

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

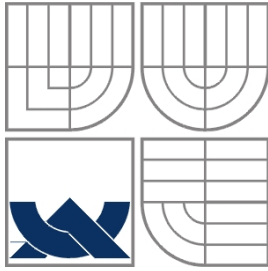
ROZŠIŘUJÍCÍ MODUL S DEKODÉREM SOUBORŮ
MP3 PRO PLATFORMU FITKIT

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

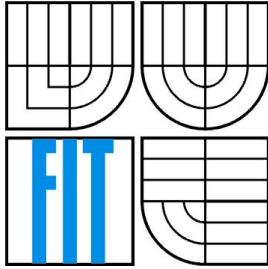
AUTOR PRÁCE
AUTHOR

MIROSLAV BRADA

BRNO 2012



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

ROZŠIŘUJÍCÍ MODUL S DEKODÉREM SOUBORŮ MP3 PRO PLATFORMU FITKIT

EXPANSION MODULE WITH MP3 FILES DECODER FOR FITKIT PLATFORMA

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

MIROSLAV BRADA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. VÁCLAV ŠIMEK

BRNO 2012

Abstrakt

Cílem tohoto projektu je vytvoření modulu, který by rozšířil možnosti platformy FITkit o dekodování souborů formátu MP3. Práce shrnuje poznatky o současných formátech kódování zvuku, dostupných dekodérech a platformě FITkit. V pozdější části je popsán postup návrhu rozšiřujícího modulu této platformy, který byl nakonec zrealizován i když neúspěšně. Modul využívá kodek VS1063. V závěru jsou popsány možnosti dalšího rozšíření projektu v případě úspěšné realizace.

Abstract

Aim of this project is building of expansion module which expands capabilities of FITkit platform by allowing decoding of MP3 files. This thesis summarize knowledge about current audio encoding formats, available decoders and FITkit platform. Later part of thesis describes design procedure of expansion module for this platform, which was completed, although unsuccessfully. Module is using VS1063 codec. The end of thesis describes possible expansions in case of successful implementation.

Klíčová slova

MP3, dekodér, kodek, rozšiřující modul, FITkit, audio kódování, audio formáty, VS1063, VLSI

Keywords

MP3, decoder, codec, expansion module, FITkit, audio encoding, audio formats, VS1063, VLSI

Citace

Miroslav Brada: Rozšiřující modul s dekodérem souborů MP3 pro platformu FITkit, bakalářská práce, Brno, FIT VUT v Brně, rok 2012

Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana Ing. Václava Šimka.

Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Miroslav Brada
10.5.2012

Pod'akovanie

Chcel by som poďakovať môjmu vedúcemu pánovi Ing. Václavovi Šimkovi za jeho odbornú asistenciu pri fyzickej výrobe modulu, ako aj za cenné rady a pripomienky pri jeho návrhu.

Obsah

1 Úvod	2
2 Teoretické poznatky	3
2.1 Kompresia zvuku	3
2.1.1 Bezstratová (lossless) kompresia zvuku	3
2.1.2 Stratová (lossy) kompresia zvuku	4
2.1.3 MP3	5
2.1.4 Dostupné riešenia dekodéru súborov MP3	7
2.2 Platforma FITkit	9
2.2.1 FPGA	10
2.2.2 MCU	10
2.2.3 Komunikačné rozhranie MCU a FPGA	11
2.3 Pamäťová karta SD	13
3 Realizácia modulu	15
3.1 Popis funkcie aplikácie	15
3.2 MP3 kódek	16
3.2.1 Popis rozhrania čipu VS1063a	16
3.2.2 Popis činnosti čipu VS1063a	18
3.3 Doska plošných spojov modulu	19
3.3.1 Analógová časť	19
3.3.2 Napájanie dosky plošných spojov	20
3.3.3 Pripojenie karty SD	21
3.3.4 Výroba dosky plošných spojov	22
3.4 Konfigurácia FPGA	23
3.4.1 Maticová klávesnica	24
3.4.2 LCD displej	24
3.4.3 Prepojenie modulu a MCU	25
3.5 Implementácia programu pre MCU a funkčný popis aplikácie	25
3.5.1 Inicializačná časť programu pre MCU	25
3.5.2 Spracovávanie užívateľského vstupu	26
4 Záver	27
Použitá literatúra	28
Príloha A: Schéma zapojenia modulu	30
Príloha B: Obrázok dosky plošných spojov	31
Príloha C: CD so zdrojovými súborami	32

1 Úvod

Formát MP3 patrí v súčasnosti k najpopulárnejším a najrozšírenejším formátom kódovania zvuku. Tento formát umožňuje ukladanie zvuku v dostatočnej kvalite pri pomerne vysokom kompresnom pomere. Dekódovanie formátu MP3 je možné realizovať ako hardwarovo alebo softwarovo. U bežných osobných počítačov sa používa softwarové dekodovanie avšak u širokej škály aplikácií je vhodné použiť hardwarové dekodovanie. Typickým príkladom sú prenosné MP3 prehrávače alebo väčšina mobilných telefónov. V týchto prípadoch sa používa špecializovaný integrovaný obvod nazývaný MP3 kódek, na ktorý sa presunie výpočtovo zložitejšie dekodovanie (prípadne kódovanie) formátu MP3.

Cieľom tejto bakalárskej práce je navrhnúť a následne zrealizovať modul pre platformu FITkit, ktorý bude umožňovať dekodovanie formátu MP3, rozšírený o možnosť kódovania externého analógového zvuku. Hlavným prínosom by mala byť možnosť jednoduchého pridania podpory MP3 v budúcich aplikáciách na platforme FITkit pomocou tohto modulu. Toto môže značne zjednodušiť vývoj množstva aplikácií využívajúcich reprodukciu alebo nahrávanie zvuku.

V prvej kapitole sú zhrnuté teoretické poznatky o možnostiach kódovania zvuku, špeciálne o formáte MP3. Ďalej nasleduje zhrnutie poznatkov o platforme FITkit, možnostiach komunikácie jednotlivých komponentov v rámci danej platformy a stručný popis pamäťových kariet formátu SD. V nasledujúcej kapitole sa nachádzajú implementačné detaily navrhovaného modulu, popisy zapojenia a detailný popis vytvoreného obvodu modulu. Po tejto kapitole nasleduje popis konkrétnej implementácie na platforme FITkit a prípadné možnosti rozšírenia. Ďalej už nasleduje posledná kapitola - záver, kde sú zhrnuté výsledky práce a možné rozšírenia.

2 Teoretické poznatky

V tejto kapitole budú zhrnuté teoretické podklady potrebné pre samotnú implementáciu modulu. Kapitola je rozdelená do troch podkapitol v ktorých je postupne rozobraná kompresia zvuku, vrátane krátkeho popisu dostupných integrovaných obvodov použiteľných pre dekódovanie súborov formátu MP3, platforma FITkit a štandard pamäťových SD kariet.

2.1 Kompresia zvuku

Potreba kompresie zvuku vznikla z dôvodu veľkého obsadenia dátových liniek pri prenose zvuku a taktiež z dôvodu obmedzenia veľkosti súborov v prípade uloženia zvuku a nahrávok. Bežné kompresné algoritmy, ako napríklad DEFLATE využívaný vo formáte .zip, nedosahujú dostatočné kompresné pomery. Príčinou je fakt, že v prípade zvukových stôp sa jedná zväčša o digitálnu konverziu pôvodnej analógovej nahrávky, a tiež prítomnosť značného faktora náhodnosti v nahrávkach. Tieto bežné algoritmy taktiež nie sú určené pre aplikácie bežiacie v reálnom čase, čo je u kompresie zvuku často veľmi podstatné. Z tohto dôvodu boli vyvinuté špeciálne algoritmy pre potreby kompresie zvuku. Z širšieho hľadiska ich môžeme rozdeliť na stratové (lossy) a bezstratové (lossless). [1]

2.1.1 Bezstratová (lossless) kompresia zvuku

Bezstratová kompresia zvuku využíva metódy kompresie, ktoré umožňujú následnú dekompresiu záznamu do pôvodnej podoby pred kompresiou. Nevýhodou sú hlavne pomerne slabé kompresné pomery oproti stratovej kompresii (avšak je to stále lepšie riešenie než bežné kompresné algoritmy). Využíva sa napríklad pri ukladaní pôvodných nahrávok a ich následnej štúdiovej úprave, aby sa predišlo strate kvality pôvodnej nahrávky, a taktiež pri ukladaní hlavnej nahrávky, v prípade masovej produkcie audio nahrávok, distribuovaných v stratových formátoch. Široké využitie nachádza aj v prípade potreby vysoko kvalitnej reprodukcie zvuku.

Existuje viacero bezstratových formátov, medzi najznámejšie patria FLAC (Free Lossless Audio Codec), ALAC (Apple Lossless Audio Codec), niekoľko kódekov zo štandardu MPEG4 part 3, TTA (True Audio codec), WMA 9 Lossless, a ďalšie. Ich porovnanie je možné napríklad vzhľadom na rýchlosť kompresie a dekompresie, alebo kompresného pomeru. Príklad takéhoto porovnania je v Tabuľke 1. Témou tohto konkrétneho porovnania bola rýchlosť kompresie a výsledná veľkosť súboru u rôznych spôsobov kompresie tej istej nahrávky. Je z neho viditeľný rozdiel vo veľkosti súboru

komprimovaného stratovou kompresiou (MP3, Ogg Vorbis), súboru komprimovaného bezstratovou kompresiou (FLAC a Monkey's Audio) a súboru formátu Wave bez kompresie, ale aj v rýchlosti kompresie.

Format.	Output size, bytes.	Ripping time, minutes seconds.
Wave	775,789,440	5m27s
FLAC	513,904,545	6m58s
Monkey's Audio	495,419,947	7m18s
MP3	140,795,904	11m53s
Ogg Vorbis	133,066,417	13m30s

Tabuľka 1: Porovnanie vybraných bezstratových a stratových formátov pri kompresii zvuku. (prebraté a upravené zo stránky <http://www.bobulous.org.uk/misc/audioFormats.html>)

Niektoré bezstratové formáty umožňujú hybridnú kompresiu kde vznikajú dve vrstvy. V prvej vrstve je uložená nahrávka komprimovaná stratovou kompresiou a v druhej vrstve sú uložené korekcie, ktoré umožňujú v kombinácii s prvou bezstratovou dekompresiou. Môžu byť uložené v rôznych súboroch (napr. OptimFrog alebo WavPack hybrid) alebo byť zapuzdrené v rámci jedného (napr. u MPEG4-SLS v obale MP4). Ich výhodou je veľmi jednoduchá konverzia na značne menšiu nahrávku, komprimovanú iba stratovou kompresiou, jednoduchým odstránením korekčnej vrstvy.

2.1.2 Stratová (lossy) kompresia zvuku

Zvuková nahrávka komprimovaná stratovou kompresiou nie je po následnej dekompresii identická (na rozdiel od bezstratovej kompresie), avšak v ideálnom prípade by ľudský sluch nemal vnímať rozdiel. Pri kompresii sa využíva psychoakustika, čo je náuka o vnímaní zvuku človekom po psychologickú a fyziologickú stránku. Z hľadiska kompresie je nutné odstrániť čo najviac informácií, ktoré nie je schopné ľudské ucho prijať. Môžu to byť napríklad príliš vysoké alebo nízke frekvencie, zvuky maskované výraznejšími zvukmi, zvuky pod absolútnym prahom počuteľnosti a podobne. Stratová kompresia dosahuje značne lepšie kompresné pomery oproti bezstratovej, pri zachovaní dostatočnej kvality (obvykle približne 3-12 krát lepšie, v závislosti od konkrétnej nahrávky). Napriek tomu k zníženiu kvality predsa len dochádza, preto sa stratová kompresia nevyužíva tam, kde sú na kvalitu nahrávky kladené vysoké nároky.

Pre stratovú kompresiu sa používa množstvo rôznych algoritmov, často špecializovaných na určitý druh nahrávky. V prípade hlasových nahrávok, ktoré sú špecifické úzkym frekvenčným spektrom, častými pauzami a nižšími nárokmi na kvalitu, sa využívajú napríklad algoritmy kódekov AMR, jeho úpravy AMR WB, WB+, algoritmy kódeku CELP (súčasť štandardu MPEG4 part 3) a ďalších. Pri hudobných nahrávkach, ktoré majú naopak veľký frekvenčný rozsah a prekrývajúce sa

zvuky, je zase potrebné využiť iné algoritmy, líšiace sa medzi sebou kvalitou reprodukovateľného zvuku, rýchlosťou kompresie, rozsahom bitrate (bitových tokov), ktoré sú schopné spracovať, prístupom ku kompresii stereo zvuku a množstvom ďalších faktorov. Medzi najznámejšie kódeky z tejto kategórie patria audio kódeky štandardov MPEG-1, MPEG-2 a MPEG-4, MP3 špeciálne popísaný v nasledujúcej kapitole, jeho následník AAC (súčasť MPEG-4 part 3), ďalej Windows Media Audio, open source kódek Vorbis a rýchly Musepack. Pre nahrávky zvuku ku kvalitnej reprodukcii filmov je zase vhodné použiť iné algoritmy, zohľadňujúce požiadavky na kvalitu takéhoto druhu nahrávok, napríklad technológie Dolby Digital.

2.1.3 MP3

Keďže pre našu aplikáciu je najpodstatnejší formát MP3, táto podkapitola bude venovaná podrobnejšiemu popisu tohto formátu.

Ako MP3 je označovaný layer 3 audio časti štandardu MPEG-1 rozšírený štandardom MPEG-2. Tieto štandardy vytvorila skupina Moving Picture Experts Group. MPEG-1 audio layer 3 bol publikovaný v roku 1993 a jeho rozšírenie, ktoré je súčasťou štandardu MPEG-2, v roku 1995.

Audio časť MPEG-1 je rozdelená do troch vrstiev (layerov) s narastajúcou zložitou, ale aj kompresným pomerom a kvalitou reprodukovanej nahrávky. Každá ďalšia vrstva je rozšírením predchádzajúcej, preto zdieľajú veľkú časť procesov kódovania a dekódovania. Toto riešenie zjednodušuje prácu pri implementácii, ale zároveň je to dôvod prečo musia byť niektoré pokročilé vlastnosti MP3 značne zložitejšie ako keby neboli postavené na nižších vrstvách. Toto sa MPEG snažila vyriešiť neskorším formátom AAC, ktorý je určený ako náhrada MP3, avšak nie je viazaný na žiadne iné vrstvy.

Princíp fungovania MP3 kódovania je pomerne jednoduchý. Najskôr sa nahrávka rozdelí na množstvo veľmi krátkych úsekov, ktoré sú ďalej spracovávané samostatne. Každý úsek sa prevedie z časovej domény do domény frekvenčnej, a následne sa pomocou psychoakustického modelu odoberie čo najviac informácií, ktoré by ľudský sluch nemal byť schopný zachytiť. Následne sa zvyšné dáta zakódujú pomocou Huffmanovho kódu a po pridaní hlavičky na začiatok každého úseku, ktorá obsahuje informácie o konkrétnych parametroch kódovania (ako napríklad bitrate, spôsob kódovania stereo zvuku a podobne vid' Obrázok 2), sa zapíšu (vzniká tak vlastne množstvo nezávislých rámcov). Príklad takého súboru je na Obrázku 1. Pred samotnými dátami sa môžu ešte nachádzať metadáta o nahrávke, napríklad v podobe ID3 tagov (označení), ktoré môžu obsahovať údaje o umelcovi, albume, z ktorého nahrávka pochádza, žánri a podobne. Špecifikácia ID3 existuje v dvoch verziách - ID3v1 a ID3v2, pričom nemajú skoro nič spoločné. ID3v1 má napríklad fixnú dĺžku, zatiaľ čo ID3v2 nie, a teda umožňuje uložiť značne viac informácií o nahrávke.

Example
MP3 Header

F F F B A 0 4 0

Colour-coding shows binary bit mapping to hex values below

Detail of an
MP3 Header

Bits	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
Binary	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	0	1	0
Hex	F			F			F			B			A							
Meaning	MP3 Sync Word												Version		Layer		Error Protection		Bit Rate	
Value	Sync Word												1 = MPEG		01 = Layer 3		1 = No		1010 = 160	

21	22	23	24	25	26	27	28	29	30	31	32
0	0	0	0	0	1	0	0	0	0	0	0
Frequency		Pad. Bit	Priv. Bit	Mode		Mode Extension (Used With Joint Stereo)		Copy	Original	Emphasis	
00 = 44100 Hz		0 = Frame is not padded	Unknown	01 = Joint Stereo		0 = Intensity Stereo Off	0 = MS Stereo Off	0 = Not Copy-righted	0 = Copy Of Original Media	00 = None	

Obrázok 1: Príklad formátu hlavičky rámca v súbore MP3 (ilustrácia prebraná a upravená zo <http://en.wikipedia.org/wiki/Mp3>)

Dekódovanie MP3 prebieha presne obrátene, identifikuje sa rámec, z hlavičky rámca sa prečítajú údaje o kódovaní, dáta sa dekodujú podľa Huffmanovho kódovania, zrekonštruujú sa na základe informácií z hlavičky, a nakoniec sa prevedú z frekvenčnej do časovej domény. Týmto vznikne krátky úsek nahrávky, ktorý je možné prehrať. Pri prehrávaní celej nahrávky sa len jednoducho prehrávajú po sebe nasledujúce rámce. Preskakovanie na určitý úsek nahrávky je možné vďaka tomu, že hlavička obsahuje synchronizačnú časť, pomocou ktorej prehrávač vie určiť začiatok rámca. Z toho vyplýva že dekodovať možno aj poškodenú alebo nesúvislú nahrávku, pretože prehrávač môže preskočiť nezmyselné dáta a nájsť najbližší rámec, od ktorého prehrávanie pokračuje. [2],[3]

Za zmienku ešte stojí spôsob kódovania stereo zvuku. MP3 môže fungovať v štyroch spôsoboch spracovania stereo informácie: *mono*, *stereo*, *dual channel* a *joint stereo*. Keďže každý rámec je samostatná jednotka, v hlavičke každého rámca je uložená informácia o tom, ktorý spôsob bol použitý. Toto umožňuje napríklad použitie spôsobu stereo v časti nahrávky kde je to podstatné a joint stereo v časti, kde to podstatné nie je. U *mono* spôsobu je ukladaný a dekodovaný iba jeden kanál, u *stereo* a *dual channel* sú pre daný časový úsek ukladané dve rôzne zvukové stopy. Tieto dva spôsoby sa líšia iba tým, že prehrávač podľa nich zistí, či sa jedná o stereo zvuk, alebo má tieto dva zvuky interpretovať ako dve rôzne stopy (napríklad rôznajazyčné zvukové stopy k filmu). Nakoniec v *joint stereo* móde nastáva aj stratová kompresia samotného sterea zlučovaním častí stereo stôp do jednej. *Joint stereo* navyše ešte existuje v dvoch módoch a to *Intensity* a *Middle-Side(M/S)*. V móde

Intensity stereo sa pre kódovanie využíva fakt, že ľudský sluch nedokáže, alebo len veľmi slabo, vnímať polohu zvuku pri príliš nízkych alebo príliš vysokých frekvenciách. Preto sa časť spektra úplne zlučí, čím sa odstráni nepotrebná informácia. V móde M/S sa vytvoria dva kanály, jeden súčtom pravého a ľavého kanálu - toto je "stredový" kanál, ktorý môže byť kódovaný vo vyššej kvalite, a druhý rozdielom pravého a ľavého kanálu - "postranný" kanál, ktorý môže byť kódovaný v nižšej kvalite. Pri dekódovaní je možné získať pravý a ľavý kanál bez straty stereo informácie jednoduchým sčítaním alebo odčítaním "stredového" a "postranného" kanálu (stredový je L+P, postranný L-P, sčítaním vznikne (L+P)+(L-P) čo je 2P, analogicky odčítaním vznikne 2L).

2.1.4 Dostupné riešenia dekodéru súborov MP3

Pre náš rozširujúci modul s dekodérom súborov MP3 je potrebné nájsť vhodný integrovaný obvod (IO), ktorý bude pracovať ako samotný MP3 dekodér. Takýchto IO existuje na trhu viacero.

2.1.4.1 STA013

IO od spoločnosti STMicroelectronics. Univerzálny MP3 dekodér podporujúci štandardy MPEG-1, MPEG-2 a MPEG-2.5 layer 3. Vstup získava cez sériové vstupné rozhranie a produkuje digitálny výstup pomocou PCM výstupného rozhrania priamo pripojiteľného na D/A konvertor, pričom toto rozhranie je softwarovo programovateľné a teda prispôsobiteľné pre veľkú väčšinu bežných DAC architektúr. IO je ďalej ovládateľné cez I2C, podporuje digitálne ovládanie hlasitosti a vyváženia *bass* a *treble*. IO je dostupný aj u českých predajcov v cene okolo 500Kč.

Od spoločnosti STMicroelectronics existuje aj ďalší IO STA310 schopný dekódovať MP3, avšak tento je určený pre aplikácie ako DVD prehrávače a HDTV, z čoho vyplývajú vlastnosti ako 6-kanálové audio, karaoke subsystém, podpora Dolby Digital a podpora S/PDIF, ktoré sú pre našu aplikáciu nepotrebné. STA310 neprináša žiadne ďalšie vlastnosti vhodné pre našu aplikáciu v porovnaní s STA013.[5]

2.1.4.2 MAS3507D

Tento dekodér od spoločnosti Micronas je schopný dekódovať oproti STA013 navyše aj layer 2 štandardov MPEG-1,2. Vstupné rozhranie umožňuje fungovať v dvoch módoch: *multimedia* a *broadcast*. V móde *multimedia* sa očakáva pripojený kontrolér, ktorý je schopný na vyžiadanie vystaviť dáta (napríklad kontrolér čítačky SD kariet), pričom prenos je ovládaný dekodérom pomocou *demand* signálu. V móde *broadcast* sa naopak očakáva že bitový tok prichádza nezávisle a dekodér sa s ním zosynchronizuje pomocou PLL. IO je ovládaný pomocou I2C.[6][7]

Výstupné rozhranie je sériové s možnosťou výberu viacerých módov, no stále je potrebný externý D/A prevodník. Výhodou je však celková cena tohto riešenia pretože aj v prípade, že bude využitý ďalší D/A prevodník (v prípade, že nebude využité audio rozhranie FITkitu alebo integrované D/A prevodníky FITkitu), celková cena je okolo 500Kč.

Od spoločnosti Micronas je dostupný ešte ďalší dekodér MAS3578F, ktorého výhodou je, že obsahuje integrovaný D/A prevodník. Nevýhodou je ale vyššia cena (približne trojnásobná oproti MAS3507D), pretože sa jedná o enkodér/dekodér. Keďže by sme schopnosť enkódovať MP3 v našej aplikácii nevyužili toto riešenie hodnotím ako neefektívne v porovnaní s MAS2507D.

2.1.4.3 BU9438KV

Ďalším dostupným dekodérom je IO od spoločnosti ROHM. Tento IO obsahuje množstvo vlastností vhodných pre našu aplikáciu. Podporuje všetky tri layere štandardov MPEG-1,2,2.5, k tomu tiež formáty WMA a AAC. Ďalšie veľmi užitočné vlastnosti sú podpora SD kariet s analýzou FAT tabuľky, podpora ID3 tagov, WMA tagov, AAC tagov a podpora matice tlačidiel.

Vstupné dáta sú načítané buď z SD karty alebo pomocou protokolu USB 2.0. IO je ovládaný pomocou zbernice I2C. Výstupné rozhranie je buď digitálne (sériové alebo S/PDIF), alebo priamo analógové s využitím integrovaného D/A prevodníku.[8][9]

Napriek všetkým vlastnostiam, ktoré tento IO kombinuje je jeho cena porovnateľná s vyššie uvedenými riešeniami - okolo 500Kč.

Spoločnosť ROHM vyrába celú škálu podobných IO avšak v ČR som ako dostupné našiel iba BU9438KV a BU9428KV, pričom druhý spomínaný je vlastnosťami skoro totožný s BU9438KV, avšak postráda podporu SD kariet, čo je pre našu aplikáciu dôležitá vlastnosť. Cenový rozdiel týchto dvoch IO je minimálny.

2.1.4.4 VS1053, VS1063

IO od fínskej spoločnosti VLSI sú radené medzi najpokročilejšie hardware audio kódeky. Oba IO sú schopné dekódovať široké spektrum formátov, konkrétne Ogg Vorbis, všetky layere štandardov MPEG-1,2, MPEG4 / 2 AAC-LC (+PNS), HE-AAC v2 (Level 3) (SBR + PS) , WMA4.0/4.1/7/8/9, WAW a FLAC (v prípade VS1053 je potrebný softwarový zásuvný modul pre FLAC).

Oba IO obsahujú integrovaný DAC. Vstupné rozhranie je sériové, dekódovanie ovládané cez I2C a výstupné rozhranie je tvorené buď priamo analógovým výstupom (s možnosťou zosilňovača pre slúchadlá), alebo digitálnym výstupom sériovo cez I2S. Okrem dekódovania zvládajú tieto IO aj enkódovanie ale pre našu aplikáciu je to nepodstatné. [10][11]

VS1063 je vylepšenou verziou VS1053 a sú pinmi spätne kompatibilné. VS1063 prináša mimo iné vylepšenia v podobe 5-kanálového ekvalizéru, podpory FLAC bez rozširujúceho softwarového zásuvného modulu, zmenu rýchlosti prehrávania a niekoľko ďalších vylepšení pri kódovaní. Cenový rozdiel nie je veľký, mnou nájdení predajcovia udávajú cenu cca 400Kč za VS1053 a cca 550Kč za VS1063.

2.1.4.5 Ostatné

Pri vyhľadávaní som našiel ešte niekoľko ďalších IO, avšak nenašiel som ich v ponuke žiadneho dodávateľa v ČR. Jedná sa napríklad o AT89C51SND2C spoločnosti Atmel a PT8405 spoločnosti Princeton. Keďže je táto podkapitola kapitola venovaná dostupným dekodérom nebudem tieto dva podrobnejšie rozoberať.

2.2 Platforma FITkit

Platforma FITkit je samostatný kus hardware určený pre vývoj širokej škály vstavaných (*embedded*) systémov. Primárny účel je umožniť študentom vyskúšať si a naučiť sa navrhovať a pracovať s vstavanými systémami, čo je v súčasnosti rozsiahle a často využívané odvetvie informačných technológií. Platforma je koncipovaná ako *open-source* a *open-core*, čo má umožniť, aby každý kto využíva platformu FITkit mohol využívať predchádzajúce výsledky ostatných užívateľov a zároveň poskytoval svoje výsledky pre budúce aplikácie. Vzhľadom na cieľ tejto bakalárskej práce, tj. vytvoriť modul dekodéru MP3, ktorý bude možné využiť v budúcich aplikáciach, je tento koncept veľmi vhodný.

Platforma FITkit obsahuje [12]:

- MCU rodiny MSP430 (Texas Instruments)
- FPGA Spartan 3 XC3S50-4PQ208C alebo XC3S400-4PQ208C (Xilinx)
- USB prevodník FT2232C
- audio rozhranie
- konektory PS2
- rozhranie VGA
- konektor RS232
- DRAM 8x8Mbit
- Klávesnice
- Riadkový LCD displej



Obrázok 2: Predná strana FITkit so znázornenou klávesnicou a displejom. Prebrané a upravené z [12].

- Rozširujúce konektory

V našej aplikácii sú dôležité súčasti MCU, FPGA hradlové pole, klávesnica a riadkový LCD displej. V nasledujúcich podkapitolách budú podrobnejšie popísané kritické komponenty FITkitu - MCU a FPGA.

2.2.1 FPGA

FPGA - *Field-Programmable Gate Array* je programovateľný integrovaný obvod, ktorý umožňuje rekonfiguráciu prepojenia obsiahnutých logických blokov, na rozdiel od bežných integrovaných obvodov, ktorých konfigurácia je pevne stanovená a u vyrobených kusov ju nie je možné zmeniť. Táto vlastnosť sa využíva pre návrh a vývoj hardware, pretože pomocou jedného integrovaného obvodu je možné vytvoriť ekvivalent nespočetného množstva iných obvodov (prakticky limitovaný len počtom programovateľných logických blokov, ktoré sú potrebné na danú aplikáciu), bez nutnosti s každou vývojovou verziou vytvárať ďalší fyzický čip. Pre popis konfigurácie sa používajú HDL jazyky, v našom prípade jazyk VHDL. Popis hardware u tohto jazyka spočíva v deklarovaní *entity*, ktorá definuje rozhranie komponentu a definovaní *architektúry*, ktorá predstavuje vnútornú štruktúru entity.

V prípade FITkit-u je použité FPGA rodiny Spartan 3 od spoločnosti Xilinx, konkrétne model XC3S50-4PQ208C, ktorý obsahuje[13]:

- rozhranie: až 124 užívateľských vstupov/výstupov (I/O), podpora až 23 rôznych I/O štandardov
- 192 konfigurovateľných logických blokov (CLB's) usporiadaných do matice o 16 riadkoch a 12 stĺpcoch, 1728 logických buniek, 50000 logických hradiel
- Pamäť RAM 12kb distribuovaných, 72kb v jednom bloku
- 2 jednotky pre správu hodín (DCM's)
- 4 blokové dvojportové pamäti BRAM s kapacitou 2kB
- 4 násobičky 18x18 bitov

2.2.2 MCU

MCU - *microcontroller unit* je integrovaný obvod obsahujúci výpočtovú jednotku, vlastnú pamäť a integrované periférie. Typicky obsahuje ROM pamäť v ktorej je uložený program a pamäť RAM pre výpočty. Mikrokontroléry je dnes možné nájsť skoro v každom elektronickom zariadení, vďaka ich cene a pomerne jednoduchému použitiu. Mikrokontrolér je prakticky samostatný počítač riadený jediným programom. Samotný program sa zväčša získava kompiláciou zdrojových súborov najčastejšie napísaných v jazyku C. Mikrokontroléry sa bežne programujú buďto použitím

programátora mikrokontrolérov, čo je samostatné zariadenie schopné nahráť vstupné dáta do pamäte mikrokontroléra alebo pomocou BSL bootloderu. Tento je súčasťou mikrokontroléru a umožňuje jednoduchú komunikáciu pomocou sériovej linky. U FITkitu je sériová linka sprístupnená počítaču pomocou rozhrania USB. Nevýhodou použitia BSL však je pomerne malá rýchlosť prenosu dát, čo je v prípade FITkitu vyriešené použitím BSL iba pre nahranie vlastného optimalizovaného bootloderu do pamäte RAM a jeho následné spustenie. Po spustení optimalizovaného bootloderu už je umožnená manipulácia s FLASH pamäťou značne vyššou rýchlosťou.

V najnovších verziách FITkitu je použitý mikrokontrolér rodiny MSP430 od spoločnosti Texas Instruments, konkrétne model MSP430F2617 s nasledujúcimi vlastnosťami[14]:

- Nízke napájacie napätie (rozmezíe 1.8 V až 3.6 V)
- Nízky príkon (330 uA v aktívnom režime pri 1 MHz a 2.2 V, 1.1 uA v stand-by režime, 0.2 uA v režime vypnuté)
- 16-bitová RISC architektúra, inštrukčný cyklus 125 ns
- Pamäť FLASH 48 KB, pamäť RAM 2kB (resp. 92kB FLASH a 8kB RAM v prípade FITkitu 2.0)
- Moduly na čipu:
 - 3-kanálové DMA
 - 16-bitové časovače Timer_A, Timer_B
 - Komparátor
 - Sériové komunikačné rozhranie USART0 (asynchrónne UART, synchrónne SPI, I2C), USART1 (asynchrónne UART, synchrónne SPI)
- 12-bitové A/D a D/A prevodníky

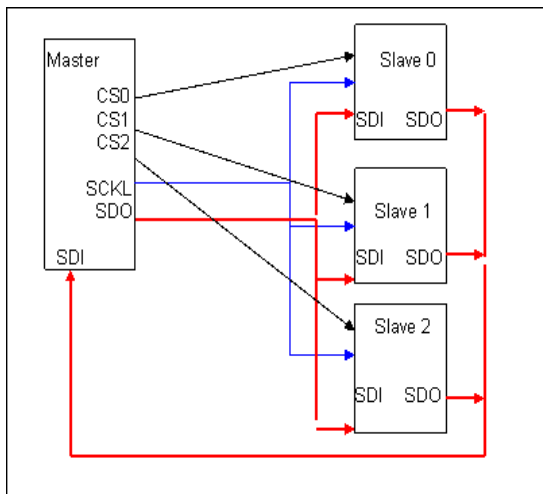
2.2.3 Komunikačné rozhranie MCU a FPGA

Keďže je FITkit vyvíjaný ako univerzálna platforma pre vývoj embedded systémov tak obsahuje aj širokú škálu možností komunikácie medzi svojimi komponentami. Pre našu aplikáciu sú podstatné dva spôsoby. Prvým je prepojenie jedného GPIO (*General Purpose Input/Output*) portu priamo na piny FPGA, čo umožňuje posielat' osem rôznych signálov. MCU obsahuje dokopy šesť takýchto portov avšak iba jeden je pripojený k FPGA, ostatné sú vyvedené na jedno z dvoch vstupno-výstupných rozhraní FITkitu. Druhým spôsobom komunikácie MCU a FPGA je použitie rozhranie SPI, ktoré bližšie popisuje nasledujúca podkapitola.

2.2.3.1 SPI

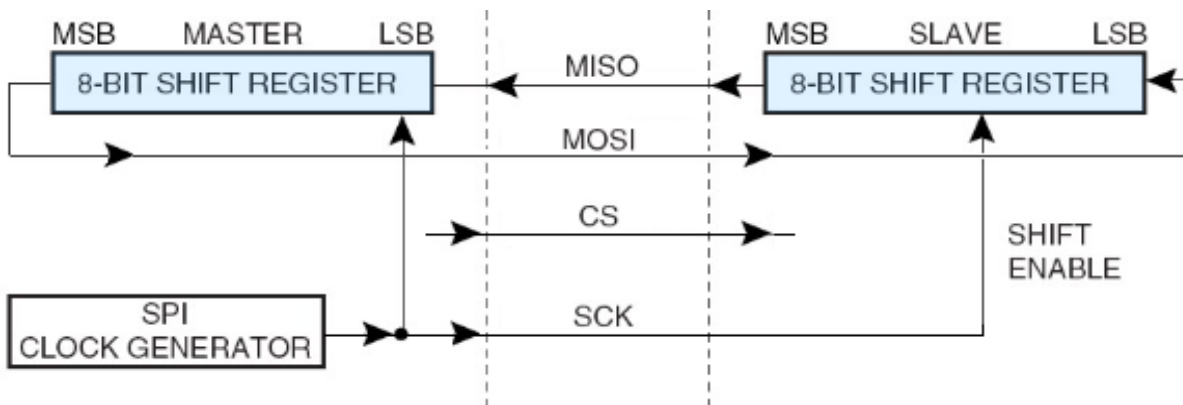
SPI - *Serial Peripheral Interface* je *de facto* štandard zavedený spoločnosťou Motorola. Je to jedna z najčastejšie používaných metód pre komunikáciu medzi rôznymi integrovanými obvodmi. Pre komunikáciu využíva tri zdieľané vodiče plus jeden vodič pre každé pripojené zariadenie, ktorý určuje adresáta komunikácie[16].

Na Obrázku 3 je možné vidieť zapojenie jedného *master* zariadenia a troch *slave*. V prípade, že *master* zariadenie chce komunikovať so *slave 1* aktivuje vodič CS1 a deaktivuje všetky ostatné. Na vodiči SCLK je *master* zariadením generovaný hodinový signál pre synchronizáciu prenosu a na vodičoch SDO resp. SDI budú vystavené každý



Obrázok 3: Schéma zapojenia viacerých zariadení pomocou SPI. Prebrané a upravené z [3].

hodinový takt vstupné dáta pre *slave* zariadenie od *master* zariadenia (niekedy je teno vodič nazývaný MOSI - *Master Out Slave In*) resp. výstupné dáta zo *slave* zariadenia (analogicky MISO - *Master In Slave Out*). Samotný prenos dát obvykle prebieha formou výmeny obsahu vstupno-výstupných



Obrázok 4: Znáznornenie použitia posuvných registrov pri prenose dát pomocou SPI. Prebrané a upravené z [15]

registrov zariadení. Na Obrázku 4 je možné vidieť ako prebieha výmena registrov. MSB registru jedného zariadenia sa vystaví na výstupný vodič, prebehne posun registra a na miesto LSB sa zapíše stav vstupného vodiča. V prípade 8-bitových registrov sa teda vymení obsah registrov po 8 taktach.

Na platforme FITkit prebieha hlavná časť komunikácie medzi FPGA a MCU pomocou SPI, pričom *master* zariadenie je MCU a *slave* zariadenia FPGA a pamäť FLASH uchovávajúca konfiguráciu FPGA.

2.3 Pamäťová karta SD

SD - *Secure Digital* karta je pamäťové médium vyvinuté spoločnosťou SanDisk v spolupráci so spoločnosťami Toshiba a Matsushita ako nasledovník MMC pamäťových kariet. Využíva sa v širokom spektre aplikácií vďaka svojim rozmerom, jednoduchému rozhraniu a vysokej kapacite. V súčasnosti existujú viaceré verzie líšiace sa fyzickými rozmermi a/alebo limitmi kapacity. S ohľadom na rozmery existujú tri typy kariet - SD, miniSD a microSD. Ich komunikačné rozhranie je však rovnaké takže sa dajú využiť adaptéry pomocou ktorých je napríklad možné zapojiť kartu microSD do slotu pre SD kartu bez zmeny funkčnosti.

S ohľadom na maximálnu kapacitu rozlišujeme dve rozšírenia pôvodného formátu a to SDHC(Secure Digital High capacity) podporujúci kapacitu do 32GB a najnovší SDXC(Secure Digital eXtended Capacity) podporujúci kapacitu až do 2TB. Rozšírenia sú pinovo kompatibilné avšak sú v nich zmeny v rozhraní, čo znamená že staršie zariadenia nemusia podporovať novšie rozšírenia. Avšak podľa špecifikácie musia byť zariadenia podporujúce novšie rozšírenia spätne kompatibilné so staršími.

Karty sa ďalej líšia aj rýchlosťou zápisu a čítania, ktorá závisí od konkrétneho modelu avšak nie od podpory rozšírení alebo rozmerového typu. SD pamäťové karty musia podporovať dve rôzne módy rozhrania. Prvým je komunikácia pomocou štandardu SPI, popísaného v minulej kapitole. V prípade čítania resp. zápisu na kartu sa najprv na kartu odošle príkaz, vyčká sa na odpoveď po ktorej sa karte odošlú dáta na zápis resp. príjmu sa dáta odoslané kartou v prípade čítania. Druhým módom komunikácie je patentovaný protokol, ktorý zrýchľuje komunikáciu použitím oddeleného vodiča pre dáta a pre príkazy. Tento druhý mód má v štandarde SD kariet definované rozšírenie, ktoré však nemusí byť implementované vo všetkých kartách. Spočíva v rozšírení počtu dátových vodičov na štyri, ako je možné vidieť v Tabulke 2, čo umožňuje značne vyššiu rýchlosť prenosu pri rovnakej taktovacej frekvencii. Konkrétne pri SPI je maximálna možná rýchlosť 50Mbit/s zatiaľ čo pri štyroch bitoch vyhradených pre dátový prenos je maximálna rýchlosť až 832Mbit/s.[17]

Pin	SD Mode			SPI Mode		
	Name	Type	Description	Name	Type	Description
1	CD/DAT 3	I/O/P P	Card detection / Connector data line 3	CS	I	Chip selection in low status
2	CMD	PP	Command/Response line	DI	I	Data input
3	Vss1	S	GND	VSS	S	GND
4	Vdd	S	Power supply	VDD	S	Power supply
5	CLK	I	Clock	SCLK	I	Clock
6	Vss2	S	GND	VSS2	S	GND
7	DAT0	I/O/P P	Connector data line 0	DO	O/PP	Data output
8	DAT1	I/O/P P	Connector data line 1	RSV		
9	DAT2	I/O/P P	Connector data line 2	RSV		

Tabulka 2: Popis jednotlivých pinov a ich použitia v prípade SPI a v prípade patentovaného rozhrania so štyrmi dátovými pinmi.

(Prebraté a upravené z http://pinouts.ru/Memory/sdcard_pinout.shtml)

V našej aplikácii však budeme využívať SPI rozhranie, pretože je ľahko implementovateľné a prenosová rýchlosť je viac ako dostatočná pre prácu s MP3 súbormi.

3 Realizácia modulu

V tejto kapitole bude na začiatku popísaná demonštračná aplikácia z užívateľského hľadiska a následne budú podrobne popísané implementačné detaily ako použitý MP3 kódex, jeho vlastnosti, rozbor navrhutej dosky plošných spojov a komunikačné rozhranie s FITkitom. Ďalej bude popísaná konfigurácia FPGA a program pre MCU platformy FITkit.

3.1 Popis funkcie aplikácie

Aplikácia na základe užívateľského vstupu prehráva súbory MP3 uložené na pamäťovej karte SD. Táto musí byť vložená do slotu na rozširujúcom module. Ihneď po spustení zariadenia začne postupné prehrávanie súborov z SD karty. Užívateľské voľby je možné zhrnúť do dvoch kategórií. Prvá kategória užívateľských volieb ovplyvňuje zvuk, druhá prehrávanie. Ku voľbám ovplyvňujúcim zvuk patrí nastavenie hlasitosti prehrávania, úprava prahovej frekvencie pre vylepšenie basových a vysokých tónov a miera zosilnenia basových a vysokých tónov. Ku voľbám ovplyvňujúcim prehrávanie patrí preskočenie na prehrávanie ďalšieho súboru a pozastavenie prehrávania. Užívateľ svoje voľby zadáva pomocou klávesnice obsiahnutej v platforme FITkit. Informácie o súčasnom stave nastavovaných parametrov sú vypísané na LCD displeji taktiež obsiahnutom v platforme FITkit. Detaily ovládania sú uvedené v Tabulke 3.

Stlačená klávesa:	Efekt:
"A"	Prepnutie do módu nastavovania zosilnenia basových tónov
"B"	Prepnutie do módu nastavovania prahovej frekvencie basových tónov
"C"	Prepnutie do módu nastavovania zosilnenia vysokých tónov
"D"	Prepnutie do módu nastavovania prahovej frekvencie vysokých tónov
"1"	Prepnutie do módu nastavovania hlasitosti
"2"	Prepnutie do módu preskakovania súborov
"3"	Pozastavenie/obnovenie prehrávania
"*"	Zníženie úrovne vybraného parametra
"#"	Zvýšenie úrovne vybraného parametra

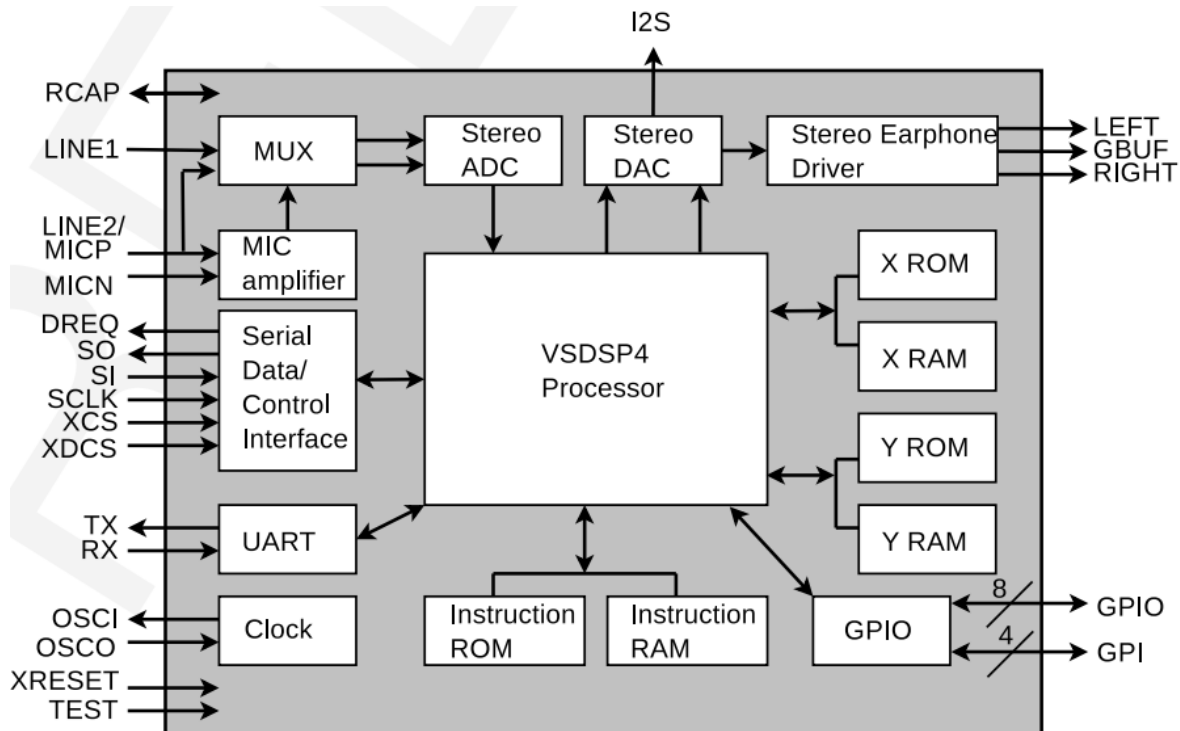
Tabulka 3: Popis ovládania demonštračnej aplikácie

3.2 MP3 kódek

V kapitole 2.1.4 boli predstavené alternatívy pre výber konkrétneho integrovaného obvodu pre dekodovanie MP3. Po zvážení možností som si vybral ako MP3 kódek integrovaný obvod od firmy VLSI solutions, konkrétne model VS1063a. Dôležitých faktorov pri výbere bolo viacero. Prvým bol veľký potenciál pre rozšírenia vďaka podpore softwarových zásuvných modulov implementovateľných do čipu. Toto umožňuje rôzne rozšírenia pre spracovanie a úpravu zvukového signálu jednoduchou úpravou softwarového zásuvného modulu, čo môže byť veľmi užitočné pri ďalších aplikáciách využívajúcich modul. Druhým dôležitým faktorom bol integrovaný D/A prevodník so zosilňovačom, čo umožňuje výstup priamo do slúchadiel bez použitia ďalších komponent. Ďalším argumentom pre toto riešenie bola kvalitne spracovaná dokumentácia čipu, ktorá značne zjednodušuje implementáciu. V neposlednom rade bola tiež dôležitá podpora širokej škály audio formátov a okrem dekodovania aj možnosť kódovania vstupného signálu do rôznych formátov.

3.2.1 Popis rozhrania čipu VS1063a

VS1063a a jeho rozhranie je principiálne možno rozdeliť na dve časti: analógovú a digitálnu. Schému je možné vidieť na Obrázku 5. Súčasťou analógovej časti sú piny LEFT, RIGHT a GBUF pre audio výstup, ktoré je možné, aj keď nie nutné priamo pripojiť na slúchadlá. Pin GBUF je možné



Obrázok 5: Schéma rozhrania čipu VS1063a. Prebrané a upravené z [11].

využiť iba pri sluchátkach lebo v opačnom prípade hrozí skrat. Piny LINE1, LINE2 a MICN sú zase použité pre audio vstup v prípade kódovania vstupného signálu. Je možné použiť buďto vstup z LINE IN alebo z mikrofónu pri ktorom je použitý integrovaný zosilňovač. V našej aplikácii sa používa ako vstup LINE IN, pretože v bežnej situácii je lepšie využiteľný.

Digitálna časť rozhrania pozostáva z SPI rozhrania, pinov OSCI a OSCO určených na pripojenie externého oscilátora pre generovanie hodinového signálu, UART rozhrania vhodného pre odlaďovanie, pinov XRESET a TEST a ôsmich pinov určených pre užívateľské rozšírenia, prípadne boot z EEPROM pamäte.

SPI rozhranie u čipu VS1063a je tvorené zdieľanými signálmi SI, SO a SCLK, dvoma CS signálmi a DREQ signálom. Dôvod prečo sú použité dva CS signály je ten, že u tohoto čipu je sériové rozhranie SPI rozdelené na dva protokoly, každý s vlastným CS signálom. Prvý je SDI - Serial Data Interface, pomocou ktorého sa jednoducho prenášajú dáta ako napríklad dekodovaná zvuková stopa v digitálnej podobe, alebo dáta z ID3 tagov. Používa XDCS pin ako CS signál.

Druhým protokolom je SCI - Serial Command Interface. Tento už má stanovený formát prenášaných údajov. Jeden príkaz pozostáva z jedného inštrukčného bytu, jedného adresového bytu a jedného dvojbytového dátového slova. Podporované inštrukcie sú zápis a čítanie. Ovládanie čipu pozostáva v zápise hodnôt do registrov. Stručný popis registrov sa nachádza v Tabulke 4, konkrétne popisy jednotlivých registrov je možné nájsť v [11].

SCI registers, prefix SCI_					
Reg	Typ.	Reset	Write Time ¹	Name	Description
0x0	rw	0x4000 ⁶	80 CLKI ⁴	MODE	Mode control
0x1	rw	0x000C ³	80 CLKI	STATUS	Status of VS1063a
0x2	rw	0	80 CLKI	BASS	Built-in bass/treble control
0x3	rw	0	1200 XTALI ⁵	CLOCKF	Clock freq + multiplier
0x4	rw	0	100 CLKI	DECODE_TIME	Decode time in seconds
0x5	rw	0	450 CLKI ²	AUDATA	Misc. audio data
0x6	rw	0	100 CLKI	WRAM	RAM write/read
0x7	rw	0	100 CLKI	WRAMADDR	Set address for RAM write/read
0x8	r	0	80 CLKI	HDATA0	Stream header data 0
0x9	r	0	80 CLKI	HDATA1	Stream header data 1
0xA	rw	0	210 CLKI ²	AIADDR	Start address of application/plugin
0xB	rw	0	80 CLKI	VOL	Volume control
0xC	rw	0	80 CLKI ²	AICTRL0	Application control register 0
0xD	rw	0	80 CLKI ²	AICTRL1	Application control register 1
0xE	rw	0	80 CLKI ²	AICTRL2	Application control register 2
0xF	rw	0	80 CLKI ²	AICTRL3	Application control register 3

Tabulka 4: Stručný popis riadiacich registrov čipu VS1063. Prebraté a upravené z [11].

Pin DREQ primárne slúži ako signalizácia, že čip môže prijať ďalšie dáta. Napríklad v prípade keď cez rozhranie SPI posielame väčšie množstvo dát ako je čip schopný dekodovať tak pomocou DREQ pinu ohlási že je plný vstupný buffer a je potrebné pozastaviť odosielanie nových vstupných dát. Ďalšie využitie je v prípade využívania SCI protokolu. Po každom príkaze sa deaktivuje DREQ po dobu plnenia príkazu a aktivuje sa až keď je čip pripravený vykonať ďalší príkaz. Avšak DREQ nie je nutné využívať v prípade že sa po každom príkaze čaká maximálnu možnú dobu trvania príkazu.

Pin XRESET je aktívny v stave logickej 0, a pri aktivácii asynchrónne spôsobí hardwarový reštart čipu.

3.2.2 Popis činnosti čipu VS1063a

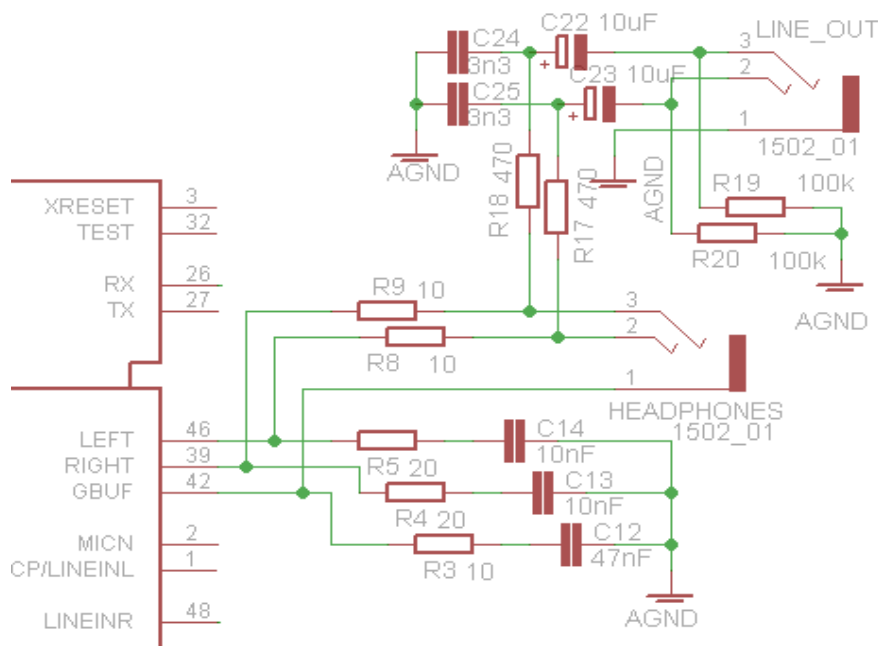
Základnou aplikáciou čipu je dekodovanie a prehranie zvukovej nahrávky. V základnej konfigurácii čip aktívne monitoruje stav SPI rozhrania konkrétne SDI protokolu popísaného v minulej podkapitole. V prípade, že sa vyskytnú vstupné dáta čip sa ich okamžite bude snažiť dekodovať a prehrať. V prípade, že nie je možné dáta dekodovať, prípadne nie sú žiadne ďalšie vstupné dáta prechádza čip do módu *idle* v ktorom stále aktívne monitoruje SDI, ale minimalizuje svoju spotrebu elektrickej energie. Všetky ostatné módy je nutné ovládať zápisom do SCI registrov.

Čip však obsahuje množstvo rôznych nastavení a módov činnosti. Príkladom môže byť nahrávanie a kódovanie vstupného analógového signálu do rôznych formátov podľa nastavenia. Pre aktivovanie kódovania analógového signálu je potrebné najprv v registri SCI_AICTRL0 nastaviť požadovaný *samplerate*, v registri SCI_AICTRL1 hodnotu zisku alebo 0 pre automatické ovládanie hodnoty zisku, v registri SCI_AICTRL2 maximálnu hodnotu zisku, správne hodnoty bitov v registri SCI_AICTRL3 pre výber kanálov, výstupného formátu a niektorých ďalších nastavení. V prípade kódovanie do MP3 alebo Ogg Vorbis formátu je potrebné ešte nastaviť *bitrate* a *kvalitu* v registri SCI_WRAMADDR. Na záver je potrebné správne nastaviť v registri SCI_MODE bity 2,12 a 14 ktorými sa aktivuje mód *encode*, nastaví sa zdroj analógového vstupu (mikrofón alebo LINE IN) a aktivuje sa software reštart. Po tomto je možné pomocou rozhrania SPI čítať kódovanú nahrávku. Na záver sa ukončuje nahrávanie zápisom bitu SM_CANCEL do registra SCI_MODE.[11]

Navyše okrem týchto základných funkcií je možné rozšíriť funkcionality pomocou softwarových zásuvných modulov. Tieto môžu napríklad sprostredkovať podporu priameho čítania z SD karty, čo je pre našu aplikáciu dôležitá vlastnosť.

3.3 Doska plošných spojov modulu

Srdcom modulu je vyššie popísaný integrovaný obvod VS1063a. V tejto podkapitole budú postupne popísané všetky časti dosky plošných spojov.



Obrázok 6: Analógová časť navrhnutej dosky plošných spojov.

3.3.1 Analógová časť

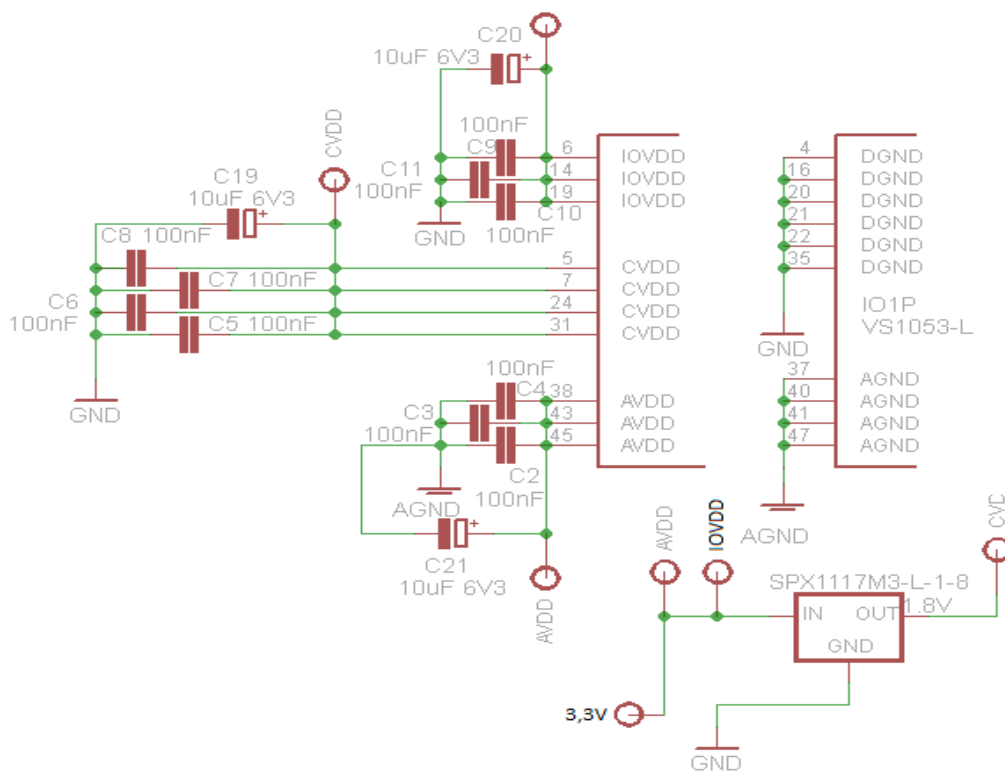
Analógová časť dosky plošných spojov je oblasť, v ktorej sa nachádzajú zvukové vstupy a výstupy. Na obrázku 6 môžeme vidieť pripojenie konektora pre slúchadlá a konektora pre pripojenie ďalšieho zariadenia (napríklad zosilňovača). Hneď pri čipe je každý vodič prepojený cez RC dvojicu na zemnenie. Toto zapojenie chráni čip pred elektrostatickými výbojmi. Výrobca tvrdí, že takéto zapojenie v laboratórnych podmienkach zneslo opakované 8kV pulzy bez poškodenia samotného čipu.[11] Ďalej nasledujú dva malé odpory, ktoré chránia čip pred preťažením v prípade skratu. Vetva pre pripojenie ďalšieho zariadenia potom ešte obsahuje prvky odporúčané výrobcom pre pripojenie zariadení s vyššou impedanciou a oddelovacie kondenzátory.

Pri pripájaní zariadení je potrebné správne zapojiť pasívne zariadenia do konektora pre slúchadlá a aktívne do LINE OUT. Okrem spôsobenia obvodov pre rôzne impedancie a zosilnenie je dôležitým faktorom použitie signálu GBUF u slúchadiel. Tento umožňuje priame pripojenie slúchadiel so zosilnením a bez potreby oddelovacích kondenzátorov. V tomto prípade je ale na vodiči GBUF interne udržiavané na napätí cca 1.2V, čo by mohlo spôsobiť preťaženie obvodu v prípade, že by bolo

pripojené na zem druhého zariadenia. Možný prípad je pripojenie k počítaču ktorý je zároveň cez rozhranie USB spojený s FITkitom takže uzemnenie počítača je priamo spojené s uzemnením modulu.

3.3.2 Napájanie dosky plošných spojov

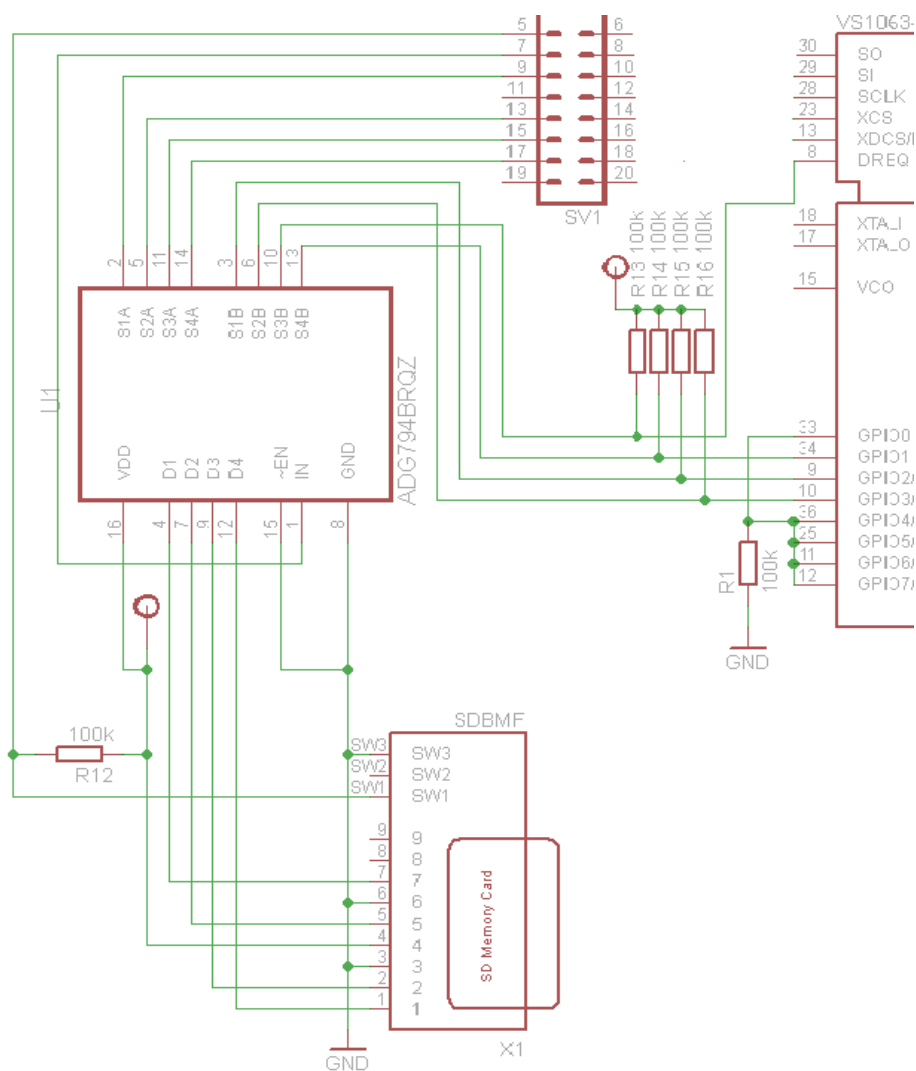
Všetky napájacie piny majú byť podľa odporúčania výrobcu čipu opatrené vyrovnávacími kondenzátormi na zrovnávanie výkyvov napätia. Výrobca zároveň odporúča umiestniť kondenzátory fyzicky čo najbližšie k čipu. Napájanie čipu je rozdelené na tri časti a to napájanie analógovej časti, napájanie vstupno-výstupného rozhrania a napájania výpočtového jadra čipu. Odporúčané rozsahy napájacieho napätia sú 1.7-3.6V pre vstupno-výstupnú časť, 3.3-3.6V pre analógovú časť a 1.7-1.85V pre výpočtové jadro.[11] Z toho vyplýva, že analógovú ani vstupno-výstupnú časť pri napájanom napätí modulu 3.3V nebude treba upravovať, ale v prípade napájania jadra je nutné použiť regulátor napätia na 1.8V. Pri každej napájacej vetve je zároveň umiestnený elektrolytický kondenzátor na vyrovnávanie krátkodobých poklesov napájacieho napätia. Ďalšie odporúčanie výrobcu je používať zvlášť zem pre analógovú časť a pre zvyšnú časť aby sa čo najviac predišlo rušivým vplyvom, avšak majú byť prepojené čo najbližšie pri čipe.



Obrázok 7: Schéma napájania navrhutej dosky plošných spojov.

3.3.3 Pripojenie karty SD

Pripojenie SD karty je z dôvodu umožnenia rozšírenia realizované pomocou štvornásobného analógového multiplexora 2:1. Pre našu aplikáciu je vhodné pripojiť kartu SD priamo ku všeobecným vstupno-výstupným pinom čipu, pretože bude použitý softwarový zásuvný modul ktorý umožní čipu komunikovať priamo s kartou SD čo odľahčí komunikáciu s FITkitom a zároveň demonštruje univerzálnosť použitého integrovaného obvodu. Analógový multiplexor je zároveň možné kedykoľvek prepnúť a umožniť komunikáciu FITkitu s práve pripojenou kartou SD, previesť na FITkite prípadné úpravy prečítaných dát a odoslať dáta do modulu štandardnou cestou cez SPI na dekódovanie.



Obrázok 8: Schéma pripojenia SD karty v navrhovanej doske plošných spojov

Prepojenie čipu s platformou FITkit je realizované pomocou rozhrania SPI. Keďže MCU FITkitu natívne podporuje SPI, poskytuje nám toto veľkú výhodu. Pri komunikácii nie je pripojený pin DREQ takže je potrebné dávať pozor na to, aby sa nepreplnil vstupný buffer, a aby mal čip po každom príkaze dostatok času na jeho vykonanie.

Na samotné prepojenie s platformou FITkit je použitý plochý kábel, pomocou ktorého sa prenášajú signály rozhrania SPI, ostatné signály použité na ovládanie modulu a napájacie napätie modulu. Kábel má dvadsať vodičov, pričom využitých je osemnásť, čo môže zjednodušiť prípadné rozšírenia.

3.3.4 Výroba dosky plošných spojov

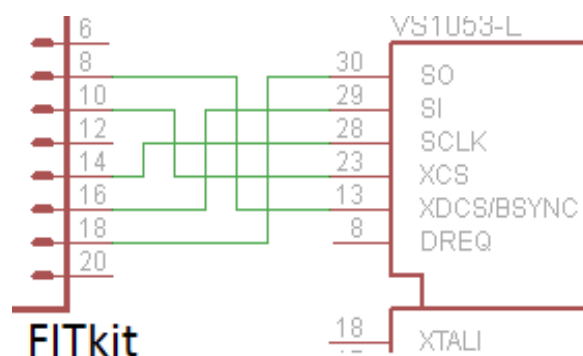
Výroba dosky prebiehala pomocou frézky LPKF ProtoMat S-62, ktorá je súčasťou Laboratória pre návrh a prototypovú realizáciu pokročilých elektronických systémov Fakulty Informačných Technológií VUTBR. V tomto laboratóriu boli vykonávané aj ďalšie úkony potrebné pre výslednú dosku, ako vŕtanie a prekovovanie prepojok, pokrývanie nespájkovaciou maskou a finálne osádzanie. Výroba dosky však trvala dlhšiu dobu ako bolo očakávané, vďaka obsadeniu laboratória a neskôr drobných problémov s prekovovačkou. Hlavným časovým problémom však bola skutočnosť, že autor ako študent nemá samostatný prístup do laboratória takže bolo nutné vždy nájsť vhodný čas, keď mohol byť v laboratóriu prítomný ako autor tak aj jeho vedúci.

Ďalší problém pri výrobe dosky sa objavil po pretavení v pretavovacej peci, keď sa čip VS1063a z neznámych príčin pri tomto procese na jednej strane nadvihol a tým stratil kontakt s doskou. Pri oprave tohto problému nastalo poškodenie niekoľkých pinov, čo pravdepodobne spôsobilo nefunkčnosť modulu.

Po rôznych pokusoch o opravu tohto problému bolo prikrôčené k riešeniu spočívajúcim vo výmene čipu za druhý.

3.4 Konfigurácia FPGA

Rekonfigurovaním hradlového pola je možné vytvoriť prakticky ľubovoľný digitálny obvod. Táto podkapitola popisuje konfiguráciu použítú v aplikácií s našim modulom. Úlohou FPGA v tejto aplikácií je sprostredkovať komunikačné rozhranie MCU s modulom, ale aj s maticovou klávesnicou a LCD displejom. Zároveň práve u FPGA je najväčší potenciál pre prípadne vylepšenia.



Obrázok 9: Schéma zapojenia rozhrania SPI

Na platforme FITkit je v rámci zjednodušenia vývoja aplikácií pre komunikáciu s FPGA navrhnutý interný sériový protokol fungujúci nad SPI rozhraním. Komunikáciu pomocou tohto protokolu je možné rozdeliť na tri časti: operačný kód, ktorý určuje, že sa komunikácia týka FPGA a zároveň operáciu ktorá sa má zrealizovať, adresu cieľa, ktorá určuje konkrétny SPI radič v rámci FPGA a samotné prenášané dáta. MCU v prípade potreby komunikácie s nejakou komponentou v FPGA aktivuje CS signál pre FPGA a odošle pomocou SPI operačný kód, adresu a dáta. Dĺžka adresy nie je obmedzená avšak treba dávať pozor aby nebolo omylom adresované iné zariadenie použitím kratšej adresy a následnými dátami zhodnými so zvyškom inej adresy.

Pre internú komunikáciu bola navrhnutá sada komponentov ktoré umožňujú komunikáciu MCU s ľubovoľným počtom zariadení v rámci FPGA. Základ tvorí radič SPI, ktorý realizuje jednoduchý prevod SPI protokolu na interný sériový protokol. Ďalšou podstatnou komponentou je SPI dekodér, ktorý transformuje sériovú komunikáciu na jednoduché paralelné rozhranie. Interná zbernica potom môže byť obsadená ľubovoľným počtom SPI dekodérov.[15] SPI radič dekoduje operačný kód, najmä jeho prvé 4 bity, ktoré sú 0001 v prípade že sa jedná o príkaz určený pre FPGA. V tom prípade aktivuje interný signál CS ktorý signalizuje SPI dekodérom prichádzajúcu komunikáciu. SPI dekodér po aktivácii interného signálu CS prečíta štyri bity určujúce vyžiadanú operáciu a potom parametricky určený počet bitov adresy. V prípade, že sa zhoduje s adresou daného dekodéru prečíta ten ďalších N bitov, pričom N je určené parametricky, a dáta vystaví na výstupnú zbernicu.

V našej aplikácií je použitý klasický SPI radič a na internú zbernicu pripojené dve zariadenia, každé s vlastným SPI dekodérom. Tieto zariadenia sú klávesnica a LCD displej. Okrem tohto je na zbernicu SPI prepojený modul s dekodérom MP3.

3.4.1 Maticová klávesnica

Maticová klávesnica platformy FITkit je tvorená jednoduchou maticou tlačidiel usporiadaných podľa Obrázku 9. Klávesnica je pripojená k vstupno-výstupným portom FPGA.

Zistenie stavu tlačidiel prebieha postupne po riadkoch, pričom sa vždy na jeden vodič KIN pripojí signál. Každý z vodičov KIN je zdieľaný jedným riadkom. V prípade, že na danom riadku bolo stlačené tlačidlo je možné tento signál zachytiť na niektorom vodiči KOUT, ktoré sú zdieľané vždy jedným stĺpcom, čím sa presne určia súradnice stlačeného tlačidla.

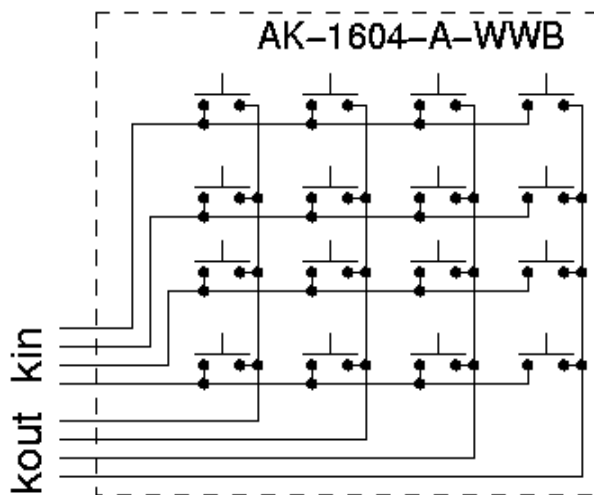
Konkrétna implementácia predpokladá, že v prípade zopnutého kontaktu tlačidla bude na danom vodiči KOUT logická 0. V tomto prípade musí byť ku každému vodiču pripojený *pull-up* rezistor, ktorý zabezpečí, že v prípade keď nie je zopnuté žiadne tlačidlo bude logický stav vodiča v logickej 1.

Klávesnica je pripojená pomocou SPI dekodéru na interné sériové rozhranie, ktorým s využitím SPI radiča komunikuje s MCU. Kontrolér implementovaný v FPGA prevádza stav vodičov KIN a KOUT na sériové rozhranie. Pre ovládanie klávesnice sú predpripravené obslužná knižnica a kontrolér, čím je značne zjednodušené použitie klávesnice v aplikáciách. V našom prípade bola použitá časť kódu ukážkovej aplikácie využívajúcej tieto predpripravené riešenia.[18]

3.4.2 LCD displej

FITkit mimo iné obsahuje aj plnohodnotný riadkový LCD displej. Modul displeja obsahuje okrem samotného LCD prvku aj interný radič s troma rôznymi pamäťami. ROM pamäťou s bitovými mapami štandardných znakov, RAM pamäť pre rozšírené užívateľské znaky a RAM pamäťou uchovávajúcou interný stav LCD displeja. Pre čo najväčšie zjednodušenie ovládania sú predpripravené obslužná knižnica a externý kontrolér realizovaný pomocou FPGA. Ovládanie LCD displeja potom prebieha odosielaním príkazov cez dátovú zbernicu.[19] Príkladom príkazu môže byť vypísanie znaku, posunutie vykreslených znakov o jeden znak alebo vymazanie displeja.

V našej aplikácii bola taktiež použitá časť kódu ukážkovej aplikácie využívajúca predpripravenú knižnicu a externý kontrolér v FPGA.



Obrázok 10: Ilustrácia zapojenia maticovej klávesnice platformy FITkit. (prebrané a upravené z [18])

3.4.3 Prepojenie modulu a MCU

Prepojenie rozširujúceho modulu a MCU platformy FITkit je zrealizované prepojením signálov vlastnou komponentou v FPGA. Vytvorená komponenta realizuje jednoduché prepojenie vstupno-výstupných portov MCU a vstupno-výstupného rozhrania FITkitu.

Na toto rozhranie je možné pripojiť konektor a pomocou neho realizovať komunikáciu s externými modulmi ktoré nie sú súčasťou platformy FITkit.

Konkrétna implementácia prepája signály portu P3 mikrokontroléru na vstupno-výstupné vodiče FPGA pripojené na vstupno-výstupný port J10. Mikrokontrolér obsahuje celkovo šesť vstupno-výstupných portov, avšak iba port P3 je pripojený na FPGA. Ostatné porty sú vyvedené na vstupno-výstupné rozhranie FITkitu J9. Rozdiel medzi portom J9 a J10 je v zariadení, ktoré je k nemu pripojené. Zatiaľ čo na J10 je pripojené FPGA, na J9 je pripojný MCU. Pri implementácii demonštračnej aplikácie by bolo možné využiť aj prepojenie iného portu mikrokontroléru na J9, avšak takéto riešenie by sťažilo neskoršie rozšírenia, ktoré by chceli využívať komponenty vytvorené v FPGA. [20]

3.5 Implementácia programu pre MCU a funkčný popis aplikácie

Hlavná logika programu pozostáva z dvoch častí. V prvej časti sa inicializujú zariadenia a v druhej sa metódou aktívneho dotazovania zisťuje užívateľský vstup. V tejto podkapitole budú popísané obe časti programu.

3.5.1 Inicializačná časť programu pre MCU

V inicializačnej časti programu je vykonaná inicializácia zariadení ako LCD displeja a klávesnice. Dôležitou súčasťou inicializácie je ale inicializácia rozširujúceho modulu. V našom prípade je totiž použitý softwarový zásuvný modul, poskytovaný výrobcom ako ukážková aplikácia, ktorý je potrebné najprv zapísať do pamäte čipu. Pre naše potreby je ale veľmi žiadaný pretože okrem iného prináša priamu podporu karty SD, vrátane podpory FAT súborového systému. Zároveň však neuberá možnosť využívať kódek bežným spôsobom, pretože vytvára druhé SPI rozhranie na nevyužitých GPIO výstupoch kódeku.

Softwarový zásuvný modul je uložený v programe v komprimovanom formáte, čo šetrí pamäťový priestor MCU. Jeho zavádzanie do kódeku prebieha pomocou rozhrania SPI. Pre zavádzanie výrobcu poskytuje aj zdrojový kód špeciálnej zavádzacej funkcie ktorá realizuje dekomprimáciu. Softwarový zásuvný modul využíva pre svoje ovládanie a vystavenie údajov registre AICTRL0,1 a 3. Register AICTRL0 obsahuje poradové číslo prehrávaného súboru, ale zároveň

umožňuje zapísaním súčtu poradového čísla súboru a konštanty 0x8000h preskočiť v prehrávaní na zadaný súbor. Register AICTRL1 obsahuje celkový počet súborov objavených na SD karte a register AICTRL3 poskytuje možnosť rôznych nastavení, ako zákaz zisťovania počtu súborov na karte SD (pozn.: toto môže neprijateľne predĺžiť čas od štartu po možnosť začiatku prehrávania v prípade veľkého množstva súborov na karte alebo rozsiahlej adresárovej štruktúry), alebo povolenie módu prehrávania v náhodnom poradí.

Po zavedení softwarového rozširujúceho modulu do kódeku je ešte vhodné nastaviť počiatočnú hlasitosť, v našom prípade sa hodnota hlasitosti inicializuje na polovicu maximálnej hlasitosti s ohľadom na možný negatívny vplyv na sluch užívateľa. Ďalej sa nastaví počiatočná hodnota vylepšenia basových a vysokých tónov a na záver inicializácie sa spustí beh softwarového zásuvného modulu zápisom konštanty 0x50h do registru SCI_AIADDR, ktorý je určený pre zadanie počiatočnej adresy softwarového zásuvného modulu.[11]

3.5.2 Spracovávanie užívateľského vstupu

Po úspešnej inicializácii prechádza mikroprocesor do nekonečnej slučky v ktorej sa každých desať milisekúnd overuje stav tlačidiel. V prípade, že nastane zmena rozlišuje sa či nastala zmena módu, alebo zvýšenie/zníženie hodnoty. Pri každej zmene sa vyčistí LCD displej a vypíše sa aktuálne nastavený mód a aktuálna hodnota. Všetky zmeniteľné hodnoty sú uložené v globálnych premenných v pamäti mikrokontroléru okrem poradového čísla prehrávaného súboru, pretože to sa môže zmeniť aj bez priameho pokynu z mikrokontroléru v prípade, že skončí práve prehrávaný súbor. Preto sa pri každej zmene najprv načíta aktuálna hodnota z registru AICTRL0, ktorá sa následne zvýši, resp. zníži v závislosti od stlačeného tlačidla, a po pripočítaní konštanty 0x8000h sa spätne zapíše do toho istého registru. Špecifické chovanie má program ešte v móde *pauza* pri ktorom sa zastaví prehrávanie. Zastavenie prehrávania je ovládané jediným bitom, takže je nutné tiež načítať celý register, zmeniť špecifický bit a následne zapísať na pôvodné miesto. V móde *pauza* tiež ovládanie nereaguje na žiadne iné tlačidlá okrem tlačidla "3" ktoré začína ale aj ukončuje pauzu prehrávania.

4 Záver

Modul by predstavoval zaujímavé a uplatniteľné rozšírenie platformy FITkit, najmä v multimediálnych aplikáciach. Celý projekt bol vyvíjaný vzhľadom na čo najväčšie možnosti budúcich rozšírení, či už využitím ďalších komponent v FPGA, ktoré umožnia napríklad pripojenie rôznych iných zdrojov dát, alebo využitím schopnosti použitého kódeku kódovať analógový signál. S využitím ethernetového pripojenia by bolo možné vytvoriť zariadenie na prehrávanie dátových *streamov*. V prípade pridania podpory bezdrôtového prenosu by zase bolo možné vytvoriť jednoduchú digitálnu vysielačku. Množstvo aplikácií by bolo možné vytvoriť pomocou softwarového rozširujúceho modulu, ktorý umožňuje okrem iného manipuláciu s dekódovanými dátami.

Praktická realizácia modulu však skončila neúspechom. Ako bolo popísané v podkapitole venovanej výrobe dosky plošných spojov počas výroby sa vyskytli neočakávané zdržania, čo znemožnilo opravu modulu, keďže by bolo nutné objednať nový čip. Pri analýze problému a meraniach som totiž prišiel na to, že čip sa ani nenašartuje. Sprievodným znakom štartu čipu je zmena logického stavu vodiča DREQ na logickú 0 a po krátkej dobe zase na 1. Avšak toto sa nedeje. Takisto v analógovej časti nie je možné namerať na referenčnom kondenzátore očakávané napätie okolo 1.3V, čo podľa výrobcu signalizuje minimálne problém analógovej časti. Na všetkých napájacích pinoch je možné namerať napätie spĺňajúce odporúčania výrobcu. Z týchto indícií je možné usúdiť, že sa jedná o hardwarový problém. Keďže návrh dosky plošných spojov spĺňa všetky požiadavky výrobcu tak chyba bude pravdepodobne v čipe, ktorý sa mohol znova poškodiť pri výmene.

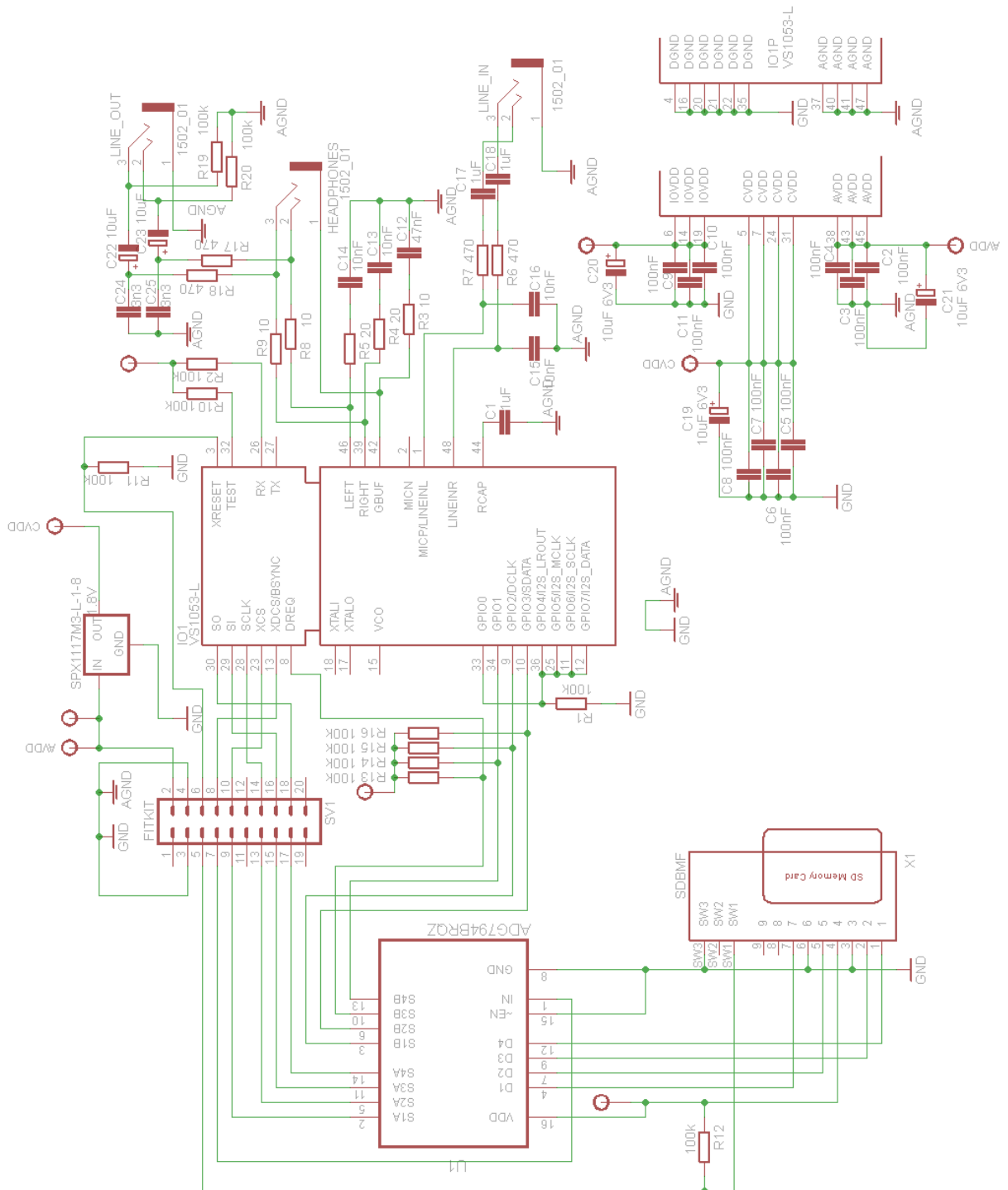
V súčasnosti je teda jediné možné budúce pokračovanie projektu vyriešenie hardwarového problému a následné odladenie implementovanej aplikácie.

Použitá literatúra

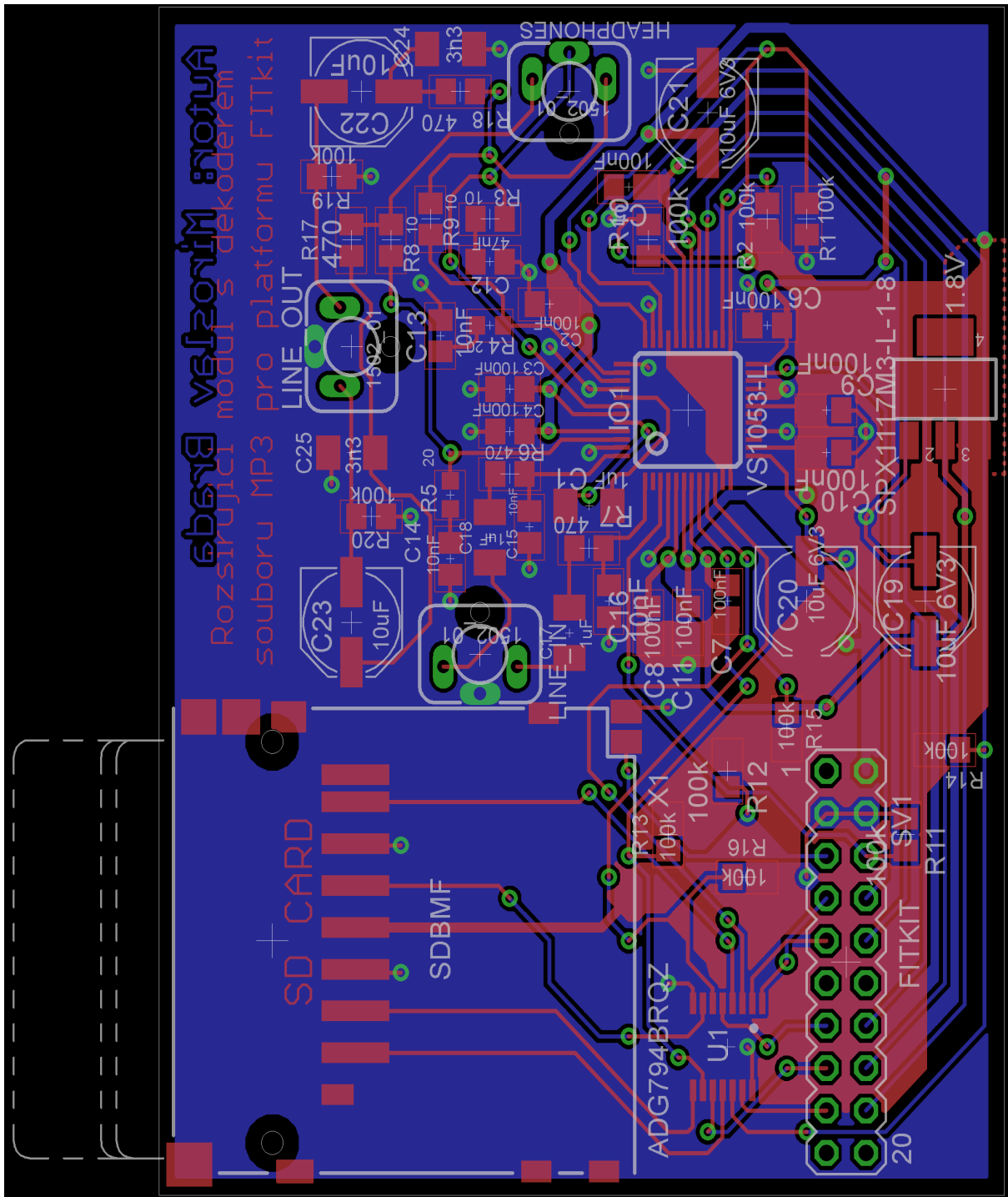
- [1] Wikipedia, Data compression [online]. [cit. 12.4.2012] URL:
[http://en.wikipedia.org/wiki/Audio_compression_\(data\)#Audio](http://en.wikipedia.org/wiki/Audio_compression_(data)#Audio)
- [2] Fältman, I., Hast, M., Lundgren, A., Malki, S., Montnemery, E., Rångevall, A., Sandvall, J., Stamenkovic, M. A hardware implementation of an MP3 decoder [online]. 2003. [cit. 12.4.2012] URL:http://www.mp3-tech.org/programmer/docs/fpga_report.pdf
- [3] Bouvigne, G., MP3Tech[web], [cit. 12.4.2012] URL:<http://www.mp3-tech.org/>
- [4] Edström, B., Let's build an MP3-decoder![online]. 2008. [cit. 12.4.2012] URL:<http://blog.bjrn.se/2008/10/lets-build-mp3-decoder.html>
- [5] Datasheet spoločnosti STMicroelectornics ku IO STA013 [online]. [cit. 12.4.2012] URL:http://www.st.com/internet/com/TECHNICAL_RESOURCES/TECHNICAL_LITERATURE/DATASHEET/CD00001694.pdf
- [6] Datasheet spoločnosti Micronas ku IO MAS3578F [online]. [cit. 12.4.2012] URL:http://www.inf.ethz.ch/personal/hvogt/proj/btmp3/Datasheets/mas3587f_2pd.pdf
- [7] Datasheet spoločnosti Micronas ku IO MAS3507D [online]. [cit. 12.4.2012] URL:<http://www.mas-player.de/mp3/download/mas3507d.pdf>
- [8] Datasheet spoločnosti ROHM ku IO BU9438KV [online]. [cit. 12.4.2012] URL:<http://www.farnell.com/datasheets/357181.pdf>
- [9] Datasheet spoločnosti ROHM ku IO BU4328KV [online]. [cit. 12.4.2012] URL:<http://www.farnell.com/datasheets/357177.pdf>
- [10] Datasheet spoločnosti VLSI ku IO VS1053 [online]. [cit. 12.4.2012] URL:<http://www.vlsi.fi/fileadmin/datasheets/vlsi/vs1053.pdf>
- [11] Datasheet spoločnosti VLSI ku IO VS1063 [online]. [cit. 12.4.2012] URL:<http://www.vlsi.fi/fileadmin/datasheets/vs1063ds.pdf>
- [12] VAŠÍČEK, Zdeněk, Hardware - FITkit [online], [cit. 15.4.2012]. URL:<http://merlin.fit.vutbr.cz/FITkit/hardware.html>
- [13] VAŠÍČEK, Zdeněk, Hardware/FPGA [online], [cit. 15.4.2012]. URL:http://merlin.fit.vutbr.cz/FITkit/docs/hardware/hw_fpga.html

- [14] VAŠÍČEK, Zdeněk, Hardware/MCU [online], [cit. 15.4.2012].
URL:http://merlin.fit.vutbr.cz/FITkit/docs/hardware/hw_mcu.html
- [15] VAŠÍČEK, Zdeněk, Firmware / Komunikační systém [online], [cit. 15.4.2012].
URL:http://merlin.fit.vutbr.cz/FITkit/docs/firmware/fpga_interconnect.html
- [16] SCHWERDTFEGGER, Martin, SPI - Serial Peripheral Interface [online],[cit. 16.4.2012],
URL:<http://www.mct.net/faq/spi.html>
- [17] SD Specifications, Part 1: Physical Layer, Simplified Specification, Version 3.01 [online].
[cit. 18.4.2012],
URL:https://www.sdcard.org/downloads/pls/simplified_specs/Part_1_Physical_Layer_Simplified_Specification_Ver_3.01_Final_100518.pdf
- [18] VAŠÍČEK, Zdeněk, Firmware / Klávesnice 4x4 [online], [cit. 25.4.2012].
URL:http://merlin.fit.vutbr.cz/FITkit/docs/firmware/fpga_keyboard.html
- [19] VAŠÍČEK, Zdeněk, Firmware / Řadič LCD displeje [online], [cit. 25.4.2012].
URL:http://merlin.fit.vutbr.cz/FITkit/docs/firmware/fpga_lcd.html
- [20] Schéma zapojenia hardwarovej časti platformy FITkit [online], [cit. 26.4.2012].
URL:http://merlin.fit.vutbr.cz/FITkit/download/schematic_v20.pdf

Príloha A: Schéma zapojenia modulu



Príloha B: Obraz dosky plošných spojov



Príloha C: CD so zdrojovými súbormi