



# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

## FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

## ÚSTAV RADIOELEKTRONIKY

DEPARTMENT OF RADIO ELECTRONICS

## THD ANALYZÁTOR

THD ANALYZER

### BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

### AUTOR PRÁCE

AUTHOR

Tuan Ha

### VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. Jan Mikulka, Ph.D.

BRNO 2018

# Bakalářská práce

bakalářský studijní obor **Elektronika a sdělovací technika**  
Ústav radioelektroniky

**Student:** Tuan Ha

**ID:** 174497

**Ročník:** 3

**Akademický rok:** 2017/18

**NÁZEV TÉMATU:**

## THD analyzátor

### POKYNY PRO VYPRACOVÁNÍ:

V rámci bakalářské práce proveďte průzkum metod pro analýzu celkového harmonického zkreslení. Zvolte metodu vhodnou pro implementaci do jednodeskového PC (např. Raspberry Pi). Proveďte návrh softwarové a hardwarové části pro analýzu THD signálů v audio oblasti. Následně proveďte realizaci THD analyzátoru jako autonomního přístroje. Přístroj bude vizualizovat spektrum analyzovaného signálu a zobrazovat míru celkového harmonického zkreslení. Na závěr proveďte analýzu vybraných signálů a srovnajte změřené THD s výsledky simulací.

### DOPORUČENÁ LITERATURA:

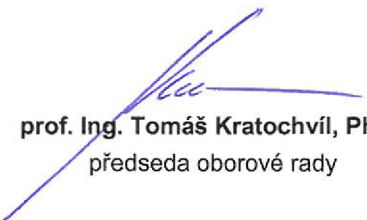
[1] Sajjad, K et al. Simplified THD measurement and analysis for electronic power inverters. Applied Sciences and Technology (IBCAST), 2015 12th International Bhurban Conference.

[2] Shmilovitz, D. On the definition of total harmonic distortion and its effect on measurement interpretation. IEEE Transactions on Power Delivery, 2005.

**Termín zadání:** 5. 2. 2018

**Termín odevzdání:** 24. 5. 2018

**Vedoucí práce:** doc. Ing. Jan Mikulka, Ph.D.

  
**prof. Ing. Tomáš Kratochvíl, Ph.D.**  
předseda oborové rady



### UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č.121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

## **ABSTRAKT**

Tato práce se zabývá návrhem *THD* analyzátoru pro audio kmitočty. Na úvod jsou popsány metody výpočtu *THD* a způsob realizace *THD* analyzátoru. Pro návrh *THD* analyzátoru je zvolena vhodná platforma schopná zpracovávat audio data. K ní je navržená rozšiřující deska, schopná snímání audio signálu. Součástí návrhu je i kompletní tvorba programu, jenž zpracovává data z AD převodníku navržené desky. Výstupem je autonomní měřicí přístroj schopný zobrazení spektra a kalkulace *THD*.

## **KLÍČOVÁ SLOVA**

AD převodník, Raspberry Pi 3, celkové harmonické zkreslení, spektrální analyzátor, antialiasingový filtr, SPI sběrnice, Fourierova transformace

## **ABSTRACT**

This thesis deals with design of a *THD* analyzer for audio frequency. In the interduction, there is a description of methods of *THD* calculation and realization of *THD* analyzer. Suitable computing platform must be chosen with capability to process audio data. The expansion board is designed for audio acquisition. Part of this design is complete program that is capable of displaying spectrum and computing *THD*.

## **KEYWORDS**

AD converter, Raspberry Pi 3, total harmonic distrotion, spectral analyzer, antialiasing filter, SPI bus, Fourirer transform

HA, Tuan. *THD analyzátor*. Brno, 2018, 69 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky. Vedoucí práce: doc. Ing. Jan Mikulka, Ph.D.

## PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „THD analyzátor“ jsem vypracoval(a) samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor(ka) uvedené bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské práce jsem neporušil(a) autorská práva třetích osob, zejména jsem nezasáhl(a) nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom(a) následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno .....

.....

podpis autora(-ky)

## PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu doc. Ing. Janu Mikulkovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno .....

.....

podpis autora(-ky)

# OBSAH

Úvod	11
<b>1 Teoretická část</b>	<b>12</b>
1.1 THD . . . . .	12
1.2 Metoda měření . . . . .	13
1.3 Simulace v prostředí Matlab . . . . .	15
1.4 Výběr platformy . . . . .	20
1.5 A/D převodník . . . . .	23
1.6 Předzesilovač . . . . .	24
<b>2 Návrhy</b>	<b>26</b>
2.1 Anti-aliasingový filtr . . . . .	26
2.2 Napájení . . . . .	29
2.3 Design desky . . . . .	30
<b>3 Software</b>	<b>36</b>
3.1 OS Raspbian . . . . .	36
3.2 Python - knihovny . . . . .	36
3.3 Struktura softwaru . . . . .	38
3.4 Zpracování dat . . . . .	41
3.5 Výpočet THD . . . . .	42
<b>4 Výsledky</b>	<b>48</b>
4.1 Kmitočtový rozsah . . . . .	48
4.2 Dynamický rozsah . . . . .	50
4.3 Stanovení odchylky . . . . .	51
<b>5 Závěr</b>	<b>54</b>
<b>Literatura</b>	<b>55</b>
<b>Seznam symbolů, veličin a zkratk</b>	<b>57</b>
<b>Seznam příloh</b>	<b>58</b>
<b>A Skript na simulaci v prostředí Matlab</b>	<b>59</b>
<b>B Návrh desky plošných spojů</b>	<b>60</b>
<b>C Fotografie desky</b>	<b>67</b>
<b>D Seznam součástek</b>	<b>68</b>



# SEZNAM OBRÁZKŮ

1.1	Zkreslení signálu nelineárním systémem . . . . .	12
1.2	Periodizace spektra a AAF . . . . .	14
1.3	Blokové schéma spektrálního analyzátoru . . . . .	15
1.4	Průběhy jednotlivých kmitočtů . . . . .	16
1.5	Průběh testovacího signálu . . . . .	16
1.6	Spektrum testovacího signálu . . . . .	17
1.7	Spektrum signálu pro délku vzorku 0,1 s . . . . .	19
1.8	Spektrum signálu pro délku vzorku 0,2 s . . . . .	19
1.9	Spektrum signálu pro délku vzorku 1 s . . . . .	20
1.10	Raspberry Pi 3 [15] . . . . .	21
1.11	RPi3 - rozložení GPIO [16] . . . . .	22
1.12	Schematické rozložení pinů AD7276 - pohled shora [4] . . . . .	23
1.13	Průběh komunikace normálního chodu ADC [4] . . . . .	24
1.14	Schematické rozložení pinů zesilovače [6] . . . . .	25
1.15	Frekvenční charakteristika zesilovače [6] . . . . .	25
2.1	Návrh tolerančního pole . . . . .	27
2.2	Obvod Sallen-Key . . . . .	28
2.3	Schéma rozvodu napájení . . . . .	30
2.4	Zdvih stejnosměrné složky . . . . .	31
2.5	obvodové zapojení zesilovače . . . . .	32
2.6	obvodové zapojení zesilovače . . . . .	32
2.7	obvodové zapojení zesilovače . . . . .	33
2.8	horní strana desky . . . . .	34
2.9	spodní strana desky . . . . .	35
3.1	Průběh CS (žlutě) a hodinového signálu (zeleně) knihovny pigpio . . . . .	37
3.2	Grafické rozhraní - THD analyzátoru (simulovaný signál s THD 5.2%) . . . . .	38
3.3	Vývojový diagram programu . . . . .	40
3.4	Grafické znázornění zpracování vektorů pro výpočet <i>THD</i> . . . . .	44
3.5	Grafické znázornění detekce amplitudy . . . . .	45
3.6	Časový a spektrální průběh při délce vzorků 524 288 . . . . .	46
3.7	Časový a spektrální průběh při délce vzorků 600 000 . . . . .	47
4.1	Navzorkovaný signál 7 kHz - časová oblast . . . . .	49
4.2	Navzorkovaný signál 200 kHz - časová oblast . . . . .	49
4.3	Navzorkovaný sinusový signál 150 kHz - časová a kmitočtová oblast . . . . .	50
4.4	Úroveň šumu (na ose y již nezbylo místo na zobrazení první číslice - hodnota 00006 [-] odpovídá hodnotě 0,00006 [-]) . . . . .	51
B.1	Spodní strana desky - osazovací výkres . . . . .	60
B.2	Spodní strana desky - plošný spoj . . . . .	61
B.3	Horní strana desky - osazovací výkres . . . . .	62
B.4	Horní strana desky - plošný spoj . . . . .	63



B.5	strana 1 (signálová část) . . . . .	64
B.6	strana 2 (signálová část) . . . . .	65
B.7	strana 1 (napájecí část) . . . . .	66
C.1	Horní strana desky - fotografie . . . . .	67
C.2	Spodní strana desky - fotografie . . . . .	67

## SEZNAM TABULEK

1.1	Srovnání modelů Raspberry Pi . . . . .	22
3.1	<i>THD</i> pro různé délky intervalu . . . . .	45
4.1	Srovnání <i>THD</i> generovaného a simulovaného signálu . . . . .	48
4.2	Výpočet odchylek pro různé střídy signálu . . . . .	51
D.1	Seznam součástí . . . . .	68

# ÚVOD

Velké  $THD$ <sup>1</sup> způsobené nelinearitou systému má vliv na účinnost zařízení, na elektromagnetické rušení okolních zařízení a na kvalitu audio zvuku. Tato práce se zabývá návrhem přístroje schopného měřit  $THD$  pro audio kmitočty.

V první kapitole je vysvětlena veličina  $THD$ . Zde jsou popsány teorie výpočtu a rozdíl mezi  $THD_R$  a  $THD_F$ . Je zde probrána metoda měření a naznačeno blokové schéma přístroje. K jednotlivým blokům jsou zvoleny vhodné integrované obvody. Důležitou částí návrhu je výběr správného druhu platformy, jako je například Raspberry Pi 3, Raspberry Pi 2 nebo Raspberry Pi zero, která vyhovuje z hlediska ceny, periférií, kmitočtu procesoru atd. Žádná z vyjmenovaných platform nemá periferii s AD převodníkem. Je tedy navrhována rozšiřující deska, jejíž součástí je AD převodník (dále jen ADC<sup>2</sup>), zesilovač, anti-aliasingový filtr (dále jen AAF<sup>3</sup>) a rozvod napájení. Ke komunikaci s řídicí platformou je využita sériová komunikace.

Druhá kapitola je věnována návrhu desky. Zde jsou probrány jednotlivá obvodová zapojení a návrh napájení obvodu. Tato kapitola také obsahuje návrh desky plošných spojů.

Kapitola Software se zabývá tvorbou programu. Program je navržen tak, aby byl řízen pouze přes dotykový displej. Jsou zde vybrány vhodné knihovny pro zpracování dat a jejich vykreslení. Tato data jsou dále zpracovávána algoritmem, jehož důležité bloky jsou popsány.

Na závěr je stanovena přesnost navrženého měřicího přístroje.

---

<sup>1</sup>Total harmonic distortion - celkové harmonické zkreslení ( $THD$ )

<sup>2</sup>Analog to digital converter - AD převodník (ADC)

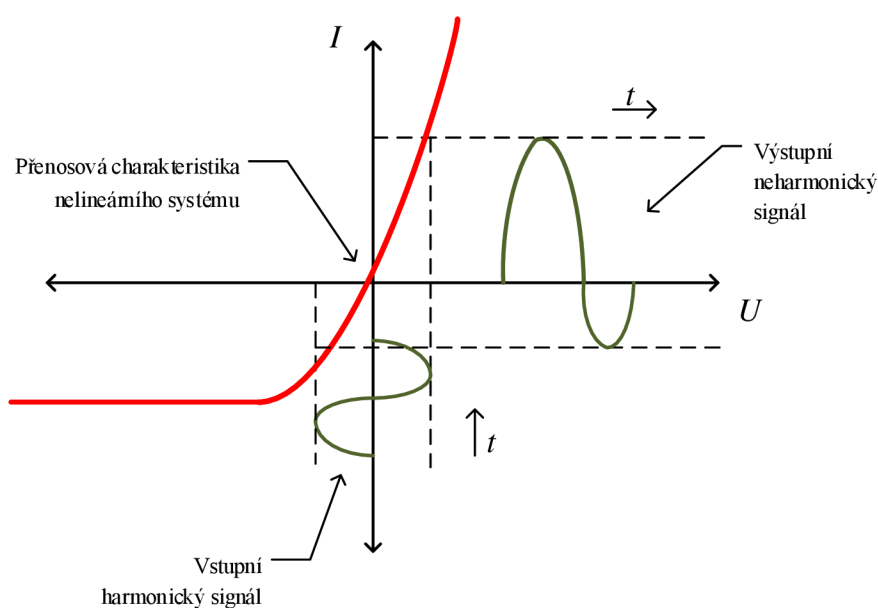
<sup>3</sup>Antialiasing filter - Antialiasingový filtr (AAF)

# 1 TEORETICKÁ ČÁST

## 1.1 THD

Již v úvodu byl zmíněn význam veličiny *THD* a jeho vliv na elektrické zařízení. V této kapitole je tento pojem vysvětlen a dále jsou také vysvětleny způsoby jeho měření

*THD* udává poměr amplitud mezi vzniklými harmonickými složkami a základním kmitočtem. Harmonické složky jsou násobky základního kmitočtu, které vznikají po průchodu nelineárním systémem (viz obr. 1.1). Z harmonického signálu se rázem stává signál peri-



Obr. 1.1: Zkreslení signálu nelineárním systémem

odický neharmonický. Jednotlivé složky zkresleného signálu je možné pozorovat v kmitočtovém spektru. Je tedy nutné převést signál z časové do kmitočtové oblasti. K tomu nám poslouží algoritmus Fourierovy transformace (FT<sup>1</sup>). Pro signál  $s(t)$  je Fourierova transformace daná vztahem:

$$S(\omega) = \int_{-\infty}^{\infty} s(t)e^{-j\omega t} dt. \quad (1.1)$$

<sup>1</sup>Fourier transform - Fourierova transformace (FT)

Fourierova věta nám říká, že periodickou neharmonickou funkci lze rozložit na nekonečnou řadu harmonických složek, jejichž kmitočty jsou dány celistvými násobky dané funkce. Algoritmus FT této vlastnosti využívá a rozkládá signál na jednotlivé komponenty sinů a cosinů, které jsou v kmitočtové oblasti zobrazené jako spektrální čáry.

Pro  $THD$  existují dvě definice -  $THD_F$  a  $THD_R$ . První hodnota je definována jako podíl součtu efektivních hodnot harmonických složek ku efektivní hodnotě základní složky. První definice je dána vztahem:

$$THD_F = \frac{\sqrt{\sum_{n=2}^{\infty} U_n^2}}{U_1}, \quad (1.2)$$

kde  $U_n$  - je RMS hodnota daného  $n$ -té harmonické složky napětí (lze místo proudu dosadit i hodnoty proudu). Pro  $THD_R$  platí:

$$THD_R = \sqrt{\frac{\sum_{n=2}^{\infty} U_n^2}{\sum_{n=1}^{\infty} U_n^2}}. \quad (1.3)$$

Rozdíly mezi těmito dvěma definicemi je při malých hodnotách zkreslení nepatrný. Například pro  $THD_F = 10 \%$  může být  $THD_R$  rovno 9,95 %. Z uvedených vztahů plyne, že hodnoty  $THD_R$  nabývají maximální hodnoty 100 % a  $THD_F$  může, pro extrémní případy, nabývat hodnot vyšších jak 100 %. K zavedení  $THD_R$  pravděpodobně došlo kvůli jeho jednoduché implementaci. Čitatel  $THD_R$  vznikne jednoduše odfiltrováním základní složky, pomocí Notchfiltru (úzkopásmový filtr), ze jmenovatele [1].

## 1.2 Metoda měření

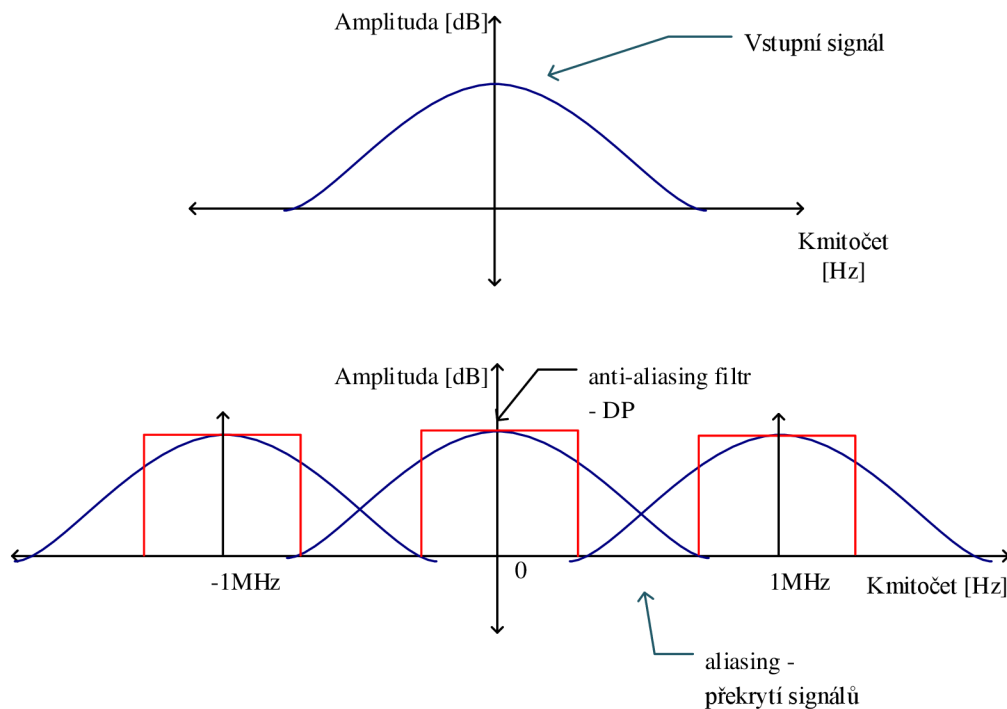
Navrhovaný měřicí přístroj je v podstatě spektrální analyzátor. Ty se dělí na dvě skupiny, a to na analyzátor, jejichž výpočet spektra probíhá přes FFT (vysvětleno v následujících kapitolách), a na analyzátor analogové, jejichž spektrum se získává pomocí filtrační metody. Analogové spektrální analyzátor mají větší rozsah měřitelného pásma (až několik GHz) [3]. Analyzátor, které zpracovávají signál pomocí FFT mají menší rozsah většinou kolem stovek kHz, což je pro tento návrh dostačující.

Tato metoda spočívá v načtení číslicových vzorků z ADC a následnému matematickému zpracování. Pro sledování jednotlivých složek kmitočtů je potřeba přesunout signál z oblasti časové do kmitočtové oblasti pomocí FT. V praxi se využívá algoritmus rychlé Fourierovy transformace (dále jen FFT<sup>2</sup>). FFT je efektivní způsob výpočtu Fourierovy transformace v diskrétním čase (DFT<sup>3</sup>). Spektrální analyzátor musí na svém vstupu obsahovat vstupní zesilovač pro signál s nízkou úrovní napětí.

<sup>2</sup>Fast Fourier transform - Rychlá Fourierova transformace (FFT)

<sup>3</sup>Discrete Fourier transform - Diskrétní Fourierova transformace (DFT)

Problém nastává při vzorkování signálu o širším spektru (viz Obr. 1.2). Vlivem periodizace spektra dochází k jejich překryvu. Část vyššího spektra, posunutá o  $f_s^4$  (vzorkovací kmitočet  $\pm 1$  MHz), zasahuje i do užitečného spektra. Tento jev se nazývá aliasing a způsobuje nežádoucí zkreslení. Je tedy nutné vložit do obvodu filtr, jenž by filtroval všechny vyšší složky spektra, které se navzájem překrývají. K tomu slouží AAF, který je ve své podstatě pouze filtr dolní propusti, popsáný v kapitole 2.



Obr. 1.2: Periodizace spektra a AAF

Dalším blokem analyzátoru je ADC. Zde dochází k navzorkování, kvantizaci a kódování signálu. Vzhledem k tomu, že by měl přístroj pracovat v oblasti audio, je nutné měřicí přístroj na tuto oblast navrhnout. Rozsah audio oblasti je od 20 Hz do 20 kHz. Pro stanovení  $THD$  je třeba navzorkovat i vyšší složky hraničního kmitočtu. Pro hraniční složku měření byla zvolena složka 10. harmonické nejvyššího měřitelného kmitočtu. Tudiž maximální přenositelný kmitočet je roven  $200 \text{ kHz}$  ( $f_{MAX}^5$ ). Nicméně pro správné měření spektra je nejnižší  $2f_s$ , dle Nyquist-Shannonova (Kotělnikova) teorému, dána dvojnásobkem maximálního kmitočtu:

$$f_s > 2 \cdot f_{MAX}. \quad (1.4)$$

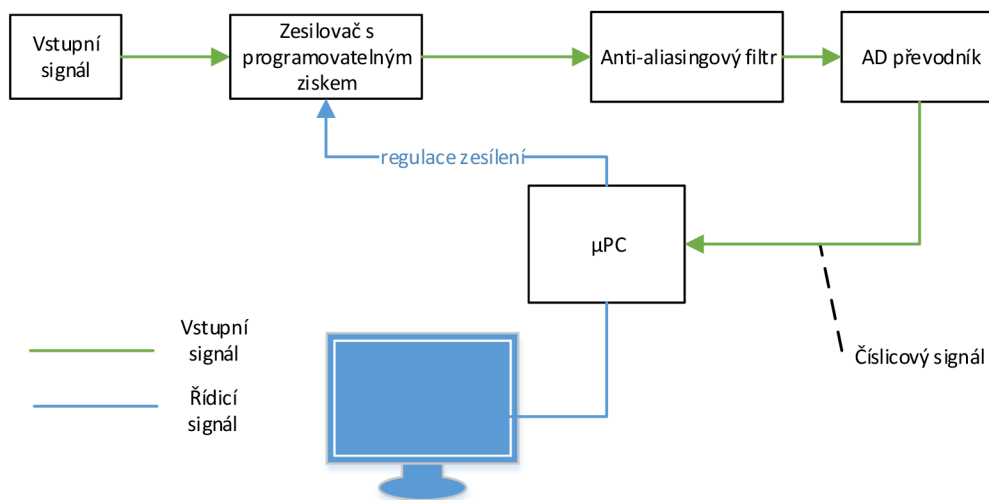
V praxi je tento kmitočet často nedostačující, proto byl zvolen  $f_s$  jako pětinašobek (1 MHz)  $f_{MAX}$ , pro zajištění detailního navzorkování nejvyšší složky o kmitočtu 200 kHz.

<sup>4</sup>sampling frequency - vzorkovací kmitočet ( $f_s$ )

<sup>5</sup>maximální kmitočet ( $f_{MAX}$ )

Dalším důležitým parametrem ADC je jeho kvantizační krok. Ten udává rozestup mezi jednotlivými hladinami a určuje dynamický rozsah přístroje. Platí, že čím více máme k dispozici bitů, tím lepší dostaneme rozlišení navzorkovaného průběhu. Pro svoji aplikaci jsem zvolil 12-bitový převodník pracující v unipolárním režimu.

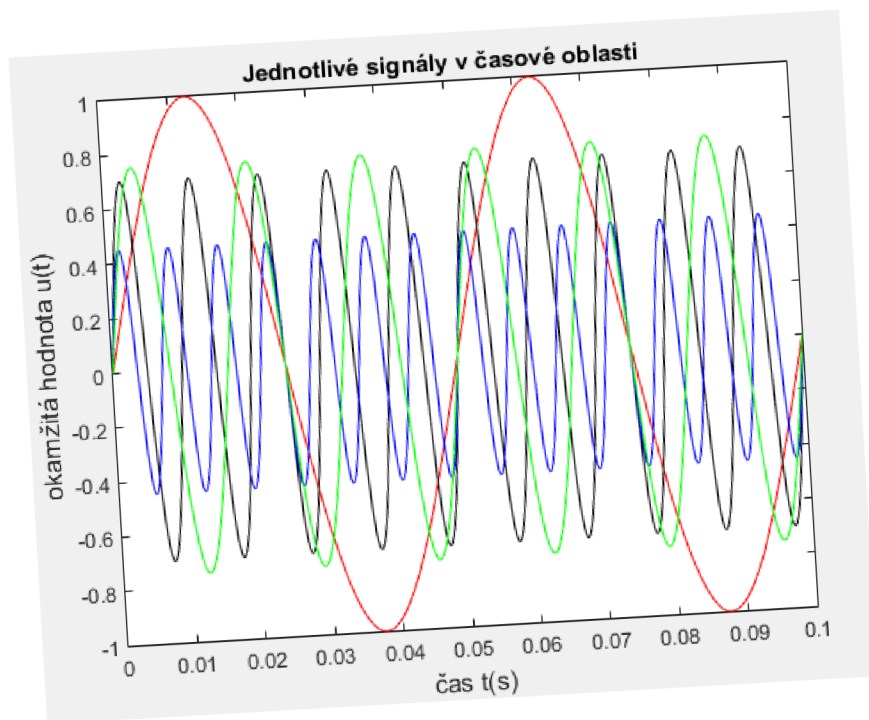
ADC dále hodnotu převádí do binárního kódu, tu posílá do platformy mikropočítače, která tyto data následně zpracovává. Zpracování naměřených dat probíhá pomocí operace DFT nebo častěji FFT. Vzorky je poté možné číslicově upravit. Data jsou pak zobrazena na dotykovém displeji. Celé blokové schéma číslicového analyzátoru je znázorněna na obr. 1.3.



Obr. 1.3: Blokové schéma spektrálního analyzátoru

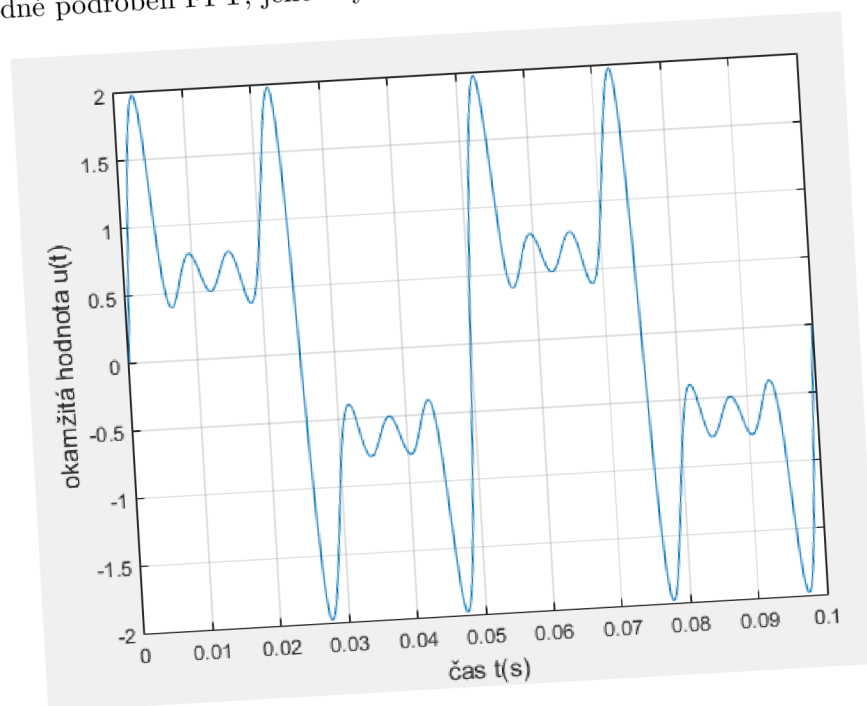
### 1.3 Simulace v prostředí Matlab

V této kapitole je simulováno softwarové zpracování signálu a výpočet  $THD$ . Pro simulaci byl vytvořen testovací signál s dominantním kmitočtem 20 Hz s jeho třetí, pátou a sedmou harmonickou složkou. Jejich amplitudy byly zvoleny tak, aby s rostoucím kmitočtem klesaly. V simulaci byl použit vzorkovací kmitočet 1 MHz, který koresponduje se vzorkovacím kmitočtem ADC. V prvním grafu jsou znázorněny jednotlivé průběhy frekvencí.



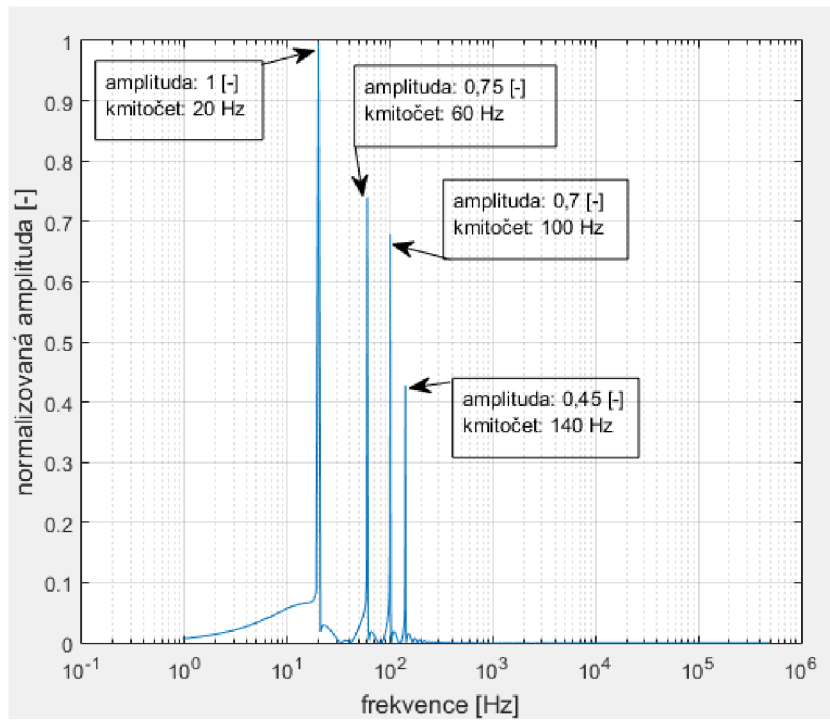
Obr. 1.4: Průběhy jednotlivých kmitočtů

Na dalším grafu je znázorněn průběh sjednocení zvolených frekvencí (obr. 1.5). Tento signál je následně podroben FFT, jehož výsledek je znázorněn na spodním grafu (obr. 1.6).



Obr. 1.5: Průběh testovacího signálu





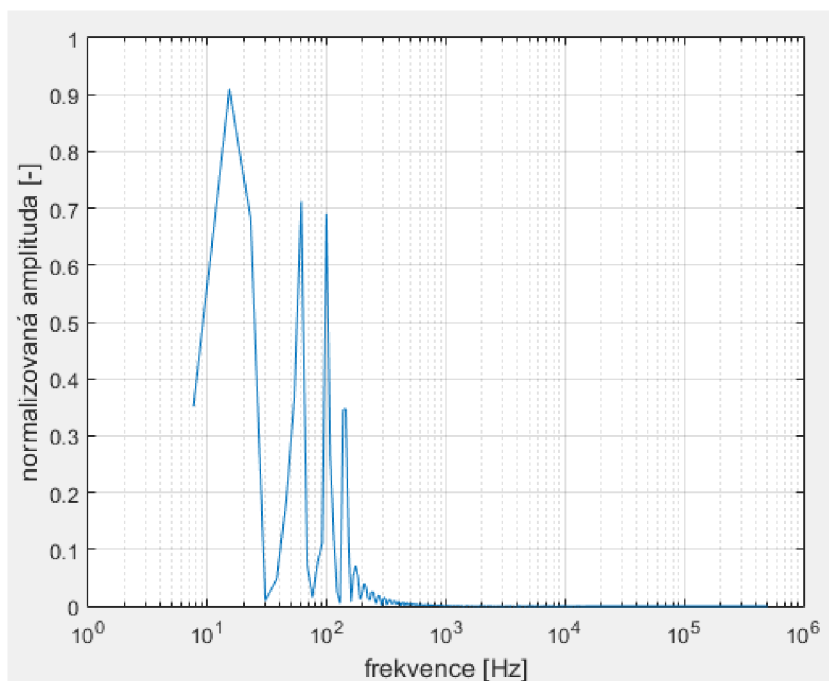
Obr. 1.6: Spektrum testovacího signálu

Úkolem navrhovaného softwaru bude načíst hodnoty amplitud násobků základního kmitočtu a provést výpočet pro  $THD_R$  a  $THD_F$ . Z uvedeného příkladu spektra je výpočet  $THD$  následující:

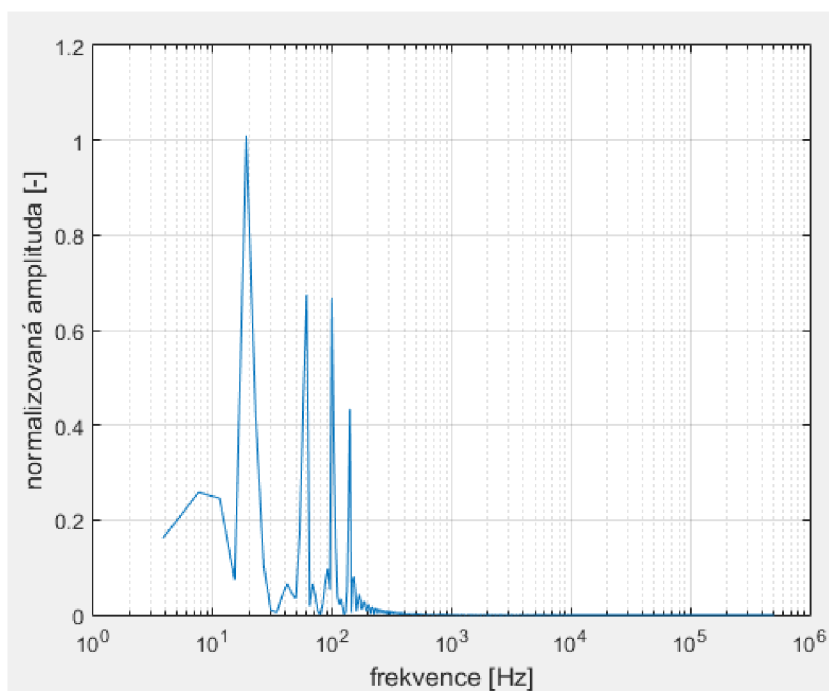
$$\begin{aligned}
 THD_F &= \frac{\sqrt{\sum_{n=2}^{\infty} U_n^2}}{U_1} = \\
 &= \frac{\sqrt{U_3^2 + U_5^2 + U_7^2}}{U_1} = \\
 &= \frac{\sqrt{(0,75)^2 + (0,7)^2 + (0,45)^2}}{(1)} = \\
 &= 1,120 \Rightarrow 112,0\%,
 \end{aligned} \tag{1.5}$$

$$\begin{aligned}
 THD_R &= \sqrt{\frac{\sum_{n=2}^{\infty} U_n^2}{\sum_{n=1}^{\infty} U_n^2}} = \\
 &= \frac{\sqrt{U_3^2 + U_5^2 + U_7^2}}{\sqrt{U_1^2 + U_3^2 + U_5^2 + U_7^2}} = \\
 &= \frac{\sqrt{(0,75)^2 + (0,7)^2 + (0,45)^2}}{\sqrt{(1)^2 + (0,75)^2 + (0,7)^2 + (0,45)^2}} = \\
 &= 0,753 \Rightarrow 75,3\%.
 \end{aligned} \tag{1.6}$$

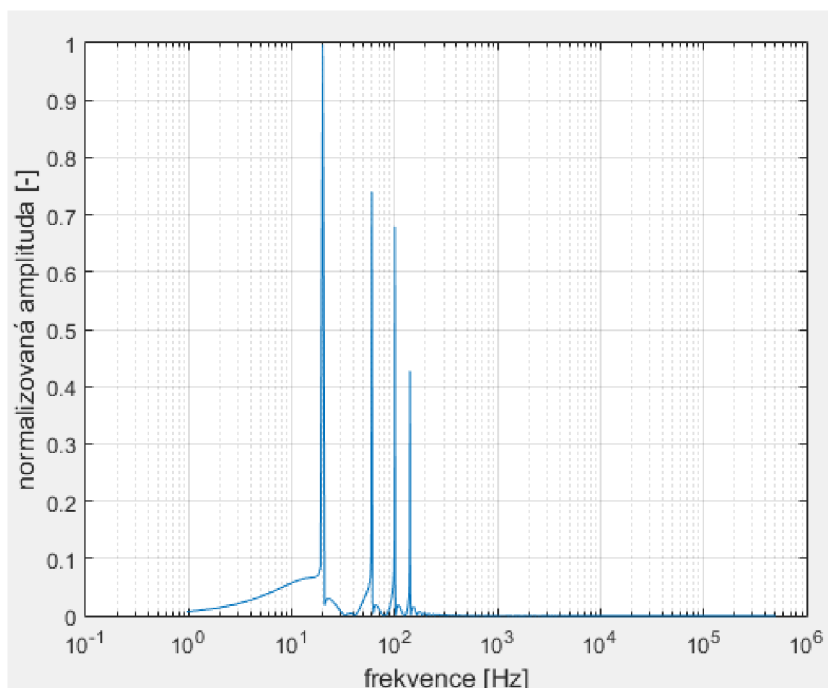
Důležitým parametrem číslicového zpracování je délka vzorku, na které je FFT aplikováno. Platí, že čím delší bude vzorek, tím přesnější bude zobrazené spektrum. Nevýhodou dlouhého vzorku je jeho doba zpracování. Doba zpracování je závislá na rychlosti procesoru a pro aplikace sledování spektra v reálném čase je cílem dosáhnout co nejkratší doby zpracování a zároveň co nejpřesnějšího výsledku. Na následujících obrázcích je výčet simulací pro různé délky vzorku.



Obr. 1.7: Spektrum signálu pro délku vzorku 0,1 s



Obr. 1.8: Spektrum signálu pro délku vzorku 0,2 s



Obr. 1.9: Spektrum signálu pro délku vzorku 1 s

Simulovány byly délky signálů od 0,1 s do 10 s. Ze simulací vyplynulo, že při délce vzorku pod 0,2 s dochází ke zkreslení spektrálních čar a nespolehlivému vykreslení amplitud.

To má pro přístroj měřící *THD* nepříznivý dopad. Docházelo by k nepřesnosti určování velikostí amplitud harmonických složek, tudíž i k nepřesnému určení *THD*.

## 1.4 Výběr platformy

Zvolená platforma by měla splňovat výše uvedené podmínky tzn. dostatečnou rychlost zpracování dat, grafický čip pro výstup na displej, schopnost komunikace s rozšiřující deskou a další. Jako platformu pro zpracování dat z ADC jsem si vybral Raspberry Pi 3, které splňuje všechny požadované parametry. Zvolil jsem jej kvůli jednoduchému spojení dotykové obrazovky přes HDMI<sup>6</sup> nebo DSI port. Rychlost procesoru je více než dostačující vzhledem k tomu, že jsou zpracovávány signály z ADC, jehož rychlost je cca 1 MS/s (1 MHz), a zároveň prováděny výpočty FFT. Jádrem celého RPi3<sup>7</sup> je totiž čtyřjádrový 1,2 GHz Broadcom BCM2837 64 bit s 1 GB velikostí RAM<sup>8</sup>. Další výhodou je oproti svému předchůdci cena. RPi3 lze totiž u některých českých dodavatelů sehnat levněji.

<sup>6</sup>z anglického označení High - Definition - Multimedia - Interface (HDMI)

<sup>7</sup>Raspberry Pi 3 (RPi3)

<sup>8</sup>z anglického názvu Random - Access - Memory (RAM)

Užitečné jsou i jeho další periferie. Například port pro síťový kabel nebo Wifi<sup>9</sup>, které lze využít pro připojení jiných zařízení (tiskárna atp.).

Důležitou součástí RPi3 jsou jeho programovatelné a snadno přístupné GPIO<sup>10</sup>, které jsou využity například ke komunikaci s periferiemi.

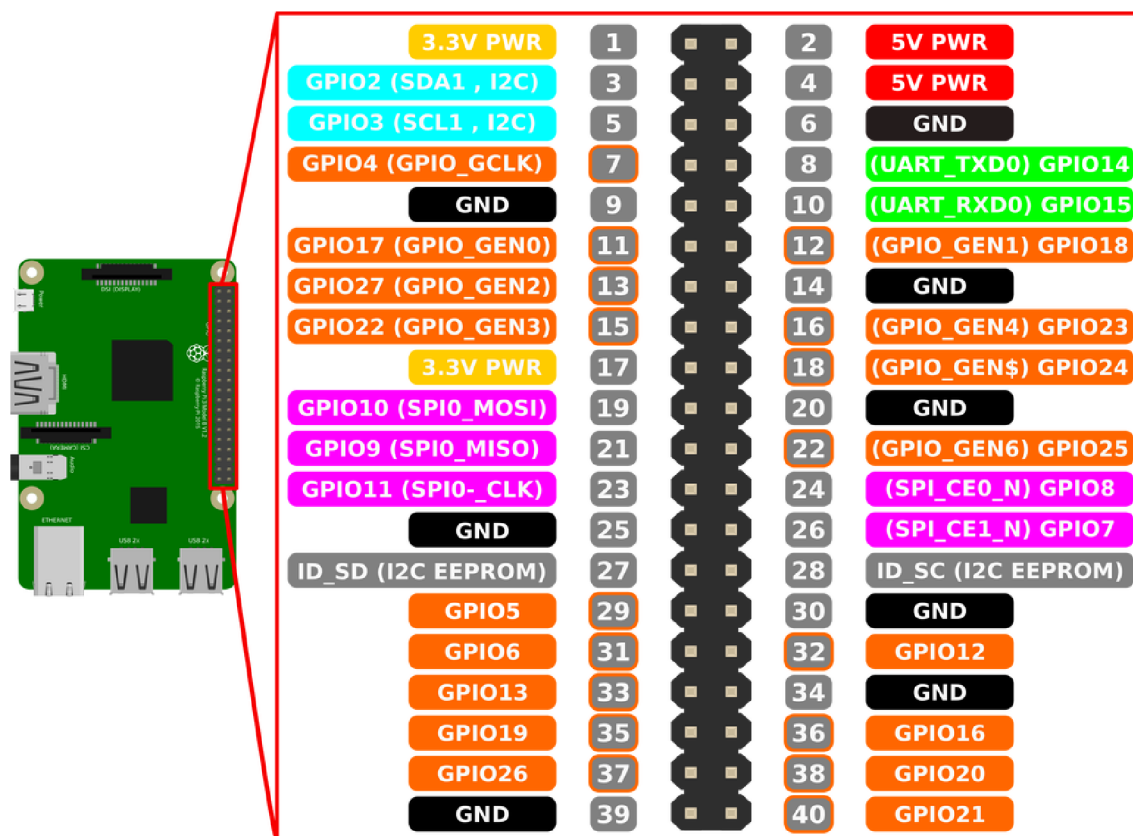


Obr. 1.10: Raspberry Pi 3 [15]

---

<sup>9</sup>z anglického označení Wireless - Fidelity (Wifi)

<sup>10</sup>General purpose input output - vstupní/výstupní piny RPi3 (GPIO)



Obr. 1.11: RPi3 - rozložení GPIO [16]

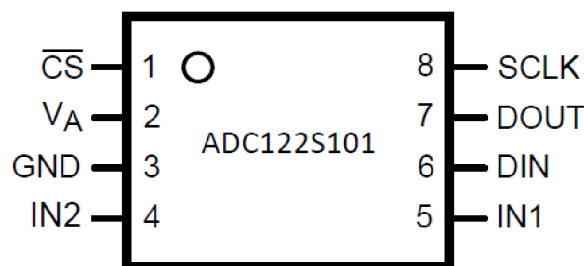
Tab. 1.1: Srovnání modelů Raspberry Pi

	Raspberry Pi 3 Model B	Raspberry Pi 2 Model B	Raspberry Pi Zero
<b>Frekvence procesoru</b>	1,2 GHz	900 MHz	250 MHz
<b>Model procesoru</b>	Broadcom BCM2837 4 jádra	ARM Cortex-A7 4 jádra	ARM1176JZF-S 1 jádro
<b>RAM</b>	1 GB	1 GB	512 MB
<b>Spotřeba</b>	1,34 A (6,7 W)	800 mA (0,4 W)	100 mA (0,5 W)
<b>Rozměry [mm]</b>	85,60 × 56,5 × 17	85,60 × 56,5 × 17	65 × 30 × 5
<b>Cena</b>	1 039 Kč	1 029 Kč	157 Kč

## 1.5 A/D převodník

Pro návrh rozšiřující desky byl zvolen ADC od výrobce Texas Instruments model ADC122S101. Převodník pracuje s rychlostí vzorkování 1 MSa/s a rozlišením 12 bitů. Pro komunikaci je nutné generovat hodinový signál o kmitočtu 16 MHz. Tento předpoklad RPi3 splňuje, jelikož je schopno vygenerovat hodinový signál až 46 MHz.

Jedná se o jednoduchý ADC typu postupné aproximace s  $THD$  -73 dB. Napájecí napětí  $V_{DD}$  nabývá hodnot od 2,7 V do 5,25 V. Tento rozsah vyhovuje rozsahu audio vstupu, který nabývá hodnot  $2 V_{pp}$  pro linkový vstup. ADC má možnost čtení jednoho ze dvou kanálů, ale pro návrh analyzátoru stačí pouze jeden, proto je nutné druhý vstup uzemnit, aby nedocházelo k indukci rušivých signálů do ADC.



Obr. 1.12: Schematické rozložení pinů AD7276 - pohled shora [4]

- **Popis pinů:**

1.  $\overline{CS}$ <sup>11</sup> (**Pin 1**): startovací signál konverze-aktivní v sestupné hraně
2.  $V_A$  (**Pin 2**): kladné napájení
3.  $GND$ <sup>12</sup> (**Pin 3**): analogová zem
4.  $V_{IN1,IN2}$  (**Pin 5,4**): analogové vstupy
5.  $SCLK$ <sup>13</sup> (**Pin 8**): hodinový signál sloužící pro synchronizaci přenosu datového slova
6.  $DOUT$ <sup>14</sup> (**Pin 7**): pin pro výstupní datový tok
7.  $DIN$ <sup>15</sup> (**Pin 6**): pin pro vstupní datový tok

Pro komunikaci s platformou využívá ADC sériové komunikační rozhraní (SPI<sup>16</sup>). SPI je sériový komunikační protokol, sloužící ke komunikaci mezi mikropočítačovými moduly

<sup>11</sup>Chip select - výběr slave systému (CS)

<sup>12</sup>Ground - analogová zem (GND)

<sup>13</sup>Serial clock - hodinový signál sériového rozhraní (SCLK)

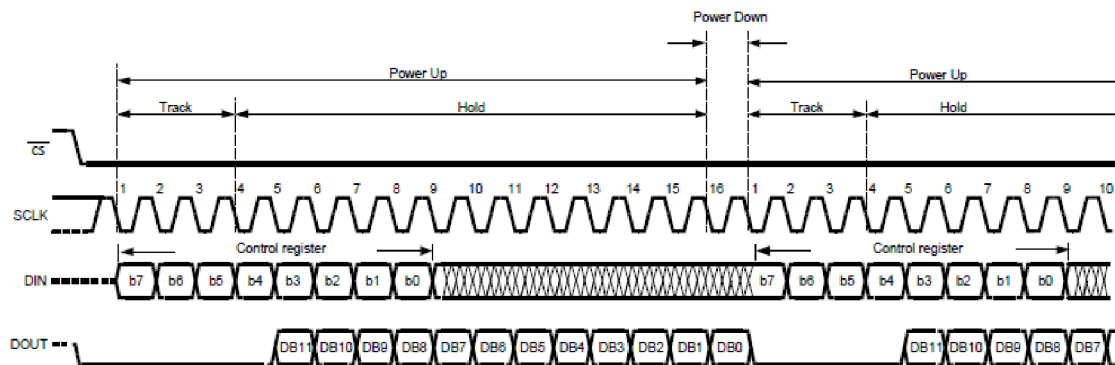
<sup>14</sup>Serial data OUT - výstupní sériová data (DOUT)

<sup>15</sup>Serial data IN - vstupní sériová data (DIN)

<sup>16</sup>Serial Peripheral Interface - sériové periferní rozhraní (SPI)

a integrovanými obvody, vyvinuté ve společnosti Motorola . Na rozdíl od I2C, umožňuje SPI rozhraní pouze jeden Master řídicí systém (mikropočítač), který generuje hodinový signál (SCLK) a limitovaný počet Slave systémů (ADC, EEPROM...). K RPi3 lze připojit až 3 periferní zařízení, kde pouze dvě jsou spojená v jednu sběrnici SPI. Výhoda SPI je v jeho schopnosti Full-duplex přenosu informací, který umožňuje současné čtení a zapisování z Master obvodu [9]. SPI má rychlejší přenos dat než jiné sériová rozhraní, ten je od 10 MS/s do 20 MS/s. Klasický protokol SPI vyžaduje čtyři datové piny pro komunikaci - MISO (master in slave out), MOSI (master out slave in), SS (slave select) a SCLK (serial clock). Nevýhodou SPI je to, že není pevně standardizován, tudíž na trhu existuje mnoho variant způsobu komunikace. Záleží tedy také na schopnosti obvodu Master (RPi3).

DIN (resp. MISO) obvodu ADC122S101 slouží k výběru jednoho ze dvou kanálů. Pro čtení kanálu 2 je nutné na vstup DIN převést 16 - bit datové slovo, které má na 3.bitu logickou 1. Pro čtení kanálu 1 stačí trvalá logická 0 na vstup DIN, proto je pin DIN v tomto návrhu uzemněn (konstantní nula na DIN).



Obr. 1.13: Průběh komunikace normálního chodu ADC [4]

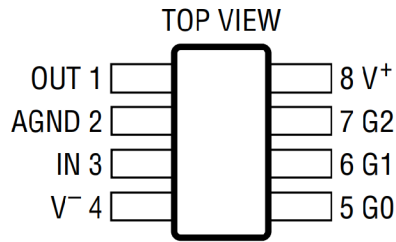
Na obr. 1.7 jsou znázorněné časové diagramy komunikace SPI. Z časových diagramů lze vyčíst, že v intervalu prvních 4 bitů začíná akvizice signálu (ADC v režimu Track). Zbytek z datového slova (16 - bit datové slovo) obsahuje převodovou hodnotu v binárním kódu (ADC v režimu Hold). Začátek snímání vstupu ADC začíná sestupnou hranou signálu  $\overline{CS}$  a končí nástupnou hranou téhož signálu.

## 1.6 Předzesilovač

Důležitou součástí rozšiřovací desky je zesilovač. Ten má za úkol zesílit nízkoúrovňové signály do optimálních mezí ADC. Vzhledem k tomu, že pro řízení a zpracování byla vybrána platforma Raspberry Pi 3, která má dostatečné množství GPIO pinů, byl zvolen zesilovač s programovatelným zesílením od firmy Linear technology model LTC6910 - 2. Důvodem je jeho jednoduchá implementace na desku plošného spoje a schopnost změny

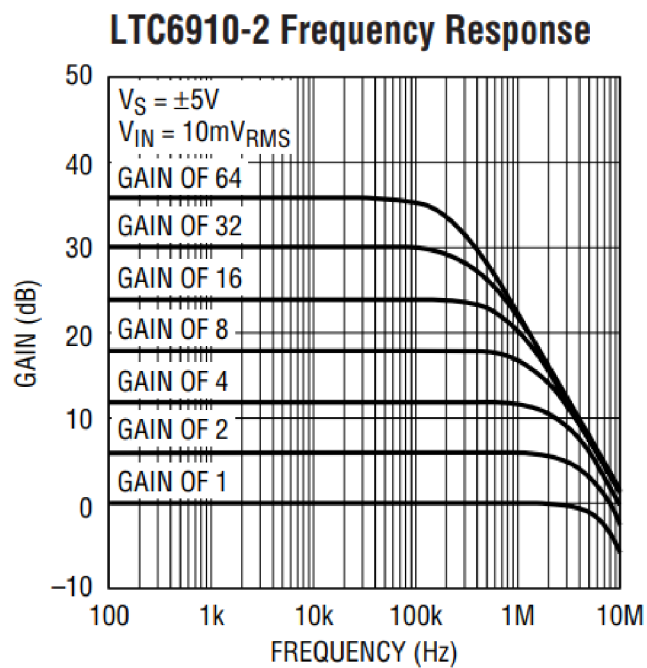


zesílení pomocí paralelní komunikace tří pinů (G2,G1,G0). Zesilovač vyhovuje z hlediska vstupního napětí, které je v rozmezí od 2,7 V do 10,5 V.



Obr. 1.14: Schématické rozložení pinů zesilovače [6]

Kombinací pinů lze dosáhnout až 8 úrovní zesílení, avšak z katalogového listu výrobce lze z kmitočtové charakteristiky vyčíst, že s rostoucím zesílením klesá šířka přenosu viz obr. 1.15. Pro hodnotu nejvyššího kmitočtu 200 kHz je možné použít maximálního zisku 16 dB, tak aby nedocházelo ke zkreslení nejvyššího zvoleného kmitočtu.



Obr. 1.15: Frekvenční charakteristika zesilovače [6]

## 2 NÁVRHY

### 2.1 Anti-aliasingový filtr

V této kapitole bude probrán způsob návrhu AAF, jehož funkce byla probrána v předešlých kapitolách. Pro návrh AAF lze využít různých druhů filtrů dolní propusti, jako jsou například filtry RC, RL nebo Sallen-Key aj.

Na začátku návrhu filtru je nutné si stanovit jeho toleranční pole. To lze sestavit se znalostmi minimální hodnoty útlumu na kmitočtu, při kterém již nedochází k aliasingu. Hraniční hodnoty útlumu a kmitočtu lze stanovit z rozlišení ADC a jeho rychlosti vzorkování. Hraniční kmitočet je v polovině  $f_s$  ADC, ta je rovna 500 kHz.

Hodnotu útlumu ( $L_{\min}^1$ ) lze určit následujícím způsobem:

1. počet úrovní 12 - bitového ADC:

$$2^{12} = 4096 \text{ úrovní} \quad (2.1)$$

2. hodnota úrovně ( $V_{\text{MIN DET}}^2$ ), které je ADC schopno rozlišit:

Tu lze určit z rozsahu napájecího napětí ( $V_{\text{DD}}^3$ ), které činí  $0\text{V} \Rightarrow 3\text{V}$ .

$$V_{\min} = \frac{V_{\text{DD}}}{4096} = \frac{3\text{V}}{4096} = 0,7324 \text{ mV}. \quad (2.2)$$

3. stanovení útlumu v dB:

Hodnoty útlumu lze stanovit jako útlum z maximálního napěťového rozsahu ( $V_{\text{DD}}$ ) k nejnižší hodnotě, který je schopný ADC rozlišit ( $V_{\text{MIN DET}}$ ). Útlum lze potom vyjádřit takto:

$$L_{\min} = 20 \cdot \log\left(\frac{V_{\text{DD}}}{V_{\text{MIN DET}}}\right) = 72,24 \text{ dB}. \quad (2.3)$$

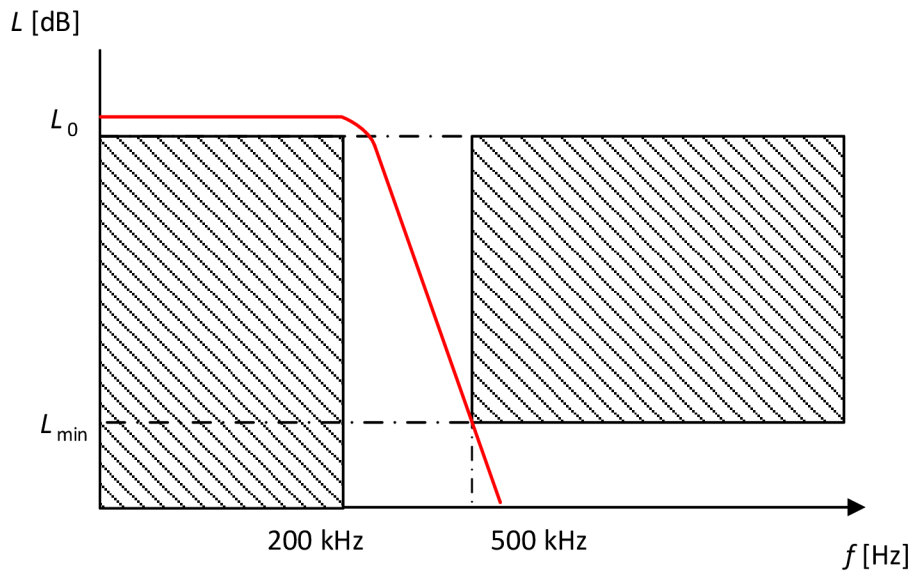
Z těchto podmínek lze stanovit toleranční pole filtru viz obr. 2.1.

---

<sup>1</sup>minimální hodnota útlumu ( $L_{\text{MIN}}$ )

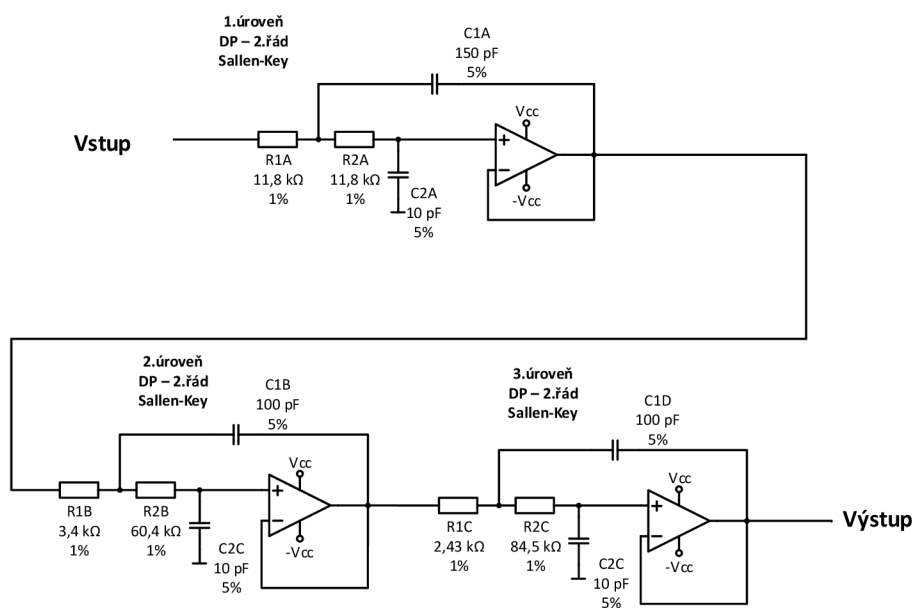
<sup>2</sup> $L_{\text{MIN}}$

<sup>3</sup>napájecí napětí ADC ( $V_{\text{DD}}$ )



Obr. 2.1: Návrh tolerančního pole

Vzhledem k takto strmým hodnotám útlumu byl vybrán integrovaný AAF s útlumem -62 dB na 500 kHz. Tento útlum není ideální ( $L_{\min} = 72,24$  dB), ale pro tento reálný obvod bude postačující. To je dáno neideálním vzorkováním ADC, který signál vzorkuje s nenulovou šířkou pulzu. Ta se ve výsledném spektru bude projevovat tak, že hodnoty na vyšších kmitočtech budou mírně tlumeny. Další možností by bylo například obvodové zapojení Sallen-Key, jenž je zobrazeno na obr. 2.2. Nevýhodou tohoto zapojení je jeho velikost, jelikož pro potřeby tohoto návrhu je nutný filtr až 6. řádu. Ten vyžaduje až 3 operační zesilovače pro 3 bloky Sallen-key filtru.



Obr. 2.2: Obvod Sallen-Key

## 2.2 Napájení

Pro napájení je využit 5 V laboratorní zdroj, a to z důvodu menšího zarušení obvodu, jak by tomu tak bylo u spínaného zdroje. Ten napájí jednotlivé části obvodu a současně RPi3. Napětí ze zdroje je přivedeno na obvod AAF a zároveň na lineární regulátor a napětovou referenci. ADC pro svoji činnost vyžaduje stabilní napětí, jelikož je od napájecího napětí odvozena referenční úroveň obvodu. Nestabilní zdroj by způsoboval nepřesnosti kvantizace signálu. ADC je tudíž je třeba ADC napájet stabilním zdrojem. Tento způsob napájení doporučuje samotný výrobce ADC [4].

Jako hodnota napětové reference byla zvolena hodnota 3 V. To je spojené s napětovou úrovní logiky RPi3 a zároveň s napětovou úrovní vstupního analogového signálu. Předpokládá se totiž linkový vstup, jehož rozsah je  $\pm 1$  V ( $2 V_{pp}$ ). Byl zvolen obvod REF1930 od Texas instruments. Ten má mimo výstupní 3 V pin i pin s napětovou úrovní 1,5 V pro nastavení stejnosměrné složky AC signálu. Referenční zdroj REF1930 není schopný dodat potřebný výkon do všech částí obvodu, proto byl do obvodu zařazen lineární regulátor MCP1700. Lineární regulátory mají sice vyšší spotřebu, než spínané zdroje, ale do obvodu nezanášejí zbytečný šum. Maximální proud tohoto regulátoru je 250 mA, to je pro tento návrh dostatečný, jelikož součet maximálních proudů všech obvodových prvků je 70 mA.

Ze znalosti maximálního odběru lze vypočítat, zda-li je obvod nutný externě chladit. Lze vycházet ze vztahů katalogového listu MCP1700 [5]. Pro maximální ztrátový výkon platí tento vztah:

$$\begin{aligned} P_{\max} &= (U_{\text{in}} - U_{\text{out}}) \cdot I_{\max} = \\ &= (5 - 3,3) \cdot 70 \cdot 10^{-3} = 119 \text{ mW}, \end{aligned} \tag{2.4}$$

$U_{\text{in}}$  je vstupní napětí hlavního napájecího zdroje (5 V),  $U_{\text{out}}$  je výstupní napětí z lineárního regulátoru a  $I_{\max}$  je maximální odběr.

Na výpočet maximálního povoleného ztrátového výkonu ( $P_{D-\max}$ <sup>4</sup>) regulátoru je třeba vyhledat některé potřebné parametry z katalogového listu. K těmto parametrům patří maximální povolená teplota spojů  $T_{J-\max}$ <sup>5</sup> (125 °C) a teplotní odpor  $R\theta_{JA}$ <sup>6</sup> (230 °C/W). Za teplotu okolí ( $T_A$ <sup>7</sup>) byla zvolena teplota 40 °C.

---

<sup>4</sup>maximální povolený ztrátový výkon, který není nutné chladit ( $P_{D-\max}$ )

<sup>5</sup>maximální teplota spojů ( $T_{J-\max}$ )

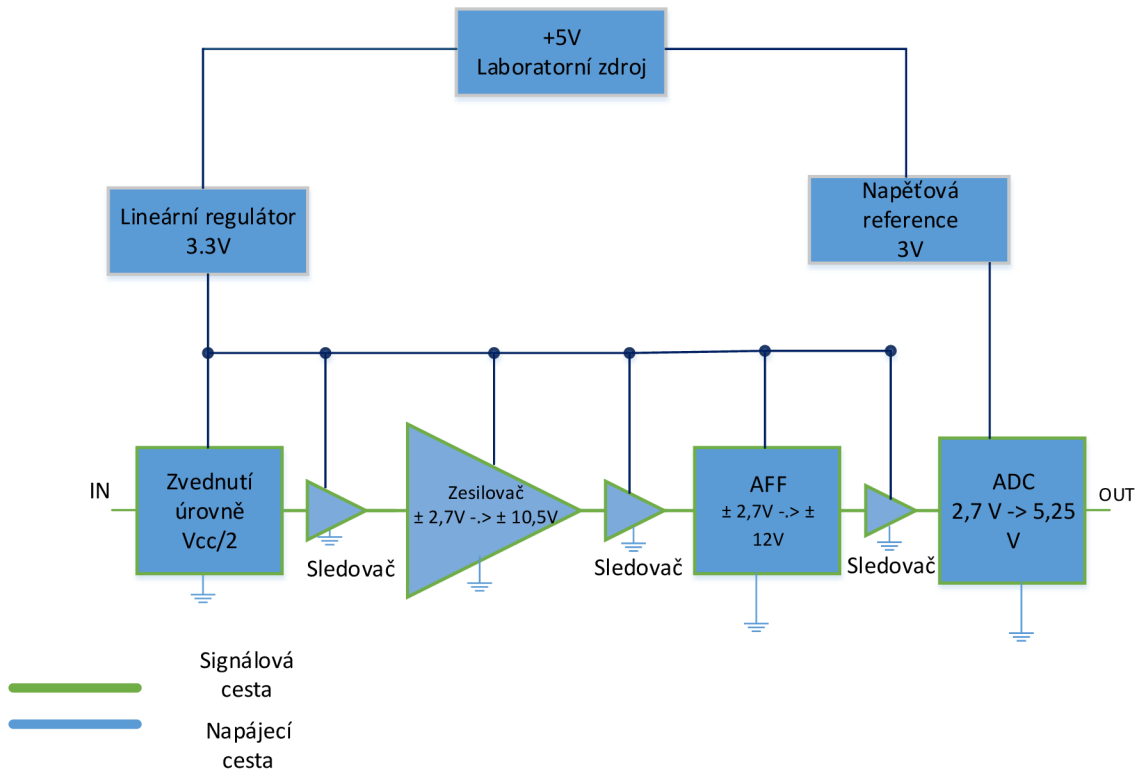
<sup>6</sup>termální odpor od spoje k okolí ( $R\theta_{JA}$ )

<sup>7</sup>teplota okolí ( $T_A$ )

Výpočet  $P_{D-max}$  je následující:

$$\begin{aligned}
 P_{D-max} &= \frac{T_{J-max} - T_A}{R\theta_{JA}} = \\
 &= \frac{125 - 40}{230} = 369mW.
 \end{aligned}
 \tag{2.5}$$

Z výsledku  $P_{D-max}$  je zřejmé, že tento regulátor není třeba externě chladit, jelikož hodnota ztrátového výkonu obvodu (119 mW) vyšla pod hranicí 369 mW.



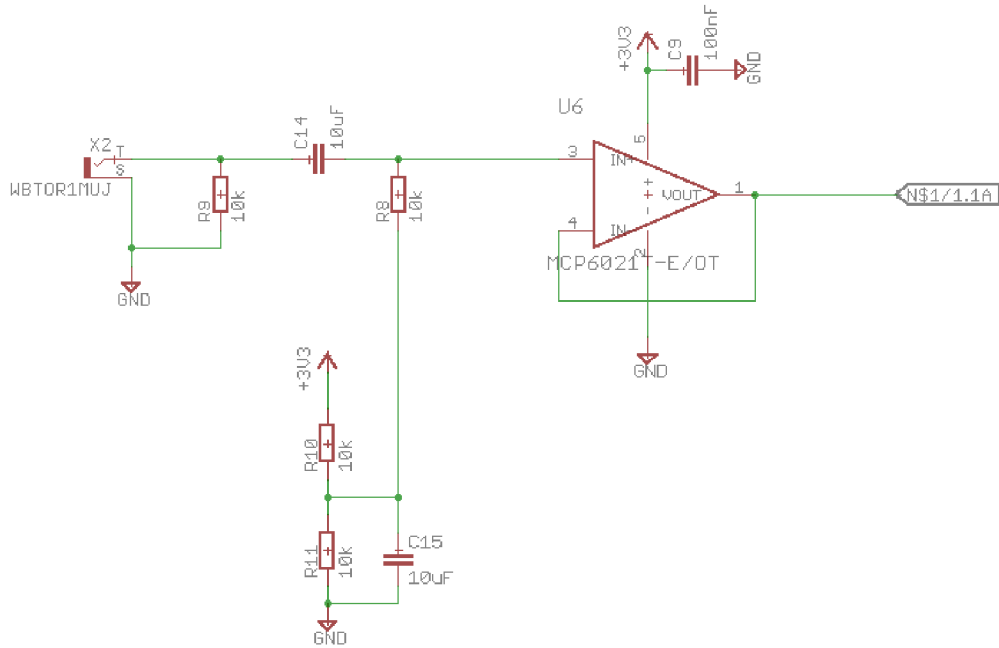
Obr. 2.3: Schéma rozvodu napájení

## 2.3 Design desky

### Zvednutí stejnosměrné úrovně napětí

Nutnost zvednutí úrovně vyplývá z charakteru napájení desky, jelikož jsou jednotlivé bloky napájeny unipolárně. Úroveň stejnosměrné složky signálu je tedy zvednutá o polovinu napětí z regulátoru (3,3 V). Existuje možnost zvednout úroveň signálu z napěťové reference, která má dedikovaný pin bias ( $V_{cc}/2$ ). Tento způsob je bohužel nevyhovující, jelikož obvod programovatelného zesilovače si za referenci bere střední hodnotu napájecího napětí (3,3 V). To je implementováno interně a nelze tuto hodnotu měnit.

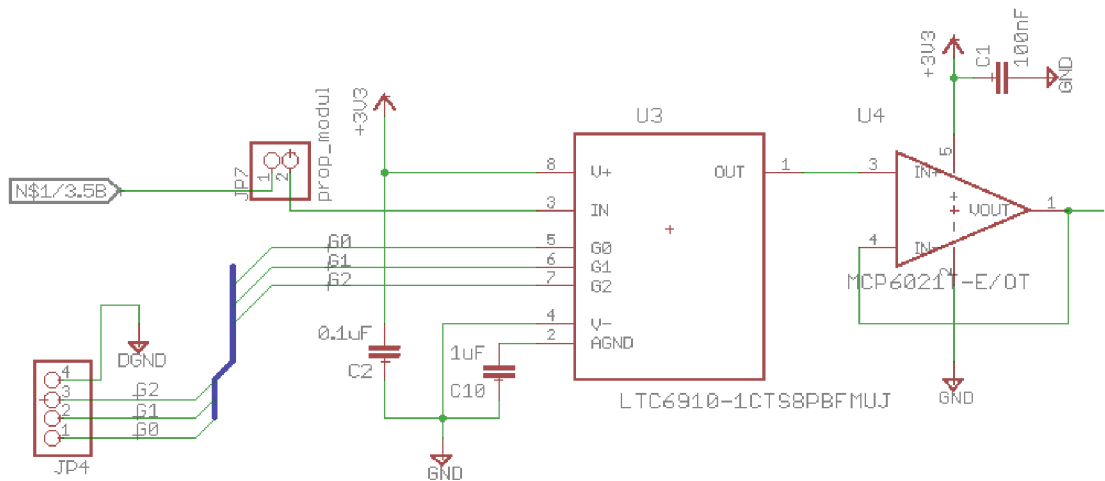
Na obr. 2.4 je zapojení obvodu pro zdvih stejnosměrné úrovně signálu.



Obr. 2.4: Zdvih stejnosměrné složky

Za tento blok je zařazen napěťový sledovač, aby impedance obvodu neovlivňovala chod zesilovače. Další z důvodů zařazení sledovače je, že se tím zmenší *THD* zesilovače [6]. To samé tvrzení platí i pro obvod AAF a ADC, před které je také zařazen sledovač.

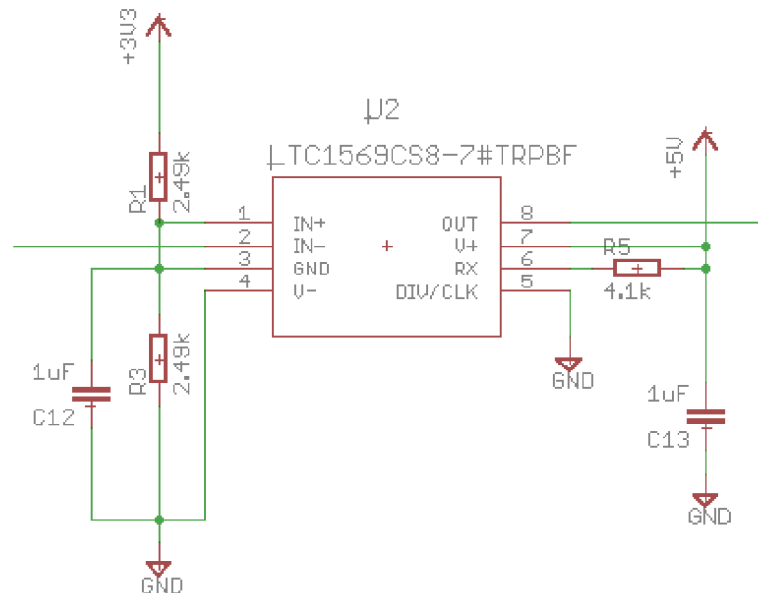
## Programovatelný zesilovač



Obr. 2.5: obvodové zapojení zesilovače

Na obrázku 2.5 je znázorněné zapojení, které vychází z katalogového listu [6]. Zesilovač je napájen napětím 3,3 V. Toto napájecí napětí je blokováno přes kondenzátor 0,1  $\mu\text{F}$ , umístěný co nejbližší k pouzdru.

## Anti-aliasingový filtr



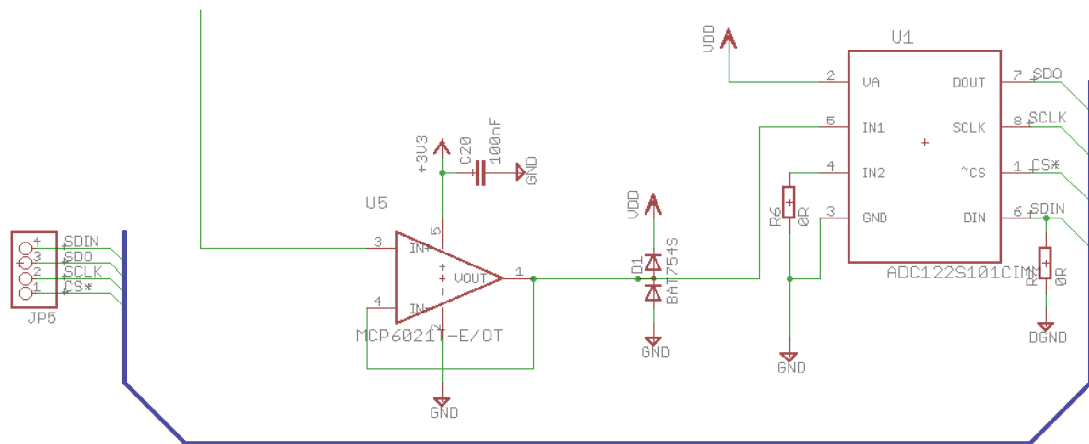
Obr. 2.6: obvodové zapojení zesilovače



Zapojení AAF vychází také z jeho katalogového listu [7]. Tento obvod je napájen napětím 5 V, kvůli nastavení mezního kmitočtu. K nastavení mezního kmitočtu je také nutné k pinu Rx připojit odpor velikosti 4,1 k $\Omega$  a přesností 1 %. Tímto se nastaví mezní kmitočet filtru na hodnotu 300 kHz. Referenční napětí je odvozeno jako polovina z napájecího napětí 3,3 V.

Vstup signálu je zapojen na invertující vstup. To je z toho důvodu, aby byl signál otočen zpět do počáteční fáze, jelikož je vstupní zesilovač invertující.

## AD převodník



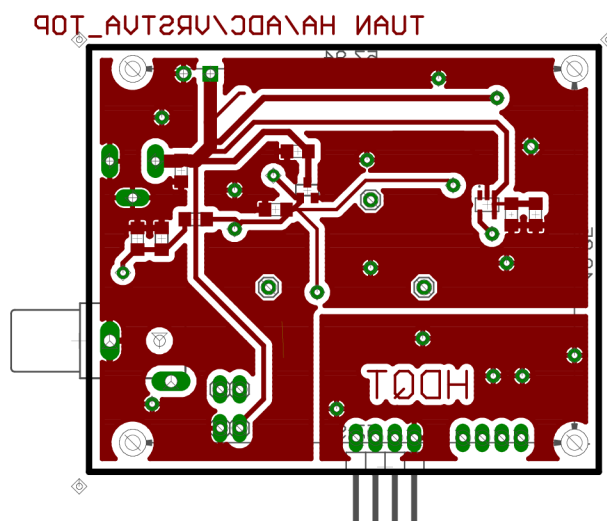
Obr. 2.7: obvodové zapojení zesilovače

Kvůli ochraně vstupu jsou před ADC zapojeny Schottkyho diody, které se při přepětí (resp. podpětí) nad 280 mV otevřou. Tímto způsobem chrání citlivý vstup ADC. Piny IN2 a SDIN jsou spojené se zemí z důvodů uvedených v kapitole A/D převodník (1.5). Napájení ADC je blokováno soustavou kondenzátorů, které na schématu nejsou zakresleny.

## Layout desky

Vytvořena je dvojrstvá deska, tak aby byla napájecí část na horní straně a signálová na spodní straně. Tímto se omezí průniky rušení do signálové části z napájení.

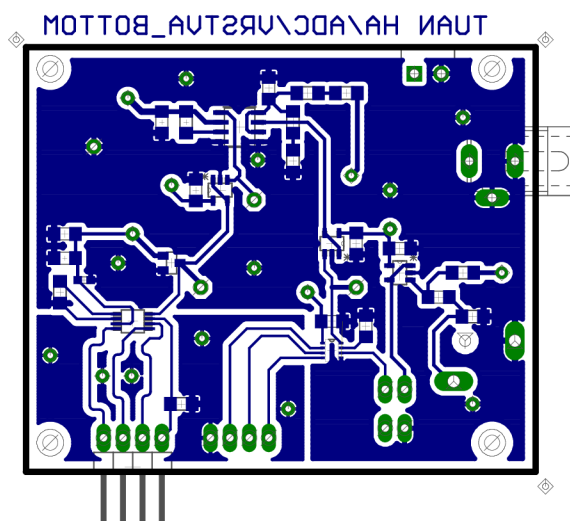
Na obr. 2.8 je návrh desky plošného spoje horní strany. Cesty byly zvolené tak, aby napětí bylo vedené ke všem blokům pouze z jednoho bodu. Tímto způsobem se limituje vzájemné rušení bloků, které jsou napájené pod stejným napětím.



Obr. 2.8: horní strana desky

Na spodním obrázku je znázorněn návrh spodní strany desky (obr. 2.9). Zde stojí za zmínku oddělení digitální části od analogové. To je provedeno oddělením digitální a analogové části země, rozdělením polygonů (levý dolní roh) a spojením v jednom místě, co nejbližší k zemnicí části zdroje.

Generujeme-li hodinový signál o kmitočtu 16 MHz, může dojít k zarušení analogového signálu. Rušení se však indukuje do digitální země a pokračuje až k hlavnímu napájecímu zdroji. Nedochozí tedy k zarušení signálové části.



Obr. 2.9: spodní strana desky

## 3 SOFTWARE

### 3.1 OS Raspbian

Na RPi3 je možné nahrát různé operační systémy. Většina z nich je založena na Linuxu. Například Ubuntu, Fedora, Debian a jiné. Pro tento návrh byl vybrán operační systém Raspbian, který patří k nejrozšířenějším distribucím a nejlépe udržovaným systémům na RPi, kterou podporuje společnost Raspberry Pi foundation.[11]

Před samotným vytvářením softwaru je nutné v operačním systému nastavit povolení SPI rozhraní a nastavit délku zásobníku. Jedná se o paměť, do které jsou ukládány data z SPI. Maximální dosažitelná hodnota tohoto uložště je 1,2 MB. To je pro účely přístroje vyhovující (vysvětleno v následující kapitole 3.4). Dále je nutné pozastavit všechny ostatní rozhraní jako je například UART, I2C atd. To je z toho důvodu, aby nedocházelo ke vzájemnému ovlivnění komunikačních rozhraní. [12] UART také využívá bluetooth, který je nutné také vypnout. Tyto změny je nutné nastavit v config.txt. Jedná se o konfigurační soubor, ze kterého se načítají data, před inicializací systému (Linuxu) a ARM CPU (centrální procesorová jednotka).[10]

Chceme-li spustit příkaz při spuštění (v tomto případě pythonovský skript), stačí tento příkaz vložit do textového souboru systému rc.local uloženého ve složce /etc [19]. Ten ve svém těle spouští všechny příkazy při inicializaci operačního systému.

### 3.2 Python - knihovny

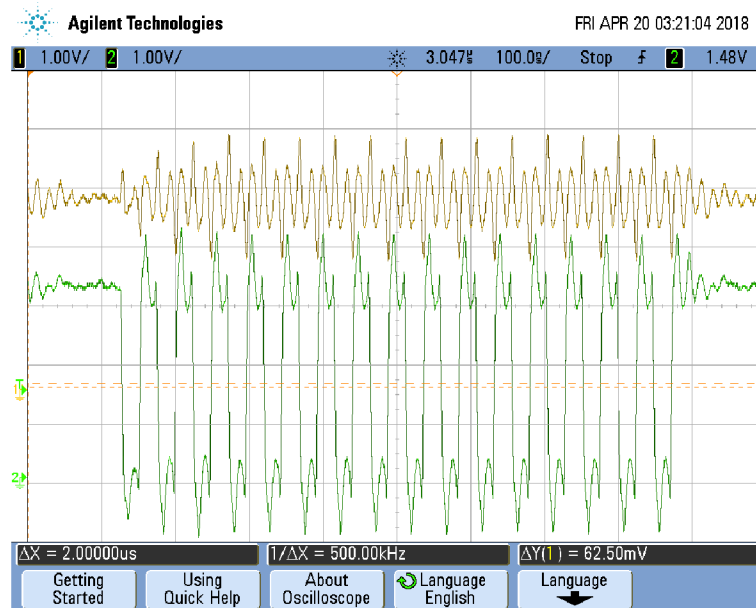
Jako programovací jazyk jsem zvolil Python. Python je dynamický interpretovaný jazyk. Jedná se o open source projekt, který nabízí instalační balíky zdarma na většinu platform. Python je již součástí OS Raspbian, tudíž není nutné další instalace.

#### **pigpio a spidev**

Rozšiřující deska (resp. ADC) komunikuje s RPi 3 pomocí SPI rozhraní, proto je nutné vybrat knihovnu, která je schopná nastavit komunikaci s ADC. Nabízí se dvě knihovny Pigpio nebo Spidev. Pigpio je rozsáhlá knihovna, obsahující jednak funkce na ovládání SPI, ale i UART, I2C a jiné. Tato knihovna je schopná nastavit přímo příznakové registry v procesoru a tím vytvářet více možností. Je například schopná nastavit libovolnou délku bitového slova od 0 do 32 bitů (ADC zpracovává 16 bit slova). Navíc je schopná vygenerovat hodinový signál až 32 MHz (ADC - 16 MHz). [17]

Po testování však vyšlo najevo, že v Pigpio knihovně nelze řídit signál CS. To může být způsobeno nekompatibilitou knihovny s novou verzí systému Raspbian. Na obr. 3.1 je vidět, že knihovna je skutečně schopná vygenerovat 16-bitové slovo s hodinovým signálem 16 MHz, ale na dedikovaném CS pinu nedocházelo k žádné změně úrovně signálu CS,

před komunikací ani během ní.



Obr. 3.1: Průběh CS (žlutě) a hodinového signálu (zeleně) knihovny pigpio

Jediné možné řešení je tedy použít knihovnu SpiDev. Tato knihovna je již součástí instalace Raspbian a je podporována Python Software Foundation. Knihovna SpiDev má na rozdíl od Pigpio mnohem omezenější možnosti. Nelze třeba číst libovolnou bitovou délku slov. SpiDev umožňuje čtení pouze po 8 bitovém slově. To pro zpracování dat není problémem, stačí brát dvě osmibitová slova jako jedno šestnáctibitové. Tento způsob bude zabírat více výpočetního času (detailnější popis v následující kapitole 3.4).

### pyQT4 a Tkinter

Vytváříme-li autonomní měřicí přístroj, který je ovládán pouze přes dotykový displej, je třeba vytvořit uživatelské rozhraní, neboli GUI<sup>1</sup> (Graphical User Interface). To lze realizovat pomocí knihoven Tkinter nebo PyQt4. Obě tyto knihovny nabízejí téměř identické možnosti a fungují na principu nekonečné smyčky. Pro vytvoření GUI Tkinter i PyQt4 je nutné vytvořit třídy, do kterých je možné vkládat vlastní kód a modifikovat jej. Pro návrh jsem vybral PyQt4, jelikož nabízí možnost návrhu GUI ve vývojovém prostředí PyQt designer, tudíž mnohem snazší návrh.

### scipy.fft a numpy.fft

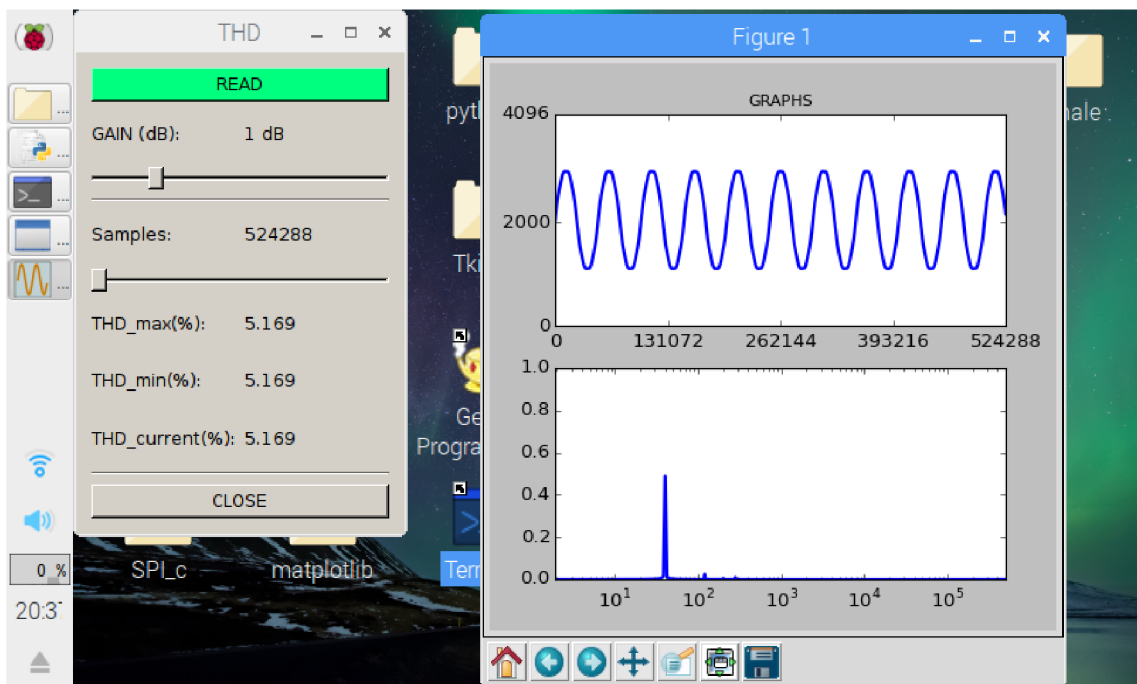
Pro výpočet FFT existují dvě knihovny Scipy a Numpy. Knihovna Scipy nabízí daleko více funkcionalit než knihovna Numpy. Hlavním rozdílem je rychlost provedení vý-

<sup>1</sup>Graphical User Interface - uživatelské rozhraní (GUI)

počtu. Knihovna Scipy má oproti Numpy, mnohem efektivnější výpočet FFT, a je tudíž rychlejší[18]. Z tohoto důvodu byla zvolena knihovna Scipy.

### 3.3 Struktura softwaru

Tato část je věnována popisu struktury a chodu navrženého softwaru. Kostru tohoto programu je z větší části tvořené třídou GUI (pyQt4). Do této třídy jsou vnořené jednotlivé funkce. Ty jsou vzájemně propojené s uživatelským rozhraním, které je znázorněné na vývojovém diagramu (obr. 3.3).



Obr. 3.2: Grafické rozhraní - THD analyzátoru (simulovaný signál s THD 5.2%)

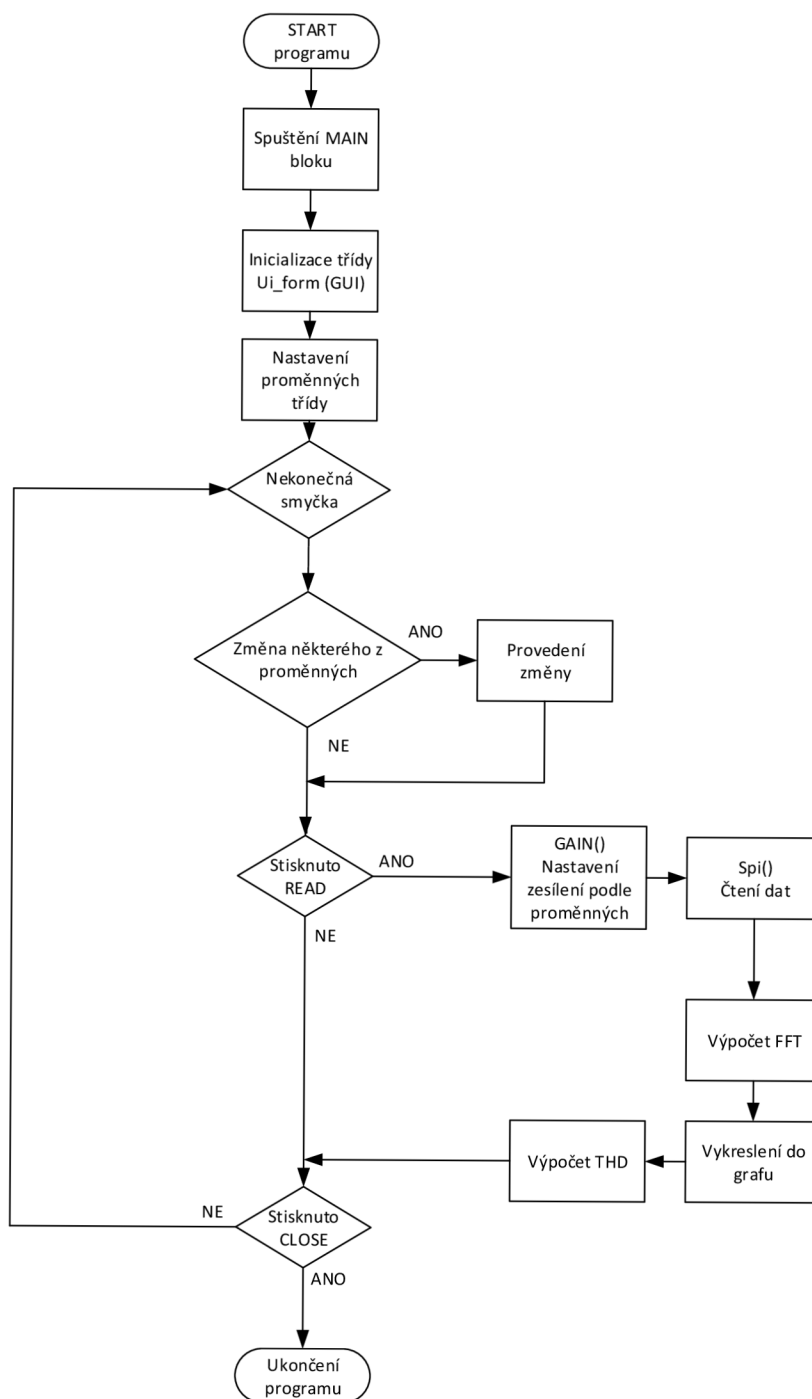
Program se spouští ihned po nastartování systému, jelikož se předpokládá, uživatelský přístup bez klávesnice a myši. Nejprve začíná čtení kódu uvnitř hlavního bloku kódu. Zde je nejdříve inicializována Qt aplikace. Poté je vytvořena třída `Ui_form`.

Uvnitř třídy se spustí metoda `__init__`, která nastaví všechny počáteční proměnné. Dále program přechází do stavu nekonečné smyčky, kde čeká na vstup od uživatele. Uživatel má možnost volit pouze velikost zesílení (GAIN) a velikost zobrazené oblasti v čase (Samples). Ostatní parametry jsou pevně dané. Nastavení parametrů je proveditelné pomocí posuvného tlačítka tak, aby bylo hodnotu možné nastavit pomocí dotykového displeje. GAIN parametr má nastavení pro šest stupňů zesílení. Na začátku spuštění programu je standartně nastavené zesílení - 120 dB, aby nedocházelo ke zbytečné saturaci a namáhání

vstupů integrovaných obvodů. Druhý parametr Samples má rozsah nastavení až dvanáct poloh tak, aby bylo možné zobrazit kmitočty od 20 Hz do 20 kHz.

Po stisknutí tlačítka READ (zelené tlačítko obr. 3.2), vyjde program z nekonečné smyčky a začne proces čtení a zpracování dat, které jsou následně vykresleny do grafů pomocí knihovny matplotlib. Stiskne-li uživatel tlačítko CLOSE, dojde k uzavření jak hlavního panelu, tak i grafu.

Z důvodu úspory místa na displeji bohužel nelze popsat osy grafů. Uživatel si tudíž musí být vědom, že v časové oblasti je osa y hodnota kvantových úrovní 12 - bitového převodníku (4096 odpovídá hodnotě 3 V), osa x je počet vzorků (Samples). Ve spektrálním grafu je na ose y normovaná amplituda [-] a na ose x je kmitočet [Hz].



Obr. 3.3: Vývojový diagram programu



### 3.4 Zpracování dat

Před tím než se přejde k výpočtu samotného THD je nutné vzorky načíst z AD převodníku. Nejdříve je nutné si stanovit délku čteného úseku. Tu lze stanovit ze znalostí maximální délky zásobníku SPI a minimální délky vzorků pro přesné stanovení spektrálních čar, která byla zjištěna v simulacích v Matlabu. Minimální délka nasnímaného časového úseku ze simulací vyšla 0,2 s, což odpovídá počtu vzorků 200 000 pro vzorkovací rychlost 1 MS/s. Tuto délku je nutné vynásobit dvěma, jelikož má jeden vzorek (slovo ADC) délku 2 bytů, tudíž je minimální velikost rovna 400 000 bytů. Tato hodnota je daleko pod hraniční hodnotou 1 200 000 bytů, je tudíž možné vybrat daleko větší úseky vzorků nepřesahující hodnotu zásobníku. Jelikož cílem této práce je stanovit, co nejpřesnější hodnotu THD, jejíž stanovení záleží na přesnosti spektra, byla vybrána nejvyšší možná hodnota 524 288 vzorků (1 048 576 bytů). Hodnota 524 288 je mocninou dvou, to protože algoritmus FFT zpracovává data s délkou mocniny dvou efektivněji.

---

```
def spi(self):
    data = spi.readbytes(self.CHUNK*2) # čtení SPI
    self.BLOK = [] # vyčištění proměnné BLOK
    #data po bytech hexa
    data_odd = data[::2]
    data_even = data[1::2]

    for i in range(len(data_odd)):
        #přeformátování do binární soustavy
        data_bits_0 = format(data_odd[i], '08b')
        data_bits_1 = format(data_even[i], '08b')
        #spojení 2 bytů - 16-bitové slovo
        data_bits_full = data_bits_0 + data_bits_1
        #ořez - 12 bit AD převodník - první 4 bity neplatné
        data_bits_part = data_bits_full[3:]
        #transformace do dekadické soustavy
        data_int = int(data_bits_part, 2)
        self.BLOK.append(data_int)
    #BLOK naplněn - konec cyklu
```

---

Výše je zobrazený způsob čtení dat ze sběrnice SPI. Data jsou ukládány pomocí funkce `readbytes()` do vektoru `data`. Parametrem této funkce je počet čtených bytů ( $2 \cdot \text{CHUNK} = 2 \cdot 524\,288$ ). Vektor vstupních dat je nutné upravit, tak aby dva po sobě jdoucí byty tvořily jedno číslo. To je vyřešeno rozdělením lichých a sudých prvků do vektorů `data_odd` a `data_even`. Prvky stejných indexů (stejně pořadí) jsou nejprve přeformátovány do binární soustavy, potom spojeny do jednoho 16-bitového slova. Toto slovo je podrobena další úpravě, protože první 4 bity ADC jsou neplatné je nutné je odštíhnout. Takto upravené slovo je následně zpětně formátováno do celočíselného datového typu (integer). Tento cyklus probíhá do naplnění vektoru `BLOK`. Vektor `BLOK` je následně vykreslen přímo do grafu časové oblasti.

Pro vykreslení spektra je nutné na BLOK aplikovat FFT. Ta vytvoří diskrétní hodnoty kmitočtů označované bin. Počet binů ve spektru je rovna počtu vzorků, platí tedy čím větší počet vzorků, tím jemnější vzorkování spektra. Velikost binu lze vypočítat jako podíl vzorkovacího kmitočtu počtem vzorků. V tomto případě je velikost jednoho binu rovna 1.907 Hz. Výsledek FFT je ve složkovém tvaru na reálnou a imaginární část. K vykreslení je použita absolutní hodnota, kterou lze získat pomocí funkce `abs()`. Vykreslená je pouze polovina z vektoru hodnot, jelikož dochází k zrcadlení kolem poloviny vzorkovacího kmitočtu.

### 3.5 Výpočet THD

---

```
def THD(self):
    #-----THD claculation -----
    yf_abs = abs(self.yf)
    yf_list_python = list(yf_abs)
    # numpy data těžko modifikovatelné -> konverze na list
    # odstranění stejnosměrné složky a symetrie po FFT
    yf_list = yf_list_python[1:len(yf_list_python)/2]
    subtract = min(yf_list)
    # maximální hodnota - základní harmonická
    value = max(yf_list)
    index = yf_list.index(value) # (najde index vektoru yf_abs) nenajde
    print(index)
    interval = int(math.floor(index*0.5))
    #interval = 5
    if value !=0 and index !=0: #podmínka max hodnota nenulová
        m = 1
        harmonics = []
        while (index*m < len(yf_list)):
            m = m+1
            param1 = (index*m)-interval
            param2 = (index*m)+interval

            #ošetření - délka vektoru spektra
            if param1 >= len(yf_list):
                param1 = (index*(m-1)-interval)
                param2 = len(yf_list)
            elif param2 >= len(yf_list):
                param2 = len(yf_list)
            else:
                Buffer = yf_list[param1:param2]

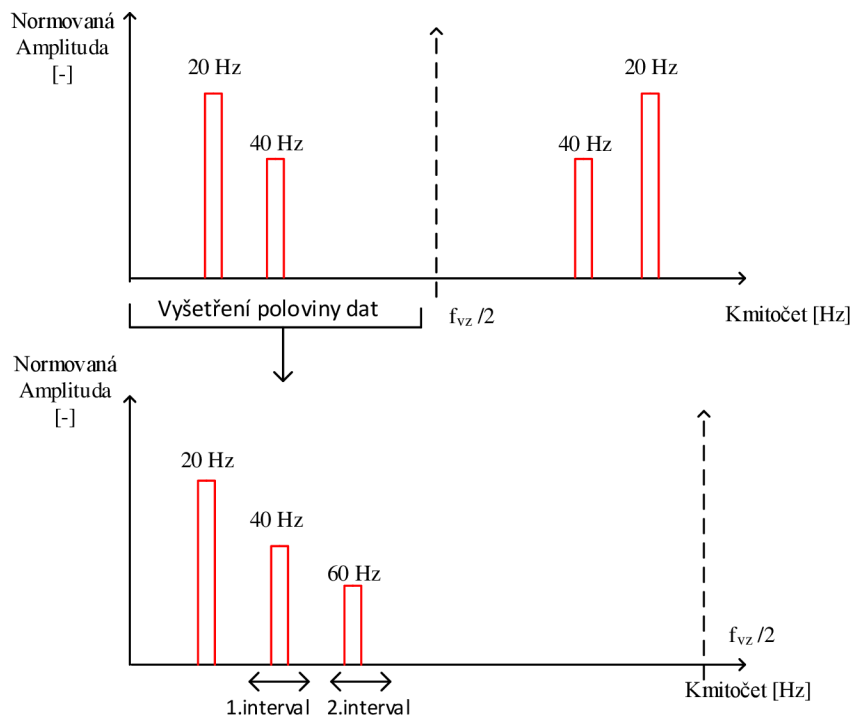
                ADD = (max(Buffer)-subtract)
                harmonics.append(ADD**2)
    # výpočet harmonické
    THD = (math.sqrt(sum(harmonics))/(value-subtract))*100
```

---

Výpočet THD je realizován jako samostatná funkce. Ve výše uvedené části kódu je vyznačena její hlavní část. V první řadě je nutné vzít výsledek z FFT v absolutní hodnotě. Výsledný vektor po FFT je datového typu math array, což je speciální datový typ, vytvořený knihovnou Scipy. Pro účely číslicového zpracování je však vhodnější převést tento datový typ na formát list. Jedná se o datový typ nativní prostředí Python, na který lze aplikovat řadu funkcí, které pro typ math array uplatnit nelze. Po konverzi se vybere polovina z vektoru, kvůli zrcadlení spektra. V této části spektra se potom vyhledá minimální (**subtract**), maximální hodnota (**value**) a index maximální hodnoty (**index**).

Je využita základní vlastnost harmonických zkreslení a to, že se zkreslení projevuje na násobcích základní harmonické. Jako základní harmonickou složku je zvolena právě maximální hodnota ze spektra (**yf\_list**), od které se odvozují vyšší harmonické složky, pomocí indexu maximální hodnoty uložené do proměnné **index**. Program počítá s diskretizací spektra a vyhledává proto intervaly kolem vyšších harmonických. Interval je uložen do vektoru **Buffer**. Zde se provede výběr největší hodnoty z vektoru, která je považována za hledanou hodnotu vyšší harmonické. Velikost intervalu byla zvolena tak, aby nedocházelo k vzájemnému překryvu vedlejších intervalů. Tento postup vybírání vyšších harmonických je naznačen na obr.DOPLNIT. Algoritmus tímto způsobem prochází celý vektor spektra. Algoritmus prochází celý **yf\_list** a ukládá maximální hodnoty z **Buffer** do proměnné **ADD**. V cyklu se pak hodnoty **ADD** umocňují přidávají na konec vektoru **harmonics**.

Je jasné, že na některých násobcích intervalů, nebude hodnota detekovatelná (bude schovaná pod hodnotou šumu). Potom se do výpočtu zahrne i tato nenulová hodnota, která může způsobit odchylku od přesné hodnoty *THD*. Algoritmus THD počítá i s touto nepřesností a kompenzuje ji odečtením minimální hodnoty od detekované maximální hodnoty v intervalu. Minimální hodnota je uložena do proměnné **subtract**. Tato hodnota je odečtená od každého prvku **ADD** včetně hodnoty **value** (hodnota základní harmonické). Na konci funkce je proveden výpočet *THD*, podle vzorového příkladu v kapitole Simulace v prostředí Matlab (rovnice 1.5). Je provedena suma všech hodnot **harmonics**. Součet je dále vydělen hodnotou **value** a vynásoben 100 pro výsledek v procentech.



Obr. 3.4: Grafické znázornění zpracování vektorů pro výpočet  $THD$

### Testovací signály

Pro testování algoritmu byl vygenerován signál identický signálu v simulaci v prostředí Matlab, jehož přesnou hodnotu  $THD$  lze stanovit ze znalosti amplitud zkreslení pomocí (1.5). Ta je rovna 112,026 %. V následující tabulce je soupis testovaných signálů, s totožnou hodnotou  $THD$ , ale rozdílným rozprostřením kmitočtů. Na každý signál byl aplikován algoritmus  $THD$  s různou délkou intervalu. Postup hledání amplitudy  $m$ -té harmonické složky je následující:

První řádek tabulky pro  $1/2$  index, 20 Hz základní harmonická :

$$20 \text{ Hz} = 11. \text{ bin} \quad (1,9 \text{ Hz} = 1 \text{ bin})$$

$$\text{index} = 11$$

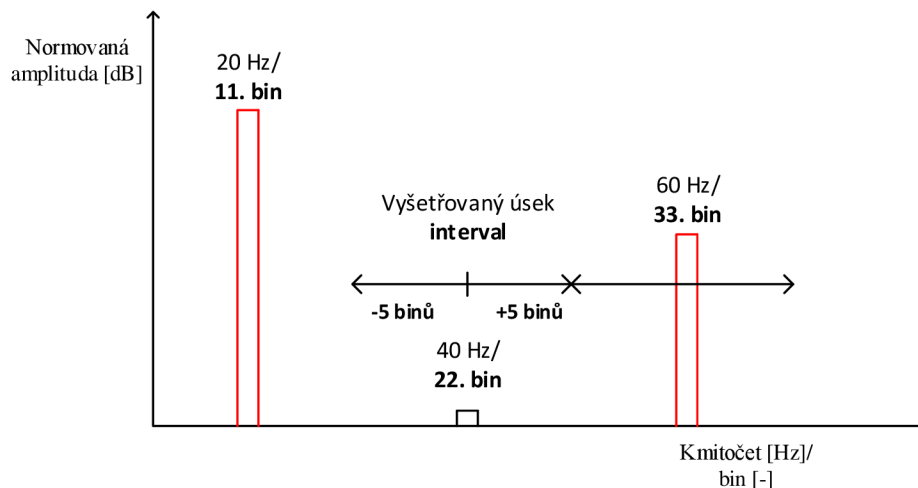
$$1/2 \text{ index} = 1/2 \text{ index} = 5,5 \Rightarrow \text{floor}(5,5) = 5 \text{ binů}$$

$$\text{interval} = \langle \text{od, střed}(\text{bin následující harmonické}), \text{do} \rangle$$

$$\text{interval} = \langle -1/2 \text{index}, m \cdot \text{index}, +1/2 \text{index} \rangle$$

$$\text{interval} = \langle -5, 22(22. \text{ bin} = \text{cca } 40 \text{ Hz}), +5 \rangle$$

$$\text{max}(\text{interval}) = \text{amplituda harmonické na } 40 \text{ Hz.}$$



Obr. 3.5: Grafické znázornění detekce amplitudy

Amplituda 22. binu je v tomto případě blízká nule (obr. 3.5). V další iteraci se hledá v oblasti 33. binu. Zmenšíme-li velikost intervalu například na  $\langle -1/3 \text{ index}, +1/3 \text{ index} \rangle$ , zmenší se prohledávaná oblast.

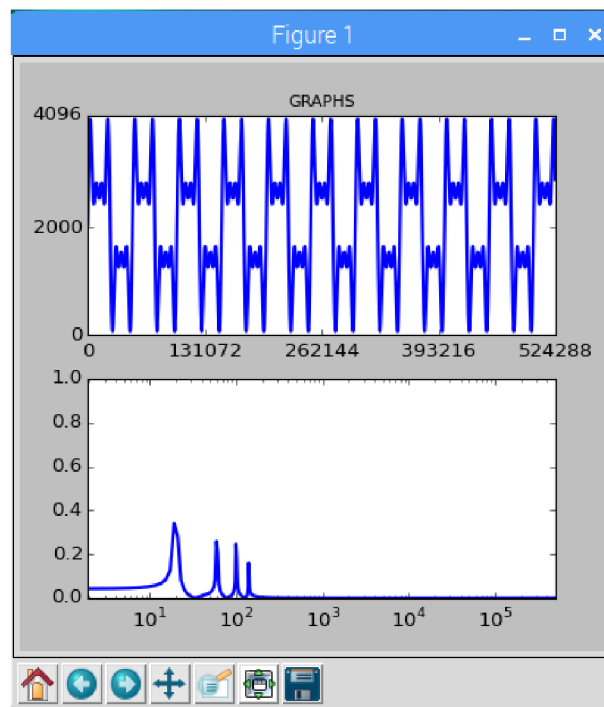
Tab. 3.1: *THD* pro různé délky intervalu

Základní harmonická složka [Hz]	Vyšší harmonické složky [Hz]	<i>THD</i> [%] 1/2 indexu	<i>THD</i> [%] 1/3 indexu	<i>THD</i> [%] 1/4 indexu
20	60,100,140	112	75	8,307
2000	6000,10000,14000	112	112	112
20000	60000,100000,140000	112	112	112
800	1600,80000,160000	112	102,5	102,5
800	1600,3200,4000	112	112	112

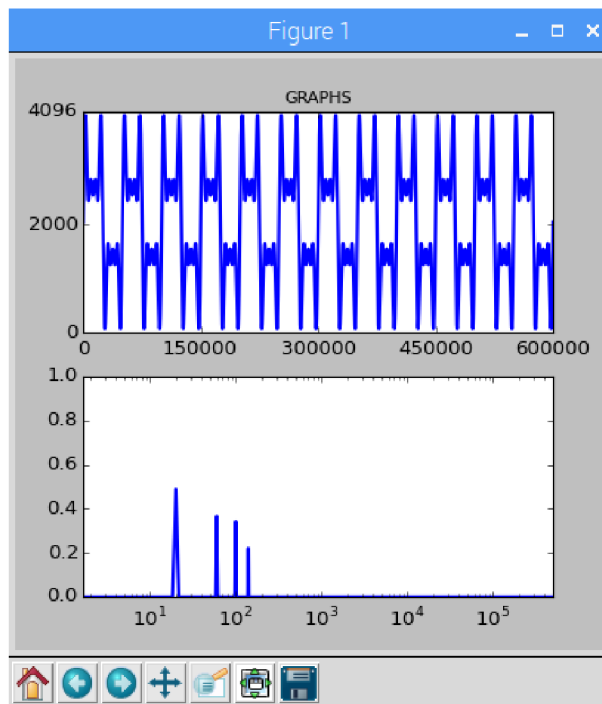
Z výše uvedené tabulky lze vyčíst, že pro korektní výpočet *THD* je nutný vzít úsek s intervalem rovným polovině **indexu** (index základní harmonické). V ideálním případě by se detekovala pouze hodnota na kmitočtu vyšší harmonické, tím by se eliminoval vliv vnějšího rušení (na harmonických s téměř nulovou amplitudou by se detekovala pouze tato

hodnota). Bohužel je v tomto případě nutné vzít celý úsek od základní harmonické do konce ( $f_s/2$ ). Neschopnost detekce vyšších harmonických na užších intervalech je pravděpodobně dána vlastnostmi FFT (diskretizace spektra) a zejména omezenou velikostí SPI zásobníku.

Během testování vyšlo najevo, že velikost vzorků hraje důležitou roli při detekci maximálních hodnot. Při délce vzorků 524 288 docházelo k výraznému zkreslení spektra (obr. 3.6). Velikost je tedy nutné zvětšit na úkor delšího výpočtu. Zvolena byla hodnota 600 000 vzorků, což je maximální dostupná hodnota SPI bufferu (obr. 3.7).



Obr. 3.6: Časový a spektrální průběh při délce vzorků 524 288



Obr. 3.7: Časový a spektrální průběh při délce vzorků 600 000

## 4 VÝSLEDKY

### 4.1 Kmitočtový rozsah

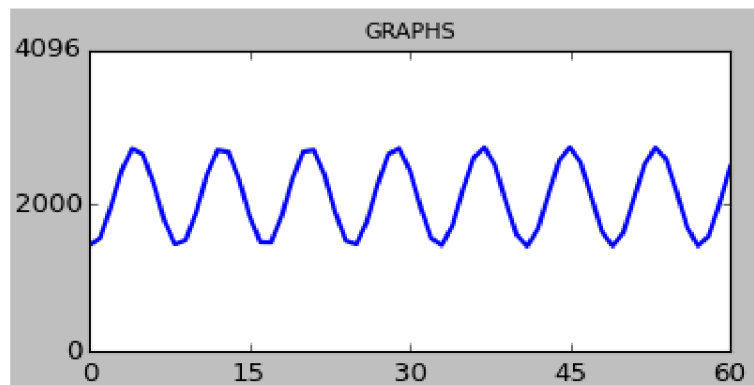
Z teoretického návrhu vyplývá, že by měl navržený přístroj měřit kmitočty do 200 kHz. Toto se testovalo pomocí generátoru funkcí (Agilent 33220A). Generován byl obdélníkový signál se střídou 50 %. Ideální obdélníkový signál má kmitočty vyšších harmonických teoreticky až do nekonečna. Na testování kmitočtového rozsahu je to ideální signál. Pro testování byl také simulován signál se stejnou střídou v jazyce Python, který je brán jako referenční. Na následující tabulce (Tab. 4.1) je srovnání generovaného signálu se signálem simulovaným.

Tab. 4.1: Srovnání *THD* generovaného a simulovaného signálu

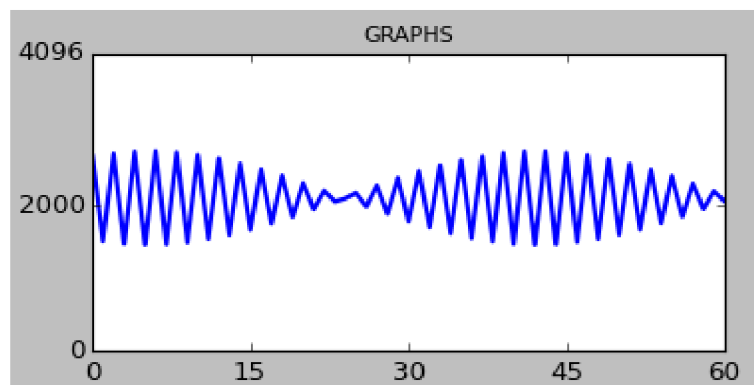
kmitočet základní harmonické složky [Hz]	<i>THD</i> simulovaného sugnálu [%]	<i>THD</i> generovaného signálu -1 [%]	<i>THD</i> generovaného signálu -2 [%]	<i>THD</i> generovaného signálu -3 [%]
20	47,813	44,127	41,120	41,122
500	48,220	52,622	52,655	52,682
5000	47,085	44,312	44,170	44,399
7000	47,948	39,828	40,870	40,000
10000	45,854	40,084	41,334	41,762
20000	43,404	31,700	31,005	30,500

Z tabulky vyplývá, že s rostoucím kmitočtem základní harmonické složky roste rozdíl mezi *THD* simulovaného a generovaného signálu. Rozdíly mezi nimi vznikají hlavně nedostatečnou rychlostí vzorkování ADC generovaného signálu pro vyšší kmitočty. To je ukázáno na obrázcích 4.1 a 4.2, na kterých je zobrazen čistý sinusový signál o kmitočtu 7 kHz a 200 kHz.



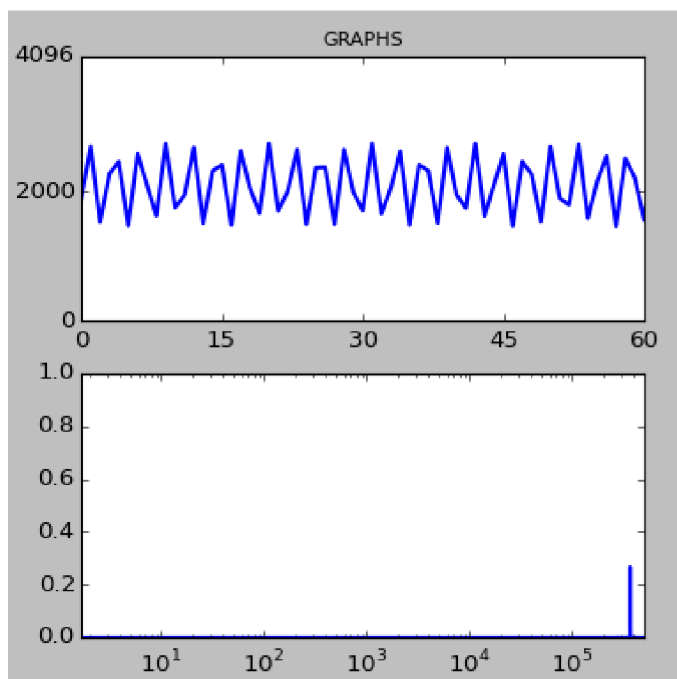


Obr. 4.1: Navzorkovaný signál 7 kHz - časová oblast



Obr. 4.2: Navzorkovaný signál 200 kHz - časová oblast

Z naměřených hodnot plyne, že přesnost  $THD$  roste s klesajícím kmitočtem základní harmonické. To dáno charakterem výpočtu  $THD$ . Na menších kmitočtech základní harmonické vychází širší pásmo (více harmonických složek), kdežto u kmitočtů vyšších základních harmonických je tomu naopak. Navíc se zde v menší míře uplatňuje konečná rychlost vzorkování ADC, i přesto je ADC schopný zobrazit spektrální čáry vyšších harmonických jak je tomu na obr. 4.3.



Obr. 4.3: Navzorkovaný sinusový signál 150 kHz - časová a kmitočtová oblast

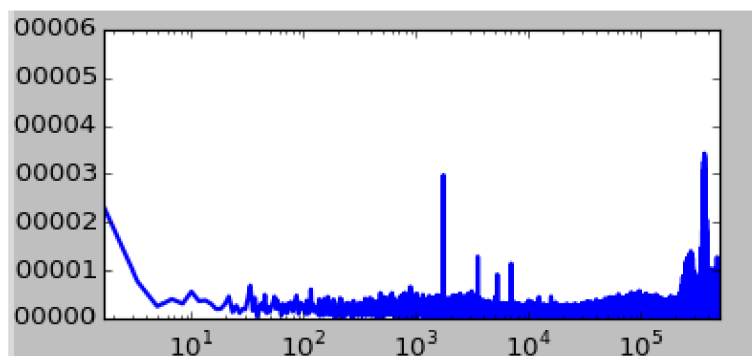
## 4.2 Dynamický rozsah

Dynamický rozsah ( $DR^1$ ) spektrálního analyzátoru je dán jako poměr mezi maximální detekovatelnou hodnotou ( $U_{\text{norm} - \text{MAX}}^2$ ) a úrovní šumu ( $U_{\text{norm} - \text{šum}}^3$ ) [3]. Pro stanovení maximální hodnoty byl vygenerován sinusový signál s maximální amplitudou, tak aby nebyl v časové oblasti zkreslen. Ve spektru se pak stanovila jeho hodnota normované amplitudy, která vyšla 0,78579 [-]. Maximální úroveň šumu se potom určovala bez přítomnosti signálu, ta je zobrazena na obr. 4.4. Maximální úroveň byla potom určena jako  $3,5567 \cdot 10^{-5}$  [-]. Obě tyto hodnoty byly určeny jako maximální hodnoty ze spektra pomocí funkce `max()`.

<sup>1</sup>dynamický rozsah (DR)

<sup>2</sup>maximální úroveň detekovatelného napětí ( $U_{\text{norm} - \text{MAX}}$ )

<sup>3</sup>maximální úroveň šumu ( $U_{\text{norm} - \text{šum}}$ )



Obr. 4.4: Úroveň šumu (na ose y již nezbylo místo na zobrazení první číslice - hodnota 00006 [-] odpovídá hodnotě 0,00006 [-])

Ze znalostí obou hodnot lze potom DR stanovit jako:

$$DR = 20 \cdot \log \left( \frac{U_{\text{norm}} - \text{MAX}[-]}{U_{\text{norm}} - \text{šum}[-]} \right) = 20 \cdot \log \left( \frac{0.78579}{3,5567 \cdot 10^{-5}} \right) = 86,885 \text{ dB.} \quad (4.1)$$

### 4.3 Stanovení odchylky

Pro stanovení odchylky byl znovu zvolen obdélníkový signál s konstantním kmitočtem (20 Hz), ale měnící se střídou. Tu lze na generátoru měnit od 20 % do 80 %. Změnou střídou se mění i rozložení a amplitudy jednotlivých vyšších harmonických složek, tedy i *THD*. Odchylka se stanoví jako nejvyšší relativní odchylka z naměřených hodnot. Na následující tabulce (Tab. 4.2) je soupis všech naměřených hodnot s vypočtenými odchylkami.

Tab. 4.2: Výpočet odchylek pro různé střídou signálu

Střída [%]	<i>THD</i> simulovaného signálu [%]	<i>THD</i> generovaného signálu 1.měř. [%]	<i>THD</i> generovaného signálu 2.měř. [%]	<i>THD</i> generovaného signálu 3.měř. [%]
20	111,440	92,998	93,005	93,006
30	75,123	66,039	65,838	65,840
40	52,739	48,268	48,249	48,171
50	47,813	41,115	41,116	41,113
60	52,734	48,366	48,478	48,318
70	75,119	66,088	66,082	66,083
80	111,441	111,328	93,383	93,407

Střída [%]	<i>THD</i> generovaného signálu 3.měř. [%]	Absolutní odchylka [%]	Relativní odchylka [%]
20	93,006	-18,437	19,824
30	65,840	-9,217	13,985
40	48,171	-4,510	9,351
50	41,113	-6,698	16,290
60	48,318	-4,347	8,984
70	66,083	-9,035	13,672
80	93,407	-12,069	12,145

Výpočet relativní ( $\delta_p^4$ ) odchylky je následující (pro střídu 20 %)[2]:

1. výpočet absolutní odchylky ( $\Delta_p$ ):

$$\Delta_p = X_M - X_P = 93,003 - 111,440 = -18,437 \%, \quad (4.2)$$

kde  $X_M^5$  je změřená hodnota a  $X_P^6$  je hodnota konvenčně pravá. Za  $X_P$  je dosazena hodnota ze simulací a za  $X_M$  je dosazen aritmetický průměr ze tří naměřených hodnot generovaného signálu.

---

<sup>4</sup>relativní odchylka ( $\delta_p$ )

<sup>5</sup>změřená hodnota ( $X_M$ )

<sup>6</sup>hodnota konvenčně pravá ( $X_P$ )

2. Výpočet relativní odchylky ( $\delta_p$ ):

$$\delta_p = \left| \frac{\Delta_p}{X_M} \right| \cdot 100 = \left| \frac{-18,437}{93,003} \right| \cdot 100 = 19,824 \%. \quad (4.3)$$

## 5 ZÁVĚR

V rámci práce byly prozkoumané vlastnosti a metody výpočtu  $THD$ . Byl vysvětlen jeho způsob vzniku a jeho projev ve spektru. Pro měření  $THD$  byla zvolena vhodná měřicí metoda založená na snímání spektra signálu a následného zpracování. K výpočtu  $THD$  je použita metoda výpočtu  $THD_F$ , která lépe vystihuje chování při velkém zkreslení (kap. 1.1). Poté byla vytvořena simulace, ze které se stanovily potřebné parametry (minimální délka vzorku) pro návrh a realizaci měřicího přístroje. Ze stanovených hraničních parametrů byly zvoleny vhodné integrované obvody a měřicí platforma.

Na základě vybraných obvodových prvků byla realizována deska plošných spojů, napájené 5 V laboratorním zdrojem. Toto napájení je nutné dále rozčlenit na úroveň 3,3 V a 3 V pomocí lineárního regulátoru (MCP1700) a obvodu referenčního napětí (REF1930). Pomocí návrhového softwaru Eagle byla vytvořena dvouvrstvá deska plošných spojů.

V kapitole software jsou popsány vybrané knihovny pro softwarovou realizaci v jazyce Python. Je zde popsána struktura a mechanismus chodu hlavního programu. Na závěr je měřicí zařízení testováno generátorem funkcí. Je stanoven kmitočtový, dynamický rozsah a relativní odchylka přístroje, která vyšla 19,824 %.

Takto velká odchylka je pro měřicí přístroj nevhodná. Důvodem způsobující takto velkou  $\delta_p$  může být například malá vzorkovací rychlost ADC, nedostatečná velikost SPI zásobníku v RPi3 nebo neefektivní algoritmus.

Původní návrh měřicího přístroje počítal s ADC (AD7276) se vzorkovací rychlostí 3 MS/s. Ten však nebyl schopný SPI komunikace s RPi3. Na vině byla špatná dokumentace knihovny SpiDev, která by měla mít schopnost změny signálu CS mezi definovanými bloky (pomocí funkce `xfer`). Tato změna signálu při čtení je pro běh AD7276 a jiné ADC nezbytná. Konečný návrh se tedy spokojil s ADC122S101 s rychlostí 1 MS/s, který vyhovoval hraničním podmínkám a byl schopný komunikace s RPi3.

## LITERATURA

- [1] DORON SHMILOVITZ, On the Definition of Total Harmonic Distortion and Its Effect on Measurement Interpretation, IEEE transactions on power delivery, vol. 20, no. 1, january 2005
- [2] BARTUŠEK, K.; GESCHEIDTOVÁ, E.; KUBÁSEK, R.; MIKULKA, J.; REZ, J.; STEINBAUER, M. Měření v elektrotechnice. Měření v elektrotechnice. Brno: VUTIM, 2010. ISBN: 978-80-214-4160- 6.
- [3] HAASZ, V. SEDLÁČEK, M., Elektrická měření, Přístroje a metody. ČVUT, Praha 2005, ISBN 80-01-02731-7.
- [4] TEXAS INSTRUMENTS INC. katalogový list ADC - ADC122S101, Data Sheet ADC122S101.
- [5] MICROCHIP TECHNOLOGY INC. katalogový list lineárního regulátoru MCP1700, Data Sheet Low Quiescent current LDO - MCP1700.
- [6] LINEAR TECHNOLOGY CORPORATION 1998 - PRINTED IN USA, katalogový list Zesilovače s programovatelným ziskem - modelu LTC6910 - 2, Digitally Controlled Programmable Gain Amplifiers in SOT-23.
- [7] LINEAR TECHNOLOGY CORPORATION 1998 - PRINTED IN USA, katalogový list AAF - modelu LTC1569-7, Linear Phase, DC Accurate, Tunable, 10th Order Lowpass Filter.
- [8] TEXAS INSTRUMENTS INCORPORATED, katalogový list atenuátoru model LM1971, LM1971 Overture™ Audio Attenuator Series Digitally Controlled 62 dB Audio Attenuator with/Mute.
- [9] DUDÁČEK, KAREL. SÉRIOVÁ ROZHRANÍ SPI, MICROWIRE, I2C A CAN [ONLINE]. PLZEŇ : ZÁPADOČESKÁ UNIVERZITA V PLZNI, 2002 [CIT. 2017-11-5]. DOSTUPNÉ Z WWW: <[HTTP://HOME.ZCU.CZ/~DUDACEK/NMS/SERIOVA\\_ROZHRANI.PDF](http://home.zcu.cz/~dudacek/nms/seriova_rozhrani.pdf)>.
- [10] RASPBERRY PI DOCUMENTATION, <https://www.raspberrypi.org>, CONFIG TXT [ONLINE]. . DOSTUPNÉ Z WWW: <<https://www.raspberrypi.org/documentation/configuration/config-txt/README.md>>
- [11] RASPBIAN EXPLAINED, By Gavin Thomas [ONLINE]. . DOSTUPNÉ Z WWW: <<https://www.gadgetdaily.xyz/moto-360-smartwatch-to-be-sold-for-249/>>
- [12] RB PI 3 FAIL TO LOAD SPI1, <https://www.raspberrypi.org>, RPI3 fórum [ONLINE]. . DOSTUPNÉ Z WWW: <<https://www.raspberrypi.org/forums/viewtopic.php?t=146291>>

- [13] THE RASPBERRY PI FOUNDATION, SPI documentation for BCM2835 [ONLINE]. UK 2014 [CIT. 2017-12-8]. DOSTUPNÉ Z WWW: <<https://www.raspberrypi.org/documentation/hardware/raspberrypi/spi/README.md>>
- [14] ANALOG DEVICES INCORPORATED, Analog filter wizard [ONLINE]. DOSTUPNÉ Z WWW: <<http://www.analog.com/designtools/en/filterwizard/>>.
- [15] THE RASPBERRY PI FOUNDATION, products Raspberry pi 3 model B [ONLINE]. UK 2014 [CIT. 2017-12-8]. DOSTUPNÉ Z WWW: <<https://www.raspberrypi.org/products/raspberry-pi-3-model-b/>>
- [16] JEREMY LINDSAY, GPIO pinout Raspberry pi 3 model B [ONLINE]. 2017 [CIT. 2017-12-8]. DOSTUPNÉ Z WWW: <<https://jeremylindsayni.files.wordpress.com/2017/04/gpiopinsv2withpi.png?w=1000>>
- [17] PIGPIO LIBRARY, dokumentace pigpio knihovny [ONLINE]. 2012 [CIT. 2018-5-5]. DOSTUPNÉ Z WWW: <<http://abyz.me.uk/rpi/pigpio/python.html>>
- [18] SCIPY LIBRARY, StackOverflow fórum [ONLINE]. 2012 [CIT. 2018-5-8]. DOSTUPNÉ Z WWW: <<https://stackoverflow.com/questions/6363154/what-is-the-difference-between-numpy-fft-and-scipy-fftpack>>
- [19] RC.LOCAL, www.raspberrypi.org [ONLINE]. 2012 [CIT. 2018-5-8]. DOSTUPNÉ Z WWW: <<https://www.raspberrypi.org/documentation/linux/usage/rc-local.md>>



## SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

<i>THD</i>	Total harmonic distortion - Celkové harmonické zkreslení
ADC	Analog to digital converter - AD převodník
<i>THD</i>	Total harmonic distortion - celkové harmonické zkreslení
$f_s$	sampling frequency - vzorkovací kmitočet
$f_{MAX}$	maximální kmitočet
RPi3	Raspberry Pi 3
AAF	Antialiasing filter - Antialiasingový filtr
HDMI	z anglického označení High - Definition - Multimedia - Interface
RAM	z anglického názvu Random - Access - Memory
Wifi	z anglického označení Wireless - Fidelity
USB	Universal Serial Bus - Universální sériová sběrnice
GPIO	General purpose input output - vstupní/výstupní piny RPi3
GND	Ground - analogová zem
SCLK	Serial clock - hodinový signál sériového rozhraní
DIN	Serial data IN - vstupní sériová data
DOUT	Serial data OUT - výstupní sériová data
CS	Chip select - výběr slave systému
SPI	Serial Peripheral Interface - sériové periferní rozhraní
FFT	Fast Fourier transform - Rychlá Fourierova transformace
DFT	Discrete Fourier transform - Diskrétní Fourierova transformace
FT	Fourier transform - Fourierova transformace
$L_{MIN}$	minimální hodnota útlumu
$V_{MIN DET}$	minimální hodnota napětí, které je schopno ADC rozlišit
$T_{J-max}$	maximální teplota spojů
$T_A$	teplota okolí
$P_{D-max}$	maximální povolený ztrátový výkon, který není nutné chladit
$R\theta_{JA}$	termální odpor od spoje k okolí
$V_{DD}$	napájecí napětí ADC
GUI	Graphical User Interface - uživatelské rozhraní
DR	dynamický rozsah
$U_{norm - MAX}$	maximální úroveň detekovatelného napětí
$U_{norm - \text{šum}}$	maximální úroveň šumu
$\delta_p$	relativní odchylka
$\Delta_p$	absolutní odchylka
$X_M$	změřená hodnota
$X_P$	hodnota konvenčně pravá

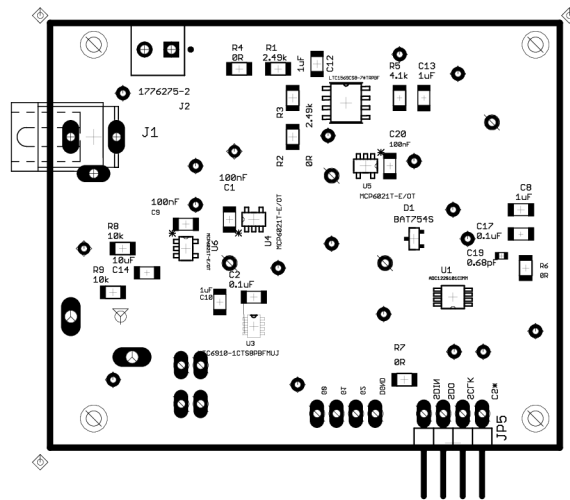
## SEZNAM PŘÍLOH

A Skript na simulaci v prostředí Matlab	59
B Návrh desky plošných spojů	60
C Fotografie desky	67
D Seznam součástek	68
E Obsah CD	69

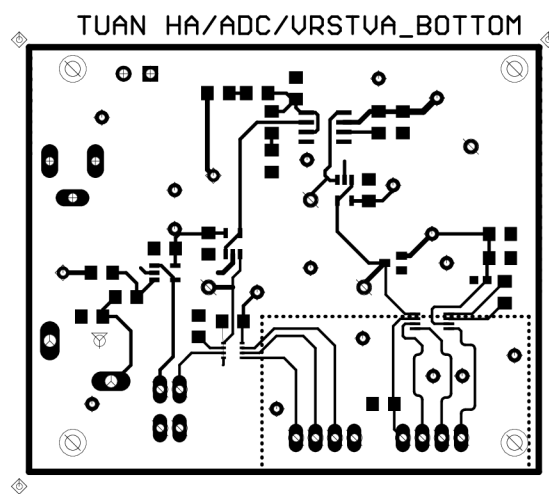
## A SKRIPT NA SIMULACI V PROSTŘEDÍ MATLAB

```
1
2 Fs = 10 *10^6; % vzorkovací frekvence
3 Ts = 1/Fs; % vzorkovací perioda - doba mezi 2 vzorky
4
5 dt1 = 0:Ts:0.2-Ts; % délka signálu(x souřadnice)
6 f1 = 20;%první
7 f4 = 60;%třetí
8 f2 = 100;%pata
9 f3 = 140;%sedma
10
11 %y = A*sin(2*pi*f+theta)
12
13 y1 = 1*sin(2*pi*f1*dt1);%první
14 y4 = 0.75*sin(2*pi*f4*dt1);%třetí
15 y2 = 0.7*sin(2*pi*f2*dt1);%pata
16 y3 = 0.45*sin(2*pi*f3*dt1);%sedma
17
18 y5 = y1 + y2 + y3 + y4;
19 Va = y5
20 L=length(Va);
21 NFFT = 2^nextpow2(L);
22 %kmitočtová normalizace + FFT
23 Y = fft(Va,NFFT)/L;
24 f = Fs/2*linspace(0,1,NFFT/2+1);
25
26 figure(3)
27 plot(dt1,y5);
28 xlabel('čas_t(s)', 'FontSize',12);
29 ylabel('okamžitá_hodnota_u(t)', 'FontSize',12);
30 %title('signál v časové oblasti');
31 grid on
32 figure(4)
33 semilogx(f,2*abs(Y(1:NFFT/2+1)));
34 xlabel('frekvence_[Hz]', 'FontSize',12);
35 ylabel('normalizovaná_amplituda_[-]', 'FontSize',12);
36 grid on
```

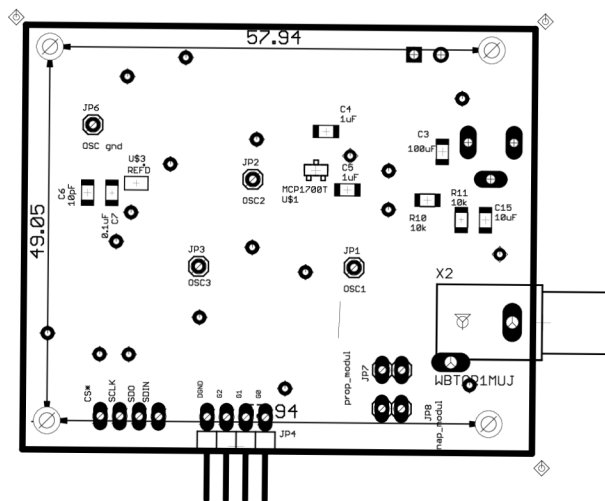
## B NÁVRH DESKY PLOŠNÝCH SPOJŮ



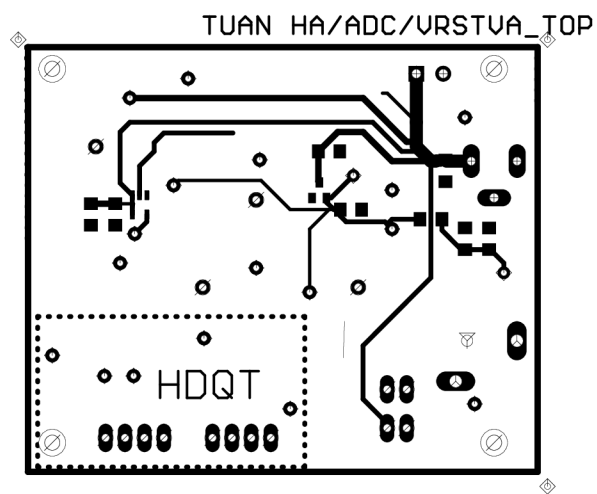
Obr. B.1: Spodní strana desky - osazovací výkres



Obr. B.2: Spodní strana desky - plošný spoj

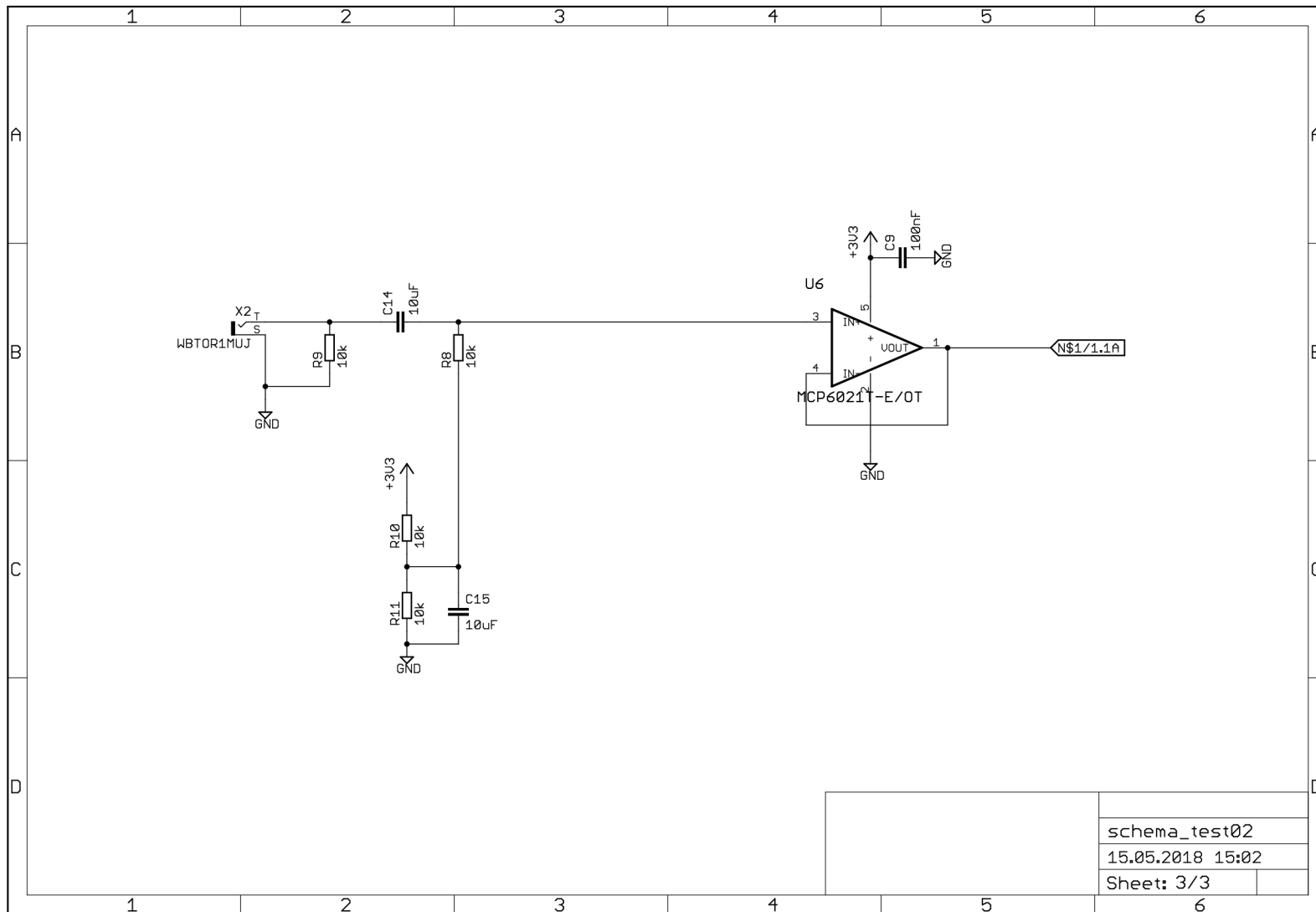


Obr. B.3: Horní strana desky - osazovací výkres



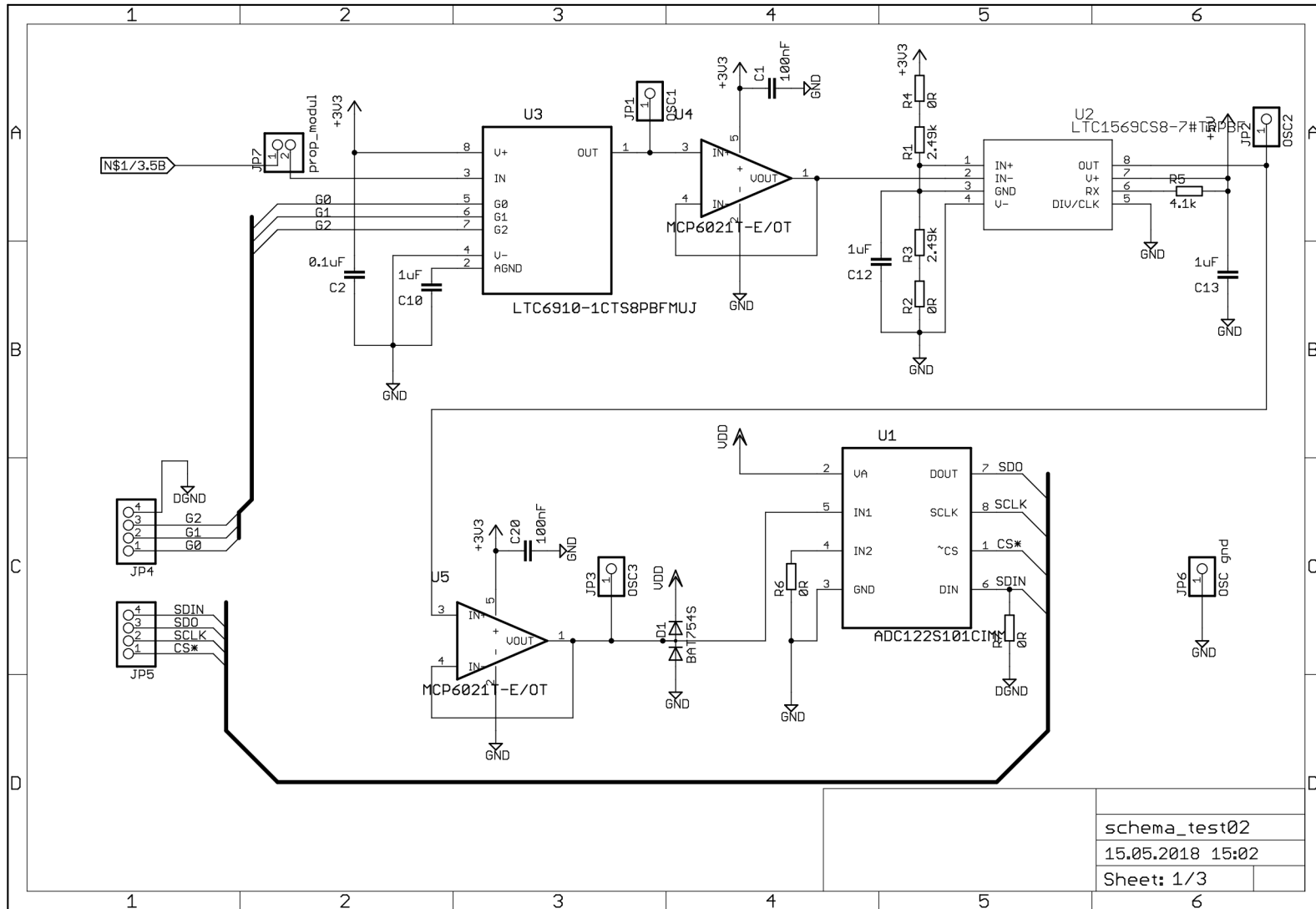
Obr. B.4: Horní strana desky - plošný spoj

79

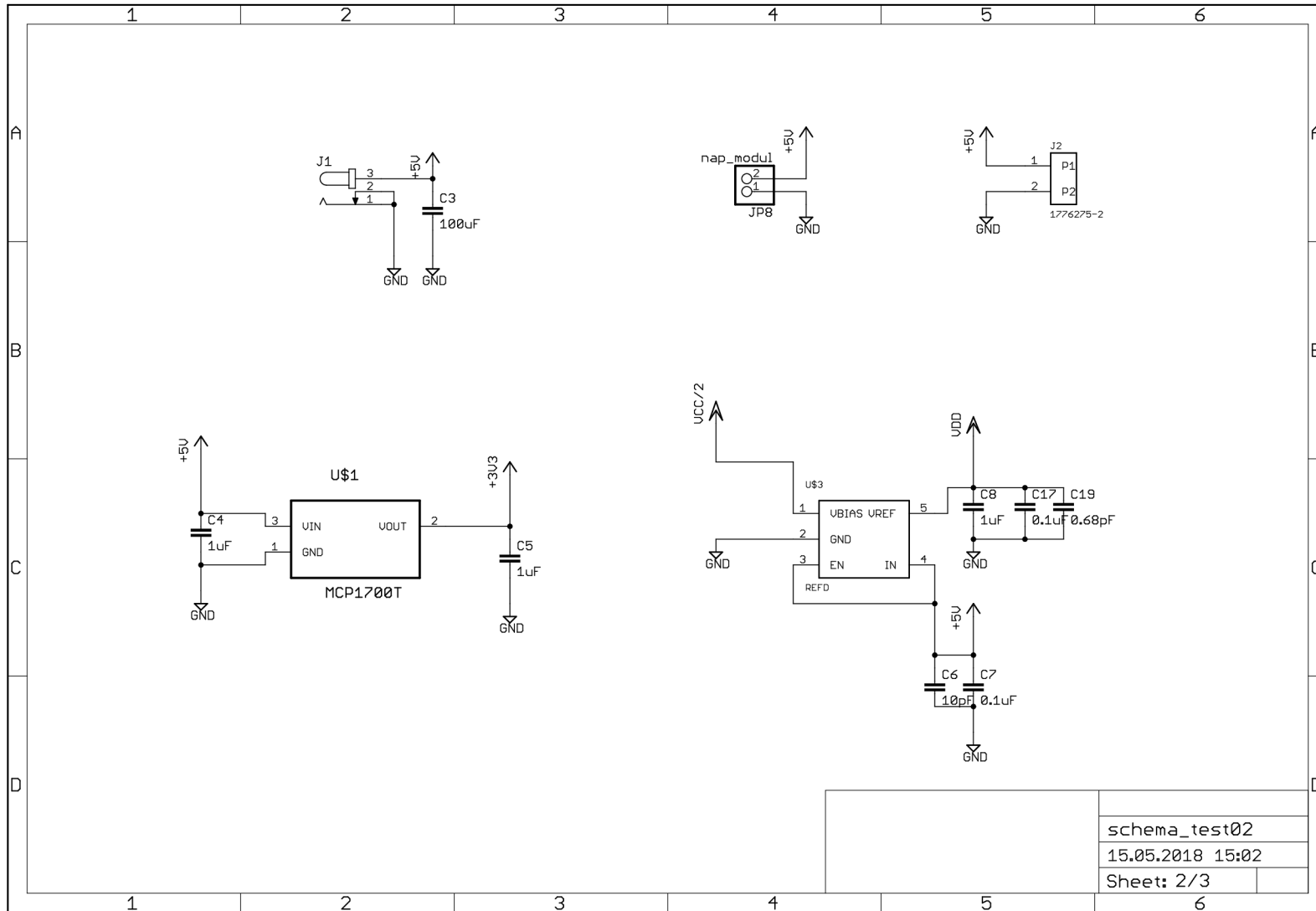


Obr. B.5: strana 1 (signálová část)





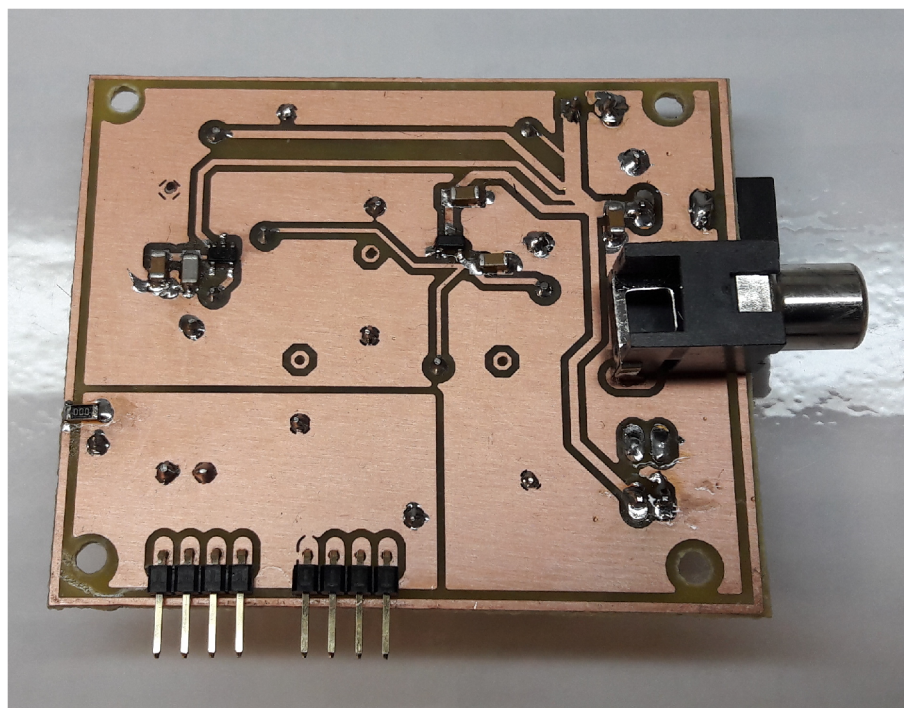
Obr. B.6: strana 2 (signálová část)



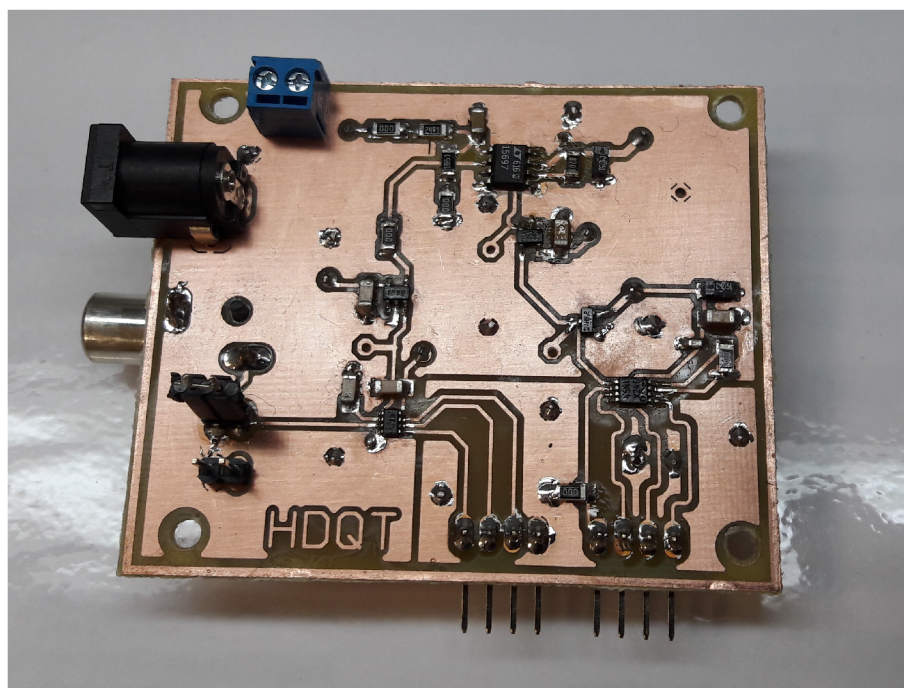
schema_test02	
15.05.2018 15:02	
Sheet: 2/3	

Obr. B.7: strana 1 (napájecí část)

## C FOTOGRAFIE DESKY



Obr. C.1: Horní strana desky - fotografie



Obr. C.2: Spodní strana desky - fotografie

## D SEZNAM SOUČÁSTEK

Tab. D.1: Seznam součástek

označení	hodnota	počet	popis
D1		1	BAT754S - Schottkyho dioda
U1		1	ADC122S101CIMM - ADC
R5	4.1k	1	R-EU_R1206
R1, R3	2.49k	2	R-EU_R1206
C4, C5, C8, C10, C12, C13	1uF	6	C-EUC1206
J2		1	1776275-2
JP7, JP3, JP2, JP1, JP6, JP8		6	pin header
U4, U5, U6		3	IC OP amp - sledovače
U\$1		1	LDO - linární regulátor 3,3 V
U3		1	Programovatelný zesilovač
U2		1	IC - AAF
C14, C15, C6	10uF	3	C-EUC1206
R8, R9, R10, R11	10k	4	R-EU_R1206
C1, C3, C9, C20	100nF	4	C-EUC1206
R2, R4, R6, R7	0R	4	R-EU_R1206
C19	0,68pF	1	C-EUC1206
C2,C7, C17	0,1uF	3	C-EUC1206
J1		1	Jack -plug
JP4, JP5	1x4/90	2	PIN header 4x

## E OBSAH CD

/	.....	kořenový adresář přiloženého CD
├	Eagle.....	návrhové soubory - vytvořené Eagle soft. verze 7.6
│	├	schema_test02.brd
│	└	schema_test02.sch
├	Obrazky.....	použité obrázky
├	Matlab.....	Matlabovský skript - vytvořený verzí R2015a
│	├	simulace01.m
├	Hlavni_program.....	Skript v jazyce Python
│	├	final_THD.py
├	BP_FEKT.pdf.....	soubor s bakalářskou prací v pdf formátu
│	├	prace.pdf