



PLC GX Works2 zaawansowany

Niniejsze szkolenie zapewnia wiedzę z zakresu funkcji usprawniających środowisko programistyczne projektanta systemu, mającego problemy z „produktywnością”, „jakością”, „zarządzaniem projektem” i „środkami bezpieczeństwa”. Kurs jest przeznaczony dla programistów, korzystających już z MELSOFT GX Works2.

» **Wprowadzenie** **Przeznaczenie kursu**

Niniejsze szkolenie zapewnia wiedzę z zakresu funkcji usprawniających środowisko programistyczne projektowanego zakładu, mającego problemy z „produktywnością”, „jakością”, „zarządzaniem projektem” i „środkami bezpieczeństwa”. Kurs jest przeznaczony dla programistów, korzystających już z MELSOFT GX Works2.

Wprowadzenie **Struktura kursu**

Treść tego kursu posiada następującą strukturę.
Zalecamy rozpoczęcie od Rozdziału 1.

Rozdział 1 – Usprawnienie środowiska programistycznego przy użyciu GX Works2

Rozdział przedstawia informacje dotyczące problemów, z jakimi boryka się projektowany zakład oraz środowisko programistyczne wymagane w celu ich rozwiązania.

Rozdział 2 – Programowanie

Rozdział przedstawia funkcje stosowane w celu programowania.

Rozdział 3 – Wyszukiwanie i usuwanie usterek

Rozdział przedstawia funkcje stosowane w celu wyszukiwania i usuwania usterek.

Rozdział 4 – Zarządzanie projektem i środki bezpieczeństwa

Rozdział przedstawia funkcje związane z zarządzaniem projektem i środki bezpieczeństwa na etapie rozwoju i konserwacji po uruchomieniu pracy systemu.

Test końcowy

Ocena zaliczająca: 60% lub wyższa

Wprowadzenie Jak korzystać z narzędzia do e-learningu

| | | |
|-----------------------------|--|--|
| 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. |

Wprowadzenie **Ostrzeżenia dotyczące użytkowania**

Zalecenia dotyczące bezpieczeństwa

Jeśli uczysz się, korzystając z rzeczywistych produktów, prosimy o dokładne przeczytanie zasad bezpieczeństwa zawartych w odpowiednich instrukcjach obsługi.

Środki ostrożności dla tego kursu

- Ekran wyświetlany dla wersji oprogramowania, którego używasz, mogą się różnić od przedstawionych w tym kursie.

Rozdział 1 Usprawnienie środowiska programistycznego przy użyciu GX Works2

Kroki procesu uczenia w Rozdziale 1

Kurs jest przeznaczony dla programistów pracujących nad rozwojem programów sekwencyjnych. Przedstawia sposób korzystania z doskonałych funkcji **MELSOFT GX Works2** w celu rozwiązywania problemów projektowanego zakładu związanych z „**produktywnością**”, „**jakością**”, „**zarządzaniem projektem**” i „**bezpieczeństwem**”.

W rozdziale 1 zostawione zostały informacje dotyczące problemów, z jakimi boryka się projektowany zakład oraz środowisko programistyczne wymagane w celu ich rozwiązania.

- 1.1 Problemy, z jakimi boryka się projektowany zakład
 - 1.1.1 Poprawa produktywności
 - 1.1.2 Poprawa jakości
 - 1.1.3 Zarządzanie projektem
 - 1.1.4 Środki bezpieczeństwa
- 1.2 Procedura uczenia



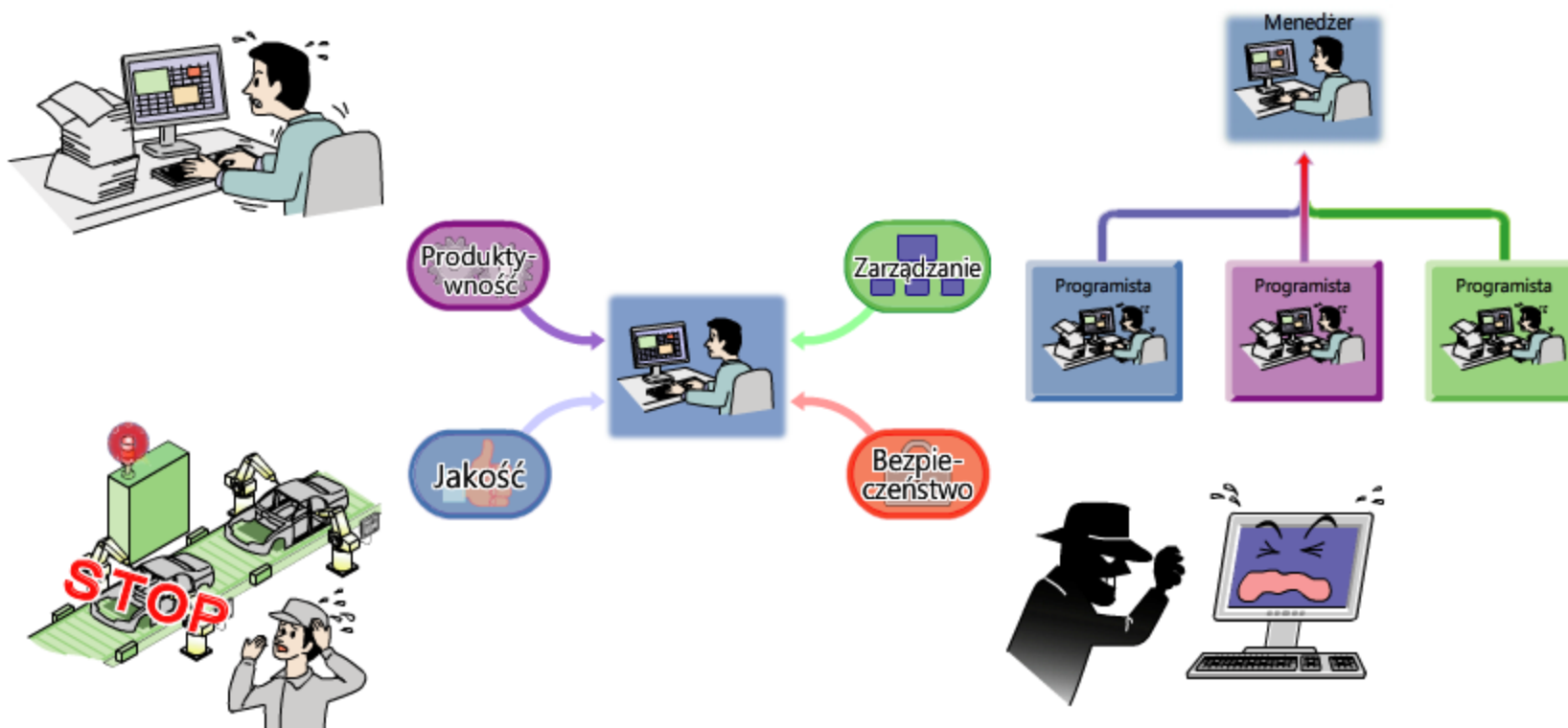
1.1

Problemy, z jakimi boryka się projektant systemu

Pomimo że od projektującego system jest wymagana poprawa produktywności i obniżenie kosztów rozwoju, musi również zapewnić, że opracowywane programy są wysokiej jakości.

Projektowany zakład musi również realizować zarządzanie projektem, umożliwiające rozwój zespołowy i zapewniające szybki powrót do normalnego stanu w przypadku problemu.

Wymagane są również środki bezpieczeństwa, ponieważ programy sekwencyjne obejmują ważną wiedzę i dane.



1.1.1 Poprawa produktywności

Od projektującego system jest wymagane tworzenia programów sekwencyjnych do obsługi złożonego przetwarzania na dużą skalę.

Jednakże koszt rozwoju wzrasta wraz z czasem rozwoju.

Z tego względu niezbędna jest poprawa produktywności podczas tworzenia programu w celu ograniczenia kosztu rozwoju.

GX Works2 zapewnia następujące funkcje do rozwiązania tego problemu:

- Etykieta
- Blok funkcyjny
- Ustawienie domyślne urządzenia i pamięć urządzenia*
- Wbudowany język strukturalny
- Import z przykładowego komentarza

* Seria MELSEC-F nie umożliwia konfiguracji ustawienia domyślnego urządzenia.

Produktywność



1.1.2 Poprawa jakości

Błąd w programie sekwencyjnym może spowodować zatrzymanie, usterkę systemu lub wypadek, które mogą wstrzymać produkcję i skutkować utratą zysku i niezawodności. Należy opracować wysokiej jakości programy bez żadnych błędów.

GX Works2 zapewnia następujące funkcje do rozwiązania tego problemu:

- Etykieta
- Blok funkcyjny
- Ustawienie domyślne urządzenia i pamięć urządzenia*
- Komentarz
- Import z przykładowego komentarza
- Obserwacja zmiennych
- Cross reference
- Sampling trace*
- Wykonalne, warunkowe testowanie urządzenia*
- Funkcja wykonania krokowego*
- Ustawienie systemu wejść/wyjść



* Seria MELSEC-F nie może stosować ustawień domyślnych urządzenia, Sampling trace , wykonalnego, warunkowego testowania urządzenia oraz funkcji wykonania krokowego.

1.1.3 Zarządzanie projektem

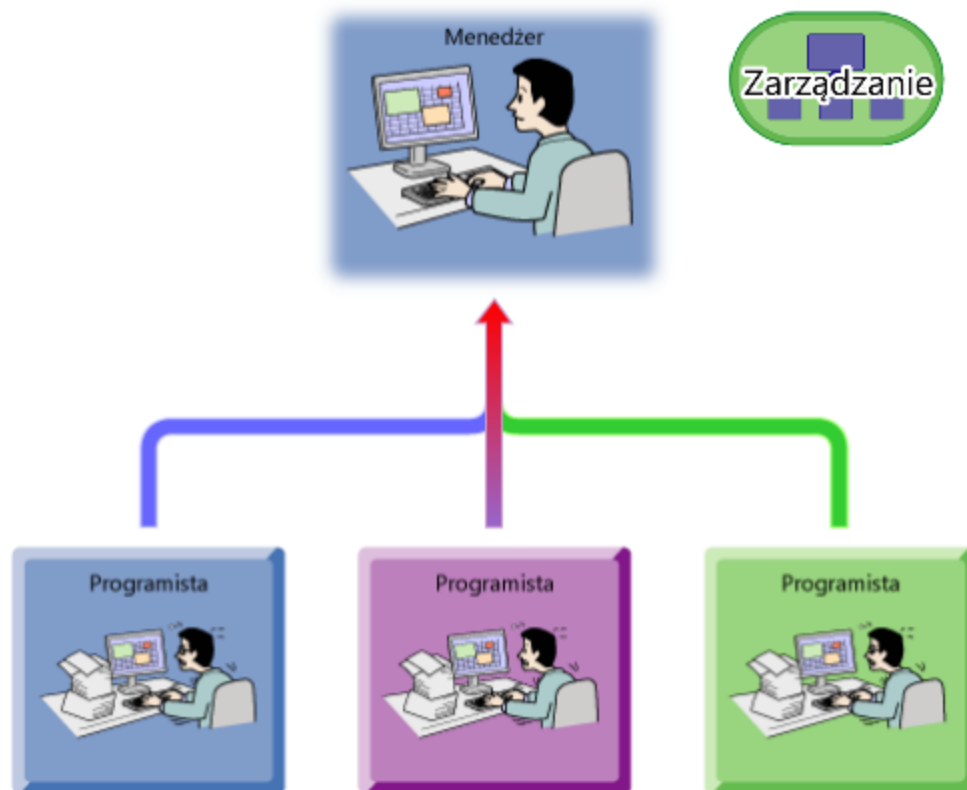
Program sekwencyjny na dużą skalę jest często opracowywany przez zespół programistów.

Z punktu widzenia uprawnień do zarządzania poszczególnych osób i poufności danych należy ustalić limity wobec zakresu dostępnych danych i stosowanych funkcji.

Ponadto, w celu zapewnienia szybkiego powrotu do normalnego stanu w przypadku utraty programu z powodu usterki sterownika programowalnego należy wdrożyć opcję zarządzania wersją programu oraz okresowe tworzenie kopii zapasowej programu.

GX Works2 zapewnia następujące funkcje do rozwiązania tych problemów:

- Bezpieczeństwo
- Historia zmian
- Weryfikacja przy użyciu komputera



1.1.4 Środki bezpieczeństwa

Program sekwencyjny obejmuje ważne pod względem strategicznym dane oraz wiedzę.

Wyciek know-how oraz danych z programu na zewnątrz może mieć niszczący wpływ na firmę.

Nieupoważnione modyfikacje programu mogą prowadzić do problemów podczas produkcji, takich jak zatrzymanie systemu. Należy podjąć odpowiednie środki bezpieczeństwa w celu zapobiegania tym problemom.

GX Works2 zapewnia następującą funkcję do rozwiązania tych problemów:

- Bezpieczeństwo



1.2 Procedura uczenia

W ramach tego kursu przedstawione zostały funkcje GX Works2 poprzez procedurę faktycznego rozwoju systemu.

(1) Programowanie..... Rozdział 2



(2) Wyszukiwanie i usuwanie usterek..... Rozdział 3



(3) Zarządzanie projektem i środki bezpieczeństwa..... Rozdział 4

<Wyjaśnienie ikon>

Ikony wyświetlone na stronach rozdziału 2, 3 i 4 odpowiadają odpowiednim funkcjom usprawniającym, jak pokazano poniżej.



Funkcje stosowane do poprawy produktywności programu



Funkcje stosowane do poprawy jakości programu



Funkcje stosowane do zarządzania projektem



Funkcje stosowane w odniesieniu do środków bezpieczeństwa

Rozdział 2 Programowanie

Kroki procesu uczenia w Rozdziale 2

W rozdziale 2 przedstawione zostały funkcje stosowane w celu programowania.

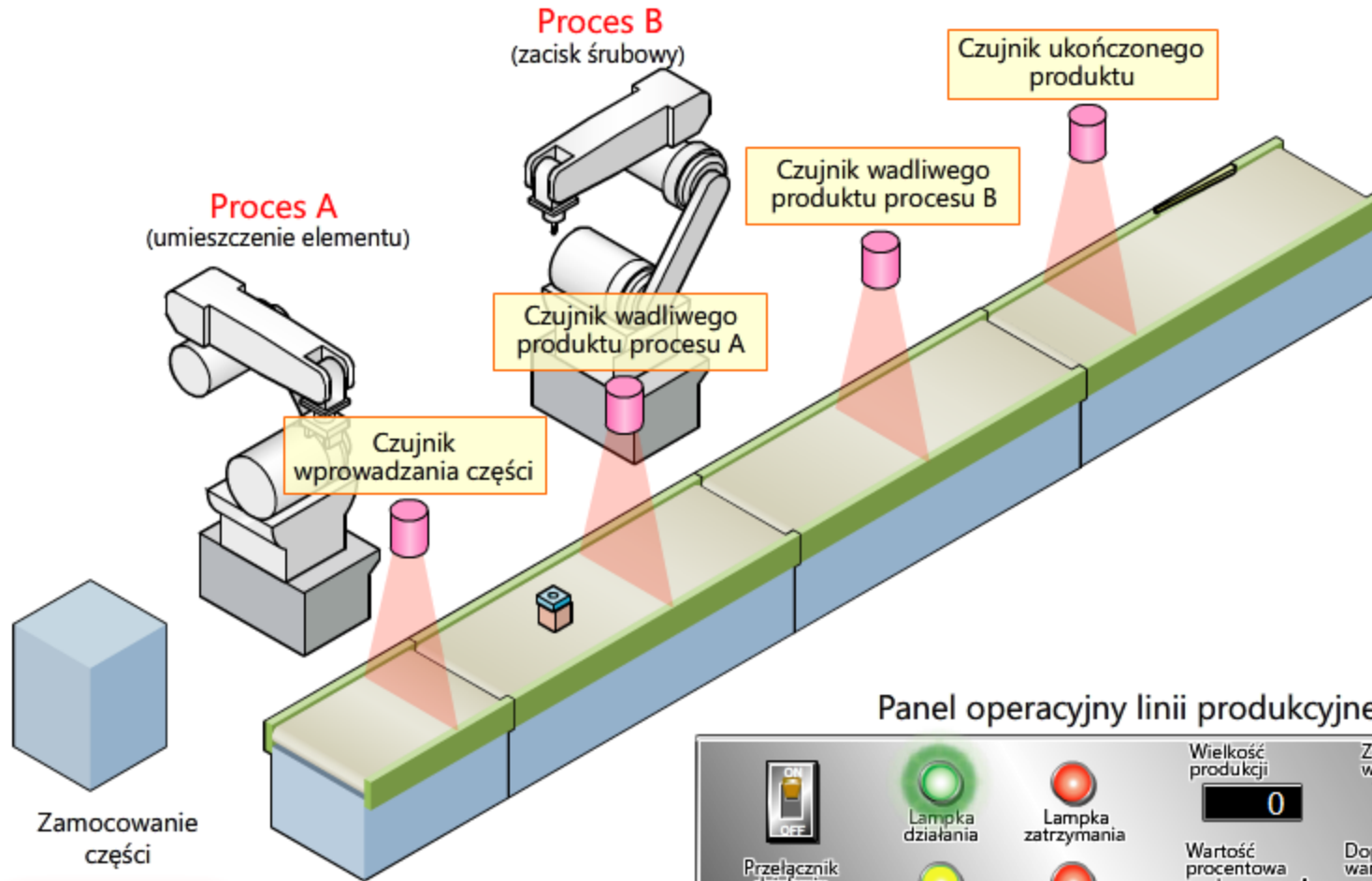
GX Works2 zapewnia wiele różnych funkcji do zapewnienia skutecznego programowania i poprawy jakości programów.

- 2.1 Zastępowanie nazwy urządzenia nazwą związaną z zastosowaniem
 - 2.1.1 Rodzaje etykiet
 - 2.1.2 Rodzaje zastosowań etykiet i zapisanych wartości
 - 2.1.3 Rejestracja etykiety
 - 2.1.4 Automatyczne przypisanie etykiet do urządzeń
- 2.2 Ustalenie powtarzalnie stosowanych bloków drabinki jako bloków funkcyjnych
 - 2.2.1 Tworzenie i umieszczanie bloków funkcyjnych
 - 2.2.2 Korzystanie z biblioteki bloków funkcyjnych
- 2.3 Zmiana ustawienia domyślnego urządzenia bez poprawy programu
- 2.4 Uproszczenie programu drabinkowego
- 2.5 Tworzenie łatwych w zrozumieniu i czytaniu programów
 - 2.5.1 Zapisywanie komentarza dla każdego bloku drabinki
 - 2.5.2 Zapisywanie komentarza dla każdego wyjścia (cewka, instrukcja)
- 2.6 Ułatwienie odczytu programów zawierających znaczniki flagi/rejestry specjalne

Rozdział 2 Przedstawienie systemu stosowanego do nauki podczas tego kursu



Niniejszy kurs wykorzystuje poniższy hipotetyczny system jako pomoc w poznaniu sposobu stosowania funkcji GX Works2:



Normalna produkcja
 Błąd wartości procentowej procesu A
 Błąd wartości procentowej procesu B

Naciśnięcie przycisku „Wróć do góry” spowoduje powrót do pierwszego wyboru działania.

Wróć do góry

Panel operacyjny linii produkcyjnej

| | | | | |
|-------------------------------|---------------------------------|---|--|--|
| Przełącznik działania | Lampka działania | Lampka zatrzymania | Wielkość produkcji 0 | Zaplanowana wielkość produkcji 20 |
| Reset wielkości produkcji | Uzyskana wielkość produkcji | Lampka błędu wartości procentowej wad procesu A | Wartość procentowa wad procesu A 0 | Dopuszczalna wartość procentowa wad procesu A 10 |
| | | Lampka błędu wartości procentowej wad procesu B | Wartość procentowa wad procesu B 0 | Dopuszczalna wartość procentowa wad procesu B 5 |

2.1

Zastępowanie nazwy urządzenia nazwą związaną z zastosowaniem

Produktywność

Jakość

Stosowane **urządzenie** w programie sekwencyjnym posiada nazwę składającą się z liter wraz z numerem, np. „M0” lub „D5”. Nazwa urządzenia nie zapewnia więc żadnych wskazówek dotyczących jego zastosowania. Program na dużą skalę korzysta z różnych rodzajów urządzeń, co oznacza, że podczas programowania należy stale sprawdzać dokumentację projektową systemu w celu określenia zastosowania urządzeń. Zmniejsza to wydajność pracy i negatywnie wpływa na jakość programu z powodu błędów podczas wyboru urządzeń.

W celu rozwiązania tych problemów można zastosować **„etykiety”**.

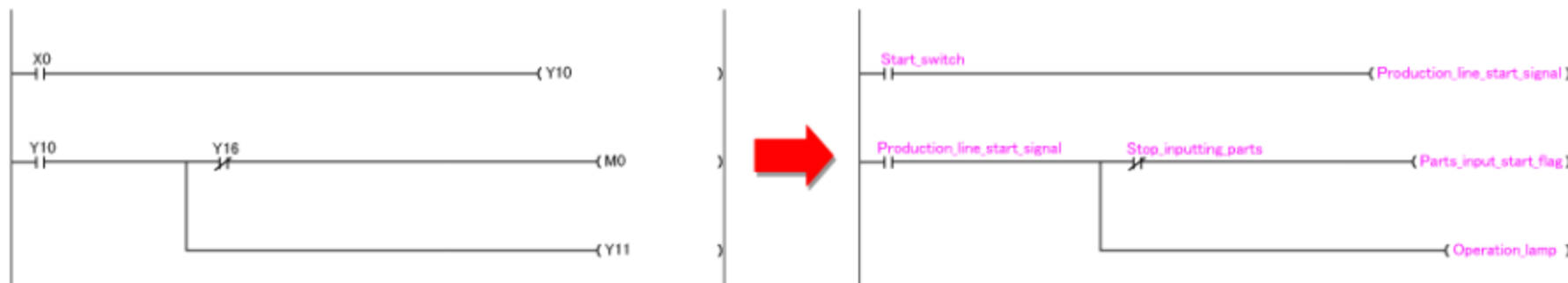
Zamiast stosować nazwę urządzenia, można użyć nazwy (etykiety), która określa faktyczne zastosowanie np. **„sygnał uruchomienia linii produkcyjnej”**.

Dla tego rodzaju nazwy mogą być stosowane znaki japońskie (hiragana, katakana i kanji) oprócz znaków alfanumerycznych.

<Wyjaśnienie ustawienia „M0” zawierającego WŁĄCZENIE sygnału uruchomienia linii produkcyjnej przez instrukcję SET (ustawienie)>

| | |
|------------|---|
| Urządzenie | —[SET M0]— |
| Etykieta | —[SET Sygnał uruchomienia linii produkcyjnej]— |

Stosowanie tego typu etykiet jest skuteczne do tworzenia łatwego w odczycie programu, poprawy skuteczności rozwoju programu i zapobiegania błędów dotyczących wprowadzania urządzeń.



2.1.1 Rodzaje etykiet

Produktywność

Jakość

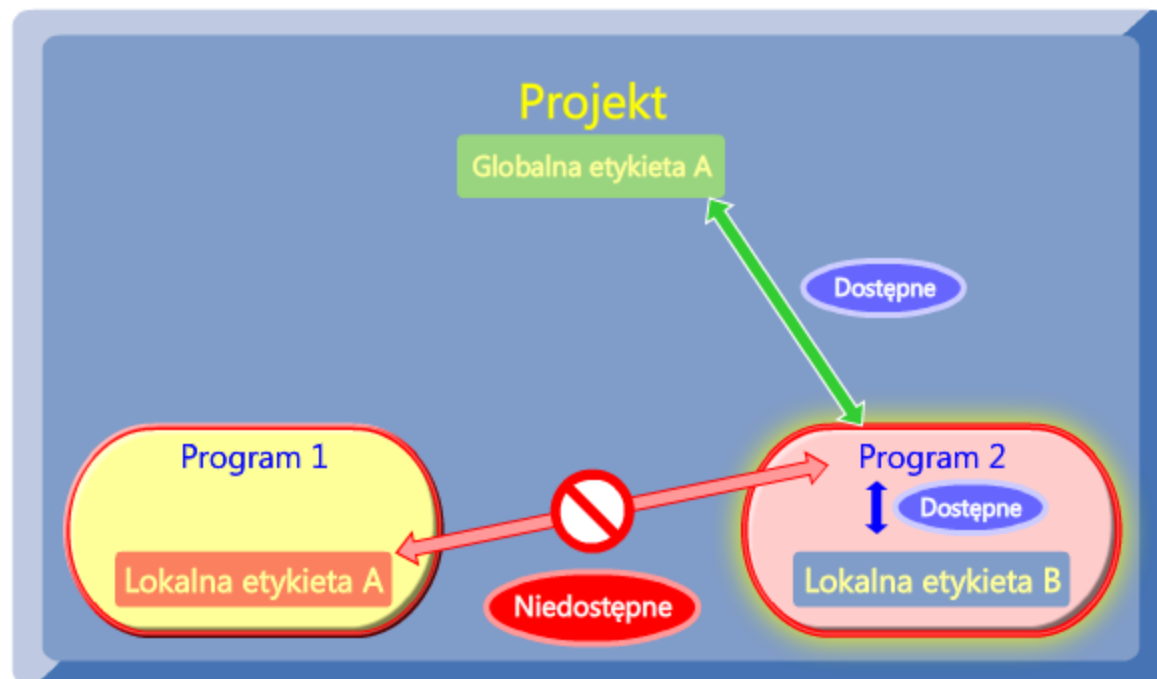
Dostępne są dwa rodzaje etykiet: „etykieta globalna” i „etykieta lokalna”.

<Etykieta globalna>

Etykiety globalne są stosowane dla całego projektu i można do nich uzyskać dostęp z poziomu dowolnego programu tego projektu.

<Etykieta lokalna>

Etykiety lokalne są stosowane w określonym programie i można do nich uzyskać dostęp z poziomu programu, w którym etykieta jest zarejestrowana.



„Program 1” (Program 1) i „Program 2” (Program 2) to przyciski. Naciśnij dowolny program, aby sprawdzić, czy może on uzyskać dostęp do dwóch rodzajów etykiet.

2.1.2

Rodzaje zastosowań etykiet i zapisanych wartości

Produktywność

Jakość

Podczas rejestracji etykiety należy określić zastosowanie etykiety oraz rodzaj wartości, które można zapisać przy użyciu opcji „**Klasa**” oraz „**Typ danej**”.

<Klasa>

Klasa określa zakres wykorzystania oraz zastosowania etykiety.

Klasy, które można wybrać różnią się w zależności od rodzaju etykiety.

| Klasa | Przeznaczenie | Obszar ustawienia etykiety | | |
|---------------------|--|-------------------------------------|-------------------------------------|-------------------------------------|
| | | Etykieta globalna | Etykieta lokalna programu | Etykieta lokalna bloku funkcyjnego |
| VAR_GLOBAL | Etykieta wspólna, która może być stosowana przez programy i bloki funkcyjne w ramach projektu | <input type="radio"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| VAR_GLOBAL_CONSTANT | Etykieta wspólna z wartością stałą, która może być stosowana przez programy i bloki funkcyjne w ramach projektu | <input type="radio"/> | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> |
| VAR | Etykieta, która może być stosowana przez programy i bloki funkcyjne, do których etykieta jest przeznaczona | <input checked="" type="checkbox"/> | <input type="radio"/> | <input type="radio"/> |
| VAR_CONSTANT | Etykieta z wartością stałą, która może być stosowana przez programy i bloki funkcyjne, do których etykieta jest przeznaczona | <input checked="" type="checkbox"/> | <input type="radio"/> | <input type="radio"/> |
| VAR_RETAIN | Etykieta (typ zamykany), która może być stosowana przez programy i bloki funkcyjne, do których etykieta jest przeznaczona | <input checked="" type="checkbox"/> | <input type="radio"/> | <input type="radio"/> |
| VAR_INPUT | Etykieta stosowana do wejść bloku funkcyjnego, do którego etykieta jest przeznaczona * Wartości nie można zmienić w elementach programu. | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="radio"/> |
| VAR_OUTPUT | Etykieta stosowana do wyjść bloku funkcyjnego, do którego etykieta jest przeznaczona | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="radio"/> |
| VAR_IN_OUT | Etykieta stosowana do wejść i wyjść bloku funkcyjnego, do którego etykieta jest przeznaczona * Wartość można zmienić w elementach programu. | <input checked="" type="checkbox"/> | <input checked="" type="checkbox"/> | <input type="radio"/> |

2.1.2

Rodzaje zastosowań etykiet i zapisanych wartości

Produktywność

Jakość

<Typ danej>

Typ danej odnosi się do rodzaju wartości zapisanej w etykiecie.

Typ danej przypisanej do etykiety określa typ i zakres wartości, które mogą być zapisane w etykiecie i odpowiadającemu jej urządzeniu.

Typy danych, które mogą być stosowane z programami drabinkowymi, są wymienione poniżej.

| Typ danej | Opis | Długość bitu | Zakres zapisanych wartości |
|--|--|--------------|--|
| Bit (Bit) | Zapisana jest wartość ON (WŁ.) lub OFF (WYŁ.). Odnosi się do urządzenia „M”. | 1 bit | 1: ON (WŁ.), 0: OFF (WYŁ.) |
| Word (signed) (Słowo danych (ze znakiem)) | Zapisywana jest liczba całkowita bez ułamków. Odnosi się do urządzenia „D”. | 16 bitów | -32768 do 32767 |
| Double-word (signed) (Słowo podwójne (bez znaku)) | | 32 bity | -2147483648 do 2147483647 |
| FLOAT (single precision) (ZMIENNE (dokładność do jednego miejsca)) | Zapisywana jest liczba rzeczywista z ułamkami. Odnosi się do urządzenia „D”. | 16 bitów | -2^{128} do -2^{-126} , 0, 2^{-126} do 2^{128} |
| FLOAT (double precision) (ZMIENNE (dokładność do dwóch miejsc)) | | 32 bity | -2^{1024} do -2^{-1022} , 0, 2^{-1022} do 2^{1024} |
| String (Ciąg) | Zapisywany jest ciąg znaków. Odnosi się do urządzenia „D”. | Zmienna | Do 255 znaków |
| Timer (Timer) | WŁĄCZA się po osiągnięciu określonego czasu. Odnosi się do urządzenia – timera „T”. | - | - |
| Retentive timer (Timer retencji) | WŁĄCZA się po osiągnięciu określonego czasu. Odnosi się do urządzenia – timera retencji „ST”. | - | - |
| Counter (Licznik) | WŁĄCZA się po osiągnięciu określonej ilości. Odnosi się do urządzenia – licznika „C”. | - | - |
| Pointer (Wskaźnik) | Zapisywane jest położenie uruchomienia podprogramu standardowego. Odnosi się do urządzenia – wskaźnika „P”. | - | - |

2.1.3

Rejestracja etykiety

Produktywność

Jakość

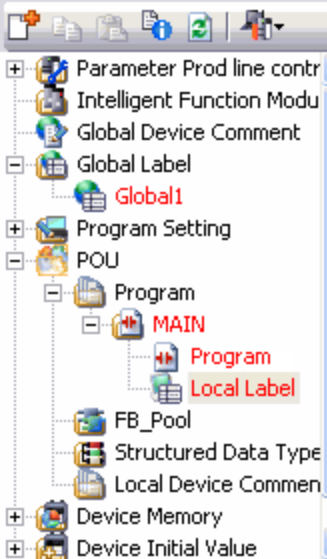
MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [Local Label Setting MAIN [PRG]]

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



Navigation

Project




Project

User Library

Connection Destination

| | Class | Label Name | Data Type | Constant |
|----|-------|-------------------------|-----------|----------|
| 1 | VAR | Production_qty_attained | Bit | ... |
| 2 | VAR | Parts_input_start_flag | Bit | ... |
| 3 | VAR | Start_reject_pct_calc | Bit | ... |
| 4 | | | | ... |
| 5 | | | | ... |
| 6 | | | | ... |
| 7 | | | | ... |
| 8 | | | | ... |
| 9 | | | | ... |
| 10 | | | | ... |
| 11 | | | | ... |
| 12 | | | | ... |
| 13 | | | | ... |
| 14 | | | | ... |
| 15 | | | | ... |
| 16 | | | | ... |
| 17 | | | | ... |
| 18 | | | | ... |
| 19 | | | | ... |
| 20 | | | | ... |
| 21 | | | | ... |
| 22 | | | | ... |
| 23 | | | | ... |
| 24 | | | | ... |
| 25 | | | | ... |
| 26 | | | | ... |

Zakończ ustawienia rejestracji etykiety.

Kliknij  , aby kontynuować.

English

Simple

Q03UDE

Host Station

Line No.:



2.1.4

Automatyczne przypisanie etykiet do urządzeń

Produktywność

Jakość

Konwertowanie programu powoduje automatyczne przypisanie odpowiedniego urządzenia do etykiety zgodnie z klasą i typem danej.

W przypadku korzystania z etykiety nie trzeba wiedzieć, które urządzenie zostało przypisane.

Należy skorzystać z opcji „**Device/Label Automatic-Assign Setting**” (ustawienie automatycznego przypisania urządzenia/etykiety), aby zmienić zakres urządzeń przypisanych do etykiety.

* Ekran ten jest oknem ustawienia automatycznego przypisania urządzenia serii MELSEC-Q i MELSEC-L. Ekran może się różnić od serii MELSEC-F.

<Uruchomienie „Device/Label Automatic-Assign Setting” (ustawienie automatycznego przypisania urządzenia/etykiety)>
W menu GX Works2 wybierz [Tool] (narzędzie) – [Device/Label Automatic-Assign Setting] (ustawienie automatycznego przypisania urządzenia/etykiety).

Device/Label Automatic-Assign Setting

Set a device range to automatically assign to labels.

Labels will be assigned from its way down the displayed device list when multiple devices are selected.

| | Device | Digit | Assign Selection | Assignment Range | | Total Points | PLC Parameter Device Setting Range |
|-----------------------------|----------|-------|-------------------------------------|------------------|-------|--------------|------------------------------------|
| | | | | Start | End | | |
| Word Device | | | | | | | |
| VAR Range | D | 10 | <input checked="" type="checkbox"/> | 6144 | 12207 | 6144 | 0 -- 12207 |
| | W | 16 | <input type="checkbox"/> | | | | 0 -- 1FFF |
| | R | 10 | <input type="checkbox"/> | | | | |
| VAR_RETAIN Range [Latch(1)] | D Latch | 10 | <input type="checkbox"/> | | | 0 | |
| | W Latch | 16 | <input type="checkbox"/> | | | | |
| | ZR Latch | 10 | <input type="checkbox"/> | | | | |
| Bit Device | | | | | | | |
| VAR Range | M | 10 | <input checked="" type="checkbox"/> | 4096 | 8191 | 4096 | 0 -- 8191 |
| | B | 16 | <input type="checkbox"/> | | | | 0 -- 1FFF |
| VAR_RETAIN Range [Latch(1)] | L Latch | 10 | <input type="checkbox"/> | | | 0 | |
| | B Latch | 16 | <input type="checkbox"/> | | | | |
| Pointer | | | | | | | |
| VAR Range | P | 10 | <input checked="" type="checkbox"/> | 2048 | 4095 | 2048 | 2048 -- 4095 |
| Timer | | | | | | | |
| VAR Range | T | 10 | <input checked="" type="checkbox"/> | 64 | 2047 | 1984 | 0 -- 2047 |
| VAR_RETAIN Range [Latch(1)] | T Latch | 10 | <input type="checkbox"/> | | | 0 | |
| Retentive Timer | | | | | | | |
| VAR Range | ST | 10 | <input type="checkbox"/> | | | 0 | |
| VAR_RETAIN Range [Latch(1)] | ST Latch | 10 | <input type="checkbox"/> | | | 0 | |
| Counter | | | | | | | |
| VAR Range | C | 10 | <input checked="" type="checkbox"/> | 512 | 1023 | 512 | 0 -- 1023 |
| VAR_RETAIN Range [Latch(1)] | C Latch | 10 | <input type="checkbox"/> | | | 0 | |

Latch(1) : Able to clear the value by using a latch clear.
Latch(2) : Unable to clear the value by using a latch clear. Clearing will be executed by remote operation or program.

(Caution)
1. Label-nonassigned devices, of the automatically assigned ones while compiling, will be allotted the device that displayed at the lowest of the selected ones. Ex:) Device will be assigned to ZR when D and ZR are selected.
2. Changing the assignment target device may also change the processing speed since the arithmetic processing speed for R and ZR is difference from other devices.

OK Cancel

Robot control - [[PRG]Write MAIN 194 Step]

Tool Window Help

- IC Memory Card
- Check Parameter...
- Options...
- Key Customize...
- Device/Label Automatic-Assign Setting...**
- Block Password...
- Confirm Memory Size...
- Set TEL Data/Connect via Modem
- LCPU Logging Configuration Tool...
- Ethernet Adapter Module Configuration Tool...
- Built-in I/O Module Tool
- Check Intelligent Function Module Parameter
- Intelligent Function Module Tool
- Language Selection...

2.2

Ustalenie powtarzalnie stosowanych bloków drabinki jako bloków funkcyjnych

Produktywność

Jakość

W programach na dużą skalę niektóre bloki drabinki mogą być stosowane powtarzalnie lub niektóre z bloków drabinki mogą być stosowane w innych programach.

Wydajność pracy nie może zostać poprawiona, jeśli konieczne jest wprowadzanie tych samych bloków drabinki, za każdym razem, gdy to konieczne.

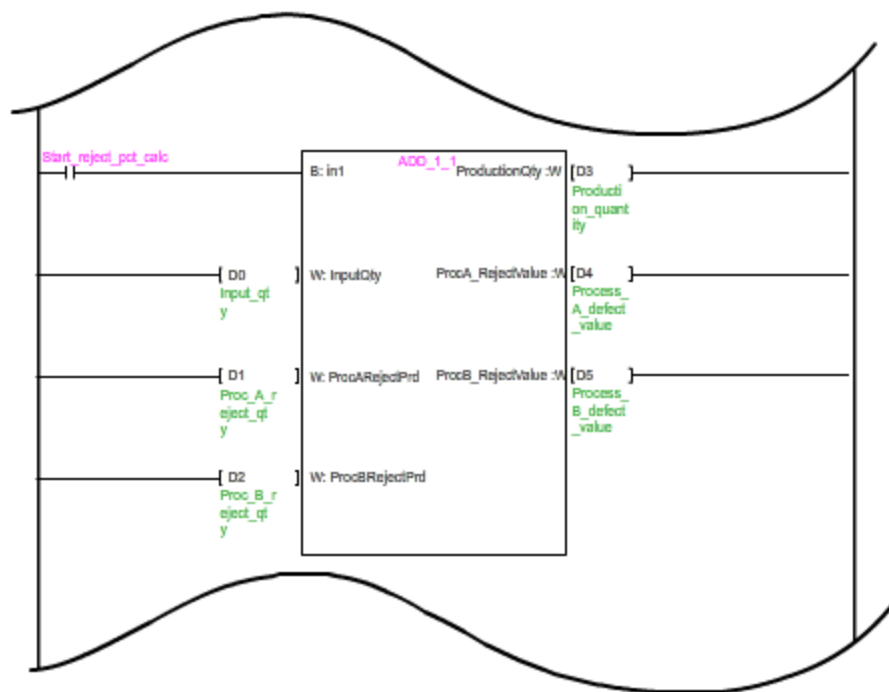
Ponadto, w przypadku znalezienia błędu we wspólnym bloku drabinki, każdy blok musi zostać znaleziony i poprawiony. Powoduje to znaczący strata czasu.

Należy skorzystać z opcji „**blok funkcyjny (FB)**”, aby rozwiązać te problemy.

Blok drabinki, który jest stosowany powtarzalnie, może zostać **ustalony jako blok funkcyjny**, który może być stosowany w każdym programie. W dużym stopniu poprawia to wydajność pracy.

W przypadku znalezienia błędu we wspólnym bloku, wystarczy tylko poprawić blok funkcyjny.

Pozwala to nie tylko na zaoszczędzenie czasu, lecz także zapobiega ryzyku niepoprawienia jednego z bloków.



Naciśnij przycisk, aby obejrzeć animację od początku.



2.2.1

Tworzenie i umieszczanie bloków funkcyjnych

Ze względu na to, że blok funkcyjny będzie stosowany w różnych programach, faktyczne urządzenia (takie jak Z, Y i D) nie mogą być stosowane w bloku funkcyjnym. Urządzenia te muszą zostać zastąpione przez **etykiety** w sposób przedstawiony w punkcie 2.1.

Procedura ustalania bloków drabinki na blok funkcyjny została przedstawiona w animacji.

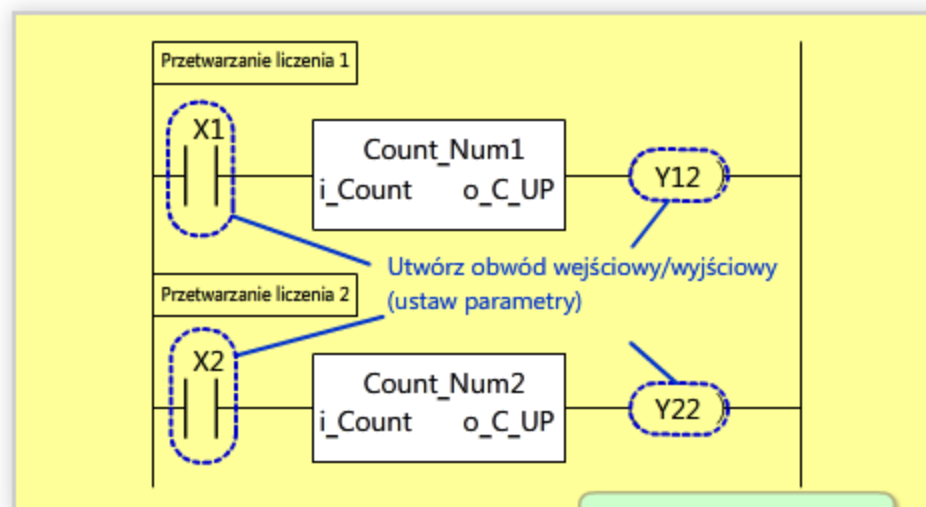
1. Przygotuj program do zorganizowania w postaci bloku funkcyjnego.

2. Podziel program na wejścia i wyjścia i zastąp urządzenia wewnętrzne na etykiety wewnętrzne, aby utworzyć blok funkcyjny.

3. Umieść utworzony blok funkcyjny w programie przy użyciu opcji przeciągnij i upuść.

4. Przypisz parametry do etykiety wejściowej i wyjściowej bloku funkcyjnego umieszczonego w programie.

5. Utwórz obwód wejściowy, który przekazuje parametr do etykiety wejściowej oraz obwód wyjściowy, który odbiera parametr od etykiety wyjściowej przed i za blokiem funkcyjnym.



Naciśnij przycisk, aby wrócić do pierwszego przepływu.



2.2.1

Tworzenie i umieszczanie bloków funkcyjnych

Produktywność

Jakość

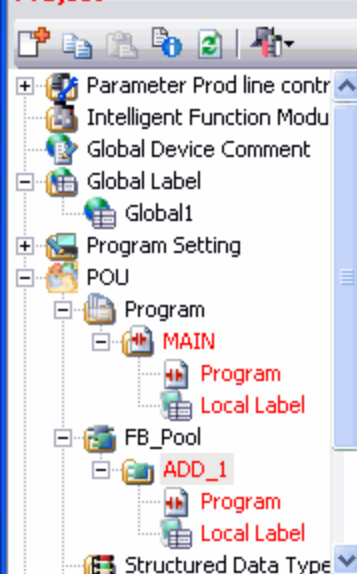
MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (188)Step *]

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



Navigation [PRG]Write MAIN (188)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG] Function/FB Lat

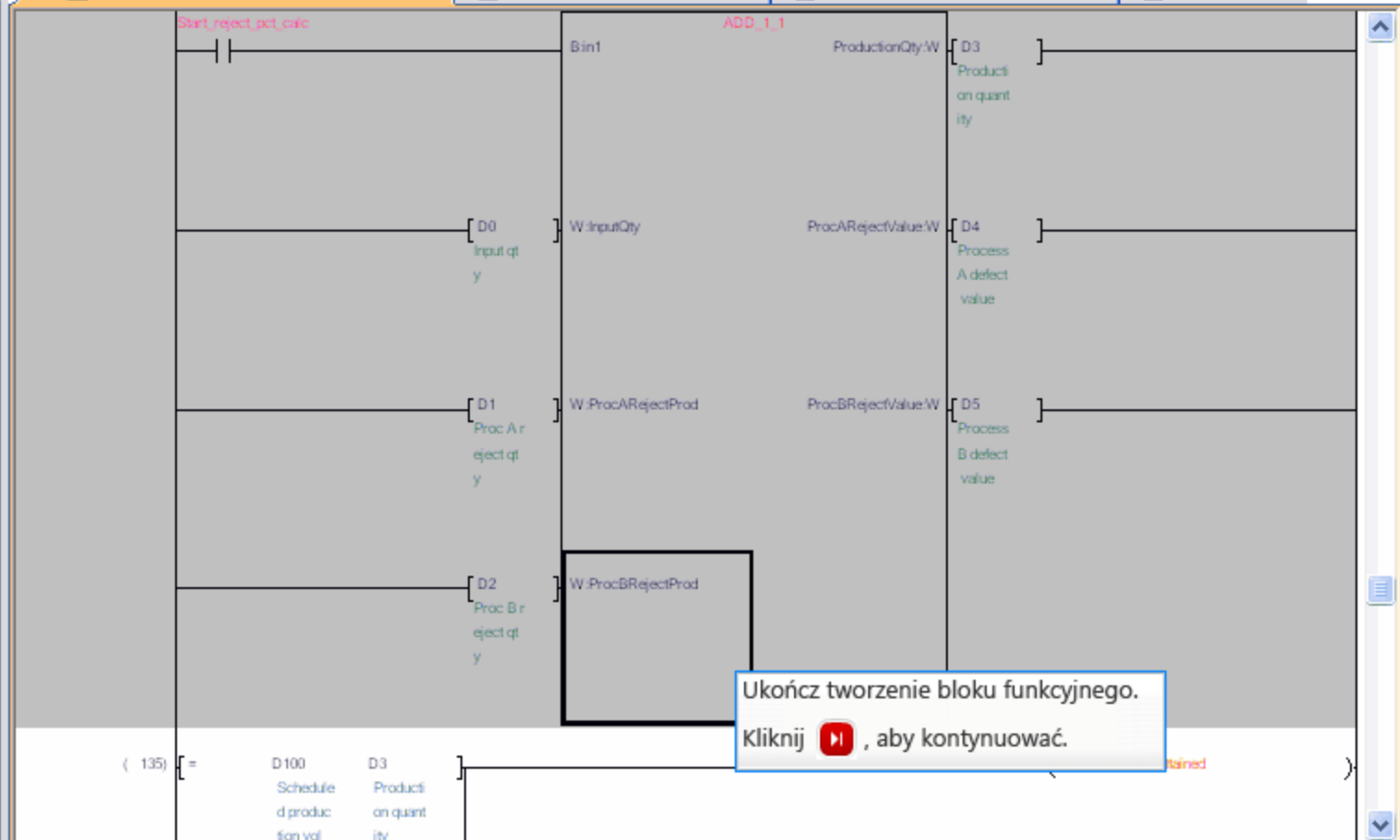
Project




Project

User Library

Connection Destination



Ukończ tworzenie bloku funkcyjnego.

Kliknij , aby kontynuować.

English

Simple

Q03UDE

Host Station

(13 ML)



2.2.2

Korzystanie z biblioteki bloków funkcyjnych

Przedstawiliśmy sposób tworzenia bloku funkcyjnego.

Firma Mitsubishi Electric uporządkowała programy sterujące różnych modułów, takich jak CPU, moduły analogowych wejść/wyjść, sieci i moduły pozycjonujące jako bloki funkcyjne, które są dostępne bezpłatnie jako „**FB Library**”.

Korzystanie z biblioteki FB eliminuje konieczność opracowania programów sterujących do modułów, co było zazwyczaj obowiązkiem użytkownika. Istnieje także możliwość wdrożenia modułów, z którymi użytkownik nie jest zaznajomiony.

Bibliotekę FB można pobrać ze strony „**MELSOFT Library Download**” na stronie internetowej **MITSUBISHI ELECTRIC FA**.

<Wcześniej>



<W przypadku korzystania z biblioteki FB>



* Dla serii MELSEC-F biblioteka FB nie jest dostępna.

Obsługiwane modele są wymienione na stronie pobierania. Należy sprawdzić obsługiwane modele przed pobraniem.

2.3 Zmiana ustawienia domyślnego urządzenia bez poprawy programu

Ustawienie domyślne lub wartość stała urządzenia są zazwyczaj ustawiane przy użyciu instrukcji MOV przed przetworzeniem programu głównego.

W tym przypadku program musi być bezpośrednio poprawiony za każdym razem, gdy działanie programu jest zmieniane zgodnie z zastosowaniem systemu.

Metoda ta jest nie tylko czasochłonna, lecz także istnieje ryzyko błędów podczas korekty lub niepowodzenia jej wprowadzenia.

Należy skorzystać z opcji „**Device Initial Value**”, aby rozwiązać te problemy.

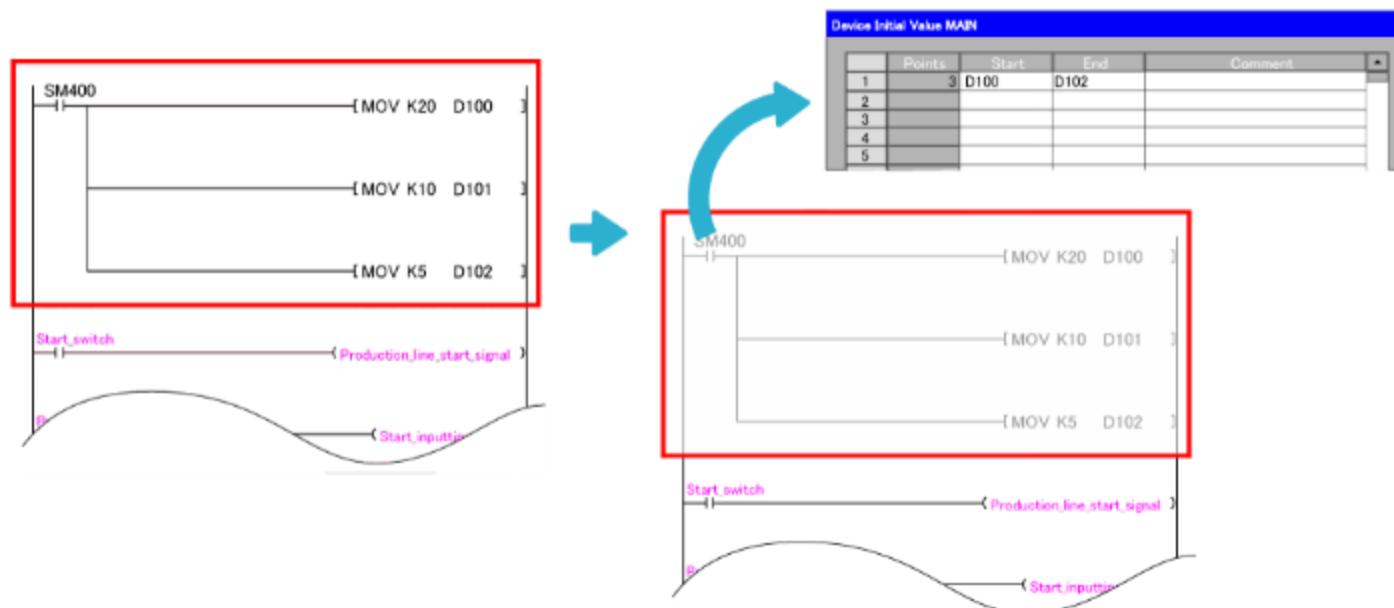
Korzystanie z funkcji GX Works2 gwarantuje prawidłowe zarządzanie ustawieniem domyślnym urządzenia i eliminuje konieczność wprowadzania poprawek do programu, co pozwala na bardziej skuteczne tworzenie programów.

Ponadto funkcja ta eliminuje konieczność istnienia programu do konfiguracji ustawień domyślnych urządzeń, co może zmniejszyć wielkość programu (wykorzystanie pamięci) i zmniejszyć ryzyko usterki.

Dla ustawienia domyślnego urządzenia należy określić zakres urządzeń, dla których ma zostać skonfigurowane ustawienie domyślne.

Faktyczne ustawienia domyślne są zapisywane w **pamięci urządzenia** i są przypisane do określonego zakresu urządzeń.

Przy odpowiednio przygotowanych obszarach pamięci urządzenia zgodnie z zastosowaniami systemu, ustawienie domyślne urządzenia można zmienić, zmieniając obszar pamięci urządzenia, który ma zostać przypisany.



2.3 Zmiana ustawienia domyślnego urządzenia bez poprawy programu

Produktywność

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN (166)Step *]

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



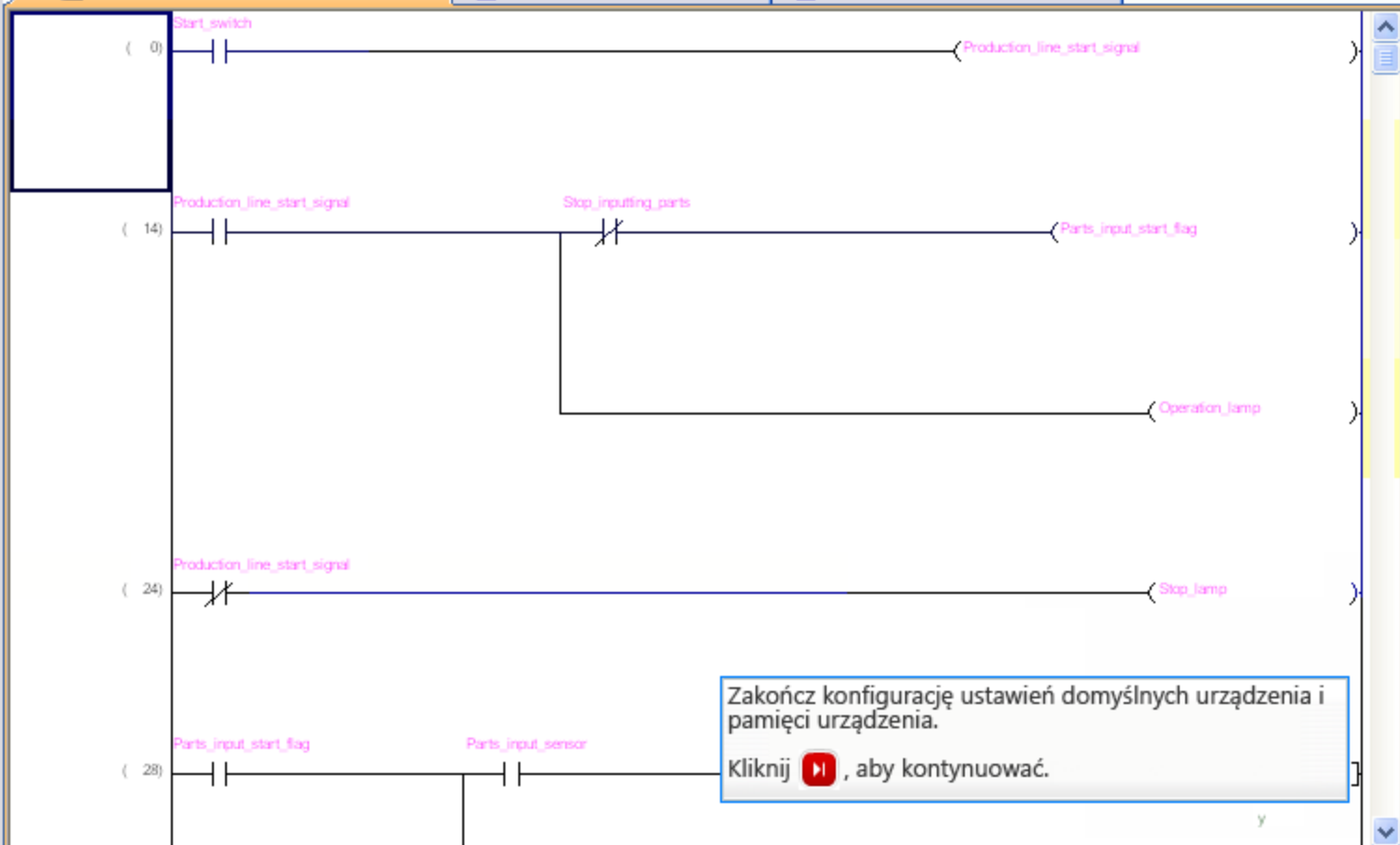
Navigation [PRG]Write MAIN (166)Ste... Global Label Setting Global1 Local Label Setting MAIN [PRG]

Project

- Global1
 - Program Setting
 - POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Type
 - Local Device Comment
- Device Memory
 - MAIN
 - ProdQty1
- Device Initial Value
 - MAIN

Project

- User Library
- Connection Destination



English

Simple

Q03UDE

Host Station

(0/ N/5



2.4

Uproszczenie programu drabinkowego

Produktyność

Jakość

W programach drabinkowych na dużą skalę połączenie urządzeń, instrukcje i linie proste mogą stać się nieco skomplikowane, co sprawia, że ustalenie przetwarzania wykonywanego przez program staje się trudne.

W szczególności obliczenia liczbowe, takie jak wzory wypełnione w pojedynczej linii muszą być programowane przy użyciu kombinacji instrukcji.

Należy skorzystać z opcji „**Inline Structured Text**”, aby rozwiązać te problemy.

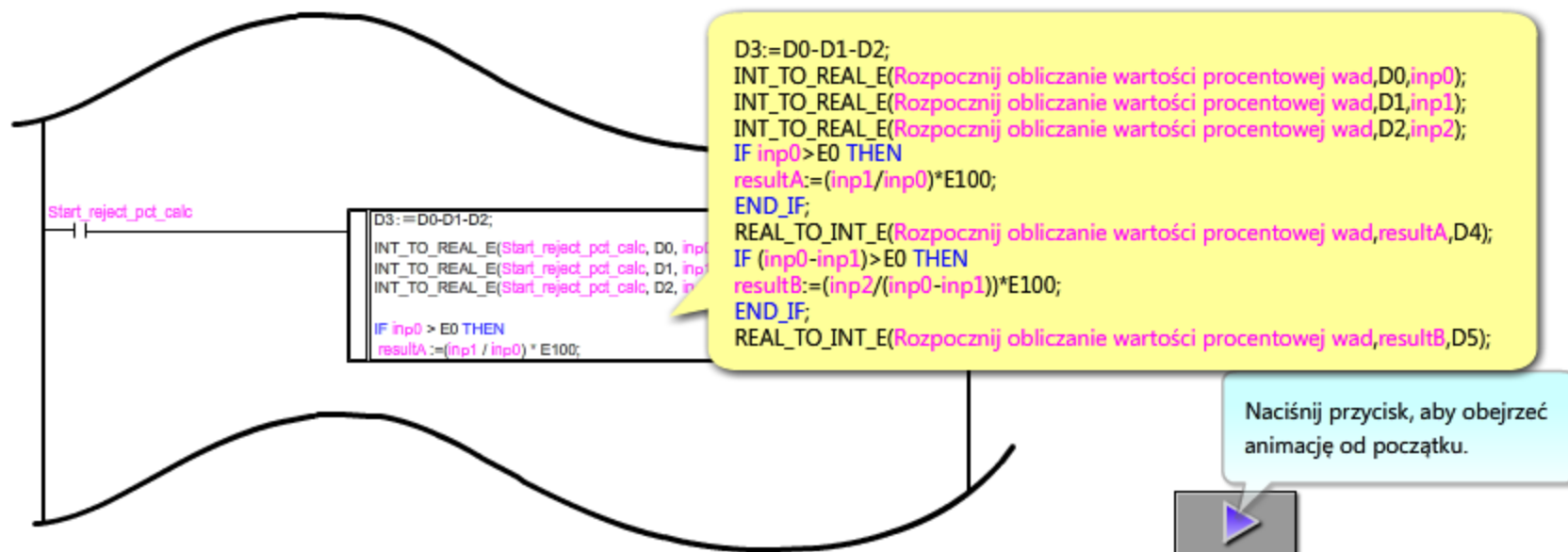
Program drabinkowy zostanie zastąpiony programem częściowo napisanym w **języku strukturalnym (ST)**.

Język ST to język programowania sterowania sekwencyjnego podobny do języka C stosowanego do programowania oprogramowania komputerowego.

Obliczenia liczbowe można napisać przy użyciu wzorów, więc nawet programiści, którzy nie znają języka C mogą korzystać z wbudowanego języka strukturalnego.

Poniższy rysunek przedstawia przykład, w którym część programu drabinkowego systemu jest zastępowana przez wbudowany język strukturalny.

Jak widać, skomplikowany program staje się łatwy do zrozumienia.



2.4 Uproszczenie programu drabinkowego



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 114 Step]

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

Navigation icons: File, Edit, View, Run, Stop, etc.

Navigation

[PRG]Write MAIN 114 Step Global Label Setting Global1 Local Label Setting MAIN [PRG]

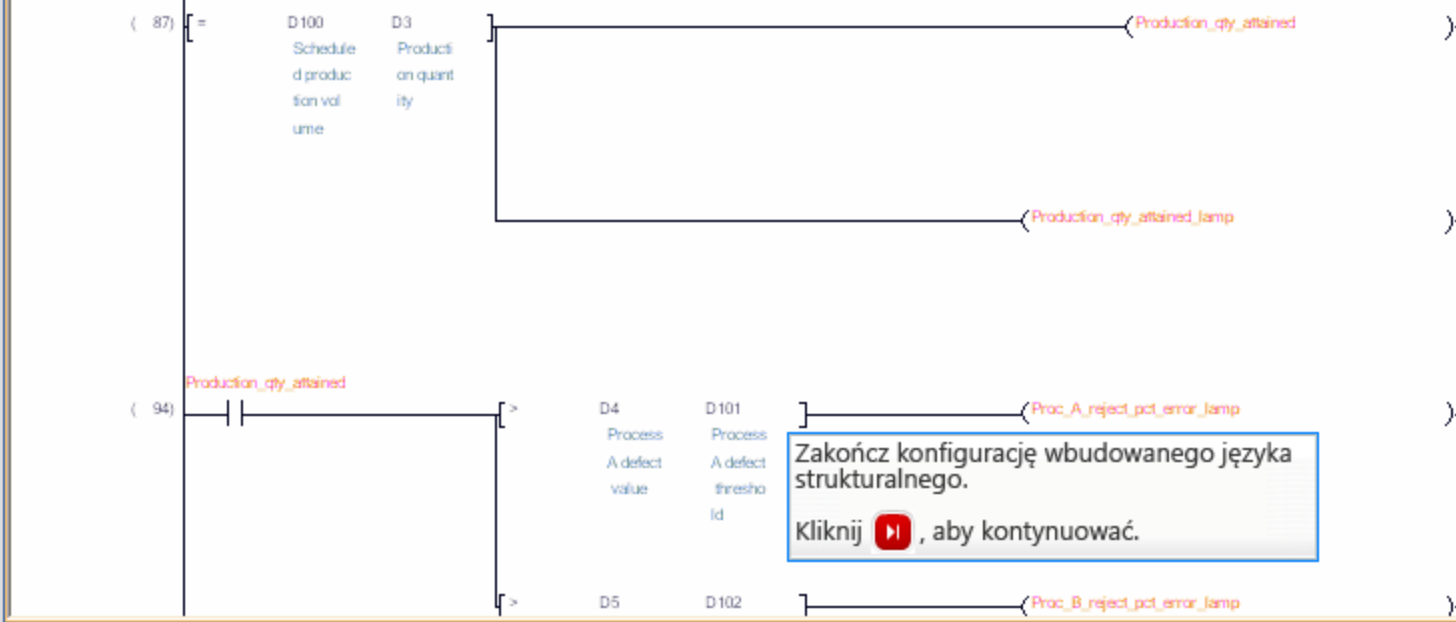
Project


- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
 - Global1
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Type
 - Local Device Commen
- Device Memory
- Device Initial Value

```

Start_reject_pct_calc
( 75)
INT_TO_REAL_E(Start_reject_pct_calc.D2.inp2);
IF inp0>E0 THEN
resultA:=(inp1/inp0)*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc.resultA,D4);
IF (inp0-inp1)>E0 THEN
resultB:=(inp2/(inp0-inp1))*E100;
END_IF;
REAL_TO_INT_E(Start_reject_pct_calc.resultB,D5)

```



Zakończ konfigurację wbudowanego języka strukturalnego.
 Kliknij , aby kontynuować.

Navigation icons: Home, Stop, Previous, Next, Volume

2.5 Tworzenie łatwych w zrozumieniu i czytaniu programów

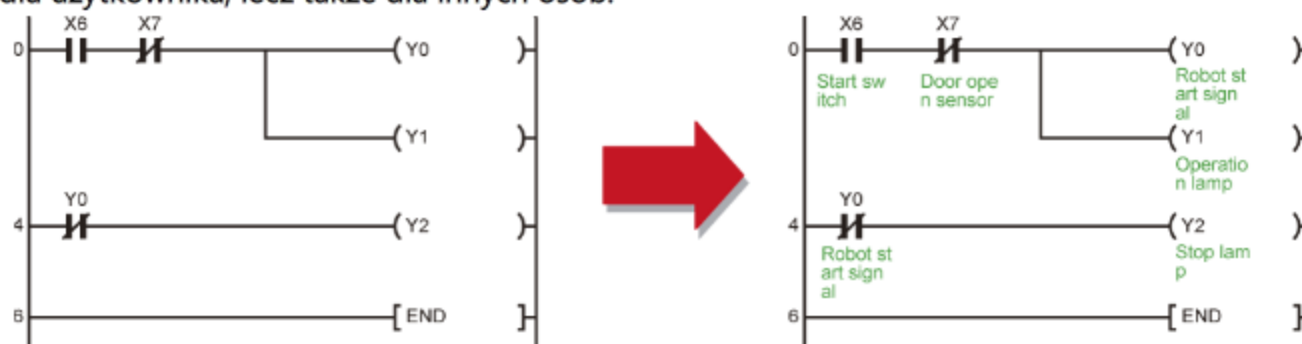
Zrozumienie szczegółów sterowania programem na dużą skalę poprzez spojrzenie na program może wydawać się trudne. W rezultacie tego mogą wystąpić następujące problemy:

- Wprowadzenie błędów programu (takich jak wprowadzenie nieprawidłowych instrukcji lub urządzeń).
- Brak możliwości znalezienia przyczyn błędów programu.
- Osoby przejmujące programowanie nie mogą zrozumieć szczegółów sterowania.

Należy skorzystać z opcji „**komentarz**”, aby rozwiązać te problemy.

Notatki, takie jak informacje dotyczące sterowania i nazwy urządzenia można dołączyć do programu, dzięki czemu zrozumienie szczegółów sterowania staje się łatwiejsze.

Komentarze te powinny być wprowadzane tak często, jak to możliwe, aby utworzyć programy, które są zrozumiałe nie tylko dla użytkownika, lecz także dla innych osób.



GX Works2 umożliwia wprowadzanie następujących komentarzy.

| Typ komentarza | Zakres komentarza |
|----------------|---|
| Device comment | Komentarz można dołączyć do urządzenia. Komentarz ten określa zastosowanie każdego urządzenia i rodzaj podłączonych urządzeń wejść/wyjść. |
| Statement | Komentarz można dołączyć do bloku drabinki. Komentarz sprawia, że przebieg przetwarzania jest łatwy w zrozumieniu. |
| Note | Komentarz można dołączyć do cewki/instrukcji zastosowania w programie. Komentarz sprawia, że treść cewki (wyjścia) i instrukcji zastosowania jest łatwa w zrozumieniu. |

2.5.1

Zapisywanie komentarza dla każdego bloku drabinki

Produktywność

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Write MAIN 145 Step]

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



Navigation

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Types
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

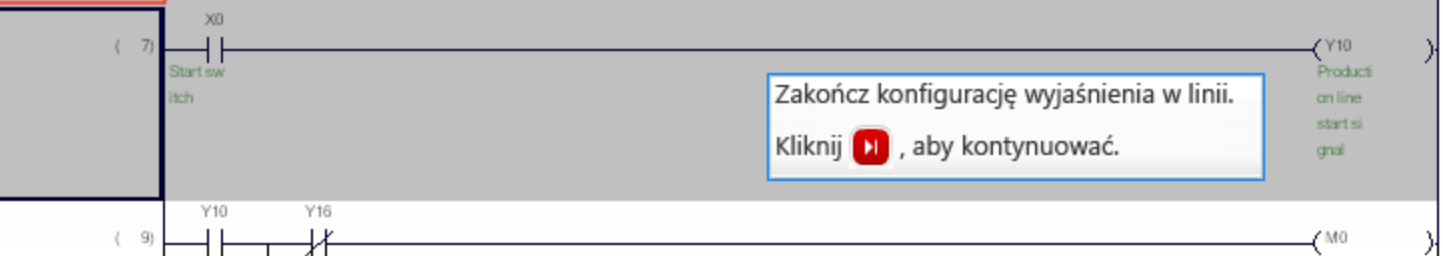
[PRG]Write MAIN 145 Step

Scheduled prod qty setting




Wyjaśnienie w linii zostało wprowadzone na początku bloku drabinki.

Prod line start



Zakończ konfigurację wyjaśnienia w linii.

Kliknij  , aby kontynuować.

English

Simple

Q03UDE

Host Station

(7/ NUS



2.5.2

Zapisywanie komentarza dla każdego wyjścia (cewka, instrukcja)

Produktywność

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 181 Step]


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

Navigation

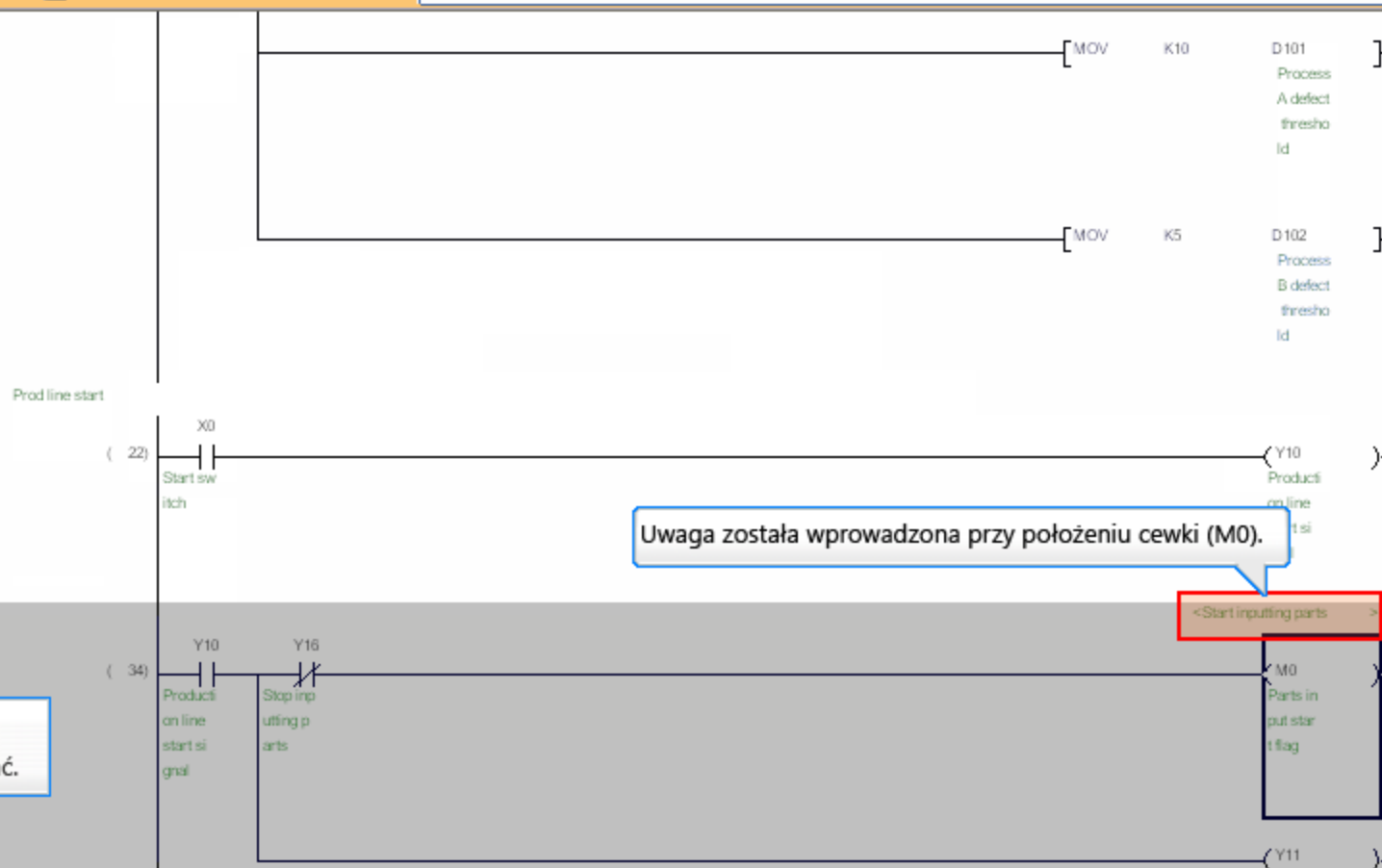
Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Types
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

Zakończ konfigurację uwagi.
Kliknij , aby kontynuować.

[PRG]Write MAIN 181 Step



English

Simple

Q03UDE

Host Station

(37 ML)



2.6

Ułatwienie odczytu programów zawierających flagi/rejestry specjalne

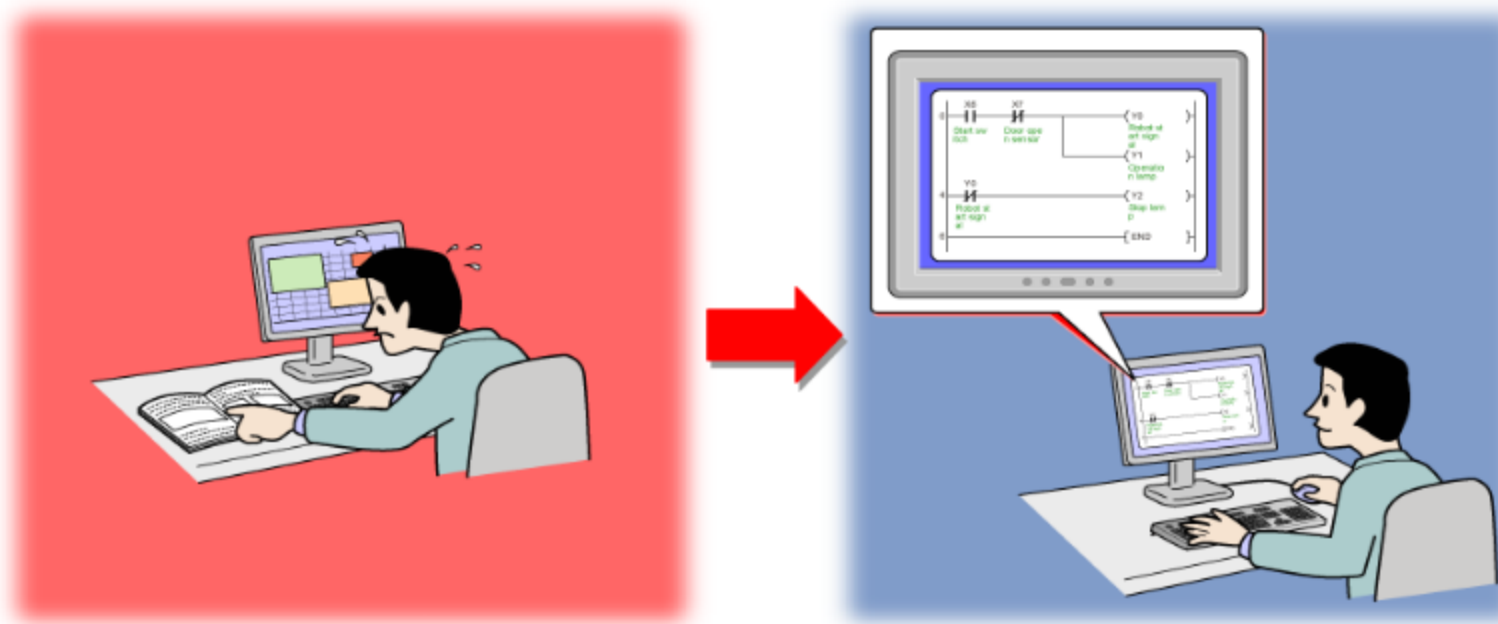
Jeśli urządzenia, takie jak flagi specjalne, rejestry specjalne i/lub moduły funkcji inteligentnych są stosowane w programie, zrozumienie wszystkich zastosowań i funkcji tych urządzeń może być trudne. Konieczne jest czytanie programu z instrukcją w rękę.

Pomimo że program byłby łatwiejszy w odczycie w przypadku dołączenia komentarza do każdego urządzenia, dołączenie komentarzy może wymagać znacznego czasu i wysiłku, jeśli jest stosowane wiele urządzeń.

Należy skorzystać z opcji „**Sample Comment**”, aby rozwiązać te problemy.

GX Works2 zapewnia przykładowe komentarze opisujące zastosowania i funkcje flag specjalnych, rejestrów specjalny i modułów funkcji inteligentnych.

Przy użyciu przykładowych komentarzy łatwiejsze jest dołączanie komentarzy do urządzeń, aby ułatwić odczyt programu. Przykładowe komentarze można w razie potrzeby modyfikować.



2.6

Ułatwienie odczytu programów zawierających flagi/rejestry specjalne

MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

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

Navigation [PRG]Write MAIN 194 Step Device Comment COMMENT

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Types
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination


English Simple Q03UDE Host Station (15 N...

(0) SM400 Always ON N [MOV K20 D100 Schedule d produc tion vol ume]

(22) X0 Start sw itch (Y10 Producti on line start si)

(34) Y10 Products on line start si Y16 Stop inp utting p arts put star t flag

Sprawdź program, aby sprawdzić, czy przykładowy komentarz „Always ON” (zawsze WŁ.) został zastosowany do SM400.

Zakończ automatyczną konfigurację przykładowego komentarza. Kliknij , aby kontynuować.

Rozdział 3 Wyszukiwanie i usuwanie błędów

Kroki procesu uczenia w Rozdziale 3

W rozdziale 3 przedstawione zostały funkcje stosowane w celu wyszukiwania i usuwania usterek. GX Works2 zapewnia wiele różnych narzędzi do monitorowania, wyszukiwania i usuwania usterek do poprawy błędów. Przy użyciu tych narzędzi do monitorowania, wyszukiwania i usuwania usterek możliwe jest tworzenie wysokiej jakości programów bez błędów.

- 3.1 Monitorowanie wybranych urządzeń i etykiet
- 3.2 Sprawdzanie statusu używania urządzeń i etykiet
- 3.3 Gromadzenie informacji o zmianie wartości urządzenia na przestrzeni czasu
- 3.4 Zmiana wartości urządzenia bez poprawy programu
- 3.5 Usuwanie błędów w działaniu programu krok po kroku
 - 3.5.1 Korzystanie z funkcji wykonania krokowego
- 3.6 Symulacja działania urządzenia zewnętrznego
 - 3.6.1 Wprowadzanie wartości urządzenia dla ustawienia systemu wejść/wyjść
 - 3.6.2 Korzystanie z formatu przebiegów czasowych dla ustawienia systemu wejść/wyjść

3.1 Monitorowanie wybranych urządzeń i etykiet

Jakość

Program stosuje wiele instrukcji i urządzeń.

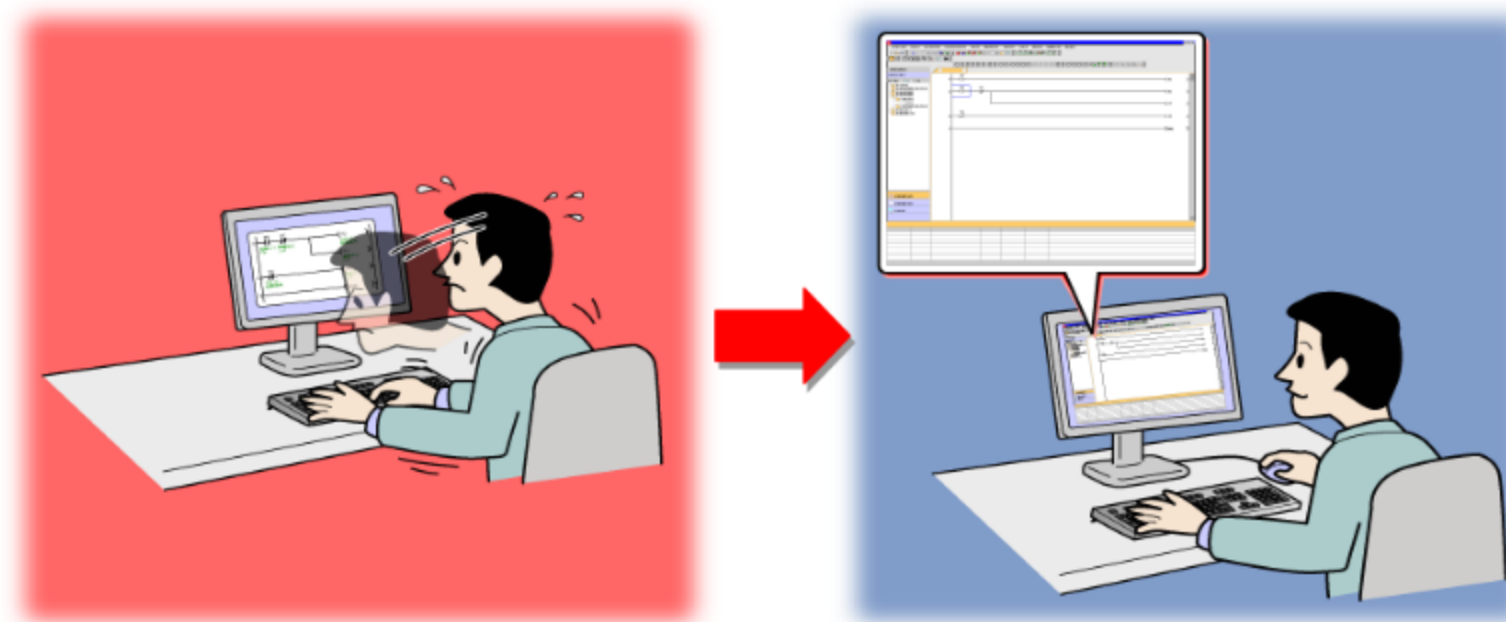
Ponadto długość pionowa programu oznacza, że tylko część jest wyświetlana w danym czasie na ekranie monitora komputera.

Z tego względu monitor drabinki nie może samodzielnie monitorować całej operacji.

Należy skorzystać z opcji „**Watch**”, aby rozwiązać te problemy.

Funkcja ta może być stosowana do monitorowania wybranych urządzeń i etykiet określonych wcześniej.

Istnieje możliwość utworzenia co najmniej dwóch obserwacji w celu zapewnienia, że każdy zakres jest monitorowany.



3.1

Monitorowanie wybranych urządzeń i etykiet

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

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

Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program

Prod qty calc

(74) M1

Start re
ject pct
calc

D1 D2 D6

GX Simulator 2

Tool Options

Switch

RESET RUN

LED

MODE


RUN

ERR.

USER

Watch 1 Monitorowane są obecne wartości urządzeń zarejestrowanych dla Watch 1.

| Device/Label | Current Value | Data Type | Class | Device | Comment |
|--------------|---------------|-----------|-------|--------|--------------------|
| X0 | 0 | Bit | | X0 | Start switch |
| X1 | 0 | Bit | | X1 | Parts input sensor |
| X2 | 0 | Bit | | X2 | |
| X3 | 0 | Bit | | X3 | |

Zakończ rejestrację Watch oraz monitorowanie.
Kliknij , aby kontynuować.

Watch 1 Watch 2

English Simple Q03UDE Simulation (84 N...

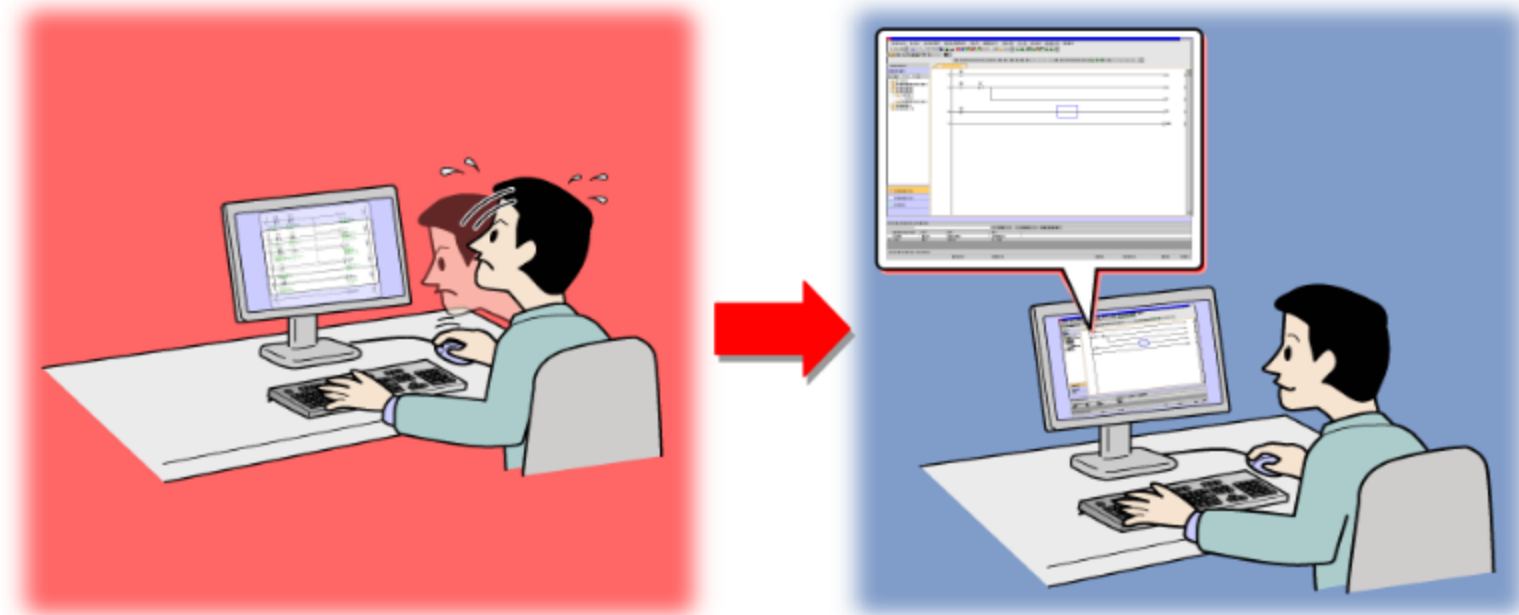
3.2 Sprawdzanie statusu używania urządzeń i etykiet

Jakość

Program korzysta z tych samych urządzeń i etykiet w różnych lokalizacjach. Istnieje możliwość sprawdzenia statusu tych urządzeń i etykiet poprzez ich porównanie pomiędzy lokalizacjami.

W tym celu należy skorzystać z „**Cross reference**”.

Funkcja ta wymienia listę lokalizacji urządzeń i etykiet pasujących do warunków wyszukiwania, dzięki czemu można je porównać i sprawdzić pod kątem nieprawidłowego zastosowania.



3.2 Sprawdzanie statusu używania urządzeń i etykiet



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Write MAIN 194 Step]

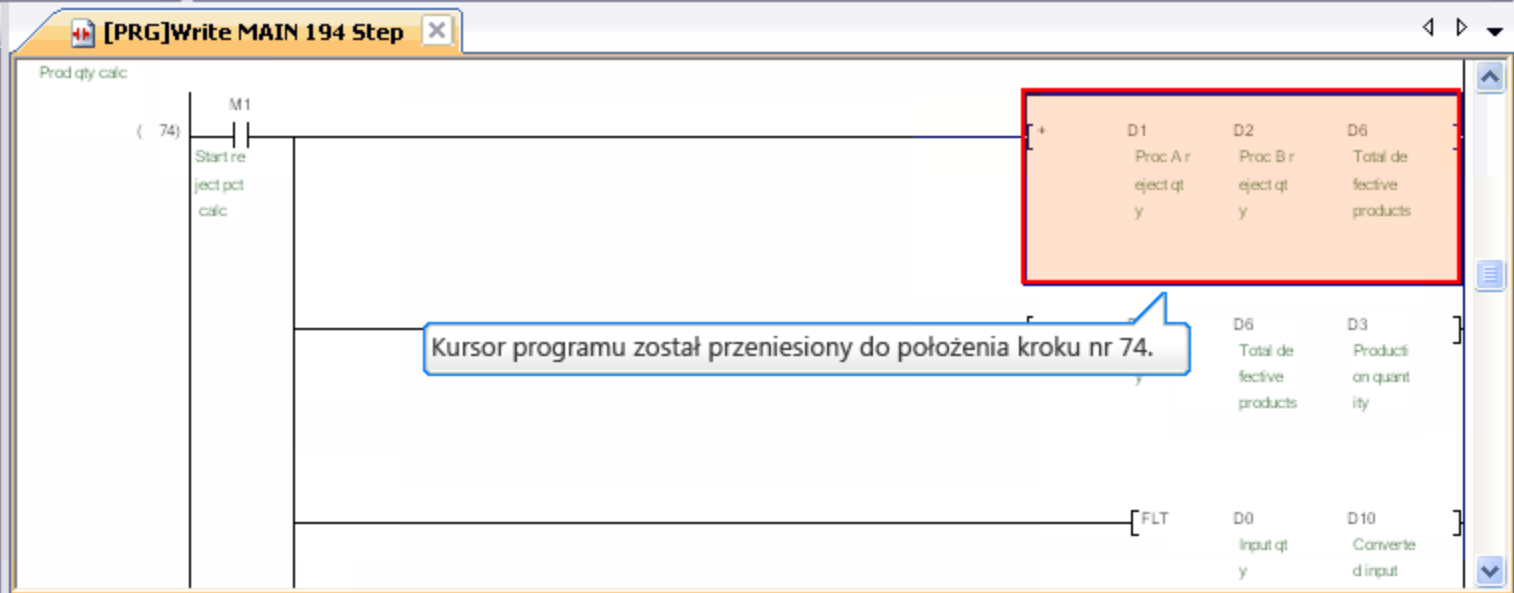
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help

Navigation icons: Save, Undo, Redo, Run, Stop, etc.

Navigation

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
- Program
 - MAIN



Kursor programu został przeniesiony do położenia kroku nr 74.

Cross Reference

Cross Reference Information | Condition Setting

Device/Label: D1

Find Print... Print Preview...

| Device/Label | Device | Instruction | Ladder Symbol | Position | Data Name |
|---------------------|------------------|---------------------|---------------------|------------|-----------|
| Filtering Condition | Filtering Con... | Filtering Condit... | Filtering Condition | | |
| D1 | D1 | +P | -[]- | Step No.63 | |
| D1 | D1 | + | -[]- | Step No.84 | |
| D1 | D1 | FLT | -[]- | Step No.92 | |

Zakończ korzystanie z opcji Cross Reference (odsylacza).
Kliknij [Play], aby kontynuować.

5: device/cross reference information of label "D1" Analyze and display current program after pressing Find.

Navigation icons: Home, Stop, Back, Forward, etc.

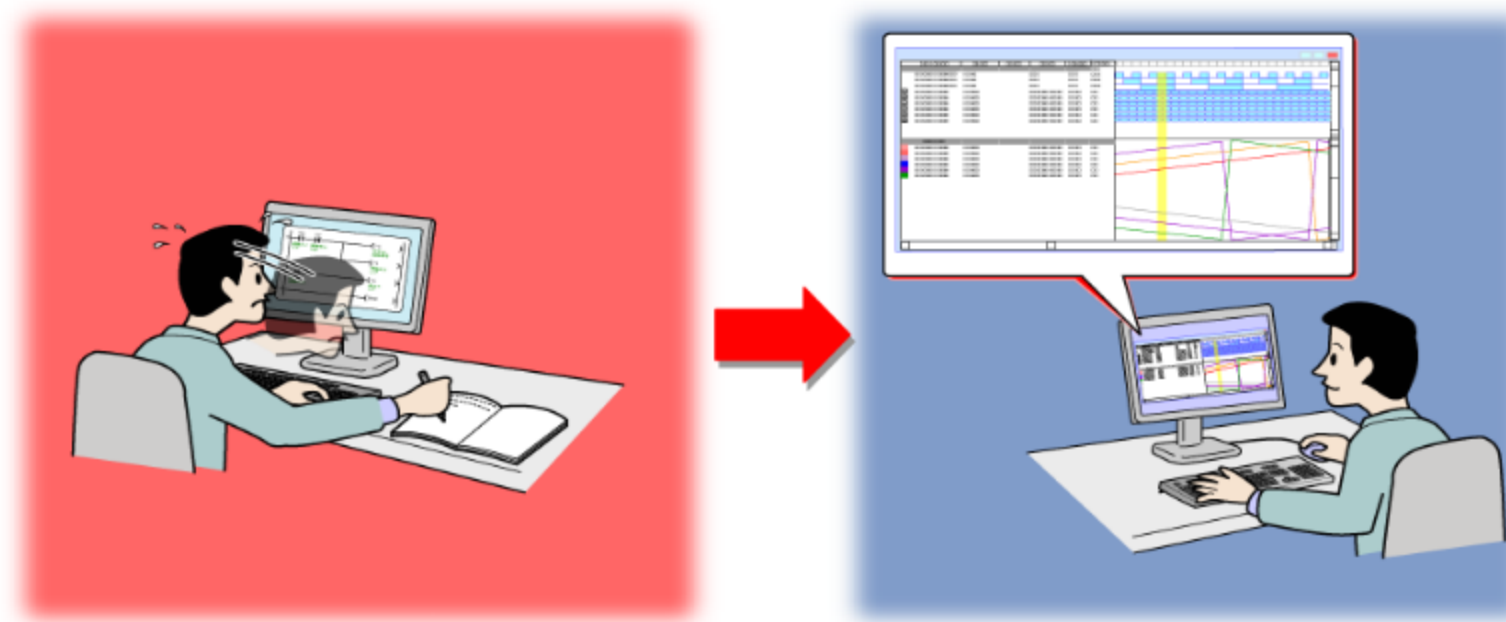
3.3

Gromadzenie informacji o zmianie wartości urządzenia na przestrzeni czasu

Jakość

Istnieje możliwość sprawdzenia, czy zmiany w wartościach urządzenia i etykiet znajdują się w projektowanym zakresie lub przeprowadzenia wyszukiwania i rozwiązywania usterek poprzez sprawdzenie zmian, które wystąpiły w przypadku usterki.

W tym przypadku należy użyć „**Sampling Trace**”. (Wyłącznie Seria MELSEC-Q i MELSEC-L)
Funkcja ta może być stosowana do monitorowania i zapisywania zmian w wartości urządzenia i etykiety na przestrzeni czasu. Rejestr można zapisać w plikach.



3.4

Zmiana wartości urządzenia bez poprawy programu

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

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



Navigation [PRG]Monitor Executing M... Sampling Trace

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool

GX Simulator2

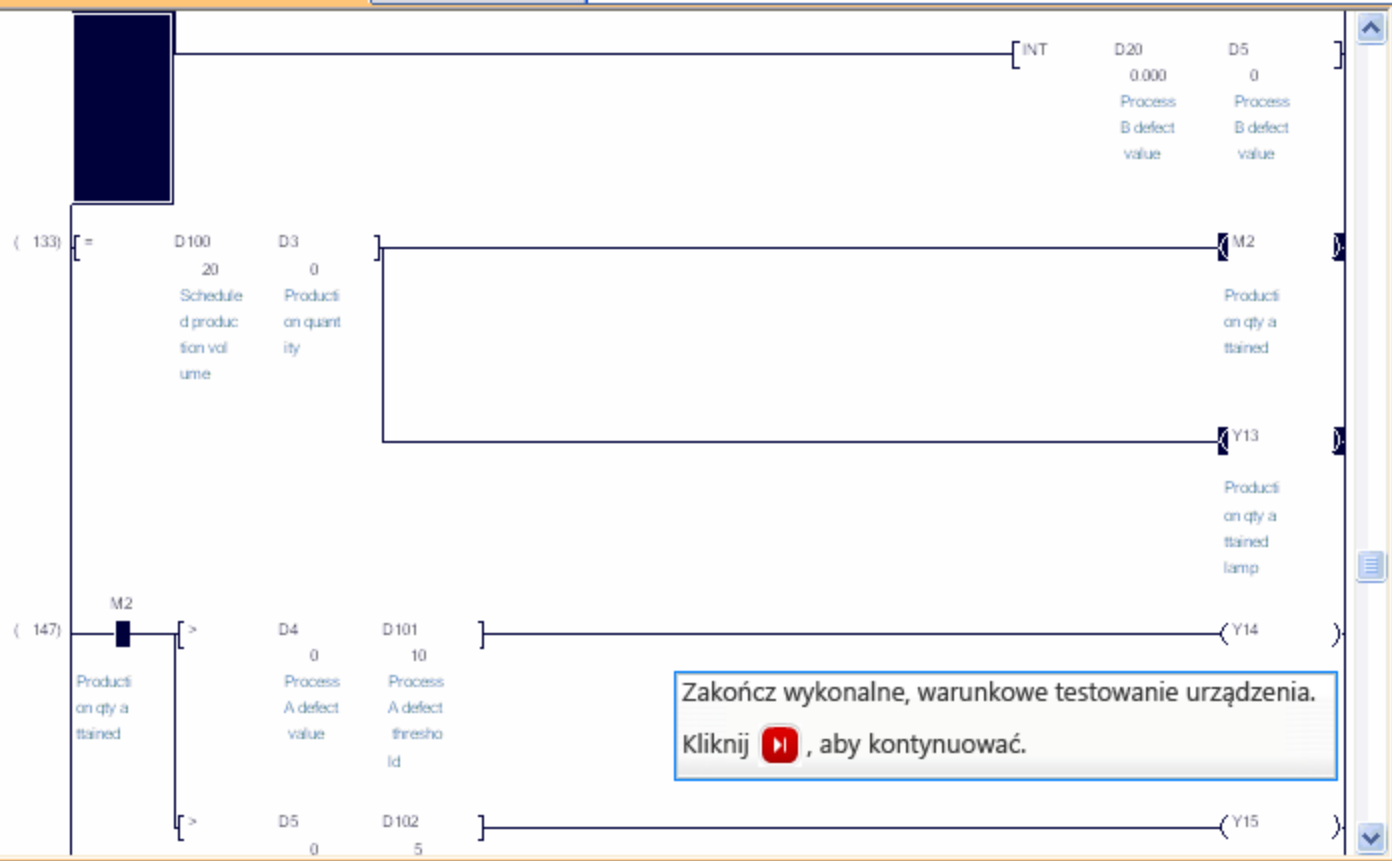
Tool Options


Switch

RESET STOP RUN

LED

| | |
|------|-------------------------------------|
| MODE | <input type="checkbox"/> |
| RUN | <input checked="" type="checkbox"/> |
| ERR. | <input type="checkbox"/> |
| USER | <input type="checkbox"/> |



Zakończ wykonalne, warunkowe testowanie urządzenia.
Kliknij , aby kontynuować.

English

Simple

Q03UDE

Simulation

(13 ML)



3.5

Usuwanie błędów w działaniu programu krok po kroku

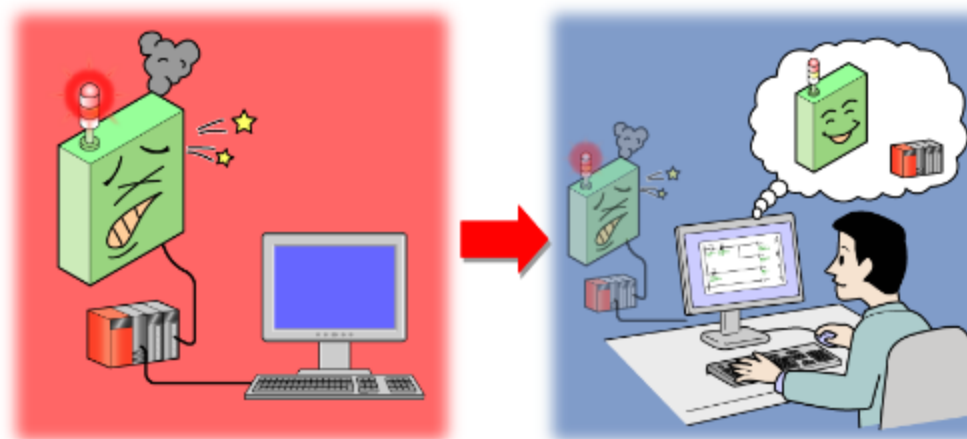
Jakość

Podczas wyszukiwania i usuwania usterek możliwe jest potwierdzenie wykonania instrukcji w każdym kroku lub sprawdzenia zmian w wartości urządzenia. Jednakże wyszukiwanie i usuwanie usterek krok po kroku może być trudne z powodu z dużej prędkości wykonania programu (czas skanowania).

Należy skorzystać z „**Step Execution Function**”, aby rozwiązać ten problem. (Wyłącznie Seria MELSEC-Q i MELSEC-L) Funkcja ta umożliwia wykonanie na raz jednego kroku programu i dzięki temu przeprowadzenie wyszukiwania i usuwania usterek krok po kroku. Należy skorzystać z funkcji wykonania kroku wraz z **funkcją symulacji**. (Funkcja wykonania kroku nie może być stosowana do wyszukiwania i usuwania usterek na faktycznym PLC.)

Następujące funkcje mogą być stosowane przez funkcję wykonania kroku.

| Funkcja | Opis |
|-----------------------|--|
| Przerywanie wykonania | Funkcja ta wykonuje program do czasu spełnienia określonych warunków przerwania. Wykonanie programu zostaje zatrzymane po spełnieniu warunku przerwania. Należy korzystać z punktu przerwania i urządzenia przerywającego w celu określenia warunków przerwania. |
| Wykonanie kroku | Funkcja wykonuje program krok po kroku. |
| Wykonanie częściowe | Funkcja wykonuje program wyłącznie do określonego miejsca. |



3.5.1

Korzystanie z funkcji wykonania krokowego

Jakość

Należy określić **punkt przzerwania** oraz **urządzenie przerywające** jako lokalizację rozpoczęcia wyszukiwania i usuwania usterek oraz warunek rozpoczęcia.

Można również określić **zakres pominięcia**, w ramach którego można tymczasowo uniknąć wykonania programu.
(Wyłącznie Seria MELSEC-Q i MELSEC-L)

Po spełnieniu warunków przzerwania po rozpoczęciu **przerwania wykonania** wykonanie programu zostanie przerwane. Następnie podczas wykonywania działania programu krok po kroku przy użyciu **funkcji wykonania kroku**, należy sprawdzić zmiany wartości urządzenia w celu znalezienia usterki.

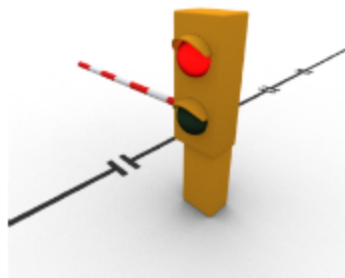
<Punkt przzerwania>

Należy ustawić punkt przzerwania, w którym ma nastąpić przerwanie wykonania programu. Określić go w jednostkach kroków. Dla całego projektu można ustawić do 64 punktów przzerwania.



<Urządzenie przerywające>

Należy ustawić urządzenie przerywające w oparciu o program, którego wykonanie zostanie przerwane, gdy wartość urządzenia lub etykiety spełni określony warunek. Można ustawić do 16 bitów i/lub słów danych.



<Zakres pominięcia>

Należy ustawić zakres, w ramach którego program nie powinien być wykonywany w jednostkach bloków krokowych, podczas wykonania kroku. Dla całego projektu można ustawić do 16 zakresów.



<Zastosowanie zakresu pominięcia>

Istnieje możliwość zawężenia punktu usterki przy użyciu **zakresu pominięcia**.

Należy przeprowadzić przerwanie wykonania z i bez ustawienia zakresu pominięcia.

W przypadku wystąpienia usterki wyłącznie po ustawieniu zakresu pominięcia zakres ten obejmuje usterkę.

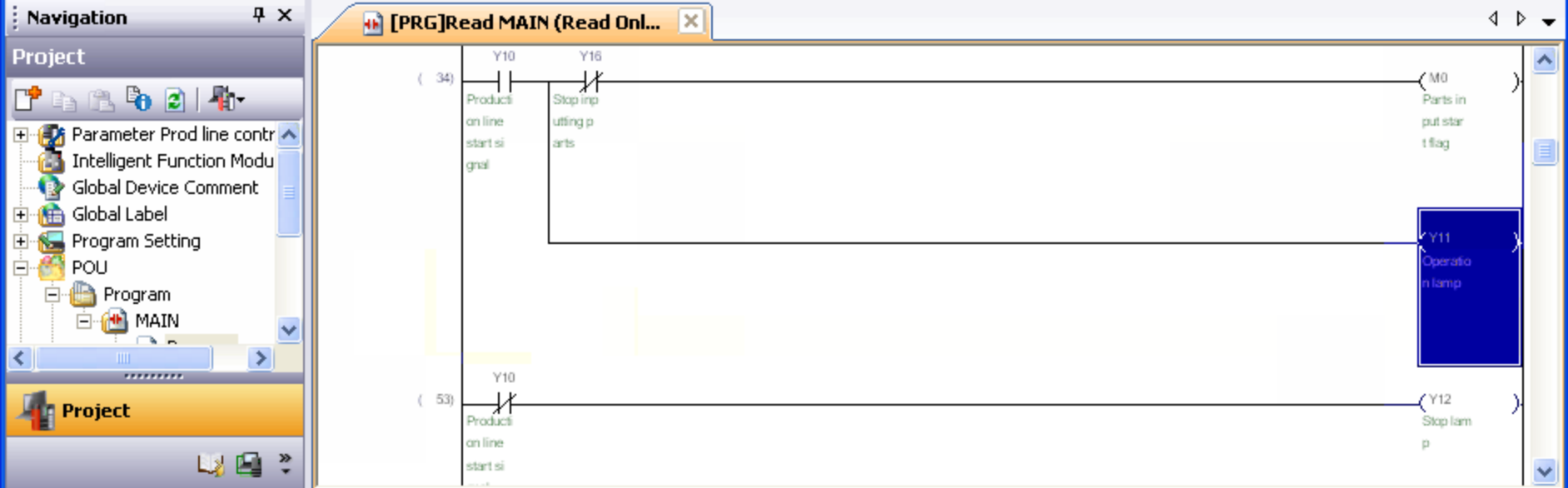
3.5.1

Korzystanie z funkcji wykonania krokowego

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Read MAIN (Read Only) 194 Step]

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




Break Device

New Condition Cancel Cancel All

Combination

Judge each break device (OR condition) Judge all break devices (AND condition)

| Enable/Disable | Comparative Source (Device/Label) | Condition | Comparative Target (Value/Device/Label) | Comparative Type |
|----------------|-----------------------------------|-----------|---|------------------|
| | | | | |

Wyszukiwanie i usuwanie usterek zostało zakończone. Zakończ funkcję wykonania kroku.
Kliknij , aby kontynuować.

Skip Range Break Point Break Device

English

Simple

Q03UDE

Host Station

(52 N)



3.6 Symulacja działania urządzenia zewnętrznego

Jakość

Wyszukiwanie i usuwanie usterek w połączeniu z działaniem urządzenia zewnętrznego nie jest możliwe w środowisku programistycznym, w którym urządzenie zewnętrzne nie może być stosowane, w takim jak funkcja symulacji. Aby rozwiązać ten problem został umownie dodany program do wyszukiwania i usuwania usterek, który symuluje działanie urządzenia zewnętrznego.

Utworzenie programu symulacji wymaga czasu i wysiłku, lecz także konieczne jest zmodyfikowanie programu w przypadku zmiany działania.

Aby rozwiązać ten problem, należy użyć „**I/O System Setting**”.

Funkcja ta może symulować działanie urządzenia zewnętrznego bez stosowania programu do wyszukiwania i usuwania usterek. Działanie urządzenia zewnętrznego można łatwo ustawić lub zmienić w oknie ustawień. Z tego względu nie jest potrzebne tworzenie/zmiana konwencjonalnego programu.

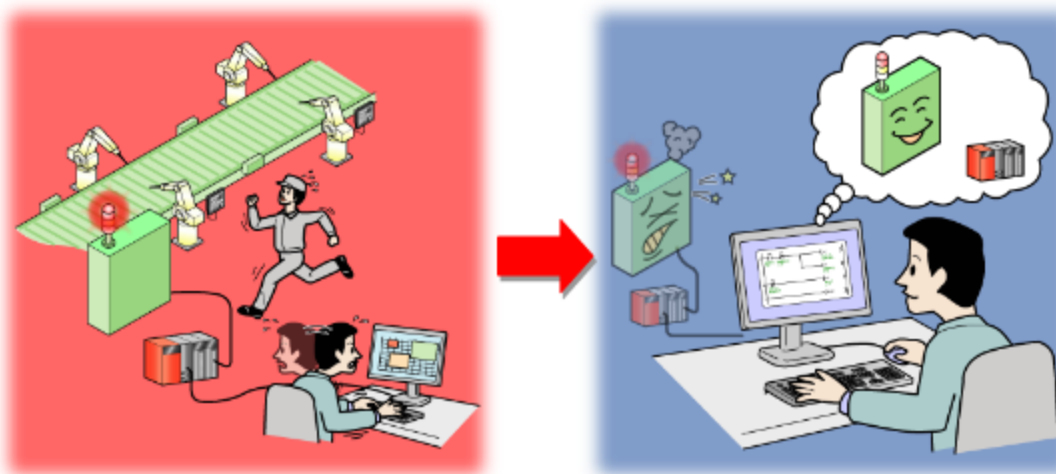
Działanie urządzenia zewnętrznego można ustawić na dwa sposoby:

<Ustawienie wartości urządzenia>

Określone wartości urządzenia można zmienić w czasie nastawy timera po spełnieniu określonych warunków.

<Ustawienie w formacie przebiegu czasowego>

Zmiana ustawienia skonfigurowana w określonym formacie przebiegu czasowego może zajść po spełnieniu określonych warunków.



3.6.1

Wprowadzanie wartości urządzenia dla ustawienia systemu wejść/wyjść

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step

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

Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Type

Scheduled prod qty setting

(0) SM400

Always ON


[MOV K20 D100 20 Scheduled production volume]

[MOV K10 D101 10 Process A defect threshold]

[MOV K5 D102 5 Process B defect threshold]

(22) X0 Start switch

start signal

Zakończ ustawienia oraz wykonanie wartości urządzenia.
Kliknij  , aby kontynuować.

ish Simple Q03UDE Simulation (0/ N...

3.6.2

Korzystanie z formatu przebiegów czasowych dla ustawienia systemu wejść/wyjść

Jakość

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Monitor Executing MAIN (Read Only) 194 Step]

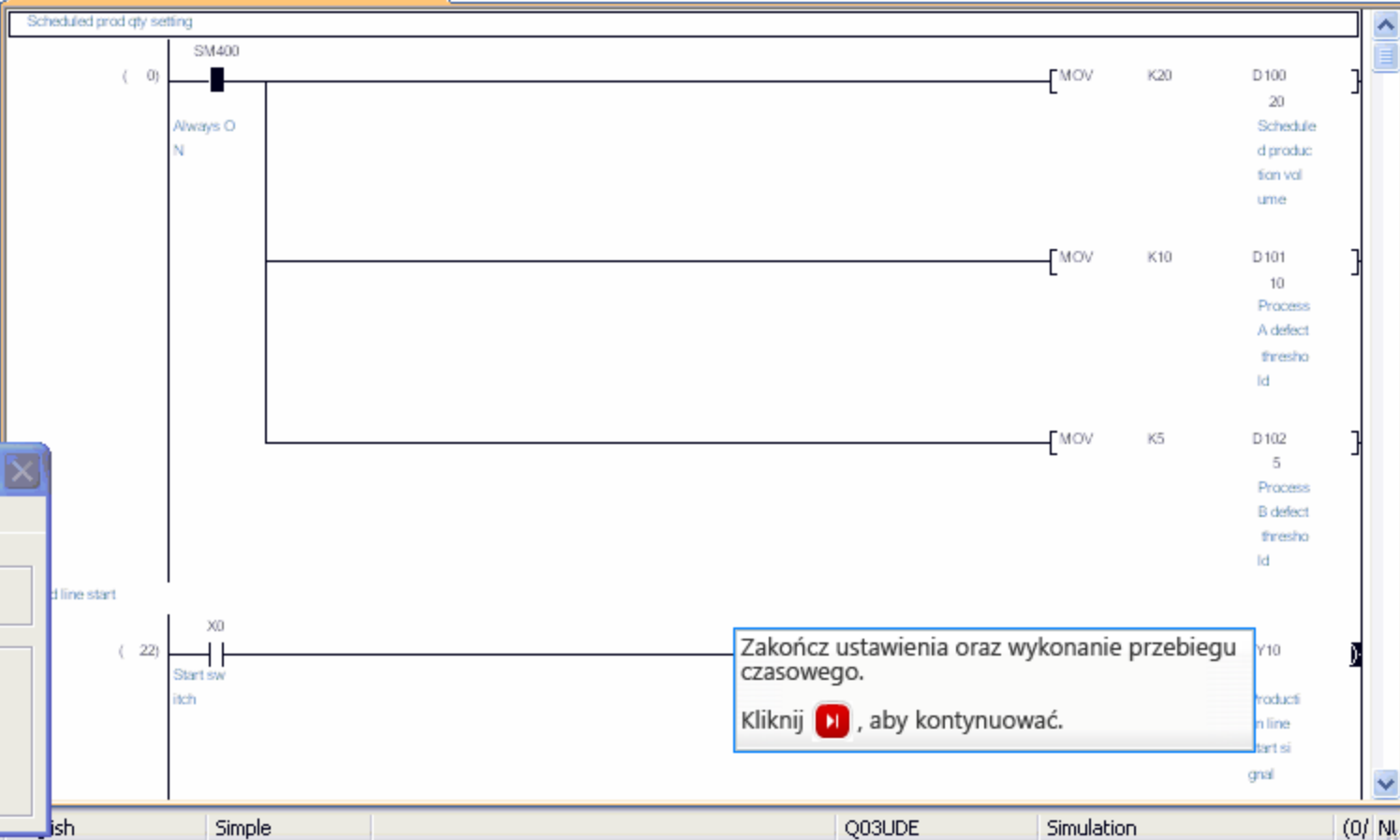
Project Edit Find/Replace Compile View Online Debug Diagnostics Tool Window Help




Navigation [PRG]Monitor Executing M...

Project

- Parameter Prod line contr
- Intelligent Function Modu
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Type



Zakończ ustawienia oraz wykonanie przebiegu czasowego.

Kliknij , aby kontynuować.

GX Simulator 2

Tool Options

Switch

RESET RUN

LED

| | |
|------|--|
| MODE | |
| RUN | |
| ERR. | |
| USER | |



Rozdział 4 Zarządzanie projektem i środki bezpieczeństwa

Kroki procesu uczenia w Rozdziale 4

W rozdziale 4 przedstawione zostały funkcje stosowane w celu zarządzania projektem oraz środki bezpieczeństwa.

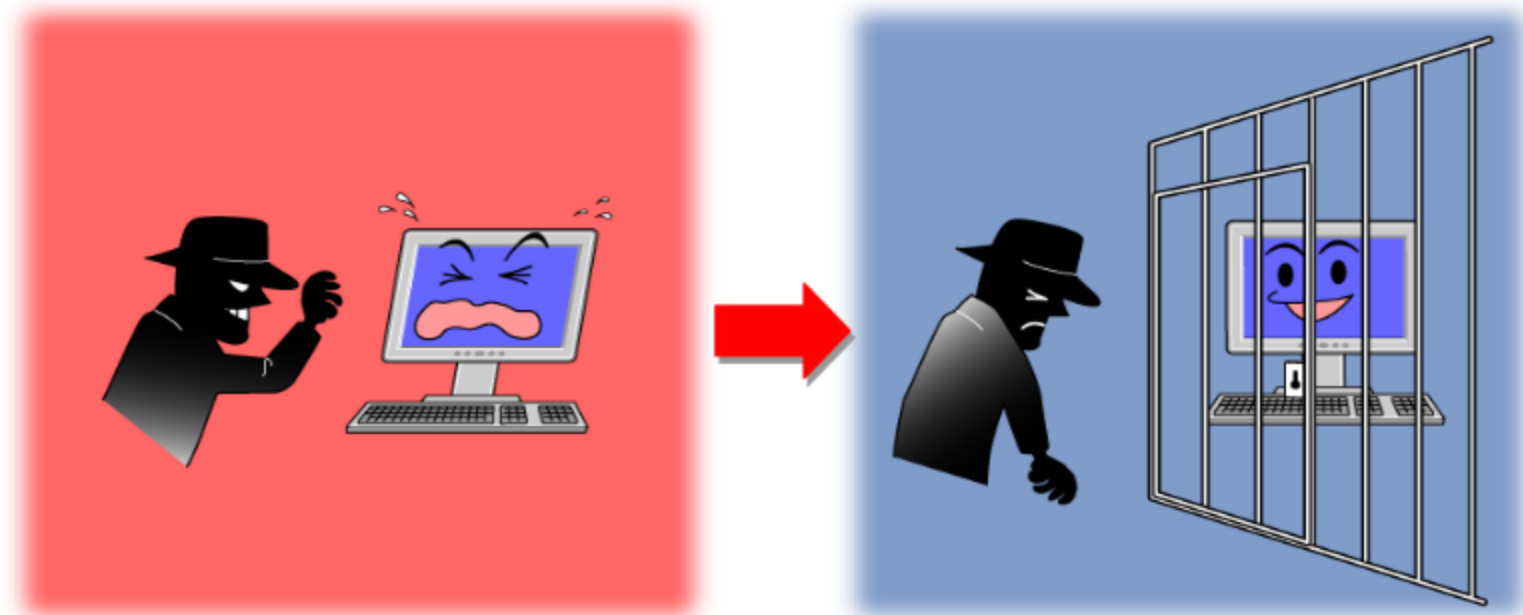
- 4.1 Zapobieganie wyciekowi know-how i nieupoważnionym modyfikacjom programów
 - 4.1.1 Ograniczenie danych dostępnych dla każdego użytkownika
- 4.2 Zarządzanie kopią zapasową projektu oraz wersją
- 4.3 Porównywanie programów zapisanych w sterowniku programowalnym i komputerze

4.1 Zapobieganie wyciekowi know-how i nieupoważnionym modyfikacjom programów



Program sekwencyjny obejmuje ważne pod względem strategicznym dane oraz know-how. Wyciek know-how oraz danych z programu na zewnątrz może mieć niszczący wpływ na firmę. Nieupoważnione modyfikacje programu mogą prowadzić do problemów podczas produkcji, takich jak zatrzymanie systemu.

Należy skorzystać z opcji „**bezpieczeństwo**”, aby rozwiązać te problemy. Funkcja ta ogranicza liczbę użytkowników, którzy mają dostęp do każdego projektu, zabezpieczając go hasłem. Może również ograniczyć zakres danych lub funkcji, do których ma dostęp lub może obsługiwać każdy użytkownik. Funkcja ta zapobiega przeglądaniu i edycji programów przez nieupoważnionych użytkowników.



4.1.1

Ograniczenie danych dostępnych dla każdego użytkownika



Program sekwencyjny na dużą skalę jest często opracowywany przez dwóch lub więcej programistów dzielących się pracą. W przypadku opracowywania przez zespół zakres dostępnych danych oraz dostępne funkcje muszą być odpowiednio zarządzane zgodnie z zakresem prac i poziomem umiejętności każdego programisty oraz poufnością danych obsługiwanych przez programistów.

Zarządzanie dostępem można wdrożyć poprzez ustawienie zabezpieczających **poziomów dostępu**.

<Poziom dostępu>

Można ustawić przywileje działania wobec danych zawartych w każdym projekcie dla każdego użytkownika.

Można ustawić poniższe pięć poziomów dostępu.

| Poziom dostępu | | Uprawnienia dot. Działania | |
|-----------------------------------|------------------------|----------------------------|---|
| Wysoki Niski | Administratorzy | Poziom administratora | Upoważniony do stosowania wszystkich funkcji. |
| | Programiści (poziom 3) | Poziom programisty | Ustawienia bezpieczeństwa, dostęp do danych i niektóre operacje są ograniczone. |
| | Programiści (poziom 2) | | |
| | Programiści (poziom 1) | | |
| | Użytkownicy | Poziom operatora | Dostępne jest tylko przeglądanie danych. Nie można odczytać z CPU PLC. |

4.1.1

Ograniczenie danych dostępnych dla każdego użytkownika



MELSOFT Series GX Works2 C:\Sequential Programs\learning\Robot control - [[PRG]Read MAIN (Read Only) 194 Step]

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



Navigation

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
 - FB_Pool
 - Structured Data Types
 - Local Device Comment
- Device Memory
- Device Initial Value

Project

User Library

Connection Destination

English Simple User Q03UDE Host Station (15 N...

[PRG]Read MAIN (Read On...

Scheduled prod qty setting



Zakończ Security Setting (ustawienia bezpieczeństwa).

Kliknij , aby kontynuować.



4.2

Zarządzanie kopią zapasową projektu oraz wersją



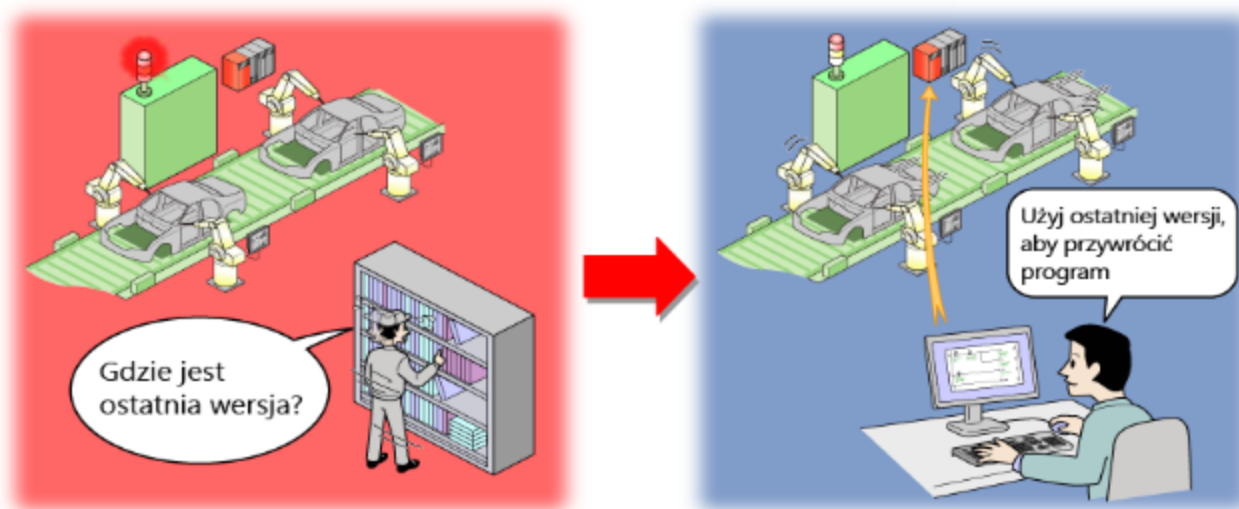
Istnieje ryzyko utraty ważnego programu z powodu usterki CPU sterownika programowalnego. Jeśli program zapasowy nie jest dostępny, istnieje niewielka szansa na szybki powrót do normalnego stanu. Nawet w przypadku dostępności programu zapasowego, trudne może być określenie, czy wersja jest taka sama, jak wersja utraconego programu, chyba że zapewnione jest zarządzanie wersją. Aby przygotować się na nieprzewidywalne okoliczności, konieczne jest wykonywanie okresowych kopii zapasowych i zapewnienie zarządzania wersją.

Należy skorzystać z opcji „**historia zmian**”, aby rozwiązać te problemy.

Funkcja ta może zarejestrować do 100 ustawień historii zmian (numer historii, data/godzina, użytkownik, tytuł, komentarz) projektu.

Dane projektu w momencie zapisywania są również jednocześnie uwzględnione w kopii zapasowej.

Zarządzanie wersją zapewnione przez funkcję historii zmian umożliwia przywrócenie utraconych programów, sprawdzenie wersji programów i wdrożenie szybkiego powrotu do normalnego stanu w przypadku nieoczekiwanych okoliczności.



4.2

Zarządzanie kopią zapasową projektu oraz wersją



MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control (Revision No.3 : Revision B) - [[PRG]Write MAIN 194 Step]

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



Navigation

Project

- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
- FB_Pool
- Structured Data Types
- Local Device Comment
- Device Memory
- Device Initial Value

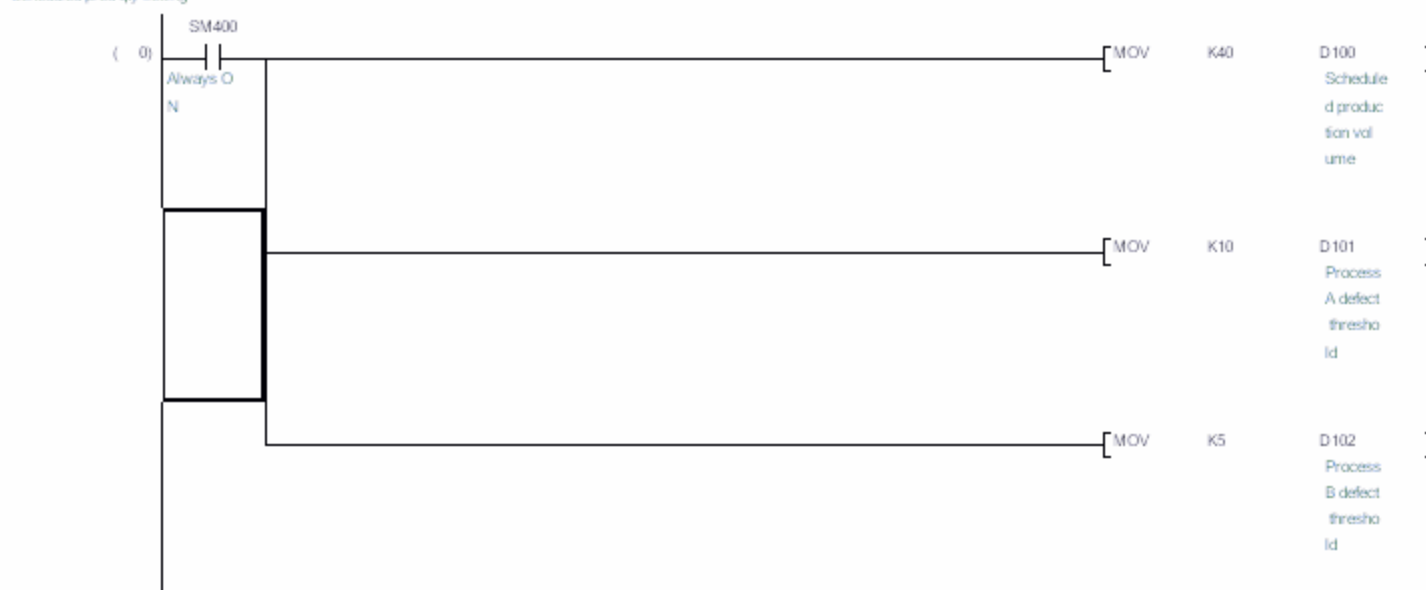
Project

User Library

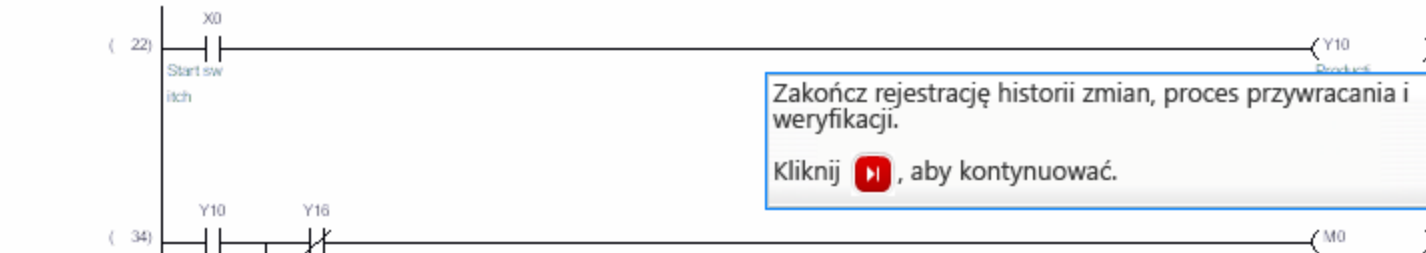
Connection Destination

[PRG]Write MAIN 194 Step

Scheduled prod qty setting



Prod line start



Zakończ rejestrację historii zmian, proces przywracania i weryfikacji.

Kliknij , aby kontynuować.

English

Simple

Q03UDE

Host Station

(18 M...

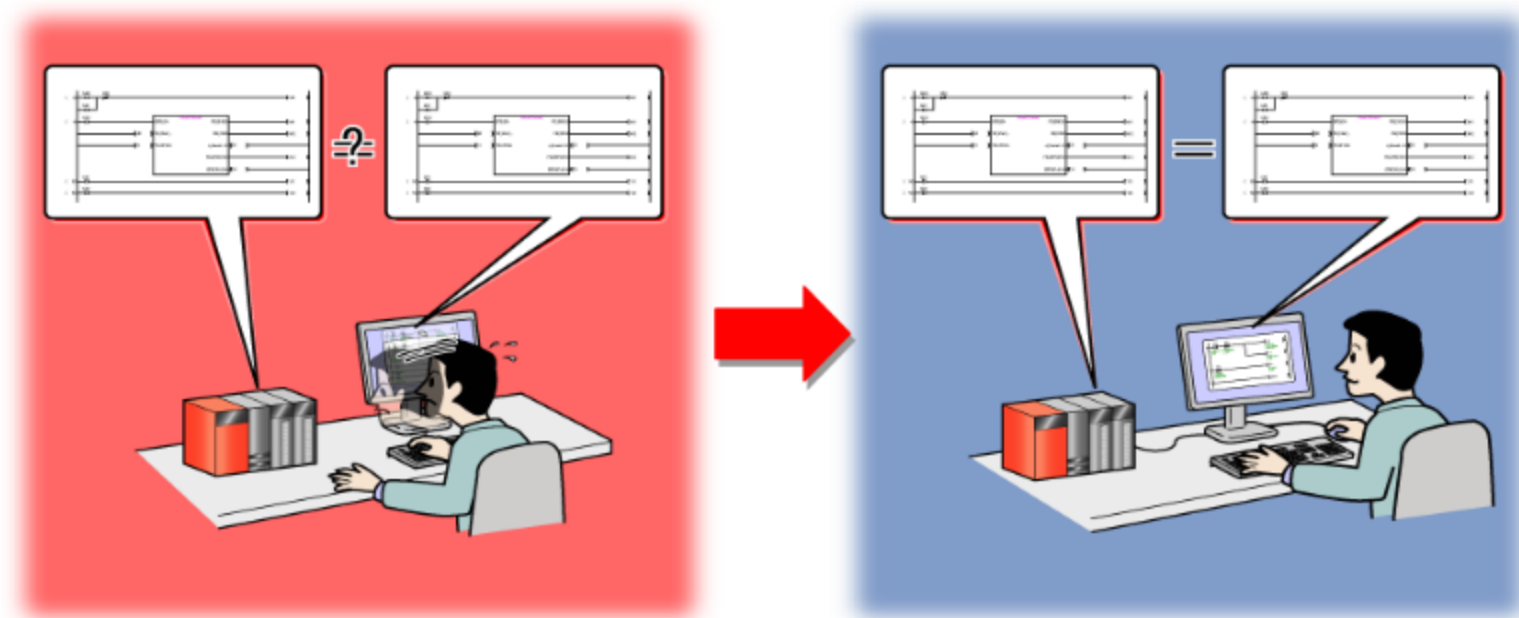


4.3 Porównywanie programów zapisanych w sterowniku programowalnym i komputerze



Zazwyczaj programy są zapisywane w komputerze w środowisku programistycznym i są również zapisywane w PLC. Te dwa programy nie zawsze są takie same. Przeprowadzenie kontroli wzrokowej w celu sprawdzenia, czy są takie same może skutkować błędami.

Aby rozwiązać ten problem, należy użyć **„Verify with PLC”**. Funkcja ta może sprawdzić, czy program otwarty przez GX Works2 pasuje do programu zapisanego w PLC.



4.3 Porównywanie programów zapisanych w sterowniku programowalnym i komputerze

Zarządzanie

MELSOFT Series GX Works2 C:\Sequential Programs\le-learning\Robot control - [[PRG]Write MAIN 194 Step]

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



Navigation [PRG]Write MAIN 194 Step Verify Result [PLC Verify]

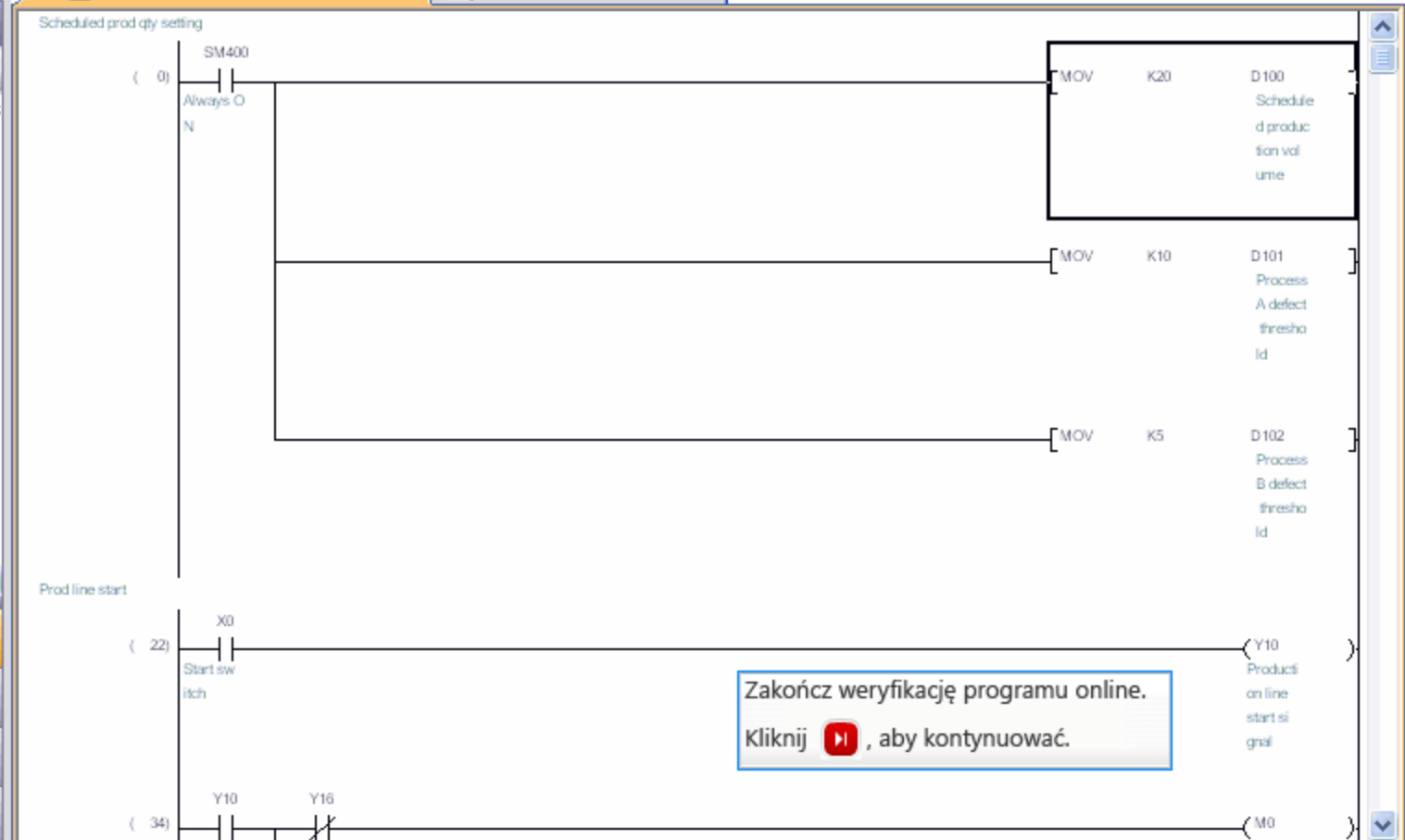
Project


- Parameter Prod line control s
- Intelligent Function Module
- Global Device Comment
- Global Label
- Program Setting
- POU
 - Program
 - MAIN
 - Program
 - Local Label
- FB_Pool
- Structured Data Types
- Local Device Comment

- Device Memory
- Device Initial Value

Project

- User Library
- Connection Destination



Zakończ weryfikację programu online.
Kliknij , aby kontynuować.

English

Simple

Q03UDE

Host Station

(16 N...



Test Test końcowy

Po zakończeniu wszystkich etapów kursu **PLC GX Works2 Advanced** możesz teraz przystąpić do testu końcowego. W razie niejasności w zakresie któregoś z tematów, wykorzystaj tę możliwość do ponownego zapoznania się z tymi zagadnieniami.

Test końcowy składa się z 8 pytań (8 elementów).

Możesz zdawać test końcowy dowolną ilość razy.

Jak rozwiązywać test

Po wybraniu odpowiedzi naciśnij koniecznie przycisk **Odpowiedz**. Twoja odpowiedź zostanie utracona, jeśli będziesz kontynuować bez kliknięcia przycisku Odpowiedz. (Zostanie potraktowana jako pytanie, na które nie udzielono odpowiedzi.)

Punktacja końcowa

Liczba prawidłowych odpowiedzi, liczba pytań, procent prawidłowych odpowiedzi i wynik zaliczony/niezaliczony pojawią się na stronie wyniku.

Prawidłowe odpowiedzi: 4

Wszystkie pytania: 4

Procent prawidłowych odpowiedzi: 100%

Aby zaliczyć test, musisz odpowiedzieć poprawnie na **60%** pytań.

Kontynuuj

Przeglądaj

- Kliknij przycisk **Kontynuuj**, aby zakończyć test.
- Kliknij przycisk **Przeglądaj**, aby przeglądać test. (Sprawdzanie prawidłowych odpowiedzi)
- Kliknij przycisk **Spróbuj**, aby powtórzyć test.

Test**Test końcowy 1**

Która z poniższych funkcji umożliwia skuteczne programowanie przez stosowanie powtarzalnych bloków drabinki jako udostępnianych elementów? (wybierz jedną odpowiedź)

- Wbudowany język ST
- Etykieta
- Blok funkcyjny

Test**Test końcowy 2**

Przy użyciu której z poniższych funkcji można utworzyć łatwe w odczycie programy poprzez zmianę nazw urządzeń na nazwy powiązane z zastosowaniem? (wybierz jedną odpowiedź)

- Komentarz do urządzenia
- Etykieta
- Uwaga

Test**Test końcowy 3**

Przy użyciu której z poniższych funkcji można utworzyć łatwe w odczycie programy poprzez podanie informacji dotyczących przetwarzania każdego bloku drabinki? (wybierz jedną odpowiedź)

- Komentarz do urządzenia
- Wyjaśnienie w linii
- Uwaga

Test

Test końcowy 4

Które z poniższych jest prawidłowym wyjaśnieniem funkcji „Verify with PLC”? (wybierz jedną odpowiedź)

- Porównuje program edytowany z programem zapisanym w historii zmian.
- Porównuje program edytowany z wybranym programem zapisanym w komputerze.
- Porównuje program edytowany z programem zapisanym w CPU PLC.

Odpowiedź

Wstecz

Test

Test końcowy 5

Które z poniższych jest prawidłowym wyjaśnieniem funkcji „I/O System Setting”? (wybierz jedną odpowiedź)

- Symuluje działanie zewnętrznego urządzenia wejść/wyjść w komputerze podczas wyszukiwania i usuwania błędów.
- Zdalnie steruje działaniem zewnętrznego urządzenia we/wy z komputera podczas wyszukiwania i usuwania błędów.
- Symuluje działanie CPU PLC w komputerze podczas wyszukiwania i usuwania błędów.

Odpowiedź

Wstecz

Test

Test końcowy 6

Które z poniższych jest prawidłowym wyjaśnieniem funkcji „historia zmian”? (wybierz jedną odpowiedź)

- Zapisuje działanie GX Works2 krok po kroku, aby zapewnić łatwe odtworzenie w przyszłości.
- Zapisuje informacje dotyczące zmian i kopie zapasowe projektu, aby ułatwić weryfikację i późniejsze przywrócenie.

Odpowiedź

Wstecz

Test**Test końcowy 7**

Której z poniższych funkcji można używać podczas wyszukiwania i usuwania błędów, aby zmienić tylko wartości urządzenia bez zmiany programu? (wybierz jedną odpowiedź)

- Przerwanie wykonania
- Ustawienie systemu wejść/wyjść
- Wykonalne, warunkowe testowanie urządzenia

Test**Test końcowy 8**

Założmy, że projekt obejmuje dwa programy, A i B, i korzystamy z funkcji „etykieta”.

Do której z poniższych rodzajów etykiet można uzyskać dostęp przez program B? (wybierz dwie odpowiedzi)

- Etykieta globalna
- Etykieta lokalna programu A
- Etykieta lokalna programu B

Odpowiedź

Wstecz

Test**Wynik testu**

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

Prawidłowe odpowiedzi: **8**

Wszystkie pytania: **8**

Procent prawidłowych odpowiedzi: **100%**

[Kontynuuj](#)[Przełóżaj](#)

Gratulacje. Test został zaliczony.

Kurs **PLC GX Works2 Advanced** został ukończony.

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.

Przeglądaj

Zamknij