

## Ćwiczenie 1 – Generator sygnałów

### 1. Cel ćwiczenia

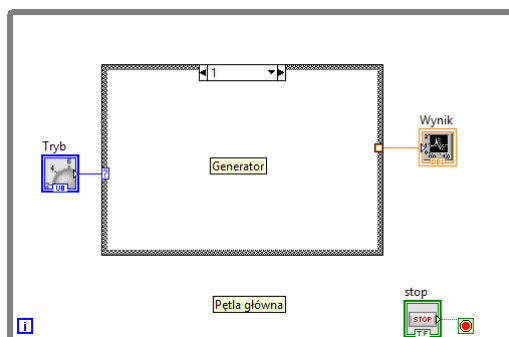
Praktyczne wykorzystanie wiedzy zdobytej na zajęciach wprowadzających i na wykładzie. Nabycie umiejętności realizacji podstawowych zadań oraz implementacji algorytmów wykorzystywanych przy opracowywaniu urządzeń wirtualnych z wykorzystaniem środowiska LabView.

### 2. Wprowadzenie do ćwiczenia

Wykonanie ćwiczenia wymaga wykorzystania wiedzy przedstawionej na zajęciach wprowadzających oraz na wykładzie. Ćwiczenie to należy traktować jako podstawę dalszych prac na zajęciach laboratoryjnych, zdobyte w trakcie realizacji zadania umiejętności zostaną wykorzystane przy tworzeniu wirtualnych przyrządów pomiarowych na kolejnych ćwiczeniach. W tym zadaniu nie wykorzystujemy jeszcze fizycznych interfejsów i możliwości komunikacji z urządzeniami zewnętrznymi.

Ćwiczenie zostało podzielone na 3 części, każda z nich niezależnie realizuje funkcję generatora sygnału. Generatory posiadają wspólne zasady obsługi i prezentacji wyników. Wartości sygnałów wyświetlane są na bieżąco na panelu z zadanyam interwałem czasowym. Do wyznaczenia kolejnych wartości generatory wykorzystują struktury w postaci pętli i warunków, których parametry pracy są ustalane na podstawie nastaw wybranych przez operatora na panelu. Realizacja zadania bazuje na możliwości zapamiętania i przekazania poprzednio wygenerowanej wartości do kolejnej iteracji pętli programu.

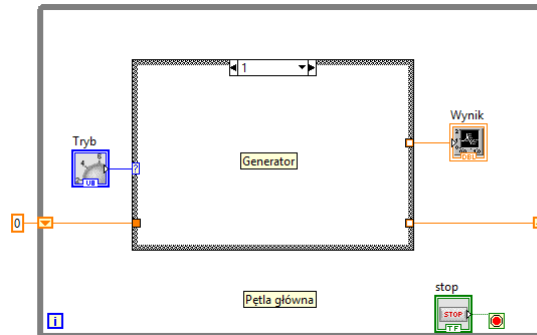
Ze względu na modułowy charakter programu (3 niezależne generatory w jednym urządzeniu) konieczne jest udostępnienie możliwości wyboru przez operatora jednego z trybów pracy. Założono, że zawsze aktywny jest tylko jeden z trybów, w związku z tym sugerowane jest wykorzystanie jednej struktury warunkowej *case* pełniącej funkcję przełącznika trybów. Struktura ta może zawierać wewnątrz pętle w których generowane będą sygnały jednak ze względu na łatwość obsługi i mniejszy stopień skomplikowania proponowane jest wykorzystanie jednej głównej pętli, w której umieszczone będą wszystkie generatory. Szkic takiego rozwiązania przedstawia rysunek 1. Należy pamiętać, że wykorzystanie struktury *case* przełączanej wartością liczbową wymaga ustalenia domyślnego wariantu, który jest realizowany gdy wartość sterująca nie znajduje się na liście wariantów struktury. Lista wariantów nie jest tworzona automatycznie i należy ją ustalić samodzielnie.



Rysunek 1. Proponowany układ głównych struktur programu

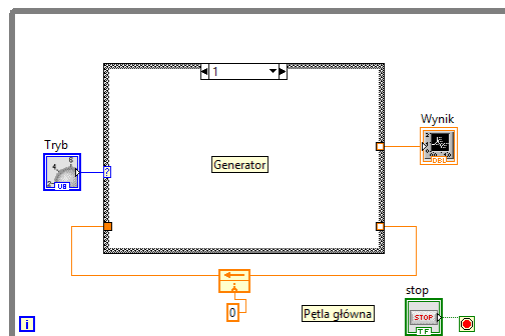
W przedstawionym rozwiązaniu w każdym obiegu pętli powstaje jeden wynik. Dwa z trybów pracy (b i c) wymagają jednak zapamiętania informacji o wyniku poprzedzającym lub wykorzystania licznika iteracji pętli. Ze względu na elastyczność

i łatwość realizacji proponuje się wykorzystanie poprzedzającej wartości generatora. Takie rozwiązanie umożliwi nabycie umiejętności tworzenia pętli ze sprzężeniem zwrotnym, która jest powszechnie wykorzystywana przy realizacji zadań w środowisku LabView. Informacje do kolejnego obiegu pętli można przekazywać na kilka sposobów, najczęściej wykorzystuje się do tego celu opcję *shift register* lub element *feedback node*, które w podstawowym zastosowaniu mają taką samą funkcjonalność. Przykładowe sposoby realizacji przedstawiają rysunki 2 i 3.



Rysunek 2. Przykład użycia opcji *shift register*

Opcja *shift register* jest modyfikacją *tunelu*. Nadanie wartości początkowej (dla „zerowej” iteracji pętli) jest możliwe poprzez wprowadzenie wartości przed wykonaniem pętli (w tym przypadku jest to wartość 0). Dane pomiędzy następującymi po sobie iteracjami przekazywane są poprzez doprowadzenie wartości do elementu oznaczonego trójkątem na prawej krawędzi pętli. W następnej iteracji pętli wartość ta jest wyprowadzana ze skojarzonego elementu umieszczonego na lewej krawędzi pętli.



Rysunek 3. Przykład użycia elementu *feedback node*

Przy domyślnych ustawieniach element *feedback node* ma taką samą funkcję jak element *shift register*. Wartość początkowa podłączana jest do dolnej części elementu. Dane do zapamiętania dla kolejnej iteracji wprowadzamy z prawej strony i zgodnie ze zwrotem strzałki dane te zostaną w następnej iteracji pętli udostępnione z lewej strony elementu (kierunek przepływu danych jest inny niż ogólnie przyjęty dla środowiska LabView). Element *feedback node* udostępnia też dodatkowe opcje takie jak przetrzymanie wprowadzonych wartości na wybraną ilość iteracji pętli, jednak w tym ćwiczeniu nie będą one wykorzystane.

Niezależnie od tego którą z opcji zapamiętywania wyników wykorzystamy, sugeruje się wykorzystanie tylko jednego takiego elementu i nie powielania go przy opracowaniu kolejnych generatorów ze względu na to, że zawsze aktywny jest tylko jeden generator.

Urządzenie ze względu na wykorzystanie kilku trybów pracy stawia dodatkowe wymagania dotyczące metod obsługi panelu operatora. Należy zadbać o to by aktywne były tylko elementy kontrolne wykorzystywane przez wybrany tryb pracy, co zdecydowanie uprości obsługę całego urządzenia. Należy też zapewnić by niemożliwe było wybranie nastaw spoza założonych zakresów oraz aby spełnione były warunki dotyczące zależności między nastawami (np. nastawa  $\min \leq \max$ ). Dynamiczne zależności pomiędzy elementami kontrolnymi można zrealizować wykorzystując węzły *property node* lub zmienne lokalne. Urządzenie opracowane w ramach ćwiczenia ma spełniać założenia opisane w punkcie 3.

### 3. Założenia do zrealizowania

- 1) Panel operatora udostępnia następujące tryby pracy generatora:
  - a) Wartości losowe o rozkładzie równomiernym [2 pkt]:
    - operator ustala suwakiem lub pokrętle: wartość minimalną i maksymalną
    - założenia dla nastaw:  $\min \leq \max$ ,  $\min \geq 0$ ,  $\max \leq 100$  (raster: 1)
  - b) Sygnał piłokształtny [2 pkt]:
    - operator ustala suwakiem lub pokrętle: czas narastania oraz amplitudę sygnału
    - założenia dla nastaw:  $0 \leq \text{amplituda} \leq 100$  (skok: 1), czas narastania (1s-60s, raster: 1s)
    - minimum globalne sygnału = 0
  - c) Sygnał trójkątny [2 pkt]:
    - operator ustala suwakiem lub pokrętle: okres oraz amplitudę sygnału
    - założenia dla nastaw:  $0 \leq \text{amplituda} \leq 100$  (skok: 1), okres (1s-60s, raster: 2s)
    - minimum globalne sygnału = 0
- 2) Zasady obsługi generatora [3 pkt]:
  - a) Tryby pracy przełączane przy użyciu pokrętła lub suwaka z opisanymi pozycjami
  - b) Parametry generatora dla każdego trybu pracy są ustalane niezależnie (osobne elementy kontrolne)
  - c) Operator może zmieniać parametry tylko dla aktywnego trybu pracy
  - d) Częstotliwość próbkowania (generowania pojedynczych wyników) wybierana jest z listy (np. z użyciem *Ring/Enum*) i może wynosić {1Hz, 2Hz, 5Hz, 10Hz}
  - e) Po uruchomieniu programu przyjmowane są domyślne wartości dla parametrów nastawianych przez operatora
- 3) Wyświetlanie wyników [2 pkt]:
  - a) Aktualny wynik wyświetlany jest w oknie liczbowym (wspólnym dla wszystkich trybów ) z zaokrągleniem do 1 cyfry po przecinku
  - b) Historia wyników:
    - historia wyświetlana jest na wykresie (100 najnowszych wyników)
    - skala wykresu ( $Y_{\max}$  i  $Y_{\min}$ ) ustawiana jest na podstawie aktualnych nastaw generatora (zależne od trybu)
    - wykres czyszczony jest po uruchomieniu programu oraz na żądanie operatora (przycisk),
    - \* można rozważyć opcję czyszczenia historii po każdej zmianie trybu pracy

#### Założenia ogólne:

- a) Program zrealizować w postaci pętli *while* zawierającej jedną strukturę *case* zawierającą trzy warianty generatorów
- b) Program uruchamiany jest w trybie ciągłym, operator może na bieżąco zmieniać parametry pracy
- c) Zatrzymanie programu odbywa się przy użyciu przycisku umieszczonego na panelu
- d) Generowane wyniki przekazywane są prezentacji na panelu jako wartości typu Double
- e) Panel jest czytelny i ergonomiczny/intuicyjny [1pkt]

#### Uwagi:

- a) Tryby pracy z sygnałem piłokształtnym i trójkątnym do poprawnego funkcjonowania wymagają uwzględnienia częstotliwości próbkowania nastawionej przez operatora, czasu narastania/okresu oraz amplitudy
- b) Analizując pracę programu należy sprawdzić czy prawidłowo realizowane są nastawy dotyczące wartości granicznych (np. czy osiągnięta jest wartość maksymalna lub minimalna, czy nie jest przekroczona?) oraz czy wyniki generowane są z wybraną częstotliwością

#### Skala ocen:

6-6,5 pkt: 3    7-8 pkt: 3,5    8,5-9,5 pkt: 4    10-11 pkt: 4,5    11,5-12 pkt: 5