Dizajn sučelja za akviziciju podataka mjerenja u nastavnom laboratoriju

Kristić, Vedran

Undergraduate thesis / Završni rad

2018

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:200:561011

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2024-05-03**

Repository / Repozitorij:

Faculty of Electrical Engineering, Computer Science and Information Technology Osijek

SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I INFORMACIJSKIH TEHNOLOGIJA U OSIJEKU

Sveučilišni studij

DIZAJN SUČELJA ZA AKVIZICIJU PODATAKA MJERENJA U NASTAVNOM LABORATORIJU

Završni rad

Vedran Kristić

Osijek, 2018.

SADRŽAJ

1. UVOD....1

1.1. Zadatak završnog rada....1

2. MJERNA OPREMA....2

2.1. Multimetri i osciloskopi....2

2.2. Strujni i naponski mjerni članovi....6

2.3. Otpornici....12

2.4. Akvizicija podataka DAQ karticama....13

2.4.1. Automatizirano mjerenje i kontrola....17

2.4.2. DAQ kartica NI USB-6218....19

2.4.3. Računalo s LabVIEW 2016 aplikacijskim softverom....22

3. IZRADA SUČELJA....24

3.1. Funkcioniranje sučelja za akviziciju podataka....60

3.2. Rukovanje sučeljem za akviziciju podataka....61

4. MJERENJA....72

4.1 Dobivanje rezultata u LabVIEW-u....72

4.2. Umjeravanje mjernih članova....77

5. ZAKLJUČAK....80

LITERATURA....81

SAŽETAK....84

ABSTRACT....85

ŽIVOTOPIS....86

1. UVOD

Nastavni laboratoriji elektrotehničkih učilišta u osnovi su poznati kao mjesta na kojima studenti pomoću velikog broja instrumenata i ostale opreme vrše mjerenja različitih električnih veličina te njihove rezultate zapisuju i dobivene podatke 'ručno' obrađuju. Ovakav način rada u laboratoriju za sobom povlači probleme vezane za fizička ograničenja na ispitnim stolovima, ali i one koji se tiču održavanja brojne mjerne opreme.

U cilju poboljšanja mjernih procedura i izbjegavanja prethodno navedenih problema, u posljednje vrijeme postoji trend korištenja suvremenih akvizicijskih (DAQ) kartica čijim se korištenjem u realnom vremenu osigurava kvaliteta mjerenja i prikaz mjerenih veličina na računalu, te njihova pohrana kako bi se podaci po potrebi naknadno mogli digitalno obrađivati.

U drugom poglavlju će se opisati pojedini elementi mjerne opreme koja se koristi u nastavnom laboratoriju elektrotehničkog učilišta (princip rada, svrha, korištenje, načini uporabe, tehničke specifikacije).

Zatim će se opisati DAQ kartica i njeno korištenje u svrhu podizanja kvalitete i prikaza mjerenja. Za komunikaciju mjernog postava s računalom koristit će se programski paket LabVIEW proizvođača DAQ kartica s već ugrađenim upravljačkim programima i podrškom korisnicima.

U trećem poglavlju će se detaljno opisati procedura izrade sučelja u programskom paketu LabVIEW čime se osigurava mogućnost preinake u skladu s korisničkim zahtjevima. Prikupljeni podaci na računalu moraju se moći prebaciti u mrežu, odnosno za njihovo korištenje u drugim aplikacijama za analizu i prezentaciju mjerenja.

Verifikacija rada sučelja će se testirati u četvrtom poglavlju kroz prikaze praktičnog umjeravanja mjernih članova, te mjerenja u realnom okruženju korištenjem prethodno opisane mjerne opreme.

1.1.Zadatak završnog rada

Zadatak završnog rada je izrada programskog sučelja koje će u prvom redu osigurati akviziciju podataka mjerenja u računalo, zatim prikaz trenutnih i TRMS vrijednosti mjerenih veličina i njihovo pohranjivanje. Programsko sučelje treba moći komunicirati s DAQ karticom s više A/D ulaza koje će s naponskih i strujnih mjernih članova prikupljati podatke mjerenja električnih veličina. U tom smislu u radu se treba opisati procedura umjeravanja postojećih mjernih članova da bi prikupljeni podaci imali pravovaljane vrijednosti.

2. MJERNA OPREMA

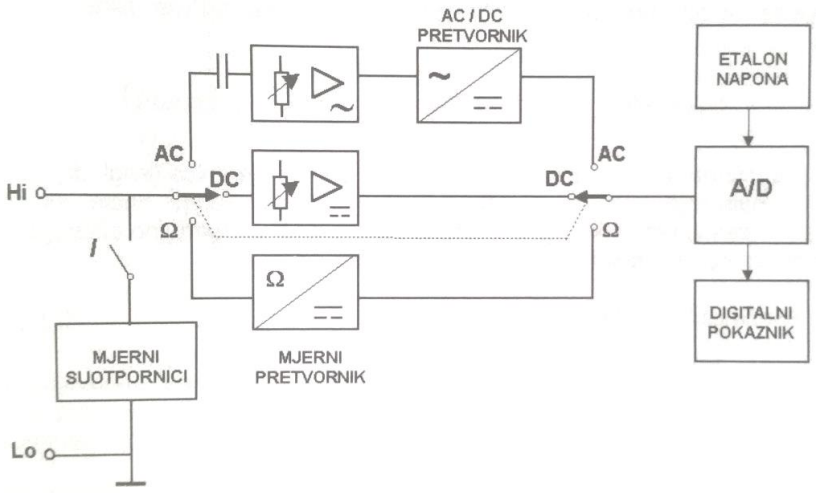
U svrhu praktične nastave studentima se kroz laboratorijske vježbe daje uvid u procedure mjerenja, upoznaje s mjernom opremom, te se povezuje teorijsko znanje s praktičnom primjenom.

Metodika nastave većine kolegija koncipirana je na način da se na predavanjima daje teorijska podloga, koja se zatim utvrđuje rješavanjem primjera iz prakse na laboratorijskim vježbama. Npr., na predavanjima se studentima prezentira Kirchhoffov zakon za napone i Kirchhoffov zakon za struje, te njihova pravila, na auditornim vježbama se kroz rješavanje nepoznatih napona i struja određuju očekivane vrijednosti, koje se konačno potvrđuju kroz mjerenje zadanih veličina određene mreže. U konačnici se očekuje da se studenti samostalno usvojenim znanjem s predavanja i auditornih vježbi pripreme za laboratorijske vježbe.

2.1. Multimetri i osciloskopi

Multimetar je elektronički mjerni instrument koji objedinjuje funkcije voltmetra, ampermetra i omometra. Dijele se na analogne i digitalne; analogni koriste kazaljku za prikaz vrijednosti na mjernoj skali, dok digitalni prikazuju numeričku vrijednost na digitalnom pokazniku. Analogni multimetri obično služe za mjerenje istosmjernih i izmjeničnih napona i struja, te otpora. S druge strane, većina današnjih digitalnih multimetara osim ovih ima ugrađene funkcije za mjerenje kapaciteta, frekvencije, ispravnosti i pojačanja tranzistora, dok napredniji mogu mjeriti temperaturu pomoću termoparova, zatim induktivitet, omogućiti spajanje s računalom, itd.

Digitalni multimetri, odnosno univerzalni instrumenti, danas su našli veliku primjenu zbog svoje preciznosti i niske cijene, ali analogni se još uvijek koriste kod mjerenja gdje je potrebno motriti trendove pada ili rasta mjerenih vrijednosti, odnosno brzo mijenjajuće vrijednosti. Prednosti numeričkog očitanja, odnosno korištenja digitalnih multimetara, ima nekoliko. Digitalni multimetri u pravilu imaju znatno veću točnost od analognih, očitanje je jednoznačno pa je isključena ljudska pogreška (paralaksa ili očitanje s pogrešne skale analognog instrumenta), veća brzina mjernog procesa u slučaju očitanja velikog broja mjerenja, automatski odabir mjernih područja (ukoliko instrument posjeduje ovu mogućnost), pohrana i daljnja obrada računalom uz odgovarajuće sučelje, automatizacija mjerenja, itd.



Sl. 2.1: *Blok shema digitalnog multimetra[1]*

Glavni dio digitalnog multimetra i svakog digitalnog mjernog instrumenta je analogno digitalni pretvornik koji analogni signal pretvara u digitalni. Kada je riječ o mjerenju napona, A/D pretvornik i brojilo se koriste za pretvorbu ulaznog napona u BCD digitalnu riječ.[1] Ulazni analogni signal mora biti gotovo nepromjenjiv zbog vremena potrebnog za analogno-digitalnu pretvorbu, a to je razlog zašto on mora biti istosmjeran. Ako se radi o izmjeničnom naponu, on se pomoću ispravljača pretvara u istosmjerni napon vrijednosti ekvivalentne izmjeničnom ulaznom naponu, a ispravljač daje odziv na srednje ispravljene, efektivne ili tjemene vrijednosti ulaznog izmjeničnog napona. Njegova točnost nije na razini digitalnog mjernog sustava za istosmjerni napon pa je ukupna točnost izmjeničnih mjerenja u pravilu manja od istosmjernih.

Mjerenje struje se odvija tako da se određuje napon na otporniku poznatog iznosa. Pri tome treba voditi računa da se mjerenje započne na najvišem mjernom području, a nakon toga postupno prelaziti na niža područja dok se ne dobije prikaz s najvećim brojem znamenaka.

Multimetar može poslužiti kao omometar kada se u njega ugradi vrlo točan strujni izvor koji tjera struju kroz mjereni otpornik, a digitalnim voltmetrom se mjeri pad napona na njemu.

Kod analognih instrumenata istosmjerne vrijednosti se mjere instrumentom s pomičnim svitkom i otpornicima dodanim u seriju i paralelu. Ispravljač je također ugrađen za mjerenja izmjeničnih veličina. Pogreške prilikom mjerenja s analognim instrumentima nastaju zbog nesavršenosti izvedbe instrumenta, utjecaja okoline te subjektivnoj procjeni očitanja sa skale.

Osciloskop je elektronički instrument koji osigurava promatranje različitih signala (raznih oblika) na svom zaslonu, kao 2D prikaz jednog ili više signala u vremenskoj domeni. Važan je instrument za analizu signala u vremenskom području. Osim napona, na njegove ulazne priključke se mogu dovesti i druge veličine koje se pretvaranjem u napon mogu prikazati na zaslonu, poput zvuka ili vibracije. Glavna podjela je na osciloskope s katodnom cijevi (CRO) i nešto modernije digitalne osciloskope (DSO).

Veliku uporabu imaju u znanosti, medicini, inženjeringu, itd. Najčešće su korišteni za održavanje elektroničke opreme i u laboratorijskim zadacima. Digitalni osciloskop također ulaznu veličinu pomoću analogno digitalnog pretvarača pretvara u digitalni oblik za prikaz na zaslonu, gdje se mogu promatrati amplituda, frekvencija, vremenski interval i razni drugi parametri. Sličnu primjenu nalazi i u laboratorijskim vježbama gdje studenti promatraju jedan ili više signala s ulaza te se bave njegovom analizom, promatranjem valnog oblika, ispitivanjem opreme ili dijela strujnog kruga, uz mogućnost mijenjanja vrijednosti amplitudne i vremenske podjele na skali zaslona, itd.

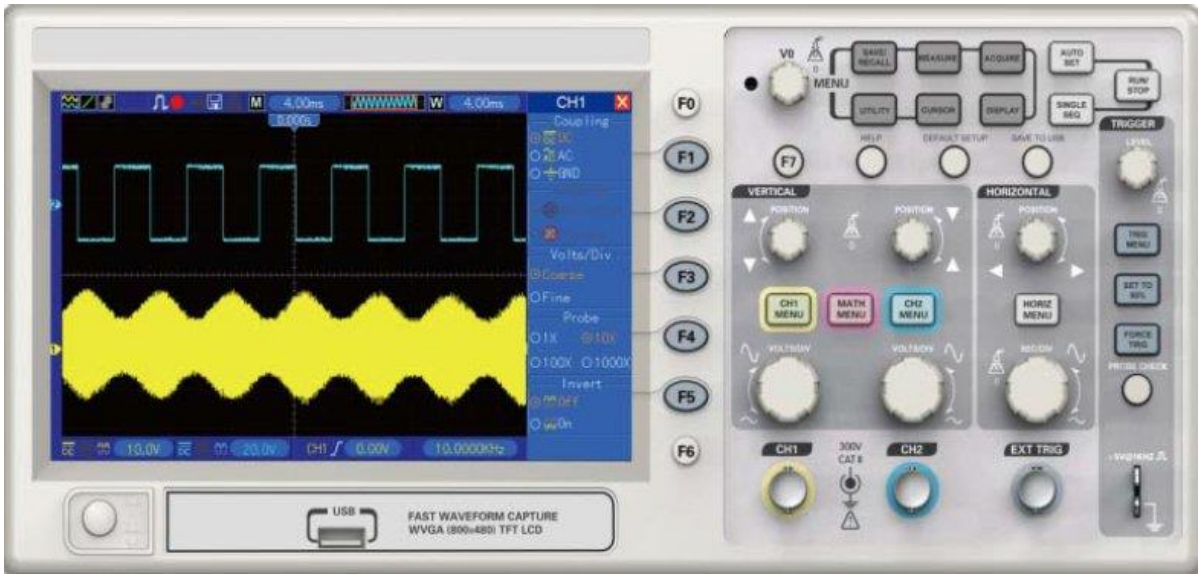
Analogni osciloskopi imaju određene nedostatke poput niže točnosti, većih troškova posjedovanja te ograničenih mjernih mogućnosti. Zato su svoju primjenu u mnogim laboratorijima pronašli digitalni osciloskopi čija cijena više nije tako visoka. Prednosti su mogućnost pohrane slika sa zaslona, velika točnost prilikom mjerenja, mogućnost spajanja računala i/ili pisača, mogućnost obrade valnog oblika uključujući matematičke funkcije kod rada s valnim oblicima, automatsko kalibriranje, itd.

Osciloskopom se može mjeriti vrijednost napona od vrha do vrha i to tako da se očitani broj podjela po vertikali pomnoži s vrijednosti koeficijenta vertikalnog otklona, a efektivna vrijednost se računa ovisno o valnom obliku. Osim mjerenja napona, moguće je izmjeriti i struju pomoću pada napona na poznatom otporu. Ako korištenje ove metode nije moguće, koriste se strujna kliješta (mjerni članovi).

Za potrebe zadatka je korišten multimetar Voltcraft VC250. Može mjeriti istosmjerne i izmjenične struje i napone, a između ostalog može poslužiti i za mjerenje otpora te testiranje dioda.[2]

Kontrolni multimetri i osciloskop služe kao uređaji za atestiranje mjernih članova, odnosno DAQ kartice i sučelja za prikupljanje podataka, te su približni pokazatelj vrijednosti koje (uz poznavanje karakteristika mjernih članova) programer može očekivati na ulazu u DAQ karticu.

Kako bi se provjerila ispravnost sučelja i točnost valnih oblika koji su na računalu dobiveni korištenjem DAQ kartice, u radu je korišten osciloskop čije su priključnice spojene s akvizicijskim sustavom i koji će poslužiti kao osciloskop za atestiranje ispitivanog sustava (DAQ sustava). Radi se o Voltcraftovom modelu DSO-1062D.



Sl. 2.2: *Osciloskop Voltcraft DSO-1062D.[3]*

2.2. Strujni i naponski mjerni članovi

Strujni mjerni članovi se prema načinu mjerenja dijele na beskontaktne i kontaktne.

Beskontaktni mjerni članovi koriste magnetsko polje oko vodiča protjecanog mjerenom strujom za detekciju i mjerenje. Primjer ove vrste mjernih članova su Rogowskijev svitak i strujni mjerni transformator.

Rogowskijev svitak je svitak namotan na nemagnetskoj jezgri stalnog presjeka i neprovodljivog materijala. Obuhvaćanjem vodiča kroz koji teče mjerena struja u svitku se inducira napon koji je proporcionalan promjeni struje po jedinici vremena. Integriranjem tog napona dobiva se napon koji je proporcionalan struji. U slučaju izmjenične struje, na izlaz integratora se mogu priključiti pretvornici s odzivom na efektivnu ili vršnu vrijednost struje.

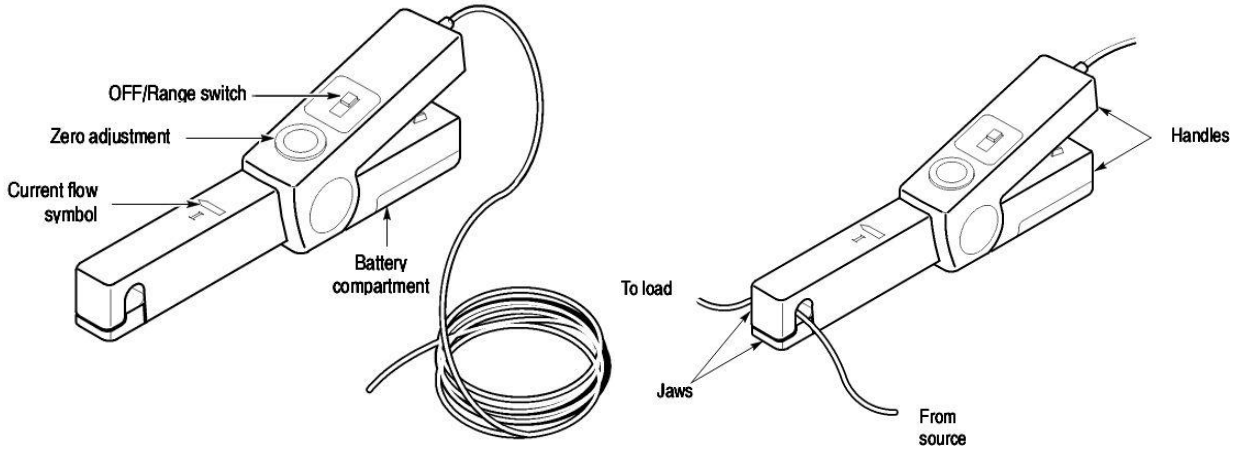
Strujni mjerni transformator se sastoji od prstena primjerice mekog željeza koji ima raspor. Namotaji žice se nalaze na jednoj ili obje strane prstena i tako čine jedan namot strujnog transformatora. Vodič koji se nalazi unutar tog prstena čini drugi namot tzv. transformatora, odnosno on se naziva primarom, a namotaji na prstenu sekundarom. Strujni mjerni transformator služi samo za mjerenje izmjeničnih vrijednosti struja.

Kontaktni mjerni članovi mogu koristiti pravilo Ohmovog zakona, što znači da mjere pad napona na poznatom otporu te se koriste kod mjerenja istosmjernih vrijednosti.

Princip rada Hallove sonde s druge strane se svodi na to da kroz magnetsko polje poznate vrijednosti protječe nepoznata struja (kroz Hallov element), dolazi do razdvajanja naboja i pojave induciranog napona proporcionalnog nepoznatoj struji i poznatoj jakosti magnetskog polja, a ovom izvedbom se mogu mjeriti istosmjerne veličine.

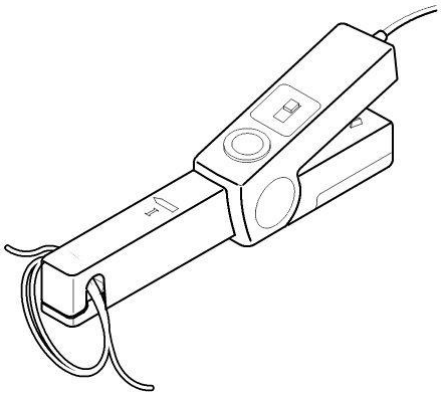
Na tržištu postoje razni tipovi i izvedbe strujnih mjernih članova: Picotech TA167 s dva stupnja osjetljivosti (1 mV/A, 10 mV/A) i mjernim opsegom 200 A/2000 A[4], Chauvin-Arnoux En serije širokih stupnjeva osjetljivosti i mjernih opsega[5], Metrel 92xx serija za industrijsku upotrebu, Fluke i310s i mnogi drugi. Većina strujnih mjernih članova radi na principu Hallovog efekta, a isto vrijedi i za Tektronixov strujni mjerni član A622 kakav se nalazi u laboratorijima FERIT-a.[6] Djelovanjem Lorentzove sile, u poluvodičkoj pločici smještenoj unutar magnetskog polja i priključenoj u krug istosmjerne struje u poprečnom se smjeru javljaju napon i struja.[7] Prednosti korištenja članova s osjetnikom koji radi na principu Hallovog efekta su jednostavnost izvedbe, male dimenzije i masa, niska cijena, jednostavna ugradnja i nedostatak utjecaja susjednih faza na mjerenje.[8]

Član A622 može mjeriti izmjenične i istosmjerne struje vršnih vrijednosti od 50 mA do 100 A i frekvencija do 100 kHz.[6] Na izlazu daje 10 mV ili 100 mV za svaki mjereni amper, a s DAQ karticom se spaja pomoću BNC priključka.[6]

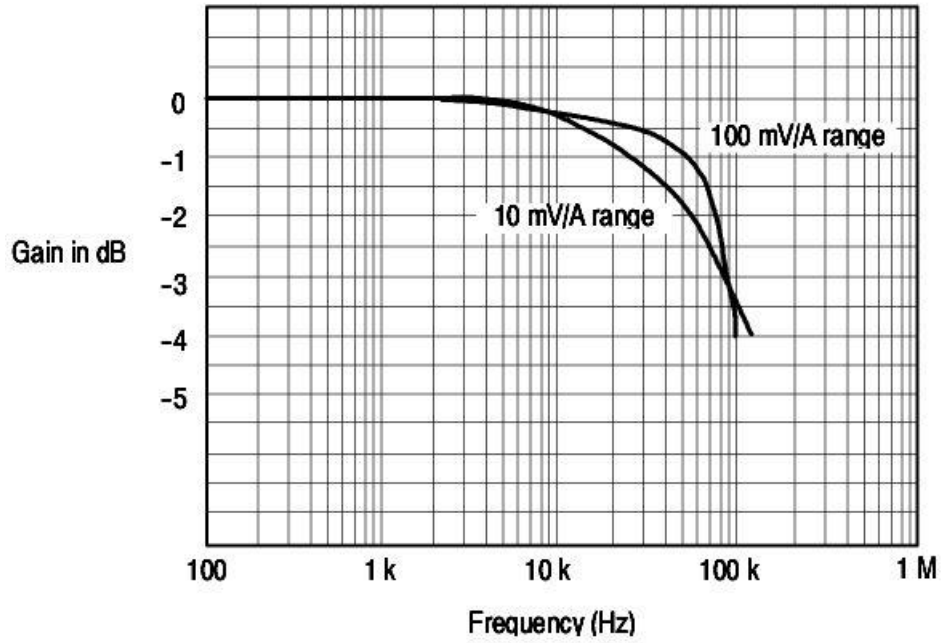


Sl. 2.3: *Strujni mjerni član Tektronix A622.[9]*

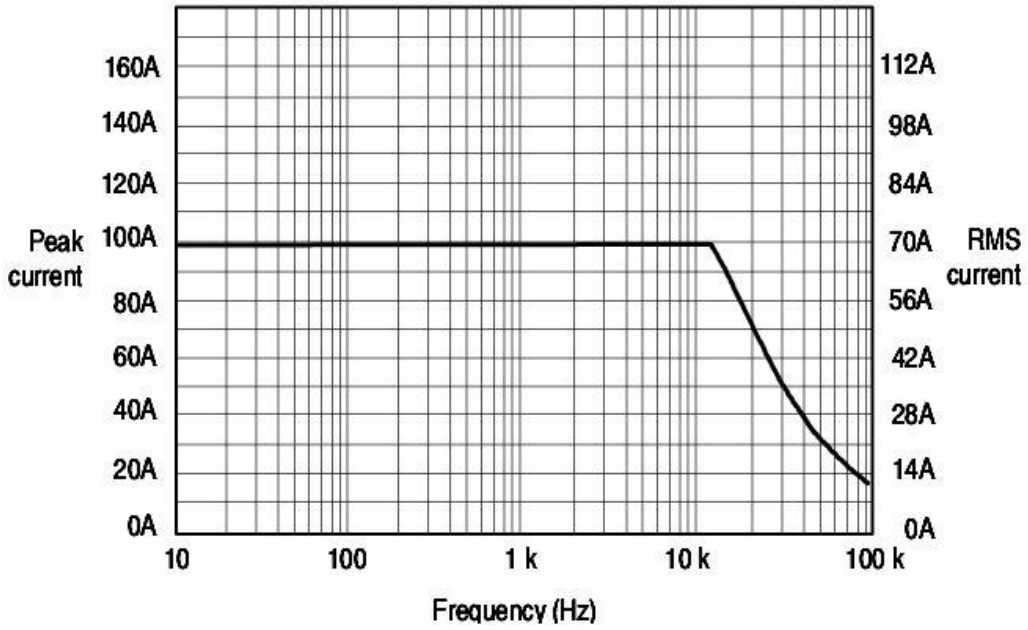
Kako bi se povećala osjetljivost strujnih mjernih članova u slučaju potrebe, potrebno je namotati nekoliko namota testnog vodiča kroz stezaljke mjernog člana, kao što je prikazano na slici 2.4. Osjetljivost mjernog člana A622 se povećava onoliko puta koliko je dodano namotaja između njegovih stezaljki.[9] [1]



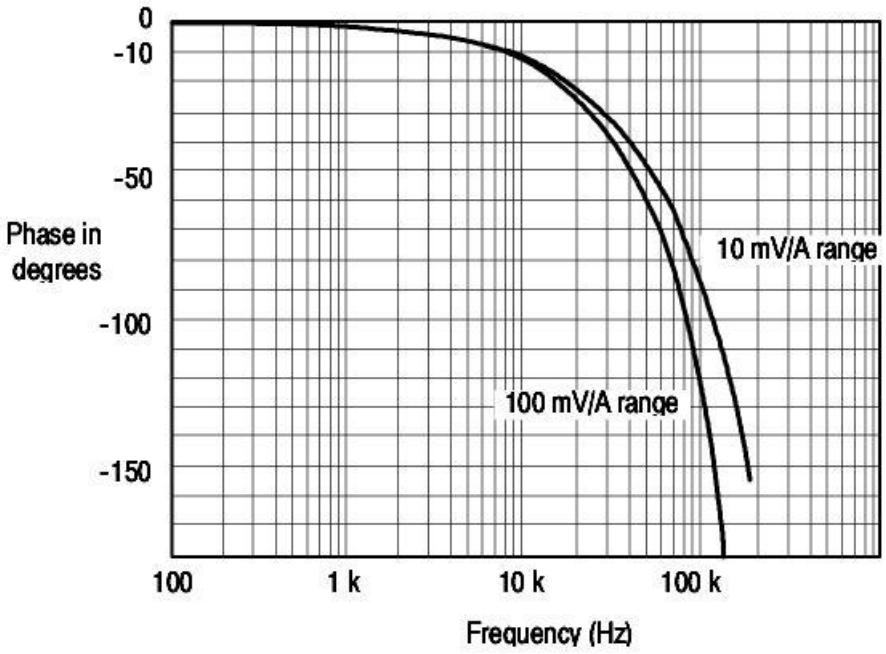
Sl. 2.4: *Povećanje osjetljivosti strujnog mjernog člana A622.[9]*



Sl. 2.5: *Frekvencijski odziv mjernog člana A622 pri struji od 1 A (vršno).[9]*



Sl. 2.6: *Maksimalna vrijednost struje u odnosu na frekvenciju mjernog člana A622.[9]*



Sl. 2.7: *Fazni pomak mjernog člana A622 pri struji od 1 A (vršno).[9]*

Naponski mjerni članovi se također dijele na kontaktne i beskontaktne.

Jedan od primjera kontaktnih naponskih mjernih članova su otpornička djelila koja se koriste kod mjerenja istosmjernih i izmjeničnih napona nižih frekvencija. Radi se o sklopu koji prigušuje ulazni napon, i to distribucijom ulaznog napona među mrežom otpornika koja čini djelilo. Jednostavan primjer čine dva otpornika spojena serijski i na koje djeluje ulazni napon, dok izlazni napon predstavlja pad napona na jednom otporniku.

Za mjerenje izmjeničnih napona upotrebljavaju se kondenzatorska djelila i naponski transformatori.

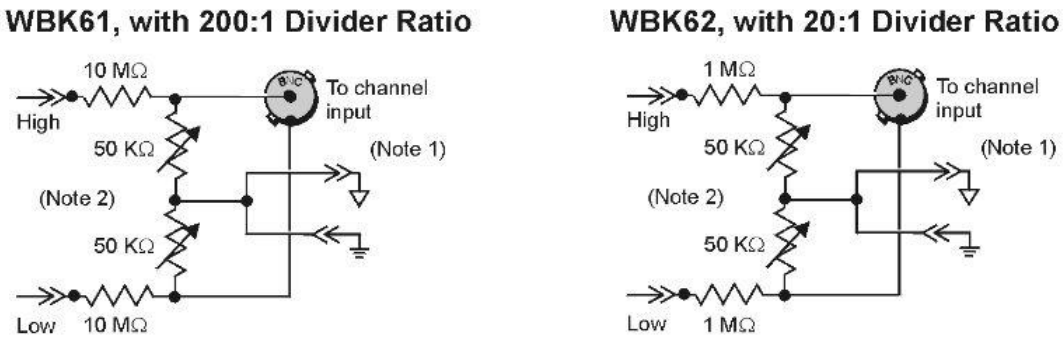
Naponski mjerni transformatori su beskontaktni mjerni članovi sačinjeni od magnetske jezgre i primarnog te sekundarnog namota. Primarni namot se spaja na visoki napon, dok se sekundar spaja na voltmetar. Naponi primara i sekundara proporcionalni su broju zavoja namota na primaru i sekundaru.

Za mjerenje napona u ovom radu korišteni su visokonaponski mjerni članovi WBK61 proizvođača Iotech. Omogućuju mjerenje napona do 1000 V te su idealni za mjerenja u trofaznim sustavima.[10] Radi se o jednokanalnim visokonaponskim adapterima koji se BNC priključkom spajaju na ulaze DAQ kartice. Oba (Lo i Hi) ulaza su opremljena otpornicima od 10 MΩ koji djeluju kao prigušivači u omjeru 200:1 i tako struju ograničavaju na 100 μA pri 1000

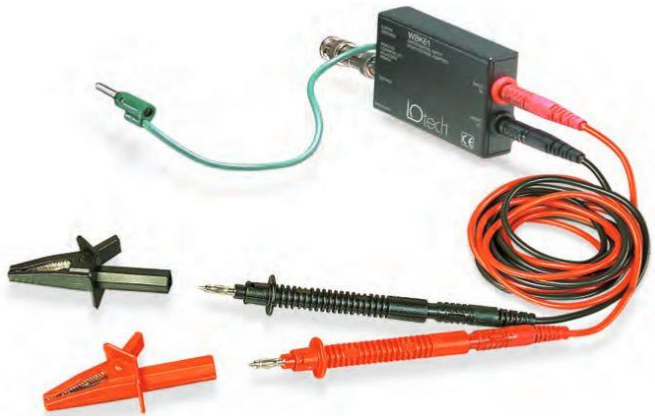
V.[10] Svaki WBK61 član ima i 4-milimetarski, tzv. banana ulaz za spajanje drugog mjernog člana (ili hardvera) na *earth ground* priključak.[10] Maksimalni napon na ulazu može biti 1000 V vršne vrijednosti (na bilo kojem ulazu u odnosu na *earth ground*), odnosno 2000 V vršne vrijednosti ako se ni na jedan ulaz ne dovodi napon veći od 1000 V vršne vrijednosti u odnosu na *earth ground*.[11]

Kod spajanja naponskih mjernih članova treba voditi računa o sljedećem: zeleni (banana) priključak prvog člana treba spojiti na *local earth ground*. U slučaju više WBK61 mjernih članova, ostali zeleni banana priključci se spajaju na *earth ground* priključak susjednih WBK61 mjernih članova, dok se *earth ground* posljednjeg člana spaja s uzemljenjem.[11]

Također, na tržištu se nalazi veliki broj različitih tipova i modela visokonaponskih mjernih članova koji mogu mjeriti istosmjerne i izmjenične vrijednosti, poput Tektronixovih P51xx modela, Amprobeovog HV231-10A, Flukeovog 80K-40 i mnogih drugih.

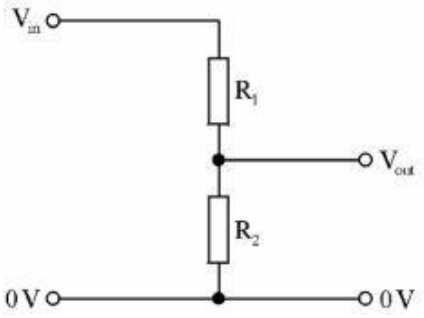


Sl. 2.8: *Shematski prikaz naponskih mjernih članova WBK61 i WBK62.[11]*



Sl. 2.9: *Naponski mjerni član WBK61.[10]*

Programer može i sam izraditi mjerne članove, tzv. 'uradi sam' ili DIY (engl. *Do It Yourself*) mjerne članove. Potrebno je napraviti otporničko djelilo, odnosno znati u kojem omjeru je potrebno izraditi član (koliko se želi smanjiti izlazni napon u odnosu na ulazni), pri čemu je potrebno poznavati DAQ karticu i njene specifikacije.



$$V_{\text{out}} = \frac{R_2}{R_1 + R_2} \cdot V_{\text{in}}$$

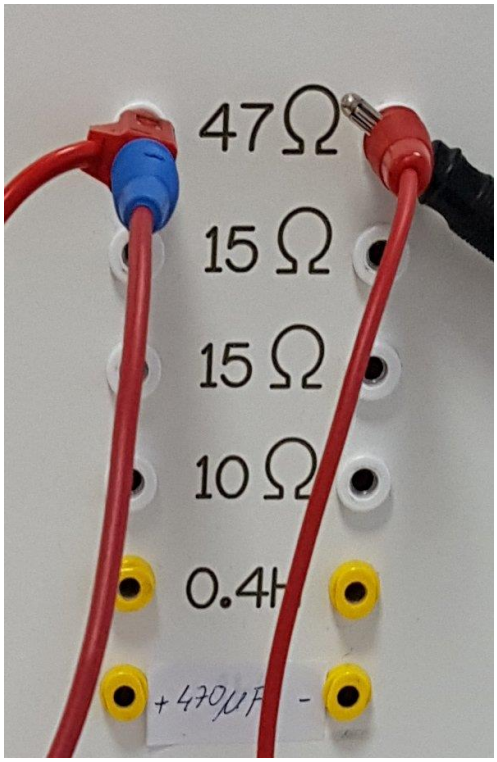
Sl. 2.10: *Shema 'uradi sam' otporničkog djelila i računanje izlaznog napona.*

Vin označava ulazni napon (visoki napon), dok *Vout* predstavlja izlazni napon tzv. DIY mjernog člana. Ako otpornik *R1* ima otpornost od 1 GΩ, a *R2* 1 MΩ, programer dobiva 1000:1 djelilo (zapravo 1001:1). Otpornost od 1 GΩ se, primjerice, može dobiti i serijskim spajanjem nekoliko otpornika manje otpornosti, a sve ih se vrlo jednostavno može ubaciti u plastičnu cijev manjeg promjera radi estetike, funkcionalnosti i sigurnosti.

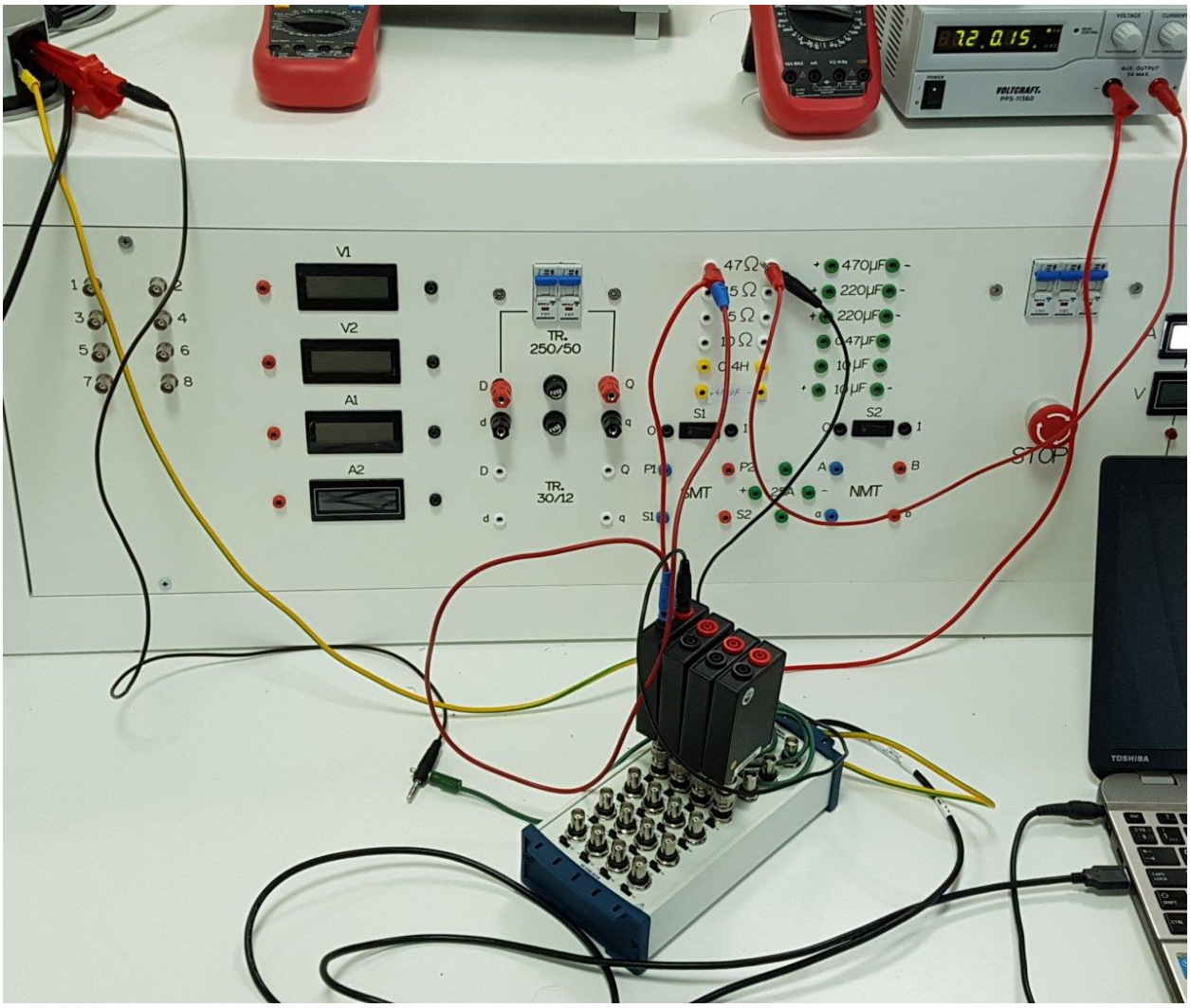
Kod rada s mjernim članovima je izuzetno važno poznavati njihove karakteristike, odnosno omjere ulaznih i izlaznih veličina njihovih osjetnika. Kod kreiranja sučelja za novi set mjerne opreme, potrebno je poznavati veličine ulaznih signala i ograničenja DAQ kartice, pa je od velike važnosti koristiti pravilan omjer (faktor dijeljenja) kod naponskih i strujnih mjernih članova kako bi mjerenja bila sigurna i točna.

2.3. Otpornici

Kao objekt na kojem će se provoditi mjerenja u cilju verifikacije sučelja za akviziciju podataka mjerenja odabrana su tri otpornika u laboratoriju fakulteta čije su otpornosti redom iznosile 47 Ω, 15 Ω i 10 Ω.



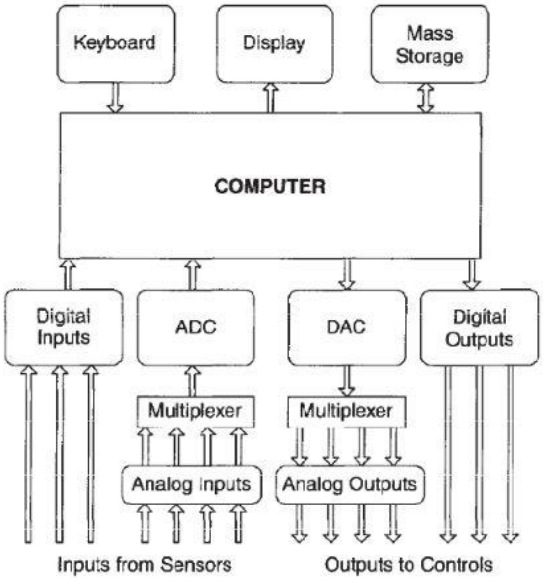
Sl. 2.11: *Otpornici u nastavnom laboratoriju.*



Sl. 2.12: *Spajanje digitalnih (kontrolnih) multimetara i naponskih mjernih članova s otpornicima.*

2.4. Akvizicija podataka DAQ karticama

Prvi korak u procesu akvizicije podataka je spajanje mjernih članova s hardverom koji će procesirati signale dobivene iz vanjskog svijeta. Taj zadatak pripada DAQ (engl. *Data Acquisition*) karticama. One imaju ulogu sučelja između okoline i računala koje prima signale iz te okoline. Tri ključne komponente za njihov rad su sklopovlje za prikupljanje signala, analogno-digitalni pretvarač i računalna sabirnica.[12] No, mnoge DAQ kartice imaju dodatne funkcije poput onih za automatizaciju mjernih sustava i procesa. Naprimjer, digitalno analogne pretvarače koje na izlazu opet daju analogne signale, brojače/tajmere koji broje i generiraju digitalne impulse.



Sl. 2.13: *Blok dijagram sustava za prikupljanje podataka.[13]*

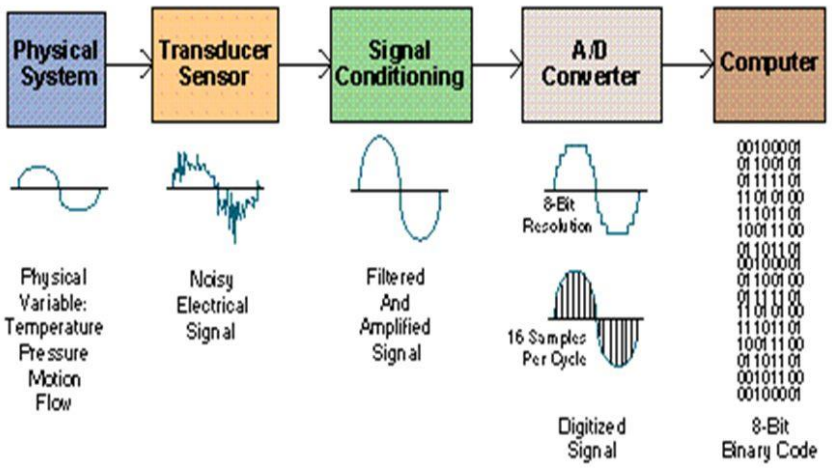
Sklopovlje za prikupljanje podataka kontrolira i po potrebi uređuje signale iz vanjskog svijeta koji mogu biti potencijalno opasni za direktno usmjeravanje na analogno digitalni pretvornik. Ti strujni krugovi mogu pojačati, smanjiti ili filtrirati dolazne signale.[14]

Analogno-digitalni pretvarač prikuplja ulazne analogne signale tako da ih uzorkuje u određenim vremenskim intervalima, zatim uzorkovane veličine kvantizira i na taj način se vrši digitaliziranje signala u oblik prikladan za računalo. ADC pretvarač (engl. *Analog-to-Digital Converter*) periodički uzima uzorke ulaznih veličina po unaprijed određenoj stopi. Odnosno, uzorkovanje, ili diskretizacija po vremenu, definira se kao postupak prikaza kontinuiranog signala slijedom brojeva koji predstavljaju signal u određenim vremenskim intervalima. Prema Nyquistovom teoremu o uzorkovanju signala,

$$f_N \geq 2 * fmax \tag{2-1}$$

slijedi da frekvencija uzorkovanja mora biti barem dvostruko veća od najveće očekivane frekvencije.[15]

Takvi uzorci se računalnom sabirnicom šalju u računalo gdje se prvotni signal rekonstruira uzorcima u računalnom softveru.

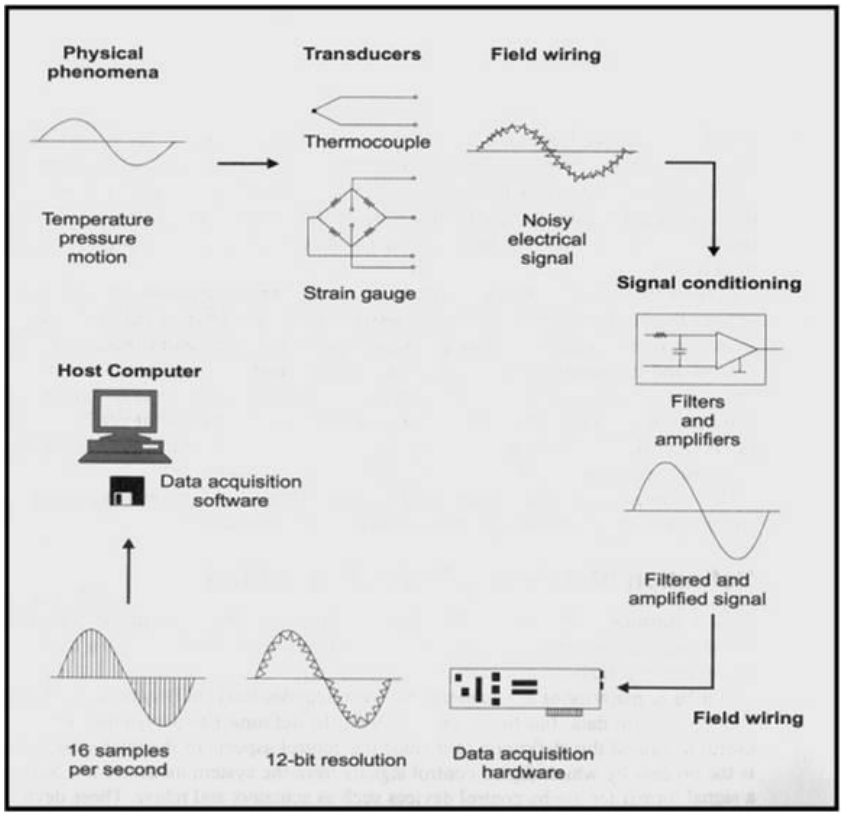


Sl. 2.14: *Pretvorba signala iz analognog u digitalni oblik.*

DAQ kartice (mjerne kartice) se s računalima spajaju kroz utore ili *portove*. Postoje serijski i paralelni protokoli spajanja, a sukladno tome i izvođenja programa u programskom jeziku. Serijsko povezivanje računalnim sabirnicama (RS232, RS485, USB) se razlikuje od paralelnog (GPIB, VXI, PCI). Kod serijskih sustava se podatak prenosi bit po bit kroz jedan komunikacijski kanal.[1] U paralelnim sustavima bitovi podataka se prenose istovremeno, odnosno paralelno kroz odgovarajući broj kanala.[1]

Prednosti serijskog načina spajanja su niži troškovi, neovisnost o platformi (MAC, PC, stolno ili prijenosno računalo), minimalni ili gotovo nikakvi zahtjevi za upravljačkim programima za sve programske jezike, uključujući BASIC, Visual BASIC, C, LabVIEW i druge, ili nepostojanje potrebe za vanjskim napajanjem kod spajanja DAQ kartice s USB računalnom sabirnicom. Nedostaci su niža brzina uzorkovanja i potreba za vanjskim napajanjem za slučaj prijenosa podataka RS232 ili RS485 vezom. Odabir sustava komunikacije u mjerom sustavu ovisi o raznim čimbenicima: potrebnoj brzini prijenosa, udaljenosti među dijelovima mjernog sustava, količini mjerne opreme i vrsti njihovih sučelja, uvjetima okoline, itd.[1]

Računalna sabirnica DAQ kartice služi kao komunikacijsko sučelje između DAQ kartice i računala u svrhu prebacivanja informacija i mjerenih podataka.[14] Prilično su kompatibilne te se nude s najčešćim računalnim sabirnicama uključujući USB, PCI, PCI Express i Ethernet, a u posljednje vrijeme s mogućnosti bežične komunikacije.[14]



Sl. 2.15: *Prikaz postupka prikupljanja podataka.*

Uloga računala, odnosno posebnog softvera i upravljačkih programa, u procesu akvizicije podataka ima jednako važnu ulogu jer bez odgovarajućeg softvera nema prikupljanja podataka s DAQ kartice. Programabilan softver osigurava njen rad te se koristi za procesiranje, vizualizaciju i pohranu mjerenih veličina. Za te potrebe mogu poslužiti Visual Basic, MATLAB ili LabVIEW.[12] Ako se programer odluči za Visual Basic, potreban je i dodatni softver NI Measurement Studio, a u radu mu također može pomoći upravljački program National Instrumentsa (NI-DAQmx).[16] Mathworksov program MATLAB također omogućuje prikupljanje podataka DAQ karticama, a za taj proces mu je potreban softver Data Acquisition Toolbox. Poput prethodna dva, i MATLAB podržava rad s akvizicijskim modulima National Instrumentsa.[17] S obzirom na to da je u radu korištena DAQ kartica proizvođača National Instruments, za rad na računalu je odabran programabilan softver LabVIEW koji također dolazi iz tvrtke National Instruments.

Upravljački program (engl. *driver*) je nezaobilazan dio za rad LabVIEW programa jer mu omogućuje interakciju s DAQ karticom. Pojednostavljuje komunikaciju s karticom za akviziciju podataka, a korisniku omogućava prikupljanje, analizu i prezentaciju mjerenih podataka. Za ovaj

završni rad je odabran upravljački program NI-DAQmx, još jedan proizvod National Instrumentsa, a može se pohvaliti jednakim funkcijama u LabVIEW-u kao i u Visual Basicu i nekim drugim aplikacijskim programima. U svom paketu sadrži DAQ Assistant, softver za jednostavniju konfiguraciju i izradu sučelja. Osim toga, već 10-ak NI-DAQmx funkcija može riješiti 80% zahtjeva za akvizicijom podataka.[18] NI-DAQmx kontrolira svaki aspekt DAQ sustava, od konfiguracije, programiranja u LabVIEW-u i kontrole rada same kartice. Kompatibilan je i sa Signal Expressom, NI LabWindows/CVI, C/C++, Visual Basicom, Visual Basic .NET i drugim programima. National Instruments preporučuje instalaciju ovog upravljačkog programa u isti direktorij gdje je instaliran LabVIEW programski paket.

2.4.1. Automatizirano mjerenje i kontrola

Akvizicija podataka predstavlja proces prikupljanja uzoraka signala iz mjeriteljeve okoline i pretvaranje tih uzoraka u digitalne signale, odnosno digitalne vrijednosti kako bi računalo nad njima moglo vršiti logičke i matematičke operacije. U te signale spadaju različite fizičke pojave pa se uz korištenje odgovarajućih mjernih članova DAQ karticama mogu mjeriti struje i naponi, ali i temperatura zraka, jakost svjetlosti, razina buke, sila i pritisak, pozicija i pomak, ubrzanje, itd.[14] To ovisi o mjernom članu koji se koristi za prikupljanje podataka, što može biti i termopar, termistor, otporni osjetnik, fotosenzor, mikrofon, elektrootporni tenzometar, piezoelektrični transduktor, potenciometar, linearni varijabilni diferencijalni pretvarač, optički enkoder, akcelerometar, itd.[14][19]

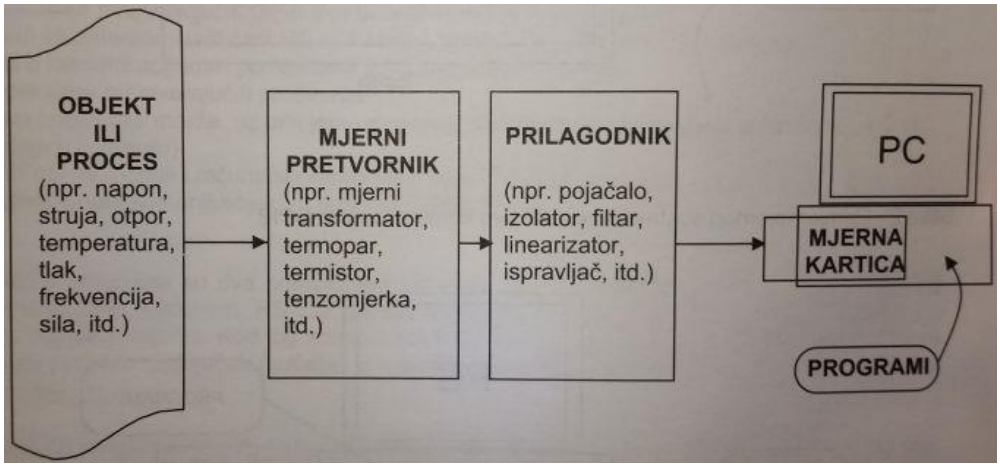
Kod mjerenja temperatura se najčešće koriste termoparovi, termistori i otporni osjetnici. Termopar je spoj dva različita metala koji mjeri razliku temperature, pa je za određivanje temperature potrebno poznavati referentnu temperaturu. Prednosti termoparova su mala cijena, široki temperaturni opseg (od -270 do 1700 °C), kratko vrijeme odziva, te točnost i ponovljivost. U nedostatke se ubrajaju niska osjetljivost, potreba za poznavanjem referentne temperature i nelinearnost.[20]

Termistori, odnosno temperaturno osjetljivi poluvodiči, dijele se na dvije vrste: termistori s negativnim temperaturnim koeficijentom (NTC) i termistori s pozitivnim temperaturnim koeficijentom (PTC). Veću primjenu imaju termistori s negativnim temperaturnim koeficijentom, čije su prednosti visoka točnost, visoka osjetljivost (10 puta bolja od otpornih osjetnika), male dimenzije, brži odziv od otpornih osjetnika, te stabilnost i ponovljivost. Nedostaci su ograničeni temperaturni opseg (od -40 do 150 °C) te nelinearni odnos otpora prema temperaturi.[20]

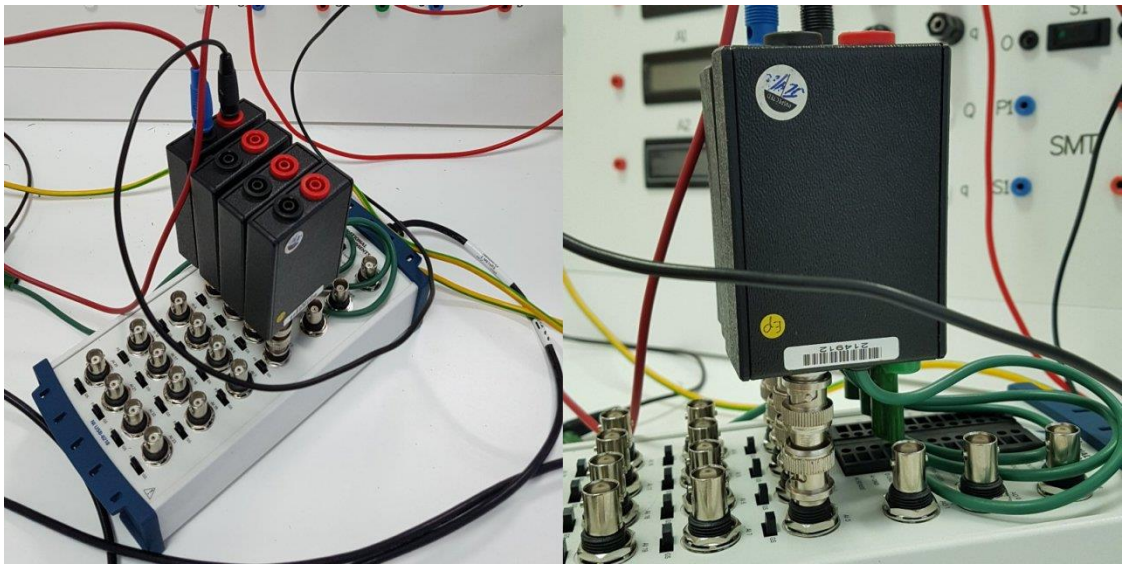
Otporni osjetnik (engl. *Resistor Temperature Detector*) također ima više vrsta. Mogu se koristiti u temperaturnom opsegu od -270 do 850 °C.[20]

Fotosenzor, mikrofon, poput ostalih mjernih članova, fizičku pojavu pretvaraju u električni signal. Postoje razni transduktori i mjerni senzori za različite primjene, a neki od njih mogu imati različite zahtjeve prilikom pretvorbe pojava u električni signal kojeg DAQ kartica može obraditi. Neki od tih zahtjeva mogu biti pobude na ulazu u obliku struja ili napona, dodatne komponente pa čak i mreže otpornika.[19]

Za ovaj završni rad su odabrani strujni i naponski mjerni članovi za mjerenje struje i napona trofaznog tronamotnog transformatora.



Sl. 2.16: *Blok dijagram automatiziranog mjernog sustava.[1]*



Sl. 2.17: *Mjerenje naponskim mjernim članovima.*

2.4.2. DAQ kartica NI USB-6218

NI USB-6218 je kartica za akviziciju podataka sa 16 analognih ulaza (16-bitna razlučivost) koja je u stanju primiti do 250 000 uzoraka u sekundi, s dva BNC analogna izlaza, 8 digitalnih ulaza, 8 digitalnih izlaza i do 12 digitalnih ulazno/izlaznih linija, te dva 32-bitna brojača.[21] Spajanje strujnih i naponskih mjernih članova se ostvaruje pomoću BNC priključnice na kartici, a veza s računalom pomoću USB priključka, što znači serijskim protokolom. Maksimalnih 250 000 uzoraka mjerenih veličina moguće je ostvariti s jednog ulaza ili više njih. U slučaju potonjeg, maksimalni broj uzoraka se dijeli s brojem ulaznih kanala pa će svaki kanal imati ograničenu maksimalnu stopu uzimanja uzoraka mjerenih veličina s ulaza DAQ kartice[15], iz čega slijedi:

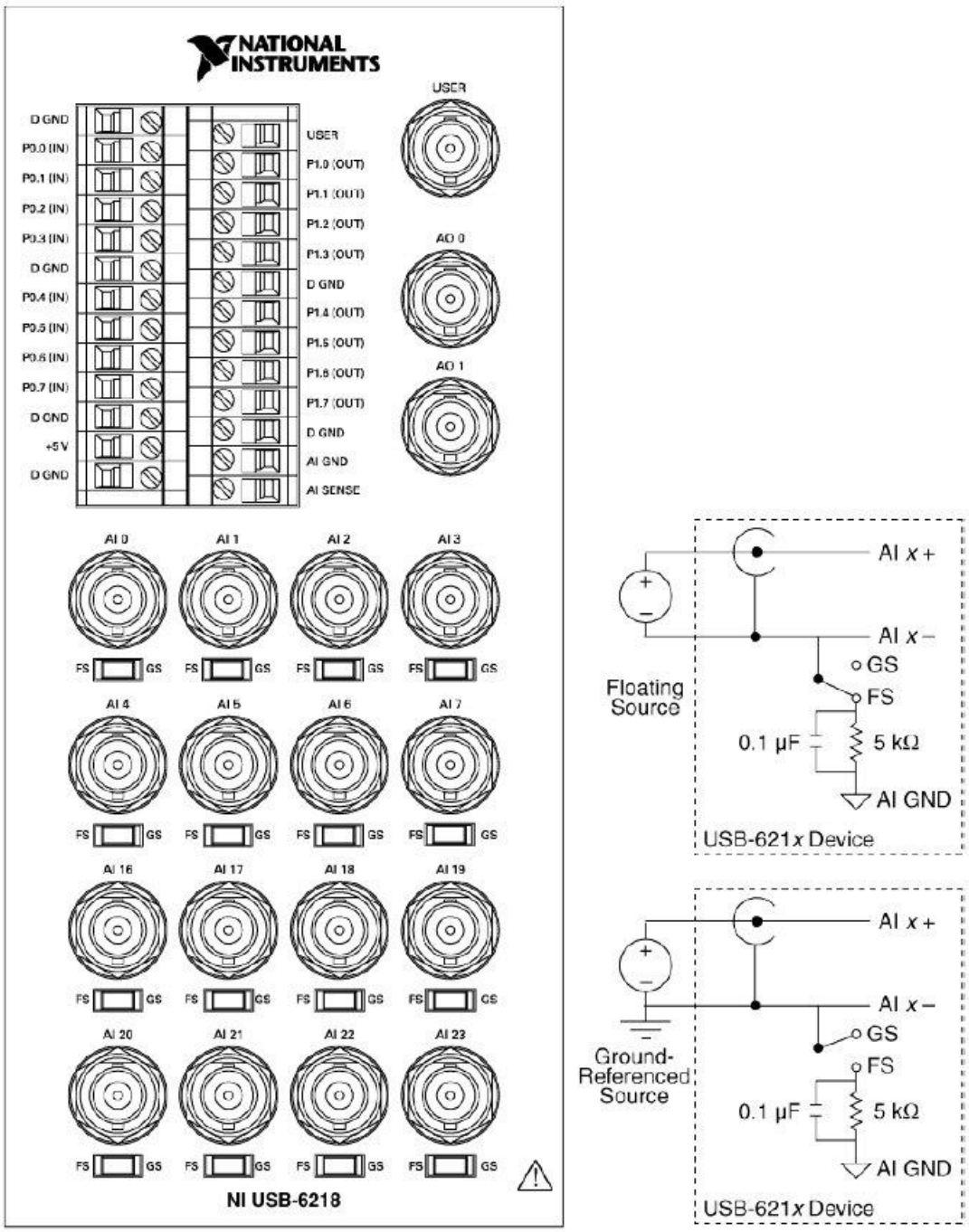
$$Nmax = \frac{fsmax * Ts}{ni} \tag{2-2}$$

gdje je:

- Nmax - maksimalni broj uzoraka pojedinog kanala
- fsmax - maksimalna frekvencija uzorkovanja
- Ts - vrijeme uzorkovanja
- broj kanala

Podržava LabVIEW-ov upravljački program NI-DAQmx te Measurement & Automation Explorer konfiguracijski softver koji dolazi s LabVIEW-om. Kartica je optimizirana za vrhunsku točnost pri prikupljanju velikog broja uzoraka u sekundi. Moguće je koristiti i 32 analogna ulaza ako se koristi tzv. single-ended način spajanja. USB-6218 je dizajnirana za korištenje naročito kod uvjeta prostorne ograničenosti ili potrebom za mobilnosti. Plug-and-play instalacija smanjuje vrijeme potrebno za konfiguraciju i uređivanje postavki, a kompatibilna je s raznim LabVIEW verzijama, uključujući verziju 7.1 pa sve do najnovije dostupne inačice.[22]

Sl. 2.18: *NI USB-6218.[22]*



Sl. 2.19: *Raspored priključaka kartice USB-6218.[22]*

NI USB-6218 kartica za akviziciju podataka ima rezoluciju od 16 bita, što znači da analogno digitalni pretvornik amplitudu ulaznog signala može podijeliti na 65536 razina (2^{16}). Najmanja vrijednost ulaznog napona koju analogno digitalni pretvarač može registrirati ovisi i o ulaznom opsegu, pa će tako za opseg ±0.2 V rezolucija iznositi 6.1 µV, a pri opsegu ±10 V 0.305 mV. Digitalne ulazno-izlazne linije se koriste za slanje digitalnih signala (stanje 1 ili stanje 0), a programer na izlazu dobiva određenu razinu napona kako bi odredio primljenu vrijednost.

Koriste se za kontrolu digitalnih signala ili upravljanje digitalnim uređajima kao što su prekidači ili LE diode.[19]

Brojači pretežno služe za brojanje događaja (pulseva, ciklusa...), mjerenje frekvencije, perioda, položaja, ili generiranja pulsa.[19]

2.4.3. Računalo s LabVIEW 2016 aplikacijskim softverom

Za potrebe nastave i znanstvenih istraživanja, te zbog svoje pristupačnosti, mogućnosti modificiranja i visoke kvalitete, vrlo se često koristi oprema National Instrumentsa koji je razvio programsku podršku za svoje proizvode za akviziciju podataka i upravljanje (LabVIEW) koja se bazira na posebno razvijenom programskom jeziku G i u sebi sadrži već razvijene potprograme za korištenje upravljačkih programa u svrhu povezivanja NI opreme, mjernih članova i računala.

• Prednosti

LabVIEW nudi veliku podršku za razne akvizicijske module i veliki broj upravljačkih programa. Tu su i velike biblioteke s različitim brojem funkcija za prikupljanje podataka, generiranje signala, matematičke i statističke operacije, alati za analizu itd., uz mnoštvo grafičkih elemenata koji dolaze s nekoliko opcija paketa za LabVIEW.[23] Programski jezik G omogućuje paralelno programiranje, odnosno izvršavanje nekoliko zadataka istovremeno, što je naročito važno kod automatizacije testnih sustava gdje do izražaja dolazi upravljanje različitim procesima.[23]

• Nedostaci

LabVIEW je vlasništvo tvrtke National Instruments pa za razliku od drugih programskih jezika, poput C-a ili FORTRAN-a, njegove značajke nisu određene ili specificirane u standardima trećih tijela ili instituta kao što su ANSI, IEEE, ISO, itd.[23] Unatoč tome što ima podršku za razne opcije i pakete za nadogradnju, njegova osnovna verzija ne sadrži alate koji su potrebni kod izrade zahtjevnijih i većih projekata. Uz ovo, LabVIEW-ov programski jezik programira aplikacije koje su sporije od onih čije se programiranje radi u jezicima kao što je C.[23] Veliki nedostatak je svakako nekompatibilnost sa starijim verzijama, što znači da blok dijagram kreiran u novijoj verziji LabVIEW-a nije moguće otvoriti u starijoj inačici (čak ni samo za pregled).[23] Ipak, aplikacijski softver nudi spremanje trenutnog projekta za raniju verziju. Još jedan neobičan nedostatak je nemogućnost povećanja ili smanjenja (engl. *zoom*) virtualnog instrumenta ili sekcije blok dijagrama[23], pa je male blok dijagrame teško analizirati s monitorima veće

rezolucije i većeg broja piksela po inču, dok problem predstavlja i složeni blok dijagram na zaslonu manje dijagonale.

3. IZRADA SUČELJA

Programski jezik G se temelji na protoku podataka (engl. *dataflow programming language*). Izvršenje programa je definirano strukturom grafičkog blok dijagrama u kojem programer koristi razne blokove te ih spaja crtanjem žica.[19][23] Svaki LabVIEW-ov program (virtualni instrument) ima tri komponente: blok dijagram, prednji panel (engl. *front panel*) i priključni panel (engl. *connector pane*).[19][23] Potonji, pojednostavljeno, reprezentira virtualne instrumente u blok dijagramu drugih, pozivajući virtualne instrumente.[19][23]

Prednji panel se sastoji od kontrola i indikatora. Kontrole su ulazi i programeru dozvoljavaju pružanje informacija virtualnom instrumentu.[23] Indikatori su izlazi, oni pokazuju rezultate na temelju ulaza koji su postavljeni virtualnom instrumentu.[23]

Stražnji panel, odnosno blok dijagram, sadrži grafički kod i funkcije koje vrše operacije nad kontrolama te isporučuju podatke indikatorima.[23] Funkcije se mogu pronaći u traci alata te se mogu postaviti u zaseban prozor na blok dijagramu. Svi objekti koji se nalaze na prednjem panelu će se naći i u blok dijagramu, odnosno obrnuto. Pritiskom kombinacije CTRL+E se jednostavno i brzo prebacuje iz prednjeg panela u blok dijagram i obrnuto, dva najvažnija dijela pri korištenju LabVIEW-a.

Potpune verzije LabVIEW-a dolaze s dodatkom LabVIEW Signal Express Full, softverom za lakšu konfiguraciju, razvoj sučelja i pohranu mjerenih veličina s DAQ kartice. Ako se s internetskih stranica National Instrumentsa preuzima *trial* verzija LabVIEW-a, potrebno je preuzeti i upravljački program NI DAQmx.

Instaliranjem LabVIEW-a se automatski instalira i NI Measurement&Automation Explorer, softver koji pruža pristup National Instruments, CAN, DAQ, FieldPoint, GPIB, IMAQ, IVI, Modular Instruments, Motion, NI Switch Executive, VI Logger, VISA i VXI uređajima. Ako se to kojim slučajem ne dogodi pri preuzimanje nekih od besplatnih verzija LabVIEW-a, NI Measurement&Automation Explorer se može preuzeti zasebno sa stranica National Instrumentsa.

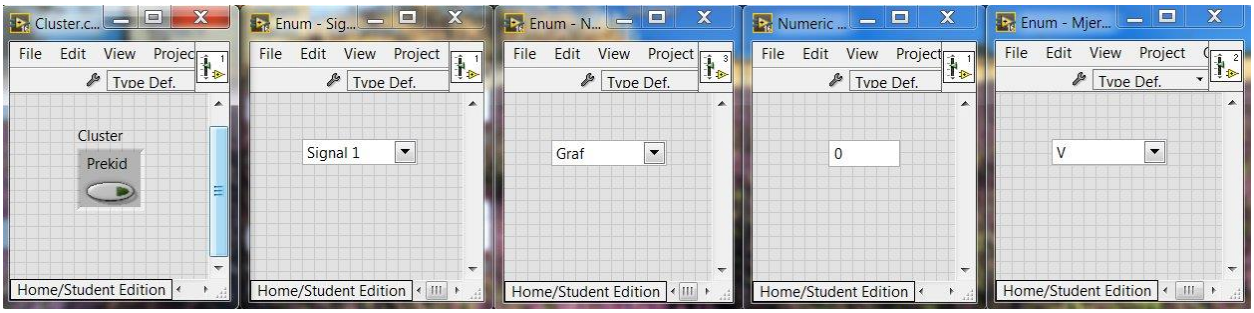
Measurement&Automation Explorer, skraćeno MAX, dozvoljava konfiguriranje NI hardvera i softvera, kreiranje i uređivanje kanala, zadataka, sučelja i virtualnih instrumenata, pogled na dijagnostiku sustava, uređaje i instrumente spojene na programerov sustav, te ažuriranje softvera National Instrumentsa.[19] Zasebno otvaranje MAX softvera nije bilo potrebno pri izradi ovog završnog rada.

Nakon LabVIEW-a je potrebno instalirati prethodno preuzeti upravljački program NI-DAQmx, a na zaslonu računala se nakon toga pojavljuju prečice LabVIEW-a i MAX-a.

Nakon otvaranja LabVIEW-a, potrebno je kliknuti na sljedeće: *File->New...->Custom Control->OK*. Potom se otvara prednji panel u kojem programer treba napraviti 5 kontrola (za cluster, ulazni signal(e), mjernu jedinicu, način prikaza i konstantu mjernog člana. To se postiže klikom desnom tipkom miša na prednjem panelu, a potom u paleti kontrola *Modern->Array, Matrix & Cluster* treba odabrati *Cluster* (za prvu navedenu kontrolu). Nakon toga je unutar dobivenog bloka u prednjem panelu ponovno potrebno kliknuti desnom tipkom, te kliknuti na sekciju *Modern->Boolean->Push Button*. Nakon toga slijedi klik desnom tipkom miša na sam *Cluster*, te kod opcije *Autosizing* treba odabrati način *Size to Fit*. Umjesto teksta *Boolean* pripadajuće kontrole unutar *Clustera* treba upisati 'Prekid', što se jednostavno odrađuje dvostrukom lijevom tipkom miša na ime i mijenjanjem naziva. Potom je u alatnoj traci umjesto *Control* potrebno odabrati *Type Def.*, a u priključnom panelu u gornjem desnom rubu kliknuti desnom tipkom miša i odabrati *Control Properties*, te u kategoriji *General* staviti kvačicu na opciju *Separate compiled file from source file* i potvrditi promjene prije izlaska iz izbornika. Nakon ovih radnji programer može izaći iz prozora, a pritom će novonastalu kontrolu spremiti u mapi *Sučelje->Kontrole* pod imenom 'Cluster'.

Nakon toga je potrebno slijediti slične korake kako bi se napravile još četiri kontrole. Za ulazni signal odabrati sljedeće: *System->Ring&Enum->System Enum* i postaviti na prednji panel. Nakon toga treba kliknuti desnom tipkom miša na samu kontrolu te odabrati: *Visible Items->Label* (maknuti kvačicu), te *Properties* nakon čega treba pronaći tab *Edit Items*, odabrati opciju *Insert* i redom upisivati jedan po jedan: 'Signal 1', 'Signal 2', ..., sve do 'Signal 8'. Također kao i kod prethodne kontrole, potrebno je odvojiti pokrenuti kôd od izvornog, u alatnoj traci odabrati *Type Def.* te kontrolu spremiti u istu mapu, ali pod nazivom 'Signali'.

Ostaju još tri kontrole koje se prave na isti način. Za mjernu jedinicu je potrebno u tabu *Edit Items* napisati samo 'V' i 'A', ovisno o tome hoće li se mjeriti napon ili struja, te kontrolu uz ostale navedene radnje spremiti kod imenom 'Enum - Mjerna jedinica', za vrstu prikaza 'Enum - Način prikaza' u tabu *Edit Items* dodati 'Graf' i 'Vrijednost', što znači da će se kasnije u sučelju moći pratiti valni oblik mjerenog signala ili će programer biti u mogućnosti pogledati brojčane vrijednosti pripadajućeg signala s ulaza. Za konstantu mjernog člana treba odabrati *System->Numeric->System Numeric*.



Sl. 3.1: *Kontrole sučelja za akviziciju podataka*

Sljedeći virtualni instrument potrebno je spremiti u novu mapu unutar glave mape 'Sučelje', a u ovom slučaju mapa nosi ime 'Međuspremnik', dok sam virtualni instrument nosi naziv 'Spremnik podataka o signalima'.

Za njegovu izradu je potrebno otvoriti LabVIEW i odabrati *File->New VI*. Nakon toga se otvaraju blok dijagram i prednji panel, a ovaj virtualni instrument poslužit će kao međuspremnik podataka prije nego što se oni dalje proslijede na virtualni instrument koji na sučelju prikazuje one prikaze koje to korisnik želi (valni oblik ili brojčana vrijednost), o čemu će više riječi biti kasnije.

U blok dijagramu se funkcijama za izradu virtualnog instrumenta pristupa desnom tipkom miša bilo gdje na praznom dijelu blok dijagrama, nakon čega se otvara paleta s raznim mogućnostima. Sama paleta nosi naziv 'Paleta funkcija', tako da će u nastavku teksta ovaj postupak tako i oslovljavati.

U paleti funkcija potrebno je odabrati četiri petlje/strukture te ih posložiti onako kako to prikazuje slika 3.2 (*case* struktura unutar *for* petlje koja se nalazi unutar *case* strukture, koja je opet postavljena unutar *case* strukture, dok se potonja nalazi unutar *while* petlje). U paleti se tim alatima pristupa klikom na *Programming->Structures->While Loop* (jednom), odnosno na *Case Structure* i *For Loop*. Na svaku od ovih pet struktura/petlji potrebno je kliknuti desnom tipkom miša i maknuti kvačicu s opcije *Auto Grow*.

Potom se treba prebaciti u prednji panel i u paleti kontrola odabrati *Modern->Array, Matrix & Cluster->Array* i postaviti na praznu površinu panela, a zatim *Modern->I/O->Waveform* i potonji blok ubaciti unutar bloka *Array*. Klikom desnom tipom miša u sredinu gdje je oznaka *t0* dobiva se padajući izbornik u kojem treba odabrati *Visible Items->Time/Date Browse Button*. U blok dijagramu potrebno je napraviti sljedeće: desni klik miša na blok *array->View As Icon*

(maknuti kvačicu), a naziv promijeniti iz 'array' u 'Podaci' (dvostruki klik lijevom tipkom miša na ime). Zatim, u paleti funkcija odabrati *Programming-> Boolean->False Constant*, te nakon toga napraviti klik desnom tipkom miša na potonji i promijeniti u kontrolu klikom na *Change to Control*, te odabrati *View as Icon* (maknuti kvačicu). Sam blok potrebno je preimenovati u 'Reset'. Nadalje, u paleti funkcija treba odabrati *Programming->Numeric-Enum Constant* (također promijeniti u kontrolu kao u slučaju s *False Constant* i maknuti kvačicu s opcije *View As Icon*). Za kraj je potrebno preimenovati blok u 'Metoda', a nakon odabira svojstava, u tabu *Edit Items* dodati opcije 'Read' i 'Write', kao i ranije kada su se kontrolama na ovaj način dodavale mogućnosti u ovom tabu.

Potom u prednjem panelu u paleti kontrola treba odabrati *Modern->Array, Matrix & Cluster->Array*, a zatim *Modern->I/O->Waveform* i potonji blok ubaciti unutar bloka *Array* (slično kao i ranije). Nakon klika desnom tipkom miša odabrati opciju *Change to Element*. U blok dijagramu s ovog bloka treba maknuti kvačicu s opcije *View As Icon* te ga preimenovati u 'Read'. U paleti funkcija potrebno je odabrati *Programming->Array->Index Array* te ga postaviti na blok dijagram kao na slici 3.2., iz njegovog čvora *index* (donji lijevi kut tog bloka dovesti novu kontrolu tako što desnom tipkom miša treba kliknuti na taj čvor i odabrati opciju *Create->Control* (također maknuti kvačicu s *View As Icon*).

Nakon ovih koraka u paleti funkcija je potrebno odabrati sljedeće blokove te ih povezati kao na slici 3.2., a to su:

Programming->Comparison->Empty Array?,

Programming->Boolean->Or,

Programming->Waveform->Get Wfm Comps (2 puta),

Programming->Array->Build Array (obuhvatiti donji rub i povećati za jedan ulaz kao što je prikazano slikom 3.3 (povući mišem prema dolje)) i nakon toga kliknuti desnom tipkom miša i odabrati *Concatenate Inputs,*

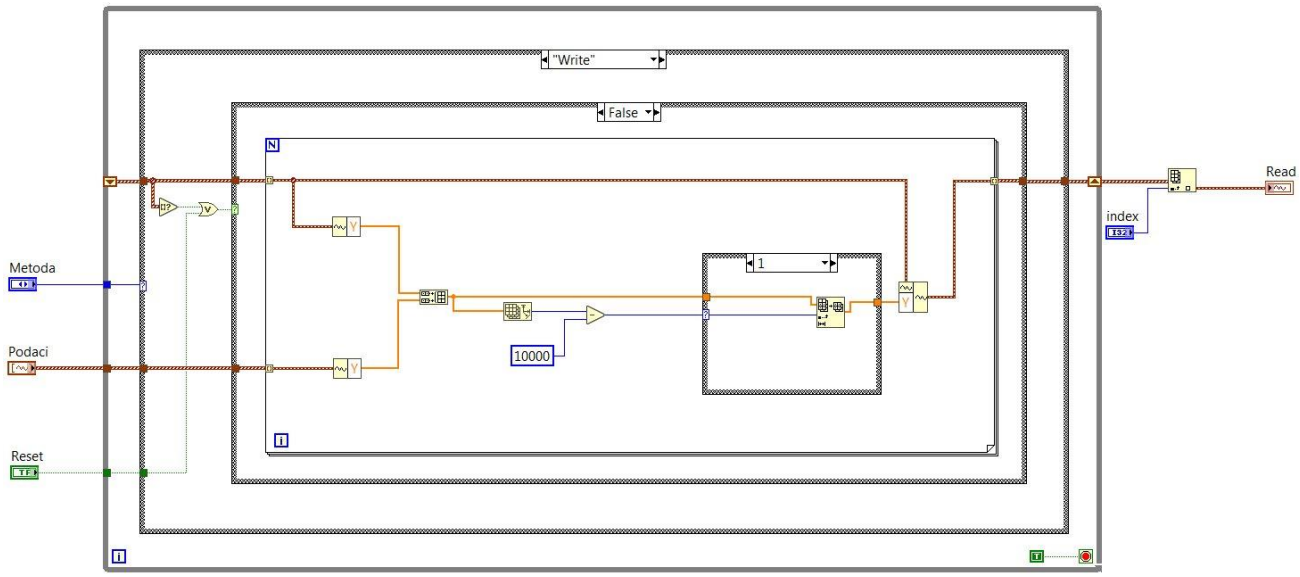
Programming->Array->Array Size,

Programming->Numeric->Subtract (na njegov drugi ulaz kliknuti desnom tipkom miša i odabrati *Create->Constant*, upisati '10 000' i zatim kliknuti desnom tipkom miša na samu konstantu i odabrati *Representation->I32*),

Programming->Waveform->Build Waveform,

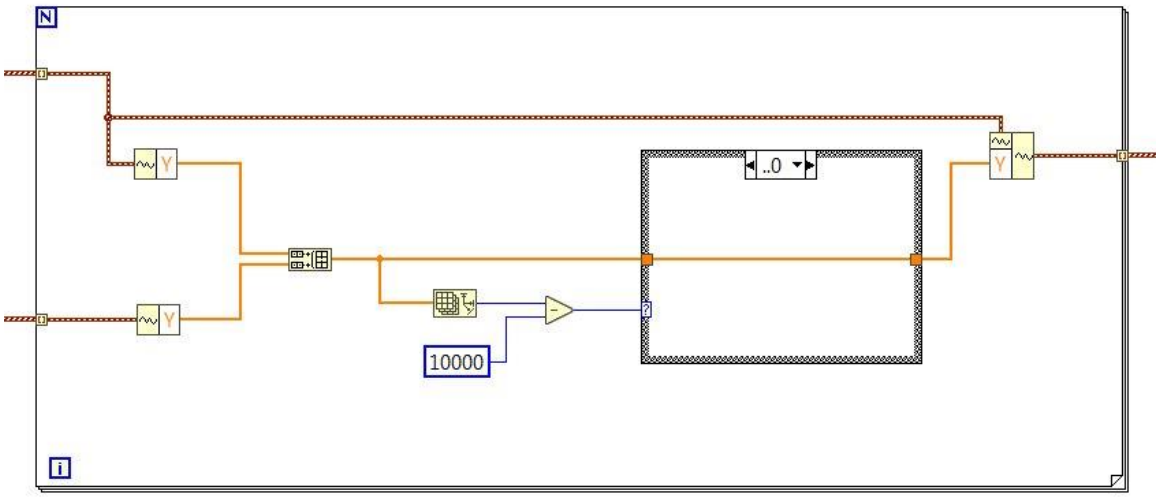
Programming->Boolean->True Constant.

Sljedeći korak je spajanje blokova kao što je prikazano na slici 3.2, s lijeva prema desno. U najmanjoj *case* strukturi će se prilikom spajanja sa *Subtract* blokom pojaviti promjena i broj '1' kao jedan od slučajeva, a unutra je potom potrebno postaviti sljedeći blok iz palete funkcija: *Programming->Array->Array Subset*. Dalje je potrebno nastaviti spajati kako je prikazano na slici 3.2. Na mjestu spoja žice s *while* petljom na desnoj strani nalazi se mali kvadratić na koji je potrebno kliknuti desnom tipkom miša i odabrati *Replace with Shift Register*, a novonastalu strelicu treba postaviti na lijevu stranu *while* petlje kako je prikazano na slici 3.2, te to mjesto također spojiti kako je prikazano na slici 3.2.



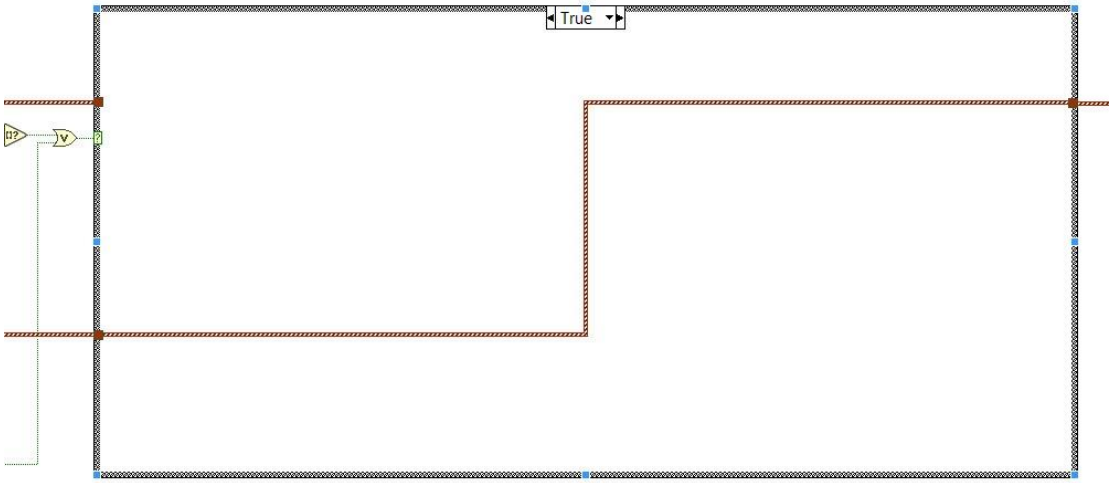
Sl. 3.2: *Blok dijagram virtualnog instrumenta 'Spremnik podataka o signalima'.*

Nakon toga je u unutarnjoj *case* strukturi potrebno iz padajućeg izbornika odabrati slučaj *0* te povezati lijevu i desnu stranu kako je prikazano na slici 3.3. Prije oznake *0* u unutarnjoj strukturi treba dodati dvije točkice, također kao što je prikazano na slici 3.3, dok za slučaj *1* treba dodati dvije točkice nakon broja 1.

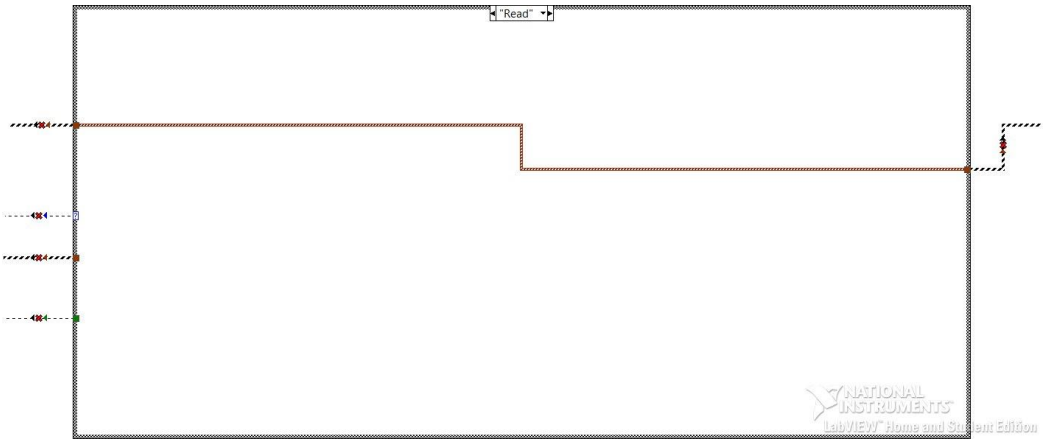


Sl. 3.3: *Najmanja case struktura u blok dijagramu 'Spremnik podataka o signalima'.*

Prva *case* struktura koja se nalazi izvan *for* petlje za slučaj *True* treba biti povezana kao na slici 3.4., a najveća *case* struktura za slučaj *Read* kao na slici 3.5.

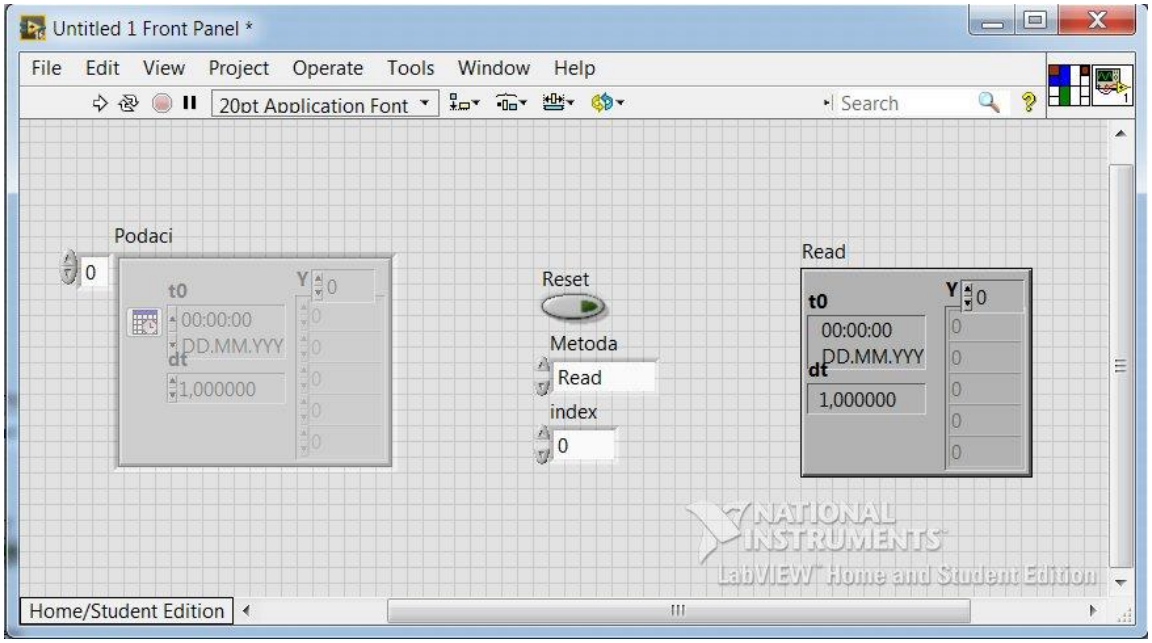


Sl. 3.4: *Srednja case struktura u blok dijagramu 'Spremnik podataka o signalima'.*



Sl. 3.5: *Najveća case struktura u blok dijagramu 'Spremnik podataka o signalima'.*

Na svakoj *case* strukturi treba odabrati sve moguće slučajeve, te ako je neki od njih postavljen kao zadani (engl. *default*), potrebno je kliknuti desnom tipkom miša na taj slučaj i kliknuti na *Remove Default*. Nakon rada u blok dijagramu potrebno se prebaciti u prednji panel i povezati kontrole i indikatore s *connector paneom*. Ovo, kao i izgled prednjeg panela, prikazano je na slici 3.6.



Sl. 3.6: *Prednji panel virtualnog instrumenta 'Spremnik podataka o signalima'.*

Povezivanje s *connector paneom* se vrši tako da se najprije klikne lijevom tipkom miša na mali kvadratić u *connector paneu* koji se nalazi u lijevom gornjem rubu. Nakon toga on poprima crnu boju, a programer potom treba kliknuti na kontrolu s lijeve strane naziva 'Podaci', nakon čega ranije spomenuti kvadratić u *connector paneu* dobiva ljubičastu boju. Krajnji desni kvadratić se istom metodom dodjeljuje indikatoru *Read*, plavi kvadratić na slici 3.6 (skroz lijevo) spojiti s kontrolom *Metoda*, unutarnji plavi kvadratić sa slike 3.6 spojiti s kontrolom *index,* a donji (zeleni kvadratić sa slike 3.6) spojiti s kontrolom *Reset*. Posljednje što preostaje jest desnom tipkom miša kliknuti na *connector pane,* zatim odabrati *VI Properties*, a u padajućem izborniku *General* staviti kvačicu na opciju *Separate compiled code from source file*. Sada programer može zatvoriti virtualni instrument i u mapi *Međuspremnik* ga pohraniti pod imenom 'Spremnik podataka o signalima'.

Sljedeći korak u razvoju sučelja za akviziciju podataka je kreiranje nove mape naziva 'Pogledi', koja će biti smještena unutar glavne mape *Sučelje*. U njoj će se nalaziti osam virtualnih instrumenata za prikaz od jednog do osam signala u jednom prozoru, te jedan virtualni instrument naziva 'Obrada podataka'.

Nakon otvaranja novog virtualnog instrumenta u LabVIEW-u, programer može započeti s kreiranjem virtualnog instrumenta *Obrada podataka* čija je funkcija direktno povezana s ranijim potprogramom *Spremnik podataka o signalima*, ali i s prethodnom napravljenim kontrolama. Na *connector pane* potrebno je kliknuti desnom tipkom miša, a zatim odabrati *Add Terminal*. U paleti funkcija je potrebno dohvatiti jednu *while* petlju, jednu *case* strukturu, te još jednu *case* strukturu koje će se nalaziti unutar 'glavne' *case* strukture, kao što je prikazano na slici 3.7. U prednjem panelu treba kliknuti desnom tipkom miša na podlogu, i nakon toga potrebno je odabrati *Select a Control...,* i odabrati bilo koju od četiri kontrole s početka ovog poglavlja (bez *clustera*). Postupak treba ponoviti još tri puta kako bi sve kontrole bile prisutne u prednjem panelu i na blok dijagramu, a svakoj od kontrola treba se promijeniti ime (napisati njeno stvarno ime) i maknuti kvačica s opcije *View as Icon* u blok dijagramu.

Iz palete funkcija na blok dijagram treba postaviti sljedeće blokove:

Programming->Synchronization->Queue Operations->Dequeue Element (iz čvora *element* napraviti kontrolu pomoću klika desnom tipkom na taj čvor te odabirom *Create->Control*, uz sveprisutno micanje kvačice s opcije *View as Icon*, te iz čvora *timeout in ms (-1)* istim potezom dobiti konstantu u koju je potrebno upisati '250' (*Create->Constant*)),

Select a VI...->(pronaći ranije kreirani virtualni instrument Spremnik podataka o signalima *te ga odabrati)*, a zatim iz njegovog čvora *Metoda* napraviti konstantu i pobrinuti se da je označena opcija *Read*,

Programming->Numeric->Conversion->To Unsigned Long Integer,

Programming->Waveform->Get Wfm Comps,

Signal Processing->Wfm Measure->Amp&Level (potom staviti kvačice na mogućnosti *Mean (DC), RMS, Peak to Peak*),

Programming->Cluster, Class & Variant->Unbundle By Name,

Programming->Boolean->Or,

Programming->Numeric->Multiply.

Nakon toga je u prednjem panelu iz palete kontrola potrebno odabrati *System->Ring&Enum->System Enum*, te taj postupak ponoviti još dva puta (u blok dijagramu maknuti kvačicu s opcije *View As Icon*, promijeniti iz kontrole u indikator, te uključiti opciju *Adapt To Source*), dok je u blok dijagramu potrebno na svaki od ta tri indikatora kliknuti desnom tipkom miša i kod opcije *Visible Items* maknuti kvačicu s oznake *Label*. Iz palete kontrola (prednji panel) još je potrebno odabrati *Modern->Containers->Tab Control* (desni klik na *Tab Control* u blok dijagramu i potom maknuti kvačicu s *View As Icon*, te odabrati *Change to Indicator*, a u prednjem panelu u jedan tab upisati 'Graf', a u drugi 'Vrijednost'). Kod opcije *Visible Items* u prednjem panelu maknuti kvačicu s oznake *Label*.

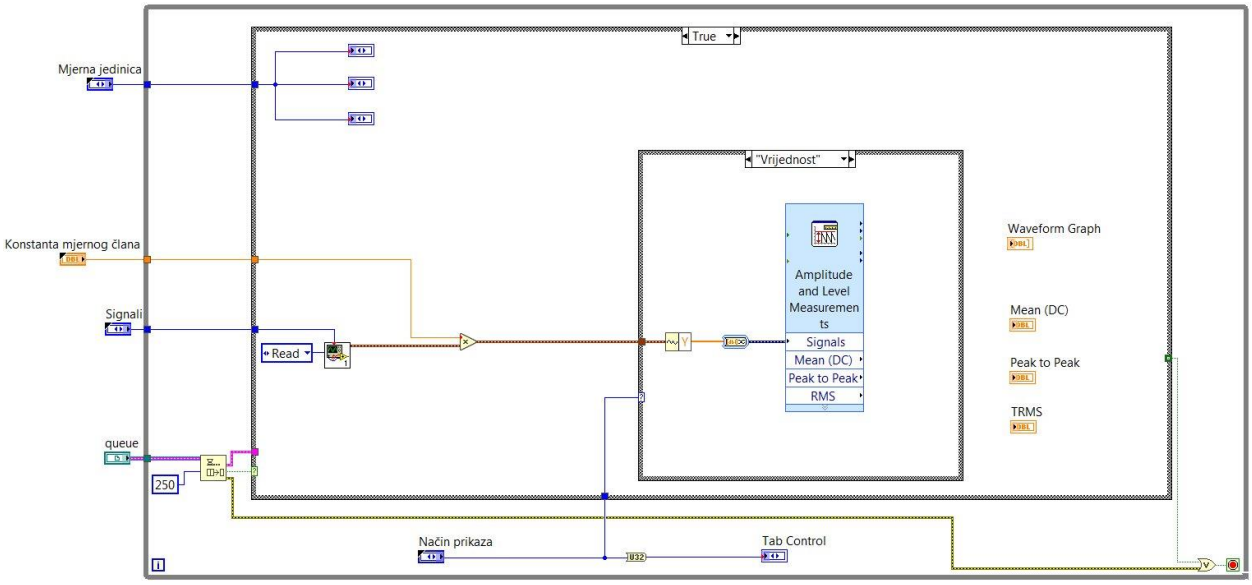
U prednjem panelu je na kontrolu *queue* potrebno kliknuti desnom tipkom miša i potom na *Show Control*, a zatim kliknuti desnom tipkom miša u bijelo polje i napraviti sljedeće: *Replace->Select a Control...* i poslije toga pronaći i odabrati kontrolu *Cluster* iz mape *Kontrole.*

Nakon toga je u blok dijagramu potrebno na svaku od opcija u bloku *Amplitude and Levels* (*Mean (DC), Peak to Peak, RMS*) kliknuti desnom tipkom miša s njegove desne strane, te odabrati sljedeće: *Create->Numeric Indicator*. Sa sva tri indikatora treba maknuti kvačicu s mogućnosti *View As Icon*, kao i u prethodnim slučajevima, radi bolje preglednosti i estetike, a kod indikatora *RMS* treba promijeniti ime u 'TRMS' jer LabVIEW-ov paket kod ovakvih mjerenja pruža točna TRMS očitanja za ovu mogućnost.

U prednjem panelu unutar taba *Graf* indikatora *Tab Control* potrebno je iz palete kontrola odabrati *Silver->Graph->Waveform Graph* (ne prikazivati kao ikonu u blok dijagramu). U drugi

tab *Vrijednost* potrebno je ubaciti tri numerička indikatora i tri indikatora mjerne jedinice kao što to prikazuje slika 3.8.

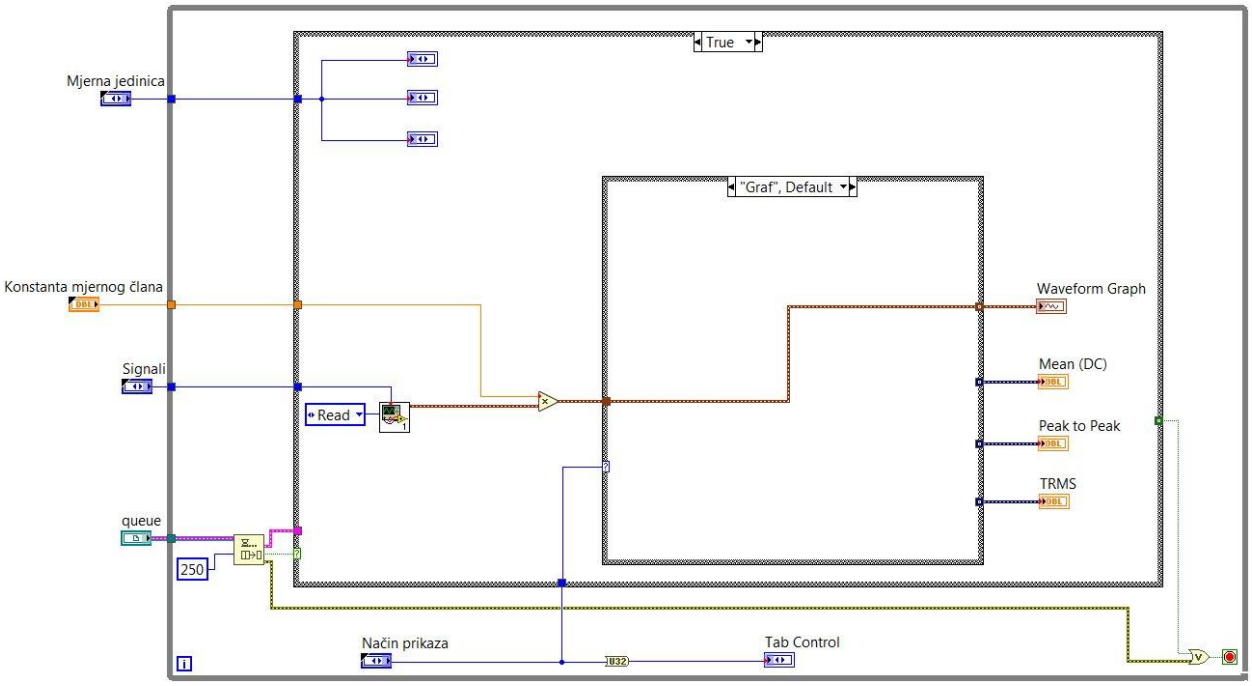
Nakon toga se treba vratiti u blog dijagram te numeričku kontrolu konstante mjernog člana povezati s blokom *Multiply*, a ostale čvorove potonjeg bloka kako je prikazano na slici 3.7.



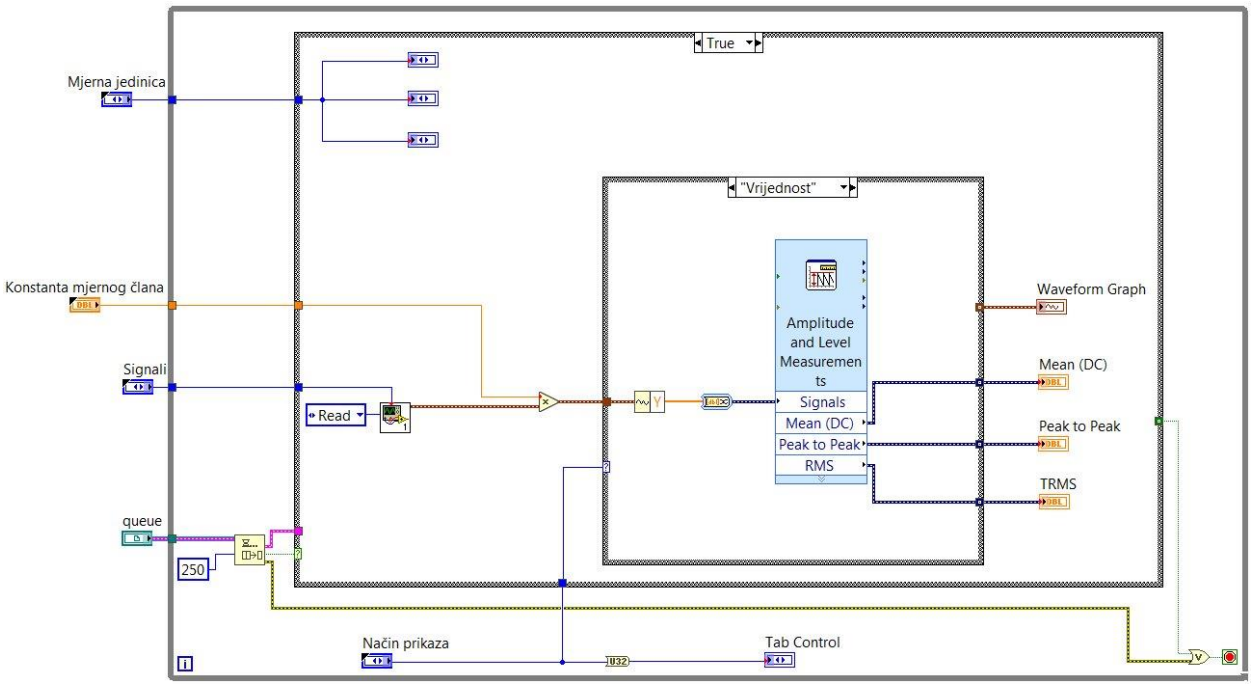
Sl. 3.7: *Blok dijagram virtualnog instrumenta 'Obrada signala'.*

Sl. 3.8: *Prikazi prednjeg panela virtualnog instrumenta 'Obrada signala'.*

Kao i kod prethodnog virtualnog instrumenta, na sve strukture/petlje potrebno je kliknuti desnom tipkom miša i maknuti kvačicu s opcije *Auto Grow*. Gledajući s lijeve prema desnoj strani blok dijagrama, kao što je prikazano na slikama 3.9 i 3.10, potrebno je spojiti blokove identično kao na što je prikazano. Blok *Get Waveform Components* u drugoj unutarnjoj *case* strukturi može se direktno povezati s ulazom opcije *Signals* bloka *Amplitude and Level Measurements*, a blok *Convert to Dynamic Data* se tada automatski stvara između njih. Na mjesta spoja četiri indikatora s desnom stranom unutarnje *case* strukture treba kliknuti desnom tipkom miša i kliknuti na opciju *Use Default If Unwired*.

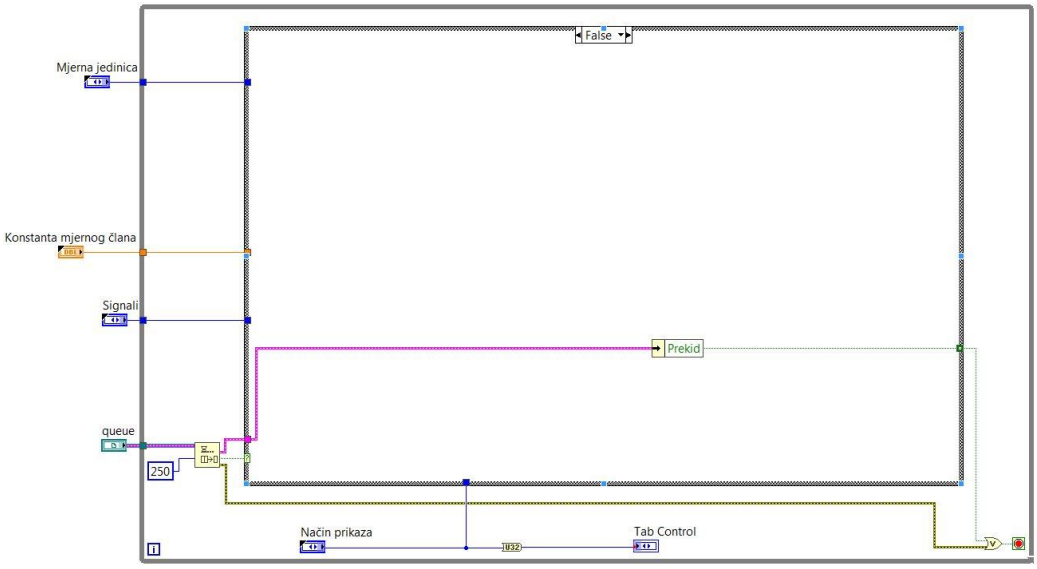


Sl. 3.9: *Blok dijagram virtualnog instrumenta 'Obrada signala'.*



Sl. 3.10*: Blok dijagram virtualnog instrumenta 'Obrada signala'.*

Za slučaj *False* vanjske *case* strukture način spajanja blokova u blok dijagramu prikazan je na slici 3.11. Valja napomenuti da je izgled bloka *Unbundle By Name* prije spajanja nešto drugačiji, ali spajanjem kao na slici on mijenja oblik i unutar njega dolazi oznaka *Prekid*, a s desne strane blok dijagrama, na mjesto dodira žice i *case* strukture, potrebno je kliknuti desnom tipom miša i zatim odabrati opciju *Use Default If Unwired*.

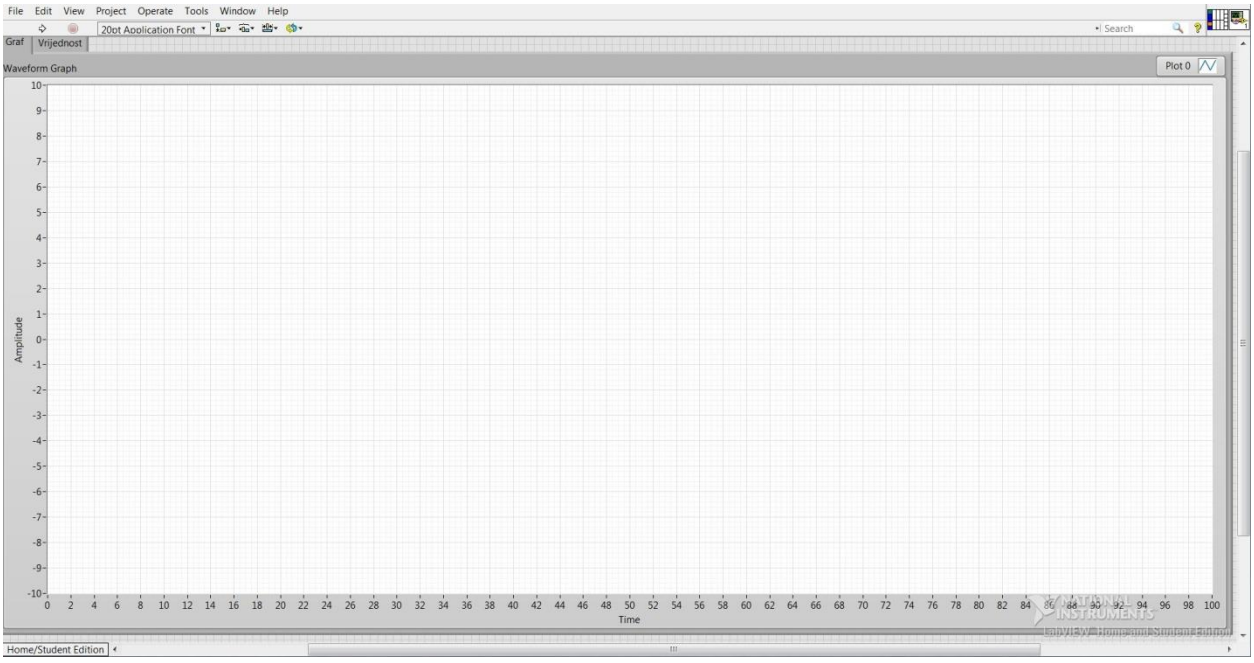


Sl. 3.11*: Blok dijagram virtualnog instrumenta 'Obrada signala'.*

Virtualni instrument Obrada signala veoma je važan u radu sučelja za akviziciju podataka jer će se u glavnom sučelju raditi s odabirom signala, načinom prikaza, odabirom konstante i odabirom mjerne jedinice. Stoga ovaj virtualni instrument treba konfigurirati na određenim mjestima, što će biti opisano u nastavku teksta.

Najprije treba, kao i slučaju virtualnog instrumenta *Spremnik podataka o signalima*, povezati kontrole i *connector pane*. Gornji lijevi kvadratić *connector panea* će se u ovom slučaju dodijeliti kontroli *queue*, a ispod toga će se narednim kvadratićima dodjeljivati kontrole *Signali*, *Način prikaza*, *Konstanta mjernog člana* i *Mjerna jedinica*, kao što je prikazano na slici 3.12. Indikator *Tab control* u ovom slučaju ne treba spajati s *connector paneom*. Nakon toga u blok dijagramu treba kliknuti desnom tipkom miša na sve od četiri navedene kontrole, te kliknuti na opciju *Hide Control* jer će korisnik sučelja pri radu na akviziciji podataka tim kontrolama pristupati izravno iz mape *Kontrole*, o čemu će više riječi biti u posljednjim odlomcima ovog poglavlja.

Poslije ove radnje programer se treba vratiti u prednji panel, te proširiti indikator *Tab Control* što je više moguće, a isto treba učiniti i s indikatorom valnog oblika kako bi svi načini prikaza u glavnom sučelju bili rasprostranjeni na cijeli zaslon računala. Sve ovo također je vidljivo na slici 3.12.

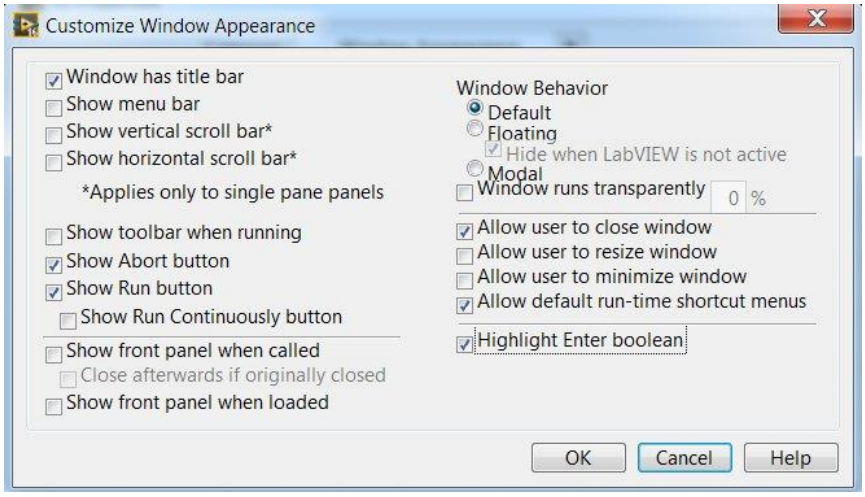


Sl. 3.12: *Prednji panel virtualnog instrumenta 'Obrada signala'.*

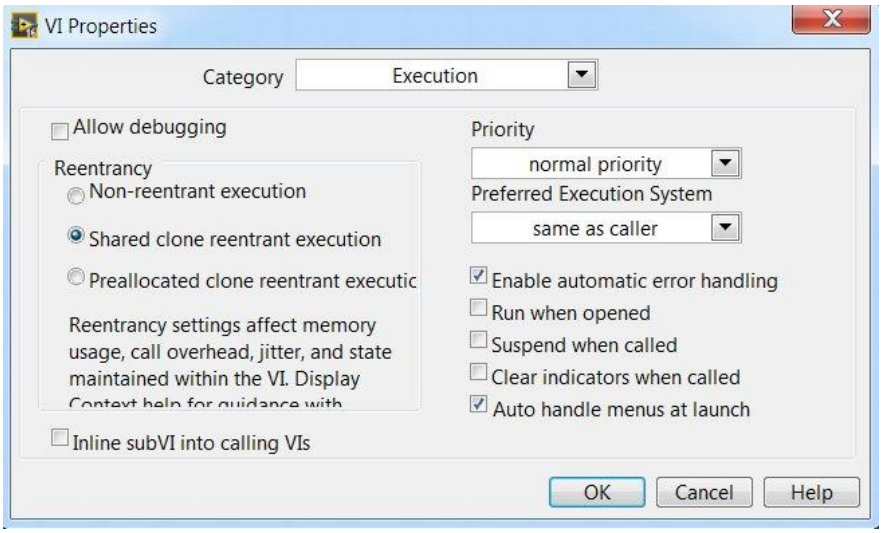
Nakon ovoga slijedi dodatno uređivanje pogleda na prednji panel. Potrebno je kliknuti desnom tipkom miša na indikator *Tab Control*, u blizini tabova *Graf* ili *Vrijednost*. Kod opcije *Visible Items* treba maknuti kvačicu s *Tabs*, a nakon toga potrebno je kliknuti na *Fit Control to Pane*. Desnom tipkom miša treba kliknuti i na sam indikator valnog oblika, a zatim ponoviti radnju klika na opciju *Fit Control to Pane*, a potom odabrati *Autosize Plot Legend*. Za kraj ostaje još jedan klik desnom tipkom miša na potonji indikator, te zatim odabir mogućnosti *Properties*. U tabu *Appearance* može se maknuti kvačica s opcije *Show plot legend*, u tabu *Scales* za X-os maknuti kvačicu s *Autoscale* te za maksimum staviti '0,1', a za Y-os učiniti isto te za minimum ostaviti '-10', a za maksimum '10'. Klikom na OK treba potvrditi izmjene. Posljednji zadatak u ovom virtualnom instrumentu je veoma važan za funkcioniranje glavnog sučelja i zadatak akvizicije podataka, jer se sljedećim izmjenama osigurava da ovisno o broju kanala korisnik doista vidi signale s ulaza kartice na zaslonu računala ako je broj tih ulaznih signala veći od 1.

Potrebno je desnom tipkom miša kliknuti na *connector pane* i odabrati *VI Properties*. U tabu *General* treba staviti kvačicu na *Separate compiled code from source file*, u izborniku *Window Appearance* treba odabrati opciju *Custom*, a potom gumb *Customize...* u kojem je potrebno odabrati opcije koje su prikazane na slici 3.13. Što se tiče izbornika Execution, potrebno je napraviti izmjene koje prikazuje slika 3.14.

Bez promjena koje su vidljive na slici 3.14 sučelje neće funkcionirati, tako da je veoma važno učiniti izmjene koje su njome prikazane.



Sl. 3.13: *Svojstva virtualnog instrumenta (izgled prozora).*



Sl. 3.14: *Svojstva virtualnog instrumenta (izvršavanje programa)*

Mogućnost *Shared clone reentrant execution* osigurava da se u slučaju više signala s ulaza DAQ kartice virtualni instrument *Obrada signala* pojavljuje onoliko puta koliko je na ulazu signala, te da se na osnovu toga na glavnom sučelju prikazuju valni oblik ili vrijednosti za svaki signal pojedinačno, ovisno o tome što korisnik želi. Izmjene treba potvrditi klikom na *OK*, a virtualni instrument treba spremiti u *Sučelje->Pogledi* pod nazivom *Obrada signala*.

Sljedeći na redu su virtualni instrumenti, odnosno jednostavni prikazi za broj signala koji se nalaze na ulazu u DAQ karticu. Način izrade ovih osam virtualnih instrumenata je identičan pa će njihovo stvaranje biti prikazano na primjeru virtualnog instrumenta za slučaj jednog signala na ulazu.

Nakon otvaranja novog virtualnog instrumenta u LabVIEW-u, najprije u *connector paneu* treba dodati još jedan terminal, na blok dijagram je potrebno postaviti dvije *for* i jednu *while* petlju, a nakon toga na blok dijagram je potrebno postaviti nove blokove na sljedeći način:

Programming->Array->Build Array,

Programming->Application Control->Static VI Reference, Property Node, Open VI Reference, Start Asynchronous Call, Invoke Node, Close Reference,

Programming->Synchronization->Queue Operations->Obtain Queue, Dequeue Element, Enqueue Element, Release Queue,

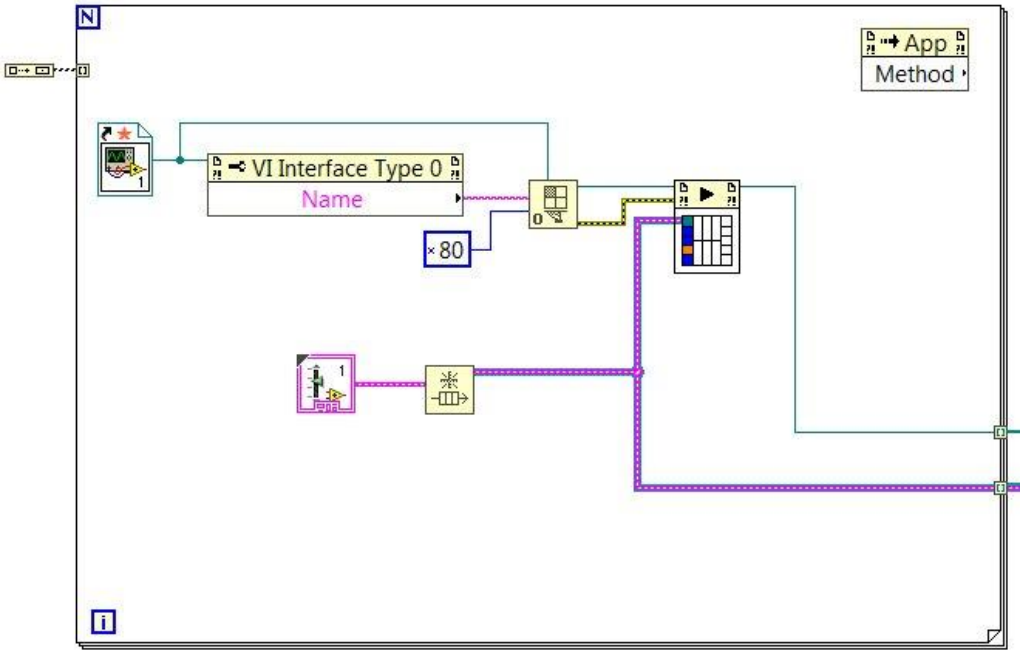
Programming->Cluster, Class, & Variant->Bundle, Unbundle By Name,

Programming->Boolean->True Constant, Or,

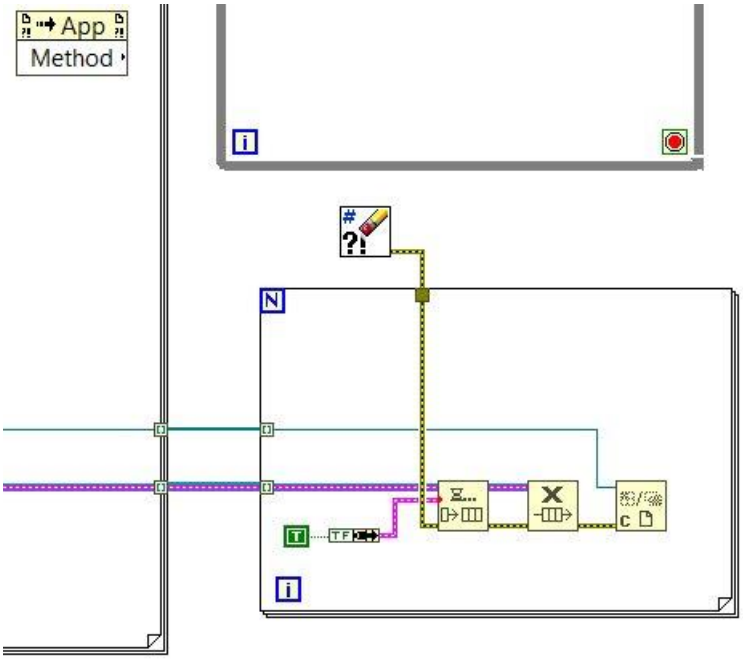
Programming->Dialog & User Interface->Clear Errors.

Desnim klikom na blok *Static VI Reference* otvara se izbornik u kojem treba kliknuti na *Browse for Path...*, a nakon toga treba pronaći i odabrati virtualni instrument *Obrada signala*. Potom je potrebno još jedom kliknuti desnom tipkom miša na isti blok i odabrati *Strictly Typed VI Reference*. Ovaj blok treba spojiti s blokom *Property Node* koji nakon toga mijenja oblik, a u njegovom podizborniku treba kliknuti na *VI Name*. Kao što je prikazano na slici 3.15, ovaj blok treba spojiti s čvorom *vi path* bloka *Open VI Reference*, a čvor *type specifier VI Refnum (for type only)* potonjeg treba spojiti s blokom *Property Node*, kao što je također prikazano slikom 3.15. Iz čvora *options* bloka *Open VI Reference* potrebno je napraviti konstantu i u nju upisati '80', čvor *vi reference* treba spojiti s čvorom *reference* bloka *Start Asynchronous Call* (koji tada mijenja oblik), a čvor *error out* s čvorom *error in* potonjeg bloka.

Potom u paleti funkcija treba kliknuti na *Select a VI...* i odabrati kontrolu *Cluster* iz mape *Kontrole*, a nakon njenog postavljanja na blok dijagram treba kliknuti desnom tipkom miša na rub bloka i staviti kvačicu na *View Cluster As Icon*, te ga spojiti s čvorom *element data type* bloka *Obtain Queue*. Čvor *queue out* potonjeg bloka treba spojiti s ulazom *queue* bloka *Start Asynchronous Call* kako je prikazano slikom 3.15, potonji s čvorom *queue* bloka *Enqueue Element* u drugoj *for* petlji. U potonjoj petlji treba postaviti blok *Bundle* te ga mišem 'smanjiti' kako to prikazuje slika 3.16., a potom ga spojiti s čvorom *element* bloka *Enqueue Element*. Konstantu *True* treba spojiti s blokom *Bundle*, blok *Enqueue Element* s blokom *Release Queue* kako to prikazuje slika 3.16, a ostatak blokova u desnoj *for* petlji s blokom *Clear Errors* kako to prikazuje slika 3.16.



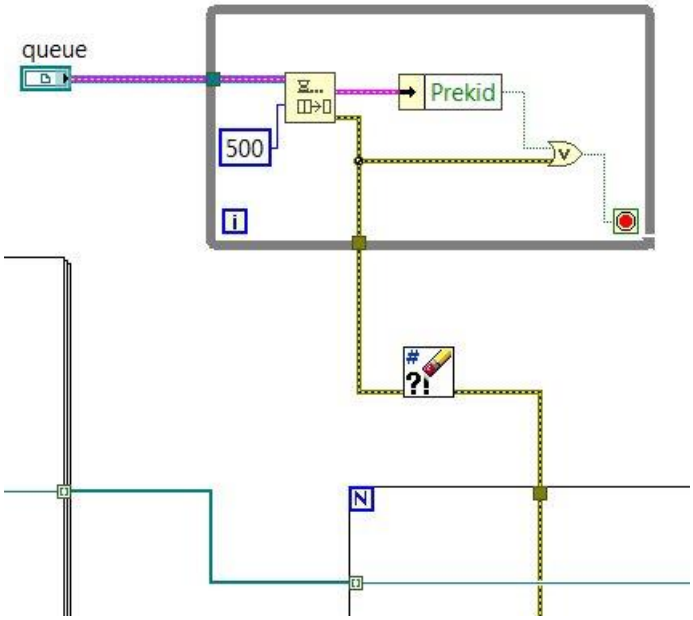
Sl. 3.15: *Blok dijagram virtualnog instrumenta za prikaz jednog signala.*



Sl. 3.16: *Blok dijagram virtualnog instrumenta za prikaz jednog signala.*

Iz čvora *queue* bloka *Dequeue Element* treba napraviti kontrolu, kako je prikazano na slici 3.17 (desni klik na taj čvor, a zatim na *Create->Control*), maknuti kvačicu s opcije *View As Icon* te ju spojiti kako je prikazano slikom. Nakon toga se treba vratiti u prednji panel i desnom tipkom

miša kliknuti na tu kontrolu. Poslije toga je potrebno odabrati opciju *Show Control*, a onda u bijelo polje unutar nje još jednom kliknuti desnom tipkom miša te odabrati *Replace->Select a Control...*, i nakon toga pronaći i odabrati kontrolu *Cluster* iz mape *Kontrole*. Povratkom u blok dijagram uočava se drugačija boja i debljina žice kojom je ta kontrola spojena s blokom *Dequeue Element*, čiji čvor *element* treba spojiti s blokom *Unbundle By Name* (koji tada također mijenja oblik kao i ranije u virtualnom instrumentu *Obrada signala*), a ostatak blokova u *while* petlji treba spojiti kao što je prikazano slikom 3.17, koja pokazuje i način spajanja s blokom *Clear Errors*. Što se tiče bloka *Dequeue Element*, iz čvora *timeout in ms (-1)* treba napraviti konstantu u koju je potrebno upisati '500', što također pokazuje slika ispod.



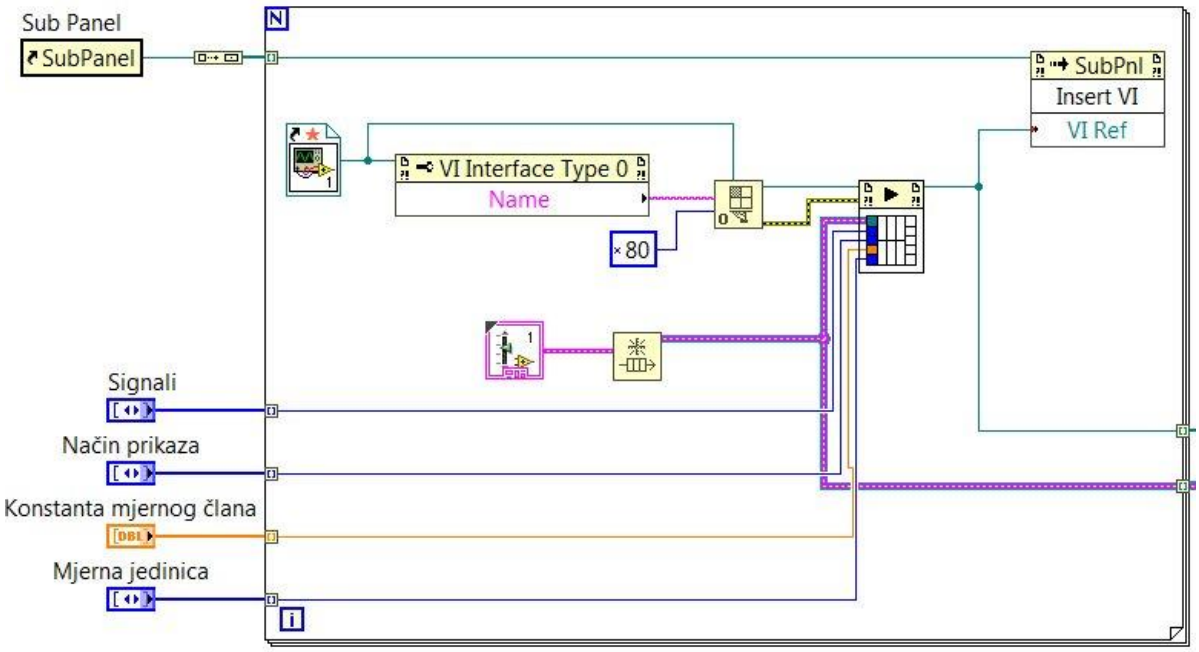
Sl. 3.17: *Spajanje blokova u virtualnom instrumentu za prikaz jednog signala.*

Nakon toga treba desnom tipkom miša kliknuti na svaki od obojenih kvadratića u bloku *Start Asynchronous Call* te iz njih napraviti kontrole, koje također ne treba prikazivati kao ikone zbog preglednosti virtualnog instrumenta. Osim toga, potrebno je na svaku od njih kliknuti desnom tipkom miša i odabrati *Change to Array*, a nakon toga ih spojiti s pripadajućim mjestima (kvadratićima) bloka *Start Asynchronous Call* kao što je prikazano na slici 3.18. Poslije ovih koraka potrebno je se prebaciti u prednji panel gdje se svaka od četiri kontrole treba spojiti s *connector paneom* istim načinom na koji je to učinjeno u prethodna dva virtualna instrumenta. Pošto je virtualni instrument *Obrada podataka* povezan s ovim, treba voditi računa da se spoj kontrola s *connector paneom* u ovom slučaju odradi na isti način kao i kod virtualnog

instrumenta *Obrada podataka*, a što je već pokazano u bloku *Start Asynchronous Call* u blok dijagramu. Dakle, gornji lijevi kvadratić dodjeljuje se kontroli *Cluster*, a ostali (jedan ispod drugog) kontrolama *Signali*, *Način prikaza*, *Konstanta mjernog člana* i *Mjerna jedinica*. Nakon obavljenog pridruživanja, potrebno je se vratiti u blok dijagram i desnom tipkom miša kliknuti na svaku od kontrola (*queue*, *Signali*, *Način prikaza*, *Konstanta mjernog člana*, *Mjerna jedinica*) te odabrati opciju *Hide Control*.

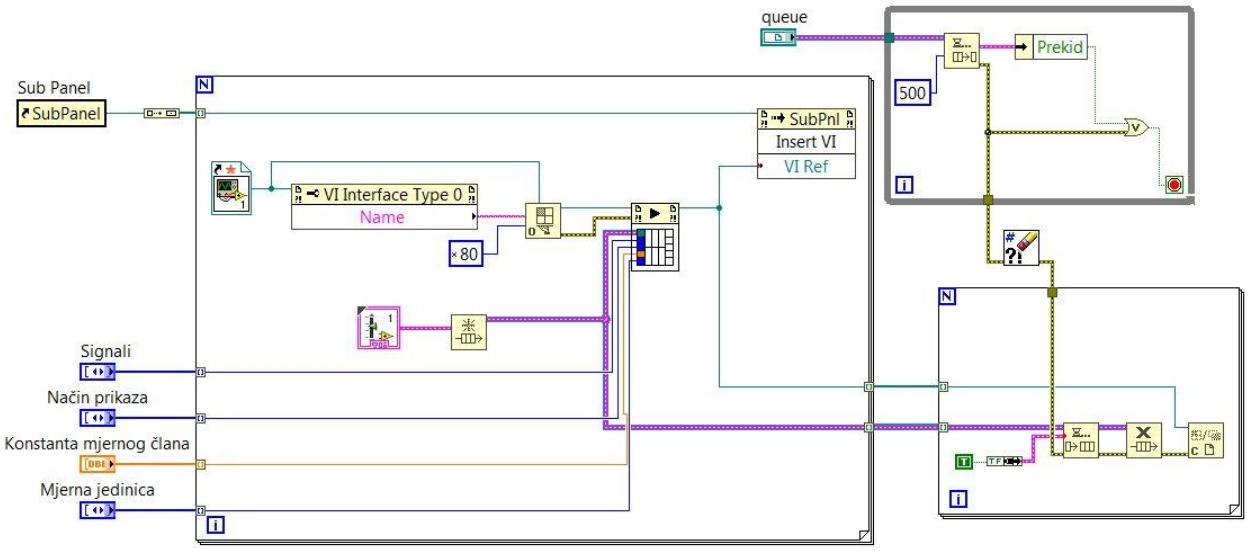
Nakon ovoga slijedi još jedan povratak u prednji panel gdje će se napraviti podloga za pogled na jedan signal s ulaza. Iz palete kontrola je potrebno odabrati *Modern->Containers->SubPanel*, kojem treba maknuti ime uklanjajući opciju *Label* iz izbornika *Visible Items*, a nakon toga je potrebno još jednom kliknuti desnom tipkom miša i odabrati *Fit Control to Pane*.

Programer nakon ovoga treba otići u blok dijagram, desnom tipkom miša kliknuti na blok *Sub Panel*, te odabrati *Replace->Application Control Palette->VI Server Reference*. U tom trenutku se mijenja izgled tog bloka, a u podizborniku treba odabrati *Pane->Sub Panel*. Potonji blok treba spojiti s blokom *Build Array*, a njega s čvorom *reference* bloka *Invoke Node* koji tada mijenja svoj oblik. U podizborniku potonjeg bloka treba odabrati *Insert VI*, te ga spojiti kako prikazuje slika 3.18.



Sl. 3.18: *Blok dijagram virtualnog instrumenta za prikaz jednog signala.*

Ovim radnjama programer je dovršio rad u blok dijagramu i prednjem panelu za potrebe pogleda jednog signala s ulaza DAQ kartice. Blok dijagram virtualnog instrumenta *Pogled 1* prikazan je na slici 3.19. Izgled prednjeg panela identičan je izgledu prednjeg panela praznog virtualnog instrumenta.



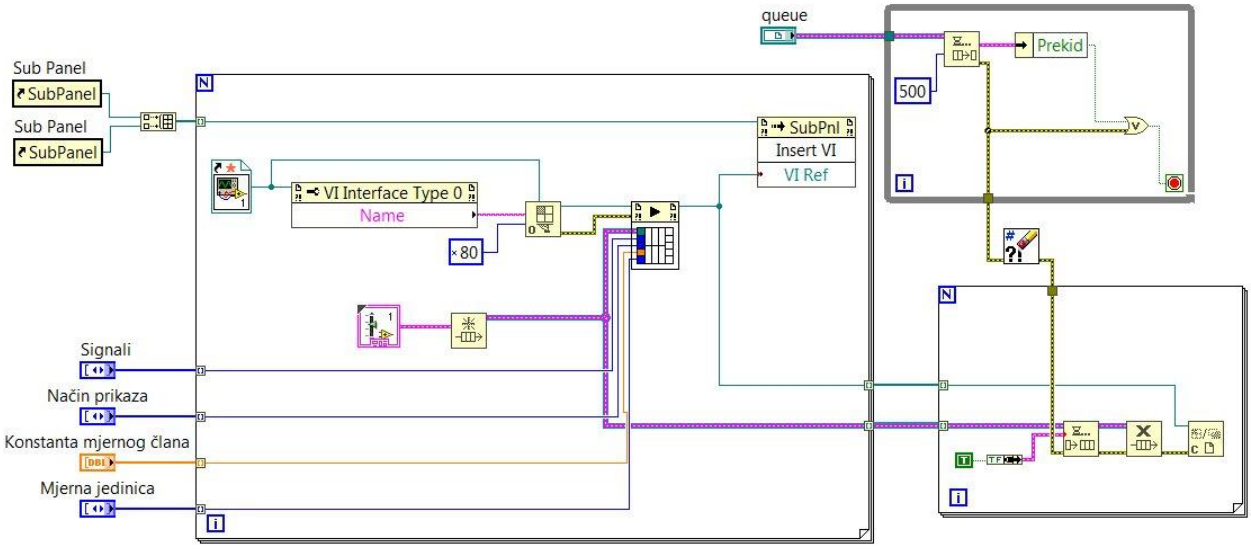
Sl. 3.19: *Blok dijagram virtualnog instrumenta* Pogled 1.

Posljednja stvar koja je potrebno učiniti prije spremanja virtualnog instrumenta je odlazak u postavke *connector panea*, gdje se u izborniku *General* također treba staviti kvačicu na opciju *Separate compiled code from source file*. Virtualni instrument naposljetku treba spremiti pod nazivom 'Pogled 1' u mapu *Pogledi*.

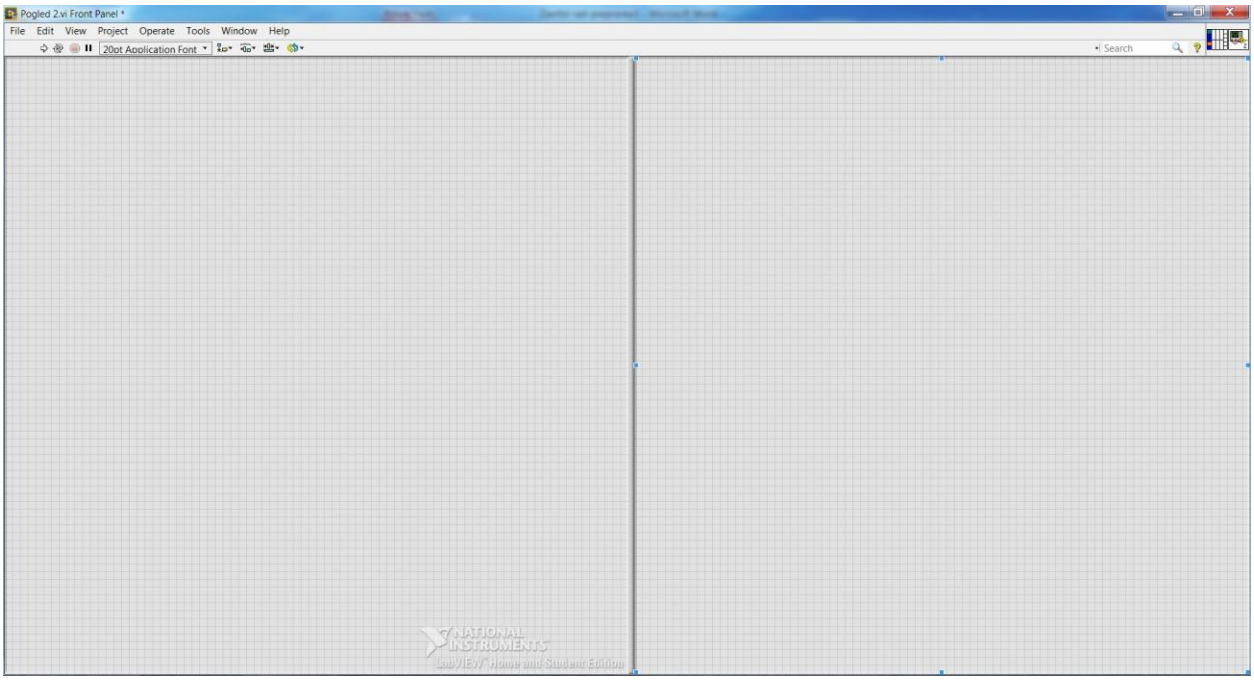
Za izradu ostalih pogleda (*Pogled 2* koji ekran dijeli na dva dijela za prikaz dva signala, *Pogled 3* koji ekran dijeli u četiri kvadrata (u tri od njih prikazuje signale) itd.) proces izrade virtualnog instrumenta ostaje isti do trenutka kada se pred sam kraj u prednji panel treba dodati blok *SubPanel* kao u prvom slučaju. Tada, naime, slijede određene promjene.

U prednjem panelu treba desnom tipkom miša kliknuti na vertikalni i horizontalni klizač te ih isključiti (*Vertical Scrollbar->Always Off, Horizontal Scrollbar->Always Off*). U paleti kontrola potrebno je odabrati *Modern->Containers->Vertical Splitter Bar* (u slučaju dijeljenja ekrana na više dijelova treba odabrati i *Horizontal Splitter Bar* kojim će programer dijeliti ekran po svojoj želji, u ovom slučaju jednoliko). Vertikalni djelitelj ekrana je za slučaj dva signala postavljen okomito na sredini ekrana, a nakon toga je na njega potrebno kliknuti desnom tipkom miša i

odabrati *Splitter Sizing->Splitter Sizes Proportionally*, te odabrati opciju *Locked*. Ovako se osigurava da se okomiti djelitelj ekrana neće pomicati po zaslonu, odnosno da će svaki kvadratić ostati istih dimenzija jedan u odnosu na drugi. Tek tada se u lijevi pa desni kvadrat unutar prednjeg panela postavljaju blokovi *SubPanel*, a daljnji postupak je isti kao i u slučaju jednog signala na ulazu. Nakon pretvorbe *Sub Panela* u blok *VI Server Reference* u blok dijagramu virtualnog instrumenta, treba paziti da prvi blok *VI Server Reference* (*Sub Panel* koji je bio s lijeve strane ekrana) dodijelimo upravo tom *Sub Panelu* iz prednjeg panela virtualnog instrumenta, a u ovom slučaju će se to napraviti tako da se u lijevi blok *VI Server Reference* klikne lijevom tipkom miša i odabere *Splitter->Pane->Sub Panel*, a za drugi *Splitter->Pane 2->Sub Panel*. Važno je povezati prave blokove *VI Server Reference* s pripadajućim podjelama u prednjem panelu, a programer to uvijek može provjeriti klikom desnom tipkom miša na blok *VI Server Reference* (koji nakon dodjele ima naziv 'Sub Panel'), i zatim na *Find->Control*. Prvi *Sub Panel* (blok *VI Server Reference*, odnosno prvi postavljeni blok *Sub Panel*) tada mora pokazivati lijevu stranu ekrana, a drugi desnu. Kod bloka *Build Array* treba mišem kliknuti na donji rub i povući prema dolje kako bi se s lijeve strane tog bloka dobila dva ulaza, a potom slijedi spajanje dva bloka *Sub Panel* s blokom *Build Array*, dok se izlaz iz potonjeg bloka povezuje s blokom *Invoke Node* na isti način kao i kod prethodnog virtualnog instrumenta. Ostatak se spaja i konfigurira isto kao i u prethodnom slučaju za jedan signal na ulazu u DAQ karticu, a virtualni instrument se sprema u mapu *Pogledi* pod imenom 'Pogled 2'. Prikazi blok dijagrama i prednjeg panela virtualnog instrumenta *Pogled 2* vidljivi su na slikama 3.20 i 3.21.



Sl. 3.20: *Blok dijagram virtualnog instrumenta 'Pogled 2'.*

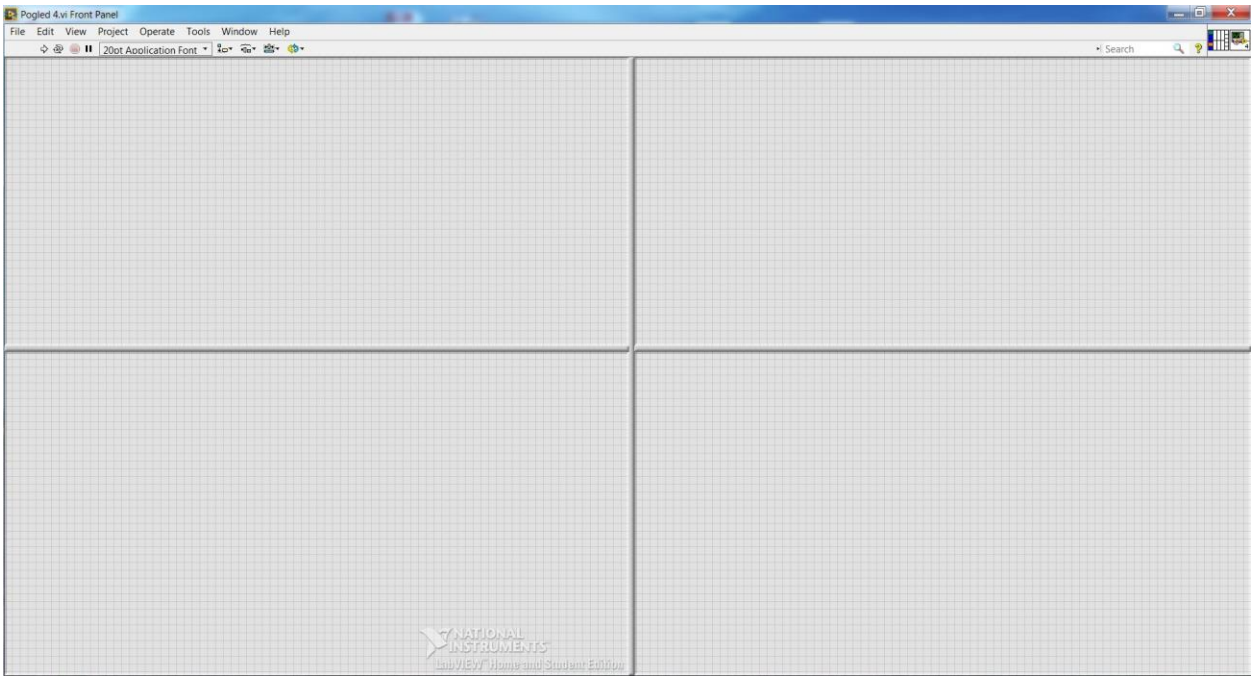


Sl. 3.21: *Prednji panel virtualnog instrumenta 'Pogled 2'.*

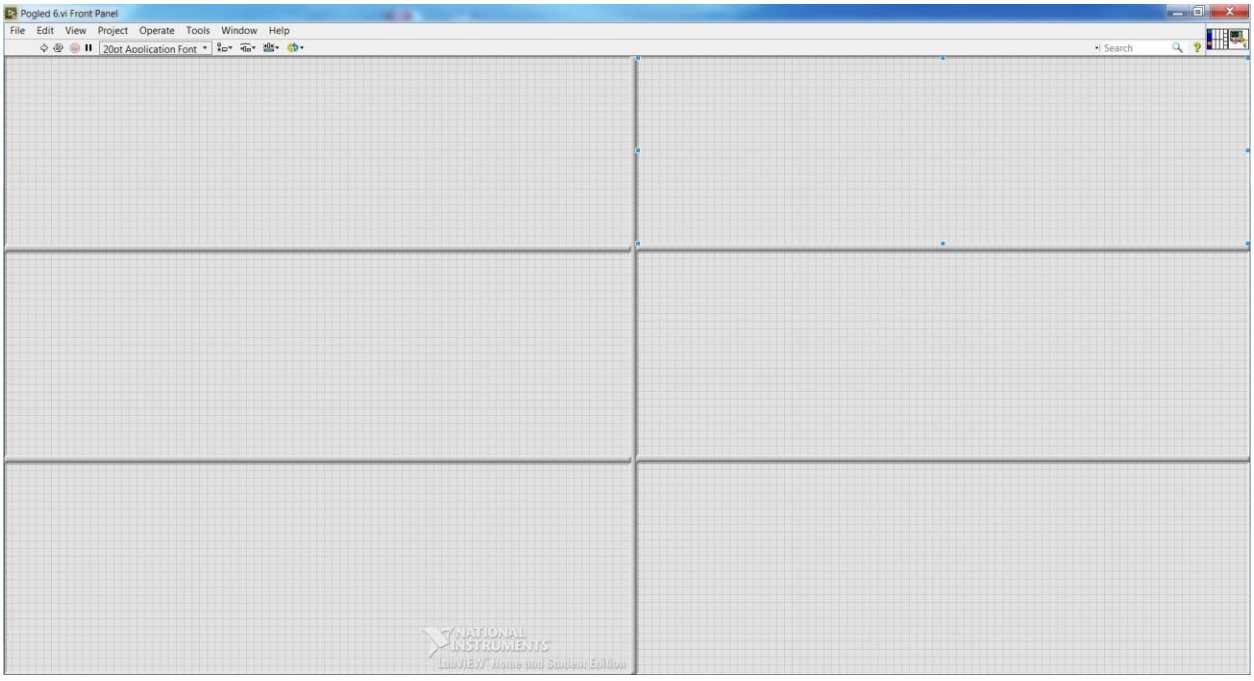
Istom analogijom kreiraju se pogledi za tri, četiri, pet, šest, sedam i osam signala s ulaza DAQ kartice, a u mapu *Pogledi* spremaju se pod nazivima 'Pogled 3', 'Pogled 4', 'Pogled 5', 'Pogled 6', 'Pogled 7' i 'Pogled 8', iako se poput ostalih virtualnih instrumenata i kontrola naziv može dati po želji programera.

Što se tiče ostalih pogleda koji prikazuju neparan broj signala s ulaza, kao što su virtualni instrumenti *Pogled 3*, *Pogled 5* i *Pogled 7*, u njihovim prednjim panelima se nalazi jedan kvadratić/pravokutnik viška, odnosno bez ikakve funkcije. Ovaj način izrade prednjeg panela kod gore navedenih virtualnih instrumenata korišten je radi estetike, s obzirom na to da bi u slučaju tri signala s ulaza u DAQ karticu ekran u drugoj varijanti (gdje se zaslon 'dijeli' na onoliko dijelova koliko ima signala) morao biti podijeljen na tri dijela, a dva pravokutnika u gornjoj polovici ekrana i jedan veliki pravokutnik veće dužine u donjoj polovici ekrana možda ne bi bili najpraktičnije rješenje. Naravno, programer po svojoj želji može mijenjati ovaj koncept pri izradi sučelja te su mu na izbor pružene razne mogućnosti u prednjem panelu LabVIEW-a.

Slike 3.22, 3.23 i 3.24 prikazuju izgled prednjeg panela (podjelu zaslona za pripadajuće signale) za slučaj tri ili četiri signala, pet ili šest signala, odnosno sedam ili osam signala za prikaz na zaslonu računala.



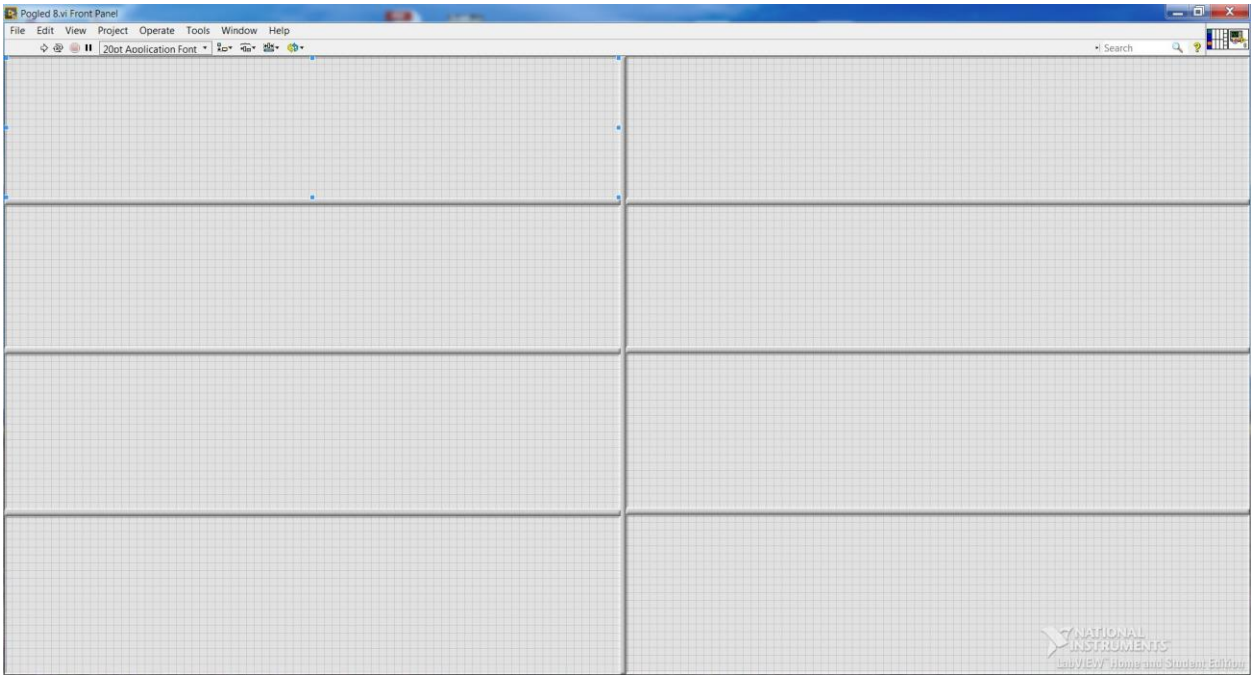
Sl. 3.22: *Prednji panel za akviziciju tri ili četiri signala.*



Sl. 3.23: *Prednji panel za akviziciju pet ili šest signala.*

Kao što je ranije rečeno, u slučaju akvizicije npr. pet signala, oni će biti raspoređeni svaki u svoj dio zaslona (prvi signal gore lijevo, drugi signal gore desno, treći signal u sredini lijevo, itd.),

dok će šesti pravokutnik ostati prazan. Ovakvo rješenje je praktičnije jer bi promatranje signala u jednom dugačkom a uskom pravokutniku bilo dosta otežano i nepraktično.



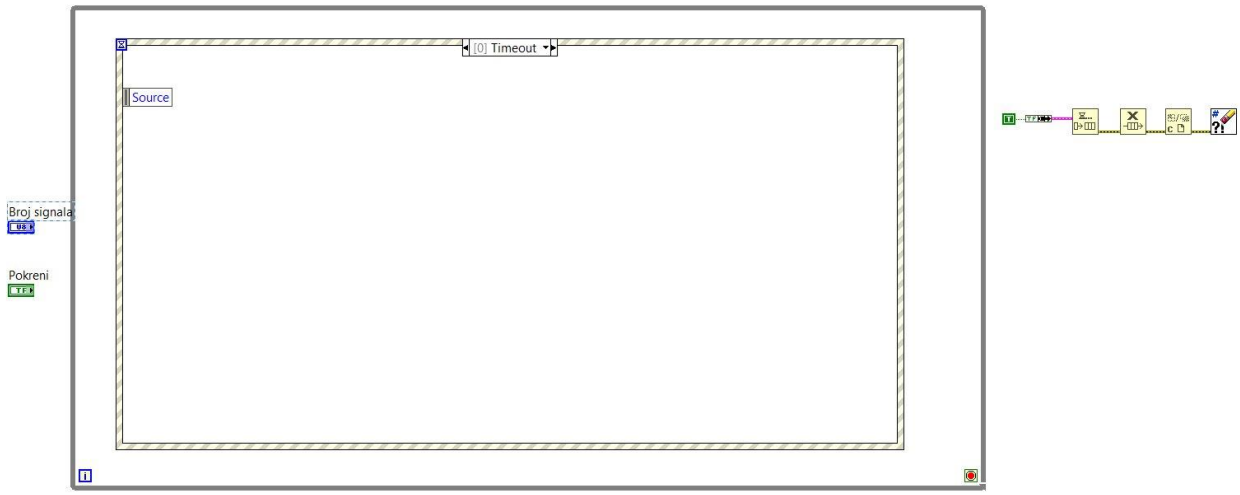
Sl. 3.24: *Prednji panel za akviziciju sedam ili osam signala.*

Nakon izrade virtualnih instrumenata za prikaz onoliko signala koliko to korisnik sučelja želi, na red dolazi i izrada samog sučelja za akviziciju podataka, odnosno glavne aplikacije s kojom će svaki korisnik raditi pri procesu akvizicije podataka.

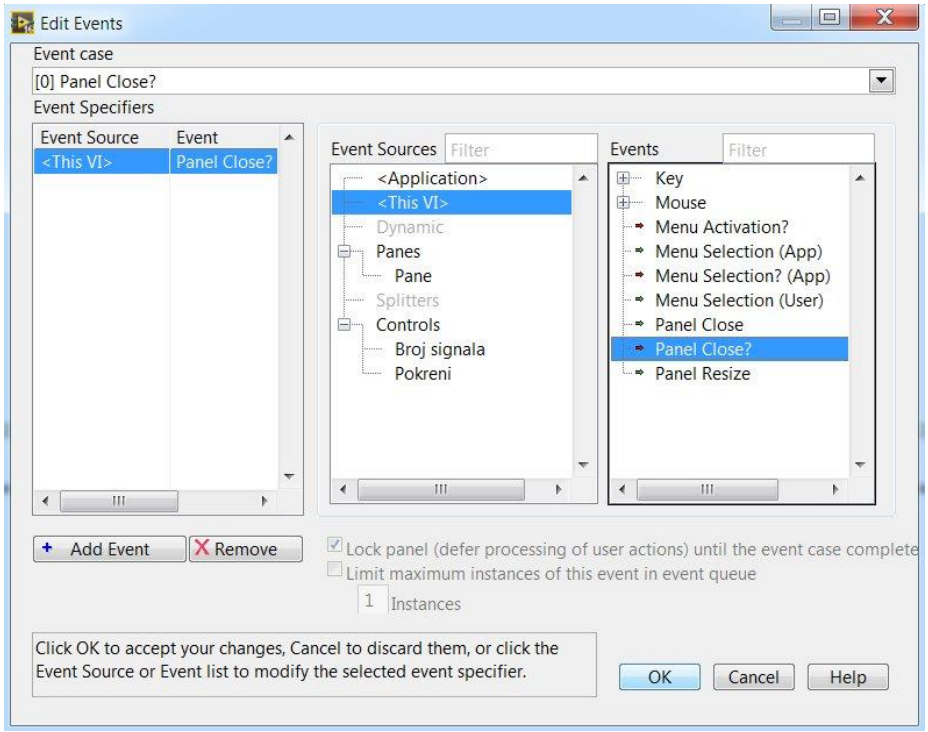
Osim dodavanja terminala u *connector paneu*, u blok dijagram je potrebno postaviti *while* petlju, a unutar nje jednu *event* strukturu (u oba slučaja ukloniti opciju *Auto Grow*). Unutar potonje s lijeve strane postoje dva menija: *source* i *type*, a desnom tipkom miša treba kliknuti na *type* te odabrati opciju *Remove Element*. Ovaj postupak treba ponoviti za svaki od tri slučaja koji će postojati u ovoj strukturi događaja. Slično kao i u slučaju virtualnih instrumenata za poglede, na blok dijagram treba postaviti sljedeće blokove: *Enqueue Element, Release Queue, Close Reference, Clear Errors, Bundle* i *True Constant*, te ih spojiti slično kao i u slučaju pogleda, odnosno kako prikazuje slika 3.25.

Nakon toga programer treba otići u prednji panel te iz palete kontrola odabrati *System->Numeric->System Spin Control*, a zatim i *Silver->Boolean->Buttons->Play Button (Silver)* te s

potonje kontrole maknuti ime u izborniku *Visible Items*. U blok dijagramu s ovih kontrola treba ukloniti kvačicu s opcije *View As Icon*, te *Boolean* kontrolu preimenovati u 'Pokreni', a numeričku kontrolu u 'Broj signala' te joj u izborniku *Representation* dodijeliti opciju *U8*. Poslije toga je potrebno kliknuti desnom tipkom miša na strukturu događaja te odabrati opciju *Edit Events Handled by This Case...*, i zatim odabrati opcije koje prikazuje slika 3.26.



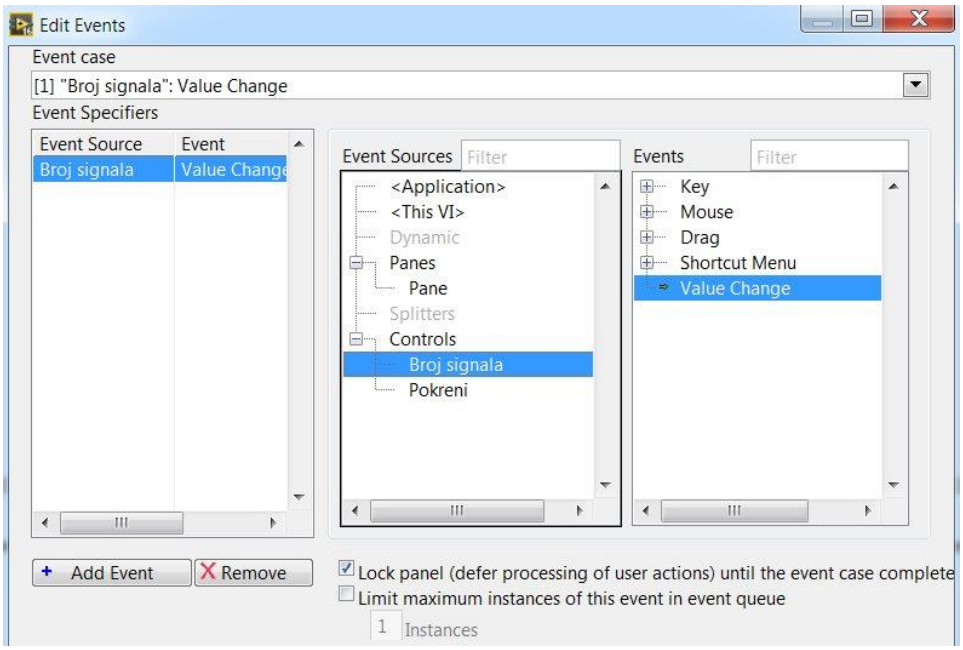
Sl. 3.25: While *petlja i struktura događaja u blok dijagramu sučelja.*



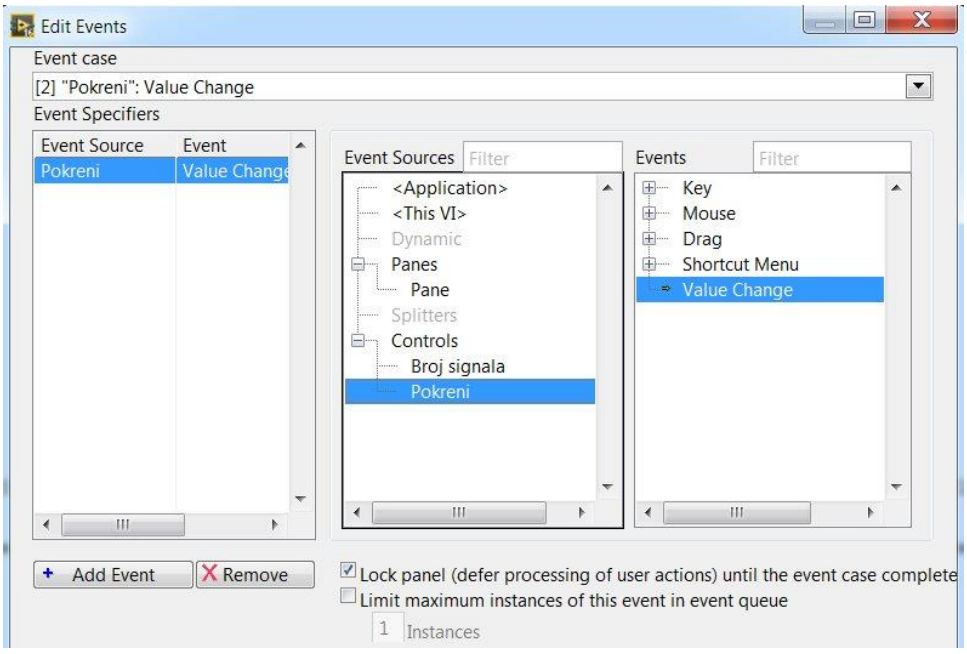
Sl. 3.26: *Postavke prvog događaja u strukturi događaja.*

Nakon toga slijedi klik desnom tipkom miša na strukturu i odabir opcije *Add Event Case...*.

Ovaj postupak treba ponoviti još jednom. Prvi puta treba odabrati postavke sa slike 3.27, a drugi puta postavke koje prikazuje slika 3.28.

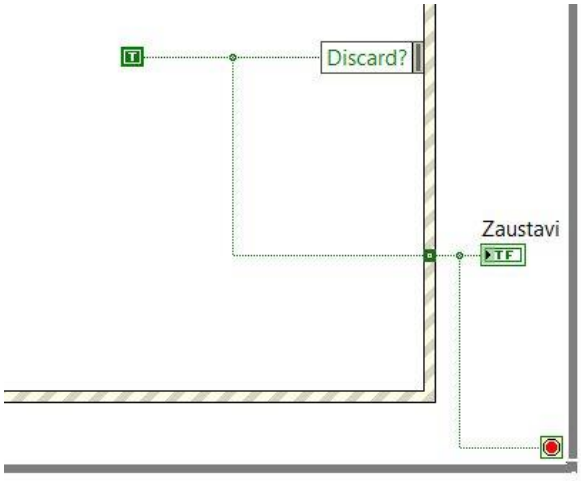


Sl. 3.27: *Drugi događaj u strukturi događaja.*



Sl. 3.28: *Treći događaj u strukturi događaja.*

Za prvi događaj unutar strukture događaja treba postaviti blok *True Constant*, a između strukture i petlje još jedan takav blok koji potom treba postaviti kao indikator, te maknuti kvačicu s opcije *View As Icon*, odabrati opciju *Hide Indicator*, a ime mu se može promijeniti u 'Zaustavi'. Blokove treba spojiti kako to prikazuje slika 3.29, a rad u blok dijagramu za ovaj događaj trenutno je završen.

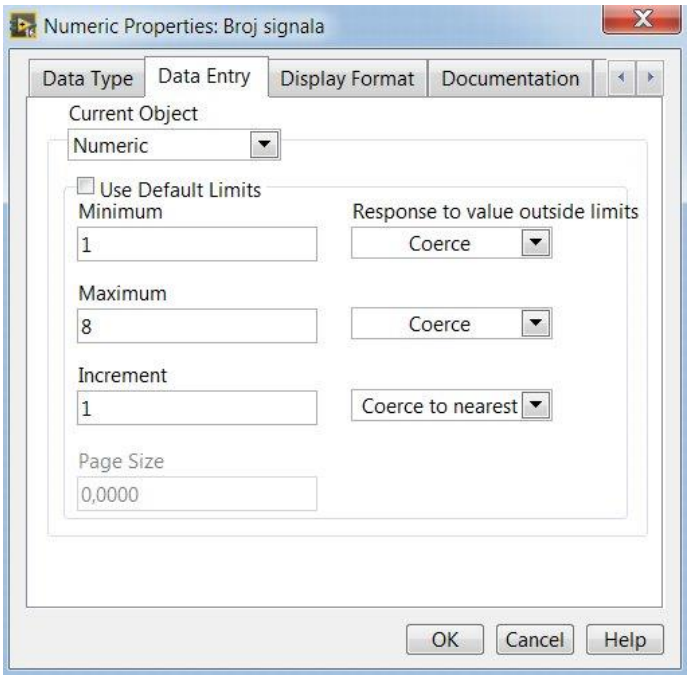


Sl. 3.29: *Dio blok dijagrama prvog slučaja.*

Potom se nakratko treba vratiti u prednji panel kako bi se uklonili vertikalni i horizontalni klizači. Zatim je potrebno postaviti jedan vertikalni *splitter* (uz odabir opcija *Locked* te *Splitter Sizes Proportionally*), otprilike tako da zaslon dijeli u omjeru 1:4. Iz palete kontrola na desni dio zaslona treba postaviti blok *SubPanel* te mu ukloniti ime i odabrati opciju *Fit Control to Pane*. Ovim se osigurava da se s lijeve strane sučelja nalaze kontrole za selekciju raznih mogućnosti, dok se s desne strane pojavljuju odgovarajući prikazi, odnosno pogledi.

Potom se treba vratiti u blok dijagram sučelja i odabrati treći događaj iz strukture događaja naziva '"Pokreni": Value Change'. Unutar strukture je potrebno postaviti elemente koji su već na početku izrade ovog virtualnog instrumenta postavljeni s desne strane blok dijagrama: *Enqueue Element*, *Release Queue*, *Close Reference*, *Clear Errors*, *Bundle* i *True Constant*, a spojiti ih treba kao i u prethodnom slučaju. Blok *Pokreni* je potrebno postaviti unutar strukture, kao i nedavno dobiveni blok *Sub Panel*. Nakon toga je potrebno dodati blokove koji su već korišteni u virtualnim instrumentima pogleda, a to su *Obtain Queue*, *Start Asynchronous Call*, *Open VI Reference*, *Property Node*, zatim četiri puta dodati blok *Array Subset* te jednu *case* strukturu. Što

se tiče kontrole *Broj signala* koja se nalazi izvan *while* petlje, nju je potrebno ubaciti u strukturu drugog događaja *"Broj signala": Value Change*, kliknuti desnom tipkom miša na nju te odabrati *Create->Property Node->Value (Signalling)* i postaviti novi blok s lijeve strane izvan *while* petlje. Osim toga, u tabu *Edit Items* kontrole *Broj signala* postavke treba konfigurirati onako kako je prikazano slikom 3.30, budući da će sučelje moći prikazati maksimalno osam signala s DAQ kartice.

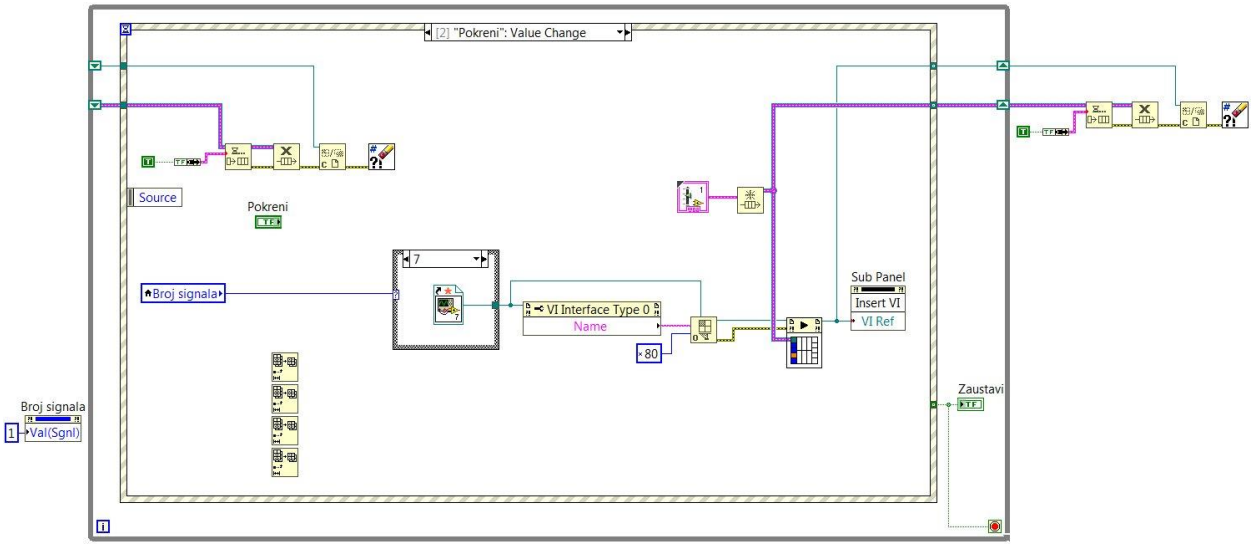


Sl. 3.30: *Postavke kontrole za odabir broja signala.*

S lijeve strane bloka *Broj signala (Property Node)* treba kreirati konstantu te u nju upisati '1'. Zatim je se potrebno vratiti u drugi događaj u strukturi događaja, te nakon klika desnom tipkom miša na kontrolu *Broj signala* odabrati *Create->Local Variable*, a novi blok treba postaviti u treći događaj u strukturi događaja koji se naziva '"Pokreni": Value Change'. Poslije toga treba kliknuti na potonji blok desnom tipkom miša i odabrati opciju *Change to Read*. U blok dijagramu u paleti funkcija treba odabrati opciju *Select a VI...* te pronaći i odabrati kontrolu *Cluster* iz mape *Kontrole* (ovaj puta prikazati kao ikonu). Lokalnu varijablu *Broj signala* treba spojiti s *case* strukturom kao što je prikazano slikom 3.31, a na samu *case* strukturu treba kliknuti desnom tipkom miša i maknuti kvačicu s opcije *Auto Grow*, i zatim odabrati *Add Case After* sve dok se ne dobije osam slučajeva (za osam signala s ulaza u DAQ kartice). Prvi slučaj (imena '0') može se obrisati odabirom opcije *Delete This Case*. Za svaki od osam slučajeva

unutar *case* strukture treba dodati blok *Static VI Reference*, zatim odabrati putanju (za slučaj *1* to će biti virtualni instrument *Pogled 1*, za slučaj *2* to će biti virtualni instrument *Pogled 2*, itd.) te na svaki od tih blokova kliknuti desnom tipkom miša i odabrati opciju *Strictly Typed VI Reference*.

Nakon toga je potrebno spojiti *case* strukturu (svaki blok *Static VI Reference* unutar potonje mora biti spojen s desnim rubom strukture) s blokovima koji su prikazani na slici 3.31., gdje se može vidjeti i način spajanja tih elemenata. Slučaj *1* treba postaviti kao zadani odabirom opcije *Make This The Default Case*. Kao što je već viđeno u virtualnim instrumentima pogleda, neki od ovih blokova su već korišteni ranije u izradi cjelokupnog sučelja za akviziciju podataka, kao i to da se prilikom spajanja nekih od navedenih elemenata mijenja njihov izgled. U podizbornicima ovih blokova se tada koriste iste funkcije te se spajaju na isti način kao i kod virtualnih instrumenata pogleda, a sve je detaljno prikazano slikom 3.31.



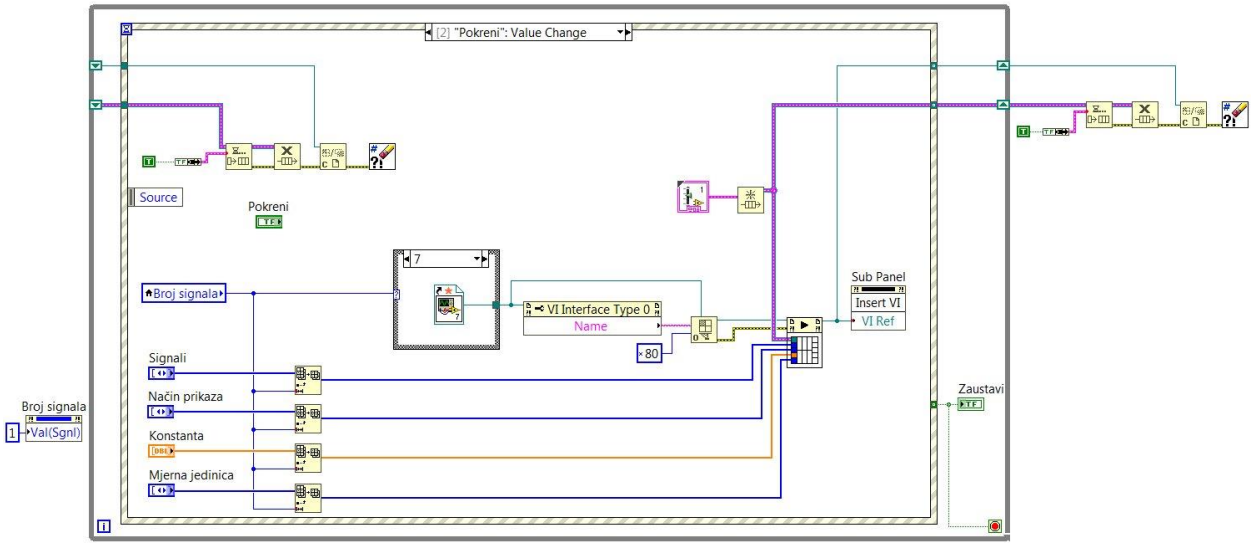
Sl. 3.31: *Blok dijagram sučelja za akviziciju podataka.*

Prilikom izrade blok dijagrama ovog virtualnog instrumenta važno je iznova postavljati nove blokove na blok dijagram, a ne kopirati postojeće iako se isti blokovi više puta nalaze na ovom blok dijagramu, kao što je vidljivo na slici 3.31.

Na mjesta spojeva žica s *while* petljom (desna strana) potrebno je kliknuti desnom tipkom miša i odabrati opciju *Replace with Shift Register*, a potom je na lijevoj strani na istom mjestu potrebno kliknuti lijevom tipkom miša i samo ispustiti strelicu koja se tada nalazi uz pokazivač miša. U

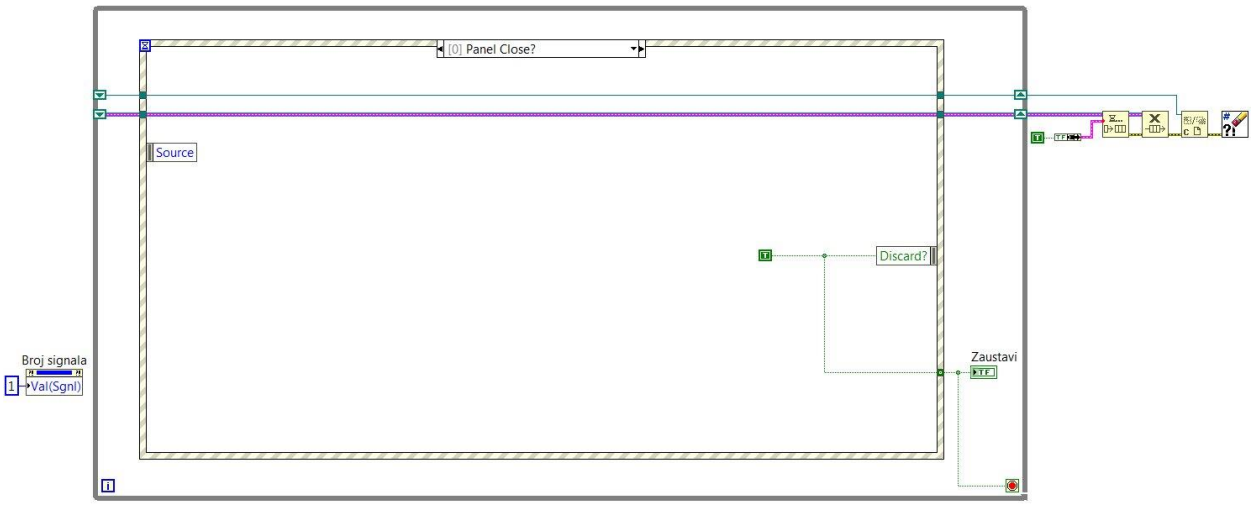
ovom slučaju strelice su orijentirane na suprotnu stranu. Izgled ovih *shift* registara vidljiv je na slici 3.31.

Iz plavih kvadratića bloka *Start Asynchronous Call* treba napraviti pripadajuće kontrole (po potrebi urediti imena), maknuti kvačicu s opcije *View As Icon* te ih spojiti kako je prikazano na slici 3.32. Lokalnu varijablu *Broj signala* također treba spojiti kako je prikazano slikom 3.32, i ovim postupcima je završen rad za događaj *"Pokreni": Value Change* u ovom virtualnom instrumentu.



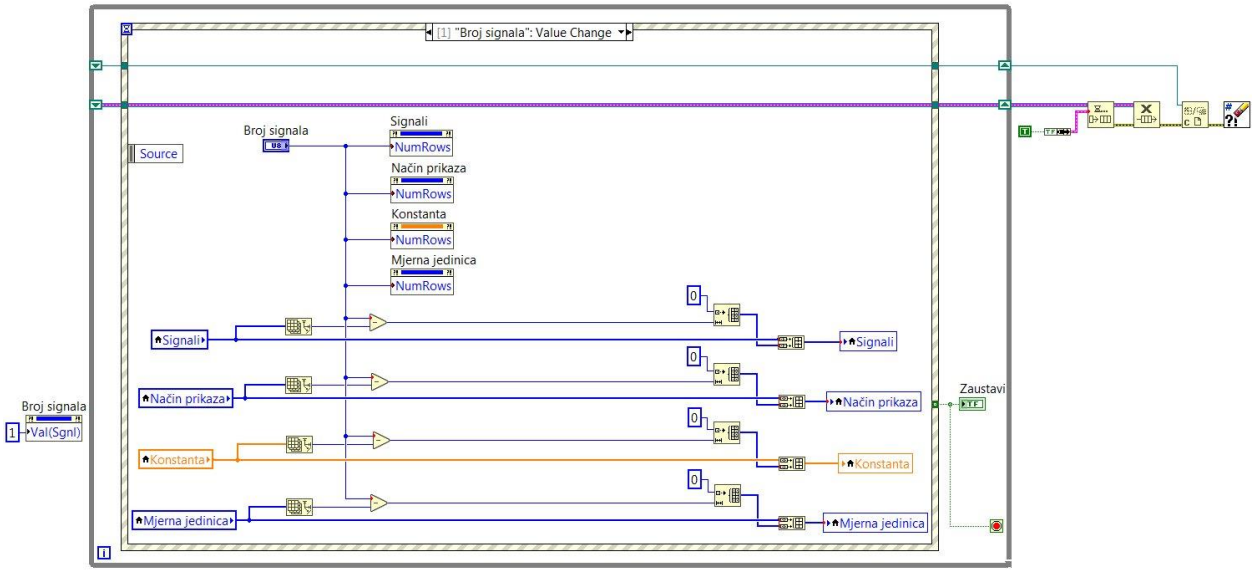
Sl. 3.32: *Blok dijagram za treći događaj u strukturi događaja.*

Blok dijagram za prvi događaj treba povezati kako je prikazano na slici 3.33, i to je sav posao koji programer treba napraviti za događaj *Panel Close?*.



Sl. 3.33: *Blok dijagram za prvi događaj u strukturi događaja.*

Programer se sada nakratko treba prebaciti u treći događaj u strukturi događaja. Tamo je potrebno kliknuti desnom tipkom miša na četiri kontrole (*Signali*, *Način prikaza*, *Konstanta* i *Mjerna jedinica*) te odabrati opciju *Create->Local Variable* (po dva puta, za prvi slučaj odabrati opciju *Change To Read*), te *Create->Property Node->Number of Rows* i odabrati opciju *Change All To Write*. Ove nove blokove potrebno je zatim postaviti unutar drugog događaja u strukturi događaja. Pridružiti im treba blokove *Array Size*, *Subtract*, *Initialize Array* i *Build Array* (za potonji odabrati opciju *Concatenate Inputs*). Iz čvora *element* blokova *Initialize Array* treba napraviti konstantu u kojoj treba biti upisano '0', a nakon toga na nju treba kliknuti desnom tipom miša i odabrati *Representation->U8*. Potom dodane blokove treba spojiti kako je prikazano slikom 3.34.



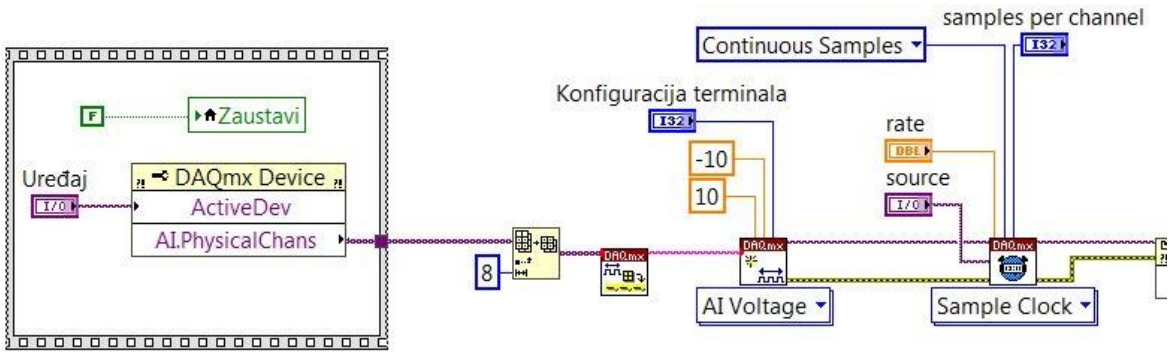
Sl. 3.34: *Blok dijagram drugog događaja u strukturi događaja.*

Ispod ove strukture naposljetku je potrebno napraviti malenu strukturu kôda koja će poslužiti za prikupljanje podataka s DAQ kartice. Iz palete funkcija je potrebno odabrati blok, odnosno strukturu *Flat Sequence* (*Programming->Structures->Flat Sequence*) te ju postaviti ispod *while* petlje, kako je prikazano slikom 3.35. Unutar potonje je potrebno dodati blok *Property Node*, nakon čega je na potonji potrebno kliknuti desnom tipkom miša i odabrati *Select Class->DAQmx->DAQmx Device*, a u podizborniku bloka treba odabrati *Active Device*. Nakon još jednog klika desnom tipkom miša, potrebno je odabrati *Add Element*, a u donjem podizborniku odabrati *I/O Type->Analog Input->Physical Channels*. Iz podizbornika *ActiveDev* treba napraviti kontrolu i nazvati je 'Uređaj' (ne prikazivati kao ikonu), a u *Flat Sequence* strukturu treba dodati i blok *False Constant*. Zatim treba kliknuti desnom tipkom miša na blok *Zaustavi* unutar *while* petlje i dva puta odabrati *Create->Local Variable*. Jednu lokalnu varijablu treba ubaciti u *Flat Sequence* strukturu i spojiti je s blokom *False Constant*.

Nakon ovih radnji je na blok dijagram potrebno postaviti blok *Array Subset* te njegov *array* čvor povezati kao na slici 3.35, dok iz čvora *length* treba napraviti konstantu i u nju upisati broj '8'. Nakon toga treba otići u paletu za DAQ zadatke (*Measurement I/O->NI DAQmx*) i odabrati *Advanced->DAQmx Utilities->DAQmx Flatten String*, i ulaz potonjeg bloka *names in* povezati s prethodnim blokom kao na slici 3.35. Iz palete s DAQmx blokovima na blok dijagram treba dodati sljedeće blokove: *Create Channel, Timing, Start, Read, Stop i Clear*. Osim ovoga, potrebno je dodati i blok *Simple Error Handler* iz palete *Programming->Dialog & User*

Interface, zatim jednu *while* petlju, blok *Or*, *Equal To 0?* iz palete *Programming->Comparison*, nakon toga pomoću opcije *Select a VI...* pronaći i odabrati virtualni instrument *Spremnik podataka o signalima*, te dodati još jedan blok *Property Node*.

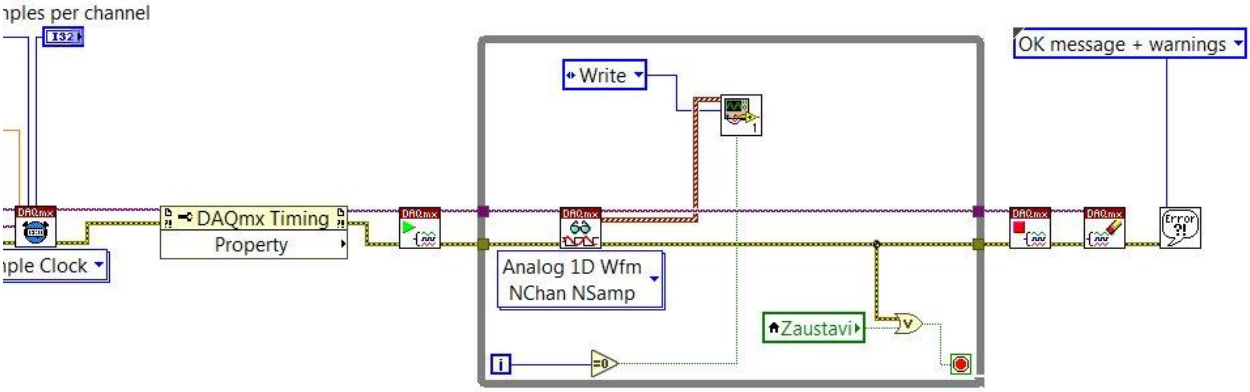
Izlaz iz bloka *Flatten String* povezuje se s čvorom *physical channels* bloka *Create Channel*, a kod potonjeg bloka iz čvorova *minimum value* i *maximum value* treba napraviti konstante, te u njih upisati '-10', odnosno '10'. Iz čvora *input terminal configuration* treba napraviti kontrolu te ju nazvati 'Konfiguracija terminala' (ne prikazivati kao ikonu), a blok *Create Channel* je nakon toga potrebno povezati s blokom *Timing* kao na slici 3.35. Iz čvorova *source* i *rate* potonjeg bloka također treba napraviti kontrole (ne prikazivati ih kao ikone), iz čvora *sample mode* napraviti konstantu i u izborniku odabrati *Continuous Samples*, a iz čvora *samples per channel* napraviti kontrolu (ne prikazivati ju kao ikonu).



Sl. 3.35: *Dio blok dijagrama za akviziciju podataka DAQ karticom.*

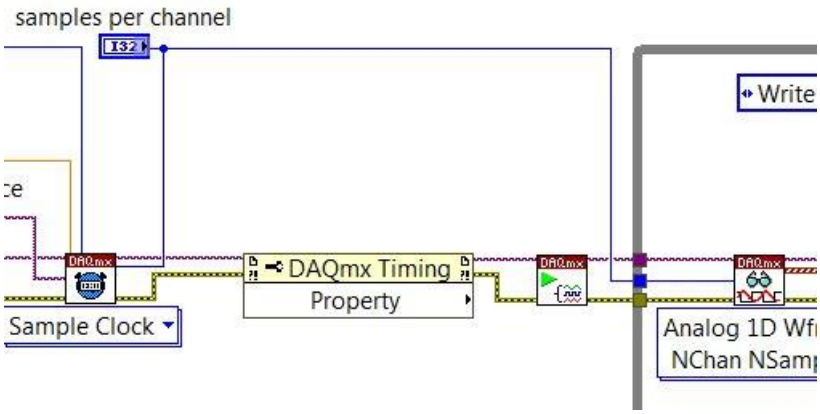
Na blok *Property Node* treba kliknuti desnom tipkom miša i odabrati *Select Class->DAQmx->DAQmx Timing*, te potonji blok spojiti sa susjednim blokovima kao na slici 3.36.

Nakon toga treba povezati blokove *Start* i *Read* (koji se nalazi unutar petlje), a u podizborniku potonjeg bloka treba odabrati *Analog->Multiple Channels->Multiple Samples->1D Waveform*. Čvor *data* bloka *Read* treba spojiti s čvorom *Podaci* bloka (virtualnog instrumenta) *Spremnik podataka o signalima*, kao što to prikazuje slika 3.36., te iz njegovog čvora *Metoda* napraviti konstantu i odabrati mogućnost *Write*. Izlaz iz bloka *Equal To 0?* spaja se na čvor *Reset* bloka *Spremnik podataka o signalima*. Drugoj lokalnoj varijabli na blok dijagramu treba svojstvo promijeniti klikom na opciju *Change To Read*, a potom preostale blokove spojiti prema slici 3.36. Iz čvora *type of dialog (OK msg:1)* potrebno je napraviti konstantu te iz izbornika odabrati opciju *OK message + warnings*.



Sl. 3.36: *Dio blok dijagrama za akviziciju podataka DAQ karticom.*

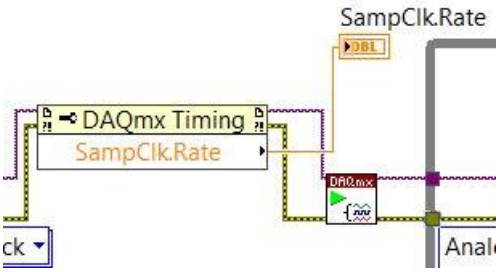
Nakon toga preostaje još jedan detalj, a to je spajanje kontrole *samples per channel* s čvorom *number of samples per channel* bloka *Read*, što prikazuje slika 3.37.



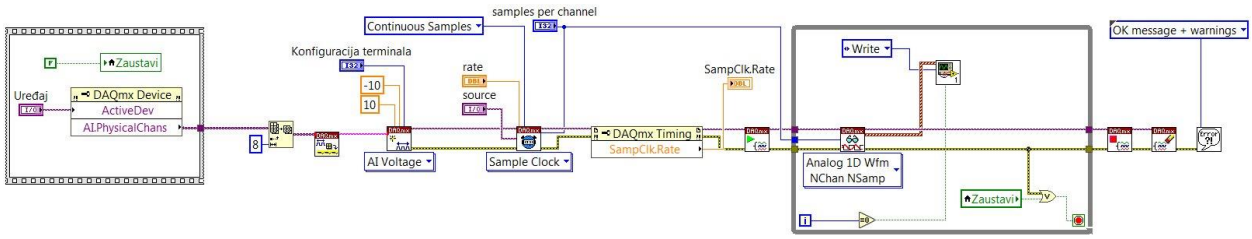
Sl. 3.37: *Bio blok dijagrama za akviziciju podataka DAQ karticom.*

S obzirom na to da DAQ kartica nije bila spojena s računalom u vrijeme pisanja ovog završnog rada, prilikom spajanja iste je potrebno u bloku *Property Node* (*DAQmx Timing* na slici 3.37) promijeniti svojstvo u podizborniku odabirom opcije *SampClk.Rate*, a potom iz njegovog izlaza napraviti indikator koji će se automatski zvati istim imenom (ne prikazivati ga kao ikonu). Prikaz ove mogućnosti vidljiv je na slici 3.38., a prikaz kompletnog dijela kôda za akviziciju signala DAQ karticom prikazan je na slici 3.39. Budući da dio kôda sa slike 3.37 izbacuje pogrešku, za potrebe pisanja ovog završnog rada i ispravnog rada sučelja ubačen je blok *Property Node* s

verzije sučelja koja je u laboratoriju korištena za akviziciju podataka i testiranje dok je s računalom bila spojena DAQ kartica.



Sl. 3.38: *Svojstva bloka DAQmx Timing.*

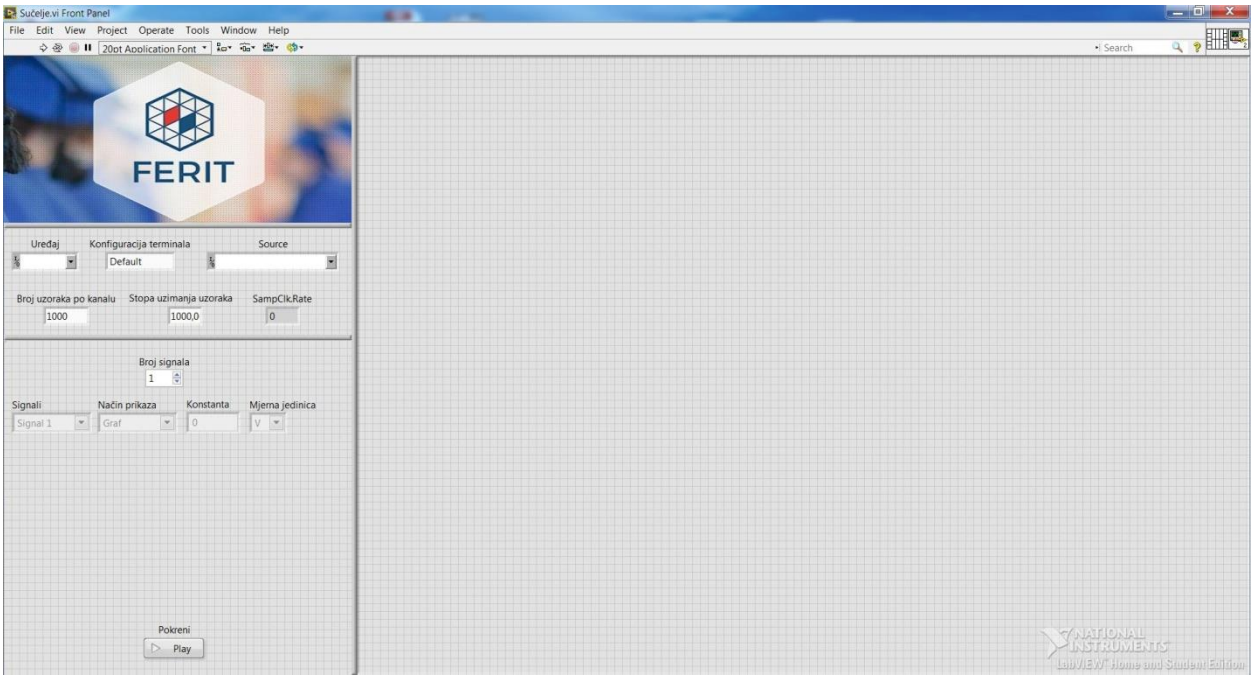


Sl. 3.39: *Dio blok dijagrama zadužen za prikupljanje signala DAQ karticom.*

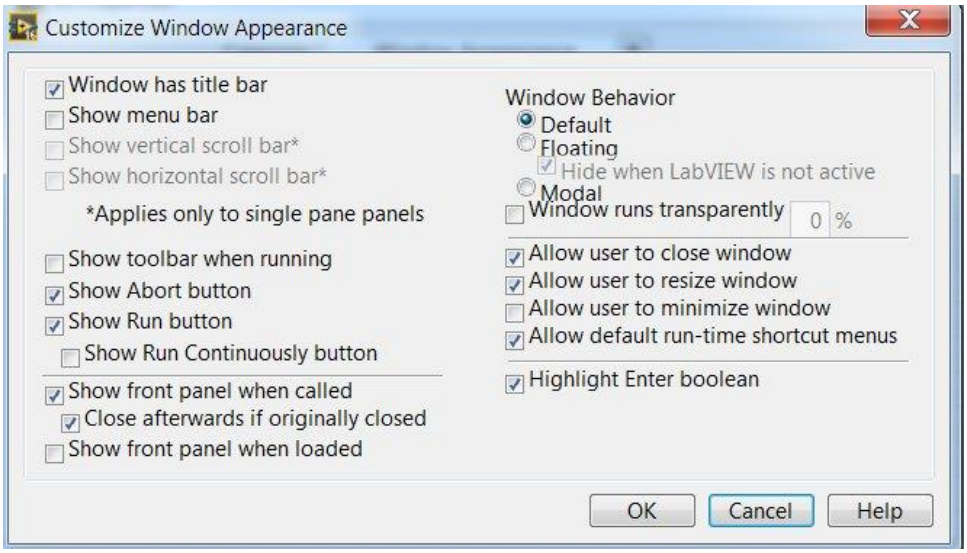
Posljednji zadatak prije kompletne izrade sučelja je uređivanje kontrola i indikatora na prednjem panelu ovog virtualnog instrumenta.

Na kontrole *Signali*, *Način prikaza*, *Konstanta* i *Mjerna jedinica* u prednjem panelu treba kliknuti desnom tipkom miša (lijevi dio gdje je brojka *0*) te iz izbornika *Visible Items* ukloniti kvačicu s opcije *Index Display*. Nakon toga ih treba poredati jednu pored druge kao što je prikazano na slici 3.40, a radi estetike i praktičnosti je uklonjen prikaz *Increment/Decrement* iz izbornika *Visible Items*, te su imena kontrola i indikatora prevedena na hrvatski jezik gdje je to bilo prikladno. Na slici je vidljiva i podjela kontrola po osobinama koristeći horizontalne razdjelnike (činiti iste korake kao i u virtualnim instrumentima pogleda, odnosno zaključati ih i odabrati opciju da svoj format mijenjaju proporcionalno), a na vrh je dodan logo Fakulteta elektrotehnike, računarstva i informacijskih tehnologija Osijek. Potonji i posljednji korak u izradi sučelja radi se tako da se na računalo s interneta preuzme slika po izboru programera, a potom u prednjem panelu iz menija *Edit* odabere opcija *Import Picture to Clipboard...*, nakon toga pronađe željena fotografija te potvrdi klikom na *OK*. Mišem nakon toga treba kliknuti na

proizvoljno područje na prednjem panelu (u ovom slučaju u gornji lijevi kut) i na tipkovnici odabrati *CTRL+V*, potom treba razvući sliku do svih rubova te kliknuti desnom tipkom miša bilo gdje na sliku i odabrati opciju *Scale Object with Pane*. Za sam kraj ostaje odlazak u svojstva virtualnog instrumenta nakon klika desnom tipkom miša na *connector pane*. U izborniku *General* treba staviti kvačicu na mogućnost *Separate compiled code from source file*, a u izborniku *Window Appearance* treba odabrati opciju *Custom* i potom kliknuti na gumb *Customize...*, a postavke u novootvorenom prozoru urediti prema slici 3.41. Virtualni instrument na kraju treba spremiti u mapu *Sučelje* pod imenom 'Sučelje'.



Sl. 3.40: *Prednji panel sučelja za akviziciju podataka.*



Sl. 3.41*: Postavke prikaza sučelja.*

3.1. Funkcioniranje sučelja za akviziciju podataka

Nakon otvaranja i pokretanja virtualnog instrumenta *Sučelje* strelicom na traci alata u LabVIEW-u, sekcija blok dijagrama ispod glavne petlje zadužena za prihvat DAQ signala počinje s akvizicijom podataka te signal/e na potom šalje u virtualni instrument *Spremnik podataka o signalima*, a pokretanjem tipke *Start* na sučelju za akviziciju podataka počinje rukovanje kontrolama od kojih korisnik traži željeni broj signala, same signale, način prikaza, otvara mu se prozor/i za unos konstantne mjernog člana te na posljetku može birati mjernu jedinicu. Događaj *Panel Close?* u strukturi događaja zadužen je stopiranje programa, događaj *"Broj signala": Value Change* zadužen je za postavljanje onoliko kontrola na prednji panel glavnog sučelja ovisno o tome koliko signala korisnik želi, a događaj *"Pokreni": Value Change* inicira jedan od osam prikaza ovisno o tome koliko signala se nalazi na ulazu u DAQ karticu.

Ti prikazi nalaze se u mapi *Pogledi*, a kao što se može vidjeti otvaranjem jednoga od tih pogleda, na ulazu u taj virtualni instrument/e se nalazi blok *Obrada signala* koji opet na svom ulazu ima spojen ranije spomenuti *Spremnik podataka o signalima* iz kojeg čita podatke s DAQ kartice, i koji se (zahvaljujući postavljenom *reentrant* statusu u postavkama samo virtualnog instrumenta *Obrada signala*) pojavljuje u pozadini i djeluje onoliko puta koliko signala na ulazu DAQ kartice postoji (i koji su uključeni u glavnom sučelju). Virtualni instrument *Obrada signala* sadrži sve kontrole koje su napravljene u ovom završnom radu, a nakon toga se sav traženi prikaz dobije zahvaljujući svojstvima jednoga od ranije spomenutih pogleda (koji dijele ekran na

temelju broja signala u glavnom sučelju) i na kojem se vidi djelovanje virtualnog instrumenta *Obrada signala*, a to se u glavnom sučelju manifestira zahvaljujući trećem događaju u strukturi događaja *("Pokreni": Value Change*) u kojoj se u petlji s desne strane nalazi blok *Sub Panel* (desni dio ekrana na prednjem panelu sučelja), gdje se konačnici vidi podjela ekrana na temelju broja ulaznih signala, te obrađivanje informacija u njemu/njima, što zahvaljujući virtualnom instrumentu *Obrada signala* može biti graf valnog oblika ili numerička vrijednost signala (*Mean (DC), TRMS, Peak to Peak*).

Budući da sučelje za akviziciju podataka (virtualni instrument *Sučelje*) za svoj rad koristi nekoliko potprograma, odnosno virtualnih instrumenata koje poziva u kratkom vremenu, moguće je da se prije traženog očitanja u prednjem panelu dobije zadrška od par sekundi, što predstavlja uobičajen događaj u akviziciji podataka sučeljem ovakvog tipa u LabVIEW-u (dinamička sučelja).

Iz tog razloga preporučuje se što manji vremenski interval između pokretanja virtualnog instrumenta (strelica u traci alata) i pokretanja događaja u prednjem panelu (tipka, odnosno blok *Pokreni*).

3.2. Rukovanje sučeljem za akviziciju podataka

Izgled prednjeg panela sučelja za akviziciju podataka s kojim će se susretati korisnik ovog alata pobliže je opisan slikom 3.42.

Sl. 3.42: *Kontrole sučelja za akviziciju podataka.*

U kontroli *Uređaj* korisnik bira karticu spojenu s računalom. Ako ona nije spojena s računalom, ovaj izbornik će biti prazan. Najčešće se radi o nazivima *Dev1*, *Dev2* i slično, ovisno o broju kartica koje su prethodno bile spojene s računalom na kojem se izvodi akvizicija podataka.

Kontrola *Konfiguracija terminala* nudi nekoliko mogućnosti: *Default*, *Differential*, *Pseudodifferential*, *RSE* i *NRSE*. Za zadatke akvizicije podataka u nastavnom laboratoriju je dovoljno ostaviti opciju *Default*, što uglavnom znači diferencijalnu konfiguraciju terminala kakva je korištena i pri testiranju ovoga rada.

Kontrola *Source* bira izvor terminala tzv. *Sample Clocka*. Iako ovisno o kartici ovaj padajući izbornik nudi nekoliko mogućnosti, u pravilu treba ostaviti uobičajeno, odnosno treba ga postaviti na *OnboardClock* koji koristi zadani *onboard clock* DAQ kartice.

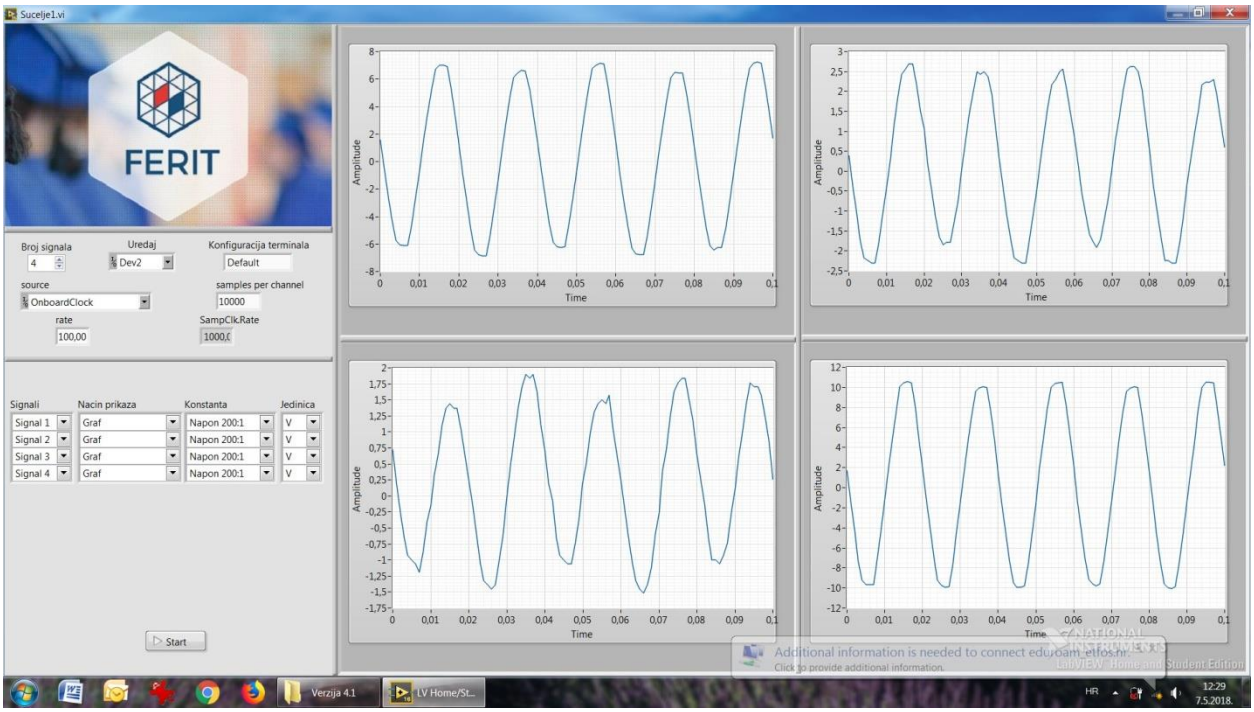
Brok uzoraka po kanalu i stopa uzimanja uzoraka ostavljene su korisniku na izbor, a ponekad ovise i o potrebama zadatka. U testiranju ovog rada upisano je bilo 100 uzoraka po kanalu i stopa

od 1000 uzoraka u sekundi. Posljednji u nizu nalazi se indikator stope uzoraka koji nije potrebno dirati.

Ispod kontrola kartice nalaze se kontrole sučelja za akviziciju podataka. Korisniku na raspolaganju stoji kontrola za biranje broja signala (od 1 do 8), kontrola za prikaz pojedinog signala (od prvog do osmog signala), kontrola konstante mjernog člana u koju korisnik proizvoljno upisuje iznos, te kontrola za mjernu jedinicu u kojoj je moguće odabrati volte ili ampere.

Važno je napomenuti da kontrola *Signal 1* čita prvi kanal kartice (*ai0*), *Signal 2* drugi kanal kartice (*ai1*), itd., a broj kanala se može povećati mijenjanjem postavki kontrole *Broj signala*, te pravljenjem dodatnih pogleda za slučaj više od osam signala na ulazu u karticu/sučelje.

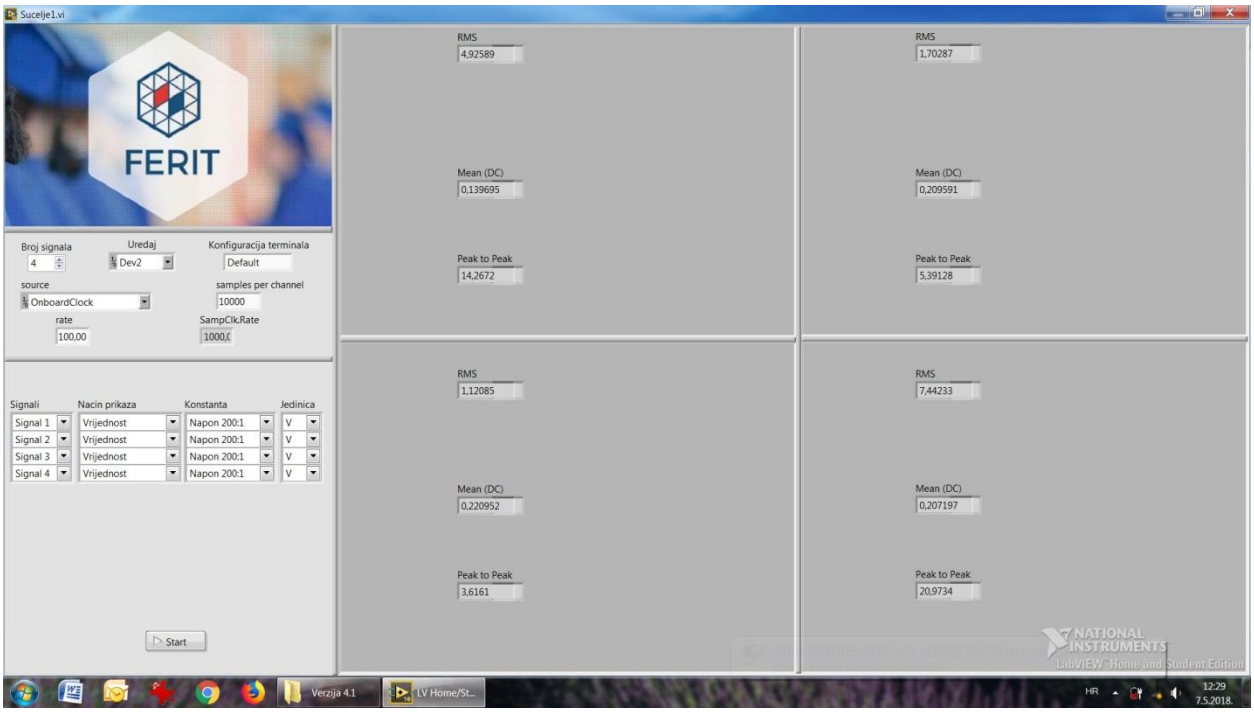
Sljedeće slike pokazuju izgled sučelja tijekom akvizicije različitih signala i pri različitim postavkama.



Sl. 3.43: *Prikupljanje četiri signala s ulaza kartice (valni oblici).*

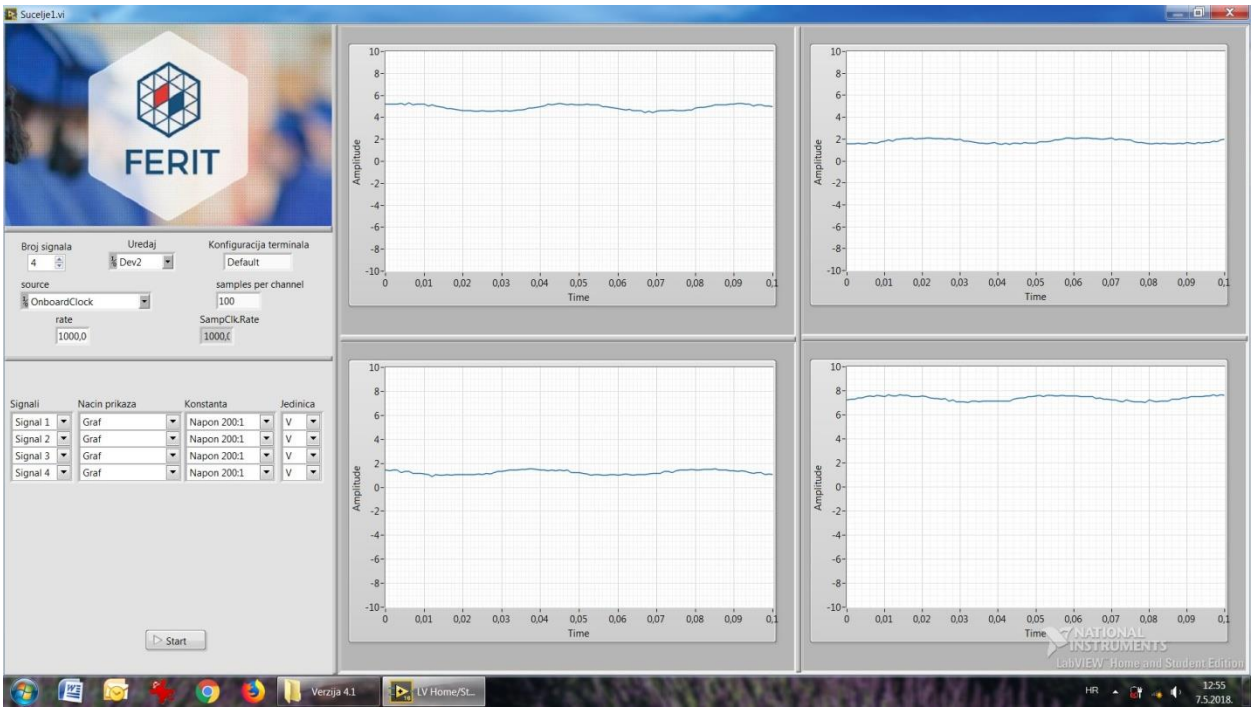
Slika 3.43 prikazuje četiri signala na ulazu DAQ kartice, pri zadanoj konfiguraciji terminala, i u ovom slučaju s 10 000 uzoraka po kanalu i stopom uzorkovanja od 100 uzoraka po sekundi, odnosno kako indikator pokazuje 1000 uzoraka što je u ovom slučaju definirano kontrolom

source, odnosno karticom NI USB-6218. Signali 1, 2, 3 i 4 odnose se na ulaze kartice *ai0*, *ai1*, *ai2* i *ai3*, prikazan je valni oblik, konstanta množi vrijednost s izlaza s kartice brojem 200 jer mjerni član WBK61 korišten u ovom radu prigušuje ulazni signal 200 puta, a mjerna jedinica su volti. Na slici se može vidjeti nešto drugačiji izgled kontrole za konstantu mjernog člana; u ovom slučaju se radilo o predefiniranim prijenosnim omjerima, dok u ovoj verziji korisnik može upisati proizvoljan iznos.



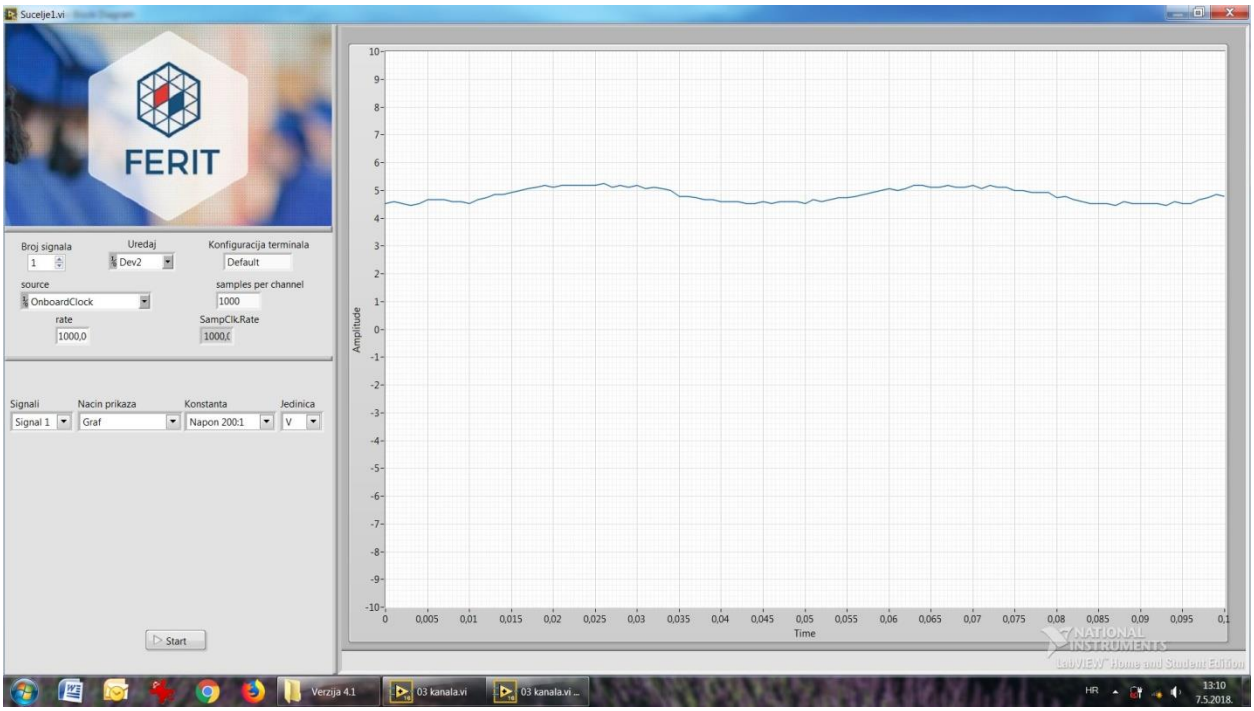
Sl. 3.44: *Prikaz numeričkih vijednosti četiri ulazna signala.*

Na slici 3.44 vidi se prikaz numeričkih vrijednosti signala s ulaza kartice, a to su *RMS*, *Mean (DC)* i *Peak to Peak*. U ovom slučaju se zahvaljujući LabVIEW-om paketu radi o točnim TRMS očitanjima, a za potrebe konačne verzije sučelja ova je kontrola preimenovana iz 'RMS' u 'TRMS', kako je prikazano ranije na slici 3.9.

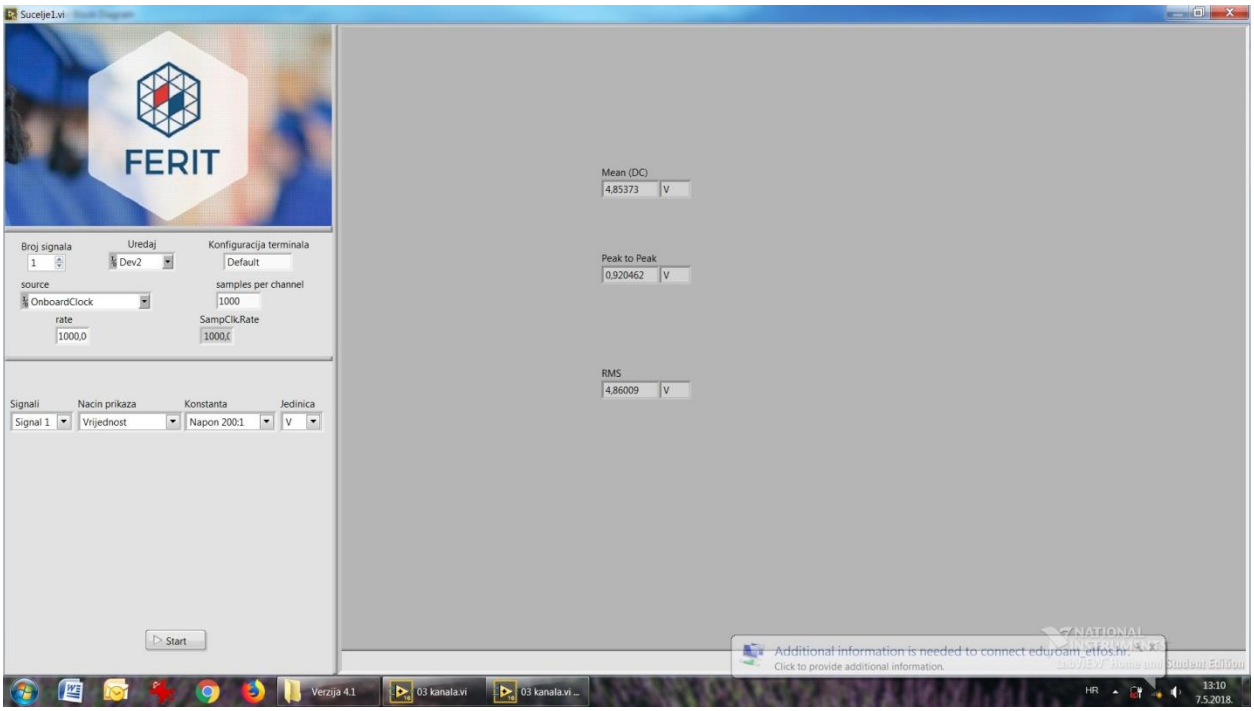


Sl. 3.45: *Akvizicija podataka DAQ karticom (istosmjerni signali).*

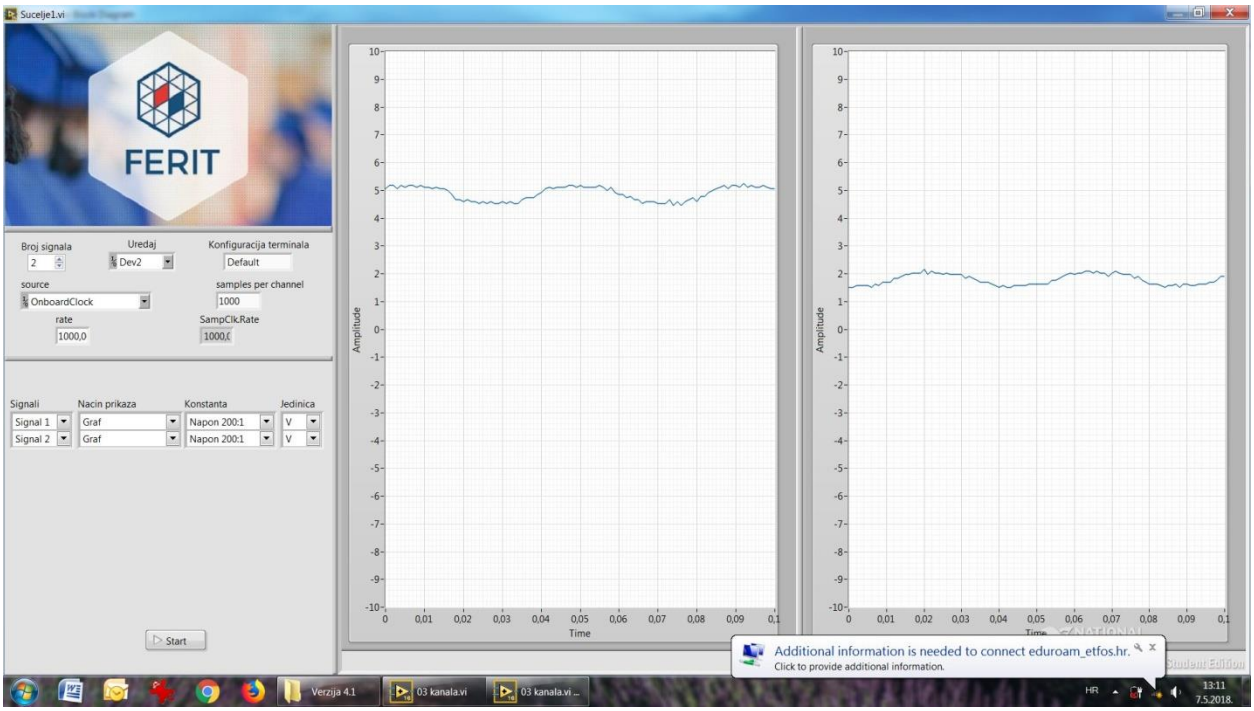
Slika 3.45 prikazuje korištenje zadane skale amplitude od -10 do 10 volti, što se uvijek može promijeniti u postavkama virtualnog instrumenta *Obrada signala*. Budući da je šum u ovom slučaju ometao prikupljanje podataka DAQ karticom za istosmjerni signal, isključena je mogućnost automatskog skaliranja Y osi (*Autoscale Y*) kako bi se dobio vjerodostojniji prikaz. Iako su šumovi bili maleni, pri automatskom skaliranju je Y os od vrha do vrha dijelilo nekoliko milivolti, a izgled valnog oblika tako možda ne bi bio vjerodostojan pa je u tu svrhu isključeno automatsko skaliranje. Ova mogućnost se može uključiti i u samom prednjem panelu glavnog sučelja (klikom na *Autoscale Y*), a na istom se mjestu može i isključiti pa u tom slučaju korisniku na raspolaganju ostaje skala ručno zadana u postavkama virtualnog instrumenta *Obrada signala*. I u testnom sučelju i u konačnoj izvedbi sučelja to iznosi *-10* i *10*.



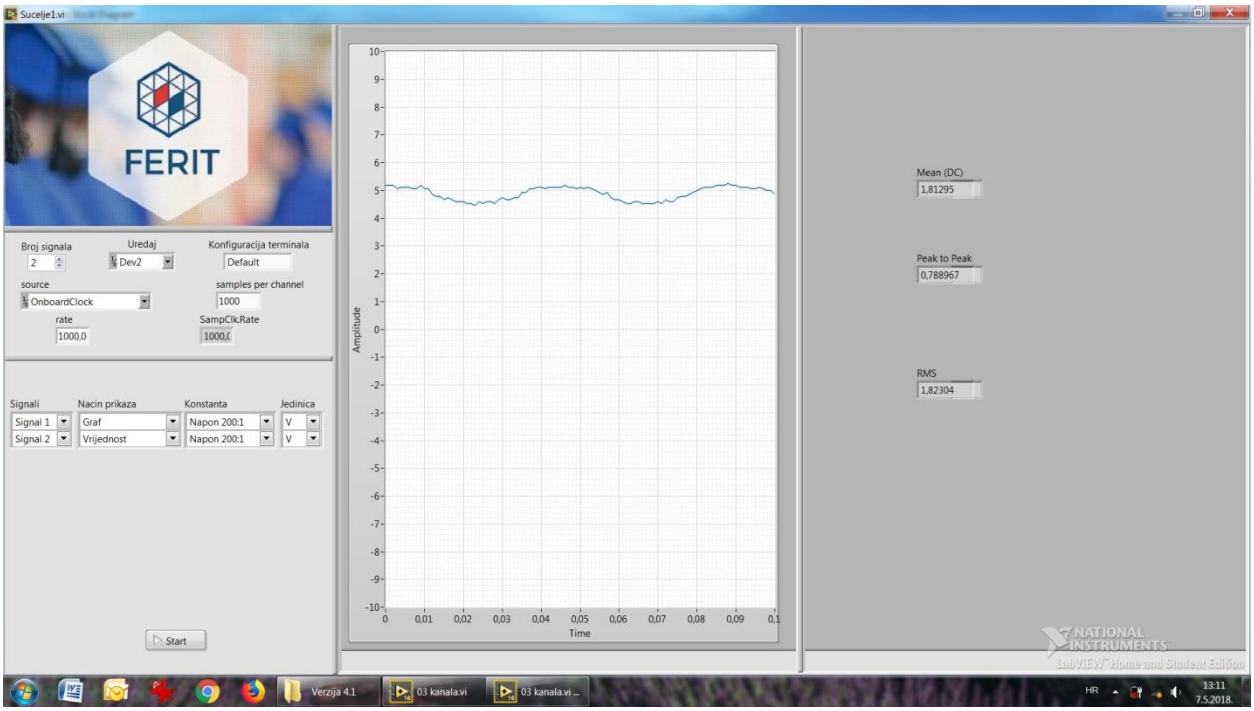
Sl. 3.46: *Prikaz jednog signala s ulaza DAQ kartice (valni oblik).*



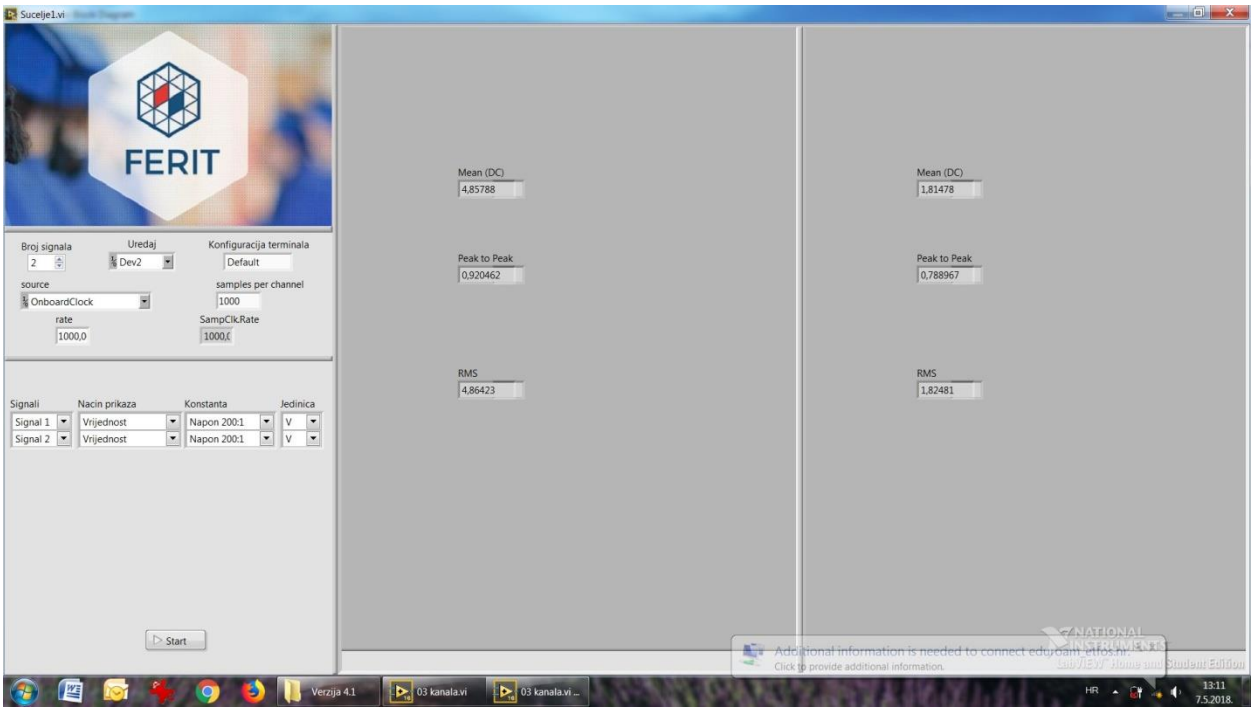
Sl. 3.47: *Prikaz jednog signala s ulaza DAQ kartice (vrijednosti).*



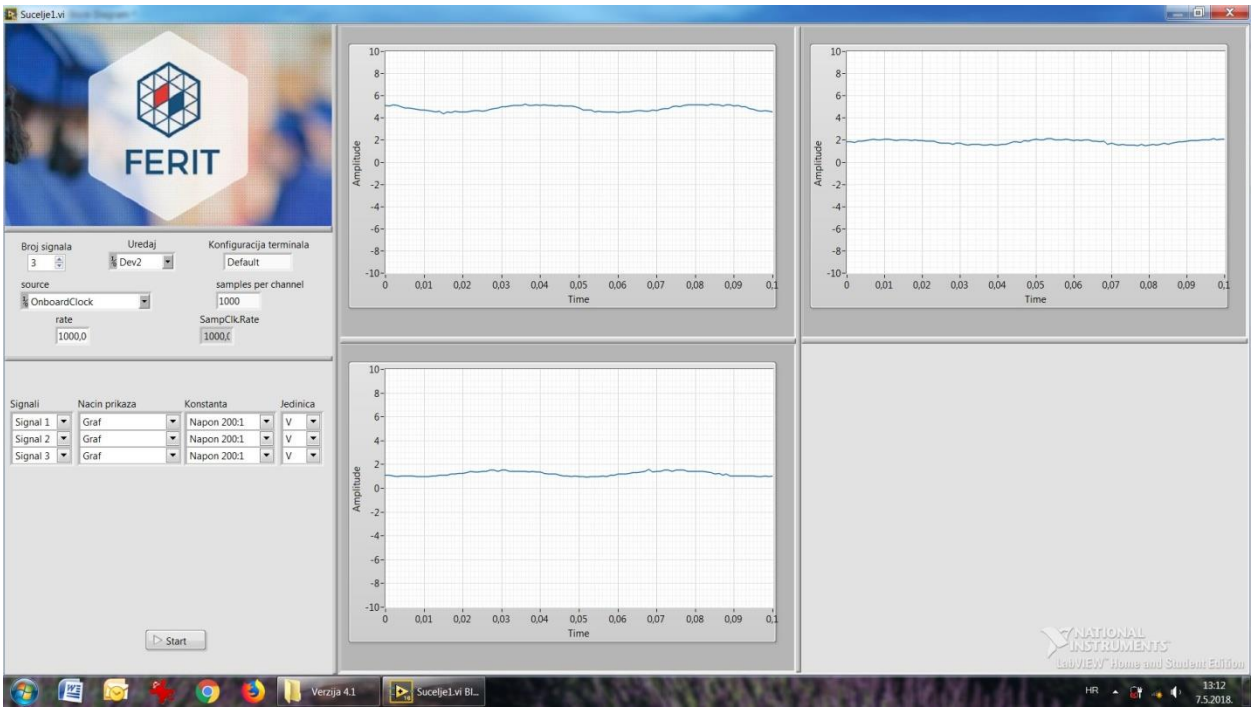
Sl. 3.48: *Prikaz dva signala s DAQ kartice (valni oblik).*



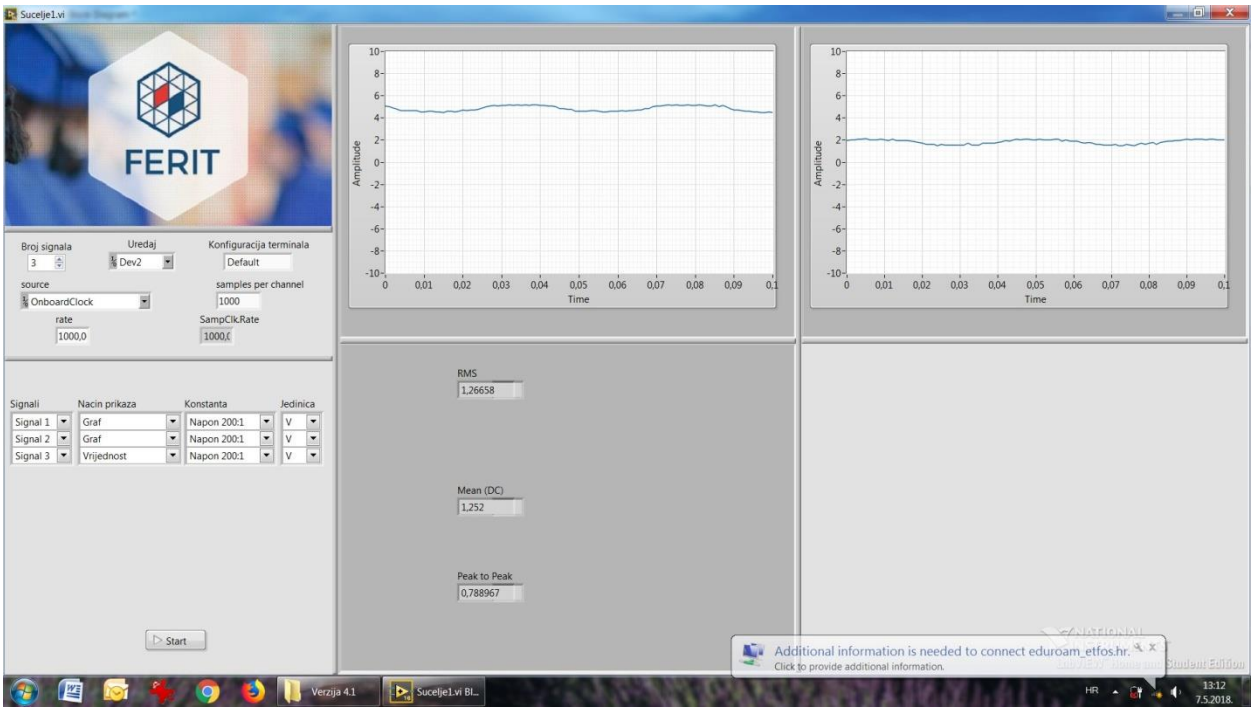
Sl. 3.49: *Prikaz dva signala s DAQ kartice (valni oblik/vrijednost).*



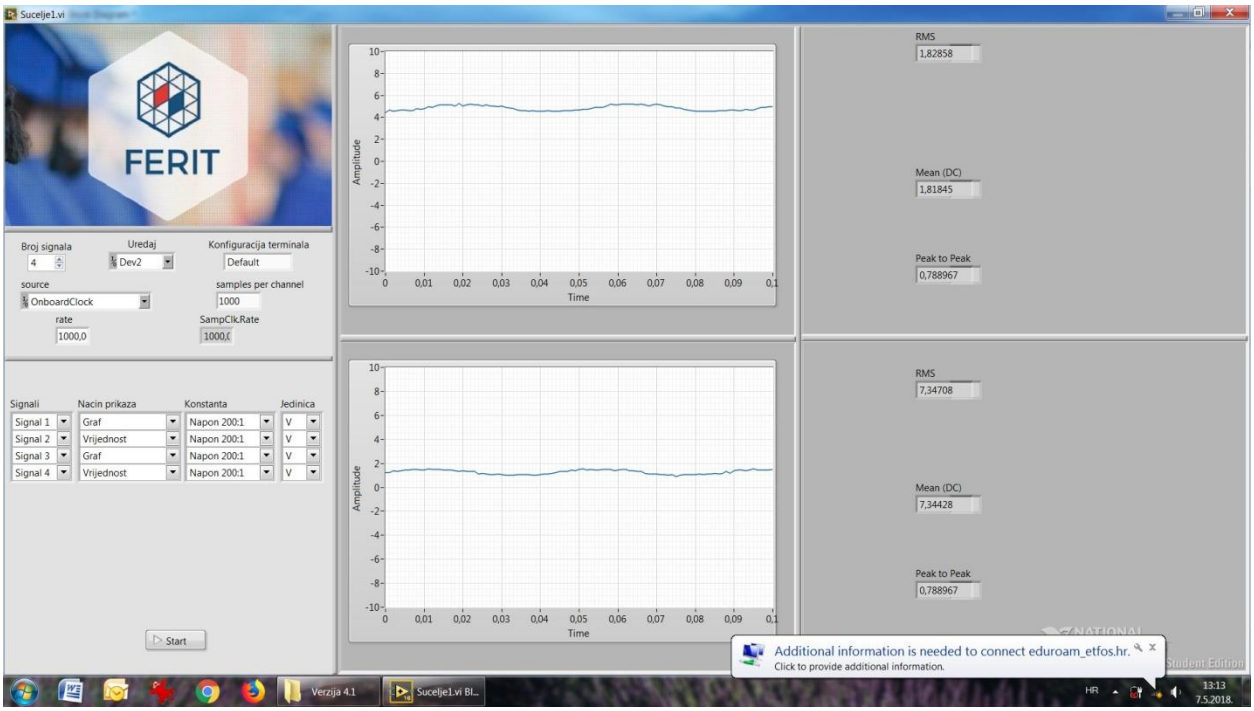
Sl. 3.50: *Prikaz dva signala s DAQ kartice (vrijednosti).*



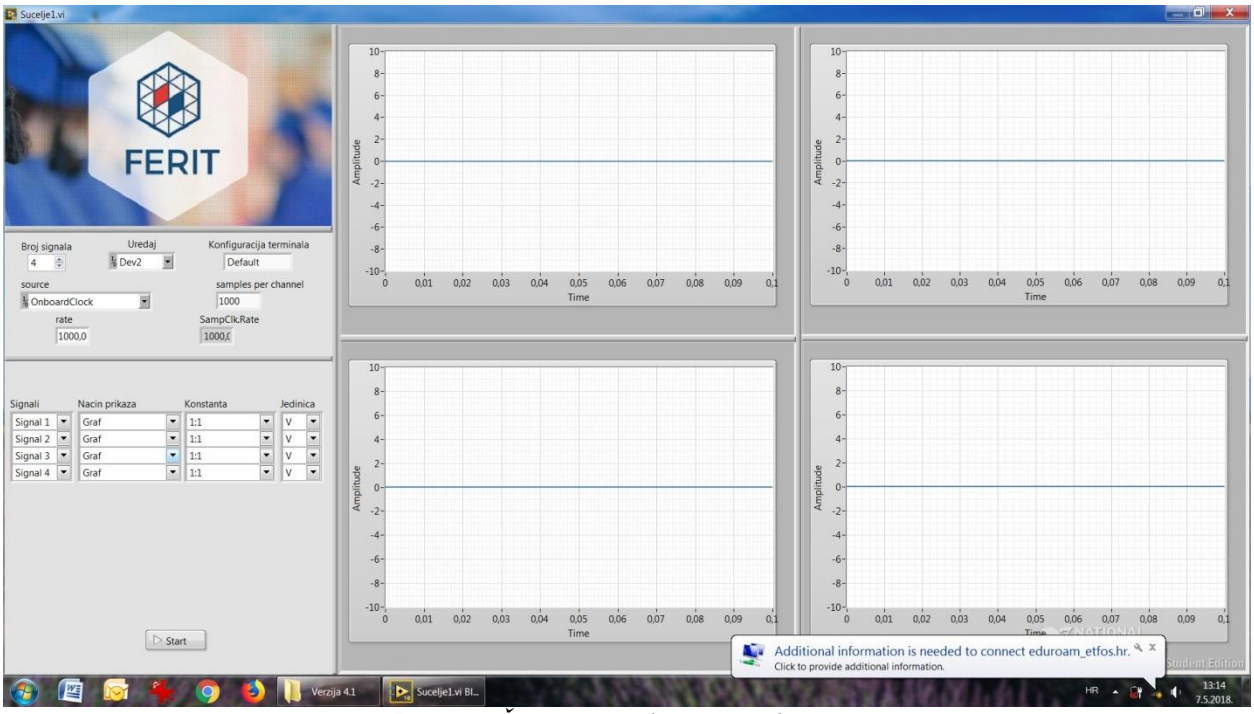
Sl. 3.51: *Prikaz tri signala s DAQ kartice.*



Sl. 3.52: *Prikaz tri signala s DAQ kartice (valni oblik/vrijednost).*

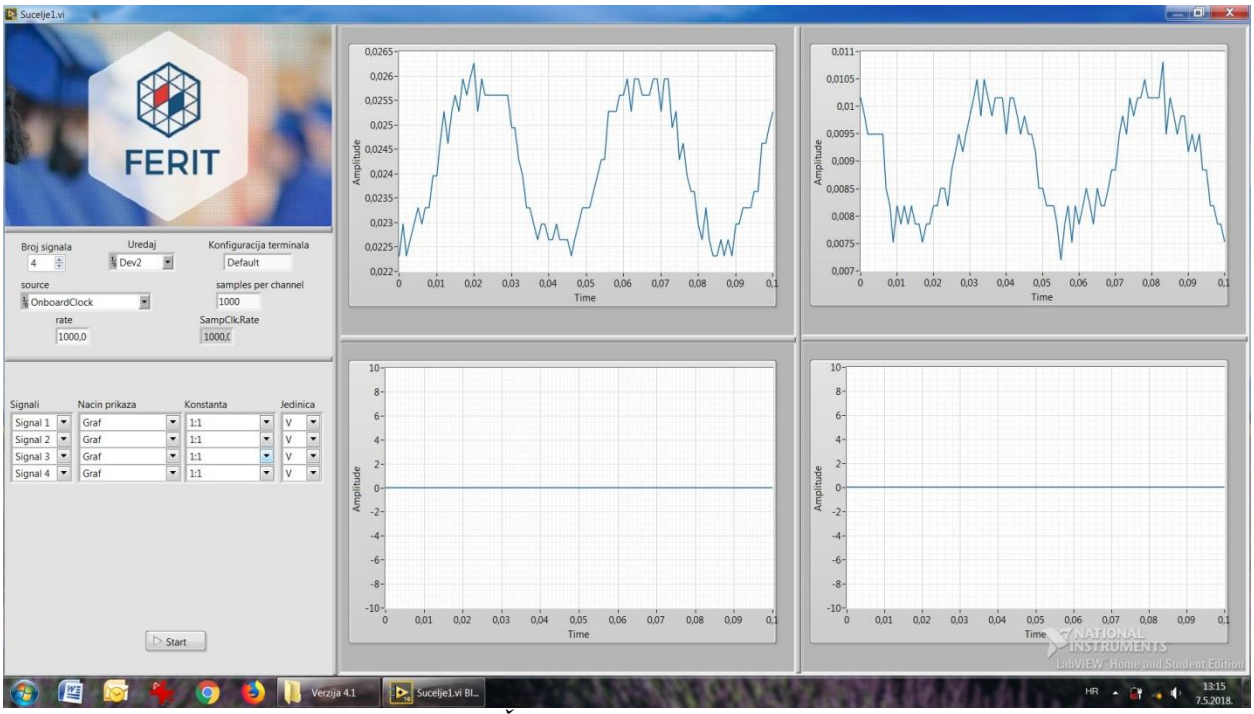


Sl. 3.53: *Prikaz četiri signala s DAQ kartice (valni oblik/vrijednost).*



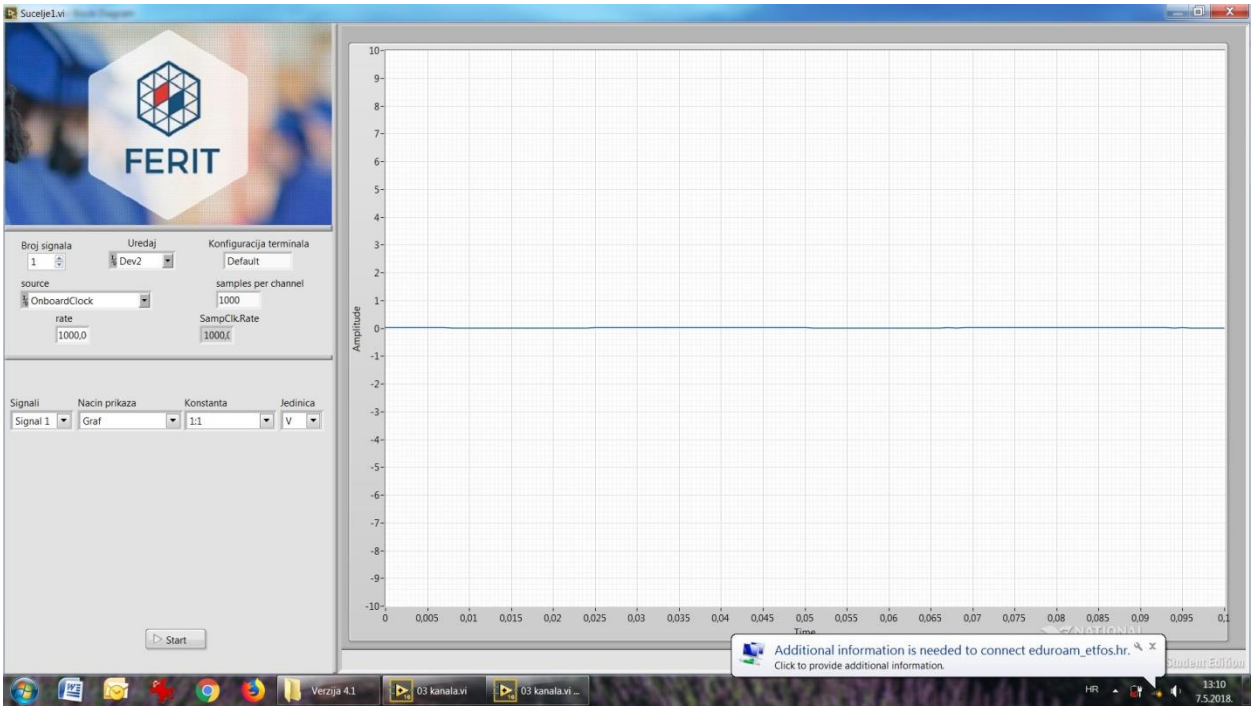
Sl. 3.54: *Četiri signala s DAQ kartice.*

Slika 3.54 prikazuje četiri signala čija je amplituda prigušena 200 puta mjernim članom WBK61, ali u postavkama sučelja ovaj puta nije odabrana kontrola koja ulazni signal pojačava 200 puta, tako da je vrijednost amplitude valnog oblika na prikazu sučelja praktički jednaka nuli.



Sl. 3.55: *Četiri signala s DAQ kartice.*

Slika 3.55 također prikazuje četiri signala kojima amplituda prigušena 200 puta, ali u kontroli konstante mjernog člana nije odabran isti iznos pojačanja ulaznog signala. Gornja dva signala pokazuju valni oblik kod uključene opcije *Autoscale Y*, a donja dva kod isključene opcije autoskaliranja.



Sl. 3.56: *Signal s DAQ kartice.*

Slika 3.56 prikazuje jedan prigušeni signal s DAQ kartice, ali također bez uključene kontrole za konstantu mjernog člana. Na ovoj slici nije korištena opcija automatskog skaliranja Y osi.

U slučaju da korisnik želi spremiti dobivene podatke, desnom tipkom miša može kliknuti na graf valnog oblika pri akviziciji podataka te je potrebno odabrati opciju *Export*, a zatim mu se nude tri mogućnosti: *Export Data To Clipboard* koja sprema podatke u međuspremnik iz kojega se one mogu prebaciti u grafički oblik, *Export Data To Excel* za pregled podataka u tabličnom obliku, te *Export Simplified Image...* koja sprema pojednostavljeni zapis signala. Osim toga, korisniku je uvijek na raspolaganju opcija snimke zaslona putem računala, koja također sve podatke, putem slike, sprema na proizvoljno mjesto na računalu.

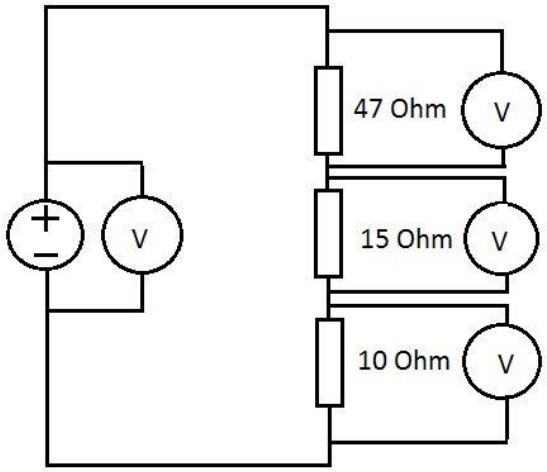
Pri izlasku iz sučelja za akviziciju podatka treba voditi računa da korisnik ni u kojem slučaju ne spremi ikakve izmjene koje mu u nekim okolnostima LabVIEW ponudi nakon izlaska iz virtualnog instrumenta.

4. MJERENJA

Verifikacija računalnog sučelja napravit će se testnim mjerenjima na mjernom postavu. Za mjerni objekt odabran je niz otpornika u nastavnom laboratoriju. Korišteno je više naponskih ulaza, a mjerni članovi su testirani korištenjem atestirane mjerne opreme (univerzalni multimetri, osciloskopi) te će se podaci prikupljeni mjerenjima provjeriti za daljnju analizu.

4.1. Dobivanje rezultata u LabVIEW-u

Nakon izrade sučelja programer je spreman za vršenje mjerenja putem DAQ kartice. Kao što je rečeno ranije u drugom poglavlju, za testiranje sučelja korišteni su otpornici u nastavnom laboratoriju fakulteta. Njihova otpornost iznosi 47 Ω, 15 Ω i 10 Ω, a shema spoja prikazana je na slici 4.1. Osim ovoga, na računalo je spojena DAQ kartica NI USB-6218, a na karticu su spojena četiri mjerna člana WBK61 koji su mjerili napone na otpornicima, te napon izvora. Za napajanje sustava poslužili su DC i AC izvor u nastavnom laboratoriju.

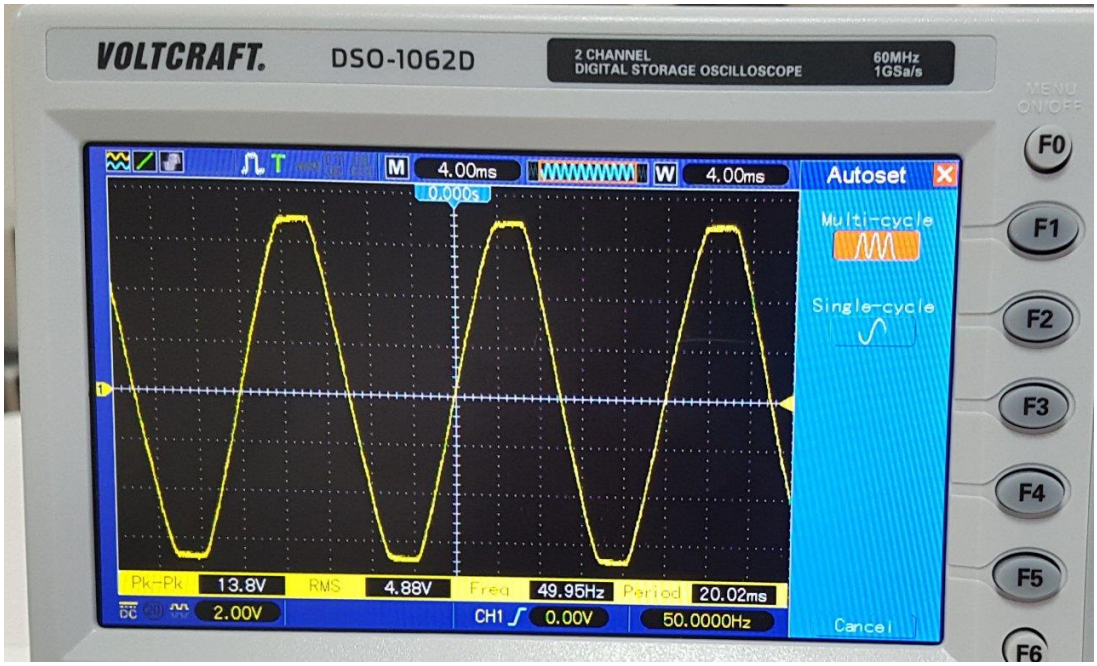


Sl. 4.1*: Shema spoja u nastavnom laboratoriju.*

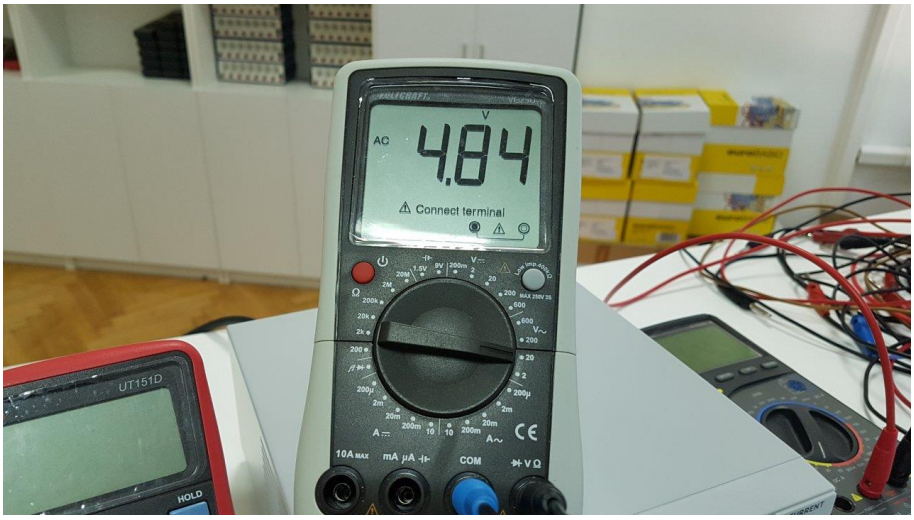
U blok dijagramu ili u prednjem panelu LabVIEW-a je potrebno kliknuti na strelicu za pokretanje programa koja se nalazi odmah ispod izborničke trake. Nakon par sekundi zadrške počinje akvizicija podataka mjerenja u LabVIEW-u.

Većih odstupanja rezultata mjerenja nije bilo, ali su ipak zabilježene manje devijacije koje su nastale kao rezultat nesavršenosti mjerne opreme u laboratoriju.

Za slučaj spajanja serije otpornika s izmjeničnim naponom od 7.2 V, osciloskop je na otporniku od 47 Ω prikazao valni oblik napona te vrijednost od 4.88 V (RMS), te vrijednost od vrha do vrha amplitude u iznosu od 13.8 V, što prikazuje slika 4.2. Multimetar je na istom otporniku izmjerio 4.84 V, što prikazuje slika 4.3.

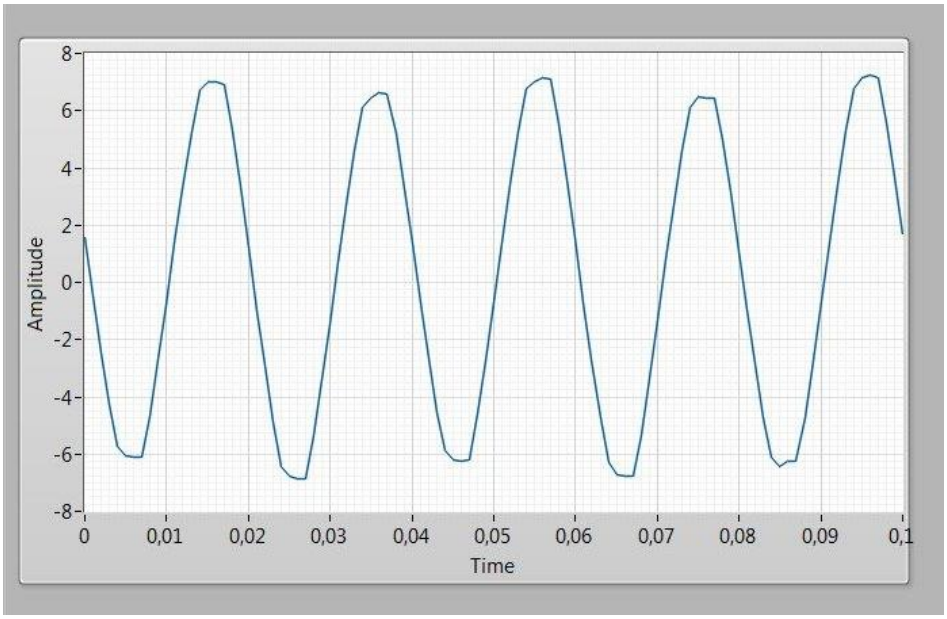


Sl. 4.2: *Mjerenje osciloskopom u nastavnom laboratoriju.*

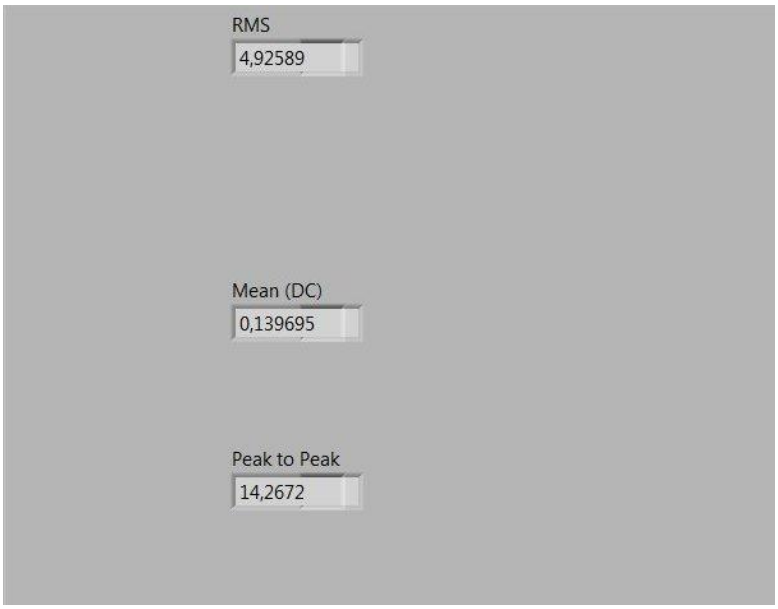


Sl. 4.3: *Mjerenje multimetrom u nastavnom laboratoriju.*

Sučelje za akviziciju podataka u LabVIEW-u je pokazalo očitanja koja su prikazana na slikama 4.4 i 4.5.



Sl. 4.4*: Valni oblik napona u LabVIEW-u.*

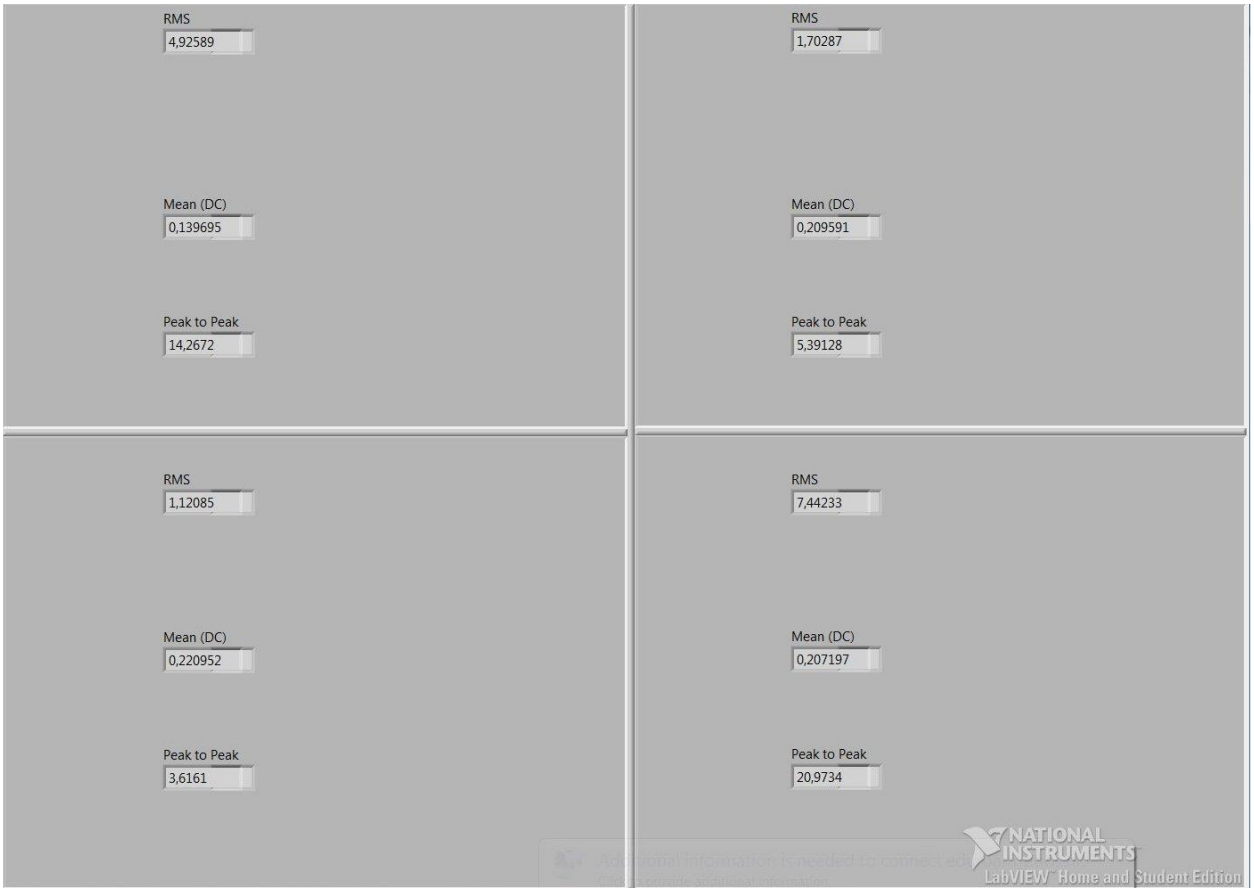


Sl. 4.5*: Vrijednosti napona u LabVIEW-u.*

Kao što se može vidjeti iz dobivenih očitanja, valni oblik napona u sučelju za akviziciju podataka je dosta sličan onome koji pokazuje osciloskop, dok su vrijednosti napona od 14.26 V (od vrha do vrha) te 4.92 V (TRMS) vrlo slične onima koje pokazuje osciloskop (13.8 V od vrha do vrha i 4.88 V RMS), ali i multimetar (4.84 V).

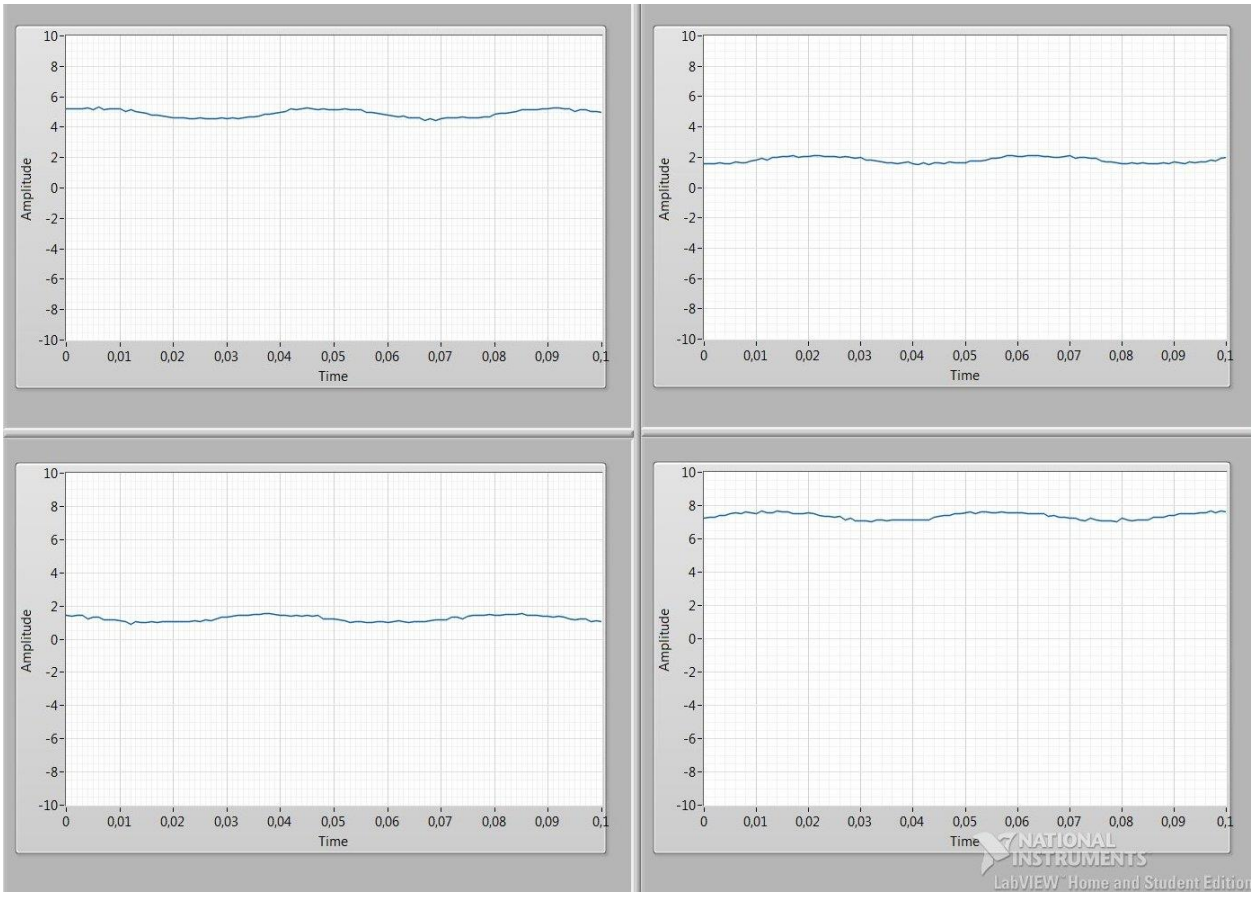
Odstupanje vrijednosti dobivenih sučeljem u odnosu na vrijednosti koje pokazuje multimetar iznosi 1,76%, a u odnosu na vrijednosti koje pokazuje osciloskop odstupanja iznose 0.92% (RMS) i 3.38% (od vrha do vrha).

S obzirom na napon od 7.2 V te poznate otpornosti sva tri otpornika, dobivene vrijednosti napona na ostala dva otpornika i na izvoru napajanja su u skladu s onime što se očekivalo. Napon na drugom otporniku od 15 Ω iznosi 1.7 V RMS, napon na otporniku od 10 Ω iznosi 1.12 V RMS, a napon izvora 7.44 V RMS, što prikazuje slika 4.6.

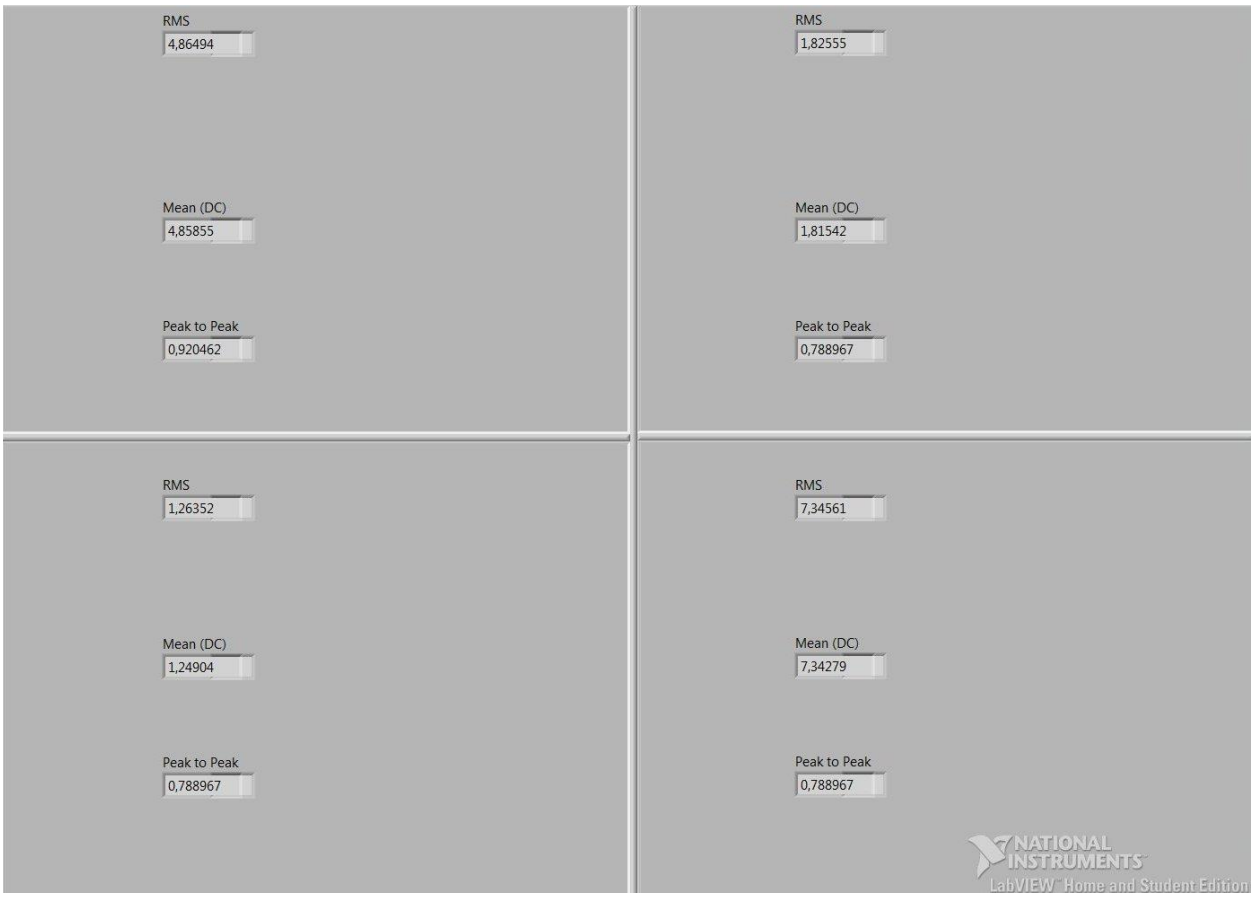


Sl. 4.6: *Vrijednosti napona na mjernom postavu u nastavnom laboratoriju.*

Za DC mjerenja su također dobivene slične vrijednosti u prikazu sučelja, osciloskopa i kontrolnih multimetara. Prikazi valnog oblika i numeričkih vrijednosti napona vidljivi su na slikama 4.7 i 4.8.



Sl. 4.7: *Valni oblici napona za istosmjerna mjerenja.*

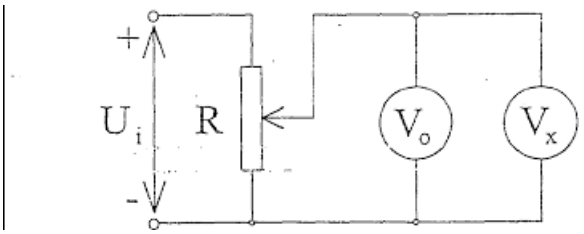


Sl. 4.8: *Vrijednosti napona za istosmjerna mjerenja.*

4.2. Umjeravanje mjernih članova

Umjeravanje je postupak ispitivanja mjerila, odnosno u ovom slučaju mjernog člana. Postupak se sastoji od usporednog mjerenja ispitivanog mjerila s etalonom kako bi mjeritelj po okončanju postupka utvrdio je li mjerilo ispravno ili ne.[1] Cilj je utvrditi sistematske pogreške mjerne opreme, tj. utvrditi zadovoljava li ono specifikacije u pogledu točnosti.[1]

Umjeravanje visokonaponskih mjernih članova se može provesti na način sličan onome pri postupku umjeravanja voltmetra. Jedina razlika je u tome što će voltmetar koji treba umjeriti zamijeniti naponskim mjernim članom, dok će drugi, referentni (etalon) voltmetar ili digitalni multimetar sa zanemarivo malom nesigurnošću poslužiti kao referenca naponskih mjernih članova. Poopćeni prikaz sheme pri postupku umjeravanja voltmetra, koji se može primijeniti i na umjeravanje naponskih mjernih članova iz ovog završnog rada, prikazan je na slici 4.9.

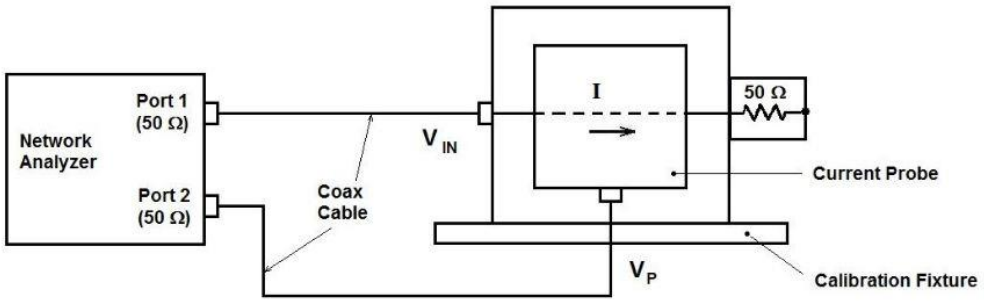


Sl. 4.9: *Umjeravanje naponskih mjernih članova referentnim voltmetrom.*

Vx označava mjerni član kojeg treba umjeriti. U praksi to znači da će se referentni voltmetar spojiti na priključnice otpornika ili nekog drugog trošila te da će služiti za umjeravanje naponskih mjernih članova koji su spojeni s DAQ karticom. Ovisno o broju mjerenja, zadatak programera je da iz svih apsolutnih odstupanja zabilježi ono najveće, te da izračuna najveću postotnu graničnu pogrešku koju na koncu treba usporediti sa specificiranom postotnom graničnom pogreškom.[1] Da bi mjerni član bio ispravan, njegovo odstupanje mora biti manje od onog koje je naznačeno u njegovim specifikacijama, odnosno manje od njegove specificirane postotne granične pogreške.

S obzirom na to da multimetri u laboratoriju nisu predstavljali etalon sa zanemarivo malom nesigurnošću, po obavljanju mjerenja u laboratoriju ustanovljenja su odstupanja vrijednosti dobivenih sučeljem za akviziciju od podataka u odnosu na vrijednosti izmjerenih multimetrom u iznosu od 1.76% (RMS), odnosno u odnosu na vrijednosti dobivene osciloskopom u iznosu od 0.92% (RMS) i 3.38% (od vrha do vrha), kao što je spomenuto u prethodnom potpoglavlju.

Procedura umjeravanja mjernih članova je od presudne važnosti da bi prikupljeni podaci imali pravovaljane vrijednosti. Umjeravanje strujnih mjernih članova (kakvi se također nalaze u nastavnim laboratorijima Fakulteta elektrotehnike, računarstva i informacijskih tehnologija) se uglavnom provodi posebnim priborom koji je često dizajniran specifično za strujne mjerne članove kako bi se izvršila što točnija procedura umjeravanja. Pribor se sastoji od mrežnog analizatora na jednoj strani, dok je s druge strane priključen teret od 50 Ω.



Sl. 4.10: *Postupak umjeravanja strujnog mjernog člana.[24]*

Mrežni analizator mjeri omjer napona koji se pojavljuje na izlazu strujnog člana (*Vp*) s onim koji upravlja kalibracijskim priborom (*Vin*).[24] Odnos između napona strujnog mjernog člana u odnosu na struju kroz član je definiran prijenosnom impedancijom *Zt*. Prema slici 4.10, to je:

$$\boldsymbol{Zt = \frac{Vp}{I} = \frac{Vp}{\frac{Vin}{50\ \Omega}} = \frac{Vp}{Vin} * 50\ \Omega} \quad (5\text{-}1)$$

S obzirom na to da je *Zt* često izražena u dBΩ, formula 5-9 može poprimiti sljedeći oblik:

$$\boldsymbol{20log\ (Zt) = 20log\ (\frac{Vp}{I})} \quad (5\text{-}2)$$

$$\boldsymbol{20log\ (Zt) = 20log\ (Vp) - 20log\ (\frac{Vin}{50})} \quad (5\text{-}3)$$

$$\boldsymbol{20log\ (Zt) = 20log\ (Vp) - 20log\ (Vin) + 20log(50)} \quad (5\text{-}4)$$

$$\boldsymbol{20log\ (Zt) = 20log\ (Vp) - 20log\ (Vin) + 34\ dB} \quad (5\text{-}5)$$

Prema formuli 5-5 se može zaključiti da na rezultat mjerenja mrežnog analizatora treba dodati 34 decibela ako se *Vp* i *Vin* mjere u dBV. U slučaju da strujni mjerni član ima prijenosnu impedanciju *Zt* od 0 dBΩ, odnosno 1 Ω, početno mjerenje mrežnog analizatora će pokazati omjer od -34 dB. Tada na rezultat također treba dodati 34 dB.

5. ZAKLJUČAK

Prikupljanje podataka akvizicijskim karticama će sasvim sigurno uskoro imati sve veću ulogu u radu nastavnih laboratorija, gdje će jedan modul, jedno računalo i jedno sučelje zahvaljujući brojnim prednostima ponuditi kvalitetnija, brža i jednostavnija mjerenja raznih, ne samo električnih veličina. Dosadašnji način rada s brojnim uređajima, multimetrima, osciloskopima i ostalim napravama ima nedostatak što, osim zauzimanja velikog prostora, zahtijeva i stalno održavanje. Ovakvi problemi bi mogli praktički nestati upotrebom DAQ kartica, univerzalnih naprava koje objedinjuju brojne uređaje, funkcije i operacije u jednom malom modulu.

No, ovaj rad je pokazao da ovakav način prikupljanja podataka još uvijek ima nedostatke koje bi trebalo otkloniti prije potpunog uvođenja u rad i zamjene svih ostalih komponenata nastavnog laboratorija. U prvom redu se to odnosi na programsku potporu do koje nije jednostavno doći. Internetskim prostorom se nude razne inačice aplikacijskog softvera, ali bez prave verzije korisnik vrlo brzo može ostati bez potpune funkcionalnosti zbog čega trpi njegov rad na akviziciji podataka. Osim toga, iako veoma solidan alat, LabVIEW uglavnom treba dodatne upravljačke programe i dodatke kako bi mogao dati sve od sebe, a sve to opet za sobom povlači pitanje pronalaska stabilnih i potpunih verzija aplikacijskog i upravljačkog softvera.

Velik problem predstavlja i nekompatibilnost među prethodnim verzijama programske podrške pa tako korisnik noviji projekt neće moći otvoriti u starijoj inačici LabVIEW-a. Ipak, ostavljena mu je mogućnost spremanja datoteke za otvaranje sa starijom inačicom sustava.

U ovom su se radu pokazala određena odstupanja kod mjerenja naponskim mjernim članovima, a umjeravanje ovih članova nije uvijek jednostavan i brz posao.

No, nema sumnje da je pred ovakvim načinom rada u laboratorijima svijetla budućnost. Cilj ovog završnog rada je ispunjen, a to je u prvom redu prikupljanje, prikaz i pohrana podataka koji su dobiveni korištenjem DAQ kartice s više ulaza i sučelja za akviziciju podataka u nastavnom laboratoriju.

LITERATURA

[1] Z., Godec, D., Dorić, Osnove mjerenja, laboratorijske vježbe, Elektrotehnički fakultet Sveučilišta J. J. Strossmayera u Osijeku, Osijek, 2007.

[2] Voltcraft VC250 Digital Multimeter
www.produktinfo.conrad.com, pristupljeno 8.5.2018.

[3] DSO-1062D Digital Oscilloscope User Manual
voltcraftdownload.info, pristupljeno 8.5.2018.

[4] Picotech TA167
https://www.picotech.com/accessories/current-probes-clamps/200a-2000a-current-probe-bnc, pristupljeno 13.5.2018.

[5] Chauvin-Arnoux En series
http://www.chauvin-arnoux.com/sites/default/files/D00SAL08_14.PDF, pristupljeno 19.7.2016.

[6] Tektronix, Tektronix Current Probes A621&A622 Datasheet, 22.1.2016.
http://www.tek.com/sites/tek.com/files/media/media/resources/A621-A622-Current-Probes-Datasheet-60W150813.pdf, pristupljeno 13.5.2018.

[7] I., Šubarić, Mjerenje struje bez prekidanja strujnog kruga, završni rad, Elektrotehnički fakultet Sveučilišta J. J. Strossmayera u Osijeku, Osijek, 2016.

[8] K., Bičanić, I., Kuzle, Nekonvencionalni mjerni pretvarači, Energija, god. 55 (2006), br. 3., str. 328-351, travanj 2006.
hrcak.srce.hr/file/7441, pristupljeno 13.5.2018.

[9] Instructions Tektronix A622
http://docs-europe.electrocomponents.com/webdocs/0065/0900766b800651a2.pdf, pristpuljeno 13.5.2018.

[10] Elmark, WBK61 & WBK62 High-Voltage Adapters With Probes
http://support.elmark.com.pl/iotech/dbk_wbk/WBK61_WBK62.pdf, pristupljeno 13.5.2018.

[11] WBK61 and WBK62
http://www.mccdaq.com/pdfs/manuals/wbk61_wbk62.pdf, pristupljeno 13.5.2018.

[12] Wikipedia, Data Acquisition, 6.4.2016.
https://en.wikipedia.org/wiki/Data_acquisition, pristupljeno 13.5.2018.

[13] S., Kuamr, S., Mohammad, PC Based Data Acquisition System
http://ethesis.nitrkl.ac.in/1791/2/DAQ-pcl208.pdf, pristupljeno 18.7.2016.

[14] What is Data Acquisition
http://www.ni.com/data-acquisition/what-is/, pristupljeno 13.5.2018.

[15] S. Štefanko, Ž. Hederić, M. Hadžiselimović, I. Zgradišnik, Analyses of Shaft Currents in Low-Voltage Induction Motor For Forklift Drive With Electronic Equipment, Informacije MIDEM 37(2007)3, str. 152-157, rujan 2007.
http://www.midem-drustvo.si/Journal%20papers/MIDEM_37(2007)3p152.pdf, pristupljeno 13.5.2018.

[16] Programming NI-DAQ in Microsoft Visual Basic .NET, 12.2.2014.
www.ni.com/product-documentation/54437en/, pristupljeno 13.5.2018.

[17] Data Acquisition Toolbox
http://www.mathworks.com/products/daq/features.html#key-features, pristupljeno 13.5.2018.

[18] National Instruments, Learn 10 Functions in NI-DAQmx and Handle 80 Percent of Your Data Acquisition Applications
http://www.ni.com/product-documentation/2835/en/#toc2, pristupljeno 13.5.2018.

[19] H.P., Halvorsen, Data Acquisition in LabVIEW, 16.8.2013.
http://home.hit.no/~hansha/documents/labview/training/Data%20Acquisition%20in%20LabVIEW/Data%20Acquisition%20in%20LabVIEW.pdf, pristupljeno 13.5.2018.

[20] T., Tomiša, Procesna mjerenja i dijagnostika u energetici
https://www.fer.unizg.hr/_download/repository/PREDAVANJE-1.pdf, pristupljeno 18.7.2016.
https://www.fer.unizg.hr/_download/repository/PREDAVANJE-2.pdf, pristupljeno 18.7.2016.

[21] National Instruments, USB-6218
http://www.ni.com/pdf/manuals/375199d.pdf, pristupljeno 13.5.2018.

[22] NI USB-6218 BNC
sine.ni.com/nips/cds/view/p/lang/en/nid/207098, pristupljeno 13.5.2018.

[23] Wikipedia, LabVIEW, 19.8.2016.
https://en.wikipedia.org/wiki/LabVIEW, pristupljeno 13.5.2018.

[24] EMCTEST Technologies, Current Probe Calibration

http://www.proveip.it/public/pages/strumentazione/, pristupljeno 22.6.2016.

SAŽETAK

Zadatak ovog rada je izrada grafičkog sučelja za akviziciju podataka mjerenja DAQ karticom s više ulaza, prikupljenih različitim naponskim mjernim članovima. Kao alat za izradu sučelja je odabran LabVIEW, a kao akvizicijski modul je poslužio NI USB-6218.

Obrađeni su osnovni pojmovi vezani za akviziciju podataka u računalo korištenjem DAQ kartica, a proučavanjem korisničkih uputstava i tehničkih specifikacija svih naprava korištenih u ovom radu pružen je detaljniji prikaz rada s LabVIEW-om i njegovim upravljačkim programima.

Nakon toga se pristupilo izradi modula koji će osigurati prikupljanje, prikaz i pohranu vrijednosti mjernih članova za mjerenje naponskih vrijednosti.

Nakon obavljenih mjerenja, podaci dobiveni programskim sučeljem (DAQ + računalo + mjerni članovi) su uspoređeni s onima koje su izmjerili instrumenti (multimetri i osciloskop) te su zabilježena određena odstupanja. U tu svrhu je na kraju opisana procedura umjeravanja mjernih članova.

Programsko sučelje omogućava daljnju prilagodbu krajnjem korisniku ovisno o mjernom objektu, mjernim članovima te svrsi samog ispitivanja.

Ključne riječi: Akvizicija podataka, DAQ kartica, LabVIEW, naponski mjerni članovi.

ABSTRACT

Creating interfaces for data acquisition measurements in the teaching lab

Mission of this paper is production of graphic interface for data acquisition using DAQ card with several inputs, which are collected by different high-voltage probes. LabVIEW was chosen as a tool for accomplishing this task, and NI USB-6218 served as an acquisition module.

Basic terms with regard to data acquisition on computer using DAQ cards are processed, while looking into user manuals and technical specifications of all devices used in this work provided more detailed insight into work with LabVIEW and its drivers.

Next step was creating a module which will ensure data acquisition, representation and saving values from voltage probes for voltage measurements.

After measurements were done, data collected with interface (DAQ + PC + probes) was compared to those which were measured by instruments (multimeters and oscilloscope) and some deviations were noticed. That's the reason why procedure of probe calibration is mentioned at the end.

This interface enables further adaption to the end user depending on objects of measurement, probes and purpose of examination.

Keywords: Data Acquisition, DAQ card, LabVIEW, voltage probes.

ŽIVOTOPIS

Vedran Kristić rođen je 16. ožujka 1988. godine u Slavonskom Brodu. Osnovnu školu je pohađao u razdoblju od 1994. do 2002. godine. Nakon toga upisuje srednju Tehničku školu u Slavonskom Brodu, područje elektrotehnika, smjer tehničar za telekomunikacije, koju završava 2006. godine. Nakon civilno odsluženog vojnog roka u tadašnjoj 11. osnovnoj školi Slavonski Brod, 2008. godine upisuje sveučilišni preddiplomski studij Elektrotehničkog fakulteta u Osijeku, smjer Elektrotehnika, odnosno od druge godine smjer Komunikacije i informatika.