



**UNIVERZITET U NOVOM SADU
FAKULTET TEHNIČKIH NAUKA
NOVI SAD**

Računarstvo i automatika

Računarska tehnika i računarske komunikacije



DIPLOMSKI - MASTER RAD

**Tema rada: Jedno rešenje programske podrške
za snimanje i reprodukciju zaštićenog digitalnog
TV toka podataka**

- Predmet: Programska podrška u televiziji i obradi slike -

**Mentor:
Prof. Dr. Nikola Teslić**

**Kandidat:
Tereza Kovač, E9976**

Novi Sad, Novembar 2006.



UNIVERZITET U NOVOM SADU • **FAKULTET TEHNIČKIH NAUKA**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KLJUČNA DOKUMENTACIJSKA INFORMACIJA

Redni broj, RBR:			
Identifikacioni broj, IBR:			
Tip dokumentacije, TD:	Monografska publikacija		
Tip zapisa, TZ:	Tekstualni štampani materijal		
Vrsta rada, VR:	Diplomski - Master rad		
Autor, AU:	Tereza Kovač		
Mentor, MN:	Prof.Dr. Nikola Teslić		
Naslov rada, NR:	Jedno rešenje programske podrške za snimanje i reprodukciju zaštićenog digitalnog TV toka podataka		
Jezik publikacije, JP:	Srpski		
Jezik izvoda, Ji:	Srpski		
Zemlja publikovanja, ZP:	Srbija		
Uže geografsko područje, UGP:	Srbija, Vojvodina		
Godina, GO:	2006		
Izdavač, IZ:	Autorski reprint ili Fakultet Tehničkih Nauka		
Mesto i adresa, MA:	Novi Sad, Trg Dositeja Obradovića 6		
Fizički opis rada, FO: (poglavlja/strana/citata/tabela/slika/grafika/priloga)	9/46/0/7/32/0/2		
Naučna oblast, NO:	Elektrotehnika i računarska tehnika		
Naučna disciplina, ND:			
Predmetna odrednica/Ključne reči, PO:			
UDK			
Čuva se, ČU:	U biblioteci FTN, Novi Sad, Trg Dositeja Obradovića 6		
Važna napomena, VN:			
Izvod, IZ:			
Datum prihvatanja teme, DP:			
Datum odbrane, DO:			
Članovi komisije, KO:	Predsednik:		
	Član:		Potpis mentora
	Član, mentor:		



UNIVERSITY OF NOVI SAD • **FACULTY TECHNICAL SCIENCES**
21000 NOVI SAD, Trg Dositeja Obradovića 6

KEY WORDS DOCUMENTATION

Accession number, ANO :	
Identification number, INO :	
Document type, DT :	Monographic's publication
Type of record, TR :	Word printed record
Contents code, CC :	Master work
Author, AU :	Tereza Kovač
Mentor, MN :	Ph. D.E.E. Nikola Teslić
Title, TI :	
Language of text, LT :	Serbian
Language of abstract, LA :	Serbian
Country of publication, CP :	Serbia
Locality of publication, LP :	Serbia, Vojvodina
Publication year, PY :	2005
Publisher, PB :	Author's reprint ili FTN
Publication place, PP :	
Physical description, PD : (chapters/pages/ref./tables/pictures/graphs/appendixes)	9/46/0/7/32/0/2
Scientific field, SF :	Electrical engineering and computer science
Scientific discipline, SD :	
Subject/Key words, S/KW :	
UC	
Holding data, HD :	Library of FTN, Novi Sad, Trg Dositeja Obradovića 6
Note, N :	
Abstract, AB :	
Accepted by the Scientific Board on, ASB :	
Defended on, DE :	

Defended Board, DB :	President:		
	Member:		Mentor's sign
	Member, Mentor:		

SKRAĆENICE

ADP	- Audio DSP dekodler (<i>eng. Audio DSP Decoder</i>)
ALU	- Aritmetičko-logička jedinica (<i>eng. Arithmetic Logic Unit</i>)
ATSC	- Standard koji se koristi u Americi za DTV (<i>eng. Advanced Television Systems Committee</i>)
CA	- Uslovni pristup (<i>eng. Conditional Access</i>)
CAT	- Tabela za uslovni pristup (<i>eng. Conditional Access Table</i>)
CI	- (<i>eng. Common Interface</i>)
CPU	- Centralni procesor (<i>eng. Central Processor Unit</i>)
DCT	- Diskretna kosinusna transformacija (<i>eng. Discrete Cosine Transform</i>)
DMA	- (<i>eng. Direct Memory Access</i>)
DnD	- Modul za dekodovanje i prikazivanje (<i>eng. Decode and Display</i>)
DTP	- Demultiplekser transportnog toka podataka
DTS	- (<i>eng. Decode Time Stamp</i>)
DTV	- Digitalna televizija (<i>eng. Digital TV</i>)
DVB	- Emitovanje digitalnog videa (<i>eng. Digital Video Broadcast</i>)
ECM	- (<i>eng. Entitlement Control Message</i>)
EMM	- (<i>eng. Entitlement Management Message</i>)
EPG	- Elektronski programski vodič (<i>eng. Electronic Program Guide</i>)
ES	- Elementarni tok podataka (<i>eng. Elementary Stream</i>)
FS	- (<i>eng. File System</i>)
FTA	- (<i>eng. Free To Air</i>)
GOP	- Grupa slika (<i>eng. Group Of Pictures</i>)
HDD	- (<i>eng. Hard Disc Drive</i>)
HDTV	- DTV visoke rezolucije (<i>eng. High Definition TV</i>)
IDCT	- Inverzna diskretna kosinusna transformacija (<i>eng. Inverse Discrete Cosine Transform</i>)
iDTV	- Integrirani digitalni televizor (<i>eng. integrated DTV</i>)
JDS	- Jedinica za dekodovanje slike
JOS	- Jedinica za obradu slike
MPTS	- Transportni tok podataka sa više programa (<i>eng. Multi Program Transport Stream</i>)
OSD	- (<i>eng. On-Screen Display</i>)
PCR	- Vremenska baza (<i>eng. Program Clock Reference</i>)
PES	- Paketizirani elementarni tok podataka (<i>eng. Packetized Elementary Stream</i>)
PID	- Identifikator paketa (<i>eng. Packet ID</i>)
PS	- Programski tok podataka (<i>eng. Program Stream</i>)
PSI	- Dodatne informacije o programu (<i>eng. Program Specific Information</i>)
PTS	- Vremenski pečat za dekodovanje (<i>eng. Presentation Time Stamp</i>)
PVR	- (<i>eng. Personal Video Recorder</i>)
SI	- (<i>eng.</i>)
SPTS	- Transportni tok podataka sa jednim programom (<i>eng. Single Program Transport Stream</i>)
STB	- Specijalni prijemnik (<i>eng. Set Top Box</i>)
TS	- (Transportni) tok podataka (<i>eng. Transport Stream</i>)

SADRŽAJ

1.	Zadatak	3
2.	Teorijske osnove rada	4
2.1	Uvod u digitalnu televiziju	4
2.2	Vrste organizacije podataka u okviru MPEG-2 standarda	5
2.3	MPEG-2 - TS format podataka	5
2.4	TS sa jednim i više programa	8
2.5	Osnove MPEG-2 kompresije video podataka	8
2.5.1	Prostorna redundansa - kodovanje piksela korišćenjem Diskretne Kosinusne Transformacije (DCT)	9
2.5.2	Prostorna redundansa - kvantizacija i entropijsko kodovanje	10
2.5.3	Vremenska redundansa	11
2.5.4	Redosled slika za slanje i dekodovanje	12
3.	Opis namenske platforme	13
3.1	Tok obrade MPEG ulaznih podataka	14
3.2	Skladištenje MPEG ulaznih podataka	15
4.	Analiza problema	16
4.1	Reprodukcija sačuvanog TS	16
4.2	Načini čuvanja zaštićenog sadržaja na HDD (DRM)	16
4.3	Kriptovanje i uslovni pristup podacima	16
4.4	Rukovanje ECM porukama	17
4.4.1	Indeksiranje kompletnog ECM toka	17
4.4.2	Indeksiranje promena unutar ECM toka	18
5.	Opis realizacije sistema	19
5.1	Opis arhitekture PVR	19
5.2	Segmentacija toka podataka pri snimanju	20
5.3	Objekat Task	20
5.4	Objekat List	21
5.5	Objekat tsStorage	21
5.6	Rukovanje snimljenim TS podacima - Objekat File	22
5.7	Indeksiranje segmenata TS - Objekat Indexer	22
5.8	Pristup segmentima snimljenog TS - Objekat Media	22
5.9	Programski modul Recorder	23
5.10	Programski modul Player	24
6.	Opis realizacije podsistema za uslovni pristup	27
6.1	Objekat ScramblingCtrl	27
6.2	Enkripcija	27
6.3	Segmentacija i indeksiranje	30
6.3.1	Indeksiranje ECM toka	30
6.3.2	Indeksiranje ostalih podataka	30
6.3.3	Indeksiranje	30
6.4	Dekripcija	32
7.	Testiranje i rezultati	33
7.1	Normalna reprodukcija	33
7.2	2, 4 i 12 puta ubrzana reprodukcija u napred	33
7.3	60 i 120 puta ubrzana reprodukcija u napred	34
7.4	Ubrzana reprodukcija u nazad	34
7.5	Rezultati merenja performansi sistema	35
8.	Zaključak	36
8.1	Pravci daljeg razvoja	36
8.1.1	Optimizacija	36

8.1.2	Prenosivost	37
9.	Literatura	38

1. Zadatak

Zadatak diplomskog (master) rada je realizacija programske podrške za snimanje i reprodukciju zaštićenog DVB sadržaja. Potrebno je obezbediti podršku za platforme bazirane na Micronas MDE® (u daljem tekstu MDE) integrisanom kolu.

U okviru DVB standarda se koriste dve tehnike raspakivanja zaštićenih tokova podataka (deskremblovanja):

- U okviru programske podrške TV dekodera, pri čemu se kodni ključevi preko ugrađenog CA sistema, izdvajaju iz transportnog toka
- U okviru spoljne kartice, kada se zaštićeni podaci usmeravaju na CI gde se raspakuju i potom otvoreni podaci sa CI se vraćaju kao otvoreni TS ka TV dekoderu.

Realizovani programski paket treba da ponudi sistemsko rešenje koje će omogućiti snimanje i reprodukciju zaštićenog sadržaja korištenjem bilo koje od navedenih metoda dekripcije.

Osnovni zahtevi realizacije su:

- omogućiti kontinualnu reprodukciju zaštićenog sadržaja
- omogućiti tzv. “trik modove” rada, odnosno ubrzanu reprodukciju
- podržati slobodnu promenu pozicije sa koje se vrši reprodukcija (omogućiti vremenske skokove)
- reprodukcija sadržaja u obrnutom redosledu (unazad)

Testiranje programske podrške izvršiti na ciljnoj platformi korištenjem dostupnih satelitskih servisa. Pri tome je potrebno podatke dodatno obraditi, na način sličan zaštiti servisa korištenjem poznatih DVB sistema zaštite (Conditional Access).

2. Teorijske osnove rada

2.1 Uvod u digitalnu televiziju

Digitalna televizija (DTV) obuhvata emitovanje, prenos i prijem digitalnog audio i/ili video sadržaja. Prva demonstracija digitalnog TV prenosa održana je 1995. godine. DTV koristi digitalnu modulaciju i digitalnu kompresiju podataka koje je moguće izdvojiti iz toka podataka pomoću televizora koji su namenjeni prijemu digitalnog signala (integrisani digitalni televizori - eng. *integrated Digital TV* - iDTV) ili specijalnih prijemnika (eng. *Set Top Box*).

Osnovne prednosti DTV u odnosu na analognu:

- digitalni prenos je binarno kodiran pa se sadržaj može komprimovati slično kao i kod računarskih datoteka
- kod digitalnog prenosa, za razliku od analognog, ne prenose se cele slike, već samo oni delovi koji su se u međuvremenu promenili (ne prenosi se statična pozadina)
- ovakvim prenosom je omogućeno da distributeri programa emituju približno deset digitalnih kanala umesto jednog analognog, koristeći pri tom isti frekventni opseg
- kako su i računari i savremene telekomunikacione mreže zasnovani na digitalnoj tehnologiji, digitalizacijom televizije njihova integracija postaje mnogo jednostavnija, a rezultat te integracije su nove aplikacije kao što je interaktivna TV
- digitalni prenos obezbeđuje bolji kvalitet slike/zvuka, koji više ne mogu biti ometani u toku prenosa, rezultat toga je isti kvalitet slike i zvuka na prijemu kao i na izvoru emitovanja
- promenljivi format slike (4:3 ili 16:9)
- mono, stereo ili “surround” zvuk
- novi servisi kao što su izbor jezika za titl i zvuk, interaktivni i multimedijalni sadržaji, pristup internetu.

U većem delu sveta za DTV prenos se koristi grupa standarda koje propisuje međunarodna organizacija DVB (eng. *Digital Video Broadcast*). DVB standard propisuje upotrebu MPEG-2 algoritma za video i audio kompresiju. Ovaj standard ne nameće nikakva ograničenja po pitanju sadržaja koji se prenosi. Tako, preko jednog satelitskog transpondera propusnog opsega 38Mbit/s moguće je emitovati razne vrste usluga, na primer:

- 4-8 standardnih TV servisa,
- 2 HDTV servisa,
- 150 radio programa,
- 550 ISDN kanala podataka na 64Kbit/s.

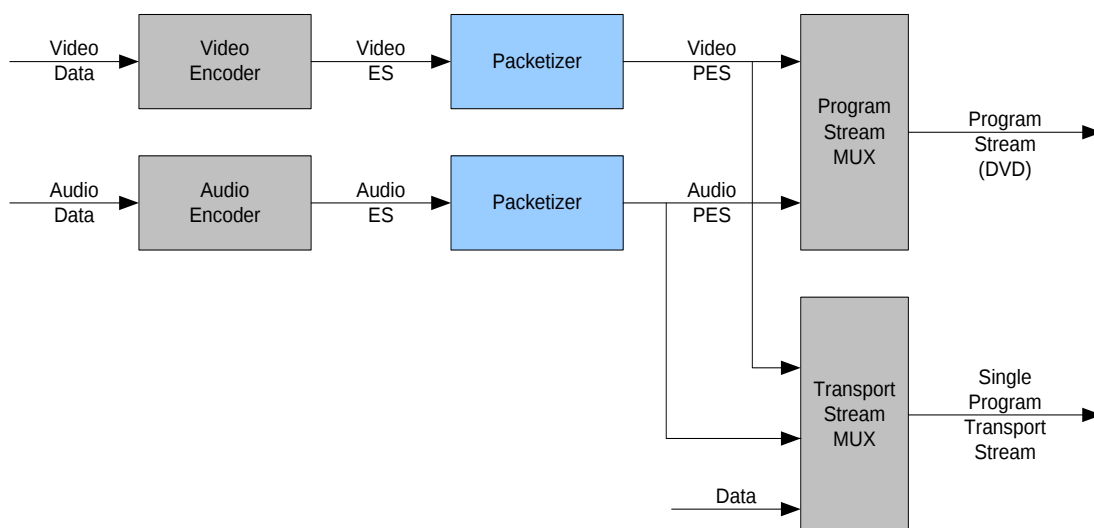
Paralelno sa DVB standardom u Americi je razvijen i trenutno se koristi ATSC standard (eng. *Advanced Television Systems Committee*). Ovi standardi se delimično razlikuju, ali kao i DVB i ATSC koristi MPEG-2 algoritam za kompresiju slike i zvuka.

Prenos digitalnog signala se može obaviti putem raznih komunikacionih medija, kao što su radio veza, digitalna satelitska veza, kablovska TV mreža, zemaljska komunikaciona veza, mikrotalasne veze (*wireless, mobile*), DSL (ADSL). DVB standard propisuje nekoliko načina distribucije signala:

- DVB-S – *DVB Satellite* - definiše prenos DTV putem satelita
- DVB-C – *DVB Cable* - definiše prenos DTV putem kablovske mreže
- DVB-T – *DVB Terrestrial*- definiše zemaljski prenos putem VHF/UHF radio talasa
- DVB-H – *DVB Handheld* - definiše DTV prenos za prenosne uređaje (mobilni telefoni)

2.2 Vrste organizacije podataka u okviru MPEG-2 standarda

Tokovi MPEG-2 video i audio podataka koji se emituju nazivaju se elementarni tokovi – ES (eng. *Elementary Stream*) i prenose se kao zasebni tokovi podataka. Kodovani podaci se prenose korišćenjem paketski orijentisanih protokola sa odgovarajućim mehanizmima za detekciju i otklanjanje grešaka. Da bi se odgovarajući ES na prijemnoj strani mogli povezati u logičke celine kao i da bi se obezbedila sinhronizacija audio i video podataka, pored ES moraju se prenositi i dodatni podaci, naziv programa, vreme emitovanja i slično.



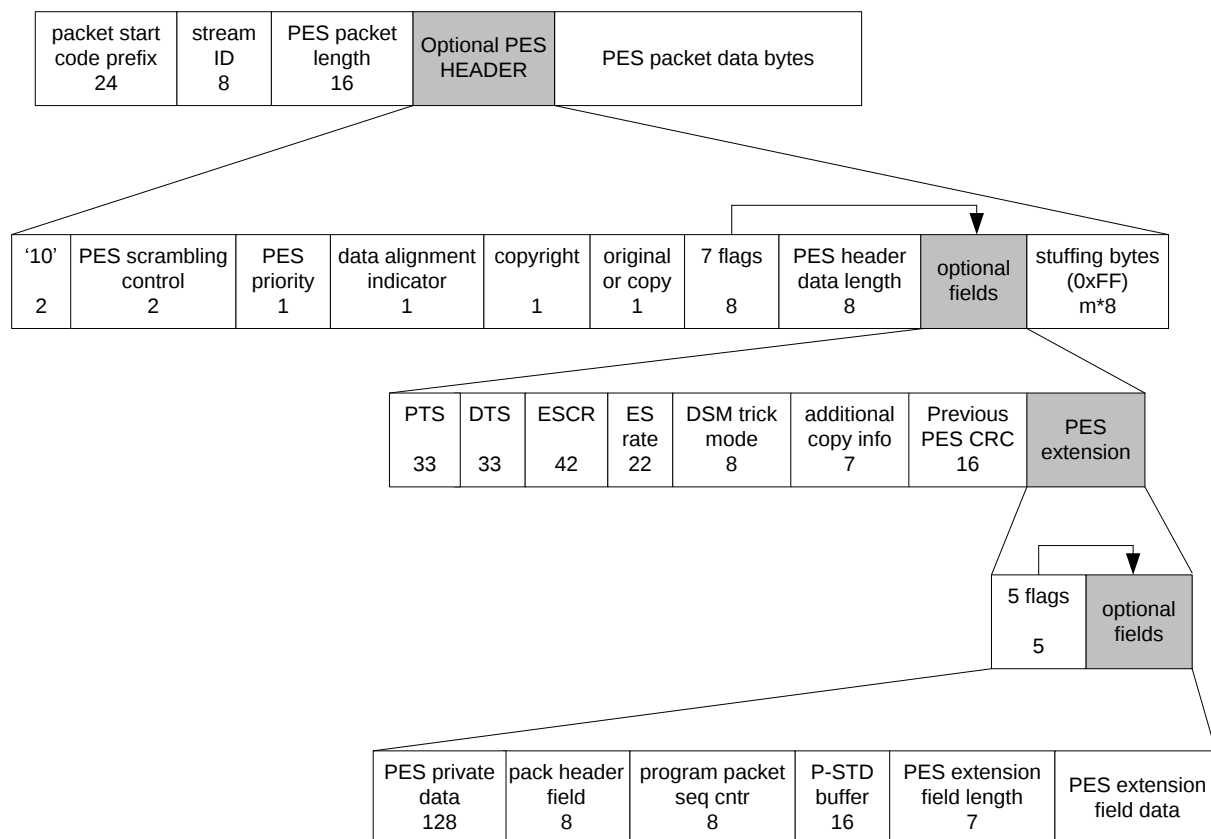
Slika 1 Vrste organizacija tokova podataka

U zavisnosti od prenosnog medija MPEG-2 standard definiše različite formate kombinovanja ES u jedinstven informacioni tok podataka. Ukoliko prenosni medij nije podložan pojavi grešaka u toku prenosa, ES se kombinuju u takozvani PS (eng. *Program Stream*). Ovakav format omogućava jednostavnije programsko rukovanje podacima i koristi se za audio/video reprodukciju (npr. CD ili DVD). Ukoliko je prenosni put podložan pojavi grešaka (sve vrste emitovanja programa), ES se kombinuju i prenose kao TS (eng. *Transport Stream*). TS format je takodje pogodan za kombinovanje više TV programa u jedan jedinstven informacioni tok podataka, taj proces se naziva multipleksiranje MPEG-2 toka podataka. Zbog postojanja verovatnoće pojave grešaka, u vidu promena na pojedinim bitima ili gubljenja celih paketa, u DTV se za prenos podataka koristi MPEG-2-TS format.

2.3 MPEG-2 - TS format podataka

Osnovne komponente MPEG-2 TS su elementarni tokovi (ES). Jedan DTV program, tipično se sastoji od jednog video ES, jednog ili više audio ES, kontrolnih podataka, titlova i slično. U terminologiji koja se koristi u DTV, jedan TV program nazivamo *DTV servis* ili *servis*. Prema MPEG-2 standardu video i audio podaci su organizovani u osnovne jedinice kojima

rukiju koder i dekoer tzv. audio i video *frame*. Elementarni tokovi se ne multipleksiraju direktno u TS već se prethodno pakerizuju. Nakon MPEG-2 kompresije, sistem procesor paketizuje dobijene ES i tako formira paketske elementarne tokove - PES (*Packetised Elementary Stream*). PES paketi su blokovi podataka fiksne ili promenljive dužine do 65536 bajta (64 KB) po bloku, uključujući i obavezno zaglavlje dužine minimum 6 bajta. PES paketi su najčešće organizovani tako da sadrže ceo broj osnovnih jedinica ES. Struktura PES paketa je prikazana na slici 2.



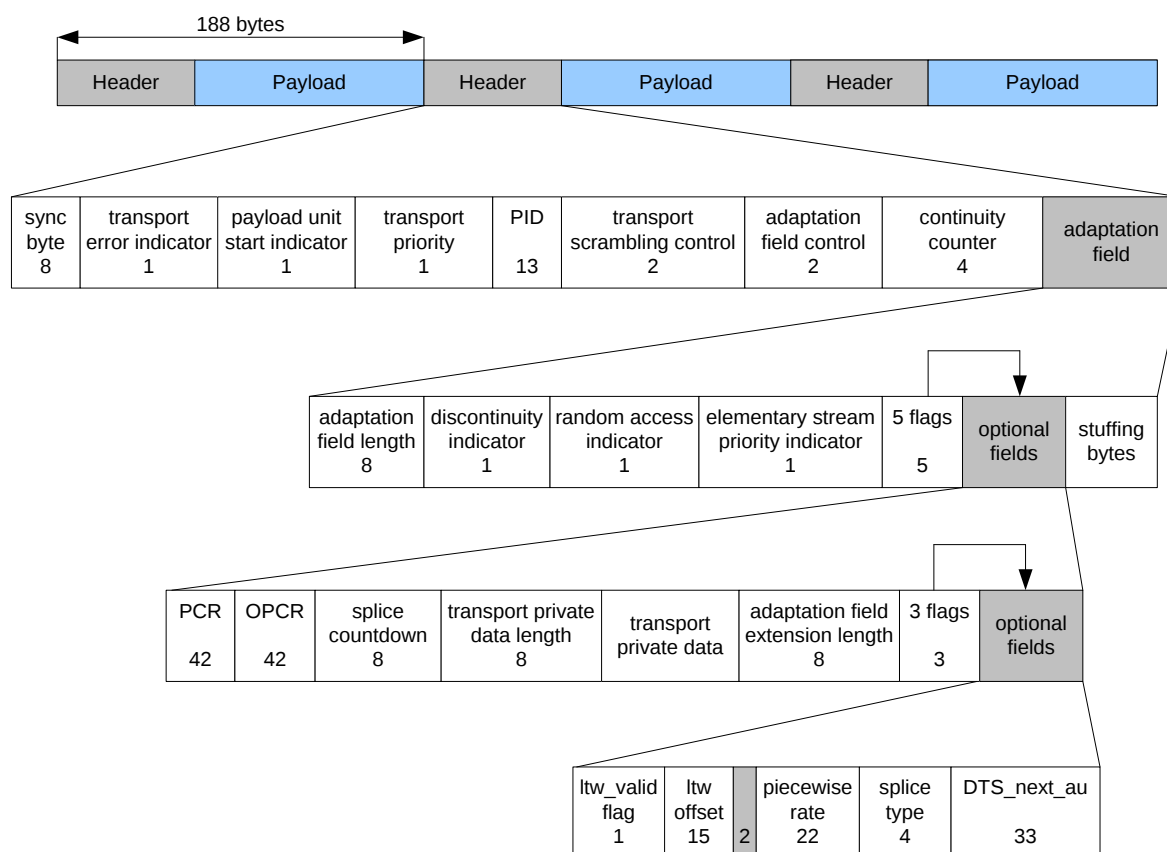
Slika 2 Struktura PES paketa

Nakon izvršene paketizacije elementarnog toka podataka, vrši se multipleksiranje više elementarnih tokova u jedinstven informacijski tok podataka, TS.

TS paketi su konstantne veličine 188 bajta, uvek podeljeni na zaglavlje i podatke. Minimalna veličina zaglavlja je 4 bajta. Struktura TS i jednog zaglavlja je prikazana na slici 3. PES paketi se mogu pakovati u TS na dva načina:

- 1) TS paket sadrži jedan ceo PES paket, što predstavlja jednostavniji način pakovanja. Ovo podrazumeva da MPEG enkoder formira PES pakete tako da mogu biti upakovani u jedan TS paket. U tom slučaju zaglavlje svakog TS paketa će imati postavljen *Payload Unit Start Indicator* bit da bi se označio početak PES paketa. S obzirom da PES paketi mogu biti promenljive dužine (mogu biti kraći od TS paketa), preostali neiskorišćeni deo TS paketa mora biti dopunjen odgovarajućim znakom (*padding value*) koji u slučaju MPEG-2 TS ima vrednost 0xFF. Ova dopuna je neophodna da bi se obezbedilo poravnavanje sledećeg PES paketa kako bi on počeo u narednom TS paketu.
- 2) U opštem slučaju PES paket može biti veći od jednog ili više TS paketa. Tada TS paket kojim započinje PES paket će u zaglavlju imati postavljen *Payload Unit Start Indicator* bit dok će se kod ostalih paketa koji nose isti PES paket uzastopno inkrementirati vrednost *Continuity Counter* polja. U jednom TS paketu se mogu nalaziti samo podaci koji pripadaju jednom PES paketu. Zbog toga PID vrednost iz zaglavlja TS paketa jednoznačno identifikuje PES čime je

omogućeno da prijemnik ne mora da obradjuje sadržaj TS da bi izdvojio pakete koji sadrže isti PES.



Slika 3 Struktura TS paketa

PID vrednost (*Packet Identifier - PID*) sadržana u TS zaglavlju igra ključnu ulogu u rukovanju TS. Slika 4 prikazuje dva ES multipleksirana u istom TS. Svaki TS paket sadrži podatke koji pripadaju jednom PES što se razlikuje odgovarajućim PID vrednostima. U primeru sa slike paketi koji nose audio podatke imaju PID vrednost 64, dok paketi sa video podacima imaju vrednost 51. Broj video paketa je veći od broja audio paketa. Takođe, paketi nisu pravilno raspoređeni u vremenu. MPEG-2 TS je vremenski multipleks ali zbog statističke prirode umetanja paketa i širine propusnog opsega, može doći do malih varijacija u vremenskom raporedu paketa. Ukoliko TS multiplekser nema raspoloživih podataka za formiranje paketa, u TS će se umetati *null* paketi (PID = 0x1FFF) da bi se održao zahtevani protok podataka (*bitrate*).



Slika 4 Multipleksirani TS sa dva ES

Multiplekser takođe ne sinronizuje PES-ove: vremenski interval izmedju dekodovanja dva PES paketa može biti, i obično jeste, različit u odnosu na vremenski interval izmedju njihovog kodovanja. Zbog toga je na prijemu neophodna posebna procedura sinhronizacije PES.

2.4 TS sa jednim i više programa

TS može u sebi nositi podatke koji pripadaju jednom TV ili multimedijalnom programu (npr. jedan video PES, jedan audio PES i pet titl PES) tj. DTV servisu. Ovakav tip TS se zove *Single Program Transport Stream (SPTS)*. Jedan SPTS sadrži sve podatke potrebne za reprodukciju kodovanog TV kanala ili multimedijalnog sadržaja. SPTS može sadržati samo jedan video i jedan audio PES, ali u praksi obično sadrži i druge PES istog ili različitog tipa. Svi PES koji pripadaju jednom SPTS dele istu vremensku bazu. Iako neki proizvođači DTV opreme omogućavaju i generisanje ovakvog tipa TS, SPTS nije uobičajen način prenosa u DVB sistemima.

U većini slučajeva jedan ili više SPTS kombinuju se u *Multi Program Transport Stream (MPTS)*. Ovaj opštiji format TS se koristi u DVB sistemima. On takodje sadrži sve potrebne kontrolne podatke (*Program Specific Information - PSI*), potrebne za pravilnu prezentaciju kodovanih informacija kao i ostale podatke koje treba poslati prijemnoj strani.

U većini slučajeva je potrebno dekodovanje jednog ES uskladiti (sinhronizovati) sa dekodovanjem drugog ES, npr. da bi se reprodukcija zvuka uskladila sa reprodukcijom odgovarajućeg videa. Neki tokovi, međjutim, ne zahtevaju sinhronizaciju (npr. ukoliko strim nudi uslugu preuzimanja softvera). Da bi se omogućila sinhronizacija strimova oni mogu sadržati vremenske markere (*time stamps*). Postoje dva tipa vremenskih markera.

1. Prvi tip markera su tzv. Referentni vremenski markeri (*reference time stamps*) koji nose informaciju o tekućem vremenu kodera. Ove markere definišu PES, PS i TS sintaksa. DVB sistemi koriste TS sintaksu za slanje referentnih vremenskih markera. Zaglavlje TS paketa može sadržati PCR polje (*PCR - Program Clock Reference*) ili vremensku bazu, koje nosi ovu informaciju.
2. Drugom tipu vremenskih markera pripadaju *Decode Time Stamp (DTS)* i *Presentation Time Stamp (PTS)*. Ovi markeri su vezani za sadržaj na koji se odnose i nalaze se u zaglavlju PES paketa. Nose informaciju o vremenskom trenutku kada je potrebno dekodovati i prezentovati sadržaj na koji se odnose (obično video i audio).

Za sinhronizaciju TS strimova na prijemnoj strani neophodno je poznavati trenutnu vrednost oba tipa vremenskih markera. PTS i DTS su blisko povezani sa PCR: izražavaju se istim jedinicama i računaju od istog vremenskog trenutka.

2.5 Osnove MPEG-2 kompresije video podataka

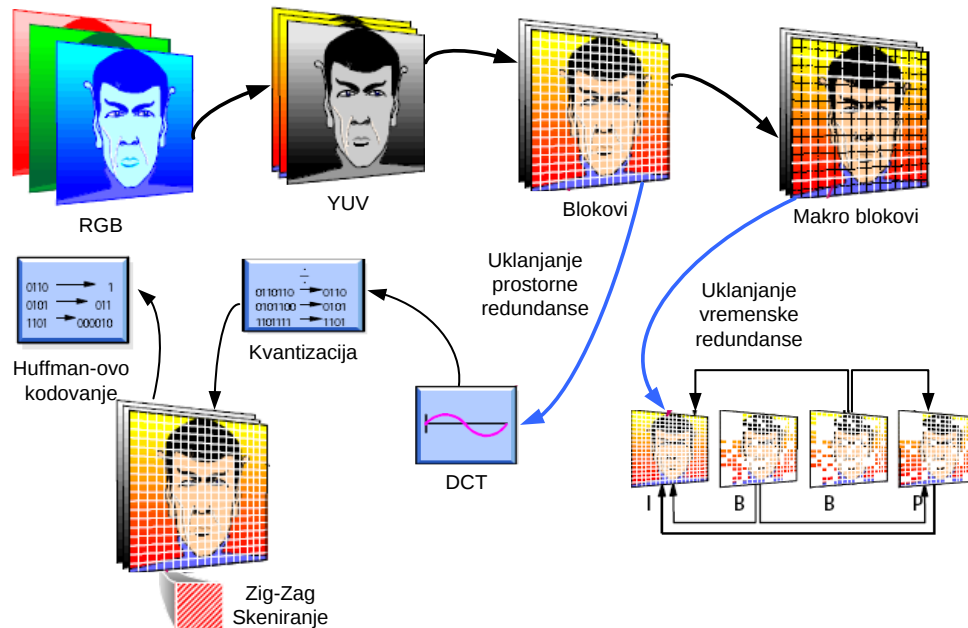
U osnovi video kompresije se nalazi činjenica da je ljudsko oko slabo osetljivo na nijanse boja dok je više osetljivo na osvetljaj. Takodje, uobičajeno je da se unutar slike koja se kodira nalaze regioni sa identičnim vrednostima piksela, redundansa, što omogućava kompresiju slike i smanjenje ukupnog bitskog protoka (*bitrate*). Redundansa postoji u okviru jedne slike i ona se naziva prostorna redundansa, na primer jednoboja pozadina iza voditelja vesti. Medju susednim slikama javlja se vremenska redundansa, kao posledica linearnog pomeranja objekta u toku vremena. Primer za vremensku redundansu je snimak broda koji plovi na pučini.

U procesu otklanjanja prostorne redundanse koriste se diskretna kosinusna transformacija (*Discrete Cosine Transform - DCT*), kvantizavija i Hafmanovo kodovanje (*Huffmann*), a sve da bi se predvidela vrednost piksela na osnovu vrednosti susednih piksela i kako bi se minimizirao ukupni bitski protok (*bitrate*). Primenom ove tri metode dobija se *Intra-Frame (I-Frame)*.

Predikcija i kompenzacija pokreta, se koriste u procesu otklanjanja vremenske redundanse za predviđanje vrednosti bloka piksela jedne slike na osnovu vrednosti iste grupe piksela u susednim slikama. Na slici 5 su prikazane osnovne metode za kompresiju slike koje se koriste prema MPEG-2 standardu.

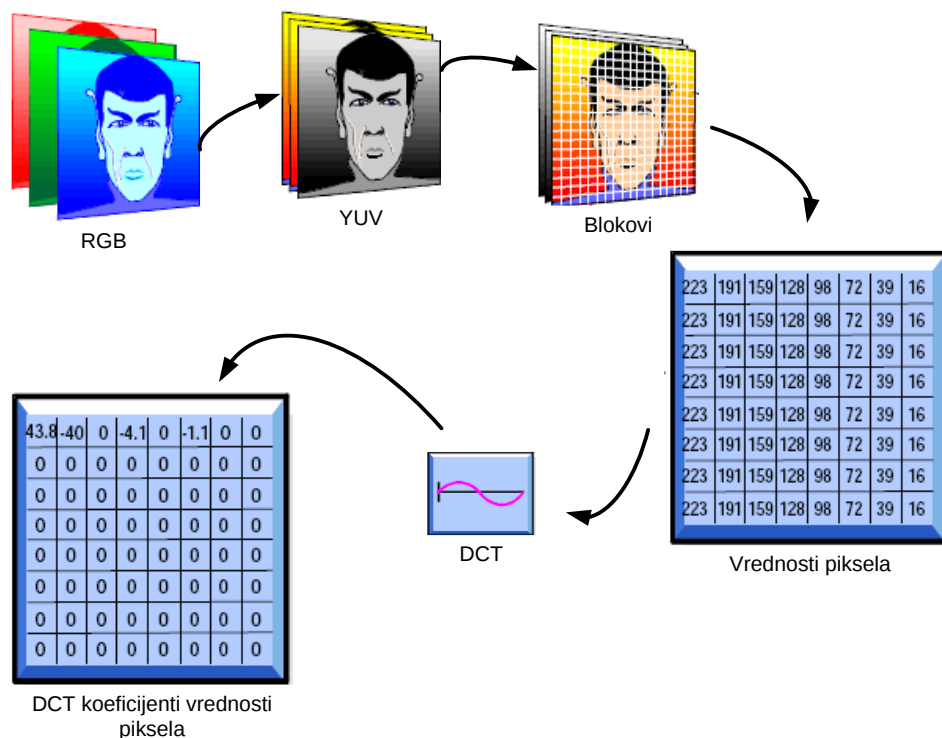
2.5.1 Prostorna redundansa - kodovanje piksela korišćenjem Diskretne Kosinusne Transformacije (DCT)

Prvi korak kompresije je da se napravi I slika (*Intra-frame*), jer će se predikcija narednih slika koje pripadaju istoj grupi slika (*Group Of Pictures - GOP*) izvršavati na osnovu njega. Pošto je ljudsko slabije osetljivo na nijanse boja, pretvaramo sliku iz RGB prostora boja (*Red-*



Slika 5 Osnove MPEG-2 video kompresije

Green-Blue - RGB) u YUV prostor boja. Y komponenta je lumen signal ili osvetljaj slike, a U i V komponente nose informaciju o boji. Koristimo YUV prostor boja pri kompresiji jer je ljudsko oko osetljivije na osvetljaj nego na boju, pa se na osnovu toga može izvršiti efikasnija kompresija slike, korišćenjem više bita za kodovanje Y, a manje za kodovanje U i V komponenti.

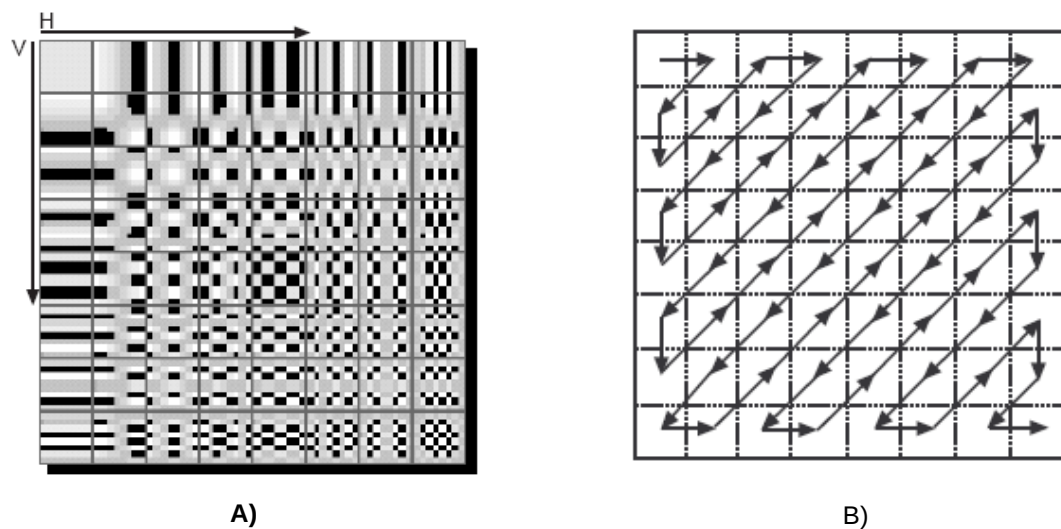


Slika 6 Prostorna redundansa - DCT

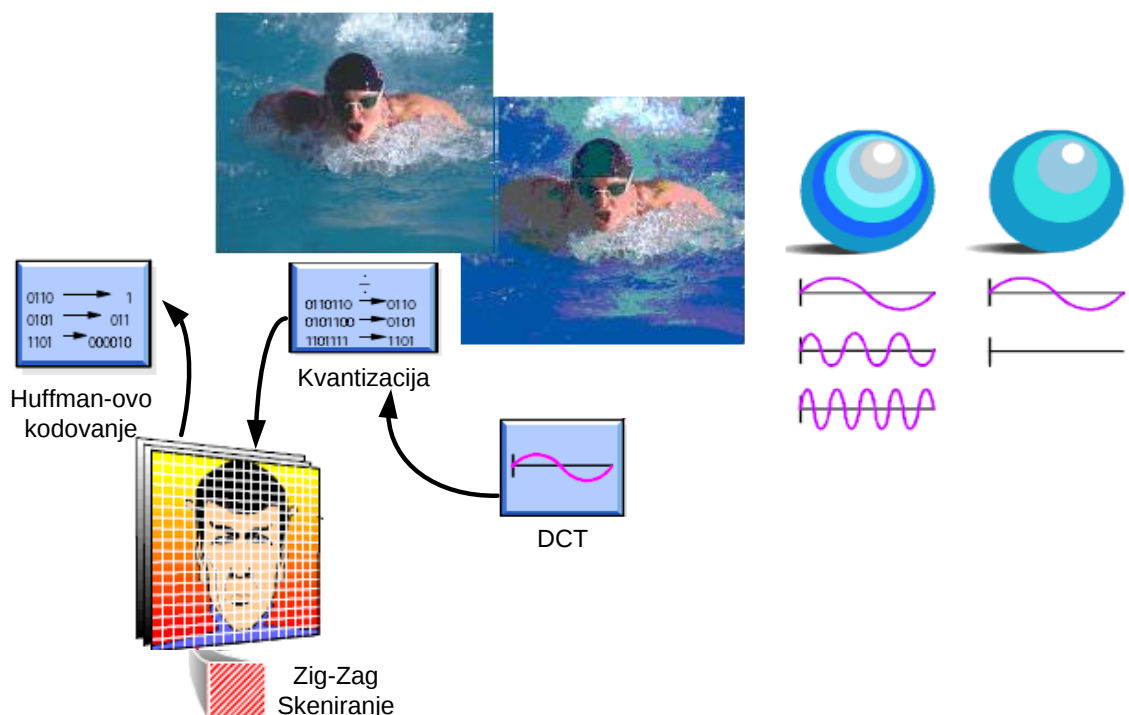
Nakon prevodjenja slike u YUV prostor, izračunava se DCT koeficijent svakog piksela korišćenjem vrednosti svih ostalih piksela iz bloka. Blokovi su veličine 8x8 piksela kako bi se skratilo vreme računanja. Gornji levi piksel iz bloka je DC komponenta bloka. Što je manja razlika vrednosti piksela i njemu susednih piksela, manja će biti i vrednost DCT koeficijenta. Korišćenjem inverzne diskretne kosinusne transformacije (*Inverse Discrete Cosine Transform - IDCT*) moguće je potpuno tačno rekonstruisati vrednost svakog piksela, jer je DCT transformacija bez gubitaka.

2.5.2 Prostorna redundansa - kvantizacija i entropijsko kodovanje

Kvantizaciona matrica je konstruisana tako da njeni koeficijenti postaju sve veći prateći glavnu dijagonalu matrice. $M[0,0]$ je DC komponenta bloka, a ostali koeficijenti predstavljaju detalje slike, slika 7a. Nakon kvantizacije, najveći broj DCT koeficijenata postane jednak nuli. Sledeći korak je da se maksimizira broj uzastopnih nula i da se izvrši entropijsko kodovanje, za to se koristi Zig-Zag skeniranje, čiji je princip prikazan na slici 7b.



Slika 7 A) Uticaj pojedinih DCT koeficijenata na kvalitet slike B) Zig-Zag skeniranje



Slika 8 Prostorna redundansa - Kvantizacija i entropijsko kodovanje

Entropijsko kodovanje je ono koje određuje kodove za pojedine vrednosti na osnovu toga koliko se one često pojavljuju, što se više puta pojavi neka vrednost, njegov će binarni kod biti predstavljen sa manje bita.

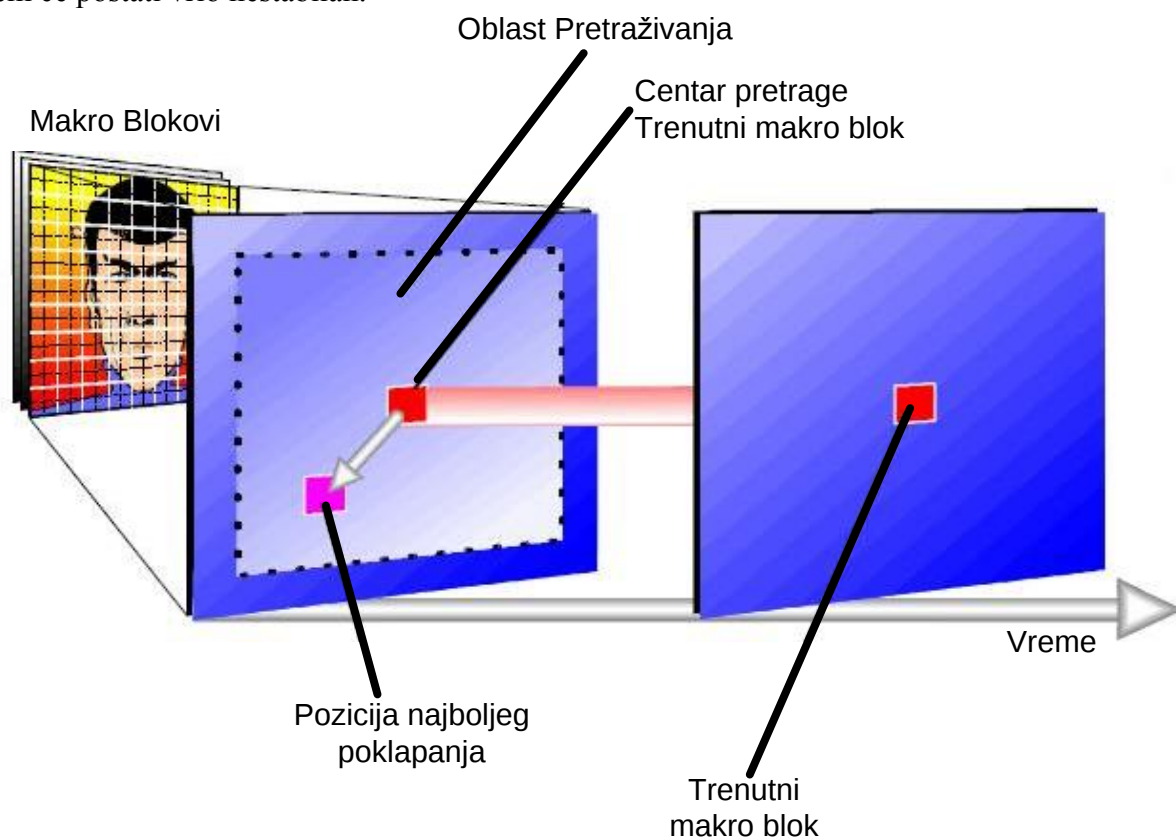
Kvantizacijom se najveći broj DCT koeficijenata svodi na nulu, a onda se entropijskim kodovanjem otklanja ponavljanje vrednosti koeficijenata. Kao posledica opisanih postupaka narušava se kvalitet slike, koji se ogleda u gubitku finih detalja slike. Na slici 8 je prikazana razlika kvaliteta slike, ako je korišćena kvantizacija i ako nije.

2.5.3 Vremenska redundansa

Uklanjanje vremenske redundanse najviše utiče na smanjenje bitskog protoka. Primenom metoda za uklanjanje prostorne redundanse, dobija se I slika, što je u osnovi slika kodovana JPEG algoritmom. Ona nosi sve potrebne informacije za dekodovanje. Sledeći korak je analiza naredne slike, tj. ispitivanje sličnosti sa prethodnom. Dve stvari je moguće uraditi kako bi se minimizirao bitski tok.

- 1) Provera da li je isti makroblok na istom mestu i u sledećem frejmu. Ako je na istom mesto, neće se izvršiti nikakvo kodovanje.
- 2) Pretraživanje unutar I slike u potrazi za istim takvim makroblokom na bilo kojoj poziciji. Ako postoji, preneće se samo vektor pomeraja sa stare lokacije na novu. Samo ako je blok sasvim nov, izvršiće se ponovno kodovanje makroblokova.

Ova procedura mnogo smanjuje ukupni bitski tok. Kao posledica uklanjanja vremenske redundanse, javlja se zavisnost između sekvence slika. U slučaju da dodje do najmanje greške sistem će postati vrlo nestabilan.

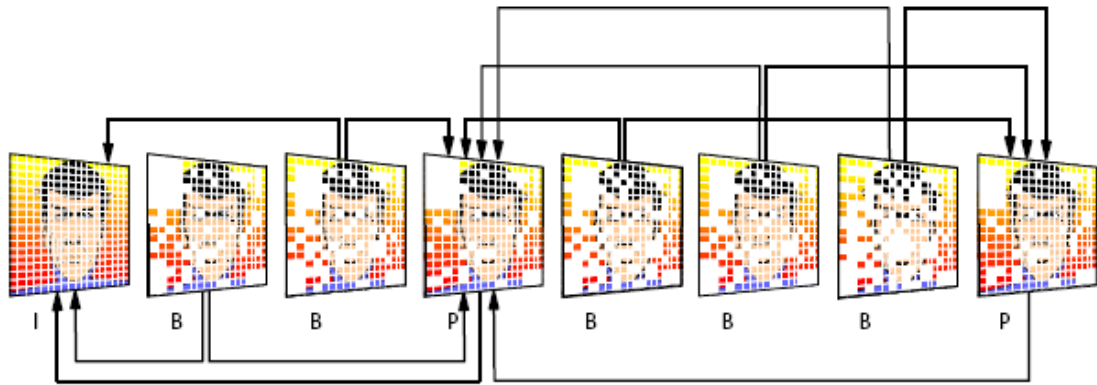


Slika 9 Vremenska redundansa

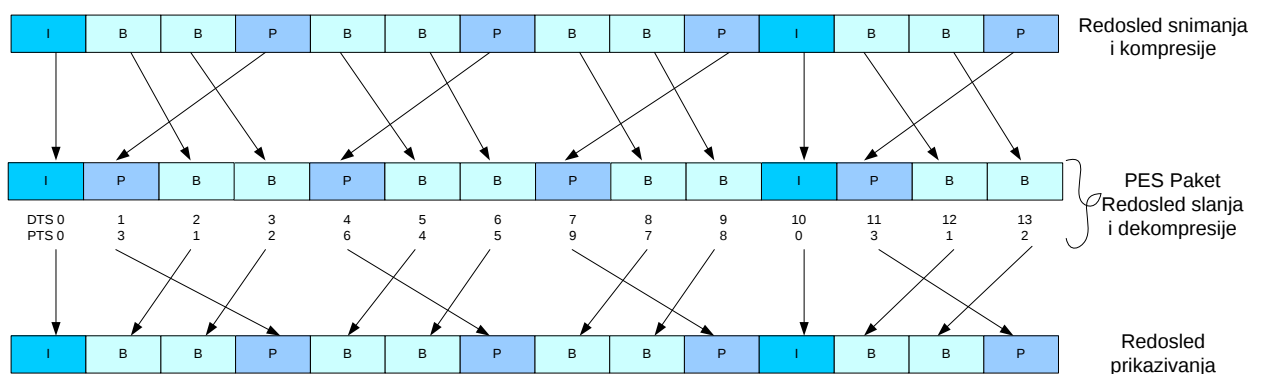
I frejm (*intra-frame*) sadrži kompletnu informaciju. Periodičnom transmisijom I slika održava se stabilnost sistema, tako da su one, iako najmanje komprimovane, najbitniji element stabilnosti sistema. P frejmovi (*predicted frame*) se rekonstruišu na osnovu prethodne I ili P slike. Bi-direkciono predikovani frejm ili B slika (*bi-directional predicted frame*) koristi prethodnu i sledeću referentnu sliku (I ili P). B slikama se postiže najbolja kompresija, ali su najpodložnija greškama.

2.5.4 Redosled slika za slanje i dekodovanje

Po preporuci koja je navedena u MPEG-2 standardu, I slike bi trebale da se šalju minimum 2 puta u sekundi. Idealni redosled bi bio kao što je prikazano na slici 11, što se u praksi retko sreće. Takođe postoje dve vrste GOP-a, otvoreni (*open GOP*) i zatvoreni (*closed GOP*). Zatvoreni GOP je potpuno nezavisna celina, dok je za dekodovanje sadržaja otvorenog GOP neophodno prisustvo narednog. Zatvoreni GOP počinje I slikom, i nakon nje su dve B koje pripadaju istom GOP-u, pa zatim P i tako dalje, dok se kod otvorenog GOP-a, nakon I slike pojavljuju dve B koje su iz prethodnog GOP-a.



Slika 10 Izgled toka podataka nakon kompresije

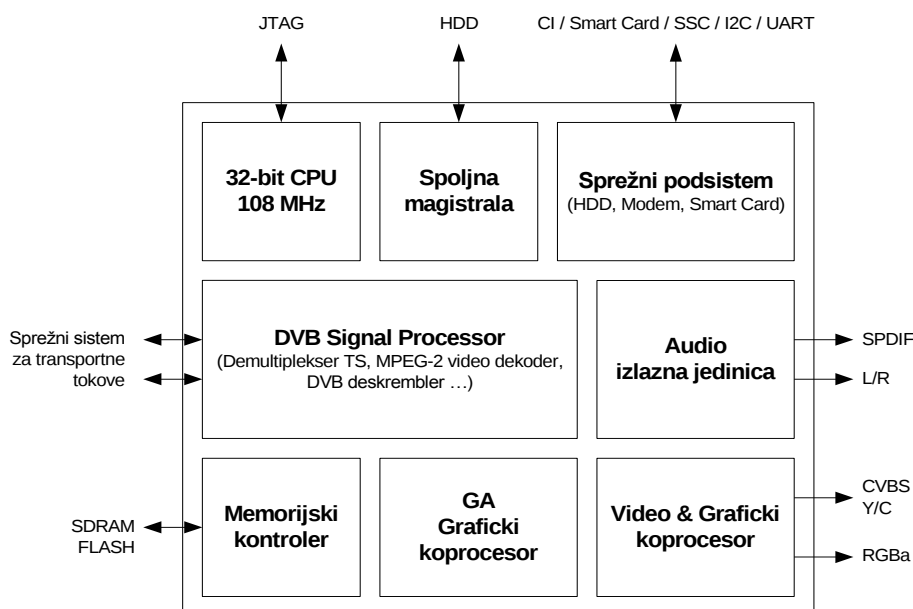


Slika 11 Redosled kodovanja, slanja, dekodovanja i prikaza slika

Na slici 11 je prikazan redosled kodovanja slika, on je isti kao i redosled kojim su one snimljene. Nakon kodovanja, slike se slažu u redosled koji je pogodan za dekodovanje. Najpre I slika, jer za njeno dekodovanje nije potrebna ni jedna druga slika. Zatim P slika, sledeća referentna slika, za njeno dekodovanje je potrebna prethodna referentna slika, pa onda B slike, za čije je dekodovanje potrebno poznavati i prethodnu i narednu referentnu sliku. Nakon dekodovanja, slike se ponovo sortiraju u redosled za prikazivanje, a to na osnovu PTS (*presentation time stamp*) reference koja je vezana za svaku sliku i označava redosled prikazivanja unutar GOP-a.

3. Opis namenske platforme

Pod imenom MDE9500 ili MDE podrazumeva se familija Micronas-ovih integrisanih kontrolera namenjenih dekodovanju digitalnih i analognih TV signala. Svi tipovi MDE kontrolera nude kompletno rešenje za razvoj prijemnika DVB i/ili analognog TV signala.



Slika 12 Shema namenske platforme

Namenska platforma sadrži MDE9500B čip koji objedinjuje sve funkcije, neophodne jednom digitalnom DVB TV dekodernu (*single-chip digital-TV DVB decoder*). Sadrži TS demultiplekser, Simulcrypt deskrembler i MPEG video i audio dekodern. Radni takt procesora je 20.25 MHz. Obezbedjena je puna podrška za DVB, kao i mogućnost dekodovanja FTA (*free-to-air*), CA (*conditional access*) i DVB-skremblovanog prenosa i podrška za MPEG-2 TS ulaz sa bitskom brzinom protoka do 15Mbps. Platforma je dizajnirana za iDTV (50Hz i 100Hz) i Set-top Box (STB) aplikacije. Ima dvostruki TS ulaz sa podrškom za CI (*CI - Common Interface*). Potpuno je funkcionalna za snimanje i reprodukciju, snimanje jednog ili dva različita servisa, odloženo gledanje jednog servisa i snimanje drugog servisa simultano. Na njoj je ugrađen analogni video dekodern sa podrškom za više standarda, NTSC sa rezolucijom 720 x 480 na 30 fps (interlace) i PAL sa 720 x 576 na 25 fps (interlace). Takodje, može da sadrži jedan SDRAM memorijski modul za jednostavnije aplikacije, ili dva SDRAM memorijska modula za naprednije aplikacije, od kojih je jedan modul video/audio SDRAM i koristi se za dekodovanje MPEG A/V, a drugi se koristi kao dodatni modul za potrebe aplikacije i procesora.

MDE9500 je čip dizajniran za STB aplikacije i integrirani digitalni TV (iDTV) za frekvencije 50Hz i 100Hz. U svim slučajevima moguće je čip koristiti za napredne proizvode sa VSDRAM+ESDRAM ili za jednostavne proizvode koristeći samo VSDRAM.

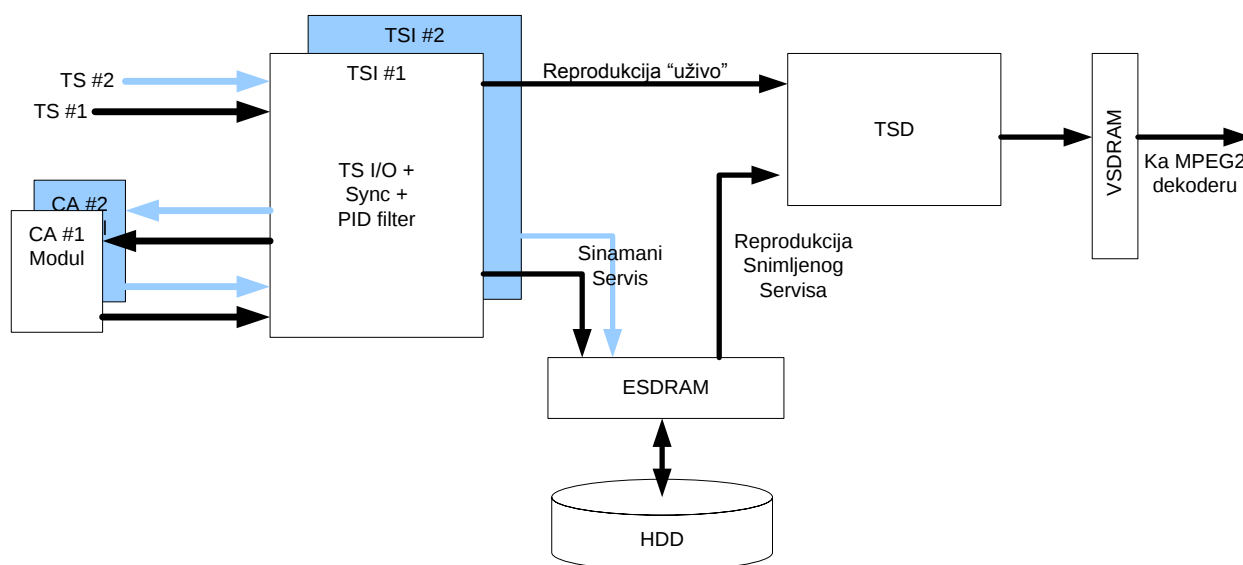
Osvežavanje od 100Hz se koristi u novim “*flicker-free*” televizorima. Svaki frejm se umesto jednom, kao kod standardnih 50Hz televizora, prikazuje dvaput, i to gledaocu stvara privid mirnije i stabilnije slike i čini je prijatnijom za oko.

Platforma se sastoji od više blokova, a najbitniji sa stanovišta aplikacije su:

- **Jedinica za prijem transportnog toka (JPTT)**- preko ulazne sprege stižu četiri TS ulaza. Sprega je implementirana kao dva identična bloka. Prvi od njih služi za odabiranje TS koji će da bude demultipleksiran, njegovo filtriranje tj. odabiranje blokova sa željenim PID vrednostima i smešta ih u odgovarajući bafer za trajno čuvanje ili odloženo gledanje. Drugi blok izdvaja sadržaj TS za sekundarni DVB servis i radi njegovo simultano trajno čuvanje i odloženo gledanje.
- **Demultiplekser toka podataka (DTP)** izvršava demultipleksiranje ulaznog TS, i smešta audio i video podatke kao i kontrolne podatke u odgovarajuće zone u memoriji.
- **Jedinica za dekodovanje slike (JDS)** je odgovorna za dekodovanje MPEG-2 video toka podataka i rekonstrukciju kodovanih frejmova. Sadrži hardverski Hafmanov dekodir. Prvo se izvršava de-skaliranje i de-kvantizacija, zatim inverzna diskretna kosinusna transformacija (*IDCT - Inverse Discrete Cosine Transform*) i na kraju rekonstrukcija slike.
- **Jedinica za obradu slike (JOS)** je odgovorna za sve obrade izlaznog video signala, skaliranje, promenu formata prikaza (*aspect-ratio*), mešanje slojeva slike, kao i za horizontalnu i vertikalnu sinhronizaciju.

3.1 Tok obrade MPEG ulaznih podataka

Ulazni podaci mogu doći iz tri izvora, sa JPTT, CA modula ili HDD. Izabrani ulaz obradjuje JPTT. TS koji se obradjuje može sadržati u sebi više servisa. Za svaki servis postoji set paketa koji su za njega vezani (koji se na njega odnose). JPTT jedinica, koja je odgovorna za filtriranje i demultipleksiranje toka podataka, se konfiguriše da izdvoji samo potrebne pakete. Svaki paket se identifikuje sa jedinstvenim PID brojem. DTP filtrira i izdvaja pakete sa relevantnim PID vrednostima i demultipleksira ih u video, audio i informacijske (*SI - System Information*) pakete.



Slika 13 Tok obrade ulaznih podataka

Svakom tipu paketa je dodeljena posebna bafer zona u VSDRAM memoriji. Video paketi se smeštaju u *Video Code Buffer*. To su neobradjeni podaci koje koristi MPEG video dekodler (JDS). Audio paketi se smeštaju u *Audio Code Buffer*, neobradjeni podaci koje koristi Audio DSP Dekoder (ADP - *Audio DSP Decoder*). I na kraju SI paketi se dalje filtriraju prema potrebnim sekcijama. CPU koristi neobrađene podatke za navigaciju u elektronskom programskom vodiču (EPG - *Electronic Program Guide*) kao i za dekodovanje teletekst informacija.

3.2 Skladištenje MPEG ulaznih podataka

MDE95000 je sposoban za snimanje jednog ili dva TS paralelno. Moguće je dekodovati i snimati isti TV servis, ili dekodovati i snimati različite servise. Za snimanje DTV servisa nezavisno od procesa dekodovanja, JPTT jedinica iz toka podataka izdvaja pakete koji su vezani za izabrani servis po njihovim PID vrednostima.

Prvo CPU izabira željeni TS i koje pakete sa kojim PID vrednostima je potrebno snimati. JPTT vrši filtriranje paketa po PID vrednostima na nivou fizičke arhitekture. Filtrirani TS paketi se privremeno smeštaju u ESDRAM preko DMA, bez opterećivanja CPU. Moguće je obavljati dva DMA tranfera paralelno za svaki TS ulaz. Dalje se podaci iz bafera šalju van MDE9500 kontrolera preko drugog DMA transfera i to do lokalnog ATA/IDE HDD. Reprodukcijska je moguća i za vreme snimanja jednog ili dva DTV servisa, znači najviše 3 konkurentna toka podataka mogu biti snimana/reprodukovana. Da bi ovakvo konkurentno korišćenje DMA bilo konzistentno, CPU upravlja magistralom kao vremenskim multipleksom. Prvo se prenose podaci iz prvog bafera za snimanje, zatim iz drugog bafera za snimanje i na kraju iz bafera za reprodukciju. CPU kontroliše popunjenost sva tri bafera kako ne bi došlo do njihovog prepunjavanja i prepisivanja ili gubljenja podataka.

4. Analiza problema

4.1 Reprodukcija sačuvanog TS

Osnovna prednost TS snimljenog na HDD jeste da se za vreme reprodukcije podacima može pristupiti na slučajan način (*random access*), što nije moguće u toku direktne reprodukcije (*live stream*). Pri reprodukciji je neophodno da slučajni pristup bude ograničen na pristupanje početku GOP, koji uvek počinje intra frejmom (I slika). Po preporuci MPEG2 standarda, trebalo bi da u jednoj sekundi bude dva GOP, kako bi se nakon promene kanala, maksimalni interval promene servisa ograničio na $\frac{1}{2}$ sekunde.

Da bi se obezbedile najbolje performanse snimljenom TS je pridružena datoteka za skladištenje indeksnih informacija, koje omogućuje jednostavnu navigaciju kroz podatke, a time i slučajan pristup datoteci. Indeksni podaci predstavljaju adrese početaka GOP, slika i vremenskih informacija. U slučaju da se radi o kriptovanom toku podataka, GOP ne može biti indeksiran, tako da indeksna datoteka ne sadrži informacije o položaju GOP, odnosno ključnih slika.

4.2 Načini čuvanja zaštićenog sadržaja na HDD (DRM)

Možemo izdvojiti tri osnovna načina snimanja zaštićenog sadržaja (DRM - *Digital Rights Management*) na HDD:

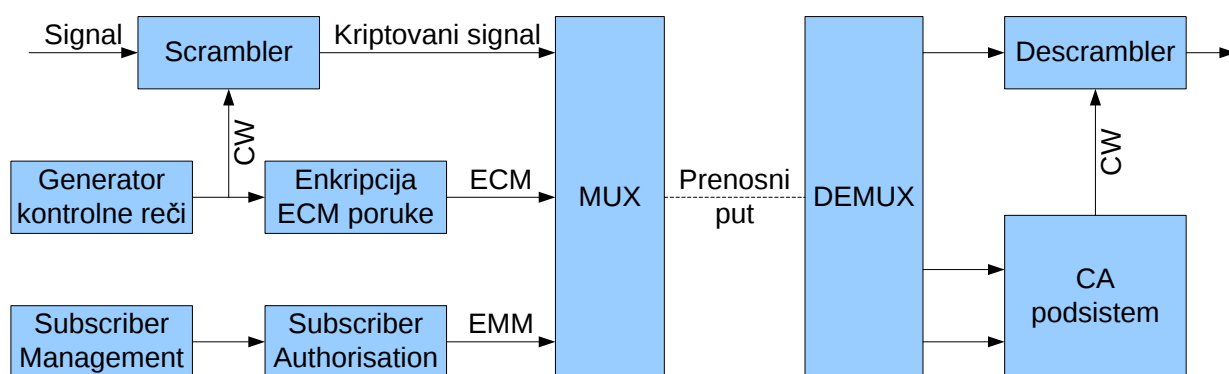
- **Bez enkripcije** - Na HDD PVR uređaja se čuva nekriptovan sadržaj. Neposredno pre upisivanja na HDD sadržaj se dekriptuje i u tom obliku čuva. Ovakav pristup čuvanju toka podataka nije prihvatljiv sa stanovišta vlasničkih prava, jer je sadržaj moguće kasnije pročitati i na uređajima koji za to ne poseduju ovlašćenje, pa se u tom slučaju radi o povredi autorskih prava.
- **Sa enkripcijom** - sadržaj se čuva u kriptovanom obliku, a dekripcija se vrši neposredno pre dekodovanja i reprodukcije.
- **Lokalna enkripcija** (ili re-enkripcija) – Ovakav pristup čuvanja kriptovanih podataka se najčešće primenjuje. Po ovom modelu, sadržaj se pre upisivanja na HDD dekriptuje, a zatim ponovo enkriptuje nekim od poznatih algoritama za kriptovanje koji je implementiran u datom uređaju. Re-enkripciju može da obavlja sama fizička arhitektura (*hardware*) ili programski CA podsistem. CA podsistem, pruža veći nivo sigurnosti i fleksibilnosti pri re-enkripciji.

4.3 Kriptovanje i uslovni pristup podacima

Sistem za uslovni pristup (*Conditional Access - CA*), čiji je blok dijagram prikazan na slici 14, obuhvata metode šifrovanja (*scrambling*) i enkripcije podataka da bi se sprečio neovlašćen pristup određenim DTV servisima. Enkripcija je proces zaštite tajnih ključeva, koji se prenose

zajedno sa šifrovanim signalom, da bi se omogućilo njegovo dešifrovanje. Ključ za šifrovanje, tzv. kontrolna reč (*Control Word*) se šalje prijemnoj strani u kriptovanom obliku u vidu tzv. entitlement control poruka (*Entitlement Control Message - ECM*).

CA podistem prijemnika će dekriptovati kontrolnu reč samo ukoliko je ovlašćen. Ovlašćenje se prijemniku šalje preko entitlement management poruka (*Entitlement Management Message - EMM*). Ova slojeviti pristup zaštiti podataka je osnova svih CA sistema koji su danas u upotrebi. Da bi dešifrovao zaštićene audio/video podatke, prijemnik mora poznavati odgovarajuću kontrolnu reč. Kontrolna reč se prenosi šifrovana u zasebnom bloku i transformisana je u ECM. Sve potrebne informacije za dešifrovanje se prenose do prijemnika putem CAT (*Conditional Access Table - CAT*) tabela u skladu sa MPEG-2 standardom. Osim ECM podataka, CAT sadrži i EMM podatke. EMM podaci su namenjeni unapred odredjenim prijemnicima, npr. *PayPerView* usluga. Svi prijemnici upoređuju kod za autorizaciju koji se nalazi u EMM sa kodom koji se nalazi u lokalnoj *Smart* kartici u prijemnom uređaju. Ukoliko je uređaj autorizovan za kdekripciju podataka, kontrolna reč se generiše iz ECM i dekripcija podataka može da počne. Ukoliko dekriptovanje nije autorizovano, korisnik se o tome obaveštava odgovarajućom porukom na ekranu. Na slici 14 je prikazan funkcionalni blok-dijagram sistema za uslovni pristup podacima. Kontrolna reč se obično menja u intervalima od 10 sekundi. EMM (ponekad nazvan i *multi-session key*) se menja obično u mesečnim intervalima da bi se smanjila mogućnost neovlašćenog pristupa podacima.



Slika 14 Blok dijagram CA podsistema

Podaci se šifruju koristeći dve kontrolne reči, parni i neparni ključ. U zavisnosti od vrednosti 2-bitnog polja **transport scrambling control**, koje se nalazi u TS paketu, poznato je da li je sadržaj kriptovan i ako jeste kojim od ova dva ključa. U Tabeli 1 su date moguće vrednosti polja i njihova interpretacija.

Vrednost	Opis
00	Nekriptovan sadržaj
01	Korisnik definiše
10	Korisnik definiše (parni ključ)
11	Korisnik definiše (neparni ključ)

Tabela 1 Vrednosti transport scrambling control polja

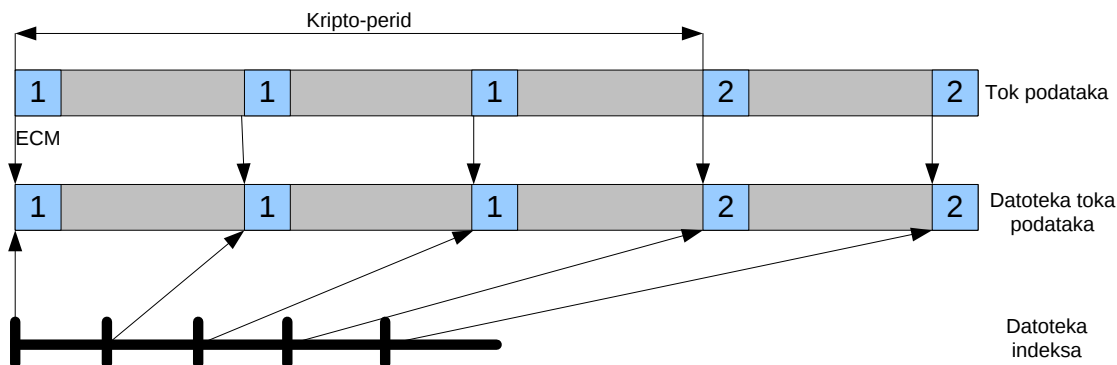
4.4 Rukovanje ECM porukama

Za pouzdanu reprodukciju zaštićenog (kriptovanog) sadržaja neophodna je dobra sinhronizacija sa podsistemom za generisanje kodnih ključeva. Pošto se kodni ključevi generišu na osnovu sadržaja ECM poruka, rukovanje ECM porukama je detaljno prikazano.

4.4.1 Indeksiranje kompletnog ECM toka

Neposredno pre prenosa, sadržaj se kriptuje sa kontrolnim rečima, koje se emituju u ECM strukturi. PVR će ove ECM poruke sačuvati zajedno sa sadržajem. Uobičajena praksa je da se

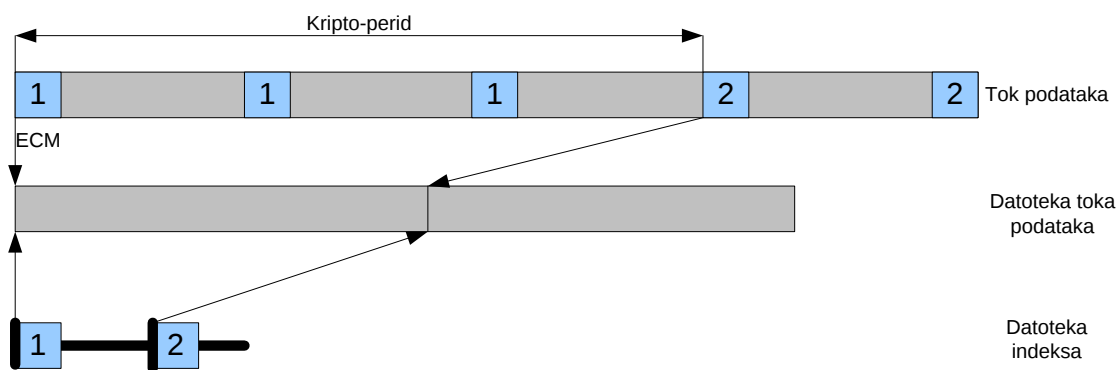
dekriptovanje, pri normalnoj brzini reprodukcije, vrši “u letu” linearno čitajući kriptovani sadržaj i ECM poruke iz multipleksiranog toka podataka. U trik modovima rada, ubrzana reprodukcija u napred ili u nazad, potrebno je da dekriptovanje bude brže od ralnog vremena, pa je stoga neophodno obezbediti indeksne informacije koje će ukazivati na adrese na kojima se nalaze ECM podaci kako bi se iz njih brzo mogli generisati ključevi i otpočeti dekriptovanje. Indeksna datoteka bi onda, kao što je prikazano na slici 15, pored podataka o početku GOP-a i vremenskih podataka, trebala da sadrži i adrese početaka ECM poruka, iz dataoteke sa tokom podataka.



Slika 15 Izgled datoteka podataka i indeksa za slučaj indeksiranja kompletnog ECM toka

4.4.2 Indeksiranje promena unutar ECM toka

Znajući da ECM poruke imaju isti sadržaj dok ne istekne krypto-period, vreme nakon kog se menjaju kontrolne reči, potrebno je sačuvati jedino poziciju promene sadržaja ECM poruka. Na ovaj način se smanjuje količina podataka koja se upisuje. Umesto adresa i sadržaja svih ECM poruka upisuju se samo adresa i sadržaj nove ECM poruke. Za razliku od prethodnog slučaja, kada se upisuju samo adrese, u indeksnu datoteku se sada upisuje i adresa i sadržaj ECM poruke. Ovo znači da će tokom snimanja ECM poruka i njena pozicija da se upišu samo u indeksnu datoteku, što se vidi na slici 16. Tokom reprodukcije će se obezbediti da se kodni ključevi pravovremeno povežu sa odgovarajućim GOP-om.



Slika 16 Izgled datoteka podataka i indeksa za slučaj indeksiranja promena unutar ECM toka

5. Opis realizacije sistema

5.1 Opis arhitekture PVR

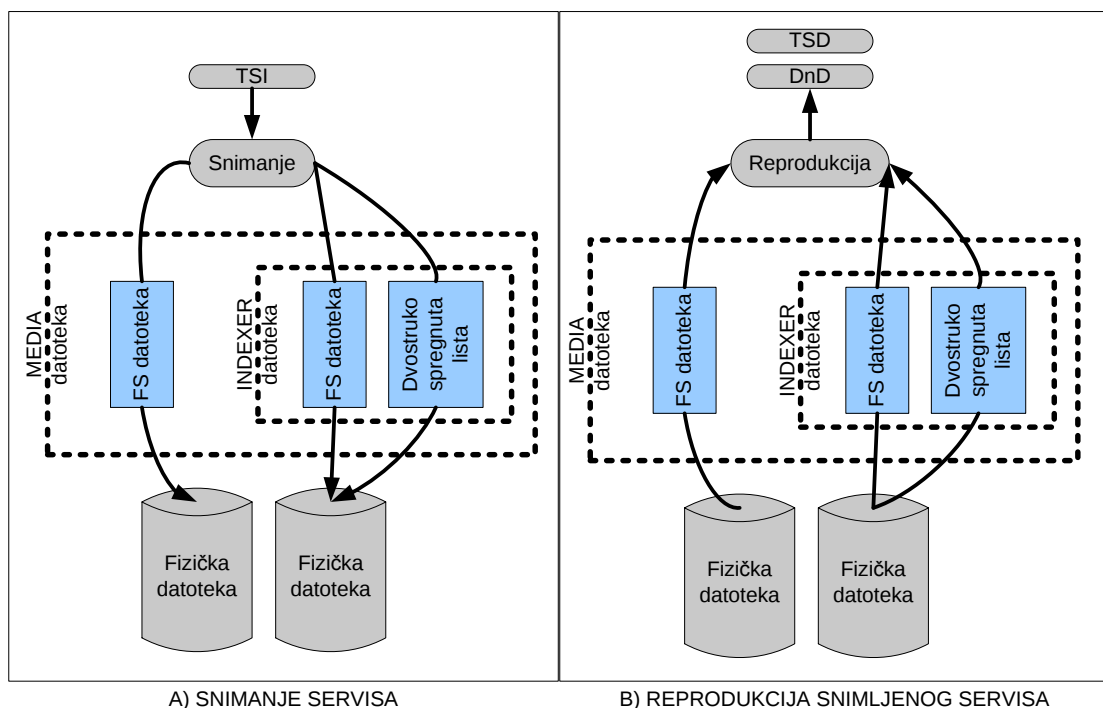
PVR se sastoji iz dva osnovna programska modula:

- **player** proces koji je zadužen za reprodukciju snimljenog DTV servisa
- **recorder** proces zadužen za snimanje izabranog DTV servisa na HDD.

Svi resursi koji su nophodni za rad ova dva procesa su implementirani kao posebni objekti.

- **Task** jednostavan konačni automat koji se koristi tokom života PVR procesa, i predstavlja vezu sa operativnim sistemom
- **List** realizacija dvostruko spregnute liste sa brzim pristupom
- **Indexer** objekat za manipulaciju indeksnim datotekama
- **File** generalizacija datotečnog sistema
- **Media** pristup pojedinim segmentima snimljenog TS-a
- **ScramblingCtrl** Kontrola rukovanja ECM podacima

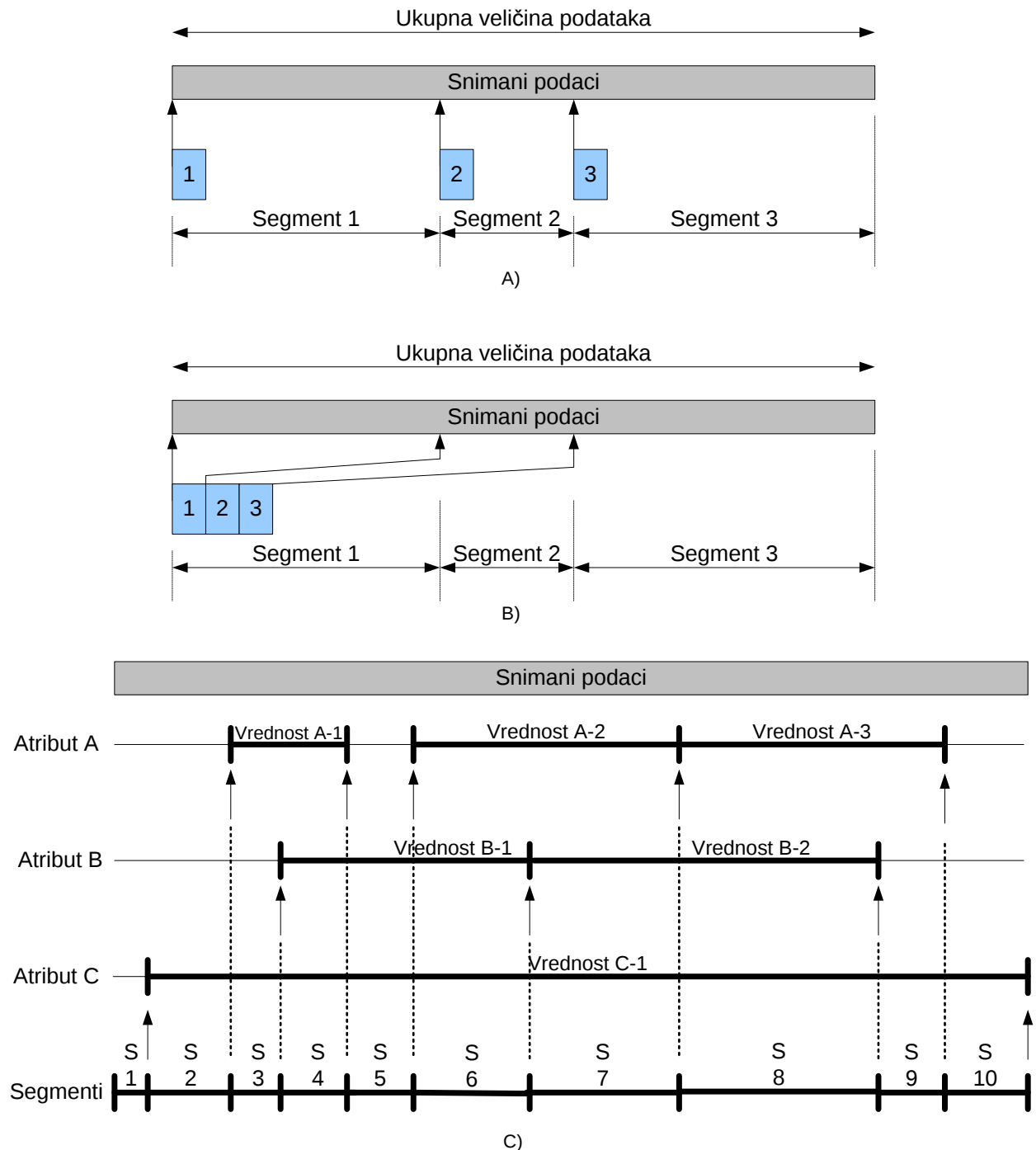
Na slici 17 su, blok dijagramom, prikazani odnosi i funkcije navedenih objekata u oba procesa PVR.



Slika 17 Odnosi i funkcije objekata

5.2 Segmentacija toka podataka pri snimanju

Tokom procesa snimanja TS na HDD, vrši se njegovao deljenje u logičke celine (segmente) pod datim kriterijumima. Segmenti su dinamičke prirode i zavise od raznih parametara toka podataka i njihovih promena tokom vremena. Npr. segment može biti definisan na osnovu veremenskog perioda, segment može predstavljati kompletan GOP, jednu sliku, audio frejm, itd. U toku snimanja konstruisaće se bar jedan segment. Zbog ove osobine, segment se može definisati i kao deo snimanog toka podataka za vreme kog se meta-podaci ne menjaju. Primeri snimanja toka podataka i podela na segmente su prikazani slikom 18.



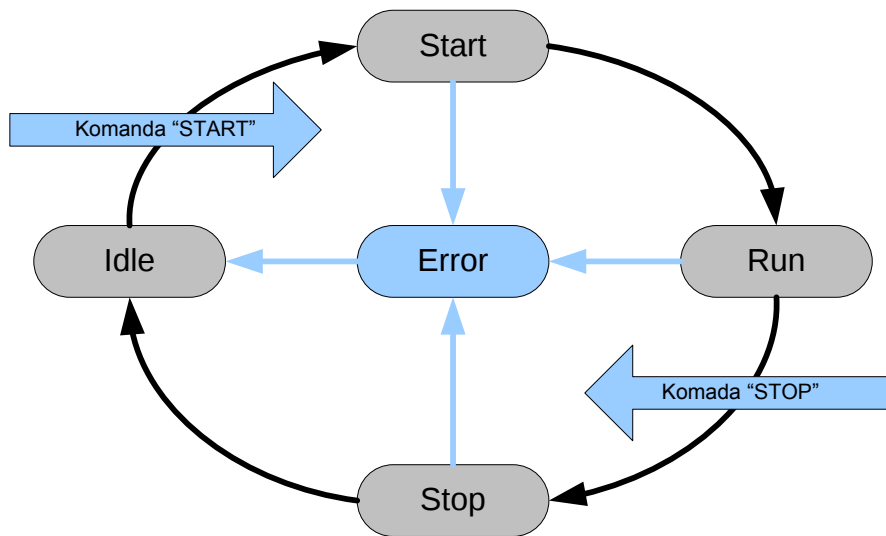
Slika 18 Snimanje i segmentacija toka podataka

5.3 Objekat Task

Objekat task je jednostvan konačni automat sa pet stanja, prikazan na slici 19. Kako bi se ponašanje svih procesa u sistemu uniformno odredilo, konačni automat stanja je implemetiran

kao poseban objekat. Svako stanje automata je realizovano preko funkcije specifične za svaki proces koji koristi Task. Pri inicijalizaciji specifičnog procesa, izvrši se inicijalizacija Task objekta tako što se svakom stanju pridruži funkcija specifična za određeni proces.

Oba programska modula, i Player i Recorder, koriste objekat task tokom svog života.



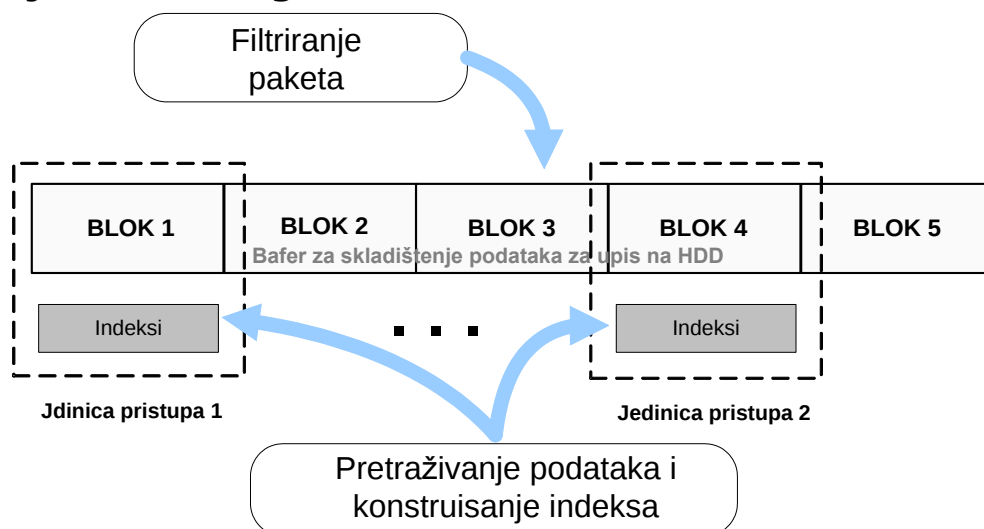
Slika 19 Automat po kom funkcioniše objekat Task

5.4 Objekat List

Tokom cirkularnog snimanja indeksni podaci se smeštaju u dvostruko spregnutu listu. Dvostruko spregnuta lista je neophodna kako bi se olakšala navigacija i pristupanje podacima za vreme reprodukcije u nazad.

Objekat *List* se koristi jedino u objektu *Indexer*, pa strukture koje predstavljaju čvorove liste nisu implementirane generički, već mogu biti isključivo tipa *TsEventList_t*. Prisutno je još jedno ograničenje i odnosi se na maksimalni broj čvorova liste. Pokazalo se da bi dinamička alokacija memorije potrebne za čvorove proizvela probleme sa fragmentacijom memorije. Zbog toga je uvedeno ograničenje na maksimalni broj čvorova liste i taj broj nije moguće promeniti za vreme izvršavanja aplikacije.

5.5 Objekat tsStorage



Slika 20 Arhitektura TsStorage objekta

Objekat TsStorage se koristi kao ulaz za proces Recorder i objekat Indexer, obezbeđujući u isto vreme i ulazni tok podataka i indeksne podatke.

Rukovanje tokom podataka koji je pripremljen za upisivanje na HDD je jedini deo PVR aplikacije koji zavisi od fizičke arhitekture. Zbog određenih razlika među MDED i MDEB čipovima, implementiran je ovaj modul kako bi sve razlike u fizičkoj arhitekturi bile pokrivene programskom podrškom i to u jednom modulu.

Asinhrono prikupljanje indeksnih informacija je izvor mogućih problema. Podaci koji se upisuju na HDD i njima odgovarajući indeksi moraju biti obradjivani i upisani u isto vreme, inače se može dogoditi da su indeksi upisani, a podaci nisu i obrnuto. Objekat *TsStorage* implementira jednosatavan mehanizam zaključavanja tako što dozvoljava pristup baferu u koji se smeštaju podaci za snimanje, samo preko blokova fiksne dužine (*access units*). Na ovaj način, bafer je realizovan kao cirkularna lista manjih blokova. Ova arhitektura je prikazana na slici 20.

Veličina blokova je izabrana tako da bude zajednički sadržalac od 188 B, što je veličina TS paketa i 512 B, što je najmanja veličina sektora na HDD, koristi se blok dužine 96256 B.

5.6 Rukovanje snimljenim TS podacima - Objekat File

Raspoloživi FS (*File System*) omogućava simultano čitanje i pisanje u jednoj datoteci, ali je zbog arhitekture PVR, koja je bazirana na dva nezavisna procesa koji istovremeno mogu biti aktivni i pristupati istoj datoteci, teško održati datoteke konzistentnim. Ako se PVR nalazi u nekom od modova za "odloženo gledanje", sinhronizacija čitanja i pisanja je od velike važnosti, kako bi se izbeglo čitanje prepisanog sadržaja iz datoteke. Problem sinhronizacije dolazi so najvećeg izražaja ako je PVR u stanju "permanentnog snimanja" jer se tada datoteka koristi kao cirkularna. Navedeni su problemi u slučaju da PVR radi u režimu odloženog gledanja:

- Kad je u toku snimanje, a reprodukcija se izvršava u nekom od trik modova, teško je razlikovati fizički i logički (pozicija pisanja) kraj datoteke.
- Ako se reprodukcija izvršava usporeno, snimanjem će se prepisati pozicija čitanja (reprodukcije) i tada će *player* čitati invalidne podatke. Sa druge strane, ako se reprodukcija izvršava ubrzano, pozicija čitanja će sustići i preći poziciju pisanja, i opet će *player* imati invalidne podatke.
- U slučaju snimanja ili reprodukcije iz cirkularne datoteke, pored problema prepisivanja, PVR mora da vodi računa o fizičkim granicama iste, i da u slučaju dostizanja granica preuzme odgovarajuće akcije, što sve zajedno proizvodi kompleksne procese čitanja i pisanja.

Da bi se rešili navedeni problemi, implementiran je poseban objekat **File**, koji nudi jedinstvenu spregu za linearne i cirkularne datoteke. U isto vreme vodi računa o sinhronizaciji čitanja i pisanja, štiteći datoteku u slučajevima simultanih pristupa.

5.7 Indeksiranje segmenata TS - Objekat Indexer

Objekat **Indexer** je odgovoran za prikupljanje indeksnih informacija o snimanom TS. Pod indeksnim informacijama se pordazumevaju početak GOP i slike bilo kog tipa, u slučaju kad DTV servis nije kriptovan. U suprotnom, ako je servis kriptovan, indeksi će ukazivati samo na početke blokova od 96256 B (94 KB) podataka, jer ostali indeksni podaci nisu dostupni pre dekriptovanja.

Indeksi će se čuvati u zavisnosti od izabranog moda snimanja. U slučaju permanentnog snimanja (cirkularno snimanje) indeksi će se čuvati u dvostruko spregnutoj listi koju obezbeđuje objekat **List**, dok će u slučaju linearnog snimanja indeksi biti sačuvani u fizičkoj datoteci, takodje korišćenjem objekta **List**.

Indeksna datoteka sadrži zaglvalje sa listom snimljenih tokova predstavljenim preko njihovih PID vrednosti.

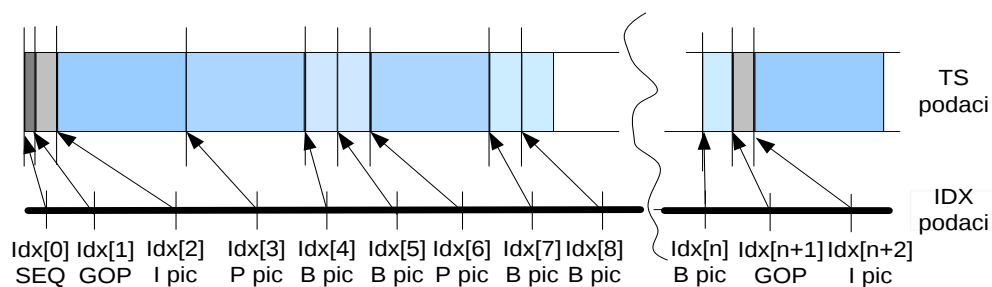
5.8 Pristup segmentima snimljenog TS - Objekat Media

Objekat *Media*, predstavlja omotač oko objekata **File** i **Index**, i skriva njihovo postojanje od korisnika i ostatka sistema.

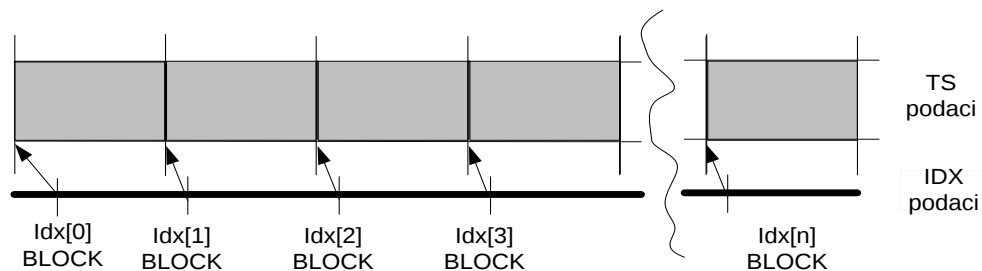
Neophodno je obezbediti jedinstvenu spregu za procese snimanja i reprodukcije (*Recording* i *Playback*) koja će ih sprečiti od simultanog pristupanja istoj datoteci, i da bi se osiguralo sigurno korišćenje svih operacija nad podacima i indeksima u *multitask* okruženju. U suprotnom, mogu da se dese sledeći problemi vezani za indeksne podatke:

- Filtrirani TS je sačuvan na HDD ali indeksi još nisu upisani. U tom trenutku *Player* može da detektuje kraj datoteke ili može pročitati invalidne, stare indeksne podatke.
- Novi indeksi su upisani, ali podaci još nisu upisani na HDD. *Player* će pročitati invalidne podatke.

I **Player** i **Recorder** koriste **Media** objekat bez direktnog pristupanja **File** i **Index** objektima, ovo pruža mogućnost da PVR procesi budu nezavisni od korišćenog FS, jer su sve razlike sakrivene unutar Media objekta. Na slici 21 je prikazan izgled jedne instance media objekta, za slučaj da su u njoj snimljeni nekriptovani podaci (slika 21A), i kriptovani podaci (21B).



(A) Struktura media datoteke za slučaj nekriptovanog toka podataka



(B) Struktura media datoteke za slučaj kriptovanog toka podataka

Slika 21 Struktura media datoteke

5.9 Programski modul Recorder

Programski modul za snimanje ulaznih podataka, **Recorder**, bi trebao da implementira parcijalno snimanje TS u datoteku i snimanje indeksnih podataka vezanih za lokaciju GOP i svake slike u okviru TS. Obe ove potrebe su zadovoljene preko **TsStorage** objekta, pa je proces Recorder jednostavan konačni automat u kojem se čeka da se u TsStorage pripreme podaci za snimanje i da se dalje ti podaci upišu u **Media** datoteku.

Postoje dva različita moda snimanja: linearno i cirkularno, ali se te razlike ne vide na nivou Recordera već na nivou Media objekta.

Koristeći ove dve sprege, Media i TsStorage, implementacija procesa Recorder je vrlo jednostavna. Nakon pokretanja procesa, prelazi se u stanje “**Start**”, inicijalizuje se TsStorage objekat i otvara nova datoteka Media tipa. Nakon inicijalizacije, proces se prevodi u stanje “**Run**”. U ovom stanju se odvija snimanje podataka na HDD, tj. čeka se dok TsStorage ne popuni bafer za snimanje najmanje do 96256 B. Izmenjeni ulazni podaci će dalje biti sačuvani na HDD korišćenjem Media objekta zajedno sa njima odgovarajućim indeksnim podacima koje je konstruisao TsStorage objekat. Nakon korisničkog zahteva (komande) za prestanak snimanja, proces prelazi u stanje “**Stop**” u kojem se zaustavlja snimanje ulaznih podataka i zatvara Media

datoteka. Sledeće je “Idle” stanje u kojem će proces ostati do ponovnog pokretanja procesa snimanja ili gašenja sistema.

5.10 Programski modul Player

Preko programskog modula **Player** transportni tok podataka koji ja snimljen na HDD se dostavlja DTP jedinici korišćenjem bafera za reprodukciju.

Pri pokretanju sistema u Player-u se izvršava inicijalizacija TSIO modula za prihvatanje podataka iz bafera za reprodukciju (*playback buffer*), inicijalizacija DTP modula za reprodukciju, inicijalizacija bafera za reprodukciju (*playback FIFO*). Nakon inicijalizacija poziva se AV menadžer da započne reprodukciju, sa PID vrednostima pročitanim iz Media datoteke.

Postoje 3 podržana moda reprodukcije:

- **Normal** - reprodukcija videa se izvršava sinhrono sa audiom
- **Pause** - player je u **Idle** stanju, ne radi ništa
- **Trick** - samo video se dekoduje i prikazuje

U svim modovima reprodukcije koristi se isti mehanizam. Podacima je moguće pristupiti samo čitanjem blokova koji su iste dužine (**access units**). Sve brzine reprodukcije i odgovarajuće podešavanja su prikazani u tabeli 2.

Identifikator	Očekivana brzina	Smer reprodukcije	Tip odsečka	Identifikator odgovarajućeg DnD moda	Mod	Br. Preskočenih Odsečaka
eBck_X60	-60	U nazad	I	DND_I_ONLY	trik	5
eBck_X12	-12	U nazad	I	DND_I_ONLY	trik	0
eBck_X4	-4	U nazad	I	DND_SLOW_1_2	trik	0
eBck_X2	-2	U nazad	I	DND_SLOW_1_4	trik	0
eBck_Step	step	U nazad	I	DND_I_ONLY	trik	-1
ePause	0	U napred	Bilo koji	DND_STOP	pauza	Bilo koji
eFwd_Step	step	U napred	I	DND_I_ONLY	trik	-1
eFwd_S4	1/4	U napred	GOP	DND_SLOW_1_4	trik	0
eFwd_S2	1/2	U napred	GOP	DND_SLOW_1_2	trik	0
eFwd_Normal	1	U napred	GOP	DND_NSPB	normalni	0
eFwd_X2	2	U napred	GOP	DND_IP_ONLY	trik	0
eFwd_X4	4	U napred	I	DND_SLOW_1_2	trik	0
eFwd_X12	12	U napred	I	DND_I_ONLY	trik	0
eFwd_X60	60	U napred	I	DND_I_ONLY	trik	5
eFwd_X120	120	U napred	I	DND_I_ONLY	trik	10

Tabela 2 Brzine reprodukcije i vrednosti odgovarajućih parametara

Reprodukciju nije moguće izvršavati bez praćenja sadržaja video bafera iz sledećih razloga:

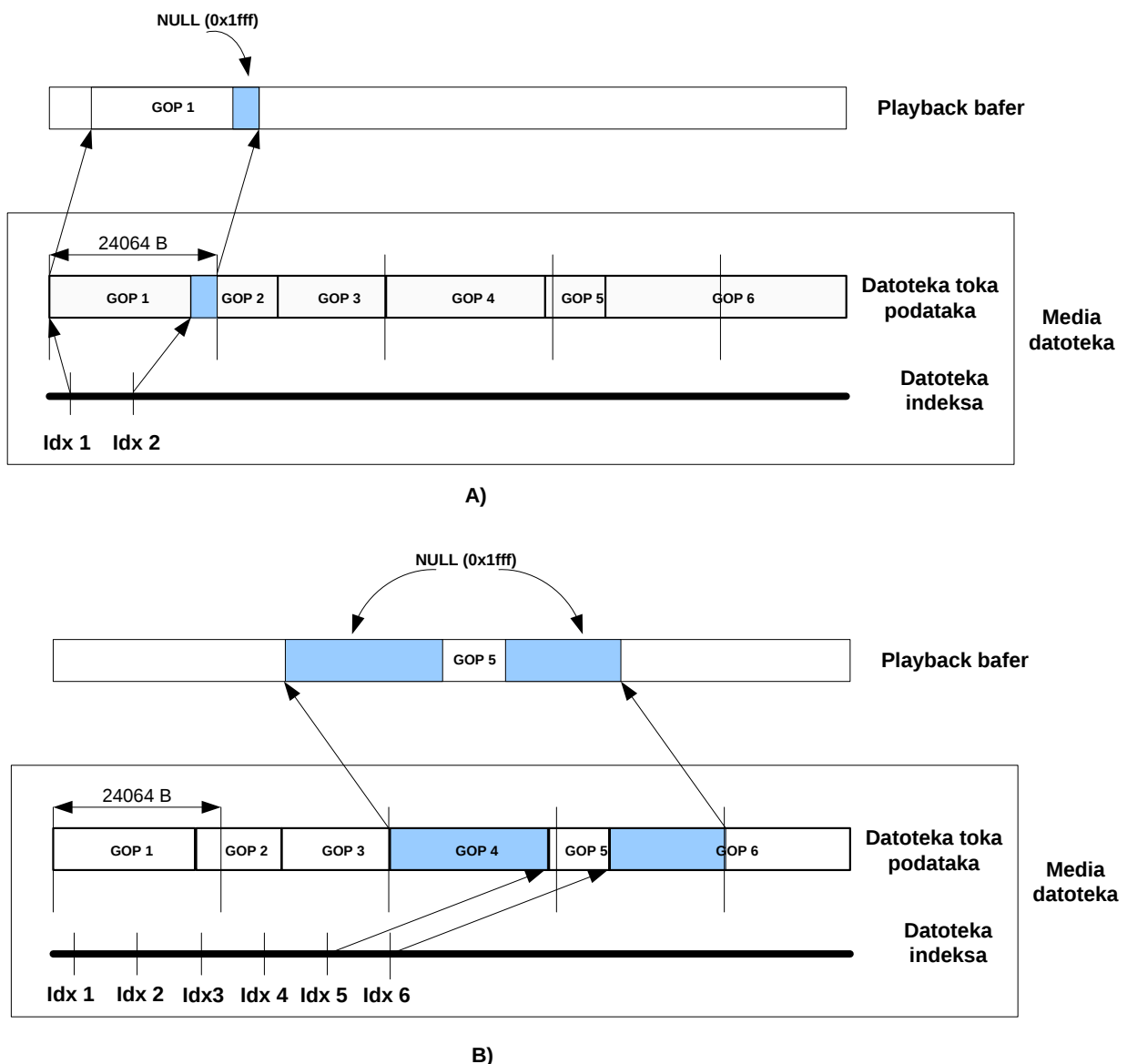
1. Video i audio dekoderi mogu da rukuju sa delimično korumpiranim slikama, pa će se kao rezultat toga proizvesti osetni prekidi u reprodukciji slike/zvuka. Ako se podaci dostavljaju previše brzo (preveliki frame-rate) JDS neće reagovati kako bi trebala i moraće uslediti prekidanje i ponovno pokretanje procesa (reset). Ovakvo ponašanje sistema nije prihvatljivo jer proizvodi treperenje i zamrzavanje slike.
2. Ukoliko nije poznat sadržaj video bafera nije moguće realizovati neprimetan prelaz sa jedne na drugu brzinu reprodukcije. Ove promene je dozvoljeno izvršiti samo na granicama GOP ili celih I slika.

Proces dekodovanja i prikazivanja zahteva validne referentne slike u svakoj situaciji. Zbog toga se reprodukcija oslanja na indeksne informacije sačuvane u objektu Indexer vodeći računa da se uvek dostavljaju kompletni GOP ili bar kompletne referentne (I) slike. Kao rezultat toga, indeksnoj datoteci se uvek pristupa preko odsečaka (**chunks**) koji predstavljaju rastojanje od početka jednog do početka drugog indeksa. Odsečak ne predstavlja samo GOP strukturu.

Implementacija mehanizma kontrole broja raspoloživih slika je neohodna kako bi se prevazišao problem sa JDS iako će se time otežati realizacija nekih od ubrzanih trik modova.

Kako bi se sprečilo nepotrebno čitanje sa HDD, u nekim od ubrzanih trik modova, onih frejmova koje će JDS odbaciti čitaju se samo I slike. Na taj način JDS se ne opterećuje nepotrebno, što osigurava njen sigurniji rad, i smanjuje se broj čitanja sa HDD.

Po prirodi MPEG videa, odsečak nije celobrojni umnožak veličine TS paketa (188 B) ni veličine pristupne jedinice HDD (512 B). Zbog toga je svaki odsečak proširen da bi se izvršilo poravnavanje i da bi se on ispravno upisao u bafer za reprodukciju. Kao posledica, svaki odsečak je celobrojni umnožak od 24064 B i sadrži invalidne podatke na početku i na kraju. Ti suvišni podaci se zamenjuju (prepisuju), nakon čitanja sa HDD, a pre upisa u bafer, sa NULL paketima (0x1fff) od strane Media objekta. Postupak zaokruživanja dužine i prepisivanje neispravnih podataka je prikazan na slici 22.



Slika 22 Zaokruživanje dužine podataka

U Player-u se vodi računa o trenutnoj poziciji reprodukcije i raspoloživom slobodnom prostoru u **playback baferu**. Takođe, prati se broj slika koje čekaju u **video baferu** (one slike koje još nisu dekodovane), kako bi se sprečili problemi sa JDS.

U Media objektu se rukuje:

1. Pozicijom čitanja iz File datoteke kao i navigacijom kroz podatke preko indeksnih informacija. Ovim se obezbeđuje da je trenutna pozicija čitanja iz datoteke nepotrebna informacija za Player.
2. Cirkularnom reprodukcijom i odloženim gledanjem.
3. Prepoznaju se različiti tipovi “**end of file**” stanja i prosledjuju se povratne vrednosti Player-u, koji u zavisnosti od te vrednosti inicira odgovarajuću akciju (tabele 3 i 4).

Tip reprodukcije	Tip Granice	Uslov	Akcija
Normalna	EndOfFile	Dostignut je fizički kraj datoteke.	Reset reprodukcije i čitanje prvog raspoloživog GOP bez menjanja brzine
Normalna	StartOfFile	Početak datoteke. Moguće jedino u slučaju reprodukcije u nazad.	Reset reprodukcije i čitanje poslednjeg raspoloživog GOP bez menjanja brzine

Tabela 3 Tipovi EoF stanja pri normalnoj reprodukciji

Tip reprodukcije	Tip Granice	Uslov	Akcija
Normalna + “odloženo gledanje”	EndOfStream	Dostignut kraj datoteke. Pozicija čitanja poklapa se sa pozicijom pisanja. Moguće je samo ako je i proces Recorder aktivan.	Reset reprodukcije i čitanje poslednjeg validnog GOP uz prebacivanje na normalni mod reprodukcije, “skoro uživo reprodukcija”.
Normalna + “odloženo gledanje”	StartOfStream	Dostignut početak datoteke. Moguće jedino u slučaju reprodukcije u nazad i ako je proces Recorder aktivan.	Reset reprodukcije i čitanje prvog validnog GOP uz prebacivanje na normalni mod reprodukcije, “maksimalno odložena reprodukcija”.
Cirkularna	EndOfStream	Dostignut logički kraj datoteka. Pozicija čitanja se poklapa sa pozicijom pisanja.	Reset reprodukcije na logički kraj (poslednji validno upisani GOP) uz prebacivanje u normalni mod, “skoro uživo”
Cirkularna	StartOfStream	Dostignut početak datoteke. Moguće jedino u slučaju reprodukcije u nazad ili pauze. Pozicija pisanja poklapa se sa pozicijom čitanja.	Reset reprodukcije na logički početak datoteke, prvi GOP na raspolaganju uz prebacivanje u normalni mod, “maksimalno odložena reprodukcija”.

Tabela 4 Tipovi EoF stanja pri trik-mod reprodukciji

Da bi se osigurali “glatki” prelazi između modova Player mora da zadovolji vrlo stroga pravila. U suprotnom postojaće vidljivi “skokovi u vremenu”, moguće treperenje slike (flickering), ili će JDS prestati da konzumira podatke iz bafera.

1. Odsečak se čita sa HDD samo ako je prethodni kompletno prebačen u playback bafer
2. Promena moda se izvršava samo kad se kompletan odsečak upiše u playback bafer. Ovim se garantuje da će TS koji se nalazi u playback baferu biti konzistentan
3. Premeštanje (transfer) odsečka je kontrolisano tako da ograniči broj raspoloživih slika u video baferu, da bi se sprečilo da JDS dodje u nedefinisano stanje
4. Promene brzina se obavljaju tek pošto je JDS konzumirala sve slike (tj. nalaze se najmanje 2 slike u video baferu zbog zahteva od strane DnD menadžera) i ovo ograničenje služi da se JDS spreči od ulaska u nedefinisano stanje

6. Opis realizacije podsistema za uslovni pristup

6.1 Objekat ScramblingCtrl

Ovaj objekat služi za kontrolu enkriptovanja i dekriptovanja i preko njega je realizovan CA podsistem. Podrška za enkripciju je imlementirana zbog potreba testiranja. Različiti sistemi za emisiju programa imaju različite metode zaštite svojih sadržaja. Kako bi se prevazišli problemi pri testiranju, proizvedeni ovakvim stanjem, uvedena je funkcija koja kriptuje sadržaj pre upisa na masovnu memoriju. Ovakvom realizacijom simuliran je model re-enkripcije za čuvanje podataka na HDD, opisan u odeljku 4.3. Enkripcije podataka je implementirana u skladu sa preporukama iz MPEG-2 standarda koje se odnose na ovu oblast.

6.2 Enkripcija

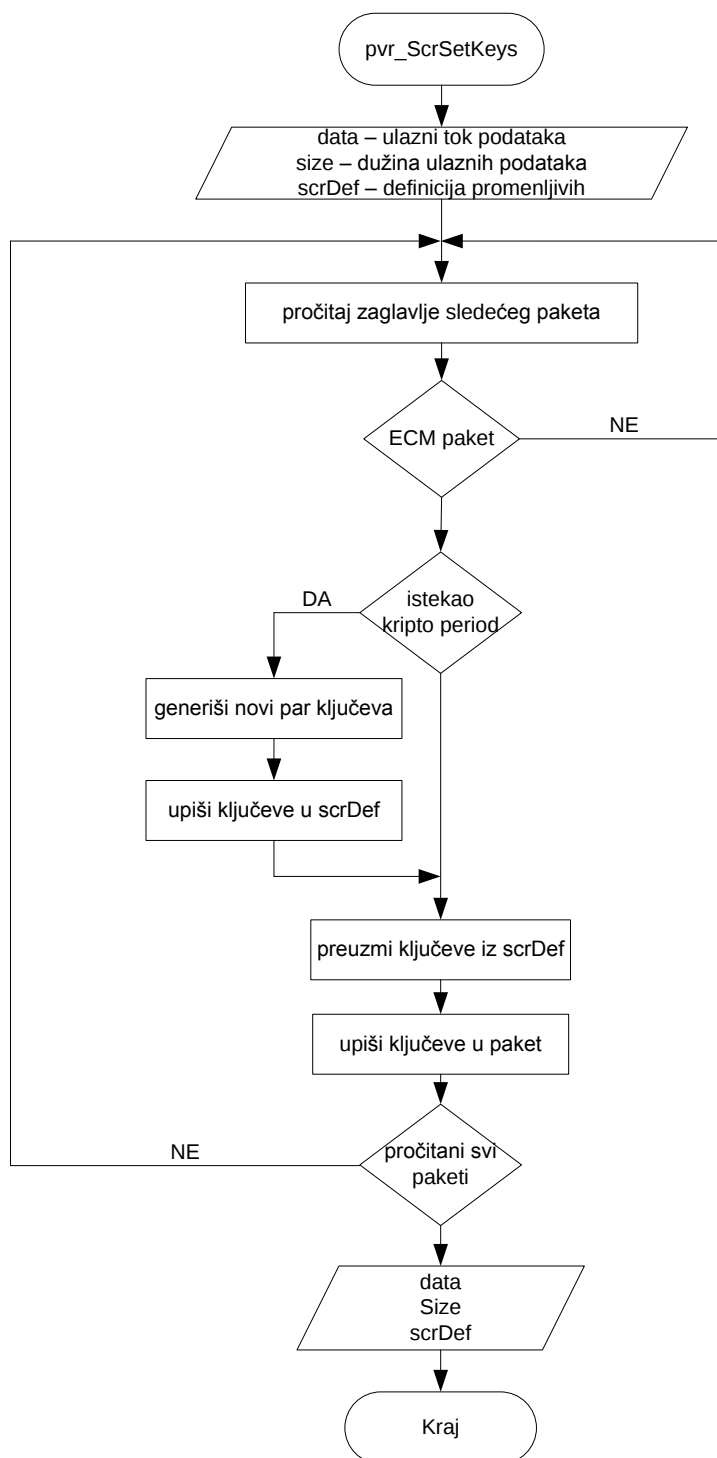
Enkripcija se izvršava neposredno pre snimanja podataka na masovnu memoriju, a u programskom modulu **Recorder**. U funkciji *pvr_ScrSetKeys*, čiji je blok dijagram algoritma prikazan na slici 23, najpre se određuju ključevi za kriptovanje, koji su u ovom slučaju neke pseudo slučajne vrednosti. Ključevi se upisuju u odgovarajuće lokalne strukture koje će se koristiti za kriptovanje podataka i u tabele koje se nalaze u toku podataka, a njihov sadržaj nije od značaja za aplikaciju. Upisivanjem ključeva u tabele simulira se postojanje ECM poruka u toku podataka. Funkciju *pvr_ScrSetKeys* poziva proces Recorder, nakon potvrde o preuzimanju podataka JPTT modulu.

Tok podataka je sada smešten u bafer iz kojeg će se upisivati na HDD. Bafer u kojem se nalaze podaci za upisivanje na HDD, se linearno čita i u zaglavlju svakog video paketa menjaju se vrednosti polja **Transport Scrambling Control**, tako da ukazuju da je on kriptovan. Ako video paket kao svoj redni broj ima paran broj, polje će dobiti vrednost '10', a ako je taj broj neparan vrednost '11'.

Nakon promene zaglavlja paketa, iz bafera se ponovo čitaju podaci koji se sada šifruju sa odgovarajućom kontrolnom rečju ili ključem, parni paketi sa parnim ključem, a neparni sa neparnim. Ključevi koji se čuvaju u strukturi koja opisuje stanje procesa, ažuriraju se svaki put kad se pročitaju tabele sa kontrolnim rečima. Kriptovanje podataka je implementirano preko funkcije *pvr_ScrScramble*. Blok dijagram algoritma ove funkcije je prikazan na slici 24.

Pošto je podrška za enkripciju imlementirana zbog potreba testiranja, nije korišten ni jedan od komercijalno definisanih sistema (CSA, DES, TDES). Kao metoda je izabrana jednostavna operacija ekskluzivno ili (**XOR**) izmedju podataka i odgovarajućeg ključa. Operacije XOR je izabrana zbog revirzibilnosti, odnosno osobine (1):

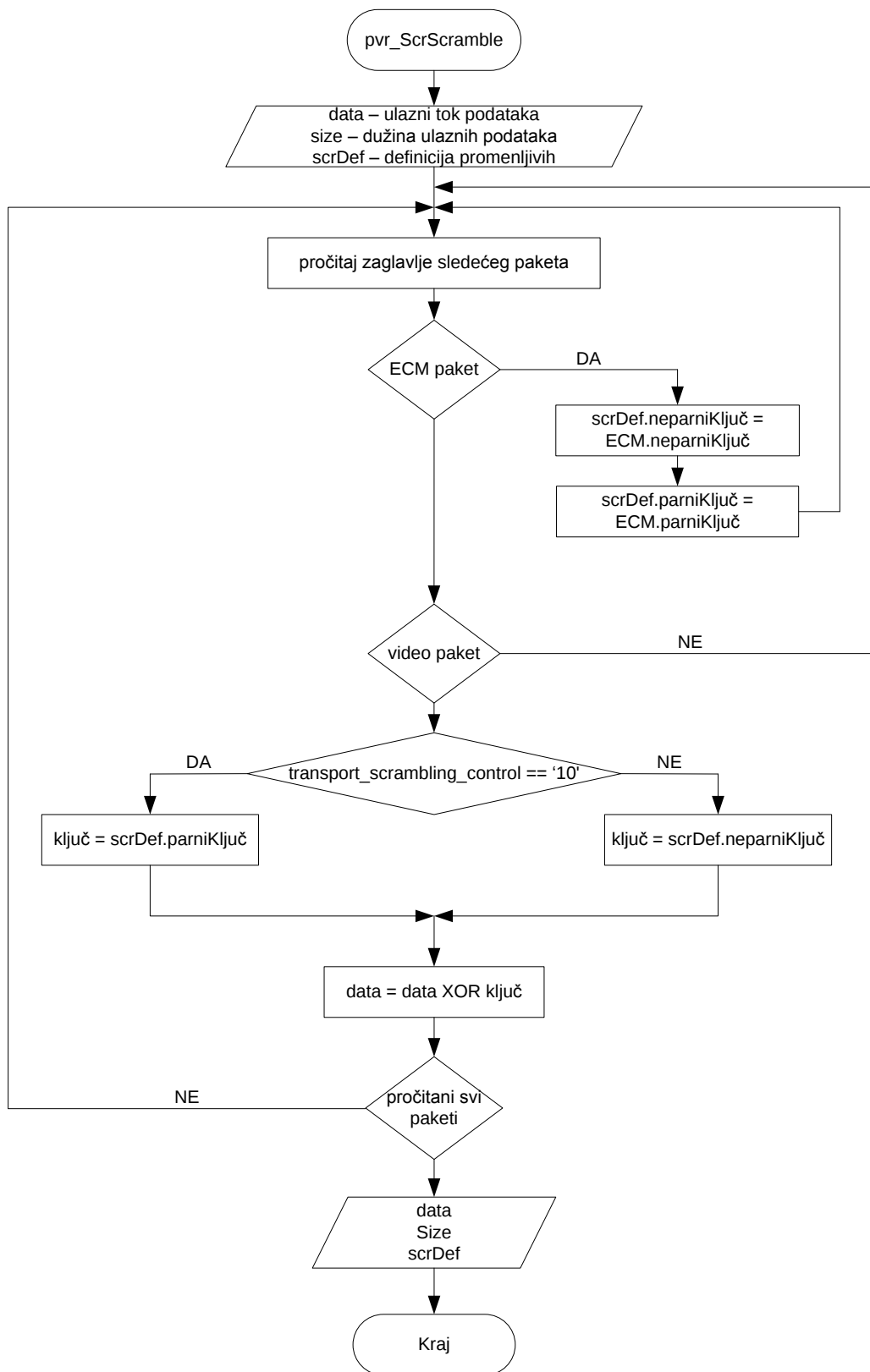
$$(A \oplus B) \oplus B = A \quad (1)$$



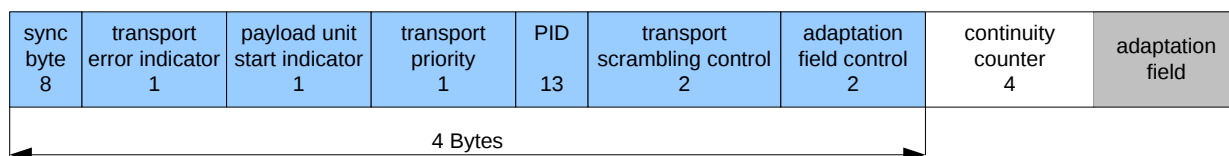
Slika 23 Blok dijagram algoritma za funkciju pvr_ScrSetKeys

Ako se dva puta primeni na iste podatke, novodobijeni podaci će biti isti kao i polazni. Kriptovanje se izvršava nad celim paketom osim nad prva 4 bajta zaglavlja. Prvih 8 bita zaglavlja predstavljaju sinhronizacionu reč, vrednost koja je rezervisana za označavanje početka TS paketa. Sledeće polje koje je bitno za realizaciju PVR aplikacije je **PID**, jedinstveni identifikacioni broj toka podataka kojem pripada dati paket. Zatim **transport scrambling control** polje koje indicira da li je paket sa kriptovanim sadržajem i na kraju **adaptation field control**, polje koje ukazuje da li u paketu postoji **adaptation field**, ili je nakon **continuity counter** polja kraj zaglavlja paketa, slika 25. Ako bi se sadržaj navedenih polja promenio na pseudo slučajan način, kao što se to radi pri enkripciji, paket bi postao invalidan.

Kriptovani podaci se dalje upisuju na HDD i postaju nečitljivi ukoliko se ne izvrši proces dekrpicije.



Slika 24 Blok dijagram algoritma za kriptovanje podataka



Slika 25 Deo zaglavlja TS paketa

6.3 Segmentacija i indeksiranje

Segmentacija i indeksiranje ulaznih podataka izvršavaju se pre upisivanja na HDD nad podacima koji su smešteni u bafer. Svaki indeks je instanca **MediaEvDef_t** strukture čija su polja prikazana u tabeli 5.

Tip podataka	Identifikator	Značenje
uint32_t	offset	Adresa iz datoteke tipa File skalirana faktorom 188, na kojoj se nalaze odgovarajući podaci.
MediaEvType_e	type	Tip indeksa. Za kriptovane datoteke moguće vrednosti su eIndexKeys i eIndexBlock .
uint32_t	value	Prazno polje, za kriptovane datoteke se ne koristi.
uint32_t	time	Vremenska odrednica. U ovo polje se upisuje vrednost sistemskog vremena u trenutku konstruisanja datog indeksa.

Tabela 5 Opis MediaEvDef_t strukture

6.3.1 Indeksiranje ECM toka

Indeksira se kompletan ECM tok, prema opisu iz odeljka 4.4.1, tj. u datoteku podataka se upisuju svi ECM podaci, a u indeksnu datoteku se upisuju svi indeksi koji ukazuju na ECM podatke. Polje **offset** je trenutna adresa upisivanja u datoteku File skalirana faktorom 188, da bi se postiglo poravnavanje sa dužinom TS paketa. Tip indeksa je **eIndexKeys**, a **time** je trenutno sistemsko vreme.

Pri indeksiranju ECM toka najpre se vrši pretraga ulaznih podataka, ukoliko se nadju paketi koji nose ovakve informacije, konstruiše se po jedan indeks na opisan način.

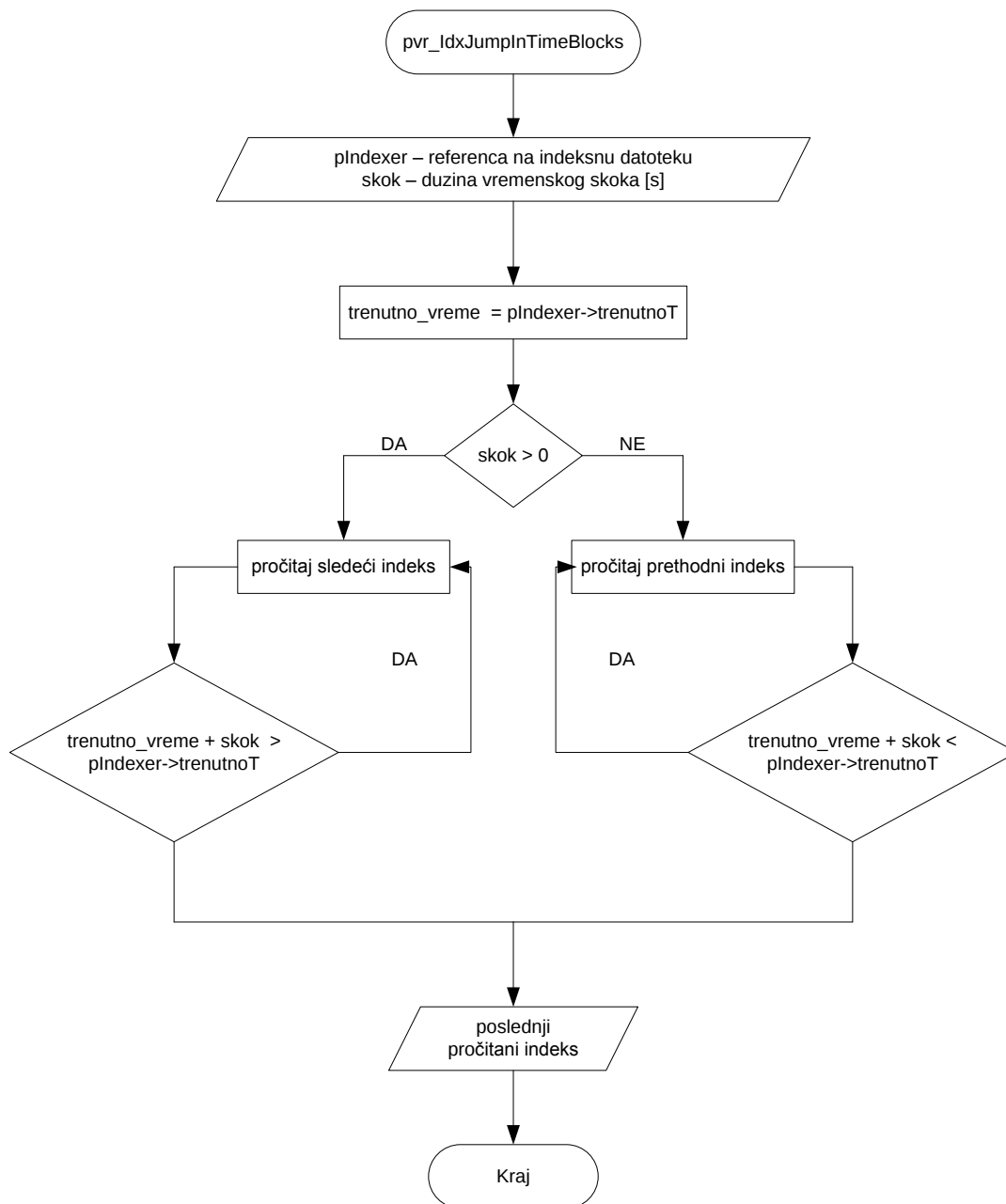
6.3.2 Indeksiranje ostalih podataka

Osim indeksa koji ukazuju na pozicije ECM paketa u datoteci, konstruiše se još jedna vrsta indeksnih podataka, oni koji ukazuju na blokove podataka dužine 96256 B. Za vreme snimanja, upisivanje na HDD se vrši u odsečcima dužine 96256 B. Pri upisivanju svakog odsečka, konstruiše se po jedan indeks. U polje **offset** će se upisati trenutna adresa upisivanja iz datoteke tipa File, skalirana sa faktorom 188. Identifikator tipa ovog indeksa je **eIndexBlock**, kao vremenska odrednica se uzima trenutno sistemsko vreme.

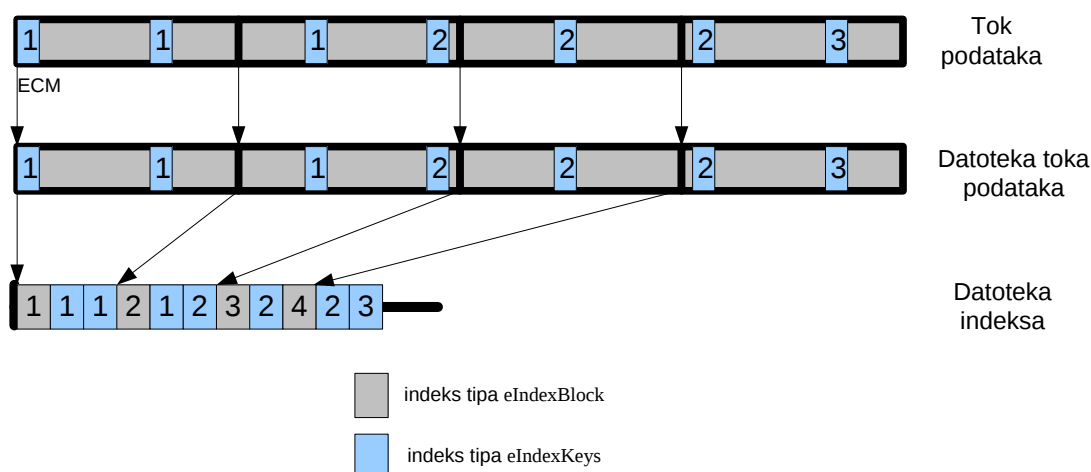
Na osnovu vrednosti polja **offset** se pri reprodukciji vrši pozicioniranje u File datoteci. Polje **time** se koristi u tri modovima reprodukcije, a za vremenske skokove. Na primer: u modu **eFwd_X60**, 60 puta ubrzana reprodukcija u napred, prikaže se jedna I slika, pa se zatim preskoči 30 sekundi (u jednoj sekundi postoje bar 2 I slike) i prikazuje se naredna I slika. Da bi se omogućilo takvo kretanje kroz indeksnu datoteku, neophodno je da indeksi imaju vremenske odrednice. Na slici 26 je blok dijagramom prikazan algoritam za vremenske skokove u datoteci sa kriptovanim sadržajem.

6.3.3 Indeksiranje

Indeksiranje podataka se izvršava u procesu Recorder. Nakon prijema 96256B ulaznih podataka, konstruiše se najpre indeks **eIndexBlock** tipa. Zatim se blok pretraži, i ukoliko se nadju ECM podaci, konstruisaće se i odgovarajući broj indeksa tipa **eIndexKeys**. Na slici 27 prikazan je primer datoteka podataka i indeksa.

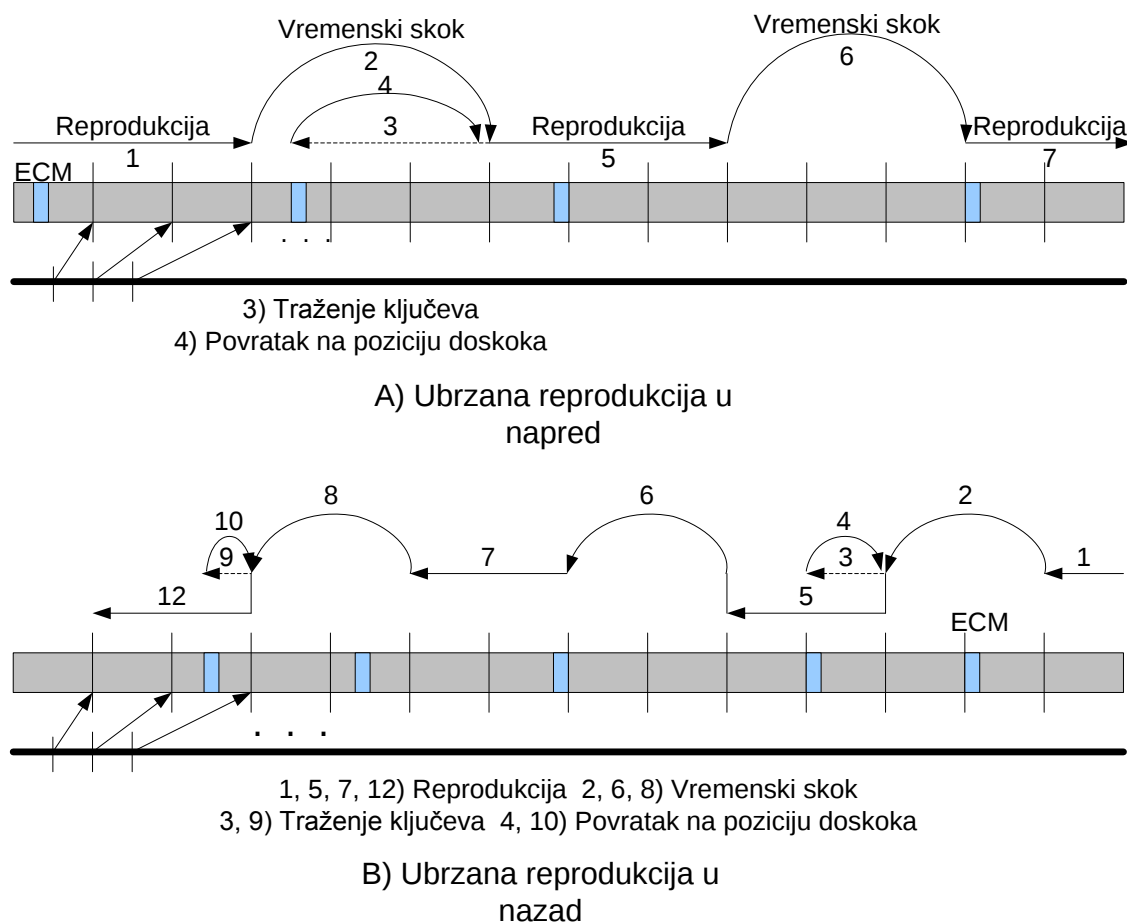


Slika 26 Blok dijagram algoritma za vremenske skokove



Slika 27 Izgled datoteka podataka i indeksa

6.4 Dekripcija



Slika 28 Čitanje i dekriptovanje podataka u trikovima

Dekripcija se izvršava nakon čitanja podataka sa HDD, a pre dekodovanja i prikazivanja. Reprodukcijska se može izvršavati normalnom brzinom (normalna reprodukcija), kao i ubrzano/usporeno u napred i ubrzano u nazad, što se naziva trikovima reprodukcije. U zavisnosti od moda reprodukcije razlikuje se i način dekriptovanja.

Ukoliko se reprodukcija izvršava normalnom brzinom, iz memorije se čita redom, bez preskakanja sekvenci. Pročitani podaci se smeštaju u bafer, koji se potom pretražuje, i u slučaju da je pronađena tabela sa ključevima, ažuriraju se polja koja čuvaju te informacije, a ako je pronađen video paket, izvršava se dekriptovanje podataka (*pvr_ScrDescramble*). Dekriptovanje je, kao i enkriptovanje, XOR operacija, sa istim ključem sa kojim je izvršeno i enkriptovanje.

Ako se *Player* nalazi u nekom od trikovnih modova rada, proces dekripcije se komplikuje. Proces reprodukcije i dekriptovanja je prikazan na slici 28. Komplikovanija dekripcija je prouzrokovana potrebom da se preskaču određene sekvence podataka, sa kojima je moguće preskočiti i tabele sa ključevima. Nakon skokova neophodno je pretražiti podatke u nazad kako bi se pronašla tabela sa odgovarajućim ključevima za dekripciju (*pvr_ScrSearchForKeys*). Po pronalazenju ključeva, proces dekripcije se odvija na isti način kao i pri normalnoj reprodukciji.

Ukoliko se pogreši pri pronalazenju ključeva i sa takvim ključevima se izvrši dekripcija, podaci će biti invalidni i neće se prikazivati ništa. Ovo je dodatna zaštita i ostvaruje se parsiranjem zaglavlja TS paketa. Ako *Adaptation Control* biti ukazuju da bi trebalo da postoji *Adaptation Field*, a pronađe se neki sadržaj koji ne odgovara očekivanom sadržaju, sistem će paket proglasiti invalidnim i odbaciti ga. Zatim, preuzeti prvi sledeći paket iz memorije i ponovo će se izvršiti proces traženja ključeva i dekripcije, ali ovaj put na način kao da je proces u normalnom režimu reprodukcije. Sistem će se ubrzo sam oporaviti od eventualne greške, jer su tabele sa ključevima relativno gusto rasporedjene u vremenu, pa će se ubrzo pronaći odgovarajući ključevi.

7. Testiranje i rezultati

Testiranje je izvršeno na platformi “LumiereD” (sa MDEB verzijom DVB dekodera) uz pomoć alata Starter. Proverena je funkcionalnost svi objekata, a posebno objekta ScramblingCtrl i utvrđena je ispravnost rada.

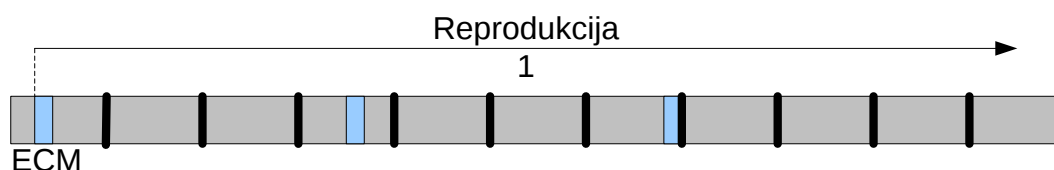
Pri testiranju su korišćeni nekriptovani servisi koji su lokalno kriptovani pre upisivanja u masovnu memoriju. *Transport Scrambling Control Bits* su prepisani tako da označavaju da se u paketu nalazi kriptovani sadržaj i izvršeno je šifrovanje sa odgovarajućim ključem. Ključevi su upisani u posebne pakete i tako izmenjeni podaci su sačuvani na HDD.

U procesu reprodukcije, pročitani podaci sa HDD se dešifruju sa odgovarajućim ključevima, *Transport Scrambling Control Bits* se ponovo postavljaju na 0 i vrši se reprodukcija.

Proces testiranja se sastojao od snimanja više raznih DTV servisa i njihove kasnije reprodukcije u 4 moda rada: **ePISpeedFwd_Normal**, **ePISpeedFwd_X2**, **ePISpeedFwd_X60** i **ePISpeedBck_X2**. Ova 4 moda rada predstavljaju 4 različita algoritma za odabir slika koje će se dekodovati i prikazivati.

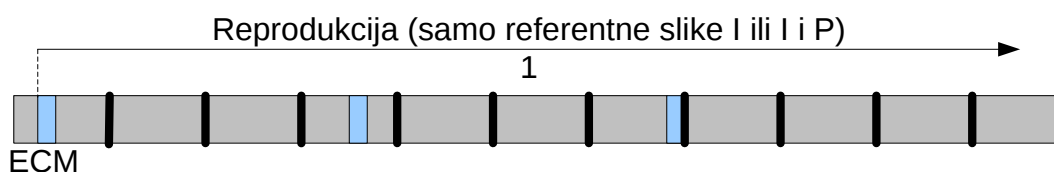
7.1 Normalna reprodukcija

ePISpeedFwd_Normal je mod normalne reprodukcije, slika 29. Blokovi se čitaju redom i dekoduju se i prikazuju sve slike, bez preskakanja.



Slika 29 *ePISpeedFwd_Normal*

7.2 2, 4 i 12 puta ubrzana reprodukcija u napred



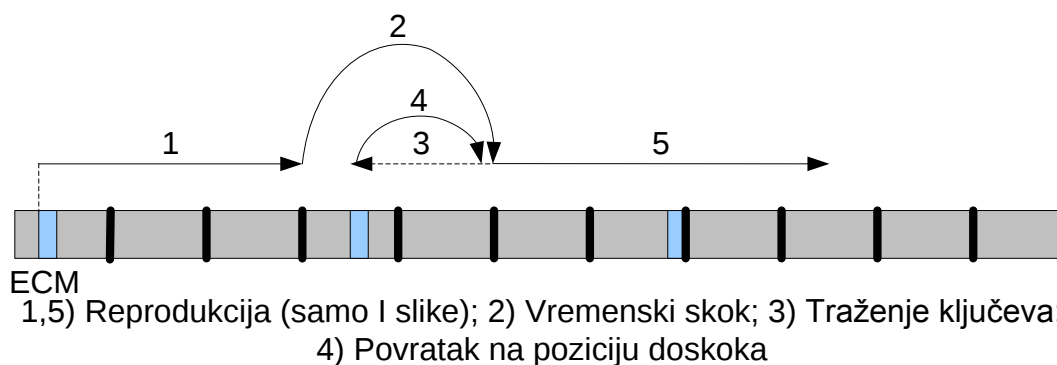
Slika 30 *ePISpeedFwd_2X*

ePISpeedFwd_X2 je 2 puta brža reprodukcija u napred od normalne. U ovom režimu rada, blokovi se čitaju redom, ali se ne dekoduju sve slike, već samo referentne, slika 30. Za 2X

se dekoduju i prikazuju i I i P, a za 4X i 12X samo I slike. Algoritam za dekodovanje i reprodukciju je isti za **ePlSpeedFwd_X2**, **ePlSpeedFwd_X4** i **ePlSpeedFwd_X12**, pa je bilo dovoljno ispitati ponašanje sistema u jednom režimu rada.

7.3 60 i 120 puta ubrzana reprodukcija u napred

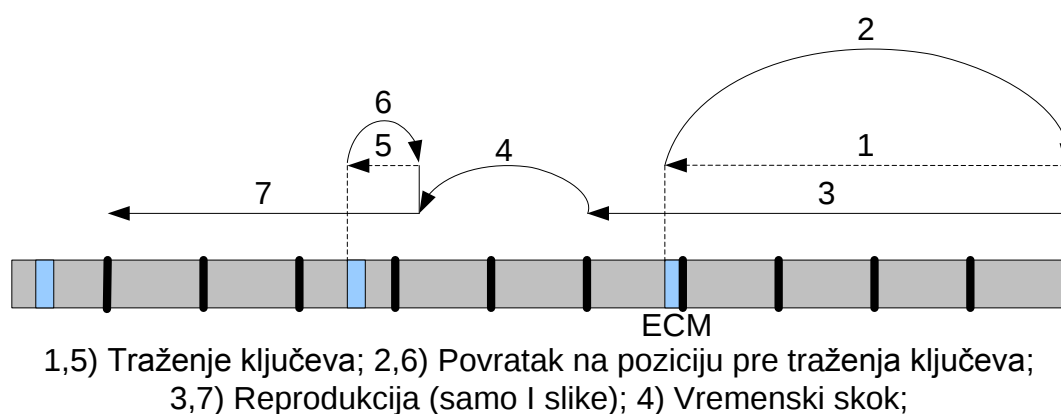
ePlSpeedFwd_X60 je 60 puta ubrzana reprodukcija u napred. Kako bi se obezbedilo potrebno ubrzanje, neophodno je preskakati odredjene sekvence. Dekoduju se i prikazuju samo I slike uz preskakanje sekvenci, slika 31. Algoritam za dekodovanje i reprodukciju u režimu **ePlSpeedFwd_X60** je isti kao i u režimu **ePlSpeedFwd_X120**, pa je dovoljno ispitati ponašanje sistema u jednom režimu rada.



Slika 31 *ePlSpeedFwd_60X*

7.4 Ubrzana reprodukcija u nazad

ePlSpeedBck_X2 je dva puta ubrzana reprodukcija u nazad. Blokovi se čitaju u nazad, sa ili bez preskakanja, u zavisnosti od brzine reprodukcije, traže se odgovarajući ključevi. Dekoduju se i reprodukuju samo referentne slike i to u obrnutom redosledu, slika 32. **ePlSpeedBck_X60**, **ePlSpeedBck_X12**, **ePlSpeedBck_X4** i **ePlSpeedBck_X2** koriste iste algoritme za dekodovanje, reprodukciju i pronalaženje ključeva. Za ispitivanje je izabrana jedna od njih.



Slika 32 *ePlSpeedBck_2X*

Testiranjem sistema u navedena 4 režima rada, ispitana je logička ispravnost svih algoritama za reprodukciju koji su implementirani u sistemu. Takodje, ispitana je stabilnost sistema pri promenama brzina reprodukcije, kao i smerova reprodukcije.

Pri svim brzinama reprodukcije za sve testirane servise, sistem se ponaša stabilno, slika je mirna (*no flickering*) i svi TS paketi su pravilno reprodukovani.

Vršeno je testiranje sa različitim periodima generisanja ECM ključeva, od 1 sekunde do 10 sekundi. U toku testiranja sistem je uspevao pravovremeno da reaguje na promene ECM sadržaja.

7.5 Rezultati merenja performansi sistema

Prikazni su rezultati merenja performansi sistema dobijeni pri testiranju za dva različita DVB servisa:

MTV - sa bitskim protokom snimljenog TS od 441431 bytes/sec

ZDF - sa bitskim protokom snimljenog TS od 720455 bytes/sec

Tabela 6 prikazuje rezultate merenja postignutih brzina i iskorišćenosti procesorskog vremena za različite modove reprodukcije. Kao test datoteka je korišćen sadržaj snimljen sa kanala MTV.

Veličina datoteke: 44566528 bytes

Bitski protok: 441431 bytes/sec

Trajanje: 101 sec

⁽¹⁾ Identifikator moda reprodukcije	Iskorišćenost CPU	Dužina datoteke		⁽²⁾ Količina podataka		⁽³⁾ Bitski protok [bytes/sec]
		msec	⁽⁴⁾ Brzina	Pročitano	Korisno	
eBck_X60	29%	1886	-54	2430464	59%	1288687
eBck_X12	29%	9448	-11	12826112	55%	1357547
eBck_X4	15%	18880	-5	12826112	55%	679349
eBck_X2	8%	37763	-3	12826112	55%	339647
eFwd_S4	3%	405215	0.25	49836544	89%	122988
eFwd_S2	4%	201844	0.5	49836544	89%	246906
eFwd_Normal	12%	100959	1	49836544	89%	493631
eFwd_X2	24%	50438	2	49836544	89%	988075
eFwd_X4	14%	18956	5	12874240	54%	679164
eFwd_X12	29%	9485	11	12874240	54%	1357326
eFwd_X60	29%	1877	54	2478592	57%	1320507
eFwd_X120	30%	1023/980 ⁽⁶⁾	99/103 ⁽⁶⁾	1251328	57%	1276865

(1) Opis modova reprodukcije se nalazi u tabeli 2

(2) Količina podataka koji su smešteni u *playback buffer*

▪ *Pročitano* Ukupno, i korisni podaci i dodatni podaci koji su uključeni zbog zaokruživanja do veličine bloka za pristup podacima. Tačana Količina podataka koji su pročitani sa HDD

▪ *Korisno* Procenat korisnih podataka koji su smešteni u *playback buffer*

(3) Bitski protok = pročitano / msec. Veličina datoteke je 44566528 bytes (bitski protok je 441431 byte/sec)

(4) Proračunati faktor skaliranja upoređeno sa normalnom brzinom

(5) Maksimalna brzina koja je postignuta

Tabela 6 Performanse sistema pri reprodukciji u raznim modovima

U tabeli 7 su prikazani rezultati merenja performansi sistema pri snimanju i reprodukciji za TS sa različitim brzinama bitskog protokoa.

Snimanje		Reprodukcija	
CPU usage	Bitski protok [bytes/sec]	Iskorišćenost CPU (Normalna reprodukcija)	Bitski protok [bytes/sec]
2%	441431	12%	441431
3%	720455	18%	720455

Tabela 7 Performanse pri snimanju i reprodukciji

8. Zaključak

PVR je realizovan kao samostalna programska podrška za digitalne televizore. Testiranje je izvršeno na platformi koja ne poseduje CA podsistem kao ni ugradjeni podsistem za dekriptovanje podataka. Umesto fizičke arhitekture koja bi obavljala dekriptovanje, projektovan je programski modul koji simulira taj proces.

Upotreba komercijalno raspoloživih CA sistema postavlja niz praktičnih problema u toku razvoja programske podrške:

- Kartice za autorizaciju su vremenski ograničene
- Mogu se koristiti samo odgovarajući TS programi, unapred pripremljeni od strane CA dobavljača
- Nije moguće menjati karakteristike CA sistema (npr. period generisanja kodnih ključeva)

Uvođenjem programskog modula za zaštitu i obradu TS podataka postignuta je veća fleksibilnost u razvoju i testiranju realizovane programske podrške.

Kao sledeći logičan korak nameće se testiranje na realnoj platformi koja poseduje fizičku arhitekturu za dekriptovanje.

8.1 Pravci daljeg razvoja

Pravci daljeg razvoja obuhvataju optimizaciju postojećeg rešenja i njegovo prilagodjenje za Linux u Windows operativne sisteme.

8.1.1 Optimizacija

Pod optimizacijom se podrazumeva optimizacija brzine rada, potrošnje memorije i iskorišćenosti pročitane memorije.

Postojeće rešenje radi u realnom vremenu na ciljnoj platformi, ali je moguća dodatna optimizacija ručnim prevodjenjem određenih sekcija u asemblerski jezik. Trenutno rešenje je testirano sa najviše 3 procesa, 2 za snimanje i 1 za reprodukciju snimljenog sadržaja. Iskorišćenost procesorskog vremena je skoro 100%, tabele 6 i 7. U slučaju uvođenja u sistem još jednog konkurentnog procesa, npr. za snimanje 3. servisa, zahtevalo bi optimizaciju brzine kako bi sistem radio u realnom vremenu.

Moguće je izvršiti optimizaciju potrošnje memorije, tj. smanjiti veličinu Indeksu datoteke. Optimizacija bi se mogla izvršiti upisivanjem indeksnih podataka samo kad se promeni njihov sadržaj. Na ovaj način bi se smanjila veličina indeksnih datoteka, ali bi se ujedno povećala dužina skokova kroz datoteku, pri traženju validnih ključeva za dekripciju.

Još jedna moguća optimizacija se odnosi na iskorišćenost pročitanih podataka. Ograničenje koje je postavljeno od strane platforme a odnosi se na čitanje sa HDD samo u odsečcima od 512 B, uz ograničenje od strane Indexer-a, da se čitaju samo celi blokovi (188 B), zajedno proizvode

ograničenje da blok za čitanje sa HDD mora biti celobrojni sadržalac od 512 i 188. Najmanji zajednički sadržalac od 512 i 188 je 24064. Zbog zahvatanja sa HDD ove količine podataka, dešava se da se među pročitanim podacima nalazi velika količina nepotrebnih podataka, što se vidi u tabeli 6. Ako se omogući rad aplikacije na drugim platformama, trebalo bi prilagoditi postojeće rešenje da se ukine fiksno ograničenje na pristup podacima.

8.1.2 Prenosivost

Postojeće rešenje je vezano za fizičku arhitekturu. Njegovim prilagodjenjem za Linux i Windows operativne sisteme, programska podrška bi postala potpuno nezavisna od korišćene fizičke arhitekture i time prenosiva sa jedne na drugu platformu.

Potrebno je nastaviti rad na razvoju platformski nezavisne programske podrške, koja obuhvata:

- sistemske pozive zavisne od operativnog sistema u koje spada rukovanje procesima, tajmerima, kritičnim sekcijama i slično
- Unapredjenje abstrakcije modula fizičke arhitekture (HW uređaja), kao što su video dekodir i jedinica za obradu ulaznog toka podataka (TS)

9. Literatura

- [1] Information Technology – “*Generic Coding of Moving Pictures and Associated Audio*”, recommendation H.222.0, ISO-IEC 13818-1, Darft International Standard, 10. Jun 1994.
- [2] Technical Report, *Mictos based PVR implementation*, Februar 2005.
- [3] Chris Mlynarczyk, *MPEG-2 – The basics of how it works (MPEG tutorial 1)*
- [4] A Guide to MPEG Fundamentals and Protocol Analysis (25W_11418_3),
<http://www.tektronix.com/>
- [5] Literatura za predavanja iz predmeta Programska podrška u televiziji i obradi slike, April 2005
- [6] Technical White Paper, *OpenTV PVR 2.0*, OpenTV Inc, 257 Sacramento Street, San Francisco CA 94111
- [7] Wikipedia, the free encyclopedia <http://www.wikipedia.org/>
- [8] "Jedno rešenje implementacije digitalnog ekvilajzera na jednoj klasi namenskih procesora za obradu video signala", Tereza Kovač, Siniša Grgić, Goran Miljković, Velibor Mihić, Nikola Teslić, Miodrag Temerinac (ETRAN 2006)
- [9] "Jedno rešenje programske podrške za snimanje i reprodukciju zaštićenog DVB sadržaja", Tereza Kovač, Goran Miljković, Velibor Mihić (TELFOR 2006)

JEDNO REŠENJE IMPLEMENTACIJE DIGITALNOG EKVILAJZERA NA JEDNOJ KLASI NAMENSKIH PROCESORA ZA OBRADU VIDEO SIGNALA

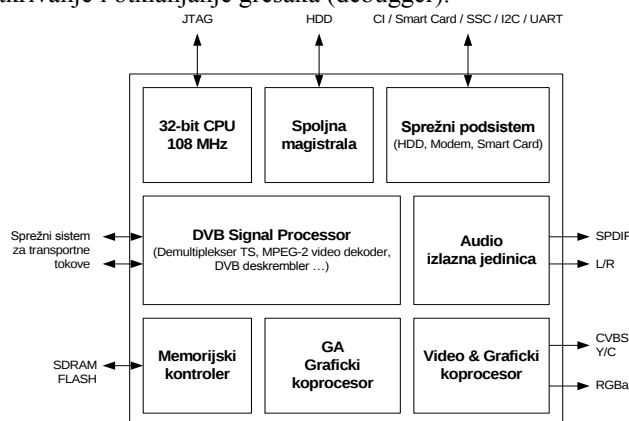
Tereza Kovač, Nikola Teslić, Miodrag Temerinac Fakultet tehničkih nauka u Novom Sadu
Siniša Grgić, Goran Miljković, Velibor Mihić MicronasNIT Novi Sad

Sadržaj – Prikazano je jedno rešenje implementacije i optimizacije digitalnog ekvilajzera na jednoj klasi namenskih procesora za obradu video signala. Date su osnove karakteristike platforme, kao i kratak opis realizovanog digitalnog ekvilajzera. Na kraju su prikazani rezultati optimizacije brzine računanja koeficijenata IIR (IIR - Infinite Impulse Response) filtra.

1. UVOD

U ovom radu je prikazano jedno rešenje implementacije i optimizacije digitalnog ekvilajzera na jednoj klasi namenskih procesora za obradu video signala (u daljem tekstu namenska platforma). Ekvilajzer je realizovan kao jedna vrsta IIR filtra koji pojačava i/ili smanjuje visoke i/ili niske frekvencije, u zavisnosti od položaja klizača, izlaznog signala u opsegu od -12dB do +12dB.

Namenska platforma zadovoljava sve zahteve za digitalne televizore, ima TS ulaze i izlaze, analogne RGB, CVBS, Y/C izlaze, digitalne audio izlaze S/PDIF i I2C, SmartCard spregu i slično. Osnovne karakteristike su: takt procesora 108 MHz, ugrađena SDRAM memorija (2x256 MBit), ugrađena Flash memorija 64Mbit kao i JTAG priključak za povezivanje programske podrške za otkrivanje i otklanjanje grešaka (debugger).



Slika 33 Shema namenske platforme

2. IMPLEMENTACIJA I OPTIMIZACIJA

Digitalni ekvilajzer se sastoji od dva bloka, IIR filtra i bloka za računanje koeficijenata tog filtra. Blok za računanje koeficijenata koristi prosledene vrednosti klizača u decibelima, na osnovu kojih, u iterativnom postupku određuje koeficijente filtra. Koeficijente filtra preuzima audio jedinica i od tog trenutka pa do sledeće promene izlazni signal (16 bitni PCM) se menja prema zadatim parametrima.

Broj instrukcija potrebnih za IIR filtar je $2L+1$ MAC instrukcija po odbirku, gde je L red filtra. Kao primer za red filtra $L=2$, i $f_s=48.000\text{Hz}$, gde je f_s -frekvencija odabiranja, dobija se 480.000 MAC instrukcija u sekundi. Ako se pretpostavi da je za jednu MAC instrukciju procesoru potrebno 4 ciklusa + 10 ciklusa za čitanje i pripremu operanada, dolazi se do cifre od 6,72 MIPS, što

predstavlja idealan slučaj. Nakon prvobitnog rešenja, bilo je potrebno optimizovati IIR filtar kako bi ekvilajzer radio u realnom vremenu.

2.1. ARITMETIKA

Rutine za računanje koeficijenata su u prvobitnoj verziji (u daljem tekstu referentna verzija) bile urađene u aritmetici pokretnog zareza sa dvostrukom tačnošću (double). Prelaskom na aritmetiku pokretnog zareza jednostruke tačnosti (float) dobili su se identični rezultati kao i u aritmetici sa pokretnim zarezom dvostruke preciznosti i u manjoj meri se povećala brzina računanja.

Nakon aritmetike pokretnog zareza, ispitane su vrednosti međurezultata u procesu računanja koeficijenata, sa ciljem prelaska na aritmetiku fiksnog zareza, tj. celobrojnu aritmetiku. Apsolutne vrednosti međurezultata pojedinih promenljivih mogu biti manje od 1, a takodje i mnogo veće od 4096. Da bi računanje koeficijenata bilo potpuno tačno, potrebno je 26 bita za decimalni deo, i bar 20 za celobrojni deo sa znakom. Iz ove računice se jasno vidi da celobrojna aritmetika nije mogla biti realizovana sa 32-bitnim celobrojnima vrednostima.

Zbog toga se prešlo na realizaciju algoritma u celobrojnoj aritmetici 64-bitnim vrednostima. Nakon optimizacije, dobijeni su identični rezultati, tj. koeficijenti dobijeni računanjem u aritmetici pokretnog zareza su identični koeficijentima dobijenim računanjem u celobrojnoj aritmetici.

2.2. MATEMATIČKE FUNKCIJE

Značajan pomak je napravljen kada su izostavljeni pozivi vremenski zahtevnih matematičkih funkcija: sin, cos, pow, sqrt i atan. Pošto je u algoritmu potrebno poznavati vrednosti sin i cos funkcija u konstantnih 20 tačaka, te vrednosti su smeštene u tabele. Ilustracije radi, u jednoj iteraciji izvršavalo se 600 poziva funkcija sinus i kosinus, umesto toga se 600 puta čita iz memorije. Ako se uzme u obzir da računanje koeficijenata može da traje do 500 iteracija, dolazi se do cifre od 300000 poziva zahtevnih matematičkih funkcija, iz čega se jasno vidi da je ušteda vremena vrlo značajna. U funkciji je pored ovih 600 poziva po iteraciji bilo još 40 poziva kosinusa po iteraciji koji ne mogu da se izostave, jer se izračunavaju kosinusi od međurezultata koje nije moguće predvideti, a samim tim nije moguće ni staviti ih u memoriju kao tabele. Zbog toga je implementirana funkcija koja u celobrojnoj aritmetici računa vrednost kosinusa i to na osnovu razvoja funkcije u Tejlorov (Taylor) red (1).

$$\cos x = \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} \quad (1)$$

Zatim su potpuno izostavljeni pozivi funkcije stepenovanja (pow), jer je domen mali skup konstantnih vrednosti. Izračunate vrednosti su smeštene u tabelu.

Takodje stepenovanje je korišćeno za popunjavanje vrednosti niza $F(i)$, koje su potrebne u toku računanja

koeficijena filtra. Vrednosti niza $F(i)$ računaju se iz izraza (2)

$$H(i) = \frac{1}{2} \left(H(i-1) + \frac{1}{H(i-1)} \right) \quad (2)$$

gde su $H(i)$ vrednosti klizača u i -tim tačkama profila, tj. celobrojne vrednosti u opsegu $[-12, 12]$ dB. Dakle, sve potrebne vrednosti su rasporedjene u dve tabele i izostavljena je matematička operacija stepenovanja.

Osim ovih funkcija algoritam je koristio i funkcije kvadratnog korena i arkus tangensa. Pozivi ovih funkcija se nisu mogli izostaviti jer je domen ovih funkcija veliki podskup realnih brojeva i nije moguće smestiti vrednosti u tabelu. Stoga su implementirane rutine za računanje kvadratnog korena i arkus tangensa.

Arkus tangens funkcija je realizovana na osnovu kordik algoritma (CORDIC - **C**Oordinate **R**otation **D**igital **C**omputer). U suštini, to je iterativni metod kojim se rotira vektor za proizvoljan ugao koristeći samo operacije sabiranja i pomeranja. Zbog toga što nam u računanju nije potrebno korišćenje zahtevnih matematičkih operacija, kao što su deljenje ili stepenovanje, algoritam je vrlo efikasan i brz. Za računanje izlaza potrebno je svega $2n+1$ pomeranja, $3n$ sabiranja i 1 množenje, gde je n broj iteracija u kordik algoritmu. Koristeći ovaj algoritam mogu se izračunati ili aproksimirati sve trigonometrijske funkcije.

Za ispitivanje položaje klizača, bilo je dovoljno 16 iteracija da se dobiju koeficijenti filtra koji su identični sa koeficijentima iz referentnog rešenja u aritmetici pokretnog zarez. Nemoguće je ispitati tačnost za sve kombinacije položaja klizača, jer ih ima 25^{10} , pa je radi osiguravanja tačnosti, broj iteracija povećan na 20.

U tabeli 1 su prikazane približne veličine potrebnog vremena za računanje koeficijenata pre i posle optimizacije:

Vreme računanja koeficijenata za sve vrednosti klizača +12:	
Pre optimizacije - dvostruka tačnost	~ 36 s
Pokretni zarez jednostruke tačnosti	~ 4 s
Celobrojna aritmetika	1,523 s

Tabela 8 Rezultati optimizacije brzine računanja koeficijenata

2.3. MEMORIJA

Pored optimizacije brzine, prelaskom sa aritmetike dvostruke tačnosti na aritmetiku jednostruke tačnosti, smanjila se i memorija potrebna za podatke.

Memorija za podatke	
Pre optimizacije - dvostruka tačnost	~ 18 KB
Pokretni zarez jednostruke tačnosti	~ 2 KB
Celobrojna aritmetika	~ 3 KB *

Tabela 9 Rezultati optimizacije veličine memorije

* - veličina memorije se povećala zbog toga što su vrednosti funkcija unesene u tabele, a izostavljeni su pozivi zahtevnih matematičkih funkcija

Zbirni rezultati optimizacije brzine računanja i veličine memorije su dati u tabeli 3. Razvoj IIR filtra je

započet u programskom okruženju Microsoft Visual C++ 6.00, kao rutine koja koristi aritmetiku pokretnog zarez, radi jednostavnosti implementacije i dobijanja referentnog koda. Nakon toga filter je preveden u aritmetiku fiksnog zarez, čiji su koeficijenti kao i međurezultati i izlazi celobrojne vrednosti. Ovim je ubrzan rad filtra na ciljnoj platformi, jer platforma u suštini samo emulira aritmetiku pokretnog zarez.

Prelaskom sa aritmetike pokretnog zarez na aritmetiku fiksnog zarez, unesena je greška na izlaznom signalu (16 bitni PCM) u rasponu od -3 do $+4$ po apsolutnoj vrednosti izlaznog signala. Optimizovani filter, napisan u C jeziku, na ciljnoj platformi zahteva oko 25% procesorskog vremena.

Nakon optimizacije u C jeziku, filter je preveden na asemblerski jezik i izvršena je njegova optimizacija. Optimizacijom je postignuto smanjenje potrebnog procesorskog vremena za ~15% čime je ukupno potrebno procesorsko vreme smanjeno na željenih 10%. Rezultati optimizacije su prikazani u tabeli 3.

	Pre optimizacije	U aritmetici pokretnog zarez	U aritmetici fiksnog zarez
Broj poziva sin i cos funkcija po iteraciji (sin, cos)	640	40	0*
Maksimalan broj poziva sin i cos funkcija u jednom ciklusu računanja koeficijenata	320 000	20 000	0*
Broj poziva funkcije stepenovanja po iteraciji (pow)	14	0	0
Maksimalan broj poziva funkcije stepenovanja u jednom ciklusu računanja koeficijenata	7 000	0	0
Broj poziva funkcije korenovanja po iteraciji (sqrt)	101	101	0*
Maksimalan broj poziva funkcije korenovanja u jednom ciklusu računanja koeficijenata	50 500	50 500	0*
Broj poziva funkcije arkus tangens po iteraciji (atan2)	60	60	0*
Maksimalan broj poziva funkcije arkus tangens u jednom ciklusu računanja koeficijenata	30 000	30 000	0*
Vreme računanja koeficijenata za vrednosti klizača 12 dB	~ 36 s	~ 4 s	1.523s
Veličina memorije podataka	~18 KB	~ 2 KB	~ 3 KB

Tabela 10 Zbirni rezultati optimizacije

*poziv funkcije nije potpuno izostavljen, već se poziva optimizovana rutina u celobrojnoj aritmetici

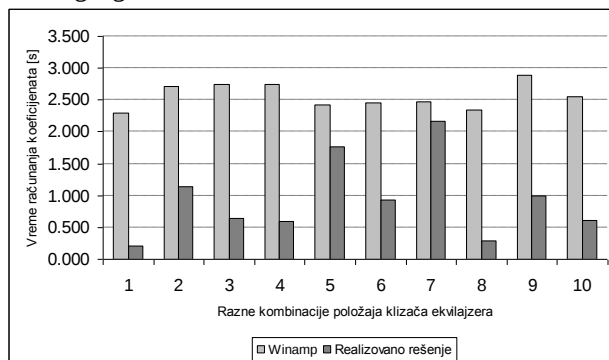
3. ISPITIVANJE I REZULTATI

Ispitivanje rutina za računanje koeficijenata i IIR filtra realizovanih u aritmetici fiksnog zareza je najpre izvedeno u okruženju Microsoft Visual C++. Poredjeni su rezultati, tj. koeficijenti filtra iz referentnog rešenja, i iz realizovanog rešenja. Nakon što je ustanovljeno da rutine računaju ispravne koeficijente i filter pojačava/utišava izlazni signal, prešlo se na ispitivanje na namenskoj platformi.

Na namenskoj platformi ispitivanje se sastojalo od merenja vremena potrebnog za izračunavanje koeficijenata filtra za razne vrednosti klizača. Vreme je mereno, tj. računato na sledeći način: nakon podešavanja željenih vrednosti klizača, korisnik je, preko grafičke sprege, komandovao početak računanja, u tom trenutku je preuzeto sistemsko vreme, i na kraju računanja je takodje preuzeto sistemsko vreme. Razlika ove dve vrednosti predstavlja ukupno vreme računanja koeficijenata filtra. Minimalno vreme računanja, dobijeno za vreme ispitivanja je 0,164 sekunde, dok je maksimalno 2,161 sekunde.

Za uporedno ispitivanje je kao referentno rešenje korišćena aplikacija Nullsoft Winamp 5.08e. Za potrebe merenja vremena računanja koeficijenata referentne aplikacije, napisana je DLL biblioteka (DLL - dinamic link library). Ovom bibliotekom je mereno vreme koje

protekle od trenutka kada se pomeri poslednji klizač do trenutka kada se promene jačine različitih frekvencija izlaznog signala.



Grafikon 1 Uporedni prikaz vremena računanja koeficijenata

U tabeli 4 i na grafikonu 1 su dati uporedni rezultati ispitivanja realizovanog i referentnog rešenja, za 10 različitih položaja klizača. Uporedno merenje i ispitivanje je izvršeno za 40 različitih položaja klizača, od toga je uzeto 10 nasumičnih slučajeva i prikazano u tabeli. U tih 10 položaja su uključena i 3 unapred definisana položaja iz referentnog rešenja.

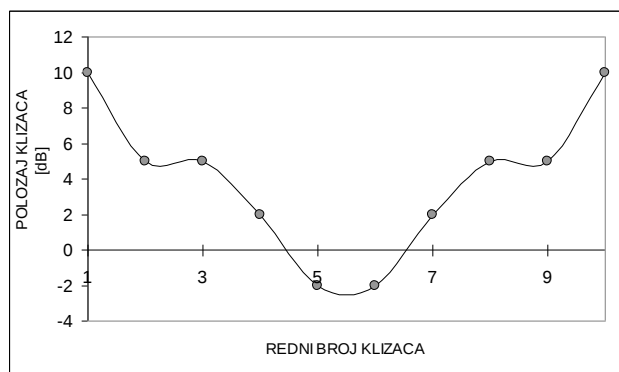
no.	Vrednosti klizača po tačkama [dB]										Vreme računanja Winamp [sec]	Vreme računanja Realizovano Rešenje [sec]	Unapred definisani položaji
	0	1	2	3	4	5	6	7	8	9			
1	-12	-12	-12	-12	-12	-12	-12	-12	-12	-12	2,300	0,209	All Low
2	0	0	0	-12	-12	-12	-12	-12	-12	-12	2,720	1,147	Mid Low Low
3	-12	-12	-12	-12	-12	-12	2	2	2	2	2,750	0,641	High Mid Low
4	3	8	1	-1	7	0	8	2	9	2	2,740	0,598	Random
5	1	0	3	7	11	11	9	5	1	-1	2,430	1,767	Random
6	12	12	12	12	12	12	12	12	12	12	2,450	0,927	All High
7	4	4	4	-12	-12	-12	4	4	4	4	2,480	2,161	Mid Low Mid
8	3	2	-2	-4	0	3	4	3	2	2	2,340	0,285	Techno
9	7	7	7	4	1	-2	-7	-8	-7	-7	2,890	0,992	Full bass
10	-7	-7	-4	-1	2	6	10	12	12	12	2,550	0,610	Full treble

Tabela 11 Uporedni prikaz vremena računanja referentnog i realizovanog rešenja

4. ZAKLJUČAK

Početna verzija IIR filtra koja je realizovana u aritmetici pokretnog zareza ne može da radi u realnom vremenu, tj. Potrebno je oko **120%** procesorskog vremena. Nakon što je filter uspešno realizovan u aritmetici fiksnog zareza njegova potrošnja procesorskog vremena se smanjila na oko **30%**. Posle optimizacije na nivou C jezika potreba je smanjena na oko **25%**, a nakon prvog koraka optimizacije u assembleru došlo se do cifre od oko **10%** procesorskog vremena.

Funkcija koja izračunava koeficijente filtra je takodje uspešno realizovana na namenskoj platformi. Računanje koeficijenata traje ispod 3 sekunde za sve ispitivane kombinacije položaja klizača. Kao pokazatelj rezultata optimizacije je uzeta podrazumevana (default) opcija položaja klizača, grafikon 2.



Grafikon 2 Podrazumevana (default) opcija digitalnog ekvilajzera

Za početnu verziju rutine za računanje koeficijenata bilo je potrebno **oko 60 sekundi**. Nakon prvog koraka optimizacije (prelazak na aritmetiku pokretnog zareza

jednostruke tačnosti) vreme računanja je spalo na oko **12 sekundi**. Posle drugog koraka optimizacije (matematičke funkcije) vreme računanja je smanjeno na oko **3 sekunde**. Poslednja verzija ekvilajzera, u kojoj je implementiran filter napisan na assembleru, zahteva **ispod 1 sekunde** za računanje početne (default) opcije položaja klizača.

LITERATURA

[1] Stevan Berber i Miodrag Temerinac, "Osnovi algoritama i struktura DSP", Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka, Novi Sad 2004.

[2] Vladimir Kovačević, Miroslav Popović, "Sistemska programska podrška u realnom vremenu", Univerzitet u Novom Sadu, Fakultet Tehničkih Nauka.

[3] http://en.wikipedia.org/wiki/Taylor_series

[4] <http://www.fpga-guru.com/cordic.htm>

[5] <http://www.fpga-guru.com/files/crdcsrvy.pdf>

Abstract – This paper presents one solution of digital equalizer for one class of video signal processors. Characteristics of platform, as well as short overview of digital equalizer are given. At the end, optimization process and results are depicted.

ONE SOLUTION OF DIGITAL EQUALIZER FOR ONE CLASS OF VIDEO SIGNAL PROCESSORS

Tereza Kovač, Siniša Grgić, Goran Miljković, Velibor Mihić, Nikola Teslić, Miodrag Temerinac

Jedno rešenje programske podrške za snimanje i reprodukciju zaštićenog DVB sadržaja

Tereza Kovač, Goran Miljković, Velibor Mihić

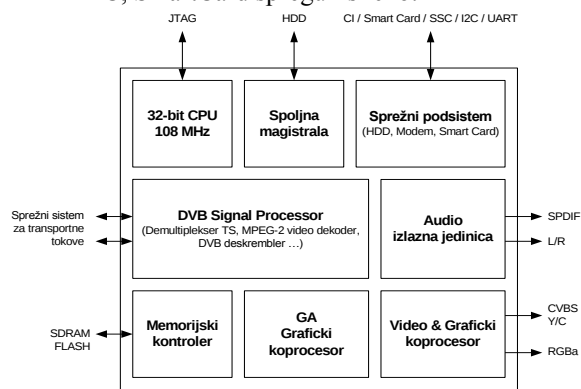
Sadržaj — Prikazano je jedno rešenje implementacije PVR sa podrškom za snimanje i reprodukciju kriptovanih digitalnih TV servisa na MDE9500B baziranoj platformi. Date su osnove karakteristike platforme, kao i kratak opis realizovanog sistema. Na kraju su prikazani rezultati testiranja.

Gljučne reči — CA, CAT, DRM, DTV, DVR, ECM, EMM, FS, GOP, HDD, MPEG, PID, PVR, TS.

I. UVOD

U ovom radu je prikazano jedno rešenje implementacije PVR sa podrškom za snimanje i reprodukciju kriptovanih DTV servisa na MDE9500B baziranoj platformi (u daljem tekstu namenska platforma).

Namenska platforma zadovoljava sve zahteve za digitalne televizore; podržava višestruke TS ulaze i izlaze, analogne RGB, CVBS, Y/C izlaze, digitalne audio izlaze S/PDIF i I2C, SmartCard spregu i slično.



Sl. 1. Shema namenske platforme

II. ANALIZA PROBLEMA

A) Reprodukcijsa sačuvanog TS

Osnovna prednost TS snimljenog na HDD jeste da se za vreme reprodukcije podacima može pristupiti na slučajan način (*random access*), što nije moguće u toku direktne reprodukcije (*live stream*). Pri reprodukciji je neophodno da slučajni pristup bude ograničen na pristupanje početku GOP, koji uvek počinje intra frejmom (I slika). Po preporuci MPEG2 standarda, trebalo bi da u jednoj sekundi bude dva GOP, kako bi se nakon promene kanala, maksimalni interval promene servisa ograničio na 1/2 sekunde. Da bi se obezbedile najbolje performanse snimljenom TS je pridružena datoteka za skladištenje indeksnih informacija, koje omogućuje jednostavnu

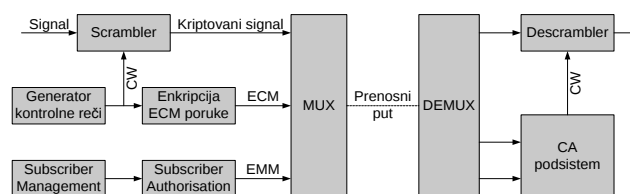
navigaciju kroz podatke, a time i slučajan pristup datoteci. Indeksni podaci predstavljaju adrese početaka GOP, slika i vremenskih informacija. U slučaju da se radi o kriptovanom toku podataka, GOP ne može biti indeksiran, tako da indeksna datoteka ne sadrži informacije o položaju GOP, odnosno ključnih slika.

B) Načini čuvanja zaštićenog sadržaja na HDD (DRM)

Možemo izdvojiti tri osnovna načina snimanja zaštićenog sadržaja:

- **bez enkripcije** na HDD se čuva nekriptovan sadržaj. Neposredno pre snimanja sadržaj se dekriptuje i u tom obliku čuva. Ovakav pristup nije prihvatljiv sa stanovišta povrede autorskih prava
- **sa enkripcijom** sadržaj se čuva u kriptovanom obliku, a dekripcija se vrši neposredno pre reprodukcije.
- **lokalna enkripcija** (ili re-enkripcija) Po ovom modelu (re-enkripcije ili lokalne enkripcije), sadržaj se pre upisivanja na HDD dekriptuje, a zatim ponovo enkriptuje nekim od poznatih algoritama za kriptovanje koji je implementiran u datom uređaju.

C) CA Sistem



Sl. 2. Blok dijagram sistema za uslovni pristup kriptovanim podacima

Sistem za uslovni pristup, CA sistem, čiji je blok-dijagram prikazan na slici 2, obuhvata metode enkripcije podataka (*scrambling*) da bi se sprečio neovlašćen pristup određenim DTV servisima. Enkripcija je proces zaštite tajnih ključeva, koji se prenose zajedno sa šifrovanim TV sadržajem, da bi se omogućilo njegovo dešifrovanje. Ključ za šifrovanje, tzv. kontrolna reč (*Control Word*) se šalje prijemnoj strani u kriptovanom obliku u vidu tzv. ECM poruka. CA podsistem prijemnika će dekriptovati kontrolnu reč samo ukoliko poseduje ovlašćenje, koje se prijemniku šalje preko tzv. EMM poruka. Ovaj slojeviti pristup zaštiti podataka je osnova svih CA sistema koji su danas u upotrebi.

U skladu sa MPEG2 standardom sve potrebne informacije za dešifrovanje se prenose do prijemnika putem CAT tabela, koje sadrže i ECM i EMM podatke. Svi prijemnici upoređuju kod za autorizaciju koji se nalazi u EMM sa kodom koji se nalazi u lokalnoj *Smart* kartici u prijemnom uređaju. Ukoliko je uređaj autorizovan za dekripciju podataka, kontrolna reč se generiše iz ECM poruke i dekripcija podataka može da počne. Ukoliko dekriptovanje nije autorizovano, korisnik se o tome obaveštava odgovarajućom porukom na ekranu. Kontrolna reč ili ECM se obično menja u intervalima od 10 sekundi, a EMM u mesečnim intervalima.

D) Rukovanje ECM porukama

Za pouzdanu reprodukciju zaštićenog (kriptovanog) sadržaja neophodna je dobra sinhronizacija sa podsistemom za generisanje kodnih ključeva. Pošto se kodni ključevi generišu na osnovu sadržaja ECM poruka, rukovanje ECM porukama je detaljno prikazano. Praćenje ECM toka može biti dvojako:

- Indeksiranje kompletnog ECM toka
- Indeksiranje promena unutar ECM toka

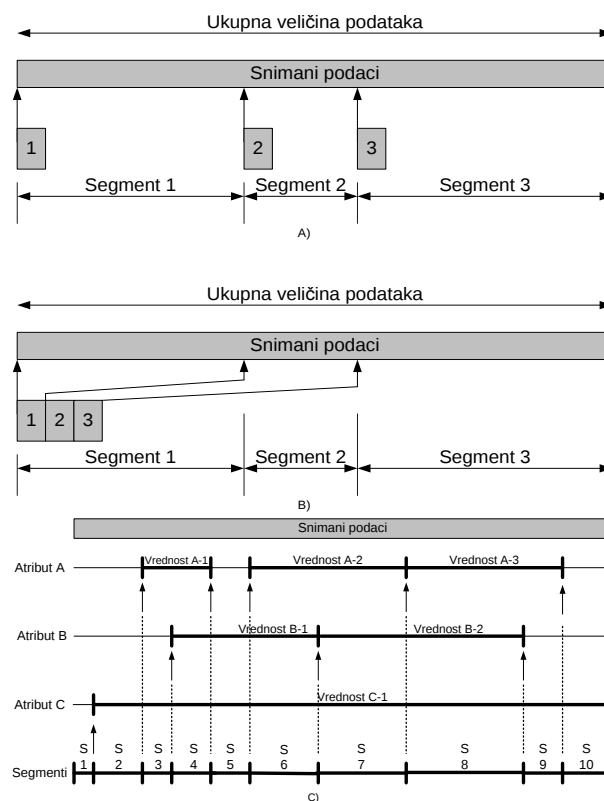
Uobičajena praksa je da se dekriptovanje, pri normalnoj brzini reprodukcije, vrši “u letu” linearno čitajući kriptovani sadržaj i ECM poruke iz multipleksiranog toka podataka. Za ovakav način rada dovoljno je ECM poruke sačuvati zajedno sa sadržajem. U trik modovima rada je potrebno da dekriptovanje bude brže od ralnog vremena, pa je stoga neophodno obezbediti indeksne informacije koje će ukazivati na adrese na kojima se nalaze ECM podaci kako bi se iz njih brzo mogli generisati ključevi i otpočeti dekriptovanje.

Znajući da ECM poruke imaju isti sadržaj dok ne istekne krypto-period, može se izbeći skladištenje kompletnog ECM toka. Potrebno je sačuvati jedino poziciju ECM poruke kad se promeni. Na ovaj način se smanjuje količina podataka koja se upisuje. Umesto svih ECM upisuju se samo novi ECM podaci. I pozicije i sadržaj će se upisati u indeksnu datoteku (za razliku od prethodnog slučaja gde se upisuju samo adrese).

III. OPIS REALIZACIJE SISTEMA

A) Opis arhitekture PVR

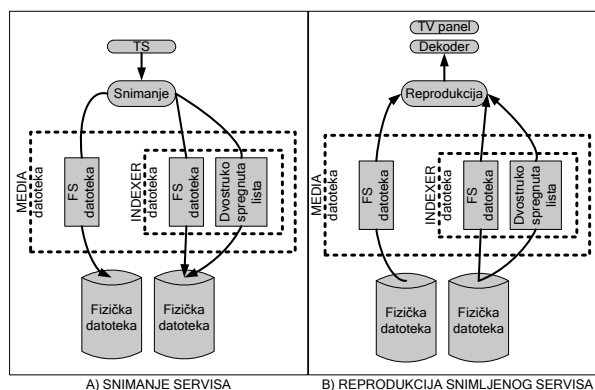
Tokom proces snimanja TS na HDD, vrši se njegovao deljenje u logičke celine (segmente) pod datim kriterijumima. Segmenti su dinamičke prirode i zavise od raznih parametara toka podataka i njegove promene tokom vremena. Npr. segment može biti definisan na osnovu vremenskog perioda, segment može predstavljati kompletan GOP, jednu sliku, audio frejm, itd. U toku snimanja postojaće bar jedan segment. Zbog ove osobine, segment se može definisati i kao deo snimanog toka podataka za vreme kog se meta-podaci ne menjaju. Primer snimanja toka podataka i podela na segmente su prikazani slikom 3.



Sl. 3. Snimanje i segmentacija toka podataka

PVR se sastoji iz dva osnovna programska modula:

- **player** proces koji je zadužen za reprodukciju snimljenog DTV servisa
- **recorder** proces zadužen za snimanje izabranog DTV servisa na HDD.



Sl. 4. Arhitektura PVR, a) snimanje DTV servisa na masovnu memoriju, b) reprodukcija snimljenog DTV servisa

Svi resursi koji su nophodni za rad ova dva procesa su implementirani kao posebni objekti.

- **Task** jednostavan konačni automat koji se koristi tokom života PVR procesa, i predstavlja vezu sa operativnim sistemom
- **List** realizacija dvostruko spregnuta liste sa brzim pristupom
- **Indexer** objekat za manipulaciju indeksnim datotekama
- **File** generalizacija datotečnog sistema
- **Media** pristup pojedinim segmentima snimljenog TS-a

- **ScramblingCtrl** Kontrola rukovanja ECM podacima

Na slici 4 su, blok dijagramom, prikazani odnosi i funkcije navedenih objekata u oba procesa PVR.

B) Rukovanje snimljenim TS podacima

Raspoloživi FS (File System) omogućava simultano čitanje i pisanje u jednoj datoteci, ali je zbog arhitekture PVR, koja je bazirana na dva nezavisna procesa koji istovremeno mogu biti aktivni i pristupati istoj datoteci, teško održati datoteke konzistentnim. Ako se PVR nalazi u nekom od modova za "odloženo gledanje", sinhronizacija čitanja i pisanja je od velike je važnosti, kako bi se izbeglo čitanje prepisanog sadržaja iz datoteke. Problem sinhronizacije dolazi so najvećeg izražaja ako je PVR u stanju "permanentnog snimanja" jer se tada datoteka koristi kao cirkularna.

Navedeni su problemi u slučaju da PVR radi u režimu odloženog gledanja:

- Kad je u toku snimanje, a reprodukcija se izvršava u nekom od trik modova, teško je razlikovati fizički i logički (pozicija pisanja) kraj datoteke
- Ako se reprodukcija izvršava usporeno, snimanjem će se prepisati pozicija čitanja (reprodukcije) i tada će *player* čitati invalidne podatke. Sa druge strane, ako se reprodukcija izvršava ubrzano, pozicija čitanja će sustići i preći poziciju pisanja
- U slučaju snimanja ili reprodukcije iz cirkularne datoteke, pored problema prepisivanja, PVR mora da vodi računa o fizičkim granicama iste, i da u slučaju dostizanja granica preuzme odgovarajuće akcije, što sve zajedno proizvodi kompleksne procese čitanja i pisanja.

Da bi se rešili navedeni problemi, implementiran je poseban objekat **File**, koji nudi jedinstvenu spregu za linearne i cirkularne datoteke. U isto vreme vodi računa o sinhronizaciji čitanja i pisanja, štiteći datoreku u slučajevima simultanih pristupa.

C) Indeksiranje Segmenata TS

Objekat **Indexer** je odgovoran za prikupljanje indeksnih informacija o snimanom TS. Pod indeksnim informacijama se pordazumevaju početak GOP i slike bilo kog tipa, u slučaju kad DTV servis nije kriptovan. U suprotnom, ako je servis kriptovan, indksi će ukazivati samo na početke blokova od 96256 B (94 KB) podataka, jer ostali indeksni podaci nisu dostupni pre dekriptovanja.

Indeksi će se čuvati u zavisnosti od izabranog moda snimanja. U slučaju permanentnog snimanja (cirkularno snimanje) indksi će se čuvati u dvostruko spregnutoj listi koju obezbeđuje objekat **List**, dok će u slučaju linearnog snimanja indksi biti sačuvani u fizičkoj datoteci, takodje korišćenjem objekta **List**.

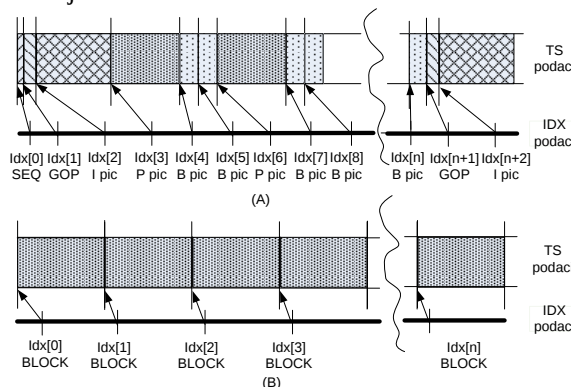
D) Pristup segmentima snimljenog TS

Objekat **Media**, predstavlja omotač oko objekata **File** i **Index**, i skriva njihovo postojanje od korisnika i ostatka sistema. Neophodno je obezbediti jedinstvenu spregu za procese snimanja i reprodukcije (*Recording* i *Playback*) koja će ih sprečiti od simultanog pristupanja istoj datoteci, i da bi se osiguralo sigurno korišćenje svih operacija nad

podacima i indeksima u višeprocenom okruženju. U suprotnom, mogu da se dese sledeći problemi vezani za indeksne podatke:

- filtrirani TS je sačuvan na HDD ali indksi još nisu upisani. U tom trenutku **Player** može da detektuje kraj datoteke ili može pročitati invalidne, stare indeksne podatke.
- novi indksi su upisani, ali podaci još nisu upisani na HDD. **Player** će pročitati invalidne podatke.

I **Player** i **Recorder** koriste **Media** objekat bez direktnog pristupanja **File** i **Index** objektima. Na ovaj način se omogućuje da PVR procesi budu nezavisni od korišćenog FS. Na slici 5 je prikazan izgled jedne instance **Media** objekta.



Sl. 5. Izgled media objekta, a) media u slučaju da je snimljni servis ne kriptovan, b) media za slučaj kriptovanog sevisa

IV. OPIS REALIZACIJE CA PODSISTEMA

A) Objekat ScramblingCtrl

Ovaj objekat služi za kontrolu enkriptovanja i dekriptovanja i preko njega je realizovan CA podsistem.

Različiti sistemi za emisiju programa imaju različite metode zaštite svojih sadržaja. Kako bi se prevazišli problemi pri testiranju, proizvedeni ovakvim stanjem, uvedena je funkcija koja kriptuje sadržaj pre upisa na masovnu memoriju. Za realizaciju enkripcije podataka korišćena je preporuka iz MPEG2 standarda koja se odnosi na ovu oblast.

B) Enkripcija

Enkripcija se izvršava neposredno pre snimanja podataka na masovnu memoriju, a u procesu *Recorder*.

Najpre se određuju ključevi za kriptovanje, koji su u ovom slučaju pseudo slučajne vrednosti, i upisuju se u odgovarajuće tabele. Zatim se postavljaju kontrolni biti paketa (*Transport Scrambling Control Bits*) tako da ukazuju da je on kriptovan i svi TS paketi se kriptuju sa odgovarajućim ključem, parnim ili neparnim, koji je pročitao iz najbliže prethodne tabele.

Pošto je podrška za enkripciju imlementirana zbog potreba testiranja, nije korišten ni jedan od komercijalno definisanih sistema (CSA, DES, TDES). Kao metoda je izabrana jednostavna operacija ekskluzivno ili (**XOR**) između podataka i odgovarajućeg ključa. Operacije XOR je izabrana zbog revirzibilnosti, odnosno osobine (1):

$$(A \oplus B) \oplus B = A \quad (1)$$

Ako se dva puta primeni na iste podatke, novodobijeni podaci će biti isti kao i polazni.

Kriptovanje se izvršava nad celim paketom osim prva 4 bajta zaglavlja, jer bi u suprotnom bilo nemoguće utvrditi o kom se paketu radi i koji se ključ koristi (parni ili neparni). Na slici 5 je prikazan izgled zaglavlja TS paketa. Tako izmenjeni podaci se upisuju na HDD i postaju nečitljivi ukoliko se ne izvrši proces dekrpcije.

sync byte 8	transport error indicator 1	payload unit start indicator 1	transport priority 1	PID 13	transport scrambling control 2	adaptatio n field control 2	continuity counter 4	adaptation field
----------------	--------------------------------------	---	----------------------------	-----------	---	--------------------------------------	----------------------------	---------------------

Sl. 6. Zaglavlje TS paketa

C) Dekripcija

Dekripcija se izvršava nakon čitanja podataka sa HDD, a pre dekodovanja i prikazivanja. Reprodukcijska se može izvršavati normalnom brzinom (normalna reprodukcija), kao i ubrzano/usporeno u napred i u nazad (što se naziva trik modovima reprodukcije) U zavisnosti od moda rada razlikuje se i način dekrpcije.

Ukoliko je *Player* u režimu normalne reprodukcije, iz memorije se čita redom, bez preskakanja sekvenci. Ažurira se tabela sa ključevima, a ako je pronađen video paket, izvršava se dekrpcovanje podataka. Dekriptovanje je, XOR operacija, sa istim ključem sa kojim je izvršeno i enkriptovanje.

