

Univerza v Ljubljani  
Fakulteta *za elektrotehniko*



# SPREJEM FM RADIA IN RDS Z UPORABO RTL-SDR

---

Seminarska naloga pri predmetu Digitalne komunikacije

Nosilca predmeta: prof. dr. Sašo Tomažič

Mentor: višji predavatelj dr. Anton Umek

1. stopnja, 3. Letnik – Elektrotehnika-IKT (6. Semester)

Avtorji: Blaž Bertalanič, vpisna številka 64110046

Leon Lončarič, vpisna številka 64140479

Simon Šoln, vpisna številka 64090289

V Ljubljani, maj 2017

## Kazalo vsebine

|   |    |
|---|----|
| Kazalo vsebine .....  | 2  |
| 1. PROBLEMATIKA SEMINARSKE NALOGE .....                                   | 3  |
| 1.1. O RTL SDR .....  | 3  |
| 1.2. IQ ZAPIS SIGNALA .....   | 4  |
| 1.3. FM DEMODULACIJA .....  | 9  |
| 1.4. RDS .....  | 9  |
| 2. DEMODULACIJA FM S PROGRAMSKIM PAKETOM MATLAB .....                     | 12 |
| 3. DEMODULACIJA FM IN RDS S PROGRAMSKIM PAKETOM GNU Radio Companion ..... | 14 |
| 4. ZAKLJUČEK .....  | 19 |
| 5. VIRI .....   | 20 |

## 1. PROBLEMATIKA SEMINARSKE NALOGE

Pri seminarski nalogi v okviru predmeta Digitalne komunikacije smo si izbrali tematiko demodulacije FM in RDS signala z uporabo dostopnega nizkocenovnega digitalnega radijskega sprejemnika RTL-SDR. V okviru seminarske naloge se bomo spoznali z različnimi stopnjami v procesu demodulacije, uporabe digitalnega radija ter uporabe različnih dostopnih orodij, namenjenih obdelavi digitalnih signalov.

### 1.1. O RTL SDR

Ime RTL-RDS izahaja iz Realtekovega čipa RTL2382U, ki je bil prvotno osnovan za uporabo v televizijskih in radijskih sprejemnikih na osnovi SDR, vendar v našem experimentu nismo uporabili tega. Kratica SDR pomeni "software defined radio", tj. radijski sistem katerega so vse komponentne vključno z filtri, ojačevalniki, seštevalniki, modulatorji, demodulatorji implementirani programsko, v programu so to običajno predstavljeni kot bloki, kateri ima vsak določeno nalogo. Danes se SDR največ uporablja v mobilnih omrežjih vojaških komunikacijah predvsem zaradi tega ker omogoča menjavo radijskega protokola kar med delovanjem, saj so ti že implementirani programsko.

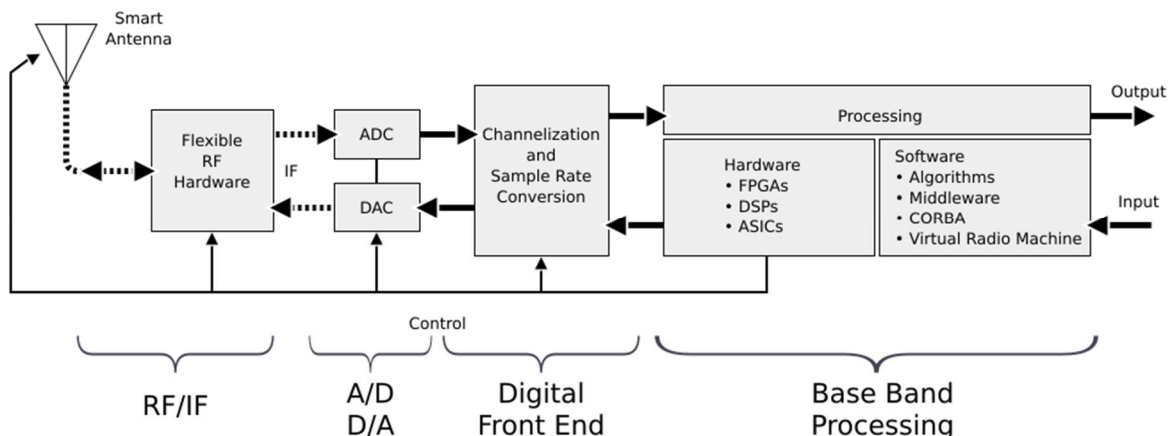


Figure 1 Splošna blokovna shema SDR

Pri našem eksperimentu smo uporabili RTL-SDR USB sprejemnik s čipoma RTL2838 in tuner čipom R820T, s nadgrajenim kristalnim oscilatorjem (TCXO – temperature compensated crystal oscillator). V osnovi je sestavljen iz RF vhoda, kateri nam določa frekvenčno območje delovanja, ta pa je priključen na RF sprejemnik, katerega glavna naloga je prestavljanje uglešene frekvence na osnovni pas. RF sprejemnik je naprej povezan z analognim digitalnim

pretvornikom, ta pa naprej pošilja vzorce preko USB vodila v računalnik za nadaljnjo obdelavo.

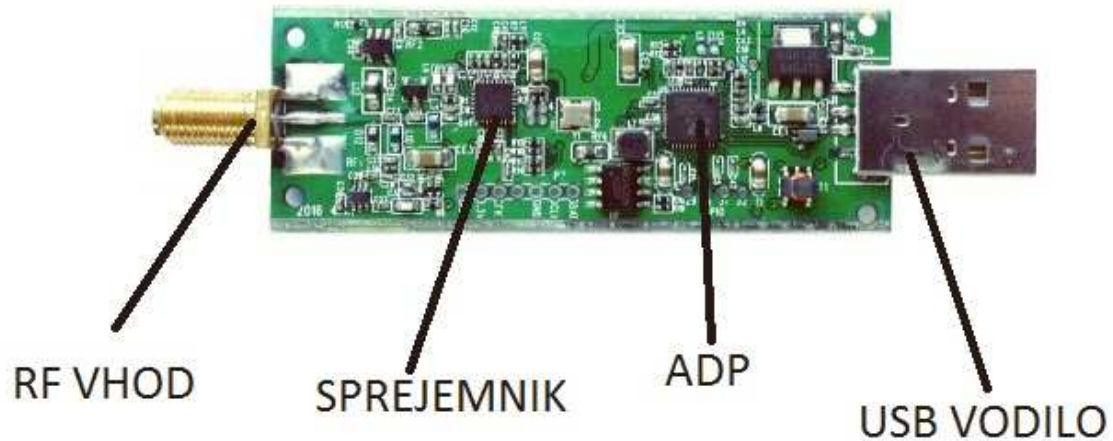


Figure 2 Zgradba RTL-SDR sprejemnika

### 1.2. IQ ZAPIS SIGNALA

Pri normalnem zapisu signala kot niz vzorcev ob vzorčnih trenutkih imamo običajno problem, da ne moremo določiti predznak frekvence signala saj npr. signal  $\cos(x)$  in  $\cos(-x)$  nam da popolnoma enako krivuljo. Ta problem pride predvsem do izraza pri ko mešamo dva signala. Drugič pa je težko določiti moč tako zapisanega signala, saj

ne vemo ali je signal periodičen in ali smo ga dovolj dobro vzorčili točno ob njegovem maksimumu.

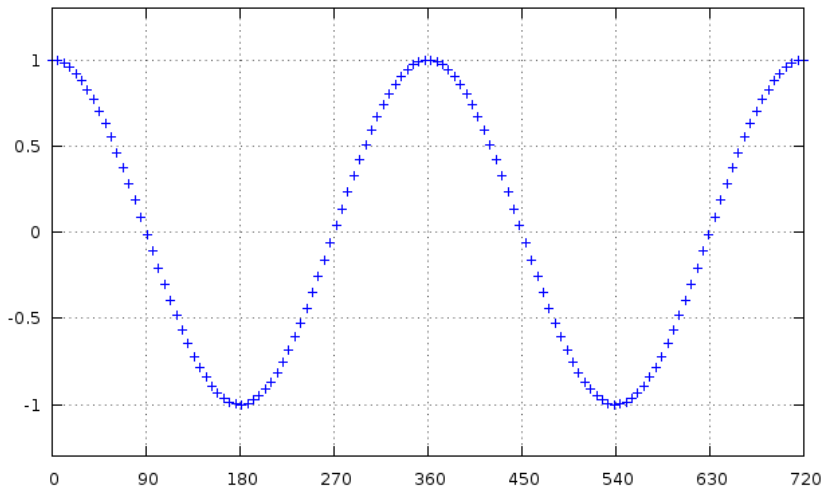


Figure 3 Vzorčen signal predstavljen kot niz vzorcev

Te pomanjkljivosti odpravi I/Q zapis signala. Namesto da signal gledamo kot 2D krivuljo, ga gledamo kot vijačnico v prostoru. Za pretvorbo signala v I/Q format potrebujemo DFT

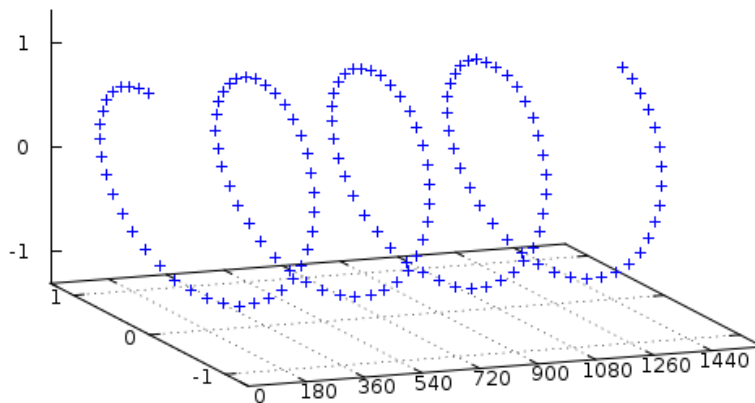


Figure 4 I/Q predstavitev signala v prostoru v obliki vijačnice in vidimo da im signal pozitivno frekvenco saj se vijačnica navija v nasprotni smer urinega kazalca. Radij vijačnice pa predstavlja amplitudo signala.

Če ta signal pogledamo iz strani predstavlja stranski pogled I del našega signala tj. Realni del našega signala. Zgornji pogled pa Q del našega signala, ki je v bistvu fazno zamaknjen za 90 stopinj.

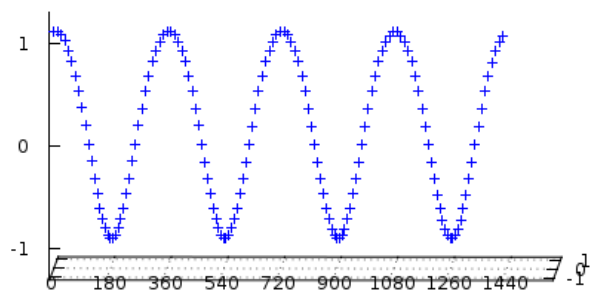


Figure 5 Zgornji pogled signala, "Q" del

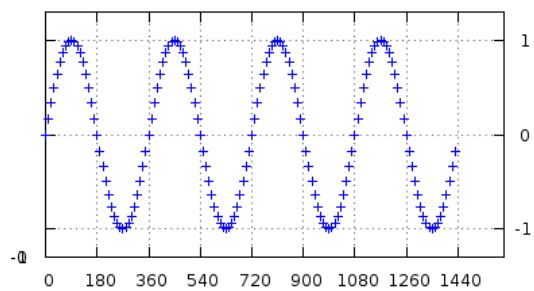


Figure 6 Stanski pogled signala, "I" del

I/Q vzorci predstavljajo koordinate signala v ravnini. Vsak signal, tudi ne harmonični lahko predstavimo kot amplituda signala krat kosinus faze  $A \cdot \cos(\varphi_i)$ , kar predstavlja tudi I del našega signala, z sinusom

pa dobimo Q del signala. Vzorec lahko predstavimo kot vektor v imaginarni ravnini. Lahko pa predstavimo v polarni obliki in z eulerjovo formo, v katere se pa ne bomo poglobljali.

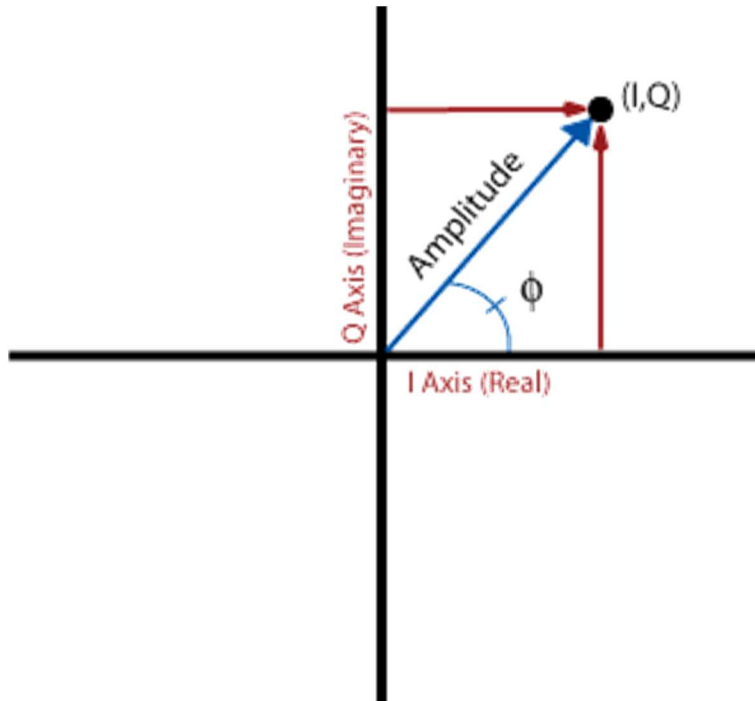


Figure 7 Pravokotna reprezentacija vzorca I/Q signala kot vektorja, ki se vrti okoli izhodišča z določeno frekvenco.

#### MEŠANJE IN MNOŽENJE IQ SIGNALOV

Kot smo že prej omenili je prednost zapisa I/Q signala predvsem v tem da dobimo pri množenju signalov ta pravi signal, saj nam ta zapis podaja predznak frekvence. Spodaj vidimo signala v frekvenčni domeni po množenju. (DODAJ ENAČBE)

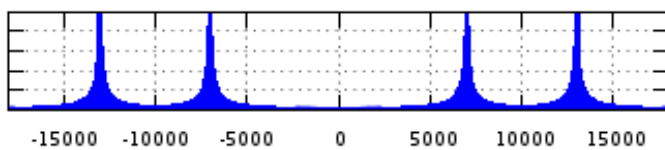


Figure 8 Rezultat množenja signalov v navadnem zapisu.

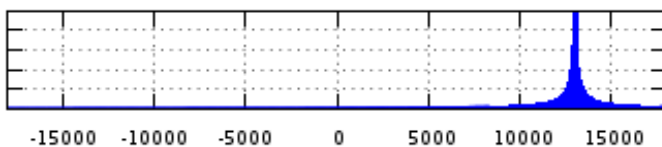


Figure 9 Rezultat množenja signalov z zapisom v I/Q obliki.

Predpostavimo da imamo dva signala z frekvencama  $+f$  in  $-f$ . Če zmešamo signala dobimo z I/Q eno rešitev to je  $0f$  torej eno smerni signal. Z navadnim zapisom pa  $0f$  in  $\pm 2f$ . Spodaj na sliki vidimo sliko v časovni domeni kaj se dogaja pri mešanju signalov.

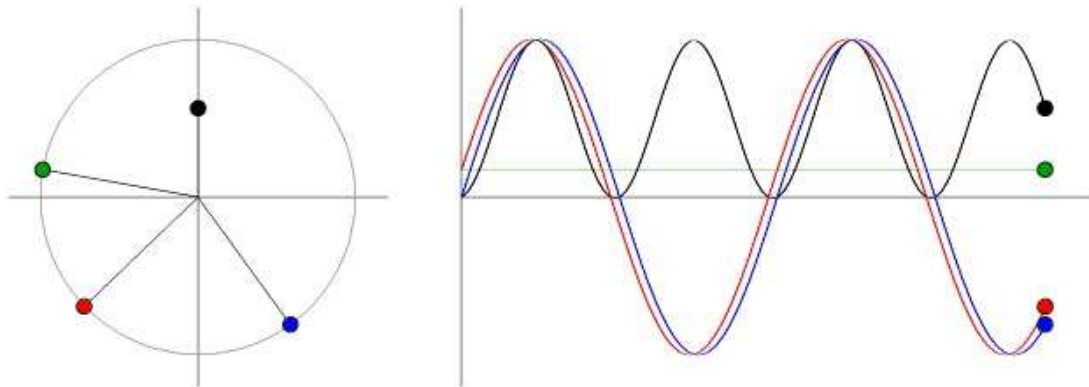


Figure 10 Rezultat mešanja signalov  $+f$ (moder) in  $-f$ (rdeč). Signala sta malo fazno zamaknjena zaradi lepše vidljivosti. Pravilni rezultat mešanja je zeleni signal. Črni signla pa predstavlja signal z dvojno frekvenco.

Z pomočjo I/Q zapisa signala lahko zapišemo dvodimenzionalni I/Q signal v enodimenzionalni RF signal, brez izgub. Nosilnemu signalu lahko spreminjamo dva parametra. Modulirani RF signal izgleda takole:  $\text{Modulated Carrier RF} = I \cos(2\pi ft) + Q \sin(2\pi ft)$ .

Za nas bo predvsem pomembne frekvenčna modulacija. Z FM modulacijo spreminjamo rekvenco nosilnega signala v ritmu modulatorskega signala, kar omogoča večji dinamični razpon informacijskega signala in je posledično bolj odporen na motnje.



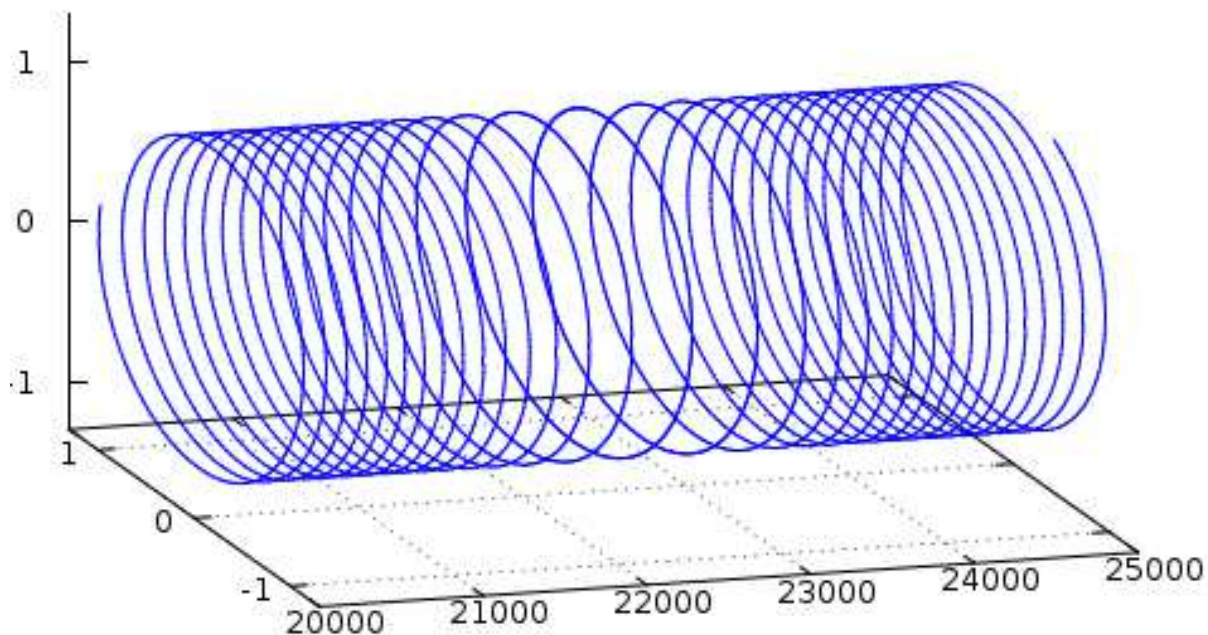


Figure 11 FM modulacija v I/Q

### 1.3. FM DEMODULACIJA

Pri frekvenčni demodulaciji v splošnem s pomočjo modulacijskega signala izločimo prvotno obliko signala iz moduliranega signala. Demodulator je lahko elektronsko vezje, računalniški program ali SDR, slednjega smo uporabili v našem primeru.

### 1.4. RDS

RDS oz. Radio Data System je protokol za prenos manjših količin digitalnih podatkov znotraj okvirov FM radija. Gre pravzaprav za meta-podatke (podatki o podatkih), katerih namen je izboljšanje interakcije med izvorom in ponorom. Izvor lahko smatramo kot radijsko postajo, ponor pa avtoradio, kjer RDS zares dobi kontekst, saj se velik del RDS podatkov nanaša na spreminjajočo se lokacijo uporabnika.

Frekvenčni pas, v katerem se nahaja naš izvorni signal imenujemo osnovni pas (ang. Baseband), katerega moduliramo kar v osnovnem pasu, nato pa ga z mešanjem prestavimo v višjo frekvenčno lego. Dandanes se v okviru FM radija v osnovnem pasu prenaša superponiran signal levega in desnega zvočnega signala (L+R). Zmoduliran signal (L-R) pa se nahaja okrog frekvence prvega podnosilca. Za fazno sinhronizacijo zgornjih dveh signalov se

uporablja pilotni ton z frekvenco 19kHz. Tretje višjeharmoniska komponenta pilotnega tona pa predstavlja nosilno frekvenco našega RDS sistema.

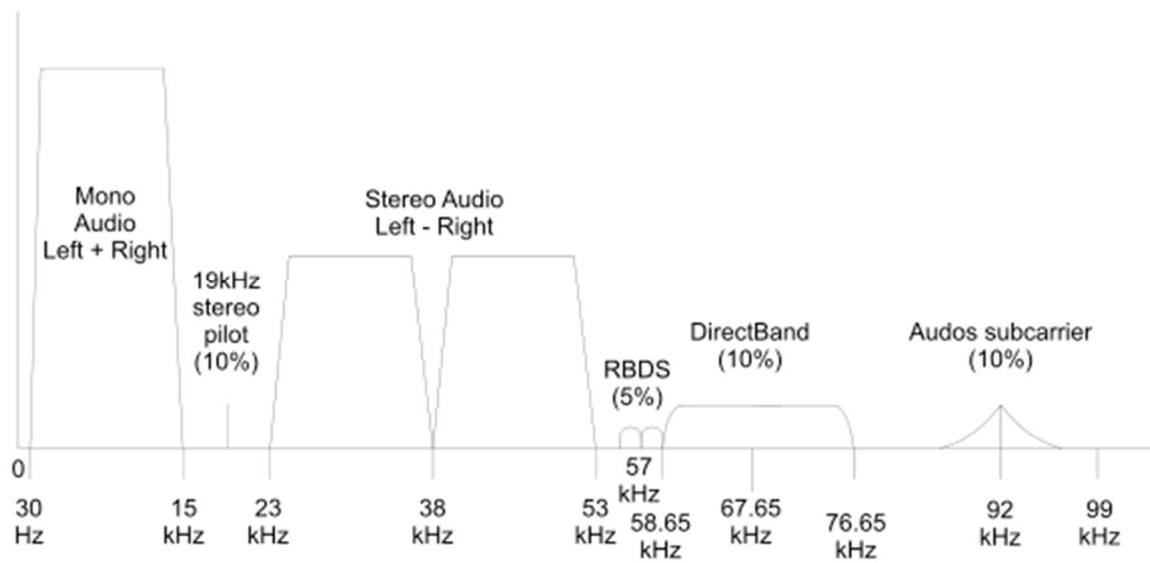


Figure 12 Predstavitev signalov v frekvenčni domeni

RDS tehnologijo je v 70. letih razvil danes zelo priznan proizvajalec HiFi opreme Blaupunkt. RDS je sčasoma nadomestil sistem ARI, ki je bil namenjen občasnemu obveščanju o

prometnem stanju. Prednost RDS je predvsem v tem, da je omogočil sodostop, torej da lahko več radijskih postajo uporablja isti frekvenčni pas.

Pri RDS gre za digitalni prenos podatkov, katerega vrsta kodiranja je odvisna od vsebine. Bitna hitrost prenosa RDS podatkov je 1187.5 bitov na sekundo. Takšen binarni signal je potem zmoduliran po principu dvobočne modulacije z zadušenim nosilcem (DSB-SC).

RDS podatki se čez medij pošiljajo kot paketi. Paketov je vsesкупaj 32. Vsak je razdeljen na štiri bloke, katerih dolžina je 26 bitov, o tega je 16 bitov uporabniške informacije, 10 bitov pa nadzorne informacije (checksum, error detection..)

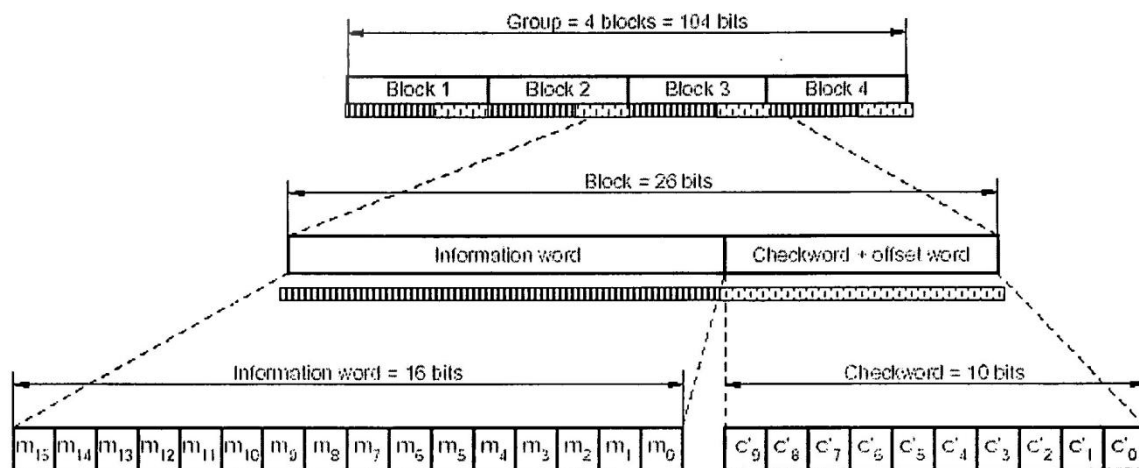


Figure 13 Struktura RDS paketa.

Vsebina oz. Oznake z informacijami, ki so ponavadi vključene v RDS podatkih pa so lahko:

- AF (Alternative Frequencies)
- CT (Clock Time)
- DI (Decoder Information)
- EON (Echanced Other Networks)
- M/S (Music Speech Code)
- PI (Program Indentification)
- PS (Program Service)
- REG (Regional)
- RT (Radio Text)
- TP (Traffic Programme)

- TA (Traffic Annoucement)

## 2. DEMODULACIJA FM S PROGRAMSKIM PAKETOM MATLAB

Ker smo imeli zgolj en RTL-SDR sprejemnik smo zanj preko raspbery Pi-aj postavili strežnik in tako preko SSHja dostopali in zajemali podatke preko ključka. Podatke smo zajemali s pomočjo programa RTL-SDR v linux konzoli. Zaradi velike količine podatkov, zaradi česar bi lahko imeli težave pri obdelavi le teh v Matlabu, smo radijski signal zajemali le 30 sekund ter signal zajemali z vzorčevalno frekvenco 2.5MHz.

Dostop in zajem podatkov s RTL-SDR sprejemnika:

Za zajem podatkov smo uporabili program `rtl_sdr`, ki je del programskega paketa GNU Radio. Podatke za demodulacijo smo zajeli z ukazom:

```
rtl_sdr -f 105700000 -g 40 -s 2500000 -n 75000000 FMcap1.dat
```

Kaj pomenijo argumenti ob programu?

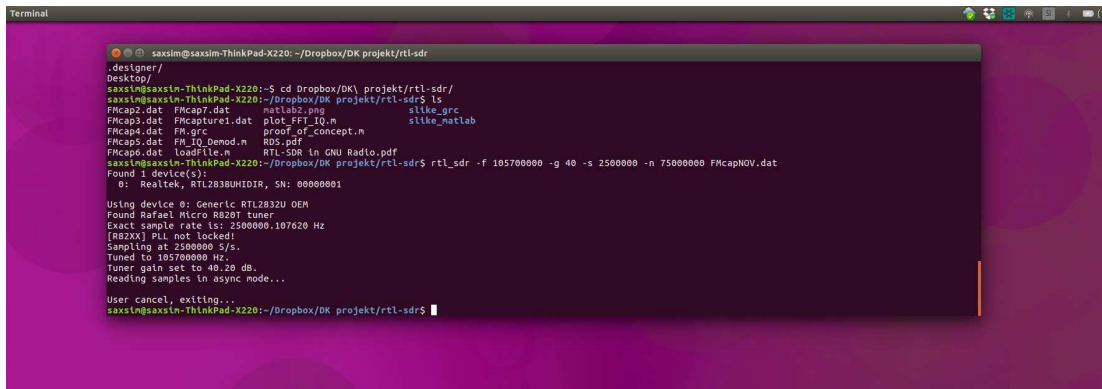
-f – centralna frekvenca, na kateri RTL SDR zajema

-g – gain zajema

-s – sample rate – vzorčevalna frekvenca (v našem primeru 2.5MHz)

-n – število vzorcev (v našem primeru 75000000, kar pomeni 30 sekund zajema)

Zadnji argument je ime datoteke, v katero želimo shraniti naš zajem IQ podatkov.

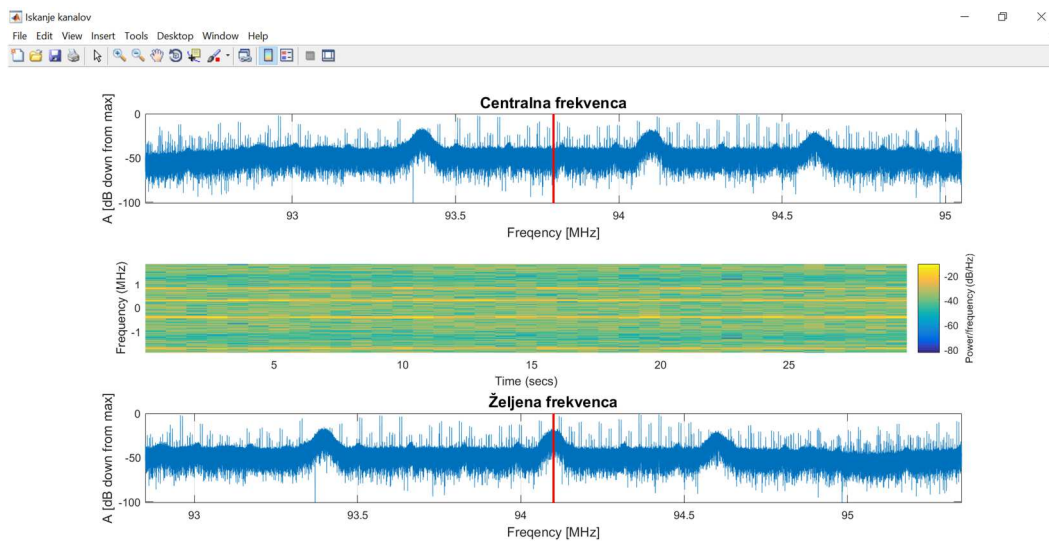


```
Terminal
saxsim@saxsim-ThinkPad-X220: ~/Dropbox/DK projekt/rtl-sdr
.designer/
Desktop/
saxsim@saxsim-ThinkPad-X220:~$ cd Dropbox/DK projekt/rtl-sdr/
saxsim@saxsim-ThinkPad-X220:~/Dropbox/DK projekt/rtl-sdr$ ls
FMcap2.dat  FMcap7.dat      matlab2.png      slike_grc
FMcap3.dat  FMcapture1.dat  plot_FFT_IQ.m     slike_matlab
FMcap4.dat  FM_grc          proof_of_Concept.m
FMcap5.dat  FM_IQ_Demod.m  RDS.pdf
FMcap6.dat  loadFile.m     RTL-SDR in GNU Radio.pdf
saxsim@saxsim-ThinkPad-X220:~/Dropbox/DK projekt/rtl-sdr$ rtl_sdr -f 105700000 -g 40 -s 2500000 -n 75000000 FMcapNOV.dat
Found 1 device(s):
0: Realtek, RTL2838UHIDIR, SN: 00000001

Using device 0: Generic RTL2832U OEM
Found Rafael Micro R820T tuner
Exact sample rate is: 2500000.107620 Hz
[R82XX] PLL not locked!
Sampling at 2500000 S/s.
Tuned to 105700000 Hz.
Tuner gain set to 40.20 dB.
Reading samples in async mode...

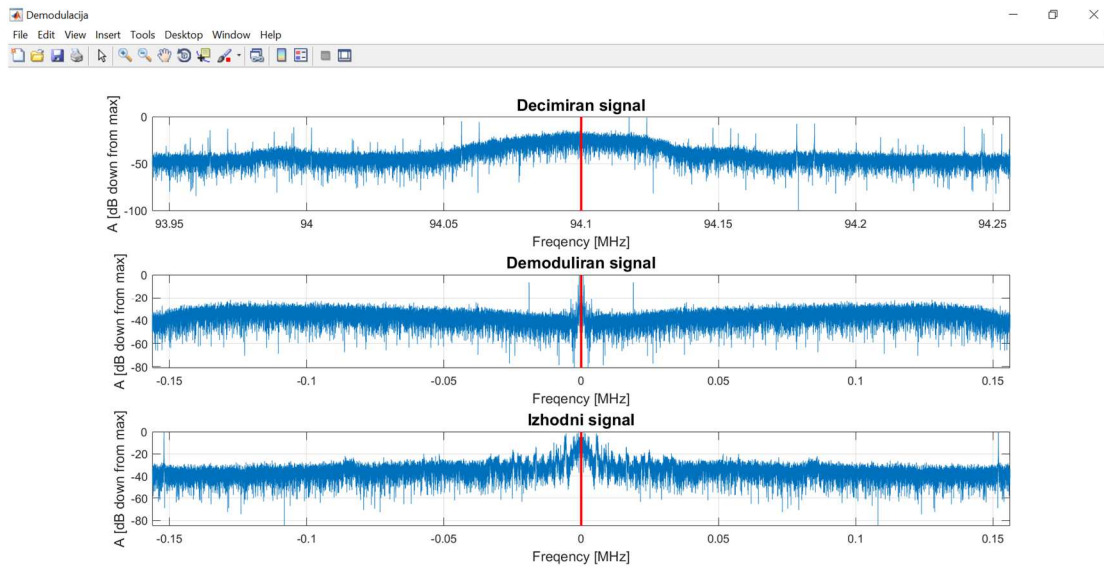
User cancel, exiting...
saxsim@saxsim-ThinkPad-X220:~/Dropbox/DK projekt/rtl-sdr$
```

Našo podatkovno datoteko smo nato vnesli v Matlab in je bila pripravljena na nadaljno obdelavo. Nato smo določili našo frekvenco vzorčenja vnesenega signala kot tudi frekvenco, ki jo želimo na koncu slišati. Določili smo še sample rate, oziroma čas zajema. Za lažjo predstavo smo si izrisali spekter našega zajetega signala, spektrogram, nato pa še signal zamaknili na željeno poslušano frekvenco. Vsak hribček na spodnji sliki predstavlja eno radijsko postajo. Rumene črte na spektrogramu predstavljajo naše radijske postaje, vendar smo za izris uporabili že vgrajeno Matlab funkcijo, zato oznaka y-osi ni najboljša. Pri frekvenci 0MHz je na spktrogramu predstavljena naša centralna frekvenca, nato pa gor in dol od nje vidmo naše postaje.



Zamaknjen signal smo nato obdelali še naprej in sicer smo na njem izvedli decimacijo (še nekoliko smo zmanjšali količino podatkov za obdelavo in lažje računanje demodulacije). Matlab funkcija za decimacijo ima že sama vgrajen Low Pass filter, zato nam tega posebej ni potrebno početi. Decimiran signal smo demodulirali in še enkrat na njem izvedli decimacijo,

da je vzorčna frekvenca signala enaka vzorčni frekvenci zvočne kartice, da lahko dani signal poslušamo.

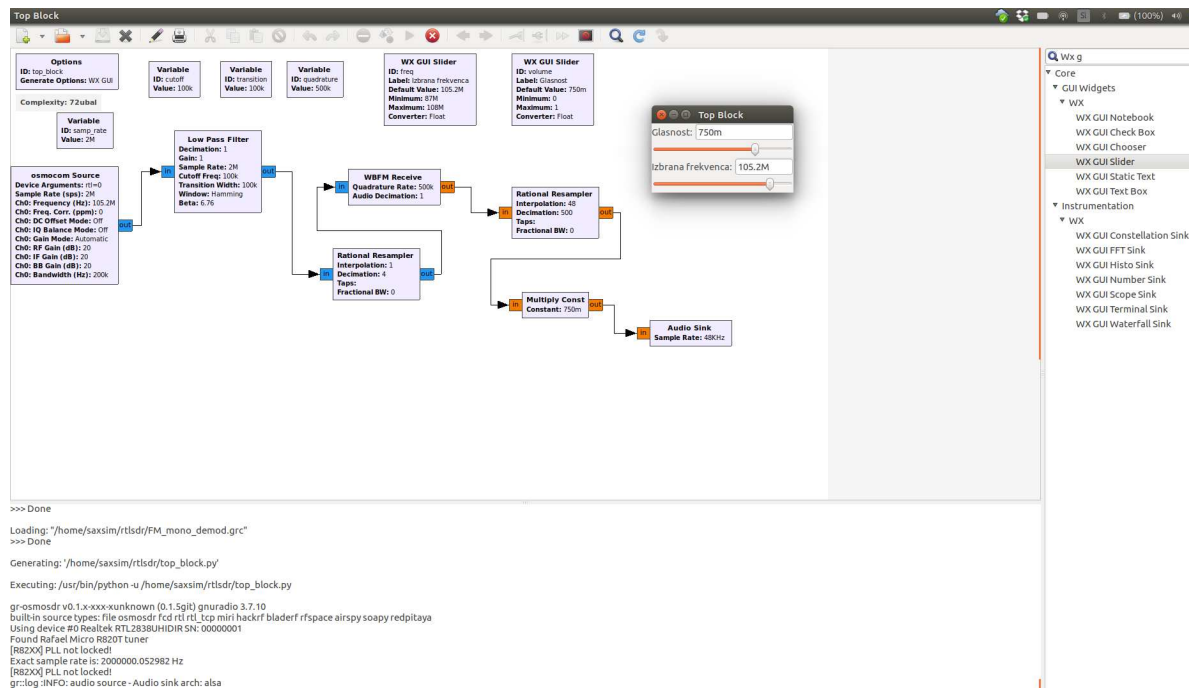


### 3. DEMODULACIJA FM IN RDS S PROGRAMSKIM PAKETOM GNU Radio Companion

GNU Radio je odprtokoden projekt, namenjen obdelavi signalov. Za uporabniški vmesnik uporablja program GNU Radio Companion, ki je po konceptu podoben programskemu paketu Simulink, ki je del Matlaba. GNU Radio deluje v okolju Linux, kjer uporabljen kot »backend« za sprejem in obdelavo signalov v drugih programih (gqrx, SDR#, ...).

#### FM DEMODULACIJA

V GNU Radio Companion-u sestavimo blokovno shemo za FM demodulacijo mono signala.



Koncept FM demodulacije je enak kot pri demodulaciji v Matlabu. Signal spustimo preko nizkopasovnega sita in ga demoduliramo. Signal v GRC pripeljemo z blokom *osmocom Source*, kjer nastavimo frekvenco sprejemnika ter pasovno širino, s katero želimo zajemati. Nato signal peljemo čez nizkopropustno sito – blok *Low Pass Filter*, kjer čez spustimo samo pasovno širino radijskega FM kanala – 200kHz. Nato zaradi lažje demodulacije signal decimiramo v bloku *Rational Resampler* in ga pripeljemo na blok *WBFM Receive*, kjer opravimo FM demodulacijo. Signal nato prevzorčimo v bloku *Rational Resampler*, kjer ga pripravimo na vzorčno frekvenco, ki je v območju vzorčenja naše zvočne kartice. Tako dobljen signal množimo s konstanto (blok *Multiply Const*), da nastavimo glasnost signala, nato pa ga pripeljemo na blok *Audio Sink*, kjer frekvenco vzorčenja zvočne kartice izenačimo prevzorčeni frekvenci.

Pri izdelavi blokovne sheme koristimo modula *WX GUI Slider*, ki nam omogočita dinamično grafično spreminjanje spremenljivk **freq**, ki nastavlja centralno frekvenco zajema ter **volume**, ki nastavlja konstanto, s katero množimo izhodni signal pred izhodom na zvočno kartico, s čimer nastavljamo glasnost.

Po shranitvi in izvedbi blokovne sheme se nam odpre *Top Block*, v katerem lahko nastavljamo glasnost in frekvenco. Prav tako lahko slišimo demoduliran signal na izbrani frekvenci. Pri izbrani frekvenci 105.2 MHz je to Radio Antena.

## FM IN RDS DEMODULACIJA

Danes večina radijskih postaj poleg analognega audio signala oddaja tudi digitalni RDS signal, ki je moduliran z DSB-SC modulacijo s centralno frekvenco 3 harmonika stereo pilotnega

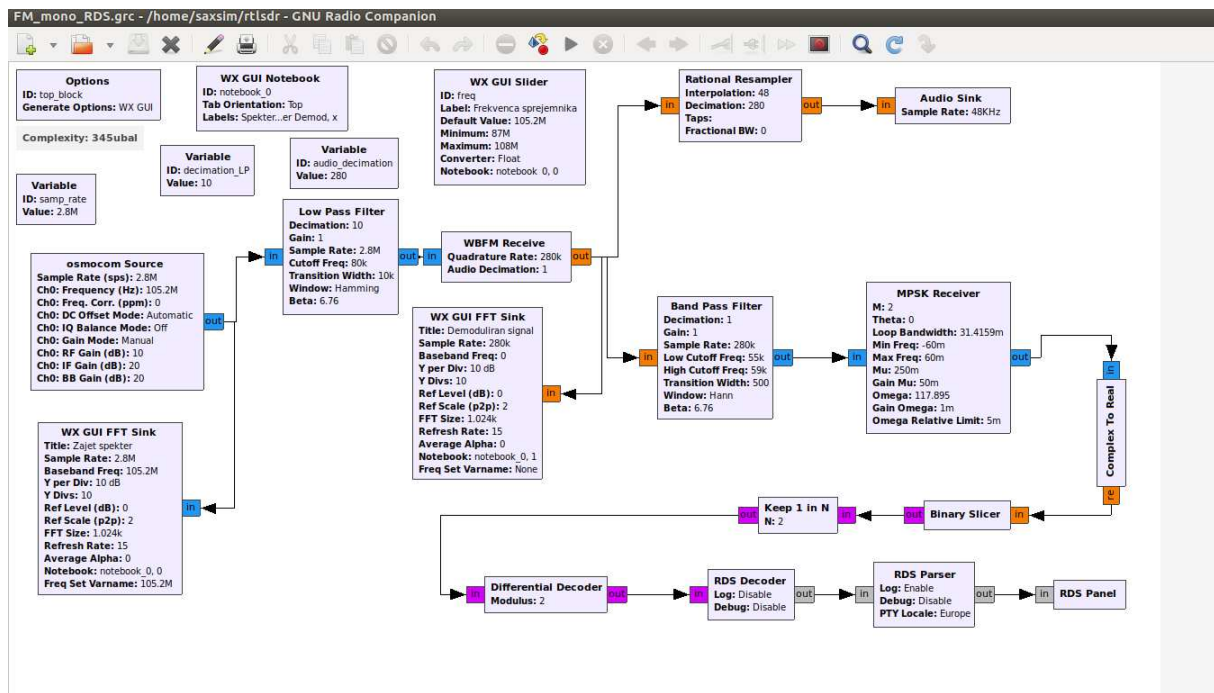
tona ( $3 \times 19\text{kHz} = 57\text{ kHz}$ ). RDS nam omogoča, da ob sprejemu audio signala prejmemo tudi informacije o radijski postaji, trenutni vsebini, alternativnih frekvencah, ...

Za izdelavo blokovne sheme bomo vzeli osnovo iz prejšnjega primera, ki mu bomo dodali elemente, ki nam bodo omogočili demodulacijo, sprejem, dekodiranje in prikaz RDS podatkov.

Signal v basebandu za RDS modulacijo poberemo iz odcepa iz modula *WBFM Receive*. Nato ga peljemo čez pasovni filter, preko katerega spustimo samo frekvence med 55 kHz in 59 kHz (območje RDS signala). Ker gre pri DSB-SC v bistvu za digitalno PSK modulacijo po modulu 2, signal peljemo na blok *MPSK Receiver*, kjer nastavimo modul demodulacije na 2 in faktor omega, ki pomeni razmerje vzorcev na simbol. Ker vemo, s kakšno vzorčno frekvenco prejemamo signale v demodulator, hkrati pa poznamo tudi bitno hitrost RDS (1178.5 b/s), vemo pa tudi, da imamo v izbranem frekvenčnem pasu dvojno hitrost prenosa zaradi Manchester kodiranja, nastavimo faktor omega na vzorčno frekvenco /  $2 \times$  bitno hitrost RDS. Z blokom *Complex To Real* kompleksne vrednosti pretvorimo v realne, nato pa jih z modulom *Binary Slicer* spremenimo v bite. Ker imamo zaradi dvojne bitne hitrosti vzorce podvojene, jih odstranimo z blokom *Keep 1 in N*, nato diferencialno dekodiramo z blokom *Differential*



Decoder ter pripeljemo na blok *RDS Decoder*. Dekodiran RDS signal nato na bloku *RDS Parser* sestavimo skupaj ter prikažemo na bloku *RDS Panel*.



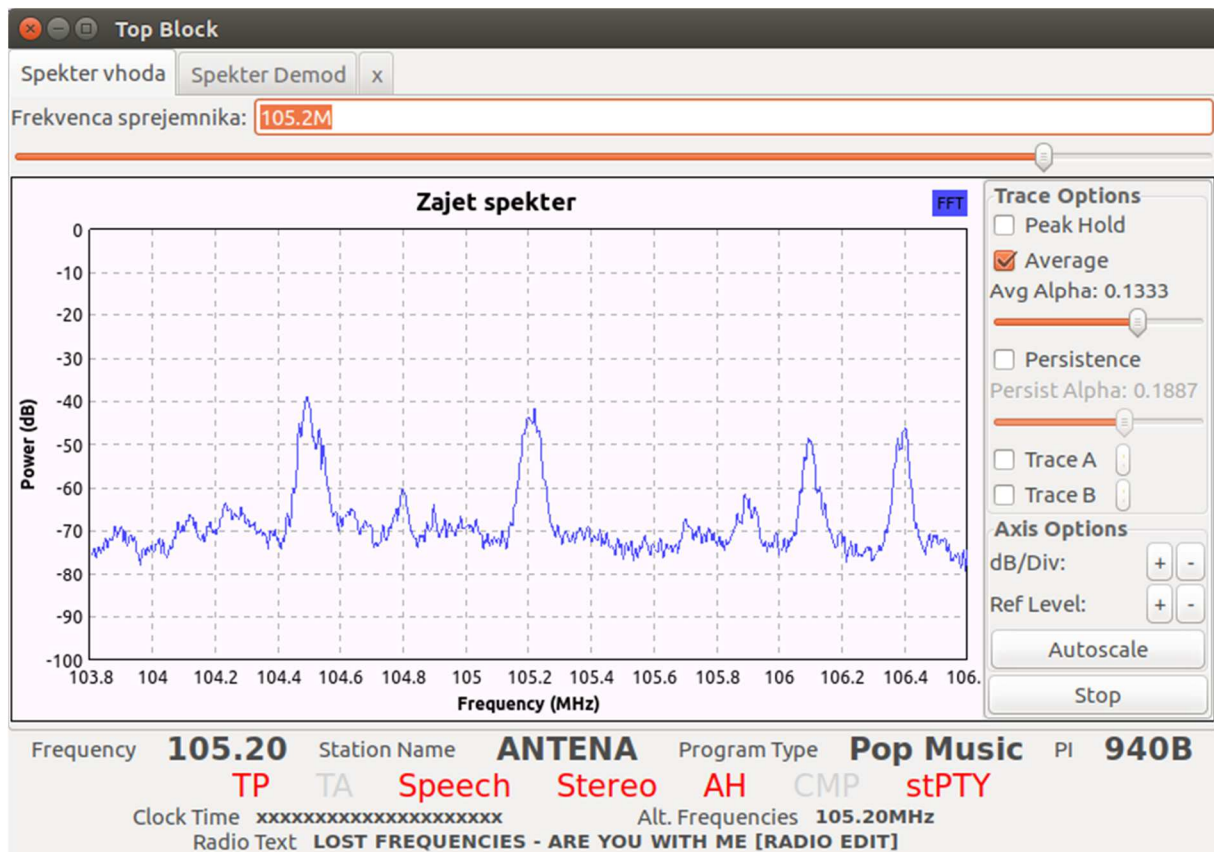
Generating: '/home/saxsim/rtlsdr/top\_block.py'

Executing: '/usr/bin/python -u /home/saxsim/rtlsdr/top\_block.py'

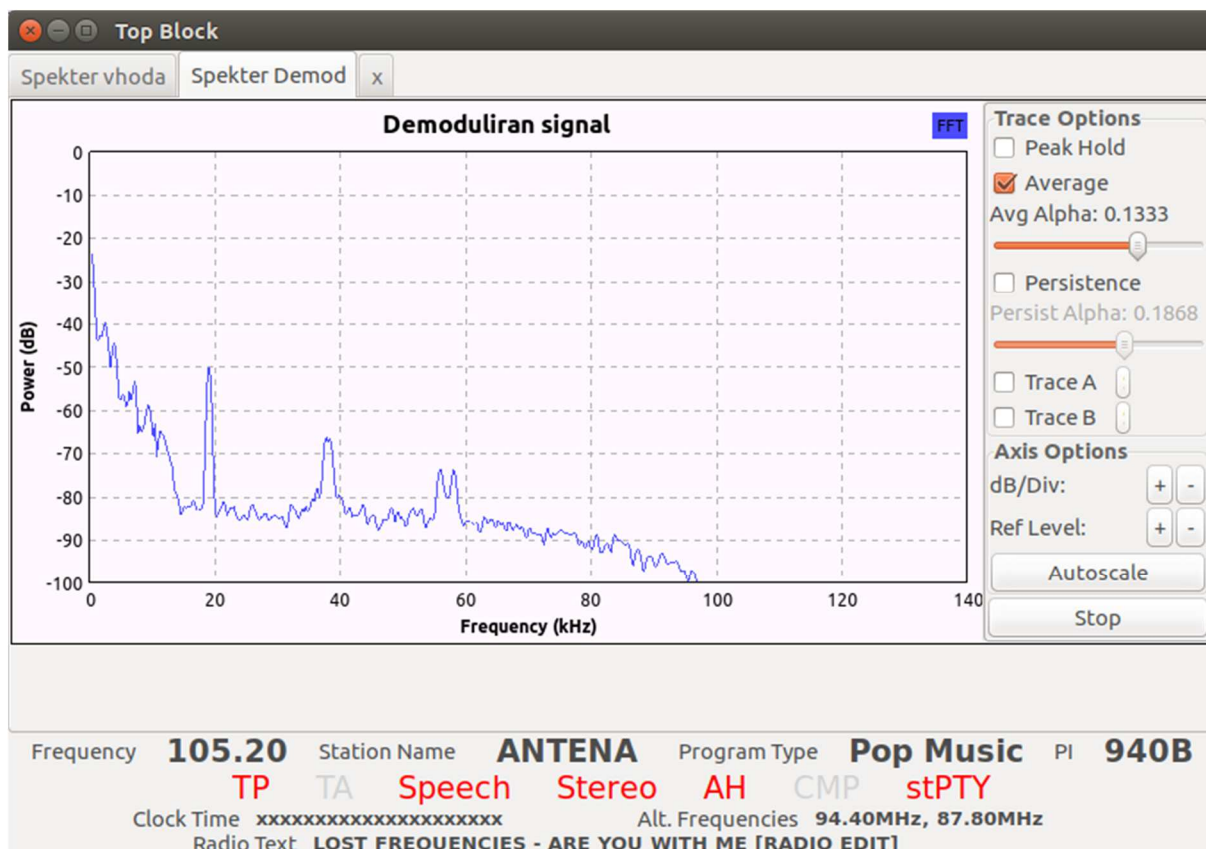
```
gr-osmosdr v0.1.x-xxx-xunknown (0.1.5git) gnuradio 3.7.10
built-in source types: file osmosdr fcd rtl_tcp miri hackrf bladerf rfspice airspy soapy redpitaya
Using device #0 Realtek RTL2838UHIDIR SN: 00000001
Found Rafael Micro R820T tuner
[R82XX] PLL not locked!
Exact sample rate is: 2000000.052982 Hz
[R82XX] PLL not locked!
gr::log :INFO: audio source - Audio sink arch: alsaaUaUaUaUaUaUaU
>>> Done (return code -9)
```

Loading: "/home/saxsim/rtlsdr/FM\_mono\_RDS.grc"

Po shranitvi in izvedbi blokovne sheme se nam odpre panel s spektralno analizo zajetega IQ signala s SDR RTL in RDS podatki:



Prav tako pa smo naredili spektralno analizo demoduliranega signala v basebandu, kjer je lepo razviden stereo pilotni signal ter oba boka RDS DSB-SC moduliranega signala.



Poleg imena radijske postaje nam modul prikazuje tudi tip programa, vsebino programa, trenutno predvajano glasbo ter bitno hitrost prenosa podatkov. Ta znaša 940 bitov/s, kar je manj od 1178.5 bitov/s, kar je rezultat kodiranja protokola RDS.

#### 4. ZAKLJUČEK

RTL SDR se je pri izvedbi projekta izkazal kot izjemno uporabno orodje, bodisi kot spektralni analizator za pregled spektra okoli nas, ki nam odpre vrata v nov svet (»kaj je v zraku?«), kot tudi pripomoček za zajem digitalnih signalov, namenjenih nadaljnji obdelavi. S svojo cenovno dostopnostjo in širokimi možnostmi uporabe je orodje, ki ga potrebuje vsak študent. V našem projektu smo sprva načrtovali dostop do RTL SDR preko TCP/IP protokolnega sklada, kar bi omogočilo oddaljen dostop in uporabo RTL SDR vsem članom skupine, vendar smo

zaradi velike količine zajetih podatkov in posledično visoke bitne hitrosti ta načrt opustili. Z optimizacijo programske opreme rtl\_tcp bi v prihodnje to lahko rešili.

Pri pregledu rezultatov, ki so odlični – imamo zvok in imamo RDS podatke, na vsaki točki pa imamo možnost preveriti, kako izgleda spekter in signal, kar je odlično tudi za samo predstavo obdelav, pa se odpira še en izziv – demodulacija stereo FM signala.

## 5. VIRI

- [1] A. Umek, Digitalne komunikacije– laboratorijske vaje, Ljubljana: Fakulteta za elektrotehniko, 2017
- [2] Whiteboard Web. *I/Q Data for Dummies*. [Online]. Dosegljivo: <http://whiteboard.ping.se/SDR/IQ>. [Dostopano: 15. maj 2017]
- [3] Aaron Scher, Course materials on software defined radio (SDR). *How to capture raw IQ data from a RTL-SDR dongle and FM demodulate with MATLAB*. [Online]. Dosegljivo: [http://aaronscher.com/wireless.com/SDR/RTL SDR AM spectrum demod.html](http://aaronscher.com/wireless.com/SDR/RTL%20SDR%20AM%20spectrum%20demod.html). [Dostopano: 15. maj 2017]
- [4] Bastian Bloessl. *First Steps in Receiving Digital Information with RDS/TMC*. [Online]. Dosegljivo: [https://archive.fosdem.org/2015/schedule/event/sdr\\_rds\\_tmc/](https://archive.fosdem.org/2015/schedule/event/sdr_rds_tmc/). [Dostopano: 4. junij 2017]
- [5] University of Victoria, Electrical and Computer Engineering. *GNU Radio Companion, Block Documentation*. [Online]. Dosegljivo: [https://www.ece.uvic.ca/~elec350/grc\\_doc/](https://www.ece.uvic.ca/~elec350/grc_doc/). [Dostopano: 4. junij 2017]
- [6] Wikipedia, The Free Encyclopedia. *Radio Data System*. [Online]. Dosegljivo: [https://en.wikipedia.org/wiki/Radio\\_Data\\_System](https://en.wikipedia.org/wiki/Radio_Data_System). [Dostopano: 5. junij 2017]
- [7] GNU Radio, The free & open software radio ecosystem. *Wiki*. [Online]. Dosegljivo: [https://wiki.gnuradio.org/index.php/Main\\_Page](https://wiki.gnuradio.org/index.php/Main_Page). [Dostopano: 6. junij 2017]
- [8] Github. *gr-rds, FM RDS/TMC Transceiver*. [Online]. Dosegljivo: <https://github.com/bastibl/gr-rds>. [Dostopano: 6. junij 2017]