

# Laboratorium Komputerowych Systemów Pomiarowych

## ĆWICZENIE NR 4

### POMIARY PARAMETRÓW SYGNAŁÓW CYFROWYCH ZA POMOCĄ MODUŁU AKWIZYCJI DANYCH POMIAROWYCH Z INTERFEJSEM USB

(opracował Eligiusz Pawłowski)

#### Cel i zakres ćwiczenia

Celem ćwiczenia jest praktyczne zapoznanie się studentów z problematyką programowania w środowisku LabVIEW systemu pomiarowego wykorzystującego uniwersalny moduł akwizycji danych pomiarowych NI USB-6008/6009 wyposażonego w interfejs USB.

Program laboratorium obejmuje następujące zagadnienia:

- zapoznanie się ze środowiskiem LabVIEW i podstawami programowania,
- umieszczanie kontrolki na panelu programu i tworzenie interfejsu użytkownika,
- tworzenie diagramu sterującego przebiegiem programu,
- zapoznanie się z budową modułu akwizycji danych pomiarowych NI USB-6008/6009,
- konfigurowanie portów wejść i wyjść cyfrowych modułu NI USB-6008/6009,
- konfigurowanie licznika impulsów modułu NI USB-6008/6009,
- konfigurowanie przetwornika cyfrowo-analogowego modułu NI USB-6008/6009,
- pomiary czasu i częstotliwości poprzez zliczanie impulsów sprzętowym licznikiem z programowym odmierzaniem czasu pomiaru, sterowanie układów zewnętrznych sygnałami cyfrowymi i analogowymi, odczytywanie stanów logicznych na wejściach cyfrowych modułu,
- dodatkowa analiza sygnałów: tworzenie histogramu, rejestracja i zapis wyników pomiarów do pliku tworzenie wykresów w funkcji czasu i wykresów XY.

#### 1. Opis stanowiska laboratoryjnego

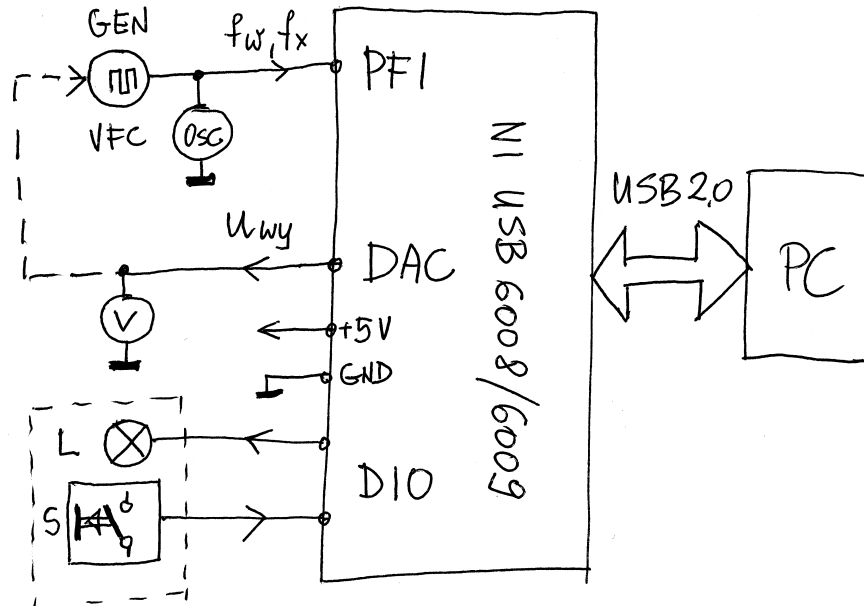
Wykorzystywane w ćwiczeniu stanowisko laboratoryjne składa się z następujących elementów składowych:

- komputera klasy PC z zainstalowanym środowiskiem National Instruments LabView,
- modułu akwizycji danych pomiarowych National Instruments USB-6008/6009,
- płytki z układami wejść - wyjść cyfrowych,
- generatora częstotliwości wzorcowych,
- przetwornika napięcie-częstotliwość,
- silnika prądu stałego z układem sterowania i przetwornikiem obrotowo - impulsowym,
- pomocniczych przyrządów pomiarowych: woltomierza i oscyloskopu.

Wszystkie elementy składowe są ze sobą połączone i gotowe do realizacji ćwiczenia. Studenci przełączają tylko źródła sygnałów w zależności od realizowanego punktu. Przed rozpoczęciem pracy w środowisku LabVIEW studenci przeprowadzają test modułu pomiarowego korzystając z programu Measurement & Automation Explorer. Zadania realizowane podczas ćwiczenia polegają na samodzielnym przygotowaniu odpowiednich programów w środowisku LabVIEW, uruchomieniu ich, przetestowaniu poprawności działania, usunięciu błędów, wykonaniu przykładowych pomiarów i opracowaniu sprawozdania.

Schemat blokowy stanowiska laboratoryjnego przedstawia rysunek 1. Moduł akwizycji danych pomiarowych USB-6008/6009 dołączony jest do komputera klasy PC za pomocą interfejsu USB. Do wejścia PFI (*Programmable Function Interface*) modułu pomiarowego

dołączone jest wyjście  $f_w$  generatora GEN częstotliwości wzorcowej lub wyjście  $f_x$  przetwornika napięcie-częstotliwość VFC (*Voltage-to-Frequency Converter*). Przetwornik VFC może być również zastąpiony silnikiem prądu stałego małej mocy z przetwornikiem obrotowo-impulsowym. Wyjście przetwornika cyfrowo-analogowego DAC (*Digital-to-Analog Converter*) steruje wejściem przetwornika VFC. Do wejść-wyjść cyfrowych DIO (*Digital Input Output*) dołączona jest płytką z układami wejść-wyjść cyfrowych zawierająca lampkę sygnalizacyjną i przycisk sterujący.



Rys. 1. Schemat blokowy stanowiska laboratoryjnego

### 1.1. Moduł akwizycji danych pomiarowych NI USB-6008/6009

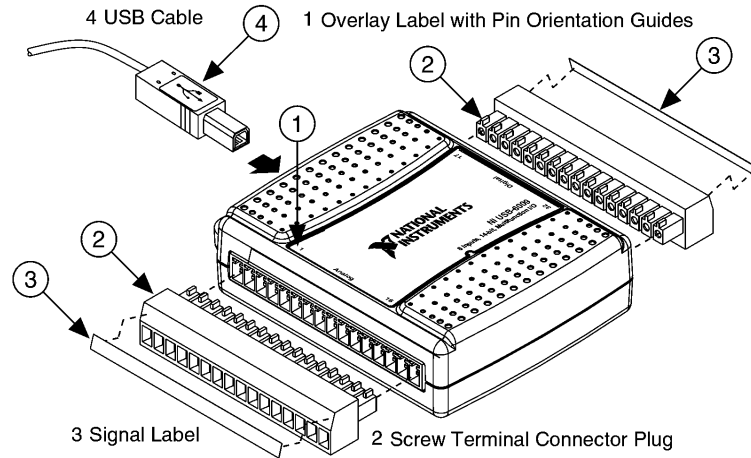
Wykorzystywany w ćwiczeniu moduł akwizycji danych występuje w dwóch odmianach: USB-6008 (wolniejszy i o niższej rozdzielczości) oraz USB-6009 (szybszy i o wyższej rozdzielczości). W tabeli 1 zestawiono najważniejsze parametry obu modułów.

Tab. 1. Zestawienie parametrów modułów USB-6008 i USB-6009

Feature	NI USB-6008	NI USB-6009
AI resolution	12 bits differential, 11 bits single-ended	14 bits differential, 13 bits single-ended
Maximum AI sample rate, single channel*	10 kS/s	48 kS/s
Maximum AI sample rate, multiple channels (aggregate)*	10 kS/s	48 kS/s
DIO configuration	Open collector <sup>†</sup>	Each channel individually programmable as open collector or active drive <sup>†</sup>
* System-dependent. <sup>†</sup> This document uses NI-DAQmx naming conventions. Open-drain is called open collector and push-pull is called active drive.		

Moduł umieszczony jest w niewielkiej obudowie z tworzywa sztucznego o wymiarach ok. 63,5mm x 85,1mm x 23,2mm i masie 54g. Wygląd zewnętrzny modułu przedstawiono na rysunku 2. Na bocznych ściankach umieszczono dwa rzędy konektorów 2 z zaciskami

śrubowymi z możliwością umieszczenia etykietek 3 z opisami sygnałów. Rozmieszczenie zacisków jest również umieszczone na etykiecie 1 umieszczonej na górnej powierzchni obudowy modułu. Na krótszej ścianie obudowy umieszczone jest złącze typu B interfejsu USB umożliwiające dołączenia kabla 4 łączącego moduł z komputerem sterującym.



Rys.2. Wygląd zewnętrzny modułu NI USB-6008/6009

Obok złącza umieszczona jest dioda LED sygnalizująca aktualny status modułu. Sposób sygnalizacji tej diody LED przedstawiono w tabeli 2. Podczas normalnej pracy, gdy moduł został prawidłowo zainstalowany i rozpoznany przez system, dioda LED błyska regularnie. Ciągłe świecenie oznacza obecność napięcia zasilającego i sygnalizuje niewłaściwą pracę modułu, np. niepoprawne zainicjowanie modułu, niekompletne lub niewłaściwie zainstalowane oprogramowanie, komputer w trybie *standby*. Zgaszona dioda LED oznacza wyłączone urządzenie.

Tab. 2. Status modułu USB-6008/6009 sygnalizowany diodą kontrolną LED

LED State	Device Status
Not lit	Device not connected or in suspend
On, not blinking	Device connected but not initialized, or the computer is in standby mode. In order for the device to be recognized, the device must be connected to a computer that has NI-DAQmx installed on it.
Single-blink	Operating normally

Każde ze złączy modułu grupuje określony rodzaj sygnałów. Pierwsze z nich posiadające stykami o numerach od 1 do 16 umożliwia dołączenie sygnałów analogowych. Drugie złącze posiadające styki o numerach od 17 do 32 umożliwia dołączenie sygnałów cyfrowych.

Szczegółowy opis poszczególnych styków na złączach modułu przedstawiono w tabelce 3. Osiem wejść analogowych AI <0..7> umożliwia pomiary czterech napięć maksymalnie do wartości 20V w trybie różnicowym DIFF lub ośmiu napięć maksymalnie do wartości 10V w trybie niesymetrycznym RSE względem zacisku masy GND. Tematyce wykorzystania przetwornika analogowo-cyfrowego poświęcone jest oddzielne ćwiczenie i tutaj zagadnienie to nie będzie omawiane.

Dwanaście linii danych cyfrowych pogrupowano w dwa porty: P0 <0..7> i P1 <0..3>. Każda z tych linii może być niezależnie skonfigurowana jako wejście lub wyjście cyfrowe. Parametry wejść-wyjść cyfrowych zestawiono w tabelce 4.

Tab. 3. Opis styków na złączach modułu USB-6008/6009

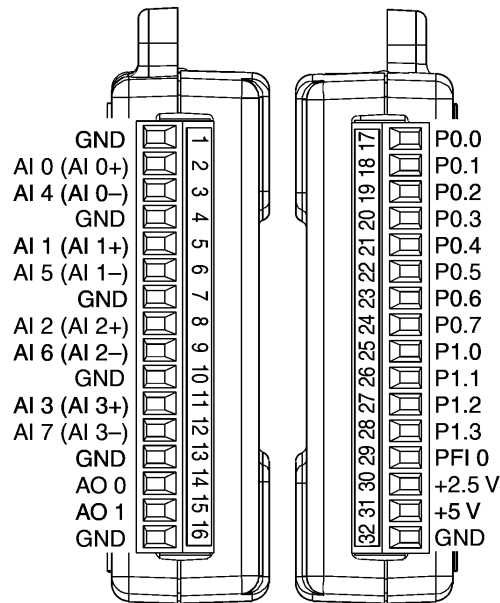
Signal Name	Reference	Direction	Description
GND	—	—	<b>Ground</b> —The reference point for the single-ended analog input measurements, analog output voltages, digital signals, +5 VDC supply, and +2.5 VDC at the I/O connector, and the bias current return point for differential mode measurements.
AI <0..7>	Varies	Input	<b>Analog Input Channels 0 to 7</b> —For single-ended measurements, each signal is an analog input voltage channel. For differential measurements, AI 0 and AI 4 are the positive and negative inputs of differential analog input channel 0. The following signal pairs also form differential input channels: AI<1, 5>, AI<2, 6>, and AI<3, 7>. Refer to the <i>Analog Input</i> section for more information.
AO <0, 1>	GND	Output	<b>Analog Output Channels 0 and 1</b> —Supplies the voltage output of AO channel 0 or AO channel 1. Refer to the <i>Analog Output</i> section for more information.
P0.<0..7>	GND	Input or Output	<b>Port 0 Digital I/O Channels 0 to 7</b> —You can individually configure each signal as an input or output. Refer to the <i>Digital I/O</i> section for more information.
P1.<0..3>	GND	Input or Output	<b>Port 1 Digital I/O Channels 0 to 3</b> —You can individually configure each signal as an input or output. Refer to the <i>Digital I/O</i> section for more information.
PFI 0	GND	Input	<b>PFI 0</b> —This pin is configurable as either a digital trigger or an event counter input. Refer to the <i>PFI 0</i> section for more information.
+2.5 V	GND	Output	<b>+2.5 V External Reference</b> —Provides a reference for wrap-back testing. Refer to the <i>+2.5 V External Reference</i> section for more information.
+5 V	GND	Output	<b>+5 V Power Source</b> —Provides +5 V power up to 200 mA. Refer to the <i>+5 V Power Source</i> section for more information.

Wejście cyfrowe PFI (*Programmable Function Interface*) może pełnić rolę wejścia zliczającego impulsy dla licznika impulsów lub wejścia wyzwalającego pomiaru przetwornikiem ADC. Pojemność licznika wynosi 32 bity i może on zliczać impulsy o parametrach zgodnych ze standardem TTL i częstotliwości do 5 MHz. Wejście licznika posiada wewnętrzny rezystor dołączony do napięcia zasilającego 5V, który wymusza stan wysoki i umożliwia bezpośrednią współpracę z pasywnymi stykami elektromechanicznymi. Pozostałe parametry licznika impulsów zestawiono w tabelce 5.

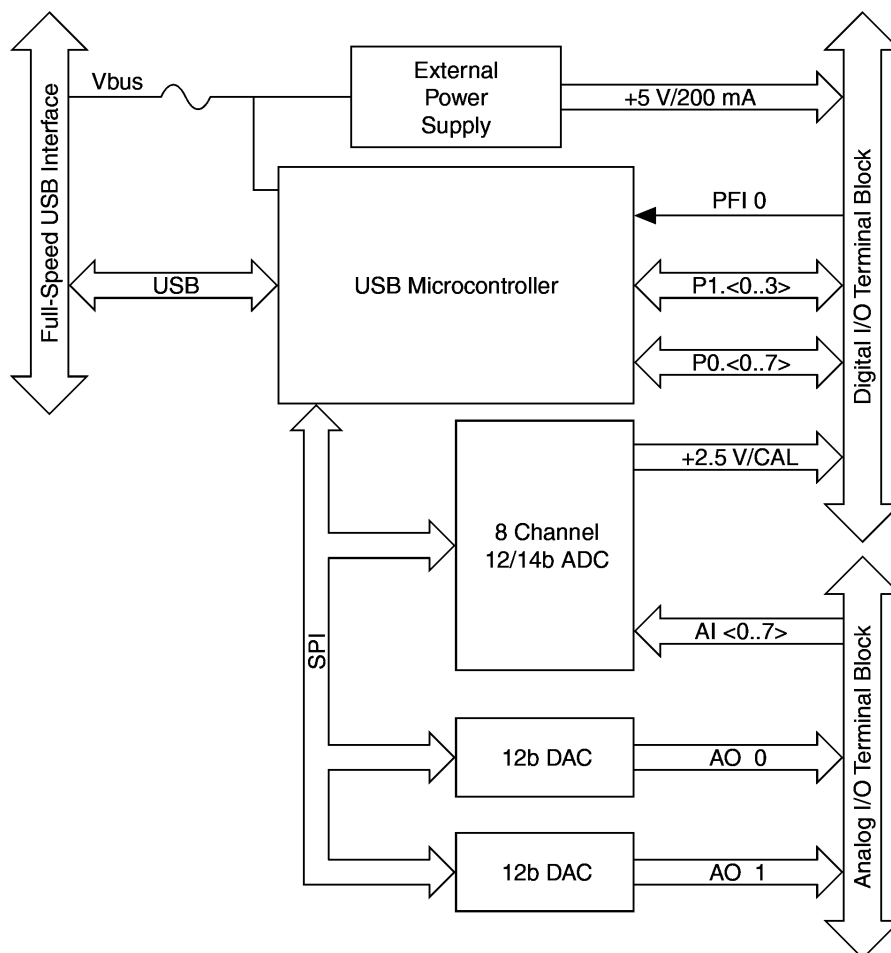
Dwa przetworniki cyfrowo-analogowe o rozdzielczości 12 bitów sterują wyjściami analogowymi AO <0, 1> i mogą dostarczyć napięcie w zakresie 0 .. +5V względem zacisku masy GND, z wydajnością prądową 5mA. Parametry tych przetworników zestawiono w tabelce 6.

Moduły USB-6008/6009 dostarczają również dwóch napięć, które mogą zasilać układy zewnętrzne. Wyjście 2,5V względem masy GND dostarcza wysokostabilnego napięcia referencyjnego o wydajności prądowej maksymalnie 1mA. Błąd tego napięcia wzorcowego nie przekracza 0,25% przy dryfcie temperaturowym nie większym od 50ppm/°C. Napięcie to można wykorzystać np. do kalibracji toru pomiarowego lub do zasilania czujników. Wyjście 5V względem masy może zasilać dowolne zewnętrzne układy prądem o wartości maksymalnie 200mA. Parametry wyjściowych napięć zasilających zestawiono w tabelce 7.

Wszystkie zaciski masy GND na obu złączach są ze sobą połączone i stanowią wspólny punkt odniesienia dla pomiarów napięć w trybie wejścia niesymetrycznego RSE, dla wyjść analogowych, dla wszystkich linii cyfrowych oraz dla napięcia referencyjnego 2,5V i napięcia zasilającego 5V. Rozmieszczenie poszczególnych sygnałów na stykach obu złączy przedstawia rysunek 3.



Rys.3. Złącza sygnałowe modułu NI USB-6008/6009



Rys.4. Schemat blokowy modułu NI USB-6008/6009

Budowę wewnętrzną modułu objaśnia schemat blokowy przedstawiony na rysunku 4. Pomiar napięć umożliwia jeden ośmiokanałowy przetwornik analogowo-cyfrowy ADC z sukcesywną aproksymacją o rozdzielczości 12 bitów (USB-6008) lub 14 bitów (USB-6009)

i maksymalną szybkością próbkowania odpowiednio 10kS/s lub 48kS/s. W trybie pomiarów różnicowych DIFF jedno wejście analogowe wykorzystuje dwa kanały przetwornika, co umożliwia pomiary napięć z czterech różnych źródeł. W trybie RSE każdy kanał umożliwia niezależny pomiar napięcia z ośmiu różnych źródeł.

Napięcia wyjściowe modułu są generowane przez dwa niezależne przetworniki cyfrowo-analogowe. Konfigurowalne linie wejść/wyjść cyfrowych dołączone są do mikrokontrolera sterującego pracą modułu. Wszystkie elementy składowe modułu są zasilane poprzez interfejs USB. Szczegółowe dane techniczne modułu dostępne są w dokumentacji [9].

Tab. 4. Parametry wejść-wyjść cyfrowych modułu USB-6008/6009

Digital logic levels	Min	Max
Input low voltage	-0.3 V	0.8 V
Input high voltage	2.0 V	5.8 V
Input leakage current	—	50 $\mu$ A
Output low voltage (I = 8.5 mA)	—	0.8 V
Output high voltage		
Active drive, I = -8.5 mA	2.0 V	3.5 V
Open collector, I = -0.6 mA, nominal	2.0 V	5.0 V
Open collector, I = -8.5 mA, with external pull-up resistor	2.0 V	—

Digital I/O lines

P0.<0..7>.....8 lines

P1.<0..3>.....4 lines

Direction control .....Each channel individually programmable  
as input or output

Output driver type<sup>1</sup>

NI USB-6008 .....Open collector

NI USB-6009 .....Each channel individually programmable as  
open collector or active drive

Compatibility .....TTL, LVTTTL, CMOS

Absolute maximum voltage range .....-0.5 to 5.8 V with respect to GND

Pull-up resistor .....4.7 k $\Omega$  to 5 V

Power-on state.....Input

<sup>1</sup> This document uses NI-DAQmx naming conventions.

Open-drain is called open collector and push-pull is called active drive.

Tab. 5. Parametry licznika impulsów modułu USB-6008/6009

Number of counters .....	1
Resolution .....	32 bits
Counter measurements.....	Edge counting (falling-edge)
Counter direction .....	Count up
Pull-up resistor.....	4.7 k $\Omega$ to 5 V
Maximum input frequency.....	5 MHz
Minimum high pulse width.....	100 ns
Minimum low pulse width .....	100 ns
Input high voltage .....	2.0 V
Input low voltage .....	0.8 V

Tab. 6. Parametry przetworników DAC modułu USB-6008/6009

Analog outputs .....	2
Output resolution .....	12 bits
Maximum update rate .....	150 Hz, software-timed
Output range .....	0 to +5 V
Output impedance .....	50 $\Omega$
Output current drive .....	5 mA
Power-on state .....	0 V
Slew rate.....	1 V/ $\mu$ s
Short circuit current .....	50 mA
Absolute accuracy (no load)	
Typical.....	7 mV
Maximum at full scale .....	36.4 mV

Tab. 7. Parametry wyjściowych napięć zasilających modułu USB-6008/6009

<b>External Voltage</b>	+5 V output (200 mA maximum)	
	Minimum .....	+4.85 V
	Typical.....	+5 V
	+2.5 V output (1 mA maximum) .....	+2.5 V
	+2.5 V accuracy .....	0.25% maximum
	Reference temperature drift .....	50 ppm/ $^{\circ}$ C maximum

## 1.2. System interfejsu USB

Moduł USB-6008/6009 jest wyposażony w interfejs standardu USB 2.0 full-speed. Szybkość transmisji danych w tym standardzie wynosi 12Mb/s. Oprócz transmisji danych interfejs USB dostarcza napięcia +5V niezbędnego do zasilania modułu. Parametry zastosowanego interfejsu zestawiono w tabelce 8.

Tab. 8. Parametry interfejsu USB modułu USB-6008/6009

<b>Bus Interface</b>	USB specification .....	USB 2.0 full-speed
	USB bus speed .....	12 Mb/s
<b>Power Requirements</b>	USB 4.10 to 5.25 VDC	
	Typical.....	80 mA
	Maximum.....	500 mA
	USB suspend	
	Typical.....	300 $\mu$ A
	Maximum.....	500 $\mu$ A

## 2. Programowanie modułu NI USB-6008/6009 w środowisku LabVIEW

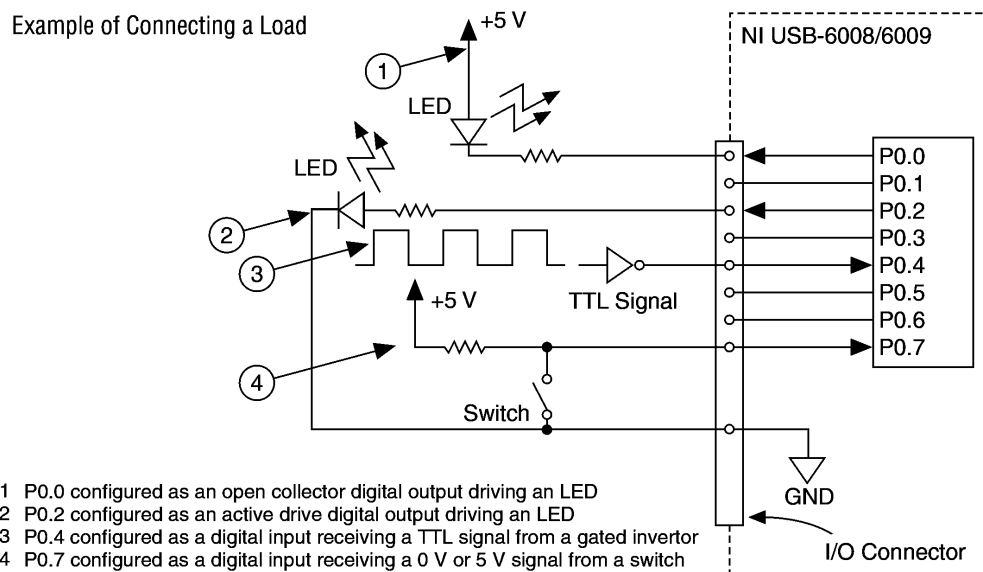
### 2.1. Instalowanie modułu NI USB-6008/6009 w systemie

Moduł akwizycji danych pomiarowych USB-6008/6009 wymaga wcześniejszego zainstalowania na komputerze oprogramowania NI-DAQmx w wersji 7.5 lub wyższej. Niezbędne jest również odpowiednie środowisko programistyczne do tworzenia aplikacji pomiarowych, jak np. LabVIEW lub inne. Gdy moduł USB-6008/6009 zostanie podłączony do komputera po raz pierwszy, automatycznie rozpoczyna się instalowanie dla niego oprogramowania. Po uruchomieniu się Kreatora znajdowania nowego sprzętu należy wybrać opcję "Zainstaluj oprogramowanie automatycznie" i kliknąć "Dalej". Gdy Kreator zainstaluje oprogramowanie, kończymy jego pracę klikając "Zakończ".

Instalacja oprogramowania jest wymagana tylko jeden raz podczas pierwszego dołączenia modułu do komputera. Przy kolejnym dołączeniu tego samego modułu do tego samego portu USB powinien on być poprawnie rozpoznany przez system.

### 2.2. Dołączanie sygnałów cyfrowych do modułu NI USB-6008/6009

Różne możliwe sposoby dołączania sygnałów cyfrowych do modułu USB-6008/6009 przedstawiono na rysunku 5. Wszystkie wejścia i wyjścia cyfrowe są zgodne ze standardem TTL, którego najważniejsze parametry podano w tabeli 4.



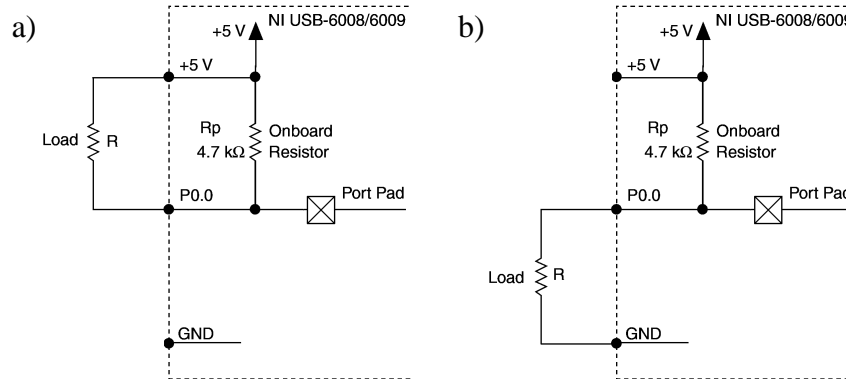
Rys. 5. Dołączanie sygnałów cyfrowych do modułu NI USB-6008/6009

Wszystkie linie cyfrowe modułu USB\_6008/6009 są dwukierunkowe: mogą pracować jako wejścia lub wyjścia. W tym celu należy je przed użyciem odpowiednio skonfigurować. Do linii skonfigurowanej jako wejście cyfrowe można dołączyć dowolne źródło sygnału cyfrowego TTL lub przycisk dołączony poprzez rezystor do napięcia +5V. Po włączeniu zasilania wszystkie linie automatycznie ustawiają się jako wejścia (*Power-on state*). Wszystkie linie cyfrowe są wyposażone w wewnętrzne rezystory  $4,7k\Omega$  podciągające do napięcia +5V (*Pull-up resistors*), które wymuszają stan wysoki na wejściu. Z tego względu po włączeniu zasilania na wszystkich liniach cyfrowych występuje stan wysoki, mimo że są one skonfigurowane jako wejścia. Jest to istotne, jeśli do linii cyfrowej dołączone jest obciążenie po stronie masy, gdyż wtedy po włączeniu zasilania będzie ono sterowane stanem wysokim.

Możliwe sposoby dołączenia obciążenia do linii cyfrowej skonfigurowanej jako wyjście przedstawiono na rysunku 6. Pokazano na nim również wewnętrzne rezystory podciągające. Bezpieczniejsze jest włączanie obciążenia po stronie napięcia zasilającego (rys. 6a), gdyż po

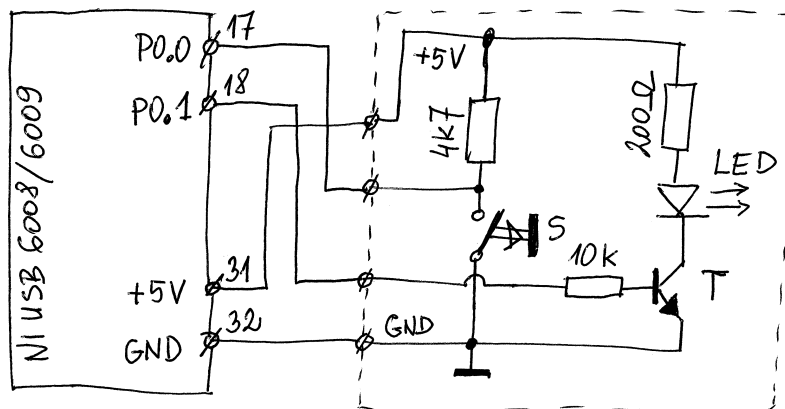


włączeniu zasilania linia ustawiona automatycznie jako wejście nie spowoduje przepływu prądu przez obciążenie. Po przeprogramowaniu linii jako wyjście można bezpiecznie sterować obciążeniem. Gdy obciążenie jest włączone po stronie masy (rys.6b), po włączeniu zasilania przez obciążenie popłynie prąd, zanim linia zostanie przeprogramowana jako wyjście i ustawiona w stan niski. Sytuacja taka może być niebezpieczna.



Rys. 6. Dołączanie obciążenia do wyjścia cyfrowego modułu NI USB-6008/6009: a) od strony napięcia zasilającego, b) od strony masy

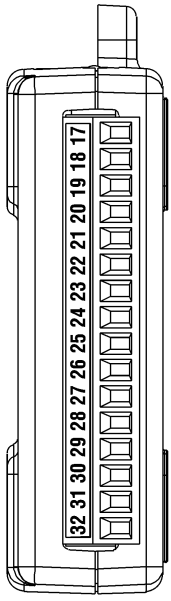
Na rysunku 7 przedstawiono układ współpracujący z liniami cyfrowymi wykorzystywany w ćwiczeniu. Przycisk S steruje linią P0.0 którą należy skonfigurować jako wejście. Linia P0.1 skonfigurowana jako wyjście steruje poprzez tranzystor T diodę LED. Stan wysoki na wyjściu P0.1 zapala diodę. Zastosowano włączenia obciążenia po stronie masy (rys.6b), co umożliwi zaobserwowanie opisanej powyżej niebezpiecznej sytuacji po włączeniu zasilania: dioda świeci się po włączeniu zasilania, gdy linia P0.1 jest automatycznie ustawiona jako wejście. W układach praktycznych należy unikać takiego połączenia, jeśli sterowany jest odbiornik mogący stwarzać zagrożenie, np. silnik elektryczny większej mocy.



Rys. 7. Schemat połączeń wykorzystywanego w ćwiczeniu układu współpracującego z wejściem i wyjściem cyfrowym modułu NI USB-6008/6009

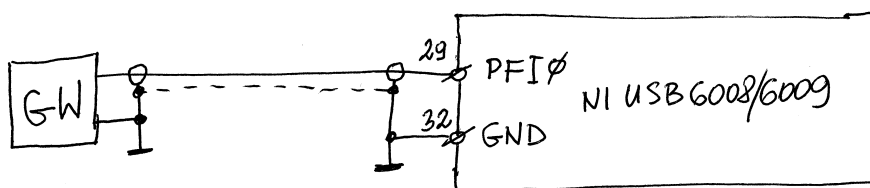
Rozmieszczenie sygnałów cyfrowych na złączu przedstawiono w tabeli 8. Łącznie dostępnych jest 12 linii cyfrowych, które mogą być dowolnie i niezależnie od siebie konfigurowane jako wejścia lub wyjścia. Linie cyfrowe zgrupowane są w dwa porty: P0 i P1. W porcie P0 dostępnych jest 8 linii oznaczonych od P0.0 do P0.7, a w porcie P1 dostępne są 4 linie oznaczone od P1.0 do P1.3. Cyfrowa linia PFI0 nie ma możliwości programowania kierunku, zawsze pracuje jako wejście i jest dołączona do 32 bitowego licznika impulsów. Wszystkie linie cyfrowe posiadają wspólną linię masy GND.

Tab. 8. Rozmieszczenie sygnałów cyfrowych na złączu modułu NI USB-6008/6009

Module	Terminal	Signal
	17	P0.0
	18	P0.1
	19	P0.2
	20	P0.3
	21	P0.4
	22	P0.5
	23	P0.6
	24	P0.7
	25	P1.0
	26	P1.1
	27	P1.2
	28	P1.3
	29	PFI 0
	30	+2.5 V
	31	+5 V
	32	GND

### 2.3. Dołączanie sygnału do wejścia licznika impulsów modułu NI USB-6008/6009

Linia PFI0 (*Programmable Function Interface*) zawsze pracuje tylko jako wejście cyfrowe i może pełnić rolę wejścia zliczającego impulsy dla licznika impulsów lub wejścia wyzwalającego pomiar przetwornikiem ADC. Pojemność licznika wynosi 32 bity i może on zliczać impulsy o parametrach zgodnych ze standardem TTL i częstotliwości do 5 MHz. Tak jak wszystkie pozostałe linie cyfrowe, również wejście licznika posiada wewnętrzny rezystor dołączony do napięcia zasilającego 5V, który wymusza stan wysoki i umożliwia bezpośrednią współpracę z pasywnymi stykami elektromechanicznymi. Na rysunku 9 przedstawiono schemat układu wykorzystywanego w ćwiczeniu: do wejścia PFI0 dołączony jest generator wzorcowy GW. Jego rolę może również spełniać częstotściomierz cyfrowy, który posiada dostępne wyjście częstotliwości wzorcowej  $f_w$ . Zazwyczaj odpowiednie gniazdko BNC umieszczone jest na tylnej ścianie obudowy częstotściomierza.

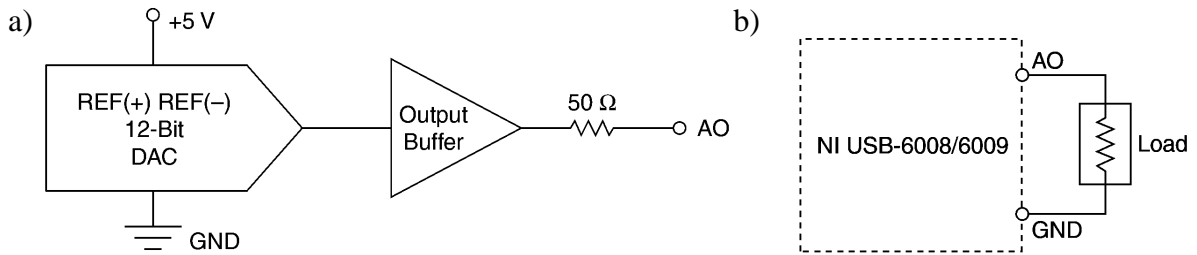


Rys. 9. Sposób dołączenia generatora wzorcowego do wejścia licznika impulsów modułu NI USB-6008/6009

### 2.4. Wykorzystanie przetwornika cyfrowo-analogowego modułu NI USB-6008/6009

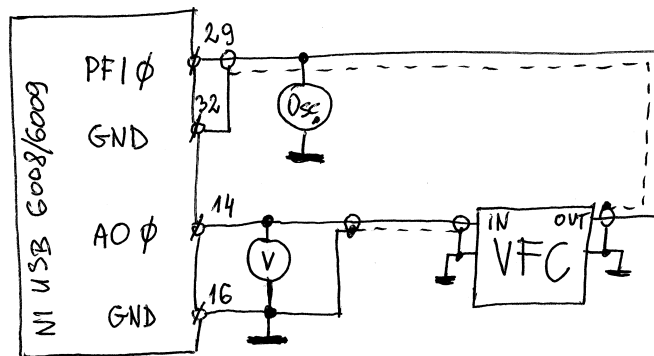
Moduł USB-6008/6009 wyposażony jest w dwa identyczne tory przetworników cyfrowo-analogowe, przedstawione na rysunku 10a. Na wyjściach analogowych można uzyskać napięcia z zakresu od 0 do +5V. Obciążenie należy dołączyć pomiędzy wyjście przetwornika i masę, zgodnie z rysunkiem 10b. Dopuszczalne obciążenie prądowe przetwornika wynosi

tylko 5mA, więc zazwyczaj konieczne jest zastosowanie dodatkowego wzmacniacza, gdy potrzebne jest sterowanie obciążeniem większej mocy.



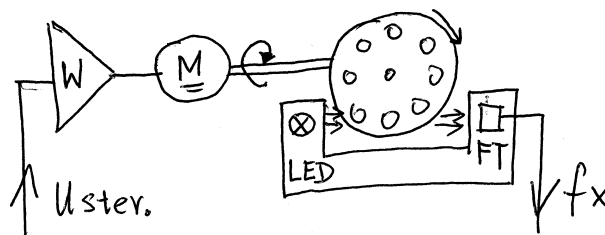
Rys.10. Schemat blokowy toru przetwornika DAC modułu NI USB-6008/6009 (a) i sposób dołączania obciążenia do wyjścia przetwornika (b)

Na rysunku 11 przedstawiono schemat układu wykorzystywanego w ćwiczeniu. Wyjście analogowe AO0 jest dołączone do wejścia przetwornika napięcie-częstotliwość VFC. Sygnał wyjściowy przetwornika VFC dołączony jest do wejścia PFIO, co umożliwia zliczanie impulsów i pomiar częstotliwości.



Rys.11. Schemat układu wykorzystującego przetwornik DAC modułu NI USB-6008/6009 do sterowania przetwornikiem napięcie-częstotliwość VFC

Zamiast przetwornika VFC można również do układu dołączyć silnik elektryczny z przetwornikiem obrotowo-impulsowym. Odpowiedni schemat blokowy przedstawiono na rysunku 12. Silnik elektryczny M małej mocy sterowany jest z wyjścia analogowego poprzez wzmacniacz W, zapewniający odpowiednią wydajność prądową. Na wale silnika umieszczona jest tarcza z otworami. Dioda świecą LED oświetla fototranzystor FT poprzez wirującą tarczę z otworami. Częstotliwość impulsów wyjściowych jest więc proporcjonalna do prędkości obrotowej silnika. Sygnał wyjściowy z fototranzystora dołączony jest do wejścia PFIO, co umożliwia zliczanie impulsów i pomiar prędkości obrotowej silnika dla różnych napięć zasilających.

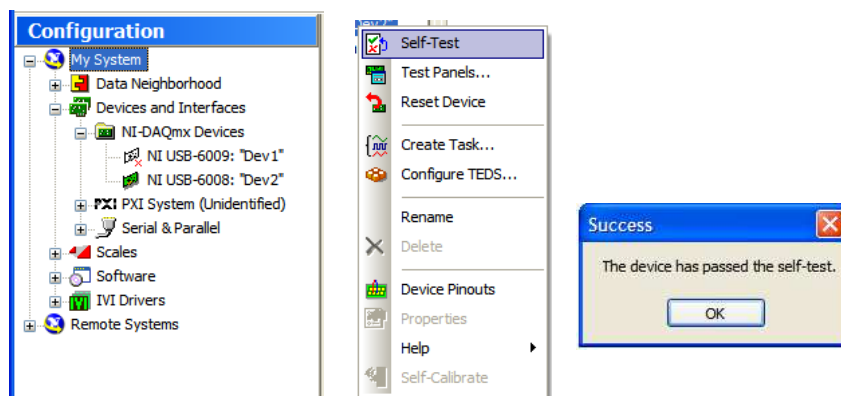


Rys.12. Schemat blokowy układu wykorzystującego silnik prądu stałego z przetwornikiem obrotowo-impulsowym włączanego w miejsce przetwornika napięcie-częstotliwość VFC

## 2.5. Testowanie modułu NI USB-6008/6009

Po dołączeniu modułu do komputera i poprawnym zainstalowaniu go w systemie oraz dołączeniu do niego odpowiednich źródeł sygnałów lub układów wykonawczych można przeprowadzić test jego działania. W tym celu należy uruchomić program Measurement & Automation Explorer klikając jego ikonę na pulpicie lub uruchamiając go z menu: Start\Wszystkie programy\National Instruments\Measurement & Automation. Po uruchomieniu programu, w okienku "Configuration" rozwijamy menu "My System\Devices and Interfaces\NI-DAQmx Devices" i sprawdzamy, czy zainstalowany moduł jest wykazywany jako aktywny (zielona ikonka). Jeśli na wykazie nie występuje aktywne urządzenie, to odłączamy od modułu wszystkie źródła sygnałów, odłączamy moduł od portu USB i zamykamy wszystkie programy. Po chwili ponownie dołączamy moduł do portu USB i powtarzamy procedurę instalacji od początku.

Jeśli urządzenie jest poprawnie zainstalowane, wybieramy je z listy w okienku "Configuration" i odczytujemy oraz notujemy Serial Number naszego modułu oraz przydzielony do niego numer urządzenia (przykładowo na rys. 13 jest to "Dev2"). Prawym kliknięciem na nazwie naszego modułu otwieramy listę zadań i wybieramy "Self-Test". Po wykonaniu testu otrzymamy informację o stanie urządzenia. Kończymy test klikając "OK". Procedurę Self-Testu ilustruje rysunek 13. Widoczna ikonka urządzenia "Dev1" jest nieaktywna i wskazuje moduł dołączony kiedyś wcześniej do komputera, ale nieobecny w systemie w chwili obecnej.



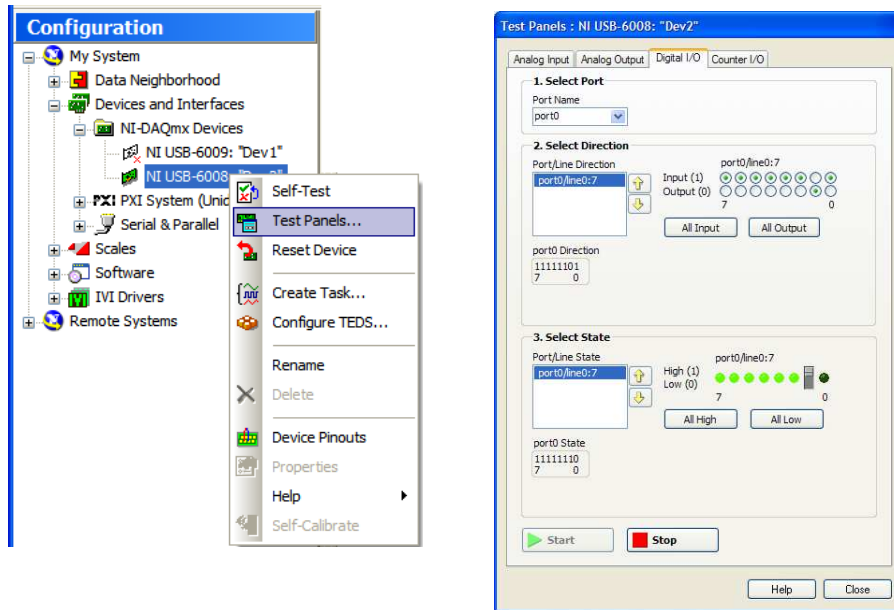
Rys. 13. Przeprowadzanie Self-Testu w programie Measurement & Automation Explorer

Po przeprowadzeniu Self-Testu przeprowadzamy test poszczególnych funkcji modułu wykorzystywanych w ćwiczeniu: test cyfrowych linii wejścia-wyjścia, test licznika impulsów, oraz test przetwornika cyfrowo-analogowego.

### 2.5.1. Test cyfrowych linii wejścia-wyjścia

Test cyfrowych linii wejścia-wyjścia przeprowadzamy z dołączonymi układami wyjściowymi i wejściowymi zgodnie ze schematem na rysunku 7. Po sprawdzeniu układu połączeń, w otwartym oknie programu Measurement & Automation Explorer, ponownie prawym kliknięciem na nazwie naszego modułu rozwijamy listę zadań i wybieramy "Test Panels ...". W nagłówku otwartego okna "Test Panels" sprawdzamy nazwę urządzenia. Następnie wybieramy zakładkę "Digital I/O" i przeprowadzamy konfigurację linii cyfrowych, zgodnie z rysunkiem 14. W okienku 1. Select Port wybieramy port0, w okienku 2. Select Direction Ustawiamy linię P0.1 jako wyjście (Output), pozostałe linie pozostawiamy jako wejścia (Input). Uruchamiamy test przyciskiem "START". W okienku 3. Select State sprawdzamy, czy po wciśnięciu przycisku S dołączonego do linii P0.0 sygnalizowany jest stan niski na pozycji port0/line0. Przełącznikiem na pozycji port0/line1 sprawdzamy, czy przy ustawionym stanie wysokim zapala się dioda LED dołączona do wejścia P0.1. Przykładowo,

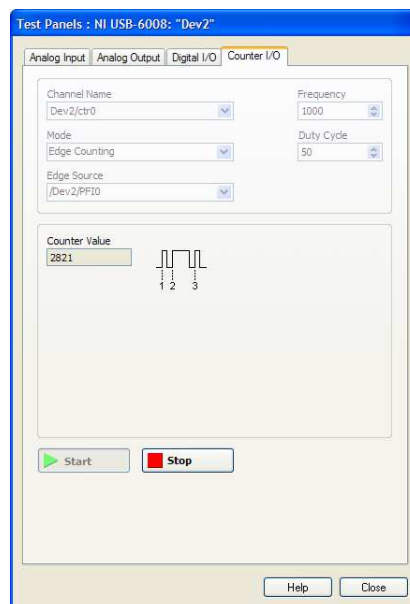
na rysunku 14 pokazana jest sytuacja, gdy wciśnięto przycisk S i zapalona jest dioda LED. Po uzyskaniu pozytywnych wyników kończymy test przyciskiem "STOP".



Rys.14. Wyniki testu linii cyfrowych w programie Measurement & Automation Explorer

### 2.5.2. Test licznika impulsów

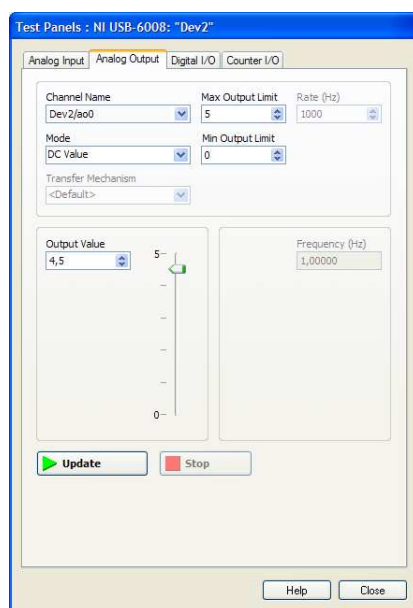
Test licznika impulsów przeprowadzamy z dołączonym do wejścia PF10 generatorem częstotliwości wzorcowej zgodnie ze schematem na rysunku 9. Po sprawdzeniu układu połączeń, w otwartym oknie programu Measurement & Automation Explorer wybieramy zakładkę "Counter I/O" i uruchamiamy test przyciskiem "START". W okienku Counter Value sprawdzamy, czy liczba impulsów zwiększa się. Zmieniamy częstotliwość wzorcową i sprawdzamy, czy dla większej/mniejszej częstotliwości zmiany zawartości licznika zachodzą szybciej/wolniej. Przykładowy wynik testowania licznika impulsów przedstawiono na rysunku 15. Po uzyskaniu pozytywnych wyników kończymy test przyciskiem "STOP".



Rys.15. Wyniki testu licznika impulsów w programie Measurement & Automation Explorer

### 2.5.3. Test przetwornika cyfrowo-analogowego

Test przetwornika cyfrowo-analogowego przeprowadzamy z dołączonym do wyjścia AO0 woltomierzem zgodnie ze schematem na rysunku 11. Przetwornika napięcie-częstotliwość VFC podczas tego testu nie trzeba dołączać do układu. Po sprawdzeniu układu połączeń, w otwartym oknie programu Measurement & Automation Explorer wybieramy zakładkę "Analog Output". W okienku Output Value wpisujemy kilka wartości z przedziału od 0 do +5V i po wciśnięciu przycisku Update sprawdzamy wskazanie woltomierza. Podobnie możemy przeprowadzić test za pomocą suwaka. Przykładowy wygląd Panelu podczas testu przetwornika A/C przedstawiono na rysunku 16.



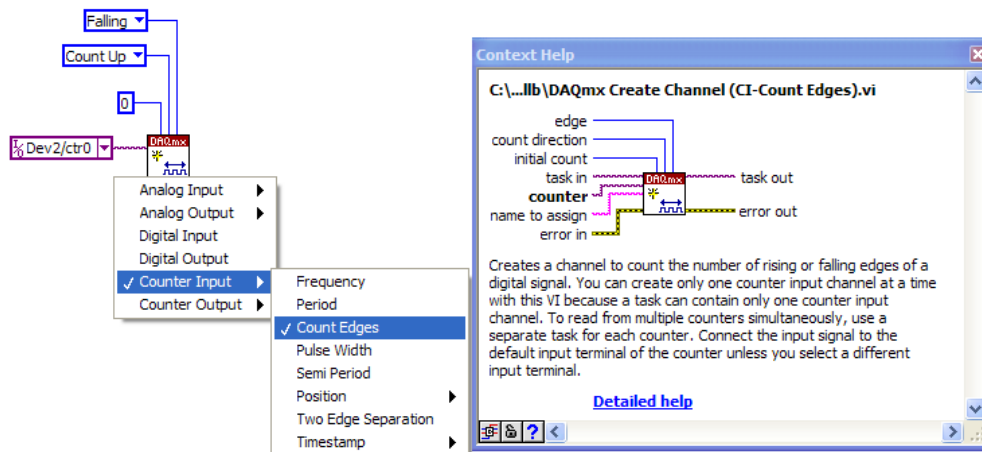
Rys.16. Wyniki testu przetwornika C/A w programie Measurement & Automation Explorer

Po uzyskaniu pozytywnych wyników wszystkich testów zamykamy okno "Test Panels" i kończymy działanie programu Measurement & Automation Explorer.

### 2.6. Odczytywanie stanu licznika impulsów modułu NI USB-6008/6009 w LabVIEW

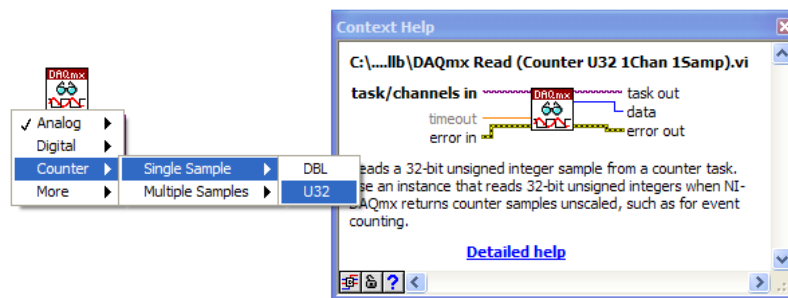
Aby w środowisku LabVIEW zrealizować odczyt zawartości licznika impulsów modułu NI USB-6008/6009 należy przygotować odpowiedni program. W pierwszej kolejności należy utworzyć wirtualny kanał pomiarowy (*Virtual Channel*), skonfigurować go przypisując mu odpowiedni kanał fizyczny modułu pomiarowego powiązany z konkretnymi zaciskami na złączu (*physical channel*) oraz definiując parametry: rodzaj zbocza, kierunek zliczania, wartość początkowa itp. Następnie należy uruchomić zadanie (*Task*) dla przyrządu wirtualnego, który będzie reprezentował ten wirtualny kanał na diagramie.

Kolejność postępowania jest następująca. Z menu Functions/Measurement I/O/NI-DAQmx wybieram obiekt Create Channel, stawiamy go na pustym diagramie i z rozwijanego pod nim menu wybieramy opcję zliczania zboczy Counter Input/Count Edges. Pomagając sobie oknem pomocy Context Help, za pomocą prawego kliknięcia wstawiam stałe (Create/Constant) na wejściach: edge/Falling, count direction/Count Up, initial count/0. Na wejściu counter tworząc stałą wybieramy opcję browse i z listy dostępnych urządzeń wybieramy aktualnie wykorzystywany moduł. Tworzenie wirtualnego kanału przedstawiono na rysunku 17.



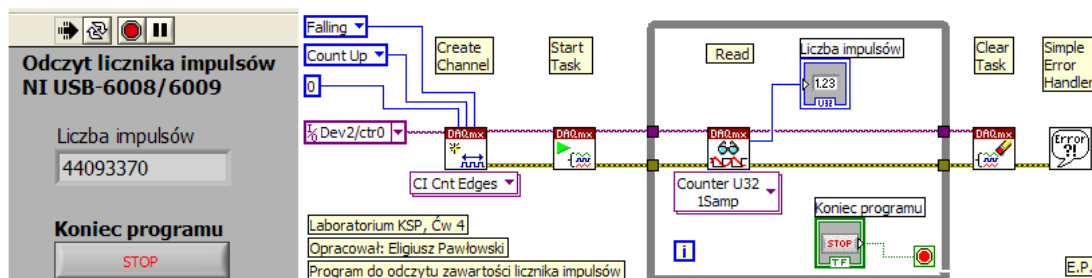
Rys. 17. Konfigurowanie kanału fizycznego - Create New Channel

Następnie stawiamy na diagramie obiekty: Start Task, Read, Clear Task oraz Simple Error Handler z menu Dialog&User Interface. Obiekt Start Task nie wymaga żadnej konfiguracji. Obiekt Read konfigurujemy następująco: Counter/Single Sample/U32, dzięki czemu będziemy odczytywać jednorazowo 32-bitową zawartość licznika. Konfigurowanie obiektu Read przedstawiono na rysunku 18.





Rys. 18. Konfigurowanie odczytu kanału - Read

Na wyjściu Data obiektu Read tworzymy wskaźnik (create/Indicator), wstawiamy na diagramie pętlę While, dodajemy przycisk STOP i łączymy diagram w sposób przedstawiony na rysunku 19.



Rys. 19. Panel i diagram programu do odczytu zawartości licznika

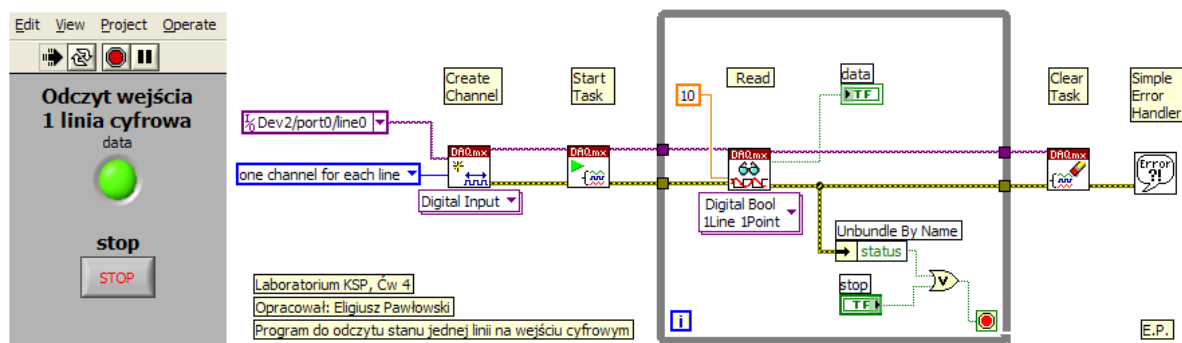
Otrzymaliśmy najprostszy program realizujący zliczanie impulsów za pomocą 32-bitowego licznika modułu NI USB-6008/6009. Uruchamiamy go przyciskiem , co sygnalizowane jest zmianą postaci przycisku na . Ponieważ program zawiera w swojej strukturze pętlę While, będzie się on wykonywał bez przerwy, aż do naciśnięcia przycisku STOP. Kolejne odczyty licznika powinny zwiększać się z szybkością proporcjonalną do częstotliwości wzorcowej podanej na wejście licznika. Każdorazowe uruchomienie licznika





rozpoczyna zliczanie impulsów od zera. Po sprawdzeniu działania programu zapisujemy go na dysku we wskazanym w instrukcji katalogu.

## 2.7. Odczytywanie stanów logicznych z wejść cyfrowych modułu NI USB-6008/6009

Na rysunku 20 przedstawiono Panel i Diagram programu do odczytu stanów logicznych na wejściu cyfrowym modułu USB\_6008/6009. Struktura programu jest podobna do diagramu odczytu licznika z poprzedniego punktu. Należy utworzyć wirtualny kanał, uruchomić zadanie, odczytać wirtualny kanał, zamknąć zadanie i obsłużyć ewentualne błędy. Konfigurację obiektów Create Channel i Read przeprowadzamy w rozwijanym menu, podobnie jak w poprzednim punkcie, wybierając opcje odpowiednie do realizowanego zadania. Po wybraniu z menu Functions/Measurement I/O/NI-DAQmx obiektu Create Channel, stawiamy go na pustym diagramie i konfigurujemy go podobnie jak na rysunku 17, tym razem jednak z rozwijanego pod nim menu wybieramy opcję wejścia cyfrowego Digital Input. Pomagając sobie oknem pomocy Context Help, za pomocą prawego kliknięcia wstawiamy stałą (Create/Constant) na wejściu line grouping o wartości: one channel for each line. Na wejściu lines tworząc stałą wybieramy opcję browse i z listy dostępnych urządzeń wybieramy aktualnie wykorzystywany moduł, a w nim port0 i linię 0, do której dołączony jest przycisk według rysunku 7 (np. na rysunku 20 wybrano opcję Dev2/port0/line0). Następnie stawiamy na diagramie obiekty: Start Task, Read, Clear Task oraz Simple Error Handler z menu Dialog&User Interface. Obiekt Start Task nie wymaga żadnej konfiguracji. Obiekt Read konfigurujemy podobnie jak na rysunku 18, tym razem jednak z rozwijanego pod nim menu wybieramy opcję: Digital/Single Channel/Single Sample/Boolean (1 linie), dzięki czemu będziemy odczytywać jednorazowo stan jednej linii, który zostanie przypisany do 1-bitowej zmiennej logicznej typu Boolean. Na wejściu timeout modułu Read tworzymy stałą definiującą czas oczekiwania, po którym zostanie zgłoszony błąd. Dodatkowy obiekt Unbundle By Name umożliwia wcześniejsze zakończenie programu przy pojawieniu się błędu obsługi. Na wyjściu Data obiektu Read tworzymy wskaźnik (create/Indicator), wstawiamy na diagramie pętlę While, dodajemy przycisk STOP i łączymy diagram w sposób przedstawiony na rysunku 20.



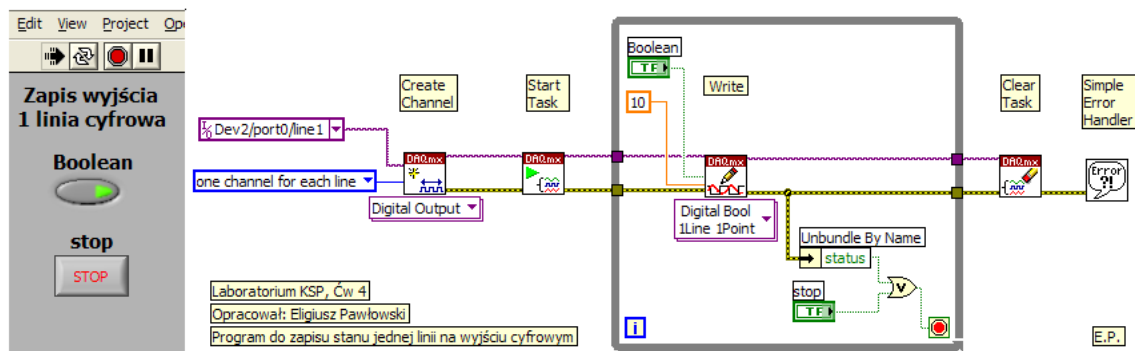
Rys. 20. Panel i diagram programu do odczytu stanów logicznych na wejściu cyfrowym

Otrzymaliśmy najprostszy program realizujący odczyt stanu logicznego na jednym, wybranym wejściu cyfrowym modułu NI USB-6008/6009. Uruchamiamy go przyciskiem , co sygnalizowane jest zmianą postaci przycisku na . Ponieważ program zawiera w swojej strukturze pętlę While, będzie się on wykonywał bez przerwy, aż do naciśnięcia przycisku STOP. Ponieważ zgodnie ze schematem na rysunku 7 wciśnięcie przycisku wymusza niski stan logiczny, to po uruchomieniu programu wskaźnik na panelu programu powinien sygnalizować stan wysoki, a po naciśnięciu przycisku - stan niski. Po sprawdzeniu działania programu zapisujemy go na dysku we wskazanym w instrukcji katalogu.





## 2.8. Zapisywanie stanów logicznych do wyjść cyfrowych modułu NI USB-6008/6009

Na rysunku 21 przedstawiono Panel i Diagram programu do ustawiania stanów logicznych na wyjściu cyfrowym modułu USB\_6008/6009. Struktura programu jest podobna do diagramu odczytu stanu wejścia cyfrowego z poprzedniego punktu. Należy utworzyć wirtualny kanał, uruchomić zadanie, zapisać odpowiednią wartość logiczną do wirtualnego kanału, zamknąć zadanie i obsłużyć ewentualne błędy. Konfigurację obiektów Create Channel i Write przeprowadzamy w rozwijanym menu, podobnie jak w poprzednim punkcie, wybierając opcje odpowiednie do realizowanego zadania. Po wybraniu z menu Functions/Measurement I/O/NI-DAQmx obiektu Create Channel, stawiamy go na pustym diagramie i konfigurujemy go podobnie jak na rysunku 17, tym razem jednak z rozwijanego pod nim menu wybieramy opcję wyjścia cyfrowego Digital Output. Pomagając sobie oknem pomocy Context Help, za pomocą prawego kliknięcia wstawiamy stałą (Create/Constant) na wejściu line grouping o wartości: one channel for each line. Na wejściu lines tworząc stałą wybieramy opcję browse i z listy dostępnych urządzeń wybieramy aktualnie wykorzystywany moduł, a w nim port0 i linię 1, do której dołączona jest dioda LED według rysunku 7 (np. na rysunku 21 wybrano opcję Dev2/port0/line1). Następnie stawiamy na diagramie obiekty: Start Task, Write, Clear Task oraz Simple Error Handler z menu Dialog&User Interface. Obiekt Start Task nie wymaga żadnej konfiguracji. Obiekt Write konfigurujemy podobnie jak na rysunku 18, tym razem jednak z rozwijanego pod nim menu wybieramy opcję: Digital/Single Channel/Single Sample/Boolean (1 linie), dzięki czemu będziemy zapisywać jednorazowo wartość 1-bitowej zmiennej logicznej typu Boolean do jednej linii cyfrowego wyjścia. Na wejściu timeout modułu Write tworzymy stałą definiującą czas oczekiwania, po którym zostanie zgłoszony błąd. Dodatkowy obiekt Unbundle By Name umożliwia wcześniejsze zakończenie programu przy pojawieniu się błędu obsługi. Na Panelu programu stawiamy zadajnik w postaci przycisku i na diagramie dołączamy go do wejścia Data obiektu Write. Wstawiamy na diagramie pętlę While, dodajemy przycisk STOP i łączymy diagram w sposób przedstawiony na rysunku 21.



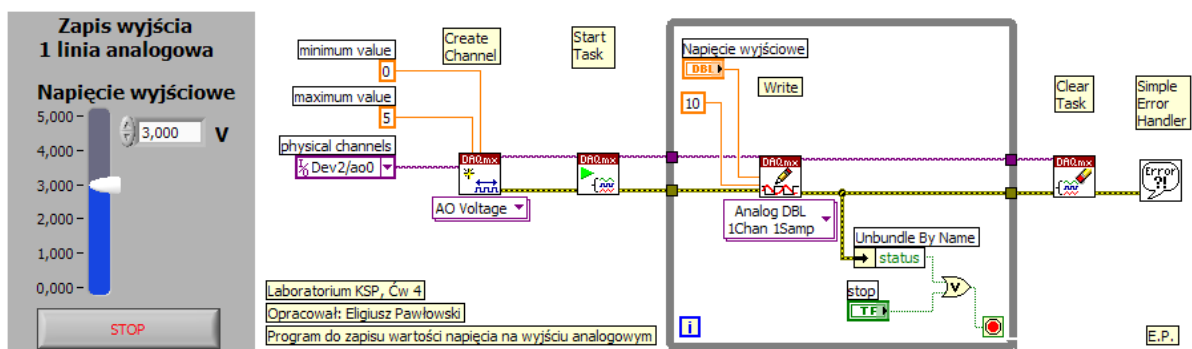
Rys. 21. Panel i diagram programu do ustawiania stanów logicznych na wyjściu cyfrowym

Otrzymaliśmy najprostszy program realizujący zapis stanu logicznego do jednej, wybranej linii na wyjściu cyfrowym modułu NI USB-6008/6009. Uruchamiamy go przyciskiem , co sygnalizowane jest zmianą postaci przycisku na . Ponieważ program zawiera w swojej strukturze pętlę While, będzie się on wykonywał bez przerwy, aż do naciśnięcia przycisku STOP. Należy zwrócić uwagę, że bezpośrednio po włączeniu komputera wszystkie linie cyfrowe są automatycznie ustawiane jako wejścia, które dołączone są poprzez rezystory do napięcia +5V (rys. 6), co powoduje od razu zapalenie diody LED podłączonej według rysunku 7. Po uruchomieniu programu linia P0.1 programowana jest jako wyjście, a ponieważ początkową wartością przycisku na panelu programu jest stan niski, to dioda LED zgaśnie. Ponieważ wciśnięcie przycisku na panelu programu oznacza wysoki stan logiczny, to po jego



naciśnięciu dioda LED powinna się zapalać. Po sprawdzeniu działania programu zapisujemy go na dysku we wskazanym w instrukcji katalogu.

## 2.9. Programowanie przetwornika cyfrowo-analogowego modułu NI USB-6008/6009

Na rysunku 22 przedstawiono Panel i Diagram programu do programowania przetwornika cyfrowo-analogowego modułu USB\_6008/6009. Struktura programu jest podobna do diagramu zapisu stanu logicznego do wyjścia cyfrowego z poprzedniego punktu. Należy utworzyć wirtualny kanał, uruchomić zadanie, zapisać odpowiednią wartość analogową do wirtualnego kanału, zamknąć zadanie i obsłużyć ewentualne błędy. Konfigurację obiektów Create Channel i Write przeprowadzamy w rozwijanym menu, podobnie jak w poprzednim punkcie, wybierając opcje odpowiednie do realizowanego zadania. Po wybraniu z menu Functions/Measurement I/O/NI-DAQmx obiektu Create Channel, stawiamy go na pustym diagramie i konfigurujemy go podobnie jak na rysunku 17, tym razem jednak z rozwijanego pod nim menu wybieramy opcję wyjścia analogowego Analog Output/Voltage. Pomagając sobie oknem pomocy Context Help, za pomocą prawego kliknięcia wstawiamy stałą (Create/Constant) na wejściach: minimum value o wartości 0V, maximum value o wartości 5V. Na wejściu physical channels tworząc stałą wybieramy opcję browse i z listy dostępnych urządzeń wybieramy aktualnie wykorzystywany moduł, a w nim wyjście analogowe zero (ao0), do którego dołączony jest woltomierz według rysunku 11 (np. na rysunku 22 wybrano opcję Dev2/ao0). Następnie stawiamy na diagramie obiekty: Start Task, Write, Clear Task oraz Simple Error Handler z menu Dialog&User Interface. Obiekt Start Task nie wymaga żadnej konfiguracji. Obiekt Write konfigurujemy podobnie jak na rysunku 18, tym razem jednak z rozwijanego pod nim menu wybieramy opcję: Analog/Single Channel/Single Sample/DBL, dzięki czemu będziemy zapisywać jednorazowo wartość zmiennej o podwójnej precyzji (double precision DBL) do jednej linii analogowego wyjścia. Na wejściu timeout modułu Write tworzymy stałą definiującą czas oczekiwania, po którym zostanie zgłoszony błąd. Dodatkowy obiekt Unbundle By Name umożliwia wcześniejsze zakończenie programu przy pojawieniu się błędu obsługi. Na Panelu programu stawiamy zadajnik w postaci suwaka (controls/numeric/Vertical Pointer Slide) lub dowolny inny o typie zmiennej DBL (double precision). Na diagramie dołączamy wyjście zadajnika do wejścia Data obiektu Write. Wstawiamy na diagramie pętlę While, dodajemy przycisk STOP i łączymy diagram w sposób przedstawiony na rysunku 22.



Rys. 22. Panel i diagram programu do programowania przetwornika cyfrowo-analogowego

Otrzymaliśmy najprostszy program realizujący zapis wartości liczbowej do jednej, wybranej linii na wyjściu analogowym modułu NI USB-6008/6009. Uruchamiamy go przyciskiem , co sygnalizowane jest zmianą postaci przycisku na . Ponieważ program zawiera w swojej strukturze pętlę While, będzie się on wykonywał bez przerwy, aż do naciśnięcia przycisku STOP. Należy zwrócić uwagę, że bezpośrednio po włączeniu komputera wszystkie linie analogowe są automatycznie ustawiane na wartość zero. Po

uruchomieniu programu na linii A00 będzie występowało napięcie o wartości ustawionej za pomocą zadajnika na panelu programu, co można sprawdzić woltomierzem dołączonym zgodnie z rysunkiem 11. Po sprawdzeniu działania programu zapisujemy go na dysku we wskazanym w instrukcji katalogu.

### 3. Ogólne zasady realizacji ćwiczenia

Ćwiczenie opiera się przede wszystkim na wykorzystaniu środowiska LabVIEW oraz modułu akwizycji danych pomiarowych NI USB-6008/6009. Dokumentacja techniczna modułu akwizycji danych pomiarowych stanowi załącznik do instrukcji. Studenci samodzielnie powinni zaopatrzyć się w literaturę wspomagającą naukę programowania w LabVIEW. Przykładowe pozycje, w tym dostępne w Internecie darmowe podręczniki, zamieszczono w wykazie literatury.

#### 3.1. Organizacja systemu plików na komputerze

Przystępując do realizacji ćwiczenia należy najpierw uruchomić komputer, a dopiero w drugiej kolejności włączyć zasilanie pozostałej aparatury, jako ostatni element zawsze załączamy źródło sygnału. Wyłączamy aparaturę w kolejności odwrotnej.

Wszystkie pliki z programami napisanymi podczas realizacji ćwiczenia należy zapisywać w katalogu C:\LABORKA\_KSP\STUDENT\RRRRMMDD\_G\_Z, gdzie RRRR, MM, DD oznaczają kolejno: rok, miesiąc i dzień rozpoczęcia wykonywania ćwiczenia, a G i Z oznaczają grupę i zespół laboratoryjny. Zapisywane pliki powinny posiadać nazwę utworzoną z początkowych liter nazwiska jednego z członków zespołu oraz kolejnych cyfr 1, 2, 3 itd., np.: **KOWALS\_1.Vi**, **KOWALS\_2.Vi** itd. Pliki utworzonych w czasie zajęć przyrządów wirtualnych do obsługi pojedynczych modułów powinny zawierać w pliku nazwie programu również symbol obsługiwane modułu.

Podczas realizacji kolejnego zadania często należy wykorzystać program opracowany w ramach wcześniejszego punktu. Nie wolno jednak w tym celu modyfikować programu będącego finalną wersją wcześniej zrealizowanego zadania. Po zapisaniu finalnej wersji programu realizującego w pełni określony punkt ćwiczenia, należy wykonać jego kopię nadając mu nową nazwę, zmieniając końcową cyfrę na kolejną wartość. Kopię programu należy wykonywać opcją File\Save As, wybierając następnie opcję Copy - create copy on disk\Substitute copy for original (Copy will be in memory. Original will be closed). Pracując na tak utworzonej kopii programu należy ją uzupełniać o nowe elementy, aż do zrealizowania kolejnego punktu ćwiczenia.

Wszystkie realizowane podczas zajęć programy powinny zawierać dane osobowe zespołu laboratoryjnego, numer grupy dziekańskiej oraz datę wykonywania ćwiczenia. Informacje te należy podać w postaci **komentarza umieszczonego na panelu sterującym i na diagramie**.


W każdym realizowanym punkcie oczekiwana jest inwencja własna studentów. Wydruki programów oraz przykładowe pliki udostępniane ćwiczącym należy traktować wyłącznie jako przykłady, a nie jako obowiązujące wzorce. Przykłady te zazwyczaj realizują zadania stawiane w ćwiczeniach tylko w ograniczonym zakresie, **mniej niż wymagane**.








#### 3.2. Podstawy programowania w środowisku LabVIEW

LabView jest graficznym środowiskiem programistycznym przeznaczonym do tworzenia programów zorientowanych na obsługę systemów pomiarowych. Aby napisać program w środowisku LabView należy:

- a- uruchomić środowisko LabView,
- b- utworzyć nowy plik programu: New\Blank\_VI,
- c- rozmieścić potrzebne kontrolki (*controls*) w oknie panelu programu,
- d- rozmieścić potrzebne elementy funkcyjne (*functions*) w oknie diagramu programu,

- e- wykonać odpowiednie połączenia na diagramie realizujące algorytm programu,
- f- uruchomić program, ocenić poprawność działania i wyszukać błędy,
- g- usunąć błędy w programie modyfikując zawartość panelu i diagramu programu,
- h- powtarzać punkty f oraz g aż do osiągnięcia oczekiwanego rezultatu.

Każda aplikacja przygotowana w środowisku LabView składa się z dwóch części: Panelu i Diagramu. Panel stanowi graficzny interfejs użytkownika aplikacji, natomiast Diagram jest graficznym zapisem algorytmu realizowanego przez tę aplikację. Po otwarciu aplikacji w środowisku LabView widoczny jest jej Panel sterujący. Przełączanie pomiędzy widokiem Panelu i Diagramu jest możliwe za pomocą kombinacji klawiszy **CTRL+E**. Kombinacją klawiszy **CTRL+T** możemy wybrać jednoczesny widok okna Panelu i Diagramu. Analizę Diagramu programu można sobie znacznie ułatwić włączając przyciskiem  okno pomocy kontekstowej **Context Help**.

Do uruchomienia programu służą przyciski  i . Proste programy nie posiadające w swej strukturze pętli programowych należy zazwyczaj uruchamiać przyciskiem , dzięki czemu pracują one w sposób ciągły i można je zatrzymać przyciskiem . Programy bardziej złożone posiadające w swej strukturze pętle programowe należy uruchamiać zazwyczaj przyciskiem , a do ich zatrzymywania służy odpowiedni przycisk sterujący w tej aplikacji. Uruchomienie programu sygnalizowane jest zmianą postaci przycisków na  i .

### 3.3. Wydruk dokumentacji programu

Środowisko LabVIEW pozwala w prosty sposób utworzyć na dysku twardym komputera pliki z dokumentacją zrealizowanego przyrządu wirtualnego. Pliki będą zawierać obraz Panelu oraz Diagramu. Kolejność postępowania jest następująca:

- wybrać opcję **File/Print..** i w oknie **Select VI(s)** zaznaczyć nazwę aplikacji do wydruku, wcisnąć **NEXT**,
- w oknie **Print Contents** zaznaczyć opcję **VI documentation**, wcisnąć **NEXT**,
- w oknie **VI Documentation** zaznaczyć opcje: **Front Panel**, **Controls (connected Controls)**, **Descriptions**, **Data type information**, **Label**, **Block Diagram**, wcisnąć **NEXT**,
- w oknie **Destination** wybrać opcję **HTML File**, wcisnąć **NEXT**,
- w oknie **HTML** wybieramy **Image format: GIF (uncompressed)**, **color depth: 256 colors**, wcisnąć **SAVE**,
- w oknie **SAVE** wybrać katalog (jeśli go jeszcze nie ma, to należy go utworzyć zgodnie z podaną wcześniej regułą). Zapisać plik.
- Odszukać zapisane pliki na dysku i sprawdzić ich zawartość.

**Zanotować w protokole nazwę utworzonego katalogu i nazwy zapisanych w nim plików z opisem zawartości.**

### 3.4. Zapisywanie wykresów do plików graficznych

Aby zapisać uzyskany w programie przebieg sygnału do pliku dyskowego, należy wykonać kolejno następujące czynności:

- podczas działania programu ustawić kursor myszki na oknie przebiegu sygnału i kliknąć prawym przyciskiem myszki,
- z otworzonego menu wybrać opcję **Export Simplified Image**, zaznaczyć opcję **Bitmap (BMP)** i **Save to file**,
- wybrać katalog utworzony przez grupę laboratoryjną na początku zajęć i wpisać nazwę pliku odpowiednio do zawartości, zatwierdzić **OK** i zapisać **Save**,

- sprawdzić zawartość pliku i zanotować w protokole nazwę pliku z zapisanym przebiegiem i opisać jego zawartość.

### 3.5. Przekształcanie przyrządu wirtualnego w podprogram

Środowisko LabVIEW umożliwia przekształcenie napisanego samodzielnie programu w podprogram, który może być potem wykorzystywany w innych aplikacjach w ten sam sposób, jak wszystkie inne elementy dostępne w paletce *Functions*. W tym celu należy utworzyć ikonę, której obraz będzie identyfikował na diagramie dany podprogram, oraz zdefiniować konektory, które będą służyć jako wejścia i wyjścia danych w diagramie programu. Kolejność postępowania jest następująca:

- otworzyć w LabVIEW program, który chcemy przekształcić w podprogram,
- przełączyć się na widok Panelu,
- w prawym górnym rogu okna Panel znajduje się ikona utworzonego programu, należy umieścić kursor na tej ikonie, prawym kliknięciem rozwinąć listę opcji i wybrać „Edit Icon”,
- dokonać edycji ikony w sposób umożliwiający łatwą identyfikację podprogramu po umieszczeniu go na diagramie,
- ponownie umieścić kursor na ikonie, prawym kliknięciem rozwinąć listę opcji i wybrać opcję „Show Connector”. W prawym górnym rogu zostanie wyświetlona siatka wejść i wyjść programu,
- ponownie umieścić kursor na ikonie i prawym kliknięciem rozwinąć listę opcji, wybrać opcję „Patterns” i z dostępnych wzorów zacisków wybrać konfigurację odpowiadającą potrzebom tworzonego podprogramu,
- w celu przypisania wejść i wyjść danych do konektorów należy kliknąć na wybrany konektor (zmieni on kolor) i następnie kliknąć w oknie Panelu na kontrolkę reprezentującą wybrane wejście lub wyjście danych w naszym podprogramie. Konektor zmieni kolor odpowiednio do typu danych z nim powiązanych,
- przypisać kolejno wszystkie konektory do wejść i wyjść podprogramu, dane do drugiego wejścia i wyjścia siatki ikony z Rys. 1.17a,
- ponownie umieścić kursor na ikonie, prawym kliknięciem rozwinąć listę opcji i wybrać opcję „Show Icon”.
- w celu utworzenia opisu podprogramu w oknie „Front Panel” wybrać opcję File\VI Properties\Documentation i wpisać w polu "VI description" tekst opisujący program, który będzie pojawiał się w menu kontekstowym,
- zapisać program do pliku dyskowego i zamknąć,

W celu wywołania utworzonego podprogramu należy w kolejnym tworzonym nowym programie przejść do okna Diagramu i z palety „*Functions*” wybrać „*Select a VI ..*”, odnaleźć na dysku komputera wcześniej utworzony podprogram i wstawić do diagramu tworzonego nowego programu.

## 4. Wykonanie ćwiczenia

### 4.1. Uruchomienie stanowiska ćwiczeniowego

Włączyć komputer i poczekać na uruchomienie systemu operacyjnego. Zanotować do protokołu konfigurację komputera: wersję systemu operacyjnego, typ procesora, częstotliwość taktowania, ilość pamięci RAM, wielkość dysku twardego. Odczytać z obudowy typ modułu pomiarowego i zanotować do protokołu.

Sprawdzić status modułu sygnalizowany diodą LED według tabelki 2, zapisać wynik obserwacji do protokołu. Sprawdzić stan diody LED dołączonej do wyjścia cyfrowego P0.1, zapisać wynik obserwacji do protokołu.



Sprawdzić układ połączeń stanowiska według schematów przedstawionych na rysunkach 7, 9 i 11. Jeśli moduł pomiarowy 6008/6009 nie jest dołączony do portu USB komputera, należy dołączyć go do komputera postępując według punktu 2.1.

Utworzyć na dysku komputera katalog do zapisywania programów i pozostałych danych zgodnie z zasadą przedstawioną w punkcie 3.1.

#### 4.2. Testowanie modułu akwizycji danych pomiarowych

Uruchomić program Measurement & Automation Explorer i przeprowadzić Self-Test modułu USB-6008/6009 według punktu 2.5. Zanotować rezultat testu do protokołu. Zapisać zrzut ekranu z otwartym oknem Self-Testu do pliku na dysku komputera. Odczytać i zapisać do protokołu numer urządzenia przypisany w systemie wykorzystywanemu modułowi oraz Serial Number modułu.

Przeprowadzić test cyfrowych linii wejścia-wyjścia modułu USB-6008/6009 według punktu 2.5.1. Sprawdzić opcje dostępne w menu konfiguracyjnym, zapisać je do protokołu oraz porównać z dokumentacją modułu USB-6008/6009. Zapisać wnioski do protokołu. Sprawdzić działanie diody LED dołączonej do linii P0.1 (rys.7), najpierw gdy linia ta jest skonfigurowana jako wejście i następnie po skonfigurowaniu jej jako wyjście. Wyjaśnić zachowanie się diody, zapisać wniosek do protokołu. Sprawdzić działanie przycisku S dołączonego do linii P0.0. Ocenić poprawność uzyskanych wyników. Zapisać wnioski do protokołu. W trakcie przeprowadzania testów, przy zapalanej diodzie LED i wciśniętym przycisku S, zapisać zrzut ekranu z otwartym oknem "Test Panels..." do pliku na dysku komputera.

Przeprowadzić test licznika impulsów modułu USB-6008/6009 według punktu 2.5.2. Sprawdzić działanie licznika dla różnych częstotliwości impulsów podanych na wejście PFI0. Zapisać wniosek do protokołu, wyjaśnić zachowanie się licznika. Sprawdzić opcje dostępne w menu konfiguracyjnym licznika i zapisać je do protokołu. Wyjaśnić, dlaczego nie wszystkie opcje konfiguracyjne są dostępne, porównać je z dokumentacją modułu USB-6008/6009. Zapisać wnioski do protokołu. W trakcie przeprowadzania testów zapisać zrzut ekranu z otwartym oknem "Test Panels..." do pliku na dysku komputera.


Przeprowadzić test przetwornika cyfrowo-analogowego modułu USB-6008/6009 według punktu 2.5.3. Sprawdzić działanie przetwornika dla kilku wartości napięcia porównując wartości zadane ze wskazaniami woltomierza. Zapisać wniosek do protokołu. Sprawdzić opcje dostępne w menu konfiguracyjnym licznika i zapisać je do protokołu oraz porównać z dokumentacją modułu. Wyjaśnić, dlaczego nie wszystkie opcje konfiguracyjne są dostępne. Sprawdzić możliwość zadania napięcia spoza zakresu działania przetwornika. Zapisać wnioski do protokołu. W trakcie przeprowadzania testów zapisać zrzut ekranu z otwartym oknem "Test Panels..." do pliku na dysku komputera.



Uwaga ! Jeśli podczas przeprowadzania testów uzyskano negatywne rezultaty, należy odłączyć moduł od portu USB i usunąć go z listy urządzeń w programie Measurement & Automation Explorer. Następnie ponownie dołączyć moduł **do tego samego portu USB** i powtórzyć testy. Poinformować prowadzącego zajęcia o zaistniałej sytuacji.

#### 4.3. Pomiary częstotliwości metodą zliczania impulsów w zadanym czasie.

- Uruchomić środowisko LabVIEW. Odczytać i zanotować do protokołu zainstalowaną wersję LabVIEW. Do wejścia PFI0 dołączyć sygnał TTL z generatora częstotliwości wzorcowej według rysunku 9.

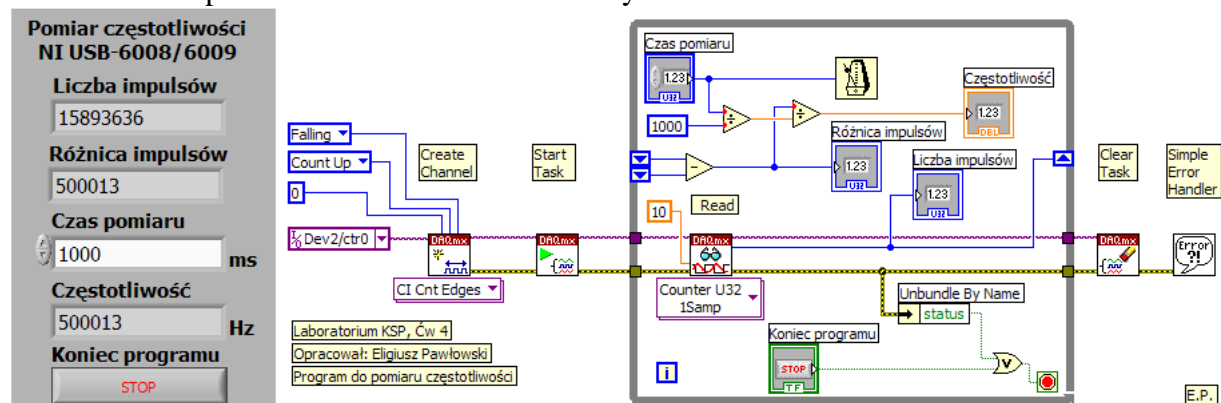
- Przygotować program do odczytu licznika według punktu 2.6. Zapisać program we wskazanym katalogu pod nazwą utworzoną zgodnie z zaleceniami podanymi w punkcie 3.1.

- Uruchomić program przyciskiem . Ponieważ program zawiera pętlę While, powinien on pracować aż do naciśnięcia przycisku STOP na panelu. Nie należy uruchamiać programu

przyciskiem , oraz nie należy go zatrzymywać przyciskiem . Sprawdzić działanie programu dla różnych wartości częstotliwości impulsów podanych na wejście PFI0. Zapisać wnioski do protokołu.

- Zmienić dane programujące licznik w module Create Channel (zobcz, kierunek zliczania, wartość początkowa). Uruchomić program i zaobserwować uzyskane efekty. Zapisać wnioski do protokołu.

- Uzupełnić program o obliczanie wartości częstotliwości. W tym celu należy pętlę While uzupełnić o rejestr przesuwany prawym kliknięciem na prawej krawędzi pętli While (*Add Shift Register*), a następnie powiększyć wyjście rejestru do dwóch zacisków (na lewej krawędzi pętli While). Obliczyć różnicę liczby impulsów odczytanych z licznika w dwóch kolejnych obiegach pętli. Wewnątrz pętli umieścić opóźnienie czasowe wstawiając do niej obiekt Timing\Wait Until Next ms Multiple z dołączonym na wejściu zadajnikiem (create\control) do zadawania czasu opóźnienia w milisekundach. Ustawić domyślną wartość opóźnienia na 100ms. Obliczyć częstotliwość dzieląc różnicę liczby impulsów z kolejnych obiegów pętli While przez czas opóźnienia (w sekundach) wykonania jednej pętli, pamiętając o przeliczeniu wartości czasu pomiaru z milisekund na sekundy.

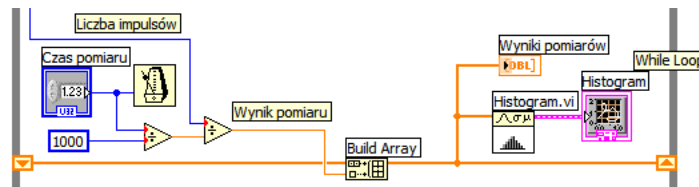


Rys. 23. Przykładowy panel i diagram programu do pomiaru częstotliwości

- Sprawdzić działanie programu dla różnych wartości częstotliwości wzorcowej i dla różnych czasów trwania pomiaru. Sprawdzić działanie programu po zadaniu czasu pomiaru równego zero. Wyjaśnić uzyskany efekt, zapisać wnioski do protokołu.

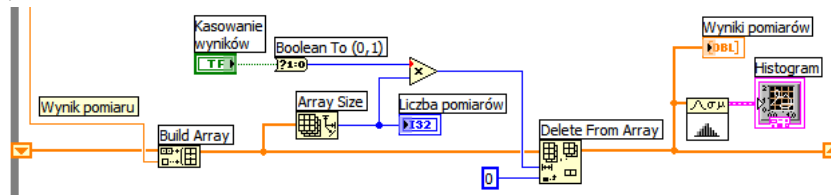
- Uzupełnić Panel i Diagram o informacje wymienione w punkcie 3.1. Na Panelu wykorzystać obiekty graficzne (*decorations*) do uporządkowania jego zawartości. Uaktualnić opisy wszystkich obiektów odpowiednio do spełnianej przez nie roli. Wskaźniki i zadajniki cyfrowe opisać podając odpowiednie jednostki wielkości układu SI. Korzystając z narzędzi Align Objects, Distribute Objects, oraz Reorder wyrównać położenie obiektów na panelu, zgrupować je w całość (Group) i zablokować położenie (Lock).

- Zapisać finalną wersję programu (nr 1) na dysku, zanotować jego nazwę do protokołu.  
 - Wydrukować dokumentację programu do pliku postępując według punktu 3.3.  
 - Wykonać kopię programu na potrzeby realizacji kolejnego zadania (wg. p. 3.1).  
 - Uzupełnić program z poprzedniego punktu o tworzenie histogramu z wartości zmierzonej częstotliwości. Do utworzenia histogramu należy wykorzystać obiekt *Mathematics/Prob&Stat/Histogram* i na jego wyjściu utworzyć wykres (*Create /Indicator*). Zmodyfikować odpowiednio opisy osi histogramu. Histogram korzysta z danych w postaci tablicy *Array*. Aby utworzyć tablicę z kolejnych wyników pomiarów należy utworzyć na krawędzi pętli *While Loop* rejestr przesuwany (*Add Shift Register*) i za pomocą funkcji *Build Array* zsumować każdy kolejny wynik pomiaru częstotliwości z poprzednimi. Dodatkowo na wyjściu funkcji *Build Array* utworzyć wskaźnik umożliwiający przeglądanie zawartości tablicy wyników. Przykładowy diagram przedstawiono na rysunku 23.



Rys. 23. Przykład fragmentu diagramu budującego tablicę i realizującego histogram

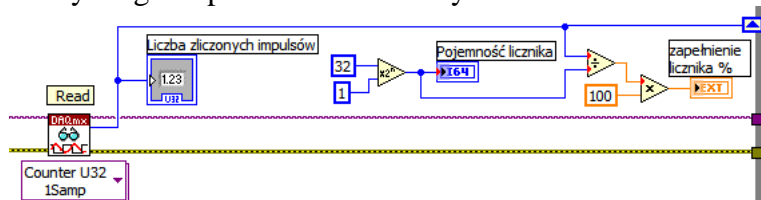
- Uzupełnić program o kasowanie macierzy z wynikami pomiarów. Wykorzystać obiekty Delete From Array i Array Size. Na panelu umieścić przycisk do kasowania macierzy. Należy kasować z macierzy elementy od indeksu 0, zadając długość (length) 0 (nic nie będzie kasowane) lub wartość uzyskaną z obiektu Array Size (cała macierz zostanie skasowana), zależnie od stanu przycisku do kasowania. Liczbę pomiarów w macierzy wyświetlić na panelu tworząc wskaźnik na wyjściu funkcji Array Size. Przykładowy diagram przedstawiono na rysunku 24.



Rys. 24. Przykład fragmentu diagramu kasującego tablicę z wynikami pomiarów

- Uzupełnić program o wykres pokazujący kolejne wyniki pomiarów. W tym celu na panelu postawić obiekt Waveform Graph do jego wejścia dołączyć tablicę z wynikami pomiarów. Zmodyfikować odpowiednio opisy osi wykresu.

- Uzupełnić program o kontrolę i wizualizację stopnia zapełnienia licznika w procentach. W tym celu należy obliczyć pojemność 32 bitowego licznika i podzielić przez nią aktualną zawartość licznika. Na panelu umieścić obiekt Horizontal Fill Slide z włączonym wyświetlaczem cyfrowym (Visible Items/Digital Display) i ustawionym zakresem wskaźników 0..100 %. Przykładowy diagram przedstawiono na rysunku 25.

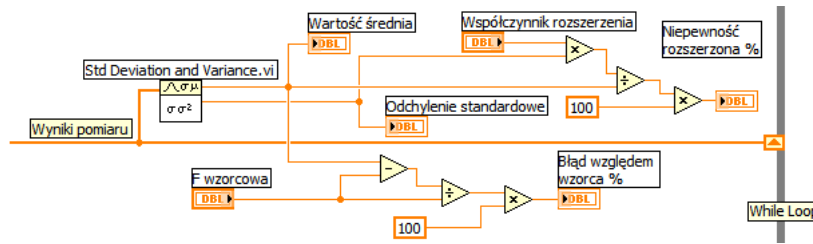


Rys. 25. Przykład fragmentu diagramu obliczającego zapełnienie licznika

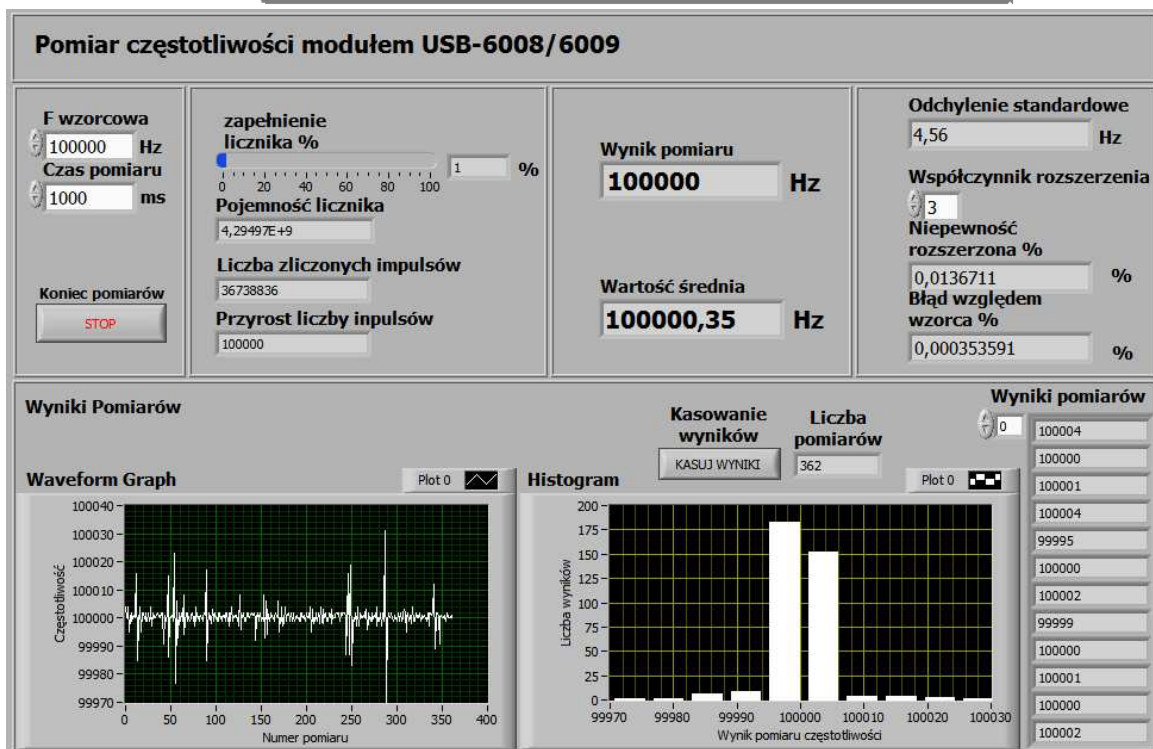
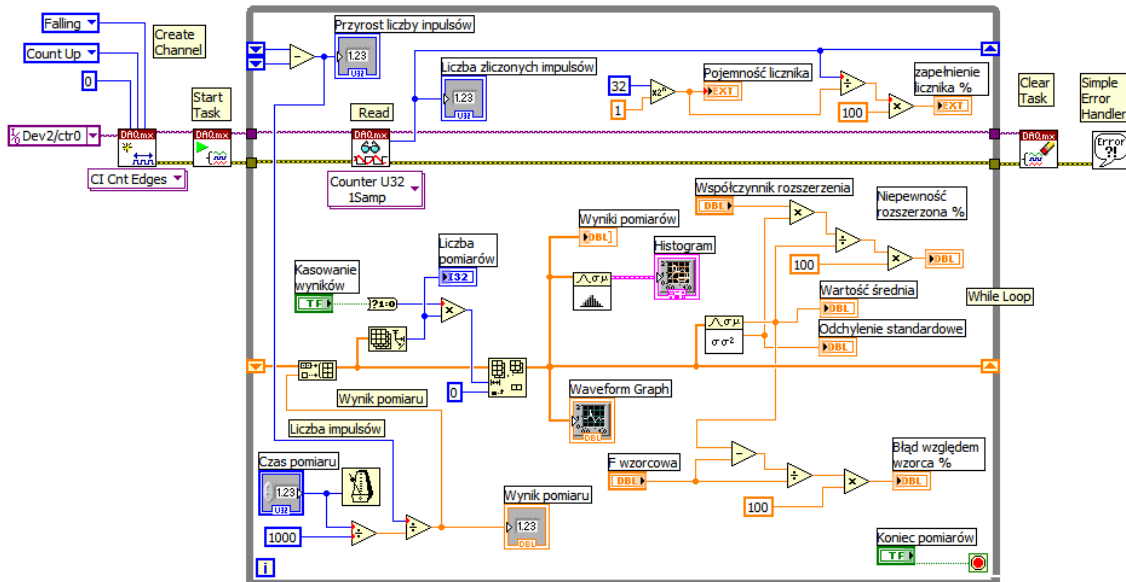
- Uzupełnić program tak, aby obliczał niepewności pomiaru częstotliwości zrealizowaną metodą oraz błąd uśrednionego wyniku pomiaru względem wartości częstotliwości wzorcowej dołączonej do wejścia licznika. Do obliczenia odchylenia standardowego i wartości średniej wykorzystać funkcję Std Deviation and Variance. Obliczyć niepewność rozszerzoną w procentach od wartości średniej. Umieścić na panelu zadajnik do wprowadzania wartości częstotliwości wzorcowej podłączonej do wejścia licznika. Obliczyć błąd względny procentowy uśrednionego wyniku pomiaru częstotliwości względem częstotliwości wzorcowej i wyświetlić jego wartość na panelu. Przykładowy diagram przedstawiono na rysunku 26.

- Uporządkować zawartość panelu i diagramu. Przykładową postać kompletnego programu przedstawiono na rysunku 27.





Rys. 26. Przykład fragmentu diagramu obliczającego błędy i niepewności pomiarów



Rys. 27. Przykładowy końcowy program do pierwszego zadania




- Sprawdzić działanie programu dla różnych wartości częstotliwości wzorcowej i dla różnych czasów trwania pomiaru. Wykonać kilkuminutową rejestrację wyników pomiarów

częstotliwości wzorcowej. Zapisać uzyskane wykresy i histogramy do plików dyskowych postępując według punktu 3.4. W sprawozdaniu należy ocenić dokładność pomiarów częstotliwości możliwych do osiągnięcia zastosowaną metodą.




- Zapisać finalną wersję programu (nr 2) na dysku, zanotować jego nazwę do protokołu.
- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).

#### 4.4. Obsługa wejść - wyjść cyfrowych

- Przygotować program do odczytu stanu wejścia cyfrowego według punktu 2.7. Zapisać program we wskazanym katalogu pod nazwą utworzoną zgodnie z zaleceniami podanymi w punkcie 3.1.

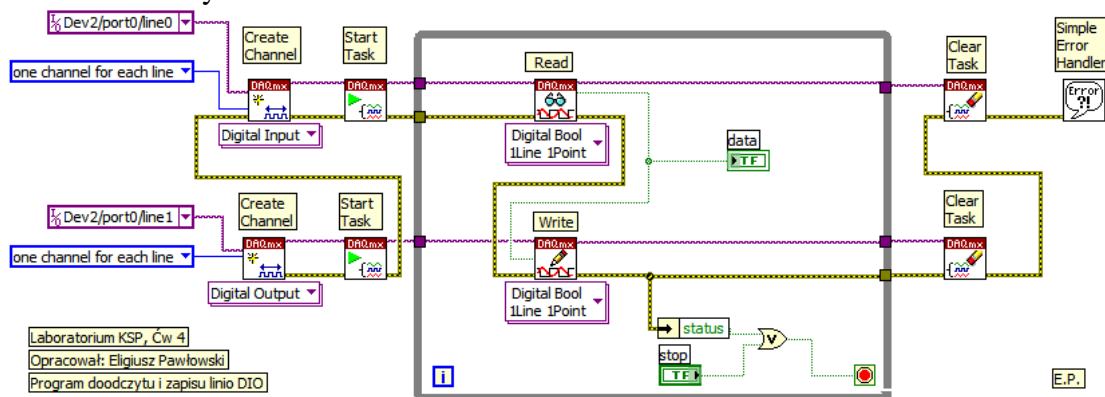
- Uruchomić program przyciskiem . Ponieważ program zawiera pętlę While, powinien on pracować aż do naciśnięcia przycisku STOP na panelu. Nie należy uruchamiać programu przyciskiem , oraz nie należy go zatrzymywać przyciskiem . Sprawdzić działanie programu poprzez naciskanie przycisku dołączonego do wejścia P0.0 modułu pomiarowego. Zapisać wnioski do protokołu.

- Zapisać finalną wersję programu (nr 3) na dysku, zanotować jego nazwę do protokołu.
- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).
- Przygotować program do zapisu stanu wyjścia cyfrowego według punktu 2.8. Zapisać program we wskazanym katalogu pod nazwą utworzoną zgodnie z zaleceniami podanymi w punkcie 3.1.

- Uruchomić program przyciskiem . Ponieważ program zawiera pętlę While, powinien pracować aż do naciśnięcia przycisku STOP na panelu. Nie należy uruchamiać programu przyciskiem , oraz nie należy go zatrzymywać przyciskiem . Sprawdzić działanie programu obserwując zaciśnięcie się diody LED dołączonej do wyjścia P0.1 modułu pomiarowego. Zapisać wnioski do protokołu.

- Zapisać finalną wersję programu (nr 4) na dysku, zanotować jego nazwę do protokołu.
- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).

- Przygotować program do jednoczesnego odczytu i zapisu cyfrowych linii DIO. Program powinien w pętli odczytywać stan przycisku dołączonego do linii P0.0 i od razu zapisywać odczytaną wartość do linii P0.1 sterując dołączoną tam diodą LED. W tym celu należy połączyć na jednym diagramie programy nr 3 i 4. Przykładowy diagram realizujący to zadanie przedstawiono na rysunku 28.






Rys. 28. Diagram programu do jednoczesnego odczytu i zapisu linii cyfrowych

- Sprawdzić działanie programu naciskając przycisk i obserwując diodę LED.

- Uporządkować zawartość panelu i diagramu. Zapisać finalną wersję programu (nr 5) na dysku, zanotować jego nazwę do protokołu.
- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).

#### 4.5. Programowanie przetwornika cyfrowo-analogowego

- Przygotować program do obsługi przetwornika analogowo-cyfrowego według punktu 2.9. Zapisać program we wskazanym katalogu pod nazwą utworzoną zgodnie z zaleceniami podanymi w punkcie 3.1.

- Uruchomić program przyciskiem . Ponieważ program zawiera pętlę While, powinien on pracować aż do naciśnięcia przycisku STOP na panelu. Nie należy uruchamiać programu przyciskiem , oraz nie należy go zatrzymywać przyciskiem . Sprawdzić działanie programu i przetwornika dla kilku wartości napięcia, zapisując do protokołu wartości zadane i zmierzone woltmierzem. Obliczyć błędy i porównać je z dokumentacją modułu USB-6008/6009. Zapisać wniosek do protokołu.

- Uporządkować zawartość panelu i diagramu. Zapisać finalną wersję programu (nr 6) na dysku, zanotować jego nazwę do protokołu.

- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).

#### 6. Badanie przetwornika napięcie-częstotliwość

- Dołączyć do modułu pomiarowego przetwornik napięcie-częstotliwość VFC według schematu na rysunku 11. Przygotować program do automatycznego wyznaczenia charakterystyki przetwarzania i jej błędów względem nominalnej wartości stałej przetwarzania. Program powinien w pętli zadawać narastające wartości do przetwornika cyfrowo-analogowego i równocześnie wykonywać pomiar częstotliwości sygnału wyjściowego VFC. W tym celu należy połączyć na jednym diagramie programy nr 2 i 6. Pętle While należy zastąpić pętlami For. W tunelu wyjściowym małej pętli For wyłączyć indeksowanie, dzięki czemu tylko ostatni wynik pomiaru zostanie przekazany do dalszych obliczeń. Umieścić na panelu zadajniki do wprowadzania: liczby kroków, wartości kroku, nominalnej stałej przetwarzania.

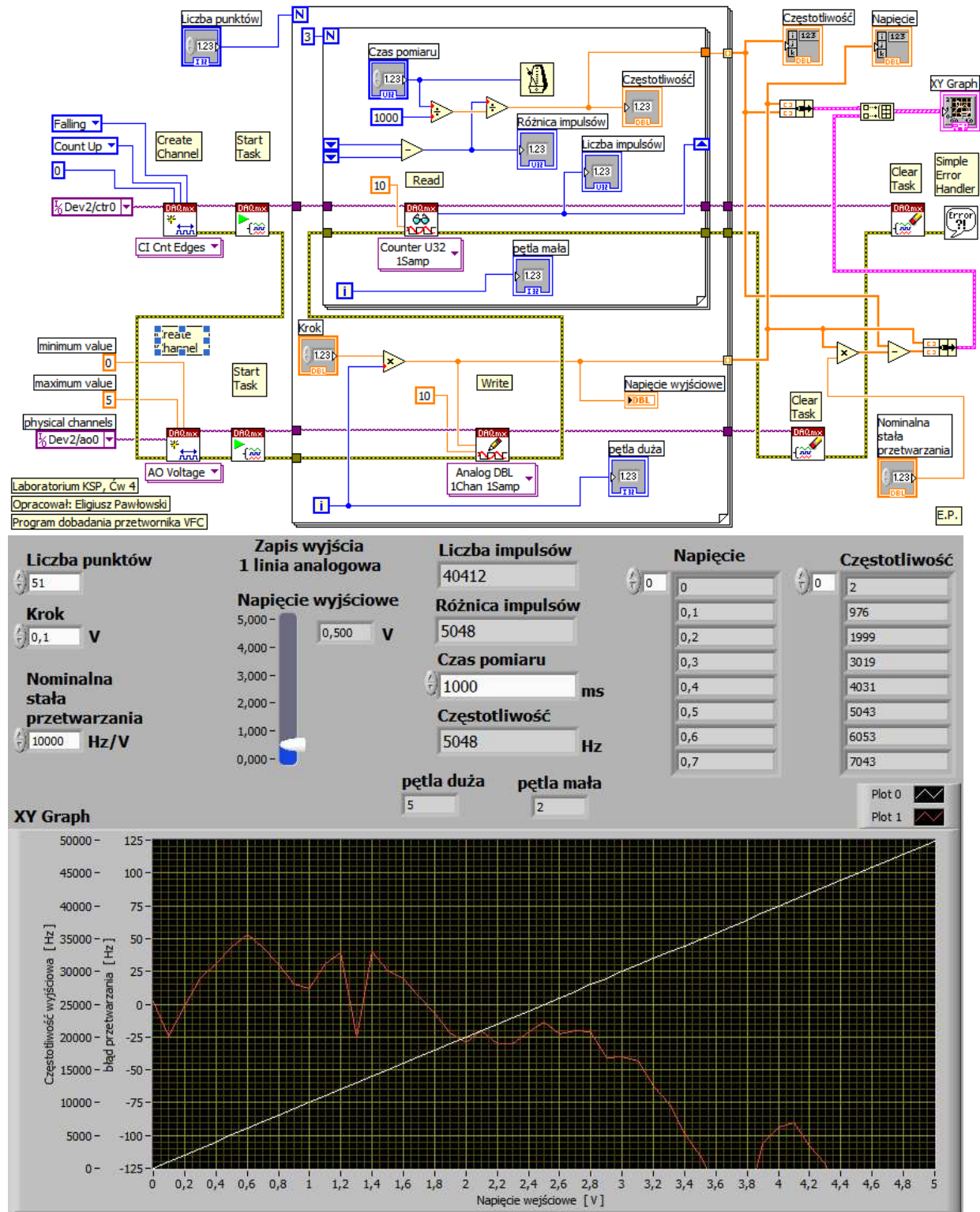
Do generowania wykresu należy użyć funkcji XY Graph. Najpierw należy połączyć ze sobą dane dla osi X (napięcie) i Y (częstotliwość i błąd) za pomocą funkcji Bundle, a następnie dane dla obu wykresów połączyć w jedną macierz za pomocą funkcji Build Array.

Przykładowy program realizujący to zadanie przedstawiono na rysunku 29.

- Sprawdzić działanie programu dla różnych wartości parametrów: liczba kroków, krok, czas pomiaru. Wykonać badanie przetwornika VFC dołączonego do modułu pomiarowego. Zapisać uzyskane wykresy do pliku dyskowego postępując według punktu 3.4. W sprawozdaniu należy ocenić błędy badanego przetwornika VFC.

- Uporządkować zawartość panelu i diagramu. Zapisać finalną wersję programu (nr 7) na dysku, zanotować jego nazwę do protokołu.

- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.
- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).



Rys. 29. Diagram programu do automatycznego badania przetwornika VFC

#### 4.7. Badanie silnika prądu stałego małej mocy

- Dołączyć do modułu pomiarowego silnik prądu stałego małej mocy z przetwornikiem obrotowo-impulsowym według schematu na rysunku 12. Zmodyfikować program z poprzedniego punktu (nr 7) przystosowując go do automatycznego zaleźności prędkości obrotowej silnika od napięcia zasilania. Program powinien w pętli zadawać narastające wartości do przetwornika cyfrowo-analogowego od 0V do +5V i malejące z powrotem do 0V.



Równocześnie powinien wykonywać pomiary częstotliwości sygnału wyjściowego z przetwornika obrotowo-impulsowego i przeliczać je na prędkość obrotową z uwzględnieniem liczby otworów w tarczy.

- Sprawdzić działanie programu dla różnych wartości parametrów: liczba kroków, krok, czas pomiaru. Wykonać badanie silnika dołączonego do modułu pomiarowego. Zapisać uzyskane wykresy do pliku dyskowego postępując według punktu 3.4.

- Uporządkować zawartość panelu i diagramu. Zapisać finalną wersję programu (nr 8) na dysku, zanotować jego nazwę do protokołu.

- Wydrukować dokumentację programu do pliku postępując według punktu 3.3.

- Wykonać kopię programu na potrzeby realizacji kolejnego punktu (wg. p. 3.1).

## 5. Sprawozdanie ze zrealizowanego ćwiczenia

Każda grupa przygotowuje z ćwiczenia jedno całościowe sprawozdanie zawierające:

- stronę tytułową z danymi osobowymi zespołu, datą wykonania ćwiczenia i jego tytułem,
- opis wykorzystywanego środowiska ćwiczeniowego z podaniem wykazu wykorzystywanej aparatury oraz zestawieniem jej parametrów,
- schemat wykorzystywanego układu pomiarowego,
- tematy kolejnych zadań przewidzianych programem ćwiczenia,
- opis słowny każdego ze zrealizowanych zadań lub podanie przyczyny niezrealizowania,
- dokumentację samodzielnie opracowanych i skutecznie uruchomionych programów (niezbędne **opisy na panelu** oraz **komentarze na diagramie** programu !!!),
- schematy blokowe algorytmów dla zrealizowanych **samodzielnie programów**,
- słowny opis zrealizowanych **samodzielnie** algorytmów,
- wyniki pomiarów uzyskane za pomocą przygotowanych programów,
- wnioski ze zrealizowanych ćwiczeń, a w szczególności: napotkane trudności i przyjęte sposoby ich rozwiązania, propozycje zmian i rozszerzeń programu realizowanych zadań, propozycje dalszych modyfikacji i ulepszeń realizowanego zadania, propozycje nowych tematów zadań, propozycje modyfikacji i rozszerzeń układu pomiarowego.

W sprawozdaniu ocenie podlegają:

- Poprawność działania programu oraz jego odporność na nietypowe sygnały i złą obsługę.
- Czytelność i przejrzystość diagramu,
- Estetyka utworzonego Panelu, dobór kolorystyki, elementów graficznych, wyrównanie położenia obiektów, funkcjonalność panelu pod kątem obsługi,
- Zakres zrealizowanych zadań,
- Trafność wniosków końcowych.

## 6. Literatura

1. Nawrocki W.: Komputerowe systemy pomiarowe, WKiŁ, Warszawa 2002.
2. Winięcki W.: Organizacja komputerowych systemów pomiarowych, Oficyna Wydawnicza PW, Warszawa 1997.
3. Wiesław Tłaczała: Środowisko LabVIEW w eksperymencie wspomaganym komputerowo. Wyd. WNT, Warszawa 2005 r.
4. Marcin Chruściel: LabVIEW w praktyce. Wyd. BTC, Warszawa 2008 r.
5. Dariusz Świsulski: Komputerowa technika pomiarowa Oprogramowanie wirtualnych przyrządów pomiarowych w LabView, Wyd. PAK, Warszawa 2005r.
6. Rosłonec S.: Wybrane metody numeryczne, Oficyna Wydawnicza PW, Warszawa 2008.
7. Witold Kaczurba: [Kurs dla początkujących w Labview](http://www.kaczurba.pl), <http://www.kaczurba.pl>

8. National Instruments: LabVIEW Tutorial Manual  
<http://ftp.ni.com/support/manuals/320998a.pdf>
9. NI USB-6008/6009 Quick Start, National Instruments Co.,  
<http://www.ni.com/pdf/manuals/375553a.pdf>
10. USB-6008/6009 User Guide and Specifications, National Instruments Co.,  
<http://www.ni.com/pdf/manuals/371303m.pdf>