Opole University of Technology

Department of Mechanics and Machine Design

LabVIEW

Laboratory manual for students

LabVIEW software in electrical circuit analysis

Opracował: Dr inż. Roland Pawliczek

Opole 2019

1. The purpose of the exercise

The aim of the exercise is to familiarize with the LabVIEW user interface and its use as a tool for programming simple engineering calculations on the example of an electrical circuit analysis.

2. The course of the exercise

The LabVIEW environment belongs to a group of systems in which the so-called graphic programming. The source code of the program are not text lines containing commands but graphical icons representing variables, functions and procedures that are described by a number of properties. The flow of information and the course of the program is created by means of links marked in the code by means of a line (Wires).
The LabVIEW environment is widely used in laboratory practice to build virtual measuring and control devices. Therefore, objects created in LabVIEW are often called virtual instruments. For this reason, the default extension for LabVIEW files is vi.
After starting LabVIEW, a window like in Fig. 1 will appear.

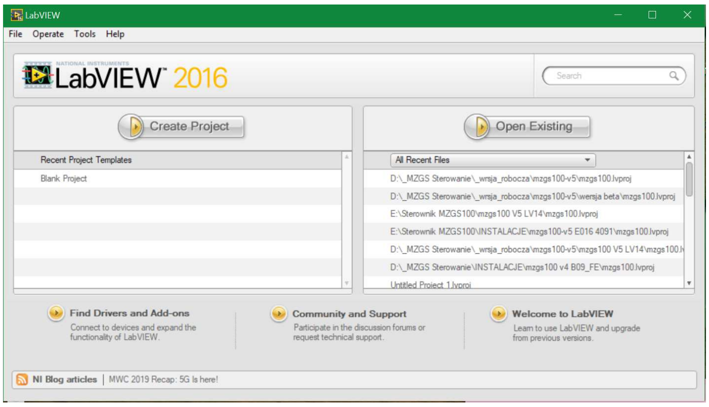


Fig. 1. Starting screen LabVIEW

From the *Open Existing* find the file named **cw1_start.vi**. The so-called Front Panel, or user panel, on which individual elements of virtual instruments will be placed. Figure 2 shows the user panel with loaded drawing elements showing an example of an electric circuit.

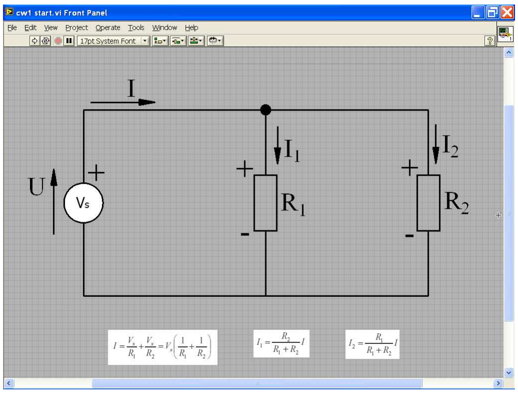
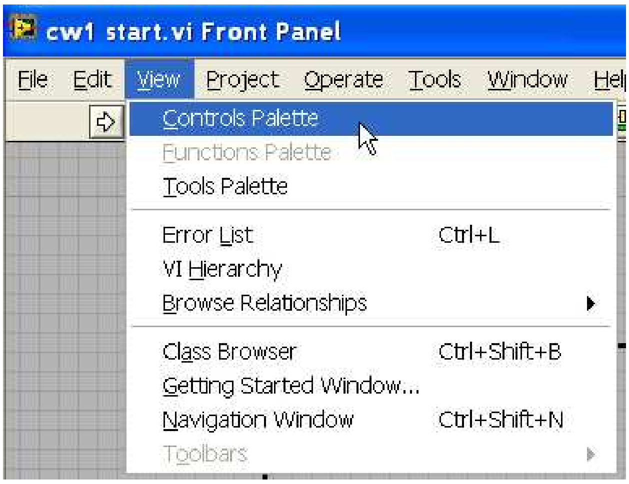


Fig. 2. Front Panel

The meaning of individual menu items and toolbars will be discussed as needed. In order to gain access to a set of elements to be used in the program, select the ***View / Control Palette*** option from the main program menu. A window will appear containing the Controls elements (Figure 3). Icons allow access to specific types of elements.



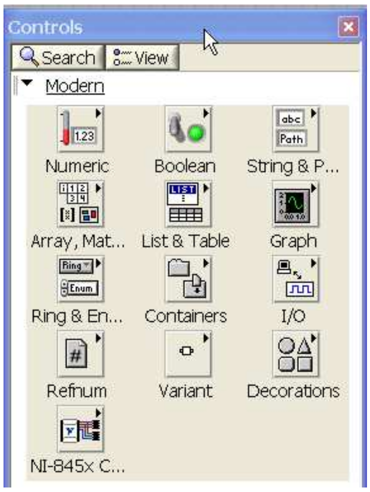


Fig. 3. Controls Palette

After selecting the *Numeric* icon, a set of tools will appear, which can be divided into two main groups (Figure 4):
- **Controls** - these are input elements ("controls"), which allow to set variable values (data input). The controls have the ability to edit and usually they are equipped with elements allowing to change the value of entered data (arrows next to the "Numeric" field in Fig. 4).
- **Indicators** - these are output elements ("displays") that allow displaying results of calculations, charts, etc. Data displayed in these elements are not editable - the user can`t change their value ("Numeric2" field in Fig. 4).

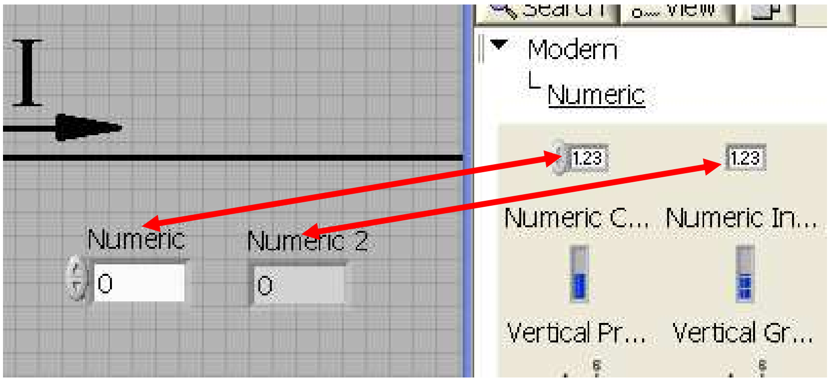


Fig. 4. Controls and Indicators

.

Using these elements, a program will be built that allows calculating the currents in individual branches of the electric circuit depending on the value of the source supply voltage and resistance in the branches of the circuit. For simplicity, the user panel contains the formulas necessary to perform calculations.

The element is transferred to the user's panel using the "drag and drop" technique using a computer mouse.

Using the controls and displays, build a panel as in Figure 5. The name of the element can be entered just after it is placed on the panel or later by double-clicking the left mouse button on the text describing the element.

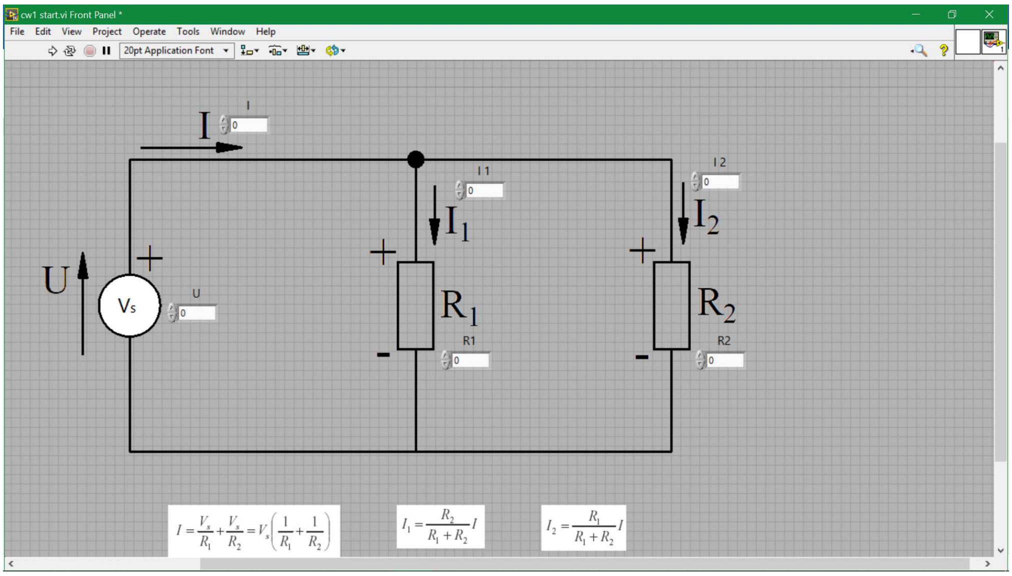


Fig. 5. User interface

For the purpose of the exercise, it was assumed that the source voltage will be within the limits of 0 - 24V and it will be possible to change them with a 2V step. In order to gain access to the control's properties, you should point it and right-click to open the so-called Context menu (Popup Menu, Fig.6). After selecting the *Properties* option, select the *Data Entry* tab and set the values *Minimum = 0, Maximum = 24* and *Increment = 2* (Figure 6). The same effect can be obtained by selecting the *Data Entry* option directly from the context menu.

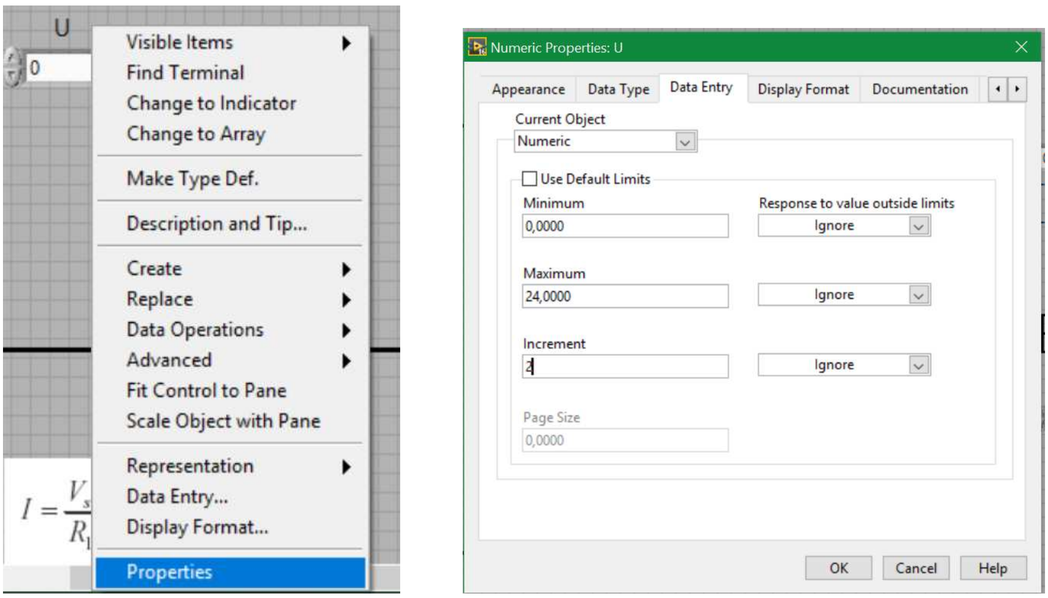


Fig. 6. Control Properties

At this point, you can check the operation of the indicator by clicking the **up / down arrows** when a small **handle with finger** (Figure 7).

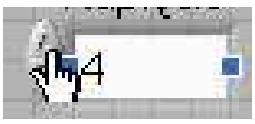


Fig. 7. Input data change

In a similar way, the properties of the R1 and R2 resistance control lamps should be changed. In this case, the Data Entry and Format & Precision options were used (Figure 8). Set *Default value = 10,000*, *Minimum = 10,000, Maximum = 500,000, Increment = 10,000*. In the *Display Format* tab, set SI notation and *Digits = 0* and *Prescision Type = Digits of precision*.

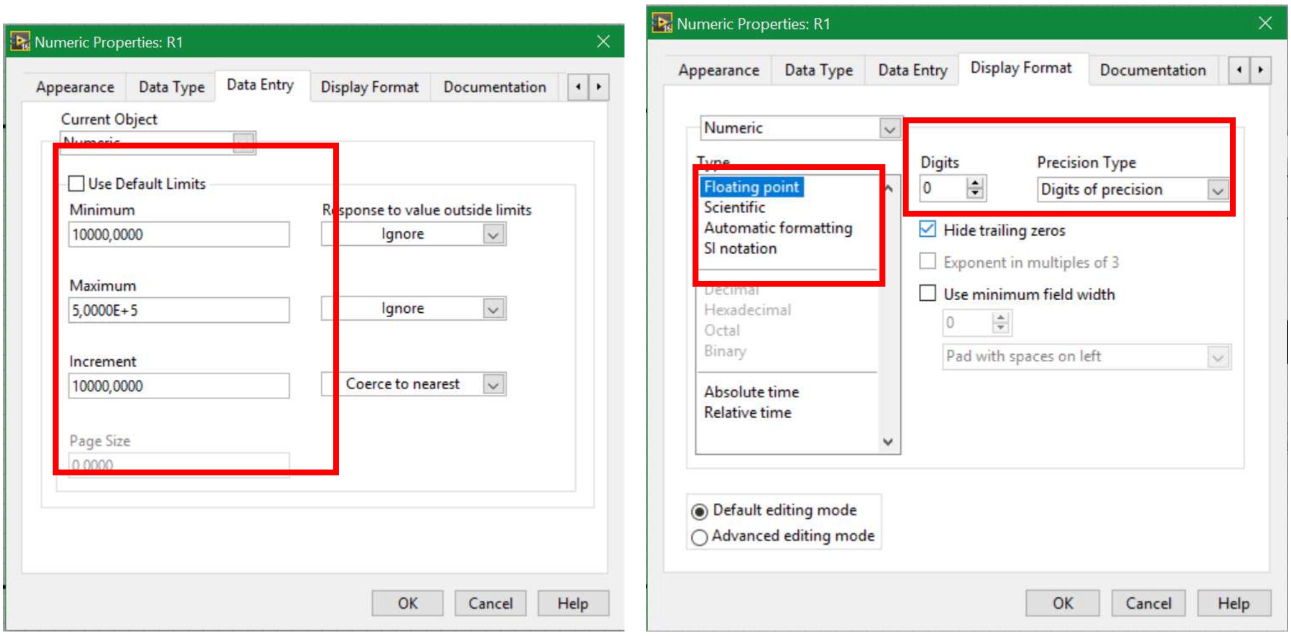


Fig. 8. Control properties for R1 and R2

Since relatively small current values should be expected, the numbers should be changed in the displays defining the currents in individual branches. Set the *Floating point and Digits = 6* and *Prescision Type = Digits of precision* settings in the *Display Format* tab (Figure 9).

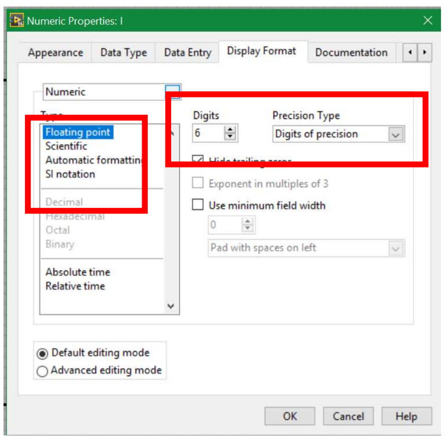


Fig. 9. Change of the properties for current I, I1, I2 indicators

The user panel was thus equipped with the necessary elements for which their properties were predicted and defined. This form of the program can\`t be started yet. No links between controls and displays have been identified so far. The construction of the graphic form of the program (source code) takes place in the so-called *Block Diagram* started using the *Show Block Diagram* option from the main program menu (Figure 10).

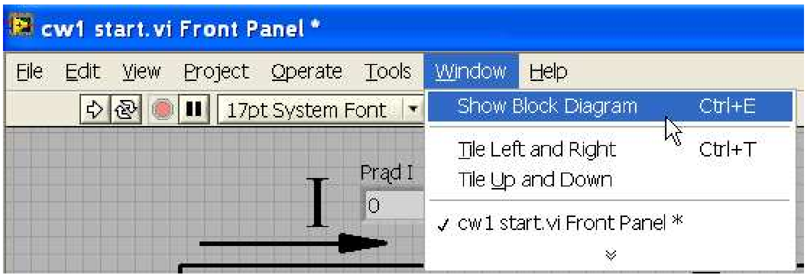


Fig. 10. Switch to Show Block Diagram window

A window will appear on the screen as in Fig. 11 containing graphic symbols representing previously inserted controls and displays. You can clearly distinguish between **controls that have a small arrow on the right (symbolizing the output) and displays that have a small arrow on the left (symbolizing the input)**.

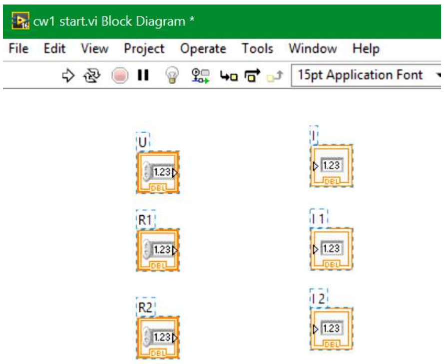


Fig. 11. Control Diagram window

The programming of activities and connections takes place using the functions available after choosing the *View / Functions Palette* option from the main window of the *Block Diagram*. An additional window will appear on the screen containing *a list of available functions and procedures* (Figure 12). Basic mathematical operations are available in *Numeric* window Fuctions (Figure 12).

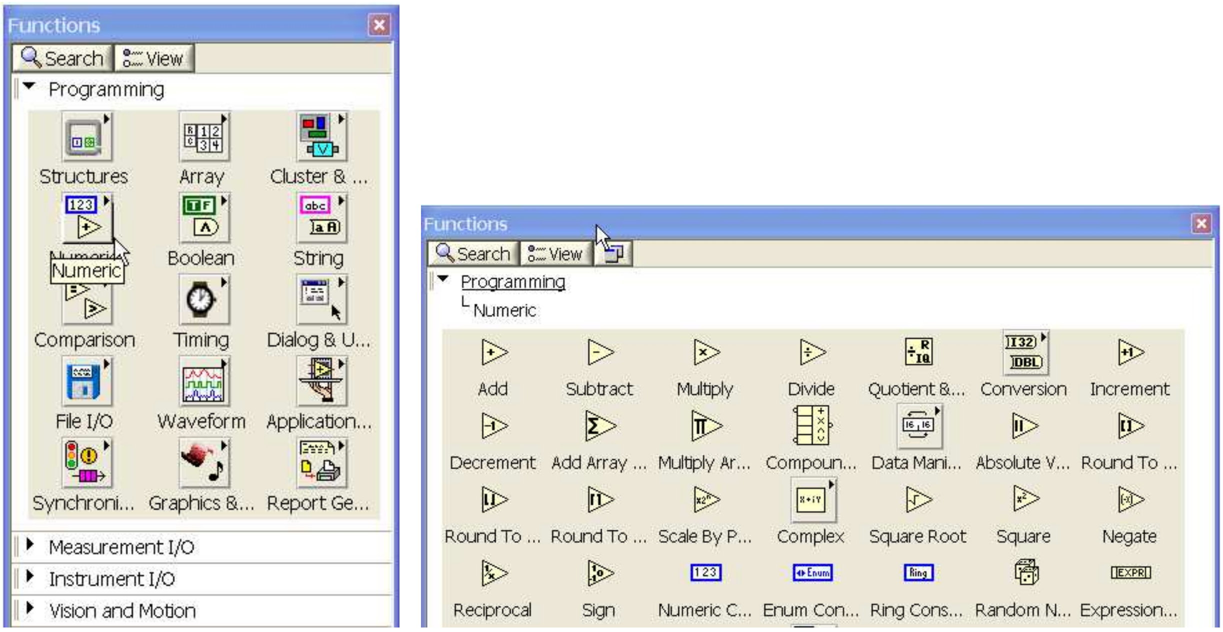
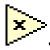


Fig. 12. Function Palette and basi Numeric functions

Using the available functions, the user must build graphical structures corresponding to particular equations:

$$I = \frac{V_s}{R_1} + \frac{V_s}{R_2} = V_s\left(\frac{1}{R_1} + \frac{1}{R_2}\right) \qquad I_1 = \frac{R_2}{R_1 + R_2} I \qquad I_2 = \frac{R_1}{R_1 + R_2} I$$

Simple math operations will suffice for the task.
In order to determine the current I, invert the resistances, sum them and then multiply them by the source voltage. To do this, select and insert elements into the diagram from the Functions menu and the Numeric option:

- reciprocal, 
- add (add), 
- multiply (multiply) .

Since the inversion operation should be done twice, two elements will be needed. You can use the standard Windows *Copy / Paste* operations to do this.
A very useful option of the LabVIEW system is help with information about the selected element (**Context Help**). To start it, select the element and press **Ctrl + H**. Then a window will appear as in Figure 13. Of course, the content of the window will depend on the selected element.

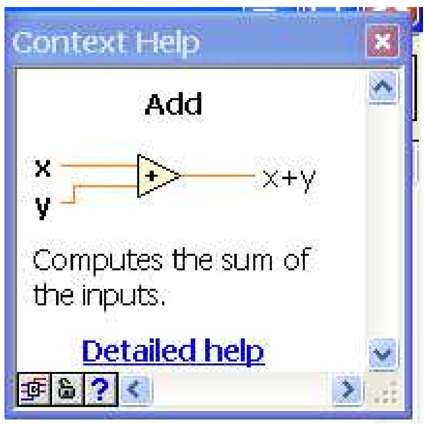


Fig. 13. Context Help window for Add function

The icon of the element and its input and output are shown. The number of inputs and outputs depends on the selected element. There is also a short description of the results of the function performed by the indicated element. If we move the mouse cursor to one of the inputs or outputs, it will change to a "thread spool" (Figure 14).

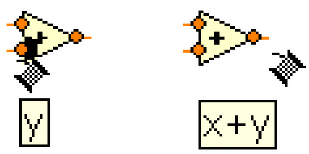


Fig. 14. Input and output terminals of the function

At this point it is possible to make a connection using "wire" (Wires). Figure 15 shows the combined resistance controls to perform the inversion operation.

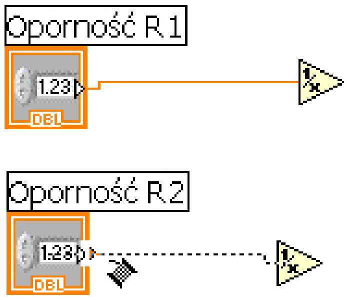


Fig. 15. Creating wires

Za pomocą połączeń należy zbudować schemat jak na rysunku 16.

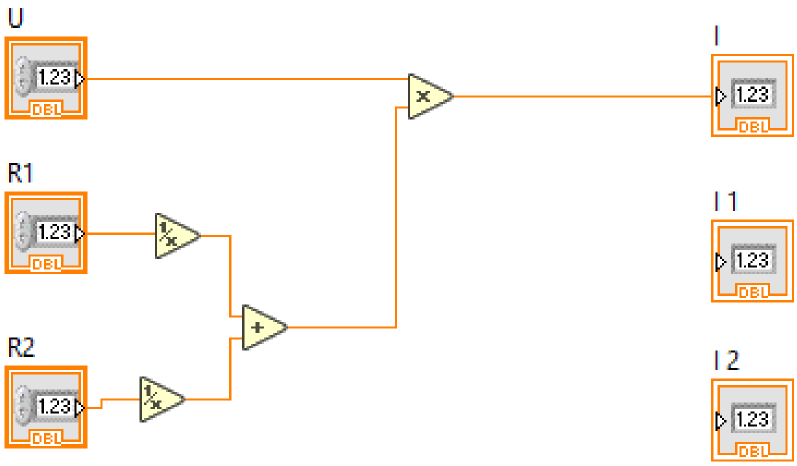
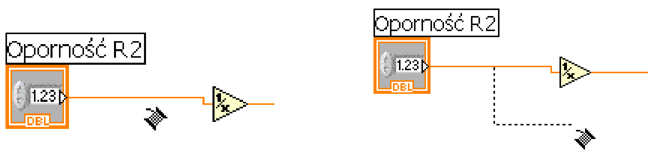


Fig. 16. Block Diagram for current I calculation

In the case of relations describing the currents I1 and I2, it is necessary to re-use the resistances R1 and R2. There is no need to redefine data - it is possible to create nodes and transfer data to other junction connections. To do this, move the mouse cursor to the selected connection and a new junction will appear (Figure 17).



Rys. 17. Creating junction

Using ,  i  complete diagram according to Figure 18.

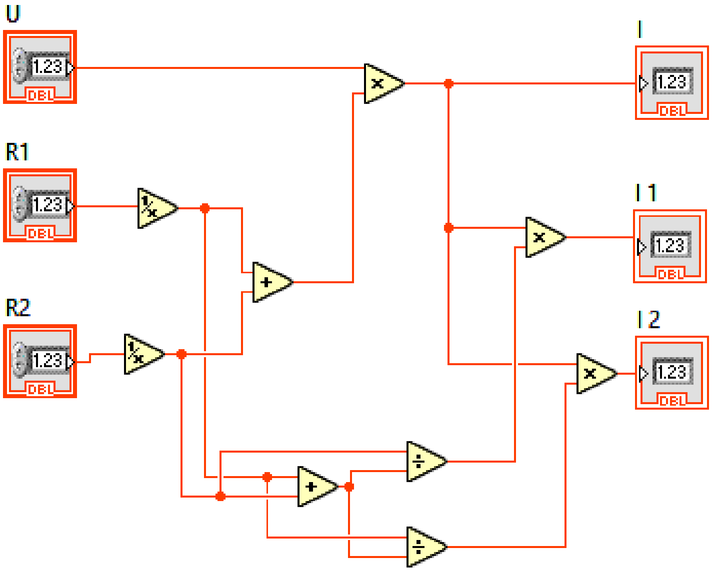


Fig. 18. Block diagram for currents calculation

The diagram covers all necessary operations. You can start calculations. To do this, switch to the user panel (option *Windows / Show Front Panel* from the main menu) and enter the data: **Voltage = 8V, Resistance R1 = 140k, Resistance R2 = 210k.**
To run the program, use the *Operate / Run* option from the main menu or the *Run button*, which is located on the toolbar located under the main program menu (Figure 19).

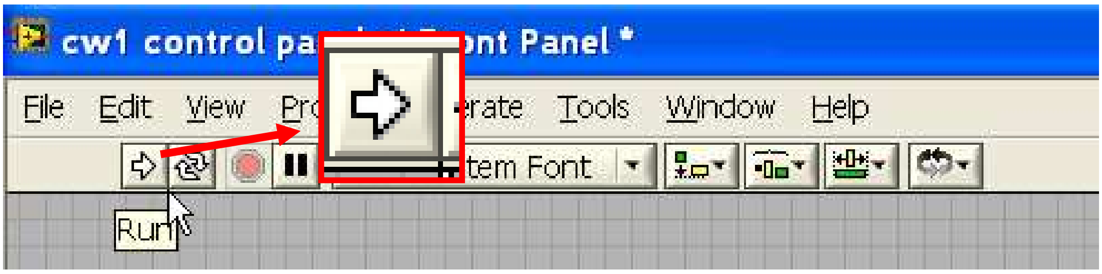


Fig. 19. Run the program

As a result, current values in the displays should appear. The program is run once.
You can change the values of the input data again and perform the calculation.
From the user's point of view, it is more convenient to run the program continuously, which will react to changing the input data. There are two ways to solve this problem. The first is to run in a loop using the *Run Continuously button*, which is next to the *Run button*. From now on, you can enter new data on a regular basis and the program will automatically perform calculations. The calculation is stopped using the *Abort Execution* key (red point on the toolbar) (Figure 20).

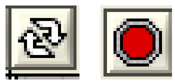


Fig. 20. Run Continuously and Stop buttons

However, this is an inelegant way and usually used when testing programs.
For continuous operation of the program, use the **While Loop**. It causes the program to be executed until the **End / Stop** key is pressed. To complete the diagram, switch to the **Block Diagram** window (Ctrl + E). Then use the *Show Functions Palette* option to display a list of functions (Fig.12). Select the **Structures** option and indicate the **While Loop** (Figure 21).

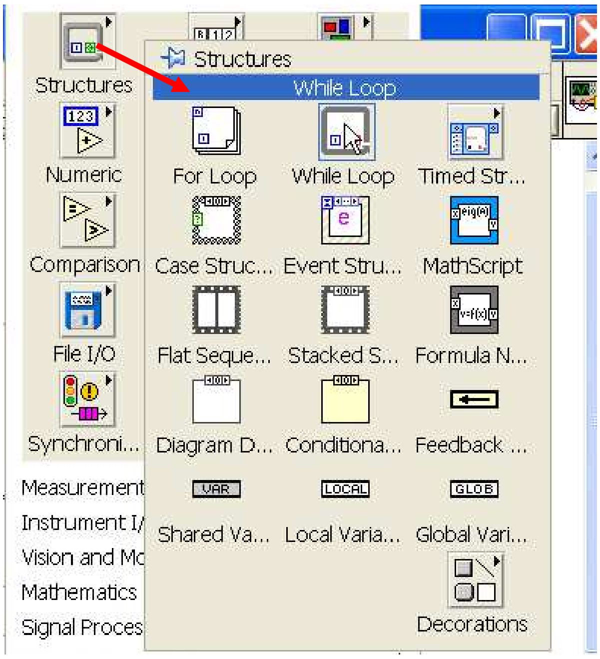


Fig. 21. Structures function palette

Then, on the diagram, surround the previous diagram with a **rectangle** symbolizing the **While Loop**. All elements that will be inside the loop will be sequentially repeated (Figure 22).

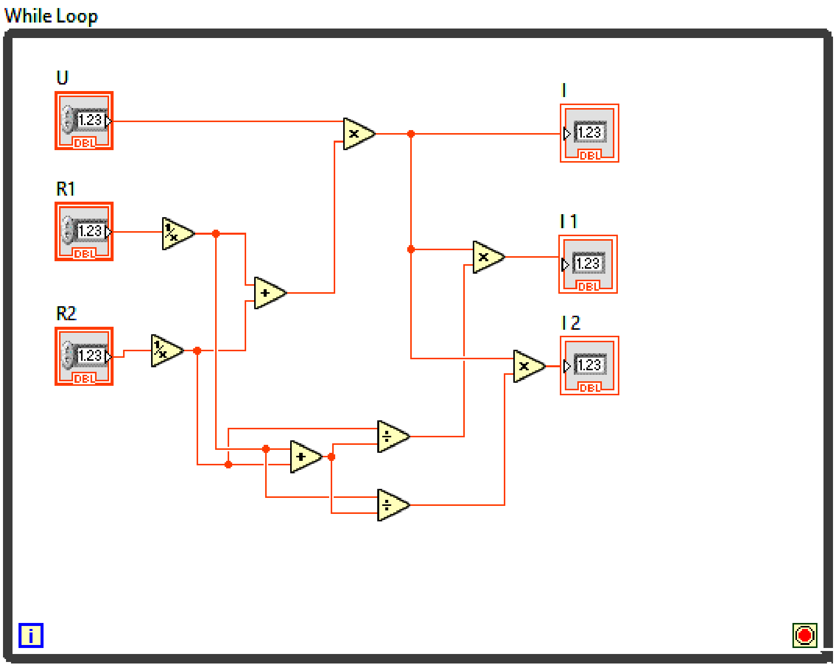


Fig. 22. Diagram with While Loop

In the **bottom right corner** of the loop is the **Loop Condition** terminal, which allows you to control the end of the loop. It is a "display" element (Indicator). To use it, define the control in the form of a button. The easiest way to do this is through the **Context Menu**, which appears when you right-click on an element (Figure 23).

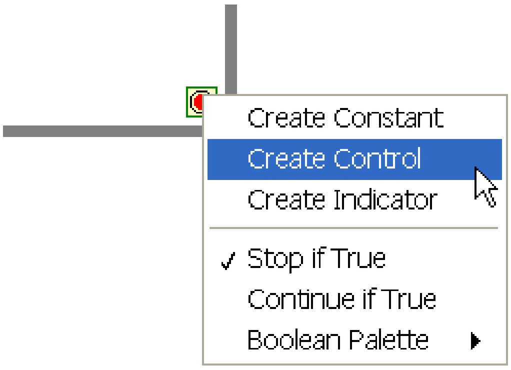


Fig. 23. Context Menu for Loop Condition terminal

After selecting the *Create Control* option, the system will automatically generate a control according to the type of display and make a connection - the corresponding key will appear in the *Front Panel* window (Figure 24).

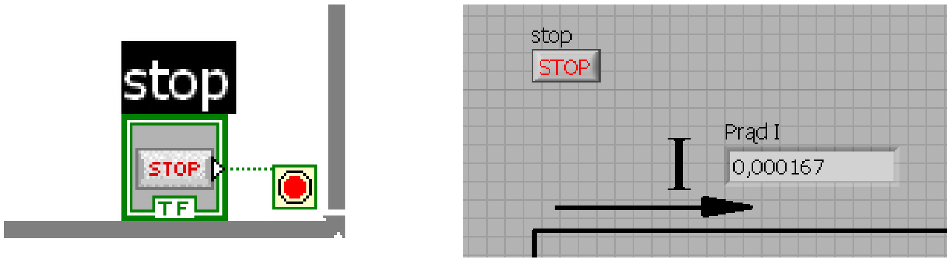


Fig. 24. Stop control on Block Diagram and Front Panel

From now on, after starting the program, it will work until you press the STOP button.

3. Panel modification

From the user's point of view, the evaluation of the current value can be troublesome with the numerical display method. More convenient are all kinds of indicators in the form of "clocks". The LabVIEW system allows you to insert these types of elements even when the program diagram is ready. The user can **Replace** elements in the Front Panel window. To do this, indicate the element (here the current value display) and select **Replace** from the **Context Menu**. The **Controls** window will appear, where the **Numeric / Meter** option should be selected (Figure 25).

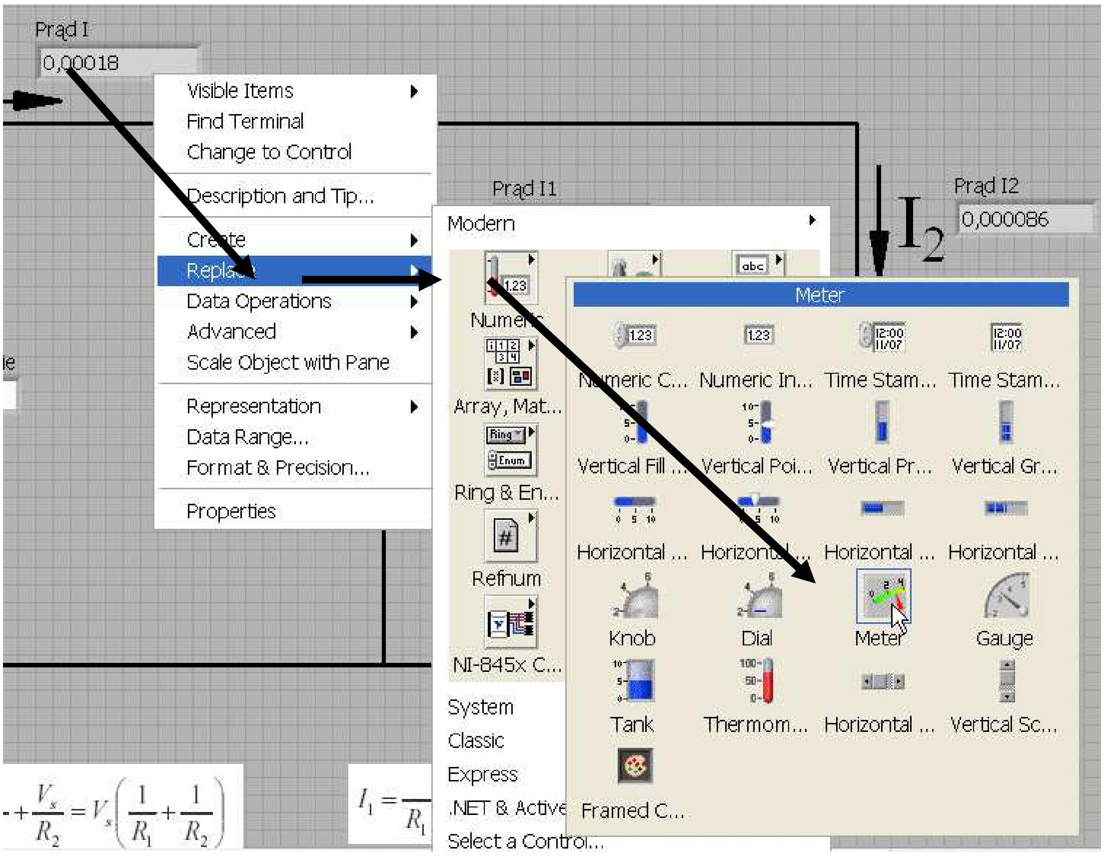


Fig. 25. Replacement of the element

The appropriate element will appear on the panel (Figure 26). Since this is a different element, check its **Properties**.

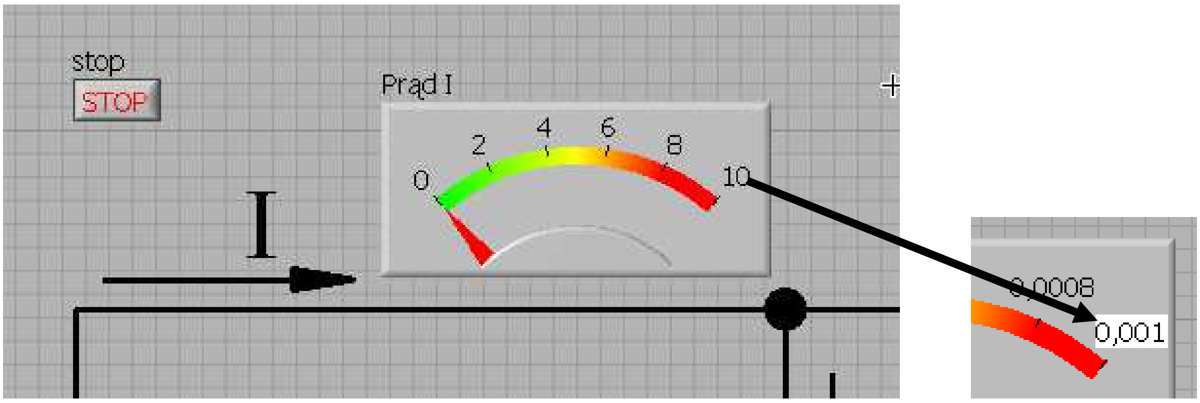


Fig. 26. Meter Indicator

In order to improve the data input, you can adjust the source voltage with the **Vertical Pointer Slide** control, and change the resistance with the **Knob** control, which are available in the **Controls / Numeric** option. (Fig. 27).

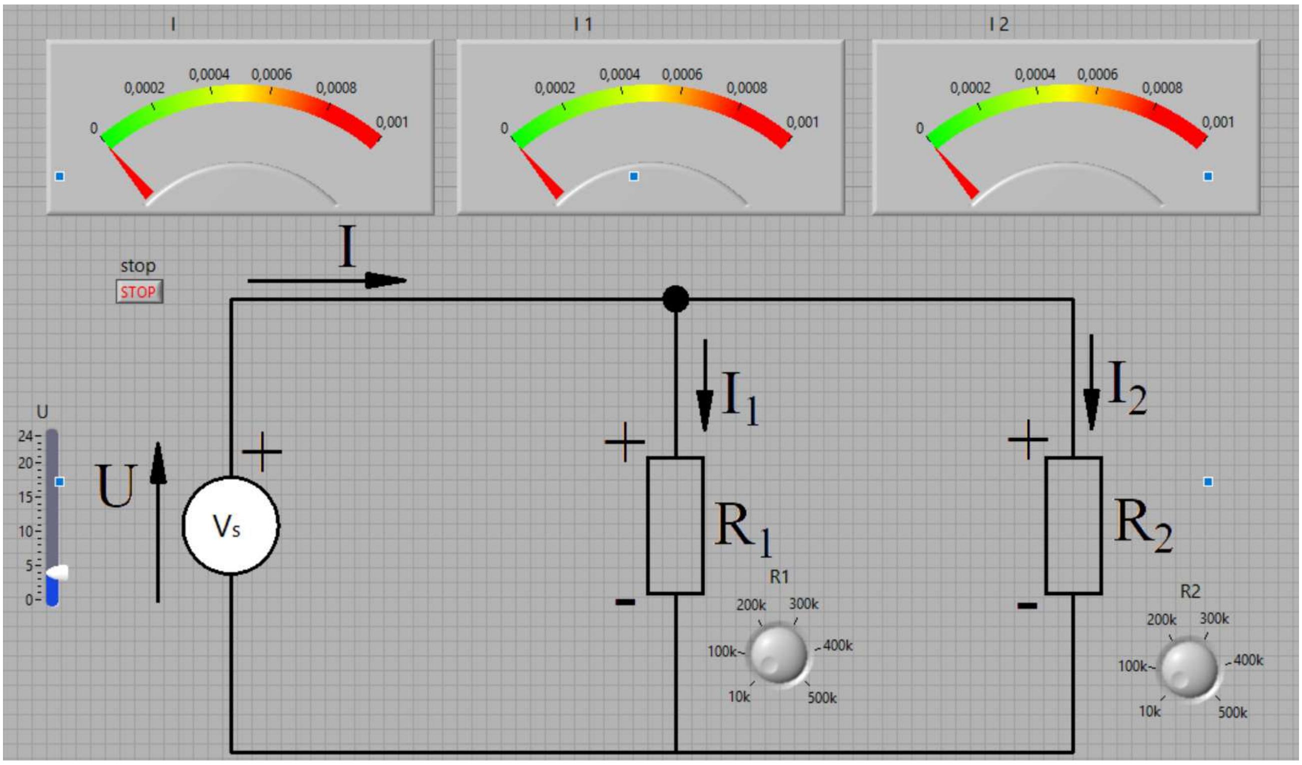


Fig. 27. The corrected Front Panel