spotřebič již spotřeboval energie a kolik mu zbývá do naplnění limitu. Wattroutery také nepoznají, zda je daný spotřebič připojený. Například bojler může být vypnutý termostatem, a tak spotřebič zapnou a začnou mu počítat dodaný proud, který ale do zařízení dodán není, neboť je zařízení vypnuté termostatem. Takové odpojení spotřebiče způsobí, že dle wattrouteru spotřebuje více, než je jeho reálná spotřeba, a pokud je ho potřeba zapnout někdy později, tak již k aktivaci nedojde z důvodu naplnění denního limitu.

2.3 Návrh zařízení na základě analýzy dostupných řešení

Návrh zařízení částečně plyne ze souhrnu výhod a nevýhod zjištěných u zkoumaných zařízení. V dostupných wattroutech jsem nenalezl funkci, která by byla přímo nežádoucí. Všechny funkce byli žádoucí, nicméně některé chybí. U přidaných funkcí jsem vycházel ze své vlastní zkušenosti s používáním wattrouteru od firmy SolarControls s.r.o. Po zvážení všech pro a proti jsem se rozhodl pro implementaci těchto funkcí:

- denní limit spotřebiče,
- běh zařízení při aktivním nízkém tarifu za předpokladu, že FVE nedodá dostatečné množství energie přes den,
- priority spínání spotřebičů,
- detekce zapnutého spotřebiče,
- bezdrátové připojení do počítačové sítě,
- ukládání naměřených dat na SD kartu,
- displej s několika tlačítky pro zjištění základních informací a provedení jednoduché změny nastavení spínání spotřebičů.

Dále jsem se rozhodl pro upravení funkce počítání spotřebovaného proudu spotřebičem. Většina wattrouterů vyhodnocuje odebraný proud spotřebičem pouze na základě doby sepnutého stykače spotřebiče. V důsledku toho dochází ke špatnému počítání odebrané energie a dřívejšímu dosažení denního limitu. Tento nedostatek je vyřešen ve funkci detekce zapnutého spotřebiče.

Další návrhy rozšiřujících funkcí jsou popsány v kapitole 5.

2.4 Dostupné platformy mikropočítačů

Pro realizaci wattrouteru je možné využít hned několik platforem mikropočítačů. Mezi hlavní patří dostupné Arduino a Raspberry Pi. Zároveň s Arduinem bych rád zmínil i možnost využití samotného čipu ATmega, který využívá Arduino. Tuto variantu ale nebudu posuzovat. Existují i další platformy s různými mikrokontrolery, například od firmy Microchip.

Hlavní rozdíl mezi Raspberry Pi a Arduinem je ten, že Raspberry Pi je v podstatě kompletní počítač velikosti kreditní karty. Tomu také odpovídá jeho výkon v porovnání s klasickým počítačem. Arduino je 8-bitový mikrokontroler a jeho výkon je proti Raspberry Pi podstatně nižší, ale pro potřeby řízení wattrouteru stále dostatečný. Také je velmi podstatný rozdíl v ceně obou platforem, Raspberry Pi je zhruba třikrát dražší než Arduino.

2.5 Výběr konkrétní platformy

Po krátké úvaze jsem se rozhodl pro použití Arduina. Důvodů jsem měl hned několik. Dva hlavní byly moje dřívější zkušenosti s touto platformou a cena Arduina.

V rámci studia jsem absolvoval předmět zaměřený na programování Arduina s mikrokontrolerem ATmega v jazyce C. Na Arduinu jsem také zprovoznil několik projektů, takže jsem měl nějaké zkušenosti.

Dalším důvodem pro Arduino byla i jeho cena. Klon desky Arduino je jedna z nejlevnějších vývojových možností, jeho cena se pohybuje kolem dvěřtř korun.



Obrázek 2.2: Wattrouter M SSR od firmy SolarControls s.r.o. [2]

Realizace

Pro realizaci wattrouteru jsem využil desku Arduino Mega 2560 R3. Ta je osazena mikrokontrolerem ATmega 2560 s 256kB paměti pro program a 8kB dynamické paměti. Mikrokontroler pracuje na 16Mhz. Deska disponuje šestnácti piny pro analogově digitální převodníky, digitálními piny s možností pulzně šířkové modulace a digitálními vstupně-výstupními piny. Na desce je osazen převodník USB na UART pro naprogramování mikrokontroleru. Proti samostatnému čipu ATmega je tento mikrokontroler vybaven zavaděčem, který slouží pro nahrávání programu pomocí sériové linky. [2]

3.1 Měření odebírané energie z rozvodné sítě

3.1.1 Elektrický výkon

Elektrický výkon je fyzikální veličina, která vyjadřuje vykonanou elektrickou práci za čas. Jeho jednotkou je watt a značí se písmenem W. U obvodů střídavého proudu se rozlišuje výkon činný, jalový a zdánlivý.

Pro výpočet elektrického výkonu musíme znát efektivní velikost napětí a proudu. Efektivní velikost periodické veličiny se spočítá jako její maximální hodnota vydělená odmocinou ze dvou.

3.1.2 Definice účinníku

Účinník je poměr činného a zdánlivého elektrického výkonu v obvodu střídavého proudu a napětí. Vyjadřuje jak velkou část energie přemění obvod na užitečný výkon. Účinník je kosinus vzájemného fázového posuvu mezi proudem a napětím elektrického obvodu.

Pro obvyklou spotřebičovou soustavu se účinník pohybuje od nuly do jedné. Pokud však máme soustavu, kde spotřebič, tedy domácnost, dodává energii do zdroje, respektive do sítě, pak může účinník nabývat i hodnot od mínus jedné

do nuly. Je-li účinník záporný, pak se jedná o dodávku energie ze spotřebiče do zdroje, v případě domácnosti s FVE o výrobu převyšující spotřebu.

3.1.3 Měření napětí a proudu

Pro správnou funkčnost wattrouteru je nutné měřit odebíranou elektrickou energii. Pro zjištění velikosti odebírané elektrické energie je nutné měřit napětí v přenosové síti a znát protékající proud. Zároveň je nutné znát účinník, neboť elektrárenské společnosti účtují činný výkon, proto je nutné vzorkovat proud i napětí a z jejich průběhu počítat účinník.

Výpočet činného výkonu je efektivní napětí vynásobené efektivním proudem a to celé ještě vynásobené účinníkem. Tedy kosinem vzájemného fázového posunu proudu a napětí.

3.1.4 Vzorkování napětí

Pro měření napětí je využit transformátor s poměrem závitů 230:6. Měření napětí probíhá na jeho výstupu přes odporový dělič. Pro vzorkování celého průběhu je výstup ještě napěťově posunut tak, aby celý průběh napětí byl v kladných hodnotách. Výstup je připojen k analogově digitálnímu převodníku Arduina a je vzorkován 100 krát za periodu, tedy každých $200\mu s$.

3.1.5 Vzorkování proudu

Měření proudu jsem v první fázi realizoval za pomoci proudového transformátoru. To je cívka o tisíci závitěch okolo fázového vodiče, ke které je připojena odporová zátěž. V závislosti na úbytku napětí na zátěži, počtu závitů cívky a velikosti zátěže lze vypočítat hodnotu protékajícího proudu. Výpočet velikosti protékajícího proudu je úbytek napětí na zátěži děleno odpor zátěže a to celé krát počet závitů cívky proudového transformátoru.

Tato metoda nebyla spolehlivá, neboť Arduino má na vstupech ochranné diody, aby nedošlo k poškození pinů. Tyto diody vykazují nějaký úbytek napětí a výrazně zkreslí průběh napětí generového protékajícím proudem na proudovém transformátoru. Z tohoto důvodu jsem provedl odfiltrování záporných půlvln měřeného napětí a zesílení operačním zesilovačem [6].

Odfiltrování záporného napětí a zesílení operačním zesilovačem se ukázalo jako možné řešení, ale opět nebylo úplně správné. Pro bezproblémovou funkčnost je nutné stále a navíc symetrické napájecí napětí pro operační zesilovač. Symetrické napětí jsem vyrobil pomocí integrovaného obvodu s nábojovou pumpou. Výstupní napětí nábojové pumpy nebylo dokonale vyhlazené, neboť i vstupní napětí používané pro napájení Arduina bylo se zákmity. Tyto zákmity způsobovaly odchylky v měření proudu o nemalé hodnoty.

Nakonec jsem pro měření proudu využil integrovaný obvod s Hallovou sondou. Obvod je napájen 5V a jeho výstup kopíruje průběh proudu. Pokud

je protékající proud nulový, pak je na výstupu integrovaného obvodu 2,5V. V tomto případě není problém vzorkovat celé periody průběhu proudu a znát tak maximum i minimum.

Rád bych zmínil, že s problémy měření protékajícího proudu mi pomohl a poradil Dr.-Ing. Martin Novotný z katedry číslicového návrhu na FIT ČVUT.

```

1 unsigned int i = 0;
2 unsigned long int correction = millis();
3 while(correction >= millis()) {}
4 while(i < 100) {
5     startMeas = micros();
6     v[i].value = analogRead(12);
7     v[i].timestamp = startMeas;
8     i++;
9     while(micros() - startMeas < 200) {}
10 }
11 i=0;
12 while(correction + 40 >= millis()) {}
13 while(i < 100) {
14     startMeas = micros();
15     if (f == 0) l[i].value = analogRead(15);
16     if (f == 1) l[i].value = analogRead(14);
17     if (f == 2) l[i].value = analogRead(13);
18     l[i].timestamp = startMeas;
19     i++;
20     while(micros() - startMeas < 200) {}
21 }

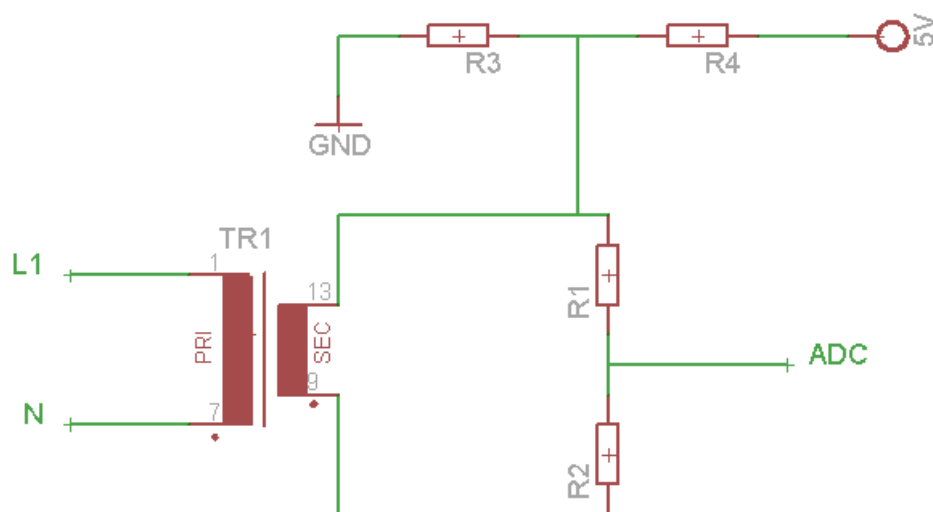
```

Listing 3.1: Ukázka vzorkování hodnot proudu a napětí

Vzorkování napětí a proudu není okamžité. Vzorkování hodnoty analogově digitálním převodníkem trvá nějaký čas. Pokud bych chtěl zaznamenávat napětí i proud současně, tak by analogově digitální převodník nestihl převést sto vzorků za periodu. Perioda odpovídá 20ms. Proto jsem zvolil variantu záznamu hodnot napětí a proudu zvlášť. Po zaznamenání hodnot napětí se vyčká tak, aby odstup začátku měření napětí a proudu byl 40ms, to odpovídá dvěma periodám. Pro výpočet účinníku odečítám tuto pauzu.

3.2 Nadvýroba a její využití

Pro vyhodnocení nadvýroby a následné spínání spotřebičů je potřeba znát směr toku proudu. Směr toku proudu přímo souvisí s účinníkem. Pokud je účinník záporný, pak teče proud ze spotřebiče do zdroje, z domu do distribuční sítě. Pokud je tedy vyhodnocen směr toku proudu z domu do distribuční sítě, pak se jedná o nadvýrobu. V takovém případě je potřeba zjistit, zda je možné



Obrázek 3.1: Zapojení měřícího modulu napětí s oddělovacím transformátorem. Vytvořeno v programu Eagle. [3]

tento přebytečný proud někde spotřebovat. Spotřeba přebytečného proudu se ve většině případů finančně vyplatí.

3.3 Nízký tarif

Takzvaný nízký tarif využívají elektrárenské společnosti k tomu, aby rozložily zátěž, kterou generují trvale připojené spotřebiče, například bojlerů či akumulací kamna. V době aktivního nízkého tarifu je proud levnější a lidé tedy mají motivaci pro zapínání bojlerů, akumulací kamen a dalších spotřebičů právě v tuto dobu. Každá přípojka k rozvodné síti má nízký tarif aktivní v jiné době tak, aby se zátěž rozložila a nebyla nárazově zvýšena v jeden okamžik.

Pro zapínání bojlerů či akumulací kamen se používají stykače, které jsou řízeny signálem ze spínače nízkého tarifu. Tento spínač může být buď samostatný nebo integrovaný přímo v elektroměru.

3.3.1 Detekce a jeho využití

Detekci aktivního nízkého tarifu jsem vyřešil elektromagnetickým relé s řídicí cívkou 230V střídavého napětí a spínacím kontaktem. Tímto relé je spínáno napětí 5V, které je přivedeno na vstupní digitální pin Arduina. Cívka je připojena na výstup spínače nízkého tarifu. Pokud se bude pin nacházet v logické jedničce, tedy bude na něm úroveň napětí 5V, pak je aktivní nízký tarif.

Využití nízkého tarifu může být zahrnuto v plánu automatizovaného spínání jednotlivých spotřebičů. Uživatel si může zvolit běh spotřebiče na nízký tarif, pokud nebyl spotřebičem vyčerpán denní limit. Tato funkce může být vhodná například pro dohřev vody v bojleru po západu slunce.

3.4 Spínání spotřebičů

Spínání spotřebičů je řešeno prostřednictvím digitálních výstupních pinů, které budou přepínány mezi dvěma stavy. K digitálnímu pinu je připojeno elektromagnetické relé řízené napětím 5V s přepínacím kontaktem do 230V střídavého napětí a maximálním proudovým zatížením 10A.

Pro spínání zátěže doporučuji tímto relé řídit stykač spotřebiče a to zejména tehdy, pokud se jedná o jinou než ohmickou zátěž. Řídící cívku relé je nutné k Arduino připojit přes tranzistor, protože výstupní pin Arduina je omezen na maximální výstupní proud 20mA a k sepnutí kontaktu relé je třeba proud větší [7].

Logika spínání spotřebičů je implementována tak, aby pokryla co největší rozsah možností požadavků uživatele. Spotřebič je možné trvale zapnout či vypnout podle potřeby nebo využít automatický režim spínání.

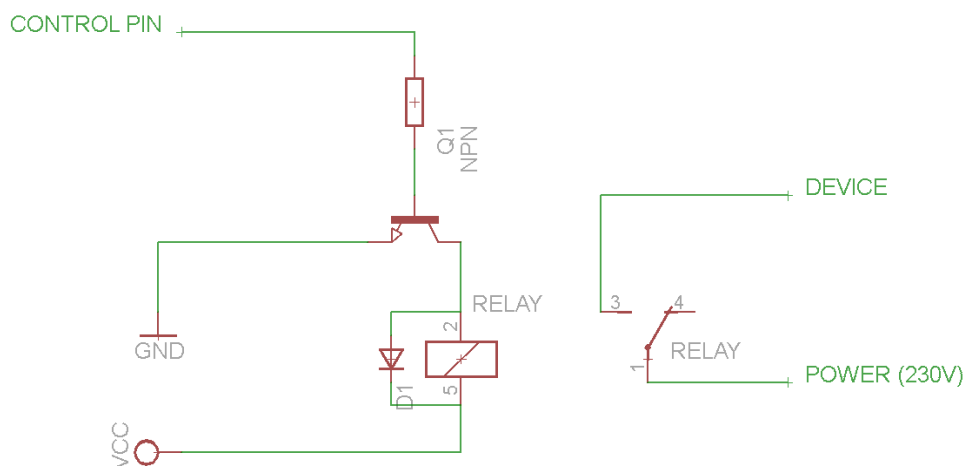
Automatickému režimu řízení spínání je potřeba nastavit:

- informaci o fázi, ke které jse postřebič připojen,
- maximální možný příkon spotřebiče,
- denní limit,
- prioritu spínání,
- povolení či zakázání běhu na nízký tarif, pokud FVE nedodá dostatečné množství energie během dne.

Wattrouter tak bude hlídat výrobu FVE a spotřebu domu. Pokud výroba přeroste spotřebu, sepne se zařízení s nejvyšší prioritou. Pokud bude stále výroba přesahovat spotřebu, tak se budou postupně spínat další spotřebiče, dle priorit, dokud nebude výroba a spotřeba vyrovnána tak, aby do sítě netekl žádný proud. Pokud spotřebič již odebral tolik proudu, že překročil denní limit, tak se již nezapíná a na řadě jsou spotřebiče s nižší prioritou.

Se zapnutou funkcí spínání spotřebiče na nízký tarif se spotřebič sepne s aktivováním nízkého tarifu a to do doby než spotřebuje tolik, kolik má povoleno denním limitem.

Tato funkce se hodí například pro ohřátí vody v bojleru v den, kdy nepřeje počasí, je zataženo a výkon FVE není dostatečný pro ohřev vody v bojleru. Také se funkce může hodit ve chvíli, kdy se zvýší spotřeba domu v důsledku praní či vaření a přes den nenastane okamžik, kdy výroba přesáhne spotřebu.



Obrázek 3.2: Zapojení spínacího relé. Vytvořeno v programu Eagle. [3]

3.5 Kontrola zapnutého spotřebiče

Po zkušenosti s wattrouterem od firmy SolarControls s.r.o. jsem se rozhodl pro implementaci funkce rozpoznávající zapnutý spotřebič. Dost často se totiž stává, že wattrouter sice sepne stykač spotřebiče, ten je ale vypnut termostatem či vypínačem a wattrouter ho považuje za zapnutý. Wattrouter mu odečítá denní limit spotřebované energie, která se reálně nespotřebovává. Tento problém lze vyřešit například senzorem průtoku proudu pro každý spotřebič. To je však finančně nákladné řešení. Další možností je kontrola zapnutého spotřebiče pomocí měření celkového odběru.

Detekce zapnutého spotřebiče funguje tak, že spotřebiči sepne stykač a změříme aktuální odběr na patřičné fázi. Následně spotřebič vypneme, opět změříme odběr proudu a poté ho znovu sepne a opět změříme odběr. Tímto cyklem dostaneme tři hodnoty, které porovnáme. První a třetí by se měly zhruba shodovat a druhá by měla být nižší o nastavený odběr spotřebiče. Pokud hodnoty odpovídají, pak vyhodnotíme spotřebič jako sepnutý. Tuto kontrolu provádím každých 5 minut. Vypnutí a opětovné zapnutí spotřebiče se děje v rámci několika vteřin, kdy je minimální pravděpodobnost, že by podobný výkyv ve spotřebě mohl vzniknout například zapnutím a vypnutím jiného většího spotřebiče.

3.6 Ovládací prvky a periferie

Navržený wattrouter disponuje displejem a tlačítky pro přímé, rychlé a základní ovládání. Dále je k dispozici připojení do počítačové sítě pomocí wifi modulu a nebo ovládání z PC pomocí USB portu. Konfigurace wattrouteru,

denní spotřeba a doby běhu spotřebičů jsou uloženy na SD kartě v textových souborech s pevně daným formátem.

3.6.1 Zobrazování dat

Pro zobrazování dat jsem použil LCD znakový displej. Připojení displeje je realizováno prostřednictvím I2C sběrnice.

Displej má čtyři řádky a každý řádek má délku dvaceti znaků. Na spodním řádku se zobrazují vždy obecné informace o době běhu wattrouteru, vložené SD kartě, připojení do sítě a aktivním režimu přístupového bodu. Dále se ze spodního řádku můžeme dozvědět aktuální čas. První tři řádky displeje obsahují informace mezi kterými se můžeme přesouvat pomocí tlačítek, celkem šest rozdílných obrazovek.

- První obrazovka obsahuje informace o celodenní spotřebě na jednotlivých fázích. Tyto počítačidla se vždy o půlnoci vynulují a jejich stav je ukládán do souboru na SD kartě.
- Druhá obrazovka zobrazuje aktuální odběry na jednotlivých fázích a napětí na první fázi. Pokud se hodnota odběru dostane do záporu pak se jedná o přesah výroby nad spotřebu, čili dodávku do sítě.
- Třetí obrazovka informuje o stavu jednotlivých připojených spotřebičů. U každého spotřebiče se můžeme dozvědět na které fázi je připojen, jaký je jeho denní limit spotřeby energie, současný stav odebraného množství energie a zda je trvale zapnut nebo vypnut či řízen automaticky. Pokud je aktivní automatické řízení, pak bude na displeji indikováno zda je spotřebič zapnutý či vypnutý.
- Čtvrtá a pátá obrazovka nesou informace o připojení do sítě, přičemž jedna je vyhrazena pro připojení k bezdrátové síti a druhá pro režim přístupového bodu. Na obrazovkách se dozvíme IP adresu zařízení a SSID sítě ke které je připojeno, respektive kterou síť poskytuje.
- Šestá obrazovka ukazuje stav wattrouteru, především jeho teplotu, aktivní či neaktivní nízký tarif a další informace.



Obrázek 3.3: Ukázka zobrazení denní spotřeby.

Pro práci s LCD displejem využívám knihovnu od Frank de Brabande z GitHub, která umí pracovat s displejem připojeným po I2C sběrnici. [8]

3.6.2 Tlačítka

Wattrouter má pro rychlé ovládání k dispozici pět tlačítek uspořádaných do pomyslného joysticku. Tlačítka vlevo a vpravo se lze posouvat mezi obrazovkami, v případě obrazovky s připojenými spotřebiči lze využít tlačítek nahoru a dolů pro přepínání zobrazovaného spotřebiče. Středové tlačítko slouží pro změny. V případě zobrazeného spotřebiče lze přepínat režim řízení, na obrazovkách s informacemi o připojení do počítačové sítě lze středovým tlačítkem vypnout či zapnout režim přístupového bodu nebo připojení k bezdrátové síti.

3.6.3 Ukládání dat

Pro ukládání historie odběru a časů, kdy byly zapnuté spotřebiče, jsem použil SD kartu připojenou po sběrnici SPI k Arduinu. Na kartě je také uložen konfigurační soubor, který obsahuje informace pro připojení k bezdrátové síti a konfiguraci připojených spotřebičů.

Všechny soubory jsou obyčejné textové s pevně daným formátem. Modul pro SD kartu je třeba připojit přes odporové děliče, neboť jeho pracovní napětí je 3,3V a Arduino pracuje s 5V. Použil jsem odporový dělič pro komunikační piny CLK (hodinový signál), MOSI (tok dat směrem na kartu), CS (pin sloužící pro výběr zařízení při připojení více zařízení). Pro pin MISO odporový dělič nepoužijeme neboť slouží pro tok dat z karty do Arduina a vysílá ho modul na úrovni 3,3V, kterou je Arduino schopné rozpoznat. Pro napájení modulu čtečky SD karty můžeme použít 5V neboť je osazen stabilizátorem napětí, který si úroveň napětí přizpůsobí.

K ukládání konfigurace wattrouteru a stavu denních počítadel se ještě nabízela paměť EEPROM, která je k dispozici na čipu ATmega. EEPROM paměť má však omezený počet přepsání buňky a její kapacita je poměrně malá, takže by nebylo možné ukládat historii.

Pro práci s SD kartou využívám volnou knihovnu od Bill Gremian z GitHub, která umí číst a zapisovat textové soubory a na rozdíl od standardní knihovny Arduina umí rozpoznat odpojení SD karty za běhu. [9]

3.6.4 Modul reálného času

Z důvodu synchronizace wattrouteru s denní dobou a z důvodu rozumného ukládání dat je nutné mít k dispozici aktuální čas. Arduino samotné nedisponuje hodinami reálného času a je tedy nutné použít externí. Modul hodin reálného času je připojen po sběrnici I2C a udržuje v sobě aktuální datum a čas. V případě výpadku napájecího napětí se o běh hodin stará baterie. Ta je vložena do patice v modulu. Modul hodin reálného času také obsahuje teplo-
měr pro korekci času. Čas je odměřován pomocí oscilujícího krystalu, který

```
1 WIFI=ON
2 WIFI-SSID=FreeWifi
3 WIFI-PASS=abcdeedcba
4 AP=ON
5 AP-SSID=Wattrouter
6 AP-PASS=0123443210
7 AP-IP=192.168.146.1
8 APPLIANCE0=1,7,2000,10000,0,1
9 APPLIANCE1=1,4,1500,2000,0,1
10
```

Obrázek 3.4: Ukázka konfiguračního souboru.

má určitou frekvenci. Tuto frekvenci však ovlivňují výkyvy teplot, které jsou snímány teplotním čidlem a čas je následně upravován.

Pro zjištění aktuálního data, času nebo teploty stačí přečíst hodnoty uložené v modulu. Pro čtení jsou použity funkce, které pracují se standardní knihovnou Arduina, Wire [10]. Ta je určena pro práci s I2C sběrnici.

3.6.5 Wifi modul

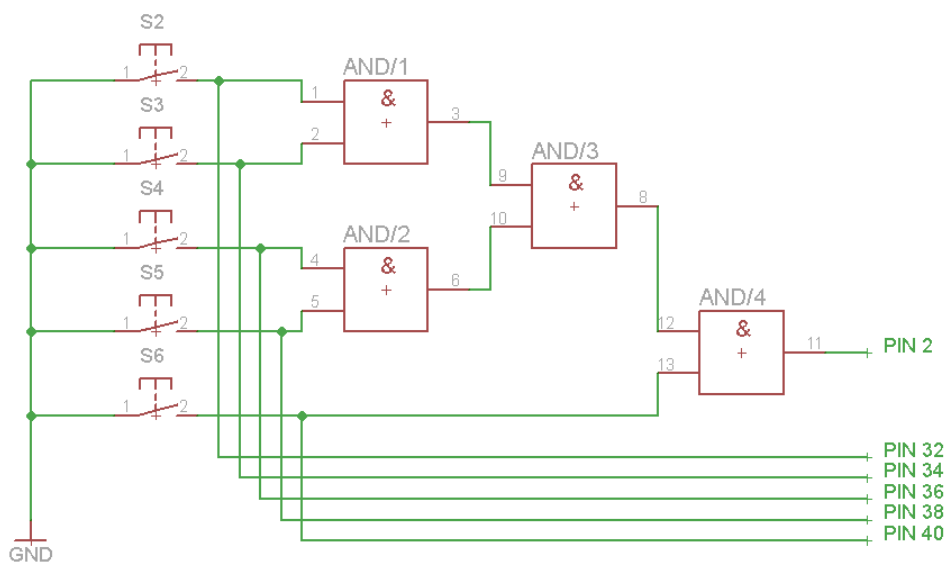
K připojení do počítačové sítě jsem použil wifi modul ESP8266-1. Tento modul komunikuje pomocí sériové linky, po které se posílají tzv. AT příkazy [11]. Pro práci s modulem existují knihovny, ale většina z nich nemá implementované funkce, které jsem potřeboval. Z tohoto důvodu jsem si napsal několik vlastních funkcí, které se starají o funkčnost wifi modulu. Ke komunikaci jsou využity tři funkce, z nichž jedna je obsluhou přerušení, které je vyvoláno příchozí komunikací od modulu. Další odesílá příkazy k vykonání do modulu a poslední slouží pro prvotní konfiguraci modulu po spuštění celého wattrouteru. Dále ještě používám jednu proměnnou jako flag, který je použit ve všech funkcích. Tento flag signalizuje zda se čeká na odpověď od modulu nebo je možné odeslání dalšího příkazu.

Funkce pro konfiguraci spustí sériovou komunikaci na pinech, ke kterým je připojen modul. Vyčká několik vteřin, protože modul po spuštění komunikace pošle zprávu o svém stavu. Následně je modul zresetován a provedena příprava pro připojení k bezdrátové síti.

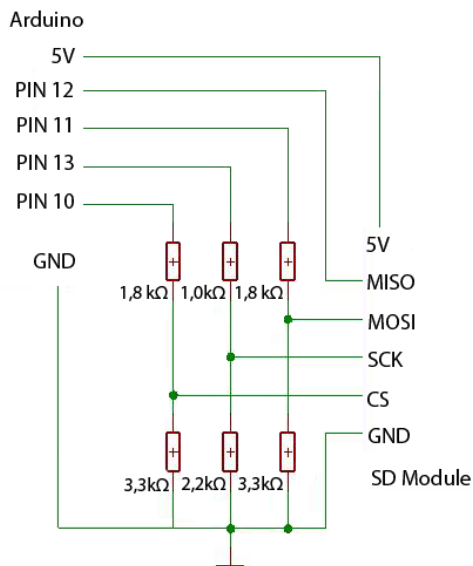
Funkce pro odesílání příkazů přijme jako parametr příkaz, který zpracuje, pošle po sériové lince do modulu a přepíše flag signalizující čekání na odpověď.

Funkce obsluhy přerušení čekající na příchozí data od modulu, se aktivuje v momentě, kdy modul odpoví na zadaný příkaz. Odpověď je zpracována a flag, signalizující čekání na odpověď, se vrátí zpět do stavu, kdy je modul volný a čeká na další příkaz.

3. REALIZACE



Obrázek 3.5: Zapojení modulu s tlačítky. Vytvořeno v programu Eagle. [3]



Obrázek 3.6: Připojení modulu čtečky SD karet pomocí odporových děličů. Vytvořeno v programu Eagle. [3]

3.7 Konfigurační a dotazovací příkazy

Pro komunikaci s počítačem či jiným zařízením používá wattrouter jak sériovou linku, tak také komunikaci po síti řešenou TCP spojením. Na sériové lince se jedná o definované textové řetězce s parametry zakončené řídicími znaky `\CR` a `\LF`. Pro síťovou komunikaci jsou to textové řetězce, které nese TCP paket.

3.7.1 Příkazy na sériové lince

Přijímání příkazů má na starost sériová linka implementovaná v mikrokontroleru. Příchozí znaky jsou ukládány do fronty, která je následně rozdělena na jednotlivé příkazy a zpracována obsluhou přerušení.

3.7.2 Tabulka příkazů

Příkaz	Parametry / Popis
SETTIME: [T]	[T] 14 číslic, vždy po dvou udávají hodnoty v pořadí: hodiny, minuty, vteřiny, den v týdnu, den v měsíci, měsíc a rok Po nastavení času dostaneme odpověď ve formátu TIME: [T] se 14 číslicemi udávajícími aktuální čas ve stejném formátu jako při nastavování.
GETTIME	bez parametrů Vrátí odpověď ve formátu TIME: [T] se 14 číslicemi, stejně jako u příkazu SETTIME.
GETTEMP	bez parametrů Vrátí odpověď ve formátu TEMP: [X], kde [X] je teplota wattrouteru ve stupních celsia.
GETUPTIME	bez parametrů Vrátí odpověď ve formátu UPTIME: [X], kde [X] je počet milisekund od zapnutí wattrouteru.

3. REALIZACE

Příkaz	Parametry / Popis
GETIP	bez parametrů Příkaz vrátí dva řádky z nichž první obsahuje STA: [X], kde [X] je buď IP adresa nebo řetězec OFF, v případě, že je režim klienta vypnutý. Druhý řádek obsahuje AP: [X], kde [X] je opět buď IP adresa přístupového bodu a nebo řetězec OFF v případě vypnutého režimu přístupového bodu.
GETWIFI	bez parametrů V odpovědi vrátí dva řádky, z nichž jeden obsahuje SSID:"[SSID]" a druhý PASS:"[PASS]", dle nastavení připojení k bezdrátové síti.
GETAP	bez parametrů V odpovědi vrátí dva řádky, z nichž jeden obsahuje SSID:"[SSID]" a druhý PASS:"[PASS]", dle nastavení poskytované bezdrátové sítě.
SETWIFI:"[SSID]", "[PASS]" SETWIFI:[X]	<ul style="list-style-type: none">• [SSID] název požadované bezdrátové sítě• [PASS] je heslo k bezdrátové síti, pokud není zabezpečená, pak se parametr vynechává <p>Po nastavení nového SSID a hesla bezdrátové sítě obdržíme dva řádky, z nichž jeden obsahuje SSID:"[SSID]" a druhý PASS:"[PASS]"</p> [X] <ul style="list-style-type: none">• ON zapnutí vysílání bezdrátové sítě• OFF vypnutí vysílání bezdrátové sítě <p>Příkaz zapne či vypne režim klienta na wifi modulu. A po přepnutí vrací oznámení ve tvaru WIFI:[X].</p>

3.7. Konfigurační a dotazovací příkazy

Příkaz	Parametry / Popis
SETAP: " [SSID] ", " [PASS] " SETAP: [X]	<ul style="list-style-type: none"> • [SSID] název vysílané bezdrátové sítě • [PASS] je heslo k bezdrátové síti <p>Po nastavení nového SSID a hesla bezdrátové sítě obdržíme dva řádky, z nichž jeden obsahuje SSID: " [SSID] " a druhý PASS: " [PASS] "</p> <p>[X]</p> <ul style="list-style-type: none"> • ON zapnutí vysílání bezdrátové sítě • OFF vypnutí vysílání bezdrátové sítě <p>Příkaz zapne či vypne režim klienta na wifi modulu. A po přepnutí vrací oznámení ve tvaru AP: [X].</p>
ADDDEVICE: " [ID] ", " [P] ", " [R] ", " [C] ", " [L] ", " [H] ", " [O] "	<ul style="list-style-type: none"> • [ID] je číslo spotřebiče (1-8) • [P] číslo fáze kde je spotřebič připojen (1-3) • [R] priorita spotřebiče (1-255), nižší číslo - vyšší priorita • [C] příkon spotřebiče ve watttech • [L] denní limit odběru energie ve watt-hodinách (0 - bez omezení) • [H] povolení sepnutí na nízký tarif (0/1) • [O] režim řízení (-1 - OFF, 0 - AUTO, 1 - ON) <p>Přidá spotřebič s požadovaným číslem a vrátí:</p> <ul style="list-style-type: none"> • ADDDEVICE:OK v případě úspěchu • ADDDEVICE:ERR v případě neúspěchu.

3. REALIZACE

Příkaz	Parametry / Popis
SETDEVICE: "[ID] ", "[P] ", "[V] "	<ul style="list-style-type: none"> • [ID] je číslo spotřebiče (1-8) • [P] udává parametr pro nastavení - písmeno z příkazu ADDDEVICE • [V] nová hodnota parametru <p>Změní požadovaný parametr spotřebiče:</p> <ul style="list-style-type: none"> • SETDEVICE:OK v případě úspěchu • SETDEVICE:ERR v případě neúspěchu.
RMDEVICE: "[ID] "	<p>[ID] je číslo zařízení (1-8)</p> <p>Odstraní spotřebič s požadovaným číslem a vrátí:</p> <ul style="list-style-type: none"> • RMDEVICE:OK v případě úspěchu • RMDEVICE:ERR v případě neúspěchu.
GETDEVICE: "[ID] "	<p>[ID] je číslo spotřebiče (1-8)</p> <p>Vrátí nastavení spotřebiče s požadovaným číslem ve formátu DEVICE: "[ID] ", "[P] ", "[R] ", "[C] ", "[L] ", "[H] ", "[O] ". Parametry odpovídají parametrům příkazu ADDDEVICE. Pokud spotřebič neexistuje vrátí GETDEVICE:ERR.</p>

Příkaz	Parametry / Popis
GETDEVICEI:" [ID] "	<p>[ID] je číslo spotřebiče (1-8)</p> <p>Vrátí stav spotřebiče s požadovaným číslem ve formátu DEVICEI:" [ID] ", "[TC] ", "[PWR] ".</p> <ul style="list-style-type: none"> • [ID] je číslo spotřebiče (1-8) • [TC] udává denní spotřebu ve watthodinách • [PWR] informuje o zapnutém (0) či vypnutém (1) spotřebiči <p>Pokud spotřebič neexistuje vrátí GETDEVICEI:ERR.</p>
GETU	<p>bez parametrů</p> <p>Vrátí odpověď ve formátu U: [X], kde [X] je efektivní hodnota napětí naměřená na první fázi.</p>
GETL [X]	<p>[X] udává číslo fáze (1-3)</p> <p>Vrátí odpověď ve formátu L [X] : [Y], kde [Y] je efektivní hodnota protékajícího proudu na fázi [X]. V případě jednofázového zapojení vrací pro fáze 2 a 3 odpověď L [X] :NC</p>
GETC [X]	<p>[X] udává číslo fáze (1-3)</p> <p>Vrátí odpověď ve formátu C [X] : [Y], kde [Y] je činný výkon spotřebovaný od půlnoci na fázi [X]. V případě jednofázového zapojení vrací pro fáze 2 a 3 odpověď C [X] :NC</p>

3.7.3 Implementované příkazy sériové linky

V rámci realizace se zabývám pouze prototypem wattrouteru a tedy nejsou implementovány všechny příkazy. Pro komunikaci sériové linky jsou implementovány příkazy: SETTIME, GETTIME, GETTEMP, GETUPTIME, GETIP, GETWIFI, SETWIFI, RMDEVICE, SETDEVICE, GETDEVICE, GETDEVICEI.

3.7.4 Příkazy po síti

Přijímání paketů s příkazy zajišťuje wifi modul, který je následně předává po sériové lince mikrokontroleru. Ten si z paketu nechá jeho datovou část,

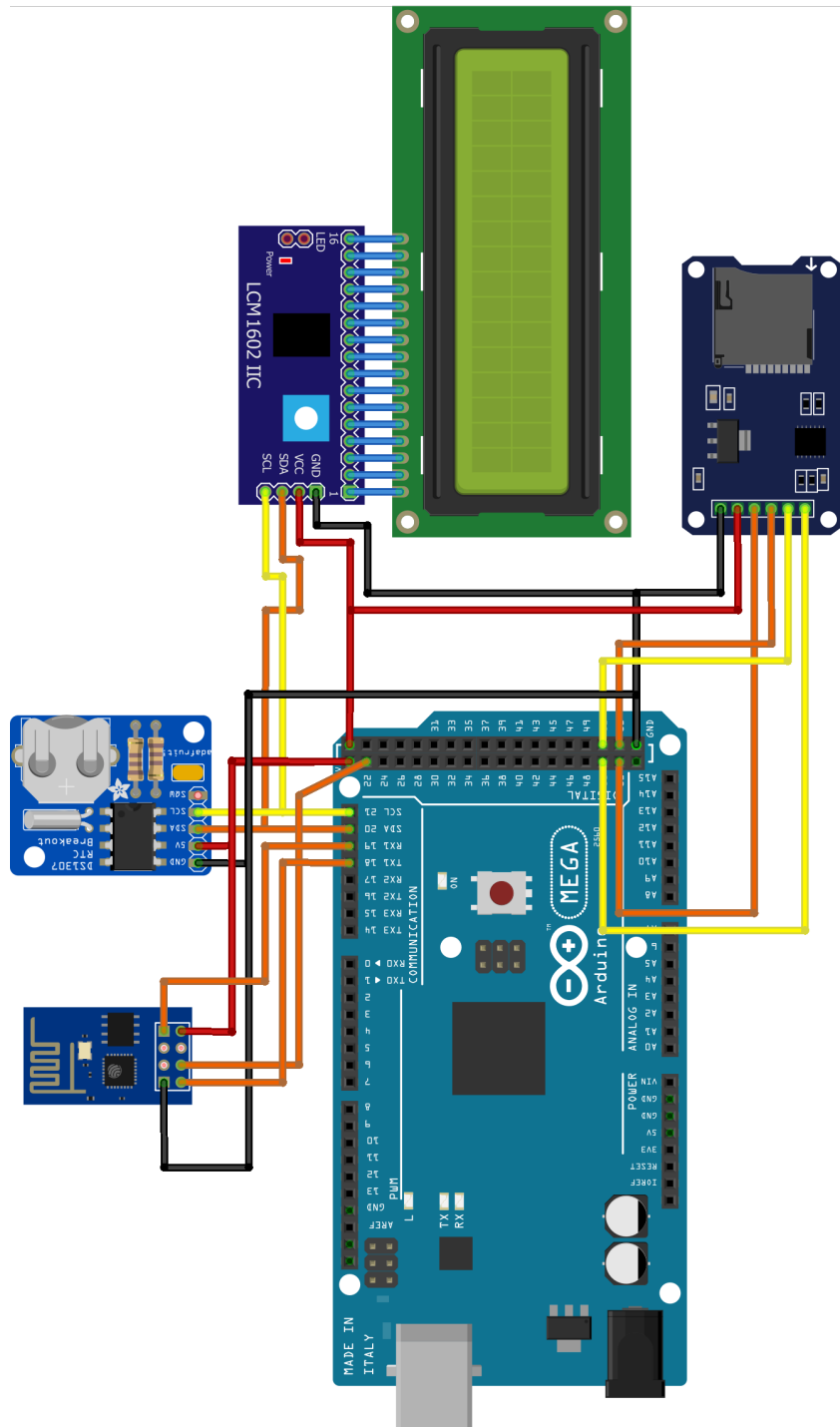
3. REALIZACE

kteřou si zařadí do fronty ke zpracování. Z důvodu možnosti připojení více zařízení je možné provádět příkazy, které vedou k nějaké změně nastavení, pouze z prvního připojeného zařízení. Po tom co se první zařízení odpojí, tak je předána možnost změny nastavení dalšímu zařízení v pořadí.

Příkazy pro síťovou komunikaci mají shodný formát s příkazy pro sériovou linku. Jediný rozdíl je v tom, že příkazy nekončí řídicími znaky `\CR` a `\LF`. Příkazy jsou zasílány jako obsah TCP paketů, každý příkaz je v samostatném paketu.

3.7.5 Implementované příkazy síťové komunikace

V rámci realizace se zabývám pouze prototypem wattrouteru a tedy nejsou implementovány všechny příkazy. Pro síťovou komunikaci jsou implementovány příkazy: `SETTIME`, `GETTIME`, `GETU`, `GETL`, `GETC`.



Obrázek 3.7: Blokové schéma připojení periférií. Vytvořeno v programu Fritzing. [4]

Testování

4.1 Měření napětí a výpočet efektivní hodnoty

Po zapojení měřicího obvodu napětí jsem provedl jednoduché změření napětí pomocí voltmetru. Měření mi potvrdilo odhad snížení napětí. Připojil jsem tedy výstup měřicího obvodu na analogově digitální převodník Arduina a zkusil navzorkovat hodnoty. Do funkce používané pro vzorkování napětí jsem přidal přepočítání naměřených úrovní na reálné napětí, které je v síti. Následně jsem zkontroloval průběh napětí pomocí osciloskopu a vyhodnotil, že výsledky jsou správné.

Měření napětí probíhalo na napětí, které bylo generované laboratorním zdrojem s výstupem sinusového průběhu napětí.

Výpočet efektivní hodnoty napětí pak už spočívá jen v nalezení maximální hodnoty napětí a jejím vydělením odmocninou ze dvou. Pro přesnější zjištění využívám maximální i minimální hodnotu napětí a pro výpočet použiji větší absolutní hodnotou ze dvou uvedených. PROČ?! Popsat proč min i max hodnotu.

4.2 Měření proudu a výpočet efektivní hodnoty

Měření proudu nebylo nakonec tak jednoduché, jak se zdálo. První měření s využitím proudového transformátoru, což je vlastně cívka okolo fázového vodiče, ukázalo že toto řešení není dostatečně přesné. Proudový transformátor funguje tak, že pokud fázovým vodičem protéká střídavý proud, tak se na cívce indukuje napětí a zátěží připojenou k cívce protéká proud. Pomocí analogově digitálního převodníku měříme úbytek napětí na zátěži. Z něho vypočteme velikost protékajícího proudu fázovým vodičem. Pokud fázovým vodičem protéká proud opačným směrem, pak je i napětí na zátěži vůči zemi opačné, tedy záporné.

Řešení s proudovým transformátorem však mělo úskalí právě ve výše zmíněném záporném napětí, které se nesmí dostat na analogový pin Arduina.

Arduino je z bezpečnostních důvodů vybaveno ochrannými diodami, které záporné napětí odfiltrují. Tyto diody však způsobují úbytek napětí a zkreslí jinak sinusový průběh napětí.

V druhém řešení jsem se rozhodl pro odfiltrování záporných půlvln, tedy záporného napětí, využít diodu. Ta ale způsobuje úbytek napětí, který by byl tak veliký, že by významně omezil přesnost měření menších proudů. Řešení se nabízelo v použití operačního zesilovače, který by signál zesílil.

Zprovoznil jsem prototyp zapojení obvodu s operačním zesilovačem, odfiltroval pomocí diody záporné napětí a výstup připojil k osciloskopu. Výsledek nebyl nijak uspokojivý. Výstupní napětí bylo zarušené zámkity, které způsoboval zdroj napětí operačního zesilovače.

Jako třetí řešení přišla na řadu Hallova sonda v integrovaném obvodu. Provedl jsem připojení sondy, zkusil základní změření výstupního napětí pomocí voltmetru a připojil jsem výstup k Arduino. Navzorkování hodnot se zdařilo, výsledný průběh proudu byl uspokojivý. Ani toto řešení však není ideální, bylo by potřeba další doladění pro zpřesnění měření menších proudů. Z důvodu kolísání výstupního napětí při průtoku malého nebo žádného proudu jsem nastavil omezení, že protékající proud do 0,5A je považován za nulový.

Měření proudu opět probíhalo za pomoci laboratorního zdroje a oddělovacího transformátoru.

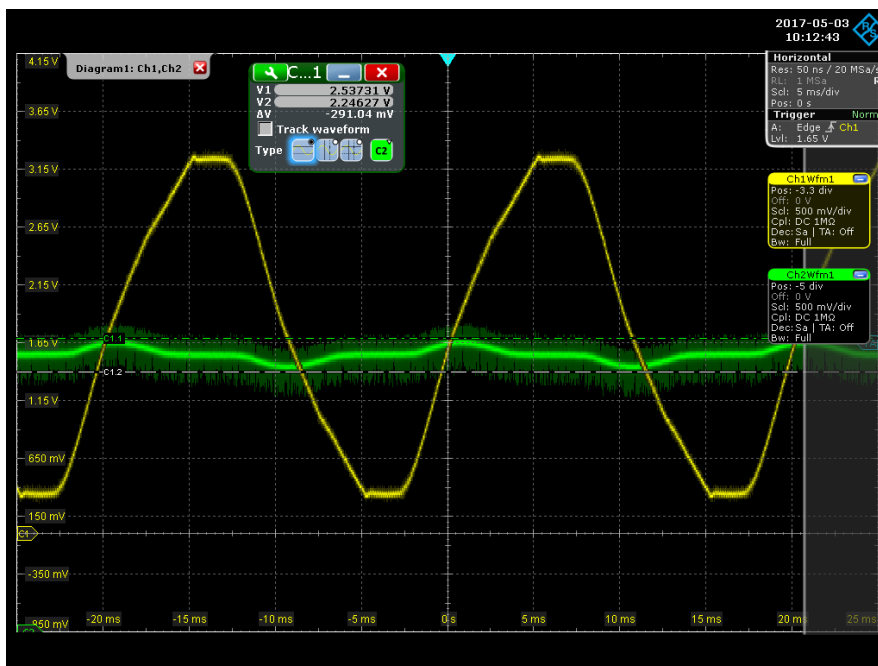
Výpočet efektivní hodnoty proudu se provádí shodně s výpočtem efektivní hodnoty napětí.

4.3 Detekce směru toku proudu

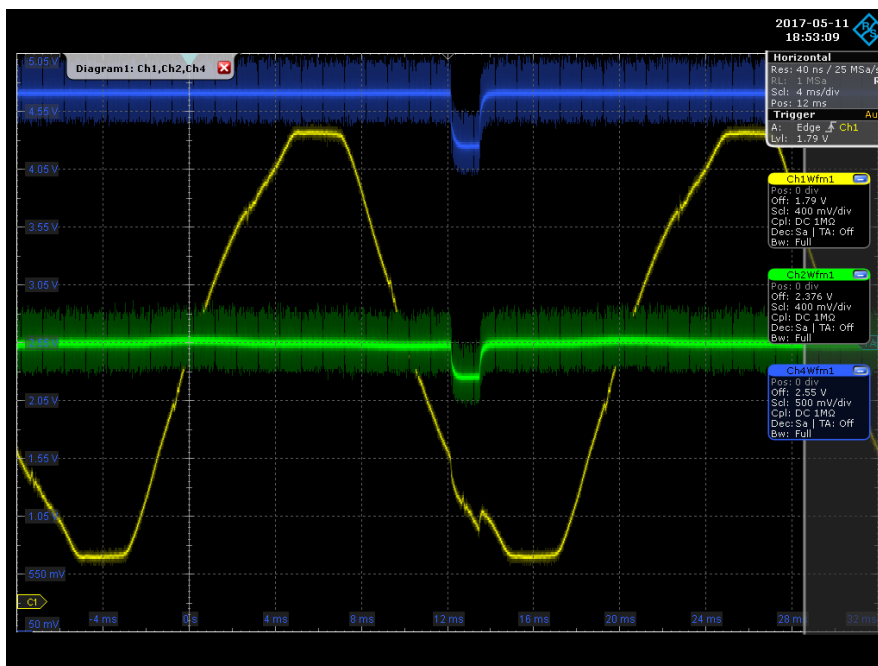
Detekce směru toku proudu přímo souvisí s účínkem elektrického výkonu. Výpočet účínku provádím následujícím způsobem. V první řadě si zjistím čas, ve kterém je napětí na první fázi nulové a pamatuji si, zda předtím bylo kladné či záporné. Dále zjistím čas, ve kterém byl nulový proud tak, aby jeho předchozí hodnoty byly znaménkově shodné s napětím. Z rozdílu času pak dopočítám úhlové zpoždění a zjistím si kosinovou hodnotu úhlu, která je vlastně účínek. Účínek nabývá hodnot od -1 do 1. Pokud je hodnota účínku záporná, pak se jedná o opačný směr toku proudu tedy o dodávku do sítě, což znamená přesah výroby nad spotřebou.

V případě toku proudu do sítě (tedy záporného účínku) běží proud tzv. proti napětí. Ve chvíli, kdy se napětí nachází v kladné půlvlně, je proud v záporné a opačně. Pokud proud teče standardně ze sítě, pak je napětí i proud vždy kladný nebo vždy záporný a účínek nabývá kladných hodnot.

4.3. Detekce směru toku proudu

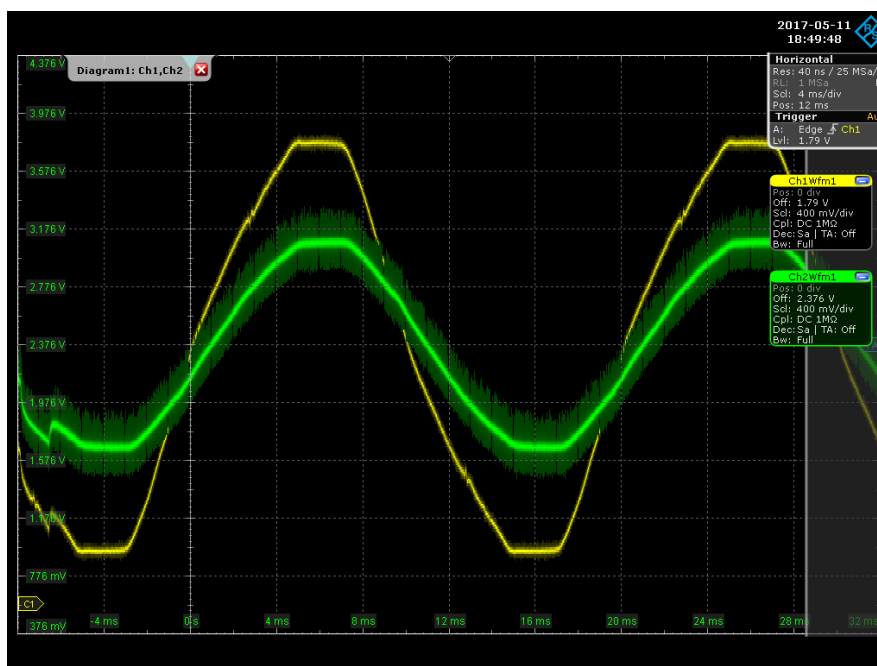


Obrázek 4.1: Ukázka nepřenosti měření bez zátěže. (žlutá: napětí v síti, zelená: protékající proud)

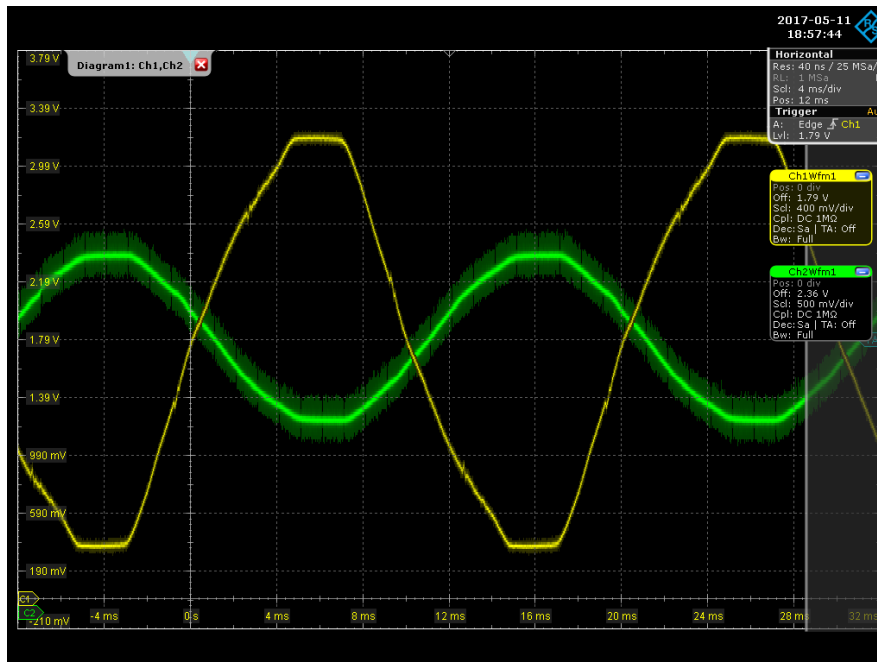


Obrázek 4.2: Ukázka záskmitů napájecího napětí wattrouteru se způsobenou nepřeností měření. (žlutá: napětí v síti, zelená: protékající proud, modrá: napájecí napětí)

4. TESTOVÁNÍ



Obrázek 4.3: Ukázka průběhu proudu a napětí při spotřebě. (žlutá: napětí v síti, zelená: protékající proud)



Obrázek 4.4: Ukázka průběhu proudu a napětí při výrobě. (žlutá: napětí v síti, zelená: protékající proud)

4.4 Připojení a testování periférií

4.4.1 Displej

Připojení displeje je také realizováno prostřednictvím sběrnice I2C. Po sběrnici proudí data o pozici kurzoru a adrese zobrazeného znaku, který je uložen v paměti displeje. Paměť displeje je realizována EEPROM pamětí a obsahuje pouze ASCII znaky. Displej má podsvícení, které lze zapínat a vypínat programově, nastavení kontrastu je možné měnit pouze pomocí osazeného trimru. S připojením displeje se neobjevili žádné problémy.

Pro práci s LCD displejem využívám knihovnu od Frank Brabande z GitHub, která umí pracovat s displejem připojeným po I2C sběrnici. [8]

4.4.2 Tlačítka

Modul s tlačítky jsem se rozhodl vyrobit vlastní.

Rozhodl jsem se pro pět tlačítek uspořádaných do kříže se středovým tlačítkem. Protože jsem chtěl reagovat na stisk tlačítek okamžitě, bylo nutné vymyslet připojení k jednomu pinu s povoleným přerušením. Arduino totiž disponuje pouze dvěma piny pro přerušení.

Z důvodu úspory práce a prostoru jsem volil řešení, kde tlačítka ukostřují vstupní pin a piny jsou v režimu vstupu s integrovaným pull-up odporem. Všechny výstupy tlačítek jsou připojeny na vstupy logického hradla AND. V klidovém režimu jsou na vstupech hradla logické jedničky a tedy i na výstupu je logická jednička, která je připojena na pin s povoleným přerušením. V případě stisku tlačítka se příslušný vstup překlápí do logické nuly a tím se změní i výstup hradla, na ten reaguje obsluha přerušení. Obsluha přerušení zkontroluje stavy všech pěti pinů, na které jsou připojena tlačítka, a vyhodnotí, která jsou stisknuta. V závislosti na stisknutém tlačítku se provede příslušná operace.

S tlačítky se objevil problém zákmitů při stisku, tyto zákmity jsem ošetřil kondenzátory. Bohužel to nebylo úplně ideální řešení. Nakonec jsem problém vyřešil díky nedostatku napájecích pinů pro periférie. Rozhodl jsem se odpojit napájení logického hradla AND a použít ho pro napájení modulu hodin reálného času. Na odpojení napájení jsem zapomněl a záhy zjistil, že hradlo i přes absenci napájení funguje a navíc pohlcuje zákmity generované tlačítky. Toto řešení není ideální, nicméně řeší popsany problém.

4.4.3 Modul pro SD kartu

Připojení modulu pro práci s SD kartou je realizováno pomocí sběrnice SPI. Modul pracuje s napětím 3,3V na rozdíl od Arduina, které pracuje s 5V. Je tedy nutné použít oddělovač logických úrovní anebo použít odporové děliče na vstupních pinech. Nižší napětí na výstupních pinech není problém, neboť ho Arduino dokáže rozpoznat. Při prvním zapojení jsem na odporové děliče

zapomněl a operace s kartou se chovaly podivně. Po přidání odporových děličů začalo vše fungovat podle mého očekávání.

Pro práci s SD kartou využívám knihovnu od Bill Gremian z GitHub, která umí číst a zapisovat textové soubory a na rozdíl od standardní knihovny Arduina umí rozpoznat odpojení SD karty za běhu. [9]

4.4.4 Modul reálného času

Modul reálného času je připojen stejně jako displej na sběrnici I2C. Napájen je pomocí 5V z desky Arduina. S připojením modulu reálného času se neprojevily žádné problémy. Pro komunikaci je použita knihovna Arduina s názvem LiquidCrystal [10], která se stará o komunikaci po sběrnici I2C.

4.4.5 Wifi modul

Připojení wifi modulu je realizováno pomocí sériové linky o rychlosti 115200 baudů. Modul podporuje režimy přístupového bodu a klienta bezdrátové sítě, je na něm možné spustit TCP server a obsluhovat zároveň až pět spojení. Veškerá komunikace je prováděna pomocí AT příkazů [11]. Tyto příkazy vznikly pro komunikaci modemů a dále se rozšířily i pro GSM moduly a další komunikační zařízení. Jedná se o definované textové řetězce, podle nich pak koncové zařízení vykonává nějakou činnost a vrací odpovědi.

Wifi modul ESP8266-1 po nějaké době nečinnosti přechází do úsporného režimu, kdy přeruší spojení a komunikace je tak zastavena. Proto je modulu v pravidelných intervalech zasílán příkaz AT, na který modul reaguje odpovědí OK, pokud je připraven. Takto udržuji modul ve stavu, kdy se neuspí a nepřeruší komunikaci.

4.5 Testování příkazů wattrouteru

4.5.1 Příkazy prostřednictvím sériové linky

Provedl jsem testování příkazů posílaných prostřednictvím sériové linky. Vše probíhalo korektně bez obtíží.

Po zpracování příkazu se zpět odesílá informace o úspěchu či neúspěchu, případně požadovaná data.

4.5.2 Příkazy prostřednictvím síťové komunikace

Každý příchozí paket je zařazen do fronty ke zpracování. Tato fronta se zpracovává průběžně, pokud není prázdná. Z paketu se čtou data, která jsou rozdělena na příkaz a jeho parametry. Následně je příkaz s parametry vykonán a odesílateli je zpět poslána informace o vykonání příkazu.

Zde jsem narazil na problém s velikostí paměti vyhrazené pro sériovou linku Arduina, která byla příliš malá. Modul ESP8266-1 komunikuje po sériové

lince. Arduino vyvolává přerušeni od seriové linky s prvním příchozím bajtem, přičemž přerušeni má za úkol vyprázdnit paměť pro příjem dat a nastavit příznak pro zpracování. Některé zprávy zasílané modulem však byly větší a tudíž bylo nutné zvětšit paměť pro příjem dat.

Budoucí práce

V budoucnu by bylo vhodné naprogramovat jednoduchou aplikaci pro počítač, která by uměla obsluhovat wattrouter jak po sériové lince, tak i prostřednictvím počítačové sítě. Někteří uživatelé wattrouteru by jistě ocenili i aplikaci pro smartphone či tablet.

Dalším návrhem do budoucna by byl i řízený modul do zásuvky, který bude propojen s wattroutrem prostřednictvím bezdrátové sítě. Tento modul by dostával informace o sepnutí či vypnutí přívodu energie od wattrouteru a zpět by zasílal informace o odběru připojeného zařízení. Díky modulu by bylo možné ušetřit práci s úpravou elektroinstalace. Pro spínání spotřebičů je jinak nutné, aby vedl fázový vodič od wattrouteru až ke spotřebiči.

5.1 Aplikace pro mobilní zařízení a PC

Pro ovládání wattrouteru prostřednictvím PC by bylo vhodné naprogramovat jednoduchou grafickou aplikaci, která by podporovala odesílání TCP paketů po síti nebo sériovou komunikaci. Aplikace by uživateli zobrazovala informace o protékajících proudech a napětí, informovala by o nastavení jednotlivých spotřebičů a o množství spotřebované energie. Pomocí aplikace by bylo možné provádět změny v nastavení a stáhnout z wattrouteru historii běhu zařízení, která je uložena na SD kartě.

5.2 Rozšíření efektivního řízení spotřeby

V rámci rozšíření možností řízení spotřeby by bylo vhodné implementovat také možnost zapnout spotřebič trvale po určitou dobu a pak ho uvést zpět do automatického režimu či trvalého vypnutí. Také by asi většina uživatelů wattrouteru ocenila možnost nastavení týdenního plánu, kde by bylo možné pro každý den nastavit jiný režim či jiné limity spotřeby.

Spotřebiče by také bylo možné spínat s proměnnou hodnotou jejich příkonu pomocí pulzně šířkové modulace. Takové spínání však není při střídavém napětí jednoduché jako při stejnosměrném, jelikož je nutné spínat a vypínat zátěž se začátkem periody napětí v síti. Pro spínání spotřebičů by bylo nutné využít jiné spínací prvky. Tato možnost spínání je vhodná pouze pro ohmickou zátěž, tedy bojler, akumulční kamna a podobné. Tato funkčnost by znamenala úplné spotřebování energie. Ve stávajícím stavu se spotřebič sepnou, až když výroba přesáhne spotřebu o hodnotu odběru spotřebiče. S funkcí proměnné hodnoty příkonu by se spotřebič mohl spínat při libolně velkém přebytku a jeho příkon by se snížil pomocí pulzně šířkové modulace.

Dále by bylo možné připojení teplotních čidel, která by vyhodnocovala teplotu vody v topném systému. Pokud by teplota vody v topném systému byla dostatečná, tak se může vypnout ohřev vody elektrickou energií v kombinovaném bojleru.

Závěr

Cílem práce bylo navrhnout a realizovat zařízení pro řízení elektrických spotřebičů. Zařízení mělo disponovat rozsáhlou mírou nastavení požadavků spínání spotřebičů v závislosti na denní době a výrobě FVE.

Vytvořené zařízení a jeho programová část jsou pouze prototypem. Zařízení umožňuje nastavit maximální dobu běhu zařízení během dne a možnost vybrat zda se může spotřebič zapínat i na nízký tarif pokud se doba běhu nenaplní z výroby FVE. K zařízení lze připojit až 8 nezávislých spotřebičů a do budoucna lze počet maximálně připojených nezávislých zařízení navýšit. Zařízení nepodporuje všechny definované příkazy a nemá připojena výstupní relé pro spínání spotřebičů.

V budoucnosti by bylo možné navrhnout zařízení s wifi modulem, které se zapojí mezi spotřebič a elektrickou zásuvku a díky síťové komunikaci bude podle pokynů řídicí jednotky spínat připojený spotřebič.

Literatura

- [1] SOLAR controls s.r.o.: *Přehled typů regulačních systémů WATTrouter [online]*. [cit. 2017-05-11]. Dostupné z: http://www.solarcontrols.cz/cz/watrouter_models.html
- [2] Arduino: *Arduino MEGA 2560 Genuino MEGA 2560 [online]*. [cit. 2017-05-11]. Dostupné z: <http://robodoupe.cz/2012/peckovy-tranzistor/>
- [3] Autodesk Inc.: *Eagle [online]*. [cit. 2017-05-11]. Dostupné z: <https://www.autodesk.com/products/eagle/overview>
- [4] *Fritzing [online]*. [cit. 2017-05-11]. Dostupné z: <http://fritzing.org>
- [5] SOLAR controls s.r.o.: *Základní popis funkce WATTrouteru [online]*. [cit. 2017-05-11]. Dostupné z: http://www.solarcontrols.cz/cz/watrouter_function.html
- [6] Punčochář, J.: *Operační zesilovače v elektronice*. BEN - technická literatura, 1996.
- [7] Bezstarosti, J.: *Péčkový tranzistor [online]*. RoboDoupě – web nejen o robotice, [cit. 2017-05-11]. Dostupné z: <https://www.arduino.cc/en/Main/arduinoBoardMega2560>
- [8] Brabander, F.: *Arduino LiquidCrystal I2C library [online]*. GitHub Inc., [cit. 2017-05-11]. Dostupné z: <https://github.com/fdebrabander/Arduino-LiquidCrystal-I2C-library>
- [9] Greiman, B.: *Arduino FAT16/FAT32 Library [online]*. GitHub Inc., [cit. 2017-05-11]. Dostupné z: <https://github.com/greiman/SdFat>
- [10] Arduino: *Arduino Standard Libraries Specification [online]*. [cit. 2017-05-11]. Dostupné z: <https://www.arduino.cc/en/reference/libraries>

LITERATURA

- [11] ITEAD: *ESP8266 Serial WIFI Module [online]*. [cit. 2017-05-11]. Dostupné z: https://www.itead.cc/wiki/ESP8266_Serial_WIFI_Module

Seznam použitých zkratk

FVE Fotovoltaická elektrárna

PC Osobní počítač

CR carriage return

LF line feed

Obsah přiloženého CD

	readme.txt.....	stručný popis obsahu CD
	src	
	impl.....	zdrojové kódy implementace
	thesis.....	zdrojová forma práce ve formátu $\text{L}^{\text{A}}\text{T}_{\text{E}}\text{X}$
	text	
	thesis.pdf.....	text práce ve formátu PDF