



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У
НОВОМ САДУ



АЗРА ОБУЋА

**ЈЕДНО РЕШЕЊЕ ПРОТОКОЛА ЗА
ОПИС АУДИО ОБЈЕКТА НА
ПЛАТФОРМИ СА ОГРАНИЧЕНИМ
РЕСУРСИМА**

МАСТЕР РАД

Нови Сад, 2017



КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:	
Идентификациони број, ИБР:	
Тип документације, ТД:	Монографска документација
Тип записа, ТЗ:	Текстуални штампани материјал
Врста рада, ВР:	Дипломски – мастер рад
Аутор, АУ:	Азра Обућа
Ментор, МН:	Др Јелена Ковачевић
Наслов рада, НР:	Једно решење протокола за опис аудио објеката на платформи са ограниченим ресурсима
Језик публикације, ЈП:	Српски / латиница
Језик извода, ЈИ:	Српски
Земља публиковања, ЗП:	Република Србија
Уже географско подручје, УГП:	Војводина
Година, ГО:	2017
Издавач, ИЗ:	Ауторски репринт
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6
Физички опис рада, ФО: (поглавља/страна/ цитата/табела/слика/графика/прилога)	7/49/17/11/23//
Научна област, НО:	Електротехника и рачунарство
Научна дисциплина, НД:	Рачунарска техника
Предметна одредница/Кључне речи, ПО:	ДСП, објекти, мета подаци
УДК	
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад
Важна напомена, ВН:	
Извод, ИЗ:	<p>У последњих десет година дошло је до великих промена како у количини тако и у начину коришћења аудио и видео садржаја. Једна од кључних карактеристика новијих технологија је тачна просторна репродукција звучног сигнала, независно од циљне поставке звучника. Приступ коришћење аудио објеката је управо то и омогућила. Неопходни пратећи подаци објектног аудио садржаја - мета подаци, су одговорни за ову адаптацију аудио садржаја циљној конфигурацији звучника, динамичко објектно рендеровање као и корисничку интерактивност. У оквиру овог рада описано је једно решење протокола за опис аудио објеката на платформи са ограниченим ресурсима, односно ДСП (eng. <i>Digital Signal Processor</i>) платформи.</p>
Датум прихватања теме, ДП:	
Датум одбране, ДО:	
Чланови комисије, КО:	Председник: Др Илија Башичевић
	Члан: Др Иван Мезеи
	Члан, ментор: Др Јелена Ковачевић
	Потпис ментора



KEY WORDS DOCUMENTATION

Accession number, ANO :	
Identification number, INO :	
Document type, DT :	Monographic publication
Type of record, TR :	Textual printed material
Contents code, CC :	Master Thesis
Author, AU :	Azra Obuća
Mentor, MN :	Dr Jelena Kovačević
Title, TI :	One solution of object audio description information on DSP platform
Language of text, LT :	Serbian
Language of abstract, LA :	Serbian
Country of publication, CP :	Republic of Serbia
Locality of publication, LP :	Vojvodina
Publication year, PY :	2017
Publisher, PB :	Author's reprint
Publication place, PP :	Novi Sad, Dositeja Obradovica sq. 6
Physical description, PD : <small>(chapters/pages/ref./tables/pictures/graphs/appendixes)</small>	7/49/17/11/23//
Scientific field, SF :	Electrical Engineering
Scientific discipline, SD :	Computer Engineering, Engineering of Computer Based Systems
Subject/Key words, S/KW :	DSP, object, metadata
UC	
Holding data, HD :	The Library of Faculty of Technical Sciences, Novi Sad, Serbia
Note, N :	
Abstract, AB :	<p>In last decade, there have been a major changes in both quality and quantity of audio and video content. One of the key features of newer technology is accurate spatial reproduction of the sound signal, regardless of the speaker configuration. Using concept of 3D object enabled it. Necessary data that describes object audio content – metadata, is the thing that provides this adaptation of audio content to any speaker configuration, dynamic object rendering and user interactivity. In this work is described one solution of object audio description information on DSP platform.</p>
Accepted by the Scientific Board on, ASB :	
Defended on, DE :	
Defended Board, DB :	President: Dr Ilija Bašičević
	Member: Dr Ivan Mezei
	Member, Mentor: Dr Jelena Kovačević
	Mentor's sign

Zahvalnost

Zahvaljujem se razvojno-istraživačkom institutu “RT-RK” iz Novog Sada.

SADRŽAJ

1. Uvod.....	8
2. Pregled sistema za reprodukciju zvuka.....	10
2.1 Koncept 3D objekta	12
2.1.1 Kanali.....	13
2.1.2 Objekti	13
2.1.3 Ambisonici.....	14
2.2 Audio sistem zasnovan na objektnoj tehnologiji	15
2.3 VBAP (Amplitudsko pomeranje bazirano na vektorima).....	17
2.3.4 Koordinatni sistemi	19
2.3.4.1 Kartezijev koordinatni sistem	19
2.3.4.2 Sferni koordinatni sistem	20
2.4 AVR (Audio/video prijemnik).....	21
2.5 Soundbar	22
3. Koncept rešenja.....	24
3.1 Protokol za opis audio objekata	24
4. Implementacija sistema za reprodukciju zvuka na bazi 3D objekata	31
4.1 Ciljna platforma	31
4.2 Metodologija razvoja sistema na DSP platformi	32
4.1.1 Radni okvir procesora CS49844	35
4.1.2 Moduli implementiranog sistema	36
4.3 Protokol za prenos podataka između jezgara.....	40
5. Testiranje i verifikacija	42
6. Zaključak	48

7. Literatura.....	49
--------------------	----

SPISAK SLIKA

Slika 2.1 Raspored zvučnika standardizovanog 5.1 sistema.....	11
Slika 2.2 Prikaz konfiguracija zvučnika 7.1 i 5.1 sistema	12
Slika 2.3 Sistem 22.2.....	12
Slika 2.4 Primer jedne putanje pokretnog objekta	14
Slika 2.5 Enkodovane funkcije ambisonika prvog reda.....	15
Slika 2.6 Enkodovane funkcije ambisonika drugog reda.....	15
Slika 2.7 Blok šema audio podistema na bazi objektnog renderovanja.....	16
Slika 2.8 Definisane vektora u stereo konfiguraciji.....	18
Slika 2.9 Konfiguracija trodimenzionalnog VBAP-a. Zvučnici formiraju trougao u kojem virtuelni izvor može biti pozicioniran	18
Slika 2.10 Kartezijev koordinatni sistem	20
Slika 2.11 Sferni koordinatni sistem	20
Slika 2.12 Prikaz izgleda i blok dijagram prijemnika audio-video signala zasnovanog na CS49844 procesoru	22
Slika 2.13 Ilustracija soundbar sistema.....	23
Slika 2.14 Ilustracija dobijanja surround efekta.....	23
Slika 4.1 Blok dijagram CS49844 čipa	32
Slika 4.2 Raspodela modula implementiranog sistema na jezgra ciljne platforme.....	37
Slika 4.3 Ping-Pong struktura memorijskog niza.....	38
Slika 4.4 Blok šema CS49844 procesora i periferija na čipu.....	41
Slika 5.1 Primer generisanog testnog slučaja za BBT alat (.tst datoteka).....	43
Slika 5.2 BBT aplikacija nakon pokretanja	44
Slika 5.3 Blok dijagram izvršavanja jednog BBT testa	44

Slika 5.4 Ispitno okruženje.....	45
Slika 5.5 Prikaz ishoda izvršavanja testova	46

SPISAK TABELA

Tabela 3.1 Prvi element meta podataka	24
Tabela 3.2 Elementi grupe meta podataka za prezentaciju objekta	25
Tabela 3.3 Elementi grupe meta podataka za prostornu prezentaciju.....	26
Tabela 3.4 Elementi grupe meta podataka za signalizaciju posebnih opcija za reprodukciju	27
Tabela 3.5 Elementi grupe meta podataka za glasnoću	28
Tabela 3.6 Elementi grupe meta podataka za interaktivnost	28
Tabela 3.7 Elementi meta podataka za opisivanje audio objekata.....	30
Tabela 4.1 Količina raspoloživih resursa ciljne platforme.....	32
Tabela 4.2 Utrošak resursa DSPA jezgra.....	38
Tabela 4.3 Utrošak resursa DSPB jezgra	40
Tabela 4.4 Utrošak resursa DSPC jezgra	40

SKRAĆENICE

DSP - Digitalni signal procesor, *Digital Signal Processor*

VBAP - Amplitudsko pomeranje na bazi vektora, *Vector Base Amplitude*

Panning

DAW - Muzička stanica, *Digital Audio Workstation*

MAC - Instrukcija za množenje i akumuliranje rezultata, *Multiply and*

Accumulate

MIPS - Milion instrukcija u sekundi, *Milion instructions per second*

SRS - Jedinica sa aritmetičko pomeranje, zaokruživanje i saturaciju Crystal

DSP familije procesora, *Shift Round Saturate*

DMA - Direktnan pristup memoriji, *Direct Memory Access*

SPDIF - Sony Philips digitalni sprežni podsistem, *Sony Philips Digital InterFace*

HDMI - Multimedijalni sprežni podsistem visoke definicije, *High Definition*

Multimedia Interface

MIF - Sprežni podsistem, *Module Interface*

MCT - Tabela pozivanja modula, *Module Call Table*

MCV - Tabela kontrolnih vektora modula, *Module Control Vector*

OS - Operativni sistem, *Operating System*

FIFO - Prvi ulaz, prvi izlaz, *First In First Out*

AVR - Audio/Video prijemnik, *Audio Video Receiver*

BBT - Ispitivanje metodom crne kutije, *Black Box Testing*

1. Uvod

U okviru ovog rada realizovan je protokol za opis audio objekata na platformi sa ograničenim resursima, odnosno digitalnom signal procesoru DSP (*eng. Digital Signal Processor*). Polazna osnova implementacije je referentni algoritam realizovan u C programskom jeziku. Kako su zahtevi podrazumevali rešenje na jednom procesoru, za implementaciju izabran je četvorोजezgarni CS49844 procesor Crystal DSP familije kompanije *Cirrus Logic*. Odabirom ciljne platforme nameću se ograničenja kako u pogledu memorije, tako i u pogledu procesorskog vremena. Tako se na samom početku modifikuje polazni algoritam uvođenjem različitih algoritamskih optimizacija u cilju što veće efikasnosti na DSP platformi. Da bi se zadovoljili resursi potrebni za implementaciju datog sistema, bilo je neophodno iskoristiti tri od četiri jezgra procesora i samim tim prilagoditi algoritam višejezgarnom okruženju. Ovim se javlja potreba i za komunikacijom između jezgara, pa je razvoj podrazumevao i rad na toku i prenosu podataka između jezgara.

Ova tehnologija zasnovana na audio objektima ima primenu u bioskopima, kućnim bioskopima, prvenstveno se ugrađuje u audio/video prijemnike AVR (*eng. Audio Video Receiver*), kao i Soundbar zvučnike.

Ubrzani napredak uređaja potrošačke elektronike poslednjih godina nije zaobišao ni oblast audio sistema. Razvoj brojnih uređaja, mobilnih telefona, 3D televizora i drugih, nameće potrebu i za razvojem u načinu emitovanja audio sadržaja. Nove generacije audio tehnologije prate ove promene. Jedna od ključnih osobina novije tehnologije jeste mogućnost prenosa celokupne audio scene umesto njene prezentacije na određenom mestu u prostoru. To znači da audio sadržaj više nije vezan za konkretnu postavku zvučnika, te da se kvalitet zvuka neće umanjiti promenom konfiguracije zvučnika. Pristup korišćenja audio objekata za prenos audio sadržaja je ovo omogućio.

Rad je organizovan u 7 poglavlja. U drugom poglavlju rada date su teorijske osnove koje podrazumevaju kratak istorijski razvoj audio tehnologije, uvod u audio tehnologiju baziranu na 3D objektima, kratak opis realizovanog audio sistema, jednu od metoda za lokalizaciju tačke u prostoru – amplitudsko pomeranje bazirano na vektorima VBAP (*eng. Vector Base Amplitude Panning*), kao i pregled koordinatnih sistema koje služe za opis pozicije u prostoru. Navedeni su i neki od uređaja potrošačke elektronike gde se ovakav sistem koristi. U trećem poglavlju izneto je jedno rešenje protokola za opis audio objekata. U četvrtom poglavlju date su tehničke specifikacije ciljne platforme, detalji implementacije audio sistema koji koristi opisani protokol. Peto poglavlje sadrži rezultate testiranja. U šestom poglavlju ukratko je sažeto sve što je u okviru ovog rada urađeno. Poslednje, sedmo poglavlje, sadrži spisak korišćene literature prilikom izrade ovog rada.

2. Pregled sistema za reprodukciju zvuka

Prostorna zvučna reprodukcija počinje sa stereo formatom, odnosno dvo-kanalnom reprodukcijom koju je uveo Blumlein 1931. godine [1]. U isto vreme Harvey Fletcher u Bellovim laboratorijama eksperimentisao je sa usmerenom reprodukcijom, koristeći veliki broj omnidirekcionih mikrofona, gde je svaki povezan sa jednim pojačavačem za odgovarajući zvučnik u slušnoj sobi. Tražeći optimalan broj zvučnika za dovoljno dobar kvalitet reprodukcije, došao je do četiri kanala, a Steinberg i Snow 1934. godine [2] pokazuju da se koristeći tri kanala takođe dobijaju dobri rezultati.

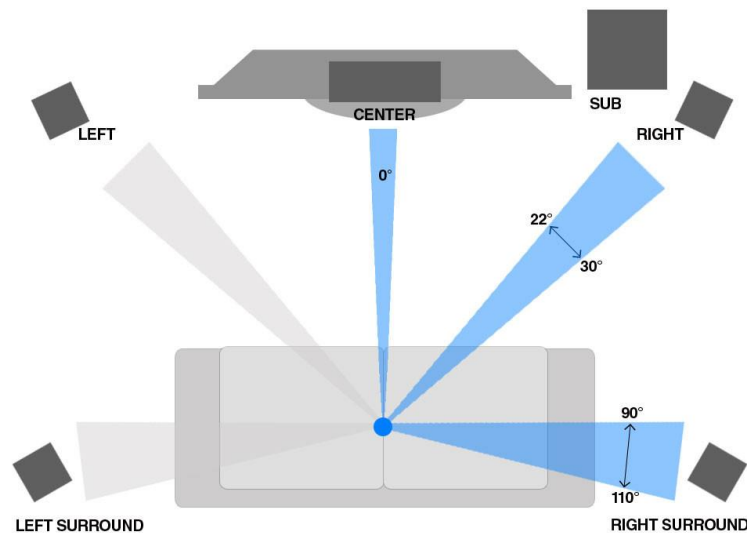
Dostupnost ovakvih sistema za kućnu upotrebu pojavila se 1968. godine kada je Peter Scheiber predstavio kvadrofonski (četvorokanalni) sistem. Ipak, ovakav sistem se nije popularizovao iz brojnih razloga: slaba standardizacija sistema, skupa oprema, potreba za dodatnom opremom, nedovoljno razvijena tehnologija.

Sledeći pomak u audio reprodukciji došao je sa Dolby stereo sistemom, 1975. godine. Uključivao je tri prednja (eng. *front*) kanala, odnosno stereo konfiguraciju sa dodatnim centralnim kanalom i jedan zadnji (eng. *rear*) kanal.

Takođe u to vreme, tačnije 1973. Michael Gerzon razvija Ambisonic [3], sistem koji je u potpunosti bio u mogućnosti da reprodukuje prostorni zvuk. Osnovna ideja je bila omogućiti snimanje sa specijalnim mikrofonom, koji je trebao da što kvalitetnije snimi potpunu zvučnu sliku pomoću 4 kanala (npr. sa *Soundfield* mikrofonom). Kao rezultat istraživanja, došlo se do bitnih otkrića: Ambisonic sistem je radio nezavisno od postavke zvučnika, na način da su se pozicije implementirale matrično u dekodneru, što se pokazalo dosta praktično. Dalje, efekat prostornosti i lokalizacija su bili stabilni na jako širokom prostoru, tako da se i izvan mreže zvučnika mogla osetiti prostorna zvučna slika koja dolazi iz sistema. Ambisonics je ponudio nešto više od prethodnih tehnika - reprodukciju informacije o visini. Ipak doživeo je loš

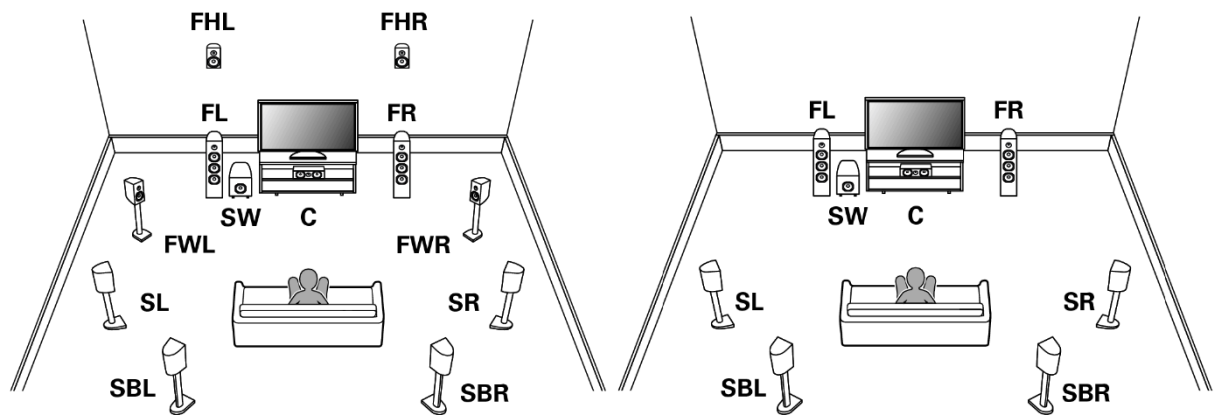
komercijalni uspeh zbog visokih troškova. Cela implementacija uključuje dodatni dekođer, i sam način snimanja i obrade ovakvog sistema je dosta složen.

Dolby stereo kao sistem sa najvećim komercijalnim uspehom se razvija i 1991. godine dovodi do standardizovanog 5.1 sistema [4] (broj pre tačke predstavlja broj zvučnika punog frekvencijskog opsega, dok broj nakon tačke predstavlja broj niskofrekvencijskih zvučnika), prikazanog na Slika 2.1. Sastoji se iz 5 diskretnih kanala, 3 prednja, 2 zadnja i jednog niskofrekventnog kanala.



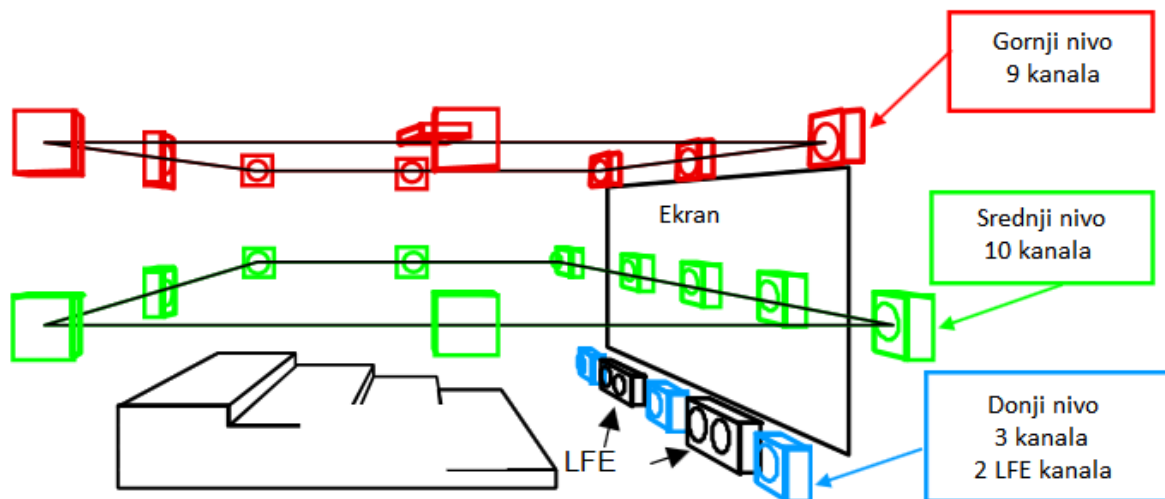
Slika 2.1 Raspored zvučnika standardizovanog 5.1 sistema

U cilju realnijeg zvuka koji dolazi do slušaoca, uvodi se i treća dimenzija dodavanjem podignutih ili spuštenih zvučnika sistemu. Thomlinson Holman 1999. godine razvija 10.2 sistem sa dodatna dva prednja kanala i samo jednim dodatnim zadnjim kanalom. To je zato što je ljudska prostorna percepcija zvuka mnogo preciznija u frontalnom delu, nego iza. Osim toga, on dodaje dubinu zvuku time što izdiže dva frontalna kanala za 45° u odnosu na ravan u kojoj se nalazi slušalac. Ovakav sistem je značajno povećao kvalitet reprodukcije, što dovodi do razvijanja i drugih sistema koji uvode treću dimenziju u postavci zvučnika. Neke od konfiguracije ovih 3D sistema zvučnika, 7.1 i 5.1 su prikazane na Slika 2.2.



Slika 2.2 Prikaz konfiguracija zvučnika 7.1 i 5.1 sistema

Na Slika 2.3 prikazan je 22.2 sistem, razvijen od strane Japanske radio-televizijske korporacije [5].



Slika 2.3 Sistem 22.2

I dok se 3D sistem zvučnika pokazao kao kvalitetniji u postizanju prostornog kvaliteta zvuka od 2D sistema, potrebno je obezbediti i interoperabilnost između proizvođača audio sadržaja, proizvođača uređaja i potrošača. Jedno od rešenja za univerzalno i efikasno kodovanje i pružanje kvalitetnog prostornog zvučnog sadržaja jeste 3D objektno orijentisano renderovanje. Osim što je u stanju da obezbedi visoki kvalitet prostornog zvuka, ima mogućnost i da objedini mnoštvo 3D audio formata prihvatajući sve ove formate na ulazu i nudeći visokokvalitetnu reprodukciju na izlazu za veliki broj audio formata od 22.2 pa do nižih, 5.1, stereo, zavisno od originalnog enkodovanog formata. Na ovaj način se postiže i kompatibilnost novih formata sa prethodnim uređajima.

2.1 Koncept 3D objekta

U ovom poglavlju dat je opis različitih formata koji se koriste u reprodukciji zvuka. To su kanalni, objektni i ambisonic.

2.1.1 Kanali

Jedan od najzastupljenijih načina reprodukcije prostornog zvuka jesu tzv. kanalni signali, gde svaki od njih predstavlja set talasnih oblika predviđenih za unapred tačno utvrđenu konfiguraciju zvučnika.

U filmu, zvuk koji publika čuje kombinacija je velikog broja pojedinačnih zvučnih elemenata, dijaloga snimljenih na setu, muzike snimljene u studiju itd. Svi ovi elementi distribuirani su na nekoliko audio zapisa i snimljeni u studiju koristeći ogromne miksete i muzičke stanice DAW (*eng. Digital Audio Workstation*) - elektronski sistem dizajniran za snimanje, uređivanje i reprodukciju digitalnog signala, u skladu sa standardizovanom konfiguracijom zvučnika. Krajnji rezultat ovog procesa jeste skup kanala, gde svaki od njih predstavlja sadržaj jednog od zvučnika [6].

Danas su različite konfiguracije u upotrebi, od dvokanalnog stera 2.0, višekanalnog 5.1, 7.1, do trodimenzionalnih konfiguracija sa različitim brojem zvučnika koji se nalaze iznad ili ispod horizontalne ravni u kojoj se nalazi slušalac. Na Slika 2.2 i Slika 2.3 prikazani su neki od ovih sistema, 7.1, 9.1 i 22.2 sistem [7].

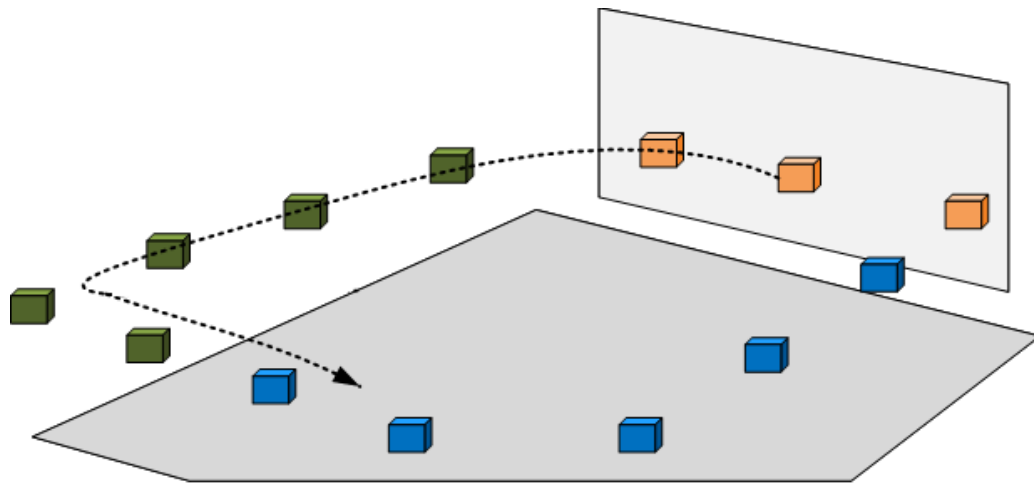
Kanalno orijentisani zvučni materijal je dosta izučavan i dugo u upotrebi, međutim kako je stvoren za specifičnu konfiguraciju zvučnika, suštinski je ograničen, nema potrebne informacije za pružanje potpunijeg i bogatijeg zvuka ukoliko se sistem proširi. Dakle reprodukcija na drugačijoj konfiguraciji zvučnika dovodi do umanjenog kvaliteta zvuka.

2.1.2 Objekti

Prostorni zvuk se danas može opisati i virtuelnim izvorima tzv. objektima koji su pozicionirani na određenoj lokaciji u prostoru. Ove pozicije mogu da se menjaju tokom vremena, nisu predefinisane pre reprodukcije zvučnog signala već se u toku nje raspoređuju (*eng. rendering*) na zvučnike ciljne konfiguracije.

Kako objekti nisu predefinisani za određene pozicije zvučnika, odgovarajućim algoritmom se raspoređuje na ciljne pozicije zvučnika. Da bi se uspešno objekat pozicionirao, osim zvučnog signala, mora sadržati i određene informacije, kao što su njegove koordinate, pojačanje itd. Dakle audio podacima pridruženi su i odgovarajući opisni podaci.

Zbog mogućnosti vremenski promenljivih pozicija objekata, moguće je i modelovati različite zvučne efekte, npr. avion koji preleće iznad slušaoca, cvrkut ptica i sl. Na Slika 2.4 dat je primer jedne trajektorije objekta, gde se njegov položaj u prostoru menja sa vremenom, od centralnog zvučnika, preko levog, levog bočnog zvučnika, zatim preko zadnjih zvučnika do zadnjeg desnog ugla. Tom putanjom i slušalac čuje zvuk koji dati objekat nosi.



Slika 2.4 Primer jedne putanje pokretnog objekta

Dakle glavne prednosti objektnog pristupa su:

- kvalitet zvučnog signala se ne gubi promenom konfiguracije zvučnika na kojoj se reprodukuje,
- mogućnost modelovanja zvučnih pomeraja, odnosno različitih zvučnih efekata kao što je npr. avion koji preleće iznad slušaoca.

2.1.3 Ambisonici

Ambisonic sistem prvog reda razvio je Gerzon [8] u sedamdesetim godinama prošlog veka i nazvao ga B-format. Zvučna slika snima se specijalnim mikrofonom (*Soundfield* mikrofonom) ili sa jednim omnidirekcionim mikrofonom i tri gradijentna mikrofona, od kojih je svaki postavljen u jedan od tri zamišljena koordinatna sistema (X, Y, Z). Princip je sledeći: od 4 kanala koja se snimaju, jedan je mono (W), dobijen preko omnidirekcionog mikrofona (ili superpozicijom svih kapsula *Soundfield* mikrofona), a ostala tri su snimljena kao razlike pritisaka na odgovarajućim osama, na primer kanal X je razlika u gradijentu „ispred“, odnosno „iza“ referentne tačke na x-osi. Analogno je i za kanale Y i Z. Daljim dekodovanjem snimka moguće je za odgovarajuću postavku zvučnika dobiti reprodukciju koja odgovara zamišljenoj: očuvane su lokalizacija i karakteristike snimka i prostorije.

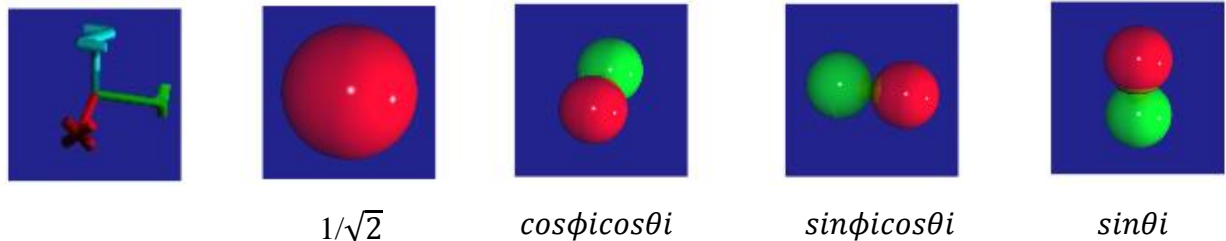
$$W = \frac{1}{k} \sum_{i=1}^n S_i \left[\frac{1}{\sqrt{(2)}} \right] \quad \text{omnidirekciona informacija} \quad (10)$$

$$X = \frac{1}{k} \sum_{i=1}^n S_i [\cos\phi_i \cos\theta_i] \quad \text{informacija x-pravca} \quad (11)$$

$$Y = \frac{1}{k} \sum_{i=1}^n S_i [\sin\phi_i \cos\theta_i] \quad \text{informacija y-pravca} \quad (12)$$

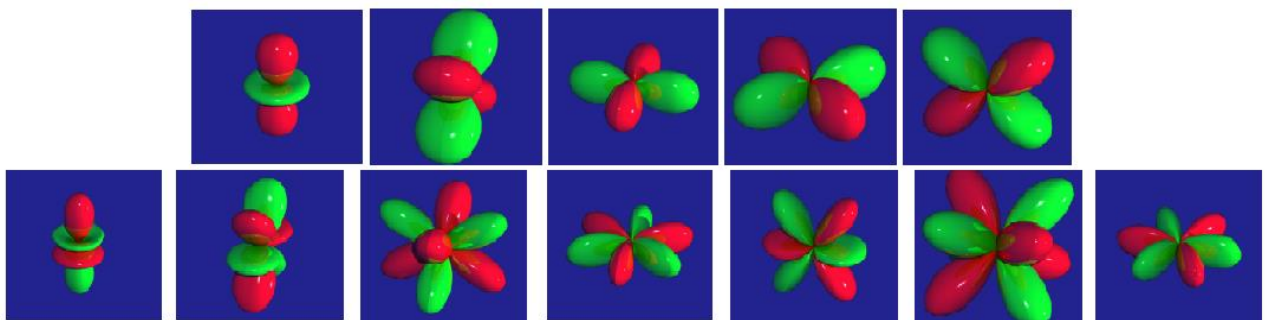
$$Z = \frac{1}{k} \sum_{i=1}^n S_i [\sin\theta_i] \quad \text{informacija z-pravca} \quad (13)$$

Fizički posmatrano, ovo predstavlja dekompoziciju zvučnog polja po sfernim koordinatama. Na Slika 2.5 prikazani su sferni harmonici nultog i prvog reda koje koriste ambisonici prvog reda.



Slika 2.5 Enkodovane funkcije ambisonika prvog reda

Princip ovih sistema proširio se i na više redove, povećavajući pritom prostor u kojem se zvučno polje precizno reprodukuje kao i ukupni kvalitet lokalizacije. Međutim, sve većim redom ambisonic snimci rastu a sa njima i broj potrebnih kanala nužnih za reprodukciju. Tako je za drugi red potrebno 5 dodatnih kanala, a za treći red još dodatnih 7 kanala [8].



Slika 2.6 Enkodovane funkcije ambisonika drugog reda

Na Slika 2.6 vidimo sferne harmonike drugog i trećeg reda. Svakim novim redom harmonika, broj kanala potreban za reprodukciju raste po kvadratnoj zavisnosti:

$$N = (M + 1)^2 \quad (14)$$

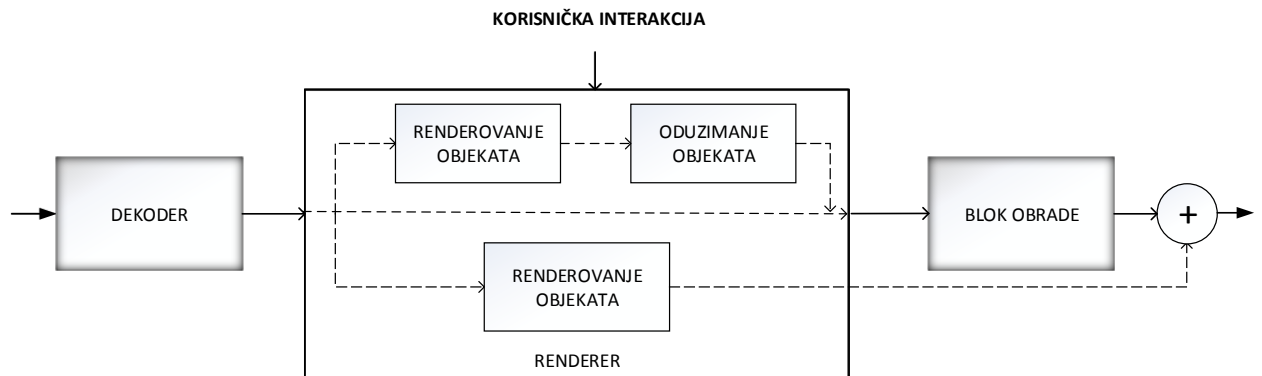
gde je N broj kanala i M red harmonika.

2.2 Audio sistem zasnovan na objektnoj tehnologiji

Na Slika 2.7 prikazana je pojednostavljena blok šema implementiranog audio sistema uređaja za reprodukciju na bazi objekata.

Enkodovani podaci se prenose bitskim tokom do dekodera, koji u zavisnosti od vrste kompresije iščitava kompresovane audio podatke iz bitskog toka. Koji objekti ili grupe objekata

će se renderovati odlučuje dekodler koji će dekodovati samo onaj sadržaj neophodan za reprodukciju na ciljnoj konfiguraciji. Podržana je i kompatibilnost sa drugim formatima, tako da je moguće na strani dekodera odbaciti objekte i reprodukovati kanalni zapis na predefinisane zvučnike. Meta podaci enkodovani su zajedno sa zvučnim zapisom u bitskom toku i prenose se u lancu enkoder – dekodler. Dinamički se menjaju i prenose na svakih n audio odbiraka.



Slika 2.7 Blok šema audio podistema na bazi objektnog renderovanja

U bloku renderer se obavlja 3D objektno audio renderovanje. Njegov primarni zadatak je da doda ili oduzme postojeće audio objekte u ili iz osnovnog zvučnog zapisa koji se prenosi bitskim tokom. Za potrebe kompatibilnosti sa drugim uređajima, moguće je u kanalnom audio sadržaju umiksati objektno audio podatke kako bi bili korišćeni od strane uređaja koji nemaju mogućnost renderovanja audio objekata. Iz tog razloga, može biti potrebno prvo oduzeti objekte iz audio sadržaja, kako nakon objektnog renderovanja ne bi bili dvostruko uključeni u izlazni reprodukovani zvuk. Da bi se tačno uklonili, moraju biti renderovani za ciljnu konfiguraciju zvučnika koja odgovara ulaznoj kanalnoj konfiguraciji uključenoj u zvučni zapis.

Svaki objekat dolazi do renderer bloka sa odgovarajućim meta podacima koji diktiraju njegovu poziciju u prostoru, faktore skaliranja itd. Renderer vrši mnogobrojne kalkulacije čiji je krajnji rezultat niz amplituda sa kojima treba primeniti objekat na svaki od zvučnika. Proračuni su bazirani na prezentaciji sfernog koordinatnog sistema. Ako se prostorija u kojoj su smešteni zvučnici proglašeni kvadrom, onda se može smatrati da se unutar renderera ovaj kvadar transformiše u sferu i nazad u kvadar, tako da je to od strane krajnjeg korisnika neprimetno.

Podržano je renderovanje i izvora zvuka koji se mogu posmatrati kao tačke u prostoru i onih izvora koji imaju i svoju visinu i širinu, odnosno koji se kreću.

Renderovanje koristi metodu amplitudskog pomeranja baziranog na objektima – VBAP metodu (*eng. Vector Base Amplitude Panning*) opisanu u narednom poglavlju.

Moguća je i interakcija sa korisnikom, gde korisnik može uticati na određene atribute objekata. U opisanj implementaciji moguća je promena samo faktora pojačanja.

U bloku audio obrade mogu se naći različite funkcionalnosti. U slučaju kada fizički aranžman zvučnika za reprodukciju ima manje kanala nego što ima dekodovanih kanala, dekodovani audio će biti pretvoren ovim blokom tako da odgovara sistemu reprodukcije. To podrazumeva sve izlaze, uključujući i slušalice. Postoje matrice koeficijenata kojima će se ulazni signal množiti i na taj način se od ulazne npr. 5.1 konfiguracije dobija izlazni 2.0 (stereo) format. Na isti način, opet primenjujući odgovarajuće matrice koeficijenata na ulazni signal, on se prilagođava različitim zvučnicima na izlazu u odnosu na one za koje je bio namenjen dekodovani audio, ukoliko za to ima potrebe.

2.3 VBAP (Amplitudsko pomeranje bazirano na vektorima)

VBAP je metoda za pozicioniranje izvora zvuka u proizvoljnim pravcima u dvodimenzionalnim i trodimenzionalnim sistemima sa proizvoljnim brojem zvučnika. Nezavisna je od položaja zvučnika u sistemu i pruža maksimalnu moguću preciznost za datu konfiguraciju zvučnika.

U dvodimenzionalnim konfiguracijama zvučnika, odnosno sistemima sa dva ili više zvučnika smeštenih u istoj, najčešće horizontalnoj ravni sa slušaocem, biraju se po dva zvučnika u paru na kojima će se primeniti audio signal u određenom trenutku i nazivaju se aktivni luk. Svaki od zvučnika može biti deo dva aktivna luka. Ako su jedinični vektori l_1 i l_2 , usmereni od pozicije slušaoca do pozicije zvučnika (Slika 2.8), vektor zvučnog izvora p može se opisati kao njihova linearna kombinacija:

$$p = g_1 l_1 + g_2 l_2 \quad (1)$$

gde su g_1 i g_2 faktori pojačanja datih zvučnika. Ako ove vektore prikažemo u matricnom obliku

$$g = [g_1 \ g_2]^T, \quad L_{12} = [l_1 \ l_2], \quad (2)$$

tada se pojačanja mogu izraziti na sledeći način:

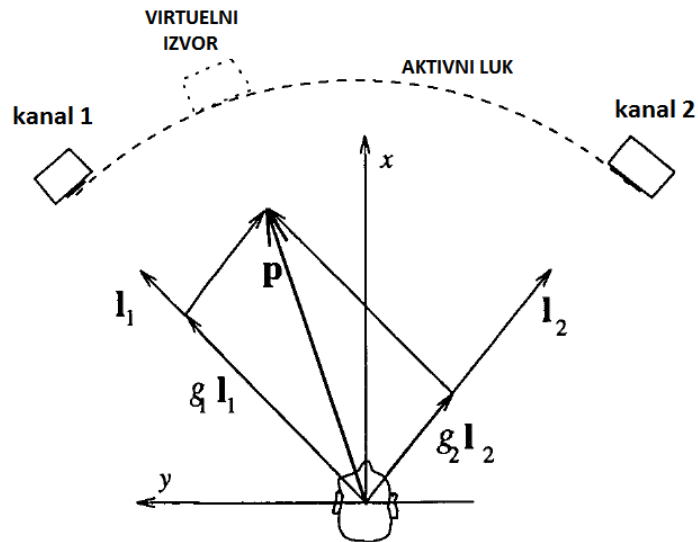
$$g = p^T L_{12}^{-1}. \quad (3)$$

Snaga zvuka je konstantna vrednost, što se može zapisati u sledećem obliku:

$$g_1^2 + g_2^2 = C. \quad (4)$$

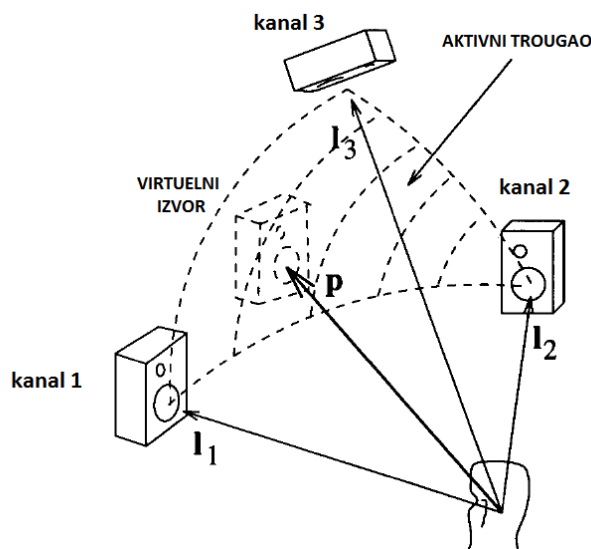
Konstanta C je pozitivna vrednost i može se posmatrati kao kontrola jačine zvuka koji se reprodukuje [9].

Dakle ova pojačanja nakon normalizacije jesu pojačanja signala kojima će se oni primeniti na odgovarajući zvučnik.



Slika 2.8 Definisanje vektora u stereo konfiguraciji

Trodimenzionalni VBAP koristi se za trodimenzionalne sisteme, odnosno sisteme gde svi zvučnici ne moraju da se nalaze u istoj horizontalnoj ravni, već i iznad i ispod nje. Na ovaj način omogućena je lokalizacija zvučnog signala i po vertikali. U ovom slučaju biraju se površine određene sa po tri zvučnika na kojima se izvor zvuka može nalaziti. Dva iz iste horizontalne ravni, dok je treći zvučnik van nje. Ovakva trojka naziva se aktivni trougao. Svaki od zvučnika može pripadati većem broju ovakvih trojki. Aktivni trouglovi ne smeju da se seku međusobno i treba ih izabrati tako da se postigne maksimalna moguća tačnost lokalizacije u svakom pravcu. Jedan ovakav sistem prikazan je na Slika 2.9 [10].



Slika 2.9 Konfiguracija trodimenzionalnog VBAP-a. Zvučnici formiraju trougao u kojem virtuelni izvor može biti pozicioniran

Vektor izvora zvuka se definiše kao linearna kombinacija vektora tri zvučnika l_1 , l_2 i l_3

$$p = g_1 l_1 + g_2 l_2 + g_3 l_3. \quad (5)$$

Faktori pojačanja izvode se iz prethodne jednačine:

$$g = p^T L_{123}^{-1}, \quad (6)$$

gde su

$$g = [g_1 \ g_2 \ g_3]^T, \quad L_{123} = [l_1 \ l_2 \ l_3]. \quad (7)$$

Kako je snaga zvuka konstanta, analogno dvodimenzionalnom sistemu može se zapisati:

$$g_1^2 + g_2^2 + g_3^2 = C. \quad (8)$$

Nakon normalizacije:

$$g_{normalizovano} = \frac{\sqrt{C}}{\sqrt{(g_1)^2 + (g_2)^2 + (g_3)^2}} \quad (9)$$

ova pojačanja koriste se kao umnošci signala prilikom njihove primene na odgovarajuće zvučnike.

VBAP ima tri važna svojstva:

1. Ako je izvor zvuka lociran u istom pravcu kao i neki od zvučnika, signal ostaje reprodukovan samo na tom zvučniku, što obezbeđuje maksimalnu oštrinu zvuka.
2. Ako se izvor zvuka nalazi na liniji koja povezuje dva zvučnika, zvuk se primenjuje samo na taj par, prateći tangentni zakon. Faktor pojačanja trećeg zvučnika je nula.
3. Ako se izvor nalazi u centru aktivnog trougla, faktori pojačanja za svaki od zvučnika su jednaki.

Ove osobine ukazuju na to da VBAP reprodukuje izvore zvuka što oštrije i preciznije je moguće sa datom konfiguracijom zvučnika.

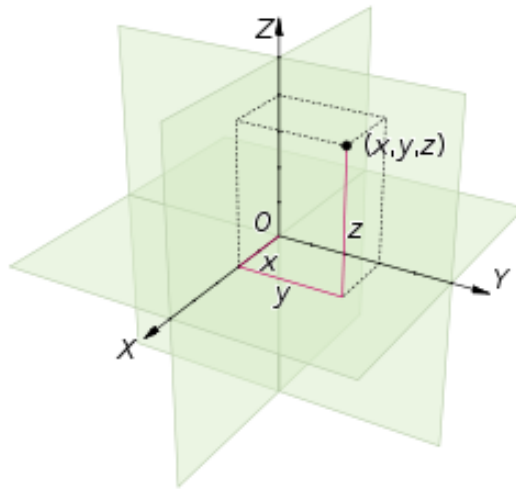
2.3.4 Koordinatni sistemi

Da bi se opisala i lokalizovala pozicija tačke u prostoru prvenstveno je potrebno izabrati odgovarajući koordinatni sistem. U nastavku dat je opis dva sistema, Kartezijev i sferni koordinatni sistem.

2.3.4.1 Kartezijev koordinatni sistem

Kartezijev (pravougaoni, Dekartov) trodimenzionalni koordinatni sistem prikazan na Slika 2.10, određuje položaj tačke u prostoru i definisan je centrom koordinatnog sistema O , u kojem se seku tri orijentisane, međusobno normalne ose (x , y i z) sa odgovarajućim jediničnim

dužinama. Tačka u ovakvom sistemu definisana je sa tri vrednosti koje predstavljaju udaljenosti projekcije tačke na koordinatne ose od koordinatnog početka.



Slika 2.10 Kartezijev koordinatni sistem

2.3.4.2 Sferni koordinatni sistem

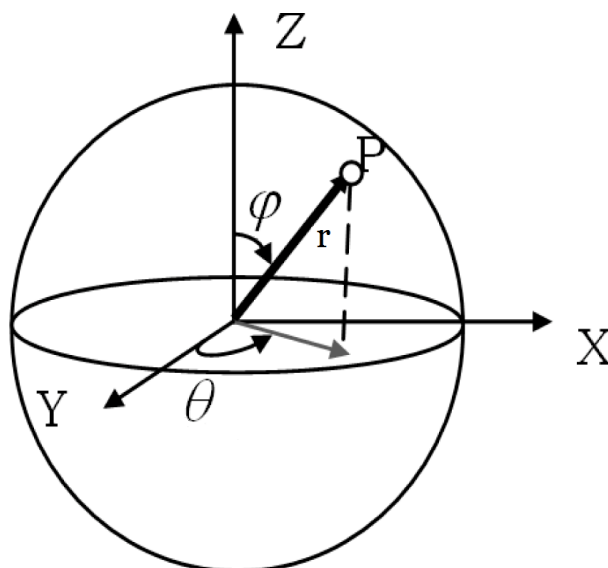
Sferne koordinate definišu položaj tačke u prostoru koristeći uglove i rastojanje. Tri koordinate (ρ , φ , θ) su definisane kao:

$r \geq 0$ je udaljenost od nulte tačke do date tačke P,

$0 \leq \varphi \leq 180^\circ$ je ugao koji zaklapa pozitivni deo z-ose sa pravom koja prolazi kroz nultu tačku i P,

$0 \leq \theta \leq 360^\circ$ je ugao koji zaklapa pozitivni deo x-osi sa pravom koja prolazi kroz nultu tačku i tačku P projektovanu na xy-ravan.

φ se naziva zenitom, a θ se naziva se azimutom. Sferni koordinatni sistem prikazan je na Slika 2.11.



Slika 2.11 Sferni koordinatni sistem

Za pretvaranje koordinata iz jednog u drugi sistem koriste se jednačine transformacije:

$$x = r \sin\theta \cos\phi, y = r \sin\theta \sin\phi, z = r \cos\theta, \quad (9)$$

$$r = \sqrt{x^2 + y^2 + z^2}, \theta = \tan^{-1}(z/\sqrt{x^2+y^2}), \phi = \tan^{-1}(y/x). \quad (10)$$

2.4 AVR (Audio/video prijemnik)

Jedan od uređaja potrošačke elektronike gde je prethodno opisani audio podsistem implementiran jeste AVR.

Razvojem potrošačke elektronike i informacionih tehnologija, a sa njima i filmske i muzičke industrije, došlo je do pojave tzv. kućnih bioskopa. Sa njihovim razvojem i željom da se funkcija više različitih uređaja objedini u jedan, nastao je AVR.

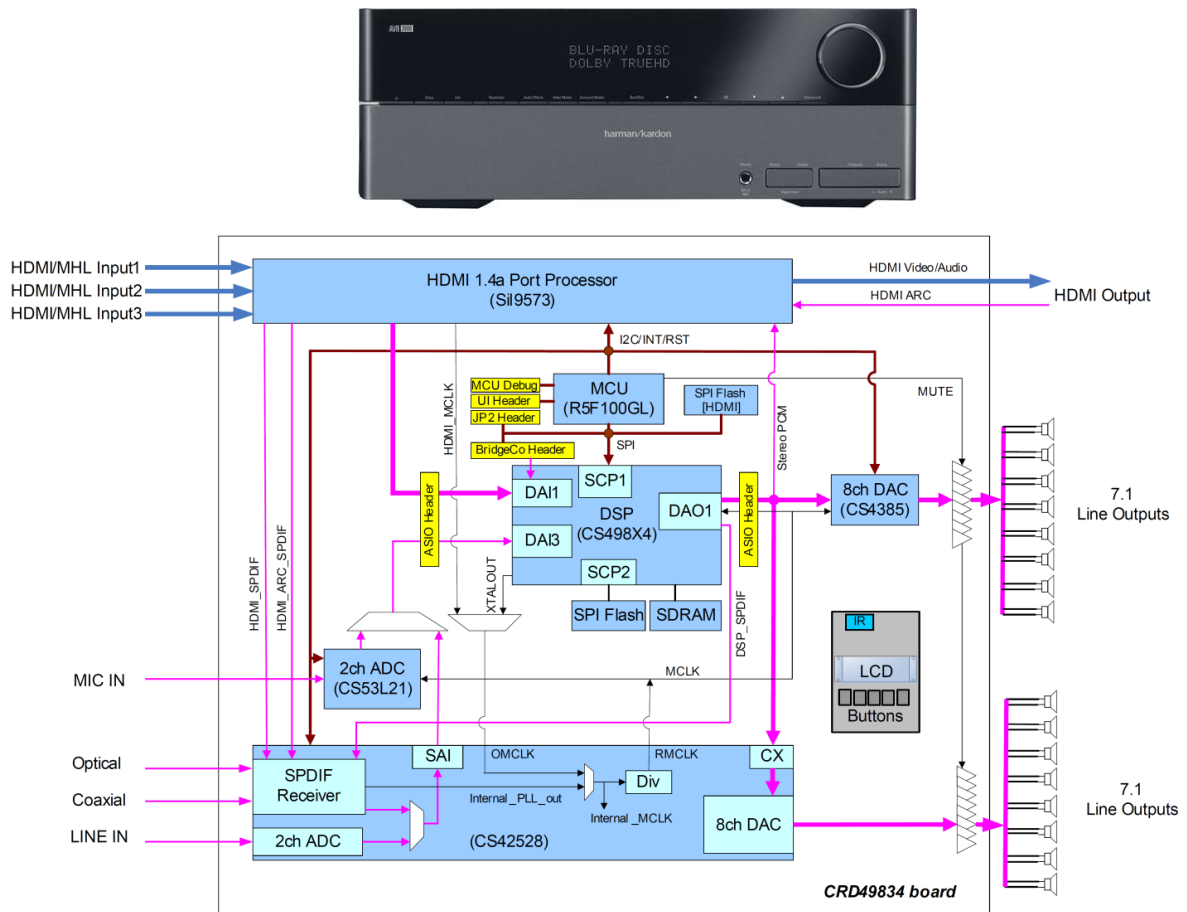
Glavna uloga AVR-a je prijem audio i video signala sa raznih izvora i obrada tog signala kako bi se mogli proslediti na zvučnike i displej (TV, projektor). Izvor signala mogu biti razni uređaji, poput satelitskog prijemnika, TV, radio prijemnika, DVD plejera, Blue-Ray plejera, video rekordera i konzola za video igre.

U audio terminologiji, pojam prijemnika podrazumeva uređaj koji objedinjuje audio pojačavač sa radio prijemnikom. AVR proširuje mogućnosti klasičnog prijemnika sa mogućnostima prijema i obrade različitih vrsta digitalnih i analognih audio signala, dok uporedo sa tim omogućava propuštanje video sadržaja na odgovarajući izlaz. Zadnjih par godina postoji tendencija da AVR-ovi takođe obrađuju i video signal.

Svaki AVR poseduje mogućnost dekodovanja jednog ili više tipova audio formata koji se nalaze trenutno na tržištu. Tendencija je proizvođača da u ponudi imaju uređaje koji mogu dekodovati sve postojeće tehnologije, dok se te tehnologije regularno unapređuju, nudeći veći broj izlaznih kanala, verniju reprodukciju zvuka, itd., što iziskuje sve moćnije sisteme za obradu audio signala.

Centralna jedinica obrade audio signala celog sistema predstavlja DSP, sa implementiranim opisanim audio podsistemom, koga nadgleda zaseban sistemski mikroracunar (eng. *host*). Kako je DSP sistem sa ograničenim resursima, zbog efikasnijeg rada, upravljanje njegovim radom, učitavanje programa sa perifernih jedinica, podešavanje sistema i sl. obavlja mikrokontroler.

Dakle, prijemnik audio i video sadržaja ima ulogu da primi analogni ili digitalni signal na ulazu, obradi ga (po potrebi dekoduje), pa tako obrađen signal da pretvori u analognu veličinu, koja se dalje vodi na audio pojačavač, na čijem se izlazu dobija signal za pobudu zvučnika. Blok dijagram sistema, kao i izgled samog uređaja dati su na **Error! Reference source not found.**



Slika 2.12 Prikaz izgleda i blok dijagram prijemnika audio-video signala zasnovanog na CS49844 procesoru

Razvojem tehnologije, analogna obrada signala je u potpunosti potisnuta digitalnom obradom signala, zbog mnogobrojnih prednosti digitalne nad analognom obradom. Jedina analogna obrada signala koja se vrši u ovom sistemu je pojačavanje na izlaznom stepenu (audio pojačavač), kako bi se korisni signal mogao sprovesti na zvučnike.

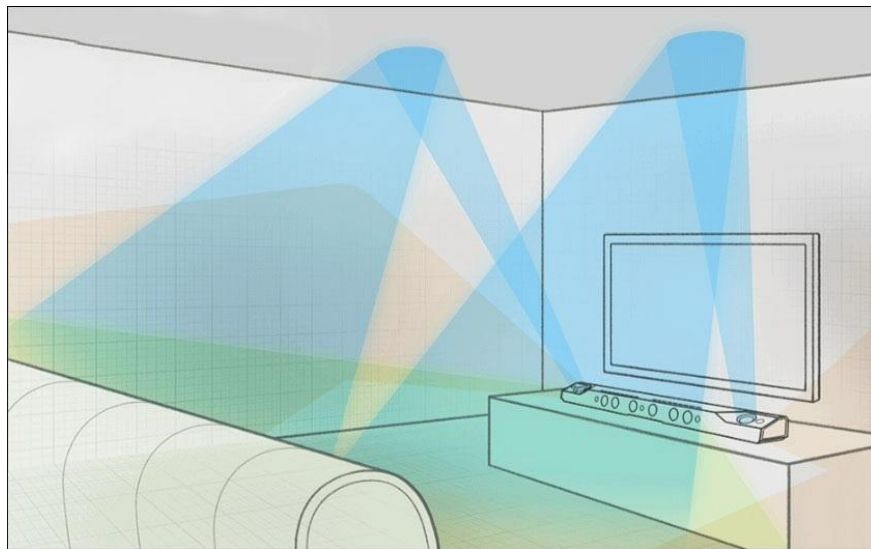
2.5 Soundbar

Ukoliko se gradi prostorija namenjena za kućni bioskop, tada postoji veća sloboda u izboru željenih zvučnika, kako po pitanju veličine ili estetike, tako i po pitanju njihovog pozicioniranja na idealno mesto u prostoriji da bi se dobile najbolje performanse.

Ukoliko ne postoji mogućnost da se praktično integriše višekanalni sistem u prostoriju, soundbar je veoma popularno rešenje. Dizajn ovakvog zvučnika omogućava lako postavljanje ispod ili iznad ekrana računara, TV uređaja ili kućnog bioskopa.

Soundbar je specijalna vrsta kućišta zvučnika koja od jedne tačke izvora stvara surround efekat. Unutar kućišta nalaze se zvučnici punog opsega, koji su pozicionirani na određeni način, tako da emitovanjem audio signala i njegovim odbijanjem o zid prostorije, zvuk dolazi do

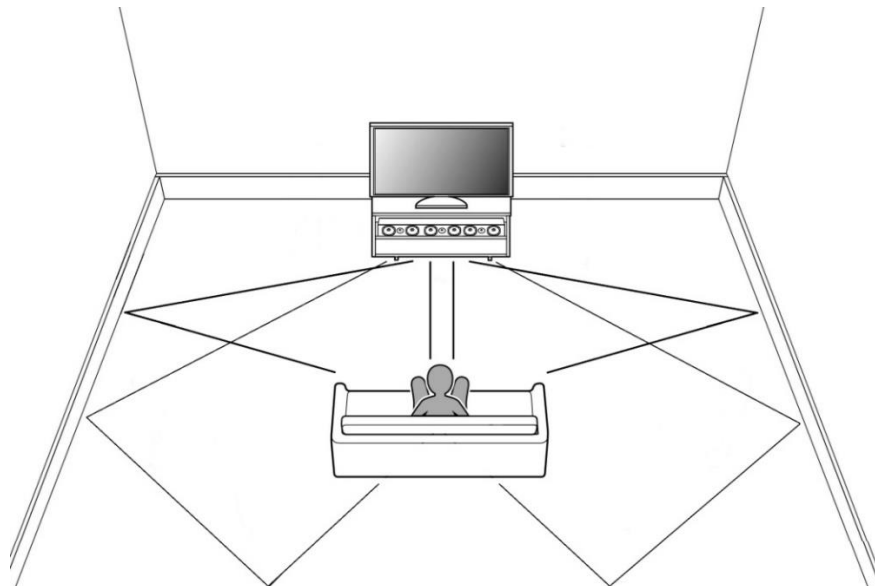
slušaoca stvarajući surround efekat. Ilustracija principa rada jednog soundbar uređaja prikazana je na Slika 2.13.



Slika 2.13 Ilustracija soundbar sistema

Kao što se sa slike može uočiti, da bi se dobio efekat levog i desnog surround zvučnika, audio signal se odbija o bočne zidove, dok signal odbijen od plafona stvara efekat pozadinskih zvučnika (eng. LB, RB - *Left Back, Right Back*).

Postoje i soundbar uređaji (Slika 2.14) koji surround efekat stvaraju reflektovanjem audio signala o zid prostorije, postavljanjem zvučnika punog opsega pod različitim uglovima unutar kućišta.



Slika 2.14 Ilustracija dobijanja surround efekta

3. Koncept rešenja

Upotreba objekata omogućava tačnu prostornu reprodukciju zvučnog signala nezavisno od konfiguracije zvučnika. Da bi to bilo omogućeno neophodno je da uz svaki zvučni signal postoje i odgovarajuće informacije koje moraju tačno da opišu kada, gde i koliko glasno će se dati signal pojaviti na zvučnicima. I upravo objekat u celosti podrazumeva zvučni zapis kojem su pridruženi ovakvi opisni podaci koje nazivamo meta podacima.

Meta podaci sadrže prostorne koordinate, pravce širenja zvuka, pojačanja, različite indikatore i kao takvi omogućavaju adaptaciju zvučnog sadržaja ciljnoj konfiguraciji zvučnika, dinamičko objektno renderovanje kao i interaktivnost sa korisnikom. Ove informacije mogu biti promenljive u toku vremena i kontrolišu proces renderovanja [11, 12].

3.1 Protokol za opis audio objekata

Prvi element meta podataka, Tabela 3.1, predstavlja indikator da li postoje promene u odnosu na prethodne podatke. On je veličine od jednog bita, i ukoliko mu je vrednost nula, validni podaci su već dobijeni prilikom prethodnog otpakivanja iz bitskog toka, dok jedinica signalizira da postoje novi podaci i da ih je potrebno iščitati. Ovi elementi meta podataka mogu se grupisati u odnosu na to koji tip informacije nosi.

POLJE	BROJ BITA
MetaDataUpdate	1

Tabela 3.1 Prvi element meta podataka

A) Prezentacija objekta

Svaki objekat ima svoj jedinstveni identifikacioni broj.

Svaki od ovih objekata nosi zvučni zapis koji može biti različitog tipa: muzika, efekti, dijalog, komentari, karaoke itd. Tip zvučnog zapisa može ali ne mora da bude deo meta podataka. U slučaju da se ova informacija ne sadrži u meta podacima, tip objekta će biti

proglašen nepoznatim. Svi ovi zvučni zapisi sastoje se iz većeg broja talasnih oblika. Talasni oblici se odnose na direktan zvuk, prve refleksije, ostale refleksije tj. reverberaciju. Koliko i koje talasne forme čine objekat su informacije koje su sadržane u meta podacima. Dalje, ovi talasne forme mogu se posmatrati kao skup tačkastih izvora. Odgovarajuće polje u meta podacima sadrži informaciju o broju ovih tačkastih izvora za svaki od talasnih oblika.

Odgovarajući indeksi nose informaciju o tačnoj poziciji objekta u bitskom toku. Takođe prenosi se i informacija o poziciji prvog talasnog oblika u objektu. Pozicija narednih talasnih formi preračunava se na osnovu pozicije prve talasne forme i pomeraja koji se sadrži u meta podacima.

Objekat može biti samostalan ili kao deo određene zamenske grupe. Zamenska grupa grupiše elemente koji su međusobno isključivi. Ona osigurava da će najviše jedan član u datom trenutku u vremenu biti aktivan. Ukoliko objekat pripada ovakvoj grupi, potrebne su informacije kojoj zamenskoj grupi pripada i da li je aktivan.

Elementi, kao i njihova bitska širina, prikazani su u Tabela 3.2.

POLJE	BROJ BITA
ObjectID	4, 6, 8, 10
ObjectTypePresent	1
ObjectType	4
NumberWF	1, 2, 3, 4
WFType	2
NumberPS	3
ChSetIndex	2, 3, 4, 5
StartChSetIndex	3, 4, 5, 6
WFOffset	3, 4, 5, 6
ReplacementGroupFlag	1
ReplacementGroupNumber	3
ActiveFlag	1

Tabela 3.2 Elementi grupe meta podataka za prezentaciju objekta

B) Prostorna reprezentacija

Kako se objekat sastoji od različitog broja tačkastih izvora zvuka, u prostoru može se pojednostavljeno posmatrati kao skup tačaka. Ona može biti stacionarna u prostoru u toku vremena (zvuk klavira na koncertu), ili nestacionarna, odnosno može se kretati u prostoru (avion koji preleće) i tada se može reći da tačka proširuje svoju dimenzionalnost, dobija širinu, dubinu, zavisno od pravca u kojem se kreće. Meta podaci moraju sadržati koordinate koje jednoznačno opisuju tačke (objekat) u prostoru kao i unapred definisane putanje po kojima će se kretati.

Odabrani sistem za opis položaja tačke u prostoru, kao i njenog kretanja, je sferni koordinatni sistem, Slika 2.11, i obzirom na to, koordinate koje opisuju položaj tačke u prostoru su: azimut - ugao θ , zenit – ugao ϕ , i poluprečnik ili udaljenost od koordinatnog početka. Azimut je u meta podacima definisan kao 8-bitna celobrojna vrednost u Q1 formatu opsega od -360 do

357, zatim se konvertuje u ugao vrednosti od -180^0 do $178,5^0$ sa rezolucijom od $1,5^0$. Ukoliko je objekat smešten u x-y ravni, znači da nema treću dimenziju i zenit je nula, $\phi = 0$. Ako je objekat iznad ili ispod x-y ravni zenit je sadržan je u meta podacima u opsegu od -180 do 180 , znači 6 bita je dovoljno da se opiše ova celobrojna vrednost, zatim se konvertuje u ugao čija je vrednost izražena u stepenima od -90^0 do 90^0 sa rezolucijom od $1,5^0$. Udaljenost objekta od koordinatnog početka data je relativno u odnosu na jediničnu sferu, definisanu poluprečnikom jednakim jedinici, u Q6 formatu. Ukoliko se objekat nalazi baš na površini jedinične sfere, udaljenost je definisana njenim poluprečnikom.

Ukoliko odgovarajući indikator signalizira da tačka menja svoj položaj, znači da meta podaci sa sobom nose i opis putanje kojom se ona kreće. U zavisnosti od toga da li se kreće tj. širi u horizontalnom, vertikalnom ili i u horizontalnom i u vertikalnom pravcu, odgovarajuće vrednosti se dobijaju iz meta podataka: ugao koji definiše širenje objekta u horizontalnoj ravni simetrično oko nultog azimuta, u opsegu od 0^0 do 360^0 , sa korakom od 3^0 u opsegu do 18^0 , i korakom od 6^0 u opsegu od 18^0 do 360^0 (veća rezolucija za manje pomeraje), ugao koji definiše širenje objekta u vertikalnoj ravni simetrično oko nultog zenita, u opsegu od 0^0 do 360^0 , sa korakom od 3^0 u opsegu do 18^0 , i korakom od 6^0 u opsegu od 18^0 do 360^0 (veća rezolucija za manje pomeraje) i ugao rotacije izražen u stepenima u opsegu $(-90^0, 90^0)$ sa rezolucijom od 3^0 . Ako se svi pokretni tačkasti izvori jednog objekta kreću na isti način, neće postojati odgovarajuće informacije o kretanju za svaki od njih pojedinačno, već samo jedan set ovih parametara u okviru meta podataka datog objekta.

Pregled ovih polja dat je u Tabela 3.3.

POLJE	BROJ BITA
MovingFlag	1
ElevationFlag	1
UnitSphereFlag	1
Azimuth	8
Elevation	7
Distance	6
SpatialExtension	3
CommonExtensionFlag	1
HorizontalSpread	6
VerticalSpread	6
Rotation	6

Tabela 3.3 Elementi grupe meta podataka za prostornu prezentaciju

C) Signalizacija posebnih opcija za reprodukciju

Meta podaci uključuju mogućnost signaliziranja direktne reprodukcije objekata na odgovarajuće zvučnike, bez renderovanja. Kako ovo odgovara kanalnom pristupu reprodukcije, ovi objekti nazivaju se kanalni objekti. U ovom slučaju, u meta podacima sadržana je i

predefinisana konfiguracija zvučnika – na koliko zvučnika i na kojim tačno zvučnicima će se reprodukovati signal.

Omogućena je i reprodukcija objekta na geometrijski najbližem zvučniku. Tačnije, dat je ugao koji definiše sferni odsečak oko objekta u okviru čijih granica će se tražiti zvučnik na kojem će se reprodukovati signal. Ukoliko u ovoj zapremini za datu konfiguraciju ne postoji ni jedan zvučnik, objekat će se renderovati normalnim tokom.

Za omogućavanje automatskog skaliranja audio scene, moguće je signalizirati da je položaj objekta vezan za vizuelni objekat na ekranu. Da bi se očuvao ovaj odnos između slike i zvuka potrebno je posebno renderovanje. Ovo trenutno nije omogućeno, element je predviđen za budući razvoj.

Koji tip meta podataka je sadržan – objektni, kanalni ili i jedan i drugi, dobija se iz dvobitskog elementa meta podataka – `MetaDataMode`, prvog elementa u Tabela 3.4.

POLJE	BROJ BITA
<code>MetaDataMode</code>	2
<code>NumberChannels</code>	5
<code>ChannelMask</code>	1-32
<code>EnableSnaptoSpeakers</code>	1
<code>SnaptoSpeakersAngle</code>	5
<code>LinkedtoVisualObject</code>	1

Tabela 3.4 Elementi grupe meta podataka za signalizaciju posebnih opcija za reprodukciju

D) Glasnoća

Glasnoća je još jedan od atributa objekta koji se sadrži u meta podacima. Ova polja nalaze se u Tabela 3.5. Ukoliko postoje informacije o amplitudi kojom će se primeniti signal na zvučnike, prenosi se u obliku indeksa preko kojeg će se iz odgovarajuće look-up tabele iščitati vrednost. U suprotnom ova vrednost je 0dB. Ova informacija prenosi se kao zasebni element u slučaju kanalnih objekata. Takođe moguća je i pojava zajedničkog faktora skaliranja za sve objekte, u tom slučaju iščitava se samo jedna vrednost koja predstavlja indeks preko kojeg se iz look-up tabele dobija vrednost faktora skaliranja.

Opseg ovog faktora je između -58dB i 0dB. Ovaj opseg podeljen je na tri podopsega sa različitim rezolucijama:

- [-58, -30] sa rezolucijom od 2dB,
- [-29, -15] sa rezolucijom od 1dB,
- [-14.5, 0] sa rezolucijom od 0,5dB.

Nad ovim faktorom dodatno može biti primenjeno pojačanje koje se prenosi kao poseban element u meta podacima. Ono dozvoljava da faktor skaliranja ima vrednosti do +12dB.

POLJE	BROJ BITA
GainPresent	1
GainIndex	6
CommonGainPresent	1
CommonGainBoost	2
CommonGainIndex	6
ContributionGain	6

Tabela 3.5 Elementi grupe meta podataka za glasnoću

E) Interaktivnost

Korisniku je omogućena kontrola nad pojedinim parametrima objekta, preko jednog od elemenata meta podataka, prikazanog u Tabela 3.6. To mogu biti pojačanja, geometrijski položaj objekta, odnosno njegove koordinate – azimut, zenit, poluprečnik.

POLJE	BROJ BITA
InteractiveFlag	1

Tabela 3.6 Elementi grupe meta podataka za interaktivnost

Prilikom dizajna meta podataka moraju se imati u vidu ograničenja ove platforme, kako u pogledu memorije za podatke tako i u programskoj memoriji. Potrebno je optimizovati ove podatke na način da ne budu predimenzionisani, a da sadrže sve neophodne informacije koje su rendereru potrebne. Na taj način dobija se bolja kompresija podataka i veći bitski protok.

Pre samih konkretnih informacija uvek postoje odgovarajući indikatori koji su veličine od jednog bita i koji signaliziraju da li određena grupa informacija sadržana u meta podacima i da li ju je potrebno iščitavati. Na ovaj način utrošak resursa se menja dinamički, odnosno srazmeran je trenutnim zahtevima korisnika. Da ovo nije slučaj, morali bi se enkodovati, prenositi, a potom i iščitavati svi podaci, što odgovara najzahtevnijem mogućem slučaju. Takođe, predviđena je i mogućnost zajedničkih parametara za više izvora, te se oni ne iščitavaju za svaki od ovih izvora pojedinačno, što umanjuje veličinu meta podataka, odnosno povećava kompresiju i umanjuje potrebno vreme za iščitavanje ovih podataka od strane dekodera.

Implementacija podržava maksimalno četiri talasne forme za jedan objekat, od kojih svaka može sadržati maksimalno dva tačkasta izvora.

U Tabela 3.7 dat je pregled svih elemenata meta podataka kao i njihovo značenje.

MetaDataUpdate	da li je bilo promene u odnosu na prethodne podatke: da/ne
Prezentacija objekta	
ObjectID	jedinstveni identifikator objekta
ObjectTypePresent	da li postoji informacija o tipu objekta: da/ne
ObjectType	tip objekta: muzika, efekti, komentari, dijalog, karaoke,

muzika i efekti	
NumberWF	broj talasnih oblika u objektu
WFType	vrsta talasnog oblika: direktna putanja, prve refleksije, kasnije refleksije (reverberacija), prve i kasnije refleksije
NumberPS	broj tačkastih izvora u talasnoj formi
ChSetIndex	indeks položaja objekta u bitskom toku
StartChSetIndex	indeks položaja prve talasne forme datog objekta u bitskom toku
WFOffset	ofset talasne forme u odnosu na indeks položaja prve talasne forme
ReplacementGroupFlag	da li objekat pripada zamenskoj grupi: da/ne
ReplacementGroupNumber	identifikator zamenske grupe
ActiveFlag	da li je objekat aktivan u zamenskoj grupi: da/ne
Prostorna reprezentacija	
MovingFlag	da li se tačkasti izvor kreće: da/ne
ElevationFlag	da li je tačkasti izvor van x-y ravni: da/ne
UnitSphereFlag	da li se tačkasti izvor nalazi na jediničnoj sferi: da/ne
Azimuth	ugao azimut
Elevation	ugao zenit
Distance	poluprečnik
SpatialExtension	da li se tačkasti izvor širi: ne, vertikalno, horizontalno, vertikalno i horizontalno
CommonExtensionFlag	da li su informacije o kretanju zajedničke za sve tačkaste izvore u objektu: da/ne
HorizontalSpread	ugao širenja u horizontalnoj ravni
VerticalSpread	ugao širenja u vertikalnoj ravni
Rotation	ugao rotacije
Signalizacija posebnih opcija za reprodukciju	
MetaDataMode	vrsta meta podataka: 3D objektni, kanalni i 3D objektni, kanalni
NumberChannels	broj kanala u ciljnoj konfiguraciji zvučnika
ChannelMask	ciljna konfiguracija zvučnika
EnableSnaptoSpeakers	da li je dozvoljena reprodukcija objekta na geometrijski najbližem zvučniku: da/ne
SnaptoSpeakersAngle	ugao koji definiše prostor geometrijski najbližeg zvučnika
LinkedtoVisualObject	da li je audio objekat povezan sa vizuelnim objektom
Glasnoća	
GainPresent	da li postoji pojačanje koje će se primeniti na signal koji se reprodukuje: da/ne
GainIndex	indeks vrednosti pojačanja
CommonGainPresent	da li postoji zajedničko pojačanje za sve tačkaste izvore u talasnoj formi: da/ne

CommonGainBoost	skaliranje zajedničkog pojačanja
CommonGainIndex	indeks vrednosti zajedničkog pojačanja
ContributionGain	faktor pojačanja kanalnih objekata
Interaktivnost	
InteractiveFlag	da li je dozvoljena interakcija korisnika za dati objekat: da/ne

Tabela 3.7 Elementi meta podataka za opisivanje audio objekata

4. Implementacija sistema za reprodukciju zvuka na bazi 3D objekata

U ovom poglavlju dat je opis implementacije sistema za reprodukciju zvuka na bazi objekata koji koristi protokol za opis audio objekata opisan u prethodnom poglavlju. Krajnje rešenje podrazumevalo je implementaciju na jednom procesoru, pa je zbog svoje arhitekture i resursa za ciljnu platformu odabran četvorojezgarni CS49844 procesor Crystal DSP familije kompanije *Cirrus Logic*. Date su glavne tehničke specifikacije ciljne DSP platforme i iskorišćenost resursa realizovanog sistema.

4.1 Ciljna platforma

Ciljna platforma ove implementacije je 32-bitni audio DSP procesor CS49844 [13] kompanije Cirrus Logic (Slika 4.1), unapredene Harvard arhitekture [14] sa dve memorijske zone za podatke i jednom memorijskom zonom za smeštanje programskog koda. Ovakva organizacija omogućava istovremeni pristup instrukcijama i podacima iz memorije, čime se znatno ubrzava rad procesora i postiže velika iskorišćenost resursa datog procesora i do čak šest instrukcija u jednom taktu – dve pomnoži i saberi MAC (*eng. Multiply and Accumulate*) instrukcije, dva čitanja ili upisa iz/u memoriju i dva povećanja/umanjenja adresnih registara.

Procesor poseduje četiri jezgra, pri čemu svaki od njih obezbeđuje 300 miliona instrukcija u sekundi (MIPS) procesne moći, 60 kilo reči memorije u svakoj od memorijskih zona, gde je reč dužine 32 bita. Svako jezgro radi na taktu od 300 MHz. U Tabela 4.1 dat je pregled ovih resursa.

Takođe, svako jezgro sadrži osam 32-bitnih registara opšte namene (x0-x3 i y0-y3), osam 72-bitnih registara (tzv. akumulatora, a0-a3 i b0-b3), namenjenih za smeštanje međurezultata, dvostruku preciznost i dr. Svako jezgro takođe sadrži dvanaest tzv. indeksnih registara (i0-i11), veličine 16 bita, namenjenih za adresiranje kod čitanja i upisa iz/u memoriju, kao i za druge

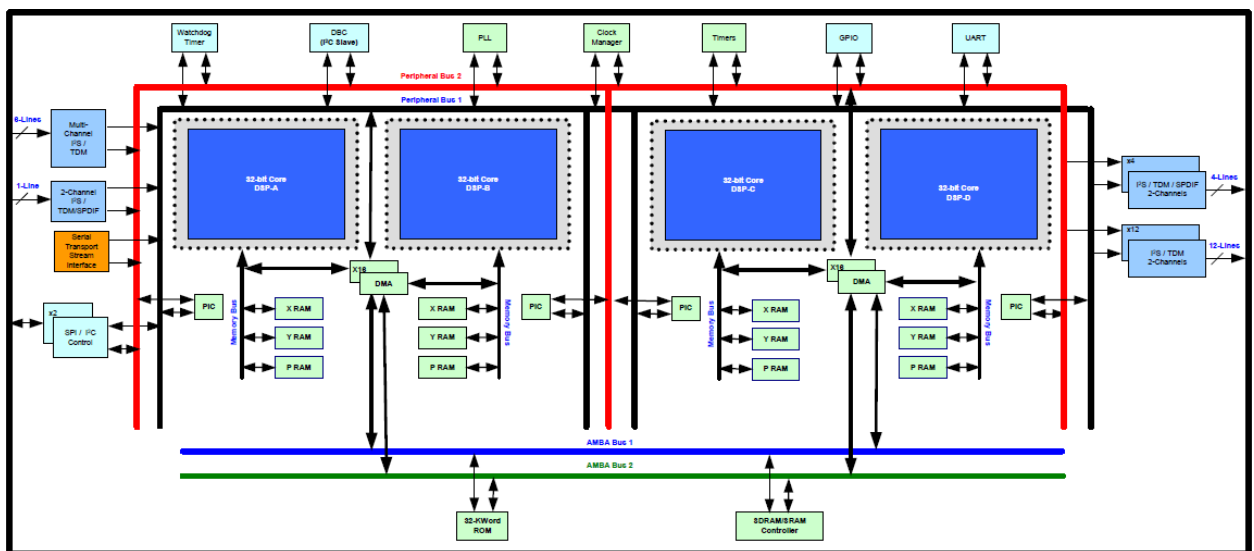
namene, kao što su brojači iteracija petlji i slično. Svakom od indeksnih registara odgovara i tzv. modulo registar, koji omogućava da odgovarajući registar radi u modulo režimu (korisno kod kružnih bafera) kao i bit-reverznom režimu (vrlo korisno kod algoritama koji zahtevaju izuzetno brzo izvršenje, kao što su brze furijeove transformacije i dr.).

Svako od jezgara sadrži tzv. SRS jedinicu (*eng. Shift Round Saturate*) koja upravlja aritmetičkim pomeranjem, zaokruživanjem i saturacijom između akumulatora i registara, kako joj i samo ime kaže. Platforma sadrži i jedinicu za direktan pristup memoriji, DMA (*eng. Direct Memory Access*), koji služi kako za prenos podataka između jezgara, tako i za prenos podataka između spoljne memorije i svakog od četiri jezgra.

Povezivanje sa periferijama omogućeno je preko SPDIF (*eng. Sony Philips Digital InterFace*), HDMI (*eng. High Definition Multimedia Interface*) kao i analognih ulaza-izlaza.

resursi	DSP A	DSP B	DSP C	DSP D
Procesorsko vreme	300 MIPS	300 MIPS	300 MIPS	300 MIPS
Memorija za podatke (X memorija)	60 kilo reči	60 kilo reči	60 kilo reči	60 kilo reči
Memorija za podatke (Y memorija)	60 kilo reči	60 kilo reči	60 kilo reči	60 kilo reči
Programska memorija (P memorija)	60 kilo reči	60 kilo reči	60 kilo reči	60 kilo reči

Tabela 4.1 Količina raspoloživih resursa ciljne platforme



Slika 4.1 Blok dijagram CS49844 čipa

4.2 Metodologija razvoja sistema na DSP platformi

Polazna osnova implementacije na ciljnoj platformi bio je komercijalni algoritam implementiran u programskom jeziku višeg nivoa C. Prvi korak je upoznavanje i analiza datog

referentnog koda, zatim procena potrebnih resursa, kako u pogledu memorije, tako i u pogledu procesorskog vremena.

Treba imati u vidu da se prilikom pisanja referentnog koda najviše obraća pažnja na ispravnost samog algoritma, dok je za pisanje koda u izvršnom programskom jeziku assembleru neophodno voditi računa o ograničenjima nametnutim odabirom ciljne platforme.

Prvi problem su, dakle, ograničenja memorijskih resursa i procesorskog vremena, te se već tokom samog početka pisanja koda uvode odgovarajuće modifikacije i algoritamske optimizacije referentnog koda koje omogućavaju efikasniju implementaciju na DSP platformi.

Prvi korak jeste funkcionalna optimizacija koda, koja podrazumeva organizaciju podataka, prilagođenje pristupa podacima, optimizaciju programskih petlji.

Smeštanje promenljivih u registre je ograničeno brojem registara koji je na DSP procesoru mali (dve grupe od po osam registara), dok je njihovo smeštanje na stek ograničeno ne samo veličinom steka, već i povećanim brojem instrukcija obrade u odnosu na neposredno smeštanje u memoriju podataka (jer je potrebno pristupiti steku i premestiti promenjive u memoriju ili registre pre obrade podataka). Zato je dobra praksa korišćenje globalnih ili lokalnih promenljivih. Primer za to jeste prosleđivanje argumenata kod funkcije obrade. Prosleđivanje argumenata uvek zahteva dodatne instrukcije prilikom poziva funkcije i na početku tela funkcije. Prilikom poziva potrebno je argumente smestiti, a unutar funkcije te parametre preuzeti. Takođe u koliko se parametri prosleđuju u funkciju potrebno je zauzeti dodatne registre za te argumente dok korišćenjem globalnih promenljivih ili korišćenjem simbola, argumentima se pristupa direktno. Takođe, u referentnom kodu se često koriste velike strukture u kojima se opisuje trenutno stanje aplikacije ili pojedinačnih procesnih modula. Ponekad se takve strukture prosleđuju kao parametri funkcija, a češće se prosleđuje pokazivač na strukturu. Ovakvo prosleđivanje rezultuje generisanjem neefikasnog koda i samim tim ga treba izbegavati. Poželjno je uočiti takve strukture i napraviti njihove globalne instance. Kada se funkciji prosleđuje pokazivač na strukturu, potrebno je obratiti pažnju da li je u kodu realizovana jedna ili više struktura. Samo ukoliko se koristi ista struktura, moguće joj je neposredno pristupati u telu funkcije.

Jedno od hardverskih proširenja digitalnih signal procesora je i jedinica za generisanje adresa podataka. Adresni generator je u stanju da obavi izvestan broj aritmetičkih operacija i da generiše adresu na kojoj se nalazi podatak. Po pravilu podržava kružno ili modulo adresiranje, kao i režim bit-obrnuto pristupa koji se koristi kod FFT-a. Iz ovog razloga operacije pristupa podacima potrebno je prilagoditi radu adresnog generatora. Ovo prilagođenje podrazumeva zamenu posrednog pristupa elementima niza preko indeksa, pristupom preko pokazivača na element.

Najveći deo procesorskih ciklusa tokom izvršavanja DSP aplikacija troši se na izvršavanje programskih petlji. Ušteda jednog instrukcionog ciklusa unutar tela petlje dovodi do uštede N instrukcijskih ciklusa, gde N predstavlja broj iteracija petlje po jedinici obrade. Deo koda unutar programskih petlji ponekad je nezavisan od trenutne iteracije ili čitave petlje, pa nema potrebe da se izvršava više puta. Ovaj kod naziva se nezavisni, odnosno invarijantni kod. Primer ovakvog koda jeste dodeljivanje konstantne vrednosti nekom resursu. Umesto da se ta dodela obavlja u svakoj iteraciji petlje, može se izvršiti samo jednom, pre ulaska u petlju. Ovaj pristup produžava životni vek promenljivih i dodatno opterećuje resurse, ali ubrzava izvršenje celokupnog koda jer skraćuje telo petlji.

Kako C radi u aritmetici pokretnog zareza, a DSP platforma u aritmetici nepokretnog zareza, ograničena na preciznost od 32 bita, drugi korak predstavlja prilagođenje koda aritmetici DSP-a.

Jedan od problema razlike u aritmetici javlja se kod koeficijenata čija je vrednost van zadatog opsega $[-1, 1)$. Konstante koje su van ovog opsega je potrebno skalirati, da bi se omogućile aritmetičke operacije nad ispravnim vrednostima. Konstante koje se nalaze u intervalu od $[1, 2)$ treba da podelimo sa 2, konstante koje se nalaze u intervalu od $[2, 3)$ treba da podelimo sa 4 i sl.

Osim ovoga, može da dođe do prekoračenja maksimalne vrednosti koja može da se smesti u akumulator. Kako bi se ovo izbeglo, moguće je uvesti bezbednosni opseg (*headroom*), odnosno, skalirati vrednosti signala na ulazu u blok odbrade (na opseg manji od maksimalne veličine tipova za podatke), a na izlazu iz bloka izvršiti inverzno skaliranje.

Zaseban problem jeste pojava da je vrednost konstante u nepokretnom zarezu isuviše mala da bi bila predstavljena sa datim tipom podataka. Preciznost tipa u nepokretnom zarezu može se predstaviti rastojanjem Δ između bilo koja dva susedna broja predstavljena ovim tipom. Ako je bilo koji tip u nepokretnom zarezu predstavljen u formatu $\langle a.b \rangle$, gde je a broj bita kojim je predstavljen celobrojni deo broja, a b broj bita kojim je predstavljen razlomljeni deo broja, tada je $\Delta=2^{-b}$. Konstanta ima isuviše malu vrednost da bi se predstavila određenim tipom ukoliko je njena vrednost manja od Δ za taj tip. U nekim slučajevima je dovoljno ovakav broj zaokružiti na najbližu vrednost. Međutim, u određenim slučajevima ovakav pristup dovodi do gubitka preciznosti i ispravnosti rezultata. Ovaj problem se često javlja u slučajevima kada je potrebno rezultat neke obrade pomnožiti sa koeficijentom pojačanja. Ukoliko je vrednost tog koeficijenta manja od Δ , i zaokružena na 0 tada je konačan rezultat obrade uvek 0. Jedan od načina da se ovaj problem reši jeste primena koeficijenta pojačanja po međufazama obrade. Neophodno je podeliti obradu na određene faze, potom razložiti konačni koeficijent tako da rezultat množenja novim koeficijentima po fazama daje rezultat ekvivalentan proizvodu ukupnog koeficijenta i izlazne

vrednosti obrade, i zatim izvršiti množenje sa odgovarajućim novim koeficijentom nakon svake faze obrade. [15]

Ukoliko je potrebno, moguće je i korišćenjem akumulatora (72 bita) za smeštanje međurezultata ostvariti dvostruku preciznost.

Nakon ovoga, funkcionalne celine je potrebno integrisati sa sistemskim softverom digitalnog signal procesora, odnosno, potrebno je sinhronizovati funkcije obrade sa događajima u realnom vremenu (kao što je prijem podataka). Ovaj sistemski softver se često zove i radno okruženje ili radni okvir (*eng. Framework*) digitalnih signal procesora.

4.1.1 Radni okvir procesora CS49844

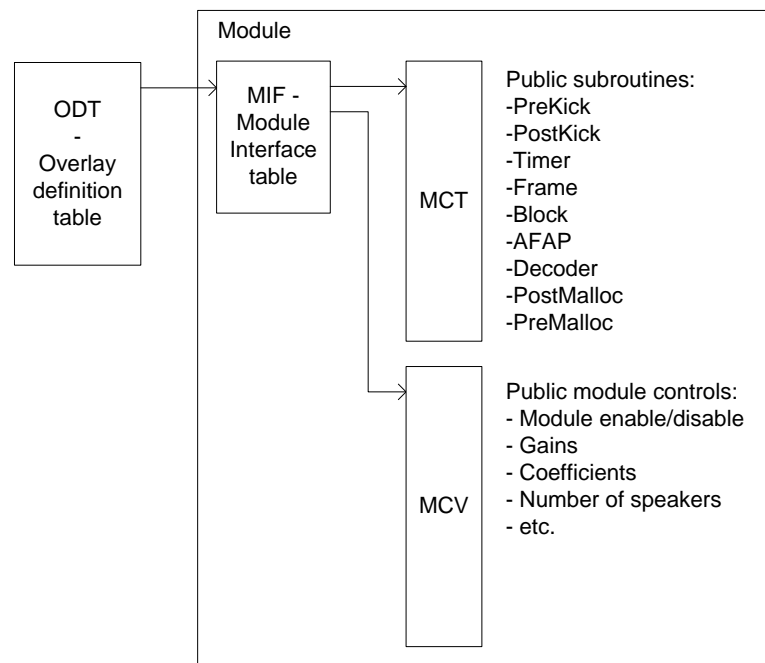
Radni okvir digitalnog signal procesora obezbeđuje raspoređivanje zadataka (funkcija obrade), rukovanje vremenski kritičnim zadacima, upravljanje tokom podataka između različitih zadataka, komunikaciju između programskih celina unutar samog procesora, kao i komunikaciju sa spoljnim uređajima.

Jezgro radnog okvira se sastoji od jednostavnog operativnog sistema (OS), čija je glavna uloga da bude raspoređivač za određen broj procesnih entiteta (modula). Uslovno rečeno, OS predstavlja monitorsku petlju koja poziva rutine odgovarajućih modula po unapred definisanom redosledu.

Pored operativnog sistema, osnovne komponente radnog okvira čine moduli. Moduli su definisani kao celine sastavljene od rutina i podataka, u skladu sa radnim okruženjem. Svaki modul ima svoj jedinstveni sprežni podsistem - MIF (*eng. Module Interface*) kojim je modul povezan sa OS-om. Njega čini MIF tabela koja sadrži pokazivače na tabele sa ostalim sprežnim informacijama. Dve najvažnije tabele su MCT tabela (*eng. Module Call Table*) i MCV tabela (*eng. Module Control Vector*). MCV tabela predstavlja niz javno dostupnih konfiguracionih parametara datog modula, i ona omogućava konfigurisanje modula od strane glavnog kontrolera uređaja (*eng. host*). Struktura ove tabele nema neku unapred definisanu formu i programeru je prepušteno da formira njen sadržaj i strukturu. MCT tabela je niz od devet elemenata – pokazivača na osnovne javne (*eng. public*) rutine. Svaka rutina ima specifičnu ulogu i pozivaju se od strane OS-a kao odgovor na pojavu odgovarajućih događaja u sistemu (*eng. event handlers*). On pristupa modulima preko ODT tabele (*eng. Overlay Definition Table*) koja sadrži pokazivače na MIF tabele svih učitanih modula.

Prva rutina, *Pre-Kickstart* rutina, poziva se nakon prijema inicijalizacione poruke (*eng. reset*). Nakon izvršene inicijalizacije svih modula, OS čeka *kickstart* poruku od kontrolera i poziva se *Post-Kickstart* rutina koja omogućava obradu konfiguracionih podataka prosleđenih modulima od strane kontrolera. Ukoliko je neki od modula zatražio dinamičko alociranje

memorije ono će se izvršiti u *Pre-Malloc* rutini, a potom je moguće i inicijalizovati zauzetu memoriju u *Post-Malloc* rutini. *Timer* rutina poziva se periodično kao odgovor na prekid (*eng. interrupt*) generisan od strane brojača realnog vremena. Izvršavanje *Block* i *Frame* rutine je upravljano ulaznim tokom podataka pri čemu se *block* rutina izvršava pri prijemu svakih 16 PCM odbiraka, dok se *frame* rutina izvršava na svakih N blokova. Ovaj broj blokova zavisi od dekoderskog modula u sistemu, odnosno od njegove jedinice dekodovanja. *AFAP* je skraćenica od “*As Fast As Possible*”. Ova rutina se poziva kada god se desi neki događaj u sistemu uz uslov da ne prekida druge rutine istog prioriteta. *AFAP*, *Timer*, *Frame* i *Block* rutine čine takozvanu *Foreground thread* (nit). *Background* rutina se izvršava periodično u pozadinskoj niti (*Background thread*), koja ima niži prioritet od *Foreground* niti i samim tim može biti prekinuta bilo kojom rutinom višeg prioriteta.



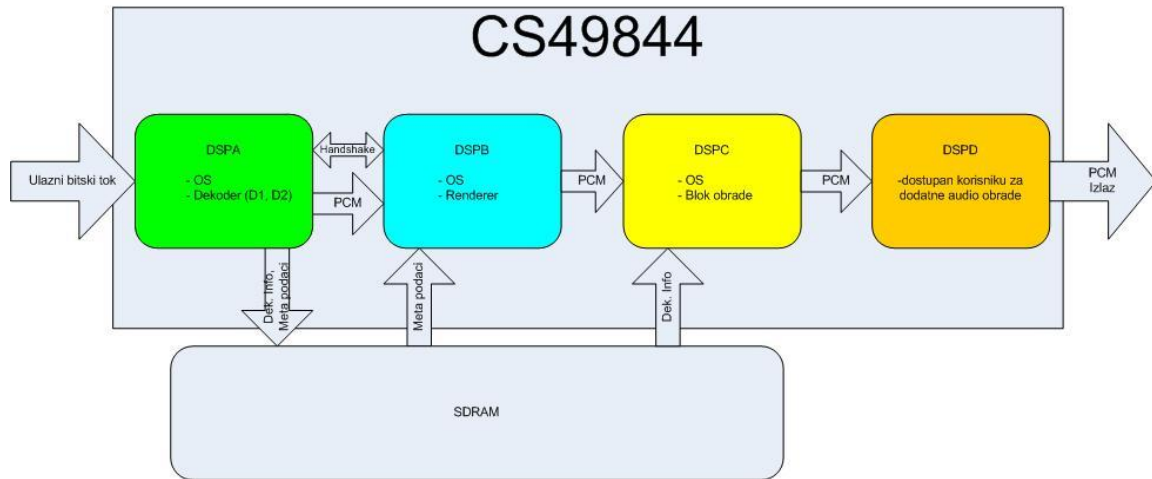
Slika 4.1 Blok dijagram sprege modula sa operativnim sistemom

Pored dela za rukovanje događajima, važan deo radnog okvira čini sistemski ulazno/izlazni memorijski niz koji služi za smeštanje audio podataka koji ulaze u sistem, nad kojima moduli vrše obradu i koji potom izlaze iz sistema (*eng. Input- Output Buffers*). Na Slika 4.1 dat je prikaz sprege modula i operativnog sistema.

4.1.2 Moduli implementiranog sistema

Celokupni implementirani sistem sastoji se iz tri modula: dekodler, renderer i blok obrade. Vrlo je složen te je i pored optimizovanog koda i pažljivog trošenja resursa vrlo zahtevan, kako memorijski tako i u pogledu procesorskog vremena. Iz tog razloga smešten je na tri od četiri jezgra CS48944 procesora. Na Slika 4.2 prikazana je raspodela modula po jezgrima. Dekoder

kao prvi nivo obrade smešten je na DSPA jezgru koje prima ulazni kompresovani tok podataka. Na DSPB jezgru smešten je blok renderer – uklanjjanje i dodavanje objekata. Na DSPC jezgru izvršavaju se različite obrade signala u cilju prilagođenja audio sadržaja ciljnoj konfiguraciji zvučnika, kao i prilagođenje nivoa signala zahtevima korisnika. Pored datih modula, za svako od jezgara postoji i zaseban operativni sistem.



Slika 4.2 Raspodela modula implementiranog sistema na jezgra ciljne platforme

▪ DSPA jezgro

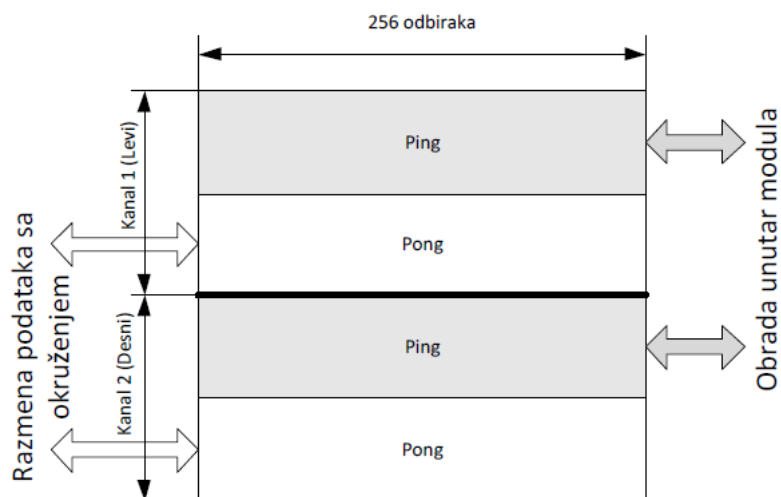
Jezgro DSPA prima ulazni bitski tok koji nosi enkodovani sadržaj sa HDMI ili SPDIF serijske sprege. On se smešta u internu Y memoriju ovog jezgra - FIFO memoriju, veličine 16k reči i dodatnih 24k reči eksterne memorije. Kako ulazni bitski tok neprekidno pristiže, u slučaju da je interna FIFO memorija puna, ulazni podaci se smeštaju u eksternu memoriju, zatim se prebacuju u internu kada je ona oslobođena - za ovo se brine OS.

Prvi nivo obrade ulaznog signala jeste dekodovanje koje obavlja dekodera. Implementirana su dva tipa dekodera, D1 i D2. Svaki dekoduje odgovarajući tip ulaznog signala. Dakle zavisno od ulaznog bitskog toka samo jedan od dva dekodera će se učitati na DSPA jezgru. Ovim upravlja autodetekcija u OS-u i mikrokontroler. Za smeštanje i prenos PCM podataka odgovoran je sistemski ulazno/izlazni niz (*eng. I/O buffer*) alociran od strane operativnog sistema u Y memoriji, izdijeljen na 32 segmenta, po 128 lokacija, za svaki od podržanih kanala. Svi moduli mogu pristupiti ovom nizu, putem pokazivača na svaki od kanala koje OS redovno ažurira.

Međutim, dekodovani podaci se od strane dekodera prvo smeštaju u interni PCM memorijski niz, odakle se kopiraju u ulazno/izlazni niz. Razlog za ovo je fiksna veličina ulazno-izlaznog niza, koja nije jednaka jedinici obrade dekodera - dekodera u jednom ciklusu obrade izdekoduje više podataka nego što se može smestiti u ulazno-izlazni niz.

Izlazni PCM odbirci iz dekodera predstavljaju ulazne odbirke narednog nivoa obrade, što znači da je nakon završenog ciklusa obrade od strane dekodera, potrebno čekati da se odbirci iz

internog PCM niza kopiraju u ulazno/izlazni niz, da bi moglo da se pristupi njihovoj obradi od strane drugih modula i kada svi moduli završe svoju obradu i krajnji izlazni odbirci se nađu u ulazno/izlaznom nizu, dekođer može da otpočne novi ciklus obrade. Sve ovo se dešava na DSP procesoru u realnom vremenu i ovakvo čekanje narušava dinamiku sistema, pa je interni PCM memorijski niz implementiran sa posebnom strukturom. Veličine je dvostruko veće od jedinice obrade dekođera, po kanalu. Ovakva struktura naziva se *Ping-Pong* memorijski niz, prikazan na Slika 4.3. Svaki kanal sastoji se iz dva dela - ping i pong. Dok dekođer dekođuje i upisuje sadržaj u Ping deo kanala, Pong deo kanala se prosleđuje na ulazno-izlazni memorijski niz na dalju obradu drugim modulima. Kada dekođer završi i popuni Ping delove kanala, Ping i Pong menjaju uloge i sada se dekođovani sadržaj upisuje u Pong deo, dok se upisani PCM podaci iz Ping dela upisuju u ulazno/izlazni memorijski niz i na taj način prosleđuju drugim modulima.



Slika 4.3 Ping-Pong struktura memorijskog niza

U Tabela 4.2 dat je pregled utrošenih resursa ovog jezgra. Ovaj modul zauzima oko 65% memorijskih resursa i oko 80% procesnog vremena ovog jezgra.

Utrošak procesorskog vremena	240 MIPS
Utrošak memorije za podatke (X memorija)	40448 reči
Utrošak memorije za podatke (Y memorija)	35626 reči
Utrošak programske memorije	25797 reči

Tabela 4.2 Utrošak resursa DSPA jezgra

Osim generisanja PCM odbiraka, dekođer raspakuje meta podatke iz ulaznih komprimovanih podataka i prosleđuje ih odgovarajućim modulima. Ovo slanje se obavlja preko spoljne SDRAM memorije i biće detaljnije objašnjeno u narednom poglavlju.

- **DSPB jezgro**

Blok renderer smešten je na DSPB jezgru. Ovaj blok inicijalno, pri uključenju uređaja, u odnosu na izlaznu konfiguraciju zvučnika koju dobija preko MCV tabele OS-a a čiju vrednost prijavljuje dekođer, preračunava skup mogućih tačaka u prostoru za smeštanje objekata, nakon čega se taj isti skup zajedno sa meta podacima koriste za raspoređivanje objekata na zadatu izlaznu konfiguraciju zvučnika. Ova inicijalizacija traje približno 180 000 000 instrukcija, odnosno oko 0,6 sekundi. Kako se obrada u bloku renderer može započeti tek nakon ove inicijalizacije, DSPB mora obavestiti DSPA jezgro kada je inicijalizacija završena. To je znak da je modul spreman za primanje podataka i obradu, nakon čega DSPA započinje slanje meta podataka u SDRAM memoriju. Ovaj *handshake* između DSPA i DSPB jezgra kontroliše OS preko odgovarajućih polja svoje MCV tabele i dešava se samo nakon pomenute inicijalizacije renderera – jednom nakon pokretanja sistema.

Nakon što primi meta podatke od strane DSPA jezgra, jezgro DSPB započinje obradu - uklanjanje objekata iz ulaznog kanalnog sadržaja i njihovo dodavanje izlaznom sadržaju zvučnika. Ova obrada se vrši uvek nad istim brojem odbiraka nezavisno od ulaznih podataka. Podržane frekvencije su 1fs: 32kHz, 44,1kHz, 48 kHz i 2fs: 64kHz, 88,2kHz i 96kHz.

U Tabela 4.3 dat je pregled utrošenih resursa ciljne platforme bloka za renderovanje. Ovi rezultati zadovoljavaju ograničenja ciljne platforme. Utrošak procesorskog vremena je oko 40 MIPS-a prilikom renderovanja objekata za njihovo uklanjanje kao i prilikom renderovanja objekata za njihovo uključivanje u audio izlaz, od čega je 25 MIPS-a utrošeno u toku preračunavanja faktora pojačanja kojim će se primeniti objekat na pojedine izlazne zvučnike, ili oduzeti iz osnovnog zvučnog zapisa, dok je 15 MIPS-a potrebno za samu primenu. Kada su kanalni objekti u pitanju potrošnja procesorskog vremena je oko 15 MIPS-a. Ovo je očekivano obzirom da se za ovu vrstu renderovanja ne vrše nikakva preračunavanja faktora pojačanja već se oni dobijaju direktno iz meta podataka.

Data potrošnja odnosi se za frekvencije od 1fs, dok je u slučaju dvostruko veće frekvencije i utrošak procesorskog vremena veći. Preračuni faktora pojačanja su nezavisni od frekvencije, pa je utrošak MIPS-a isti, dok se potrebni MIPS-i za samo uklanjanje i dodavanje objekata udvostručuje. Ovo je takođe očekivano ponašanje obzirom da je veličina ulaznih PCM odbiraka duplo veća u slučaju 2fs frekvencija. Sumirano, DSPB troši oko 155 MIPS-a na frekvencijama od 88,2kHz, 64kHz i 96kHz.

Utrošak procesorskog vremena	125 MIPS
Utrošak memorije za podatke (X memorija)	47975 reči
Utrošak memorije za podatke (Y memorija)	41612 reči
Utrošak programske memorije	9375 reči

Tabela 4.3 Utrošak resursa DSPB jezgra

- **DSPC jezgro**

Na DSPC jezgru nalazi se obrada signala kao što je spuštanje ili povećanje izlaznog broja kanala u skladu sa ciljnom konfiguracijom zvučnika, prilagođavanje zvučnog sadržaja različitom prostornom rasporedu zvučnika, normalizovanje nivoa signala. Ovaj modul takođe zahteva određene informacije od strane dekodera kao što su izlazna konfiguracija zvučnika, frekvencija signala, zatim opseg izlaznog signala, i slično. Prenos ovih informacija obavlja se na isti način kao i prenos podataka između DSPA i DSPB jezgra, preko SDRAM memorije. Veličina ovih podataka je 16 reči. Utrošeni resursi prikazani su u Tabela 4.4.

Utrošak procesorskog vremena	250 MIPS
Utrošak memorije za podatke (X memorija)	51652 reči
Utrošak memorije za podatke (Y memorija)	52609 reči
Utrošak programske memorije	19365 reči

Tabela 4.4 Utrošak resursa DSPC jezgra

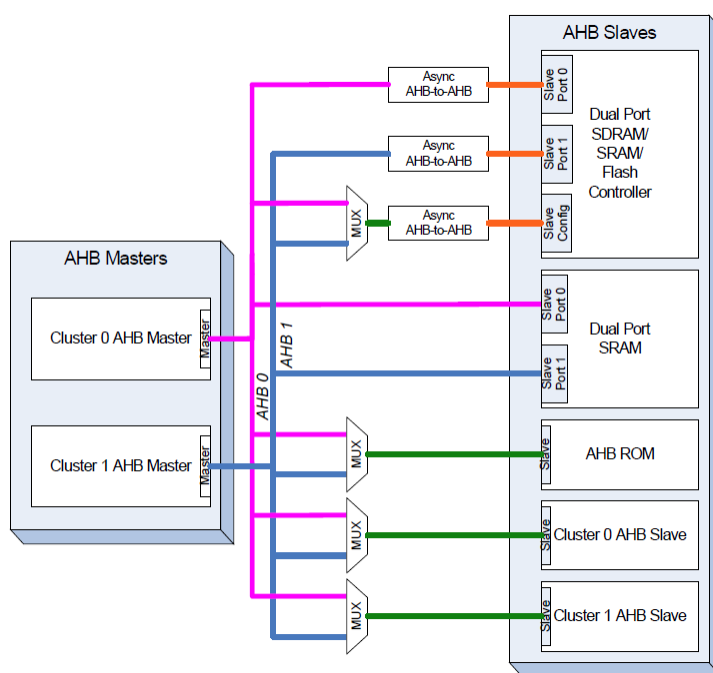
4.3 Protokol za prenos podataka između jezgara

Opisani meta podaci neophodni za obradu u sklopu bloka renderer prenose se od strane dekodera raspakovani iz kompresovanog ulaznog bitskog toka. Ovo uvećava količinu podataka koji se prenose u odnosu na slanje komprimovanog sadržaja, međutim otpakivanje enkodovanih podataka na strani 3D renderera bi dodatno usložilo modul koji je već dosta procesorski opterećen mnogobrojnim matematičkim proračunima. Takođe, pojedine informacije potrebne su i samom dekoderu, pa je implementacija algoritamski jednostavnija u slučaju da se kontinualan ulazni komprimovani sadržaj celokupan raspakuje na strani dekodera, odnosno jezgra koji i prima ove podatke u svoju internu memoriju.

Podaci između jezgara prenose se na početku svakog ciklusa obrade dekodera koji je dužine od 512, 1024 ili 2048 odbiraka, u zavisnosti od prirode ulaznog signala. Ovaj ciklus obrade dekodera naziva se *Frame*. Meta podaci su sinhronizovani sa audio sadržajem preko jedinstvenog broja Frame-a čije podatke nosi.

Ovako raspakovani podaci zauzimaju 5 reči meta podataka zajedničkih za sve prisutne objekte, zatim 522 reči meta podataka za svaki od postojećih objekata, kao i 17 reči koje nose informacije o ciljnoj konfiguraciji zvučnika, parametara zadatih od strane korisnika i sl. Najveći prenos obavlja se u slučaju maksimalnog broja objekata, što je 14 objekata, pa je tako maksimalan broj reči za prenos jednak 7330 reči.

Na Slika 4.4 prikazan je CS49844 procesor i njegova sprega sa periferijama na čipu.



Slika 4.4 Blok šema CS49844 procesora i periferija na čipu

Najjednostavnija razmena podataka između jezgara DSP-ja jeste preko deljene SRAM memorije, koju vide sva jezgra i veličine je 8k. Kako je ova vrednost približna vrednosti veličine podataka za slanje, rizično je da se podaci šalju na ovaj način. Iz tog razloga podaci se šalju preko SDRAM memorije veličine 64k reči. Sva komunikacija obavlja se preko DMA AHB protokola. Postoje dve AHB magistrale, AHB0 i AHB1, gde jednoj pristupa DSPA i DSPB jezgro - *Cluster 0*, a drugoj DSPC i DSPD - *Cluster 1*. DSPA kao nadređeni - *master* pristupa AHB0 magistrali nakon što dobije potvrdu od SDRAM-a da može da smesti podatke. DSPB potom pristupa istoj magistrali i čita podatke od podređenog - *slave* uređaja, SDRAM-a. SDRAM radi na taktu od 133 MHz sa bitskim protokom od 64 Mb u jednom transferu.

5. Testiranje i verifikacija

Testiranje i verifikacija sistema predstavljaju krajnji korak u implementaciji. Svrha ovog procesa je da se pokaže tačnost implementacije algoritma na ciljnoj platformi, tako što se utvrdi da se generisani izlazi iz implementiranog sistema slažu sa izlazima generisanim iz referentnog koda - referentnim izlazima. U toku ovog procesa osim detektovanja grešaka u softveru proverava se i kvalitet softvera.

U okviru ovog rada urađeno je funkcionalno ispitivanje metodom crne kutije – BBT [16] (*eng. Black Box Testing*). BBT alat je razvijen na Odseku za računarsku tehniku i računarske komunikacije Fakulteta tehničkih nauka u Novom Sadu i predstavlja moćan alat za ispitivanje i verifikaciju. Generalno rečeno, BBT alat je automatizovana i daleko naprednija verzija batch skripti. Prednosti BBT alata nad batch skriptama su brojne, od relativno jednostavne sintakse za pisanje ispitnih slučajeva, samim tim i bržeg razvoja istih, do lakog korišćenja, prenosivosti, mnoštva ugrađenih alata koji pojednostavljaju proces ispitivanja, sve do preglednih izveštaja o rezultatima. Ovaj alat dakle omogućava brz razvoj ispitnih slučajeva, a ugrađeni alati omogućavaju izvršavanje širokog spektra operacija. Jednom razvijeni ispitni slučajevi mogu se ponoviti neograničeni broj puta, kao i na proizvoljnom računaru (što je ponekad problem kod običnih batch skripti).

Od proizvođača tehnologije koja je bila predmet ispitivanja dobijena je dokumentacija koja predstavlja ispitnu proceduru. Ova dokumentacija sadrži specifikacije testnih slučajeva, ulazne podatke i ostale dokumente potrebne za verifikaciju implementacije testirane audio tehnologije. Ova dokumentacija je dostupna samo softverskim kompanijama koje žele da verifikuju svoju implementaciju i da dobiju sertifikat od kompanije koja je licencirala tu tehnologiju.

Početni korak u fazi razvoja testova podrazumeva organizaciju direktorijuma radnog okruženja. Pošto i testni okvir i BBT alat imaju određene specifičnosti i unose određene zahteve, radno okruženje mora biti organizovano tako da odgovori sve zahteve.

Uz svaki testni slučaj date su i tekstualne datoteke u kojima se nalaze sve informacije po kojima je određen testni slučaj karakterističan. U ovim datotekama navedeni su ulazni testni vektori, vrednosti konfiguracionih parametara i drugo. Kako bi se sve ove informacije, za sve testne slučajeve, nalazile sumirane na jednom mestu, napisana je Python skripta koja parsira prethodno pomenute datoteke i kreira Excel tabelu. U ovoj tabeli, u prvoj koloni izlistani su svi testni slučajevi, dok su u narednim kolonama upisane vrednosti svih parametara neophodnih za odgovarajući testni slučaj. Dalje se ova Excel tabela prosleđuje skripti u kojoj je implementirana logika pomoću koje se generišu BBT testovi. Uzima se jedna po jedna linija, odnosno željeni testni slučaj sa svim njegovim parametrima i generišu se testni slučajevi za BBT alat (.tst datoteke) čiji je deo prikazan na Slika 5.1, kao i konfiguracioni fajlovi (.cfg datoteke).

```

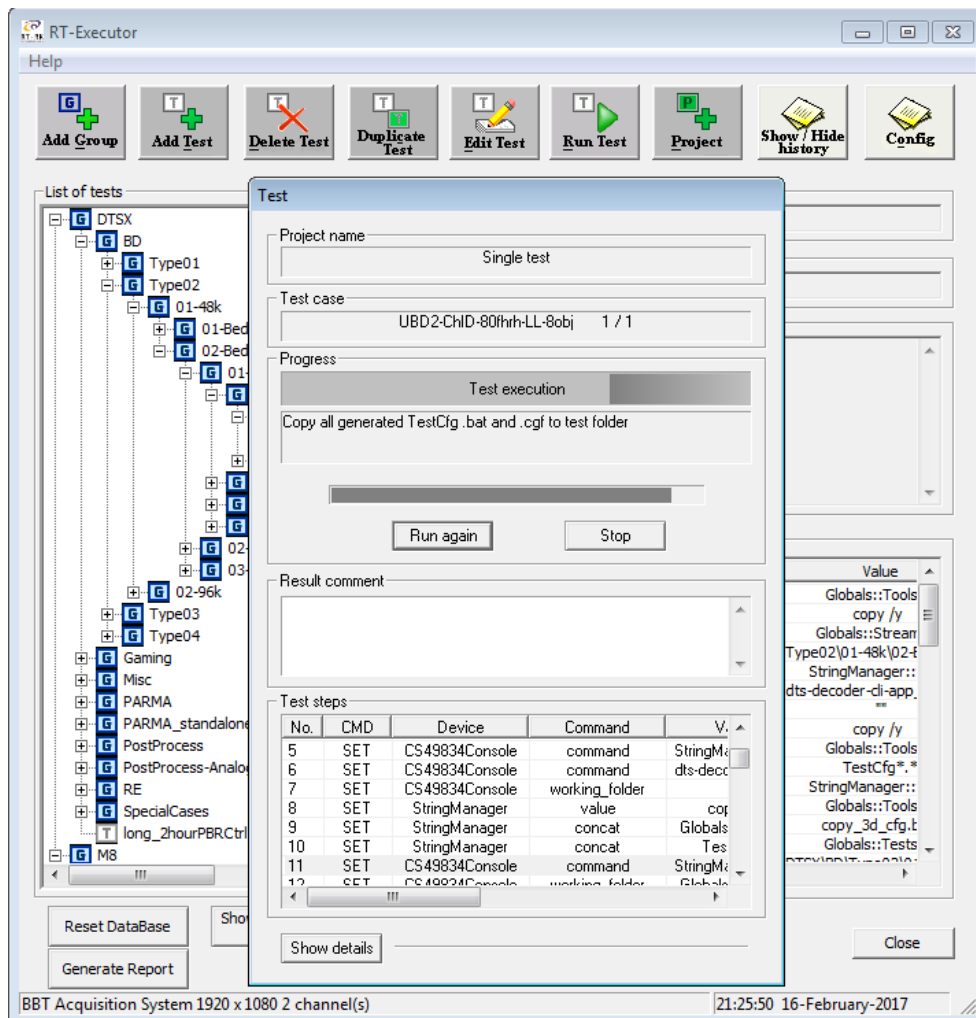
1  [test]
2  ID = 121
3  name = UBD2-ChID-80fhrh-LL-8obj
4  description = UBD2-ChID-80fhrh-LL-8obj test file
5  log_file = UBD2-ChID-80fhrh-LL-8obj.log
6
7  //////////////////////////////////////
8  //
9  //   Copy input stream to tools folder
10 //
11 //////////////////////////////////////
12
13 [step]
14 description = go to the Tools folder
15 device = CS49834Console
16 command = working_folder
17 value = Globals::ToolsDir
18 delay = 0
19 option = [SET]
20
21 [step]
22 description = Copy DTSUHD input streams to Tools folder
23 device = StringManager
24 command = value
25 value = copy /y
26 delay = 0
27 option = [SET]
28
29 [step]
30 description = Setup path to the input stream
31 device = StringManager
32 command = concat
33 value = Globals::StreamsBD
34 delay = 0
35 option = [SET]
36
37 [step]
38 description =
39 device = StringManager
40 command = concat
41 value = Type02\01-48k\02-BednObjects\01-ChID\01-8x\UBD2-ChID-4824LL-80fhrh-LL-8obj.dtshd
42 delay = 0
43 option = [SET]

```

Slika 5.1 Primer generisanog testnog slučaja za BBT alat (deo .tst datoteke)

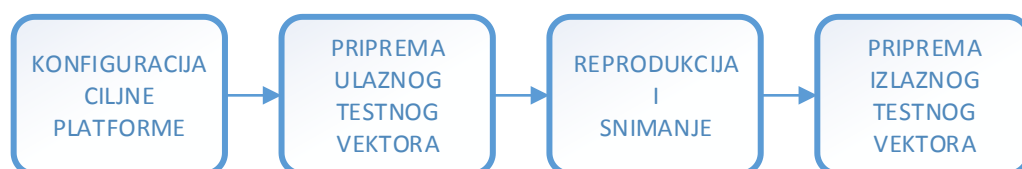
Ukupno je korišćen 3621 test, od kojih je sa 1418 testova testiran dekodler D1, dok se sa 2203 testa verifikovala funkcionalnost sistema sa dekodlerom D2. Testovi su podeljeni u 8 grupa, gde svaka grupa odgovara određenoj funkcionalnosti sistema.

Nakon generisanja testova, pokretanjem BBT aplikacije, Slika 5.2, počinje i samo izvršavanje testova.



Slika 5.2 BBT aplikacija nakon pokretanja

Na sledećoj slici je prikazan blok dijagram koji ilustruje osnovne korake jednog BBT testnog slučaja, koji redom prate korake date u .tst datotekama.

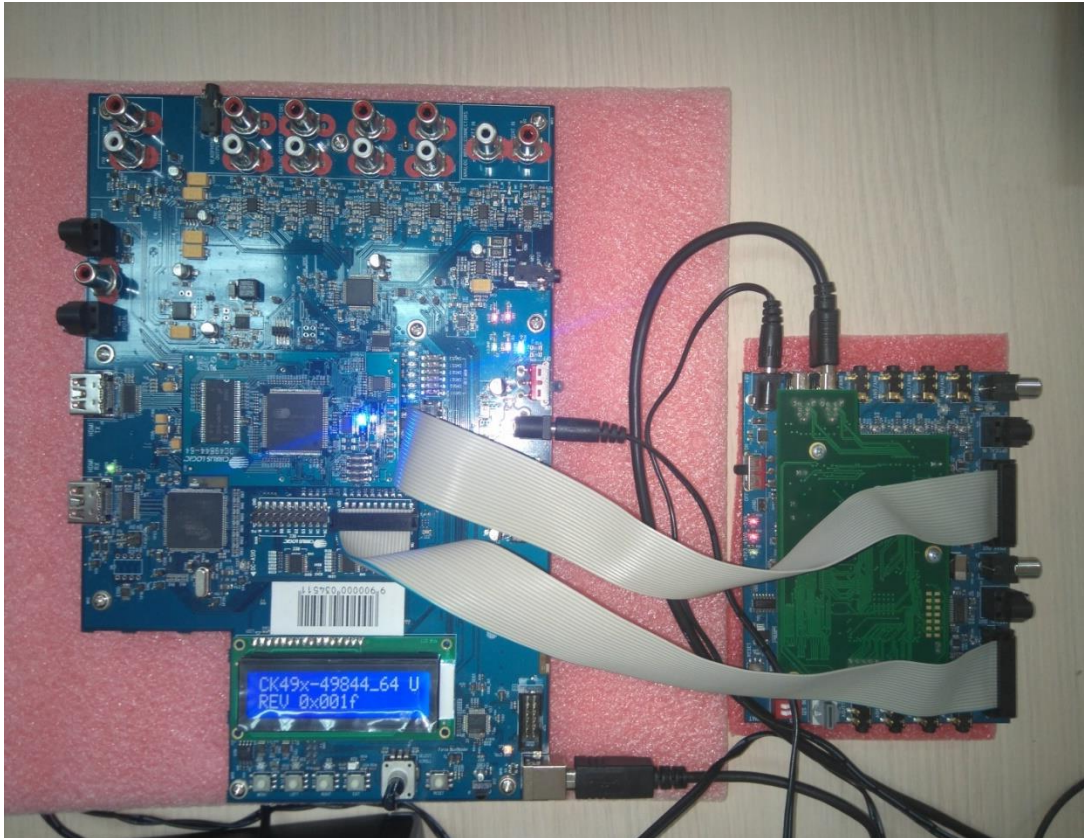


Slika 5.3 Blok dijagram izvršavanja jednog BBT testa

Prvi korak tiče se konfiguracije ciljne platforme, odnosno „spuštanja“ svih izvršnih datoteka koje su potrebne (firmware svih komponenti koje se paralelno izvršavaju) na procesor. U ovom koraku se takođe vrši i pozivanje konfiguracionih datoteka, odnosno komandi koje su u

njima navedene, a pomoću kojih se uključuju odnosno isključuju pojedine funkcionalnosti, šalju vrednosti pojedinih parametara, kao što su razna pojačanja, vrednosti graničnih frekvencija i tako dalje.

Na Sliku 5.4 dat je primer testnog okruženja. Kao ciljni procesor koristi se već pomenuti CS48944, četvororojezgarni digitalni signal procesor iz Crystal DSP familije procesora firme *Cirrus Logic*, a za reprodukciju i snimanje signala korišćena je zvučna kartica Echo Audio [17]. Na levoj strani slike je razvojna ploča CDB49X, u čijem centralnom delu je pozicioniran digitalni signal procesor, a sa desne strane se nalazi Echo Audio zvučna kartica.



Slika 5.4 Ispitno okruženje

Sledeći korak podrazumeva sve aktivnosti vezane za pripremu ulaznog testnog vektora. U nekim slučajevima, ulazni testni vektor se nalazi u formatu koji se ne može reprodukovati a da se prethodno ne dekoduje. Ponekad se dešava da je neophodno povećati frekvenciju odabiranja kod ulaznog testnog vektora.

U narednom koraku vrše se podešavanja Echo Audio zvučne kartice preko koje se obavlja reprodukcija i snimanje. Zvučna kartica koja je korišćena u ovom radu ograničena je da može reprodukovati testne vektore koji sadrže maksimalno osam kanala. Ukoliko ulazni testni vektor sadrži više od osam kanala, reprodukcija i snimanje se mora izvoditi u nekoliko iteracija.

Poslednji korak odnosi se na aktivnosti nakon snimanja izlaznog testnog vektora. Najčešće su to vremensko poravnanje snimljenog izlaza koje se vrši umetanjem sinhronizacione reči,

kombinovanje testnih vektora snimljenih kroz nekoliko iteracija u jedan izlazni testni vektor i slično.

Mogući ishodi izvršavanja BBT testova su *PASS*, *FAIL* i *INCOLCLUSIVE*. Ukoliko se svi koraci predviđeni testom uspešno izvrše, ishod testa biće *PASS*, sa druge strane, ukoliko se desi da prilikom izvršavanja nekog od koraka predviđenih testom dođe do greške ishod testa biće *FAIL*. Ukoliko se, pak, desi da ishod izvršavanja testa bude *INCOLCLUSIVE* to znači da je izvršavanje testa prekinuto pre nego što su izvršeni svi predviđeni koraci. Primer prikaza ishoda izvršavanja testova generisanog od strane BBT alata dat je na Slika 5.5.

No.	Test Name	Result	Time
1	sys_alg_arle_pkprotect_distortionrel...	PASS	15.08.2016,10:29:32
2	sys_alg_arle_pkprotect_distortionrel...	PASS	15.08.2016,10:30:53
3	sys_alg_arle_pkprotect_timbrepres...	PASS	15.08.2016,10:32:03
4	sys_alg_arle_pkprotect_timbrepres...	PASS	15.08.2016,10:33:35
5	sys_alg_arle_pkprotect_io7ch1_io48k	PASS	15.08.2016,10:34:45
6	sys_alg_arle_pkprotect_io2ch_io32k	FAIL	15.08.2016,10:36:38
7	sys_alg_arle_pkprotect_io2ch_io44k1	FAIL	15.08.2016,10:37:17
8	sys_alg_arle_pkprotect_io2ch_io48k	PASS	15.08.2016,10:37:55
9	sys_alg_arle_pkprotect_io5ch1_io32k	FAIL	15.08.2016,10:38:32
10	sys_alg_arle_pkprotect_io5ch1_io44k1	FAIL	15.08.2016,10:39:12
11	sys_alg_arle_pkprotect_io7ch1_io32k	INCONCLUSIVE	15.08.2016,10:39:42
12	sys_alg_arle_pkprotect_timbrepres...	INCONCLUSIVE	15.08.2016,10:40:00

Slika 5.5 Prikaz ishoda izvršavanja testova

Da bi se omogućilo potpuno automatizovano ispitivanje prihvatljivosti implementiranog rešenja, proizvođač tehnologije je obezbedio i posebnu softversku komponentu, odnosno testni okvir, koja omogućuje spregu pomoću komandne linije (eng. Command-Line Interface - CLI) za izvršavanje ispitivanja. Osim za ispitivanje prihvatljivosti, testni okvir se koristi i za generisanje referentnih signala, koji će biti upoređivani sa snimljenim signalima. Testni okvir je potpuno automatizovan i ne zahteva dalju interakciju sa korisnikom nakon pokretanja.

Postoji više načina ispitivanja rezultata:

1. Slušni testovi – preslušavanje ispitnih testova, provera čujnih distorzija u odnosu na referentni izlaz.
2. Testovi identičnosti u bit – poređenje referentnih i ispitnih izlaza na nivou bit-a.
3. Spektralni testovi – poređenje u spektralnom domenu.

Kriterijum za poređenje pojedinačnih testnih slučajeva dat je u pomenutom testnom okviru. Spektralni testovi čine oko 77% ukupnog broja testova, testova identičnosti u bit ima oko 17%, dok ostalih, 6% testova, čine testovi koji se porede bitski u vremenskom domenu sa

dozvoljenim određenim odstupanjima. Ovim testovima testiraju se pojedini delovi signala: tranzicije, amplitudski prelazi.

Ispitivanje rezultata završava se pokretanjem glavne skripte koja generiše izveštaj na unapred definisanoj lokaciji, u obliku HTML datoteke. Izveštaj ima formu tabele, a najbitnija polja su: jedinstveni identifikator testnog slučaja, metoda evaluacije koja je upotrebljena u konkretnom testnom slučaju i status - *PASSED*, *FAILED*, *UNRESOLVED*. Ukoliko je status testa *FAILED* ili *UNRESOLVED*, potrebno je preduzeti određene koraka, i ispitati razlog zbog koga test ne prolazi.

Dobijeni rezultati nakon izvršene opisane procedure testiranja zadovoljili su unapred definisane kriterijume za verifikaciju, status svakog od 3621 testnog slučaja je *PASSED*, i na taj način pokazano je da sistem odgovara zahtevima.

6. Zaključak

U ovom radu urađena je implementacija sistema za audio obradu zasnovanu na objektnoj tehnologiji, na višejezgarnoj platformi sa ograničenim resursima, CS49844 DSP procesorom firme *Cirrus Logic*. Polazna osnova je referentni kod napisan u višem programskom jeziku C. Prilagođenjem algoritma tako da zadovolji ograničenja nametnuta izborom ciljne platforme, kod je napisan u programskom jeziku assembler, integrisan u radno okruženje procesora i optimizovan.

Realizovani sistem sastoji se iz tri modula: dekođer, renderer i blok obrade. Zbog memorijskih zahteva, kao i zahteva u pogledu procesorskog vremena, sistem je izdđeljen po jezgrima upravo po pomenutim funkcionalnim celinama. Dekoder troši oko 60% memorijskih resursa i 80% procesorskog vremena DSPA jezgra. Blok renderer troši oko 80% memorijskih resursa i 50% procesorskog vremena jezgra DSPB, dok blok obrade koristi 70% memorijskih resursa i 85% procesorskog vremena DSPC jezgra. Protokol za opis audio objekata obavlja se preko spoljne SDRAM memorije. Ovi opisni podaci - meta podaci veličine su približno 8k reči za 14 objekata, što je maksimalan podržan broj objekata realizovanog sistema.

Rešenje je verifikovano koristeći namenski alat za testiranje, BBT, razvijen na Odseku za računarsku tehniku i računarske komunikacije Fakulteta tehničkih nauka u Novom Sadu. Nakon uspešno izvršene unapred definisane testne procedure koja se sastoji od 3621 testnih slučajeva, a potom i analize dobijenih rezultata, zaključeno je da sistem zadovoljava zadate kriterijume.

Ovakav sistem predstavnik je novije audio tehnologije koja je doprinela značajno kvalitetnijoj obradi zvučnog signala. Uvođenjem koncepta 3D objekta omogućena je tačna prostorna reprodukcija audio signala nezavisno od ciljne postavke zvučnika. Obezbeđena je fleksibilnost, bolji kvalitet obrade i veća interakcija sa korisnikom.

7. Literatura

- [1] A.D. Blumlein, Improvements in and relating to sound – reproduction systems, british patent 394,325, 1933.
- [2] J. C. Steinberg and W. B. Snow : Auditory Perspective – Physical Factors, Electrical Engineering, pp 12-17, January 1934.
- [3] M.A. Gerzon, Ambisonics: Part two: Studio techniques, Studio Sound August 1975:24-30, 1975.
- [4] Dolby, Dolby digital – the sound of the future here today, 1998.
- [5] K. Hamasaki et al, A 22.2 Multichannel Sound System for Ultrahigh-Definition TV (UHDTV), SMPTE Motion Imaging Journal, April 2008.
- [6] Guillaume Potard: 3D-audio object oriented coding, University of Wollongong, 2006.
- [7] Multichannel sound technology in home and broadcasting applications International Telecommunication Union, Report 2009.
- [8] Florian Hollerweger, An Introduction to Higher Order Ambisonic, 2008.
- [9] Ville Pulkki, Virtual Sound Source Positioning Using Vector Base Amplitude Panning, J. Audio Eng. Soc., Vol. 45, No. 6, June 1997.
- [10] Ville Pulkki, Generic panning tools for MAX/MSP, Laboratory of Acoustics and Audio Signal Processing, Helsinki, University of Technology Center for New Music and Audio Technologies (CNMAT), University of California, Berkeley, 2001.

-
- [11] Jürgen Herre, Johannes Hilpert, Achim Kuntz and Jan Plogsties: MPEG-H Audio - The New Standard for Universal Spatial / 3D Audio Coding, Ieee Journal of Selected Topics in Signal Processing, 2015.
- [12] Robert Oldfield, Ben Shirley, Jens Spille: An object-based audio system for interactive broadcasting, AES 137th Convention Los Angeles USA, 2014.
- [13] CS49844 Data Sheet, Cirrus Logic, Inc., May 2012.
- [14] V. Kovačević, M. Temerinac, M. Popović, N. Teslić, Arhitekture i algoritmi DSP-a I, Novi Sad, FTN, 2004.
- [15] J. Kovačević, D. Bokan, Arhitekture i algoritmi digitalnih signal procesora, Zbirka zadataka i laboratorijski priručnik, Novi Sad, 2016.
- [16] M. Popović, J. Kovačević, A Statistical Approach to Model-Based Robustness Testing, 14th Annual IEEE International Conference and Workshop on Engineering of Computer Based Systems, 2007.
- [17] Owner's Manual Version 2.2 for Windows, Echo Digital Audio Corporation, September 2009.