

Laboratorium Automatyki i Regulacji Automatemycznej

AiRA I 01: Wprowadzenie do LabVIEW, cz. 1

1. Podstawy środowiska LabVIEW

W tej sekcji:

- Front Panel – panel przedni
- Controls Palette – paleta kontroltek
- Controls and Indicators – zadajniki i wskaźniki
- Block Diagram – diagram blokowy
- Terminals – terminale(porty)
- Controls, Indicators, and Constants – zadajniki, wskaźniki i stałe
- Block Diagram Nodes – węzły diagramu blokowego
- Functions Palette – paleta funkcji
- Searching for Controls, VIs and Functions – wyszukiwanie kontroltek, VI oraz funkcji

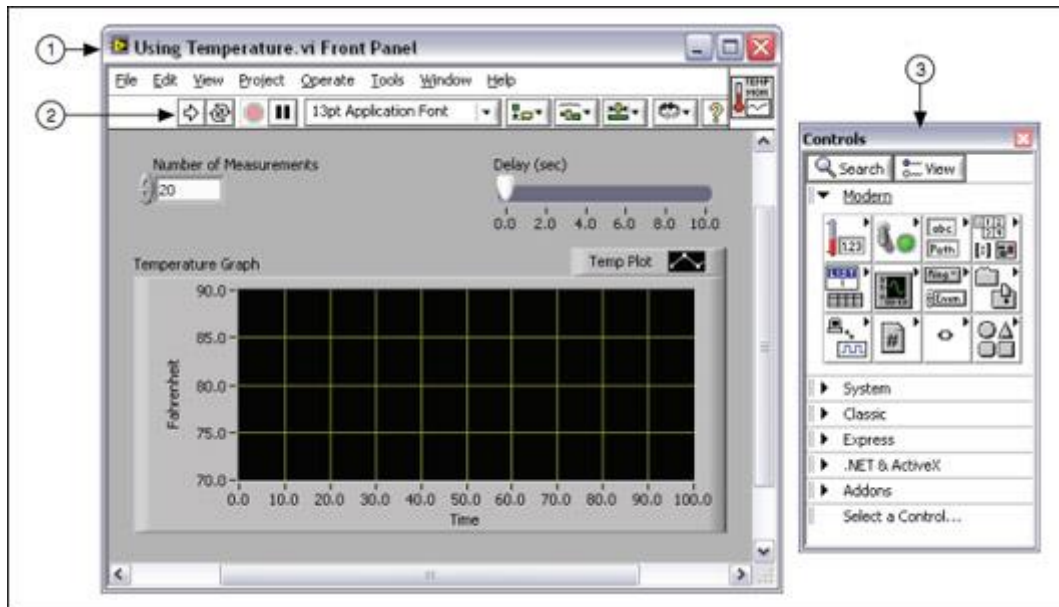
Witamy w LabVIEW. Celem tego podręcznika jest szybkie zapoznanie użytkownika z podstawowymi pojęciami i zagadnieniami programowania graficznego w LabVIEW.

Programy LabVIEW nazywane są wirtualnymi instrumentami lub po prostu VI (ang. Virtual Instrument), ponieważ ich wygląd i działanie często naśladuje przyrządy fizyczne, takie jak oscyloskopy czy multimetry. LabVIEW zawiera kompleksowy zestaw narzędzi do pozyskiwania, analizy, prezentacji i przechowywania danych, a także narzędzia, które pomagają rozwiązywać problemy z błędnym kodem.

Podczas tworzenia nowego programu (VI), widać dwa okna: okno panelu przedniego (ang. front panel) i schemat blokowy (ang. block diagram).

Front Panel – panel przedni

Po otwarciu nowego lub istniejącego VI, pojawi się okno panelu przedniego. Okno panelu przedniego zawiera projektowany interfejs użytkownika danego VI. Rysunek 1 pokazuje przykład okna na panelu przednim.

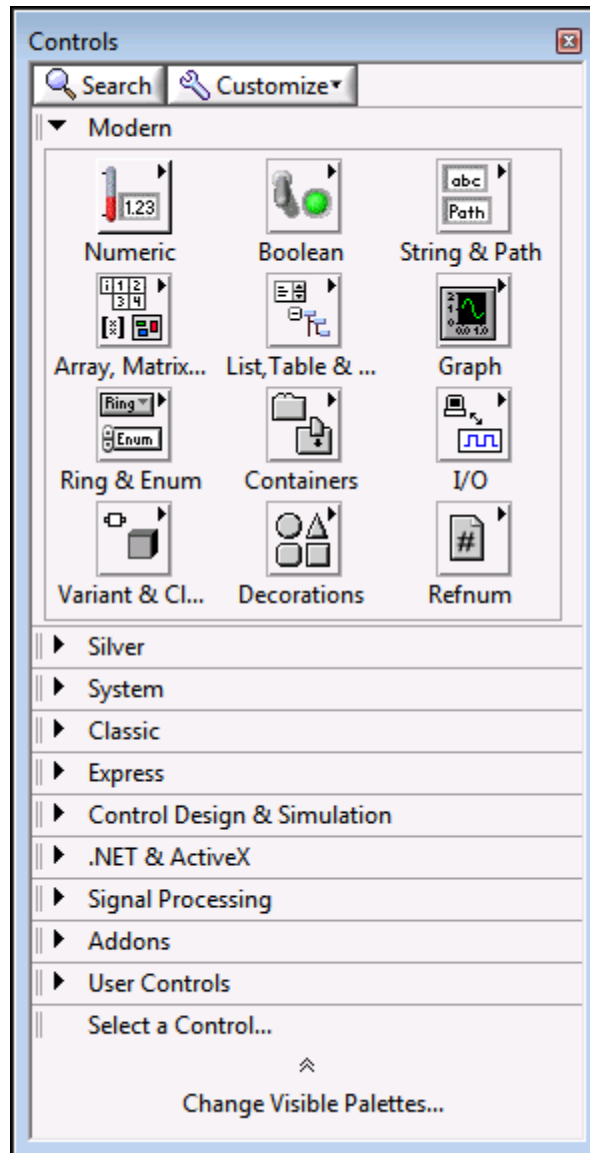


(1) Okno panelu przedniego | (2) Pasek narzędzi | (3) Paleta kontrolerek

Rysunek 1. Przykładowy widok panelu przedniego.

Controls Palette- paleta kontrolerek

Paleta kontrolerek zawiera zestaw zadajników i wskaźników używanych do tworzenia panelu przedniego. Dostęp do palety kontrolerek uzyskasz w oknie panelu przedniego wybierając View»Controls Palette lub klikając prawym przyciskiem myszy na dowolnym pustym miejscu w oknie panelu przedniego. Paleta kontrolerek jest podzielona na różne kategorie. Rysunek 2 przedstawia paletę kontrolerek ze wszystkimi kategoriami.



Rysunek 2. Paleta kontrolerek

Aby wyświetlić lub ukryć kategorie (subpalety), należy kliknąć przycisk **Customize** i należy wybrać **Change Visible Palettes**.

Controls and Indicators – zadajniki i wskaźniki

Każdy VI ma przedni panel, który można zaprojektować jako interfejs użytkownika. Można również użyć panelu przedniego jako sposobu przekazywania danych wejściowych i wyjściowych, podczas wywoływania jednego VI w schemacie blokowym innego VI. Interfejs użytkownika tworzy się umieszczając zadajniki i wskaźniki na panelu przednim danego VI. Podczas interakcji na przednim panelu (interfejsie użytkownika), można modyfikować wartości zadajników i zobaczyć wyniki na wskaźnikach. Kontrolki określają wartości wejść, wskaźniki wyświetlają wartości wyjść.

Zadajniki to zazwyczaj gałki, przyciski, pokrętła, suwaki i pola tekstowe. Symulują wejścia urządzeń i dostarczają dane do bloków na diagramie. Wskaźniki to zwykle wykresy, tabele, diody LED i wyrażenia tekstowe. Wskaźniki symulują wyjścia przyrządów i prezentują dane wyjściowe wynikające z działania schematu blokowego.

Laboratorium Automatyki i Regulacji Automatem

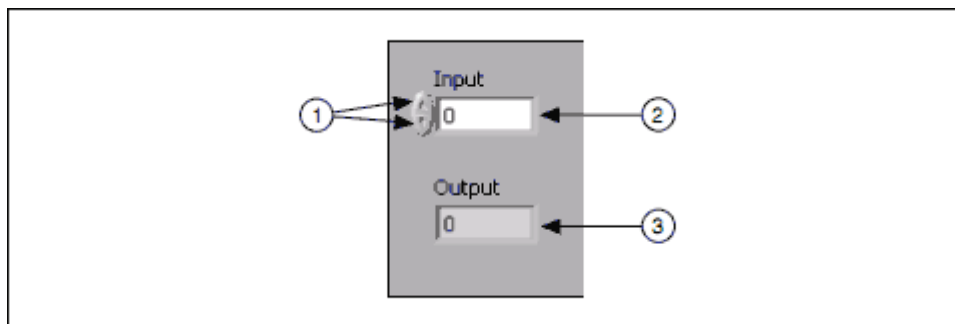
Kod z rysunku 1 ma dwa zadajniki: Number of Measurements i Delay (sec) oraz jeden wskaźnik - XY graph o nazwie Temperature Graph.

Użytkownik może zmienić wartość wejściową zadajników: Number of Measurements i Delay (sec) oraz zobaczyć wartość wygenerowaną przez VI na wskaźniku Temperature Graph. VI generuje wartości wskaźników na podstawie kodu utworzonego na schemacie blokowym.

Każdy wskaźnik lub kontrolka ma typ danych z nim związany. Na przykład, Delay (sec) jest numerycznym typem danych. Najczęściej stosowane typy danych to numeryczne, Boolean i łańcuchowe (string).

Numeryczne zadajniki i wskaźniki

Liczbowy typ danych może reprezentować liczby różnych typów, takich jak liczby całkowite lub rzeczywiste. Dwa wspólne obiekty numeryczny zadajnik i numeryczny wskaźnik pokazano na rysunku 3. Obiekty takie jak zegary oraz tarcze stanowią także dane numeryczne.



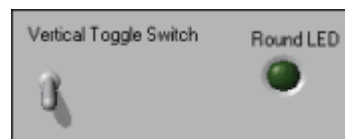
(1) Przyciski inkrementujące/dekrementujące | (2) Zadajnik numeryczny | (3) Wskaźnik numeryczny

Rysunek 3. Numeryczne zadajniki i wskaźniki

Aby wprowadzić lub zmienić wartości w zadajniku numerycznym, należy kliknąć przycisk dekrementujący lub inkrementujący lub kliknąć dwukrotnie zadajnik, wprowadzić nową liczbę, a następnie nacisnąć klawisz <Enter>.

Zadajniki i wskaźniki typu Boolean(dwustanowe)

Typ danych Boolean reprezentuje dane, które mają tylko dwa możliwe stany, takie jak PRAWDA i FAŁSZ lub ON(1) i OFF(0). Zadajniki i wskaźniki typu Boolean służą do zadawania i wyświetlania wartości logicznych. Obiekty typu Boolean symulują przełączniki, przyciski i diody LED. Pionowy przełącznik i okrągłe diody LED przedstawiono na rysunku 4.



Rysunek 4. Zadajniki i wskaźniki typu Boolean.

Zadajniki i wskaźniki łańcuchowe (string)

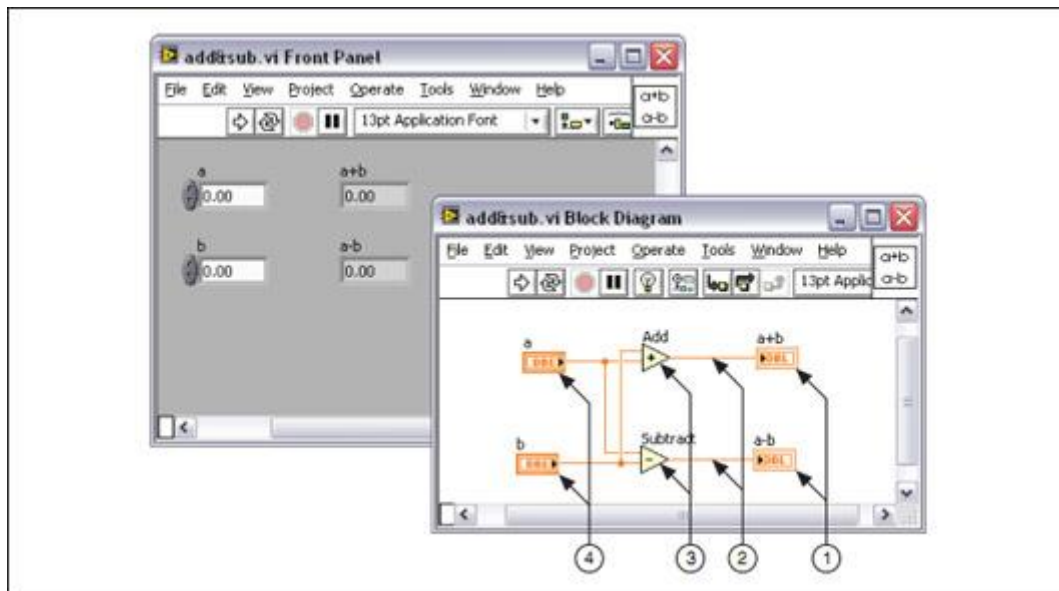
Łańcuchowy typ danych jest ciągiem znaków ASCII. Tych zadajników używa się, by otrzymywać tekst od użytkownika, taki jak hasło lub nazwę użytkownika. Wskaźników łańcuchowych używa się do wyświetlania tekstu dla użytkownika. Najczęstsze obiekty typu string to tabele i pola wprowadzania tekstu, jak pokazano na rysunku 5.



Rysunek 5. Zadajniki i wskaźniki łańcuchowe

Block Diagram

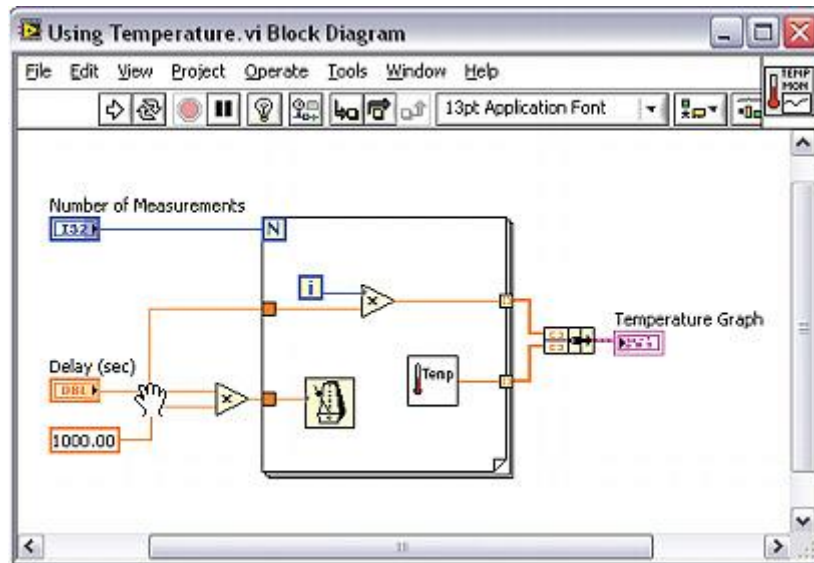
Obiekty schematu blokowego to terminale, subVI, funkcje, stałe, struktury i przewody, które przenoszą dane między różnymi obiektami schematu blokowego.



(1) Terminale wskaźników | (2) Przewody | (3) Węzły | (4) Terminale zadajników

Rysunek 6. Przykładowy schemat blokowy i odpowiadający mu panel przedni.

Po utworzeniu okna przedniego panelu, należy dodać kod za pomocą graficznej reprezentacji funkcji do kontroli obiektów przedniego panelu. Okno schematu blokowego zawiera graficzny kod źródłowy.



Rysunek 7. Schemat blokowy

Terminals

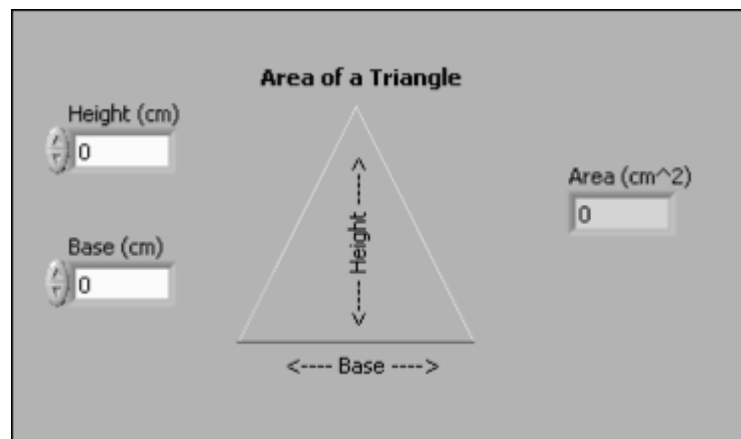
Obiekty na panelu przednim pojawiają się jako terminale na schemacie blokowym. Terminale to obiekty wejściowe i wyjściowe, które służą do wymiany informacji pomiędzy panelem przednim i schematem blokowym. Są one analogiczne do parametrów i stałych w tekstowych językach programowania. Rodzaje terminali obejmują terminale zadajników, wskaźników lub węzłów. Dane, które wprowadzasz na panelu przednim (a i b na rysunku 6) trafiają do schematu blokowego poprzez terminale zadajników. Dane następnie trafiają do funkcji Add (dodawanie) i Subtract (odejmowanie). Gdy te funkcje zakończą swoje obliczenia, produkują nowe wartości danych. Dane wyjściowe przepływają do terminali wskaźników, po czym aktualizują wartości wyświetlane na wskaźnikach na panelu przednim (a + b i a-b na rysunku 6).

Controls, Indicators, and Constants

Zadajniki, wskaźniki i stałe zachowują się jak wejścia i wyjścia algorytmu schematu blokowego. Rozważmy algorytm obliczającego powierzchnię trójkąta :

$$\text{Powierzchnia} = 0,5 * \text{Podstawa} * \text{Wysokość}$$

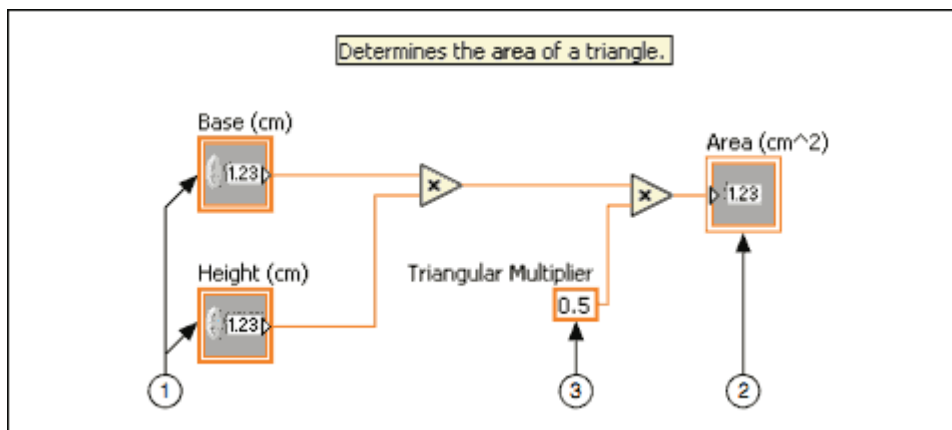
W tym algorytmie, podstawa i wysokość są wejściami, powierzchnia jest wyjściem, jak pokazano na rysunku 8.



Rysunek 8. Panel przedni "Powierzchnia trójkąta"

Użytkownik nie ma dostępu do stałej 0.5, więc nie ma jej na panelu przednim, jest natomiast na schemacie blokowym.

Rysunek 9 przedstawia możliwą realizację algorytmu w LabVIEW. Ten schemat blokowy ma cztery różne terminale utworzone przez dwa zadajniki, jedną stałą i jeden wskaźnik.

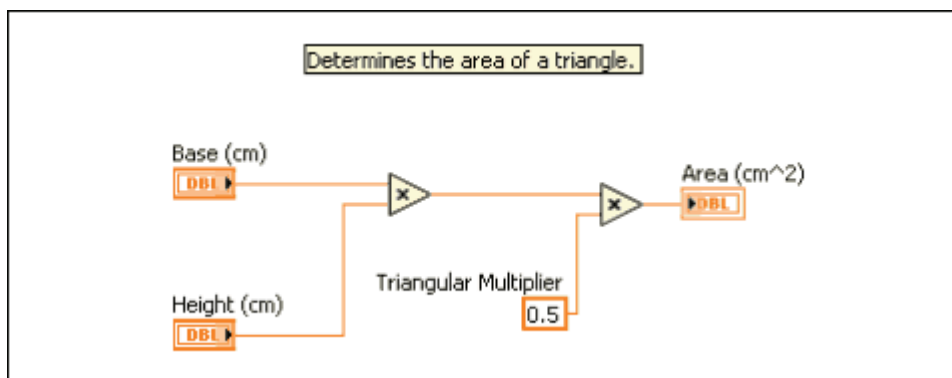


(1) Zadajniki | (2) Wskaźnik | (3) Stała

Rysunek 9. Schemat blokowy "Powierzchni trójkąta" z widokiem ikon terminali.

Terminale Base(cm) i Height(cm) na schemacie blokowym mają inny wygląd od terminalu Area(cm²). Istnieją dwie cechy wyróżniające zadajnik od wskaźnika na schemacie blokowym. Pierwszym jest strzałka na terminalu, która wskazuje kierunek przepływu danych. Zadajniki mają strzałki pokazujące dane wychodzące z terminala, podczas gdy wskaźnik ma strzałkę pokazującą dane wchodzące na terminal. Drugą cechą jest różnica obwódki wokół terminalu. Zadajniki mają grubą obwódkę, a wskaźniki mają cienką.

Można zobaczyć terminale z lub bez widoku ikon. Rysunek 10 pokazuje ten sam blokowy schemat bez użycia widoku ikon terminali; jednak cechy wyróżniające pozostają bez zmian.



Rysunek 10. Schemat blokowy "Powierzchni trójkąta" bez widoku ikon terminali.

Block Diagram Nodes

Węzły są to obiekty na schemacie blokowym, które mają wejścia i/ lub wyjścia i służą do wykonywania operacji, kiedy VI jest uruchomiony. Są one analogiczne do operatorów, funkcji i podprogramów w tekstowych językach programowania. Węzły mogą być funkcjami, podprogramami,

Express VI, lub strukturami. Struktury służą do kontrolowania kolejności wykonywania kodu, np. struktury case, pętle for czy while.

Functions

Funkcje są podstawowymi elementami operacyjnymi LabVIEW. Funkcje Add i Subtract na rysunku 6 to węzły funkcyjne. Funkcje nie mają odpowiednika w oknie panelu, ale mają wejścia i wyjścia dla sygnałów. Dwukrotne kliknięcie funkcji powoduje tylko jej wybranie. Funkcja ma jasnożółte tło ikony w schemacie blokowym.

SubVIs

Po zbudowaniu VI, można go używać w innym VI. VI wywołany ze schematu blokowego innego VI nazywa się subVI (podprogramem). Można ponownie użyć subVI w innych VI. Aby utworzyć subVI, trzeba utworzyć panel złącza oraz ikonę i zdefiniować wejścia/wyjścia danego węzła.

Węzeł subVI odpowiada wywołaniu procedury w tekstowych językach programowania. Węzeł nie stanowi subVI sam w sobie. Schemat blokowy, który zawiera kilka identycznych węzłów subVI wywołuje te same subVI kilka razy.

Zadajniki i wskaźniki danego subVI otrzymują i zwracają dane z i do VI wywołującego. Po dwukrotnym kliknięciu subVI na schemacie blokowym, pojawia się jego okno panelu przedniego. Schemat blokowy zawiera przewody, ikony, funkcje, ewentualnie subVI i inne obiekty LabVIEW.



Każdy VI wyświetla ikonę w prawym górnym rogu okna panelu przedniego oraz w oknie schematu blokowego. Przykładowo, domyślna ikona pokazana jest powyżej. Ikona jest graficznym przedstawieniem VI. Ikona może zawierać tekst i obrazy. Podczas używania VI jako subVI ikona identyfikuje subVI na schemacie blokowym VI. Domyślna ikona zawiera numer, który wskazuje, ile nowych VI zostało otwartych po uruchomieniu LabVIEW.



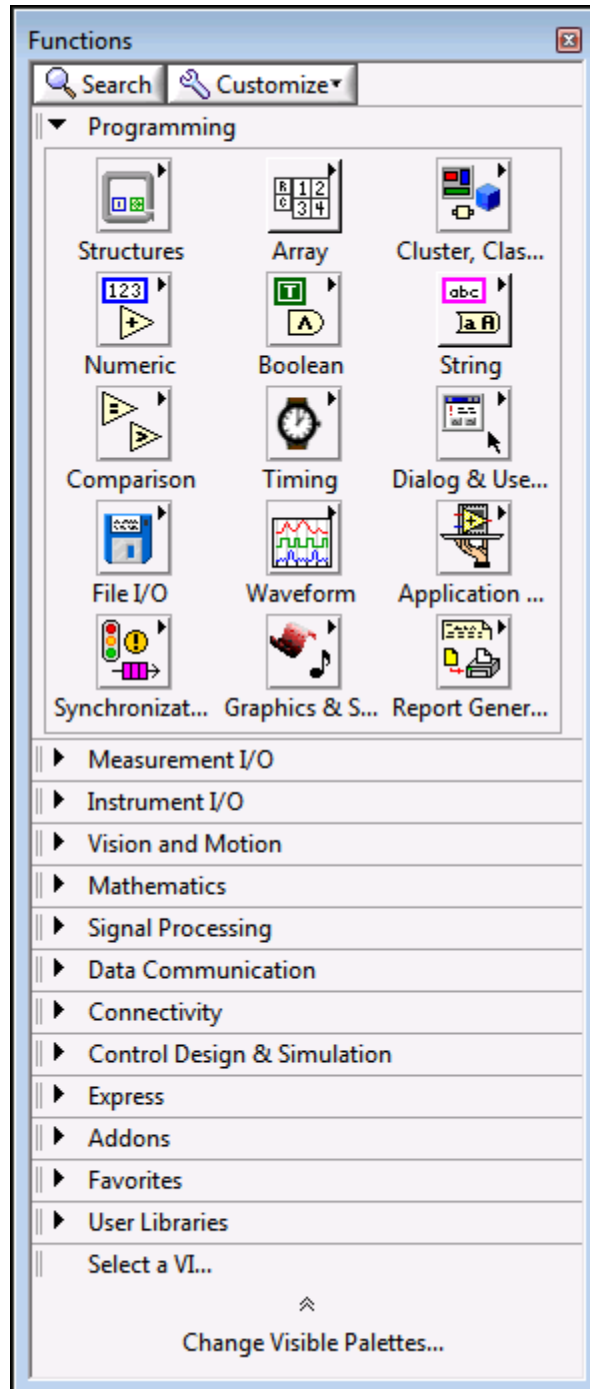
Aby korzystać z VI jako subVI, trzeba zbudować panel złącza, jak pokazano powyżej. Panel złącza to zestaw zacisków dla ikony, która odpowiada zadajnikom i wskaźnikom tego VI, podobnie jak w liście parametrów wywołania funkcji w tekstowych językach programowania. Dostęp do panelu złącza można uzyskać, klikając prawym przyciskiem myszy na ikonę w prawym górnym rogu okna panelu przedniego. Nie można uzyskać dostępu do panelu złącza przez ikonę w oknie schematu blokowego. Ikona subVI ma białe tło.

Express VIs

Express VI to węzły, które wymagają minimalnego okablowania, ponieważ konfiguruje się je z okien dialogowych. Express VI wykorzystuje się dla typowych zadań pomiarowych. Zapoznaj się z tematem Express VI w pomocy LabVIEW, aby uzyskać więcej informacji. Pojawiają się one na schemacie blokowym jako węzły z możliwością rozbudowy(rozciągnięcia) z ikonami otoczonymi niebieskim polem.

Functions Palette

Paleta Funkcji zawiera inne VI, funkcje i stałe używane do tworzenia schematu blokowego. Dostęp do palety funkcji ze schematu blokowego można uzyskać wybierając View» Functions Palette. Paleta Funkcji jest podzielona na różne kategorie. Rysunek 11 pokazuje Paletę Funkcji ze wszystkimi kategoriami.



Rysunek 11. Paleta funkcji

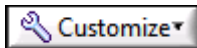
Aby wyświetlić lub ukryć kategorie, należy kliknąć przycisk Dostosuj i należy wybrać Change Visible Palettes.

Searching for Controls, VIs and Functions

Po wybraniu View » Controls lub View» Functions, w górnej części palety wyświetlane są dwa przyciski.

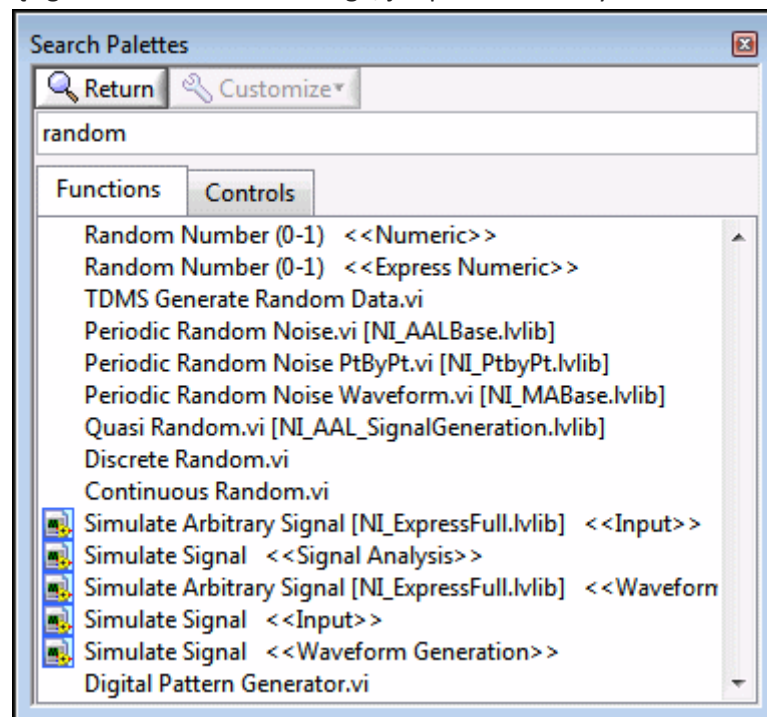


Search - Zmienia paletę w tryb wyszukiwania, dzięki czemu można przeprowadzać wyszukiwania tekstowe VI lub funkcji na paletach. Gdy paleta jest w trybie wyszukiwania, należy kliknąć przycisk Return, aby wyjść z trybu wyszukiwania i powrócić do normalnego trybu palety.



Customize - zawiera opcje umożliwiające wybór odpowiedniego formatu aktualnej palety, pokazując i ukrywając kategorie dla wszystkich palet, także sortując alfabetycznie pozycje w formacie tekstu lub drzewa.

Dopóki użytkownik nie zaznajomi się z lokalizacjami VI i funkcji, powinien on szukać funkcji lub VI za pomocą przycisku Search. Na przykład, jeśli chcesz znaleźć funkcję Random Number, należy kliknąć przycisk Search na pasku narzędzi palety funkcji i rozpocząć wpisywanie Random Number w polu tekstowym w górnej części palety. LabVIEW wyświetli listę wszystkich pasujących elementów, które albo zaczynają się, albo zawierają wpisany tekst. Można kliknąć jeden z wyników wyszukiwania, a następnie przeciągnąć go do schematu blokowego, jak pokazano na rysunku 12.



Rysunek 12. Wyszukiwanie obiektu na palecie funkcji

Należy kliknąć dwukrotnie wynik wyszukiwania, aby zobaczyć jego położenie na palecie.

Quick Drop

Alternatywnie, można znaleźć i umieścić VI i obiekty panelu przedniego wg ich nazwy w oknie dialogowym Quick Drop. Aby uruchomić Quick Drop naciśnij klawisze <Ctrl-Space>.

Quick Drop jest szczególnie przydatny przy szukaniu bardzo specyficznej funkcji lub operacji. Podczas wpisywania, Quick Drop automatycznie uzupełni nazwę pasującymi funkcjami. Po

zaznaczeniu odpowiedniej funkcji, należy kliknąć na schemat blokowy lub panel przedni, aby umieścić obiekt w danym miejscu.

2. Graphical Programming – programowanie graficzne

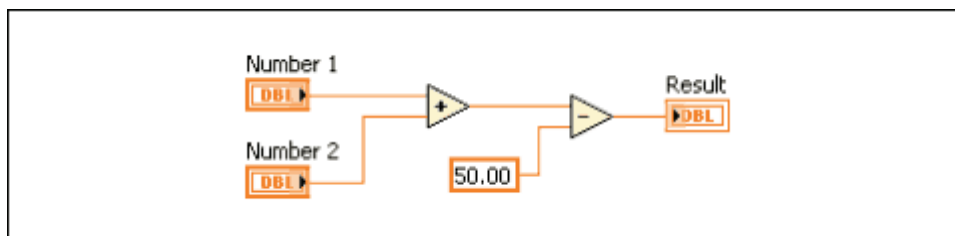
W tej sekcji:

- Wires - przewody
- Automatically Wiring Objects – automatyczne łączenie obiektów
- Manually Wiring Objects – ręczne łączenie obiektów

LabVIEW używa modelu przepływu danych do wykonywania VI. Węzeł schematu blokowego wykonuje się po otrzymaniu wszystkich wymaganych danych wejściowych. Kiedy węzeł jest wykonywany, wytwarza dane wyjściowe i przekazuje je do następnego węzła w ścieżce przepływu danych. Ruch danych poprzez węzły określa kolejność wykonywania kodu.

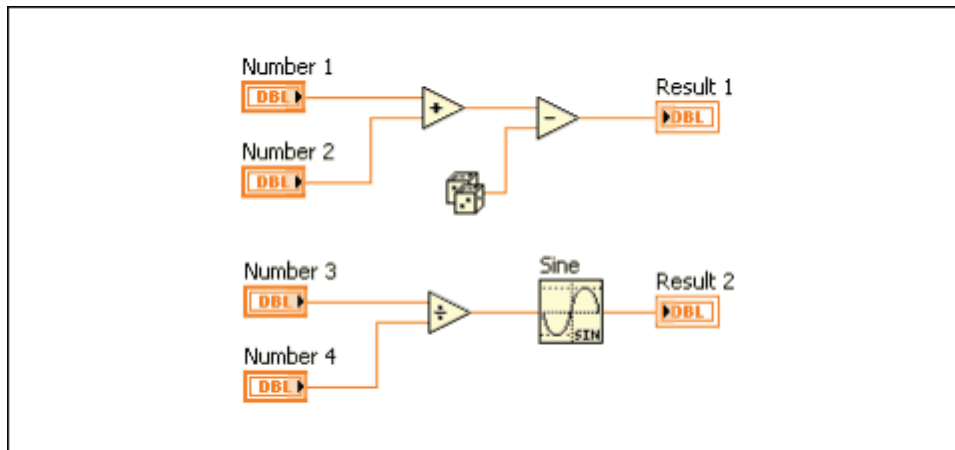
Visual Basic, C ++, Java, i większość innych tekstowych języków programowania używa modelu kontroli przepływu danych, a tym samym wykonywania programu. W kontroli przepływu, sekwencyjna kolejność elementów programu określa kolejność wykonywania programu.

Przykładowy program przepływu danych: program, który dodaje dwie cyfry, a następnie odejmuje 50 od wyniku dodawania tak, jak to pokazano na rysunku 1. W tym przypadku program wykonywany jest od lewej do prawej strony, ale nie dlatego, że przedmioty są umieszczane w tej kolejności, ale dlatego, że funkcja Subtract (Odejmovanie) nie może wykonać się aż zakończy się wykonywanie funkcji Add (dodawania) i dane przekazane zostaną do funkcji Subtract (odejmowanie). Pamiętaj, że węzeł jest wykonywany tylko wtedy, gdy dane są dostępne na wszystkich jego portach wejściowych. Węzeł dostarcza dane do portów wyjściowych tylko wtedy, gdy zakończy swoje działanie.



Rysunek 1. Przykładowy program z przepływem danych.

Rozważmy, który fragment kodu będzie wykonywać się w pierwszej kolejności (wg rysunku 2) - Add, Random Number czy funkcja Divide?. Nie można tego przewidzieć, ponieważ wejścia do funkcji Add i Divide są dostępne w tym samym czasie, a funkcja Random Number nie ma wejścia. W sytuacji, gdy jeden segment kodu należy wykonać przed drugim, a nie istnieje zależność danych pomiędzy nimi należy zastosować inne sposoby programowania, takie jak struktury sekwencyjne lub klastry błędów, aby wymusić kolejność wykonywania kodu.



Rysunek 2. Przykład przepływu danych z kilkoma segmentami kodu.

Wires

Przesyłanie danych pomiędzy obiektami schematu blokowego wykonywane jest poprzez przewody. Na rysunkach 1 i 2, przewody łączą porty zadajników i wskaźników z funkcjami Add i Subtract. Każdy przewód ma jedno źródło danych, ale można podłączyć go do wielu VI i funkcji, które odczytują dane. Przewody mają różne kolory, style i grubości, w zależności przewodzonych typów danych.



Niepoprawne połączenie pojawia się jako czarna, przerywana linia z czerwonym X w środku, jak pokazano powyżej. Przewody takie mogą wystąpić z wielu powodów, np. podczas próby połączenia obiektów o niezgodnych typach danych.

Tabela 1 przedstawia najczęstsze typy przewodów.

Wire Type	Scalar	1D Array	2D Array	Color
Numeric				Orange (floating-point), Blue (integer)
Boolean				Green
String				Pink

Table 1. Najczęstsze typy przewodów.

W LabVIEW można użyć przewodów do podłączania różnych portów ze sobą w celu przekazywania danych w VI. Należy podłączyć przewody do wejść i wyjść, które są zgodne z typem danych, który jest przenoszony przez dany przewód. Na przykład, nie można podłączyć wyjścia typu array do wejścia typu numerycznego. Ponadto kierunek przewodów musi być poprawny. Należy podłączać przewody tylko do jednego wyjścia i co najmniej jednego wejścia. Na przykład, nie można podłączyć dwóch wskaźników ze sobą. Elementy, które określają zgodność okablowania to typ danych zadajnika i/lub wskaźnika oraz typ danych portu. Na przykład, jeśli przełącznik ma zieloną obwódkę, można podłączyć go do dowolnego wejścia z zieloną etykietą obiektu Express VI. Jeśli

pokrętko ma pomarańczową obwódkę, można podłączyć je do dowolnego wejścia z etykietą pomarańczową. Jednakże, nie można podłączyć pomarańczowej gałki do wejścia z zieloną etykietą. Zauważ, że przewody są tego samego koloru, co ikonki funkcji lub VI.

Automatically Wiring Objects

W momencie przeniesienia wybranego obiektu w pobliżu innych obiektów na schemacie blokowym, LabVIEW pokazuje przewody tymczasowe, aby zaproponować połączenia. Po zwolnieniu przycisku myszy LabVIEW automatycznie łączy przewody. Użytkownik także może automatycznie połączyć przewody, gdy obiekty są już na schemacie blokowym. LabVIEW łączy porty, które najlepiej pasują i nie podłącza tych, które nie pasują.

Domyślnie automatyczne okablowanie jest włączone, gdy użytkownik należy wybrać obiekt z palety funkcji lub gdy kopiuje obiekt ze schematu, należy nacisnąć klawisz <Ctrl> i przeciągnąć obiekt. Automatyczne okablowanie jest domyślnie wyłączone w przypadku korzystania z narzędzia pozycjonowania (Positioning Tool).

Można dostosować ustawienia automatycznego okablowania wybierając Tools» Options i wybierając Block diagram z listy kategorii.

Manually Wiring Objects

Po najechaniu narzędziem Wiring Tool nad port, pojawia się pasek podpowiedzi z nazwą portu. Dodatkowo port miga w oknie pomocy kontekstowej oraz na ikonie, co pomoże sprawdzić, czy użytkownik łączy przewód do odpowiedniego portu. Aby połączyć dwa obiekty ze sobą należy użyć Wiring Tool na pierwszym porcie, następnie należy kliknąć, najechać na drugi port oraz kliknąć znowu. Połączeniu, można kliknąć prawym przyciskiem myszy i wybrać Clean Up Wire z menu kontekstowego, aby automatycznie wybrać ścieżkę dla przewodu. Jeśli istnieją niepoprawne przewody do usunięcia, należy nacisnąć < Ctrl + B >.

3. Common Tools in LabVIEW – ogólne narzędzia w LabVIEW

W tej sekcji:

- Selecting a Tool – wybór narzędzia
- Shortcut Menus – menu kontekstowe
- Property Dialog Boxes – okna właściwości obiektów
- Front Panel Window Toolbar – pasek narzędzi panelu przedniego
- Block Diagram Window Toolbar – pasek narzędzi schematu blokowego

W graficznym języku programowania, takim jak LabVIEW, mysz jest głównym sposobem interakcji ze środowiskiem programowania. To znaczy, że kursor musi być w stanie wykonać wiele różnych zadań, takich jak wybór, okablowanie, zaznaczanie tekstu i tak dalej.

Selecting a Tool

Można tworzyć, modyfikować i debugować VI za pomocą narzędzi dostarczonych przez LabVIEW. Narzędzie to specjalny tryb pracy kursora myszy. Tryb pracy definiuje ikonę kursora w

zależności od wybranego narzędzia. LabVIEW automatycznie wybiera narzędzie na podstawie bieżącego położenia kursora.

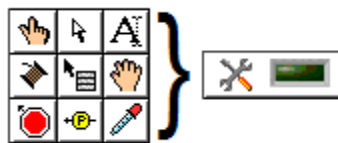


Rysunek 1. Paleta narzędzi

Można ręcznie wybrać potrzebne narzędzie, wybierając go na palecie narzędzi. Należy kliknąć View» Tools Palette, aby wyświetlić paletę narzędzi

Automatic Tool Selector

Górny element w palecie narzędzi to przycisk Automatic Tool Selection. Po jego wybraniu, LabVIEW automatycznie wybiera narzędzie bazując na lokalizacji kursora. Można wyłączyć automatyczny wybór narzędzia, usuwając zaznaczenie elementu lub wybierając inną pozycję w palecie.



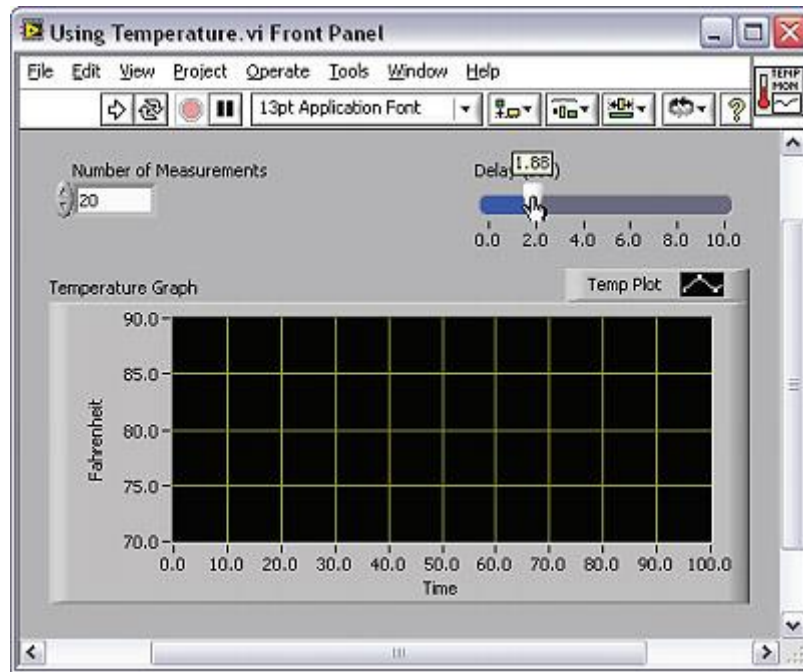
Rysunek 2. Pojedyncze narzędzia i Automatic Tools Selection

Poniżej wymienione są niektóre z bardziej popularnych narzędzi, które można spotkać w LabVIEW. Należy pamiętać, że po naciśnięciu przycisku Automatic Tool Selection narzędzie zmienia się automatycznie.

Operating Tool



Należy użyć Operating Tool, aby zmienić wartości zadajnika. Na przykład, na rysunku 3 Operating Tool przesuwa wskaźnik na Horizontal Pointer Slider. Gdy mysz jest nad wskaźnikiem, kursor automatycznie uzyskuje dostęp do Operating Tool.



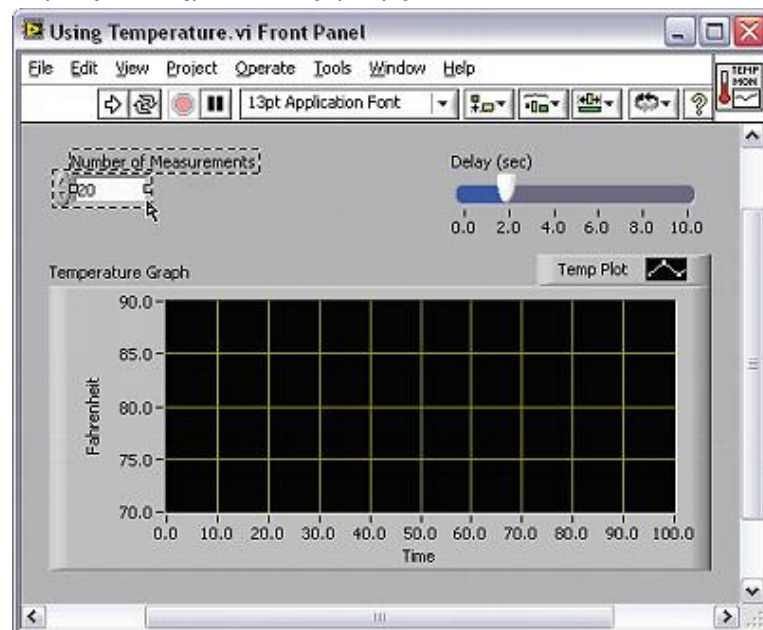
Rysunek 3. Użycie Operating Tool

Operating Tool jest stosowany głównie w oknie panelu przedniego, ale można również skorzystać z tej funkcji na schemacie blokowym, aby zmienić wartość stałej Boolean.

Positioning Tool



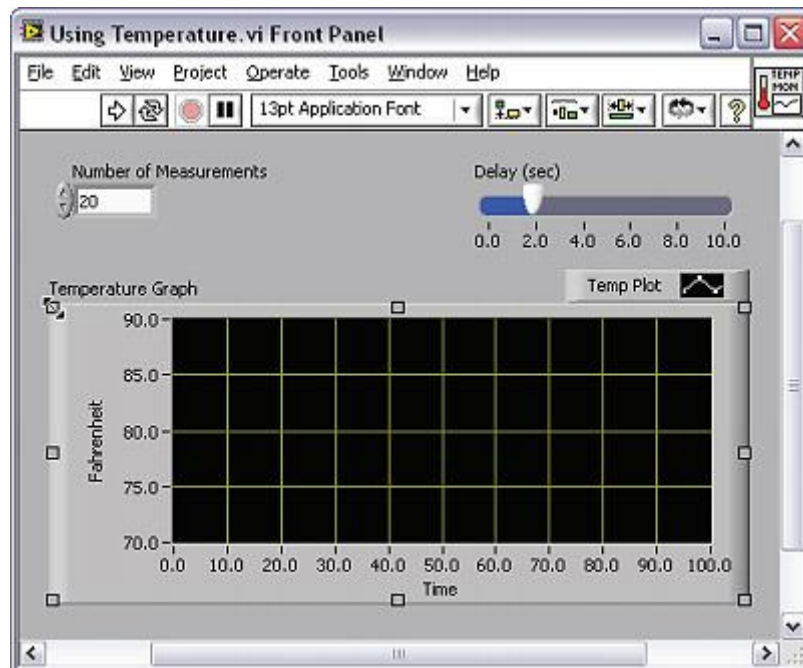
Należy użyć Positioning Tool, aby wybrać lub zmienić rozmiar obiektów. Na przykład, na rysunku 4 Positioning Tool wybiera Number of Measurements w zadajniku numerycznym. Po wybraniu obiektu, można go przenieść, skopiować lub usunąć. Gdy mysz jest nad krawędzią obiektu, kursor automatycznie uzyskuje dostęp do funkcji pozycjonowania.



Rysunek 4. Użycie Positioning Tool do wybrania obiektu.

Laboratorium Automatyki i Regulacji Automatem

Jeśli mysz znajdzie się nad punktem zmiany rozmiaru obiektu to zmienia tryb pracy kursora, aby wskazać, że możliwa jest zmiana rozmiar obiektu (rysunek 5). Cursor znajduje się nad rogiem XY Graph w punkcie zmiany rozmiaru i zmienił tryb pracy na strzałkę dwustronną.



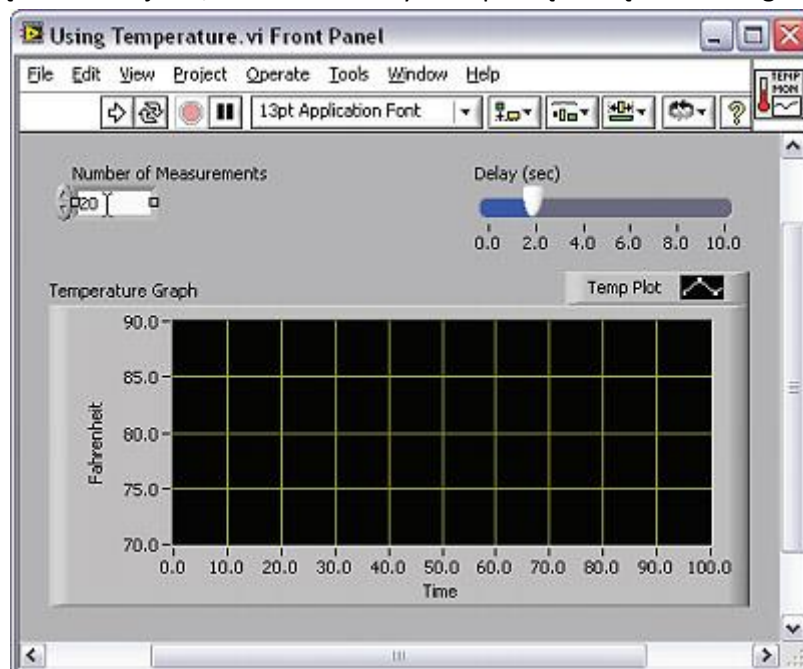
Rysunek 5. Użycie Positioning Tool do zmiany rozmiaru obiektu

Można użyć Positioning Tool zarówno na przednim panelu okna jak i w schemacie blokowym.

Labeling Tool



Za pomocą Labeling Tool można wprowadzić tekst w odpowiednim zadajniku. Na przykład, na rysunku 6 Labeling Tool pozwala wprowadzić tekst w zadajniku Number of Measurements. Gdy mysz znajdzie się nad wnętrzem zadajnika, cursor automatycznie przełącza się do Labeling Tool.



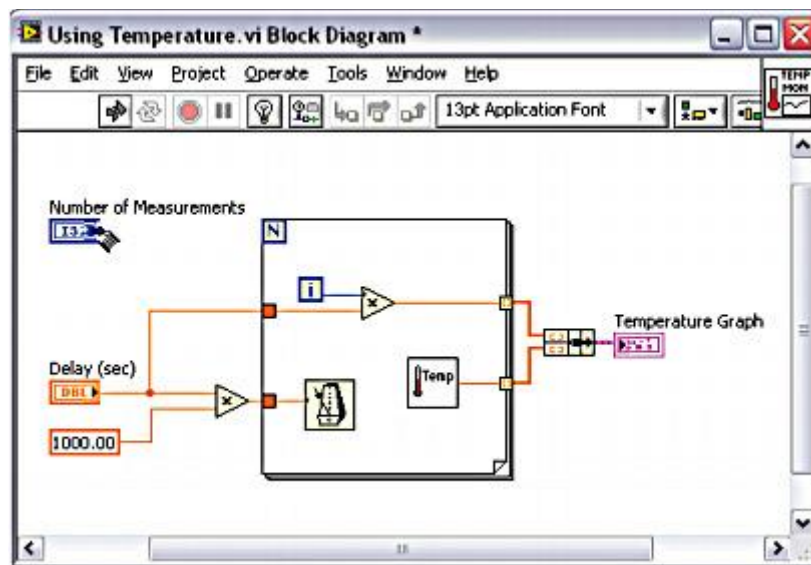
Rysunek 6. Użycie Labeling Tool

Kiedy kursor nie znajduje się w obszarze, który wywołuje określony typ pracy kursora przybiera on postać krzyżyka. Jeśli automatyczny wybór narzędzi jest włączony, można dwukrotnie kliknąć dowolną otwartą przestrzeń, aby uzyskać dostęp do Labeling Tool i utworzyć etykietę.

Wiring Tool



Za pomocą Wiring Tool można łączyć obiekty na schemacie blokowym. Na przykład, na rysunku 7 Wiring Tool łączy port zadajnika Number of Measurements z portem licznika pętli for. Gdy mysz znajdzie się nad portem danego obiektu wówczas kursor automatycznie uzyskuje dostęp do Wiring Tool.



Rysunek 7. Użycie Wiring Tool

Other Tools Accessed from the Palette – Inne narzędzia

Można uzyskać dostęp do następujących dodatkowych narzędzi na paletce:



Należy użyć funkcji Object Shortcut Menu, aby uzyskać dostęp do menu kontekstowego lewym przyciskiem myszy. Można również uzyskać dostęp do tego menu, klikając prawym przyciskiem myszy na dowolny obiekt w LabVIEW.



Należy użyć Scrolling Tool do przewijania okien bez użycia pasek przewijania.



Należy użyć Breakpoint Tool, aby ustawić punkty przerwania w schemacie blokowym. Spowoduje to wstrzymanie wykonywania programu w miejscach punktów przerwań.



Za pomocą Probe Tool można utworzyć sondy na przewodach w schemacie blokowym. Również warto skorzystać z Probe Tool by sprawdzić wartości pośrednie w programie, które generują wątpliwe lub nieoczekiwane wyniki.



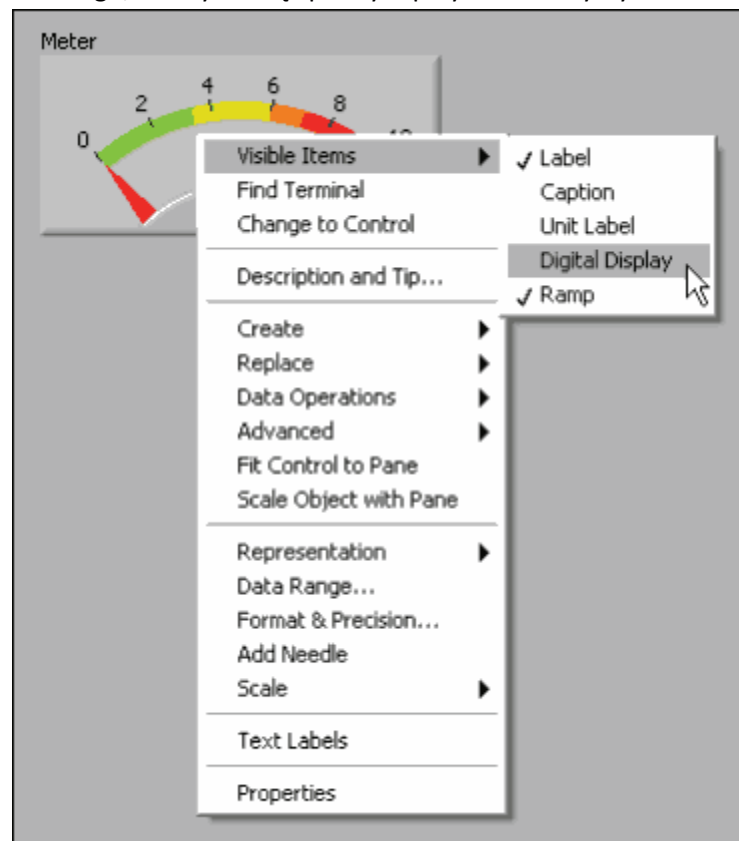
Należy użyć narzędzia Coloring Tool do kolorowania obiektów. Coloring Tool wyświetla również bieżące ustawienia planu oraz tła.



Za pomocą Get Color Tool można skopiować kolory do wklejenia za pomocą Coloring Tool.

Shortcut Menus – menu kontekstowe

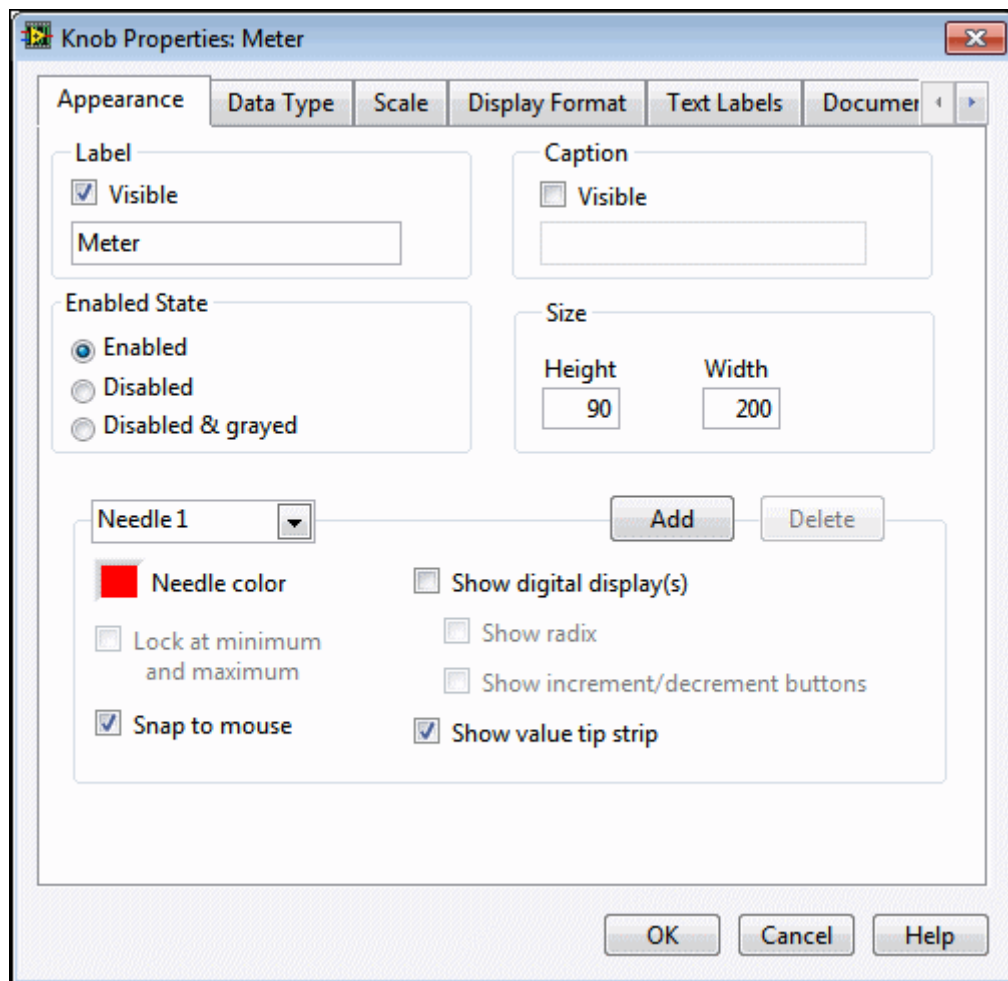
Wszystkie obiekty LabVIEW mają własne menu skrótów, znane również jako menu kontekstowe, menu pop-up. Podczas tworzenia VI, należy użyć elementów menu kontekstowego, aby zmienić wygląd lub zachowanie obiektów panelu przedniego lub schematu blokowego. Aby uzyskać dostęp do menu kontekstowego, należy kliknąć prawym przyciskiem myszy na obiekt.



Rysunek 8. Menu kontekstowe dla wskaźnika

Property Dialog Boxes – Okna właściwości obiektów

Obiekty w oknie panelu przedniego również posiadają okna właściwości, które można wykorzystać do zmiany wyglądu lub zachowania obiektów. Należy kliknąć prawym przyciskiem myszy na obiekt i wybrać polecenie Properties z menu kontekstowego. Rysunek 9 przedstawia okno właściwości dla wskaźnika pokazanego na rysunku 8. Opcje dostępne w oknie właściwości dla obiektu są podobne do opcji dostępnych w menu kontekstowym dla tego obiektu.



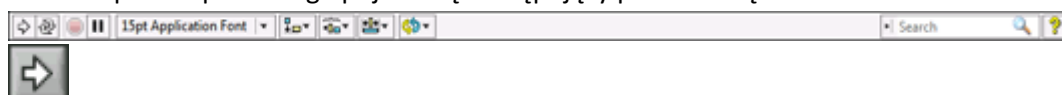
Rysunek 9. Okno właściwości dla wskaźnika

Można wybrać wiele obiektów na panelu przednim lub schemacie blokowym i edytować właściwości obiektów, które są dla nich wspólne. Aby zaznaczyć wiele obiektów, należy użyć Positioning tool, przeciągnąć prostokąt zaznaczenia wokół wszystkich obiektów, które chcesz edytować i przytrzymać klawisz <Shift> podczas klikania każdego obiektu. Należy kliknąć prawym przyciskiem myszy na jeden z zaznaczonych obiektów oraz wybrać Properties z menu podręcznego. Okno dialogowe Properties wyświetla tylko wspólne karty i właściwości dla zaznaczonych obiektów. Jeśli wybrane zostały obiekty, które nie udostępniają żadnych wspólnych właściwości, w oknie dialogowym Properties nie wyświetlą się żadne karty ani właściwości.

Front Panel Window Toolbar – Pasek narzędzi panelu przedniego

Każde okno posiada pasek narzędzi z nim związanych. Należy użyć przycisków paska narzędzi przedniego panelu, aby uruchomić i edytować VI.

W oknie panelu przedniego pojawi się następujący pasek narzędzi:



Należy kliknąć przycisk Run, aby uruchomić VI. LabVIEW kompiluje VI (w razie potrzeby). Można uruchomić VI, jeżeli przycisk Uruchom pojawia się jako stałe białe strzałki, jak pokazano po lewej stronie. Biała strzałka wskazuje również, że można skorzystać z VI jako subVI.



Podczas gdy VI jest uruchomiony, Przycisk Run zmienia wygląd, jak pokazano powyżej, ale tylko gdy VI jest najwyższego poziomu, a zatem nie jest subVI (podprogramem).



Jeśli VI jest uruchomiony jako subVI to przycisk Run wygląda jak powyżej.



Przycisk Run wygląda na uszkodzony, gdy podczas tworzenia lub edycji VI zawiera błąd. Jeśli po zakończeniu okablowania przycisk Run wygląda jak uszkodzony to oznacza, że kod zawiera błędy uniemożliwiające uruchomienie. Należy kliknąć ten przycisk, aby wyświetlić okno Error List, które zawiera listę wszystkich błędów i ostrzeżeń.



Należy kliknąć przycisk Run Continuously w celu uruchomienia VI w sposób ciągły. Można także kliknąć przycisk ponownie, aby wyłączyć tryb pracy ciągłej.



Podczas gdy VI jest uruchomiony, pojawia się przycisk Abort Execution. Należy kliknąć ten przycisk, aby natychmiast zatrzymać VI, jeżeli nie ma innego sposobu.

Uwaga: Przycisk Abort Execution zatrzymuje VI natychmiast, zanim VI zakończy bieżącą iterację. Przerwanie VI, która korzysta z zewnętrznych zasobów, np. sprzętowych, może pozostawić środki w nieznanym stanie przez brak resetowania. Należy zaprojektować VI z przyciskiem Stop, aby uniknąć tego problemu.



Należy kliknąć przycisk Pause, aby wstrzymać pracę VI. Po kliknięciu na przycisk Pause, LabVIEW podkreśla schemat blokowy w lokalizacji, w której wstrzymano wykonywanie, za to pojawia się czerwony przycisk pauzy. Należy kliknąć przycisk Pause, aby kontynuować działanie VI.



Należy wybrać Text Settings z menu rozwijanego, aby zmienić ustawienia czcionki dla wybranych części VI, w tym wielkość, styl i kolor.



Należy wybrać Align Objects z menu rozwijanego w celu wyrównania obiektów wzdłuż osi pionowej.



Należy wybrać Distribute Objects z rozwijanego menu, aby rozłożyć obiekty równomiernie.

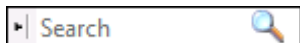


Należy wybrać Resize Objects z menu rozwijanego, aby zmienić rozmiar wielu obiektów panelu przedniego do tego samego rozmiaru.



Należy wybrać Reorder z menu rozwijanego, kiedy są przedmioty, które nakładają się na siebie i należy określić, które z nich mają pozostać w przedniej lub tylnej części ekranu. Należy wybrać jeden

z obiektów używając Positioning Tool, a następnie należy wybrać Move Forward, Move Backward, Move To Front lub Move To Back.



Wyszukiwanie w LabVIEW pomaga znaleźć informacje w pomocy.

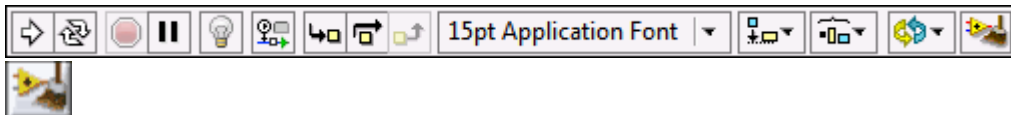


Należy wybrać przycisk Show Context Help Window, aby włączyć wyświetlanie okna Context Help lub użyć skrótu <Ctrl + H>.



Enter Text pojawia się, gdy gdzieś w panelu przednim zmieniła się wartość zadajnika i czeka na zatwierdzenie. Przycisk Enter Text znika po zatwierdzeniu nowo wprowadzanej wartości przez klawisz <Enter>

Block Diagram Window Toolbar – Pasek narzędzi schematu blokowego



Należy kliknąć przycisk Clean Up Diagram, aby automatycznie na nowo poprowadzić wszystkie przewody i wygenerować bardziej przejrzysty układ obiektów na schemacie blokowym. Aby skonfigurować opcje oczyszczania należy wybrać Tools » Options, a następnie w oknie dialogowym Options należy wybrać schemat blokowy: Cleanup.

Pozostałe przyciski na pasku narzędzi są unikalne dla schematu blokowego i są używane przede wszystkim do rozwiązywania problemów z kodem. Zostały szerzej opisane w dziale Debugging Tools.

4. Debugging Tools in LabVIEW – Narzędzia do debugowania

W tej sekcji:

- Fixing Broken VIs – naprawianie niepoprawnego kodu.
- Finding Causes for Broken VIs – odnajdowanie źródeł błędów
- Common Causes of Broken VIs – typowe źródła błędów
- Fixing Incorrect Behavior – odnajdowanie niepoprawnego zachowania kodu
- Block Diagram Toolbar – pasek narzędzi schematu blokowego
- Probe Tool – narzędzie do sondowania

Oprogramowanie LabVIEW zawiera zaawansowane narzędzia do debugowania, które pomagają zidentyfikować obszary problemowe w kodzie, dzięki czemu można dokonać odpowiednich zmian. Można spotkać dwa ogólne typy błędów podczas programowania: te, które uniemożliwiają uruchamianie programu oraz te, które generują złe wyniki lub nieprawidłowe zachowanie programu.

Jeśli nie można uruchomić VI, to LabVIEW informuje o tym zmieniając strzałkę uruchamiania na pękniętą, a w oknie Error List wymienione są konkretne możliwe przyczyny problemów. Drugi rodzaj błędu jest zazwyczaj trudniejszy do wyśledzenia, ale w LabVIEW jest kilka narzędzi, które można użyć, aby obejrzeć jak wykonuje się kod.

Fixing Broken Vis



Jeśli VI nie uruchamia się to jest niepoprawnie stworzony. Przycisk Run zmienia wygląd już podczas tworzenia lub edycji kodu.

Zazwyczaj oznacza to, że wymagane wejście nie jest podłączone albo przewód jest niepoprawnie poprowadzony. Naciśnij przycisk Run, aby uzyskać dostęp do okna listy błędów (Error List). Okno Error List zawiera każdy błąd wraz z wyjaśnieniem. Można kliknąć dwukrotnie pozycję w oknie Error List, aby przejść bezpośrednio do błędu.

Finding Causes for Broken VIs

Ostrzeżenia nie uniemożliwiają uruchomienia VI. Mają one na celu pomóc uniknąć potencjalnych problemów. Błędy mogą jednak uniemożliwić uruchomienie VI. Należy rozwiązać wszelkie błędy, zanim będzie można uruchomić VI.

Należy kliknąć "uszkodzony" przycisk Run lub wybrać View »Error List, aby dowiedzieć się dlaczego VI jest niepoprawny. Okno Error List zawiera listę wszystkich błędów. Elementy z sekcji Errors zawierają listę plików, które zawierają błędy. Sekcja Errors and Warnings zawiera listę błędów i ostrzeżeń dla wybranego VI w pozycji z sekcji Errors. Sekcja Details zawiera opisy błędów i, w niektórych przypadkach, zaleca w jaki sposób skorygować błędy. Należy kliknąć przycisk Help, aby wyświetlić pomoc LabVIEW, która opisuje szczegółowo błąd i zawiera krok po kroku instrukcje dotyczące korekty błędu.

Należy kliknąć na przycisk Show Error lub kliknąć dwukrotnie opis błędu, aby zaznaczyć obszar na schemacie blokowym lub panelu przednim, który zawiera błąd.



Rysunek 1. Przykładowe okno Error List

Common Causes of Broken VIs

Poniższa lista zawiera typowe źródła błędów spotykane podczas edycji:

- Schemat blokowy zawiera niepoprawne połączenia przewodów ze względu na niezgodność typów danych lub przewód jest niepodłączony. Zapoznaj się z tematem pomocy Correcting Broken Wires, aby uzyskać więcej informacji o naprawianiu niepoprawnych przewodów.

Laboratorium Automatyki i Regulacji Automatem

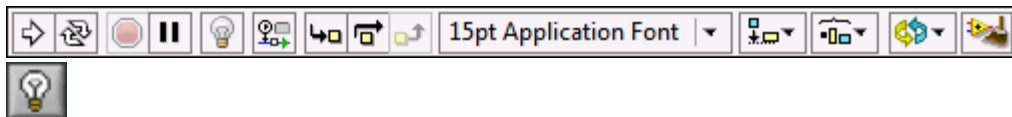
- Port wymagający oprzewodowania jest niepodłączony. Zapoznaj się z tematem pomocy Using Wires to Link Block Diagram, aby uzyskać więcej informacji.
- subVI jest uszkodzony lub został poddany edycji po umieszczeniu jego ikony na schemacie blokowym VI.

Fixing Incorrect Behavior

Jeśli twoje VI uruchamia się, ale daje niepoprawne wyniki to istnieje błąd funkcjonalny w kodzie. LabVIEW oferuje kilka narzędzi, które pomagają przeanalizować dlaczego VI nie zachowuje się zgodnie z oczekiwaniami.

Block Diagram Toolbar

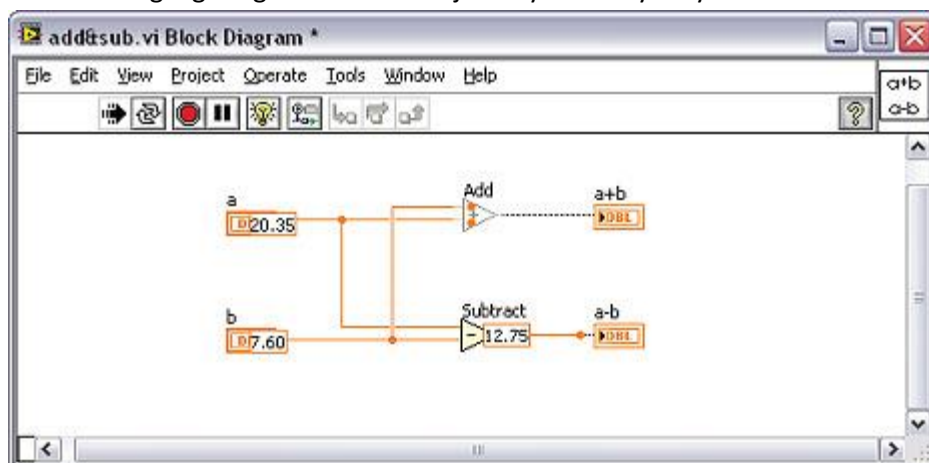
Po uruchomieniu VI, na pasku narzędzi schematu blokowego pojawiają się następujące przyciski, których można użyć do debugowania VI.



Należy kliknąć przycisk Highlight Execution, aby wyświetlić animację wykonywania programu w schemacie blokowym po uruchomieniu VI. Warto zwrócić uwagę na przepływ danych w schemacie blokowym. Należy kliknąć ponownie przycisk, aby wyłączyć podświetlanie wykonywania.

Execution Highlighting pokazuje przepływ danych na schemacie blokowym z jednego węzła do drugiego za pomocą pęcherzyków, które poruszają się wzdłuż przewodów. Używaj tego narzędzia z pracą single-stepping (krok po kroku), aby zobaczyć, jak dane przebiegają od węzła do węzła.

Uwaga: Execution highlighting znacznie zmniejsza szybkość wykonywania kodu VI.



Rysunek 2. Przykład Execution Highlighting w użyciu



Należy kliknąć przycisk Retain Wire Values, aby zapisać wartości w przewodach w każdym punkcie wykonywania tak, że po umieszczeniu sondy na przewodzie można natychmiast odczytać najbardziej aktualną wartość danych, która przebiegła przez przewód. Musisz poprawnie uruchomić VI przynajmniej raz, zanim będzie można zachować wartości z przewodu.





Należy kliknąć przycisk Step Into, aby otworzyć węzeł i wstrzymać. Po ponownym kliknięciu przycisku Step Into, wykonuje pierwszą akcję i pauzuje wraz z następnym działaniem subVI lub struktury. Można również nacisnąć klawisz <Ctrl> i klawisz strzałki w dół.



Należy kliknąć przycisk Step Over, aby wykonać węzeł i wstrzymać w następnym węźle. Można również nacisnąć klawisz <Ctrl> i prawą strzałkę.



Należy kliknąć przycisk Step Out, aby zakończyć wykonywanie bieżącego węzła i wstrzymać wykonywanie. Gdy VI zakończy wykonywanie, przycisk Step Out jest przyciemniony. Można również nacisnąć klawisz <Ctrl> i klawisze strzałek.



Przycisk Warning pojawia się, gdy VI zawiera ostrzeżenie i zaznaczyłeś w polu wyboru „Show Warnings” w oknie Error List. Ostrzeżenie wskazuje na potencjalny problem ze schematem blokowym, ale nie zatrzyma wykonywania VI.

Probe Tool



Za pomocą Probe Tool można sprawdzić wartości pośrednie w przewodzie, podczas pracy VI.

Należy użyć narzędzia Probe Tool, gdy schemat blokowy jest skomplikowany. Warto używać Probe Tool z Execution Highlighting, single-stepping i breakpoints w celu ustalenia czy i gdzie dane są nieprawidłowe. Jeśli dane są dostępne, sonda natychmiast aktualizuje i wyświetla dane w oknie Watch Probe. Gdy wykonywanie VI jest wstrzymane, można również podejrzeć wartość w przewodzie, przez który dane już przepłynęły.

Przydatne skróty klawiaturowe

- <Ctrl+E> - Przełączanie pomiędzy panelem przednim i schematem blokowym
- <Ctrl+H> - Włączanie/wyłączanie okna pomocy kontekstowej
- <Ctrl+B> - Usunięcie wszystkich niepoprawnych połączeń na schemacie blokowym
- <Ctrl+Z> - Cofnij
- <Ctrl+Shift+Z> - Ponów

5. Instrukcja wykonania ćwiczenia

1. Napisać program obliczający objętość prostopadłościanu na podstawie wprowadzanych danych.
2. Napisać program przeliczający stopnie Celsjusza, stopnie Fahrenheit'a i Kelwiny w każdej możliwej kombinacji,
 - a. nie zaokrąglać wyników
 - b. wyniki zaokrąglać do jedności
3. Napisać program zawierający 4 włączniki (switch) i 1 LED. Włączenie dowolnych 3 włączników ma zapalić diodę.
4. Napisać program, który pobiera tekst (imię) i wypisuje go w odwrotnej kolejności.

Laboratorium Automatyki i Regulacji Automatycznej

5. Napisać program, który pobiera tekst i dodaje do niego stałą na początku i końcu, a następnie wyświetla na panelu, np. „Witamy **użytkownika** w LabVIEW!”, gdzie „użytkownika” to tekst pobrany z panelu przedniego.
6. Napisać program, który pobiera 3 liczby: górną granicę, dolną granicę i liczbę testowaną i zwraca użytkownikowi informację, czy testowana liczba jest w powyżej, poniżej czy wewnątrz podanego zakresu,
7. Rozwinąć program z zadania nr 5 tak, aby w oknie dialogowym wyświetlił tekst z zadania lub upomnienie, jeśli nie wpisze żadnego tekstu w polu tekstowym.
8. Stworzyć generator całkowitych liczb losowych z zakresu 0-99.