

LabVIEW

Prva vežba

Opis LabVIEW programskog paketa

Programi napisani u LabVIEW programskom paketu predstavljaju virtuelne instrumente jer njihov izgled i funkcionalnost odgovaraju realnim instrumentima.

LabVIEW sadrži brojne alate za prikupljanje, analizu, prikaz i skladištenje podataka.

Programiranje u okviru LabVIEW-a se svodi na dva koraka

1. Kreiranje korisničkog interfejsa (front panel) postavljanjem raznih kontrola i indikatora.

Kontrole su: okretna dugmad (knobs), klizači (sliders), tasteri (push buttons), prekidači (switches) itd..

Indikatori su: grafovi, LE diode, displeji, lampice, itd...

2. Pisanje koda za kontrolu korisničkog interfejsa. Ovde pisanje koda uglavnom znači kreiranje blok dijagrama sa već definisanim strukturama koje se nalaze u bibliotekama LabVIEW-a gde se strukture prilagođavaju potrebama podešavanjem izmenjivih parametara (konfigurisanjem).

LabVIEW se može koristiti za povezivanje sa raznim hardverom za akviziciju podataka, upravljanje procesima, vizualizaciju (kamere), kao i sa GPIB, RS232, RS485 uređajima

Kreiranje virtuelnog instrumenta

Pokretanjem LabVIEW okruženja (v 8.6) pojavljuje se “Getting Started” ekran koji ubrzava početak rada. U okviru ovog ekrana nalaze se opcije za:

1. Kreiranje novog projekta
2. Otvaranje postojećeg projekta
3. Upoznavanje sa raznim aspektima programa kroz help sistem



Files

New

- Blank VI
- Empty Project
- VI from Template...
- More...

Kreiranje novog projekta

Open

- Primer.vi
- Browse...

Upoznavanje sa okruzenjem

Otvaranje postojeceg projekta

Resources

New To LabVIEW?

- Getting Started with LabVIEW
- LabVIEW Fundamentals
- Guide to LabVIEW Documentation
- LabVIEW Help

Upgrading LabVIEW?

- Automatic Block Diagram Clean Up
- Quick Drop
- Properties of Multiple Objects
- List of All New Features

Web Resources

- Discussion Forums
- Training Courses
- LabVIEW Zone

Examples

- Find Examples...

LabVIEW sadrži ugrađene šablone virtuelnih instrumenata (templates) koji sadrže funkcije, strukture, panele neophodne za većinu jednostavnih aplikacija za merenje.

Za primer virtuelnog mernog instrumenta koristiće se upravo jedan ovakav šablon.

Virtuelni instrument koji se ovde ilustruje ima jednostavnu funkciju

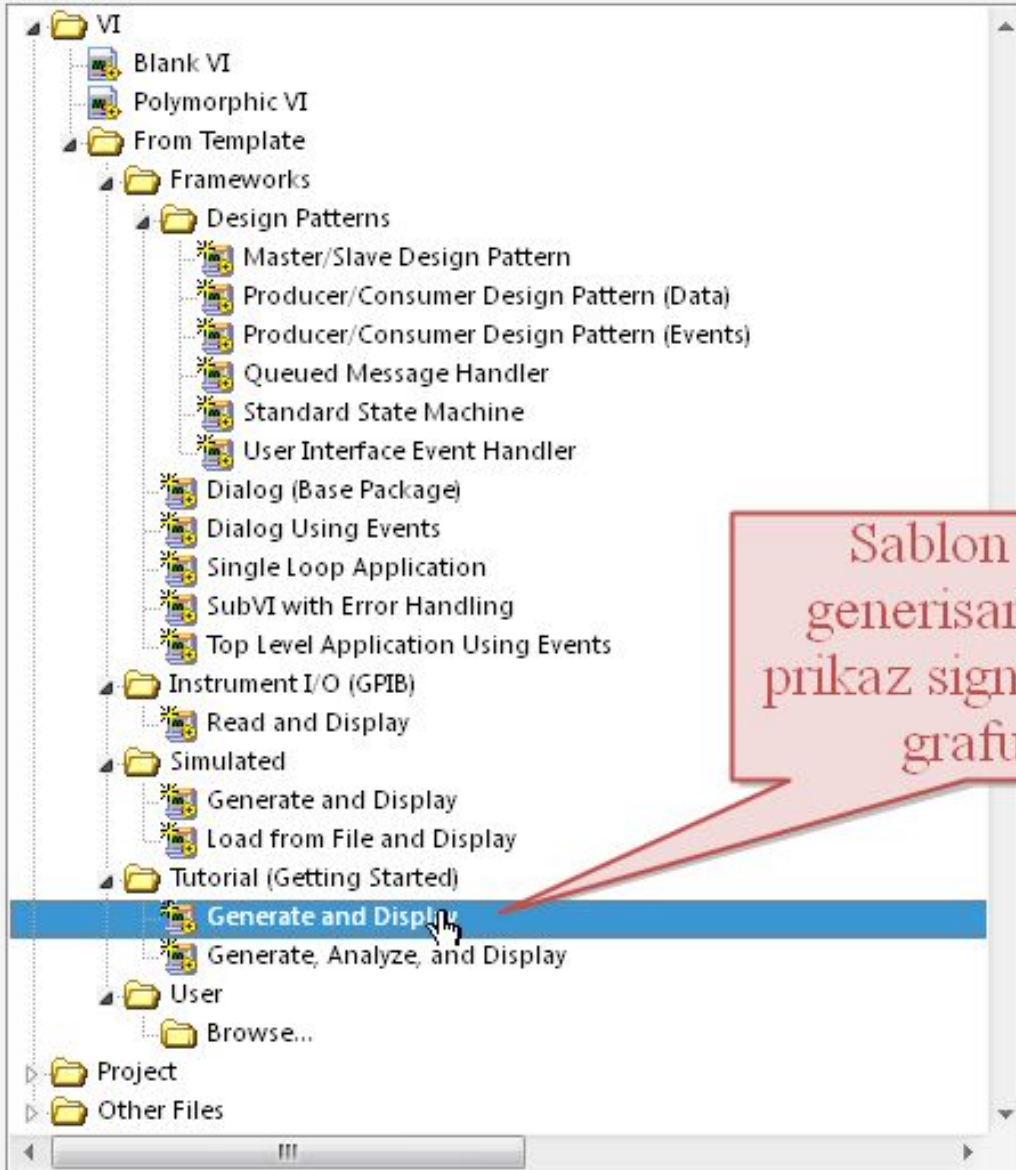
1. Generisanja signala
2. Prikaza generisanog signala na grafu panela

Za kreiranje virtuelnog instrumenta pomoću šablona u okviru “Getting started” ekrana selektovati **VI from Template** u okviru **New** grupe.

Nakon ove akcije pojavljuje se ekran kao na slici, gde je ponuđen veći broj šablona koji su razvrstani u razne kategorije.

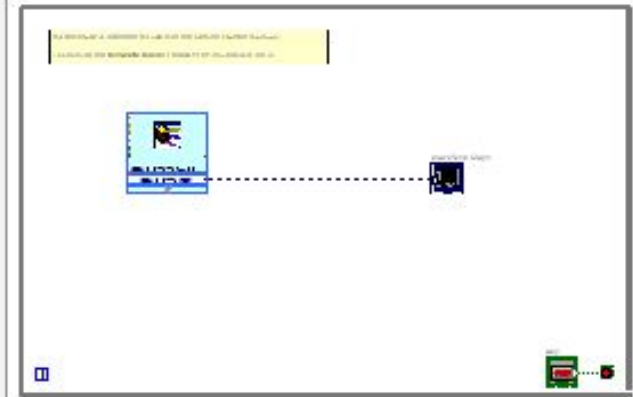
Iz kategorije **Tutorial (Getting Started)** odabrati šablon **Generate and Display**

Create New



Sablon za generisanje i prikaz signala na grafu

Description



Use this template to generate a signal and display the generated data in a graph. You can use this template with the exercises in the Getting Started with LabVIEW

W enables automatic error handling by Is you create from this template.

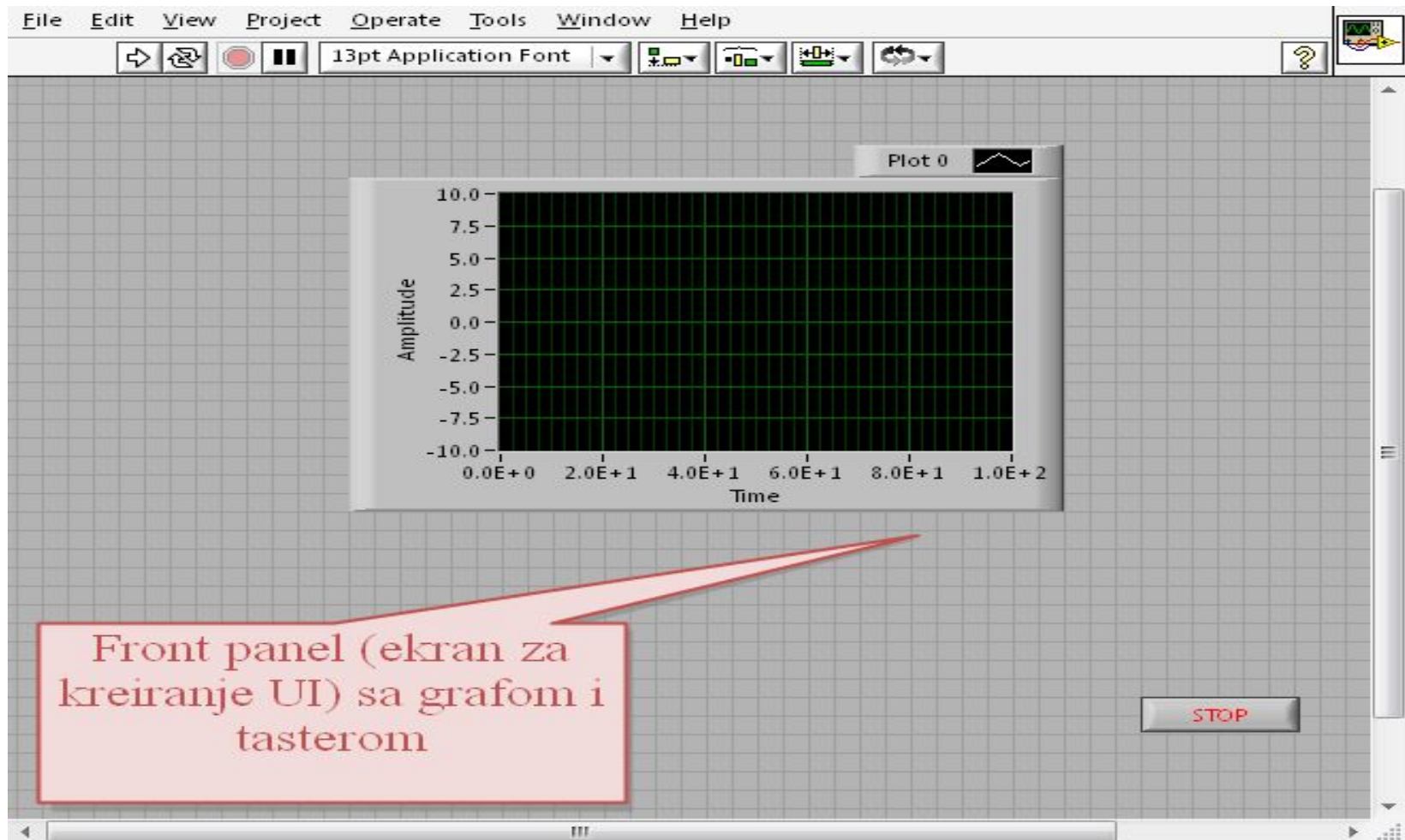
Add to project

OK

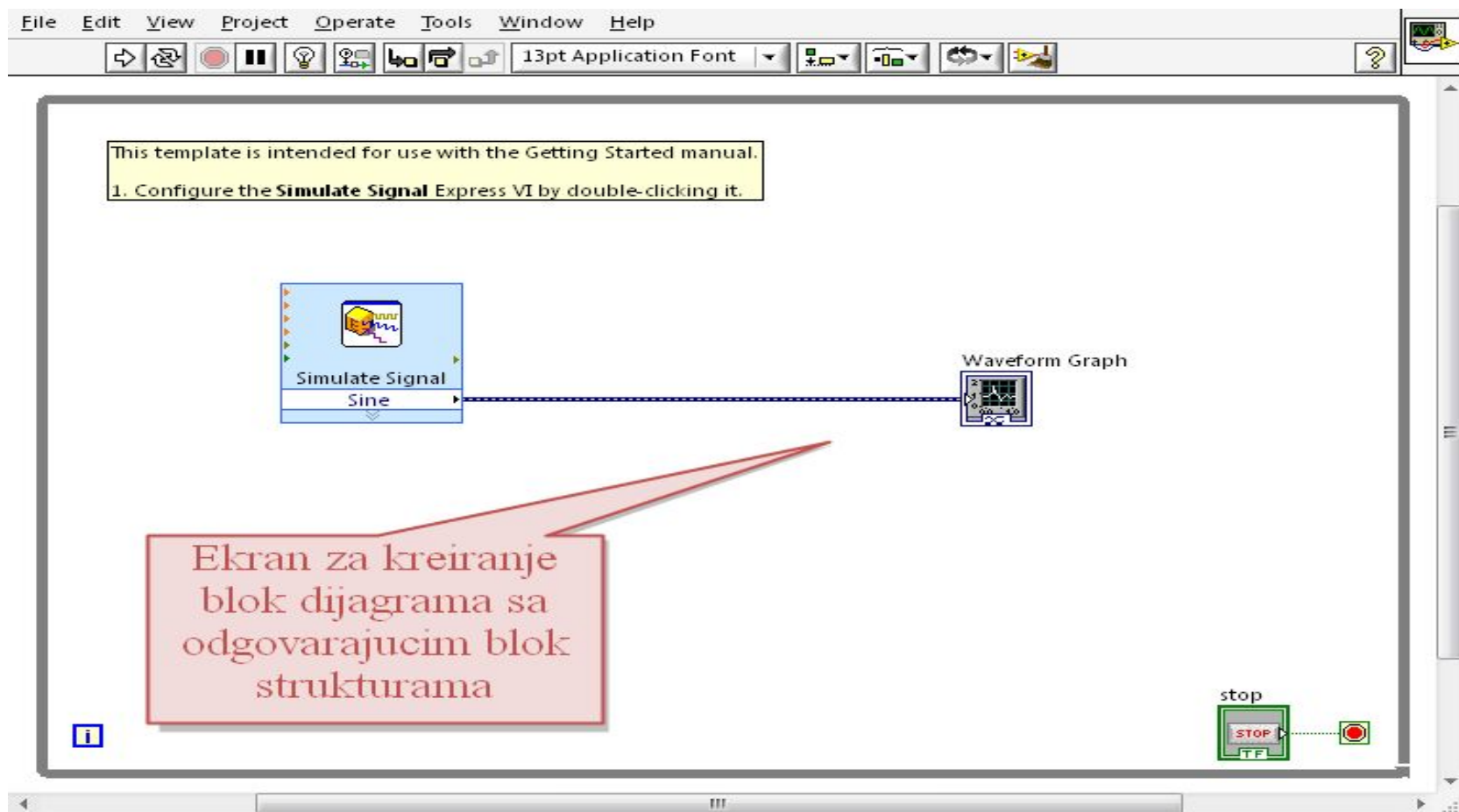
Cancel

Help

Nakon odabira šablona pojavljuje se Front Panel sa odgovarajućim kontrolama

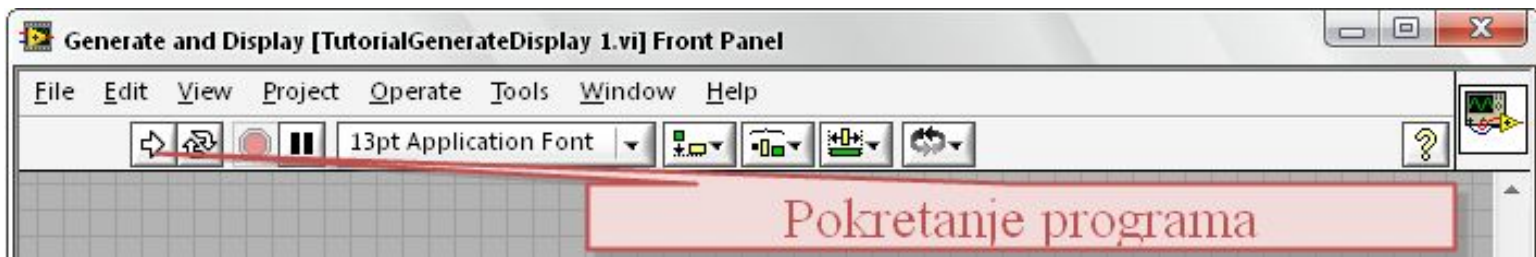


I ekran za kreiranje blok dijagrama virtuelnog instrumenta sa blokovima za generisanje i prikaz signala koji su u odgovarajućoj sprezi sa kontrolama na Front Panelu

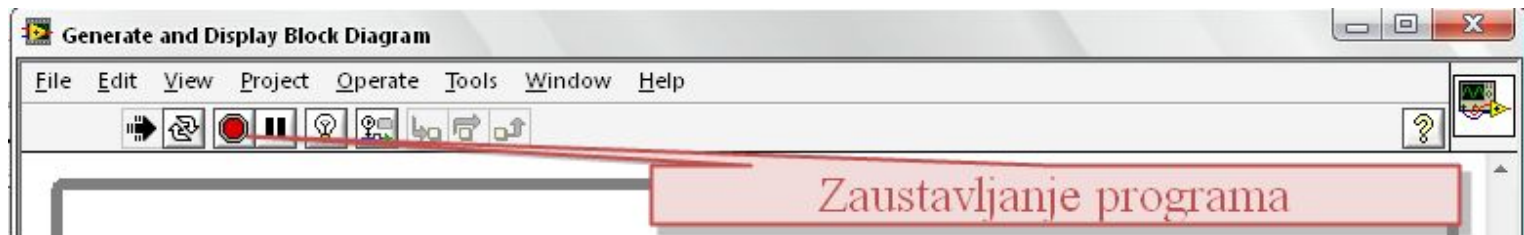
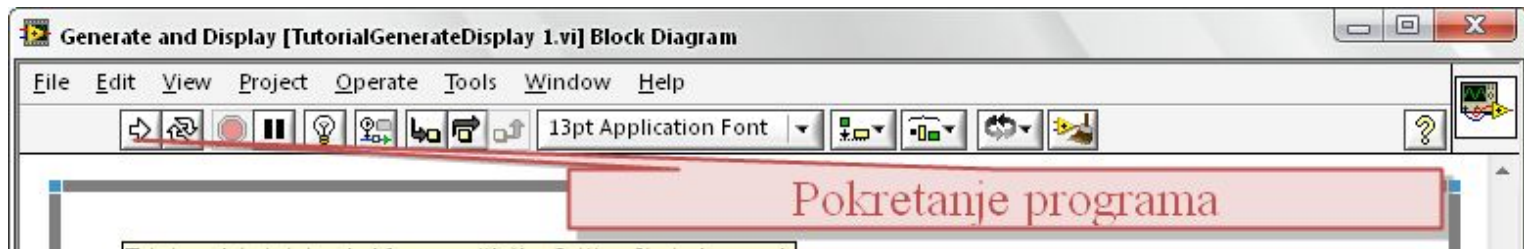


Navigacija sa jednog na drugi ekran se može postići klikom na odgovarajući u okviru Taskbar-a ili selektovanje opcije **Show Front Panel** odnosno **Show Block Diagram** u padajućem meniju **Window** u nekom od ova dva ekrana.

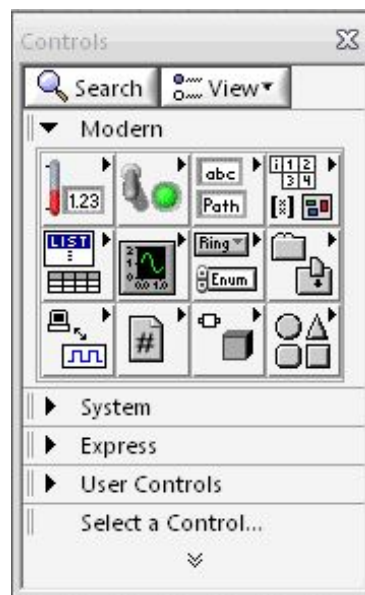
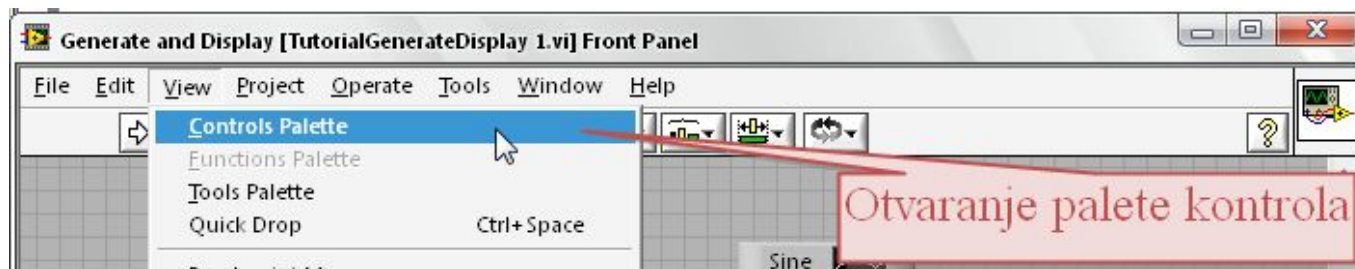
U okviru Front Panela je moguće pokrenuti izvršavanje LabView programa gde bi se prikazao njegov rad sa trenutnim kontrolama koje su iskorištene.



Program je moguće pokrenuti i u okviru Block Diagram editora na sličan način

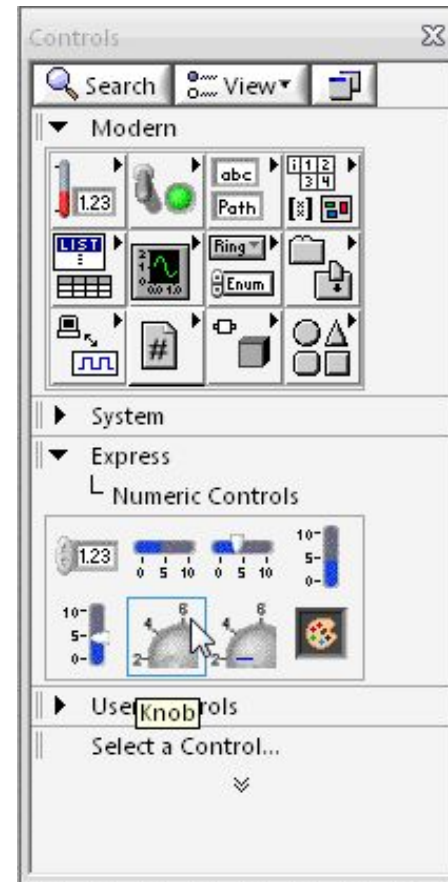
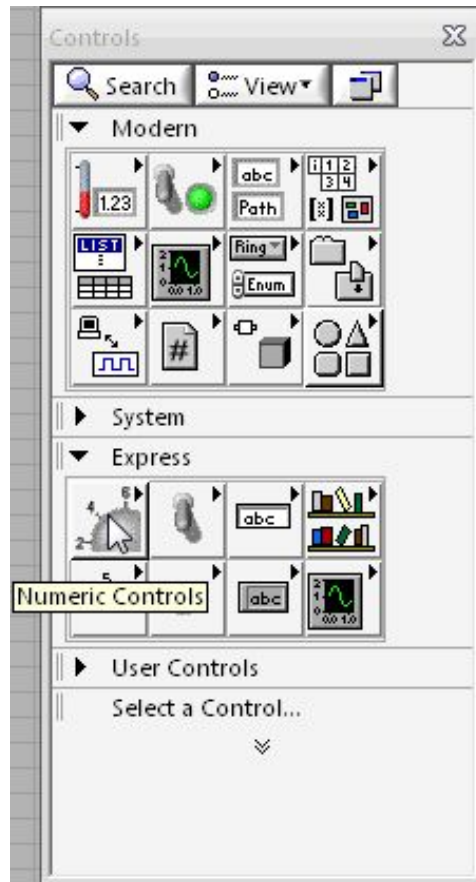


Dodavanje novih kontrola u okviru Front Panel-a se postiže selektovanjem odgovarajuće iz palete kontrola



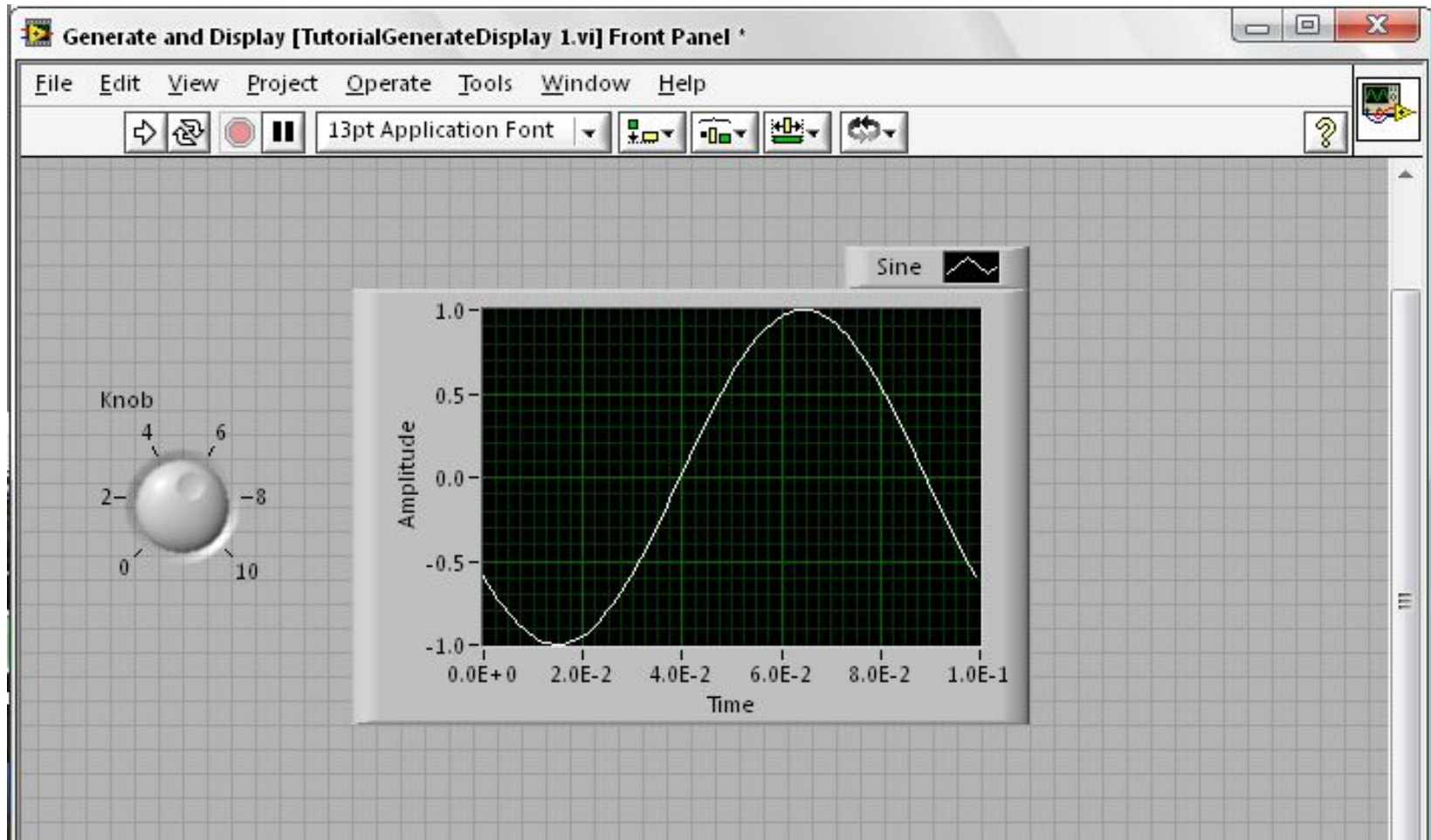
Kontrole su grupisane po kategorijama gde se selektovanjem odgovarajuće kategorije prikazuju kontrole koje joj pripadaju.

U okviru **Express** kategorije selektovati **Numeric Controls**, a zatim odabrati **Knob** kontrolu i postaviti je na površinu panela (pored grafa npr).



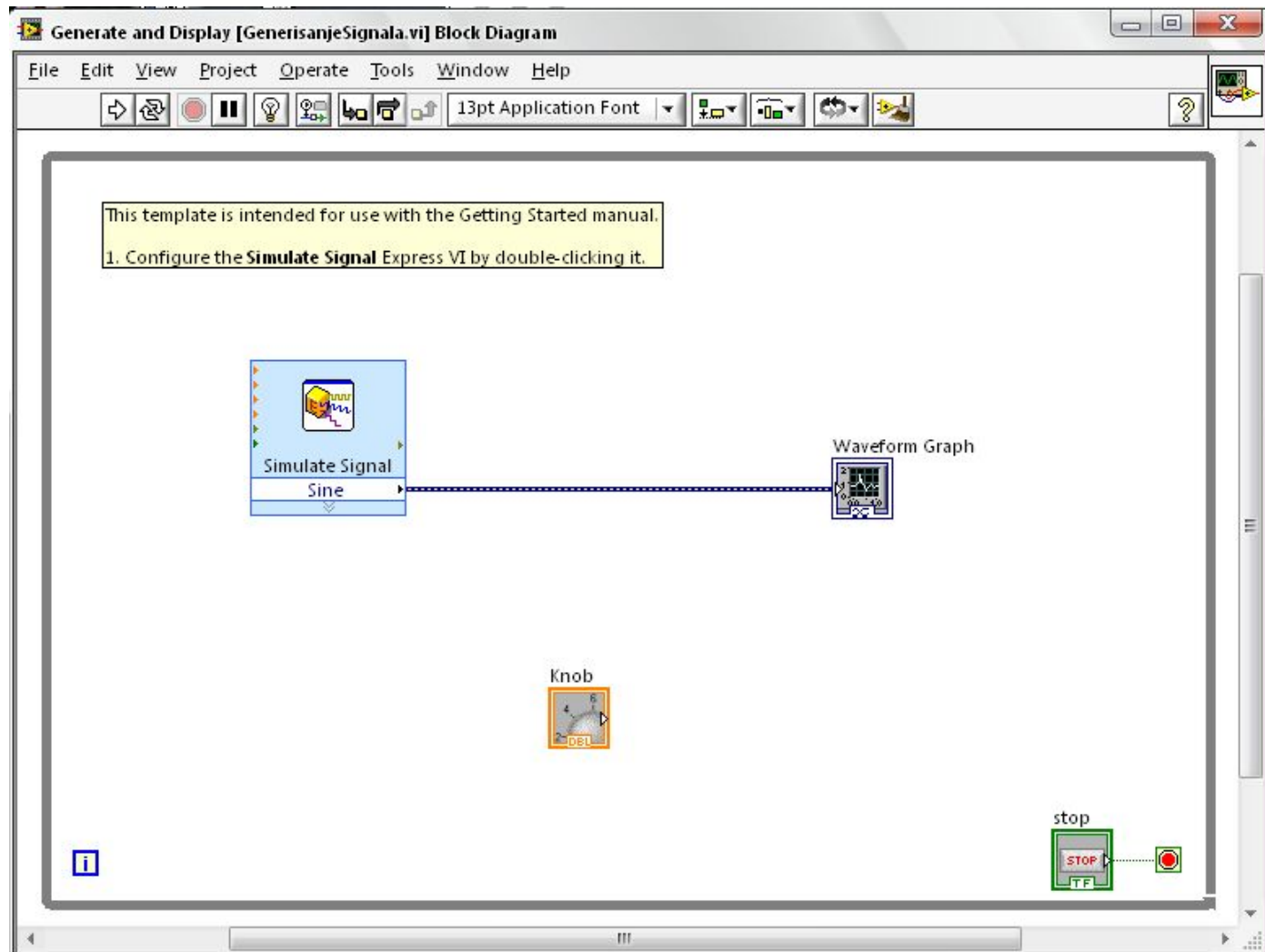
Knob će biti iskorišten za promenu amplitude generisanog signala.

Snimiti ceo projekat kao **GenerisanjeSignala.vi** npr (**File-Save As..**)



Na radnoj površini Block Diagram editora nakon svih akcija nalaze se blokovi kao na slici.

Neki blokovi su povezani, neki ne, ali svi sadrže informacije (atribute) kontrole kojima su pridruženi. Izmenom odgovarajućih atributa utiče se na ponašanje kontrole.



Promena tipa generisanog signala se može ostvariti izmenom atributa kontrole **Simulate Signal**. Desnim klikom na simbol ove kontrole se selektuje **properties**, gde se nakon toga pojavljuje ekran za izmenu atributa kontrole

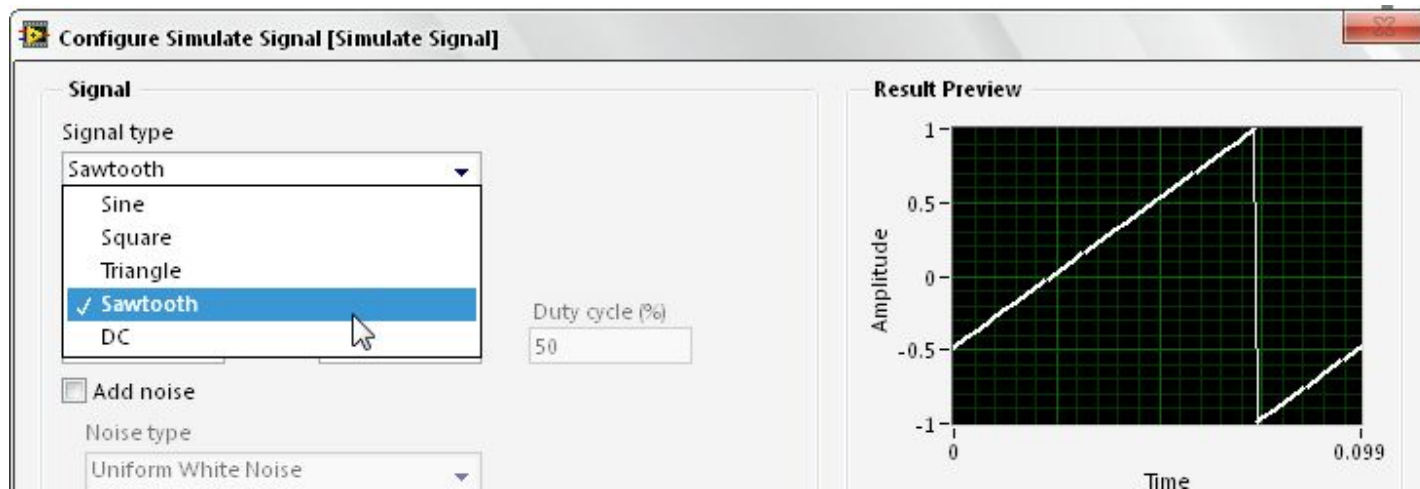
The image shows a software interface for configuring a signal. A context menu is open over the 'Simulate Signal' icon, with 'Properties' selected. The 'Configure Simulate Signal [Simulate Signal]' dialog box is displayed, showing the following settings:

- Signal**
 - Signal type: Sine
 - Frequency (Hz): 10.1
 - Phase (deg): 0
 - Amplitude: 1
 - Offset: 0
 - Duty cycle (%): 50
 - Add noise
 - Noise type: Uniform White Noise
 - Noise amplitude: 0.6
 - Seed number: -1
 - Trials: 1
- Timing**
 - Samples per second (Hz): 1000
 - Number of samples: 100
 - Simulate acquisition timing
 - Run as fast as possible
 - Automatic
 - Integer number of cycles
 - Actual number of samples: 100
 - Actual frequency: 10.1
- Time Stamps**
 - Relative to start of measurement
 - Absolute (date and time)
- Reset Signal**
 - Reset phase, seed, and time stamps
 - Use continuous generation
- Signal Name**
 - Use signal type name
 - Signal name: Sine

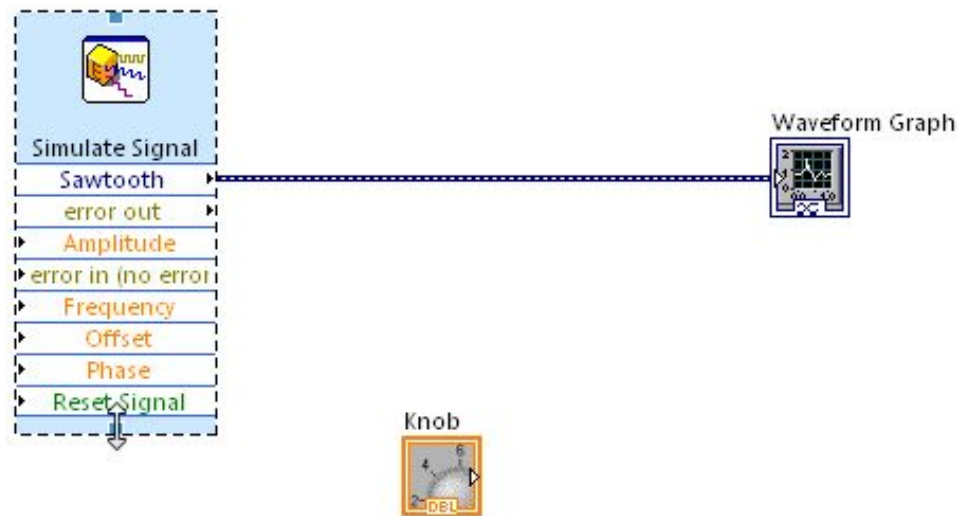
The 'Result Preview' graph shows a sine wave with an amplitude of 1 and a time range from 0 to 0.099. The 'OK', 'Cancel', and 'Help' buttons are visible at the bottom of the dialog.

U okviru liste **Signal Type** može se promeniti tip generisanog signala, npr. selektuje se **Sawtooth** što će rezultirati generisanjem testerastog signala.

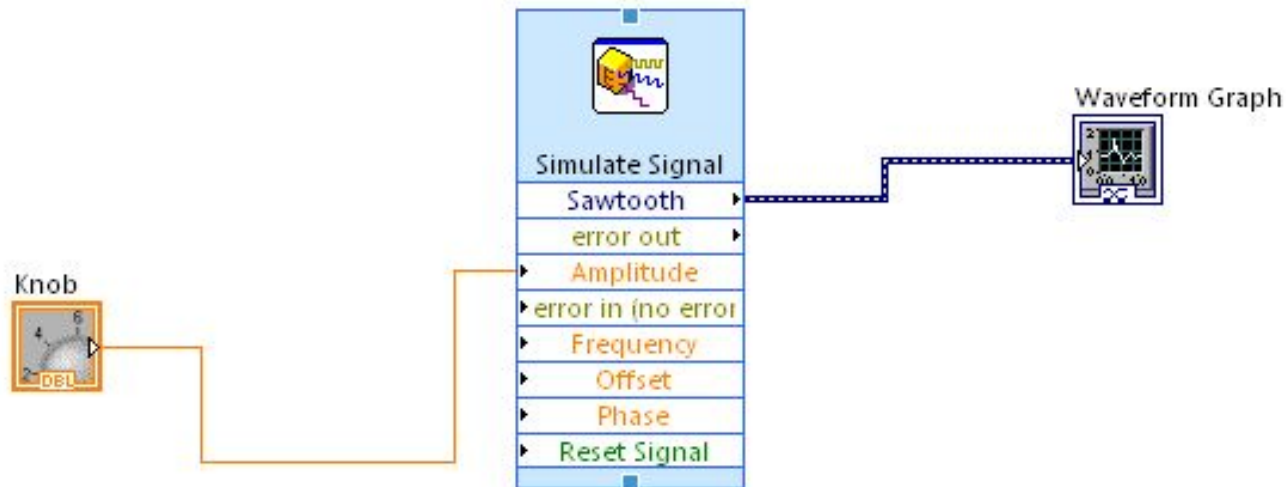
Na isti način se mogu promeniti i ostali atributi ove kontrole



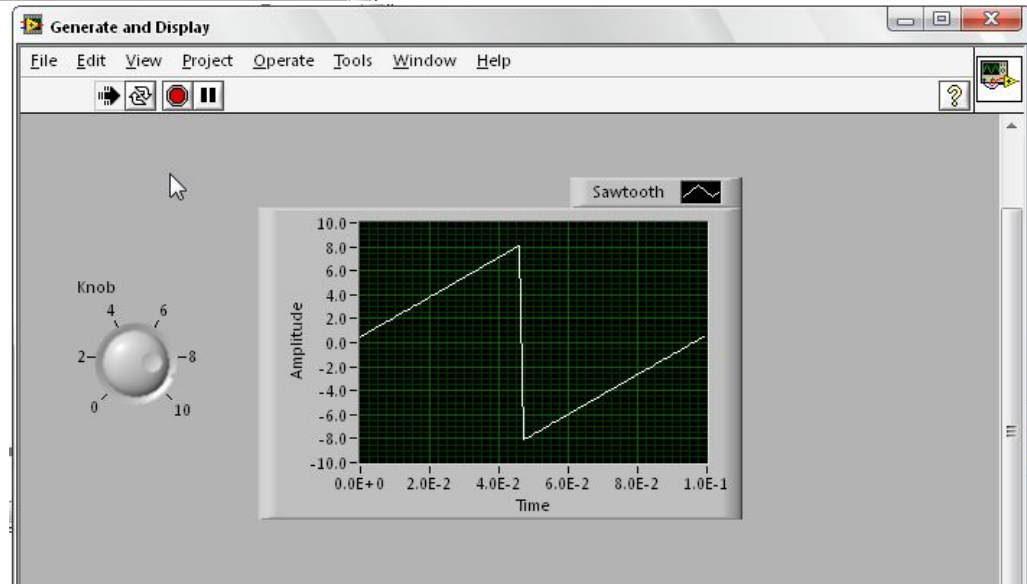
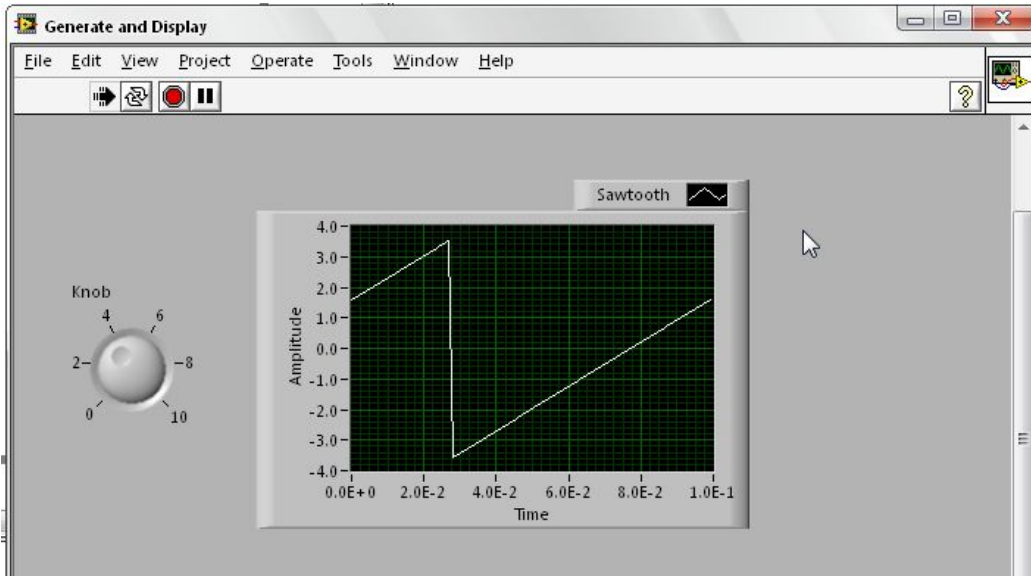
Blok za simuliranje signala sadrži atribute koji se mogu kontrolisati drugim kontrolama. Za prikaz ovih atributa je potrebno proširiti blok **Simulate Signal** (kao na slici).



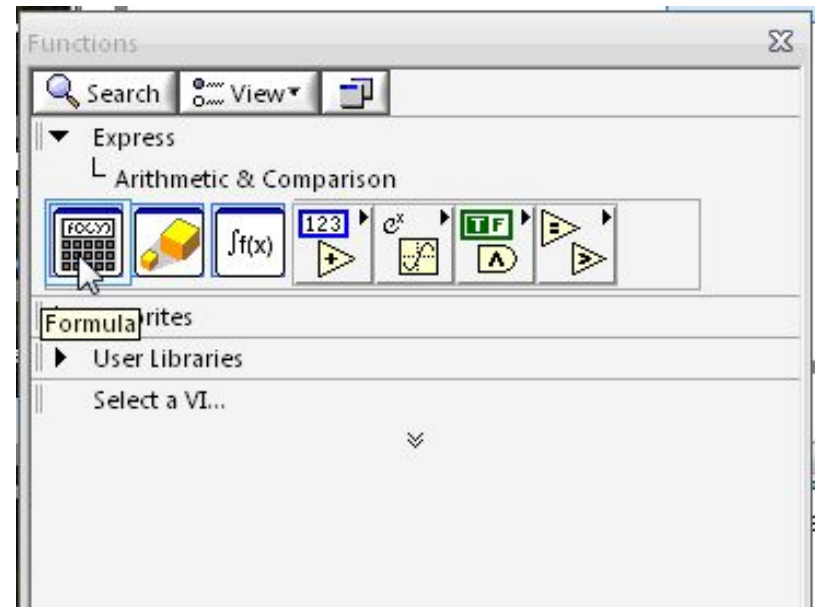
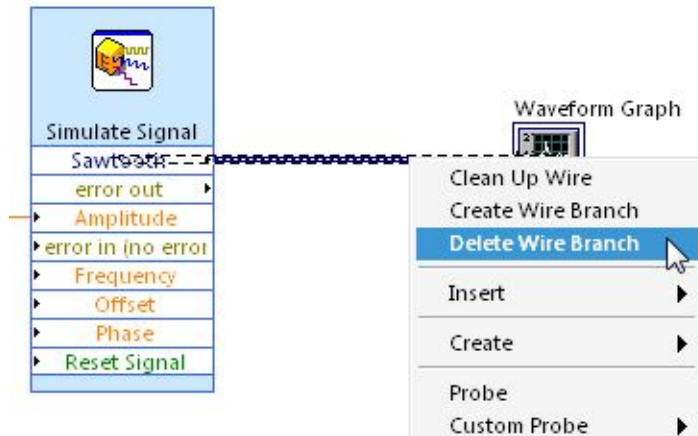
Povezivanje **Knob** kontrole sa **Simulate Signal Amplitude** atributom se vrši njihovim “ožičavanjem”, poziciniranjem kursora kod odgovarajućih izvoda (portova) na oba bloka i uspostavljanjem veze.



Startovanjem programa sa **Run** se može videti uticaj **Knob** kontrole na amplitudu generisanog signala

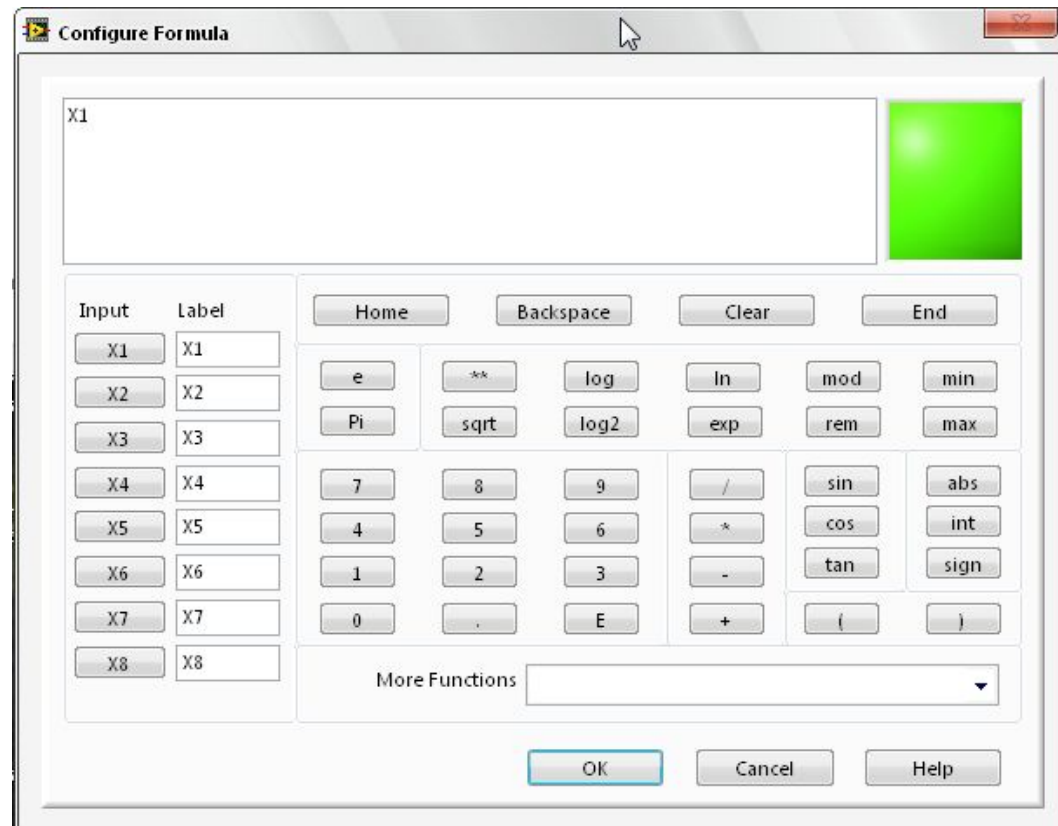
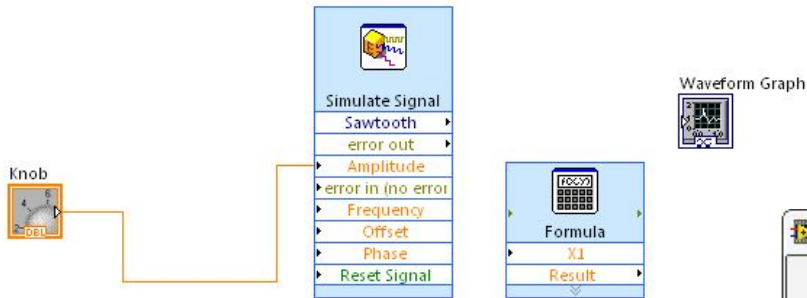


Selektovati vezu između **Graph** kontrole i **Simulate signal** i obrisati je. Otvoriti paletu funkcija (View-Function Palette)

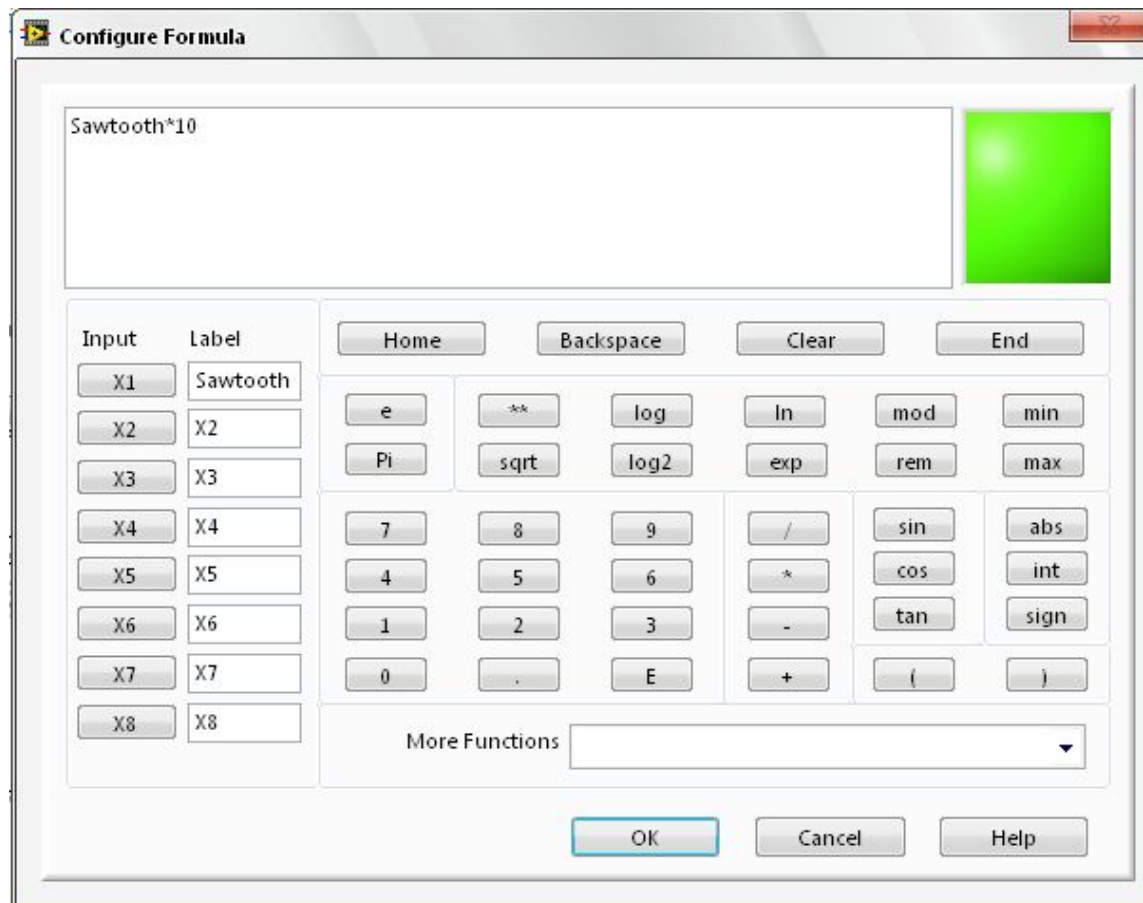


U okviru **Express** kategorije selektovati **Arithmetic & Comparison** a zatim **Formula** kontrolu postaviti između **Simulate signal** i **Graph** kontrole

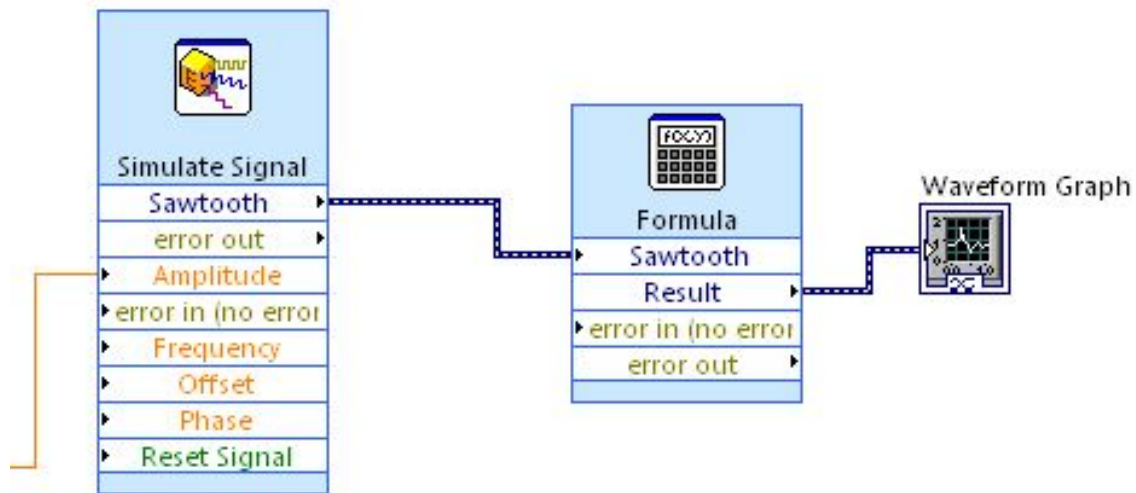
Configure Formula prozor se automatski otvara prilikom postavljanja **Formula** kontrole na radnu površinu



Sa leve strane ovog prozora se nalaze raspoloživi ulazni signali nad kojima se vrši neka transformacija. U prvo polje (Label) upisati **Sawtooth** (signal iz generatora) a potom njegovu vrednost izmnožiti sa npr 10 (skaliranje)

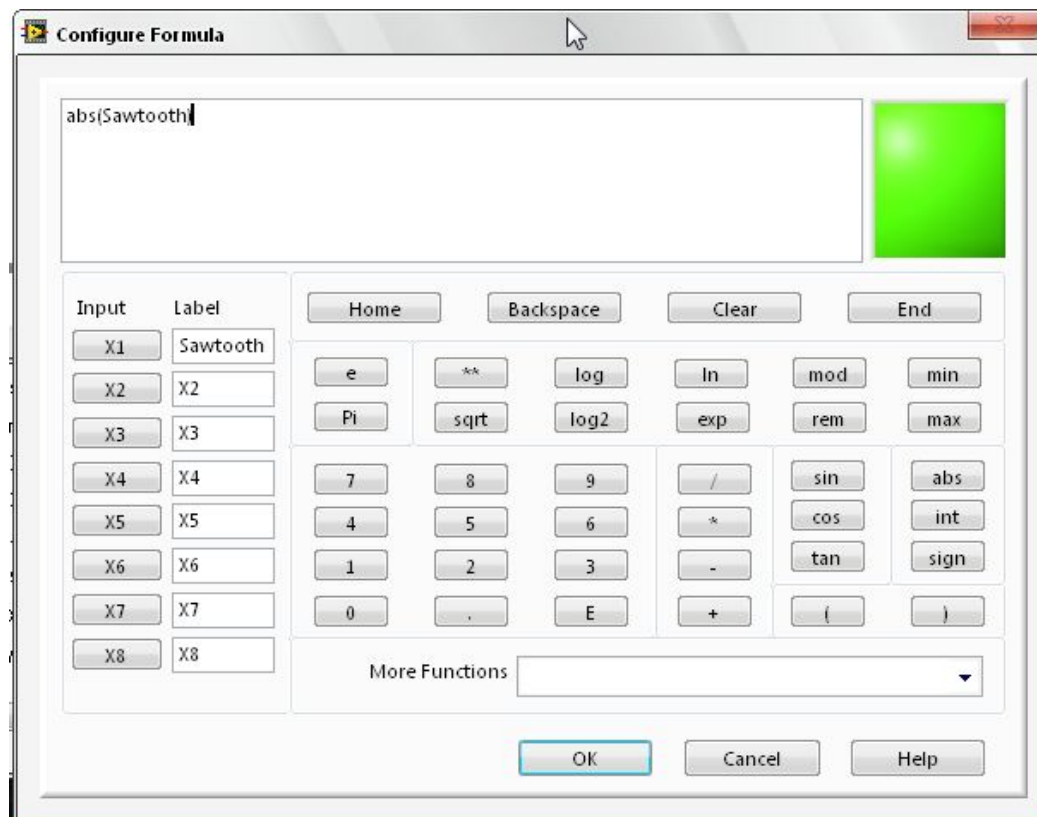
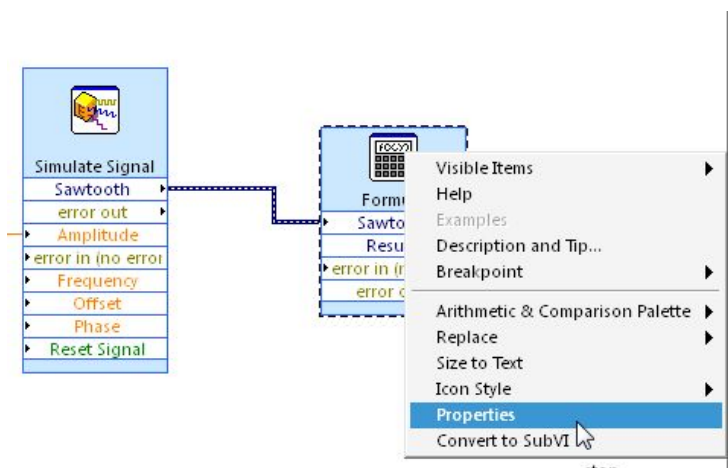


- Povezati blokove kao na slici
- Na ovaj način je izvršeno skaliranje signala iz generatora
- Pokrenuti program i proveriti njegov rad

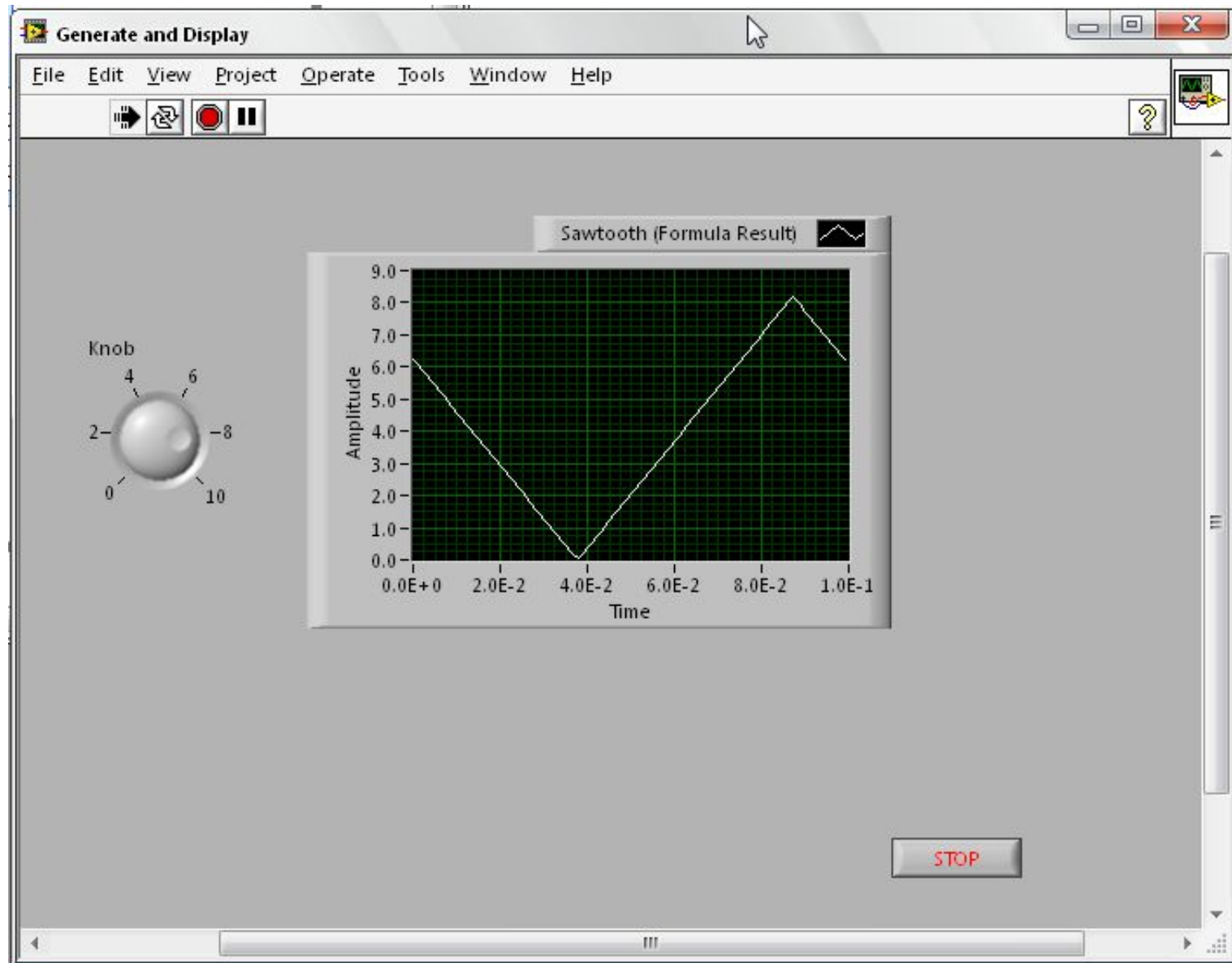


Izmena modifikacije signala iz generatora se može obaviti selektovanjem **Formula** bloka i odabirom **properties** nakon čega se ponovo pojavljuje konfiguracioni dijalog sa opcijama.

Selektujemo funkciju **abs**, zatim signal **Sawtooth** i zatvorimo zagradu sa)



- Rezultat ove akcije je “ispravljeni” signal (trouglasti)



Druga vežba

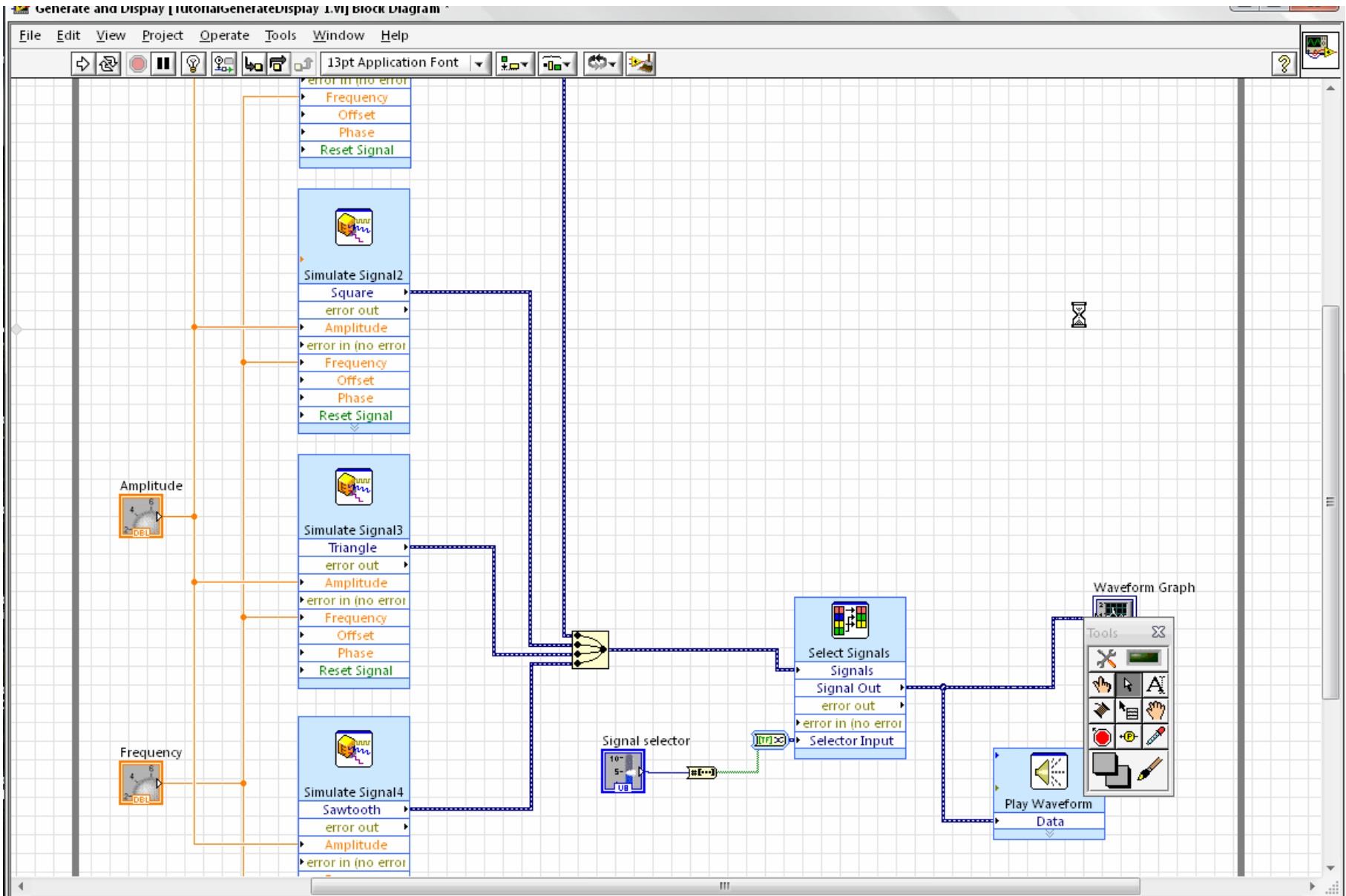
Primer 1

Modifikovati prethodno izgrađeni virtuelni instrument za generisanje signala promenjive učestanosti i amplitude tako da se preko front panela na jednostavan način može izabrati različiti tip generisanog signala (sinusni, trougaoni, pravougaoni, testerasti)

1. U okviru Block diagram editora selektovati kontrolu **select signals** i postaviti je na radnu površinu
2. Podesiti da kontrola kao ulaz prima četiri signala (postaviti prethodno merge signals sa četiri ulaza)
3. Iskopirati Simulate signal kontrolu u četiri instance i povezati ulaze za amplitudu i učestanost svih instanci na iste promenjive kontrole
4. U okviru front panela kreirati **vertical point slide** i formirati enum strukturu sa vrednostima prema bitu maske

5. Za tip podataka slider-a postaviti **UBYTE**
6. U block diagram-u dodati konverziju **byte->array** of bytes
7. Povezati izlaz konverzije sa **Selector input** portom, **Select signals** bloka

Sadržaj block diagram editora nakon svih koraka



Treća vežba

Primer 2

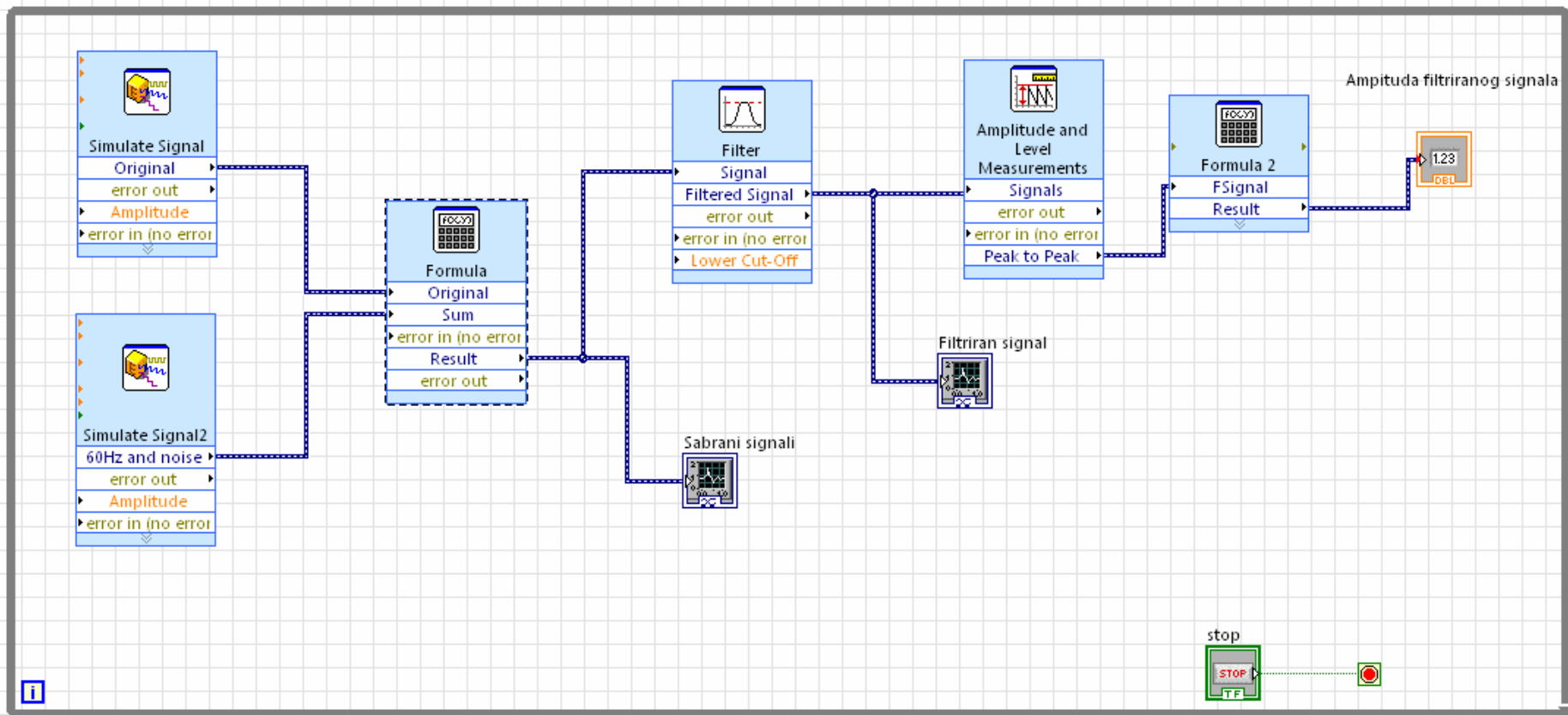
Kreirati virtuelni instrument koji generiše signal, vrši filtriranje tog signala, određuje da li vrednost signala prelazi neku zadatu vrednost i snima vrednosti signala u vremenu

1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Kreirati dva generatora signala (**simulate signal**) i postaviti ih u okviru **while** petlje
4. Podesiti jedan generator da generiše sinusoidu učestanosti 60Hz, amplitude 0.1

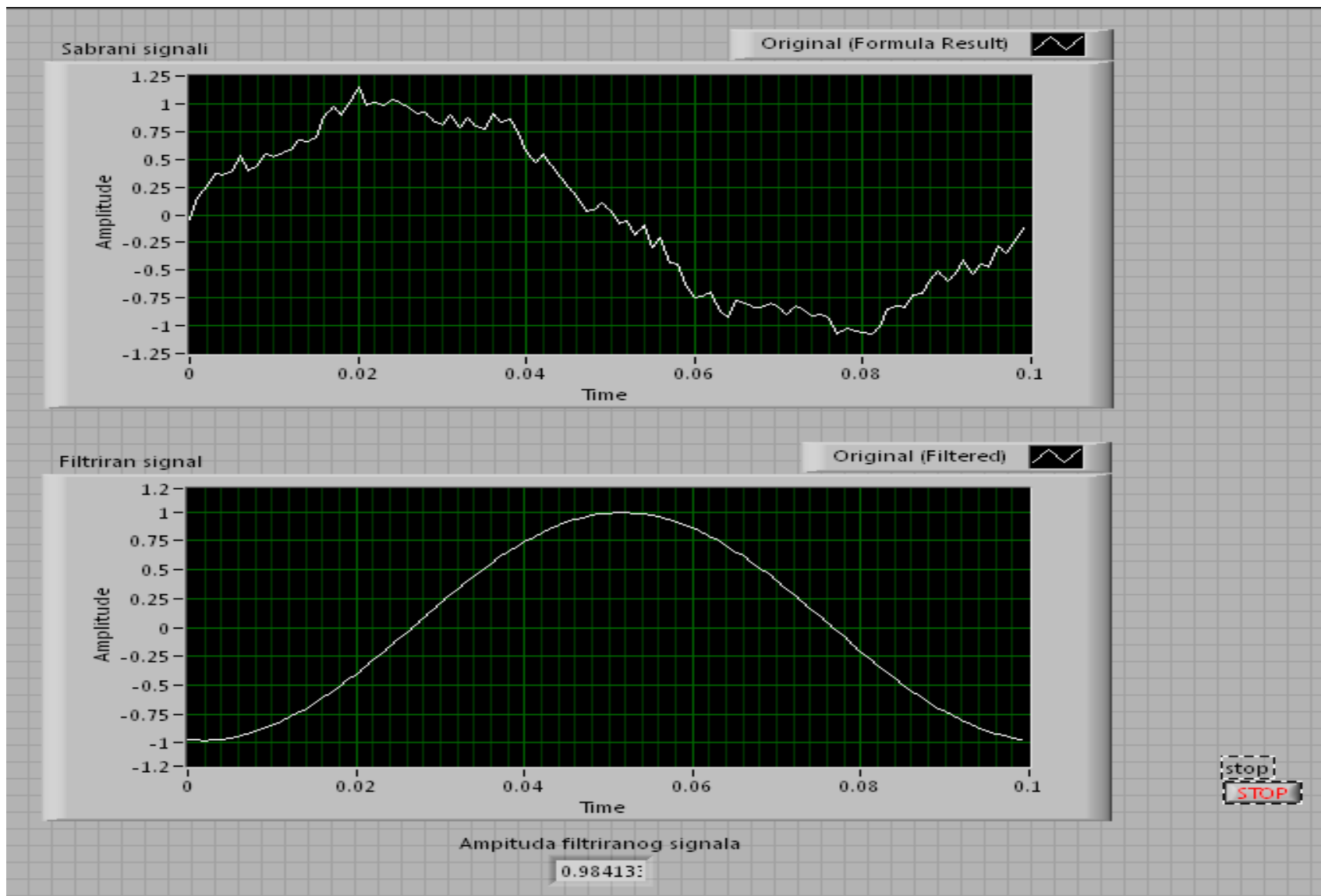
5. Dodati beli šum amplitude 0.1 prethodno podešenom generatoru signala, promeniti naziv generatora signala
6. Kreirati drugi generator signala učestanosti 10Hz, amplitude signala 1, sa sinusnim tipom signala
7. Sabrati signale koje generišu ova dva generatora signala koristeći se **formula** blokom
8. Dodati graf na kojem će se prikazati vremenski oblik sabranog signala

9. Dodati blok **filtra** za filtriranje sabranog signala
(**Sygnal analysis**)
10. Filtar konfigurirati tako da radi kao niskopropusnik
npr na 25Hz
11. Povezati filter na izlaz **formula** bloka
12. Dodati blok za merenje amplitude i drugih nivoa
signala - **Amplitude and level measurements**
13. Dodati drugi graf za snimanje filtriranog signala

14. Dodati indikator amplitude signala



Izgled interfejsa

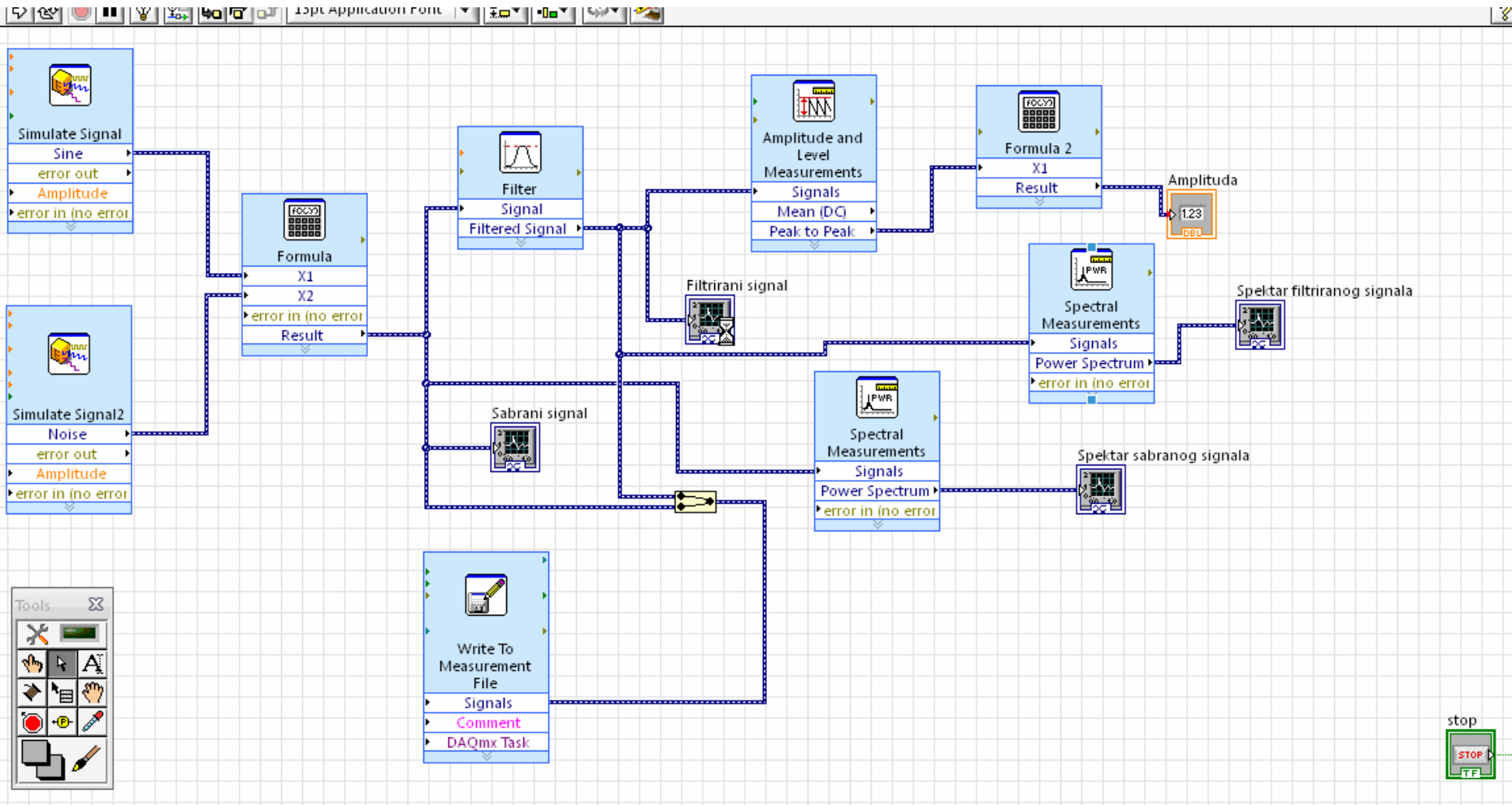


14. Dodati blok **Write to measurement file** za zapis u fajl (**Sygnal output**)

15. Probat i zapis raznih signala i uočiti način na koji kontrola funkcioniše

16. Dodati blok za spektralnu analizu signala i poseban grafik na kojem će se prikazati spektar smetnje i filtriranog signala (objasniti dobijene vrednosti)

Izgled blok dijagrama nakon dodavanja blokova za spektralnu analizu



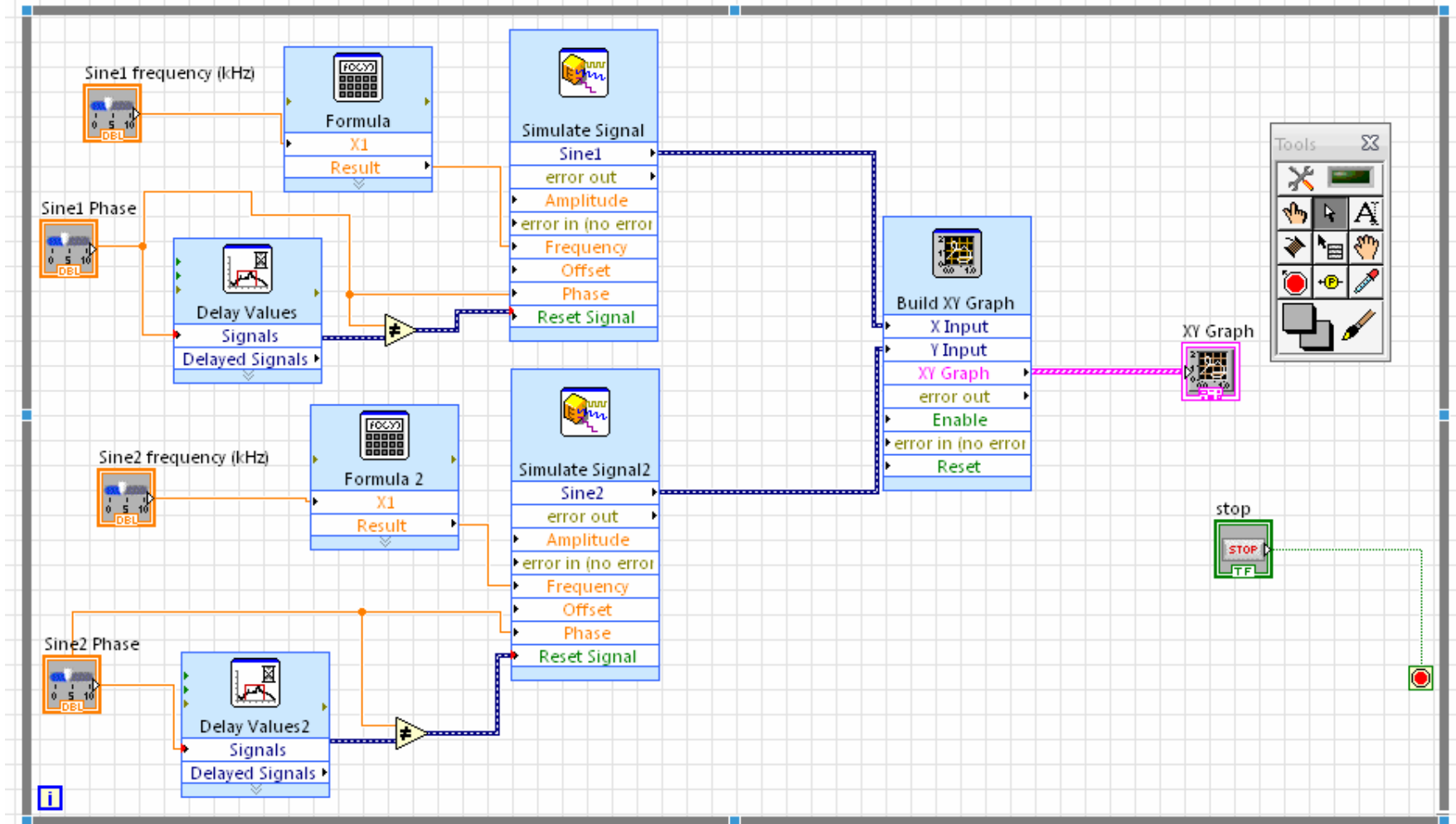
Primer 3

Kreirati virtuelni instrument koji generiše dva sinusna signala i iscrtava njihovu međuzavisnost (Lisazuovu figuru).

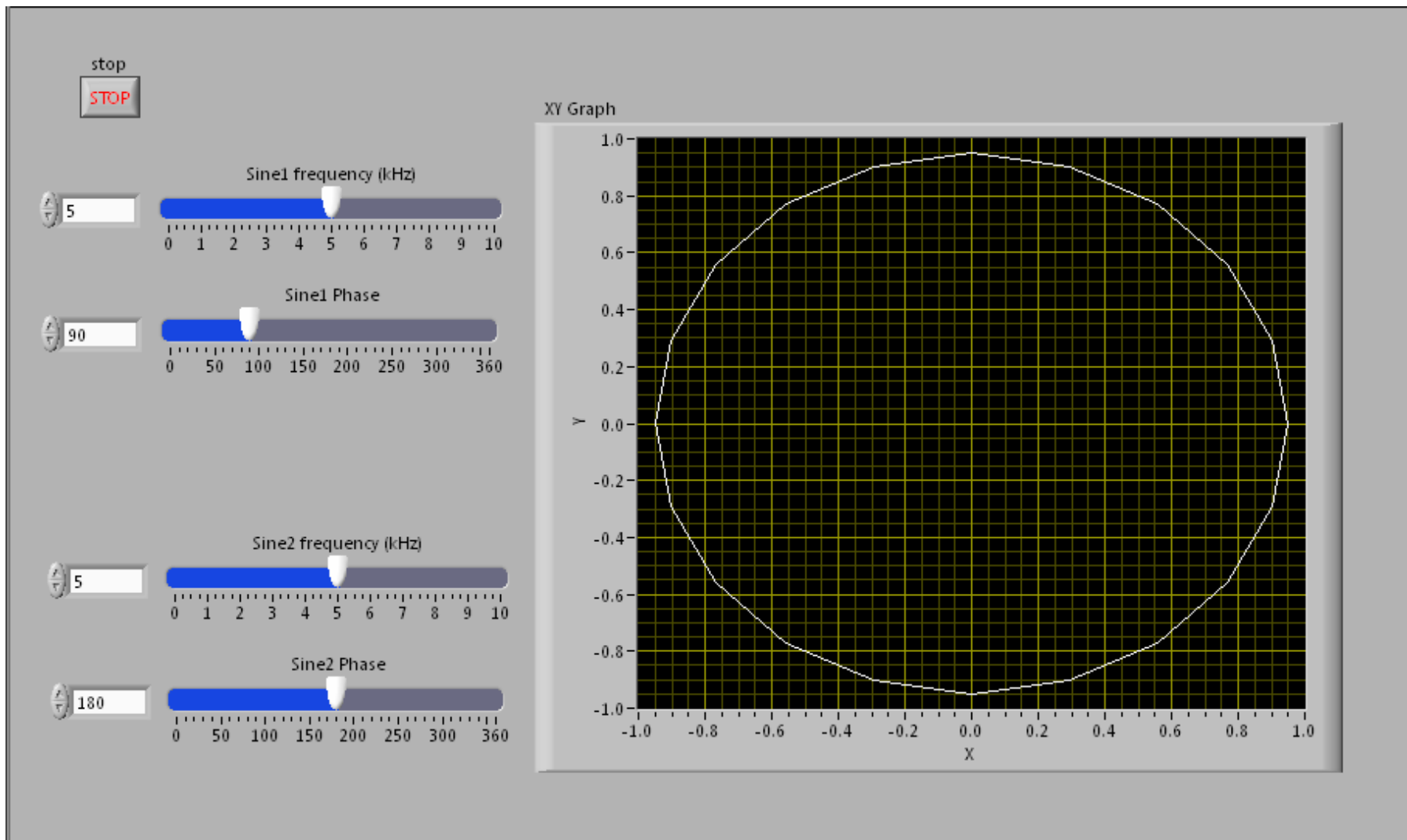
1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Kreirati dva generatora signala (**simulate signal**) i postaviti ih u okviru **while** petlje
4. Podesiti generatore tako da imaju istu amplitudu 0.95

5. Na front panelu kreirati kontrole za promenu učestanosti i faze sinusnih generatora
6. Kreirati graf za iscrtavanje Lisažuove figure
7. Uvesti Reset dugme za resetovanje generatora zbog promene faze
8. Automatizovati proces bez reset dugmeta

Izgled blok dijagrama



Izgled front panela



Četvrta vežba

Primer 4

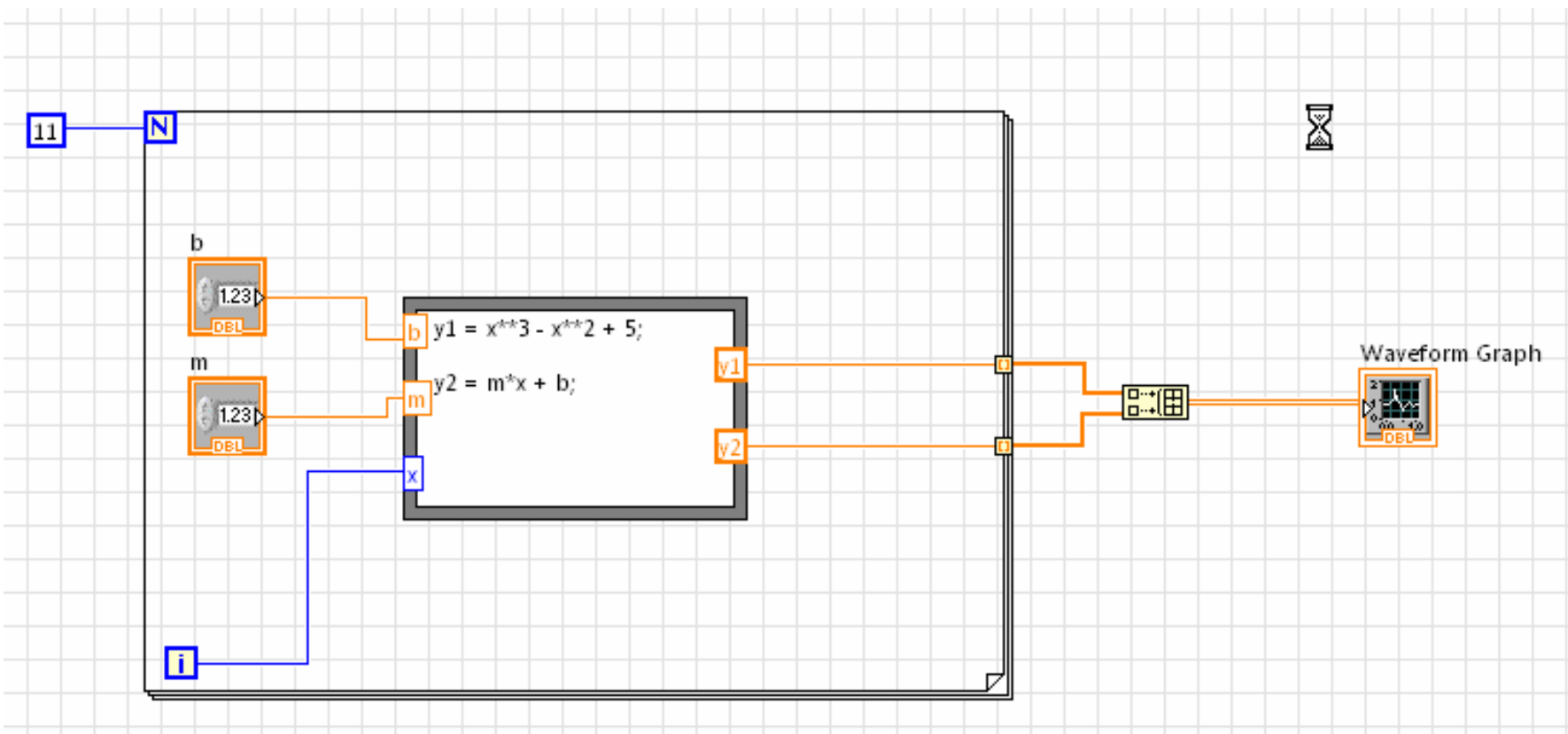
Kreirati virtuelni instrument koji generiše dve funkcije na osnovu **Formula Node** bloka i grafički prikazuje vrednosti ovih funkcija u tačno jedanaest tačaka u opsegu 0-10

1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **for** petlju (programming-structures) u okviru block diagram editora i definisati tačno deset izvršavanja
3. Kreirati **formula node** blok (structures) i definisati dve funkcije $y1 = x^{**}3 - x^{**}2 + 5$; $y2 = m*x + b$;
4. Dodati numeričke kontrole u okviru UI za podešavanje m i b parametara

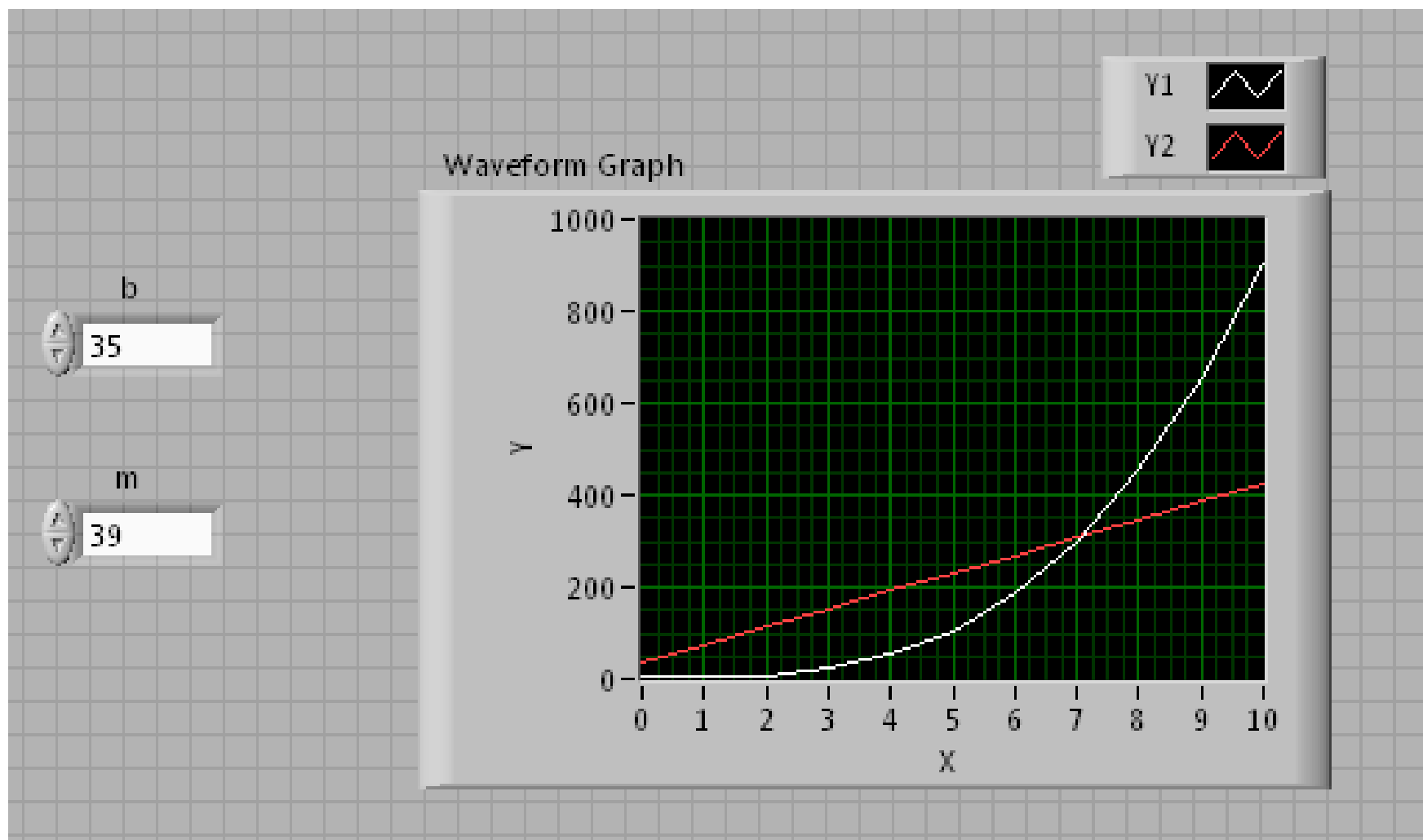
5. Na front panelu kreirati **Waveform Graph** i povezati ga sa izlaznim portovima for strukture

6. Dodati Build Array blok i povezati ga između **Waveform Graph** i nodova **for** strukture

Izgled blok dijagrama



Izgled front panela



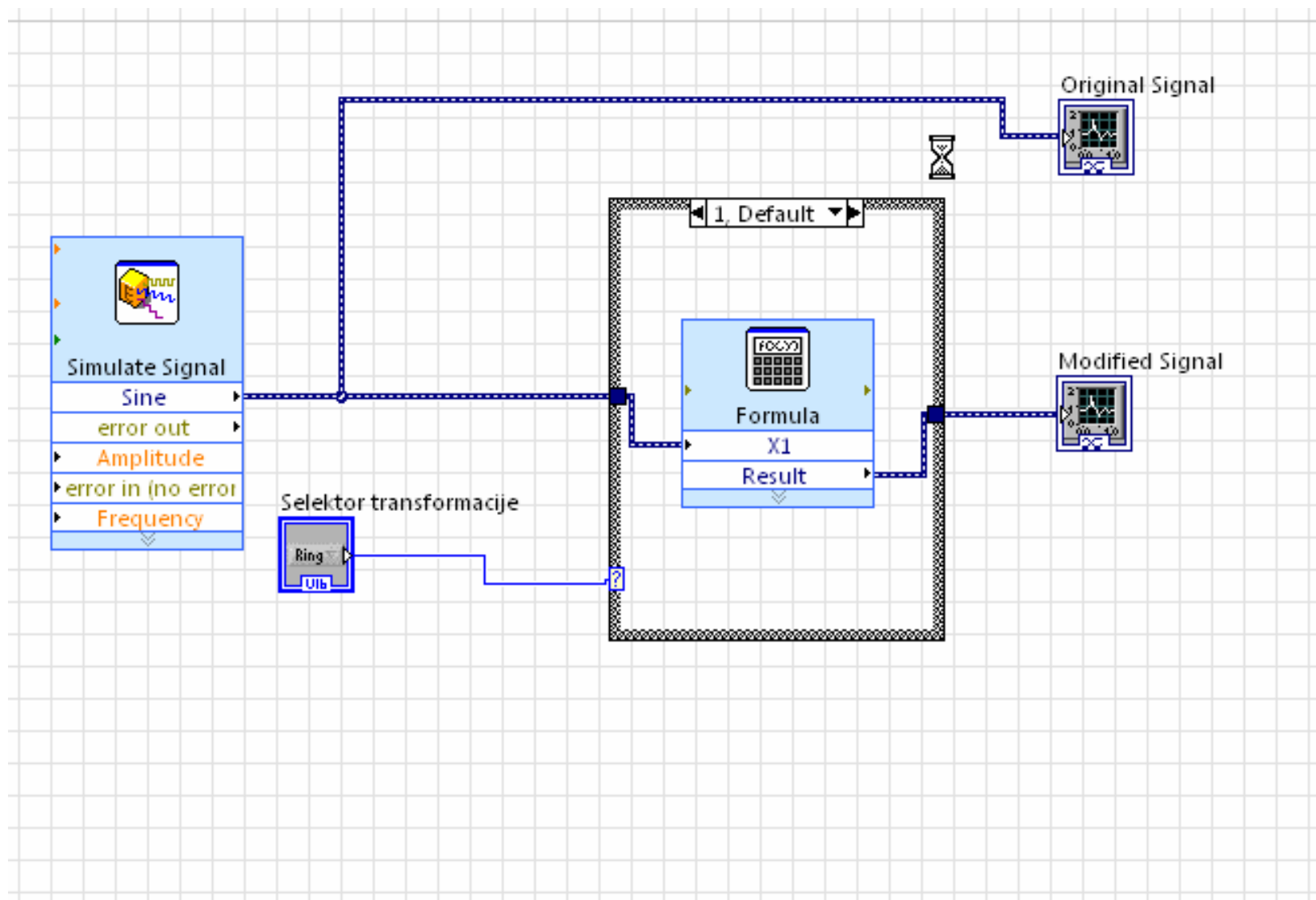
Primer 5

Kreirati virtuelni instrument koji generiše sinusoidalni signal i na izlaz prosleđuje modifikovan signal u zavisnosti od položaja selektora (modifikacije su kvadrat, kub i ispravljeni signal)

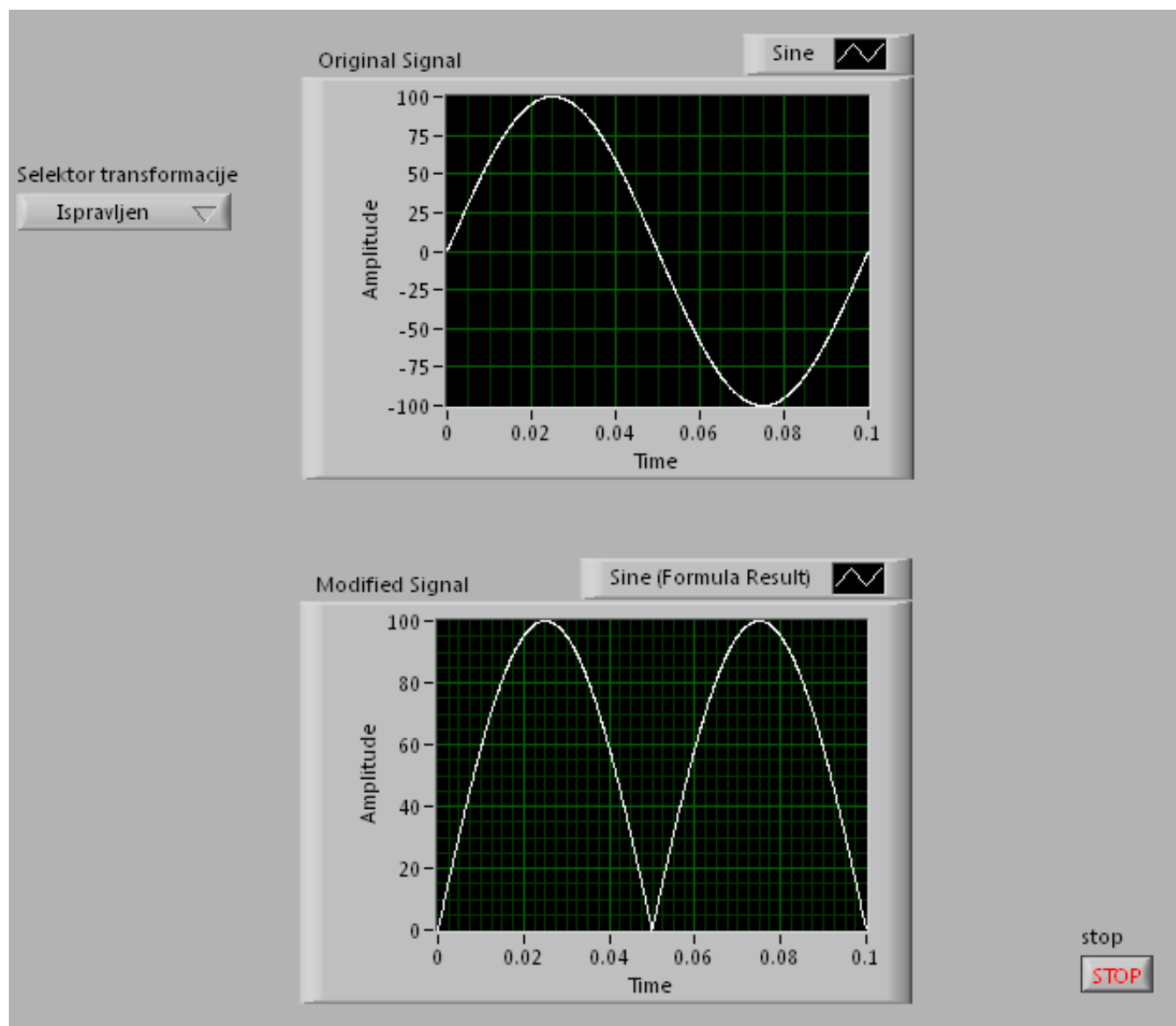
1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Kreirati generator signala (**simulate signal**) i postaviti ga u okviru **while** petlje
4. Kreirati kontrole za promenu amplitude i učestanosti

6. Dodati **case** strukturu koja će voditi računa o generisanoj funkciji na izlazu
7. Dodati grafove na kojima će se prikazati vremenski oblik ulaznog i modifikovanog signala
8. Dodati selektor u okviru UI za odabir modifikacije signala

Izgled blok dijagrama



Izgled front panela



Peta vežba

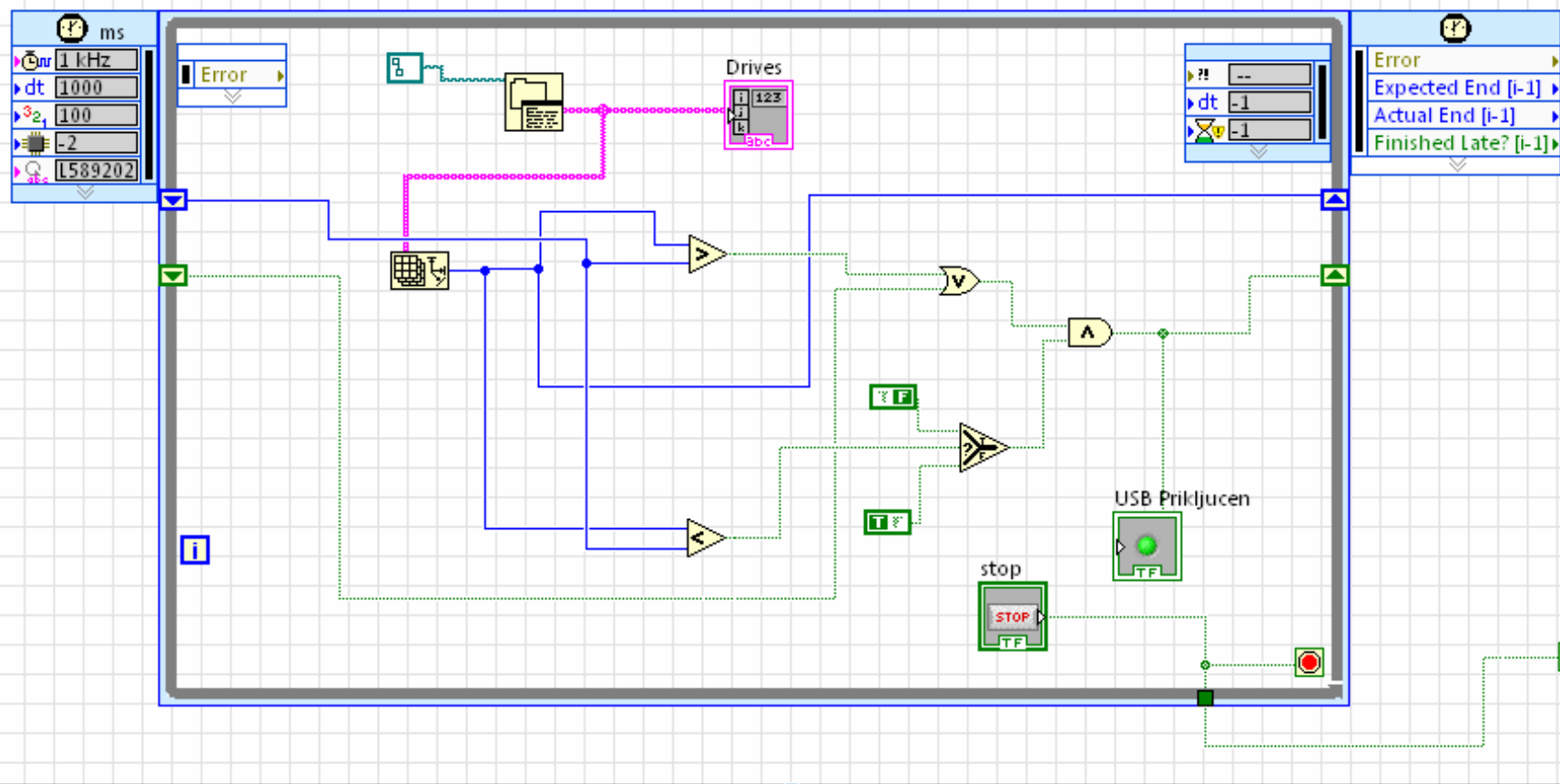
Primer 6

Kreirati virtuelni instrument koji detektuje prisustvo USB stick modula. Indikaciju prisustva izvršiti vizuelnom signalizacijom

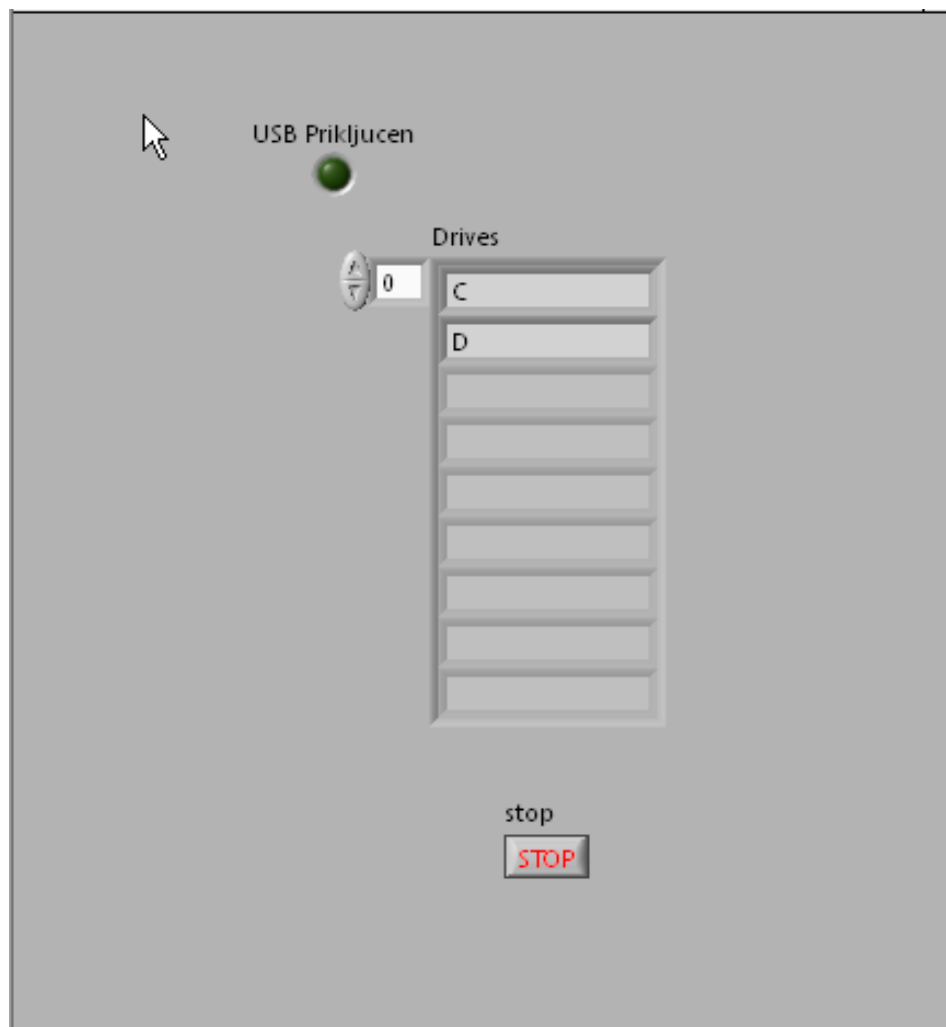
1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Postaviti **List Folder** kontrolu (Programming-File IO-Advanced File Functions-List Folder)
4. Postaviti **Path Constant** (prazna) (Programming-File IO-File Constants)

5. Dodati **Array** strukturu (**Modern-Array Matrix Cluster-Array**) sa pridruženim String tipom podataka
6. Dodati blok za određivanje dužine niza i porediti prethodnu vrednost sa sadašnjom (poređenje vršiti i na veće i manje vrednosti, dva izlaza)
7. Izlaze prethodnog bloka dovesti na realizovan latch
8. Izlaz latch-a dovesti na LED indikator
9. Obezbediti da se provera prisustva sitck-a vrši jednom u sekundi

Izgled blok dijagrama



Izgled front panela



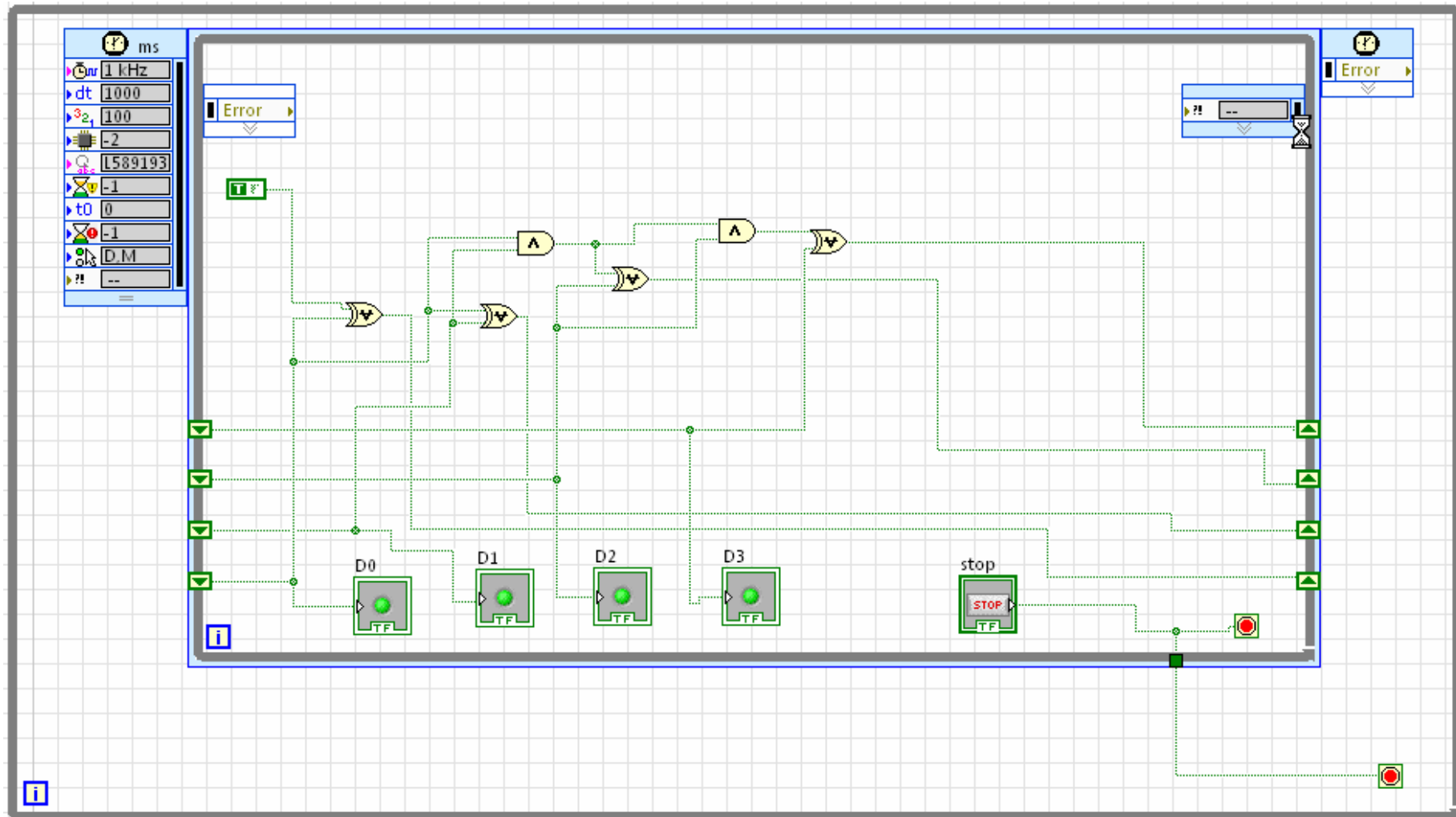
Primer 7

Kreirati virtuelni instrument koji simulira četvorobitni sinhroni brojač sa indikacijom brojanja i promenom osnovne periode. Indikacija se ostvaruje pomoću LE dioda

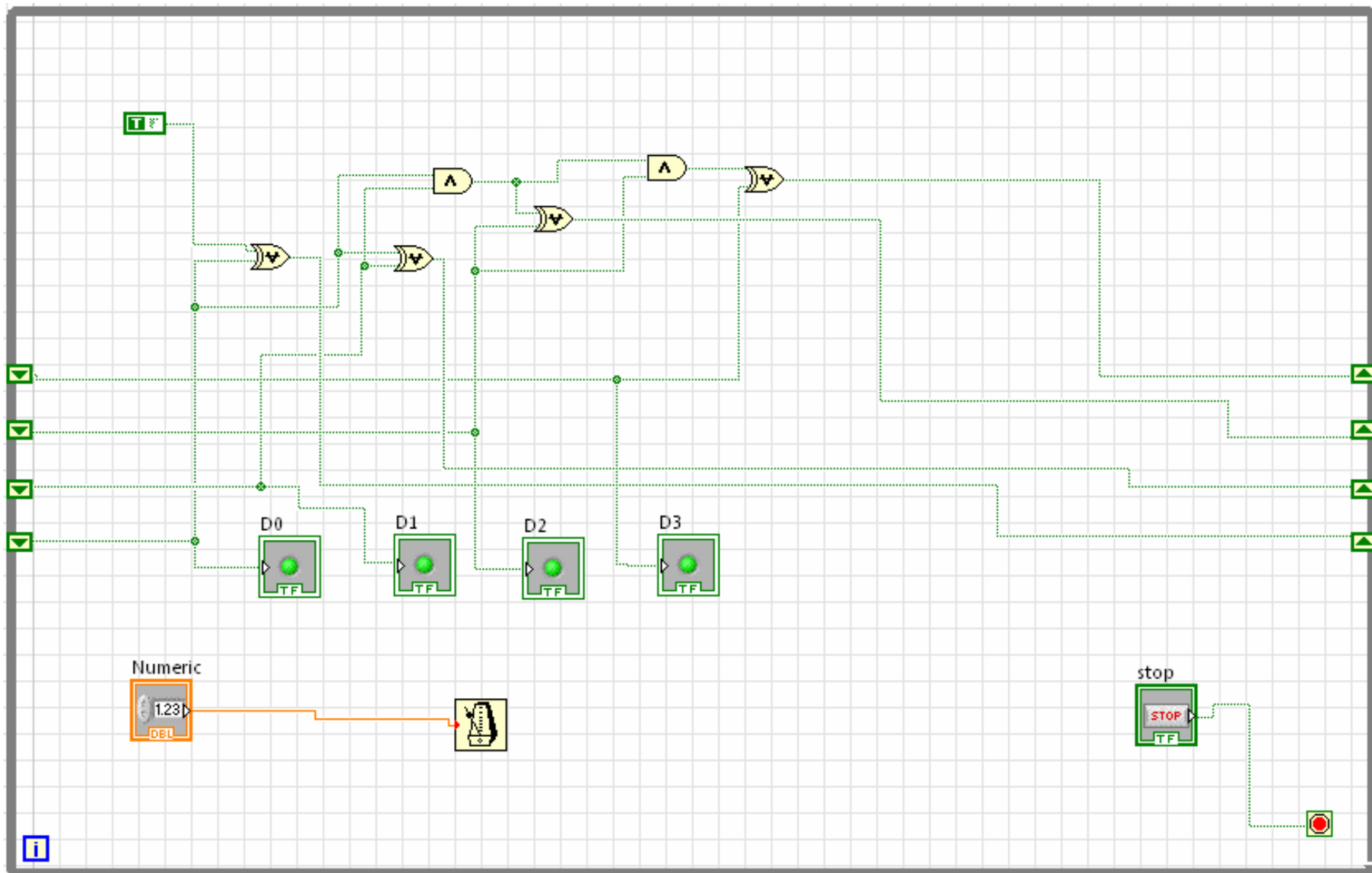
1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Postaviti **Timed loop** petlju (programming-structures-timed structures) u okviru while petlje
4. Koristeći pomeračke registre i osnovna logička kola realizovati četvorobitni brojač

5. U okviru front panela dodati LED indikatore kao i numričku kontrolu za promenu periode vremenske petlje
6. Uraditi isti primer bez korišćenja **Timed Loop**, (samo while petlja)

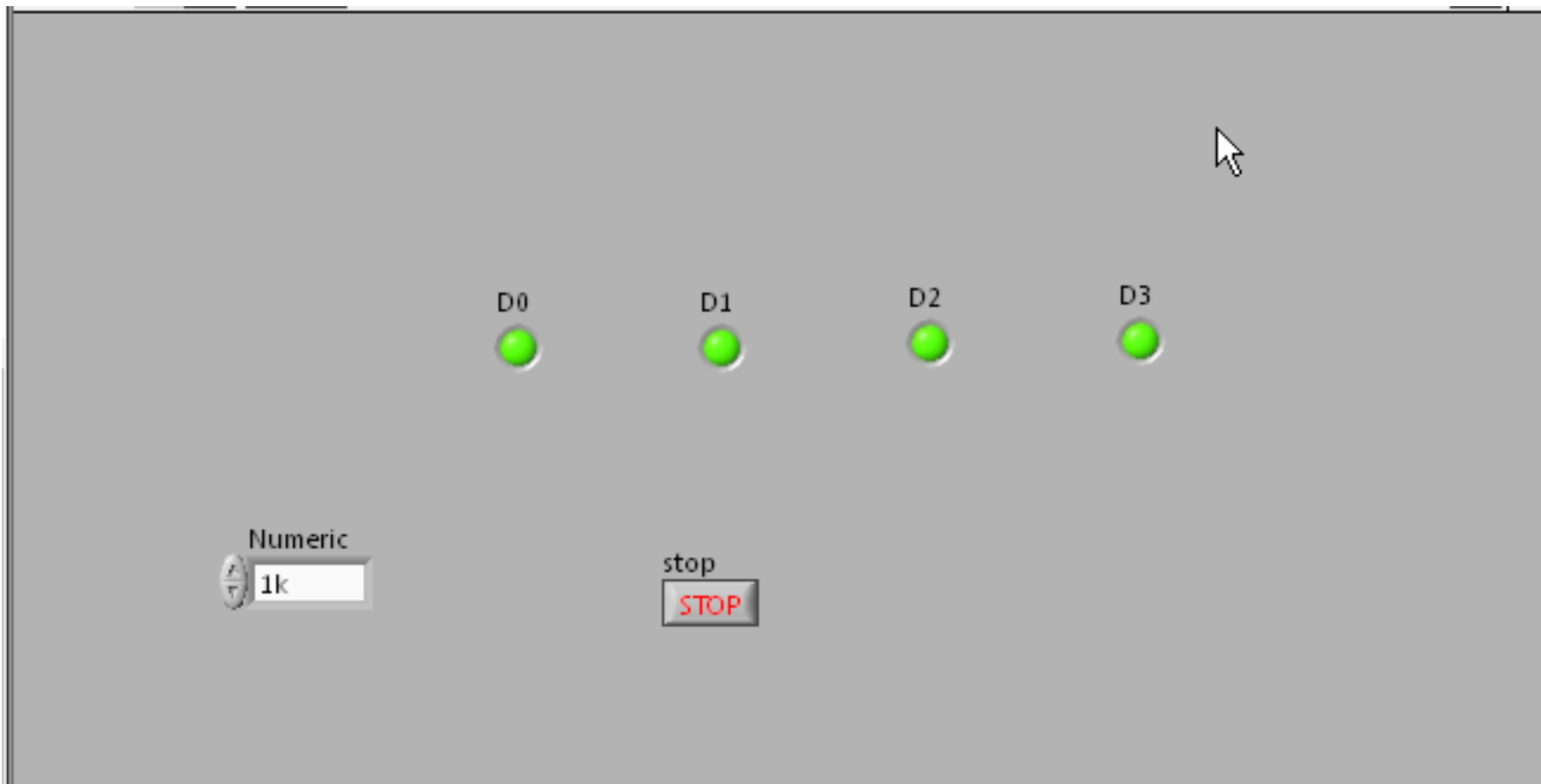
Izgled blok dijagrama u prvom slučaju



Izgled blok dijagrama u drugom slučaju



Izgled front panela



Šesta vežba

Primer 7

Kreirati virtuelni instrument koji snima talasni oblik napona dovedenog na neki analogni ulaz NI USB-6008 akvizicionog modula.

1. U okviru Getting started ekrana selektovati opciju za kreiranje praznog VI
2. Postaviti **while** petlju (programming-structures) u okviru block diagram editora, potom kreirati stop dugme u okviru front panela i povezati while petlju i **stop** dugme
3. Selektovati blok **DAQ Assistant** u okviru **Express-Input** kategorije
4. Pojavljuje se **Create New Express Task** wizard

5. U okviru prethodnog wizarada selektovati **Acquire Signals-Analog Input**
6. Odabrati **Voltage** kao merenu kategoriju
7. Ukoliko je priključen **NI USB-6008** modul pojaviće se lista sa raspoloživim analognim kanalima
8. Selektovati odgovarajući analogni kanal na kojem se nalazi konektovan priključak napona koji se meri (npr ai0)
9. U okviru podešavanja ulaznog napona selektovati **RSE** za **terminal configuration**

10. **Acquisition mode** postaviti na **Continuous Samples**
11. Ostale parametre ne menjati, sa OK zatvoriti wizard
12. Selektovati **DAQ Assistant** blok i akcijom **Create-Graph indicator** kreirati grafik za prikaz talasnog oblika priključenog napona, na njemu podesiti mereni opseg - 10v-10v
13. Na generatoru signala postaviti sinusni signal amplitude 0.5V i referentne vrednosti 1V
14. Kreirati kontrolu za dinamičku promenu X i Y ose grafika

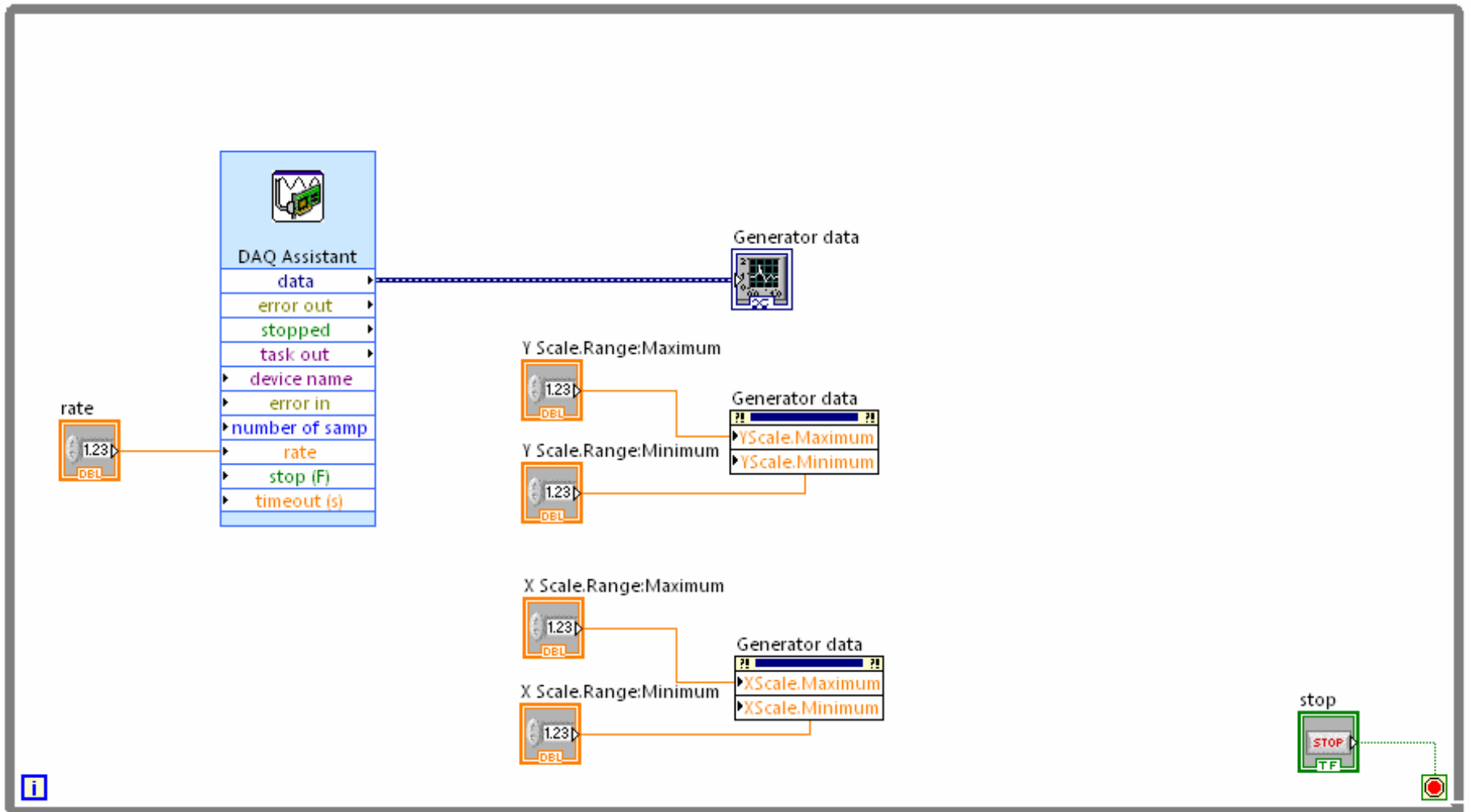
15. Selektovati blok grafa, **Create-Property node-X**
Scale-Range-Maximum i Minimum

16. Isto uraditi i za Y osu

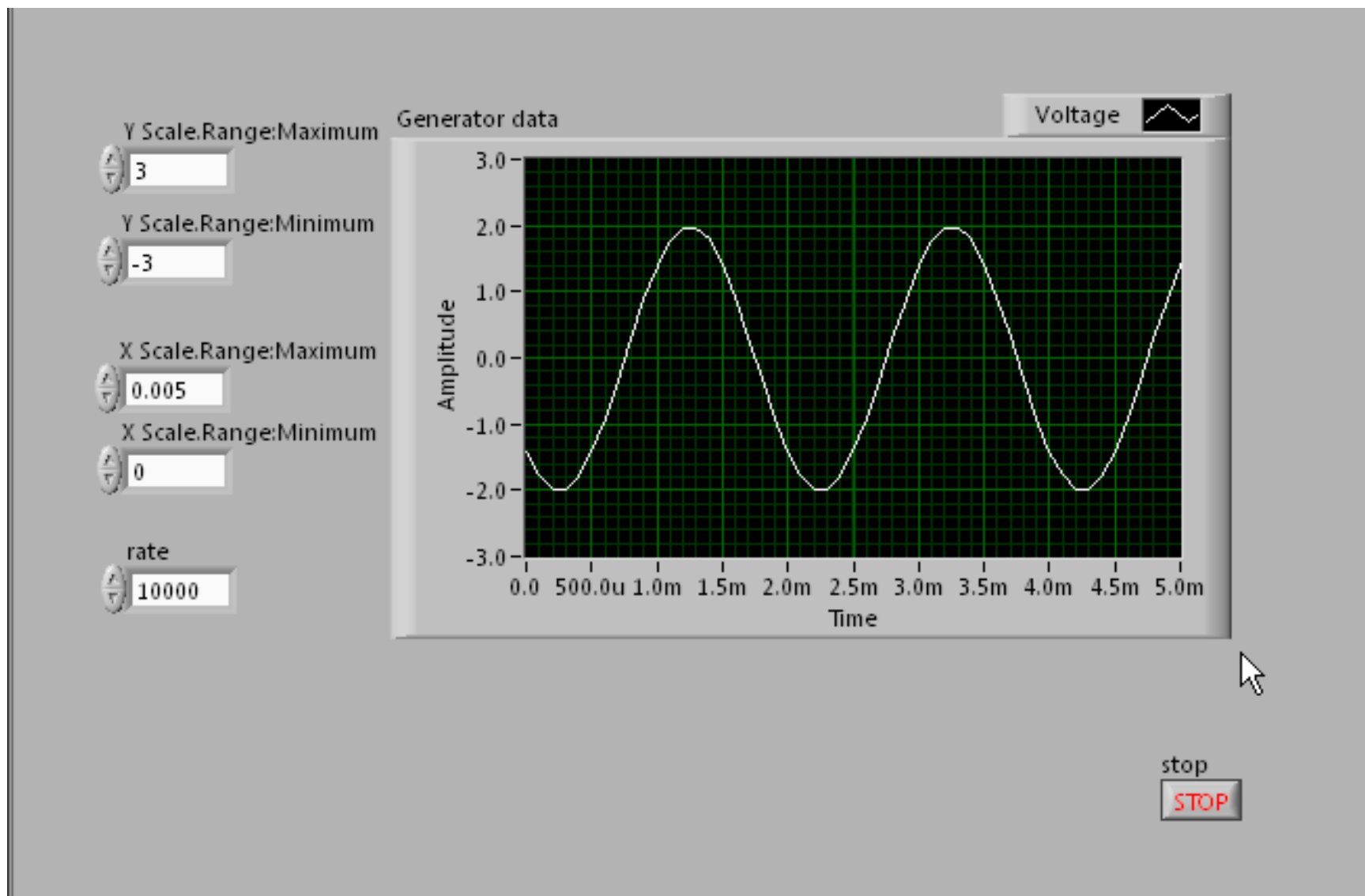
17. Kreirati numeričke kontrole za promenu **Range**
atributa osa

18. Kreirati numeričku kontrolu za promenu periode
odabiranja akvizicionog modula (**rate**)

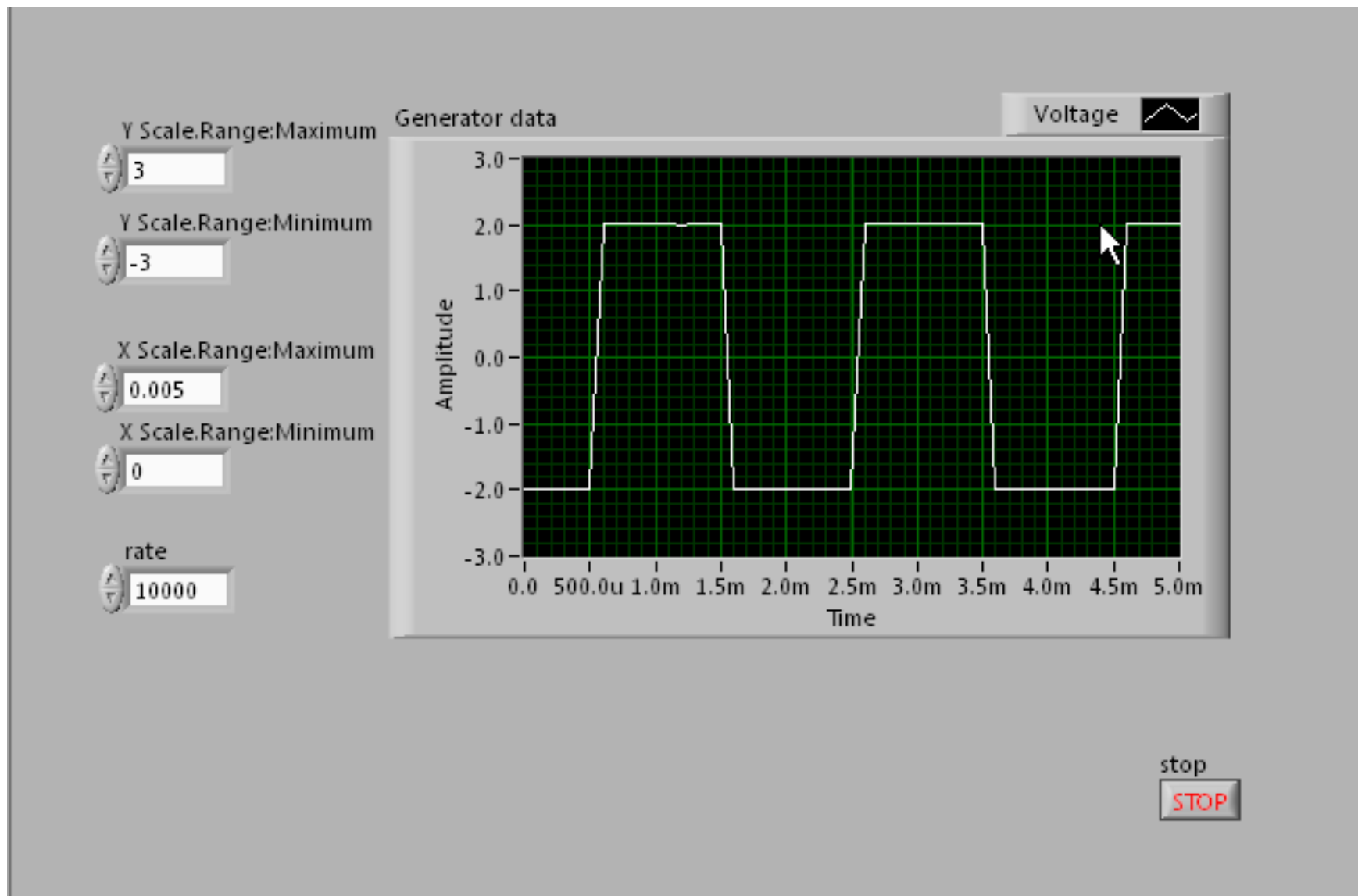
Izgled blok dijagrama



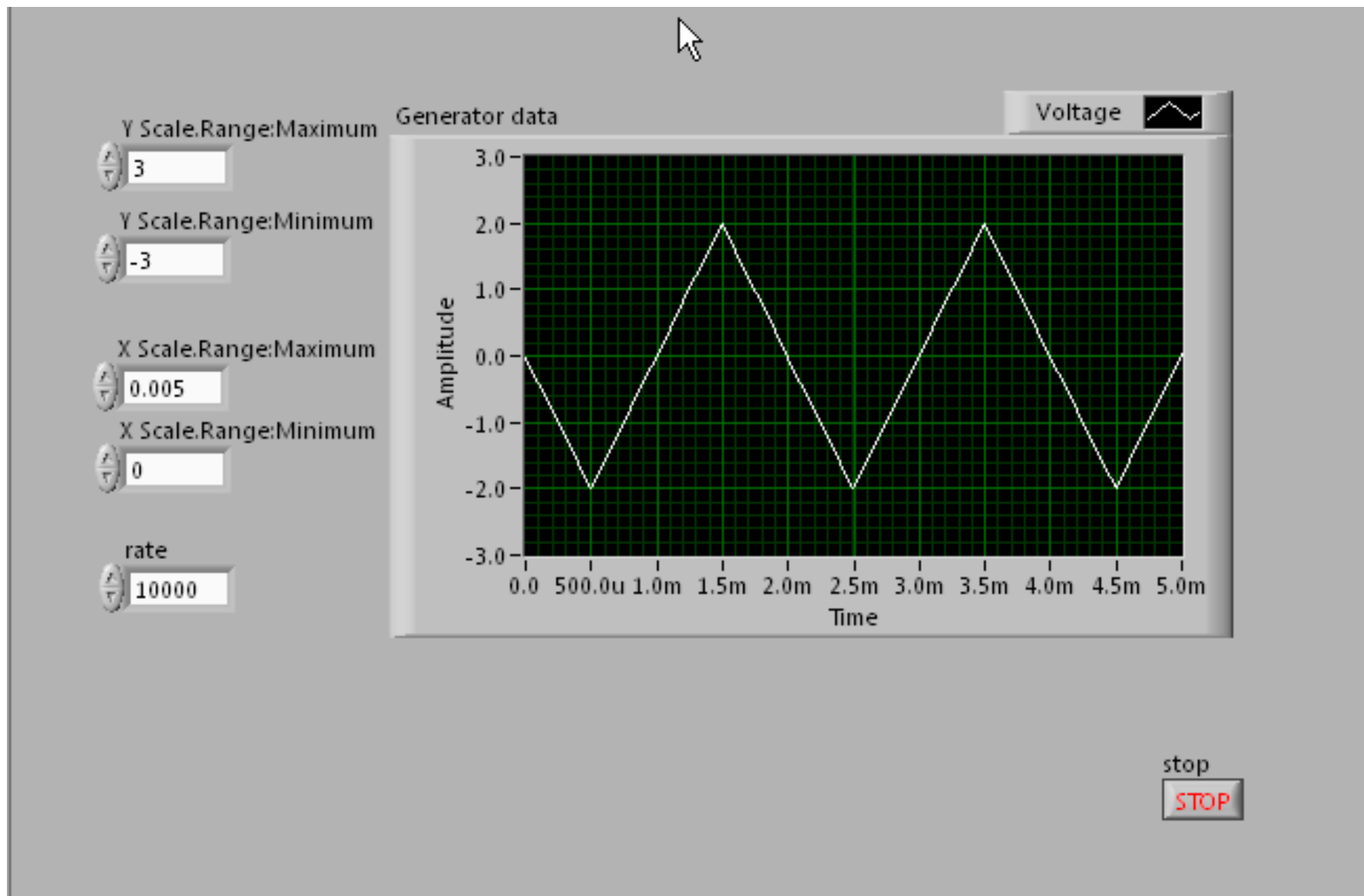
Izgled front panela



Izgled front panela



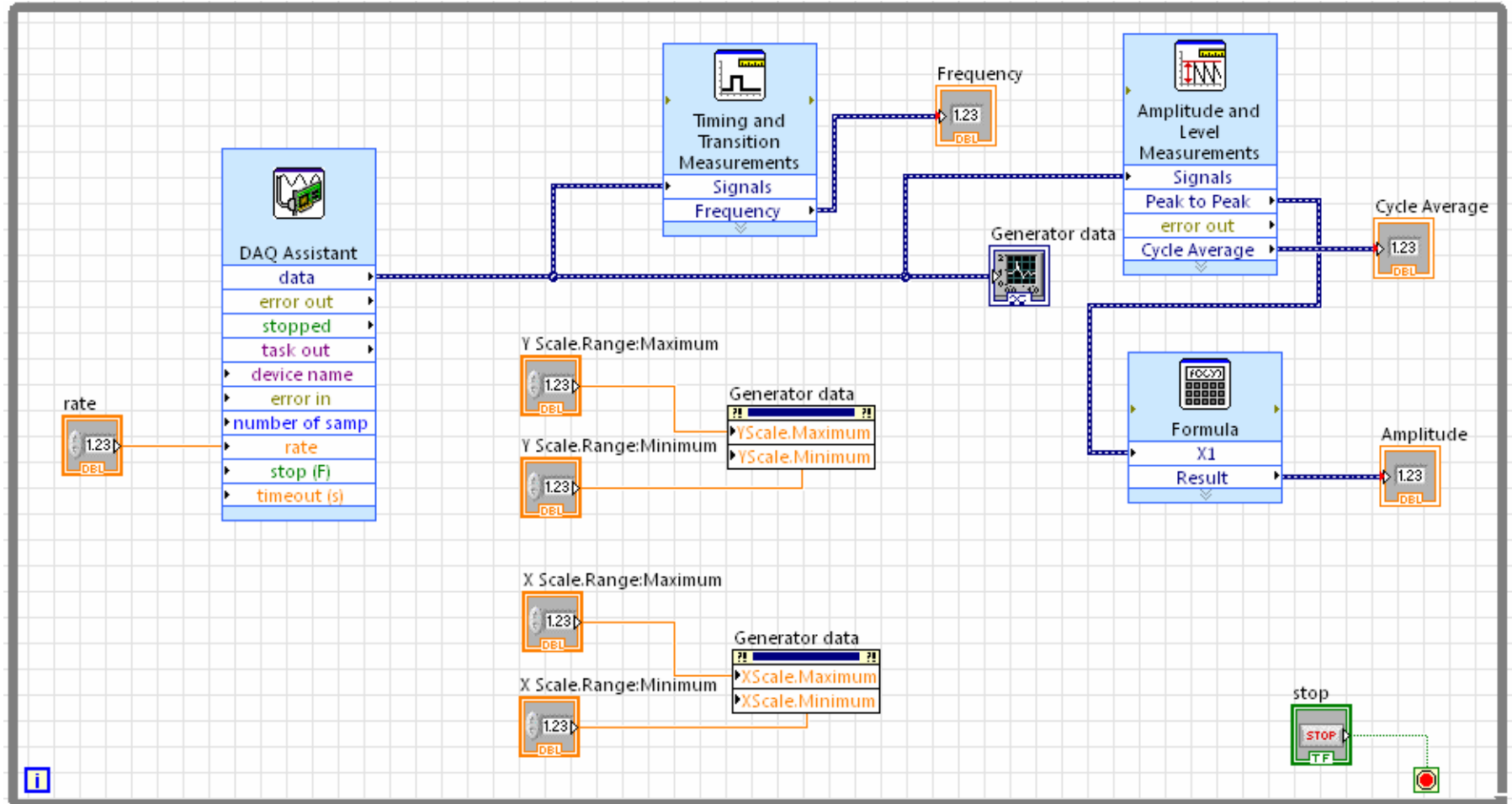
Izgled front panela



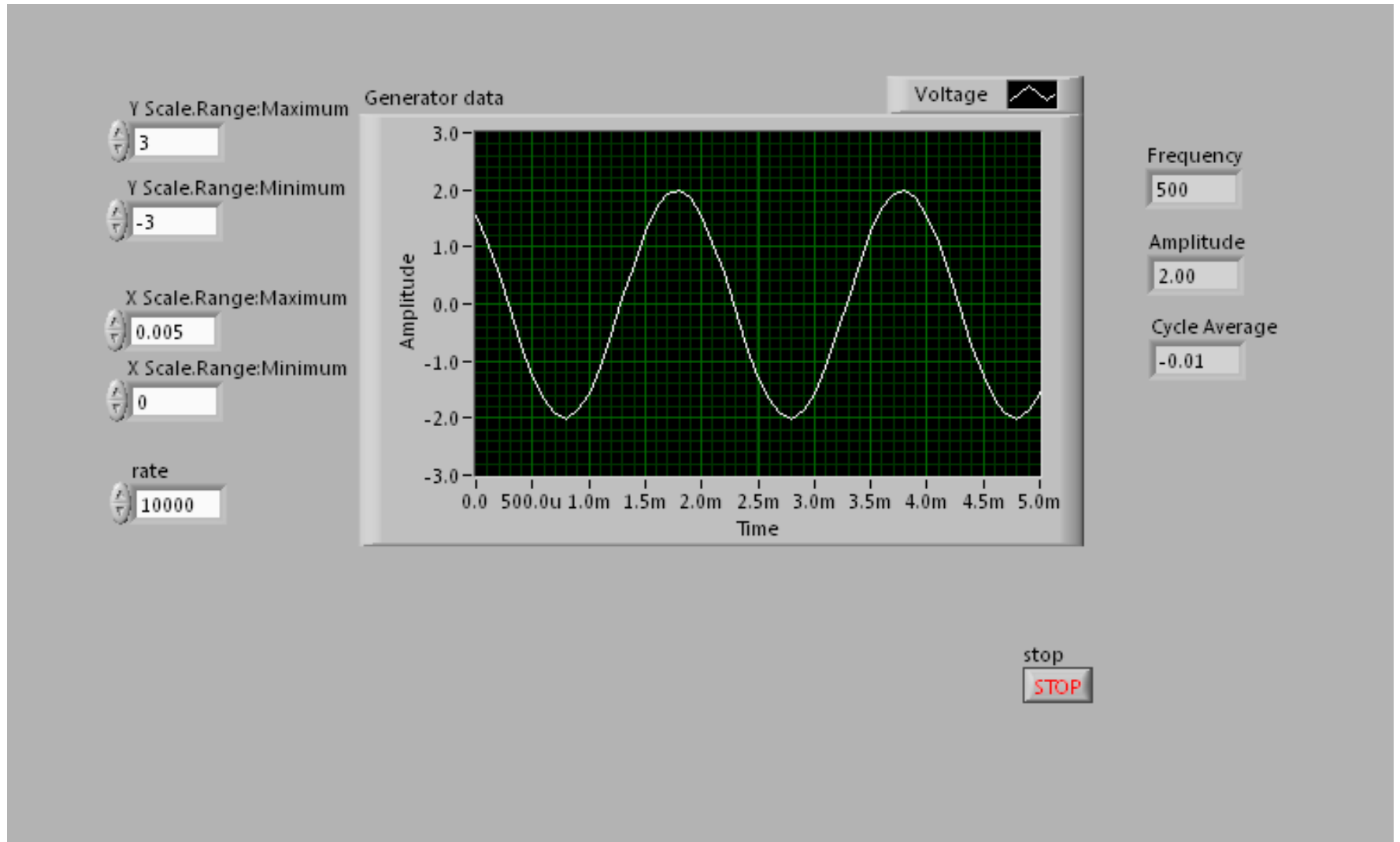
Primer 8

Dograditi prethodni instrument tako da prikazuje osnovne parametre signala, učestanost, amplitudu i srednju vrednost. Vrednosti za amplitudu i srednju vrednost treba da budu prikazane sa tačnošću 0.01

Izgled blok dijagrama



Izgled front panela



Izgled front panela

