

SVEUČILIŠTE J. J. STROSSMAYERA U OSIJEKU

**FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TENHOLOGIJA**

Matej Jukić

**DOKUMENTACIJA ZA IZRADU ZADATKA
PPUTVIOS_23_2020_OS**

IZVJEŠTAJ

Osijek, 2022.

Sadržaj

1. UVOD	1
2. KONCEPT RJEŠENJA.....	2
2.1. Parser xml datoteke	2
2.1.1. Funkcije	2
2.2. Parser tablica	3
2.2.1. Enumi, strukture i funkcije	3
2.3. Grafički upravljač.....	3
2.3.1. Enumi i funkcije	3
2.4. Upravljač <i>timera</i>	4
2.4.1. Funkcije	4
2.5. Upravljač toka sadržaja	5
2.5.1. Enumi, strukture i funkcije	5
2.6. Daljinski upravljač	6
2.6.1. Enumi i funkcije	6
3. PRIMJERI RADA.....	7
4. ZAKLJUČAK	12

1. UVOD

U današnje vrijeme digitalna televizija postaje najzastupljenija vrsta televizije, pogotovo u razvijenijim zemljama. Digitalnu televiziju može se definirati kao televiziju čiji se signal prenosi u binarnom formatu. Sam binarni format omogućuje raznovrsne metode kompresije te se još uvijek radi na novim metodama. Osim toga binarni signal je otporniji na interferenciju u odnosu na analogni te omogućuje kvalitetniji prijenos slike i zvuka, više audio kanala za jedan video, izbor formata slike i zvuka, odabir podnaslova, interaktivni multimedijски sadržaj, pristup na internet itd.

2. KONCEPT RJEŠENJA

Projektni zadatak je riješen koristeći znanja stečena na laboratorijskim vježbama, kao što su višenitnost, parsiranje tablica, crtanje na zaslonu, reprodukcija audio i video kanala i sl.

Projekt je podijeljen u šest modula: parser xml datoteke, grafički upravljač, daljinski upravljač, upravljač *timera*, upravljač toka sadržaja, parser tablica.

Svaki od tih modula tvori zasebnu logičku cjelinu što poboljšava preglednost koda i olakšava pronalazak grešaka.

2.1. Parser xml datoteke

Modul je zadužen za čitanje početnih vrijednosti iz config.xml datoteke.

Config datoteka sadrži vrijednosti kao što su frekvencija, širina pojasa/*bandwidth*, vrsta modula, audio i video PID-ovi početnog kanala, tip audio i videozapisa, datum i vrijeme sustava.

2.1.1. Funkcije

Modul sadrži sljedeće funkcije:

parseConfigurationFile(char *fileName, initialConfig *config) – čita podatke iz xml datoteke i sprema ih u initialConfig strukturu

initValues(initialConfig *config) – poziva se u parseConfigurationFile funkciji i sprema podatke u initialConfig strukturu.

checkValues(initialConfig *config) – provjerava vrijednosti pohranjene u initialConfig strukturu.

printValues(initialConfig *config) – ispisuje vrijednosti pohranjene u initialConfig strukturu.

2.2. Parser tablica

Ovaj modul je zadužen za parsiranje vrijednosti PAT tablice i PMT sekcije. Pomoću njih se dohvaća i pamti popis svih usluga s parametrima potrebnim za dekodiranje. Nalazi se u datotekama `tables_parser.h` i `tables_parser.c`.

2.2.1. Enumi, strukture i funkcije

Sadrži enum `_tablesParserStatus`, koji pamti stanje parsera i pojavu greški, zatim strukturu `_patTableHeader` koja sadrži zaglavlje PAT tablice, `_patTableProgramInformation` koja sadrži popis programa, `_patTable` koja se sastoji od prethodno navedenih struktura `_patTableHeader`

i `_patTableProgramInformation`. Zatim sadrži `_pmtTableHeader` koja pamti zaglavlje PMT tablice, `_pmtTableElementaryInformation` s osnovnim informacijama PMT tablice, te `_pmtTable` koja spaja ove dvije strukture u jednu.

Od funkcija sadrži `parsePAT(uint8_t *buffer, patTable *pat)` za parsiranje PAT tablice, `parsePMT(uint8_t *buffer, pmtTable *pmt)` za parsiranje PMT tablica, `printPAT(patTable *pat)` za ispis PAT tablice, `printPMT(pmtTable *pmt)` za ispis PMT tablice.

2.3. Grafički upravljač

Sadrži funkcije potrebne za iscrtavanje elemenata iz zadatka. Iscrtava grafički element sa stanjem zvuka i grafički element s prikazom trenutnog programa i brojem podnaslova koji se nalaze na zadanom kanalu. Modul koristi datoteke `graphics_controller.c` i `graphics_controller.h`.

2.3.1. Enumi i funkcije

Sadrži enum `_graphicsControllerStatus` u kojem se nalaze stanja grafičkog upravljača.

Od funkcija sadrži:

graphicsControllerInit() inicijalizira grafički upravljač,

graphicsControllerDeinit() koja služi za deinicijalizaciju istog,

drawChannelNumber(uint16_t channelNumberValue) za iscertavanje broja unesenog na daljinskom upravljaču u gornjem lijevom kutu ekrana,

drawChannelNumberMessage(uint16_t channelNumberValue) za isporuke u gornjem lijevom kutu ekrana ukoliko uneseni kanal ne postoji,

drawChannelInfo(uint16_t channelNumberValue, uint8_t subtitleCount, char *subtitles) za iscertavanje okvira unutar kojeg se nalazi poruka s brojem trenutnog kanala i brojem dostupnih podnaslova,

drawVolumeInfo(float volumePercent) za prikazivanje okvira koji sadrži grafički i brojevi prikaz glasnoće u desnom kutu ekrana,

drawOnScreen() funkciju u kojoj se pozivaju prethodno navedene funkcije za konačno iscertavanje na ekranu,

clearScreen(uint8_t alpha) funkciju koja briše sadržaj sa zaslona.

2.4. Upravljač *timera*

Služi za mjerenje vremena kako bi se moglo iskoristiti u funkcijama koje su ovisne o njemu.

Modul koristi datoteke `timer_controller.c` i `timer_controller.h`

2.4.1. Funkcije

Modul koristi funkcije:

timerSetAndStart(timer_t *timerId, time_t triggerSec, void *callback) za postavljanje i pokretanje *timera* te funkciju

timerStopAndDelete(timer_t *timerId) za zaustavljanje i brisanje *timera*.

2.5. Upravljač toka sadržaja

Služi za pokretanje odabranih kanala i odabir sadržaja na njima. Također koristi prethodno navedene module. Nalazi se u datotekama `stream_controller.c` i `stream_controller.h`.

2.5.1. Enumi, strukture i funkcije

Sadrži enum `_streamControllerStatus` s informacijama o postojanju grešaka kod upravljača toka sadržaja te

`_dvbStreamType` s podržanim tipovima kodiranja audio i video sadržaja.

Od struktura tu su `_channelData` koja sadrži podatke o trenutnom sadržaju kao što su vrijeme početka, trajanja, ime sadržaja itd.,

`_channels` koja sadrži u sebi prethodno navedenu strukturu i broj kanala.

Od funkcija se pozivaju:

`streamControllerInit(initialConfig *config)` za inicijalizaciju *tunera* i *playera*,

`streamControllerDeinit()` za deinicijalizaciju,

`startPlayerStream(startingChannelInit *channel)` za pokretanje toka sadržaja,

`stopPlayerStream()` za zaustavljanje toka sadržaja,

`channelsSetup()` za postavljanje kanala na temelju PAT i PMT tablice,

`playChannel(uint16_t channelNumber)` za pokretanje kanala,

`playNextChannel()` za pokretanje idućeg kanala,

`playPreviousChannel()` za pokretanje prethodnog kanala,

`volumeMute()` za potpuno utišavanje zvuka,

`volumeUp()` za pojačavanje zvuka,

`volumeDown()` za stišavanje zvuka,

showChannelInfo() za pokazivanje informacija o kanalu,
showVolumeInfo() za prikaz trenutne glasnoće,
showChannelNumber(uint16_t channelNumberValue) za prikaz unesenog broja kanala,
showChannelNumberMessage(uint16_t channelNumberValue) za prikaz poruke o nepostojanju kanala.

2.6. Daljinski upravljač

Sadrži funkcionalnosti potrebne za tumačenje signala zaprimljenih s daljinskog upravljača. Nalazi se u `remote_controller.c` i `remote_controller.h` datoteci.

2.6.1. Enumi i funkcije

Sadrži **_remoteControllerStatus** koji sadržava informacije o statusu upravljača.

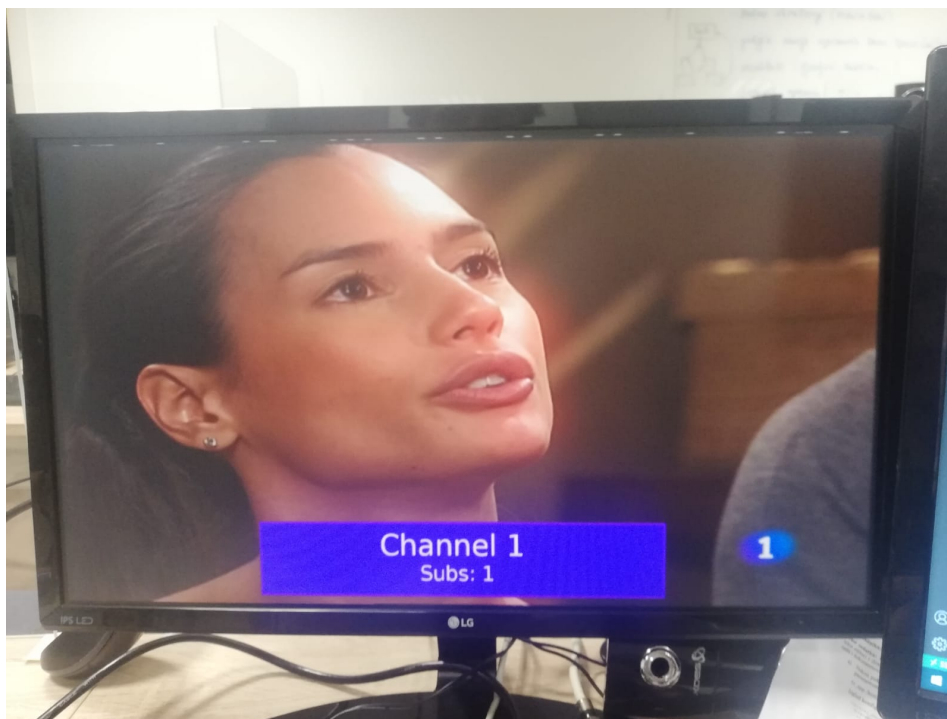
Od funkcija tu su **remoteControllerInit()** koja služi za inicijalizaciju upravljača,

remoteControllerEvent() za izvršavanje funkcija ovisno o pritisnutom gumbu na daljinskom upravljaču,

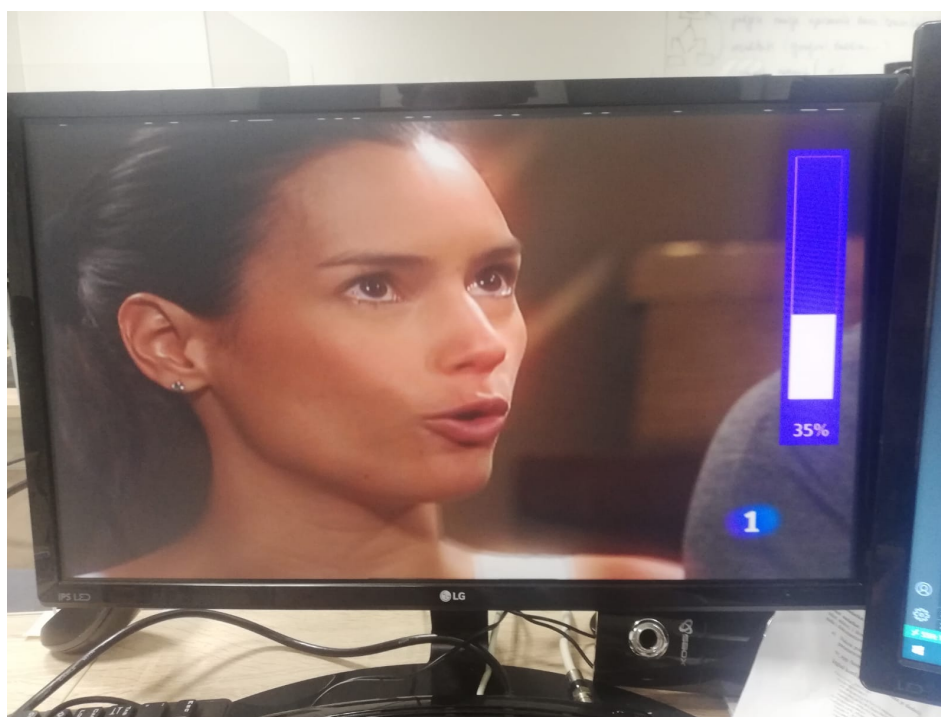
getKeys(int32_t count, uint8_t *buf, int32_t *eventsRead) za dohvaćanje pritisnutih tipki na daljinskom upravljaču,

generateChannelNumber(uint8_t remoteKey) za generiranje broja kanala iz stringa pritisnutih tipki.

3. PRIMJERI RADA



Slika 1. prikaz prvog kanala s info okvirom.



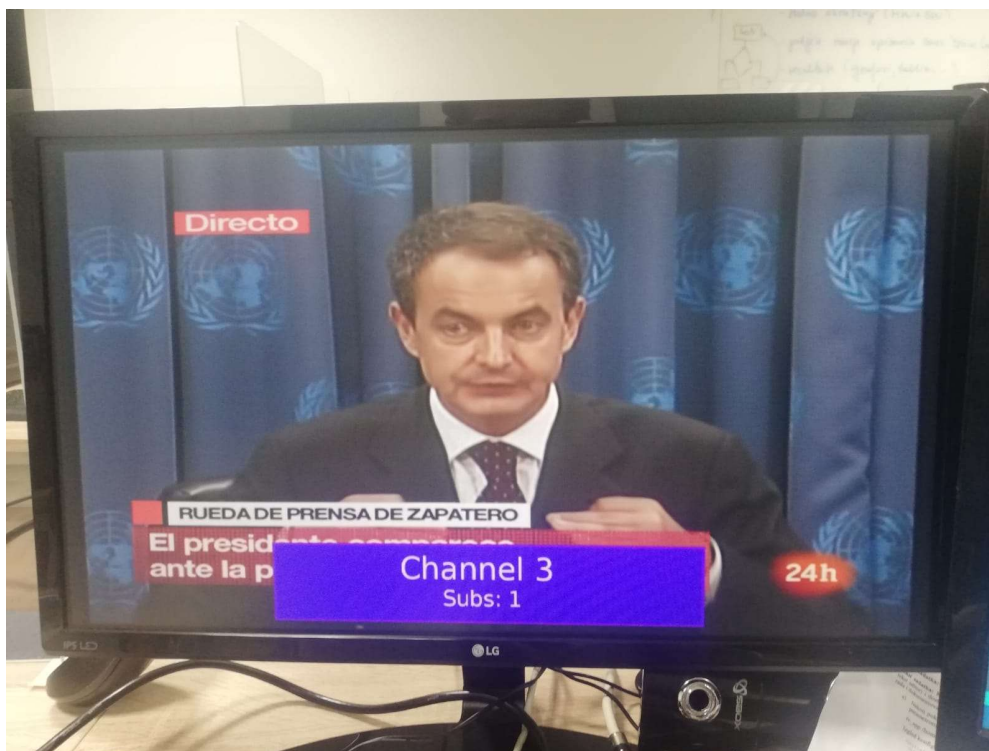
Slika 2. prikaz prvog kanala sa prikazom glasnoće



Slika 3. prikaz drugog kanala s info okvirom



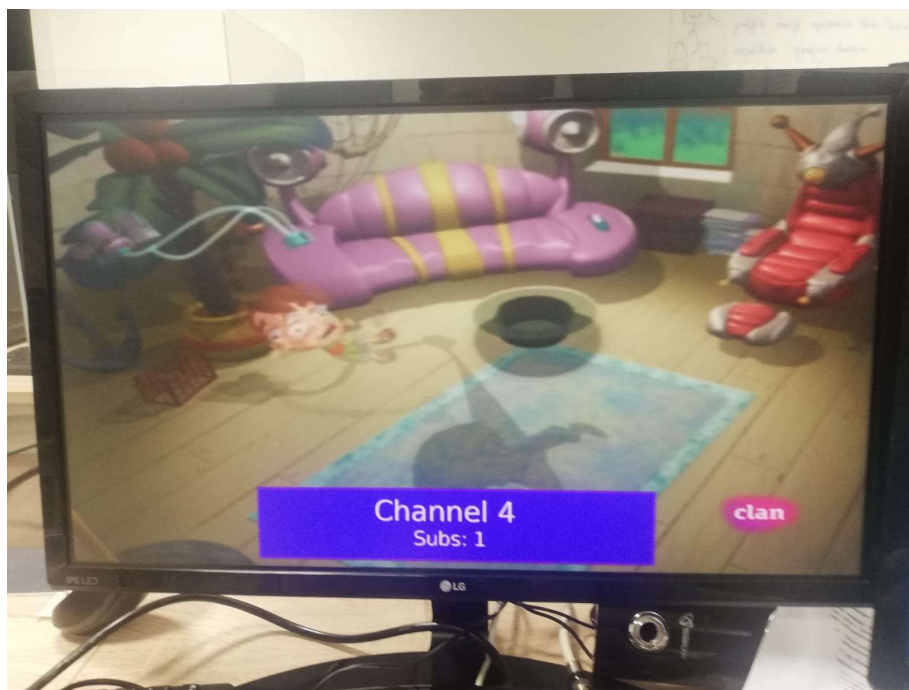
Slika 4. Prikaz drugog kanala s prikazom glasnoće



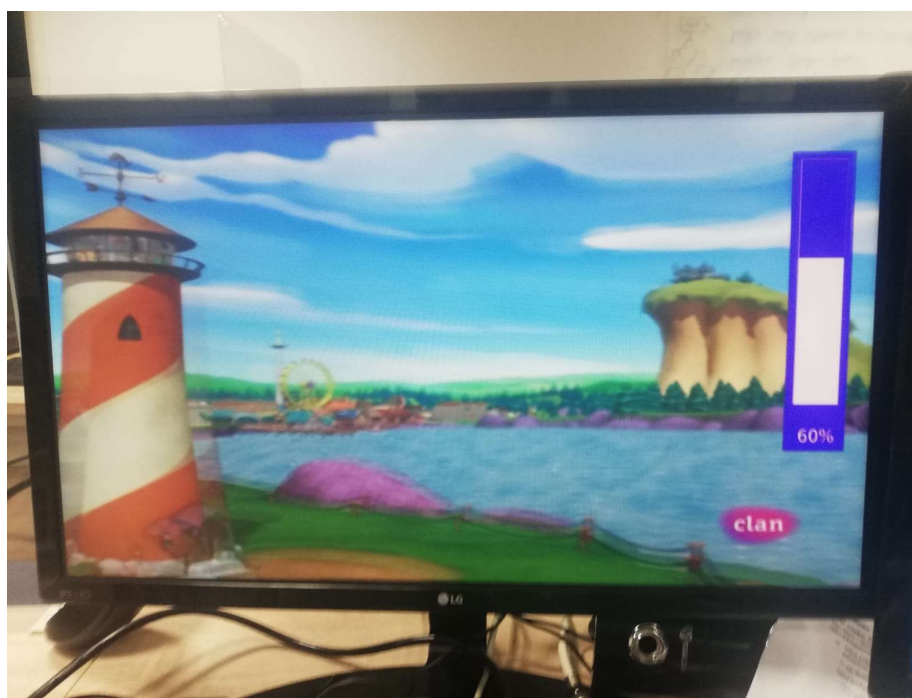
Slika 5. prikaz trećeg kanala s prikazom info okvira



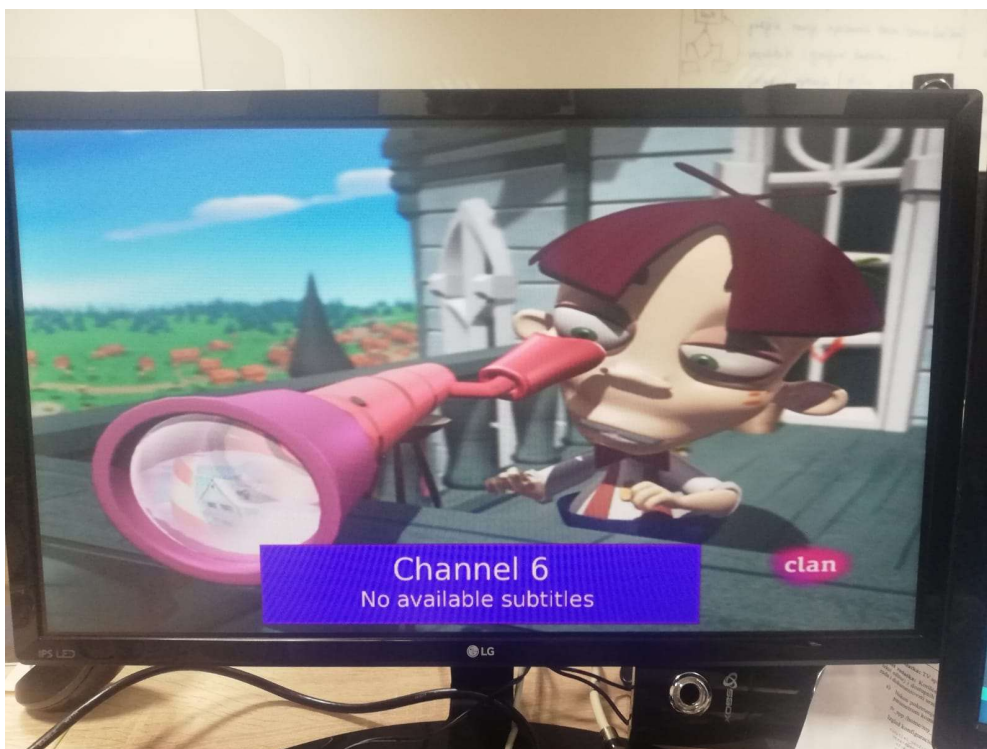
Slika 6. prikaz trećeg kanala s prikazom glasnoće zvuka



Slika 7. prikaz četvrtog kanala s info okvirom



Slika 8. prikaz četvrtog kanala s prikazom glasnoće



Slika 9. primjer kanala bez podnaslova

4. ZAKLJUČAK

Aplikacija je izrađena da zadovolji minimalne zahtjeve navedene u projektnom zadatku. Sadrži funkcionalnosti kao što su mijenjanje kanala, stišavanje, pojačavanje i isključivanje zvuka, te prikaz osnovne informacije o kanalu i jačini zvuka. Mogla se još proširiti dodavanjem EIT tablice iz koje bi se čitale informacije o sadržaju te bi se tako zadovoljilo i ostale zahtjeve navedene u zadatku.