



UNIVERZA  
V MARIBORU



FAKULTETA ZA ELEKTROTEHNIKO,  
RAČUNALNIŠTVO IN INFORMATIKO  
2000 Maribor, Smetanova ulica 17

Marko Kočevar

**NEREKURZIVNO DIGITALNO SITO 63. STOPNJE V  
PORAZDELJENI ARITMETIKI V POLJU PROGRAMIRNIH  
LOGIČNIH VEZIJ**

Diplomska naloga

MARIBOR, september 2008



UNIVERZA  
V MARIBORU



FAKULTETA ZA ELEKTROTEHNIKO,  
RAČUNALNIŠTVO IN INFORMATIKO  
2000 Maribor, Smetanova ulica 17

Diplomsko delo visokošolskega strokovnega programa

# NEREKURZIVNO DIGITALNO SITO 63. STOPNJE V PORAZDELJENI ARITMETIKI V POLJU PROGRAMIRNIH LOGIČNIH VEZIJ

Študent: Marko Kočevar  
Študijski program: visokošolski strokovni, Elektrotehnika  
Smer: Elektronika

Mentor: izred. prof. dr. Rudolf Babič  
Sommentor: asist. dr. Davorin Osebik

Maribor, september 2008



UNIVERZA  
V MARIBORU



FAKULTETA ZA ELEKTROTEHNIKO,  
RAČUNALNIŠTVO IN INFORMATIKO  
2000 Maribor, Smetanova ulica 17

Številka: E.0752

Datum: 15.09.2008

## SKLEP O DIPLOMSKEM DELU

1. **Marko Kočever**, študent visokošolskega strokovnega študijskega programa Elektrotehnika, smer Elektronika, izpolnjuje pogoje, zato se mu dovoljuje izdelati diplomsko delo.
2. **Tema diplomskega dela** je s področja Inštituta za elektroniko in telekomunikacije pri predmetu **PRAKTIKUM**

**MENTOR:** izred. prof. dr. Rudolf Babič  
**SOMENTOR:** asist. dr. Davorin Osebik

3. **Naslov diplomske dela:**  
**NEREKURZIVNO DIGITALNO SITO 63. STOPNJE V PORAZDELJENI ARITMETIKI V POLJU PROGRAMIRNIH LOGIČNIH VEZIJ**
4. **Vsebina diplomske naloge**  
V programirnem polju logičnih vezij na razvojni enoti Spartan 3 s FPGA vezjem XC3S400 načrtajte nerekurzivno digitalni sito s 64 koeficienti. Izmerite in primerjajte različne frekvenčne karakteristike sit pri manj in bolj strogih pogojih izbire prehodnega frekvenčnega področja. Meritve odzivov digitalnega sita izvedite z enotnim signalom, sinusnim signalom in z belošumnim signalom na vhodu. Za primerjavo med teoretičnim izračunom in merilnimi rezultati uporabite okolje MATLAB.
5. Diplomsko nalogo izdelajte skladno z "Navodili za izdelavo diplomske naloge" in jo oddajte v 3 izvodih do 15.09.2009 v referatu za študijske zadeve.

PREDSTOJNIK INŠTITUTA

DEKAN

### ZAHVALA

Iskreno se zahvaljujem mentorju Izr. prof. dr. Rudolfu Babiču in še posebej doc. dr. Davorinu Osebiku za vsestransko pomoč pri izdelavi diplomske naloge.

Posebno se zahvaljujem tudi mojim staršem in vsem ostalim, ki so me podpirali pri študiju.

# NEREKURZIVNO DIGITALNO SITO 63. STOPNJE V PORAZDELJENI ARITMETIKI V POLJU PROGRAMIRNIH LOGIČNIH VEZIJ

**Ključne besede:** digitalna sita, nerekurzivna digitalna sita, porazdeljena aritmetika, programirna FPGA vezja

**UDK:** 621.391 (043.2)

## **Povzetek**

V diplomski nalogi je opisano in izvedeno nerekurzivno digitalno sito 63. stopnje v strukturi porazdeljene aritmetike. Izračunali in izmerili smo različne frekvenčne karakteristike sit, pri manj in bolj strogih pogojih izbire prehodnega frekvenčnega področja, ter na podlagi tega dobljeno odstopanje oz. pogrešek. Opisan je postopek povezave programskega okolja MATLAB z razvojno enoto Spartan 3 s FPGA vezjem XC3S400. Samo implementacijo v programirljivo vezje FPGA smo izvršili s programskim paketom Xilinx ISE. Izmerili in opisali smo odziv FIR sita na belošumni vhodni signal, na enotin impulz (dirak), stopničasti signal in signal sinusne oblike.

# IRRECURSIVE DIGITAL SIEVE OF THE 63. LEVEL IN DISTRIBUTIVE ARITHMETIC IN THE FIELD OF PROGRAMMED LOGICAL CHIPSETS

**Key words:** digital sieve, irrecursive digital sieve, distributive arithmetic, programmed FPGA chipsets, USB interface, Matalab.

UDK: 621.391 (043.2)

## Summary

In the diploma there is a description and execution of an irrecursive digital sieve of the 63. level in a structure of a distributive arithmetic. We have calculated and measured different frequency characteristics of the sieves under more and less strict conditions of predetermined frequency area; through these measurements we were able to obtain some irregularities. There is also a description of the procedure of combining the software MATALAB environment with the Spartan 3 development unit, which has a XC3S400 FGPA chipset. The implementation into the programmable FGPA chipset was achieved by the Xilinx ISE package. We have also measured and described the response of the FIR sieves on the white-noised input signal, homogenous impulse (dirak), gradated signal and sinus shaped signal.

## KAZALO

<b>1</b>	<b>UVOD</b> .....	<b>1</b>
1.1	Namen in cilj diplomskega dela.....	1
1.2	Predpostavke in omejitve raziskave.....	2
1.3	Predvidene metode raziskave.....	2
<b>2</b>	<b>DIGITALNA SITA</b> .....	<b>3</b>
2.1	Koncentrirana aritmetika.....	4
2.2	Porazdeljena aritmetika.....	4
2.2.1	Digitalno FIR sito s 64 koeficienti v strukturi porazdeljene aritmetike s sprotnim izračunom delne vsote koeficientov.....	5
<b>3</b>	<b>XILINX PROGRAMIRNA VEZJA</b> .....	<b>9</b>
3.1	Programiranje FPGA vezja s 64 koeficienti.....	11
<b>4</b>	<b>ORODJA</b> .....	<b>19</b>
4.1	Opis razvojne enote.....	19
4.2	Programirno FPGA vezje Spartan 3 serije XC3S400.....	20
4.3	Razvojno okolje MATLAB in povezava z FPGA vezjem.....	23
4.4	Serijski EEPROM za shranjevanje konfiguracijskih podatkov.....	23
4.5	Inicializacija razvojne enote.....	24
<b>5</b>	<b>SIGNALI</b> .....	<b>25</b>
5.1	Kvantizacija signalov.....	25
5.2	Enotin impulz ali DIRAK.....	27
5.2.1	Predstavitev enotinega impulza v MATLABU.....	28
5.3	Stopnični signal.....	29
5.3.1	Predstavitev stopničnega signala v MATLABU.....	29
5.4	Signal sinusne oblike.....	30
5.4.1	Predstavitev sinusnega signala v MATLABU.....	30
5.5	Belošumni signal.....	31
5.5.1	Predstavitev belošumnega signala v MATLABU.....	32
<b>6</b>	<b>MERITEV</b> .....	<b>34</b>
6.1	Odziv FIR sita z 64 koeficienti na različne vhodne signale.....	34
6.1.1	Odziv FIR sita na belošumni signal.....	36
6.1.2	Odziv FIR sita na enotin impulz.....	38
6.1.3	Odziv FIR sita na stopnični signal.....	40
6.1.4	Odziv FIR sita na signal sinusne oblike.....	41
6.2	Amplitudni odziv FIR sita s 64 koeficienti za različno zahtevane oblike frekvenčnih karakteristik z belošumnim signalom na vhodu.....	43
<b>7</b>	<b>REZULTATI</b> .....	<b>46</b>
7.1	Visokoprepustno sito.....	46
7.2	Visokoprepustno sito ostrejše zahteve.....	48
7.3	Nizkoprepustno sito.....	50
7.4	Nizkoprepustno sito ostrejše zahteve.....	53
7.5	Pasovnoprepustno sito.....	55

7.6	Pasovnozaporno sito.....	57
8	SKLEP.....	59
9	LITERATURA.....	60
10	PRILOGE .....	61

## KAZALO SLIK

Slika 1:	Delitev digitalnih sit .....	3
Slika 2:	Nerekurzivno digitalno FIR sito s 64 koeficienti v strukturi porazdeljene aritmetike ..	6
Slika 3:	Postopek programiranja FPGA vezja .....	11
Slika 4:	Izvedba nerekurzivnega digitalnega FIR sita s 64 koeficienti .....	12
Slika 5:	Izvedba nerekurzivnega FIR sita v porazdeljeni aritmetiki z več podstrukturami.....	12
Slika 6:	16 ROM pomnilnikov za pomnjenje karakterističnih delnih vsot koeficientov .....	13
Slika 7:	Podstruktura ROM pomnilnika, kjer so zapisane KDVK.....	13
Slika 8:	IMPACT za pripravo EEPROM datoteke.....	14
Slika 9:	Nalaganje vsebine bitne datoteke imena gl_vezje.bit v FPGA vezje .....	15
Slika 10:	Naložena vsebina bitne datoteke v FPGA vezju .....	15
Slika 11:	Imenovanje mcs datoteke.....	16
Slika 12:	Izbira vrste EEPROM-a (xcf04s) .....	16
Slika 13:	Pogovorno okno za dodajanje FPGA VEZJA z vsebino bitne datoteke gl_vezje.bit	17
Slika 14:	Pogovorno okno za dodajanje FPGA vezja z vsebino bitne datoteke pom_vezje_32bitni_prenos.bit .....	17
Slika 15:	Generiranje mcs datoteke.....	18
Slika 16:	Programiranje .....	18
Slika 17:	Prikaz komunikacije med programskim delom in FPGA vezjem .....	19
Slika 18:	Zasnova razvojne plošče; Vir[1] .....	20
Slika 19:	Notranja zgradba Spartan 3 programirnega vezja; Vir[1].....	21
Slika 20:	Razvojna plošča - pogled zgoraj; Vir [1].....	22
Slika 21:	Razvojna plošča – pogled spodaj; Vir [1].....	22
Slika 22:	Povezava preko dveh USB priključkov; Vir [6].....	23
Slika 23:	Inicializacijski kabel; Vir[1].....	24
Slika 24:	Diracov impulz; Vir [4] .....	27
Slika 25:	Premaknjeni enotin impulz za 100 otipkov.....	28
Slika 26:	Enotina stopnica; Vir [4].....	29
Slika 27:	Premaknjeni stopnični signal za 100 otipkov (impz) .....	29
Slika 28:	Sinusni signal v MATLABU.....	30
Slika 29:	Časovni potek Gaussovega signala belega šuma s 1024 vzorci v periodi in njegova gostota spektra moči pri $f_v=100\text{kHz}$ .....	31
Slika 30:	Porazdelitev amplitud signala belega šuma s 1024 vzorci v periodi.....	32
Slika 31:	Predstavitve spektra moči belošumnega signala .....	33
Slika 32:	Blokovna shema za merjenje odziva FIR sita .....	35



Slika 33: Odziv FIR sita na belošumni signal – 1024 otipkov.....	37
Slika 34: Ojačanje FIR sita s 64 koeficienti na belošumni signal.....	38
Slika 35: Odziv FIR sita na enotin impulz.....	39
Slika 36: Odziv FIR sita na enotin impulz – impulzni način.....	39
Slika 37: odziv FIR sita na stopnični signal.....	40
Slika 38: Odziv FIR sita na stopnični signal – impulzni način.....	41
Slika 39: Odziv FIR sita na signal sinusne oblike – ena perioda.....	42
Slika 40: Definiranje frekvenčnega in amplitudnega vektorja.....	45
Slika 41: Izračunan amplitudni odziv visokoprepustnega sita.....	47
Slika 42: Izračunan in izmerjen amplitudni odziv visokoprepustnega sita.....	47
Slika 43: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	48
Slika 44: Izračunan amplitudni odziv visokoprepustnega sita ostrejše zahteve.....	49
Slika 45: Izračunan in izmerjen amplitudni odziv visokoprepustnega sita ostrejše zahteve....	49
Slika 46: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	50
Slika 47: Izračunan amplitudni odziv nizkoprepustnega sita.....	51
Slika 48: Izračunan in izmerjen amplitudni odziv nizkoprepustnega sita.....	52
Slika 49: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	52
Slika 50: Izračunan amplitudni odziv nizkoprepustnega sita ostrejše zahteve.....	53
Slika 51: Izračunan in izmerjen amplitudni odziv nizkoprepustnega sita ostrejše zahteve.....	54
Slika 52: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	54
Slika 53: Izračunan amplitudni odziv pasovnoprepustnega sita.....	55
Slika 54: Izračunan in izmerjen amplitudni odziv pasovnoprepustnega sita.....	56
Slika 55: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	56
Slika 56: Izračunan amplitudni odziv pasovnozaporenega sita.....	57
Slika 57: Izračunan in izmerjen amplitudni odziv pasovnozaporenega sita.....	58
Slika 58: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek.....	58

## Uporabljene kratice

<b>FPGA</b>	Field Programmeable Gate Array – Logična programirna vezja
<b>USB</b>	Universal Serial Bus – Univerzalno serijsko vodilo
<b>FIR</b>	komponent Finite Impulse Response – Filter z omejenim odzivom
<b>IIR</b>	Infinite impulse response – Filter z neomejenim odzivom
<b>FFT</b>	Fast Fourier transform - Hitra Fourijeve transformacija
<b>TTL</b>	Transistor-Transistor Logic - Tranzistor-tranzistor logika
<b>SMD</b>	Surface Mount Devices – Tehnologija površinskega nameščanja elektronskih komponent
<b>EEPROM</b>	Electrically Erasable Programmable Read Only Memory – Električno zbrisljiv in programirljiv bralni pomnilnik
<b>JTAG</b>	Joint Test Action Group ali Standard Test Access Port and Boundary-Scan Architecture - Standard IEEE 1149.1, standardizirano testiranje elektronskih komponent ali sklopov
<b>VHDL</b>	Very High Speed Integrated Circuit Hardware Description Language - jezik za opis zelo hitrih integriranih vezij komponent ali sklopov
<b>I/O</b>	Input Output Connector – Vhodno izhodni priključek
<b>DSP</b>	Digital Signal Processing – Digitalno signalno procesiranje
<b>PZ</b>	Pasovnozaporno sito
<b>VS</b>	Visokoprepustno sito
<b>NS</b>	Nizkoprepustno sito
<b>PP</b>	Pasovnoprepustno sito

## 1 UVOD

Digitalna obdelava signalov je dinamično in hitro razvijajoče se področje elektronike, predvsem na področju komunikacij in prenosu signalov v digitalni obliki. Glavni veji digitalne obdelave signalov sta digitalno filtriranje in spektralna analiza. V laboratoriju za elektronske sisteme že več let razvijamo digitalna sita. Sprva so bile to le računalniške simulacije algoritmov na računalnikih, v zadnjem času pa tudi aparturne izvedbe digitalnih sit. Zanimajo nas predvsem nerekurzivna digitalna sita, ki so izvedena v strukturi porazdeljene aritmetike [3].

FPGA vezja so integrirana vezja, v katerih lahko sami načrtujemo svojo strukturo digitalnega vezja ali sistema v integrirani obliki. Polje programirnih logičnih gradnikov nam omogoča načrtovanje od enostavnih logičnih funkcij, pa vse do zapletenih digitalnih sistemov. Programirna vezja delujejo podobno kot mikrokrmilniški sistemi, ki uporabljajo razne vrste pomnilnikov, v katerih je zapisana funkcijska tabela.

Obstaja več proizvajalcev FPGA - vezij. Naša razvojna plošča uporablja FPGA vezje firme Xilinx [4], ki se v laboratoriju za elektronske sisteme (LES) uporabljajo že dlje časa. Z vse zmogljivejšimi integriranimi vezji pa je možno tudi enostavno načrtovanje sistemov za digitalno obdelavo signalov in s tem tudi digitalnih sit. Z uporabo LCA vezij se je spremenila izvedba digitalnih sit v aparturni izvedbi.

Diplomsko delo začnemo z uvodom, v drugem poglavju so predstavljena digitalna sita in struktura v porazdeljeni aritmetiki, v tretjem poglavju so opisana Xilinx programirna vezja in prenos podatkov v FPGA vezje, četrto poglavje obsega orodja, s katerimi smo prišli do rezultatov meritev, v petem poglavju so predstavljeni signali, v šestem poglavju so opisane meritve našega dela, sedmo poglavje zajema rezultate, v osmem poglavju je zapisan sklep, katerim sledita še dva poglavja, literatura in priloge.

### 1.1 Namen in cilj diplomskega dela

Namen diplomskega dela je spoznati in predstaviti FIR sita, njihovo implementacijo v FPGA vezja, spoznati programsko okolje MATLAB in njegovo povezavo z razvojno ploščo Spartan 3 s FPGA vezjem XC3S400. Cilj diplomskega dela je načrtovanje in izvedba digitalnega FIR sita s 64 koeficienti, ter primerjava izračunanih in izmerjenih različnih frekvenčnih karakteristik, pri manj in bolj strogih pogojih izbire prehodnega frekvenčnega področja, ter na podlagi tega dobljeno odstopanje oz. pogrešek. Odziv FIR sita bomo izmerili na belošumni vhodni signal, na enotin imulz (dirak), stopnični signal in signal sinusne oblike.

## 1.2 Predpostavke in omejitve raziskave

Merili bomo digitalna FIR sita s 64 koeficienti. Amplitudo in delne vsote koeficientov bomo normirali na vrednost ena. Frekvenca vzorčenja bo omejena z razvojnim orodjem. Za vhodni signal bomo pri merjenju amplitudnega odziva uporabili Gaussov belošumni signal.

## 1.3 Predvidene metode raziskave

Pri teoretičnem delu načrtovanja diplomske naloge si bomo pomagali z gradivom, ki je navedeno, v seznamu predvidenih virov, v poglavju 9. Za primerjavo med teoretičnim izračunom in merilnimi rezultati bomo uporabili okolje MATLAB in že napisane skriptne datoteke, ki omogočajo prikaz rezultatov v grafični obliki. Dejanska meritev se bo izvajala na razvojni plošči v kateri sta uporabljeni dve Spartan 3 integrirani vezji iz serije XC3S400. Razvojna plošča je diplomsko delo Andreja Greifonerja in je priključena preko JTAG vmesnika na računalnik. Samo implementacijo v programirljivo FPGA vezje XC3S400 pa bomo izvršili s programskim paketom Xilinx ISE.

## 2 DIGITALNA SITA

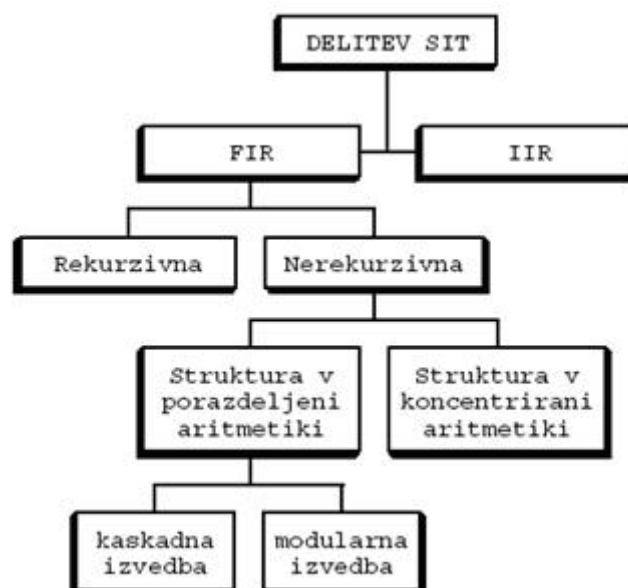
Digitalno sito zajema vhodni digitalni signal in ga po vnaprej določenem algoritmu obdela. Algoritme so razvili avtorji že pred desetletji, vendar je šele uporaba računalnika omogočila njihovo uporabo v praksi. Veliko digitalnega procesiranja signalov (DPS), ki temelji na ustrezno zapisanem algoritmu, opravljajo danes specializirani mikroprocesorji, imenovani digitalni signalni procesorji (DSP), ki so sposobni izvajanja zelo hitrih množenj.

Glavna lastnost digitalnega sita je, da določen frekvenčni pas prepušča frekvence, ostale pa duši, kar je določeno v algoritmu. Pomemben faktor je slabljenje digitalnega sita, katerega velikost je odvisna od reda digitalnega sita. Algoritem, ki je zapisan v obliki programa ali rutine, je napravljen tako, da želen frekvenčni pas prepušča, neželenega pa čim bolj slabi.

### Razdelitev digitalnih sit

Osnovna delitev poteka glede na odziv digitalnih sit, ki je lahko omejen (Finite Impulse Response) ali neomejen (Infinte Impulse Response).

**FIR** sita [2] se delijo glede na prenosno funkcijo, na sita s povratno vezavo (rekurzivna) in sita brez povratne vezave (nerekurzivna). Nerekurzivna sita se delijo glede na izvedbeno strukturo s koncentrirano aritmetiko in strukturo s porazdeljeno aritmetiko. Glede na kompleksnost izvedbe delimo sita na kaskadna in modularna. Blokovno shemo delitev digitalnih sit prikazuje slika 1.



Slika 1: Delitev digitalnih sit

## 2.1 Koncentrirana aritmetika

Digitalna FIR sita, ki so izvedena v strukturi koncentrirane aritmetike (z uporabo množilnikov) se uporabljajo v sistemih adaptivnih digitalnih FIR sit za digitalno procesiranje signalov (DPS).

Pri digitalnih FIR sitih izvedenih v strukturi koncentrirane aritmetike, je potrebno pri vsakem otipku izvesti produkt vektorjev po enačbi (1).

$$y(k) = \mathbf{h}(k)^T \mathbf{u}(k) \quad (1)$$

kjer je:  $\mathbf{h}(k)$  – vektor koeficientov sita

$\mathbf{u}(k)$  – vektor koeficientov vhodnega signala

## 2.2 Porazdeljena aritmetika

S porazdeljeno aritmetiko [2] označujemo postopek izračuna skalarnega produkta dveh vektorjev na bitnem nivoju brez uporabe množilnikov. Izračun izhodne vrednosti  $y$  digitalnega FIR sita podaja enačba,

$$y = \mathbf{h}^T \mathbf{u} = \sum_{n=0}^{M-1} h_n u_n \quad (2)$$

V enačbi (2) je z  $h_n$  označena komponenta vektorja konstantnih vrednosti  $\mathbf{h}$ , z  $u_n$  pa komponenta vektorja  $\mathbf{u}$ . Za izračun izhodne vrednosti  $y$  potrebujemo  $N$  množenj. Izračun delne vsote koeficientov podaja enačba (3),

$$v_{n,0} = \sum_{n=0}^{N-1} h_n b_{n,0} \quad (3)$$

kjer predstavlja  $v_{n,0}$ , izračun delnih vsot za najbolj utežen bit  $b_{n,0}$ , vhodne besede  $u_k$ .

$$v_{n,i} = \sum_{n=0}^{N-1} h_n b_{n,i} \quad i = 0, 1, \dots, B_u - 1 \quad (4)$$

Enačba (4) predstavlja izraz za izračun delnih vsot  $v_{n,i}$ , za preostale bite  $b_{n,i}$  vhodne besede  $u(k)$ . Z  $\mathbf{h}$  določimo vektor koeficientov digitalnega FIR sita z  $N$  koeficienti. Izhodno vrednost sita  $y(k)$  izračunamo po enačbi (5),

$$y(k) = \sum_{i=1}^{B_u-1} v_i(k-1)2^{-i} - v_{B_u}(k-1) \quad (5)$$

ki podaja izračun izhodnega signala po principu porazdeljene aritmetike in jo je možno enostavno realizirati v programirnem vezju. V enačbi (5) je z  $B_u$  določena dolžina binarne besede.

Pri digitalnih FIR sitih se spreminja tudi vektor koeficientov  $\mathbf{h}$ . Zaradi tega je potrebno delne vsote izračunavati sproti, za vsak otipek posebej. Pri implementaciji digitalnega FIR sita s 64 koeficienti, smo uporabili metodo izračuna delnih vsot na osnovi karakterističnih delnih vsot koeficientov.

## Digitalna sita v strukturi porazdeljene aritmetike

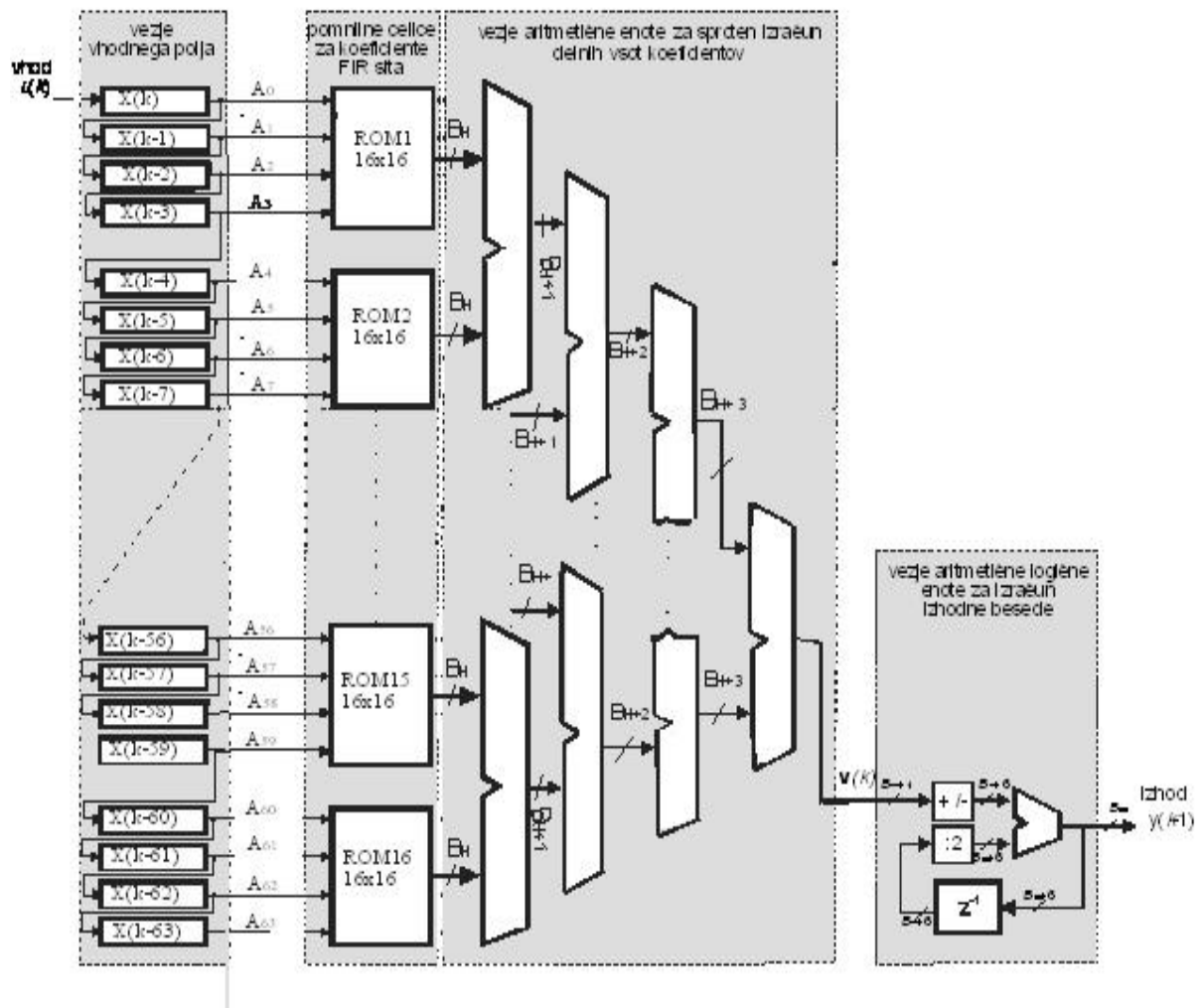
Digitalna FIR sita v strukturi porazdeljene aritmetike [2] delimo glede na njihove lastnosti, na aparaturne kompleksnosti, kot so rešitve nadomestitve klasičnih množilnikov, potrebne velikosti pomnilnika in potrebne velikosti vodil znotraj strukture, na:

- nerekurzivno digitalno sito v klasični obliki porazdeljene aritmetike
- struktura porazdeljene aritmetike z reduciranim številom naslovnih linij
- struktura porazdeljene aritmetike razdeljene na več podstruktur
- digitalno FIR sito v strukturi porazdeljene aritmetike s sprotnim izračunom delnih vsot koeficientov (uporabljen za naš primer)

### 2.2.1 Digitalno FIR sito s 64 koeficienti v strukturi porazdeljene aritmetike s sprotnim izračunom delne vsote koeficientov

Pri izvedbi nerekurzivnega digitalnega FIR [2] sita v porazdeljeni aritmetiki, ki je razdeljeno na več podstruktur, lahko teoretično izberemo toliko podstruktur, kot je število koeficientov digitalnega sita. Pri digitalnem FIR situ z  $N$  koeficienti je možno dobiti  $N$  podstruktur. Pri tem se izračun delnih vsot koeficientov v izračuna sproti iz koeficientov  $\mathbf{h}$ . Takšno obliko nerekurzivnega digitalnega FIR sita imenujemo tudi digitalno sito v strukturi porazdeljene

aritmetike s sprotnim izračunom delnih vsot koeficientov. V našem primeru smo uporabili metodo, ki uporablja predhodno izračunane karakteristične delne vsote koeficientov. Za zapis karakterističnih delnih vsot koeficientov, ki so kar enake koeficientom digitalnega sita, je potrebno pri tej strukturi zagotoviti le  $N$  lokacij. Strukturo takšne oblike prikazuje slika 2.



Slika 2: Nerekurzivno digitalno FIR sito s 64 koeficienti v strukturi porazdeljene aritmetike

Za popolno inicializacijo z vsemi koeficienti je v strukturo potrebno prenesti le koeficiente digitalnega sita. Prenos  $N$  koeficientov je možno izvesti v času enega otipka vhodnega signala. Kompleksnost predstavljenega digitalnega sita narašča sorazmerno s številom koeficientov digitalnega sita.

Digitalno FIR sito s 64 koeficienti na sliki 2 sestavljajo:

- vezje vhodnega polja, ki je zgrajeno v strukturi FIFO (First In- First Out),
- pomnilnik ROM, v katerem so predhodno izračunane delne vsote  $v_{n,i}(k)$ ,



- vezje aritmetične enote za sproten izračun delnih vsot koeficientov,
- vezje aritmetične enote za izračun izhodne besede.

### **Vezje vhodnega polja**

Z  $u(k)$  označujemo vhodno vrednost FIR sita. V postopku porazdeljene aritmetike je potrebno opraviti množenje vektorja koeficientov digitalnega FIR sita  $\mathbf{h}$  in vektorja koeficientov vhoda  $\mathbf{u}(k)$ . Pri tem postopku vrednosti vhodnega signala  $u(k)$  zapišemo v bitni obliki z dvojiškim komplementom. V vhodnem polju je zapisana bitna beseda dolžine  $B_u$  v dvojiški obliki. Z 1000...0 je zapisana najmanjša (negativna) vrednost, z 0111...1 pa največja (pozitivna) binarna vrednost.

### **Pomnilne celice za koeficiente FIR sita**

V ROM pomnilnikih imamo shranjene že v naprej izračunane karakteristične delne vsote koeficientov, ki so kar enake koeficientom digitalnega FIR sita. Velikost pomnilnika je določena s številom koeficientov  $2^N$  in številom bitov  $B_v$  za zapis delnih vsot. Pri izvedbi sita z velikim številom koeficientov impulznega odziva, postane uporaba osnovne strukture porazdeljene aritmetike problematična, zaradi eksponencialne rasti velikosti pomnilnika v odvisnosti od števila koeficientov impulznega odziva. To težavo smo odpravili z razporeditvijo delnih vsot koeficientov na več vzporednih pomnilnikov, kateri imajo mnogo manjše število lokacij kot en skupen pomnilnik.

Za naš primer digitalnega sita s 64 koeficienti potrebujemo 64 naslovnih linij. Zato lahko pomnilnike vežemo vzporedno in dosežemo želeno število naslovnih linij. S tem se ustrezno zmanjša število pomnilniških lokacij za vpis delnih vsot koeficientov  $d_{vi}$ . V en element ROM16XB $d_v$  lahko shranimo le 16 delnih vsot dolžine  $B_{d_v}$ . V 16 vzporednih pomnilnikov ROM16XB $d_v$  je tako možno shraniti le 256 delnih vsot. Vseh delnih vsot je  $16 \times 16$ , ki jih dosežemo tako, da se pomnilnik razdeli na dva enaka dela s po 16 naslovnimi linijami. V prvi pomnilnik se vpišejo vse delne vsote koeficientov od  $h_0$  do  $h_{15}$ , v drugega pa delne vsote preostalih koeficientov. Ob naslavljanju pomnilnikov je naslovljena ena lokacija v prvem in ena v drugem pomnilniku. Do popolnega rezultata enkratnega naslavljanja pridemo, če obe naslovljeni delni vsoti seštejemo in dobimo enak rezultat kot pri naslavljanju enotnega pomnilnika z  $2^{64}$  lokacijama. Torej, če razdelimo pomnilnik na dva dela, potrebujemo en seštevalnik. Velikost pomnilnika se je zmanjšala na  $P \cdot 2^{N/P}$ , kjer je  $P$  število vzporedno vezanih

pomnilnikov. Za vsako razdelitev pomnilnika se veriga seštevalnikov poveča za eno enoto, kar pomeni, da se podaljša čas izračuna delne vsote.

Tabela 1 prikazuje zapis delnih vsot koeficientov v pomnilniku.

**Tabela 1: Zapis delnih vsot vni(k) v pomnilniku**

N	naslov ROM pomnilnika	Vrednost v pomnilniku
1	0...0000	0
2	0...0001	h(0)
3	0...0010	h(1)
4	0...0011	h(0)+h(1)
5	0...0100	h(2)
⋮	⋮	⋮
2 <sup>N</sup>	1...1111	h(0)+h(1)+...h(N-1)

Najprej izračunamo delne vsote po enačbah (3) in (4), ter jih vpišemo v pomnilnik 2<sup>N</sup> lokacij.

Izhodno vrednost sita y(k) izračunamo po enačbi (5), ki opisuje izračun vseh delnih vsot koeficientov.

### Vezje aritmetične enote za sproten izračun delne vsote koeficientov

Vsote koeficientov sproti izračunavamo iz karakterističnih delnih vsot koeficientov s poljem seštevalnikov. Manj kot imamo vnaprej izračunanih karakterističnih delnih vsot koeficientov, večja je aparturna kompleksnost vezja za izračun manjkajočih delnih vsot koeficientov.

### Vezje aritmetične enote za izračun izhodne besede

Pri aparturni izvedbi digitalnega FIR sita je vrednost izhodne besede y(k) podana z enačbo,

$$y(k) = \sum_{i=1}^N h_i \cdot u(k-i) \quad (6)$$

V enačbi (6) so h<sub>i</sub> koeficienti digitalnega FIR sita, N je število vseh koeficientov in u(k) je vhodna vrednost FIR sita.

### 3 XILINX PROGRAMIRNA VEZJA

Podjetje Xilinx [8] je 1985. leta, leto dni po ustanovitvi, na trgu ponudilo prvo generacijo programirnih FPGA vezij, zasnovano na revolucionarni ideji združevanja gostote VLSI - logike z možnostjo programiranja s strani uporabnika. Ko gre za izdelke firme Xilinx, pogosto namesto FPGA- govorimo kar o LCA-vezjih, saj je to njihova zaščitna oznaka. LCA - vezja ali logične celične mreže je ime za vrste vezi, ki temeljijo na SRAM tehnologiji. Podatki za njihovo inicializacijo so običajno zapisani v zunanjem pomnilniku. Pomnilnik je lahko standardni vzporedni EPROM, elegantnejša pa je rešitev uporabe zaporednega EEPROM-a. Ob inicializaciji logike za vpis konfiguracijske vsebine, prenese podatke o delovanju LCA-vezja v vezje in ga pripravi za delovanje.

#### **Dobre lastnosti LCA-vezij so [8]:**

- preprosto načrtovanje in testiranje: Xilinx FPGA in EPLD vezja zahtevajo za načrtovanje in verifikacijo le nekaj dni, zaradi tega potrebujejo zelo kratek čas od razvoja do pojavitve na trgu; zato lahko naprava prinaša dobiček dlje časa;
- vezja za načrtovanje in verifikacijo zahtevajo kratek čas;
- velika fleksibilnost, spremembe so preproste in se lahko vedno izvedejo;
- preprogramirnost - večkratna uporaba vezij (posebej koristno v razvojni fazi);
- zanesljivost. Odločitev o primernem tipu Xilinx čipa se izvede v treh korakih:
- izbira družine: izbira se na osnovi arhitekture naprave, logične kapacitete, časovnih zahtev itn.;
- izbira serije in tipa: izbira se na osnovi števila vrat, števila vhodno-izhodnih priključkov, števila flip-flopov, RAM-bitov, CLB-celic in makro celic;
- izbira ohišja: ohišja so standardizirana in pogosto je izbrani čip možno zamenjati s čipom večje ali manjše gostote brez sprememb na ploščici; od vrste ohišja sta odvisni maksimalna hitrost in temperatura.

Poleg omenjenih izdelkov so na trgu prisotna tudi fiksno ožičena polja vrat, ki so po arhitekturi enaka FPGA-vezjem. FPGA je možno na koncu razvojne faze preprosto zamenjati s trdo ožičenim poljem logičnih gradnikov in tako zmanjšati stroške pri izdelavi serije izdelkov.

Laboratorijsko razvojni sistem je izdelan s tretjo generacijo Xilinxovih LCA-vezij družine XC4000, ki temelji na programirnem polju logičnih mrež FPGA.

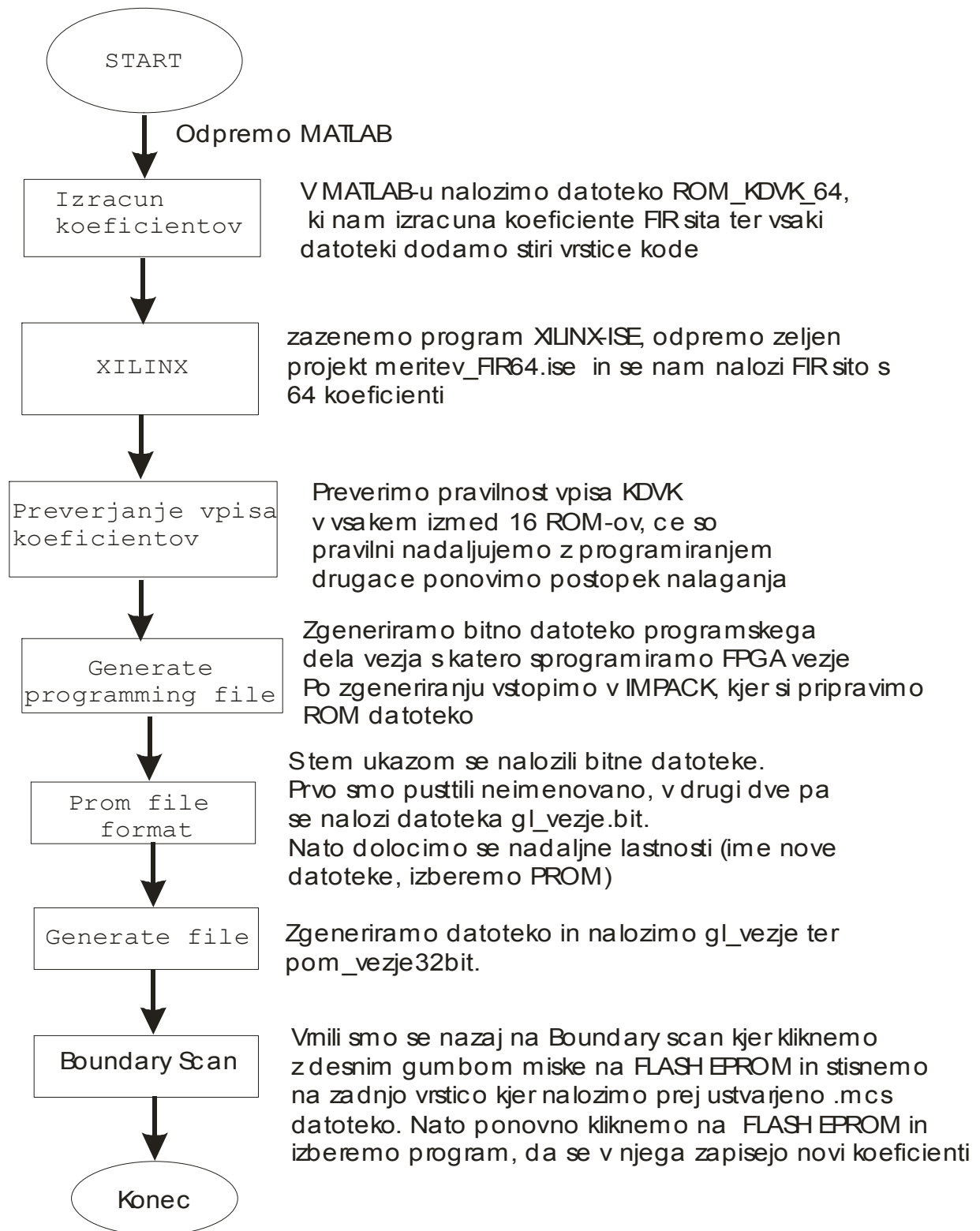
Družina vsebuje do 25.000 uporabniško dostopnih vrat in podpira frekvence ure do 50 MHz.

Trenutno se v laboratoriju za elektronske sisteme uporablja razvojna plošča z integriranim vezjem iz družine XC4000. Reševanje zahtevnega digitalnega problema s takšnim vezjem je glede na zahtevnost procesiranja in obseg vhodnih podatkov nemogoče. Omejeni smo s številom logičnih gradnikov in z zmogljivostjo programirnega vezja. Podjetje Xilinx ponuja veliko cenovnih rešitev z različnimi tipi FPGA vezij. Trenutni glavni produkt so programirna vezja iz družine Vitrex in Spartan. Vezja se med seboj razlikujejo po električnih lastnostih in po namembnosti. Ena od glavnih razlik med tema dvema družinama je cena. Vitrex integrirana vezja so namenjena za uporabo v najzahtevnejših sistemih. Razvoj te družine temelji na doseganju čim večjih hitrosti, zmogljivosti, zanesljivosti in čim večji integraciji, pri čemer cena ni omejitvev. Integrirana vezja te družine se prodajajo samo v PGA integrirani obliki.

Družina Spartan pa se orientira k temu, da ponudi uporabniku čim ugodnejši produkt, s katerim lahko rešujemo na določenih področjih primernejše probleme kot pri družini Vitrex. Iz družine Spartan izhajajo tri glavne skupine, katere nas usmerjajo k pravilni izbiri programirnega vezja. V naši razvojni plošči, s katero se ukvarjamo, sta uporabljena dva Spartan 3 integrirana vezja, serije XC3S400 [1].

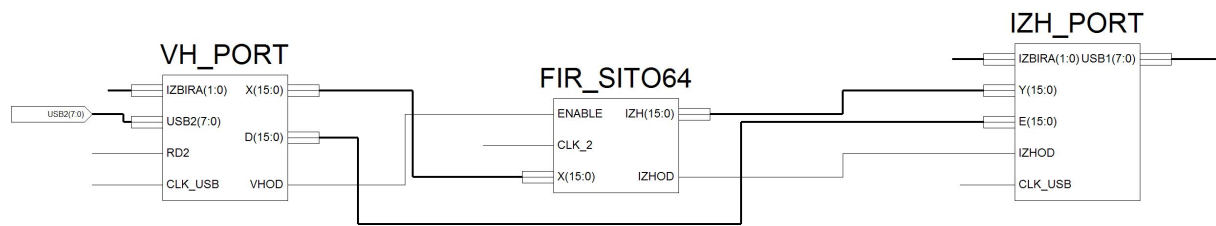
FPGA družina programirnih vezij je načrtovana za različne elektronske aplikacije. Najdemo jih v napravah, kjer poteka obdelava različnih vhodnih digitalnih signalov in kjer so izhodi lahko raznoliki prikazovalniki ali komunikacijski vmesniki.

### 3.1 Programiranje FPGA vezja s 64 koeficienti



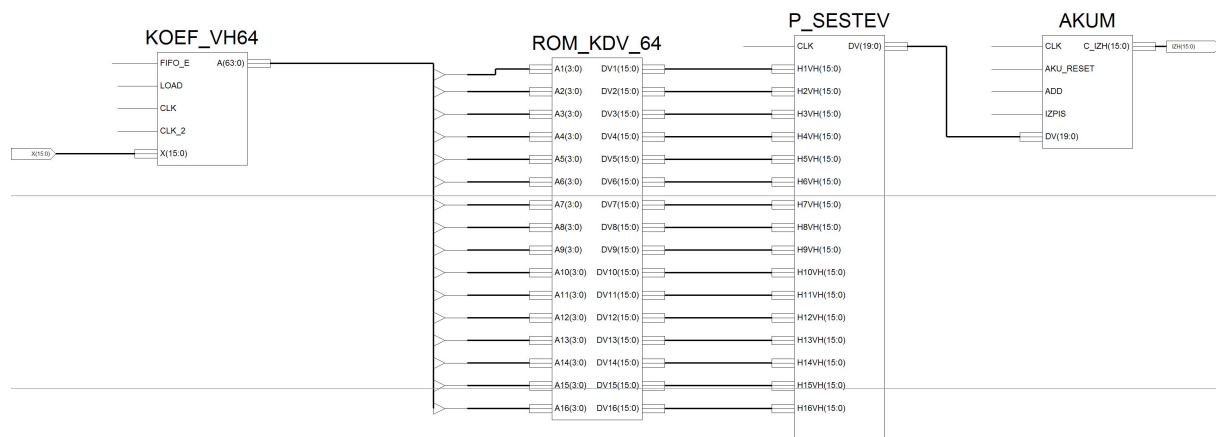
Slika 3: Postopek programiranja FPGA vezja

Koeficiente digitalnega FIR sita smo izračunali izmerili z okoljem MATLAB in s pomočjo skriptne datoteke ROM\_KDVK\_64, v kateri smo spreminjali frekvenčni in amplitudni vektor.



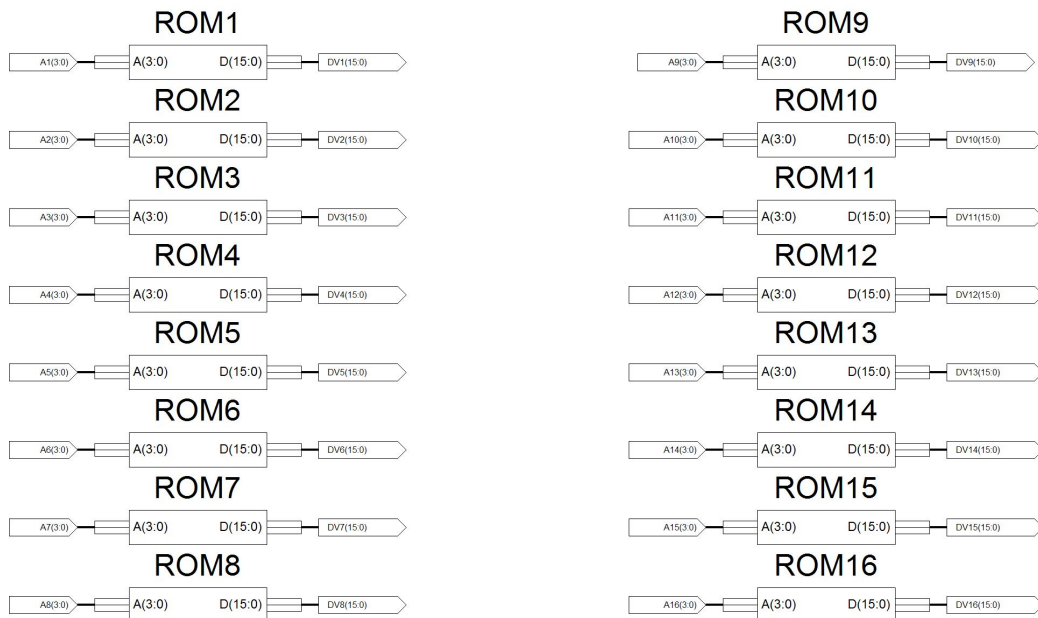
**Slika 4: Izvedba nerekurzivnega digitalnega FIR sita s 64 koeficienti**

Slika 4 prikazuje glavno vezje, ki vsebuje FIR\_SITO64.



**Slika 5: Izvedba nerekurzivnega FIR sita v porazdeljeni aritmetiki z več podstrukturami**

Slika 5 predstavlja podstrukturo digitalnega FIR sita s 64 koeficienti, ki je sestavljena iz vhodnega polja za generiranje naslovnega vektorja (KOEF\_VH64), polja pomnilnikov karakterističnih delnih vsot (ROM\_KDV\_64), koeficientov polja seštevalnikov aritmetične enote (P\_SESTEV), ki s seštevanjem delnih vsot koeficientov izračuna izhodno vrednost in akumulatorja (AKUM), ki na izhodu sešteje 16 delnih vsot in dobljeno vsoto deli z 2.



Slika 6: 16 ROM pomnilnikov za pomnjenje karakterističnih delnih vsot koeficientov

V vsakem izmed 16 ROM-ov (slika 6), ki so opisani v VHDL jeziku, so zapisane karakteristične delne vsote koeficientov v hexadecimalnem zapisu (slika 7).

```

29 entity ROM1 is
30   Port ( A : in  STD_LOGIC_VECTOR (3 downto 0);
31         D : out  STD_LOGIC_VECTOR (15 downto 0));
32 end ROM1;
33
34 architecture Behavioral of ROM1 is
35
36
37   --signal tmp : std_logic_vector(1 downto 0) := B"00";
38   type ROM_Array is array (0 to 15) of std_logic_vector(15 downto 0);
39
40   constant Content: ROM_Array := (
41     0 => x"0000",
42     1 => x"0040",
43     2 => x"0080",
44     3 => x"00C0",
45     4 => x"0100",
46     5 => x"0140",
47     6 => x"0180",
48     7 => x"01C0",
49     8 => x"0200",
50     9 => x"0240",
51    10 => x"0280",
52    11 => x"02C0",
53    12 => x"0300",
54    13 => x"0340",
55    14 => x"0380",
56    15 => x"03C0");

```

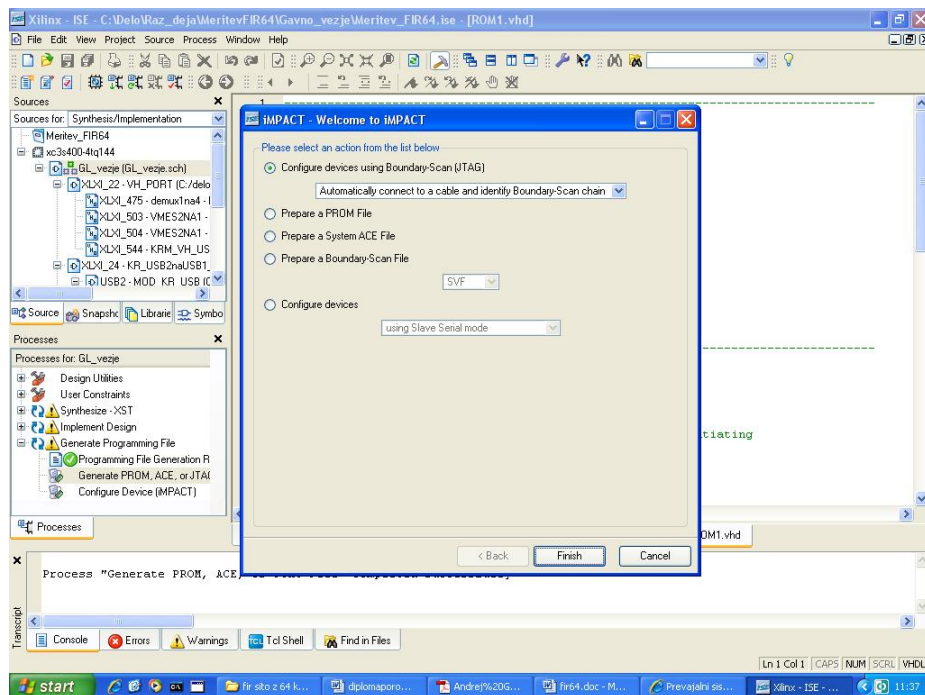
Slika 7: Podstruktura ROM pomnilnika, kjer so zapisane KDVK

## Xilinx in postopek prenosa podatkov v FPGA vezje

Za konfiguracijo podatkov v FPGA vezje smo uporabili inicializacijski kabel, s katerim smo povezali FPGA vezje preko JTAG vmesnika na računalnik. Pomembno je, da je inicializacijski kabel pravilno priključen. Rdeča žička na kablu mora biti priključena na strani Uref, tako na FPGA vezju, kot na JTAG vmesniku.

Zaženemo program XILINX – ISE, odpremo želen projekt meritev\_FIR64.ise in naloži se nam FIR sito s 64 koeficienti (slika 4).

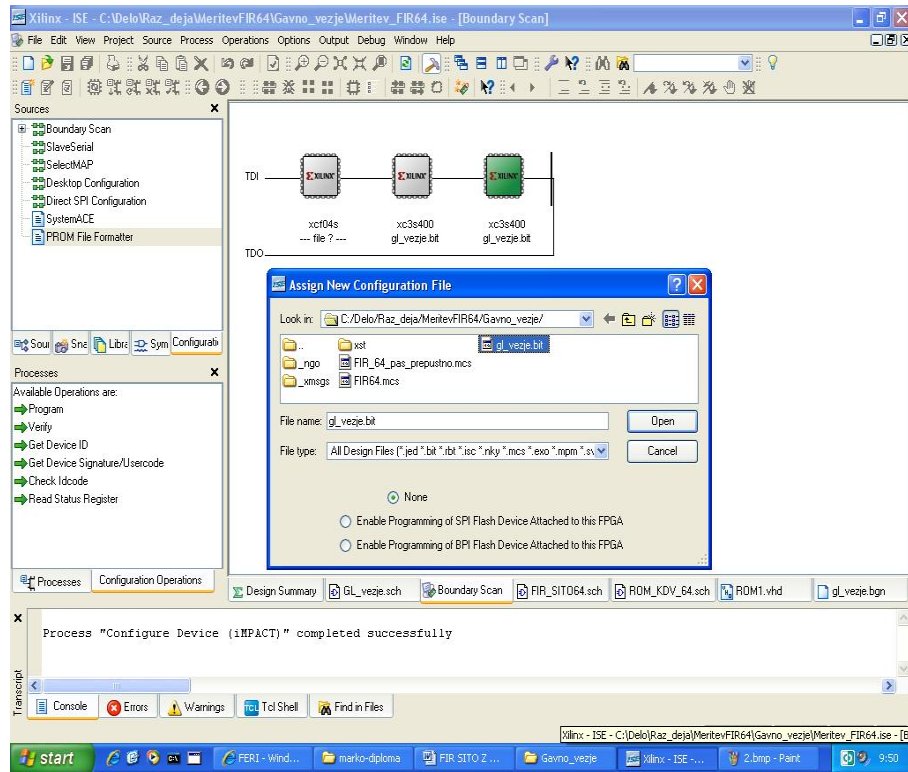
Potem, ko smo preverili koeficiente FIR sita (slika 7), če so pravilno naloženi, smo v programu Xilinx dvokliknili **Source** in izvedli proces **generate programming file** → **generate PROM**, kjer smo zgenerirali bitno datoteko programskega dela vezja in jo neposredno prenesli v FPGA vezje. Po zgeneriranju vstopimo v IMPACT, kjer si pripravimo EEPROM datoteko (slika 8).



Slika 8: IMPACT za pripravo EEPROM datoteke

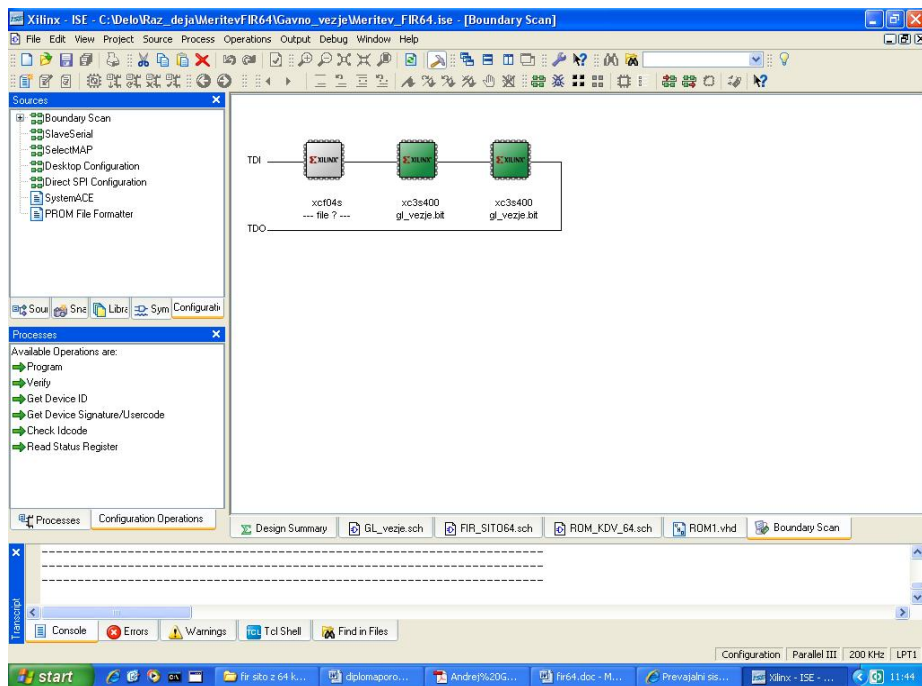
Z ukazom **PROM file formatter** (slika 9) smo naložili vsebino bitne datoteke neposredno v FPGA vezje. Najprej smo pritisnili tipko **Cancel**, ker smo prvo vezje določili šele v nadaljevanju postopka, v ostala dva programirna vezja serije xc3s400, pa smo neposredno naložili z postopkom **gl\_vezje.bit** → **open** vsebino bitne datoteke imena gl\_vezje.bit.





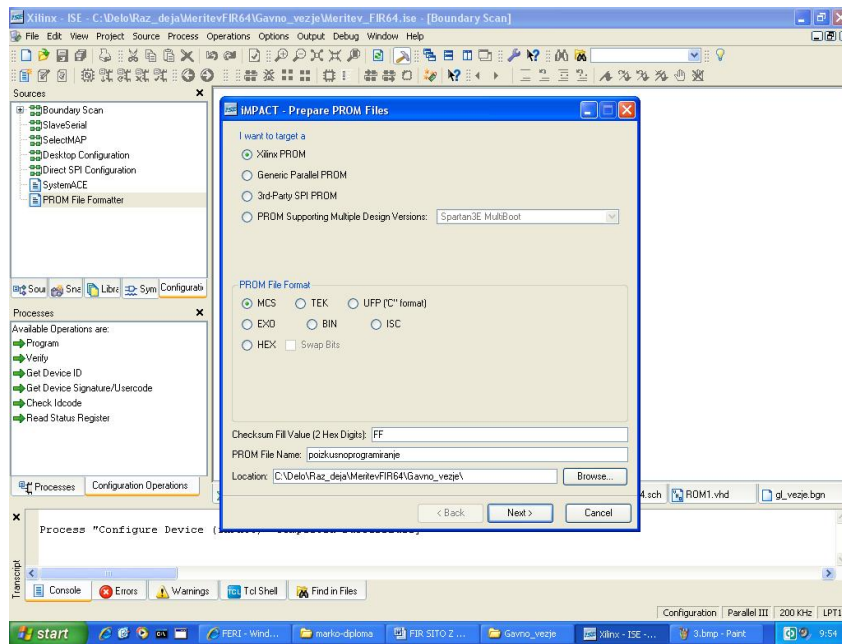
Slika 9: Nalaganje vsebine bitne datoteke imena gl\_vezje.bit v FPGA vezje

Bitna datoteka za inicializacijo FPGA vezja.



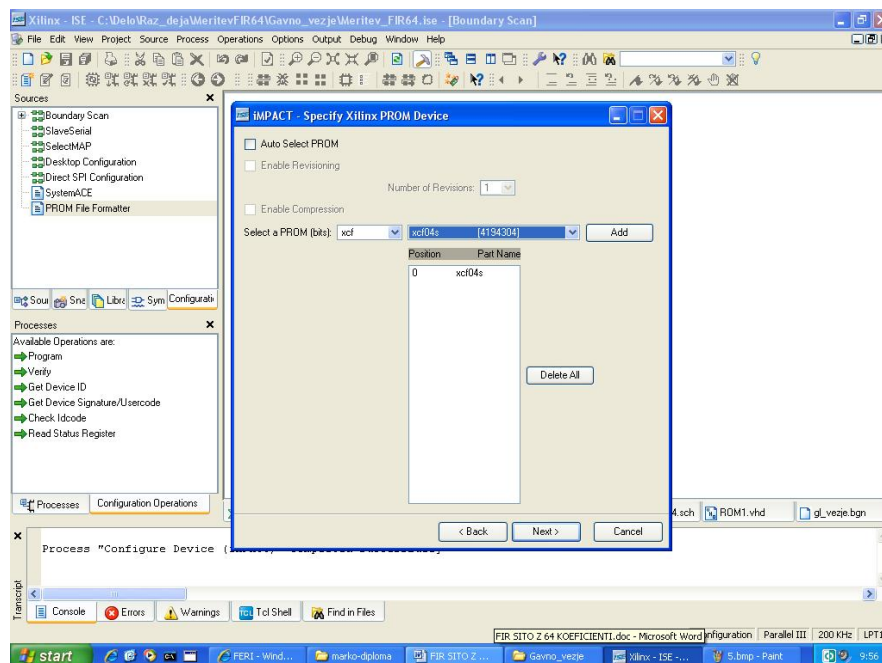
Slika 10: Naložena vsebina bitne datoteke v FPGA vezju

V FPGA vezju, ki sta označena z zelenim kvadratom (slika 10), je naložena vsebina bitne datoteke imena **gl\_vezje.bit**.



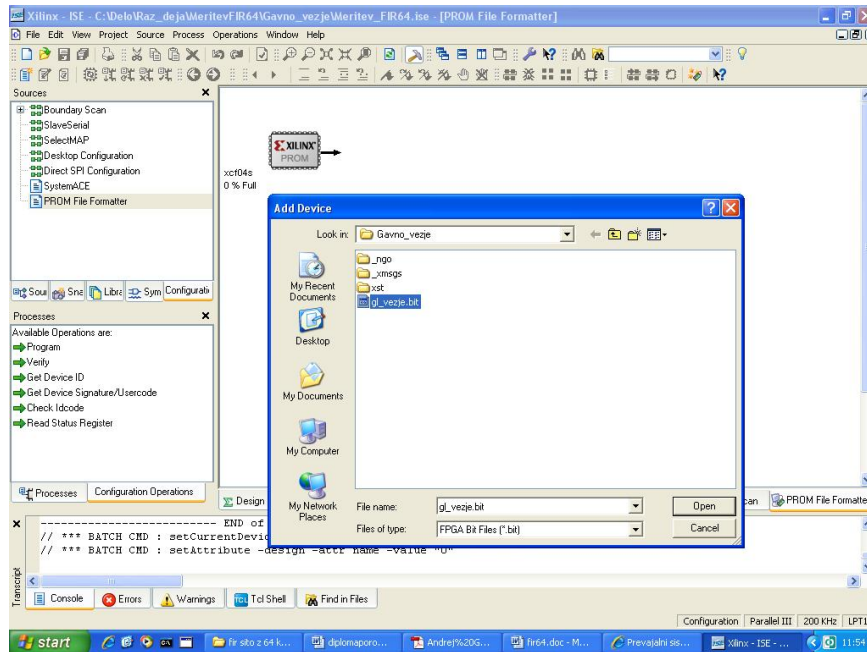
Slika 11: Imenovanje mcs datoteke

Slika 11 prikazuje kako smo imenovali mcs datoteko, ki jo naložimo v EEPROM. Izberemo **PROM file formatter** → **PROM file name**, ter izberemo vrsto EEPROMA (slika 12).

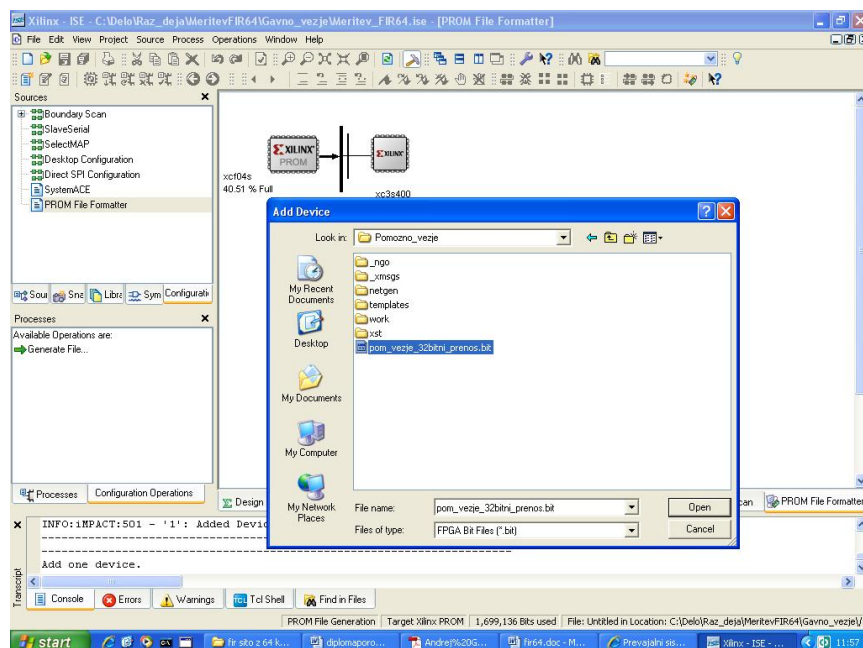


Slika 12: Izbira vrste EEPROM-a (xcf04s)

Zaključimo z nastavitvijo našega vezja, kjer smo izbrali EEPROM in ustvarili mcs datoteko.

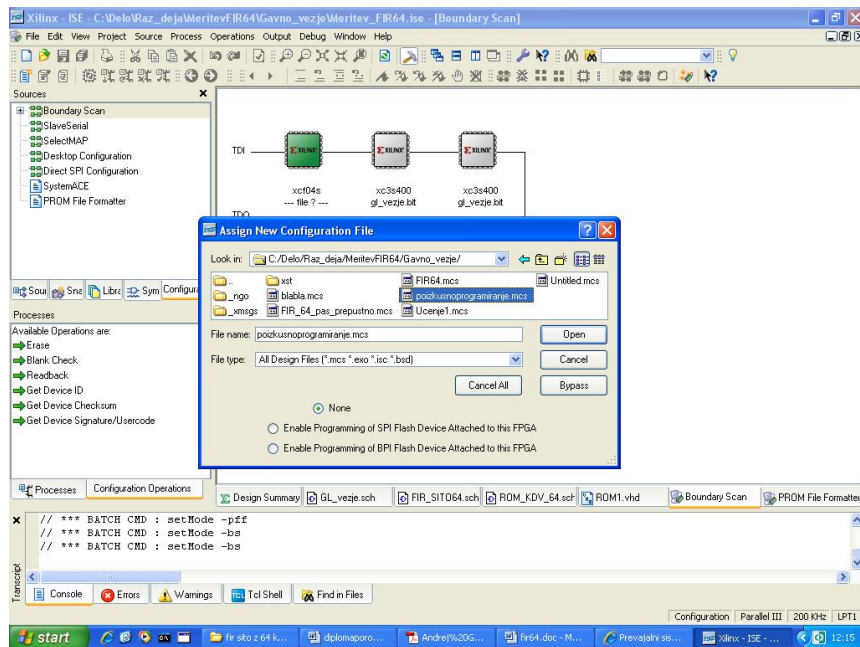


Slika 13: Pogovorno okno za dodajanje FPGA VEZJA z vsebino bitne datoteke gl\_vezje.bit



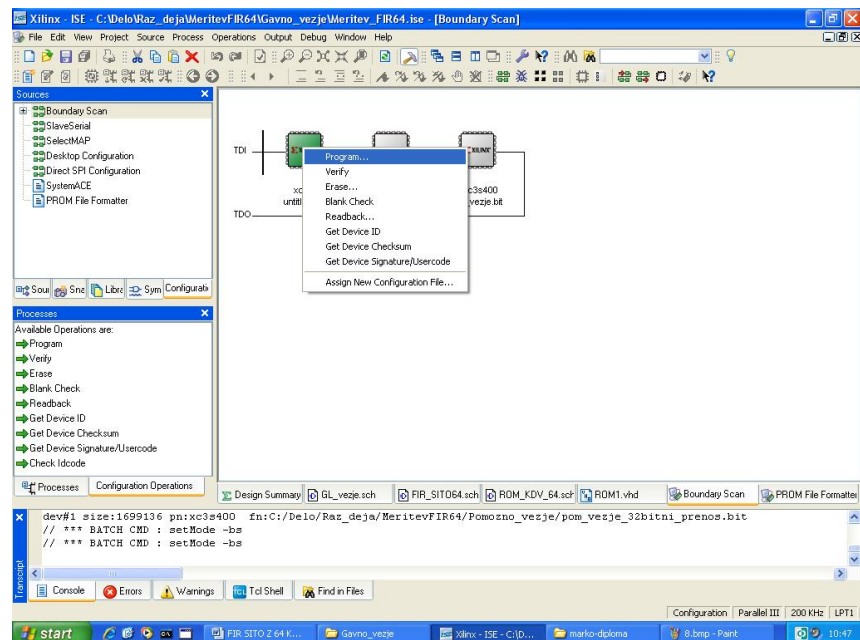
Slika 14: Pogovorno okno za dodajanje FPGA vezja z vsebino bitne datoteke pom\_vezje\_32bitni\_prenos.bit

Bitno datoteko zgeneriramo z **Generate file** in se pretvori v mcs format. Kliknemo **Boundary scan** in prenesemo mcs datoteko v EEPROM pomnilnik (slika 15).



Slika 15: Generiranje mcs datoteke

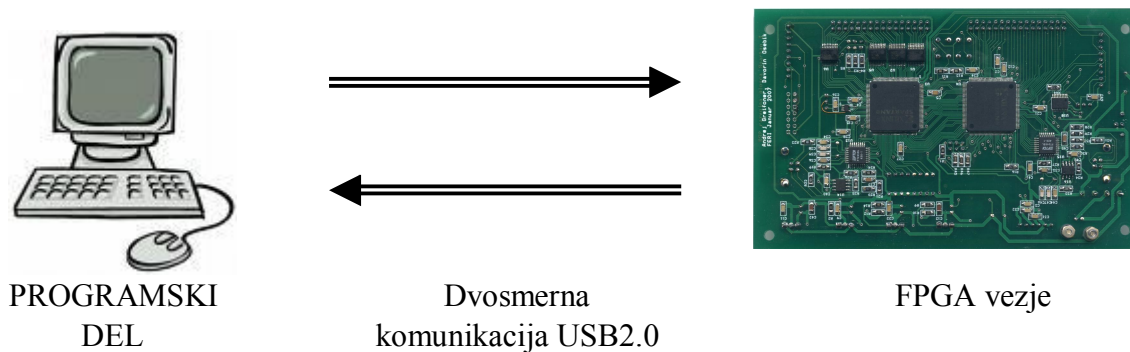
Po prenosu spet označimo z desnim gumbom na EEPROM in izberemo **program** (slika 16).



Slika 16: Programiranje

Tako smo uspešno naložili vsebino mcs datoteke v EEPROM pomnilnik. Če se pojavi napaka **PROGRAM FAILED**, postopek ponovimo (označimo z desnim gumbom na prvi kvadratale in izberemo program), dokler se nam ne izpiše **program succeeded**.

## 4 ORODJA



Slika 17: Prikaz komunikacije med programskim delom in FPGA vezjem

### 4.1 Opis razvojne enote

Razvojna plošča [1] je skrbno načrtovan sistem namenjen za raziskovalne namene. Vsebovati mora vse elemente, s pomočjo katerih lahko obdelujemo podatke, pridobljene z vhodnih naprav. Zato je zelo pomembna kompatibilnost z drugimi napravami. V glavnem je takšna plošča zgrajena iz osrednjega programirnega vezja in ostalih perifernih naprav. Te naprave so lahko različni prikazovalniki ali vmesniki.

Programirni vezji sta med seboj povezani z 32 bitnim vodilom. S tem je omogočen razvoj aplikacij, ki so prostorsko porazdeljene. Dva USB uporabniška vmesnika služita za sočasno sprejemanje in oddajanje podatkov. Namenjena sta za komunikacijo med razvojno ploščo in programskim okoljem MATLAB.

Napajalni del razvojne plošče je načrtovan za 9 V izmenično napajanje. Sestavljen je iz AC priključka, diodnega usmernika, gladilnega in blokirnega kondenzatorja, treh linearnih nastavljivih regulatorjev LM317T in enega linearnega 5 V regulatorja LM7805.

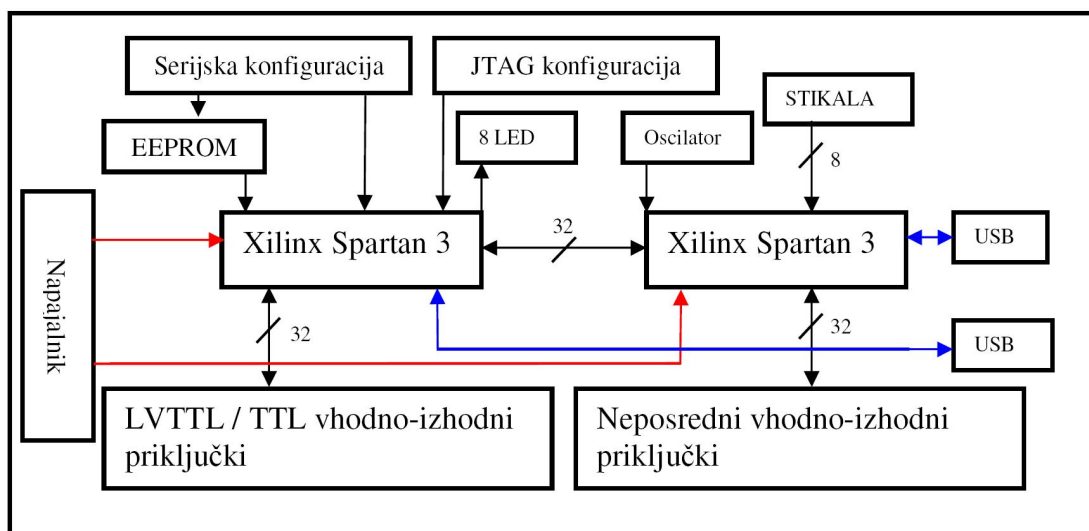
Razvojno ploščo velikokrat uporabljamo skupaj s starejšimi napravami, zato imamo vmesnik, ki zniža višje napetostne nivoje starejših naprav in nam tako omogoča priklop teh naprav. Vmesnik SN74LVC245A je narejen na osnovi nizkonapetostne CMOS tehnologije. Omogoča asinhrono preklapljanje med dvema osem-bitnima vodiloma.

Zasnova USB vmesnika na razvojni plošči temelji na tem, da ponuja uporabniški dostop do vhodno-izhodnih priključkov. S pomočjo ustrezne programske opreme v okolju MATLAB lahko med tekom procesa v programirnem vezju sprejemamo ali pošiljamo podatke. Takšen vmesnik

nam omogoča enostavno izvedbo meritev razvitih aplikacij v FPGA strukturi. Dva takšna vmesnika omogočata sočasno sprejemanja in pošiljanja podatkov v FPGA strukturo. Tako lahko na takšno ploščo priklopimo dve različni, med seboj neodvisni USB napravi.

Vmesnik sestavlja integrirano vezje FT245BM neposredno priklopljeno na dvojno namenske vhodno-izhodne priključke programirnega vezja. Prenos podatkov poteka vzporedno z osem-bitno dolgo besedo. Poleg priključkov za prenos podatkov imamo še štiri kontrolne signale.

Konfiguracijske podatke shranjujemo na zunanje pomnilne enote zaradi trajnega shranjevanja konfiguracijskih podatkov. Tako vedno ostanejo konfiguracijski podatki, tudi če pride do izgube napajanja. Ob vklopu napajanja pa se ti samodejno prenesejo v programirno vezje. Za shranjevanje konfiguracijskih podatkov posameznega Spartan 3 programirnega vezja potrebujemo 2 Mb velik pomnilnik, zato je tu uporabljen 4 Mb Xilinx XCF04S EEPROM.



Slika 18: Zasnova razvojne plošče; Vir[1]

## 4.2 Programirno FPGA vezje Spartan 3 serije XC3S400

FPGA družina programirnih vezij [1] je načrtovana za različne elektronske aplikacije. Najdemo jih v napravah, kjer poteka obdelava različnih vhodnih digitalnih signalov in kjer so izhodi lahko raznoliki prikazovalniki ali komunikacijski vmesniki.

Iz skupine Spartan 3 izhaja osem serij integriranih vezij. Razlikujejo se po notranji strukturi, po številu logičnih gradnikov, po številu vhodno - izhodnih priključkov in predvsem po tipu ohišja.

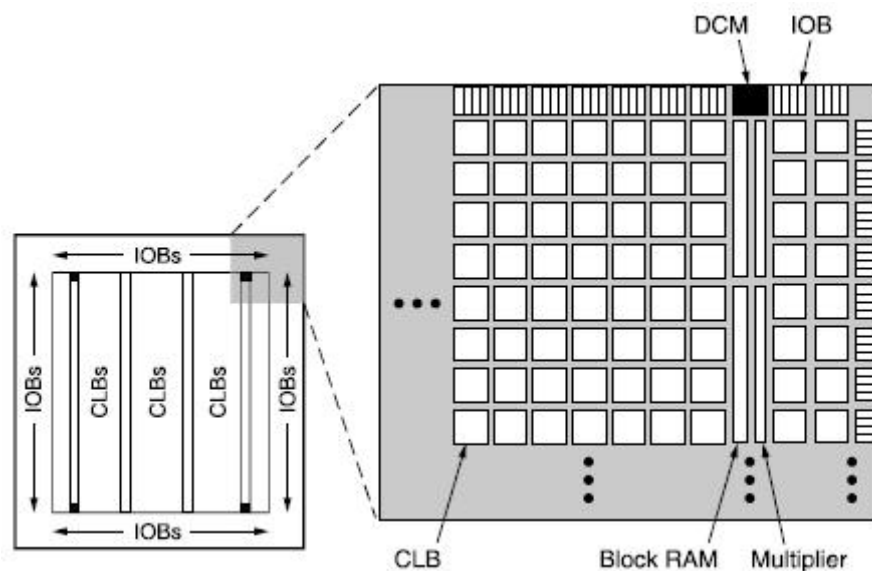
V našem diplomskem delu se ukvarjamo z razvojno ploščo, z FPGA vezji serije XC3S400 iz družine Spartan 3.

Tabela 2: Lastnosti programirnega vezja Spartan 3 serije XC3S400

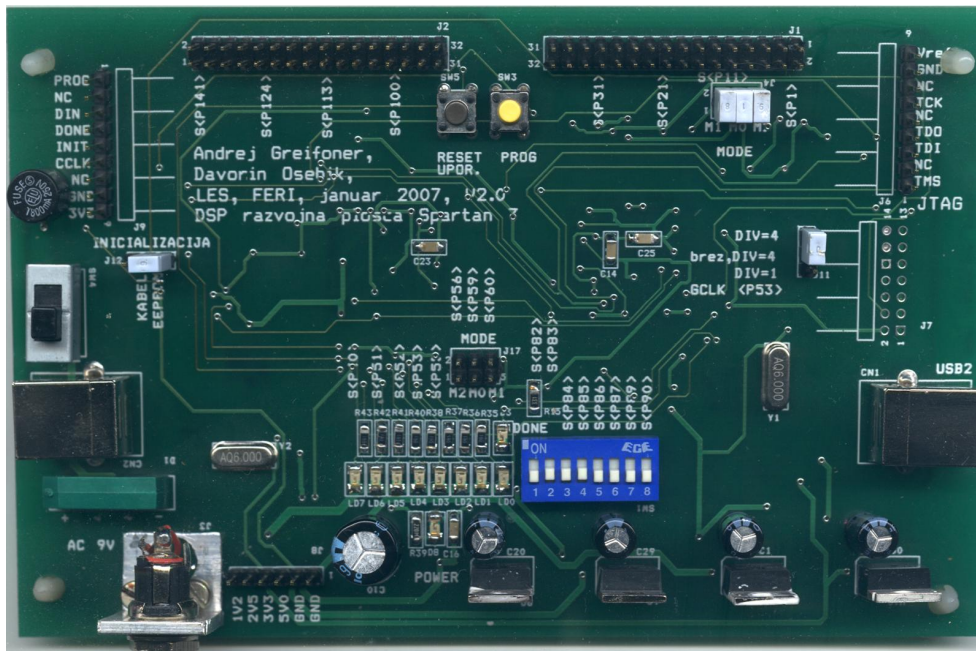
Device	System Gates	Equivalent Logic cells	CLB Array			Distributed RAM bits	Block RAM bits	Dedicated Multipliers	DCM	Maximum User I/O	Maximum Differential I/O Pairs
			Rows	Columns	Total CLBs						
XC3S400	400K	8,064	32	28	896	56K	288K	16	4	264	116

Pomen besed:

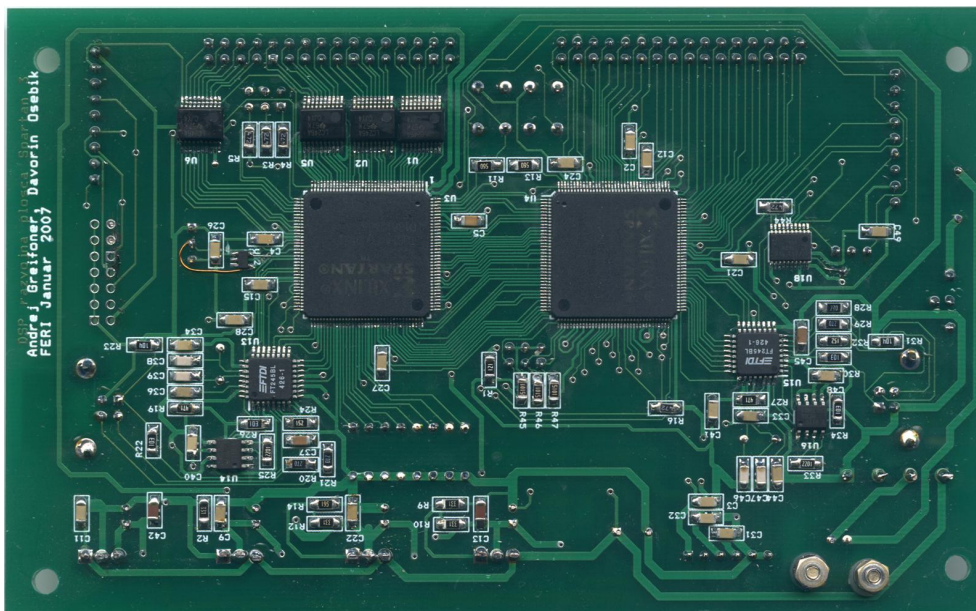
<b>Device</b>	Serija
<b>System Gates</b>	Število sistemskih vrat
<b>Equivalent Logic cells</b>	Število logičnih celic
<b>CLB Array</b>	Število konfiguracijskih logičnih blokov
<b>Distributed RAM</b>	Velikost RAM pomnilnika v čipu
<b>Block RAM</b>	Velikost RAM pomnilnika v blokih
<b>Dedicated Multipliers</b>	Število množilnikov
<b>DCM</b>	Število upraviteljev z vhodno uro
<b>Maximum User I/O</b>	Število uporabniških vhodno-izhodnih priključkov
<b>Maximum Differential I/O Pairs</b>	Največje število diferenčnih vhodno-izhodnih priključkov



Slika 19: Notranja zgradba Spartan 3 programirnega vezja; Vir[1]



Slika 20: Razvojna plošča - pogled zgoraj; Vir [1]

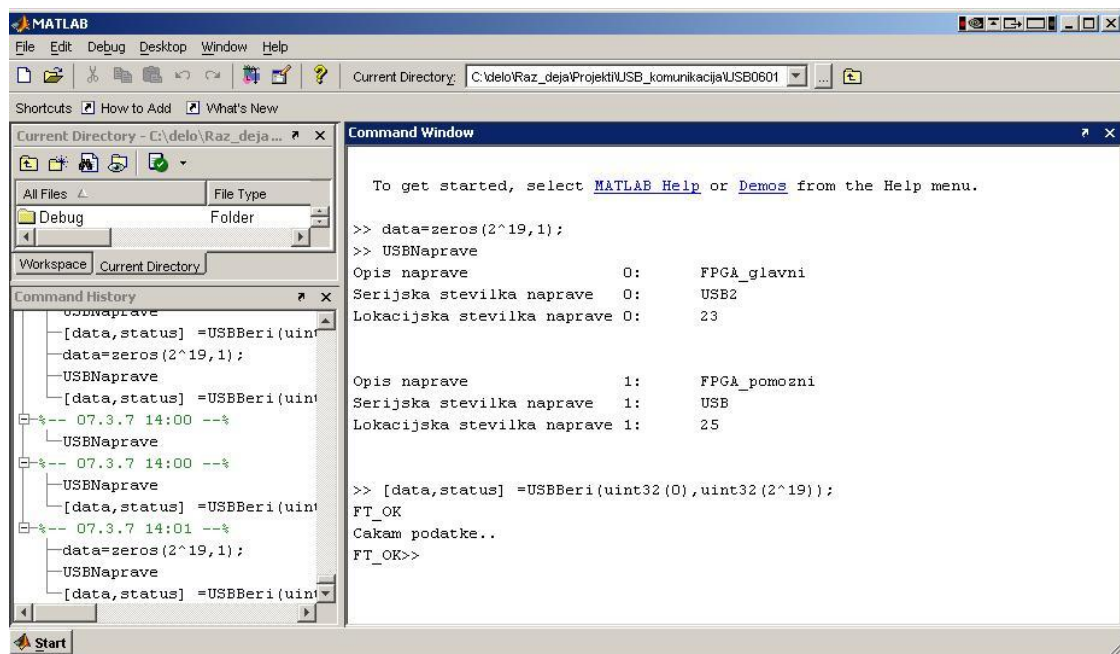


Slika 21: Razvojna plošča – pogled spodaj; Vir [1]



### 4.3 Razvojno okolje MATLAB in povezava z FPGA vezjem

Upravljanje naprav s programirnimi vezji poteka neposredno iz programskega okolja MATLAB preko dveh USB priključkov. Oba USB priključka omogočata sočasen prenos podatkov med programirnimi vezji XC3S400 in okoljem MATLAB hitrosti 1Mbyte/s [6].



Slika 22: Povezava preko dveh USB priključkov; Vir [6]

### 4.4 Serijski EEPROM za shranjevanje konfiguracijskih podatkov

Konfiguracijske podatke [1] shranjujemo na zunanje pomnilne enote zaradi trajnega shranjevanja konfiguracijskih podatkov. Tako vedno ostanejo, tudi če pride do izgube napajanja. Ob vklopu napajanja pa se samodejno ti prenesejo v programirno vezje. Za shranjevanje konfiguracijskih podatkov posameznega Spartan 3 programirnega vezja potrebujemo 2 Mb velik pomnilnik, zato bomo uporabili 4 Mb Xilinx XCF04S EEPROM.

Vpis podatkov poteka preko inicializacijskega kabla Xilinx Parallel Cabel IV. Z JTAG vmesnikom lahko konfiguriramo programirno vezje, ali pa vpisujemo podatke v pomnilnik. Pomnilnik lahko uporabimo v vezavi glavnega serijskega načina, kar pomeni, da bo prenos podatkov prejemalo programirno vezje preko DIN serijskega vhoda. Napajanje pomnilnika

moramo izvesti z enosmerno napetostjo 3,3 V, čeprav je delovanje zagotovljeno tudi pri nižjih napetostih.

## 4.5 Inicializacija razvojne enote

Za inicializacijo [1] smo uporabili inicializacijski kabel Xilinx pararel Cable IV, s pomočjo katerega lahko konfiguriramo programirna vezja FPGA, CPLD ali pa vpisujemo podatke v EEPROM pomnilnike. Podpira IEEE 1284 protokol in je popolnoma združljiv s programsko opremo iMPACT firme Xilinx. Prav tako je združljiv s protokolom IEEE 1149.1. Omogoča priklop naprav preko JTAG vodila in na Xilinx programirna vezja v pomožnem serijskem načinu. Priklop naprav je mogoč preko ploščatega (ribbon) kabla. Napajanje je izvedeno s pomočjo odcepnika na priključku miške ali tipkovnice osebnega računalnika. Kabel je spojen z osebnim računalnikom preko LTP tiskalniških vrat, podprt pa je z operacijskimi sistemi Microsoft Windows XP in Linux. Avtomatsko zaznavanje referenčne napetosti priključene naprave omogoča komuniciranje z različnimi napetostnimi standardi.



**Slika 23: Inicializacijski kabel; Vir [1]**

Paziti moramo da pravilno priključimo kabel. Na kablju mora biti rdeča žička priključena na Vref, na FPGA plošči pa mora biti rdeča žička tudi na Vref.

## 5 SIGNALI

Pogosto nas pri digitalnih sistemih zanimajo odzivi za različno zahtevane oblike frekvenčnih karakteristik, pri različnih vrstah vhodnih signalov (sinusni signal, stopnični signal, belošumni signal, enotin impulz), ter na podlagi tega dobljeno odstopanje. Za kvalitetno izvedbo meritve moramo imeti na razpolago kvalitetne testirane signale, ki jih dobimo s pomočjo ustreznega generatorja.

Za nas je predvsem zanimivo generiranje belošumnega signala. Predvsem zato, ker je najbolj primeren za testiranje raznih sistemov. S pomočjo belošumnega vhodnega signala, bomo poiskali odziv FIR sita za različne frekvenčne karakteristike.

### 5.1 Kvantizacija signalov

Tabela 3 nam predstavlja primer za predznačena [7], 4 bitna števila, na intervalu od -1 do 1. Po istem postopku lahko predstavimo 16 bitna števila, ki smo jih uporabili v našem primeru.

**Tabela 3:** Zapis 4 bitnega števila v različnih št. sistemih

Realna kvantizirana števila	Cela števila	Binarno predznačena števila	Hexadecimalna števila
7/8	7	0111	7
6/8	6	0110	6
5/8	5	0101	5
4/8	4	0100	4
3/8	3	0011	3
2/8	2	0010	2
1/8	1	0001	1
0	0	0000	0
-1/8	-1	1111	F
-2/8	-2	1110	E
-3/8	-3	1101	D
-4/8	-4	1100	C
-5/8	-5	1011	B
-6/8	-6	1010	A
-7/8	-7	1001	9
-8/8	-8	1000	8

V našem primeru uporabljamo 16-bitna predznačena števila. Ta števila imajo predznak in lahko zavzamejo pozitivne ali negativne vrednosti. In sicer najvišja pozitivna vrednost je 7FFF, najnižja negativna vrednost pa 8000.

Signali (sinusni signal, stopnični signal, belošumni signal, enotin impulz-dirak) imajo najvišjo vrednost 7FFF, ker je največja vrednost 16 bitnega kvantiziranega števila.

Tako dobimo:  $2^{16} = 65536$  (7)

Ker je to predznačeno število, ga moramo deliti z dva (  $65536/2$  ) in dobimo rezultat 32768. Temu rezultatu moramo odšteti 1, ker moramo pri pozitivni smeri upoštevati vrednost 0. Tako dobimo rezultat 32767.

Ta rezultat pretvorimo v hexadecimalni zapis in dobimo vrednost 7FFF.

$$7FFF = 32767$$

$$8000 = -32768 = -2^{15}$$

Najvišja negativna vrednost našega predznačenega števila pa je hexadecimalno število FFFF, ki predstavlja realno število -1.

Primer pretvorbe iz dvojiškega zapisa števila v desetiški zapis:

Dvojiško število: 0111111111111111

$$\text{Obtežitev: } 2^{15} 2^{14} 2^{13} \dots 2^2 2^1 2^0 \quad (8)$$

$$0111111111111111 = 2^{14} + 2^{13} + 2^{12} + \dots + 2^1 + 2^0 \quad (9)$$

$$= 16384 + 8192 + 4096 + \dots + 2 + 1 = 32768 \quad (10)$$

Primer pretvorbe iz desetiškega zapisa števila v dvojiški zapis:

Desetiško število: 32768

$$32768 = 16384 + 8192 + 4096 + \dots + 2 + 1 = 2^{14} + 2^{13} + 2^{12} + \dots + 2^1 + 2^0 = 0111111111111111 \quad (11)$$

Dvojiško število: 0111111111111111

Primer pretvorbe iz dvojiškega zapisa števila v šestnajstiški zapis:

Dvojiško število: 1000000000000000

$$\underbrace{1}_{8} \underbrace{00000000}_{0} \underbrace{00000000}_{0} \underbrace{00000000}_{0} \quad (12)$$

Šestnajstiško število: 8000

Primer pretvorbe iz šestnajstiškega zapisa števila v dvojiški zapis:

Šestnajstiško število: 7FFF

$$\overset{7}{\underbrace{0}}\overset{F}{\underbrace{1111}}\overset{F}{\underbrace{1111}}\overset{F}{\underbrace{1111}}\overset{F}{\underbrace{1111}} \quad (13)$$

Dvojiško število: 0111111111111111

Primer pretvorbe iz šestnajstiškega zapisa števila v desetiški zapis

Šestnajstiško število: 7FFF

$$0111111111111111 = 2^{14} + 2^{13} + 2^{12} + \dots + 2^1 + 2^0 = 16384 + 8192 + 4096 + \dots + 2 + 1 = 32768 \quad (14)$$

Desetiško število: 32768

## 5.2 Enotin impulz ali DIRAK

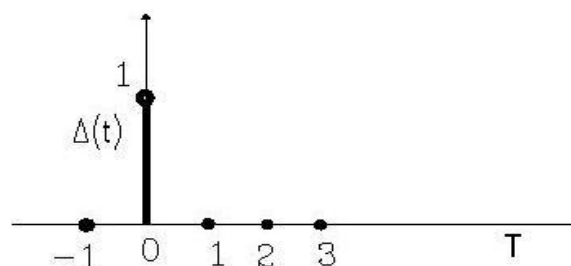
$$\lim_{h \rightarrow 0} \int_{-h}^h \frac{1}{2h} dt = \lim_{h \rightarrow 0} \left\{ 2h \cdot \frac{1}{2h} \right\} = \begin{cases} 1, & \text{pri } t = 0 \\ 0, & \text{sicer} \end{cases} \quad (15)$$

V enačbi (15) vidimo, da je rezultat limitnega postopka "pulz" [4] s širino nič (zato ga imenujemo impulz) in neskončno višino, medtem ko je ploščina enaka 1. Funkcijo, ki opisuje ta pulz, imenujemo Diracova delta funkcija ali na kratko Diracova funkcija oziroma Diracov impulz [4]. Označujemo ga s  $\delta(t)$ .

Lahko je spoznati, da ima  $\delta(t)$  naslednjo lastnost:

$$\int_{-\infty}^{\infty} x(t) \delta(t) dt = x(0)$$

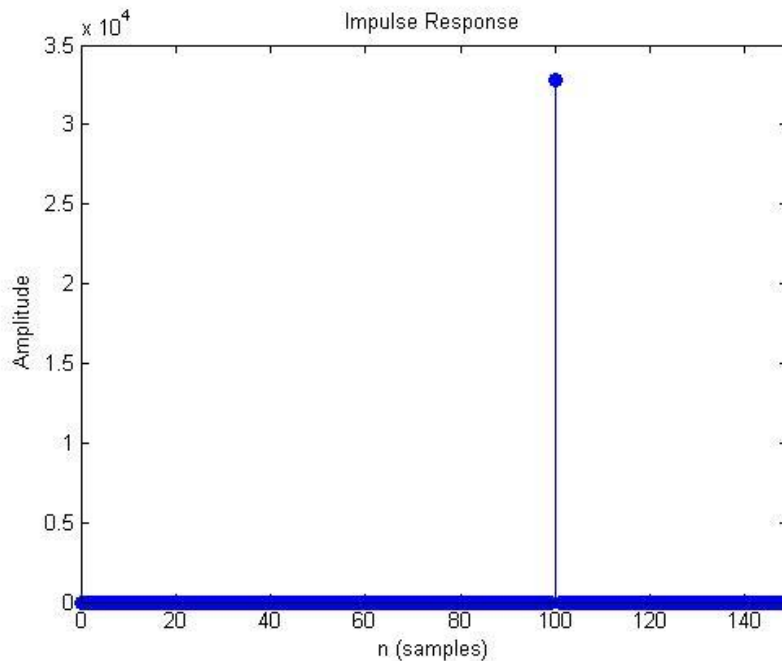
kjer je  $x(t)$  omejena regularna funkcija s signalno osjo  $T \in \mathbb{R}$  in zveznostjo v trenutku  $t = 0$ . Zaradi tega, ker integral (z mejama  $-\infty, \infty$ ) produkta Diracove funkcije s to funkcijo ohrani samo vrednost pri  $t = 0$ , imenujemo Diracovo funkcijo tudi enotin impulz.



Slika 24: Diracov impulz; Vir [4]

## 5.2.1 Predstavitev enotinega impulza v MATLABU

```
>> load dirak_usb8;
>> [y e]=usb8naYE(dirak_usb8);
>> impz (y(1:150));
```



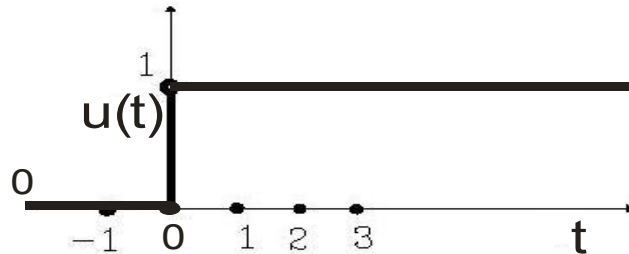
Slika 25: Premaknjeni enotin impulz za 100 otipkov

V našem primeru smo morali za izmerjeni odziv FIR sita na enotin impulz zakasniti vhodni enotin impulz za 100 otipkov, da smo se izognili prehodnim pojavom FIR sita z 64 koeficienti. Prehodni pojav je tako izrazil zaradi tega, ker nismo RESETIRALI tipke na DSP ploščici. Ker imamo FIR sito z 64 koeficienti, je prvih 64 otipkov napačnih oz. posledica predhodne meritve, katerih vrednosti so shranjene v registrih FIR sita. Maksimalno vrednost smo dosegli do 7FFF, ker je to največja vrednost 16 bitnega kvantiziranega števila.

## 5.3 Stopnični signal

Stopnični signal [4] je definiran z enačbo 10:

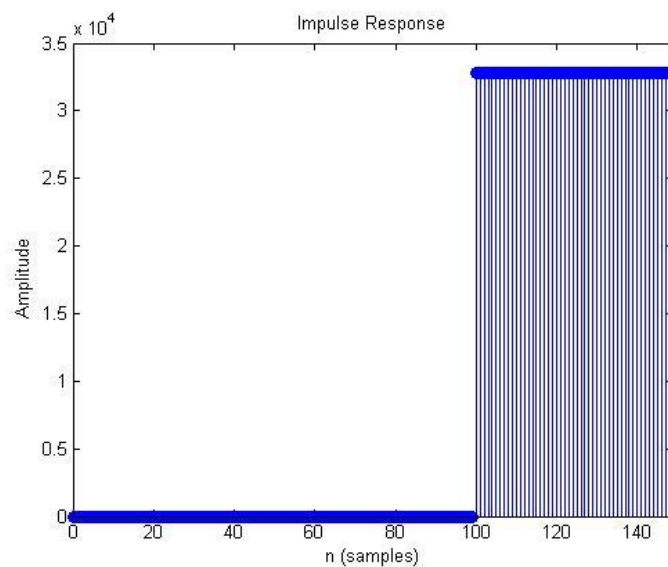
$$u(t) = \begin{cases} 1, & t \geq 0 \\ 0, & t < 0 \end{cases} \quad t \in T = \mathbb{R} \quad (10)$$



Slika 26: Enotina stopnica; Vir [4]

### 5.3.1 Predstavitev stopničnega signala v MATLABU

```
>> load stopnica_usb8;
>> [y e]=usb8naYE(stopnica_usb8);
>> impz (y(1:150));
```



Slika 27: Premaknjeni stopnični signal za 100 otipkov (impz)

Tako kot pri enotinem impulzu, smo za meritve v MATLABU zakasnili tudi stopnični signal za 100 otipkov, da smo se izognili prehodnim pojavom FIR sita z 64 koeficienti. Prehodni pojav je izrazit zaradi tega, ker nismo stisnili RESET tipke na DSP ploščici. Ker imamo FIR sito s 64

koeficienti, je prvih 64 otipkov napačnih oz. posledica predhodne meritve, katerih vrednosti so shranjene v registrih FIR sita.

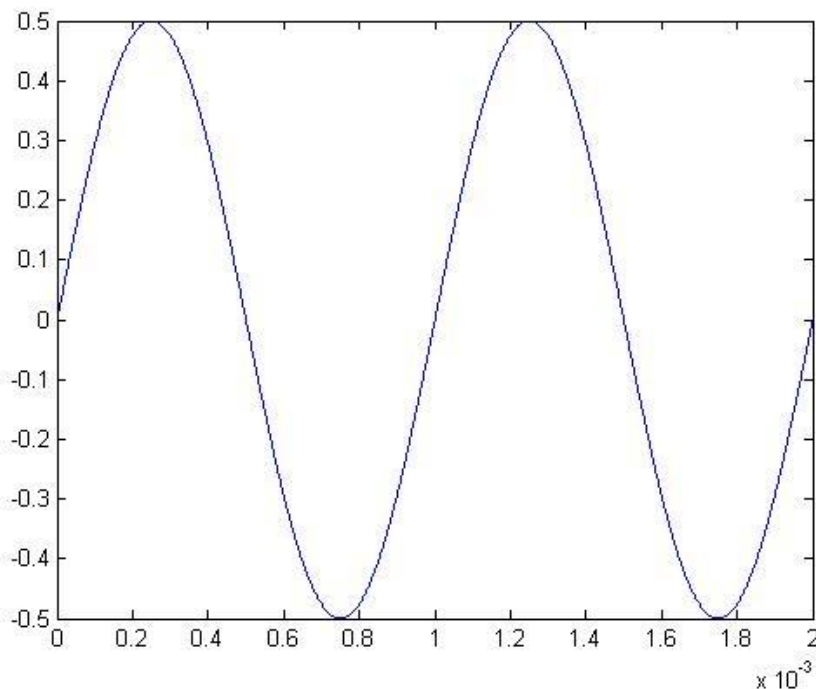
## 5.4 Signal sinusne oblike

Eden od osnovnih signalov je tudi signal sinusne oblike [5]. Splošni realni signal sinusnega poteka opisujejo parametri amplitude ( $U_{\max}$ ), frekvence( $\omega t$ ) in faze( $\varphi$ ) v enačbi 9.

$$u(t) = u_{\max} \cdot \sin(\omega t) + \varphi \quad (9)$$

### 5.4.1 Predstavitev sinusnega signala v MATLABU

```
>> y=0.5;                ;definiranje amplitude
>> x = 0:10^(-6):0.06554; ;definiramo otipke po koraku 1us od 0 do 65,54ms
>> u=y*(sin(2*pi*1000*x)); ;frekvenca 1000Hz
>> plot(x(1:2000),u(1:2000));
```



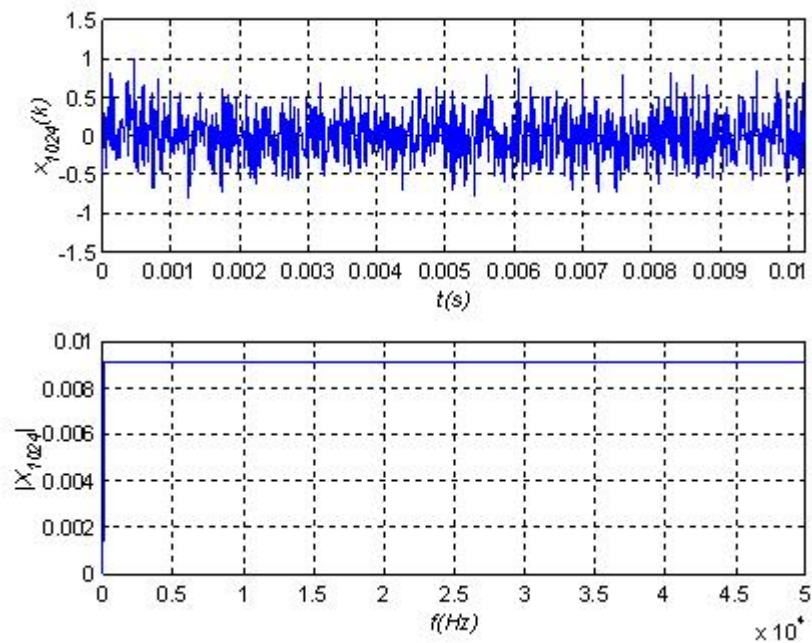
Slika 28: Sinusni signal v MATLABU

Z normiranjem maksimalne amplitude sinusnega signala na 0,5 se izognemo prekoračitvi aritmetike znotraj FIR sita.

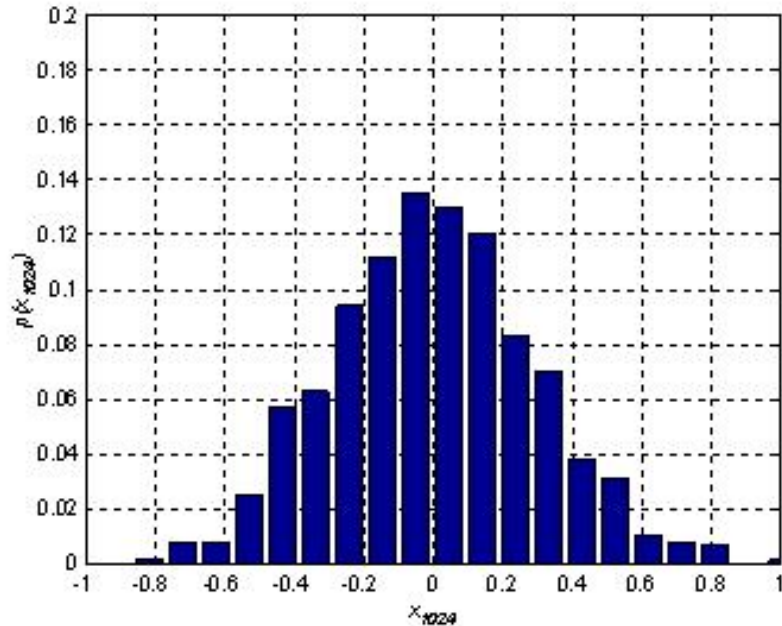


## 5.5 Belošumni signal

Belošumni signal [2] je signal, ki ima enak spekter moči v vsem frekvenčnem spektru. Generatorje belega šuma uporabljamo v laboratorijih pri testiranju vezij (za simulacijo naravnega šuma).



Slika 29: Časovni potek Gaussovega signala belega šuma s 1024 vzorci v periodi in njegova gostota spektra moči pri  $f_v=100\text{kHz}$



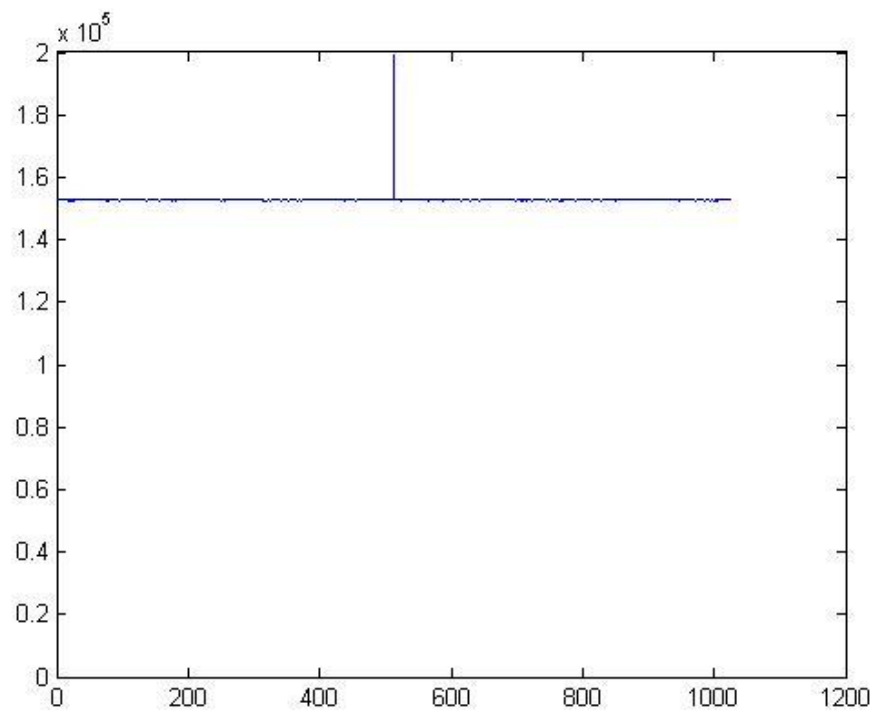
Slika 30: Porazdelitev amplitud signala belega šuma s 1024 vzorci v periodi

Belošumni signal je najbolj primeren za testiranje raznih sistemov. Uporablja se za opis signala v nekem sistemu ali pri meritvi. Idealen belošumni signal naj bi vseboval popolnoma naključne vrednosti, srednjo vrednost enako nič, ter konstanten amplitudni in močnostni spekter. Če testiramo belošumni signal kot stacionarni naključni signal  $v(n)$ , bomo s simbolom  $n$  označevali  $n$ -to vrednost belošumnega signala. Naključni signal  $v(n)$  imenujemo belošumni signal, če ima uniformni spekter v precej širokem pasu frekvenc. Po analogiji z belo svetlobo, katera ima v vidnem pasu spektra praktično uniformirano in konstantno spektralno gostoto moči. Belošumni signal je signal, katerega avtokorelacijska funkcija je enaka enotinem impulzu pomnoženim s konstanto in je podan z enačbo (12):

$$R_v(n) = \sigma_v^2 \cdot \delta(n) \quad (12)$$

### 5.5.1 Predstavitev belošumnega signala v MATLABU

```
>> load sum1024usb8.mat;
>> [sum1 sum2]=usb8naYE(sum1024usb8(1:2^15));
>> plot(abs(fft(sum1(1:1024),1024)));
```



Slika 31: Predstavitev spektra moči belošumnega signala

Špica na sliki se pojavi zaradi računanja.

## 6 MERITEV

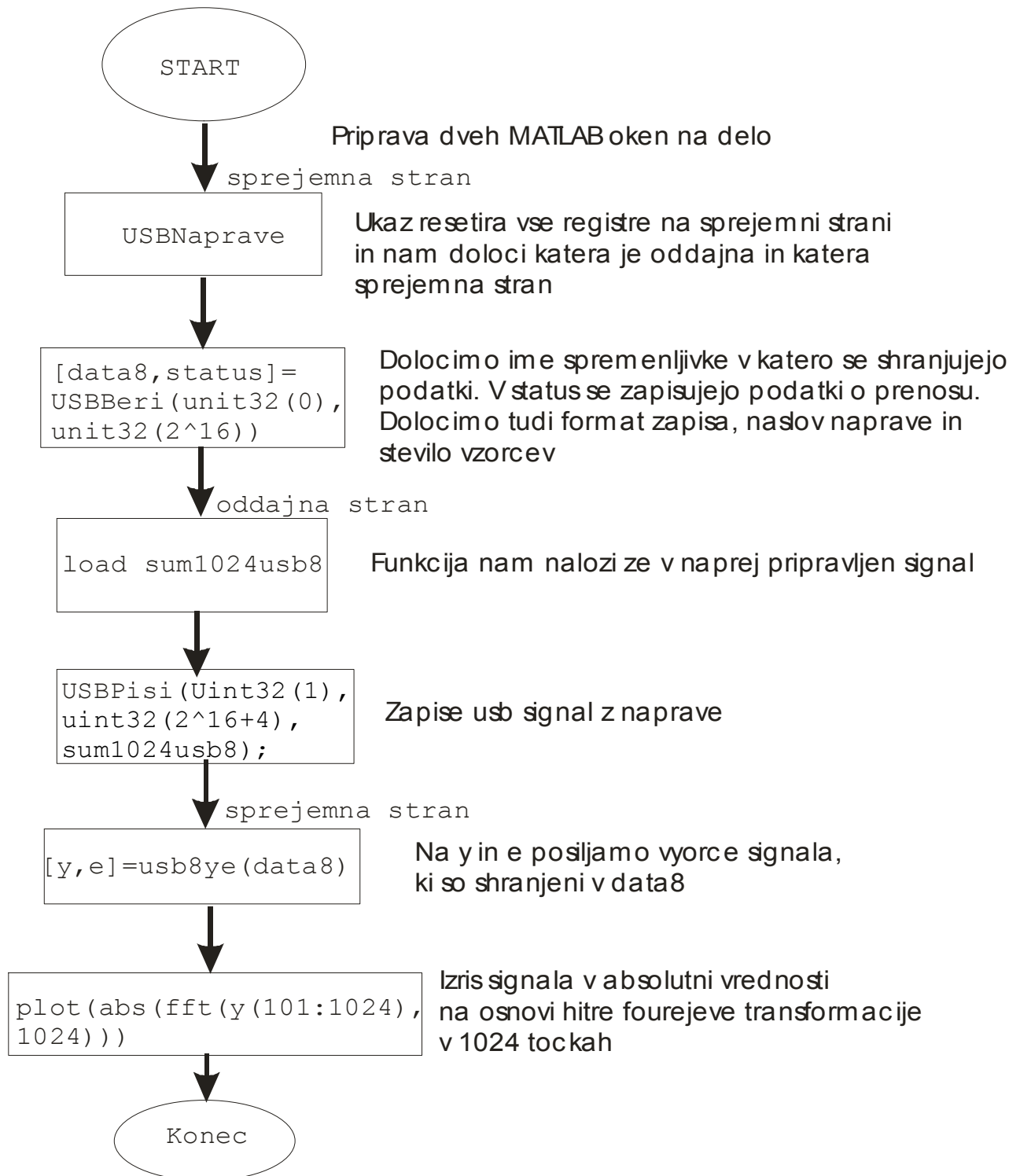
V programu Matlab [5] smo morali za pravilno delovanje in upravljanje DSP razvojne ploščice [1] nastaviti poti do datotek:

- 'C:\Delo\Raz\_deja\Matlab\Matlab\_funkcije\USB'
- 'C:\Delo\Raz\_deja\Matlab\Matlab\_funkcije'
- 'C:\Delo\Raz\_deja\Matlab\Signali'

To storimo v zavihku **File**, kjer izberemo možnost **Set Path**, nato dodamo zgoraj napisane poti. Nove nastavitve shranimo s klikom na gumb **SAVE**. Sedaj lahko dostopamo in upravljamo podatke, ki jih bomo potrebovali pri naših meritvah.

### 6.1 Odziv FIR sita z 64 koeficienti na različne vhodne signale

Za opazovanje oz. merjenje odzivov FIR sita na podani DSP ploščici, moramo imeti odprti dve MATLAB-ovi programski okni, s pomočjo katerih preko USB priključkov krmilimo oddajno (vhod – priključek USB2 na ploščici) in sprejemno (izhod – priključek USB1 na ploščici) stran FPGA vezja. Paziti moramo, da je priključitev obeh USB priključkov pravilna. Ko je USB1 = 0, mora biti na sprejemni strani  $\rightarrow \text{USBberi} = 0$ . Ko je USB2 = 1, mora biti na oddajni strani strani  $1 \rightarrow \text{USBpisi} = 1$ . Preden naložimo nove koeficiente, moramo vedno pritisniti RESET tipko na ploščici, da izbrišemo vrednosti v registrih pomnilnika. Paziti moramo tudi na to, da imamo naložen signal (to storimo z ukazom load – npr.: **load sum1024usb8**), s katerim želimo opraviti meritev, saj nam program MATLAB sicer javi napako. Na oddajni strani moramo obvezno imeti izbran direktorij: 'C:\Delo\Raz\_deja\Matlab\Matlab\_funkcije\USB'



Slika 32: Blokovna shema za merjenje odziva FIR sita

## 6.1.1 Odziv FIR sita na belošumni signal

### Postopek za merjenje odziv FIR sita na belošumni signal

#### Sprejemna stran:

```
>> USB Naprave; ;Ukaz resetira vse registre na sprejemni strani
```

```
Opis naprave          0:          FPGA_pomožni
Serijska številka naprave 0:          USB1
Lokacijska številka naprave 0:          130
```

```
Opis naprave          1:          FPGA_glavni
Serijska številka naprave 1:          USB2
Lokacijska številka naprave 1:          49
```

```
;USB1 = IZHOD
```

```
;USB2 = VHOD
```

```
>> [data8,status] = USBBeru(uint32(0),uint32(2^16));
```

#### Pomen simbolov:

<b>data8</b>	Ime spremenljivke, v katero se bodo shranili podatki
<b>status</b>	V njega se zapišejo podatki, če je bil prenos opravljen ali ne
<b>USBBeru</b>	Prebere signal z USB naprave
<b>uint32</b>	Format zapisa, zaradi izvorne kode v C++
<b>(0)</b>	Naslov naprave
<b>2^16</b>	Število vzorcev

#### Oddajna stran:

```
>> load sum1024usb8;
```

```
>> USBPisi(uint32(1),uint32(2^16+4),sum1024usb8);
```

**load,** Naložimo vzorce, ki so že vnaprej določeni (belošumni signal)

**USBPisi,** Zapiše signal z USB naprave

; paziti moramo, kako si naprava izbere indeksno vrednost, npr.: FPGA\_glavni je 1, to pomeni, da ima oddajna stran indeksno vrednost 1, sprejemna pa 0.

Z zgoraj navedenim ukazom naložimo na oddajno stran predhodno definiran signal (v tem primeru `sum1024usb8`).

V programskem oknu za sprejemno stran moramo počakati z naslednjimi ukazi, da se oddajna in sprejemna stran oz. naprava sinhronizira. To ugotovimo, da se je zgodilo tedaj, ko se v programskem oknu izpiše `FT_OK>>`.

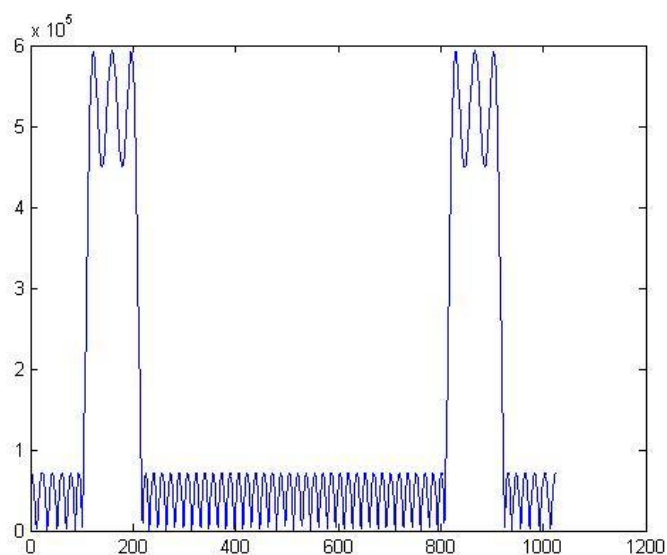
### Sprejemna stran:

```
FT_OK>> [y e]=usb8naYE(data8);
>> plot(abs(fft(y(101:1124),1024)));
```

### Pomen simbolov:

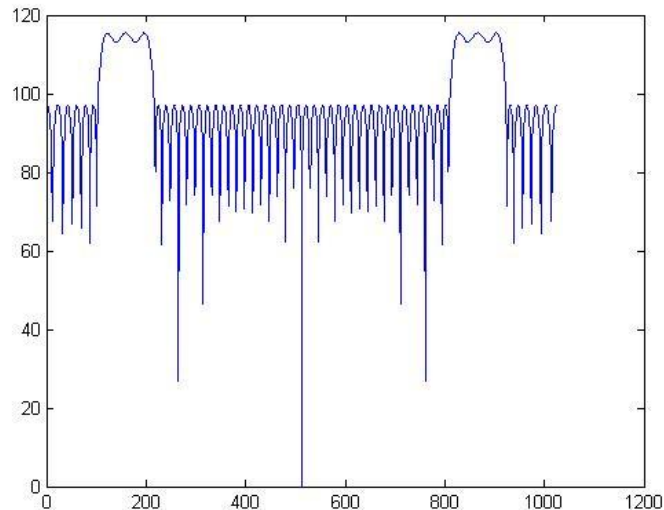
<code>[y e]</code>	Na y in e pošiljamo iste vzorce signala
<code>plot</code>	Ukaz MATLABA, kateri nam omogoča izris signala
<code>abs</code>	Absolutna vrednost
<code>fft</code>	Hitra fourierjeva transformacija v 1024 točkah.

;fft analiza gre od 101 otipka naprej (slika 33), ker nas ne zanima prehodni pojav (prvih 100 otipkov) oz. nočemo videti njegovih motenj, ki so posledica registra v FIR situ.



Slika 33: Odziv FIR sita na belošumni signal – 1024 otipkov

```
>> plot(20*log10(abs(fft(y(101:1124),1024))));
```



Slika 34: Ojačanje FIR sita s 64 koeficienti na belošumni signal

## 6.1.2 Odziv FIR sita na enotin impulz

### Postopek za merjenje odziv FIR sita na enotin impulz

Postopek je podoben kot pri belošumnem signalu, le da imamo tu enotin impulz na vhodu.

#### Sprejemna stran:

```
>> USBNaprave;
>> [data8,status] = USBBeru(uint32(0),uint32(2000));
```

#### Oddajna stran:

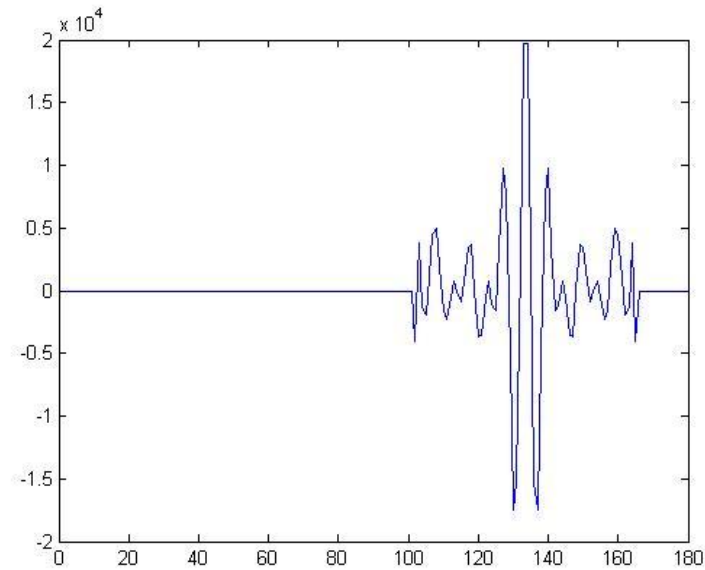
```
>> load dirak_usb8;
>> USBPisi(uint32(1), uint32(2^16+4),dirak_usb8);
```

#### Sprejemna stran:

```
FT_OK>> [y e]=usb8naYE(data8);
>> plot(y(1:180));
```

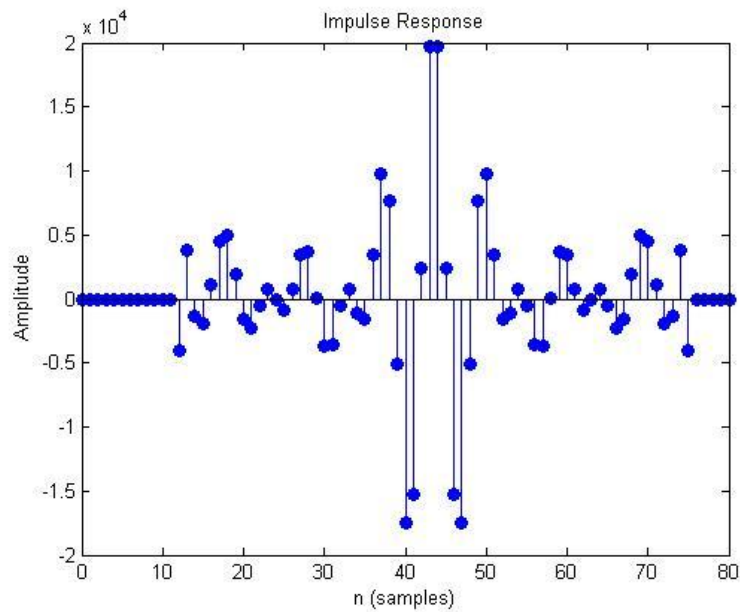


Prehodni pojav je viden v prvih 90 otipkih in je tako izrazit zaradi tega, ker nismo stisnili RESET tipke na DSP ploščici – ker imamo FIR sito s 64 koeficienti je prvih 90 otipkov napačnih oz. posledica predhodne meritve katerih vrednosti so shranjene v registrih FIR sita.



Slika 35: Odziv FIR sita na enotin impulz

```
>> impz(y(90:170));
```



Slika 36: Odziv FIR sita na enotin impulz – impulzni način

### 6.1.3 Odziv FIR sita na stopnični signal

#### Postopek za merjenje odziv FIR sita na stopnični signal

##### Sprejemna stran:

```
>> USBNaprave;
>> [data8,status] = USBBeru(uint32(0),uint32(2000));
```

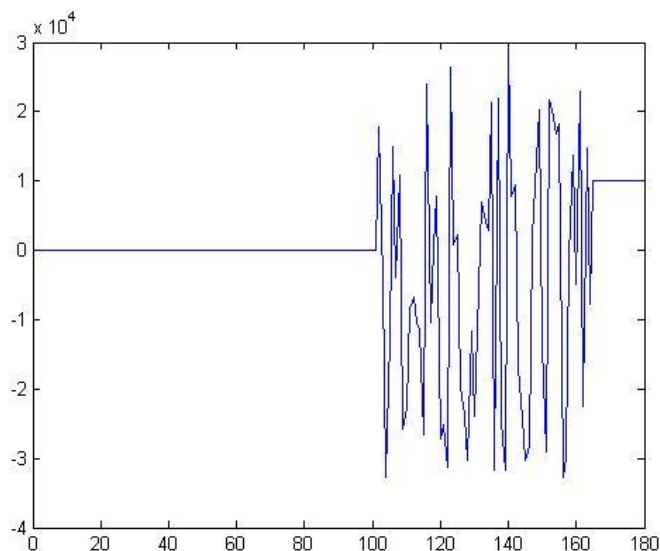
##### Oddajna stran:

```
>> load stopnica_usb8;
>> USBPisi(uint32(1), uint32(2000+4),stopnica_usb8);
```

##### Sprejemna stran:

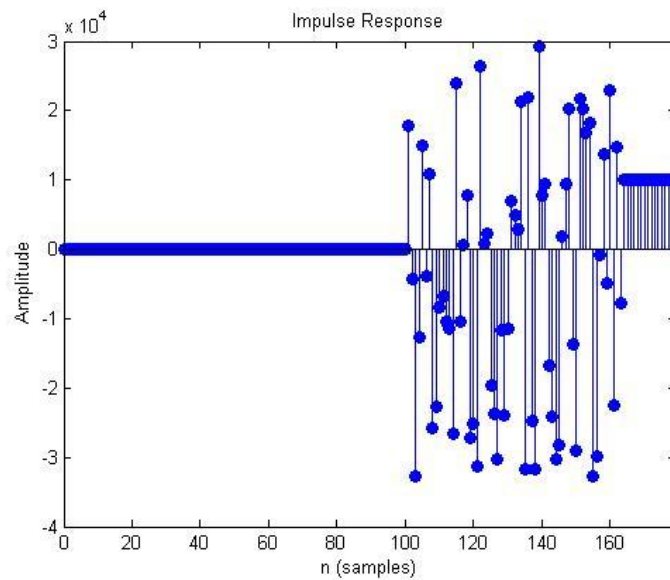
```
FT_OK>> [y e]=usb8naYE(data8);
>> plot(y(1:180));
```

Prehodni pojav je viden v prvih 90 otipkih in je tako izrazit zaradi tega, ker nismo stisnili RESET tipke na DSP ploščici – ker imamo FIR sito s 64 koeficienti je prvih 90otip kov napačnih oz. posledica predhodne meritve katerih vrednosti so shranjene v registrih FIR sita.



Slika 37: odziv FIR sita na stopnični signal

```
>> impz(y(90:170));
```



Slika 38: Odziv FIR sita na stopnični signal – impulzni način

## 6.1.4 Odziv FIR sita na signal sinusne oblike

### Postopek za merjenje odziv FIR sita na signal sinusne oblike

#### Sprejemna stran:

```
>> USBNaprave;
>> [data8,status] = USBBeru(uint32(0),uint32(2^16));
```

#### Oddajna stran:

```
>> [usb8]=xy2usb8(u,u);
>> sinusb8=sinusb8;
>> save sinusb8 sinusb8;
>> USBPisi(uint32(1),uint32(2^16+4),sinusb8);
```

#### Sprejemna stran:

```
>> [y e]=usb8naYE(data8);
>> ploty(x(1:1000),y(1:1000),x(1:1000),e(1:1000));
>> grid;
```

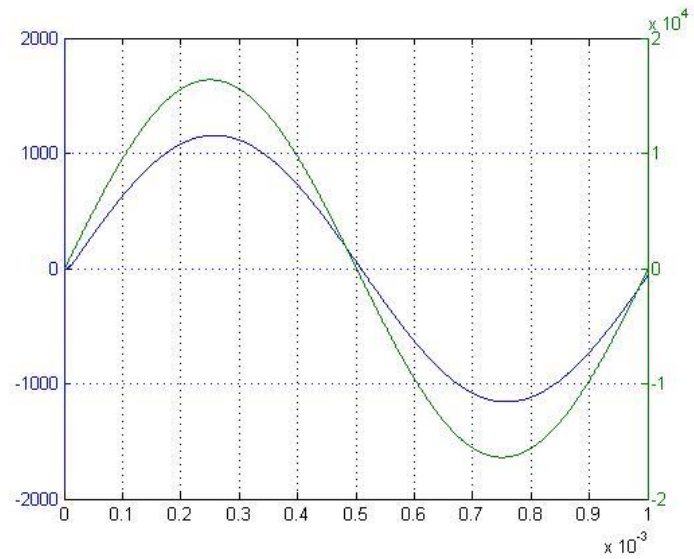
**LEGENDA:**

Signal na portu E(15:0)

VHODNI SIGNAL

Signal na portu Y(15:0)

IZHODNI SIGNAL

**Slika 39: Odziv FIR sita na signal sinusne oblike – ena perioda**

## 6.2 Amplitudni odziv FIR sita s 64 koeficienti za različno zahtevane oblike frekvenčnih karakteristik z belošumnim signalom na vhodu

Po uspešnem programiranju v FPGA vezja, ki je opisano v poglavju 3, smo izračunali in izmerili amplitudni odziv za različno zahtevne oblike frekvenčnih karakteristik, (visokoprepustno sito, nizkoprepustno sito, pasovnoprepustno sito in pasovnozaprorno sito), ter na podlagi tega dobljeno odstopanje. Izračun in meritev smo opravili v programskem okolju MATLAB z belošumnim signalom na vhodu.

### Postopek za merjenje in izračun amplitudnega odziva FIR sita s 64 koeficienti v okolju MATLAB

#### Sprejemna stran:

```
>> USB Naprave; ;Ukaz resetira vse registre na sprejemni strani
```

```
Opis naprave          0:          FPGA_pomožni
Serijska številka naprave 0:          USB1
Lokacijska številka naprave 0:          130
```

```
Opis naprave          1:          FPGA_glavni
Serijska številka naprave 1:          USB2
Lokacijska številka naprave 1:          49
```

```
;USB1 = IZHOD
```

```
;USB2 = VHOD
```

```
>> [data8,status] = USBBeru(uint32(0),uint32(2^16));
```

#### Oddajna stran:

```
>> load sum1024usb8;
```

```
>> USBPisi(uint32(1),uint32(2^16+4),sum1024usb8);
```

V programskem oknu za sprejemno stran moramo počakati z naslednjimi ukazi, da se oddajna in sprejemna stran oz. naprava sinhronizira. To ugotovimo, da se je zgodilo tedaj, ko se v programskem oknu izpiše **FT\_OK**>>.

### Sprejemna stran:

```
FT_OK>> [y e]=usb8naYE(data8);
>> plot(abs(fft(y(101:1124),1024)));

>> FIR64NS_USB8 = data8
>> save FIR64NS ;rezultat shranimo v datoteko .mat (ime datoteke
koeficientov FIR sita).
>> save FIR64NS_USB8 ;rezultat shranimo v datoteko .mat (ime datoteke
izmerjenega vhodnega signala in odziva FIR sita).
```

Ko smo sprogramirali FPGA vezje z eno izmed vrst FIR sita (npr. nizkoprepustno sito), smo opravili meritev v okolju MATLAB po postopku:

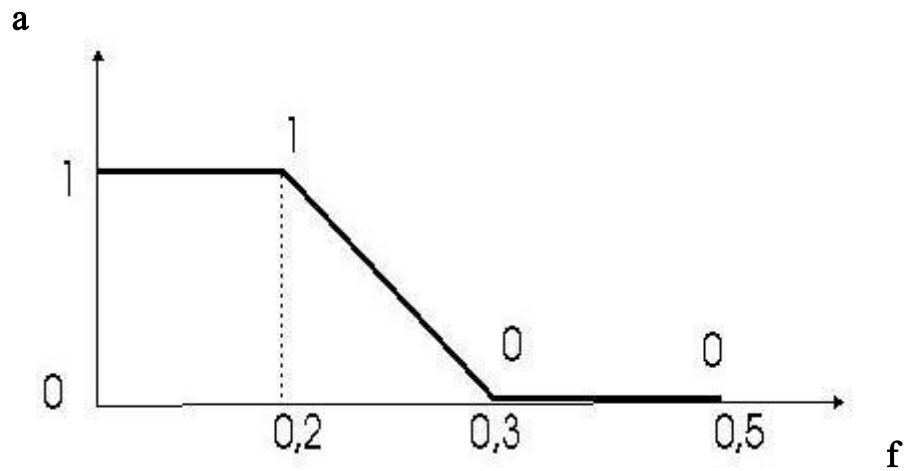
```
>> meritev_FIR FIR64NS.mat
>> meritev_FIR FIR64NS_USB8.mat
```

### Definiranje frekvenčnega in amplitudnega vektorja

**f** pomeni frekvenčni vektor

**a** pomeni amplitudni vektor

```
f = [0 0.2 0.3 0.5];
a = [1 1 0 0];
```



Slika 40: Definiranje frekvenčnega in amplitudnega vektorja

Če hočemo dobiti karakteristiko, kot je na sliki 40, moramo za  $a$  in  $f$  vnesti koeficiente, ki jih razberemo iz te karakteristike. Bolj kot je strma karakteristika, bolj so ostrejšje zahteve. V našem primeru smo to izmerili za visokoprepustno in nizkoprepustno sito. Na podlagi frekvenčnega in amplitudnega vektorja smo izračunali karakteristične delne vsote koeficientov.

## 7 REZULTATI

Rezultat smo izvršili z programskim okoljem MATLAB in s pomočjo skriptne datoteke `meritev_FIR`. Analizo med izračunanimi in izmerjenimi amplitudnimi odzivi smo opravili za šest vrst sita. Vsi rezultati so prikazani grafično, ki so opremljeni z različnimi grafi in komentarji. Pri vsakem situ je prikazan frekvenčni ( $f$ ) in amplitudni ( $a$ ) vektor, ter največja in najmanjša karakteristična delna vsota koeficientov, ki je zapisana v hexadecimalnem zapisu. Amplitudo in delno vsoto smo normirali tako, da ne preseže vrednost 1.

### 7.1 Visokoprepustno sito

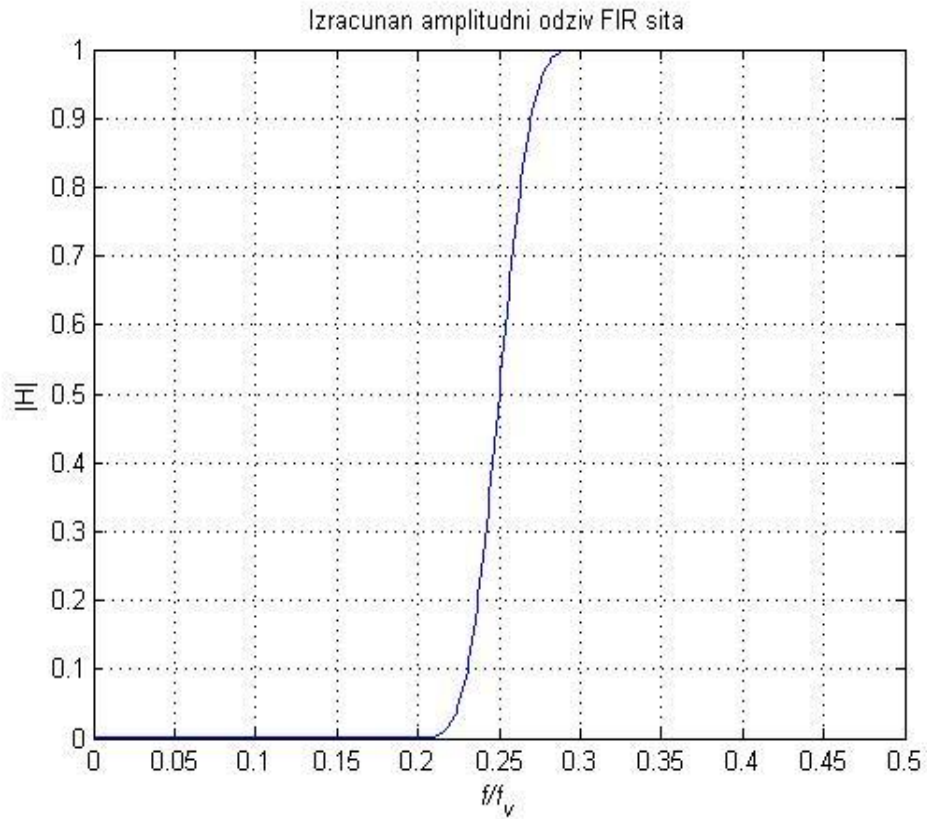
Pri merjenju visokoprepustnega sita smo morali zaradi neoptimalnega algoritma znižati njegovo stopnjo na 62. Manjkajoči koeficienti, ki niso določeni, imajo zato vrednost enako 0. Na slikah 41, 42 in 43 so prikazani izmerjeni in izračunani odzivi visokoprepustnega sita, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati `FIR63VSo.mat`, `FIR63VSo_USB8.mat` in `mcs` datoteko `FIR64VSo.mcs`

V zapornem pasu dosežemo pri izračunanem odzivu slabljenje nad 100 dB, pri izmerjenem pa okoli 80 dB. V prepustnem pasu je slabljenje 0 dB. Takšno sito je dobro načrtano.

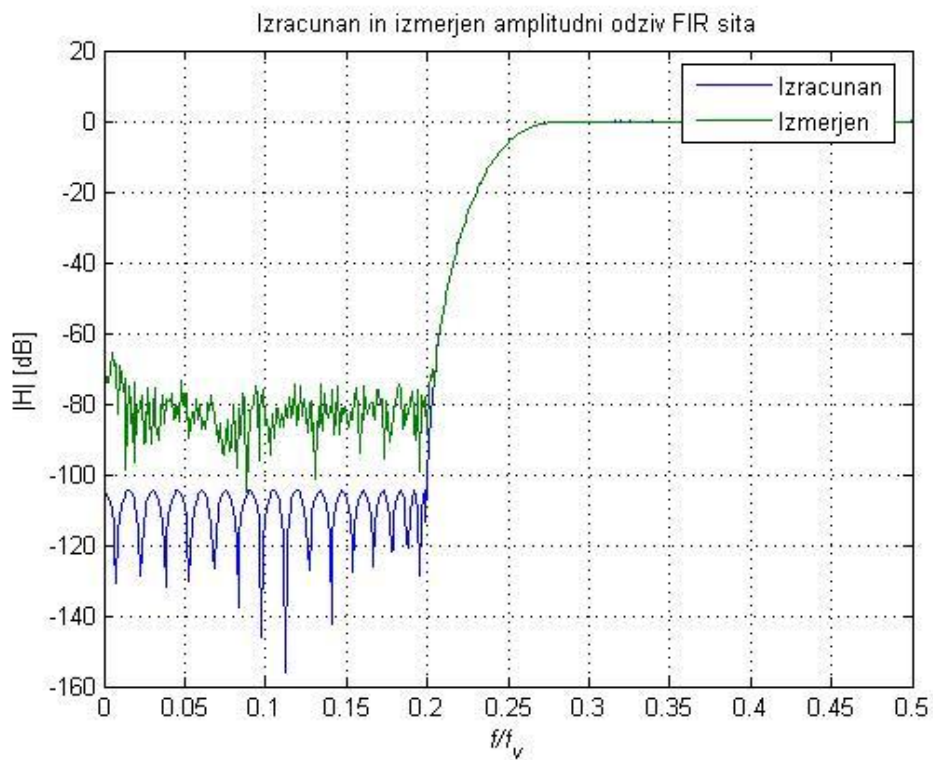
Absolutno odstopanje med izmerjeno in izračunano vrednostjo je v meji  $0,5 \cdot 10^{-3}$ . Največje odstopanje je pri prehodnem pasu, v prepustnem pasu je malo manjše, v zapornem pasu pa ni zaznati velikih pogreškov

```
f = [0 0.2 0.3 0.5]*2; a = [0 0 1 1];
h = firpm(62,f,a);
ans =
4D0B
ans =
D770
```

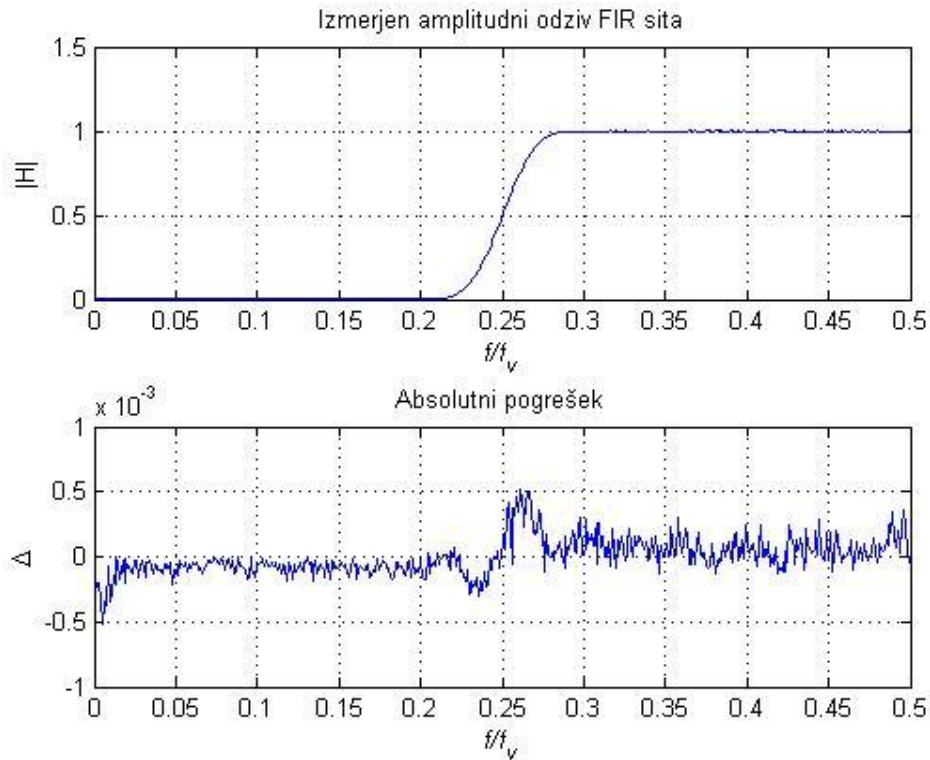




Slika 41: Izračunan amplitudni odziv visokoprepustnega sita



Slika 42: Izračunan in izmerjen amplitudni odziv visokoprepustnega sita



Slika 43: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom visokoprepustnega sita

## 7.2 Visokoprepustno sito ostrejše zahteve

Pri merjenju visokoprepustnega sita smo morali zaradi neoptimalnega algoritma znižati njegovo stopnjo na 62. Manjkajoči koeficienti, ki niso določeni, imajo zato vrednost enako 0. Na slikah 44, 45 in 46 so prikazani izmerjeni in izračunani odzivi visokoprepustnega sita, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati FIR63VS.mat, FIR63VS\_USB8.mat in mcs datoteko FIR64VS.mcs

Odstopanje med izmerjenim in izračunanim odzivom je minimalno v celotnem frekvenčnem pasu. Pri obeh dosežemo slabljenje v zapornem pasu nad 20 dB.

Absolutno odstopanje je v meji  $0,5 \cdot 10^{-3}$ . Največje je v prehodnem pasu, v prepustnem pasu in zapornem pasu pa malo manjše.

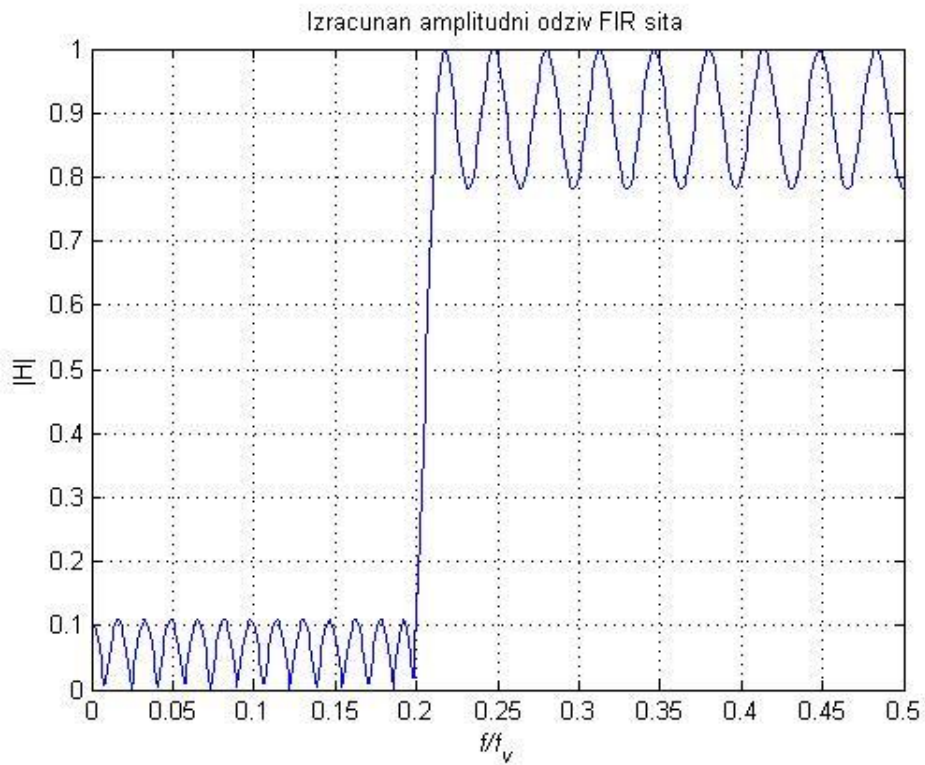
```
f = [0 0.2 0.21 0.5]*2; a = [0 0 1 1];
```

```
h = firpm(62, f, a);
```

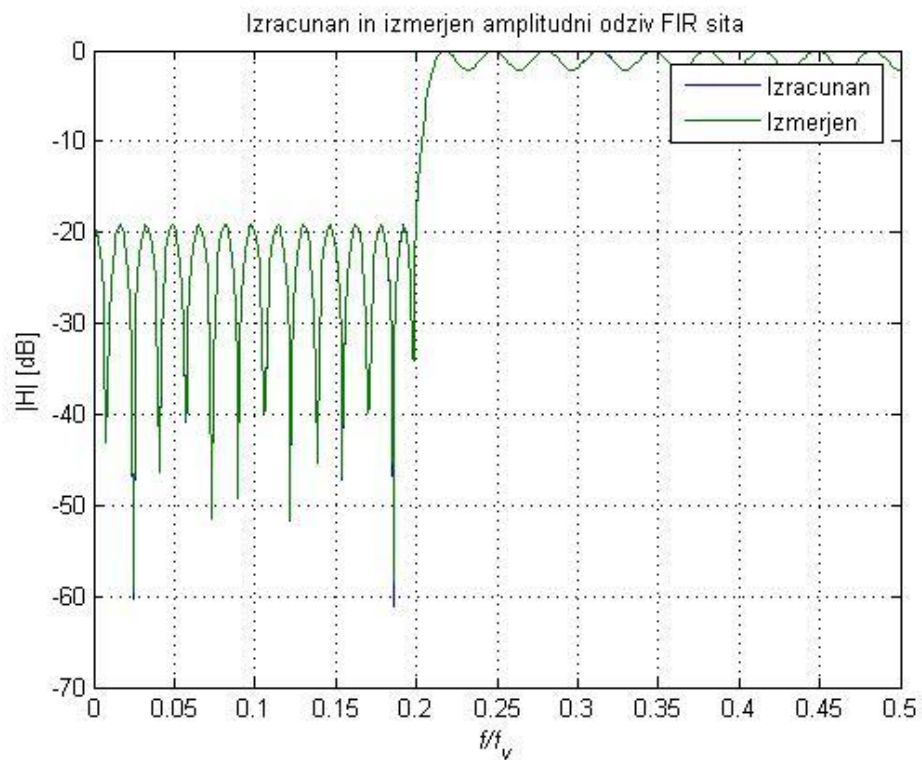
```
ans =
```

```
4C7D
```

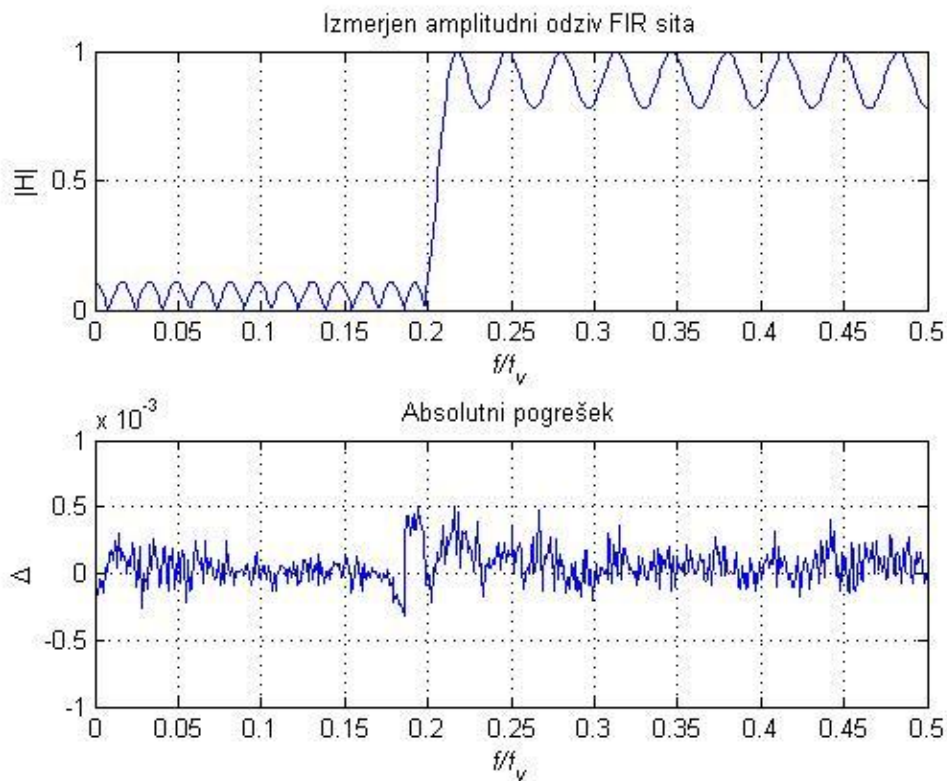
ans =  
D2B8



Slika 44: Izračunan amplitudni odziv visokoprepustnega sita ostrejše zahteve



Slika 45: Izračunan in izmerjen amplitudni odziv visokoprepustnega sita ostrejše zahteve



Slika 46: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom visokoprepustnega sita ostrejšje zahteve

### 7.3 Nizkoprepustno sito

Na slikah 47, 48 in 49 so prikazani izmerjeni in izračunani odzivi nizkoprepustnega sita, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati FIR64NSo.mat, FIR64NSo\_USB8.mat in mcs datoteko FIR64NSo\_.mcs

Pri izračunanem odzivu dosežemo slabljenje nad 100 dB, pri izmerjenem pa je slabljenje okoli 85 dB. V prepustnem pasu je slabljenje 0 dB, kar je zelo dobro.

Absolutno odstopanje med izmerjeno in izračunano vrednostjo presega določeno mejo  $0,5 \cdot 10^{-3}$  v prehodnem pasu. Odstopanje je v prehodnem pasu zelo veliko, v prepustnem pasu pa malo večje kot v zapornem pasu, kjer ni zaznati velikih pogreškov.

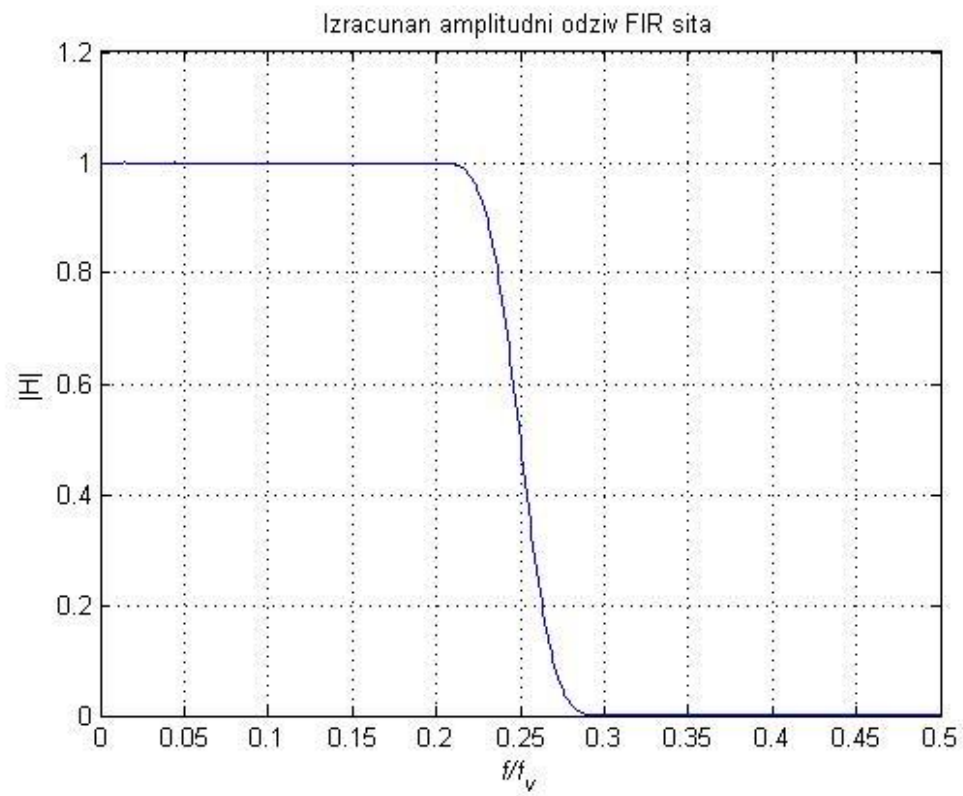
```
f = [0 0.2 0.3 0.5]*2; a = [1 1 0 0];
h = firpm(63,f,a);
```

```
ans =
```

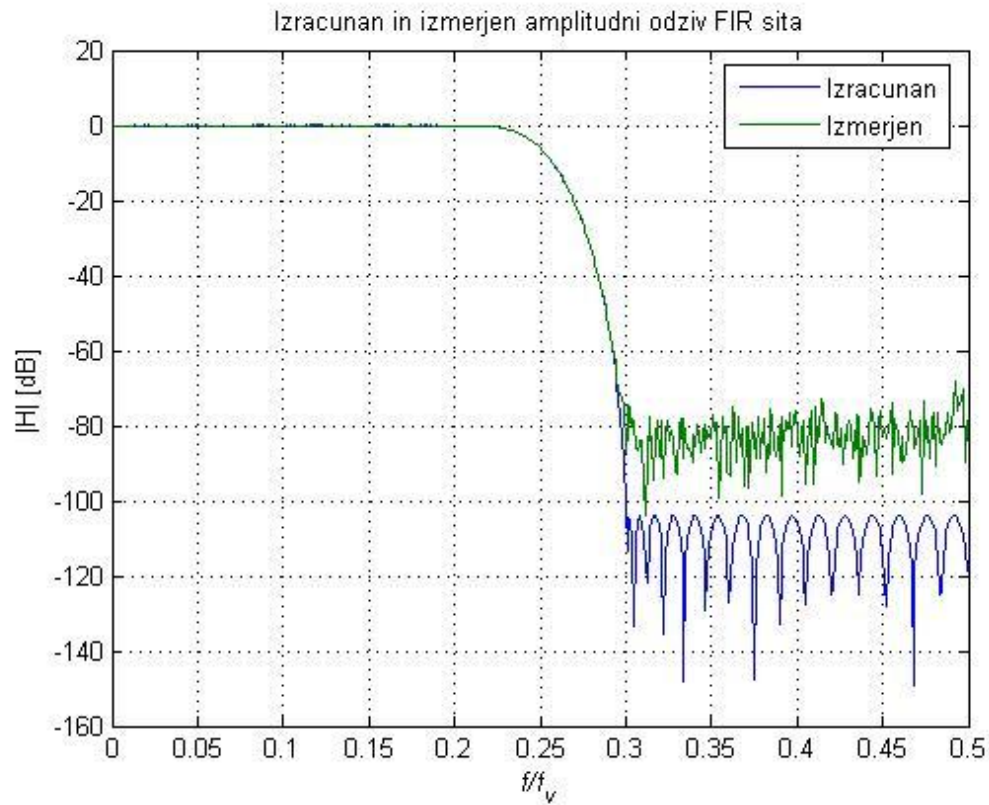
4C91

ans =

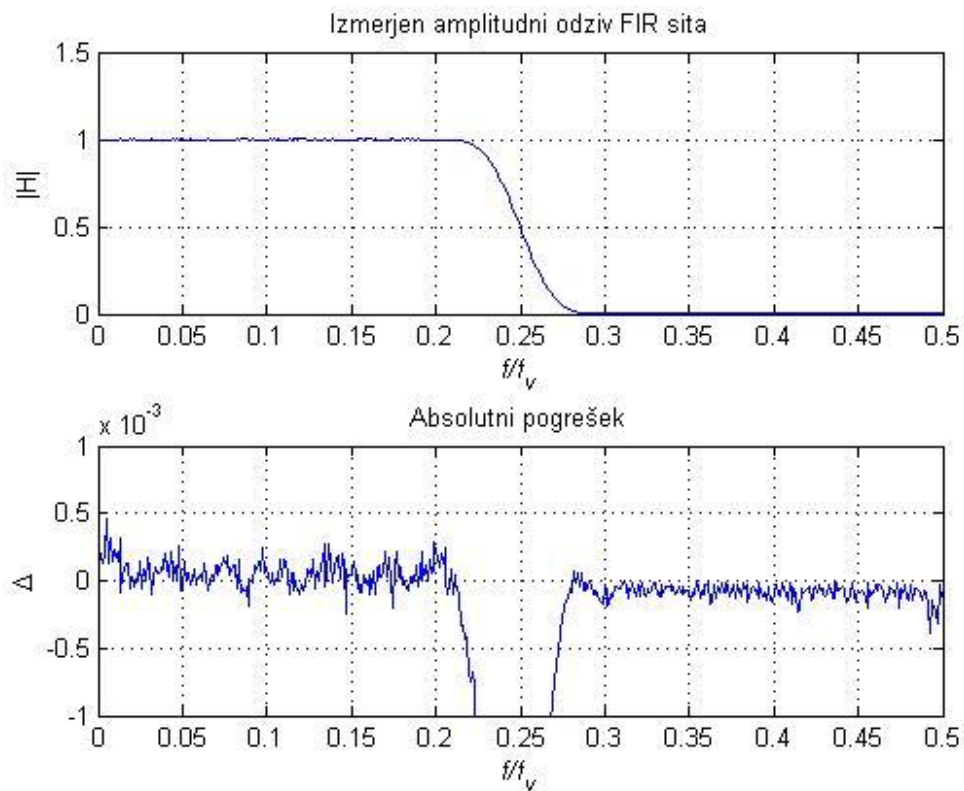
ED03



Slika 47: Izračunan amplitudni odziv nizkoprepustnega sita



Slika 48: Izračunan in izmerjen amplitudni odziv nizkoprepustnega sita



Slika 49: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom nizkoprepustnega sita

## 7.4 Nizkoprepustno sito ostrejše zahteve

Na slikah 50, 51 in 52 so prikazani izmerjeni in izračunani odzivi nizkoprepustnega sita z ostrejšimi zahtevami, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati FIR64NS.mat, FIR64NS\_USB8.mat in mcs datoteko FIR64NS\_.mcs

Odstopanje med izmerjenim in izračunanim odzivom je minimalno v celotnem frekvenčnem pasu. Pri obeh dosežemo slabljenje v zapornem pasu nad 20 dB.

Absolutno odstopanje je v meji  $0,5 \cdot 10^{-3}$ . Največje je v prehodnem pasu, v prepustnem pasu in zapornem pasu pa je malo manjše.

```
f = [0 0.2 0.21 0.5]*2; a = [1 1 0 0];
```

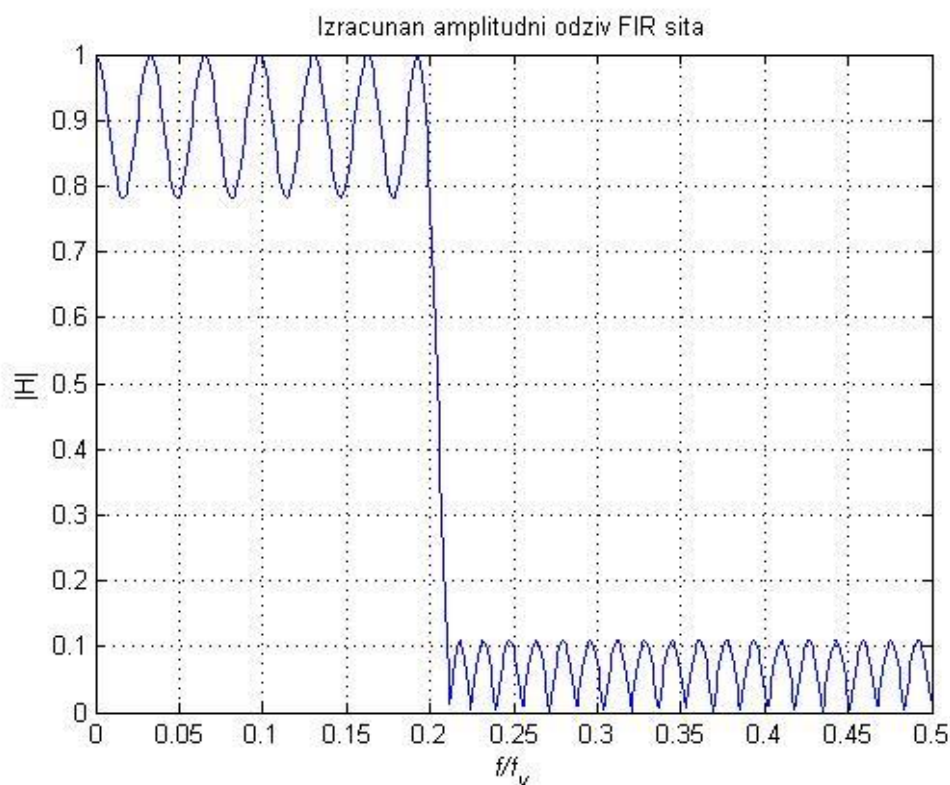
```
h = firpm(63, f, a);
```

```
ans =
```

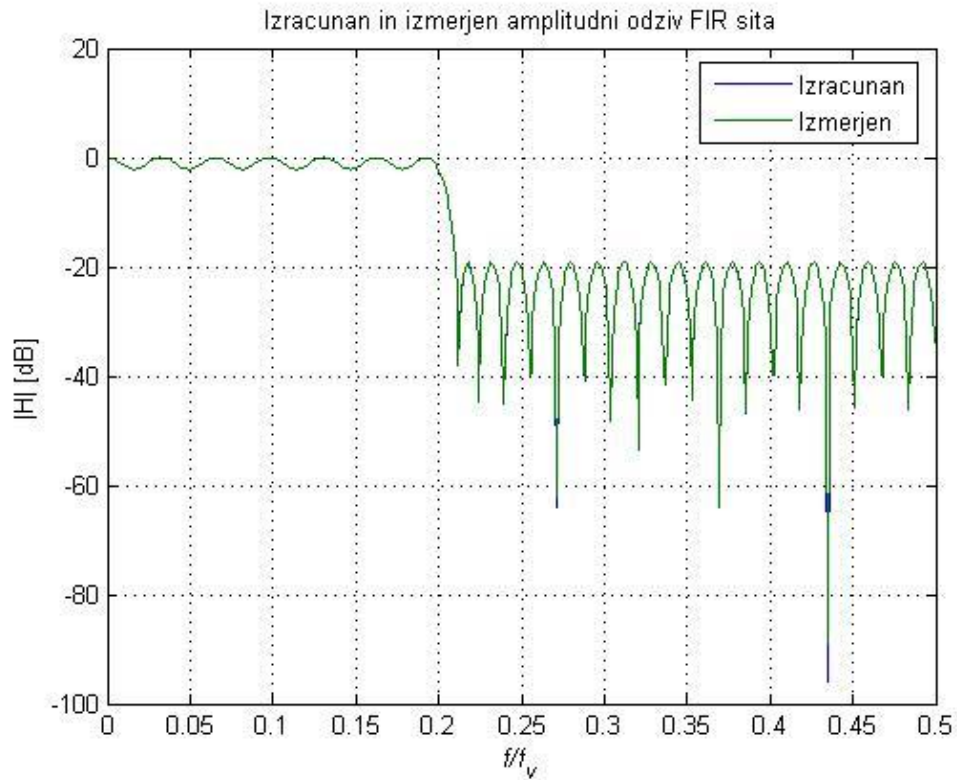
```
423E
```

```
ans =
```

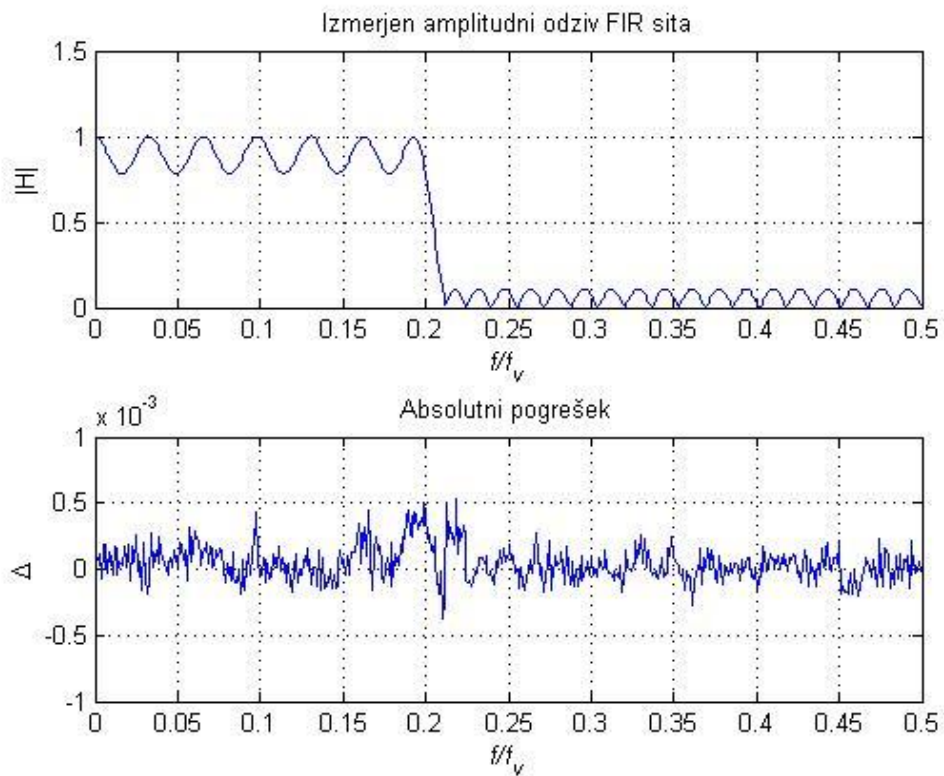
```
F4B9
```



Slika 50: Izračunan amplitudni odziv nizkoprepustnega sita ostrejše zahteve



Slika 51: Izračunan in izmerjen amplitudni odziv nizkoprepustnega sita ostrejšje zahteve



Slika 52: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom nizkoprepustnega sita ostrejšje zahteve



## 7.5 Pasovnoprepustno sito

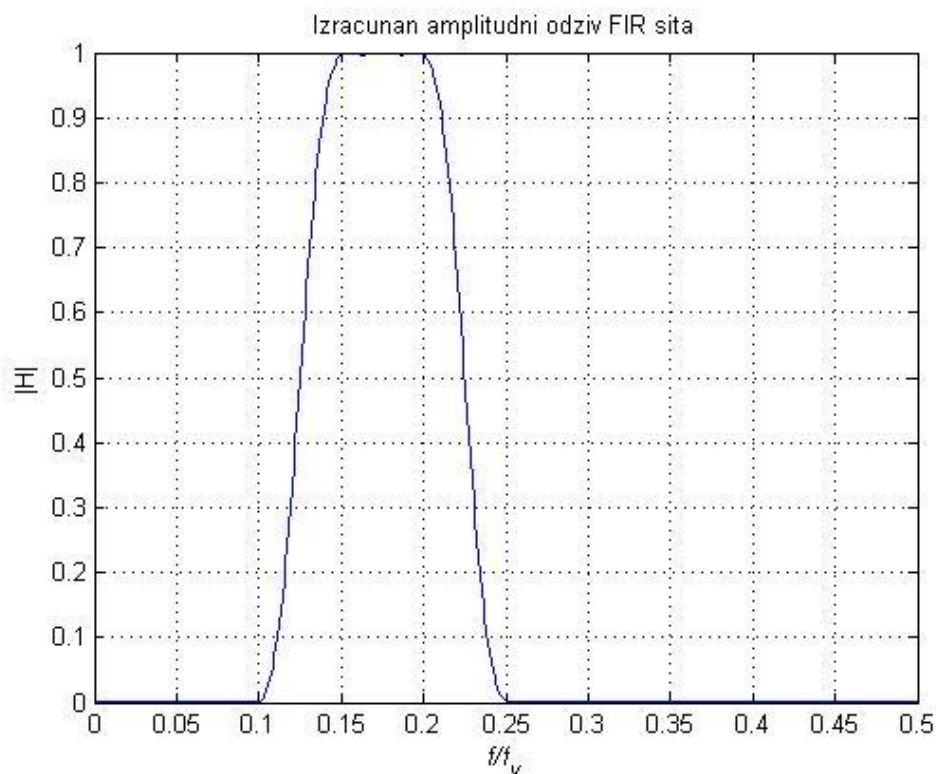
Na slikah 53, 54 in 55 so prikazani izmerjeni in izračunani odzivi pasovnoprepustnega sita, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati FIR64PPo.mat, FIR64PPo\_USB8.mat in mcs datoteko FIR64PPo\_.mcs

Odstopanje med izmerjenim in izračunanim odzivom je minimalno v celotnem frekvenčnem pasu. Pri obeh je slabljenje v zapornem pasu nad 58 dB.

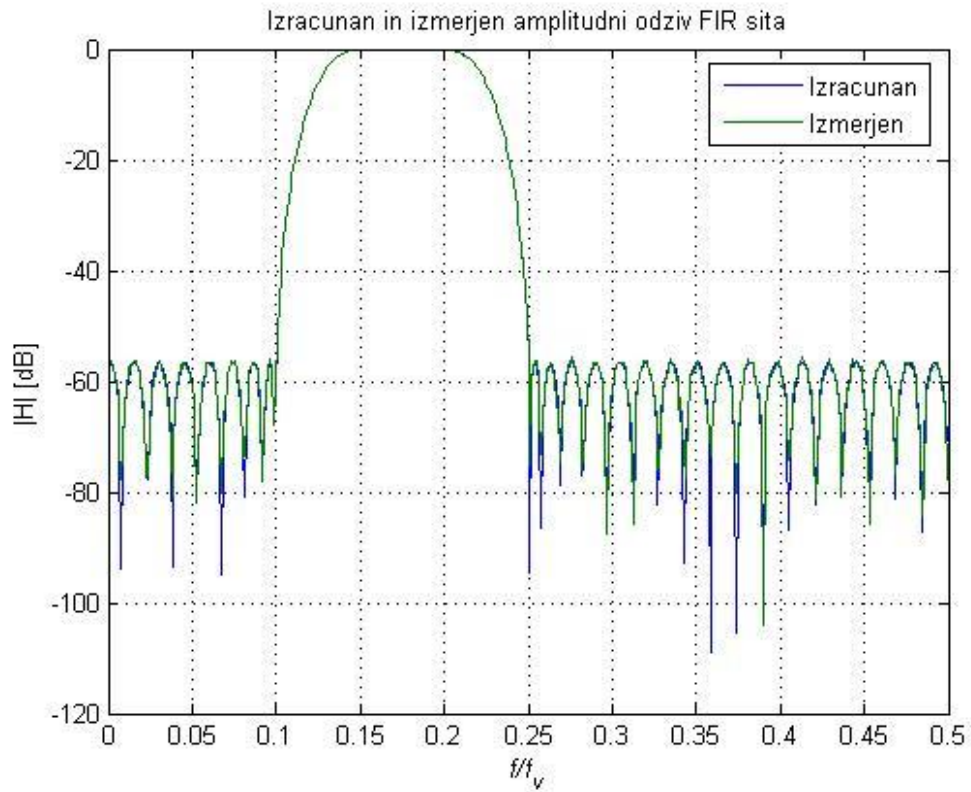
Absolutno odstopanje je v meji  $0,5 \cdot 10^{-3}$ . Odstopanje je največje v prepustnem pasu, v zapornem in prehodnem pasu pa je skoraj enako, čeprav je v prehodnem pasu malo manjše.

```
f = [0 0.1 0.15 0.2 0.25 0.5]*2; a = [0 0 1 1 0 0];
h = firpm(63,f,a);
```

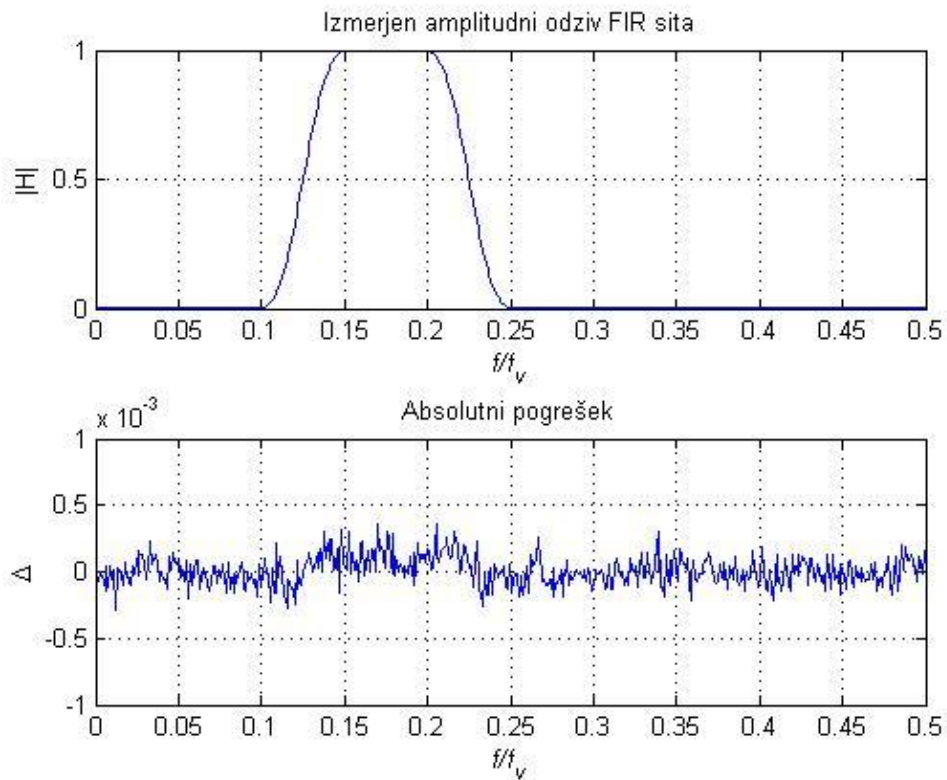
```
ans =
17ED
ans =
D9BB
```



Slika 53: Izračunan amplitudni odziv pasovnoprepustnega sita



Slika 54: Izračunan in izmerjen amplitudni odziv pasovnoprepustnega sita



Slika 55: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom pasovnoprepustnega sita

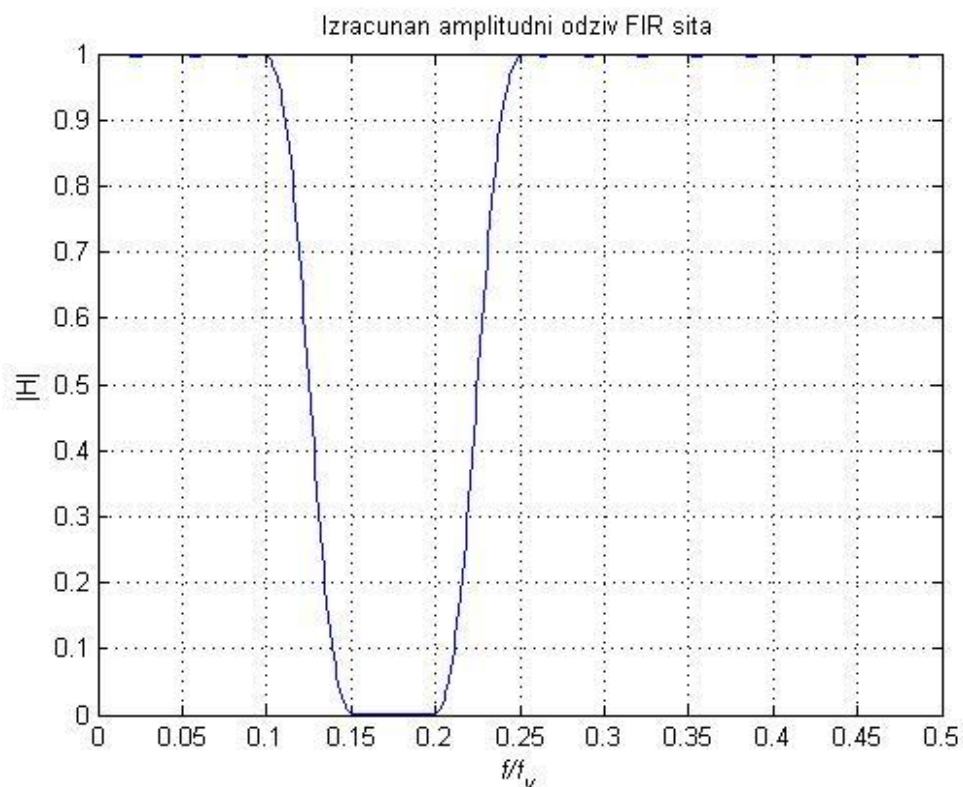
## 7.6 Pasovnozaporno sito

Pri merjenju pasovnoprepustnega sita, smo morali zaradi neoptimalnega algoritma znižati njegovo stopnjo na 62. Manjkajoči koeficienti, ki niso določeni, imajo zato vrednost enako 0. Na slikah 56, 57 in 58 so prikazani izmerjeni in izračunani odzivi pasovnoprepustnega sita, ter absolutni pogrešek. V primeru ponovne obdelave rezultatov uporabimo datoteke z rezultati FIR63PZo.mat, FIR63PZo\_USB8.mat in mcs datoteko FIR63PZo\_.mcs

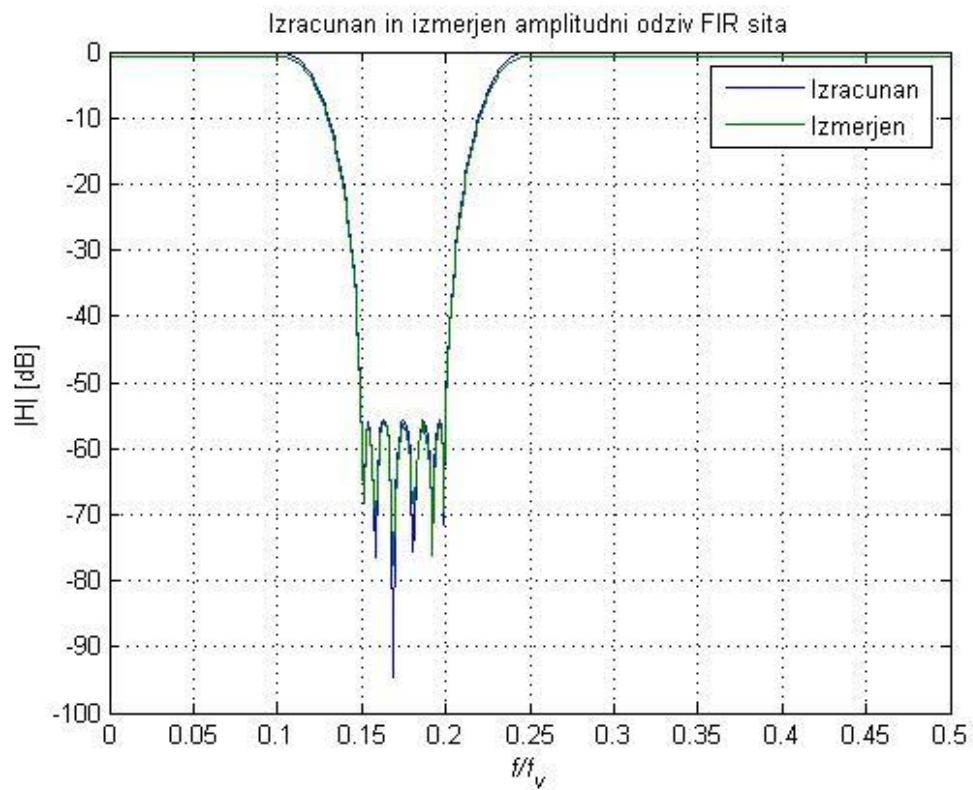
Odstopanje med izmerjenim in izračunanim odzivom je minimalno v celotnem frekvenčnem pasu. Pri obeh je slabljenje v zapornem pasu nad 57 dB. Absolutni pogrešek je izven meje  $0,5 \cdot 10^{-3}$ .

```
f = [0 0.1 0.15 0.2 0.25 0.5]*2; a = [1 1 0 0 1 1];
h = firpm(62, f, a);
```

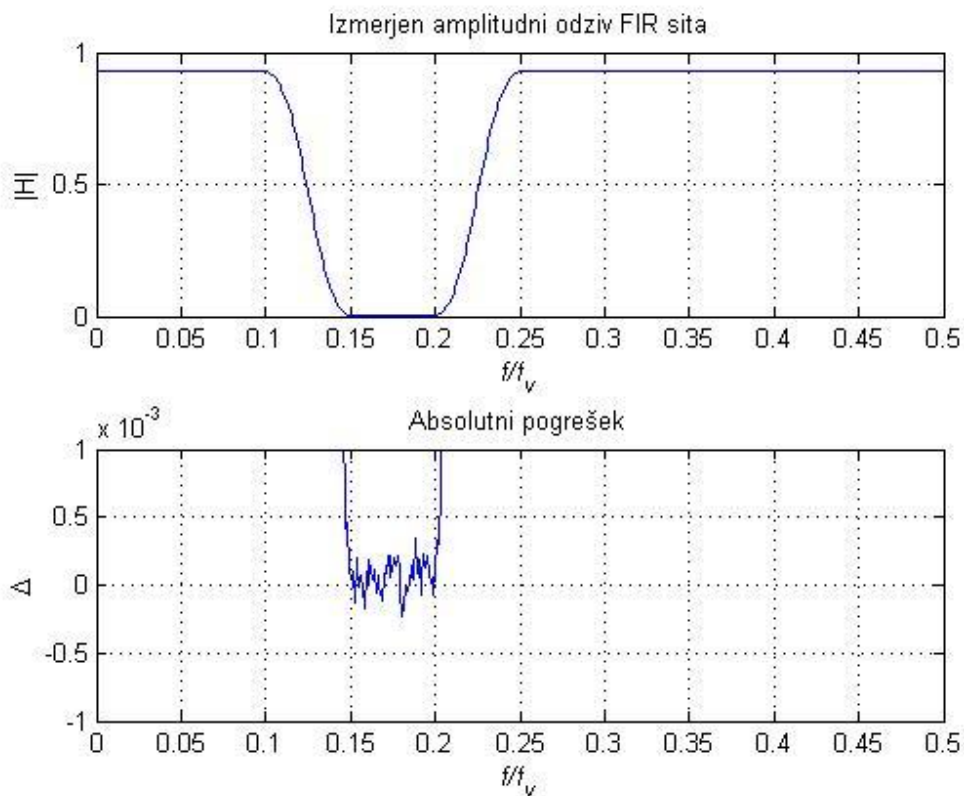
```
ans =
7FFF
ans =
EA06
```



Slika 56: Izračunan amplitudni odziv pasovnozapornega sita



Slika 57: Izračunan in izmerjen amplitudni odziv pasovnozapronega sita



Slika 58: Izmerjena linearna karakteristika ojačanja in absolutni pogrešek med izmerjenim in izračunanim odzivom pasovnozapronega sita

## 8 SKLEP

V diplomskem delu smo opisali možnost izvedbe nerekurzivnega digitalnega sita 63. stopnje s programirnimi logičnimi (FPGA) vezji v strukturi porazdeljene aritmetike. Za izračun in meritev odziva digitalnega sita, smo uporabljali razvojno okolje MATLAB, s katerim smo neposredno upravljali z razvojno enoto, Spartan 3 s FPGA vezjem XC3S400, preko dveh USB priključkov. S programskim paketom Xilinx - ISE smo opravili implementacijo v dve programirni FPGA vezji. Za izvedbo strukture porazdeljene aritmetike smo uporabili postopek sprotnega izračuna delnih vsot koeficientov. Ta logična struktura je najkompleksnejši del vezja in omogoča dovolj hiter izračun delnih vsot, ter vpis koeficientov sita. Žal pa aparturna kompleksnost narašča z  $2^N$ , pri čemer je N število koeficientov digitalnega sita.

Za verifikacijo delovanja smo digitalno sito in vhodne signale opisali v VHDL jeziku. Izdelano digitalno sito je bilo namenjeno izločanju šuma iz koristnega signala. Za testiranje digitalnega sita smo uporabili belošumni vhodni signal.

Rezultate smo dobili na podlagi primerjave izračunanih in izmerjenih različnih frekvenčnih karakteristik sit pri manj in bolj strogih pogojih, ter na podlagi tega dobljeno odstopanje oz pogrešek.

Dobro so se izkazala tako digitalna sita pri manj strogih pogojih, kot tudi sita z strogimi pogoji. Pri visokoprepustnem situ, z manj, kot tudi z bolj strogimi pogoji in pri pasovnozaporjem situ, algoritem ni uspel določiti koeficientov, zato smo morali znižati FIR sito za eno stopnjo.

Zaradi 16-bitne kvantizacije v sami izvedbi za implementacijo v FPGA se odstopanje pokaže šele ko je slabljenje sita pod 80dB. Absolutna odstopanja smo opisali in podali pri rezultatih za vsako sito posebej. Pri pasovnozaporjem situ smo dobili absolutno odstopanje izven določene meje in je predmet nadaljevalnega dela.

## 9 LITERATURA

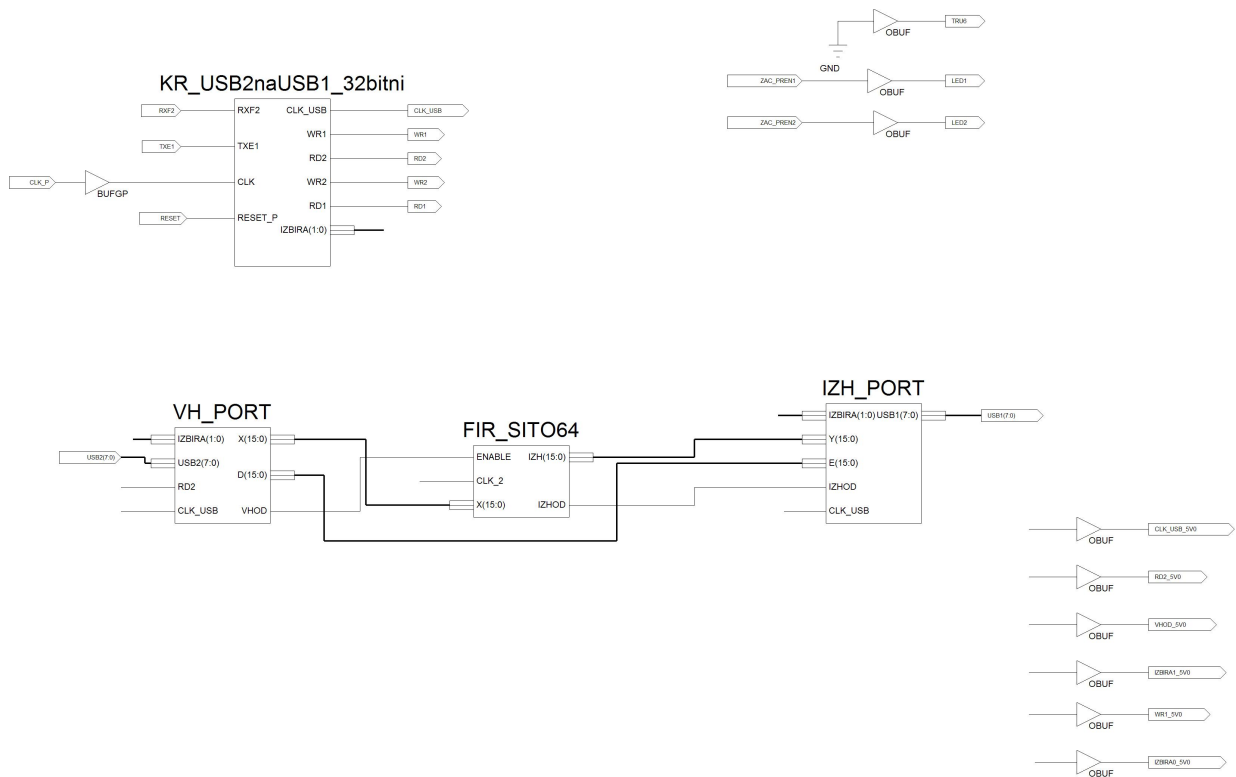
- [1] A. Greifoner, Razvojna plošča za programirna vezja Spartan 3 firme Xilinx, Diplomsko delo, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, 2007.
- [2] D. Osebik, Novi postopki implementacije adaptivnih digitalnih sit s programirnimi vezji, Doktorska disertacija, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, 2005.
- [3] OSEBIK Davorin, KOSTANJEVEC Boris, JARC Bojan, SOLAR Mitja, BABIC Rudolf. Izvedba nerekurzivnega digitalnega sita s programirljivim poljem logičnih vezij v strukturi porazdeljene aritmetike.
- [4] Žarko Čučej, Peter Planinšič, teorija signalov, Maribor: Fakulteta za elektrotehniko, računalništvo in informatiko, 2000.
- [5] Navodila za uporabo programskega paketa MATLAB s simulinkom, Maribor, Institut za robotiko.
- [6] <http://www.xilinx.com/>
- [7] <http://www.fe.uni-lj.si/~lrnv/izpiti/binarno1.html>
- [8] D. Hajsok, izvedba generatorja belega šuma s programirljivim poljem logičnih gradnikov, diplomsko delo, Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor, 1998.
- [9] Davorin Osebik, Rudolf Babič, Bogomir Horvat, Adaptivna digitalna sita v strukturi porazdeljene aritmetike, članki z raziskovalnega področja, Fakulteta za elektrotehniko, računalništvo in informatiko Univerza v Mariboru.

## 10 PRILOGE

Priloga a: Meritev skupnega amplitudnega odziva digitalnih sit

Priloga b: Seznam vseh datotek, potrebnih za analizo digitalnih sit

### Priloga a: Meritev skupnega amplitudnega odziva digitalnih sit



### Priloga b: Seznam vseh datotek, potrebnih za analizo digitalnih sit

FIR63VSo.mat

FIR63VSo\_USB8.mat

FIR64VSo.mcs

FIR64NSo.mat

FIR64NSo\_USB8.mat

FIR64NSo.mcs

FIR64PPo.mat

FIR64PPo\_USB8.mat

FIR64PPo.mcs

FIR63PZo.mat

FIR63PZo\_USB8.mat

FIR63PZo.mcs

FIR64VS.mat

FIR64VS\_USB8.mat

FIR64VS.mcs

FIR64NS.mat

FIR64NS\_USB8.mat

FIR64NS.mcs

KDVK\_64.m

Meritev\_FIR.m

Rom\_KDVK\_64.m



## Življenjepis

Ime in priimek: Marko Kočever  
Rojstni podatki: 27. 06. 1984 Celje  
Šolanje: 1991 – 1993 Osnovna šola Prevorje  
1993 – 1999 Osnovna šola Lesično  
1999 – 2003 Šolski center Celje, Poklicna in tehnična elektro in kemijska šola  
2003 – 2008 Fakulteta za elektrotehniko, računalništvo in informatiko, Maribor

### Naslov študenta:

Marko Kočever

Krivica 1

3262 Prevorje

E-mail: [kocevar.marko@gmail.com](mailto:kocevar.marko@gmail.com)

## **IZJAVA**

Podpisani Marko Kočever izjavljam, da sem diplomsko delo izdal sam. Zavedam se tudi morebitnih posledic v primeru kršitve izjave.

---