



Datalogger napětí, proudu a dalších veličin

Bakalářská práce

Studijní program:

B2612 Elektrotechnika a informatika

Studijní obor:

Elektronické informační a řídicí systémy

Autor práce:

Jiří Motl

Vedoucí práce:

Ing. Lukáš Krčmář

Ústav mechatroniky a technické informatiky

Konzultant práce:

Ing. Pavel Jandura, Ph.D.

Ústav mechatroniky a technické informatiky





Zadání bakalářské práce

Datalogger napětí, proudu a dalších veličin

Jméno a příjmení: Jiří Motl
Osobní číslo: M17000048
Studijní program: B2612 Elektrotechnika a informatika
Studijní obor: Elektronické informační a řídicí systémy
Zadávací katedra: Ústav mechatroniky a technické informatiky
Akademický rok: 2019/2020

Zásady pro vypracování:

1. Proveďte rešerši možných způsobů digitálního měření signálů, jako je napětí, proud, teplota, frekvence, a jejich záznamu. Rešerši proveďte také na možnosti přenosné profesionální zobrazovací techniky společně s bezdrátovým přenosem dat.
2. Zvolte vhodnou programovací platformu, proveďte návrh a testování zařízení, schopného zaznamenávat a zobrazovat měřená data, uvedená v bodu 1.
3. Na daném zařízení proveďte testy v laboratorních podmínkách.

Rozsah grafických prací:
Rozsah pracovní zprávy:
Forma zpracování práce:
Jazyk práce:

dle potřeby dokumentace
30–40 stran
tištěná/elektronická
Čeština



Seznam odborné literatury:

- [1] BANZI, Massimo a Michael SHILOH. Getting started with Arduino. Third edition. Sebastopol, CA: MakerMedia, [2015]. ISBN 1449363334.
- [2] ĎAĎO, Stanislav a Marcel KREIDL. Senzory a měřicí obvody. Praha: Vydavatelství ČVUT, 1996. ISBN 80-010-1500-9.

Vedoucí práce:

Ing. Lukáš Krčmář
Ústav mechatroniky a technické informatiky

Konzultant práce:

Ing. Pavel Jandura, Ph.D.
Ústav mechatroniky a technické informatiky

Datum zadání práce:

10. října 2019

Předpokládaný termín odevzdání:

18. května 2020

prof. Ing. Zdeněk Plíva, Ph.D.
děkan

L.S.

doc. Ing. Milan Kolář, CSc.
vedoucí ústavu

Prohlášení

Prohlašuji, že svou bakalářskou práci jsem vypracoval samostatně jako původní dílo s použitím uvedené literatury a na základě konzultací s vedoucím mé bakalářské práce a konzultantem.

Jsem si vědom toho, že na mou bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci nezasahuje do mých autorských práv užitím mé bakalářské práce pro vnitřní potřebu Technické univerzity v Liberci.

Užiji-li bakalářskou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti Technickou univerzitu v Liberci; v tomto případě má Technická univerzita v Liberci právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Současně čestně prohlašuji, že text elektronické podoby práce vložený do IS/STAG se shoduje s textem tištěné podoby práce.

Beru na vědomí, že má bakalářská práce bude zveřejněna Technickou univerzitou v Liberci v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů.

Jsem si vědom následků, které podle zákona o vysokých školách mohou vyplývat z porušení tohoto prohlášení.

26. května 2020

Jiří Motl

Poděkování

Chtěl bych poděkovat vedoucímu své bakalářské práce Ing. Lukášovi Krčmářovi a konzultantovi své bakalářské práce Ing. Pavlovi Jandurovi Ph.D., za odborné vedení práce a cenné rady, které mi pomohli při kompletaci.

Abstrakt

Bakalářská práce se zaměřuje na vývoj dataloggeru pro sběr dat během nabíjení a vybíjení baterií. Práce popisuje metody měření veličin jako napětí, proud, teplota a kmitočet. Popisuje také vývoj programu, konkrétně obsluhu jednotlivých modulů pro sběr dat. Tyto dílčí moduly jsou následně spojeny do výsledného dataloggeru pomocí jednodeskového počítače a dílčí programy pro obsluhu modulů jsou implementovány do výsledného programu pro sběr veškerých dat.

Klíčová slova

Raspberry, datalogger, měření elektrických veličin, měření teploty, Python

Abstract

The bachelor's thesis focuses on the development of a datalogger for data collection during battery charging and discharging. The work describes methods of measuring quantities such as voltage, current, temperature and frequency. It also describes the development of the program, specifically the operation of individual modules for data collection. These sub-modules are then connected to the final data logger by means of a single-board computer and the sub-programs for the operating modules are implemented in the final program for the data collection.

Keywords

Raspberry, datalogger, measurement of electrical quantities, measurement of temperature, Python

Obsah

Úvod	10
1. Teoretická část	11
1.1 Měření proudu.....	11
1.2 Dotykové měření teploty	13
1.3 Bezdotykové měření teploty	14
1.4 Měření frekvence.....	15
1.5 Měření napětí.....	16
1.6 A/D převodníky	16
1.7 Zobrazovací technika	19
1.8 Bezdrátový přenos dat	21
1.9 Programovací jazyky	22
1.10 Jednodeskové počítače	24
1.11 Komunikace	25
2. Praktická část.....	27
2.1 Moduly	27
2.2 Obsluha jednotlivých modulů pomocí jazyka Python	34
2.3 Uložení informací do souboru	37
2.4 Program	39
2.5 Měření	41
Závěr	45
Seznam použité literatury	46
Příloha A Schéma zapojení.....	48
Příloha B Datalogger.....	49

Seznam obrázků

Obrázek 1: Princip hallovy sondy [11].....	12
Obrázek 2: Paralelní A/D převodník [12].....	17
Obrázek 3: A/D převodník s postupnou aproximací [13].....	17
Obrázek 4: Integrační A/D převodník [14].....	18
Obrázek 5: Sigma-delta A/D převodník [15]	19
Obrázek 6: Komunikace po sběrnici SPI [16].....	26
Obrázek 7: Komunikace po sběrnici I2C [17].....	26
Obrázek 8: Raspberry Pi Zero W	28
Obrázek 9: Raspberry GPIO 40 pin header [18]	28
Obrázek 10: Stepdown měnič měnič	29
Obrázek 11: Závislost výstupního proudu na vstupním napětí	29
Obrázek 12: A/D převodník ADS1115	30
Obrázek 13: Hallova sonda	31
Obrázek 14: Sonda s teplotním čidlem.....	31
Obrázek 15: LCD displej.....	32
Obrázek 16: Mikrotlačítka pro ovládání.....	32
Obrázek 17: Spínač pro napájení.....	33
Obrázek 18: Konektory pro připojení teplotních čidel.....	33
Obrázek 19: Výkonové konektory XT90	34
Obrázek 20: Schéma zapojení	48
Obrázek 21: Zhotovený datalogger	49

Seznam zdrojových kódů

Zdrojový kód 1: Obsluha převodníku.....	34
Zdrojový kód 2: Obsluha teplotního čidla.....	35
Zdrojový kód 3: Obsluha LCD displeje	36
Zdrojový kód 4: Příklad obsluhy tlačítka pomocí přerušení	37
Zdrojový kód 5: Vytvoření složky a načtení nebo vytvoření souboru s nastavením	38
Zdrojový kód 6: Příklad vytvoření souboru formátu CSV	38

Seznam tabulek

Tabulka 1: Porovnání A/D převodníků	19
Tabulka 2: Verze IEEE 802.11	21
Tabulka 3: Verze Bluetooth.....	22
Tabulka 4: Porovnání Arduino a Raspberry Pi.....	25
Tabulka 5: Měření napětí.....	42
Tabulka 6: Měření proudu	43
Tabulka 7: Měření odběru při různém jasu	43
Tabulka 8: Měření odběru při různém napájení	44

Seznam zkratek

NTC	Negative Temperature Coefficient, záporný teplotní součinitel
PTC	Pozitive Temperature Coefficient, kladný teplotní součinitel
LED	Light Emitting Diode, elektroluminiscenční dioda
OLED	Organic Light Emitting Diode, organická elektroluminiscenční dioda
AMOLED	Active Matrix Organic Light Emitting Diode, organická elektroluminiscenční dioda s aktivní maticí
PMOLED	Passive Matrix Organic Light Emitting Diode, organická elektroluminiscenční dioda s pasivní maticí
TFT LCD	Thin Film Transistor Liquid Crystal Display, tenkofilmový tranzistorový displej s tekutými krystaly
AP	Access Point, přístupový bod
IEEE	Institute of Electrical and Electronics Engineers, Institut pro elektrotechnické a elektronické inženýrství
SPI	Serial Peripheral Interface, sériové periferní rozhraní
I2C	Inter-Integrated Circuit, sériová počítačová sběrnice
SDA	Serial Data, vodič přenášející datovou informaci
SCL	Serial Clock, vodič přenášející hodinovou informaci
HDMI	High Definition Multi-media Interface, multimediální rozhraní s vysokým rozlišením
GPIO	General Purpose Input/Output, univerzální vstupně výstupní pin
PWM	Pulse Width Modulation, pulzně-šířková modulace
CSV	Comma-Separated Values, hodnoty oddělené čárkami

Úvod

Cílem této bakalářské práce je vytvořit vhodné zařízení pomocí jednodeskového počítače Raspberry nebo Arduino pro sběr dat během nabíjení a vybíjení baterií. Výstupem z takového zařízení je tedy datový soubor vhodného formátu, který by měl obsahovat data o napětí, proudu a teplotě v časové linii.

Úvodní část se zabývá návrhem zařízení, zde je potřeba nejdříve zjistit informace o dostupných metodách pro měření jednotlivých veličin, a to metody pro měření proudu, napětí a teploty. Dále je třeba zjistit informace o dostupných jednodeskových počítačích a následně zvolit nejvhodnější počítač pro danou úlohu. Následující problematikou jsou možnosti komunikace mezi stolním počítačem a jednodeskovým počítačem pro bezdrátový přenos nasbíraných dat.

Další část práce je věnována jednotlivým modulům, které slouží pro sběr dat jednotlivých veličin. Zde je nutné vybrat takové moduly, aby plně vyhovovali zadání práce a je zde také kladen požadavek na velikost, aby výsledné zařízení bylo kompaktní, ale ne příliš velké. Poslední částí práce je vývoj obslužných programů pro jednotlivé moduly a vývoj hlavního programu pro vybraný jednodeskový počítač, do kterého budou zmíněné obslužné programy jednotlivých modulů implementovány. Tato část se zabývá nejen problematikou programu, ale i zvolením vhodného programovacího prostředí a vhodného programovacího jazyka.

Součástí praktické části je i měření na dataloggeru. Je potřeba zjistit s jakou přesností datalogger dokáže měřit napětí a proud. Součástí měření je i měření odběrového proudu, a to při různém podsvícení displeje a při různém napájecím napětí.

1. Teoretická část

Teoretická část této bakalářské práce se věnuje rešerši možných způsobů měření signálů. Jedná se o měření signálů jako napětí, proud, teplota a frekvence. Dále se teoretická část zaměřuje na možnosti zobrazovací techniky a bezdrátového přenosu dat včetně jedno-deskových počítačů.

1.1 Měření proudu

1.1.1 Nepřímé měření

Nepřímé měření elektrického proudu vychází z Ohmova zákona. Měřenou veličinou je zde elektrické napětí, které je pomocí Ohmova zákona převedeno na elektrický proud.

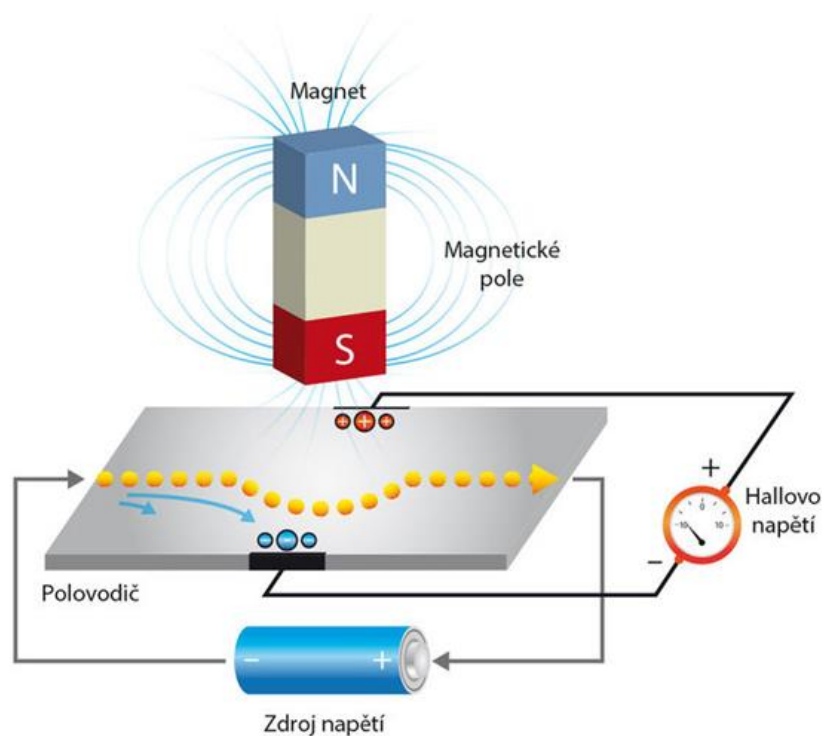
$$U = R * I \quad (1)$$

kde U je měřené napětí, které je rovno součinu procházejícího proudu I a odporu R . Používají se tři metody pro určení velikosti měřeného elektrického proudu. Metoda využití nízko-ohmového odporu neboli bočnicku, připojeného sériově se zátěží. Procházející proud tímto bočníkem vytvoří úbytek napětí, který odpovídá velikosti procházejícího proudu. Další metodou je využití Hallovy sondy, která měří magnetické pole, které je generováno průchodem elektrického proudu ve vodiči. Metodou poslední je proudový transformátor, který slouží převážně pro měření vyšších hodnot elektrických proudů.

1.1.2 Hallova sonda

Jedná se o elektronickou součástku a její princip činnosti je založen na tzv. Hallově jevu. Slouží zejména pro měření velkých stejnosměrných proudů. Hallova sonda je tvořena polovodičovou destičkou, kterou prochází proud viz obrázek 1. Po vložení této destičky do magnetického pole se přeskupuje náboj na jednu stranu destičky, a tak v této destičce vzniká napětí. Toto napětí je přímo úměrné velikosti procházejícího proudu I , magnetické indukci B a konstantě k , která zahrnuje tloušťku destičky a typ materiálu, z něhož je sonda vyrobena.

$$U = k * I * B \quad (2)$$



Obrázek 1: Princip hallovy sondy [11]

1.1.3 Proudový transformátor

Proudové transformátory jsou tvořeny primárním vinutím a sekundárním vinutím se samotným měřicím proudem. Slouží tedy pro měření velkých elektrických proudů, které jsou transformátorem transformovány na proudy nižší pro lepší měření. Od transformátorů napěťových se výrazně liší v počtu závitů primárního vinutí, jejichž počet je velmi nízký. V některých případech je primární vinutí tvořeno pouze vodičem procházejícím jádrem. Pro proudový transformátor platí, že poměr proudů tekoucích v jednotlivých vinutí je roven převrácenému poměru počtu závitů ve vinutí.

$$\frac{I_1}{I_2} = \frac{N_2}{N_1} \quad (3)$$

kde I_1 je proud procházející primárním vinutím, N_1 počet závitů primárního vinutí, I_2 proud procházející sekundárním vinutím a N_2 počet závitů sekundárního vinutí.

1.2 Dotykové měření teploty

Měření teploty je vždy možné pouze metodou nepřímou, nelze tedy snímat přímo teplotu. Teplotu tedy snímáme na základě fyzikálních jevů, při nichž známe hodnotu teploty. Snímání může probíhat bezdotykově nebo dotykově, přičemž dotykové teploměry většinou pracují s principem objemové roztažnosti látek vlivem teploty.

1.2.1 Dilatační teploměry

Vlivem teploty se mění objem látek, právě na tomto principu jsou založeny dilatační teploměry. Při tomto jevu se tedy měří objem délka nebo tlak a poté jsou tyto hodnoty převáděny na hodnotu teploty. Teploměry pracující na tomto principu mohou být skleněné, tyčové, bimetalové, tlakové a podobně.

1.2.2 Elektrické teploměry

Elektrické teploměry využívají závislosti elektrického odporu materiálu na teplotě, vlastní rezonanční frekvence nebo elektrického potenciálu v místě spoje dvou různých materiálů. U odporových snímačů teploty se jako základní materiál využívá platina, popřípadě nikl. U platiny je základním provedením Pt₁₀₀, který při 0 stupních Celsia má odpor roven 100 Ohmů. Nikl, se používá v základním provedení Ni₁₀₀₀, který při teplotě 0 stupňů Celsia má odpor roven 1000 Ohmů. Nikl má oproti platině větší teplotní součinitel, je tedy citlivější na změnu teploty, ale jeho charakteristika je více zakřivená než u platiny, proto nemusí být měření tak přesné. [4]

Dalšími snímači teploty v kategorii elektrických jsou snímače polovodičové. Tyto snímače opět využívají závislosti odporu na teplotě. Polovodičové teploměry se rozdělují na monokrystalické s různým počtem PN přechodů a polykrystalické neboli termistory NTC a PTC. Monokrystalické snímače pracují na principu změny charakteristiky polovodičového prvku vlivem teploty. Prvky termistorové ale využívají změny měrného odporu. Termistory PTC mají pozitivní teplotní koeficient a vyznačují se při určité teplotě prudkým nárůstem odporu, využívají se tedy spíše jako regulační, respektive spínací prvky než pro přesné měření okolní teploty. Výjimkou PTC je lehce dopovaný křemík, ve kterém bylo nalezeno velmi lineární chování oproti běžným PTC termistorům. Termistory NTC, které mají

negativní teplotní koeficient, se vyznačují malými rozměry a velkou hodnotou teplotního součinitele ovšem nevýhodou je značná nelinearita a běžně se používají pro užší teplotní rozsahy.

Posledním typem elektrických snímačů jsou snímače termo elektrické, které jsou založeny na Seebeckově jevu. Při tomto jevu dochází k převodu tepelné energie na energii elektrickou. Termo elektrické napětí je poté dáno rozdílem potenciálů. Pro výrobu termočlánků se vždy využívá dvojice materiálů tvořící článek vykazující nárůst termoelektrického napětí s nárůstem teploty. Tato dvojice materiálů se volí dle požadavku měřeného rozsahu. Pro střední rozsahy teplot se využívá odlišná dvojice nežli pro teploty vysoké. Závislost těchto snímačů je v řádu jednotek až desítek mikrovoltů na jeden stupeň Celsia a u polovodičů může přesahovat hodnotu vyšší než 100 mikrovoltů na jeden stupeň Celsia. Jelikož při takto nízkých hodnotách napětí mohou mít rušivé signály velký vliv, tak je nutné klást požadavky na měření tohoto termoelektrického napětí. Požaduje se tedy minimalizace vlivu odporu přívodů a potlačení rušivých signálů, většinou jsou tedy využity různé kompenzační obvody.

1.3 Bezdotykové měření teploty

Měření bezdotykové funguje na principu vyzařování elektromagnetického záření, jelikož každé těleso vyzařuje toto záření v závislosti na teplotě. V oblasti viditelného záření teplotě odpovídá právě barva daného záření, to ale platí pouze pro černé těleso. Největším problémem u této metody je emisivita, jelikož těleso nejen vyzařuje, ale také odráží nebo propouští světlo, a to se v tomto případě jeví jako rušení.

1.3.1 Pyrometry

Pyrometry jsou jasové, pásmové a zobrazující a měří objektivně nebo subjektivně, to znamená, srovnávají. Pyrometry srovnávací srovnávají přicházející záření s pyrometrickou žárovkou. Například pyrometr pásmový srovnávací obsahující různé filtry je vhodný pro teplotu pomalu se měnící a pyrometr jasový srovnávací naopak má velkou výhodu, že dokáže velmi rychle reagovat na změnu teploty i u objektů méně rozměrných.

1.3.2 Infračervené bezdotykové teploměry

Tyto teploměry využívají faktu, že z povrchu těles září infračervené záření. Disponují možností velmi rychlého měření a možností sledování změn, základem těchto snímačů je detektor infračerveného záření s optikou.

1.3.3 Termovize

Termovize též snímají záření infračervené, které je vyzařováno z povrchu těles. U tohoto snímače, který disponuje monitorem, je přenášén obraz ze speciální termokamery. Rozložení teplot zobrazovaných na monitoru je dáno termogramem. Detektory v kameře mohou být kvantové, bolometrické nebo s postupným rozkladem obrazu.

1.4 Měření frekvence

Frekvence je jednou z nejpřesněji měřitelných veličin. Je svázána s jednotkou času. Určuje periodicitu jevu. Jednotka frekvence, respektive kmitočtu je Hertz, zkráceně Hz. Pojednává o počtu period za jednu sekundu. Měřiče frekvence pracují na různých principech, metody měření pak lze dělit na analogové a digitální. Analogové metody převážně využívají rezonančních obvodů, ale v dnešní době už jsou spíše na ústupu a nahrazují je metody digitální. Jednou z digitálních metod je čítání průchodu napěťovou úrovní. Například čítač čítá průchody nulou, čítání trvá určitou dobu a frekvence je poté vypočítána z počtu načítaných průchodů a doby čítání. Největším problémem této metody je signálový šum. Při průchodu úrovní může signál kvůli šumu projít hned několikrát a výsledná frekvence je poté tímto šumem zkreslena. U takovéto metody je tedy vhodné využít určité toleranční pásmo, pokud signál alespoň jednou neopustí toleranční pásmo v opačné polaritě, tak je čítání zastaveno a eliminuje se tak vliv šumu na samotné měření frekvence. Volba šířky pásma už poté závisí na konkrétní aplikaci a na konkrétním signálu. [3],[5]

1.5 Měření napětí

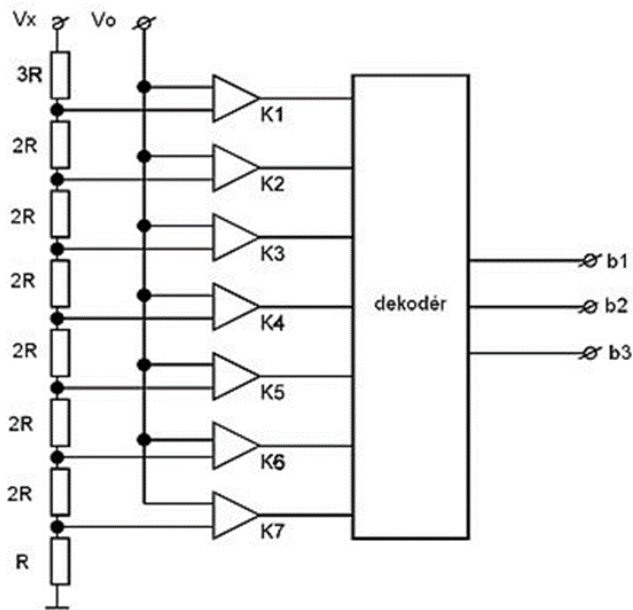
Měření napětí pomocí jednodeskových počítačů je nejdůležitější částí, jelikož většina měřených signálů, jak elektrických veličin, tak neelektrických veličin, je převáděna právě na elektrické napětí, a to je dále zpracováno. Nejjednodušší metodou měření napětí pomocí jednodeskového počítače je přímo využít analogový vstup počítače, nevýhodou je, že tyto vstupy obsahují většinou méně přesné analogově digitální převodníky, proto je vhodnější využít samostatný obvod s více přesným analogově digitálním převodníkem a naměřené napětí získávat pomocí komunikace právě s tímto převodníkem.

1.6 A/D převodníky

A/D převodníky převádějí vstupní analogový signál, nejčastěji napětí, na výstupní signál digitální. Celý převod se vykonává ve dvou krocích, nejprve se signál vzorkuje s určitou frekvencí, kde tato frekvence musí být minimálně dva krát větší než maximální frekvence vstupního signálu, aby byl splněn Nyquistův teorém a poté jsou tyto vzorky v druhém kroku kvantovány, to znamená, že každému vzorku je přidělena binární hodnota. Mezi nejvýznamnější parametry patří právě frekvence vzorkování a rychlost převodu. Maximální možná frekvence vzorkování právě vyplývá z rychlosti převodu. Dalším důležitým parametrem je kvantizační krok. Kvantizační krok Q lze definovat jako maximální možnou změnu vstupního signálu při kterém se binární hodnota signálu změní o jeden bit. Maximální možná chyba je tedy rovna \pm polovině kvantizačního kroku. [1]

1.6.1 Paralelní A/D převodníky

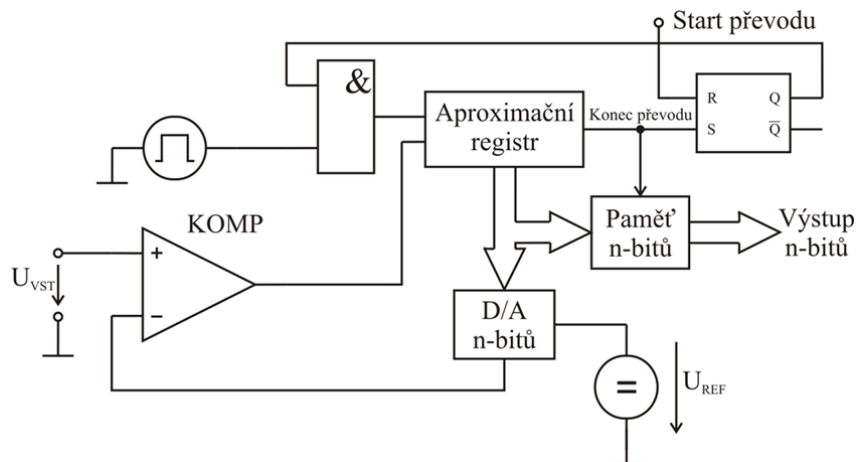
Nejjednodušší a zároveň nejrychlejší metodou jsou právě paralelní A/D převodníky, kde je napětí přiváděno současně na vstupy paralelních komparátorů a na každém komparátoru je také přivedeno referenční napětí. Toto referenční napětí je na každém komparátoru jiné a je dáno odporovou sítí. Výstup komparátorů je přiveden do dekodéru, který hodnoty výstupů komparátorů dekoduje na binární datové slovo. Nevýhodou této metody je potřebný počet komparátorů, jelikož se tento počet rovná $2^n - 1$, kde n je počet bitů výstupního slova neboli rozlišení převodníku.



Obrázek 2: Paralelní A/D převodník [12]

1.6.2 A/D převodníky s postupnou aproximací

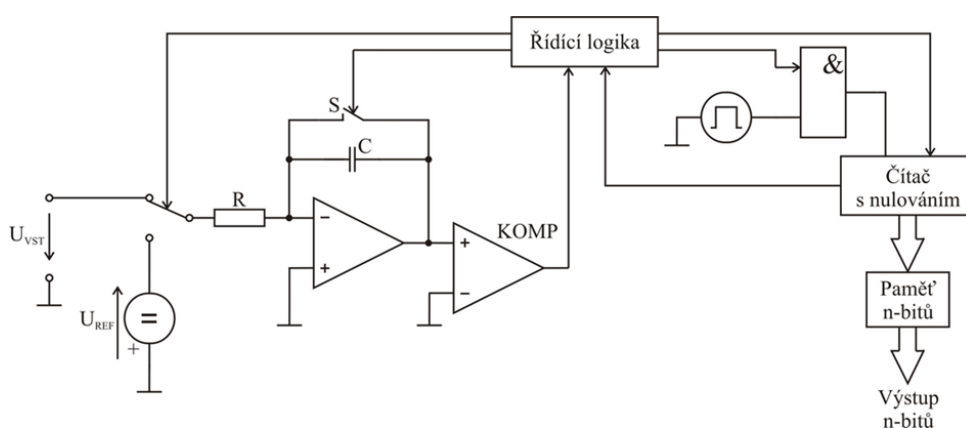
U převodníků s postupnou aproximací je převod vykonáván po krocích, kde napětí vstupní je porovnáváno s referenčním napětím a toto referenční napětí se mění s každým krokem. Počet bitů výstupního binární kódu je roven počtu vykonaných kroků, protože v každém kroku se získává právě jeden bit. U tohoto převodu je důležité, aby během kroků převodu bylo napětí na vstupu pořád ve stejné hodnotě, proto se využívá vzorkovacího obvodu Sample & Hold, který hodnotu vstupu „podrží“. Výhodou tohoto převodníku je, že k převodu stačí pouze jeden komparátor, ale nevýhodou je rychlost převodu.



Obrázek 3: A/D převodník s postupnou aproximací [13]

1.6.3 Integrační A/D převodníky

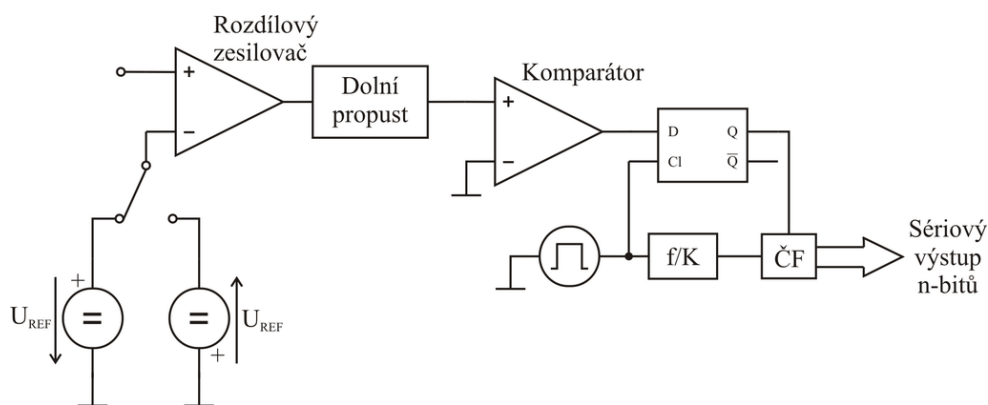
Oproti ostatním převodníkům je tento odlišný, protože neměří napětí přímo, ale měří čas v závislosti na vstupním napětí. V principu s dvojitou integrací se vstupní napětí přivede na integrátor, který toto napětí integruje, po uplynutí určité doby překlopí klopný obvod a integrátor začne integrovat záporné referenční napětí, v tuto chvíli se začne měřit čas a integrované napětí začne klesat k nule, jakmile napětí klesne k nule, tak naměřený čas vypovídá, o jakou hodnotu napětí na vstupu se jedná. Jelikož je na převod potřebný určitý čas pro vybíjení a nabíjení kondenzátoru, tak tento převodník patří mezi pomalejší s malou rychlostí převodu, ovšem dosahuje velice vysokých přesností.



Obrázek 4: Integrační A/D převodník [14]

1.6.4 A/D převodníky sigma-delta

Tento typ převodníku se používá především v audio technice a měřicí technice pro svoji schopnost velmi přesného měření. Základem je opět integrátor a komparátor, který generuje sled pulzů, střední hodnota těchto pulzů v určitém intervalu poté odpovídá hodnotě vstupního napětí.



Obrázek 5: Sigma-delta A/D převodník [15]

Tabulka 1: Porovnání A/D převodníků

Typ	Rozlišení [bit]	Rychlost převodu [Hz]
Paralelní	6 - 12	$10^7 - 10^{10}$
Aproximační	8 - 18	$3 \cdot 10^4 - 10^8$
Integrační	10 - 27	$10^{-1} - 10^3$
Sigma-delta	16 - 24	$10^1 - 10^7$

1.7 Zobrazovací technika

1.7.1 OLED

OLED displej je typ LED displeje, kde se jako elektroluminiscenční látky využívají látky organické. Tyto organické látky jsou uloženy mezi dvě elektrody a aby mohlo dojít k šíření světla do okolí, musí být alespoň jedna z těchto elektrod průhledná. Většinou se využívá několik vrstev organické látky mezi neprůhlednou katodou a průhlednou anodou. V principu jsou vrstvy organické látky čtyři, a to vrstvy přenášející díry, vypuzující díry, přenášející elektrony a vrstva vyzařovací. Po přivedení napětí na elektrody jsou vyvolané kladné a záporné náboje spojovány ve vyzařovací vrstvě a následně produkují světlo.

Pro OLED displej existují dvě základní metody, metoda PMOLED neboli OLED s pasivní maticí a AMOLED neboli OLED s aktivní maticí. Jednodušší metoda PMOLED matice mřížkové, kdy jednotlivé spoje mřížky tvoří jednotlivé pixely. Signály jsou poté přiváděny v cyklu po jednom řádku. Úroveň jasu je dána velikostí přiváděného proudu na jeden pixel. Pro 10 řádkový displej a pro dosažení jasu 100 nitů je nutné, aby byly pixely

buzeny na úroveň jasu 1000 nitů po 1/10 snímkovacího času. Příčinou těchto proudových impulzů je ale nižší účinnost displeje. Proto jsou tyto displeje vhodné především pro malé úhlopříčky a pro zobrazování statických textových informací. Druhou metodu s aktivní maticí využívají náročnější grafické aplikace s vyšším rozlišením. V této metodě má každý pixel svůj vlastní tranzistor. Výhodou oproti PMOLED je zde vyšší zobrazovací frekvence a nižší spotřeba, nevýhodou je pak složitější struktura a vyšší pořizovací cena.

1.7.2 LCD

LCD displeje jsou displeje z tekutých krystalů. Je tvořen barevnými nebo monochromatickými pixely, které se nacházejí před zdrojem světla. Každý pixel je tvořen tekutými krystaly, které jsou osazeny mezi dvě průhledné elektrody a dva polarizační filtry. Osy polarizačních filtrů jsou proti sobě posunuty o 90 stupňů. Pokud není tedy světlo při průchodu mezi filtry nijak polarizováno, tak filtry neprojde. Molekuly mezi elektrody a filtry jsou bez vnějšího elektrického zdroje srovnány do spirálové struktury a při průchodu světla touto strukturou se polarizace stáčí o 90 stupňů a následně světlo projde i druhým filtrem. Jelikož mají filtry polarizaci pouze ve dvou na sebe kolmých směrech tak průchodem nepolarizovaného světla touto soustavou se vždy polovina absorbuje.

Pokud je na elektrody přivedeno napětí z vnějšího elektrického zdroje, tak vzniká elektrické pole a molekuly krystalů jsou staženy k sobě rovnoběžně ve směru vzniknutého pole. Při průchodu prvním filtrem je tedy světlo polarizováno do kolmého směru oproti filtru druhému a průchod molekulami ho nijak neovlivní, tudíž světlo druhým filtrem zcela neprojde. Toto uspořádání je pro LCD displeje obvyklé, avšak někdy se může využívat uspořádání molekul při kterém je soustava při přivedení elektrického napětí průhledná a naopak, bez přivedení napětí veškeré světlo blokuje.

U barevných LCD displejů je každý pixel tvořen třemi subpixely. Každý z těchto subpixelů vyzařuje pouze jednu barvu a to červenou, modrou a zelenou. Pomocí tranzistorů lze kontrolovat svítivost každého ze subpixelů a pomocí kombinací těchto tří barev lze dosáhnout až několik milionů různých barev.

1.8 Bezdrátový přenos dat

1.8.1 Wi-Fi

Wi-Fi je označení pro standardy IEEE 802.11. Tyto standardy popisují bezdrátovou komunikaci v počítačových sítích. Přednostmi standardu IEEE 802.11 je pokrytí plochy a podpora mobility, záporům je poté rychlost a větší chybovost oproti drátové komunikaci. Architektura sítě je tvořena buňkami neboli skupinou stanic. Stanice mohou komunikovat navzájem, ale nejčastějším řešením je komunikace řízená základnovou stanicí Access Point, zkráceně AP. AP řídí veškerou komunikaci, ukládá rámce, vysílá pravidelně beacon frame, který obstarává synchronizaci času, vyzývá nové stanice ke vstupu do buňky a nese systémové parametry.

Realizace bezdrátového přenosu je potom standardizována pomocí infračerveného světla nebo pomocí mikrovln. Infračervené světlo se ale nikdy nevyrobělo, protože nese problém s přímo viditelností. Realizace je tedy pomocí mikrovln, a to v bezlicenčních pásmech 2,4 Ghz a 5 Ghz. Nižší frekvence má lepší propustnost, ale menší šířku pásma oproti frekvenci vyšší. Verzí bezdrátových komunikací je hned několik a liší se využitým pásmem a rychlostí přenosu.

Tabulka 2: Verze IEEE 802.11

Verze IEEE 802.11	Rok vydání	Využití pásma	Maximální rychlost
802.11a	1999	5 Ghz	54 Mb/s
802.11b	1999	2,4 Ghz	11 Mb/s
802.11g	2003	2,4 Ghz	54 Mb/s
802.11h	2003	5 Ghz	54 MB/s
802.11n	2009	2,4 Ghz / 5 Ghz	100 Mb/s
802.11ac	2013	5 Ghz	1 linka až 867 Mb/s
802.11ad (WiGig)	2012	60 Ghz	7 Gb/s

1.8.2 Bluetooth

Bluetooth je technologický standard umožňující bezdrátový přenos dat. Byl vyvinut firmou Ericsson, aby sloužil pro bezdrátovou komunikaci mezi dvěma a více zařízeními. Verze

Bluetooth 1.0 vyšlo už v roce 1998 a umožňovala bezdrátový přenos rychlostí až 721 kb/s. Pro porovnání prozatím nejnovější verzí je Bluetooth 5.2 kde lze volit mezi rychlostmi a dle rychlosti je pot dán dosah s tím, že při nižší rychlosti větší dosah. Aktuálně nejrozšířenější verzí je Bluetooth 4.2, která se nachází ve většině zařízení s Bluetooth. Rádiová frekvence, na které Bluetooth pracuje je stejná jako u Wi-Fi, jedná se o 2,4 Ghz. Aby bylo dosaženo vysoké stability přenosu dat, tak technologie pracuje s 79 radiovými kanály, které jsou od sebe vzdáleny šířkou 1 Mhz a Bluetooth poté mezi těmito kanály přeskakuje až 1600krát za sekundu. [2]

Tabulka 3: Verze Bluetooth

Verze	Vydání	Rychlost	Dosah venku	Dosah uvnitř
2.0	2007	2,2 Mb/s	standartně 10 m	
3.0	2009	24 Mb/s	standartně 10 m	
4.0	2010	1 Mb/s	50 m	10 m
5.0	2016	až 2Mb/s	až 240 m	40 m
5.1	2019	až 2Mb/s	až 240 m	40 m
5.2	2020	až 2Mb/s	až 240 m	40 m

1.9 Programovací jazyky

1.9.1 C#

C# neboli C Sharp je vysokoúrovňový objektově orientovaný programovací jazyk, který byl vyvinut společností Microsoft. Tento programovací jazyk je založen na programovacích jazycích C++ a Java. Programovací jazyk C# je využíván pro tvorbu formulářových, grafických, databázových nebo webových aplikací.

V tomto programovacím jazyce se nejčastěji objevují objekty a instance třídy. V informatice je objekt označení pro datovou strukturu nebo proměnnou, která je v paměti vytvořena a má tak přidělené své místo a má nějakou hodnotu. Objekt je v objektově orientovaném programování instancí třídy, myšleno nějaký konkrétní datový objekt, který byl odvozen od nějaké konkrétní třídy neboli vzoru.

Každá proměnná musí mít vždy svůj datový typ. Může to být text, znak, číslo a podobně, v podstatě záleží, k čemu je proměnná využívána, dle toho se jí přiřazuje datový typ. Jelikož programovací jazyk C# je statický typový systém, je nutné, aby každá proměnná vyskytující se v programu byla nejdříve deklarována, a to včetně datového typu. Datové typy v programovacím jazyce C# jsou celočíselné Sbyte, Byte, Short, Ushort, Int, UInt, Long, Ulong, desetinné Float a Double, ostatní hodnotové Char, Decimal a Bool a v neposlední řadě Referenční. Referenční typy jsou odlišné od hodnotových datových typů, jelikož sami nenesou hodnotu ale pouze referenci, která ukazuje na hodnotu uloženou v paměti. Proto dva různé referenční datové typy mohou ukazovat na stejnou hodnotu, pokud pomocí první reference hodnotu upravíte, druhým referenčním type uvidíte již upravenou hodnotu. Navíc referenční datové typy mohou s sebou nést metody. Typickým příkladem je datový typ „string“. String je řetězec charakterů a má právě své metody jako například „Replace“.

V objektově orientovaném programování je základním konstrukčním prvkem třída. Třída slouží jako předpis pro generování objektu jako instance. Mnohdy odpovídá pojmům z reálného světa, typickým příkladem je třída „Zákazník“. Třída definuje atributy (Jméno, Adresa, Objednávky a podobně) a metody neboli funkce (Vložit do košíku, Zaplatit, Změnit Adresu atd.). Objektů typu třída Zákazník může vzniknout několik, ale každý tento objekt má vlastní hodnoty atributů a na každý tento objekt lze zavolat metodu definovanou v třídě Zákazník.

Sada hodnot, nejčastěji sada hodnot objektů dané třídy je datový typ kolekce. Cílem kolekce je právě sloužit jako úložiště objektů. Například kolekce „Seznam zaměstnanců“ může obsahovat všechny objekty typu „Zaměstnanec“. Poté lze přistupovat ke každému zaměstnanci z kolekce zvlášť. Kolekce má výhodu oproti poli v tom, že jeho velikost není striktně daná, lze tak do ní přidat libovolné množství objektů a není potřeba předem definovat velikost.

1.9.2 Python

Python je vysokoúrovňový programovací jazyk, jedná se o jazyk skriptovací. Podporuje programování objektové včetně procedurálního, imperativního i funkcionálního programování. Objektové programování je obdobné jako u programovacího jazyka C#, popsáno v kapitole 1.9.1. Jazyk Python je vyvíjen jako open source, takže je nabízen zdarma pro

většinu platforem jako například Windows, Unix, macOS, Android a podobně. Postupným vývojem již vznikly tři verze, Python 1, Python 2 a Python 3. Python 1 se již nepoužívá, jeho poslední verze byla vydána v roce 2000, Python 2 je již v útlumu, poslední verze bude pravděpodobně vydána v roce 2020. Naopak Python 3 je velmi aktivní verze, která byla prvně vydána v roce 2008 a stále je ve vývoji.

Mezi důležité vlastnosti programovacího jazyka Python patří jeho dynamika. Má velké možnosti a programátor si tak při tvorbě programu může vybírat z několika paradigmat, které program nabízí ačkoli některé jen v omezené míře. Oproti ostatním programovacím jazykům je kód programu v jazyce Python krátký a velmi dobře čitelný. Proto další významnou vlastností je jeho jednoduchost, zejména z hlediska učení. Proto je tento jazyk vhodný jak pro začátečníky, tak pro tvorbu rozsáhlých programů, které jsou díky jednoduchosti velice produktivní.

Mezi hlavní datové typy pro programovací jazyk Python patří boolean, který může nabývat hodnot true nebo false, pak čísla, která mohou být celá (integer), reálná (float), zlomky (fraction) anebo čísla komplexní. Dalším datovým typem jsou řetězce což jsou posloupnosti znaků Unicode. Mezi ostatní hlavní datové typy patří seznamy, n-tice, množiny a slovníky jenž vždy reprezentují posloupnosti nebo kolekce. [6],[7]

1.10 Jednodeskové počítače

1.10.1 Arduino

Arduino je jednodeskový počítač založený na mikrokontrolerech ATmega. Tyto mikrokontrolery jsou od firmy Atmel a mají 8bitovou architekturu. Arduino samotné je tedy otevřená elektronická platforma do které je nahráván uživatelem vytvořený program v programovacím jazyce Arduino, který je založený na základě jazyka Wiring. Nahrávání programu do mikrokontroléru je pomocí Arduino software, který vychází z prostředí Processing. Arduino se vyznačuje především nízkou nákladovostí, multiplatformou a jednoduchým a čistým programovacím prostředím.

Dále pro Arduino existují takzvané Arduino shields. Jedná se o moduly rozšiřující základní desku. Pro připojení Arduina a přídatného modulu je nutné využít komunikaci přes

SPI, I²C nebo sériovou linku. Nejznámějšími Arduino shiedly jsou Ethernet shield, Wifi shield, Motor shield a GPS shield.

1.10.2 Raspberry

Raspberry je malý jednodeskový mikropočítač vyvinutý nadací Raspberry Pi Foundation pro podpoření výuky informatiky. Architektura raspberry je 32bitová nebo 64bitová. Oproti arduinu se opravdu jedná o jednodeskový počítač srovnatelný se stolním počítačem. K raspberry je možné připojit periferie jako je monitor, klávesnice a myš. Raspberry bylo vyvinuto již několik generací a modelů a liší se výkonem a možnostmi aplikací. Pro vývoj programu pro raspberry není nutné využívat jiné vývojové prostředí a je možné program vytvářet v samotném raspberry. Na raspberry je možné provozovat Microsoft Windows, RISC OS i Linux.

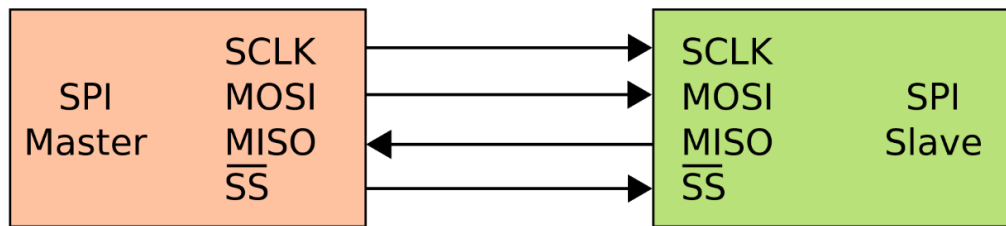
Tabulka 4: Porovnání Arduino a Raspberry Pi

Typ	Frekvence	Flash paměť	Digital I/O	Analog I/O
Arduino	Až 400 Mhz	16 - 256 kB	14-54	6-12
Raspberry	Až 1,5 Ghz	Až 4 GB	12-40	neobsahuje

1.11 Komunikace

1.11.1 SPI

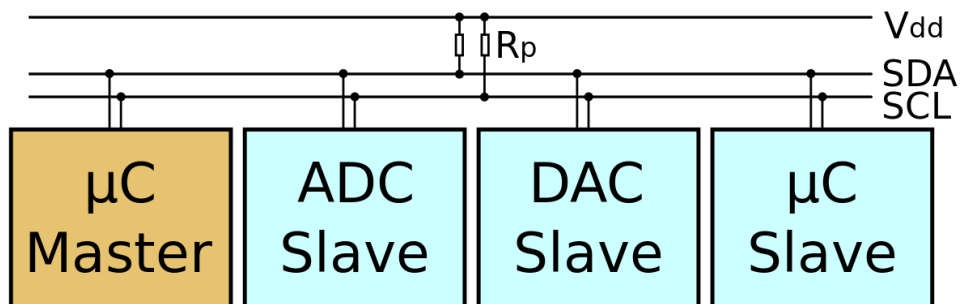
Serial Peripheral Interface neboli sériové periferní rozhraní, zkráceně SPI viz obrázek 6. SPI se používá pro komunikaci mezi mikroprocesory a ostatními periferiemi jako displeje, převodníky a podobně. Společná sběrnice realizuje konkrétní komunikaci mezi zařízeními. Adresace zařízení se provádí pomocí adresačních vodičů. Master řídí veškerou komunikaci na sběrnici, vysílá hodinové signály na SCLK a určuje se kterým zařízením bude komunikovat. Pro toto určení slouží vodič SS – Slave Select nebo CS – Chip Select. Slave vysílá je-li zvolen pomocí SS nebo CS dle hodinového signálu. Komunikace poté probíhá po vodičích MOSI a MISO, kde MOSI znamená Master Out, Slave In a MISO znamená Master In, Slave Out. [9]



Obrázek 6: Komunikace po sběrnici SPI [16]

1.11.2 I2C

I²C, anglicky Inter-Integrated Circuit je multi-masterová sběrnice viz obrázek 7. Slouží pro komunikaci mezi mikroprocesorem a periferiemi. Výhodou této sběrnice je, že veškerý obousměrný tok dat probíhá pouze po dvou vodičích. Vodiče jsou označeny jako SDA - serial data a SCL – serial clock. Na jednu sběrnici může být připojeno několik zařízení a každé má přiděleno sedmibitovou adresu, to umožňuje připojit až 128 zařízení na jednu sběrnici. Reálný počet zařízení na sběrnici je ale nižší, protože některé adresy jsou rezervovány pro zvláštní účely. Co se týče rychlosti přenosu, základní rychlostí je 100 kbps a některé vylepšené verze sběrnice mohou dosahovat až 3,4 Mbps. [8],[10]



Obrázek 7: Komunikace po sběrnici I2C [17]

2. Praktická část

Praktická část této práce je věnována zhotovení přípravku pro měření proudu, napětí a teploty. Pro tvorbu existuje řada jednodeskových počítačů jako například Raspberry Pi nebo Arduino. Konkrétně tento případ zhotovení se bude věnovat vývoji pro jednodeskový počítač Raspberry Pi. Raspberry je oproti arduinu většinou rychlejší a obsahuje i bezdrátové technologie jako W-Fi a Bluetooth, pro bezdrátový přenos tudíž nebude potřeba využít dodatečný modul. Naopak nevýhodou oproti arduinu jsou vstupní periferie. Raspberry obsahuje pouze digitální vstupy, tudíž není možné připojit analogové vstupní periferie na přímo, ale je nutné využít analogově digitální převodník, který bude zpracovávat analogový signál a informace o tomto signálu bude digitálně přenášet do raspberry.

2.1 Moduly

Zhotovený přípravek se bude skládat z několika modulů. Zakladním modulem bude jednodeskový počítač Raspberry Pi, který bude zpracovávat informace od ostatních periférií a následně je ukládat do souboru. Aby mohl počítač být napájen je třeba využít měniče napětí. Konkrétně stepdown měnič, který bude napětí měřeného zdroje až 70 voltů měnit na napětí 5 voltů pro napájení raspberry a periférií. Pro měření napětí a proudu bude sloužit analogově digitální převodník, odporový dělič a Hallova sonda. Pro měření teploty bude využito sondy s teplotním čidlem.

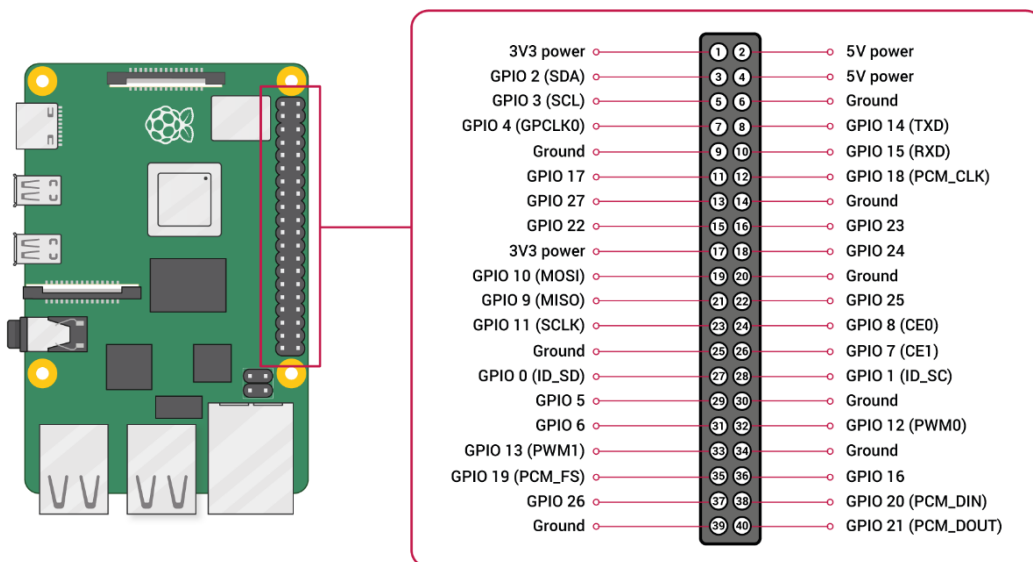
Dále pro zobrazení aktuálních hodnot a také nastavení bude využito LCD displeje a tlačítek. Poslední částí je přepínač a konektory. Pro měření se zdroji s napětím menším než 7 voltů je nutné externí napájení a jako volba mezi napájením ze zdroje a externím napájením bude sloužit právě zmíněný přepínač. Konektory jsou zde nutné pro připojení baterie, respektive zdroje a zátěže. Na konektory je kladen velký požadavek, jelikož zde budou téci proudy o hodnotě až 50 ampér. Konektory zde jsou ještě potřeba dva tří pinové pro připojení externích čidel teploty. Posledním důležitým požadavkem je celková kompaktnost přípravku proto při volení jednotlivých modulů je vždy nutné dbát právě na velikost modulu.

2.1.1 Raspberry Pi Zero W

Raspberry Pi Zero W viz obrázek 8, je mini počítač, respektive jednodeskový počítač s jednojádrovým procesorem z řady ARM, konkrétně Broadcom BCM2835 o frekvenci 1 GHz a sběrnici o šířce 64 bitů. Počítač obsahuje dále operační paměť o velikosti 512 MB. Ohledně bezdrátových technologií je zde Wi-Fi 802.11b/g/n a Bluetooth ve verzi 4.0. Pro připojení periférií deska obsahuje dvakrát microUSB port, jeden miniHDMI port a jeden microSD slot pro připojení externí paměti. Pro připojení digitálních periférií je zde ještě GPIO header 40pin viz obrázek 9. Tento GPIO header obsahuje napájecí piny, piny pro zemnění, komunikační piny a digitální vstupní a výstupní piny



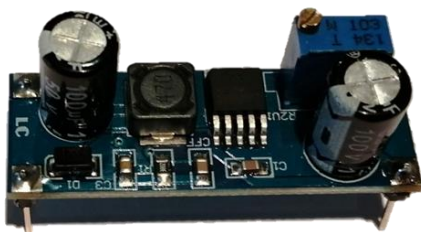
Obrázek 8: Raspberry Pi Zero W



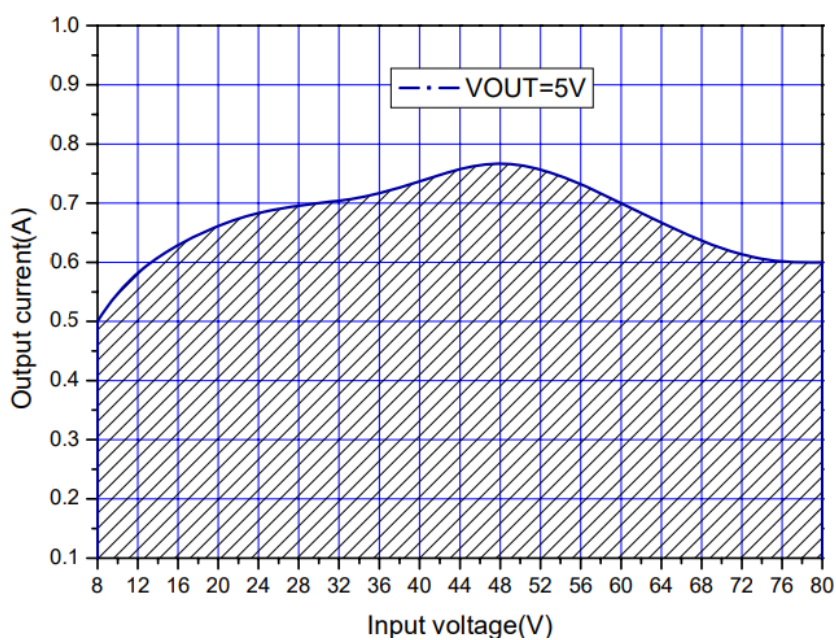
Obrázek 9: Raspberry GPIO 40 pin header [18]

2.1.2 Stepdown měnič

Jako stepdown měnič je využit měnič XL-7015 v2 viz obrázek 10, jenž svými parametry vyhovuje právě této práci. Rozsah pracovního napětí je zde od 5 V až po 80 V. Výstup lze poté regulovat od 1,25 V až po 20 V. V tomto případě bude nastaven výstup konkrétně na 5 V pro napájení raspberry a periferií. Minimální Drop Out napětí, tedy minimální rozdíl mezi výstupním napětím a vstupním napětím je 2 V, tudíž je možné připojit zdroje od napětí 7 V, aby zařízení pracovalo správně, pro nižší zdroje bude nutno využít externí napájení. Dalším parametrem měniče je výstupní proud. Dle charakteristiky odběru proudu v závislosti na vstupním napětí viz obrázek 11 je možno odebírat proud od 0,5 A až do 0,75 A což je dostatečný proud, protože odběrový proud Raspberry Pi Zero W a periferií je maximálně 350 mA. Maximální příkon je tedy 1,75W při zátěži.



Obrázek 10: Stepdown měnič měnič



Obrázek 11: Závislost výstupního proudu na vstupním napětí

2.1.3 A/D převodník

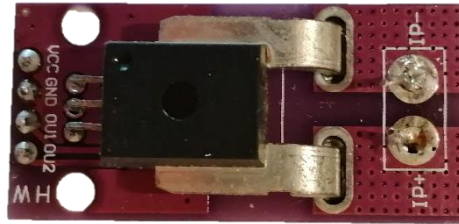
Jako analogově digitální převodník je zvolen převodník ADS1115 viz obrázek 12. Jedná se o šestnáctibitový převodník se čtyřmi kanály a programovatelným zesílením. Vstupní napětí kanálu může být od $\pm 0,256$ V až po $\pm 6,144$ V dle zvoleného zesílení. Rychlost vzorkování je také volitelná, a to od 8 sps až po 860 sps. Tento převodník bude zpracovávat analogový signál z odporového děliče a bude tak sloužit pro měření napětí. Další z kanálů bude připojen na výstup hallové sondy, která měřený proud převádí na napětí. Změřené hodnoty analogových signálů z děliče a ze sondy jsou dále digitálně přeneseny do raspberry. Pro přenesení dat mezi převodníkem a raspbbery slouží sběrnice I²C.



Obrázek 12: A/D převodník ADS1115

2.1.4 Hallova sonda

Jelikož jedním z požadavků je měřit proud až do 50 A, je zde zvolena pro měření proudu hallova sonda, konkrétně ACS770LCB-050B viz obrázek 13. Jedná se o sondu od firmy Allegro. Tento konkrétní typ umožňuje měření proudu právě až do 50 A. Měřený proud je lineárně převáděn na elektrické napětí, a to s citlivostí 40 mV na jednu ampéru. Napájení sondy může být od 4,5 V do 5,5 V, proto zde bude opět sonda napájena z Raspberry z pinu 5V. Výstup sondy bude dále zapojen do AD převodníku pro změření napětí na výstupu. Dalším parametrem sondy je offset napětí. U této verze sondy, která je bilineární tedy měření proudu do kladných i do záporných hodnot je offset V_{OE} dán napájecím napětím V_{CC} , konkrétně $V_{OE} = V_{CC}/2$. Při měření napětí a následném zpracování je tedy nutné s tímto parametrem počítat.



Obrázek 13: Hallova sonda

2.1.5 Sonda s teplotním čidlem

Jako senzory teploty jsou zvoleny sondy s teplotním čidlem DS18B20 od firmy Dallas, které se vyrábí i ve vodotěsném provedení viz obrázek 14. Sonda má 3 vývody z nichž 2 jsou napájecí VDD a GND a třetí je datový, DATA IN/OUT. Sonda umožňuje měření teploty v rozsahu od $-55\text{ }^{\circ}\text{C}$ do $+125\text{ }^{\circ}\text{C}$ s přesností $\pm 2\text{ }^{\circ}\text{C}$, přičemž v rozsahu od $-10\text{ }^{\circ}\text{C}$ do $+85\text{ }^{\circ}\text{C}$ zaručuje přesnost $0,5\text{ }^{\circ}\text{C}$. Sonda je možné napájet tzv. parazitně nebo z externího zdroje, a to v rozsahu od 3 V do 5,5 V, je možné ji tedy napájet z raspberry a informace o teplotě je možné přenášet digitálně pomocí digitální sběrnice OneWire.



Obrázek 14: Sonda s teplotním čidlem

2.1.6 LCD displej

Pro zobrazení aktuálních hodnot teploty, proudu a napětí a pro nastavení je využit LCD TFT Displej ST7735 viz obrázek 15. Displej má velikost 1,8 palce a rozlišení 128x160 pixelů a umožňuje zobrazit širokou škálu barev. Pro komunikaci mezi displejem a raspberry je využito sériové rozhraní SPI. Napájení displeje je možné jak 3,3 V, tak 5 V a maximální odběr displeje při maximálním podsvícení je 150 mA. Podsvícení lze regulovat pomocí PWM regulace pinem BL neboli Backlight.



Obrázek 15: LCD displej

2.1.7 Tlačítka a konektory

Dalším důležitým článkem celé práce jsou tlačítka a konektory. Pomocí tlačítek a informací na LCD displeji je totiž možné celý datalogger ovládat, tlačítka slouží pro spínání pouze malých proudů, aby předávali raspberry informaci o stisknutí, pro tuto konkrétní práci jsou tedy dostačující mikrotlačítka, viz obrázek 16.



Obrázek 16: Mikrotlačítka pro ovládání

Dalším potřebným spínačem je spínač pro spínání napájecího napětí pro stepdown měnič, pomocí tohoto spínače lze tedy zvolit, jestli celý datalogger má být napájen z interního zdroje, popřípadě že je spínač rozepnut, je nutné pro správnou funkci

dataloggeru, aby byl připojen na zdroj externího napájení pomocí vyvedeného microUSB konektoru. Zdroj pro externí napájení musí mít napětí 5 voltů a proud alespoň 500 mA, aby byla zajištěna správná funkce veškerých použitých modulů a nedošlo tak ke zkreslení měřené informace.



Obrázek 17: Spínač pro napájení

Konektory pro připojení externích teplotních čidel jsou potřeba se třemi piny, jelikož teplotní čidla DS18B20 disponují třemi vývody, a to pro napájení, zem a data. Zde opět nejsou kladené nijak velké důrazy na proudovou zatížitelnost, proto jsou zde zvojeny konektory se zámkem PFH02-03P viz obrázek 18. Na vodič se pomocí krimpovacích kleští připne zlacený kontakt PFF02-01FG, tyto kontakty se při vsunutí do konektoru pomocí zámků zafixují na určené pozici, a tím vznikne jedna strana konektoru, konkrétně samice. U druhé strany, tedy samce, je zde pro kompatibilitu zvojen konektor se zámkem KK254-90°. Tento konektor je úhlový, aby jej bylo možné připájet k desce plošných spojů.



Obrázek 18: Konektory pro připojení teplotních čidel

Posledními konektory jsou konektory pro připojení zdroje a spotřebiče. Na tyto konektory je kladen největší důraz, protože je nutné, aby jejich parametry vyhovovaly. Jelikož je potřeba měřit napětí až do 70 voltů a proud až do 50 ampér, jsou zde zvoleny výkonné konektory XT90 viz obrázek 19. Tyto konektory snesou vyšší proudovou zatížitelnost, a to trvalý proud o velikosti až 90 ampér. Dutinky jsou dostatečně rozměrné, aby se do konektoru mohl umístit vodič o průřezu až 4,5 mm². Konektory jsou pozlacené a jejich rozměry jsou 20x15x8 mm, co se týče konektorů spojených, tak délka spoje je 30 mm.



Obrázek 19: Výkonové konektory XT90

2.2 Obsluha jednotlivých modulů pomocí jazyka Python

2.2.1 A/D Převodník

Pro obsluhu A/D převodníku slouží knihovna *Adafruit ADS1x15*. Pomocí této knihovny lze do programu nainportovat objekt převodníku *ADS1115* a metodu *AnalogIn* pro získávání hodnoty na jednom ze čtyř analogových vstupů převodníku P0 až P3. Důležitou součástí je nadefinování komunikace I2C, pomocí které lze s převodníkem komunikovat. Aby bylo možné komunikaci využívat, je nutné ji nejprve v konfiguraci raspberry povolit. V programu je tedy vytvořen objekt převodníku a nadefinována konfigurace komunikace. Třída *chan* poté obsahuje aktuální informace o zvoleném vstupu, třída má definované proměnné *voltage*, která obsahuje informaci o úrovni napětí na vstupu a *value*, která obsahuje informaci a digitální hodnotě.

```
import board
import busio

i2c = busio.I2C(board.SCL,board.SDA)

import adafruit_ads1x15.ads1115 as ADS
from adafruit_ads1x15.analog_in import AnalogIn

ads = ADS.ADS1115(i2c)

ads.gain = 1

chan = AnalogIn(ads, ADS.P0)
chan1 = AnalogIn(ads, ADS.P1)
```

Zdrojový kód 1: Obsluha převodníku

2.2.2 Teplotní čidlo

Teplotní čidlo předává informace pomocí směrnice OneWire. Pro právnou funkčnost programu je nutné v konfiguraci povolit komunikaci po dané sběrnici. Digitální informace získané z komunikace jsou ukládány do systémového adresáře pro sběrnice. V tomto případě konkrétně do adresáře `/sys/bus/w1/devices/` do souboru začínajícím číslem 28. Program tedy vyhledává tento soubor v adresáři, následně ho otevře a čte po řádcích, v jednotlivých řádcích poté vyhledává informaci obsahující data o teplotě. Po vydělení informace se získává konkrétní naměřená hodnota teploty. Pro získávání informace z teplotního čidla slouží nadefinovaná metoda `read_temp()`.

```
import os
import glob
import time

os.system('modprobe w1-gpio')
os.system('modprobe w1-therm')

base_dir = '/sys/bus/w1/devices/'
device_folder = glob.glob(base_dir + '28*')[0]
device_file = device_folder + '/w1_slave'

def read_temp_raw():
    f = open(device_file, 'r')
    lines = f.readlines()
    f.close()
    return lines

def read_temp():
    lines = read_temp_raw()
    while lines[0].strip()[-3:] != 'YES':
        lines = read_temp_raw()
    equals_pos = lines[1].find('t=')
    if equals_pos != -1:
        temp_string = lines[1][equals_pos+2:]
        temp_c = float(temp_string) / 1000.0

    cas = time.strftime("%Y-%m-%d %H:%M:%S")
    return temp_c, cas
```

Zdrojový kód 2: Obsluha teplotního čidla

2.2.3 LCD displej

LCD displej, konkrétně ST7735 komunikuje pomocí sběrnice SPI, proto je zde opět nutné v konfiguraci povolit komunikaci po sběrnici SPI. Pro správnou funkci program využívá předdefinovanou knihovnu ST7735. Aby komunikace správně fungovala je nutné ji správně nadefinovat, a to jak výšku a šířku displeje, tak komunikační rychlost a zapojení řídicích a hodinových signálů. Pomocí parametrů displeje a SPI sběrnice se pak definuje objekt displej, v tomto konkrétním případě *disp*, který má metody *begin()* pro inicializaci a *display()* s parametrem obrazu jenž má zobrazit.

```
from PIL import Image
import ST7735 as TFT
import Adafruit_GPIO as GPIO
import Adafruit_GPIO.SPI as SPI

WIDTH = 128
HEIGHT = 160
SPEED_HZ = 4000000

DC = 24
RST = 25
SPI_PORT = 0
SPI_DEVICE = 0

disp = TFT.ST7735(
    DC,
    rst=RST,
    spi=SPI.SpiDev(
        SPI_PORT,
        SPI_DEVICE,
        max_speed_hz=SPEED_HZ))

disp.begin()
image = Image.open('img.png')
disp.display(image)
```

Zdrojový kód 3: Obsluha LCD displeje

2.2.4 Tlačítka

Aby tlačítka nebylo potřeba v programu neustále dotazovat na stisknutí, je zde využito mechanismu přerušení. Mechanismus nezávisle na procesoru registruje požadavky od tlačítek na přerušení a při stisknutí předává informaci procesoru, který obslouží metodu přerušení daného tlačítka. Všechna tlačítka jsou připojena jedním pinem k napájení a druhým pinem k GPIO headeru a jsou nastavena programem do pull down režimu.

```
import RPi.GPIO as GPIO

pin = 21

GPIO.setmode(GPIO.BCM)
GPIO.setup(pin, GPIO.IN, pull_up_down=GPIO.PUD_DOWN)

def my_callback(channel):
    print ("Rising edge detected on pin {0}".format(pin))
    # Kód pro obsluhu přerušení od pinu 21

GPIO.add_event_detect(
    up_pin,
    GPIO.RISING,
    callback=my_callback,
    bouncetime=200)
```

Zdrojový kód 4: Příklad obsluhy tlačítka pomocí přerušení

2.3 Uložení informací do souboru

2.3.1 Tvorba složky a načtení nastavení

U tvorby složky a načtení nastavení je využito bloku try/catch pro zachycení vyjímek. V bloku *try*, při tvorbě složky, program zjišťuje existenci uvedené složky a v případě neexistence se snaží složku vytvořit. V bloku *catch* je poté kód pro zobrazení chybové chlášky včetně informace o zachycené chybě. Pro načtení nastavení jsou zde dva bloky *try*, *a to* pro přidání, kdy se program snaží načíst nastavení a pro případ, kdy se snaží nastavení vytvořit, bloky *catch* jsou poté obdobné jako u tvorby složky, slouží pouze pro zobrazení zachycené chyby.

```

import os
cesta = ""
try:
    cesta = "/home/pi/DATA"
    if not os.path.exists(cesta):
        os.mkdir(cesta)
except:
    print("Nepodařilo se vytvořit složku {0}, zkontrolujte prosím svá oprávnění.".format(cesta))
if os.path.exists(os.path.join(cesta, "nastaveni.dat")):
    try:
        # Kód pro načtení nastavení ze souboru
        pass
    except Exception as e:
        print("Při načítání nastavení došlo k následující chybě: {0}".format(e))
else:
    try:
        # Kód pro vytvoření nastavení
        pass
    except Exception as e:
        print("Při vytvoření nastavení došlo k následující chybě: {0}".format(e))

```

Zdrojový kód 5: Vytvoření složky a načtení nebo vytvoření souboru s nastavením

2.3.2 Tvorba souboru

Pro tvorbu souboru je vybrán formát CSV, protože se jedná o jednoduchý souborový formát pro uložení tabulkových dat, jelikož program musí v každém intervalu snímání dat otevřít tento soubor, zapsat do něj data a následně soubor uzavřít, tak s tímto formátem ušetří čas a data jsou tak zachována například i při výpadku elektrické energie.

```

try:
    with open("/home/pi/DATA/data.csv", "a",
              encoding="utf-8") as f:
        s = "Text;"+"\\n"
        f.write(s)
        print("Bylo připsáno:",s)
except Exception as e:
    print("Při zapisu se vyskytla chyba:",e)

```

Zdrojový kód 6: Příklad vytvoření souboru formátu CSV

2.4 Program

2.4.1 Hlavní část

Hlavní část programu je cyklická a neustále se periferií dotazuje na aktuální naměřené hodnoty. Součástí hlavní části programu je nejen cyklické dotazování, ale hlavně počáteční nastavení. Před komunikací s periferiemi je totiž nutné nastavit všechny potřebné komunikační parametry. Je tedy potřeba nastavit parametry seriové linky SPI pro komunikaci s LCD displejem a zobrazování tak aktuální informace na displej, dále je potřeba nastavit parametry multi-masterové sběrnice I²C pro komunikaci s A/D převodníkem, ze kterého se získávají informace o naměřeném napětí a proudu. Posledním komunikačním nastavením je nastavení pro sběrnici OneWire. Pomocí této sběrnice může být připojeno několik teplotních sond, v tomto případě dvě, program tak získává informace o aktuální teplotě.

Aby bylo možné datalogger ovládat a program nemusel čekat na stisknutí tlačítka a tím se zacyklil a nemohl tak vykonávat jiné důležité potřeby, tak je zde využito mechanismu přerušení od vstupního pinu. Při stisku tlačítka se tedy vyvolá přerušení, opustí se hlavní cyklus a vykoná se potřeba daného přerušení, poté se načte návratová adresa a hlavní cyklus pokračuje ve fázi, ve které skončil před přerušením. Přerušení od tlačítek slouží pro spuštění a ukončení měření, pro nastavení veškerých parametrů jako je jas displeje, frekvence snímání dat, povolení teplotních sond a podobně.

Nejdůležitějším uživatelským parametrem je frekvence ukládání naměřených dat. Zde je opět využito mechanismu přerušení. Pokud uživatel zmáčkne příslušné tlačítko pro start měření, je načtena hodnota frekvence z nastavení a nastaví se časovač s periodou odpovídající zadané frekvenci. Jako další parametr časovače se nastaví metoda, která se má vykonat po přetečení časovače, po přetečení tedy opět dojde k přerušení, které otevře soubor s naměřenými daty, připiše aktuální hodnoty do souboru a soubor uzavře, tím se docílí zachování naměřených hodnot i při výpadku elektrické energie.

2.4.2 Nastavení

Aby bylo možné si nastavení pamatovat i bez elektrické energie, tak je pro uchování parametrů založen konfigurační soubor ve stejném adresáři, jako naměřená data. Konfigurační soubor obsahuje následující povinné parametry: frekvence snímání dat, jas a povolení teplotních sond, tyto parametry jsou povinné. Jako volitelné parametry jsou parametry: název Wi-Fi sítě a heslo k dané síti, které slouží pro připojení dataloggeru k síti Wi-Fi a sdílení tak naměřených dat bezdrátově bez nutnosti exportu dat z paměťové karty. Pokud tyto parametry nejsou zadány, tak datalogger vytváří vlastní zabezpečenou bezdrátovou Wi-Fi síť s názvem „Datalogger“. V síti je poté přístupný síťový adresář „RASPERRYPI“, ve kterém jsou uložena veškerá naměřená data a zmiňovaný konfigurační soubor. Aby uživatel mohl ke sdílenému adresáři přistupovat je potřeba uživatelské jméno a heslo, poté uživatel může data číst i měnit.

Další důležitou částí je nenalezení konfiguračního souboru, nebo načtení konfiguračního souboru s neznámými nebo špatně zadanými parametry. V obou z těchto případů program reaguje obdobně. Pokud je soubor nalezen, ale nesprávně zadaný, tak je soubor smáznán. Dalším krokem je načtení továrních hodnot a následné vytvoření konfiguračního souboru s již správnými parametry.

2.4.3 Podprogramy

Vzhledem k vysokým nárokům na rychlé snímání a ukládání dat je třeba, aby hlavní část programu byla co nejkratší. Proto veškeré méně často potřebné části kódu jsou definovány jako podprogramy. Podprogramy lze dělit na dvě skupiny dle jejich volání, a to na podprogramy, které jsou volány na základě činnosti uživatele a podprogramy, které jsou volány na základě činnosti programu. Hlavní část programu volá například podprogram *Uloz*, který naměřená data uloží do souboru. Uživatelem vyvolané podprogramy jsou pak například přerušení vyvolané stiskem tlačítek.

2.4.4 Grafická část

LCD displej slouží pouze pro zobrazování aktuální činnosti dataloggeru pro uživatele. Proto je třeba, aby zobrazovaná informace byla lehce pochopitelná a srozumitelná. Aby se docílilo

zobrazování informací na displeji, je v programu využita python knihovna *PIL* a z této knihovny konkrétně instance *Image*, *ImageFont* a *ImageDraw*. Instance *Image* slouží pro tvorbu podkladu obrazu. *ImageDraw* slouží pro generování obrázků a psaní textu do podkladu společně s *ImageFont*.

Samotná grafická stránka dataloggeru je pak rozdělena do dvou částí. První část obsahuje textovou informaci o aktuálním dění a druhá část, která je umístěna ve spodní části displeje tvoří takzvanou lištu. Jelikož pro ovládání celého dataloggeru slouží čtyři mikro-tlačítka umístěné pod LCD displejem, tak tato lišta je rozdělena do čtyř částí, kde každá část přísluší jednomu tlačítku. Každá část tedy popisuje funkční význam daného tlačítka, například přesun na další stránku, zvýšení frekvence ukládání dat, uložení nastavení a start měření.

2.5 Měření

2.5.1 Měření napětí

Pro zjištění přesnosti měření dataloggeru bylo měřeno napětí v rozsahu od 1 V do 42 V. Jako referenční měřicí přístroj byl zvolen digitální univerzální multimetr TO-81783. U multimetru byly využity rozsahy 20 V a 200 V u obou rozsahů je udávaná přesnost $\pm 0,5\%$. Tabulka 5 obsahuje záznam z měření. Ve sloupci U_z je uvedeno přibližně nastavené napětí zdroje, ve sloupci U_m je naměřené napětí na měřicím přístroji a ve sloupci U_d je naměřené napětí dataloggerem. Sloupce U_{min} a U_{max} vyjadřují toleranční pásmo měřicího přístroje a sloupec A/N pak vyjadřuje, jestli naměřené napětí dataloggerem se nachází v tolerančním pásmu. Poslední sloupec $diff$ pak udává procentuální rozdíl naměřené hodnoty dataloggeru od referenčního měřicího přístroje. Z měření je vyjádřena průměrná odchylka od referenčního přístroje, která je 0,18%.

Tabulka 5: Měření napětí

U_z [V]	U_m [V]	U_{min} [V]	U_{max} [V]	A/N	U_d [V]	diff [%]
2	1,98	1,97	1,99	Ano	1,99	0,50
3	3,00	2,99	3,02	Ano	3,00	0,00
4	4,02	4,00	4,04	Ano	4,02	0,00
5	5,02	4,99	5,05	Ano	5,01	0,20
6	6,05	6,02	6,08	Ano	6,04	0,17
8	8,03	7,99	8,07	Ano	8,00	0,37
10	10,04	9,99	10,09	Ano	10,00	0,40
12	12,04	11,98	12,10	Ano	11,99	0,42
14	14,06	13,99	14,13	Ano	14,09	0,21
16	16,04	15,96	16,12	Ano	16,03	0,06
18	18,13	18,04	18,22	Ano	18,12	0,06
20	19,94	19,84	20,04	Ano	19,86	0,40
22	22,1	21,99	22,21	Ano	22,14	0,18
24	24,1	23,98	24,22	Ano	24,16	0,25
26	26,0	25,87	26,13	Ano	26,06	0,23
28	28,0	27,86	28,14	Ano	28,03	0,11
30	30,0	29,85	30,15	Ano	30,02	0,07
32	32,2	32,04	32,36	Ano	32,28	0,25
34	34,0	33,83	34,17	Ano	34,01	0,03
36	36,1	35,92	36,28	Ano	36,16	0,17
38	38,1	37,91	38,29	Ano	38,09	0,03
40	40,0	39,80	40,20	Ano	40,06	0,15
41	41,1	40,89	41,31	Ano	41,14	0,10
42	42,0	41,79	42,21	Ano	41,96	0,10
Průměrná odchylka v %:						0,18

2.5.2 Měření proudu

Pro měření proudu datalogger využívá hallovu sondu ACS770 LCB-050B. Celková udávaná chyba sondy je maximálně $\pm 1,9\%$. Pro měření proudu je opět využit měřicí přístroj TO-81783, který je při měření brán jako referenční. Tabulka 6 opět obsahuje naměřené hodnoty. Veškeré sloupce tabulky jsou obdobné jako u měření napětí. Jsou zde hodnoty nastaveného proudu na zdroji, změřeného proudu měřicím přístrojem a dataloggerem, toleranční pásmo přístroje a relativní odchylka. Navíc je zde sloupec s absolutní odchylkou vyjádřenou v mA. Průměrná naměřená odchylka od referenčního přístroje je 0,99%.

Tabulka 6: Měření proudu

I_z	I_m	I_{min}	I_{max}	A/N	I_d	diff [mA]	diff [%]
1	1,01	0,99	1,03	Ano	1,02	10	0,99
1,5	1,51	1,48	1,54	Ano	1,49	20	1,32
2	2,00	1,96	2,04	Ano	1,98	20	1,00
2,5	2,51	2,46	2,56	Ano	2,48	30	1,20
3	3,01	2,95	3,07	Ano	2,98	30	1,00
3,5	3,52	3,45	3,59	Ano	3,48	40	1,14
4	4,01	3,93	4,09	Ano	3,97	40	1,00
4,5	4,56	4,47	4,65	Ano	4,51	50	1,10
5	5,02	4,92	5,12	Ano	4,98	40	0,80
6	6,03	5,91	6,15	Ano	5,97	60	1,00
7	7,06	6,92	7,20	Ano	6,99	70	0,99
8	8,02	7,86	8,18	Ano	7,97	50	0,62
9	9,00	8,82	9,18	Ano	8,92	80	0,89
10	9,95	9,75	10,15	Ano	9,86	90	0,90
Průměrná odchylka v %:							0,99

2.5.3 Měření odběru

Pro zjištění odběru v závislosti na nastaveném jasu displeje byl změřen odběrový proud digitálním multimetrem TO-81783 s přesností $\pm 1,2\%$. Tabulka 7 obsahuje naměřené hodnoty. Pro měření bylo zvoleno konstatní napětí zdroje 10 V. Z měření vyplývá, že maximální odběrový proud dataloggeru při napájení 10 volty je přibližně 200 mA.

Tabulka 7: Měření odběru při různém jasu

Jas [%]	I [mA]
100	200
90	190
80	180
70	170
60	160
50	150
40	140
30	140
20	130
10	120

Aby měření bylo kompletní, tak byl změřen i odběrový proud při různém vstupním napětí a z naměřených hodnot byl vypočten celkový příkon dataloggeru. Tabulka 8 obsahuje veškeré naměřené a vypočtené hodnoty. V tabulce je U napětí napájecího zdroje, I je odběr ze zdroje v mA (měřen měřicím přístrojem TO-83783). Při odběru proudu do 200 mA je měřeno s přesností $\pm 1,2\%$ a při odběru vyšším než 200 mA je měřeno s přesností $\pm 2\%$. Z měření vyplývá, že při snižování vstupního napětí roste odběrový proud, ale roste i účinnost stepdown měniče.

Tabulka 8: Měření odběru při různém napájení

U [V]	I [mA]	P [W]
42	66	2,77
40	67	2,68
35	75	2,63
30	80	2,40
25	97	2,43
20	121	2,42
15	162	2,43
14	167	2,34
13	175	2,28
12	181	2,17
11	192	2,11
10	200	2,00
9	230	2,07
8	250	2,00
7	310	2,17

Závěr

Pro navrhnutí dataloggeru byl zvolen jednodeskový počítač Raspberry Pi Zero W. Oproti arduinu sice strácí výhodu analogových vstupů, ale disponuje bezdrátovými rozhraními Wi-Fi a Bluetooth, slotem pro připojení vysokokapacitní paměťové karty a vyšší frekvencí procesoru. Pro tvorbu aplikace bylo využito vývojové prostředí Raspbian, jelikož se jedná o intuitivní prostředí pro tvorbu aplikací pro jednodeskové počítače Raspberry Pi.

Pro měření proudu byl zvolen hallův senzor proudu ACS770LCB-050B od firmy Allegro Microsystems, který disponuje možností měření ± 50 A a měřený proud poté odpovídá analogové hodnotě napětí na výstupu s citlivostí 40 mV na jednu ampéru. Převod napětí z výstupu hallova senzoru a zároveň převod měřeného napětí zdroje obstarává A/D převodník ADS1115, který má rozlišení 16 bitů a převede tak analogovou hodnotu s vysokou přesností. Aby bylo možné měřit napětí až do výše 70 V, je přivedeno napětí do převodníku přes odporový dělič. Pro získání digitálních hodnot napětí a proudu z A/D převodníku slouží sběrnice I2C, pomocí které převodník komunikuje s raspberry.

Měření teploty zajišťuje teplotní čidlo DS18B20. Čidlo umožňuje měřit teplotu v rozsahu od -50 °C až do $+125$ °C s přesností ± 2 °C, přičemž v rozsahu od -10 °C až do $+85$ °C má zaručenou přesnost $\pm 0,5$ °C. Čidlo je připojeno k raspberry přes sběrnici OneWire, pomocí které předává informace do raspberry o aktuální teplotě.

Hlavní program pro raspberry nastavuje veškeré parametry pro jednotlivé periferie, definuje metody pro obsluhu tlačítek a ukládání dat. V programu je cyklicky vykonáváno dotazování periferií na aktuální naměřené informace a zobrazování aktuálních hodnot na displeji. Pro obsluhu stisknutých tlačítek je zvoleno přerušení. Každému tlačítku je tedy přiděleno přerušení a metoda pro obsluhu. Docílí se tím větší efektivity při běhu programu a není třeba se na stav tlačítek neustále dotazovat. Pro ukládání dat do souboru je zvoleno přerušení od časovače.

Posledním krokem volby bylo zvolení výstupního formátu souboru s naměřenými daty. Pro tento soubor je vybrán formát CSV, jedná se o jednoduchý souborový formát pro uložení tabulkových dat. Jelikož datalogger musí v každém intervalu snímání dat otevřít soubor, zapsat do něj data a následně soubor uzavřít, tak s tímto formátem ušetří spoustu času a data jsou tak zachována například i při výpadku elektrické energie.

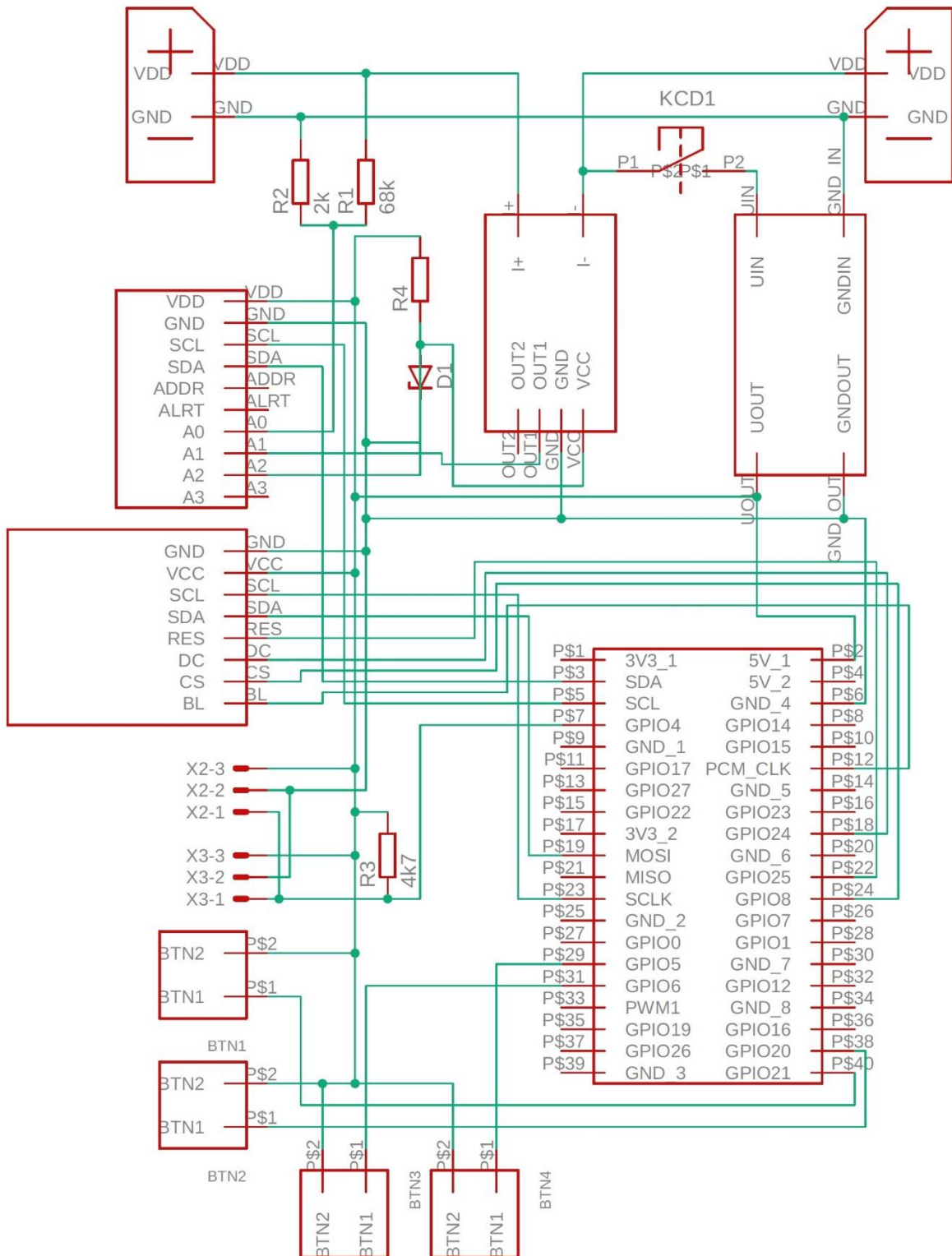
Seznam použité literatury

- [1] DOLEŽAL, I. aj. Analogová elektronika 1. vyd. Liberec: TU v Liberci, 2014 DOI: 10.15240/tul/002/2014-11-003 ISBN: 978-80-7494-136-8
- [2] MIKÉŠKA, Z. Specifikace rádiové části systému Bluetooth [online]. [cit. 2020-05-17]. Dostupné z: <http://www.elektrorevue.cz/clanky/04003/index.htm>
- [3] HOREVAJ, M. Měření frekvence [online]. [cit. 2020-04-12]. Dostupné z: <https://vf-elektronika-foto.cz/K5.html>
- [4] Odporové senzory teploty [online]. [cit. 2020-03-11]. Dostupné z: http://www.umel.feec.vutbr.cz/~adamek/uceb/DATA/s_3_2_4.htm
- [5] Měření kmitočtu. Sotovo [online]. [cit. 2020-03-12]. Dostupné z: <http://www.sotovo.cz/str/elm.php?id=40>
- [6] Python. *Wikipedie* [online]. [cit. 2020-04-14]. Dostupné z: <https://cs.wikipedia.org/wiki/Python>
- [7] Přirozené datové typy. *Ponořme se do Pythonu 3* [online]. [cit. 2020-04-14]. Dostupné z: <http://diveintopython3.py.cz/native-datatypes.html>
- [8] I2C. *Wikipedie* [online]. [cit. 2020-04-17]. Dostupné z: <https://cs.wikipedia.org/wiki/I%C2%B2C>
- [9] SPI interface Tutorial. *Best Microcontroller Projects* [online]. [cit. 2020-04-17]. Dostupné z: <https://www.best-microcontroller-projects.com/spi-interface.html>
- [10] Komunikace po sériové sběrnici I2C. *ROOT.CZ* [online]. [cit. 2020-04-17]. Dostupné z: <https://www.root.cz/clanky/komunikace-po-seriove-sbornici-isup2supc/>

Seznam použitých obrázků

- [11] Hallův jev. *In: WikiSkripta* [online]. [cit. 2020-03-11]. Dostupné z:
https://www.wikiskripta.eu/w/Hall%C5%AFv_jev#/media/File:Hall%C5%AFv_jev_b.jpg
- [12] Paralelní převodník, *In: Wikimedia Commons* [online]. [cit. 2020-04-17]. Dostupné z:
<https://upload.wikimedia.org/wikipedia/commons/b/be/AD-par.jpg>
- [13] Převodník s postupnou aproximací. *In: Wikipedie* [online]. [cit. 2020-04-17].
Dostupné z:
https://cs.wikipedia.org/wiki/A/D_p%C5%99evodn%C3%ADk#/media/Soubor:Kompenzacni_sledovaci_AD.png
- [14] Integrovaný převodník. *In: Wikipedie* [online]. [cit. 2020-04-17]. Dostupné z:
https://cs.wikipedia.org/wiki/A/D_p%C5%99evodn%C3%ADk#/media/Soubor:AD_s_dvoutaktn%C3%AD_integraci.png
- [15] Sigma delta převodník. *In: Wikipedie* [online]. [cit. 2020-04-17]. Dostupné z:
https://cs.wikipedia.org/wiki/A/D_p%C5%99evodn%C3%ADk#/media/Soubor:Sigma_delta_prevodnik.png
- [16] Serial Peripheral Interface. *In: Wikipedie* [online]. [cit. 2020-04-17]. Dostupné z:
https://cs.wikipedia.org/wiki/Serial_Peripheral_Interface#/media/Soubor:SPI_single_slave.svg
- [17] I2C. *In: Wikipedie* [online]. [cit. 2020-04-17]. Dostupné z:
<https://cs.wikipedia.org/wiki/I%C2%B2C#/media/Soubor:I2C.svg>
- [18] GPIO Pinout Diagram. *In: Raspberry Pi* [online]. [cit. 2020-04-17]. Dostupné z:
<https://www.raspberrypi.org/documentation/usage/gpio/images/GPIO-Pinout-Diagram-2.png>

Příloha A Schéma zapojení



Obrázek 20: Schéma zapojení

Příloha B Datalogger



Obrázek 21: Zhotovený datalogger