

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE  
FAKULTA ELEKTROTECHNICKÁ



## BAKALÁŘSKÁ PRÁCE

Optimální výpočet digitálního audio  
zesilovače

Lysá nad Labem, 2009

Autor: Vojtěch Polásek



## Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne \_\_\_\_\_

\_\_\_\_\_  
podpis

## **Poděkování**

Děkuji ing. Petru Kujanovi za odborné vedení bakalářské práce a za cenné rady.

## Abstrakt

Tato práce se zabývá problémem optimálního výpočtu digitálního audio zesilovače. Nejprve je zde rozebrán výpočet frekvenčního spektra liše prodlouženého signálu spektra obecného signálu, poté je zde ukázán způsob jak určit spínací časy PWM signálu z tohoto spektra. Dále je rozebrán problém výpočtu pro příliš dlouhý signál, rozdělení na kratší části a napojování dílčích PWM signálů. Nakonec je tento postup implementován do funkcí v systému Mathematica a vyzkoušen na krátké zvukové nahrávce.

## **Abstract**

This work deals with the problem of calculating the digital audio amplifiers. First, there is discussed the calculation of frequency spectrum odd extended signal range of general signal, then there is shown a way to determine the switching times of the PWM signal from the spectrum. Furthermore, it dismantled the problem of calculating for too long signal, divided into shorter parts and connecting sub PWM signals. Finally, this procedure is implemented to function in the Mathematica system and tested for a short audio recording.

vložit zadani





# Obsah

Seznam obrázků	ix
<b>1 Úvod</b>	<b>1</b>
<b>2 Digitální audio zesilovače</b>	<b>3</b>
<b>3 Frekvenční spektrum signálu</b>	<b>5</b>
3.1 Frekvenční spektrum lichého reálného diskrétního signálu . . . . .	5
3.2 Výpočet frekvenčního spektra v systému <i>Mathematica</i> . . . . .	8
<b>4 Výpočet PWM signálu</b>	<b>11</b>
4.1 Krátký signál . . . . .	11
4.1.1 Výpočet spínacích časů . . . . .	11
4.1.2 THD . . . . .	12
4.1.3 Filtrace PWM signálu . . . . .	13
4.1.4 Simulace . . . . .	16
4.2 Dlouhý signál . . . . .	20
4.2.1 Rozdělení dlouhého signálu na krátké části . . . . .	20
4.2.2 Napojení dílčích PWM . . . . .	21
4.2.3 Implementace v systému <i>Mathematica</i> . . . . .	22
<b>5 Zvukový signál</b>	<b>25</b>
5.1 Zvuk . . . . .	25
5.2 Doba výpočtu . . . . .	28
<b>6 Závěr</b>	<b>29</b>
<b>Literatura</b>	<b>31</b>



# Seznam obrázků

2.1	Základní schéma PWM modulátoru . . . . .	3
2.2	Princip PWM modulátoru . . . . .	4
3.1	Interpolace . . . . .	9
4.1	Spektrum vypočteného PWM signálu . . . . .	13
4.2	Závislost THD na amplitudě PWM . . . . .	14
4.3	Závislost THD na $n$ . . . . .	14
4.4	Frekvenční charakteristiky Butterworthova filtru různých řádů . . . . .	15
4.5	Filtrovaný PWM signál . . . . .	16
4.6	Vliv změny hodnoty $n$ na výsledný signál . . . . .	17
4.7	Vliv změny amplitudy PWM signálu . . . . .	18
4.8	Spektrum výsledného PWM signálu, kde $A = 5$ . . . . .	19
4.9	Spektrum výsledného PWM signálu, kde $A = 50$ . . . . .	19
4.10	Rozdělování dlouhého signálu . . . . .	20
4.11	Detail překrývání dílčích PWM signálů . . . . .	21
4.12	Napojení PWM . . . . .	22
4.13	Filtrace napojovaného PWM signálu . . . . .	23
5.1	Zvukový signál s různou amplitudou PWM . . . . .	26
5.2	Originál zvukové nahrávky . . . . .	26
5.3	Filtrovaný PWM . . . . .	27



# Kapitola 1

## Úvod

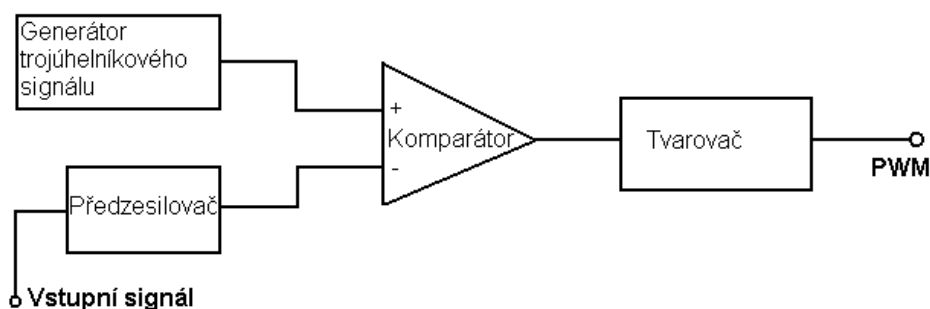
Úkolem této bakalářské práce je otestovat nový algoritmus výpočtu digitálního audio zesilovače. Nejprve zde stručně rozeberu princip, výhody a nevýhody digitálních audio zesilovačů označovaných jako class-D zesilovače, dále se budu zabývat matematickým rozбором frekvenčního spektra dikrétního signálu. Dale představím funkce pro výpočet PWM signálu jehož spektrum se shodovalo se zadaným spektrem audio signalem a navíc mělo dalších  $n$  harmonických nulových. Poté ukáži způsob jak napojovat dílčí PWM signály tak, aby nevznikaly deformace po filtraci a nakonec představím funkci která provede celý tento postup. V této práci jsem se zaměřil na názornost, je zde množství grafů a simulací, které graficky znázorňují daný problém.



# Kapitola 2

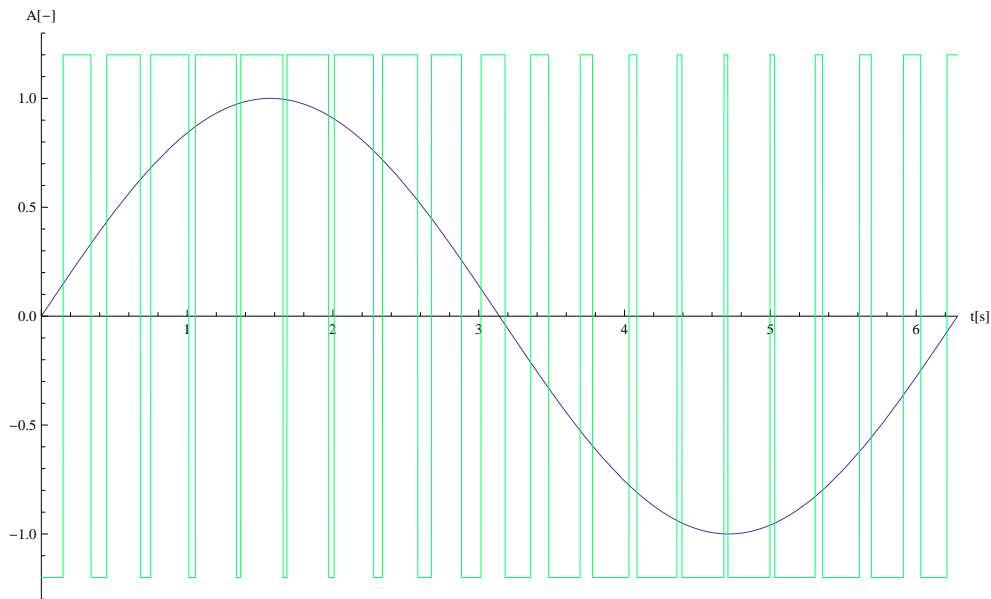
## Digitální audio zesilovače

Digitální audio zesilovače pracující ve třídě D (D-class zesilovače) se dnes využívají hlavně díky své vysoké účinnosti, která je teoreticky 100%. Jejich základem je převod audio signálu na pulzně šířkově modulovaný (PWM) signál (Pulse Width Modulation), který má stejnou střední hodnotu jako původní signál. Tento signál má konstantní periodu, která bývá řádově vyšší než nejvyšší harmonická frekvence v původním signálu a proměnnou střídu. V dnešní době se standardně používá převod na tento PWM signál pomocí PWM modulátoru, který se dá realizovat pomocí generátoru trojúhelníkového signálu a komparátoru, který porovnává trojúhelníkový signál s modulovaným signálem. Za komparátorem se ještě používá tvarovač pro lepší kolmost hran. Blokové schéma tohoto komparátoru je na obr. obr. 2.1. Více viz. [6].



Obrázek 2.1: Základní schéma PWM modulátoru

Ukázka PWM signálu je na obr. 2.2.



Obrázek 2.2: Princip PWM modulátoru

Takový to modulátor je možné realizovat buď analogově, nebo digitálně. V případě analogového řešení je třeba použít generátor trojúhelníkového signálu, který by měl mít co nejdokonalější průběh, důležitá je zejména linearita. Od kvality tohoto signálu se totiž odvíjí kvalita celého PWM signálu. Dále je také důležitá rychlost komparátoru, který vlastně detekuje časy změny amplitudy PWM signálu. V případě digitálního řešení tyto problémy odpadají. Celý postup je možné realizovat pomocí digitálních signálových procesorů (DSP). Perioda PWM signálu musí být dostatečně vysoká. V praxi se používají frekvence od stovek kHz do jednotek, i desítek MHz.

Jak už bylo řečeno, PWM signál má stejnou střední hodnotu jako původní modulovaný signál. Abychom získali tuto střední hodnotu signálu, je nutné výstup z PWM modulátoru připojit na filtr typu dolní propust. To je z toho důvodu, že PWM je obdélníkový signál, který má teoreticky nekonečné spektrum. Tyto vyšší harmonické frekvence je proto nutné odfiltrout.



# Kapitola 3

## Frekvenční spektrum signálu

### 3.1 Frekvenční spektrum lichého reálného diskrétního signálu

Zde si ukážeme souvislost frekvenčního spektra obecného diskrétního signálu a frekvenčního spektra téhož signálu, liše prodlouženého. Nejdříve si ukážeme jak vypadá frekvenční spektrum reálného diskrétního signálu definovaného posloupností  $x_k$ . Spektrum tohoto signálu  $X_k$  vyjadřuje Diskrétní Fourierova transformace, která je definována vztahem

$$X_k = \sum_{n=0}^{N_x-1} x_n e^{\frac{-2\pi i k n}{N_x}}; \quad k = 0, 1, \dots, N_x - 1, \quad (3.1)$$

kde  $N_x$  je počet hodnot signálu  $x_k$ . S využitím vztahu

$$e^{i\varphi} = \cos \varphi + i \sin \varphi \quad (3.2)$$

se tedy spektrum reálného signálu dá vyjádřit jako

$$X_k = \sum_{n=0}^{N_x-1} x_n \cos\left(\frac{-2\pi k n}{N_x}\right) + i \sum_{n=0}^{N_x-1} x_n \sin\left(\frac{-2\pi k n}{N_x}\right); \quad k = 0, 1, \dots, N_x - 1. \quad (3.3)$$

Dále si definujeme reálný diskrétní signál  $y_k$ , který je lichým prodloužením výše uvedeného signálu  $x_k$ . Pro tento lichý signál platí rovnice

$$y(k) = -y(N_y - k), \quad (3.4)$$

kde  $N_y$  je počet hodnot signálu  $y_k$ . Spektrum  $Y_k$  signálu  $y_k$  je tedy

$$Y_k = \sum_{n=0}^{N_y-1} y_n e^{\frac{-2\pi i k n}{N_y}}; \quad k = 0, 1, \dots, N_y - 1. \quad (3.5)$$

Tuto sumu můžeme rozdělit na součet dvou sum

$$Y_k = \sum_{n=0}^{(N_y-1)/2} y_n e^{\frac{-2\pi i k n}{N_y}} + \sum_{n=(N_y+1)/2}^{N_y-1} y_n e^{\frac{-2\pi i k n}{N_y}}; k = 0, 1, \dots, N_y - 1, \quad (3.6)$$

kde první suma vyjadřuje první polovinu signálu  $y_k$  a druhá suma vyjadřuje jeho liché prodloužení. Tuto rovnici můžeme s pomocí vztahu (3.2) ještě rozepsat na následující vztah

$$Y_k = \sum_{n=0}^{\frac{N_y-1}{2}} y_n \cos\left(\frac{-2\pi k n}{N_y}\right) + i \sum_{n=0}^{\frac{N_y-1}{2}} y_n \sin\left(\frac{-2\pi k n}{N_y}\right) + \sum_{n=\frac{N_y+1}{2}}^{N_y-1} y_n \cos\left(\frac{-2\pi k n}{N_y}\right) + i \sum_{n=\frac{N_y+1}{2}}^{N_y-1} y_n \sin\left(\frac{-2\pi k n}{N_y}\right). \quad (3.7)$$

Dále platí následující rovnosti

$$\cos(k\pi + \varphi) = \cos(k\pi - \varphi); \quad k \in \mathbb{Z}, \quad (3.8)$$

$$\sin(k\pi + \varphi) = -\sin(k\pi - \varphi); \quad k \in \mathbb{Z}. \quad (3.9)$$

Z nich plynou rovnice

$$\cos(k\pi(1+q)) = \cos(k\pi(1-q)); \quad k \in \mathbb{Z}, q \in \mathbb{R}, \quad (3.10)$$

$$\sin(k\pi(1+q)) = -\sin(k\pi(1-q)); \quad k \in \mathbb{Z}, q \in \mathbb{R}, \quad (3.11)$$

které vyjadřují symetričnost funkcí sinus a cosinus okolo bodů  $k\pi$ . Díky této skutečnosti a platnosti vztahu (3.4) můžeme spektrum diskrétního lichého signálu vyjádřit jako

$$Y_k = \sum_{n=0}^{\frac{N_y-1}{2}} y_n \cos\left(\frac{-2\pi k n}{N_y}\right) + i \sum_{n=0}^{\frac{N_y-1}{2}} y_n \sin\left(\frac{-2\pi k n}{N_y}\right) - \sum_{n=0}^{\frac{N_y-1}{2}} y_n \cos\left(\frac{-2\pi k n}{N_y}\right) + i \sum_{n=0}^{\frac{N_y-1}{2}} y_n \sin\left(\frac{-2\pi k n}{N_y}\right). \quad (3.12)$$

Zde je vidět že reálné části se odečtou a imaginární části se projeví 2 krát. Výsledná rovnice pro Fourierovo spektrum lichého signálu je tedy

$$Y_k = 2i \left( \sum_{n=0}^{(N_y-1)/2} y_n \sin\left(\frac{-2\pi k n}{N_y}\right) \right); \quad k = 0, 1, \dots, N_y - 1. \quad (3.13)$$

### 3.1. FREKVENČNÍ SPEKTRUM LICHÉHO REÁLNÉHO DISKRÉTNÍHO SIGNÁLU 7

Nyní si všimneme že platí

$$N_y = 2 \cdot N_x - 1. \quad (3.14)$$

Z této rovnosti dosadíme  $N_y$  do horní meze sumy. Také si připomeneme, že první polovina signálu  $y_k$  se shoduje se signálem  $x_k$ . Vztah (3.13) tedy můžeme přepsat do podoby

$$Y_k = 2i \left( \sum_{n=0}^{N_x-1} x_n \sin \left( \frac{-2\pi kn}{N_y} \right) \right); \quad k = 0, 1, \dots, N_x - 1. \quad (3.15)$$

Pomocí této rovnice se dá získat první polovina frekvenčního spektra liše prodlouženého signálu. Toto spektrum je reálné. Nyní si všimneme že tato rovnice se podobá imaginární části pravé části rovnice (3.3). Rozdíl je jen ve jmenovateli argumentu funkce sinus. Nyní si zavedeme upravenou diskretní Fourierovu transformaci s koeficientem  $b$  v exponenciále:

$$X_k = \sum_{n=0}^{N_x-1} x_n e^{\frac{-2\pi bkn}{N_x}}; \quad k = 0, 1, \dots, N_x - 1, \quad (3.16)$$

Rovnice (3.15) potom bude mít tvar

$$Y_k = 2i \left( \sum_{n=0}^{N_x-1} x_n \sin \left( \frac{-2\pi bkn}{N_y} \right) \right); \quad k = 0, 1, \dots, N_x - 1. \quad (3.17)$$

Když položíme

$$b = \frac{N_y}{N_x} \quad (3.18)$$

a dosadíme ho do rovnice (3.17), výsledek bude

$$Y_k = 2i \left( \sum_{n=0}^{N_x-1} x_n \sin \left( \frac{-2\pi kn}{N_x} \right) \right); \quad k = 0, 1, \dots, N_x - 1. \quad (3.19)$$

Nyní je vidět že výraz v závorce se shoduje s imaginární částí pravé části rovnice (3.3). Z toho plyne, že první polovina frekvenčního spektra liše prodlouženého signálu  $y_k$  je rovna dvojnásobku imaginární části spektra původního signálu  $x_k$ , při použití koeficientu  $b$  z rovnice (3.17). Vzhledem k tomu, že spektrum lichého reálného signálu je imaginární a také liché, dá se z frekvenčního spektra reálného diskretního signálu jednoduše získat spektrum příslušného liše prodlouženého signálu. První polovina frekvenčního spektra lichého signálu  $y_k$  se tedy dá vypočítat z frekvenčního spektra signálu  $x_k$  pomocí rovnice

$$Y = 2 \operatorname{Im}(X_b), \quad (3.20)$$

kde  $Y$  je spektrum signálu  $y_k$  a  $X_b$  je spektrum signálu  $x_k$  vypočteného z upravené rovnice pro Fourierovu transformaci (3.16) s dosazením koeficientu  $b$  z rovnice (3.18).

## 3.2 Výpočet frekvenčního spektra v systému *Mathematica*

Systém *Mathematica* [7] umožňuje vypočítat frekvenční spektrum diskrétního signálu pomocí funkce

```
fs = Fourier[s, FourierParameters -> {a, b}];,
```

kde  $\mathbf{s}$  je posloupnost reálných hodnot a  $\mathbf{fs}$  je příslušné frekvenční spektrum, které představuje taktéž posloupnost hodnot, jejichž počet je stejný jako počet hodnot signálu  $\mathbf{s}$  a je obecně komplexní. Volba `FourierParameters` umožňuje počítat s koeficientem  $\mathbf{b}$  v exponenciále tak jako je to uvedeno v rovnici (3.16), koeficient  $\mathbf{a}$  má význam normalizace. Při defaultních hodnotách je  $\mathbf{a} = 0$ ,  $\mathbf{b} = 1$ . Pro spektrum diskrétního signálu vypočítané pomocí diskrétní Fourierovy transformace platí následující tvrzení:

- spektrum reálného sudého signálu je reálné a sudé,
- spektrum reálného lichého signálu je imaginární a liché,
- spektrum reálného signálu, který není sudý ani lichý je komplexní.

Další vlastnosti a symetrie signálů a jejich frekvenčních spekter jsou podrobněji popsány např. v [1]. Chceme-li tedy získat reálné frekvenční spektrum, je nutné mít signál buď sudý, nebo lichý, ze kterého bychom vzali imaginární část. Vzhledem k tomu, že funkce popsaná v předešlé části počítá s reálným spektrem, které má nulovou stejnosměrnou složku, bude výhodné pracovat s lichými signály. Pokud tedy máme lichý signál, frekvenční spektrum vypočtené pomocí funkce `Fourier` bude ryze imaginární. Z tohoto komplexního spektra určíme reálné spektrum jednoduše vynásobením konstantou  $(-i)$ . Další možností je využití vztahu (3.20) a vypočítat reálné frekvenční spektrum z frekvenčního spektra obecného signálu. Vzorec pro výpočet tohoto spektra zapsaný v systému *Mathematica* vypadá následovně:

```
b = Length[s]/(Length[s] 2 - 1);
fs = 2 Im[Fourier[s, FourierParameters -> {1, b}]];
```

Jedná se tedy o první polovinu spektra lichého signálu. Dále si všimneme, že první prvek ve spektru je vždy nulový (stejnosměrná složka lichého signálu je 0). S touto nulou nebudeme dále počítat a odstraníme jí příkazem `Drop`. Pokud bychom chtěli interpolovat

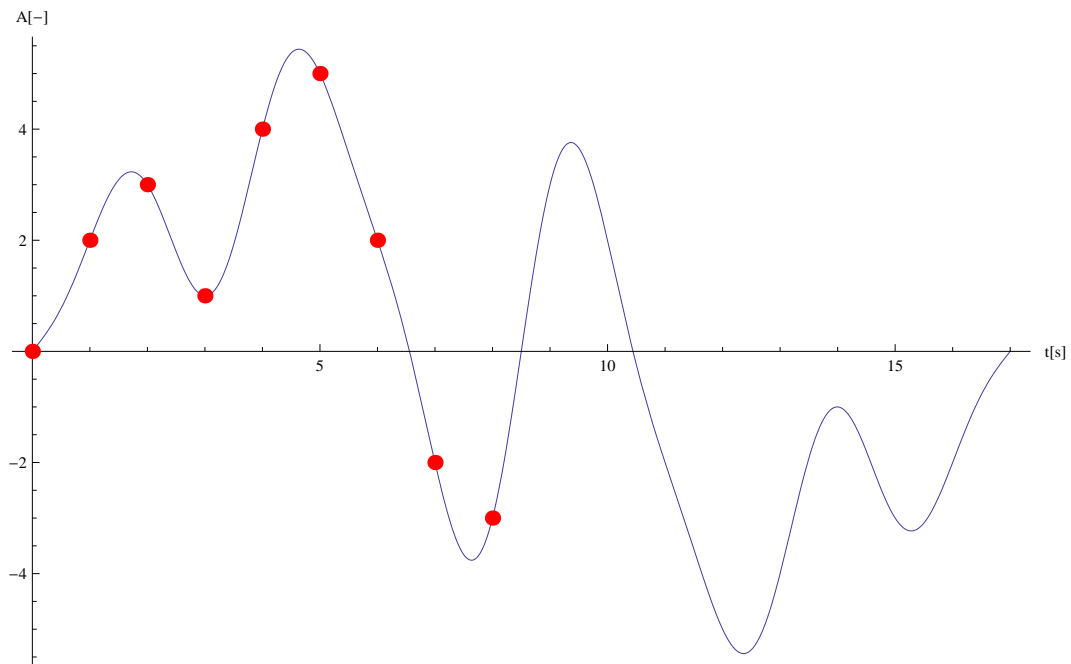
signál z jeho frekvenčního spektra, musíme toto spektrum ještě vynásobit konstantou  $1/n$ , kde  $n$  je počet prvků spektra lichého signálu. My máme ale vypočtenou pouze jeho první polovinu. Díky platnosti rovnice (3.11) a díky tomu že druhá polovina spektra lichého signálu je pouze převrácená první polovina s opačným znaménkem, dá se výše uvedený kód přepsat do tvaru:

```
b = Length[s]/(Length[s] 2 - 1);
fs = Drop[(1/(Length[s] 2 - 1))
  4 Im[Fourier[s, FourierParameters -> {1, b}]]];
```

Alternativní zápis by mohl být např.

```
b = Length[s]/(Length[s] 2 - 1);
fs = 2 Drop[2 koef
  Im[Fourier[worksignal, FourierParameters -> {-1, b}]], 1];
```

Tato část kódu tedy vypočítá frekvenční spektrum, které je dvojnásobkem frekvenčního spektra lichého signálu  $s$  a je normované výše uvedeným koeficientem  $1/n$ .  $fs$  také neobsahuje nulu na začátku. Na obr. 3.1 je vidět, že se signál interpolovaný z frekvenčního spektra  $fs$  shoduje v okamžicích vzorkování s původním signálem.



Obrázek 3.1: Interpolace



# Kapitola 4

## Výpočet PWM signálu

Algoritmus výpočtu spínacích časů PWM signálu z frekvenčního spektra je detailně popsán např. v [3, 4, 5]. Tento algoritmus vypočítává spínací časy PWM signálu tak, aby měl shodné frekvenční spektrum s původním signálem a navíc měl další vyšší harmonické frekvence nulové. Tento algoritmus mi byl dodán v sadě funkcí naprogramovaných v systému *Mathematica*.

### 4.1 Krátký signál

Nejprve se budem zabývat signálem, který má takový počet vzorků, že z něho jako z celku dokážeme pomocí níže uvedených funkcí vypočítat spínací časy odpovídajícího PWM signálu. To znamená počet vzorků menší než přibližně 50.

#### 4.1.1 Výpočet spínacích časů

Nejdůležitější je funkce, která z reálného frekvenčního spektra zadaného jako posloupnost reálných hodnot a dalších parametrů vypočítá spínací časy PWM signálu. Její zápis je

```
alfa = SolveML2AlfaHankel[fs, n, A, T, WorkingPrecision];,
```

kde užití symboly znamenají

- **alfa** - sada časů, ve kterých dojde ke změně amplitudy PWM signálu,
- **fs** - reálné frekvenční spektrum diskrétního signálu,
- **n** - počet harmonických, které mají být nulové, případně které mají přesně odpovídat spektru **fs**,

- $A$  - amplituda PWM signálu,
- $T$  - délka lichého signálu,
- `WorkingPrecision` - je volba, která určuje přesnost výpočtu.

Parametr `fs` představuje sadu hodnot frekvenčního spektra diskrétního signálu. Není třeba ho počítat z lichého signálu, ale stačí použít spektrum vypočtené z rovnice (3.20). Jeho délka by měla korespondovat s parametrem  $n$ , který rovněž odpovídá počtu spínacích časů. Hodnota THD popsaná níže klesá se zvyšováním  $n$ . Parametr  $n$  je použitelný zhruba do hodnoty 150, kde výpočet trvá již poměrně dlouho a projevuje se numerická nestabilita výpočtu, kterou lze částečně ovlivnit pomocí volby `WorkingPrecision`.

Pomocí této funkce tedy vypočteme spínací časy pro PWM signál, jehož frekvenční spektrum se shoduje s původním zadaným spektrem audio signálu. Navíc další hodnoty tohoto spektra jsou nulové až do hodnoty  $n$ . Ostatní vyšší hodnoty spektra jsou neřiditelné a mají náhodnou hodnotu.

Důležitá je také volba amplitudy  $A$ . Při různých simulacích se ukázalo, že amplitudu stačí volit větší, nebo rovnu 1.25 násobku maximální hodnoty v původním signálu. Při nižších hodnotách je možné, že funkce nenalezne řešení. S velikostí  $A$  souvisí i hodnota THD a to tak, že při rostoucí amplitudě roste i THD. Takže nejlepší volba je zvolit  $A$  nejnižší, avšak takové, aby funkce našla řešení.

Jak vypadá frekvenční spektrum signálu vypočteného pomocí funkce `SolveML2AlphaHankel` ukazuje obr. 4.1. Zadané spektrum je  $fs = \{1, 3, -2\}$ ,  $n = 30$ ,  $A = 5$ .

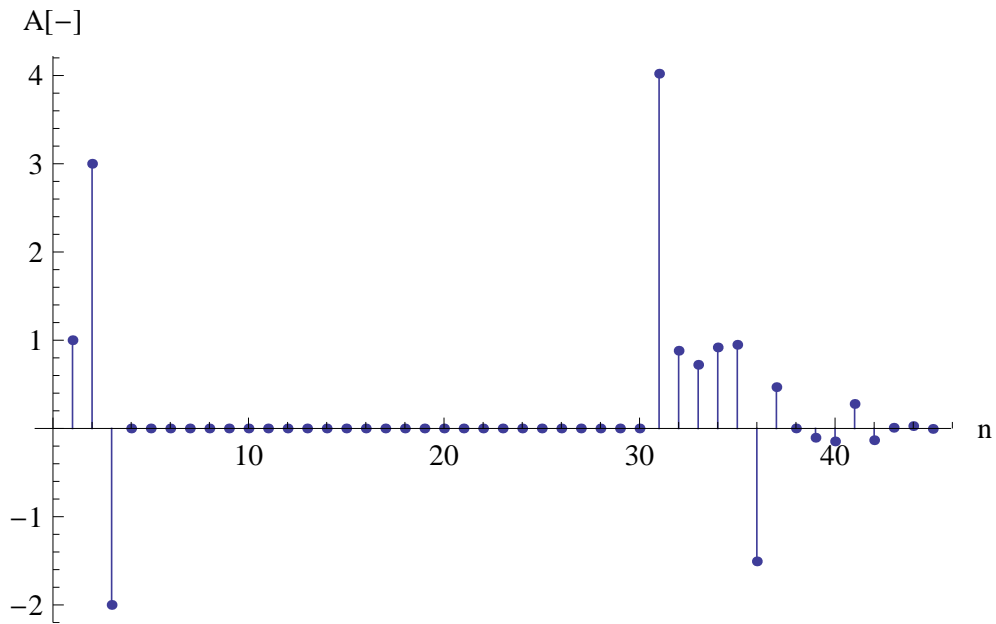
V tomto případě je hodnota  $THD = 11.5$ . Na tomto obrázku je vlevo vidět užitečná část frekvenčního spektra, která se shoduje s původním spektrem, uprostřed je nulová část spektra, která sahá až do počtu  $n$  harmonických a napravo je neřiditelná část spektra. Pokud se tuto část podaří dobře odfiltrovat dolnofrekvenční propustí, výsledkem bude signál, jehož frekvenční spektrum se shoduje se zadaným signálem, který má spektrum `fs`.

### 4.1.2 THD

THD (Total harmonic distortion) je celkové harmonické zkreslení definované podle vztahů uvedených např. na [3, 4]. Čím je THD vyšší, tím je zkreslení větší. V našem případě THD odpovídá přesnosti shody s původním signálem. Výpočet THD umožňuje funkce

$$THD = ML20ddTHD[alfa, fs, A, T, n, Nn];,$$





Obrázek 4.1: Spektrum vypočteného PWM signálu

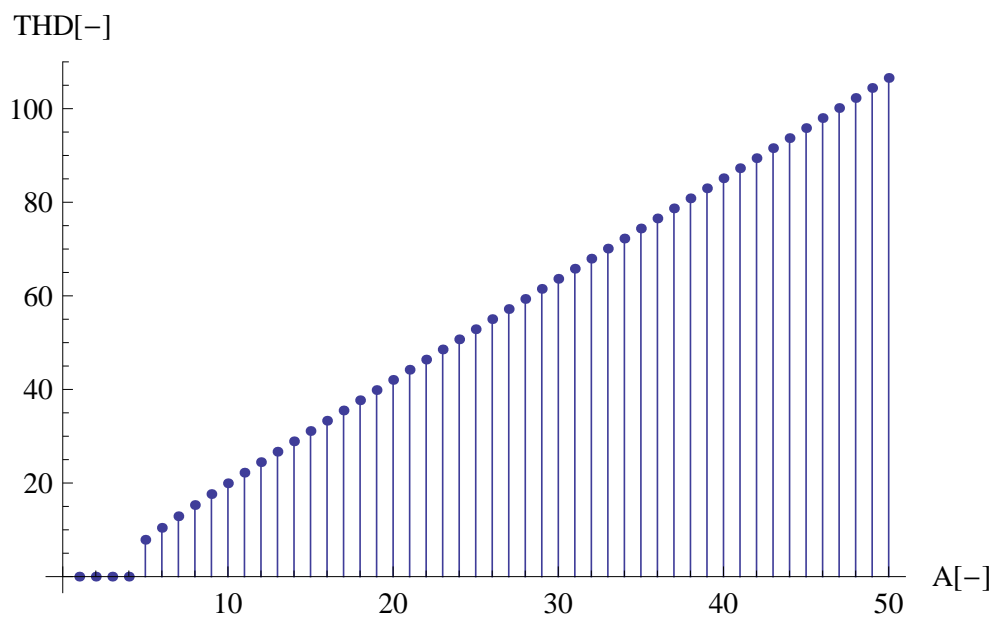
Parametry mají stejný význam jako u funkce `SolveML2AlphaHankel`. Na obr. 4.2 je vidět jak se mění THD v závislosti na velikosti amplitudy  $A$ . Ostatní parametry byly použity stejné jako pro obr. 4.1.

Pro hodnoty amplitud, kde je THD zdánlivě rovno nule ve skutečnosti nebylo nalezeno řešení, čili dané hodnoty amplitud jsou nepoužitelné. Z ostatních hodnot THD je vidět, že roste lineárně se zvyšující se amplitudou PWM. Následující obrázek 4.3 ukazuje závislost THD na zvyšujícím se  $n$ . Zbylé parametry jsou opět stejné jako v předchozích případech.

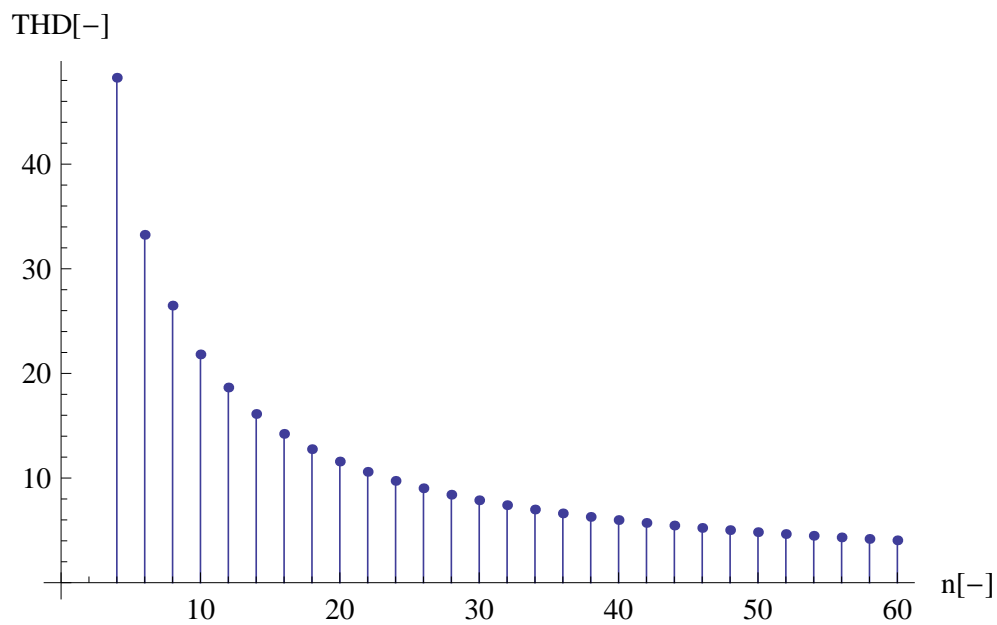
Zde je zřetelně vidět, že se snižujícím se  $n$  roste THD exponenciálně. Připomeňme, že volit  $n$  menší než je počet prvků ve spektru nemá smysl, protože se pak část spektra výsledného signálu nebude shodovat s původním. Hodnotu  $n$  je potřeba volit vzhledem k THD a také je potřeba přihlídnout ke sklonu frekvenční charakteristiky frekvenčního filtru.

### 4.1.3 Filtrace PWM signálu

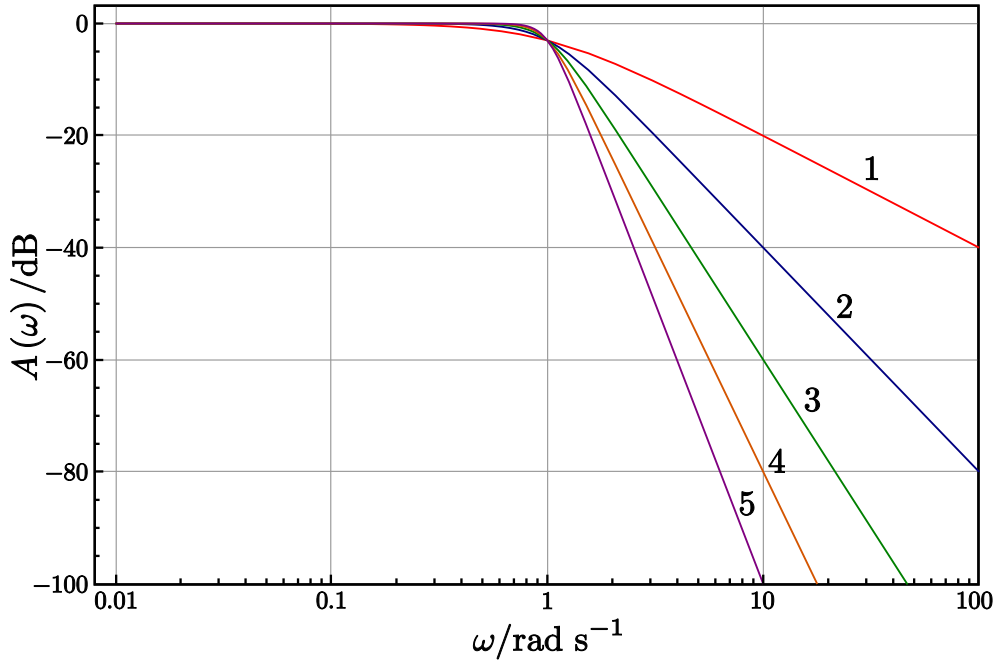
Pokud už máme PWM signál vypočítaný pomocí funkce `SolveML2AlphaHankel`, je potřeba ho odfiltrovat filtrem typu dolní propust, aby se potlačila neřiditelná část spektra. Čím bude větší odstup užitečného spektra od neřiditelné části, tím lépe frekvenční filtr na výstupu tuto nekontrolovanou část odstraní. Na obr. 4.4 jsou vidět frekvenční charakte-



Obrázek 4.2: Závislost THD na amplitudě PWM



Obrázek 4.3: Závislost THD na n



Obrázek 4.4: Frekvenční charakteristiky Butterworthova filtru různých řádů

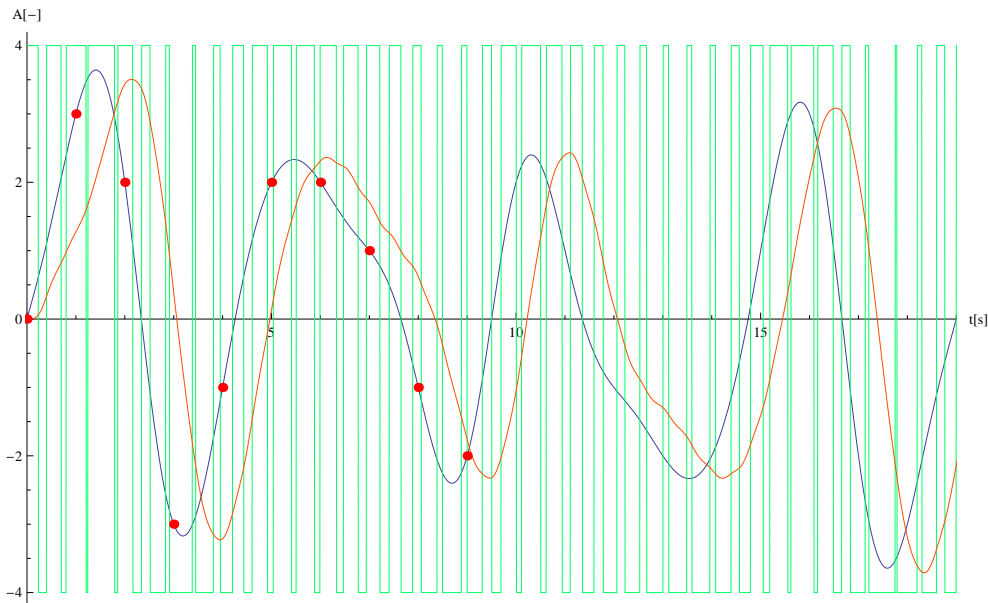
řistiky Butterworthova filtru různých řádů.

Z tohoto obrázku je například vidět, že při použití Butterworthova filtru řádu 4 dosáhneme útlumu nekontrolované části spektra o 80dB při použití hodnoty  $n$  10x větší než je počet prvků ve spektru  $f_s$ . Důležitá je také volba zlomové frekvence tohoto filtru. Zlomová frekvence by teoreticky měla být stejná jako frekvence nejvyšší složky v užitečné části frekvenčního spektra. Kvůli zaoblení charakteristiky okolo frekvence zlomu by ovšem měla být vyšší a to v závislosti na řádu filtru. Vzorkovací frekvence je

$$f_{vz} = \frac{1}{T_{vz}}, \quad (4.1)$$

kde  $T_{vz}$  je perioda mezi vzorky signálu. Jednotlivé hodnoty spektra jsou od sebe vzdáleny o tzv. frekvenční bin, jehož hodnota je  $1/(T_{vz} \cdot n_s)$ , kde  $n_{fs}$  je počet hodnot frekvenčního spektra signálu  $s$ . Počet hodnot signálu  $s$  je taktéž roven  $n_{fs}$ . Nejvyšší harmonická frekvence v signálu  $s$  je frekvence poslední hodnoty frekvenčního spektra tohoto signálu. Podle vzorkovací věty musí být vzorkovací frekvence nejméně 2x větší než maximální frekvence. Z toho plyne že maximální frekvence v signálu  $s$  je

$$f_{max} = \frac{1}{2} f_{vz} = \frac{1}{2T_{vz}}. \quad (4.2)$$



Obrázek 4.5: Filtrovaný PWM signál

Zlomovou frekvenci filtru  $f_c$  je vhodné volit za  $f_{max}$ , např.

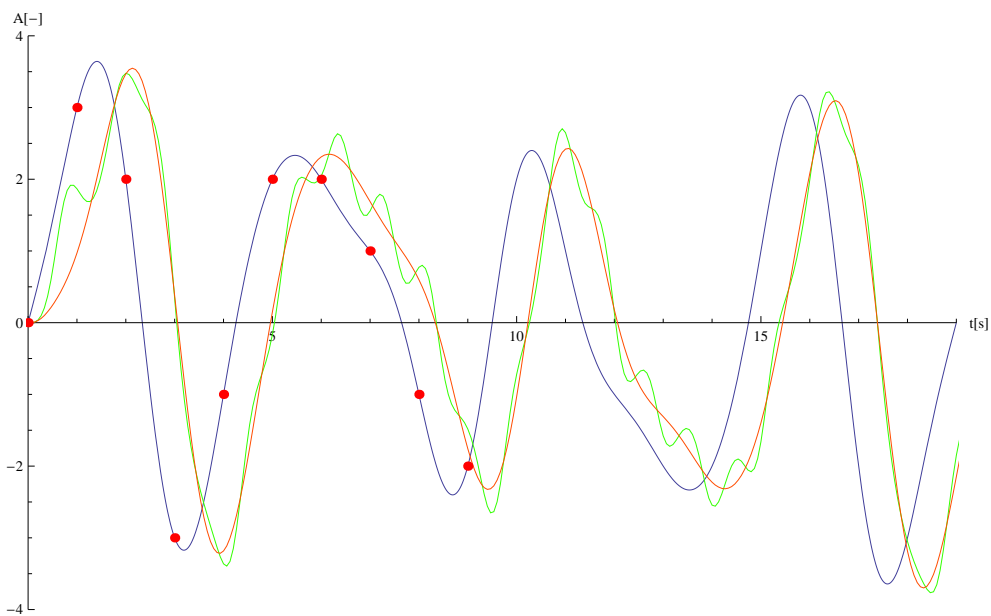
$$f_c = 1.2f_{max} = \frac{1.2}{2}f_{vz} = \frac{1.2}{2T_{vz}}. \quad (4.3)$$

U digitální filtrace je nutné vstupní signál nejdříve navzorkovat. To s sebou přináší problém, neboť vzorkování PWM signálu musí být velmi jemné, aby vzorky co nejpřesněji odpovídaly PWM signálu. Vzorkovací frekvenci je tedy nutné volit poměrně vysokou, např. jako tisícinásobek  $f_{vz}$ . Pro všechny následující filtrace bude použit filtr typu Butterworth 4. řádu se zlomovou frekvencí podle (4.3). Frekvence vzorkování PWM signálu bude volena jako 3000 násobek  $f_{vz}$ . Při reálné aplikaci výpočtu PWM a následné reprodukce odpadne potřeba digitální filtrace tohoto PWM signálu. Dolnofrekvenční filtr bude realizován jako analogový hardwarový prvek, který bude provádět filtraci v reálném čase. Díky omezenému frekvenčnímu rozsahu reproduktorů je navíc možné je přímo využít jako dolnofrekvenční filtr.

#### 4.1.4 Simulace

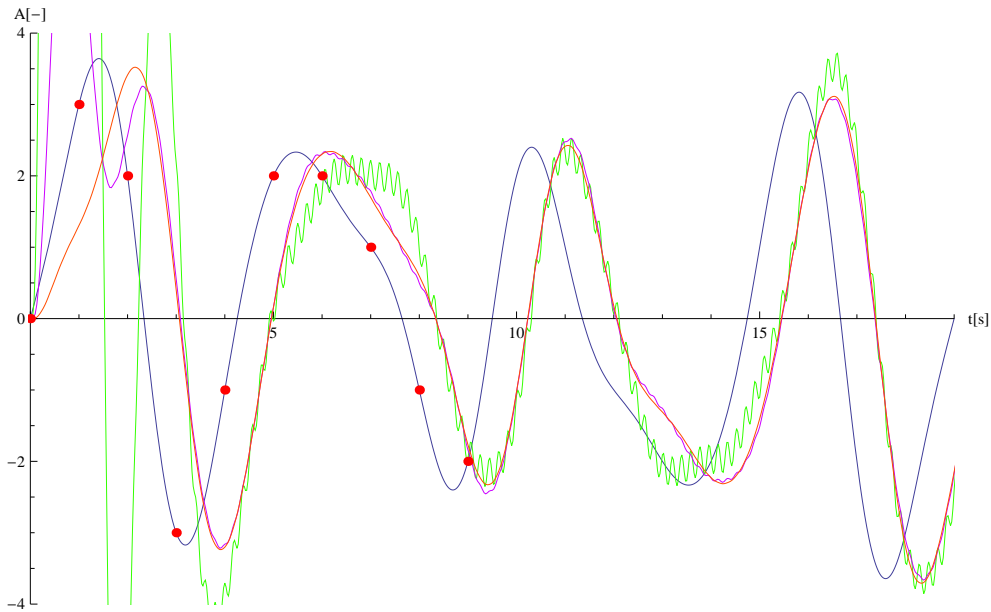
Celý postup výpočtu spínacích časů PWM signálu tedy spočívá ve výpočtu frekvenčního spektra zadaného signálu, následný výpočet spínacích časů a filtrace. Výsledek tohoto postupu je znázorněn na obrázku obr. 4.5.

Vzorky původního signálu  $\mathbf{s} = \{0, 3, 2, -3, -1, 2, 2, 1, -1, -2\}$ ; jsou znázorněny červenými body, interpolovaný signál je znázorněn modře, vypočtený PWM signál je zelený a výsledný signál po filtraci je červený. Počet vzorků je 10, hodnota  $n$  zde byla použita 40 (4 krát větší než počet vzorků), amplituda  $A$  PWM signálu je 4, perioda mezi vzorky je 1s. Je zde také patrné zpoždění signálu, které způsobuje dolnofrekvenční filtr, který je typu Butterworth. Na následujícím obrázku si ukážeme, jak vypadá výsledný signál při volbě různého počtu sepnutí PWM signálu  $n$ .



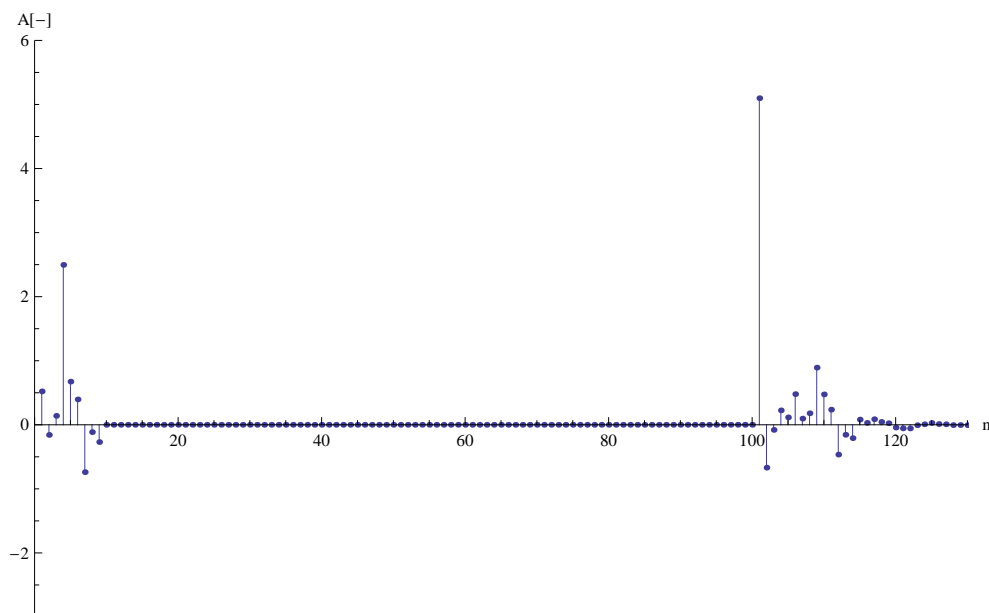
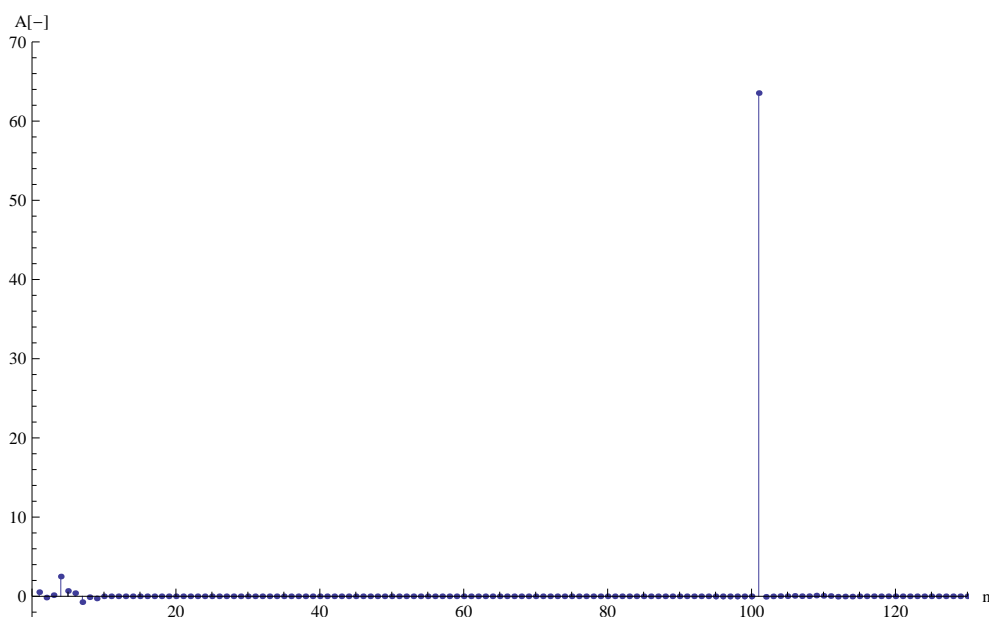
Obrázek 4.6: Vliv změny hodnoty  $n$  na výsledný signál

Modře je znázorněn původní signál, zeleně signál, kde bylo  $n = 20$  a červeně je signál, kde  $n$  bylo 100. Počet vzorků je 10, čili počet hodnot užitečného spektra je taktéž 10. Jako výhodná volba hodnoty  $n$  se ukazuje právě 10ti násobek velikosti užitečného spektra při filtraci filtrem typu Butterworth řádu 4. Při nižších hodnotách  $n$  je nekontrovaná část spektra nedostatečně utlumena a dochází k deformaci signálu. Hodnota  $n$  nemůže být libovolně velká. Optimální je do hodnoty 150. Při vyšších hodnotách trvá výpočet příliš dlouho, dokonce řešení nemusí být nalezeno vůbec. Dále si ukážeme jak se mění výsledný signál při různé velikosti amplitudy PWM signálu. Na obr. 4.7 je zobrazen výsledný průběh pro tři různé  $A$  při  $n = 100$ .

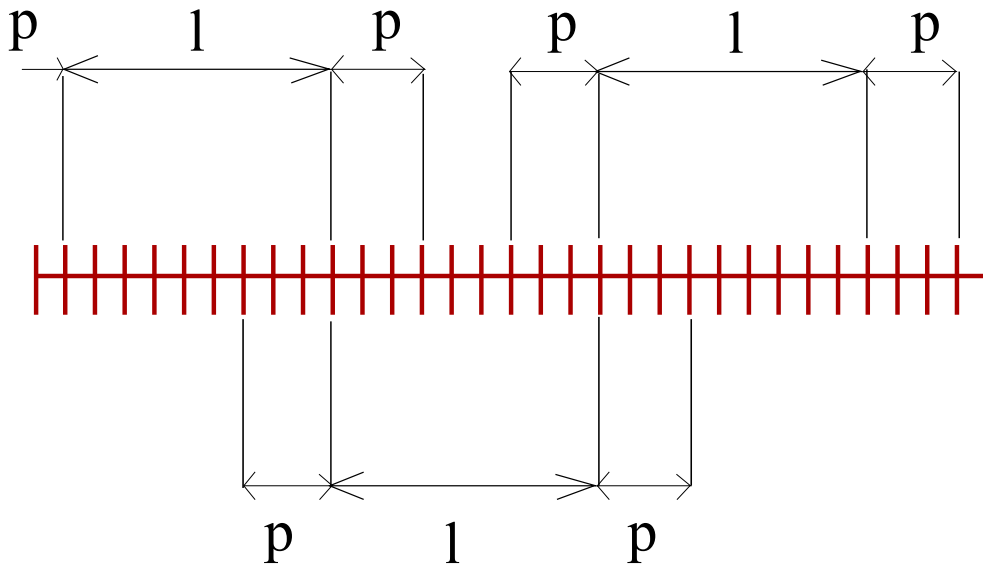


Obrázek 4.7: Vliv změny amplitudy PWM signálu

Modře je znázorněn původní signál, zeleně signál s  $A = 1000$ , fialově signál s  $A = 100$ , červeně signál s  $A = 10$ . Maximální amplituda v původním signálu je 3. Největší zkreslení je při zvětšující se amplitudě na začátku signálu. Dále se objevuje zvlnění signálu s velkou amplitudou  $A$ . To je způsobeno velmi vysokou hodnotou spektra v neřiditelné části. Hodnota spektrální čáry, která je na pozici  $n+1$  je totiž při velké amplitudě velmi vysoká. Souvisí s tím také hodnota THD. Dále zobrazeno spektrum PWM signálu vypočtené z výše uvedeného signálu  $s$ . Na obr. 4.8 je frekvenční spektrum pro amplitudu  $A = 5$  (THD = 6), na obr. 4.9 pro amplitudu  $A = 50$  (THD = 74).

Obrázek 4.8: Spektrum výsledného PWM signálu, kde  $A = 5$ Obrázek 4.9: Spektrum výsledného PWM signálu, kde  $A = 50$ 

Při možnosti volby budeme volit  $A$  co nejmenší. Problémy s příliš velkou  $A$  nastanou např. u zvukového signálu, kde rozlišení vzorků je obvykle 16bit. To znamená, že maximální amplituda tohoto zvukového signálu se pohybuje v rozmezí  $\pm 32768$ .



Obrázek 4.10: Rozdělování dlouhého signálu

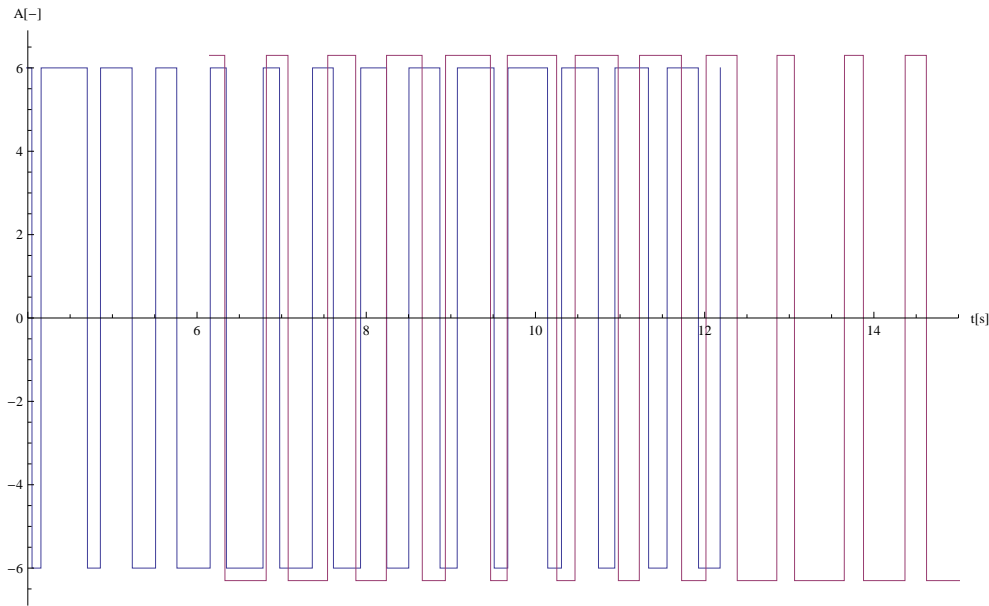
## 4.2 Dlouhý signál

Vzhledem k omezením která jsou popsána výše rozumíme dlouhým signálem takový signál, ze kterého nedokážeme pomocí výše uvedených funkcí vypočítat PWM signál, nebo by byl výsledný signál po filtraci příliš zkreslený.

### 4.2.1 Rozdělení dlouhého signálu na krátké části

Dosud jsme si popsali postup jak vypočítat optimální PWM signál do maximálního počtu vzorků přibližně 50. Už i v takovém případě by při  $n = 150$  docházelo k nedostatečnému utlumení necontrolované části spektra. V praxi se ovšem vyskytují signály s ještě mnohem větším počtem vzorků. Typicky na hudebním CD je vzorkovací frekvence 44100kHz. To znamená že na 1 sekundu záznamu je 44100 vzorků. Pokud bychom chtěli vypočítat výše uvedenými postupy PWM signál, je třeba dlouhý signál rozdělit na krátké části, ze kterých už vypočítat PWM umíme. Samotné rozdělení by ještě tak náročné nebylo, problém nastane při spojování dílčích PWM částí. Prosté umístění PWM signálů za sebe není možné, protože výše uvedený postup uvažuje vždy s nulovým prvním vzorkem. To je způsobeno tím, že se uvažuje liše prodloužený signál, pro který platí rovnice (3.14). Pro napojování budem muset uvažovat s jistým přesahem napojovaných signálů. Dlouhý signál rozdělíme na bloky s počtem prvků  $l$  a s přesahem  $p$  prvků na každou stranu. Celková délka bloku tedy bude  $l + 2p$ . Naznačeno je to na následujícím obrázku.





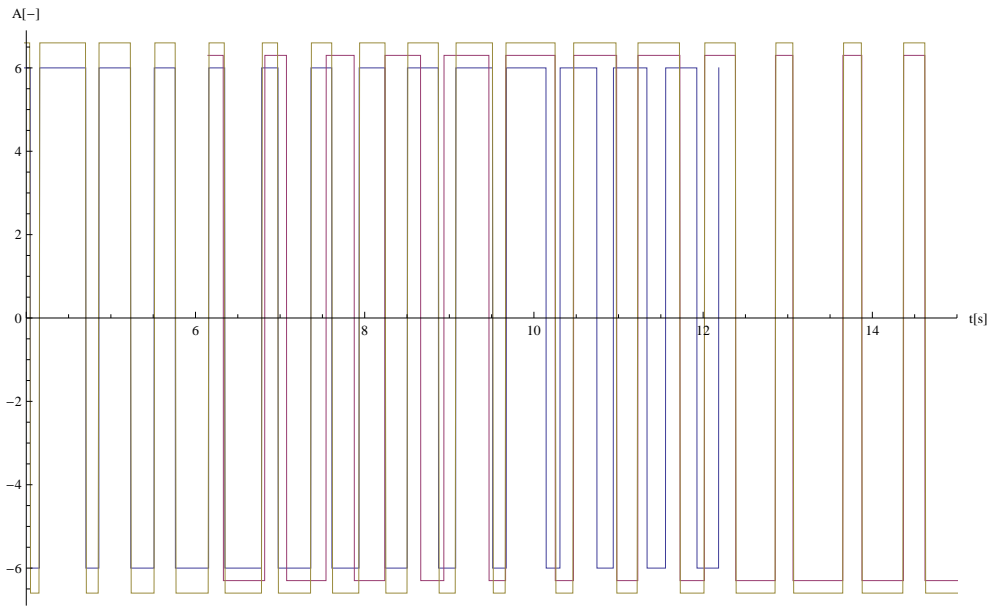
Obrázek 4.11: Detail překrývání dílčích PWM signálů

Zde je  $l = 10, p = 3$ . Jak vypadá odpovídající překrývání dílčích PWM signálů ukazuje obr. 4.11.

Na obrázku mají oba signály různou amplitudu. To je kvůli přehlednosti, ve skutečnosti ji mají oba signály stejnou.

### 4.2.2 Napojení dílčích PWM

První možností je napojit PWM uprostřed části, kde se překrývají. Toto je jednoduché, avšak může dojít ke špatnému napojení. PWM signál totiž vypočteme jako množinu časů, ve kterých dojde k převrácení amplitudy buď z  $A$  na  $-A$ , nebo obráceně. Na obrázku výše by mohl nastat případ, kdy bychom vzali PWM (spínací časy) do  $t = 11s$  z modré části, a od  $t = 11s$  z fialové části. Ve fialovém signálu má v prvním spínacím čase po  $t = 11s$  dojít ke změně amplitudy z  $A$  na  $-A$ , avšak díky tomu, že ve fialové části se v posledním čase před  $t = 11s$  mění amplituda také z  $A$  na  $-A$ , v následujícím spínacím čase by došlo k převrácení amplitudy. Tohoto případu je tedy nutné se vyvarovat. Pravděpodobně nejvýhodnější místo napojení bude tam, kde se spínací časy obou překrývaných signálů shodují, nebo alespoň jsou si velmi podobné a nedochází k výše popsanému problému. Na obr. 4.11 se shodují v čase  $t=9.7s$ . Algoritmus, který jsem navrhl, najde v obou překrývaných PWM signálech spínací časy, které jsou si nejbližší a zároveň tam nenastává problém s převrácením amplitudy (u obou signálů musí v tom čase dojít ke stejné změně



Obrázek 4.12: Napojení PWM

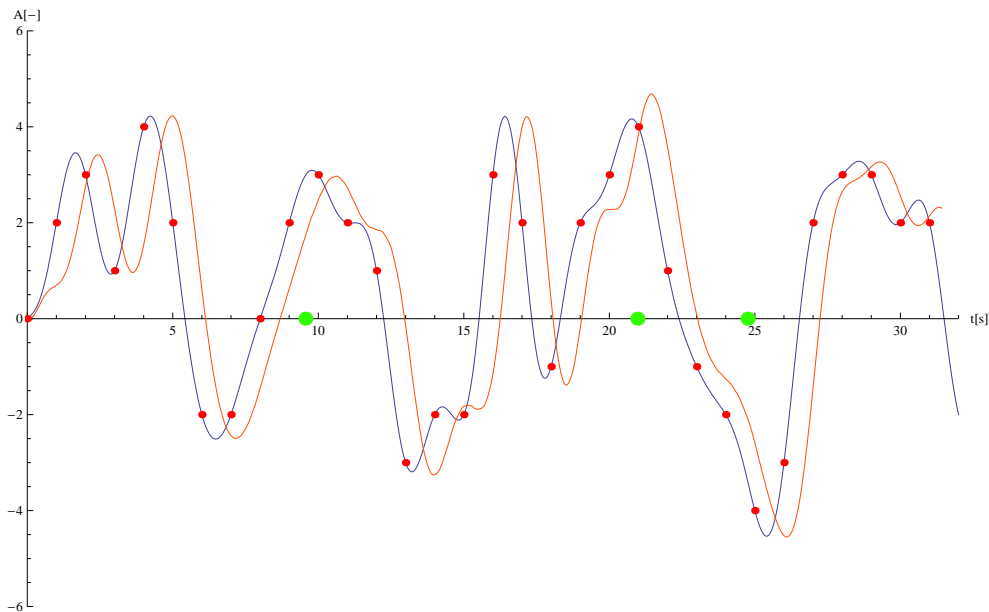
amplitudy, buď z  $A$  na  $-A$  nebo obráceně). V místě napojení se použije spínací čas z druhého (napojovaného signálu). Další možností by bylo např. použít průměr z obou časů. Způsob napojování je naznačen na obr. 4.12.

Signály, které se mají spojit mají modrou a červenou barvu, výsledný signál má béžovou barvu. Amplitudy jsou opět rozdílné jen kvůli přehlednosti obrázku, ve skutečnosti jsou stejné. Je zde vidět, že béžový signál se shoduje s modrým do času  $t=9.7s$  a od tohoto času se shoduje s červeným signálem. Ukázka, jak vypadá napojený PWM signál po filtraci, je na obr. 4.13.

Na tomto obrázku byla použita délka bloku  $l = 10$ , přesah  $p = 3$ , amplituda PWM byla  $A = 6$ , počet spínacích časů  $n = 100$ . Časy napojení jsou zde naznačeny zelenými puntíky. Je vidět, že v okolí časů napojení nedošlo k nijak výrazným deformacím, mírná deformace výstupního signálu je jen okolo napojení v čase  $t=21s$ . Pro lepší napojování je také výhodné každý druhý blok počítat pro trochu jiné  $n$ . Např. každý lichý blok počítat s  $n$ , každý sudý blok počítat s  $n + q$ , kde  $q$  je číslo oproti  $n$  malé (maximálně do 10%  $n$ ). Pro výše uvedený obrázek bylo  $q = 1$ .

### 4.2.3 Implementace v systému *Mathematica*

Výše uvedený algoritmus jsem implementoval do funkce



Obrázek 4.13: Filtrace napojovaného PWM signálu

```
{alfa, ds, casy} = OptimalPWMSignal[s, dT, l, p, n, A, wp, q];,
```

kde  $s$  je posloupnost hodnot reprezentující daný dlouhý signál,  $wp$  je `WorkingPrecision` (přesnost výpočtů z funkce `SolveML2AlfaHankel`), ostatní parametry mají stejný význam jako je výše uvedeno. Funkce vrátí 3 proměnné:

- `alfa` - sada časů, ve kterých dojde ke změně amplitudy signálu,
- `ds` - sada rozdílů časů při napojování,
- `casy` - sada časů, v kterých došlo k napojení.

Nejdůležitější je `alfa`, která udává jak bude vypadat výsledný PWM signál, ostatní 2 výsledky jsou jen pro kontrolu nebo pro ladění. Hodnoty v `ds` by měly být co nejmenší. Zdrojový kód v jazyce *Mathematica* všech funkcí výše uvedených je v příloze na CD.



# Kapitola 5

## Zvukový signál

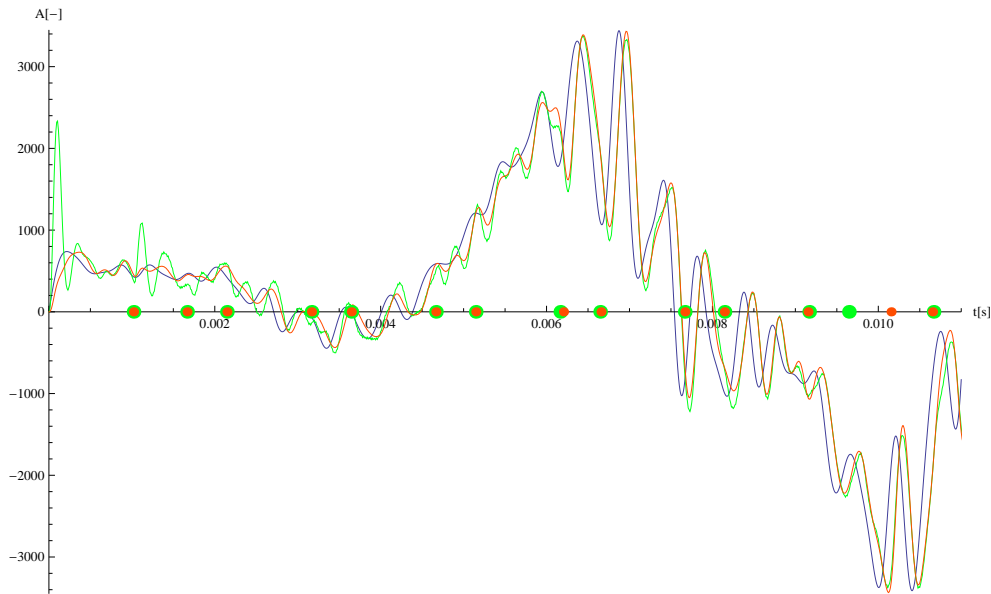
Typický zvukový formát např. na hudebním CD-ROM je 16bit PCM (Pulse Code Modulation) se vzorkovací frekvencí 44100Hz. Za jednu vteřinu se tedy vystřídá 44100 vzorků, z nichž každý může nabývat hodnoty od 0 do  $2^{16} = 64536$ , resp od -32768 do 32767. Dále jsou běžné vzorkovací frekvence 22050Hz, 16000Hz nebo 8000Hz. Stejně tak rozlišení vzorků může být 8bit nebo třeba i 24bit nebo 32bit.

### 5.1 Zvuk

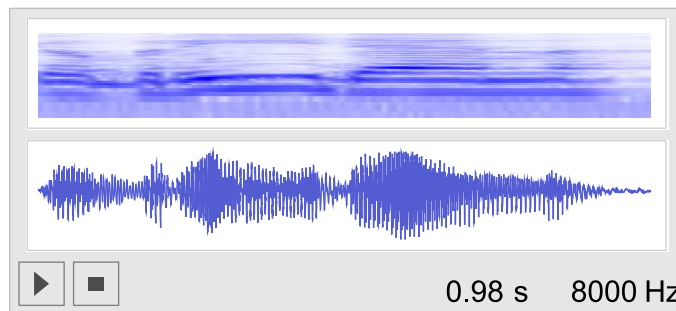
Při výpočtu PWM pomocí metod výše uvedených je nutné počítat s velmi dlouhým trváním výpočtu. Dále také nastává problém, kdy část zvukového signálu dosahuje vysokých hodnot např. okolo  $\pm 30000$  a část zvukového signálu má malou amplitudu např. do 1000. Amplitudu výsledného PWM signálu je nutné volit asi o 25% vyšší než je maximální amplituda v celém signálu. Tím vznikají problémy popsané v části Simulace 4.1.4. Zároveň se zvyšují nároky na kvalitu napojení, protože při velké amplitudě PWM se ve výsledném signálu projeví deformace i při poměrně dobré shodě spínacích časů napojovaných PWM signálů.

Pro účel testování jsem nahrál krátkou zvukovou nahrávku, jejíž vzorkovací frekvence je 8000Hz, formát vzorků je 16bit, záznam je dlouhý 1s. Na tuto nahrávku jsem nahrál slova "Dobrý den". Na následující obrázku je ukázáno, jaký vliv má velká amplituda PWM na část signálu s malou amplitudou.

Zde je krátký výřez ze začátku nahrávky "Dobrý den", ve kterém je maximální amplituda rovna hodnotě 3440. V celé nahrávce je ovšem maximální amplituda 32768. Modrou barvou je původní signál, zelenou barvou je signál získaný z PWM o amplitudě  $A = 1.25 \cdot 32768 = 40960$ , červenou barvou je pro porovnání signál získaný z PWM o am-



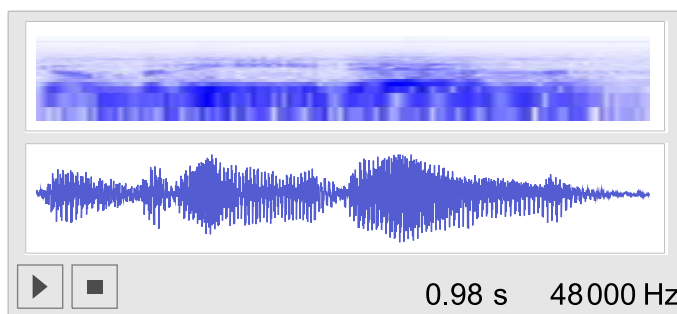
Obrázek 5.1: Zvukový signál s různou amplitudou PWM



Obrázek 5.2: Originál zvukové nahrávky

plitudě  $A = 1.25 \cdot 3440 = 4300$ , který v tomto krátkém úseku lépe vyhovuje. Ostatní parametry jsou délka bloku  $l = 7$ , přesah  $p = 3$ , počet sepnutí na blok  $n = 120$ ,  $q = 6$ . Originální zvuková nahrávka ve formátu WAV (`dobryden_original.wav`) je v příloze na CD, zvuk získaný z filtrace PWM signálu vypočteného s parametry uvedenými v tomto odstavci je také na CD ve formátu WAV (`dobryden_filteredPWM.wav`). Při poslechu obou nahrávek zjistíme, že výsledná nahrávka z PWM signálu se shoduje s originálem, obsahuje jen jakési slabé rušení. To je zřejmě způsobeno jemnými deformacemi v místech napojení nebo také deformací vzniklou na malém signálu při velké amplitudě PWM.

Je zde také možnost využít přímo reproduktory spolu se zesilovačem jako filtr typu dolní propust. Jak zesilovač tak reproduktor má určitou omezenou šířku pásma, která



Obrázek 5.3: Filtrovaný PWM

bývá přizpůsobena možnostem lidského ucha. Většina reprosoustav tedy dokáže reprodukovat zvuky s frekvencí 20Hz - 20000kHz. Právě horní mez tohoto frekvenčního rozsahu je možné využít pro filtraci PWM. Při převodu PWM signálu do zvukového souboru je nutné ho nejdříve navzorkovat. Jak už bylo řečeno v části Filtrace PWM, je nutné volit velmi vysokou vzorkovací frekvenci. Při volbě  $f_{vzPWM} = 1000f_{vz}$ , kde  $f_{vzPWM}$  je frekvence vzorkování PWM signálu a  $f_{vz}$  je frekvence vzorkování původního zvukového signálu, je ještě deformace vzniklá vzorkováním zanedbatelná. U zvukové nahrávky "Dobrý den" se tedy dostaneme na frekvenci vzorkování 8MHz. Problémem je, že takovou frekvenci nedokáže zahrát žádná zvuková karta. Použil jsem tedy  $f_{vzPWM} = 192000Hz$ , což je sice méně obvyklá vzorkovací frekvence, nicméně používá se např. u zvukových DVD a novější zvukové karty zvuk s touto frekvencí dokáží přehrát.

Původní nahrávka "Dobrý den" má vzorkovací frekvenci 8000Hz, čili podle vzorkovací věty je maximální frekvence obsažená v této nahrávce 4000Hz. Všechny vyšší frekvence než 4000Hz je tedy potřeba z PWM odfiltrovat. Jak už bylo řečeno, většina reprosoustav má maximální frekvenci nastavenou na hodnotu okolo 20000Hz. Filtrace je tedy nedostatečná a zvuk je zkreslený. Zvukový soubor WAV, na kterém je přímo PWM signál je také součástí přílohy (dobryden\_pwm.wav).

Pokud bychom použili v původní nahrávce frekvenci vzorkování běžných 44100Hz a bychom volili opět jako 10ti násobek velikosti užitečného spektra, tak by při přehrávání PWM signálu došlo k utlumení vyšších harmonických frekvencí a zvuk by byl téměř stejný jako původní nahrávka.

## 5.2 Doba výpočtu

Je třeba ukázat, jak je tento postup výpočtu PWM náročný na výpočetní výkon počítače. Všechny výpočty jsem prováděl na notebooku značky Acer s procesorem Intel Core2Duo T7300 (2GHz, 800MHz FSB, 4MB L2 cache), 3GB DDR2 RAM. Výpočet spínacích časů PWM z nahrávky "Dobrý den", která je dlouhá 1s, trval přibližně 25 minut. Pokud bychom pro filtraci PWM použili filtr s větším sklonem frekvenční charakteristiky, bylo by možné volit nižší počet sepnutí  $n$ . Tím by se výpočet výrazně zrychlil.



# Kapitola 6

## Závěr

V této práci jsem se zabýval problémem výpočtu spínacích časů PWM signálu tak aby tento signál měl shodné frekvenční spektrum s původním audio signálem a navíc dalších  $n$  harmonických nulových. Toho bylo dosaženo pomocí algoritmu, který je popsán např. v [3]. PWM signál se zde počítá z frekvenčního spektra, proto je zde také rozepsáno, jak je možné vypočítat spektrum lichého signálu, pro který je algoritmus navržen. Je zde také probrán problém filtrace PWM signálu filtrem typu dolní propust. Dále je zde rozebráno jak pracovat se signály, které mají velké množství vzorků a výše uvedený algoritmus takto dlouhé signály nedovede zpracovat. Tento problém byl vyřešen pomocí rozdělení signálu na krátké části, které už zpracovat umíme. Důležitá část je napojování dílčích PWM signálů tak, aby v místech napojení nedocházelo k deformacím ve výsledném signálu po filtraci. Celý postup byl implementován v systému *Mathematica* do funkce, která provede celý výpočet a vrátí časy sepnutí odpovídajícího PWM signálu. Nakonec jsem ověřil funkčnost tohoto postupu na krátké zvukové nahrávce a je zde také naznačena náročnost na výpočetní výkon počítače.

Tato práce pojednává o tom, jak daný problém řešit, ale také dává podněty k dalšímu zkoumání v této oblasti. Výsledný zvukový signál získaný postupem uvedeným v této práci je téměř stejný jako výchozí zvuk. Slyšet je pouze slabé rušení. Výhodou přepočtu klasického signálu na PWM je fakt, že koncový stupeň digitálního PWM zesilovače může být realizován pouze dvojicí výkonových MOS-FET tranzistorů. Odpadá tedy potřeba použití složitých D/A převodníků, které jsou často nejslabším článkem v reprodukci zvuku v digitálním formátu.



# Literatura

- [1] Hlaváč V. - Sedláček M., *Zpracování signálu a obrazu*. Vydavatelství ČVUT, Praha 2007.
- [2] Průcha L., *Řady*. Vydavatelství ČVUT, Praha 2005.
- [3] Kujan, P. - Hromčík, M. - Šebek, M., *Solving the optimal PWM problem for odd symmetry waveforms*. Preprints of the 17th IFAC World Congress [CD-ROM]. Seoul: IFAC, 2008, p. 8672-8677. ISBN 978-3-902661-00-5.
- [4] Kujan P. - Hromčík M. - Šebek M., *Single-Phase Optimal Odd PWM Problem*. In proceedings of the 34th Annual Conference of the IEEE Industrial Electronics Society; p. 371-378. ISBN 978-1-4244-1766-7, 2008.
- [5] Czarkowski D. - Chudnovsky D.V. - Selesnick I.W., *Solving the optimal PWM problem for single-phase inverters*. IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications, vol. 49, April, 2002.
- [6] Munk S.M. - Andersen K.S., *State of the Art Digital Pulse Modulated Amplifier System*. 23rd International AES Conference on Signal Processing in Audio Recording and Reproduction; Helsingør, Denmark; May, 2003.
- [7] Wolfram S., *Wolfram Research, Inc.*, 100 Trade Center Drive, Champaign, IL 61820-7237, USA, <http://www.wolfram.com>, [on-line].



# Příloha A

## CD-ROM

Obsah přiloženého CD:

*Mathematica* - v tomto adresáři jsou uloženy zdrojové kódy použitých funkcí napsaných v systému *Mathematica*,

*Wav* - v tomto adresáři jsou zvukové soubory ve formátu *WAV*,

*Tex* - v tomto adresáři se nacházejí zdrojové kódy k tomuto dokumentu napsané v systému *L<sup>A</sup>T<sub>E</sub>X*.