

Zagadnienie 3

INTERAKCJA CZŁOWIEK MASZYNA

Programowanie w środowisku LabVIEW – okna konfiguracyjne, mechanizmy przesyłania danych „z potwierdzeniem”.

Celem ćwiczenia jest zapoznanie studenta z metodyką programowania, projektowania i tworzenia aplikacji. Podstawowym zadaniem jest przybliżenie zagadnień i problemów występujących w systemach SCADA. W trakcie ćwiczenia student nabędzie podstawowe informacje dotyczące środowiska i umiejętności posługiwania się nim w procesie tworzenia aplikacji pomiarowych. Niniejsze zagadnienie dotyczy wykorzystania mechanizmów bezpiecznego przesyłania danych takich jak sygnały pomiarowe, komendy.



**Zakład Systemów Informacyjno-
Pomiarowych**

IETiSIP, Wydział Elektryczny, PW



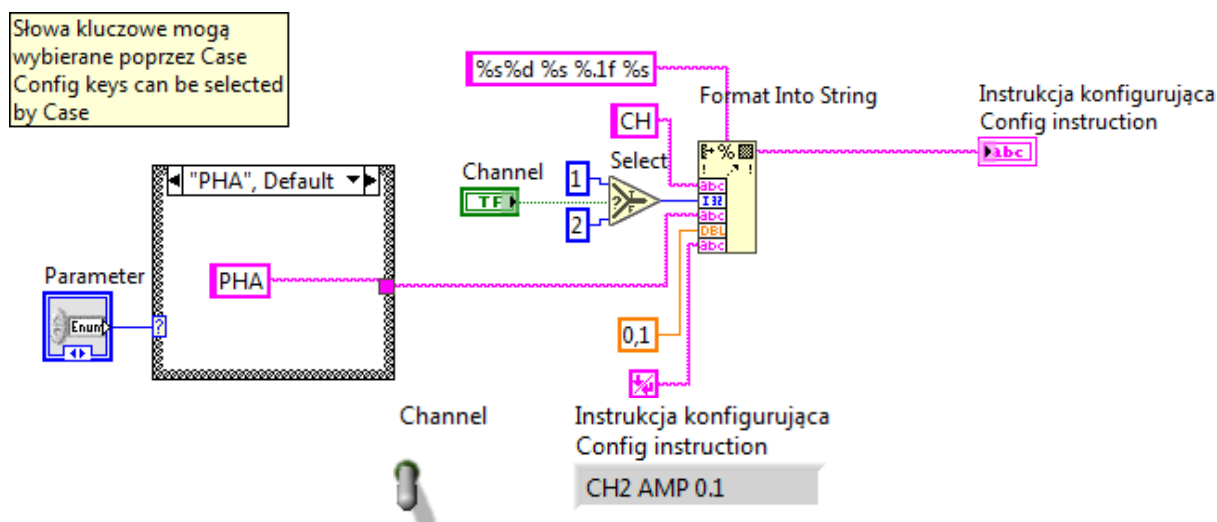
W materiałach do przedmiotu znajdują się przykładowe programy, które mogą okazać się pomocne.

1. Zagadnienie: Dodatkowy ekran sterujący oscyloskopem/generatorem i wysyłający komendy konfiguracyjne. Realizacja tego zadania możliwa jest za pomocą zmiennych globalnych, jednakże ich wadą jest możliwość utraty (pominięcia danych). Z tego względu w tym przypadku wymagany jest "bezpieczny" mechanizm przesyłania danych gwarantujący ich dostarczenie.

2. Możliwe rozwiązanie: Wykorzystując mechanizm **Network Stream** i przykładowy własny protokół opisujący instrukcje konfiguracyjne proszę zrealizować "zdalne sterowanie" poprzez dodatkowy ekran oscyloskopem/generatorem np:

```
CH1 SIN - ustaw przebieg sinusoidalny na kanale 1
CH1 AMP 0.1 - ustaw amplitudę na 0.1 V
CH1 FRE 100 - ustaw częstotliwość na 100 Hz
CH1 PHA 045 - ustaw fazę na 45 stopni itd.
```

Proponowane rozwiązanie wymaga wykorzystania funkcji operujących na ciągach tekstowych. Na rysunkach 1 i 2 przedstawione zostały stosowne przykłady. Stanowią one wskazówkę i na pewno mogą zostać rozwinięte. Warto zwrócić uwagę, że każdy człon "instrukcji" ma stałą długość i jest oddzielony od "sąsiadów" spacjami. To upraszcza tworzenie takiego ciągu i z drugiej strony jego analizę. Rozwiązanie zadania należy realizować w projekcie LabVIEW. Dobrze jest też dodać syntetyczną instrukcję obsługi.

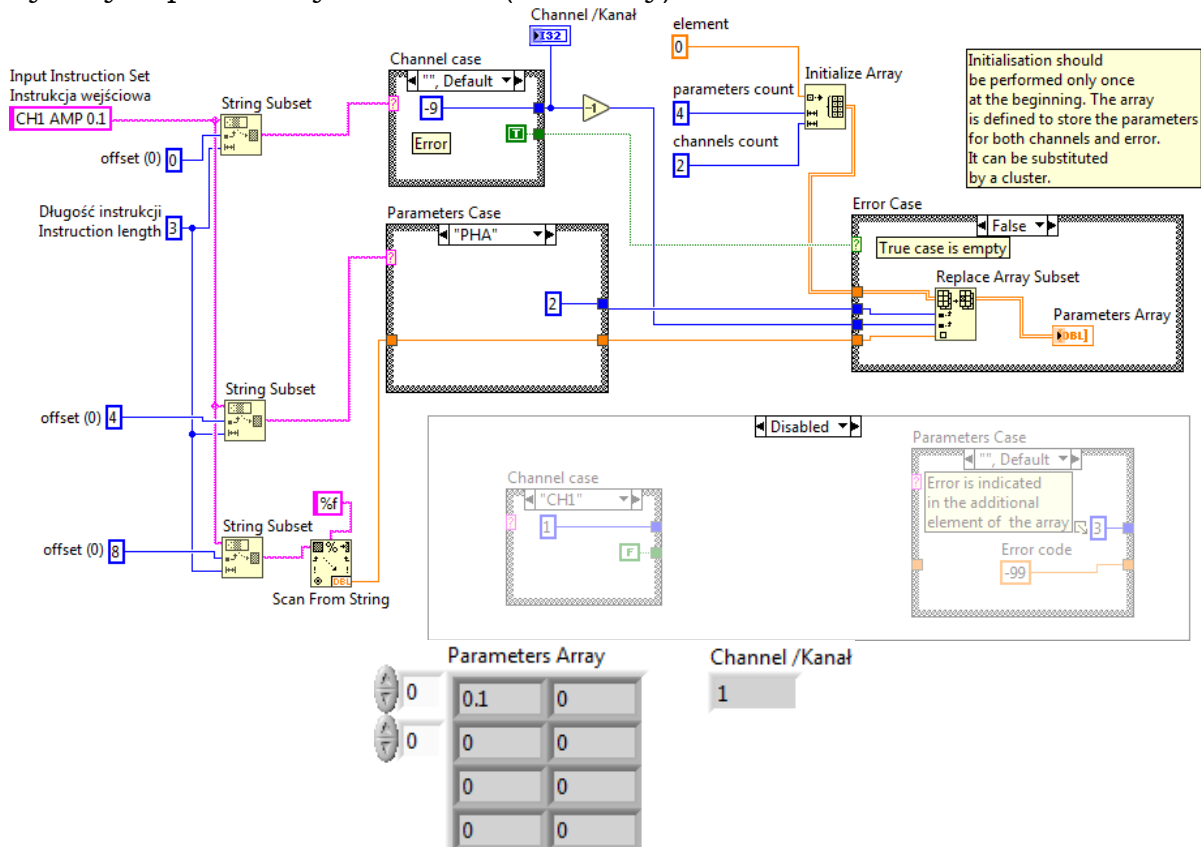


Rys. 1. Przykład tworzenia instrukcji konfiguracyjnej

Na rysunku 2 przedstawiony został przykład odczytu i interpretacji instrukcji konfiguracyjnej. Przykład dotyczy zestawu przykładowych instrukcji



zaprezentowanych powyżej. Dodatkowo uwzględniona została ewentualność pojawienia się błędów. Stąd dodatkowe przypadki **Default** w instrukcjach **Case** dla kanału i parametrów oraz ograniczenie zapamiętywania parametrów tylko jeżeli wszystko się zgadza (**Error Case**). W przykładzie przyjęto, że parametry przechowywane są w tablicy, w której poszczególne kolumny zawierają informacje o osobnych kanałach. Lepszym rozwiązaniem było by zapewne użycie klastra (struktury).



Rys. 2. Przykład odczytu instrukcji konfiguracyjnej

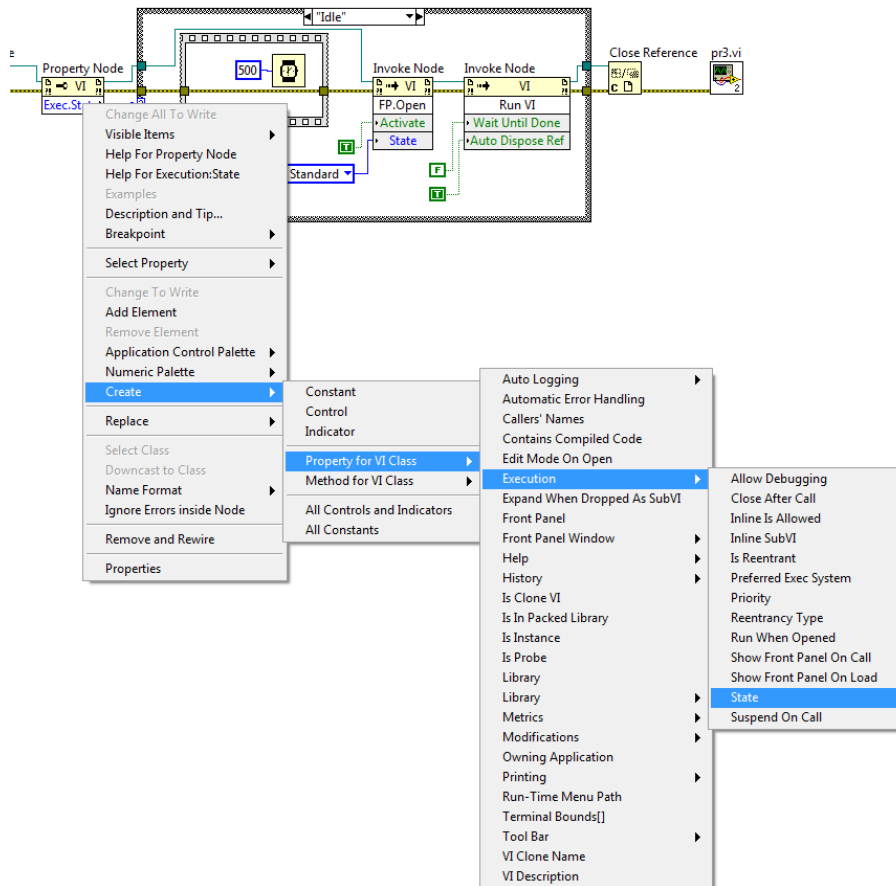
Przed przystąpieniem do uruchomienia mechanizmu Network Stream we własnym programie, warto uruchomić program **LVzmiennie** ze zbioru przykładów. Po uruchomieniu programów "nadawcy" - pr1_writer i "odbiorcy" - pr2_reader przełącznikami z grupy Test (pr1_writer) proszę sprawdzić czy wszystkie zmiany stanów przełączników są rejestrowane po stronie odbiorcy (pr2_reader). Proszę zaobserwować skuteczność przesyłania danych niezależnie dla mechanizmu **Network Stream** oraz **Shared Variables**.

W dalszej części zadania najważniejsza będzie część przykładu obsługująca **Network Stream**. Należy w tym celu przede wszystkim zmienić typ obsługiwanych danych z **Boolean** na **String**. Część nadawcy znajdować się będzie w dodatkowym oknie do konfiguracji, część odbiorcy natomiast w głównym programie symulatora oscyloskopowo-generatorowego. Odebrane dane można wyświetlać w dedykowanym oknie, a zmianę jego wartości traktować jako zdarzenie obsługi (Property node -> Val(Sgnl)). Przy



przesyłaniu danych po stronie nadawczej, może być pomocne wykorzystanie zdarzenia **Mouse Up** w miejsce **Value Change** w **Event Structure**.

Ustawień **Property Node** można dokonać z menu kontekstowego:



Rys. 2. Konfiguracja **Property Node**.

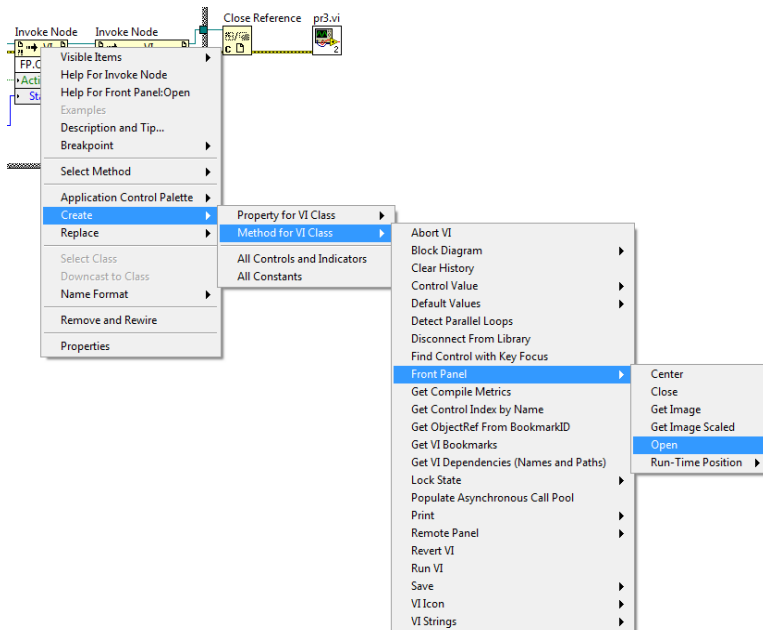
Ustawień **Invoke Node** można dokonać z menu kontekstowego:



Zakład Systemów Informatycznych
Pomiarowych

IETiSIP, Wydział Elektryczny, PW





Rys. 3. Konfiguracja **Invoke Node**.



Zakład Systemów Informacyjno-Pomiarowych

IETiSIP, Wydział Elektryczny, PW

