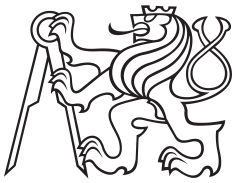
Bakalářská práce



České
vysoké
učení technické
v Praze

F3

Fakulta elektrotechnická
Katedra radioelektroniky

Modulární měřicí přístroj

Jan Hamacek

Vedoucí: doc. Ing. Stanislav Vítek, Ph.D.
Studijní program: Elektronika a komunikace
Květen 2023

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Hamacek** Jméno: **Jan** Osobní číslo: **499050**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra radioelektroniky**

Studijní program: **Elektronika a komunikace**

II. ÚDAJE K BAKALÁŘSKÉ PRÁCI

Název bakalářské práce:

Modulární měřicí přístroj

Název bakalářské práce anglicky:

Modular Measurement Device

Pokyny pro vypracování:

Navrhněte a implementujte modulární měřicí přístroj. Při vypracování se řiďte následujícími pokyny:
- navrhněte rozhraní pro připojení výpočetní jednotky
- přístroj bude obsahovat jednotné rozhraní pro připojení měřicích modulů
- navrhněte softwarové rozhraní pro interakci komponent
- realizujte měřicí přístroj s Raspberry PI
- realizujte měřicí modul konstantní proudové zátěže

Seznam doporučené literatury:

[1] HAASZ, Vladimír, Jan HOLUB, Michal JANOŠEK, Petr KAŠPAR a Vojtěch PETRUCHA. Elektrická měření: přístroje a metody. 3. přepracované vydání. Praha: Česká technika - nakladatelství ČVUT, 2018. ISBN 978-80-01-06412-2.
[2] HÄBERLE, Heinz O. Průmyslová elektronika a informační technologie. Praha: Europa-Sobotáles, 2003. ISBN 80-867-0604-4.
[3] MOLLOY, Derek. Exploring Raspberry Pi: interfacing to the real world with embedded Linux. Indianapolis, IN: Wiley, [2016]. ISBN 978-1-119-18868-1.

Jméno a pracoviště vedoucí(ho) bakalářské práce:

doc. Ing. Stanislav Vítek, Ph.D. katedra radioelektroniky FEL

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

Datum zadání bakalářské práce: **13.02.2023** Termín odevzdání bakalářské práce: **26.05.2023**

Platnost zadání bakalářské práce: **22.09.2024**

doc. Ing. Stanislav Vítek, Ph.D. podpis vedoucí(ho) práce	doc. Ing. Stanislav Vítek, Ph.D. podpis vedoucí(ho) ústavu/katedry	prof. Mgr. Petr Páta, Ph.D. podpis děkana(ky)

III. PŘEVZETÍ ZADÁNÍ

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

. Datum převzetí zadání

Podpis studenta

© ČVUT v Praze, Design: ČVUT v Praze, VIC

Poděkování

Chtěl bych poděkovat svému vedoucímu doc. Ing. Stanislavu Vítkovi, Ph.D. za umožnění vytvoření této práce a podporu při její realizaci.

Prohlášení

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

V Praze, 23. 5. 2023

Abstrakt

Tato práce se zabývá návrhem a realizací platformy pro měřicí přístroje. V první části jsem se zaměřil na návrh standardizovaného modulárního měřicího přístroje. Poté jsem realizoval přístroj včetně softwaru pro jeho provoz. Následně byla ověřena funkčnost přístroje realizací modulu konstantní proudové zátěže.

Klíčová slova: softwarově definovaný přístroj, Raspberry Pi, modulární měření, konstantní proudová zátěž

Vedoucí: doc. Ing. Stanislav Vítek, Ph.D.

Abstract

This work describes the design and realization of a measurement platform. Firstly we designed a standardized platform for modular measurement instrument. Then we built a standalone instrument and it's software. Finally, we designed a measurement module with a constant current load.

Keywords: software defined instrument, Raspberry Pi, modular measurement, constant current load

Title translation: Modular Measurement Device

Obsah

1 Úvod **1**
Existující přístroje 1
2 Architektura platformy **3**
2.1 Rozhraní mezi moduly 3
2.1.1 Využité sběrnice 3
2.1.2 Napěťové úrovně rozhraní 4
2.2 Rozhraní měřicího modulu 4
2.3 Rozhraní ovládacího modulu 4
2.4 Základní deska 5
2.4.1 Napájení modulů 5
2.4.2 A/D a D/A převodníky 5
2.4.3 Digitální rozhraní základní desky 6
2.4.4 Podpůrné funkce základní desky 6
2.4.5 I^2C zařízení 7
2.5 Mechanický návrh 7
2.6 Software 7
2.6.1 Rozhraní s hardware 7
2.6.2 Software základní desky 8
2.6.3 Softwarové moduly 8
2.6.4 Komunikace měřicího modulu s IO moduly 9
2.6.5 Kolize I^2C sběrnice 9
2.6.6 Start a běh systému 9
2.6.7 Klávesnice 9
2.7 Měřicí modul 9
2.7.1 Časová náročnost výroby modulu 10
3 Realizace platformy **13**
3.1 Procesor 13
3.2 Základní deska 13
3.2.1 Analogové převodníky 13
3.2.2 Digitální signály 14
3.2.3 Napájení 14
3.3 Ovládací modul 14
3.4 Softwarová implementace 15
3.4.1 Grafické rozhraní 15
3.4.2 Softwarové moduly 16
3.5 Náklady 17
3.6 Opravy ve schématech 17
4 Realizace měřicího modulu **19**
4.1 Teoretický rozbor 19
4.1.1 Regulační smyčka 19
4.2 Implementace 20
5 Závěr **23**
Bibliografie **25**
A Seznam zkratek **27**
B Seznam příloh **29**

Obrázky

1.1 Jednodeskovy SDI 2
1.2 SDI od Red Pitaya 2

2.1 Architektura platformy přístroje. 4
2.2 Prototypovací deska 10

3.1 Realizovaný přístroj. 14
3.2 Základní deska. 15
3.3 Grafické rozhraní 16

4.1 Regulace konstantního proudu. . 20
4.2 Realizace výstupního modulu . . . 21
4.3 Výstupní charakteristika 21

Tabulky

2.1 Rozhraní měřicího modulu 5
2.2 Rozhraní ovládacího modulu 6

3.1 Náklady na realizaci přístroje . . 17

Kapitola 1
Úvod

V této práci se zabývám návrhem a realizací platformy pro modulární měřicí přístroje. Návrh je založen na softwarově definovaných přístrojích, což jsou přístroje, které dodávají základní stavební bloky přístroje a samotná metoda měření je vytvořena externě. Tento způsob umožňuje lépe využít místo a snižuje cenu přístrojů. Příkladem těchto přístrojů je například Moku:Go [1], nebo Red Pitaya [2] na obrázku 1.2. Tyto přístroje nabízejí pokročilé funkce na rozdíl od jednočipových řešení, jako LEO [3] na obrázku 1.1, který implementuje pouze velice základní rozhraní. Všechny tyto přístroje ovšem potřebují připojení k počítači pro svojí funkčnost, případně mají vysokou pořizovací cenu. Cílem této práce bylo vytvořit nový přístroj, který by umožňoval samotnou měřicí funkcionalitu integrovat do jednoho zařízení včetně ochranných obvodů.

Samotný přístroj je navrnut v několika funkčních blocích, které jsou zaměnitelné a tím umožňují přizpůsobení přístroje požadavkům. Chtěl jsem umožnit fungování přístroje bez výpočetní jednotky s přípojením k osobnímu počítači. Tento princip jsem přenesl i do softwaru, který využívá minimální rozhraní pro komunikaci s fyzickým rozhraním a jeho nastavení, kdy zanechává široké možnosti pro jeho rozšíření o další funkce.

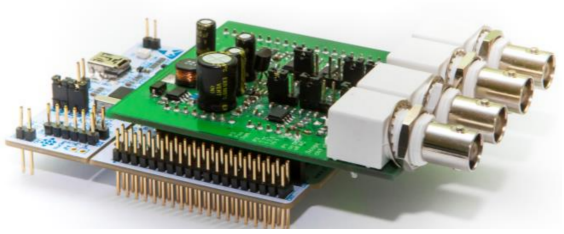
Důležitým prvkem při realizaci byla také srozumitelnost. Samotné moduly se skládají z diskrétních funkčních bloků, které mají jasnou funkcionalitu a je možné je vyměnit. Motivací pro tento krok bylo případné umožnění použití přístroje jako výukové pomůcky, kdy je na přístroji možné demonstrovat základní principy elektronických měřicích systémů.

Následně jsem pro tuto platformu navrhl a realizoval konstantní proudovou zátěž, která umožňuje i výpočet spotřebované energie.

Existující přístroje

Aktuálně jsou na trhu k dostání přístroje například od *Liquid Instruments* přípravek *Moku:Go* nebo od *Red Pitaya* přípravek *STEMlab 125-14*. Tyto přípravky dosahují výrazně lepších parametrů, než přístroj navržený v této práci. Pro porovnání *Moku:Go* a *STEMlab 125-14* zvládnou zpracovávat signály se vzorkovacím kmitočtem 125 Msa/s zatímco řešení popsané v této práci dosahuje 7 Sa/s. Ovšem cena těchto zařízení se pohybuje od 13 000 Kč pro *Moku:Go* a 11 000 Kč pro *STEMlab 125-14* zatímco základní implementace

přístroje popsaného v této práci je okolo 4000 Kč bez displeje nebo počítače. Tímto se přístroj popsaný v této práci blíží spíše jednočipovým řešením a je vhodný pro měření v reálném čase ve výukovém prostředí.



Obrázek 1.1: Realizace jednodeskového SDI LEO [4].

Obrázek 1.2: Komerční SDI od firmy Red Pitaya *STEMlab 125-14* [2].

Kapitola 2
Architektura platformy

Přístroj využívá architektury založené na jednotlivých modulech, které spolu komunikují přes standardizované rozhraní. Tento princip modularity je využíván jak v hardwaru, tak i v softwaru. Jednotlivé moduly umožňují jednoduchou změnu funkce přístroje. Samotný přístroj se skládá ze tří hlavních modulů. Těmi jsou základní deska, měřicí modul a ovládací modul. Ovládací modul implementuje rozhraní měřicího přístroje s okolním světem. To může znamenat například samostatnou jednotku na které běží software, nebo překlad potřebných sběrnic pro připojení k PC s řídícím SW. Měřicí modul implementuje samotnou měřicí funkcionalitu. Základní deska implementuje A/D převodníky, digitální výstupy, překlad napěťových hladin a další funkcionality potřebné pro funkci zbylých modulu.

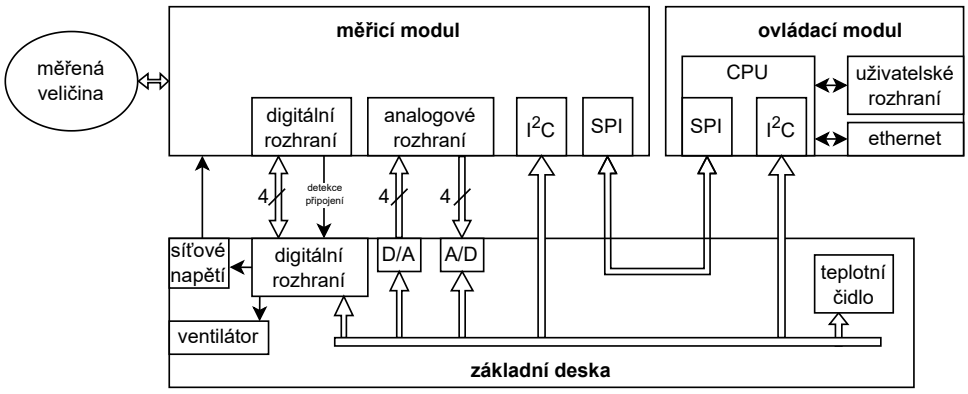
Jednotlivé funkce modulů jsou zobrazeny na obrázku 2.1, který definuje jednotlivé rozhraní mezi moduly.

2.1 Rozhraní mezi moduly

Návrh komunikace mezi moduly je důležitý kvůli využitelnosti modulů. Při realizaci jsem se tedy zaměřil na to, aby byl návrh co nejvíce univerzální a nezamezoval vytvoření specifických modulů.

2.1.1 Využité sběrnice

Při návrhu bylo potřeba vybrat rozhraní, po kterém bude celé zařízení komunikovat. Možné sběrnice jsou například I^2C, SPI nebo UART, které jsou jednoduše dostupné na procesorech a počítačích. Pro účel ovládání jednotlivých komponent nebylo možné využít sběrnici UART kvůli chybějící možnosti multiplexu a relativně velkému počtu zařízení, která by vyžadovalá rozměrnou sběrnici. Dále jsem zvažoval sběrnici SPI, kterou by bylo možné multiplexovat, ale tento krok by vyžadoval vodič pro každé zařízení, což by vyžadovalo velké množství digitálních výstupů z procesorové desky. Nakonec byla zvolena sběrnice I^2C, která umožňuje využití dvou vodičů. Tato volba zároveň umožňuje jednoduché přidávání jiných zařízení pokud jsou adresně kompatibilní. Nevýhodou této volby je relativně nízká přenosová rychlost I^2C sběrnice vůči SPI a nemožnost připojení měřicích modulů využívajících integrované obvody



Obrázek 2.1: Architektura platformy přístroje.

s SPI. Proto byla přidána i SPI sběrnice, která nahlíží na měřicí modul jako na jedno zařízení a přímo ho spojuje s ovládacím modulem. To umožňuje jednoduchou konfiguraci pomalých periferii pomocí sběrnice I^{2}C a případný vysokorychlostní přenos dat přes SPI.

2.1.2 Napěťové úrovně rozhraní

Jednotlivé moduly můžou volit napětí na svém digitálním rozhraní. To snižuje komplexnost jednotlivých modulů, kdy nemusí duplikovat jednotlivé převodníky, pokud jejich napětí není stejné jako využívá základní deska. Napěťové rozmezí analogového rozhraní je pevně stanovené na 0 V až 4 V.

2.2 Rozhraní měřicího modulu

Měřicí modul obsahuje samotnou měřicí funkcionalitu. Pro ovládání můžou být použity A/D a D/A převodníky ze základní desky, nebo digitální SPI a I^{2}C rozhraní. Tento modul musí obsahovat I^{2}C EEPROM pro identifikaci modulu a načtení příslušného programu. Samotné rozhraní není galvanicky izolované. Modul také obsahuje i přední panel s výstupy potřebnými pro modul. Samotné A/D a D/A převodníky mají rozmezí vstupního napětí od 0 V do 4 V. Digitální rozhraní má vstupní napětí od 0 V do 3,3 V, nebo 5 V podle volby na kontaktu *napájení sběrnice*. Dále musí být spojen kontakt *detekce připojení* s 5 V pro správnou detekci připojení napětí. Samotné kontakty jsou popsané v tabulce 2.1.

2.3 Rozhraní ovládacího modulu

Rozhraní mezi ovládacím modulem je čistě digitální přes sběrnici I^{2}C a SPI. Tato volba umožňuje ovládat přístroj z dedikovaného procesoru jako samostatnou jednotku nebo jako periferie jiného přístroje. Samotné rozhraní funguje v digitálních úrovních od 0 V do 3,3 V. Fyzické rozhraní modulu je popsáno

GND	1	2	GND
I^2C SCL	3	4	I^2C SDA
SPI MISO	5	6	SPI MOSI
SPI CLK	7	8	SPI CE
GND	9	10	GND
napájení sběrnice	11	12	detekce připojení
GND	13	14	GND
5 V	15	16	5 V
GND	17	18	GND
3,3 V	19	20	3,3 V
GND	21	22	GND
D/A 0	23	24	A/D 0
D/A 1	25	26	A/D 1
D/A 2	27	28	A/D 2
D/A 3	29	30	A/D 3
GND	31	32	GND
IO 0	33	34	IO 2
IO 1	35	26	IO 3

Tabulka 2.1: Rozhraní měřicího modulu

v tabulce 2.2. Pro správnou funkci je potřeba spojit vodič *napájení sběrnice* s vodičem 5 V nebo 3,3 V, čímž se nastaví digitální úroveň.

2.4 Základní deska

Základní deska obsahuje funkce potřebné pro fungování měřicího přístroje, například napájení modulů, A/D a D/A převodníky, teplotní senzor a převodníky úrovní. Také obsahuje rezistory pro správné fungování I^2C sběrnice. Při návrhu desky je potřeba zachovat rozměry a polohy konektorů a montážních otvorů.

2.4.1 Napájení modulů

Základní deska musí poskytovat ostatním modulům napájecí napětí 5 V a 3,3 V. V případě, že je přístroj napájen ze síťového napětí, je možné aby bylo implementované bezpečné spínání tohoto napětí do měřicího modulu, aby nemohlo dojít k připojení modulu k síťovému napětí bez jeho správného nastavení a řízení.

2.4.2 A/D a D/A převodníky

Výstupnímu modulu je potřeba umožnit využít čtyři analogové kanály pro vstup a čtyři analogové kanály pro výstup. Modul také obsahuje napěťovou referenci s nominální hodnotou 4,096 V, jejíž přesná hodnota musí být uložena

GND	1	2	GND
I^2C SCL	3	4	I^2C SDA
SPI MISO	5	6	SPI MOSI
SPI CLK	7	8	SPI CE
GND	9	10	GND
napájení sběrnice	11	12	nevyužito
GND	13	14	GND
5 V	15	16	5 V
5 V	17	18	5 V
GND	19	20	GND
3,3 V	21	22	3,3 V
3,3 V	23	24	3,3 V
GND	25	26	GND
nevyužito	27	28	nevyužito
nevyužito	29	30	nevyužito
nevyužito	31	32	nevyužito
nevyužito	33	34	nevyužito
nevyužito	35	26	nevyužito

Tabulka 2.2: Rozhraní ovládacího modulu

pro pozdější výpočty rozsahů a může být korigována na základě aktuální teploty okolí.

2.4.3 Digitální rozhraní základní desky

Primární komunikační sběrnicí je sběrnice I^2C. Tato sběrnice je použita pro konfiguraci a přenos nízkého objemu dat. V případě potřeby přenosu velkého množství dat je ovládací modul přímo spojen s měřicím modulem pomocí sběrnice SPI. Měřicí modul zároveň obsahuje digitální výstupy připojené na sběrnici, které jsou využity pro ovládání ventilátoru nebo síťového napětí. Dále musí být čtyři digitální kanály připojeny na rozhraní měřicího modulu.

2.4.4 Podpůrné funkce základní desky

Kromě samotné komunikace modulů základní deska dodává i funkce potřebné pro funkce zbylých částí přístroje.

Ventilátor

Základní deska musí umožnit spínání ventilátoru ze softwaru. Konkrétní nároky na ventilátor jsou specifikovány základní deskou. Tento krok byl zvolen z důvodu různých nároků na ventilátor, primárně u napájecího napětí, které může být v základní implementaci 5 V, ovšem není důvod omezovat návrh pouze na tyto ventilátory vzhledem k větší dostupnosti 12 V ventilátorů.

Síťové napětí

Některé měřicí moduly můžou vyžadovat síťové napětí. Není ovšem vhodné ho dodávat, pokud není správně nakonfigurován modul. Proto může základní deska umožňovat spínání síťového napětí ze softwaru. Tato funkce musí být označena v EEPROM základní desky.

Měření teploty

Základní deska musí umožnit softwaru měření okolní teploty pro případnou kalibraci. Toto může být realizováno I^2C teplotním senzorem s přesností minimálně $\pm 5\,°C$, například TC74[5].

2.4.5 I^2C zařízení

Návrh vyžaduje existenci dvou I^2C EEPROM kompatibilních s řadou 24C02 [6]. Tyto integrované obvody musí mít nastavenou adresu 0x50 pro paměť modulu a 0x51 pro paměť základní desky. Další integrované obvody můžou mít libovolné adresy a případné konflikty budou vyřešeny za běhu programu. Výstupní modul si může přidal vlastní I^2C komponenty.

2.5 Mechanický návrh

Samotný přístroj se skládá ze tří desek plošného spoje. Jejich rozměry jsou standardizované a v příloze jsou dostupné v návrhu základní desky. Rozměry základní desky by měly být dodrženy pro kompatibilitu. Ovšem rozměr modulu se může měnit, pokud bude zachována pozici konektoru a profil horní části modulu, který umožňuje upevnění modulů. Pro připojení modulů byl zvolen konektor do kterého lze přímo zapojit tištěný spoj, což zjednodušuje realizaci měřicího modulu.

2.6 Software

Důležitou součástí návrhu je i software. Při jeho vytváření byl brán důraz na modularitu a abstrakci. V případě změny procesorové jednotky je potřeba vyměnit soubor *IO_HAL.py*. Dále je vyžadována existence objektu MainBoard v *MainBoard<ID>.py*. Tento objekt je možné načítat na základě ID uloženého v EEPROM na základní desce. Po načtení základní desky se získají data o připojeném modulu a načítají se softwarové moduly. Měřicí moduly také obsahují přední panel, který realizuje potřebné rozhraní přístroje. Model předního panelu je v příloze této práce.

2.6.1 Rozhraní s hardware

Pro interakci s hardware je použita třída objekt *IO_HAL* v souboru *IO_HAL.py*. Tato třída musí implementovat rozhraní pro práci s I^2C a to metodou

I2C_WR, dále metodu pro práci s rozhraním SPI metodou *SPI_Transfer*. Tento objekt je prvním načteným objektem a musí tedy implementovat metodu pro přečtení EEPROM základní desky metodou *GetMainBoardEEP*. Minimální metody jsou v 2.1.

```
class ConnectionHAL:
    def __init__(self):
    def cleanup(self):
    def I2C_WR(self, address, Wdata, RdataSize):
    def SPI_Transfer(self, data):
    def GetMainBoardEEP(self):
```

Vzor kódu 2.1: Metody třídy implementující rozhraní s přístrojem.

2.6.2 Software základní desky

Funkce základní desky jsou implementované ve složce *Main_Boards* a jsou načítány podle ID v EEPROM. Objekt *MainBoard* implementuje metody na ovládání A/D a D/A převodníků, spínání IO, ventilátoru, síťového napětí a získání teploty prostředí. Tyto funkce jsou ovládané metodami v 2.2.

```
    class MainBoardClass:
        def GetInfo(self):
        def GetAmbientTemp(self):
        def Set_HV_Status(self, status):
        def GetCardInsertion(self):
        def FanRequest(self, status, key):
        def GetModuleEEP(self):
        def SetIOpins(self, dat):
        def GetIOpins(self):
        def GetADCchannel(self, channel):
        def SetDACchannel(self, channel, voltage):
```

Vzor kódu 2.2: Minimální metody základní desky.

2.6.3 Softwarové moduly

V realizaci není možné pokrýt všechny případné periferie a rozšíření přístroje. Pro tyto nespecifická rozšíření slouží tyto moduly. Načítají se při startu systému a jsou opakovaně volány. Příkladem využití těchto modulů může být nastavení displeje, připojení přístroje k internetu nebo jiné komunikační rozhraní jako GPIB [7]. Tyto moduly jsou ve složce *IO_Modules* a mají přímou komunikaci s měřicím modulem. Objekt musí implementovat metody v 2.3.

```
class Module:
    def __init__(self, IO, MB, GUI, SR):
    def tick(self, Broadcast):
```

Vzor kódu 2.3: Minimální metody softwarového modulu.

2.6.4 Komunikace měřicího modulu s IO moduly

Měřicí modul má možnost přímé komunikace s IO moduly za pomocí *Broadcast*, což je slovník navrácený z metody *tick* měřicího modulu a je předán jako argument všem modulům. Tento slovník umožňuje efektivnější synchronizaci stavu měřicího modulu napříč SW. Pokud IO modul vyžaduje speciální komunikaci s měřicím modulem je využito rozhraní *SR*, které je předáno IO modulu jako argument při inicializaci a umožňuje zaslat měřicímu modulu textový řetězec a dostat další řetězec jako odpověď.

2.6.5 Kolize I²C sběrnice

V rozsahu I²C adres jsou předem alokované pouze adresy `0x50` a `0x51`. Během načítání modulů se následně alokují další adresy jednotlivým modulům, pro toto slouží třída *GlobalStructureClass* v *GlobalStructure.py*, která uchovává informace mezi moduly. Pro alokaci adresy je využita metoda *I2Cres*, která bere jako argument I²C adresu. V případě úspěšné alokace adresy pokračuje běh programu. Při kolizi je ukončen běh programu.

2.6.6 Start a běh systému

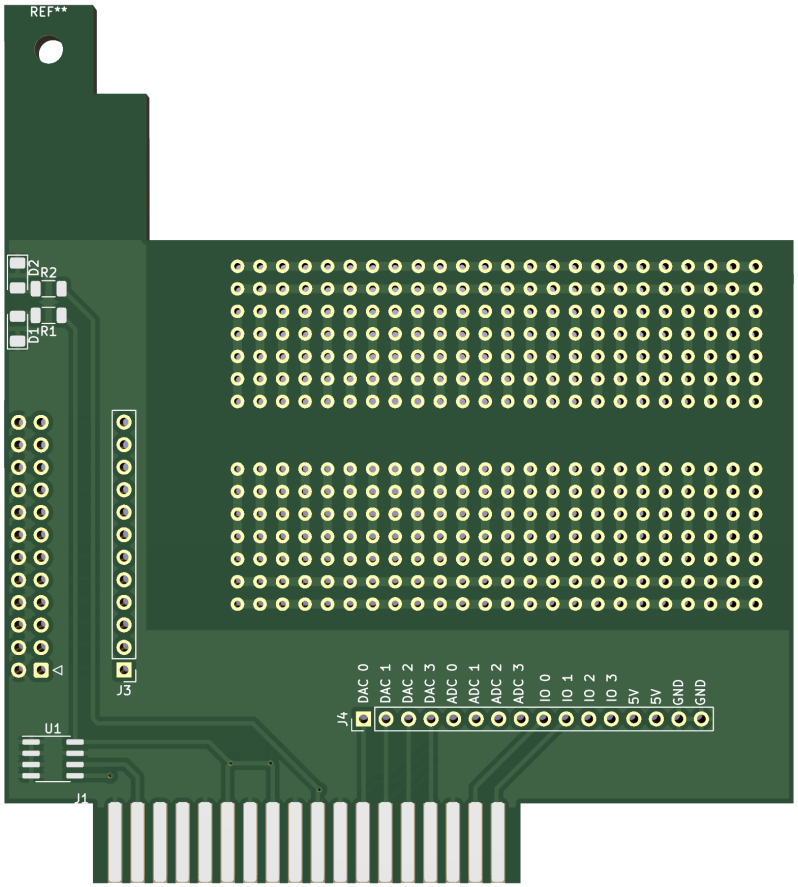
Při startu systému se spustí soubor *Init.py*, který inicializuje komunikaci s digitálními sběrnicemi inicializací souboru *IO_HAL.py*. Následně se přečte informace z EEPROM základní desky. Podle informací o základní desce se načte příslušný ovladač. Po načtení ovladače je již možné detekovat, jestli je vložen měřicí modul. Pokud je vložen měřicí modul, je inicializován objekt pro komunikaci implementovaný v *GlobalStructure.py*. Následně jsou načteny informace o měřicím modulu a připraveno grafické rozhraní. Poté se načte software měřicího modulu. Po načtení modulu se inicializují všechny softwarové moduly. Tím jsou načteny všechny potřebné moduly a je spuštěna smyčka měřicího přístroje, která opakovaně aktualizuje informace o všech modulech.

2.6.7 Klávesnice

V případě, že přístroj nedisponuje fyzickou klávesnicí, je možné využít softwarovou v *Keyboard.py*. Při implementaci této klávesnici jsem se zaměřil na možnost jednoduchého rozšíření o jiné znaky a možnost využít klávesnici bez blokování běhu programu.

2.7 Měřicí modul

Tento modul implementuje samotnou měřicí funkcionalitu přístroje za pomocí rozhraní základní desky. Samotné rozhraní je navrženo tak, že modul může určit jestli je provozován na 5 V, nebo 3,3 V pomocí propojených kontaktů. Rozhraní je přizpůsobeno pro pomalý přenos dat a v případě potřeby vysokorychlostního přenosu je možné využít samostatné výpočetní jednotky



Obrázek 2.2: Vizualizace desky pro prototypování měřicího modulu.

a následně zpracovaná data přesunout po rozhraní SPI. Modul nemusí využívat pouze analogové rozhraní, ale může být založen i na integrovaném obvodu s I^2C, nebo SPI rozhraním. Takovýto modul využívá převážně softwarových vlastností měřicího přístroje. Vzhledem k potenciálně velice odlišným funkcím přístroje není možné implementovat software pro všechny moduly. Proto je možné nahrát ke každému modulu do přístroje software pro řízení. Jednotlivé programy jsou rozlišeny podle názvu, který je uložen v EEPROM.

2.7.1 Časová náročnost výroby modulu

Samotný modul může mít velice odlišné nároky, pokud se ovšem zaměříme na jednoduchý případ, kdy je modul postaven kolem integrovaného obvodu, který má jako ovládání analogové napětí, je možné vytvořit tento modul na univerzální desce zobrazené na obrázku 2.2. Tento postup by znamenal jednoduché připojení, které může trvat přibližně hodinu. Následné vytvoření softwaru modulu obsahuje vytvoření GUI a implementace přepočtu hodnot získaných ze základní desky. Základní software modulu lze najít v 2.4.

```
import tkinter
class Module:
    def __init__(self, IO, MB, GUI, SR):
        self.MB = MB
        self.Frame = tkinter.Frame(GUI["Root"])
        self.GUI = GUI

        SR.SetCallback(self.callback, self)

        GUI["Mod_Main"] = self.Frame
        self.Frame.grid(row=0, column=0, sticky='wesn')
        GUI["MenuObj"].add_command(label="MODUL", command=
lambda:self.Frame.tkraise())

    def callback(SR, self, msg, source):
        return "OK" # Odpoved na komunikaci od modulu

    def tick(self):
        return {}
```

Vzor kódu 2.4: Minimální implementace programu modulu

Kapitola 3
Realizace platformy

Při realizaci platformy jsem se zaměřil na jednoduchost a univerzálnost přístroje. Jedním z požadavků bylo, aby přístroj mohl fungovat samostatně bez přídavného počítače. Samotná realizace přístroje je na obrázku 3.1. Soubory využité při realizaci a budoucích úpravách zařízení jsou dostupné na gitlab.fel.cvut.cz/hamacjan/bp.

3.1 Procesor

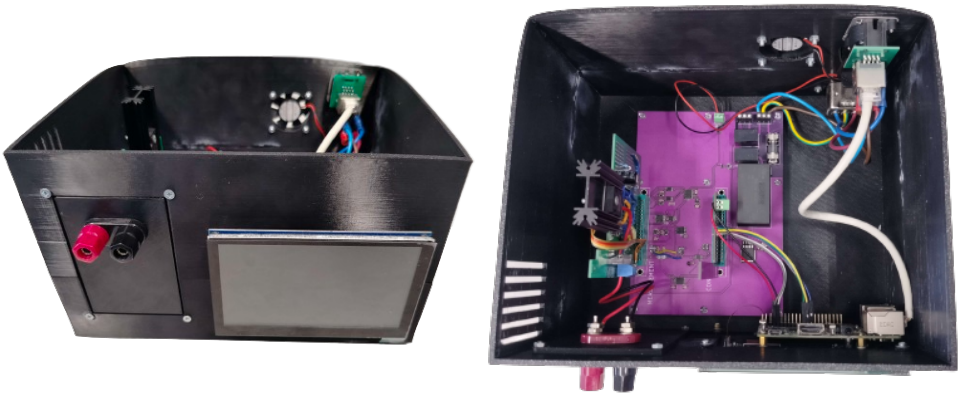
Jako procesor byl využit jednodeskový počítat Raspberry Pi kvůli dobré softwarové podpoře digitálního rozhraní. K počítači byl nadále připojen displej [8], který byl vybrán pro jednoduchost připojení díky připravené podpoře pro Raspberry Pi.

3.2 Základní deska

Základní deska je nejkomplikovanější částí systému. Výběr komponent zároveň silně ovlivnil nedostatek integrovaných obvodů na trhu a dostupnost byla hlavním kritériem při výběru, pokud není v textu specifikováno jinak. Samotný modul implementuje všechny požadované funkce pro funkci přístroje. Při návrhu základní desky byl kladen důraz na přehlednost umístěných komponent, který je vidět na obrázku 3.2. Samotnou desku by bylo možné navrhnout na výrazně nižších rozměrech, ovšem to by limitovalo případné funkce pokročilejších implementací.

3.2.1 Analogové převodníky

Pro teplotní stabilizaci analogových výstupu byla použita externí napěťová reference *MCP1501-40xCH* [9], která dodává 4,096 V ± 0,1 % s teplotním koeficientem 50 ppm/°C. Jako A/D převodník byl vybrán *MAX1236* [10] díky vstupu pro čtyři kanály, I^2C rozhraní a 12 b rozlišení. Tento převodník byl aktuálně nejlepší dostupnou volbou, není ovšem optimální. Lepší volbou by byl převodník s externím referenčním napětím. Tento nedostatek byl ovšem vyřešen napájením převodníku přímo z reference. Pro D/A převod byl vybrán



Obrázek 3.1: Realizovaný přístroj.

převodník *DAC6573* [11], který vyhovoval díky čtyřem kanálům, vstupem pro externí referenci, rozhraní I^{2}C a 10 b rozlišení.

3.2.2 Digitální signály

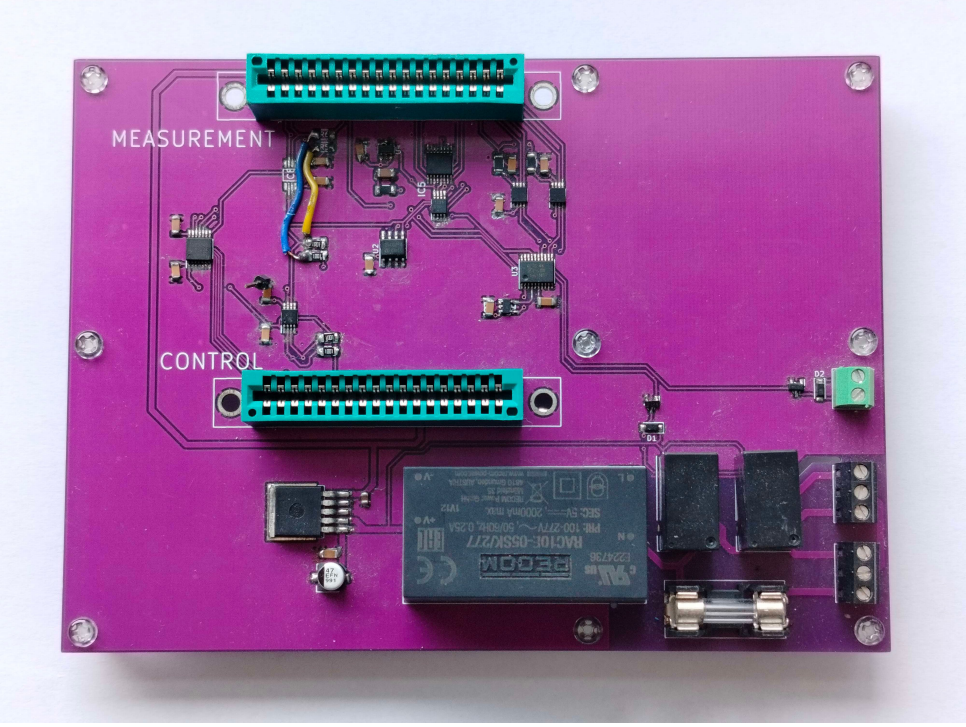
Modul je řízen primárně rozhraním I^{2}C, což ovlivňovalo volbu součástek. Většina převodníků pracuje na 5 V, ovládací modul ovšem vyžaduje 3,3 V, pro převod byl tedy využit I^{2}C převodník *PCA9517ADMR2G* [12]. Pro převod sběrnice SPI byl použit převodník *TXU0304QPWRQ1* [13]. Digitální rozhraní implementuje integrovaný obvod *PCF8574ATS* [14], který byl vybrán pro možnost obousměrného přenosu, nastavení adresy a dostatečného počtu kanálů. Dále je na I^{2}C sběrnici připojen integrovaný obvod *TC74A2* [5] implementující snímání teploty okolí. Na I^{2}C sběrnici je také připojena potřebná EEPROM *M24C02-FMN* [6].

3.2.3 Napájení

Pro napájení přístroje ze síťového napětí byl použit integrovaný modul *RAC10E-05SK_277* [15]. Tento modul dodává 5 V s výkonem až 10 W. Z této úrovně je napájen ovládací modul, počítač a další integrované obvody. Následně je toto napětí regulované na $qty3.3V$. Tento výstup se používá pouze pro integrované obvody a není potřeba tedy dodávat významné výkony, proto byl pro jednoduchost zvolen lineární regulátor *850B0V33* [16]. Následně pro funkci spínání byly využity dvě relé schopné dodávat 16 A. Spínání těchto relé je zajištěno mosfetem ovládaným digitálním I^{2}C rozhraním.

3.3 Ovládací modul

V této realizaci byl pro ovládání vybrán jednodeskový počítač Raspberry Pi [17]. Tento počítač umožňuje komunikaci po hardwarovém rozhraní I^{2}C a SPI. Tyto požadavky splňují i jiné jednodeskové počítače, které ale mají výrazně horší dokumentaci a kompatibilitu s jinými výrobky. Deska tedy obsahuje



Obrázek 3.2: Výsledná realizace základní desky.

pouze připojení napájení sběrnice na 3,3 V a vyvedení konektorů pro napájení a digitální rozhraní.

3.4 Softwarová implementace

Software implementuje všechny potřebné třídy pro chod programu, ovládání přístroje a komunikaci.

3.4.1 Grafické rozhraní

Grafické rozhraní využívá knihovny *Tkinter*. Tato knihovna je dostupná pro většinu Unixových systémů, MacOS a Windows [18]. To umožňuje spustit řídící program i na počítači jiném než je Raspberry pi, pokud je implementované fyzické rozhraní. Přístup ke kořenovému objektu grafického rozhraní mají všechny moduly přes objekt *GUI["Root"]*, což dovoluje jednotlivým modulům vytvářet vlastní okna a přepínat mezi nimi. Tento krok byl zároveň zvolen při návrhu samotného grafického rozhraní, kdy je grafické rozhraní členěné do jednotlivých oken, která se přepínají pomocí menu na horní hraně obrazovky. Toto menu v základním nastavení obsahuje pouze výchozí okno, které obsahuje informaci o načtené základní desce a měřicím modulu. Příklad grafického rozhraní je vidět na obrázku 3.3, kde je zobrazeno okno měřicího modulu, který umožňuje základní ovládání funkcí základní desky. Zároveň byl načten modul pro ovládání jasu displeje a pro ovládání přes rozhraní

Obrázek 3.3: Příklad grafického rozhraní při připojení modulu pro ovládání základní desky.

Ethernet. Grafické rozhraní bylo navrženo bez použití fyzické klávesnice kvůli zvolenému dotykovému displeji.

3.4.2 Softwarové moduly

V této verzi bylo implementováno několik softwarových modulů, které využívaly již existující komponenty přístroje, jako je připojení přes Ethernet nebo LCD displej.

LCD Displej

Tento modul implementuje možnost nastavení jasu u připojeného kompatibilního displeje. V případě připojení nekompatibilního displeje modul neregistruje své metody a není tedy načten. Samotné ovládání jasu je realizované přes systémový soubor *'/sys/class/backlight/10-0045/brightness'*, tento krok ovšem vyžaduje spuštění programu se zvýšeným oprávněním pro možnost přístupu k hardware.

Rozhraní Ethernet

Tento modul umožňuje komunikaci po síti Ethernet na portu *1234*. Z důvodu abstrakce komunikace s měřicím modulem není využit speciální protokol, ale jsou přijatá data přeposílána měřicímu modulu na komunikační rozhraní. Při odesílání dat ovšem přidává další paket obsahující data o stavu modulu.

Součást	Částka
Komponenty	3400 Kč
Tištěné spoje	100 Kč
Displej	1249 Kč
Řídící počítač	1099 Kč
Jiné	300 Kč
Celkem	6148 Kč

Tabulka 3.1: Náklady na realizaci přístroje

3.5 Náklady

Celkové náklady na přístroj zahrnovaly desky tištěného spoje, komponenty, řídící počítač a displej. Některé součástky byly dostupné pouze ve vyšším množství a byly následně přepočítány na jeden kus. Z důvodu nedostatku na trhu nebylo možné získat nový počítač Raspberry Pi a byl využit starší, ale kvůli pro nové návrhy je uvedena částka aktuálního modelu [19]. Samotné náklady na výrobu jednoho přístroje bez přístrojové krabice jsou 6148 Kč, kde jednotlivé položky jsou v tabulce 3.1.

3.6 Opravy ve schématech

Poté co byl přístroj vyroben podle přiložených návrhů jsem našel nedostatky, které byly dodatečně opraveny. Potřebné změny jsou zmíněny níže.

Zemnění

Na základním modulu byla špatně nastavena výplň spodní vrstvy země. Bylo tedy potřeba přidat vodič, který vede od kontaktu záporného pólu napájecího zdroje k montážnímu otvoru pro řídící modul.

I²C

Na základní desce byly osazeny rezistory pro správné fungování sběrnice s hodnotou 10 kΩ. To způsobilo nefunkční I^2C sběrnici kvůli nízké rychlosti návratu signálu. Z tohoto důvody byly vyměněny za rezistory s hodnotou 1 kΩ.

Převodníky napětí

V návrhu platformy je definováno, že jednotlivé moduly můžou své napětí nastavit připojením napětí na převodníky úrovně. Tento krok byl realizován. Při vybrání byl speciálně zvolen převodník, který nezakazoval stejné napětí na obou kanálech. Při realizaci se ovšem ukázalo, že tento převodník nefunguje, pokud má stejné napětí na obou kanálech. Základní deska využívá napětí

5 V, byl proto zanechán převodník k ovládacímu modulu, který funguje na 3,3 V a odebrán převodník pro I^2C sběrnici u měřicího modulu a je tedy nutné využívat 5 V měřicí moduly. Do budoucna by bylo možné tuto funkci realizovat galvanicky oddělenými převodníky, které by tento problém neměly.

Kapitola 4

Realizace měřicího modulu

V této kapitole je popsán návrh měřicího modulu využívající základní funkce přístroje. Samotný modul je na obrázku 4.2.

4.1 Teoretický rozbor

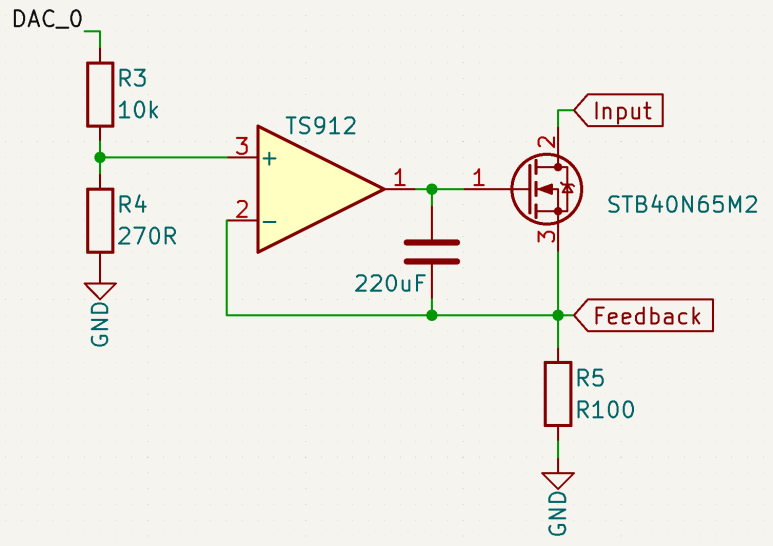
Zapojení využívá principu záporné zpětné vazby pro regulaci vstupního proudu. Zapojení regulující proud je zobrazeno na obrázku 4.1.

4.1.1 Regulační smyčka

Hlavní regulační smyčka obsahuje rezistor R_5 o hodnotě 100 mΩ, který se používá pro snímání proudu. Toto napětí je následně pomocí operačního zesilovače porovnáno se zpracovaným nastaveným napětím z DAC_0. Výstup operačního zesilovače je použit pro otevírání tranzistoru STW40N65M2. Jelikož je při plném otevření na rezistor R_5 napětí 100 mV a výstup operačního zesilovače při napájení 5 V může jít až do napětí 4,98 V [20], je možné otevřít tranzistor napětím $U_{GS} = 4{,}88$ V, které dostatečně otevře tranzistor [21]. Do regulační smyčky je přidán i kondenzátor, který nastaví zesílení pro střídavé složky na G = 1. Jelikož je střídavá složka U_{DAC_0} zanedbatelná, tranzistor by měl fungovat pouze v DC režimu. Výsledný proud je závislý na neinvertujícím vstupu operačního zesilovače. Toto napětí je regulováno na R_5 a výsledný proud je

$$I = \frac{U_+}{R_5} = \frac{U_+}{100\,\text{m}\Omega} = U_+ \cdot 10\,\text{S}. \tag{4.1}$$

Rozsah DAC_0 je 0 V až 4 V. Ovšem při rozsahu 1 A z obrázku 4.1 vychází, že U_+ je v rozsahu 0 V až 0,1 V. Toto silně limituje rozlišení nastaveného proudu při přímém zapojení a proto je použit napěťový dělič na vstupu, který sníží rozsah na DAC_0 až 105 mV. Maximální napětí vstupu ADC_0 je 4 V. Toto silně limituje jaké obvody můžeme zatěžovat. Proto je na vstupu U_{ADC_0} napěťový dělič. Pro měření proudu je využit rezistor R_5 z regulační smyčky. Při plném rozsahu 1 A je napětí na tomto rezistor pouze 100 mV. Proto je přidán neinvertující zesilovač zvyšující rozsah signálu na 0 V až 4 V.

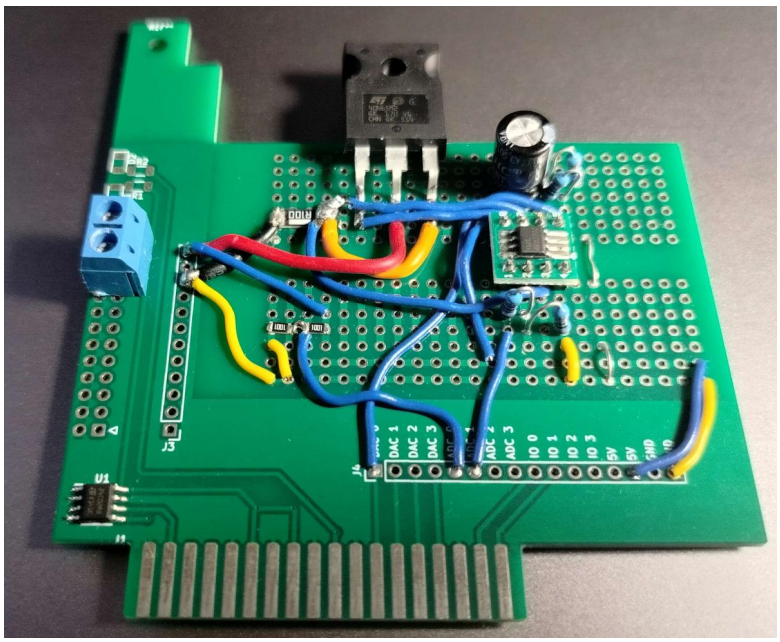


Obrázek 4.1: Regulace konstantního proudu.

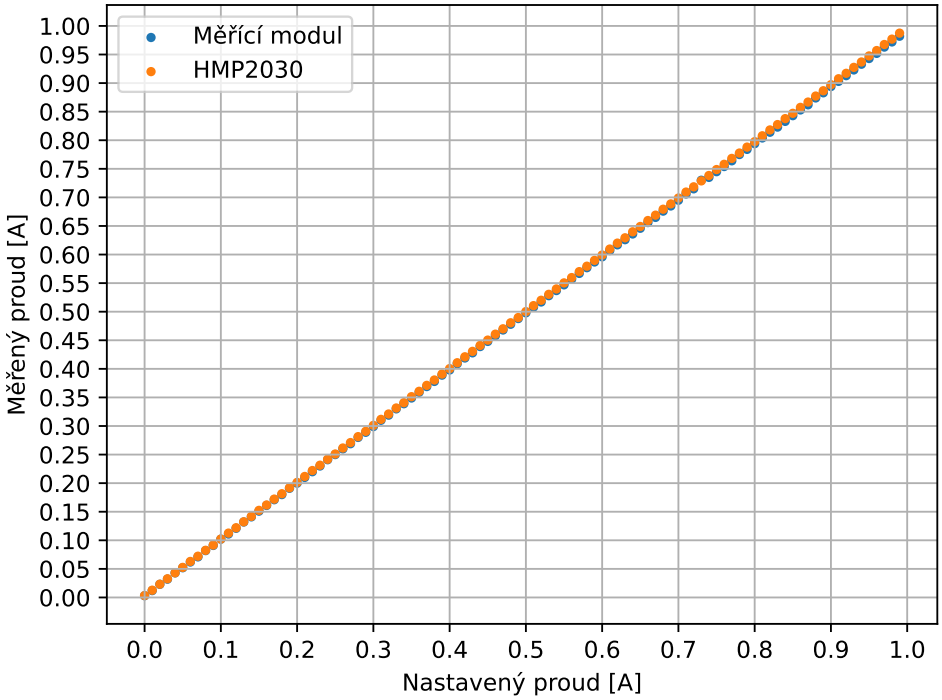
4.2 Implementace

Samotný modul byl realizován na prototypové desce měřicího modulu podle schémat v sekci 4.1 s výměnou operačního zesilovače za OPA2388 [20]. Výsledná realizace je zobrazena na obrázku 4.2. Dále jsem připevnil na chladič výkonového tranzistoru čidlo teploty TC74 [5], které v případě zvýšení teploty aktivuje ventilátor.

Ověření funkce bylo prováděno pomocí napájecího zdroje Rohde & Schwarz HMP2030, který byl zatěžován testovaným modulem. Tento zdroj byl použit i pro měření výstupního napětí a proudu. V obrázku 4.3 je zobrazena charakteristika proudu naměřeného zdrojem a modulem v závislosti na nastaveném proudu. Maximální odchylka mezi měřenými proudy byla 3 mA.



Obrázek 4.2: Realizace výstupního modulu



Obrázek 4.3: Proud měřený jednotlivými přístroji v závislosti na nastaveném proudu.

Kapitola 5

Závěr

V této práci jsem navrhl platformu, která umožňuje jednoduchou realizaci měřicích modulů. Platforma je navržena tak, aby minimalizovala duplikování funkčních bloků. Přístroj umožňuje připojení měřicích modulů, na které jsou kladené minimální požadavky. To vyžaduje vyšší nároky na základní desky, které je ovšem možné specifikovat podle nároků konkrétní aplikace. Výsledkem byla možnost vytvořit základní verzi platformy s potřebou připojení k PC v cenové úrovni 4000 Kč, nebo rozšířenou realizaci jako samostatně fungující přístroj za 6148 Kč. Díky přesunu většiny potřebných komponent mimo měřicí modul je možné realizovat měřicí modul s minimálními náklady na čas a komponenty, což umožňuje realizovat měřicí modul v řádu jednotek hodin. Tyto vlastnosti jsou podporované návrhem software přístroje, který využívá principů objektového programování pro přizpůsobení softwaru konkrétní hardwarové implementaci. Tento modulární přístup také umožňuje využití pro výuku měřicích technik a principu měřicích přístrojů.

Na základě této platformy jsem následně realizoval měřicí přístroj, který nevyžaduje připojení k PC. Tento přístroj dodává měřicímu modulu rozsah analogových kanálů od 0 V do 4 V s rozlišením pro A/D převod 1 mV a pro D/A 4 mV. Přístroj dále umožňuje funkce chlazení a spínání síťového napětí.

Pro tento přístroj jsem následně realizoval měřicí modul konstantní proudové zátěže, která je schopna odebírat proudy v rozsahu 0 A až 1 A při vstupním napětí 0 V až 8 V.

Na základě těchto výsledků usuzuji, že se podařilo navrhnout přístroj efektivní pro jednorázová měření konstantních veličin i pro výuku měřicích přístrojů.

Bibliografie

1. INSTRUMENTS, Liquid. *Moku:Go datasheet* [online `https://download.liquidinstruments.com/documentation/datasheet/hardware/mokugo/Datasheet-MokuGo.pdf`]. [B.r.]. zobrazeno 11.05.2023.
2. PITAYA, Red. *STEMlab 125-14* [online `https://redpitaya.com/product/stemlab-125-14/`]. [B.r.]. zobrazeno 11.05.2023.
3. FAKULTA ELEKTROTECHNICKÁ, Katedra měření. *Little Embedded Oscilloscope* [online `https://embedded.fel.cvut.cz/platformy/leo`]. [B.r.]. zobrazeno 11.05.2023.
4. HLADÍK, Jiří. *Single chip software defined instrumentation for educational purposes* [online `https://poster.fel.cvut.cz/poster2017/proceedings/Poster_2017/Section_EI/EI_015_Hladik.pdf`]. Fakulta elektrotechnická, Katedra měření, 2017.
5. *TC74 Datasheet*. Microchip Technology Incorporated, 2012. ISBN 9781620768396. Rev. D.
6. *M24C02-DRE Datasheet*. STMicroelectronics, 2017. Rev. 2.
7. IEEE Standard Codes, Formats, Protocols, and Common Commands For Use with ANSI/IEEE Std 488.1-1987 IEEE Standard Digital Interface for Programmable Instrumentation. *ANSI/IEEE Std 488.2-1987*. 1988, s. 1–206. Dostupné z DOI: `10.1109/IEEESTD.1988.120316`.
8. ELECTRONICS, Waveshare. *pěti palcový DSI LCD* [online `https://www.waveshare.com/wiki/5inch_DSI_LCD`]. [B.r.]. zobrazeno 11.05.2023.
9. *MCP1501-40xCH Datasheet*. Microchip Technology Incorporated, 2022. Rev. G.
10. *MAX1236 Datasheet*. Analog Devices, Inc, 2010. Rev. 7.
11. *DAC6573 datasheet*. Texas Instruments, 2003.
12. *PCA9517A Datasheet*. Semiconductor Components Industries, LLC, 2018. Rev. 2.
13. *TXU0304-Q1 Datasheet*. Texas Instruments, 2022. Rev. B.
14. *PCF8574ATS Datasheet*. NXP Semiconductors, 2013. Rev. 5.

15. *RAC10E-K/277 Datasheet*. RECOM Power GmbH, 2022. Rev. 0.

16. *TLS850B0TBV33 Datasheet*. Infineon Technologies AG, 2017. Rev. 1.

17. FOUNDATION, Raspberry Pi. *Raspberry Pi 2 Model B* [online `https://www.raspberrypi.com/products/raspberry-pi-2-model-b/`]. 11. 05. 2023.

18. FOUNDATION, Python Software. *tkinter — Python interface to Tcl/Tk* [online `https://docs.python.org/3/library/tkinter.html`]. [B.r.]. zobrazeno 09.05.2023.

19. *RPishop.cz Raspberry Pi 4 Model B* [online `https://rpishop.cz/raspberry-pi-4/1600-RPI401-7657569311684.html`]. [B.r.]. zobrazeno 09.05.2023.

20. *operational amplifier*. Texas Instruments, 2020. Č. OPA2388. Rev. D.

21. *Power MOSFET*. STMicroelectronics, 2015. Č. STW40N65M2. Rev. 1.

Příloha A

Seznam zkratek

A/D Analogově-digitální převodník.

D/A Digitálně-analogový převodník.

EEPROM Electrically Erasable Programmable Read-Only Memory.

GUI Graphic User Interface.

I^{2}C Inter-Integrated Circuit.

IO Input Output.

SDI Software Defined Instrument.

SPI Serial Peripheral Interface.

UART Universal asynchronous receiver-transmitter.

Příloha B

Seznam příloh

- `Příloha.zip`
  - `HW` *Tato složka obsahuje mechanický návrh přístroje*
    - `Mechanický návrh.FCStd`
    - `Tištěné spoje`
      - `ControlSCH.pdf` *Schéma ovládacího modulu k Raspberry Pi*
      - `ControlPCB.pdf` *Návrh desky plošného spoje ovládacího modulu*
      - `MainSCH.pdf` *Schéma základní desky*
      - `MainPCB.pdf` *Návrh desky plošného spoje základní desky*
      - `DebugSCH.pdf` *Schéma protypového modulu*
      - `DebugPCB.pdf` *Návrh desky plošného spoje protypového modulu*

- `Příloha.zip`
  - `SW` *Tato složka obsahuje software přístroje*
    - `IO_Modules` *Složka obsahující softwarové moduly*
      - `__init__.py`
      - `DSI_LCD.py` *Modul pro nastavení jasu DSI displeje*
      - `remote.py` *Modul pro připojení k Ethernet*
    - `Out_Modules` *Složka měřicích modulů*
      - `__init__.py`
      - `ConstantCurrentLoad_01_00_00.py` *Testovací modul*
      - `GeneralIOBoard_01_00_00.py` *Modul konstantní proudové zátěže*
    - `Main_Boards` *Složka se SW základních desek*
      - `__init__.py`
      - `MainBoard_01_00_00.py`
    - `Utils` *Obsahy EEPROM jednotlivých modulů*
      - `MainBoardData.hex`
      - `ConstantCurrentLoad.hex`
      - `IOBoardData.hex`
    - `GlobalStructure.py` *Třída pro komunikaci mezi moduly*
    - `Init.py` *Program spouštějící přístroj*
    - `IO_HAL.py` *Třída implementující rozhraní s HW*
    - `Keyboard.py`
    - `Parsers.py` *Třída zpracovávající data v EEPROM*