



УНИВЕРЗИТЕТ У НОВОМ САДУ  
ФАКУЛТЕТ ТЕХНИЧКИХ НАУКА У  
НОВОМ САДУ



---

Вања Комадина

# АУТОМАТИЗАЦИЈА НОРДИГ ВЕРИФИКАЦИЈЕ ЗА ДИГИТАЛНЕ ТВ ПРИЈЕМНИКЕ

ДИПЛОМСКИ РАД  
- Основне академске студије -

Нови Сад, октобар 2015.



## КЉУЧНА ДОКУМЕНТАЦИЈСКА ИНФОРМАЦИЈА

Редни број, РБР:		
Идентификациони број, ИБР:		
Тип документације, ТД:	Монографска документација	
Тип записа, ТЗ:	Текстуални штампани материјал	
Врста рада, ВР:	Завршни (Bachelor) рад	
Аутор, АУ:	Вања Комадина	
Ментор, МН:	проф. др Жељен Трповски	
Наслов рада, НР:	Аутоматизација НорДиг верификације за дигиталне ТВ пријемнике	
Језик публикације, ЈП:	Српски / латиница	
Језик извода, ЈИ:	Српски	
Земља публиковања, ЗП:	Република Србија	
Уже географско подручје, УГП:	Војводина	
Година, ГО:	2015	
Издавач, ИЗ:	Ауторски репринт	
Место и адреса, МА:	Нови Сад; трг Доситеја Обрадовића 6	
Физички опис рада, ФО: (поглавља/страна/цитата/табела/слика/графика/прилога)	8/24/0/2/8/0/0	
Научна област, НО:	Електротехника и рачунарство	
Научна дисциплина, НД:	Рачунарска техника	
Предметна одредница/Кључне речи, ПО:	Аутоматизација, ББТ окружење, ДВБ-Т/Т2, ИРД, НорДиг, Пајтон скрипте	
УДК		
Чува се, ЧУ:	У библиотеци Факултета техничких наука, Нови Сад	
Важна напомена, ВН:		
Извод, ИЗ:	Овај рад описује процес аутоматизације НорДиг тестних процедура и представља уштеду времена која је постигнута аутоматизацијом дела тестних процедура.	
Датум прихватања теме, ДП:		
Датум одбране, ДО:		
Чланови комисије, КО:	Председник:	проф. др Никола Ђурић
	Члан:	проф. др Никша Јаковљевић
	Члан, ментор:	проф. др Жељен Трповски
		Потпис ментора



## KEY WORDS DOCUMENTATION

Accession number, <b>ANO:</b>		
Identification number, <b>INO:</b>		
Document type, <b>DT:</b>	Monographic publication	
Type of record, <b>TR:</b>	Textual printed material	
Contents code, <b>CC:</b>	Bachelor Thesis	
Author, <b>AU:</b>	Vanja Komadina	
Mentor, <b>MN:</b>	prof. dr Željen Trpovski	
Title, <b>TI:</b>	Automation of NorDig verification for digital TV receivers	
Language of text, <b>LT:</b>	Serbian	
Language of abstract, <b>LA:</b>	Serbian	
Country of publication, <b>CP:</b>	Republic of Serbia	
Locality of publication, <b>LP:</b>	Vojvodina	
Publication year, <b>PY:</b>	2015	
Publisher, <b>PB:</b>	Author's reprint	
Publication place, <b>PP:</b>	Novi Sad, Dositeja Obradovica sq. 6	
Physical description, <b>PD:</b> <small>(chapters/pages/ref./tables/pictures/graphs/appendices)</small>	8/24/0/2/8/0/0	
Scientific field, <b>SF:</b>	Electrical Engineering	
Scientific discipline, <b>SD:</b>	Computer Engineering, Engineering of Computer Based Systems	
Subject/Key words, <b>S/KW:</b>	Automation, BBT platform, DVB-T/T2, IRD, NorDig, Python scripts	
<b>UC</b>		
Holding data, <b>HD:</b>	The Library of Faculty of Technical Sciences, Novi Sad, Serbia	
Note, <b>N:</b>		
Abstract, <b>AB:</b>	This paper describes process of automation of NorDig test suite and presents savings in time that have been achieved by automating part of the test suite.	
Accepted by the Scientific Board on, <b>ASB:</b>		
Defended on, <b>DE:</b>		
Defended Board, <b>DB:</b>	President:	prof. dr Nikola Đurić
	Member:	prof. dr Nikša Jakovljević
	Member, Mentor:	prof. dr Željen Trpovski
		Menthor's sign

## SADRŽAJ

1.	Uvod .....	1
2.	NorDig IRD .....	2
3.	Okruženje za automatsko testiranje.....	4
4.	Analiza NorDig testova.....	9
5.	Python skripte.....	12
6.	Analiza rezultata.....	19
7.	Zaključak .....	22
8.	Literatura.....	24

## SPISAK SLIKA

Slika 1. Arhitektura NorDig digitalnog prijemnika .....	2
Slika 2. BBT okruženje .....	4
Slika 3. Komunikacija sa uređajima preko RT-Executora.....	5
Slika 4. RT-Executor .....	6
Slika 5. Reference Creator.....	8
Slika 6. Algoritam Python testova .....	16
Slika 7. Primer Python testa (Task 7:2) .....	17
Slika 8. Primer Excel datoteke (Task 3:37) .....	21

## **SPISAK TABELA**

Tabela 1. Lista testova koji se odnose na DVB-T/T2 digitalni prijemnik. ....	10
Tabela 2. Rezultati merenja vremena potrebnog za izvršenje testa (Task 3:35) ..	19

## SKRAĆENICE

<b>API</b>	- Application Programming Interface, Aplikacioni programski sprežni sistem
<b>BBT</b>	- <i>Black Box Testing</i> , Testiranje metodom crne kutije
<b>CVBS</b>	- <i>Composite Video Broadcast Signal</i> , Kombinovani prenos video signala
<b>DLL</b>	- <i>Dynamic-link library</i> , Dinamička biblioteka
<b>DUT</b>	- <i>Device Under Testing</i> , Uredaj koji se testira
<b>DVB</b>	- <i>Digital Video Broadcasting</i> , Emitovanje digitalnog signala
<b>DVB-T</b>	- <i>DVB Terrestrial</i> , Standard za zemaljsko-digitalno video emitovanje
<b>DVB-T2</b>	- <i>DVB-T2 Terrestrial</i> , Druga generacija DVB-T standarda
<b>HDMI</b>	- <i>High-Definition Multimedia Interface</i> , Sprežni sistem za prenos kombinovanog zvučnog/video podatka u nekomprimovanom obliku
<b>IP</b>	- <i>Internet Protocol</i> , Internet protokol
<b>IR</b>	- <i>Infrared</i> , Infracrvena svetlost
<b>IRD</b>	- <i>Integrated Receiver Decoder</i> , Integrisani prijemnik/dekoder
<b>NORDIG</b>	- <i>Nordic Digital</i> , Specifikacija platforme za digitalnu televiziju koja se koristi u nordijskim zemljama
<b>RC</b>	- <i>Remote Controller</i> , Daljinski upravljač
<b>RF</b>	- <i>Radio Frequency</i> , Radio frekvencija
<b>STB</b>	- <i>Set Top Box</i> , Uredaj za prijem i obradu digitalnih zemaljskih televizijskih i satelitskih signala u domaćinstvima
<b>UHF</b>	- <i>Ultra-High Frequency</i> , Ultra visoka frekvencija

**VHF**

- *Very-High Frequency, Veoma visoka frekvencija*

## 1. Uvod

Razvoj programske podrške i fizičke arhitekture u potrošačkoj elektronici podrazumeva razvoj tehnike i alata za njihovo testiranje i proveru usglašenosti sa standardima. Jedan od dobro definisanih standarda jeste *NorDig*, koji definiše specifikaciju okruženja za digitalnu televiziju u nordijskim zemljama.

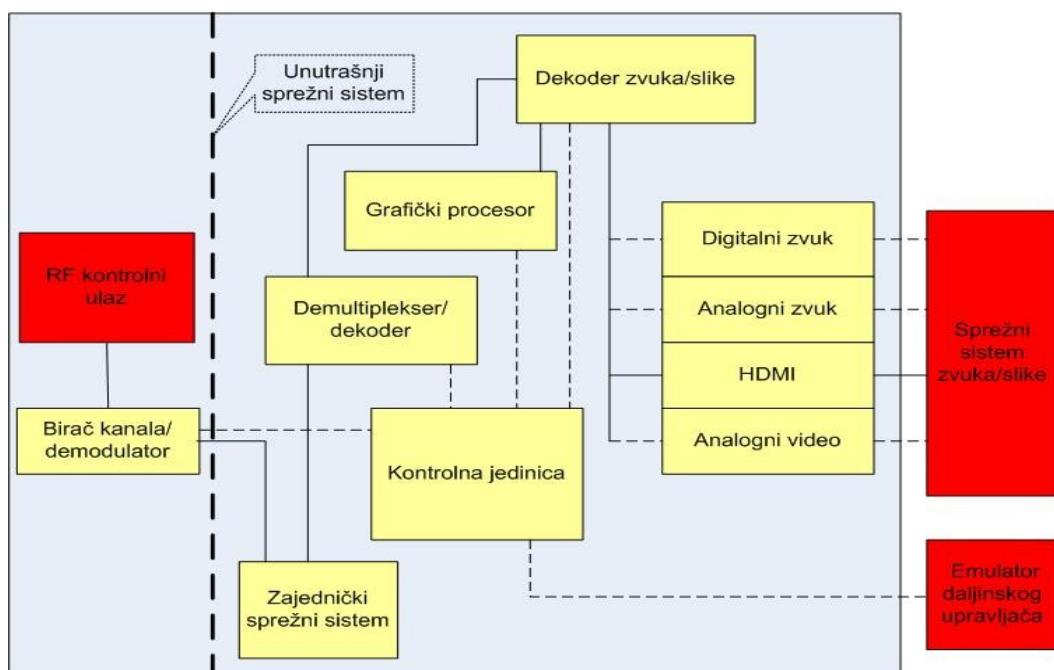
Rad opisuje proces automatizacije dela NorDig testnih procedura pomoću BBT okruženja i prednosti koje su dobijene procesom automatizacije. Prvi korak u procesu automatizacije bila je analiza testnih slučajeva za DVB-T/T2 digitalne prijemnike, koji su opisani u *NorDig Unified Test* planu, verziji 2.4. [1]. Analizirani su zahtevi u testovima, i oprema potrebna za izvršenje testova i proverena je kompatibilnost sa BBT okruženjem, što je rezultovalo listom testova koji su bili pogodni za automatizaciju. Sledеći korak bila je realizacija testnih procedura u Python jeziku. Na kraju su automatski testovi izvršavani, a dobijeni rezultati smeštani su u odgovarajuću bazu podataka i vršena je njihova analiza.

Takođe, rad pruža uvid u probleme koji onemogućavaju automatizaciju određenih testova i mere uštede vremena dobijene automatizacijom dela NorDig testova.

U poglavlju 2. opisan je digitalni prijemnik koji zadovoljava NorDig specifikacije i delovi tog prijemnika koji su bili relevantni za proces automatizacije. Poglavlje 3. opisuje okruženje koje je korišćeno za kreiranje automatskih testova. U poglavlju 4. prikazana je analiza NorDig testova i uslovi koji su trebali biti ispunjeni da bi se mogle kreirati Python skripte, koje su opisane u poglavlju 5. Konačni rezultati i mere uštede vremena postignute automatskim testiranjem prikazani su u poglavlju 6.

## 2. NorDig IRD

IRD odnosi se na digitalne prijemnike ili na relevantne delove integriranog digitalnog TV (iDTV) seta. NorDig IRD implementira servise kombinacijom rešenja fizičke arhitekture i programske podrške [2]. Treba da ima ugrađen ulaz za IP mrežu, zemaljski, kablovski ili satelitski birač kanala/demodulator i odgovarajući sprežni sistem. Proizvođač IRD uređaja sloboden je da odluči o fizičkoj arhitekturi uređaja, pod uslovom da su ispunjeni NorDig zahtevi. Tipična arhitektura NorDig digitalnog prijemnika prikazana je na slici 1.



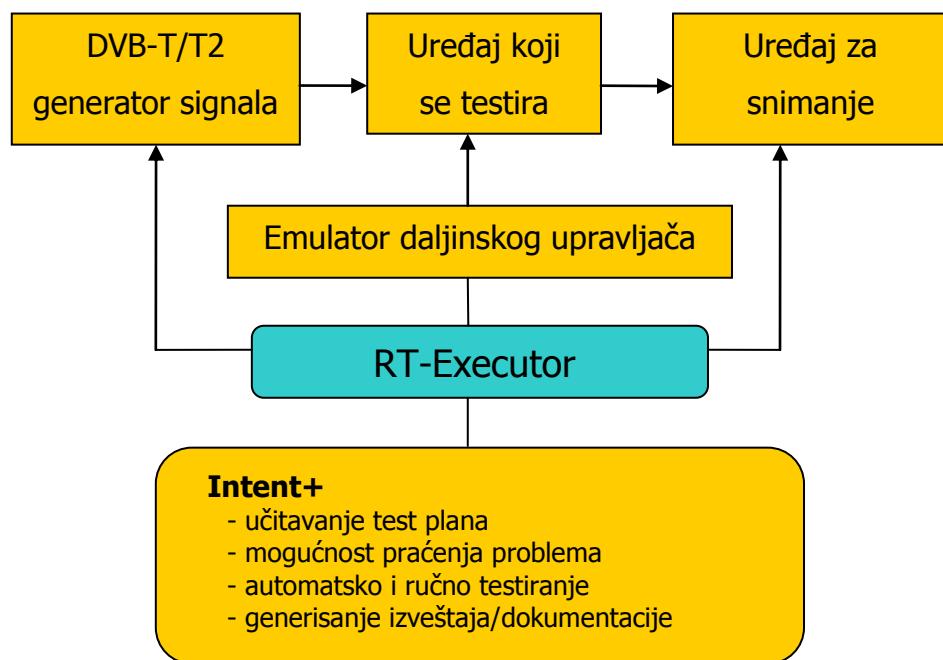
Slika 1. Arhitektura NorDig digitalnog prijemnika

Testiranje metodom crne kutije prepostavlja da se kontrolišu samo ulazni i izlazni delovi uređaja koji se testira, ne razmatrajući njegovu unutrašnju strukturu. Zbog toga su u ovom procesu automatizacije uzeti u razmatranje blokovi koji predstavljaju ulaze i izlaze uređaja koji se testira. Na slici 1 ti blokovi označeni su crvenom bojom. U pitanju su: *RF kontrolni ulaz, Sprežni sistem zvuka/slike i Emulator daljinskog upravljača*.

*RF sprežni sistem* prenosi ulazni modulisani signal do *Birač kanala/demodulator bloka*. Ovaj blok vrši selekciju kanala (frekvencija), demodulaciju i korekciju grešaka ulaznog signala. Izlaz iz demodulatora je korigovani signal koji se prenosi do demultipleksera. *Demultiplekser* je zadužen za odabir odgovarajućeg zvuka, slike ili tekstualnih podataka, u zavisnosti od zahteva korisnika. Privatnim podacima upravlja IRD kontroler, dok zvučnim i video podacima upravlja Dekoder zvuka/slike. U *Dekoderu zvuka/slike* vrši se dekodovanje, dekompresija, sinhronizacija sa povezanim servisima, digitalno-analogna konverzija i drugi procesi. Zatim se analogni signali dovode do SCART ili nekog drugog analognog izlaznog video konektora (Y, Pb, Pr, RF-PAL ili CVBS), dok se digitalni signali dovode do HDMI sprežnog sistema. *Grafička procesorska jedinica* generiše grafičke i tekstualne podatke omogućavajući njihov prikaz korisniku. *Kontrolna jedinica* upravlja svim unutrašnjim jedinicama i svim spoljnijim jedinicama koje su povezane sa uređajem. *Sprežni sistem zvuka/slike* predstavlja izlaz iz IRD uređaja i povezuje se na uređaj za prikaz slike i zvuka, koji snima odziv IRD uređaja. *Emulator daljinskog upravljača* upravlja sa IRD uređajem i njegovim aplikacijama, dovodeći uređaj u stanje/poziciju koja se očekuje u testu.

### 3. Okruženje za automatsko testiranje

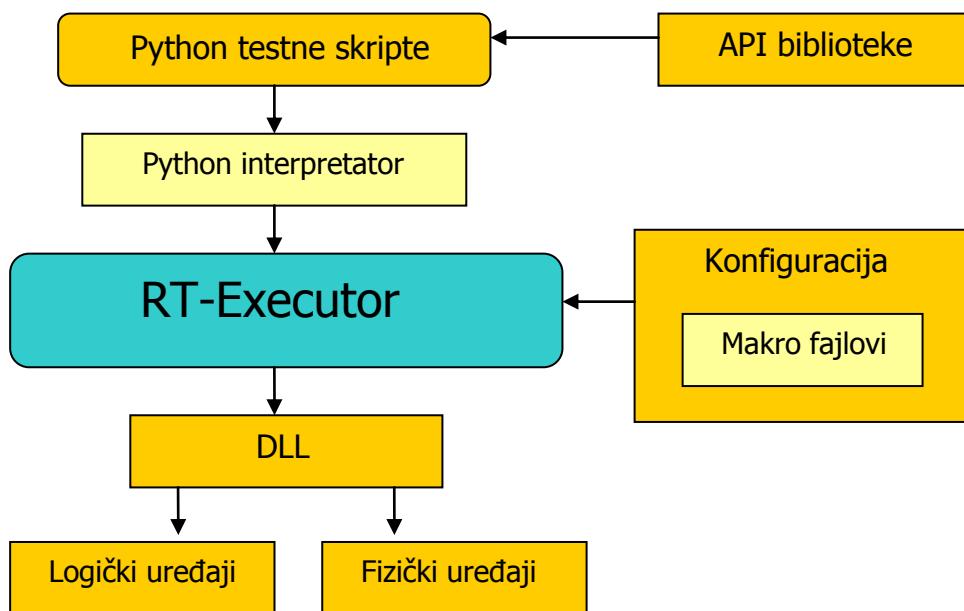
Za potrebe realizacije NorDig testnih procedura, korišćeno je BBT okruženje za testiranje koje se zasniva na testiranju pomoću metode crne kutije. *Metoda crne kutije* ne uzima u obzir način rada uređaja koji se testira. U toku testiranja posmatraju se samo izlazni podaci iz uređaja, koji se porede sa očekivanim, i na osnovu dobijenih rezultata donosi se odluka o ispravnosti testiranog urađaja. BBT okruženje sastoji se od programskih i fizičkih komponenata koje nude širok opseg mogućnosti za testiranje uređaja koji su u procesu razvoja [3]. Konfiguracija BBT okruženja prikazana je na slici 2.



Slika 2. BBT okruženje

### 3.1. RT-Executor

*RT-Executor* predstavlja alat za kontrolu, razvoj i izvršenje automatskih testova [4]. To je aplikacija pomoću koje se testne skripte unose u sistem i izvršavaju, i koja obezbeđuje kontrolu fizičkih i logičkih uređaja koji se koriste u procesu testiranja. Sadrži set biblioteka za analizu zvučnih i video podataka, bazu podataka za čuvanje dobijenih rezultata, višestruki korisnički pristup i podršku za Python skripte. Upravljanje uređajima preko RT-Executora prikazano je na slici 3.



Slika 3. Komunikacija sa uređajima preko RT-Executora

Funkcije koje se koriste u više različitih testnih procedura, radi lakšeg i bržeg kreiranja automatskih testova, objedinjene su u API bibliotekama. *Python interpretator* učitava testnu skriptu, interpretira svaku liniju koda te skripte i prosleđuje interpretirane komande RT-Executoru na dalju obradu. *Konfiguracija* testnog okruženja predstavlja listu fizičkih i logičkih uređaja koji su neophodni za uspešno izvršenje testnih procedura. *DLL* je biblioteka koja sadrži funkcije i varijable za konfiguraciju i kontrolu uređaja koji se koriste tokom izvršavanja testa.

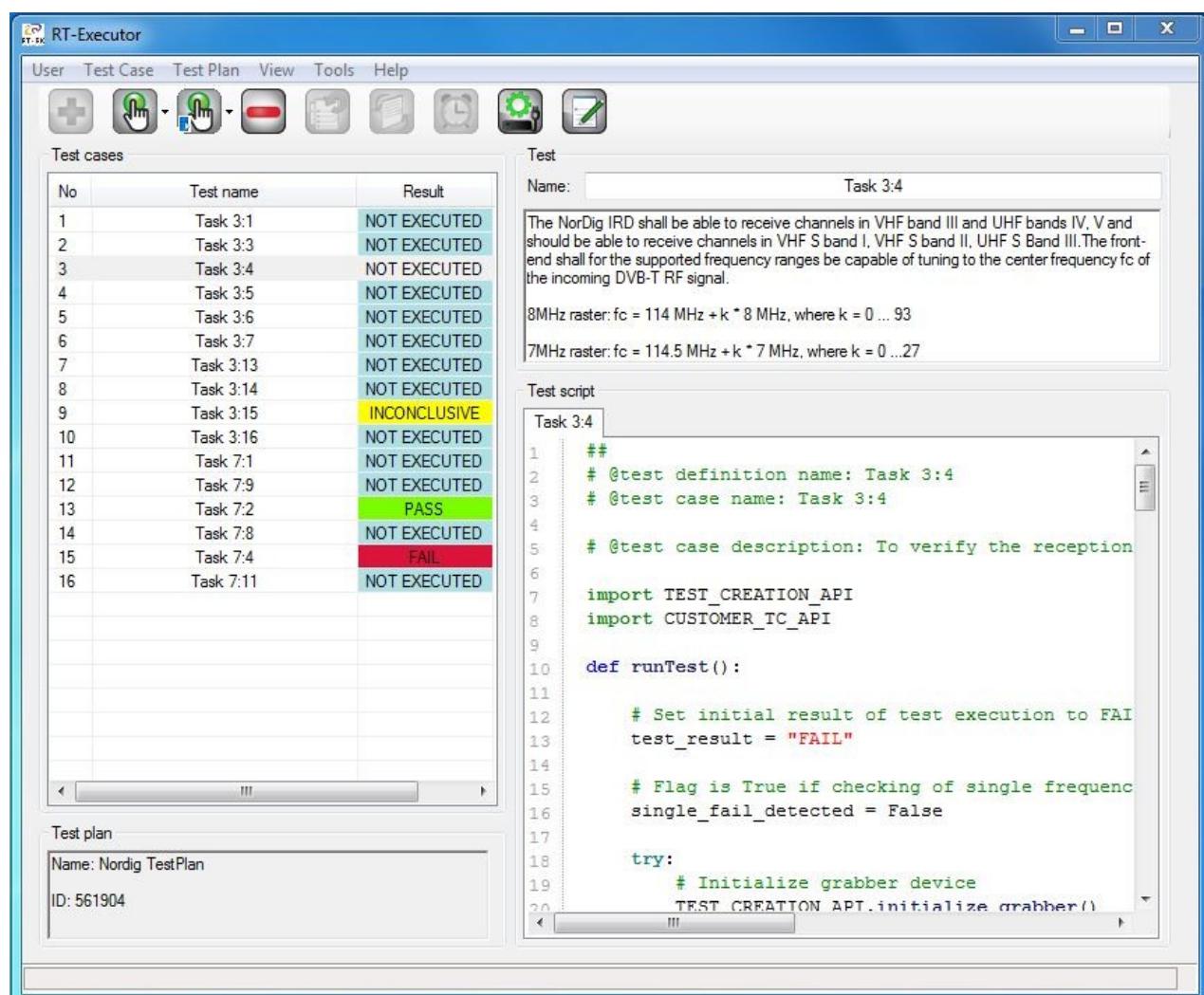
Fizički uređaji koji su korišćeni u ovom procesu automatizacije su sledeći:

- *RT-AV101* - uređaj koji se koristi za snimanje zvuka/slike u realnom vremenu
- *UT100B I DTA* - DVB-T i DVB-T2 generatori digitalnog toka podataka

Logički uređaji koji su korišćeni u ovom procesu automatizacije su sledeći:

- *PICTUREBLOCKCOMPARE* – algoritam za poređenje referentne slike sa slikom snimljenom pomoću uređaja za snimanje
- *FFT* – algoritam za poređenje referentnog zvuka sa zvukom snimljenim pomoću uređaja za snimanje
- *OCR* - algoritam za prepoznavanje tekstualnih karaktera na slici
- *REMOTECONTROLLER* – emulator daljinskog upravljača koji se koristi za kontrolu uređaja koji se testira
- *EXCEL* - biblioteka za upis i čuvanje dobijenih rezultata u Excel datoteku
- *MANUAL* – biblioteka za prikaz poruka definisanih unutar Python testa

RT-Executor aplikacija prikazana je na slici 4.



Slika 4. RT-Executor

### 3.2. Intent+

*Intent+* predstavlja centralno mesto testnog okruženja u koje se testovi unose i odakle se manipuliše njima tokom čitavog ciklusa automatizacije. Mnogobrojne mogućnosti koje pruža *Intent+* alat omogućuju jednostavno kreiranje i organizaciju automatskih testova [5].

Ovaj alat omogućava sledeće funkcionalnosti:

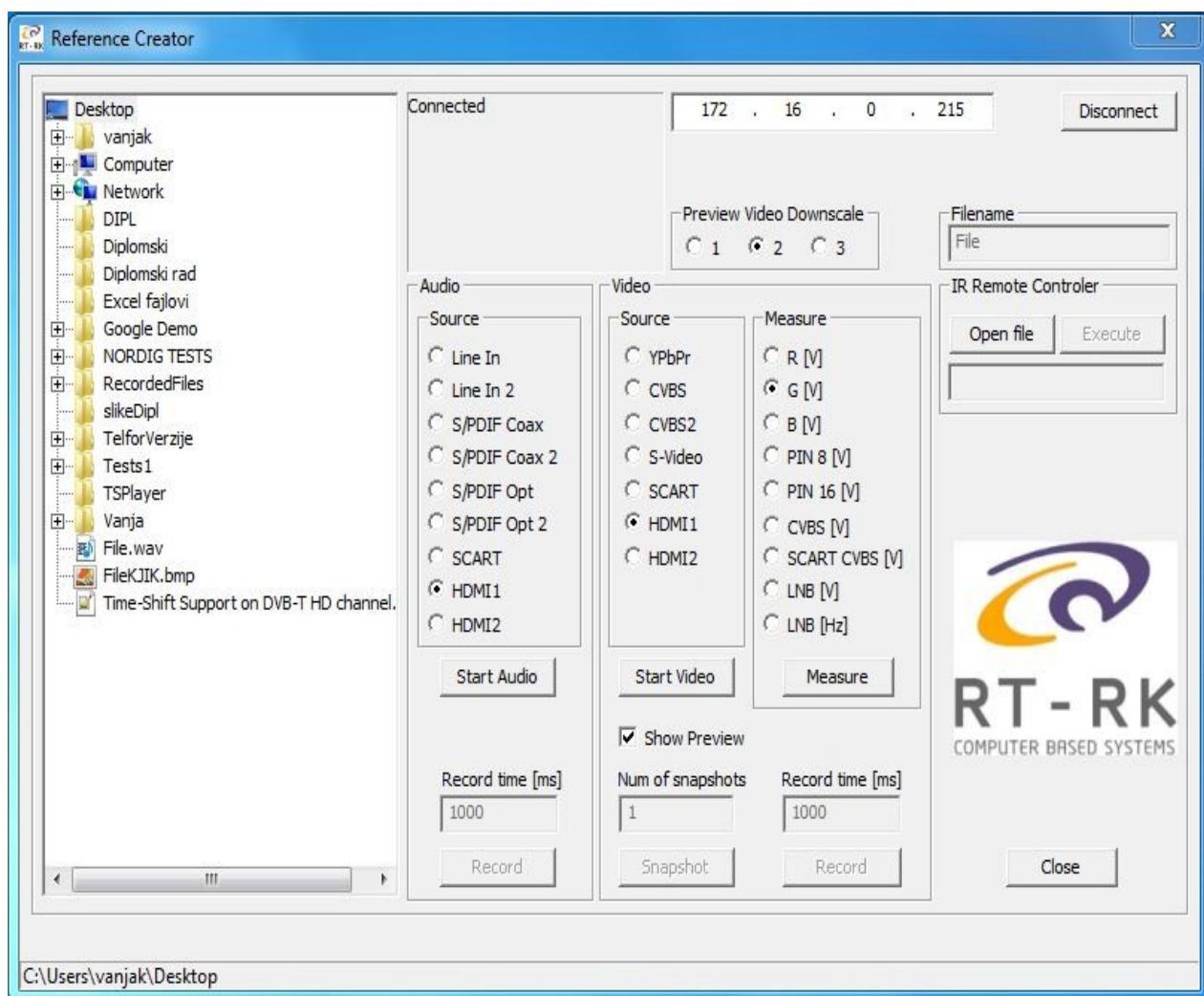
- unošenje zahteva korisnika i lako prilagođavanje zahtevima
- hijerarhijsku organizaciju unutar testnog paketa
- kreiranje testnog plana
- dodeljivanje osobe/osoba koja vrši testiranje i opreme koja će se koristiti za testiranje
- kreiranje automatskih testova pomoću ugrađenog uređivača Python koda
- pregled tokom ručnog izvršavanja testova
- automatsku integraciju rezultata testova sa sistemom za praćenje problema
- merenje vremena izvršenja testova
- dobru organizaciju projekta koja omogućava brz pristup svim resursima
- reorganizaciju unutar projekta
- pristupanje pomoći koja se odnosi na određenu stranicu aplikacije
- poređenje gotovih projekata

što je samo deo mogućnosti koje nudi *Intent+*.

### 3.3. Reference Creator

Budući da metoda crne kutije ispituje uređaj na osnovu poređenja izlaznih podataka iz uređaja koji se testira sa podacima koji su definisani u specifikacijama i predstavljaju referentne podatke, potrebno je da se referentni podaci definišu pre početka testiranja. U ovom procesu automatizacije ti podaci bili su datoteke slike i zvuka, a referentni zvuk i slika dobijeni su korišćenjem *Reference Creator* aplikacije [6]. Pored toga, aplikacija služi i za maskiranje određenih delova originalne slike ili za postavljanje regiona od interesa. Da bi aplikacija mogla da se koristi, potrebno je da se

uspostavi veza između računara i uređaja za snimanje koji je povezan sa uređajem koji se testira, i da se selektuje odgovarajući sprežni sistem, u zavisnosti od toga da li su podaci koji se porede analogni ili digitalni. Za digitalni video korišćen je HDMI sprežni sistem, za analogni video CVBS sprežni sistem i za analogni zvuk LINEIN sprežni sistem. Reference Creator aplikacija prikazana je na slici 5.



Slika 5. Reference Creator

## 4. Analiza NorDig testova

U svakom procesu automatizacije testova postoji nekoliko osnovnih koraka za njihovu analizu i pripremu. Pripremna faza za izvršenje testa podrazumeva dobijanje liste testnih slučajeva, kao i liste o fizičkim i logičkim uređajima koji će se koristiti. Nakon analize testa sledi realizacija testne skripte u Python programskom jeziku, izvršavanje testa i smeštanje dobijenih međurezultata i rezultata u bazu podataka.

Analiza testnih slučajeva u ovom procesu automatizacije izvršena je u zavisnosti od ograničenja BBT pristupa. Razmatrane su testne procedure i uređaji potrebni za izvršavanje testova. Da bi neki test mogao da se automatizuje, bilo je potrebno da zadovolji sledeće uslove:

1. Parametri RF ulaza moraju da se setuju automatski
2. Uređaj koji se testira može da se dovede u željeno stanje iz bilo kojeg drugog stanja, korišćenjem komandi daljinskog upravljača
3. Prolazni kriterijum može da se odredi na osnovu analize uslikane slike ili snimljenog zvuka

U NorDig Unified Test planu, verziji 2.4, postoji 251 test koji odgovara DVB-T/T2 digitalnom prijemniku [1]. Ovi testovi navedeni su u tabeli 1.

Tabela 1. Lista testova koji se odnose na DVB-T/T2 digitalni prijemnik

<b>Front-end issues</b>
Task 1: Satellite tuner and demodulator
Task 2: Cable tuner and demodulator
Task 3: Terrestrial tuner and demodulator
Task 4: IP-based front-end
<b>Hardware issues</b>
Task 5: MPEG2 demultiplexer and video/audio decoder
Task 6: System Software update
Task 7: Interfaces and signal levels
<b>System Software issues</b>
Task 8: SI and navigator
Task 9: Teletext and subtitling
Task 10: The remote control
Task 11: User preferences
Task 12: PVR functionality

Kao što se vidi u tabeli 1, NorDig testne specifikacije podeljene su u tri grupe srodnih testova.

Prva grupa testova odnosi se na testiranje ulaza uređaja. Testovi proveravaju sposobnost prijema signala za različite parametre modulatora DVB-T/T2 signala. NorDig IRD treba da bude sposoban da prima kanale (frekvencije) iz:

- VHF III opsega (177.5 – 226.5 [MHz])
- UHF IV opsega (474.0 – 858.0 [MHz]),

a opsezi:

- VHF S I (114.0 – 170.5 [MHz])
- VHF S II (233.5 – 298.0 [MHz])
- UHF S III (306.0 -466.0 [MHz]) su opcioni.

Za testove koji se odnose na DVB-T2 digitalni prijemnik ispitivane su i frekvencije u opsegu (174.928 – 239.200 [MHz]) sa propusnim opsegom od 1.7 MHz. U testovima se ispituju: centralne frekvencije, imunost uređaja na promenu centralne frekvencije za  $\pm 50$  kHz, različiti propusni opsezi za iste frekvencije i različiti modovi. Ispitivanje različitih modova podrazumeva podešavanje DVB-T/T2 modulatora na one parametre koji se zahtevaju u NorDig specifikacijama. U ovoj grupi postoji 70 testova koji su bili pogodni za automatizaciju, a automatizovano je njih 21.

Druga grupa testova odnosi se na pitanja vezana za fizičku arhitekturu. NorDig digitalni prijemnik treba da poseduje analogne i digitalne sprežne podsisteme. Ovi

testovi ispituju mogućnost prikazivanja različitih zvučnih i video formata u zavisnosti od sprežnog sistema koji se koristi. Za analogne signale koristi se CVBS, a za digitalne HDMI sprežni sistem. U ovoj grupi postoji ukupno 69 testova, 12 testova bilo je pogodno za automatizaciju, a njih 9 prošlo je kroz proces automatizacije.

Treća grupa testova odnosi se na pitanja vezana za sistemsku programsku podršku. Oni ispituju mogućnost digitalnog prijemnika da dekodira teletekst servise iz analognog i digitalnog signala, kao i mogućnost prikaza teleteksta i prevoda na ekranu. Takođe, ovi testovi ispituju postojanje servisne liste i mogućnost vraćanja uređaja na fabrička podešavanja. U ovoj grupi testova postoji ukupno 112 testova, od kojih je 14 bilo pogodno za automatizaciju, a automatizovano je njih 8.

Testovi koji su bili pogodni za automatizaciju su u tabeli 1 označeni plavom bojom. Neki od testova planiranih za automatizaciju su generalni i predstavljaju opis testova koji su u specifikaciji navedeni posle njih, tako da oni nisu automatizovani. Testovi koji se odnose na proveru mogućnosti dekodovanja digitalnog toka podataka definisanog NorDig specifikacijama ili proveru nekih osnovnih funkcionalnosti digitalnog prijemnika, takođe nisu automatizovani, i oni su napisani kao ručni testovi. U tom slučaju odluku o tome da li je test prošao ili ne, donosi osoba koja testira uređaj. U slučajevima kada nije bilo moguće da se automatski postave parametri koji su se zahtevali u testnoj proceduri, testovi su izvršavani za deo parametara za koje je to bilo moguće, a ostali su navedeni u komentaru. Ovakvi slučajevi vezani su za DVB-T2 modulator, koji ima ograničenja po pitanju postavljanja parametara transportnog toka i nije mogao da generiše signal za sve parametre koji su bili zahtevani u specifikaciji.

## 5. Python skripte

Automatske testne procedure u ovom radu implementirane su pomoću Python skriptnog jezika. To je objektno orijentisani, programski jezik višeg nivoa sa dinamičkom semantikom. Zbog svoje jednostavnosti olakšava razvoj aplikacija i koristi se kao skriptni jezik za povezivanje postojećih komponenti. Python testne skripte sadrže niz komandi/koraka preko kojih se, uz pomoć API biblioteka, šalju poruke uređajima koji čine sklop testnog okruženja. Python skripte praćene su odgovarajućim testom.

Svaki Python test u okviru ovog procesa automatizacije sadrži sledeće elemente:

1. zaglavlje, u kojem se nalaze informacije o testnom slučaju
2. uključivanje biblioteka i konstanti koje su korišćene u testu
3. glavnu funkciju runTest()

Informacije koje se nalaze u *zaglavljisu* su: naziv skripte, naziv testa i kratak opis testa.

Rukovanje fizičkim i logičkim uređajima realizovano je preko različitih *Python biblioteka*. Korišćene su sledeće biblioteke: *TEST\_CREATION\_API* i *CUSTOMER\_TC\_API*. U njima se nalaze funkcije koje su se koristile za lakše i brže kreiranje testova. *TEST\_CREATION\_API* sadrži funkcije koje su potrebne za izvršavanje testova metodom crne kutije, i ne mogu da se menjaju. *CUSTOMER\_TC\_API* sadrži funkcije koje mogu da se menjaju da bi se testiranje prilagodilo načinu rada uređaja koji se testira.

Funkcije koje su se koristile u ovim testovima jesu sledeće:

- ***initialize\_grabber*** – funkcija kojom se inicijalizuje uređaj za snimanje slike i zvuka
- ***deinitialize\_grabber*** – funkcija koja deinicijalizuje uređaj za snimanje slike i zvuka
- ***check\_initial\_DUT\_state*** – funkcija koja proverava da li je uređaj koji se testira inicijalizovan na odgovarajući način
- ***grabber\_start\_video\_source*** – funkcija koja selektuje video ulaz na uređaju za snimanje slika
- ***grabber\_stop\_video\_source*** – funkcija koja zaustavlja rad izabranog video ulaza
- ***grabber\_start\_audio\_source*** - funkcija koja selektuje zvučni ulaz na uređaju za snimanje zvuka
- ***grabber\_stop\_audio\_source*** – funkcija koja zaustavlja rad izabranog zvučnog ulaza
- ***stream\_play*** – funkcija koja postavlja parametre DVB-T modulatora i pokreće transportni niz
- ***stream\_play\_DVB\_T2*** – funkcija koja postavlja parametre DVB-T2 modulatora i pokreće transportni niz
- ***stream\_stop*** – funkcija koja zaustavlja rad DVB-T ili DVB-T2 modulatora
- ***send\_ir\_rc\_command*** – funkcija koja šalje IR komande pomoću emulatora daljinskog upravljača do uređaja koji se testira
- ***grab\_picture*** – funkcija koja snima sliku sa uređaja koji se testira
- ***record\_audio*** - funkcija koja snima zvuk sa uređaja koji se testira
- ***compare\_pictures*** – funkcija koja poređi testnu i referentnu sliku
- ***compare\_audio*** – funkcija koja poređi testni i referentni zvuk
- ***update\_test\_result*** – funkcija koja služi za ažuriranje rezultata testa
- ***write\_log\_to\_file*** – funkcija koja upisuje komentare u datoteku sa izveštajem o grešci
- ***add\_data\_to\_excel*** - funkcija za dodavanje elemenata u Excel tabelu
- ***save\_data\_to\_excel*** - funkcija za snimanje izmenjenog Excel dokumenta
- ***OCR\_recognize\_text*** – funkcija koja služi za prepoznavanje teksta koji se nalazi na testnoj slici pomoću OCR algoritma

- 
- ***fix\_string\_recognized\_by\_OCR*** - funkcija za korekciju niza karaktera prepoznatih OCR algoritmom
  - ***RepresentsInt*** – funkcija koja ispituje da li su simboli nekog niza brojevi
  - ***get\_av\_format\_info*** – funkcija koja analizira tip zvuka/slike prisutnog na selektovanoj sprezi uređaja za snimanje.

Neke od tih informacija jesu sledeće:

- tip zvučnog/video dekodovanja
- dužina i širina test slike (rezolucija)
- format ekrana
- broj ramova u sekundi
- način skeniranja

Prvi korak u *runTest()* funkciji jeste postavljanje inicijalne vrednosti rezultata i međurezultata testa, pri čemu se polazi od pretpostavke da uređaj ne radi dobro, pa se kao inicijalna vrednost rezultata testa postavlja vrednost NE PROLAZI.

Posle toga sledi *try-except* blok koji se koristi da bi se lakše rukovalo greškama i izuzecima koji mogu da se dese tokom izvršavanja testa. Prvi korak ovog bloka jeste inicijalizacija uređaja i postavljanje parametara za svaku novu iteraciju u testu. Od zahteva testa zavisi koji parametri će biti postavljeni kao početni.

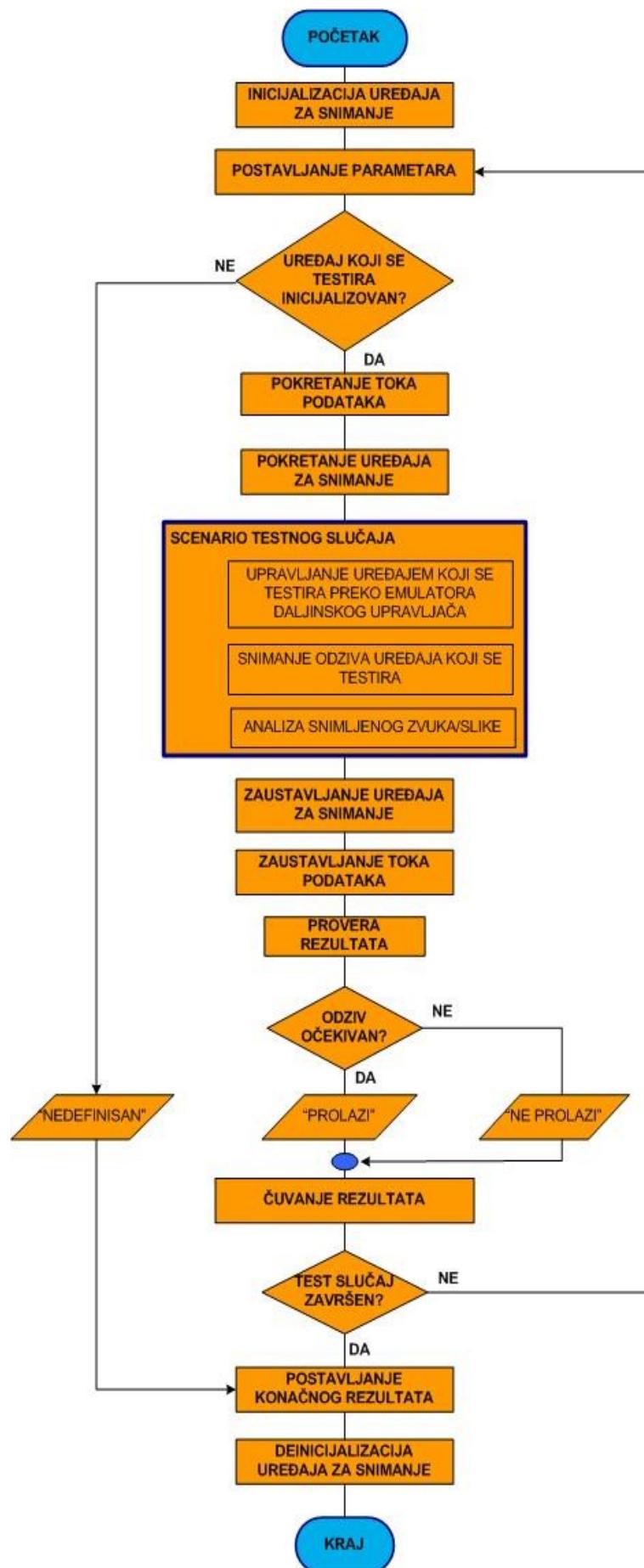
Ukoliko uređaji nisu dobro inicijalizovani, izvršavanje testa se zaustavlja i rezultat testa jeste NEDEFINISAN. Rezultat testa može biti NEDEFINISAN i ako postoji greška u kodu. U tom slučaju informacija o izuzetku koji se desio smešta se u odgovarajući datoteku. Tada se vrši analiza greške koja se desila, nakon čega se pristupa ispravci greške.

Ukoliko uređaji jesu dobro inicijalizovani, izvršenje testa se nastavlja pozivanjem funkcija. Koje funkcije se pozivaju zavisi od scenarija testnog slučaja.

Nakon poređenja rezultata dobijenih izvršavanjem testa sa očekivanim rezultatima, oni se smeštaju u Excel datoteku. Ukoliko očekivani rezultati jesu zadovoljeni, vrednost rezultata testa koja se upisuje u Excel datoteku jeste PROLAZI, u suprotnom je NE PROLAZI. Kada su sve iteracije završene, ukoliko je rezultat za samo jednu od njih NE PROLAZI, konačan rezultat testa jeste NE PROLAZI. Kao što je već navedeno, za svaku novu iteraciju u testu postavlja se nova kombinacija parametara generatora toka podataka.

Poslednji korak u Python testu je deinicijalizacija uređaja za snimanje, odnosno vraćanje ovog uređaja u početno stanje.

Uobičajeni koraci ovih Python testova prikazani su na slici 6, a primer jednog Python testa na slici 7. U pitanju je test *Task 7:2* koji ispituje usklađenost uređaja koji se testira sa brojem analognih sprežnih podsistema koje propisuje NorDig specifikacija. U slučaju analognog zvuka ispitivao se LINEIN sprežni podsistem, a u slučaju analogne slike CVBS sprežni podsistem.



Slika 6. Algoritam Python testova

```

##

# @test definition name: Task 7:2
# @test case name: Task 7:2
# @test case description: To verify if the IRD has at least one analog video and audio output interface.

import TEST_CREATION_API
import CUSTOMER_TC_API
import time

def runTest():

    # Set initial result of test execution to FAIL
    test_result_video = "FAIL"
    test_result_audio = "FAIL"
    final_test_result = "FAIL"

    # Flag is True if checking of single frequency fail
    single_fail_detected = False

    try:
        # Initialize grabber device
        TEST_CREATION_API.initialize_grabber()

        # Continue with test if grabber is properly initialized
        if (CUSTOMER_TC_API.check_initial_DUT_state(TEST_CREATION_API.VideoInterface.CVBS)):

            # Starting to grab video signal
            TEST_CREATION_API.grabber_start_video_source(TEST_CREATION_API.VideoInterface.CVBS, True)

            # Starting to play stream
            TEST_CREATION_API.stream_play("[GENERAL_DVBT]")
            time.sleep(5)

            # Grab picture
            TEST_CREATION_API.grab_picture("screen")

            # Stop grabbing video
            TEST_CREATION_API.grabber_stop_video_source()

            # Compare and update test results
            if      (TEST_CREATION_API.compare_pictures("File21",           "screen_0_0",           "[FULL_SCREEN]",

TEST_CREATION_API.DEFAULT_CVBS_VIDEO_THRESHOLD)):
                test_result_video = "PASS"
            else:
                test_result_video = "FAIL"
                single_fail_detected = True

            # Starting to grab audio signal
            TEST_CREATION_API.grabber_start_audio_source(TEST_CREATION_API.AudioInterface.LINEIN)

            # Record audio
            TEST_CREATION_API.record_audio("record")

```

```

# Stop playing stream
TEST_CREATION_API.stream_stop()

# Stop grabbing audio
TEST_CREATION_API.grabber_stop_audio_source()

# Compare and update test results
if (TEST_CREATION_API.compare_audio("File", "record_0", 2, "[ANALOG_AUDIO])):
    test_result_audio = "PASS"
else:
    test_result_audio = "FAIL"
    single_fail_detected = True

# Add and save data to excel file
CUSTOMER_TC_API.add_data_to_excel("Sheet1", str(3), str(2), str(test_result_video))
CUSTOMER_TC_API.add_data_to_excel("Sheet1", str(4), str(2), str(test_result_audio))
CUSTOMER_TC_API.save_data_to_excel("Task_7_2_results.xlsx")

# Compare and update final test results
if ((test_result_video == "FAIL") or (test_result_audio == "FAIL")):
    final_test_result = "FAIL"
else:
    final_test_result = "PASS"

else:
    # Set test result to INCONCLUSIVE if DUT is not initialized
    final_test_result = "INCONCLUSIVE"
    TEST_CREATION_API.write_log_to_file("DUT is not initialized!")

# Set test result variable to INCONCLUSIVE if exception is caught
except Exception as error:
    final_test_result = "INCONCLUSIVE"
    TEST_CREATION_API.write_log_to_file(error)

# Update test result with current state of test_result variable
if (final_test_result == "INCONCLUSIVE"):
    TEST_CREATION_API.update_test_result("INCONCLUSIVE")
else:
    if (single_fail_detected == False):
        TEST_CREATION_API.update_test_result("PASS")
    else:
        TEST_CREATION_API.update_test_result("FAIL")

# Deinitialize grabber device
TEST_CREATION_API.deinitialize_grabber()

```

Slika 7. Primer Python testa (Task 7:2)

## 6. Analiza rezultata

Neke od prednosti automatizacije testiranja jesu smanjenje vremena potrebnog za izvršavanje testova i izbacivanje faktora ljudske greške prilikom uzastopnog izvršavanja testova.

Ušteda u vremenu koja je postignuta u ovom procesu automatizacije može da se pokaže na sledećem primeru.

Za test *Task 3:35*, koji ispituje imunost uređaja na promenu centralne frekvencije za  $\pm 50$  kHz, automatsko izvršavanje testa ubrzava proces ispitivanja uređaja za 63,5 %. U testu je 6 različitih vrednosti nominalnih frekvencija. Podešavanje parametara digitalnog toka podataka za svaku frekvenciju i pretraga digitalnog toka podataka na uređaju koji se testira, prilikom ručnog testiranja trajalo je u proseku 48 sekundi, dok je za isti proces u automatskim testovima bilo potrebno u proseku 15 sekundi. Ukupno vreme potrebno da se test izvrši i da se dokumentuju dobijeni rezultati prikazano je u tabeli 2.

Tabela 2. Rezultati merenja vremena potrebnog za izvršenje testa (Task 3:35)

TIP IZVRŠAVANJA TESTA	VREME POTREBNO ZA IZVRŠENJE TESTA
Ručno	14 min 30 s
Automatski	5 min 50 s

Na osnovu merenja trajanja ručnog i automatskog izvršavanja istog testa i njihovog poređenja, može se zaključiti da automatizacija testova u velikoj meri

smanjuje vreme potrebno za testiranje. Automatizacija je posebno korisna za testove koji se odnose na DVB-T2 digitalni prijemnik. Za razliku od DVB-T modulacije, DVB-T2 modulacija obuhvata mnogo veći broj parametara koje je potrebno menjati sa ciljem sveobuhvatnog testiranja. Zbog toga je vreme potrebno za ručno podešavanje ovih parametara duže i veća je mogućnost greške osobe koja testira. Automatskim testiranjem se ovi problemi smanjuju na minimum.

Rezultati testova smeštaju se u bazu podataka i dostupni su za prikaz usklađenosti sa NorDig zahtevima. Za smeštaj međurezultata dobijenih tokom izvršavanja test procedure, korišćene su Excel datoteke, čijim se pregledom mogu dobiti detaljnije informacije o prolaznosti testne procedure. Primer jedne Excel datoteke prikazan je na slici 8.

Task\_3\_37\_results - Microsoft Excel

The screenshot shows an Excel spreadsheet titled "Task\_3\_37\_results". The ribbon menu includes Home, Insert, Page Layout, Formulas, Data, Review, and View. The main content consists of four tables:

- Table 1. FFT sizes** (Rows 13-32):
 

Constellation	Guard interval	Pilot pattern	Convolutional rate	FFT mode	Carrier mode	Result
QAM64	1/8	PP2	2/3	1K	Normal	FAIL
QAM64	1/8	PP2	2/3	2K	Normal	FAIL
QAM64	1/8	PP2	2/3	4K	Normal	PASS
QAM64	1/8	PP2	2/3	8K	Normal	PASS
QAM64	1/8	PP2	2/3	8K	Extended	FAIL
QAM256	1/8	PP2	2/3	16K	Normal	FAIL
QAM256	1/8	PP2	2/3	16K	Extended	FAIL
QAM256	1/8	PP2	3/4	32K	Normal	FAIL
QAM256	1/8	PP2	3/4	32K	Extended	FAIL
- Table 2. Constellation rotation** (Rows 33-42):
 

Constellation	Guard interval	Pilot pattern	Convolutional rate	FFT mode	QAMRotation	Result
QAM256	1/16	PP4	2/3	32K	Rotated	FAIL
QAM256	1/16	PP4	2/3	32K	Non-rotated	FAIL
- Table 3. Pilot patterns** (Rows 43-52):
 

Constellation	Guard interval	Convolutional rate	FFT mode	Pilot pattern	Result
QAM256	1/4	2/3	16K	PP1	FAIL
QAM256	1/8	3/4	32K	PP2	FAIL
QAM256	1/8	2/3	16K	PP3	FAIL
QAM256	1/16	2/3	32K	PP4	FAIL
QAM256	1/16	2/3	16K	PP5	FAIL
QAM256	1/32	3/5	32K	PP6	PASS
QAM256	1/128	2/3	32K	PP7	FAIL
QAM256	1/16	3/4	32K	PP8	FAIL
- Table 4. Guard intervals(Tg), constellations and code rates for allowed combinations of FFT sizes, pilot patterns and L1-ACE & TR PAPR** (Rows 53-62):
 

Modulation	ConvonvutionalRate	Tg = 1/128	Tg = 1/32	Tg = 1/16	Tg = 19/256	Tg = 1/8	Tg = 19/128	Tg = 1/4
QPSK	1/2	FAIL	PASS	PASS	FAIL	PASS	FAIL	PASS
QPSK	3/5	PASS	FAIL	PASS	PASS	FAIL	PASS	PASS
QPSK	2/3	PASS	PASS	PASS	PASS	FAIL	FAIL	PASS
QPSK	3/4	FAIL	PASS	PASS	PASS	PASS	FAIL	PASS
QPSK	4/5	PASS	PASS	PASS	PASS	PASS	PASS	PASS
QPSK	5/6	PASS	PASS	PASS	PASS	PASS	PASS	FAIL
QAM16	1/2	FAIL	FAIL	PASS	FAIL	FAIL	PASS	PASS
QAM16	3/5	FAIL	PASS	FAIL	FAIL	FAIL	PASS	FAIL
QAM16	2/3	FAIL	PASS	FAIL	FAIL	FAIL	FAIL	PASS
QAM16	3/4	FAIL	PASS	FAIL	FAIL	PASS	FAIL	FAIL
QAM16	4/5	PASS	PASS	FAIL	FAIL	PASS	PASS	FAIL
QAM16	5/6	FAIL	PASS	PASS	PASS	FAIL	FAIL	FAIL
QAM64	1/2	FAIL	FAIL	FAIL	PASS	PASS	PASS	PASS
QAM64	3/5	PASS	PASS	PASS	FAIL	FAIL	FAIL	FAIL
QAM64	2/3	PASS	PASS	FAIL	FAIL	FAIL	PASS	FAIL
QAM64	3/4	PASS	FAIL	FAIL	FAIL	PASS	FAIL	PASS
QAM64	4/5	PASS	PASS	FAIL	FAIL	PASS	FAIL	FAIL
QAM64	5/6	FAIL	FAIL	FAIL	PASS	PASS	PASS	PASS
QAM256	1/2	PASS	PASS	FAIL	FAIL	PASS	FAIL	FAIL
QAM256	3/5	PASS	PASS	PASS	PASS	FAIL	FAIL	FAIL
QAM256	2/3	FAIL	PASS	PASS	PASS	PASS	FAIL	FAIL
QAM256	3/4	PASS	PASS	PASS	PASS	PASS	FAIL	FAIL
QAM256	4/5	FAIL	PASS	FAIL	FAIL	PASS	PASS	PASS
QAM256	5/6	PASS	PASS	FAIL	PASS	PASS	PASS	FAIL

Slika 8. Primer Excel datoteke (Task 3:37)

## 7. Zaključak

Automatsko testiranje postaje neophodan deo procesa razvoja svakog softverskog sistema. Neki od osnovnih ciljeva automatskog testiranja jesu sledeći:

- redukovanje ručnih testnih aktivnosti
- omogućavanje fokusa inženjera na izvršavanje testova
- smanjenje faktora ljudske greške
- ponavljanje testnih procedura više puta

Rad opisuje proces automatizacije NorDig testnih procedura korišćenjem metode crne kutije i predstavlja uštedu vremena koja se postiže procesom automatizacije.

Neki od testova koji su automatizovani tokom procesa morali su više puta da se ponavljaju. Automatsko testiranje može da redukuje vreme izvršavanja testova koji se često ponavljaju sa nekoliko nedelja na nekoliko sati. U ovom procesu automatizacije to je posebno značajno za testove koji se odnose na DVB-T2 digitalni prijemnik, zbog velikog broja parametara DVB-T2 modulatora koji se menjaju za svaku novu iteraciju tokom izvršavanja testa. Pisanje automatskih testova zahteva dosta vremena, ali oni mogu da se izvršavaju više puta i njihovo izvršavanje je brže nego u slučaju ručnog izvršavanja, čime su smanjeni i troškovi razvoja softverskog sistema.

Takođe, jedna od prednosti automatskog testiranja jeste ta što se koraci napisani u testnim skriptama izvode precizno svaki put kada se testovi izvršavaju. Ljudi nisu precizni kao mašine i čak i najstrpljivija osoba koja izvršava testove može da napravi osnovne greške tokom ručnog testiranja.

Izvođenje ovih automatizovanih testova na drugim uređajima, koji treba da zadovolje NorDig zahteve, biće efikasnije, budući da već postoji napisane automatske skripte.

## 8. Literatura

- [1] *NorDig Unified Test Plan for Integrated Receiver Decoders for use in cable, satellite, terrestrial and IP-based networks, NorDig Unified Test plan, ver 2.4*  
Dostupno online:  
[http://www.nordig.org/pdf/NorDig-Unified\\_Test\\_plan\\_ver\\_2.4.pdf](http://www.nordig.org/pdf/NorDig-Unified_Test_plan_ver_2.4.pdf)
- [2] *NorDig Unified Requirements for Integrated Receiver Decoders for use in cable, satellite, terrestrial and IP-based networks, NorDig Unified ver 2.5.1*  
Dostupno online:  
[http://www.nordig.org/pdf/NorDig-Unified\\_ver\\_2.5.1.pdf](http://www.nordig.org/pdf/NorDig-Unified_ver_2.5.1.pdf)
- [3] *BBT Testing System*  
Dostupno online:  
<http://bbt.rt-rk.com/automated-testing-getting-started/>
- [4] *RT-Executor, Automated test development and excution tool*  
Dostupno online:  
<http://bbt.rt-rk.com/software/rt-executor/>
- [5] *Intent+, End-to-End requirement and test management system*  
Dostupno online:  
<http://bbt.rt-rk.com/software/intent/>
- [6] *Reference Creator*  
Dostupno online:  
<http://bbt.rt-rk.com/reference-creator/>