

LabView

4

Dr Piotr Sitarek

Katedra Fizyki Doświadczalnej, Politechnika Wroclawska

Temat na dziś

Dokumentacja, wizualizacja danych

ni.com



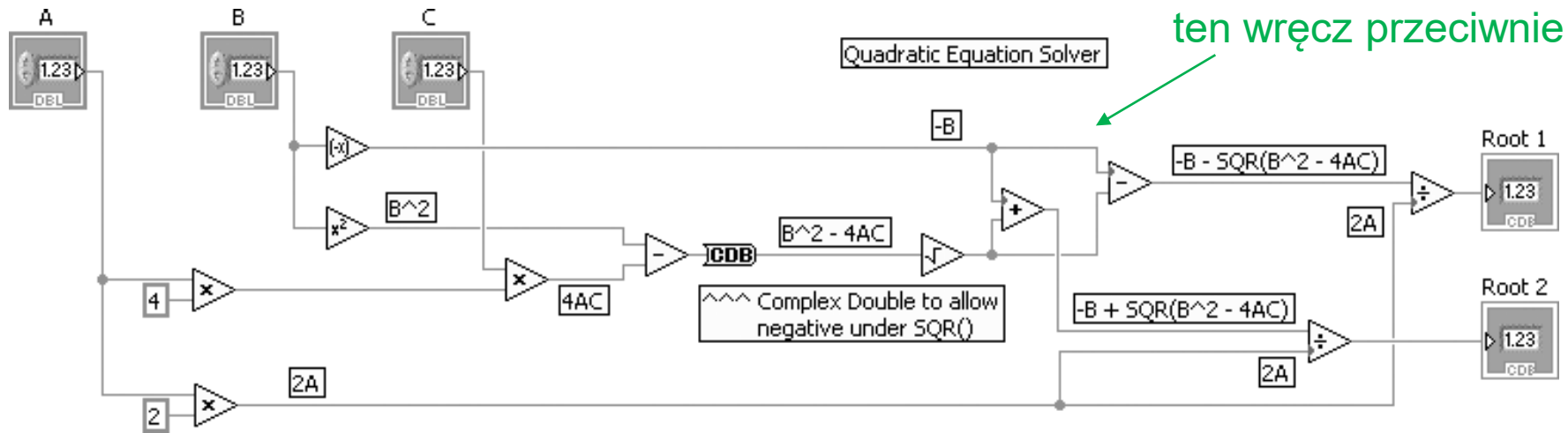
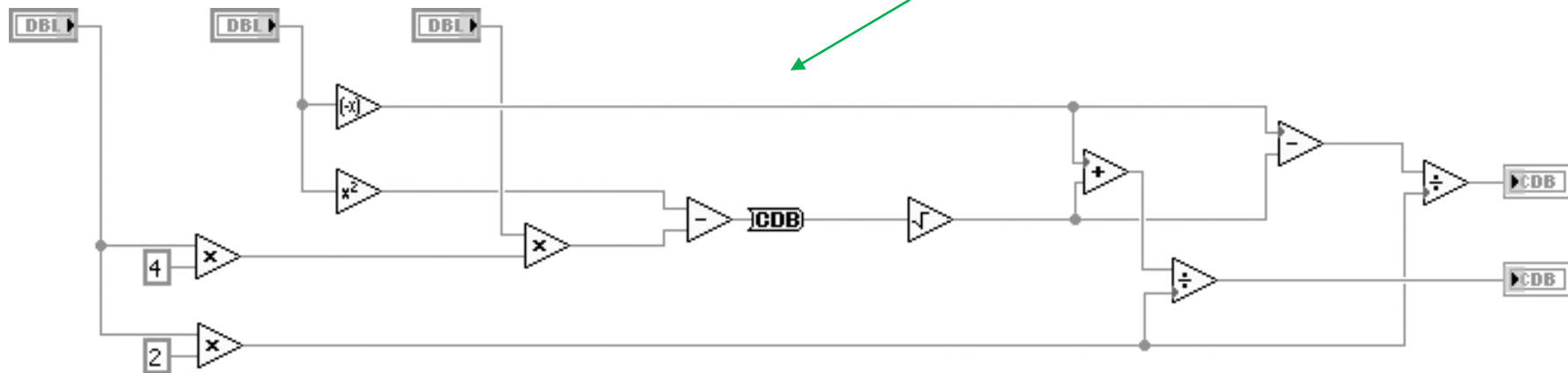
(część materiałów zaczerpnięta ze strony
producenta)

RONALD W. LARSEN, *LabVIEW for Engineers*

Dokumentacja

Opis BD

ten kod niewiele mówi



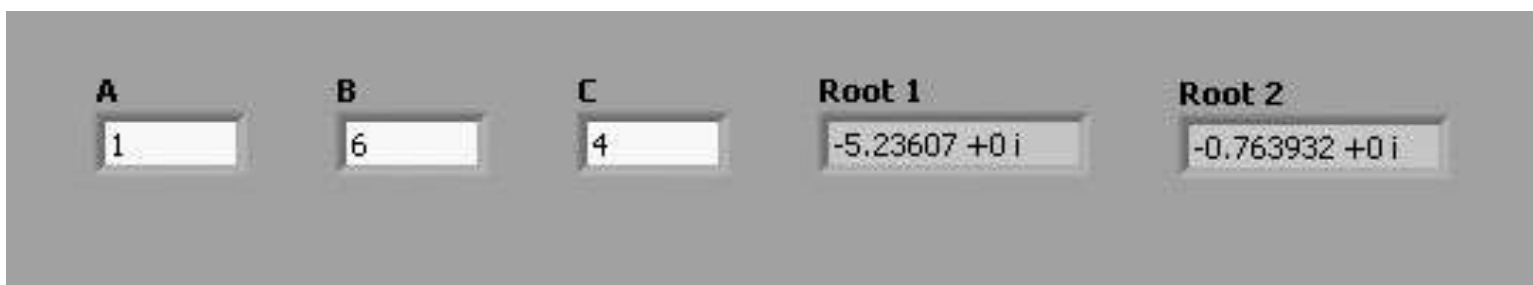
Dokumentacja

Opis FP

taki Front Panel jest nieczytelny



1 6 4 -5.23607 +0 i -0.763932 +0 i



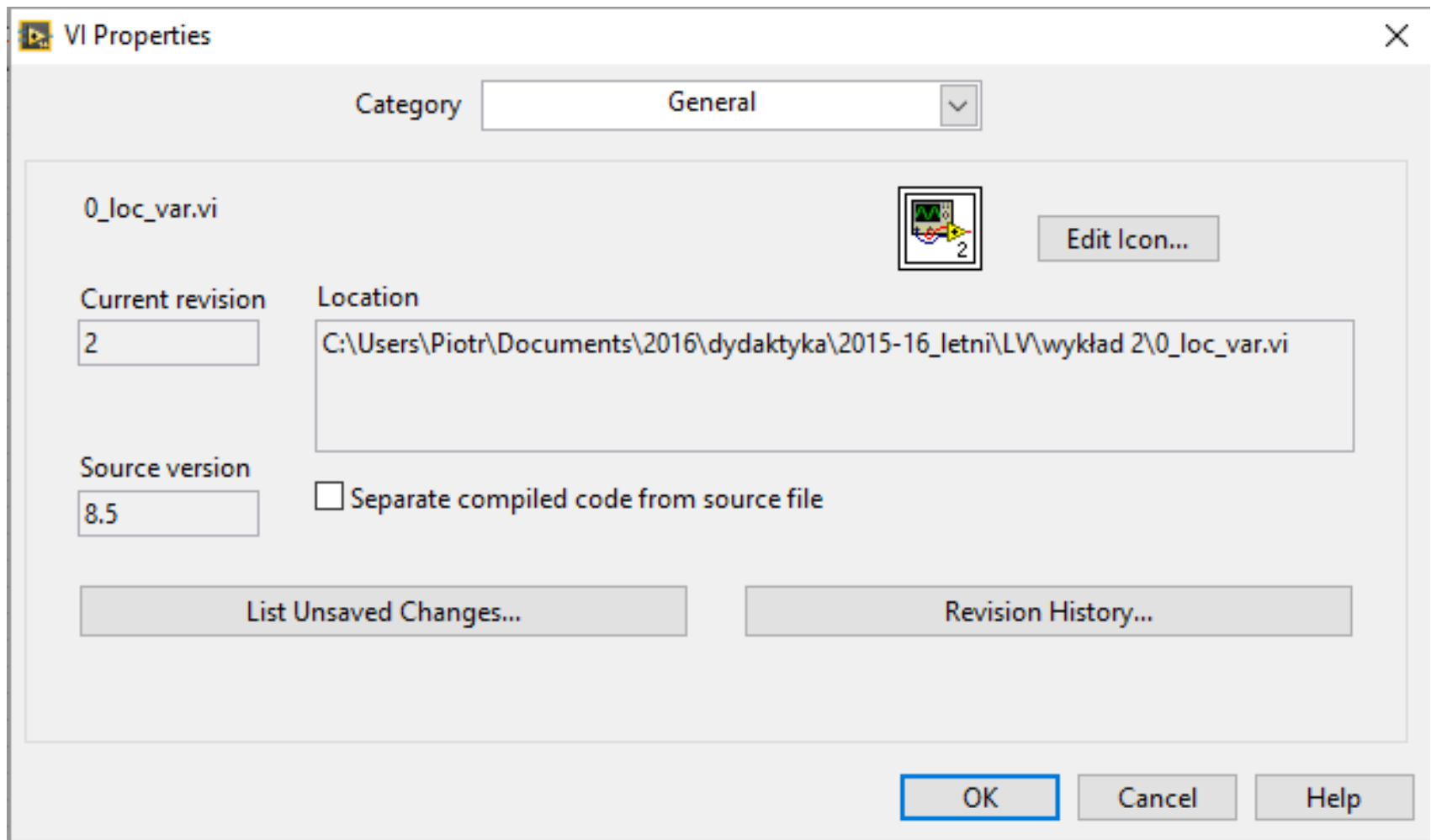
A B C Root 1 Root 2
1 6 4 -5.23607 +0 i -0.763932 +0 i

Quadratic Equation Solver				
Quadratic Coefficients			Calculated Results	
A	B	C	Root 1	Root 2
1	6	4	-5.23607 +0 i	-0.763932 +0 i

$$A x^2 + B x + C = 0$$

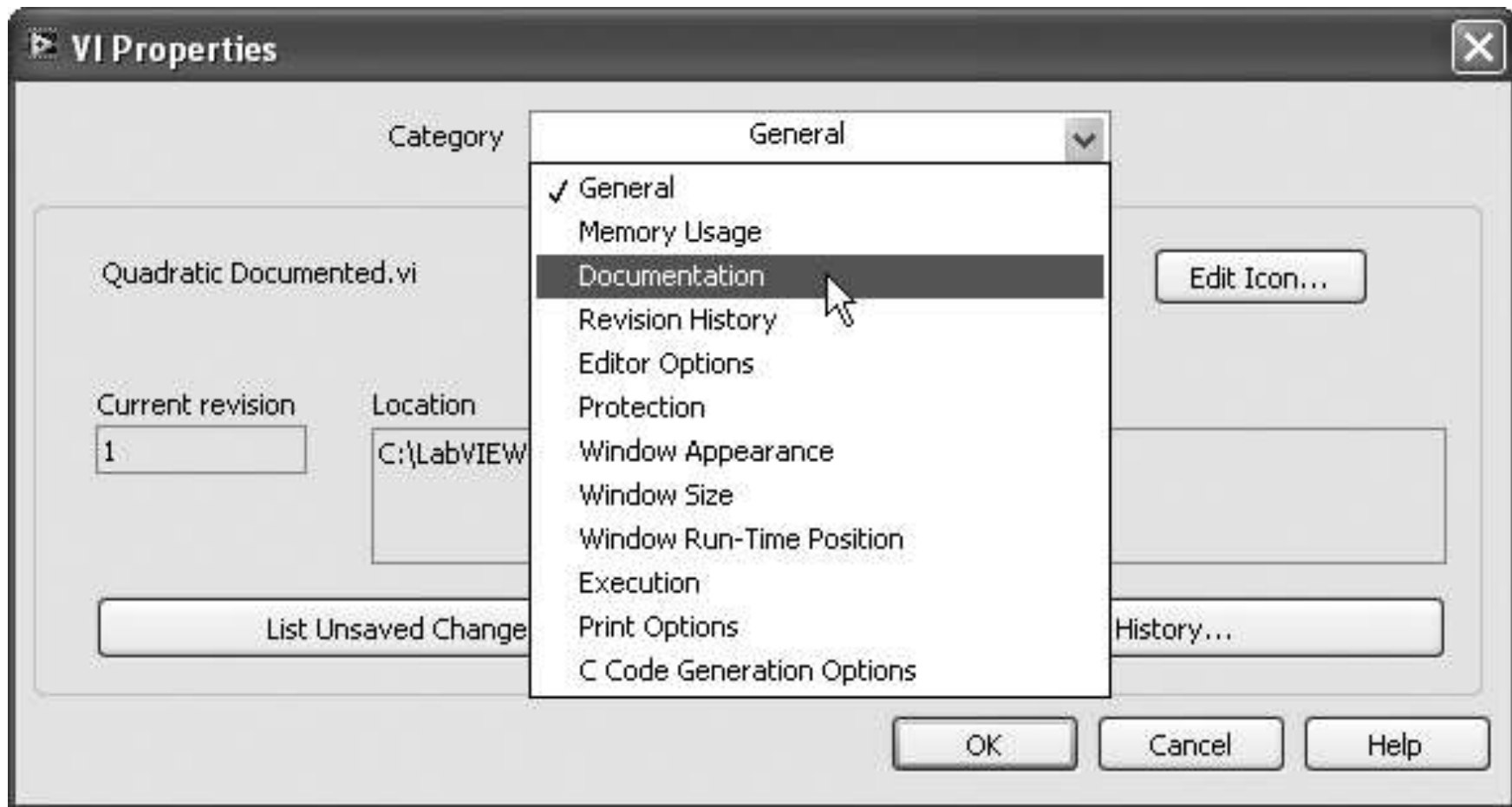
Dokumentacja

■ Informacje opisowe - **File / VI Properties**



Dokumentacja

- Informacje opisowe - **File / VI Properties**



Dokumentacja

■ Documentation:

- Tytuł
- Autor i data
- składnia i co VI ma robić
- informacje o typach danych
- edycja tekstu

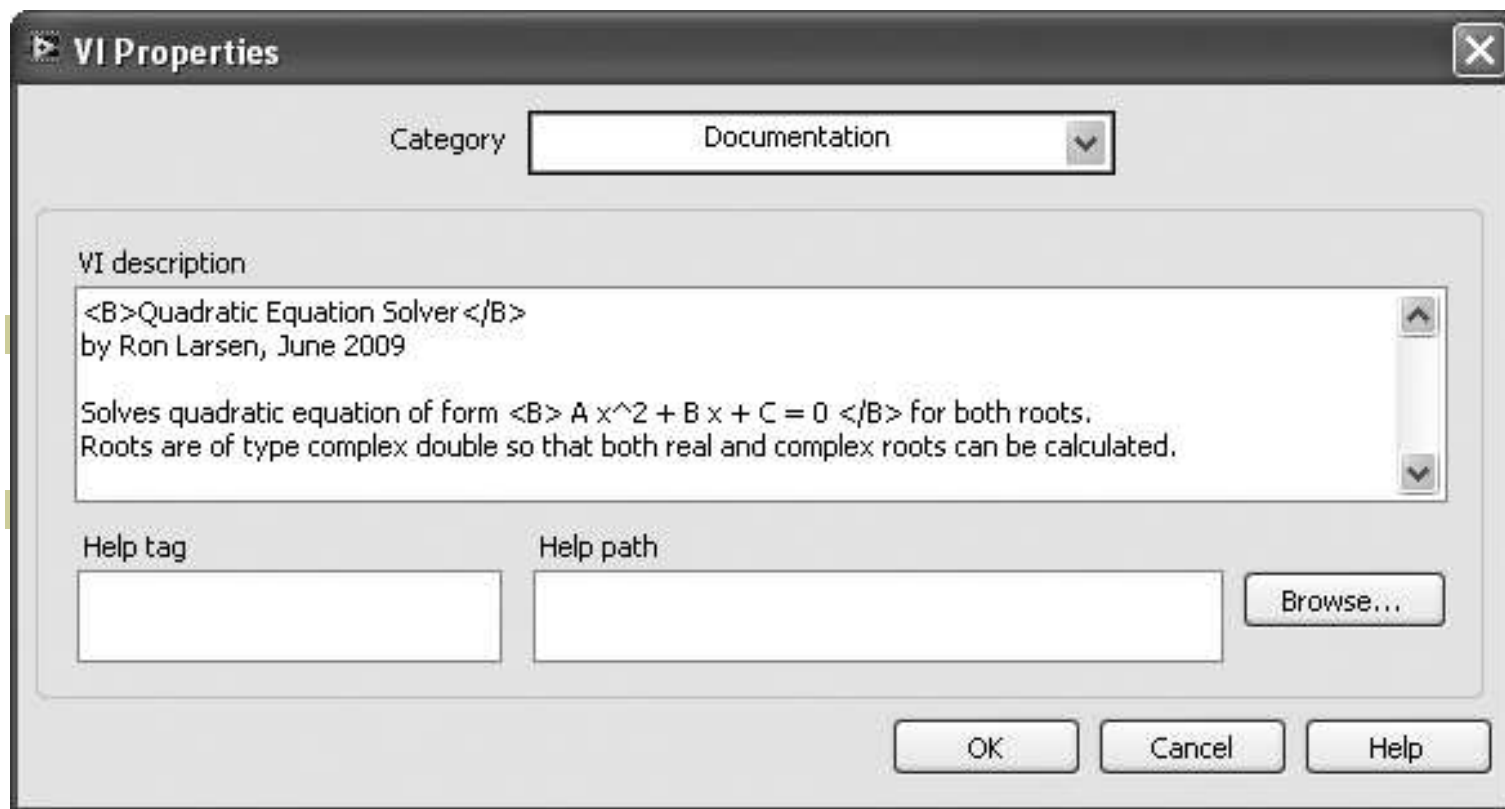
■ **Help tag** - nowe określenia do indeksu (związane z VI)

■ **Help path** – ścieżka do pliku HTML help file (**Detailed Help** pojawi się w Context Help window)

Dokumentacja

■ Documentation:

- Tytuł
- Autor i data




Dokumentacija

■ Context Help


Context Help

Quadratic Documented.vi



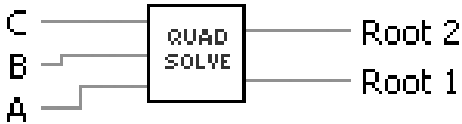
Quadratic Equation Solver
by Ron Larsen, June 2009

Solves quadratic equation of form $Ax^2 + Bx + C = 0$ for both roots.
Roots are of type complex double so that both real and complex roots can be calculated.



Context Help

Quad Solve.vi



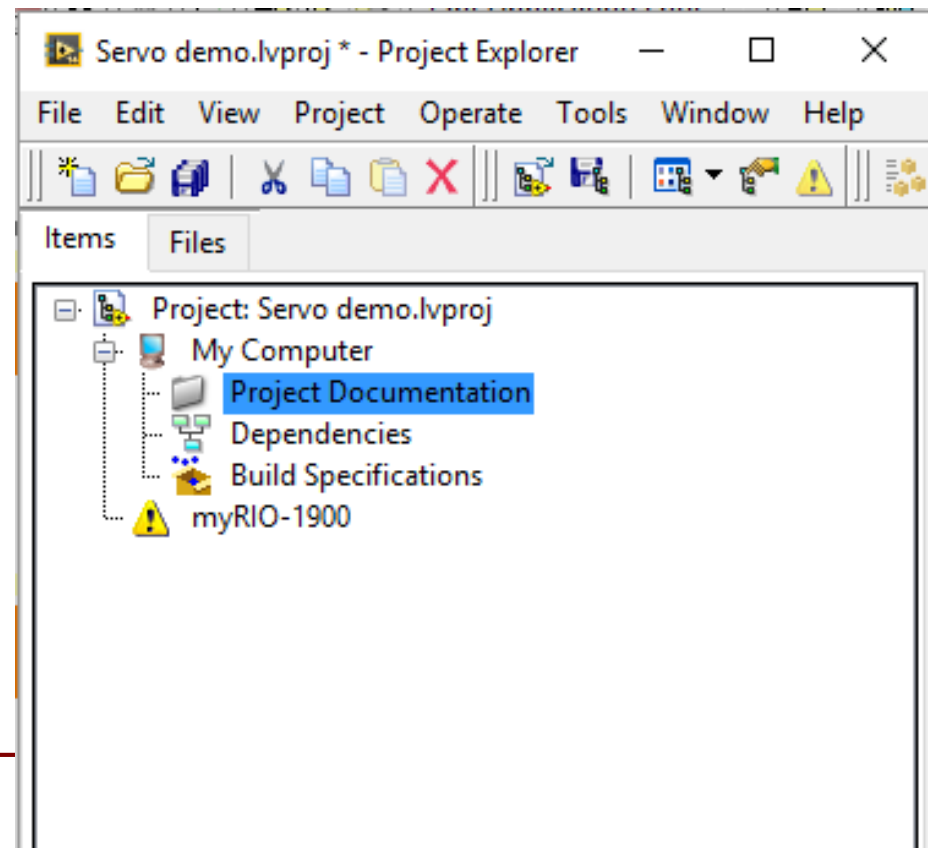
Quadratic Equation Solver
by Ron Larsen, June 2009

Solves quadratic equation of form $Ax^2 + Bx + C = 0$ for both roots.
Roots are of type complex double so that both real and complex roots can be calculated.

Projekty

Projekt – zbiór wszystkich VIs i SubVIs niezbędnych do działania.

Project / New Project . . . - Add



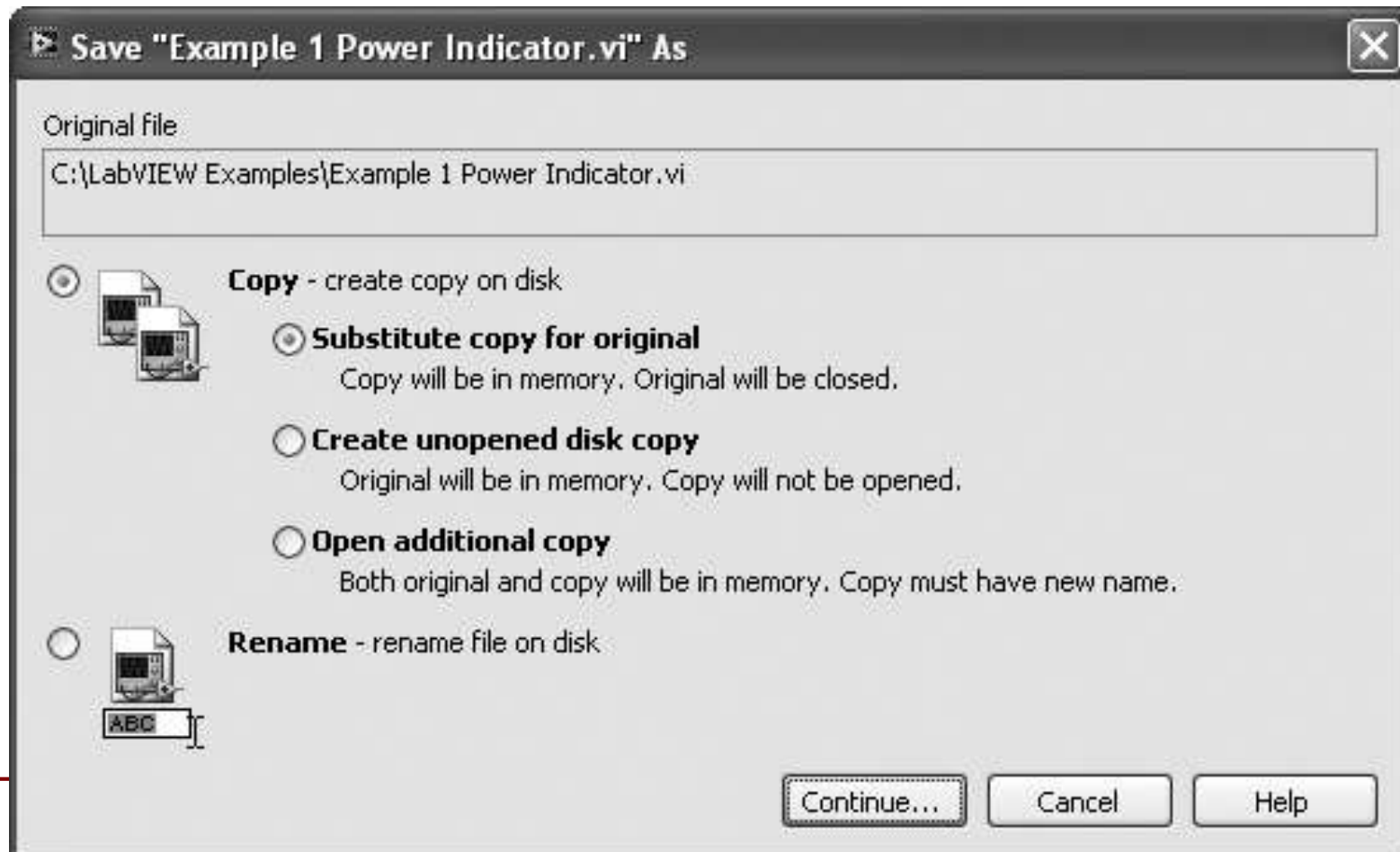
Dokumentacja

■ Zachowywanie

- **Save** – zachowuje aktualny VI.
 - **Save As . . .** (dialog box).
 - **Save All** – zachowuje wszystkie otwarte VI
 - **Save for Previous Version . . .** zachowuje w poprzedniej wersji
-

Dokumentacja

- **Zachowywanie**
 - **Save As . . .** (dialog box).

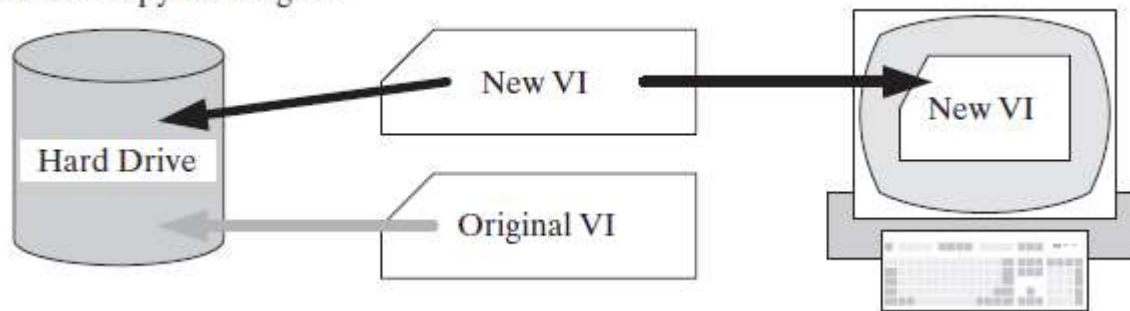


Dokumentacja

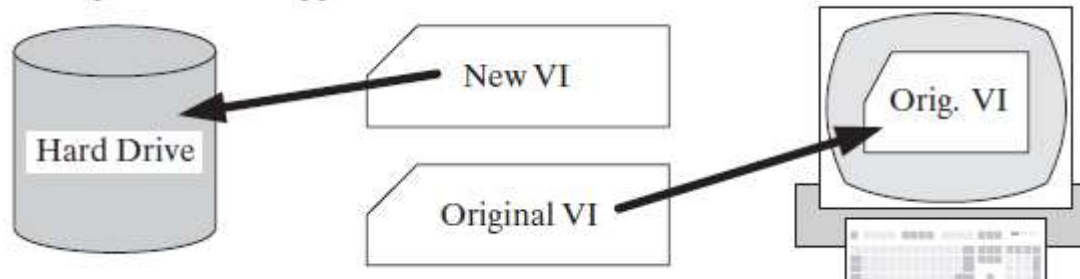
■ Zachowywanie

■ Save As . . . (dialog box).

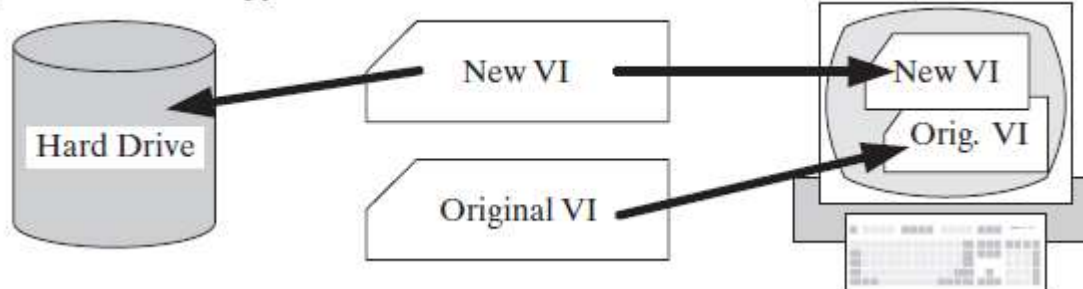
Substitute Copy for Original



Create Unopened Disk Copy

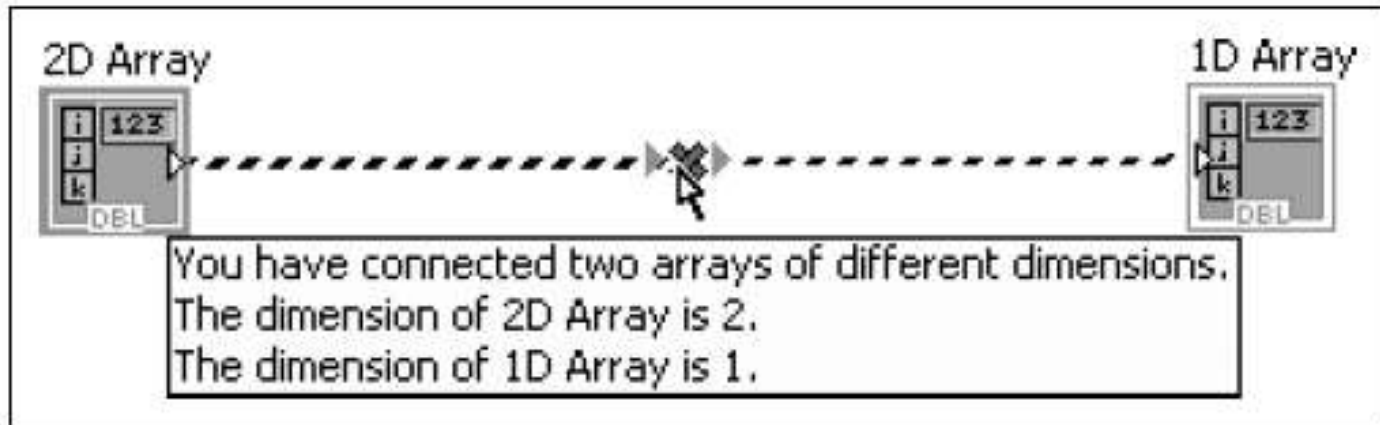


Open Additional Copy



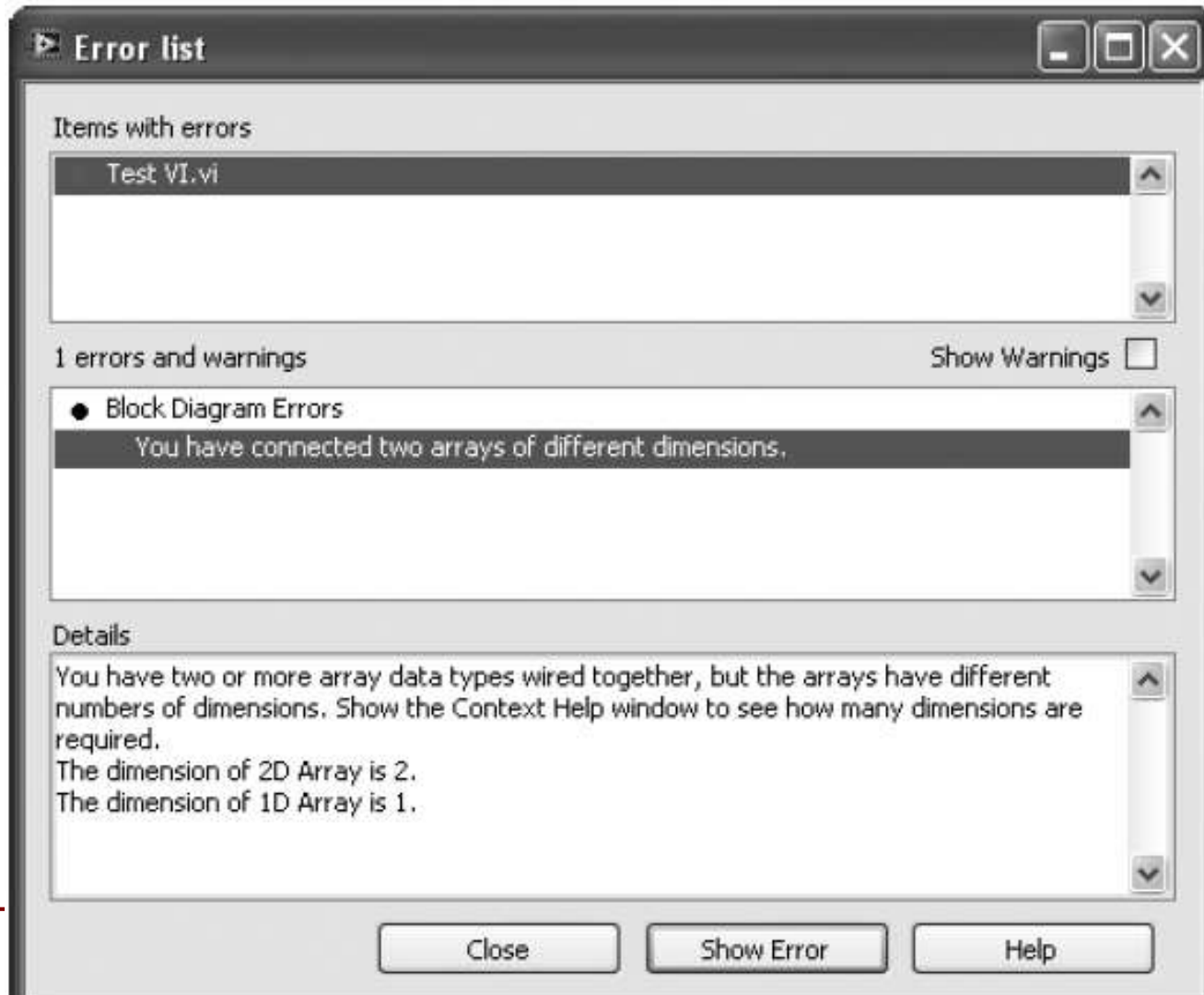
Programowanie

■ Debugowanie



Programowanie

■ Debugowanie



Programowanie

■ Debugowanie

The screenshot displays the LabVIEW software interface during a debugging session. At the top, a toolbar contains various icons for navigation and execution, including a font size dropdown set to 15pt. The main workspace shows a block diagram with three knobs labeled 'Knob', 'Knob 2', and 'Knob 3'. The 'Knob' is connected to an adder block (+), which is then connected to a numeric display labeled 'Wartość średnia' (Average Value) showing a value of 1.23. A 'Numeric' control is also visible on the right, showing a value of 1.23. A 'Probe Watch Window' is open in the foreground, displaying a table of probe data:

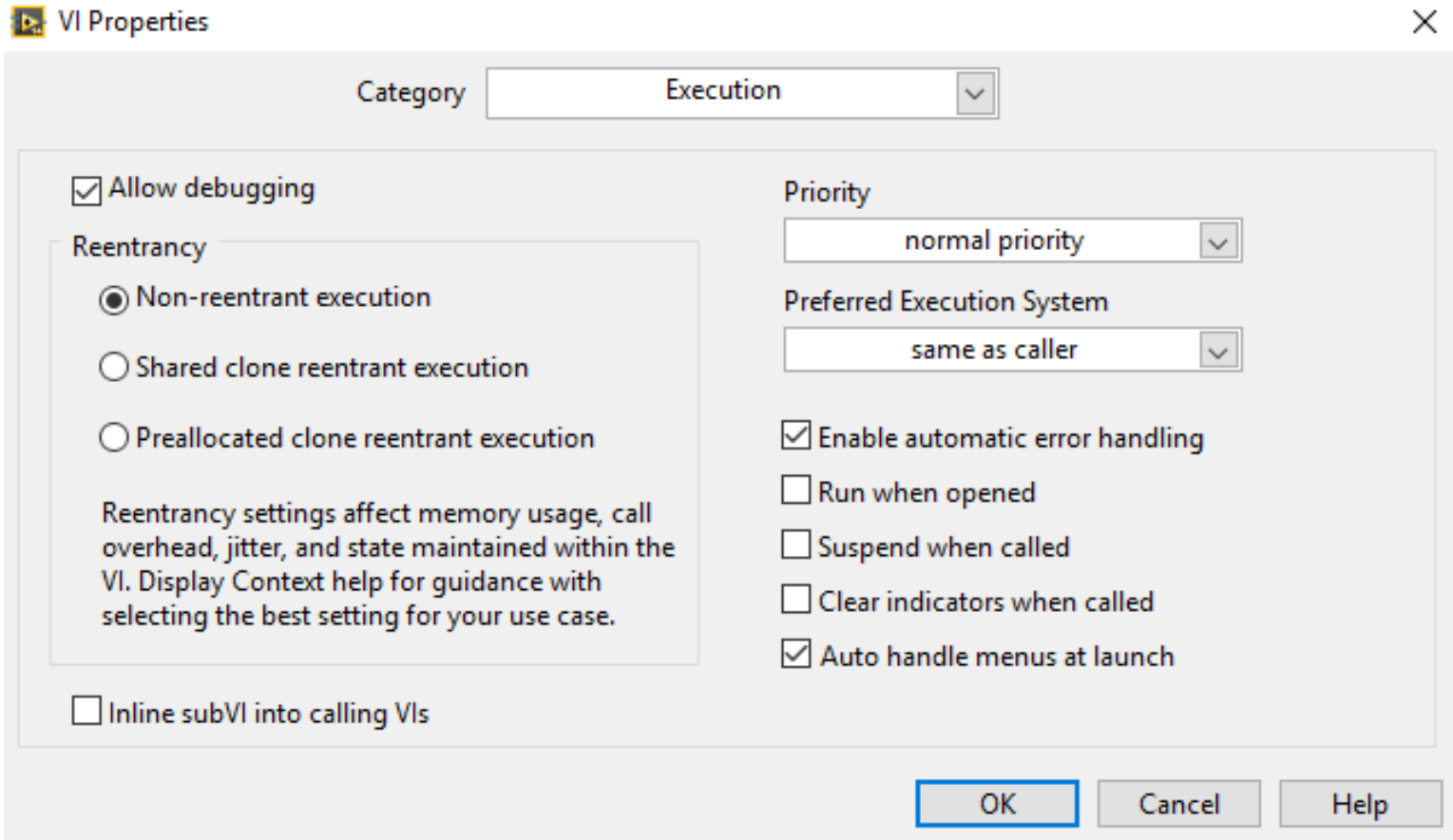
Probe(s)	Value	Last Update
0_loc_var.vi		
[1] Probe	0.000E+0	2016-03-13 21:12:27

The 'Probe Display' area on the right shows a numeric field with the value 0.

Wielokrotny dostęp do procedur

- normalnie dostęp do VI lub subVI jest jednokrotny – wielokrotne wywołanie tego samego kodu działa krokowo – musi zakończyć się „jeden dostęp” aby rozpoczął się kolejny
 - można to zmienić ustawiając tzw. reentrant execution – jednoczesny dostęp do danego VI lub subVI z różnych miejsc programu – każdy dostęp rezerwuje własny obszar pamięci – przyspiesza działanie programu ale obciąża pamięć
 - File / VI Properties
-

Wielokrotny dostęp do procedur



VI Properties

Category: Execution

Allow debugging

Reentrancy

- Non-reentrant execution
- Shared clone reentrant execution
- Preallocated clone reentrant execution

Reentrancy settings affect memory usage, call overhead, jitter, and state maintained within the VI. Display Context help for guidance with selecting the best setting for your use case.

Inline subVI into calling VIs

Priority: normal priority

Preferred Execution System: same as caller

- Enable automatic error handling
- Run when opened
- Suspend when called
- Clear indicators when called
- Auto handle menus at launch

OK Cancel Help

Biblioteki .LLB

Library Benefits and Drawbacks

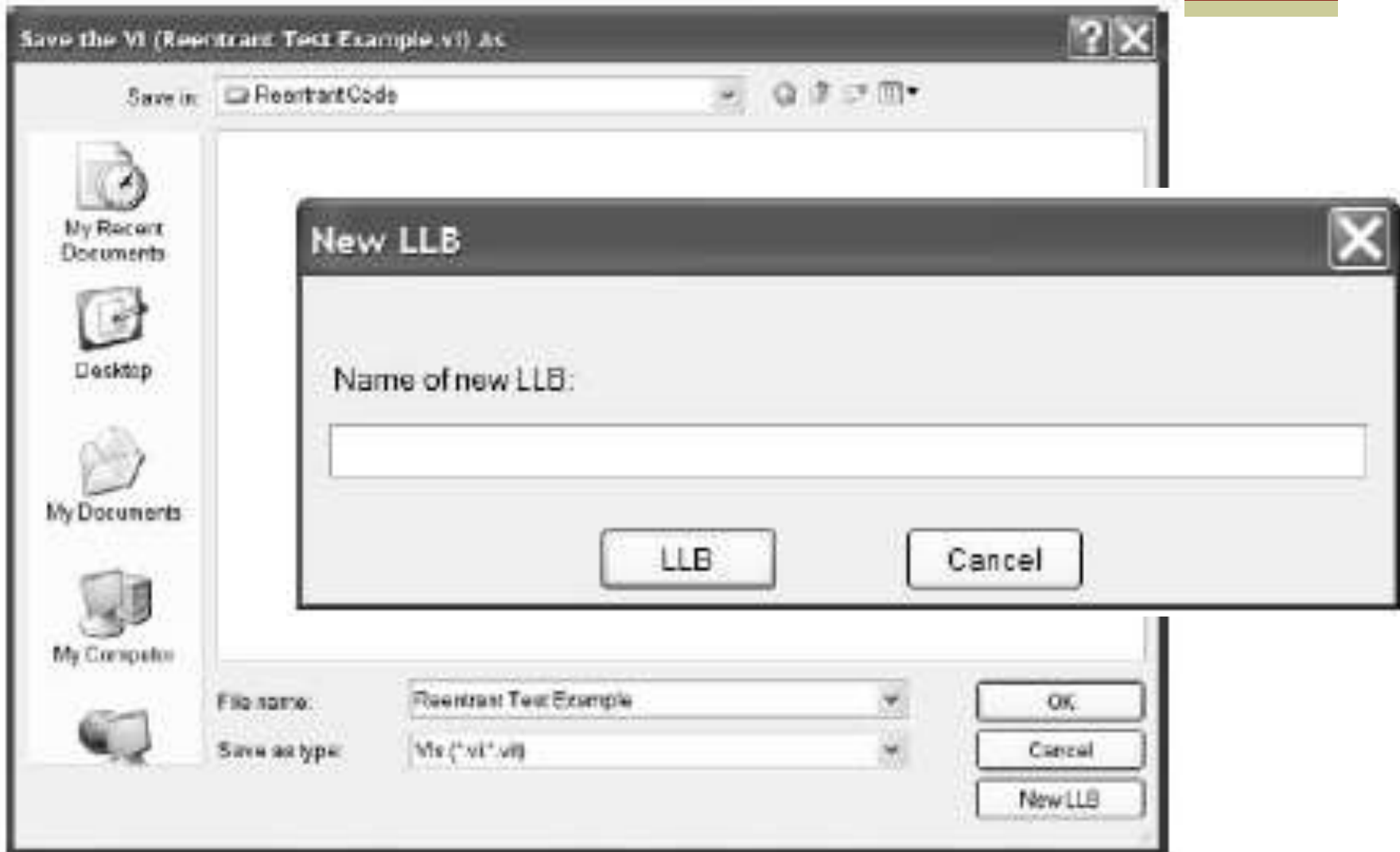
Benefits of Saving as .VI

1. You can use your own file system to manage and store VIs.
2. Libraries are not hierarchical. Libraries cannot contain subdirectories or sublibraries.
3. Loading and saving VIs is faster and requires less disk space for temporary files.
4. More robust than storing entire project in the same file.
5. There is the possibility of a library becoming corrupt.
6. Source Control cannot operate on individual VIs in a LLB, only the entire LLB.

Benefits of Saving as .LLB

1. 255 characters can be used for naming files (may be useful for Macintosh, where filenames are limited to 31 characters).
 2. Easier for transporting VIs to different platform or to a different computer
 3. Libraries are compressed and require less disk space.
 4. Can now be viewed in Windows Explorer (starting in LabVIEW 7).
 5. Can set one or several VIs to start when the library is opened (top level VIs).
-

Biblioteki .LLB

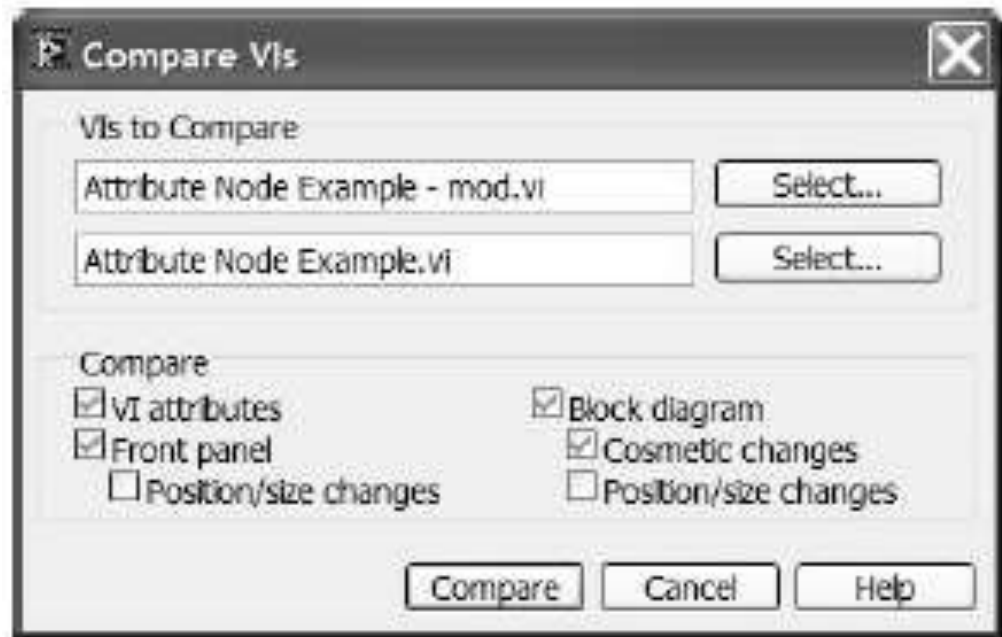


Biblioteki .LLB

- Save As ... VI Library
 - New Library
 - dodać do istniejącej biblioteki – llb są traktowane w Windows jako folder
 - Tools / LLB Menager
-

Porównywanie VIs

- Tools / Compare / Compare VIs
- / Show Differences
- / Compare VI Hierarchies



Tworzenie aplikacji

- Tools / Build Application (exe)

Basic Input Demo Properties

Category

Information

Source Files

Destinations

Source File Settings

Icon

Advanced

Additional Exclusions

Version Information

Windows Security

Shared Variable Deployment

Run-Time Languages

Web Services

Pre/Post Build Actions

Preview

Source Files

Project Files

My Computer

Basic Input Demo.vi

Startup VIs

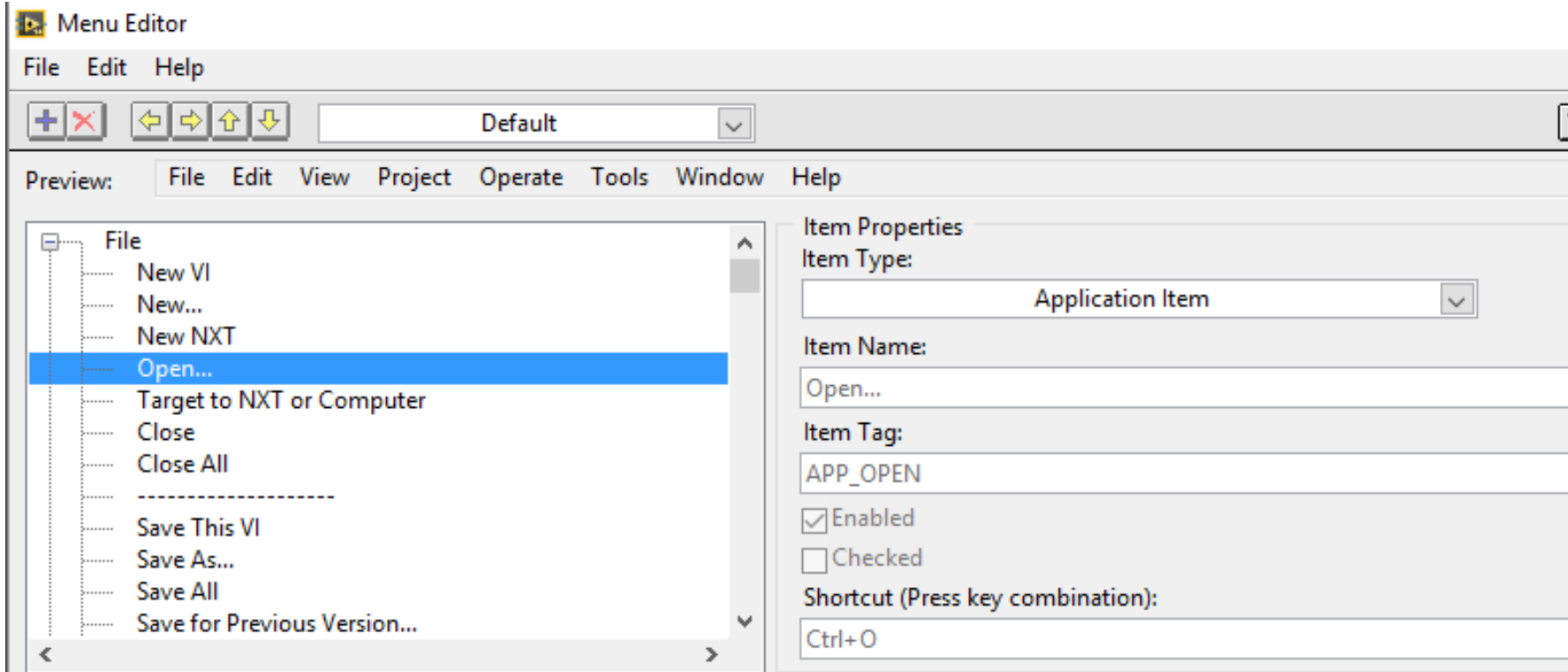
Basic Input Demo.vi



Always Included

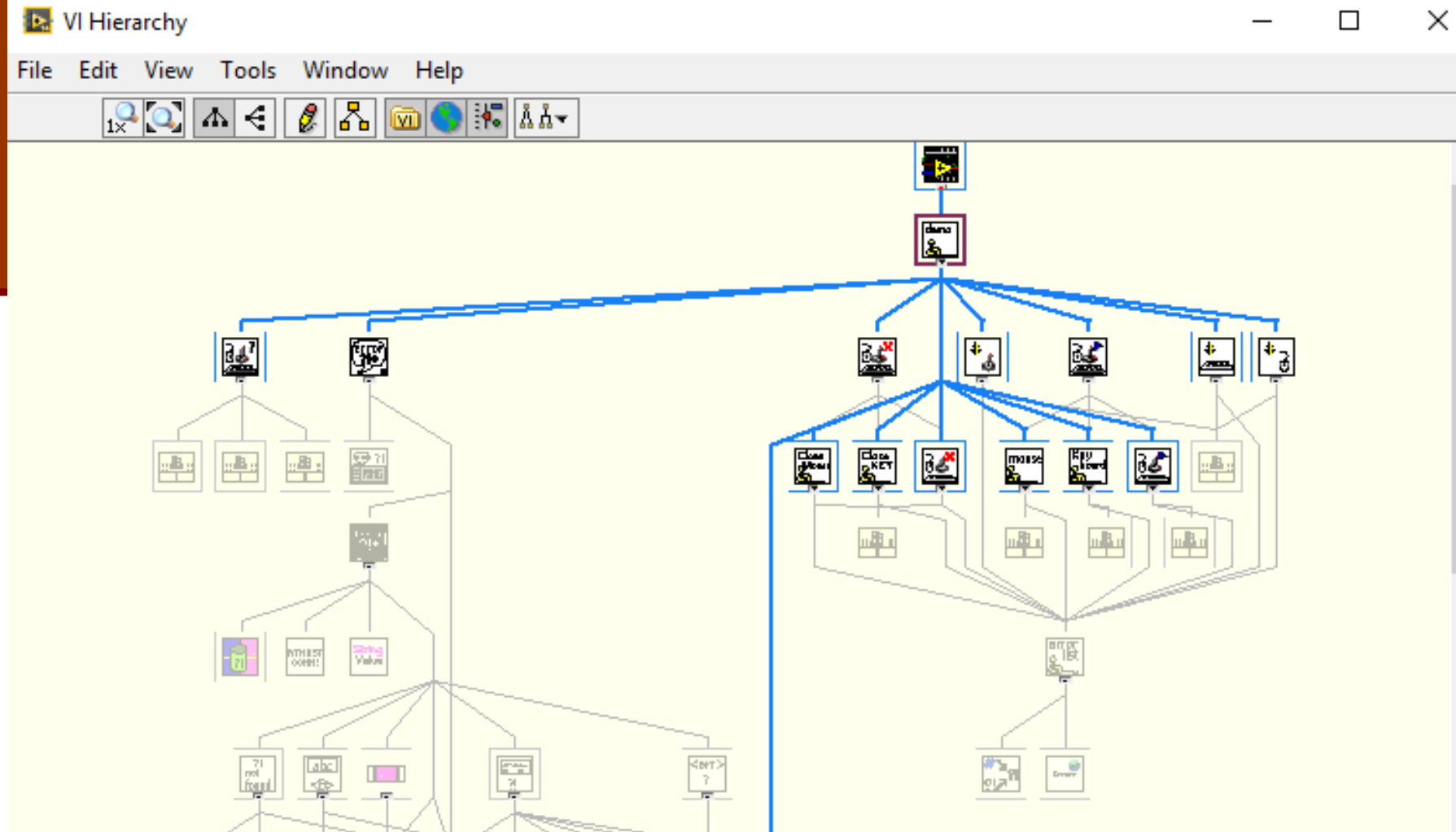
Menu Editor

Edit / Run-Time Menu



VI Hierarchy

View / VI Hierarchy



Wykresy

Chart Types

- Waveform Chart
- Intensity Chart

Graph Types

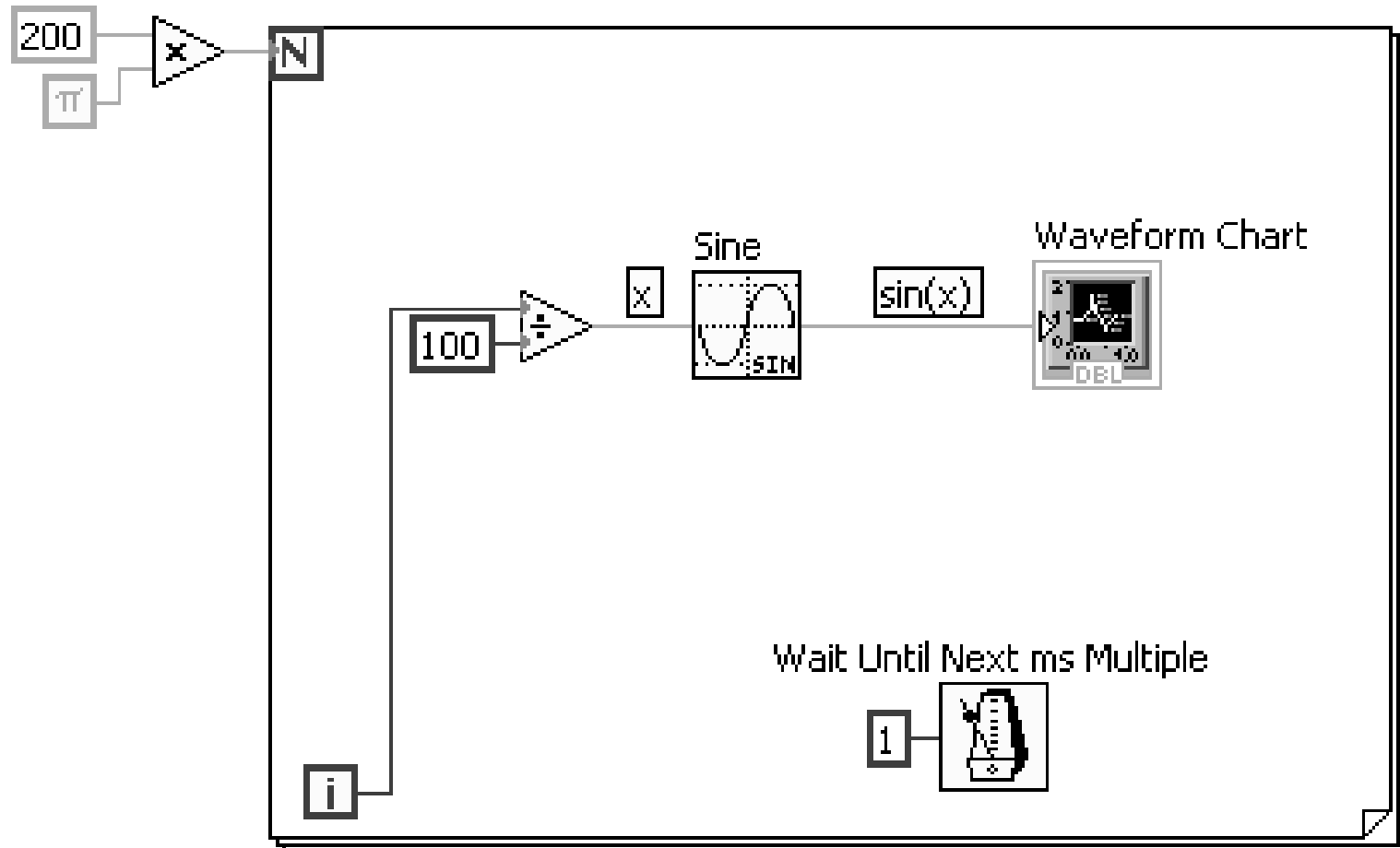
- Waveform Graph
- XY Graph
- Intensity Graph
- Digital Waveform Graph
- Mixed Signal Graph
- 3D Surface Graph
- 3D Parametric Graph
- 3D Curve Graph

Wykresy

- **chart** – pojedyncze punkty ciągle uaktualniane - *Waveform Chart*
- **graph** – otrzymuje całość danych (jako tablicę) zanim je wyświetli
- LabVIEW umożliwia
 - wykresy graph i chart - 1D (stały odstęp pomiędzy x-ami)
 - wykres graph - 2D (XY Graph)
 - wykres graph - 3D

Wykresy

Waveform Chart – rysowanie punkt po punkcie



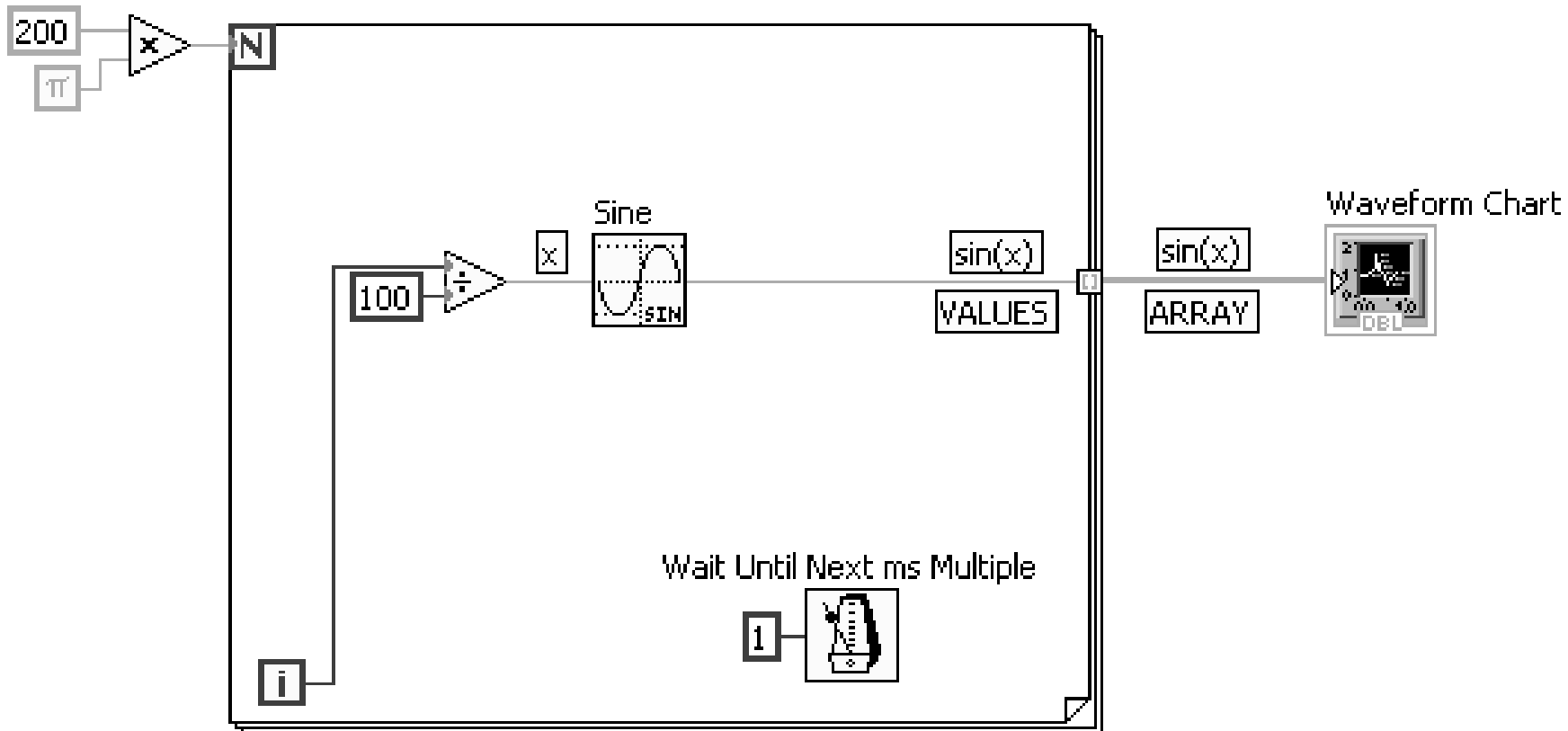
Wykresy

Waveform Chart – rysowanie punkt po punkcie

- **chart history length**
 - **update mode**
 - **Strip Chart (default)**
 - **Scope Chart**
 - **Sweep Chart**
-

Wykresy

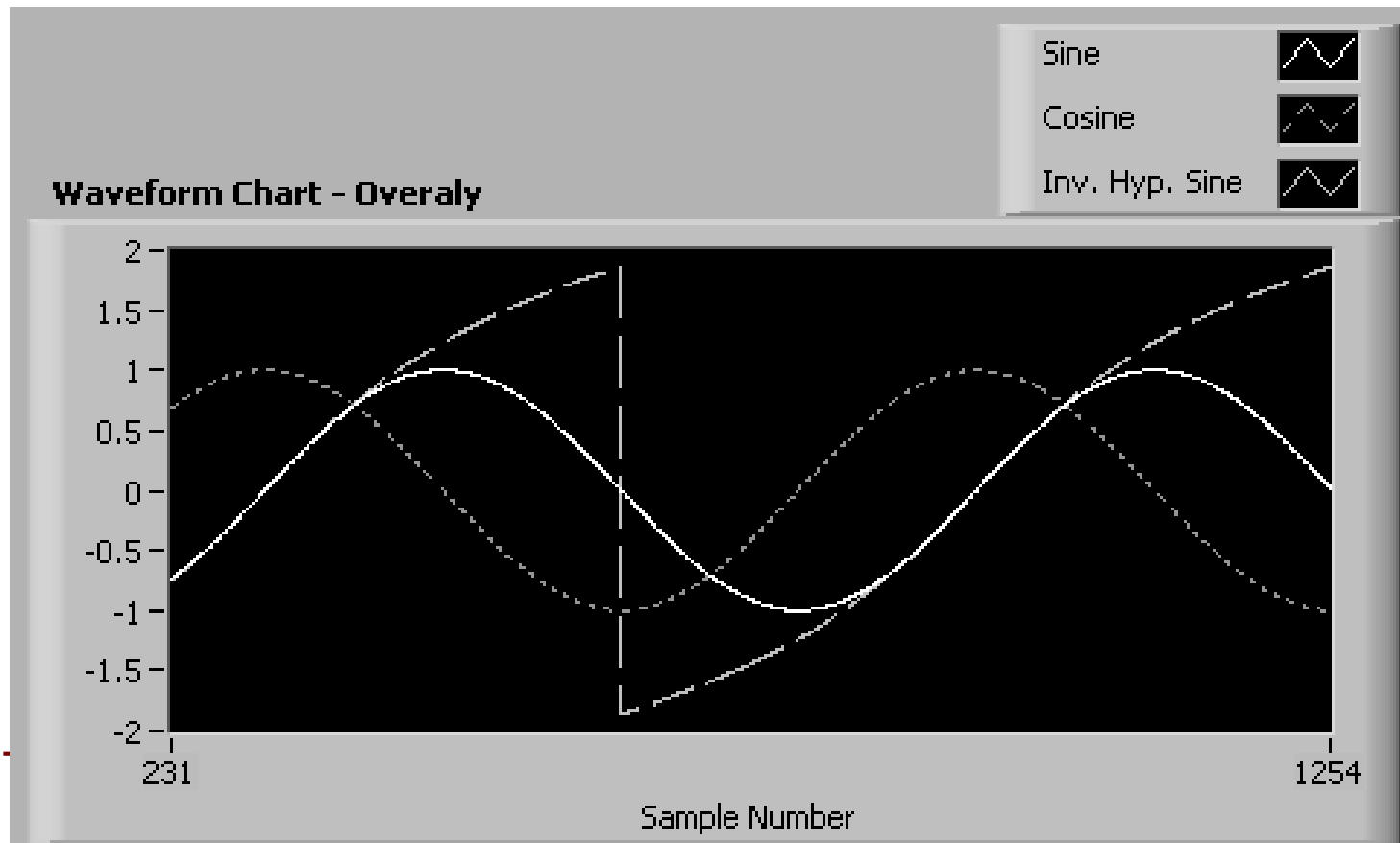
Waveform Chart – rysowanie całości



Wykresy

Waveform Chart – kilka krzywych

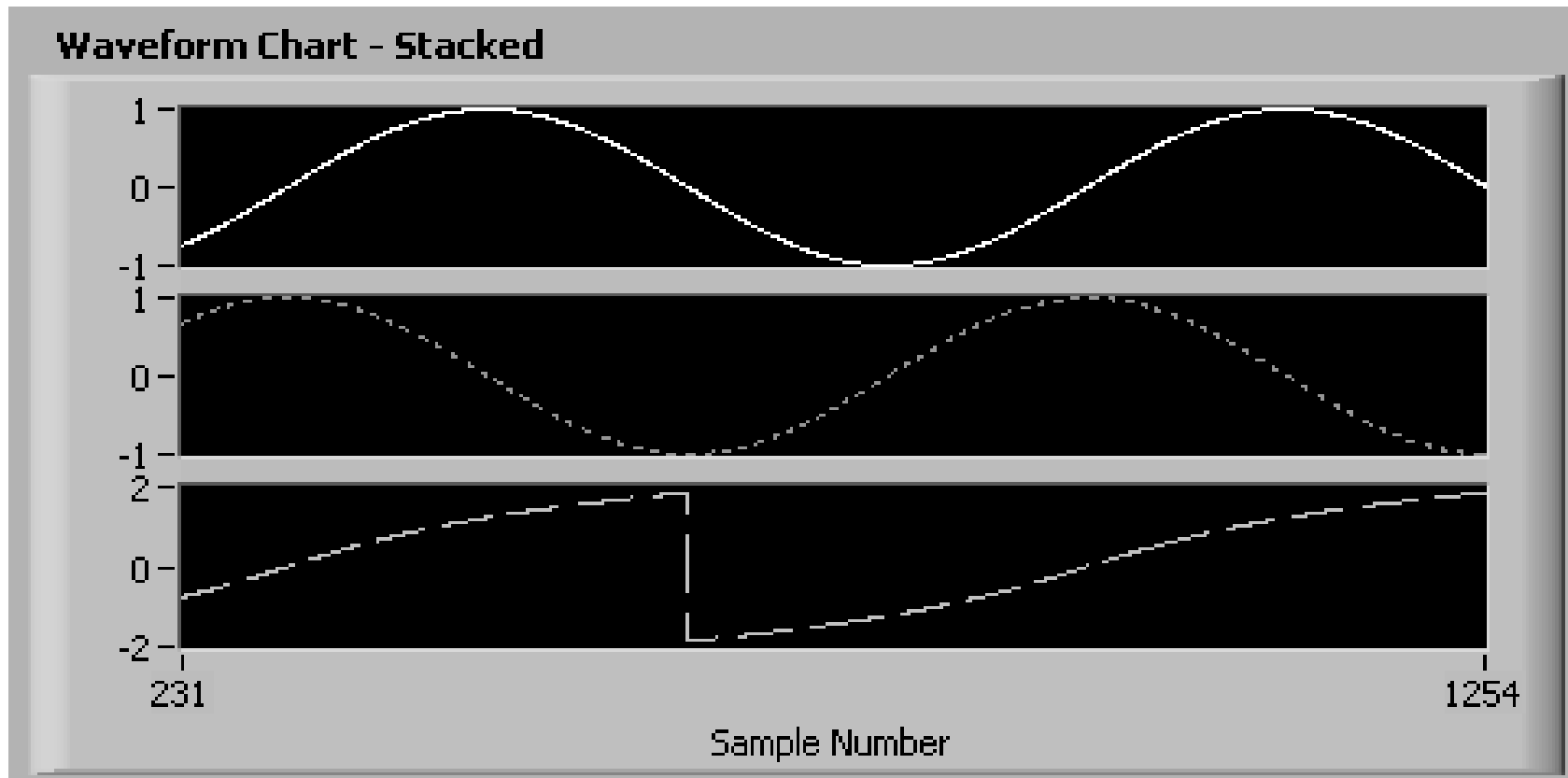
- **Overlay Plots** – wszystkie krzywe na tym samym wykresie



Wykresy

Waveform Chart – kilka krzywych

■ **Stacked Plots – krzywe osobno**



Wykresy

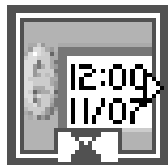
Waveform Graphs

- tablica (1D) do wyświetlenia
 - wszystkie wartości wyświetlane od razu
 - wiele wykresów – trzeba zbudować i przekazać odpowiednią tablicę
-

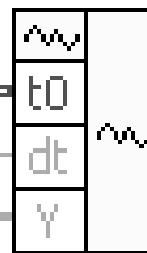
Wykresy

Waveform Graphs i Charts - opcje

Start Time, t_o



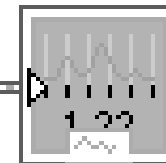
Build Waveform function



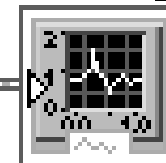
Time Interval, Δt



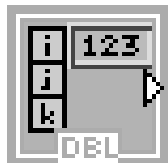
Waveform (bundled)



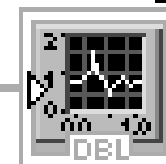
Plotting the Waveform



Data Array

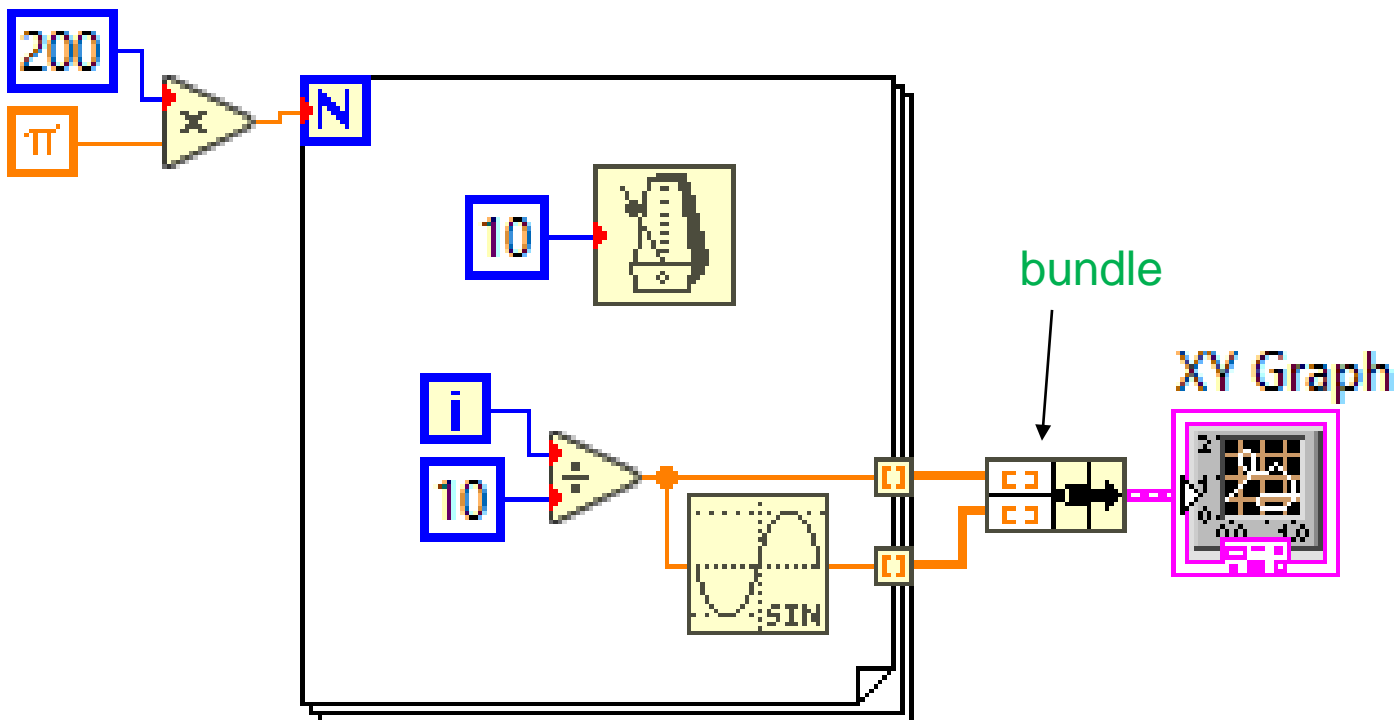


Plotting the Data Array



Wykresy

XY Graphs – wykresy 2D



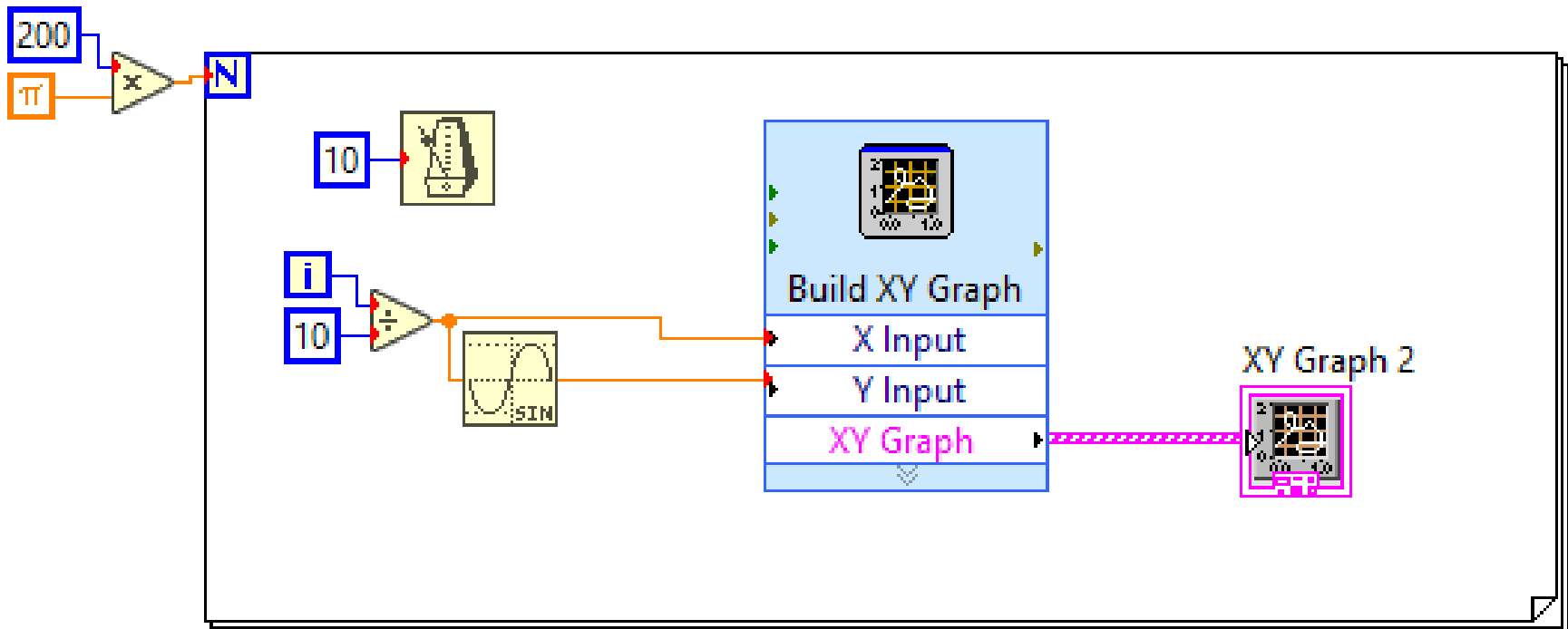
zwykle XY Graph używany jest do konstrukcji całego wykresu – jak wyżej

Wykresy

XY Graphs – wykresy 2D

Można też rysować punkt po punkcie:

- korzystając z XY Graph'u z funkcją Build XY Graph (wstawia się automatycznie gdy wybierzemy XY Graph z zakładki Express FP

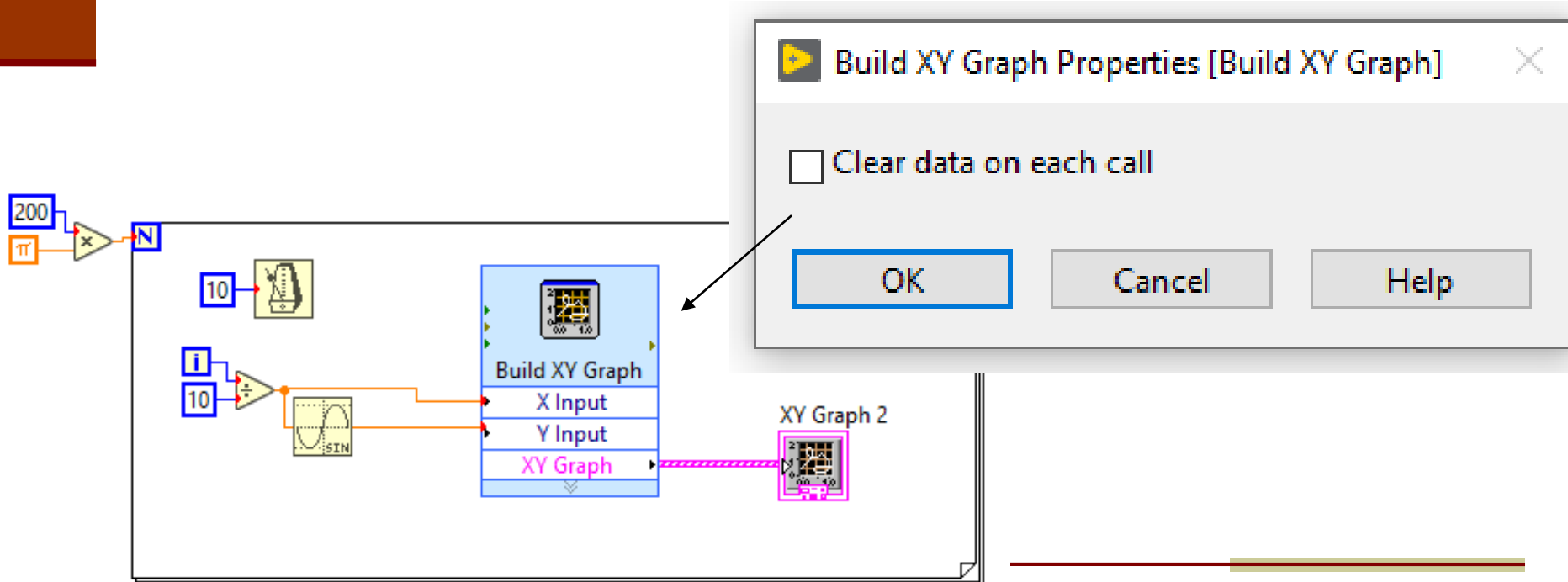


Wykresy

XY Graphs – wykresy 2D

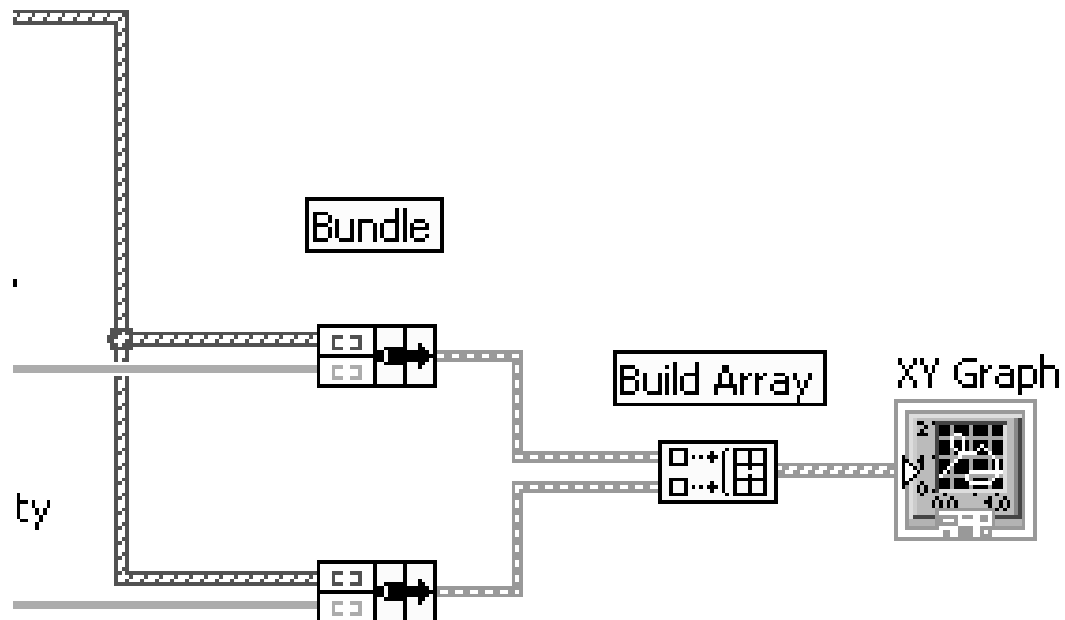
Można też rysować punkt po punkcie:

- trzeba jeszcze w Properties dla „Build XY Graph” odznaczyć „Clear data on each call”



Wykresy

XY Graphs – więcej wykresów



Ćwiczenie 1

- odtworzyć program ze strony 28 wykładu,
 - sprawdzić działanie „update mode” (w Properties dla Waveform Chart):
 - Strip Chart (default)
 - Scope Chart
 - Sweep Chart
 - zmodyfikować program tak jak na str. 30
 - zaobserwować różnice w działaniu
-

Zadanie 1

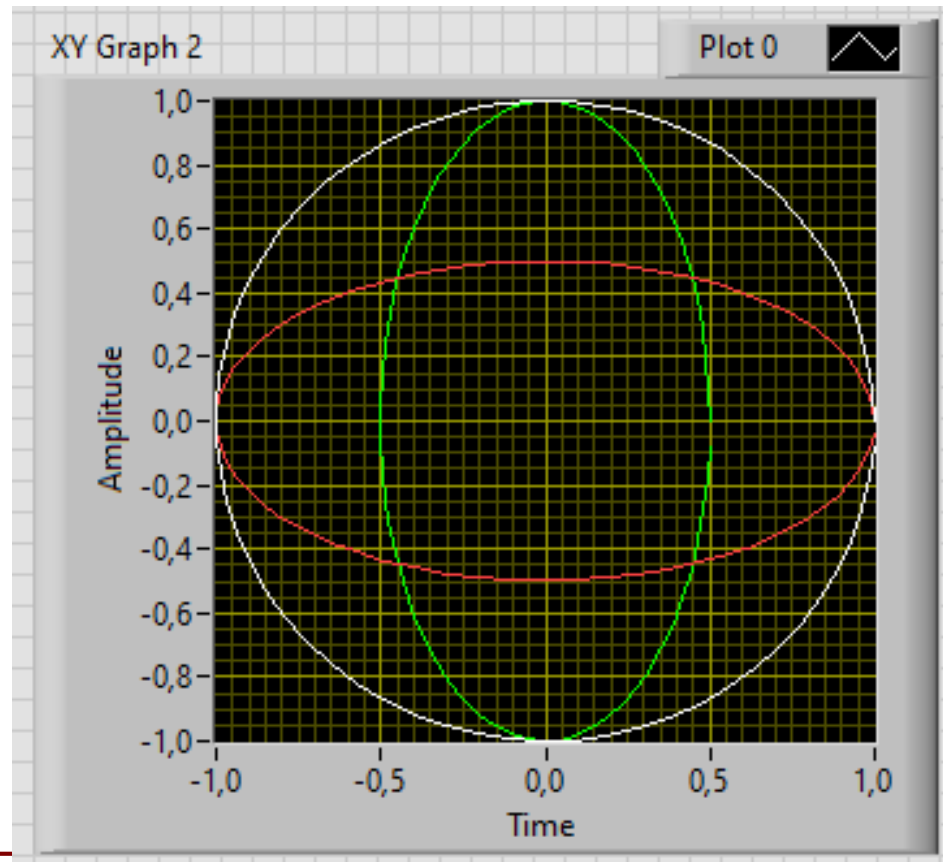
- zmodyfikować program ze strony 28 wykładu tak aby można było narysować trzy różne przebiegi równocześnie na dwóch wykresach Waveform Chart – jeden w trybie Overlay Plots, drugi Stacked Plots
-

Ćwiczenie 2

- odtworzyć program ze strony 35 wykładu,
 - sprawdzić działanie XY Graph
 - zmodyfikować program do postaci ze strony 36/37
-

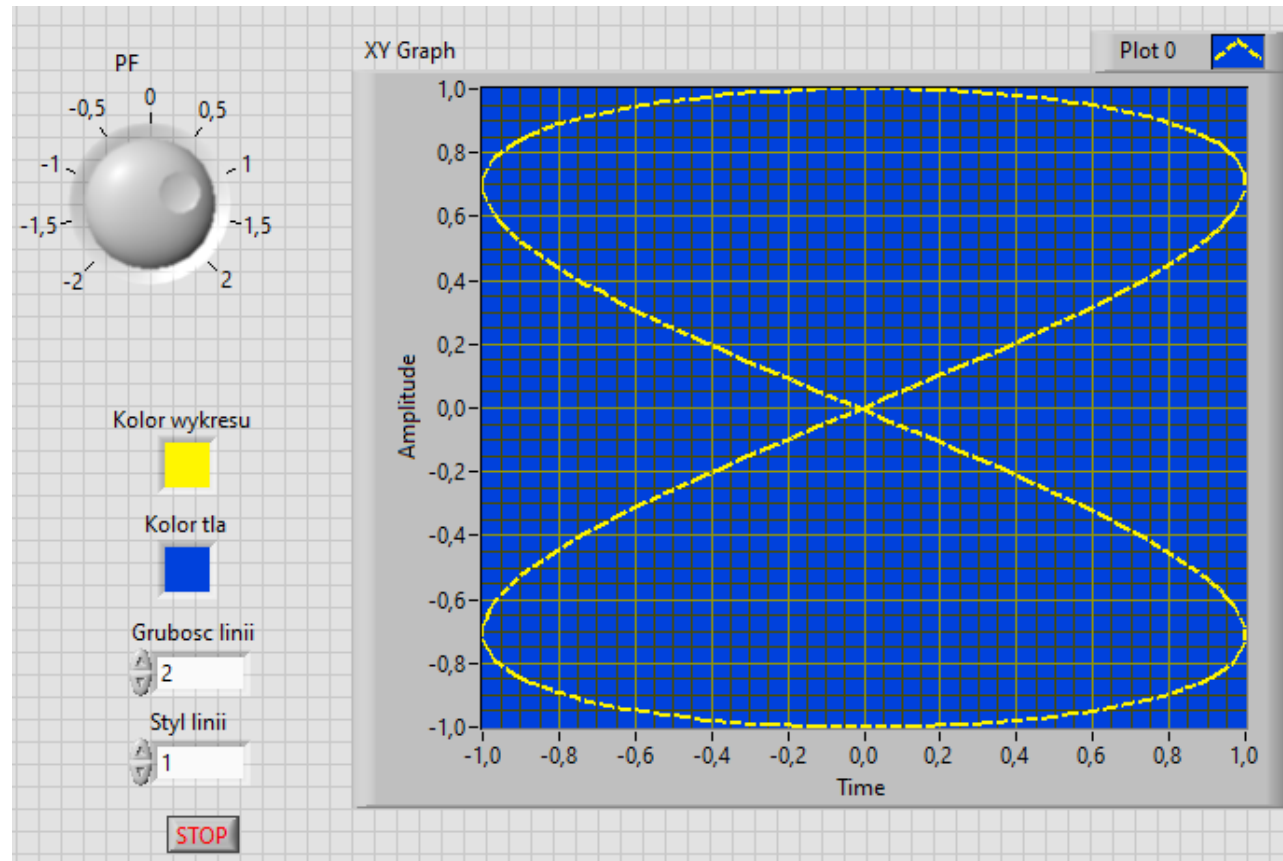
Zadanie 2

- zmodyfikować program ze str. 35, tak aby otrzymać analogiczny wykres (lub wariację na temat)



Zadanie 3 (trudniejsze)

- napisać program do rysowania krzywych Lissajous z możliwością kontroli kolorów i parametrów linii



Zadanie 3 (cd)

podpowieź – skorzystaj z „Property Nodes” dla XY Graph –
np. klikając prawym przyciskiem na XY Graph i wybierając
Create/Property Node/Plot/Plot Color

