

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

SIGNÁLOVÝ ANALYZÁTOR NA BÁZI UNIVERZÁLNÍHO USB DVB-T
PŘIJÍMAČE

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

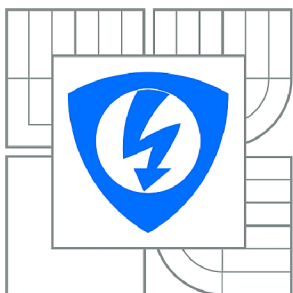
JAKUB HOFEREK

BRNO 2015



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

SIGNÁLOVÝ ANALYZÁTOR NA BÁZI UNIVERZÁLNÍHO USB DVB-T PŘIJÍMAČE

SIGNAL ANALYZER BASED ON UNIVERSAL USB DVB-T RECEIVER

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

JAKUB HOFEREK

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Dr. Techn. VOJTĚCH DERBEK

BRNO 2015



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav radioelektroniky

Bakalářská práce

bakalářský studijní obor
Elektronika a sdělovací technika

Student: Jakub Hoferek

ID: 159200

Ročník: 3

Akademický rok: 2014/2015

NÁZEV TÉMATU:

Signálový analyzátor na bázi univerzálního USB DVB-T přijímače

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s možnostmi využití USB DVB-T přijímače jako univerzálního přijímače komplexního IQ signálu. Analyzujte jeho možnosti pro využití jako softwarového rádia pro zobrazování frekvenčního spektra, detekci bitů a paketů (RTL-SDR). Navrhněte algoritmy pro zpracování signálů Vámi vybraného komunikačního protokolu.

Implementujte SW rozhraní pro komunikaci mezi Vámi zvoleným prostředím pro zpracování signálu (např. Labview, Matlab, GNU Radio) a USB. Implementujte navržené funkce a algoritmy (jednoduchý spektrální analyzátor a dekodér QPSK signálu).

DOPORUČENÁ LITERATURA:

[1] REED, J.H., Software Radio: A Modern Approach to Radio Engineering, Prentice Hall, 2002.

[2] OsmocomSDR [online]. [cit. 11.5.2014]. Dostupné na <http://sdr.osmocom.org/trac/wiki/rtl-sdr>.

Termín zadání: 9.2.2015

Termín odevzdání: 13.8.2015

Vedoucí práce: Ing. Dr. Techn. Vojtěch Derbek

Konzultanti bakalářské práce:

doc. Ing. Tomáš Kratochvíl, Ph.D.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské 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 semestrální práce se zabývá využitím běžně dostupného USB DVB-T tuneru pro analýzu spektra. V první části jsou zde srovnány jiné možné platformy pro příjem, popsán hardware konkrétního přijímače s tunerem R820T a AD převodníkem RTL2832. V další části je realizován spektrální analyzátor a jednoduchý FM přijímač. Tyto funkční celky jsou vytvořeny na platformě GNU Radio i Matlab.

KLÍČOVÁ SLOVA

Softwarově definované rádio, SDR, RDS, RTL2832, R820T, E4000, USB DVB-T tuner, Matlab, GNU Radio, spektrální analýza.

ABSTRACT

This thesis deals with using a commercially available USB DVB-T tuner for spectrum analysis. In the first part, there are other possible platforms compared to income described hardware specific receiver tuner R820T and AD converter RTL2832. The next part is realized by a spectrum analyzer and a simple FM receiver. These functional units are created on GNU Radio and Matlab.

KEYWORDS

Software defined radio, SDR, RDS, RTL2832, R820T, E4000, USB DVB-T tuner, Matlab, GNU Radio, spectral analyse.

HOFEREK, Jakub *Signálový analyzátor na bázi univerzálního USB DVB-T přijímače*: bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav radioelektroniky, 2015. 44 s. Vedoucí práce byl Ing. Dr. Techn. Vojtěch Derbek,

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Signálový analyzátor na bázi univerzálního USB DVB-T přijímače“ jsem vypracoval samostatně pod vedením vedoucího bakalářské 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 bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské 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 autorského 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.

Brno

.....

(podpis autora)

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu Ing. Dr. Techn. Vojtěchu Derbekovi, za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

(podpis autora)

OBSAH

Úvod	9
1 Teoretická část studentské práce	10
1.1 Softwarové zpracování	10
1.1.1 Vzorkování a kvantování signálu	10
1.1.2 FFT	10
1.1.3 Filtrace signálů	10
1.2 Rozdělení číslicových rádiových přijímačů	11
1.2.1 Přijímače s číslicovým zpracováním v základním pásmu	11
1.2.2 Přijímače s číslicovým zpracováním v mezifrekvenčním pásmu	11
1.2.3 Přijímače s číslicovým zpracováním na kmitočtu vstupního signálu	11
1.3 Další HW platformy pro realizaci SDR rádia	12
1.3.1 DVB-T tuner s RTL2832	12
1.3.2 FUNcube Dongle Pro+	12
1.3.3 USRP	12
1.3.4 HackRF	13
1.3.5 Perseus	13
1.3.6 Použití SDR v radiostanici Baofeng	13
1.3.7 Použití SDR v radiostanici Yaesu FT-450	14
1.3.8 Použití DSP v radiostanici Yaesu FT-857	14
2 Popis přijímače	15
2.1 Blokové schéma přijímače	15
2.2 Tuner R820t	15
2.2.1 AD převodník a demodulátor RTL2832	17
2.2.2 Anténa	17
2.2.3 Frekvenční stabilita	17
2.2.4 Proudová spotřeba	17
3 RDS protokol	18
3.1 Popis systému RDS	18
3.2 Implementace systému RDS	18
4 Širokopásmová analýza rádiového spektra	22
4.1 Širokopásmová spektrální analýza řešená jazykem Python	22
4.2 Možnosti programu	22
4.3 Popis skriptu v Pythonu	22

4.4	Širokopásmová spektrální analýza pomocí Matlabu	23
5	Graficky znázorněné softwarové zpracování	25
5.1	Zpracování IQ dat v Matlabu s využitím Simulinku	25
5.1.1	Spektrální analyzátor	25
5.1.2	FM přijímač	27
5.2	Zpracování IQ dat v GNU Radiu	28
5.2.1	Spektrální analyzátor	28
5.2.2	FM přijímač	31
6	Demonstrace dalších možností příjmu s tunerem RTL	32
6.1	Zpracování signálu na platformě Raspberry PI B+	32
6.1.1	Realizace FM přijímače na platformě Raspberry PI	32
6.2	Zpracování signálu na platformě Android	33
7	Závěr	34
	Literatura	35
	Seznam symbolů, veličin a zkratk	37
	Seznam příloh	38
A	Fotografie použitého DVB-T tuneru	39
A.1	Osazaná deska plošného spoje tuneru	39
A.2	Přijímač v původním plastovém krytu	39
B	Zdrojový kód v jazyce Python pro přehledový spektrální analyzátor	40

SEZNAM OBRÁZKŮ

2.1	Obecné blokové schéma DVB-T přijímače	16
2.2	Blokové schéma tuneru R820t	16
3.1	Spektrum rádiového signálu po FM demodulaci	19
3.2	Průběh signálu po DSB demodulaci	20
3.3	Obnova nosného kmitočtu 57 kHz DSB modulace	20
3.4	Blokové schéma v Simulinku	21
4.1	Širokopásmová spektrální analýza 100 MHz - 110 MHz, využitím ja- zyka Python	24
4.2	Širokopásmová spektrální analýza v Matlabu pro rozsah 100 - 110 MHz	24
5.1	Spektrální analyzátor v Simulinku.	26
5.2	Spektrální analýza.	26
5.3	FM přijímač v Simulinku.	27
5.4	Spektrální analýza v GNU radiu.	29
5.5	Okamžité spektrum.	30
5.6	Průměrné spektrum.	30
5.7	FM přijímač realizovaný v prostředí GNU radia.	31
6.1	Přijímač na mobilním telefonu se systémem Android.	33
A.1	Přijímač bez krytu	39
A.2	Přijímač s krytem	39

ÚVOD

Technologie softwarově definovaného rádia je v současné době perspektivní a velké množství analogově řešených komunikačních systémů postupně na tuto technologii přechází. SDR (softwarově definované rádio – Software Defined Radio) umožňuje využívat nové způsoby modulace a zpracování signálů, jež by v analogové formě buď nebylo možné zrealizovat, nebo by to bylo velmi obtížné.

Tato práce bude využívat na trhu běžně dostupný USB DVB-T přijímač. Tento sice nevykuká svými rádiovými vlastnostmi, ale svou cenou se však řadí mezi nejdostupnější SDR na trhu. Prvotní určení - použití jen jako DVB-T, DAB, FM přijímač, však může díky své variabilitě rozšířit na zpracování signálů rozsahu 24 - 1766 MHz, s šířkou pásma do 2,4 MHz. To vše díky použité knihovně pro přímý přenos dat z ADC v RTL2832u do PC. Mezi hlavní nevýhody tohoto přijímače se řadí malá citlivost, ta je dána i mimo jiné i nízkým rozlišením vstupního ADC - 8 bitů. Při vyšším nastavení zisku LNA - nízkošumového zesilovače se taktéž zvýrazní parazitní zrcadlové signály v okolí přijímaného kmitočtu.

Přijímač navíc projevil velkou náchylnost k zahlcení blízkým silným signálem, tato vlastnost se projeví ztlumením i jinak silných signálů v přijímaném spektru. Dále, pokud se silný signál objeví přímo v přijímaném pásmu, zareaguje na něj systém AGC a okolní signály ztratí na amplitudě úměrně snížení zesílení. Tento předzesilovač zatím nelze plně manuálně ovládat.

Kvalitě příjmu se taky projeví velmi jednoduchá vstupní část. Hlavní selektivitu přijímače tvoří integrovaný široce přeladitelný vstupní filtr. Horší parametry přijímače je možné částečně kompenzovat vhodnou ziskovou, laděnou anténou, nicméně ne vždy je toto řešení reálné.

1 TEORETICKÁ ČÁST STUDENTSKÉ PRÁCE

V teoretické části práce budou objasněny principy softwarového zpracování dat a základní rozdělení číslicových přijímačů. Dále zde budou uvedeny příklady již používaných a aktuálně dostupných konceptů SDR přijímačů.

1.1 Softwarové zpracování

Pro zpracování se v konkrétních aplikacích nejčastěji používají DSP, či FPGA. V této práci bude ovšem využito méně efektivní metoda zpracování pomocí osobního počítače. V takovém případě se uplatní vysoký výpočetní výkon kompenzující neoptimální distribuci rutinních výpočtů. Toto řešení ovšem nabízí velkou variabilitu a snadnou modifikaci kódu.

1.1.1 Vzorkování a kvantování signálu

Prvním krokem při převodu do číslicové formy je vzorkování spojitého analogového signálu, diskretizace v časové oblasti. Rychlost vzorkování odpovídá vzorkovací frekvenci. Dle Shannon-Kotělnikova teorému musí být vzorkovací frekvence dvakrát vyšší než maximální frekvence užitečného signálu.

Dále se souběžně provádí kvantování na konkrétní hladinu v AD převodníku. Jedná se o diskretizaci signálu v oboru jeho hodnot. Jemnost kvantování je dána rozlišením AD převodníku - množstvím hladin na které je možno kvantovat vstupní signál.

1.1.2 FFT

Rychlá Fourierova transformace je algoritmus pro vypočítání diskrétní Fourierovy transformace. Tato funkce je základem dále v práci popsaného spektrálního analyzátoru.

1.1.3 Filtrace signálů

Filtrace signálů je jednou ze základních operací při zpracování signálu, používá se pro omezení šířky přenášeného pásma. V číslicové oblasti zpracování signálů se používají především dva typy filtrů - nerekurzivní FIR s konečnou impulzní charakteristikou a rekurzivní filtr IIR s nekonečnou impulzní charakteristikou.

1.2 Rozdělení číslicových rádiových přijímačů

Číslicové rádiové přijímače se z hlediska způsobu zpracování signálu dělí na 3 základní typy. Tyto se liší zejména frekvenčním pásmem ve kterém se provádí vzorkování a převod analogového signálu na číslicový, se kterým se dále pracuje. Toto pásmo odpovídá frekvenci jíž je užitečný signál vzorkován. Postupně s pokračujícím vývojem AD převodníků a DSP procesorů se nyní dosahuje vzorkovací frekvence stovek MHz, adekvátně k této hodnotě musí být použit i dostatečně rychlý DSP či jiný systém pro zpracování tak velkého toku dat.

V návaznosti se zvyšujícími-se nároky na číslicovou část přijímače však klesají požadavky na analogovou část. Jedná se zejména o mezifrekvenční filtry a směšovače, jež tvoří podstatné bloky rádiových zařízení. Taktéž se podílí významnou měrou na výsledné kvalitě přijímače.

1.2.1 Přijímače s číslicovým zpracováním v základním pásmu

Blokové schéma přijímače je analogické k homodynu. Přijímaný signál prochází přes vstupní, filtr přímo do kvadraturního detektoru. Zde je užitečný signál ve směšovači konvertován do základního pásma ve kterém je vzorkován a kvantován AD převodníkem. Další zpracování je již řešeno softwarově.

Zpracování v základním pásmu vyniká jednoduchostí a minimálním počtem použitých prvků, ovšem je problém realizovat tak velké zesílení na nízkých frekvencích, problém s napěťovým driftem zesilovače v základním pásmu.

1.2.2 Přijímače s číslicovým zpracováním v mezifrekvenčním pásmu

Tento typ přijímače kombinuje výhody obou předchozích systémů. Signál je dostatečně vyfiltrován a zesílen analogovou cestou, nejsou proto tak vysoké nároky na AD převodníky. Zesílení je největší v mezifrekvenčním pásmu.

1.2.3 Přijímače s číslicovým zpracováním na kmitočtu vstupního signálu

Tento typ přijímačů se zatím v běžném použití mnoho neprosadil z důvodu nedostatku dostatečně rychlých a levných obvodů pro zpracování. Rádiová komunikace probíhá převážně na tak vysokých kmitočtech kde by se přímé vzorkování signálu provedlo jen s obtížemi, nebo by to vůbec nebylo možné. Ovšem postupně vývoj nezadržitelně směřuje tímto směrem.

Přijímače se zpracováním na kmitočtu vstupního signálu však lze bez větších obtíží realizovat v pásmu krátkých vln. Ovšem aplikace v tomto rádiovém pásmu se omezují především pro armádní využití. V civilní sféře by zde mohly být realizováno SDR pro příjem rozhlasu digitálního či analogového, nicméně stále je levnější realizace konvenčním plně analogovým způsobem. Jen v případě technologie DRM je vynuceno použití číslicového zpracování a i to se provádí většinou na nižší mezifrekvenci.

1.3 Další HW platformy pro realizaci SDR rádia

Ve světě je mnoho použitelných HW platforem, například zde uvedu 6 přijímačů pro stručné srovnání :

1.3.1 DVB-T tuner s RTL2832

Tento tuner jako jediný z níže uvedených je v další části této práce využít. Šířka pásma dosahuje až 3,2 MHz, nicméně používat bez rizika ztráty dat je možno jen 2,4 MHz. Frekvenční rozsah je dán tunerem R820T a to 24 - 1766 MHz. AD převodník je jen 8 bitový kvadrurní. Jeho hlavní výhoda je v ceně přibližně 180 Kč za kus a dostupnosti na běžném trhu.

1.3.2 FUNcube Dongle Pro+

Tento SDR přijímač ve velikosti USB flash disku, je určený primárně pro snadný příjem telemetrie ze satelitů řady CubeSat, jež se nyní hojně vypouští na oběžnou dráhu Země. Pro přenos se používá nejčastěji radioamatérské pásmo 2m, nebo 70cm.

U CubeSatu používaná modulace AFSK je spolu s protokolem AX.25 velmi úzkopásmová a proto není na škodu nižší vzorkovací frekvence přijímače. Rozlišení ADC je naopak velmi dobré což přispívá k vynikající citlivosti. A nakonec frekvenční rozsah předurčuje toto zařízení k velmi širokému poli použití. Cena se pohybuje řádově kolem 4500 Kč.

1.3.3 USRP

USRP je rodina univerzálních SDR přijímačů. Využívají se pro výzkum, vývoj, výuku a jsou velmi kvalitně zpracovány. V závislosti na požadavcích může tato platforma obsahovat FPGA, nejen VF vstup(přijímač), ale i VF výstup(vysílač). Cena těchto zařízení se však pohybuje v rádech desítek tisíc korun.

1.3.4 HackRF

Toto zařízení má také číslicový přijímač a i vysílač. Frekvenční rozsah 10 MHz - 6 GHz. Nicméně AD převodník je jen 8 bitový a částka za pořízení se pohybuje kolem 6300 Kč.

1.3.5 Perseus

Perseus je krátkovlnný přijímač s šířkou pásma 40 MHz a rozlišením ADC 14 bitů. Uvnitř obsahuje jako jediný 10 reléově přepínaných vstupních filtrů, které zajišťují citlivost a odolnost přijímače za všech podmínek. Zvláště odolnost je v dnešní době velmi důležitá v silně zarušeném městském prostředí. Cenově zařízení vychází na 25000 Kč.

1.3.6 Použití SDR v radiostanici Baofeng

V roce 2010 se objevila na trhu nová řada radiostanic od firmy Baofeng, tato se vyznačovala velmi nízkou cenou (řádově 700,-). Pro snížení výrobních nákladů je zde použit integrovaný obvod RDA1846. Tento v sobě integruje kompletní přijímací a vysílací obvody radiostanice. Jako podpůrné vnější obvody využívá pouze výkonový VF zesilovač a velmi jednoduché vstupní filtry, jež jsou zároveň hlavní slabinou radiostanice. Tato je sice dostatečně citlivá, nicméně reaguje na velké spektrum průmyslových rušení a v oblasti měst je její použití často problematické. Vlastnosti

integrovaného obvodu RDA1846

- Dvoupásmový přijímač/vysílač 134 - 174 MHz a 400 - 500 MHz.
- Použití s šířkou pásma 12,5 kHz a 25 kHz.
- Integrovaný AFC, AGC, VOX, indikátor úrovně, squelch.
- Integrovaná dvojtónová selektivní volba DTMF pro příjem i vysílání(tato funkce ovšem není ve firmwaru radiostanice běžně implementována).

1.3.7 Použití SDR v radiostanici Yaesu FT-450

Dalším příkladem využití technologie SDR je moderní krátkovlnná radiostanice FT-450 od firmy Yaesu. Zde SDR zastupuje hlavní roli při filtrování a demodulaci signálu. Radiostanice i přes průkopnickou pozici v koncepci a relativně nízkou cenu má dobré vlastnosti a obstojí i v dnešních náročných podmínkách rušení v krátkovlnném spektru.

1.3.8 Použití DSP v radiostanici Yaesu FT-857

Na rozdíl od předchozí radiostanice FT-450, je zde digitální zpracování použito jen pro filtraci a demodulaci je řešena analogově. Radiostanice je koncipovaná konvenčně - s čistě analogovým zpracováním signálu a modul DSP je zde jen jako doplněk. Je možná aktivovat 3 funkce :

- Omezovač šumu, omezí především frekvenční rozsah na vyšších kmitočtech kde se šum projevuje výrazněji pro lidské ucho.
- Notch filtr - úzkopásmová zadrž proti nežádoucímu frekvenčnímu zázněji.
- Pásmový filtr, omezuje rušení v blízkém okolí komunikačního spektra, lze nastavit dle potřeby, obvykle 300 Hz až 2,4 kHz.

2 POPIS PŘIJÍMAČE

První verze verze přijímače používala konfiguraci s tunerem E4000 s nulovou mezi-frekvencí, firma Elonics jež tyto obvody vyráběla se ale dostala do úpadku.

Nástupcem se stal tuner R820t s nímž se přijímače vyrábějí dodnes. Následující kapitola konkrétně popíše strukturu tohoto zde použitého DVB-T tuneru. Oproti tuneru Elonics se parametry přijímače mírně zlepšily, zejména se snížila výrazně přítomnost stejnosměrné složky.

Poslední revize přijímače obsahuje tuner R820T2 jež se vyznačuje o něco lepšími parametry než testovaný model. Zlepšila se zde i stabilita a přesnost naladění použitím precizního teplotně kompenzovaného oscilátoru.

2.1 Blokové schéma přijímače

Přijímač na obr. 2.1 se skládá především ze dvou integrovaných obvodů, rádiového tuneru R820t a rychlého AD převodníku s demodulátorem a rozhraním USB umožňující i přímý přenos navzorkovaných I a Q dat do počítače. Oba obvody mají společný zdroj hodin z oscilátoru při RTL2832u, rezonujícího krystalem 28,8 MHz.

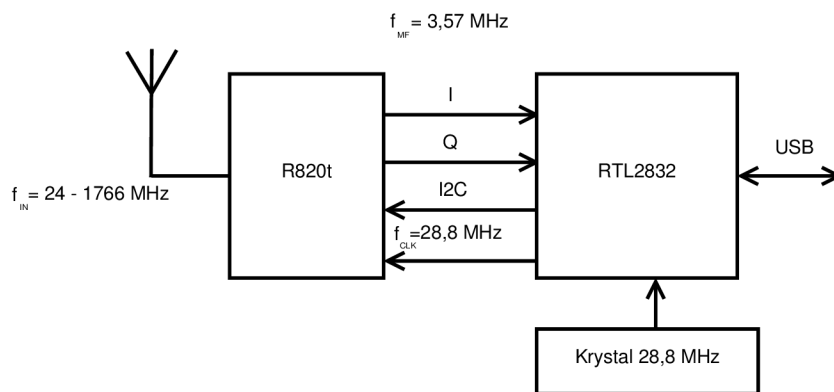
Směrem z tuneru do AD převodníku je symetricky veden I a Q signál. Opačným směrem je tuner řízen přes sběrnici I2C, zde se řídí zisk LNA tuneru, přijímaná frekvence a šířka MF filtru = přijímaného pásma. Zařízení může pracovat ve dvou

základních režimech:

- Zpracování dat přímo v obvodu RTL2832u. Tento systém je využit pouze při zpracování DVB-T signálu, který má šířku jednoho kanálu 8 MHz. Takto široké spektrum nelze tímto obvodem přímo přenést do počítače. Navíc by byl počítač neúměrně zatěžován rutinními výpočty pro demodulaci signálu, proto se tato část řeší již uvnitř obvodu a ven po USB sběrnici se posílají jen digitální demodulované data.
- Tato práce se zabývá především druhou možností - posíláním surových navzorkovaných IQ dat přímo do počítače a jejich následným zpracováním výhradně procesorem počítače. Tato vlastnost není podporována výrobcem přijímače, ale jen podpůrnými balíčky a knihovnami.

2.2 Tuner R820t

Na vstupu část přijímače, pro připojení dodávané antény je přítomen miniaturní konektor MCX. Za konektorem následuje ochrana proti elektrostatické elektřině zajištěná dvěma antiparalelně zapojenými diodami. Pokud se na vstupu přijímače objeví

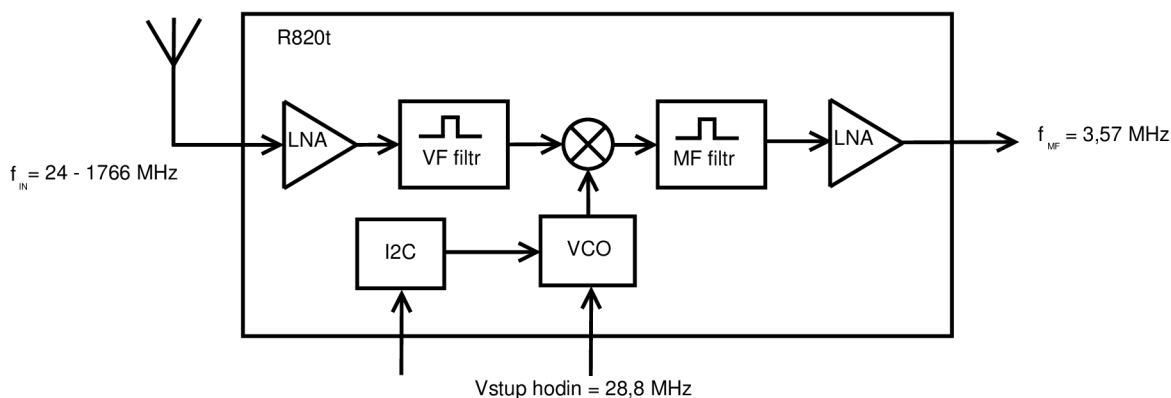


Obr. 2.1: Obecné blokové schéma DVB-T přijímače

napětí větší než difuzní napětí diod (0.2 - 0.7 V) tak je jimi bezpečně staženo do země. Poté již je přímo připojen vstup obvodu tuneru R820T.

Tento obvod obr. 2.2 je dobře zdokumentován v technickém listu výrobce. Jeho vstupní frekvenční rozsah 24 - 1766 MHz se konvertuje směrem dolů na mezifrekvenci 3.57 MHz. (Pro starší tuner E4000 je mezifrekvenční kmitočet nulový). Výstup jsou dvě symetrické linky - pro I a Q složku signálu.

Na vstupu obvodu je nízkošumový zesilovač s řízením zisku, poté následuje integrovaný vstupní vysokofrekvenční filtr jež využívá jedinou nutnou vnější indukčnost. Hlavním prvkem tuneru je směšovače, zde vznikne užitečný signál na mezifrekvenčním kmitočtu a několik parazitních signálů. Proto je poté zařazen mezifrekvenční filtr s proměnnou šířkou pásma dle aktuálně vzorkovaného signálu. Poslední část tvoří opět nízkošumový zesilovač, zisk je řízen úrovní přivedeného stejnosměrného napětí na vývodu 14.označeného VAGC integrovaného obvodu R820t. Jeho úkolem je kompenzovat kolísající úroveň zpracovávaného signálu.



Obr. 2.2: Blokové schéma tuneru R820t

2.2.1 AD převodník a demodulátor RTL2832

Tento obvod bohužel nemá veřejně k dispozici datový list s jeho technickými parametry, jako zdroj informací je možné použít pouze jednoduchý popis výrobce.

2.2.2 Anténa

Dodávaná anténa má magnetickou základnu pro upevnění na feromagnetický materiál, k němuž je poté kapacitně připojení. Feromagnetický materiál na který se anténa připevní pak slouží jako elektrická protiváha k zářiči. Zářič je 10 cm dlouhý ocelový a je možné ho přes šroub na základně demontovat či vyměnit.

V této práci ovšem byl pro účely měření použit otevřený symetrický dipól s délkou ramene 75 cm, rezonující v okolí frekvence 100 MHz. Toto frekvenční pásmo bylo zvoleno z důvodu velké obsazenosti rozhlasovými stanicemi na kterých se jasně demonstuje frekvenční analýza rádiového spektra a RDS. Anténa se chová při příjmu velmi širokopásmově a lze ji použít v celém frekvenčním rozsahu přijímače.

2.2.3 Frekvenční stabilita

Dlouhodobá frekvenční stabilita závisí na teplotě, dva podobné přijímače mají při pokojové teplotě chybu oscilátoru okolo 30ppm. Tuto chybu je možno kompenzovat buď pomocí softwaru, nebo lépe hardwarovým řešením. V tom případě by se místo krystalového výbrusu na 28,8 MHz osadil teplotně kompenzovaný oscilátor. Stabilita pak může dosahovat hodnot jednotek ppm.

2.2.4 Proudová spotřeba

Přijímač má v pohotovosti proudový odběr 110 mA při napájecím napětí 4,8 V . Na přijímané frekvenci 100 MHz vykazuje při práci proudový odběr 310 mA při 4,7 V napájecím napětí, tento proud je nezávislý na nastaveném zisku LNA, mírná závislost lze pozorovat při změně vzorkovací frekvence. Výkonová ztráta je poté cca 1.4W, tato je primárně rozdělena mezi dva SMD integrované obvody.

Při vzorkovací frekvenci 0,25 MHz přijímač odebírá proud 0,29 A a při vzorkovací frekvenci 2,4 MHz se proud zvýší na 0,31 A. Přijímač se při příjmu taktéž dost zahřívá, až na teplotu cca 52°C.

3 RDS PROTOKOL

Po konzultaci s vedoucím práce byl pro implementaci digitálního protokolu zvolen systém RDS s DBPSK modulací. Důvody pro zvolení tohoto systému jsou především:

- Přenos dat je realizován v otevřené (nešifrované) formě, lze spočítat kontrolní součet a tím ověřit korektní práci s daty.
- Pokrytí rozhlasovými FM stanicemi s RDS systémem je velmi dobré a stabilní.

3.1 Popis systému RDS

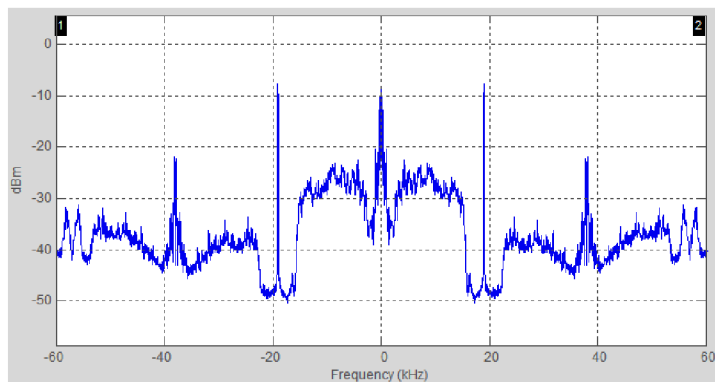
Systém (Radio Data System) je nástupce staršího systému ARI (Autofahrer-Rundfunk-Informationssystem) jež byl uveden do provozu v Německu roku 1972. ARI byl schopen pouze detekovat přítomnost dopravního hlášení a rádiový přijímač následně zareagoval zvýšením hlasitosti, či utlumením hudby z kazetového přehrávače. Subnosná systému ARI byla umístěna na kmitočtu 57 kHz namodulovaném na hlavní nosný kmitočet. Přítomnost signálu o frekvenci 125 Hz na této subnosné pak jednoduše indikovala prioritní dopravního hlášení.

V současné době se používá již pouze systém RDS, který je moderním nástupcem ARI. RDS je již schopen přenosu komplexních dopravních informací. Je zachována poloha subnosné na 57 kHz, tato je zvolena jako trojnásobek pilotního kmitočtu 19 kHz, což je užitečné pro obnovu subnosné vlny DSB modulace. Informace je diferencially kódována dvoustavovou fázovou BPSK modulací. Modulace BPSK je zvolena pro nízký požadovaný poměr S/N pro spolehlivé rozlišení symbolů a díky diferencially kódování nezáleží na polaritě průběhu.

Celé přenášené spektrum můžeme vidět na obr.: 3.1. Hlavní monofonní kanál se skládá ze součtu levého a pravého kanálu a je přenášen v pásmu do 15 kHz. Na 19 kHz je poté přítomen pilotní signál využíván pro obnovu nosné vlny RDS a stereo přenosu. Následuje úsek 23 kHz až 53 kHz (na 38 kHz je potlačená nosná vlna), ve kterém je přenášen diferencially pravá kanál odečtený od levého. Na konci spektra se pak nachází signál služby RDS s potlačenou nosnou umístěnou na 57 kHz a celkovou zabranou šířkou pásma 4,8 kHz.

3.2 Implementace systému RDS

Pro implementaci RDS dekodéru byl zvolen systém Simulink s názorným vyjádřením prováděných operací při zpracování vstupního signálu. Kompletní zapojení funkčních bloků RDS dekodéru je na obr.: 3.4 .



Obr. 3.1: Spektrum rádiového signálu po FM demodulaci

Signál rozhlasové stanice je zachycen z bloku přijímačem a poté prochází frekvenčním demodulátorem.

Následují úzkopásmové propusti:

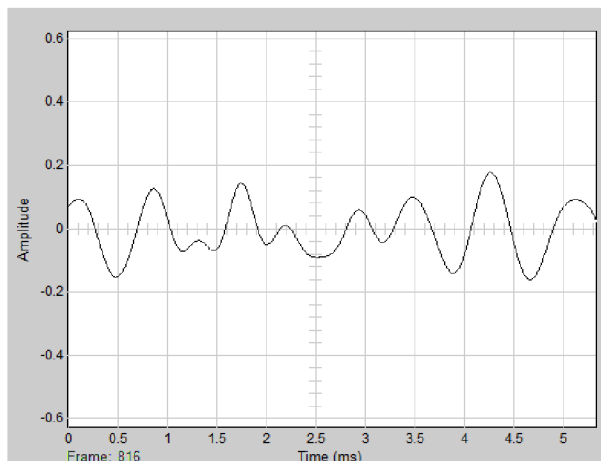
- Pro pilotní signál 19 kHz, tento slouží jako synchronizační prvek, za filtrem je vytvarován do obdélníkového tvaru s vyšším počtem harmonických složek. Pro dekodér symbolů BPSK je signál 16x vydělen čtyřmi paralelními děličkami. A pro demodulaci DSB modulace, je z tohoto spektra vybrána dalším úzkopásmovým filtrem složka o frekvenci 57 kHz.
- Propust pro 57 kHz přenáší samotnou DSB modulaci, následuje vynásobení s obnovenou nosnou harmonickou vlnou a výsledek

Datový přenos na obr.: 3.2 má bitovou rychlost 1187,5 b/s. Z obrázku je patrné že demodulace DSB signálu není zcela synchronní a proto kolísá amplituda demodulovaného signálu. Tuto chybu se v aktuálním zapojení nepodařilo odstranit.

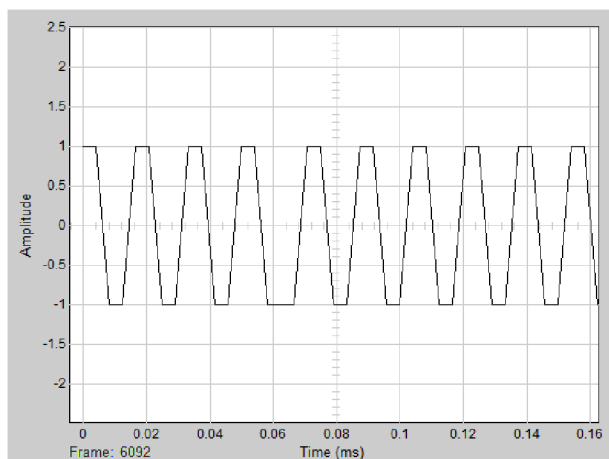
Zde na obr.:3.3 je zachycen signál na vstupu DSB demodulátoru, na první pohled není patrná výraznější změna fáze či frekvence v průběhu těchto deseti cyklů. Jedná se zřejmě o malou odchylku, tato se však při demodulaci výrazně projevuje a znemožňuje dále pracovat se signálem.

Pokud by byla synchronizace DSB demodulátoru v pořádku. Pokračovalo by zpracování dále do dekodéru BPSK symbolů.

Posledním krokem je spočtení CRC součtu a identifikování příslušných bloků. RDS zpráva o délce 104 bitů je rozdělena na 4 bloky po 26 bitech. Každý blok je pak složen ze dvou částí - 16 bitů informačního slova a 10 bitů kontrolního součtu přičtenému k offsetové hodnotě, jež rozlišuje blok zprávy (A, B, C, D). Přenos je realizován jako kontinuální bez přerušení a synchronizaci je nutné vypočítat. Nejprve

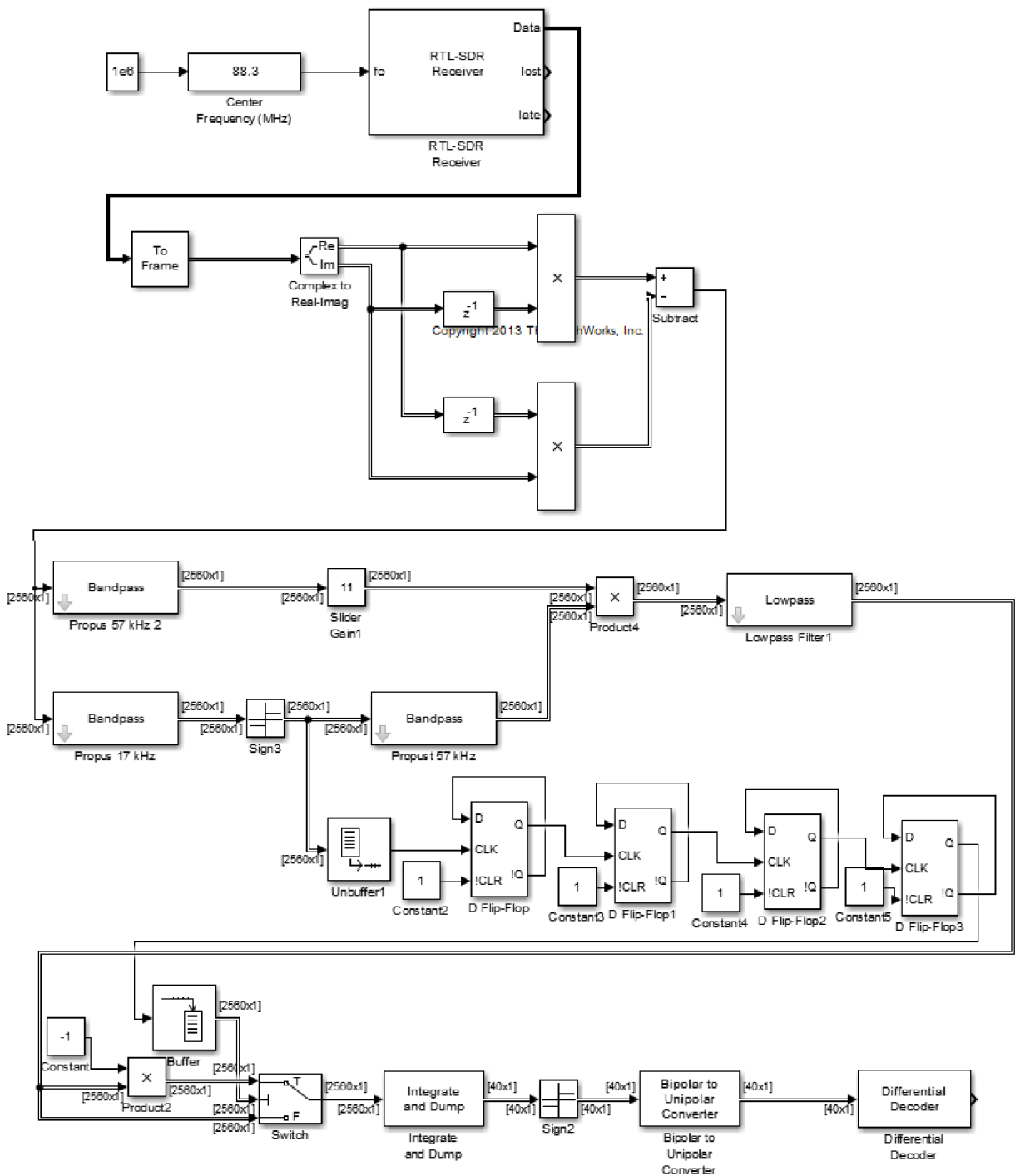


Obr. 3.2: Průběh signálu po DSB demodulaci



Obr. 3.3: Obnova nosného kmitočtu 57 kHz DSB modulace

se vypočte kontrolní součet prvních 16 bitů, poté se se postupně postupně přičítají offsetová slova pro skupiny A, B, C, D. Tento součet kontrolního součtu a offsetového slova se vždy porovná s následujícími 10 bity přijatých dat. Pokud nastane shoda pokračuje se následující sadou bitů. Pokud shoda nenastane, tak se program při výpočtu přesune o jeden bit ve směru nových přijatých bitů. Obdobně se postupuje dále, dokud není nalezena shoda.



Obr. 3.4: Blokové schéma v Simulinku

4 ŠIROKOPÁSMOVÁ ANALÝZA RÁDIOVÉHO SPEKTRA

4.1 Širokopásmová spektrální analýza řešená jazykem Python

Python je moderní multiplatformní vysokoúrovňový skriptovací jazyk, je vyvíjen a šířen jako open source projekt. Nevyniká sice tak propracovaným vývojovým prostředím jako Matlab, ovšem na druhou stranu je tento jazyk jednoduchý a rychlý. Disponuje taktéž propracovaným balíčkovým systémem knihoven funkcí. Celý kód je zveřejněn v příloze, na konci práce.

4.2 Možnosti programu

Program po spuštění požádá o vložení první a poslední frekvence na rozsahu jež chceme změřit. Dále pak je nutné zvolit jednu ze 4 přednastavených hodnot RBW kterou bude mít výsledná FFT transformace (RBW = 244 Hz, 61 Hz, 15 Hz a 1 Hz). Následuje dotaz na logaritmickou či lineární vertikální osu s hodnotami změřeného napětí, jednotky jsou pouze relativní.

Dále skript obsahuje tyto specifické funkce:

- Odečtení šumového pozadí. Provede dvě měření po sobě s časovým odstupem a tyto se vzájemně odečtou. Výsledkem je graf změn v rádiovém spektru, projeví se zde jak zvýšení amplitudy tak pokles.

Tato funkce lze využít pro srovnání dvou různých antén, jejich natočení. Nebo monitorovat vznik rušivých záznějů po zapnutí konkrétního přístroje.

- Monitorování spektra s upozorněním, pokud se vyskytne v prohledávaném úseku signál s vyšší než předem nastavenou hodnotou. Užitečná funkce pro vyhledávání nahodilých a krátkých rádiových relací, detekci a změření blízkých stanic s velmi občasným aktivním provozem. Funkce běží cyklicky stále dokola a průběžně vypisuje výsledky - frekvenci a amplitudu do příkazové řádky.

4.3 Popis skriptu v Pythonu

Skript napsaný v jazyce Python pro širokopásmovou spektrální analýzu využívá několika přídatných modulů které je nutno nainstalovat do základního prostředí Pythonu 2.7 .

Jedná se především o balíček RtlSdr zajišťující komunikaci s HW rozhraním přijímače. Dále byly použity knihovny pro matematické zpracování, FFT a zobrazení do grafu.

Následuje načtení vstupních hodnot jako je startovní a koncová frekvence FFT analýzy, zvolit se musí i jedna ze čtyř hodnot RBW v rozsahu 1 Hz až 244 Hz. Poté se definuje pole proměnných pro výsledné široké okno FFT. Obdobné pole se vytvoří s hodnotami frekvencí pro každý index pole - používá se k popisu frekvenční osy grafu a výpočtu reálných frekvencí ze znalosti indexu příslušné FFT hodnoty.

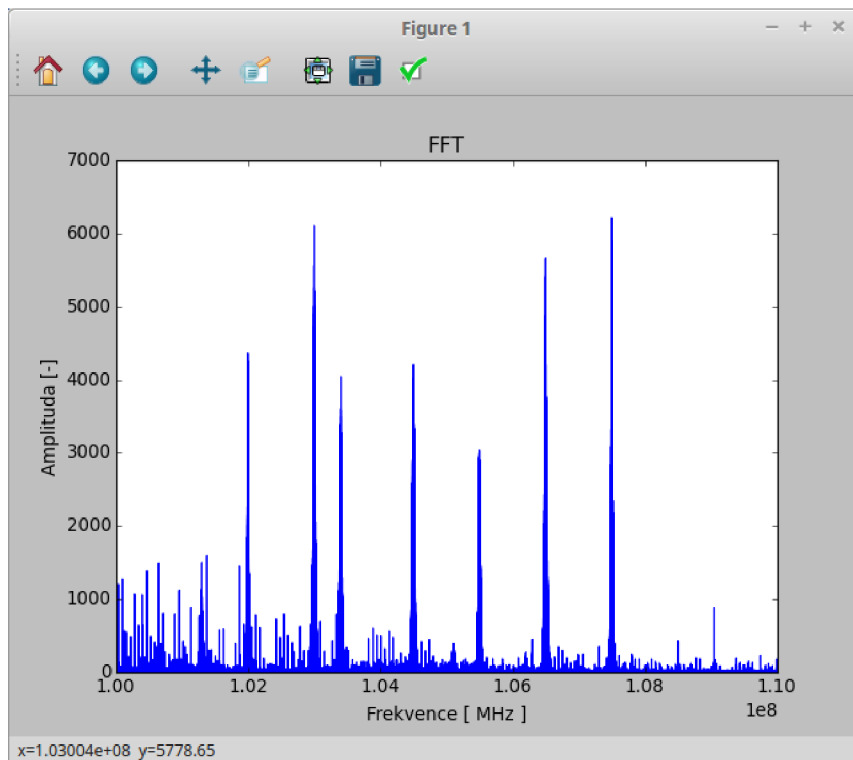
Hlavní část programu vzorkuje postupně okny 1,2 MHz širokými předem definovaný frekvenční rozsah. U každého malého okna se provede DFT transformace, převod do absolutních jednotek, spektrum se rotací posune na svůj střed. V této fázi je již malé okno FFT nachystané a postupně se přepisují jednotlivé indexy do hlavního výsledného okna široké FFT, protože je malé okno široké 1,2 MHz - dva boční vzorky o celkové šířce 100kHz se zanedbají (kvůli nevyrovnanosti okrajů spektra). Takto se postupně malé okna rádiových spekter přepisují do hlavního viz obr.4.1.

Proces analýzy spektra širokého 10 MHz s rozlišením 61 Hz trvá cca 4 sekundy, na rychlosti zpracování je zde vidět čistota a efektivita použitého jazyka.

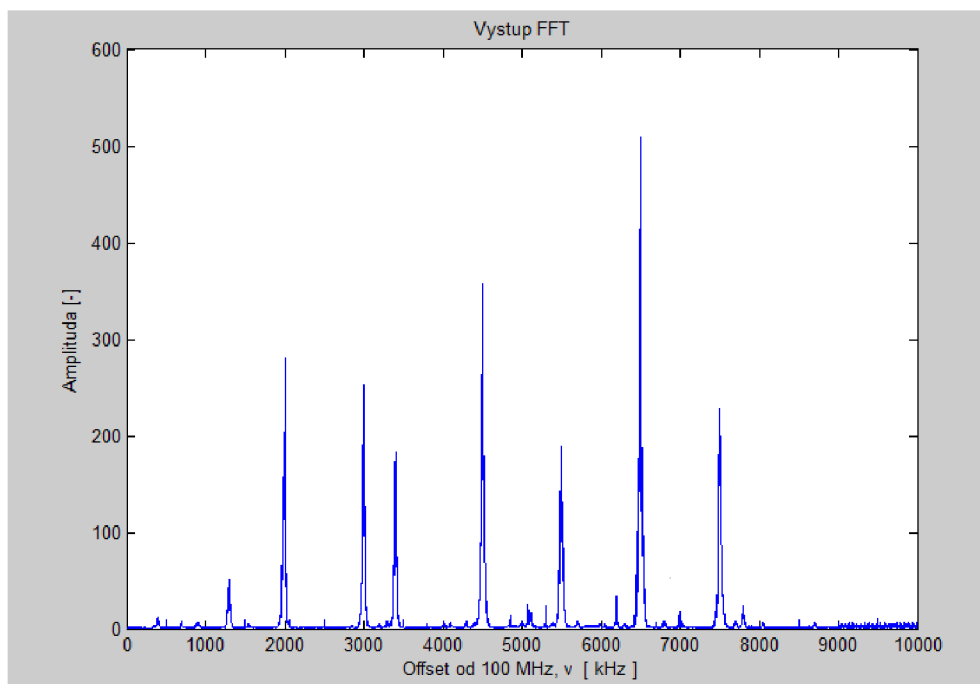
4.4 Širokopásmová spektrální analýza pomocí Matlabu

Obdobným postupem jako v předchozím případě se dosáhlo širokopásmové spektrální analýzy i v Matlabu viz obr.4.2. Navíc je zde řešeno průměrování vzorků na pevně dané $RBW = 1$ kHz. Výsledné spektrum má poté hladší průběh, tato vlastnost se ovšem projeví jen při bližším zkoumání frekvenční analýzy.

Při použití tohoto skriptu pro vytvoření spektrální analýzy na rozsahu 100 - 110 MHz ovšem celý proces získání vzorků trvá cca 19 sekund + následné 2 sekundy zpracování dat.



Obr. 4.1: Širokopásmová spektrální analýza 100 MHz - 110 MHz, využitím jazyka Python



Obr. 4.2: Širokopásmová spektrální analýza v Matlabu pro rozsah 100 - 110 MHz

5 GRAFICKY ZNÁZORNĚNÉ SOFTWAREVÉ ZPRACOVÁNÍ

Pro zpracování dat je možné využít mimo jiné tyto dvě rozdílné platformy:

- Matlab
- Gnuradio

Vyznačují rozdílným přístupem a strukturou. V současnosti je běžnější použití GNURadia založeného na platformě Python. Jedním z důvodů obliby použití s RTL SDR přijímačem je šíření tohoto softwaru pod GNU licenci. V rámci bakalářské práce byly sestaveny funkční bloky do následujících zapojení.

5.1 Zpracování IQ dat v Matlabu s využitím Simulinku

Matlab používá pro získání surových IQ dat knihovnu "libsub" nainstalovanou pomocí softwaru Zadig, ta spolupracuje přímo s RTL SDR přijímačem. Instalace podpory RTL-SDR přijímače se provádí formou balíčku Matlabu.

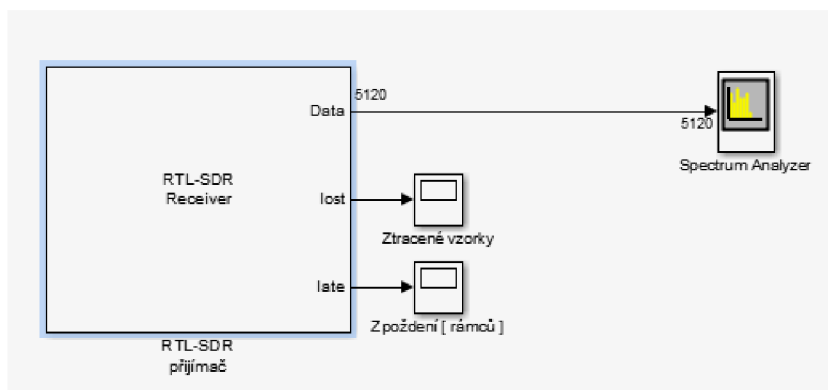
Pro zprovoznění SDR přijímače v Matlabu jsou potřeba tyto balíčky.

- Communications System Toolbox™
- DSP System Toolbox™
- Signal Processing Toolbox™

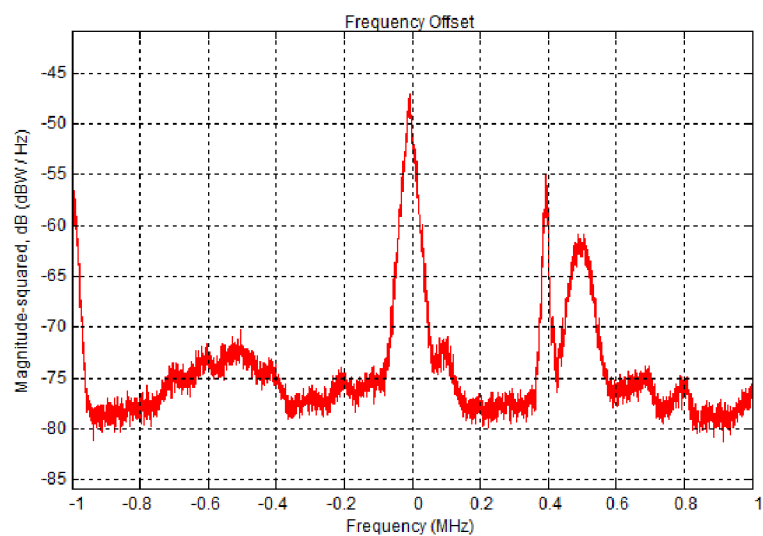
5.1.1 Spektrální analyzátor

Spektrální analyzátor spočívá v propojení bloků zdroje signál (DVB-T tuner) a zobrazovacího prvku s vykonáním FFT obr.5.1.

Zobrazované spektrum obr.5.2 má střední frekvenci 103 MHz a šířku pásma 2 MHz, v této oblasti se nachází na frekvenci 103,4 MHz rádiová rozhlasová stanice Petrov a na frekvenci 103 MHz rádiová rozhlasová stanice Krokodýl. Je vidět že rozhlasová stanice Krokodýl má výrazně silnější signál. Oproti tomu, se v blízkosti frekvence stanice Petrov nachází jiný zdroj rušení, nebo jiná rozhlasová stanice.



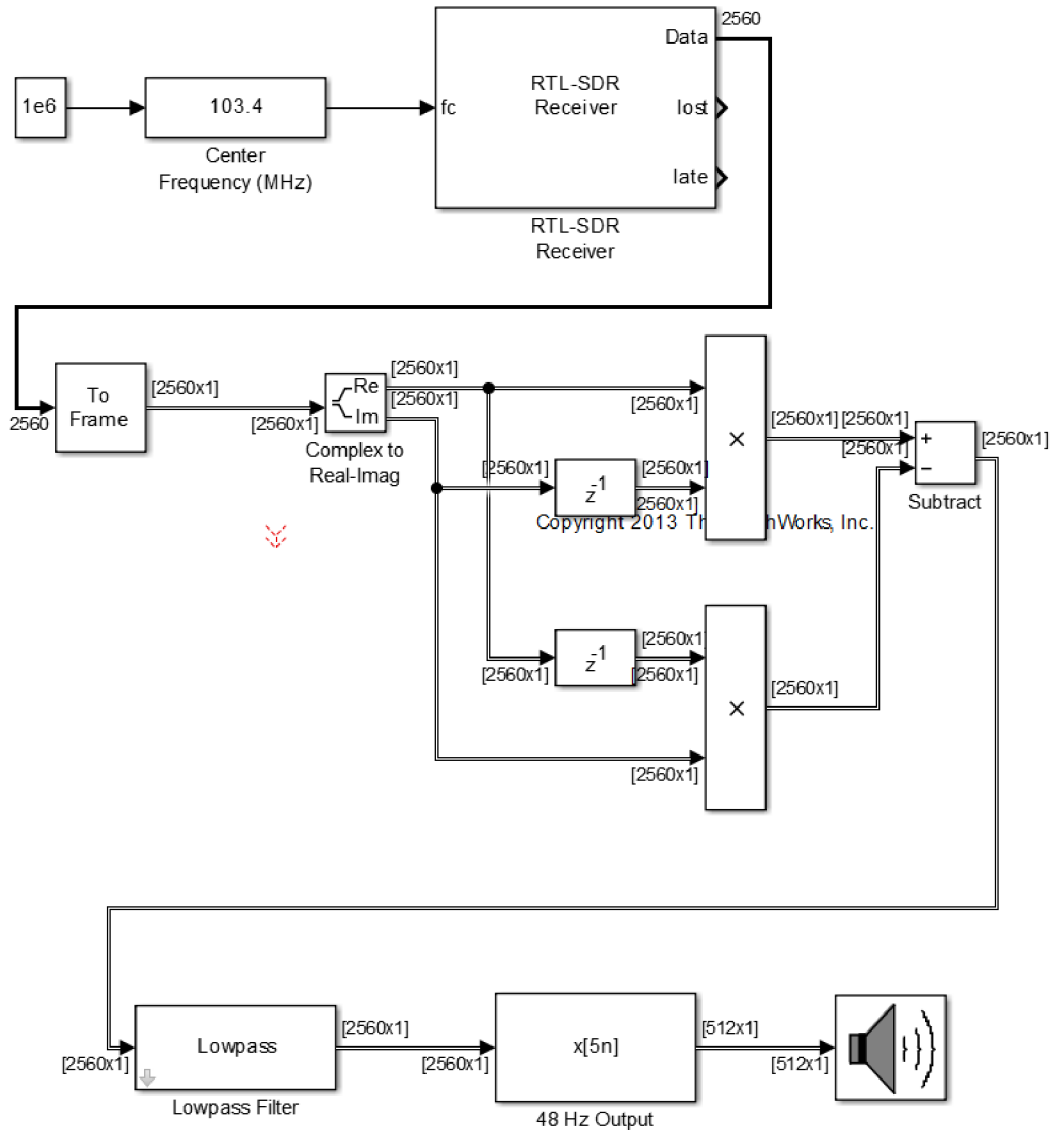
Obr. 5.1: Spektrální analyzář v Simulinku.



Obr. 5.2: Spektrální analýza.

5.1.2 FM přijímač

V simulinku realizovaný přijímač je znázorněn na obr. 5.3. Přijímanou centrální frekvenci je možné za běhu programu měnit a ladit tak přijímač na potřebné rozhlasové stanice. FM demodulátor je řešen disktrétně pomocí oddělení reálné a imaginární složky signálu, jejich následné zpoždění, vynásobení a odečtení. Na výstupu je již jen dolní propust pro propuštění jen užitečného signálu a decimátor 5x.



Obr. 5.3: FM přijímač v Simulinku.

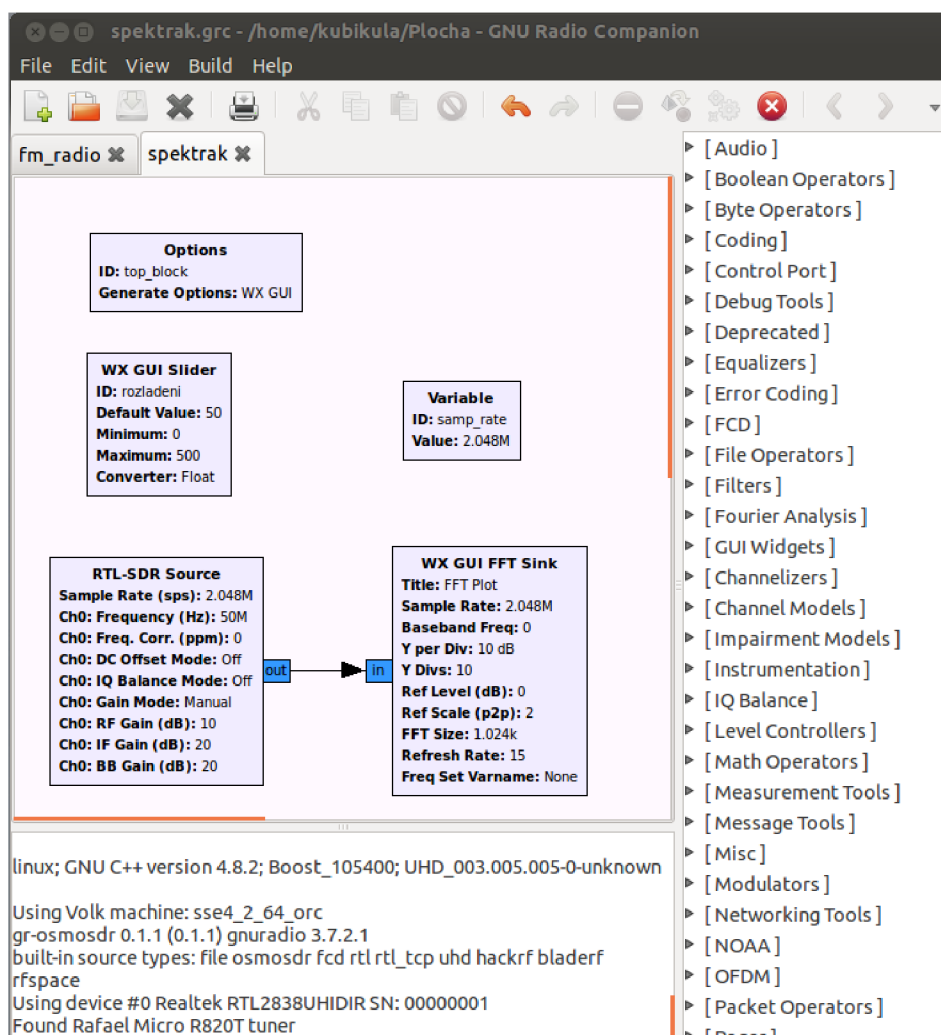
5.2 Zpracování IQ dat v GNU Radiu

GNU Radio je software s otevřenými zdrojovými kódy. Tento je založen na jazyce Python, ale výkonově kritické části jsou naprogramovány v jazyce C++ pro maximální efektivitu. Tento software je určen primárně pro platformu Linux, ale funguje i na systému Windows. Vyznačuje se názorným grafickým prostředím, jež například zpřehledňuje uživateli používání různých datových typů tím že je barevně odlišuje. Tento jednoduchý prvek je na první pohled patrný a zpřehledňuje práci v navržené blokové struktuře.

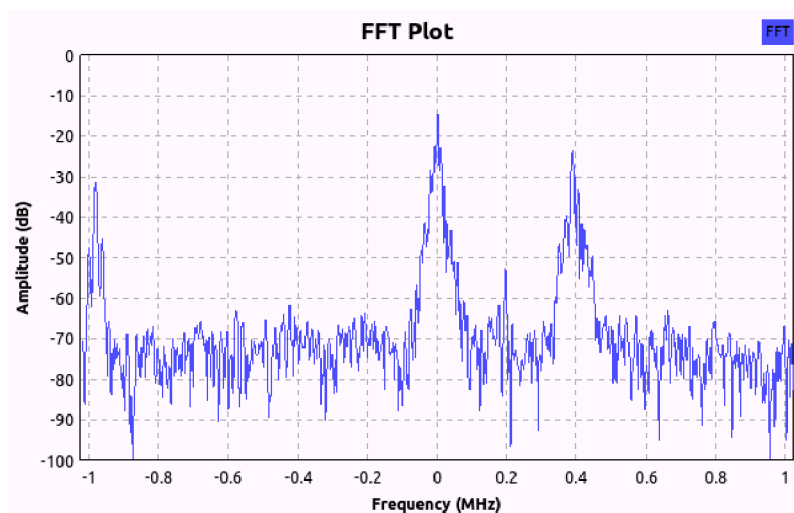
5.2.1 Spektrální analyzátor

Bloková struktura je zde velmi jednoduchá viz obr.5.4 a obsahuje zdroj signálu "RTL-SDR Source" a zobrazovací prvek "WX GUI FFT Sink". Tyto prvky si předávají data v komplexním tvaru. Je zde navíc použit i posuvný ovládací prvek "WX GUI Slider", umožňující uživateli změnit centrální frekvenci pomocí grafického posuvného prvku. Krok je nastaven na 1, maximální hodnota 500 a minimální 0. Tato hodnota je uvnitř prvního bloku vynásobena 1000000 a představuje centrální frekvenci v MHz.

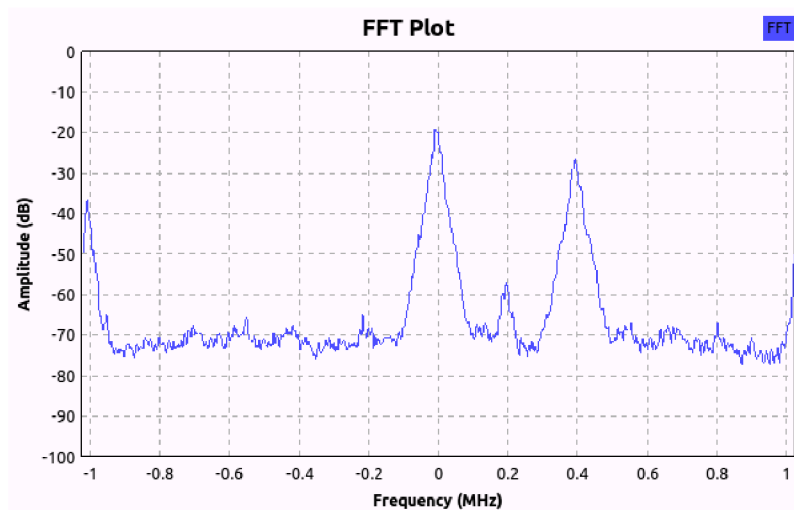
Zde je na obr. 5.5 rádiového signálu s centrální frekvencí 103 MHz a šířkou pásma 2 MHz. Na dalším obr. 5.6 je vidět průměrné spektrum se stejnými, výše uvedenými parametry.



Obr. 5.4: Spektrální analýza v GNU radiu.



Obr. 5.5: Okamžité spektrum.



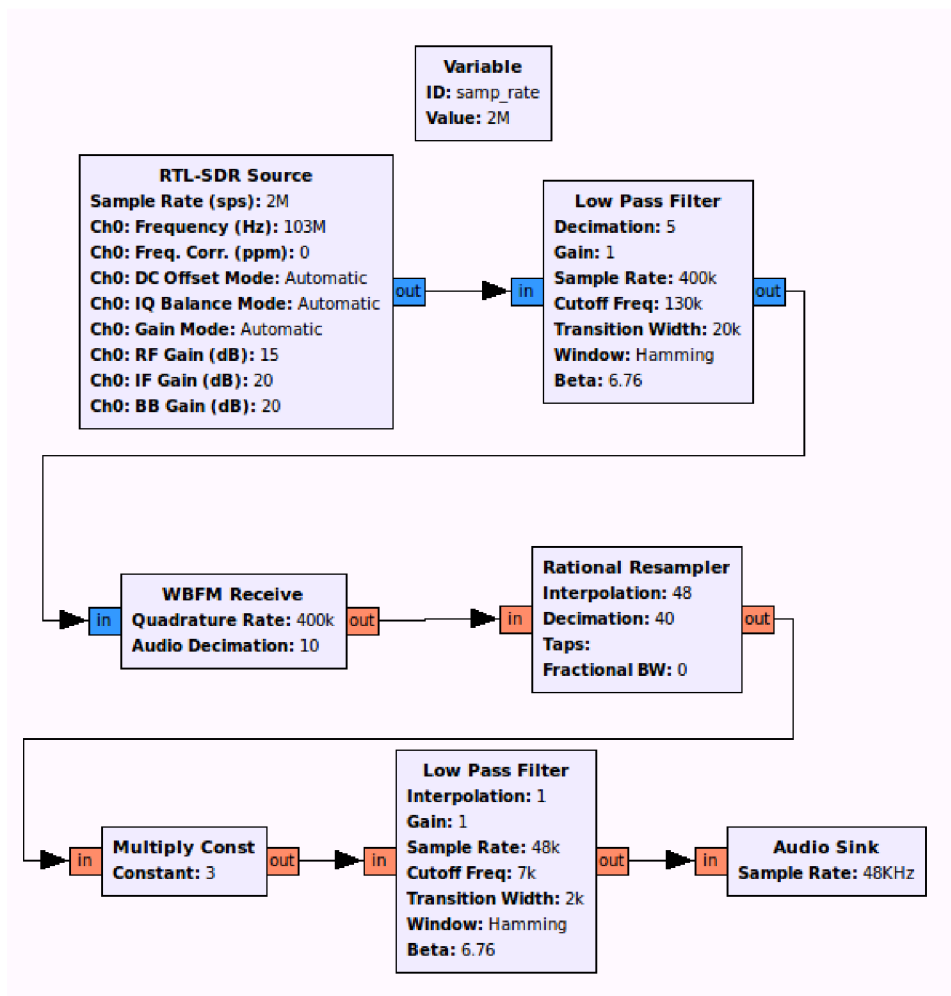
Obr. 5.6: Průměrné spektrum.

5.2.2 FM přijímač

Realizace FM přijímače je v GNU Radiu zobrazena na obr. 5.7 . Jako zdroj signálu je zde opět použit blok "RTL-SDR Source", následuje však dolní propust s 5 násobnou decimací signálu. Zde se odfiltrují ostatní rozhlasové stanice a sníží se 5x vzorkovací frekvence.

Signál pokračuje v komplexní formě na FM demodulátor počítající s velkou šířkou zdvihu FM modulace, jež se zde používá. Nastává převod na reálnou hodnotu, a znova decimace signálu 10x. Decimace se provádí pro nadbytečnost vzorků, jelikož je cílem reprodukce akustických kmitočtů.

Pro potřeby synchronizace se vzorkovací frekvencí výstupu je použit blok "Rational Resampler", dále je použita dolní propust pro odfiltrování rušivých složek v NF části.



Obr. 5.7: FM přijímač realizovaný v prostředí GNU radiu.

6 DEMONSTRACE DALŠÍCH MOŽNOSTI PŘÍJMU S TUNEREM RTL

6.1 Zpracování signálu na platformě Raspberry PI B+

Tento jednodeskový malý počítač je vhodný pro dlouhodobé nasazení a monitorování rádiového spektra, testovaná konfigurace byla spuštěna na systému Rasbian založeném na Linuxu. Sestava je chlazená pasivně a má nízkou spotřebu - do 2 W.

Popis platformy:

Základem počítače je obvod BCM2836 obsahující všechny nejdůležitější části:

- CPU jednotku ARM1176JZF-S s taktovací frekvencí 700 MHz.
- Operační paměť o velikosti 512 Mb.
- GPU Broadcom Videocore IV

Dále tento obvod obsahuje periferie : HDMI, audio výstup, SD Card slot, Micro USB konektor pro napájení, 26 GPIO vývodů, S-video, Ethernet RJ-45 a dva USB sloty.

6.1.1 Realizace FM přijímače na platformě Raspberry PI

Počítač byl ovládán přes vzdálený terminál SSH, tj. bez grafického rozhraní. Prvním krokem byla instalace knihoven RTL-SDR, dále je nutno zakázat původní knihovnu obsaženou v jádře pro DVB-T příjem. Balík RTL-SDR obsahuje i základní software pro otestování funkčnosti a FM přijímač. Po instalaci příslušných balíčků pro podporu skriptovacího jazyka Python je možno realizovat vlastní zpracování a ovládání SDR přijímače.

Raspberry je vhodný jako základ pro embedded RTL-SDR FM přijímač. Změna naladěné frekvence a hlasitosti uživatelem je možno realizovat pomocí GPIO sběrnice a i2c pro zobrazení na LCD display.

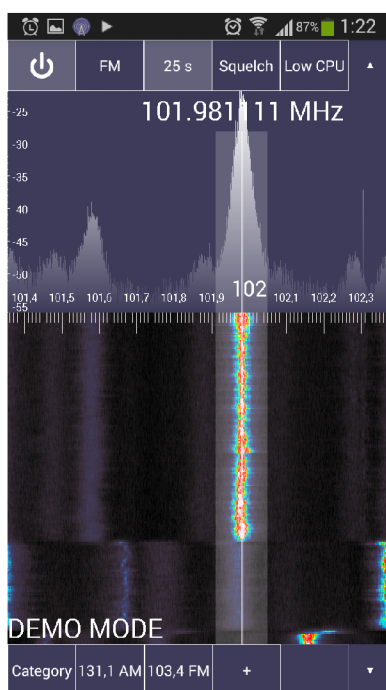
V rámci práce byla pouze ověřena funkčnost knihoven a otestován příjem FM rozhlasu funkcí integrovanou v této knihovně. Příjem byl z hlediska rychlosti zpracování signálu plynulý a kvalitní.

6.2 Zpracování signálu na platformě Android

Pro platformu Android existuje více již hotových aplikací. Zde byla vybrána aplikace [14] SDRTouch obr.6.1 řešící příjem a demodulaci FM, AM, SSB. V programu je rovněž zahrnuto grafické znázornění aktuálního přijímaného spektra a jeho historie - "vodopád". Lze tak rychle a přehledně získat informaci o obsazení přijímaného pásma 1 MHz.

Aplikace byla otestována na telefonu Samsung i9300 s tovární verzí systému Android 4.3 . Instalace i příjem proběhl bez problému. Telefon disponuje 1GB paměti RAM a čtyř jádrovým procesorem ARM Cortex-A9, příjem na VKV - FM rozhlasové stanice byl čistý a plynulý.

Je však nutné podotknout energetickou náročnost připojeného přijímače, (spotřeba 310 mA) což může při běžném způsobu OTG napájení z telefonu činit problém. Řešení je použít i při OTG vnější zdroj napájení 5V separátně jen pro přijímač.



Obr. 6.1: Přijímač na mobilním telefonu se systémem Android.

7 ZÁVĚR

Tato práce se zabývala využitím RTL-SDR přijímače za účelem analýzy rádiového spektra a RDS. I přes malý dynamický rozsah přijímače daný nízkým rozlišením AD převodníku a jeho jednoduchou konstrukcí, lze získat použitelný IQ signál pro následnou analýzu. Je však důležité využít kvalitní, nejlépe venkovní anténu. Ve zde uvedeném zapojení se nejlépe osvědčila anténa typu otevřený dipól, tento byl naladěný na 100 MHz, nicméně pracoval úspěšně širokopásmově od rozsahu cca 66 MHz až po minimálně 1090 MHz. Analýza systému

RDS distribuovanému spolu s rozhlasovými stanicemi nebyla zcela úspěšná pro nedokonalou synchronizaci DSB demodulátoru. Synchronizační signál byl vybrán z třetí harmonické složky pilotního signálu. Při bližším zkoumání pomocí funkce Scope nebyly shledány zřejmé defekty na tvaru, amplitudě či fázi. Širokopásmová spektrální analýza byla provedena obdobně skriptem napsaným

v Pythonu i Matlabu. Praktické použitelnosti je však možné využít jen pomocí Pythonu, ať již z důvodu hardwarové nenáročnosti či a rychlosti vykonání. Pro srovnání, spektrální analýza zabrala Matlabu násobně více výpočetního času než trval výpočet skriptu v Pythonu. Hlavní problém v Matlabu byl s časem, nutným pro získání dat z přijímače. Ten byl neúměrně dlouhý a tvořila většinu výpočetního času, tudíž je na vině špatná optimalizace interního ovladače pro RTL SDR přijímač.

Při srovnání grafického softwarového prostředí GNU Radio Companion a Matlab Simulink je nutné konstatovat že se jedná o diametrálně odlišné prostředí. Výhodou GNU Radia je nicméně pouze jeho svobodná licence a nenáročnost na výpočetní prostředky. Má méně vnitřních funkcí (kupříkladu absence grafického znázornění frekvenční charakteristiky filtru), menší výběr funkčních bloků a hlavně vzájemná nekompatibilita jednotlivých pravidelně vydávaných verzí a množství chyb které se náhodně projevují při překladu.

Oproti tomu Matlab nabízí oficiální podporu a velmi široké možnosti, umí nejenom zpracovávat digitální signál, je stabilní a spolehlivý. Hlavní nedostatek lze spatřovat ve vyšších nárocích na výpočetní výkon při kompilování navržené struktury. Tato operace zabere běžně desítky sekund.

LITERATURA

- [1] JAN, Jiří. *Číslicová filtrace, analýza a restaurace signálů*. 2. upr. a rozš. vyd. Brno: VUTIUM, 2002, 427 s. ISBN 80-214-2911-9.
- [2] SMÉKAL, Zdeněk. *Systémy a signály: 1D a 2D diskrétní a číslicové zpracování*. 1. vyd. Praha: nakladatelství Sdělovací technika, 2013, 254 s. ISBN 978-80-86645-23-0.
- [3] SRUTHI, M. B., M. ABIRAMI, A. MANIKKOTH, R. GANDHIRAJ a K. P. SOMAN. *Low cost digital transceiver design for Software Defined Radio using RTL-SDR. 2013 International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (iMac4s)* [online]. IEEE 2013 s. 852-855 [cit. 15. 12. 2014]. Dostupné z URL: <<http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=6526525>>.
- [4] Rda Microelectronics *RDA1846* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <http://www.rdamicro.com/Products/Detail_122.aspx>.
- [5] Rafael Microelectronics *R820T* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://www.rafaelmicro.com/downloads/R820T.pdf>>.
- [6] NOOELEC *E4000* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <www.nooelec.com/files/e4000datasheet.pdf>.
- [7] OsmocomSDR *rtl-sdr* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://sdr.osmocom.org/trac/wiki/rtl-sdr>>.
- [8] RTL-SDR.COM *rtl-sdr* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://www.rtl-sdr.com>>.
- [9] Superkruh *rtl-sdr* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://www.superkuh.com/rtlsdr.html>>.
- [10] BARTOŠ, Vladimír *sdr* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://www.sdr.ipip.cz>>.
- [11] Matlab *RTL-SDR Support from Communications System Toolbox* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://www.mathworks.com/hardware-support/rtl-sdr.html>>.
- [12] Sodankylä Geophysical Observatory *Kilpisjärvi Atmospheric Imaging Receiver Array* [online]. [cit. 15. 12. 2014]. . Dostupné z URL: <<http://kaira.sgo.fi/2013/09/passive-radar-with-16-dual-coherent.html>>.

- [13] Python Package Index *Python interface supported RTL2832U chipset* [online]. [cit. 20. 4. 2015]. . Dostupné z URL: <<https://pypi.python.org/pypi/pyrtlsdr>>.
- [14] MARTIN MARINOV *Receiving FM,AM,SSB with Android phone* [online]. [cit. 25. 3. 2015]. . Dostupné z URL: <<http://www.sdrtouch.com>>.
- [15] PROKEŠ, Aleš *Rádiové přijímače a vysílače: přednášky*. Vyd. 1. Brno: Vysoké učení technické, 2005, 178 s. ISBN 80-214-2263-7.
- [16] PETR, Ondřej *Vysoké učení technické Brno, Fakulta elektrotechniky a komunikačních technologií: diplomová práce* [online]. [cit. 20. 7. 2015]. . Dostupné z URL: <https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=85540>.
- [17] VEDRA, Lukáš *Vysoké učení technické Brno, Fakulta elektrotechniky a komunikačních technologií: diplomová práce* [online]. [cit. 20. 7. 2015]. . Dostupné z URL: <https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=83499>.

SEZNAM SYMBOLŮ, VELIČIN A ZKRATEK

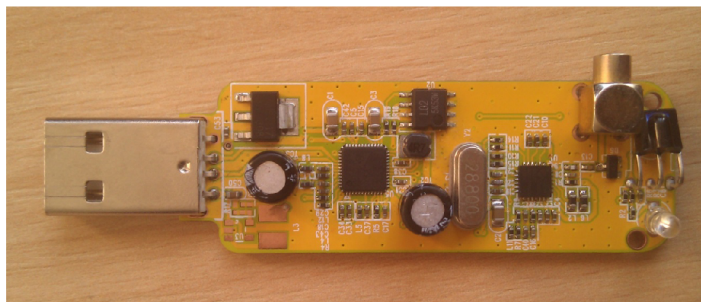
ADC	analogově digitální převodník – Analog-to-digital converter
AGC	automatické řízení zisku – Automatic gain control
ARI	rádiový informační systém pro motoristy – Autofahrer Rundfunk Informationssystem
DAB	pozemní digitální rozhlasové vysílání – Digital Audio Broadcasting
DSP	číslicové zpracování signálů – Digital Signal Processing
DTMF	dvoutónová selektivní volba – Dual-tone multi-frequency
DVB-T	pozemní digitální televizní vysílání – Digital Video Broadcasting - Terrestrial
FFT	rychlá Fourierova transformace – Fast Fouxier Transformation
FM	frekvenční modulace
f_{vz}	vzorkovací kmitočet
OTG	původně USB slave zařízení s funkcí USB master – USB On-The-Go
RBW	frekvenční rozlišení – Resolution Bandwidth
RDS	rádiový datový systém – Radio Data System
SDR	softwarově definované rádio – Software Definited Radio
SNR	poměr užitečného signálu k šumu – Signal to noise ratio
USB	univerzální sériová sběrnice – Universal serial bus
USRP	univerzální platforma pro softwarové rádio – Universal software radio peripheral.
VOX	hlasem ovládaný spínač klíčování – Voice-operated switch

SEZNAM PŘÍLOH

A Fotografie použitého DVB-T tuneru	39
A.1 Osazaná deska plošného spoje tuneru	39
A.2 Přijímač v původním plastovém krytu	39
B Zdrojový kód v jazyce Python pro přehledový spektrální analyzér	40

A FOTOGRAFIE POUŽITÉHO DVB-T TUNERU

A.1 Osazená deska plošného spoje tuneru



Obr. A.1: Přijímač bez krytu

A.2 Přijímač v původním plastovém krytu



Obr. A.2: Přijímač s krytem

B ZDROJOVÝ KÓD V JAZYCE PYTHON PRO PŘEHLEDOVÝ SPEKTRÁLNÍ ANALY- ZÉR

```
1 from rtlsdr import RtlSdr
import numpy as np
import time
import scipy
import math
6 import matplotlib.pyplot as plt

# Cast kodu pro konfiguraci spektralniho analyzeru
f_start = input("Vloz startovni frekvenci [ MHz ] :")
print "Fstart =", f_start
11 f_stop = input("Vloz koncovou frekvenci [ MHz ] :")
print "Fstop =", f_stop
sirka_volba = raw_input("Vloz volbu pro rozliseni, RBW (
    a = 244 Hz // b = 61 Hz // c = 15 Hz // d = 1 Hz):")
print "volba =", sirka_volba
log_volba = raw_input("Logaritmicke stupnice : a = ano
    // n = ne :")
16 print "volba =", log_volba
pozadi_volba = raw_input("Odecist sumove pozadi = a //
    Hlidat spektrum = h // Nedelat nic navic = n :")
print "volba =", pozadi_volba

21 if sirka_volba is 'a' :
    pocet_vzorku = 8 * 512
if sirka_volba is 'b' :
    pocet_vzorku = 32 * 512
if sirka_volba == 'c' :
26 pocet_vzorku = 128 * 512
if sirka_volba == 'd' :
    pocet_vzorku = 1024 * 1024
if pozadi_volba == 'h' :
    detekce_uroven = input("Vloz uroven nad kterou
        bude indikovat detekovan signal :")
31 print "Uroven detekce =", detekce_uroven
```

```

sdr = RtlSdr()
pocet_oken = f_stop-f_start
f_start = f_start * 1e6
36 f_stop = f_stop * 1e6
sdr.sample_rate = 1.2e6 # Hz
sdr.freq_correction = +30 # korekce odchylky hodin, v
    PPM
sdr.gain = 12

41 sdr.center_freq = f_start + 0.5e6 # Hz
x = np.linspace(f_start, f_stop, pocet_oken * (int(round(
    pocet_vzorku*(1/1.2)))))) # definice velikosti pole pro
    vyslednou FFT - jen pro indexy
max_hodnota = 0
data = sdr.read_samples(pocet_vzorku) # testovaci
    aktivace prijimace, prvni vzorky nejsou kvalitni
provede_mereni = 1
46 pozadi_temp = 0
if pozadi_volba == 'a' :
    pozadi_temp = 1
    raw_input("Probehne mereni pozadi, pro
        pokracovani stiskni ENTER")

51 while provede_mereni==1 : # test zda-li se provede
    dvakrat pro kontrolu pozadi

        fft = np.linspace(0,0, pocet_oken * (int(round(
            pocet_vzorku*(1/1.2)))))) # definice velikosti
            pole pro vyslednou FFT - zde se zapisuje
        for aktualni_okno in range(1, pocet_oken+1):
            if pozadi_volba != 'h' : # Zobrazuje
                prubeh mereni, krome hlidani spektra
                    print('Merim')
56 time.sleep(0.08) # bezpecnostni prodleva
                pro preladeni a ustaleni
            data = sdr.read_samples(pocet_vzorku)
            sdr.center_freq = sdr.center_freq + 1e6;
            data = abs(scipy.fft(data)) # provede se
                FFT

61 for index in range(-1,1): # vymaze se

```

```

        pripadna velika stejnosmerna hodnota
            data[index] = 0

data = np.roll(data, pocet_vzorku/2) #
    vycentruje se

66
for index in range(0, int(round(
    pocet_vzorku*(1/1.2)))): # sklada
    obraz do jednoho velkeho po 1 MHz
    oknech, ovsem navzorkovanych je 1.2
    MHz, okraje 100kHz se zahodi
        fft[index+(int(round(pocet_vzorku
            *(1/1.2)))*(aktualni_okno - 1)
            )] = data[index+int(round(
                pocet_vzorku*(0.1/1.2)))]
        if log_volba is 'a' :
            if fft[index] < 1 :
71
                fft[index]= 1
            fft[index]=20*math.log10(fft[
                index])
        provede_mereni = 0
***** konec jednoho mereni > zacatek mozneho
cyklickeho mereni *****
        if pozadi_temp == 1 : # pozadi_temp je
            "1" pri prvnm kole mereni pozadi, pri mereni
            uzitecneho signalu je "0"
76
            pozadi_data = fft # zaloha pozadi
            provede_mereni = 1
            pozadi_temp = 0
            raw_input("Probehne mereni s pritomnosti
                uzitecneho signalu, pro pokracovani
                stiskni ENTER")

81
        if pozadi_volba == 'h' : # aktivace funkce
            hlidani
            provede_mereni=1 # bezi stale
            dokola
            max_hodnota = 0
            index = 0
            for index in range(0, pocet_oken *(int(
                round(pocet_vzorku*(1/1.2))))): #

```

```

86         proscanuje indexy vysledneho FFT okna
            if fft[index] > max_hodnota:
                max_hodnota = fft[index]
                index_max_hodnoty = index

            if 1 : # vyhodnoceni na zaver - zda-
                li doslo k detekci nosne
91                 print('Detekovana signal o
                    frekvenci[ MHz ]')
                print(x[index_max_hodnoty])
                print('A amplitude')
                print(max_hodnota)
                if log_volba is 'a' :
96                     print(' [dB]')
                else:
                    print(' [-]')
                print('***** opet merim
                    *****')

101

#*****      konec cyklickeho mereni > zacatek
            vyhodnoceni *****

            for index in range(0, pocet_oken *(int(round(pocet_vzorku
                *(1/1.2))))): # proscanuje indexy vysledneho FFT okna
106                 if fft[index] > max_hodnota:
                    max_hodnota = fft[index]
                    index_max_hodnoty = index
                if pozadi_volba == 'a' :
                    fft[index] = fft[index] - pozadi_data[
                        index]

111

            plt.close('all')
            f, ax = plt.subplots()
            ax.plot(x,fft )
            ax.set_title('FFT')
116 ax.set_ylabel('Amplituda [-]')
            if log_volba is 'a' :
                ax.set_ylabel('Amplituda [dB]')
            ax.set_xlabel('Frekvence [ MHz ]')

```

```

121 print('*****')
    print('Start [ MHz ]:')
    print(f_start/1e6)
    print('-----')
    print('Stop  [ MHz ]:')
126 print(f_stop/1e6)
    print('-----')
    print('Sirka pasma [ MHz ]:')
    print((f_stop - f_start)/1e6)
    print('-----')
131 print('Pocet oken [ - ]:')
    print(pocet_oken)
    print('-----')
    print('Rozliseni, RBW :')
    print(1e6/pocet_vzorku)
136 print('-----')
    print('Dominantni frekvence:')
    print(x[index_max_hodnoty])
    print('*****')
    plt.show()

```