

Bakalářská práce



České  
vysoké  
učení technické  
v Praze

Fakulta elektrotechnická  
Katedra radioelektroniky

# Měření směrových charakteristik mikrofonů

**Ondřej Šupka**

Komunikace, multimédia, elektronika

Květen 2014

Vedoucí práce: Ing. František Rund, Ph.D.

# Zadání

## Poděkování / Prohlášení

Na tomto místě bych chtěl poděkovat zejména svému vedoucímu práce Ing. Františku Rundovi, Ph.D. za mnoho cenných rad, ochotu a trpělivost při vedení bakalářské práce.

Poděkování patří také RNDr. Petru Olšákovi za zhotovení této TeXové šablony.

V neposlední řadě děkuji všem respondentům, kteří mi poskytli potřebné informace.

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

V Praze dne 23. května 2014

.....

## Abstrakt / Abstract

Práce se zabývá směrovými charakteristikami mikrofonů a zejména jejím měřením. V práci je zpracována rešerše používaných metod a návrh metod vlastních. Metody jsou navrhovány pro měření v improvizovaných podmínkách a jsou schopny eliminovat vlivy prostoru a další rušivé elementy. Navržené metody byly implementovány v prostředí Matlab, prakticky testovány a jejich výsledky byly porovnány s údaji udávanými výrobcí a patřičně komentovány. Testy dopadly poměrně dobře, implementované metody jsou tedy použitelné. Součástí práce je program pro měření směrových charakteristik mikrofonů, návod na měření a komentovaný kód.

**Klíčová slova:** směrová charakteristika, mikrofon, impulsová odezva, sweep, MLS.

The thesis concerns with microphone polar patterns and their measurements in particular. The thesis contains a research of current measuring methods and own measuring conceptions. Methods are designed for measurements in improvised conditions with the ability of eliminating the area influences and other disruptive elements. Concepts were implemented in the Matlab software, field tested and their results were compared to data stated by the manufacturers and commented. Results of the tests were quite satisfying, implemented methods are applicable. A part of the thesis is a microphone polar pattern measurement programme, measurement manual and an annotated code.

**Keywords:** polar pattern, microphone, impulse response, sweep, MLS.

# Obsah /

<b>1 Úvod</b> .....	1
<b>2 Mikrofony</b> .....	2
2.1 Základní principy mikrofonů ....	2
2.2 Směrová charakteristika .....	3
2.3 Mikrofony s měnitelnou směr. charakteristikou .....	4
2.4 Standardní metody měření směr. charakteristik .....	6
<b>3 Signálové operace</b> .....	8
3.1 Fourierova transformace a FFT .....	8
3.2 Korelace .....	8
3.3 Konvoluce .....	9
<b>4 Impulsní odezva a její měření</b> ...	10
4.1 MLS .....	10
4.2 SineSweep .....	12
<b>5 Krokové motory a jejich řízení</b> ..	14
5.1 Krokové motory .....	14
5.2 Arduino Uno .....	15
<b>6 Implementace metody</b> .....	16
6.1 Měřicí přípravek .....	16
6.2 Návrh algoritmu .....	18
6.3 Implementace .....	19
6.3.1 GUI .....	19
6.3.2 Připojení k Arduino .....	21
6.3.3 Měření .....	21
6.3.4 Analýza .....	22
<b>7 Experimentální ověření</b> .....	25
7.1 Měřicí řetězec .....	25
7.2 Naměřené výsledky .....	26
<b>8 Závěr</b> .....	31
Literatura .....	32
<b>A Návod k měření</b> .....	34
<b>B Popis kódu</b> .....	36
B.1 Měření .....	36
B.2 Analýza .....	39
<b>C Obsah příloženého CD</b> .....	46

## Tabulky / Obrázky

<b>2.1.</b> Průzkum metod měření směrových charakteristik.....7	<b>2.1.</b> Principiální schéma tlakových a rychlostních mikrofonů...3
	<b>2.2.</b> Princip skládání směrových charakteristik.....4
	<b>2.3.</b> Konstrukce mikrofonu s kardioidní směrovou charakteristikou .....4
	<b>2.4.</b> Příklad mechanicky a elektricky realizované měnitelné charakteristiky .....5
	<b>2.5.</b> Polarflex .....5
	<b>4.1.</b> Amplitudové a fázové spektrum MLS ..... 11
	<b>4.2.</b> Příklad posuvného registru pro generování MLS..... 12
	<b>4.3.</b> Zkreslení u SineSweptu ..... 13
	<b>4.4.</b> Pre-ringing artefakt a jeho potlačení..... 13
	<b>5.1.</b> Full Step režim A3967 ..... 14
	<b>6.1.</b> Kompenzace excentricity membrány ..... 17
	<b>6.2.</b> Měřicí přípravek a detail motoru ..... 17
	<b>6.3.</b> Elektronická část přípravku ... 18
	<b>6.4.</b> GUI ..... 20
	<b>6.5.</b> Vývojové diagramy ..... 24
	<b>7.1.</b> Zapojení měřicího řetězce..... 25
	<b>7.2.</b> Uspořádání mikrofonreprosoustava..... 25
	<b>7.3.</b> Směrové charakteristiky mikrofonu Audix CX111..... 27
	<b>7.4.</b> Směrové charakteristiky mikrofonu RØDE NT2-A ..... 28
	<b>7.5.</b> Směrové charakteristiky mikrofonu Beyerdynamic TG I50d ..... 29
	<b>7.6.</b> Směrové charakteristiky mikrofonu AKG CK80..... 30

# Kapitola 1

## Úvod

Popsat vlastnosti mikrofonu je vždy těžká úloha a jen některé z nich můžeme vyjádřit tzv. „na papír“. Jednou ze specifikací, kterou takto vyjádřit lze, je právě směrová charakteristika. Ta nám ukazuje schopnost mikrofonu snímat z různých směrů na různých frekvencích. Od této charakteristiky se odvíjí využití mikrofonu – jinou charakteristiku budeme používat pro snímání zpěvu v prostředí s výbornou akustikou, kde chceme snímat i prostor, a jinou zas pro oblíbené „close miking“ bicích nástrojů, kde se snažíme o co nejmenší přeslechy sousedních komponentů.

Většina mikrofonů obsahuje datasheet, kde bývá mj. uvedena i směrová charakteristika, ale pokud k dispozici datasheet nemáme, nebo chceme detailnější směrovou charakteristiku, popřípadě i na jiných frekvencích než ji výrobce uvádí, nezbyvá než mikrofon přeměřit. Měření vyžaduje velmi specifické podmínky a vybavení (mj. i bezodraznou komoru) a nezbyvá než využít specializovanou laboratoř. Metody, kterými se zabývám v této práci, vycházejí z mé předchozí práce v rámci předmětů Individuální projekt I.,II., kde jsem metod testoval více, došel jsem však k závěru, že jen některé z testovaných metod jsou vhodné pro použití v improvizovaných podmínkách (tj. bez bezodrazné komory).

V kapitolách 2-5 této práce bude čtenář seznámen s nezbytnými teoretickými znalostmi důležitými pro pochopení některých částí této práce, v kapitole 6 bude následovat analýza měřicích metod, návrh algoritmu a implementace zvolených měřicích metod. V kapitole 7 budou prezentovány, porovnávány a hodnoceny naměřené výsledky pomocí implementovaných metod. Na konci práce je celkové shrnutí a zhodnocení výsledků, kterých jsem v této práci dosáhl. V přílohách čtenář nalezne ještě návod k měření (příloha A), detailnější popis kódu (B) a CD s výsledky práce (C).

# Kapitola 2

## Mikrofony

Mikrofon je elektroakustický měnič, který je definován Škvorem v [12] jako zařízení, kde dochází k přeměně akustického tlaku a objemové rychlosti na elektrické napětí a proud. Elektroakustické měniče, které lze řadit do skupin – měniče s magnetickým polem a s elektrickým polem. Do skupiny měničů s magnetickým polem řadíme měniče elektrodynamické, elektromagnetické a magnetostrikční. Mezi měniče s elektrickým polem jsou řazeny měniče elektrostatické a piezoelektrické. Všechny uvedené měniče jsou reciproké – tzn. lze je využít jak na konverzi akustického signálu na elektrický, tak naopak. Principy jednotlivých měničů v mé práci rozebírat nebudu, jelikož tyto principy nesouvisí se směrovou charakteristikou mikrofonu.

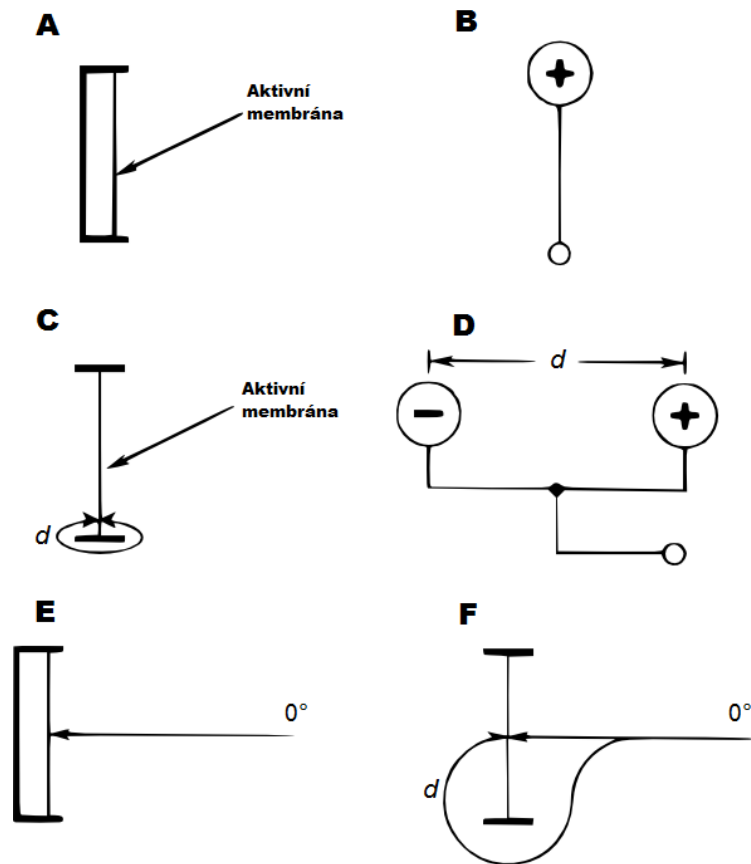
### 2.1 Základní principy mikrofonů

Existují dva hlavní principy snímání zvuku a dle těchto principů se mikrofony dělí na gradientní mikrofony nultého řádu (**tlakové**) a prvního řádu (**rychlostní**).

Jak vysvětluje Eargle v [8], ideální tlakový mikrofon reaguje pouze na akustický tlak bez ohledu na směr, a jeho směrová charakteristika (viz obrázek 2.2) je tedy kulová. Reálné mikrofony však malou směrovou závislost vykazují, a to především na vysokých frekvencích (u studiových mikrofonů typicky nad 8 kHz) na hlavní ose mikrofonu. Konstrukce je provedena tak, že membrána je upevněna na okraji téměř vzduchotěsné, uzavřené dutiny, takže tlak na zadní stranu membrány je konstantní. V dutině bývá malý otvor – tzv. kapilára pro vyrovnávání atmosferického tlaku. U měřicích mikrofonů se kapilára před měřením ucpává záslepkou pro vyšší přesnost měření.

Rychlostní mikrofony zachycují akustický tlak ve dvou velmi blízko umístěných bodech, které odpovídají přední a zadní straně membrány. Membrána je vychylována na základě rozdílu těchto dvou tlaků. Zvukové vlny dopadající z  $0^\circ$  a  $180^\circ$  způsobí tedy největší rozdíl tlaků. Naopak vlny přicházející z  $90^\circ$  nezpůsobí rozdíl žádný. Principy rychlostních mikrofonů v porovnání s tlakovými můžeme vidět na obrázku 2.1. Z principu rychlostního mikrofonu vyplývá, že bude mít osmičkovou směrovou charakteristiku. Velkou nevýhodou tohoto principu mikrofonu je tzv. „proximity efekt“, který způsobuje nárůst nízkých frekvencí se zmenšující se vzdáleností zdroje zvuku k mikrofonu. Čím je vzdálenost menší tím se uplatní více rozdíl drah od zdroje k oběma stranám membrány a vzhledem k fázovému rozdílu mezi oběma signály pak dochází ke zmíněnému jevu.





**Obrázek 2.1.** Zjednodušený pohled na tlakový mikrofon (A); náhradní schéma tlakového mikrofonu (B); pohled na rychlostní mikrofon (C); náhradní schéma rychlostního mikrofonu (D); zvuk z  $0^\circ$  dopadající na tlakový mikrofon (E); zvuk z  $0^\circ$  dopadající na obě strany membrány rychlostního mikrofonu zpožděn o  $d$  (F). Převzato z [8].

## 2.2 Směrová charakteristika

Jak jsem zmínil v úvodu, směrová charakteristika je jednou z nejzákladnějších charakteristik, kterou může být mikrofon popsán. Tato charakteristika o mikrofonu vypovídá, jak je schopen snímat zvuk z daného směru. Schopnost mikrofonu snímat zvuk přicházející z různých úhlů je samozřejmě frekvenčně závislá, proto je v polárním diagramu (do kterého se charakteristika většinou zaznamenává) vykresleno několik křivek pro vybrané frekvence.

O směrových charakteristikách jsem se již několikrát zmínil v předchozích větách (2.1), takže víme, že existují dvě základní charakteristiky – kulová a osmičková, které vycházejí ze základních principů mikrofonů. Pokud chceme dosáhnout jiných směrových charakteristik než výše zmíněných, musíme tyto dvě charakteristiky kombinovat (viz obrázek 2.2). Kombinací dostáváme kardioidní charakteristiku a její modifikace (hyperkardioida, superkardioida a jiné). Při kombinacích vyšších řádů získáváme pak další charakteristiky, které se využívají například u vysoce směrových mikrofonů a dalších mikrofonů se specifitějšími účely využití. Na obrázku 2.3 můžeme vidět praktickou realizaci mikrofonu s kardioidní směrovou charakteristikou, která využívá obou hlavních principů.









# Kapitola 3

## Signálové operace

V následujících odstavcích uvedu základní fakta o hlavních operacích, o kterých se v mé práci zmiňuji. Jelikož zpracovávám signály pouze v diskretním čase, následující vzorce tomu budou přizpůsobeny.

### 3.1 Fourierova transformace a FFT

Fourierova transformace (FT), jak je popsáno v [23], je matematická transformace, která umožňuje transformaci signálů z časové oblasti do frekvenční, zpětná FT (IFT) pak naopak z frekvenční oblasti zpět do časové. FT vychází z Fourierových řad (FS), které umožňují rozklad periodického signálu na siny a kosiny. FT to umožňuje i u infinitních neperiodických signálů. Základní vlny se většinou zapisují pomocí Eulerova čísla, jako  $e^{2\pi j\theta}$ . Tento zápis lze rozložit jako  $e^{2\pi j\theta} = \cos 2\pi\theta + j \sin 2\pi\theta$ . Zápisem do komplexních exponenciál je zřejmé, že jednotlivé koeficienty jsou komplexně sdružené, nicméně tento komplexní zápis obsahuje jak amplitudu tak i fázi vlny.

Jelikož signály zpracováváné v této práci jsou v diskretním čase, je třeba představit především diskretní Fourierovy řady (DFS) a diskretní Fourierovu transformaci (DFT). Platí pro ně vše, co jsem zmínil v předchozím odstavci, jen pracují se signály v diskretním čase. DFS je definována jako:

$$d_n = \frac{1}{N_0} \sum_{k=0}^{N_0-1} s[k] e^{-2\pi j (\frac{n}{N_0})k}, \quad (1)$$

a DFT takto:

$$D[n] = \sum_{k=0}^{N-1} s[k] e^{-2\pi j (\frac{n}{N})k}. \quad (2)$$

DFT je však často implementována jako FFT (Fast Fourier Transform), což je algoritmus efektivnějšího výpočtu DFT. Přímé vyhodnocení sum DFT by zabralo  $O(N^2)$  matematických operací, díky FFT můžeme počet těchto operací minimalizovat na  $O(N \log(N))$ . Standardně jsou FFT algoritmy založeny na faktorizaci  $N$ , ale existují i stejně efektivní algoritmy pro všechna  $N$ . Nejpoužívanějším FFT algoritmem je Cooley-Tukey algoritmus, ten funguje tak, že rekurzivně dělí DFT jakéhokoliv složeného čísla o velikosti  $N = N_1 N_2$  do menších DFT o velikostech  $N_1$  a  $N_2$ .

### 3.2 Korelace

Korelace znamená vzájemný vztah mezi dvěma veličinami, v našem případě mezi dvěma signály. Pro reálné signály v diskretním čase je definována následovně:

$$\text{corr}(x[n], h[n]) = \sum_{k=-\infty}^{\infty} h[k] x[n-k]. \quad (3)$$



# Kapitola 4

## Impulsní odezva a její měření

Impulsní odezva je odezva lineární, časově invariantní soustavy (LTI – linear, time invariant) na Diracův impuls (nekonečně úzký a vysoký impuls, nabývající nenulové hodnoty jen v jediném bodě) nebo v případě diskrétní časové soustavy jednotkový impuls. Jak se dočteme v [9], impulsová odezva pak tuto soustavu plně charakterizuje – pro každý vstup může být se znalostí impulsní odezvy vypočten konvolucí výstup (viz 3.3). Často je jednodušší soustavu analyzovat pomocí přenosové funkce, která bude odpovídat Laplaceově transformaci impulsní odezvy. Pomocí IFT z impulsově odezvy můžeme vypočítat křivku rozložení energie v čase nebo získat frekvenční charakteristiku použitím FT.

Jak je uvedeno v [20], za běžnou metodu měření impulsní odezvy (jako akustické soustavy) je považováno použití známého vstupu a měření odpovídajícího výstupu. Volbou je tedy zdrojový signál a dekonvoluční technika (dekonvoluce = zpětná konvoluce). Obecně pro signály pro měření impulsní odezvy platí několik parametrů:

- měřicí signál by měl být perfektně reprodukovatelný;
- naměřený signál a dekonvoluční technika by měly mít vhodné vlastnosti pro co největší SNR (odstup signál-šum, Signal to Noise Ratio);
- naměřený signál a dekonvoluční technika by měly mít vhodné vlastnosti na eliminaci nelineárních artefaktů, které se mohou projevit v dekonvolvaném signálu.

Od roku 1979, kdy byla poprvé navržena technika MLS (Maximum-Length Sequence) [19], je poměrně hojně používána. Podobnou technikou je IRS (Inverse Repeated Sequence), další využívané jsou tzv. time-stretched pulsy a jim blízký přeladovaný sinus (SineSweep). V této práci se zabývám jen dvěma z nich a to MLS a SineSweep. Jednak jsou to asi nejpoužívanější techniky a navíc zastupují obě dvě zmíněné „skupiny“.

### 4.1 MLS

MLS je periodická dvojúrovňová (0 a 1 nebo chceme-li -1 a 1) pseudonáhodně generovaná posloupnost délky  $N = 2^m - 1$ , kde  $m$  je celé číslo. Tato sekvence má několik výhodných vlastností. Hlavními vlastnostmi jsou rovné frekvenční spektrum, entropické fázové spektrum a dobrá odolnost vůči šumu. Odolnost vůči šumu je dána činitelem tvaru signálu (crest factor), který je roven 1 – nejnižší možný. V porovnání například s přeladovaným sinem, který se pro měření také hojně používá, který má činitel tvaru  $\sqrt{2}$ , což je dobré, ale ne tak dobré jako u MLS.

MLS má výbornou autokorelační vlastnost, kterou zdůvodňuje Bristow–Johnson v [6] nebo Kemp v [11]. Pro MLS bude autokorelace vycházet z (3) a bude vypadat takto:







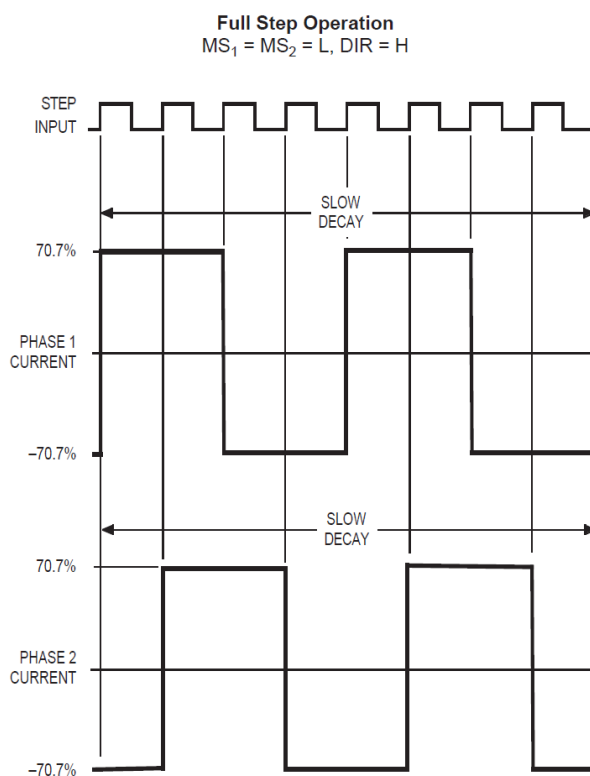


# Kapitola 5

## Krokové motory a jejich řízení

### 5.1 Krokové motory

K automatizaci měření (konkrétně k otáčení mikrofonu) jsem zvolil krokový motor. Krokový motor jsem vybral proto, že nabízí dostatečnou přesnost a krouticí moment. V úvahu přišel ještě DC motor a servo motor. DC motor je sice rozhodně levnější, ale k nastavení přesného azimutu by bylo třeba implementace nějakého snímání polohy, např. Hallovým senzorem, nicméně toto řešení mi přijde zbytečně složité. Druhou alternativou byl servo motor, u kterého je zase častým problémem krouticí moment. Motor, který používám, disponuje základním rozlišením (pokud není používáno řízení s mikrokroky)  $1,8^\circ$ , což je pro mé účely naprosto dostačující.



**Obrázek 5.1.** Režim Full Step čipu A3967. Převzato z [2].

Krokový motor, jak vysvětluje Rydlo v [16], je impulsně napájený motor, jehož funkční pohyb je nespojitý a děje se po jednotlivých krocích. V našem případě se jedná o motor s pasivním rotorem. Rotor je tvořen pouze svazkem plechů nalisovaných na hřídel a stator je tvořen určitým počtem pólů s navinutými cívkami (v mém případě 8 pólů). Přivedením proudu na cívky vzniká magnetické pole vychylující rotor. Pokud spínáme cívky ve vhodném intervalu a pořadí, docílíme otáčení rotoru. Řízení z PC probíhá přes Arduino Uno (více v 5.2) a open-source hardware EasyDriver [17]. EasyDriver



# Kapitola 6

## Implementace metody

### 6.1 Měřicí přípravek

Je zřejmé, že ke změření směrové charakteristiky mikrofonu je třeba mikrofon změřit z několika směrů. Nabízejí se dvě možnosti jak toto zrealizovat:

- statický mikrofon a pohyblivý reproduktor;
- pohyblivý mikrofon a statický reproduktor.

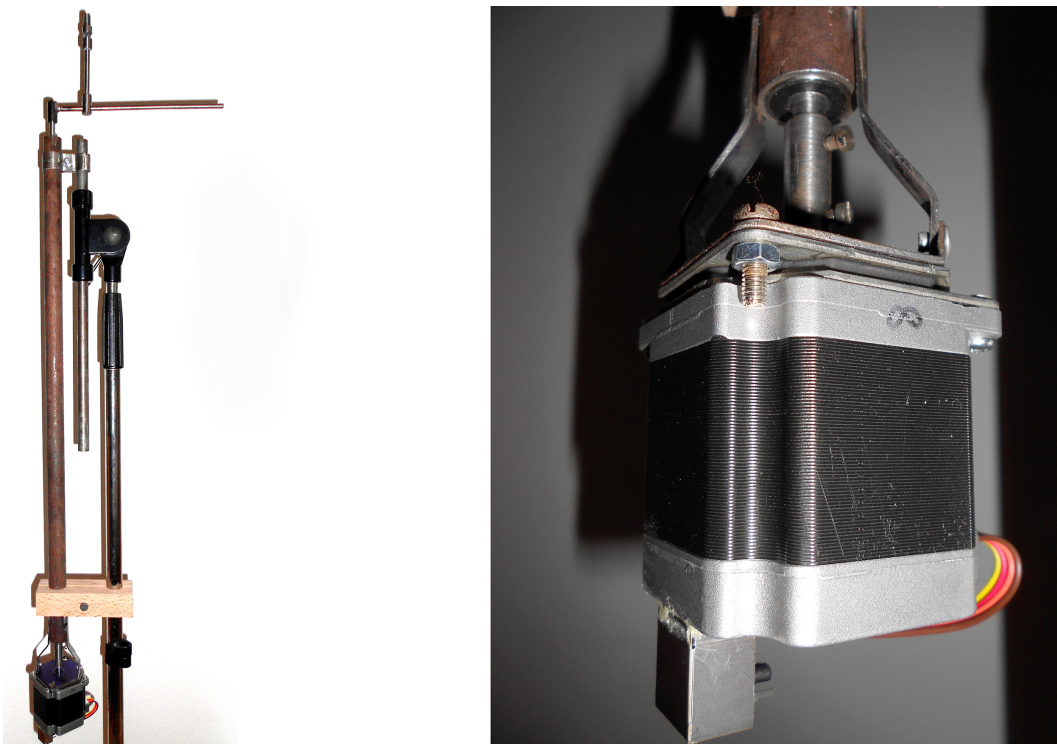
Jelikož z počátku byla vize taková, že se bude jednat jen o orientační, snadno přenosnou metodu, ke které by měl stačit nějaký mobilní zdroj zvuku (například mobilní telefon), jevila se tato možnost jako jednodušší. Mikrofon by byl staticky na stojanu, na stojanu by byl připevněn nějaký prvek definující vzdálenost zdroje zvuku, například obyčejný provázek, na který by se na druhém konci připevnil zdroj zvuku a s ním by bylo ručně otáčeno podle stupnice na stojanu. Nicméně zdroj zvuku (alespoň v případě reproduktorů nebo reprosoustav) má obvykle větší hmotnost než mikrofon a často nemívá standardizované řešení upnutí (narozdíl od mikrofonů, které, až na výjimky, lze upevnit do dodávaných držáků se závity 3/8" nebo 5/8" se stoupáním 27 závitů na palec). V případě hmotnosti je problémem konstrukce, která by dokázala být stabilní při otáčení zdrojem zvuku v požadované vzdálenosti od mikrofonu (jelikož musíme měřit mimo blízká pole mikrofonu a reproduktoru, mluvíme tu o desítkách centimetrů) a i motor by musel být mnohem silnější. Tyto nedostatky vedou k variantě druhé, ve které se bude, narozdíl od otáčení zdroje v několika desítkách centimetrů od zdroje, otáčet mikrofonem téměř v jednom bodě.

Zde narážíme na problém excentričnosti membrány mikrofonu vůči bodu uchycení. Pokud použijeme analogii k fotoaparátům, můžeme bodu, kde by měla být osa otáčení, říkat „nodal point“. Pokud chceme měřit přesně musíme, tento nedostatek eliminovat. Jednou možností je umístit zdroj do vzdálenosti, kde už tato výchylka, která se bude obvykle pohybovat v rámci 10 cm, bude zanedbatelná. Toto řešení by však vyžadovalo přísnější podmínky – nižší okolní hluk, vyšší hlasitost reprodukováného signálu a vyšší zisk mikrofonu, který s sebou samozřejmě přináší i více šumu. Druhou možností je umístit zmíněný nodal point do osy otáčení. Jelikož je každý mikrofon stavěný jinak, musí to být řešení univerzální. K tomuto účelu jsem zhotovil přípravek (je součástí celého měřicího systému), díky kterému můžeme tento nedostatek kompenzovat (viz obrázek 6.1). Na vršku přípravku je klasický mikrofonní 3/8" závit (redukcí jde snadno zredukovat na 5/8"), pod ním je matka sloužící k aretaci přesné polohy natočení příslušného držáku nebo mikrofonu. Tyč, která je pod touto matkou, je poměrně dlouhá (cca 10 cm). V původní verzi tohoto přípravku bylo toto uchycení krátké, ale ukázalo se, že pak není pro některé mikrofony použitelný, obvykle kvůli kabelu s rozměrným XLR konektorem, proto je tato tyč takto dlouhá. Vpravo na obrázku 6.1 můžeme vidět jeden z problematičtějších mikrofonů, konektor se pak nachází přímo v ose otáčení, kde na něj

(i přes delší tyč) už není místo, proto je vhodnější používat v takovém případě lomené konektory, se kterými tento problém nenastává.



**Obrázek 6.1.** Část měřicího přípravku, sloužící na kompenzaci excentricity membrány vůči uchycení mikrofonu.



**Obrázek 6.2.** Měřicí přípravek namontovaný na standardní mikrofonní stojan. Vpravo je detail krokového motoru se silenblokem a přilepeným konektorem RJ45.

Celý měřicí přípravek můžeme vidět na obrázku 6.2. Jeho konstrukci jsem navrhoval tak, aby se dal připevnit na většinu mikrofonních stojanů. Původně jsem se rozhodoval, zda zvolím tuto konstrukci, nebo zda udělám spíše nějaký otočný stolek. Výsledky obou možností by měly být shodné, nicméně stolek by byl asi stabilnější, ale také náročnější















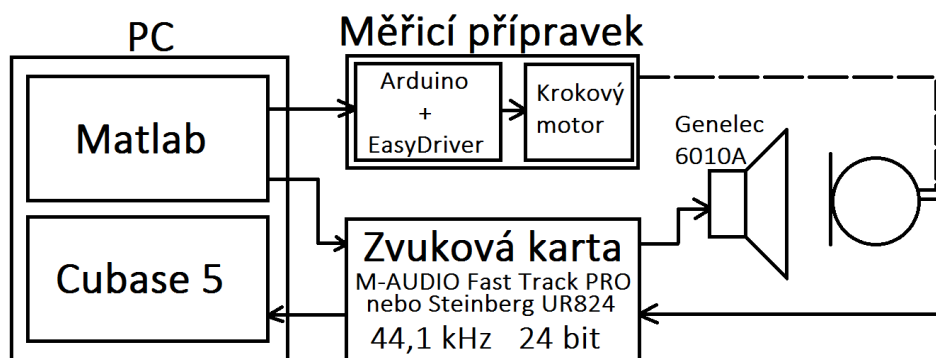


# Kapitola 7

## Experimentální ověření

### 7.1 Měřicí řetězec

Uspořádání měřicího řetězce můžeme vidět na obrázku 7.1. Hlavní část měření se odehrávala v SW Matlab R2013a, záznam zvuku v Cubase 5. Byly použity zvukové karty M-AUDIO Fast Track PRO (pro mikrofony RØDE NT2-A, Beyerdynamics TG I50d a AKG CK 80) a Steinberg UR824 (pro Audix CX111). Volba zvukové karty souvisí jen čistě s místem použití, mikrofon Audix CX111 jsem měřil v běžném pokoji, ostatní mikrofony v částečně akusticky přizpůsobeném prostředí (studio). Obě zvukové karty pracovaly na vzorkovací frekvenci 44,1 kHz a bitové hloubce 24 bitů. Pro reprodukcí signálu byl použit aktivní studiový monitor Genelec 6010A, jehož frekvenční charakteristika byla změřena pomocí SW EASERA (používá se pro korekci). Uspořádání mikrofon – reproduktor (resp. reprosoustava) můžeme vidět na obrázku 7.2. Vzdálenost mikrofonu a reproduktoru byla vždy 20–30 cm, přičemž mikrofon byl vždy minimálně 1,5 m od nejbližší odrazové plochy (to byl zpravidla strop nebo podlaha).



Obrázek 7.1. Zapojení měřicího řetězce.

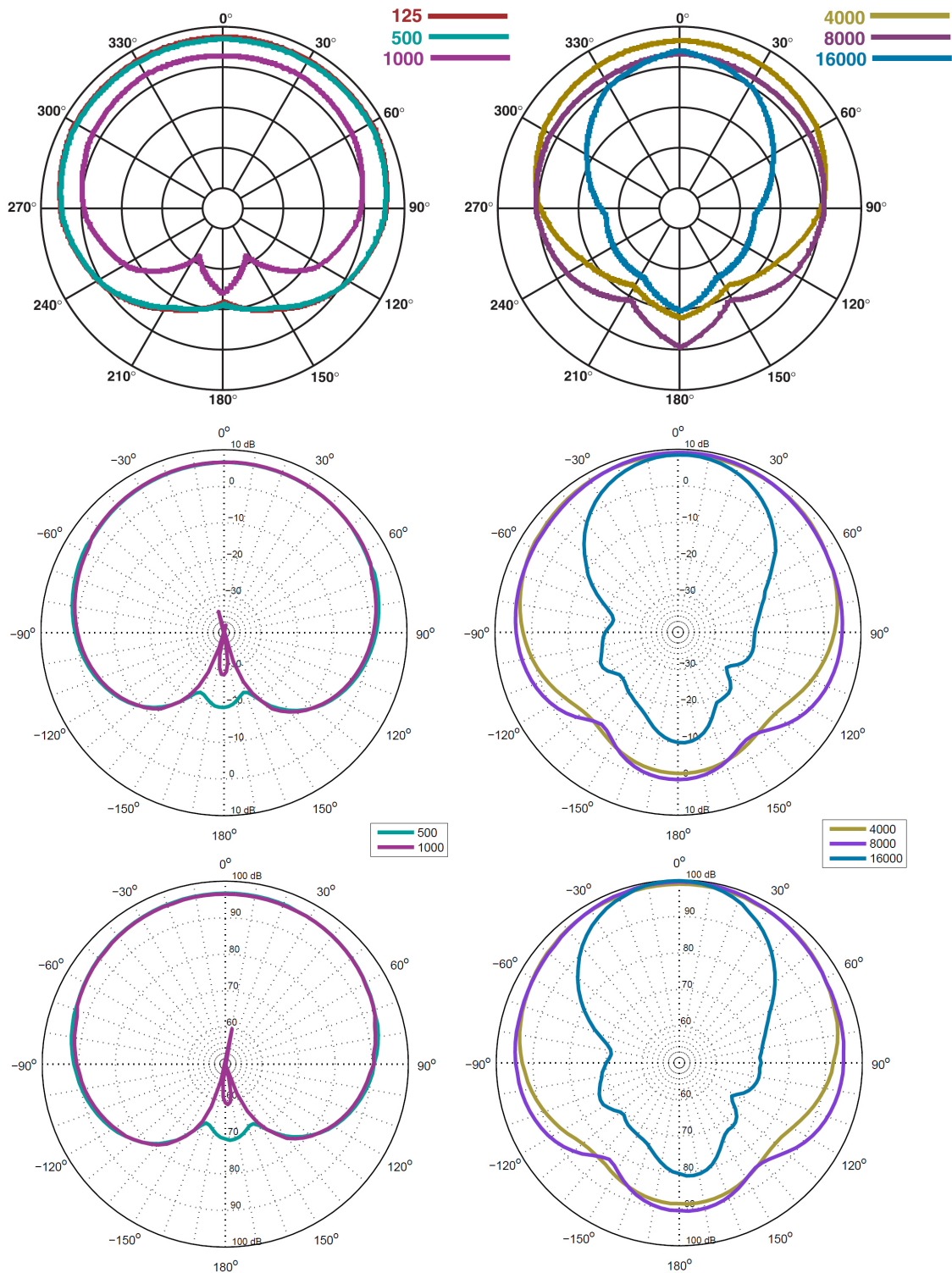


Obrázek 7.2. Uspořádání mikrofon (Audix CX111)–reprosoustava (Genelec 6010A).

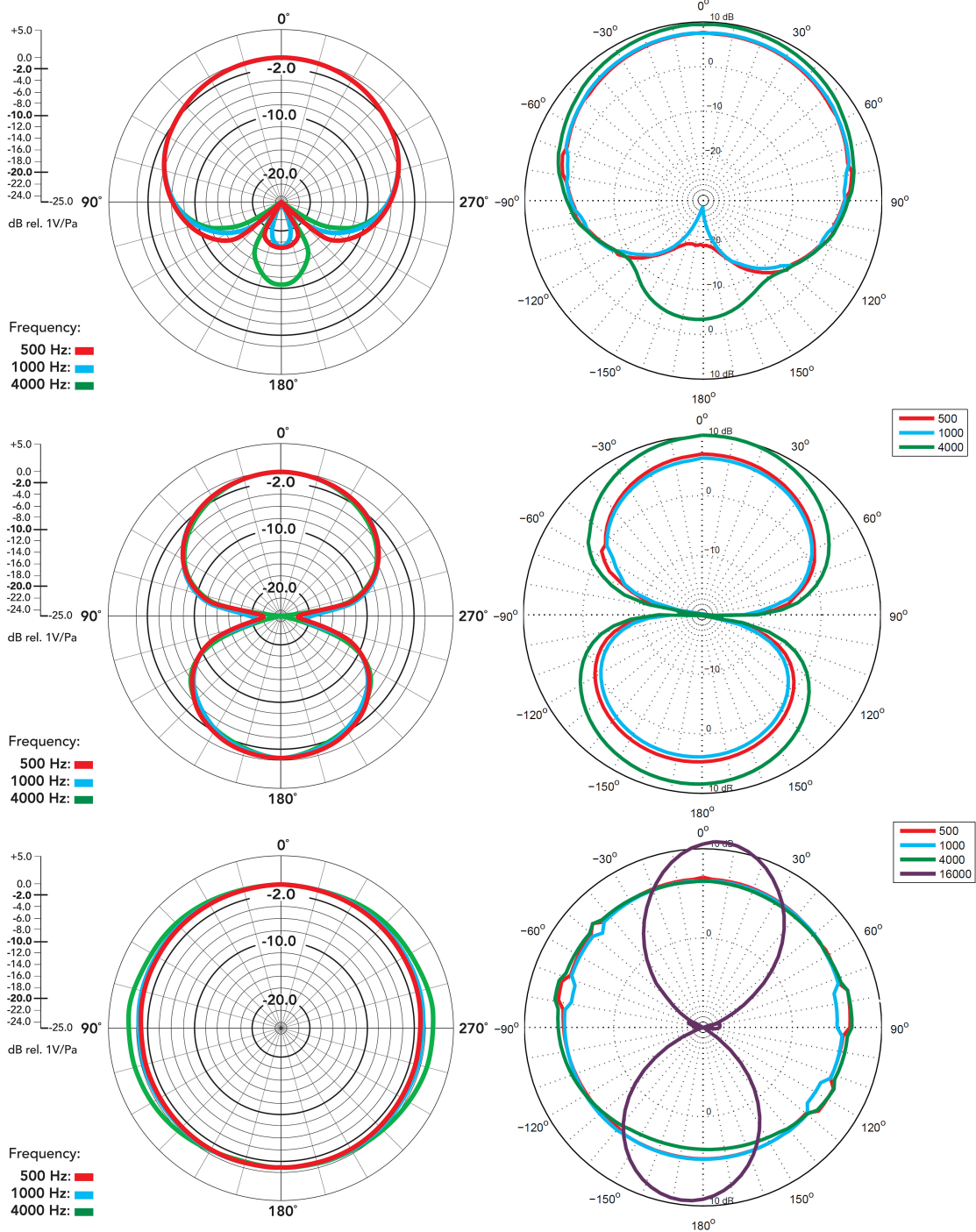




ale jelikož na ostatních diagramech nejsou odchylky tak výrazné, bude asi tato informace relevantní.



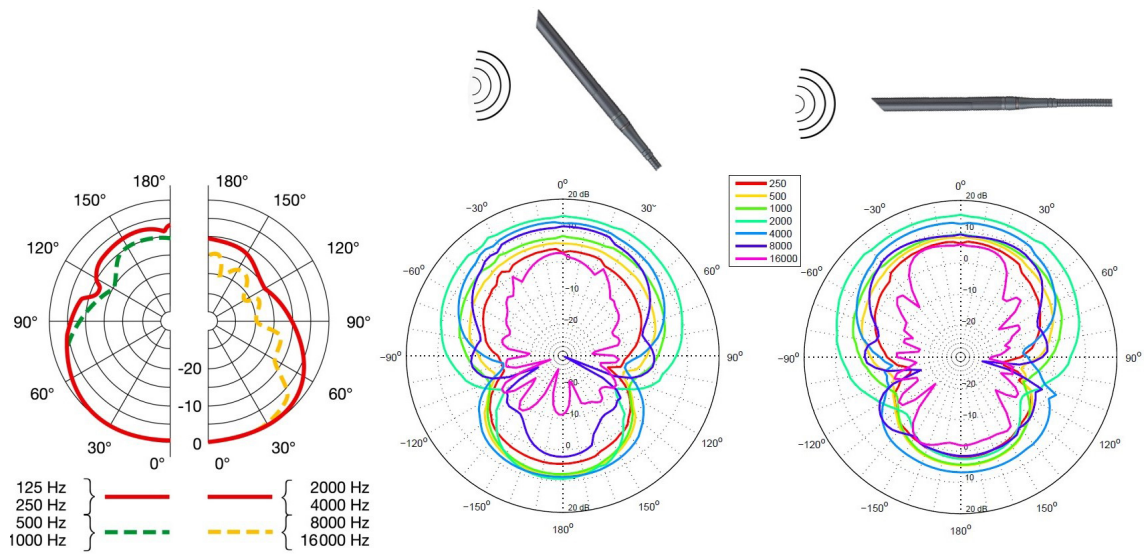
**Obrázek 7.3.** Směrové charakteristiky kondenzátorového mikrofону Audix CX111. Nahoře charakteristiky udávané výrobcem (převzato z [4]), uprostřed charakteristika měřená pomocí sweepu a v dole pomocí MLS.



**Obrázek 7.4.** Směrové charakteristiky kondenzátorového mikrofonu RØDE NT2-A s měnitelnou směrovou charakteristikou. Vlevo jsou charakteristiky udávané výrobcem (převzato z [14], vpravo charakteristiky naměřené použitím sweepu. Nahoře je mikrofon měřen s kardioidní s charakteristikou, níže s osmičkovou a dole s kulovou.

Poslední volitelnou charakteristikou je charakteristika kulová. Na frekvencích vykreslených výrobcem se naměřené výsledky shodují (až na občasné odchylky, které by možná opět upřesnilo opětovné měření). Mimo tyto tři frekvence jsem však vykreslil ještě křivku 16 kHz, kterou již výrobce ve svém diagramu nemá. Jedním z možných důvodů, proč datasheet výrobce tuto křivku neobsahuje je to, že křivka neodpovídá





**Obrázek 7.6.** Směrové charakteristiky modulárního kondenzátorového mikrofonu AKG CK80. Vlevo charakteristika udávaná výrobcem (převzato z [1]), uprostřed a vpravo charakteristiky měřené sweepem s daným uspořádáním zdroje a mikrofonu (viz výše).

# Kapitola 8

## Závěr

Vytvořená práce seznamuje se směrovými charakteristikami a metodami jejich měření. V rámci zkoumaných metod se zabývá měřením impulsní odezvy pomocí metod sweep a MLS. Metody na měření směrových charakteristik byly implementovány v prostředí Matlab, prakticky ověřeny a výsledky byly srovnány s daty udávanými výrobcí mikrofonů. S daty výrobci nemusí být srovnání objektivní, jelikož nevíme přesně jak výrobce mikrofon měřil a měřená data zpracoval, nicméně hlavní tvar křivek zpravidla odpovídá a měření se liší jen v detailech. Metody se tedy jeví jako použitelné. Implementované metody jsou výjimečné svou možností použití v běžných prostorech. Byl vytvořen program do prostředí Matlab, který umožňuje (s příslušným vybavením) měření směrových charakteristik mikrofonů v běžných prostorách. Zadání práce bylo splněno.

Naměřené výsledky jsou porovnány a zhodnoceny v kapitole 7, vyzdvihl bych jen to, že i přes měření ověřené dvěma metodami se v mnoha případech výsledky s údaji výrobců neshodují, což je dané pravděpodobně tím, že výrobci nejsou vázáni normou a tak si charakteristiky často dokreslují k ideálním předpokladům nebo neuvádějí skutečnosti, které předpokladům neodpovídají.

Osobně pro mě práce byla velkým přínosem, ať už v osvojení starých a získání nových znalostí v elektroakustice, v osvojení si prostředí Matlab a jazyku Wiring, tak v technickém vyjadřování. Ukázalo se, že lepší volbou k implementaci by pravděpodobně byl Simulink nebo Labview, což nabízí myšlenku v těchto prostředích metody implementovat a dále rozvinout. Momentálně mě napadá několik vizí jak v práci pokračovat. Hezkou myšlenkou je vytvořit komplexnější měřicí systém, který by při implementaci například v Labview nabízel, díky možné práci v reálném čase, mnohem více možností (detailnější a přesnější měření směrových charakteristik, frekvenčních charakteristik, měření THD, atd.).

## Literatura

- [1] AKG Acoustics GmbH: Datasheet CK80. [online] 2.5.2014.  
<http://www.akg.com/CK80-1280.html?pid=1058>
- [2] Allegro MicroSystems, LLC: Datasheet A3967. [online] 9.4.2014.  
<http://www.allegromicro.com/~media/Files/Datasheets/A3967-Datasheet.ashx>
- [3] Audio Engineering Society, Inc.: *AES standard on acoustics - Sound source modeling - Loudspeaker polar radiation measurements*. 2008, aES56-2008.
- [4] Audix Corporation: Datasheet CX111. [online] 2.5.2014.  
<http://www.showtehnika.lv/pdf/Audix%20CX111%20specifications.pdf>
- [5] Beyerdynamic GmbH & Co. KG: Datasheet TG I50d. [online] 2.5.2014.  
[http://europe.beyerdynamic.com/shop/media//datenblaetter/DAT\\_TGI50d\\_EN\\_A2.pdf](http://europe.beyerdynamic.com/shop/media//datenblaetter/DAT_TGI50d_EN_A2.pdf)
- [6] Bristow-Johnson, R.: A little MLS tutorial. [online] 12.3.2014.  
<http://www.dspguru.com/dsp/tutorials/a-little-mls-tutorial>
- [7] Capetanopoulos, C. D.: *Measurement of the Directivity Characteristics of Loudspeakers and Microphones in a Reverberant Enclosure*. Dep. Elec. Eng., Manhattan Coll., Bronx, N. Y. 10471: IEEE, 1972, ISBN 02405-1961-2.
- [8] Eargle, J.: *The Microphone Book*. Linacre House, Jordan Hill, Oxford OX2 8DP: Focal Press, druhé vydání, 2005, ISBN 02405-1961-2.
- [9] Černocký, J.: Konvoluce, spojité harmonické signály. [online] 12.3.2014.  
[http://www.fit.vutbr.cz/study/courses/ISS/public/oldlabs/konvol\\_harmon.pdf](http://www.fit.vutbr.cz/study/courses/ISS/public/oldlabs/konvol_harmon.pdf)
- [10] Farina, A.: Advancements in Impulse Response Measurements by Sine Sweeps. In *Audio Engineering Society Convention 122*, May 2007.  
<http://www.aes.org/e-lib/browse.cfm?elib=14106>
- [11] Kemp, J.: Auto-correlation property of MLS. [online] 12.3.2014.  
<http://www.kempacoustics.com/thesis/node84.html>
- [12] Škvor, Z.: *Elektroakustika a akustika*. Thákurova 1, 160 41, Praha 6: Česká technika - nakladatelství ČVUT, první vydání, 2012, ISBN 978-80-01-025034-7.
- [13] Listen, Inc.: SoundCheck. [online] 8.3.2014.  
<http://www.listeninc.com/>
- [14] RØDE Microphones: Datasheet NT2-A. [online] 2.5.2014.  
[http://wpc.660d.edgecastcdn.net/80660D/downloads/nt2-a\\_datasheet.pdf](http://wpc.660d.edgecastcdn.net/80660D/downloads/nt2-a_datasheet.pdf)
- [15] Renkus-Heinz: Easera. [online] 8.3.2014.  
<http://easera.afmg.eu/>
- [16] Rydlo, P.: Krokové motory a jejich řízení. [online] 9.4.2014.  
<http://www.mti.tul.cz/files/ats/krok2.pdf>
- [17] Schmalz, B.: EasyDriver. [online] 9.4.2014.  
<http://www.schmalzhaus.com/EasyDriver/>



- [18] Schoeps, GmbH: Polarflex. [online] 7.3.2014.  
<http://www.schoeps.de/en/products/categories/polarflex>
- [19] Schroeder, M. R.: Integrated-Impulse Method for Measuring Sound Decay without Using Impuls. *J. Acoust. Soc. Am.*, ročník 66, 1979.
- [20] Stan, G.-B.; Embrechts, J.-J.; Archambeau, D.: Comparison of Different Impulse Response Measurement Techniques. *J. Audio Eng. Soc.*, ročník 50, č. 4, 2002: s. 249–262.  
<http://www.aes.org/e-lib/browse.cfm?elib=11083>
- [21] Thomas, M.: MLS Theory. [online] 12.3.2014.  
<http://www.commsp.ee.ic.ac.uk/~mrt102/projects/mls/MLS%20Theory.pdf>
- [22] Vlachý, V.: *Praxe zvukové techniky*. Novákových 8, Praha 8: Nakladatelství Muzikus, třetí vydání, 2008, ISBN 978-80-86253-46-5.
- [23] Wikipedia: Fourier transform. [online] 13.3.2014.  
[http://en.wikipedia.org/wiki/Fourier\\_transform](http://en.wikipedia.org/wiki/Fourier_transform)
- [24] Wikipedia: Impulse response. [online] 12.3.2014.  
[http://en.wikipedia.org/wiki/Impulse\\_response](http://en.wikipedia.org/wiki/Impulse_response)
- [25] Wikipedia: Maximum length sequence. [online] 12.3.2014.  
[http://en.wikipedia.org/wiki/Maximum\\_length\\_sequence](http://en.wikipedia.org/wiki/Maximum_length_sequence)

# Příloha A

## Návod k měření

Na následujících stránkách popíšu návod k měření pomocí programu PolarMeasure a přípravku, který jsem za účelem měření zhotovil. K měření je třeba specifického vybavení, zde je seznam:

- mechanická část měřicího přípravku (viz 6.2);
- elektronická část měřicího přípravku (viz 6.3) + DC zdroj 9–30 V;
- počítač se softwarem Matlab, ovladači pro Arduino Uno a nainstalovaným ArduinoIO package v Matlabu;
- zvuková karta s minimální vzorkovací frekvencí 44,1 kHz (na přehrávání);
- zařízení s mikrofonním předzesilovačem (event. i a fantomovým napájením) a možností digitálního exportu při minimální vzorkovací frekvencí 44,1 kHz (na záznam);
- reproduktor (+ zesilovač) se známou frekvenční charakteristikou (resp. impulsní odezvou);
- měřený mikrofon s držákem;
- kabely, stojany a další příslušenství.

Prvním krokem je sestavení měřicí aparatury. Na mechanickou část přípravku připevníme mikrofon a upravíme tak, aby se membrána mikrofonu nacházela nad hlavní osou. Zhruba 30–50 cm od mikrofonu umístíme reproduktor tak, aby byla osa reproduktoru v ose mikrofonu (v případě reproduktorové soustavy tak, aby byl mikrofon v ose výškového reproduktoru). Správné propojení mikrofonu, reproduktoru se zvukovou kartou nechám na uživateli. Motor připojíme přímým UTP kabelem k druhé části přípravku, poté můžeme na přípravek připojit zdroj stejnosměrného napětí (9–30 V). Měla by se rozsvítit kontrolní LEDka. Nepřipojujte napájení pokud není připojena zátěž (motor)! Může dojít k poškození EasyDriveru. Poté připojíme přípravek k PC pomocí USB kabelu. Spustíme Matlab a program PolarMeasure (při prvním použití ho je třeba nainstalovat ze souboru PolarMeasure.prj, který je přiložen na CD).

V případě měření, je třeba nejprve připojit Arduino. Zadáme sériový port, na kterém je Arduino emulováno (ve tvaru 'COM3') a stiskneme „Připojit“. V případě problémů s připojením zkontrolujeme, že je Arduino připojeno, eventuálně ho odpojíme, připojíme a pokus opakujeme. V případě neúspěchu je ještě možné restartovat Matlab. Se samotným informačním oknem v PolarMeasure je třeba také sledovat Command Window Matlabu. Po úspěšném připojení se na přípravku rozsvítí kontrolní LEDka. Nyní můžeme otestovat správnou funkčnost motoru tlačítkem „Spustit test“ v bloku Motor. Pokud je rychlost příliš vysoká a mikrofon sebou škube, je třeba zatrhnout volbu „Snížení rychlosti“. Dále nastavíme stejnou vzorkovací frekvenci jaká je nastavená na zvukové kartě a můžeme výstup rovnou otestovat tlačítkem „Test výstupu“. Tato volba také slouží k nastavení hlasitosti výstupu a zisku vstupu. Poté zvolíme metodu, kterou chceme měřit, zvolíme počet vzorků  $N$  a počet opakování (to především u MLS). Poslední důležitou volbou před měřením je počet měřicích kroků. Pokud máme vše nastaveno, ujasníme se ještě, že má mikrofon dostatečně dlouhý kabel pro otočení o 360°.





# Příloha B

## Popis kódu

V následujících dvou kapitolách jsou popsány stěžejní části kódu a to část určená na měření a na analýzu. Ostatní části kódu lze najít ve zdrojových souborech v příloze na CD.

### B.1 Měření

```
global a;  
global data;
```

Deklarace globálních proměnných,

```
if data.connect==1
```

kontrola připojení Arduina,

```
set(handles.text_info,'String','Probíhá měření...')
```

oznámení o probíhajícím měření,

```
if get(handles.check_delay,'Value')==1  
    pause(str2num(get(handles.text_delay,'String')));  
end
```

kontrola zatržení zpoždění, pokud je zvoleno zpoždění, funkce *pause* čeká danou dobu,

```
data.rychlost=get(handles.check_rychlost,'Value')/100000,
```

nastavení konstanty rychlosti – resp. zpoždění mezi instrukcemi,

```
switch get(get(handles.panel_krok,'SelectedObject'),'Tag');  
    case 'button_1_8',data.krok=1.8;  
    case 'button_3_6',data.krok=3.6;  
    case 'button_7_2',data.krok=7.2;  
    case 'button_14_4',data.krok=14.4;  
end  
switch get(get(handles.panel_fs,'SelectedObject'),'Tag');  
    case 'button_441', data.fs=44100;  
    case 'button_48', data.fs=48000;  
    case 'button_custom', ...  
        data.fs=str2num(get(handles.text_customfreq,'String'))*1000;  
end  
data.N=str2num(get(handles.text_N,'String'));  
data.repetitions=str2num(get(handles.text_repet,'String'));
```

načtení dat k měření – krok, vzorkovací frekvence, N a počet opakování,

```
if isfield(data,'cas')==1  
    data=rmfield(data,'cas');  
end
```

vymazání eventuálního pole *cas* – v případě, že zůstané plné z minulého měření a měříme s menším počtem bodů,

```
switch get(get(handles.panel_metoda,'SelectedObject'),'Tag');
    case 'button_sweep',
        data.metoda=1;
        signal=generate_sinesweeps2(100,22000,data.fs,data.N,...
        data.repetitions);
    case 'button_mls',
        data.metoda=2;
        if data.repetitions==1
            signal=GenerateMLS(data.N,1);
        else
            signal=GenerateMLSSequence(data.repetitions,data.N,1);
        end
end
```

zjištění měřicí metody a vygenerování příslušného signálu, v případě sweepu se do pole *metoda* zapíše 1, v případě MLS 2,

```
stoop=0;
data.stoop=0;
```

reset proměnných, které indikují přerušování měření uživatelem,

```
a.digitalWrite(2,0);
```

nastavení směru motoru – 0 na pin 2 Arduino,

```
data.sync_sum=rand(1*data.fs,1);
yRange=[-0.5,0.5];
soundsc(data.sync_sum,data.fs,yRange);
pause((2*(length(data.sync_sum))/data.fs));
```

vygenerování náhodného šumu, přehrání s rozsahem *yRange* a zpoždění po dvojnásobnou dobu přehrávání (šum potom slouží k synchronizaci začátku),

```
k=1;
while k<((360/data.krok)+1) && stoop==0
    tic
```

začátek cyklu pro daný počet bodů+1 nebo do té doby, než ho uživatel nepřerušuje a začátek měření času,

```
soundsc(signal,data.fs);
pause((length(signal)/data.fs)+0.5);
```

přehrání měřicího signálu,

```
for l=1:((data.krok*8)/1.8)
    a.digitalWrite(3,1);
    pause(data.rychlost);
    a.digitalWrite(3,0);
    pause(data.rychlost);
end
data.pozice=data.pozice+((data.krok*8)/1.8);
```

posílání sekvence 1, 0 na pin 3 (otáčení motoru) dle velikosti kroku a zápis pohybu do pole *pozice*, aby se v případě nutnosti návratu motoru na původní pozici vědělo, kolik kroků motor urazil,



```

        a.digitalWrite(3,0);
    end
    set(handles.text_info,'String','Kabel byl rozmotán.')
end

```

navrácení do původní polohy, vyresetování pole *poloha* a oznámení o otočení,

```
[file_name,path] = uiputfile('*mat');
```

dialog, pro uložení – získá jméno souboru a cestu,

```

full_path = strcat(path, file_name);
set(handles.text_info,'String',['Měření bylo dokončeno,...
údaje o měřicím signálu byly zapsány do souboru ' file_name ' .'])
save(full_path,'data');
end

```

získání celé cesty, oznámení o dokončení měření a uložení souboru a uložení souboru na vybranou cestu,

```

else
    warndlg('Pro spuštění měření připojte prosím Arduino!','...
    'Arduino není připojeno')
end

```

hláška v případě nepřipojeného Arduina, konec měření.

## B.2 Analýza

```
global data2;
```

Načtení globální proměnné,

```

if data2.id_loaded==1
    if data2.sig_loaded==1

```

kontrola načtení id filu, kontrola načtení signálu (*if* je dvojitě kvůli rozdílným varovným hláškám)

```

        delka=sum(data2.id.cas(1:(360/data2.id.krok)))*data2.id.fs...
        +length(data2.id.sync_sum)-2*data2.id.fs;

```

teoretická minimalní délka signálu - součet všech časů + délka počátečního šumu (resp. délka mezery mezi šumem a prvním signálem – je stejně dlouhá jako šum samý) + 2 s rezerva,

```

        rv = length(data2.sig)-delka;
        z_sig = data2.sig(1:rv);

```

rozdílné délky naměřeného a teoretického signálu ve vzorcích a zkrácený signál na maximální možný čas na začátku (za předpokladu, že bylo doměřeno) za účelem minimalizace početního času při korelaci,

```
[hodnota,pozice] = max(xcorr(data2.id.sync_sum,z_sig));
```

zjištění korelačního maxima,

```

        pos=abs(rv-pozice);
        s_sig=data2.sig(pos+2*length(data2.id.sync_sum):end);

```









```

window_sync=window(pos1:pos1+length(signal)-1);
h=xcorr(window_sync,mlss);

```

Zde je jediná změna od analýzy sweepu, a to výpočet IR pomocí korelace (resp. cyklické korelace) zasynchronizovaného naměřeného signálu a originálního MLS. Následující část je opět shodná.

```

    if i==1
        set(gcf,'currentch',char(1))
        axes(handles.axes_ir);
        plot(h); title('Zazoomujte v IR, aby byla videt....
        IR az po první odraz - potvrďte Enterem');
        zoom on;
        waitfor(gcf,'CurrentCharacter',13)
        zoom reset
        zoom off
        title('Kliknete na zacatek IR');
        [x_start]=ginput(1);
        title('Kliknete pred prvni odraz');
        [x_stop]=ginput(1);
        title('Impulsová odezva na 0 degree');
        four=abs(fft(h(round(x_start):round(x_stop),1),...
        data2.id.fs));
        fr=mag2db(abs(four(2:data2.id.fs/2)));
        data2.fr(i,:)=fr;
        axes(handles.axes_fr);
        semilogx(data2.fr(i,:))
        title('Frekvenční charakteristika před korekcí');
    else
        four=abs(fft(h(round(x_start):round(x_stop),1),...
        data2.id.fs));
        fr=mag2db(abs(four(2:data2.id.fs/2)));
        data2.fr(i,:)=fr;
    end
end
end
end

```

Konec analýzy MLS.

```

    if data2.ir_loaded==1
        four=abs(fft(data2.ir,data2.id.fs));
        fr_z=mag2db(abs(four(2:data2.id.fs/2)));
    end

```

V případě, že je načtena IR zdroje, je z ní spočítána frekvenční charakteristika zdroje,

```

    for i=1:(360/data2.id.krok)
        data2.fr(i,:)=data2.fr(i,:)+fr_z';
    end

```

a je přičtena ke frekvenčním charakteristikám na každém úhlu.

```

    axes(handles.axes_fr2);
    semilogx(data2.fr(1,:));
    title('Frekvenční charakteristika po korekci');
end

```

Následně je korigovaná frekvenční charakteristika na  $0^\circ$  vykreslena.

```

if isfield(data2, 'f')==1
    data2=rmfield(data2,'f');
end
if isfield(data2, 'vk')==1
    data2=rmfield(data2,'vk');
end
i=1;

```

Vymazání polí *f* a *vk*, aby nezůstávaly z předchozí analýzy, *i* je 1.

```

if get(handles.check_f1, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f1,'String'));
    i=i+1;
end
if get(handles.check_f2, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f2,'String'));
    i=i+1;
end
if get(handles.check_f3, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f3,'String'));
    i=i+1;
end
if get(handles.check_f4, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f4,'String'));
    i=i+1;
end
if get(handles.check_f5, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f5,'String'));
    i=i+1;
end
if get(handles.check_f6, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f6,'String'));
    i=i+1;
end
if get(handles.check_f7, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f7,'String'));
    i=i+1;
end
if get(handles.check_f8, 'Value')==1
    data2.f(i)=str2num(get(handles.text_f8,'String'));
    i=i+1;
end
data2.pf=i-1;

```

V těchto řádcích jsou čteny frekvence, které zvolil uživatel, v poli *pf* je pak celkový počet frekvencí.

```

data2.theta=-180:data2.id.krok:180;
data2.cc=hsv(data2.pf);

```

*theta* bude osa polárního diagramu, do *cc* se pomocí příkazu *hsv* vygenerují barvy pro jednotlivé frekvenční křivky dle jejich počtu.

```

for k=1:360/data2.id.krok
    for i=1:data2.pf
        data2.vk(k,i)=(data2.fr(k,data2.f(i)));
    end
end

```



# Příloha C

## Obsah přiloženého CD

app	
├─ navod.pdf .....	návod k měření
├─ PolarMeasure.mlappinstall .....	app pro Matlab
namerena_data	
├─ CX111_100_16m1_p.mat .....	ID file
├─ CX111_100_16m1_p.wav .....	naměřený zvukový soubor
├─ CX111_100_16s1_p.mat	
├─ CX111_100_16s1_p.wav	
├─ ir_6010A.mat .....	IR monitoru Genelec 6010A
├─ NT2_100_16s1_kardio.mat	
├─ NT2_100_16s1_kardio.wav	
├─ NT2_100_16s1_koule.mat	
├─ NT2_100_16s1_koule.wav	
├─ NT2_100_16s1_osm.mat	
├─ NT2_100_16s1_osm.wav	
├─ TGI50_100_16s1_s.mat	
├─ TGI50_100_16s1_s.wav	
text	
├─ supka_ondrej.pdf .....	text práce ve formátu PDF
zdrojove_kody	
├─ arduino.m .....	class pro Arduino
├─ Dirplot.m .....	funkce na kreslení polárních diagramů
├─ generate_sinesweeps.m .....	funkce na generování sweepu
├─ generate_sinesweeps_2.m .....	modifikovaná předchozí funkce
├─ GenerateMLS.m .....	funkce na generování samostatné MLS
├─ GenerateMLSSequence.m .....	funkce na tvoření více MLS sekvencí
├─ PolarMeasure.fig .....	GUI v prostředí GUIDE
├─ PolarMeasure.m .....	hlavní funkce

Pzn.: naměřená data jsou pojmenovávána

<mikrofon>\_<početkroků>\_<N><s-sweep;m-MLS><opakování>\_<poznámka>.