



UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
Odsek za elektrotehniku i računarstvo
Institut za računarstvo i automatiku
Katedra za računarsku tehniku i računarske komunikacije

Arhitekture i algoritmi DSP II

Projektni zadatak:
Mixer NxM

Student: Aleksa Arsić

Mentor: Dejan Bokan

Novi Sad, 2018.

Sadržaj

1.	Opis zadatka.....	1
2.	Opis realizacije.....	2
3.	Ispitivanje i verifikacija	3

1. Opis zadatka

Potrebno je realizovati module 0, 1 i 2 na osnovu datog koda mixera $N \times M$, gde N predstavlja broj ulaznih, a M broj izlaznih kanala. Sam mixer svaki odbirak ulaznog signala množi sa članovima njegove strukture `mixerNxM_data`, `output_gain` i `gain_matrix[N][M]` i takvog ga predstavlja na 6 izlaznih kanala. Takođe moguće je zaobići mixer prosleđivanjem logički netačnog ulaznog parametra `enable`. Tada je signal preslikan onakav kakav je na ulazu na onoliko kanala od koliko se sastoji (maksimalno šest).

U datom rešenju predstavljeni su modeli 0, 1 i 2 u potpunosti, dok je model 3 delimično završen (do optimizacije asmeblerom). Počev od modela 0, `processing` funkcija se uspešno poziva i obavlja traženi posao bez deformacija signala. Razlike između modela 0 i modela 1 kroz sve test modele ne postoje. Razlike između modela 1 i modela 2, za sve test primere koji su dati, su na nivou bit-razlike, dok razlike između modela 2 i modela 3 ne postoje.

Rešenje ovog programa može da radi u pet različitih modova. Modovi predstavljaju kombinacije kanala koji su uključeni i na kojima će se pojaviti signal na izlazu.

Problem koji se provlaci kroz rešenje jeste problem izlaska van opsega koji se resava klipovanjem signala.

Ispitivanje programa vrši se putem `Test.bat` skripte, o kojoj će biti više reči u poglavlju 3. Ispitivanje i verifikacija.

2. Opis realizacije

Dati opis realizacije zadatka i navesti sve izvršene korake po modelima, kao i analizu potrošnje resursa (memorija za podatke, programska memorija, instrukcijski ciklusi, ...)

Model 0 se sastoji od datog referentnog koda koji je inicijalizovan dodatno napisanom funkcijom i pozvan kao blok obrade.

U modelu 1 optimizovale su se programske petlje, svi suvišni delovi koji su se mogli izmestiti izmešteni su van njih. Prešlo se iz običnog indeksiranja u aritmetiku pokazivača kako bi obrada bila brža, a suvišni delovi koda izmešteni su u common.h zaglavlje. Izmešteni su makroi i deklaracije globalnih promenljivih.

Prelaskom sa modela 1 na model 2 sa aritmetike u pokretnom zarezu prešlo se na aritmetiku u fiksnom zarezu i to vrlo uspešno, na nivou bit-razlike za sve date test ulaze. Pored toga ništa se ostalo nije promenilo.

U modelu 3 programski kod prilagodio se CCC kompajleru, kao što je urađeno i smeštanje po memorijskim zonama X i Y. Dodatno većina petlji je prepravljena kako bi CCC generisao asemblerski kod koji će koristiti hardverske petlje, u cilju bržeg izvršavanja programa. Ono što je dodatno potrebno jeste pregledati generisani asemblerski kod i dodato ga optimizovati, takođe u cilju smanjenja instrukcijskih ciklusa.

Iz sledeće slike vidi se da i dalje nakon nepotpune optimizacije funkcija processing() je najzahtevnija i zahteva najviše procesorskog vremena.

Symbols	Source File	Line Num...	Address	Base Time	Average Base ...	Cumulative Time	Calls
▼ root			0x0000	8	8.00	25526977	1
▼ main	..\..\src\main.c	121	0x0001	4834694	4834694.00	25526972	1
mixerNxM_processing	..\..\src\main.c	48	0x016b	17944796	12646.09	17944796	1419
mixerNxM_init	..\..\src\main.c	12	0x012b	416	416.00	416	1
_strcpy_x2x			0x0318	442	221.00	442	2
_cl_wavwrite_sendsample			0x0342	816768	6.00	816768	136128
_cl_wavwrite_open			0x033f	6	6.00	6	1
_cl_wavread_recvsample			0x0339	136224	6.00	136224	22704
_cl_wavread_open			0x0336	6	6.00	6	1
_cl_wavread_number_of_frames			0x0333	6	6.00	6	1
_cl_wavread_getnchannels			0x0330	6	6.00	6	1
_cl_wavread_frame_rate			0x032d	6	6.00	6	1
_cl_wavread_bits_per_sample			0x0327	6	6.00	6	1
> _div			0x02de	83721	59.00	1793616	1419

Slika 1. Profiling prozor CLIDE okruženja

3. Ispitivanje i verifikacija

Provera ispravnosti zadatka sastoji se od pokretanja test skripte Test.bat koja pokreće model 0, 1 i 2 za sve ulazne primere koji su dati i na osnovu njih, uz pomoć PCMCompare alata generiše .txt fajlove koji sadrže informacije o razlikama izlaza između modela 0 i 1, i modela 1 i 2, kao i modela 2 i 3. Dodatno skripta prosleđuje argumente modelima. I to redom: gain mode enable. Tekstualne fajlove koji se nalaze u OutCmp folderu lako je analizirati. Između modela 0 i 1 ne sme da postoji razlika. Između modela 1 i 2 dozvoljena razlika je maksimalno 1 bit, a između modela 2 i 3 razlika takođe ne sme postojati.

Moduli su ispitani za različite ulazne parametre (različit gain, kao i različiti modovi) i nije uočeno nikakvih odudaranja od gore pomenutih pravila.