

Realizacija algoritma kombinovanja kanala na Cirrus Logic DSP platformi

Cilj projekta

Realizovati kombinovanje kanala na osnovu date šeme i tabele. Za implementaciju koristiti razvojno okruženje dato na vežbama (Windows, CLIDE, Visual Studio).

Postavka i opis zadatka

1. Na osnovu šeme kombinovanja kanala prikazane na slici i tabele kontrola za dato kombinovanje kanala realizovati referentni C kod u aritmetici pokretnog zareza.
2. Obrada se vrši po blokovima. Jedan blok predstavlja 16 odbiraka signala. Na ulaz dolazi 2 kanala, a izlaz predstavlja do 8 kanala.
3. Omogućiti prosleđivanje naziva ulaznih i izlaznih datoteka kao i vrednosti kontrola preko parametara komandne linije.
4. Na osnovu referentnog C koda realizovati C kod prilagođen aritmetici Crystal DSP procesora, poštujući metodologiju izrade modula (od modela 1 do modela 3).
5. Testirati izlaze modela 0, 1, 2 i 3 koristeći date test vektore (za svaki od efekata dati su testni vektori u tabeli ispod - koristiti 2 zadata vektora i bar 1 proizvoljni vektor). Koristiti PCMCompare.exe alat za poređenje datoteka. Voditi računa da se pokriju sve karakteristične vrednosti korisničkih kontrola. Automatizovati proces testiranja upotrebom python test okruženja datog na vežbama. Smatra se da je test vektor prošao proveru ukoliko su izlazi iz referentnog koda bit-identični sa izlazima implementacije koda prilagođenog za prevođenje.
6. Izvršiti asemlerske optimizacije u Modelu 3 i profilisanje funkcije obrade (procenu utroška MIPS-a) pre i posle optimizacija. Utvrditi da je postignuta odgovarajuća optimizacija.
7. Integrisati rešenje u finalni model tj. u radno okruženje Cirrus Logic OS-a. Za integraciju preuzeti example_module, example_overlay i example_application projekte iz D1, VE-1.
8. Na kraju utvrditi da su izlazi iz finalnog modela bit-identični sa izlazima iz Modela 3.
9. Napisati prateću dokumentaciju u koju će biti po svakom modelu dokumentovano ukratko kakve su izmene napravljene i detaljno izložene metode na koji je svaki model testiran. Prikazati obavezno rezultate izlaza iz svakog modela, poređenja i utrošak MIPS-a i memorije.

Test vektori po zadatom efektu:

Effect	Stream 1	Stream 2
Filter	WhiteNoise	Freq_sweep
Tremolo	WhiteNoise	Amp_sweep
Mixer/Gains	2ch_lvl_amt_48k	Tone_L1k_R3k
Compressor/Expander	WhiteNoise	speech_2ch,
Distortion	Tone_L1k_R3k	funky_sample
Delay	speech_2ch	titanic_horn
Inverter	Tone_L1k_R3k	speech_2ch/funky_sample

Napomena: Za realizaciju efekata koristiti date referentne kodove iz “refCode” foldera projektnog zip fajla.

Rezultati rada

Nakon implementacije i testiranja realizovane programske podrške potrebno je isporučiti kod programa u elektronskoj formi na Canvas platformi u okviru Assignmenta 18.3. Rok za predaju projekta je 18. dan predmeta tj. **30.11.2022. godine, 23:59:59**.

Predati projekat je kompletan ukoliko sadrži:

1. Visual Studio projekte za Model 0, 1 i 2 (predati isključivo izvorne fajlove i .sln, .suo, .vcxproj, koji zauzimaju zajedno do nekoliko desetina ili stotina KB-a, a **NE kačiti rezultate prevođenja Debug, Release foldere, Testne vektore i sl.,**) - imenovati modele prema sličnoj konvenciji korištenoj na vežbama (“imeAlgoritmalliEfekta_modelBrojN”).
2. Standalone ULD projekat sa kodom Modela 3
3. DSP application, Overlay i Code Module projekte koji sadrže integraciju Modela 3 u programsko okruženje
4. Skriptu za automatizaciju testiranja i poslednji ažurni log (izlaz iz skripte sa testiranim modelima).
5. Dokumentaciju o radu koja sadrži 1-5 stranica Word dokumenta.