Razvoj programske podrške za nadzor obradnih strojeva i procesa

Župan, Luka

Master's thesis / Diplomski rad

2019

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:235:924546

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-04-25**

Repository / Repozitorij:

Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Luka Župan

Zagreb, 2019.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentor: Student:

Dr. sc. Tomislav Staroveški, dipl. ing. Luka Župan

Zagreb, 2019.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem se mentoru doc. dr. sc. Tomislavu Staroveškom na iznimnom strpljenju, korisnom savjetovanju i pomoći tijekom izrade praktičnog i teoretskog rada.

Zahvaljujem se obitelji i prijateljima na podršci u najtežim trenutcima studiranja kojih nije bilo malo.

A posebnu zahvalu uputio bih djevojci koja mi je bila najveća potpora tijekom velikog dijela studija, koja je bila uz mene u trenutcima slabosti i radosti.

Luka Župan

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE
Središnje povjerenstvo za završne i diplomske ispite
Povjerenstvo za diplomske radove studija strojarstva za smjerove:
proizvodno inženjerstvo, računalno inženjerstvo, industrijsko inženjerstvo i menadžment, inženjerstvo materijala te mehatronika i robotika

Sveučilište u Zagrebu Fakultet strojarstva i brodogradnje	
Datum: 17 -01- 2019	Prilog:
Klasa: 602-04/19-6/3	
Ur. broj: 15-1703-19-24	

DIPLOMSKI ZADATAK

Student: **LUKA ŽUPAN** Mat. br.: 0035193519

Naslov rada na hrvatskom jeziku: **Razvoj programske podrške za nadzor obradnih strojeva i procesa**

Naslov rada na engleskom jeziku: **Development of machine tools and process monitoring software**

Opis zadatka:

Sustavi za nadzor obradnih procesa nezaobilazan su čimbenik u razvoju suvremenih alatnih strojeva. Njihovom se integracijom omogućuje izravna ili posredna procjena stanja alata, čime je u konačnici moguće značajno podići razinu autonomnosti alatnih strojeva. Posredni sustavi nadzora od osobitog su značaja, jer je njima moguće procijeniti stanje alata u stvarnom vremenu bez prekidanja obradnog procesa. Među prvim koracima u razvoju takvih sustava može se ubrojiti odabir hardverske podrške za akviziciju i obradu signala vibracija i akustične emisije u stvarnom vremenu, kao i izrada odgovarajuće programske podrške. S obzirom na dostupnu mjernu opremu proizvođača National Instruments, u radu potrebno:

1. Opisati značajke mjernih stanica temeljenih na cRIO platformi.
2. Obarati odgovarajući višekanalni akvizicijski modul prikladan za akviziciju signala frekvencijskog područja do 300 kHz, te modul za okidanje početka i kraja snimanja.
3. Izraditi odgovarajuću programsku podršku za akviziciju signala i pohranu mjernih podataka.
4. Dati zaključke rada.

U radu je potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:
15. studenog 2018.

Rok predaje rada:
17. siječnja 2019.

Predviđeni datum obrane:
23. siječnja 2019.
24. siječnja 2019.
25. siječnja 2019.

Zadatak zadao:

doc. dr. sc. Tomislav Staroveški

Predsjednica Povjerenstva:

prof. dr. sc. Biserka Runje

SADRŽAJ

SADRŽAJ I
POPIS SLIKA III
POPIS TABLICA V
SAŽETAK VI
SUMMARY VII
1. UVOD 1
2. LABVIEW 2
2.1. Korisničko sučelje 3
2.2. Blok dijagram 4
2.3. Pokretanje LabVIEW softvera 5
2.3.1. Scan Interface (CompactRIO Scan Mode) 5
2.3.2. LabVIEW FPGA Interface Mode 6
3. COMPACT RIO (cRIO) 7
3.1. RT kontroler 8
3.2. Rekonfigurabilno FPGA postolje 10
3.3. FPGA 11
3.4. Industrijski I/O Moduli 14
4. PROGRAMSKA PODRŠKA 15
4.1. Modul NI 9223 C serije 17
4.2. Izrada akvizicijskog programa na Compact RIO platformi 19
4.3. VI na „Real time“ uređaju 25
4.3.1. Izrada kanala 27
4.3.2. Podešavanje vremenske oznake 28
4.3.3. Memorijski međuspremnik 30
4.3.4. Početak akvizicije 32
4.3.5. Čitanje podataka 34
4.3.6. TDMS 37
5. REZULTATI 39
6. ZAKLJUČAK 43
7. LITERATURA 44

PRILOZI.....45

POPIS SLIKA

Slika 1. Korisničko sučelje [1] 3
Slika 2. Blok dijagram 4
Slika 3. Prozor za odabir načina programiranja 5
Slika 4. CompactRIO stanica [3] 7
Slika 5. Struktura izmjenjivog ugradbenog uređaja 8
Slika 6. NI cRIO-9015 Real-Time kontroler [4] 8
Slika 7. Grafičko objašnjenje Jitter-a 9
Slika 8. Izmjenjivo FPGA postolje CompactRIO mjerne stanice [4] 10
Slika 9. Dijelovi FPGA čipa [6] 11
Slika 10. Simbol D bistabila [6] 12
Slika 11. Simbol LookUP tablice [6] 12
Slika 12. Logička I operacija [6] 13
Slika 13. Shematski crtež implementacije množenja dva 4-bitna broja [6] 13
Slika 14. Neki od 50 različitih modula Compact RIO [4] 14
Slika 15. Postavke mjerne stanice u NI-MAX-u 15
Slika 16. Dodavanje NI cRIO 9067 u LabVIEW projekt 16
Slika 17. Modul NI 9223 [7] 17
Slika 18. Raspored uzlaza modula NI 9223 [7] 18
Slika 19. Struktura ravnog slijeda 19
Slika 20. Protok podataka 20
Slika 21. FPGA VI 21
Slika 22. Prikupljanje jedne točke po kanalu 22
Slika 23. Postavljanje/mjerenje brzine petlje 22
Slika 24. Provjera grešaka i uvjeti za zaustavljanje aplikacije 23
Slika 25. Korisničko sučelje FPGA VI-a 24
Slika 26. Real Time VI 26
Slika 27. Channel configuration blok dijagram 27
Slika 28. Channel configuration front panel 28
Slika 29. Timing configuration blok dijagram 29
Slika 30. Timing configuration Front panel 30

Slika 31. Buffer configuration blok dijagram 31
Slika 32. Buffer configuration front panel 32
Slika 33. Start Acquisition blok dijagram 33
Slika 34. Start Acquisition Front panel 34
Slika 35. Read Data blok dijagram 35
Slika 36. Read Data front panel 36
Slika 37. Svaka TDMS datoteka zapisana u LabVIEW sadrži opisne informacije o razinama [9] 38
Slika 38. Korisničko sučelje VI-a na RT uređaju 39
Slika 39. Dijelovi sustava 40
Slika 40. Struktura TDMS datoteke 40
Slika 41. Tablica uzoraka 41
Slika 42. Graf uzoraka 42

POPIS TABLICA

Tablica 1. Tablica istine za AND funkciju 13
Tablica 2. Ulazne karakteristike [7] 17
Tablica 3. Prikaz nekih opcija za pohranu 37

SAŽETAK

Suvremeni trendovi u razvoju alatnih strojeva usmjereni su integraciji većeg broja različitih senzora u cilju povećanja autonomije obradnih sustava. Time se neminovno uključuju senzori vibracija i akustične emisije, čiji se frekvencijski spektar može kretati i do 1 MHz. Standardni akvizicijski moduli dostupni u sklopu upravljačkih sustava alatnih strojeva nisu prikladni za obradu takvih signala zbog neodgovarajuće niske frekvencije uzorkovanja. Stoga je cilj ovog rada razvoj sustava za akviziciju i obradu signala frekvencijom uzorkovanja do 1MS/s na 4 kanala istovremeno u stvarnom vremenu, kao i izrada odgovarajuće programske podrške. Sustav je realiziran primjenom LabVIEW okruženja i opreme koja se sastoji od mjerne stanice CompactRIO-9067 i modula NI 9223 C serije. Kod se sastoji od 2 dijela, od kojih se jedan dio koda se izvršava na RT uređaju, a drugi dio većinom na FPGA.

Ključne riječi: CompactRIO, FPGA, akvizicija signala, stvarno vrijeme, Virtualni instrument

SUMMARY

Contemporary trends in the development of machine tools are focused on the integration of a number of different sensors in order to increase the autonomy of machining systems. This necessarily involves utilisation of vibration and acoustic emission sensors, whose frequency spectrum can range up to 1 MHz. Standard acquisition modules available as a part of the machine tools control systems are not suitable for processing such signals due to inadequately low sampling frequency. Therefore, the purpose of this paper is to develop a data acquisition and processing system using sampling frequency up to 1MS/s on 4 simultaneous channels in real time, as well as making appropriate software support. The system is implemented using LabVIEW environment and equipment consisting of the CompactRIO-9067 and the module NI 9223 C series.

Key words: CompactRIO, FPGA, Data aquisition, Real time, Virtual instrument

1. UVOD

Pri razvoju suvremenih alatnih strojeva neophodni su sustavi za nadzor obradnih procesa. Njihovom integracijom moguća je izravna ili posredna procjena stanja alata, čime je moguće značajno podići razinu autonomnosti alatnih strojeva. Posredni sustavi nadzora vrlo su značajni, jer njihovom uporabom omogućuje procjena stanja alata u stvarnom vremenu bez prekidanja obradnog procesa. Njihova integracija nije jednostavna jer se u takvim sustavima koriste senzori visokog frekvencijskog spektra kao što su senzori vibracija i akustične emisije. Frekvencijski spektar takvih senzora može se kretati i do 1MHz, a za akviziciju takvih signala potrebna je visoka frekvencija uzorkovanja koju sa standardnim akvizicijskim modulima nije moguće postići. Stoga je cilj ovog rada razvoj sustava za akviziciju i obradu signala frekvencijom uzorkovanja do 1 MS/s.

S obzirom na dostupnost opreme proizvođača National Instruments u radu je potrebno opisati značajke mjernih stanica temeljenih na CompactRIO platformi te izraditi odgovarajuću programsku podršku za akviziciju mjernih podataka na odabranom modulu i pohranu istih.

2. LABVIEW

LabVIEW, skraćeno od „Laboratory Virtual Instrument Engineering Workbench“, je platforma, mjerni softverski paket i razvojno okruženje za grafičko programiranje razvijen od poznate tvrtke National Instruments (NI). Programiranje se izvodi spajajući funkcionalne blokove vodovima kroz koje prolaze podaci te se po tome razlikuje od suvremenog programiranja kao što su C, C++, ili Java, u kojem se programira unošenjem tekstualnih kodova. Programski jezik na kojem se zasniva LabVIEW naziva se „G“ od Graphical. [1]

Program kreiran u LabVIEW-u sastoji se od jednog ili više „virtualnih instrumenata“ (VI). Izgledom i načinom rada oponašaju stvarne instrumente pa se zato tako nazivaju. Softver virtualnog instrumenta razvija se pomoću programskih alata koji rade u grafičkom okruženju, kao alternativa klasičnom programiranju. [1]

LabVIEW nudi veću fleksibilnost od standardnih laboratorijskih instrumenata jer se temelji na softveru. Funkcionalnost instrumenta definira korisnik, a ne proizvođač instrumenta. Koristeći LabVIEW, korisnik stvara točno onu vrstu virtualnog instrumenta koji je potreban, a pritom je puno jeftiniji od standardnih instrumenata koji se koriste. VI se može prilagoditi svakoj potrebnoj funkciji.

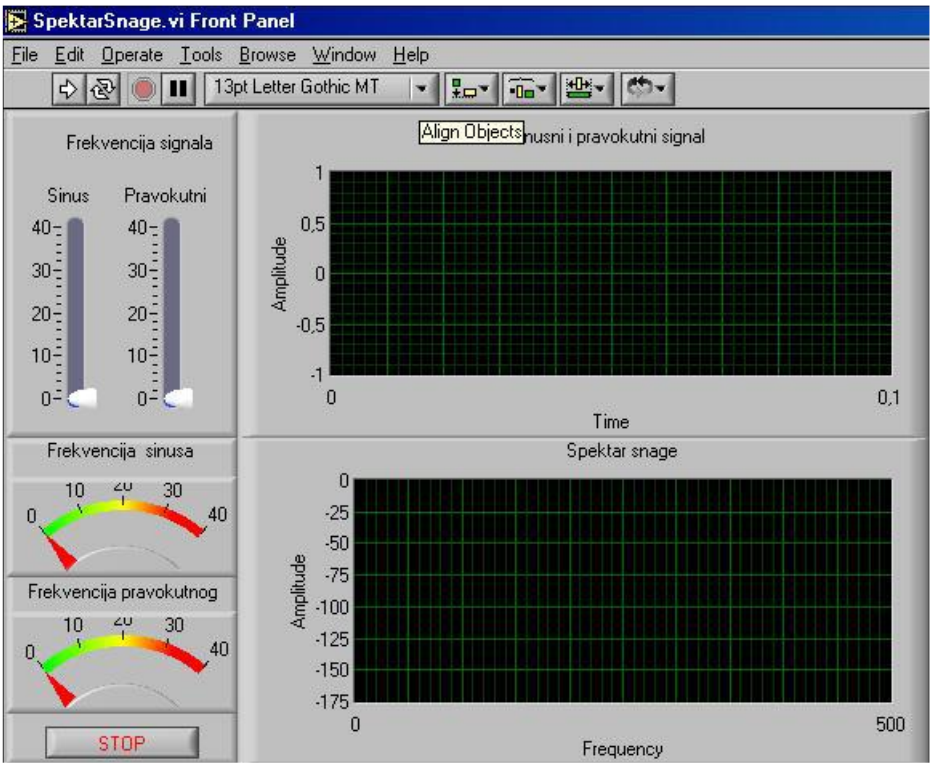
Način na koji se izvršava program u softveru LabVIEW određeno je strukturom blok dijagrama koji su linijama povezani. Svaki blok ima svoju funkciju i njihovim povezivanjem stvara se logika izvršavanja. Linijama se kodiraju veze između blokova odnosno one su vodovi kroz koje prolaze vrijednosti od funkcije do funkcije. [2]

VI se sastoji od dva glavna dijela:

- Korisničkog sučelja (engl. Front panel)
- Blok dijagrama (engl. Block diagram)

2.1. Korisničko sučelje

Preko Korisničkog sučelja korisnik komunicira sa VI-em, koristi se za podešavanje vrijednosti, unos podataka i za prikazivanje obrađenih podataka odnosno izlaza. Tako je nazvana zato što simulira prednju ploču fizičkog instrumenta. Korisničko sučelje može sadržavati pokazivače, kontrolnike, indikatore, testere, grafove i mnoge druge elemente koji predstavljaju ulazne i izlazne funkcije kao što je prikazano na slici 1. Podaci se mogu upisati pomoću miša i tipkovnice te se rezultati pomoću napisanog programa mogu vidjeti na ekranu. [1]

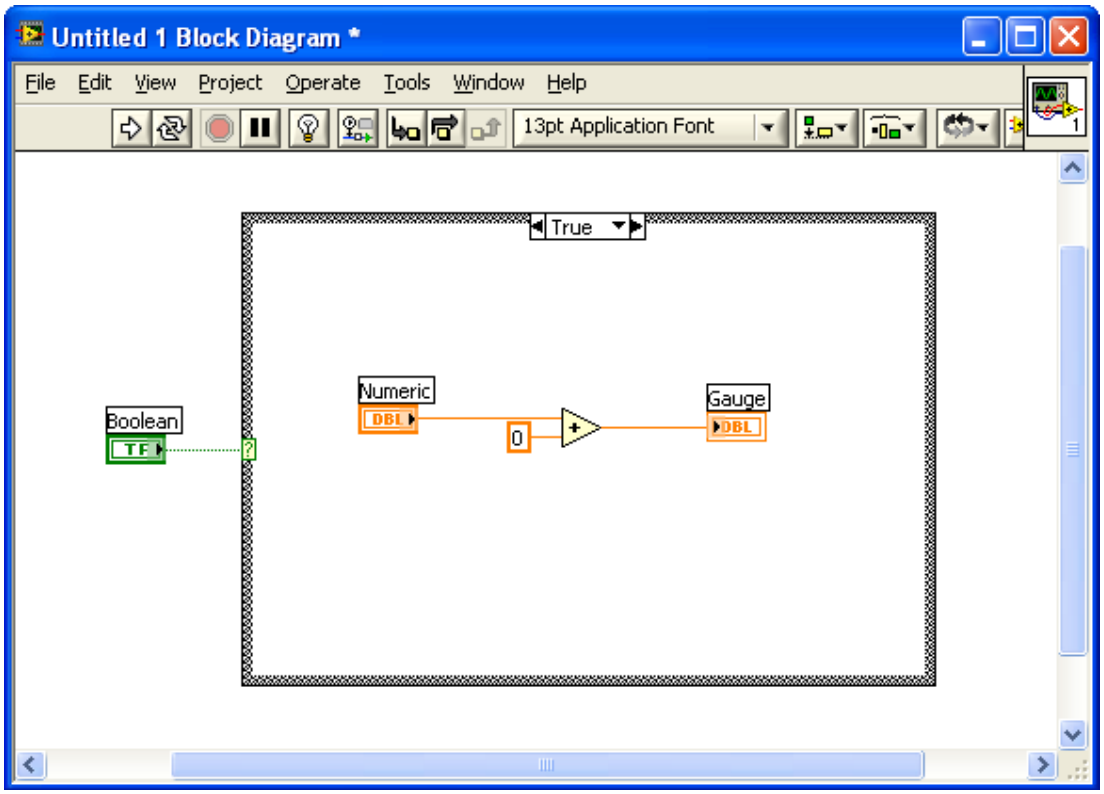


Slika 1. Korisničko sučelje [1]

Alatna traka prednje ploče sadrži komandne alate i statusne indikatore koji se koriste za pokretanje/zaustavljanje programa i otklanjanje grešaka debugging.

2.2. Blok dijagram

Blok dijagram predstavlja pozadinu korisničkog sučelja. Prozor blok dijagrama sadrži grafički izvorni kod VI-a, pa se za može reći da je dijagram blok jezgra svakog VI-a. Blok diagram je tzv. "kod" VI-a koji se "piše" slaganjem raznih blokova gdje svaki blok predstavlja jednu funkciju. Primjer jednog jednostavnog blok dijagrama prikazan je na slici 2. [1]

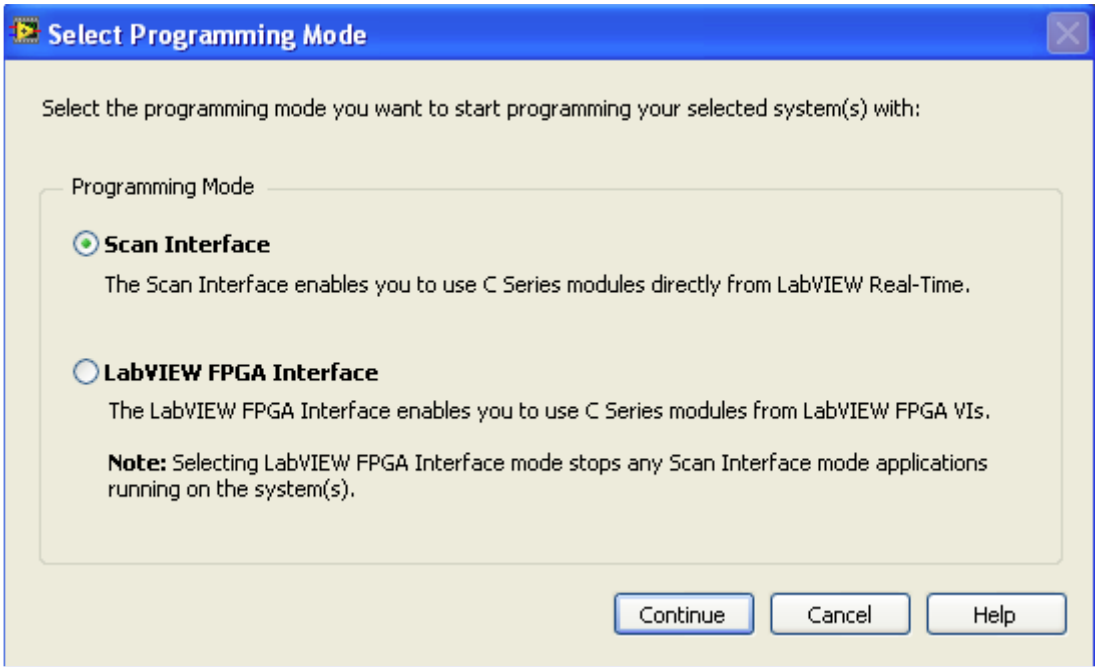


Slika 2. Blok dijagram

Ideja blok dijagrama je odvojiti grafički izvorni kod od korisničkog sučelja na logičan i jednostavan način. Objekti korisničkog sučelja VI-a pojavljuju se kao blokovi na blok dijagramu. Oni odražavaju promjene na odgovarajućim predmetima na korisničkom sučelju VI-a i obratno.

2.3. Pokretanje LabVIEW softvera

Ako se na računalu koje se koristi za programiranje nalazi LabVIEW Real Time i LabVIEW FPGA, prilikom dodavanja CompactRIO kontrolera u LabVIEW projekt, pojavljuju se dvije opcije za programiranja „The Scan Interface“ ili „LabVIEW FPGA Interface“ kao što je prikazano na slici 3.



Slika 3. Prozor za odabir načina programiranja

2.3.1. Scan Interface (CompactRIO Scan Mode)

Ova opcija omogućuje programiranje procesora CompactRIO sustava te pristup ulazima i izlazima bez potrebe za programiranjem FPGA. U ovom načinu rada, NI daje unaprijed definirani FPGA čip koji povremeno skenira I/O i smješta ga u memorijsku mapu. „CompactRIO Scan Mode“ dovoljan je za sustave koji se izvršavaju na frekvencijama do 1 kHz.

2.3.2. LabVIEW FPGA Interface Mode

Ova opcija omogućava direktno programiranje FPGA čipa u stvarnom vremenu, postižući svojstva koja obično zahtijevaju prilagođeni hardver. Koristeći LabVIEW FPGA, moguće je implementirati različite vrste logičkih sklopova prikladnih za izvršavanje u preciznim vremenskim intervalima te visokobrzinsku obradu i analizu kao i mnoge druge aplikacije.

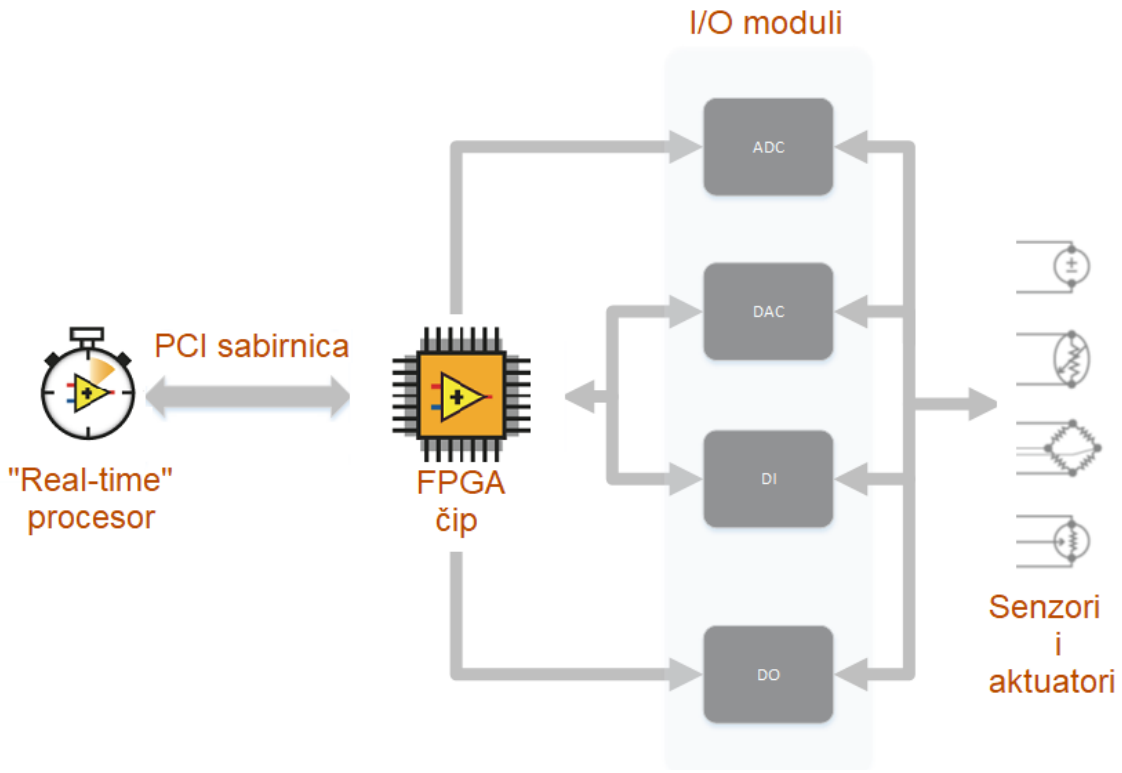
3. COMPACT RIO (cRIO)

Za akviziciju signala frekvencijskog područja do 300kHz upotrebljena je dostupna oprema, odnosno CompactRIO platforma tvrtke National Instruments. CompactRIO je prilagodljivi kontrolni i akvizicijski sustav za izradu aplikacija koje zahtijevaju visoka svojstva i pouzdanost. Sastoji od tri komponente kao što je prikazano na slici 4. [4]

- Operacijski sustav u stvarnom vremenu (engl. Real time operating system, RTOS)
- Rekonfigurabilno FPGA postolje
- industrijski I/O moduli koji se biraju prema aplikaciji.

Slika 4. CompactRIO stanica [3]

„Real time“ (RT) sustav omogućava da se kod izvršava u preciznim vremenskim intervalima, dok se FPGA ističe u manjim zadacima koji zahtijevaju visokobrzinsku logiku i precizno vremensko okidanje logičkih funkcija. CompactRIO aplikacije često sadrže „human machine interface“ (HMI), koje korisniku pružaju grafičko sučelje „graphical user interface“ (GUI) za prikazivanje stanja sustava i postavljanje radnih parametara. GUI se najčešće koristi u sklopu dizajniranja korisničkog sučelja na monitoru PC-a. Na slici 5. prikazana je struktura CompactRIO-a. [4]



Slika 5. Struktura izmjenjivog ugradbenog uređaja

3.1. RT kontroler

RT kontroler prikazan na slici 6. u sebi sadrži industrijski procesor koji pouzdano izvršava aplikacije u stvarnom vremenu, te također nudi mogućnost višeprocesorskog izvršavanja, praćenje rada, zapisivanje podataka i komunikaciju s okolinom. Dodatne opcije uključuju sat, brojila, dvostruka Ethernet sučelja, ugrađeni USB i RS232 ulaz. Količina memorije varira od modela do modela. [4]

Slika 6. NI cRIO-9015 Real-Time kontroler [4]

Operacijski sustav (OS) je odgovoran za upravljanje hardverskim resursima računala i za upravljanje aplikacijama koje se izvode na računalu. RTOS izvršava procese u stvarnom vremenu i visokim stupnjem pouzdanosti. To može biti posebno važno u mjernim i automatizacijskim sustavima gdje je zastoj ili kašnjenje programa može izazvati sigurnosnu opasnost. Operacijski sustav mora imati poznato maksimalno vrijeme za svaku od kritičnih operacija koje obavlja (ili barem u najvećoj mogućoj mjeri zajamčeno), da bi se moglo smatrati da se izvršava u stvarnom vremenu. Neke od tih operacija uključuju pozive OS-a i upravljanje prekidima (engl. Interrupt). Operacijski sustavi koji mogu jamčiti maksimalno vrijeme za određene operacije obično se nazivaju "hard real-time", dok se operativni sustavi koji mogu jamčiti većinu vremena odnosno postoji određena greška i odstupanje, nazivaju "soft real-time".[5] U literaturi se pri opisivanju svojstava RTOS-a spominje pojam „Jitter“, tj. maksimalno odstupanje u odnosu na zadano vrijeme prilikom izvršavanja. Ona se javlja prilikom višestrukog izvođenja programa ili petlje. Na slici 7. prikazano je kako svaka iteracija petlje mora imati približno jednako vrijeme izvršavanja.

Slika 7. Grafičko objašnjenje Jitter-a

3.2. Rekonfigurabilno FPGA postolje

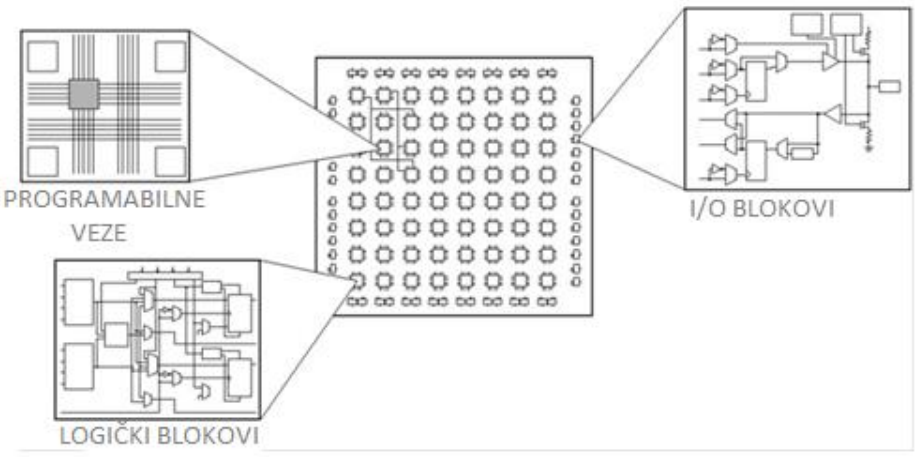
Rekonfigurabilno FPGA postolje prikazano na slici 8. je središte cijelog sustava. Postolje je direktno povezano s ulazno izlaznim (engl. Input/Output - I/O) modulima što omogućuje vrlo učinkoviti pristup ulazno izlaznim krugovima svakog od modula i mogućnost njihove sinkronizacije. Prema zadanim postavkama, FPGA automatski komunicira s I/O modulima te pruža deterministične I/O prema procesoru. Zbog visoke brzine FPGA postolje često se koristi u svrhu kreiranja kontrolnih sustava koji sadrže visokobrzinske ulaze i izlaze, regulacijske krugove, ili filtriranje signala. [4]

Slika 8. Izmjenjivo FPGA postolje CompactRIO mjerne stanice [4]

Koristeći FPGA, jedno postolje može izvršavati više od dvadeset analognih „proportional integral derivative“ (PID) kontrolnih petlji zajedno na frekvenciji od 100kHz što ovisi o vrsti, odnosno kapacitetu FPGA čipa. Na slici 8. prikazano je Rekonfigurabilno FPGA postolje. [4]

3.3. FPGA

FPGA (engl. Field programmable gate array) je integrirani krug koji se sastoji od reprogramibilnih logičkih blokova. Mogu se reprogramirati po potrebi za obavljanje različitih operacija za razliku od integriranih sklopova koji služe da bi obavljali samo jednu određenu funkciju. Ross Freeman, suosnivač Xilinxa, ih je osmislio 1985. godine. Kada je FPGA programiran, unutarnja struktura povezana je na način na koji je korisnik postavio u softveru. FPGA koristi namjenski hardver za logičku obradu i nema operativni sustav. Jedna od mnogobrojnih prednosti FPGA je to što može izvršavati više operacija paralelno, uz to sadrži funkcije koje mu omogućavaju rad pri vrlo visokim brzinama. Također, više kontrolnih petlji mogu se izvršavati pri različitim brzinama na jednom FPGA uređaju. Izvedba jednog djela aplikacije nema utjecaja na dodavanje nekih dodatnih operacija. Kao nedostatak navodi se mala količina resursa na raspolaganju. FPGA može zamijeniti tisuće različitih komponenti konstruirajući mnogobrojne logičke jedinice s čipom koji sadrži jedan integrirani strujni krug. Unutrašnjost FPGA čipa sastoji se od mreže rekonfigurabilnih logičkih blokova (engl. Configurable Logic Blocks – CLB-s), funkcijskih logičkih blokova. Signali su usmjereni programabilnim međusobno spojenim prekidačima i vodovima unutar FPGA mreže kao što se može vidjeti na slici 9. [6]

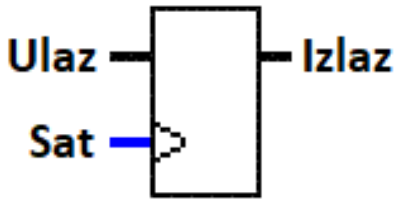


Slika 9. Dijelovi FPGA čipa [6]

Programabilni logički blokovi (engl. Configurabile logic blocks – CLB-s) su osnovna logička jedinica svakog FPGA. Ponekad se također nazivaju "Slice-ovi" ili "logičke

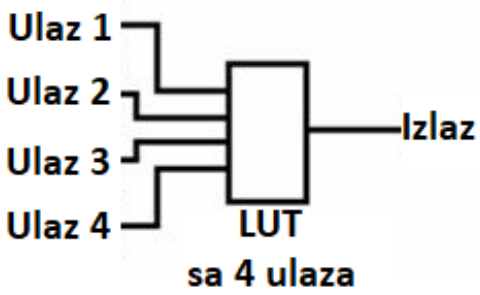
ćelije". CLB-i su sastavljeni od dvije osnovne komponente: D bistabili (eng. Flip-Flopovi) i pregledne tablice (eng. LookUp Table - LUTs). Razni FPGA skupovi razlikuju se po načinu na koji su D bistabili i LUT-ovi strukturirani zajedno. [6]

Nizom D bistabila tvore se strukture registara pomaka (eng. Shift register). Koriste se za sinkronizaciju logike i spremanje logičkih stanja između dva takta FPGA procesora. Pri svakom taktu procesora, D bistabil prima na svom ulazu logičku 1 ili 0 (True ili False) i sprema vrijednost do idućeg takta. Na slici 10. prikazan je simbol D bistabila. [6]



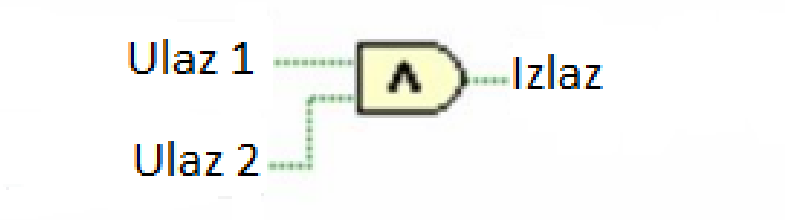
Slika 10. Simbol D bistabila [6]

Programabilni logički blokovi sadrže veliki broj logičkih nizova u RAM memoriji u obliku LUT. Sva kombinatorna logika FPGA procesora implementirana je u obliku skupa tablica istine u memoriji LUT. Simbol LookUP tablice prikazan je na slici 11.[6]



Slika 11. Simbol LookUP tablice [6]

Kao objašnjenje čemu služe LookUp tablice uzet je primjer logičkih I vrata. Umjesto da svaki puta FPGA iznova računa izlaz, poziva se na već predefiniranu funkciju LookUp tablice i odabere izlaz. Takvih funkcija programu ima puno te se na svakoj uštedi pomalo resursa i tako se ubrzava cijeli sustav. Primjer je prikazan na slici 12. i tablici 1. [6]

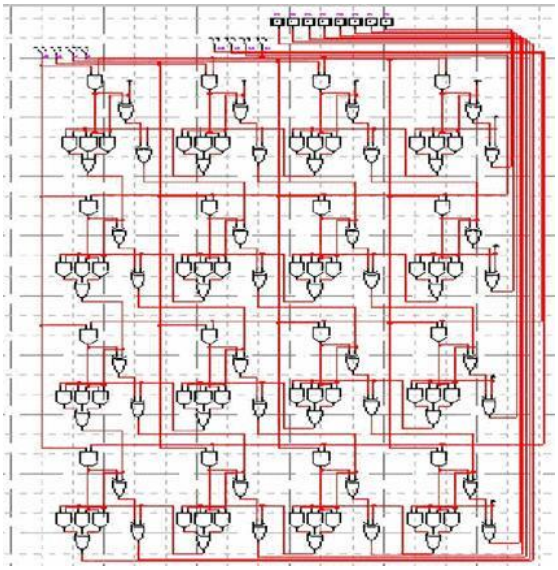


Slika 12. Logička I operacija [6]

Tablica 1. Tablica istine za AND funkciju

Ulaz 1	Ulaz 2	Izlaz
0	0	0
0	1	0
1	0	0
1	1	1

Naočigled jednostavne operacije kao što je množenje dva broja je vrlo komplicirano implementirati u digitalnom obliku. Kao primjer uzeto je množenje dva 4-bitna broja u digitalnom obliku kao što je prikazano na slici 13.



Slika 13. Shematski crtež implementacije množenja dva 4-bitna broja [6]

Teško je i zamisliti kako bi to izgledalo kad se množe dva 32-bitna broja što bi zahtijevalo više od 2000 operacija. U tu svrhu FPGA koristi multiplikatore i unaprijed izgrađene blokove za obradu (engl. DSP Sliceove) koji su zapravo predefinirani sklopovi multiplikatora zbog uštede preglednih tablica i funkcija D bistabila kod obavljanja matematičkih operacija. Blokovi RAM memorije, ugrađeni u FPGA čip, korisni su za spremanje skupova podataka ili prolaznih vrijednosti između paralelnih zadataka. [6]

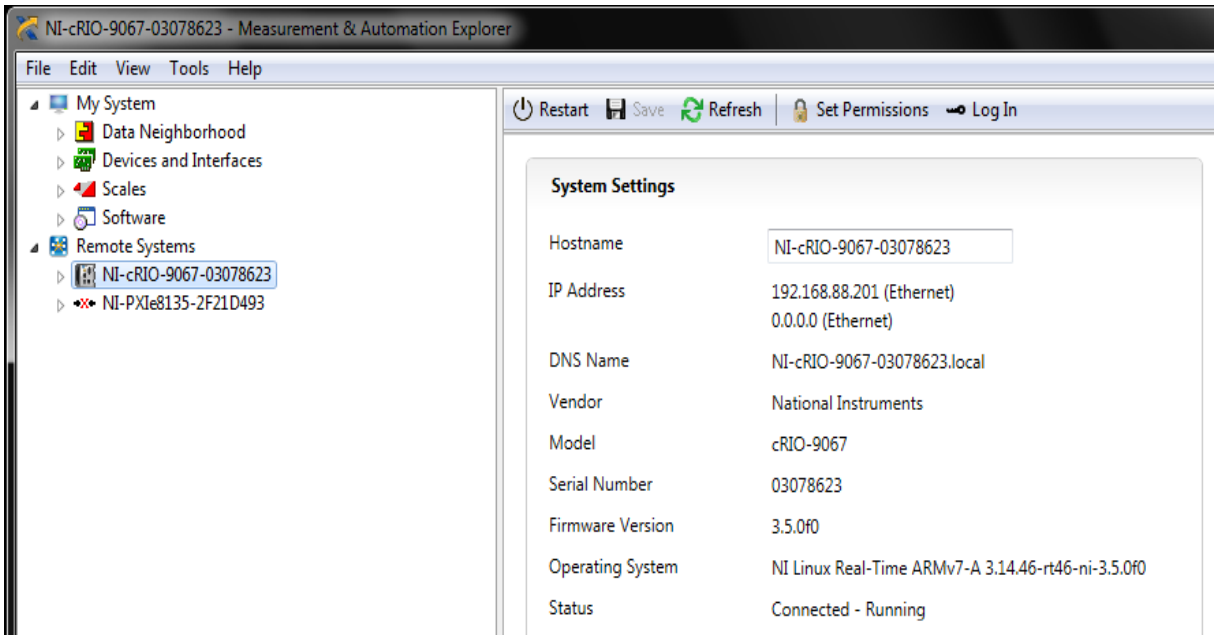
3.4. Industrijski I/O Moduli

Industrijski I/O moduli podijeljeni su na ulazne, izlazne i module koji sadrže kombinaciju ulaznih i ulaznih kanala, a druga podjela je na analogne i digitalne. Svaki model ima različitu funkciju, a postoji više od 50 različitih vrsta koji se koriste. Postoje moduli za akviziciju signala s različitih vrsta senzora, moduli za generiranje digitalnih ili analognih signala te moduli za komunikaciju i mnogi drugi. Na slici 14. prikazani su neki od modula tvrtke National Instruments.

Slika 14. Neki od 50 različitih modula Compact RIO [4]

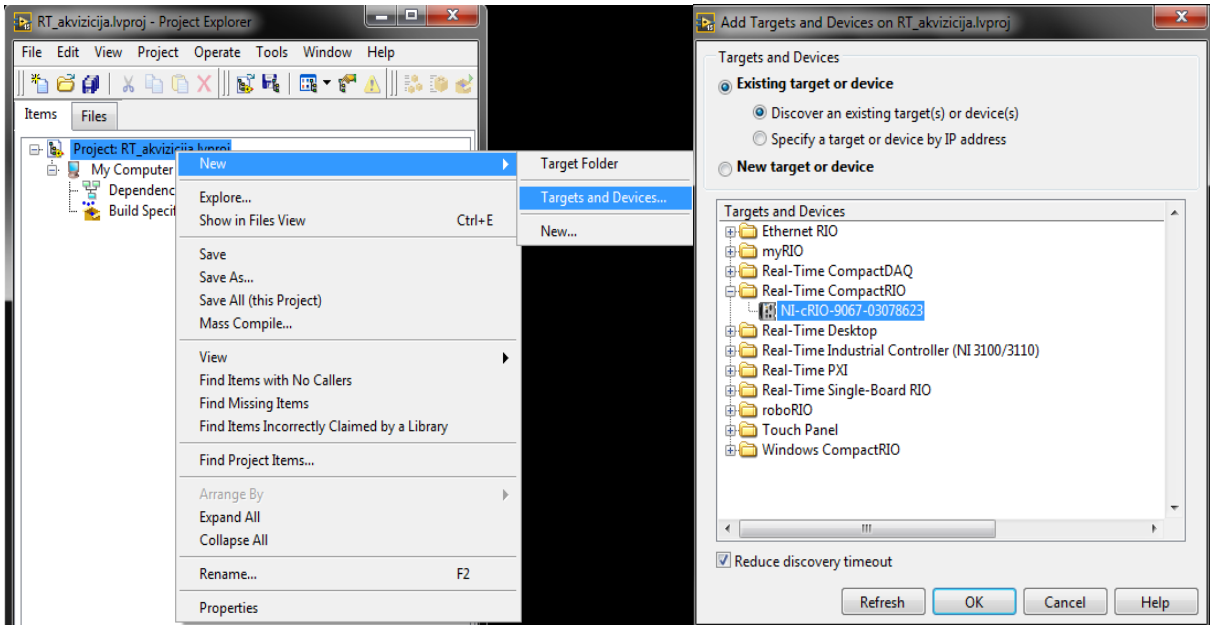
4. PROGRAMSKA PODRŠKA

Parametriranje mjerne stanice CompactRIO 9067 izvršeno je u programskom paketu NI Measurement & Automation Explorer (NI MAX). Mjerna stanica spaja se s računalom putem Ethernet sučelja kako bi se konfigurirala IP adresa mjerne stanice kao što se može vidjeti na slici 15. Mjerna stanica se odabire u stablu na slici 29. unutar mape *Remote Systems* nakon što je učitana.



Slika 15. Postavke mjerne stanice u NI-MAX-u

Drugi korak u izradi programske podrške u softveru LabVIEW 2015 je kreiranje novog projekta. U ovom slučaju naziv je *RT_akvizicija.lvproj*. Nakon toga potrebno je u projekt dodati mjernu stanicu, a taj postupak prikazuje slika 16.



Slika 16. Dodavanje NI cRIO 9067 u LabVIEW projekt

Desnim klikom na projekt otvara se izbornik i potom se odabire New -> Targets and Devices. LabVIEW automatski prepoznaje mjernu stanicu CompactRIO ako je spojena na PC što se može vidjeti također na slici 15. desno.

4.1. Modul NI 9223 C serije

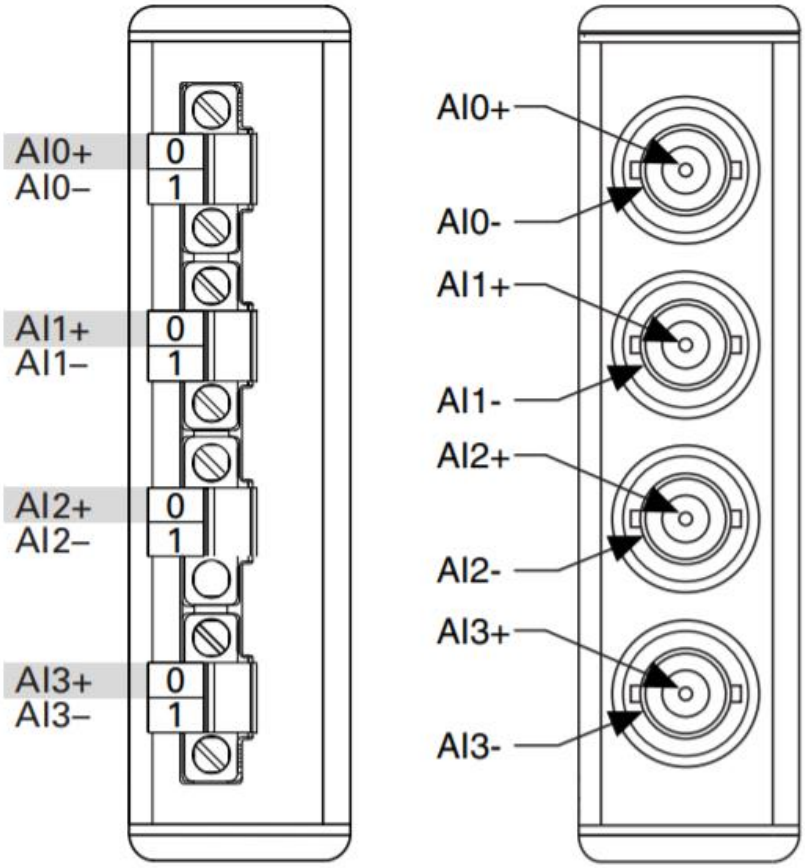
NI 9223 modul C serije je visoko brzinski, simultani modul za korištenje u bilo kojem CompactRIO postolju. Brzina uzorkovanja do 1MS/s i simultane analogno-digitalne pretvornike (ADCs). Ovaj modul je dobro prilagođen za aplikacije kao što su mjerenje vibracija i akustične emisije, balistička ispitivanja, ispitivanje udaraca te ispitivanje udarnih valova itd. Na slici 17 prikazan je modul NI 9223. Može imati BNC (engl. Bayonet Neill–Concelman) priključke ili povezivanje s vijčanim priključkom. Na tablici 2. mogu se vidjeti ulazne karakteristike modula NI 9223 C. [7]

Slika 17. Modul NI 9223 [7]

Tablica 2. Ulazne karakteristike [7]

Broj kanala	4 analogna ulazna kanala
ADC rezolucija	16 bit-ova
Vrsta ADC	(SAR) Successive approximation register
Ulazni napon	+/-10.5 V
Prenaponska zaštita	+/-30 V
Maksimalna brzina uzorkovanja	1MS/s

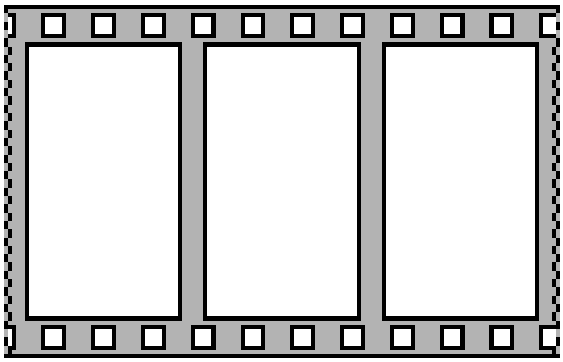
Na slici 18. prikazan je raspored ulaza za oba dva načina povezivanja.



Slika 18. Raspored uzlaza modula NI 9223 [7]

4.2. Izrada akvizicijskog programa na Compact RIO platformi

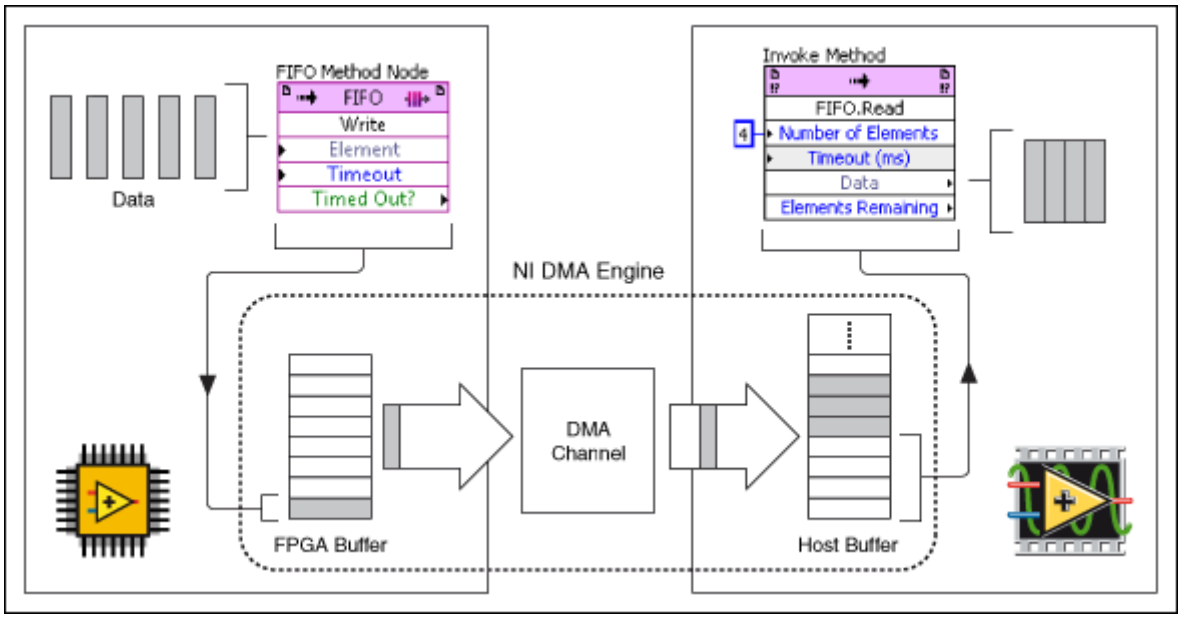
Unutar projekta *RT_akvizicija.lvproj.* stvara se glavni dio koda pod nazivom FPGA.vi gdje se vrši sama akvizicija. Ekstenzija .vi je skraćenica za već spomenuti virtualni instrument, a zapravo se radi o programu ili potprogramu (engl. subVI). U ovom programu sadržan je blok dijagram glavnog djela programa i potrebni potprogrami. Kod je podijeljen u četiri dijela koji se izvršavaju u nizu tako da kad jedan dio završi, nakon njega kreće sljedeći dio u izvršavanje. To znači kako ulazni kanal jednog okvira može ovisiti o izlazu drugog okvira. Takvo izvršavanje koda omogućava struktura ravnog slijeda (engl. Flat sequence) prikazana na slici 19.



Slika 19. Struktura ravnog slijeda

U prvom djelu blok dijagrama generiraju se prekidi (engl. Interrupt) u svrhu obavještavanja VI-a koji se nalazi na RT uređaju (engl. Host VI) kako bi se sinkronizirao s logičkim blokovima unutar FPGA. To je često rješenje sinkronizacije dva VI-a. Prekidi se mogu generirati u svrhu obavještavanja Host VI-a kao što je u ovom projektu za sinkronizaciju, za obavještavanje Host VI-a da se u programu desila neka greška ili obavijest za izvršeni zadatak. Sinkronizacija sprječava aplikaciju od pozivanja spremnika za izravan pristup memoriji (engl. Direct Memory Access First In First Out - DMA FIFO) prije nego što FPGA počne slati svoje podatke. Također sprječava FPGA od slanja svojih podataka prije nego što je aplikacija spremna primiti ih.

DMA sastoji se od dvije vrste FIFO međuspremnika (engl. Buffer-a), jedan koji se nalazi na RT uređaju i drugi koji se nalazi na FPGA. Nakon što se kreira funkcija DMA FIFO, prilikom programiranja koda na blok dijagramu mora se specificirati memorijski međuspremnik u koji se upisuju podaci i iz kojeg se čitaju isti. Budući da se DMA komunikacija temelji na FIFO-ovima, prijenos podataka se odvija jedan po jedan element. Prvi element u jednom međuspremniku je prvi element koji se prenosi u drugi međuspremnik. Na slici 20. prikazan je protok podataka iz DMA FIFO-a koji zapisuje u međuspremnik i drugi DMA FIFO koji čita iz drugog međuspremnika. [10]

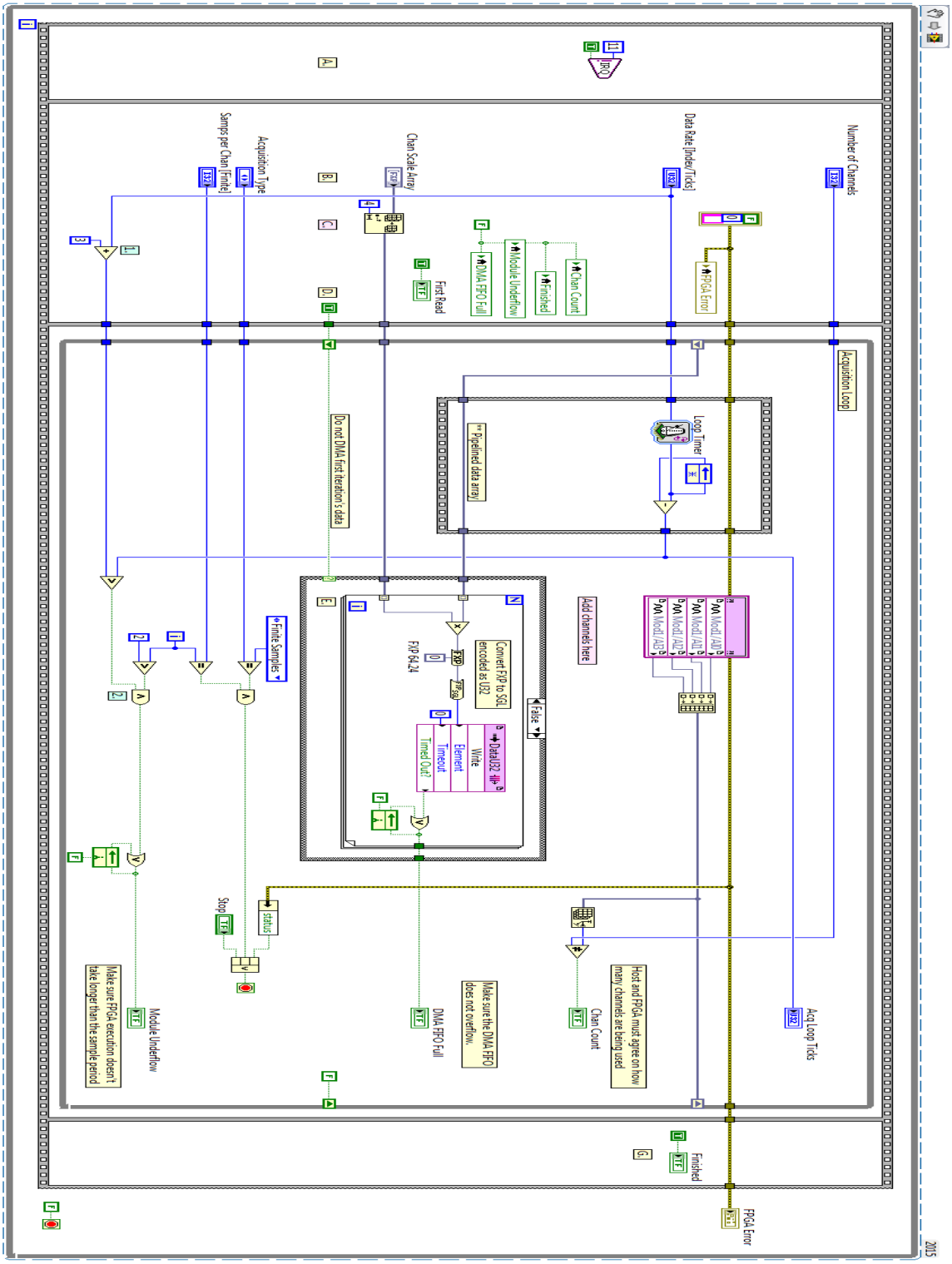


Slika 20. Protok podataka

U drugom dijelu koda kontrola koja se nalazi na korisničkom sučelju „Channel Scale Array“ je niz koji govori o tome koliko kanala sudjeluje u projektu. Ima fiksiranu vrijednost od 40 kanala koja se može mjenjati ovisno o tome s koliko korisnik kanala raspolaže. Budući da se u ovom projektu koristi 4 kanala na jednom modulu numerička konstanta spojena s tim nizom mora biti ista kao i broj spojenih ulaza.

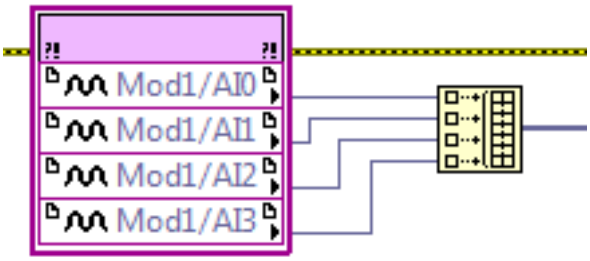
Ako aplikacija izgradi svoje podatke u LabVIEW valnom obliku, „First Read“ logička funkcija koja se nalazi na korisničkom sučelju ovog blok dijagrama u obliku lampice koja govori aplikaciji, a i korisniku da mora biti generirana nova vremenska oznaka.

U trećem dijelu koda izvršava se akvizicija. Svaka iteracija od „For“ petlje određuje podatke vrijede jednog uzorka (1 Sample of data) i pretvara ih u pojedinačne (engl. Single-Cycle Timed Loop - SGL) vrijednosti i stavlja ga u red za DMA FIFO. Na slici 21. nalazi se cjelokupni kod FPGA.vi-a.



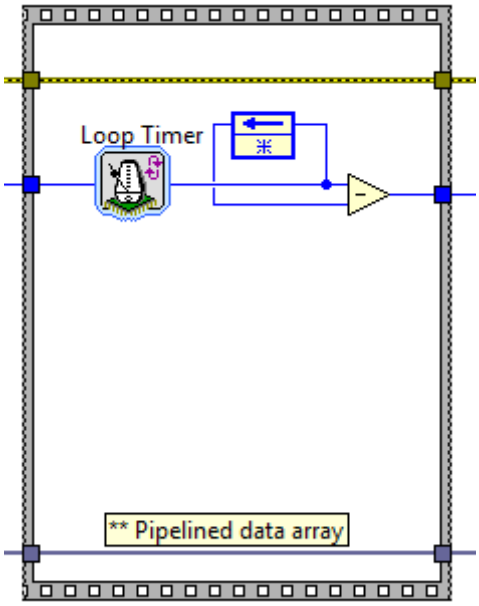
Slika 21. FPGA VI

Na mjestu ovog dijela koda počinje akvizicija s modula NI 9223, njegova 4 ulaza su prikazani na slici 22. Ulazi su spojeni u jedan niz pomoću funkcije za izgradnju niza nazvane „building array“.



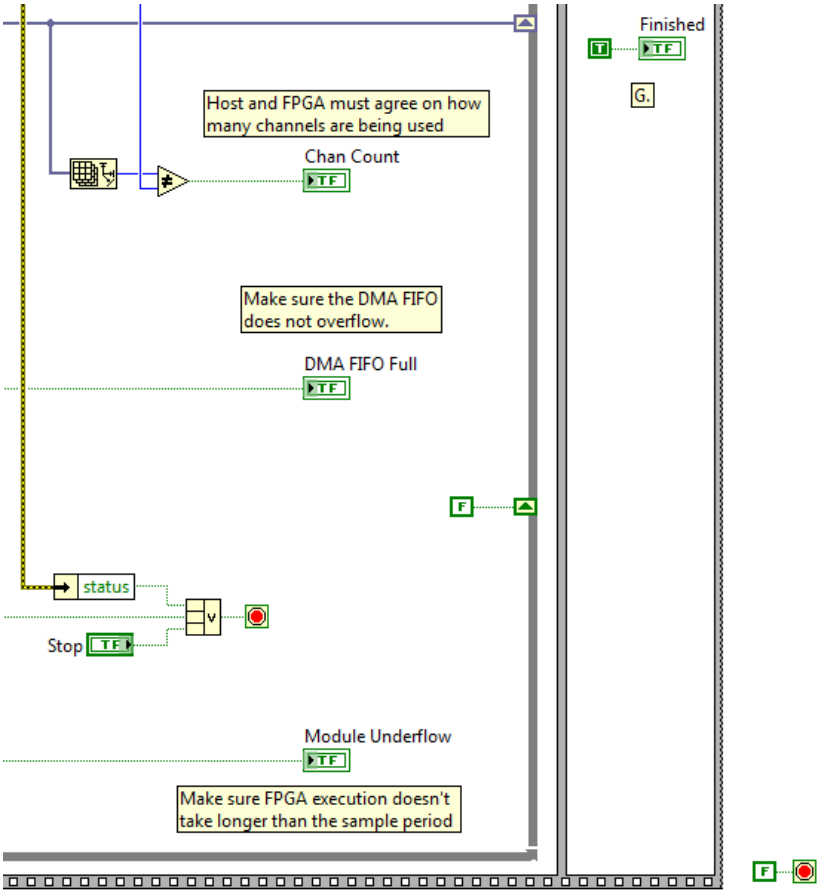
Slika 22. Prikupljanje jedne točke po kanalu

Uobičajeni način praćenja brzine petlje je koristiti funkciju brojača čija se vrijednost vraća kroz registar pomaka, a zatim funkciju oduzimanja da bi se razlikovale vrijednosti dvije petlje kako bi se dobilo ukupno vrijeme petlje. U ovom kodu korišten je drugačiji, kompaktniji način kod kojeg se koristi povratna veza kao što je prikazano na slici 23. Ova metoda je jednostavna i njome se izbjegava dodavanje vodova preko cijele petlje.



Slika 23. Postavljanje/mjerenje brzine petlje

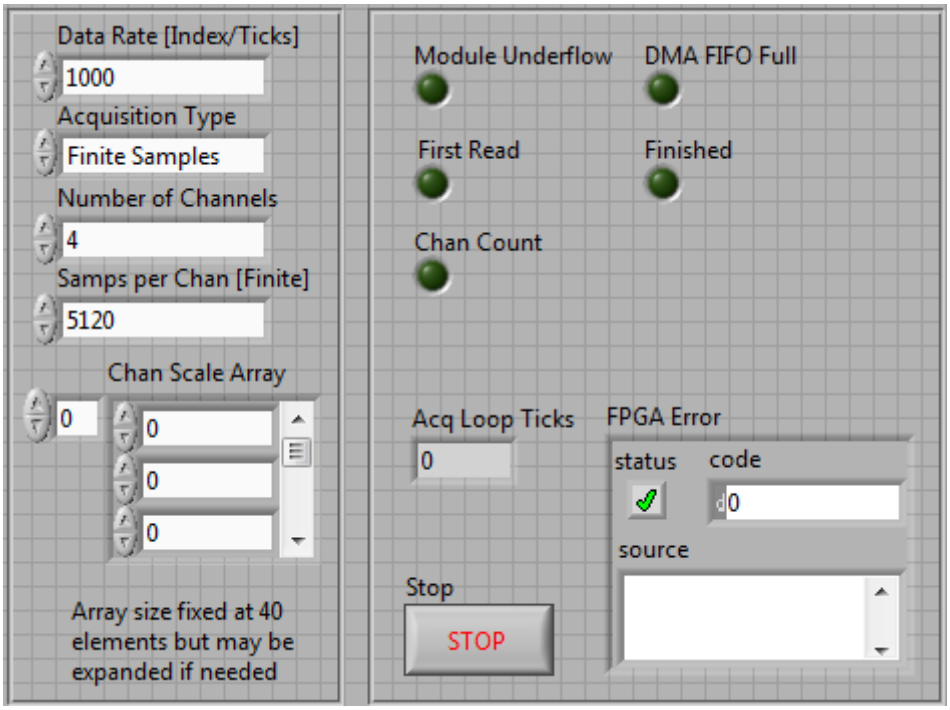
Završni dio FPGA VI-a prikazan na slici 24. radi provjere grešaka i čeka uvjete za zaustavljanje aplikacije. Indikator „Chan count“ provjerava dali je jednak broj kanala koji se koristi postavljen na FPGA VI-u i Host VI-u. „DMA FIFO Full“ prikazuje kada se DMA FIFO napuni podacima. „Module underflow“ indikator se uključuje ako za izvršavanje FPGA traje dulje od jednog perioda uzorka te ako je to slučaj izbacuje grešku i ne dozvoljava daljnju akviziciju.



Slika 24. Provjera grešaka i uvjeti za zaustavljanje aplikacije

Akvizicija će se zaustaviti ako je prikupljen određeni broj uzoraka u slučaju kada je postavka na korisničkom sučelju namještena na „finite aquisition“, ako se desi greška u akviziciji ili ako aplikacija prekine prikupljanje. „Finished“ govori aplikaciji da je petlja za prikupljanje podataka prestala. Kada prikupljanje prestane vanjska „while“ petlja neposredno priprema FPGA za novo prikupljanje.

Na slici 25. može se vidjeti kako je korisničko sučelje blok dijagrama FPGA VI-a podijeljeno na dva djela, kontrolni dio i pokazivački dio. Vrijednost brzine prikupljanja podataka (engl. „Data rate) mijenja se promjenom brzine uzimanja uzoraka na Host VI-u. Mjerna jedinica ovisi o tome kako je namješteno vremensko okidanje petlje. „Acquisition Type“ govori o vrsti akvizicije, a može biti konačna ili neprekidna. „Number of Channels“ govori koliko je kanala aktivno u projektu. „Samps per Chan“ je kontrola aktivna samo ako je vrsta akvizicije konačna i govori koliko treba uzoraka uzeti po kanalu. Pokazivači su objašnjeni u sklopu slike 25.



Slika 25. Korisničko sučelje FPGA VI-a

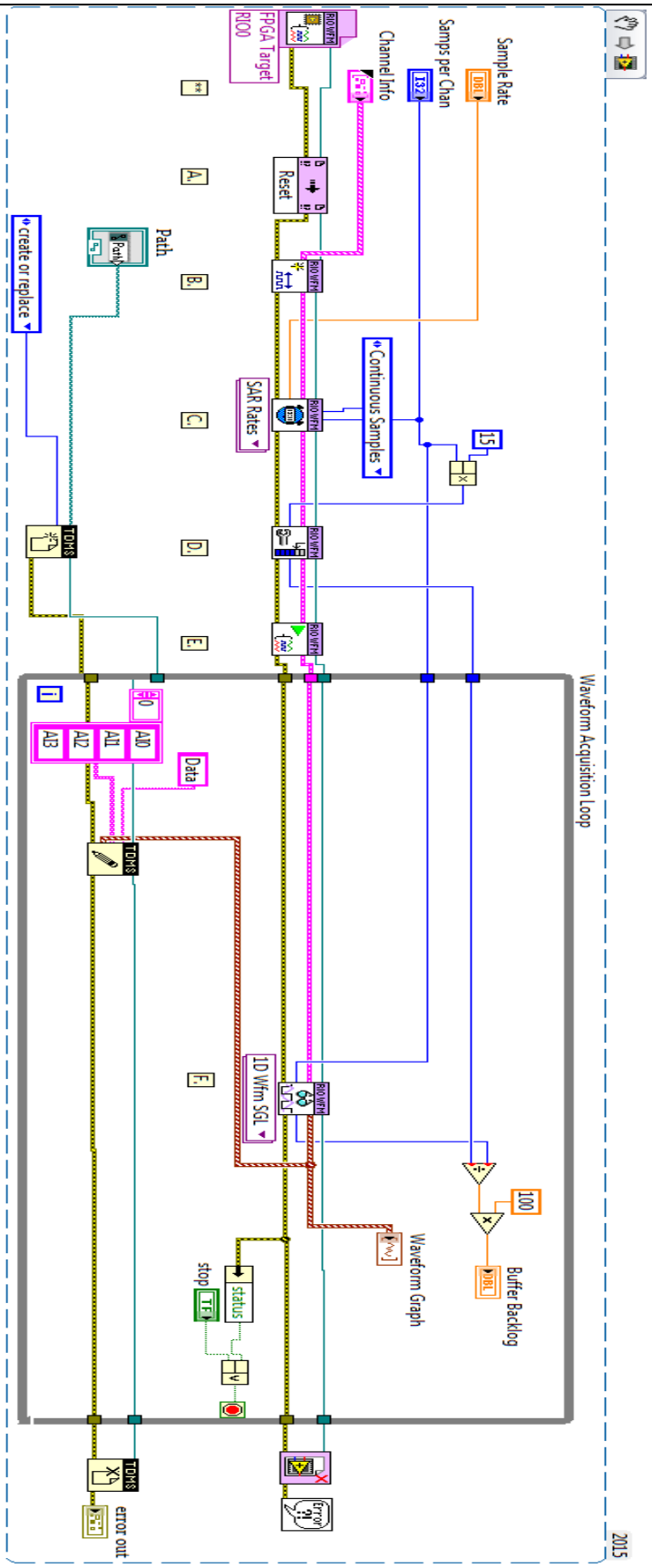
4.3. VI na „Real time“ uređaju

VI na „Real Time“ uređaju prikazan na slici 26. prikazuje kontinuiranu akviziciju. Takav način akvizicije govori CompactRIO kontroleru da prikuplja podatke sve dok se ne pojavi naredba za zaustavljanje aplikacije ili dok aplikacija ne naiđe na grešku. Memorijski međuspremnik je postavljen da bude višestruko veći od veličine za čitanje. U ovoj aplikaciji je postavljen da bude deset puta veći. Ako se memorijski međuspremnik napuni, posljedično dolazi do greške i FPGA se u tom trenutku zaustavlja.

Sve ikone na koje su strukturirane kao „RIO WFM“ su ustvari subVI-evi, tj. imaju svoj blok dijagram i korisničko sučelje koji se ne vide dok se program izvršava ali su jako bitan dio koda. U nastavku će biti detaljnije objašnjeni, a referencirat će se dodijeljenim slovima u prikazanom VI-u na slici 26.

„Invoke method“ VI služi za pozivanje FPGA Interface metode ili akcije sa VI-a na FPGA VI. Može se koristiti za sljedeće: Preuzimanje, prekidanje, resetiranje, i pokretanje FPGA VI-a, za čekanje i potvrđivanje prekida FPGA VI-a, za čitanje ili pisanje DMA FIFO. Metoda koja se koristi ovisi o hardveru i o FPGA VI-u. U ovom projektu je postavljen na „reset“ u svrhu resetiranja postavki hardvera na zadane vrijednosti definirane projektom, čisti FIFO-e, i reinicijalizira kontrole/indikatore.

Kontrolom „Sample Rate“ se na korisničkom sučelju određuje frekvencija uzimanja uzoraka dok kontrola „Channel Info“ prikazuje aktivne kanale.

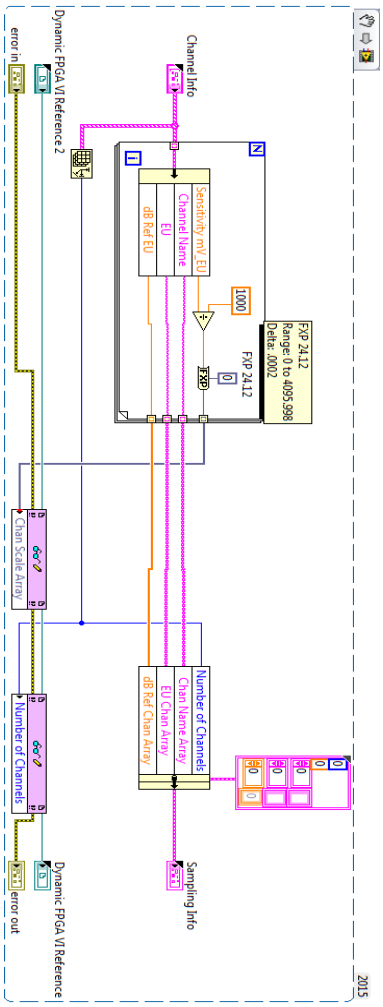


Slika 26. Real Time VI

4.3.1. Izrada kanala

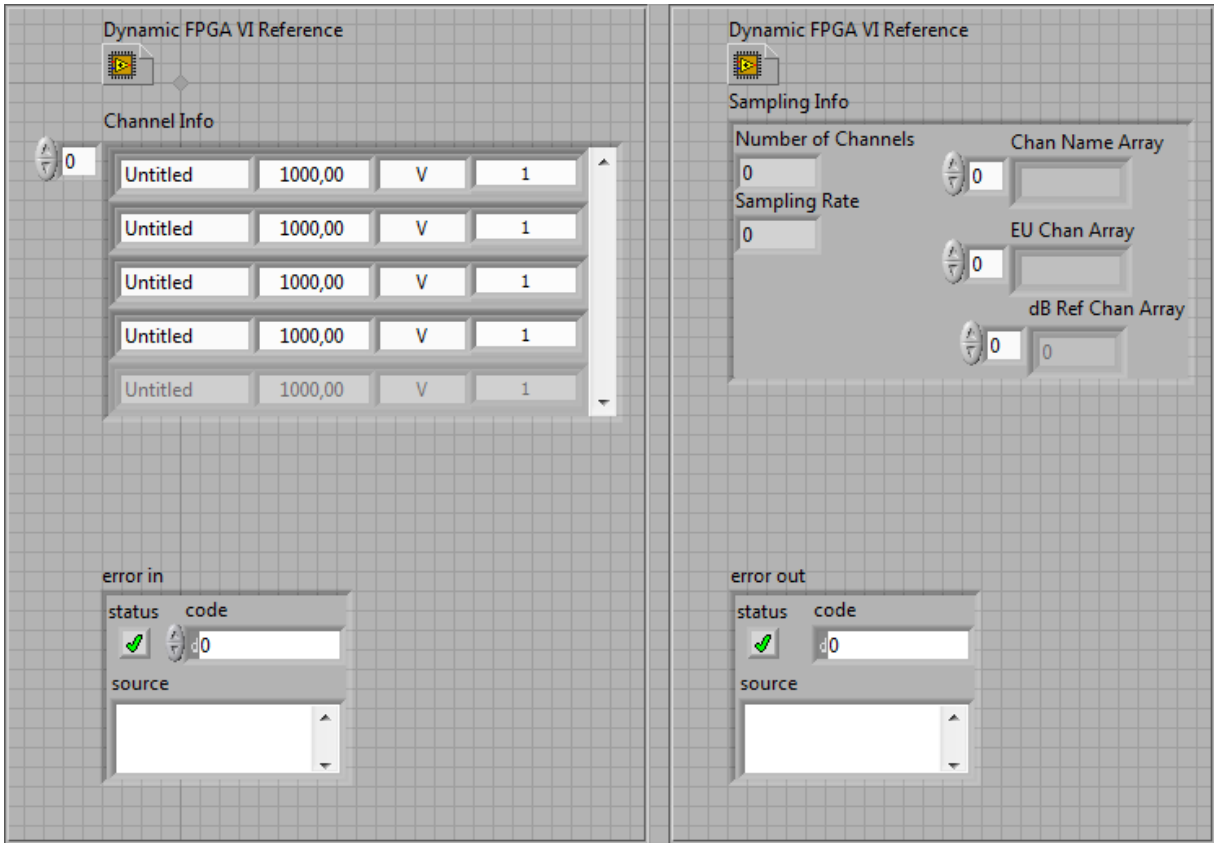
„Channel configuration“ SubVI pod slovom „B“ služi za prikupljanje i slanje broja kanala FPGA-u. Broj elemenata u „Channel Info“ mora biti isti kao broj kanala prikupljen od strane FPGA. Za isto to služi i u projektu.

VI-a na slici 27. prikazan je Blok dijagram „Channel Configuration“ sub VI-a. „Channel Info“ kontrola prikazuje aktivne kanale tako da se vrijednost dobivena s korisničkog sučelja ulazi u „For“ petlju i izvršava se onoliko puta koliko ima kanala. Svakom kanalu dodaje ime i skalu pomoću funkcije „Unbundle by name“. Zatim te vrijednosti prebacuje u drugu funkciju „Unbundle by name“ koja prikazuje izlazne vrijednosti koje se pojavljuju na korisničkim sučeljima svim subVI-ova.



Slika 27. Channel configuration blok dijagram

Na slici 28. prikazano je korisničko sučelje „Channel configuration“ blok dijagrama i također je podijeljen na dva djela. U prvom djelu se vide informacije u aktivnim kanalima dok se u drugom dijelu vide informacije o uzorkovanju.

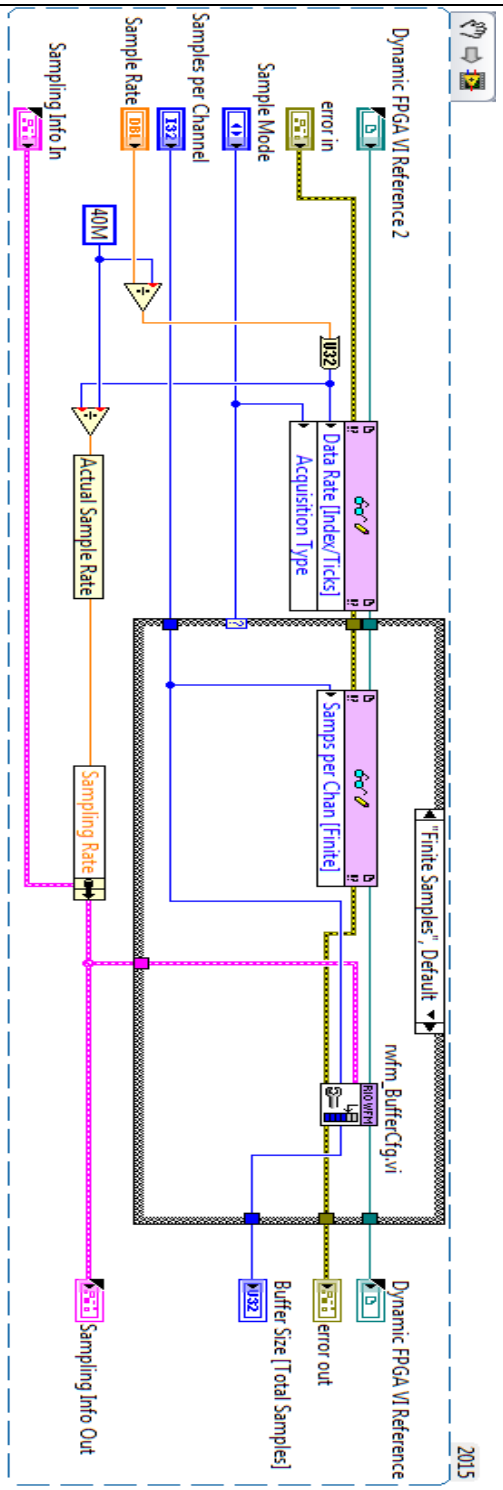


Slika 28. Channel configuration front panel

4.3.2. Podešavanje vremenske oznake

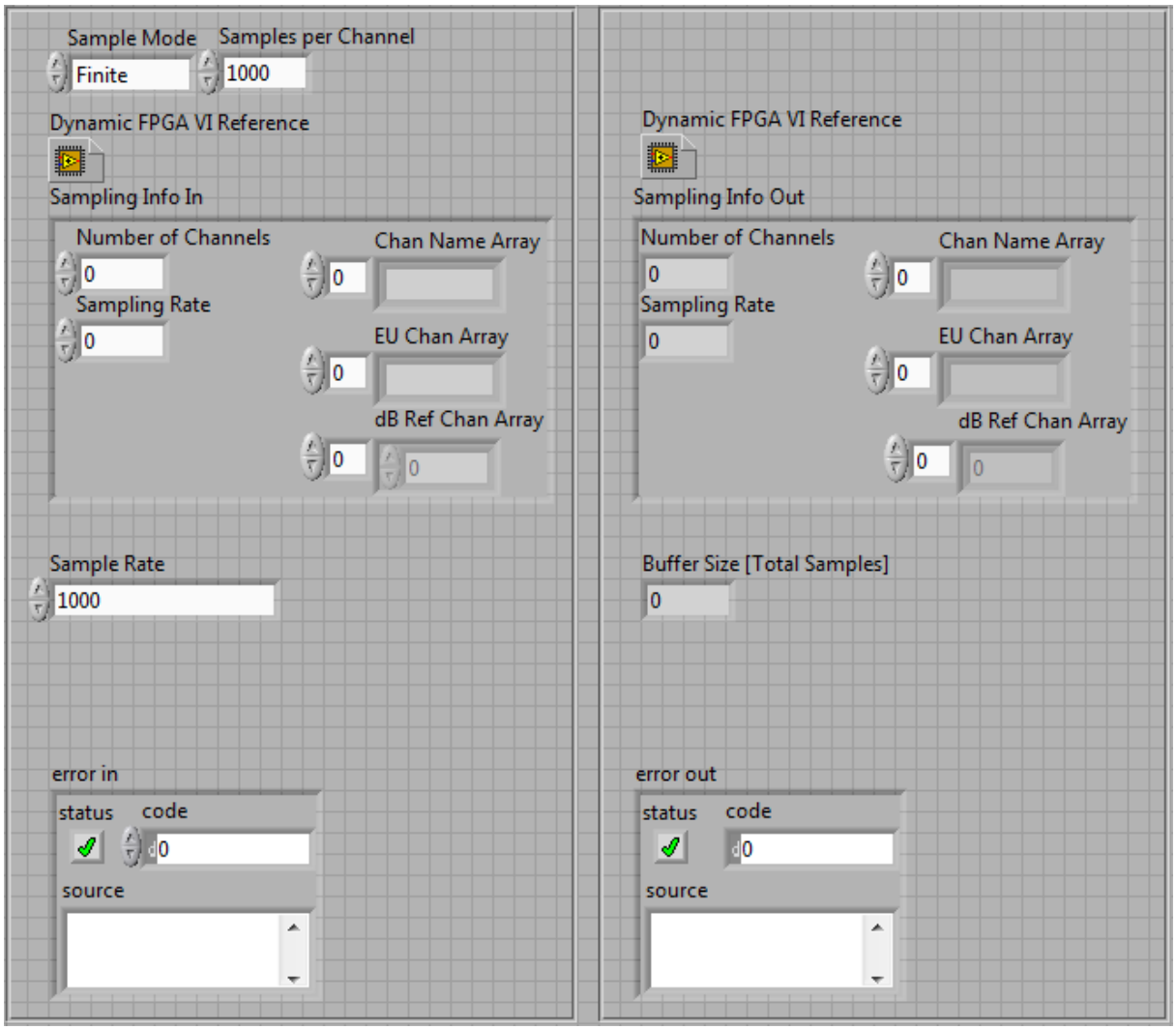
SubVI-em „Timing configuration“ pod slovom „C“ postavlja se metoda uzorkovanja (konačna ili neprekidna) i postavlja brzina uzorkovanja u FPGA.

U blok dijagramu prikazanom na slici 29. vrijednost odabrane metode akvizicije odlazi na tzv. Strukturu slučaja. Kod koji će se izvršavati unutar strukture slučaja ovisi o odabranoj metodi akvizicije. Također se numerička konstanta 40 milijuna dijeli s vrijednosti odabranom u kontroli „Sample Rate“ i s vrijednosti brzine prikupljanja podataka koja dolazi funkcijom „Read/Write“ sa FPGA VI-a. Nakon toga prikazuje stvarnu vrijednost frekvencije uzorkovanja.



Slika 29. Timing configuration blok dijagram

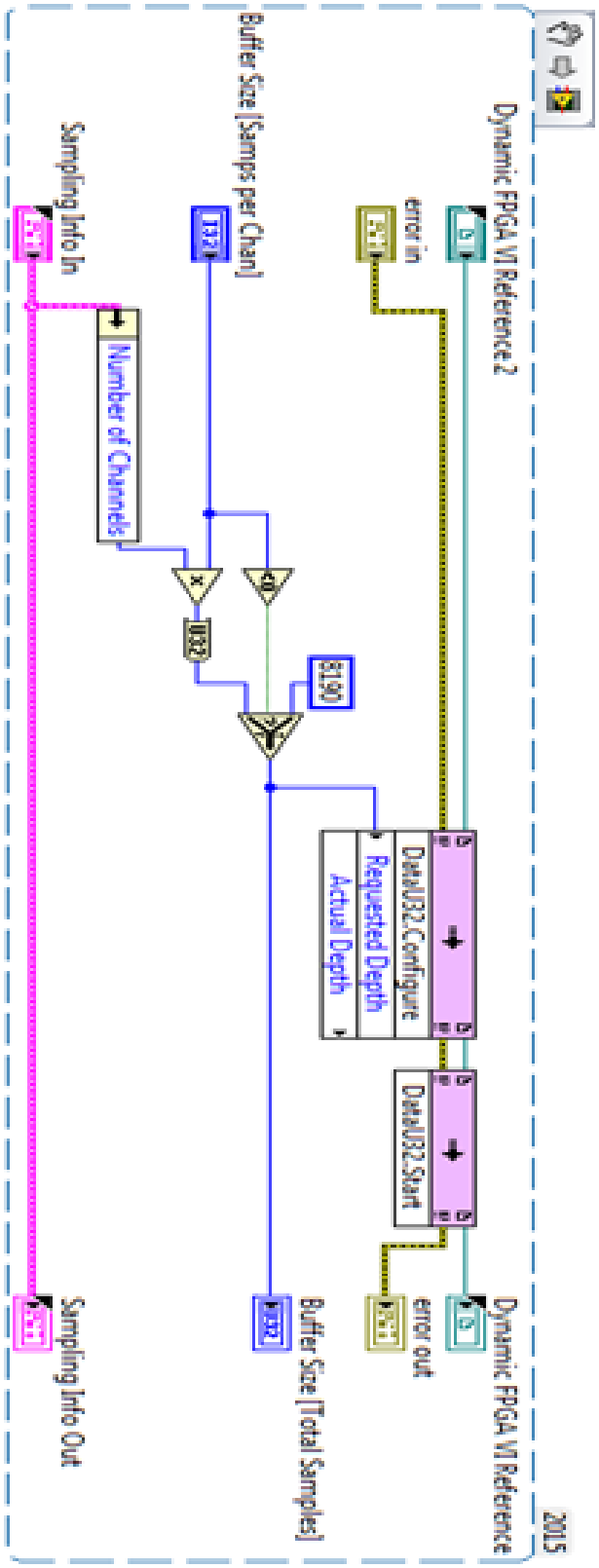
Na korisničkom sučelju prikazanom na slici 30. nalaze se ulazne i izlazne vrijednosti blok dijagrama.



Slika 30. Timing configuration Front panel

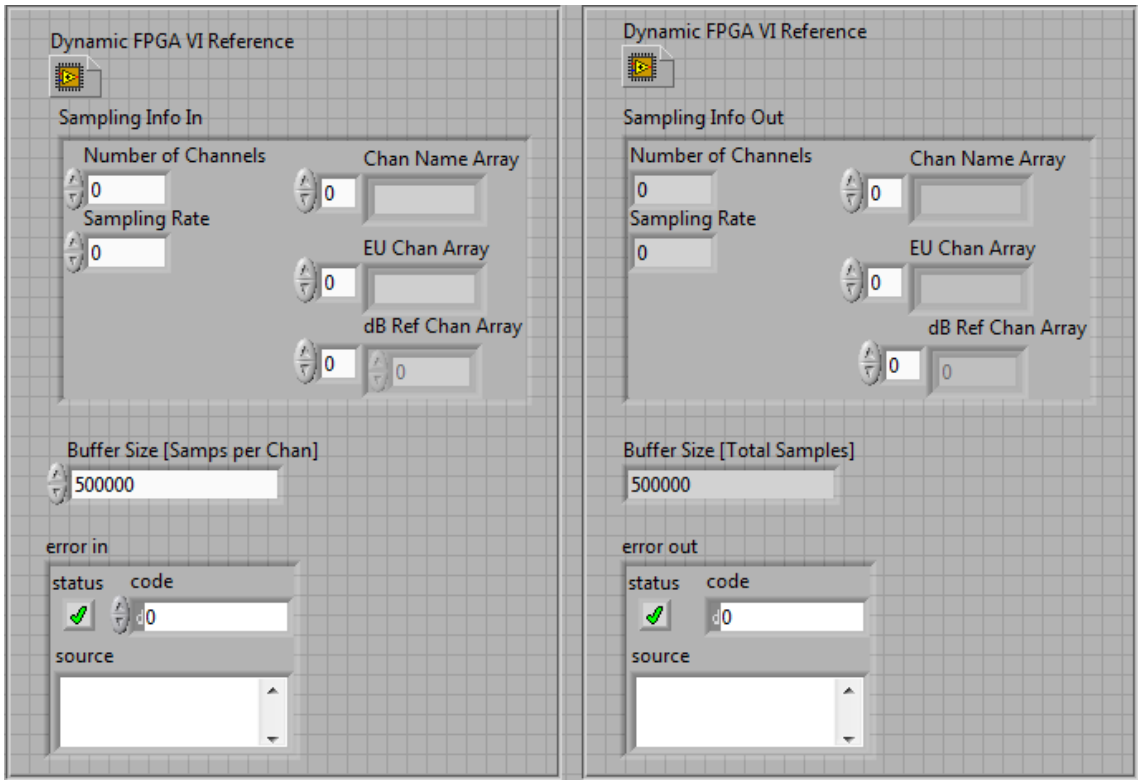
4.3.3. Memorijski međuspremnik

Na slici 31. prikazan je blok dijagram subVI-a „Buffer configuration“ prikazan slovom „D“ koji postavlja veličinu podatkovnog međuspremnika u aplikaciji, koja je važna za maksimalnu propusnost podataka.



Slika 31. Buffer configuration blok dijagram

Osim sadržaja koji se nalazi na korisničkom sučelju svih subVI-eva, na ovom se nalaze ulazna i izlazna vrijednost veličine memorijskog međuspremnika kao što je prikazano na slici 32.



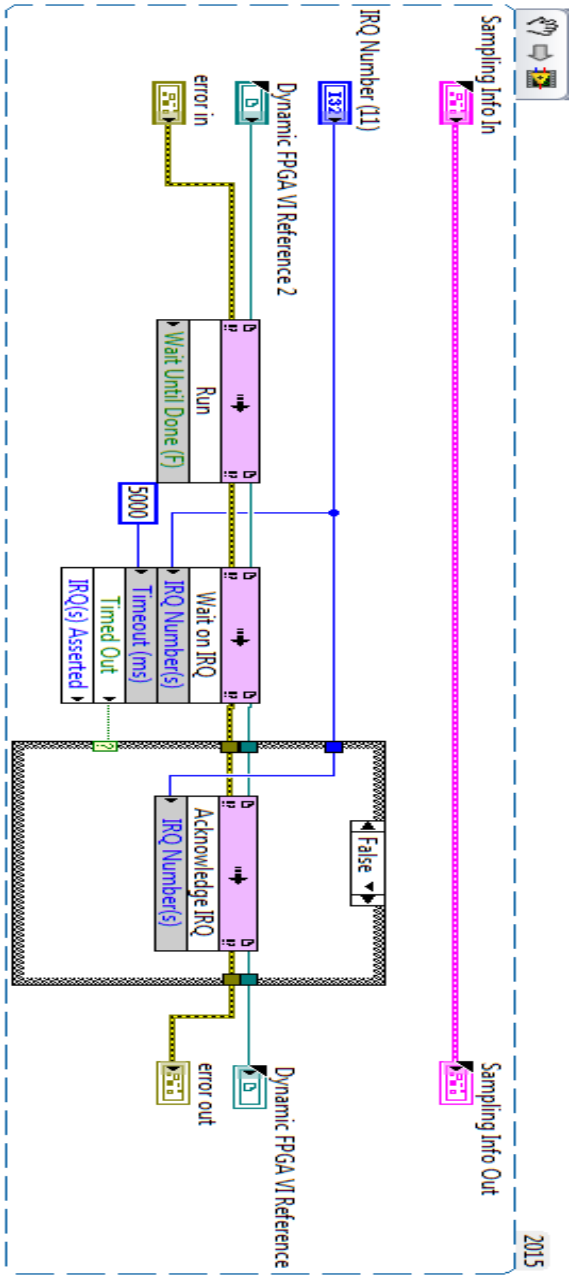
Slika 32. Buffer configuration front panel

4.3.4. Početak akvizicije

Slika 33. prikazuje blok dijagram subVI-a „Start acquisition“ pod slovom „E“ koji čeka prekid s FPGA.vi-a, time FPGA pokazuje da je spreman početi prikupljati podatke i pokreće se akvizicija. Ovaj VI može biti pozvan više puta da se resetira ili pauzira prikupljanje ili za okidanje konačnog prikupljanja. Ako je start pozvan više puta pojavit će se upozorenje da je FPGA već pokrenut.

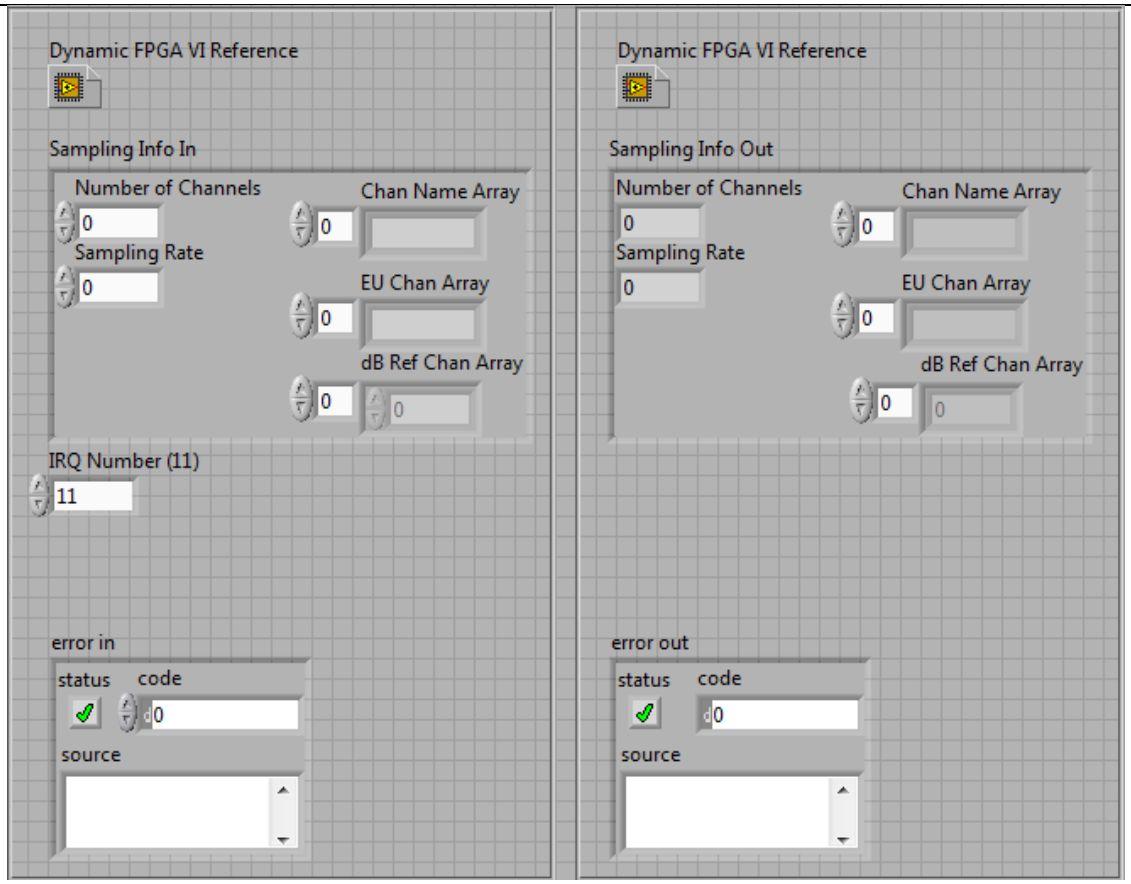
U ovom djelu koda se čeka na prekid koji generira prvi dio FPGA VI-a. „Invoke Methode“ funkcija je postavljena da čeka logički prekid. Nakon nje u strukturi slučaja još jedna „invoke method“ funkcija koja je postavljena da potvrđuje logički prekid. Struktura slučaja postavljena je na netočno sve dok se ne pojavi logički prekid.

Posljedično tome struktura slučaja (engl. Case structure) se prebacuje na točno stanje. U tom trenutku se Vi-evi sinkroniziraju i počinje akvizicija.



Slika 33. Start Acquisition blok dijagram

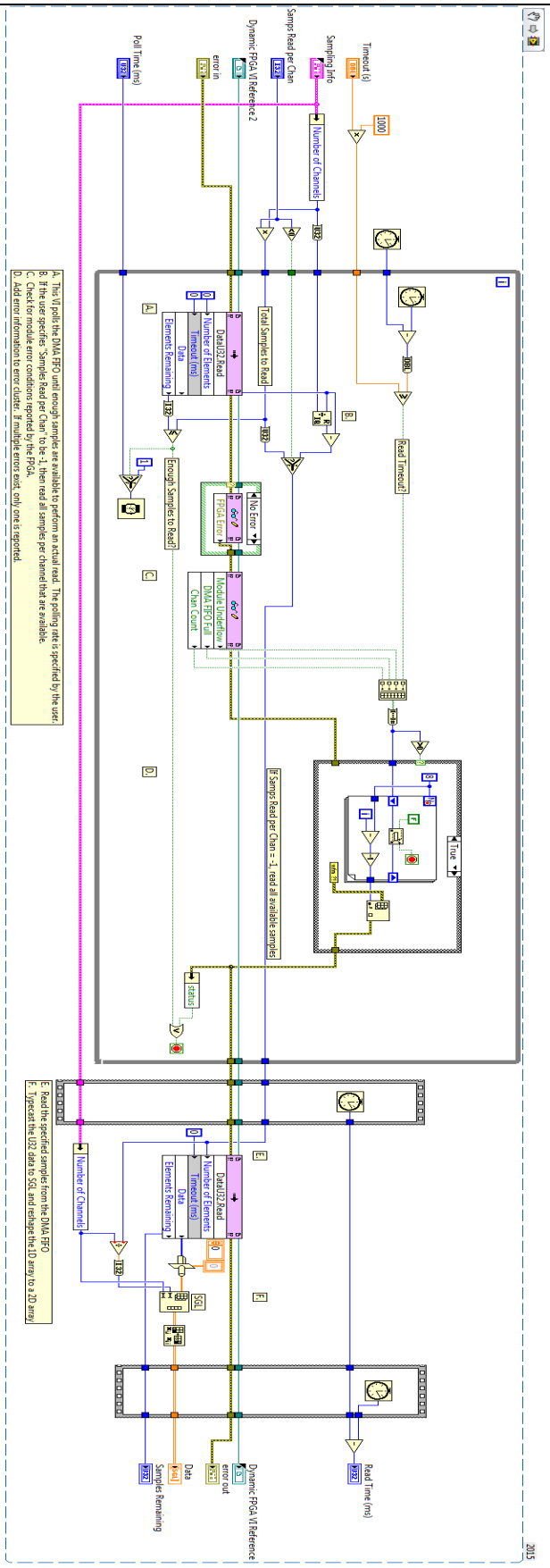
Na slici 34. prikazano je korisničko sučelje „Start aquisition“ subVI-a. Uz komponente koji su dio svih subVI-evi sadrži kontrolu koja govori koliki broj logičkih prekida treba dobiti da bi započela akvizicija.



Slika 34. Start Acquisition Front panel

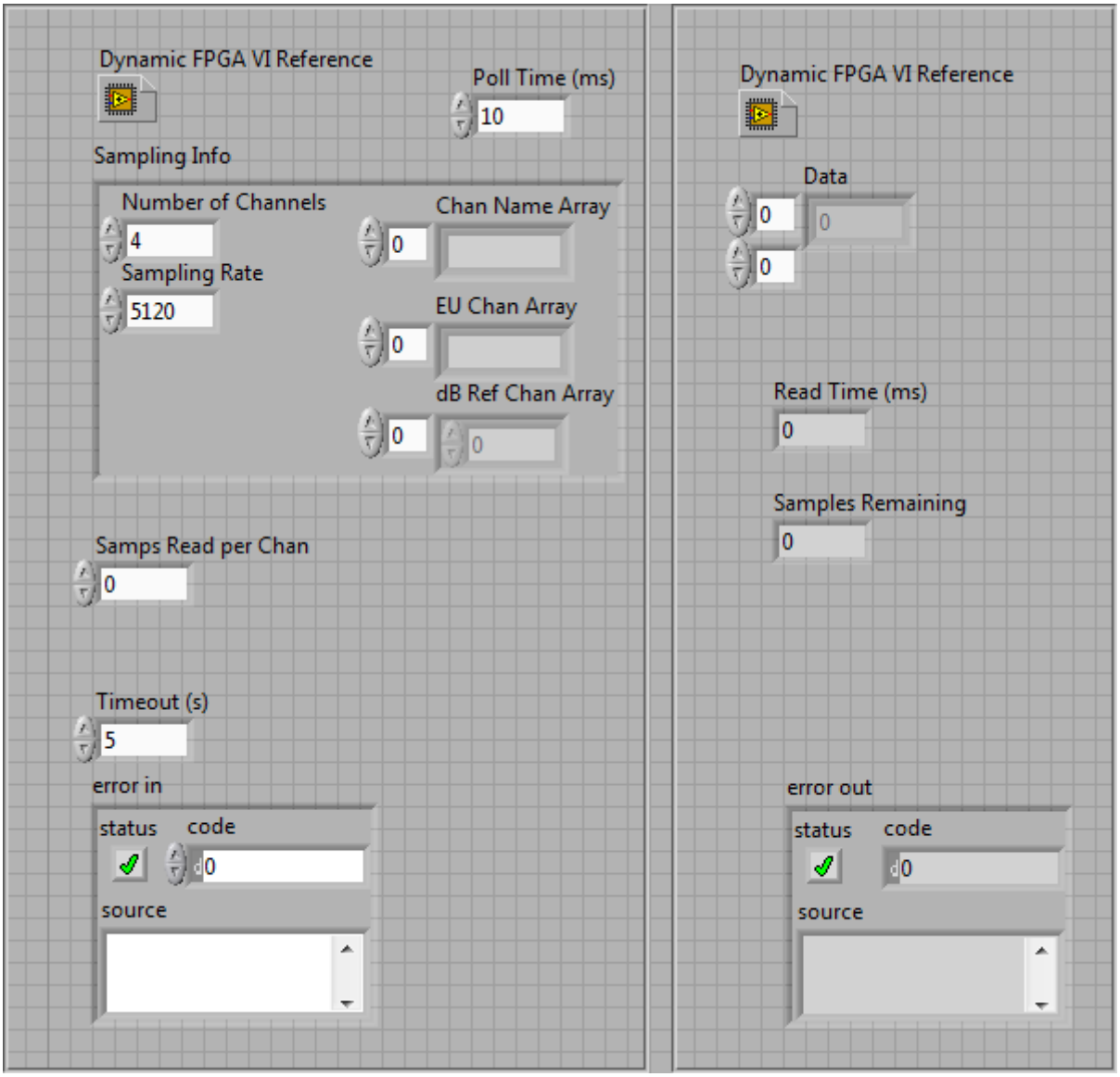
4.3.5. Čitanje podataka

Na slici 35. prikazan je blok dijagram subVI „Read data“ pod slovom „F“ koji stvara vremenske oznake za signal. Kako bi osigurao kontinuitet zadavanja vremenskih oznaka ovaj subVI kreira novu vremensku oznaku samo na početku nove akvizicije. Svaka dodijeljena vremenska oznaka proračunava se preko brzine uzimanja uzoraka i broja uzoraka koji je uzet.



Slika 35. Read Data blok dijagram

Na slici 36. može se vidjeti korisničko sučelje „Read data“ subVI-a.



Slika 36. Read Data front panel

4.3.6. TDMS

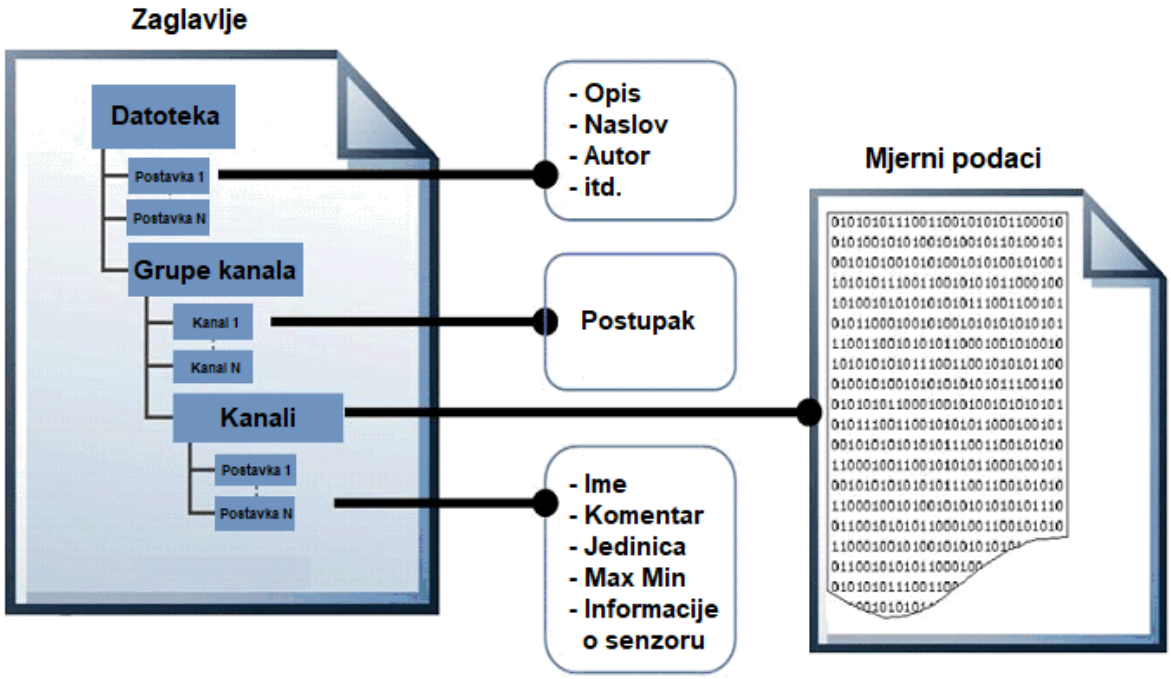
Pohrana podataka vrši se pomoću funkcije koju je tvrtka National Instruments razvila baš za ovakve potrebe kao što je pohrana toka podataka, a naziva se TDMS. Format datoteke TDMS razvijen je u nastojanju da se otklone nedostaci postojećih formata pohranu koje su se najčešće koristile u aplikacijama za mjerenje i testiranje. Također je razvijen kako bi smanjio potrebu za osmišljavanjem i održavanjem oblika korisnikove datoteke. Budući da se potrebe i aplikacije tijekom vremena mijenjanju, čak i najpopularniji formati za pohranu ubrzo postaju nezadovoljavajući za zahtjeve inženjera i znanstvenika koji pohranjuju mjerne podatke temeljene na vremenu. U tablici 3. prikazane su prednosti i nedostaci nekih opcija za pohranu podataka. [8]

Tablica 3. Prikaz nekih opcija za pohranu

	ASCII	Binarno	XML	Baza podataka	TDMS
Zamjenjivi	●		●		●
Otisak na malom disku		●			●
Pretraživački				●	●
Nerazdvojiva svojstva			●		●
Visokobrzinski protok		●			●
Podržana NI platforma	●	●	●	●*	●

TDMS model pohrane podataka pruža tri razine hijerarhije – Datoteka, grupa, kanal. Datoteka može sadržavati neograničeni broj grupa, a svaka grupa može sadržavati neograničeni broj kanala. Zbog grupiranja kanala korisnik može odabrati kako organizirati podatke u svrhu boljeg razumijevanja podataka. Na svakoj razini moguće je spremanje prilagođenih skalarnih svojstava. Svaka razina odobrava neograničen broj prilagođenih svojstava kako bi se postigle dobro dokumentirane i podatkovne datoteke spremne za pretraživanje. Opisne informacije koje se nalaze u datoteci,

ključna su prednost ovog modela, jer pružaju jednostavan način dokumentiranja podataka bez dizajniranja vlastite strukture zaglavlja. Ako se zahtjevi za dokumentacijom povećavaju, korisnik ne mora redizajnirati svoju aplikaciju nego jednostavno proširi model kako bi se zadovoljile te specifične potrebe. Što je više prilagođenih svojstava korišteno za pohranu mjernih podataka, to se lakše može pronaći kasnije pomoću NI DataFinder klijenta koji sažima složenu komunikaciju baze podataka od korisnika. Na slici 37. je prikazana struktura TDMS datoteke zapisane u software-u LabVIEW.[9]

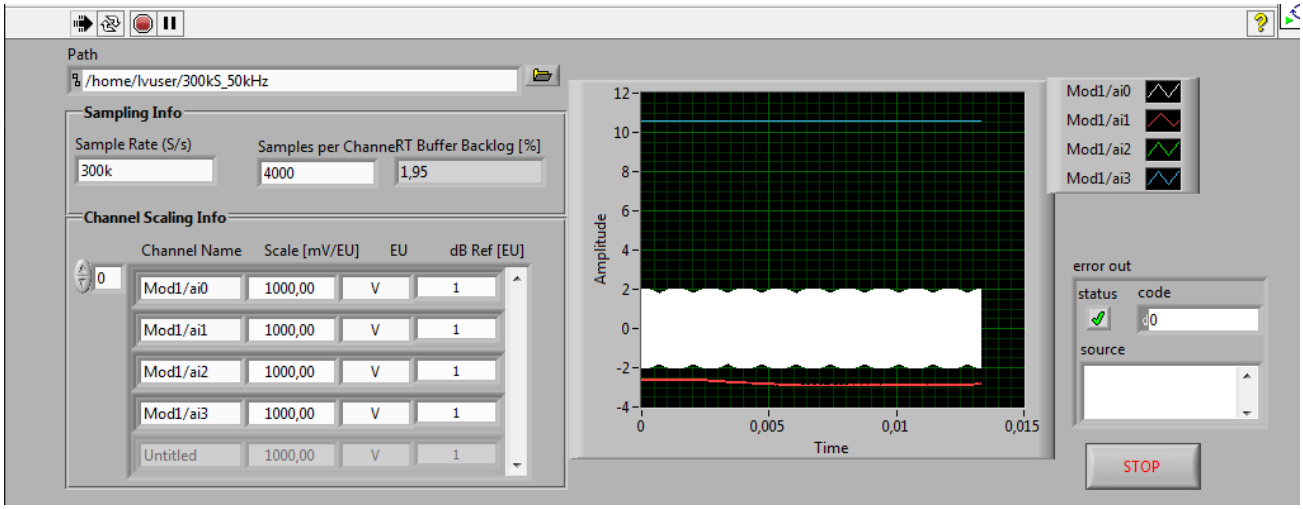


Slika 37. Svaka TDMS datoteka zapisana u LabVIEW sadrži opisne informacije o razinama [9]

Uz TDMS datoteku pojavljuje se i binarna indeksna datoteka sa ekstenzijom .TDMS_index. Datoteka TDMS_Index pruža konsolidirane informacije o svim postavkama i pokazivačima u datoteci za skupne podatke i ubrzava pristup podacima tijekom čitanja. Datoteka TDMS_Index automatski se regenerira ako se izgubi. [9]

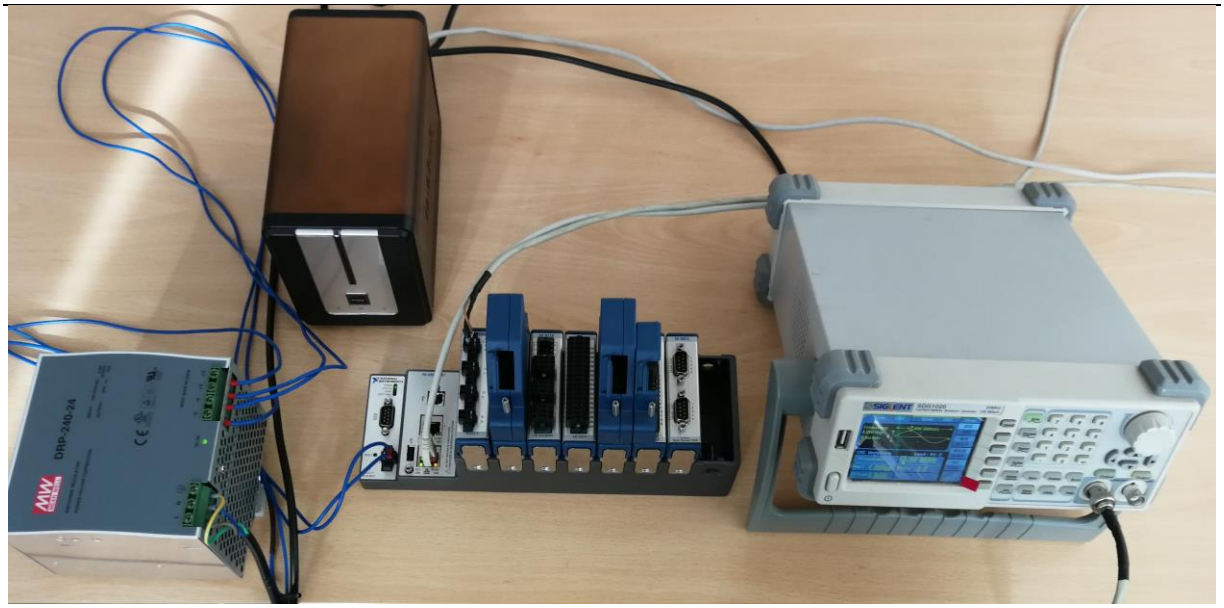
5. REZULTATI

Kao što se može vidjeti na slici 38. na korisničkom sučelju VI-a na RT uređaju se prikazuje signal čiji se uzorci pohranjuju. U gornjem lijevom kutu je prostor u kojem se zapisuje putanja gdje će se datoteka .tdms kreirati jednom kad počne akvizicija. Također se nalaze kontrole za brzinu uzimanja uzoraka. „Buffer backlog“ prikazuje preostalu rezervu u memorijskom međuspremniku.



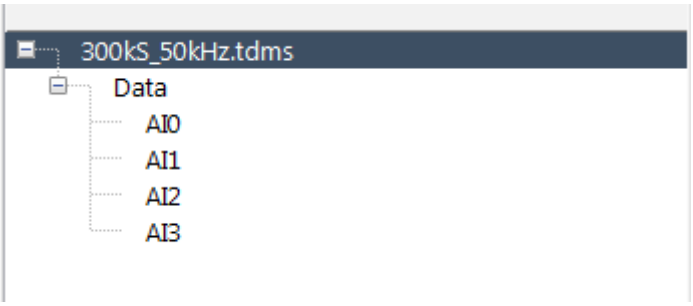
Slika 38. Korisničko sučelje VI-a na RT uređaju

Sustav je testiran spajanjem generatora funkcija koji simulira senzor na jedan kanal modula NI 9223 i puštanjem sinusnog signala frekvencije 50kHz. Nakon što je akvizicija završena i datoteka je pohranjena na zadanom mjestu u memoriji pomoću softvera TDMS viewer provjereno da li je signal poslan na mjernu stanicu i pohranjen i je li frekvencija uzorkovanja bila ona koja je postavljena na korisničkom sučelju VI-a na RT uređaju. U ovom slučaju to je bila frekvencija uzorkovanja od 300kS/s. Na slici 39. mogu se vidjeti svi dijelovi sustava, CompactRIO 9067 sa spojenim odgovarajućim modulima, napajanje, uređaj za pohranu podataka i generator funkcija spojen u svrhu testiranja sustava.



Slika 39. Dijelovi sustava

Kao što je u radu rečeno podaci se spremaju u .tdms formatu. Na slici 40. može se vidjeti njegova struktura. Datoteka se naziva *300kS_50kHz.tdms* što označava brzinu uzimanja uzoraka i frekvenciju referentnog signala. Grupa je nazvana *Data,* a kanali su nazvani onako kako je zapisano na modulu na kojem se nalaze.



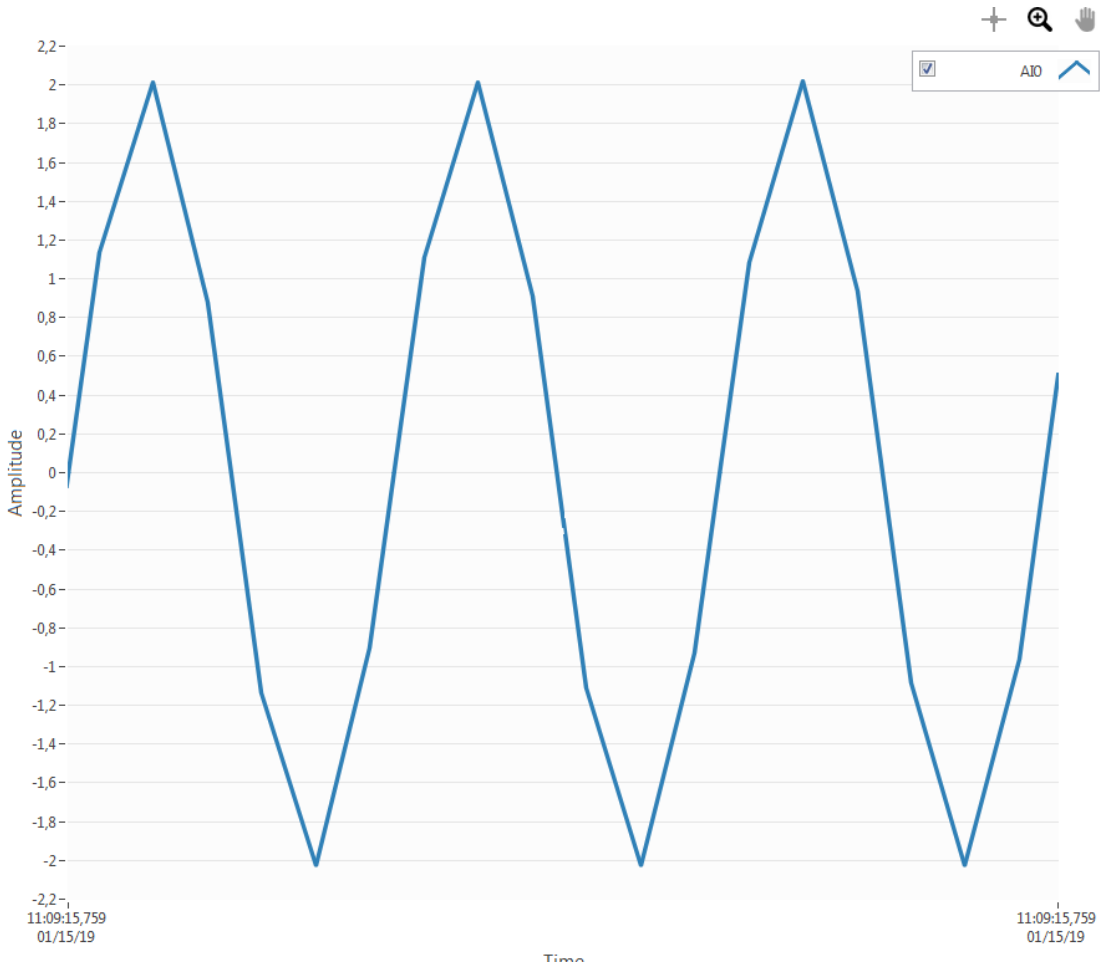
Slika 40. Struktura TDMS datoteke

Na slici 41. mogu se vidjeti vrijednosti uzoraka. Ako se malo bolje pogledaju rezultati primijetit će se da je samo na kanalu AI0 spojen generator funkcija, a na ostalima je šum. U budućoj upotrebi sva 4 kanala će biti u upotrebi ali za prikaz rezultata nije bilo potrebe za time.

Data AI0	Data AI1	Data AI2	Data AI3
1,757845	-2,115696	-2,793417	10,612408
0,349926	-2,13541	-2,789211	10,612408
-1,549276	-2,16288	-2,797623	10,612408
-1,90966	-2,161264	-2,79277	10,612408
-0,375364	-2,145105	-2,790831	10,612408
1,530626	-2,1241	-2,795683	10,612408
1,907816	-2,119251	-2,789536	10,612408
0,380955	-2,133471	-2,795359	10,612408
-1,52956	-2,161911	-2,7973	10,612408
-1,920326	-2,168697	-2,793095	10,612408
-0,406071	-2,138641	-2,796005	10,612408
1,50897	-2,127008	-2,792124	10,612408
1,917835	-2,121838	-2,793741	10,612408
0,411659	-2,127331	-2,795359	10,612408
-1,508228	-2,165789	-2,793417	10,612408
-1,930346	-2,166759	-2,798271	10,612408
-0,438068	-2,138966	-2,790831	10,612408
1,487961	-2,126038	-2,789536	10,612408
1,926886	-2,117958	-2,794065	10,612408
0,442041	-2,135733	-2,789211	10,612408
-1,487543	-2,160618	-2,795683	10,612408
-1,940042	-2,167082	-2,796329	10,612408
-0,468451	-2,149954	-2,790182	10,612408
1,466953	-2,115696	-2,792448	10,612408

Slika 41. Tablica uzoraka

Kada je sinusni signal poslan frekvencijom 50 kHz i uzorkovan frekvencijom 300 kS/s svaka sinusoida trebala bi biti opisana sa šest točaka. Na slici 42. prikazan je uzorkovani signal u kojem se vidi da je svaka sinusoida opisana sa šest točaka.



Slika 42. Graf uzoraka

6. ZAKLJUČAK

U razvoju suvremenih alatnih strojeva sustavi za nadzor obradnih procesa nezaobilazan su čimbenik. Razvojem takvih sustava moguće je podići razinu autonomnosti alatnih strojeva. U tu svrhu nastala je ideja o razvoju sustava za akviziciju signala visoke frekvencije kojim bi se omogućila integracija različitih senzora u obradne sustave. Neki od takvih senzora su senzor vibracija i akustične emisije čiji su parametri vrlo dobar pokazatelj trošenja alata u obradnim procesima. Njihov frekvencijski spektar može se kretati do 1 MHz što čini akviziciju standardnim akvizicijskim modulima gotovo nemoguću. Stoga je cilj rada bio je realizirati sustav koji ima mogućnost akvizicije signala na frekvencijama do 1MS/s.

U radu je realiziran sustav primjenom LabVIEW okruženja koji se sastoji od CompactRIO mjerne stanice spojene direktno na PC i modula NI 9223 C serije. Kod se sastoji iz dva dijela, odnosno dio koji se izvršava na uređaju CompactRIO i dio koji se svojim većim djelom izvršava na FPGA dok je pohrana podataka riješena mrežnim sustavom za pohranu datoteka. Programska podrška uspješno je realizirana i testirana uporabom generatora funkcija, a rezultatima je pokazano da je takva akvizicija moguća.

7. LITERATURA

[1] Travis, J., Kring, J.: LabVIEW for Everyone: Graphical Programming Made Easy and Fun, Prentice Hall, 2006.

[2] Halvorsen, H. P.: Introduction to LabVIEW, University College of Southeast Norway, 2016.

[3] National Instruments: „Understanding NI CompactRIO Scan Mode“, dostupno na: http://www.ni.com/white-paper/7338/en/#toc6

[4] NI LabVIEW for CompactRIO Developer's Guide: Recommended LabVIEW Architectures and Development Practices for Control and Monitoring Applications

[5] National Instruments: „What is a Real-Time Operating System (RTOS)?“, dostupno na: http://www.ni.com/white-paper/3938/en/

[6] National Instruments: „FPGA Fundamentals“, dostupno na: http://www.ni.com/white-paper/6983/en/

[7] National Instruments: „NI 9223“, dostupno na: http://www.ni.com/pdf/manuals/374223d.pdf

[8] National Instruments: „The NI TDMS File Format“, dostupno na: http://www.ni.com/white-paper/3727/en/

[9] National Insruments: „Introduction to LabVIEW TDM Streaming VI-s“, dostupno na: http://www.ni.com/white-paper/3539/en/

[10] National Instuments: „How DMA Transfers Work (FPGA Module)“, dostupno na: http://zone.ni.com/reference/en-XX/help/371599M01/lvfpgaconcepts/fpga_dma_how_it_works/

[11] Popović, A., Marković, M., Panić, B., Nikolić, M.: DATA ACQUISITION AND PROCESSING, Institut za fiziku Beograd, 2006.

PRILOZI

1. CD-R Disc