



Fakultet tehničkih nauka u Novom Sadu
Departman za energetiku, elektroniku i telekomunikacije
Katedra za električna merenja



LabVIEW™

razvojno okruženje
-uvod-

Školska godina
2022 / 2023

Jelena Rafailović

jelenarafailovic@uns.ac.rs
jelenarafailovic98@gmail.com

Literatura

- www.ni.com/academic/students/learn-labview/
- *Getting Started with LabVIEW, National Instruments*
- *LabVIEW User Manual, National Instruments*
- LABVIEW CORE 1 PARTICIPANT GUIDE
- LABVIEW CORE 1 PARTICIPANT GUIDE
- LabVIEW Examples
- *J. Jerome: Virtual Instrumentation Using LabVIEW*



LabVIEW

Vežba 1

Osnove virtuelne instrumentacije i LabVIEW
programskog okruženja



LabVIEW je grafički programski jezik koji koristi ikonice umesto tekstualnih linija koda za programiranje i kreiranje aplikacija. Za razliku od tekstualno koncipiranih programskih jezika, gde instrukcije određuju izvršavanje programa, LabVIEW koristi dataflow programiranje, gde je izvršavanje programa određeno protokom podataka.

U LabVIEW-u, korisnički interfejs se kreira pomoću skupa alata i objekata. Za korisnički interfejs koristi se naziv *front panel*. Dodavanjem koda u obliku grafičke reprezentacije raznih funkcija vrši se kontrola nad front panelom. Prostor gde se nalazi programski kod naziva se *block diagram*. Na neki način, block diagram liči na dijagram toka (flowchart)

Pored osnovnog seta funkcija koje dolaze sa LabVIEW-om, omogućeno je i integrisanje dodatnih alata za razvoj specijalizovanih aplikacija.

LabVIEW- Laboratory Virtual Instrumentation Engineering Workbench

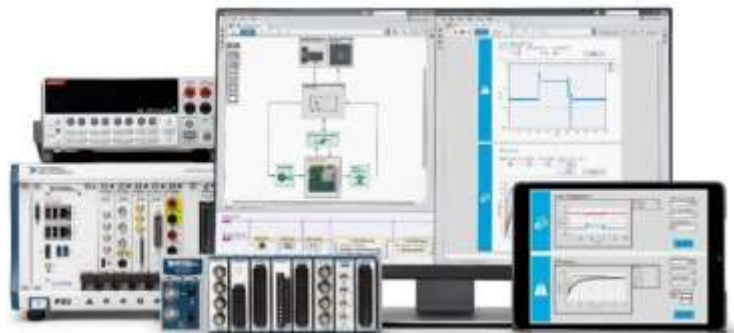
LabVIEW omogućava:

- Razvoj uređaja i sistema i njihovo testiranje
- Realizaciju akvizicije i obrade mernih rezultata
- Izradu celokupnih kontrolnih sistema (SCADA)



Široko rasprostranjen:





Virtuelni instrument može biti definisan kao:

„Skup softvera i/ili hardvera dodatog personalnom računaru na način da korisnici imaju mogućnost interakcije sa istim kao da koriste namenski dizajniran tradicionalni električni instrument“

-Virtual Bio-instrumentation by Jon B. Olansen and Eric Rosow

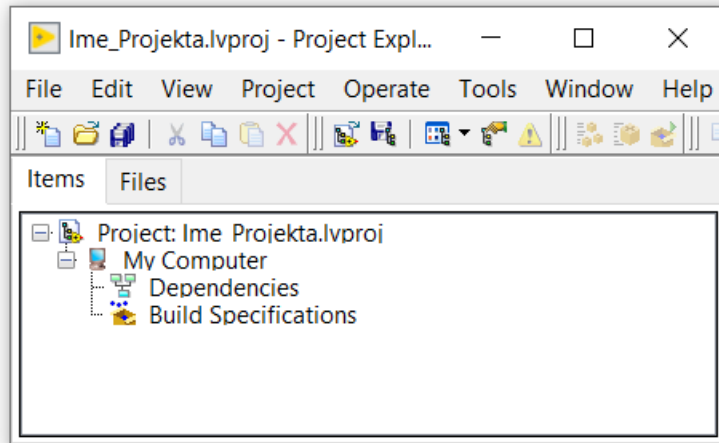


U LabVIEW terminologiji, program pisan u LabVIEW okruženju naziva se upravo virtuelni instrument, ili skraćeno VI, iz razloga što njegov izgled i operacije mogu da imitiraju stvarne instrumente, poput osciloskopa ili multimetra.

LabVIEW Project

LabVIEW project sadrži VI-ove, fajlove potrebne za pravilan rad tih VI-ova, kao i dodatne fajlove kao što je npr. dokumentacija o samom projektu. LabVIEW project omogućava organizaciju svih VI-ova i fajlova neke aplikacije.

Novi project: **File > Create Project > Blank Project > Finish**

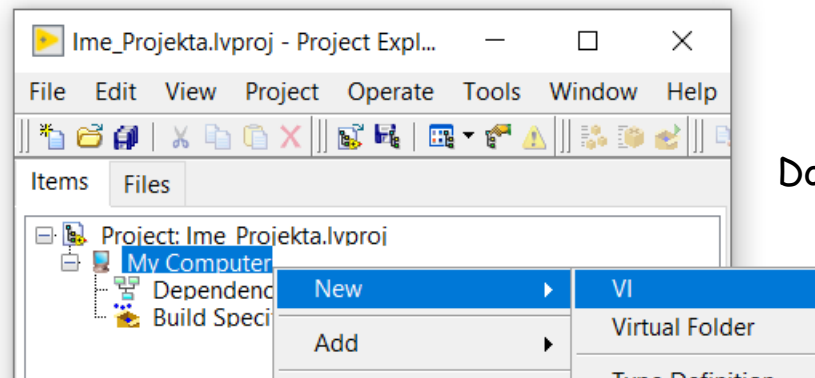


Project root - sadrži sve komponente projekta

My Computer - platforma za izvršavanja projekta

Depedencies - Sadrži VI-ove i druge fajlove koju su potrebni za izvršavanje projekta na zadatoj platformi

Build Specification - Konfiguracija za kreiranje .exe (stand alone) , installer-a, i drugih distribucija projekta



Dodavanje novog VI-a u projekat

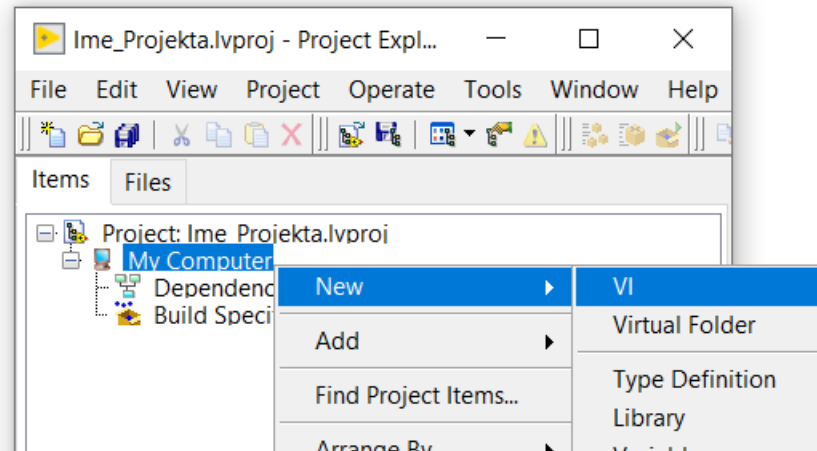
Novi *Project*

1. Kreirati novi projekat:

File > Create Project > Blank Project > Finish

2. Sačuvati opcijom ***File > Save As...***

3. Dodavanje novog VI-a u projekat:



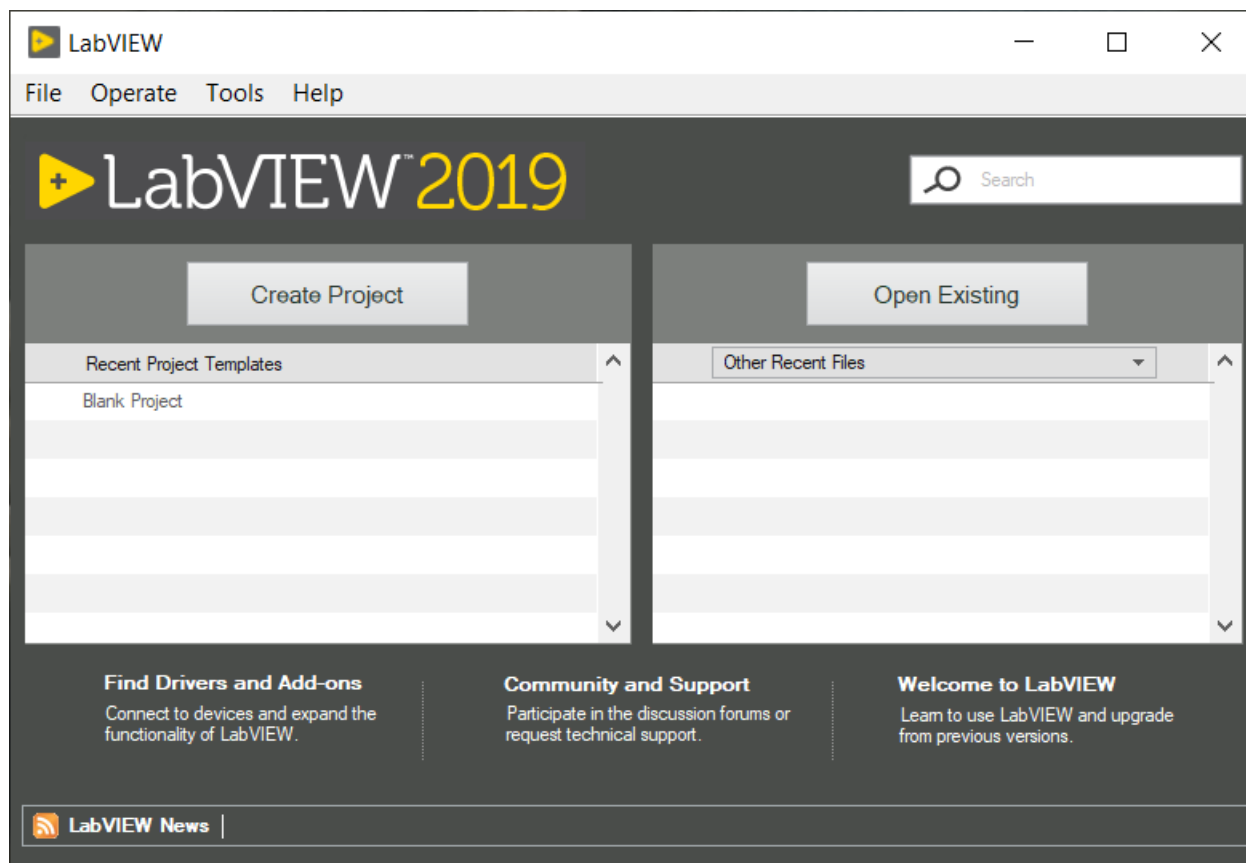
4. Sačuvati ga opcijom ***File > Save As...***

Ekstenzije LabVIEW fajlova:

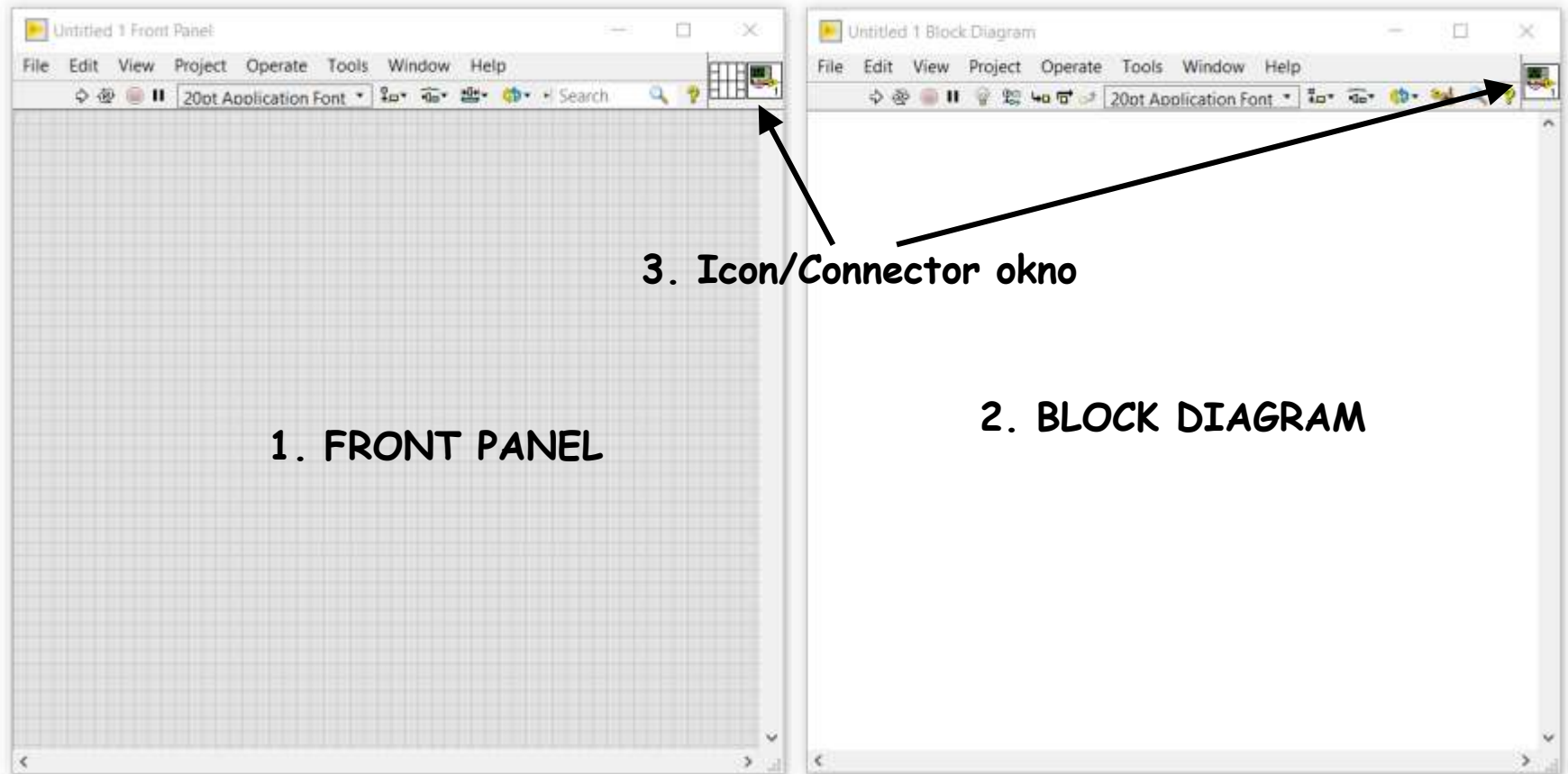
LabVIEW Project - **.lvproj**

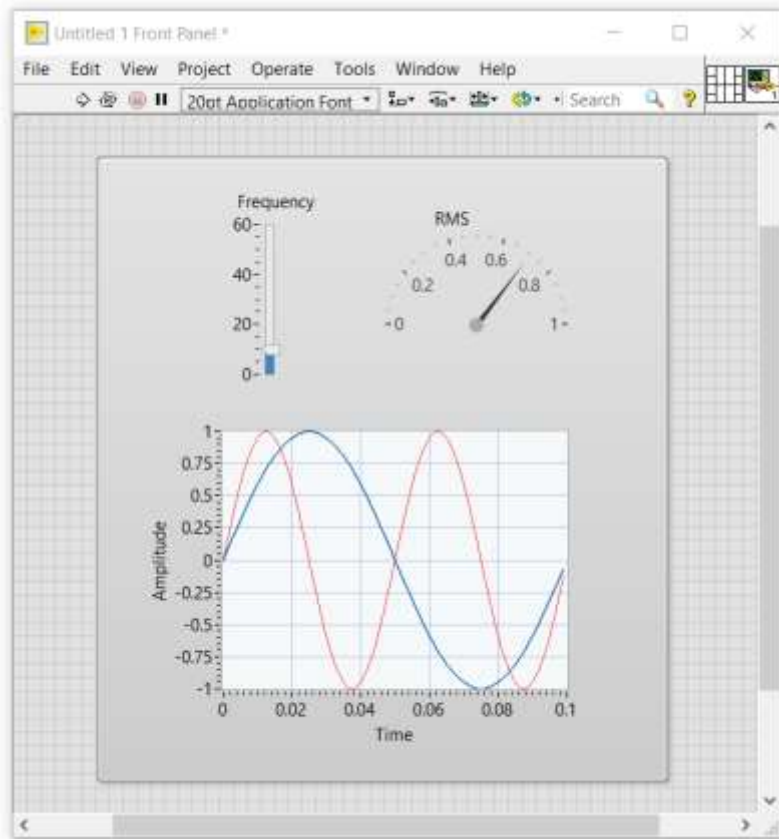
Virtual Instrument (VI) - **.vi**

Početni prozor



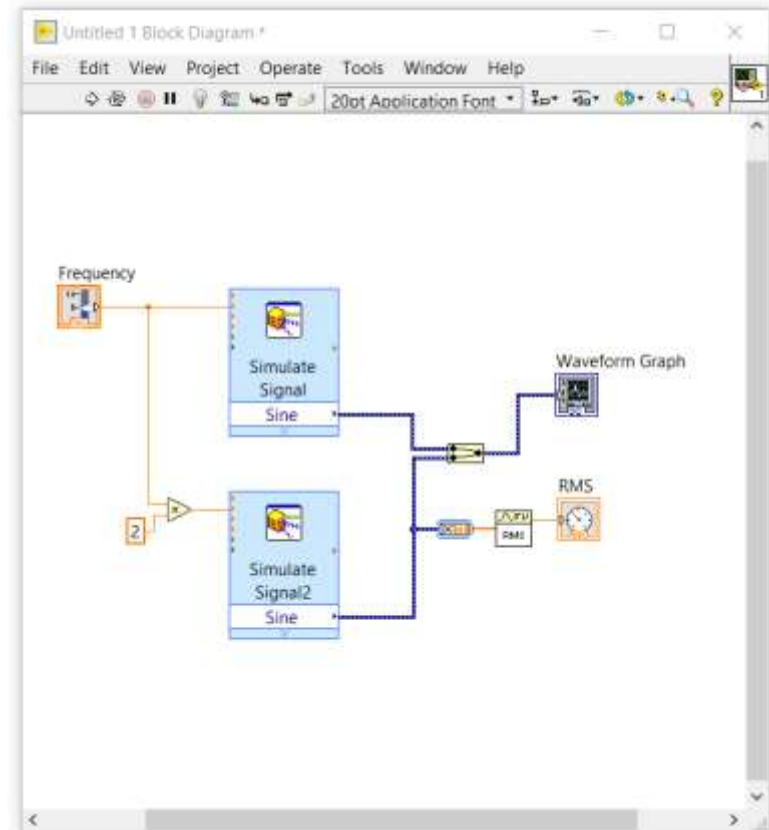
Tri komponente VI-a





Front panel - interakcija korisnika sa aplikacijom (korisnički interfejs)

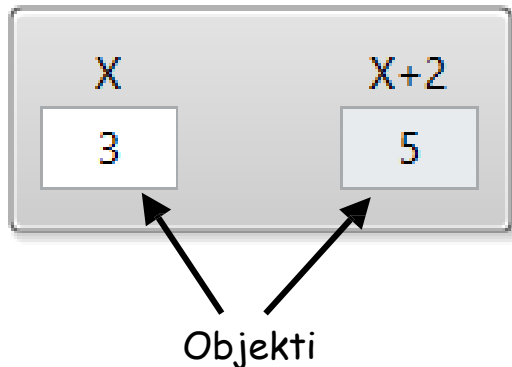
Block diagram - programski kod aplikacije



Objekti vs Terminali

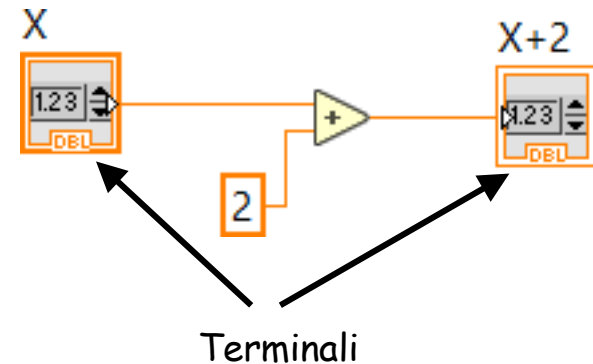
Objekti

- Nalaze se na **Front Panel-u**
- Služe za interakciju korisnika sa aplikacijom



Terminali

- Nalaze se na **Block Diagram-u**
- Služe za povezivanje objekata na Front Panel-u i programskog koda



OBJEKTI sa Front Panel-a se pojavljuju kao TERMINALI na Block Diagram-u.

OBJEKTI  TERMINALI

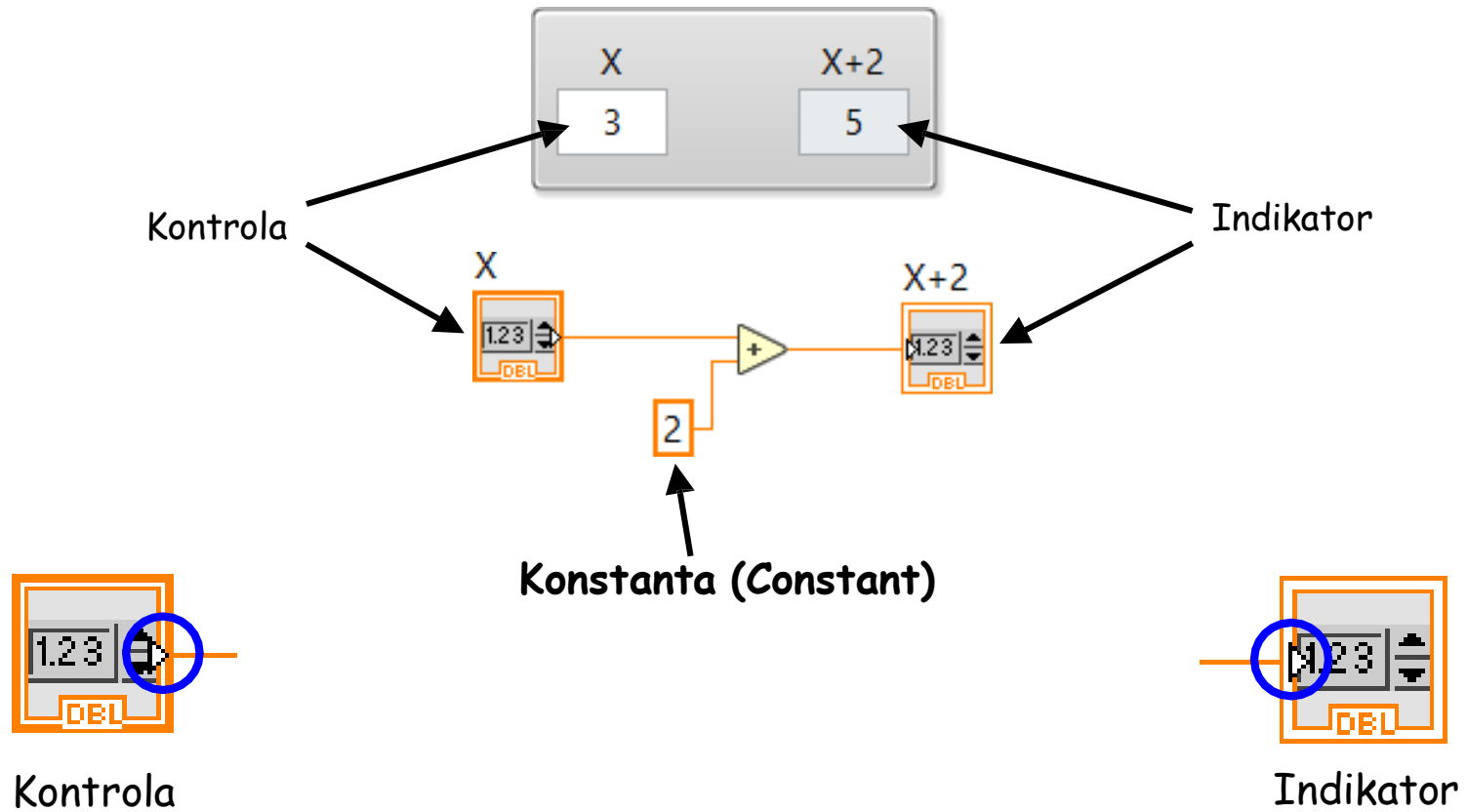
Kontrole, Indikatori i Konstante

Kontrole (Controls)

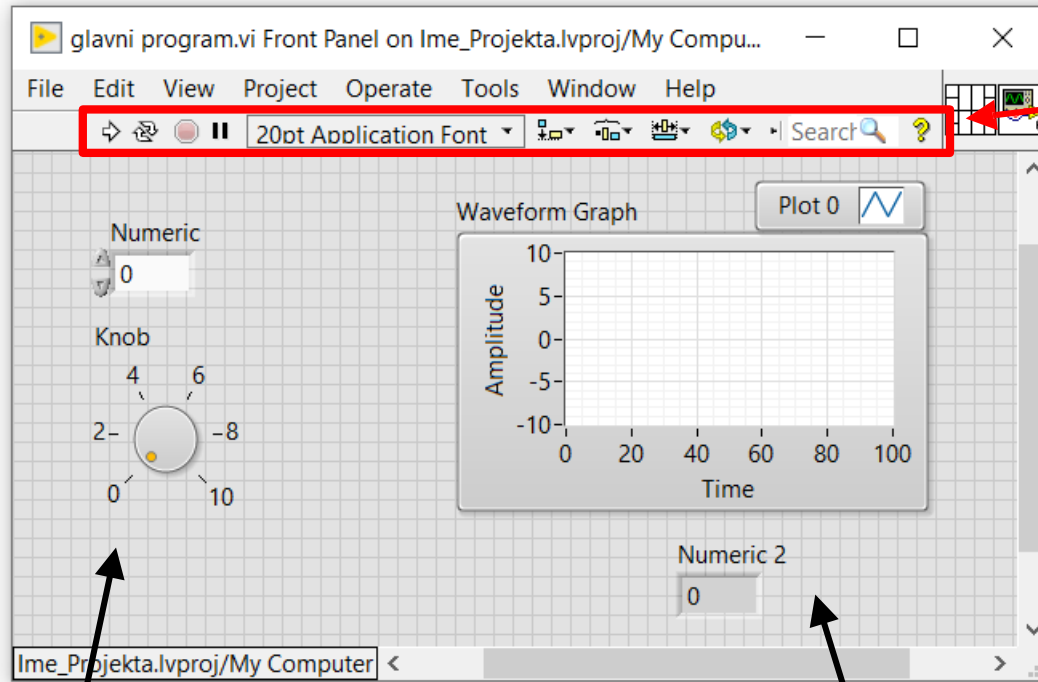
Služe za ulaz podataka

Indikatori (Indicators)

Služe za izlaz podataka



Front panel



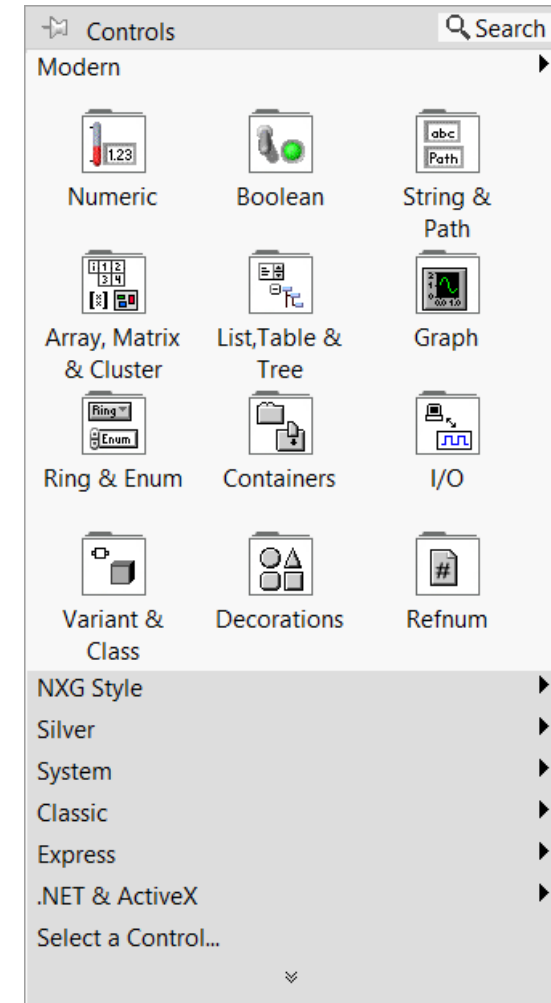
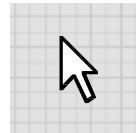
Window Toolbar

Kontrole - ulazi

Indikatori - izlazi

Izbor kontrola i indikatora - **View > Controls Palette**
ili

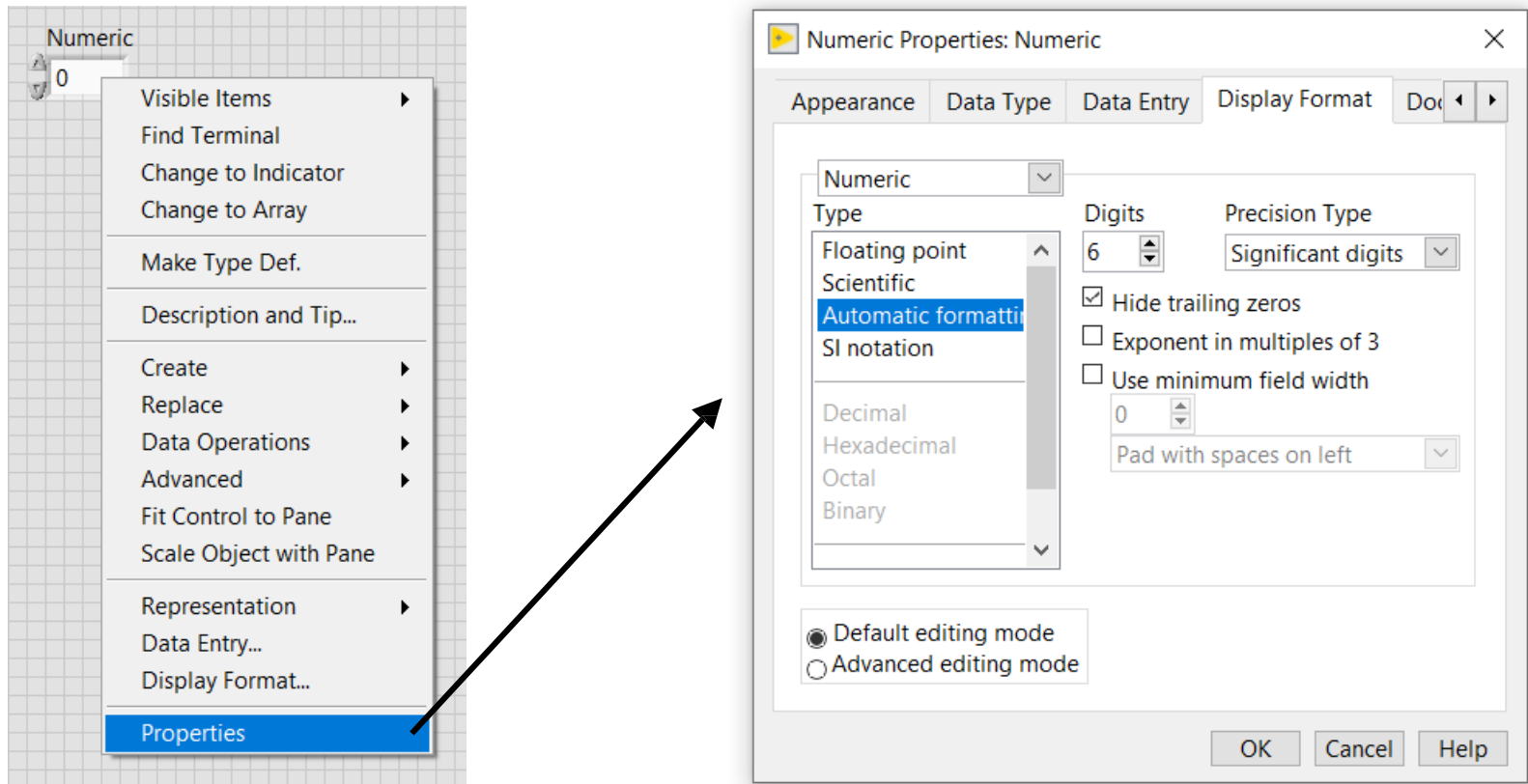
desni klik na pozadinu



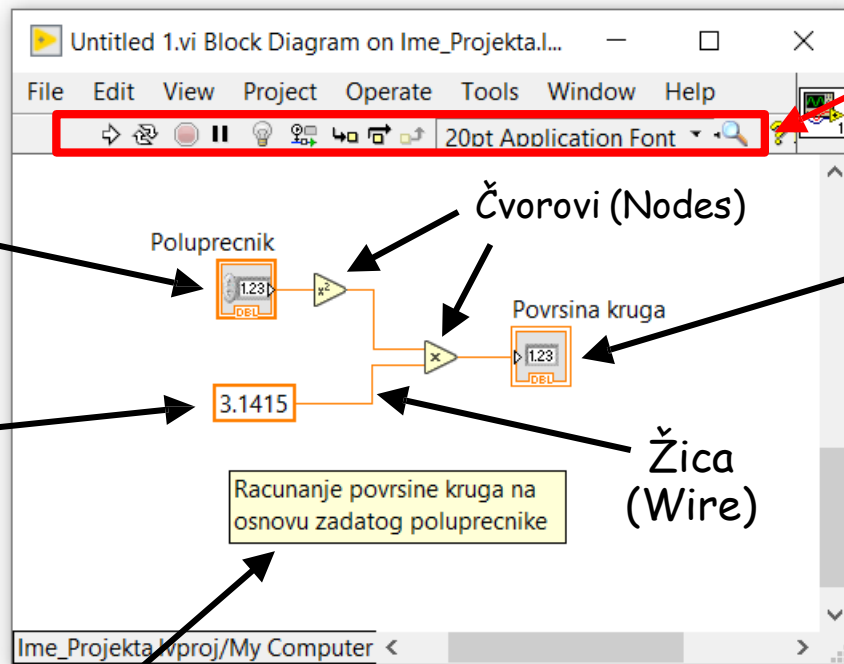
Front panel

Desni klik na određenu kontrolu ili indikator daje mogućnosti podešavanja

Izborom **Properties** sve karakteristike tog objekta postaju dostupne



Block diagram



Window Toolbar

Kontrola

Čvorovi (Nodes)

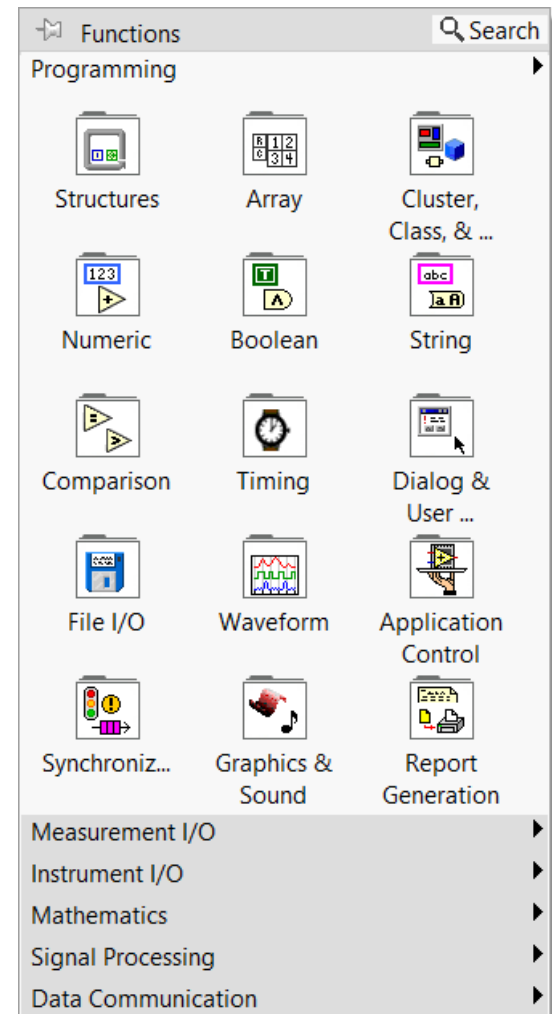
Indikator

Povrsina kruga

Konstanta

Žica (Wire)

Komentar



Izbor funkcija i konstanti - **View > Functions Palette**

ili

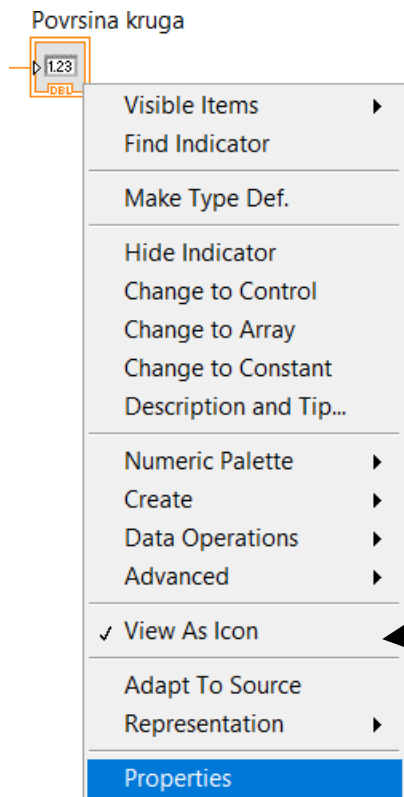
desni klik na pozadinu



Block diagram

Desni klik na određenu kontrolu, indikator, konstantu ili funkciju daje mogućnosti podešavanja

Izborom **Properties** sve karakteristike tog objekta postaju dostupne



Control



Control



Indicator

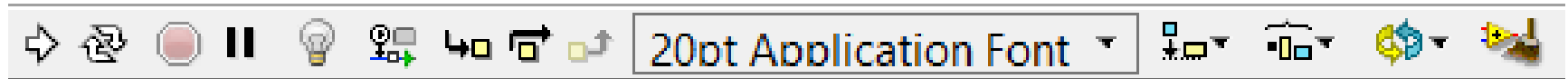







Indicator



Umanjen prikaz kontrole/indikatora:
Desni klik na kontrolu/indikator >> View As Icon

Window Toolbar



-  - **Run** (pokretanje programa)
-  - **Run Continuously** (kontinualno pokretanje programa nakon završetka)
-  - **Abort Execution** (zaustavlja izvršavanje programa)
-  - **Pause** (pauzira izvršavanje programa)
-  - **Highlight Execution** (grafički prikazuje tok podataka na block diagram-u)

 - dodatne funkcije

 - rad sa tekstom (npr. font kontrole/indikatora)

 - estetski rad sa objektima i terminalima

 - **Clean_Up Diagram** („autorouter“ alat)

 - brza pretraga

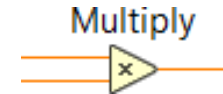
 - Show Context Help (Ctrl + H)

Čvorovi (Nodes)

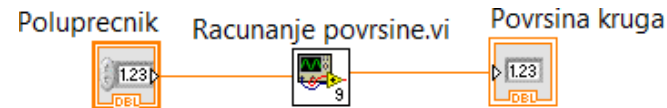
Komponente na Block Diagram-u koji imaju ulaze i izlaze i obavljaju određene operacije.

Čvorovi se mogu podeliti na:

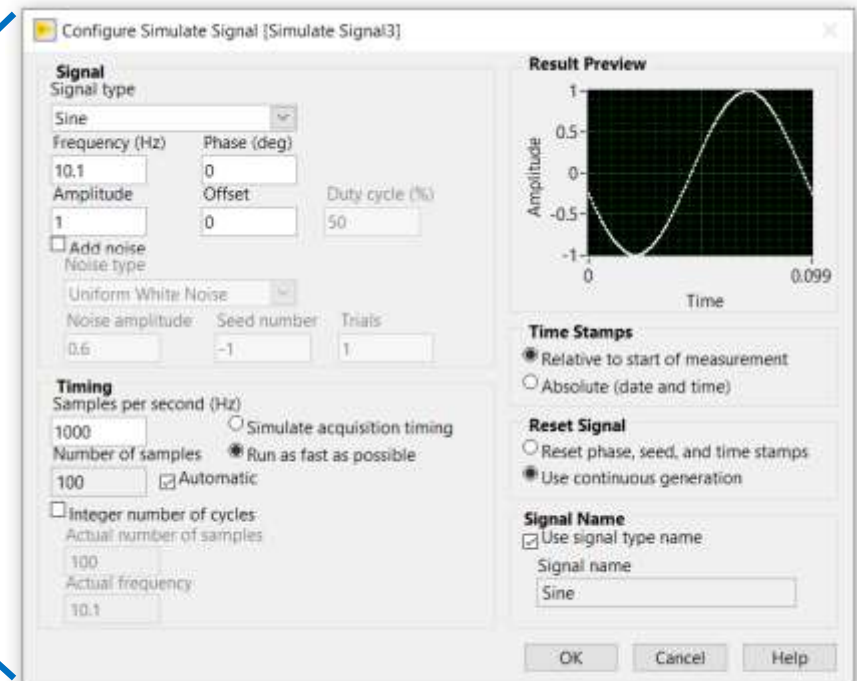
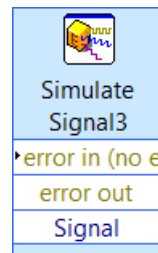
- **Functions** - fundamentalni funkcijski blokovi LabVIEW-a, ne sadrže ni front panel ni block diagram



- **Standardni VI** - koriste se ka SubVI



- **Express VI** - interaktivni VI koji poseduju dijalog za konfigurisanje



Tools Palette (Paleta alata)

View>>Tools Palette ili *Shift + desni klik*

- Alat predstavlja specijalni mod funkcionisanja kursora
- LabVIEW može automatski da izabere alat koji mu je potreban
- Dostupan je i na Front Panel-u i na Block Diagram-u, i služi za modifikovanje sadržaja na njima

Automatska selekcija
alata uključena

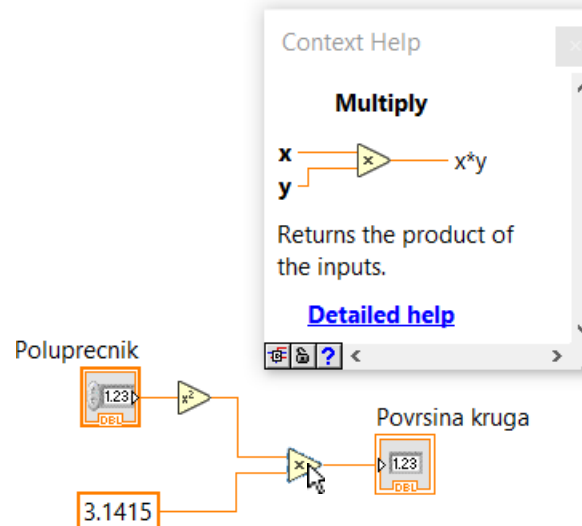


Automatska selekcija
alata isključena



Pomoć

1. *Help>>Show Context Help* ili *Ctrl+H* i pozicionirati se kursorom na funkciju

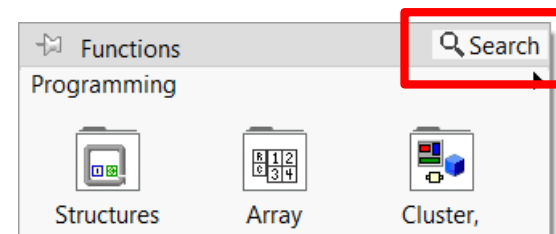
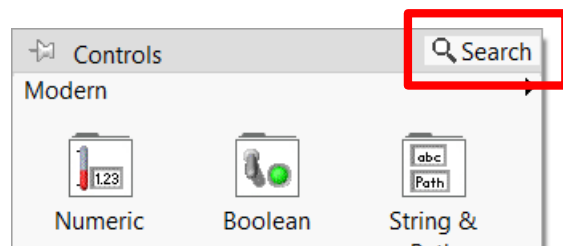


2. *Help>>LabVIEW Help...*

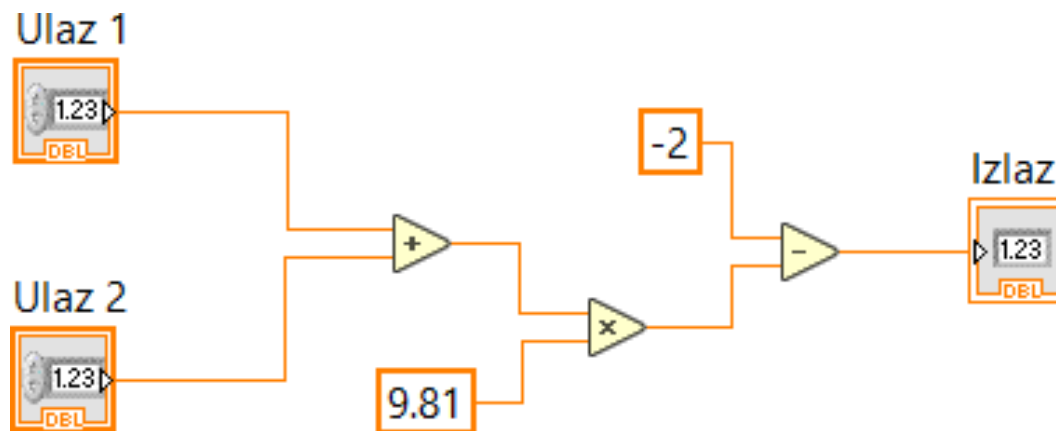
3. *Help>>Find Examples...* – primeri koji dolaze sa LabVIEW-om

4. *View>>Quick Drop* ili *Ctrl+space* – Brzo pronalaženje kontrola, funkcija

5. *Controls/Functions Palette>>Search*

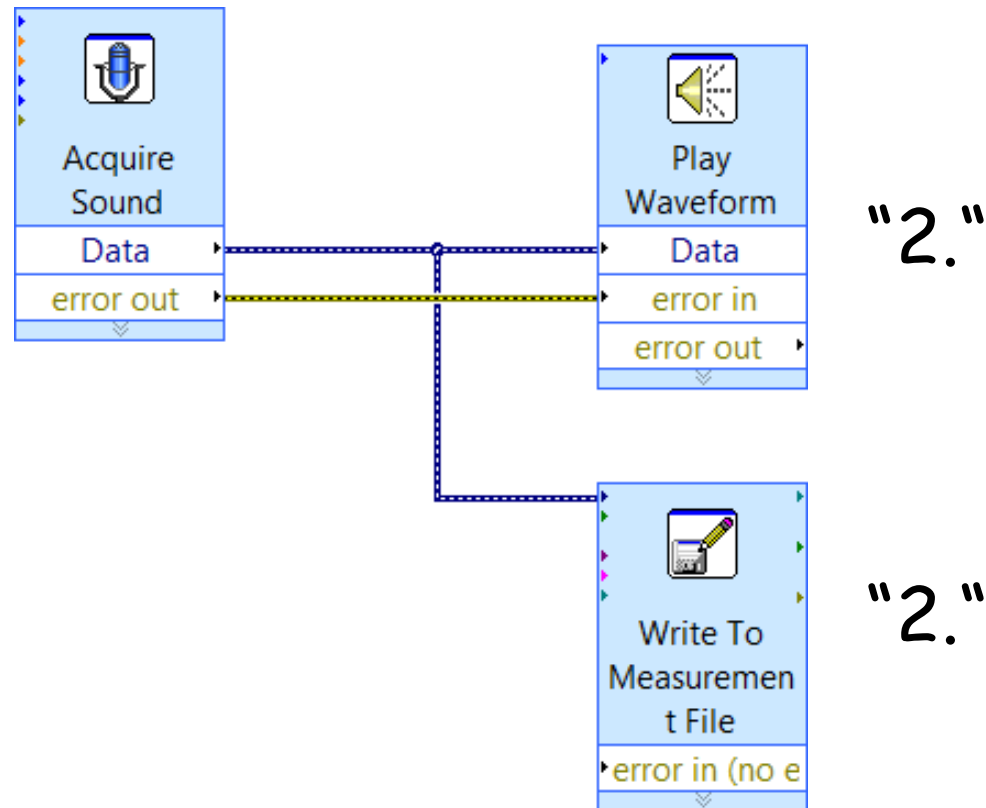


Dataflow programiranje



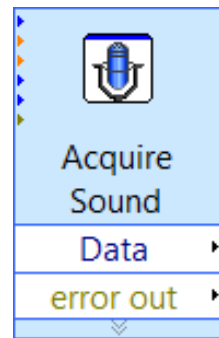
Dataflow programiranje

1.

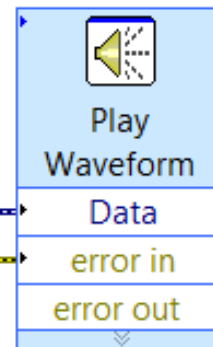


Dataflow programiranje

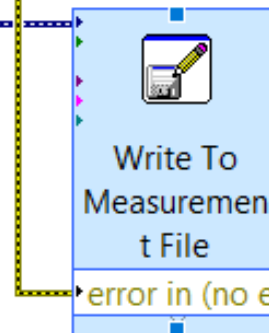
1.



2.



3.
























Strukture i tipovi podataka u LabVIEW

Osnovna podela:

- **Numeric** tip podatka
- **Boolean** tip podatka
- **String** tip podatka
- **Path** tip podatka
- **Waveform** tip podatka
- **Dynamic** tip podatka
- **Enum** tip podatka
- **Array** struktura
- **Cluster** struktura

Osnovne strukture i tipovi podataka u LabVIEW

Svakom tipu podataka odgovara druga boja žica (veza)

	Skalar	1D niz	2D niz
Numeric (Floating-point)			
Numeric (Integer)			
Boolean			
String			
Path			
Waveform			
Dynamic			

Numeric tip podatka

Tip podatka zadužen za rad sa broječanim vrednostima

Desni klik na numeric konstantu/control/indicator >> Representation

The image shows a software interface with a numeric control labeled 'Numeric' containing the value '1.23'. A right-click context menu is open, listing various actions. The 'Representation' option is highlighted. Below the menu, a numeric palette is displayed, showing various numeric types and their representations. An arrow points to the 'Representation' option in the menu, and another arrow points to the numeric palette. A third arrow points to the 'Numeric' control in the diagram.

Visible Items
Find Control
Make Type Def.
Hide Control
Change to Indicator
Change to Array
Change to Constant
Description and Tip...
Numeric Palette
Create
Data Operations
Advanced
✓ View As Icon
Representation
Properties

EXT DBL SGL FXP
I64 I32 I16 I8
U64 U32 U16 U8
CXT CDB CSG

Primer:

Numeric tip podatka

Numeric Numeric 2

U donjem delu terminala je označeno koji se tip podatka koristi

Biranje načina reprezentacije brojčane vrednosti

Numeric tip podatka

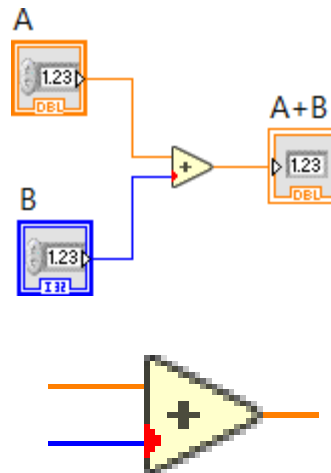
Numeric tip podatka	Oznaka	Veličina zapisa u bitima	Opseg
Extended-precision	EXT	128	-1.19e+4932 do 1.19e+4932
Double-precision	DBL	64	-1.79e+308 do 1.79e+308
Single-precision	SGL	32	-3.40e+38 do 3.40e+38
Fixed-point	FXP	64 ili 72	zavisi od podešavanja
Quad signed integer	I64	64	-1e+19 do 1e+19
Long signed integer	I32	32	-2 147 483 648 do -2 147 483 647
Word signed integer	I16	16	-32 768 do 32 767
Byte signed integer	I8	8	-128 do 127
Quad unsigned integer	U64	64	0 do 2e+19
Long unsigned integer	U32	32	0 do 4 294 967 295
Word unsigned integer	U16	16	0 do 65 535
Byte unsigned integer	U8	8	0 do 255
Complex extended-precision	CXT	256	Opsezi <i>Re</i> i <i>Im</i> dela su kao kod EXT
Complex double-precision	CDB	128	Opsezi <i>Re</i> i <i>Im</i> dela su kao kod DBL
Complex single-precision	CSG	64	Opsezi <i>Re</i> i <i>Im</i> dela su kao kod SGL

Numerička konverzija

Konverzija jednog numeričkog tipa podatka u drugi

Kada se javi potreba za tim, LabVIEW automatski vrši numeričku konverziju

Tamo gde je došlo do konverzije, automatski je obeleženo crvenim trouglom:



NAPOMENA: Pri konverziji realnih na cele brojeve, zaokružuje se na najbliži ceo broj. U slučaju da se broj nalazi na sredini između dva cela broja, zaokružuje se na najbliži paran broj.

Primer zaokruživanja:

6.3 → 6

7.8 → 8

2.5 → 2

3.5 → 4

Boolean tip podatka

Logički tip podatka

Moguća dva stanja:

1. False
2. True

Primer:



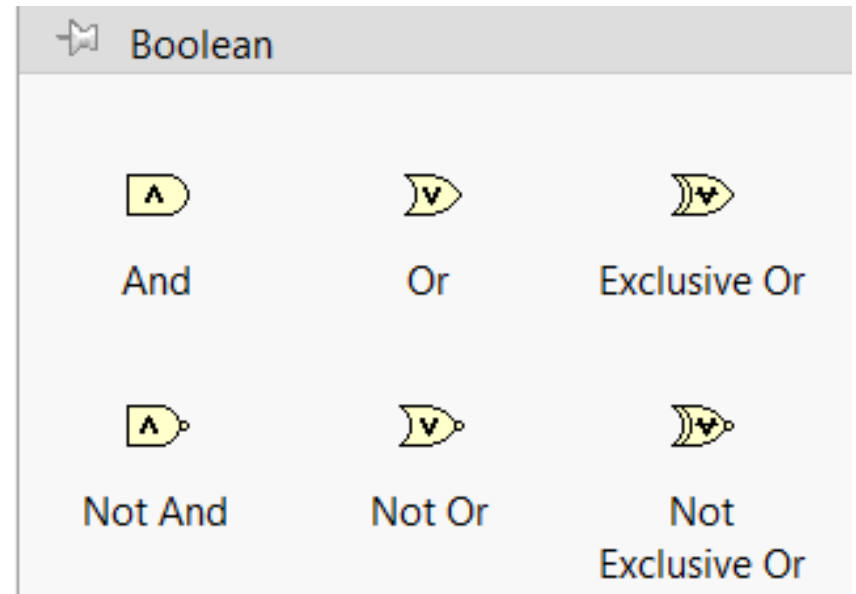
Boolean tip podatka



Prekidač = False
(OFF)



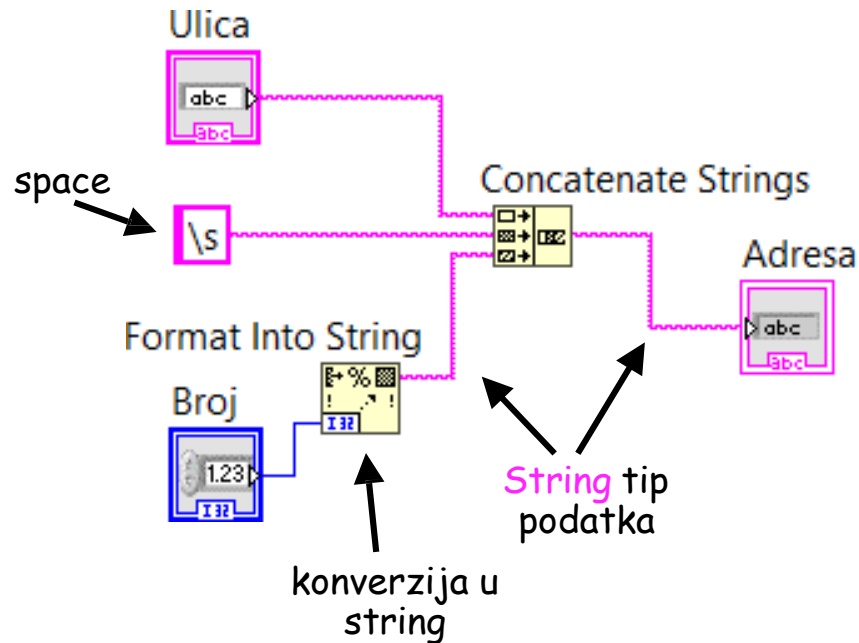
Prekidač = True
(ON)



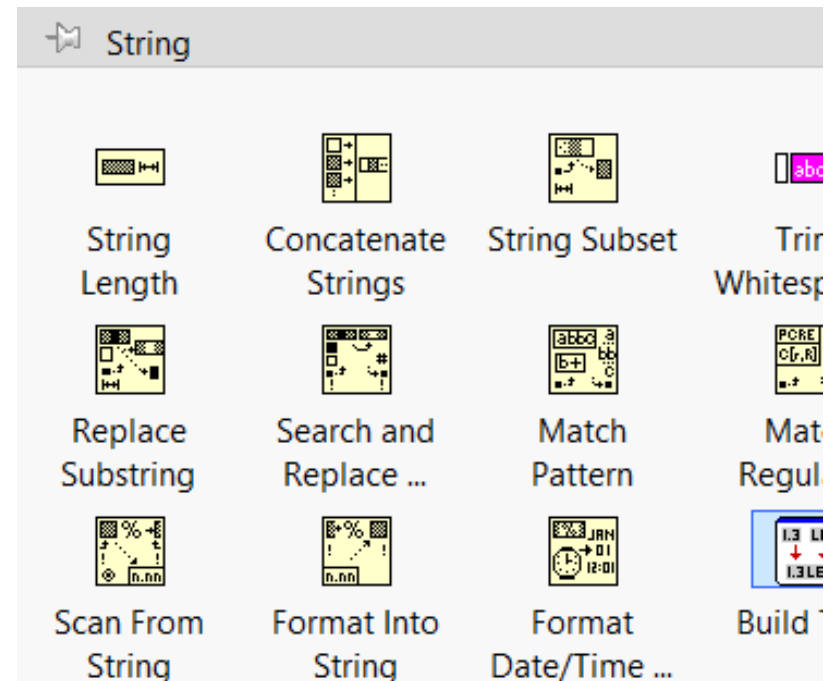
String tip podatka

Tip podatka predviđen za rad sa vidljivim i nevidljivim ASCII karakterima

Primer:



Ulica	Kralja Petra	Adresa
Broj	99	Kralja Petra 99

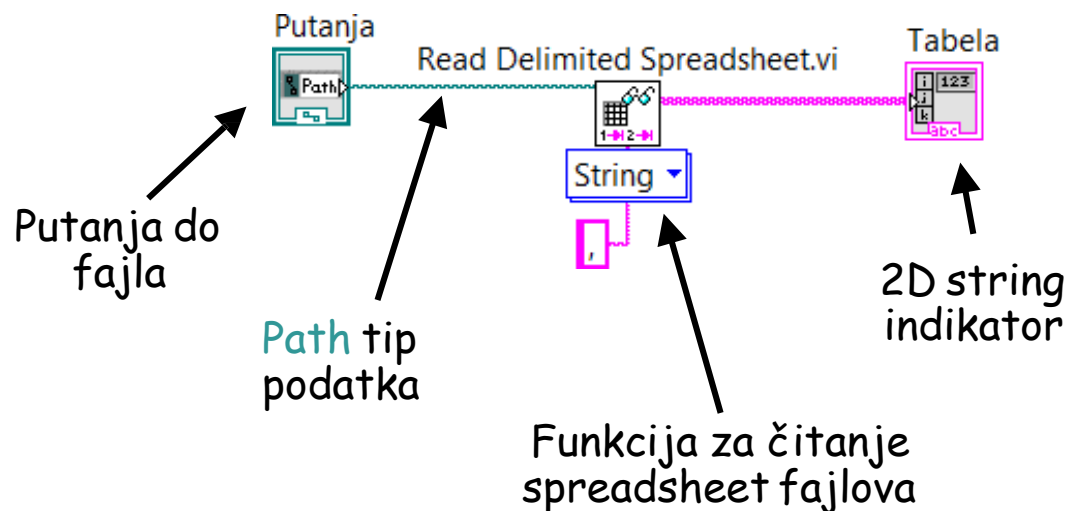


Path tip podatka

Tip podatka zadužen za rad sa lokacijama fajlova na disku

Koristi se kada je potrebno pristupiti nekom fajlu, kada je potrebno sačuvati fajl na određenoj lokaciji i sl.

Primer:



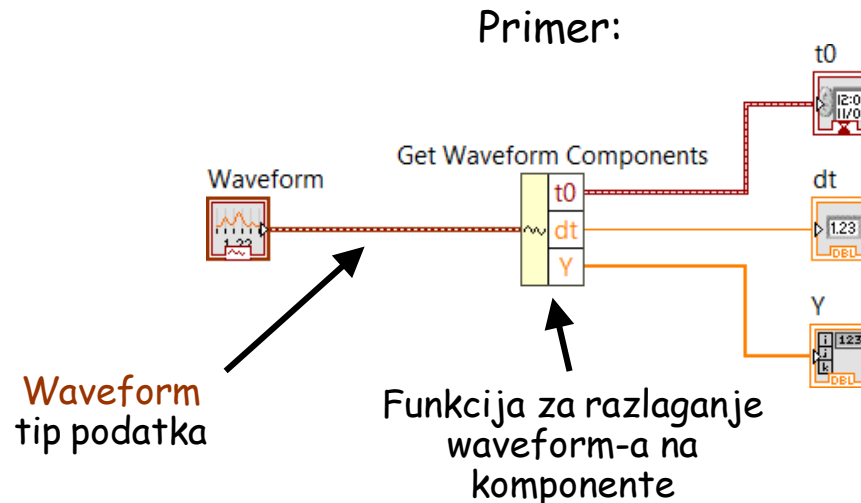
Putanja			
C:\rezultati.csv			
Tabela			
Napon 1	Struja 1	Napon 2	Struja 2
3.560309	3.570601	3.561527	3.574914
3.560556	3.574940	3.563305	3.578446
3.560579	3.575002	3.563329	3.578528
3.560937	3.567944	3.561122	3.572423

Waveform tip podatka

Tip podatka zadužen za rad sa vremenski ekvidistantnim broječanim vrednostima

Waveform tip podatka sadrži tri komponente:

1. **t0** - vremenski trenutak početka signala
2. **dt** - vreme između dva susedna sempla
3. **Y** - 1D niz semplova



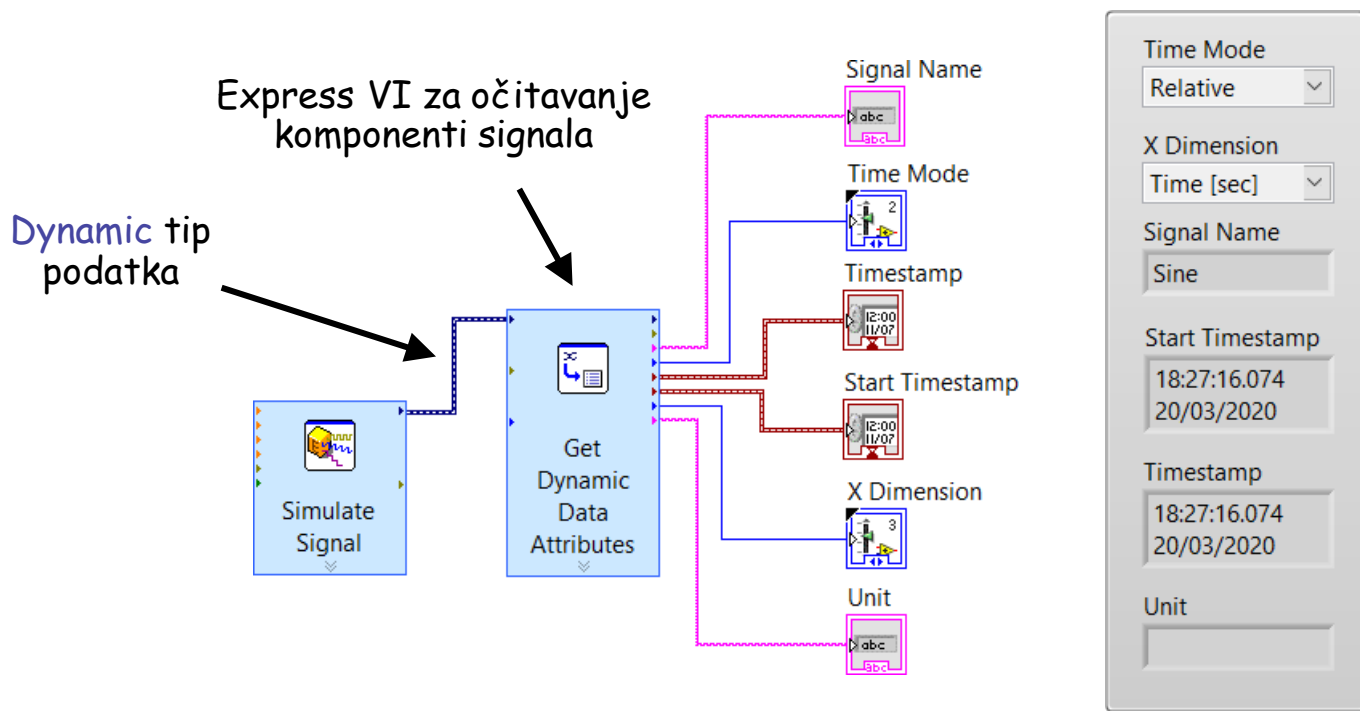
Primer je merenje ili generisanje signala konstantnim sampling rate-om

Dynamic tip podatka

Tip podatka koji se dodeljuje Express VI-u za akviziciju, obradu ili generisanje signala (waveform)

Osim komponenti signala koje nudi Waveform tip podatka, ovde se nalaze još i dodatne informacije: kada je izvršena određena operacija na signalu, naziv signala, fizična jedinica signala i sl.

Primer:

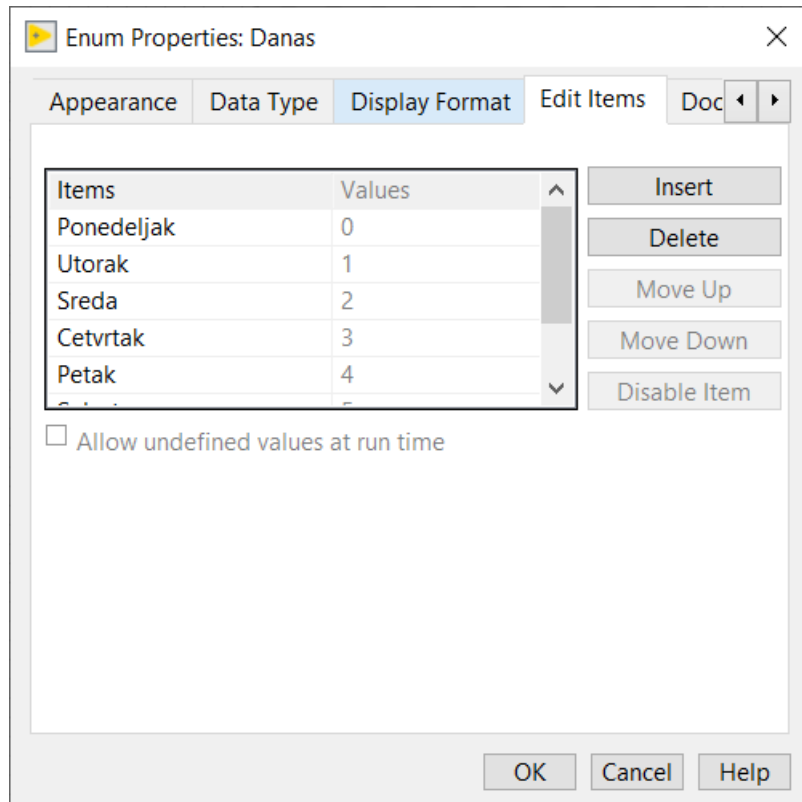


Enum tip podatka

Enumerated (enum) tip podatka je lista elemenata tipa string, gde svaki element ima vrednost tipa numeric

Vrednosti elemenata se beleže kao U8, U16 (default) ili U32 tip podatka.

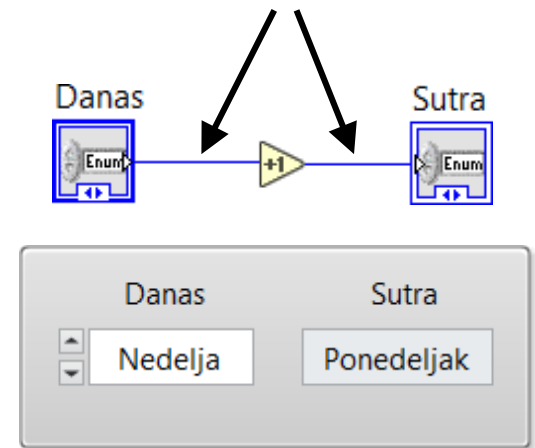
Vrednost prvog elementa je 0, drugog 1, itd. i ne može biti drugačije.



Desni klik na Enum kontrolu/konstantu>> Properties>>Edit Items

Primer:

Enum tip podatka



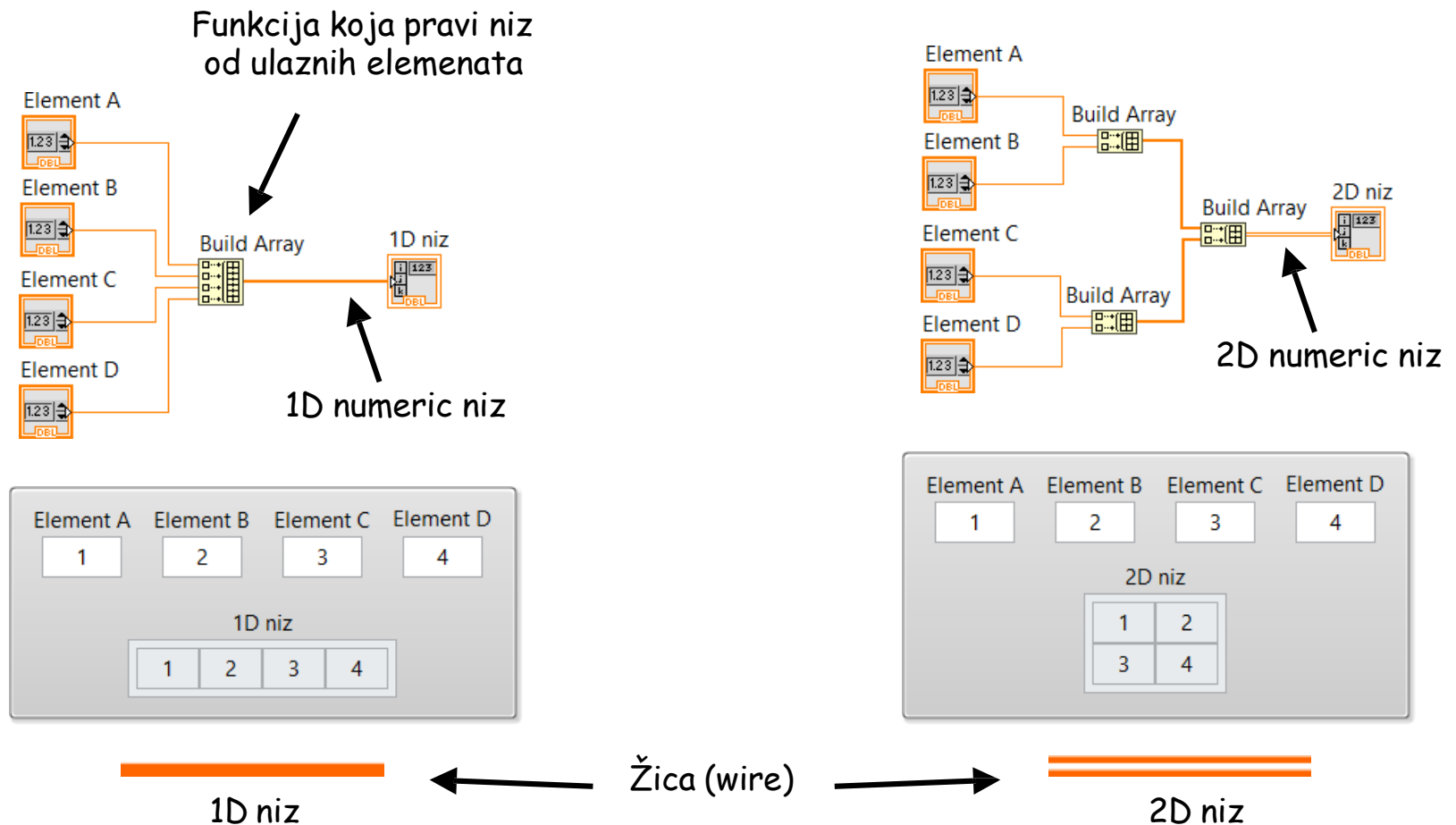
enumerated = nabrojano

Array struktura podatka

Arrays (nizovi) su skup podataka istog tipa

- Svaki niz karakterišu: njegovi elementi i dimenzija
- Moguće je napraviti niz numeric, boolean, string, path, dynamic, waveform ili cluster tipa

Primer:



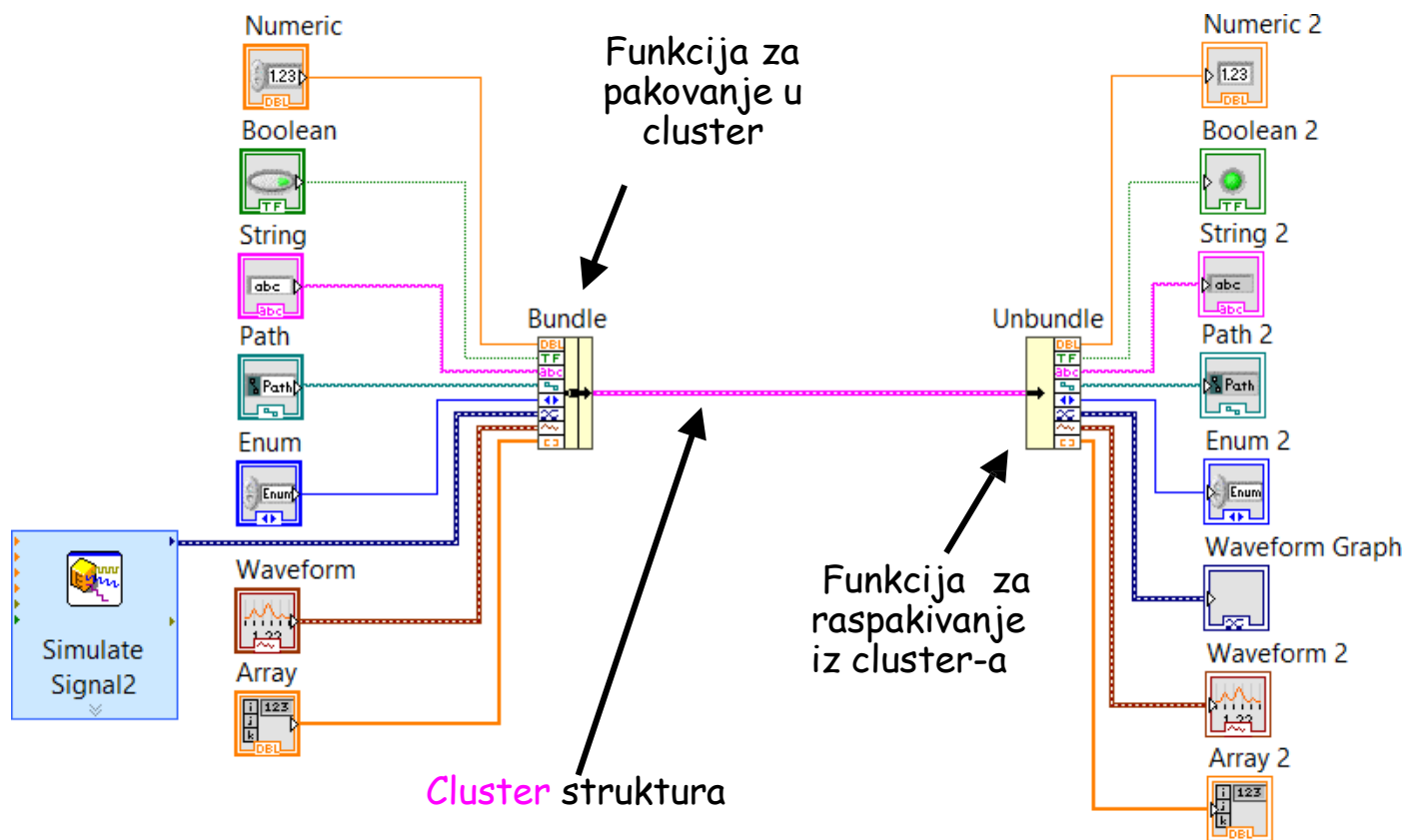
Cluster struktura podatka



Struktura koja grupiše podatke koji mogu biti različitog tipa

Analogno *struct* u programskom jeziku C/C++

Primer:



Zadatak 1

Kreirati projekat pod nazivom *Projekat_Ime_Prezime* i unutar njega dodati novi virtuelni instrument (VI) pod nazivom *Zadatak 1*. Realizovani VI na Front Panel-u treba da sadrži 5 kontrola i 4 indikatora.

Kontrole:

1. Numeric (DBL) pod nazivom *A*
2. Numeric (I32) pod nazivom *B*
3. Boolean (Push Button) pod nazivom *Prekidac*
4. String pod nazivom *Ulazni tekst*
5. Enum kontrolu pod nazivom *Izabran mesec*, sa 12 elementa: Januar, Februar, ..., Decembar.

Indikatori:

1. Numeric (DBL) pod nazivom *C*
2. Boolean (Round LED) pod nazivom *LED Indikator*
3. Numeric (I32) pod nazivom *Duzina stringa*
4. Enum indikator pod nazivom *Trazeni mesec*

Indikator *C* prikazuje vrednost prema jednačini:
$$C = 3 \cdot (A + B)^2 + \sqrt{\frac{A \cdot 4,2}{B}} + (\pi + 6,333)$$

Ako je *Prekidac* uključen, *LED indikator* treba da je ugašen, a ako je *Prekidac* isključen, *LED indikator* treba da je upaljen. Na osnovu stringa koji korisnik unese u *Ulazni string*, indikator *Duzina stringa* treba da ispiše kolika je dužina unetog stringa. Enum indikator *Trazeni mesec* treba da prikaže mesec koji je šesti po redu u odnosu na mesec koji je izabran u enum kontroli *Izabran mesec*.