

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATION

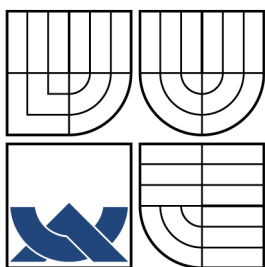
PROGRAMOVÁ IMPLEMENTACE SUBJEKTIVNÍCH TESTŮ
ZVUKOVÉ KVALITY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

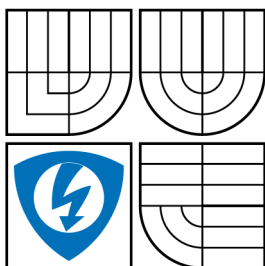
AUTOR PRÁCE
AUTHOR

BC. MAREK ŠPETA

BRNO 2011



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ



FACULTY OF ELECTRICAL ENGINEERING AND
COMMUNICATION
DEPARTMENT OF TELECOMMUNICATION

PROGRAMOVÁ IMPLEMENTACE SUBJEKTIVNÍCH TESTŮ ZVUKOVÉ KVALITY

SOFTWARE IMPLEMENTATION OF THE SUBJECTIVE ASSESSMENTS OF
SOUND QUALITY

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

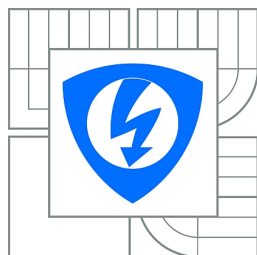
AUTOR PRÁCE
AUTHOR

BC. MAREK ŠPETA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. ONDŘEJ RÁŠO

BRNO 2011



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Diplomová práce

magisterský navazující studijní obor
Telekomunikační a informační technika

Student: **Ó&ĚŤ æ^\ Á] ^æ**

ID: 72179

Ročník: 2

Akademický rok: 2010/2011

NÁZEV TÉMATU:

Programová implementace subjektivních testů zvukové kvality

POKYNY PRO VYPRACOVÁNÍ:

Práce se zpracovává na základě zadání, které je součástí této práce. Práce musí být vypracována v souladu s pokyny uvedenými v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání.

DOPORUČENÁ LITERATURA:

Žák musí seznámit se s literaturou uvedenou v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání.

Práce musí být vypracována v souladu s pokyny uvedenými v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání. Práce musí být vypracována v souladu s pokyny uvedenými v zadání.

Termín zadání: 7.2.2011

Termín odevzdání: 26.5.2011

Vedoucí práce: **Q* ĚŮ} á ^bŮı z[**

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

Autor diplomové práce nesmí při vytváření diplomové 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 diplomová práce se zabývá subjektivními metodami testování kvality zvuku dle doporučení Mezinárodní telekomunikační unie ITU. Práci lze tématicky rozdělit na čtyři části. V první části je to výklad metod vycházejících z mezinárodně uznávaných norem (metoda nepatrných zhoršení, MUSHRA, obecné metody). Druhá část popisuje funkční bloky aplikace vytvořené pro tuto práci v prostředí LabVIEW. Další část vysvětluje použití této aplikace v praxi, zejména možnosti jejího spouštění. V poslední části je zpracován poslechový experiment. Jeho cílem bylo ověření funkcí aplikace a srovnání výsledků použité subjektivní metody s objektivní metodou PEAQ.

KLÍČOVÁ SLOVA

subjektivní testy, objektivní testy, PEAQ, MUSHRA, LabVIEW, SDG, ODG, seřazování do pořadí

ABSTRACT

The focus of this diploma thesis is on methods for the subjective assessment of sound quality according to recommendations given by International Telecommunication Union ITU. The thesis is thematically divided into four parts. The first part is an interpretation of methods based on internationally accepted standards (method of small impairments, MUSHRA, general methods). The second part describes the functional blocks of application developed for this thesis in LabVIEW environment. Next part explains its practical application, especially its running possibilities. The last part describes a listening experiment, aim of which was to verify the application's features and to compare the results of the subjective method used in the experiment with the results of the objective method PEAQ.

KEYWORDS

subjective assessment, objective assessment, PEAQ, MUSHRA, LabVIEW, SDG, ODG, Rank Order

ŠPETA M. *Programová implementace subjektivních testů zvukové kvality*. Brno: VUT v Brně. Fakulta elektrotechniky a komunikačních technologií. Ústav Telekomunikací, 2011. 69 s. Vedoucí diplomové práce Ing. Ondřej Rášo.

PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma „Programová implementace subjektivních testů zvukové kvality“ jsem vypracoval samostatně pod vedením vedoucího diplomové 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 uvedené diplomové práce dále prohlašuji, že v souvislosti s vytvořením této diplomové práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení § 152 trestního zákona č. 140/1961 Sb.

V Brně dne

.....

(podpis autora)

OBSAH

Úvod	10
1 Kvalita zvuku a její měření	11
1.1 Objektivní metody - současný stav	11
1.2 Subjektivní metody měření kvality zvuku	13
1.2.1 Obecné metody	14
1.2.2 Metoda pro střední kvalitu	14
1.2.3 Preselektivní metoda	14
1.2.4 Metoda pro nepatrné zhoršení	14
1.3 Základní pojmy	15
2 Metoda nepatrného zhoršení	16
2.1 Výběr posluchačů	17
2.1.1 Pre-screening	17
2.1.2 Post-screening	18
2.2 Metodika	18
2.2.1 Atributy	19
2.3 Výběr podnětů	20
2.4 Reprodukce	21
2.4.1 Poslechové parametry	21
2.5 Testovací postup	22
2.5.1 Tréninková seznamovací fáze	22
2.5.2 Hodnotící fáze	22
2.6 Zpracování výsledků	23
2.7 Metoda preselekce	24
2.7.1 Výběr posluchačů	24
2.7.2 Metodika	24
2.7.3 Zpracování výsledků	24
3 MUSHRA	25
3.1 Výběr posluchačů	25
3.2 Metodika	25
3.3 Zpracování výsledků	26
4 Obecné metody	28
4.1 Párové srovnávání – AB test	28
4.1.1 Výběr posluchačů	28
4.1.2 Metodika	28

4.1.3	Další kroky testu	29
4.2	Seřazování do pořadí - Rank Order	30
5	Programovací prostředí LabVIEW	31
5.1	Vizuální programování	31
5.2	Programování datového toku	31
5.3	LabVIEW	32
5.3.1	Charakteristika	32
5.3.2	Virtuální instrumentace	33
5.3.3	Práce v LabVIEW	33
6	Návrh aplikace	36
6.1	Motivace	36
6.2	Návrh aplikace	36
7	Výsledná aplikace	38
7.1	Programové součásti	39
7.2	main.vi	40
7.3	choose.vi	40
7.4	AB testy	41
7.4.1	open.vi	41
7.4.2	paths.vi	41
7.4.3	check.vi - kontrola 1	42
7.4.4	list.vi	43
7.4.5	ab.vi	43
7.4.6	randomization.vi	44
7.4.7	play.vi	45
7.4.8	check.vi - kontrola 2	45
7.4.9	about.vi	46
7.4.10	Výstup	46
7.5	MUSHRA test	47
7.5.1	Rozdíly ve vstupní fázi	47
7.5.2	mushra.vi	47
7.5.3	Rozdíly ve výstupní fázi	47
7.6	Rank Order test	48
7.6.1	Rozdíly ve vstupní fázi	48
7.6.2	order.vi	48
7.6.3	Rozdíly ve výstupní fázi	48

8	Použití aplikace	49
8.1	Součásti	49
8.2	Příprava souborů	49
8.3	Výstupní soubory	51
8.4	Způsoby spuštění	51
8.4.1	Zkompilovaný program	51
8.4.2	Originální VI	53
8.4.3	Web server	53
9	Praktické měření	54
9.1	Návrh experimentu	54
9.2	Realizace	54
9.2.1	Poslechový materiál	54
9.2.2	Výběr posluchačů	55
9.2.3	Poslechové podmínky	56
9.3	Průběh experimentu	57
9.4	Zpracování dat	57
9.5	Vyhodnocení	59
10	Závěr	60
	Literatura	61
	Seznam symbolů, veličin a zkratk	64
	Seznam příloh	66
A	Schéma běhu aplikace	67
B	Tabulka naměřených hodnot	68
C	Popis souborů na médiu	69

SEZNAM OBRÁZKŮ

1.1	Objektivní metoda měření SNR.	12
1.2	Schéma objektivní metody využívající modelů lidského vnímání. . . .	12
1.3	Rozhodnutí subjektivních metod.	14
2.1	Křivky NR 10 a NR 15.	22
5.1	Logo pracovní verze programu LabVIEW.	32
5.2	Čelní panel - jednoduché zobrazení sinusoidy.	34
5.3	Blokové schéma - jednoduché zobrazení sinusoidy.	35
6.1	Návrh aplikace pro subjektivní hodnocení kvality zvuku.	37
7.1	Část blokového schématu main.vi.	40
7.2	Blokové schéma choose.vi.	40
7.3	Blokové schéma open.vi.	41
7.4	Blokové schéma paths.vi v režimu kombinace.	42
7.5	Část blok. schématu check.vi.	42
7.6	List.vi - blokové schéma.	43
7.7	Ab.vi - blokové schéma přepínače vzhledu.	44
7.8	Výběr stimulů.	45
7.9	Knuthův algoritmus.	45
7.10	Schéma subVI play.vi	46
7.11	Část blok. schématu check.vi.	46
7.12	Vytvoření kotvy v mushra.vi	48
8.1	Zaváděcí program	52
8.2	Typické GUI - Test malých zhoršení	52
9.1	Spektrum šumu „Bar Crowd“	55
9.2	Spektrum šumu „Empty Casette“	56
9.3	Spektrum šumu „Vinyl“	56
A.1	Schéma běhu aplikace.	67
B.1	Tabulka hodnot	68

ÚVOD

Náplní této diplomové práce je seznámení se s dostupnými doporučeními pro subjektivní testování zvukových systémů a vytvoření aplikace dle požadavků těchto norem. Nedílnou součástí je také nasazení aplikace v praktickém experimentu.

Se slovem „subjektivní“ se pojí také zaměření těchto testů na osobní hledisko lidí, posluchačů (odborně subjektů), kteří tyto testy podstupují a hodnotí subjektivní kvalitu atributů testu. Subjektivní kvalita zvuku je veličina závislá na lidském úsudku a osobní preferenci. Vyjadřuje posluchačovo vnímání daného podnětu [1]. Atributy testu jsou otázky testu zaměřené na kvalitativní posouzení určitých parametrů systému [4].

Doporučeními se zabývá mezinárodní organizace ITU [27]. Tyto normy jsou obecně považovány za referenční. Jednotlivá doporučení jsou založena na obecných poznatcích psychoakustického testování a zaměřují se na všechny fáze testu. Od jeho návrhu, výběru posluchačů a pracovního materiálu, přes parametry poslechové místnosti a reprodukčních zařízení až po zpracování výsledků.

V zadání uvedený zdroj [4] odkazuje na další doporučení, zabývající se obecnou zvukovou kvalitou. Především zdroj [7] nám předkládá kostru, na níž je vystavěna struktura teoretické části zabývající se těmito metodami:

- Method of small impairments – metoda malých zhoršení
- Pre-selection method for small impairments – metoda předvýběru užívaná jako rozšíření předešlé metody.
- MUSHRA – metoda pro střední kvalitu zvuku.
- General methods – obecné metody, mezi které patří AB testy nebo metoda seřazování do pořadí.

Součástí práce je realizace aplikace zahrnující všechny teoreticky prostudované metody. Při návrhu je nutné zohlednit všechny aspekty dané normou. Důležitá je také uživatelská ergonomie s důrazem na modularitu a jednoduchost následné analýzy dat. To jsou také důvody použití programovacího prostředí LabVIEW, které produkuje pochopitelný a snadno upravitelný kód ve formě blokového schématu.

Prověření aplikace je realizováno poslechovým experimentem na úzkém vzorku posluchačů. Výzkumnou otázkou praktického testu je srovnání výsledků zvolené subjektivní metody s hodnotami metodou PEAQ, což je moderní objektivní metoda, jejíž podstatou je softwarová simulace lidského sluchu a vnímání.

1 KVALITA ZVUKU A JEJÍ MĚŘENÍ

Měření kvality zvuku je poměrně složitá disciplína, která je již několik let zkoumána vědeckými týmy po celém světě. Kvalitou zvuku rozumíme to, jak posluchač vnímá zvuk (hudební, řečový) a jak přirozeně tento zvuk zní [1]. Z této definice vyplývá, že pojem kvalita zvuku je značně subjektivní veličina. Každý posluchač má jiné preference a každý posluchač je i nepatrně jinak biologicky vybaven (rozdílné vlastnosti sluchového řetězce vlivem genetickým či vlivem prostředí).

Velkým hnacím motorem potřeby popsat přesné metody stanovení kvality zvuku jsou trendy číslicového zpracování signálu. Za účelem zvýšení efektivity přenosu komunikačním kanálem je potřeba snižovat datový tok zvuku. Kromě známých kompresních algoritmů pro ztrátovou i bezztrátovou kompresi se vědci zabývají i minoritními a málo známými jevy, které lze uplatnit na poslechový řetězec. Může se jednat o nové kompresní algoritmy, nové postupy obohacování zvuků a algoritmy u vícekanálových systémů.

Z uvedených příčin je třeba zpracovaný zvuk posoudit z hlediska kvality výstupu, tedy změřit jeho kvalitu. Existují dva základní principy hodnocení kvality - *objektivní* a *subjektivní*. Není cílem této práce zabývat se řečovými signály, které mají ze své podstaty zcela jiné příznaky (klade se důraz na srozumitelnost) než hudební, či obecné zvukové signály, které jsou naopak předmětem zájmu.

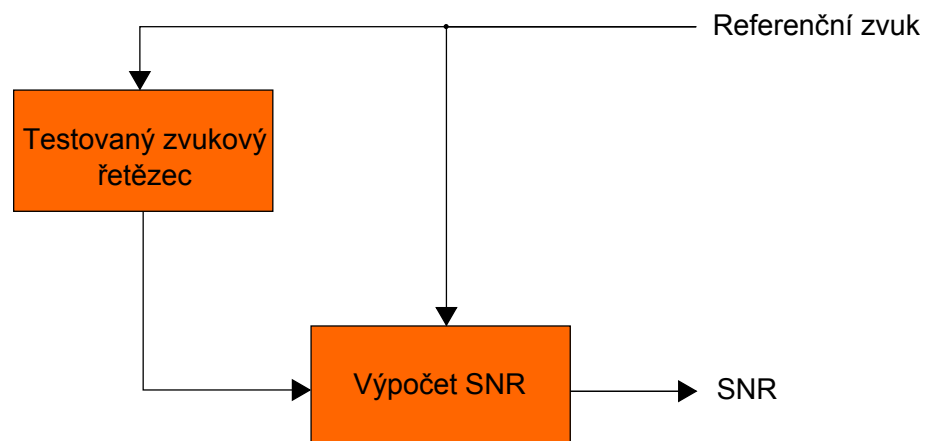
1.1 Objektivní metody - současný stav

Základním předpokladem objektivních metod je jejich zaměnitelná opakovatelnost. Objektivita je tedy dána postupem, který, pokud je dodržen, zaručuje stejný výsledek měření za stejných okolností. Tohoto lze dost těžko u subjektivních metod dosáhnout. Objektivní metody jsou stále velmi studovanou a nejasnou vědeckou otázkou. Jejich cílem je vysoká korelace se subjektivními metodami a jejich motivací je zejména snížení časové a prostorové náročnosti subjektivních testů.

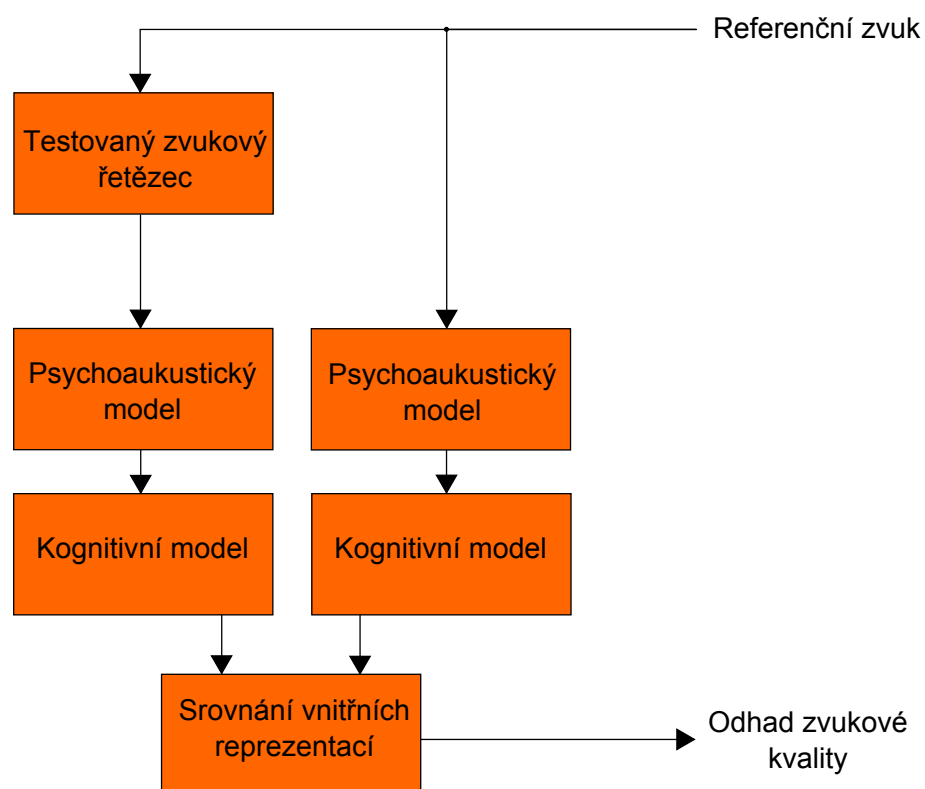
Jednou z prvních a dnes už do značné míry přežitých metod je *určení odstupu signálu od šumu* (SNR - Signal to noise ratio). Ke stanovení SNR je třeba originální i zpracovaný signál. Je to dvoukanálová (intrusivní) metoda [1], viz obr. 1.1.

Dalším stupněm ve vývoji objektivních metod bylo srovnání tzv. vnitřních reprezentací zvuku [2]. Ve stručnosti se jedná o zapojení psychoakustického modelu, tedy matematicko-fyzikálního modelu vnějšího, středního a vnitřního ucha. Dalším krokem bylo zapojení kognitivního modelu simulujícího vnímání zvuku v mozku. Na obr. 1.2 můžeme vidět jednoduché schéma této metody.

V současnosti nejpoužívanější a nejvyspělejší metoda objektivního měření kvality je tzv. **PEAQ** (Perceptual Evaluation of Audio Quality) metoda zpracovaná



Obr. 1.1: Objektivní metoda měření SNR.



Obr. 1.2: Schéma objektivní metody využívající modelů lidského vnímání.

v doporučení [3]. Dochází zde ke srovnání referenčního a zhoršeného signálu. Tyto nahrávky nejprve projdou bloky psychoakustického modelu. Výstupem je několik (až 11) hodnot *MOV* (Model Output Variables). Tyto proměnné jsou pak mapovány do jednoho kvalitativního indikátoru pomocí neuronové sítě s jednou skrytou vrstvou. Výstupem jsou tedy hodnoty *ODG* (Objective Difference Grade), které vykazují vysokou korelaci s *SDG* (Subjective Difference Grade) u moderních subjektivních metod. Hodnoty *ODG* se pohybují v intervalu $\langle -4; 0 \rangle$, přičemž $ODG = 0$ značí neslyšitelnost rozdílu mezi referencí a zhoršeným signálem.

Objektivní metody jsou nadále vylepšovány a zkoumány. Jedním z cílů je také zapojení binárního modelu do řetězce zpracování. Výše zmíněné metody pracují s jednotlivými zvukovými kanály odděleně, což má nepochybně dopad na zkreslení výsledků.

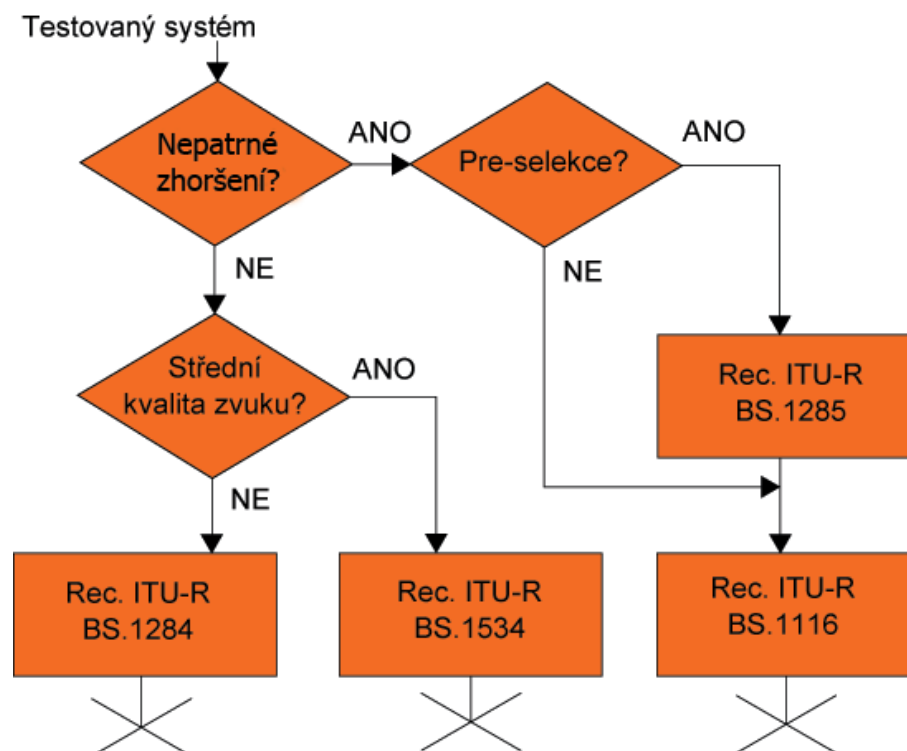
1.2 Subjektivní metody měření kvality zvuku

Dříve, než se začneme věnovat samotným metodám, je potřeba si objasnit několik pojmů. V některých případech jsou pojmy doslovným překladem anglických ekvivalentů a jsou zavedené pro účely této práce:

- **Nepatrné zhoršení zvuku** (*small impairment*) [4] – běžně se jedná o zhoršení zvuku ve velmi kvalitních audiosystémech (včetně kódovacích algoritmů), které je za normálních okolností špatně detekovatelné.
- **Střední kvalita zvuku** (*intermediate quality*) [5] – jedná se o kompromisní kvalitu zvuku, kdy potřebujeme docílit určité komprese za cenu toho, že zhoršená kvalita bude za normálních podmínek snadno postřehnutelná.
- **Pre-selekce** (*pre-selection*) [6] - proces výběru vhodného pracovního materiálu před samotným testem.

Předpokládáme, že vedoucí testu si je jistý, s jakým systémem pracuje a jaké kvality chce docílit. V tomto případě při výběru vhodného subjektivního testu postupuje dle schématu na obr. 1.3. Toto schéma však záměrně vynechává disciplínu, která kromě zvuku vyžívá videosignál [7]. Tento druh je velmi srozumitelně zpracován v doporučení ITU-R BS.1286 [8].

Je zřejmé, že některé metody se mohou překrývat z hlediska použitelnosti. Záleží na účelu testu a podle toho by se měl navrhnout. Je také možné jeden systém otestovat více druhy testů. V této práci se především zaměříme na nejsložitější okruh metod, které se zabývají nepatrnými změnami.



Obr. 1.3: Rozhodnutí subjektivních metod.

1.2.1 Obecné metody

Tyto metody jsou na základě ITU-R BS.1284 [9] určeny pro obecné základní testování kvality zvuku.

1.2.2 Metoda pro střední kvalitu

Tzv. MUSHRA test dle ITU-R BS.1534 [5]. Testy pokrývají zhoršení kvality u systémů, které by u BS.1116 dostávaly známky v horší polovině hodnotící škály.

1.2.3 Preselektivní metoda

Dle ITU-R BS.1285 [6] je určena pro odhalení zbytečného testovacího materiálu pro nejstříktnější testy systému s nepatrnými rozdíly v kvalitě.

1.2.4 Metoda pro nepatrné zhoršení

Tato sekce je založena na ITU-R BS.1116 [4] a používá se pro testování kvality systémů s nepatrnými změnami v kvalitě.

1.3 Základní pojmy

V následujícím textu se vyskytují některé pojmy, jež je důležité předem specifikovat. Část vychází z [4], část je představena pouze pro účely diplomové práce.

- **Vedoucí testu** – osoba zodpovědná za přípravu, průběh a následné hodnocení testu.
- **Subjekt** – posluchač, osoba hodnotící.
- **Test** – soubor zkoušek. Zpravidla se test připravuje za účelem potvrzení nebo vyvrácení hypotézy. Test se provádí za určitých testovacích podmínek, které u subjektivních testů nemohou být nikdy 100% opakovatelné.
- **Zkouška** – zkouška je část testu, ve které subjekt porovnává a hodnotí předložené stimuly.
- **Stimul** – zvukový podnět předkládaný subjektu za účelem ohodnocení systému, jehož je stimul výstupem.
- **Systém** – soubor hardwarových či softwarových filtrů a algoritmů, jejichž výstupem je zvukový podnět.

2 METODA NEPATRNÉHO ZHORŠENÍ

Podstata této metody je založena na předložení tří nahrávek posluchači. Jedna z nich je známá reference a dvě další obsahují referenci a zhoršený stimul. Od subjektu se očekává rozpoznání druhé reference a kvalitativní ohodnocení zhoršené nahrávky. Hlavní nevýhoda této metody je nutnost dodržení velmi přísných podmínek a hodnocení pouze zkušenými posluchači.

Nejprve si zavedeme dva pojmy „základní ustanovení“ a „striktní podmínky výběru posluchačů“, které určitým způsobem charakterizují základní vlastnosti této metody a dále budou rozvedeny, stejně tak na ně bude v dalších kapitolách odkazováno.

Základní ustanovení:

- Přísná kontrola testovacích podmínek.
- Nevhodné pro větší zhoršení.
- Důsledný návrh experimentu pro zamezení nekontrolovatelných faktorů.
- Náhodné přehrávání podnětů v rámci zkoušky i v rámci testu.
- Dvacet posluchačů, dvacet sekundové podněty a maximálně dvacet minut dlouhý test.

Striktí podmínky výběru posluchačů:

- Expertní posluchači preferováni, minimálně obeznámení posluchači.
- Nevybíráme subjekty z široké veřejnosti.
- Čím větší transparentnost mezi podněty, tím větší počet subjektů.

Systémy s nepatrnými změnami definujeme jako pro běžného posluchače téměř transparentní vůči referenčnímu systému. Je použitelná jak u jednobandových, tak i vícebandových systémů. Bez přísné kontroly testovacích podmínek a příslušné statistické analýzy by byly velmi těžko detekovatelné. Pokud se použije pro větší zhoršení, je takovéto testování velmi kontraproduktivní, jelikož se jedná o zbytečně vynaložený čas a úsilí.

Pro kvalitní výsledky je nutné celý experiment správně navrhnout a naplánovat tak, abychom zajistili, zda test neovlivní nekontrolovatelný faktor. Například pokud nevybereme správné nahrávky pro porovnání a tyto budou příliš transparentní, nemůžeme si být jisti, zda posluchači tipovali nebo opravdu nepatrné zhoršení postřehli.

Mohou nastat dvě situace, které je nutné řešit náhodným rozdělením testu. První situací je homogenita změn ve všech nahrávkách (charakteristické vlastnosti

se v testu projevují stejně). Druhou situací je opak, kdy je v jednotlivých dílčích zkouškách obtížnost rozdílná. Proto je vhodné použít aplikaci, která toto (a mnohé další) řeší, viz další kapitoly práce.

Poslechový test musí být také navrhnout s ohledem na to, že posluchači podléhají únavě. V případě velkého soustředění, jaké je u těchto případů očekáváno, se ospalost a snížení citu sluchového orgánu dostavuje rychle.

2.1 Výběr posluchačů

Ukázalo se, že pokud jsou dodrženy přísné podmínky jak po technické, tak po funkční stránce, jsou postačující data od dvaceti subjektů. Pokud máme možnost analyzovat data v průběhu testu, je ho možné ukončit (po dosažení adekvátní statistické významnosti).

Pokud jsou zhoršení v systému tak minimální, že jsou změny téměř transparentní, je potřeba mnohem většího množství posluchačů. Je pak na vedoucím testu a jeho profesionálním odhadu, kolik subjektů je potřeba. Stejná situace nastává, pokud nemůžeme zaručit přísné poslechové podmínky dle normy [4]. Obecně platí, že čím větší je počet posluchačů, tím přesvědčivější výsledky získáme.

Nevybíráme skupinu posluchačů z široké veřejnosti, preferujeme expertní posluchače. Se vzrůstající náročností i tuto skupinu musíme dále zužovat. Cílem výběru je skupina expertů, kteří jsou schopni nejenom zjistit jemné degradace signálu, ale také musí být schopni kvantitativně odhadnout tuto jemnost na dané škále.

V některých případech máme důvod k zavedení zamítacích technik před (pre-screening) a po (post-screening) testu. Obě techniky mohou být použity zároveň. Následkem je vyřazení všech potenciálních (pre-screening) i již proběhlých (post-screening) hodnotících dat určitého subjektu z testu. Pokud tyto techniky nepoužijeme citlivě, mohou vést k posunutí výsledku. Musíme do zprávy detailně popsat použití technik.

2.1.1 Pre-screening

Při pre-screeningu vybíráme posluchače na základě audiometrických testů, předchozích zkušeností a výsledků nebo na základě statistické analýzy vhodného testu.

Jedním z hlavních důvodů pro zavedení této techniky je zvýšení efektivity poslechového testu. Pokud však použijeme nepřiměřeně přísná pravidla může dojít k degradaci výsledků.

2.1.2 Post-screening

Tyto techniky jsou primárně určeny pro eliminaci subjektů, kteří nejsou schopni učinit patřičné hodnocení. Tyto techniky se rozdělují na dvě třídy. První třída je založena na inkonzistenci se střední hodnotou a druhá na schopnosti subjektu správně identifikovat signály.

2.2 Metodika

Pro zvolený případ systémů, které generují nízké zhoršení, je zejména určena metoda „double-blind triple-stimulus with hidden reference“. Českým překlad by mohl být „dvojitě-slepý test se třemi podněty a skrytou referencí.“

- Dvojitě slepý znamená, že subjekt má informace pouze z podnětů a vedoucí testu nijak nezasahuje do procesu.
- Tři podněty odpovídají třem signálům, ze kterých subjekt vybírá právě jeden k poslechu.
- Skrytá reference - dva ze tří zvuků jsou identické a referenční, ale pouze jeden je označená reference.

U nejcitlivějších případů je požadováno, aby byl testován právě jeden posluchač. Mělo by být na posluchačově vůli, který zvuk „A“, „B“, „C“ si vybere pro poslech. Reference se vždy přiřazuje zvuku „A“. Podněty „B“ a „C“ tak, aby právě jeden obsahoval referenční signál a druhý odpovídal testovanému systému. Jakýkoliv rozdíl mezi referencí a dalším podnětem se považuje za zhoršení.

Jakmile posluchač dokončí jednu zkoušku v testu, musí být možné pokračovat další zkouškou. Test se tedy sestává z jednotlivých zkoušek. Při jedné zkoušce má posluchač k dispozici jednu sadu podnětů. Zkouška je považována za hotovou, až posluchač učiní posudek, ohodnotí podněty.

Hodnotící škála byla představena v [9], viz tabulka. 1. Aplikuje se jako „Plynulá stupnice s pěti definovanými body“. V praxi se používá číslování se zaokrouhlením na jedno desetinné místo.

Zhoršení (<i>Impairment</i>)	Hodnocení
Neznatelné - <i>Imperceptible</i>	5.0
Znatelné, ne nepříjemné - <i>Perceptible, but not annoying</i>	4.0
Trochu nepříjemné - <i>Slightly annoying</i>	3.0
Nepříjemné - <i>Annoying</i>	2.0
Velmi nepříjemné - <i>Very annoying</i>	1.0

Tabulka č. 1 - Hodnotící škála.

2.2.1 Atributy

Atribut je vnímaná charakteristika poslechové události. Jednoduše řečeno to, na co se u testu ptáme, co posluchač hodnotí. Je potřeba si uvědomit, že pokud má subjekt hodnotit více než jeden atribut při každé zkoušce, může být více zmatený než jistý, a proto se tento způsob nedoporučuje. Volíme proto raději časově náročnější cestu dvou rozdílných testů (s totožným materiálem).

Atribut	Jednokanálové	Dvoukanálové	Vícekanálové
Základní zvuková kvalita - <i>Basic audio quality</i>	x	x	x
Kvalita stereofonického obrazu - <i>Stereophonic image quality</i>		x	
Kvalita předního obrazu - <i>Front image quality</i>			x
Kvalita prostorového dojmu - <i>Impression of surround quality</i>			x

Tabulka č. 2 - Atributy testu.

- **Basic audio quality** (*Základní zvuková kvalita*) - Jediný atribut hodnotící všechny rozdíly mezi referencí a hodnoceným systémem. U vícekanálových systémů se definuje jako lineární kombinace všech následujících atributů.
- **Stereophonic image quality** (*Kvalita stereofonického obrazu*) - Hodnocení systému dle dopadu na hloubku stereofonního obrazu, přirozenost a pocit reálnosti.
- **Front image quality** (*Kvalita předního obrazu*) - Hodnotí se lokalizace předních zdrojů, vychází z předchozího atributu.

- **Impression of surround quality** (*Kvalita prostorového dojmu*) - Hodnotí se prostorové efekty a dopad na vjem prostoru a podobně.

Lepší představu o attributech hodnotící dvou a vícekanálové systémy získáme v mezinárodní normě IEC 60268 [10], která definuje následující pod-atributy, na které se subjekt zaměří.

- **Image localization** (*Obrazová lokalizace*) – Zkoumá se, zda se mění směr některých zdrojů zvuku.
- **Image stability** (*Stabilita obrazu*) – V některých případech se může hloubka, přirozenost a reálnost chovat velmi nepředvídatelně v závislosti na intenzitě či barvě.
- **Width homogeneity** (*Homogenita šířky*) – Jednotlivé kanály nemusí uniformě přenášet informaci o prostoru a tím narušovat stereofonní obraz.

2.3 Výběr podnětů

Používá se pouze kritický materiál, to je takový, který namáhá testovaný systém. Bohužel neexistuje žádný obecný programový materiál, který by mohl být použit pro všechny systémy a všechny podmínky. Materiál musíme vybrat speciálně pro každý systém. Toto hledání je časově náročné, ale ušetřený čas může v případě unáhlenosti vést až k nesprávnému zhodnocení dat a k mylnému závěru.

Počáteční podmínkou je zvukový materiál, pro který je systém navržen. Pokud bude například kódovací algoritmus použit na zpracování hudební nahrávky, implicitně použijeme různé žánry hudební produkce. Musíme se ale vyvarovat toho, aby materiál nebyl příliš nudný, nebo naopak příliš atraktivní pro posluchače. Také by mu neměl být nepříjemný. Jedině tak bude subjekt plně soustředěn na experiment.

Odpovědnost za vybrané podněty by měla být převedena na skupinu zkušených posluchačů, kteří jsou obeznámeni s očekávaným zhoršením. Tato skupina by měla mít přístup k široké paletě materiálu. Stejná skupina také rozhodne s jakou hlasitostí budou jednotlivé podněty prezentovány tak, aby poslechové zařízení mohlo být nastaveno na jednu úroveň po celou dobu sezení (na které se taktéž dohodnou).

Minimální počet zkoušek je pět a délka jednotlivých podnětů by neměla přesáhnout pětadvacet sekund. Celé sezení (tedy všechny zkoušky pro jednoho člověka) by nemělo překročit třicet minut.

2.4 Reprodukce

Pro reprodukci testovaného materiálu používáme referenční reproduktory nebo sluchátka. Tyto vybíráme tak, aby jakýkoliv testovaný signál byl reprodukován optimálně s neutrálním zvukem. Neutrální reprodukci zajišťují reprodukční zařízení, která neobohacují kmitočtové pásmo a mají ideálně lineární kmitočtovou charakteristiku.

Některé změny jsou více znatelné při sluchátkovém poslechu a jiné u reproduktorů. Musíme si tedy předem ujasnit, jakým směrem se vydáme a pokud testovaný systém působí na stereofonický vjem, měli bychom použít reproduktory.

Můžeme také test přichystat tak, že jedna skupina subjektů bude mít k dispozici sluchátka a druhá reproduktory. Snížíme tím efektivní počet subjektů, ale můžeme pozorovat vliv systému u obou reprodukčních zařízeních. Je také možné posluchači umožnit přepínání mezi reproduktory a sluchátky během každé zkoušky, ale tím nutně ztratíme výhodu předchozí techniky.

V multikanálovém případě je samozřejmé použití výhradně reproduktorů, které musí být nejlépe identické, nebo alespoň akusticky zaměnitelné.

2.4.1 Poslechové parametry

V tomto projektu se zaměříme na poslech pomocí sluchátek. Parametry reproduktorů a poslechové místnosti pro reproduktory jsou definovány v ITU-R BS.1116 [4].

Používáme výhradně referenční sluchátka, to jsou taková sluchátka, jejichž kmitočtová odezva difúzního pole je v mezích obálky udané v ITU-R BS.708 [11] a časové zpoždění mezi kanály nepřesáhne $20 \mu s$.

Poslechovou místností pro sluchátkový poslech může být jakákoliv místnost, jejíž šum/hluk pozadí nepřekročí dané limity. Hluk pozadí vzniká průnikem vnějších elementů dovnitř místnosti, hlukem zařízení apod. Hluk pozadí je nepřetržitý, neobsahuje špičky vzniklé nepředvídatelnými ději.

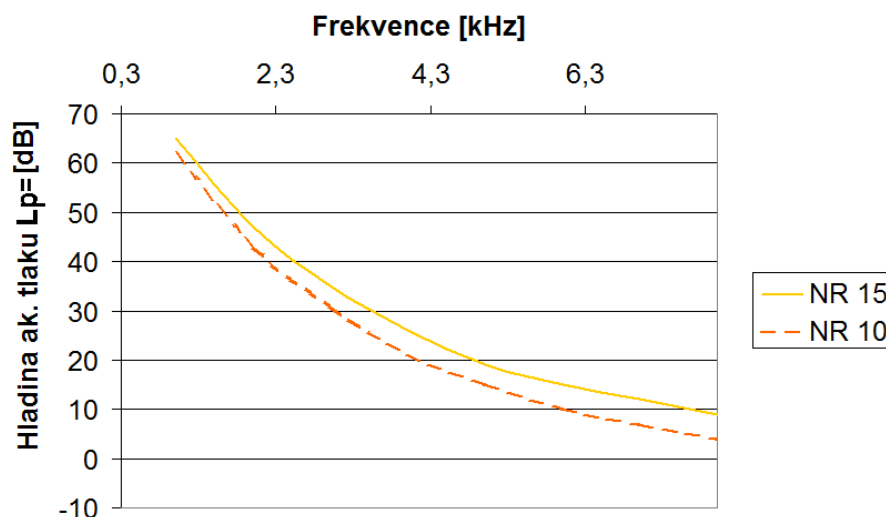
Naměřený hluk v poslechovém místě $1,2 m$ nad podlahou by neměl překročit hodnoty dané křivkou NR 10 a v žádném případě NR 15, viz (obr. 2.1). NR křivky (Noise rating curves) definují závislost hladiny akustického tlaku na kmitočtu a tvoří obálku, do níž spadající charakteristika splňuje požadavky na hladinu zvuku dané právě touto křivkou.

Hladina akustického tlaku (SPL - Sound pressure level, značí se L_p) je definována jako logaritmus poměru mezi naměřeným akustickým tlakem p_{rms} a referenční hodnotou akustického tlaku $p_{ref} = 20 \mu Pa$ [18].

$$L_p = 20 \cdot \log_{10} \left(\frac{p_{rms}}{p_{ref}} \right) [dB] \quad (2.1)$$

	Stř. frekvence oktávových pásem v (HZ)								
Křivka	31.5	62.5	125	250	500	1000	2000	4000	8000
NR 0	55	36	22	12	5	0	-4	-6	-8
NR 10	62	43	31	21	15	10	7	4	2
NR 20	69	51	39	31	24	20	17	14	13

Tabulka č. 3 - Hodnoty ak. tlaku v třetino-oktávovém pásmu [22].



Obr. 2.1: Křivky NR 10 a NR 15.

2.5 Testovací postup

2.5.1 Tréninková seznamovací fáze

Před hodnoticí fází je potřeba všechny subjekty seznámit s testovacím prostředím, prvky, hodnocením, škálou a vším, s čím dojdou v průběhu testu do kontaktu. Posluchači se seznámí s hledanými artefakty ve zvuku a v citlivých případech mohou být seznámeni s celým testovacím materiálem. Je vhodné, aby všichni posluchači prošli tréninkem v jedné skupině, mohou tak mezi sebou diskutovat o dílčích problémech.

2.5.2 Hodnoticí fáze

Navazuje na předchozí fázi dle náročnosti ihned po tréninku, nebo po pauze. Veškeré podněty by měly být neustále k dispozici pro jejich přepínání. To je z toho důvodu, že lidská paměť je efektivní pouze u krátkodobé sluchové paměti. V nejkritičtějších

případech je nutné, aby byli posluchači vystaveni podnětům jednotlivě (neplatí u několika uzavřených sluchátek). Jedině tak mají naprostou volnost při přepínání mezi podněty a můžou detailně prostudovat jejich obsah [4].

Subjekt by neměl mít potřebu se dívat na přepínací mechanismus (nehledal myš či klávesy) a tento mechanismus nesmí vydávat žádné zvuky.

Pokud máme nachystáno více než patnáct zkoušek je nutné hodnotící část rozdělit do více sezení. V praxi se totiž ukázalo, že při 10-15 zkouškách se délka sezení pohybuje mezi 20-30 minutami. Tato doba už má dopad na posluchačovu únavu.

2.6 Zpracování výsledků

Výstupem testu jsou absolutní hodnoty, které posluchači udali při testování. Střední hodnota rozličných parametrů je prvním výsledkem deklarované práce. Pro statistickou analýzu se však tato data nehodí, proto definujeme SDG (Subjective Difference Grade) jako:

$$SDG = GRADE_{systemundertest} - GRADE_{referencesystem}. \quad (2.2)$$

SDG je tedy hodnota rozdílu mezi hodnotou reference a testovaného systému [4]. Z metodiky vyplývá, že se hodnota SDG pohybuje v intervalu $\langle -4; 0 \rangle$, podobně jako *ODG* hodnoty v objektivní metody PEAQ (kap. 1.1). Stejně tak platí, že pro $SDG \rightarrow 0$ je výsledek srovnání reference a stimulu transparentní, tedy neslyšitelný. Grafickou prezentací *SDG* můžeme čtenáři ukázat transparentci (resp. netransparenci) jednotlivých systémů.

Při statistické analýze volíme většinou metodu ANOVA a hladinu významnosti 0.05 tak, jak to doporučuje norma [4]. ANOVA je statistická metoda zaměřující se na střední hodnoty skupin celku a celek samotný. Základním cílem takové analýzy je identifikace průměrného výkonu testovaného systému a dopad jakýchkoli rozdílů na tyto hodnoty. Dalším cílem je odhad variability a variance výsledků testu. Jiné statistické metody jsou používány pro specifické hypotézy, které chceme potvrdit či vyvrátit. Nutno podotknout, že u dalších metod uvedených v této práci je statistická analýza méně časově náročná (kap. 3.3).

Výsledky experimentu včetně popisu přípravy a samotného postupu přehledně zpracujeme v závěrečné zprávě, jejíž forma je předepsána v normě [4].

2.7 Metoda preselekce

Jelikož má metoda pro nepatrné zhoršení značnou časovou náročnost (kap. 1.2), byla ustanovena norma [6] pojednávající o metodě preselekce. Jejím jediným účelem je snížení chybovosti metody malých zhoršení na základě vyloučení nesprávného poslechového materiálu. Jednoduše lze tuto metodu charakterizovat jako výběr poslechového materiálu za účasti zkušených posluchačů. Vedoucí vybere širší skupinu podnětů a poslechová skupina ji pomůže zúžit na výběr nejspornějších stimulů, co se týká kvality.

Při navrhování testu bychom měli postupovat podle **základních ustanovení** uvedených v kapitole 2.

2.7.1 Výběr posluchačů

Výběr posluchačů by měl být oproti předchozí metodě ještě striktnější s důrazem na experty a subjekty obeznámené s chováním a zhoršeními v testovaném systému. Vycházíme z poznatků pasáže 2.1.

2.7.2 Metodika

Přejímáme metodiku z podkapitoly 2.2, tzn. využíváme fakticky test pro nepatrné zhoršení. Jelikož se jedná o časově nejnákladnější část, můžeme v tomto případě pre-selektivního testu ustoupit z přísných podmínek. Základem pro snížení náročnosti je nižší počet subjektů s vysokými nároky na obeznámení s vlastnostmi sledovaného zhoršení. U systémů, jejichž zhoršení nejsou vázána na reproduktory, je možné zavést poslech a hodnocení ve skupině za pomoci několika referenčních sluchátek [11]. Tímto ale není možné zajistit náhodné rozdělení stimulů pro každý subjekt, proto se doporučuje rozdělit posluchače do několika skupin a v každé této skupině docílit různého pořadí stimulů ve zkoušce [6].

Ještě výraznější úspory lze docílit použitím některého z testů obecných metod z kap. 4.

Atributy testu, reprodukční zařízení a programový materiál jsou vybírány na základě doporučení [4] resp. dle kapitoly 2.2.1 a dále.

2.7.3 Zpracování výsledků

Je nutné si uvědomit, že tato metoda není určena k prezentování výsledků. Vedoucímu testu pomáhá určit, zda programový materiál je vhodný pro metodu nepatrného zhoršení, či ne.

3 MUSHRA

Metoda resp. test MUSHRA (MULTi Stimulus test with Hidden Reference and Anchor - Více-stimulový test se skrytou referencí a kotvou) je nejnovější standardizovanou metodou pro subjektivní testování. Při testování je posluchači předloženo několik nahrávek včetně známé reference. Nahrávky obsahují jak skrytou referenci známou z kapitoly 2, tak tzv. kotvu, což je nejvíce zhoršený stimul ve zkoušce. V podstatě je tak subjektu zaměřena celá hodnotící škála mezi tyto dva body (referenci a kotvu). Všechny pojmy budou přesněji vysvětleny dále v textu.

Metodika se zaměřuje na hodnocení programového materiálu dnes hojně využívaného při přenosu zvuku prostřednictvím internetu, satelitního vysílání a podobně. Jedná se tedy o systémy, kde očekáváme jistou sníženou kvalitu a i pozornější posluchač (nejen expert) ji může odhalit [5].

Při navrhování testu bychom měli postupovat podle **základních ustanovení** uvedených v kapitole 2.

3.1 Výběr posluchačů

Na rozdíl od **striktních podmínek** můžeme využít i méně zkušené posluchače. Nezkušené posluchači nejsou doporučeni, ale ani vyloučeni. Platí, že čím zkušenější posluchače pro hodnocení máme, tím vyšší je efektivita celého procesu (menší počet subjektů, absence vylučovacích metod). Pokud nemáme jistotu zkušenosti subjektů je vhodné stejně jako v testu nepatrných metod zavést zamítací techniky (kap. 2.1). U post-selektivní techniky počítáme s faktem, že střední kvalita je snadno rozpoznatelná a tedy extrémní výchyly v hodnocení lze lépe rozpoznat a eliminovat.

3.2 Metodika

Metodika vychází ze **základních ustanovení**. Jedna zkouška testu MUSHRA může obsahovat maximálně patnáct stimulů, kde jeden stimul je známá reference, druhý stimul je náhodně skrytá reference, třetí je náhodně skrytá kotva a zbývající stimuly jsou testované systémy. Kotva v tomto případě slouží pro lepší vymezení hodnotící škály mezi ní a referencí. Mezi nejvíce používané kotvy patří referenční signál filtrovaný dolní propustí s mezním kmitočtem $3,5\text{kHz}$ [5]. Z tohoto faktu odvodíme, že kotva má nejhorší poslechové vlastnosti a posluchač ji tak ohodnotí nejnižší známkou. Pokud subjekt rozpozná kotvu a zbývající podněty srovnává i s referencí, dostáváme jemnější hodnocení za předpokladu, že kotva získá nízké hodnocení. Platí i fakt, že žádný stimul nesmí přesáhnout délku dvaceti sekund.

Hodnotící škála byla představena v [9], viz tabulka 4. Aplikuje se jako „Plynulá stupnice se šesti definovanými body“. V praxi se používá celé číslování od 0 do 100. Překlad jednotlivých pojmů je převzat z publikace [18].

Quality (<i>Kvalita</i>)	Hodnocení
Excellent - <i>Vynikající</i>	100-80
Good - <i>Dobrá</i>	80-60
Fair - <i>Přijatelná</i>	60-40
Poor - <i>Špatná</i>	40-20
Bad - <i>Nepřijatelná</i>	20-0

Tabulka č. 4 - Hodnotící škála.

Atributy testu, reprodukční zařízení a programový materiál jsou vybírány na základě doporučení [4] resp. dle kapitoly 2.2.1 a dále.

3.3 Zpracování výsledků

Ze základů statistiky [20] je patrné, že získané hodnoty se vztahují pouze na testovaný vzorek posluchačů. V statistice se tento vzorek posluchačů označuje jako **výběr** (výběr z populace). Cílem vedoucího je však z hodnot výběru dosáhnout závěru pro celou populaci. Zavádíme pojem **interval spolehlivosti** zkonstruovaný pomocí **výběrového průměru**. Uvnitř tohoto intervalu pak s určitou vysokou pravděpodobností leží hodnota **populačního průměru**.

Při zpracování výsledků vycházíme z absolutních hodnot, které posluchači udělili jednotlivým systémům v jednotlivých zkouškách, či celých testech. Tyto hodnoty, jak je patrné z hodnotící škály, jsou v rozmezí intervalu $\langle 0, 100 \rangle$. Absolutní škála není vhodná pro prezentaci výsledků, vypočítáme tedy průměrnou hodnotu každé prezentace (stimulu systému), tedy **výběrový průměr**:

$$\bar{u}_{jk} = \frac{1}{N} \sum_{i=1}^N u_{ijk}, \quad (3.1)$$

kde \bar{u}_{jk} je průměrná hodnota stimulu, N počet subjektů hodnotících systém a u_{ijk} absolutní hodnocení stimulů systému jednotlivých subjektů. Podobně můžeme vypočítat průměrnou hodnotu u jednotlivých testů j a testovacích podmínek k [5].

Výsledky je dle [5] nutné prezentovat 95% intervalem spolehlivosti s krajními body:

$$\left[\bar{u}_{jk} - \delta_{jk}, \bar{u}_{jk} + \delta_{jk} \right], \quad (3.2)$$

kde δ_{jk} je polovina **šířky intervalu spolehlivosti**:

$$\delta_{jk} = t_{0,05} \frac{S_{jk}}{\sqrt{N}}. \quad (3.3)$$

Proměnná $t_{0,05}$ označuje **kritickou hodnotu Studentova t rozdělení** při hladině významnosti $\alpha = 0,05$. Studentovo rozdělení, je jedno z rozdělení pravděpodobnosti, jež je při větším výběru totožné s normálním rozdělením. Mezi hladinou významnosti α u Studentova rozdělení a koeficientem spolehlivosti P u intervalu spolehlivosti je úzké spojení:

$$\alpha = 1 - P. \quad (3.4)$$

Hodnotu $t_{0,05}$ odečteme ze statistických tabulek [26] při znalosti **počtu stupňů volnosti**. Stupně volnosti odpovídají jednou dekrementovanému rozsahu výběru, tedy **N-1**. Následující tabulka je pouhý výňatek význačných hodnot z tabulek [26] při zvoleném koeficientu spolehlivosti.

Rozsah výběru	5	10	15	20	25	50	100
Stupně volnosti	4	9	14	19	24	49	99
$t_{0,05}$	2,132	1,833	1,761	1,729	1,711	1,677	1,660

Tabulka č. 5 - Hodnoty kvantil $t_{0,05}$ při 95% koeficientu spolehlivosti.

Výběrová směrodatná odchylka S_{jk} je dána vztahem:

$$S_{jk} = \sqrt{\sum_{i=1}^N \frac{(\bar{u}_{jk} - u_{ijk})^2}{N-1}}. \quad (3.5)$$

Tímto získáme 95% jistotu, že hodnota **populačního průměru** leží uvnitř vypočítaného intervalu spolehlivosti. V praxi se pro zobrazení intervalů spolehlivosti využívá statistických (krabicových) grafů. Závěrečnou zprávu formulujeme dle doporučení [4] [5].

4 OBECNÉ METODY

Následující typy testů vychází z praxí uznávaných zdrojů [18] [23] a jejich provádění se uskutečňuje dle obecného doporučení [9].

4.1 Párové srovnávání – AB test

Základním cílem této metody je párové srovnávání podnětů, kdy jedno srovnání odpovídá jedné zkoušce. Posluchač v obecném případě označuje preferenci jednoho ze stimulů. U tohoto testu by měl být brán ohled na **základní ustanovení** z kapitoly 2. Jelikož se však neočekávají nepatrná zhoršení, můžeme si při návrhu testu dovolit nepoužít referenční stimul. Pokud referenci použijeme, jedná se o zcela čistý podnět bez jakýchkoliv zhoršení.

4.1.1 Výběr posluchačů

V tomto bodě vycházíme opět ze **striktních podmínek** a kapitoly 2.1. V této metodě je povolen přístup laiků k hodnocení. Dle normy [9] je věcí diskuze, zda právě laici nevnáší do testu onen názor široké veřejnosti. Nepopiratelnou věcí však zůstává, že experti dokáží být v testu kritičtější a rychlejší než laická veřejnost. Doporučuje se minimální vzorek deseti expertů a (nebo) dvaceti laiků. Postupem času se také z laiků mohou stát experti. Použití vylučovacích technik je doporučováno jen u nejstriktnějších testů.

4.1.2 Metodika

Metodika vychází ze **základních ustanovení** uvedených v pasáži 2 a v kapitole 3.2. Samotný test se skládá z několika zkoušek, ve kterých dochází k srovnávání dvou stimulů za případné přítomnosti referenčního podnětu. Můžeme pouze hodnotit subjektivní preferenci, zda stimul **A** nebo **B** je v dané zkoušce subjektivně kvalitnější než druhý podnět, nebo můžeme použít následující hodnotící stupnice (překlady vychází z publikace [18]).

Quality (<i>Kvalita</i>)	Hodnocení	Srovnání <i>Comparsion</i>	Hodnocení
Excellent - <i>Vynikající</i>	5	Much better - <i>Mnohem lepší</i>	3
Good - <i>Dobrá</i>	4	Better - <i>Lepší</i>	2
Fair - <i>Přijatelná</i>	3	Slightly better - <i>Trochu lepší</i>	1
Poor - <i>Špatná</i>	2	The same - <i>Stejný</i>	0
Bad - <i>Nepřijatelná</i>	1	Slightly worse - <i>Trochu horší</i>	-1
		Worse - <i>Horší</i>	-2
		Much worse - <i>Mnohem horší</i>	-3

Tabulka č. 6 - Hodnotící stupnice pro AB testy.

Volba konkrétních stupnic závisí na podstatě testu a cílů, kterých chceme docílit. Pokud hledáme například jen nejlepší stimul ze souboru, stačí nám párové srovnání. Jestliže však chceme zjistit, jak hodně jsou od sebe stimuly kvalitativně vzdálené, je přítomnost hodnotící stupnice na místě.

Tento test lze také koncipovat jako srovnání dvou stimulů s třetím stimulem. Jedná se o **ABX test**, kde podnět **X** může být referencí, ale častěji se jedná o identický zvukový podnět s jedním z hodnocených stimulů. Subjekt je vyzván k identifikaci stimulu **A** nebo **B** jako shodného s **X**.

V tomto bodě narážíme na pojem *nucená volba* [18]. Posluchač je formou testu veden k preferenci jednoho z podnětů. Ve většině testů **AB** a **ABX** je nepřipustné, aby měl subjekt možnost tzv. *neutrálního soudu*, tedy v případě prvním preferovat oba stimuly (označit je za shodné) nebo v případě druhém označit oba stimuly za identické s **X**[23].

4.1.3 Další kroky testu

Atributy testu, reprodukční zařízení a programový materiál jsou vybírány na základě doporučení [4], resp. dle kapitoly 2.2.1 a dále. Zpracování výsledků je doporučováno shodné jako v části 3.3 s ohledem na použití hodnotících stupnic. V případě absence škály je vhodné jednotlivé stimuly ohodnotit pořadím dle součtu preferenčních bodů, viz publikace [18], která se tímto tématem podrobně zabývá.

4.2 Seřazování do pořadí - Rank Order

Při využití této metody je posluchači předloženo několik podnětů (maximálně patnáct), jež seřazuje do pořadí. Užití reference je dobrovolné. Subjektům není ve většině případů testování dovoleno přiřadit stejné pořadí několika stimulům. V jistých konkrétních případech se tato funkce povoluje a jedná se o tzv. *slučitelné pořadí* [18]. Národným příkladem použití sluchitelného pořadí je i praktické měření k této diplomové práci, viz kap. 9.

U této metody platí stejné ustanovení a poznatky, jako u předchozího **AB testu**. Liší se pouze v metodice.

5 PROGRAMOVACÍ PROSTŘEDÍ LABVIEW

Programovací prostředí LabVIEW je názorným příkladem toho, že vizuální programování může konkurovat klasickému sekvenčnímu programování. Jeho hlavní výhodou je přehlednost, mezi nevýhody patří občasná složitá realizace základních úloh. V následující kapitole jsou uvedeny základní myšlenky a pojmy spojené s tímto prostředím.

5.1 Vizuální programování

Vizuální programovací jazyky vznikaly nezávisle na vývoji běžných jazyků jako C, Java, atp. V současnosti nalézáme na trhu jak poměrně univerzální prostředí LabVIEW, tak v akademickém prostředí známý Simulink [36], tedy nastavbu prostředí Matlab [37]. Vizuální programování se také usídlilo ve speciálních programech, například prostředí CrypTool se zaměřuje na šifrovací mechanismy v telekomunikaci [38].

Z hlediska syntaktického můžeme vizuální jazyky rozdělit na:

- **Transformované vizuální jazyky** – samotné jádro jazyka má klasický syntax, v programovacím prostředí se však jeví jako vizuální jazyk [13].
- **Nativní vizuální jazyky** – neexistuje syntax popisující jednotlivé funkční prvky. Toto je případ prostředí LabVIEW.

Vizuální jazyky se také liší svou koncepcí, primárním hlediskem. Rozlišujeme tak tři potenciální toky, které můžeme sledovat, a to: **tok dat**, **tok materiálu** a **tok řízení** [21].

5.2 Programování datového toku

Principem této koncepce je tok dat, který je v běhu programu ovlivňován jednotlivými funkčními prvky a elementy. Na rozdíl od klasického pohledu (kdy se vykonávají určité instrukce a tok dat je až vedlejší hledisko), jsou u tohoto typu programování primární spoje a operace mezi nimi jsou až druhotné hledisko.

Asi nejmarkantnější výhodou u takovýchto jazyků je jednoduchá paralelizace. V dnešní době, kdy výkon procesorů stoupá především díky paralelizaci jejich jader se programátoři snaží tento trend sledovat a chod aplikací dělit na paralelní vlákna [21].

U datového toku jsou pouze nastaveny počáteční hodnoty vstupu a pak již sledujeme tok dat jednotlivými funkčními bloky. Funkce se vykonají až k nim data „přitečou“, nebo až je funkce vyvolána. Všechny operace a funkce jsou již v počátku aktivní a připravené na vstup dat.

Jedním z prvních kvalitních průkopníků v této oblasti je kanadský jazyk Prograph [29]. Původně byl vyvinutý pro platformu Macintosh, což byla také jeho největší nevýhoda, protože tato platforma zůstala po celou dobu vývoje jazyka vázána na jedno procesorové jádro.

Nejznámějším a mnohem univerzálnějším zástupcem je LabVIEW americké společnosti National Instruments [28].

5.3 LabVIEW



Obr. 5.1: Logo pracovní verze programu LabVIEW.

5.3.1 Charakteristika

LabVIEW je vývojový systém společnosti National Instruments, která je průkopníkem v oblasti virtuální instrumentace, která poslední dobou zažívá velký rozvoj v oblastech školství, vývoje a výzkumu [15]. Je to systém na úrovni jazyka C a jeho odnoží. Je tedy naprosto univerzální a díky bohatým knihovnám ho lze uplatnit i ve všech fázích výzkumného procesu, tedy sběru, analýzy a prezentaci dat.

Jak už bylo předesláno, LabVIEW je prostředí založené na programování datového toku. Samotný programovací jazyk se nazývá „jazyk G“. Chod programu je jazykem určen jako blokový diagram, ve kterém programátor spojuje funkční bloky virtuálními spoji. Těmito spoji prochází data od vstupu a funkční bloky jsou spouštěny v momentě, kdy tyto bloky mají přístup k datům. Tok dat může být jednoduše rozdělen mezi bloky, proto je jazyk v základu vhodný pro paralelní zpracování. V samotném jádru je zabudován plánovač, který na základě nastavení a hardwaru rozděluje vhodně datový tok ke zpracování. Velkým pozitivem tohoto systému je také

jeho určení k vytváření distribuovaných aplikací, které jsou založeny na principu klient/server.

Vizuální princip programování dovoluje lidem neholdujícím programování vytvářet (pouhým přetahováním myši) virtuální reprezentace jím známého laboratorního vybavení. Programové prostředí LabVIEW s jeho zabudovanými příklady, tutoriály a dokumentací dovolují snadně vytvářet jednoduché funkční aplikace. Na druhou stranou takovýto přístup neznamena, že aplikace bude kvalitní a přehledná. Pro komplexní algoritmy a složité aplikace je nutná znalost gramatiky, topologie a řízení přidělování paměti [17].

5.3.2 Virtuální instrumentace

Virtuální instrumentace je pojem označující náhradu (dočasnou, či trvalou) hardwarových prostředků řešením virtuálním za pomoci softwaru. Podnětem k této náhradě může být prostorová či finanční náročnost klasického řešení. Smyslem virtualizace je zprostředkovat uživateli maximální názornost, rychlý návrh, konfiguraci a realokaci prvků programu [15].

Jedním z příkladů velkého ušetření nákladů na hardware a prostor je použití A/Č převodníku ze zvukové karty jako vstup sondy osciloskopu. Software (LabVIEW) se postará o zpracování a zobrazení dat. Virtuální zařízení však také mohou požadovat specifický hardware, který se nachází v portfoliu National Instruments.

5.3.3 Práce v LabVIEW

Jak už bylo dříve řečeno, LabVIEW je programem pro virtuální instrumentaci. Tento pojem ovlivnil pojmenování jednotlivých aplikací v LabVIEW. VI (Virtual Instrument) je i přípona souborů programů „*.vi“.

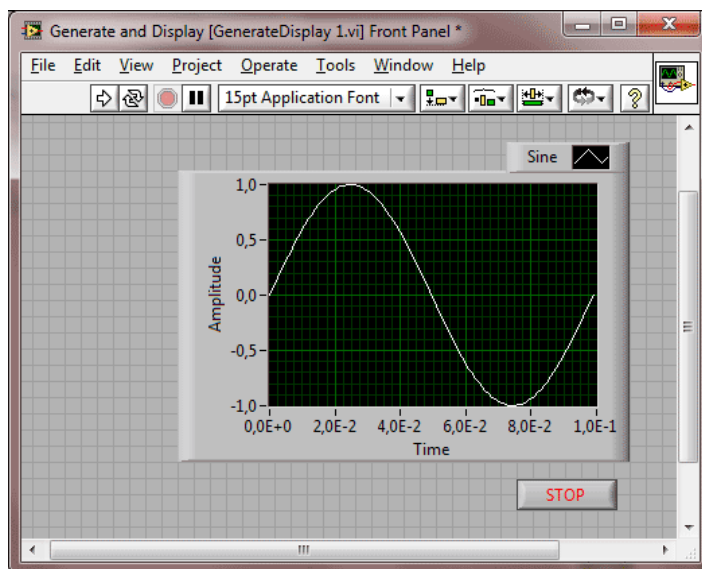
Virtuální přístroj má dvě části:

- **Front panel – čelní panel** - Interaktivní grafické rozhraní (GUI) ke koncovému uživateli.
- **Block diagram – blokové schéma** - zdrojový kód aplikace.

Virtuální přístroj má hierarchickou a modulární strukturu. Lze jej používat jako celý program, nebo využívat jeho jednotlivé podprogramy, které se nazývají podřízenými virtuálními přístroji (SubVI). Dělení programu na SubVI zvyšuje jeho přehlednost a modularitu. Každý VI a SubVI je prezentován svou vlastní ikonou v blokovém schématu [16].

Čelní panel – určuje vzhled a chování aplikace. Obsahuje ovládací a indikační prvky. Ovládacími prvky můžeme řídit a ovlivňovat běh aplikace a zadávat parametry. Indikačními prvky (textové pole, LED dioda, osciloskop) získáme informace o výsledcích zpracování.

Z funkčního hlediska můžeme entity na čelním panelu dělit na *vstupy* a *výstupy*. Vstupy mohou mít podobu různých spínačů, přepínačů, textových polí a otočných knoflíků. Výstupy jsou již zmíněné osciloskopy, ručkové měřiče a grafy. Tyto entity můžeme vytvářet nebo vybírat z velkého množství předprogramovaných prvků programu LabVIEW. U bloků lze měnit velikost barvu a pod. Čelní panel se vytváří jako první.



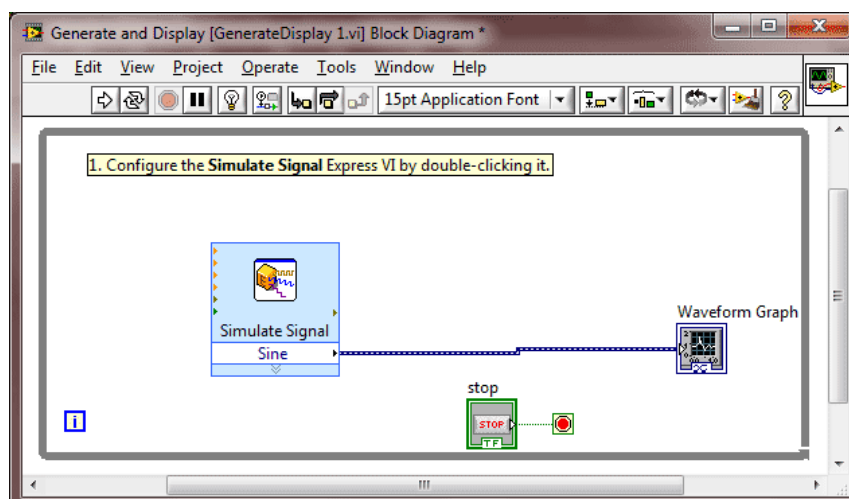
Obr. 5.2: Čelní panel - jednoduché zobrazení sinusoidy.

Blokové schéma – Blokové schéma je grafickým vyjádřením algoritmu programu VI. Konstruuje se propojováním jednotlivých bloků signálovými cestami. Bloky jsou jednak koncové (zdrojové a cílové), které jsou na panel blokového schématu umístěny automaticky při vytváření čelního panelu, a jednak uzlové, které reprezentují zpracování signálu [16]. Každý blok má dle své funkce definované vstupy a výstupy, které jsou reprezentovány přípojnými body.

Základní bloky můžeme rozdělit na:

- **Expresní VI** – interaktivní VI, jsou konfigurovatelné jednoduchým dialogovým oknem, je to většina SubVI z knihovny LabVIEW.
- **Standardní VI** – podprogramy SubVI, které mají blokové schéma i čelní panel. Z velké části se jedná o uživatelem naprogramované podprogramy SubVI.
- **Funkce** – základní operace a primitivy typu násobička, sčítačka.

Jednotlivé bloky propojujeme pomocí signálových cest, tzv. „drátů“. Nelze propojit dva zdrojové bloky, ale lze propojit několik koncových bloků na jeden zdrojový blok.



Obr. 5.3: Blokové schéma - jednoduché zobrazení sinusoidy.

V jazyce G také nachází programové struktury určené pro řízení průběhu výpočtu. Ekvivalentem u klasických jazyků jsou cykly a příkazy pro větvení [15]. Základními strukturami jsou:

- **For Loop** – cyklus s předem daným počtem opakování.
- **While Loop** – cyklus zakončený po splnění podmínky.
- **Case** – přepínač při větvení programu.
- **Flat Sequence** – zajištění postupného provádění funkcí (pokud tak už není naprogramován samotný datový tok).
- **Formula Node** – slouží pro matematický zápis potřebné funkce.

Pokud se budeme hlouběji věnovat LabVIEW zjistíme, že tento systém má výhodu i co se týká práce s pamětí. Funkce a VI automaticky alokují paměť pro data a pokud už nejsou využívána, je paměť uvolněna. Automaticky také dochází k zvětšování alokované paměti, pokud to běh programu vyžaduje.

Dalším pozitivem je vizuální odhalování chyb v aplikaci, kdy vidíme datový tok a k čemu při běhu programu dochází, jelikož datový tok jednoznačně určuje směr provádění programu. S tímto také souvisí funkce *probe*, kde podobně, jako kdybychom používali osciloskop u reálných obvodů, přikládáme sondy k datovým spojům a pozorujeme jaká data (i jejich druh) tudy proudí.

6 NÁVRH APLIKACE

6.1 Motivace

Cílem praktické části diplomové práce je zhotovení aplikace, která bude splňovat následující kritéria:

- Program bude vytvořen ve vývojovém prostředí LabVIEW.
- Program zakomponuje co největší počet subjektivních testů [18] [23].
- Program bude splňovat požadavky doporučení ITU-R [4] [5] [9].
- Formát souboru stimulů bude PCM soubory ***.wav**.
- Aplikace bude co se týká kódu efektivní (využití subVI).
- Jednoduchá použitelnost pro vedoucího i subjekt.
- Srozumitelné GUI.
- Editovatelný výstup s daty pro další zpracování údajů.

Vývojové prostředí Labview je také voleno z důvodu, že obsahuje v základu i moduly pro zpracování zvukových souborů. Existuje také možnost záměny vstupních souborů za konkrétní algoritmus, což při zpracování v reálném čase nabývá na významu.

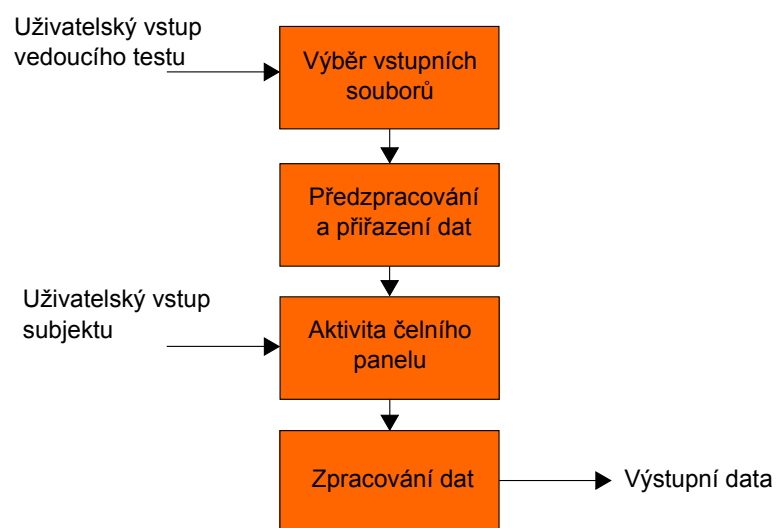
Dle ITU-R BS.1116 [4] by měla aplikace zjednodušit metodický postup a splnit také nejpřísnější kritéria doporučení, jako je náhodné rozmístění skryté reference a také nutnost přenesení ovládání aplikace na subjekt. Posluchač musí mít volné pole působení při přehrávání zvuků, aniž by do tohoto procesu vstupoval vedoucí testu.

Výstupní a vstupní soubory by měly být standardní, zpracovatelné na jakémkoli stolním počítači. Tím odpadá nutnost vyhodnocování dat na PC s LabVIEW.

6.2 Návrh aplikace

Na základě předpokladů můžeme vytvořit blokové schéma budoucí aplikace (VI), viz obr. 6.1.

Na počátku běhu programu je uživatelský vstup vedoucího, který vybere vstupní soubory. Následně dojde ke zpracování těchto souborů na úrovni jejich náhodného rozdělení jednotlivým ovládacím prvům. Zároveň dojde k aktivaci čelního panelu. V tuto chvíli přistupuje uživatel (subjekt) a tyto aktivní prvky použije. Na závěr chodu aplikace dochází k vyhodnocení dat a jejich zpracování na výstup.



Obr. 6.1: Návrh aplikace pro subjektivní hodnocení kvality zvuku.

7 VÝSLEDNÁ APLIKACE

Ve vývojovém prostředí LabVIEW byla vytvořena aplikace, jejíž princip a běh je znázorněn na diagramu v příloze A.1. Na počátku je spuštěno VI *main.vi*, kde je uživatel (vedoucí), vyzván k výběru testu. Tento výběr je zpracován ve VI *choose.vi* a uložen v globální proměnné **test**. Dochází ke spuštění hlavních aplikací pro testování *ab.vi*, *mushra.vi*, nebo *order.vi*. VI *ab.vi* zpracovává testy dle [9] [4] [18], *mushra.vi* vychází z doporučení [5] a *order.vi* je založen na zdrojích [18] [23].

V následující tabulce je kompletní seznam všech testů:

Typ testu	VI	Stručná definice
AB simple	<i>ab.vi</i>	Výběr ze 2 stimulů
AB 5	<i>ab.vi</i>	Hodnocení na 5-kotevní stupnici
AB 5 s referencí	<i>ab.vi</i>	Navíc s referencí
AB 7	<i>ab.vi</i>	Hodnocení na 7-kotevní stupnici
AB 7 s referencí	<i>ab.vi</i>	Navíc s referencí
ABX	<i>ab.vi</i>	Srovnání 2 stimulů se stimulem X
Malé zhoršení	<i>ab.vi</i>	2 stimuly se skrytou referencí
MUSHRA	<i>mushra.vi</i>	2-5 systémů, 4-7 stimulů, kotva, skrytá reference, 7-kotevní stupnici
Rank Order	<i>order.vi</i>	3-10 stimulů, seřazování

Tabulka č. 7 - Implementované testy.

Použití je dále rozšířeno dvěma možnostmi výběru vstupních souborů u AB testů, toto je zpracováno v části 8.2.

Při aktivaci hlavních VI, je nejprve vyvoláno subVI *open.vi* a uživatel (vedoucí) je vyzván k výběru souborů. Formát názvu souborů je zpracován v částech 7.4.2 a 8.2. Validita výběru je konfrontována s vybraným testem (proměnná **test**) v subVI *check.vi*. V případě druhé možnosti formátu názvu souborů (kde) jsou jednotlivé stimuly zkombinovány v subVI *paths.vi*. Cesty k souborům jsou uloženy pro další použití v **paths** a v subVI *list.vi* se vytvoří seznam zkoušek.

Nyní je vstupní fáze dokončena a hlavní VI je aktivní pro uživatele (subjekt). Na pozadí pracuje subVI *randomization*, která náhodně rozděljuje jednotlivé stimuly pod aktivní objekty programu (tlačítka stimulů).

Do výstupní fáze programu patří subVI *play.vi* (resp. *play-mushra.vi*), což jsou podprogramy, jež spouští/zastavují přehrávání jednotlivých stimulů. Opět se uplatní subVI *check.vi* při kontrole přetečení zkoušek. Uživatel může zobrazit informace o programu *about.vi*. Na základě průběhu testu se získané informace ukládají do *.xls souboru v místě vstupních souborů.

7.1 Programové součásti

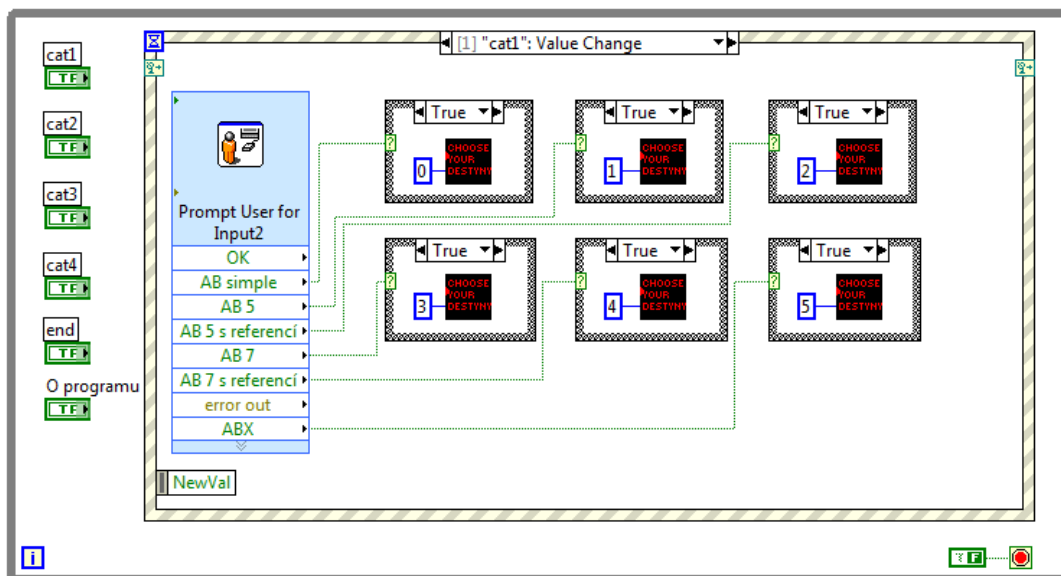
Nezkompileovaná aplikace se dle diagramu z přílohy A.1 a souborové struktury sestává z následujících souborů:

- **main.vi** – výběr testu.
- **main.ico** – ikona aplikace.
- **main.lvproj** – projektový soubor LabVIEW.
- **ab.vi** – hlavní VI AB testů a testu Malých zhoršení.
- **mushra.vi** – hlavní VI testu MUSHRA.
- **order.vi** – hlavní VI testu Rank Order.
- **subVI/about.vi** – zobrazení informací o programu
- **subVI/global.vi** – veškeré globální proměnné.
- **subVI/check.vi** – kontroly.
- **subVI/choose.vi** – spuštění vybraného testu.
- **subVI/list.vi** – vytvoření seznamu zkoušek.
- **subVI/open.vi** – načtení souborů.
- **subVI/paths.vi** – kombinace stimulů.
- **subVI/play.vi** – přehrání a zastavení stimulů.
- **subVI/play-mushra.vi** – přehrání a zastavení stimulů.
- **subVI/randomization.vi** – náhodné rozdělení Knuthovým algoritmem.

Soubor obsahující globální proměnné nemá žádnou funkci, slouží jen k ukládání dat tak, aby k nim bylo možné přistupovat odkudkoliv z kódu (z kteréhokoliv VI). Globální proměnné mají tedy stejnou funkci jako u textových programovacích jazyků, jsou však reprezentovány vlastním ***.vi** souborem.

V následující podkapitole se budeme zabývat blokovými schématy jednotlivých VI aktivních při spuštění AB testu, v navazujících podkapitolách o MUSHRA testu a Rank Order testu probereme pouze hlavní VI a rozdíly ve vstupní a výstupní fázi. Přední panely (uživatelské rozhraní) budou probrány v následující kapitole 8.4.1. Cílem tohoto textu také není podrobně popsat schémata, ale spíše se zaměřit na zajímavé funkční části.

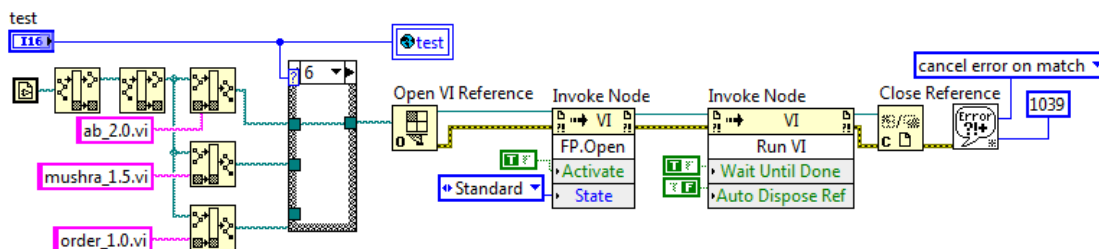
7.2 main.vi



Obr. 7.1: Část blokového schématu main.vi.

Zaváděcí část aplikace má triviální strukturu. Základem je *while* smyčka čekající na uživatelský vstup. Uživatel má k dispozici šest *boolean* tlačítek pro výběr chodu programu. Na výběr reaguje programová struktura *event*, která přiřazuje právě jednu hodnotu *integer* na vstupní konektor **test** subVI *choose.vi*. Na schématu 7.1 je zobrazen jednoduchý způsob výběru AB testu po aktivaci tlačítka **cat1**. Uživatel je vyzván dialogovým oknem k výběru testu, *case* struktury hlídají tento výběr a odesílají data na zmiňovaný konektor **test**.

7.3 choose.vi



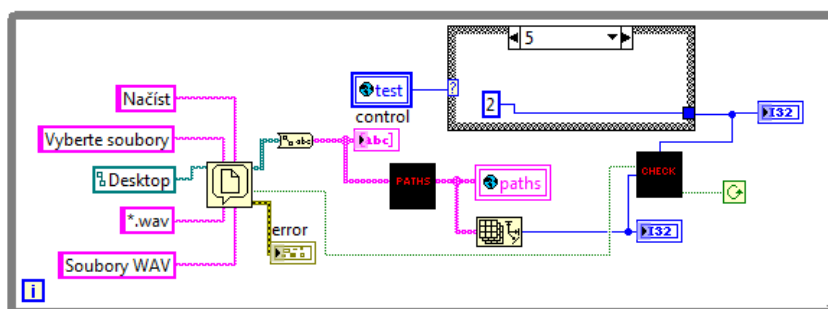
Obr. 7.2: Blokové schéma choose.vi.

Účelem tohoto subVI je spuštění hlavních VI na popředí *main.vi*. Na blokovém schématu 7.2 vidíme, že *integer* hodnota z konektoru **test** je uložena do globální

proměnné **test** a také přivedena na *case* strukturu. Tato struktura slouží jako přepínač jednotlivých vstupních cest. Jedna vybraná cesta je pak zpracována a hlavní VI je spuštěno. Na konci datového toku je ošetření chyby 1039, která nastane pokud zavřeme VI spuštěné na popředí.

7.4 AB testy

7.4.1 open.vi



Obr. 7.3: Blokové schéma open.vi.

Hlavním úkolem tohoto subVI je zobrazení průzkumníku pro výběr souborů (obr. 7.3). Po vybrání souborů jsou cesty převedeny na datový typ *string* a uloženy v globální proměnné **paths**. *Case* struktura nám v tomto případě slouží k přepnutí počtu stimulů u jednotlivých zkoušek (u AB jsou to tedy dva stimuly, resp. tři u testů s referencí). Tento počet **no_of.sys** je přiváděn na konektor subVI *check.vi*, stejně jako informace o počtu souborů **no_of_files** a přepínač **cancel?**, jenž je aktivován v případě, že uživatel nevybere žádné soubory.

7.4.2 paths.vi

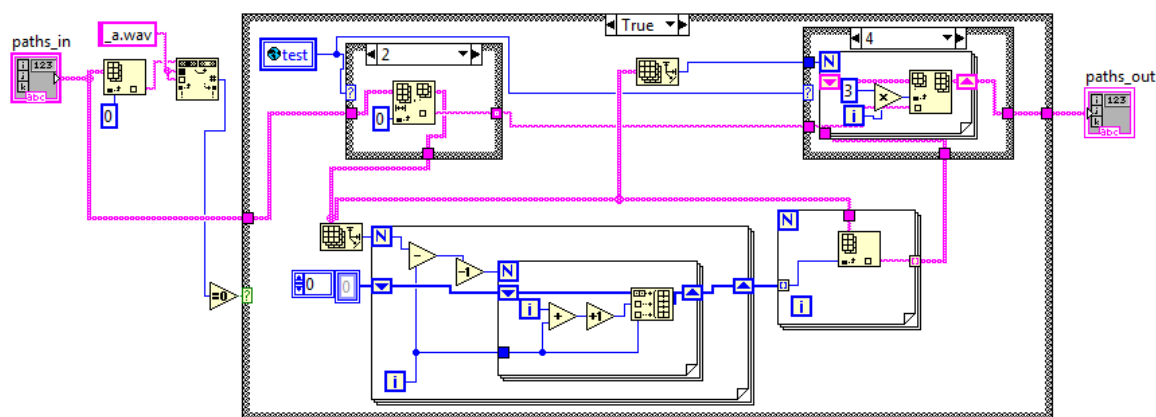
Aplikace očekává určitý formát názvu souborů. Jednotlivé stimuly zkoušek se musí jmenovat dle klíče **zkouskaX.A.wav**, kde **X** je volitelný atribut pořadí (tzn. dále se v kódu nevyužívá) a **A** je písmenné označení stimulů ve zkoušce **X**. Kořenový název souboru **zkouska** také není vyžadován, v tomto textu ho používáme jen pro přehlednost:

zkouska1_a.wav	zkouska1_b.wav	zkouska1_c.wav
zkouska2_a.wav	zkouska2_b.wav	zkouska2_c.wav
zkouska3_a.wav	zkouska3_b.wav	zkouska3_c.wav

Tabulka č. 8 - Příklad formátu názvů souborů pro AB s ref.

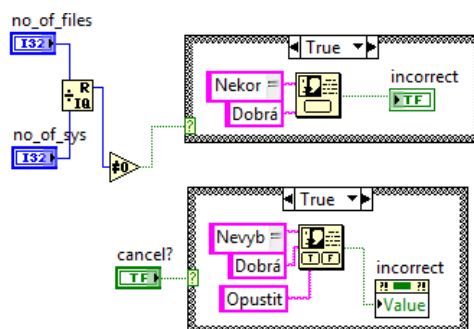
Na začátku datového toku subVI *paths.vi* zkontroluje přítomnost **_a.wav** řetězce v první souborové cestě. Jestliže nastane shoda, je celé pole **paths_in** převedeno *case* strukturou na výstup **paths_out** a následně v *open.vi* uloženo do **paths** proměnné.

Při přípravách testu AB může ale nastat situace, kdy není cílem vedoucího mít připravené jednotlivé dvojice (resp. trojice) stimulů. Cílem je srovnat navzájem jednotlivé stimuly; potřebujeme tedy vytvořit párovou kombinaci bez opakování. Blokové schéma takovéto kombinace je ukázáno na obr. 7.4. Kombinační režim tedy aktivujeme tak, že nedodržíme předešlé předeepsané formátování názvů souborů. Pouze pokud testujeme AB s referencí, první soubor v pořadí musí být reference. V LabVIEW je první soubor v pořadí ten, který je dle abecedy na prvním místě (čehož využívá i předeepsané formátování). Je důležité poznamenat, že v případě testu MUSHRA, Rank Order a Small Impairments je vždy očekáván předeepsaný typ formátování názvů.



Obr. 7.4: Blokové schéma *paths.vi* v režimu kombinace.

7.4.3 check.vi - kontrola 1



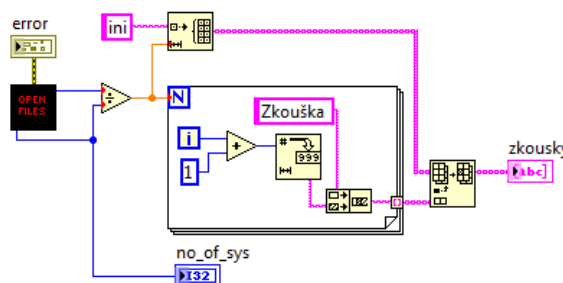
Obr. 7.5: Část blok. schématu *check.vi*.

Kontrola korektnosti spočívá dle obr. 7.5 v podílu celkového počtu souborů počtem stimulů (resp. systémů). Předpokládáme, že vedoucí má pečlivě nachystané vstupní soubory. Pokud má tedy pro AB test s referencí nachystáno 10 souborů, zbytek po vydělení vyjde roven 0. Pokud by vybral 11 souborů, zbytek vyjde neroven 0 a zobrazí se dialog s výzvou pro opětovné vybrání souborů.

Druhá *case* struktura operuje s **cancel?** hodnotou a využije pro zobrazení dialogu v případě nevybrání souborů.

7.4.4 list.vi

Podle diagramu A.1 je subVI *list.vi* mezičlánkem mezi hlavním VI a předchozím subVI. Má poměrně jednoduchou funkci na základě známého podílu z předchozího textu vytvoří pole řetězců ve formátu **Zkouška X**, viz obr. 7.6.



Obr. 7.6: List.vi - blokové schéma.

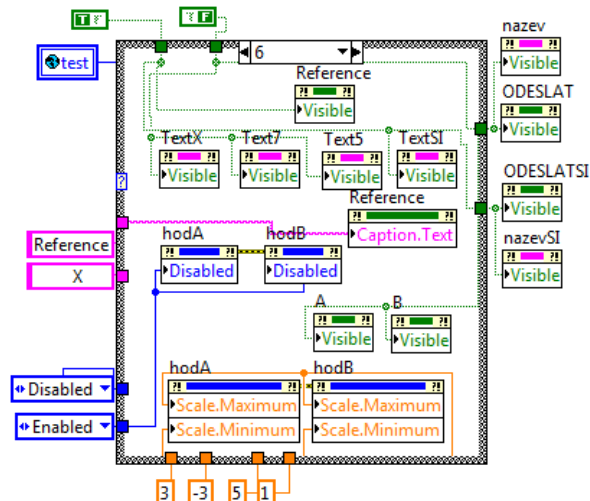
7.4.5 ab.vi

Než se aktivuje samotné uživatelské rozhraní, dochází ke změně jeho vzhledu na základě proměnné **test**, viz obr. 7.7. Pro jednoduchý přehled (blokové schéma ukazuje jen jednu možnost ze 7) jsme vytvořili následující tabulku:

	AB smpl	AB 5	AB 5ref	AB 7	AB 7ref	ABX	SI
Tlač. A/B	x					x	
Úkol	prefer					shoda	
Stupnice	neaktiv	$\langle 1, 5 \rangle$	$\langle 1, 5 \rangle$	$\langle -3, 3 \rangle$	$\langle -3, 3 \rangle$	neaktiv	$\langle 1, 5 \rangle$
Reference			x		x	jen X	x

Tabulka č. 9 - Přepínač vzhledu.

Dále dochází k aktivaci a deaktivaci různých textů nadpisů. Také test malých zmenšení má rozdílné tlačítko pro odeslání dat, jelikož tento test zpracovává data pro výstup odlišně od AB testů.



Obr. 7.7: Ab.vi - blokové schéma přepínače vzhledu.

Hlavní běh programu je situován stejně jako všechny GUI v LabVIEW v smyčce *while*. Dominantní částí této smyčky je struktura *event*, která obstarává akce spojené s uživatelskou činností. Probereme si jen nejvýznačnější události (*events*).

Událost **Probíhá - Value Change** sleduje aktivitu na seznamu zkoušek **Probíhá**(obr. 7.8). Pořadí na seznamu je vynásobeno počtem stimulů ve zkoušce a je proveden výběr z pole cest **paths**. Výběr je přiveden do **randomization.vi**, kde dojde k náhodnému promíchání a uložen v globálních proměnných **randomized** a **ref**. Neustále také probíhá kontrola přetečení seznamu a pokud k tomuto dojde jsou zresetovány hodnoty ukazatele seznamu a kolonky pro jméno. Dále dochází k inicializaci hodnotících prvků aplikace do základní polohy.

V aplikaci existují dvě rozdílné události typu **ODESLAT**, což jsou činnosti programu související s aktivací ovládacího prvku **ODESLAT**. Toto tlačítko aktivuje subjekt po ukončení hodnocení zkoušky. Program pak vytvoří dvourozměrné pole, které je zapsáno do **report.xls**. Jak už bylo zmíněno, události jsou dvě zejména z důvodu rozdílného výstupu testů AB a Malých zhoršení. Ukázka výstupních souborů je v poslední části této podkapitoly.

Při ukončení programu dojde k výzvě, zda uživatel chce opravdu opustit aplikaci. Toto opatření je zavedeno především pro případ, že by subjekt omylem aktivoval ovládací prvek ukončení. Vypínání VI by mělo být v režii vedoucího. Ukončovací proceduru doprovází inicializace ovládacích prvků.

7.4.6 randomization.vi

Hlavním funkčním blokem v blokovém schématu subVI je implementace **Knuthova algoritmu** [24] pomocí *loop* smyčky s posuvným registrem (obr. 7.9). Celkový počet

jméno	zkouška	datum	čas
název souboru stimulu A	název souboru stimulu B		
hodnocení A (1-5)	hodnocení B (1-5)		

Tabulka č. 11 - Ukázkový výstup testu AB 5 s referencí.

- **ABX:** Zde je důležité poznamenat, že stimul X je vždy první soubor v pořadí. Takže pokud máme od subjektu hodnocení **1** u druhého souboru, bylo označení shody nesprávné. Formát výstupu je totožný s testem **AB simple**
- **Test Malých zhoršení:**

jméno	zkouška	datum	čas
A ref/zpr?	hodnocení A	B ref/zpr?	hodnocení B

Tabulka č. 12 - Ukázkový výstup testu Malých zhoršení.
(ref - skrytá reference, zpr - zpracovaný systém)

7.5 MUSHRA test

7.5.1 Rozdíly ve vstupní fázi

Ve vstupní fázi je oproti AB testům rozdíl v dialogovém okně, které se zobrazí před vybráním vstupních souborů. Musíme si uvědomit, že pokud testujeme např. čtyři systémy (z pěti možných), vybíráme pět souborů (čtyři systémy a reference) a na předním panelu se subjektu zobrazí reference a dalších šest stimulů (čtyři systémy, skrytá reference a kotva). Formát názvů souborů je striktně vyžadován **zkouskaX_A.wav**, kde **zkouskaX_a.wav** je výhradně reference.

7.5.2 mushra.vi

Na počátku je podobným přepínačem jako u AB testu aktivován příslušný počet hodnotících ovládacích prvků. Při aktivaci *randomization.vi* dochází k uložení ukazatelů pro referenci **ref** a pro kotvu **anchor**, jak ostatně můžeme pozorovat na diagramu A.1. Přímo za chodu programu je vytvářena kotva (dolní propust s mezním kmitočtem 3500 Hz, viz obr. 7.12). Kotva je uložena v dočasném souboru. V subVI *play_mushra.vi* se rozhoduje, zda požadovaný stimul je kotva, v případě, že ano, je kotva přehrána subjektu. Při hodnocení je použita škála 0-100 dle [5].

7.5.3 Rozdíly ve výstupní fázi

Následující tabulka zobrazuje výstupní soubor MUSHRA testu při třech testovaných systémech (hod A - hodnocení stimulu A, :



Obr. 7.12: Vytvoření kotvy v mushra.vi

jméno	zkouška	datum	čas	
kotva? nebo název souborů pod stimuly A-D				
hod A	hod B	hod C	hod D	hod E

Tabulka č. 13 - Ukázkový výstup testu MUSHRA.

7.6 Rank Order test

7.6.1 Rozdíly ve vstupní fázi

Na rozdíl od AB testů, ale stejně tak jako u MUSHRA testu, je vedoucí vyzván k zadání počtu stimulů ve zkoušce. Názvy souborů opět dle předchozí definice, reference u tohoto typu testu není.

7.6.2 order.vi

Podobně jako u předchozího testu je nejprve aktivováno příslušné množství ovládacích prvků dle zadaného počtu stimulů v minulém kroku. Nejdůležitějším prvkem je programová kontrola v události **ODESLAT - Value Change**, aplikace kontroluje zda subjekt vyhodnotil všechny stimuly a jejich pořadí. Pokud tak neučinil, nebo několik stimulů ohodnotil stejně, je vyzván ke korekci. Hodnotící prvky jsou ukotveny na jednotlivých pořadích.

7.6.3 Rozdíly ve výstupní fázi

Výstupní soubor má následující formát pro čtyři stimuly:

jméno	zkouška	datum	čas
název souborů pod stimuly A-D			
hod A	hod B	hod C	hod D

Tabulka č. 14 - Ukázkový výstup testu Rank Order.

8 POUŽITÍ APLIKACE

8.1 Součásti

Zkompilovaný program se sestává z těchto součástí:

- **main.exe** – spustitelný soubor aplikace.
- **data/lvanlys.dll** – matematická knihovna.
- **data/lvsound2.dll** – knihovna pro práci se zvukovými soubory.
- **data/main.ico** – hlavní ikona programu.

8.2 Příprava souborů

V předchozí kapitole jsme narazili na problém přípravy souborů pro vlastní testy a jejich jednotlivé zkoušky. V této části diplomové práce se spíše zabýváme uživatelským přístupem k aplikaci. Všechny předešlé poznatky o formátování názvu jsme shrnuli do následující tabulky (stimuly A/B zde nevystupují jako stimuly v programu, jedná se jen o označení před náhodným rozdělením).

Typ testu	předepsané formátování?	zkouška 1	zkouška 2	význam
AB simple	nemusí	zkouska1_a.wav zkouska1_b.wav	zkouska2_a.wav zkouska2_b.wav	stimul A stimul B
AB 5/7	nemusí	zkouska1_a.wav zkouska1_b.wav	zkouska2_a.wav zkouska2_b.wav	stimul A stimul B
AB 5/7 s referencí	nemusí	zkouska1_a.wav zkouska1_b.wav zkouska1_c.wav	zkouska2_a.wav zkouska2_b.wav zkouska2_c.wav	reference stimul A stimul B
ABX	nemusí	zkouska1_a.wav zkouska1_b.wav	zkouska2_a.wav zkouska2_b.wav	stimul A stimul B
Malé zhoršení	ano musí	zkouska1_a.wav zkouska1_b.wav	zkouska2_a.wav zkouska2_b.wav	reference zhoršení
MUSHRA	ano musí	zkouska1_a.wav zkouska1_b.wav zkouska1_c.wav	zkouska2_a.wav zkouska2_b.wav zkouska2_c.wav	reference stimul A stimul B
Rank Order	ano musí	zkouska1_a.wav zkouska1_b.wav zkouska1_c.wav	zkouska2_a.wav zkouska2_b.wav zkouska2_c.wav	stimul A stimul B stimul C

Tabulka č. 15 - Vstupní soubory.

Pro další zkoušky se postupuje analogicky. Z tabulky lze snadno vyčíst, že v případech testů, které využívají referenci, je tato první v pořadí souborů. U testů, kde nemusí být splněno formátování názvu, můžeme vybírat zvukové soubory s jakýmkoliv názvy a z těchto souborů se nám vytvoří párové kombinace bez opakování. Tato volba je výhodná zejména pro urychlení přípravy a pro případy, kdy není naším cílem srovnávat jednotlivé dvojice, ale porovnání všech stimulů navzájem. Pokud test využívá referenci, je opět očekávána jako první v pořadí souborů. Tato funkce je snadno pochopitelná z této tabulky:

	AB simple	AB 5 s referencí
vstup	stimul1.wav stimul2.wav stimul3.wav stimul4.wav	stimul0.wav stimul1.wav stimul2.wav stimul3.wav
výstup	stimul1.wav stimul2.wav stimul1.wav stimul3.wav stimul1.wav stimul4.wav stimul2.wav stimul3.wav stimul2.wav stimul4.wav stimul3.wav stimul4.wav	stimul0.wav stimul1.wav stimul2.wav stimul1.wav stimul3.wav stimul2.wav stimul3.wav
počet zkoušek	6	3

Tabulka č. 16 - Vstupní soubory - kombinace.

Počet výsledných zkoušek u těchto testů lze předem vypočítat dle matematického vztahu pro kombinaci bez opakování [19]:

$$C_k(n) = \frac{n!}{k!(n-k)!}. \quad (8.1)$$

V našem případě je $k = 2$ pro dvojice a n je počet stimulů, které máme (vedoucí) nachystány pro test. Upravený výraz má tedy podobu:

$$Pocet_{zkousek} = \frac{n!}{2(n-2)!}. \quad (8.2)$$

Z časových důvodů a z důvodů možné únavy subjektů se musíme volit maximální počet stimulu ke kombinaci. Dle normy má jedno sezení trvat nejdéle dvacet

minut a uvažujeme, že subjekt bude jednu zkoušku A/B testu hodnotit (v naší aplikaci) průměrně třicet sekund. Zjišťujeme, že již pro šest stimulů je doporučená doba překročena:

$$Pocet_{zkousek} = \frac{6!}{2(6-2)!} = 48, \quad (8.3)$$

$$Doba_{testu} = Pocet_{zkousek} \cdot Doba_{zkousky} = 48 \cdot 0.5min = 24min. \quad (8.4)$$

Jednoduchými operacemi se dostáváme k závěru, že:

$$n \in \{2, 3, 4, 5\}. \quad (8.5)$$

8.3 Výstupní soubory

Výstupní soubor pro jednotlivé testy je ukládán v téže složce, ze které jsou nahrávky načítány. Toto ukládání je automatické a subjekt o této činnosti neví. V testování je možné kdykoliv pokračovat, jelikož aplikace zapisuje i do již existujícího souboru *report.xls*.

Jednotlivé ukázkové výstupy jsme ukázali v předchozí kapitole. Formát výstupu má jednu nespornou výhodu. Vzhledem k využití textu odděleného tabulátory je *report.xls* multiplatformní, tedy nezávislý na editovací aplikaci. Lze jej dokonce editovat v textovém editoru, pro plný komfort a následné zpracování dat se však doporučuje použití tabulkového procesoru.

8.4 Způsoby spuštění

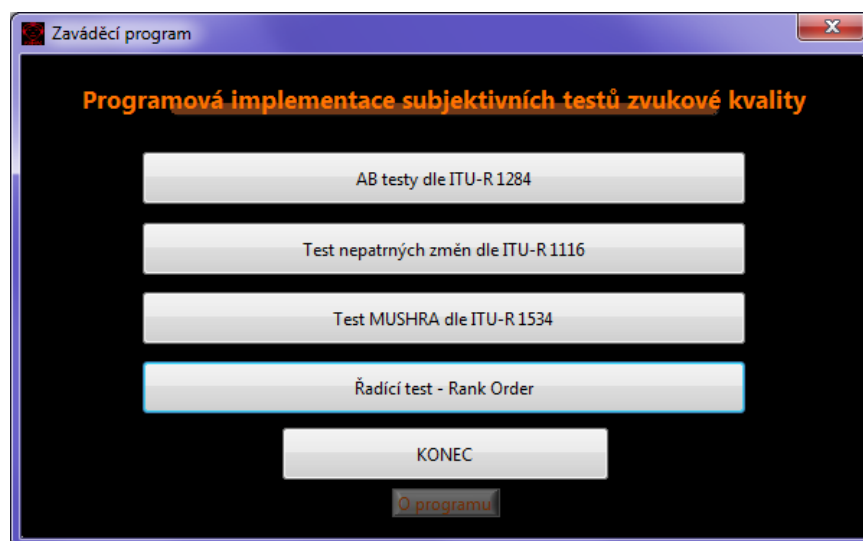
8.4.1 Zkompilovaný program

Ke spuštění programu není potřeba mít nainstalované prostředí LabVIEW, stačí méně náročné a volně šiřitelné spouštěcí prostředí **LabVIEW Runtime** [30].

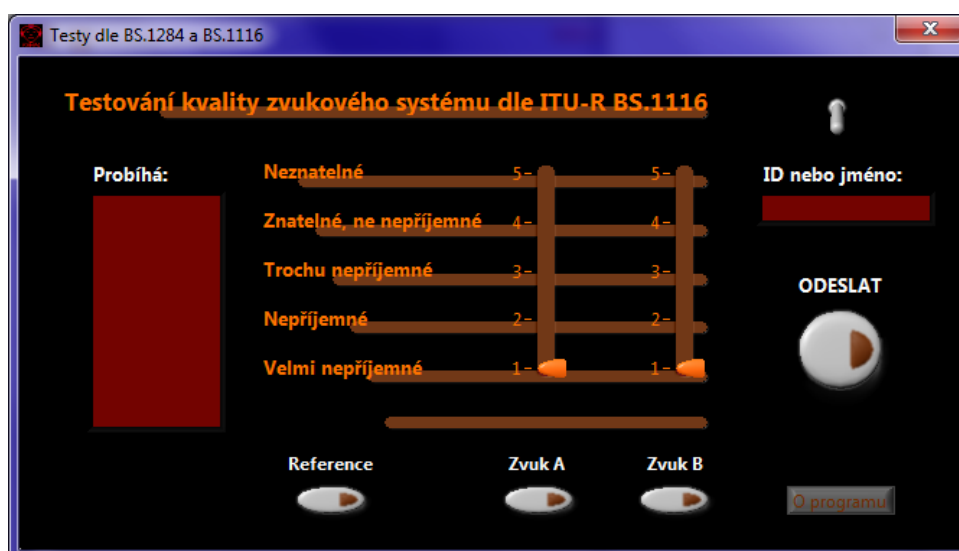
V případě uživatelské zkompilované aplikace můžeme místo předního panelu používat pojem grafické uživatelské rozhraní, zkráceně GUI.

Prvním GUI pro aktivitu vedoucího je zaváděcí program z obr. 8.1. Účelem zaváděče je pohodlný výběr vedoucího při výběru testu. V případě, že zvolí **AB testy dle ITU-R 1284** je vyzván k označení podtypu testu.

Dále vedoucí vybere pomocí průzkumníku soubory k testování. Ve Windows 7 nezáleží na pořadí výběru, ve Windows XP ano. Pokud se jedná o test MUSHRA či Rank Order zobrazí se dialogové okno pro zadání počtu testovaných systémů (resp. počet stimulů v jedné zkoušce).



Obr. 8.1: Zaváděcí program



Obr. 8.2: Typické GUI - Test malých zhoršení

Aplikace je nyní připravena pro přístup subjektů, GUI jednotlivých testů vychází z unifikovaného vzhledu (obr. 8.2). Nejprve subjekt zadá své jméno nebo ID a poté pomocí tlačítek **Reference**, **Zvuk A** a **Zvuk B**, přehrává jednotlivé stimuly a pomocí posuvných ovládacích prvků je hodnotí. Jestliže je spokojen se svým hodnocením, potvrdí ho tlačítkem **ODESLAT** a pokračuje v dalších zkouškách až do chvíle, kdy všechny splní a je vyzván k opuštění stanoviště. U **AB simple** a **ABX** jsou posuvníky neaktivní a subjekt používá dotazníkový výběr.

Aplikace se opouští pomocí vertikálního přepínače v pravém horním rohu. Ne-

úmyslné vypnutí uživatelem je ošetřeno dotázáním, zda opravdu chceme aplikaci ukončit. Poté jsou již data nachystána pro vedoucího na dříve zmiňovaném místě.

8.4.2 Originální VI

Nevýhodou tohoto způsobu je nutnost mít nainstalované prostředí LabVIEW dle požadavků [31]. LabVIEW je finančně náročné prostředí, což nás limituje v počtu testovacích stanovišť (dle licenčních podmínek).

Největší výhodou tohoto postupu je možnost zasahování přímo do kódu aplikace a jednotlivé testy si přizpůsobit dle vlastních požadavků, například na formát výstupu. Funkce jednotlivých VI je popsána v minulé kapitole a hlavní zaváděcí program se spouští pomocí *main.vi*.

8.4.3 Web server

Prostředí LabVIEW (ne zkompilevaná aplikace) nám umožňuje ovládat VI prostřednictvím sítě v prohlížečích Internet Explorer 6 a vyšší [34] a Mozilla Firefox [35]. Podmínkou je nainstalované spoštěcí prostředí [30].

Jednotlivé hlavní VI lze nakonfigurovat tak, aby byly přístupné v síti. Na serveru (PC s LabVIEW) se spustí server a na pracovní stanici (LabVIEW Runtime) se k tomuto serveru přistoupí. Celý proces konfigurace je popsán v [25]. Nutno podotknout, že koncepce této aplikace dovoluje tento spouštěcí přístup aplikovat pouze na jednu pracovní stanici (vzhledem k tomu, že soubory se zpracovávají na serveru). Použití je tedy velice specifické.

9 PRAKTICKÉ MĚŘENÍ

Praktické měření této diplomové práce má dva hlavní úkoly:

- Ověřit funkčnost a stabilitu vytvořené aplikace.
- Srovnat výsledky subjektivního testu s hodnotami *ODG* získanými objektivní metodou PEAQ.

9.1 Návrh experimentu

Návrh experimentu je od začátku ovlivněn cílem experimentu, a to srovnáním s metodou **PEAQ**. Ta objektivně srovnává zhoršené nahrávky s nahrávkou referenční na základě matematických modelů slyšení a vnímání. Její vlastnosti jsou uvedeny v kap. 1.1. Pokud budeme jako zhoršení nahrávky uvažovat přidaný šum, jehož úroveň se bude regulovat, z principů psychoakustiky lze očekávat, že při určitém vyšším odstupu signálu od šumu *SNR* nebude již zašumění slyšitelné. Zhoršení je tedy nepodstatné a *ODG* = 0. Čím větší zhoršení objektivní metoda zjistí, tím menší bude hodnota *ODG* z rozsahu $\langle -4; 0 \rangle$. *SNR* definujeme vztahem (9.1), kde *P* značí průměrné výkony signálů:

$$SNR = 10 \log_{10} \left(\frac{P_{sig}}{P_{sum}} \right) [\text{dB}]. \quad (9.1)$$

Pro srovnání volíme subjektivní metodu **Rank Order** z obecných metod zpracovaných v části 4. Subjektům bude ve zkoušce předloženo několik stimulů s odlišným *SNR*. Úkolem bude seřazení stimulů od nejslyšitelnějšího šumu (odpovídá pořadí 1) po hodnotu, kdy už šum nelze poslechem rozeznat. Všechny nahrávky se subjektivně neslyšitelným šumem budou ohodnoceny stejným pořadím (nutnost povolení *slučitelného pořadí* v aplikaci), což koreluje s nulovým *ODG*.

Ve zpracování využijeme statistickou metodu *korelační analýzy* pro vyjádření míry shody mezi metodami [20].

9.2 Realizace

9.2.1 Poslechový materiál

Nahrávky byly vybírány z veřejně dostupného materiálu doporučeného pro subjektivní testování kvality zvuku Evropskou vysílací unií – EBU [39].

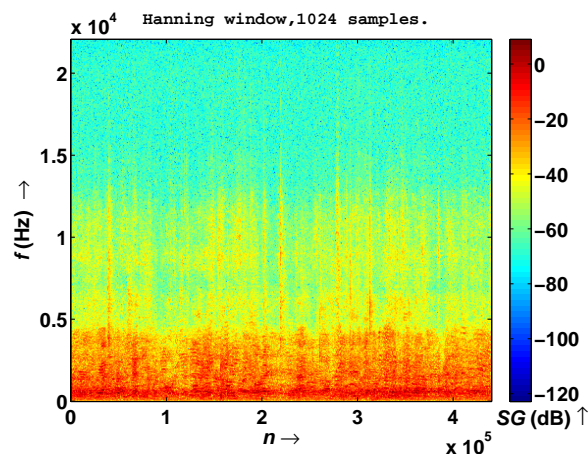
Jako referenční stimuly byly zvoleny spektrálně i poslechově odlišné nahrávky:

- Track 43 – Organ – Varhanové sólo.

- Track 50 – Male speech english – Promluva muže v angličtině.
- Track 65 – Orchestra – Úryvek ze symfonické básně Tak pravil Zarathustra.
- Track 69 – ABBA – Zástupce populární hudby.
- Track 70 – Eddie Rabbit – Country skladba.

Pro šumovou složku byly použity opět spektrálně velmi odlišné nahrávky:

- Bar Crowd – hluk konverzací v restauraci (obr. 9.1).
- Empty Casette Tape – šum při přehrávání prázdné pásky (obr. 9.2).
- Vinyl – šum s typickým praskáním vynilové desky (obr. 9.3).

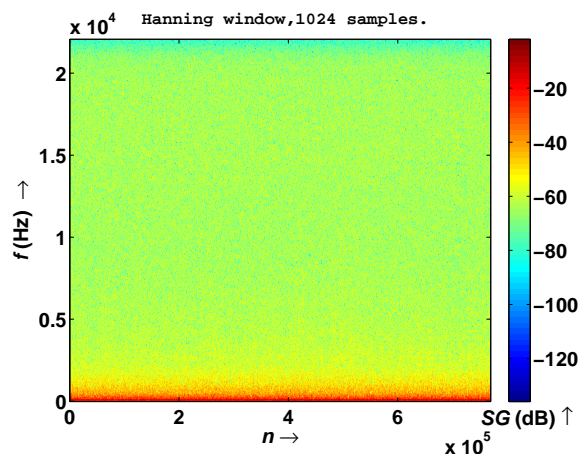


Obr. 9.1: Spektrum šumu „Bar Crowd“

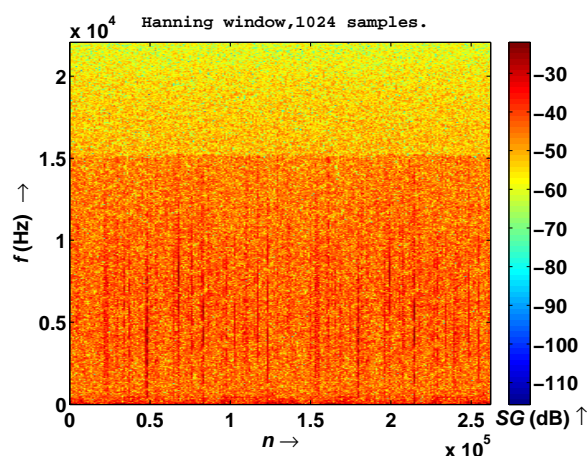
Jednotlivé referenční stimuly byly smíchány s šumy za dodržení výstupních odstupů signálu od šumu $SNR = [15, 25, 35, 40, 45, 50, 55]$ dB. Ke každé referenci bylo vygenerováno 21 stimulů (tři druhy šumu při sedmi hladinách SNR). Výsledný počet zkoušek byl tedy 15 (tři druhy šumu při pěti referencích).

9.2.2 Výběr posluchačů

Vzhledem k povaze experimentu (ověření funkčnosti) bylo rozhodnuto, že výběr posluchačů nemusí být nikterak přísný a může být ovlivněn místem měření (vysoká škola). Vzorek posluchačů tedy představuje úzkou skupinu mužů ve věku 25–30 let. Velikost vzorku je 15 subjektů. Tato hodnota nebyla zvolena náhodně. Bylo potřeba každou zkoušku ohodnotit čtyřikrát a každému subjektu byly předloženy právě čtyři zkoušky.



Obr. 9.2: Spektrum šumu „Empty Casette“



Obr. 9.3: Spektrum šumu „Vinyl“

9.2.3 Poslechové podmínky

K poslechu bylo použito šesti standardních počítačů s operačním systémem Windows XP. Každá stanice byla vybavena externí zvukovou kartou *M-Audio MobilePre 16-bit USB Sound Card* [41] opatřenou sluchátky *AKG K-66* [40]. Pro jednoduchý sběr dat byla veškerá data i samotná aplikace spouštěna ze síťového disku. Díky této centralizaci byla veškerá hodnocení přístupná v jediném souboru *report.xls* právě na tomto síťovém úložišti. Další výhodou je pak jednoduchá rozšiřitelnost na další počítače.

Specifikace sluchátek K-66:

- Frekvenční rozsah: 18 Hz – 22 kHz
- Zkreselní: Menší než 1 %

- Citlivost: 112 dB (SPL)
- Impedance: 32Ω
- Hmotnost: 210 g

9.3 Průběh experimentu

Posluchači byly rozděleny do tří skupin. Jednotlivé skupiny byly před hodnocením obeznámeny s charakterem testovaného materiálu, ovládáním aplikace a postupem hodnocení. Upřesnění a úkol byly po celou dobu napsány na tabuli tak, aby se nejistý posluchač nemusel ptát vedoucího testu a rušit své kolegy.

V průběhu celého testovacího sezení nedošlo k jakékoliv chybě ve funkčnosti aplikace a celý experiment tak proběhl bez větších problémů. Nutno podotknout, že výsledky jsou s velkou pravděpodobností zkresleny, jelikož v místnosti byl přítomen hluk pozadí způsobený ventilátorem přepínače. Bohužel přepínač byl součástí síťové infrastruktury, bez kterého by se centralizovaná distribuce aplikace a poslechového materiálu neobešla.

9.4 Zpracování dat

Vstupem pro zpracování dat je tabulka hodnocení n jednotlivých subjektů u každé zkoušky. Jedna zkouška byla ohodnocena čtyřikrát. Po úpravě získáme vstupní tabulku hodnot jediné zkoušky:

Eddie Rabbit - Bar Crowd							
SNR [dB] n [-]	sub10	zk 4	11.5.	15:40			
	15	25	35	40	45	50	55
	1	2	3	3	3	3	3
SNR [dB] n [-]	sub12	zk 3	11.5.	15:46			
	15	25	35	40	45	50	55
	1	2	3	4	4	4	4
SNR [dB] n [-]	sub13	zk 1	11.5.	15:47			
	15	25	35	40	45	50	55
	1	2	3	4	4	4	4
SNR [dB] n [-]	sub14	zk 2	11.5.	15:49			
	15	25	35	40	45	50	55
	1	2	3	3	3	3	3

Tabulka č. 17 - Tabulka vstupních hodnot pro jednu zkoušku.

Z těchto hodnot vypočítáme medián \tilde{n} hodnocení n jednotlivých SNR . Medián je hodnota, která rozdělí pozorování na dvě stejně velké skupiny. Jednoduše řečeno, máme-li řadu čísel seřazených od nejmenšího po největší, je medián prostřední hodnota. Pokud máme sudý počet hodnot je medián aritmetickým průměrem dvou prostředních hodnot. V tabulkovém procesoru **EXCEL** se pro výpočet používá funkce *MEDIAN*.

Pro srovnání s objektivním testem PEAQ, jsme použili veřejně dostupnou softwarovou implementaci [42]. Výstupní hodnoty *ODG* jsou spolu s mediány \tilde{n} zapsány v následující zjednodušené tabulce:

Eddie Rabbit - Bar Crowd							
SNR [dB]	15	25	35	40	45	50	55
n [-]	1	2	3	3	3	3	3
n [-]	1	2	3	4	4	4	4
n [-]	1	2	3	4	4	4	4
n [-]	1	2	3	3	3	3	3
\tilde{n} [-]	1	2	3	3,5	3,5	3,5	3,5
<i>ODG</i> [-]	-3	-2	-1	0	0	0	0

Tabulka č. 19 - Tabulka mediánů a *ODG* hodnot.

Na základě těchto hodnot lze vyvodit předběžný závěr. $ODG = 0$ značí neslyšitelný šum pro $SNR \geq 40dB$ odpovídá předpokladu, že subjekty hodnotili stejnou hodnotou neslyšitelné stimuly. V tomto případě je to $\tilde{n} = 3,5$ taktéž pro $SNR \geq 40dB$.

Toto tvrzení potvrdíme korelační analýzou. Analyzujeme závislost dvou proměnných \tilde{n} a *ODG* vypočítáním Pearsonova korelačního koeficientu r [20]. Korelační koeficient nabývá hodnot z intervalu $\langle -1; 1 \rangle$, přičemž hodnota -1 značí nepřímou závislost, hodnota 1 přímou závislost a hodnota 0 nezávislé veličiny. Obecný vztah (9.2) pro získání koeficientu r dvou proměnných x a y je určen takto:

$$r = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^n (x_i - \bar{x})^2 \sum_{i=1}^n (y_i - \bar{y})^2}}. \quad (9.2)$$

V tabulkovém procesoru **EXCEL** se pro výpočet používá funkce *CORREL*. Pro vzorový případ jsme vypočítali $r = 0,99$. Proměnné \tilde{n} a *ODG* jsou tedy přímo závislé. Jelikož nejsou závislé funkčně, jedná se zároveň i o tzv. volnou (stochastickou) závislost [20].

Upravená a seřazená tabulka všech naměřených i vypočítaných hodnot se nachází v příloze B.1.

9.5 Vyhodnocení

Splnili jsme oba zadané cíle. Aplikace se po dobu testovacího sezení chovala korektně. Jediná výtká směřuje k nestandardnímu ukončení programu, které bylo pravděpodobně způsobeno chybou v ovladačích nebo chybou ve spouštěcím prostředí LabVIEW pro Windows XP. V systému Windows 7 toto chování nebylo zaznamenáno. Nicméně vliv na funkci aplikace je bezvýznamný.

Po zpracování naměřených dat jsme zjistili vysokou korelaci $r > 0.9$ mezi hodnotami subjektivního testu a objektivní metody PEAQ. Za daných zhoršených poslechových podmínek lze usoudit, že byl test dobře navržen a byl zvolen vhodný poslechový materiál.

Následují doporučení pro příští testovací sezení:

- Použít jinou poslechovou místnost s lepšími akustickými vlastnostmi.
- V případě stejné poslechové místnosti decentralizovat testovací aplikaci a odstavit síťový prvek.

10 ZÁVĚR

Cílem této diplomové práce bylo na základě požadavků a diskuze zadání prostudovat doporučení týkající se subjektivních testů zvukové kvality, následně tyto poznatky implementovat v aplikaci a provést praktický poslechový experiment. Text se tak logicky dělí na čtyři části.

První část práce je zaměřena vymezení obecných poznatků týkajících se kvality zvuku a jejího měření. Kvalita zvuku je velmi subjektivní veličina závislá na hodnocení člověka. Z tohoto důvodu je velmi složité nahradit subjektivní metody objektivními, jejichž principem je náhrada poslechových testů vhodným algoritmem [2]. Mezi nejmodernější objektivní metody patří PEAQ, která do výpočtů nasazuje psychoakustické modely sluchu a také matematický model lidského vnímání [3].

V další části testu se několik kapitol zabývá subjektivními metodami popsány v doporučeních mezinárodní organizace ITU. Jedná se o **test s nepatrným zhoršením**, jeho zaměření jsou systémy s laiky nepostřehnutelnými změnami. Je nutné dodržovat striktní podmínky a využít expertních posluchačů. V praxi se tento test provádí jen na specializovaných pracovištích [4]. Méně náročná metoda je **MUSHRA**, cílená na systémy s většími změnami kvality zvuku. MUSHRA se od ostatních metod odlišuje tím, že soustřeďuje posluchačovu pozornost mezi dvě nahrávky s opačnými parametry kvality (reference a degradovaná kotva) [5]. Nejméně náročné subjektivní testy patří do skupiny **Obecných metod**. Spadají sem **AB** testy, předkládající posluchači dva podněty k posouzení, a test **Seřazování do pořadí** [9].

Ve třetím oddílu je pozornost zaměřena na programovací prostředí LabVIEW a vypracování aplikace pro testování kvality zvuku na základě poznatků z teorie. LabVIEW je prostředí využívající programovacího jazyka G [28]. Jeho hlavní výhodou je přehlednost, mezi nevýhody patří občasná složitá realizace základních úloh. Realizace aplikace je doprovázena stručným popisem zajímavých částí aplikace tak, aby ji bylo možné v budoucnu vylepšovat. Jedna kapitola je věnována použití programu v praxi.

Poslední pasáž je věnována praktickému experimentu. Cílem bylo ověření funkcí aplikace a srovnání výsledků subjektivního testu s hodnotami obecné metody PEAQ. Při návrhu experimentu byl zvolen test **seřazování do pořadí** a poslechový materiál byl založen na smíchání referenčních nahrávek s vhodným šumem. Experimentu se zúčastnilo patnáct posluchačů a po celou dobu testování nedošlo k žádnému problému s aplikací. K vyhodnocení byla zvolena statistická metoda **korelační analýzy** [20], která vykazala vysokou korelaci mezi použitou subjektivní metodou a hodnotami metody PEAQ.

Závěrem lze říci, že všechny stanovené cíle byly splněny.

LITERATURA

- [1] Smékal, Z.: *Číslíkové zpracování řeči*. BRNO, 2009. 123 s. Výuková skripta.
- [2] Bureš, Z.: *Objektivní hodnocení kvality zvuku. V Audio Technologies and Processing - ATP 2005..* BRNO, VUT, 2005. 6 s. ISBN 8021429259.
- [3] ITU-R Rec. BS.1387-1: *Method for objective measurements of perceive audio quality*. GENEVA, 2001. 100 s.
- [4] ITU-R Rec. BS.1116-1: *Methods for the subjective assesment of small impairments in audio systems including multichannel sound systems*. GENEVA, 1997. 26 s.
- [5] ITU-R Rec. BS.1534-1: *Method for the subjective assessment of intermediate quality level of coding systems*. GENEVA, 2001. 18 s.
- [6] ITU-R Rec. BS.1285: *Pre-selection methods for the subjective assessment of small impairments in audio systems*. GENEVA, 1997. 3 s.
- [7] ITU-R Rec. BS.1283-1: *A guide to ITU-R Recommendations for subjective assesment of sound quality*. GENEVA, 2003. 6 s.
- [8] ITU-R Rec. BS.1286-1: *Methods for the subjective assessment of audio systems with accompanying picture*. GENEVA, 1997. 8 s.
- [9] ITU-R Rec. BS.1284-1: *General methods for the subjective assesment of sound quality*. GENEVA, 2003. 13 s.
- [10] IEC 60268-13: *Sound System Equipment – Part 13: Listening Tests on Loudspeakers*. GENEVA, 2003. 5 s.
- [11] ITU-R Rec. BS.708: *Determination of the electro-acoustical properties of studio monitor headphones* . GENEVA, 1990. 5 s.
- [12] TSEVEENBAYAR, A.: *Vizuální programování*. PRAHA, ČVUT, 2009. 42 s. Bakalářská práce.
- [13] SUHRONE, L., TIMPLEDON, M.: *Visual programming language*. MAURITIUS, Betascript Publishing, 2010. 72 s. ISBN 6130530773.
- [14] GEHANI, N.: *Ada: concurrent programming*. NEW JERSEY, Silicon Press, 1991. 216 s. ISBN 0929306082.
- [15] VLACH, J., HAVLÍČEK, J., VLACH, M.: *Začínáme s LabVIEW*. PRAHA, BEN, 2008. 243 s. ISBN 9788073002459.

- [16] ŠTĚTINA, J., JAROŠ, M., RAMÍK, P.: *Virtuální Laboratoř - Experimentální metody*. BRNO, VUTUM, 2003. 160 s. Výuková skripta.
- [17] ŽÍDEK, J.: *Grafické programování ve vývojovém prostředí LabVIEW*. OSTRAVA, 2006. 215 s. Výuková skripta.
- [18] MELKA, A.: *Základy experimentální psychoakustiky*. PRAHA, AMU, 2005. 328 s. ISBN 9788071963653.
- [19] CALDA, E., DUPAČ, V.: *Matematika pro gymnázia Kombinatorika, pravděpodobnost, statistika*. PRAHA, Prometheus, 2008. 170 s. ISBN 8073310430.
- [20] ZVÁROVÁ, J.: *Základy statistiky pro biomedicínské obory*. PRAHA, Karolinum, 1999. 220 s. ISBN 8071847861.

Internetové články:

- [21] VARIOUS: *Vizuální programování pro automatizaci* [online], 2010, poslední aktualizace 6. 8. 2010 [cit. 15. 11. 2010]
Dostupné z URL: <<http://www.controlengcesko.com/>>.
- [22] VARIOUS: *HVAC Tool Box - Noise rating curves* [online], 2005, poslední aktualizace 16. 5. 2005 [cit. 5. 12. 2010]
Dostupné z URL: <<http://www.hvac-toolbox.com>>.
- [23] GEOFF, M.: *Introduction to Sound Recording* [online], 1999, poslední aktualizace 15. 10. 2006 [cit. 25. 3. 2011]
Dostupné z URL: <<http://www.tonmeister.ca>>.
- [24] VARIOUS: *Knuth-Morris-Pratt string matching* [online], 1996, poslední aktualizace 2. 5. 2000 [cit. 25. 3. 2011]
Dostupné z URL: <<http://www.ics.uci.edu/>>.
- [25] VARIOUS: *Simultaneously Controlling LabVIEW Remote Front Panels* [online], 2006, poslední aktualizace 6. 9. 2006 [cit. 25. 3. 2011]
Dostupné z URL: <<http://zone.ni.com/devzone/cda/tut>>.
- [26] KARPÍŠEK, Z.: *Statistické tabulky* [online], 2004 [cit. 5. 4. 2011]
Dostupné z URL: <<http://mathonline.fme.vutbr.cz>>.

WWW odkazy:

- [27] *ITU* [online]. 2010, [cit. 12. 12. 2010].
Dostupné z URL: <www.itu.int>.
- [28] *National Instruments* [online]. 2009. 2000, 12. 11. 2010 [cit. 15. 11. 2010].
Dostupné z URL: <<http://www.ni.com/>>.

- [29] *Prograph* [online]. 2004. 2004, 19. 6. 2004 [cit. 15. 11. 2010].
Dostupné z URL: <<http://www.mactech.com/PrographCPX/>>.
- [30] *LabVIEW Runtime Engine* [online]. 2010. 2010, 2. 1. 2010 [cit. 12. 12. 2010].
Dostupné z URL: <<http://joule.ni.com/nidu/cds/view/p/id/1599/lang/en>>.
- [31] *LabVIEW Requirements* [online]. 2008. 2010, 2. 6. 2010 [cit. 12. 12. 2010].
Dostupné z URL: <<http://www.ni.com/labview/requirements/>>.
- [32] *Lame enkodér* [online]. 1998. 2010, 1. 3. 2010 [cit. 12. 12. 2010].
Dostupné z URL: <<http://lame.sourceforge.net/>>.
- [33] *KOSS TD/85* [online]. 1995. 2009, 2. 8. 2009 [cit. 12. 12. 2010].
Dostupné z URL: <<http://www.koss.com>>.
- [34] *Internet Explorer* [online]. 1995. 2011, 2. 3. 2011 [cit. 25. 3. 2011].
Dostupné z URL: <<http://windows.microsoft.com/cs-CZ/>>.
- [35] *Mozilla* [online]. 2002. 2011, 12. 2. 2011 [cit. 25. 3. 2011].
Dostupné z URL: <<http://www.mozilla.org>>.
- [36] *Simulink* [online]. 1994. 2011 [cit. 18. 4. 2011].
Dostupné z URL: <<http://www.mathworks.com/products/simulink/>>.
- [37] *Matlab* [online]. 1994. 2011 [cit. 18. 4. 2011].
Dostupné z URL: <<http://www.mathworks.com/products/matlab/>>.
- [38] *CrypTool* [online]. 1998. 2011 [cit. 18. 4. 2011].
Dostupné z URL: <<http://www.cryptool.org/>>.
- [39] *Sound Quality Assessment Material recordings for subjective tests* [online]. 2008. 2011 [cit. 13. 5. 2011].
Dostupné z URL: <<http://tech.ebu.ch/publications/sqamcd>>.
- [40] *AKG – K-66* [online]. 2000. 2011 [cit. 13. 5. 2011].
Dostupné z URL: <<http://www.akg.com/>>.
- [41] *M-Audio USB Sound Card* [online]. 2004. 2005 [cit. 13. 5. 2011].
Dostupné z URL: <<http://reviews.cnet.com/external-audio-adapters/>>.
- [42] *PEAQ SW Implementation* [online]. 2002. 2008 [cit. 13. 5. 2011].
Dostupné z URL: <<http://www-mmsp.ece.mcgill.ca/>>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

ANOVA Analysis of variance – Statistická metoda analýzy

A/Č převodník Analogově-číslicový převodník – Zařízení převádějící analogové veličiny na posloupnost diskretních vzorků

EBU European Broadcasting Union – Evropská vysílací unie

GUI Graphical User Interface – Grafické uživatelské rozhraní

HTML HyperText Markup Language – Značkovací jazyk používaný na internetu

IEC International Electrotechnical Commission – Mezinárodní elektrotechnická komise

ITU International Telecommunication Union – Mezinárodní telekomunikační unie

ITU-R ITU-Recommendation – Doporučení mezinárodní telekomunikační unie

LabVIEW Laboratory Virtual Instrumentation Engineering Workbench – Laboratorní prostředí pro vývoj virtuálních přístrojů

LED Light Emitting Diode – Elektroluminiscenční dioda

Matlab Matrix Laboratory – Programové prostředí

MOV Model Output Variables – Výstupní hodnoty psychoakustického modelu

MP3 MPEG-1/MPEG-2 Audio Layer 3 – ztrátová komprese zvuku

MUSHRA MUlti Stimulus test with Hidden Reference and Anchor – Více-stimulový test se skrytou referencí a kotvou

NI National Instruments – Americká společnost

NR curve Noise Rating Curve – Křivka určující maximální hladinu akustického tlaku hluku v závislosti na kmitočtu

ODG Objective Difference Grade – Výstupní parametr některých subjektivních testů

PC Personal Computer – Osobní počítač

PCI Peripheral Component Interconnect – Sběrnice pro připojení periférií

PCM Pulse-code Modulation – Pulzně-kódová modulace

PEAQ Perceptual Evaluation of Audio Quality – Objektivní metoda měření kvality zvuku

SDG Subjective Difference Grade – Výstupní parametr některých subjektivních testů

SNR Signal to Noise Ratio – Poměr signálu k šumu

SPL Sound Pressure Level – Hladina akustického tlaku

SubVI Subroutine Virtual Instrument – Podprogram virtuálního zařízení

USB Universal Serial Bus – Univerzální sériová sběrnice

VI Virtual Instrument – Virtuální zařízení

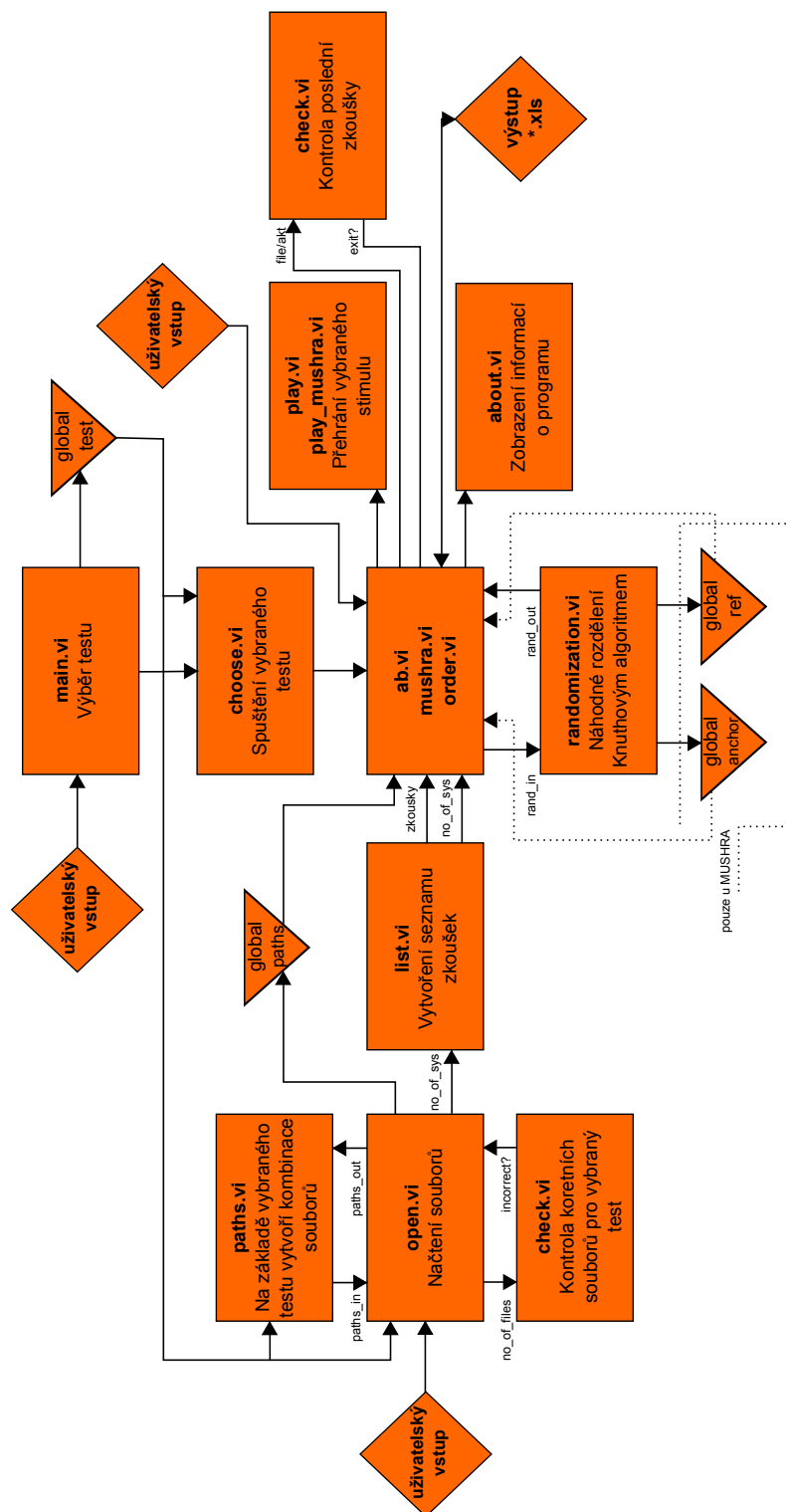
WAV/WAVE Waveform Audio File Format – Standardní formát pro ukládání zvuku

XLS Microsoft Excel spreadsheet file – Standardní formát tabulkových procesorů

SEZNAM PŘÍLOH

A Schéma běhu aplikace	67
B Tabulka naměřených hodnot	68
C Popis souborů na médiu	69

A SCHÉMA BĚHU APLIKACE



Obr. A.1: Schéma běhu aplikace.

B TABULKA NAMĚŘENÝCH HODNOT

ABBA - Bar Crowd								ABBA - Empty Cassette								ABBA - Vinyl							
sub5 zk1 11.5, 15:16								sub3 zk1 11.5, 15:14								sub3 zk2 11.5, 15:16							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	4	3	5	5	5	1	2	3	4	5	6	6		
sub1 zk2 11.5, 15:16								sub4 zk2 11.5, 15:19								sub2 zk4 11.5, 15:22							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	2	2	2	2	2	1	2	3	4	4	4	4	1	2	3	4	5	5	5		
sub6 zk3 11.5, 15:27								sub2 zk3 11.5, 15:20								sub4 zk3 11.5, 15:22							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	2	2	2	2	2	2	1	2	3	4	5	6	6		
sub15 zk4 11.5, 15:58								sub6 zk4 11.5, 15:22								sub7 zk1 11.5, 15:31							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	3	4	4	4	4	1	2	3	4	5	6	7		
ODG [-]	1	2	3	3.5	3.5	3.5	3.5	1	2	3	3.5	4	4	4	1	2	3	4	5	6	6		
r	-2	-1	-1	0	0	0	0	-3	-1	-1	0	0	0	0	-4	-4	-2	-1	-1	0	0		
0.979								0.981								0.973							

Eddie Rabbit - Bar Crowd								Eddie Rabbit - Empty Cassette								Eddie Rabbit - Vinyl							
sub10 zk4 11.5, 15:40								sub9 zk1 11.5, 15:31								sub2 zk1 11.5, 15:13							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	4	4	2	3	3	4	1	2	3	5	4	5	5		
sub12 zk3 11.5, 15:46								sub8 zk3 11.5, 15:37								sub5 zk2 11.5, 15:18							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	3	3	3	3	3	1	2	3	4	5	6	6		
sub13 zk1 11.5, 15:47								sub11 zk2 11.5, 15:38								sub1 zk1 11.5, 15:19							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	4	4	4	4	3	1	2	3	4	4	4	4		
sub14 zk2 11.5, 15:49								sub7 zk4 11.5, 15:40								sub6 zk4 11.5, 15:31							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	3	2	3	3	3	3	1	2	3	4	5	5	5		
ODG [-]	1	2	3	3.5	3.5	3.5	3.5	1	2.5	3.5	3	3	3.5	3	1	2	3	4	4.5	5	5		
r	-3	-2	-1	0	0	0	0	-2	0	0	0	0	0	0	-4	-3	-2	-1	-1	0	0		
0.980								0.940								0.980							

Male Speech - Bar Crowd								Male Speech - Empty Cassette								Male Speech - Vinyl							
sub9 zk4 11.5, 15:37								sub9 zk2 11.5, 15:33								sub6 zk1 11.5, 15:18							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	3	3	3	3	3	1	2	3	4	5	6	7		
sub10 zk3 11.5, 15:39								sub10 zk1 11.5, 15:34								sub13 zk3 11.5, 15:50							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	3	4	4	4	4	1	2	3	4	5	6	7		
sub12 zk2 11.5, 15:45								sub11 zk3 11.5, 15:40								sub14 zk4 11.5, 15:51							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	4	4	4	4	3	1	2	3	4	5	6	6		
sub14 zk1 11.5, 15:47								sub8 zk4 11.5, 15:41								sub15 zk2 11.5, 15:55							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	2	3	4	5	5	5	1	2	3	3	4	4	4		
ODG [-]	1	2	3	4	4	4	4	1	2	3	4	4	3.5	4	1	2	3	4	5	6	6.5		
r	-4	-3	-2	-1	-1	0	0	-4	-2	-1	0	0	0	0	-4	-4	-4	-3	-2	-1	-1		
0.988								0.988								0.948							

Orchestra - Bar Crowd								Orchestra - Empty Cassette								Orchestra - Vinyl							
sub12 zk4 11.5, 15:48								sub8 zk2 11.5, 15:35								sub6 zk2 11.5, 15:22							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	3	3	2	3	3	3	1	2	3	4	5	6	6		
sub13 zk2 11.5, 15:48								sub11 zk1 11.5, 15:36								sub1 zk1 11.5, 15:13							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	4	4	4	4	1	4	4	2	3	3	4	1	2	3	4	4	4	4		
sub14 zk3 11.5, 15:50								sub7 zk3 11.5, 15:36								sub13 zk4 11.5, 15:50							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	2	2	2	2	2	2	1	2	3	4	5	6	6		
sub15 zk1 11.5, 15:54								sub3 zk4 11.5, 15:24								sub15 zk3 11.5, 15:55							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	2	2	2	2	2	2	1	2	3	4	5	5	5		
ODG [-]	1	2	3	3.5	3.5	3.5	3.5	1	2.5	2.5	2	2.5	2.5	2.5	1	2	3	4	5	5.5	5.5		
r	-2	-1	0	0	0	0	0	-2	-1	0	0	0	0	0	-4	-4	-3	-2	-1	-1	0		
0.977								0.903								0.985							

Organ - Bar Crowd								Organ - Empty Cassette								Organ - Vinyl							
sub3 zk3 11.5, 15:21								sub4 zk1 11.5, 15:15								sub9 zk3 11.5, 15:35							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	2	2	2	2	2	1	2	5	4	3	5	5	1	2	3	4	6	5	7		
sub4 zk4 11.5, 15:25								sub2 zk2 11.5, 15:16								sub10 zk2 11.5, 15:38							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	2	2	2	2	2	2	1	2	3	4	5	6	7		
sub8 zk1 11.5, 15:31								sub5 zk3 11.5, 15:20								sub11 zk4 11.5, 15:41							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	3	3	3	3	3	1	2	3	3	3	3	3	1	2	4	3	5	6	6		
sub7 zk2 11.5, 15:32								sub1 zk4 11.5, 15:25								sub12 zk1 11.5, 15:44							
SNR [dB]	15	25	35	40	45	50	55	15	25	35	40	45	50	55	15	25	35	40	45	50	55		
n [-]	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1	2	3	4	5	6	6		
ODG [-]	1	2	2.5	2.5	2.5	2.5	2.5	1	2	2.5	2.5	2.5	2.5	2.5	1	2	3	4	5	6	6.5		
r	-2	-1	0	0	0	0	0	-2	-1	0	0	0	0	0	-4	-4	-3	-2	-1	-1	0		
0.988								0.993								0.985							

Obr. B.1: Tabulka hodnot

C POPIS SOUBORŮ NA MÉDIU

Na přiloženém médiu jsou (kromě elektronické verze práce) umístěny zdrojové kódy realizované aplikace. Tato příloha slouží ke snadné orientaci v nich.

V adresáři **zdrojove VI** se nachází tyto složky a soubory:

- Adresář **subvi** – obsahuje soubory:
 - **about.vi** – zobrazení informací. o programu
 - **global.vi** – veškeré globální proměnné.
 - **check.vi** – kontroly.
 - **choose.vi** – spuštění vybraného testu.
 - **list.vi** – vytvoření seznamu zkoušek.
 - **open.vi** – načtení souborů.
 - **paths.vi** - kombinace stimulů.
 - **play.vi** – přehrání a zastavení stimulů.
 - **play-mushra.vi** – přehrání a zastavení stimulů.
 - **randomization.vi** – náhodné rozdělení Knuthovým algoritmem.
- **main.vi** – výběr testu, kořenové VI.
- **main.ico** – ikona aplikace.
- **main.lvproj** – projektový soubor LabVIEW, pro kompilaci.
- **ab.vi** – hlavní VI AB testů a testu malých zhoršení.
- **mushra.vi** – hlavní VI testu MUSHRA.
- **order.vi** – hlavní VI testu Rank Order.

V adresáři **zkompilovane** se nachází tyto složky a soubory:

- Adresář **data** – obsahuje soubory:
 - **lvanlys.dll** – matematická knihovna.
 - **lvsound2.dll** – knihovna pro práci se zvukovými soubory.
 - **main.ico** – hlavní ikona programu.
- **LVRTE901min.exe** – LabVIEW runtime engine nutný pro běh.
- **main.exe** – spustitelný soubor aplikace.

V adresáři **mereni** se nachází *.WAV soubory použité při měření.