

Digitalno krmiljenje analognega studijskega izenačevalnika

Nejc Kirn¹, Franc Policardi-Antoncich²

¹ DISTOPIK d.o.o., Tobačna ulica 5, SI-1000 Ljubljana, Slovenija

² LUCAMI, Fakulteta za elektrotehniko, Tržaška cesta 25, SI-1000 Ljubljana, Slovenija

nejc.kirn47@gmail.com - franc.policardi@fe.uni-lj.si

Audio industry has greatly developed in the last century and digital audio has been a standard for forty years already. In this article we demonstrate the process of creating a software program controlling a custom designed analog studio processor i.e. an equalizer. First we briefly overview why, despite this technology is decades old, it remains popular in recording studios and the problems that users face. Then we propose the digital control solution for the custom circuit boards. The finished system uses proprietary potentiometers and switches allowing users to both control the parameters by using the instrument's front panel, and to read, save and control the unit from a computer program. This article follows the ERK 2017 [1].

1 Uvod

Kljub relativni starosti tehnologije je analogna avdio oprema v zvočnih snemalnih studiih še vedno priljubljena. Glavni razlog za njeno uporabo je edinstven zvočni podpis, ki ga prinesejo fizične komponente. Čelne plošče s fizičnimi stikali, potencimetri, tipkami in ostalimi komponentami omogočajo direkten pristop k oblikovanju zvoka in so dodana vrednost pri uporabniški izkušnji. Klasična izvedba takih procesorjev ima tudi nekaj pomanjkljivosti, tako da uporaba digitalnega kontrolnega sistema za krmiljenje parametrov v signalni poti analognega avdio procesorja prinese veliko prednosti; možnost primerjave nastavitvev in shranjevanje teh pomenijo za studijskega inženirja lažje, hitrejše in optimizirano delo ter kakovostnejši končni izdelek.

V prispevku bomo predstavili proces izdelave programske opreme, digitalnega krmiljenja in elektronskega vezja novega visokokakovostnega zvočnega izenačevalnika Distopik Precision Series Mastering EQ. Ostale podrobnosti so poslovne skrivnosti podjetja Distopik d. o. o. [2], kateremu se vnaprej zahvaljujemo.

2 Analogni studijski procesorji

Analogni studijski procesorji so, zaradi stranskih učinkov v elektronskih vezjih in elementih, še danes priljubljeni v kontekstu snemanja in obdelave zvoka. Dober primer so filtri za oblikovanje specifičnih frekvenc, t.i. *equalizers* (izenačevalniki), dinamični procesorji, osnovani na fotocelicah ali vakuumskih ceveh ter magnetni trak.

Prvi problem uporabe analognih procesorjev je njihov uporabniški vmesnik z nastavitvami, kjer so stikala, tipke, gumbi in ostale komponente, ki delujejo kot spremenljiv del signalne poti vezja. S spreminjanjem pozicije v vezje vstavljamo elektronske komponente kot so npr. upori in kondenzatorji in s tem spreminjamo vpliv vezja na zvok z vsemi "zaželenimi" distorzijami originalnega signala. Shranjevanje nastavitvev na analognih napravah izvajamo ročno z uporabo slik oz. zapisovanja števil, na katere kažejo indikatorji čelne plošče. Tako shranjevanje nastavitvev je dokaj težavno pri zveznih potencimetrih, zato se za zahtevnejše aplikacije kot je npr. mastering uporabljajo koračni potencimetri z diskretnimi nastavitvami, ki zvišajo zahtevnost izdelave in ceno.

Druge problematiko predstavlja kalibracija sistema v celoti oz. zmanjšanje nezaželenih vplivov motečih signalov. Eden najpogostejših je električno napajanje, saj se iz tega velikokrat v sistem prenese šum okoli 50Hz in njegove harmonske frekvence do f_n ter s tem poslabša S/N. To je značilnost analognega obratovanja; na ta šum moramo biti pozorni, ker za določene frekvence lahko preseže 30dB. Šum pripomore k intermodulacijski distorziji v vsaki opremi v verigi in lahko večkrat zaokroži skozi verigo, tako da postane, kljub vsem poskusom, praktično neizogiben. Le tri možnosti zmanjševanja šuma imamo na voljo: zelo kakovosten vir napajanja, visoko kakovosten UPS sistem ali baterijsko napajanje. V primeru neupoštevanja tega dejavnika se bomo od začetka snemalne verige do končnih post-produkcijskih del vedno soočali s t.i. zmanjšanjem "jasnosti" izdelka. Tudi zaradi tega vzroka v digitalnih sistemih produkcije pogosto uporabljamo t.i. *plug-in (p-i)* oz. vtičnike, ki posnemajo rokovanje s čelno ploščo z nastavljivimi parametri, kateri vplivajo na delovanje določenih algoritmov. Za razliko od analogne opreme se nastavitve teh parametrov lahko shranijo v računalniškem spominu, na katerem se izvaja Digital Audio Workstation (DAW) in jih kasneje lahko dokaj enostavno priključimo kot prednastavitve. *P-i-i* omogočajo tudi avtomatizacijo parametrov, ki so v programu zapisani kot dodatna sled parametra, katero je kasneje mogoče spremeniti. Dodatno se s takojšnjo uporabo *p-i*-ov izognemo motečim električnim šumom, ki jih omejimo na najkrajšo verigo mikrofona-kabel-predajačevalnik-A/D pretvornik. Te tri lastnosti se izkažejo za glavne razloge, zaradi katerih dandanes vse več poklicnih avdio inženirjev svoje delo opravlja preko *p-i*, saj jim ti prihranijo veliko truda in časa.

3 Digitalno krmiljenje izenačevalnika

Omogočanje digitalnega krmiljenja izenačevalnika je pomenilo zamenjavo določenih elementov iz signalne poti in njihovo krmiljenje s pomočjo mikrokontrolerja. V izvirnih vezjih je signal potoval od glavne avdio plošče do potenciometrov na čelni plošči, tako da smo s spreminjanjem lege potenciometra v signalno pot spremenili vrednost upornosti in s tem sredinsko frekvenco, obliko in ojačenje. Čelno in avdio ploščo smo ločili in med njiju vstavili vezje z mikrokontrolerjem. Iz čelne plošče sedaj beremo nastavitve potenciometrov in te informacije uporabimo za preklap stikala na po meri izdelanem tiskanem vezju. Avdio signal tako potuje po enaki poti, le da imamo sedaj možnost digitalnega krmiljenja. Želimo tudi, da ima vezje z mikrokontrolerjem možnost omrežne povezave, kar bi nam omogočilo pošiljanje ukazov iz računalnika brez uporabe fizične prednje plošče. Idejna oblika vezja za digitalno krmiljenje algoritma za nastavljanje izenačevalnika poteka sledeče: a) pozicije potenciometrov oz. enkoderjev se berejo v ekspanderje, b) ekspanderji informacije spremenijo v format SPI in jih pošljejo mikrokontrolerju, c) mikrokontroler podatke obdela in jih pošlje na multiplekserje, d) multiplekserji izberejo ustrezen ekspander na glavni plošči, e) ekspander krmili releje in čipe ter tako nastavi filtre na želeno vrednost; kadar krmilimo izenačevalnik iz računalnika pa tako: a) ob spremembi parametrov na zaslonu *p-i* pošlje sporočilo preko mreže na prednastavljen IP-naslov in vrata v formatu OSC, b) mikrokontroler to sporočilo prevede v ukaz, ki ga multiplekserji in ekspanderji razumejo, c) naprej je postopek enak, kot od zgornje 4. točke.

Za obliko filtra smo uporabili koračne potenciometre Lorlin z 12-imi koraki, medtem ko smo za frekvence uporabili 24-koračne, od katerih smo porabili le 22 korakov. Pini so vezani na ekspanderje, ki trenutne nastavitve ob ustreznem proženju sestavijo v dva bajta podatkov in jih pošljejo mikrokontrolerju.

Za nastavljanje ojačenja smo uporabili pretvornik LTC1596-1 D/A oz. MDAC; ta nam omogoča, da s 16-bitno natančnostjo nastavljamo izhod med 0V in referenčno napetostjo, v tem primeru našim avdio signalom. Ker je enkoder le 8-biten, se s čelne plošče ne da izkoristiti polne ločljivosti; to nam dopušča *p-i* oz. krmiljenje iz računalnika.

Mikrokontroler se z računalniškim *p-i*-om sporazumeva preko kabla ethernet in protokola OSC, tako se vsaka sprememba na čelni plošči ustrezno odraža na uporabniškem vmesniku *p-i*-a. Obratno to ne deluje, saj bi v tem primeru morali biti potenciometri in stikala motorizirani. Spremembe se v *p-i*-u vseeno odražajo v dejanskih nastavitvah na kontrolni plošči oz. v relejih.

Elektronska vezja

Naštetim modelom pripadajo posamezna fizična tiskana vezja. Kakor je prikazano na sliki 1, so v ohišju izenačevalnika štiri vezja:

- analogni del na dnu ohišja
- vezje z mikrokontrolerjem
- glavna plošča, ki je povezana z analognim delom
- čelna plošča



Slika 1: Od vrha slike navzdol: čelna plošča, glavna plošča in vezje z mikrokontrolerjem

Pomembnejše komponente na vezju z mikrokontrolerjem so sam mikrokontroler, stikalni napajalnik in priključek za ethernet. Preko tega vezja se izvaja komunikacija z protokoloma OSC za računalnik in SPI za vse ostale komponente, kot so ekspanderji in multiplekserji ter preko njih tudi z releji (bele kvadraste komponente na sliki 1), stikali, statusnimi LED diodami itn..

Na glavni plošči je večji del kontrolnega vezja: štiri multiplekserji in za vsak filter po dva 16-bitna in en 8-bitni ekspander, en MDAC ter do 36 relejev. Z njimi nastavljamo sredinsko frekvenco in kakovost zvonastih filtrov, jih vklapljamo in menjamo med pozitivnim in negativnim ojačenjem na MDAC. Ta plošča je na vezje z mikrokontrolerjem povezana z oklopljenim kablom. Čelna plošča je sestavljena iz treh tiskanih vezij, ki so med sabo povezane z tračnim kablom. Na vsaki od treh najdemo dva 16-bitna ekspanderja in ustrezne povezave za komponente, kot so stikala, LED diode, potenciometre itn., s katerimi uporabnik nastavlja izenačevalnik.

4 Izdelava programske opreme

Orodja

Pri programiranju krmilnega programa izenačevalnika smo uporabljali naslednja orodja: 1) programator ST-LINK/V2 za mikrokontrolerje STM8 in STM32; 2) Digilent Analog Discovery za merjenje, analizo, vizualizacijo in snemanj dogajanja na vezjih; skupaj s 3) programsko opremo Digilent WaveForms nam omogoča preverjanje delovanja SPI komunikacije, ker vsebuje dvokanalni osciloskop, dvokanalni funkcijski generator, 16 digitalnih vhodov in izhodov ter dva izhoda za napajanje.

Programska oprema

Za programiranje mikrokontrolerja STM32 v jeziku C++ smo uporabili okolje Eclipse.

Za razvoj *p-i*-a, ki skrbi za komunikacijo med računalnikom in mikroprocesorjem v izenačevalniku smo uporabili Microsoft Visual Studio 2013. *P-i* je programiran v jeziku C++ in uporablja knjižnice JUCE za razvoj namiznih in mobilnih aplikacij v namen avdia. Microsoft Visual Studio 2013 vsebuje metode, razrede itn., ki so optimizirani za izvajanje v realnem času in take, ki olajšajo izdelavo uporabniškega vmesnika. Glavni namen tega skupka knjižnic je, da olajša razvoj avdio aplikacij, saj ista izvorna koda lahko teče na različnih namiznih in mobilnih operacijskih sistemih.

Za merjenje delovanja izenačevalnika smo uporabili program Reaper DAW, ki nudi snemanje več kanalov zvoka in/ali MIDI naenkrat ter program Fuzzmeasure, s katerim smo kalibrirali izenačevalnik: ta nudi natančne

ši prikaz rezultatov, ki je zaradi uporabe sinusnega preleta namesto belega šuma vseeno počasnejši in zato za zagotavljanje kakovosti pri izenačevalniku s tako velikim številom parametrov manj uporaben.

Zvočna kartica Audient iD4 in programska oprema iD Audient sta nam omogočala predvajanje in zajemanje avdio signala med računalnikom in izenačevalnikom.

Modeli

Ključnega pomena za uspešen razvoj kode so t.i. "modeli" izenačevalnika, čelne plošče in *p-i-a*, pri osnovanju katerih smo se poslužili tehnike »*domain driven design*«-u, s katero smo morali, za uspešno reševanje neke problematike z računalniškim programom, poznati njeno domeno in poznati vrhno arhitekturo, delovanje in dinamiko znotraj naše domene. Dobro načrtovan sistem je zmanjšal število konceptualnih razlik med delovanjem domene v praksi in delovanjem abstraktnega programa, ki to domeno opisuje ter hkrati zmanjšal število kasnejših zahtev za spremembe in popravila. Glavni pomembni modeli so trije: a) model izenačevalnika, ki predstavlja stanje na glavni avdio plošči, b) model čelne plošče, ki predstavlja stanje nastavitvev na čelni plošči in c) model *p-i-a*, ki predstavlja nastavitve na računalniškem uporabniškem vmesniku. Razdelili smo jih zato, ker moramo omogočiti smiselno delovanje pri kombiniranem upravljanju s čelno ploščo oz. *p-i-om*. Zahteve so bile: a) sprememba na čelni plošči se mora poznati tako na zvoku kot tudi na *p-i-u*, b) sprememba na *p-i-u* se mora poznati na zvoku, pri čemer se trenutno stanje na čelni plošči ignorira in c) če uporabnik odpre projekt, v katerem teče *p-i*, kjer so zapisane nastavitve parametrov, se le-te ne smejo prepisati s tistimi, ki so trenutno na čelni plošči razen, če prepis nastavitvev uporabnik sam sproži preko za to namenjene tipke.

Z uporabo treh modelov lahko vsem trem zahtevam zadostimo dokaj enostavno, ker v vsakem posebej hranimo nastavitve (glavne avdio plošče oz. čelne plošče oz. *p-i-a*): dejansko stanje na relejih in MDAC-ih oz. nastavitve, ki jih uporabnik sliši, so vedno shranjene v modelu izenačevalnika. Da modelu glavne plošče lahko posredujemo ustrezno informacijo, moramo v modelih čelne plošče in *p-i-a* hraniti ločene spremenljivke za vse nastavitve. Metoda, ki nam je omogočila hiter prepis parametrov v *p-i-u* z nastavitvami fizične čelne plošče je `invalidFrontPanel`. Kadar spreminjamo nastavitve v *p-i-u* izgubimo sinhronizacijo med čelno ploščo in *p-i-om*. Zaradi velikega števila parametrov bi bilo ročno nastavljanje za uporabnika zamudno in zato smo dodali tipko `Load From Panel`, ki naenkrat prekopira vse vrednosti iz modela čelne plošče v modelu izenačevalnika in *p-i-a*. S tem smo s pritiskom ene tipke omogočili ponastavitev približno 50 parametrov.

Osnovna logika programa

Določene dele kode, ki so se uporabljali pri vseh modelih, je bilo smiselno zapisati v svojo knjižnico. V njej smo med drugim definirali metode in razrede, ki so namenjeni neposrednemu pisanju in branju pinov ekspanderjev na čelni oz. glavni plošči. Prvi pomemben del so metode, ki preberejo izbran pin oz. bit iz ekspanderja v primeru čelne plošče oz. zapišejo

ustrezno vrednost v ekspander, ki krmili releje v primeru glavne plošče. Metodo postavi *n-ti* bit na 1 in ostale na 0, sestavili smo jo iz osnovnejših, ki so na primer postavile *n-ti* bit na 0 oz. 1.

Glavna plošča

Na glavni plošči se nahajajo multiplekserji, ekspanderji, releji in MDAC-i, ki vplivajo na nastavitve frekvenc, oblike itn. filtrov izenačevalnika.

Ko po SPI pošljemo vrednost enega parametra filtra, se ta vrednost pošlje na vse filtre, kar ni zaželeno. Zato smo morali poskrbeti, da le določeni ekspanderji dobijo ta podatek in to smo izvedli z metodo `selectBlock`, kateri na podlagi podanega parametra »i« ugotovi kombinacijo vhodov ABCD na multiplekserjih, ki ustreza na *i*-temu filtru; nato ekspanderjem na tem filtru pošlje signal `Chip Select`.

`InitEQ` in `initBlock` sta metodi, ki ob zagonu programa oz. vklopu izenačevalnika inicializirata ekspanderje na glavni plošči. Ob klicu `initEQ` se sproži for zanka, ki eno za drugim naredi inicializacijo na vseh filtrihi.

Čelna plošča

Čelna plošča je sestavljena iz treh posameznih tiskanih vezij, ki so med sabo povezane z tračnim kablom. Na plošče so pritrjeni koračni potenciometri, enkoderji, stikala in LED diode, ki služijo kot fizični uporabniški vmesnik. Iz vseh teh komponent, nameščeni na vseh treh ploščah, ekspanderji berejo nastavitve in jih posredujejo mikrokontrolerju. Vsak ekspander bere en tip parametra z izjemo frekvence, kjer je zaradi 22 možnih nastavitvev treba uporabiti dva 16-bitna ekspanderja.

V razredu `FrontPanelExpanders` najdemo metodo `initExpanders`, ki sama kliče metodo `initInput` oz. `initOutput` iz prejšnje knjižnice `CommonIO.hpp`. Z njeno pomočjo se ekspanderji pravilno inicializirajo in pripravijo na delovanje. Naslednja metoda je `scanFrontPanel`, kateri podamo številko od 0-9. Glede na podano številko se najprej izbere ustrezen filter in nato preberejo nastavitve vseh potenciometrov in stikal na čelni plošči. Za branje nastavitvev smo uporabili metode `write` in `read`, ki so definirane v `CommonIO.hpp`. S tema dvema metodama zagotovimo povezavo med knjižnico `FrontPanelIO.hpp`, ki se ukvarja z domeno na čelni plošči in `CommonIO.hpp`, ki se ukvarja z delovanjem samih čipov na nižjem nivoju. Koda postane s tem bolj berljiva.

`FPLowFilter`, `FPMidFilter` in `FPHighFilter` se ukvarjajo z nastavitvijo oblik posebej nizkih, srednjih in visokih filtrov, saj imajo različno funkcionalnost. Nizki filter mora pri določenih nastavitvi iz vezja izklopiti potenciometer in vklopiti ločen del vezja, ker s tem omogoči preklon iz filtra oblike zvonca v obliko police. Za nastavljanje ojačenja smo uporabili enkoder Bourns ACE 128. Na osmih podatkovnih pinih generira število, ki ga moramo preko tabele pretvoriti tako, da pri vrtenju v smeri urinega kazalca dobimo števila od 0 do 127. Programsko vrednosti pretvorimo v število s plavajočo vejico z razponom od 0 do 1, ki ga uporabimo za shranjevanje vrednosti ojačenja, saj ga prek 16-bitnega MDAC lahko nastavljammo skoraj zvezno.

P-i in komunikacija

Za komunikacijo med računalnikom in izenačevalnikom uporabimo protokol OSC, razvit za upravljanje in

komunikacijo med računalniki, sintetizatorji zvoka, aplikacijami itd. preko mreže za namen glasbenega nastopanja. Protokol OSC ima podoben namen kot protokol MIDI, vendar omogoča bolj podrobno, razširjeno in brezžično komunikacijo.

Za namen komunikacije *p-i-a* in izenačevalnika smo opisali metode modela izenačevalnika in *p-i-a*, ki sestavljajo in pošiljajo OSC sporočila na ustrezna vrata na mreži. OSC sporočilo je sestavljeno iz naslova, tipa argumenta in argumenta [3]. Naslov nam pove, kateri parameter izenačevalnika spreminjamo, tip argumenta pri nas loči med 32-bitnim integer-jem in 32-bitnim float-om, argument pa je nastavek parametra. Pri OSC moramo biti pozorni na število znakov; na konec naslova, tipa in argumenta moramo dodati toliko ničel, da je število znakov v OSC sporočilu deljivo s 4 in se konča z 0. Napisali smo avtomat stanj tako, da je izenačevalnik uporaben tudi brez priklopa na mrežo; stanja so Disconnected, Initialized, NoEthernet, ConnectedPending in Connected. Del komunikacije smo spisali tudi v kodi, ki opisuje delovanje *p-i-a*. Ta nam ponuja grafični uporabniški vmesnik (Slika 4), s katerim lahko nastavitve izenačevalnika spreminjamo iz DAW.



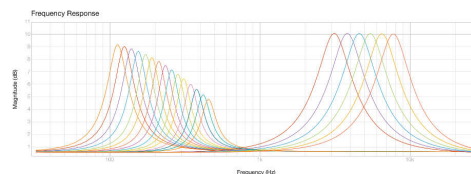
Slika 4: Uporabniški vmesnik plug-in-a

Skupina knjižnic JUCE, po definiranju imen parametrov in njihove povezave na ustrezne elemente v grafičnem vmesniku, sama poskrbi za delovanje. Dodali smo še metode za pošiljanje in sprejemanje OSC sporočil ter tipke Load From Panel, Preset A oz. B in Load Preset. Prva vzame nastavitve iz čelne plošče in povesi trenutno nastavljene v *p-i-u*, zadnja deluje obratno. Tipki A in B omogočata priklic dveh prednastavitev za hitro in enostavno primerjavo. *P-i* ima tudi gumb za ojačenje signala za +24dB.

5 Preizkus delovanja

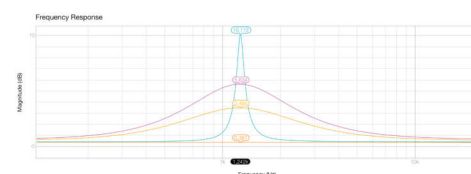
Verifikacija krmiljenja parametrov je potekala z uporabo več orodij. Prvi je bil logični analizator Analog Discovery, s katerim smo že ob programiranju knjižnic MainboardIO.hpp, MEQmain.hpp, FrontPanelIO.hpp in CommonIO.hpp sproti preverjali delovanje SPI komunikacije. Z uporabo programa Reaper in Fuzzmeasure smo preverili, ali se z izbiranjem številke napisanih na čelni plošči filtri res nastavijo na ustrezno sredinsko frekvenco. Iz prvega smo preko zvočne kartice v izenačevalnik pošiljali beli šum in ga nato analizirali s *p-i* FFT analizatorjem; preverili smo kombinacije vseh nastavitev frekvenc in kakovost filtrov tako pri pozitivnem kot pri negativnem ojačenju. Naleteli smo na nekaj nepravilnosti v kodi kot tudi v strojni opremi. Primera takih napak sta bila metoda za izbiro frekvence, ki je le-te izbirala v obratnem vrstnem redu in ni pravilno upoštevala vezav na čelni in glavni plošči, zaradi česar sta bila ob izboru frekvence 1 in 22

dejansko izbrani 2 in 1. Naleteli smo tudi na pregoreli upor na enem izmed kanalov, ki je onemogočal pravilno delovanje uporovne lestve za izbiro frekvenc (Slika 5).



Slika 5: Pokvarjen upor povzroči preskok frekvenc

Izkazalo se je tudi, da ob izbiri +3dB ojačenja na čelni plošči le-tega nismo dobili na zvoku in da niti krivulja ojačenja ni bila pravilna. Za te meritve smo uporabili program Fuzzmeasure, ki namesto belega šuma uporablja sinusni prelet. S tem smo si zagotovili bolj natančne meritve za iskanje aproksimacijske funkcije. To funkcijo smo ustrezno vstavili v kodo in s tem poskrbeli, da vse številke nastavitve ojačenja na čelni plošči ustrezajo ojačenju, ki vpliva na avdio (Slika 6).



Slika 6: Meritev ojačenja

6 Sklepne ugotovitve

V prispevku smo predstavili izdelavo programske opreme, digitalno krmiljenje in elektronsko vezje novega zvočnega izenačevalnika. Tekom našega dela smo razdelili in postopoma rešili analogne izzive povezane z električnim napajanjem, potenciometri in stikali, vezjem z mikrokontrolerjem, programiranjem na različnih nivojih, komunikacijo med komponentami kot npr. glavno in čelno ploščo in z digitalnim shranjevanjem nastavitev in računalniškim krmiljenjem parametrov. Naš končni izenačevalnik omogoča klasično ročno nastavljanje parametrov preko potenciometrov in stikal do čelne plošče in istočasno branje, shranjevanje ter spreminjanje vseh parametrov preko računalniškega programa. Naprava sama poskrbi, da so parametri na pravih vrednostih; takoj, ko uporabnik odpre projekt v DAW lahko nadaljuje z delom in s tem prihrani čas.

Menimo, da bo sčasoma na trgu prišlo do večje razširjenosti digitalnega krmiljenja procesorjev, kar bo znižalo cene in povečalo dostopnost.

Literatura

- [1] N. Kirn, F. Policardi Antoncich, *Optimizacija uporabniške izkušnje in krmilnih sistemov v modernih studijskih avdio procesorjih*, ERK 2017;
- [2] mix:analog [Online]. Dosegljivo mixanalog.com. [Dostopano: 9.1.2018];
- [3] The Open Sound Control Specification [Online]. Dosegljivo: opensoundcontrol.org/spec-1_0. [Dostopano: 9.1.2018].