



Fakulteta za elektrotehniko,  
računalništvo in informatiko

***Inštitut za avtomatiko***

Smetanova ulica 17  
2000 Maribor

Tel.: (02) 220 7000  
Fax: (02) 220 7272  
[www.au.feri.uni-mb.si](http://www.au.feri.uni-mb.si)  
[feri@uni-mb.si](mailto:feri@uni-mb.si)

# Uvod v LabVIEW

---

Gradivo za vaje

Darko HERCOG  
Bojan GERGIČ

Oktober 2010

## Kolofon

CIP - Kataložni zapis o publikaciji  
Univerzitetna knjižnica Maribor

004.42(075.8)(076)

HERCOG, Darko

Uvod v LabVIEW [Elektronski vir] : gradivo za vaje / Darko Hercog, Bojan Gergič. - Besedilni podatki. - Maribor : Fakulteta za elektrotehniko, računalništvo in informatiko, 2010

Način dostopa (URL): <http://dkum.uni-mb.si/>

ISBN 978-961-248-220-6  
1. Gergič, Bojan  
COBISS.SI-ID 64903681

Naslov:	Uvod v LabVIEW: gradivo za vaje
Avtor:	Darko Hercog, Bojan Gergič
Strokovni recenzenti:	Izr. prof. dr. Marjan Golob
Jezikovni recenzenti:	Klavdija Hočevar, prof.
Tipologija publikacije:	2.05 Drugo učno gradivo
Založnik:	UM FERI
Kraj založbe:	Maribor
Datum izida:	11. oktober 2010
Različica:	R 1.0
URL:	<a href="http://dkum.uni-mb.si">http://dkum.uni-mb.si</a>
Sistemske zahteve:	Osebni računalnik
Programske zahteve:	Adobe Reader

## Kazalo

Vaja 2.1: Convert C to F VI .....	1
Besedilo naloge .....	1
Čelna plošča.....	1
Blok diagram.....	1
Vaja 2.2: Editing Exercise VI .....	3
Besedilo naloge .....	3
Čelna plošča.....	5
Blok diagram.....	6
Vaja 2.3: Debug Exercise (Main) VI.....	7
Besedilo naloge .....	7
Vaja 3.1: Convert C to F VI.....	9
Besedilo naloge .....	9
Čelna plošča.....	9
Ikona .....	9
Priključek .....	9
Vaja 3.2: Termometer vi.....	11
Besedilo naloge .....	11
Čelna plošča.....	11
Blok diagram.....	11
Ikona .....	12
Priključek .....	13
Vaja 4.1: Temperature Monitor VI .....	14
Čelna plošča.....	14
Blok diagram.....	14
Nastavitev privzetih vrednosti.....	16
Mehanske lastnosti stikal .....	16
Vaja 4.2: Shift Register Example VI.....	18
Čelna plošča.....	18
Blok diagram.....	18
Vaja 4.3: Temperature Running Average VI .....	19
Čelna plošča.....	19

Blok diagram.....	19
Uporaba grafikonov.....	22
Vaja 5.1: Array Exercise VI.....	25
Čelna plošča.....	25
Blok diagram.....	25
Barva in debelina povezav v blok diagramu.....	26
Vaja 5.2: Temperature Analysis VI.....	28
Čelna plošča.....	28
Blok diagram.....	29
Vaja 5.3: Cluster Exercise VI.....	32
Čelna plošča.....	32
Blok diagram.....	32
Vaja 6.1: Temperature Control VI.....	35
Čelna plošča.....	35
Blok diagram.....	36
Vaja 7.1: Build String VI.....	39
Čelna plošča.....	39
Blok diagram.....	39
Vaja 7.2: Temperature Logger VI.....	42
Čelna plošča.....	42
Blok diagram.....	43
Vaja 8.1: Voltmeter VI.....	46
Čelna plošča.....	46
Blok diagram.....	46
Vaja 8.2: Acquire Waveforms VI.....	49
Čelna plošča.....	49
Blok diagram.....	49
Vaja 8.3: Voltage Output Example.....	52
Čelna plošča.....	52
Blok diagram.....	52
Vaja 8.4: Digital Output.....	56
Čelna plošča.....	56
Blok diagram.....	56
Vaja 8.5: Temperature Control VI.....	59

---

Dodatek A: Paleta orodij .....	60
Dodatek B: Programske strukture .....	62
While loop (Zanka While) .....	62
For Loop (Zanka For).....	62
Case Structure (Programska struktura Case) .....	62
Dodatek C: Priključna plošča NI PCI-6024E .....	64
Dodatek D: Vežalni načrt .....	65
Dodatek E: LabVIEW slovarček.....	66



## Vaja 2.1: Convert C to F VI

### Besedilo naloge

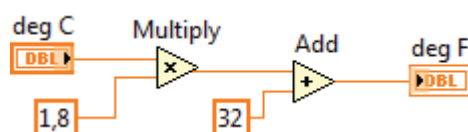
Izdelajte virtualni instrument, ki bo pretvoril temperaturo iz ° Celzija v ° Fahrenheita.

### Čelna plošča



1. Odprite nov VI (File > New VI).
2. Na zaslonu si hkrati prikažite obe okni (čelno ploščo in blok diagram). Uporabite opcijo *Window > Tile Left and Right* ali *Window > Tile Up and Down*.
3. Na čelno ploščo namestite numerično kontrolo. Ta se bo uporabila za vnos temperature v °C.
  - a) Izberite numerično kontrolo (Paleta *Controls > Numeric > Numeric Control*) in jo namestite na čelno ploščo.
  - b) Za ime kontrole (*Label*) vnesite deg C.
4. Na čelno ploščo namestite numerični indikator, ki bo služil za prikaz temperature v ° Fahrenheitov.
  - a) Izberite numerični indikator (Paleta *Controls > Numeric > Numeric Indicator*) in ga prenesite na čelno ploščo.
  - b) Za ime indikatorja (*Label*) vnesite deg F.

### Blok diagram



123

5. Pomaknite se v blok diagram (Window > Show Block Diagram).
6. V blok diagram namestite funkciji *Multiply* in *Add* (paleta *Functions > Programming > Numeric*).
7. V blok diagram namestite dve numerični konstanti (*Numeric Constant*), ki se nahajata na paleti *Functions > Programming > Numeric* in jima podajte začetne vrednosti (1,8 in 32,0).
8. Uporabite orodje za ožičenje in povežite funkcije v blok diagramu, kot je prikazano na zgornji sliki.

9. Preverite delovanje virtualnega instrumenta. Pri 20 °C morate dobiti 68 °F.
10. Shranite virtualni instrument pod imenom *Convert C to F*.



## Vaja 2.2: Editing Exercise VI

### Besedilo naloge

Odprite obstoječ virtualni instrument *Editing Exercise.vi* (slika 1). Modificirajte čelno ploščo in blok diagram tako, kot je prikazano na slikah 2 in 3.

Čelno ploščo virtualnega instrumenta modificirajte s pomočjo sledečih orodij:

#### Orodje

#### Opis



#### Operating Tool (Orodje za delo)

Uporablja se za spreminjanje vrednosti kontrol in konstant.



#### Positioning Tool (Orodje za urejanje)

Uporablja se za urejanje čelne plošče in blok diagrama. Orodje omogoča izbiro, premikanje in spreminjanje velikosti objektov.



#### Labeling Tool (Orodje za pisanje)

Omogoča pisanje tekstov in spreminjanje imen objektov.




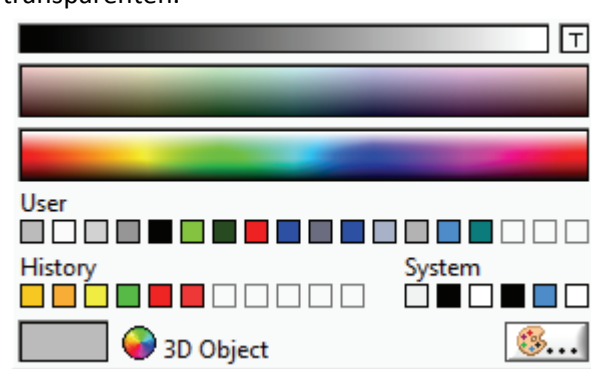
#### Connecting Tool (Orodje za povezovanje)

Omogoča povezovanje objektov v blok diagramu.



#### Coloring Tool (Orodje za barvanje)

Omogoča spreminjanje barv. Z orodjem se pomaknemo na objekt ali del objekta, ki bi mu radi spremenili barvo in kliknemo na desno tipko miške. Ob tem se pojavi barvna paleta (slika spodaj). S kurzorjem miške se pomaknemo na želeno barvo in jo potrdimo s klikom na levo tipko miške. Izбира oznake  v zgornjem desnem kotu barvne palete naredi objekt transparenten.



15pt Application Font ▼

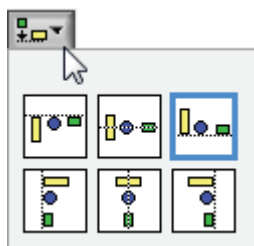
#### Text Setting (Orodje za nastavitve pisave)

Omogoča spreminjanje tipa, velikosti in barve izbranega teksta. Z orodjem za pisanje (*Labeling Tool*) se najprej izbere tekst ali del teksta, ki bi ga radi spremenili, nato pa se uporabi želeni opcija tega padajočega menija.

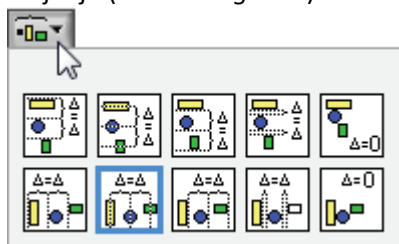


#### Align Objects (Orodje za poravnavo objektov)

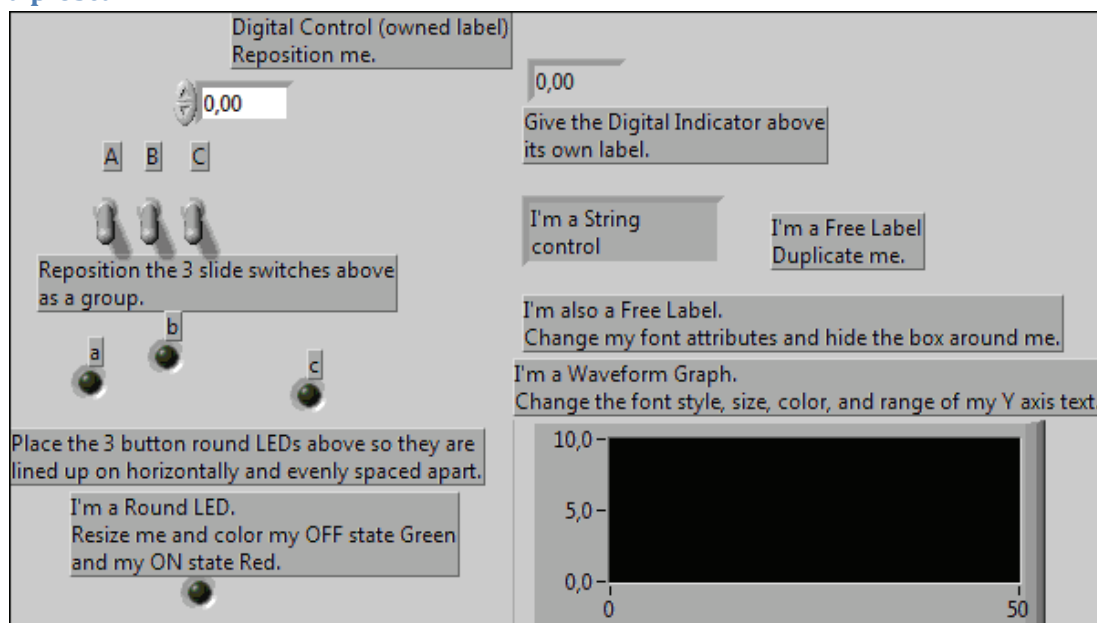
Omogoča različne poravnave dveh ali več objektov čelne plošče ali blok diagrama. Pred poravnavo je potrebno objekte izbrati z orodjem za urejanje (*Positioning Tool*).

**Distribute Objects (Orodje za enakomerno porazdelitev objektov)**

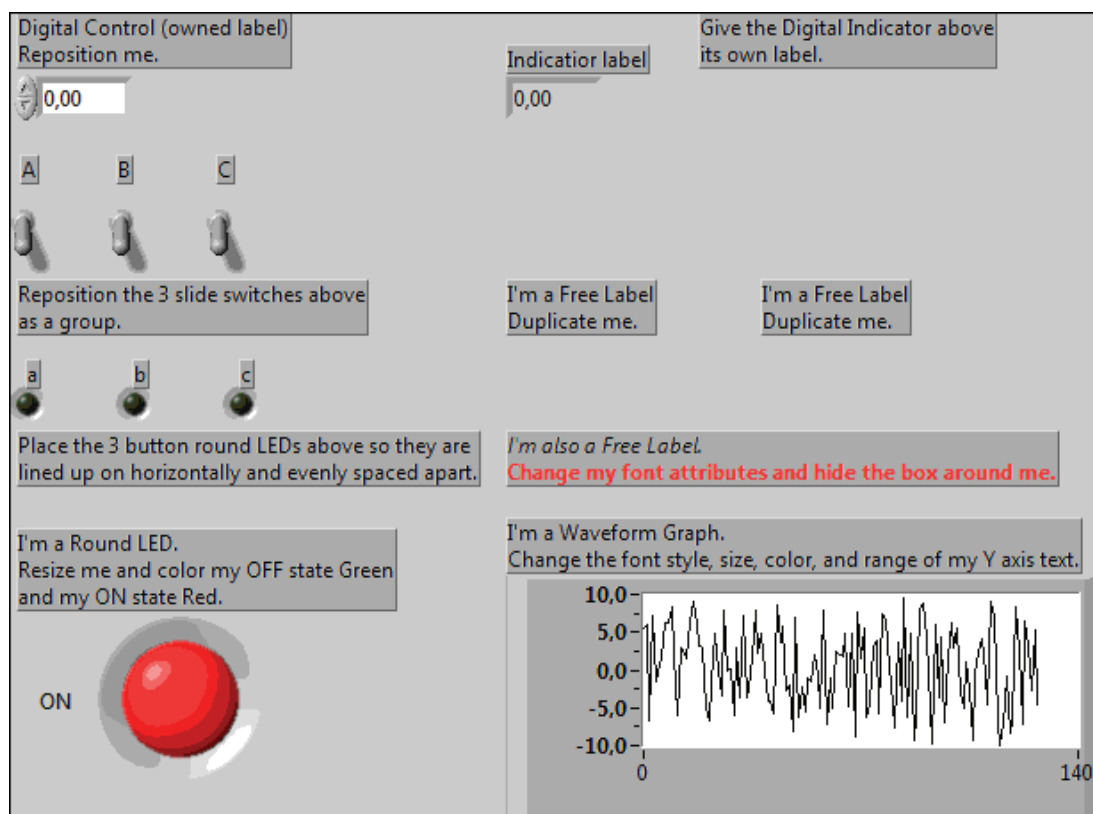
Omogoča enakomerno porazdelitev dveh ali več objektov čelne plošče ali blok diagrama. Za izbiro objektov je potrebno uporabiti orodje za urejanje (*Positioning Tool*).



## Čelna plošča

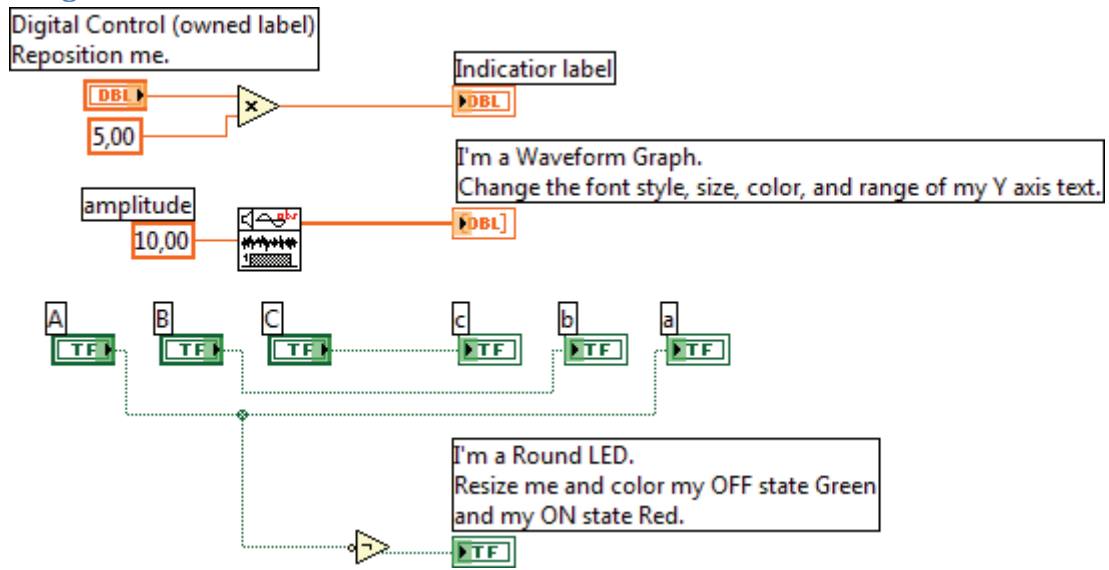


Slika 1: Originalna čelna plošča virtualnega instrumenta.



Slika 2: Modificirana čelna plošča virtualnega instrumenta.

## Blok diagram



Slika 3: Blok diagram

## Vaja 2.3: Debug Exercise (Main) VI

### Besedilo naloge

Odprite virtualni instrument *Debug Exercise (Main).vi* in s pomočjo orodij za iskanje napak odkrijte in odpravite vse napake, ki jih vsebuje virtualni instrument.

Pri iskanju napak si pomagajte s sledečimi orodji:

#### Orodje

#### Opis



#### Highlight Execution (Poudarjeno izvajanje)

Pri poudarjenem izvajanju LabVIEW v blok diagramu grafično ponazarja, kateri del kode trenutno izvaja. Takšen način izvajanja kode je bistveno počasnejši od normalnega načina.



Poudarjeno izvajanje izključeno.



Poudarjeno izvajanje vključeno.



#### Breakpoint (Prekinitvena točka)

Orodje omogoča začasno zaustavitev izvajanja virtualnega instrumenta. Prekinitvena točka se uporablja samo v blok diagramu. Ob kliku na povezavo se na njej pojavi rdeča pika (slika spodaj), ki ponazori, da se bo ob izvajanju tega dela kode virtualni instrument začasno zaustavil. Prekinitvena točka se odstrani tako, da se z istim orodjem ponovno klikne na povezavo, na kateri se prekinitvena točka že nahaja, ali z uporabo orodja *Breakpoint Manager*.


Dodajanje  
prekinitvene  
točke



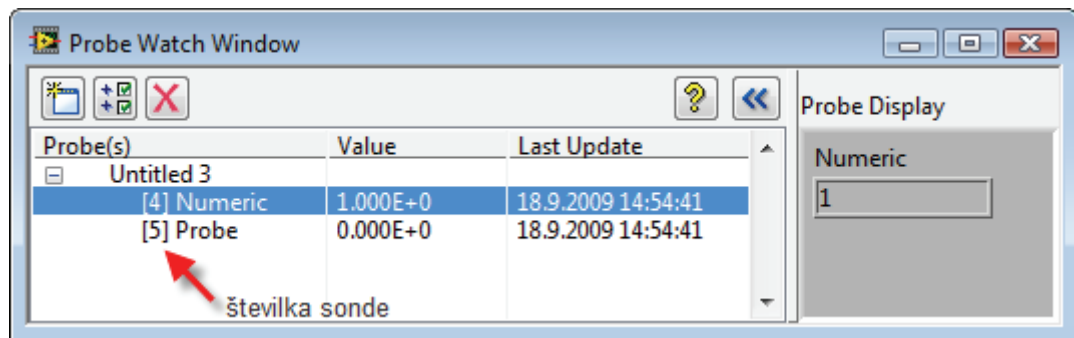
Odstranitev  
prekinitvene  
točke



#### Probe Data (Sonda)

Sonda omogoča prikaz podatka, ki se prenese po izbrani povezavi v blok diagramu. Sonda se uporablja samo v blok diagramu, namestiti pa jo je možno le na povezavo. Ob kliku na povezavo se pojavi dodatno okno (*Probe Watch Window*), v katerem se prikažejo vse uporabljene sonde v virtualnem instrumentu. Ob izbiri ustrezne sonde se sonda in njena vrednost prikažeta v indikatorju *Probe Display*. Za lažje ločevanje sond se ob namestitvi sonde na povezavi pojavi številka sonde. Sonda se odstrani tako, da se v oknu *Probe Watch Window* izbere ustrezna sonda, nato pa se izbere gumb *Remove Selected Probe* .





### Single Step Execution (Izvajanje po korakih)

Orodje omogoča izvajanje virtualnega instrumenta po korakih. Pri tem načinu delovanja LabVIEW izvede samo eno operacijo virtualnega instrumenta, nato pa se izvajanje začasno zaustavi. Izvajanje je tako podobno, kot da bi bile na vseh povezavah v blok diagramu nameščene prekinitvene točke.

Izvajanje virtualnega instrumenta po korakih se zažene s klikom na *Step Into* ali *Step Over* gumb na orodni vrstici v blok diagramu.



#### Step Into

S klikom na ta gumb se zažene izvajanje virtualnega instrumenta po korakih. LabVIEW izvede eno operacijo virtualnega instrumenta, nato pa se začasno zaustavi. Nov korak se izvede s ponovnim klikom na ta gumb ali pa na gumb *Step Over*.



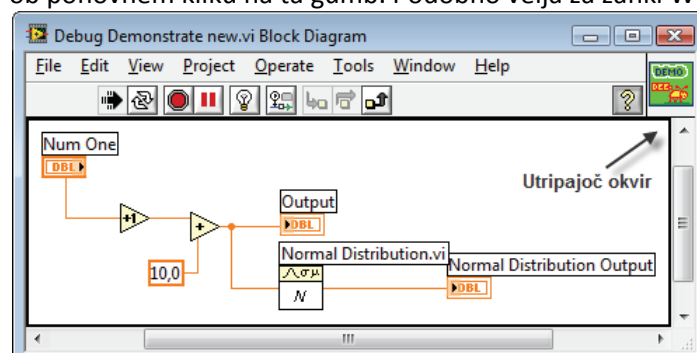
#### Step Over

Ob izbiri opcije *Step Over* LabVIEW izvede eno operacijo virtualnega instrumenta, nato pa se začasno zaustavi. Tekom izvajanja virtualnega instrumenta po korakih ima ta opcija podoben pomen kot *Step Into*. Do razlike v delovanju pride v primeru podprograma (SubVI). Ob izbiri te opcije LabVIEW preskoči podprogram, ob izbiri opcije *Step Into* pa prikaže podprogram in tudi tega začne izvajati po korakih.



#### Step Out

S klikom na ta gumb LabVIEW preskoči izvajanje po korakih na zadnji korak. Ob tem začne okrog blok diagrama utripati črn okvir, ki ponazuje, da gre za zadnji korak pri izvajanju po korakih. Izvajanje po korakih se dokončno zaključi ob ponovnem kliku na ta gumb. Podobno velja za zanki *While* in *For*.



## Vaja 3.1: Convert C to F VI

### Besedilo naloge

Virtualnemu instrumentu *Convert C to F* izdelajte ikono in priključke.

### Čelna plošča



### Ikona

1. Odprite virtualni instrument *Convert C to F*.
2. Pomaknite se na ikono virtualnega instrumenta (nahaja se v desnem zgornjem kotu čelne plošče). Kliknite na desni gumb miške in v priročnem meniju izberite opcijo *Edit Icon*.
3. V urejevalniku ikon dvokliknite na orodje za izbiro (*Select tool*) in izbrala se bo celotna ikona. Vsebino privzete ikone izbrišite s pritiskom tipke *Delete*.
4. Dvokliknite na orodje za kreiranje okvirjev in avtomatsko se bo ustvaril okvir ikone.
5. Kreirajte ikono, prikazano na spodnji sliki. Ob tem si pomagajte s sledečimi orodji:

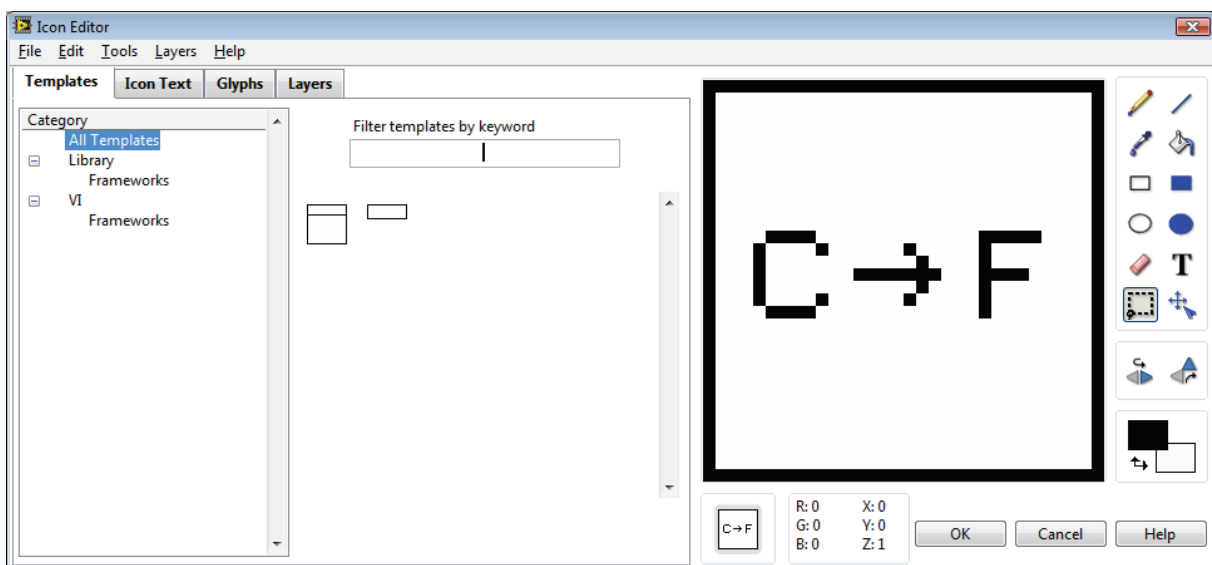


a) Črki C in F napišite z orodjem za pisanje.



b) Puščico narišite z uporabo pisala ali črtala.

c) Kreiranje ikone zaključite s klikom na gumb OK.



### Priključek

6. Na čelni plošči se pomaknite na ikono virtualnega instrumenta. Kliknite na

desno tipko miške in v priročnem meniju izberite opcijo *Show Connector*.  
OPOZORILO: Opcija *Show Connector* se v priročnem meniju ne pojavi, če se nahajate na ikoni v blok diagramu!



7. Namesto ikone se prikaže privzeti priključitveni vzorec.



8. Pomaknite se na priključitveni vzorec in kliknite na desno tipko miške. V priročnem meniju izberite opcijo *Patterns*, nato pa izberite priključitveni vzorec, ki ima priključek na levi in na desni strani.



9. Levi priključek povežite s kontrolno *deg C*, desni pa z indikatorjem *deg F* čelne plošče. Ob pravilno kreirani povezavi se priključka obarvata v oranžno barvo.

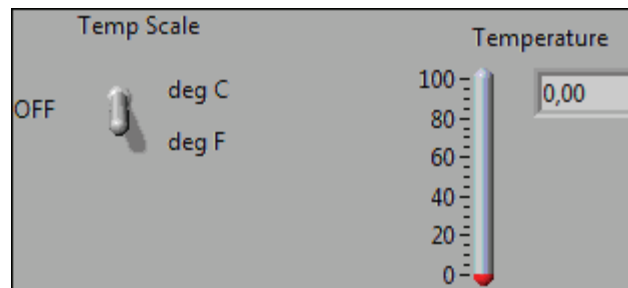


## Vaja 3.2: Termometer vi

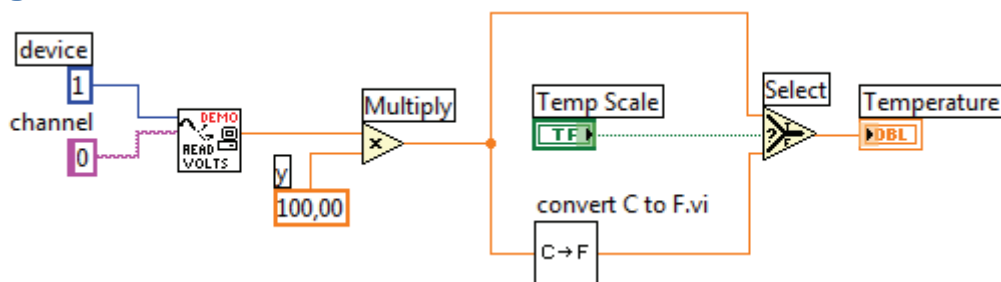
### Besedilo naloge

Izdelajte virtualni instrument, ki bo izmeril temperaturo in jo prikazal v indikatorju na čelni plošči. Na njej naj se nahaja stikalo za izbiro enote. Če bo stikalo vključeno, naj se v indikatorju izpiše vrednost temperature v °C, v nasprotnem primeru pa v °F.

### Čelna plošča



### Blok diagram



#### (Demo) Read Voltage

**Paleta:** Functions > User Libraries > Basic 1 Course



**Opis:** Funkcija da na izhodu simulirano vrednost napetosti, ki je proporcionalna temperaturi. Napetost 10 mV predstavlja temperaturo 1 °C.



#### OPOZORILO:

1. V ikoni funkcije se mora nahajati napis DEMO!
2. Konstanti na vhodu funkcije kreirajte tako, da se z orodjem za ožičenje pomaknete na izbran vhod, kliknete na desno tipko miške in v priročnem meniju izberete opcijo *Create > Constant*.



#### Multiply (Množilnik)

**Paleta:** Functions > Programming > Numeric



#### Numeric Constant (Numerična konstanta)

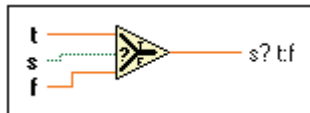
**Paleta:** Functions > Programming > Numeric

Vrednosti konstante tekom izvajanja virtualnega instrumenta ni možno spreminjati.



#### Select

**Paleta:** Functions > Programming > Comparison

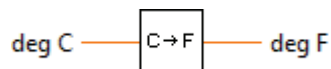


**Opis:** Funkcija ima 3 vhode (**t**, **s** in **f**) in en izhod (**s? t:f**). Delovanje funkcije je odvisno od logične (Boolean) vrednosti, pripeljane na vhod **s**. Če je vhod **s** enak TRUE, da funkcija na izhod vrednost vhoda **t**, v nasprotnem primeru pa vrednost vhoda **f**.



### Convert C to F

**Paleta:** Functions > Select a VI... (vaja 2.1)



**Opis:** Funkcija pretvori temperaturo iz ° Celzija v ° Fahrenheita.

## Ikona

1. Izbrišite privzeto ikono:

a) Pomaknite se na ikono virtualnega instrumenta (nahaja se v desnem zgornjem kotu čelne plošče). Kliknite na desno tipko miške in v priročnem meniju izberite opcijo *Edit Icon*.

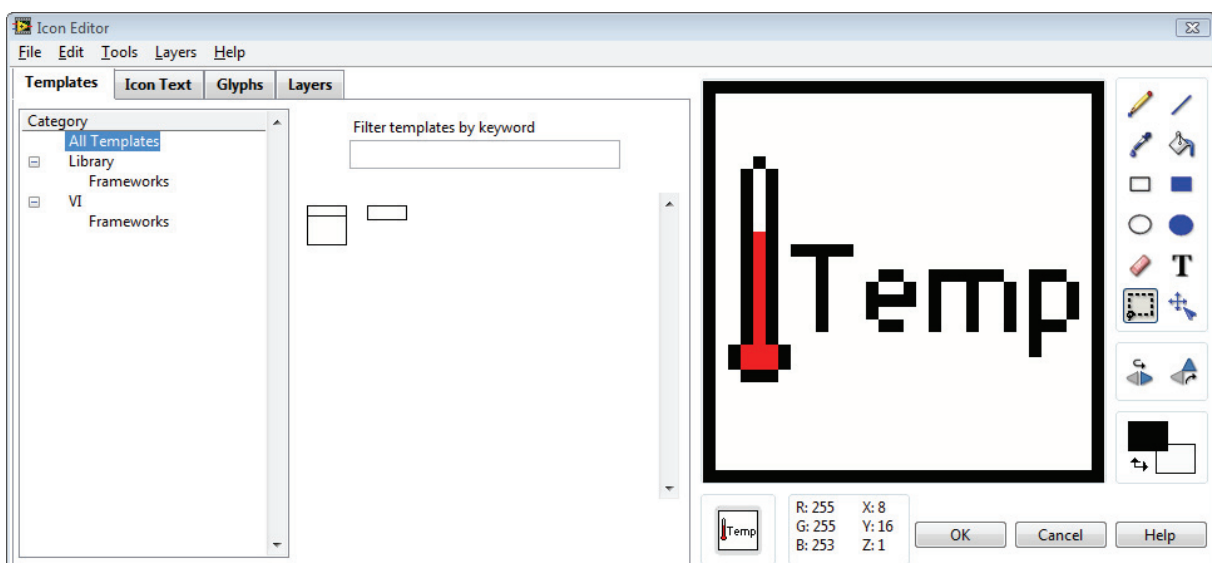


b) V urejevalniku ikon dvokliknite na orodje za izbiro (*Select tool*) in izbrala se bo celotna ikona. Vsebinsko privzete ikone izbrišite s pritiskom tipke *Delete*.



c) Dvokliknite na orodje za risanje okvirjev in avtomatsko se bo ustvaril okvir ikone.

2. Narišite ikono, kot je prikazana na spodnji sliki.



## Priključek

1. Na čelni plošči se pomaknite na ikono virtualnega instrumenta. Kliknite na desno tipko miške in v priročnem meniju izberite opcijo *Show Connector*.  
OPOZORILO: Opcija *Show Connector* se v priročnem meniju pojavi le, če se nahajate na ikoni na čelni plošči!



2. Pomaknite se na priključitveni vzorec in kliknite na desno tipko miške. V priročnem meniju izberite opcijo *Patterns*, nato pa izberite priključitveni vzorec, ki ima en priključek na levi in en na desni strani.

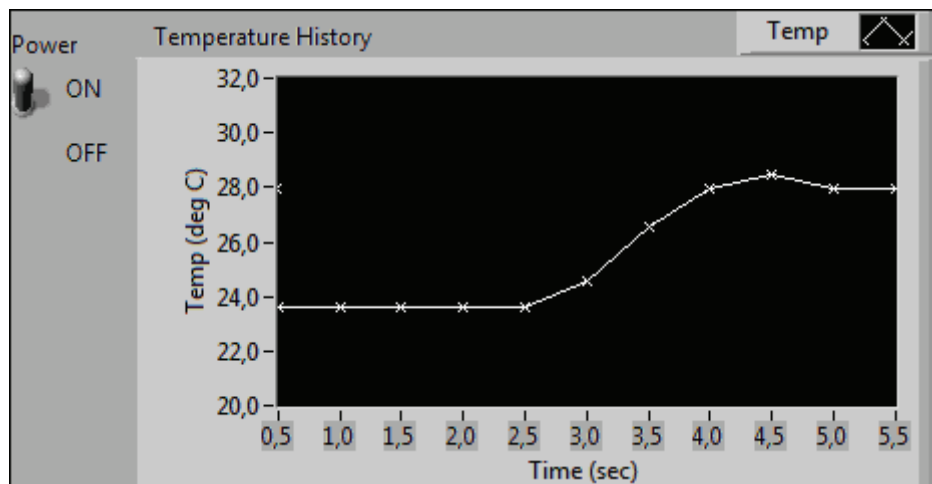


3. Levi priključek povežite s stikalom *Temp Scale*, desni pa z indikatorjem *Temperature*. Ob pravilno izdelani povezavi se levi priključek obarva v zeleno, desni pa v oranžno barvo.

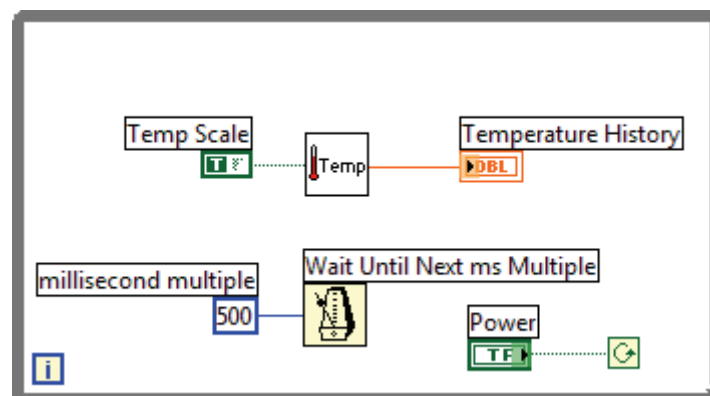
## Vaja 4.1: Temperature Monitor VI

Izdelajte virtualni instrument, ki bo ponavljal meritev temperature in jo sproti prikazoval v grafikonu na čelni plošči. Virtualni instrument naj se ustavi ob izklopu stikala *Power*. To naj ima privzeto vrednost *TRUE* in mehansko lastnost tipa *Latch When Pressed*.

### Čelna plošča





### Blok diagram





### While loop (Zanka While)


**Paleta:** Functions > Programming > Structures



**Opis:** Zanka While omogoča ponavljajoče izvajanje dela kode, ki se nahaja znotraj območja zanke. Zanka While se izvaja, dokler je izpolnjen določen pogoj, in se vedno izvede vsaj enkrat. Na pogojni priključek (  ali  ) mora biti pripeljan podatek tipa Boolean. Glede na nastavitve pogojnega priključka in vhodnega podatka velja sledeče:

- Če je pogojni priključek nastavljen na  (Stop if True), se izvajanje zanke ustavi, če je vhodni podatek enak TRUE, sicer se izvajanje zanke nadaljuje;

- Če je pogojni priključek nastavljen na  (Continue if True), se izvajanje zanke nadaljuje, če je vhodni podatek enak TRUE, in konča, ko je enak FALSE.

Priključek  predstavlja števec iteracij oz. ponovitev zanke in teče od 0 naprej.



### Thermometer

**Paleta:** Functions > Select a VI... > Thermometer.vi (vaja 3.2)



**Opis:** Funkcija na izhodu vrne vrednost izmerjene (simulirane) temperature v °C ali °F. Skala termometra se določi z vhodnim Boolean podatkom *Temp Scale*.

Temp Scale	Temperature
TRUE	°C
FALSE	°F



### True Constant (Konstanta tipa Boolean)

**Paleta:** Functions > Programming > Boolean

**Opis:** Boolean konstanta, privzeto nastavljena na vrednost TRUE.



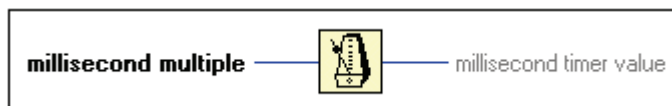
### Numeric Constant (Numerična konstanta)

**Paleta:** Functions > Programming > Numeric



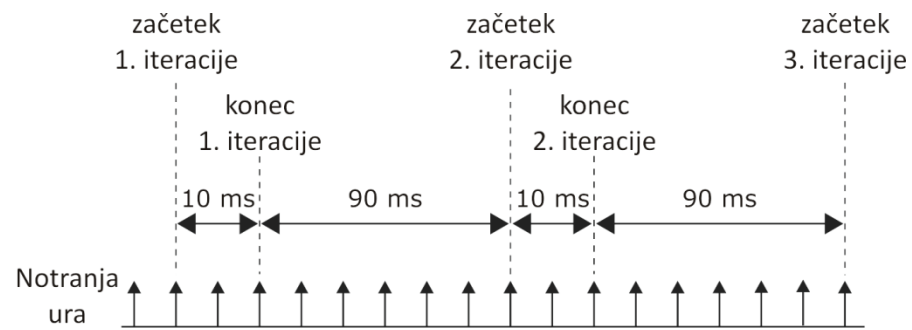
### Wait Until Next ms Multiple (Metronom)

**Paleta:** Functions > Programming > Timing



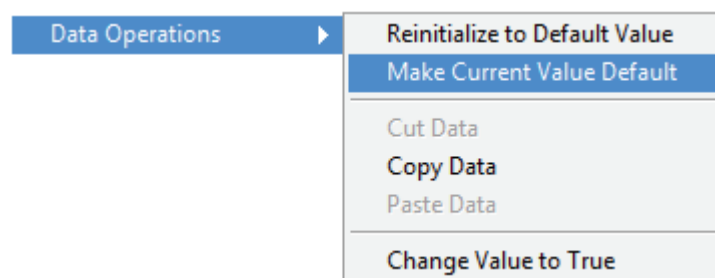
**Opis:** Funkcija *Wait Until Next ms Multiple* se najpogosteje uporablja za časovni nadzor *While* zank. Funkcija omogoča sinhronizacijo izvajanja *While* zank in hkrati onemogoča preveliko uporabo procesorja. V primeru uporabe te funkcije znotraj *While* zanke se koda zanke izvede na vsak celoštevilčni mnogokratnik vhodnega podatka (*millisecond multiple*), ki je definiran v ms.

**Primer:** VI vsebuje *While* zanko s kodo, ki jo LabVIEW izvede v 10 ms. Zanka vsebuje funkcijo *Wait Until Next ms Multiple*, na katero je pripeljana vrednost 100. LabVIEW bo tako izvedel algoritem *While* zanke ob vsakem celoštevilčnem mnogokratniku 100 ms (100 ms, 200 ms, 300 ms, itd.). Preostalih 90 ms je procesor na voljo za druga opravila.



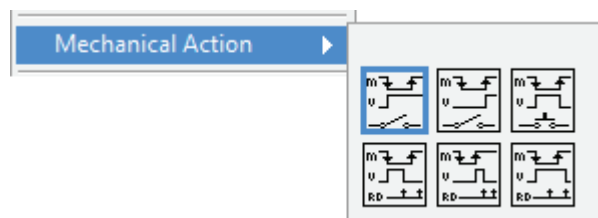
### Nastavitev privzetih vrednosti

Kontrolam čelne plošče lahko nastavimo privzeto vrednost. Le-to naredimo tako, da kontroli najprej nastavimo želeno vrednost, nato pa v priročnem meniju kontrole izberemo opcijo: *Data Operations > Make Current Value Default*.



### Mehanske lastnosti stikal

LabVIEW omogoča nastavitve mehanskih lastnosti stikalom čelne plošče. Le-te nastavimo tako, da v priročnem meniju stikala izberemo opcijo *Mechanical Action*.



#### Tip stikala

#### Opis



#### Switch When Pressed

Vrednost stikala se spremeni ob vsakokratnem pritisku leve tipke miške.



#### Switch When Released

Vrednost stikala se spremeni ob vsakokratni sprostitvi leve tipke miške.



#### Switch Until Released

Vrednost stikala se spremeni ob pritisku leve tipke miške in ostane na novi vrednosti vse do njene sprostitve.



#### Latch When Pressed

Vrednost stikala se spremeni ob pritisku leve tipke miške. Stikalo ostane v novem položaju, dokler LabVIEW nove vrednosti stikala ne prebere, in ga šele nato postavi nazaj na privzeto vrednost.

**Latch When Released**

Vrednost stikala se spremeni ob sprostitvi leve tipke miške. Stikalo ostane v novem položaju, dokler LabVIEW nove vrednosti stikala ne prebere, in ga šele nato postavi nazaj na privzeto vrednost.

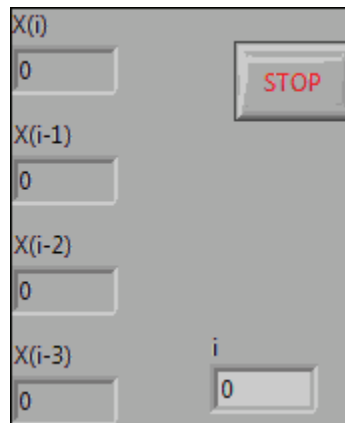
**Latch Until Released**

Vrednost stikala se spremeni ob pritisku leve tipke miške. Stikalo ostane v novem položaju do sprostitve leve tipke miške in se vrne na privzet položaj, ko LabVIEW prebere njegovo vrednost.

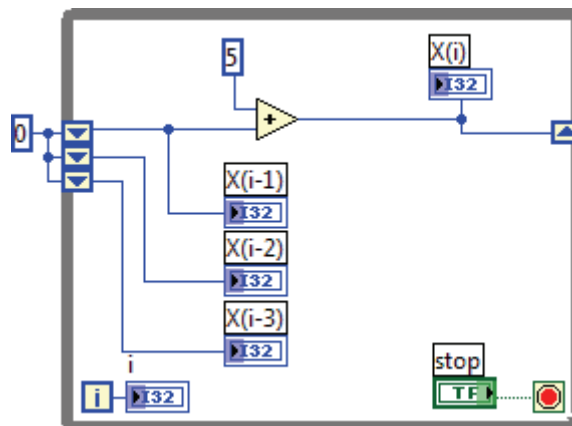
## Vaja 4.2: Shift Register Example VI

Odprite predpripravljen virtualni instrument *Shift Register Example.vi* in preverite delovanje pomikalnega registra (Shift Register). Program izvajajte po korakih s poudarjenim izvajanjem (*Highlight Execution*).

### Čelna plošča



### Blok diagram

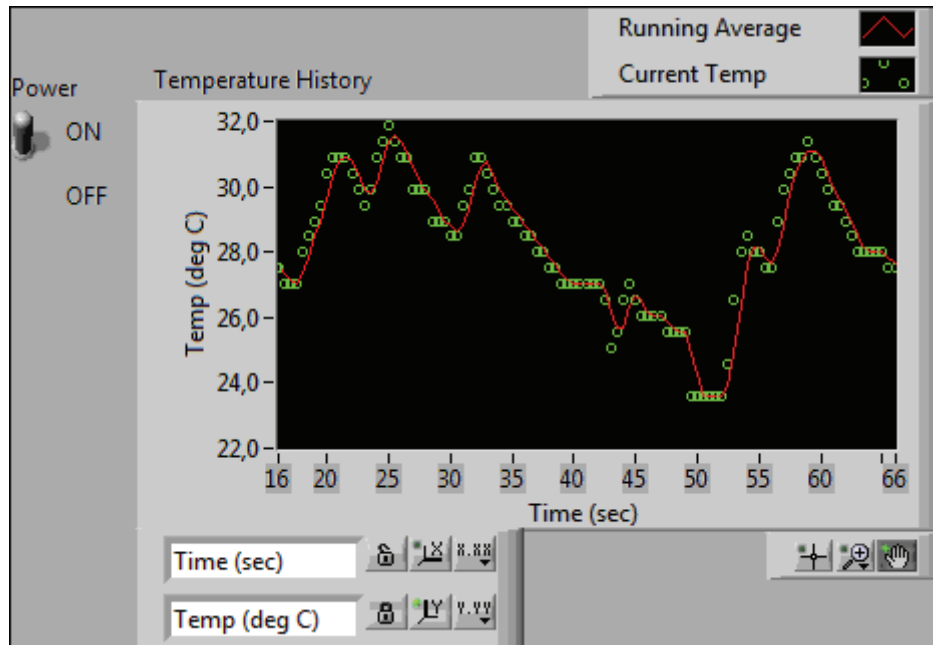




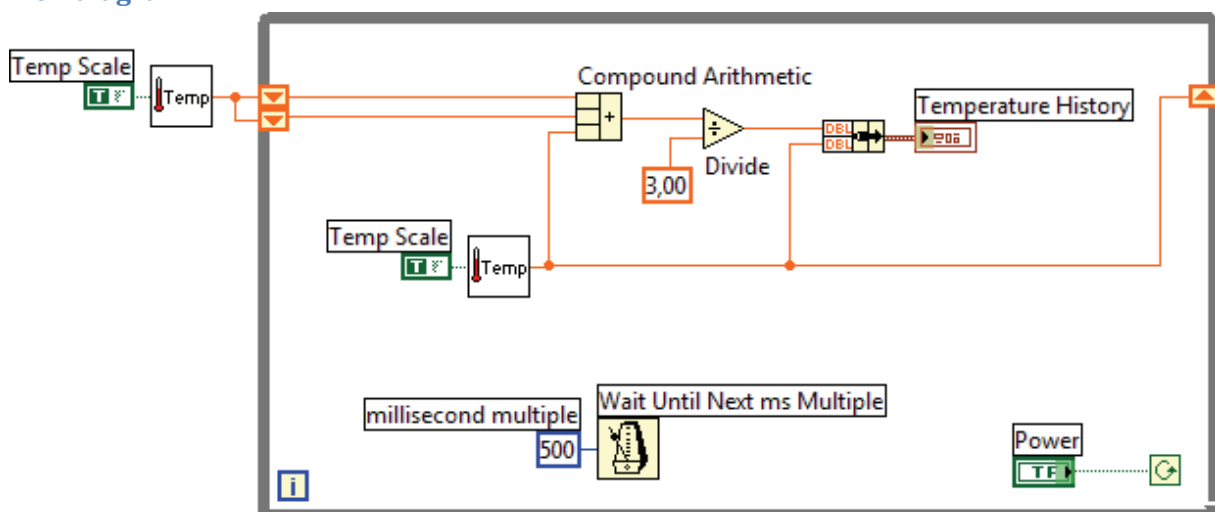
## Vaja 4.3: Temperature Running Average VI

Nadgradite virtualni instrument *Temperature Monitor* z izračunom povprečne vrednosti zadnjih treh izmerjenih temperatur in prikazom le-te v grafikonu čelne plošče.

### Čelna plošča



### Blok diagram



#### Thermometer

**Paleta:** Functions > Select a VI... > Thermometer.vi (iz vaje 3.2)



**Opis:** Funkcija na izhodu vrne vrednost izmerjene (simulirane) temperature v °C ali °F. Skala termometra se določi z vhodnim Boolean podatkom *Temp Scale*.

Temp Scale	Temperature
TRUE	°C
FALSE	°F

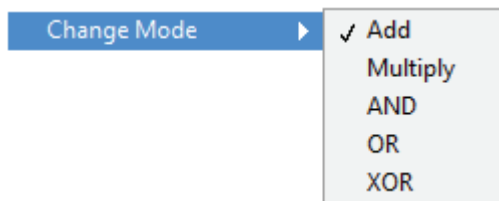


### Compound Arithmetic

**Paleta:** Functions > Programming > Numeric

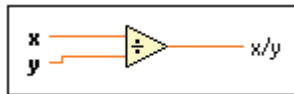


**Opis:** Funkcija izvede aritmetično operacijo (seštevanje, množenje, logični AND, logični OR ali logični XOR) vhodnih podatkov. Privzeto je izbrana operacija seštevanja (*Add*). Aritmetično operacijo je možno spremeniti v priročnem meniju funkcije pod opcijo *Change Mode*.



### Divide (Delilnik)

**Paleta:** Functions > Programming > Numeric

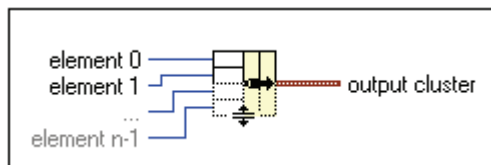


**Opis:** Funkcija izračuna kvocient vhodnih podatkov ( $x/y$ ).



### Bundle (Grupiranje)

**Paleta:** Functions > Programming > Cluster, Class & Variant



**Opis:** Funkcija *Bundle* programsko kreira gručo vhodnih podatkov. Vhodni priključki funkcije se avtomatsko obarvajo glede na tip vhodnega podatka.



### Numeric Constant (Numerična konstanta)

**Paleta:** Functions > Programming > Numeric



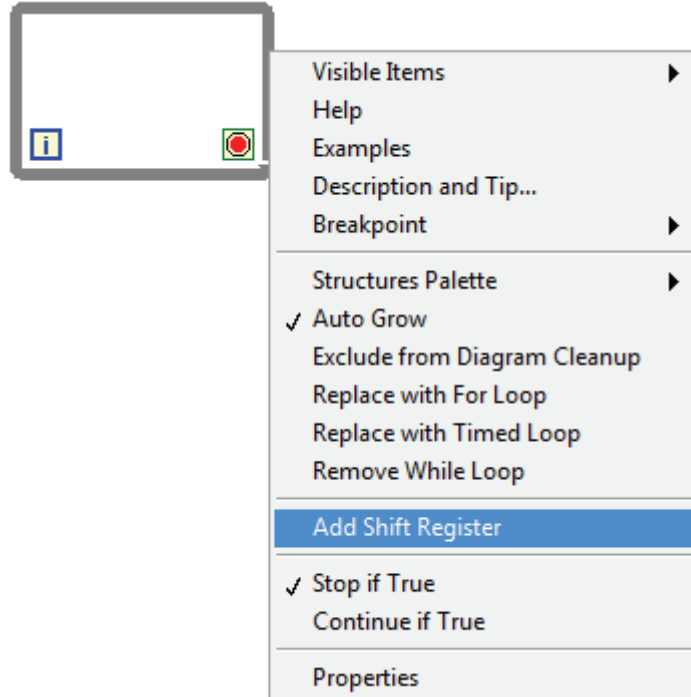
### Shift Register (Pomikalni register)

Pomikalni register se uporablja pri For ali While zankah in omogoča dostop do podatkov iz prejšnjih iteracij zanke. Vrednost, ki je pripeljana na desno stran pomikalnega registra, se v naslednji iteraciji zanke pojavi na levi strani pomikalnega registra.

#### Dodajanje pomikalnih registrov

Pomikalni register dodamo na zanko tako, da se s kurzorjem miške pomaknemo na

rob zanke, nato pa v priročnem meniju zanke izberemo opcijo *Add Shift Register*.



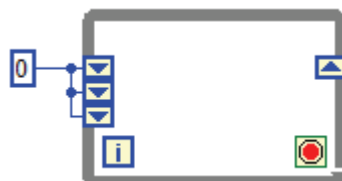
### Dodajanje elementov pomikalnim registrom

Dodatne elemente pomikalnemu registru dodamo tako, da se s kurzorjem miške pomaknemo na obstoječ pomikalni register, nato pa v priročnem meniju izberemo opcijo *Add Element*. Ti dodatni elementi omogočajo dostop do podatkov za več prejšnjih iteracij zanke.



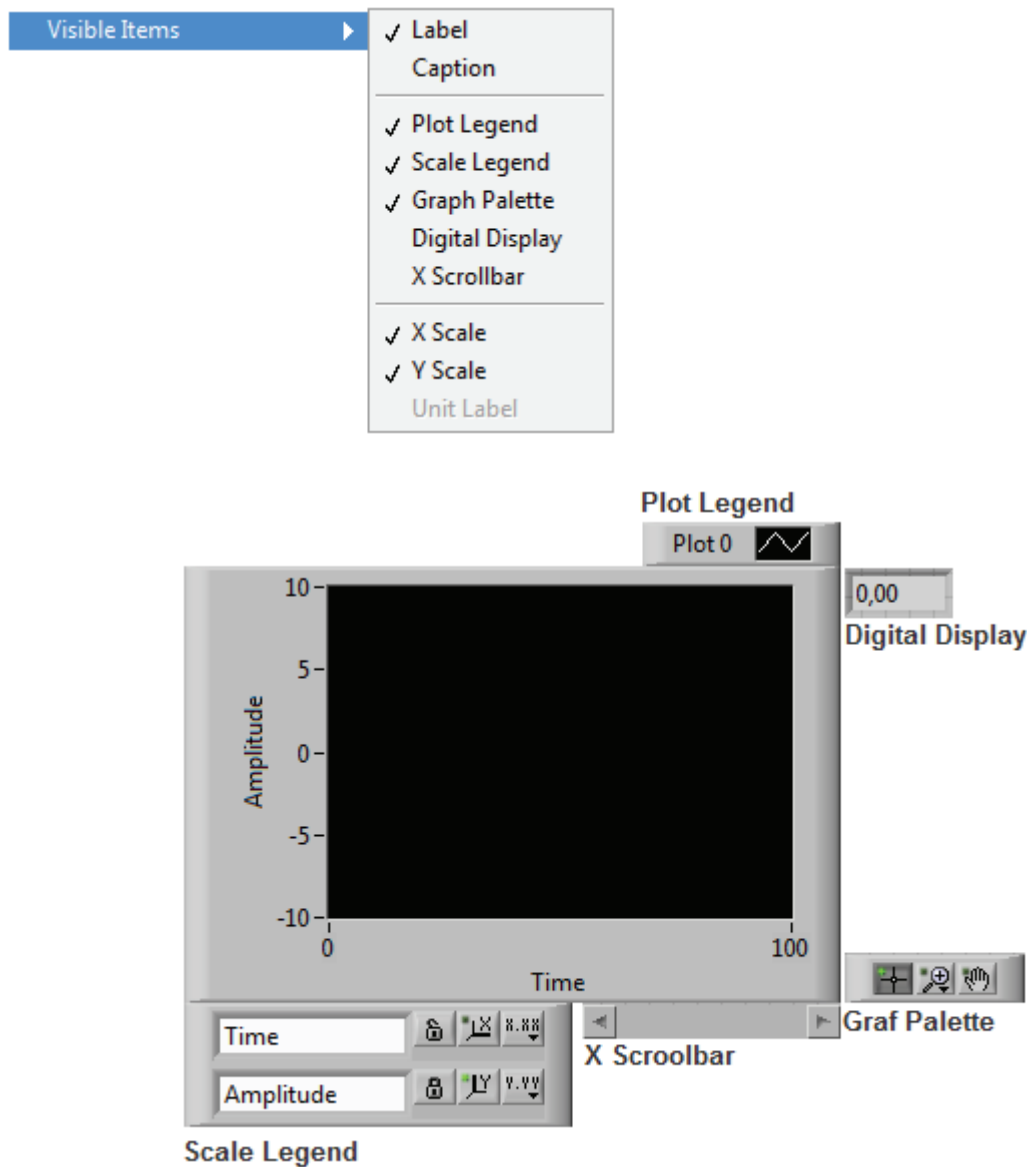
### Inicializacija pomikalnih registrov

Elemente pomikalnega registra lahko inicializiramo. Elementi se inicializirajo tako, da se na vse elemente leve strani pomikalnega registra priključi začetna vrednost (slika spodaj). Ob tem se elementi pomikalnega registra obarvajo v ustrezno barvo. LabVIEW inicializira elemente pomikalnega registra pred začetkom izvajanja zanke.




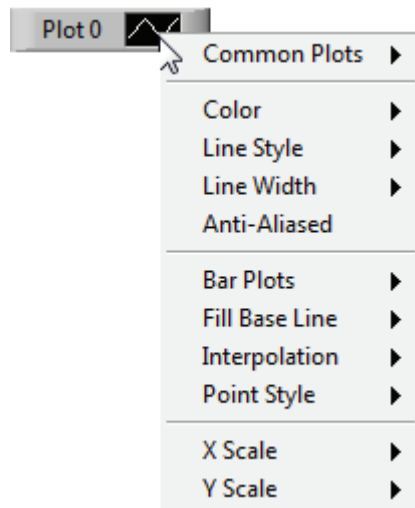
## Uporaba grafikonov

Grafikon se uporablja za grafični prikaz skalarnih podatkov. Vsebuje dodatna orodja, kot so legenda izrisa (Plot Legend), legenda skale (Scale Legend), graf paleta (Graf Palette), digitalni prikaz (Digital Display) in drsnik po x osi ( X Scrollbar), ki omogočajo prilagoditev grafikona. Dodatna orodja se prikažejo ob robu grafikona, vključiti pa jih je možno v priložnem meniju grafikona pod opcijo *Visible Items*.



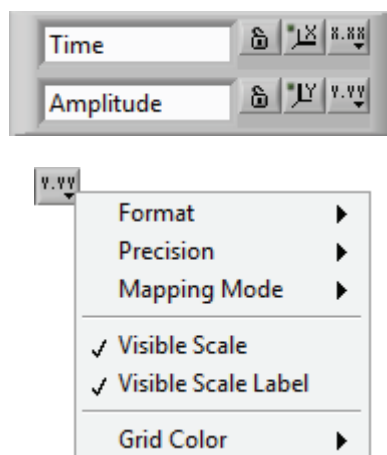
### Plot Legend (Legenda izrisa)

Legenda izrisa se privzeto pojavi v desnem zgornjem kotu grafikona in omogoča spreminjanje barve (Color), stila (Style) in debeline (Line Width) izrisa, določitev izgleda posamezne točke (Point Style), vključitev/izključitev interpolacije med izrisanimi točkami (Interpolation) itd. Spreminjanje teh lastnosti dosežemo z izbiro opcij v priložnem meniju grafikona legende .



### Scale Legend (Legenda skale)

Legenda skale omogoča prilagoditev x in y skale grafikona. Z orodjem spreminjamo imeni x in y osi, izbiramo format izpisa x in y skale (Format), določamo število decimalnih mest izpisa (Precision) ter dodajamo mrežo (Grid Color).



### Graf Palette (Graf paleta)

Graf paleta se privzeto pojavi v desnem spodnjem delu grafikona. Paleta vsebuje orodja za pomikanje in zumiranje prikazanih podatkov.



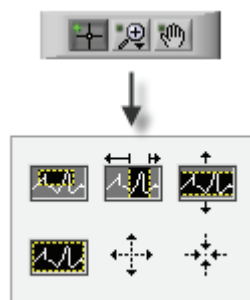
#### **Panning tool** (Orodje za slikovno drsenje)

Orodje omogoča slikovno drsenje vseh podatkov grafikona.



#### **Zoom** (Orodje za zumiranje)

Ob izbiri tega orodja se pojavi dodatna paleta, ki omogoča sledeča zumiranja:





**Zoom to Rectangle** (Okensko zumiranje)



**X-zoom** (Zumiranje po x osi)



**Y-zoom** (Zumiranje po y osi)



**Zoom In about Point** (Povečevanje v okolici točke)



**Zoom Out about Point** (Pomanjševanje v okolici točke)



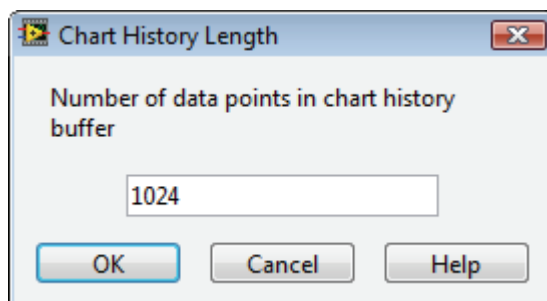
**Zoom to Fit** (Prilagajanje zuma podatkom)

### **Digital Display** (Digitalni prikaz)

Ob izbiri tega orodja se ob grafikonu doda numerični indikator, v katerem se prikaže vrednost zadnjega podatka, ki je bil dodan v grafikon.

### **X Scrollbar** (Drsnik po x osi)

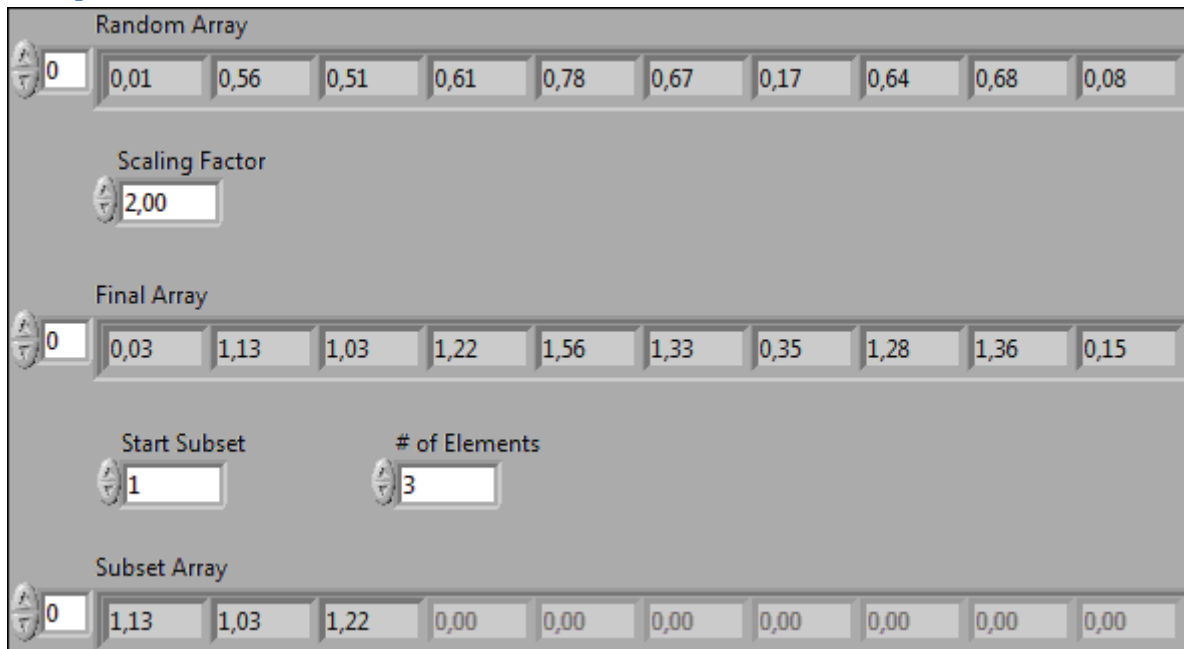
Drsnik po X osi omogoča dostop do zgodovine podatkov grafikona. Privzeto grafikon ohrani zadnjih 1024 podatkov. Privzeto velikost pomnilnika grafikona je možno spremeniti v priročnem meniju grafikona pod opcijo *Chart History Length ...*



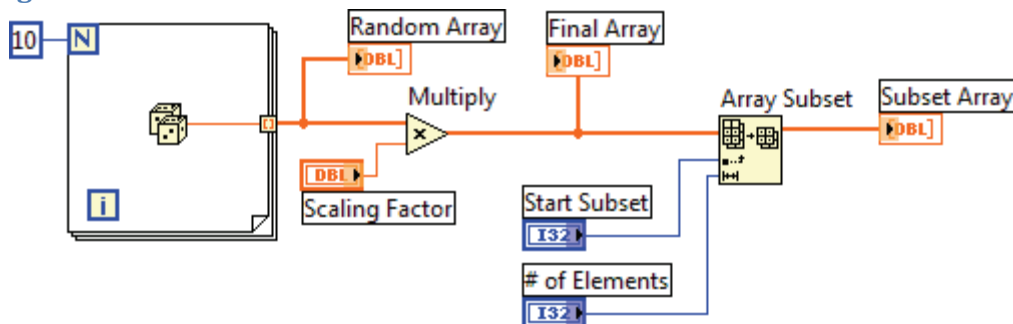
## Vaja 5.1: Array Exercise VI

Izdelajte virtualni instrument, ki bo programsko kreiral enodimenzionalno polje naključnih numeričnih podatkov, ga skaliral s skalirnim faktorjem, kreiral del tega polja, ter vsa kreirana polja prikazal v indikatorjih na čelni plošči.

### Čelna plošča

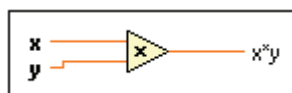


### Blok diagram



#### Multiply (Množilnik)

**Paleta:** Functions > Programming > Numeric

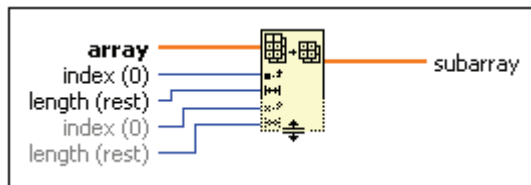


**Opis:** Množilnik izračuna produkt vhodnih podatkov. Podobno kot ostale LabVIEW matematične funkcije je tudi množilnik polimorfična funkcija. To pomeni, da je možno uporabiti isto funkcijo za različne tipe vhodnih podatkov (skalarna vrednost, polje, gruča itd.). V primeru, ko je na en vhod množilnika priključena skalarna vrednost, na drugi vhod pa enodimenzionalno polje, funkcija pomnoži vsak element vhodnega enodimenzionalnega polja s skalarno vrednostjo.



### Array Subset (Del polja)

**Paleta:** Functions > Programming > Array

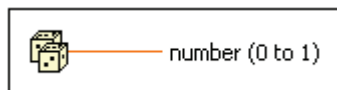


**Opis:** Funkcija *Array Subset* omogoča kreiranje dela vhodnega polja. Na vhod *array* je lahko priključeno eno ali večdimenzionalno polje. Vhodni skalarni podatek *index (0)* določa začetni element izhodnega polja, *length (rest)* pa število elementov.

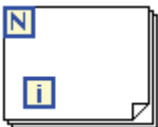


### Random Number (0-1) (Generator naključne vrednosti)

**Paleta:** Functions > Programming > Numeric



**Opis:** Funkcija na izhodu *number (0 to 1)* generira naključno realno število z dvojno natančnostjo. Generirano število je večje ali enako 0 in manjše od 1.



### For Loop (Zanka For)

**Paleta:** Functions > Programming > Structures

**Opis:** Zanka For omogoča ponavljajoče izvajanje kode, ki se nahaja znotraj območja zanke. Zanka se ponovi N-krat, pri čemer je N celoštevilčna vrednost priključena na N priključek zanke. Priključek (i) je števec, ki se poveča po vsaki iteraciji zanke in teče od 0 do N-1.

## Barva in debelina povezav v blok diagramu

Barva in debelina povezav v blok diagramu podata informacijo o tem, kakšen podatek se po posamezni povezavi prenaša.













### Barva

Oranžna barva ponazarja prenos realnega števila, modra prenos celega števila, zelena prenos podatka tipa Boolean in vijolična prenos znakovnega niza (glej naslednjo sliko).

### Debelina povezav

Tanka povezava ponazarja prenos skalarne vrednosti, debelejša (poudarjena) prenos enodimenzionalnega polja (1D polje) in dvojna povezava prenos dvo ali večdimenzionalnega polja (2D polje). Večja kot je dimenzija polja, večja je razdalja med vzporednima črtama povezave.

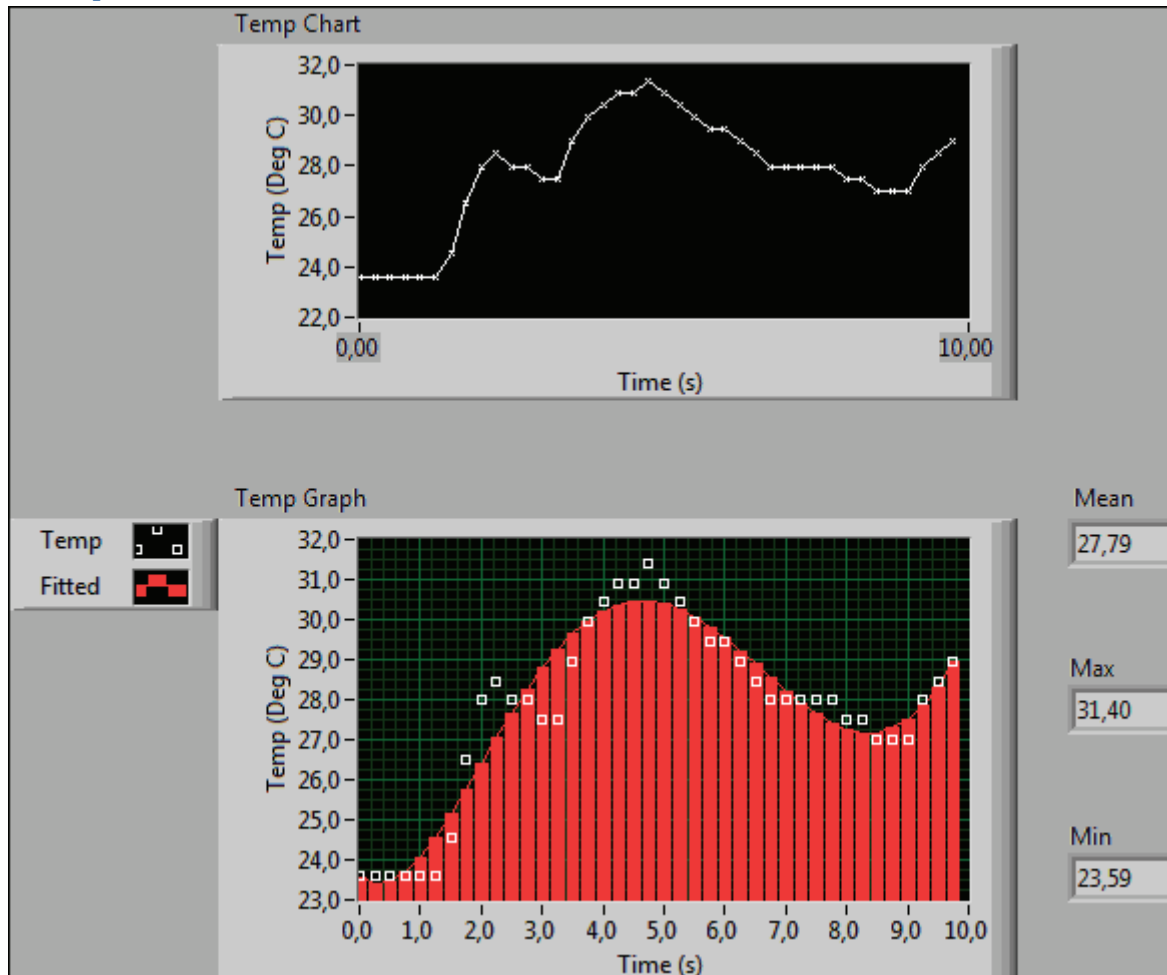


	Skalarna vrednost	1D polje	2D polje
Realno število			
Celo število			
Boolean			
Znakovni niz			

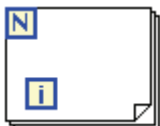
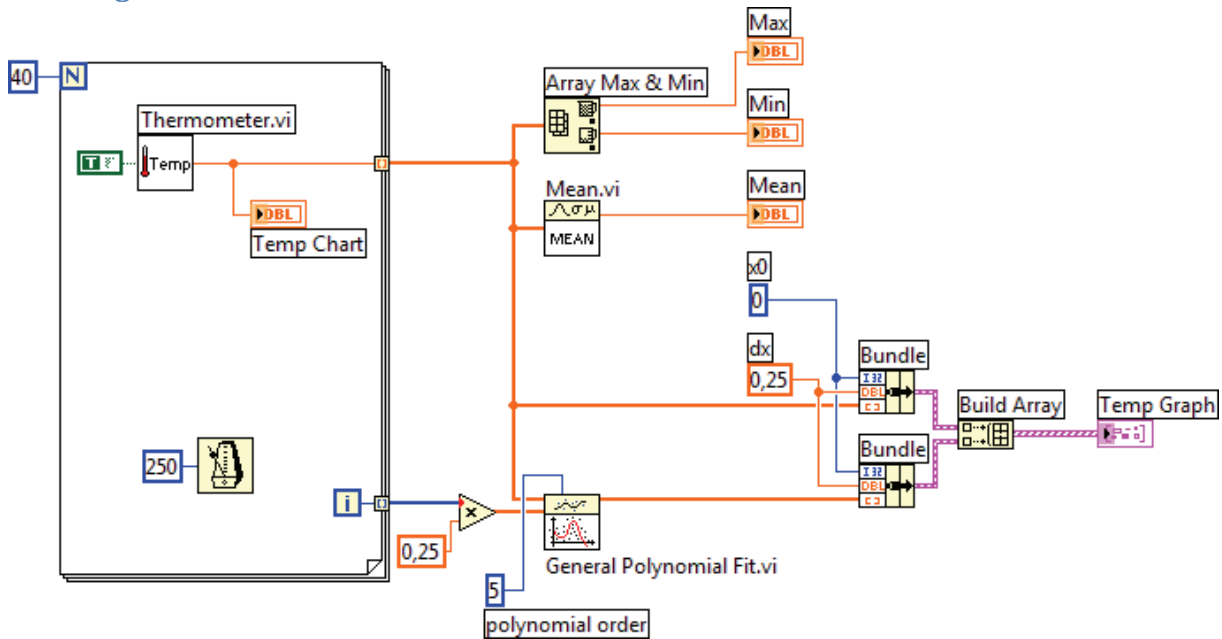
## Vaja 5.2: Temperature Analysis VI

Izdelajte virtualni instrument, ki bo vsakih 0,25 sekunde izmeril temperaturo in jo prikazal v grafikonu čelne plošče. Po 10 sekundah naj se zajemanje temperature zaključi, vrednosti vseh izmerjenih temperatur pa naj se prikažejo v grafu čelne plošče. Dodatno naj se izvede še izračun povprečne, minimalne in maksimalne izmerjene temperature.

### Čelna plošča



### Blok diagram

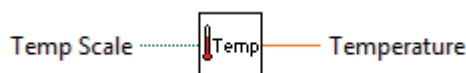


**For Loop** (Zanka For)  
**Paleta:** Functions > Programming > Structures

**Opis:** Zanka For omogoča ponavljajoče izvajanje kode, ki se nahaja znotraj območja zanke. Zanka se ponovi N-krat, pri čemer je N celoštevilčna vrednost priključena na N priključek zanke. Priključek (i) je števec, ki se poveča po vsaki iteraciji zanke in teče od 0 do N-1.



**Thermometer**  
**Paleta:** Functions > Select a VI... > Thermometer.vi (vaja 3.2)

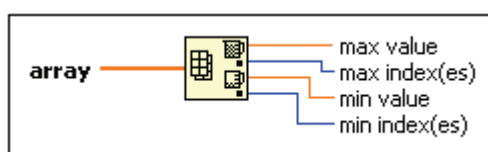


**Opis:** Funkcija na izhodu vrne vrednost izmerjene (simulirane) temperature v °C ali °F. Skala termometra se določi z vhodnim Boolean podatkom *Temp Scale*.

Temp Scale	Temperature
TRUE	°C
FALSE	°F



**Array May & Min**  
**Paleta:** Functions > Programming > Array

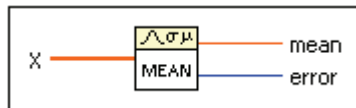


**Opis:** Funkcija *Array Max & Min* izračuna maksimalno (*max value*) in minimalno (*min value*) vrednost vhodnega polja. Dodatno k vrednostim funkcija vrne tudi indeks elementa maksimalne (*max index*) in indeks elementa minimalne (*min index*) vrednosti.



**Mean** (Povprečna vrednost)

**Paleta:** Functions > Mathematics > Probability & Statistics



**Opis:** Funkcija *Mean* izračuna povprečno vrednost elementov vhodnega polja po sledeči enačbi:

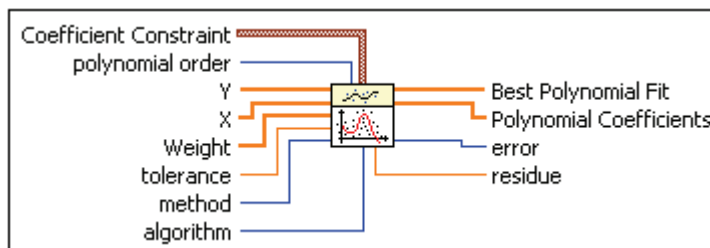
$$\mu = \frac{1}{N} \sum_{i=0}^{N-1} x_i,$$

pri čemer je  $\mu$  povprečna vrednost in  $N$  število elementov vhodnega polja  $x$ .



**General Polynomial Fit**

**Paleta:** Functions > Mathematics > Fitting

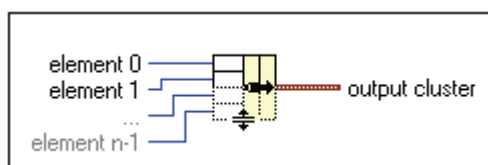


**Opis:** Funkcija *General Polynomial Fit* izračuna polinom določenega reda za točke (X,Y), ki so podane s poljema X in Y. Red polinoma se določi s parametrom *polynomial order*.



**Bundle** (Grupiranje)

**Paleta:** Functions > Programming > Cluster, Class & Variant

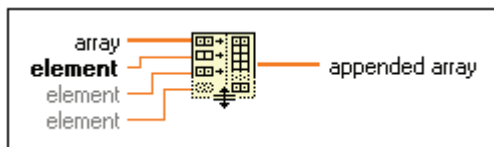


**Opis:** Funkcija *Bundle* programsko kreira gručo vhodnih podatkov. Vhodna polja gruče se avtomatsko obarvajo glede na tip vhodnega podatka.



**Build Array**

**Paleta:** Functions > Programming > Array

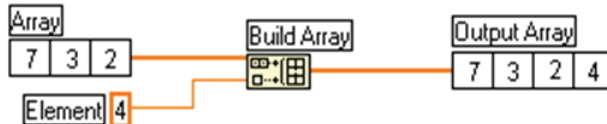


**Opis:** Funkcija *Build Array* omogoča programsko kreiranje polja, spojitve več vhodnih polj ali dodajanje elementov n-dimenzionalnemu polju. Ob namestitvi funkcije v blok diagram ima funkcija samo en vhod in en izhod. Funkciji dodamo vhod tako, da v priročnem meniju funkcije izberemo opcijo *Add Input*.

## Uporaba funkcije

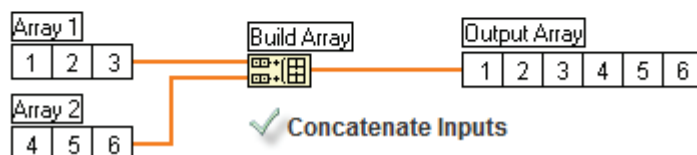
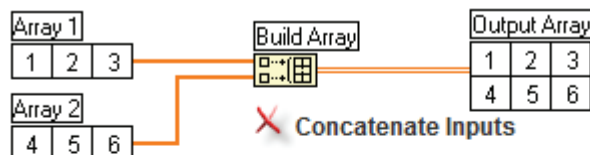
### 1. Enodimenzionalno vhodno polje in skalarna vrednost

V primeru, ko je na en vhod funkcije *Build Array* priključeno enodimenzionalno polje, na drugi vhod pa skalarna vrednost, funkcija *Build Array* doda skalarno vrednost enodimenzionalnemu vhodnemu polju (poveča se število elementov polja).



### 2. Enodimenzionalna vhodna polja

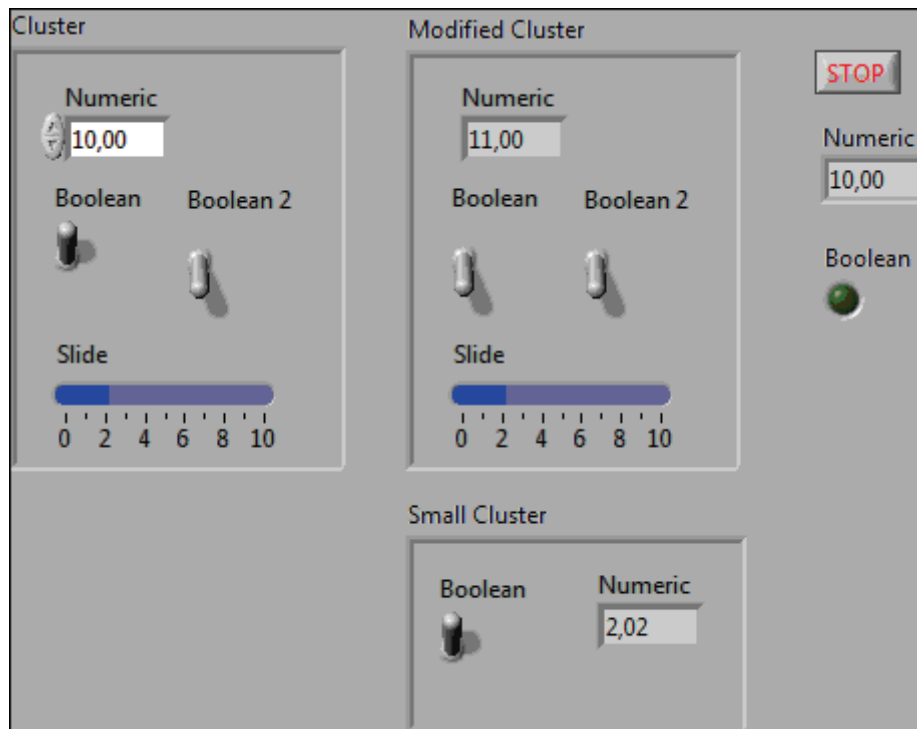
V primeru, ko so na vhode funkcije *Build Array* priključena enodimenzionalna polja, lahko funkcija *Build Array* kreira enodimenzionalno ali dvodimenzionalno izhodno polje. Dimenzija izhodnega polja se določi z dodanim parametrom *Concatenate Inputs*, ki se nahaja v priročnem meniju funkcije. Če je ta opcija vključena, funkcija kreira enodimenzionalno izhodno polje, v nasprotnem primeru pa dvodimenzionalno polje.



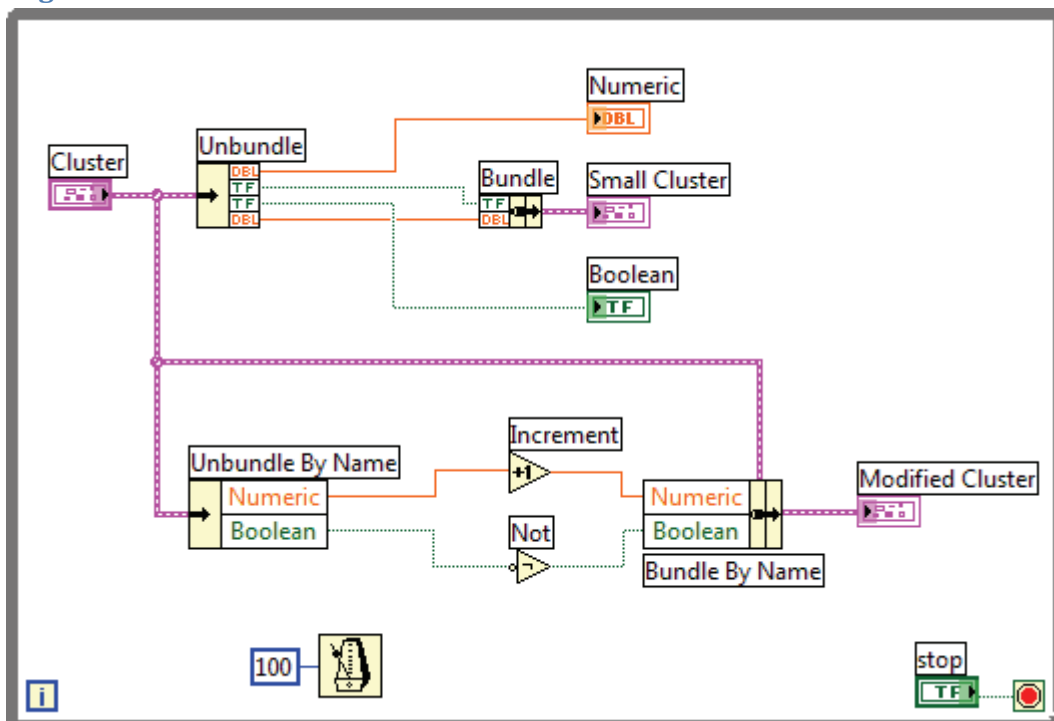
## Vaja 5.3: Cluster Exercise VI

Izdelajte virtualni instrument, ki ponazarja delo z gručami in je prikazan na spodnjih slikah.

### Čelna plošča



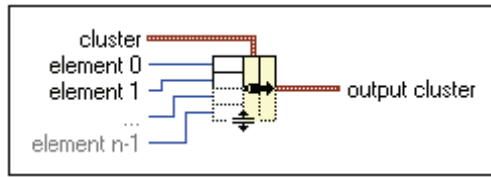
### Blok diagram





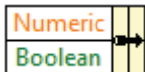
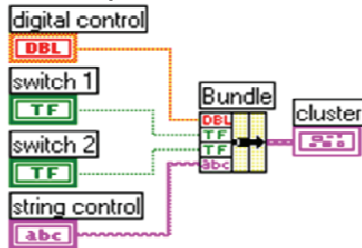
### Bundle (Grupiranje)

**Paleta:** Functions > Programming > Cluster, Class & Variant



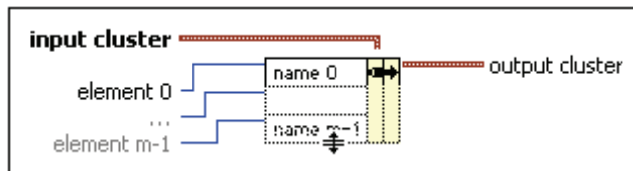
**Opis:** Funkcija *Bundle* omogoča programsko kreiranje gruč ali spreminjanje podatkov elementom obstoječe gruče. Če na sredinski priključek funkcije (imenovan *cluster*) ni priključena obstoječa gruča, funkcija kreira novo gručo vhodnih podatkov. Ob pravilni povezavi vhodnih podatkov se na vhodnih priključkih te funkcije pojavijo oznake podatkovnih tipov vhodnih podatkov.

#### Primer uporabe



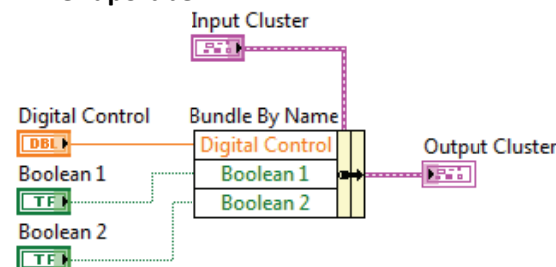
### Bundle by Name

**Paleta:** Functions > Programming > Cluster, Class & Variant



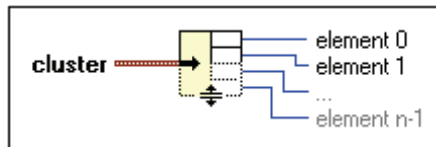
**Opis:** Funkcija *Bundle by Name* omogoča spreminjanje podatkov elementom obstoječe gruče. Na sredinski vhod funkcije (imenovan *input cluster*) mora biti priključena obstoječa gruča. Ob priključitvi te gruče se na levi strani funkcije pojavijo imena elementov gruče, na te vhode pa se nato priključi nova vrednost elementa.

#### Primer uporabe



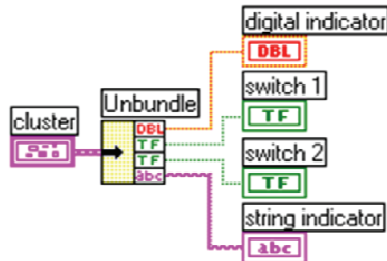
### Unbundle

**Paleta:** Functions > Programming > Cluster, Class & Variant



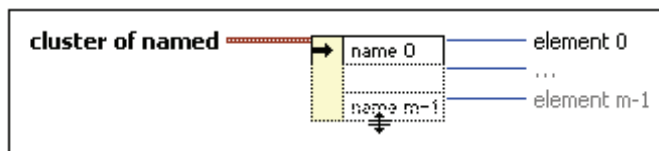
**Opis:** Funkcija *Unbundle* omogoča razčlenitev gruče na posamezne elemente. Ob priključitvi vhodne gruče se na izhodu funkcije pojavi toliko izhodnih priključkov, kot je elementov gruče. Na izhodnih priključkih se izpišejo oznake podatkovnih tipov elementov gruče.

**Primer uporabe**



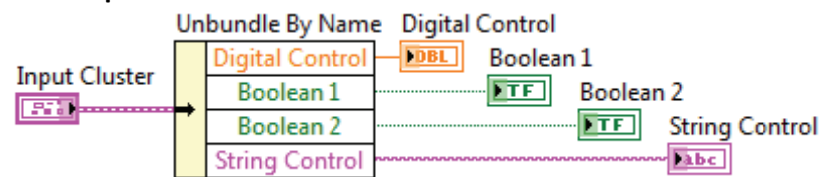
**Unbundle by Name**

**Paleta:** Functions > Programming > Cluster, Class & Variant



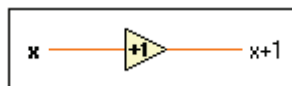
**Opis:** Funkcija *Unbundle by Name* omogoča razčlenitev gruče na posamezne elemente. Ob priključitvi vhodne gruče se na izhodu funkcije pojavi toliko izhodnih priključkov, kot je elementov gruče. Za razliko od funkcije *Unbundle* se na izhodnih priključkih izpišejo imena elementov gruče.

**Primer uporabe**



**Increment (Inkrement)**

**Paleta:** Functions > Programming > Numeric

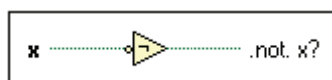


**Opis:** Funkcija poveča vhodni podatek x za 1.



**Not (Negator)**

**Paleta:** Functions > Programming > Boolean



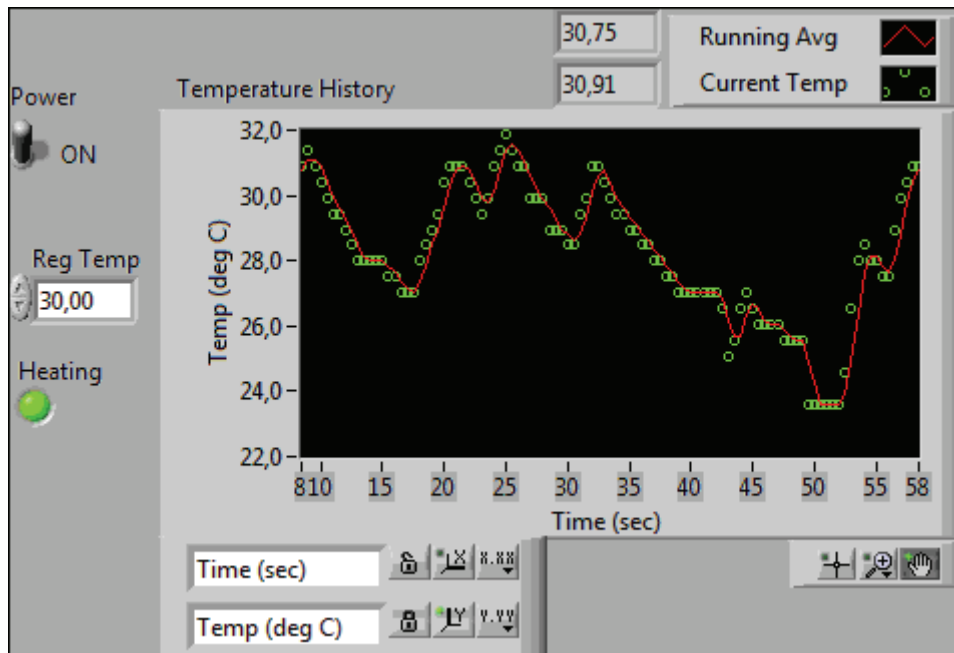
**Opis:** Funkcija negira vhodni Boolean podatek.



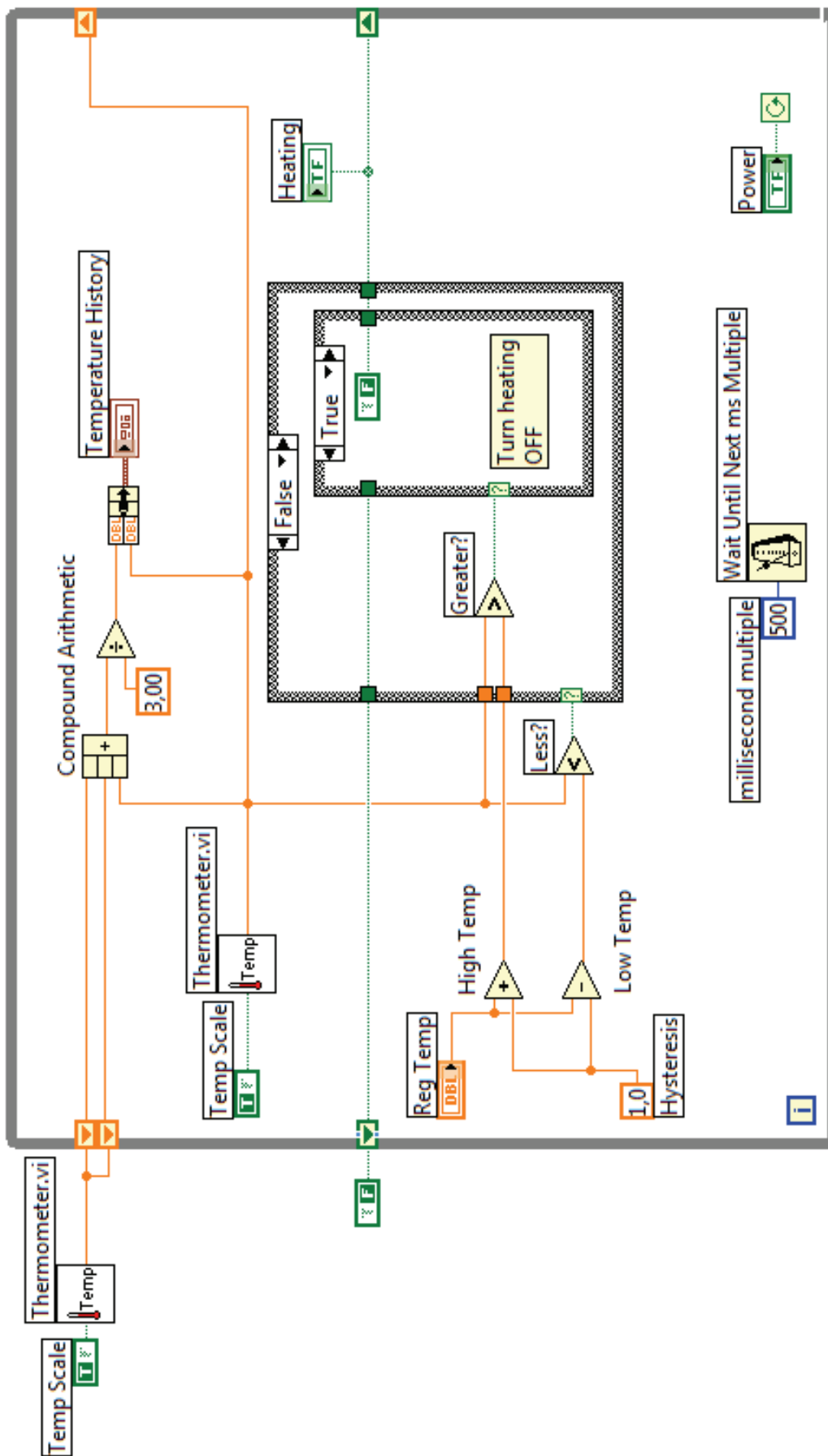
## Vaja 6.1: Temperature Control VI

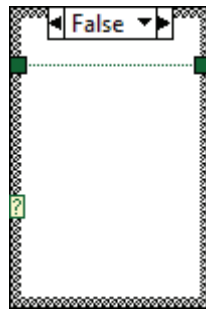
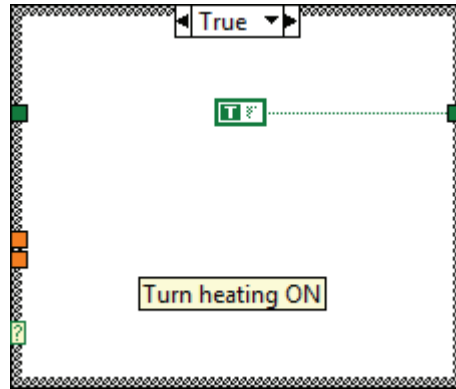
Nadgradite virtualni instrument *Temperature Running Average* tako, da bo vklopil in izklopil LED-diodo (*Heating*) na čelni plošči v odvisnosti od trenutne in zelene temperature. Ko temperatura pade za več kot stopinjo pod želeno, naj se LED-dioda vključi in ostane vključena, vse dokler temperatura ne naraste za stopinjo čez želeno. Želeno vrednost temperature naj uporabnik vnese v kontrolo čelne plošče (*Reg Temp*).

### Čelna plošča



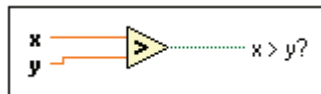
Blok diagram





### Greater? (Večje?)

**Paleta:** Functions > Programming > Comparison

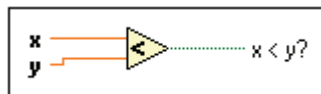


**Opis:** Funkcija vrne logično vrednost TRUE, če je vhodni podatek x večji od y.



### Less? (Manjše?)

**Paleta:** Functions > Programming > Comparison

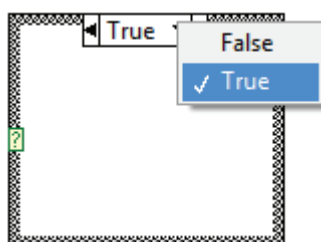


**Opis:** Funkcija vrne logično vrednost TRUE, če je vhodni podatek x manjši od y.





### Case Structure (Programska struktura Case)

**Paleta:** Functions > Programming > Structures



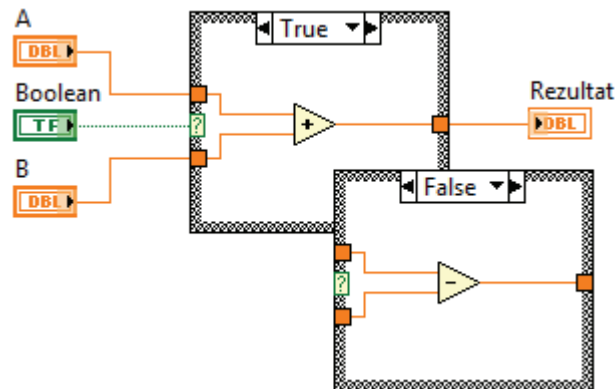
**Opis:** Programska struktura Case omogoča razdelitev algoritma v najmanj dve veji. Privzeto se ob namestitvi strukture kreirata dve veji oz. dva okvirja (True in False), ki

sta nameščena drug nad drugim. Veja, po kateri se nadaljuje izvajanje algoritma, je odvisna od podatka, pripeljanega na izbirni priključek  (*selector terminal*), in podatka, vnesenega v selektor strukture Case . LabVIEW izvede tisto vejo algoritma, ki ustreza vhodnemu podatku na izbirnem priključku. Če nobena veja ne ustreza vhodnemu podatku, pa se izvede privzeta veja (*Default*).

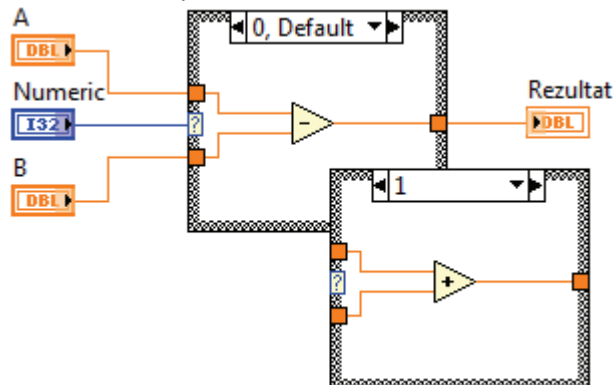
Veje (okvire) lahko dodajamo ali odstranimo s pomočjo opcij v priročnem meniju strukture.

### Primeri uporabe programske strukture Case

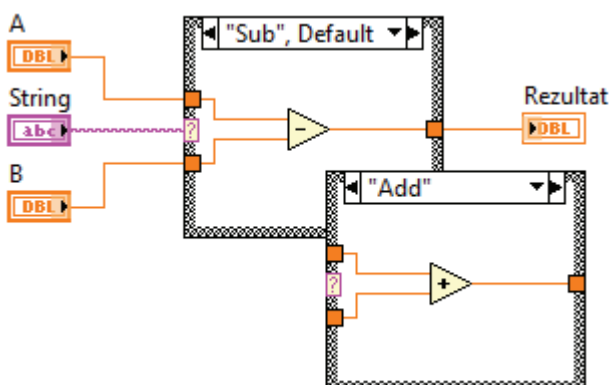
#### 1. Podatek tipa Boolean



#### 2. Numerični podatek



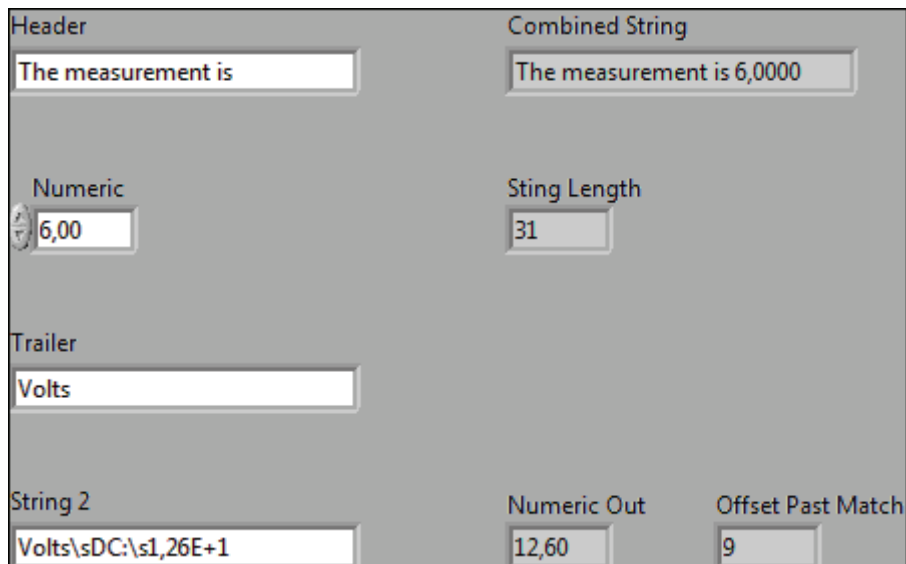
#### 3. Znakovni niz



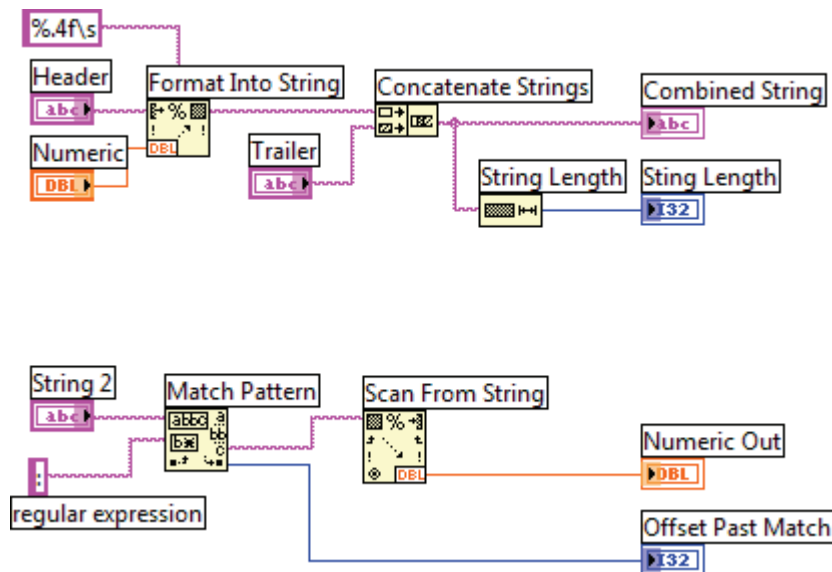
## Vaja 7.1: Build String VI

Izdelajte virtualni instrument, ki bo pretvoril število v znakovni niz, ga spojil z drugimi znakovnimi nizi in določil število znakov v rezultirajočem znakovnem nizu.

### Čelna plošča

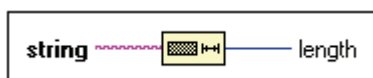


### Blok diagram



**String Length** (Dolžina znakovnega niza)

**Paleta:** Functions > Programming > String

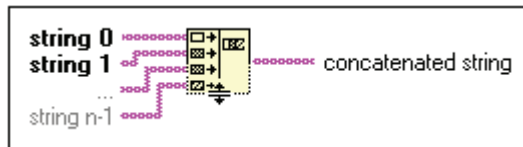


**Opis:** Funkcija *String Length* vrne število znakov, ki jih vhodni znakovni niz vsebuje.



**Concatenate String**

**Paleta:** Functions > Programming > String

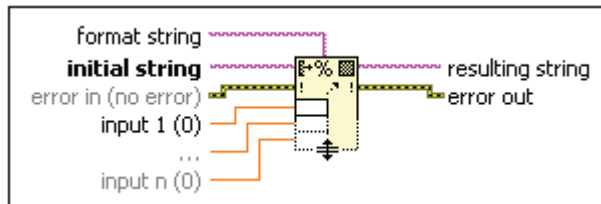


**Opis:** Funkcija *Concatenate String* omogoča spojitve vhodnih znakovnih nizov v skupen znakovni niz. Vhodni znakovni nizi so lahko tudi v obliki 1D-polja.

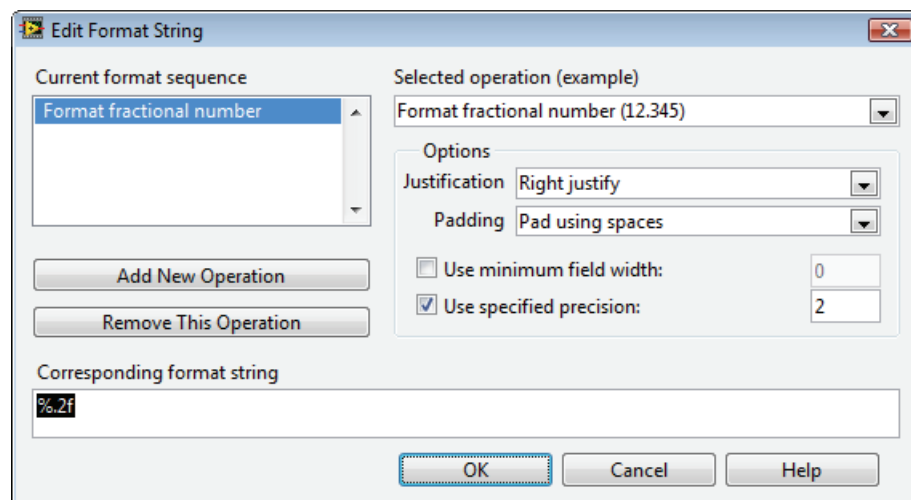


### Format Into String

**Paleta:** Functions > Programming > String

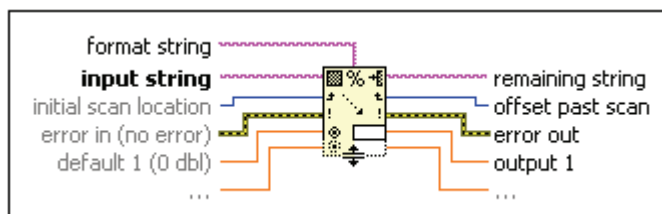


**Opis:** Funkcija *Format Into String* omogoča pretvorbo vhodnih števil v znakovni niz. Način pretvorbe se določi s posebnim formatnim zapisom, priključenim na vhod *format string*. Če formatnega zapisa ne poznamo, si lahko pomagamo z dialognim oknom funkcije (prikaže se ob dvojnem kliku na funkcijo). Ob potrditvi nastavljenih opcij, LabVIEW avtomatsko kreira znakovno konstanto z ustreznim formatnim zapisom in jo poveže na vhod *format string*.



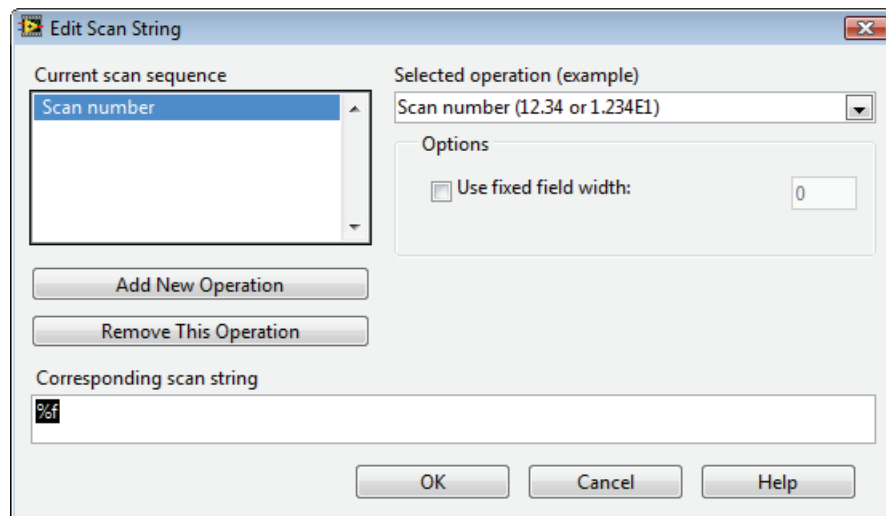
### Scan From String

**Paleta:** Functions > Programming > String



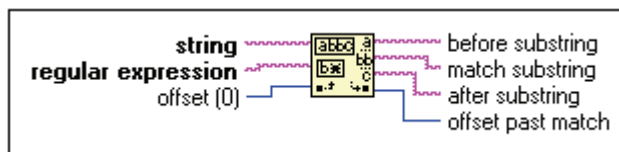
**Opis:** Funkcija *Scan From String* omogoča pretvorbo vhodnega znakovnega niza v število. Način pretvorbe se določi s posebnim formatnim zapisom, priključenim na vhod *format string*. Če formatnega zapisa ne poznamo, si lahko pomagamo z

dialognim oknom funkcije (prikaže se ob dvojnem kliku na funkcijo). Ob potrditvi nastavljenih opcij LabVIEW avtomatsko kreira znakovno konstanto z ustreznim formatnim zapisom in jo poveže na vhod funkcije *format string*.



### Match Pattern

**Paleta:** Functions > Programming > String



**Opis:** Funkcija *Match Pattern* se uporablja za iskanje znakovnih nizov (*regular expression*) znotraj vhodnega znakovnega niza (*string*). Funkcija razdeli vhodni niz (*string*) v tri podnize, in sicer: (1) niz, ki se nahaja pred iskanim nizom (*before substring*), (2) iskani niz (*match substring*) in (3) niz, ki se nahaja za iskanim nizom (*after substring*). Funkcija na izhodu poda tudi položaj (*offset past match*), kjer je bil iskani znakovni niz najden. Če iskanega niza v vhodnem nizu ni bilo moč najti, je izhod *offset past match* enak -1.

## Vaja 7.2: Temperature Logger VI

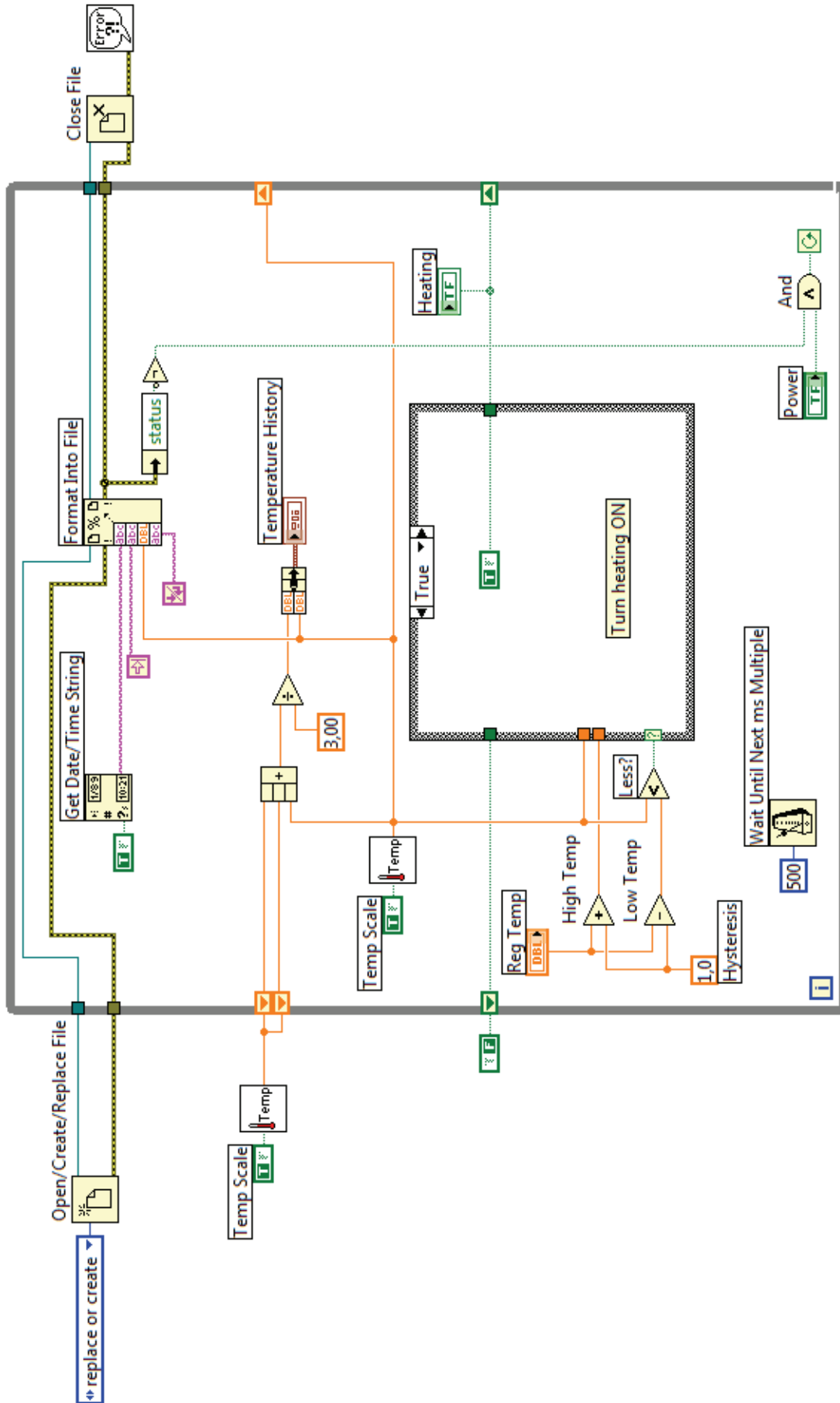
Nadgradite virtualni instrument *Temperature Control* z algoritmom, ki bo shranjeval trenutni čas in trenutno temperaturo v datoteko.

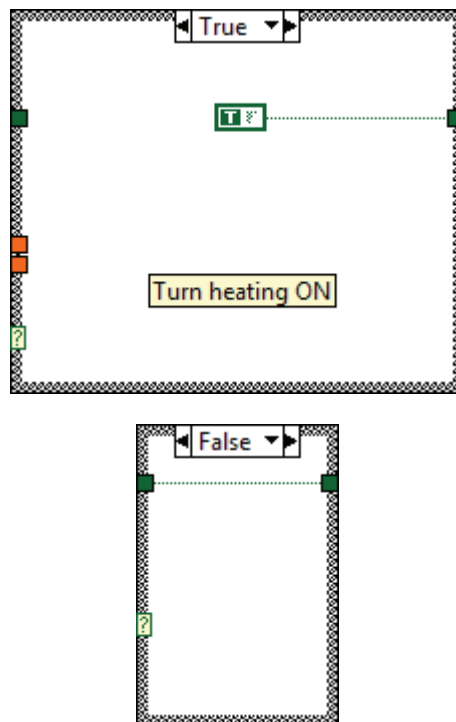
### Čelna plošča





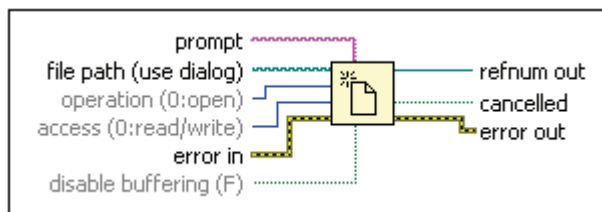
Blok diagram





### Open/Create/Replace File

**Paleta:** Functions > Programming > File I/O

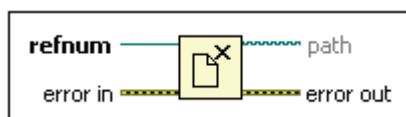


**Opis:** Funkcija *Open/Create/Replace File* odpre obstoječo ali kreira novo datoteko. Če na vhod *file path* ni priključena pot do obstoječe datoteke, se ob zagonu virtualnega instrumenta pojavi dialogno okno, kjer je potrebno izbrati datoteko, ki bi jo radi odprli. Na izhodu funkcije se pojavi referenčna številka datoteke (*refnum out*), na katero se je potrebno sklicevati pri delu s to datoteko.



### Close File

**Paleta:** Functions > Programming > File I/O

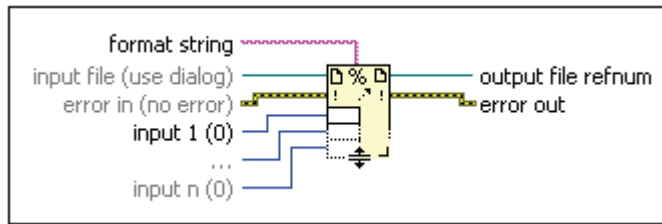


**Opis:** Funkcija *Close File* zapre datoteko z referenčno številko *refnum*.



### Format Into File

**Paleta:** Functions > Programming > File I/O

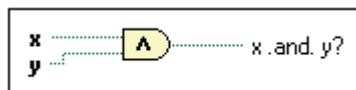


**Opis:** Funkcija pretvori vhodne podatke v znakovne nize in jih shrani v datoteko.



### And (Logični AND)

**Paleta:** Functions > Programming > Boolean

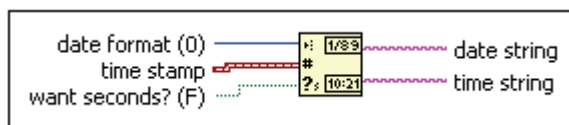


**Opis:** Funkcija izvede logični AND nad vhodnima podatkom x in y.



### Get Date/Time String

**Paleta:** Functions > Programming > Timing

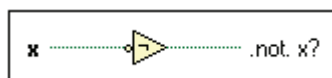


**Opis:** Funkcija poda na izhodu trenutni čas (*time*) in datum (*date*) v obliki znakovnih nizov.



### Not (Negator)

**Paleta:** Functions > Programming > Boolean



**Opis:** Funkcija negira vhodni Boolean podatek.



### Tab Constant

**Paleta:** Functions > Programming > String

**Opis:** *Tab Constant* je znakovna konstanta, ki vsebuje ASCII znak za tabulator.



### End of Line Constant

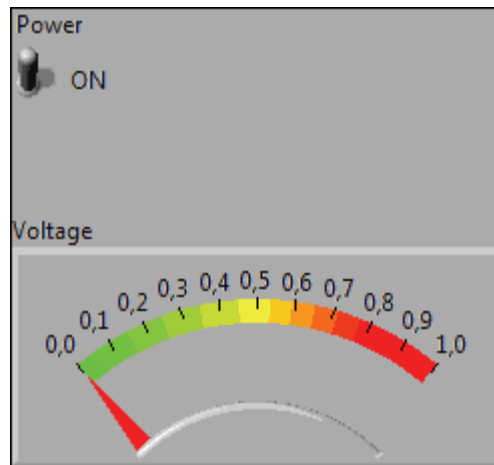
**Paleta:** Functions > Programming > String

**Opis:** *End of Line Constant* je znakovna konstanta, ki vsebuje ASCII znak za določitev konca vrstice.

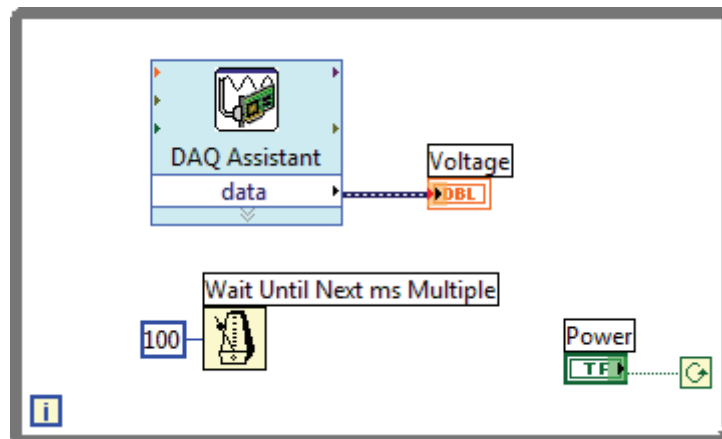
## Vaja 8.1: Voltmeter VI

Izdelajte virtualni instrument, ki bo kontinuirano zajemal analogno napetost, priključeno na analogni vhod 0 merilne kartice, in jo prikazoval v indikatorju *Voltage* na čelni plošči. Zajemanje naj se izvaja vsakih 100 ms, dokler je stikalo *Power* na čelni plošči vključeno.

### Čelna plošča

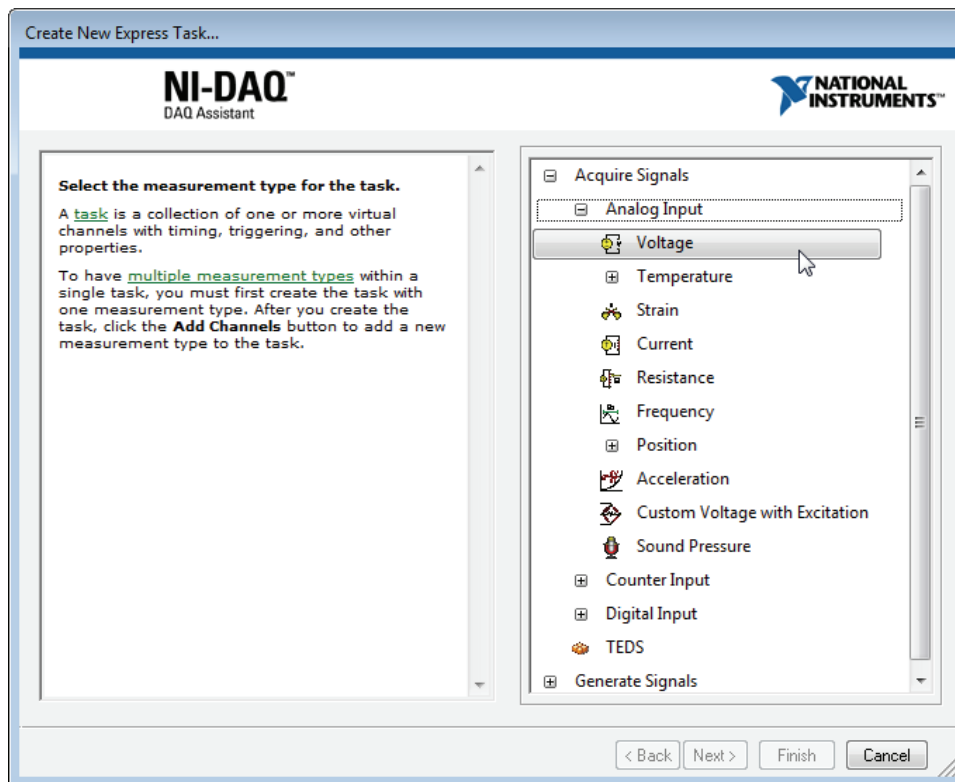


### Blok diagram

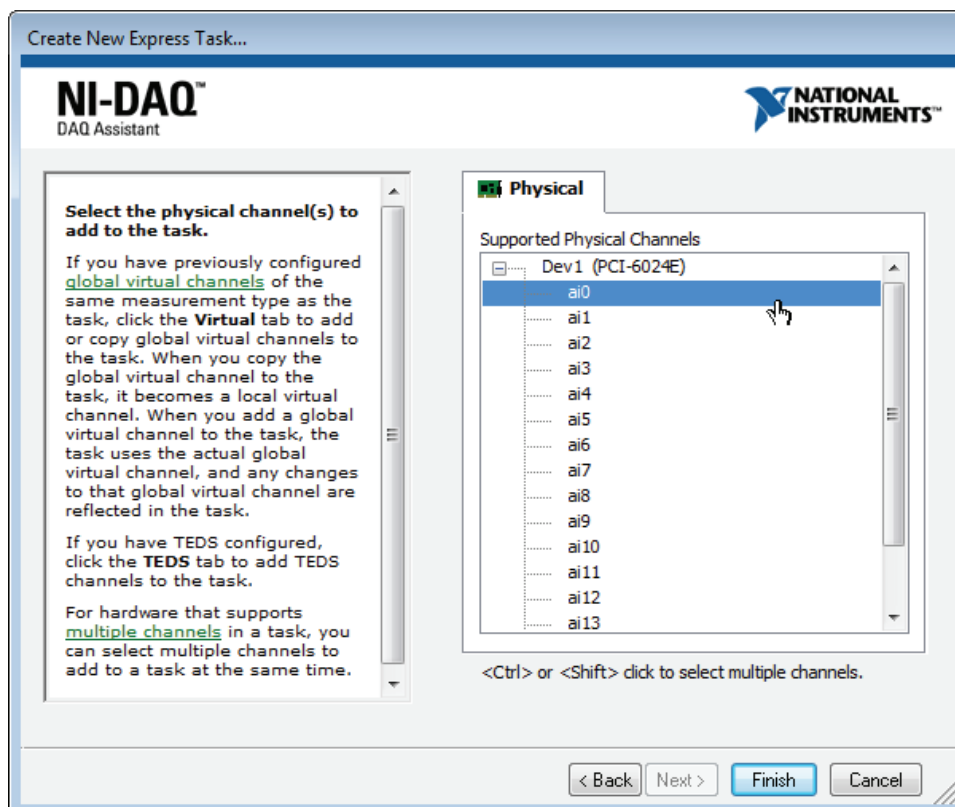


### Namestitev in nastavitev funkcije DAQ Assistant

1. V While zanko namestite funkcijo *DAQ Assistant*, ki se nahaja na paleti *Measurement I/O > NI-DAQmx*.
2. Po dvojnem kliku na funkcijo se odpre okno *Create New Express Task*. V desnem okviru okna izberite *Acquire Signals > Analog Input > Voltage* ter nato izberite gumb *Next*.

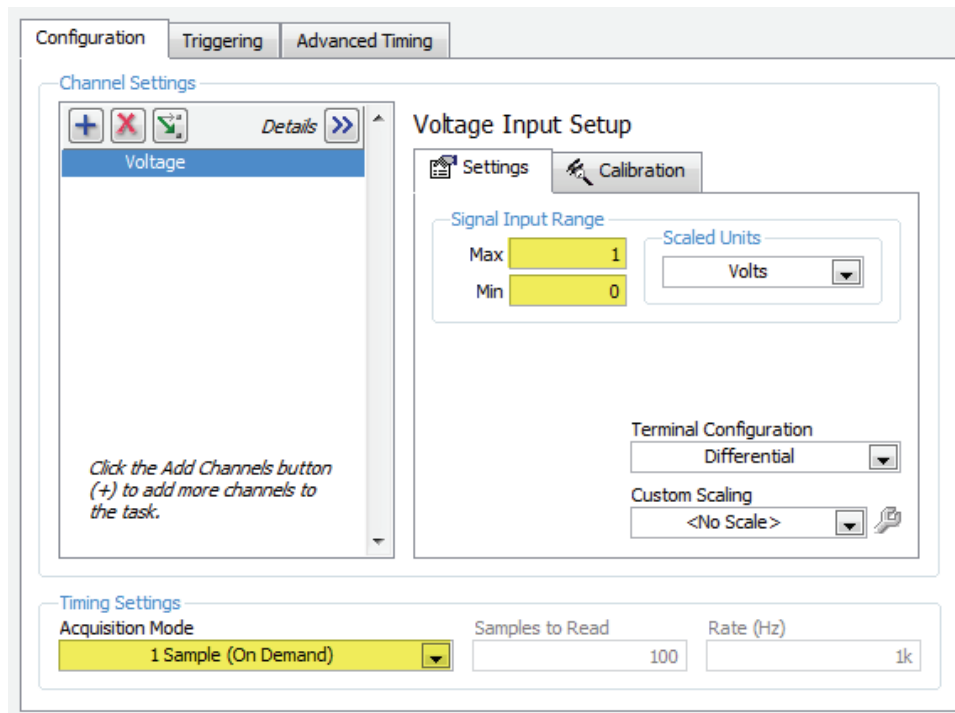


3. V okvirju *Supported Physical Channels* izberite *Dev1 (PCI-6024E)* > *ai0* in izberite gumb *Finish*.

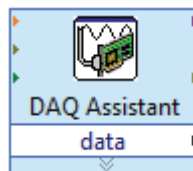


4. V zavihku *Configuration* > *Settings* nastavite:
- Voltage Input Setup
    - Signal Input Range

1. Max: 1
  2. Min: 0
- b) Timing Settings
- i. Acquisition Mode: 1 Sample (One Demand)



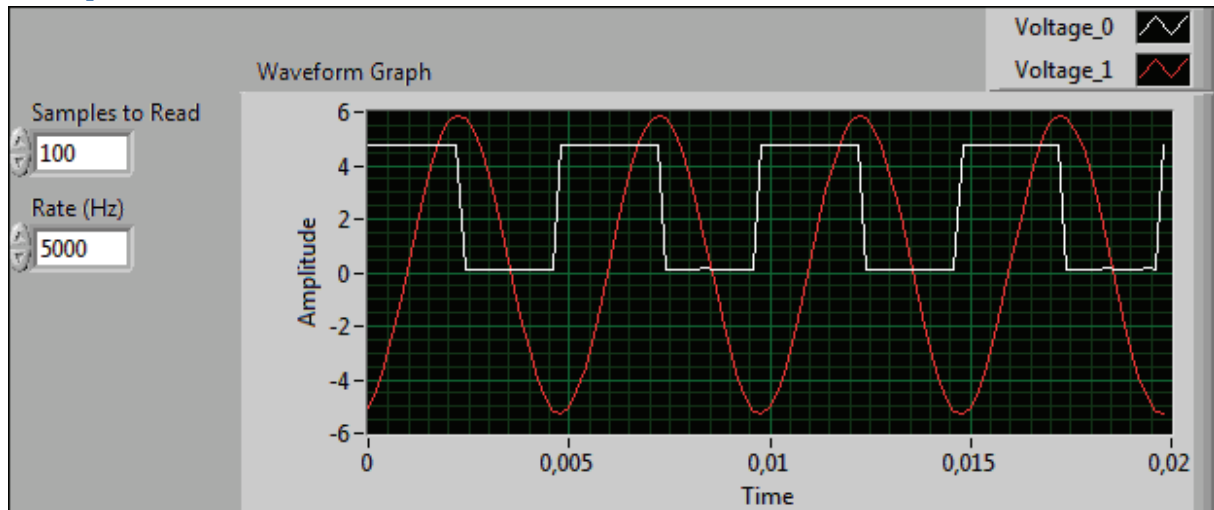
5. Izberite gumb OK. Funkcija DAQ Assistant se bo v blok diagramu avtomatsko posodobila tako, kot je prikazano na spodnji sliki. Povežite izhod *data* s priključkom indikatorja *Voltage*.



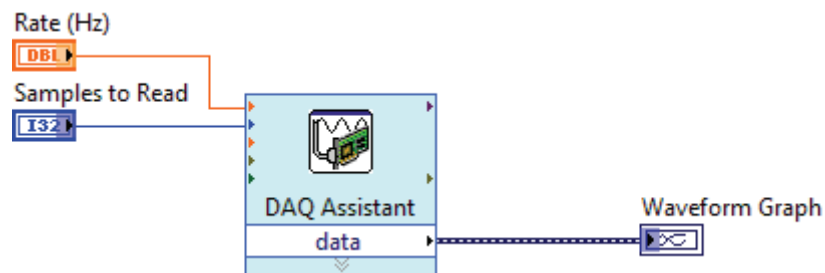
## Vaja 8.2: Acquire Waveforms VI

Izdelajte virtualni instrument, ki bo zajel dva analogna signala, priključena na analogni vhod 1 in 2 merilne kartice. Število vzorcev in frekvenca zajemanja naj bosta nastavljivi preko kontrol čelne plošče.

### Čelna plošča

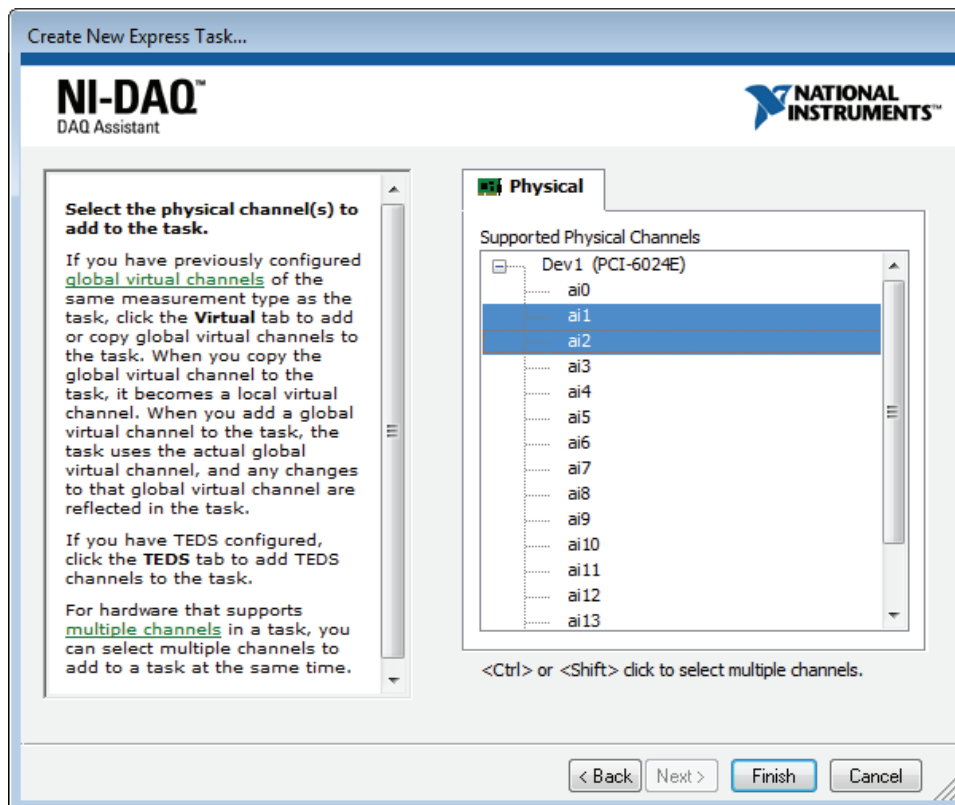


### Blok diagram



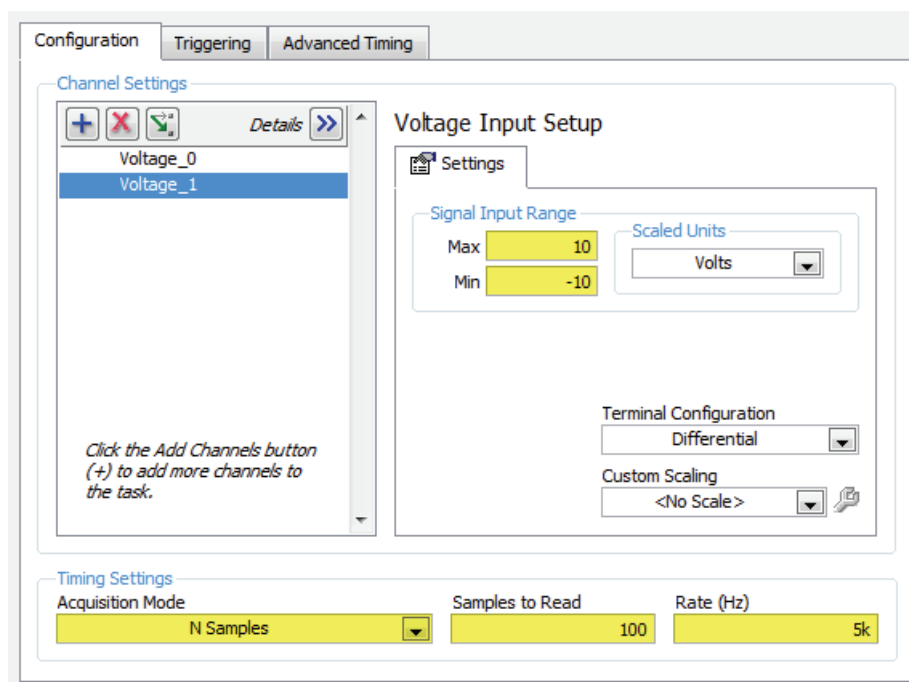
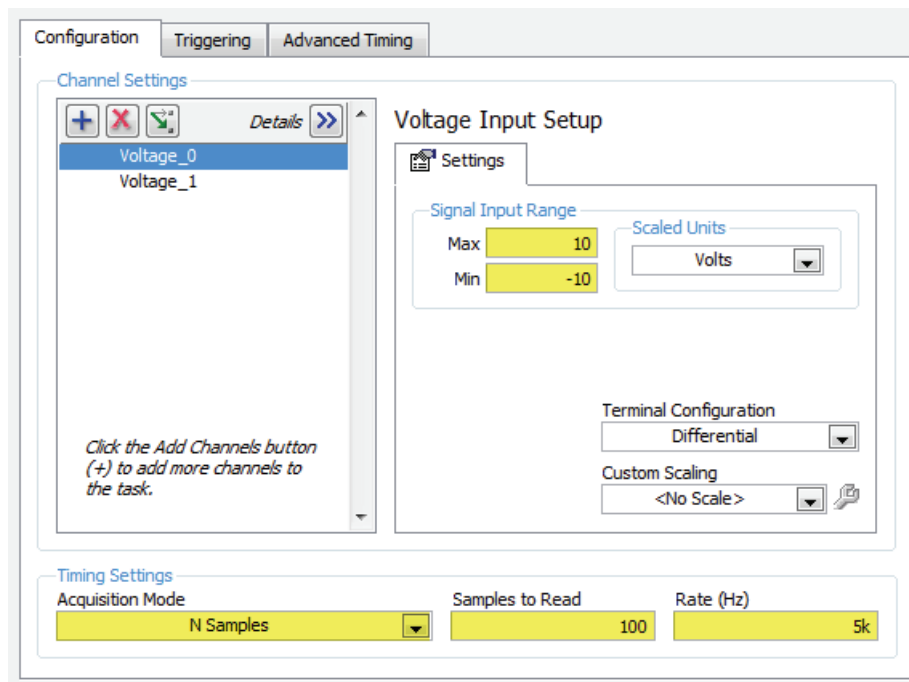
### Namestitev in nastavitev funkcije DAQ Assistant

1. V blok diagram namestite funkcijo *DAQ Assistant* (paleta: *Measurement I/O > NI-DAQmx*).
2. V nastavitvenem oknu funkcije *DAQ Assistant* izberite *Acquire Signals > Analog Input > Voltage* ter nato izberite gumb *Next*.
3. V okviru *Supported Physical Channels* izberite *Dev1 (PCI-6024E) > ai1* in *ai2* ter izberite gumb *Finish*.

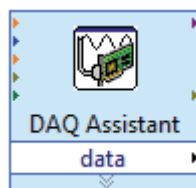


4. Pomaknite se na zavihek *Configuration*. V okviru *Channel Settings* nastavite parametre obeh kanalov (*voltage\_0* in *voltage\_1*) na sledeče vrednosti:
  - a) Voltage Input Setup
    - i. Signal Input Range
      1. Max: 10
      2. Min: -10
5. Časovne parametre (*Timing Settings*) nastavite sledeče:
  - a) Acquisition Mode: N Samples
  - b) Samples to Read: 100
  - c) Rate (Hz): 5k





6. Izberite gumb OK. Funkcija DAQ Assistant se bo v blok diagramu avtomatsko posodobila tako, kot je prikazano na spodnji sliki. Povežite izhod *data* s priključkom grafa čelne plošče.

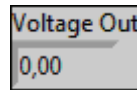


7. Na funkcijskem generatorju nastavite frekvenco na 200 Hz in amplitudo na polovico območja ter poženite virtualni instrument.

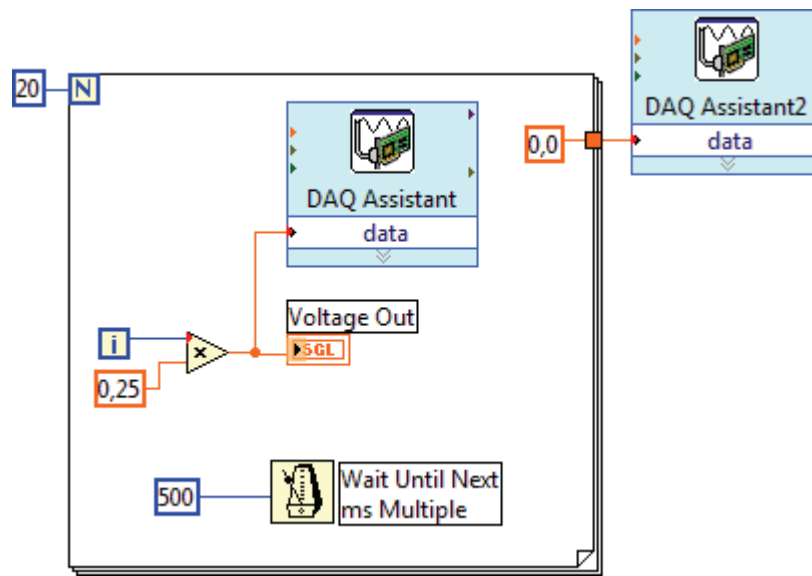
## Vaja 8.3: Voltage Output Example

Izdelajte virtualni instrument, ki bo generiral analogno napetost (v območju 0 do 5 V) na analognem izhodu 0 merilne kartice. Generirana napetost naj se poveča vsake 0,5 sekunde za 0,25 V.

### Čelna plošča

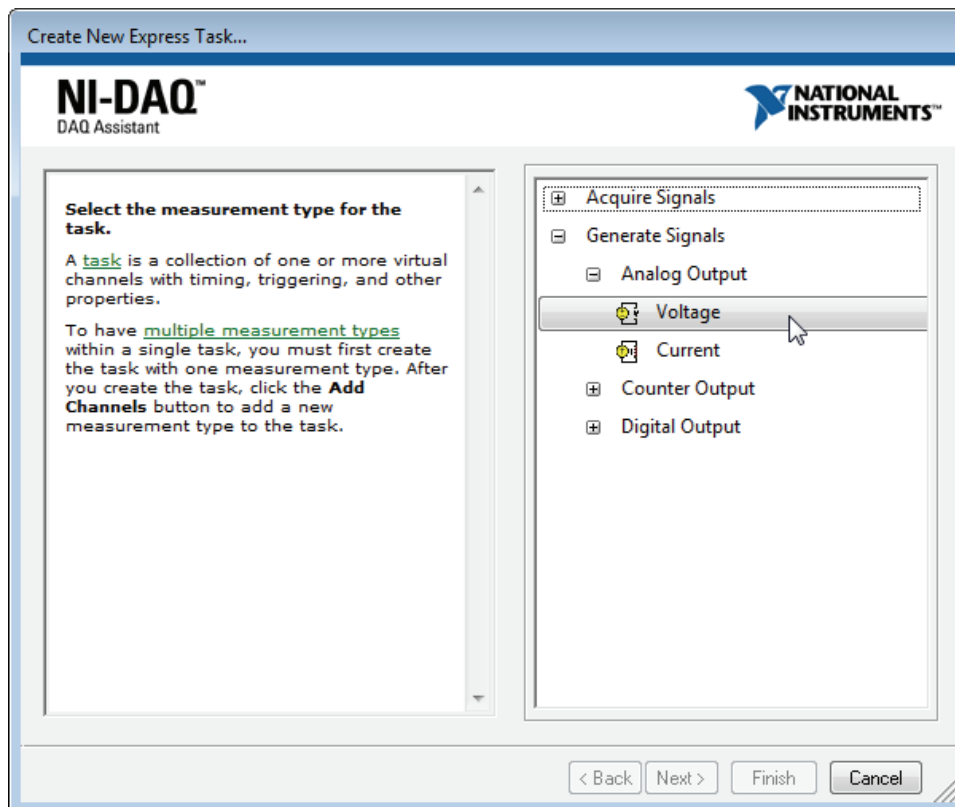


### Blok diagram

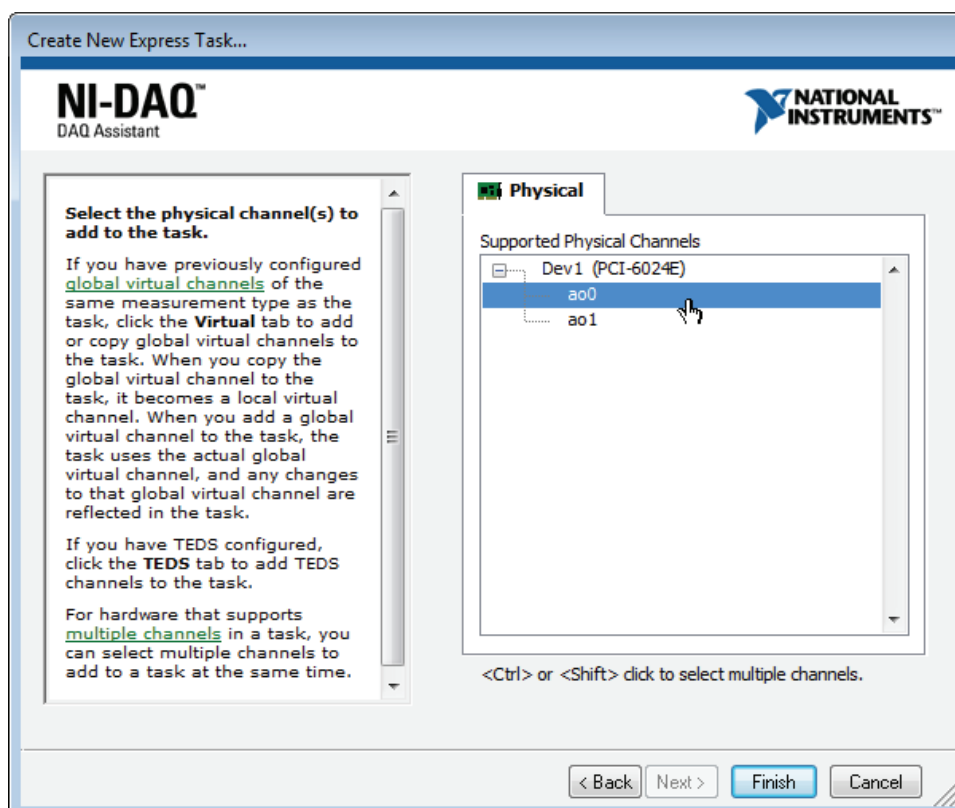


### Namestitev in nastavitev funkcije DAQ Assistant

1. V blok diagram namestite funkcijo *DAQ Assistant* (paleta: *Measurement I/O > NI-DAQmx*).
2. V nastavitvenem oknu funkcije *DAQ Assistant* izberite *Generate Signals > Analog Output > Voltage* ter nato izberite gumb *Next*.

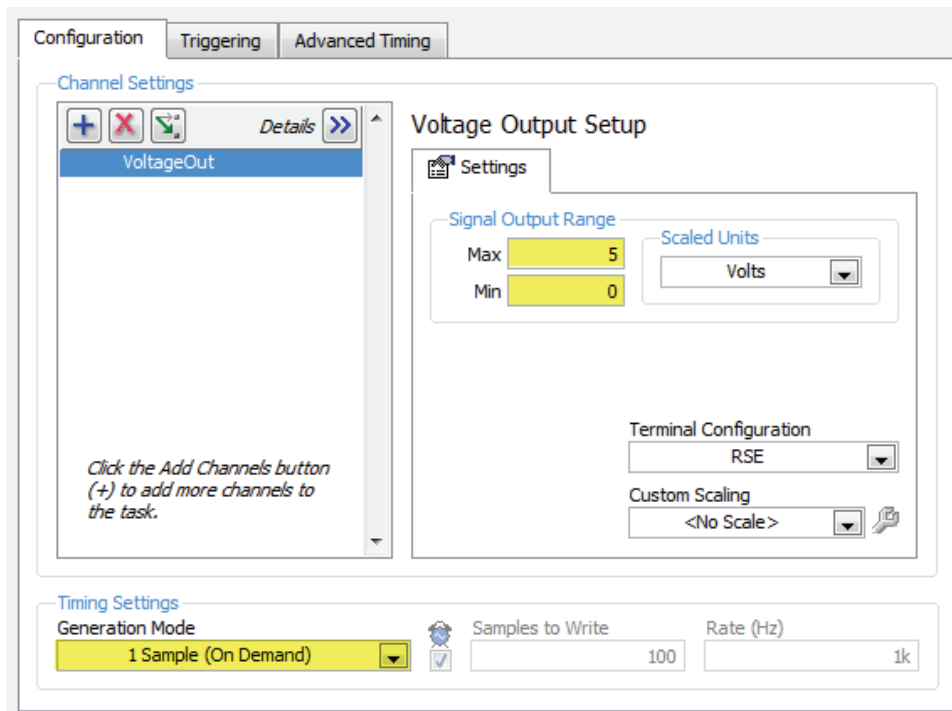


3. V okviru *Supported Physical Channels* izberite *Dev1 (PCI-6024E) > ao0* in izberite gumb *Finish*.

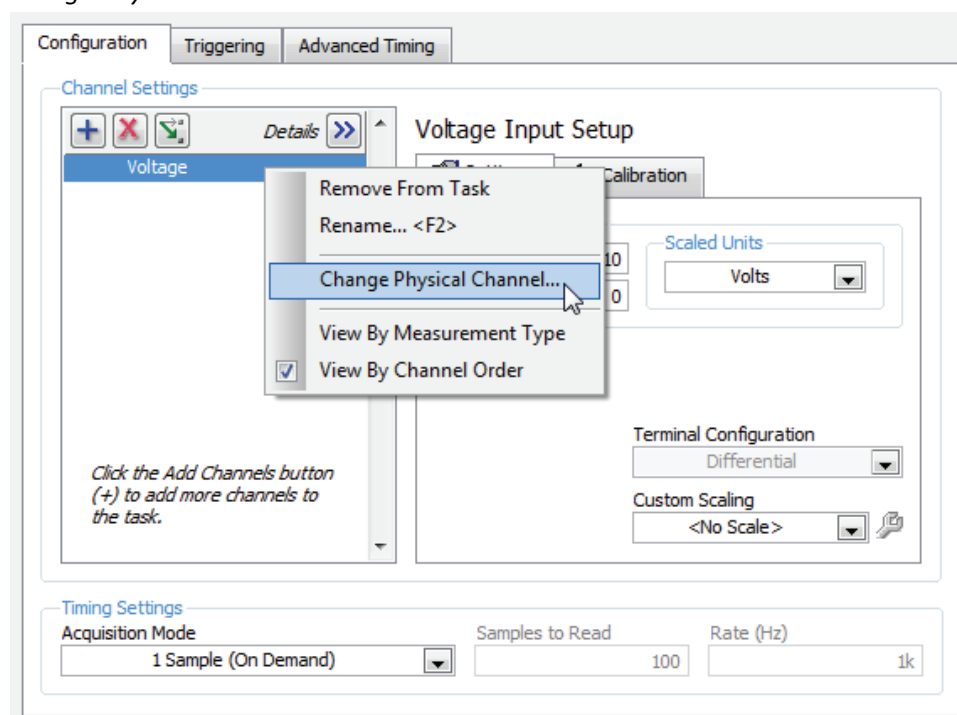


4. V zavihku *Configuration* nastavite:
- Voltage Input Setup
    - Signal Input Range

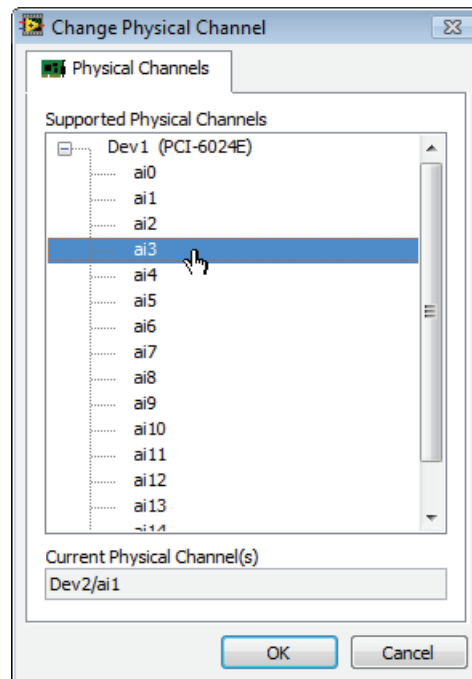
1. Max: 5
  2. Min: 0
- b) Timing Settings
- i. Generation Mode: 1 Sample (One Demand)



5. Izberite gumb OK in funkcija DAQ Assistant se bo v blok diagramu avtomatsko posodobila.
6. Odprite virtualni instrument *Voltmeter.vi* (vaja 8.1) in v blok diagramu dvokliknite na funkcijo *DAQ Assistant*. V razdelku *Channel Settings* se pomaknite na *Voltage* in v priročnem meniju izberite *Change Physical Channel ...*

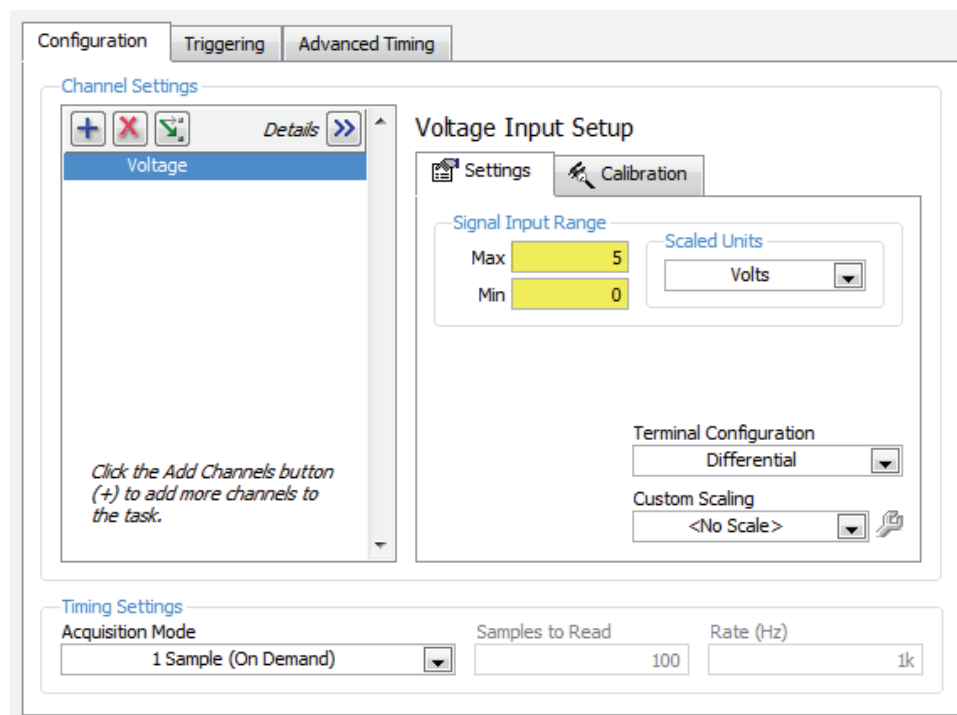


7. V okviru *Supported Physical Channels* okna *Change Physical Channel* izberite *ai3* in kliknite na gumb OK.



8. V zavihku *Configuration* nastavite nove meje kanala *Voltage*:

- a) Voltage Input Setup
- i. Signal Input Range
    1. Max: 5
    2. Min: 0

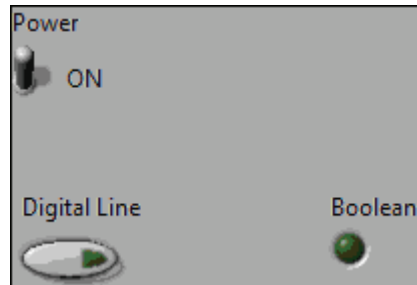


9. Hkrati poženite virtualna instrumenta *Voltage Output Example in Voltmeter*.

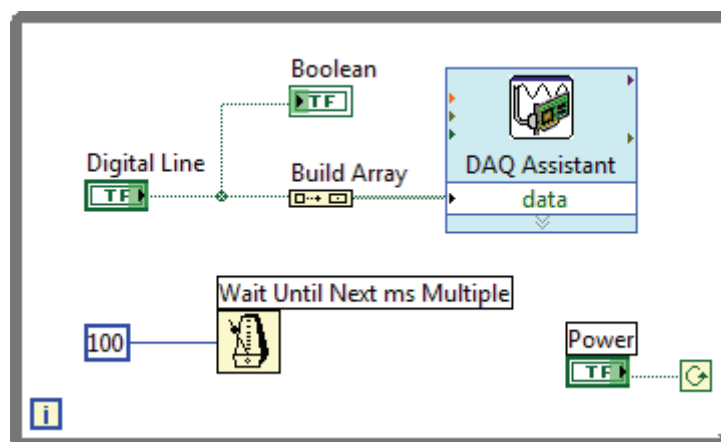
## Vaja 8.4: Digital Output

Izdelajte virtualni instrument, ki bo preko stikala čelne plošče vključil/izključil digitalni izhod 0 merilne kartice.

### Čelna plošča

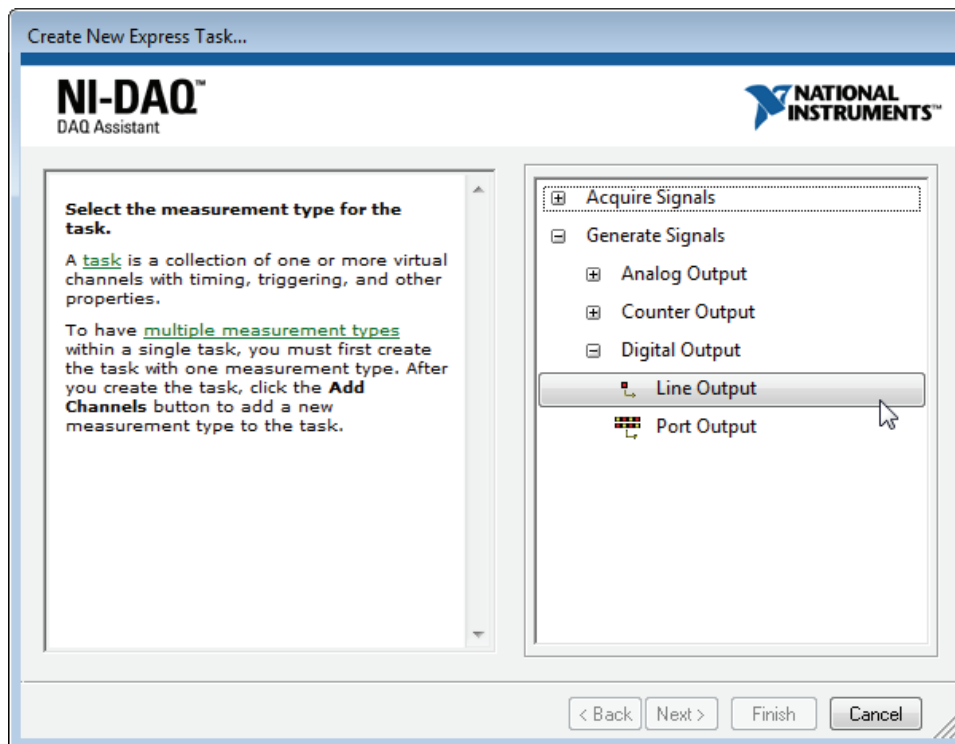


### Blok diagram

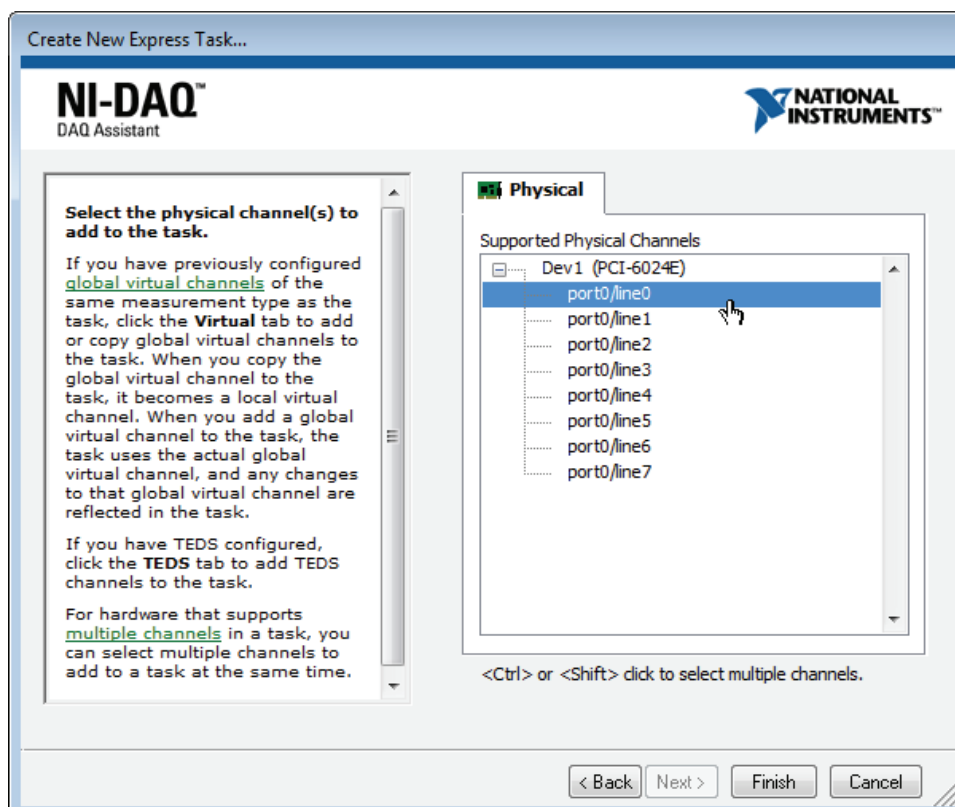


### Namestitev in nastavitev funkcije DAQ Assistant

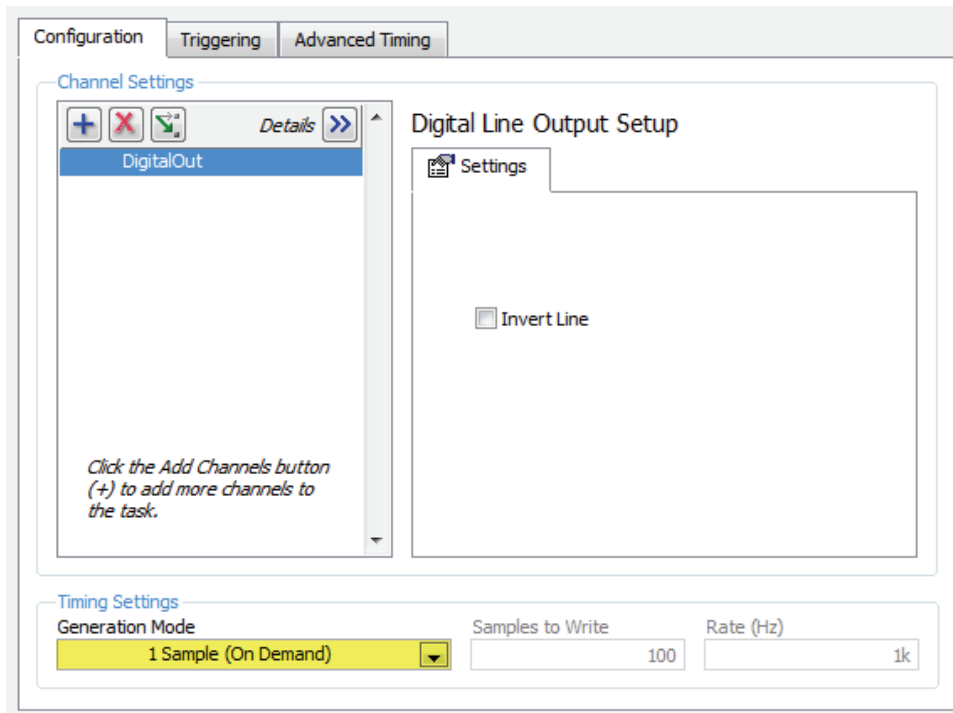
1. V blok diagram namestite funkcijo *DAQ Assistant* (paleta: *Measurement I/O > NI-DAQmx*).
2. V nastavitvenem oknu funkcije *DAQ Assistant* izberite *Generate Signals > Digital Output > Line output* ter nato izberite gumb *Next*.



3. V okviru *Supported Physical Channels* izberite *Dev1 (PCI-6024E) > port0/line0* in izberite gumb *Finish*.



4. V zavihku *Configuration* nastavite:
  - a) Timing Settings
    - i. Generation Mode: 1 Sample (One Demand)



5. Izberite gumb OK. Funkcija DAQ Assistant se bo v blok diagramu avtomatsko posodobila. Povežite vhod *data* preko funkcije *Build String* priključkom stikala čelne plošče in poženite virtualni instrument.



## Vaja 8.5: Temperature Control VI

Nadgradite virtualni instrument *Temperature Control* tako, da bo vklapljal in izklapljal žarnico na temperaturnem modulu v odvisnosti od trenutne in zelene temperature. Ko temperatura pade za več kot stopinjo pod želeno, naj se žarnica vključi in ostane vključena, vse dokler temperatura ne naraste za stopinjo čez želeno.

## Dodatek A: Paleta orodij

Paleta orodij je mogoče prikazati na dva načina, in sicer:

1. V meniju *View* izberemo opcijo *Tools Palette*. Paleta orodij ostane na zaslonu, vse dokler je ne zapremo.
2. Kjerkoli v praznem delu čelne plošče ali blok diagrama kliknemo na desni gumb miške (prikaz priločnega menija) in ob tem držimo SHIFT tipko. Paleta orodij se po izbiri orodja avtomatsko zapre.



### Orodje

### Opis



#### Operating Tool (Orodje za delo)

Uporablja za spreminjanje vrednosti kontrol in konstant.



#### Positioning Tool (Orodje za urejanje)

Uporablja za urejanje čelne plošče in blok diagrama. Orodje omogoča izbiro, premikanje in spreminjanje velikosti objektov.



#### Labeling Tool (Orodje za pisanje)

Omogoča pisanje tekstov in spreminjanje imen objektov.



#### Connecting Tool (Orodje za povezovanje)

Omogoča povezovanje objektov v blok diagramu.



#### Breakpoint (Prekinitvena točka)

Orodje omogoča začasno zaustavitev izvajanja virtualnega instrumenta. Prekinitvena točka se uporablja samo v blok diagramu. Ob kliku na povezavo se na povezavi pojavi rdeča pika (slika spodaj), ki ponazori, da se bo ob izvajanju tega dela kode virtualni instrument začasno zaustavil. Prekinitvena točka se odstrani tako, da se z istim orodjem ponovno klikne na povezavo, na kateri se prekinitvena točka že nahaja ali z uporabo orodja *Breakpoint Manager*.

Dodajanje

prekinitvene  
točke




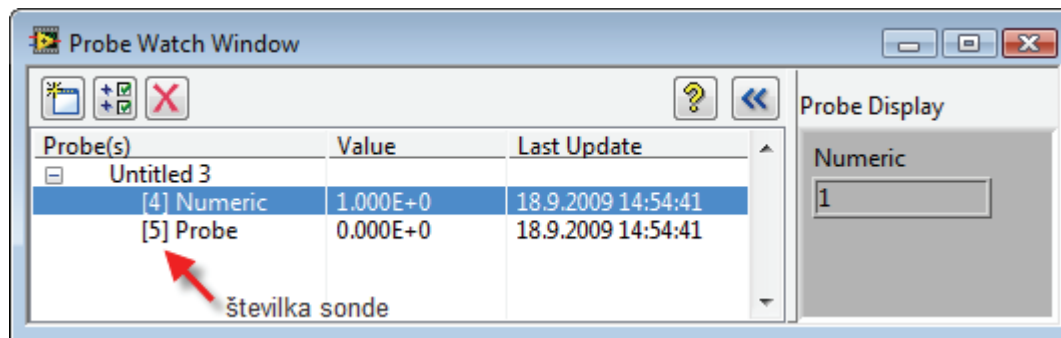
Odstranitev  
prekinitvene  
točke




#### Probe Data (Sonda)

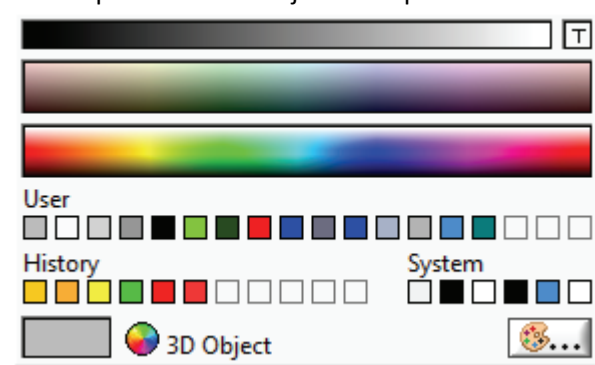
Sonda omogoča prikaz podatka, ki se prenese po izbrani povezavi v blok diagramu.

Sonda se uporablja samo v blok diagramu, namestiti pa jo je možno le na povezavo. Ob kliku na povezavo se pojavi dodatno okno (*Probe Watch Window*), v katerem se prikažejo vse uporabljene sonde v virtualnem instrumentu. Ob izbiri ustrezne sonde se sonda in njena vrednost prikažeta v indikatorju *Probe Display*. Za lažje ločevanje sond se ob namestitvi sonde na povezavi pojavi številka sonde. Sonda se odstrani tako, da se v oknu *Probe Watch Window* izbere ustrezna sonda, nato pa se izbere gumb *Remove Selected Probe* .



#### Coloring Tool (Orodje za barvanje)

Omogoča spreminjanje barv. Z orodjem se pomaknemo na objekt ali del objekta, ki bi mu radi spremenili barvo, in kliknemo na desno tipko miške. Ob tem se pojavi barvna paleta (slika spodaj). S kurzorjem miške se pomaknemo na želeno barvo in jo potrdimo s klikom na levo tipko miške. Izbira oznake  v zgornjem desnem kotu barvne palete naredi objekt transparenten.



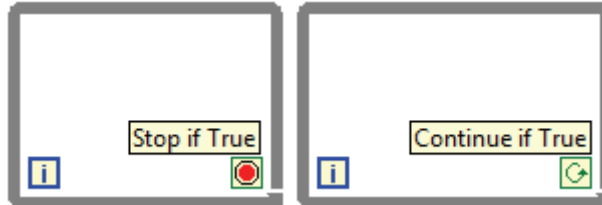
## Dodatek B: Programske strukture



### While loop (Zanka While)






**While loop** (Zanka While)

**Paleta:** Functions > Programming > Structures

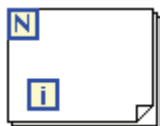


**Opis:** Zanka While omogoča ponavljajoče izvajanje dela kode, ki se nahaja znotraj območja zanke. Zanka While se izvaja, dokler je izpolnjen določen pogoj, in se vedno izvede vsaj enkrat. Na pogojni priključek (  ali  ) mora biti pripeljan podatek tipa Boolean. Glede na nastavev pogojnega priključka in vhodnega podatka velja sledeče:

- Če je pogojni priključek nastavljen na  (Stop if True), se izvajanje zanke ustavi, če je vhodni podatek enak TRUE, sicer se izvajanje zanke nadaljuje.
- Če je pogojni priključek nastavljen na  (Continue if True), se izvajanje zanke nadaljuje, če je vhodni podatek enak TRUE in konča, ko je enak FALSE.

Priključek  predstavlja števec iteracij oz. ponovitev zanke in teče od 0 naprej.

### For Loop (Zanka For)



**For Loop** (Zanka For)

**Paleta:** Functions > Programming > Structures

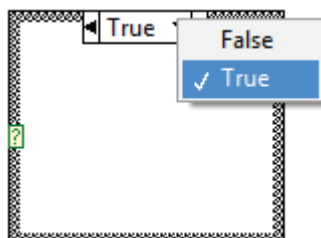
**Opis:** Zanka For omogoča ponavljajoče izvajanje kode, ki se nahaja znotraj območja zanke. Zanka se ponovi N-krat, pri čemer je N celoštevilčna vrednost priključena na N priključek zanke. Priključek (i) je števec, ki se poveča po vsaki iteraciji zanke in teče od 0 do N-1.


### Case Structure (Programska struktura Case)




**Case Structure** (Programska struktura Case)

**Paleta:** Functions > Programming > Structures



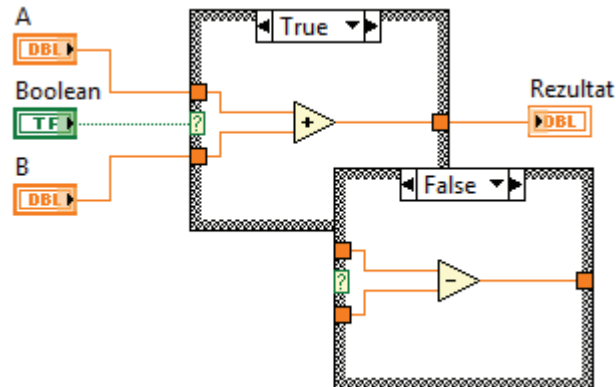
**Opis:** Programska struktura Case omogoča razdelitev algoritma v najmanj dve veji. Privzeto se ob namestitvi strukture kreirata dve veji oz. dva okvira (True in False), ki sta nameščena drug nad drugim. Veja, po kateri se nadaljuje izvajanje algoritma, je odvisna od podatka, pripeljanega na izbirni priključek  (*selector terminal*) in

podatka, vnesenega v selektor strukture Case . LabVIEW izvede tisto vejo algoritma, ki ustreza vhodnemu podatku na izbirnem priključku. Če nobena veja ne ustreza vhodnemu podatku, pa se izvede privzeta veja (*Default*).

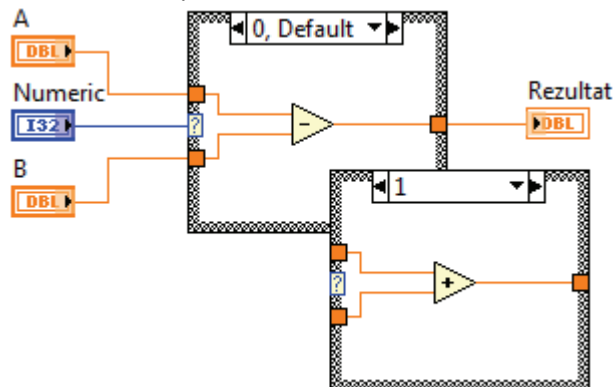
Veje (okvire) lahko dodajamo ali odstranimo s pomočjo opcij v priročnem meniju strukture.

### Primeri uporabe programske strukture Case

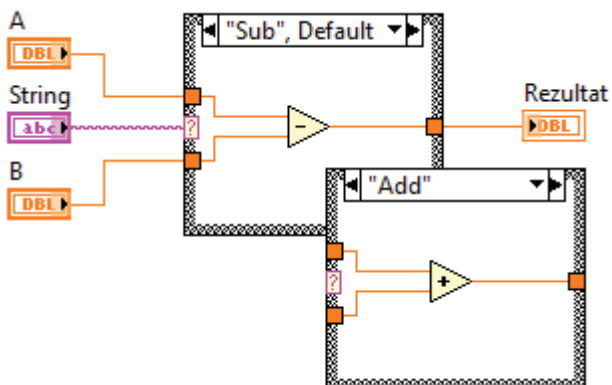
#### 4. Podatek tipa Boolean



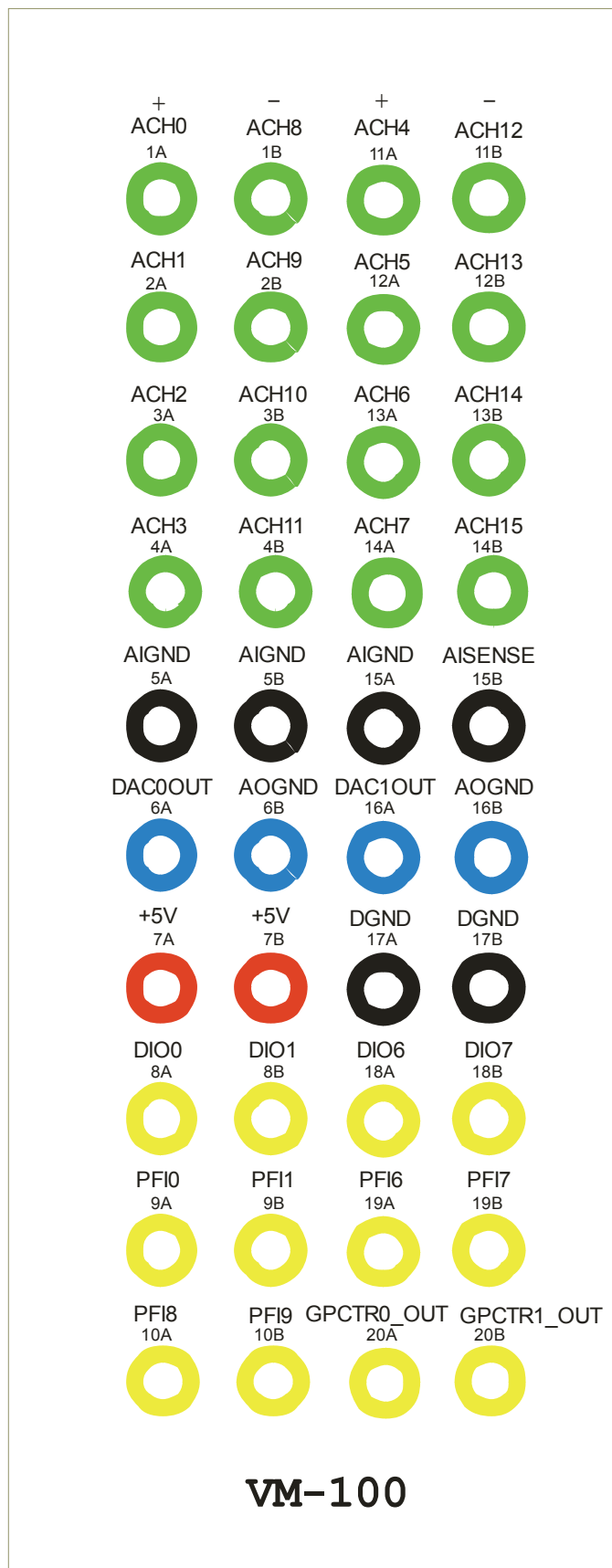
#### 5. Numerični podatek



#### 6. Znakovni niz

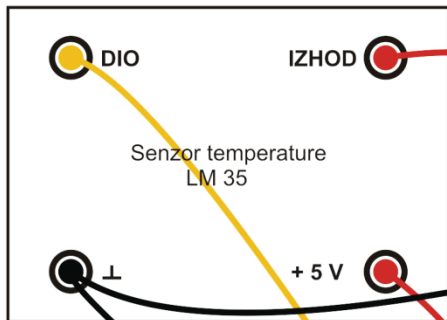


## Dodatek C: Priključna plošča NI PCI-6024E

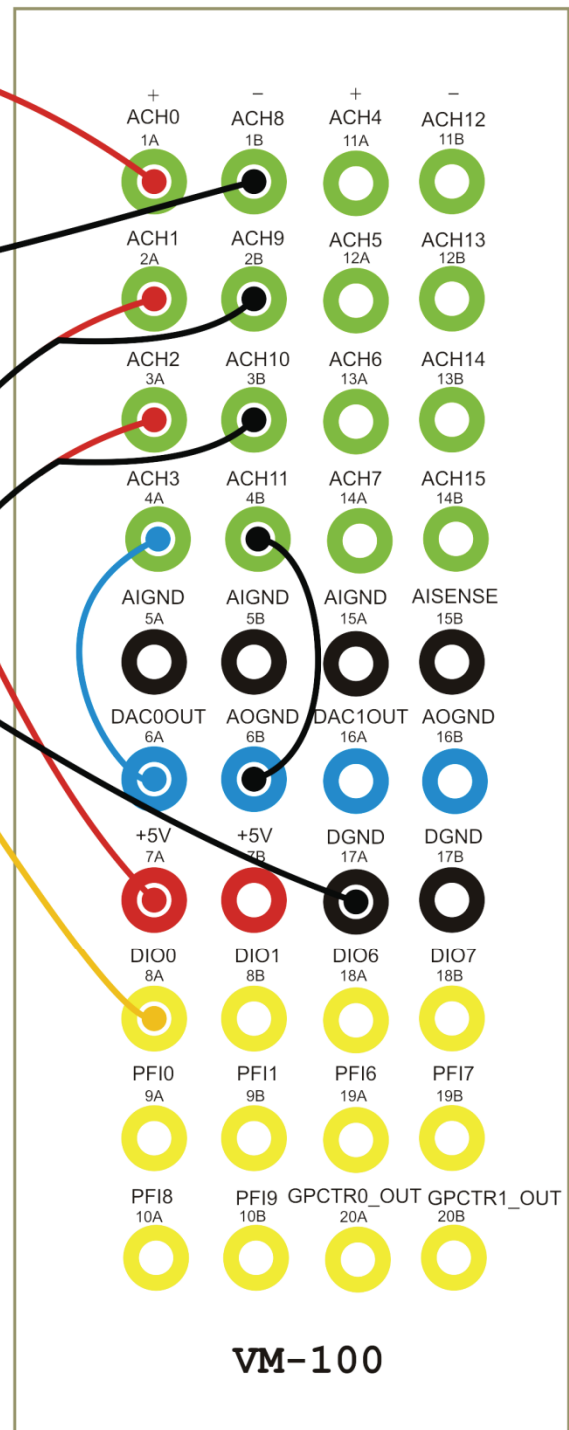


## Dodatek D: Vežalni načrt

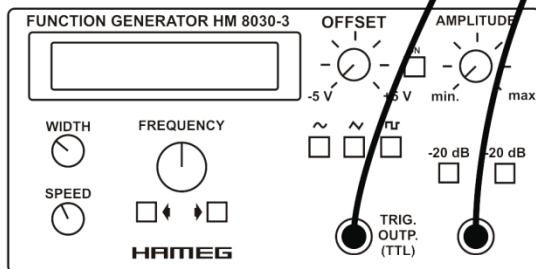
### TEMPERATURNI MODUL



### PRIKLJUČNA PLOŠČA NI PCI-6024E



### FUNKCIJSKI GENERATOR



## Dodatek E: LabVIEW slovarček

Align Objects	Poravnava objektov
Array Subset	Del polja
Block Diagram	Blokovni diagram
Breakpoint	Prekinitvena točka
Bundle	Grupiranje
Case Structure	Programska struktura Case
Cluster	Gruča
Coloring Tool	Orodje za barvanje
Connecting Tool	Orodje za povezovanje
Connector Pane	Priključna plošča
Connector Pattern	Priključni vzorec
Control	Kontrola
Časovne nastavitve	Timing Settings
Data Queue	Podatkovna vrsta
Default value	Privzeta vrednost
Digital Display	Digitalni prikaz
Distribute Objects	Porazdelitev objektov
For Loop	Zanka For
Front panel	Čelna plošča
Graf Palette	Graf paleta
Grid Color	Barva mreže
Highlight Execution	Poudarjeno izvajanje
Icon	Ikona
Indicator	Indikator
Label	Labela
Labeling Tool	Orodje za pisanje
Message Queue	Sporočilna vrsta
Notifier	Objavitelj
Numeric constant	Numerična konstanta
Operating Tool	Orodje za delo
Panning tool	Orodje za slikovno drsenje
Plot Legend	Legenda izrisa
Point Style	Izgleda točke
Positioning Tool	Orodje za urejanje
Precision	Število decimalnih mest izpisa
Probe Data	Sonda
Scale Legend	Legenda skale
Select tool	Orodje za izbiro
Selector terminal	Izbirni priključek
Shift Register	Pomikalni register
Single Step Execution	Izvajanje po korakih
String	Znakovni niz
String Length	Dolžina znakovnega niza
SubVI	Podprogram
Terminal	Priključek



---

Text Setting	Orodje za nastavitve pisave
True Constant	Konstanta tipa Boolean nastavljena na vrednost TRUE
Wait Until Next ms Multiple	Metronom
While loop	Zanka While
X Scrollbar	Drsnik po x osi
X-zoom	Zumiranje po x osi
Y-zoom	Zumiranje po y osi
Zoom In about Point	Povečevanje v okolici točke
Zoom Out about Point	Pomanjševanje v okolici točke
Zoom to Fit	Prilagajanje zuma podatkom
Zoom to Rectangle	Okensko zumiranje
Zooming tool	Orodje za zumiranje

ISBN 978-961-248-220-6



9 789612 482206