



УНИВЕРЗИТЕТ У НОВОМ САДУ
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У
НОВОМ САДУ



Филип Узуновић

Аутоматизација радног окружења за испитивање сложених аудио система

МАСТЕР РАД

Нови Сад, 2019.



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:	Филип Узуновић
Ментор, МН:	Др Јелена Ковачевић
Наслов рада, НР:	Аутоматизација радног окружења за испитивање сложених аудио система
Језик публикације, ЈП:	Српски / латиница
Језик извода, ЈИ:	Српски
Земља публиковања, ЗП:	Република Србија
Уже географско подручје, УГП:	Војводина
Година, ГО:	2019
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)	7/43/20/5/20/0/0
Научна област, НО:	Електротехника и рачунарство
Научна дисциплина, НД:	Рачунарска техника
Предметна одредница/Кључне речи, ПО:	аутоматизација, испитивање, дигитални аудио системи, аудио технологије
УДК	
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад
Важна напомена, ВН:	
Извод, ИЗ:	У овом раду је представљено аутоматизовано радно окружење за испитивање сложених аудио система, чијом употребом се смањује време неопходно за испитивање и до 50%. Као прецизан и поуздан уређај за ефикасно испитивање ових система који замењује скупа решења на тржишту, коришћен је <i>RT-AG (RT – Audio Grabber)</i> . <i>RT - Executor</i> који омогућава аутоматско извршавање испитних случајева, предложен је као програмска подршка испитивању. У раду је такође наведено и на који начин је унапређена сама процедура за испитивање.
Датум прихватања теме, ДП:	
Датум одбране, ДО:	
Чланови комисије, КО:	Председник: Др Иван Каштелан
	Члан: Др Јелена Радић
	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 :	Filip Uzunović
Mentor, MN :	PhD Jelena Kovačević
Title, TI :	Automated framework for testing of complex audio systems
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 :	2019
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>	7/43/20/5/20/0/0
Scientific field, SF :	Electrical Engineering
Scientific discipline, SD :	Computer Engineering, Engineering of Computer Based Systems
Subject/Key words, S/KW :	automation, testing, digital audio systems, audio technologies
UC	
Holding data, HD :	The Library of Faculty of Technical Sciences, Novi Sad, Serbia
Note, N :	
Abstract, AB :	This paper presents an automated framework for testing complex audio systems, whose use reduces the time required for testing up to 50%. RT – Audio Graber was used as an accurate and reliable device for efficient testing of these systems, which replaces the expensive solution on the market. RT-Executor, which allows the automatic execution of test cases, is proposed as a software support for testing. The paper also outlines the way in which the test procedure itself has been improved.
Accepted by the Scientific Board on, ASB :	
Defended on, DE :	
Defended Board, DB :	President: Dr Ivan Kaštelan
	Member: Dr Jelena Radić
	Member, Mentor: Dr Jelena Kovačević
	Mentor's sign

Zahvalnost

Zahvaljujem se svom mentoru Jeleni Kovačević, tehničkom mentoru Branku Đorđeviću kao i svom kolegi sa instituta „RT-RK“ Nenadu Pekezu za pomoć i saradnju tokom izrade projekta.

Posebno bih se zahvalio kolegi Petru Jeftiću i kolegici Lani Misirlis na razumevanju kao i stručnoj pomoći tokom izrade master rada.

Takođe se zahvaljujem svojoj porodici i prijateljima na konstantoj podršci tokom master studija.

SADRŽAJ

1. Uvod.....	1
2. Teorijske osnove	4
2.1 Istorijat razvoja digitalnih audio tehnologija	4
2.1.1 Nulta generacija	5
2.1.2 Prva generacija	5
2.1.3 Druga generacija	6
2.1.4 Treća generacija	7
2.2 Razvoj i ispitivanje softverskih sistema	8
2.3 Vrste i nivoi ispitivanja softvera	9
2.3.1 Jedinično ispitivanje softvera.....	9
2.3.2 Integraciono ispitivanje	10
2.3.3 Sistemsko ispitivanje.....	10
2.3.4 Regresivno ispitivanje	10
2.3.5 Funkcionalno i nefunkcionalno ispitivanje.....	10
2.3.6 Ručno i automatsko ispitivanje.....	11
2.4 Ispitivanje digitalnih audio sistema	11
2.4.1 Ispitivanje pomoću uređaja za reprodukciju i snimanje signala	12
2.4.2 Postojeći uređaji za reprodukciju i snimanje signala	14
2.4.2.1 Audio analizator APx	14
2.4.2.2 Echo ASIO zvučna kartica	15
3. Koncept rešenja	16
3.1 Tipična ispitna procedura.....	16
3.2 Tipičan hardverski sistem	17

3.2.1	Audio Precision – APx58x	18
3.2.2	Evaluaciona ploča CRD498x4	18
3.2.3	DSP CS49844 – Dagda čip	21
3.3	Unapređen hardverski sistem	23
3.3.1	RT – Audio Graber	23
3.4	Unapređena ispitna procedura	25
3.5	Izvršavanje ispitnih slučajeva	27
3.6	Analiza snimljenih izlaza	29
4.	Programsko rešenje	31
4.1	Organizacija radnog okruženja	31
4.2	Generisanje ispitnih slučajeva	32
4.2.1	Generisanje individualnih i poruka za menjanje režima rada	34
5.	Rezultati	38
6.	Zaključak	41
7.	Literatura	43

SPISAK SLIKA

Slika 1. Topologija ispitnog okruženja	12
Slika 2. Topologija ispitnog okruženja u praksi	13
Slika 3. Tipičan hardverski sistem.....	17
Slika 4. Audio analizator iz serije APx58x	18
Slika 5. Identifikovanje komponenti sa ploče CRD498x4	19
Slika 6. CDB-MCU-DEBUG-B dongla.....	20
Slika 7. Blok dijagram procesora CS49844	21
Slika 8. Blok šema funkcionisanja složene audio tehnologije na ciljnoj platformi.....	22
Slika 9. Unapređen hardverski sistem.....	23
Slika 10. RT-AG platforma	25
Slika 11. Prikaz unapređene ispitne procedure.....	26
Slika 12. RT-Executor početni prozor	27
Slika 13. Kreiranje ispitnog plana u RT-Executoru.....	28
Slika 14. Učitani ispitni plan	28
Slika 15. History prozor RT-Executora	29
Slika 16. Primer odsustva zvuka u audio signalu	30
Slika 17. Primer isecanja u audio signalu	30
Slika 18. Blok dijagram proces generisanja ispitnih slučajeva	32
Slika 19. Blok dijagram koji ilustruje korake ispitnog slučaja.....	33

SPISAK TABELA

Tabela 1. Pregled razvoja digitalnih audio sistema	8
Tabela 2. Lista komponenti ploče CRD498x4	20
Tabela 3. Raspoloživa memorija procesora CS49844	21
Tabela 4. Prikaz potrošnje resursa ciljne platforme.....	22
Tabela 5. Rezultati ispitivanja SoundBar uređaja na audio tehnologiji Dolby Atmos HT 1.6.....	39

SKRAĆENICE

PCM	- <i>Pulse Code Modulation</i> , Pulsno kodna modulacija,
VCR	- <i>Videocassete Recorder</i> ,
DAT	- <i>Digital Audio Tape</i> , digitalna audio traka,
DSP	- <i>Digital Signal Processor</i> , digitalni signal procesor,
S/PDIF	- <i>Sony/Phillips Digital Interface Format</i> ,
DVD	- <i>Digital Video Disc</i> , Digitalni video disk,
HDMI	- <i>High-Definition Multimedia Interface</i> ,
I2S	- <i>Inter-IC Sound</i>
PC	- <i>Personal Computer</i> , Personalni računar
DUT	- <i>Device Under Test</i> , uređaj koji se ispituje
ETH	- <i>Ethernet</i> ,
USB	- <i>Universal Serial Bus</i> ,
APx	- <i>Audio Precision</i> ,
QA	- <i>Quality Assurance</i> ,
MIPS	- <i>Million Instructions Per Second</i> ,
MCU	- <i>Microcontroller Unit</i> ,
UART	- <i>Universal Asynchronous Receiver-Transmitter</i> ,
AQM	- <i>Audio Quality Meter</i>

1. Uvod

U ovom radu predstavljeno je radno okruženje za automatizaciju ispitivanja složenih audio sistema zasnovano na hardverskom uređaju RT – Audio Grabber, koji se koristi za reprodukciju i snimanje audio signala, i softverskom alatu RT-Executor koji je odabran kao programska podrška ispitivanju.

Audio tehnologije su doživele ogroman napredak od početne dvokanalne (stereo) prezentacije, preko više kanalnih prezentacija, pa sve do savremenih 3D objektnih prezentacija.

Kanalne prezentacije zapravo predstavljaju tradicionalan način prenosa i reprodukovanja zvuka. Svaki kanal je dizajniran tako da se reprodukuje na tačno određenom mestu u sistemu u odnosu na slušaoca. 2D zvučna scena kombinacija je velikog broja pojedinačnih zvučnih elemenata, dijaloga snimljenih na setu, muzike snimljene u studiju itd. Ovi elementi distribuirani su na nekoliko audio zapisa u skladu sa standardizovanom konfiguracijom zvučnika [1]. Krajnji rezultat ovog procesa jeste skup kanala, gde svaki od njih predstavlja sadržaj jednog od zvučnika. Međutim takav signal pravilno će se reprodukovati na samo jednoj izlaznoj konfiguraciji zvučnika odnosno zvučnoj sceni. Naravno postoji mogućnost reprodukcije takvog signala i na drugim zvučnim scenama za koje nije predviđen, ali to zahteva još neke dodatne korake u obradi a sam doživljaj zvuka neće biti prostorni ukoliko zvučnici nisu dobro postavljeni. Danas su audio sistemi bazirani na kanalima i dalje veoma rasprostranjeni i vrlo stabilni, iako postoji sve veća težnja ka sistemima za reprodukciju 3D zvuka.

Za opisivanje prostiranja 3D zvuka koristi se određen broj 3D audio objekata [2]. Za razliku od kanala čije pozicije su tačno definisane, pozicija objekata u prostoru može biti:

- potpuno nezavisna od lokacije prisutnih zvučnika u sistemu
- mogu varirati tokom vremena za modelovanje pokretnih objekata kao što je avion koji se kreće iznad glave slušaoca

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 od nižih stereo, 5.1, do 22.1. Na ovaj način se postiže i kompatibilnost novih formata sa prethodnim uređajima.

Kako su nove generacije audio tehnologija donele određene promene, dovele su i do povećanja složenosti sistema. Samim tim proces verifikacije i ispitivanja ovih rešenja postaje složeniji i predstavlja sve veći izazov za inženjere. Podatak koji najbolje oslikava složenost ovog procesa jeste porast broja ispitnih slučajeva od nekoliko stotina do nekoliko hiljada ispitnih slučajeva. Metode manualnog ispitivanja postaju neefikasne u primeni, kako sa stanovišta utrošenog vremena, tako i pouzdanosti vezane za ocenu ispravnosti testiranog sistema. Ručno ispitivanje je zavisno od ljudske interpretacije, odnosno podložno greškama QA inženjera prilikom ocene ispravnosti sistema, usled nedovoljnog poznavanja specifikacije sistema ili iskustva. Sa druge strane, primena metoda ručnog testiranja oduzima značajan deo vremena potrebnog za razvoj sistema. Usled težnje proizvođača da obezbede konkurentnost na tržištu, razvoj sistema u ovoj oblasti je u velikoj meri uslovljen potrebom za skraćanjem vremena pojavljivanja proizvoda na tržištu [3]. Jedan od načina da se to postigne jeste skraćanje procesa ispitivanja. Sa druge strane, nedovoljno ispitan proizvod dovodi do smanjenja kvaliteta, što negativno utiče na konkurentnost. Da bi se obezbedilo kraće vreme proizvodnje i pružio bolji kvalitet, potrebno je više pažnje posvetiti analizi kvaliteta sistema i unaprediti proces ispitivanja, čime bi se poboljšao njegov kvalitet. Time se javlja potreba za automatizacijom ovog procesa, što predstavlja glavnu temu rada.

U drugom poglavlju dat je kratak istorijski pregled razvoja digitalnih audio tehnologija. Navedene su vrste i nivoi ispitivanja softvera, a potom je objašnjeno kako se vrše ispitivanja digitalnih audio sistema. Istaknuta je uloga koju imaju sistemi za puštanje i snimanje audio signala u ovim procesima, a navedeni su i neki od uređaja koji se najčešće sreću u praksi kao i njihove prednosti i nedostaci.

U trećem poglavlju najpre je opisana postojeća tipična ispitna procedura za ispitivanje složenih audio sistema, predstavljen je tipični hardverski sistem a opisani su i njegovi elementi. Sledi predstavljanje unapređenog hardverskog sistema, sa osvrtom na uređaj RT-Audio Graber (RT-AG) koji je razvijen u okviru RT-RK instituta. Potom je objašnjeno na koji način je unapređena sama ispitna procedura, kao i na koji način se automatski izvršavaju ispitni slučajevi. Ističe se kako upotreba RT-Executora, koji osim za automatizaciju izvršavanja ispitnih slučajeva, pomoću svojih ugrađenih algoritama omogućava i analizu snimljenih izlaza.

U četvrtom poglavlju predstavljena su programska rešenja. Prvo je predstavljena struktura i organizacija radnog okruženja, da bi se potom detaljno opisao postupak generisanja ispitnih slučajeva kao i proces generisanja konfiguracionih poruka.

U petom poglavlju poređeni su rezultati koji se dobijaju primenom stare i nove metodologije.

U šestom poglavlju dat je zaključak celokupnog rada.

Sedmo poglavlje sadrži spisak korišćene literature.

2. Teorijske osnove

U ovom poglavlju dat je kratak istorijski pregled razvoja digitalnih audio tehnologija. Navedene su vrste i nivoi ispitivanja softvera, a potom je objašnjeno kako se vrše ispitivanja digitalnih audio sistema. Istaknuta je uloga koju imaju sistemi za puštanje i snimanje audio signala u ovim procesima, a navedeni su i neki od uređaja koji se najčešće sreću u praksi kao i njihove prednosti i nedostaci.

2.1 Istorijat razvoja digitalnih audio tehnologija

Smatra se da pojava impulsno-kodne modulacije (eng. *Pulse Code Modulation, PCM*) predstavlja jedan od ključnih momenata za nastanak i razvoj digitalnih tehnologija. Tvorac ovog metoda za digitalnu reprezentaciju signala, koji je nastao 1938. godine, je britanski naučnik Alek Rivs (eng. *Alec Reeves*). Rivs je prvi prepoznao potencijal i uticaj koji PCM ima na smanjenje buke prilikom prenosa govora na velikim udaljenostima. Shodno tome, digitalna reprezentacija signala je prvobitnu primenu pronašla u telekomunikacionim sistemima. Tokom Drugog svetskog rata telefonska linija između Londona i Pentagona bila je kompromitovana i Nemci su uspevali da razbiju ne-digitalni bezbednosni sistem. Inženjeri kompanije Bell labs (eng. *Bell Labs*) razvili su prvi telekomunikacioni sistem za šifrovano prenošenje zasnovan na impulsno-kodnoj modulaciji pod nazivom SIGSALY. Bio je to prvi sistem koji je izvršio digitalizaciju govornog signala i njegov prenos pomoću PCM modulacije [4]. Međutim, PCM sistemi su u tim godinama bili vrlo nepouzdana, nestabilni i neefikasni po pitanju potrošnje energije. Sa razvojem poluprovodničkih tehnologija i integriranih kola, implementacija PCM sistema postala je ostvarivija i isplativija.

Razvoj digitalnih audio tehnologija započet je krajem šezdesetih godina prošlog veka i može se podeliti u nekoliko značajnih faza odnosno generacija [5].

2.1.1 Nulta generacija

Tokom 60-ih godina prošlog veka upotreba digitalnih signala u multimedijalne svrhe postaje značajnija. 1967. godine inženjeri japanske nacionalne televizije NHK razvili su sistem koji je bio u mogućnosti da snima audio signal na frekvenciji odabiranja od 32kHz i sa rezolucijom odmerka od 13 bita. Do proširenja sistema na 2 stereo kanala sa istim performansama dolazi 1969. godine od strane istih inženjera. Po uzoru na ovaj sistem, nakon saradnje sa inženjerima NHK televizije, kompanija Denon (eng. *Denon*) razvila je novi sistem pod nazivom DN-023R koji je mogao da snima 8 mono kanala na frekvenciji odabiranja od 47.25 kHz i sa rezolucijom od 13 bita. Ovaj uređaj odmah biva upotrebljen u komercijalne svrhe, pa su tokom 70-ih godina sa ovim sistemom snimljeni i prvi muzički albumi.

Još jedna japanska kompanija Soni (eng. *Sony*) je 1977. godine pustila u prodaju uređaje PCM-1 a godinu dana kasnije i PCM-1600, čijim pojavama su PCM sistemi još više komercijalizovani [6]. Ovi uređaji su predstavljali adaptere koji su omogućavali snimanje audio signala pomoću video rekordera (eng. *Videocassette Recorder, VCR*). U to vreme video rekorderi su bili jedini uređaji koji su omogućavali skladištenje podataka u propusnom opsegu koji je zadovoljavao opsege digitalnog zvuka. Digitalni signal zahteva mnogo veći propusni opseg u odnosu na analogne signale. PCM adapteri su na ulazima dobijali analogni signal koji su pretvarali u digitalni, koji bi zatim pretvarali u pseudo-video signal koji je mogao biti snimljen video rekorderom. Većina PCM adaptera radila je sa rezolucijom od 14 bita na frekvenciji odabiranja od 44.1 kHz koja je bila uslovljena sa najčešće korišćenim video standardima koje su video rekorderi koristili.

2.1.2 Prva generacija

Sa pojavom optičkih diskova (eng. *Compact Disc, CD*) dogodila se revolucija u digitalnim audio tehnologijama. Prvi put bivaju pušteni u prodaju 1982. godine od strane kompanija Soni i Philips (eng. *Phillips*). U narednih nekoliko godina prodaja optičkih diskova će čak premašiti prodaju gramofonskih ploča, što je posledica njihove kompaktnosti i kvaliteta reprodukcije. Format za audio optičke diskove definisan standardom podrazumevao je frekvenciju odabiranja od 44 kHz i rezoluciju od 16 bita. Pomenuta frekvencija je bila prihvaćena iz razloga što je u to vreme snimanje pomoću video rekordera bio najisplativiji način za prenos podataka od studija za snimanje do proizvođača diskova.

1987. godine na tržištu se pojavila digitalna audio kasetna (eng. *Digital Audio Tape, DAT*) kao modernizovana, integrisana i minijaturna verzija postojećeg PCM-VCR sistema. Pomoću digitalnih audio kasetna bilo je moguće snimati stereo signale na frekvencijama od 48, 44.1 i 32 kHz sa 16-bitnom kvantizacijom. Takođe, sve kopije izvornog signala bile bi identične.

Do kraja 80-ih godina digitalni audio sistemi su većinom zamenili analogne uređaje, kako profesionalne tako i one potrošačke, i na taj način digitalne audio tehnologije su u potpunosti preuzele primat. Na ovakav razvoj situacije najveći uticaj je imala pojava digitalnih signal procesora (u daljem tekstu **DSP**). Ova integrisana kola namenjena obradi digitalnih signala se prvi put pojavljuju ranih 80-ih godina. Kako su poluprovodničke tehnologije vremenom napredovale, performanse DSP-a su bivale sve bolje i bolje a digitalni sistemi sve pristupačniji što je uticalo da dobiju značajnu primenu u uređajima potrošačke elektronike. Tipičan DSP sistem podrazumeva pretvaranje analognih signala u digitalne, obradu digitalnih signala i pretvaranje digitalnih signala nazad u analogne.

Kritične tačke digitalnih sistema su veoma širok propusni opseg i njihovo međusobno povezivanje što dovodi do potrebe za uvođenjem standarda za digitalni prenos signala. AES3 je bio među prvim takvim standardima namenjen za povezivanje profesionalne audio opreme, a razvijen je 1985. godine. U isto vreme kompanije Soni i Philips po ugledu na AES3 standard razvijaju sopstveni standard pod nazivom S/PDIF (eng. *Sony/Phillips Digital Interface Format*) koji propisuje povezivanje potrošačkih audio uređaja na kratkim rastojanjima. Originalni S/PDIF standard je bio namenjen za prenos stereo PCM signala odmerenih na učestanostima od 44.1 kHz za CD uređaje i 48 kHz za DAT uređaje.

2.1.3 Druga generacija

Razvoj digitalnih tehnologija kao i multimedijalnih aplikacija zahtevao je sve kvalitetniju reprezentaciju zvučnog signala. Sredinom devedesetih godina pojavljuju se digitalni video diskovi (eng. *Digital Video Disc, DVD*) a višekanalna reprodukcija počela je masovno da se koristi u uređajima potrošačke elektronike. Kao standardni format audio zapisa na DVD diskovima usvojen je Dolby Digital standard sa bitskim brzinama od 384 i 448 Kb/s, a nešto kasnije usvojen je i DTS Digital Surround standard sa brzinama od 1.5 i 3 Mb/s. Upravo zbog toga prvobitni S/PDIF standard morao je biti proširen kako bi podržavao prenos višekanalnog komprimovanog audio signala odmerenog na 88.2kHz i 96kHz.

Sredinom prve decenije novog milenijuma usledila je nova generacija video diskova sa pojavom Blu-Rej (eng. *Blu-Ray*) diskova koji su omogućili reprodukciju signala komprimovanih bez gubitaka (eng. *lossles*). Audio zapis mogao je da nosi do 8 (7.1) kanala, sa učestanošću

odabiranja do 192kHz i 24 bita po odbirku. Kako kompresiju bez gubitaka karakteriše dosta niži stepen kompresije maksimalne bitske brzine povećale su se do 24Mb/s. Ograničenja S/PDIF sprege nisu dozvolila prenos ovakve vrste sadržaja, pa su za digitalni prenos signala korišteni drugi protokoli kao što su HDMI (eng. *High-Definition Multimedia Interface*) i I2S (eng. *Inter-IC Sound, IIC*).

HDMI omogućava istovremeni prenos nekomprimovanog video signala i komprimovanog i nekomprimovanog audio signala. Ova sprege predstavlja defakto standard za međusobno povezivanje multimedijalnih uređaja (TV prijemnici, konzole, audio-video prijemnici, audio plejeri, itd.) i ujedno je jedna od najrasprostranjenijih sprege u potrošačkoj elektronici [7]. U poslednjoj i dalje aktuelnoj verziji HDMI podržava protok podataka brzine do 48 Gb/s.

I2S standard nastao je sa ciljem da se omogući digitalni prenos zvučnog signala preko što je moguće manje fizičkih linija. Ukupan broj linija sveden je na 3, gde dve linije služe za prenos takt signala a preko treće linije vrši se vremenski multipleksiran (eng. *Time-Division Multiplex, TDM*) prenos 2 audio kanala. I2S sprege omogućava prenos podataka brzinama do 21.5 Mb/s [8].

2.1.4 Treća generacija

Pojava audio signala zasnovanih na objektima (eng. *Object-based audio*) predvodi poslednji talas razvoja digitalnih audio tehnologija. Sve do sada, najzastupljeniji audio signali zasnivali su se na kanalima. U slučaju kanalne prezentacije, zvuk koji publika čuje je kombinacija velikog broja pojedinačnih zvučnih elemenata. Na primeru filma, to su dijalozi snimljeni na setu, muzika snimljena u studiju itd. Svi ovi elementi distribuirani su na nekoliko audio zapisa i snimljeni u studiju koristeći kompleksne sisteme. Krajnji rezultat ovog procesa je skup kanala, gde svaki od njih predstavlja sadržaj jednog od zvučnika [9]. Mana ovakvog pristupa se ogleda u tome što se ovakvi signali mogu reprodukovati na samo jednoj unapred definisanoj izlaznoj konfiguraciji zvučnika.

U slučaju signala zasnovanih na objektima, svi pojedinačni zvučni elementi nose sa sobom meta podatke odnosno dodatne informacije, sa kojima zajedno formiraju audio objekte. Meta podaci tačno definišu kada, gde i koliko glasno objekti treba da se reprodukuju u trodimenzionalnom prostoru [9]. Oni omogućavaju adaptaciju zvučnog sadržaja ciljnoj konfiguraciji zvučnika a sadrže i informacije koje su promenljive u toku vremena kao što su prostorne koordinate, pojačanje, širenje virtuelnog izvora. Na taj način, variranjem pozicije objekata u toku vremena, moguće je modelovati različite zvučne kao što su npr. avion koji proleće iznad slušaoca, cvrkut ptica i sl. Zbog prenosa meta podataka tj. objekata ukupna

količina informacija koja se prenosi u slučaju ovih signala je znatno veća u odnosu na kanalne signale. Najtipičniji protoci idu čak do 86Mb/s, što je ekvivalentno količini podataka od 14 audio kanala koji su odmereni na frekvenciji od 192kHz i sa 32 bita po odmerku. Primena objektnih audio tehnologija trenutno se najviše ostvaruje u okviru filmske industrije i komercijalnih audio-video prijemnika koji se koriste u sistemima kućnih bioskopa.

Godina / Generacija	Tehnologija	Frekvencija / Rezolucija / Kanali	Nekomprimovani bitski protok	Kompromovani bitski protok
1967/0	NHK	32kHz/13b/1	416 Kb/s	-
1969/0	NHK	32kHz/13b/2	832 Kb/s	-
1972/0	DN-023R	47.25kHz/13b/1	614 Kb/s	-
1973/0	DN-034R	47.25kHz/14b/1	661 Kb/s	-
1976/0	Soundstream	37.5kHz/16b/1	600 Kb/s	-
1977/0	PCM-1/PCM-1600	44.1kHz/14b/2	1.2 Mb/s	-
1982/1	CD	44.1kHz/16b/2	1.4 Mb/s	-
1987/1	DAT	48kHz/16b/2	1.5 Mb/s	-
1996/2	DVD	96kHz/24b/6	13.8 Mb/s	3 Mb/s
2006/2	Blu-Ray	192kHz/24b/8	36.8 Mb/s	24.5 Mb/s
2015/3	Object-based audio	192kHz/32b/14	86 Mb/s	49 Mb/s

Tabela 1. Pregled razvoja digitalnih audio sistema

2.2 Razvoj i ispitivanje softverskih sistema

Kako softverske aplikacije postaju sve složenije, obimnije i strukturno kompleksnije tako se i vreme za razvoj istih povećava i meri čak i godinama. Osnovni kriterijumi koji određuju kvalitet softverskog proizvoda ipak moraju biti zadovoljeni, a to su:

- prilagodljivost – mogućnost prilagođavanja softvera prema izmenjenim zahtevima klijenata,
- pouzdanost – pouzdana i neškodljiva reakcija po fizičku arhitekturu u slučaju sistemskih otkaza,
- efikasnost – upotreba što je moguće manje resursa sistema (memorija, procesor..),
- upotrebljivost – posedovanje lako razumljivog korisničkog interfejsa i adekvatne dokumentacije [10].

Sistematičnost i formalizovan pristup razvoja softvera javlja se kao potreba kako bi se navedeni zahtevi ispunili u što je moguće kraćim vremenskim rokovima. Takva vrsta pristupa se ostvaruje najčešće putem jasno definisanih modela razvoja koji apstrahovanjem glavnih komponenata pomažu efikasnijem razvoju aplikacija. Konkretan model razvoja se odabira u zavisnosti od prirode projekta i aplikacije, tehničke orijentacije inženjera koji će raditi na razvoju, metoda i alata koji će se koristiti.

Pored sistematičnosti i formalizovanog pristupa, jednu od glavnih uloga u povećanju efikasnosti razvoja softvera zauzimaju ispitivanja istih. Dakle, ispitivanje softvera predstavlja izuzetno bitan deo procesa razvoja softvera. Prema nekim procenama, za ispitivanje i pronalaženje grešaka utroši se čak i do 50% ukupnog vremena za razvoj softverskog proizvoda [11]. Ukoliko je moguće vršiti ga na brz i jednostavan način, i ukoliko je u svakom momentu jednostavno reprodukovati ispitne sekvence, ubrzanje razvoja softvera može biti drastično jer je uviđanje i otklanjanje grešaka u softveru na taj način znatno lakše. Takođe, ispitivanja imaju izuzetno značajnu ulogu i nakon faze razvoja softvera, kada je potrebno izvršiti validaciju i verifikaciju sistema bez kojih isti ne može biti sertifikovan, što podrazumeva da se ne može koristiti u komercijalne svrhe.

2.3 Vrste i nivoi ispitivanja softvera

Ispitivanja softvera se mogu vršiti na različitim nivoima. Prema nivou, ispitivanje se deli na jedinično, integraciono, sistemsko, regresivno, ispitivanje prihvatljivosti i alfa i beta ispitivanje. Ispitivanje na različitim nivoima radi se sa ciljem da se sistem sagleda iz različitih perspektiva kako bi se pronašle greške različitih tipova. Ako se ovi nivoi sagledavaju sa strane automatizacije, nivoi mogu da budu jedinično ispitivanje, ispitivanje komponenti i sistemsko ispitivanje.

2.3.1 Jedinično ispitivanje softvera

Jedinično ispitivanje se odnosi na ispitne slučajeve koji proveravaju funkcionalnost svakog dela koda i to obično na nivou funkcije. Osnovni cilj ovakvog ispitivanja je otkrivanje strukturnih i funkcionalnih defekata u odgovarajućoj jedinici. Jediničnim ispitivanjem se potvrđuje da li pojedinačne komponente sistema mogu raditi zasebno ali se ne može proveriti funkcionalnost sistema.

2.3.2 Integraciono ispitivanje

Integraciono ispitivanje se vrši nad komponentama sistema odnosno podsistemima koji nastaju spajanjem osnovnih jedinica sistema sa ciljem da verifikuje da li sprege između komponenata rade ispravno. Takođe integracioni ispitni slučajevi verifikuju i korektnost toka operacija i toka podataka između komponenti.

2.3.3 Sistemsko ispitivanje

Sistemske ispitne slučajeve moguće je izvršavati kada se ispitivani podsistemi sjedine u finalni sistem. Na ovom nivou proverava se funkcionalno ponašanje sistema kao i nefunkcionalne osobine sistema. Sistemsko ispitivanje ima za cilj da proveri da li se sistem ponaša ispravno kada se sve komponente koriste zajedno.

2.3.4 Regresivno ispitivanje

Regresivno ispitivanje ima za cilj da utvrdi da li su stare karakteristike sistema zadovoljene i nakon određenih izmena u sistemu, kao i da verifikuje da promene nisu proizvele nove defekte. Kod sistema koji se održavaju duži vremenski period i koji se objavljuju u više verzija važnost regresivnog ispitivanja je veća. Ranije implementirane funkcionalnosti treba da rade zajedno sa novim funkcionalnostima, samim tim verifikovanje ispravnosti na ovom nivou može oduzeti dosta vremena pa se za ovakve zadatke preporučuje automatizacija.

2.3.5 Funkcionalno i nefunkcionalno ispitivanje

U zavisnosti od toga da li se ispituju funkcionalnosti sistema opisane zahtevima ili se ispituju kvalitativni aspekti softvera, ispitivanje se deli na funkcionalno i nefunkcionalno. Ispitivanje funkcionalnosti se fokusira na unošenje ulaznih parametara u sistem i verifikaciju izlaznih podataka, kako bi se utvrdilo da li se ceo sistem ili deo sistema ponaša onako kako je to definisano u specifikaciji. Nefunkcionalno ispitivanje odnosi se na brzinu rada, skalabilnost, sigurnost itd. Neki od primera nefunkcionalnog ispitivanja su:

- ispitivanje performansi,
- ispitivanje bezbednosti,
- ispitivanje pouzdanosti (ispitivanje u stresnim uslovima),
- ispitivanje zagušenja sistema.

Postoje dve metodologije koje se mogu primenjivati prilikom ispitivanja softvera, metodologije crne kutije i bele kutije. Kod ispitivanja zasnovanog na metodi crne kutije sistem se posmatra na taj način kao da se ne zna ništa o njegovoj unutrašnjoj strukturi. Cilj ovakvog ispitivanja je provera da li sistem za dovedene ulazne parametre daje izlaze u skladu sa specifikacijom. Kada se koristi metodologija bele kutije, unutrašnja struktura sistema je poznata. Cilj ovakvog ispitivanja je verifikacija korektnog ponašanja unutrašnjih strukturnih elemenata.

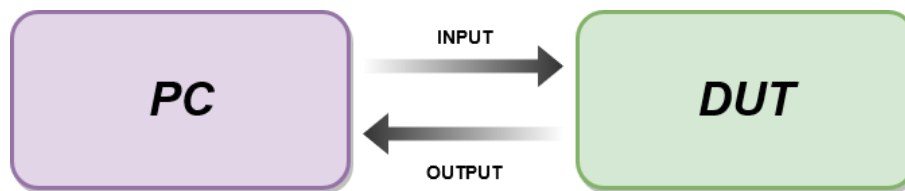
2.3.6 Ručno i automatsko ispitivanje

Prema načinu izvršavanja ispitnih slučajeva proces ispitivanja se deli na ručno i automatsko ispitivanje. Ručno, odnosno manuelno, ispitivanje izvršava čovek. QA inženjer, prateći uslove postavljene u ispitnim slučajevima, manuelno ispituje da li softver zadovoljava sve kriterijume i manuelno unosi rezultate ispitivanja. On koristi softver kao što bi to činio krajnji korisnik i odlučuje da li se softver ponaša na očekivan i unapred propisan način. Manuelno ispitivanje je korisno ukoliko je automatizacija nemoguća ili ukoliko se ispitni slučajevi izvršavaju jednom do dva puta. Sa druge strane, veoma je neefikasno sa stanovišta utrošenog vremena i pouzdanosti. Pošto je zavisno od ljudske interpretacije podložno je greškama QA inženjera prilikom procene ispravnosti sistema. Automatsko ispitivanje oslanja se na specijalan softver, nezavisan od softvera koji je predmet ispitivanja. Služi za kontrolu izvršavanja ispitnih slučajeva, poređenje dobijenih i očekivanih rezultata simulaciju stanja sistema neophodnih za izvršavanje ispitnih slučajeva kao i aktivnosti vezane za izveštavanje [12]. U odnosu na manuelno, automatsko ispitivanje je brže, efikasnije, pouzdanije i preciznije. Posebno je značajno za ispitivanje softvera koji imaju dugačak period održavanja, jer čak i najmanje promene u novoj verziji koda mogu izazvati kvar aplikacije i ako je u prethodnim verzijama radila na očekivan način. Kada se razmišlja o automatizaciji, potrebno je pre svega odgovoriti na dva ključna pitanja: šta i kada automatizovati? Odluku o tome donosi razvojni odnosno ispitni tim. Uspeh automatizacije takođe veoma zavisi od odabira alata za automatizaciju.

2.4 Ispitivanje digitalnih audio sistema

Uopšteno gledano ispitivanja digitalnih audio sistema svode se na prosleđivanje ulaznih audio signala (eng. *input*) od računara opšte namene (eng. *Personal Computer, PC*) do ciljne platforme tj. uređaja koji se testira (eng. *Device Under Test, DUT*), preuzimanje izlaznih audio signala (eng. *output*) i poređenje izlaznih rezultata sa referentnim vrednostima. Kompanije koje

obezbeđuju algoritme (eng. *IP provider*), odnosno vlasnici digitalnih audio tehnologija, obezbeđuju specifikacije ispitivanja kao i referentne vrednosti.



Slika 1. Topologija ispitnog okruženja

Postoji nekoliko pristupa proveru ispravnosti dobijenih rezultata. Objektivna provera ispravnosti rezultata ostvaruje se izvršavanjem ispitnih slučajeva identičnosti na nivou bita. Ova vrsta ispitivanja vrši se tako što se poredi datoteka koja sadrži rezultat obrade sa referentnom datotekom na nivou binarnog zapisa. Ukoliko između dve datoteke koje se porede ne postoje razlike na nivou bita, ili su te razlike u okviru dozvoljenih granica odstupanja definisanih specifikacijom, verifikacija sistema smatra se uspešnom. Postoje slučajevi kada je unos određenih izmena u izlazni signal očekivan, odnosno, kada se ne očekuje da izlazni rezultat i referentni rezultat budu identični na nivou bita. Tada se koristi spektralna analiza rezultata i vrši se poređenje rezultujućeg i referentnog signala u spektralnom domenu.

2.4.1 Ispitivanje pomoću uređaja za reprodukciju i snimanje signala

Ključan faktor u procesu ispitivanja digitalnih audio sistema predstavlja način prenošenja podataka do i od uređaja koji se ispituje a kvalitet rešenja ovog problema direktno utiče na kvalitet celokupnog procesa ispitivanja. Postoji više načina na koje može da se obavi razmena podataka između računara i platforme. Jedno rešenje je direktno povezivanje računara sa uređajem koji se ispituje putem nekih od digitalnih ili analognih sprege. Topologija ovakvog rešenja prikazana je na slici 1. Ovakav način prenosa signala je moguć iz razloga što najveći broj DSP platformi podržava analogne ulaze/izlaze i digitalne sprege kao što su S/PDIF i HDMI, a ove sprege se takođe mogu naći na velikom broju opštenamenskih računara. Ipak, uočena su mnoga ograničenja u ovakvom načinu razmene signala, od kojih su najznačajnija:

- **Razlike u fizičkim arhitekturama računara** – različiti računari mogu podržavati različite audio sprege što onemogućava reprodukciju ispitnih sekvenci na velikom broju računara,
- **Zasebne programske podrške za audio sprege** – svaka audio sprega može imati zasebnu programsku podršku kojom se kontroliše rad sprege, samim tim je potrebno

prilagođavati parametre ispitnih slučajeva različitim programskim podrškama, što dodatno komplikuje proces reprodukcije ispitivanja,

- **Niski bitski protoci** – sprege koje računari najčešće podržavaju nisu namenjene za testiranje aktuelnih audio sistema koji očekuju izuzetno visoke bitske protoke; na primer, računari mogu podržavati S/PDIF spregu, ali ta sprega nije u mogućnosti da vrši prenos stereo signala karakteristika boljih od 192kHz/24b, što znatno sužava izbor signala koji se mogu koristiti prilikom testiranja,
- **Mali broj kanala** – aktuelni audio sistemi imaju velik broj ulaznih i izlaznih linija za podatke, tj. podržavaju veliki broj audio kanala, dok računari u najvećem broju slučajeva ne podržavaju više od 2 kanala za snimanje i 2 kanala za puštanje po jednoj audio sprezi,
- **Nepodržavanje I2S sprega** – jedna od najčešće korišćenih sprega za ispitivanje audio sistema jeste I2S sprega, a upravo ova sprega se veoma retko može naći na računarima opšte namene,
- **Nestabilan izvor signala** – audio sistemi koji se ispituju zahtevaju veoma stabilan i pouzdan izvor signala koji se dovodi na ulaze, jer se u suprotnom može ugroziti rad samih sistema; audio sprege na računarima ne pružaju dovoljan nivo pouzdanosti, jer taj zahtev nije u centru pažnje prilikom razvoja arhitekture računara.

Upravo ova ograničenja ukazuju na to da bi ispitivanja u praksi ipak trebalo vršiti na neki drugi način. Najzastupljeniji način razmene audio podataka između računara i ciljne platforme jeste pomoću namenskog uređaja koji služi kao posrednik između ova dva sistema. Uloga ovog uređaja svodi se na prsoleđivanje signala, tj. reprodukciju i snimanje signala putem audio sprega koje bira korisnik, pa se samim tim oslovljava kao uređaj za reprodukciju i snimanje signala (eng. *player/recorder, P/R*). Na slici 2 prikazana je topologija ispitnog okruženja koja se tipično može sresti u praksi.



Slika 2. Topologija ispitnog okruženja u praksi

Upotrebom ovih uređaja zaobilaze se svi problemi koji se javljaju prilikom direktne razmene signala. Najznačajnije karakteristike uređaja za reprodukciju i snimanje signala su:

- **Nezavisnost od fizičkih karakteristika računara** – ovi uređaji objedinjuju više različitih sprega na jednoj platformi i na taj način u potpunosti čine ispitivanja audio sistema nezavisnim od hardverske arhitekture računara putem kojih se ispitivanja vrše,

- **Jedinstvena programska podrška** – za sve sprege koje ovi uređaji podržavaju postoji samo jedna programska podrška, a biranje sprega se vrši jednostavnim zadavanjem odgovarajućih parametara u toj programskoj podršci
- **Veliki broj kanala i visoki bitski protoci** – na ovim uređajima se može nalaziti prizvoljan broj ulaznih i izlaznih linija za podatke (audio kanala) čime se omogućava rad sa visokim bitskim protocima i signalima sa većim brojem kanala,
- **Stabilna I2S podrška** – ove platforme najčešće podržavaju I2S spregu, štaviše, stabilnost ove sprega jedan je od nabitnijih zahteva uređaja,
- **Stabilnost signala** – za razliku od računara, ovi uređaji su projektovani namenski za audio aplikacije, pa je iz tog razloga posebna pažnja usmerena ka ostvarivanju izuzetno stabilnih signala za podatke, a posebno takt signala.

Korišćenje platformi za reprodukciju i snimanje se u osnovi zasniva na zadavanju datoteke koja sadrži ulazne audio podatke, naziva datoteke u kojoj će biti smešteni izlazni audio podaci, biranju sprega putem koje će se vršiti reprodukcija i/ili snimanje i zadavanju odgovarajućih parametara za reprodukciju i snimanje (frekvencija, širina odbirka, broj kanala..). Razmena podataka između računara i platforme se obavlja putem sprega koje su za računare znatno karakterističnije i stabilnije kao što su ethernet (eng. *Ethernet, ETH*) i univerzalna serijska sprega (eng. *Universal Serial Bus, USB*). Uređaj takođe vrši prenos audio podataka prema zadatim korisničkim parametrima do i od samog sistema koji se testira. Zahvaljujući svim navedenim karakteristikama ovih uređaja, moguće je praktično putem svakog opštenamenskog računara lako i brzo izvršiti ispitivanja i identične reprodukcije testova uz minimalne resurse. Ti resursi podrazumevaju sam P/R uređaj, odgovarajuću programsku podršku i računar koji podržava sprega kao što su ETH i USB.

2.4.2 Postojeći uređaji za reprodukciju i snimanje signala

Na tržištu postoji nekoliko uređaja za puštanje i snimanje signala koji se koriste u procesima razvoja i ispitivanja audio sistema. U ovom radu će biti predstavljena dva koja se najčešće mogu sresti u praksi:

- APx, audio analizator kompanije Audio Precision, Inc
- Echo Fireworks ASIO zvučna kartica kompanije Echo Digital Audio Corporation

2.4.2.1 Audio analizator APx

Američka kompanija Audio Precision, Inc. proizvodi vodeće audio uređaje za snimanje i reprodukciju signala na tržištu pod nazivom APx [13]. Ovi uređaji takođe vrše i analizu i merenja performansi audio sistema (u realnom vremenu) kao što su amplituda, nivo šuma, harmonijske distorzije, frekvencijski odzivi i dr. Postoji nekoliko serija ovih uređaja a najznačajnije su: APx555, APx52x, APx58x i APx515. Za detaljnije karakteristike ovih modela pogledati [14]. APx sistemi dolazi zajedno sa pratećim softverom (za računar) koji ih svojim širokim skupom funkcionalnosti po pitanju audio analize i metrike čini audio analizatorima. Radi

se o naprednom i pouzdanom softveru za audio analizu koji omogućava da ovi sistemi u potpunosti zadovolje sve uslove koje ispitivanje kompleksnih softverskih sistema zahteva. Razlog zašto su ovi uređaji nepristupačni širem skupu audio inženjera jeste cena. Bez ugrađenih sprega, cena samog uređaja može prevazići i 30000\$, dok se sa kupovinom sprega može dostići cifra veća i od 50000\$. Takođe, veličina i težina uređaja čine ga teško prenosivim i nekompaktnim što predstavlja potencijalnu manu.

2.4.2.2 Echo ASIO zvučna kartica

Drugi uređaj koji se često sreće u praksi jeste Echo Fireworks ASIO zvučna kartica američke kompanije Echo Digital Audio Corporation [15]. Kako imaju znatno uži skup funkcionalnosti po pitanju audio analize, ASIO kartice se više smatraju uređajima za reprodukciju i snimanje signala. Prateći softver koji dolazi uz ove uređaje svodi se na osnovnu komandnu aplikaciju kojoj se zadaju osnovni parametri za reprodukciju i snimanje. Za dodatne analize potrebni su nezavisni alati. Podrška za digitalne sprege (HDMI, S/PDIF, I2S) prisutna je u svim modelima ovih uređaja, dok po pitanju stabilnosti signala i pouzdanosti rada sistema zadovoljavaju osnove kriterijume sa sprovođenjem velikog broja ispitivanja. Znatno su kompaktniji u odnosu na APx uređaje a cena im je takođe znatno pristupačnija (~1000\$). Kad je reč o nedostacima, treba napomenuti da je proizvodnja ovih uređaja prestala, pa ih je na tržištu sve teže pronaći, dok podrška za postojeće uređaje ne postoji. Takođe, zbog ograničenja po pitanju broja kanala za reprodukciju i snimanje (do 8) i širine odbiraka (16, 24), svojim performansama ASIO kartice ne mogu ispratiti najaktuelnije standarde u audio potrošačkoj elektronici.

3. Koncept rešenja

S obzirom na složenost današnjih audio tehnologija sa jedne strane i zahtevima tržišta za skraćenjem vremena pojavljivanja proizvoda na tržištu sa druge, javila se potreba za unapređenjem tipične ispitne procedure i radnog okruženja za ispitivanje složenih audio sistema. Cilj unapređenja je da se sam proces ubrza i automatizuje a da se u isto vreme poveća pouzdanost i preciznost eliminisanjem faktora ljudske greške i da se smanji potrošnja resursa prilikom ispitivanja složenih sistema.

U ovom poglavlju je najpre objašnjena tipična ispitna procedura i prikazan je hardverski sistem tipičnog radnog okruženja za ispitivanje složenih audio sistema. Detaljnije su opisani elementi hardverskog sistema. Potom je objašnjeno kojim izmenama je hardverski sistem unapređen a takođe je predstavljena i nova ispitna procedura. Akcenat je stavljen na uređaj RT-Audio Graber koji menja APx uređaj u hardverskom sistemu, kao i na RT-Executor koji se nalazi u ulozi kontrolne aplikacije prilikom automatizovanog izvršavanja ispitnih slučajeva. Na kraju je objašnjeno kako je automatizovan i proces analize snimljenih izlaza.

3.1 Tipična ispitna procedura

Kompanije koje obezbeđuju algoritme (eng. *IP provider*) u isto vreme obezbeđuju i pakete za ispitivanje tih algoritama. Tipična struktura paketa za ispitivanje se sastoji od dokumentacije, test materijala i alata koji se koriste. Kompanija, kao što je npr. Dolby, obezbeđuje QA inženjeru upitnik čijim popunjavanjem se dobija spisak ispitnih slučajeva koje je neophodno izvršiti kako bi se verifikovala ispravnost ispitivanog uređaja. QA inženjer specificira o kom se konkretnom

uređaju radi, koliko izlaznih kanala podržava taj uređaj, koji audio izlazi postoje, koje izlazne konfiguracije zvučnika podržava taj uređaj itd.

Popunjavanjem upitnika generiše se spisak ispitnih slučajeva, ispitni signali za svaki slučaj, konfiguracione poruke koje je neophodno poslati uređaju prilikom samog izvršavanja i druge informacije koje se smeštaju u nekoliko tekstualnih datoteka.

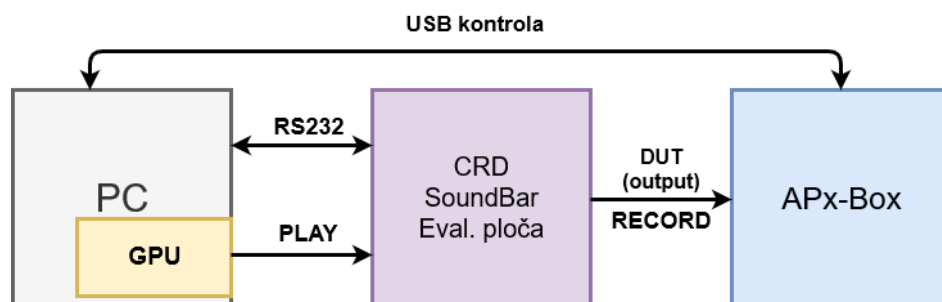
Zatim ispitni inženjer piše namensku Python skriptu kojom će parsirati ove informacije iz tekstualnih datoteka i smestiti ih u jedinstvenu Excel (.xlsx) tabelu. Excel tabela se kreira kako bi se sve informacije od značaja, za sve ispitne slučajeve, nalazile sumirane na jednom mestu.

Na osnovu ovih informacija QA inženjer piše Python skriptu koja služi kao šablon po kom se kreira Audio Precision (APx) projekat u kom se izvršavaju ispitni slučajevi. Sledeći korak je postavljanje i povezivanje hardverskog sistema, čija struktura je detaljnije objašnjena u narednom potpoglavlju. Sačinjavaju ga PC, APx, evaluaciona ploča ili konkretan uređaj koji se ispituje. PC vrši upravljanje grafičkom karticom i izvršavanje testova po koracima:

- podešavanje samog uređaja slanjem konfiguracionih poruka
- reprodukovanje ispitnog signala preko grafičke kartice
- merenje i analiza izlaza snimljenih APx mernim instrumentom sa ploče/uređaja

Svaki od navedenih koraka izvršava se zasebno za svaki ispitni slučaj, a samo izvršavanje je ručno. Izvršavanje jednog testa traje oko 5 min a broj ispitnih slučajeva može da ide od nekoliko stotina do nekoliko hiljada. Slanje konfiguracionih poruka je usko vremenski ograničeno i ukoliko se ne ispoštuje tačno definisan vremenski interval slanje neće biti uspešno. Sve ovo zahteva potpunu pažnju i manuelni rad QA inženjera, uz činjenicu da oduzima značajan broj radnih sati. Takođe sam APx, zbog svoje kompleksnosti, važi za veoma skup uređaj. Samim tim, evidentna je potreba i javlja se prostor za unapređenjem ove procedure.

3.2 Tipičan hardverski sistem



Slika 3. Tipičan hardverski sistem

U tipičnoj ispitnoj proceduri, prikazanoj na slici 3, učestvovali su:

- personalni računar, PC
- konkretan uređaj koji se ispituje ili evaluaciona ploča
- merni instrument Audio Precision (APx)

3.2.1 Audio Precision – APx58x

Na slici 4 dat je prikaz uređaja iz serije APx58x koja se odlikuje analognim i digitalnim spregama. Konkretno uređaj sa slike ima po 8 analognih ulaza i izlaza balansiranog i nebalansiranog tipa, digitalne ulaze/izlaze (S/PDIF) i digitalne serijske ulaze/izlaze (I2S) takođe balansiranog i nebalansiranog tipa, HDMI + ARC sprege i Bluetooth spregu. Sve sprege odlikuju se izuzetno stabilnim signalima. Takođe, postoji opcija da se ulazno/izlazne sprege fleksibilno menjaju (slotovi sa spregama se zasebno kupuju).

U ispitnoj proceduri nalazi se u ulozi mernog instrumenta, koji reprodukuje ili snima audio signale. Takođe vrši analize frekvencijskog odziva, nivoa glasnoće, nivoa šuma itd. Kako je napomenuto APx sistemi dolaze zajedno sa pratećim softverom. U tipičnoj ispitnoj proceduri, upravo se za ovaj softver kreira projekat u kom se posle izvršavaju testovi. APx ima ograničenje za veličinu fajla koji se reprodukuje od 200 MB i ne može da reprodukuje Dolby MAT tipove audio signala. Zbog ovih ograničenja, PC preko grafičke kartice vrši reprodukciju signala umesto APx-a.



Slika 4. Audio analizator iz serije APx58x

3.2.2 Evaluaciona ploča CRD498x4

Ploča CRD498x4, firme Cirrus Logic, predstavlja samostalni dekoderski sistem odnosno generični hardver kompletnog audio sistema – evaluaciona ploča [16]. Ovaj sistem može da

dekoduje neke od najpopularnijih formata audio signala koji se mogu pronaći u uređajima potrošačke elektronike kao što su Dolby Digital, Dolby Digital Plus, Dolby TrueHD, DTS, DTS HD, Dolby Atmos, DTS UHD. Takođe može da reprodukuje višekanalne PCM audio signale. Audio signali se do ploče šalju koristeći HDMI, S/PDIF sprege ili analogne ulaze. Ploča ima i ugrađenu korisničku spregu pomoću koje se biraju ulazi. Ploča i njene označene komponente prikazane su na slici 5 a ispod u tabeli 2 dat je i spisak svih komponenti.

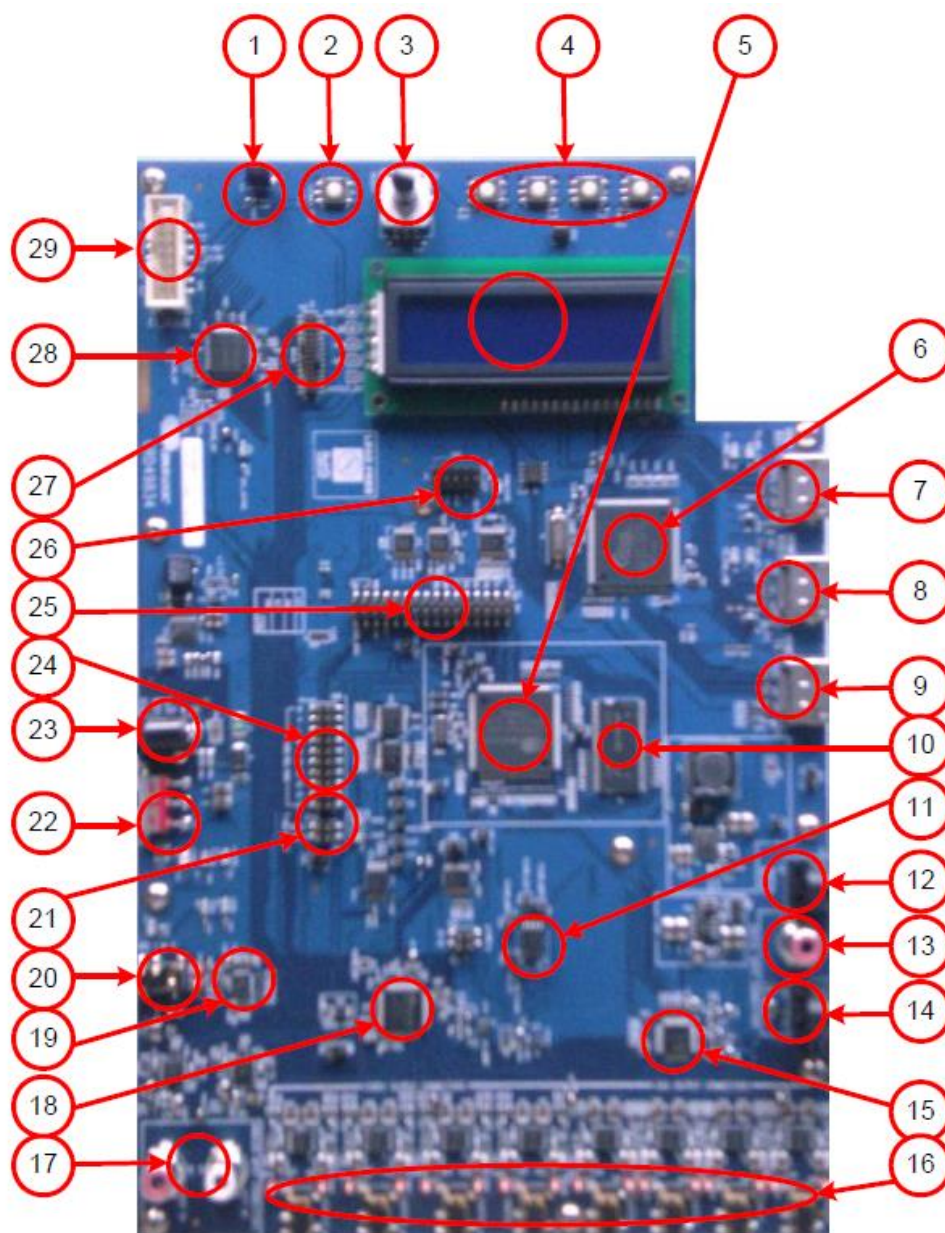


Figure 1-7. Identifying the CRD498X4 Components

Slika 5. Identifikovanje komponenti sa ploče CRD498x4[16]

Number	Component
1	IR receiver
2	Reset push button
3	Rotary encoder with push button
4	Four user interface push buttons
5	CS498xx DSP
6	HDMI transceiver
7	Input connector HDMI RX 1
8	Input connector HDMI RX 0
9	Output connector HDMI TX
10	External DSP SDRAM
11	DSP Serial Flash memory
12	TOSLINK S/PDIF output connector
13	S/PDIF RCA input connector
14	TOSLINK S/PDIF input connector
15	CS4385 DAC
16	3.5 mm (1/8") stereo, analog output connectors
17	Line in, RCA analog input connectors
18	CS42528 CODEC
19	CS53L21 ADC
20	MIC ic, 3.4 mm (1/8") stereo, analog input connector
21	DAO mux selector (uses 0.100" jumper for switching)
22	Power on/off switch
23	Power DC input connector
24	DAO output connector
25	DAI1 input connector (must be used with CRD498X4 DAI1 daughter board)
26	DAI3 input connector (connect J19.9 to GND to use this connector)
27	JP2 debug connector
28	MCU
29	MCU E1 debug connector

Tabela 2. Lista komponenti ploče CRD498x4

Ploča CRD498x4 korišćena je u svom specijalnom JP2 režimu rada. JP2 režim predstavlja specifično stanje ploče gde je ugrađeni CS498x4 DSP kontrolisan od strane PC-a preko CDB-MCU-DEBUG-B “dongle” (slika 6) dok ugrađeni mikrokontroler i dalje kontroliše CRD498x4 periferije kao što su CODEC, DAC, i sl.



Slika 6. CDB-MCU-DEBUG-B dongla

3.2.3 DSP CS49844 – Dagda čip

Digitalni signal procesor firme Cirrus Logic, pod nazivom CS49844 [17], korišćen je kao ciljni procesor u tipičnoj hardverskoj strukturi. DSP procesori familije CS49xxx namenjeni su obradi audio signala u uređajima potrošačke elektronike a zasnovani su na Crystal arhitekturi DSP jezgra. Razlike između pojedinih modela odnose se na broj jezgara, radni takt i količinu radne memorije.

Jedan od najmoćnijih procesora Crystal familije je upravo četvor jezgarni procesor CS49844 sa radnim taktom od 300 MHz. Zasnovan je na unapređenoj Harvard (eng. *Harward*) arhitekturi koja podrazumeva odvojenu memoriju za podatke i memoriju za instrukcije, dok je memorija za smeštanje podataka podeljena u dva segmenta. Svako jezgro predstavlja tridesetdvobitni programabilan DSP koji radi sa aritmetikom u fiksnom zarezu. Funkcionalni blok dijagram procesora CS49844 prikazan je na slici 7, a u tabeli 3 su predstavljeni podaci o memoriji.

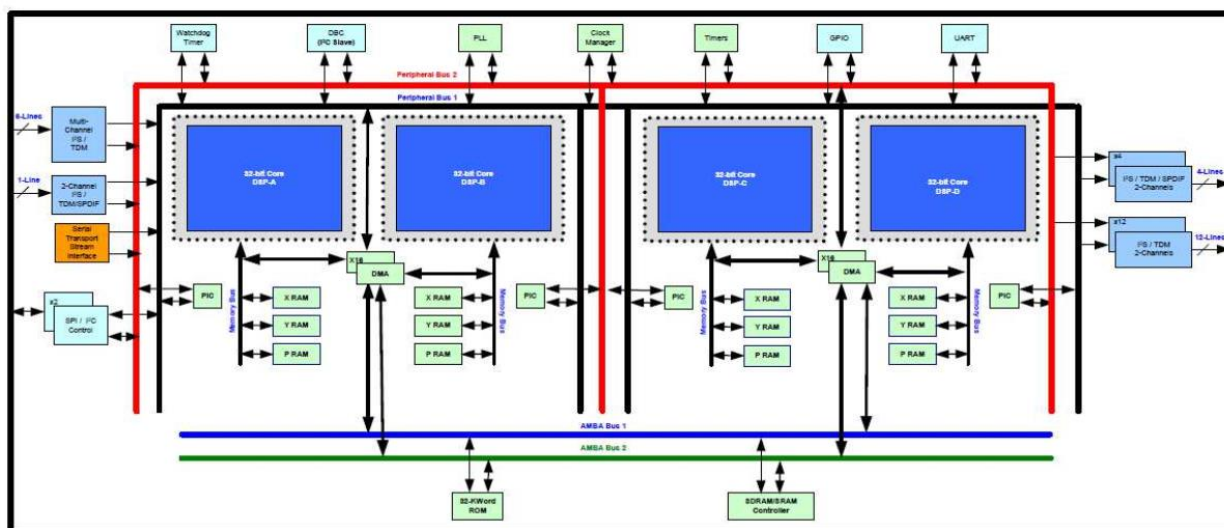


Figure 4-2. CS49844 Top-Level Block Diagram

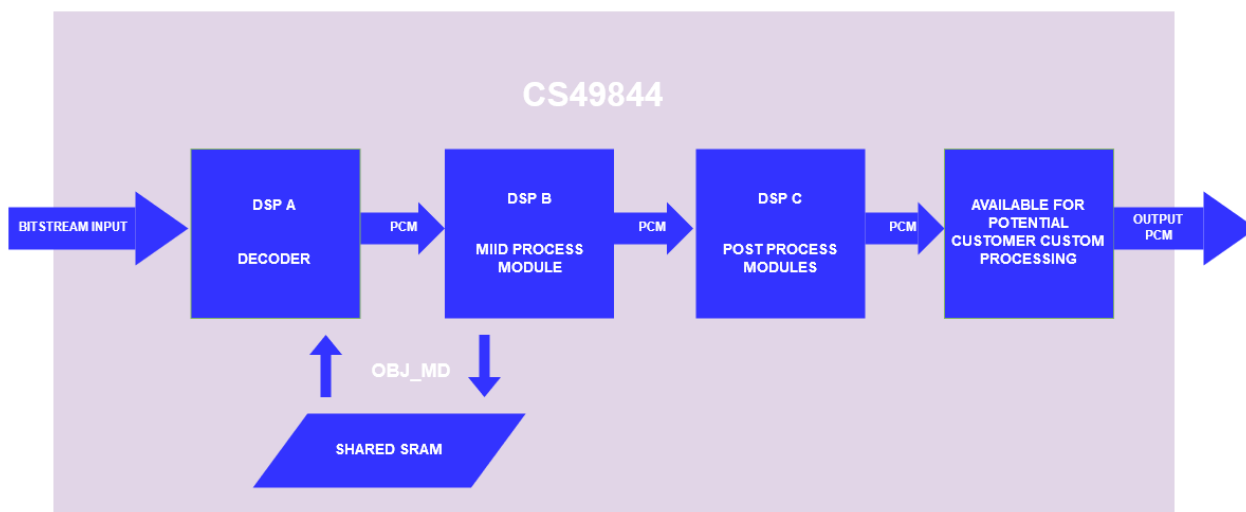
Slika 7. Blok dijagram procesora CS49844

Tip memorije	Jezgro A	Jezgro B	Jezgro C	Jezgro D
X,Y,P	60k Word SRAM	60k Word SRAM	60k Word SRAM	60k Word SRAM

Tabela 3. Raspoloživa memorija procesora CS49844

U jednom od ranijih poglavlja napomenuto je da su savremene audio tehnologije veoma zahtevne u pogledu memorije i MIPS-a (Million Instructions Per Second). Pojednostavljena blok

šema organizacije modula jedne audio tehnologije na ciljnoj platformi prikazana je na slici 8 a u nastavku poglavlja dat je i kratak opis funkcionisanja.



Slika 8. Blok šema funkcionisanja složene audio tehnologije na ciljnoj platformi

Na prvo procesorsko jezgro A, “spušta” se firmware jednog od dekodera. U zavisnosti od ulaznog bitskog toka bira se koji decoder će biti korišćen. U slučaju novih tehnologija zasnovanih na audio objektima, meta podaci se parsiraju a zatim smeštaju u deljenu memoriju odakle će ih kasnije koristiti modul na jezgru B. Na jezgru B izvršavaju se moduli među obrade (eng Mid-Process Module, MPM), koji su zaduženi za renderovanje na izlaznu konfiguraciju zvučnika. Renderovanje se obavlja koristeći PCM-ove i meta podatke sačuvane u deljenoj memoriji. Moduli završne obrade izvršavaju se na jezgru C. Završna obrada izvršava se nad podacima koji su rezultat prethodnih nivoa obrade. Podaci o potrošnji resursa ciljne platforme dati su u tabeli 4.

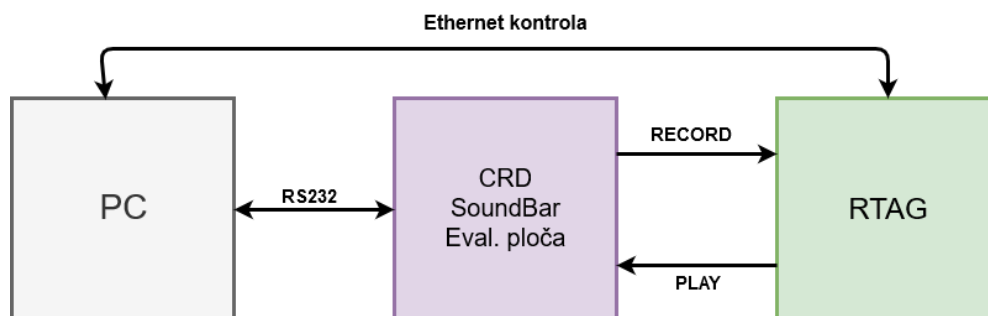
	Jezgro A	Jezgro B	Jezgro C
X [reči]	51216	57332	40548
Y [reči]	50654	37000	42868
P [reči]	50504	17222	29132
MIPS	~220	~200	~180

Tabela 4. Prikaz potrošnje resursa ciljne platforme

Iz podataka koji su dati u tabeli vidi se koliko su savremene audio tehnologije zahtevne, čak i kada su maksimalno optimizovane i implementirane u asemblerskom jeziku.

3.3 Unapređen hardverski sistem

Hardverski sistem, prikazan na slici 9 unapređen je zamenom APx-a sa audio uređajem RT-Audio Graber (RT-AG). RT-AG predstavlja eksternu zvučnu karticu koja obezbeđuje ulaze i izlaze audio signala, do i sa računara, pod kontrolom kompjuterskog programa/aplikacije [18]. Podržava kako reprodukciju tako i snimanje audio signala do 14 kanala na frekvencijama 192kHz/32b. Konekcija sa računarom ostvaruje se putem Ethernet-a. Primarno RT-AG je napravljen da se koristi u kombinaciji sa DSP evaluacionim pločama, ali je moguće korsistiti ga u paru sa krajnjim korisničkim audio uređajima kao što su Blu-Ray, Sound Bar, AVR itd. O samom uređaju detaljnije je napisano u narednom potpoglavlju.



Slika 9. Unapređen hardverski sistem

3.3.1 RT – Audio Graber

Ovaj uređaj razvijen je na institutu za Računarsku tehniku i računarske nauke sa ciljem da se nadomeste nedostaci postojećih sistema za reprodukciju i snimanje audio signala, kao što je na primer uređaj Echo ASIO. Tačnije RT-AG predstavlja napredniju verziju ASIO kartice.

Ideja za nastanak ovog uređaja je bila da se omogući reprodukcija i snimanje što je moguće šireg spektra audio signala kako bi pomoću njega bilo moguće analizirati i razvijati audio sisteme starijih i novijih generacija. U taj spektar spadaju kako signali odmereni na niskim učestanostima (32, 44.1, 48 kHz) i sa malim brojem kanala, tako i signali odmereni na visokim učestanostima (176.4, 192 kHz) i sa velikim brojem kanala (do 14). Sa podrškom za sve

digitalne audio sprege koje su najčešće u upotrebi (I2S, HDMI, S/PDIF), RT-AG je prilagodljiv veoma velikom broju različitih platformi. Pored ovih sprega, postoji i nekoliko sprega za povezivanje sa računarom od kojih su najznačajnije ETH i USB sprege.

Uređaj podržava sledeće digitalne serijske audio formate:

- I2S i varijante I2S formata – RJ (eng. *Right-Justified*), i LJ (eng. *Left-Justified*) i TDM
- S/PDIF, snimanje i reprodukcija stereo signala do 96kHz i do 24 bita po odbirku
- HDMI, reprodukcija do 8 audio kanala na frekvencijama do 192kHz i sa širinama odbiraka do 24 bita

Pored digitalnih, uređaj podržava i istovremeno snimanje i reprodukciju putem analognih audio sprega sa istim performansama kao i putem I2S sprege (snimanje: 14/32b/192kHz, reprodukcija: 8/32/192kHz).

Uređaj podržava sledeće bitske protoke:

- snimanje 14 audio kanala na frekvenciji od 192kHz i sa širinom odbiraka od 32 bita, što daje bitski protok od 86 Mb/s (14 kanala * 32b * 192kHz)
- reprodukciju 8 audio kanala na frekvenciji od 192kHz i sa širinom odbiraka od 32 bita, što daje bitski protok od 49 Mb/s (8 kanala * 32b * 192kHz)
- istovremenu reprodukciju i snimanje (dupleks komunikacija), što daje ukupan bitski protok od 135 Mb/s

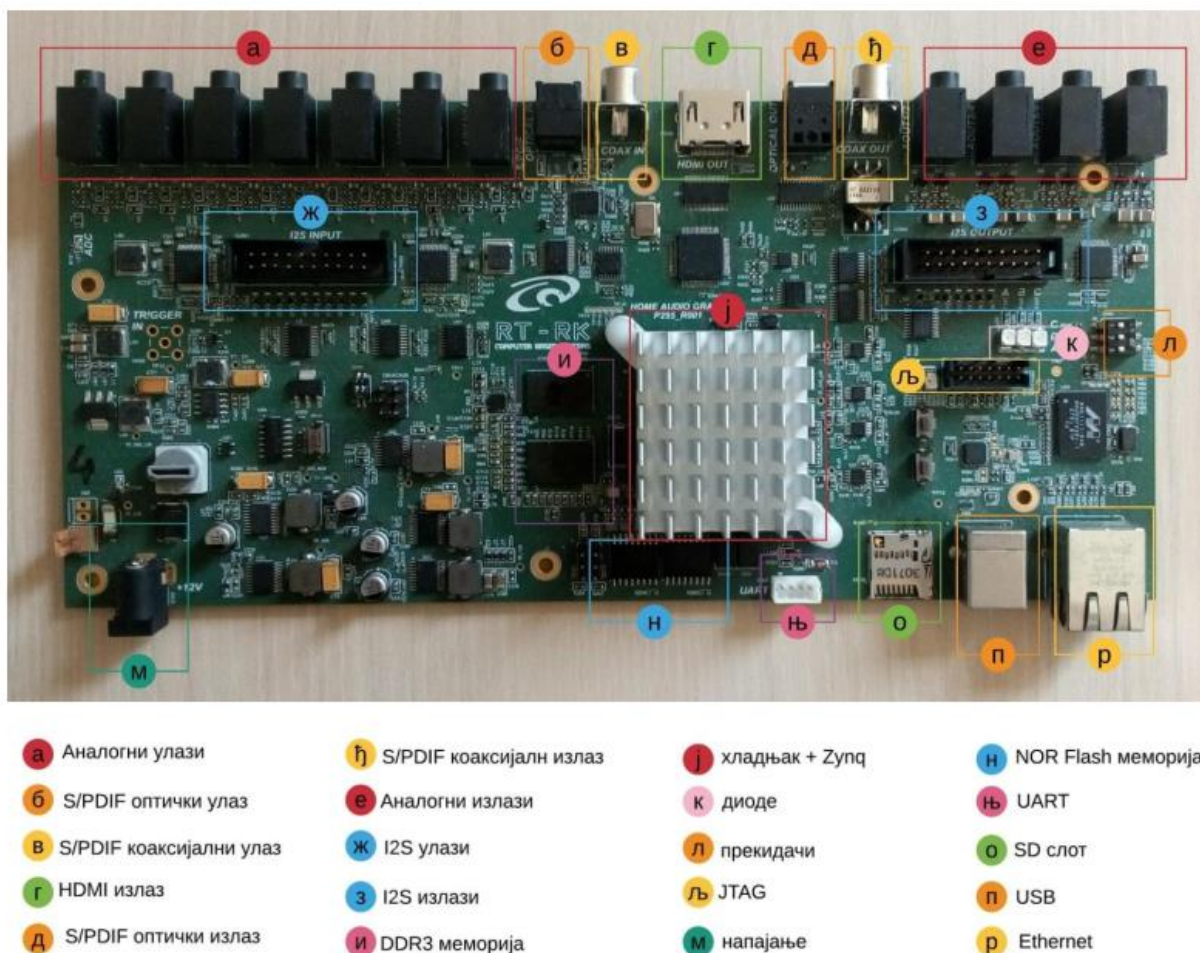
Po pitanju frekvencija, širina odbiraka i broja kanala uređaj podržava:

- reprodukciju signala na baznim frekvencijama (fs): 32, 44.1, 48 kHz i množiocima ovih frekvencija (2fs i 4fs): 64, 88.2, 96, 128, 176.4 i 192 kHz
- reprodukciju signala sa širinama odbiraka: 16, 24 i 32 bita
- reprodukciju 2, 4, 6 i 8 audio kanala
- snimanje signala na baznim frekvencijama (fs): 32, 44.1, 48 kHz i množiocima ovih frekvencija (2fs i 4fs): 64, 88.2, 96, 128, 176.4 i 192 kHz
- snimanje signala sa širinama odbiraka: 16, 24 i 32 bita
- snimanje 2, 4, 6, 8, 10, 12 i 14 audio kanala
- istovremenu reprodukciju i snimanje signala na nezavisnim frekvencijama, sa nezavisnim širinama odbiraka i brojem kanala
- merenje i prepoznavanje ulaznih takt signala

Fizička arhitektura sastoji od već pomenutih sprega tj. periferija, centralnog sistema i memorijskog sistema. Glavna komponenta centralnog sistema je hibridni sistem na čipu (eng. *System-on-Chip, SoC*) Zynq XC7Z030, iz familije čipova Zynq-7000 [19] kompanije Xilinx. Memorijski sistem uređaja sastoji se od nekoliko memorija: DDR3 SDRAM (1GB), BootROM (128KB), OCM (256KB), SD (2GB). DDR3SDRAM memorija je najznačajnija za rad samog sistema i u njoj se nalazi kod softverskog sistema koji se izvršava i sve privremene promenljive kojima se u softveru upravlja. Da bi se kod softverskog sistema uopšte našao u SDRAM memoriji potrebno je prvo učitati ga iz neke druge memorije. Slika (eng. *image*) za podizanje

sistema u kojoj je smešten i kod softverskog sistema inicijalno se nalazi u serijskoj NOR (Quad-SPI) FLASH memoriji, koja je značajna upravo prilikom podizanja sistema.

RT-AG platforma sa označenim najbitnijim komponentama sistema predstavljena je na slici 10.



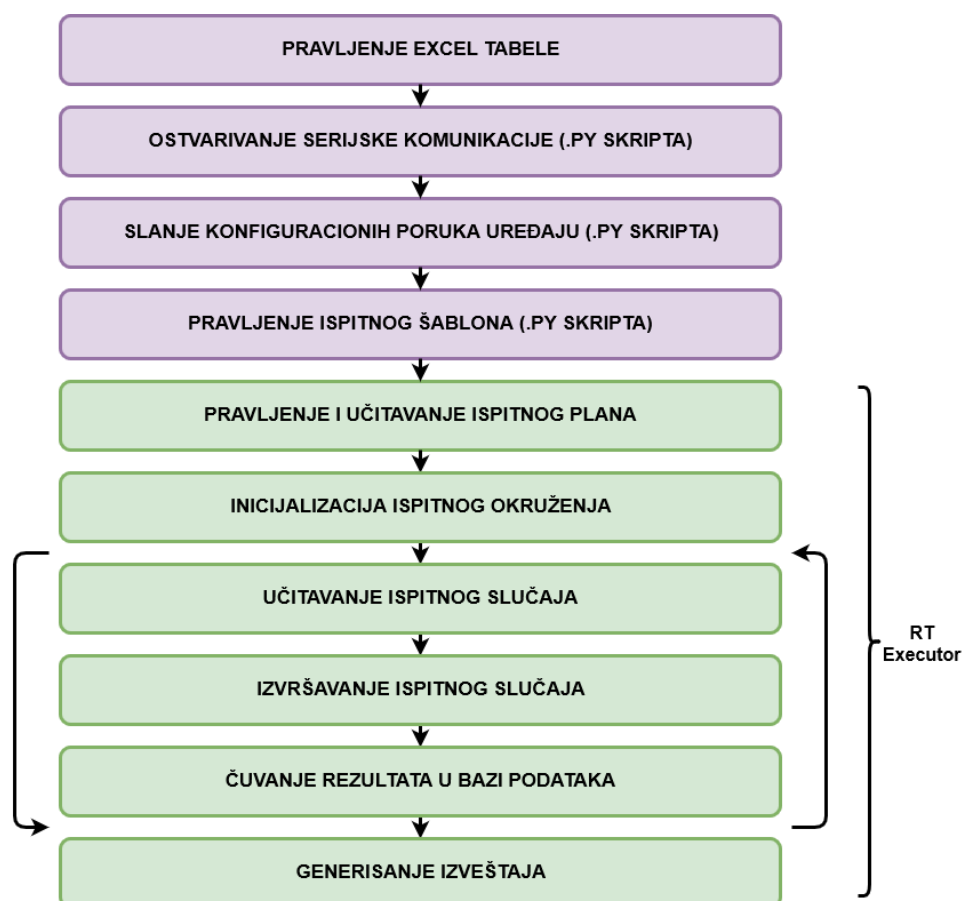
Slika 10. RT-AG platforma

3.4 Unapređena ispitna procedura

Redosled kreiranja i izvršavanja automatskih ispitnih slučajeva predstavljen je na slici 11. Za programsku podršku ispitivanju odabran je RT-Executor, program razvijen u istraživačko razvojnom institutu RT-RK, čime je postignuta automatizacija procesa samog izvršavanja ispitnih slučajeva. RT-Executor je samostalna aplikacija, osposobljena da kontroliše veliki broj softverskih i hardverskih modula neophodnih za automatsko ispitivanje [20]. Podržava ispitivanje po principu “crne kutije”.

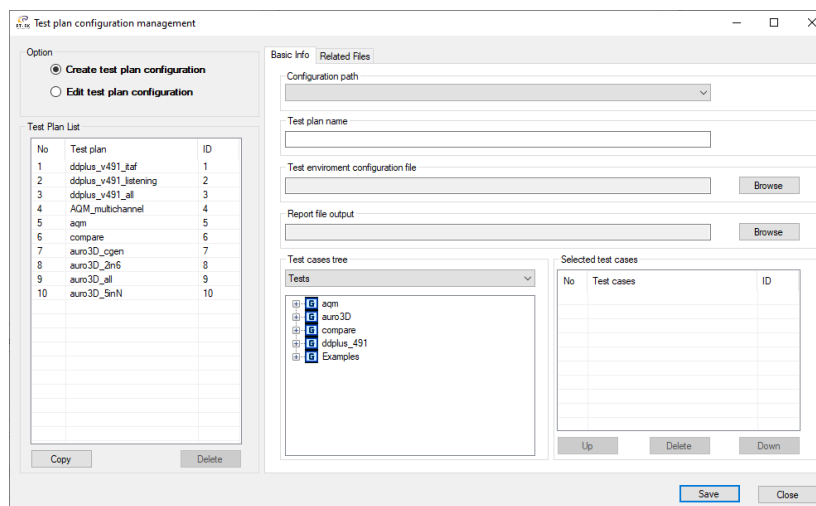
Na osnovu specifikacije ispitivanog uređaja pišu se testovi kojima će se ispitati da li uređaj uspešno izvršava sve predviđene operacije. Ispitni slučajevi se prave na osnovu šablona, napisanog u programskom jeziku Python, i redom izvršavaju sledeće korake za svaki ispitni slučaj:

- otvaranje serijskog porta,
- prebacivanje MCU (eng. *Microcontroller Unit, MCU*) u UART mode (eng. *Universal Asynchronous Receiver-Transmitter*),
- slanje odgovarajućih konfiguracionih poruka MCU,
- reprodukciju odgovarajućeg ispitnog signala,
- upisivanje konfiguracionih poruka iz MCU na DSP,
- snimanje izlaza.



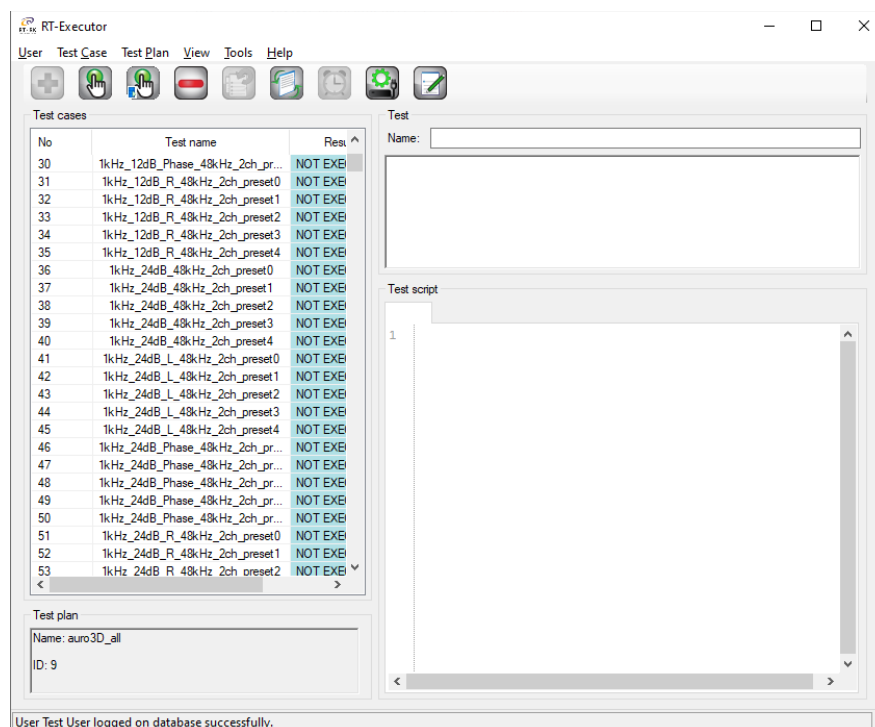
Slika 11. Prikaz unapređene ispitne procedure

Slanje konfiguracionih poruka obavlja se putem serijske komunikacije, a poruke za svaki ispitni slučaj mogu biti različite. Poruke se dobijaju pomoću namenske Python skripte koja parsira svaki ispitni slučaj posebno. Prethodno isparsirane poruke za promenu režima rada i individualne poruke, prvenstveno se šalju MCU koji ih dalje upisuje na DSP. Ove korake takođe izvršava zasebna Python skripta koja se poziva unutar testa za taj konkretan ispitivan slučaj.



Slika 13. Kreiranje ispitnog plana u RT-Executoru

Nakon što je ispitni plan uspešno napravljen, neophodno ga je učitati. Učitani ispitni plan prikazan je na slici 14.



Slika 14. Učitani ispitni plan

Ispitni slučajevi mogu biti izvršeni pojedinačno, po nekoliko odabranih ili svi ispitni slučajevi koji se nalaze u ispitnom planu odjednom. Nakon pokretanja jednog ispitnog slučaja, prikazuje se dodatni prozor kojim se prati napredovanje izvršavanja datog ispitnog slučaja. Rezultat izvršavanja može biti jedan od tri statusa: PASS, FAIL, UNRESOLVED koji jasno

govore da li je ispitni slučaj uspešno izvršen (PASS), neuspešno izvršen (FAIL) ili je njegov slučaj nerazrešen (UNRESOLVED). Ukoliko je ispitni slučaj iz određenih razloga neuspešno izvršen ili je status nerazrešen, QA inženjer kroz sam izveštaj može da vidi i opis greške koja postoji. Rezultati izvršavanja ispitnih slučajeva se mogu pratiti u toku izvršavanja otvaranjem History prozora prikazanog na slici 15.

No.	Test name	Result	Time	Test case folder	Python error
1	1kHz_0dB_48kHz_2ch_preset0	PASS	14/10/2019 17:54:49	Open	No Error
2	1kHz_0dB_48kHz_2ch_preset3	PASS	14/10/2019 17:55:39	Open	No Error
3	1kHz_0dB_Phase_48kHz_2ch_pre...	PASS	14/10/2019 17:56:29	Open	No Error
4	1kHz_0dB_Phase_48kHz_2ch_pre...	PASS	14/10/2019 17:57:19	Open	No Error
5	1kHz_12dB_48kHz_2ch_preset2	PASS	14/10/2019 17:58:10	Open	No Error
6	1kHz_0dB_6dB_48kHz_2ch_preset2	INCONCLUSIVE	14/10/2019 18:00:15	Open	PythonError.log
7	1kHz_0dB_48kHz_2ch_preset1	FAIL	14/10/2019 18:01:50	Open	PythonError.log

Slika 15. History prozor RT-Executora

Rezultati se, takođe, čuvaju i u bazi podataka u vidu HTML (Hyper Text Markup Language) stranice koja se otvara u nekom internet pretraživaču. Da bi proglasili da je uređaj uspešno ispitan, svi ispitni slučajevi moraju biti "PASS". U slučaju da nakon provere i ponavljanja ispitnih slučajeva rezultati ponovo nisu pozitivni, QA inženjer prijavljuje grešku timu koji razvija softver. Ispitni inženjer je dužan da detaljno opiše problem, kod kog ispitnog slučaja se javlja, odnosno koju funkcionalnost pogađa. Što više informacija tim koji razvija softver poseduje, lakše će locirati grešku. Nakon popravljanja greške, vrši se novi ciklus ispitivanja, kako bi se proverilo da je greška zaista otklonjena.

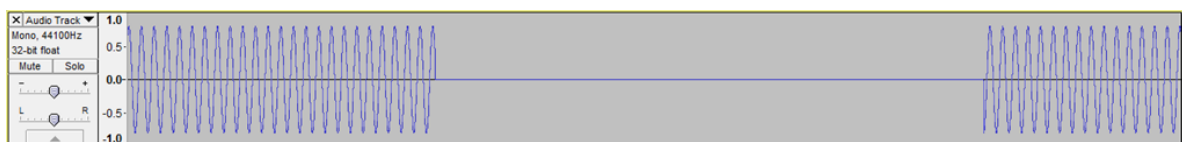
3.6 Analiza snimljenih izlaza

Analiziranje snimljenih izlaza je u tipičnoj ispitnoj proceduri obavljano manuelno od strane QA inženjera otvaranjem svakog ispitnog vektora zasebno u nekom od alata kao što je npr. Adobe Audition. U unapređenoj ispitnoj proceduri ovaj korak je automatizovan takođe. RT-Executor pomoću svojih ugrađenih alata omogućava jednostavnu analizu audio izlaza sa ispitivanog uređaja, po svakom snimljenom kanalu. Jedan od takvih alata je Audio Quality Meter (AQM).

AQM je softverski dodatak za RT-Executor koji u sebi ima audio algoritme za prepoznavanje različitih audio artifakta kao što su isecanje, prekidi, zamrzavanje i izostanak audio signala. Na kraju svakog ispitnog slučaja vrši se pozivanje određenog AQM audio

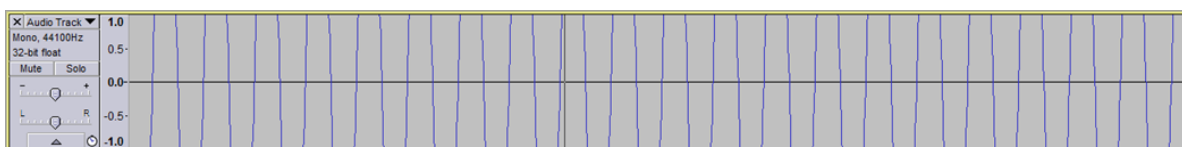
algoritma koji nam javlja da li je prisutan neki od artifakta. Ukoliko ne postoje, RT-Executor proglašava ishod ispitnog slučaja kao PASS, ili u suprotnom će ishod biti FAIL.

Algoritam odsustva zvuka analizira audio signal za nultu vremensku aktivnost. Ako audio signal nedostaje duže od navedenog vremena, detektuje se odsustvo. Postoje dva tipa algoritama odsustva: provera odsutnosti u vremenskim i frekvencijskim domenima. Algoritam takođe podržava dodavanje nivoa buke u dB (podrazumevano -60dB) koju treba zanemariti tokom analize. Primer odsustva zvuka u audio signalu prikazan je na slici 16.



Slika 16. Primer odsustva zvuka u audio signalu

Isecanje predstavlja izobličenje talasnog oblika koji se javlja kada je pojačalo preopterećeno i pokušava da isporuči izlazni napon ili struju van svojih opsega. Algoritam za audio isecanje u RT-Executoru vrši analizu u vremenskim i frekvencijskim domenima i detektuje periode kada je signal isečen zbog vrlo viskog nivoa amplitude. Primer takvog signala dat je na slici 17.



Slika 17. Primer isecanja u audio signalu

4. Programsko rešenje

U ovom poglavlju biće opisan celokupan proces kreiranja ispitnih slučajeva. Najpre će biti prikazano na koji način je organizovano radno okruženje i sama struktura direktorijuma u kojima će biti smešteni ispitni slučajevi i alati neophodni da bi se isti izgenerisali. Potom će se detaljno objasniti na koji način se ispitni slučajevi generišu kao i koje korake će ispitni slučajevi izvršavati. Poseban akcenat biće dat procesu kreiranja konfiguracionih poruka koje simuliraju aktivnosti krajnjeg korisnika.

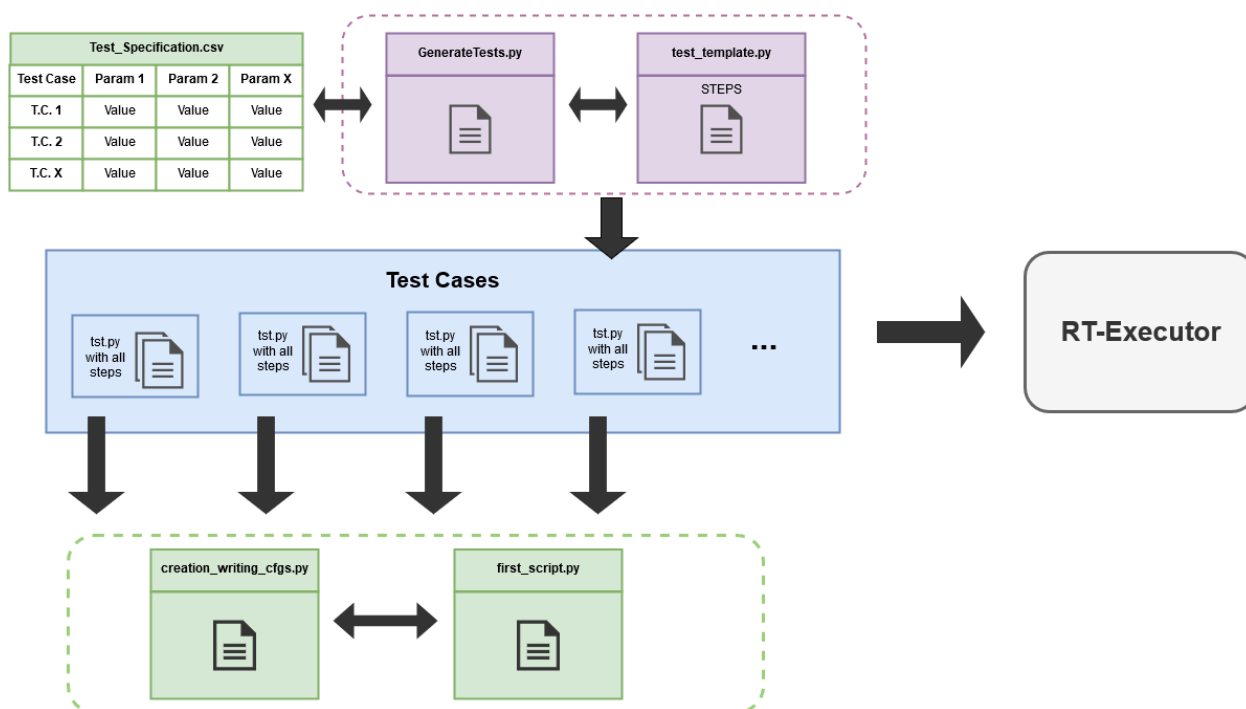
4.1 Organizacija radnog okruženja

Početni korak u fazi razvoja ispitnih slučajeva jeste organizacija direktorijuma radnog okruženja. Spram specifičnosti i zahteva koje zadaju i radni okvir i RT-Executor, kao alat u kome će se ispitni slučajevi izvršavati, struktura direktorijuma je organizovana na sledeći način:

- U **docs** direktorijumu nalaze se sve neophodne tekstualne datoteke koje sadrže instrukcije i uputstva za ceo proces ispitivanja, počev od generisanja referentnih izlaza, generisanja ispitnih slučajeva, uputstava za proces izvršavanja ispitnih slučajeva kao i za poređenje referentnih i izlaza snimljenih na ciljnoj platformi.
- **Install** direktorijum, sadrži instalacione datoteke svih aplikacija i alata neophodnih za izvršavanje procesa ispitivanja kao što su na primer Python, programski jezik na kom je zasnovan ceo radni okvir, zatim RT-Executor itd..
- **Test_templates** direktorijum sadrži sve datoteke i skripte neophodne kako bi se izgenerisali ispitni slučajevi. O ovim skriptama i datotekama biće više reči u nastavku poglavlja.
- **Tests** direktorijum sadrži sve ispitne slučajeve podeljene po grupama (ukoliko postoje različite grupacije ispitnih slučajeva). Svaki ispitni slučaj sačuvan je u posebnom direktorijumu, čiji je naziv isti kao i naziv ispitnog slučaja (u većini slučajeva).

- **Tools** direktorijum sadrži sve alate, skripte i konfiguracione datoteke koje će se pozivati tokom izvršavanja ispitnih slučajeva.
- **Working** direktorijum ima strukturu koja je identična folderu **Tests**, sadrži poddirektorijume sa nazivima grupa ispitnih slučajeva i u njima poddirektorijume sa nazivima konkretnih ispitnih slučajeva. Ovi poddirektorijumi su prazni sve dok se ispitni slučaj ne izvrši u RT-Executoru, dok se nakon izvršavanja ovde čuvaju izlazni ispitni signali kao i log dateteke.

4.2 Generisanje ispitnih slučajeva



Slika 18. Blok dijagram procesa generisanja ispitnih slučajeva

Na slici 18 prikazan je blok dijagram koji ilustruje ceo proces generisanja ispitnih slučajeva.

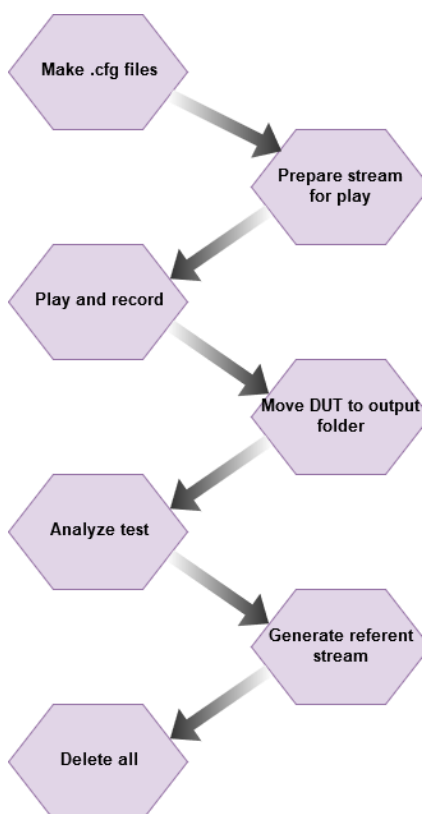
Za generisanje ispitnih slučajeva koristi se Python skripta `GenerateTests.py`. Kako je napomenuto u poglavlju 3.1 (Tipična ispitna procedura), skripta parsira Excel tabelu u kojoj svaki red sadrži sve neophodne informacije i parametre za taj ispitni slučaj. Prolazeći red po red kroz skriptu, skripta smešta ove informacije u promenljive koje posle koristi `test_template.py` skripta.

`GenerateTests.py` skripta uključuje (eng. *import*) drugu skriptu `test_template.py` koja u sebi sadrži implementiranu logiku pomoću koje se generišu ispitni slučajevi. Ova skripta će poslužiti

kao šablon po kom će se generisati sami ispitni slučajevi sa svim svojim koracima koje je potrebno izvršiti (.py datoteke).

Kao krajnji rezultat izvršavanja skripte `GenerateTests.py` dobijaju se ispitni slučajevi u formatu Python skripti od kojih se posle pravi RT-Executor projekat.

Na slici 19 prikazan je blok dijagram, koji ilustruje korake jednog ispitnog slučaja. Kako svaki ispitni slučaj može imati određene specifičnosti, na dijagramu su prikazani osnovni koraci koji se izvršavaju za svaki ispitni slučaj.



Slika 19. Blok dijagram koji ilustruje korake ispitnog slučaja

Make .cfg files – u ovom koraku izvršava se proces generisanja konfiguracionih poruka, tačnije komandi koje su njima navedene, a pomoću kojih se uključuju odnosno isključuju pojedine funkcionalnosti. Konfiguracione poruke mogu biti individualne i poruke za promenu režima rada. U ovom koraku se poziva `creation_writing_cfgs.py` skripta koja dalje poziva `first_script.py`. Detaljnije o ovom procesu generisanja biće reči u narednom poglavlju.

Prepare stream for play – ovaj korak podrazumeva sve aktivnosti koje treba uraditi kako bi se ulazni ispitni vektor pripremio za reprodukciju i snimanje. Neretko se dešava da se ulazni ispitni vektor nalazi u formatu koji se ne može reprodukovati a da se prethodno ne dekoduje. U nekim slučajevima se javlja potreba da se poveća frekvencija odabiranja kod ulaznog vektora.

Ponekad je potrebno spojiti (eng. *combine*) dva ulazna ispitna vektora kako bi se kreirao novi ispitni vektor, ili postojeći ispitni vektor podeliti (eng. *split*) po kanalima.

Play and record – u okviru ovog bloka izvršavaju se koraci koji podešavaju RT-AG, uređaj koji služi za reprodukciju i snimanje. Neki od parametara koje je potrebno postaviti su putanja i naziv ulaznog ispitnog vektora, putanja i naziv izlaznog ispitnog vektora, frekvencija na kojoj se pušta, broj kanala ulaznog i izlaznog ispitnog vektora itd. Uređaj RT-AG ima ograničenje da može da snima do 14 audio kanala i da pušta do 8 audio kanala. U aktuelnim složenim audio tehnologijama nije čudno da se pojavi ispitni vektor i do 35 kanala. U tom slučaju snimanje i reprodukcija se moraju vršiti u nekoliko iteracija. Samim tim izvršavanje takvog ispitnog slučaja traje duže jer je pre naredne iteracije snimanja potrebno ponovo konfigurisati ciljnu platformu na odgovarajući način, kao i izvršiti dodatne korake nakon snimanja izlaznog vektora.

Move DUT to output folder – nakon što je izlazni ispitni vektor snimljen, neophodno ga je prebaciti u direktorijum gde se čuvaju svi izlazni vektori (Working direktorijum). Svaki izlazni vektor ima svoj poddirektorijum čiji je naziv isti kao naziv ispitnog slučaja. Ovaj korak je neophodan kako bi se svi izlazni ispitni vektori nalazili na unapred određenoj lokaciji čime se olakšava dalja analiza.

Analyze test – nad snimljenim izlaznim ispitnim vektorima neophodno je izvršiti analizu, odnosno proveriti da li se pojavljuju neki od audio artifakta koji su od značaja. Analiza audio izlaza sa ispitivanog uređaja detaljno je opisana u poglavlju 3.6.

Generate referent stream – u ovom koraku vrši se generisanje referentnih ispitnih slučajeva pozivanjem određene izvršne datoteke, uglavnom dobijene od IP providera.

Delete all – u okviru ovog bloka vrši se brisanje svih privremenih datoteka izgenerisanih tokom izvršavanja ispitnog slučaja. Imajući u vidu složenost današnjih audio tehnologija i veliki broj ispitnih slučajeva koje je potrebno izvršiti prilikom ispitivanja istih, memorija koju zauzmu ove privremene datoteke nije zanemarljiva. I ako nije neophodan, ovaj korak se pokazao kao dobra praksa.

4.2.1 Generisanje individualnih i poruka za menjanje režima rada

Poruke za menjanje režima rada i individualne poruke ustvari simuliraju podešavanja koja zadaje krajnji korisnik prilikom korišćenja složenog audio sistema. Neka od mogućih podešavanja su:

- Dolby audio processing mode – movie/music/night
- Surround decoder (Upmixer) - enable/disable
- Speaker virtualizer - enable/disable

- Postprocessing - enable/disable
- Dynamic range control - enable/disable/auto
- Direct decoding mod - enable/disable
- Loudness management - enable/disable

Nakon popunjavanja upitnika, za svaki ispitni slučaj je tačno određeno koja od ovih podešavanja treba da budu uključena (eng. *Enable*) a koja isključena (eng. *Disable*) i koja vrednost im je podrazumevana (eng. *default*). Ove informacije se izvršavanjem Python skripte parsiraju i smeštaju u Excel tabelu.

GenerateTests.py skripta ce proslediti informacije o podešavanjima test_template.py skripti. Prilikom izvršavanja ispitnog slučaja, u koraku **Make .cfg files**, će biti pozvana skripta creation_writing_cfgs.py kojoj će kao parametar biti prosledena sva podešavanja koja je potrebno izvršiti i njima pridružene vrednosti (npr. enable/disable). Skripta najpre prolazi „for“ petljom kroz ove informacije i parsira ih u rečnik (eng. *dictionary*). U nastavku prikazano je parče koda koje će popuniti rečnik.

```
import time
import first_script

def make_tst(controls_line, row):

    first_script.windowclass1(row)

    out_mc_messg = 'appwr 0x4A 0x00 EF000000000000100'
    out_in_messg = 'appwr 0x4E 0x00 EF000000000000100'
    dict = {}
    dolby_names = ['Dolby audio processing mode', 'Surround Decoder (Upmixer)', 'Speaker
Virtualizer', 'Postprocessing',
                  'Dynamic Range Control', 'Direct decoding mode', 'Loudness management']

    for i in dolby_names:
        if controls_line.find(i) != -1:
            start = controls_line.find(i) + len(i) + 2
            end = start + 7
            values = controls_line[start:end]
            stripped_value = values.replace(' ', '')
            dict[i] = stripped_value
        else:
            continue
```

Individualne i poruke za promenu režima rada se šalju MCU tako što se kroz komandnu liniju pošalje niz karaktera i hexa vrednosti koje predstavljaju vrednosti podešavanja koja je potrebno simulirati. Kako bi se ove poruke sastavile neophodno je „for“ petljom proći kroz prethodno sastavljeni rečnik i u zavisnosti od vrednosti konkretnog podešavanja “nalepiti” odgovarajuću hexa vrednost. Deo koda koji vrši ovu aktivnost prikazan je u nastavku.

```

for k, v in dict.items():

    if k == 'Postprocessing':
        if v == 'disable':
            out_in_messg += 'BA800000FFECEFFF'
        elif v == 'enable' or v == 'default':
            out_in_messg += 'BA40000000131000'

    elif k == 'Surround Decoder (Upmixer)':
        if v == 'disable':
            out_in_messg += 'BA800000FFFFFFEF'
        elif v == 'enable' or v == 'default':
            out_in_messg += 'BA400000000000100'

    elif k == 'Loudness management':
        if v == 'enable' or v == 'default':
            continue
        elif v == 'disable':
            out_in_messg += '9700000400000000' + '9D00000700000001' + '9C00000200000002'
            out_in_messg += '9C00000300000000' + '9C00000400000000'

```

Nakon što je sastavljena poruka za promenu režima rada kao i individualna poruka sa svim neophodnim vrednostima podešavanja, potrebno ih je putem serijske komunikacije (RS232) poslati MCU i verifikovati da li su ove poruke uspešno poslate. Deo koda koji vrši ovu aktivnost prikazan je ispod.

```

#Write prepared mode change message via RS232
first_script.ser.write(out_mc_messg)
temp_msg = first_script.ser.read(200)
cnt = 100
while True:
    if cnt == 0:
        print "No mode change"
        break
    if '>NOK<' in temp_msg:
        print "NOK"
        break

    elif '>MOK<' in temp_msg:
        print "MOK"
        break

    cnt -= 1
    temp_msg = first_script.ser.read(200)

time.sleep(5)

#Write prepared individual message via RS232
first_script.ser.write(out_in_messg)
temp_msg = first_script.ser.read(200)
while True:
    if cnt == 0:
        print "No individual change"
        break
    if '>IOK<' in temp_msg:
        print "IOK"
        break

    elif '>NOK<' in temp_msg:
        print "NOK"
        break

    cnt -= 1
    temp_msg = first_script.ser.read(100)

```

Slanje konfiguracionih poruka MCU obavlja skripta `first_script.py`. Ova skripta zapravo inicira serijsku komunikaciju, otvara odgovarajući port, podešava ga i prebacuje evaluacionu ploču u UART režim koji služi za komunikaciju sa MCU. Ispod je prikazan kod čijim izvršavanjem se realizuju navedene aktivnosti.

```
class windowclass1():
    def __init__(self,variable):
        global ser
        ser = serial.Serial(
            port='COM1',
            baudrate=19200,
            parity=serial.PARITY_NONE,
            stopbits=serial.STOPBITS_ONE,
        )
        ser.close()
        ser.open()
        ser.isOpen()

    if variable == 1:
        ser.write('appcon')
        time.sleep(2)
        out = ''
        cnt = 0
        # let's wait two seconds before reading output (let's give device time to answer)
        time.sleep(2)

        # while ser.inWaiting() > (80):
        out += ser.read(81)
        out_txt = out.replace("\r\n", "")
        cnt += 1
        if cnt == 1:
            if 'ef00004400000001' in out_txt:
                print 'Flash image verified'
            elif 'ef00004480000001' in out_txt:
                print 'Flash image verification failed'

        ser.write('appwr 0x52 0x01')
        time.sleep(0.5)
        temp_msg = ser.read(200)
        cnt = 100

    while True:
        if cnt == 0:
            print "Message mode IN"
            break
        if '>OK<' in temp_msg:
            print "OK"
            break

        elif '>NOK<' in temp_msg:
            print "NOK"
            break
```

5. Rezultati

Kako bi se uvidelo u kojoj meri automatizacija radnog okruženja i unapređena ispitna procedura doprinose i ubrzavaju ispitivanje složenih audio sistema, rezultati dobijeni primenom nove metodologije poređeni su sa rezultatima dobijenim primenom stare metodologije po nekoliko ključnih tačaka:

- uticaj na pouzdanost ispitivanja
- ušteda vremena
- smanjenje potrošnje resursa

Predloženo radno okruženje i ispitna procedura primenjena je prilikom ispitivanja nekoliko različitih audio tehnologija sa širokim spektrom različitih ispitnih signala, kao i različitom izlaznom konfiguracijom zvučnika za različite audio uređaje. U tabeli 5 dati su rezultati ispitivanja SoundBar uređaja na audio tehnologiji Dolby Atmos HT 1.6.2:

- za odabranu konfiguraciju zvučnika 2.1
- za odabranu konfiguraciju zvučnika 5.1.2
- za odabranu konfiguraciju zvučnika 7.1.4

Izlazna konfiguracija zvučnika	Primenjivana metodologija ispitivanja	Broj ispitnih slučajeva	Vreme potrebno za izvršavanje jednog slučaja	Broj ponavljanih slučajeva	Ukupno vreme potrebno za ispitivanje jednog uređaja
2.1	APx bazirana metodologija TIPIČNA	113	3 min	10	~ 6h
2.1	RT-AG bazirana metodologija UNAPREĐENA	113	1.5 min	0	~ 3h
5.1.2	APx bazirana metodologija TIPIČNA	138	5 min	15	~ 13h
5.1.2	RT-AG bazirana metodologija UNAPREĐENA	138	3 min	0	~ 8h
7.1.4	APx bazirana metodologija TIPIČNA	145	6 min	18	~ 16h
7.1.4	RT-AG bazirana metodologija UNAPREĐENA	145	3.5 min	0	~ 8,5h

Tabela 5. Rezultati ispitivanja SoundBar uređaja na audio tehnologiji Dolby Atmos HT 1.6.

U tabeli 5 jasno se vidi kako ponuđeno rešenje utiče na pouzdanost samog ispitivanja. Na broj ispitnih slučajeva koji se moraju ponovljati najviše je uticao faktor ljudske greške koji je u novoj proceduri, upotrebom RT-Executora, uspešno eliminisan. Kako je izvršavanje ispitnih slučajeva u staroj metodologiji obavljano manualno, neretko se dešavalo da se određen broj slučajeva mora ponoviti. Ponavljanja su najčešće bila uzrokovana neadekvatnim trenutkom slanja konfiguracionih poruka, čiji je interval usko vremenski ograničen. S obzirom na broj ispitnih slučajeva koje je neophodno izvršiti prilikom ispitivanja uređaja, i imajući u vidu da taj broj sa dolaskom novih audio tehnologija konstantno raste, apsolutno je izvesno očekivati gubitak koncentracije QA inženjera. Primenom nove metodologije ovaj proces izvršavanja je u potpunosti automatizovan. I ako i dalje može da se pojavi potreba za ponavljanjem, broj ovakvih slučajeva sveden je na minimum. Razlog za ponavljanjem ispitnih slučajeva prilikom primene nove metodologije leži u nesavršenosti hardverskog sistema i nepredvidivim faktorima na koje QA inženjer ne može uticati. Čak i u tim situacijama, ponovno izvršavanje ispitnih slučajeva vrlo je jednostavno i obavlja se sa par klikova mišem.

Trajanje izvršavanja jednog ispitnog slučaja, kao što je prikazano u tabeli, smanjeno je za oko 40%, dok je ukupno vreme neophodno za ispitivanje uređaja smanjeno čak i do 50%. Ušteda vremena može se posmatrati i sa stanovišta potrošnje radnih dana, a u tom slučaju još jedan podatak ide u prilog novoj metodologiji. Zbog automatizovanog ispitivanja, gde gotovo da nema potrebe za intervencijom čoveka, ovaj proces može da se obavi i van radnog vremena što dodatno smanjuje ukupan broj potrošenih radnih sati. Ušteda vremena još je značajnija ukoliko se uzme u obzir podatak da je ukupan broj ispitnih slučajeva neophodnih za verifikaciju npr Dolby tehnologija porastao sa 4226 za Dolby Atmos HT 1.5 na 6539 za Dolby Atmos HT 1.6.2.

Pored uštede vremena kao resursa a samim tim i ljudskih resursa, novi hardverski sistem donosi i značajnu finansijsku uštedu. Ovo se postiglo zamenom APx uređaja sa RT-AG uređajem. Cena APx uređaja na tržištu iznosi čak i do 50000\$. Kako je cena ovog uređaja veoma visoka, na tim od na primer 10 QA inženjera dostupan je uglavnom jedan APx uređaj. Cena RT-AG uređaja iznosi oko 1000\$ pa samim tim svaki QA inženjer može imati svoj uređaj za ispitivanje.

6. Zaključak

Povećanje složenosti audio sistema sa dolaskom novih generacija audio tehnologija dovodi do povećanja složenosti ispitivanja i verifikacije ovakvih sistema. Kako metode manualnog ispitivanja postaju neefikasne u primeni pojavila se potreba za automatizacijom procesa ispitivanja.

Tema ovog rada je upravo automatizacija radnog okruženja za ispitivanje složenih audio sistema.

U ovom radu objašnjeno je na koji način je izmenjen i unapređen hardverski sistem stare metodologije ispitivanja, zamenom APx uređaja sa RT-AG uređajem. Sam proces ispitivanja je takođe unapređen. Za generisanje ispitnih slučajeva napisano je nekoliko namenskih Python skripti, a za izvršavanje istih korišćen je RT-Executor kojim se proces ispitivanja u potpunosti automatizuje. Jedan od zahtevnijih koraka prilikom izvršavanja svakog ispitnog slučaja predstavljalo je generisanje i slanje konfiguracionih poruka, koje je u dosadašnjoj primenjivanoj metodologiji bilo izvršavano ručno. Ovaj proces je takođe automatizovan pisanjem Python skripti. RT-Executor omogućio je i jednostavnu i automatizovanu analizu snimljenih izlaza, koja je u tipičnoj ispitnoj proceduri bila vršena ručno.

Ponudeno radno okruženje zajedno sa celokupnim sistemom za automatsko ispitivanje eliminiše sve nedostatke ručnog ispitivanja. Automatizacijom ovog procesa dobija se značajna ušteda na vremenu. Čak i do 50% manje vremena je potrebno kako bi se verifikovala ispravnost jednog uređaja. Takođe, kako se testovi automatski izvršavaju, QA inženjer ne mora konstantno nadgledati proces, što mu ostavlja prostor da obavlja još jedan posao istovremeno čime se dodatno dobija na vremenu. Sve ovo smanjuje vreme potrebno za pojavljivanje samog proizvoda na tržištu. Eliminisanjem faktora ljudske greške, pouzdanost ispitivanja je takođe podignuta na

viši nivo. Upotreba RT-AudioGrabera, čija cena je manja čak i do 50 puta u odnosu na prethodno korišćeni APx, donosi značajnu finansijsku uštedu.

7. Literatura

- [1] Dolby Laboratories Inc: Surround Past, Present and Future, 1999.
- [2] B. Claypool, W. Van Baelen, B. Van Daele, „Auro 11.1 versus object based sound in 3D“, 2015.
- [3] N. Paunović, J. Kovačević, I. Rešetar, „Jedno rešenje metodologije ispitivanja složenih sistema profesionalne elektronike“, Conference for Electronics, Telecommunications, Computers, Automatic Control and Nuclear Engineering (ETRAN), Zlatibor, Srbija, Jun, 2011
- [4] T. Fine, „The dawn of commercial digital recording“, ARSC, vol. 39, pp. 1-10, 2008.
- [5] N. Pekez, J. Kovačević, „Jedno rešenje softverske arhitekture uređaja za snimanje i reprodukciju audio signala“, 2017.
- [6] M. V. J. Maes, Digital Audio Technology: A guide to CD, MiniDisc, SACD, DVD(A), MP3 and DAT, Waltham: Focal Press, 2001.
- [7] HDMI Licensing, LLC, "HDMI Interface Extends Exceptional Digital Quality with SingleCable Simplicity to Over 4 Billion Consumer Devices," 6 January 2015. [Online]. Available: www.hdmi.org/press/press_release.aspx?prid=137. [Accessed 17 September 2019].
- [8] Philips Semiconductors, "I2S Specification," Eindhoven, 1986.
- [9] W. V. B. B. V. D. B. Claypool, "Auro 11.1 versus object-based sound in 3D," Kortrijk, 2011.
- [10] I. Sommerville, Software Engineering, 3rd ed., London: Pearson, 2010.
- [11] G. Myers, „Art of software testing“, ISBN: 0471043281, 1979
- [12] Kolawa, A. and D. Huizinga, „Automated Defect Prevention: Best Practices in Software Management“. Wiley-IEEE Computer Society Press, September 2007.

-
- [13] Audio Precision, Inc., "APx Overview," [Online]. Available: <https://www.ap.com/analyzersaccessories/apx-overview/>. [Accessed 25 September 2019].
- [14] Audio Precision, Inc., "Audio Analyzers Comparison," [Online]. Available: <https://www.ap.com/analyzers-accessories/compare/>. [Accessed 25 September 2019].
- [15] Echo Digital Audio Corporation , "Echo Digital Audio Pro Audio," [Online]. Available: <https://echoaudio.com/pages/pro-audio>. [Accessed 25 September 2019].
- [16] „CRD498x4 Boards User’s Manual“, Cirrus Logic, November, 2015.
- [17] „CS498xx_Datasheet“, Cirrus Logic, Inc., March, 2013.
- [18] N. Pekez, J. Kovačević, M. Beljulji, „Tehničko rešenje softverske arhitekture digitalnog audio grebera i plejera RT-AG“, Conference for Electronics, Telecommunications, Computers, Automatic Control and Nuclear Engineering (ETRAN), Zlatibor, Srbija, Jun, 2017.
- [19] Xilinx, Inc., "Zynq-7000 All Programmable SoC Data Sheet: Overview," 6 June 2017. [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds190-Zynq-7000-Overview.pdf. [Accessed 29 September 2019].
- [20] V. Peković, N. Teslić, I. Rešetar, T. Tekcan, „Test Management and Test Execution System for Automated Verification of Digital Television Systems“, 14th International Symposium on Consumer Electronics , Braunschweig, Germany, Jun, 2010.