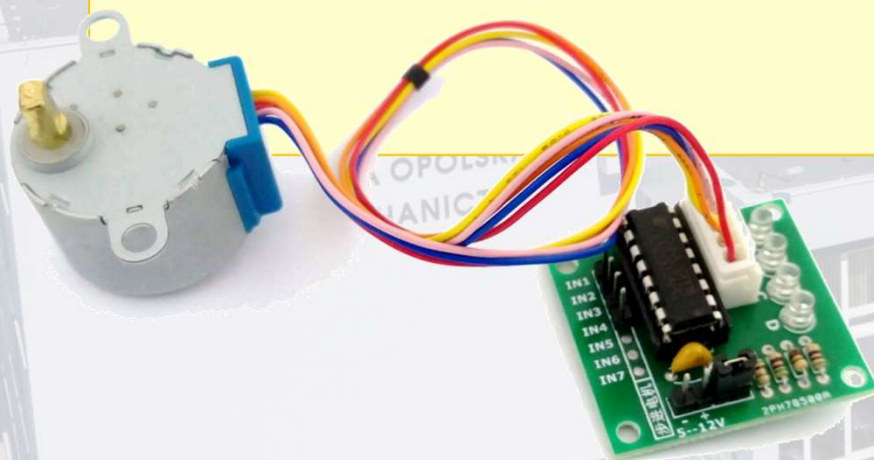




Katedra Mechaniki i Podstaw Konstrukcji Maszyn
Laboratorium z mechatroniki

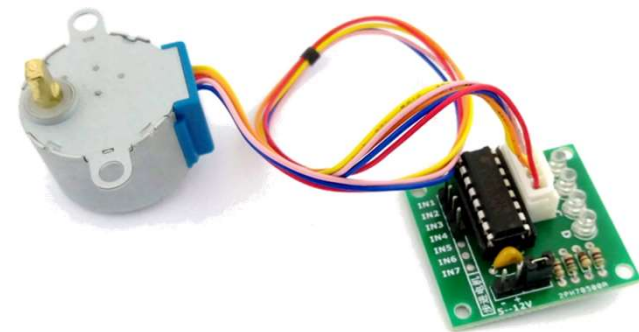
Wirtualny sterownik silnika krokowego

Roland
Pawliczek



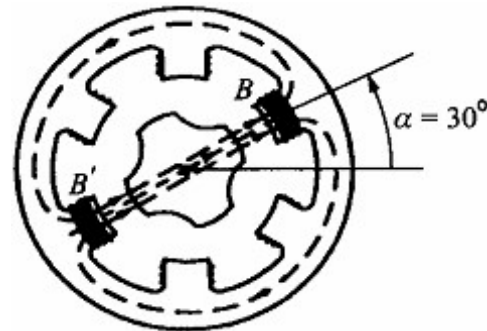
Agenda

- *Budowa i działanie silnika krokowego*
- *Układ sterowania silnika krokowego*
- *Środowisko programowania graficznego LabVIEW*
- *Program sterujący*
- *Co to jest wirtualny układ kontrolo-pomiarowy*
- *Przykłady*



Budowa i zasada działania silnika krokowego

Wykorzystuje się zjawisko zmiany oporności magnetycznej (reluktancji).



Kąt obrotu na 1 krok:

$$\alpha = \frac{360^\circ}{pn}$$

*p - liczba par biegunów (faz) stojana,
n - liczba zębów wirnika.*

Silnik krokowy jest silnikiem synchronicznym. Obroty rotora uzyskuje się przez wirowanie pola magnetycznego - wirnik podąża do nowego stabilnego położenia.

Budowa i zasada działania silnika krokowego

Liczba kroków: 200 (1,8° na krok)

Napięcie znamionowe: 7,4V

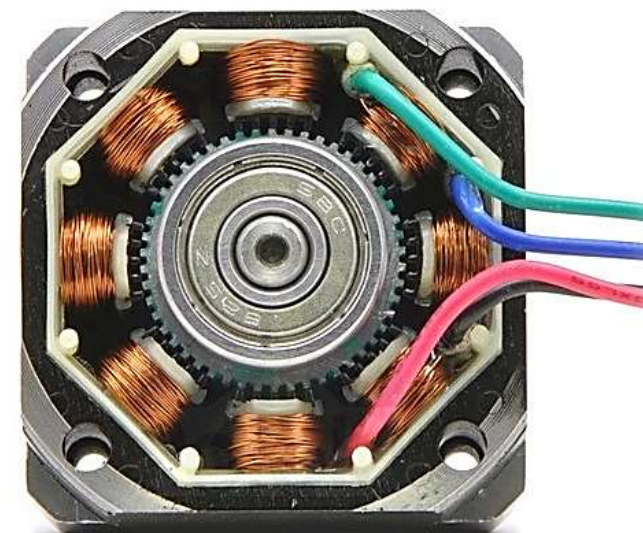
Pobór prądu na cewkę: 280mA

Rezystancja cewki: 26 Ohm

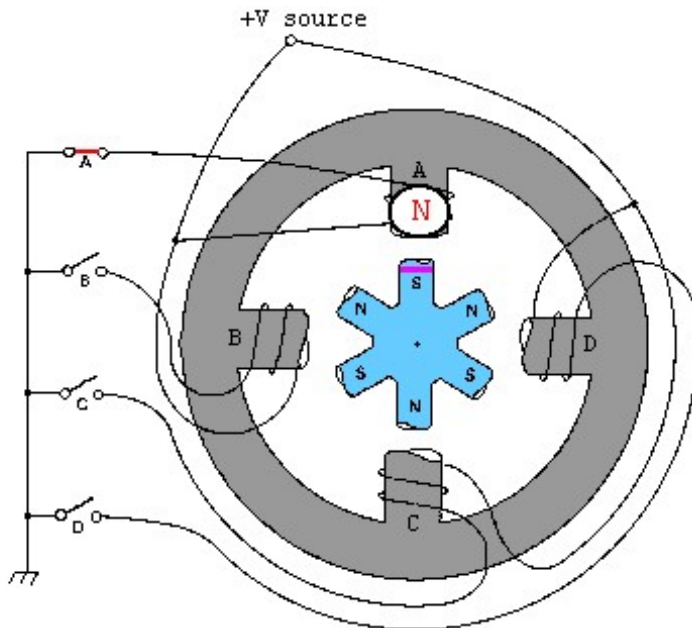
Indukcyjność uzwojenia: 19,2mA

Moment trzymający 650g·cm (0,0637 N·m)

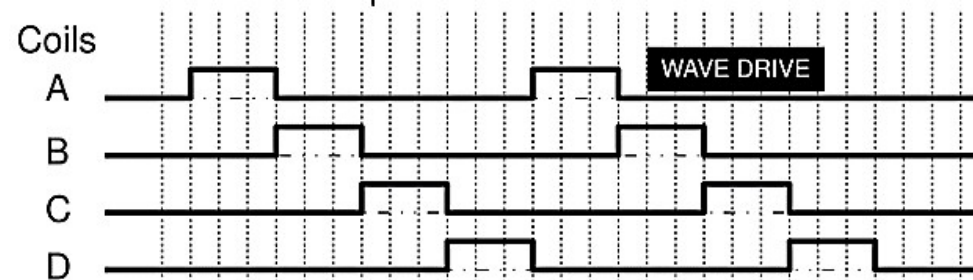
Wymiary: 35mm x 35mm x 26mm



Budowa i działanie silnika krokowego

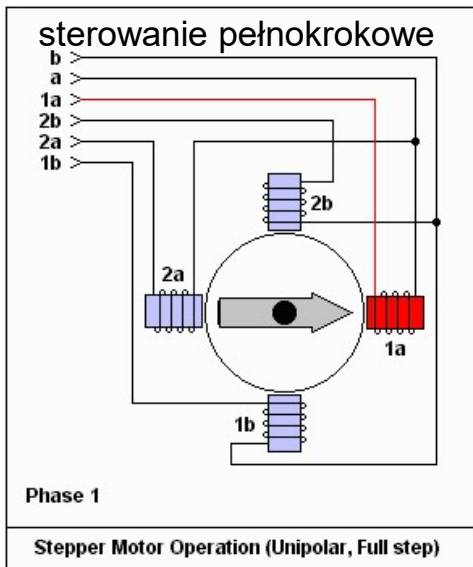


- **Obrót wirnika** zależy tylko od sekwencji wysyłanych impulsów.
- **Kąt obrotu** → liczba obrotów zależy tylko od **liczby impulsów**.
- **Prędkość obrotowa** wału zależy od **częstotliwości impulsów**.
- Jeżeli silnik zostanie zatrzymany pojawia się **stały moment trzymający** (statyczny), który może być wykorzystany jako hamulec.
- Jeżeli zliczane będą impulsy sterujące i kontrolowana ich częstotliwość układ sterowania może być w pętli otwartej (nie potrzeba czujnika prędkości obrotowej i regulatora).



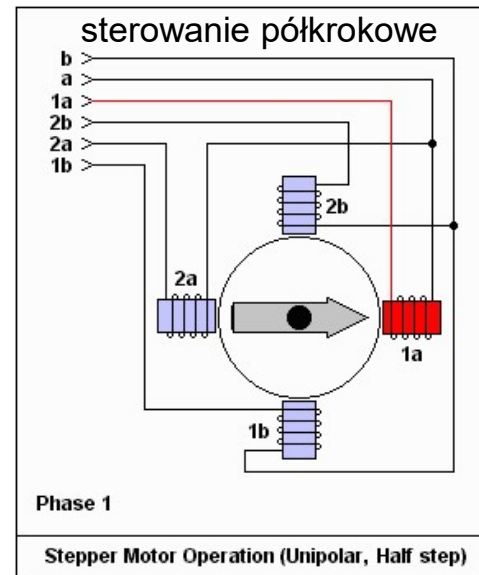
Układ sterowania silnika krokowego

Binarne kody sterowania:



Clockwise Rotation ↻

Index	1a	1b	2a	2b
1	1	0	0	0
2	0	1	0	0
3	0	0	1	0
4	0	0	0	1
5	1	0	0	0
6	0	1	0	0
7	0	0	1	0
8	0	0	0	1

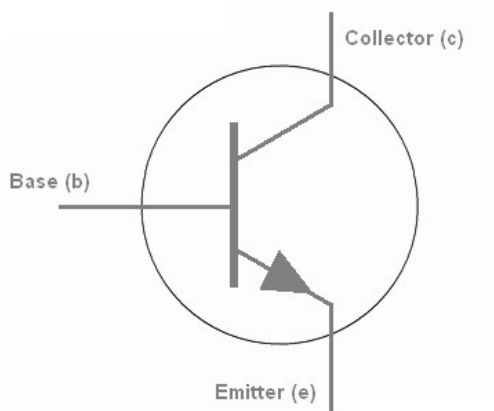


Clockwise Rotation ↻

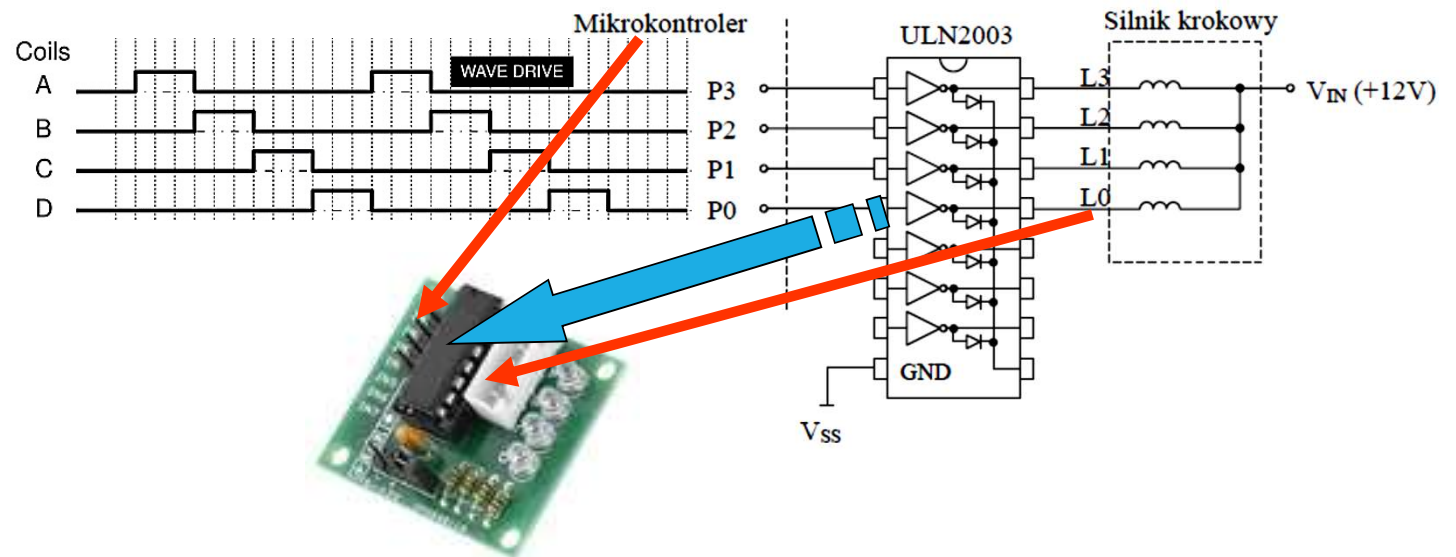
Index	1a	1b	2a	2b
1	1	0	0	0
2	1	1	0	0
3	0	1	0	0
4	0	1	1	0
5	0	0	1	0
6	0	0	1	1
7	0	0	0	1
8	1	0	0	1
9	1	0	0	0
10	1	1	0	0
11	0	1	0	0
12	0	1	1	0
13	0	0	1	0
14	0	0	1	1
15	0	0	0	1
16	1	0	0	1

Układ sterowania silnika krokowego

Prądy w uzwojeniach silnika są kluczowane przez tranzystory zawarte w układzie scalonym ULN2003



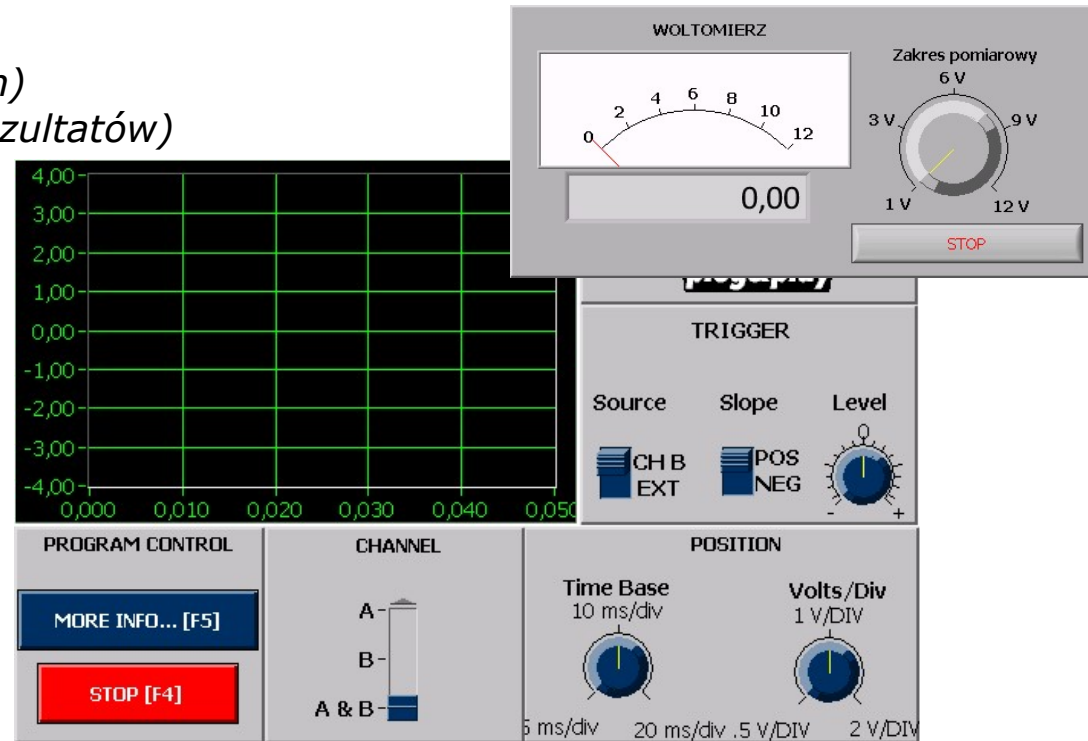
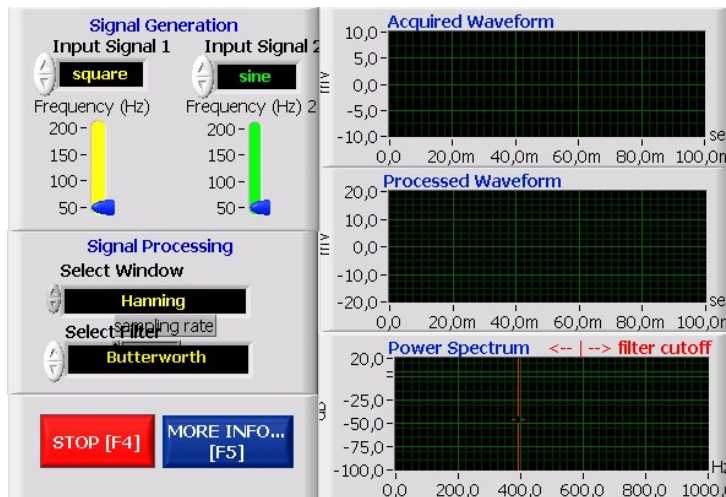
© www.petervis.com



Środowisko programowania graficznego LabVIEW

Panel czołowy jest płytą czołową urządzenia

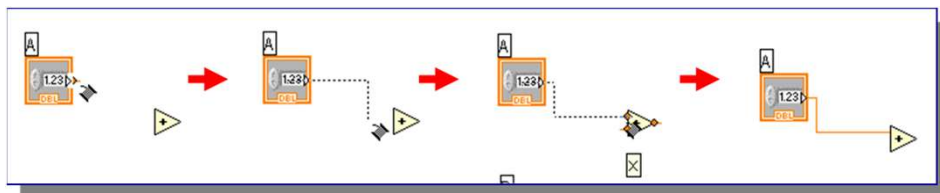
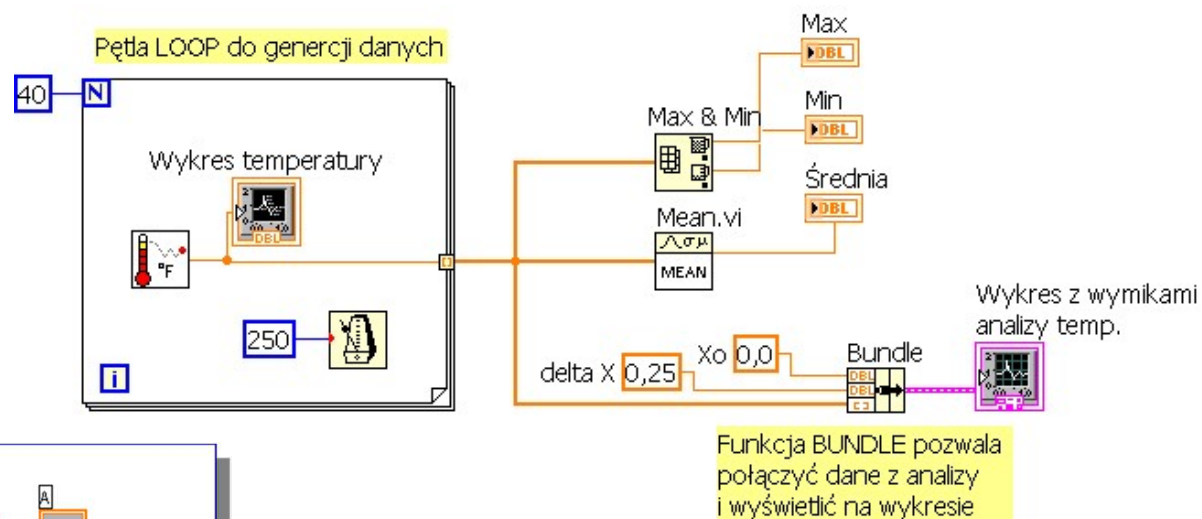
- Kontrolki - Controls (wprowadzanie danych)
- Wyświetlacze - Indicators (wyświetlanie rezultatów)
- inne elementy (dekoracje, rysunki, teksty)



Środowisko programowania graficznego LabVIEW

Schemat Blokowy (Block Diagram) (Kod graficzny aplikacji)

- ikony połączone liniami
- pętle struktury, funkcje, podprogramy
- inne elementy (dekoracje, rysunki, teksty)



Program sterujący

Otworzyć program **sterownik start.vi**

The image displays a LabVIEW interface for a stepper motor controller, divided into a control panel on the left and a block diagram on the right.

Control Panel:

- Header:** POLITECHNIKA OPOLSKA, LABORATORIUM MECHATRONIKI, Sterownik silnika krokowego.
- Sekwencja sterowania:** A 3x3 grid of green indicator lights.
- Częstotliwość:** A rotary knob with a scale from 1 to 200. The current value is 1.
- Zasilanie uzwojeń:** Four orange indicator lights.
- Footer:** KONIEC PROGRAMU.

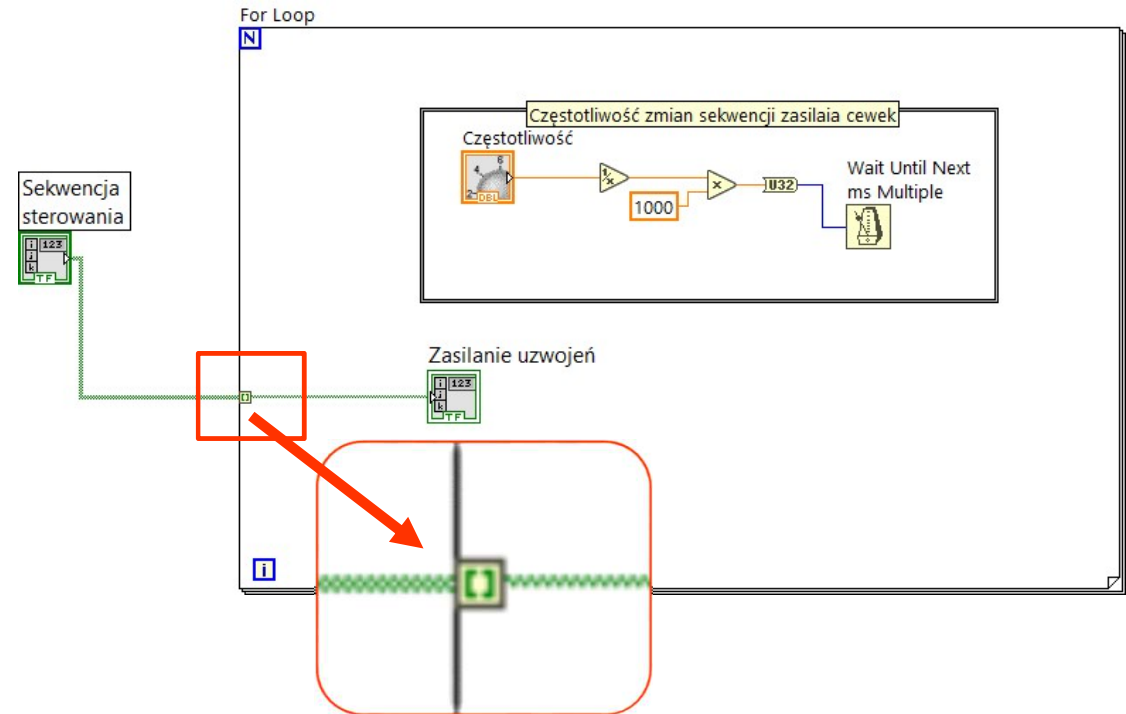
Block Diagram:

- Sekwencja sterowania:** A green 'Start' button icon.
- Częstotliwość zmian sekwencji zasilania cewek:** A sub-diagram containing:
 - A 'Częstotliwość' control knob.
 - A multiplier block with a value of 1000.
 - A 'Wait Until Next ms Multiple' block with a value of 1000.
- Zasilanie uzwojeń:** A green 'Start' button icon.
- Zakończenie pracy silnika:** A sub-diagram containing:
 - A 'STOP' button icon.
 - A red indicator light.

Program sterujący

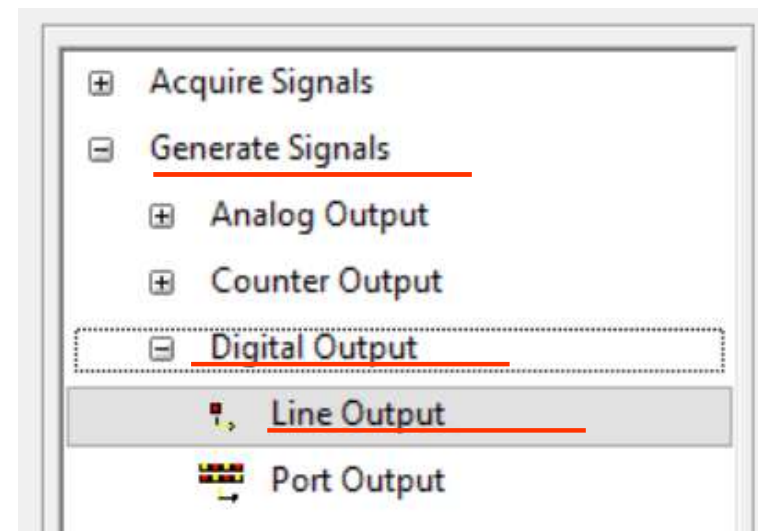
1. Z palety funkcji wybrać **STRUCTURES / FOR LOOP**.
2. Narysować pętlę FOR LOOP tak, aby **Zasilanie uzwojeń** oraz **Częstotliwość** znalazły się wewnątrz pętli.
3. Połączyć **Sekwencję sterowania** z **Zasilaniem uzwojeń** przez pętlę FOR LOOP.

Sprawdzić, czy tunel wejściowy jest ustawiony jako indeksowany → to zmusi program pobierania danych kolejnymi wierszami z tabeli.



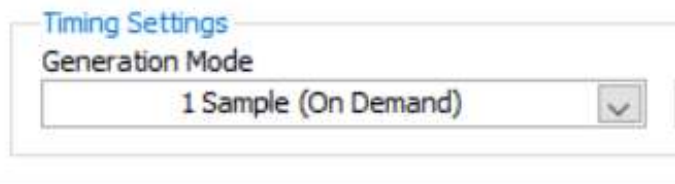
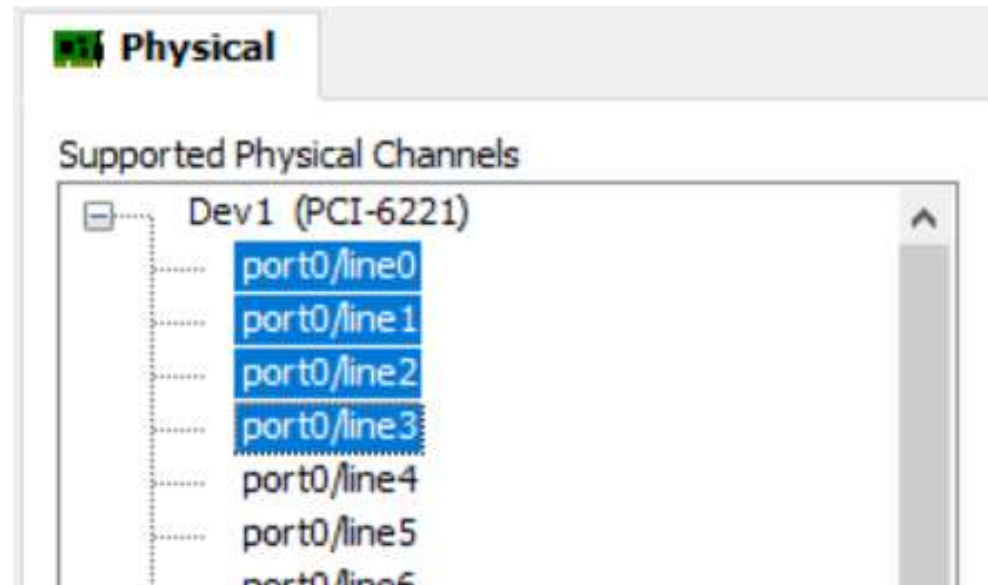
Program sterujący

3. Z palety funkcji wybrać **EXPRESS / INPUT / DAQAssistant** i wstawić funkcję do wnętrza pętli FOR LOOP.
4. Wybrać kolejno: **Generate Signals / Digital Output / Line output**



Program sterujący

5. Na liście sprzętu odszukać kartę PCI-6221.
6. Przytrzymać klawisz CTRL i zaznaczyć terminale:
port0/line0
port0/line1
port0/line2
port0/line3
i zatwierdzić wybór.
7. W oknie konfiguracji sprawdzić ustawienie
Timing Settings = 1 Sample (On Demand)



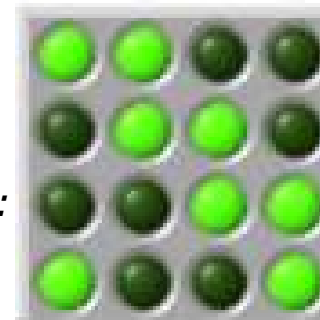
8. Zatwierdzić konfigurację.
-

Program sterujący

9. Połączyć linię z **Zasilania uzwojeń** z linią data funkcji **DAQ Assistant**.



Sekwencja sterowania



10. Przełączyć się na Panel czołowy i ustawić Sekwencję sterowania:

9. Uruchomić program, zweryfikować zapalenie się diod kontrolnych terminali **P0.0**, **P0.1**, **P0.2** i **P0.3**.

Zatrzymać program !

Program sterujący

11. Podłączyć na płytce stykowej przewody terminali z BNC 2120 do sterownika.

P0.0 → IN1
P0.1 → IN2
P0.2 → IN3
P0.3 → IN4

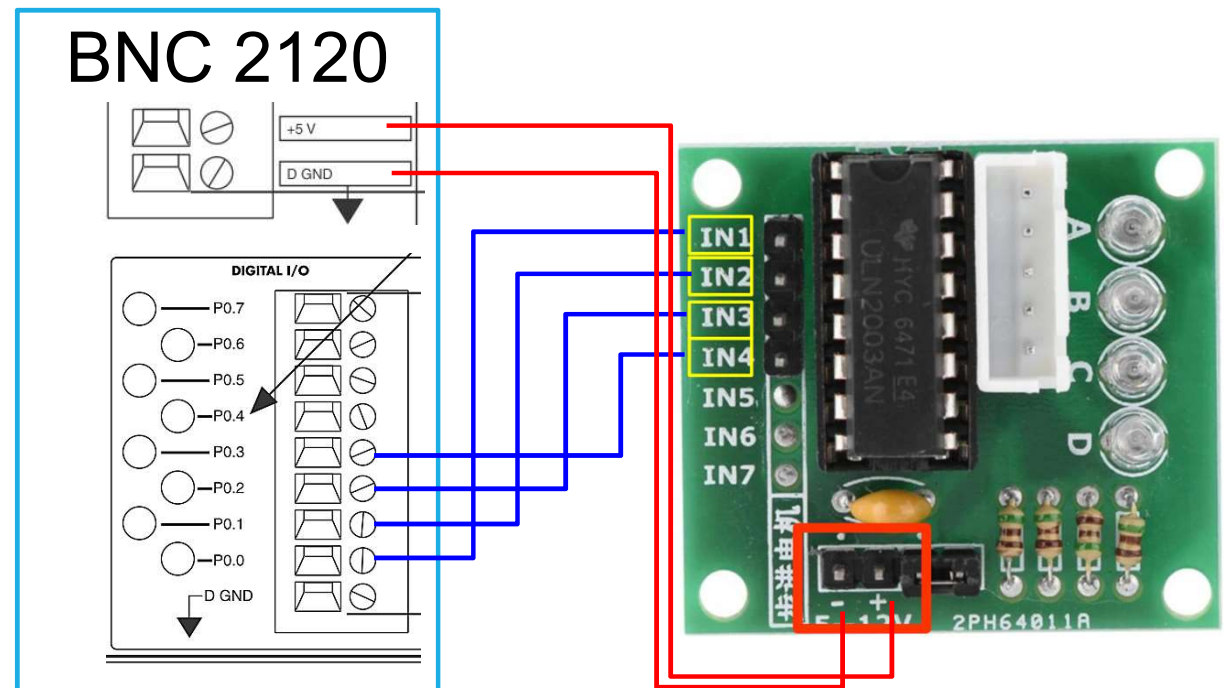
12. Podłączyć przewody zasilania +5V do sterownika.

GND → -
+5V → +

13. Uruchomić program.

Zweryfikować pracę sterownika.

13. Zatrzymać program.

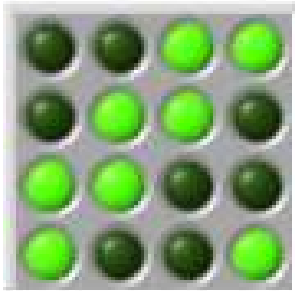


Program sterujący

ZMIANA KIERUNKU OBROTU:

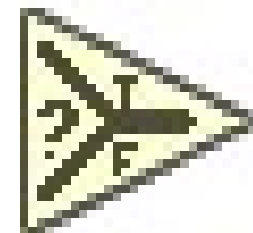
1. W oknie Frant Panel skopiować tablicę Sekwencji sterowania → zaznaczyć / Ctrl+C / Ctrl+V i zmienić sekwencję:

Sekwencja
sterowania 2



„Lustrzane
odbicie”

Select

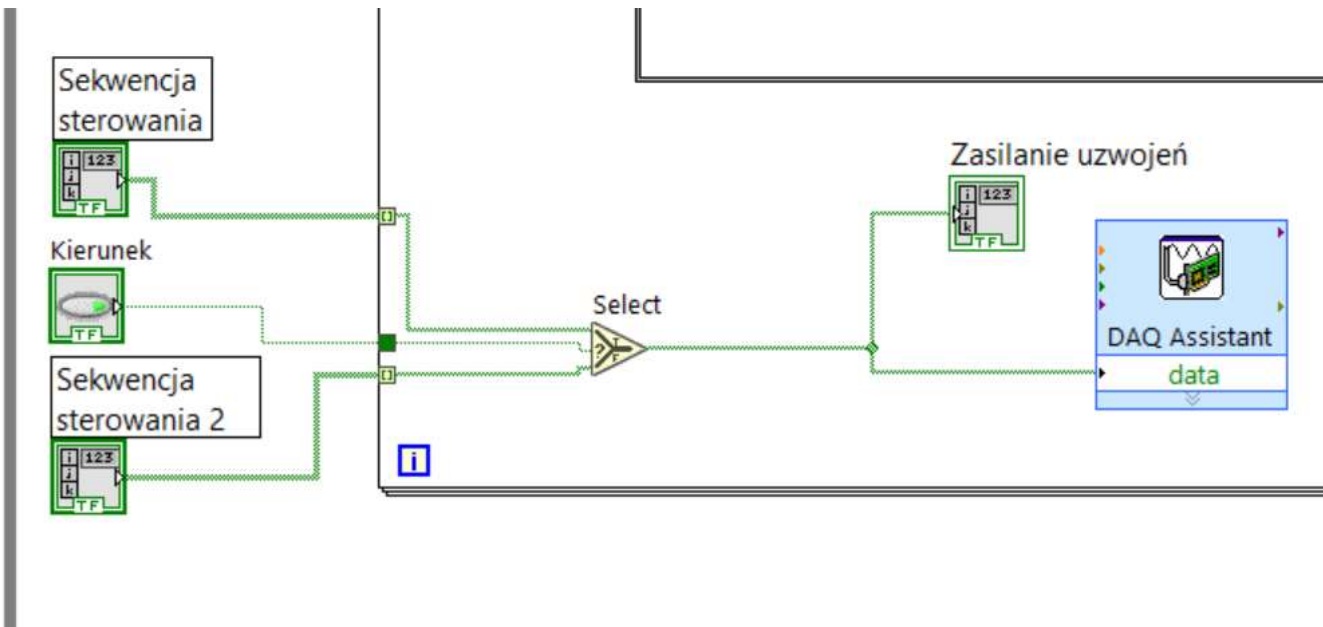


2. W oknie Block Diagram wstawić funkcję PROGRAMMING / COMPARISON / SELECT
-

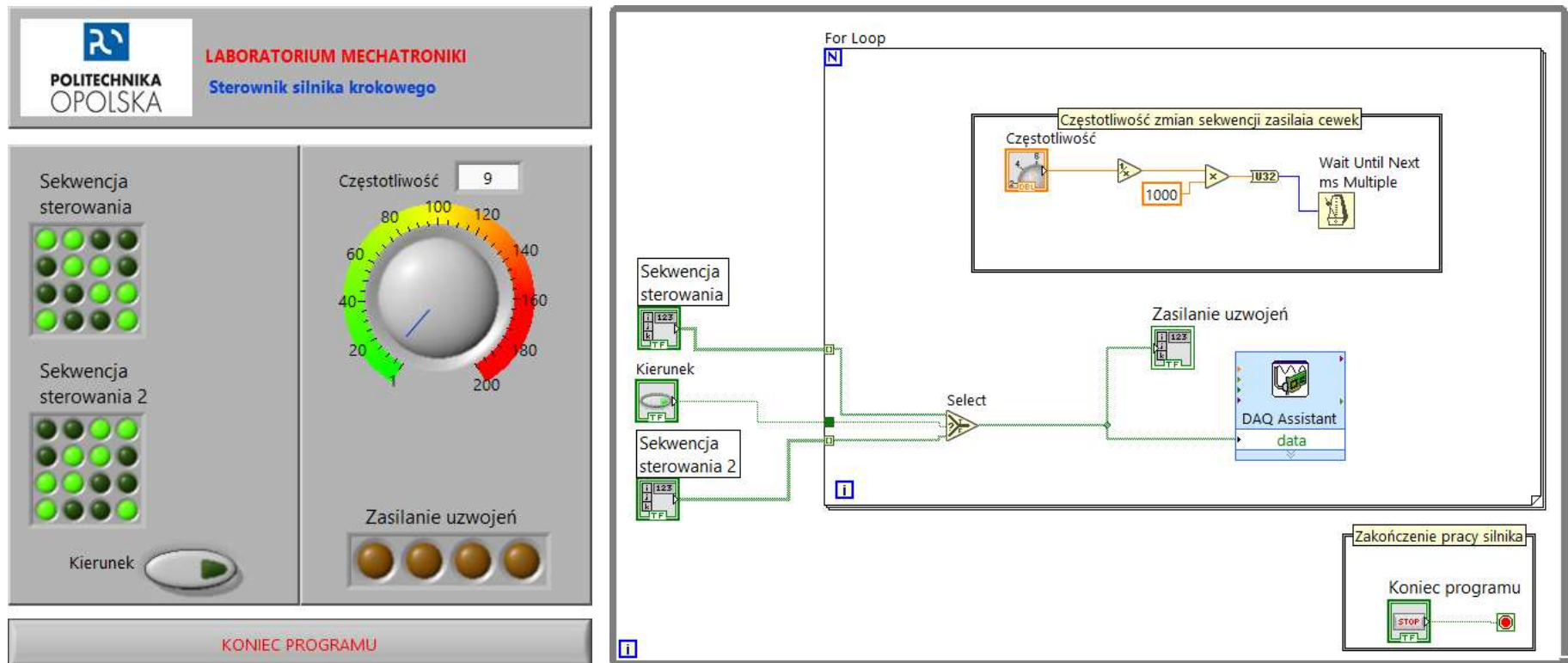
Program sterujący

3. W oknie Front Panel wstawić przycisk: BOOLEAN / PUSH BUTTON
4. W oknie Block Diagram zmodyfikować program.

Kierunek 



Program sterujący



„Tradycyjny” układ sterujący

Sterowniki przemysłowe → **tradycyjne**

Programowalny sterownik
WOBIT SMC64v2 ©



Zadajnik trajektorii
WOBIT MG-Z1 ©



**OPROGRAMOWAINE
PRODUCENTA**



Czasami !!!

**Ograniczona
funkcjonalność !!!**

„Tradycyjny” układ sterujący



Przyrząd „WIRTUALNY”

Komunikacja CZUJNIK → STACJA ROBOCZA

CompactDAQ



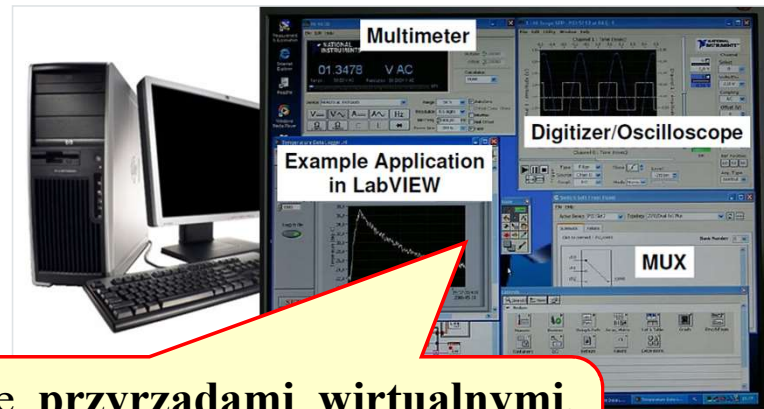
Konektory + karta pomiarowa DAQ



Arduino ???

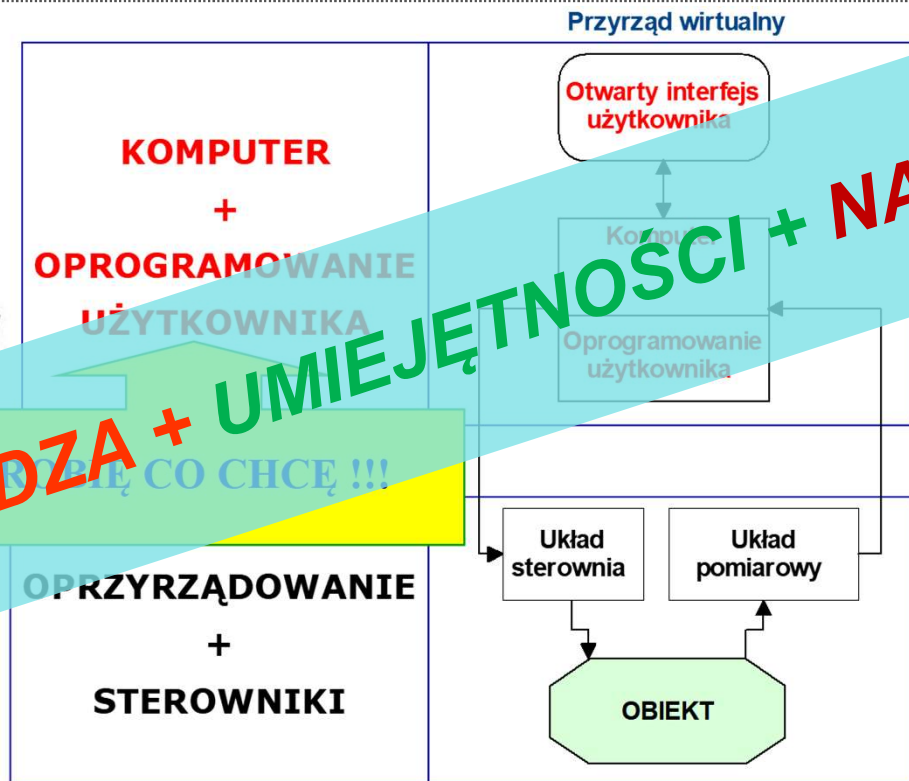


Stacja robocza + oprogramowanie



Programy są nazywane przyrządami wirtualnymi, ponieważ ich wygląd i działanie przypomina rzeczywiste przyrządy, jak oscyloskop czy multimetr.

Przyrząd „WIRTUALNY”



INŻYNIER = WIEDZA + UMIEJĘTNOŚCI + NARZĘDZIA !!!

ROBIĘ CO CHCĘ !!!

Dziękuję za uwagę !

Zapraszam do odwiedzenia **WIRTUALNEGO** Wydziału Mechanicznego !!!
wm.po.edu.pl

... a przede wszystkim **Poziom -1** → **Korytarz E** → **Sala 011** → **Laboratorium Mechatroniki**

