|  |  |  |
| --- | --- | --- |
|  | **UNIVERZITET U NOVOM SADU  FAKULTET TEHNIČKIH NAUKA** |  |

|  |
| --- |
| **Departman za računarstvo i automatiku**  **Smer računarstvo i automatika**  **PREDMETNI PROJEKAT II**  **Kandidat: Ognjen Stojisavljević**  **Broj indeksa: RA155/2019**  **Predmet: Osnovi algoritama i struktura DSP-a**  **Tema rada: Kompresija singala** |
| **Novi Sad, decembar 2021.** |

**Sadr** **žaj**

**0)** Uvod

**1)** Zadatak 1

**2)** Zadatak 2

**3)** Zadatak 3

**4)** Zadatak 4

**Uvod**

U predmetnom projektu naš zadatak je bio da na zadati ulazni signal

dobijemo komprimovan izlazni signal, što većeg kvaliteta, sa minimalnim šumom.

Kompresija omogućava smanjenje propusnog opsega za prenos digitalnog signala.

Zatim rekonstrukcijom dobijamo ulazni signal sa nekim stepenom oštećenja.

Hofmanovim kodovanjem eliminišemo elemente koji nemaju korisnu informaciju(redudansa). Hofmanovo kodovanje omogućava i kompresiju bez gubitaka. Kodovanjem stvaramo nove datoteke sa .dsp1 i .dict nazivom, koje sadrze komprimovanu datoteku i njeno zaglavlje.

Kvantizacija se vrši u frekventnom domenu, jer raspodela šuma zavisi od amplitudskog spektra kompresovanog signala.

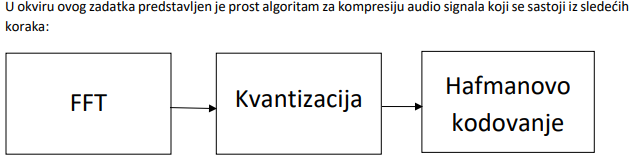
Pošto su prisutna 2 kanala, na kraju radimo uČešljavanje kanala.

**Alate koje koristimo za kvantizaciju su :**

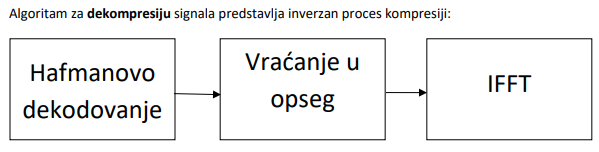
- Izvršne datoteke hoffmanovog kodovanja/dekodovanja

- CodeComposerStudio pomoću kojeg implementiramo odredjene funkcionalnosti (FFT, kompresija, rekonstrukcija...)

Algoritam za kodovanje:



Algoritam za dekodovanje:



Primenjujemo brzu furijeovu transformaciju na poČetku a inverznu na kraju obrade. Suprotna operacija kvantizaciji je vraćanje u opseg, odnosno rekonstrukcija.

**Zadatak 1**

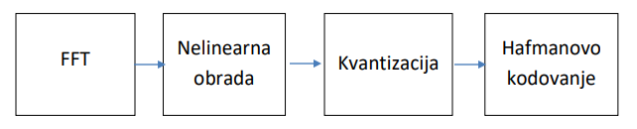
Za kvantove 16 za oba kanala, dobija se slabo komprimovan a skoro isti kao i ulazni signal.

Za kvantove 10 za oba kanala, kompresija je primetno veća, a kvalitet primetno lošiji nego kod prošlog primera.

Za kvantove 8 za oba kanala, kompresija velika, ali veliki gubitak kvaliteta signala.

**Zadatak 2**

Proširujemo prethodni zadatak ubacivanjem nelinearne obrade po formuli: X=sign(X)\*sqrt(X), sign je funkcija za odredjivanje znaka broja.



Za kvantove 10 za oba kanala, kompresija je dobra, a kvalitet malo lošiji nego kod originala.

Za kvantove 8 za oba kanala, kompresija veća, medjutim i dalje dobar kvalitet signala(min gubitaka).

**Zadatak 3**

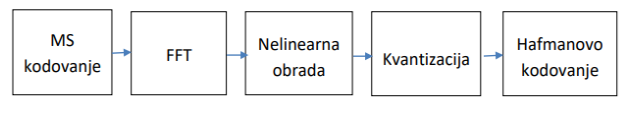
Implementiramo joint coding, gde je:

M[i] = (inputBufferL[i]+inputBufferR[i]) /2

S[i] = (inputBufferL[i]-inputBufferR[i]) /2

Ovo dovodi do bolje kompresije ukoliko su levi i desni kanal podjednaki.

Korisniku omogućavamo izbor kompresije.



Za kvantove 16 dobijam komprimovan zvuk uz primetan šum, a za parove kvantova L=12, R=8 i L=11, R=6 iz meni nepoznatih razloga dobijam skroz uništen signal šumom.

**Zadatak 4**

Modifikujemo naš projekat da bi dobili funkcionalnost kad je zadat broj kvantova jednak 0. Tad se aktivira specijalna implementacija za koju se u zavisnosti od opsega poziva kvantizacija sa 0, 8, 12 ili 6 bita.

Za kvantove R=0 i L=0 dobija se mala komprsija sa dobrim kvalitetom zvuka.

Za kvantove R=0 i L=8 dobija se veća kompresija sa dobrim kvalitetom zvuka.