



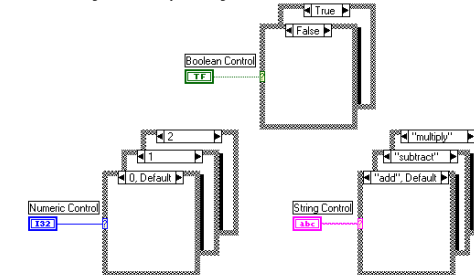
Struktury:

Case, Sequence, Formula Node

dr inż. Roland PAWLICZEK

Case – struktura wyboru

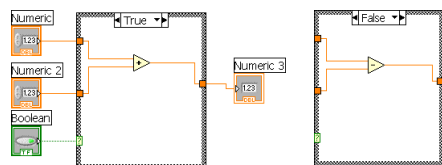
- Struktura **Case** jest analogiczna do struktur **case** lub **if...then...else** w tekstowych językach programowania.
- Struktura Case jest skonfigurowana jak **talia kart**; tylko wierzchnia ramka jest widoczna – pozostałe są ustawione za nią. Tylko jedna ramka jest wykonywana w czasie pracy programu zależnie od wartości **zmiennej sterującej**.
- Zmienna sterująca jest połączona z terminalem wejściowym i może być zmienną typu numerycznego, logicznego lub tekstowego. Jeśli zmienna sterująca jest typu logicznego struktura CASE generuje dwie ramki: **true** i **false**. Dla zmiennej numerycznej można zdefiniować $2^{31}-1$ przypadków.



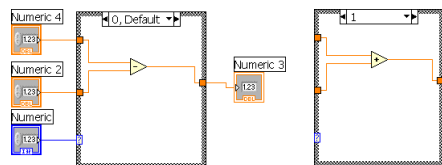
2

Case – struktua wyboru

- **Boolean Case structure:** przykład dwóch ramek z kodem sterowanym zmiennymi logicznymi true i false.



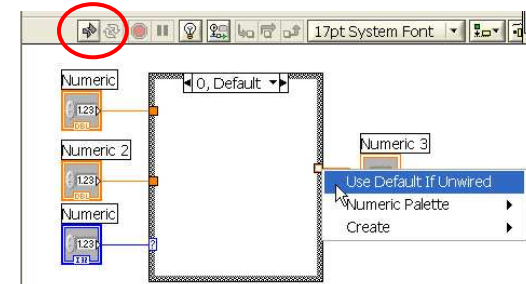
- **Numeric Case structure:** przykład dwóch ramek z kodem sterowanym zmiennymi numerycznymi 0 i 1.



3

Case – struktura wyboru

- Dla numerycznej zmiennej sterującej domyślnie tworzone są dwie ramki: „0, Default” i „1”. Jeżeli potrzebne jest więcej ramek z menu kontekstowego dla struktury **Case** należy wybrać **Add Case After/Before** lub **Duplicate Case**.
- Jeżeli jakaś zmienna jest wykorzystywana tylko w jednej ramce musi być dla niej zdefiniowany tunel wyjściowy lub tunel wyjściowy należy zdefiniować jako **Use Default If Unwired**. Programu nie będzie można uruchomić, dopóki wszystkie tunele wejściowe nie będą miały zdefiniowanych tuneli wyjściowych dla każdej ramki struktury Case.



4

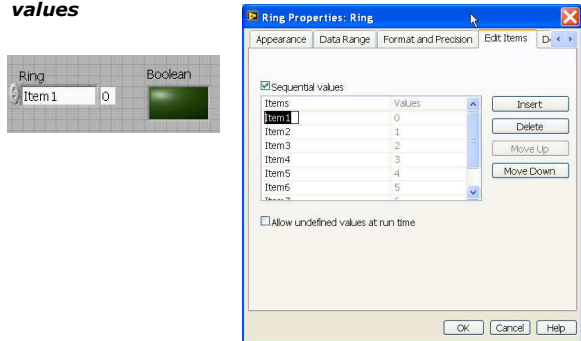
Case structure – proste menu

- Korzystając z kontrolki **Text Ring** z palety **Controls>>Modern>>Ring & Enum** można przypisać poszczególnym numerom odnośniki tekstowe i zbudować proste menu wyboru opcji.

- Menu kontekstowe:

- **Visible Items>>Digital Display**

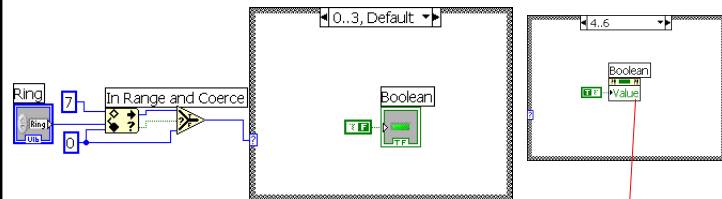
- **Edit Items**, zakładka **Sequential values**: lista pozycji **Sequential values**



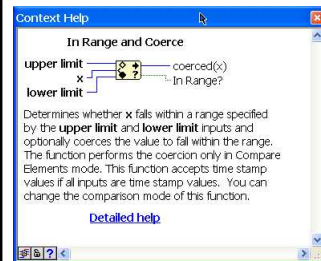
5

Case structure przykład

- Kod programu: **Block Diagram**



Funkcja **In Range and Coerce Functions/Comparison Palette**:

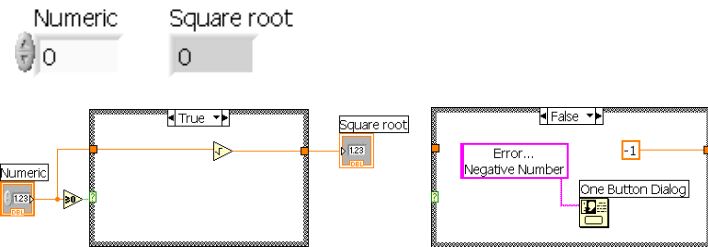


Property Node dla prostokątnej diody LED (Boolean)

6

Case structure – obsługa błędu

- Struktura Case może być wykorzystywana do prostej obsługi błędów: np.. Nie istnieje pierwiastek rzeczywisty z liczby ujemnej.



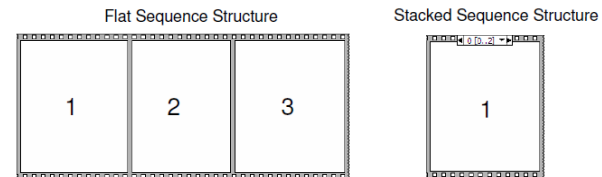
- Funkcja **One Button Dialog (Time & Dialog palette)** wyświetla okienko dialogowe z odpowiednio zdefiniowanym komunikatem.



7

SEQUENCE STRUCTURE

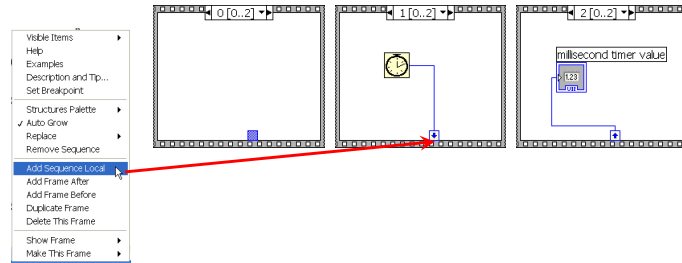
- W tekstowych językach programowania kolejność wykonywania operacji jest definiowana kolejnymi wierszami kodu tekstowego zdefiniowanego w programie głównym
- LabVIEW wykorzystuje **Sequence Structure** do kontroli kolejności wykonywania operacji. Sequence Structure stanowi uszeregowany zestaw ramek, które są wykonywane sekwencyjnie.
- Ramki są wykonywane kolejno i dopiero po zakończeniu ostatniej ramki dane przesyłane tunelami są dostępne.



8

Sequence Structure

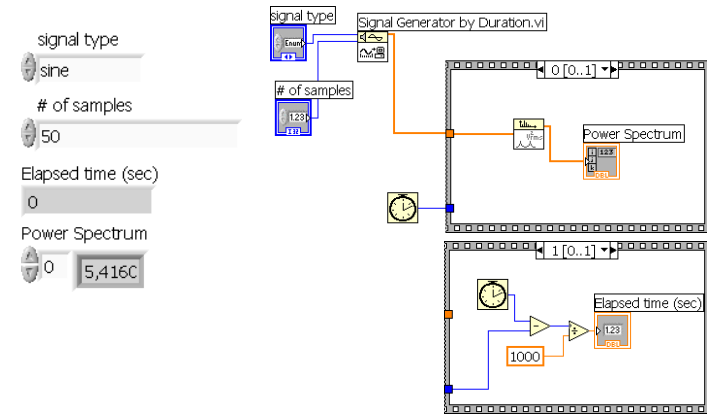
- **Sequence locals** są zmiennymi pozwalającymi przekazywać z ramki dane do następujących po niej ramek
- Sequence locals nie są dostępne w ramach poprzedzających ramkę, w której nastąpiło zdefiniowanie zmiennej.
- Aby utworzyć zmienną lokalną sekwencji należy wskazać wybraną ramkę i używając prawego klawisza myszy wybrać opcję Add Sequence Local.



9

Sequence Structure – ćwiczenie

- Program oblicza czas obliczeń dla różnej liczby próbek generowanego ciągu liczb.

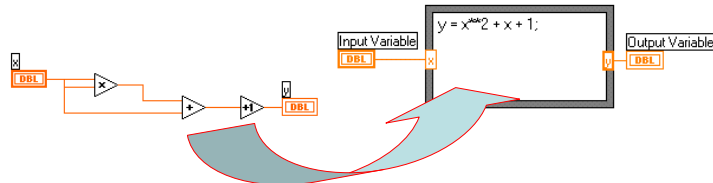


10

FORMULA NODE

Formula Node jest strukturą pozwalającą na wykonanie kodu zapisanego w postaci tekstowej. Jest bardzo wygodna przy programowaniu dużej liczby obliczeń algebraicznych.

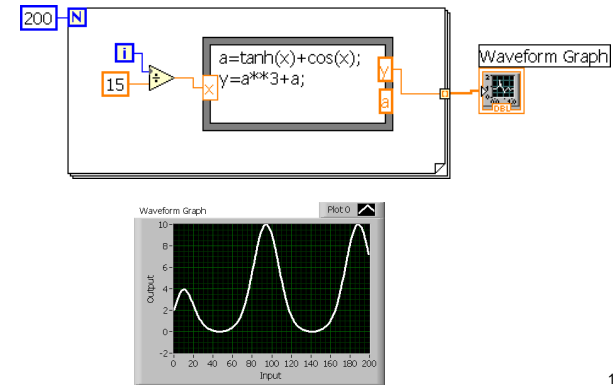
- Aby dodać zmienne wejściowe i wyjściowe należy aktywować menu kontekstowe (prawy klawisz myszy) i wybrać opcję Add Input lub Add Output. W ramce należy wpisać nazwę zmiennej, która będzie stosowana w kodzie tekstowym. Każda linia komend musi być zakończona średnikiem „;”.
- Każda zmienna używana w ramce musi być zadeklarowana.
- Pomoc kontekstowa LabVIEW (**Help»Contents and Index**) pozwala uzyskać opis możliwych do zastosowania operacji i funkcji.



11

FORMULA NODE ćwiczenie

- Zbudować aplikację, która wykreśli wykres funkcji $y = f(x)^3 + f(x)$, where $f(x) = \tanh(x) + \cos(x)$.



12