

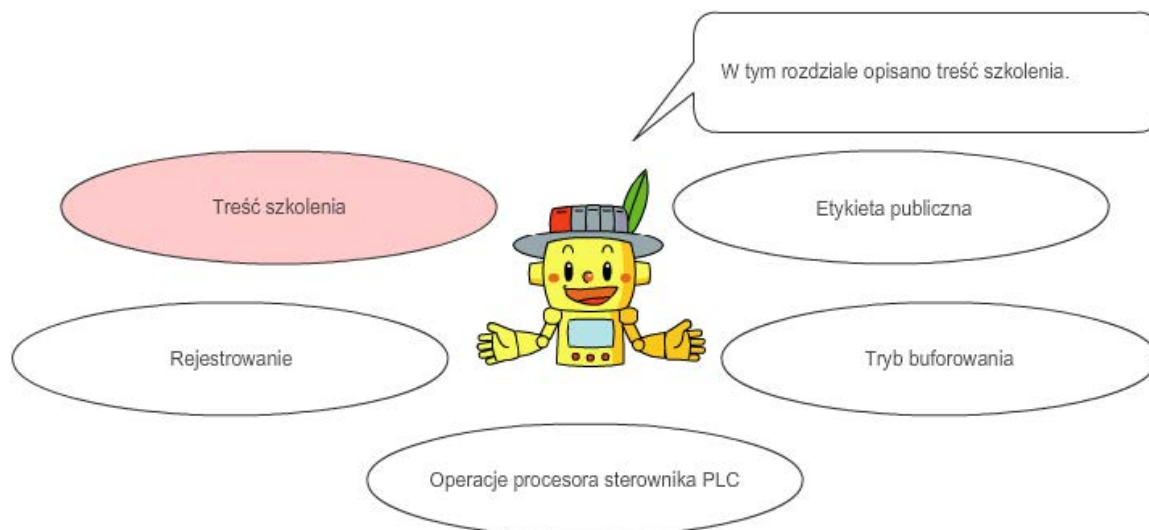
Sterownik systemu serwomechanizmów

Moduł ruchu MELSEC iQ-R – informacje podstawowe (sterowanie pozycjonowaniem RD78G(H))

Szkolenie jest przeznaczone dla osób, które po raz pierwszy będą konstruować system sterowania ruchem wykorzystujący moduł ruchu serii MELSEC iQ-R.

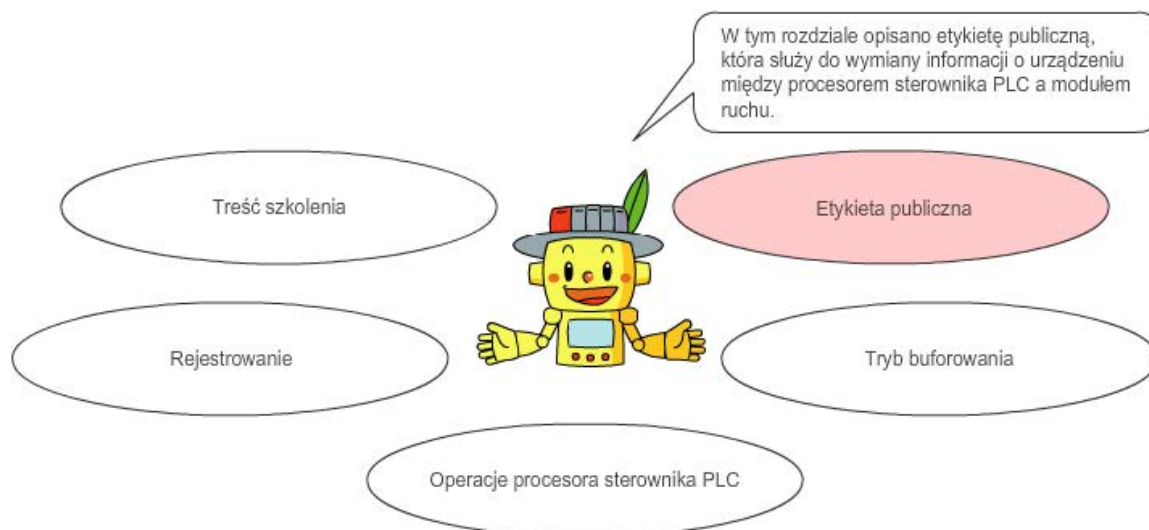
Kliknij przycisk Dalej w prawym górnym rogu ekranu, aby przejść do następnej strony.

Celem niniejszego szkolenia jest rozwinięcie wiedzy i zrozumienia na temat sterowania pozycjonowaniem systemu sterowania ruchem z wykorzystaniem modułu ruchu serii MELSEC iQ-R.



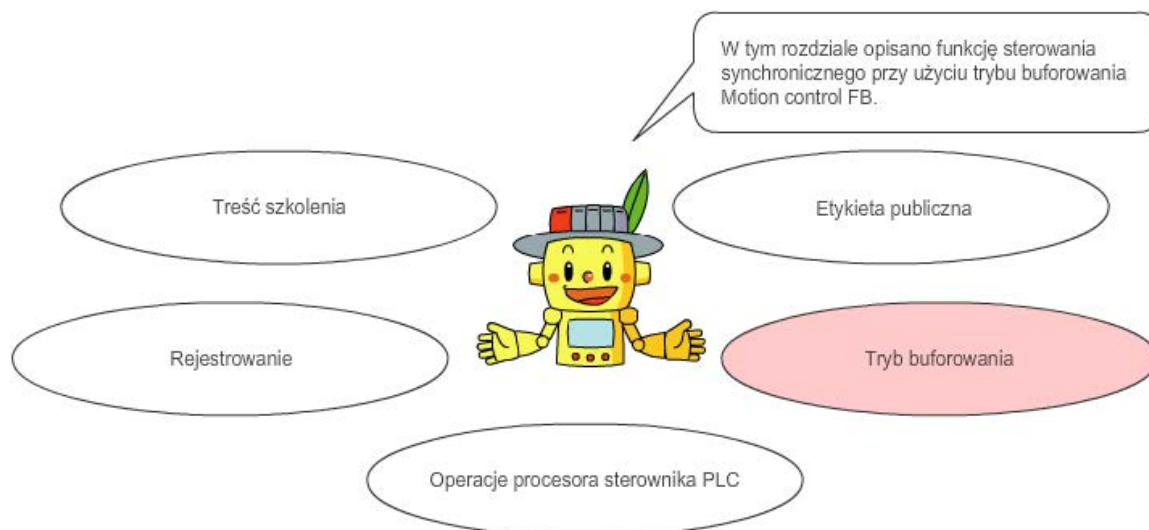
Szkolenie to jest kontynuacją szkolenia Moduł ruchu MELSEC iQ-R – informacje podstawowe (Pierwsze uruchomienie RD78G(H)). Przed przystąpieniem do tego szkolenia należy ukończyć szkolenie z zakresu uruchamiania.

Celem niniejszego szkolenia jest rozwinięcie wiedzy i zrozumienia na temat sterowania pozycjonowaniem systemu sterowania ruchem z wykorzystaniem modułu ruchu serii MELSEC iQ-R.



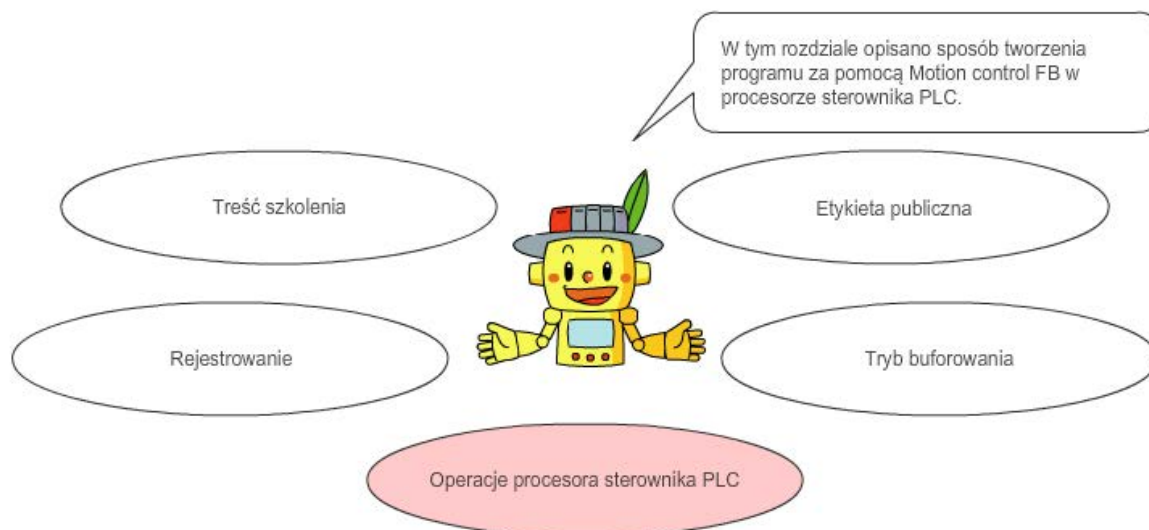
Szkolenie to jest kontynuacją szkolenia Moduł ruchu MELSEC iQ-R – informacje podstawowe (Pierwsze uruchomienie RD78G(H)). Przed przystąpieniem do tego szkolenia należy ukończyć szkolenie z zakresu uruchamiania.

Celem niniejszego szkolenia jest rozwinięcie wiedzy i zrozumienia na temat sterowania pozycjonowaniem systemu sterowania ruchem z wykorzystaniem modułu ruchu serii MELSEC iQ-R.



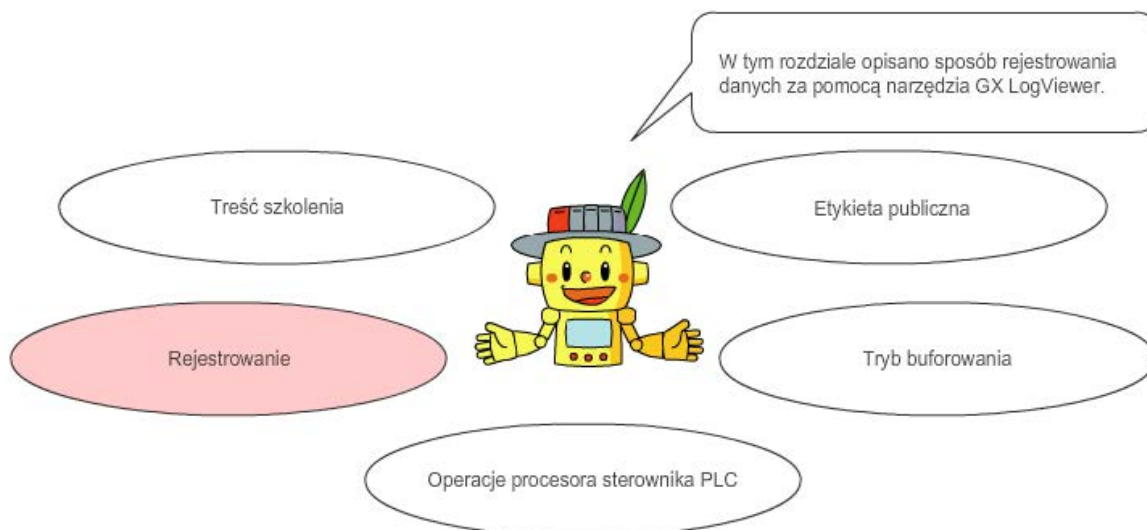
Szkolenie to jest kontynuacją szkolenia Moduł ruchu MELSEC iQ-R – informacje podstawowe (Pierwsze uruchomienie RD78G(H)). Przed przystąpieniem do tego szkolenia należy ukończyć szkolenie z zakresu uruchamiania.

Celem niniejszego szkolenia jest rozwinięcie wiedzy i zrozumienia na temat sterowania pozycjonowaniem systemu sterowania ruchem z wykorzystaniem modułu ruchu serii MELSEC iQ-R.



Szkolenie to jest kontynuacją szkolenia Moduł ruchu MELSEC iQ-R – informacje podstawowe (Pierwsze uruchomienie RD78G(H)). Przed przystąpieniem do tego szkolenia należy ukończyć szkolenie z zakresu uruchamiania.

Celem niniejszego szkolenia jest rozwinięcie wiedzy i zrozumienia na temat sterowania pozycjonowaniem systemu sterowania ruchem z wykorzystaniem modułu ruchu serii MELSEC iQ-R.



Szkolenie to jest kontynuacją szkolenia Moduł ruchu MELSEC iQ-R – informacje podstawowe (Pierwsze uruchomienie RD78G(H)). Przed przystąpieniem do tego szkolenia należy ukończyć szkolenie z zakresu uruchamiania.

Program szkolenia przedstawiono poniżej.
Zalecamy rozpoczęcie szkolenia od rozdziału 1.

Rozdział 1 – Treść szkolenia

W tym rozdziale opisano treść szkolenia.

Rozdział 2 – Etykieta publiczna

W tym rozdziale opisano etykietę publiczną, która służy do wymiany informacji o urządzeniu między procesorem sterownika PLC a modułem ruchu.

Rozdział 3 – Tryb buforowania

W tym rozdziale opisano funkcję sterowania synchronicznego przy użyciu trybu buforowania Motion control FB.

Rozdział 4 – Praca z procesorem sterownika PLC





W tym rozdziale opisano sposób tworzenia programu za pomocą Motion control FB w procesorze sterownika PLC.

Rozdział 5 – Rejestrowanie

W tym rozdziale opisano sposób rejestrowania danych za pomocą narzędzia GX LogViewer.

Test końcowy

Łącznie 4 części (7 pytań)

Przejdź do następnej strony		Przejdź do następnej strony.
Wróć do poprzedniej strony		Wróć do poprzedniej strony.
Przejdź do żądanej strony		Wyświetli się „Spis treści”, umożliwiający przejście do żądanej strony.
Zakończ naukę		Zakończ naukę. Okna takie jak okno „Zawartość” zostaną zamknięte i nauka zostanie zakończona.

■ Środki bezpieczeństwa

W przypadku korzystania z rzeczywistych produktów do celów edukacyjnych należy uważnie zapoznać się z częścią „Środki ostrożności” w instrukcji obsługi produktu i zwracać szczególną uwagę na bezpieczeństwo i prawidłowe użytkowanie.

■ Środki ostrożności dotyczące szkolenia


Zrzuty ekranu pokazywane w niniejszym szkoleniu mogą różnić się od rzeczywistego oprogramowania użytkownika w zależności od posiadanej wersji. W szkoleniu wykorzystywane są następujące wersje oprogramowania. Najnowszą wersję oprogramowania możesz pobrać ze strony internetowej Mitsubishi Electric FA.

MELSOFT GX Works3	Ver.1.066U	Motion Control Setting function	Ver.1.012N
GX LogViewer	Ver.1.106K		
MELSOFT MR Configurator2	Ver.1.110Q lub nowsza		

Oprogramowanie firmware wgrane do procesora sterownika PLC musi być w wersji 44 lub nowszej (w przypadku RD78GH– w wersji 46 lub nowszej).

Oprogramowanie firmware wgrane do modułu sterowania ruchem musi być w wersji 10 lub nowszej.

Informacje na temat aktualizacji wersji oprogramowania układowego można znaleźć na stronie internetowej MITSUBISHI ELECTRIC FA lub w instrukcji konfiguracji modułu.

Ikona  wskazuje podręcznik referencyjny.

Zawartość podręczników wspomnianych w niniejszym szkoleniu pochodzi z podanych poniżej wersji.

W przypadku innej wersji lokalizacja i zawartość podręcznika mogą być inne.

Nazwa podręcznika	Nr podręcznika	Wersja
MELSEC iQ-R Motion Module User's Manual (Startup)	IB-0300406	E
MELSEC iQ-R Motion Module User's Manual (Application)	IB-0300411	E
MELSEC iQ-R Motion Module User's Manual (Network)	IB-0300426	E
MELSEC iQ-R Programming Manual (Motion Module Instructions, Standard Functions/Function Blocks)	IB-0300431	E
MELSEC iQ-R Programming Manual (Motion Control Function Blocks)	IB-0300533	C
MELSEC iQ-R Structured Text (ST) Programming Guide Book	SH-081483	F
MELSEC iQ-R Programming Manual (CPU Module Instructions, Standard Functions/Function Blocks)	SH-081266	Z
MELSEC iQ-R CPU Module User's Manual (Application)	SH-081264	AK

Poniżej przedstawiono program szkolenia.

Rozdział 1 – Treść szkolenia

W tym rozdziale opisano treść szkolenia.



Rozdział 2 – Etykieta publiczna

W tym rozdziale opisano etykietę publiczną, która służy do wymiany informacji o urządzeniu między procesorem sterownika PLC a modułem ruchu.



Rozdział 3 – Tryb buforowania

W tym rozdziale opisano funkcję sterowania synchronicznego przy użyciu trybu buforowania Motion control FB.



Rozdział 4 – Praca z procesorem sterownika PLC

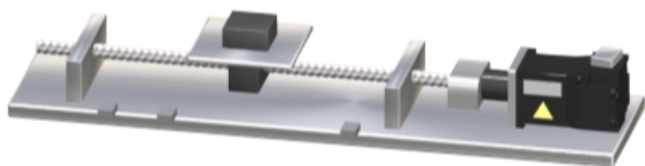
W tym rozdziale opisano sposób tworzenia programu za pomocą Motion control FB w procesorze sterownika PLC.



Rozdział 5 – Rejestrowanie

W tym rozdziale opisano sposób rejestrowania danych za pomocą narzędzia GX LogViewer w celu sprawdzenia działania modułu ruchu.

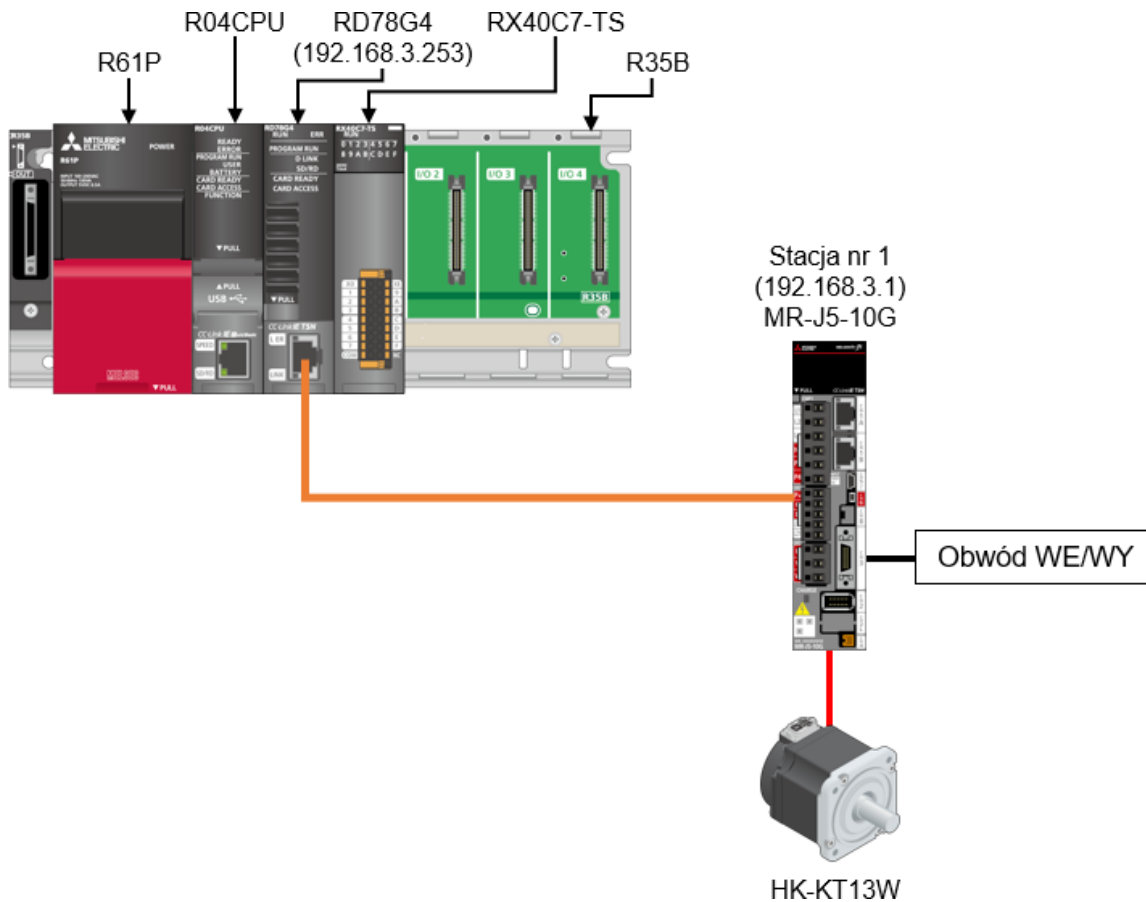
W tym szkoleniu wykorzystywany jest ten sam jednowałowy mechanizm śruby kulowej, co w szkoleniu z zakresu pierwszego uruchomienia.



Konfiguracja systemu docelowego jest następująca.

Odłącz zdalny moduł wejściowy z systemu wykorzystanego w szkoleniu z zakresu pierwszego uruchomienia i podłącz moduł wejściowy RX40C7-TS do gniazda 1 płyty bazowej sterownika PLC.

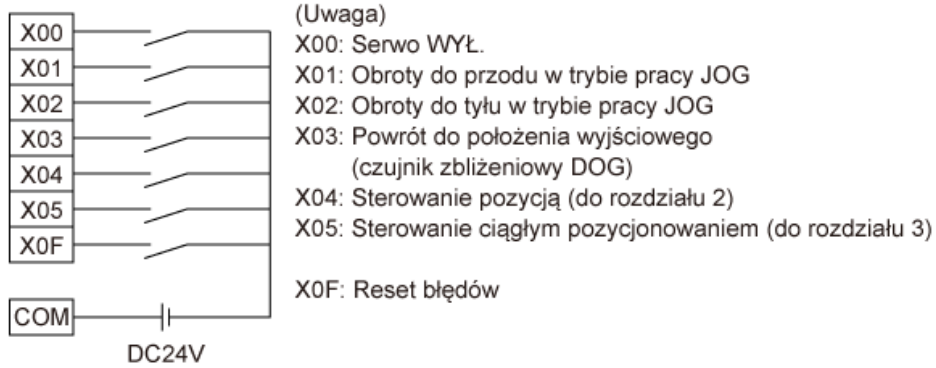
Numer stacji serwowzmacniacza MR-J5-10G został zmieniony na 1, a adres IP uległ zmianie na 192.168.3.1.



Okablowanie zasilania sterownika programowalnego i serwowzmacniacza oraz sposób podłączania serwomotoru są takie same, jak w przypadku szkolenia z zakresu pierwszego uruchomienia.

Poniżej przedstawiono okablowanie obwodu zewnętrznego modułu wejściowego.

RX40C7-TS



(Uwaga) Jako że adres startowy modułu WE/WY RX40C7-TS to 0020H, w programie używane są od X20 do X25 oraz X2F.

Gdy moduł ruchu jestysterowywany przez moduł wejściowy sterownika programowalnego (jak w przypadku systemu użytego w tym szkoleniu w rozdziale 1), procesor sterownika PLC i moduł ruchu muszą wymieniać informacje o urządzeniu.

Istnieją następujące dwie metody.

1. Używanie etykiet publicznych.
2. Używanie pamięci buforowej modułu ruchu.

W tym rozdziale opisano sposób wymiany danych przy użyciu etykiet publicznych.

Pobierz przykładowy program do wykorzystania w tym rozdziale i rozdziale 3, klikając poniższe łącze.

[RD78GBasic2_sample1.zip \(1.34MB\)](#)

[Wskazówka]

W przypadku korzystania z pamięci buforowej skopiuj dane do wymiany do obszaru użytkownika (U0\G11478000 do G11997999).

(Przykładowy program)

<Procesor sterownika PLC>

Uruchomienie pozycjonowania



<Moduł ruchu>

MC_MoveAbsolute_1(

Execute:= G11478000.0 ,

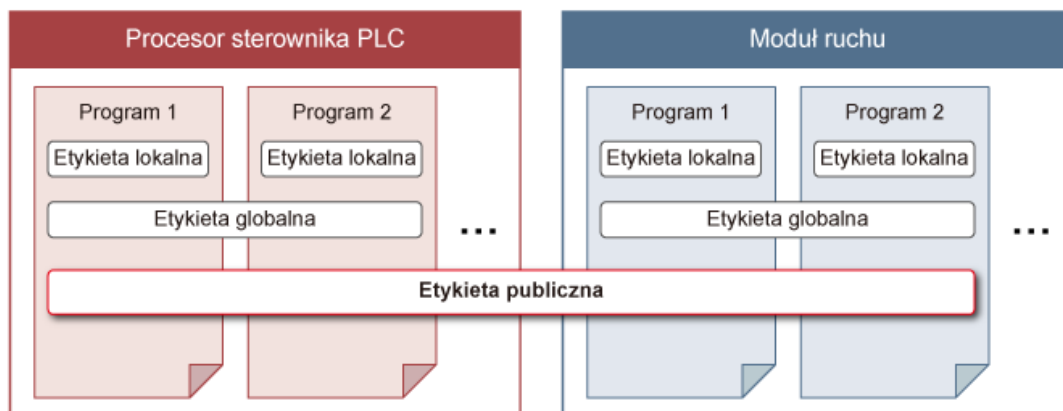
:

:

);

Etykieta publiczna to współdzielona etykieta, której można używać zarówno w module ruchu, jak i w procesorze sterownika PLC.

Poniżej przedstawiono obszary zastosowań etykiety lokalnej, etykiety globalnej i etykiety publicznej.



(1) Rejestrowanie etykiet publicznych

Zarejestruj etykiety publiczne z etykiet globalnych modułu ruchu.

Upewnij się, że kolumna „Public Label” jest widoczna w edytorze etykiet globalnych na ekranie Motion Module Setting Function.

Ustaw etykiety, które mają zostać zarejestrowane jako publiczne, na „Enabled”.

Spowoduje to aktywację kolumny „Motion Control Attribute”.

Dla każdej etykiety wybierz, czy ma być odczytywana, czy zapisywana z/do procesora sterownika PLC.

	Label Name	Data Type	Class	Initial	Constant	Japanese	English(Display Target)	Chinese	Remark	Public Label	Motion Control Attribute
1	G_bSVONCMD	Bit	VAR_GLOBAL				Servo ON			Enabled	WRITE (=> Motion)
2	G_leJogVelocity	FLOAT [Double Precision]	VAR_GLOBAL				JOG Velocity			Enabled	WRITE (=> Motion)
3	G_bJogFwd	Bit	VAR_GLOBAL				JOG Forward			Enabled	WRITE (=> Motion)
4	G_bJogBwd	Bit	VAR_GLOBAL				JOG Backward			Enabled	WRITE (=> Motion)
5	G_bJogBusy	Bit	VAR_GLOBAL				JOG Busy			Enabled	READ (Motion =>)
6	G_lePosition0	FLOAT [Double Precision]	VAR_GLOBAL				Position0 Address			Disabled	-
7	G_bHomingCMD	Bit	VAR_GLOBAL				Homing Command			Enabled	WRITE (=> Motion)
8	G_bHomingDone	Bit	VAR_GLOBAL				Homing Done			Enabled	READ (Motion =>)
9	G_bHomingReq	Bit	VAR_GLOBAL				Homing Request			Enabled	READ (Motion =>)
10	G_bPosCMD	Bit	VAR_GLOBAL				Positioning Command			Enabled	WRITE (=> Motion)
11	G_bPosDone	Bit	VAR_GLOBAL				Positioning Done			Enabled	READ (Motion =>)
12	G_bPosReq	Bit	VAR_GLOBAL				Positioning Start Request			Enabled	READ (Motion =>)
13	G_bErrorReset	Bit	VAR_GLOBAL				Error Reset			Enabled	WRITE (=> Motion)
14	G_bContPosCMD	Bit	VAR_GLOBAL				Continuous Positioning Command			Enabled	WRITE (=> Motion)
15	G_bContPosReq	Bit	VAR_GLOBAL				Continuous Positioning Start Request			Enabled	WRITE (=> Motion)
16	G_bContPosDone	Bit	VAR_GLOBAL				Continuous Positioning Done			Enabled	READ (Motion =>)
17											

[Wskazówka]

Jeśli kolumna etykiety publicznej nie jest widoczna, przewiń tabelę w prawo.

(2) Typy danych, które można zarejestrować jako etykietę publiczną

W poniższej tabeli przedstawiono typy danych, które można zarejestrować jako etykietę publiczną.

Typ zmiennej	Typ	Wybór tablicy	Ustawienie etykiety publicznej	Uwagi
Etykieta globalna	Typ prosty	Nie	○	Ustawienia nie są możliwe dla następujących etykiet i klas. ■Etykieta <ul style="list-style-type: none"> Etykieta typu - ciąg znaków (String) Etykieta typu timer (Timer) Etykieta typu licznik (Counter) Etykieta typu długi licznik (Long counter) Etykieta typu timer retencyjny (Retentive timer) Etykieta typu długi timer retencyjny (Long Retentive timer) Etykieta typu długi timer (Long Timer) ■Klasa <ul style="list-style-type: none"> Klasa VAR_GLOBAL_CONSTANT
		Tak	△(uwaga 1, 2)	
	Strukturalny typ danych	Nie	△(uwaga 3)	
		Tak	△(uwaga 1, 2, 4, 5)	
	FB (w tym Motion control FB)	Nie	×	
		Tak	×	
Program	-	-	×	
Etykieta lokalna bloku programu	-	-	×	
Strukturalny typ danych	-	-	△(uwaga 3,5)	
Motion control FB – dane strukturalne	-	-	△(uwaga 6,7)	

(Uwaga)

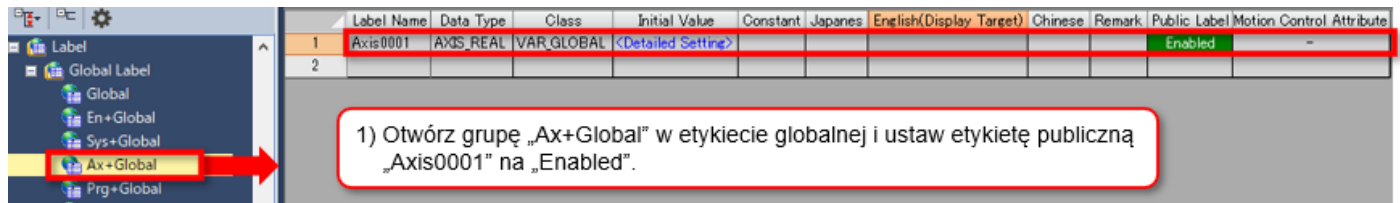
1. Ustawienia etykiety publicznej nie można skonfigurować dla każdego elementu tablicy.
2. Gdy używana jest tablica bitowa, etykiety publicznej nie można ustawić na „Enabled”. (W przypadku danych strukturalnych tylko konkretnego członu nie można ustawić na „Enabled”).
3. Gdy jako człon danych strukturalnych używany jest ciąg znaków, członu tego nie można ustawić na „Enabled”.
4. Dane strukturalne z maksymalnie czterema warstwami można ustawić na publiczne.
5. Gdy jako człon danych strukturalnych używana jest tablica danych strukturalnych, członu tego nie można ustawić na „Enabled”.
6. Mogą zostać wykorzystane w programie PLCopen Motion control FB przez procesor.
7. Gdy w danych strukturalnych Motion control FB używany jest ciąg znaków, nie można ustawić samego typu danych strukturalnych Motion control FB.

(3) Rejestrowanie danych strukturalnych jako etykiety publicznej

Aby ustawić człony danych strukturalnych przygotowane w systemie, np. dane z monitoringami osi, jako etykietę publiczną, należy zarejestrować etykiety publiczne według warstwy danych strukturalnych w sposób przedstawiony poniżej.

Niniejsze szkolenie opisuje sposób rejestrowania atrybutów Set Position (SetPosition) i Set Velocity (SetVelocity), dane z monitoringami (Md) rzeczywistej osi napędowej (Axis_Real), jako etykiety publiczne.

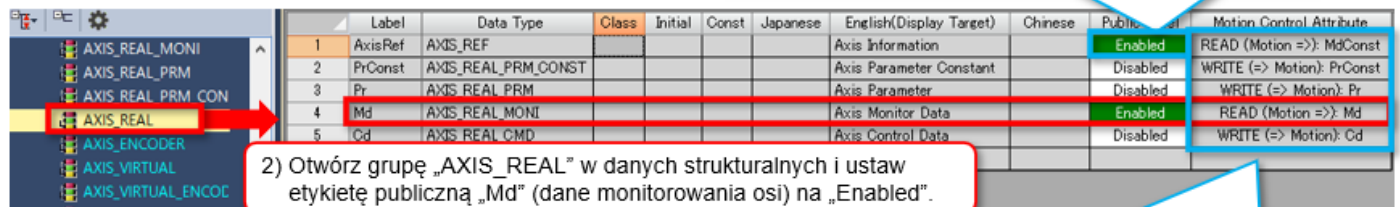
[Ustawianie AxisName.Md.SetPosition (bieżąca pozycja polecenia) i AxisName.Md.SetVelocity (bieżąca prędkość polecenia) na etykietę publiczną]



Label Name	Data Type	Class	Initial Value	Constant	Japanese	English(Display Target)	Chinese	Remark	Public Label	Motion Control Attribute
1	Axis0001	AXIS_REAL	VAR_GLOBAL	<Detailed Setting>					Enabled	-
2										

1) Otwórz grupę „Ax+Global” w etykiecie globalnej i ustaw etykietę publiczną „Axis0001” na „Enabled”.

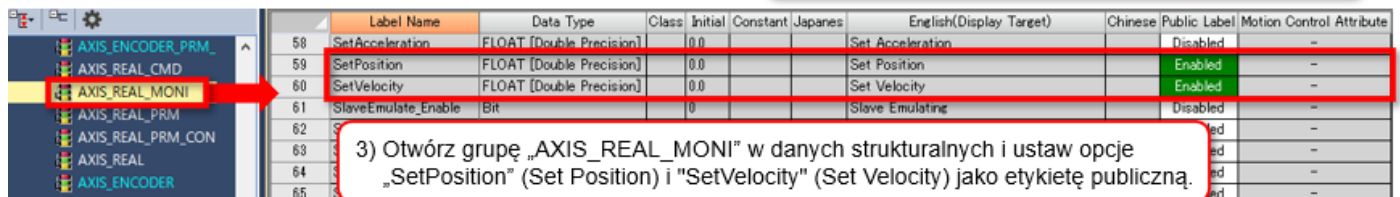
Etykieta AxisRef jest domyślnie ustawiona na „Enabled”.



Label	Data Type	Class	Initial	Const	Japanese	English(Display Target)	Chinese	Public Label	Motion Control Attribute
1	AxisRef	AXIS_REF				Axis Information		Enabled	READ (Motion =>): MdConst
2	PrConst	AXIS_REAL_PRM_CONST				Axis Parameter Constant		Disabled	WRITE (=> Motion): PrConst
3	Pr	AXIS_REAL_PRM				Axis Parameter		Disabled	WRITE (=> Motion): Pr
4	Md	AXIS_REAL_MONI				Axis Monitor Data		Enabled	READ (Motion =>): Md
5	Cd	AXIS_REAL_CMD				Axis Control Data		Disabled	WRITE (=> Motion): Cd

2) Otwórz grupę „AXIS_REAL” w danych strukturalnych i ustaw etykietę publiczną „Md” (dane monitorowania osi) na „Enabled”.

Atrybut Motion Control danych strukturalnych jest stały.



Label Name	Data Type	Class	Initial	Constant	Japanese	English(Display Target)	Chinese	Public Label	Motion Control Attribute
58	SetAcceleration	FLOAT [Double Precision]	0.0			Set Acceleration		Disabled	-
59	SetPosition	FLOAT [Double Precision]	0.0			Set Position		Enabled	-
60	SetVelocity	FLOAT [Double Precision]	0.0			Set Velocity		Enabled	-
61	SlaveEmulate_Enable	Bit	0			Slave Emulating		Disabled	-
62									
63									
64									
65									

3) Otwórz grupę „AXIS_REAL_MONI” w danych strukturalnych i ustaw opcje „SetPosition” (Set Position) i „SetVelocity” (Set Velocity) jako etykietę publiczną.

(4) Odzwierciedlenie etykiet publicznych

Wybierz z menu kolejno [Convert] → [Rebuild All].

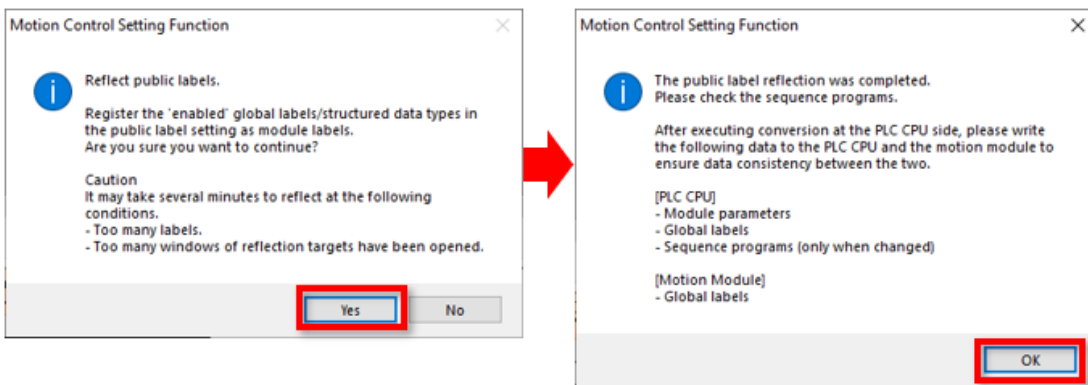
Wolna pojemność etykiety publicznej jest wyświetlana jako informacja w oknie danych wyjściowych.

No.	Result	Data Name	Category	Content	Error Code
1	Information	Public Label	Free Volume	99.88[%] (32728 [Word] = 32768 [Word] - (Global: 40 [Word]))	-

Po pomyślnym zakończeniu procesu przebudowy wszystkich elementów wybierz z menu kolejno [Convert] → [Reflect Public Labels].

W wyświetlonym oknie kliknij przycisk [Yes].

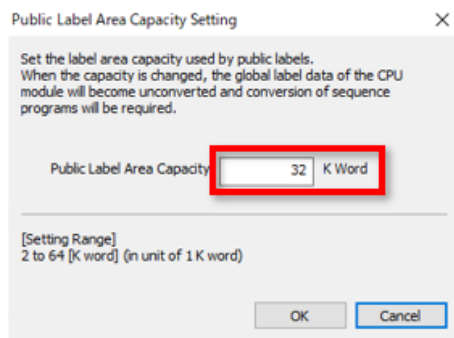
Gdy pojawi się komunikat dotyczący pomyślnego odzwierciedlenia etykiet publicznych, kliknij przycisk [OK].



(Uwaga) Pojemność pamięci, której można użyć do rejestracji etykiet publicznych, wynosi domyślnie 32 000 słów.

Pojemność można zwiększyć do 64 000 słów.

Aby zmienić pojemność, ustaw rozmiar pamięci, wybierając z menu kolejno [Convert] → [Public Label Capacity Setting].



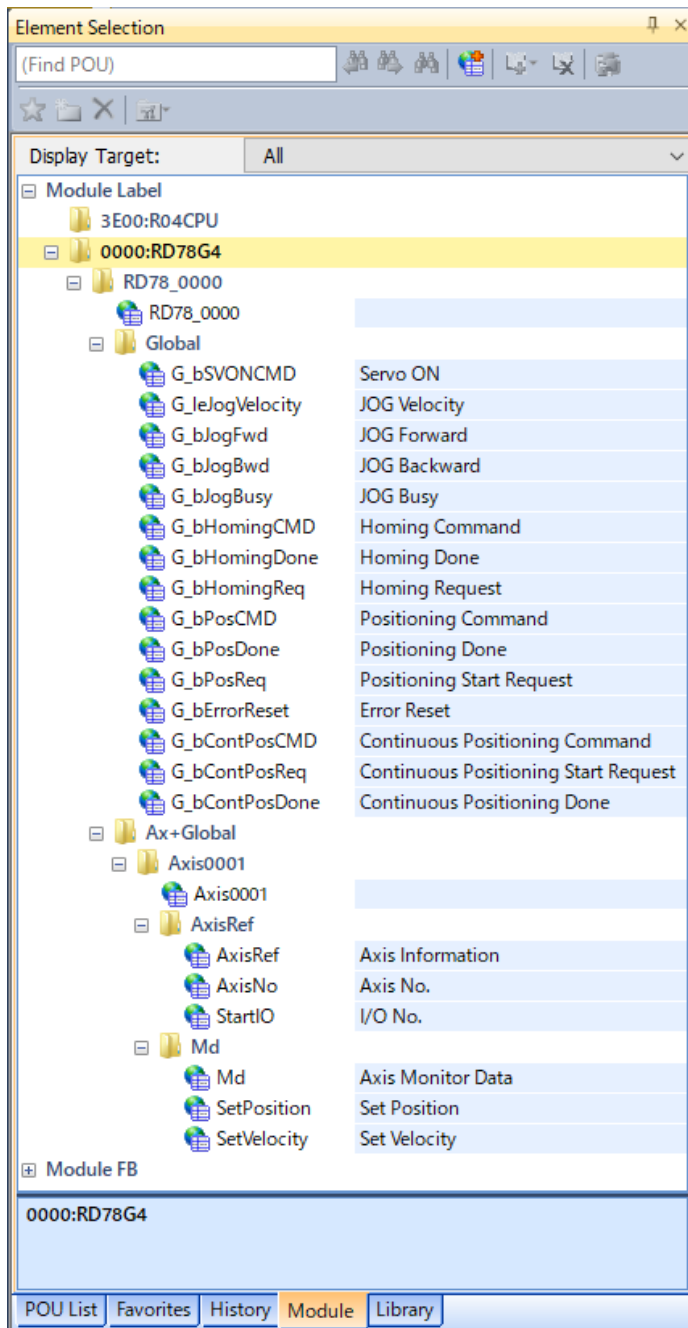
(5) Sprawdzanie etykiet po stronie procesora sterownika PLC

Odzwiedziane etykiety publiczne są rejestrowane w etykiecie modułu po stronie procesora sterownika PLC.

Wybierz etykietę modułu w oknie Element Selection narzędzia GX Works3, a następnie sprawdź, czy etykiety publiczne zostały zarejestrowane pod numerem [0000:RD78G4] w obszarze [Module Label].

Po zmianie ustawienia etykiety publicznej zawsze należy ponownie wykonać polecenie „Reflect Public Labels”.

Używając etykiet publicznych w procesorze sterownika PLC, przebuduj wszystkie programy.

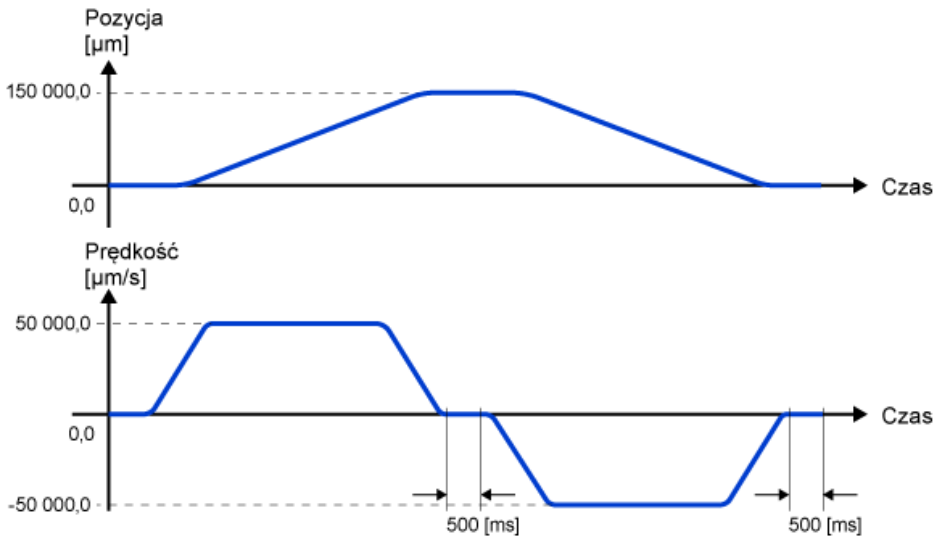


(1) Operacje przykładowego programu

Sygnały wejściowe przykładowego programu używanego w tym rozdziale są przypisane w następujący sposób.

Wejście	Operacja
X20	Serwo wył. (uwaga)
X21	Obrót do przodu w trybie pracy JOG
X22	Obroty do tyłu w trybie pracy JOG
X23	Powrót do pozycji wyjściowej
X24	Sterowanie pozycjonowaniem
X25	Sterowanie pozycją w sposób ciągły (rozdział 3)

Poniżej przedstawiono wzorzec operacji X24: sterowanie pozycjonowaniem.

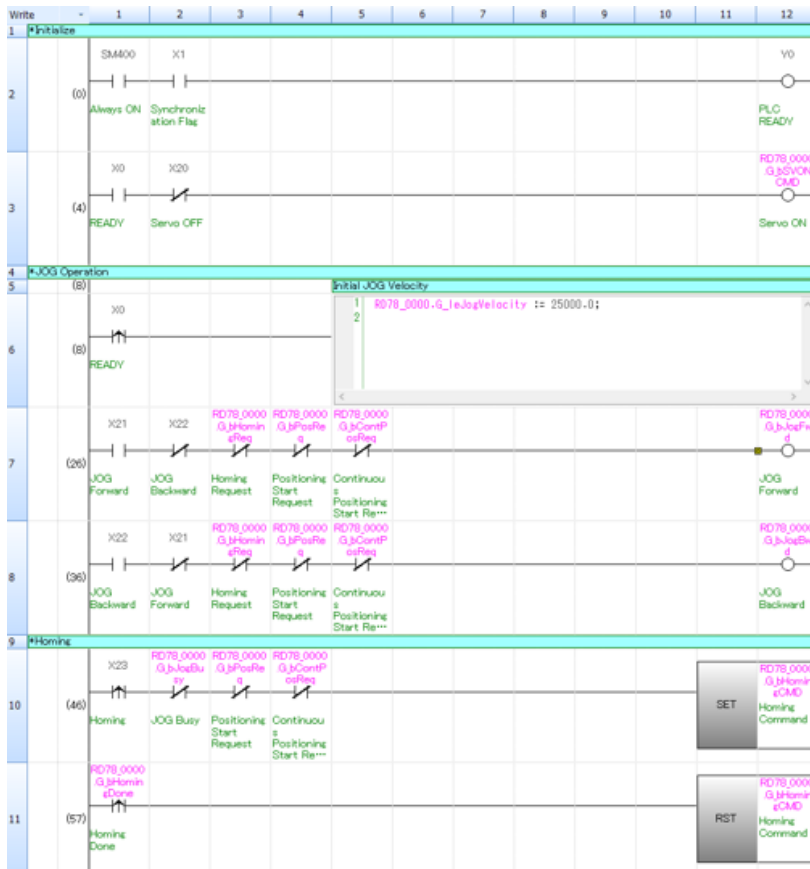


(Uwaga) Ten przykładowy program wykonuje polecenie włączenia serwo automatycznie, gdy procesor sterownika PLC jest ustawiony na RUN.

Gdy zasilanie jest włączone przy włączonych sygnałach uruchomienia, serwomotor może zostać uruchomiony.

(2) Program procesora sterownika PLC

1) MAIN (drabinka, program skanujący)



Y0 jest włączany jako pierwszy.

Po włączeniu X0 następuje włączenie serwa. Włącz X20, aby wyłączyć serwo.

Ustaw wartość początkową prędkości JOG.

Ta część programu wykorzystuje język programowania ST.

Jako że atrybut sterowania ruchem etykiety globalnej „G_JogVelocity”, która przechowuje prędkość JOG, jest ustawiony na „WRITE (→Motion)”, wartość liczbowa należy zadać z procesora sterownika PLC.

Włącza sygnał uruchomienia operacji JOG.

Zapobiega uruchomieniu obrotów do przodu i do tyłu w tym samym czasie.

Ustawiana jest blokada uniemożliwiająca uruchomienie operacji JOG podczas wykonywania innego programu.

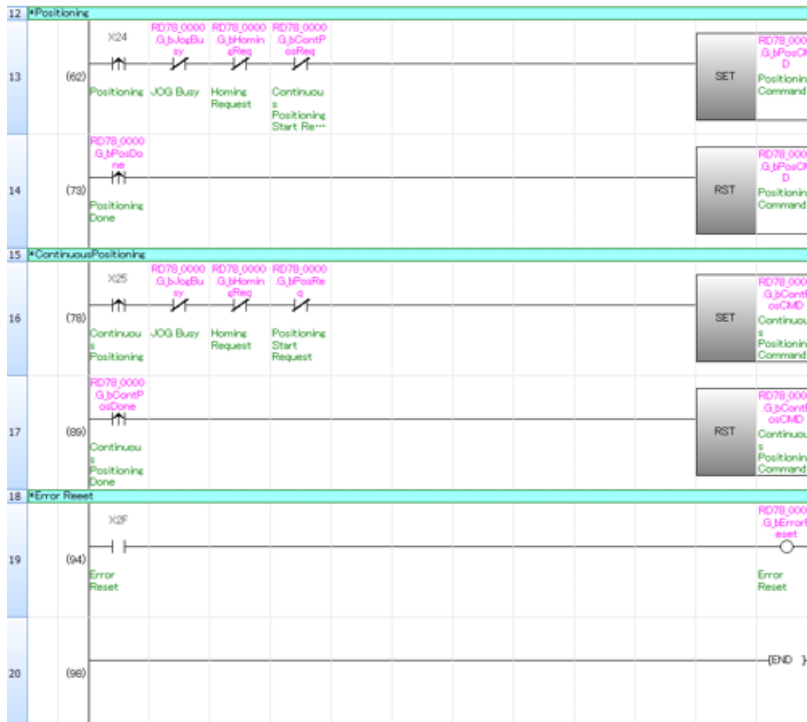
Uruchomienie powrotu do położenia wyjściowego (X23) jest podtrzymane przez etykietę publiczną G_bHomingCMD i wysyłane do modułu ruchu jako warunek uruchomienia powrotu do położenia wyjściowego.

Ustawiana jest blokada uniemożliwiająca uruchomienie powrotu do położenia wyjściowego podczas wykonywania innego programu. Etykieta G_bHomingCMD jest resetowana na nastające zboczę sygnału potwierdzającego ukończenie bazowania.

(2) Program procesora sterownika PLC

1) Kontynuacja części MAIN (drabinka, program skanujący)

(Kontynuacja poprzedniej strony)



Narastające zbocze sygnału uruchomienia sterowania pozycjonowaniem (X24) jest podtrzymywane przez G_bPosCMD i wysyłane do modułu ruchu jako warunek uruchomienia sterowania pozycjonowaniem. Ustawiana jest blokada uniemożliwiająca uruchomienie sterowania pozycjonowaniem podczas wykonywania innego programu. Etykieta G_bPosCMD jest resetowana na narastające zbocze sygnału potwierdzającego ukończenie pozycjonowania.

Program uruchomienia pozycjonowania w sposób ciągłym jest opisany w rozdziale 3. Szczegółowe informacje można znaleźć w punkcie 3.4.

Błędy są resetowane po włączeniu X2F.

(2) Program procesora sterownika PLC

2) MONITOR (ST, program skanujący)

SetPosition (Set Position) i SetVelocity (Set Velocity) monitora osi ustawionego jako etykieta publiczna są przechowywane w rejestrach D0 i D2.

Jako że SetPosition i SetVelocity są liczbami rzeczywistymi o podwójnej precyzji, są one konwertowane na typ słowa podwójnego ze znakiem, aby mogły być łatwo obsługiwane przez procesor sterownika PLC. (Uwaga)

Choć te środki nie są używane w tym szkoleniu, są one używane do wyświetlania danych w innych programach sekwencyjnych i GOT, a także do innych celów.

```
1 D0:D := LREAL_TO_DINT(RD78_0000.Axis0001.Md.SetPosition);  
2 D2:D := LREAL_TO_DINT(RD78_0000.Axis0001.Md.SetVelocity);  
3
```

Rejstr podwójny z znakiem określamy w sposób „D0:D”.

(Uwaga) Gdy liczba rzeczywista o podwójnej precyzji jest konwertowana na słowo podwójne ze znakiem, jeśli konwertowana wartość wykracza poza zakres od -2147483648 do 2147483647, wystąpi błąd obliczeń.

(3) Program modułu ruchu

1) ServoON_JOG (normalny typ wykonania)

```

1 //-----Servo ON-----
2 MC_Power_1(
3     Axis      := Axis0001.AxisRef,
4     Enable    := TRUE,
5     ServoON   := G_bSVONCMD,
6     Busy      => bPowerBusy
7 );
8
9 //-----JOG Operation-----
10 //Initial Value Setting
11 IF (bPowerBusy) THEN
12     leJogAcceleration := 50000.0;
13     leJogDeceleration := 50000.0;
14     leJogJerk          := 0.0;
15 END_IF;
16
17 //JOG
18 MCv_Jog_1(
19     Axis      := Axis0001.AxisRef,
20     JogForward := G_bJogFwd,
21     JogBackward := G_bJogBwd,
22     Velocity   := G_leJogVelocity,
23     Acceleration := leJogAcceleration,
24     Deceleration := leJogDeceleration,
25     Jerk       := leJogJerk,
26     Busy       => G_bJogBusy
27 );
28

```

Sygnal ServoON załączany jest (G_bSVONCMD) z procesora sterownika PLC i załącza on serwo.

Wpisuje wartości przyspieszenia JOG, spowolnienia JOG i ruchu z szarpnięciem JOG do etykiet, gdy wyjście zajętości MC_Power_1 jest włączone.

Odbiera sygnały uruchomienia i prędkości JOG z procesora sterownika PLC.

Zwraca wyjście zajętości do procesora sterownika PLC.

(Uwaga) W tym przykładowym programie pominięto sygnały WE/WY FB, które nie są używane lub które nie zostały zmienione w stosunku do wartości początkowych.

- (3) Program modułu ruchu
 2) Powrót do położenia wyjściowego (normalny typ wykonania)

```

1 //-----Homing Operation-----
2 //Initial Value Setting, Operation Start Request
3 IF G_bHomingCMD THEN
4   G_lePosition0 := 0.0 ;
5   G_bHomingReq := TRUE ;
6 ELSE
7   G_bHomingReq := FALSE ;
8 END_IF;
9
10 //Homing
11 MC_Home_1(
12   Axis           := Axis0001.AxisRef ,
13   Execute        := G_bHomingReq,
14   Position       := G_lePosition0 ,
15   Done           => bHomingDone ,
16   Busy           => G_bHomingBusy ,
17   CommandAborted => bHomingAborted ,
18   Error          => bHomingError
19 );
20
21 //Done Signal => PLC CPU
22 G_bHomingDone := bHomingDone OR bHomingAborted OR bHomingError;
23

```

Odbiera sygnał powrotu do położenia wyjściowego (G_bHomingCMD) z procesora sterownika PLC. Wpisuje adres położenia wyjściowego do etykiety i włącza żądanie powrotu do położenia wyjściowego (G_bHomingReq).

Wyłącza G_bHomingReq po wyłączeniu G_bHomingCMD.

Zwraca sygnał zakończenia wykonywania do procesora sterownika PLC po pomyślnym zakończeniu powrotu do położenia wyjściowego (wyjście Done wł.), przerwaniu wykonywania (wyjście CommandAborted wł.) lub wystąpieniu błędu (wyjście Error wł.).

(3) Program modułu ruchu

3) Pozycjonowanie (normalny typ wykonania)

```

1 //-----Positioning Operation-----
2 //Initial Value Setting, Operation Start Request
3 IF G_bPosCMD THEN
4     lePosition1      := 150000.0;
5     lePosVelocity    := 50000.0;
6     lePosAcceleration := 100000.0;
7     lePosDeceleration := 100000.0;
8     lePosJerk        := 200000.0;
9     G_bPosReq := TRUE;
10 ELSE
11     G_bPosReq := FALSE;
12 END_IF;
13
14 //Positioning1
15 MC_MoveAbsolute_1(
16     Axis      := Axis0001.AxisRef ,
17     Execute   := G_bPosReq ,
18     Position  := lePosition1 ,
19     Velocity   := lePosVelocity ,
20     Acceleration := lePosAcceleration ,
21     Deceleration := lePosDeceleration ,
22     Jerk      := lePosJerk ,
23     Direction := MC_DIRECTION__mcShortestWay ,
24     BufferMode := MC_BUFFER_MODE_mcAborting ,
25     Done      => bMoveAbs1Done ,
26     CommandAborted => bMoveAbs1Aborted ,
27     Error     => bMoveAbs1Error
28 );
29 //Dwell
30 TON_1(IN:= bMoveAbs1Done ,PT:= T#500ms ,Q=> bDwell1_out );
31 //Positioning2
32 MC_MoveAbsolute_2(
33     Axis      := Axis0001.AxisRef ,
34     Execute   := bDwell1_out ,
35     Position  := G_lePosition0 ,
36     Velocity   := lePosVelocity ,
37     Acceleration := lePosAcceleration ,
38     Deceleration := lePosDeceleration ,
39     Jerk      := lePosJerk ,
40     Direction := MC_DIRECTION__mcShortestWay ,
41     BufferMode := MC_BUFFER_MODE_mcAborting ,
42     Done      => bMoveAbs2Done ,
43     CommandAborted => bMoveAbs2Aborted ,
44     Error     => bMoveAbs2Error
45 );
46 //Dwell
47 TON_2(IN:= bMoveAbs2Done ,PT:= T#500ms ,Q=> bDwell2_out );
48
49 //Error Signal, Aborted Signal
50 bPosError := bMoveAbs1Error OR bMoveAbs2Error;
51 bPosAborted := bMoveAbs1Aborted OR bMoveAbs2Aborted;
52
53 //Done Signal => PLC CPU
54 G_bPosDone := bDwell2_out OR bPosError OR bPosAborted;
55

```

Odbiera sygnał uruchomienia sterowania pozycjonowaniem (G_bPosCMD) z procesora sterownika PLC. Dane wymagane do pozycjonowania są wpisywane do etykiety i włączone jest żądanie uruchomienia sterowania pozycjonowaniem (G_bPositioningReq).

Wyłącza G_bPositioningReq po wyłączeniu G_bPosCMD przez procesor sterownika PLC.

MC_MoveAbsolute wykonuje pozycjonowanie.

Wyjście Done w MC_MoveAbsolute_1 uruchamia timer wprowadzający czas oczekiwania.

Po upływie czasu oczekiwania MC_MoveAbsolute ponownie wykonuje pozycjonowanie.

Wyjście Done w MC_MoveAbsolute_2 uruchamia timer wprowadzający czas oczekiwania.

Zwraca sygnał zakończenia wykonywania do procesora sterownika PLC po upływie czasu oczekiwania lub po włączeniu wyjścia Error lub CommandAborted w MC_MoveAbsolute.

(3) Program modułu ruchu

4) ErrorReset (normalny typ wykonania)

```
1 //Axis Error Reset
2 MC_Reset_1(
3     Axis      := Axis0001.AxisRef ,
4     Execute   := G_bErrorReset
5 );
6
7 //System Error Reset
8 MCv_MotionErrorReset_1(
9     Execute   := G_bErrorReset
10 );
11
```

Odbiera sygnał resetu błędu (G_bErrorReset) z procesora sterownika PLC i wykonuje reset błędu osi oraz reset błędu systemu.

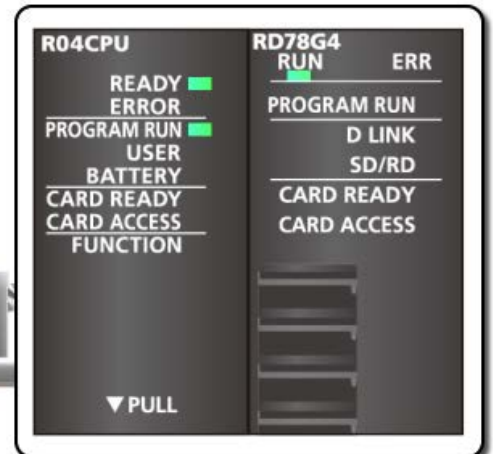
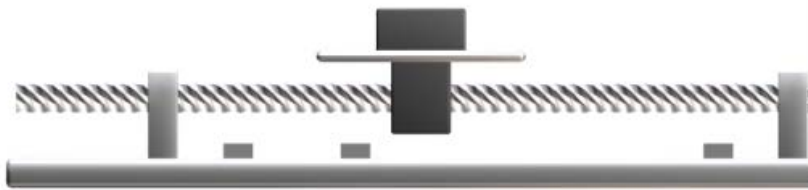
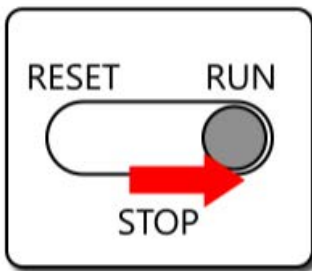
Zapisz program i parametry do procesora sterownika PLC i modułu ruchu.

- 1) Po przebudowaniu wszystkich programów w procesorze sterownika PLC wybierz kolejno [Online] → [Write to PLC] na pasku narzędzi GX Works3, aby zapisać wszystkie dane do procesora sterownika PLC.
- 2) Po zapisaniu parametrów do procesora sterownika PLC komunikacja z modułem ruchu zostaje włączona. Wybierz kolejno [Online] → [Write to Module] na pasku narzędzi Motion Control Setting Function, aby zapisać wszystkie dane do modułu ruchu.
- 3) Zresetuj procesor sterownika PLC, aby zakończyć operację zapisu.

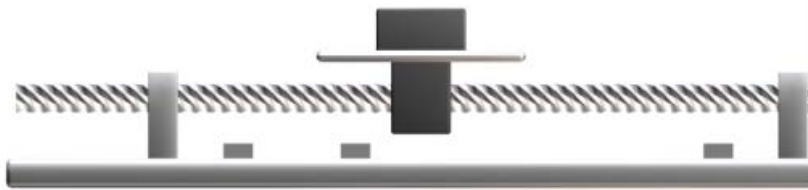
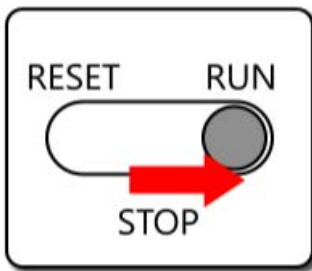
Kliknij przycisk odtwarzania w lewym dolnym rogu okna.



Sprawdź działanie przykładowego programu.
Przed rozpoczęciem pracy należy upewnić się, że programy procesora sterownika PLC i modułu ruchu zostały wgrane.



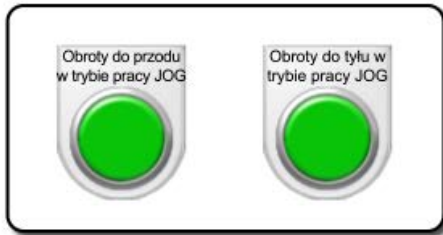
Ustaw przełącznik RUN/STOP/RESET procesora sterownika PLC w położeniu RUN.
Zapali się kontrolka READY i kontrolka PROGRAM RUN procesora sterownika PLC.
Zapali się kontrolka RUN modułu ruchu.



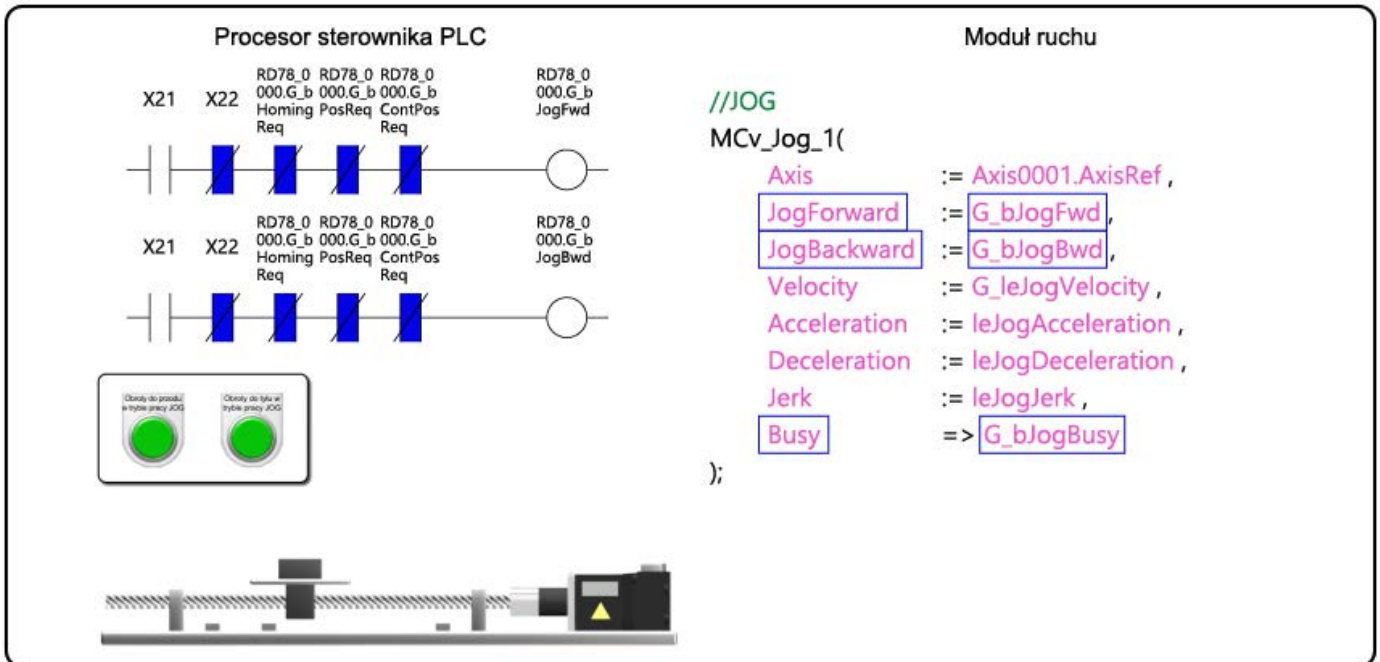
Poczekaj, aż zapali się kontrolka PROGRAM RUN modułu ruchu.
 Na serwowzmacniaczu wyświetlane jest wskazanie „r.01”.
 (Kropki świecą).
 Serwomotor przechodzi w stan włączenia.



Włącz X20, aby wykonać polecenie wyłączenia serwa.
Na serwowzmacniaczu wyświetlane jest wskazanie „r.01”.
(Kropki migają).
Wyłącz X20, aby ponownie wykonać polecenie włączenia serwa.



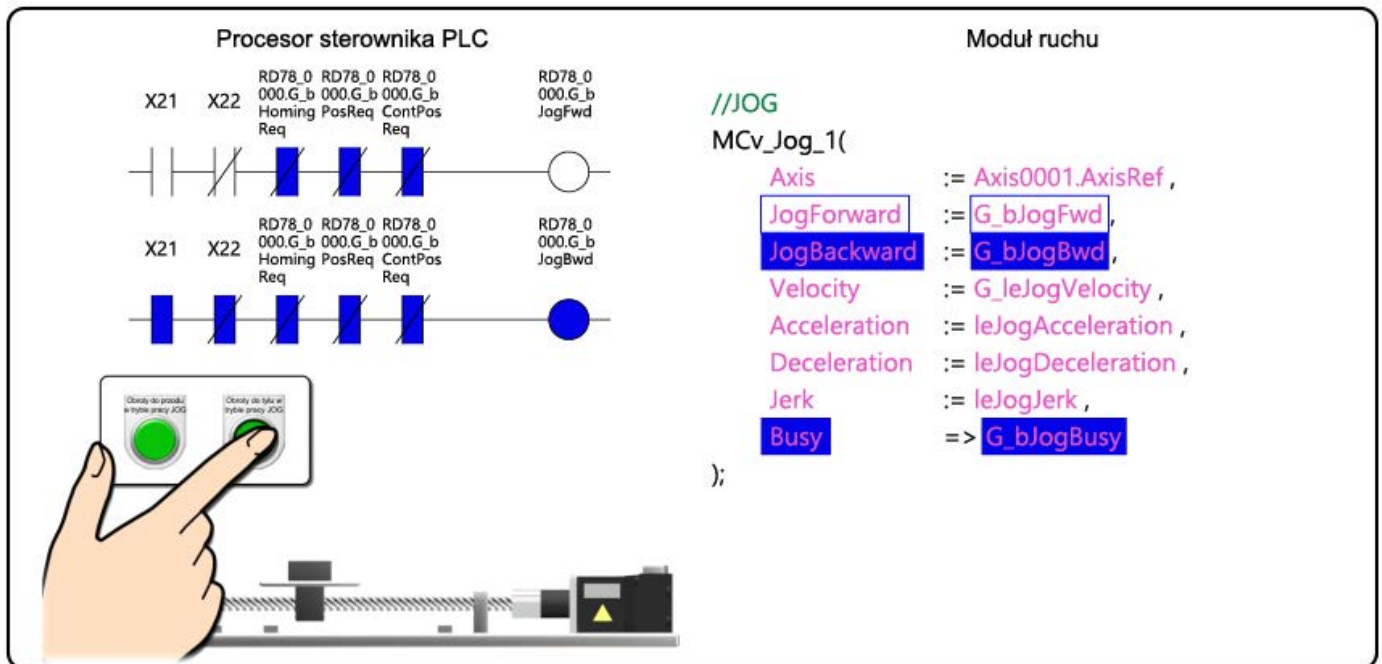
Włącz obroty do przodu w trybie pracy JOG (X21), aby przesunąć oś w kierunku zwiększania adresu, i wyłącz obroty, aby zatrzymać pracę.
Włącz obroty do tyłu w trybie pracy JOG (X22), aby przesunąć oś w kierunku zmniejszania adresu, i wyłącz obroty, aby zatrzymać pracę.



Sprawdź monitor programu.

Po włączeniu X21 następuje włączenie „RD78_0000.G_bJogFwd” i „G_bJogFwd” po stronie modułu ruchu.

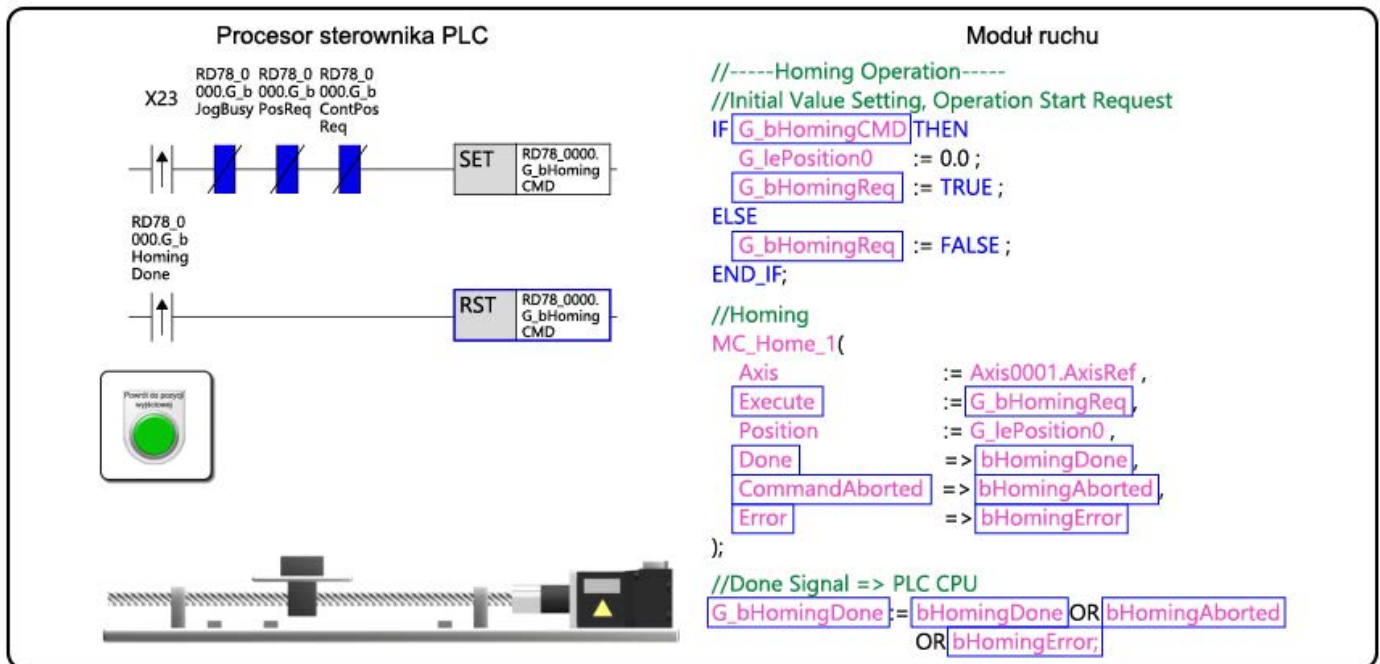
Po włączeniu wejścia JogForward w MCv_Jog_1 następuje uruchomienie obrotów do przodu w trybie pracy JOG.



Po włączeniu X22 następuje włączenie „RD78_0000.G_bJogBwd” i „G_bJogBwd” po stronie modułu ruchu.
Po włączeniu wejścia JogBackward w MCv_Jog_1 następuje uruchomienie obrotów do tyłu w trybie pracy JOG.



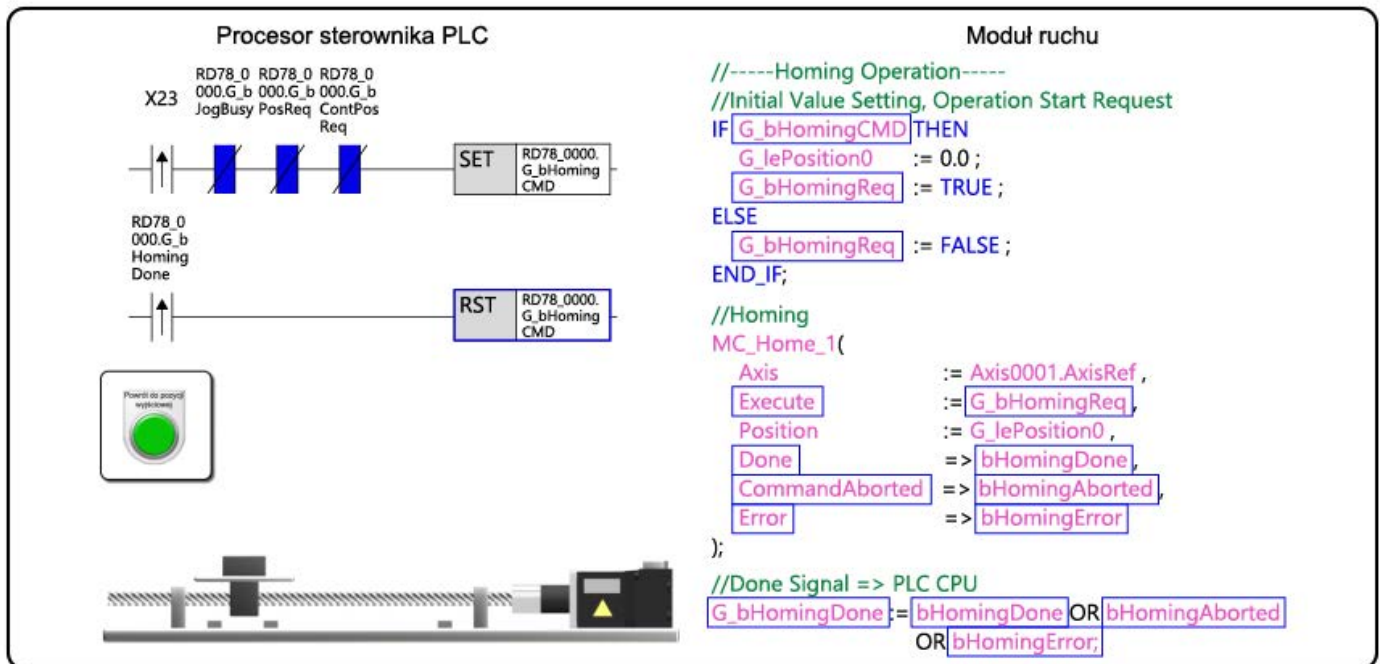
Włącz powrót do położenia wyjściowego (X23), aby rozpocząć powrót do położenia wyjściowego. Wykonaj powrót do położenia wyjściowego na czujnik zbliżeniowy DOG (33 jest odejmowane od Pr.PT45) Oś zatrzymuje się nieco dalej za czujnikiem DOG i ustawia ten punkt jako położenie wyjściowe.



Sprawdź monitor programu.

Po włączeniu X23 ustawiane jest polecenie
„RD78_0000.G_bHomingCMD”.

Następuje włączenie „G_bHomingCMD” po stronie modułu ruchu oraz włączenie „G_bHomingReq”, czyli polecenia wykonania MC_Home_1.

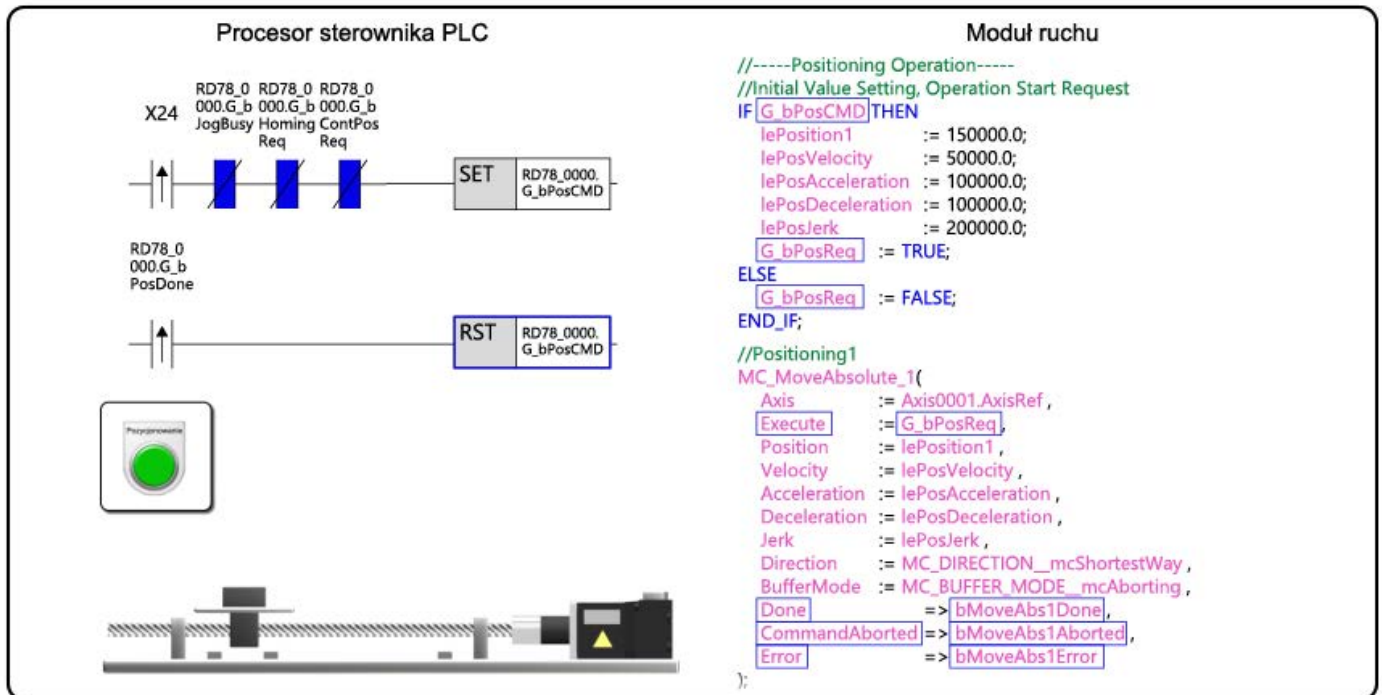


Po zakończeniu powrotu do położenia wyjściowego następuje włączenie wyjścia Done i „G_bHomingDone”. „G_bHomingCMD” po stronie procesora sterownika PLC jest resetowane do stanu początkowego.



Włączenie uruchomienia pozycjonowania (X24) powoduje uruchomienie ruchu posuwisto-zwrotnego.

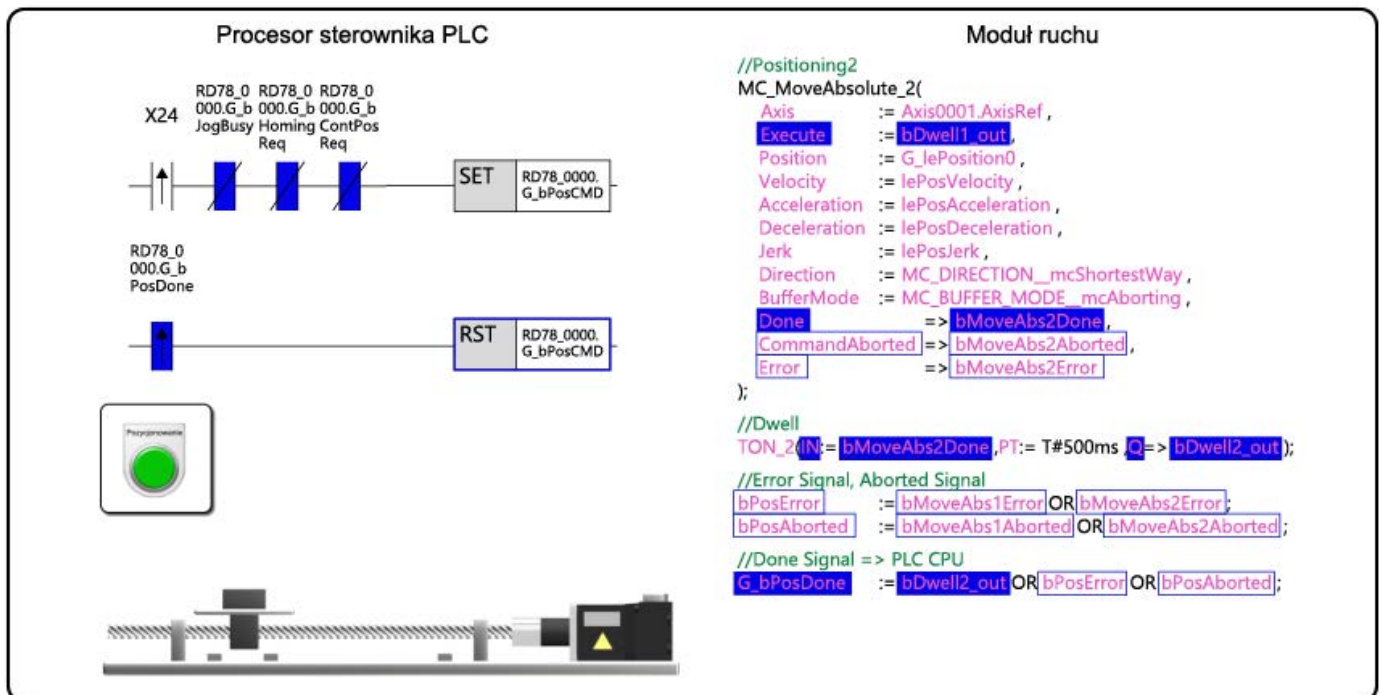
Oś przesuwana do przodu o 150 mm i zatrzymuje się na 0,5 sekundy, a następnie przesuwana do tyłu o 150 mm i zatrzymuje się na 0,5 sekundy.



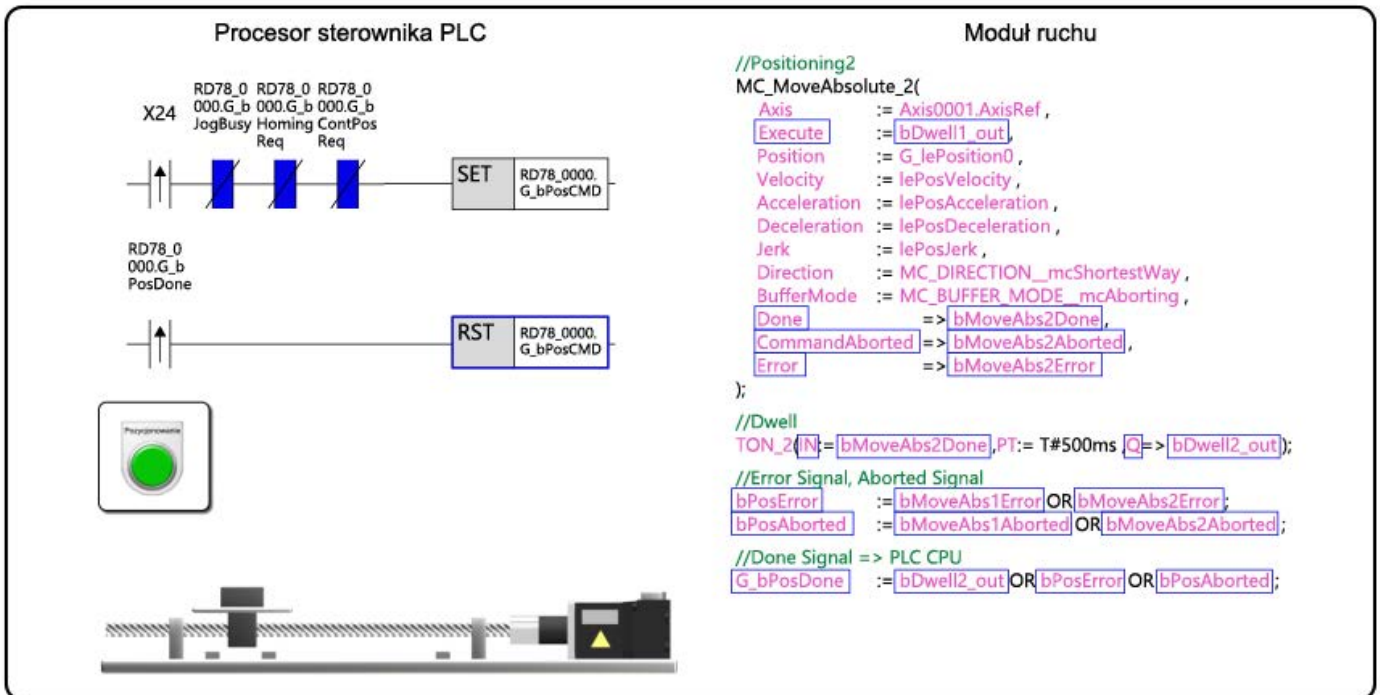
Sprawdź monitor programu.

Po włączeniu X24 ustawiane jest polecenie „RD78_0000.G_bPosCMD”.

Następuje włączenie „G_bPosCMD” po stronie modułu ruchu oraz włączenie „G_bPosReq”, czyli polecenia wykonania MC_MoveAbsolute_1.



Po zakończeniu ruchu posuwisto-zwrotnego i upłynięciu czasu oczekiwania następuje włączenie „G_bPosDone”. „G_bPosCMD” po stronie procesora sterownika PLC jest resetowane do stanu początkowego.



Na tym kończy się kontrola działania.
Przejdź do następnej strony.

Informacje zdobyte w tym rozdziale:

- Co to jest etykieta publiczna?
- Ustawienie etykiety publicznej
- Przykładowy program
- Zapisywanie programu
- Kontrola działania

Wskazówka

Co to jest etykieta publiczna?	<ul style="list-style-type: none"> • Etykieta publiczna to współdzielona etykieta, której można używać zarówno w module ruchu, jak i w procesorze sterownika PLC.
Ustawienie etykiety publicznej	<ul style="list-style-type: none"> • Zarejestruj etykiety publiczne z etykiet globalnych modułu ruchu. • Dla każdej etykiety wybierz, czy ma być odczytywana, czy zapisywana z/do procesora sterownika PLC. • Aby ustawić element struktury systemowej jako etykiety publiczną należy zarejestrować strukturalny typ danych jako etykiety publiczną • Po ustawieniu etykiet publicznych w module ruchu, przebuduj wszystkie programy i odzwierciedl etykiety publiczne. • Etykiety publiczne są rejestrowane w etykietach modułu po stronie procesora sterownika PLC.
Przykładowy program	<ul style="list-style-type: none"> • W tym rozdziale opisano następujący przykład programu: program drabinkowy procesora sterownika PLC, który wykorzystuje etykiety publiczne do wymiany sygnału uruchomienia pozycjonowania i sygnału ukończenia pozycjonowania.
Zapisywanie programu	<ul style="list-style-type: none"> • Najpierw zapisz dane do procesora sterownika PLC, a następnie do modułu ruchu.
Kontrola działania	<ul style="list-style-type: none"> • Działanie przykładowego programu zostało sprawdzone na filmie.

Tryb buforowania wykonuje operacje w sposób ciągły, uruchamiając wiele FB operacji Motion control FB. Można go ustawić za pomocą wejścia BufferMode dla Motion control FB. Dla każdej osi i grupy osi można uruchomić jednocześnie do dwóch FB.

(Przykład) MC_MoveAbsolute

```
MC_MoveAbsolute_1(
  Axis      := Axis0001.AxisRef ,
  Execute   := G_bPositioningReq ,
  ContinuousUpdate := FALSE ,
  Position  := lePosition1 ,
  Velocity  := lePosVelocity ,
  Acceleration := lePosAcceleration ,
  Deceleration := lePosDeceleration ,
  Jerk      := lePosJerk ,
  Direction := MC_DIRECTION_mcShortestWay ,
  BufferMode := MC_BUFFER_MODE__mcAborting ,
  Options   := 0 , //mcAccDec
  Done      => bMoveAbs1Done ,
  CommandAborted => bMoveAbs1Aborted ,
  Error     => bMoveAbs1Error
);
```

- | | |
|--|--|
| 0 lub MC_BUFFER_MODE__mcAborting | ... Wykonywany FB zostaje przerwany i natychmiast wykonywany jest następny FB. |
| 1 lub MC_BUFFER_MODE__mcBuffered | ... Gdy operacja wykonywanego FB zostanie zakończona, wykonywany jest następny FB. |
| 2 lub MC_BUFFER_MODE__mcBlendingLow | ... Jak prędkość przełączania ustawiana jest niższa z prędkości docelowych: wykonywanego FB i buforowanego FB. |
| 3 lub MC_BUFFER_MODE__mcBlendingPrevious | ... Prędkość docelowa wykonywanego FB jest ustawiana jako prędkość przełączania. |
| 4 lub MC_BUFFER_MODE__mcBlendingNext | ... Prędkość docelowa buforowanego FB, jest ustawiana jako prędkość przełączania. |
| 5 lub MC_BUFFER_MODE__mcBlendingHigh | ... Jako prędkość przełączania, ustawiana jest wyższa z docelowych prędkości: wykonywanego FB i buforowanego FB. |

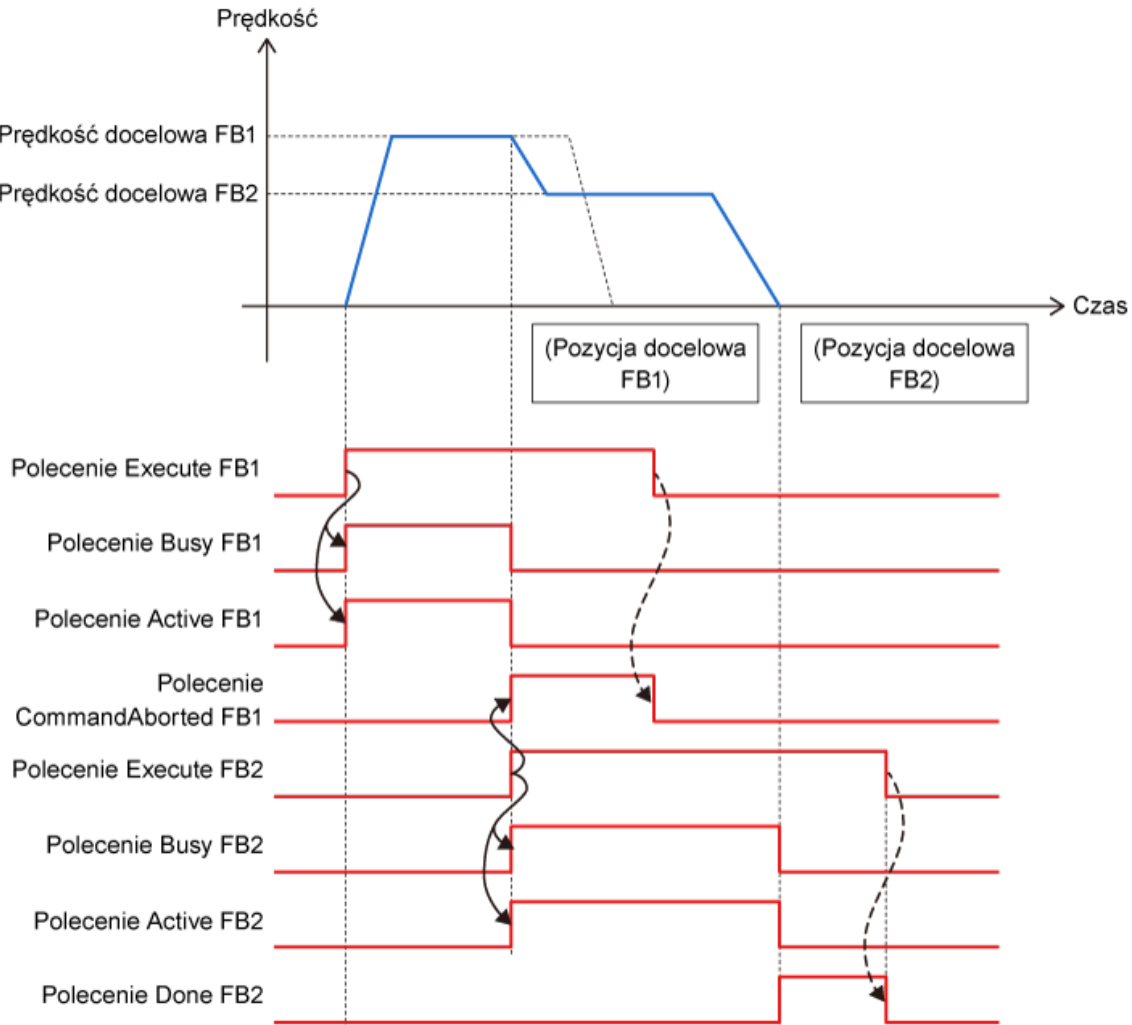
[Wskazówka]

W przypadku wejść Direction i BufferMode określ liczby lub enumeratory ENUM zaczynające się od MC_BUFFER_MODE i MC_DIRECTION.

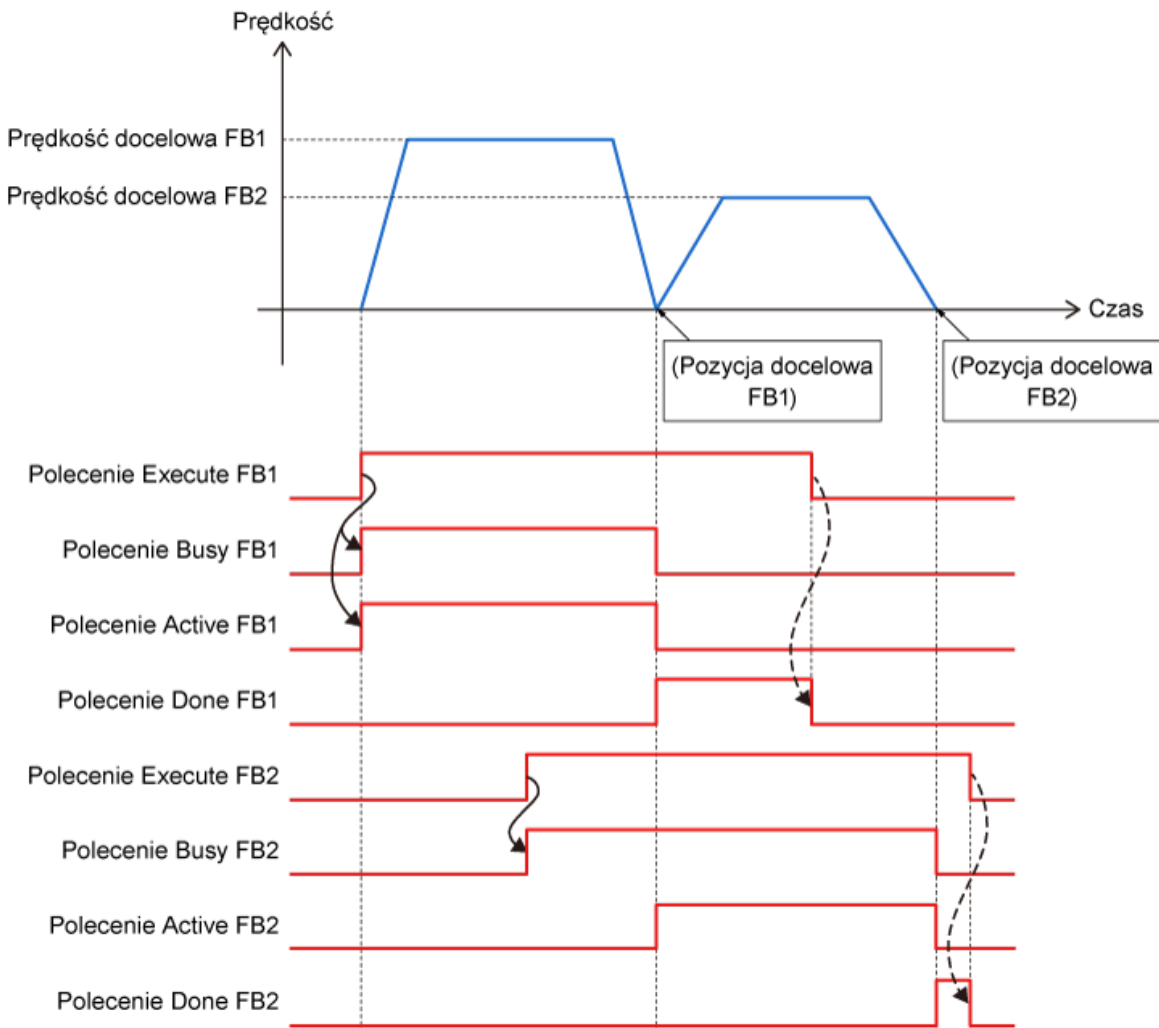
Szczegółowe informacje na temat enumeratorów ENUM można znaleźć w poniższej instrukcji obsługi.

- 📖 MELSEC iQ-R Programming Manual (Motion Control Function Blocks)
- 2 VARIABLES AND MOTION CONTROL FB
- 2.2 List of Enumerators

Poniższy schemat przedstawia operację, gdy atrybut BufferMode jest ustawiony na 0: mcAborting. Wykonywany FB zostaje przerwany i natychmiast wykonywany jest następny FB.



Poniższy schemat przedstawia operację, gdy atrybut BufferMode jest ustawiony na 1: mcBuffered. Gdy operacja wykonywanego FB zostanie zakończona, wykonywany jest następny FB.



Gdy atrybut BufferMode jest ustawiony na mcBlending***, następny FB jest wykonywany w sposób ciągły po osiągnięciu pozycji docelowej wykonywanego FB.

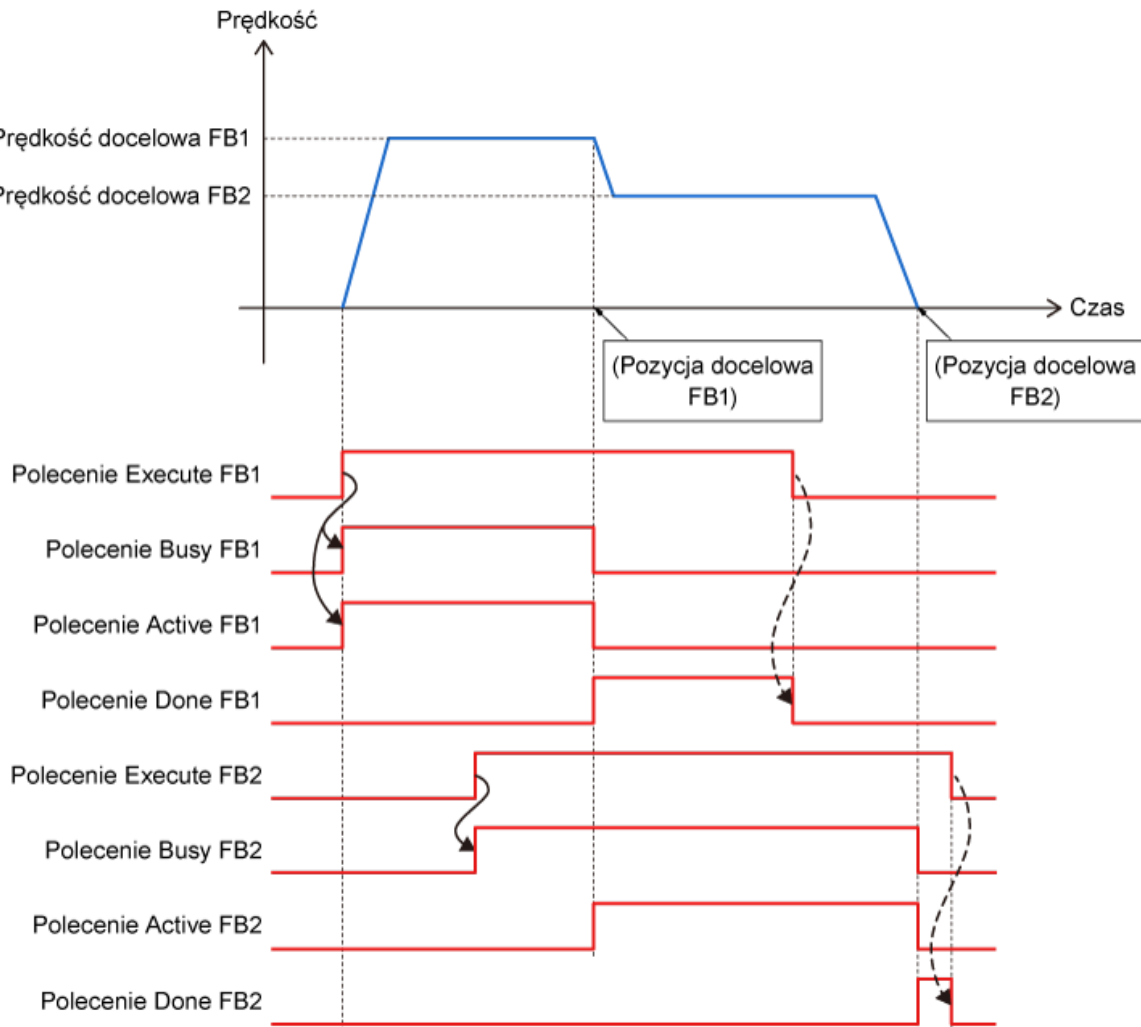
W poniższym opisie FB, który ma zostać wykonany jako pierwszy, to FB1, a FB, który ma być buforowany, to FB2.

(1) BlendingPrevious

Poniższy schemat przedstawia operację, gdy atrybut BufferMode jest ustawiony na 3: mcBlendingPrevious.

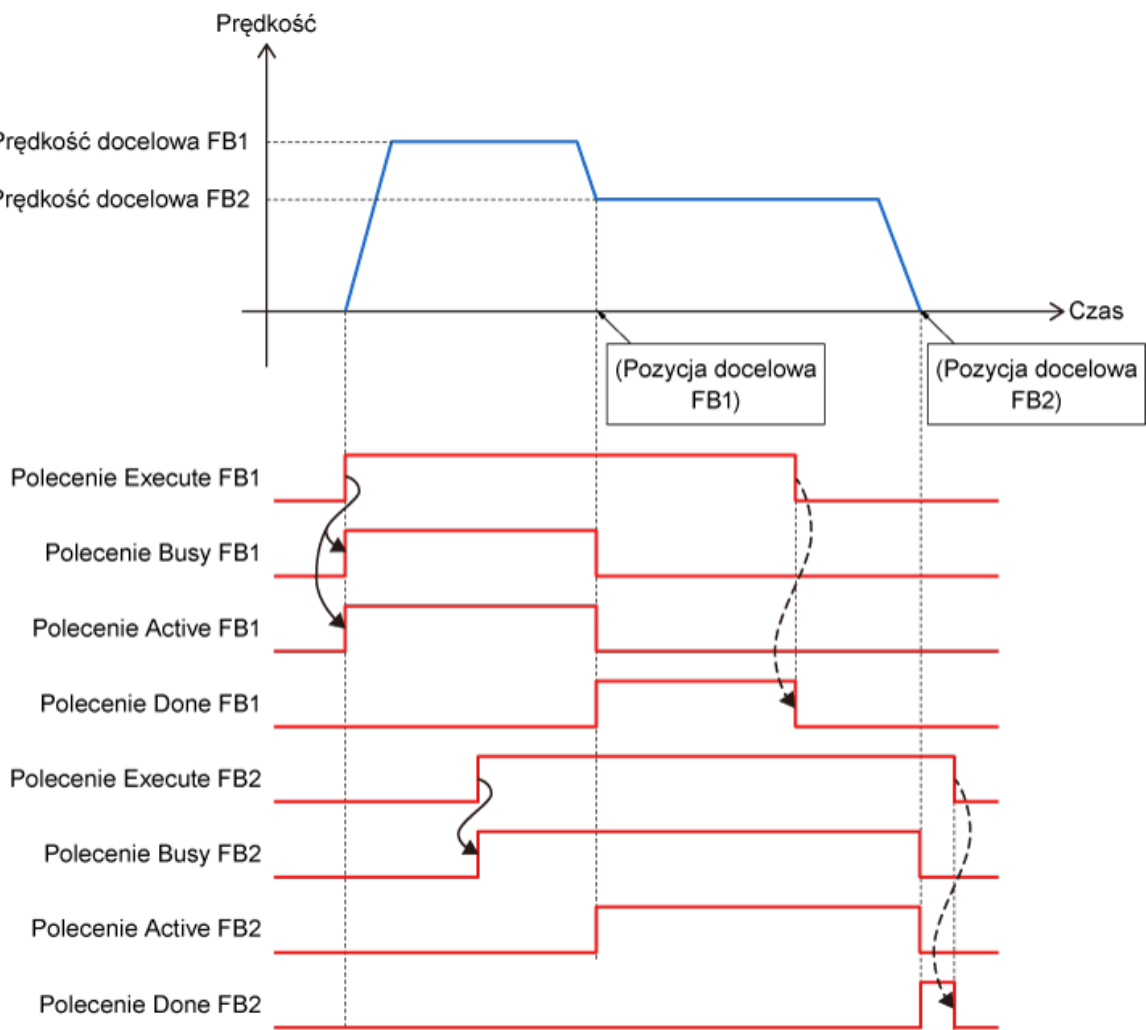
Operacja jest wykonywana z prędkością docelową FB1 do pozycji docelowej FB1.

Gdy operacja zostanie przełączona na FB2, prędkość zostanie zmieniona na prędkość docelową FB2 i nastąpi przejście do pozycji docelowej FB2.



(2) BlendingNext

Poniższy schemat przedstawia operację, gdy atrybut BufferMode jest ustawiony na 4: mcBlendingNext. Prędkość ulega zmianie na prędkość docelową FB2, gdy operacja osiągnie pozycję docelową FB1.



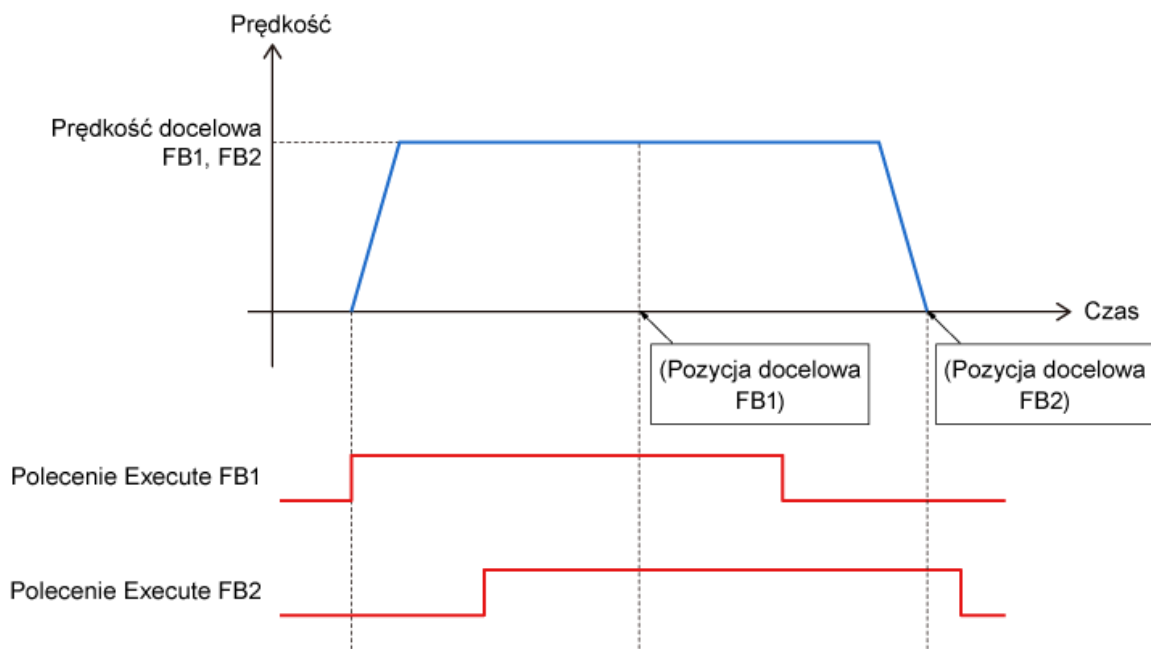
(3) BlendingLow, BlendingHigh

Operacja, gdy atrybut BufferMode jest ustawiony na 2: mcBlendingLow lub 5: mcBlendingHigh, różni się w zależności od tego, która prędkość docelowa – FB1/FB2 – jest wyższa.

Wartość ustawienia	Prędkość docelowa FB1 > Prędkość docelowa FB2	Prędkość docelowa FB1 < Prędkość docelowa MFB2
2: mcBlendingLow	Taka sama operacja jak w przypadku BlendingPrevious	Taka sama operacja jak w przypadku BlendingNext
5: mcBlendingHigh	Taka sama operacja jak w przypadku BlendingNext	Taka sama operacja jak w przypadku BlendingPrevious

[Wskazówka]

Poniższy schemat przedstawia przebieg prędkości dla BlendingPrevious, BlendingNext, BlendingHigh i BlendingLow, gdy prędkości docelowe FB1 i FB2 są takie same.



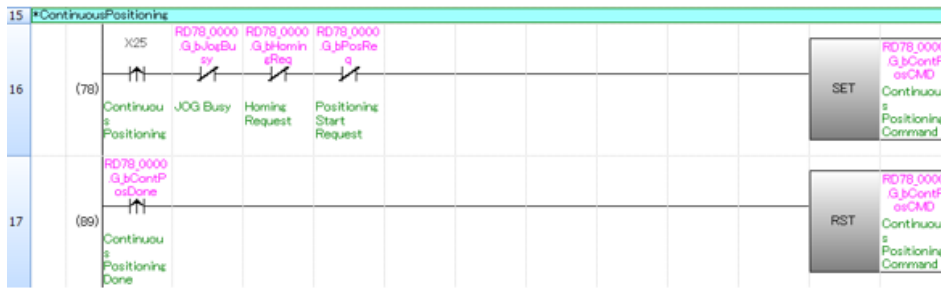
(1) Operacje przykładowego programu

W tym rozdziale wykorzystano przykładowy program z rozdziału 2.

Sprawdź różnicę operacji trybu buforowania w programie rozpoczynającym się od X25.

Element	FB1 (MC_MoveAbsolute)	FB2 (MC_MoveAbsolute)
Adres pozycjonowania	75 000,0 [μm]	150 000,0 [μm]
Prędkość	50 000,0 [$\mu\text{m/s}$]	25 000,0 [$\mu\text{m/s}$]
Przyspieszenie, spowolnienie	100 000,0 [$\mu\text{m/s}^2$]	50 000,0 [$\mu\text{m/s}^2$]
Ruch z szarpnięciem	200 000,0 [$\mu\text{m/s}^3$]	100 000,0 [$\mu\text{m/s}^3$]

- (2) Program procesora sterownika PLC
MAIN (drabinka, program skanujący)



Narastające zbocze sygnału uruchomienia sterowania ciągłym pozycjonowaniem (X25) jest podtrzymywane przez G_bContPosCMD i wysyłane do modułu ruchu jako warunek uruchomienia sterowania ciągłym pozycjonowaniem.

Ustawiana jest blokada uniemożliwiająca uruchomienie sterowania pozycjonowaniem podczas wykonywania innego programu. Etykieta G_bContPosCMD jest resetowana na narastające zbocze sygnału potwierdzającego ukończenie pozycjonowania ciągłego.

- (3) Program modułu ruchu
ContinuousPositioning (normalny typ wykonania)

```

1  //-----Continuous Positioning Operation-----
2  //Initial Value Setting, Operation Start Request
3  IF G_bContPosCMD THEN
4      lePosition1 := 75000.0;
5      lePosVelocity1 := 50000.0;
6      lePosAcceleration1 := 100000.0;
7      lePosDeceleration1 := 100000.0;
8      lePosJerk1 := 200000.0;
9      lePosition2 := 150000.0;
10     lePosVelocity2 := 25000.0;
11     lePosAcceleration2 := 50000.0;
12     lePosDeceleration2 := 50000.0;
13     lePosJerk2 := 100000.0;
14     G_bContPosReq := TRUE;
15 ELSE
16     G_bContPosReq := FALSE;
17 END_IF;
18
19 //Positioning1
20 MC_MoveAbsolute_1(
21     Axis := Axis0001.AxisRef,
22     Execute := G_bContPosReq,
23     Position := lePosition1,
24     Velocity := lePosVelocity1,
25     Acceleration := lePosAcceleration1,
26     Deceleration := lePosDeceleration1,
27     Jerk := lePosJerk1,
28     Direction := MC_DIRECTION_mcShortestWay,
29     Done => bMoveAbs1Done,
30     Active => bMoveAbs1Active,
31     CommandAborted => bMoveAbs1Aborted,
32     Error => bMoveAbs1Error
33 );
34 //Positioning2
35 MC_MoveAbsolute_2(
36     Axis := Axis0001.AxisRef,
37     Execute := bMoveAbs1Active,
38     Position := lePosition2,
39     Velocity := lePosVelocity2,
40     Acceleration := lePosAcceleration2,
41     Deceleration := lePosDeceleration2,
42     Jerk := lePosJerk2,
43     Direction := MC_DIRECTION_mcShortestWay,
44     BufferMode := MC_BUFFER_MODE_mcBuffered,
45     Done => bMoveAbs2Done,
46     CommandAborted => bMoveAbs2Aborted,
47     Error => bMoveAbs2Error
48 );
49 //Dwell
50 SET(bMoveAbs2Done,bDwell_In);
51 TON_1(IN:= bDwell_In ,PT:= T#100ms ,Q=> bDwell_out);
52
53 //Error Signal, Aborted Signal
54 bError := bMoveAbs1Error OR bMoveAbs2Error;
55 bAborted := bMoveAbs1Aborted OR bMoveAbs2Aborted;
56
57 //Done Signal => PLC CPU
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
59 //Reset Dwell_In
60 RST(G_bContPosDone,bDwell_In);
61

```

Odbiera sygnał uruchomienia sterowania ciągłym pozycjonowaniem (G_bContPosCMD) z procesora sterownika PLC.
Dane wymagane do pozycjonowania są wpisywane do etykiety i włączane jest żądanie uruchomienia sterowania ciągłym pozycjonowaniem (G_bContPositioningReq).

Wyłącza G_bContPositioningReq po wyłączeniu G_bContPosCMD.

FB1 (MC_MoveAbsolute_1) wykonuje pozycjonowanie.

FB2 (MC_MoveAbsolute_2) jest uruchamiany przez wyjście Active FB1 (MC_MoveAbsolute_1). Następnie FB2 jest buforowany podczas wykonywania FB1.

Zmień wejście BufferMode FB2 (MC_MoveAbsolute_2) i sprawdź działanie trybu buforowania.

Wyjście Done FB2 (MC_MoveAbsolute_2) uruchamia timer wprowadzający opóźnienie.

Zwraca sygnał zakończenia wykonywania do procesora sterownika PLC po upływie czasu oczekiwania lub po włączeniu wyjścia Error lub CommandAborted w MC_MoveAbsolute.
Jednocześnie resetowane jest wejście timera wprowadzającego czas opóźnienia włączenia.

Kliknij przycisk odtwarzania w lewym dolnym rogu okna.

```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction  := MC_DIRECTION__mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBuffered ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Sprawdź działanie trybu buforowania.

Otwórz „LinearInterpolation” w przykładowym programie i zmień wejście BufferMode w MC_MoveAbsolute_2, aby sprawdzić działanie trybu buforowania.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```



```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction  := MC_DIRECTION_mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE_mcBuffered ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

mcBuffered jest ustawiany podczas pobierania programu.

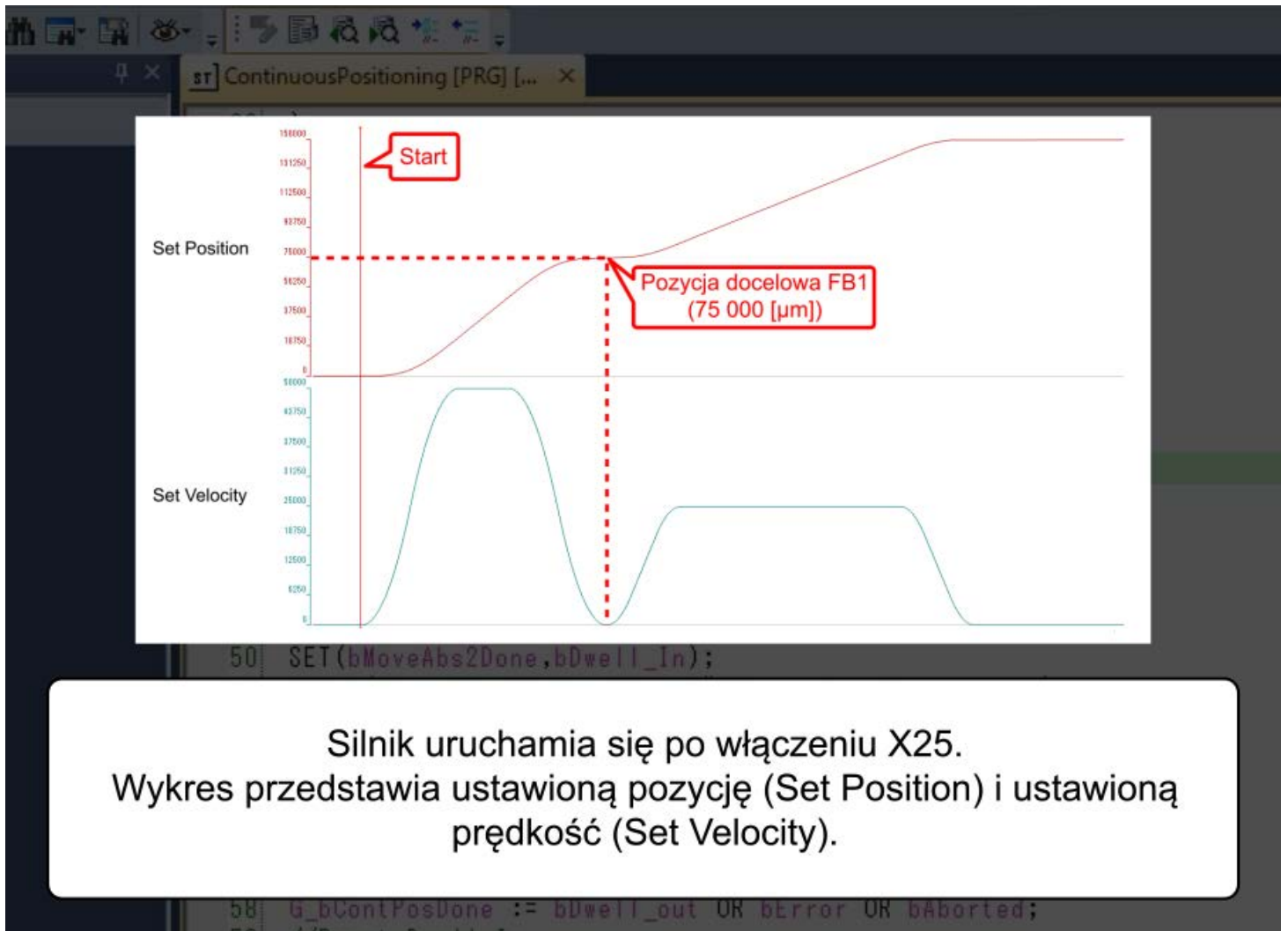
```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```

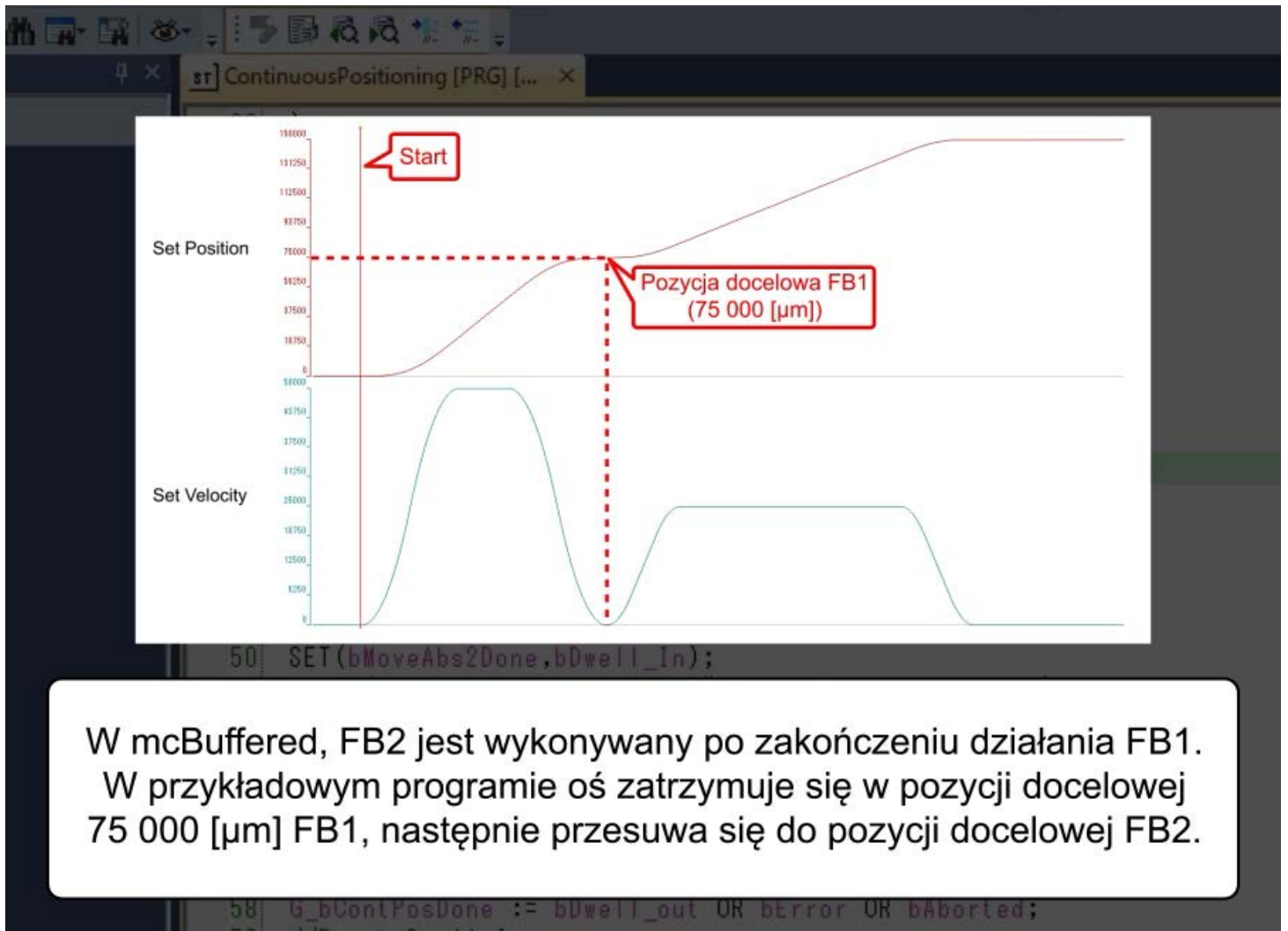
```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction  := MC_DIRECTION_mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBuffered ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Najpierw sprawdź działanie mcBuffered.

Sprawdź, czy wejście BufferMode w MC_MoveAbsolute_2 jest ustawione na „MC_BUFFER_MODE__mcBuffered”, i wgraj program do modułu ruchu.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```





```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction := MC_DIRECTION_mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcAborting ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Następnie sprawdź działanie mcAborting.
Zmień wejście BufferMode w MC_MoveAbsolute_2 na „MC_BUFFER_MODE__mcAborting”, przebuduj wszystkie programy i wgraj do modułu ruchu.

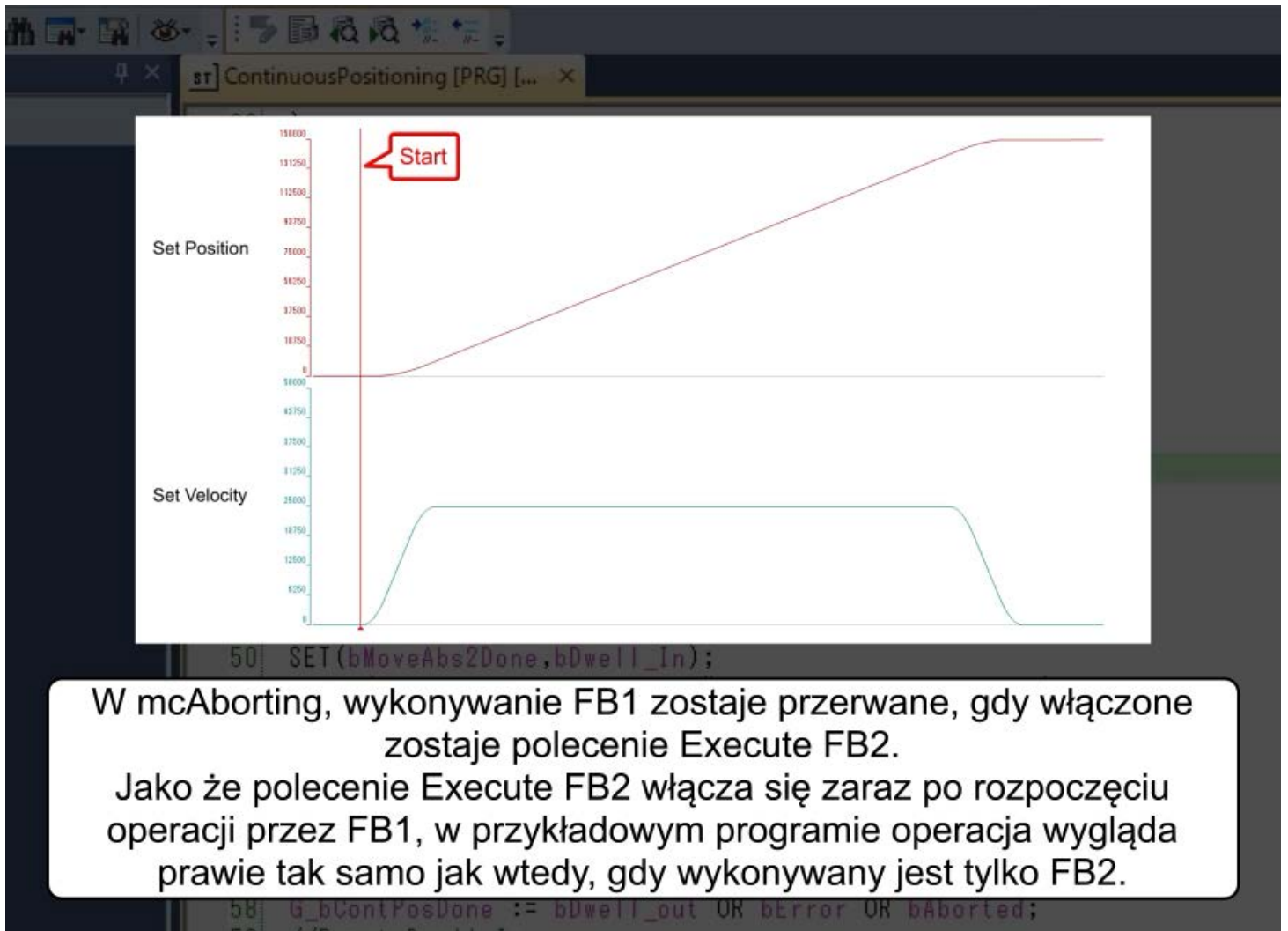
```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```



```
50 SET(bMoveAbs2Done,bDwell_In);
```

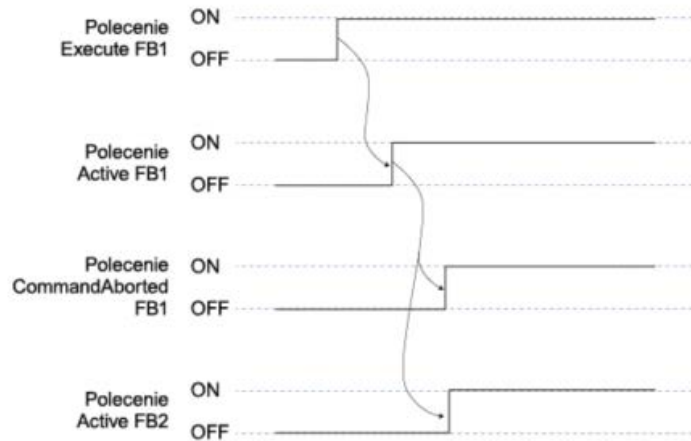
Silnik uruchamia się po włączeniu X25.
Wykres przedstawia ustawioną pozycję (Set Position) i ustawioną prędkość (Set Velocity).

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```



W mcAborting, wykonywanie FB1 zostaje przerwane, gdy włączone zostaje polecenie Execute FB2.

Jako że polecenie Execute FB2 włącza się zaraz po rozpoczęciu operacji przez FB1, w przykładowym programie operacja wygląda prawie tak samo jak wtedy, gdy wykonywany jest tylko FB2.



Sprawdź sygnały WE/WY FB1 i FB2 w momencie uruchomienia.
Wyjście CommandAborted FB1 jest włączone, co wskazuje, że wykonywanie FB1 jest przerwane.

```
50 SET(bMoveAbs2Done,bDwell_In);
```

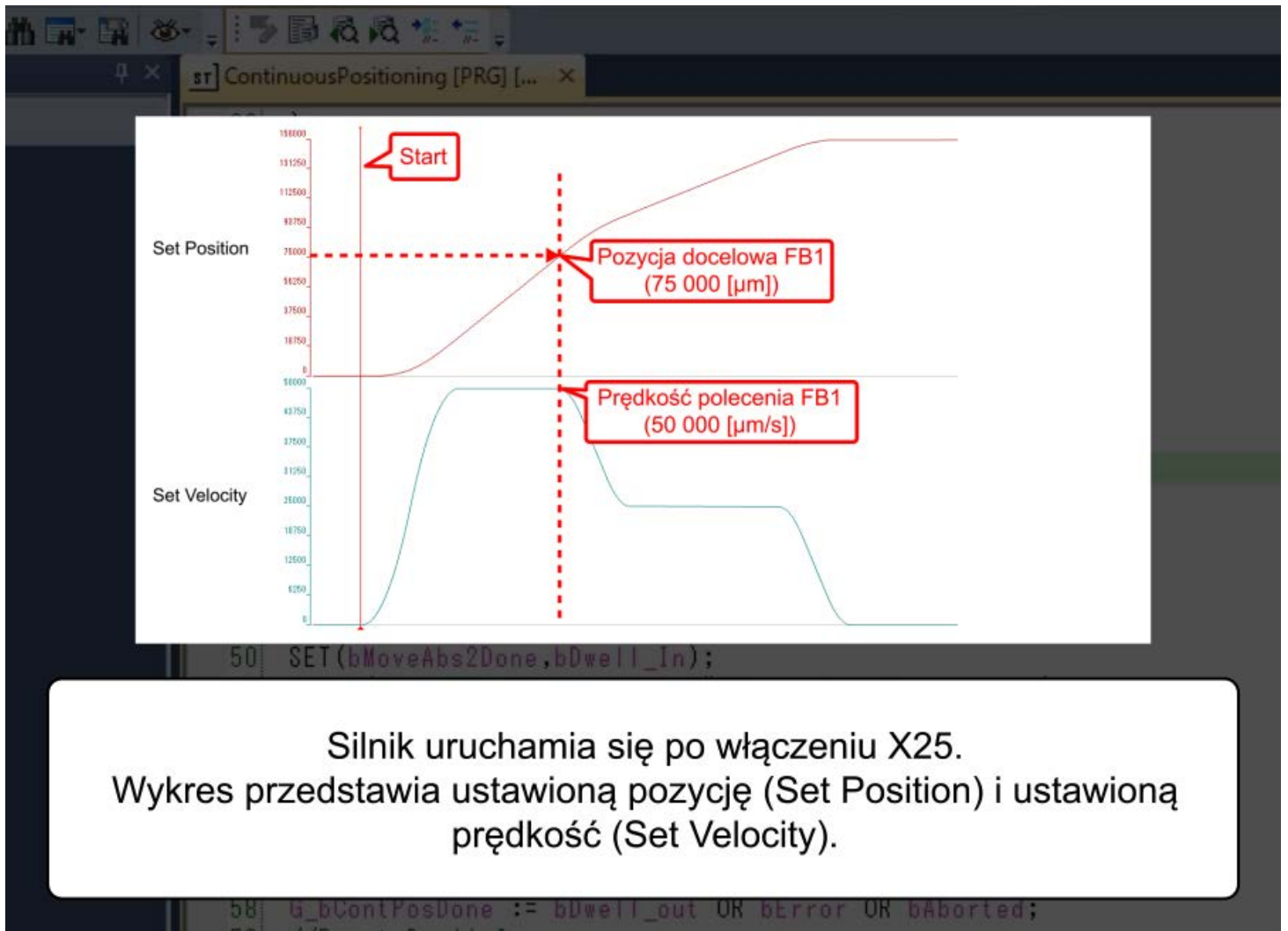
```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```



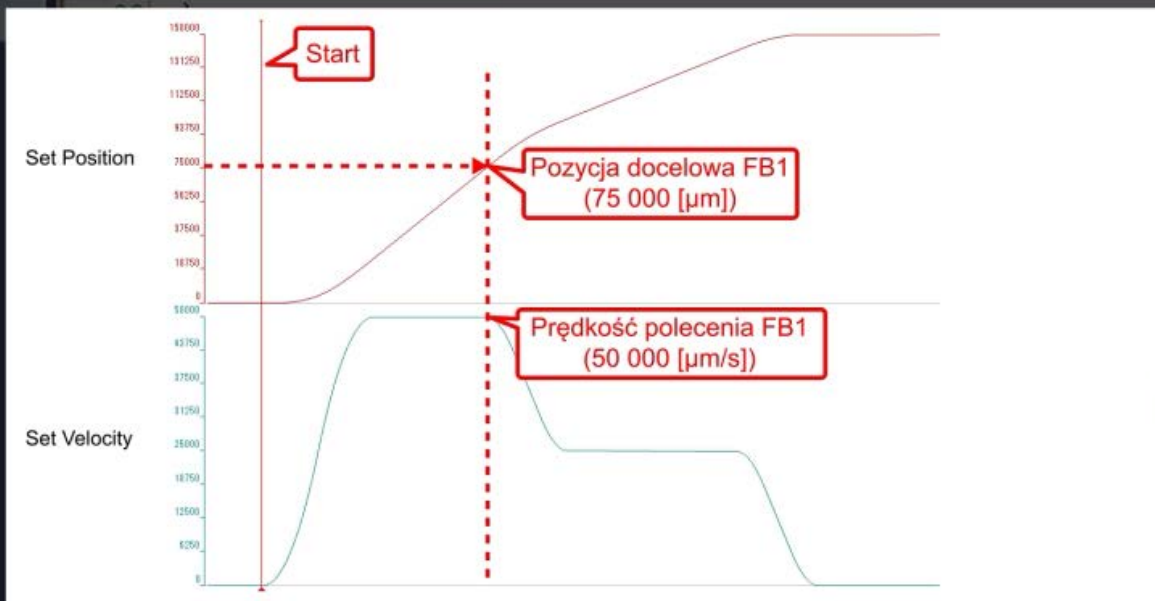
```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction := MC_DIRECTION__mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBlendingPrevious ,  
45     Done      => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error     => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Następnie sprawdź działanie mcBlendingPrevious.
Zmień wejście BufferMode w MC_MoveAbsolute_2 na „MC_BUFFER_MODE__mcBlendingPrevious”, przebuduj wszystkie programy i wgraj do modułu ruchu.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```



Silnik uruchamia się po włączeniu X25.
Wykres przedstawia ustawioną pozycję (Set Position) i ustawioną prędkość (Set Velocity).



```
50 SET(bMoveAbs2Done,bDwell_In);
```

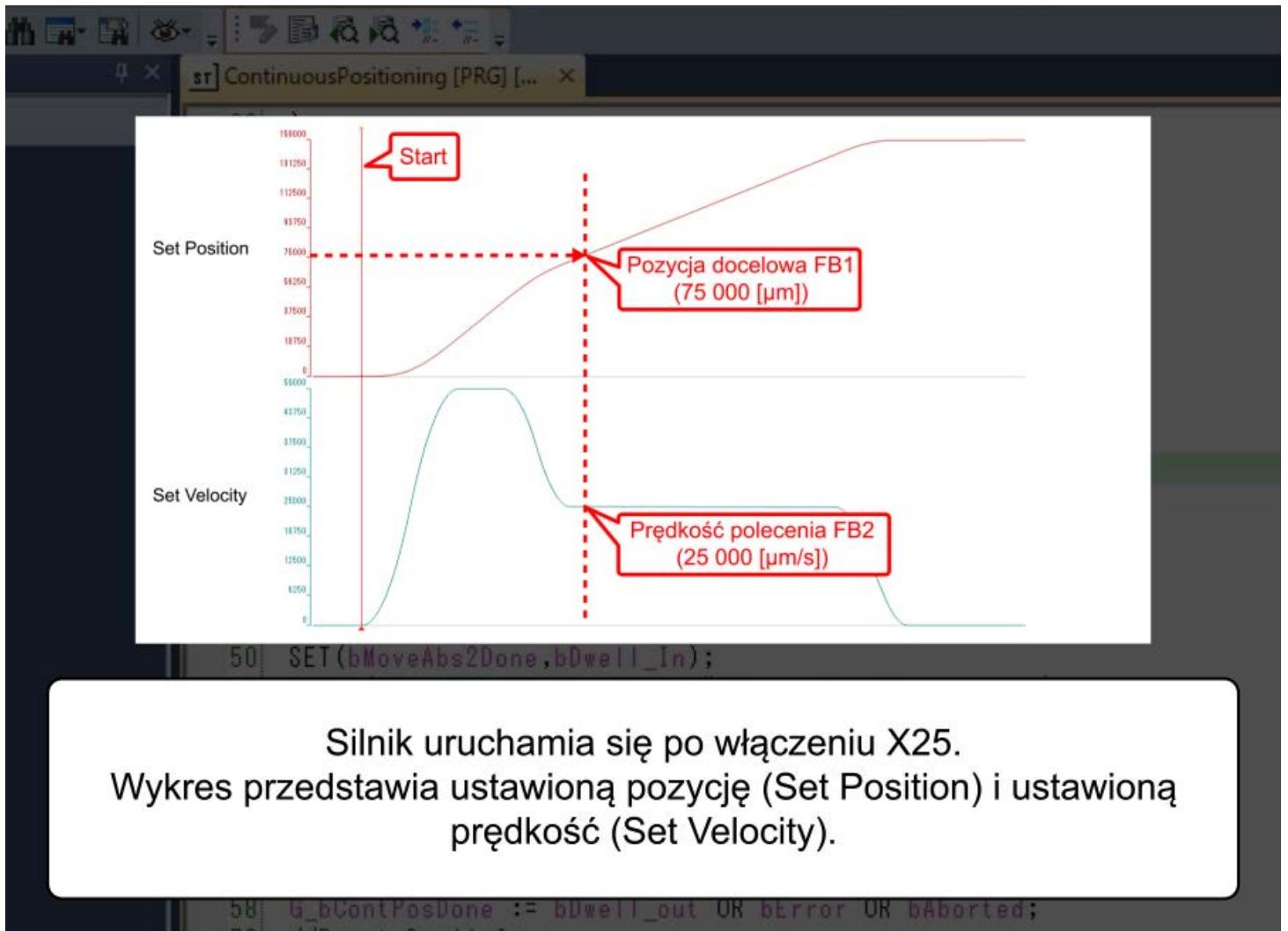
W `mcBlendingPrevious` zadana prędkość FB1 jest stosowana po osiągnięciu pozycji docelowej FB1.
Następnie stosowana jest prędkość polecenia FB2,
a oś przesuwa się do pozycji docelowej FB2.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```

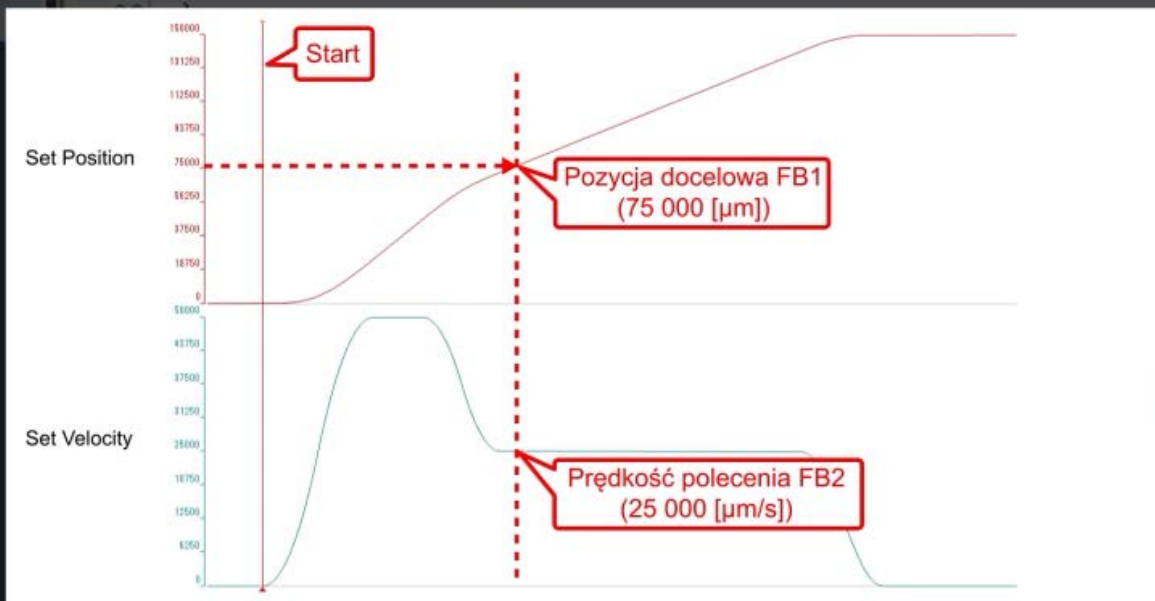
```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction := MC_DIRECTION__mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBlendingNext ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Następnie sprawdź działanie mcBlendingNext.
Zmień wejście BufferMode w MC_MoveAbsolute_2 na „MC_BUFFER_MODE__mcBlendingNext”, przebuduj wszystkie programy i wgraj do modułu ruchu.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```



Silnik uruchamia się po włączeniu X25.
Wykres przedstawia ustawioną pozycję (Set Position) i ustawioną prędkość (Set Velocity).



```
50 SET(bMoveAbs2Done,bDwell_In);
```

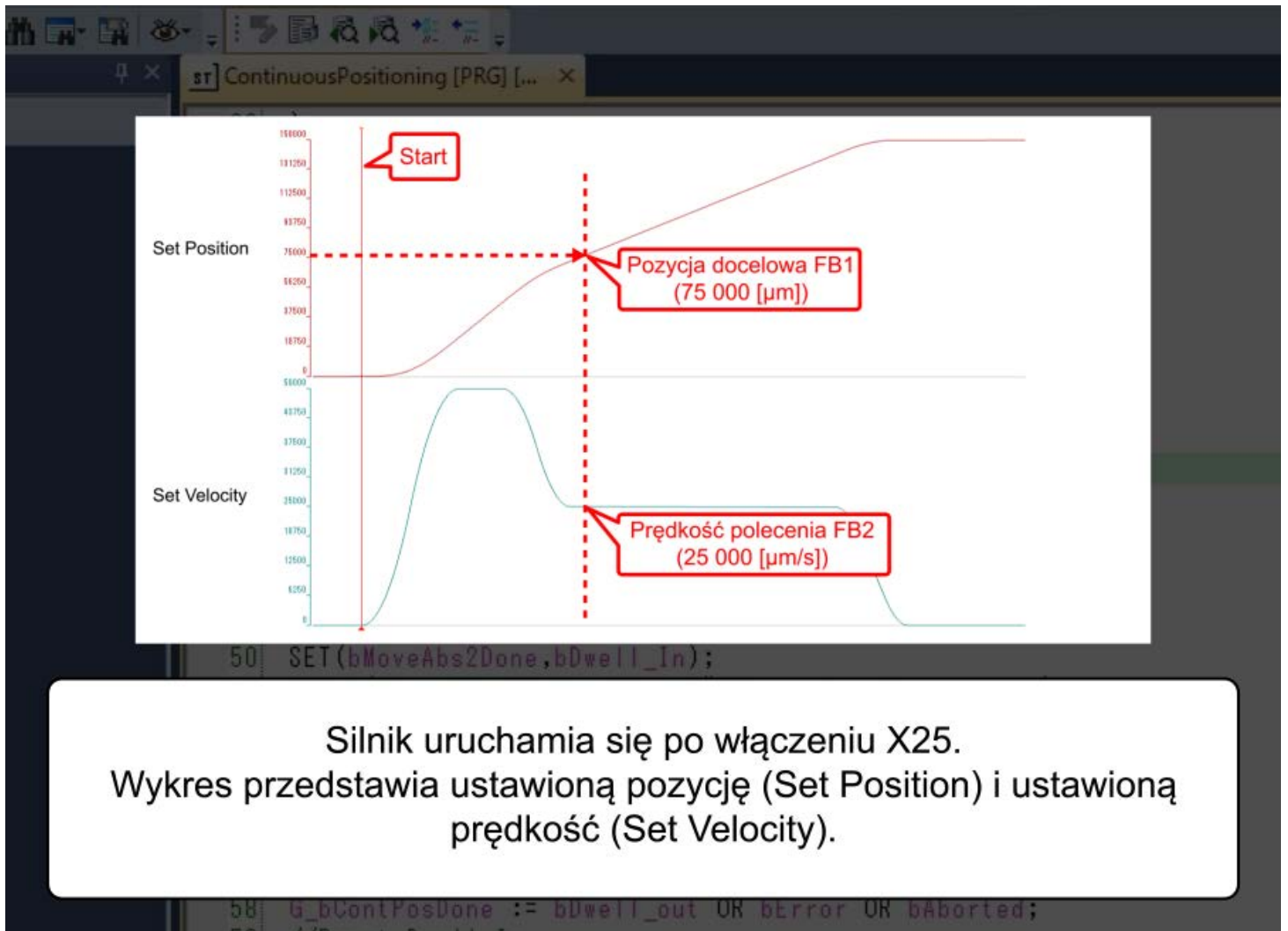
W mcBlendingNext zadana prędkość FB2 jest stosowana po osiągnięciu pozycji docelowej FB1.
Następnie oś przesuwa się do pozycji docelowej FB2.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```

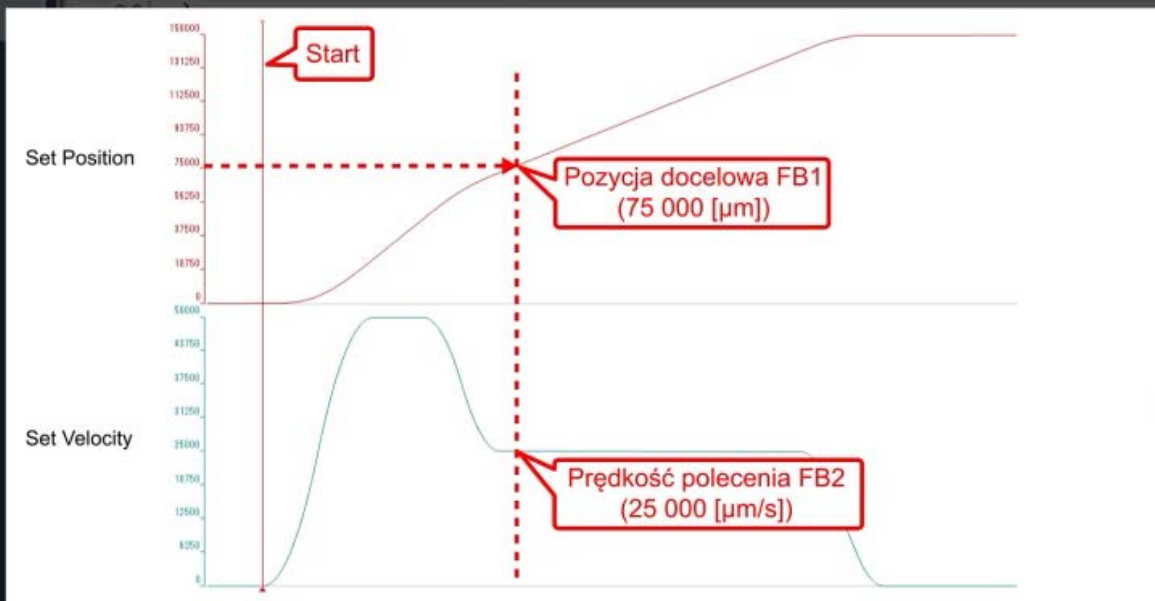
```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction := MC_DIRECTION__mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBlendingLow ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Następnie sprawdź działanie mcBlendingLow.
Zmień wejście BufferMode w MC_MoveAbsolute_2 na „MC_BUFFER_MODE__mcBlendingLow”, przebuduj wszystkie programy i wgraj do modułu ruchu.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```



Silnik uruchamia się po włączeniu X25.
Wykres przedstawia ustawioną pozycję (Set Position) i ustawioną prędkość (Set Velocity).



```
50 SET(bMoveAbs2Done,bDwell_In);
```

W mcBlendingLow stosowana jest mniejsza zadana prędkość po osiągnięciu pozycji docelowej FB1.

Jako że prędkość polecenia FB2 jest mniejsza, w przykładzie z tego szkolenia przebieg prędkości jest taki sam jak w przypadku BlendingNext.

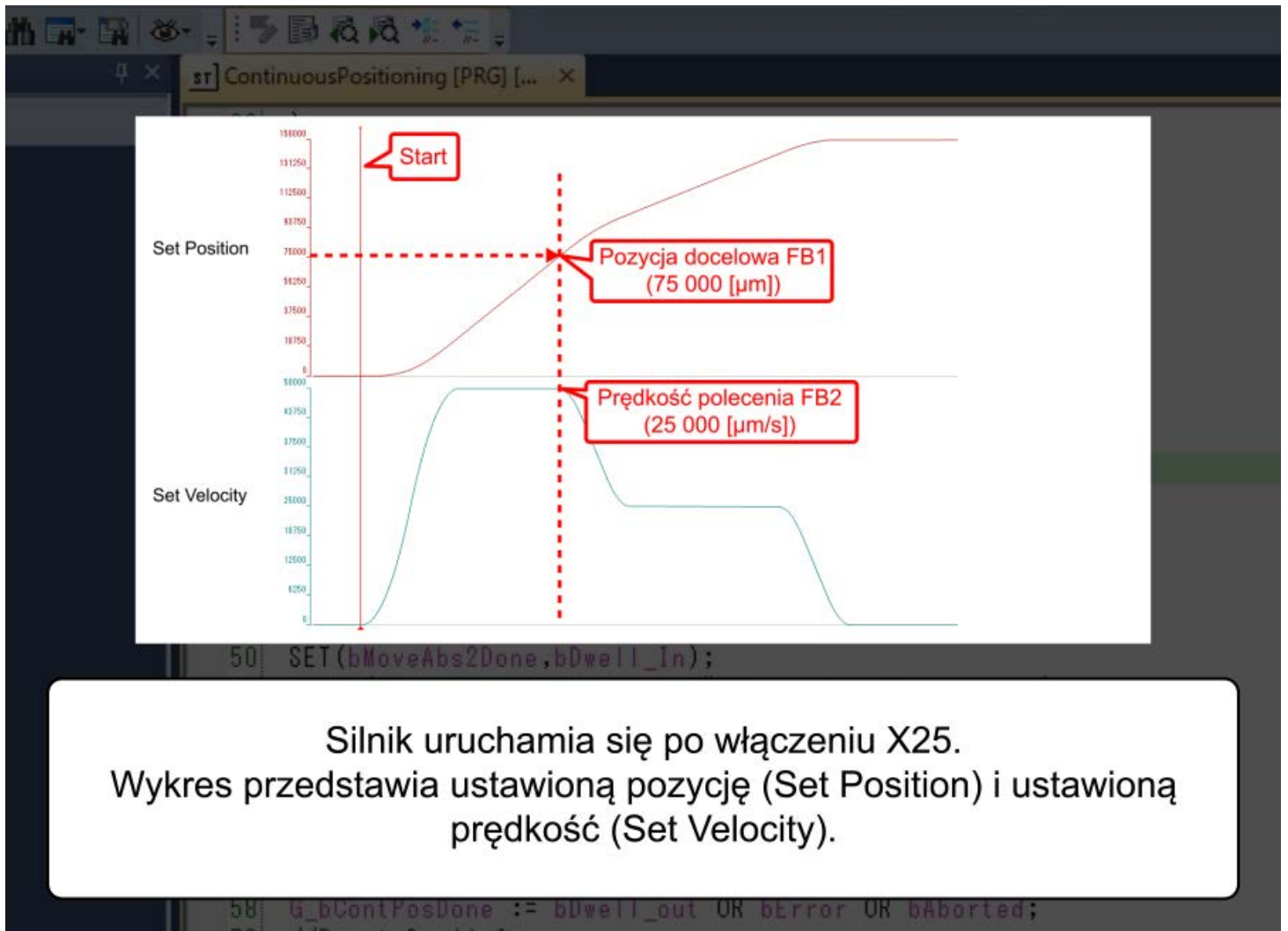
```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```

```
59 /
```

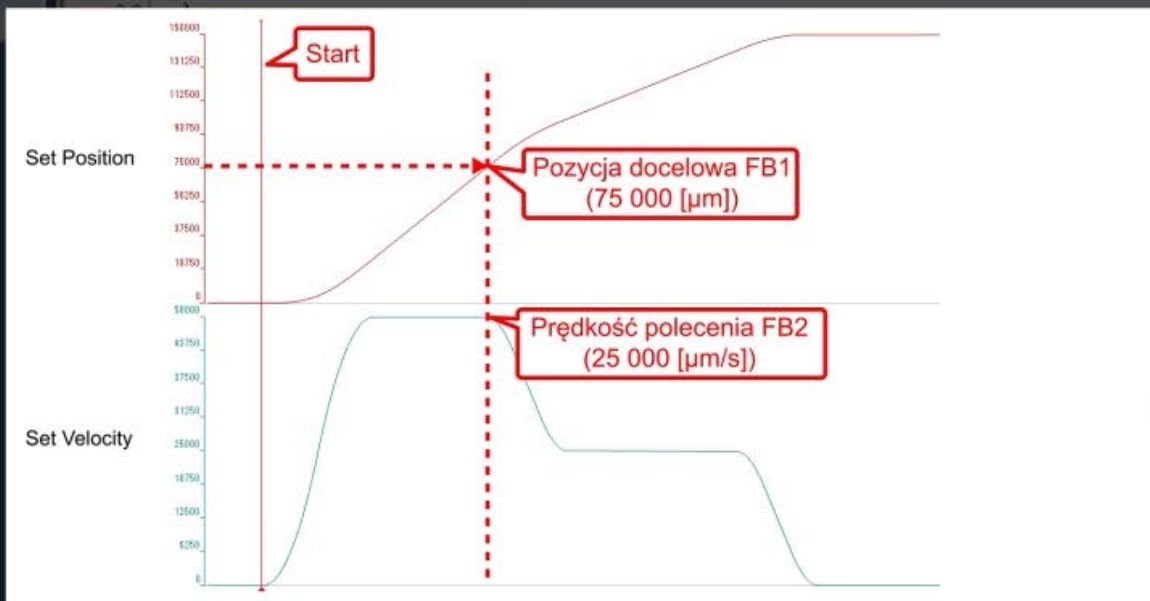
```
33 );  
34 //Positioning2  
35 MC_MoveAbsolute_2(  
36     Axis      := Axis0001.AxisRef ,  
37     Execute   := bMoveAbs1Active ,  
38     Position  := lePosition2 ,  
39     Velocity  := lePosVelocity2 ,  
40     Acceleration:= lePosAcceleration2 ,  
41     Deceleration:= lePosDeceleration2 ,  
42     Jerk      := lePosJerk2 ,  
43     Direction  := MC_DIRECTION__mcShortestWay ,  
44     BufferMode := MC_BUFFER_MODE__mcBlendingHigh ,  
45     Done       => bMoveAbs2Done ,  
46     CommandAborted => bMoveAbs2Aborted ,  
47     Error      => bMoveAbs2Error  
48 );  
49 //Dwell  
50 SET(bMoveAbs2Done,bDwell_In);
```

Na koniec sprawdź działanie mcBlendingHigh.
Zmień wejście BufferMode w MC_MoveAbsolute_2 na
„MC_BUFFER_MODE__mcBlendingHigh”, przebuduj wszystkie
programy i wgraj do modułu ruchu.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;  
59 //Dwell
```



Silnik uruchamia się po włączeniu X25.
Wykres przedstawia ustawioną pozycję (Set Position) i ustawioną prędkość (Set Velocity).

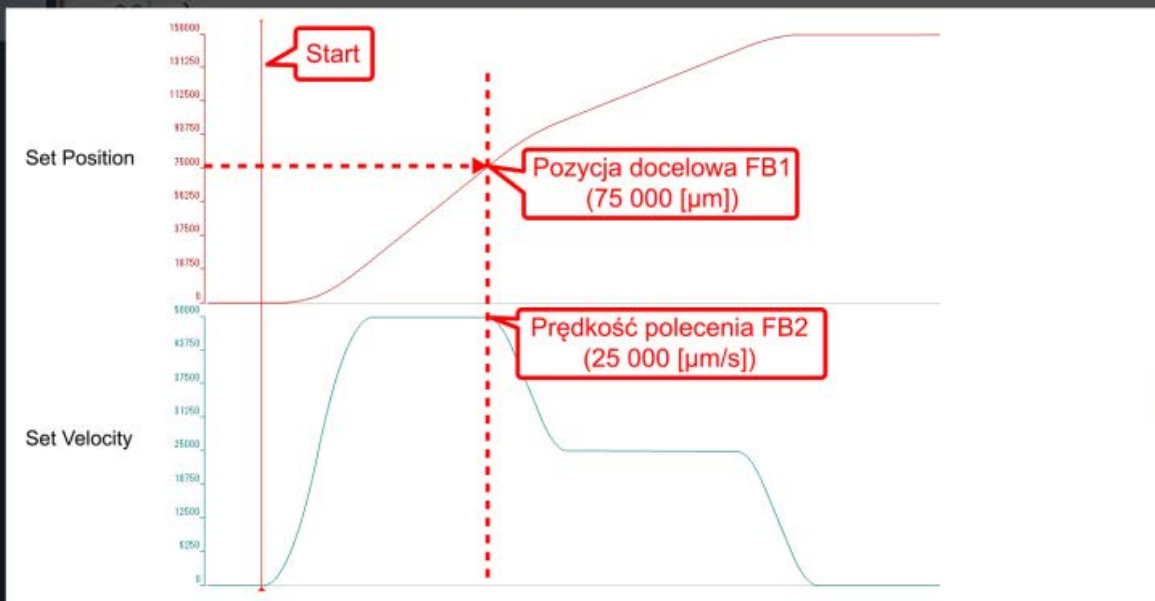


```
50 SET(bMoveAbs2Done,bDwell_In);
```

W mcBlendingHigh stosowana jest wyższa prędkość zadana po osiągnięciu pozycji docelowej FB1.

Jako że prędkość polecenia FB1 jest wyższa, w przykładzie z tego szkolenia przebieg prędkości jest taki sam jak w przypadku BlendingPrevious.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```



```
50 SET(bMoveAbs2Done,bDwell_In);
```

Na tym kończy się kontrola BufferMode.
Przejdź do następnej strony.

```
58 G_bContPosDone := bDwell_out OR bError OR bAborted;
```

Informacje zdobyte w tym rozdziale:

- Przerwanie wykonywania
- Buforowanie
- Łączenie
- Przykładowy program
- Kontrola działania

Wskazówka

Przerwanie wykonywania	<ul style="list-style-type: none"> • Gdy uruchomiony jest FB typu operacyjnego i wykonywany jest następny FB typu operacyjnego, operacja przerywania przerywa wykonywany FB i wykonuje następny FB.
Buforowanie	<ul style="list-style-type: none"> • Gdy uruchomiony jest FB typu operacyjnego i wykonywany jest następny FB typu operacyjnego, operacja buforowania oczekuje na zakończenie wykonywanego FB, po czym wykonuje następny FB.
Łączenie	<ul style="list-style-type: none"> • Gdy uruchomiony jest FB typu operacyjnego i wykonywany jest następny FB typu operacyjnego, operacja łączenia wykonuje następny FB bez zatrzymywania operacji aktualnie wykonywanego FB. • W przypadku operacji łączenia dostępne są cztery metody przełączania prędkości: BlendingLow, BlendingHigh, BlendingPrevious i BlendingNext.
Przykładowy program	<ul style="list-style-type: none"> • Wybierz tryb buforowania za pomocą wejścia BufferMode FB operacji.
Kontrola działania	<ul style="list-style-type: none"> • Różnice w działaniu poszczególnych trybów buforowania zostały sprawdzone na filmie.

Rozdział 4 Praca z procesorem sterownika PLC

Pobierz przykładowy program do wykorzystania w tym rozdziale, klikając poniższe łącze.
Treść programu jest taka sama jak przykładowego programu z rozdziału 2 i 3.
Inna jest tylko metoda programowania.

[RD78GBasic2_sample2.zip \(1.39 MB\)](#)

4.1 Rejestrowanie biblioteki FB modułu ruchu

(1) Pobierz bibliotekę FB

Motion control FB można używać w programie procesora sterownika PLC, rejestrując bibliotekę FB dla modułu ruchu w GX Works3.

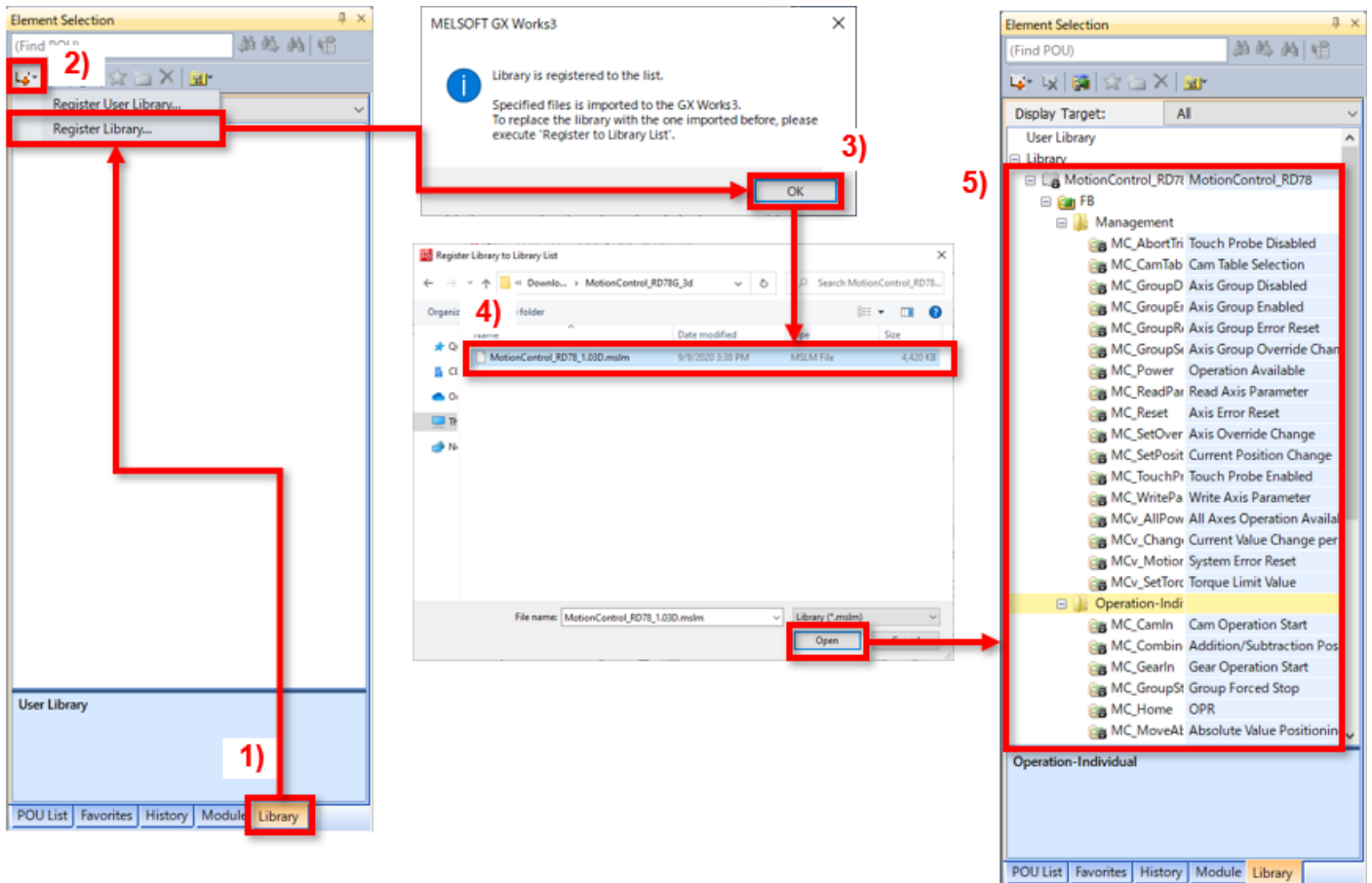
Pobierz bibliotekę FB z poniższego łącza i rozpakuj plik ZIP dożądanego miejsca docelowego.

[MotionControl_RD78G_3d.zip\(4.29 MB\)](#)

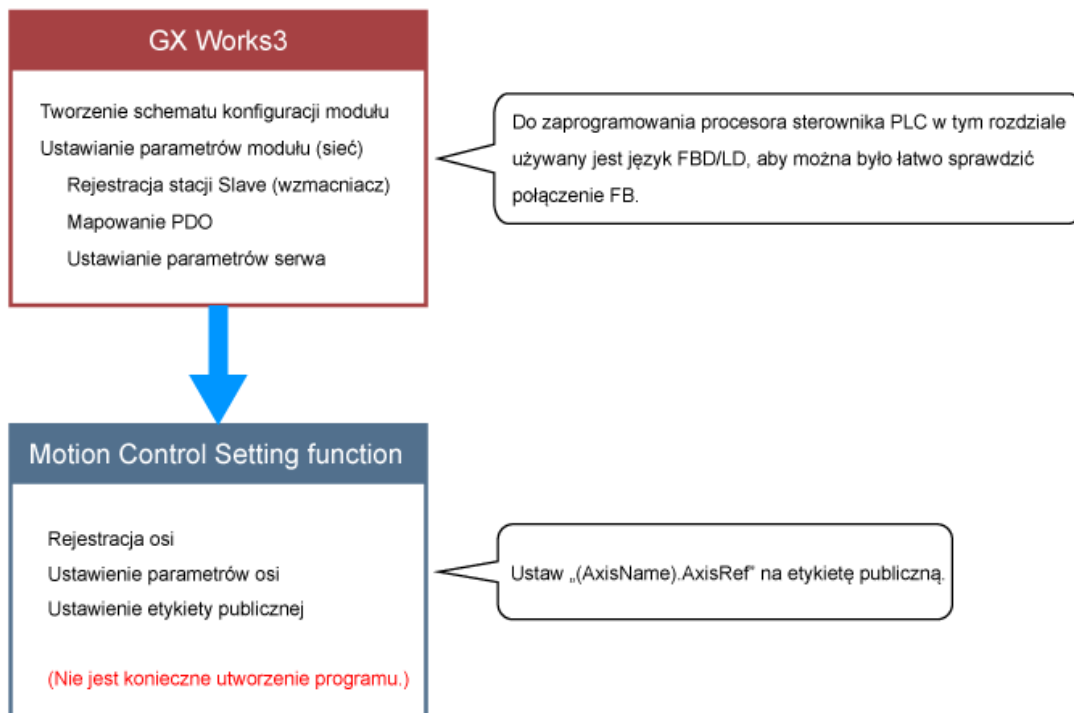
(Uwaga) Najnowszą wersję biblioteki FB można pobrać ze strony internetowej MITSUBISHI ELECTRIC FA.

(2) Rejestrowanie biblioteki FB

- 1) Otwórz dowolny projekt w GX Works3, a następnie przejdź do karty Library w oknie Element Selection.
- 2) Kliknij przycisk [Register to Library List] w górnej części i wybierz opcję [Register Library].
- 3) Po wyświetleniu komunikatu „Library is registered to the list” kliknij przycisk [OK].
- 4) Wybierz plik biblioteki FB „MotionControl_RD78_****.mslm”, a następnie kliknij przycisk [Open].
(**** wskazuje wersję).
- 5) Motion control FB jest zarejestrowany w bibliotece w oknie Element Selection.



Procedura tworzenia projektu jest taka sama jak w poprzednim rozdziale.



MELSOFT GX Works3 E: Sample.gx3 - [Global [Global Label Setting]]

Project Edit Find/Replace Convert View Online Debug Recording Diagnostics Tool Window Help

Navigation Global [Global Label Setting] ProgPou [PRG] [Local Label Sett... ProgramBody : ProgPou [PRG] [...]

Project

- Module Configuration
- Program
 - Initial
 - Scan
 - MAIN
 - ProgPou
 - Local Label
 - ProgramBody
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
 - FB/FUN
 - Label
 - Global Label
 - Global
 - M+RD78_0000
 - Structured Data Types
 - Device
 - Parameter

Global [Global Label Setting]

Label Name	Data Type	English/Display Target	Access from External Device
G.JogFW	Bit	JOG Forward	<input type="checkbox"/>
G.JogBW	Bit	JOG Backward	<input type="checkbox"/>
G.JogVelocity	FLOAT [Double Precision]	JOG Velocity	<input type="checkbox"/>
G.JogBusy	Bit	JOG Busy	<input type="checkbox"/>

Kliknij przycisk odtwarzania.

Extended Display Do Not Show Always

System label is reserved to be registered. System label is reserved to be released. The system label is already registered to the

To execute the Reservation to Register/Release for the system label, reflection to the system label database is required. Please execute 'Reflect to System Label Database'. It is unnecessary to change reference side project when assigned device is changed in system label Ver.2.
 * Only Q-R series/GOT 2000 series is available for system label Ver.2.
 * To execute Online Program Change, execute Online Program Change and save.

Reservation to Register System Label
 Reservation to Release System Label
 Import System Label

Not Reflected: 0
 Total: 0

Element Selection (Find POU)

Display Target: All

User Library
 Library

Library

POU... Favori... History Mod... Library

Output Progress

R04 Host Row 1 Column 1 CAP NUM

MELSOFT GX Works3 E: Sample.gx3 - [Global [Global Label Setting]]

Project Edit Find/Replace Convert View Online Debug Recording Diagnostics Tool Window Help

Navigation Global [Global Label Setting] ProgPou [PRG] [Local Label Sett... ProgramBody : ProgPou [PRG] [...]

Label Name	Data Type	English (Display Target)	Access from External Device
G.L-JogFW	Bit	JOG Forward	<input type="checkbox"/>
G.L-JogBW	Bit	JOG Backward	<input type="checkbox"/>
G.L-JogVelocity	FLOAT (Double Precision)	JOG Velocity	<input type="checkbox"/>
G.L-JogBusy	Bit	JOG Busy	<input type="checkbox"/>

Display Target: All

User Library

Library

Extended Display Do Not Show Always

System label is reserved to be registered. System label is reserved to be released. The system label is already registered to the

To execute the Reservation to Register/Release for the system label, reflection to the system label database is required. Please execute 'Reflect to System Label Database'. It is unnecessary to change reference side project when assigned device is changed in system label Ver.2.
 * Only Q-R series/GOT 2000 series is available for system label Ver.2.
 * To execute Online Program Change, execute Online Program Change and save.

Reservation to Register System Label

Reservation to Release System Label

Import System Label

Not Reflected: 0
Total: 0

Library

POU... Favori... History Mod... Library

Row 1 Column 1

NUM

Ten film przedstawia sposób tworzenia programu FB (MCv_Jog) dla operacji JOG.

The screenshot shows the MELSOFT GX Works3 E interface. The main window is titled "Global [Global Label Setting]". It contains a table with the following data:

Label Name	Data Type	English/Display Target	Access from External Device
G.JogFW	Bit	JOG Forward	<input type="checkbox"/>
G.JogBk	Bit	JOG Backward	<input type="checkbox"/>
G.JogVelocity	FLOAT (Double Precision)	JOG Velocity	<input type="checkbox"/>
G.JogBusy	Bit	JOG Busy	<input type="checkbox"/>

Below the table, there is a section for "Reservation to Register/Release for the system label". It includes the following text:

Extended Display: Do Not Show Always

System label is reserved to be registered. System label is reserved to be released. The system label is already registered to the

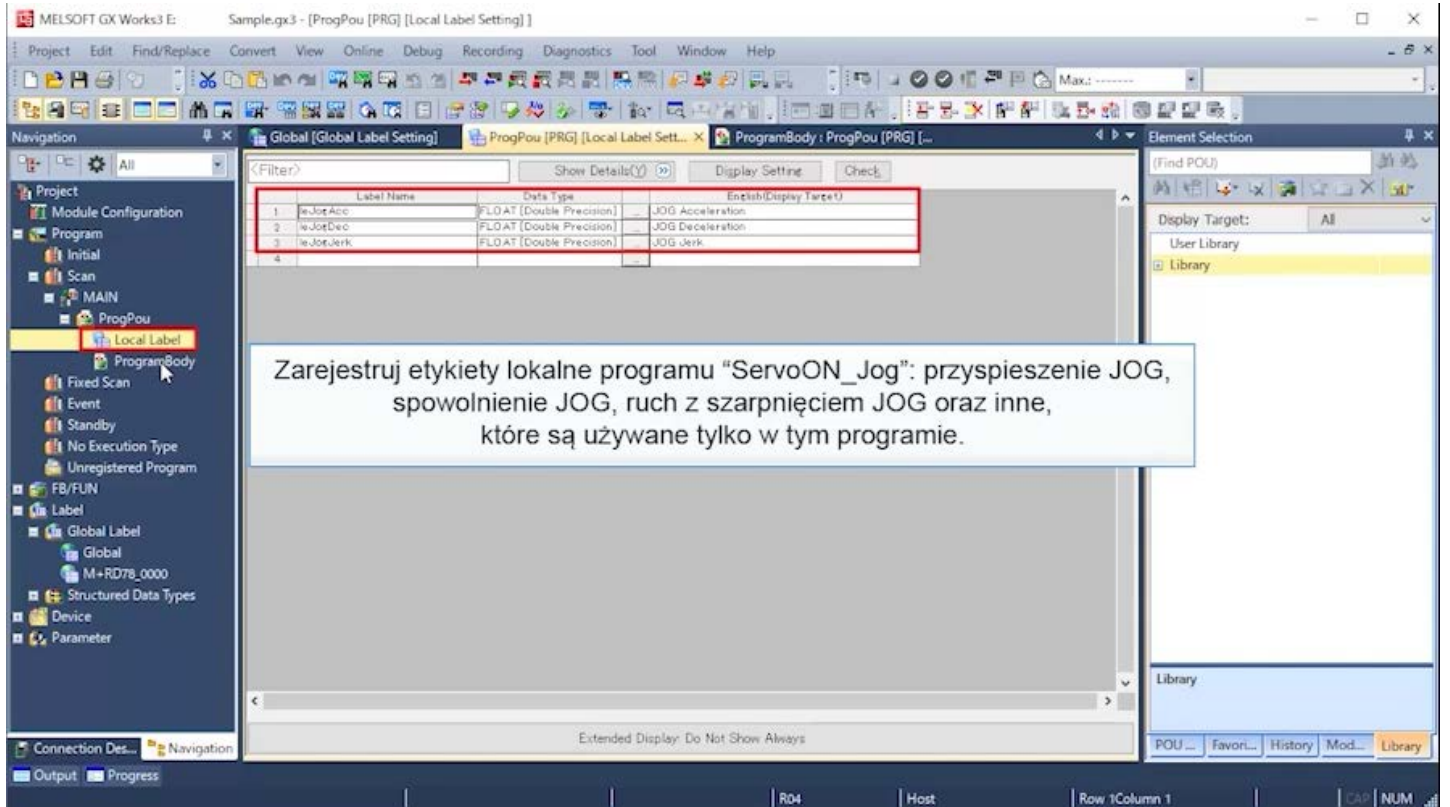
To execute the Reservation to Register/Release for the system label, reflection to the system label database is required. Please execute 'Reflect to System Label Database'. It is unnecessary to change reference side project when assigned device is changed in system label Ver.2.
 * Only Q-R series/GOT 2000 series is available for system label Ver.2.
 * To execute Online Program Change, execute Online Program Change and save.

Buttons: Reservation to Register System Label, Reservation to Release System Label, Import System Label

Not Reflected: 0 Total: 0

A text box overlay on the screenshot contains the following text:

Zarejestruj polecenie JOG, prędkość JOG i zajętość JOG w etykietach globalnych. (Prędkość JOG rejestrujemy się w etykiecie globalnej zakładając możliwość ustawiania jej przy pomocy urządzeń zewnętrznych, takich jak GOT).

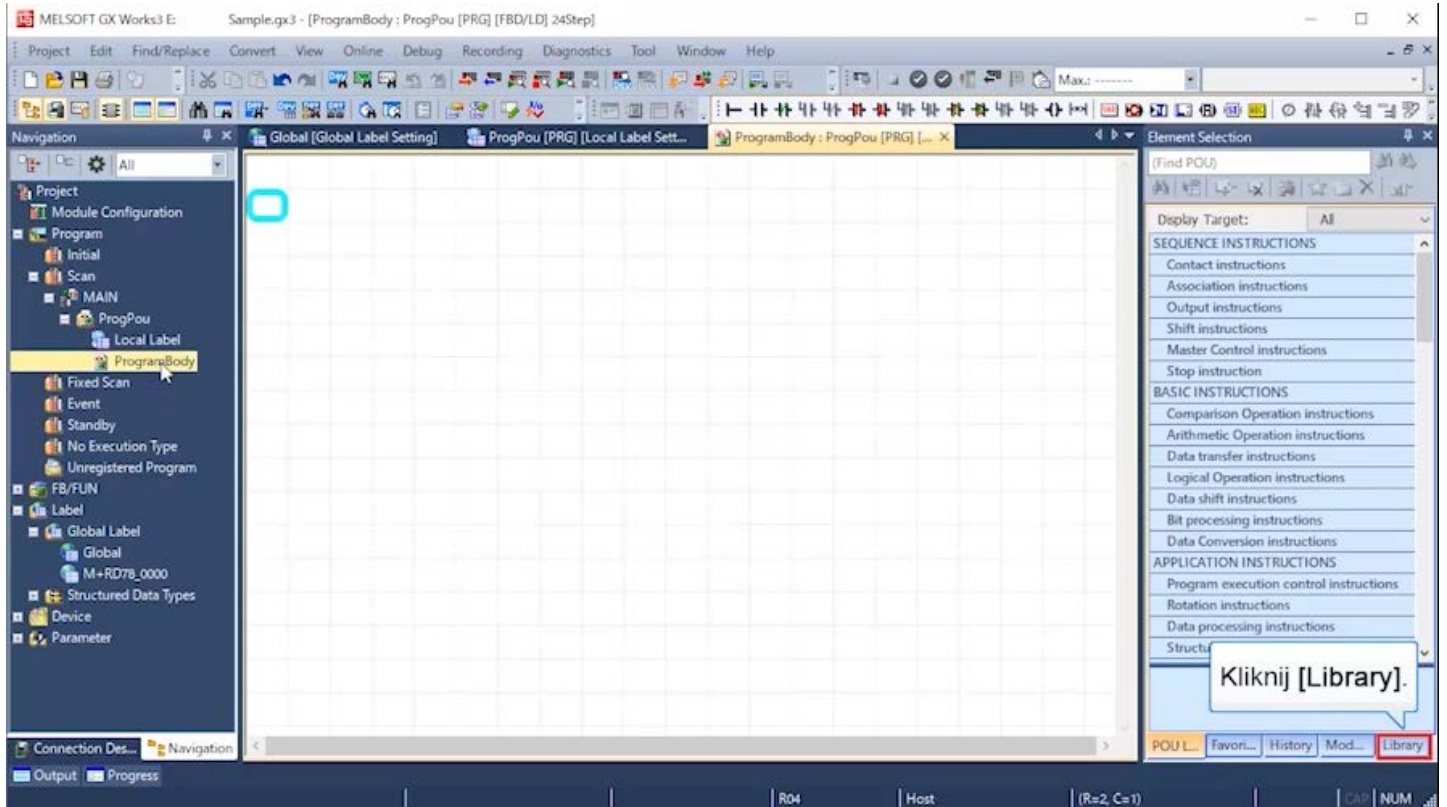


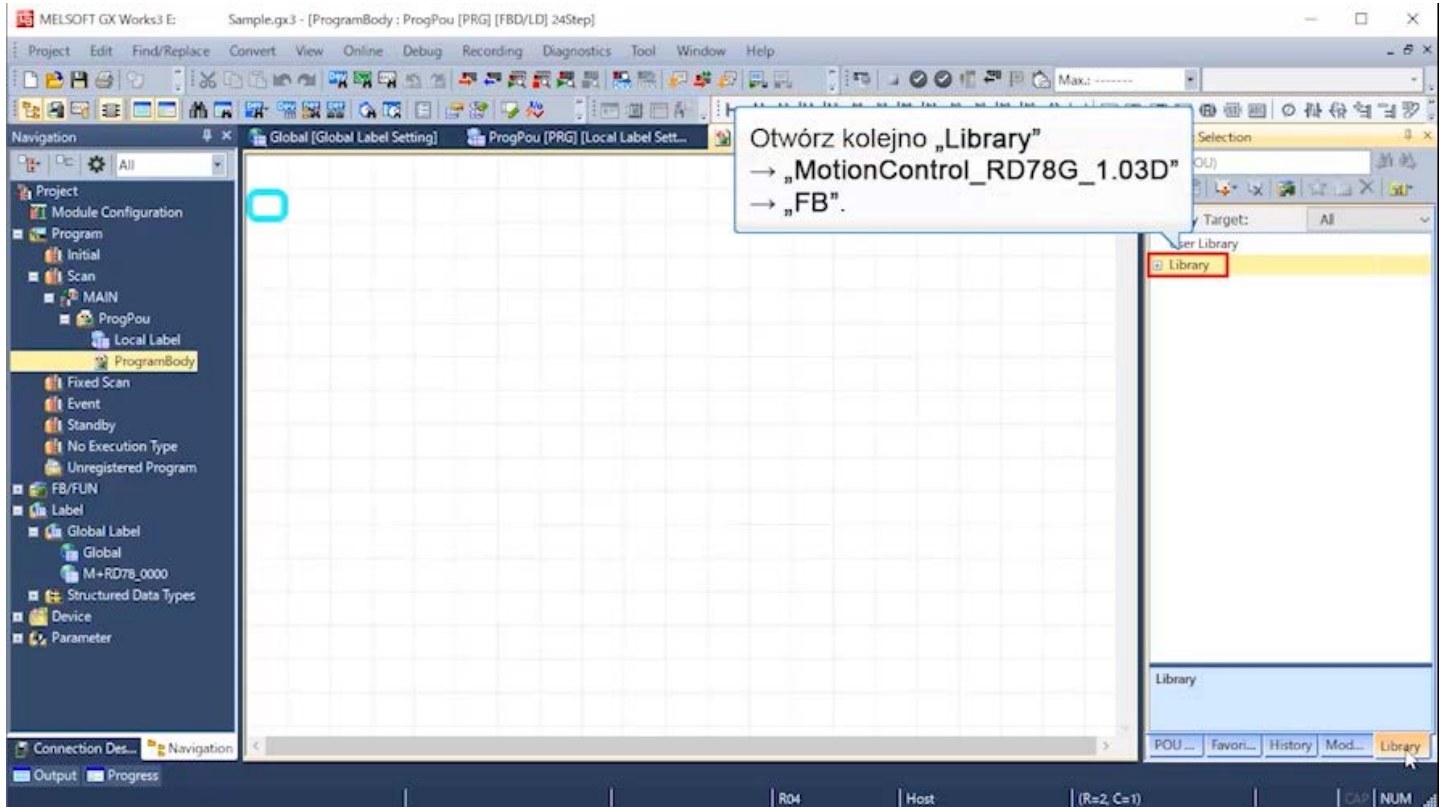
The screenshot shows the MELSOFT GX Works3 E software interface. The main window is titled "Sample.gx3 - [ProgPou [PRG] [Local Label Setting]]". The interface includes a menu bar, a toolbar, and a navigation pane on the left. The navigation pane shows a tree view of the project structure, with "Local Label" selected under "ProgPou". The main window displays a table of local labels with the following data:

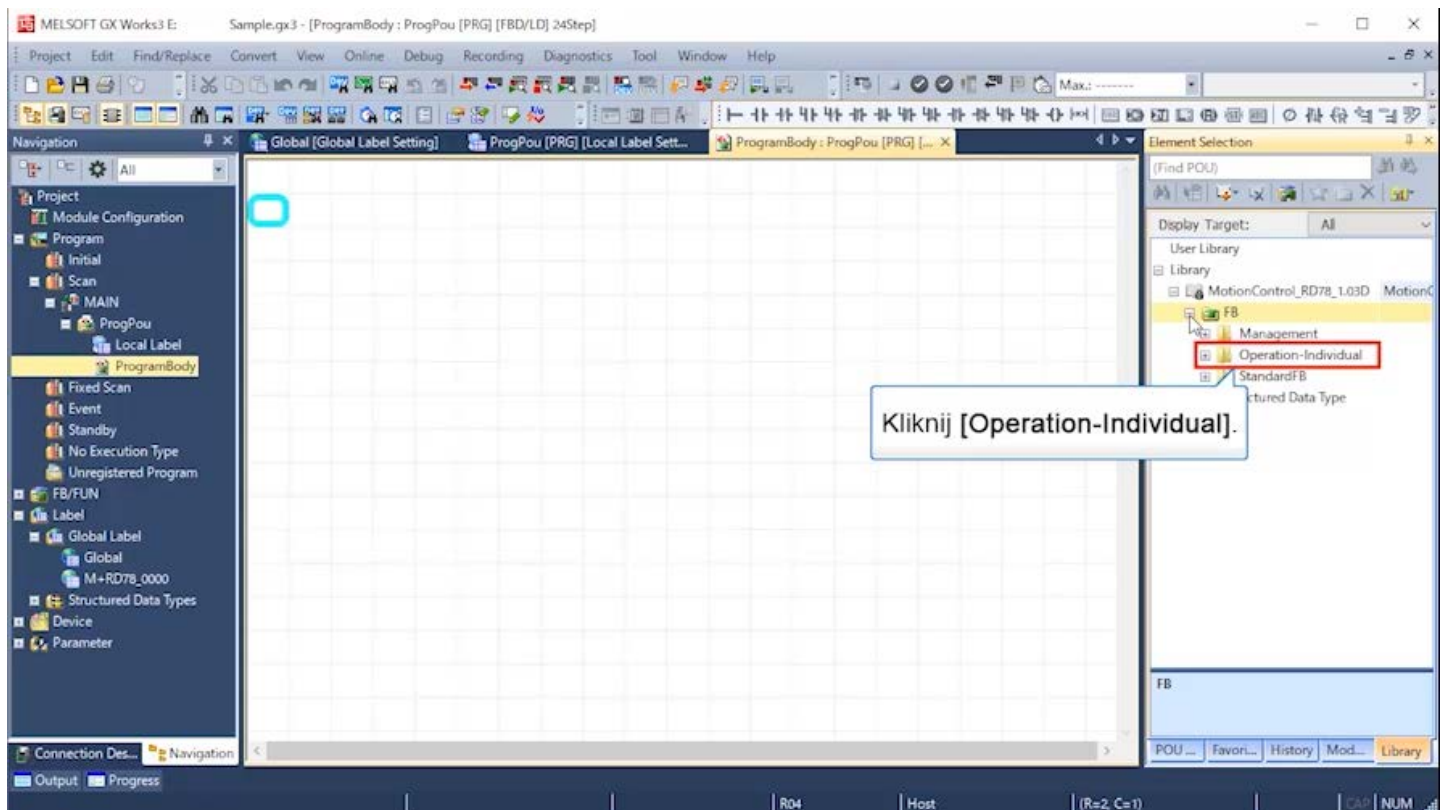
	Label Name	Data Type	English/Display Target
1	%JogAcc	FLOAT (Double Precision)	JOG Acceleration
2	%JogDec	FLOAT (Double Precision)	JOG Deceleration
3	%JogJerk	FLOAT (Double Precision)	JOG Jerk
4			

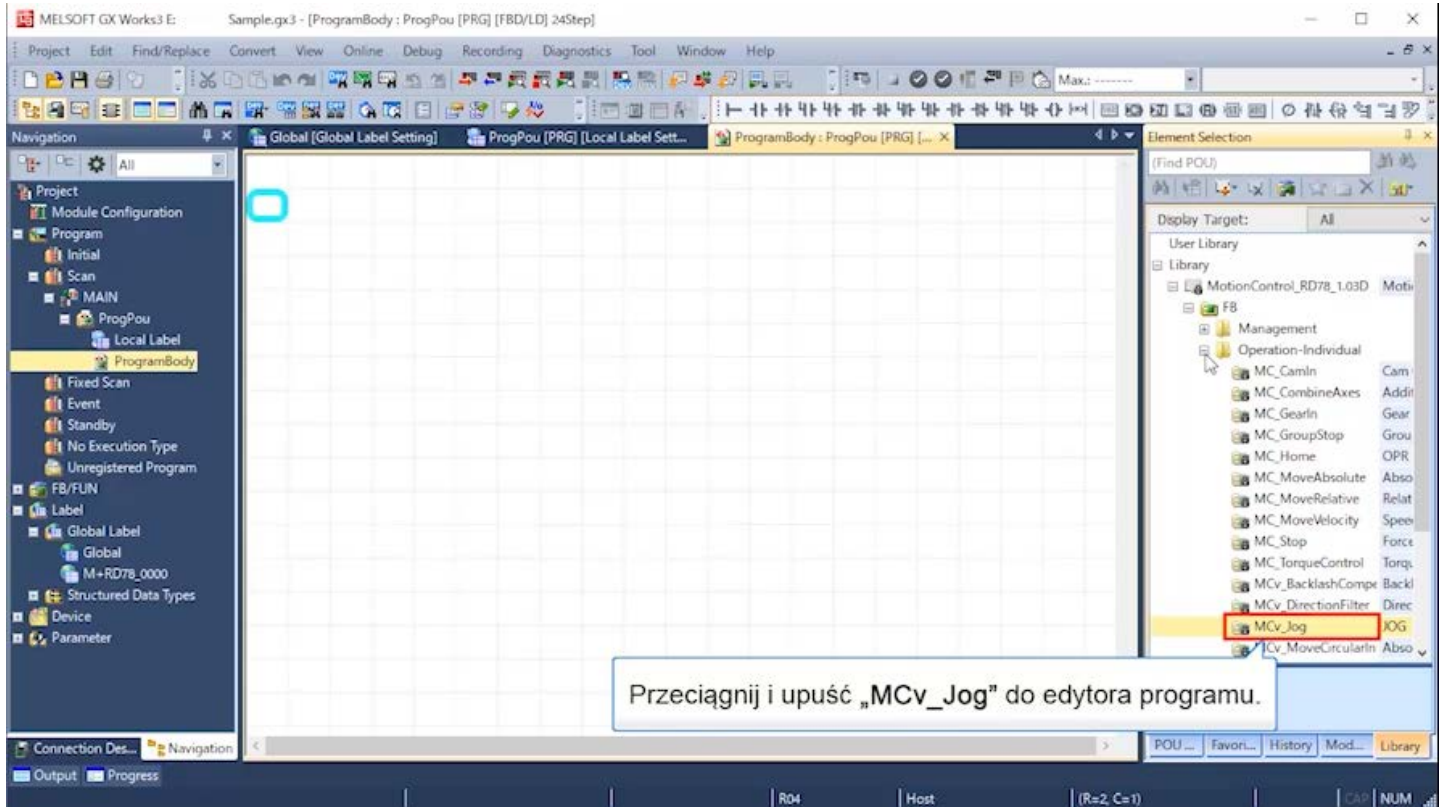
A text box is overlaid on the table with the following text:

Zarejestruj etykiety lokalne programu "ServoON_Jog": przyspieszenie JOG, spowolnienie JOG, ruch z szarpnięciem JOG oraz inne, które są używane tylko w tym programie.



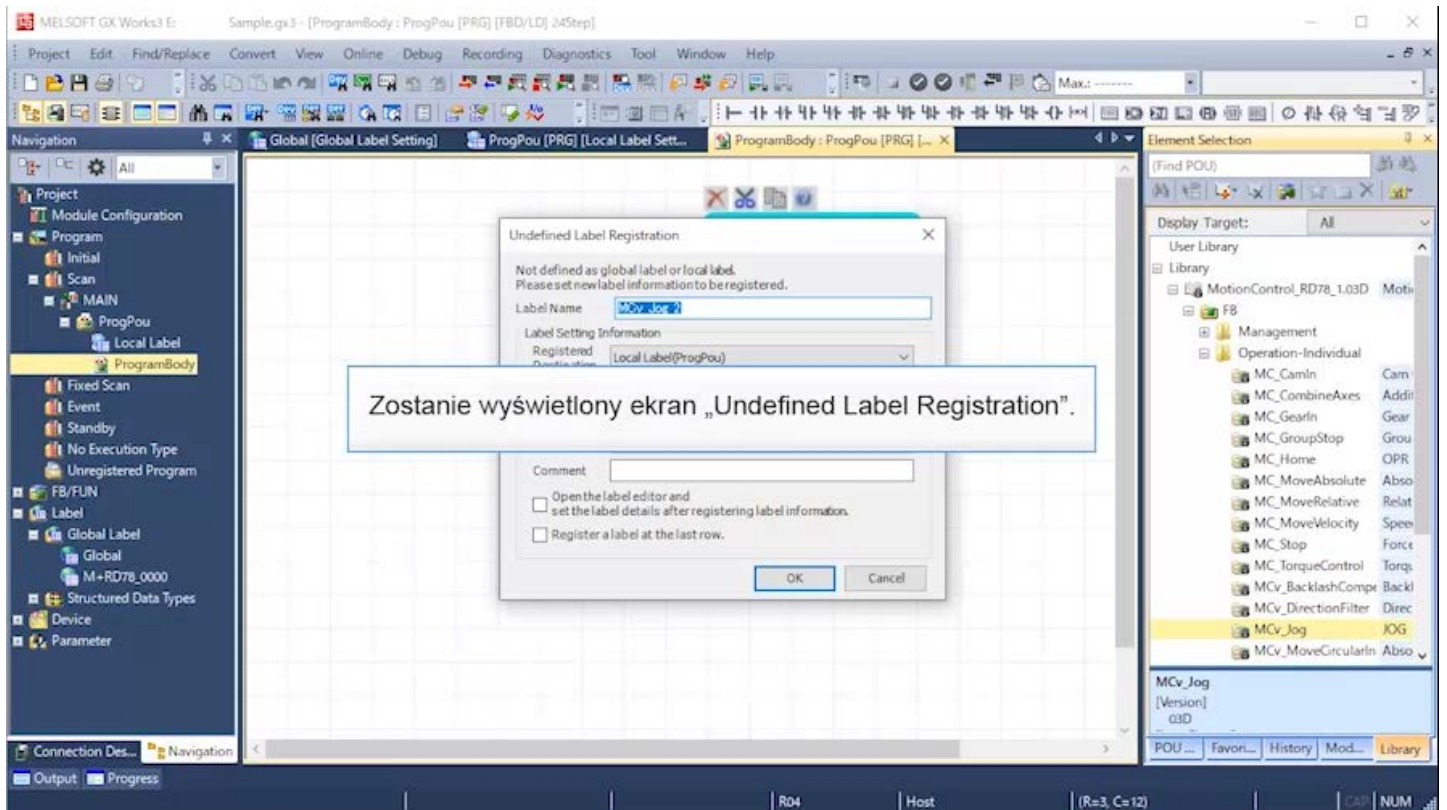






The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace is a grid editor. On the left, the 'Navigation' pane shows the project structure, with 'ProgramBody' selected. On the right, the 'Element Selection' window is open, showing a library of Motion Control Function Blocks (FB). The 'MCv_Jog' block is highlighted with a red box. A text box at the bottom of the grid editor contains the instruction: "Przecignij i upuść „MCv_Jog” do edytora programu." (Drag and drop "MCv_Jog" into the program editor.)

Przecignij i upuść „MCv_Jog” do edytora programu.



Undefined Label Registration

Not defined as global label or local label.
Please set new label information to be registered.

Label Name: MCv_log_2

Label Setting Information

Registered Destination: Local Label(ProgPou)

Class: VAR

Data Type: MCv_log

Constant: [Empty]

Comment: [Empty]

Open the label editor and set the label details after registering label information.

Register a label at the last row.

OK Cancel

Kliknij przycisk [OK].

The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace shows a ladder logic diagram with a grid. A blue rectangular block, representing a Motion Control Function Block (FB), is being placed on the grid. The block is labeled "MCv_Jog_2" and "MCv_Jog". It contains several input and output fields: "JogForward" (Done), "JogBackward" (Busy), "Velocity" (Active), "Acceleration" (CommandAborted), "Deceleration" (Error), "Jerk" (ErrorID), "Options", and "Axis". A red square on the grid indicates the target location for the block. A callout box with the text "Kliknij obszar, w którym ma zostać umieszczony element LD." (Click the area where the LD element should be placed.) points to this red square. The left sidebar shows the project structure, including "ProgramBody" and "Local Label". The right sidebar shows the "Element Selection" panel, which lists various Motion Control FBs, including "MCv_Jog" (JOG). The bottom status bar shows "R04", "Host", "(R=3, C=12)", and "CAP NUM".

Kliknij obszar, w którym ma zostać umieszczony element LD.

Field	Value/Label
JogForward	Done
JogBackward	Busy
Velocity	Active
Acceleration	CommandAborted
Deceleration	Error
Jerk	ErrorID
Options	
Axis	

The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace shows a grid with a blue square icon being selected. A callout box with the text "Kliknij ikonę elementu LD." (Click the LD element icon) points to this icon. The right-hand side of the interface features the "Element Selection" panel, which lists various Motion Control Function Blocks (FB) under the "MotionControl_RD78_1.03D" library. The "MCv_Jog" block is highlighted in the list. The bottom status bar shows "R04 | Host | (R=3, C=4) | CAP | NUM".

Kliknij ikonę elementu LD.

MCv_Jog_2	Done
JogForward	Done
JogBackward	Busy
Velocity	Active
Acceleration	CommandAborted
Deceleration	Error
Jerk	ErrorID
Options	
Axis	

MCv_Jog
[Version]
03D

POU... Favori... History Mod... Library

The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace shows a ladder logic diagram with a blue-bordered block labeled "MDv_Jog_3" and "MCv_Jog". The block has several input and output ports: "JogForward", "Done", "Acceleration", "CommandAborted", "Deceleration", "Error", "Jerk", "ErrorID", "Options", and "Axis". A callout box with a blue border and a white background points to a specific element in the diagram, containing the text: "Określony element LD zostanie umieszczony." (The specified LD element will be placed).

The interface includes a menu bar (Project, Edit, Find/Replace, Convert, View, Online, Debug, Recording, Diagnostics, Tool, Window, Help) and a toolbar. The left sidebar shows a project tree with "ProgramBody" selected. The right sidebar shows the "Element Selection" panel with a search bar and a list of motion control blocks, including "MCv_Jog".

The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace shows a ladder logic diagram with a function block call. A callout box with a white background and a black border contains the text "Kliknij dwukrotnie [???].". The function block is labeled "MDV_jog_2" and "MDV_jog". The block has several input and output ports: "Done", "Busy", "Active", "Error", "ErrorID", "Axis", "Options", "Jerk", "Deceleration", "Acceleration", "Velocity", and "MDV_Units". The "Axis" port is connected to a terminal labeled "1". The "Options" port is connected to a terminal labeled "1". The "Jerk" port is connected to a terminal labeled "1". The "Deceleration" port is connected to a terminal labeled "1". The "Acceleration" port is connected to a terminal labeled "1". The "Velocity" port is connected to a terminal labeled "1". The "MDV_Units" port is connected to a terminal labeled "1".

The left sidebar shows the "Project" tree with the following structure:

- Project
 - Module Configuration
 - Program
 - Initial
 - Scan
 - PrologPou
 - Local Label
 - ProgramBody
 - Fixed Scan
 - Event
 - Standby
 - No Execution Type
 - Unregistered Program
 - FB/FUN
 - Label
 - Global Label
 - M+RD78_0000
 - Structured Data Types
 - Device
 - Parameter

The right sidebar shows the "Element Selection" panel with the following structure:

- Display Target: All
- User Library
 - Library
 - MotionControl_RD78_1.03D
 - Moti
 - FB
 - Management
 - Operation-Individual
 - MC_CamIn Cam
 - MC_CombineAxes Addi
 - MC_GearIn Gear
 - MC_GroupStop Grou
 - MC_Home OPR
 - MC_MoveAbsolute Abso
 - MC_MoveRelative Relat
 - MC_MoveVelocity Spee
 - MC_Stop Force
 - MC_TorqueControl Torq
 - MCV_BacklashComp Backl
 - MCV_DirectionFilter Direc
 - MCV_Jog JOG
 - MCV_MoveCircularIn Abso

The bottom status bar shows the following information: R04 | Host | (R=3, C=8) | CAP | NUM

MELSOFT GX Works3 E: Sample.gx3 - [ProgramBody : ProgPou [PRG] [FBD/LD] 24Step *]

Project Edit Find/Replace Convert View Online Debug Recording Diagnostics Tool Window Help

Navigation Global [Global Label Setting] Wprowadź nazwę etykiety operandu bitowego.

Element Selection (Find POU)

Display Target: All

User Library

Library

- MotionControl_RD78_1.03D Moti
 - FB
 - Management
 - Operation-Individual
 - MC_CamIn Cam
 - MC_CombineAxes Addit
 - MC_GearIn Gear
 - MC_GroupStop Grou
 - MC_Home OPR
 - MC_MoveAbsolute Abso
 - MC_MoveRelative Relat
 - MC_MoveVelocity Speec
 - MC_Stop Forcc
 - MC_TorqueControl Torq
 - MCv_BacklashComp Backl
 - MCv_DirectionFilter Direc
 - MCv_Jog JOG
 - MCv_MoveCircularIn Abso

MCv_Jog [Version] 03D

POU ... Favori... History Mod... Library

Global [Global Label Setting]

Wprowadź nazwę etykiety operandu bitowego.

G

G_bJogBusy	BOOL	JOG Busy
G_bJogBW	BOOL	JOG Backward
G_bJogFW	BOOL	JOG Forward
G_leJogVelocity	LREAL	JOG Velocity

Settings...

MDv_Jog_0

MDv_Jog

JogForward	Done
JogBackward	Busy
Velocity	Active
Acceleration	CommandAborted
Deceleration	Error
Jerk	ErrorID
Options	
Axis	

1

Connection Des... Navigation

Output Progress

R04 Host (R=3, C=5) CAP NUM

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram with a callout box containing the text: "Etykiety już zarejestrowane w etykietach globalnych są wyświetlane jako rekomendacje." (Labels already registered in global labels are displayed as recommendations.)

The callout box highlights a table of global labels:

Label	Type	Description
G_bJogBusy	BOOL	JOG Busy
G_bJogBW	BOOL	JOG Backward
G_bJogFW	BOOL	JOG Forward
G_leJogVelocity	LREAL	JOG Velocity

The right-hand side of the interface shows the "Element Selection" panel with a tree view of the "MotionControl_RD78_1.03D" library. The "MCv_Jog" block is highlighted in the library, and its details are shown at the bottom of the panel:

MCv_Jog
[Version]
03D

The bottom status bar shows "R04", "Host", "(R=3, C=5)", and "CAP NUM".

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram with a normally closed contact labeled 'G_bJogFW' and a normally open contact labeled 'G_bJogBW'. A callout box with a red border points to the 'G_bJogBW' contact, containing the text: "Połącz element LD z wejściem JogForward FB." (Connect the LD element to the JogForward input of the FB).

To the right, the 'Element Selection' panel is open, showing a library of function blocks. The 'MCv_Jog' block is highlighted. The 'MCv_Jog' block has the following inputs and outputs:

Input	Output
JogForward	Done
JogBackward	Busy
Velocity	Active
Acceleration	CommandAborted
Deceleration	Error
Jerk	ErrorID
Options	
Axis	

The 'MCv_Jog' block is identified as 'JOG' in the library. The status bar at the bottom shows 'R04', 'Host', '(R=3, C=8)', and 'CAP NUM'.

Może to wyglądać tak, gdy element LD jest podłączony do wejścia JogForward i wejścia JogBackward.

The screenshot displays the MELSOFT GX Works3 E software interface. The main workspace shows a ladder logic diagram for a motion control function block named 'MDV_Jog_2'. The diagram consists of two normally open contacts labeled 'G.L.JogFW' (1) and 'G.L.JogBW' (2) connected to the 'JogForward' input of the block. A second set of contacts labeled 'G.L.JogBW' (3) and 'G.L.JogFW' (4) is connected to the 'JogBackward' input. A red 'X' is placed over the 'JogForward' input, and a blue square highlights the 'JogBackward' input. The block has various status outputs like Done, Busy, Active, Error, etc. The software interface includes a navigation tree on the left, a top menu bar, and an element selection panel on the right showing a library of motion control functions.

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram with four normally open contacts labeled G.L. JogFW (1, 2, 3, 4) connected to a function block call for MDV_Jog_2. The function block has inputs for JogForward, JogBackward, Velocity, and Active, and outputs for Done, Busy, and Active. A callout box with a white background and black border points to the area where the function block is being inserted, containing the text: "Kliknij obszar, w którym ma zostać umieszczony element FBD." The right-hand side of the screen shows the Element Selection panel with a search bar and a tree view of the User Library. The tree view is expanded to show the MotionControl_RD78_1.03D library, with the MCv_Jog block selected. The bottom status bar shows the current location as R04, Host, and coordinates (R=5, C=10).

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram with a function block call. A callout box points to the call, containing the text: "Wprowadź etykietę globalną „G_leJogVelocity”." Below the callout, a tooltip for the function block "G_leJogVelocity" is visible, showing parameters: LREAL JOG Velocity, Active, CommandAborted, Error, ErrorID, Axis, and Jerk. The left sidebar shows the project structure, including "Global Label" and "Local Label" sections. The bottom status bar shows "R04", "Host", "(R=6, C=8)", and "CAP NUM".

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram for a motion control function block. The diagram includes several global labels (G.L.) and a function block (MDV_Jog_2) with inputs like JogForward and JogBackward. A callout box highlights the 'G.L. JogVelocity' label and the 'REAL JOG Velocity' parameter, stating that registered global labels are shown as recommendations.

Etykiety już zarejestrowane w etykietach globalnych są wyświetlane jako rekomendacje.

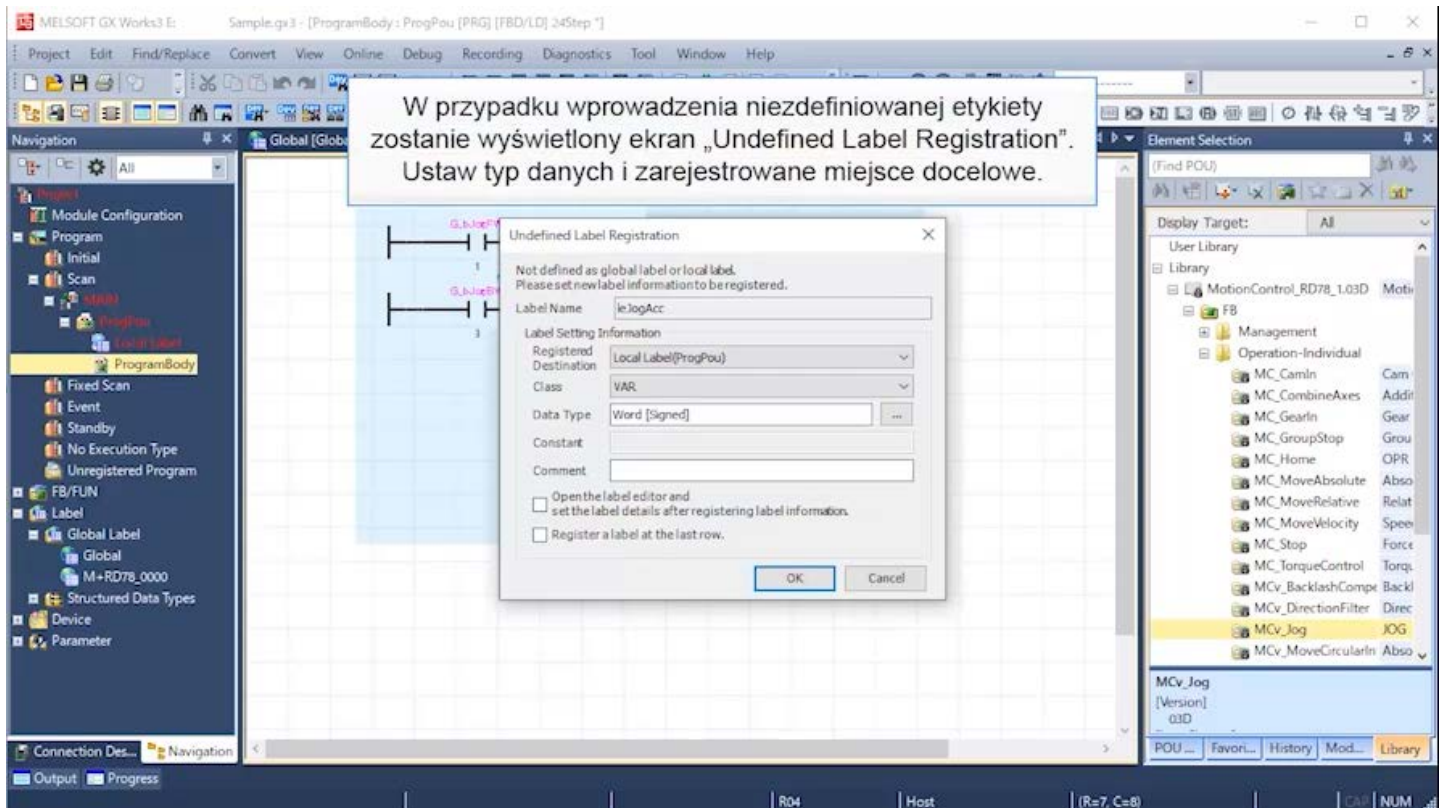
Połącz element FBD z wejściem Velocity FB.

Input	Output
JogForward	Done
JogBackward	Busy
Velocity	Active
Acceleration	CommandAborted
Deceleration	Error
Jerk	ErrorID
Options	
Axis	

MCv_Jog
[Version]
03D

POU... Favori... History Mod... Library

R04 Host (R=6, C=8) CAP NUM



W przypadku wprowadzenia niezdefiniowanej etykiety zostanie wyświetlony ekran „Undefined Label Registration”. Ustaw typ danych i zarejestruj miejsce docelowe.

Undefined Label Registration

Not defined as global label or local label.
Please set new label information to be registered.

Label Name: JogAcc

Label Setting Information

Registered Destination: Local Label(ProgPou)

Class: VAR

Data Type: Word [Signed]

Constant:

Comment:

Open the label editor and set the label details after registering label information.

Register a label at the last row.

OK Cancel

Navigation: Project, Module Configuration, Program, Initial, Scan, ProgramBody, Fixed Scan, Event, Standby, No Execution Type, Unregistered Program, FB/FUN, Label, Global Label, Global, M+RD78_0000, Structured Data Types, Device, Parameter.

Element Selection: Display Target: All, User Library, Library, MotionControl_RD78_1.03D, FB, Management, Operation-Individual, MC_CamIn, MC_CombineAxes, MC_GearIn, MC_GroupStop, MC_Home, MC_MoveAbsolute, MC_MoveRelative, MC_MoveVelocity, MC_Stop, MC_TorqueControl, MCv_BacklashComp, MCv_DirectionFilter, MCv_Jog, MCv_MoveCircularIn.

MCv_Jog [Version] 03D

POU... Favori... History Mod... Library

R04 Host (R=7, C=8) CAP NUM

The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram for a motion control function block (MCv_Jog). The diagram includes the following elements:

- Inputs:**
 - NO (Normally Open) contact labeled `G.JogFW` with value 1.
 - NC (Normally Closed) contact labeled `G.JogBW` with value 2.
 - NO contact labeled `G.JogBW` with value 1.
 - NO contact labeled `G.JogFW` with value 1.
 - Parameter inputs: `G.JogVelocity` (5), `G.JogAcc` (6), `G.JogDec` (7), `G.JogJerk` (8), and `G.JogOptions` (9).
- Function Block:** `MCv_Jog` (Axis 10). It has outputs for `JogForward`, `JogBackward`, `Velocity`, `Acceleration`, `Deceleration`, `Jerk`, `Options`, `Axis`, `Done`, `Active`, `CommandAborted`, `Error`, and `ErrorID`.
- Output:** A coil labeled `G.JogPulse` with value 11.

The right-hand pane shows the 'Element Selection' window with a library of motion control function blocks. A callout box with the text 'Kliknij [Module].' points to the 'Mod...' button in the library pane. The library structure is as follows:

- User Library
- Library
 - MotionControl_RD78_1.03D Moti
 - FB
 - Management
 - Operation-Individual
 - MC_CamIn Cam
 - MC_CombineAxes Addit
 - MC_GearIn Gear
 - MC_GroupStop Grou
 - MC_Home OPR
 - MC_MoveAbsolute Abso
 - MC_MoveRelative Relat
 - MC_MoveVelocity Spee
 - MC_Stop Forcc
 - MC_TorqueControl Torq
 - MCv_BacklashComp Backl
 - MCv_DirectionFilter Direc
 - MCv_Jog JOG

The status bar at the bottom shows 'R04 | Host | (R=5, C=16) | CAP | NUM'.

Otwórz kolejno „Module Label”
 → „0000:RD78G4” → „RD78_0000”
 → „Ax+Global” → „Axis0001”
 → „AxisRef”.

Module Label
 3E00-R04CPU
 0000-RD78G4
 Module FB

Module FB

PDU... Favor... History Module Library

R04 Host (R=5, C=16) CAP NUM

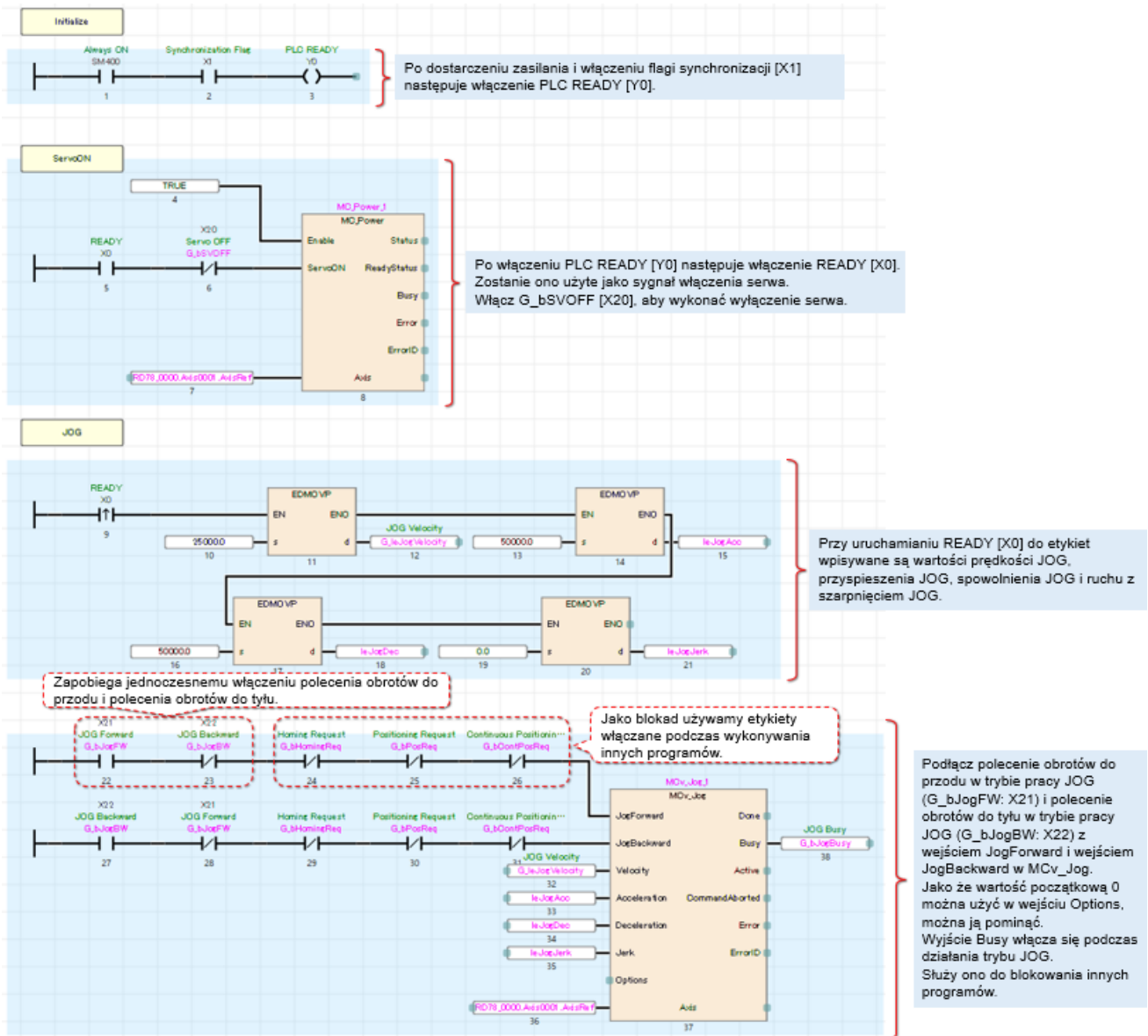
The screenshot displays the MELSOFT GX Works3 E interface. The main workspace shows a ladder logic diagram for a motion control function block (MOv_Axis). The diagram includes several input contacts (G.JokFV, G.JokBV, G.JokFV, G.JokBV) and output coils (G.JokBusy). The function block is connected to various motion control parameters such as Velocity, Acceleration, Deceleration, and Jerk. The project tree on the right shows the structure of the program, with the AxisRef structure highlighted in red. A callout box provides instructions on how to connect the AxisRef structure to the program editor.

Podłącz do wejścia Axis strukturę danych typu AxisRef zarejestrowaną jako etykieta publiczna. Przeciągnij i upuść „AxisRef” (informacje osi) do edytora programu.

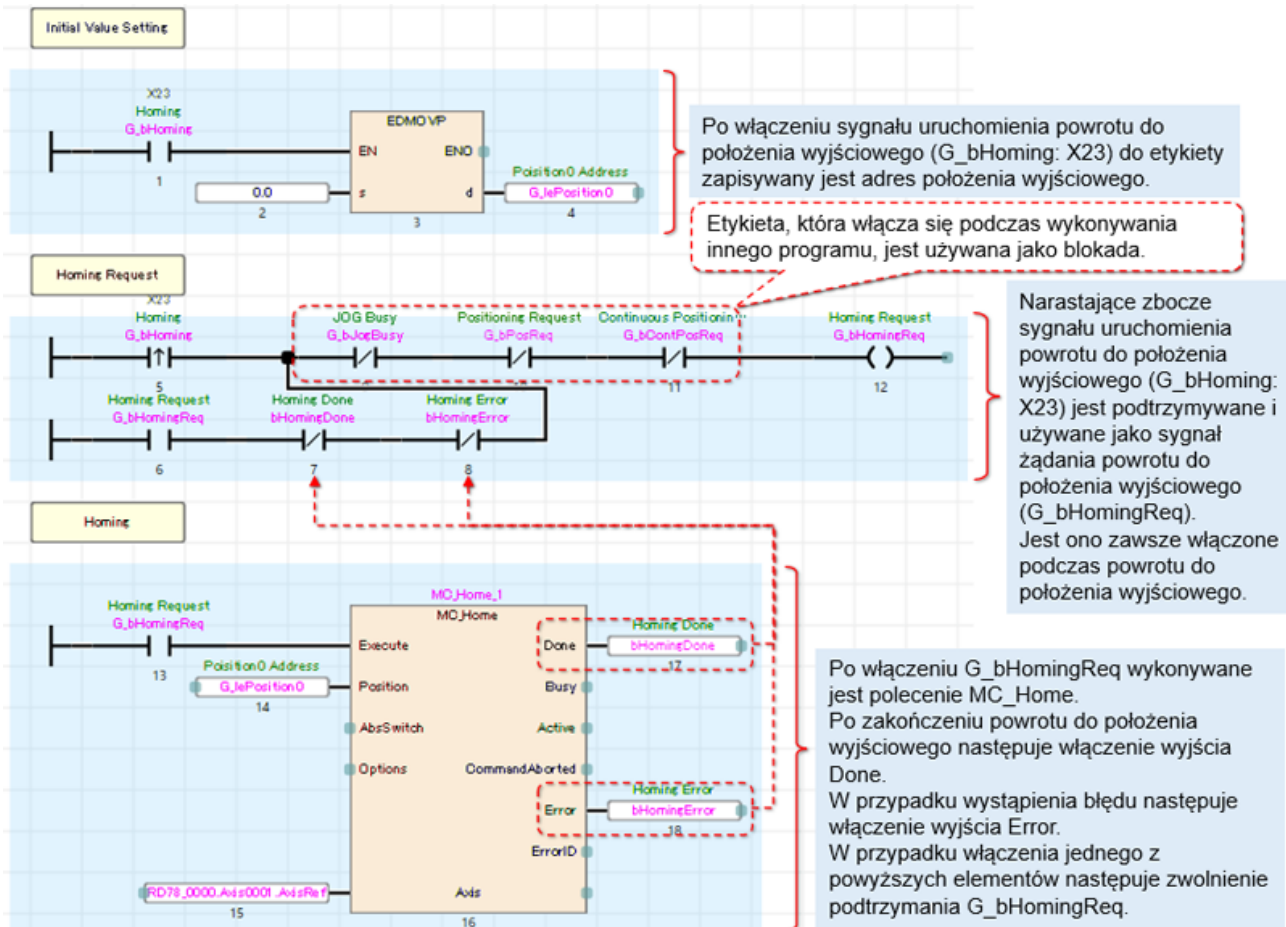
Na tym kończy się wprowadzanie danych w programie.
Kliknij , aby przejść do następnej strony.

(1) ServoON_Jog

Ten program wykonuje operacje PLC ready ON, servo ON i JOG.

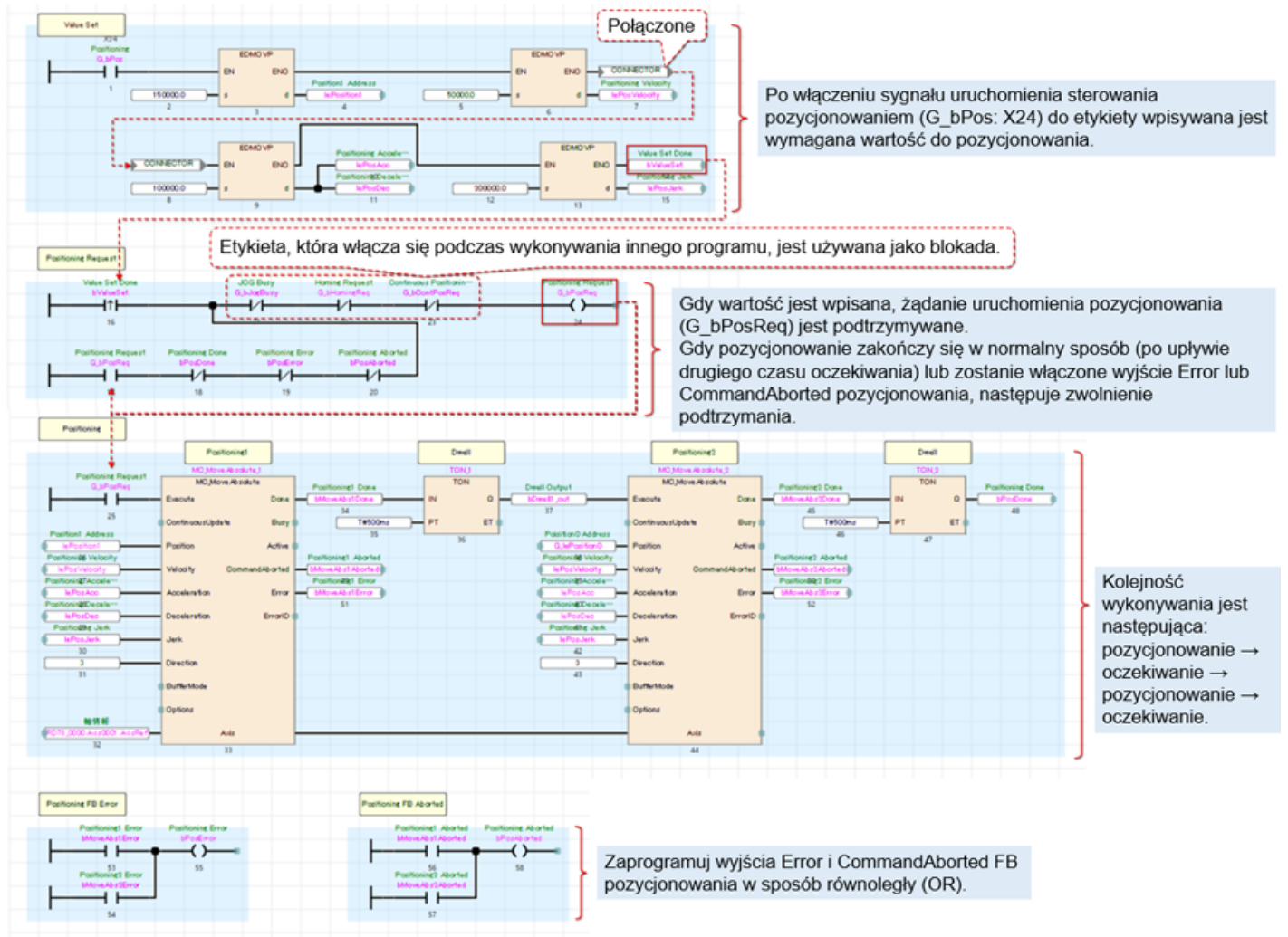


- (2) Powrót do położenia wyjściowego
 Ten program wykonuje powrót do położenia wyjściowego.



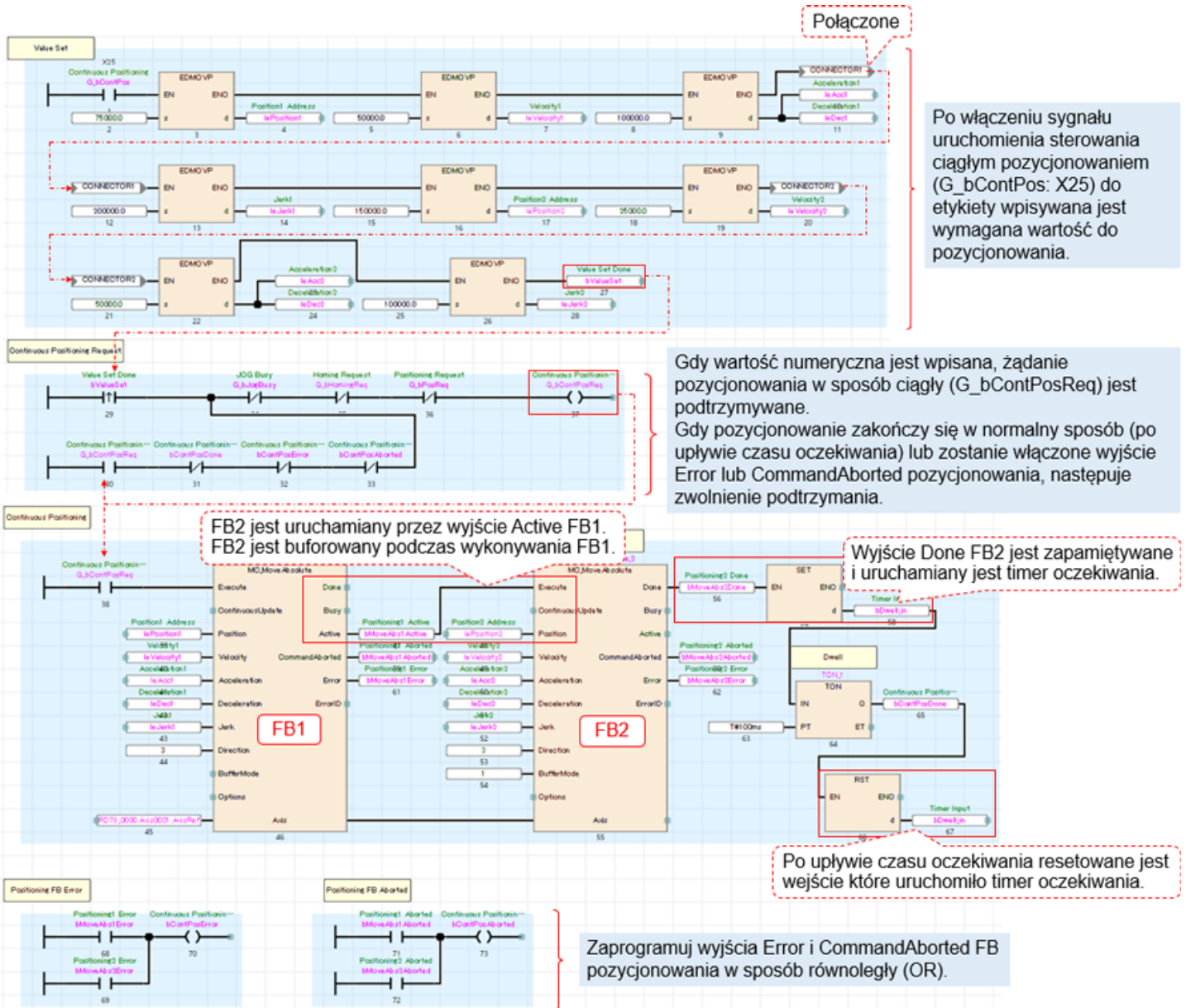
(3) Pozycjonowanie

Ten program wykonuje operację pozycjonowania pojedynczej osi.



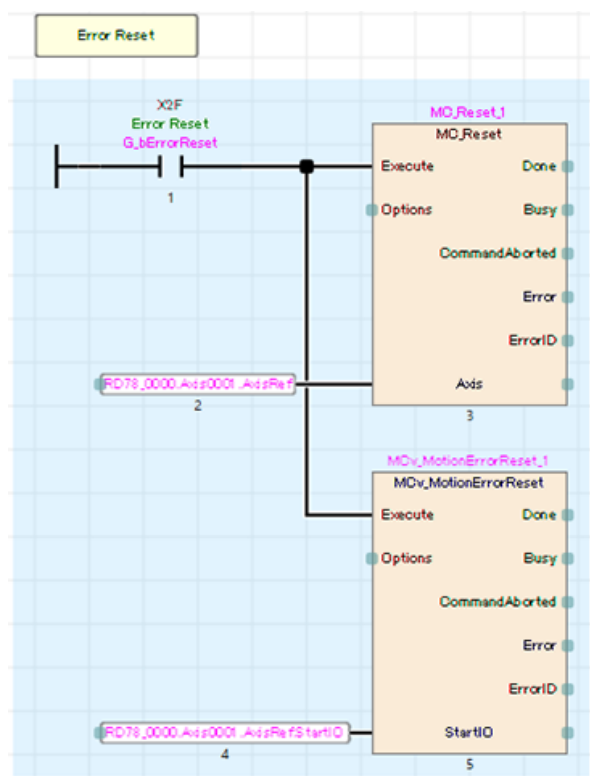
(4) ContinuousPositioning

Ten program wykonuje ciągłe pozycjonowanie przy użyciu trybu buforowania.



(5) ErrorReset

Ten program wykonuje reset błędów.



Po włączeniu sygnału resetowania błędu (G_bErrorReset: X2F) wykonywany jest reset błędu osi (MC_Reset) i reset błędu systemowego (MCv_MotionErrorReset).

(6) Monitor

Ten program zapisuje atrybuty SetPosition (Set Position) i SetVelocity (Set Velocity) etykiety globalnej monitora osi przypisanej do D0 i D2 procesora sterownika PLC.

Jako że SetPosition i SetVelocity są liczbami rzeczywistymi o podwójnej precyzji, są one konwertowane na typ słowa podwójnego ze znakiem, aby mogły być łatwo obsługiwane przez procesor sterownika PLC. (Uwaga)

Te środki nie są używane w temacie.

Są one używane do wyświetlania w innych programach sekwencyjnych i GOT, a także do innych celów. Ten program jest opisany za pomocą języka tekstowego (ST).

```
1 G_dSetPosition := LREAL_TO_DINT(RD78_0000.Axis0001.Md.SetPosition);  
2 G_dSetVelocity := LREAL_TO_DINT(RD78_0000.Axis0001.Md.SetVelocity);  
3
```



G_dSetPosition → D0
G_dSetVelocity → D2

(Uwaga) Gdy liczba rzeczywista o podwójnej precyzji jest konwertowana na słowo podwójne ze znakiem, a konwertowana wartość wykracza poza zakres od -2147483648 do 2147483647 to wystąpi błąd obliczeń.

Zapisz program i parametry do procesora sterownika PLC i modułu ruchu.

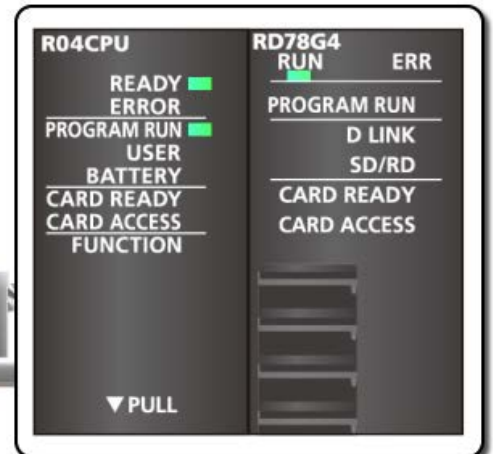
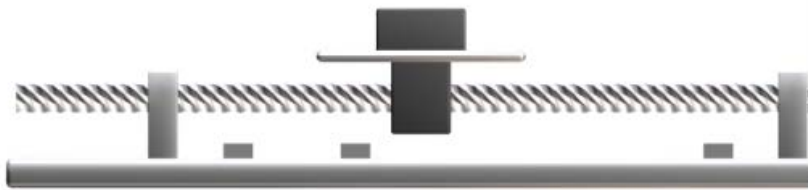
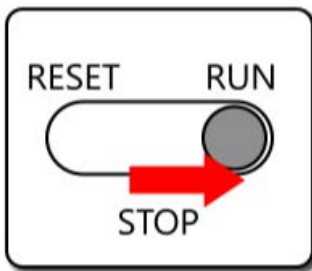
Program jest zapisywany tylko do procesora. Ustawienia parametrów osi i etykiety publicznej muszą być zapisane po stronie modułu ruchu.

- 1) Po przebudowaniu wszystkich programów w procesorze sterownika PLC wybierz kolejno [Online] → [Write to PLC] na pasku narzędzi GX Works3, aby zapisać wszystkie dane do procesora sterownika PLC.
- 2) Po zapisaniu parametrów do procesora sterownika PLC komunikacja z modulem ruchu zostaje włączona. Wybierz kolejno [Online] → [Write to Module] na pasku narzędzi Motion Control Setting Function, aby zapisać wszystkie dane do modułu ruchu.
- 3) Zresetuj procesor sterownika PLC, aby zakończyć operację zapisu.

Kliknij przycisk odtwarzania w lewym dolnym rogu okna.

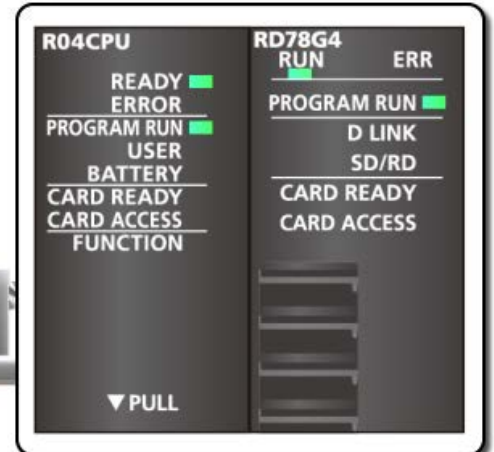
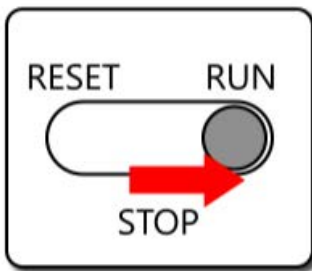


Sprawdź działanie przykładowego programu.
Przed uruchomieniem operacji upewnij się, że programy i parametry zostały wgrane do procesora sterownika PLC i modułu ruchu.



Ustaw przełącznik RUN/STOP/RESET procesora sterownika PLC w położeniu RUN.

Zapali się kontrolka READY i kontrolka PROGRAM RUN sterownika programowalnego. Zapali się kontrolka RUN modułu ruchu.



Poczekaj, aż zapali się kontrolka PROGRAM RUN modułu ruchu.
Na serwowzmacniaczu wyświetlane jest wskazanie „r.01”.
(Kropki świecą).

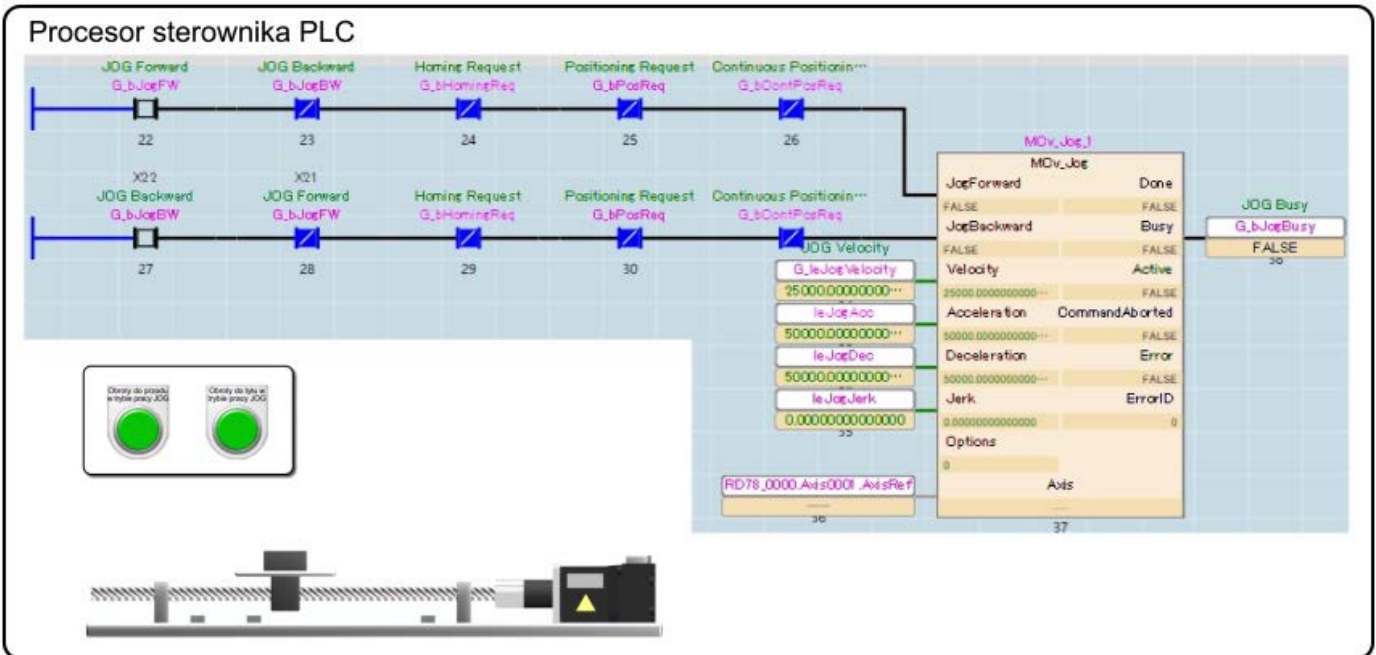


Włącz X20, aby wykonać polecenie wyłączenia serwa.
Na serwowzmacniaczu wyświetlane jest wskazanie „r.01”.
(Kropki migają).



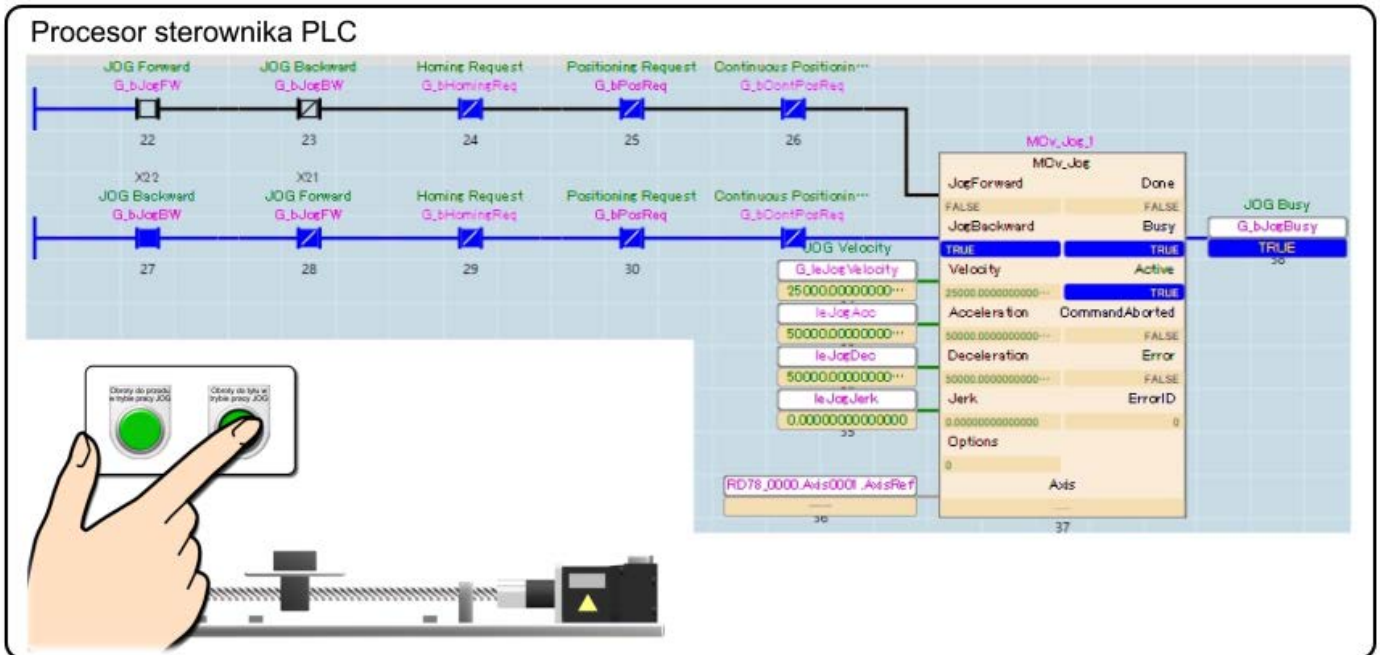
Włącz obroty do przodu w trybie pracy JOG (X21), aby wykonać przesunięcie w kierunku zwiększania adresu i wyłącz obroty, aby zatrzymać pracę.

Włącz obroty osi do tyłu w trybie pracy JOG (X22), aby przesunąć ją w kierunku zmniejszania adresu i wyłącz obroty, aby zatrzymać pracę.



Sprawdź monitor programu.

Po włączeniu X21 następuje włączenie wejścia JogForward w MCv_Jog_1. Wykonywana jest operacja normalnych obrotów w trybie pracy JOG. Wyjście Busy i „G_bJogBusy” włącza się podczas działania.



Po włączeniu X22 następuje włączenie wejścia JogBackward w MCv_Jog_1.

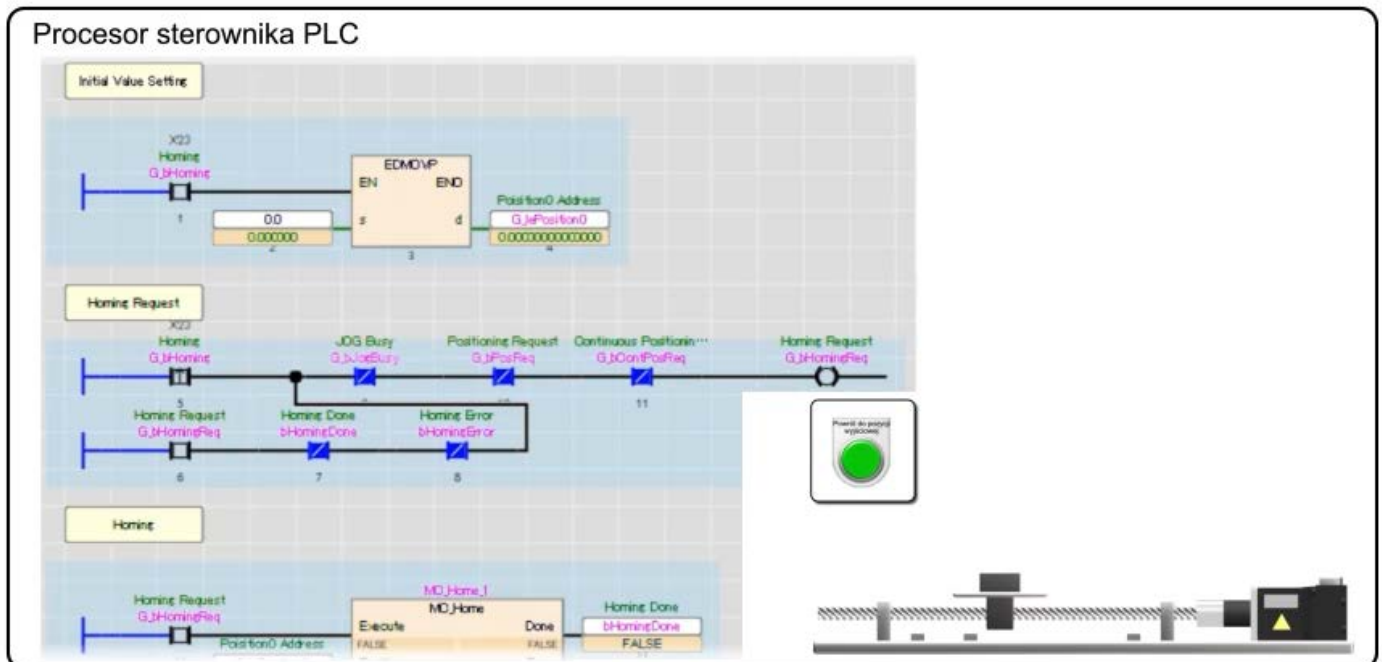
Wykonywana jest operacja obrotów do tyłu w trybie pracy JOG. Wyjście Busy i „G_bJogBusy” włącza się podczas działania.



Włącz powrót do położenia wyjściowego (X23), aby rozpocząć powrót do położenia wyjściowego.

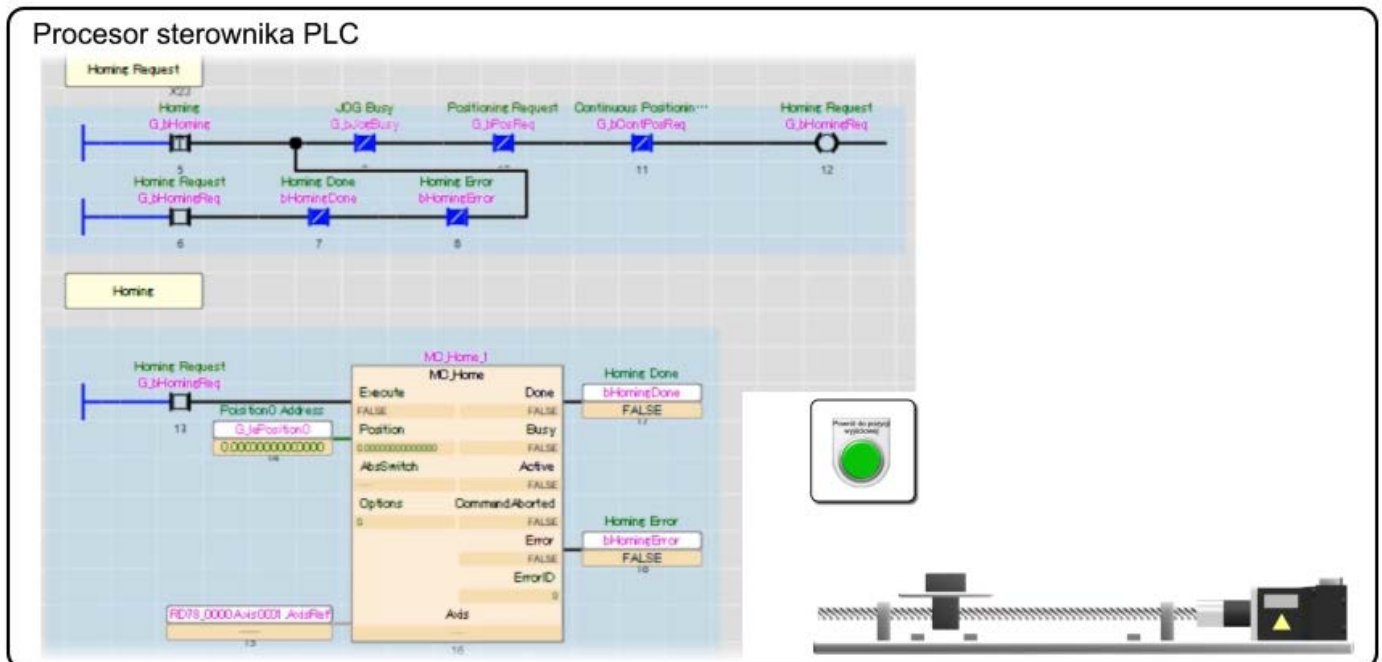
Wykonaj powrót do położenia wyjściowego na czujnik zbliżeniowy DOG
(33 jest odejmowane od Pr.PT45)

Oś zatrzymuje się nieco dalej za czujnikiem DOG i ustawia ten punkt jako położenie wyjściowe.



Sprawdź monitor programu.

Po włączeniu X23 do etykiety wpisywany jest adres położenia wyjściowego. Następuje włączenie i podtrzymanie „G_bHomingReq”, które jest poleceniem wykonania MC_Home_1.

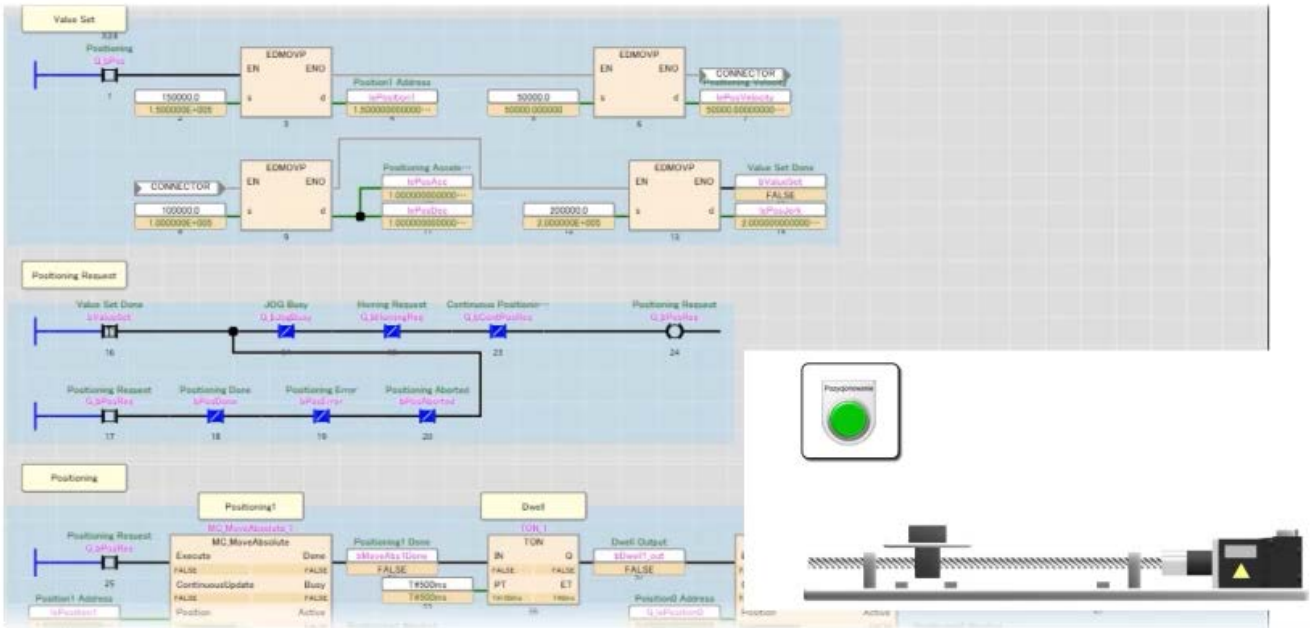


Uruchomiona zostaje operacja powrotu do położenia wyjściowego. Po zakończeniu powrotu do położenia wyjściowego następuje włączenie wyjścia Done i „bHomingDone” oraz zwolnienie podtrzymania „G_bHomingReq”.



Włączenie uruchomienia pozycjonowania (X24)
powoduje uruchomienie ruchu posuwisto-zwrotnego.
Oś przesuwana się do przodu o 150 mm i zatrzymuje się na 0,5 sekundy, a
następnie przesuwana się o 150 mm do tyłu i zatrzymuje się na 0,5 sekundy.

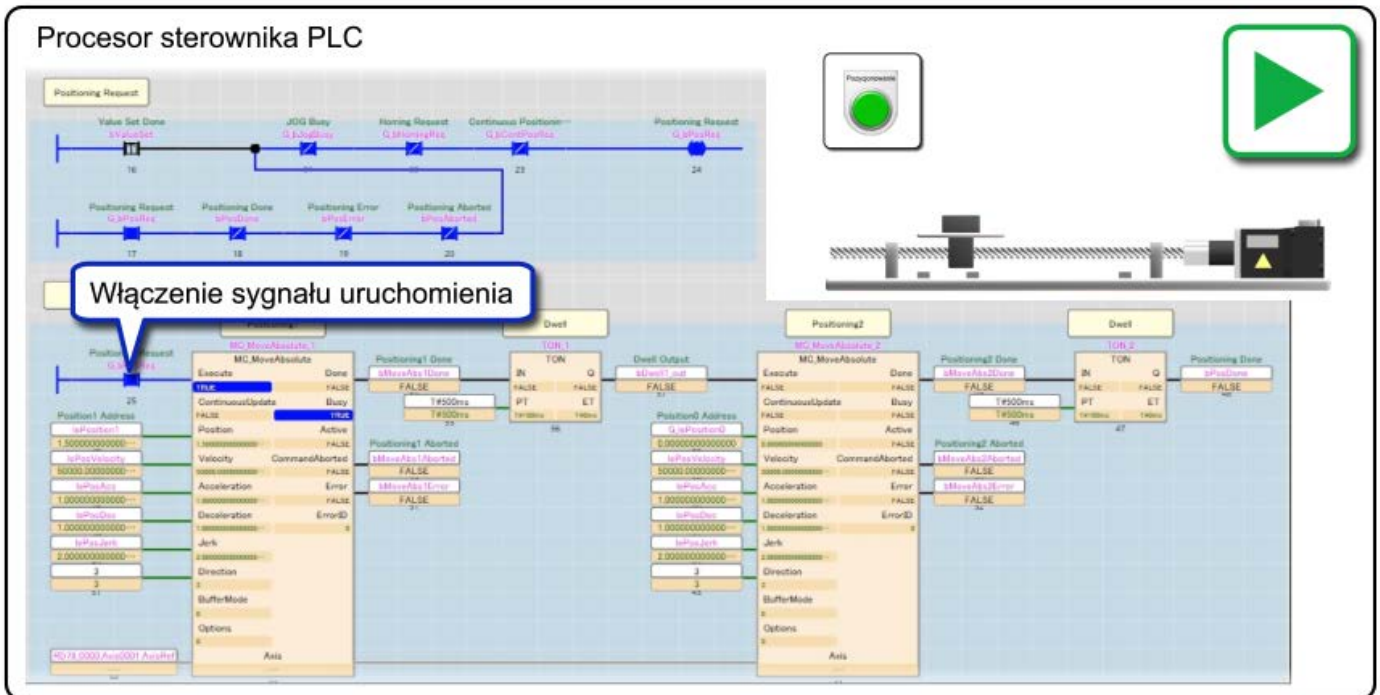
Processor sterownika PLC



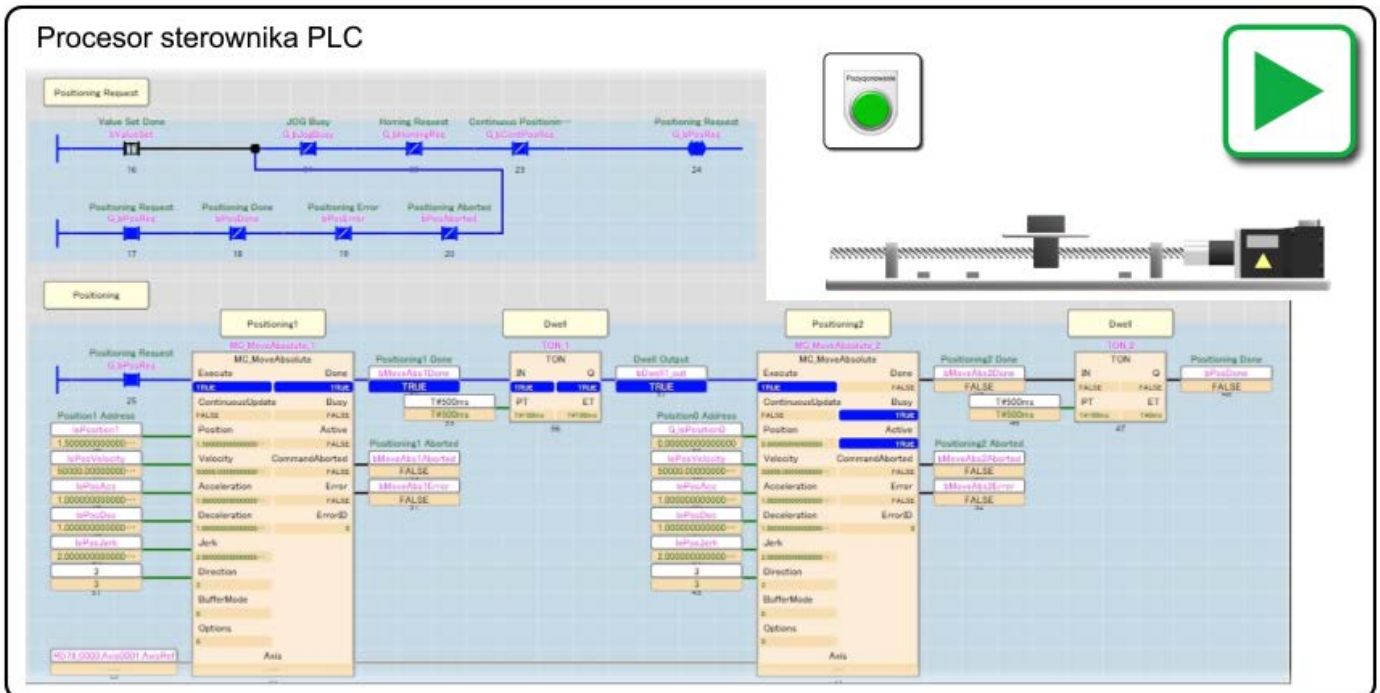
Sprawdź monitor programu.

Po włączeniu X24 następuje wpisanie danych pozycjonowania do każdej etykiety i włączenie „bValueSet”.

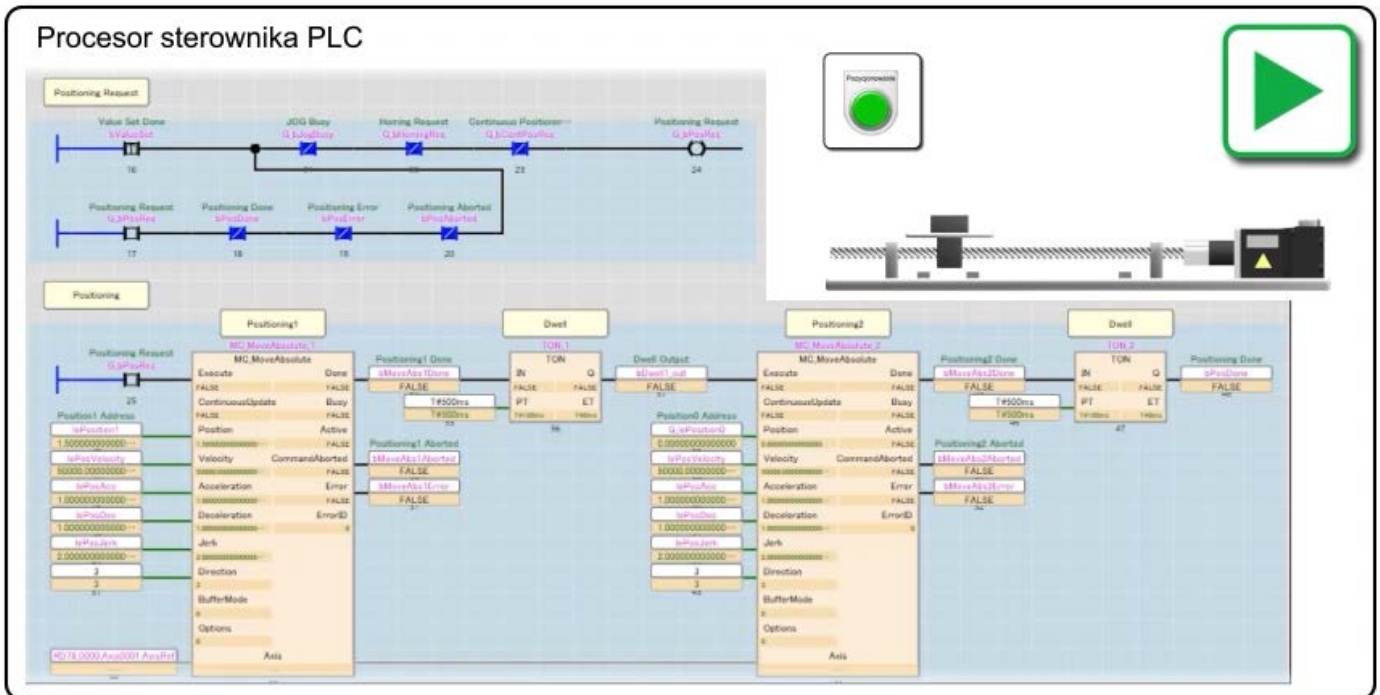
Następuje włączenie i podtrzymanie „G_bPosReq”, które jest poleceniem wykonania `MC_MoveAbsolute_1`, na narastającym zboczcu „bValueSet”.



Po włączeniu „G_bPosReq” następuje uruchomienie MC_MoveAbsolute_1 i rozpoczęcie pracy serwomotoru.



Po zakończeniu pozycjonowania za pomocą MC_MoveAbsolute_1 następuje uruchomienie TON_1 (oczekiwanie).
Po upływie 500 ms wykonywany jest MC_MoveAbsolute_2 i uruchamiany jest serwomotor.

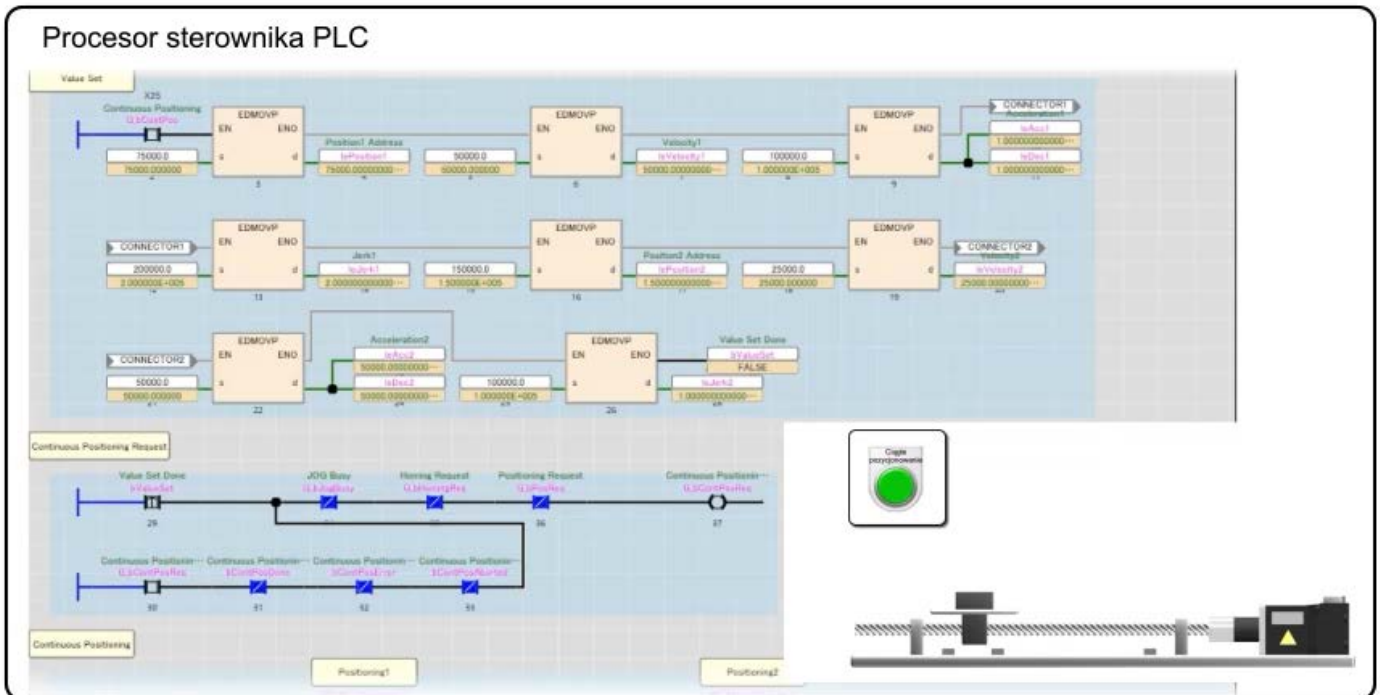


Po zakończeniu pozycjonowania za pomocą MC_MoveAbsolute_2 następuje uruchomienie TON_2 (oczekiwanie).

Po upływie 500 ms następuje zwolnienie podtrzymania „G_bPosReq” i zresetowanie go do stanu początkowego.



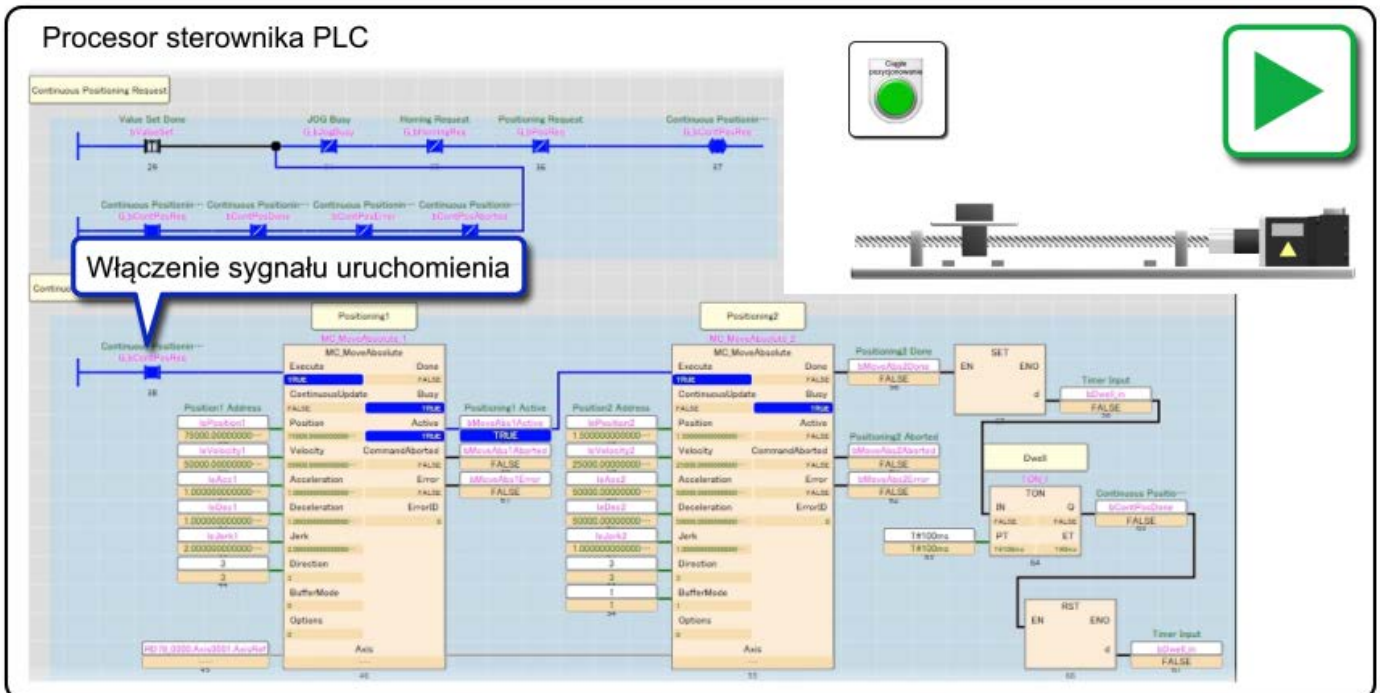
Włącz pozycjonowanie w sposób ciągły (X25), aby rozpocząć działanie trybu buforowania (mc_Buffered).



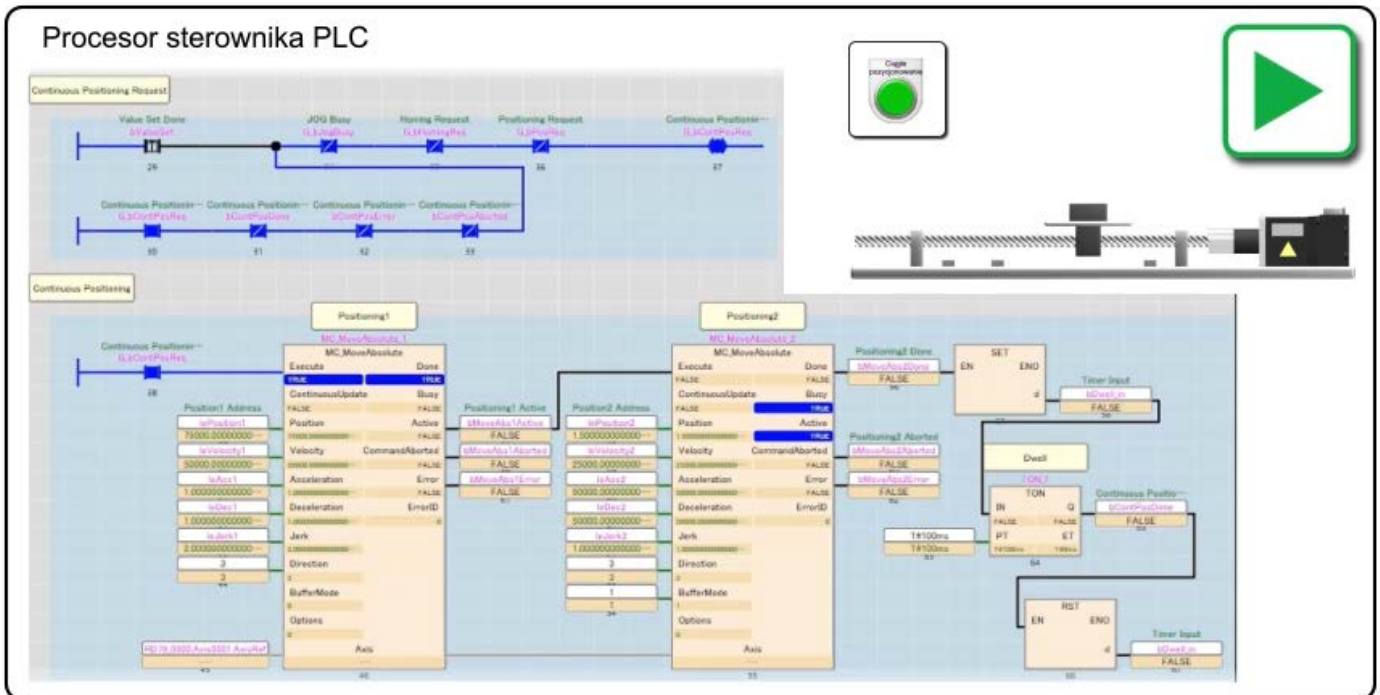
Sprawdź monitor programu.

Po włączeniu X25 następuje wpisanie danych pozycjonowania do każdej etykiety i włączenie „bValueSet”.

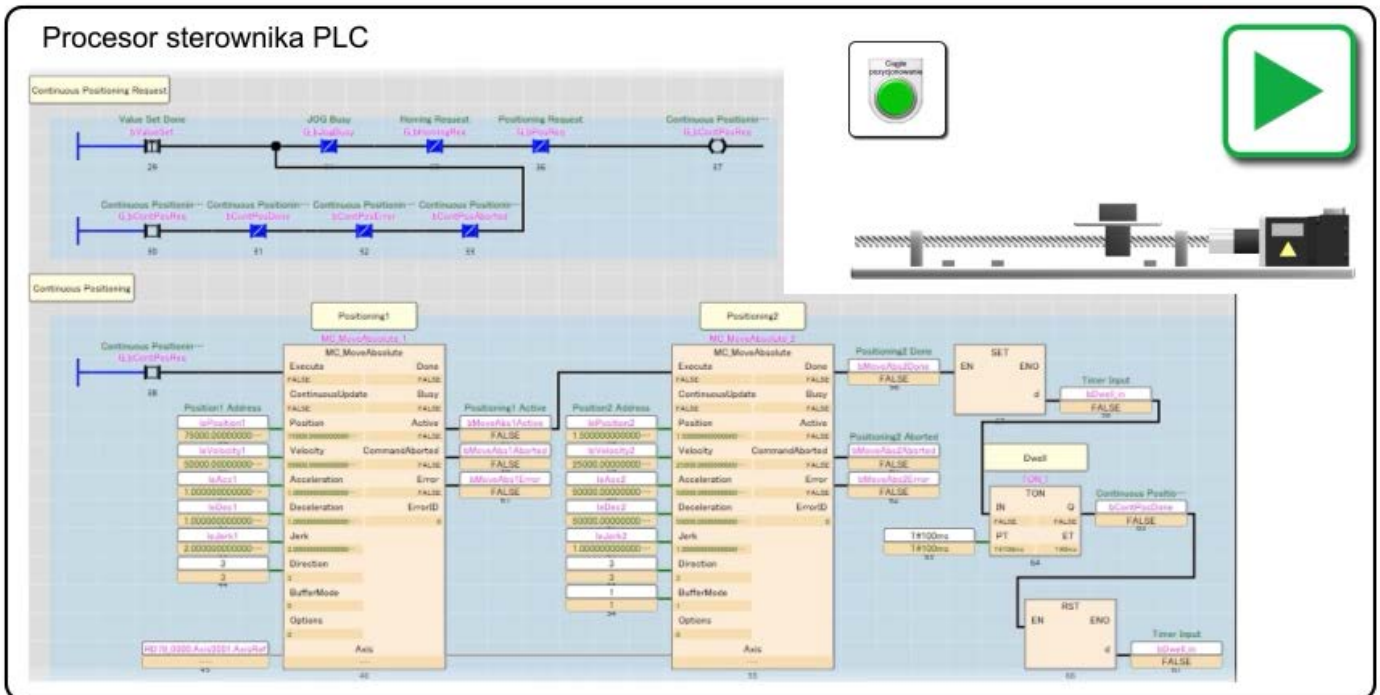
Następuje włączenie i podtrzymanie „G_bContPosReq”, które jest poleceniem wykonania MC_MoAbsolute_1, na narastającym zboczcu „bValueSet”.



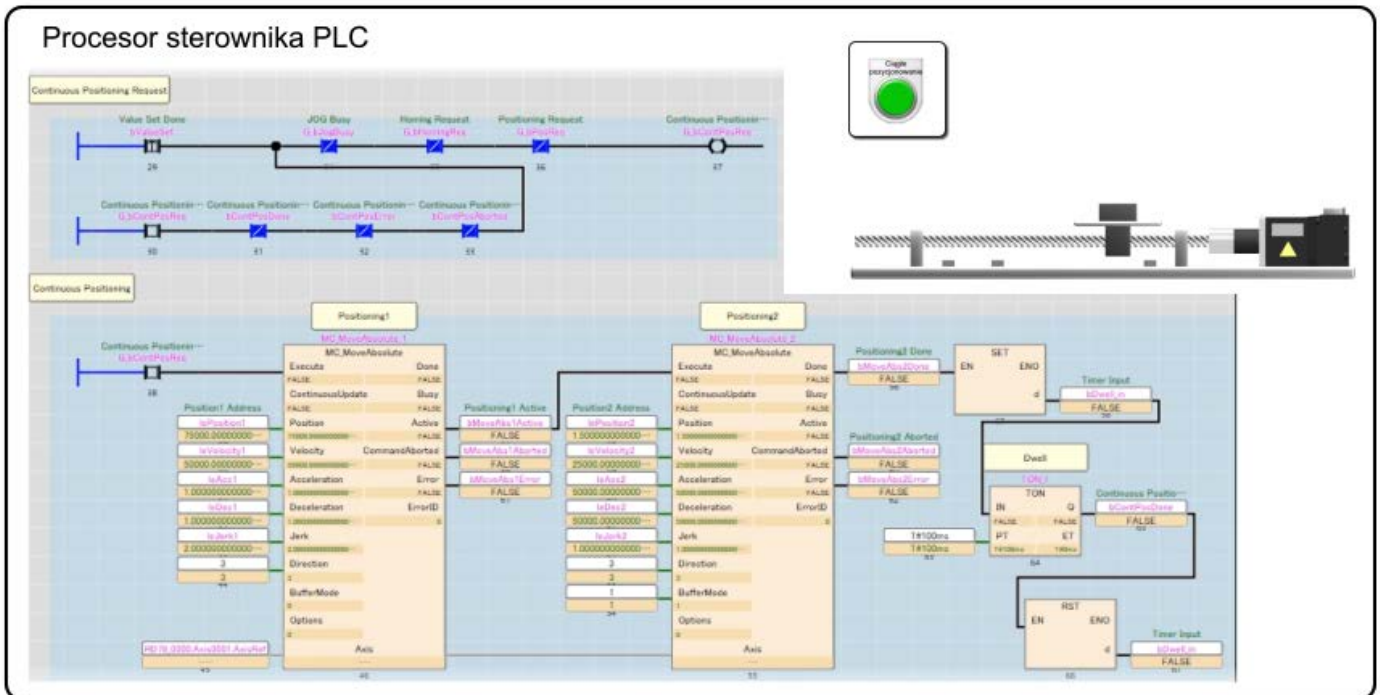
Po włączeniu „G_bContPosReq” następuje uruchomienie MC_MoveAbsolute_1 i rozpoczęcie pracy serwomotoru. Jako że wyjściem Active jest polecenie wykonania MC_MoveAbsolute_2, buforowany jest MC_MoveAbsolute_2.



Gdy operacja wykonywanego MC_MoveAbsolute_1 zostanie zakończona, wykonywany jest buforowany MC_MoveAbsolute_2. Po zakończeniu operacji MC_MoveAbsolute_2 następuje wykonanie TON_1 (oczekiwanie).



Po upływie 100 ms następuje zwolnienie podtrzymania „G_bContPosReq” i zresetowanie go do stanu początkowego.



Na tym kończy się kontrola działania.
Przejdź do następnej strony.

Informacje zdobyte w tym rozdziale:

- Rejestrowanie biblioteki FB modułu ruchu
- Tworzenie projektów
- Używanie Motion Control FB
- Opis przykładowego programu
- Kontrola działania przykładowego programu

Wskazówka

Rejestrowanie biblioteki FB modułu ruchu	<ul style="list-style-type: none"> • Biblioteka FB musi być zarejestrowana w GX Works3, aby móc użyć Motion control FB w procesorze sterownika PLC.
Tworzenie projektów	<ul style="list-style-type: none"> • Skonfiguruj parametry osi i inne ustawienia w taki sposób, jak podczas programowania do modułu ruchu.
Używanie Motion Control FB	<ul style="list-style-type: none"> • FB sterownika ruchu można umieścić w edytorze programu, przeciągając i upuszczając go z karty Library w oknie Element Selection narzędzia GX Works3. • Podłącz styk i etykietę do wejścia/wyjścia FB.
Opis przykładowego programu	<ul style="list-style-type: none"> • Utworzono program podobny do przykładowych programów z rozdziałów 2 i 3 przy użyciu wyłącznie procesora sterownika PLC.
Kontrola działania przykładowego programu	<ul style="list-style-type: none"> • Działanie przykładowego programu zostało sprawdzone na filmie.

Rozdział 5 Rejestrowanie

W tym rozdziale opisano sposób rejestrowania danych modułu ruchu i wyświetlania ich na wykresie.

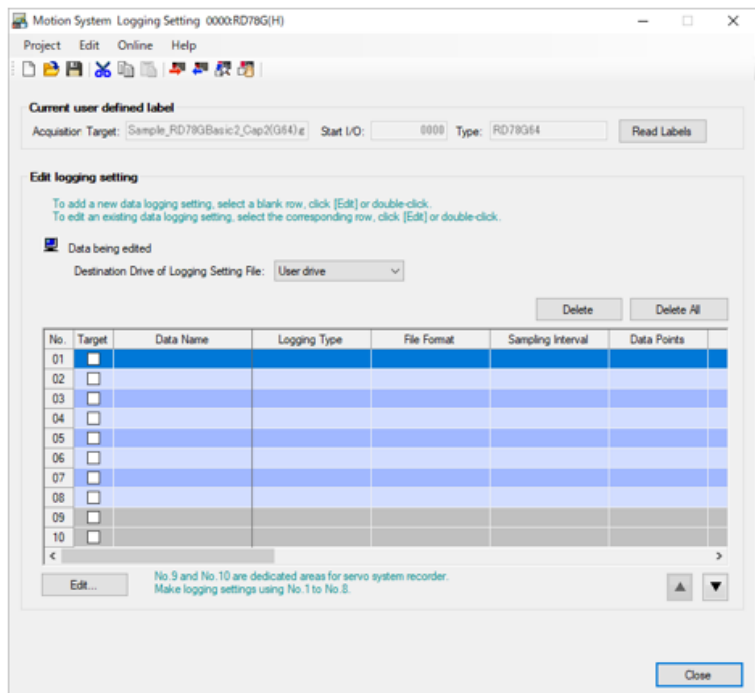
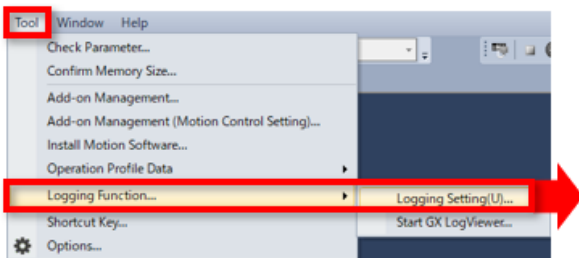
W tym szkoleniu jako przykład zostanie wykorzystany program uruchamiania pozycjonowania z przykładowego programu z rozdziału 2 i 3.

(Uwaga) Programu z rozdziału 4 nie można zarejestrować za pomocą procedury opisanej w tym rozdziale.

Należy użyć narzędzia „CPU Module Logging Configuration Tool”.

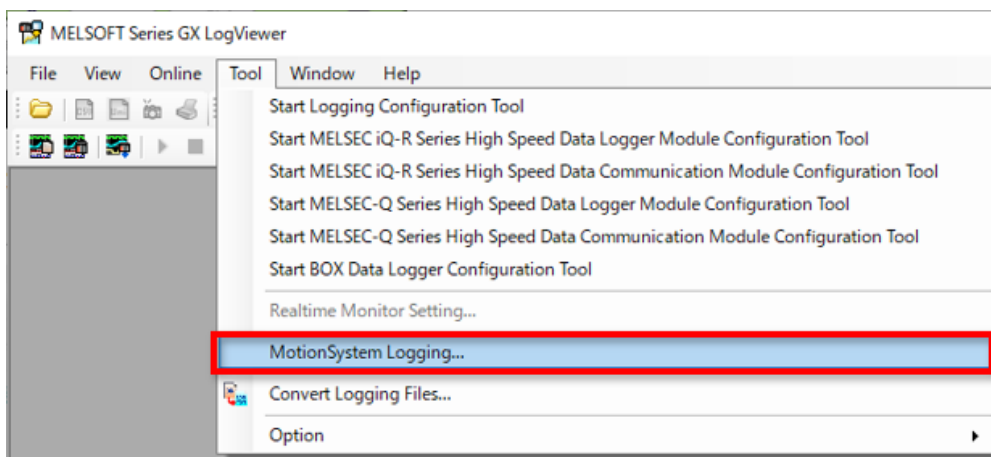
5.1 Uruchamianie narzędzia konfiguracji rejestrowania

Wybierz kolejno [Tool] → [Logging Function] → [Logging Setting] na pasku narzędzi ekranu Motion Control Setting Function. Zostanie uruchomione narzędzie konfiguracji rejestrowania systemu ruchu.



[Wskazówka]

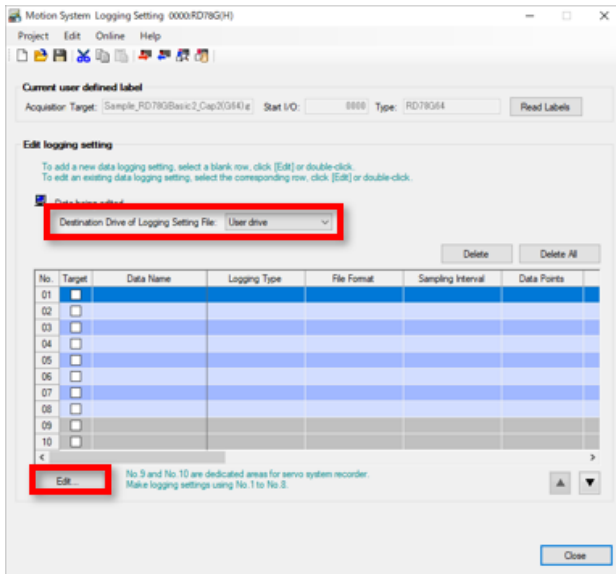
Narzędzie konfiguracji rejestrowania systemu ruchu można uruchomić, wybierając kolejno [Tool] → [MotionSystem Logging] w narzędziu GX LogViewer.



(1) W polu Edit Logging Setting narzędzia konfiguracji rejestrowania systemu ruchu ustaw miejsce docelowe zapisu danych rejestrowania.

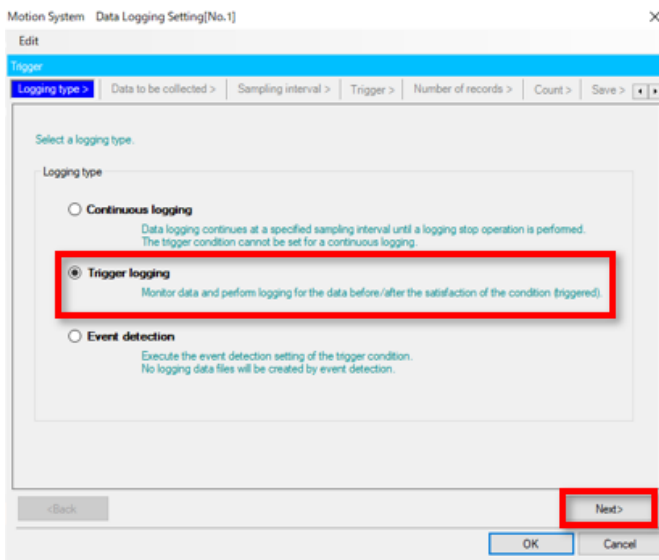
Następnie kliknij przycisk [Edit].

Zostanie wyświetlony ekran Data Logging Setting.



(2) Wybierz typ rejestrowania [Logging type] spośród opcji Continuous Logging, Trigger Logging i Event Detection. W tym szkoleniu opisano opcję Trigger Logging.

Wybierz opcję Trigger Logging, a następnie kliknij przycisk [Next].



(3) Etykieta danych do rejestrowania jest widoczna w polu [Data to be collected].

1) Etykieta globalna

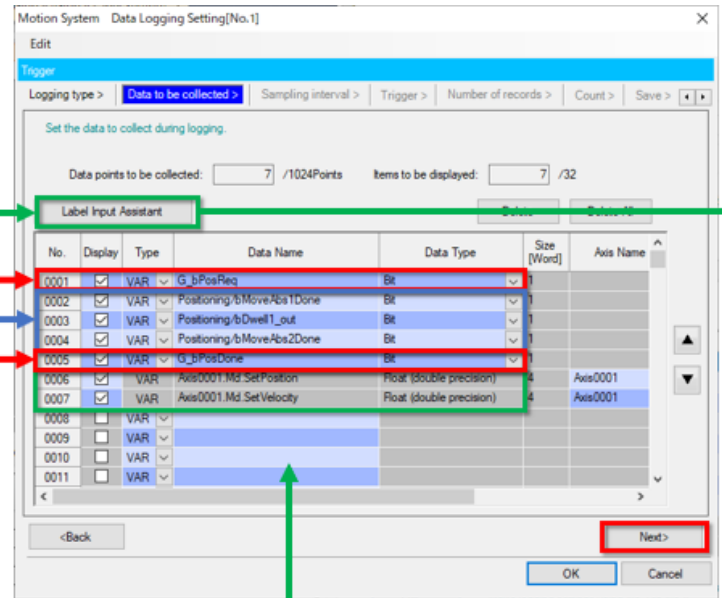
Wprowadź nazwę etykiety globalnej w polu nazwy danych.
W polu typu danych wybierz typ danych etykiety.

2) Etykieta lokalna

Wprowadź nazwę danych w formacie „nazwa programu/nazwa etykiety lokalnej”.
W polu typu danych wybierz typ danych etykiety.

3) Struktury danych

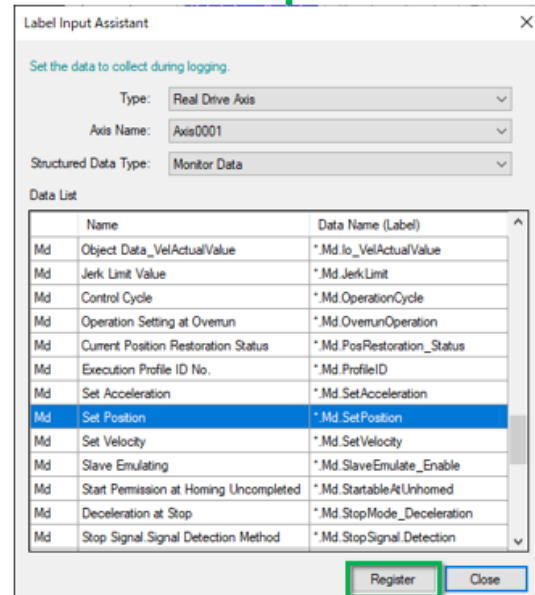
Kliknij przycisk [Label Input Assistant] i wybierz z listy dane typu struktura.
Wybierz je z listy i kliknij przycisk [Register], aby odzwierciedlić które dane będą gromadzone.



W tym szkoleniu dla przykładu rejestrowane są następujące dane.

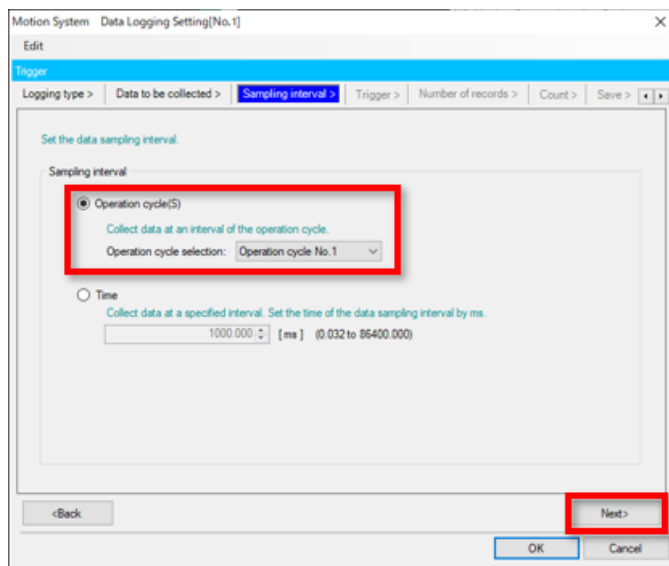
Nazwa danych
G_bPosReq
Positioning/bMoveAbs1Done
Positioning/bDwell1_out
Positioning/bMoveAbs2Done
G_bPosDone
Axis0001.Md.SetPosition
Axis0001.Md.SetVelocity

Kliknij przycisk [Next] po zakończeniu rejestracji.



- (4) Ustaw częstotliwość próbkowania w polu [Sampling Interval].
W tym szkoleniu do próbkowania użyj cyklu operacji nr 1.

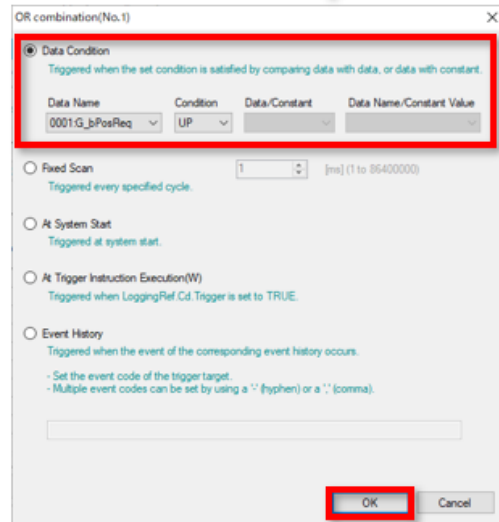
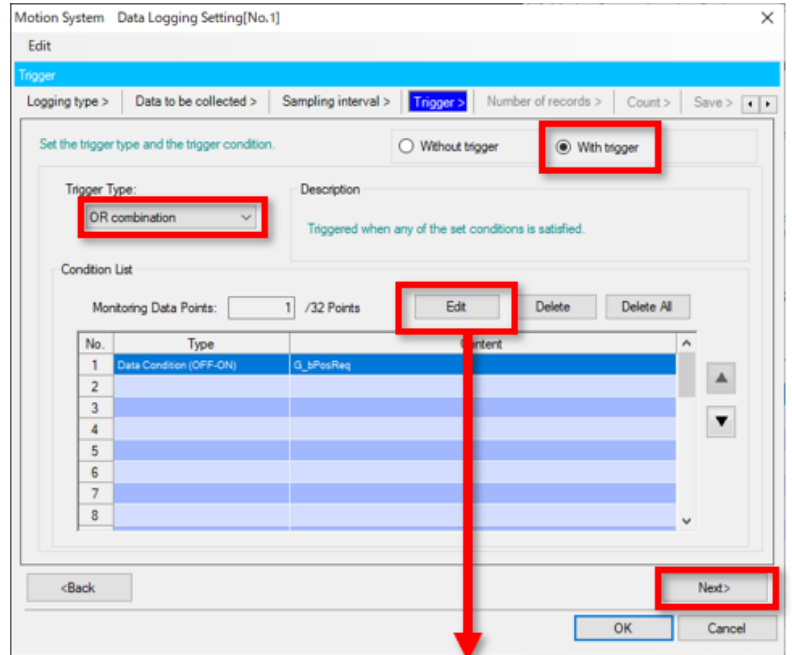
Po wybraniu częstotliwości próbkowania kliknij przycisk [Next].



(5) Warunek rozpoczęcia rejestrowania jest ustawiony w polu [Trigger].

W tym szkoleniu jako element wyzwalający używany jest bit początkowy, który jest sygnałem uruchomienia pozycjonowania.

- 1) Wybierz opcję [With trigger].
- 2) Wybierz opcję „OR combination” dla typu elementu wyzwalającego.
- 3) Wybierz nr 1 z listy warunków i kliknij przycisk Edit. Zostanie wyświetlone nowe okno.
- 4) Wybierz opcję „Data Condition”, a następnie wybierz opcję „0001:G_bPosReq” w polu Data Name. Wybierz opcję „UP” w polu Condition. Po zakończeniu wyboru kliknij przycisk [OK].
- 5) Po powrocie do pierwotnego ekranu kliknij przycisk [Next].



(6) Liczba punktów próbkowania jest ustawiana w polu [Number of records].

W tym szkoleniu opcja No. of records (before trigger) jest ustawiona na „500”, a opcja No. of records (after trigger) jest ustawiona na „19500”.

Po zakończeniu ustawiania kliknij przycisk [Next].

The screenshot shows the 'Data Logging Setting' dialog box with the 'Number of records' tab selected. The dialog has a breadcrumb trail: Logging type > Data to be collected > Sampling interval > Trigger > Number of records > Count > Save >. The main area contains the instruction 'Set the No. of records output to file when triggered during trigger logging.' Below this are three input fields: 'No. of records (before trigger):' with a value of 500, 'No. of records (after trigger):' with a value of 19500, and 'Total No. of records:' with a value of 20000. Each field has a dropdown arrow and a range in parentheses. At the bottom right, the 'Next >' button is highlighted with a red box. Other buttons include '<Back', 'OK', and 'Cancel'.

(7) Liczba rejestracji jest ustawiona w polu [Count]. W tym szkoleniu liczba jest ustawiona na 1.

Po zakończeniu ustawiania kliknij przycisk [Next].

The screenshot shows the 'Data Logging Setting' dialog box with the 'Count' tab selected. The breadcrumb trail is: Logging type > Data to be collected > Sampling interval > Trigger > Number of records > Count > Save >. The main area contains the instruction 'Specify the trigger counts.' Below this is a 'Count' section with two radio button options. The first option, 'Specified Count', is selected and has a value of 1 in its dropdown menu. The second option is 'Specified Number of Saved Files'. At the bottom right, the 'Next >' button is highlighted with a red box. Other buttons include '<Back', 'OK', and 'Cancel'.

- (8) Format i liczbę zapisanych plików z rejestrowanymi danymi ustawia się w polu [Save].
W tym szkoleniu ustawiona jest wartość domyślna (format: JSON, liczba zapisanych plików: 1).
Po zakończeniu ustawiania kliknij przycisk [Next].

Motion System Data Logging Setting[No.1]

Edit

Trigger

Sampling interval > Trigger > Number of records > Count > **Save >** Start condition > Finish

Save setting for logging data file

Set the destination folder of the logging data file and the file format.

File format: **JSON** Save in the same folder as the setting file

Destination drive:

Folder:

Maximum number of saved files

Set the maximum number of saved files of the destination drive.
Files are created until the capacity of the destination becomes full when 'Do not specify the maximum value' has been enabled.

1 (1 to 65535) Do not specify the maximum value

Operation when the maximum number is exceeded

Overwrite Files with lower numbers are deleted

Stop Logging stops.

<Back **Next >**

OK Cancel

- (9) Warunek rozpoczęcia rejestrowania jest ustawiony w polu [Start condition].
W tym szkoleniu ustawiona jest opcja „Start by User Operation”.
Po zakończeniu ustawiania kliknij przycisk [Next].

Motion System Data Logging Setting[No.1]

Edit

Trigger

Sampling interval > Trigger > Number of records > Count > Save > **Start condition >** Finish

Select the condition to start the data logging.

Logging Start Condition

Auto Start
The data logging starts automatically at the start of the motion system.
It is possible to stop the logging manually.

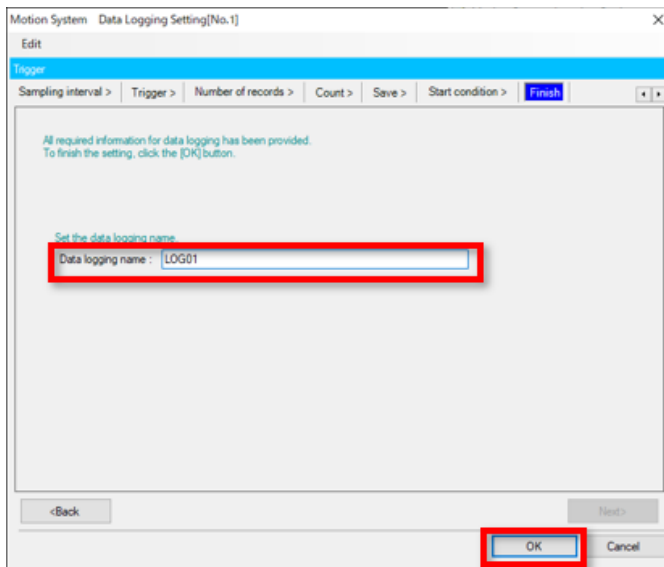
Unstoppable After Auto Start
The data logging starts automatically at the start of the motion system.
Unable to stop the logging manually.

Start by User Operation
Execute the data logging start operation in the Logging Status and Operation window, etc.
It is also possible to stop the logging manually.

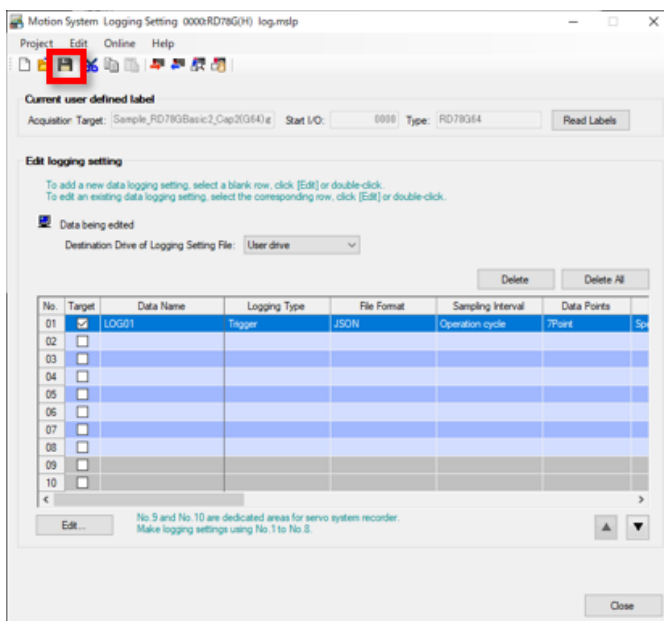
<Back **Next >**

OK Cancel

- (10) Nazwa rejestrowania danych jest ustawiona w polu [Finish].
W tym szkoleniu ustawiona jest wartość domyślna (LOG01).
Po zakończeniu ustawiania kliknij przycisk [Next].



- (11) Powrót do narzędzia konfiguracji rejestrowania systemu ruchu.
Skonfigurowane ustawienia można zapisać.
Kliknij ikonę zapisu i zapisz ustawienia w wybranym miejscu docelowym.

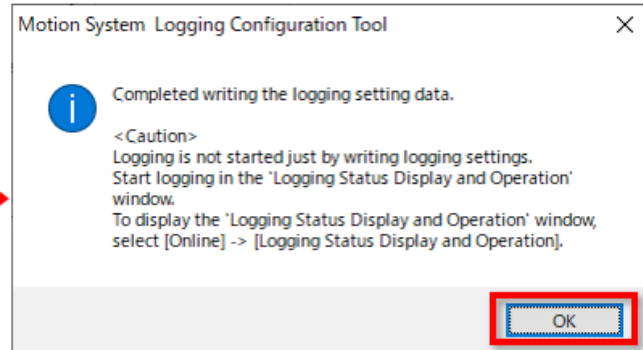
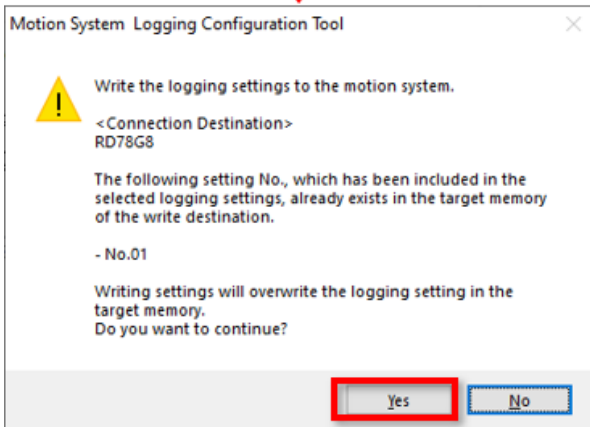
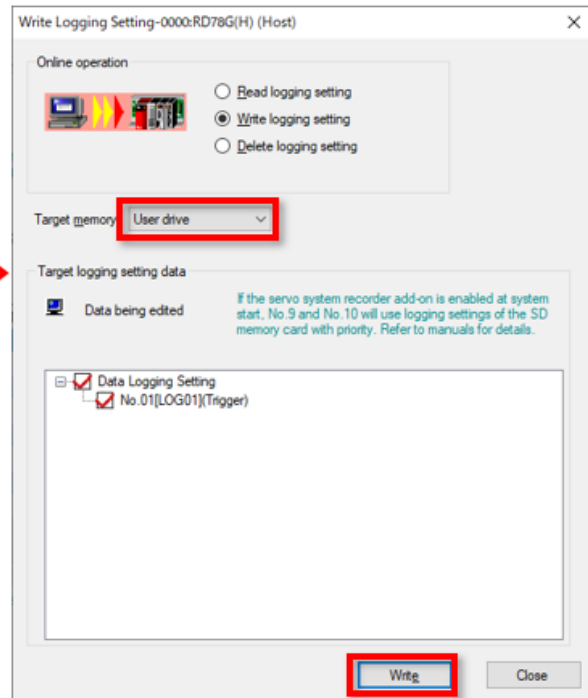
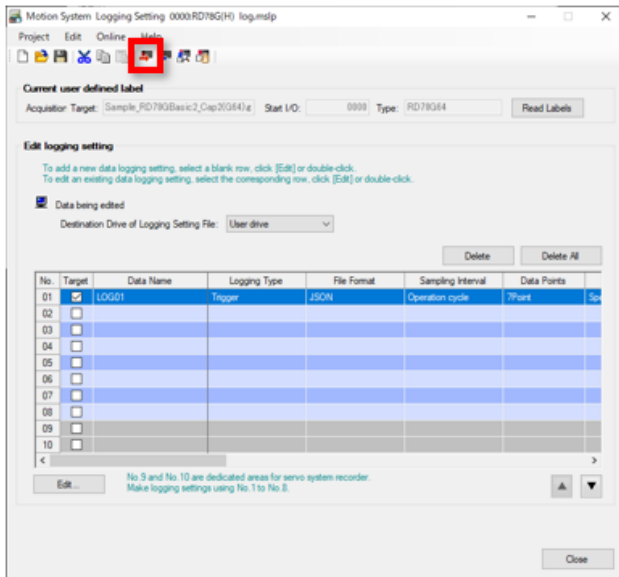


Informacje o ustawieniach rejestrowania można zapisać.

Kliknij ikonę zapisu ustawień rejestrowania, wybierz pamięć docelową i kliknij przycisk [Write].

Zostanie wyświetlone okno potwierdzenia.

Kliknij przycisk [Yes], aby kontynuować. Po zakończeniu zapisu kliknij przycisk [OK] i zamknij ekran.



Jeśli w punkcie 5.2 (9) ustawiono opcję „Start by User Operation”, kliknij ikonę [Logging Status and Operation], aby wyświetlić ekran [Logging Status and Operation] i uruchomić rejestrowanie.

Po wybraniu nazwy ustawień danych rejestrowania do wykonania kliknij przycisk [Start], opcja LoggingStatus zmieni się na „Waiting for trigger”.

Gdy program jest wykonywany w tym stanie, a warunek elementu wyzwalającego (gdy X24 jest włączony w tym przykładzie szkolenia) jest spełniony, stan zmienia się na „Triggered”.

Po zakończeniu rejestrowania stan zmienia się na „CollectionCompleted” z „Saving”.

The image displays two screenshots of the Motion System software interface and a flowchart illustrating the logging status transitions.

Left Screenshot: Motion System Logging Setting

Current user defined label: Sample_RD790Basic2_Cap3(044) e Start I/O: 0000 Type: RD79064 Read Labels

Edit logging setting

Destination Drive of Logging Setting File: User drive

No.	Target	Data Name	Logging Type	File Format	Sampling Interval	Data Points
01	<input checked="" type="checkbox"/>	LOG01	Trigger	.LOG	Operation cycle	7Phase
02	<input type="checkbox"/>					
03	<input type="checkbox"/>					
04	<input type="checkbox"/>					
05	<input type="checkbox"/>					
06	<input type="checkbox"/>					
07	<input type="checkbox"/>					
08	<input type="checkbox"/>					
09	<input type="checkbox"/>					
10	<input type="checkbox"/>					

No. 9 and No. 10 are dedicated areas for servo system recorder. Make logging settings using No. 1 to No. 8.

Right Screenshot: Motion System Logging Status and Operation

Monitor status: Monitoring Stop

User Drive Free Volume: 47852 MB RAM Drive Free Volume: 47852 MB SD Memory Card Free Volume: 47852 MB

Logging status and operation

Motion System Data

No.	Target	Data Name	Logging Type	Sampling Interval(ms)	Stopped
01	User drive	LOG01	Trigger		<input checked="" type="checkbox"/>
02	User drive				<input type="checkbox"/>
03	User drive				<input type="checkbox"/>
04	User drive				<input type="checkbox"/>
05	User drive				<input type="checkbox"/>
06	User drive				<input type="checkbox"/>
07	User drive				<input type="checkbox"/>
08	User drive				<input type="checkbox"/>
09	User drive				<input type="checkbox"/>
10	User drive				<input type="checkbox"/>

Logging operation

Start Stop

Flowchart: Logging Status Transitions

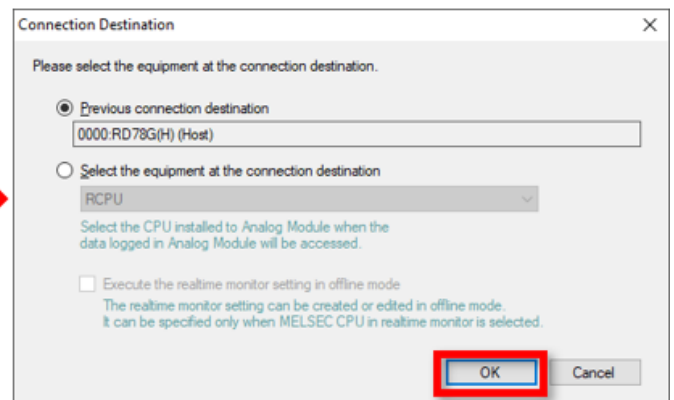
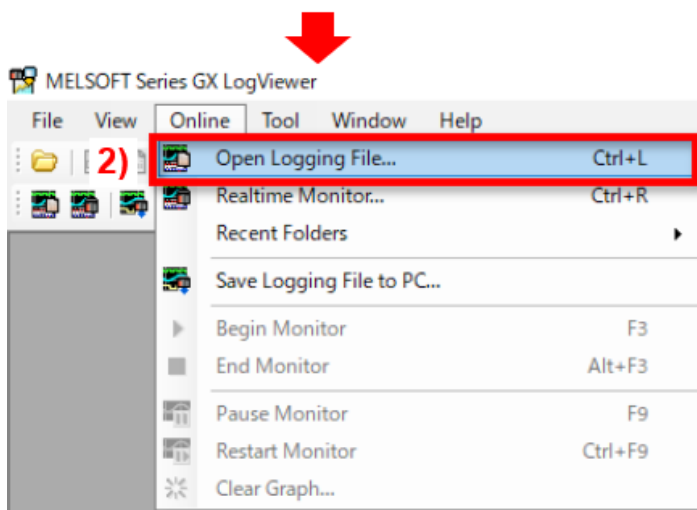
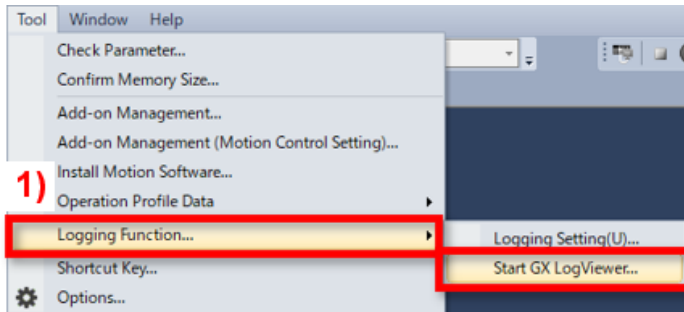
- Logging Status: Waiting for trigger
- Logging Status: Triggered
- Logging Status: Saving
- Logging Status: CollectionCompleted

Do odczytu zarejestrowanych danych służy narzędzie GX LogViewer.

Wybierz kolejno [Tool] → [Logging Function] → [Start GX LogViewer] na pasku narzędzi ekranu Motion Control Setting Function.

Po uruchomieniu narzędzia GX LogViewer wybierz kolejno [Online] → [Open Logging File].

Na ekranie Connection Destination wybierz opcję „0000:RD78G(H) (Host)”. (Uwaga)

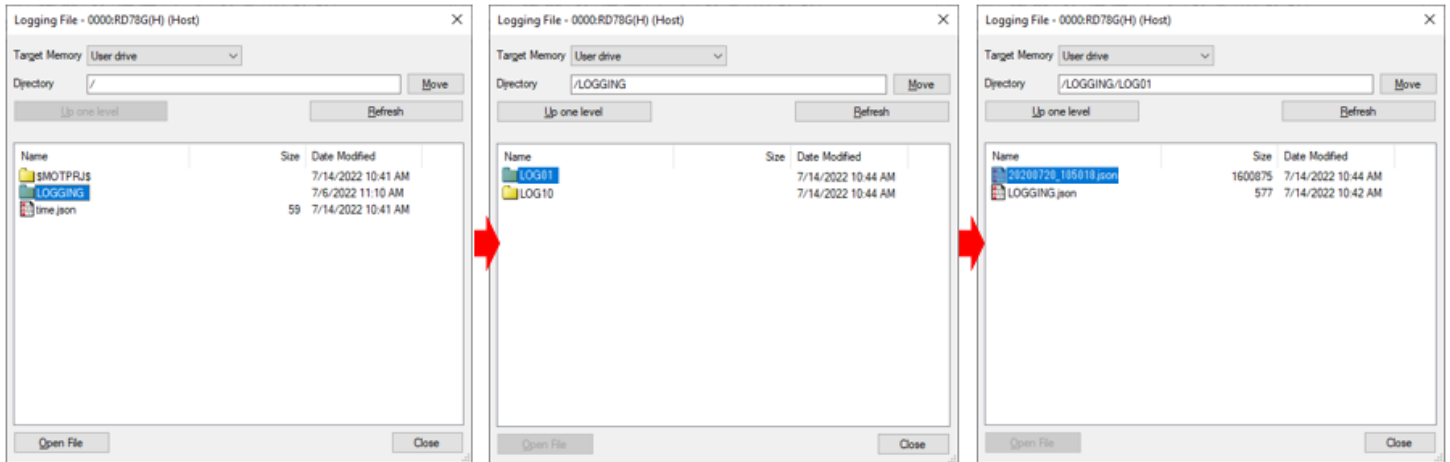


(Uwaga) Jeśli narzędzie GX LogViewer jest już uruchomione, a komunikacja z modułem ruchu jest już ustanowiona, ekran ten nie jest wyświetlany.

Wybierz plik z zarejestrowanymi danymi do odczytania.

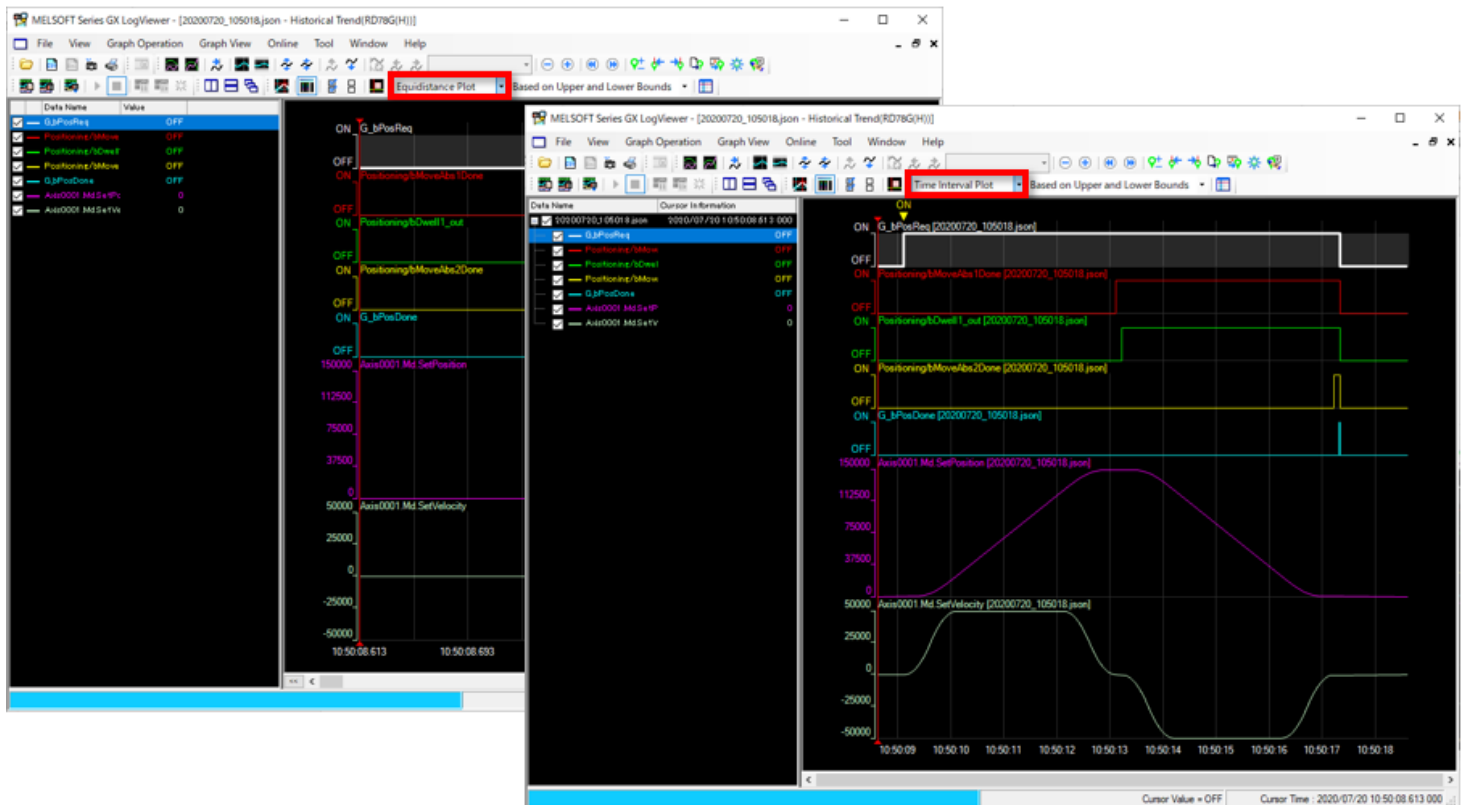
W przykładzie w tym rozdziale wybrany jest dysk użytkownika w obszarze „LOGGING” → „LOG1” → „(Logged date and time).json”.

Wybierz nazwę pliku i kliknij przycisk [Open File].



W narzędziu GX LogViewer wyświetlone zostaną zarejestrowane dane.

Po zmianie formatu wykresu z „Equidistance Plot” na „Time Interval Plot” można wyświetlić cały zarejestrowany przebieg.



Zarejestrowane dane przebiegu można zapisać jako plik csv lub json.
(Jeśli dane są zarejestrowane w formacie CSV, można je zapisać jako plik CSV).

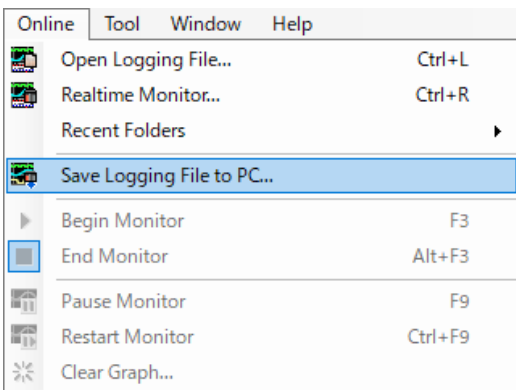
1) Zapis do pliku csv

Wybierz kolejno [File] → [Save As] → [Save CSV File] na pasku narzędzi GX LogViewer.



2) Zapis do pliku json

Wybierz kolejno [Online] → [Save Logging File to PC] na pasku narzędzi GX LogViewer.



Informacje zdobyte w tym rozdziale:

- Uruchamianie narzędzia konfiguracji rejestrowania
- Ustawianie danych do rejestrowania
- Zapisywanie ustawień rejestrowania
- Uruchamianie rejestrowania
- Odczyt zarejestrowanych danych
- Zapis zarejestrowanych danych

Wskazówka

Uruchamianie narzędzia konfiguracji rejestrowania	<ul style="list-style-type: none">• Uruchom narzędzie konfiguracji rejestrowania systemu ruchu za pomocą Motion Control Setting Function.
Ustawianie danych do rejestrowania	<ul style="list-style-type: none">• Ustaw dane do rejestrowania, warunki wyzwania i inne, postępując zgodnie z procedurą wyświetlaną w narzędziu konfiguracji rejestrowania systemu ruchu.
Zapisywanie ustawień rejestrowania	<ul style="list-style-type: none">• Przed rozpoczęciem rejestrowania zapisz dane ustawień rejestrowania do modułu ruchu.
Uruchamianie rejestrowania	<ul style="list-style-type: none">• Gdy warunek uruchomienia rejestrowania jest ustawiony na „Start by User Operation”, kliknij przycisk Start na ekranie „Logging Status and Operation”, aby uruchomić rejestrowanie.
Odczyt zarejestrowanych danych	<ul style="list-style-type: none">• Do odczytu zarejestrowanych danych służy narzędzie GX LogViewer.er.
Zapis zarejestrowanych danych	<ul style="list-style-type: none">• Zarejestrowane dane przebiegu można zapisać jako plik csv lub json.

Wybierz prawidłowe opisy etykiety publicznej. (Można wybrać kilka odpowiedzi).

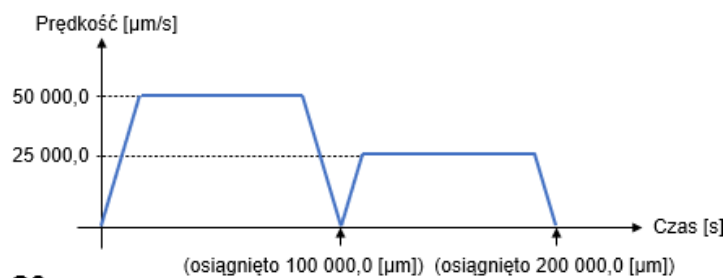
- Etykieta publiczna to współdzielona etykieta, której można używać zarówno w module ruchu, jak i w procesorze sterownika PLC.
- Etykieta publiczna jest rejestrowana z etykiety globalnej procesora sterownika PLC.
- W przypadku ustawienia etykiety globalnej na etykietę publiczną należy wybrać, czy etykieta ma zostać odczytana, czy zapisana z/do procesora sterownika PLC.

Najpierw wykonywany jest FB1, a następnie FB2.

Gdy pozycja docelowa i prędkość docelowa FB1 i FB2 są zgodne z poniższą tabelą, należy wybrać tryb buforowania, który zostanie wykonany w następnej kolejności.

	Pozycja docelowa	Prędkość docelowa
FB1	100 000,0 [μm]	50 000,0 [μm/s]
FB2	200 000,0 [μm]	25 000,0 [μm/s]

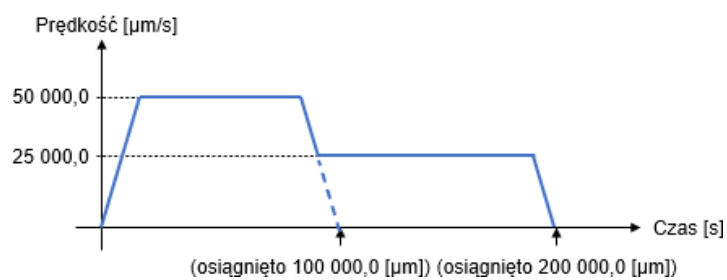
Q1



Q1

Q2

Q2



Q1: 1: mcAborting
2: mcBuffered
3: mcBlendingNext
4: mcBlendingPrevious

Q2: 1: mcBlendingNext i mcBlendingHigh
2: mcBlendingPrevious i mcBlendingHigh
3: mcBlendingNext i mcBlendingLow
4: mcBlendingPrevious i mcBlendingLow

Wybierz poprawne zdanie(-a) z poniższych dla programowania z wykorzystaniem procesora sterownika PLC. (Można wybrać kilka odpowiedzi).

- Biblioteka FB musi być zarejestrowana w GX Works3, aby móc użyć Motion control FB dla modułu ruchu w procesorze sterownika PLC.
- Umieść motion control FB w edytorze programu z drzewa projektu w narzędziu GX Works3.
- Nie ma żadnych parametrów do ustawienia dla modułu ruchu.

Wybierz prawidłowe odpowiedzi w celu uzupełnienia pustych pól.

- Uruchom (Q1), aby ustawić dane do rejestrowania.
- Zapisz zarejestrowane dane do (Q2), aby wykonać rejestrowanie.
- (Q3) służy do odczytu zarejestrowanych danych i sprawdzania ich przebiegu.

Q1

Wybierz prawidłową odpowiedź.



Q2

Wybierz prawidłową odpowiedź.



Q3

Wybierz prawidłową odpowiedź.



Q1: 1: CPU module logging configuration tool
2: Narzędzie konfiguracji rejestrowania systemu ruchu

Q2: 1: Procesor
2: Moduł ruchu
3: Serwowzmacniacz

Q3: 1: MR Configurator2
2: GX LogViewer

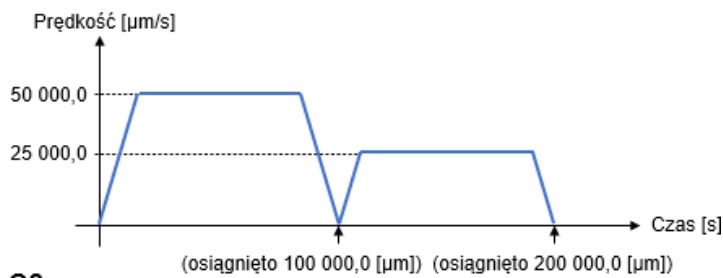
Wybierz prawidłowe opisy etykiety publicznej. (Można wybrać kilka odpowiedzi).

- Etykieta publiczna to współdzielona etykieta, której można używać zarówno w module ruchu, jak i w procesorze sterownika PLC.
- Etykieta publiczna jest rejestrowana z etykiety globalnej procesora sterownika PLC.
- W przypadku ustawienia etykiety globalnej na etykietę publiczną należy wybrać, czy etykieta ma zostać odczytana, czy zapisana z/do procesora sterownika PLC.

Najpierw wykonywany jest FB1, a następnie FB2.
 Gdy pozycja docelowa i prędkość docelowa FB1 i FB2 są zgodne z poniższą tabelą, należy wybrać tryb buforowania, który zostanie wykonany w następnej kolejności.

	Pozycja docelowa	Prędkość docelowa
FB1	100 000,0 [µm]	50 000,0 [µm/s]
FB2	200 000,0 [µm]	25 000,0 [µm/s]

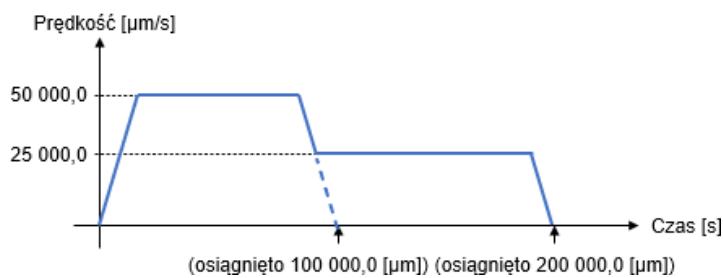
Q1



Q1 2: mcBuffered

Q2 3: mcBlendingNext i mcBlendingLow

Q2



- Q1:** 1: mcAborting
 2: mcBuffered
 3: mcBlendingNext
 4: mcBlendingPrevious

- Q2:** 1: mcBlendingNext i mcBlendingHigh
 2: mcBlendingPrevious i mcBlendingHigh
 3: mcBlendingNext i mcBlendingLow
 4: mcBlendingPrevious i mcBlendingLow

Wybierz poprawne zdanie(-a) z poniższych dla programowania z wykorzystaniem procesora sterownika PLC. (Można wybrać kilka odpowiedzi).



Biblioteka FB musi być zarejestrowana w GX Works3, aby móc użyć Motion control FB dla modułu ruchu w procesorze sterownika PLC.



Umieść motion control FB w edytorze programu z drzewa projektu w narzędziu GX Works3.



Nie ma żadnych parametrów do ustawienia dla modułu ruchu.

Wybierz prawidłowe odpowiedzi w celu uzupełnienia pustych pól.

- Uruchom (Q1), aby ustawić dane do rejestrowania.
- Zapisz zarejestrowane dane do (Q2), aby wykonać rejestrowanie.
- (Q3) służy do odczytu zarejestrowanych danych i sprawdzania ich przebiegu.

Q1

2: Narzędzie konfiguracji rejestrowania systemu ruchu



Q2

2: Moduł ruchu



Q3

2: GX LogViewer



Q1: 1: CPU module logging configuration tool
2: Narzędzie konfiguracji rejestrowania systemu ruchu

Q2: 1: Procesor
2: Moduł ruchu
3: Serwowzmacniacz

Q3: 1: MR Configurator2
2: GX LogViewer

Test końcowy został ukończony. Twoje wyniki są przedstawione poniżej.
Aby zakończyć test końcowy, przejdź do następnej strony.

	1	2	3	4	5	6	7	8	9	10
Test końcowy 1	✓									
Test końcowy 2	✓	✓								
Test końcowy 3	✓									
Test końcowy 4	✓	✓	✓							

Wszystkie pytania: **7**

Prawidłowe odpowiedzi: **7**

Procent prawidłowych
odpowiedzi: **100 %**

Wyczyść

Ukończono szkolenie „Moduł ruchu MELSEC iQ-R – zastosowanie (sterowanie pozycjonowaniem RD78G(H))” .

Dziękujemy za wzięcie udziału w kursie.

Mamy nadzieję, że poruszone tematy były interesujące, a informacje uzyskane w trakcie tego kursu będą przydatne w przyszłości.

Możesz przeglądać kurs dowolną ilość razy.

Sprawdź

Zamknij