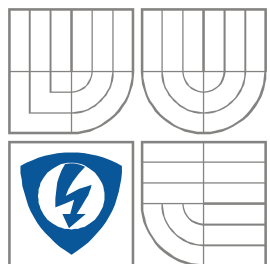




VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH  
TECHNOLOGIÍ

ÚSTAV RADIOELEKTRONIKY

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF RADIO ELECTRONICS

# D/A PŘEVODNÍK PRO AUDIO S EXTERNÍM OVLÁDÁNÍM POMOCÍ MIKROKONTROLÉRU

AUDIO D/A CONVERTER WITH EXTERNAL CONTROL USING MICROCONTROLLER

DIPLOMOVÁ PRÁCE

MASTER THESIS

AUTOR PRÁCE

AUTHOR

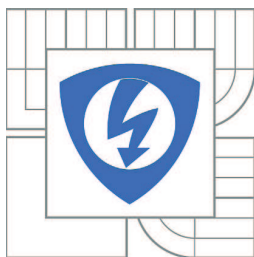
Bc. JAN ŠPAČEK

VEDOUCÍ PRÁCE

SUPERVISOR

doc. Ing. TOMÁŠ KRATOCHVÍL, Ph.D.

BRNO, 2011



VYSOKÉ UČENÍ  
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky  
a komunikačních technologií

Ústav radioelektroniky

# Diplomová práce

magisterský navazující studijní obor  
**Elektronika a sdělovací technika**

**Student:** Bc. Jan Špaček

**ID:** 106811

**Ročník:** 2

**Akademický rok:** 2011/2012

## NÁZEV TÉMATU:

**D/A převodník pro audio s externím ovládáním pomocí mikrokontroléru**

## POKYNY PRO VYPRACOVÁNÍ:

V teoretické části práce navrhnete blokové schéma D/A převodníku s výstupními audio obvody, který bude doplněn o mikroprocesorové řízení (např. ATmega), ovládací tlačítka a indikaci pracovního režimu (vzorkovací kmitočet, počet bitů pro kvantizaci apod.). Navrhnete algoritmus mikroprocesorového řízení a vývojový diagram řídicího programu. V praktické části práce navrhnete obvodové schéma převodníku včetně výstupních audio obvodů, mikroprocesorového řízení a nutných periférií. Vytvořte kompletní konstrukční podklady pro realizaci (schéma zapojení, návrh desky plošného spoje, soupisku součástek, návrh mechanického uspořádání). V experimentální části práce realizujte převodník formou funkčního prototypu a laboratorním měřením ověřte jeho správnou činnost. Ověřte správnost navrženého programu pro řízení převodníku pomocí mikrokontroléru. Výsledky měření zpracujte formou standardního protokolu o měření. Stanovte technické parametry převodníku.

## DOPORUČENÁ LITERATURA:

- [1] WIRSUM, S. Abeceda nf techniky. Praha: BEN – technická literatura, 2003.
- [2] MATOUŠEK, D. Práce s mikrokontroléry. ATMEL AVR – Atmega16. 4. díl – edice µP a praxe. Praha: BEN – technická literatura, 2006.
- [3] Analog Devices [online]. Datasheet AD1852 - Stereo, 24-Bit, 192 kHz, Multibit, Sigma-Delta DAC. Analog Devices, Inc., 2009 - [cit. 14. 1. 2011]. Dostupné na [www: http://www.analog.com/static/imported-files/data\\_sheets/AD1852.pdf](http://www.analog.com/static/imported-files/data_sheets/AD1852.pdf)

**Termín zadání:** 6.2.2012

**Termín odevzdání:** 18.5.2012

**Vedoucí práce:** doc. Ing. Tomáš Kratochvíl, Ph.D.

**Konzultanti diplomové práce:**

**prof. Dr. Ing. Zbyněk Raida**

*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 pojednává o konstrukci laboratorního přípravku s D/A převodníkem pro audio a nízkofrekvenční signály. V úvodní kapitole je nastíněn teoretický úvod do problematiky A/D a D/A převodů. V následujících kapitolách je pak již prezentováno vlastní technické řešení laboratorního přípravku, kde obvod s D/A převodníkem AD1852 je doplněn o digitální přijímač audio signálu CS8416. Ten zajišťuje kompatibilitu připojení k externím audio zařízením disponujícím digitálními rozhraními, jako jsou např. SPDIF nebo AES3. Digitální přijímač i audio D/A převodník jsou navíc opatřeny sériovým komunikačním rozhraním, pomocí kterého je možné nastavovat většinu parametrů vstupního audio signálu. Toto nastavení je prováděno pomocí navrženého ovládacího softwaru pro externí mikrokontrolér řady ATmega32. Práce obsahuje blokový a obvodový návrh přípravku a to včetně technické a konstrukční dokumentace celého zařízení.

## **KLÍČOVÁ SLOVA**

D/A převodník, digitální přijímač, mikrokontrolér, audio, řízení, převod, signál

## **ABSTRACT**

This diploma thesis deals with the design of a laboratory measurement tool with ADC converter for audio and low frequency signals. In the introduction part a theoretical background of the ADC and DAC conversions are outlined. In the next chapters the technical solution of the laboratory converter is presented, including DAC with integrated circuit AD1852 and completed with digital audio receiver CS8416. This circuit secures compatibility with external audio devices with digital interfaces, like SPDIF or AES3. The digital receiver and audio DAC are added with serial communication interface for setup of the most audio and input parameters. This setup is operated by external microcontroller ATmega32 and original firmware. The thesis contains block diagram and schematic layout of the laboratory converter, including complete technical documentation for the construction and design.

## **KEYWORDS**

D/A converter, digital receiver, microcontroller, audio, control, conversion, signal

ŠPAČEK, J. *D/A převodník pro audio s externím ovládním pomocí mikrokontroléru*.  
Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních  
technologií. Ústav radioelektroniky, 2011. 30 s., 31 s. příloh. Semestrální práce.  
Vedoucí práce: doc. Ing. Tomáš Kratochvíl, Ph.D.

## PROHLÁŠENÍ

Prohlašuji, že svou diplomovou práci na téma D/A převodník pro audio s externím ovládáním pomocí mikrokontroléru 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/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne .....

.....

(podpis autora)

## PODĚKOVÁNÍ

Děkuji vedoucímu diplomové práce doc. Ing. Tomáši Kratochvílovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé semestrální práce.

Dále děkuji mému kolegovi a spolužákovi, Bc. Radomíru Strakovi za zapůjčení vývojového kitu pro mikrokontrolér ATmega32 a také za drobnou pomoc a spolupráci při zpracování dílčí části mé semestrální práce.

V Brně dne .....

.....

(podpis autora)

# OBSAH

|   |             |
|---|-------------|
| <b>Seznam obrázků</b>   | <b>viii</b> |
| <b>Seznam tabulek</b>   | <b>ix</b>   |
| <b>Úvod</b>   | <b>1</b>    |
| <b>1 Teoretický úvod do problematiky</b>                        | <b>2</b>    |
| 1.1 Princip A/D převodu .....                                   | 2           |
| 1.2 Princip D/A převodu .....                                   | 4           |
| 1.3 Základní princip Sigma-delta převodu .....                  | 4           |
| <b>2 Vlastní návrh technického řešení</b>                       | <b>6</b>    |
| 2.1 Obecný popis systému .....                                  | 6           |
| 2.2 Blokové schéma zapojení .....                               | 7           |
| <b>3 Konstrukční řešení projektu</b>                            | <b>8</b>    |
| 3.1 Volba D/A převodníku.....                                   | 8           |
| 3.2 Volba digitálního přijímače .....                           | 9           |
| 3.3 Volba řídicího mikrokontroléru a dalších periférií .....    | 10          |
| 3.3.1 Volba mikrokontroléru .....                               | 10          |
| 3.3.2 Volba LCD displeje .....                                  | 10          |
| 3.3.3 Volba výstupních audio obvodů .....                       | 10          |
| 3.4 Obvodové schéma zapojení .....                              | 11          |
| 3.4.1 Deska A – schéma zapojení .....                           | 12          |
| 3.4.2 Deska B – schéma zapojení .....                           | 13          |
| 3.4.3 Deska C + Deska D – schéma zapojení .....                 | 14          |
| 3.5 Důležité poznatky ke konstrukční části projektu .....       | 14          |
| 3.5.1 Deska A – digitální část .....                            | 14          |
| 3.5.2 Deska A – analogová část .....                            | 16          |
| 3.5.3 Deska A – rozdělení zemí .....                            | 17          |
| 3.5.4 Deska A – napájecí okruhy .....                           | 18          |
| 3.5.5 Deska A, deska B – vstupně-výstupní řídicí rozhraní ..... | 19          |
| 3.5.6 Deska B.....  | 20          |

|          |   |           |
|----------|---|-----------|
| <b>4</b> | <b>Návrh řídicího programu</b>                              | <b>22</b> |
| 4.1      | Řízení D/A převodníku AD1852 .....                          | 22        |
| 4.1.1    | První způsob (Hardwarový).....                              | 22        |
| 4.1.2    | Druhý způsob (Softwarový).....                              | 22        |
| 4.2      | Řízení digitálního přijímače CS8416.....                    | 23        |
| 4.2.1    | První způsob (Hardwarový).....                              | 23        |
| 4.2.2    | Druhý způsob (Softwarový).....                              | 23        |
| 4.3      | Vývojový diagram řídicího programu .....                    | 24        |
| 4.4      | Důležité funkce pro řízení periférií.....                   | 26        |
| 4.4.1    | Ovládací knihovna pro D/A převodník AD1852 .....            | 26        |
| 4.4.2    | Ovládací knihovna pro Digitální audio přijímač CS8416 ..... | 26        |
| 4.4.3    | Popis hlavního řídicího programu.....                       | 26        |
| 4.4.4    | Soupis hlaviček funkcí.....                                 | 27        |
| <b>5</b> | <b>Závěr</b>  | <b>28</b> |
|          | <b>Literatura</b>   | <b>29</b> |
|          | <b>Seznam příloh</b>  | <b>30</b> |



# SEZNAM OBRÁZKŮ

|   |    |
|---|----|
| Obr. 1.1: K tématu vzorkování – příklad vzorkovače s pamětí, detailní průběh napětí při vzorkování analogového signálu a ukázka bitových hladin pro 3-bitový digitální signál (převzato z [1])..... | 3  |
| Obr. 1.2: A/D převodník s modulátorem Sigma-Delta [3] .....   | 5  |
| Obr. 2.1: Blokové schéma obvodu s D/A převodníkem .....   | 7  |
| Obr. 3.1: Blokové schéma integrovaného obvodu AD1852 (převzato z [4]) .....   | 8  |
| Obr. 3.2: Zapojení pinů D/A převodníku (převzato z [4]) .....   | 9  |
| Obr. 3.3: Blokové schéma digitálního přijímače CS8416 (převzato z [5]).....   | 9  |
| Obr. 3.4: Obvodové schéma zapojení desky A .....  | 12 |
| Obr. 3.5: Obvodové schéma zapojení desky B .....  | 13 |
| Obr. 3.6: Schéma zapojení desky s napájecím zdrojem a desky s tlačítky.....   | 14 |
| Obr. 3.7: Způsob spojení digitální a analogové země s ochranou proti napěťové špičce .....  | 18 |
| Obr. 3.8: Zapojení pinů v napájecím konektoru.....  | 19 |
| Obr. 4.1: Vývojový diagram řídicího programu .....  | 25 |

## SEZNAM TABULEK

|  |    |
|--|----|
| Tab. 3.1: Zapojení pinů v napájecím konektoru.....                 | 19 |
| Tab. 3.2: Zapojení pinů konektoru vstupně-výstupního rozhraní..... | 20 |

# ÚVOD

Jako téma pro tuto diplomovou práci byla zvolena konstrukce obvodu s digitálně-analogovým (tedy D/A) převodníkem pro audio aplikace. V této oblasti se v dnešní době nejčastěji používají moderní, tzv. Sigma-delta převodníky. Jedná se o typ převodníků, které používají metody několikanásobného převzorkování, kdy je použito vzorkovací frekvence několikanásobně vyšší než je pro dané vzorkování potřeba a dále je pro převod do binární formy použit jednobitový kvantizér. Výhodou těchto převodníků je mj. i schopnost dosažení daleko větší dynamiky signálu (menšího šumu), než klasické D/A převodníky (s tzv. Nyquistovým vzorkováním). Více o této problematice bude zmíněno v následující kapitole (popř. dalších kapitolách) této práce.

Protože se jedná o převodníky tzv. multistandardní, je možno měnit podmínky příjmu i zpracování vstupního digitálního audio signálu anebo (častěji) též vlastnosti či parametry výstupního analogového audio signálu, v závislosti na daných kritériích použití převodníku, případně na rozhodnutí uživatele. Zjednodušeně řečeno, nabízí se zde možnost nastavit převodník (pomocí předem daných postupů) takovým způsobem, že může zpracovávat digitální audio signál různé kvality či o různých parametrech, např. pro různé vzorkovací frekvence či bitové hloubky, též je možno na tento signál aplikovat různé filtry, je možné řídit jeho hlasitost na výstupu a další funkce. Kromě návrhu a konstrukce samotného D/A převodníku s jeho nezbytnými perifériemi se bude tato práce zabývat též možnostmi jeho řízení.

První kapitola této práce bude tedy obecně pojednávat o analogově-digitálních (A/D) a pak hlavně o digitálně-analogových (D/A) převodech a převodnících, pozornost bude zaměřena zejména na výše zmíněné Sigma-delta převodníky. Také bude nastíněna oblast použití tohoto typu převodníků. V druhé kapitole bude předveden vlastní návrh provedení konkrétního technického řešení, kterým se tato práce zabývá. Bude zde uvedeno základní blokové schéma se stručným popisem celkové funkce a také informace, jaký konkrétní D/A převodník a další nutné periférie k jeho správnému chodu budou použity. V následující kapitole budou předloženy konkrétní obvodová schémata pro jednotlivé funkční bloky. Čtvrtá kapitola se bude zabývat softwarovým řízením zvoleného D/A převodníku. Budou zde uvedeny potřebné vývojové diagramy a v příloze samozřejmě také zdrojové kódy pro řízení pomocí zvoleného mikrokontroléru. V dalších kapitolách (nebo v přílohách) budou případně připojeny další náležitosti a užitečné informace k tomuto projektu.

# 1 TEORETICKÝ ÚVOD DO PROBLEMATIKY

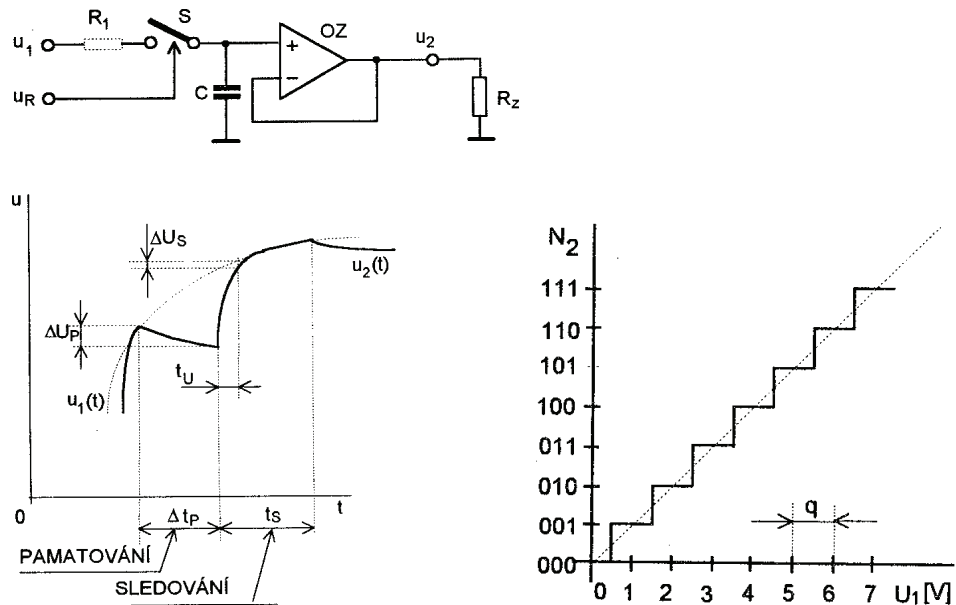
Následujících několik podkapitol bude zaměřeno velice stručně na problematiku A/D a D/A převodů. Dále budou zmíněny některé typy převodníků, které se používají a nakonec v závěru této kapitoly bude věnována pozornost sigma-delta převodníkům, jejichž model bude použit pro konstrukci výše zmíněného obvodu, jakožto hlavní náplně této diplomové práce.

## 1.1 Princip A/D převodu

Série technologických pokroků ve 20. století vedly k postupnému přechodu záznamu a uchování informací z analogové na digitální formu, jelikož začala postupně převažovat relativně citelná nevýhoda pro analogová data, což bylo poměrně komplikované uchování a skladování analogových záznamů a také jejich neustálé udržování v dobré kvalitě. Je třeba mít též na paměti, že každý analogový záznam, uložený na nějakém standardním používaném médiu, jehož přehrávání bylo zpravidla vždy alespoň z části řešeno mechanickou cestou (např. dotykem s nahranou stopou), se s přibývajícím počtem přehrávek stával postupně méně kvalitním a čistým, naopak více zašuměným, zarušeným nebo jinak znehodnoceným. Proto bylo do budoucna potřeba data (a to nejen zvuková) uchovávat takovým způsobem, aby jejich interpretace v „surovém“ formátu byla pokud možno co nejjednodušší. A tak se zavedl záznam v tzv. binární formě, tedy používající pouze dva stavy, a to logickou 0 a logickou 1. Tyto stavy se dají interpretovat různým způsobem (např. průběh napětí v čase dosahuje pouze hodnot 0 V a +5 V, průběh napětí je určitým způsobem vyvoláván z přehrávaného média – v dnešní době je velmi známý a rozšířený princip záznamu na optické nosiče). Určitá posloupnost těchto dvou stavů (dnes se nazývají nejčastěji jako bity) může definovat binární číslo, někdy nazývané jako symbol. Poté, často po převodu tohoto symbolu např. do dekadické soustavy, toto číslo nějakým způsobem interpretuje určitou hodnotu nějakého signálu, většinou hodnotu dané vstupní (nebo naopak výstupní) proměnné v daném čase, prostoru či jiné nezávislé veličině.

Protože binární data ze zásady nemohou být čistě spojitá, jako je tomu u analogových veličin, je potřeba je nějakým způsobem diskretizovat, tedy určit přesný počet jeho jednotlivých datových slov, tzv. vzorků. Tyto vzorky jsou odvozeny původně z analogového signálu a následně jsou diskretizovány jak v čase, tak i ve vlastní hodnotě (tedy přesně řečeno, signál je vzorkován a kvantován – vždy se jedná o konečný počet stavů). Pro diskretizaci analogového signálu v čase slouží tzv. vzorkovače. Tyto (z principu velmi jednoduché) obvody mají za úkol analogový signál navzorkovat, a to tak, že si vzorkovač detekuje hodnotu okamžité analogové veličiny na svém vstupu v přesně definovaném čase a tuto hodnotu si zapamatuje po nezbytně dlouhou dobu převodu pomocí paměťového obvodu (v nejjednodušších případech lze použít kapacitor o vhodné hodnotě kapacity, případně v sériovém spojení s napěťovým sledovačem, pro menší impedanční ovlivnění kapacitoru). Dále je signál kvantován, tzn., že se dané vzorky porovnávají s několika rozhodovacími úrovněmi napětí, které

mají svoje přesné úrovně a následně se těmito vzorkům přiřazuje přesně daná amplitudová hodnota (podle rozhodovacích úrovní, mezi kterými se navzorkovaný signál pohybuje). Výstupní signál tedy poté nabývá již pouze „několika“ přesně definovaných hladin v přesně definovaných časových okamžicích. Celou situaci vzorkování, zapamatování a interpretace digitálního signálu dokresluje obrázek 1.1.



Obr. 1.1: K tématu vzorkování – příklad vzorkovače s pamětí, detailní průběh napětí při vzorkování analogového signálu a ukázka bitových hladin pro 3-bitový digitální signál (převzato z [1])

Vzorkování signálu, které probíhá s určitou frekvencí vždy jistým způsobem souvisí s šířkou pásma vzorkovaného signálu. O tomto problému pojednává tzv. Vzorkovací teorém, známý též jako Shannon-Kotělnikovův teorém či Nyquistův teorém. Ve své podstatě říká, že frekvence vzorkování musí být vždy dvojnásobná, než maximální frekvence vzorkovaného signálu.

$$F_{VZ} \geq 2 \cdot f_{\max} \quad (1)$$

Kde  $F_{VZ}$  je frekvence vzorkovacího signálu a  $f_{\max}$  je maximální frekvence vzorkovaného signálu.

Dále se nabízí možnost každou z těchto zmiňovaných hladin určitým způsobem zakódovat – tedy popsat je nějakým definovaným binárním číslem (sekvencí bitů o několika prvcích) a takto zaznamenávat na dané záznamové médium, případně přenášet signál pomocí přenosové trasy, pouze pomocí stavů log. 0 a log. 1. Tímto celým procesem byla popsána funkce A/D převodníku, resp. A/D převodu.

## 1.2 Princip D/A převodu

Signál, který je nyní v digitální podobě, je potřeba při přehrávání, nebo na konci přenosové trasy, opět převést do analogové podoby. K tomu slouží tzv. D/A převodník. Tento by měl být konstruován tak, aby v daném napěťovém (popř. proudovém) rozsahu dokázal pokud možno co nejlépe převést digitální signál zpět na soubor amplitud (tedy „diskrétních“ veličin) a posléze jej zrekonstruovat na výsledný (spojitý a hladký) analogový signál. D/A převodník tedy převádí vstupní digitální veličinu na výstupní veličinu analogovou, a to podle jednoduchého vztahu (převzat z [2]):

$$u_v = M \cdot d \quad (2)$$

Kde  $u_v$  je výstupní analogová veličina (v tomto případě napětí),  $M$  značí míru převodu a  $d$  je číselný ekvivalent vstupního kódového slova.

D/A převodníky jsou nejčastěji koncipovány jako soustava váhovaných zdrojů s řízenými spínači, ovládanými signály jednotlivých bitů v digitálním slově. Každý bit má svoji přidělenou váhu, která je závislá na pozici v daném slově (dva koncové z nich jsou MSB – „most significant bit“, tedy nejvýznamnější bit (bit s největší vahou) a LSB – „least significant bit“, tedy nejméně významný bit (bit s nejmenší vahou)). Obdobným způsobem jsou váhy interpretovány i váhovanými zdroji v podobě výstupní veličiny (nejčastěji napětí nebo proudu). Při samotném převodu se každý zdroj (např. napětí) aktivuje (při log. 1) nebo naopak deaktivuje (při log. 0), přičemž po sečtení napětí ze všech váhovaných zdrojů napětí dostaneme celkovou hodnotu výstupního signálu.

Zde je potřeba připomenout a zdůraznit, že se jedná o velmi zjednodušené popisy postupů při vzájemných převodech mezi analogovou a digitální formou signálu. Oba tyto procesy jsou však mnohdy daleko složitější a je zapotřebí uvažovat v daném případě daleko více rozhodujících faktorů. Více o této problematice se zabývá literatura [2].

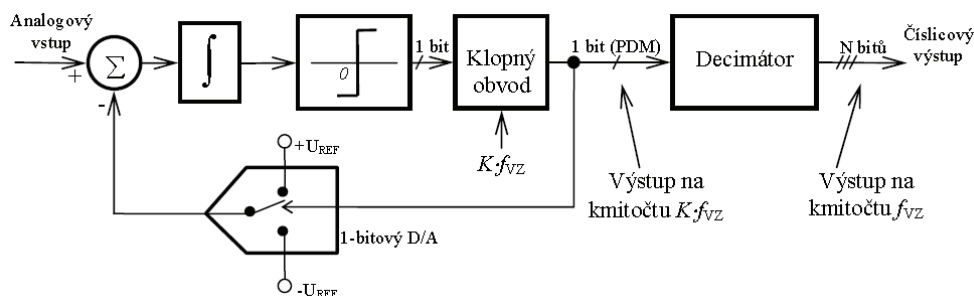
Analogově-digitální a digitálně-analogové převodníky v dnešní době mají velmi široké uplatnění v mnoha odvětvích. Jedním z nich jsou i aplikace pro zpracování zvuku. Mezi nejvíce rozšířené patří A/D a D/A převodníky paralelní, potom také sériové. V audio aplikacích však dnes nacházejí největšího uplatnění převodníky zvané Sigma-delta.

## 1.3 Základní princip Sigma-delta převodu

Základní myšlenka Sigma-delta převodníku vychází z myšlenky modulace delta. Tato myšlenka je založená na modulaci (nebo přenosu) pouze rozdílu mezi dvěma vzorky (rozdíl tedy označený symbolem  $\Delta$  – tedy delta). Modulátory delta obsahují integrátor, jednobitový kvantizér a rozdílový člen. Předpokládejme, že vstupní signál je rostoucí kladný, při sejmutí prvního vzorku se na výstupu kvantizéru objeví kladné napětí, které by mělo odpovídat rozdílu vstupního signálu a nuly, tento signál je však interpretován např. hodnotou napájecího napětí log. 1. Při sejmutí druhého vzorku by měl být na výstupu kvantizéru opět rozdíl druhého (právě sejmutého) vzorku a prvního vzorku.

Pokud se jedná o jednobitový kvantizér, na jeho výstupu bude opět signál interpretující log. 1. Při klesajícím signálu může být na výstupu kvantizéru např. signál reprezentující log. 0. Modulátor tak vlastně vytváří průběh signálu pomocí jednotlivých skoků o velikosti  $\Delta$ . [3]

Delta modulátor se pro modulaci delta často doplňuje o integrátor, který se nachází ve zpětné vazbě modulátoru. Pro demodulaci je potřeba použít na „přijímací“ straně další integrační člen. Matematicky je však dokázáno, že pokud tento integrační člen je přesunut do přímé větve za součtový člen zpětné vazby (často označován symbolem  $\Sigma$  – Sigma), výsledek se nezmění. Tímto způsobem lze principiálně dosáhnout modulace Sigma-delta ( $\Sigma - \Delta$ ), z níž vychází základní myšlenka pro Sigma-delta převodníky. Více o této problematice a celý princip převodníků lze dále najít v lit. [3]. Následující obrázek znázorňuje příklad blokového zapojení A/D převodníku se Sigma-Delta modulátorem.



Obr. 1.2: A/D převodník s modulátorem Sigma-Delta [3]

## 2 VLASTNÍ NÁVRH TECHNICKÉHO ŘEŠENÍ

V této kapitole je uvedena vlastní představa technického řešení celého obvodu s D/A převodníkem, jehož návrh, konstrukce a následné oživení tvoří hlavní náplň této diplomové práce. Jedná se o D/A převodník multistandardní, tedy je možno (jak již bylo zmíněno v předchozím textu) tento převodník nastavovat tak, aby zpracovával vstupní i výstupní signál na základě nastavení uživatele. Řídící část pro nastavování převodníku a dalších nutných periférií pomocí mikrokontroléru bude zmíněna v další kapitole.

### 2.1 Obecný popis systému

Klíčovou úlohu v celém systému tvoří D/A převodník. Tento přijímá digitální audio data ve formátu I<sup>2</sup>S (třívodičové rozhraní pro posílání audio dat, obsahuje jeden digitální audio signál a dva pomocné hodinové signály) a na základě nastavení svých vnitřních obvodů převádí digitální audio signál do odpovídající analogové podoby. Protože formát I<sup>2</sup>S není příliš rozšířený jako formát pro posílání digitálního zvukového signálu mezi jednotlivými zařízeními (spíše pro posílání digitálního zvukového signálu mezi vnitřními perifériemi jednoho zařízení), bude nejspíš nutné připojit před D/A převodník ještě digitální audio přijímač, který na svých vstupech podporuje daleko rozšířenější digitální rozhraní pro posílání audio signálu mezi zařízeními (jako např. AES3, S/PDIF rozhraní apod.).

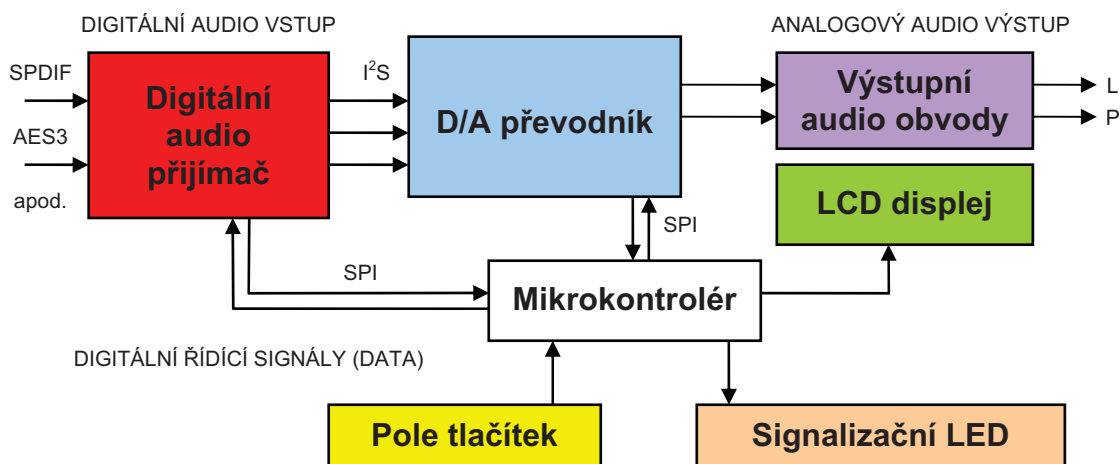
D/A převodník bude nastavován, stejným způsobem jako digitální audio přijímač, pomocí mikrokontroléru (např. ATmega16, ATmega32...), který bude zároveň sloužit jako ústřední člen pro veškeré řízení, a dále s přidáním LCD displeje a několika ovládacích tlačítek (případně několika signalizačních diod), bude mít celý systém implementován jednoduché uživatelské rozhraní s ovládacím MENU (zobrazeným na displeji), pomocí kterého bude uživatel moci ovládat jednotlivá nastavení všech dostupných parametrů jak D/A převodníku, tak digitálního přijímače.

Také se zde nabízí možnost, že mikrokontrolér pomocí detekce signálů s různými příznaky (z digitálního přijímače) automaticky nastaví potřebné parametry zpracování signálu pro D/A převodník.



## 2.2 Blokové schéma zapojení

Na následujícím obrázku je vyobrazeno blokové schéma celého zapojení. Více podrobností k jeho jednotlivým blokům a řídicí části bude řečeno v dalších kapitolách.



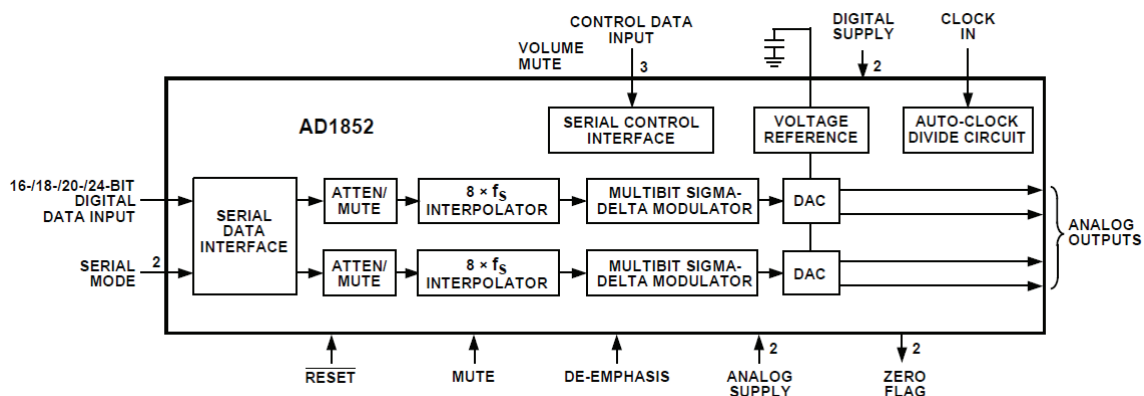
Obr. 2.1: Blokové schéma obvodu s D/A převodníkem

### 3 KONSTRUKČNÍ ŘEŠENÍ PROJEKTU

Minulá kapitola obsahující obecný popis systému, doplněný blokovým schématem, měla za úkol lépe a obecně osvětlit všechny možnosti a funkce, které bude hotový funkční výrobek poskytovat. Tato kapitola konkrétně obsahuje poznámky ke konstrukční stránce projektu (tj. návrhy vhodných integrovaných obvodů, obvodová schémata atd.).

#### 3.1 Volba D/A převodníku

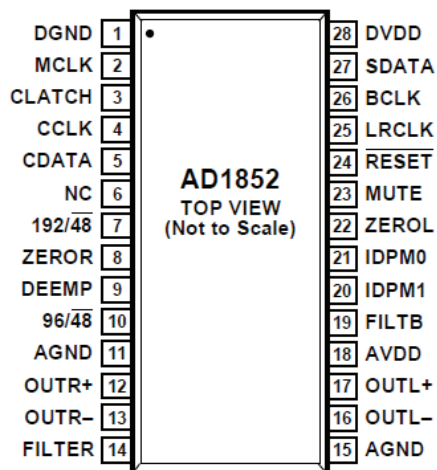
Jako D/A převodník, aplikovaný v navrhovaném obvodu, byl vybrán obvod AD1852 od firmy Analog Devices. Jedná se o integrovaný obvod napájený zdrojem napětí +5 V. Podporuje vstupní digitální audio data o vzorkovacích frekvencích 32 kHz, 44.1 kHz, 48 kHz, 88.2 kHz, 96 kHz a 192 kHz a to ve formátu 16 bitů, 20 bitů a 24 bitů. Převodník má dynamiku signálu kolem 117 dB (podle katalogového listu, reálná hodnota závisí na mnoha parametrech celého návrhu a obvykle bývá nižší) a je vybaven interní digitální deemfází pro vzorkovací kmitočty 32 kHz, 44.1 kHz a 48 kHz. Níže je vyobrazeno blokové schéma obvodu AD1852. [4]



Obr. 3.1: Blokové schéma integrovaného obvodu AD1852 (převzato z [4])

Pro úplnost informací je níže přiložen ještě obrázek se zapojením pinů zvoleného D/A převodníku. Při bližším prostudování katalogového listu k tomuto obvodu lze snadno zjistit, že některé z funkčních pinů nám umožní ovládat funkce jako Nastavení vzorkovací frekvence, Zarovnání bitů digitálního signálu, Bitová hloubka signálu, Utlumení signálu, atd. O způsobu řízení těchto funkcí (nabízí se konkrétně dva různé způsoby) bude řečeno v další kapitole.

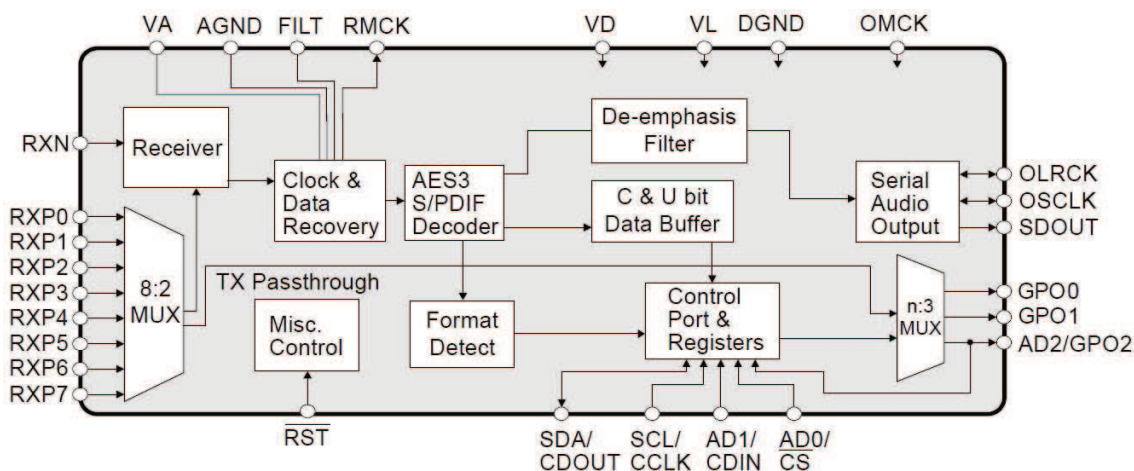
Další skupinou pinů (která již neplní funkci řízení samotného převodníku) jsou piny pro přenos digitálního signálu na vstup (konkrétně piny SDATA, BCLK, LRCLK) a piny pro přenos analogového signálu z výstupu (jsou to OUT L+, OUT L-, OUT R+, OUT R-), dále jsou to napájecí piny, pin pro přivedení hlavního hodinového signálu pro D/A převodník (MCLK), nebo příznakové piny informující o nulovém či chybějícím signálu na vstupech D/A převodníku pro každý kanál zvlášť (ZEROL a ZEROR). [4]



Obr. 3.2: Zapojení pinů D/A převodníku (převzato z [4])

### 3.2 Volba digitálního přijímače

Jako vhodný digitální přijímač, který byl zvolen pro tuto aplikaci, je obvod od Cirrus Logic, a to model CS8416. Jedná se o integrovaný obvod digitálního přijímače audio signálu, který podporuje stejné zvukové formáty jako výše zvolený obvod AD1852 (maximální vzorkovací frekvence  $f_{vz} = 192$  kHz, maximální bitová hloubka  $N_b = 24$  bitů). Na datových výstupech obvodu CS8416 budou odebrány potřebné signály formátu I<sup>2</sup>S, které již lze posílat přímo do datových vstupů D/A převodníku. Na obrázku níže bude uvedeno blokové schéma tohoto obvodu. [5]



Obr. 3.3: Blokové schéma digitálního přijímače CS8416 (převzato z [5])

Podobným způsobem jako u obvodu AD1852, i zde se nabízí z principu dvě možnosti řízení tohoto obvodu. Na základě toho, do jakého módu řízení je obvod nastaven (podle katalogového listu se může jednat o tzv. Hardwarový nebo Softwarový mód), mají jednotlivé piny různé řídicí nebo informační funkce. Další podrobnosti o těchto způsobech řízení budou opět detailněji rozebrány v další kapitole, která se bude zabývat řízením všech periférií a také návrhy řídicích algoritmů. [5]

Z obrázku výše lze dále zjistit, že piny RXP0-7 jsou vstupní signálové piny pro zdroje digitálního audio signálu, pin RXN je společná zem pro piny RXP (nebo negativní vstup pro symetrický vstupní digitální signál), OMCK je vstupní pin pro hodinový signál digitálního přijímače, RMCK je pin hlavního hodinového signálu pro D/A převodník a dále výstup audio dat ve formátu I<sup>2</sup>S (pro D/A převodník) probíhá prostřednictvím pinů OLRCK, OSCLK a SDOUT. [5]

### **3.3 Volba řídicího mikrokontroléru a dalších periférií**

#### **3.3.1 Volba mikrokontroléru**

Jako řídicí prvek celého obvodu s digitálním přijímačem a D/A převodníkem byl zvolen mikrokontrolér od firmy ATMEL, a sice model ATmega32. Jedná se o mikrokontrolér v pouzdře DIL40. V úvahu by též připadal i model ATmega16 nebo podobný typ mikrokontroléru z této rodiny. Kritérii pro bezchybné fungování jsou zde pouze dostatečný počet vstupně-výstupních portů, dostatečně velká programová paměť (tohle kritérium je v drtivé většině případů splněno) a podpora sériového komunikačního rozhraní SPI.

#### **3.3.2 Volba LCD displeje**

LCD displej je možno vybírat z velké škály typů i výrobců. Pro naše účely je vhodný displej podporující především zobrazování textu (jako je např. typ displeje s řadičem HD44780 od firmy Hitachi). Pro účel projektu byl vybrán 4-řádkový LCD displej bez podsvětlení. Výběr displeje se odvíjí od uživatelského prostředí hlavního ovládacího menu k řídicímu programu běžícím v mikrokontroléru ATmega32.

#### **3.3.3 Volba výstupních audio obvodů**

Jak již bylo zmíněno v zadání a v předchozím textu, v obvodu s D/A převodníkem nemohou být opomenuty výstupní audio obvody. Tyto obvody jsou připojeny k analogovým (linkovým) audio výstupům D/A převodníku AD1852. Jedná se o zapojení antialiasingových filtrů a předzesilovacích obvodů tvořených operačními zesilovači AD797 a dalšími diskrétními součástkami. Tyto výstupní obvody navíc plní funkce zesílení výstupního signálu D/A převodníku a převodu symetrického signálu na nesymetrický. Celá situace bude názorněji viditelná z obvodového schématu zapojení v další kapitole.

### 3.4 Obvodové schéma zapojení

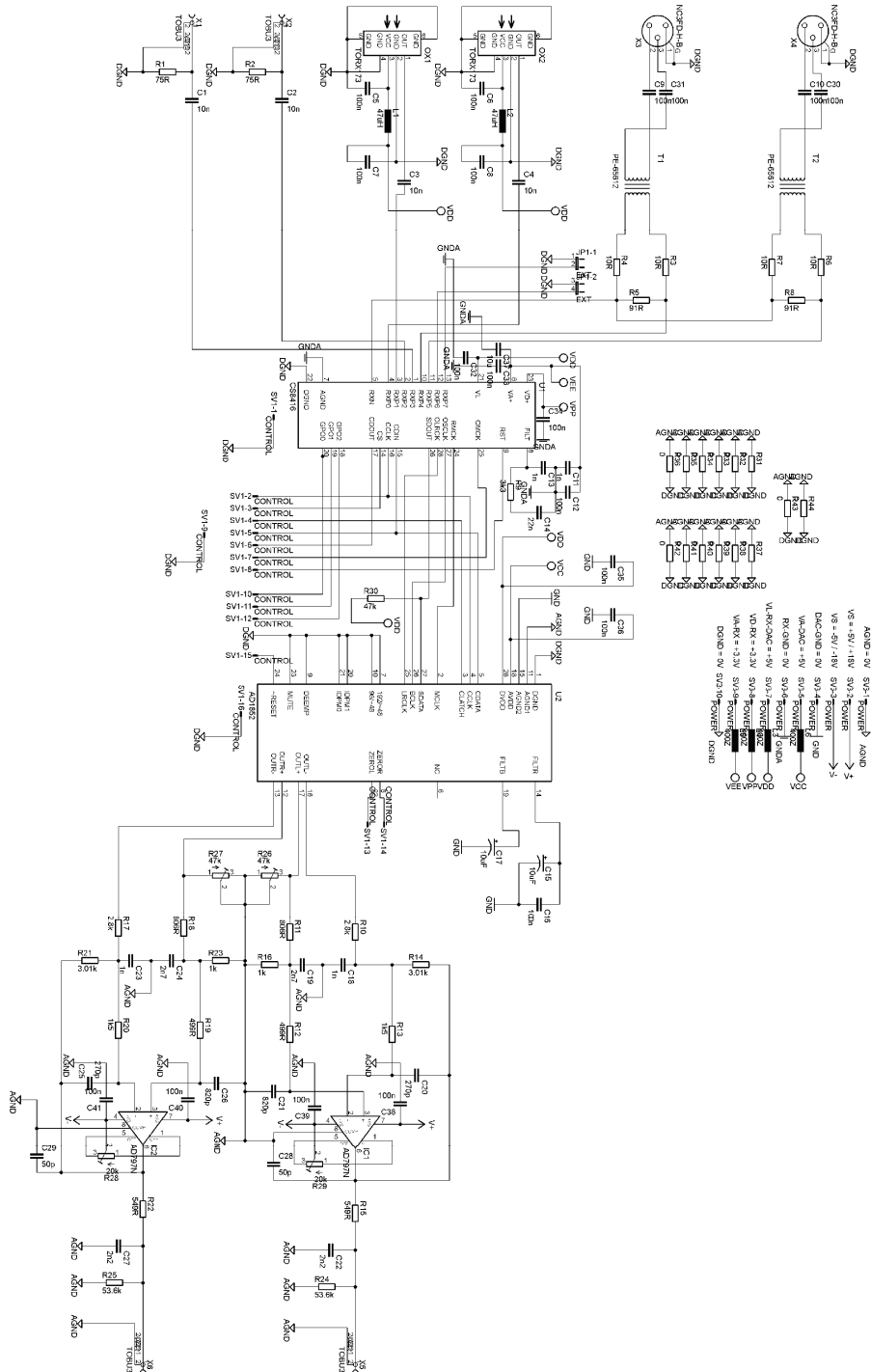
Jak již bylo řečeno výše, následující podkapitoly obsahují schémata zapojení celého obvodu. Návrhy desek plošných spojů budou umístěny do příloh. Každá podkapitola bude zahrnovat obvodové schéma jednotlivých částí obvodu, návrhy desek plošných spojů (DPS) a soupisky použitých součástek budou umístěny v příloze.

Z důvodů lepší robustnosti bude celé zapojení rozděleno na několik desek plošných spojů. Stěžejní část celého zapojení bude tvořit hlavní DPS, pracovně označená jako Deska A. Tato bude zahrnovat integrovaný obvod CS8416 digitálního přijímače, integrovaný obvod AD1852 s D/A převodníkem, dále audio vstupní a výstupní konektory, konektor sloužící k propojení s řídicí deskou a také konektor pro napájení desky. Obvod hlavní desky je koncipován tak, že digitální přijímač bude moci vybírat až z osmi signálových vstupů, z toho šest z nich je vyvedeno na desce v různých rozhraních, a to jak SPDIF koaxiální – na konektorech CINCH (někdy označovaný též TOBU3), SPDIF optický, na konektorech TORX173 (v některých systémech nazývaný též jako TOSLINK), tak AES3 symetrický digitální signál, na konektorech XLR. Další dva signálové vstupy nejsou vyvedeny na konektorech, ale jsou vyvedeny na jumperu JP1 nacházející se na hlavní desce. V případě, kdy nejsou používány, jsou tyto jumpery spojkou přemostěny a vstupy jsou tím uzemněny. Jinak, v případě potřeby, je možno tyto kontakty rozpojit a připojit k obvodu CS8416 prostřednictvím pinů tohoto jumperu další dva libovolné digitální vstupy.

Obvod s mikrokontrolérem ATmega32, je umístěn na druhé, samostatnou desku, tedy řídicí DPS, též pracovně označovanou jako Deska B. Obsahuje mikrokontrolér s jeho nezbytnými externími součástkami, resetovacím tlačítkem pro případnou potřebu hardwarového resetu mikrokontroléru, konektory pro připojení tlačítek a displeje, propojovací 16-pinový konektor s deskou A a také programovací konektor pro ISP programátor, který slouží k naprogramování ATmegy na desce B. K této desce (nebo zcela mimo ni) poté může být mechanicky připevněn panel s displejem a ovládacími tlačítky, jejichž čelní části budou umístěny na vrchní straně šasi hotového prototypu.

### 3.4.1 Deska A – schéma zapojení

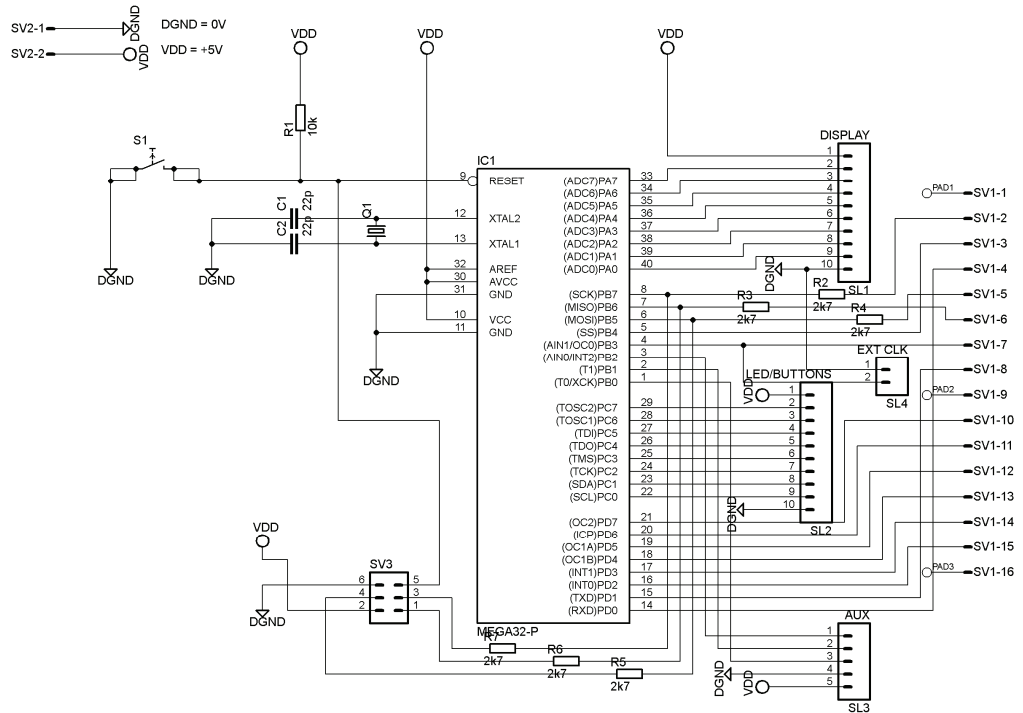
Následující obrázek představuje obvodové schéma zapojení hlavní DPS.



Obr. 3.4: Obvodové schéma zapojení desky A

### 3.4.2 Deska B – schéma zapojení

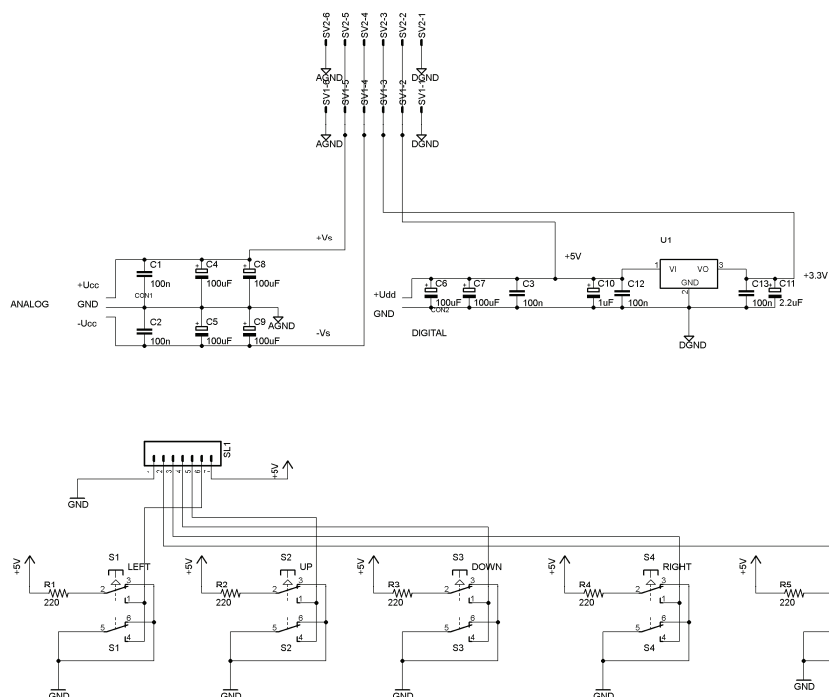
Na následujícím obrázku je vyobrazeno obvodové schéma zapojení řídicí DPS.



Obr. 3.5: Obvodové schéma zapojení desky B

### 3.4.3 Deska C + Deska D – schéma zapojení

Následující schéma obsahuje návrh desky s tlačítky a taky desky s napájecím zdrojem. Z důvodů úspory místa a času při výrobě DPS byly tyto dvě desky vyráběny v jednom kroku na stejném kusu kuprextitu.



Obr. 3.6: Schéma zapojení desky s napájecím zdrojem a desky s tlačítky

## 3.5 Důležité poznatky ke konstrukční části projektu

Aby bylo dosaženo co nejlepší funkčnosti celého prototypu, je potřeba dodržet několik pravidel nebo doporučení při jeho návrhu, zpravidla se jedná o informace a doporučení z katalogových listů. Také by bylo v rámci této kapitoly vhodné doplnit několik doplňkových informací k návrhu, které nejsou při pohledu do schémat na první pohled zřejmé. Zaměříme se na každou desku plošných spojů zvlášť, u desky A též zvlášť na část digitální i analogovou.

### 3.5.1 Deska A – digitální část

Prvním a hlavním kritériem v oblasti návrhu zapojení s D/A převodníkem je striktní oddělení analogové a digitální části obvodu. Vždy je potřeba umístit na DPS všechny pasivní i aktivní prvky obvodu takovým způsobem, aby mohla být na DPS jasně vytyčena část digitální i analogová. Ruku v ruce s tímto krokem je potřeba také pečlivě rozdělit společnou signálovou zem zvlášť na analogovou a digitální, dále je také vhodné oddělit země napájecí pro oba integrované obvody (digitální přijímač, D/A převodník). Zamezí se tím pronikání rušení do signálové země.

Konektory vstupních signálů jsou připojeny k digitální signálové zemi. Jak již bylo



řečeno, dva digitální vstupy jsou koaxiální, další dva vstupy jsou optické (oba pro rozhraní SPDIF) a poslední dva vstupy jsou symetrické (pro rozhraní AES3). Koaxiální vstupy SPDIF jsou opatřeny konektory CINCH. Symetrické vstupy AES3 jsou opatřeny konektory XLR. Signál z těchto konektorů je doporučeno galvanicky oddělit od výstupu zdroje signálu. To lze provést pomocí dvou způsobů: Pomocí dvojice kapacitorů a přídavného rezistoru, nebo pomocí digitálních oddělovacích transformátorů. Pro tento projekt bylo zvoleno kvalitnější provedení galvanického oddělení pomocí digitálních oddělovacích transformátorů (konkrétně se jedná o typové označení PE65612NL od firmy Pulse Transformers Audio & Signal). Optické vstupy SPDIF jsou opatřeny konektory TORX173, což jsou vlastně přijímací moduly optického signálu. Tyto konektory jsou napájeny z digitálního zdroje napětím +5 V a podle doporučení katalogového listu jsou dále blokovány kondenzátory s kapacitou 100 nF a chráněny proti napěťovým špičkám pomocí induktorů 47  $\mu$ H. Navíc konektor je uvnitř vybaven polovodičovými částmi, tudíž je potřeba si počínat obezřetně, hlavně při pájení. [6] Bohužel, tato součástka nespĺňuje certifikát RoHS a v současné době postupně začíná být z trhu stahována a nahrazována jinými (často kvalitnějšími) typy přijímacích modulů.

Digitální přijímač CS8416 je možno nastavit buď do hardwarového módu (tzn. řízení funkčními piny) nebo do softwarového módu (řízení pomocí komunikačního rozhraní). Výběru jednoho z těchto dvou módů docílíme následovně: připojením pull-down rezistoru na zem z pinu SDOUT pro řízení hardwarové, nebo připojením pull-up rezistoru na napájecí napětí z pinu SDOUT pro řízení softwarové (jako v případě tohoto návrhu). Přijímač dále obsahuje interní smyčku fázového závěsu PLL, jejímž úkolem je rekonstruovat hodinový signál z přijatého digitálního audio signálu. Tato smyčka fázového závěsu je napájena zvláštní větví analogového napájení (v katalogu má příslušný pin pro přívod napájení označení VA) a je velmi náchylná na změny napájecího napětí, proto potřebuje poměrně stabilní zdroj napětí s co nejmenším zvlněním a minimalizovaným vnějším rušením. Obvod digitálního přijímače též obsahuje pin pro připojení externího filtračního RC členu pro lepší stabilitu fázového závěsu. Tento filtrační člen musí být velmi dobře navržen, hlavně blokovací kapacitor musí být umístěn co nejbližší tělu součástky digitálního přijímače. Hodinový signál, který je takto pomocí smyčky fázového závěsu obnoven, je v katalogových listech označován jako hlavní hodinový signál (master clock) a slouží pro odvození správné vzorkovací frekvence a správné bitové synchronizace přijímaného digitálního audio signálu. Tento hodinový signál pak řídí správné časování vnitřních obvodů digitálního přijímače a současně je posílán prostřednictvím pinu s označením RMCK na vstup master clock D/A převodníku, kde plní stejnou funkci. Pokud má uživatel možnost přivést externí synchronizaci k digitálnímu audio signálu, např. z generátoru, který touto možností disponuje, je možno tuto synchronizaci přivést na vstup hodinového signálu digitálního přijímače na vstupní pin s označením OCMK. V nastavení kontrolního registru přijímače lze poté smyčku PLL snadno vypnout a vstup OCMK se v podstatě kopíruje na výstupní pin RCMK. V případě potřeby je možno také tento hodinový signál přivést na děličku kmitočtu a podělit jeho frekvenci na polovinu. Pokud potřebujeme vyčítat stavové příznaky o různých událostech probíhajících s digitálním signálem v přijímači (např. chybějící signál na vstupu, detekce signálu s vzorkovací frekvencí větší než 48 kHz, chyba v přijetí signálu apod.), můžeme použít příznakových bitů GPO (GPO2, GPO1, GPO0) nebo lze využít kontrolního registru přístupného pomocí SPI rozhraní. Samotný digitální audio signál, který je přijímaný z libovolného

vstupu z rozhraní SPDIF nebo AES3, je již v podobě dekódovaného signálu I<sup>2</sup>S odebrán z výstupních pinů OSCLK (výstupní bitový hodinový signál), OLRCLK (výstupní hodinový signál pro levý a pravý kanál) a SDOUT (výstupní sériová audio data). [5]

D/A převodník AD1852, na rozdíl od digitálního přijímače, má všechny své funkční piny, které nemění svoji funkci v závislosti na řízení hardwarovém, či softwarovém. Funkční piny, které nastavují potřebné parametry D/A převodníku, v případě, kdy je používáno 3-vodičové komunikační rozhraní SPI, se jednoduše mohou uzemnit na digitální signálovou zem a nebudou se používat. Katalogový list uvádí, že funkční piny jsou k registrům 3-vodičového rozhraní SPI v logickém součtu (neboli v logické funkci OR). Takže pokud bude spolehlivě zajištěno, že ve stejný čas nebudou používány SPI komunikace i funkční piny zároveň, lze ponechat i funkční piny aktivní a obvod ovládat také hardwarově. Ovšem je třeba podotknout, že hardwarové ovládání nedokáže obsáhnout všechny parametry funkcí, které lze měnit pomocí registrů. Digitální audio signál, který je poslán z digitálního přijímače ve formátu I<sup>2</sup>S, přijímá D/A převodník pomocí pinů označených BCLK (vstupní bitový hodinový signál), LRCLK (vstupní hodinový signál pro levý a pravý kanál) a SDATA (výstupní sériová audio data). Obvod D/A převodníku disponuje také oddělenými napájeními pro analogovou i digitální část. Pro obě části je napětí +5 V. Na desce A jsou obě tyto napájecí větve vedeny zvlášť. [4]

### 3.5.2 Deska A – analogová část

Analogové části desky A je potřeba při návrhu věnovat podobnou péči jako části digitální. Protože D/A převodník zahrnuje jak část digitální, tak analogovou, bude o něm zmínka ještě v této podkapitole. Analogová část desky dále obsahuje také výstupní obvody a konektory CINCH analogového audio linkového výstupu.

D/A převodník, tentokrát z analogové části, je doporučeno napájet i na analogové straně velmi dobře vyhlazeným a stabilním napětím +5 V. Protože obvod disponuje dvěma piny s označením AGND, vyplývá z toho, že jedna analogová zem je určena pro napájení a druhá pro připojení signálové země. Obvod D/A převodníku také potřebuje analogovou napěťovou referenci, která musí být dostatečně stabilní a s časem neměnná. Proto i zde je potřeba k určeným pinům obvodu připojit externí součástky pro dodatečnou filtraci tohoto napětí (tyto jsou připojeny na napájecí zem). Výstupní piny pro analogový audio signál jsou symetrické. Piny OUTL+ a OUTL- jsou výstupy pro levý kanál (s pozitivní a negativní polaritou), stejně tak piny OUTR+ a OUTR-, což jsou stejné výstupy pro pravý kanál. Měděné spoje vedoucí diferenční páry signálů je vhodné navrhnut tak, aby obě cesty vedoucí signály byly těsně u sebe a kopírovaly pokud možno co nejvíce stejnou dráhu (pokud je to možné). Plocha mezi smyčkami tak bude menší a zamezí se tím vzniku napěťové resp. proudové nesymetrie. V případě obvodu AD1852, který disponuje proudovými výstupy, budeme mluvit o nesymetrii proudové. Tuto nesymetrii jsme schopni eliminovat pomocí SMD trimrů R26 a R27 s označením OPT na straně BOTTOM. Jejich hodnota byla zvolena jako 47 kΩ. Případné rušení, které by se do obou diferenčních párů indukovalo (za předpokladu, že oba signály budou zarušeny stejným rušivým signálem či brumem), by mělo být vzájemně odečteno, po součtu obou signálů z diferenčního páru a inverzi polarity jednoho z nich. Docílí se tím dvojnásobné amplitudy užitečného signálu a naopak velmi

kvalitního potlačení rušivých signálů. Jedná se o analogii k přenosu symetrických signálů, např. z dynamických mikrofonů. [4]

Další zpracování analogového symetrického signálu poskytují velice kvalitní, nízkošumové operační zesilovače AD797. Tyto mají za úkol převést signál z proudového do napěťového módu a sečíst signály s opačnými polaritami z obou kanálů. Dále tyto operační zesilovače mají za úkol impedančně přizpůsobit proudové výstupy D/A převodníku fyzickému (napěťovému) linkovému výstupu, zajistit antialiasingovou filtraci (pomocí rekonstrukčního filtru, který je tvořen pomocí těchto operačních zesilovačů a několika dalších diskrétních součástek kolem operačního zesilovače) a samozřejmě poskytnout dostatečné předzesílení pro přívod na fyzický audio výstup celého obvodu. Hodnoty externích diskrétních součástek (nejčastěji rezistorů a kapacitorů) u operačního zesilovače AD797, tvořících komplexní zapojení analogového výstupního obvodu, nepatří ke standardním hodnotám série E12, ale odpovídají sérii E48. Jedná se ovšem o hodnoty doporučené na základě výsledků simulace firmy Analog Devices. Není nezbytně nutné tyto hodnoty dodržovat, ovšem při jejich zvolení (série E48 až E96) máme zajištěnu vysokou pravděpodobnost správné funkčnosti, ideálně rovné kmitočtové charakteristiky v audio pásmu, minimálního šumu a minimálního zkreslení výstupního signálu. V našem projektu byly z technických důvodů použity hodnoty externích součástek série E12 a E24 (a to vždy nejbližší možná hodnota). Dále, napájecí napětí pro operační zesilovače AD797 je symetrické, může se pohybovat mezi  $\pm 5$  V až  $\pm 18$  V, v tomto případě je možno hodnotu napájení zvolit např.  $\pm 9$  V. SMD trimry R28 a R29 na straně TOP kompenzují proudovou nesymetrii výstupního signálu v nule. [4], [7]

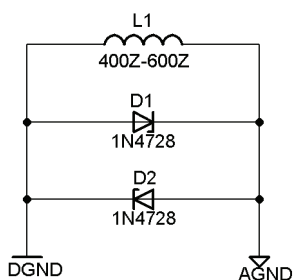
### 3.5.3 Deska A – rozdělení zemí

Jak již bylo řečeno v předchozích podkapitolách, v tomto obvodu je třeba dbát na striktní rozdělení zemí. Při pohledu shora na stranu TOP, na levé části desky (kde se nachází digitální vstupy signálu) je rozlita zem digitální a na pravé části desky (kde jsou umístěny analogové výstupy audio signálu) je rozlita zem analogová. Jak je vidět z návrhu DPS, celá deska je poměrně hustě pokryta měděnými prokovy umístěnými na obou rozlitych zemích (jak digitální – DGND, tak analogové – AGND), jejichž cílem je kvalitně přizemnit všechny zemnicí plochy a snížit tak jejich zemní odpor na obou stranách desky, TOP i BOTTOM. To zajišťuje např. dobré vzájemné odstínění všech možných digitálních i analogových signálů mezi sebou.

Kromě signálových zemí (digitální a analogové) jsou na desce pomocí samostatných cest doplněny ještě země napájecí: Zvlášť je vedena zem k digitálnímu přijímači (který je napájen z digitální napájecí větve +3,3 V) a zvlášť k D/A převodníku (jež je napájen z digitální napájecí větve +5 V). Operační zesilovače nejsou přímo vybaveny pinem pro připojení napájecí země, každopádně země symetrického napájecího zdroje pro tyto operační zesilovače musí být galvanicky spojená s analogovou zemí. Navíc pin č. 8 tohoto operačního zesilovače slouží jako přípojný bod zpětné vazby pro kompenzaci zkreslení výstupního signálu. Část výstupního signálu je zde přes kapacitor 50 pF odváděna na tento zpětnovazební pin, který je zároveň spojen se signálovou zemí.

Dále je nutno dodat, že v závěrečné části konstrukce, je nutno, podle doporučení výrobců digitálních obvodů, spojit země DGND a AGND v jednom místě, nejlépe pod

pouzdrém D/A převodníku, či v jeho bezprostřední blízkosti. V praxi je ovšem taky možné, že obvod nemusí správně fungovat nebo bude vykazovat nepříliš kvalitní parametry výstupního signálu. Důvodem může být mj. i špatné umístění propojovacího bodu obou zemí. Z tohoto důvodu bylo do návrhu přidáno několik nulových propojek (resp. rezistorů s nulovou hodnotou ( $R = 0 \Omega$ )). Jedná se o rezistory R31 až R44 (v obvodovém schématu zapojení umístěny nahoře vedle napájecího konektoru). Všechny tyto rezistory s hodnotou „0“ pouze připravují plošky na potenciálně vhodných místech, kde by bylo možno tyto dvě signálové země propojit. V žádném případě není vhodné osadit všechny tyto pozice nulovými propojkami. Podle doporučení v katalogu D/A převodníku AD1852 od Analog Devices je také vhodné místo použití klasické nulové propojky při spojení zemí použít feritový kroužek (který lze v našem případě nahradit SMD tlumivkou vhodné hodnoty indukčnosti), případně dále pro ochranu před napěťovou špičkou (která by mohla vzniknout při zapnutí všech zdrojů mezi oběma zeměmi) lze tuto SMD tlumivku doplnit antiparalelním zapojením rychlých Schottkyho diod. Tyto diody zamezí vzrůstu napěťové špičky na vyšší hodnotu napětí, než 0,65 V. Tak zůstanou aktivní prvky celého obvodu řádně chráněny. Typ Schottkyho diod lze použít libovolný. Následující obrázek naznačuje popisované doporučené zapojení.



Obr. 3.7: Způsob spojení digitální a analogové země s ochranou proti napěťové špičce

V následující podkapitole budou popsány používané napájecí okruhy pro všechny aktivní prvky v obvodu.

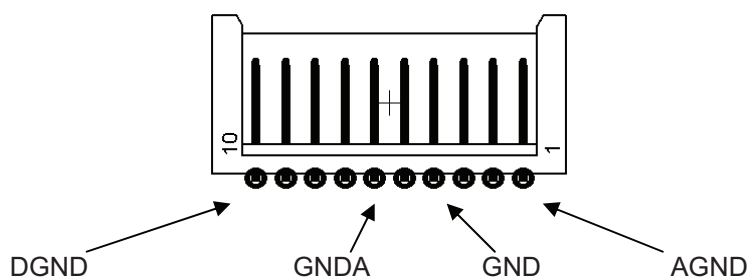
### 3.5.4 Deska A – napájecí okruhy

Jak již bylo naznačeno v minulých kapitolách, budou zde popsány druhy napájecích napětí, které jsou potřebné pro napájení všech aktivních prvků a také bude definován napájecí konektor desky A.

- **+3,3 V:** Jedná se o napájecí napětí pro digitální přijímač. Tento druh napětí je připojen k pinům digitálního přijímače s označením VA (analogové napájecí napětí, určené k napájení PLL) a VD (digitální napájecí napětí, určené k napájení dalších vnitřních obvodů přijímače).
- **+5 V:** Jedná se o napájecí napětí primárně určeno pro D/A převodník. Napájení je přivedeno na piny s označením DVDD (digitální napájení pro vnitřní digitální obvody převodníku) a AVDD (analogové napájení pro analogovou část a referenci převodníku). Dále je tohle napětí přivedeno na pin digitálního přijímače s označením VL (napěťová úroveň logické 1 v zapojení) a také jako napájení ke konektorům TORX173 pro příjem optického SPDIF signálu.

- **+/- 5 V až +/- 18 V:** Jedná se o symetrické napájecí napětí určeno pro operační zesilovače v analogové části obvodu. Napětí pro operační zesilovače může být z tohoto intervalu zvoleno takřka libovolně. Zpravidla platí, čím větší je napájecí napětí nastaveno, tím většího zesílení výstupního signálu lze dosáhnout.

Na následujícím obrázku a přiložené tabulce níže je přehledně znázorněno číslování pinů konektoru a popis, na kterém pinu se nachází který druh napětí:



Obr. 3.8: Zapojení pinů v napájecím konektoru

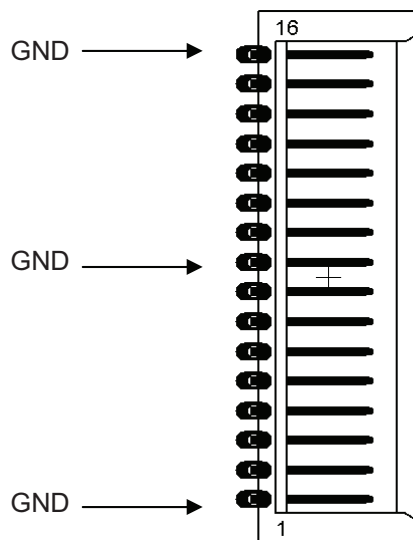
Tab. 3.1: Zapojení pinů v napájecím konektoru

| Č. pinu | Označení ve schematu | Hodnota napětí | Popis signálu                         |
|---------|----------------------|----------------|---------------------------------------|
| 1       | AGND                 |                | Analogová signálová zem               |
| 2       | +VS / V+             | +5 V až +18 V  | Pozitivní větev symetrického zdroje   |
| 3       | -VS / V-             | -5 V až -18 V  | Negativní větev symetrického zdroje   |
| 4       | DAC-GND / GND        |                | Napájecí zem pro D/A převodník        |
| 5       | VA-DAC / VCC         | +5 V           | Analogové napájení pro D/A převodník  |
| 6       | RX-GND / GNDA        |                | Napájecí zem pro digitální přijímač   |
| 7       | VL-RX-DAC / VDD      | +5 V           | Digitální napájecí větev obvodů       |
| 8       | VD-RX / VPP          | +3,3 V         | Napájecí větev pro digitální přijímač |
| 9       | VA-RX / VEE          | +3,3 V         | Napájecí větev pro PLL dig. přijímače |
| 10      | DGND                 |                | Digitální signálová zem               |

### 3.5.5 Deska A, deska B – vstupně-výstupní řídicí rozhraní

Obdobným způsobem jako v minulé podkapitole, kde byly popsány piny napájecího konektoru, tato podkapitola popisuje piny konektoru vstupně-výstupního řídicího rozhraní, který slouží k propojení hlavní desky A s řídicí deskou B. Na následujícím obrázku bude vyobrazen propojovací konektor a v tabulce níže bude naznačeno, jaké fyzické piny digitálního přijímače a D/A převodníku jsou spojeny s jakými piny řídicího mikrokontroléru (ATmegy32).





Tab. 3.2: Zapojení pinů konektoru vstupně-výstupního rozhraní

| Č. pinu | Ozn. pinu Dig. př. a DAC | Ozn. pinu ATmega32 | Popis funkce   |
|---------|--------------------------|--------------------|--|
| 1       | DGND                     | GND                | Zemnicí pin  |
| 2       | CCLK/MCLK                | SCK / PB7          | D.př. , D/A přev. – hodinový signál pro SPI rozhraní |
| 3       | CS                       | SS / PB4           | D.př. – výběrový signál pro SPI rozhraní             |
| 4       | CLATCH                   | RXD / PD0          | D/A přev. – výběrový signál pro SPI rozhraní         |
| 5       | CDIN/CDATA               | MOSI / PB5         | D.př. , D/A přev. – výstupní data SPI rozhraní       |
| 6       | CDOUT                    | MISO / PB6         | D.př. – vstupní data SPI rozhraní                    |
| 7       | OCMK                     | OC0 / PB3          | D.př. – hodinový signál pro audio digitální rozhraní |
| 8       | RST                      | TXD / PD1          | D.př. – resetovací pin                               |
| 9       | DGND                     | GND                | Zemnicí pin  |
| 10      | GPO0                     | OC2 / PD7          | D. př. – pin č.0 stavových příznaků audio signálu    |
| 11      | GPO1                     | ICP / PD6          | D. př. – pin č.1 stavových příznaků audio signálu    |
| 12      | GPO2                     | OC1A / PD5         | D. př. – pin č.2 stavových příznaků audio signálu    |
| 13      | ZEROL                    | OC1B / PD4         | D/A přev. – nulový audio signál v levém kanálu       |
| 14      | ZEROR                    | INT1 / PD3         | D/A přev. – nulový audio signál v pravém kanálu      |
| 15      | RESET                    | INT0 / PD2         | D/A přev. – resetovací pin                           |
| 16      | DGND                     | GND                | Zemnicí pin  |

V následující podkapitole bude dále popsána deska B s řídicím mikrokontrolérem, tedy ATmegou32.

### 3.5.6 Deska B

Klíčovým prvkem řídicí desky je mikrokontrolér ATmega32 v provedení DIL40. Jeho základní a nezbytné diskrétní součástky nutné k fungování jsou krystal Q1 s frekvencí 16,00 MHz, keramické kondenzátory C1 a C2 s kapacitou 22 pF a resetovací tlačítko S1 umístěné na levé straně desky. Kromě propojovacího konektoru, jehož funkce byla podrobně popsána v předchozí podkapitole, se na desce nachází ještě několik dalších propojovacích konektorů. Na vstupně-výstupním portu A mikrokontroléru se nachází

konektor pro připojení LCD displeje, port B obsluhuje komunikační rozhraní SPI na pinech PB4 až PB7. Kromě těchto pinů se na komunikaci po SPI sběrnici podílí taky pin PD0. Pin PB3 na portu B je hardwarově uzpůsoben jako výstup děličky hodinového signálu. Může tak být využit jako výstupní pin ke generování hodinového signálu pro externí synchronizaci digitálního přijímače (pokud je možné docílit dostatečně přesné frekvence synchronizace). Anebo může být využit jako pin vstupní, kdy plní úlohu detekce hodinového signálu z výstupu přijímače, anebo ze vstupu externího synchronizačního hodinového signálu, který je přiveden z vnějšího rozhraní na desku pomocí konektoru BNC. K tomuto účelu je na desce připravený konektor s označením EXT CLK, který se nachází pod vstupně-výstupním řídicím konektorem. Zbytek nevyužitých pinů z portu B je vyveden na konektor označený jako AUX. Port C mikrokontroléru je v tomto případě přednastaven na připojení k ovládacím tlačítkům a případně taky signalizačním diodám. Port D slouží k přijímání a posílání signálů na desku A, které již nespádají pod SPI komunikaci. Všechny konektory pro připojení periférií (tzn. tlačítek, displeje atd.) mají vždy na jednom z krajních pinů přivedenou zem a na druhém krajním pinu napájecí napětí +5 V. Tohle řešení je velmi pohodlné pro přivedení napájecího napětí ke všem perifériím, ovšem v praxi je použitelné pouze pro proudově nenáročné aplikace (do 200 mA). V případě větší proudové zátěže je nutno odebírat napájení přímo z napájecího zdroje. Dále byl na desku umístěn ISP programovací konektor pro rychlé programování a ladění programového kódu. Programovací rozhraní mikrokontroléru je vyvedeno přes rozhraní SPI a jedná se tak o sdílenou linku s komunikačním rozhraním s oběma čipy na desce A. Aby bylo zamezeno nechtěnému přijetí dat určených pro programování ATmegy některým z čipů, byly na komunikačním rozhraní v obou směrech mezi konektorem pro ISP programátor a desku A umístěny ochranné rezistory o hodnotě 2,7 k $\Omega$ . Tyto rezistory jsou zapojené takovým způsobem, že signály putující mezi deskou a ATmegou nebo ATmegou a ISP programátorem jsou těmito rezistory ovlivněny poměrně málo. Při průchodu signálu mezi ISP programátorem a deskou A již signál musí projít dvojicí těchto ochranných rezistorů, což znamená, že signál z ISP programátoru bude ovlivněn dosti podstatně. Navíc tyto signály, i kdyby byly některým z obvodů na desce A přijaty, by měly být ignorovány. Mikrokontrolér je programován v průběhu resetu, tedy neposílá žádné signály a signál CS, potvrzující adresu daného obvodu (v tomto případě digitálního přijímače) by neměl být přijímačem vůbec přijat. Piny na ISP konektoru jsou zapojeny podle doporučení firmy ATMEL, jak je prezentováno v katalogových listech. [8], [9], [10]

## 4 NÁVRH ŘÍDÍČÍHO PROGRAMU

V kapitole 4 bude stručně nastíněna problematika řízení D/A převodníku a také digitálního přijímače. Kapitola bude zaměřena obzvláště na popis funkcí, které budou potřeba pro nastavování parametrů zpracování digitálního audio signálu.

### 4.1 Řízení D/A převodníku AD1852

#### 4.1.1 První způsob (Hardwarový)

První způsob řízení D/A převodníku je možný pomocí funkčních pinů. Tyto lze najít na obr. 3.2 v kapitole 3.1. Jedná se o skupinu pinů, kdy každý z nich, případně dvojice pinů, zastává nějakou konkrétní funkci řízení. Např. piny 192/48 a 96/48 jsou určeny k nastavování správného vzorkovacího kmitočtu, pin DEEMP zapíná deemfázi na kmitočtu 44.1 kHz, pin MUTE ztlumí signál na minimální hodnotu apod. Řízení se provádí pomocí přivedení napětí +5 V nebo připojením pinu na digitální zem. Nastavením určité kombinace na některé dvojici pinů lze řídit více možností dané funkce. [4]

Způsob řízení spočívá v tom, že k těmto funkčním pinům by byly připojeny vstupně-výstupní porty řídicího mikrokontroléru ATmega32, který by pro jednotlivé piny převodníku připojoval k napájecímu napětí nebo je uzemňoval. Tento způsob řízení je sice jednodušší, logický a nenáročný, nicméně katalogové listy D/A převodníku uvádějí, že převodník AD1852 disponuje více funkcemi, než je možno takto nastavovat pomocí funkčních pinů. V následující kapitole bude zmíněn další způsob, který zajišťuje možnost ovládní více prvků nežli pomocí těchto funkčních pinů.

#### 4.1.2 Druhý způsob (Softwarový)

Daleko lepším a univerzálnějším, ikdyž o poznání složitějším způsobem pro řízení D/A převodníku, je způsob řízení pomocí sériového komunikačního rozhraní SPI (Serial Peripheral Interface – tedy sériové periferní komunikační rozhraní). Katalogové listy AD1852 uvádějí, že D/A převodník disponuje piny CLATCH, CCLK a CDATA, které slouží k jednosměrnému přenosu sériových komunikačních dat pomocí sériového komunikačního rozhraní SPI. Stejným rozhraním disponuje i mikrokontrolér ATmega32 a tedy se veškerá komunikace s převodníkem dokáže zredukovat na pouhé 3 vodiče, navíc je možno spravovat více funkcí (než u funkčních pinů), které převodník nabízí a v neposlední řadě na straně mikrokontroléru zůstanou i nadále volné další zbylé vstupně výstupní piny, které mohou být použity na další přídatné funkce, např. zobrazování údajů na LCD displeji (komunikace s displejovým řadičem), vstupy pro pole ovládacích tlačítek, výstupy pro signalizační LED diody apod. [4]

Kromě D/A převodníku disponuje sériovým komunikačním rozhraním také digitální přijímač. Oproti ovládní funkčními piny lze pomocí SPI navíc řídit např. hlasitost nebo vzorkovací frekvenci pro interní deemfázi.



## 4.2 Řízení digitálního přijímače CS8416

### 4.2.1 První způsob (Hardwarový)

Podobně, jako byl uveden případ řízení pro obvod AD1852 pomocí funkčních pinů, i zde je možno se setkat s možností řízení digitálního přijímače pomocí funkčních pinů (tento způsob se v katalogových listech nazývá „Hardware Mode“, proto i zde byl odvozen název způsobu takového řízení jako Hardwarový). V tomto případě však nastává zajímavá skutečnost, a sice, že funkce některých funkčních pinů je jiná, než když je digitální přijímač nastaven v druhém módu (o kterém pojednává další podkapitola). [5]

Takovýto způsob řízení přijímače je z hlediska funkčnosti možný, ovšem v tomto případě je vhodnější v jiných aplikacích, které vyžadují pouze jednorázové a trvalé nastavení parametrů (např. v aplikacích pro domácí kino, zvukový výstup z DVD přehrávače apod.) Navíc takový způsob řízení skýtá daleko užší spektrum možností než u softwarového módu.

### 4.2.2 Druhý způsob (Softwarový)

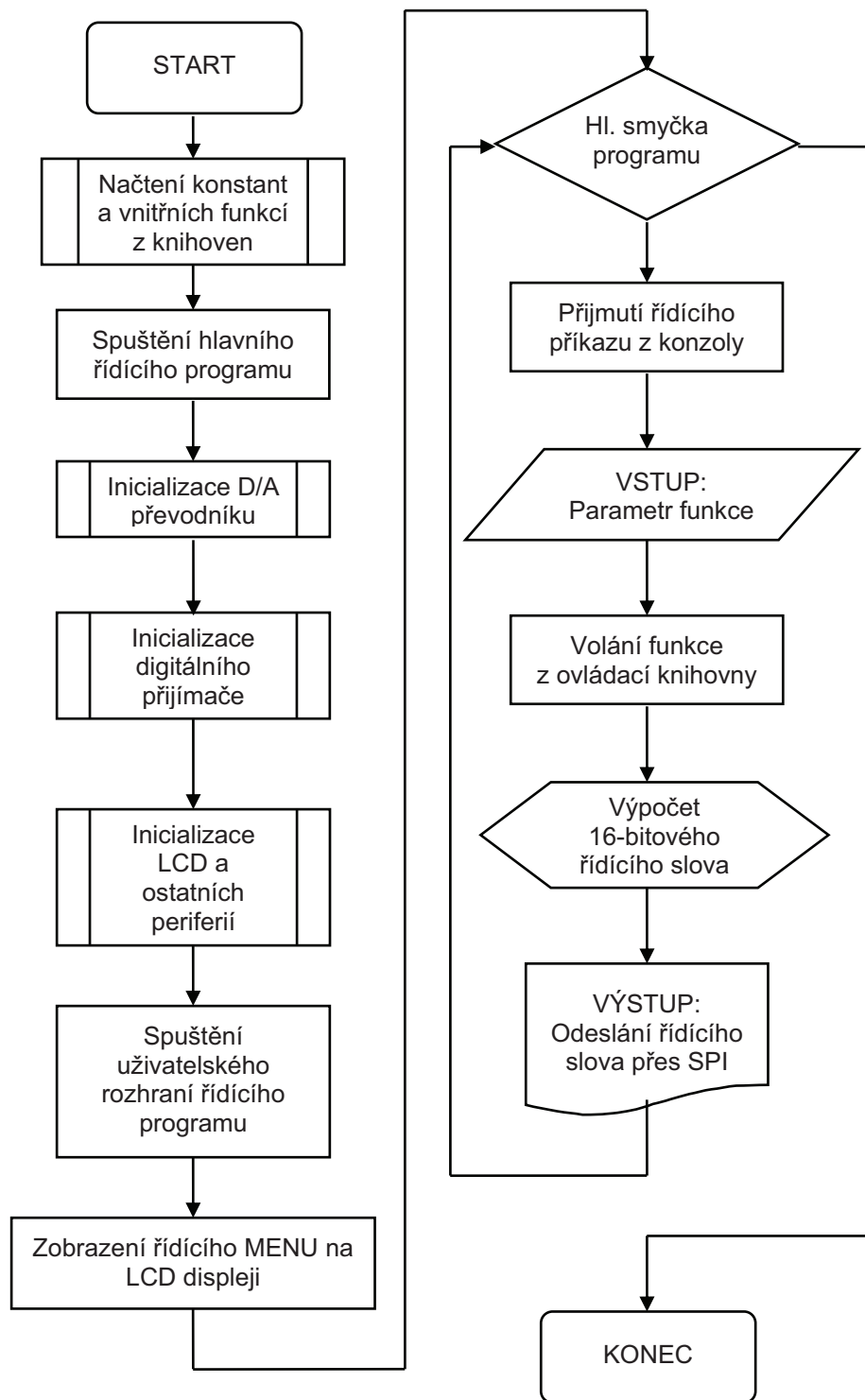
Druhý způsob řízení digitálního přijímače, pomocí sériové komunikační sběrnice, patří, podobně jak bylo naznačeno v kapitole 4.1.2, k velmi univerzálním a také rozšířeným. Při implementaci sériového komunikačního rozhraní pro jeden obvod a mikrokontrolér lze použít nesrovnale méně řídicích pinů, než je tomu při řízení pomocí funkčních pinů. Sériová komunikace pro svůj chod totiž potřebuje zde asi 4 piny, zatím co pro řízení funkčními piny by muselo být na mikrokontroléru k dispozici odhadem asi 10 vstupně-výstupních pinů. Funkce použití přijímače při softwarovém řízení jsou taktéž velmi široké. Částečně lze např. automaticky detekovat formát digitálního signálu posílaný vysílacím zařízením. Tento údaj poté může zpracovat mikrokontrolér a poslat danou řídicí zprávu D/A převodníku a také zobrazit daný údaj na displeji.

Pro oba obvody, mezi kterými probíhá komunikace (buť zprostředkovaně prostřednictvím řídicího členu, zde ATmega32), bude spotřeba řídicích pinů u sériové komunikace o jeden pin navíc (čili celkem 5 pinů), zatím co pomocí funkčních pinů bychom najednou museli použít výše zmíněných vstupně-výstupních pinů (nebo portů) asi dvakrát tolik (tedy kolem 20-ti). Což rozhodně není zanedbatelný rozdíl.

Jak se dále uvádí v katalogových listech digitálního přijímače, tento podporuje více formátů pro sériovou komunikaci, tedy nejen SPI (formát, který bude použit), ale také formát I<sup>2</sup>C. Pro formát komunikace SPI tedy budou použity piny s označením CDO<sub>UT</sub>, CCLK, CDIN a CS. V rámci minulých projektů byla zkoumána i možnost fyzicky rovnou propojit výstupní piny SPI rozhraní digitálního přijímače se vstupními piny SPI rozhraní D/A převodníku. V tomto případě, jak bylo zjištěno, by posílané příkazy z přijímače nemusely přesně korespondovat s registry D/A převodníku, protože příkazy obou obvodů nejsou nijak standardizované a každý výrobce si vyvíjí svoji sadu instrukcí, tudíž by se D/A převodník nemusel nastavovat korektně. Také by samozřejmě nebylo možné nastavovat potřebné parametry pomocí uživatelského programu v mikrokontroléru, ani by nebylo možné je z patřičných registrů vyčítat (a na základě toho pak informovat uživatele o různých stavech zařízení pomocí nápisu na LCD displeji, rozsvícením indikační LED diody apod.). [5]

### 4.3 Vývojový diagram řídicího programu

Následující vývojový diagram níže naznačuje chování řídicího programu mikrokontroléru. Tento bude koncipován jako hlavní řídicí program obsahující jednak samotné jádro, které zahrnuje uživatelské rozhraní, tedy výpis na displej, přijímání příkazů a posílání předem definovaných kódů pomocí SPI a dále pomocných knihoven, které obsahují všechny potřebné ovládací funkce pro oba obvody. Tím je zajištěno, že každé existující funkci (jinými slovy každému měnitelnému parametru obvodu digitálního přijímače či D/A převodníku) bude přiřazena unikátní řídicí sekvence, která je posléze odesílána SPI rozhraním danému obvodu. Signálové cesty pro posílání hodinových signálů a surových dat z mikrokontroléru jsou pro oba obvody společné. Dalšími signály, které mají za úkol rozlišování „adresátů“ dané řídicí sekvence, jsou tzv. signály Slave Select (zkratka SS). Tento signál je vždy aktivován během datového přenosu na odpovídajícím pinu toho zařízení, pro které je daný příkaz určen.



Obr. 4.1: Vývojový diagram řídicího programu

## 4.4 Důležité funkce pro řízení periferií

### 4.4.1 Ovládací knihovna pro D/A převodník AD1852

V této podkapitole budou sepsány všechny dostupné funkce pro řízení D/A převodníku AD1852 pomocí rozhraní SPI. Obvod obsahuje celkem 3 řídicí registry, které nastavují celkem 8 různých funkcí. Jedná se především o tyto funkce:

- **Hlasitost levého kanálu**
- **Hlasitost pravého kanálu**
- **Nastavení módu interního hodinového signálu (souvisí s nastavením vzorkovací frekvence)**
- **Bitové rozlišení vstupního digitálního signálu**
- **Zarovnání bitů sériového digitálního signálu**
- **Výběr filtru pro deemfázi**

### 4.4.2 Ovládací knihovna pro Digitální audio přijímač CS8416

V této podkapitole se nachází soupis všech důležitých funkcí pro řízení digitálního audio přijímače CS8416 pomocí rozhraní SPI. Je nutno též podotknout, že přijímač CS8416 obsahuje celkem 40 ovládacích a informačních registrů. Některé jsou určeny pro čtení i zápis, jiné pouze pro čtení. Pro účely tohoto prototypu nebude potřeba některé registry používat. Hlavní funkce, které se budou nastavovat jsou:

- **Zapnutí či vypnutí interní smyčky fázového závěsu PLL (souvisí s rekonstrukcí hodinového signálu ze vstupního audio signálu)**
- **Nastavení módu výstupního hodinového signálu pro D/A převodník**
- **Výběr filtru pro deemfázi**
- **Výběr vstupů audio signálu**
- **Bitové rozlišení výstupního digitálního signálu**

### 4.4.3 Popis hlavního řídicího programu

Úkolem hlavního řídicího programu je zajistit chod uživatelského prostředí s ovládacím MENU zobrazovaným na displeji. Dále je toto MENU a jeho ovládání uživatelem pevně svázáno s interními ovládacími funkcemi pro každý s obvodů. Tyto funkce, resp. jejich označení a parametry (vyjádřené binárně) jsou obsaženy v ovládacích knihovnách a za běhu programu jsou podle potřeby vyčítány a aplikovány. Tudíž, při potřebě změnit nesprávně naprogramovaný kód funkce nebo parametru není potřeba zasahovat do řídicího jádra a je možno funkci vyhledat a předělat v knihovně. Stejně tak ovládací knihovny zprostředkují asociaci binárních kódů s přehlednými názvy funkcí nebo jejich zkratkami. Tímto způsobem lze všechny funkce v aplikačním jádru přehledně uspořádat a pracovat s nimi. Řídicí MENU je koncipováno takovým způsobem, že jednotlivé funkce uživatel vybírá pomocí šipek nahoru a dolů, parametry v jednotlivých funkcích vybírá šipkami doleva nebo doprava.

#### 4.4.4 Soupis hlaviček funkcí

V této kapitole jsou sepsány hlavičky všech funkcí v knihovnách a v hlavním řídicím programu, které budou používány. Celý zdrojový kód bude k dispozici v příloze.

```
void DRx_init(void);
void SPI_MasterAndSlaveInit(void);
void SPI_Tx(char m, char data);
char SPI_Rx(char m);
void DAC_init(void);
void SPI_MasterInit(void);
void SPI_Tx(char cData2, char cData1);
void set_volume_left(unsigned int vol);
void set_volume_right(unsigned int vol);
void set_int_mode(unsigned char im);
void set_serial_mode(unsigned char sdim);
void set_no_of_bits(unsigned char nob);
void set_deemp_filter(unsigned char df);
void mute(unsigned char m);
void soft_reset(void);
void f_ctrl00(char c0);
void f_ctrl01(char c1);
void f_ctrl02(char c2);
void f_ctrl03(char c3);
void f_ctrl04(char c4);
void f_ctrl05(char c5);
void f_ctrl06(char c6);
void f_ctrl07(char c7);
void f_ctrl08(char c8);
void f_ctrl09(char c9);
lcd_init(LCD_DISP_ON);
lcd_clrscr();
```

## 5 ZÁVĚR

V první kapitole této diplomové práce bylo provedeno stručné seznámení s problematikou A/D a D/A převodů, také s činností moderních D/A převodníků Sigma-delta, dále bylo zmíněno, co to jsou multistandardní převodníky a jakým způsobem je lze využívat a nastavovat. V teoretické části práce pak byla také postupně nastíněna problematika celého návrhu obvodu s D/A převodníkem a bylo popsáno také vlastní technické řešení obvodu a popis zapojení včetně blokového schématu. Byl zde též navržen vývojový diagram řídicího programu a část algoritmu pro mikroprocesorové řízení.

V praktické části této práce bylo navrženo vlastní obvodové schéma celého zapojení, včetně digitálního přijímače, D/A převodníku, výstupních audio obvodů, mikroprocesorového řízení a dalších nutných periférií. Byly zde též s konečnou platností zvoleny integrované obvody pro D/A převodník a pro digitální přijímač, jakož i typ mikrokontroléru pro řízení obvodu, LCD displeje i výstupních operačních zesilovačů. Byly také vyhotoveny konstrukční podklady pro realizaci funkčního prototypu (jako návrhy desky, soupisky součástek, osazovací plány atd.).

V experimentální části práce pak byly vyrobeny desky plošných spojů a celé zapojení bylo zhotoveno do fáze hotových modulů. Dále byl upraven celý řídicí algoritmus pro ovládání obou obvodů na hlavní desce. K datu odevzdání diplomové práce se bohužel nepodařilo celé zapojení úspěšně oživit a proměřit. Nicméně i po datu odevzdání této práce budou podniknuty další patřičné kroky k oživení celého zapojení nebo alespoň některých jeho částí. V případě úspěšného oživení budou hotové moduly zasazeny do krabičky a bude provedeno závěrečné měření zapojení a stanovení technických parametrů D/A převodníku.

## LITERATURA

- [1] DŘÍNOVSKÝ, J., *Radioelektronická měření, A/D a D/A převodníky* [online]. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav radioelektroniky, – [cit. 14. května 2012]. Dostupné na [www: <http://www.urel.feec.vutbr.cz/~drinovsky/?download=MREM\\_2010\\_P\\_02.pdf>](http://www.urel.feec.vutbr.cz/~drinovsky/?download=MREM_2010_P_02.pdf)
- [2] VRBA, K., *Vzájemný převod analogových a digitálních signálů* [online]. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, – [cit. 11. května 2012]. Dostupné na [www: <https://www.vutbr.cz/elearning/mod/resource/view.php?inpopup=true&id=163140>](https://www.vutbr.cz/elearning/mod/resource/view.php?inpopup=true&id=163140).
- [3] HANÁK, P., VRBA, K., *A/D převodníky (kapitola sigma-delta)* [online]. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií. Ústav telekomunikací, – [cit. 11. května 2012]. Dostupné na [www: <https://www.vutbr.cz/elearning/mod/resource/view.php?id=163149>](https://www.vutbr.cz/elearning/mod/resource/view.php?id=163149).
- [4] Analog Devices [online]. Datasheet *AD1852 – Stereo, 24-Bit, 192 kHz, Multibit, Sigma-Delta DAC. Analog Devices, Inc., 2009* – [cit. 14. ledna 2011]. Dostupné na [www: <http://www.analog.com/static/imported-files/data\\_sheets/AD1852.pdf>](http://www.analog.com/static/imported-files/data_sheets/AD1852.pdf)
- [5] Cirrus Logic [online]. Datasheet *CS8416 – 192 kHz Digital Audio Interface Receiver. Cirrus Logic, Inc., 2007* – [cit. 23. března 2011]. Dostupné na [www: <http://www.cirrus.com/en/pubs/proDatasheet/CS8416\\_F3.pdf>](http://www.cirrus.com/en/pubs/proDatasheet/CS8416_F3.pdf)
- [6] Toshiba [online]. Datasheet *TORX173 – Fiber optic receiving module for digital audio equipment. Toshiba, corp., 2001* – [cit. 10. května 2012]. Dostupné na [www: <http://www.datasheetcatalog.org/datasheet/toshiba/2033.pdf>](http://www.datasheetcatalog.org/datasheet/toshiba/2033.pdf)
- [7] Analog Devices [online]. Datasheet *AD797 – Ultralow Distortion, Ultralow Noise Op Amp. Analog Devices, Inc., 2010* – [cit. 16. května 2012]. Dostupné na [www: <http://www.analog.com/static/imported-files/data\\_sheets/AD797.pdf>](http://www.analog.com/static/imported-files/data_sheets/AD797.pdf)
- [8] Atmel [online]. Datasheet *ATmega32, ATmega32L – 8-bit AVR Microcontroller with 32KBytes In-System Programmable Flash. Atmel, Corp., 2011* – [cit. 16. května 2012]. Dostupné na [www: <http://www.atmel.com/Images/doc2503.pdf>](http://www.atmel.com/Images/doc2503.pdf)
- [9] Atmel [online]. Datasheet *AVR, 8-bit RISC Microcontroller – AVR151: Setup And Use of The SPI. Atmel, Corp., 2008* – [cit. 16. května 2012]. Dostupné na [www: <http://www.atmel.com/Images/doc2585.pdf>](http://www.atmel.com/Images/doc2585.pdf)
- [10] Atmel [online]. Datasheet *AVR, 8-bit RISC Microcontroller – AVR910: In-System Programming. Atmel, Corp., 2008* – [cit. 16. května 2012]. Dostupné na [www: <http://www.atmel.com/Images/doc0943.pdf>](http://www.atmel.com/Images/doc0943.pdf)

## SEZNAM PŘÍLOH

|   |                                   |    |
|---|-----------------------------------|----|
| A | Schéma obvodu.....                | 31 |
| B | Návrhy desek plošných spojů ..... | 32 |
| C | Osazovací plány .....             | 36 |
| D | Soupisky součástek .....          | 41 |
| E | Zdrojové kódy .....               | 46 |
| F | Fotodokumentace .....             | 60 |
| G | CD s dokumentací.....             | 62 |



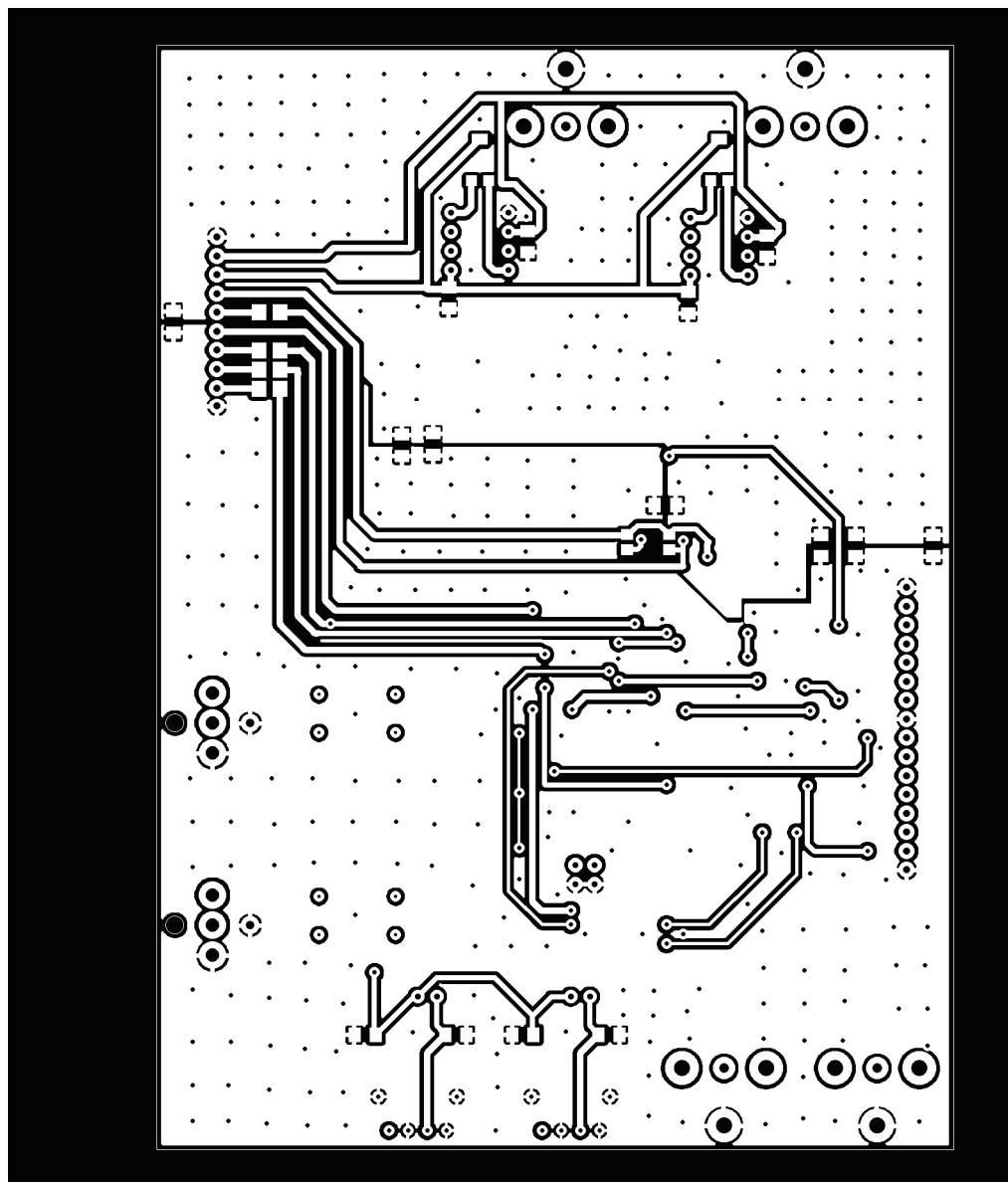
# **A SCHÉMA OBVODU**

## **A.1 Celkové obvodové schéma zapojení**

- Celkové obvodové schéma zapojení je vytištěno na samostatný list formátu A3.

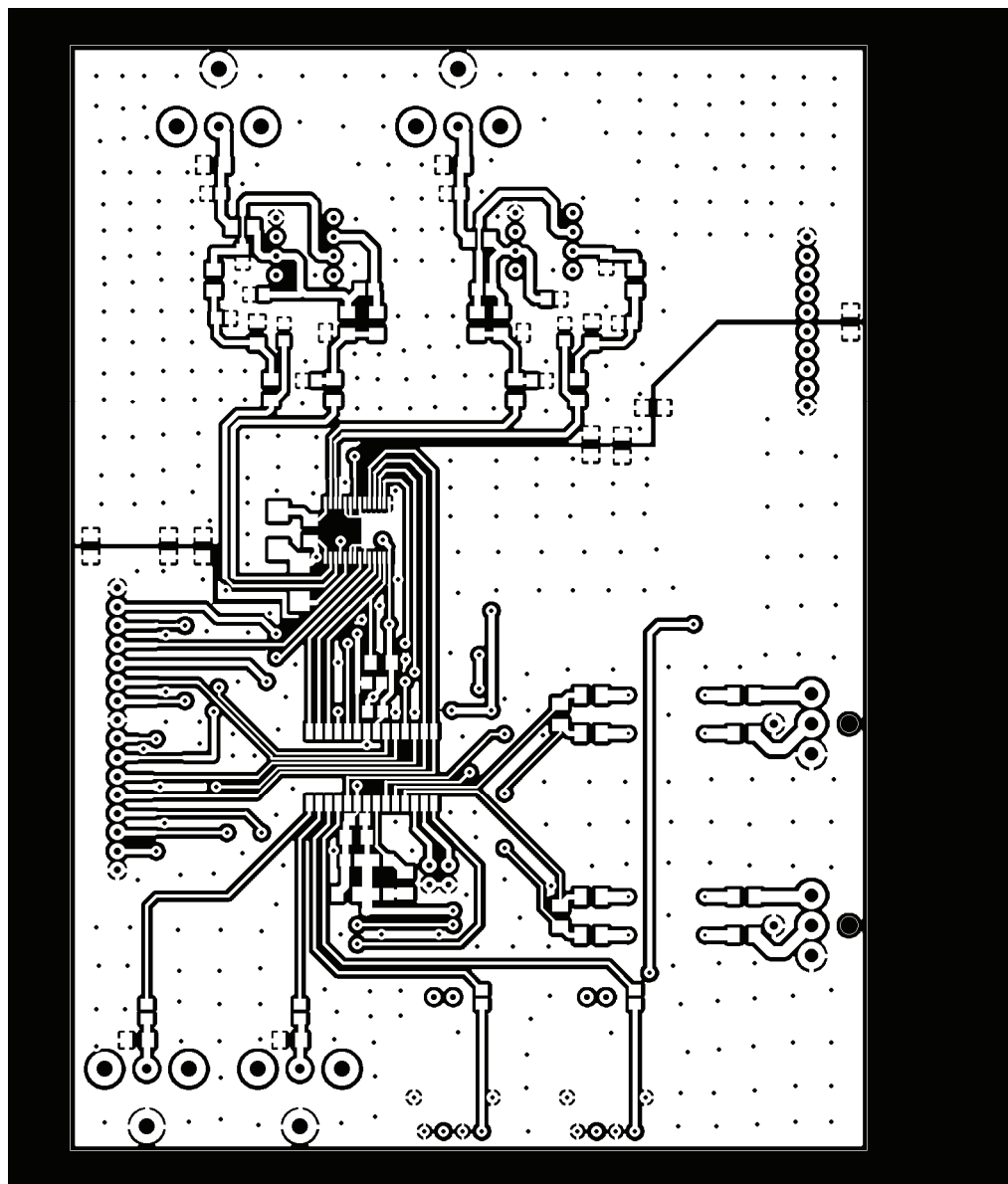
## B NÁVRHY DESEK PLOŠNÝCH SPOJŮ

### B.1 Deska A – strana TOP



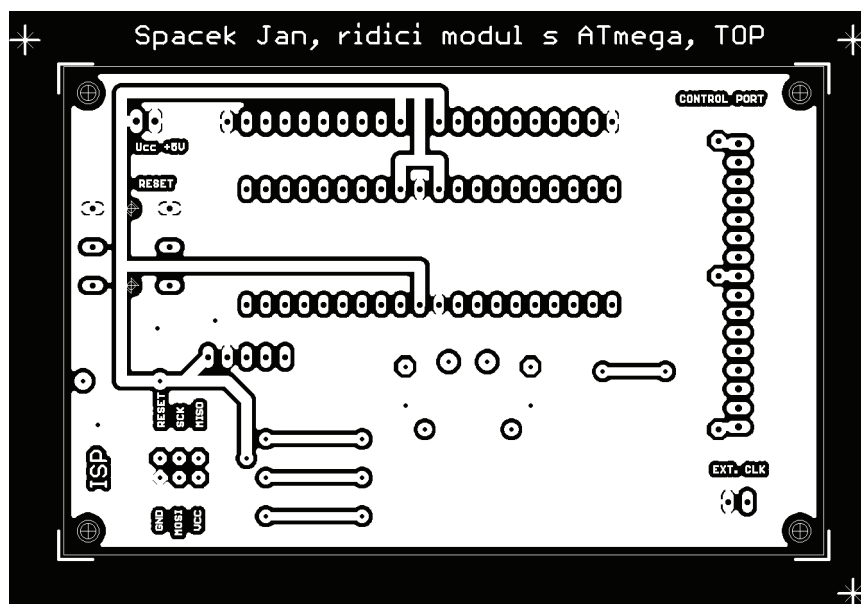
Rozměr desky 105,73 x 146,79 [mm], měřítko M1:1

## B.2 Deska A – strana BOTTOM



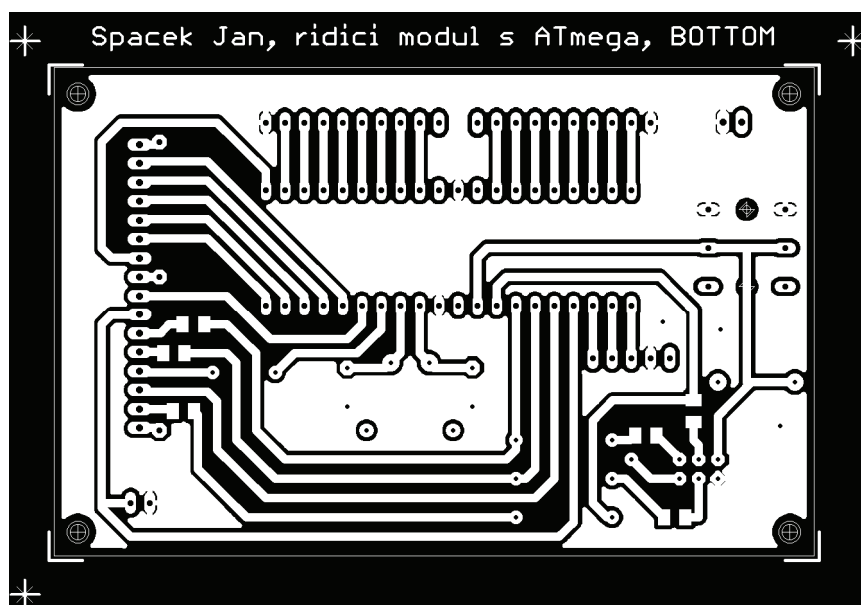
Rozměr desky 105,73 x 146,79 [mm], měřítko M1:1

### B.3 Deska B – strana TOP



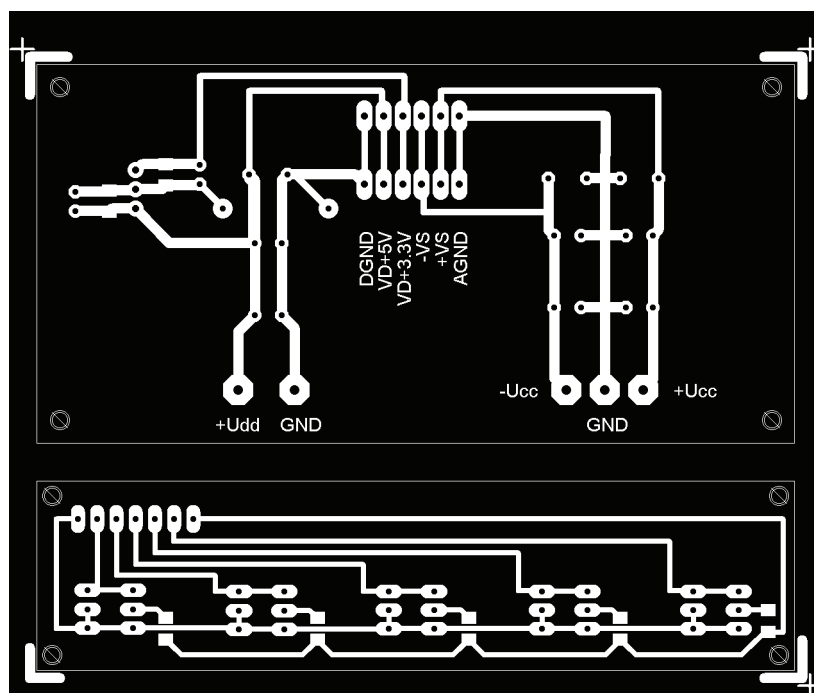
Rozměr desky 100,33 x 64,14 [mm], měřítko M1:1

### B.4 Deska B – strana BOTTOM



Rozměr desky 100,33 x 64,14 [mm], měřítko M1:1

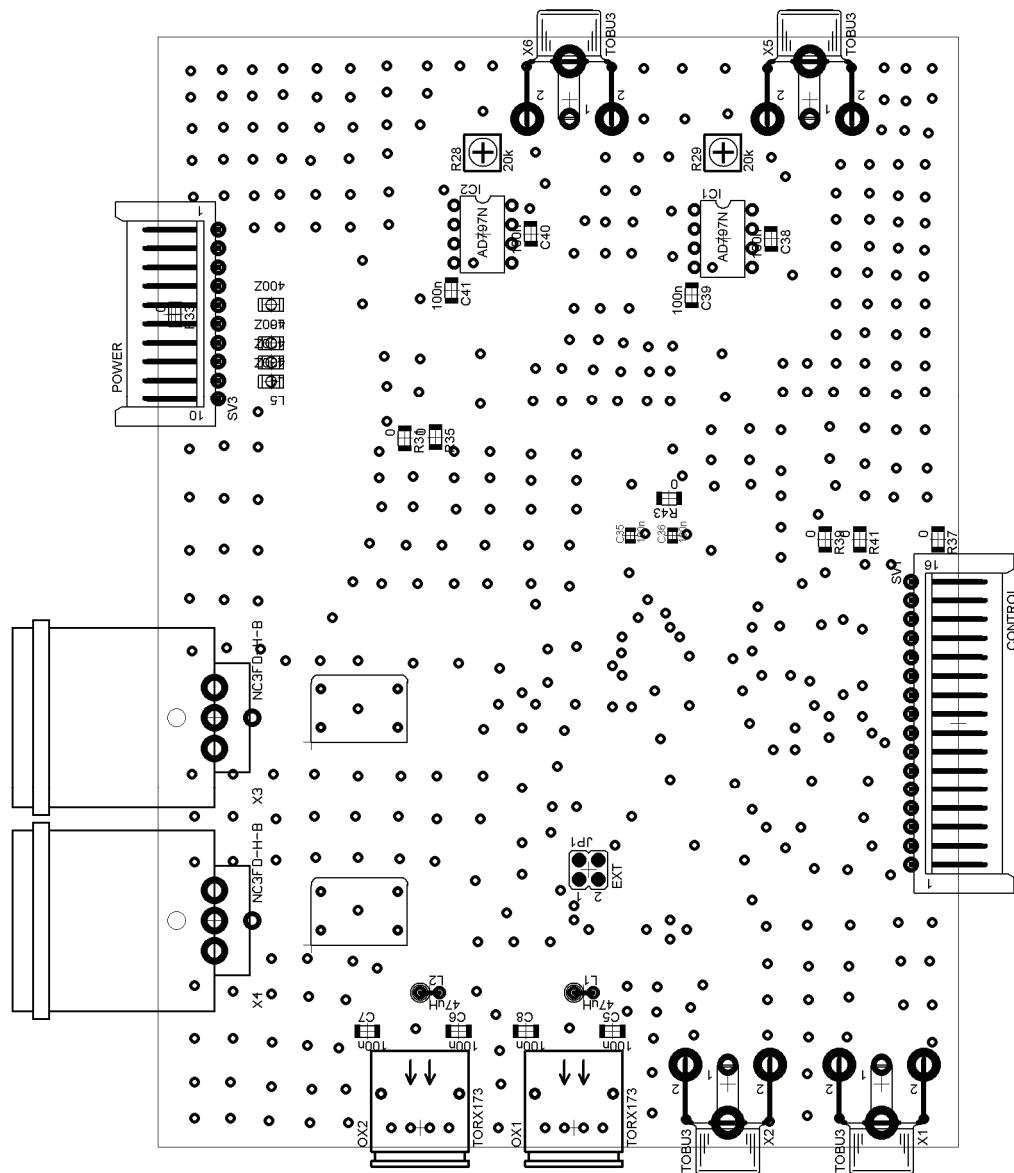
## B.5 Deska C + Deska D – strana BOTTOM



Rozměry desek: Deska C (zdroj, nahoře) 100,00 x 50,00 [mm]; Deska D (tlačítka, dole) 100,00 x 25,00 [mm], měřítko M1:1

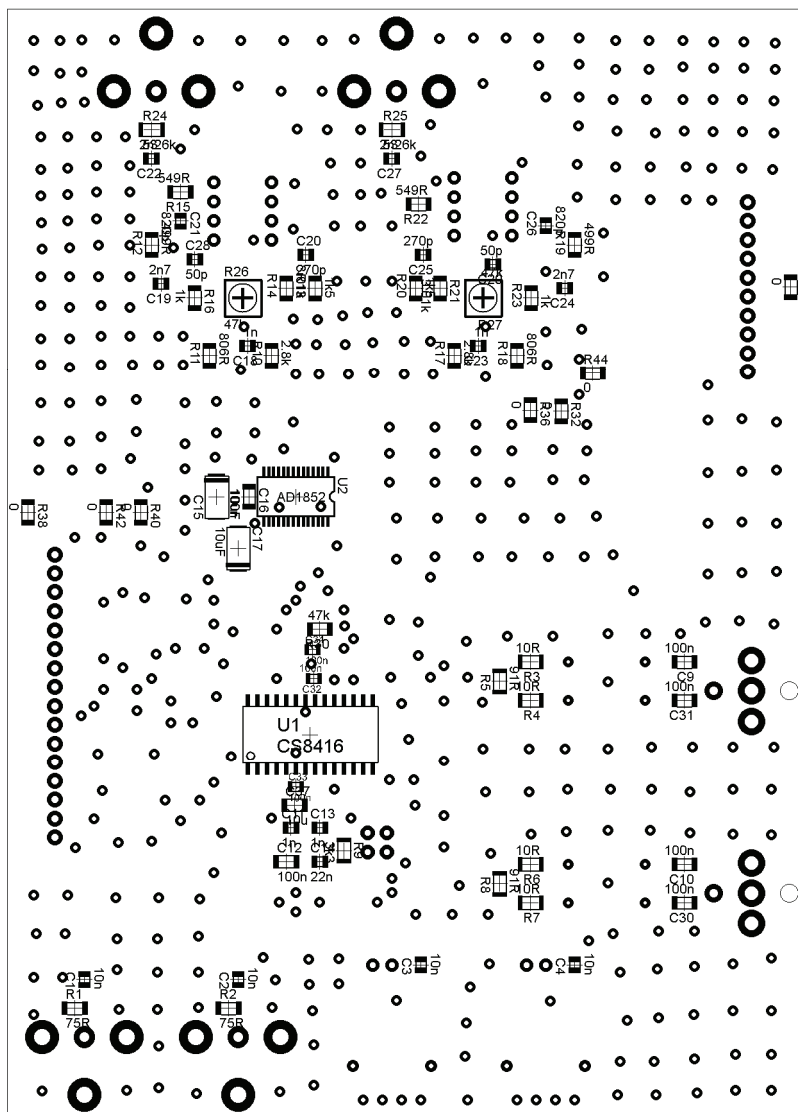
# C OSAZOVACÍ PLÁNY

## C.1 Deska A – strana TOP



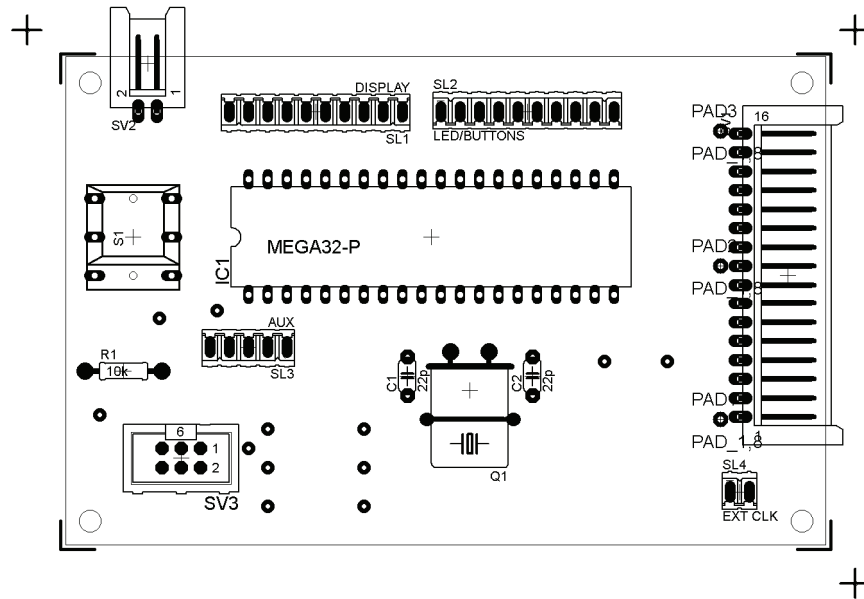
Rozměr desky 105,73 x 146,79 [mm], měřítko M1:1

## C.2 Deska A – strana BOTTOM



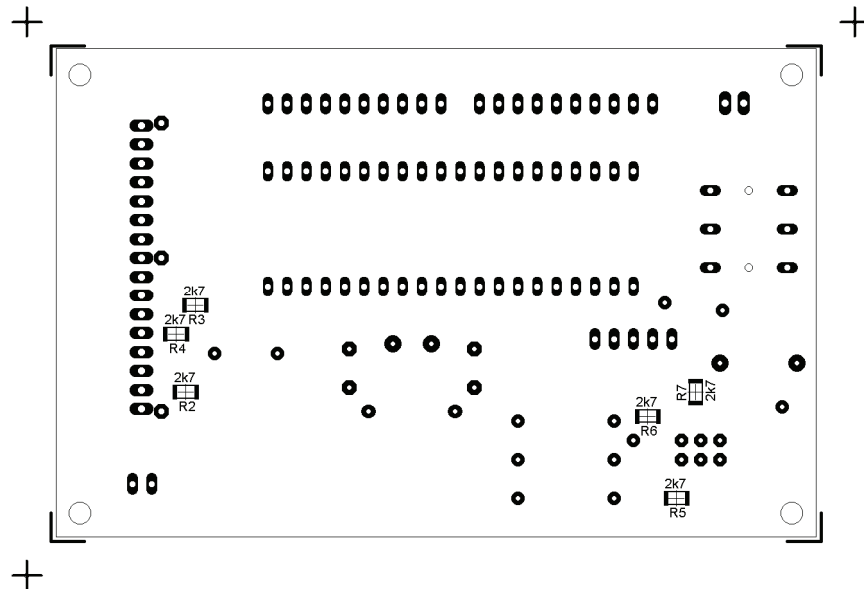
Rozměr desky 105,73 x 146,79 [mm], měřítko M1:1

### C.3 Deska B – strana TOP



Rozměr desky 100,33 x 64,14 [mm], měřítko M1:1

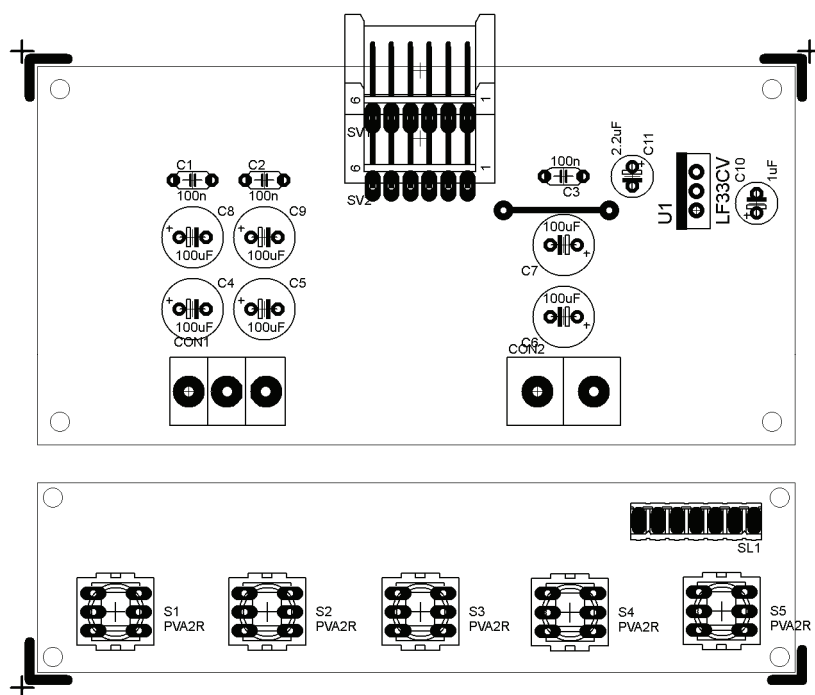
### C.4 Deska B – strana BOTTOM



Rozměr desky 100,33 x 64,14 [mm], měřítko M1:1

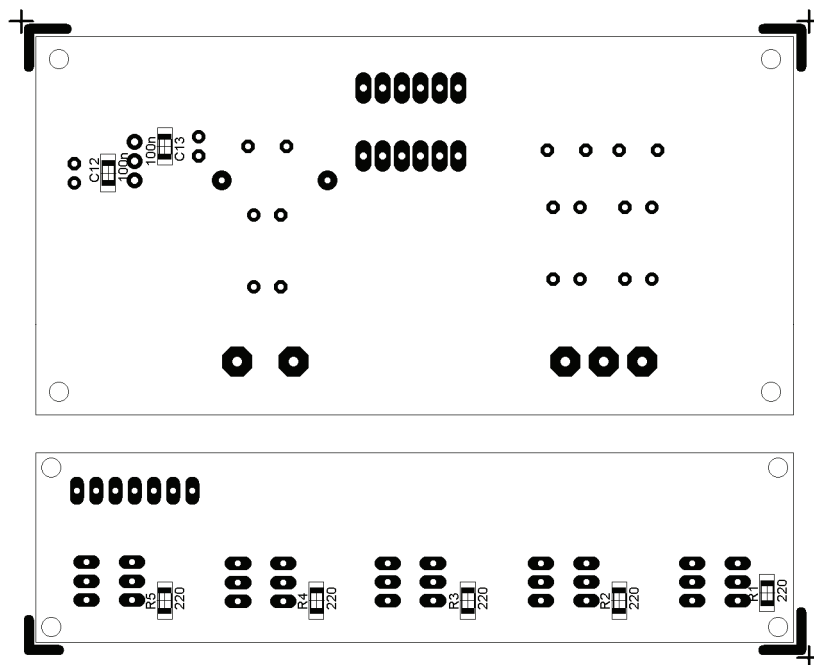


## C.5 Deska C + Deska D – strana TOP



Rozměry desek: Deska C (zdroj, nahoře) 100,00 x 50,00 [mm]; Deska D (tlačítka, dole) 100,00 x 25,00 [mm], měřítko M1:1

## C.6 Deska C + Deska D – strana BOTTOM



Rozměry desek: Deska C (zdroj, nahoře) 100,00 x 50,00 [mm]; Deska D (tlačítka, dole) 100,00 x 25,00 [mm], měřítko M1:1

# D SOUPOISKY SOUČÁSTEK

## D.1 Deska A

| Označení | Hodnota | Pouzdro    | Popis         |
|----------|---------|------------|---------------|
| C1       | 10n     | C0805      | SMD kapacitor |
| C2       | 10n     | C0805      | SMD kapacitor |
| C3       | 10n     | C0805      | SMD kapacitor |
| C4       | 10n     | C0805      | SMD kapacitor |
| C5       | 100n    | C1206      | SMD kapacitor |
| C6       | 100n    | C1206      | SMD kapacitor |
| C7       | 100n    | C1206      | SMD kapacitor |
| C8       | 100n    | C1206      | SMD kapacitor |
| C9       | 100n    | C1206      | SMD kapacitor |
| C10      | 100n    | C1206      | SMD kapacitor |
| C11      | 1n      | C0805      | SMD kapacitor |
| C12      | 100n    | C1206      | SMD kapacitor |
| C13      | 1n      | C0805      | SMD kapacitor |
| C14      | 22n     | C0805      | SMD kapacitor |
| C15      | 10uF    | C/6032-28R | SMD kapacitor |
| C16      | 100n    | C1206      | SMD kapacitor |
| C17      | 10uF    | C/6032-28R | SMD kapacitor |
| C18      | 1n      | C0805      | SMD kapacitor |
| C19      | 2n7     | C0805      | SMD kapacitor |
| C20      | 270p    | C0805      | SMD kapacitor |
| C21      | 820p    | C0805      | SMD kapacitor |
| C22      | 2n2     | C0805      | SMD kapacitor |
| C23      | 1n      | C0805      | SMD kapacitor |
| C24      | 2n7     | C0805      | SMD kapacitor |
| C25      | 270p    | C0805      | SMD kapacitor |
| C26      | 820p    | C0805      | SMD kapacitor |
| C27      | 2n2     | C0805      | SMD kapacitor |
| C28      | 50p     | C0805      | SMD kapacitor |
| C29      | 50p     | C0805      | SMD kapacitor |
| C30      | 100n    | C1206      | SMD kapacitor |
| C31      | 100n    | C1206      | SMD kapacitor |
| C32      | 100n    | C0805K     | SMD kapacitor |
| C33      | 100n    | C0805K     | SMD kapacitor |
| C34      | 100n    | C0805K     | SMD kapacitor |
| C35      | 100n    | C0805K     | SMD kapacitor |
| C36      | 100n    | C0805K     | SMD kapacitor |

|     |         |           |                  |
|-----|---------|-----------|------------------|
| C37 | 10u     | C1206     | SMD kapacitor    |
| C38 | 100n    | C1206     | SMD kapacitor    |
| C39 | 100n    | C1206     | SMD kapacitor    |
| C40 | 100n    | C1206     | SMD kapacitor    |
| C41 | 100n    | C1206     | SMD kapacitor    |
| IC1 | AD797N  | DIL08     | Nízkošumový OZ   |
| IC2 | AD797N  | DIL08     | Nízkošumový OZ   |
| JP1 | EXT     | JP2QE     | Jumper – 2x      |
| L1  | 47uH    | 0207/2V   | Cívka axiální    |
| L2  | 47uH    | 0207/2V   | Cívka axiální    |
| L3  | 400Z    | L3225P    | SMD tlumivka     |
| L4  | 400Z    | L3225P    | SMD tlumivka     |
| L5  | 400Z    | L3225P    | SMD tlumivka     |
| L6  | 400Z    | L3225P    | SMD tlumivka     |
| OX1 | TORX173 | TORX173   | Optický konektor |
| OX2 | TORX173 | TORX173   | Optický konektor |
| R1  | 75R     | R1206     | SMD rezistor     |
| R2  | 75R     | R1206     | SMD rezistor     |
| R3  | 10R     | M1206     | SMD rezistor     |
| R4  | 10R     | M1206     | SMD rezistor     |
| R5  | 91R     | M1206     | SMD rezistor     |
| R6  | 10R     | M1206     | SMD rezistor     |
| R7  | 10R     | M1206     | SMD rezistor     |
| R8  | 91R     | M1206     | SMD rezistor     |
| R9  | 3k3     | M1206     | SMD rezistor     |
| R10 | 2.8k    | R1206     | SMD rezistor     |
| R11 | 806R    | R1206     | SMD rezistor     |
| R12 | 499R    | R1206     | SMD rezistor     |
| R13 | 1k5     | R1206     | SMD rezistor     |
| R14 | 3.1k    | R1206     | SMD rezistor     |
| R15 | 549R    | R1206     | SMD rezistor     |
| R16 | 1k      | R1206     | SMD rezistor     |
| R17 | 2.8k    | R1206     | SMD rezistor     |
| R18 | 806R    | R1206     | SMD rezistor     |
| R19 | 499R    | R1206     | SMD rezistor     |
| R20 | 1k5     | R1206     | SMD rezistor     |
| R21 | 3.1k    | R1206     | SMD rezistor     |
| R22 | 549R    | R1206     | SMD rezistor     |
| R23 | 1k      | R1206     | SMD rezistor     |
| R24 | 53.6k   | M1206     | SMD rezistor     |
| R25 | 53.6k   | M1206     | SMD rezistor     |
| R26 | 47k     | RTRIM4G/J | SMD trimr        |
| R27 | 47k     | RTRIM4G/J | SMD trimr        |
| R28 | 20k     | RTRIM4G/J | SMD trimr        |

|     |           |           |                      |
|-----|-----------|-----------|----------------------|
| R29 | 20k       | RTRIM4G/J | SMD trimr            |
| R30 | 47k       | R1206     | SMD rezistor         |
| R31 | 0         | R1206     | Ploška pro propojku  |
| R32 | 0         | R1206     | Ploška pro propojku  |
| R33 | 0         | R1206     | Ploška pro propojku  |
| R34 | 0         | R1206     | Ploška pro propojku  |
| R35 | 0         | R1206     | Ploška pro propojku  |
| R36 | 0         | R1206     | Ploška pro propojku  |
| R37 | 0         | R1206     | Ploška pro propojku  |
| R38 | 0         | R1206     | Ploška pro propojku  |
| R39 | 0         | R1206     | Ploška pro propojku  |
| R40 | 0         | R1206     | Ploška pro propojku  |
| R41 | 0         | R1206     | Ploška pro propojku  |
| R42 | 0         | R1206     | Ploška pro propojku  |
| R43 | 0         | R1206     | Ploška pro propojku  |
| R44 | 0         | R1206     | Ploška pro propojku  |
| SV1 | CONTROL   | L16P      | Propojovací konektor |
| SV3 | POWER     | L10P      | Napájecí konektor    |
| T1  | PE-65612  | PE-65612  | Odděl. transformátor |
| T2  | PE-65612  | PE-65612  | Odděl. transformátor |
| U1  | CS8416    | SO-28L    | Digitální přijímač   |
| U2  | AD1852    | SSOP28    | D/A převodník        |
| X1  | TOBU3     | TOBU3     | Konektor CINCH       |
| X2  | TOBU3     | TOBU3     | Konektor CINCH       |
| X3  | NC3FD-H-B | NC3FD-H-B | Konektor XLR         |
| X4  | NC3FD-H-B | NC3FD-H-B | Konektor XLR         |
| X5  | TOBU3     | TOBU3     | Konektor CINCH       |
| X6  | TOBU3     | TOBU3     | Konektor CINCH       |

## D.2 Deska B

| Označení | Hodnota     | Pouzdro     | Popis                 |
|----------|-------------|-------------|-----------------------|
| C1       | 22p         | C050-024X44 | Keramický kapacitor   |
| C2       | 22p         | C050-024X44 | Keramický kapacitor   |
| IC1      | MEGA32-P    | DIL40       | Mikrokontrolér        |
| PAD1     |             | PAD         | Zemnicí ploška        |
| PAD2     |             | PAD         | Zemnicí ploška        |
| PAD3     |             | PAD         | Zemnicí ploška        |
| Q1       | 16,00 MHz   | HC18U-H     | Krystal 16,00 MHz     |
| R1       | 10k         | 0207/10     | Rezistor 0207         |
| R2       | 2k7         | M1206       | SMD rezistor          |
| R3       | 2k7         | M1206       | SMD rezistor          |
| R4       | 2k7         | M1206       | SMD rezistor          |
| R5       | 2k7         | M1206       | SMD rezistor          |
| R6       | 2k7         | M1206       | SMD rezistor          |
| R7       | 2k7         | M1206       | SMD rezistor          |
| S1       | RESET       | REKN        | Tlačítko resetovací   |
| SL1      | DISPLAY     | 10P         | Konektor pro LCD      |
| SL2      | LED/BUTTONS | 10P         | Konektor pro tlačítka |
| SL3      | AUX         | 05P         | Konektor přídatný     |
| SL4      | EXT CLK     | 02P         | Konektor pro hod. s.  |
| SV1      | CONTROL     | L16P        | Konektor k hlavní d.  |
| SV2      | POWER       | L02P        | Konektor napájecí     |
| SV3      | ISP         | ML6         | Konektor pro ISP      |

### D.3 Deska C + Deska D

| Označení | Hodnota | Pouzdro      | Popis                 |
|----------|---------|--------------|-----------------------|
| C1       | 100nF   | C050-024X044 | Keramický kapacitor   |
| C2       | 100nF   | C050-024X044 | Keramický kapacitor   |
| C3       | 100nF   | C050-024X044 | Keramický kapacitor   |
| C4       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C5       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C6       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C7       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C8       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C9       | 100uF   | E3,5-8       | Elektrolyt. kapacitor |
| C10      | 1uF     | E2,5-6       | Elektrolyt. kapacitor |
| C11      | 2.2uF   | E2,5-6       | Elektrolyt. kapacitor |
| C12      | 100nF   | C1206        | SMD kapacitor         |
| C13      | 100nF   | C1206        | SMD kapacitor         |
| CON1     | L03     | ARK120/3     | Plošky pro kabely (3) |
| CON2     | L02     | ARK110/2     | Plošky pro kabely (2) |
| R1       | 220     | R1206        | SMD rezistor          |
| R2       | 220     | R1206        | SMD rezistor          |
| R3       | 220     | R1206        | SMD rezistor          |
| R4       | 220     | R1206        | SMD rezistor          |
| R5       | 220     | R1206        | SMD rezistor          |
| S1       | LEFT    | PVA2R        | Tlačítko (š. vlevo)   |
| S2       | UP      | PVA2R        | Tlačítko (š. nahoru)  |
| S3       | DOWN    | PVA2R        | Tlačítko (š. dolů)    |
| S4       | RIGHT   | PVA2R        | Tlačítko (š. doprava) |
| S5       | OK      | PVA2R        | Tlačítko (OK)         |
| SL1      |         | 07P          | Plošky pro kabel (7)  |
| SV1      |         | L06P         | Konektor výst. zdroje |
| SV2      |         | L06P         | Konektor výst. zdroje |
| U1       | LF33CV  | TO-220S      | Stabilizátor +3,3 V   |

# E ZDROJOVÉ KÓDY

## E.1 Ovládací knihovna pro obvod AD1852

```
/* ===== */
/*
/*   Ovladani_DAC.c
/*   (c) 2001 Jan Špaček
/*
/*   Popis knihovny:
/*   Knihovna pro ovládání D/A převodníků (implicitně programováno pro D/A
/*   převodník AD1852 od Analog Devices). Ovládání AD1852 je prováděno
/*   mikrokontrolérem ATmega32 pomocí rozhraní SPI.
/*
/* ===== */

#include "Ovladani_DAC.h"
#include <avr/io.h>
#include <util/delay.h>

char v_bits1 = 0x00, v_bits2 = 0x00;
char c_bits1 = 0x01, c_bits2 = 0x00;

//-----

void DAC_init(void)                //inicializace D/A převodníku
{
PORT_DAC &= ~(1<<PIN_DAC_RESET); //nulování prvního pinu, aktivace hardwarového
                                //Resetu
_delay_ms(20);
PORT_DAC |= (1<<PIN_DAC_RESET);  //nastavení prvního pinu, deaktivace
                                //hardwarového Resetu
PORT_DAC |= (1<<PIN_DAC_CLATCH);
}

void SPI_MasterInit(void)          //inicializace SPI rozhraní
{
DDR_SPI = (1<<DD_MOSI)|(1<<DD_SCK)|(0<<DD_MISO)|(0<<DD_SS);
// Nastavení MOSI a SCK jako výstupy, MISO a SlaveSelect jako vstupy

SPCR = (1<<SPE)|(1<<MSTR)|(1<<CPOL)|(1<<CPHA);
// Povolení SPI jako Master, Nastavení hodin: fck/4, Polarita hodin - log.1,
// (LSB first -> DORD = 1) - (MSB first -> DORD = 0)
}

void SPI_Tx(char cData2, char cData1)
//funkce pro odesílání 16-bitových dat po SPI
{
SPDR = cData2;                    //druhý byte (MSB -> ...)
while(!(SPSR & (1<<SPIF)));      //čekání na odvysílání slova (8
bitů)
SPDR = cData1;                    //první byte (... -> LSB)
while(!(SPSR & (1<<SPIF)));      //čekání na odvysílání slova (8
bitů)
PORT_DAC &= ~(1<<PIN_DAC_CLATCH); //Pin CLATCH - impuls do log.0 (potvrzení
                                //kompletního přenosu)
_delay_us(1);                     //Trvání log.0 = 1 us
PORT_DAC |= (1<<PIN_DAC_CLATCH); //Pin CLATCH - impuls do log.1
                                //(konec potvrzovacího impulsu)
_delay_us(100);                   //čekání 100 us
}
```



```

//-----
void set_volume_left(unsigned int vol)          //Bity = xxxxxxxx xxxxxx00
{
    //X-bity=volume(dig.); 14 bitů

    unsigned int b;                            //256 úrovní hlasitosti(0-255),
    krok 64

    b = vol*64;                                //určení kroku
    b = b << 2;                                //posun o 2 místa (registr pro vol)
    b = b + 0x00;                              //označení registru
    v_bits1 = b;                               //zkopírování 1. bytu
    v_bits2 = b >> 8;                          //zkopírování 2. bytu

    SPI_Tx(v_bits2,v_bits1);                  //odeslání dat po SPI
}

void set_volume_right(unsigned int vol) //Bity = xxxxxxxx xxxxxx10
{
    //X-bity=volume(dig.); 14 bitů

    unsigned int b;                            //256 úrovní hlasitosti(0-255), krok 64

    b = vol*64;                                //určení kroku
    b = b << 2;                                //posun o 2 místa (registr pro vol)
    b = b + 0x02;                              //označení registru
    v_bits1 = b;                               //zkopírování 1. bytu
    v_bits2 = b >> 8;                          //zkopírování 2. bytu

    SPI_Tx(v_bits2,v_bits1);                  //odeslání dat po SPI
}

void set_int_mode(unsigned char im)           //Bity = xxxxYYxx xxxxxx01 ->
{
    //YY = [00;01;10]
    // [00] = 32-48 kHz; [01] = 88-96 kHz;
    // [10] = 176-192 kHz

    c_bits2 &= ~0x0C;                          //vynulování předchozího stavu

    switch (im)
    {
        case 0:                                //INT Mode = 2X
            c_bits2 |= 0x08;                    //Bity = xxxx10xx xxxxxx01
        case 1:                                //INT Mode = 4X
            c_bits2 |= 0x04;                    //Bity = xxxx01xx xxxxxx01
        case 2:                                //INT Mode = 8X
            c_bits2 |= 0x00;                    //Bity = xxxx00xx xxxxxx01
    }

    SPI_Tx(c_bits2,c_bits1); //odeslání dat po SPI
}

void set_serial_mode(unsigned char sdim)      //Bity = xxxxxxxx xxYYxx01 ->
{
    //YY = [00;01;10;11]

    c_bits1 &= ~0x30;                          //vynulování předchozího stavu

    switch (sdim)
    {
        case 0:                                //[IDPM1:IDPM0 == 0:0] -> Right-justified serial mode
            {
                c_bits1 |= 0x00;                //Bity = xxxxxxxx xx00xx01
            }
        case 1:                                //[IDPM1:IDPM0 == 0:1] -> I2S serial mode
            {

```

```

        c_bits1 |= 0x10;          //Bity = xxxxxxxx xx01xx01
    }
    case 2:                       //[IDPM1:IDPM0 == 1:0] -> Left-justified serial mode
    {
        c_bits1 |= 0x20;          //Bity = xxxxxxxx xx10xx01
    }
    case 3:                       //[IDPM1:IDPM0 == 1:1] -> DSP serial mode
    {
        c_bits1 |= 0x30;          //Bity = xxxxxxxx xx11xx01
    }
}

SPI_Tx(c_bits2,c_bits1); //odeslání dat po SPI

}

void set_no_of_bits(unsigned char nob)          //Bity = xxxxxxYY xxxxxx01 ->
                                                //YY = [00;01;10]
{
    c_bits2 &= ~0x03;                      //vynulování předchozího stavu

    switch (nob)
    {
        case 0:                          //[0:0] -> 24 - bit
        {
            c_bits2 |= 0x00;              //Bity = xxxxxx00 xxxxxx01
        }
        case 1:                          //[0:1] -> 20 - bit
        {
            c_bits2 |= 0x01;              //Bity = xxxxxx01 xxxxxx01
        }
        case 2:                          //[1:0] -> 16 - bit
        {
            c_bits2 |= 0x02;              //Bity = xxxxxx10 xxxxxx01
        }
    }

    SPI_Tx(c_bits2,c_bits1); //odeslání dat po SPI

}

void set_deemp_filter(unsigned char df)        //Bity = xxxxxxxx xxxxYY01 ->
                                                //YY = [00;01;10;11]
{
    c_bits1 &= ~0x0C;

    switch (df)
    {
        case 0:                          //[0:0] -> no filter
        {
            c_bits1 |= 0x00;              //Bity = xxxxxxxx xxxx0001
        }
        case 1:                          //[0:1] -> 44.1 kHz filter
        {
            c_bits1 |= 0x04;              //Bity = xxxxxxxx xxxx0101
        }
        case 2:                          //[1:0] -> 32 kHz filter
        {
            c_bits1 |= 0x08;              //Bity = xxxxxxxx xxxx1001
        }
        case 3:                          //[1:1] -> 48 kHz filter
        {
            c_bits1 |= 0x0C;              //Bity = xxxxxxxx xxxx1101
        }
    }

    SPI_Tx(c_bits2,c_bits1);              //odeslání dat po SPI
}

```

```

void mute(unsigned char m)                                     //Bity = xxxxxxxx xlxxxx01 ->
                                                             // [BIT 6 ->> 1 = muted; 0 = unmuted; default = 0]
{
    if (m==1) {c_bits1 |= 0x40;}
    else if (m==0) {c_bits1 &= ~0x40;}

    SPI_Tx(c_bits2,c_bits1);                                //odeslání dat po SPI
}

void soft_reset(void)                                        //Bity = xxxxxxxx lxxxxx01 -> [BIT 7 ->> 1 = RESET active;
                                                             //default = 0]
{
    c_bits1 |= 0x80;
    SPI_Tx(c_bits2,c_bits1);                                //odeslání dat po SPI
    _delay_ms(100);
    c_bits1 &= ~0x80;
    SPI_Tx(c_bits2,c_bits1);                                //odeslání dat po SPI
}

```

## E.2 Ovládací knihovna pro obvod CS8416

```
/* ===== */
/*
/*   Ovladani_DRx.c
/*   (c) 2001 Jan Špaček
/*
/*   Popis knihovny:
/*   Knihovna pro ovládání digitálního přijímače pro audio signál CS8416
/*   (integrovaný obvod od Cirrus Logic). Ovládání CS8416 je prováděno
/*   pomocí mikrokontroléru ATmega32 a komunikačního rozhraní SPI.
/*
/*   Legenda ke všem potřebným funkcím a jejich parametrům se nachází pod
/*   zdrojovým kódem.
/*
/* ===== */

#include "Ovladani_DRx.h"
#include <avr/io.h>
#include <util/delay.h>

char map = 0x00, data = 0x00; // MAP ... memory address pointer (proměnná
// udávající adresu řízeného registru)

// DATA .. pomocná proměnná pro čtení a zápis dat
// pomocí SPI

//-----

void DRx_init(void) //inicializace digitálního přijímače
{
PORT_DRx &= ~(1<<PIN_DRx_RESET); //nulování resetovacího pinu, aktivace
//hardwarového Resetu
_delay_ms(20);
PORT_DRx |= (1<<PIN_DRx_RESET); //nastavení resetovacího pinu, deaktivace
//hardwarového Resetu
PORT_DRx |= (1<<PIN_DRx_CS); //nastavení signálu Chip Select pro rozhraní
//SPI na hodn. log.1
}

void SPI_MasterAndSlaveInit(void) //inicializace SPI rozhraní
{
DDR_SPI = (1<<DD_MOSI) | (1<<DD_SCK) | (0<<DD_MISO) | (0<<DD_SS);
// Nastavení MOSI a SCK jako výstupy, MISO a SS (SlaveSelect) jako vstupy

SPCR = (1<<SPE) | (1<<MSTR) | (1<<CPOL) | (1<<CPHA);
// Povolení SPI jako Master, Nastavení hodin: fck/4, Polarita hodin - log.1, (LSB
first -> DORD = 1) - (MSB first -> DORD = 0)
}

void SPI_Tx(char m, char data) //funkce pro zápis dat do registrů přijímače po SPI
{
PORT_DRx &= ~(1<<PIN_DRx_CS); //Pin CS - překlopení do log.0 (signál
//začátku přenosu)
SPDR = 0b00100000; //první byte, označující zápis do registru
while(!(SPSR & (1<<SPIF))); //čekání na odvysílání slova (8 bitů)
SPDR = m; //druhý byte - MAP (Memory Address
//Pointer)...adresa registru
while(!(SPSR & (1<<SPIF))); //čekání na odvysílání slova (8 bitů)
SPDR = data; //třetí byte - DATA
while(!(SPSR & (1<<SPIF))); //čekání na odvysílání slova (8 bitů)
PORT_DRx |= (1<<PIN_DRx_CS); //Pin CS - překlopení do log.1 (potvrzení
//kompletního přenosu)
_delay_us(100); //čekání 100 us
}
```

```

char SPI_Rx(char m) //funkce pro čtení dat z registrů přijímače po SPI
{
PORT_DRx &= ~(1<<PIN_DRx_CS); //Pin CS - překlopení do log.0 (signál
//začátku přenosu)
SPDR = 0b00100001; //první byte, označující zápis do registru
while(!(SPSR & (1<<SPIF))); //čekání na odvysílání slova (8 bitů)
SPDR = m; //druhý byte - MAP (Memory Address
//Pointer)...adresa registru
while(!(SPSR & (1<<SPIF))); //čekání na odvysílání slova (8 bitů)
while(!(SPSR & (0<<SPIF))); //čekání na příjem slova (8 bitů)
data = SPDR; //zápis přijatého slova do proměnné data
PORT_DRx |= (1<<PIN_DRx_CS); //Pin CS - překlopení do log.1 (potvrzení
//kompletního přenosu)
_delay_us(100); //čekání 100 us
return data;
}

```

//----- FUNKCE OVLÁDÁNÍ DIGITÁLNÍHO AUDIO PŘIJÍMAČE -----

```

void f_ctrl00(char c0) // Funkce pro ovládání registru 0x00
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x00;
ctrl = SPI_Rx(map);

switch (c0)
{
case 0: { ctrl |= 0b01000000; } // FSWCLK
case 1: { ctrl |= 0b00001000; } // PDUR
case 2: { ctrl |= 0b00000100; } // TRUNC
}

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl01(char c1) // Funkce pro ovládání registru 0x01
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x01;
ctrl = SPI_Rx(map);

switch (c1)
{
case 0: { ctrl |= 0b10000000; } // SWCLK
case 1: { ctrl |= 0b01000000; } // MUTESAO
case 2: { ctrl |= 0b00010000; } // INT[0:1]
case 3: { ctrl |= 0b00100000; } // INT[1:0]
case 4: { ctrl |= 0b00110000; } // INT[1:1]
case 5: { ctrl |= 0b00000100; } // HOLD[0:1]
case 6: { ctrl |= 0b00001000; } // HOLD[1:0]
case 7: { ctrl |= 0b00001100; } // HOLD[1:1]
case 8: { ctrl |= 0b00000010; } // RMCKF
case 9: { ctrl |= 0b00000001; } // CHS
}

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl02(char c2) // Funkce pro ovládání registru 0x02
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x02;
ctrl = SPI_Rx(map);

switch (c2)
{
case 0: { ctrl |= 0b10000000; } // DETCI
case 1: { ctrl |= 0b00010000; } // EMPH_CNTL[0:0:1]
}

```

```

case 2: { ctrl |= 0b00100000; } // EMPH_CNTRL[0:1:0]
case 3: { ctrl |= 0b00110000; } // EMPH_CNTRL[0:1:1]
case 4: { ctrl |= 0b01000000; } // EMPH_CNTRL[1:0:0]
case 5: { ctrl |= 0b01010000; } // EMPH_CNTRL[1:0:1]
case 6: { ctrl |= 0b01100000; } // EMPH_CNTRL[1:1:0]
case 7: { ctrl |= 0b01110000; } // EMPH_CNTRL[1:1:1]
case 8: { ctrl |= 0b00000001; } // GPO0SEL[0:0:0:1]
case 9: { ctrl |= 0b00000010; } // GPO0SEL[0:0:1:0]
case 10: { ctrl |= 0b00000011; } // GPO0SEL[0:0:1:1]
case 11: { ctrl |= 0b00000100; } // GPO0SEL[0:1:0:0]
case 12: { ctrl |= 0b00000101; } // GPO0SEL[0:1:0:1]
case 13: { ctrl |= 0b00000110; } // GPO0SEL[0:1:1:0]
case 14: { ctrl |= 0b00000111; } // GPO0SEL[0:1:1:1]
case 15: { ctrl |= 0b00001000; } // GPO0SEL[1:0:0:0]
case 16: { ctrl |= 0b00001001; } // GPO0SEL[1:0:0:1]
case 17: { ctrl |= 0b00001010; } // GPO0SEL[1:0:1:0]
case 18: { ctrl |= 0b00001011; } // GPO0SEL[1:0:1:1]
case 19: { ctrl |= 0b00001100; } // GPO0SEL[1:1:0:0]
case 20: { ctrl |= 0b00001101; } // GPO0SEL[1:1:0:1]
case 21: { ctrl |= 0b00001110; } // GPO0SEL[1:1:1:0]
case 22: { ctrl |= 0b00001111; } // GPO0SEL[1:1:1:1]
}

```

```

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl03 (char c3) // Funkce pro ovládání registru 0x03
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x03;
ctrl = SPI_Rx(map);

```

```

switch (c3) {
case 0: { ctrl |= 0b00010000; } // GPO1SEL[0:0:0:1]
case 1: { ctrl |= 0b00100000; } // GPO1SEL[0:0:1:0]
case 2: { ctrl |= 0b00110000; } // GPO1SEL[0:0:1:1]
case 3: { ctrl |= 0b01000000; } // GPO1SEL[0:1:0:0]
case 4: { ctrl |= 0b01010000; } // GPO1SEL[0:1:0:1]
case 5: { ctrl |= 0b01100000; } // GPO1SEL[0:1:1:0]
case 6: { ctrl |= 0b01110000; } // GPO1SEL[0:1:1:1]
case 7: { ctrl |= 0b10000000; } // GPO1SEL[1:0:0:0]
case 8: { ctrl |= 0b10010000; } // GPO1SEL[1:0:0:1]
case 9: { ctrl |= 0b10100000; } // GPO1SEL[1:0:1:0]
case 10: { ctrl |= 0b10110000; } // GPO1SEL[1:0:1:1]
case 11: { ctrl |= 0b11000000; } // GPO1SEL[1:1:0:0]
case 12: { ctrl |= 0b11010000; } // GPO1SEL[1:1:0:1]
case 13: { ctrl |= 0b11100000; } // GPO1SEL[1:1:1:0]
case 14: { ctrl |= 0b11110000; } // GPO1SEL[1:1:1:1]
case 15: { ctrl |= 0b00000001; } // GPO2SEL[0:0:0:1]
case 16: { ctrl |= 0b00000010; } // GPO2SEL[0:0:1:0]
case 17: { ctrl |= 0b00000011; } // GPO2SEL[0:0:1:1]
case 18: { ctrl |= 0b00000100; } // GPO2SEL[0:1:0:0]
case 19: { ctrl |= 0b00000101; } // GPO2SEL[0:1:0:1]
case 20: { ctrl |= 0b00000110; } // GPO2SEL[0:1:1:0]
case 21: { ctrl |= 0b00000111; } // GPO2SEL[0:1:1:1]
case 22: { ctrl |= 0b00001000; } // GPO2SEL[1:0:0:0]
case 23: { ctrl |= 0b00001001; } // GPO2SEL[1:0:0:1]
case 24: { ctrl |= 0b00001010; } // GPO2SEL[1:0:1:0]
case 25: { ctrl |= 0b00001011; } // GPO2SEL[1:0:1:1]
case 26: { ctrl |= 0b00001100; } // GPO2SEL[1:1:0:0]
case 27: { ctrl |= 0b00001101; } // GPO2SEL[1:1:0:1]
case 28: { ctrl |= 0b00001110; } // GPO2SEL[1:1:1:0]
case 29: { ctrl |= 0b00001111; } // GPO2SEL[1:1:1:1]
}

```

```

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl04(char c4) // Funkce pro ovládání registru 0x04
char ctrl = 0x00; // vynulování předchozího stavu

```

```

{
map = 0x04;
ctrl = SPI_Rx(map);

switch (c4) {
case 0: { ctrl |= 0b10000000; } // RUN
case 1: { ctrl |= 0b01000000; } // RXD
case 2: { ctrl |= 0b00001000; } // RXSEL[0:0:1]
case 3: { ctrl |= 0b00010000; } // RXSEL[0:1:0]
case 4: { ctrl |= 0b00011000; } // RXSEL[0:1:1]
case 5: { ctrl |= 0b00100000; } // RXSEL[1:0:0]
case 6: { ctrl |= 0b00101000; } // RXSEL[1:0:1]
case 7: { ctrl |= 0b00110000; } // RXSEL[1:1:0]
case 8: { ctrl |= 0b00111000; } // RXSEL[1:1:1]
case 9: { ctrl |= 0b00000001; } // TXSEL[0:0:1]
case 10: { ctrl |= 0b00000010; } // TXSEL[0:1:0]
case 11: { ctrl |= 0b00000011; } // TXSEL[0:1:1]
case 12: { ctrl |= 0b00000100; } // TXSEL[1:0:0]
case 13: { ctrl |= 0b00000101; } // TXSEL[1:0:1]
case 14: { ctrl |= 0b00000110; } // TXSEL[1:1:0]
case 15: { ctrl |= 0b00000111; } // TXSEL[1:1:1]
}

SPI_Tx(map,ctrl);
}

void f_ctrl05(char c5) // Funkce pro ovládání registru 0x05
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x05;
ctrl = SPI_Rx(map);

switch (c5) {
case 0: { ctrl |= 0b10000000; } // SOMS
case 1: { ctrl |= 0b01000000; } // SOSF
case 2: { ctrl |= 0b00010000; } // SORES[0:1]
case 3: { ctrl |= 0b00100000; } // SORES[1:0]
case 4: { ctrl |= 0b00110000; } // SORES[1:1]
case 5: { ctrl |= 0b00001000; } // SOJUST
case 6: { ctrl |= 0b00000100; } // SODEL
case 7: { ctrl |= 0b00000010; } // SOSPOL
case 8: { ctrl |= 0b00000001; }
}

SPI_Tx(map,ctrl);
}

void f_ctrl06(char c6) // Funkce pro ovládání registru 0x06
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x06;
ctrl = SPI_Rx(map);

switch (c6) {
case 0: { ctrl |= 0b01000000; } // QCRCM
case 1: { ctrl |= 0b00100000; } // CCRCM
case 2: { ctrl |= 0b00010000; } // UNLOCKM
case 3: { ctrl |= 0b00001000; } // VM
case 4: { ctrl |= 0b00000100; } // CONFM
case 5: { ctrl |= 0b00000010; } // BIPM
case 6: { ctrl |= 0b00000001; } // PARM
}

SPI_Tx(map,ctrl);
}

void f_ctrl07(char c7) // Funkce pro ovládání registru 0x07
char ctrl = 0x00; // vynulování předchozího stavu

```

```

{
map = 0x07;
ctrl = SPI_Rx(map);

switch (c7) {
case 0: { ctrl |= 0b01000000; } // PCCHM
case 1: { ctrl |= 0b00100000; } // OSLIPM
case 2: { ctrl |= 0b00010000; } // DETCM
case 3: { ctrl |= 0b00001000; } // CCHM
case 4: { ctrl |= 0b00000100; } // RERRM
case 5: { ctrl |= 0b00000010; } // QCHM
case 6: { ctrl |= 0b00000001; } // FCHM
}

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl08(char c8) // Funkce pro ovládání registru 0x08
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x08;
ctrl = SPI_Rx(map);

switch (c8) {
case 0: { ctrl |= 0b01000000; } // PCCH1
case 1: { ctrl |= 0b00100000; } // OSLIP1
case 2: { ctrl |= 0b00010000; } // DETC1
case 3: { ctrl |= 0b00001000; } // CCH1
case 4: { ctrl |= 0b00000100; } // RERR1
case 5: { ctrl |= 0b00000010; } // QCH1
case 6: { ctrl |= 0b00000001; } // FCH1
}

SPI_Tx(map,ctrl);
}

```

```

void f_ctrl09(char c9) // Funkce pro ovládání registru 0x09
char ctrl = 0x00; // vynulování předchozího stavu
{
map = 0x09;
ctrl = SPI_Rx(map);

switch (c9) {
case 0: { ctrl |= 0b01000000; } // PCCH0
case 1: { ctrl |= 0b00100000; } // OSLIP0
case 2: { ctrl |= 0b00010000; } // DETC0
case 3: { ctrl |= 0b00001000; } // CCH0
case 4: { ctrl |= 0b00000100; } // RERR0
case 5: { ctrl |= 0b00000010; } // QCH0
case 6: { ctrl |= 0b00000001; } // FCH0
}

SPI_Tx(map,ctrl);
}

```



## E.3 Řídící program mikrokontroléru pro ATmega32

```
/* ===== */
/*
/* projekt.c */
/* (c) 2011 Jan Špaček */
/*
/* Popis programu: */
/* Ovládací program pro D/A převodník (implicitně programováno pro D/A */
/* převodník AD1852 od Analog Devices). Ovládání AD1852 je prováděno */
/* mikrokontrolérem ATmega32 pomocí rozhraní SPI. Dále se vypisují */
/* veškeré funkce a nastavované parametry na LCD. Elementární funkce jsou */
/* brány z knihoven "Ovladani_DAC.c" a "Ovladani_DAC.h" . */
/*
/* ===== */

#include "Ovladani_DAC.h"
#include "Ovladani_DRX.h"
#include <avr/io.h>
#include <util/delay.h>
#include <stdio.h>
#include <stdlib.h>
#include "lcd.h"

int main(void)
{
DAC_init(); //inicializace DA převodníku
DRX_init();
SPI_MasterInit(); //inicializace SPI rozhraní

//----- deklarace proměnných -----

int a = -1;
char x = 0;
char y = 0;

char d = 0;
char e = 0;
char f = 0;
char g = 0;
char h = 0;
char i = 0;
char j = 0;
char k = 0;
char l = 0;
char m = 0;
char n = 0;
char o = 0;
char p = 0;

//-----

char buf [5]; //pomocná proměnná

DDRB &= 0b11110000;
PORTB |= 0b00001111;

//----- inicializace LCD -----

lcd_init(LCD_DISP_ON);
lcd_clrscr();

lcd_gotoxy(0,0);
lcd_puts("Control APP for");
lcd_gotoxy(0,1);
lcd_puts("DAC and DRX ");
```

```

//-----

//----- JÁDRO OVLÁDACÍ APLIKACE -----

while(1)
{

//----- obsluha tlačítek -----

if(bit_is_clear(PINB,0)) {a++; if (a==19) a=0;}; //změna 19ti funkcí ->
if(bit_is_clear(PINB,1)) {a--; if (a==-1) a=18;}; //změna 19ti funkcí <-

if(bit_is_clear(PINB,2)) x=1; //tlačítko změny nastavení ->
else x=0;
if(bit_is_clear(PINB,3)) y=1; //tlačítko změny nastavení <-
else y=0;

//----- výpis menu na LCD -----

lcd_gotoxy(0,0);
switch(a){
case 0: {lcd_puts("1.VOLUME LEFT: "); //FUNKCE VOLUME LEFT

if(x==1) { if(d!=255) d+=5;}; //volume 0-255 s krokem 5
if(y==1) { if(d!=0) d-=5;};

lcd_gotoxy(0,1);
lcd_puts("Level - ");
itoa(d, buf, 10); //převod čísla hlasitosti na string
lcd_puts(" ");
lcd_gotoxy(8,1);
lcd_puts(buf);

set_volume_left(d); //volání funkce z ovládací knihovny

};
break;

case 1: {lcd_puts("2.VOLUME RIGHT:"); //FUNKCE VOLUME RIGHT

if(x==1) { if(e!=255) e+=5; }; //volume 0-255 s krokem 5
if(y==1) { if(e!=0) e-=5; };

lcd_gotoxy(0,1);
lcd_puts("Level - ");
itoa(e, buf, 10); //převod čísla hlasitosti na string
lcd_puts(" ");
lcd_gotoxy(8,1);
lcd_puts(buf);

set_volume_right(e); //volání funkce z ovládací knihovny

};
break;

case 2: {lcd_puts("3.INT Mode: "); //FUNKCE INT MODE

if(x==1) {f++; if(f==3) f=0;};
if(y==1) {f--; if(f==-1) f=2; };

lcd_gotoxy(0,1);
switch(f){
case 0: lcd_puts(" 2x "); break;
case 1: lcd_puts(" 4x "); break;
case 2: lcd_puts(" 8x "); break;
}
}
}
}

```

```

};

set_int_mode(f);      //volání funkce z ovládací knihovny

break;

case 3: {lcd_puts("4.Serial Mode: ");      //FUNKCE SERIAL MODE

if(x==1) {g++; if(g==4) g=0;};
if(y==1) {g--; if(g==-1) g=3; };

lcd_gotoxy(0,1);
switch(g) {
case 0: lcd_puts("Right-Justif. "); break;
case 1: lcd_puts("I2S "); break;
case 2: lcd_puts("Left-Justif. "); break;
case 3: lcd_puts("DSP "); break;
}
};

set_serial_mode(g);  //volání funkce z ovládací knihovny

break;

case 4: {lcd_puts("5.NumberOfBits:");      //FUNKCE NUMBER OF BITS

if(x==1) {h++; if(h==3) h=0;};
if(y==1) {h--; if(h==-1) h=2; };

lcd_gotoxy(0,1);
switch(h) {
case 0: lcd_puts(" 24 bit "); break;
case 1: lcd_puts(" 20 bit "); break;
case 2: lcd_puts(" 16 bit "); break;
}
};

set_no_of_bits(h);   //volání funkce z ovládací knihovny

break;

case 5: {lcd_puts("6.Deemp.filter:");      //FUNKCE DEEMPHASIS FILTER

if(x==1) {i++; if(i==4) i=0;};
if(y==1) {i--; if(i==-1) i=3; };

lcd_gotoxy(0,1);
switch(i) {
case 0: lcd_puts("no filter "); break;
case 1: lcd_puts("44.1 kHz "); break;
case 2: lcd_puts(" 32 kHz "); break;
case 3: lcd_puts(" 48 kHz "); break;
}
};

set_deemp_filter(i); //volání funkce z ovládací knihovny

break;

case 6: {lcd_puts("7.Mute: ");      //FUNKCE MUTE

if(x==1) {j++; if(j==2) j=0;};
if(y==1) {j--; if(j==-1) j=1; };

lcd_gotoxy(0,1);
switch(j) {
case 0: lcd_puts("un-muted "); break;
case 1: lcd_puts("muted "); break;
}
};

mute(j);      //volání funkce z ovládací knihovny

break;

```

```

case 7: {lcd_puts("8.Soft Reset: "); //FUNKCE SOFT RESET
        if(x==1) {k++; if(k==2) k=0;};

        lcd_gotoxy(0,1);
        switch(k) {
            case 0: lcd_puts("press button->"); break;
            case 1: { lcd_puts("reseting... ");

soft_reset(); //volání funkce z ovládací knihovny

                _delay_ms(2000);
                lcd_gotoxy(0,1);
                lcd_puts("reseted! ");
                _delay_ms(1000);
                k = 0;
            }; break;

        }

};

break;

case 8: {lcd_puts("9.Default Val. "); //FUNKCE SET DEFAULT VALUES

        if(x==1) {l++; if(l==2) l=0;};

        lcd_gotoxy(0,1);
        switch(l) {
            case 0: lcd_puts("press button->"); break;
            case 1: { lcd_puts("setting... ");
                _delay_ms(1000);

//nastavení všech proměnných na defaultní hodnoty
                a = -1;
                x = 0;
                y = 0;

                d = 0;
                e = 0;
                f = 0;
                g = 0;
                h = 0;
                i = 0;
                j = 0;
                k = 0;
                m = 0;
                n = 0;
                o = 0;
                p = 0;

                lcd_gotoxy(0,1);
                lcd_puts("default set ");
                _delay_ms(1000);
                l = 0;

}; break;

        }

}; break;

case 9: {lcd_puts("10.OMCK ") //FUNKCE ZAPNUTI HOD. SIGNALU
        if(x==1) {m++; if(l==2) m=0;};
            if(y==1) {m--; if(l==-1) m=1; };
        lcd_gotoxy(0,1);
        switch(m) {
            case 0: lcd_puts("signal ON "); break;
            case 1: lcd_puts("signal OFF "); break;

        }
        f_ctrl100(m);
    }

```

```

        case 10: {lcd_puts("11.RMCKFreq  ") //FUNKCE PREDEL. VYST. HODIN
if(x==1) {n++; if(n==2) n=0;};
if(y==1) {n--; if(n==-1) n=1};
lcd_gotoxy(0,1);
switch(n); {
case 0; lcd_puts("256 * Fs      "); break;
case 1; lcd_pust("128 * Fs      "); break;
}
f_ctrl101(n);
}

case 11: {lcd_puts("12.INPUT Rx  ") //FUNKCE VYBER VSTUPU
if(x==1) {o++; if(o==8) n=0;};
if(y==1) {o--; if(o==-1) n=7};
lcd_gotoxy(0,1);
switch(o); {
case 0; lcd_puts("inp. no.01  "); break;
case 1; lcd_puts("inp. no.02  "); break;
case 2; lcd_puts("inp. no.03  "); break;
case 3; lcd_puts("inp. no.04  "); break;
case 4; lcd_puts("inp. no.05  "); break;
case 5; lcd_puts("inp. no.06  "); break;
case 6; lcd_puts("inp. no.07  "); break;
case 7; lcd_puts("inp. no.08  "); break;
}
f_ctrl102(o);
}

case 12: {lcd_puts("12.BIT Res.  ") //FUNKCE VYBER BIT. ROZLISENI
if(x==1) {p++; if(p==4) n=0;};
if(y==1) {p--; if(p==-1) n=3};
lcd_gotoxy(0,1);
switch(o); {
case 0; lcd_puts("24 bit - res.  "); break;
case 1; lcd_puts("20 bit - res.  "); break;
case 2; lcd_puts("16 bit - res.  "); break;
case 3; lcd_puts("AES3 - NRZ   "); break;
}
f_ctrl105(p);
}
}

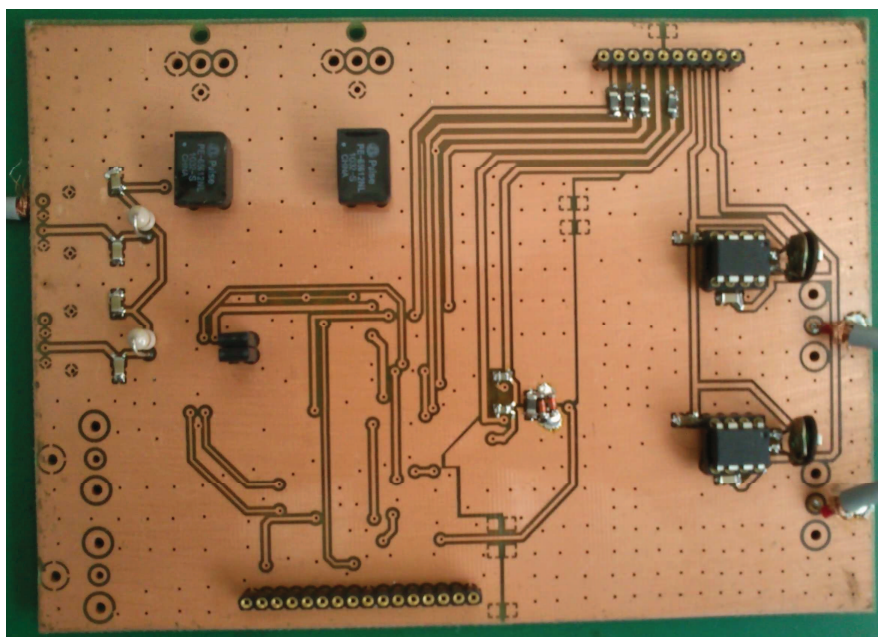
_delay_ms(200); //zpoždění kvůli stisku tlačítek

//-----
}
}

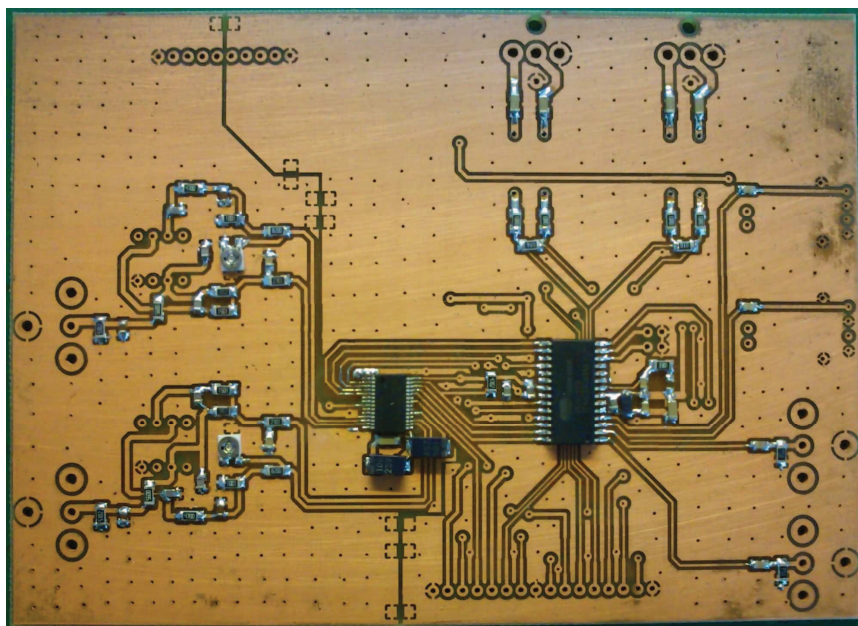
```

## F FOTODOKUMENTACE

### F.1 Deska A – strana TOP

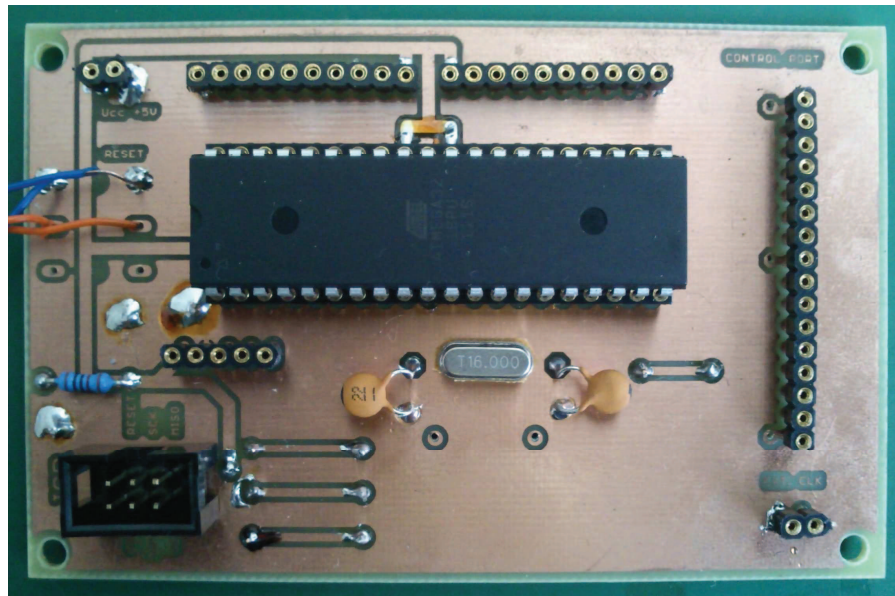


### F.2 Deska A – strana BOTTOM

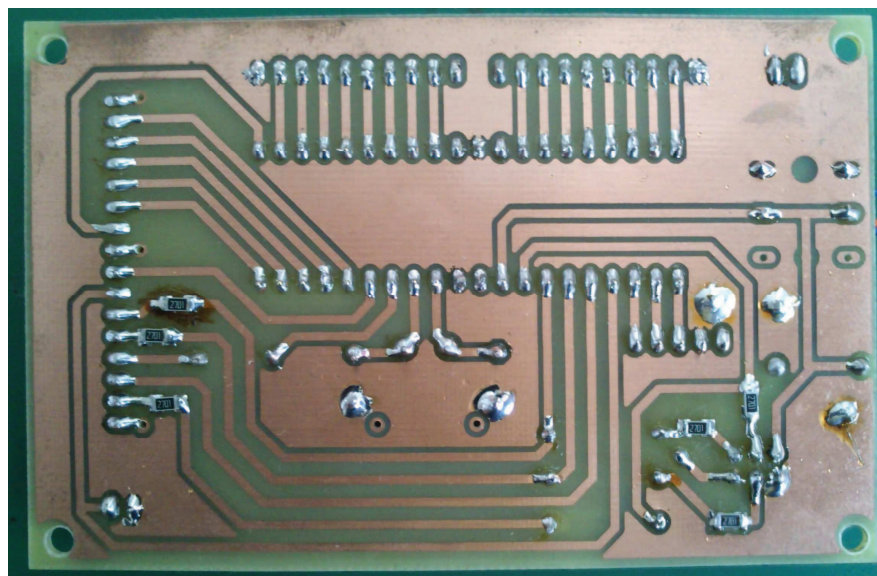




### F.3 Deska B – strana TOP



### F.4 Deska B – strana BOTTOM



# G CD S DOKUMENTACÍ

## Obsah CD:

- Elektronická verze dokumentu (formát PDF)
- Kompletní obvodové schéma zapojení (obrázek)
- Soubory s DPS vytvořené v programu EAGLE (formát SCH a BRD)
- Soubory s exportem z Post-Scriptu (formáty PS a PDF)
- Soupisky součástek vygenerované programem EAGLE (formát TXT)
- Zdrojové kódy řídicího programu mikrokontroléru (formáty C a H)
- Soubory fotodokumentace (obrázky)