

Zadania - LabView

LabVIEW

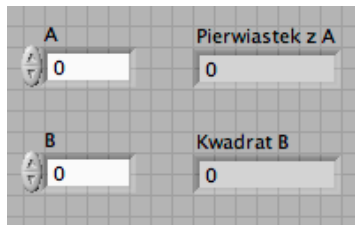
Pierwsze zadania polegają w większości na odtworzeniu programów pokazanych na rysunkach, w celu nabrania wprawy w technice pisania programów w LabView.

Programy w LabView mają rozszerzenie `.vi`. Każdy program składa się z dwóch okienek: diagramu blokowego (*block diagram*) i panelu sterującego (*front panel*).

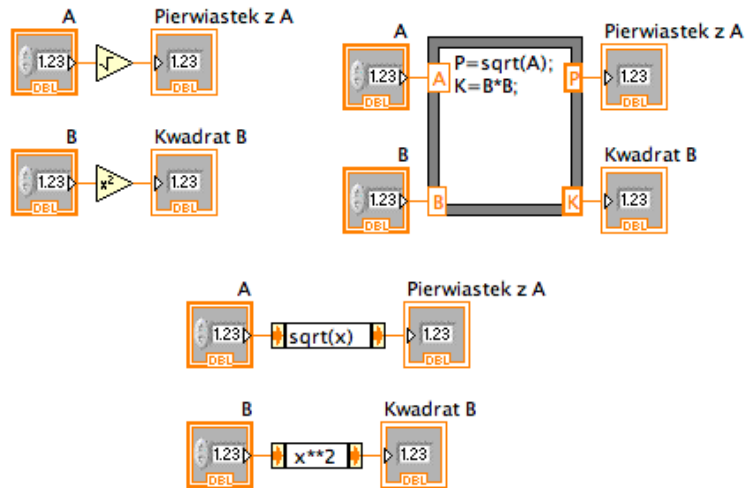
Zad. 1. Napisz program `pierwiastek_i_kwadrat.vi` (rys. 1 i 2) umożliwiający wprowadzenie dwóch liczb i obliczający pierwiastek kwadratowy pierwszej oraz kwadrat drugiej liczby. Na Rys. 2 przedstawione są trzy wersje tego samego programu. Wyjątkowo można je zapisać w jednym pliku.

W schemacie programu skorzystaj z obiektów:

- kwadratu liczby (Programming → Numeric → Square) i pierwiastka kwadratowego (Programming → Numeric → Square Root),
- obiektu umożliwiającego zapisanie wyrażeń matematycznych (Programming → Structures → Formula Node lub — dla wyrażeń jednej zmiennej — Programming → Numeric → Expression Node). W LabView 2019 lokalizacja obiektów jest inna: Mathematics → Scripts & Formulas
- pola tekstowego umożliwiającego wprowadzenie liczby (menu kontekstowe złącza → Create → Control),
- pola tekstowego umożliwiającego prezentację liczby (menu kontekstowe złącza → Create → Indicator).



Rysunek 1: Wygląd panelu sterującego programu `pierwiastek_i_kwadrat.vi`.



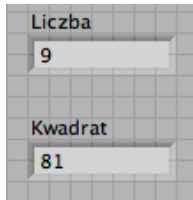
Rysunek 2: Schematy **trzech wersji** programu `pierwiastek_i_kwadrat.vi`.

Zad. 2. Napisz program `suma_kwadratow.vi` umożliwiający wprowadzenie dwóch liczb i obliczający sumę ich kwadratów. Skorzystaj z obiektu `Formuła` z palety `Express`. Następnie znajdź w palecie `Express` obiekt `Simulate Signal`. Użyj dwóch egzemplarzy takiego obiektu do generacji dwóch sygnałów sinusoidalnych o różnych parametrach, a następnie dodaj do siebie te sygnały korzystając z obiektu `Formuła`.

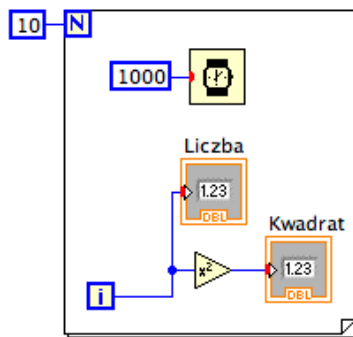
Zad. 3. Napisz program `kwadraty.vi` (rys. 3 i 4) obliczający kwadraty pierwszych 10 liczb całkowitych $0, 1, \dots, 9$. Pomiedzy kolejnymi obliczeniami wprowadź opóźnienie o długości 1 sekundy.

W schemacie programu skorzystaj z obiektów

- pętli `for` (`Programming` → `Structures` → `For Loop`),
- stałej liczbowej (menu kontekstowe złącza → `Create` → `Constant`),
- obiektu wstrzymującego pracę programu na n milisekund (`Programming` → `Timing` → `Wait`).



Rysunek 3: Wygląd panelu sterującego programu kwadraty.vi.



Rysunek 4: Schemat programu kwadraty.vi.

Zad. 4. Napisz program `pierwiastek_z_warunkiem.vi` (rys. 5 i 6) obliczający pierwiastek kwadratowy liczby. W przypadku podania liczby ujemnej program powinien wypisać komunikat `Błędna dana wejściowa!` i zakończyć działanie. Następnie przerób napisany program na funkcję (SubVi), co doprowadzi do stworzenia własnego obiektu:

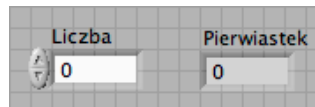


którego będzie można użyć w innych programach VI.

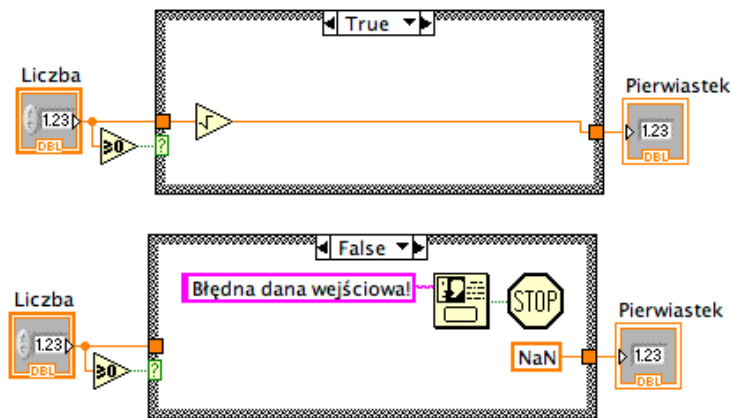
W schemacie programu skorzystaj z obiektów

- instrukcji warunkowej (Programming → Structures → Case Structure),
- warunku $x \geq 0$ (Programming → Comparison → Greater or Equal To 0),

- okienka dialogowego do wyświetlenia komunikatu (Programming → Dialog & User Interface → One Button Dialog),
- obiektu przerywającego pracę programu (Programming → Application Control → Stop).



Rysunek 5: Wygląd panelu sterującego programu pierwiastek z warunkiem.vi.



Rysunek 6: Schemat programu pierwiastek z warunkiem.vi. Uwaga ! To są dwa obrazy tego samego programu ! Zawartość ramki zmienia się w zależności od tego co wybierzemy na jej górze.

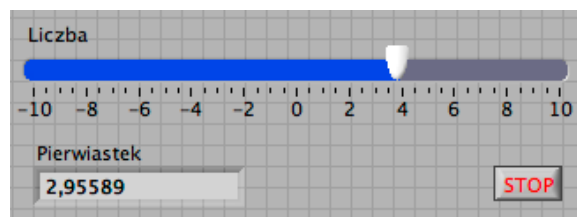
Zad. 5. Korzystając z programu pierwiastek_z_warunkiem.vi napisz program pierwiastek_w_petli (rys. 7 i 8), w którym na bieżąco obliczaj wartość wyrażenia $1 + \sqrt{x}$.

W panelu sterującym skorzystaj z obiektów

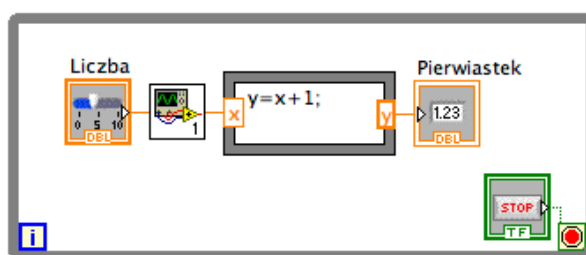
- suwaka zadającego wartość x w zakresie od -10 do 10 (Express → Numeric Controls → Horizontal Pointer Slide),
- przycisku do zatrzymania pracy programu (Express → Buttons & Switches → Stop Button).

W schemacie programu skorzystaj z obiektów

- pętli `while` (Programming → Structures → While Loop),
- ikony reprezentujące program `pierwiastek_z_warunkiem.vi`.



Rysunek 7: Wygląd panelu sterującego programu `pierwiastek_w_petli.vi`.



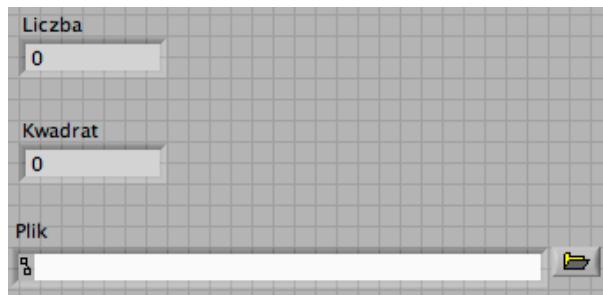
Rysunek 8: Schemat programu `pierwiastek_w_petli.vi`.

Zad. 6. Na podstawie programu `kwadraty.vi` napisz program `kwadraty_z_zapisem_do_pliku.vi` (rys. 9 i 10) zapisujący wyniki obliczeń do pliku tekstowego. W kolejnych wierszach zapisuj liczbę i jej kwadrat oddzielone spacją.

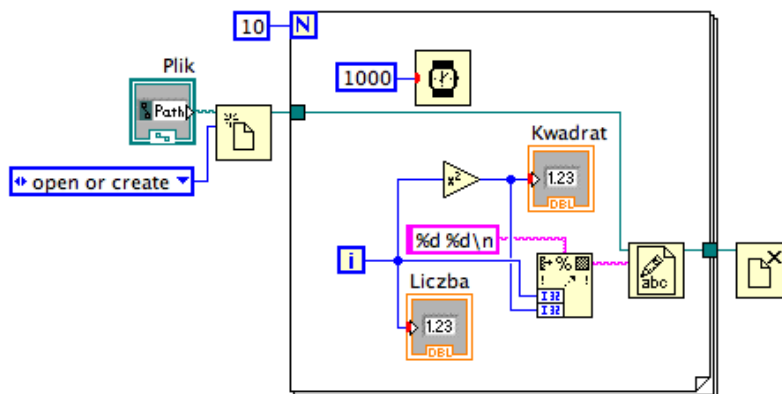
W schemacie programu skorzystaj z

- obiektu przygotowującego plik do użycia (Programming → File I/O → Open/Create/Replace File).
- Obiektu realizującego czynność zapisu do pliku (Programming → File I/O → Write To Text File).
- Obiektu kończącego użycie pliku (Programming → File I/O → Close File).

- Obiektu przygotowującego napis (Programming → String → Format Into String). Do złącza format string należy podać napis, w którym mogą występować symbole specjalne, m.in.:
 - %d — liczba całkowita,
 - %f — liczba zmiennopozycyjna (rzeczywista),
 - %s — napis,
 - %. ; — stosowanie kropki jako znaku rozdzielającego część całkowitą i ułamkową liczby,
 - \n — koniec wiersza.



Rysunek 9: Wygląd panelu sterującego programu kwadraty_z_zapisem_do_pliku.vi.



Rysunek 10: Schemat programu kwadraty_z_zapisem_do_pliku.vi.

Zad. 7. Znajdź teoretyczne wartości tłumienia

$$t(f) = \frac{1}{\sqrt{(2\pi fRC)^2 + 1}}$$

i przesunięcia fazy

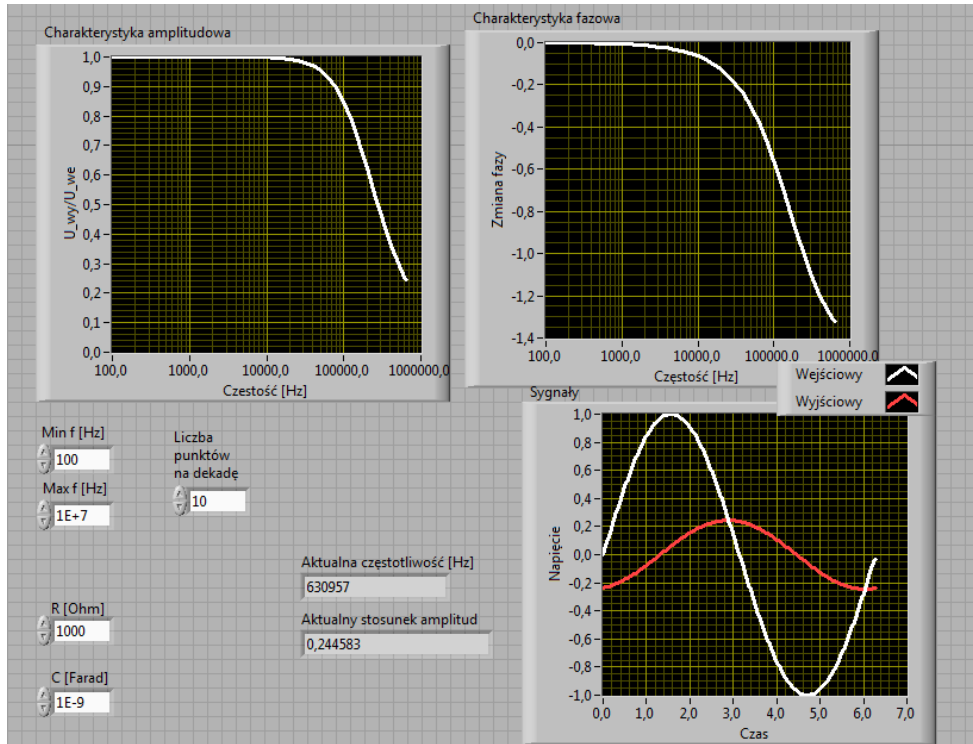
$$\varphi(f) = \arctan(-2\pi fRC)$$

sygnału wyjściowego w filtrze dolnoprzepustowym RC napisz program `teoria_RC` (rys. 11 i 12) będący komputerową symulacją pomiaru charakterystyk układu. Do tworzenia wyświetlaczy graficznych, w panelu sterującym użyj obiektów: `Express` → `GraphIndicators` → `XY Graph`. W schemacie programu, we własnościach (`Properties`) obiektu `Build XV Graph` poeksperymentuj ze zmianą ustawienia: `Clear Data on Each Call`.

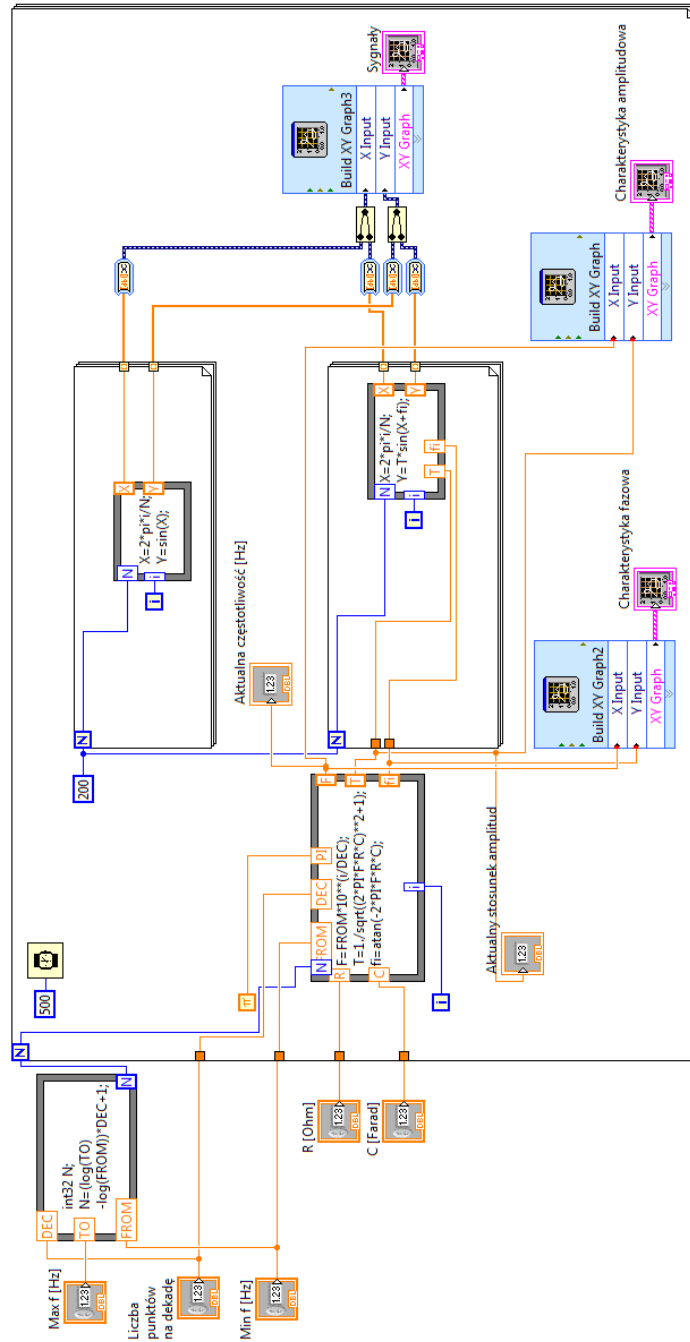
W wersji podstawowej zrób tylko rysowanie charakterystyk: amplitudowej i fazowej. Dodatkowo można dodać rysowanie sygnałów jak na rysunku 12 a w kolejnym etapie uprościć program w porównaniu do tego pokazanego na rysunkach, tworząc subvi `sinus.vi` przygotowujący dane do wykresu funkcji

$$y = T \sin(x + \phi) \text{ dla } x \in 0 \dots 2\pi$$

i używając go dwukrotnie.



Rysunek 11: Wygląd panelu sterującego programu teoria_RC.



Rysunek 12: Schemat programu teoria_RC.