



**AKADEMIA GÓRNICZO-
HUTNICZA
IM. STANISŁAWA STASZICA
W KRAKOWIE**

**LABORATORIUM PRZEMYSŁOWYCH
SYSTEMÓW STEROWANIA**



**Wydział Inżynierii Mechanicznej i
Robotyki**



Katedra Automatykacji Procesów

Przedmiot:

Przemysłowe Systemy Sterowania (PSS)

Laboratorium 08:

Programowanie sterowników s7-1200



Kraków

*Wszelkie prawa zastrzeżone dla KAP, WIMiR, AGH
Jakiegokolwiek kopiowanie materiałów (w tym na potrzeby prac dyplomowych) bez zgody autorów niedozwolone.*

Cel ćwiczeń laboratoryjnych

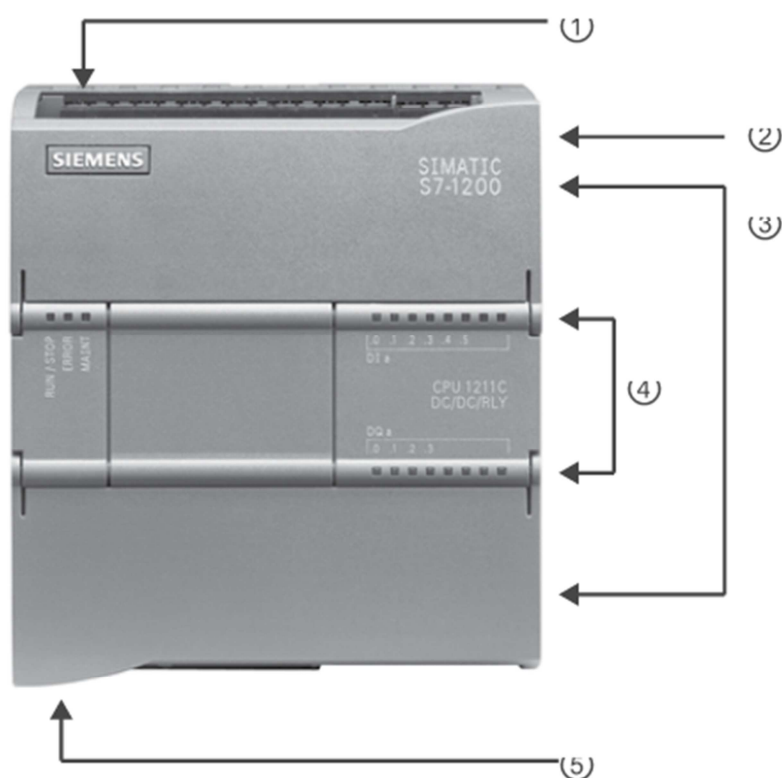
1. Zapoznanie z budową i zasadą działania sterowników s7-1214C.
2. Poznanie podstawowych zasad konfiguracji sterowników s7-1200.
3. Realizacja sterowania fizycznymi wejściami i wyjściami za pomocą języków LAD oraz FBD

Po ukończeniu zajęć student powinien potrafić:

1. Samodzielnie skonfigurować i skomunikować się z s7-1200.
2. Potrafić obsłużyć i zaprogramować zatrzaski w językach LAD oraz FBD.
3. Zrealizować układy wykorzystujące zbrocza styków.

Platforma Siemens Simatic S7-1200

Simatic S7-1200 jest rodziną Programowalnych Sterowników Logicznych (ang. *Programmable Logic Controller* – PLC) produkowanych przez firmę Siemens. Jednostka centralna zawiera między innymi mikroprocesor, wbudowany zasilacz, port PROFINET, obwody wejść i wyjść oraz szybkie wejścia/wyjścia. Do programowania kontrolerów z serii Simatic służy dedykowane środowisko o nazwie TIA Portal. Dodatkowo sterowniki z tej rodziny charakteryzują się kompaktową budową oraz obsługą wielu modułów, co umożliwia dostosowanie ich do zadań prostych oraz rozbudowanych.



Rys. 1. Budowa jednostki centralnej z rodziny S7-1200, 1) Złącze zasilające, 2) Gniazdo karty pamięci pod górną klapką, 3) Wejścia i wyjścia zabezpieczone klapką, 4) Diody LED statusu wejść/wyjść, 5) Złącze PROFINET

Laboratorium wyposażone jest w 8 sterowników S7-1214 C DC/DC/DC. W lewym górnym rogu (Rys. 1) znajdują się przyłącza zasilania sterownika. W prawym górnym rogu pod klapką znajduje się gniazdo karty pamięci. Karta pamięci przechowuje program i może być wykorzystana do przenoszenia programu do innych sterowników. Jeżeli karta pamięci jest używana to sterownik wykona program zapisany na niej zamiast programu z pamięci stałej CPU. Karta może być także używana do gromadzenia plików lub wykonywania aktualizacji oprogramowania. Pod klapkami z góry i dołu znajdują się przyłącza kablowe do obsługi wyjść i wejść sterownika. W środkowej części znajdują się diody informujące o stanie wyjść i wejść oraz o trybie pracy sterownika. Pod dolną osłoną z lewej strony znajduje się złącze PROFINET.

Rodzina sterowników S7-1200 zawiera wiele typów modułów rozszerzeń i płyt sygnałowych służących do rozszerzania możliwości jednostki centralnej. Istnieją również moduły komunikacyjne obsługujące inne protokoły komunikacyjne.

Tryby pracy i cykl pracy

Podczas konfiguracji CPU można ustawić następujące właściwości procesora:

- PROFINET interface – ustawienie adresu IP CPU i synchronizacji czasu.
- Startup – ustalanie sposobu działania CPU po przejściu ze stanu wyłączenia do stanu włączenia.
- Konfiguracja lokalnych cyfrowych i analogowych I/O, szybkich liczników oraz generatorów impulsowych.
- System clock – ustawienie czasu, strefy czasowej oraz czasu letniego.
- Ustawienie zabezpieczenia przed odczytem/zapisem oraz hasła dostępu do CPU.
- Określenie maksymalnego czasu cyklu lub ustalonego minimalnego czasu cyklu i przypisanie procentowej części czasu cyklu zadaniom komunikacyjnym.
- Właściwości web serwera.

Sterownik może pracować w jednym z trzech trybów sygnalizowanych odpowiednim kolorem diody LED RUN/STOP:

- W trybie STOP jednostka nie wykonuje programu i użytkownik może wczytać projekt do sterownika. Dioda LED RUN/STOP pali się na żółto.
- W trybie STARTUP wykonywany jest jednokrotnie blok startowy OB100 – jeżeli istnieje. W fazie startowej CPU nie obsługuje przerwań. Dioda LED RUN/STOP miga naprzemiennie na zielono i żółto.
- W trybie RUN regularnie wykonywany jest cykl programu. Mogą pojawiać się przerwania i CPU może przetwarzać je w dowolnym miejscu cyklu programu. W trybie RUN można ładować niektóre części projektu. Dioda LED RUN/STOP świeci się na zielono.

Dodatkowo za pomocą diod sterownik może również sygnalizować:

- ERROR – świecenie lub miganie diody na czerwono,
- MAINT – świecenie lub miganie diody na pomarańczowo.

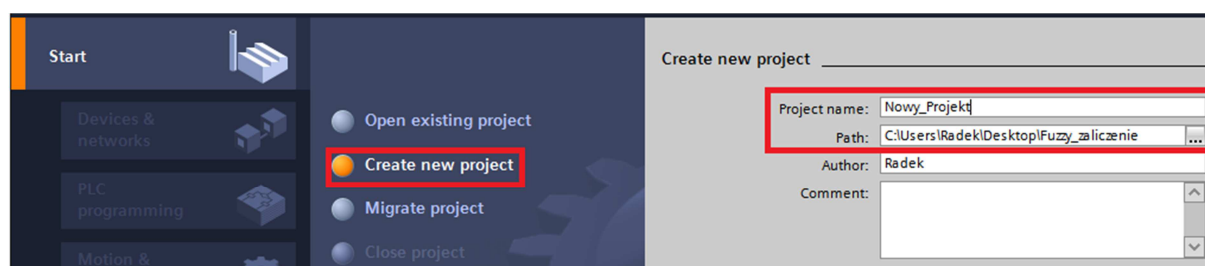
Pierwsze kroki w Tia Portal

Do programowania sterowników SIMATIC S7 służy oprogramowanie SIMATIC STEP 7, które pozwala na tworzenia aplikacji oraz na konfigurację, diagnostykę, serwis oraz optymalizację tworzonych aplikacji. Oprogramowanie STEP 7 wbudowane jest w multiprogramową platformę TIA Portal (ang. Totally Integrated Automation Portal). „Pakiet TIA Portal ma spełniać założenia "totalnie zintegrowanej automatyki". Skraca czas oraz zapewnia wygodę programowania, poprzez możliwość przygotowania programów dla całego systemu za pomocą jednej aplikacji. TIA Portal umożliwia

programowanie sterowników PLC, paneli HMI, napędów czy zarządzania energią. Za pomocą TIA Portal można w graficznych językach programowania (LAD,FBD), tekstu strukturalnego SCL, listy instrukcji STL (bez s7-1200) oraz programowaniu sekwencyjnym (Graph - bez s7-1200).

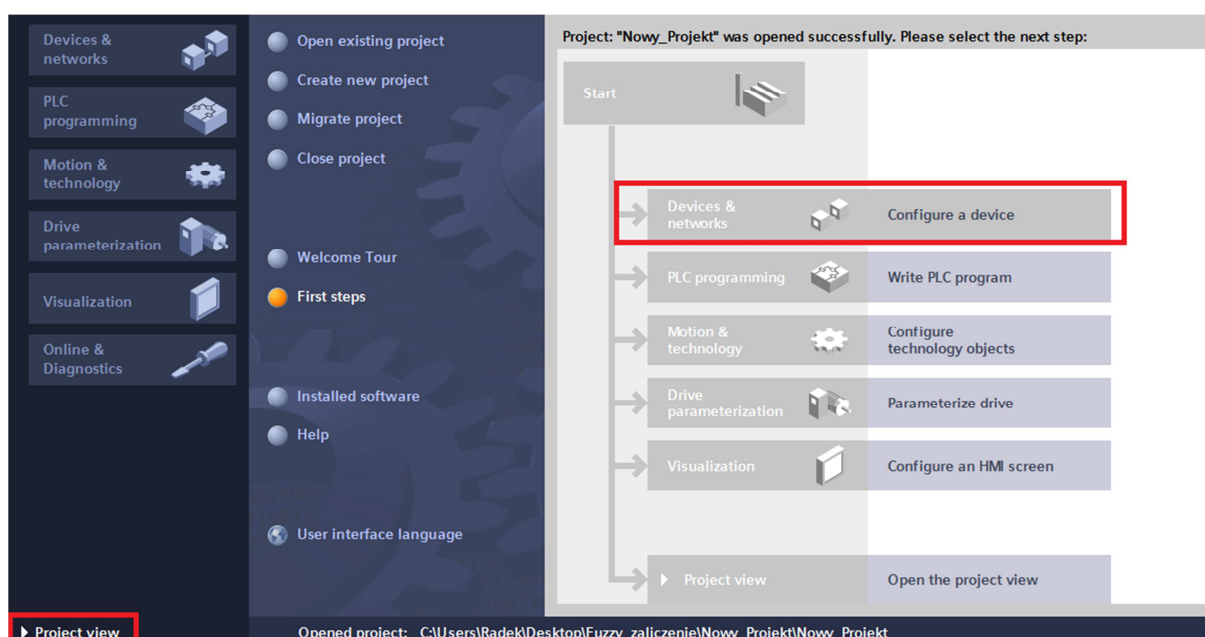
1. Łączenie ze sterownikiem

Po uruchomieniu programu pojawia się interfejs *Portal view*, z poziomu którego można otworzyć wcześniejszy projekt, bądź stworzyć nowy. W celu stworzenia nowego projektu należy użyć przycisku *Create new project* oraz wpisujemy nazwę w polu *Project name* (Rys. 2).



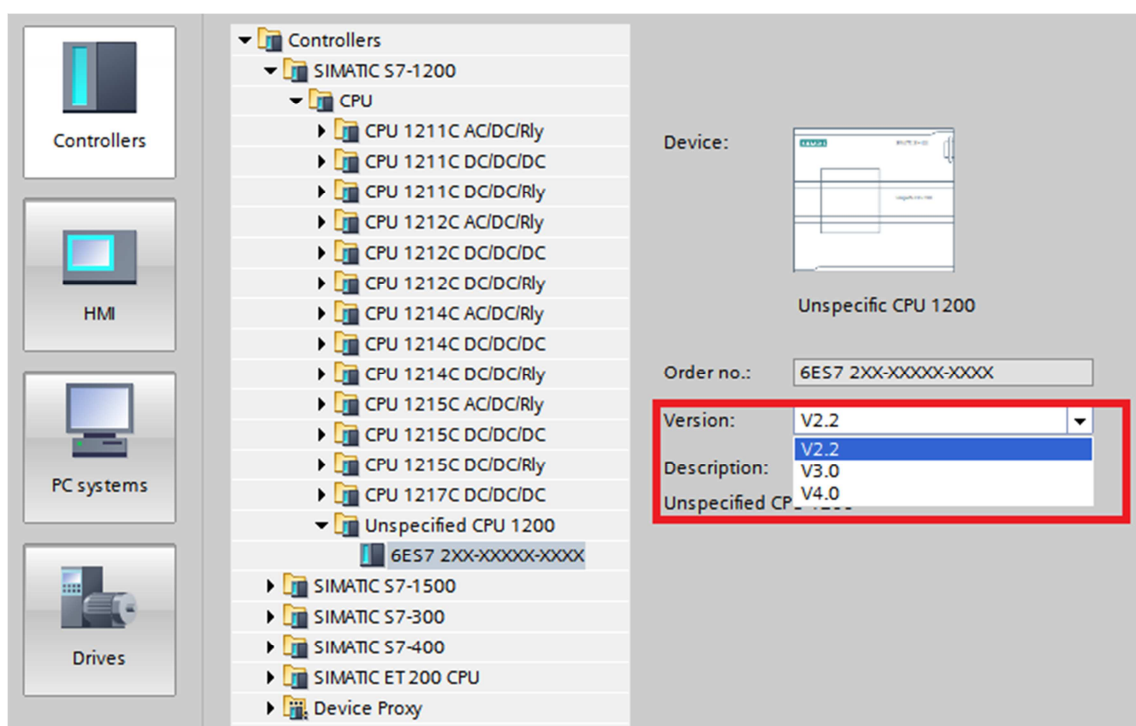
Rys. 2. Tworzenie nowego projektu

Po stworzeniu projektu pojawi się nowe okno, w którym istnieje możliwość realizacji kolejnych kroków. Należy zwrócić uwagę na lewy dolny róg. W zależności od widoku, w którym znajduje się program, można przechodzić pomiędzy *Project View* a *Portal View* (Rys. 3). Widok portalu jest widokiem dla mniej zaawansowanych użytkowników. Działania w tym widoku powodują najczęściej uruchomienie Kreatora (ang. *Wizard*), który upraszcza postępowanie z projektem. Widok projektu ma tę samą funkcjonalność, ale najczęściej nie uruchamia Kreatorów. W pierwszym kroku należy dodać sterownik PLC. Należy kliknąć *Create a device*"a następnie *Add new device*.



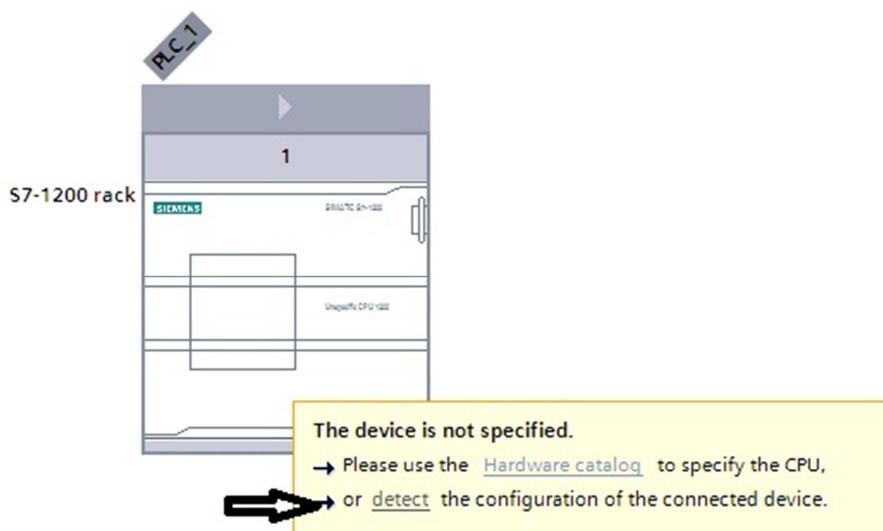
Rys. 3. Konfiguracja sterownika

Pojawi się okno z listą sterowników do wybrania podzielonych ze względu na rodziny sterowników (S7-1200, S7-1500) jak i typ sterownika (np. CPU 1214C AC/DC/RLY). Przy wyborze typu sterownika należy zwrócić uwagę na litery przy numerze sterownika. Pierwsza pozycja oznacza rodzaj zasilania sterownika, gdzie AC oznacza zasilanie prądem zmiennym 120/230 V, a DC zasilanie prądem stałym 24 V DC. Na drugiej pozycji jest rodzaj wejść. W tym wypadku są tylko wejścia prądu stałego (24 V DC). Trzecia pozycja to rodzaj wyjść, gdzie DC - wyjścia tranzystorowe, Rly - wyjścia przekaźnikowe. Można zatem dodać sterownik ręcznie poprzez wybór odpowiedniego modelu. Wygodniejszym rozwiązaniem jest jednak zadeklarowanie urządzenia poprzez pozycję *Unspecified CPU 1200*. Przy wyborze tej metody, pojawi się lista dostępnych firmware'ów sterowników. **Przy wyborze sterownika wybiera się jego najstarszą wersję** (Rys. 4). Jeśli nie jest obecny sterownik w wersji 2, przeszukane zostaną nowsze wersje sterowników.



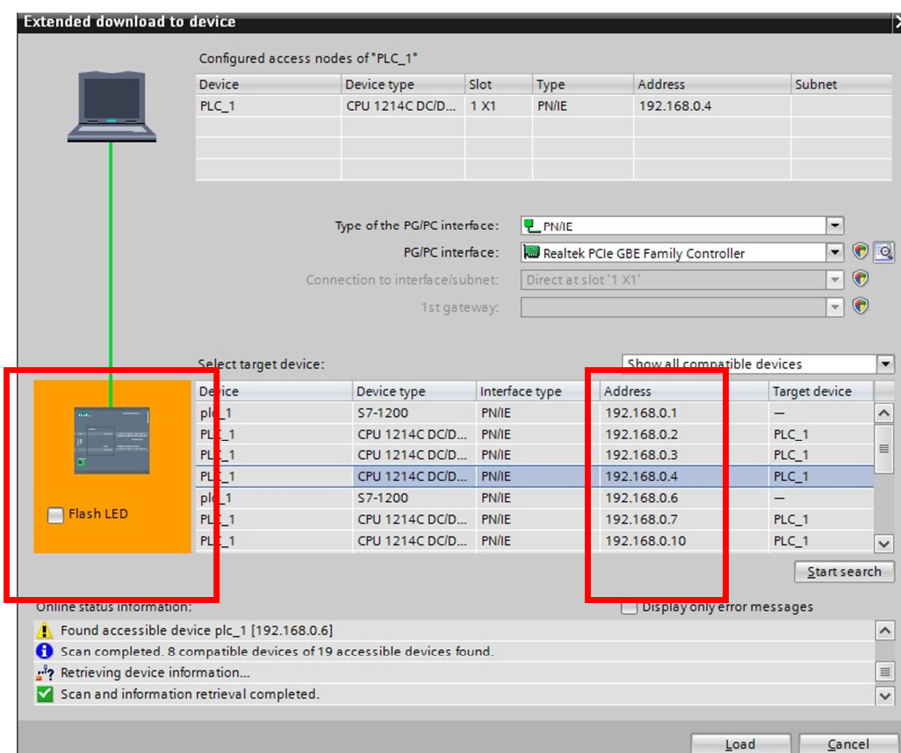
Rys. 4. Wybór odpowiedniego sterownika z listy sterowników oraz wybranie najstarszej wersji

Po dodaniu niesprecyzowanego sterownika, Tia Portal przejdzie do okna *Devices & Networks*. Należy nacisnąć pozycję *Detect* (Rys. 5). Ponieważ w laboratorium sterowniki połączone są ze sobą po jednej sieci PN/IE, to każdy z zasilonych sterowników będzie dostępny w liście dostępnych urządzeń. Zaznaczając opcję *flash led* można fizycznie sprawdzić, z którym sterownikiem łączy się użytkownik (Rys. 6), gdy na danym sterowniku zaczną migać diody *Run/Error/Maint*.



Rys. 5. Detekcja niesprecyzowanego sterownika

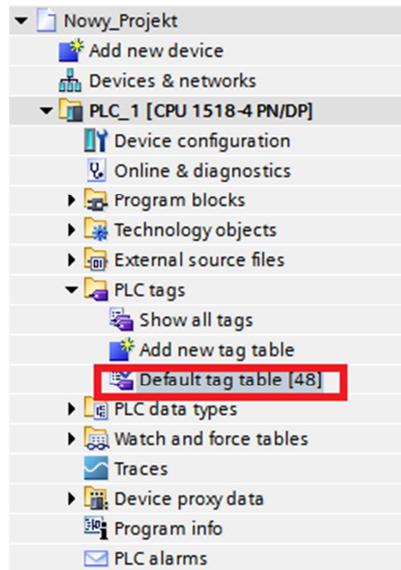
Uwaga: Nie należy wykrywać sterownika równocześnie na wielu komputerach



Rys. 6. Łączenie się z odpowiednim sterownikiem

Pierwszy program w Tia Portal

Po wykonaniu konfiguracji urządzenia, można przejść do utworzenia pierwszego programu. Dobra praktyka inżynierska sugeruje, aby zacząć od definicji tagów (Rys. 7). *Tagi PLC* są nazwami symbolicznymi dla wejść/wyjść lub adresów. Aby utworzyć nowe tagi należy otworzyć tablicę tagów w drzewie projektu.



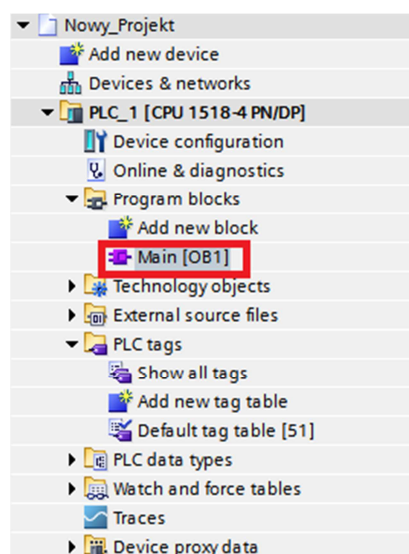
Rys. 7. Wybór tablicy tagów

Po otwarciu tablicy tagów oznaczamy przycisk start, przycisk stop i lampkę. (Rys. 8). Wszystkie te trzy zmienne są zmiennymi typu *Bool*.

Start	Bool	%I0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Stop	Bool	%I0.1	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>
Lampka	Bool	%Q0.0	<input type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>

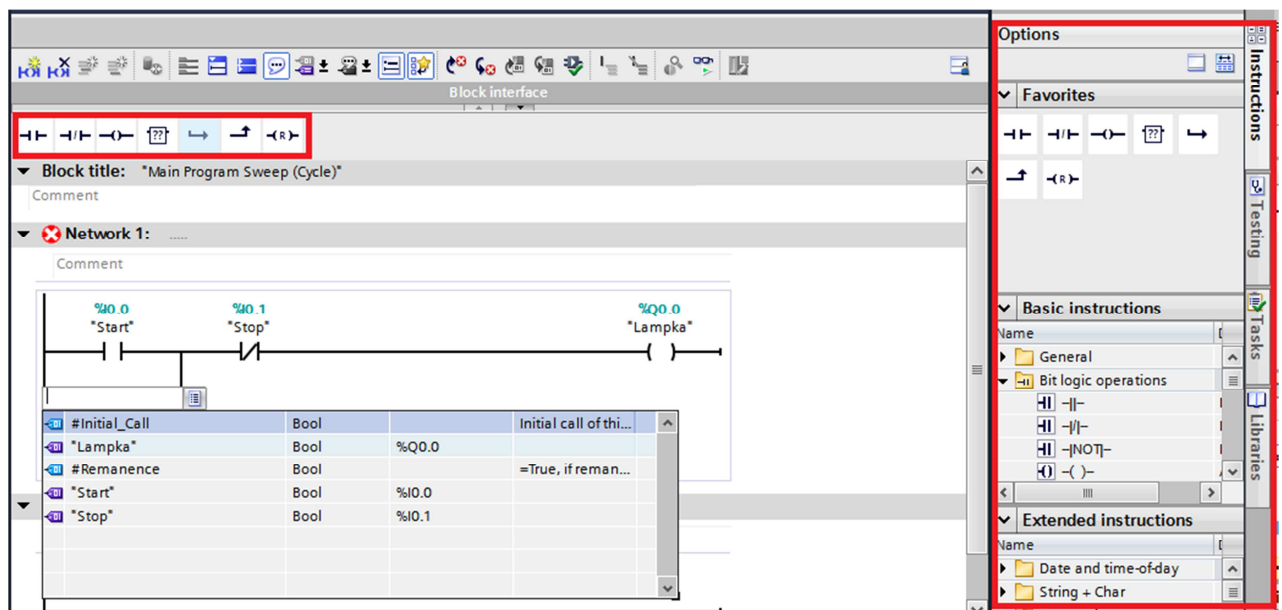
Rys. 8. Przypisanie tagów do adresów

Po utworzeniu zmiennych, przechodzi się do napisania pierwszego programu. Będzie to program pozwalający na załączenie cewki z podtrzymaniem i priorytetem na stop (tzw. Zatrząsk). W drzewie projektu należy wybrać *Program Blocks*, a następnie otworzyć blok organizacyjny Main [OB1], który został domyślnie utworzony wraz z projektem (Rys. 9).



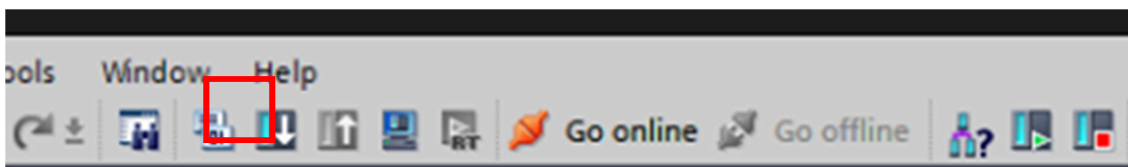
Rys. 9. Wybór funkcji "Main"

Aby stworzyć program należy wybrać odpowiednie styki i cewkę. Najprościej jest je wybrać z ulubionych styków znajdujących się u góry ekranu programowania (Rys. 10). W przypadku, gdy styki tam się nie znajdują, możemy je odnaleźć rozwijając listę instrukcji po prawej stronie ekranu (Rys. 10). W celu dodania dowolnej instrukcji do ulubionych, wystarczy ją przeciągnąć do okna *Favorites*. Tagi można, po ich wcześniejszym zdefiniowaniu, przypisać wybierając z listy tagów. Nazwę tagu można wpisać także bezpośrednio, lub możemy po prostu wpisać nazwę tagu do okienka.



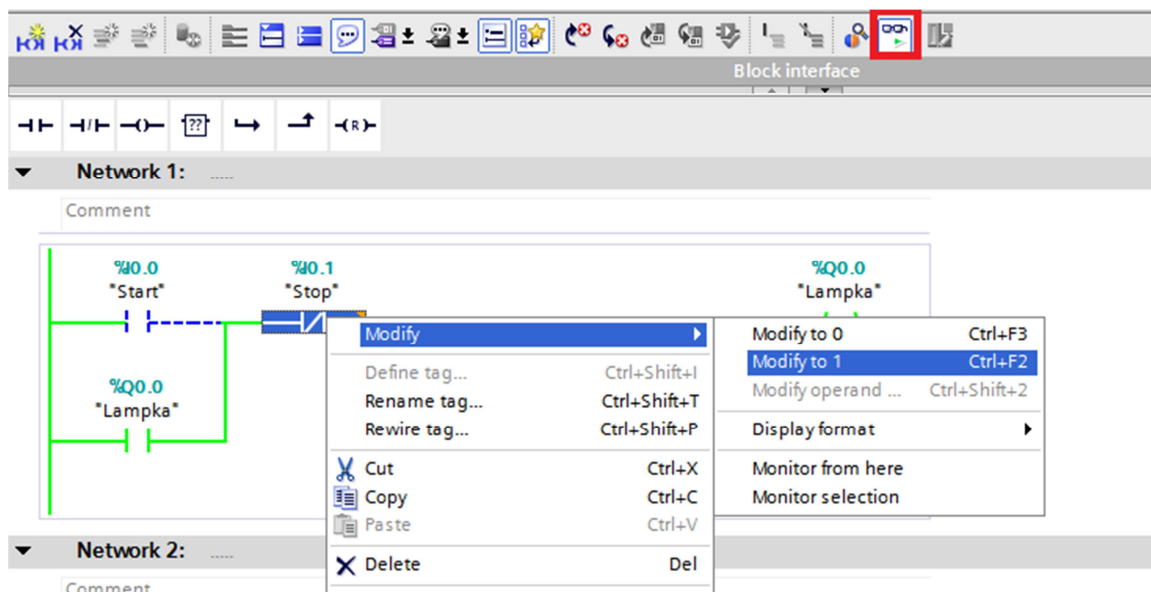
Rys. 10. Instrukcje *Favorite* oraz konstrukcja programu

Po utworzeniu programu należy wgrać go do sterownika PLC (Rys. 11) za pomocą przycisku *Download to device*.



Rys. 11. Ikona wgrania programu do sterownika

W pakiecie TIA Portal istnieje możliwość monitorowania działania programu. W tym celu należy przycisnąć ikonę z okularami w pasku nad ulubionymi (*Monitoring On/Off* - Rys. 12). Można również zmienić wartość adresu klikając prawym przyciskiem myszy na styk, a następnie na *Modify* i wybrać daną wartość.

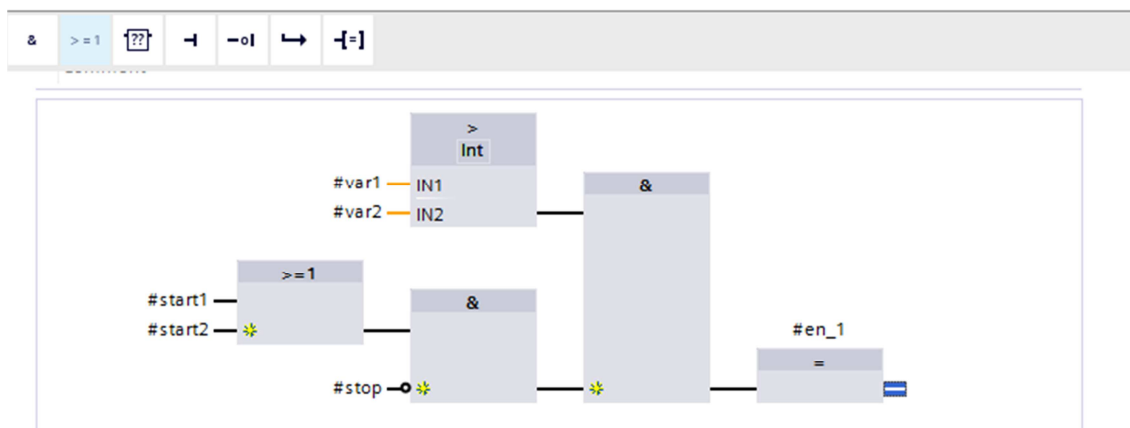


Rys. 12. Monitorowanie programu oraz zmiana wartości adresów

Pozostałe języki programowania

Sterowniki rodziny s7-1200 mogą być programowane także w dwóch innych językach określonych przez normę IEC 61131-3. Są to FBD oraz SCL. Aby móc programować w tych językach, należy dodać blok OB typu *Program Cycle* z drzewa projektu za pomocą *Add new block*, a następnie wybrać odpowiedni *Programming Language*.

FBD (ang. Functional Block Diagram) – diagram bloków funkcyjnych jest językiem graficznym używanym do budowy programu wykorzystując bibliotekę funkcji. Programy są przedstawiane jako połączenie bloków funkcyjnych, zadających zależność między wejściami, a wyjściami. W celu stworzenia logiki złożonych operacji należy połączyć symbole logiczne równoległymi gałęziami (Rys. 13). Podstawowymi blokami tego języka są: instrukcja Koniunkcji (&) oraz Alternatywa (≥ 1) oraz blok cewki (=).



Rys. 13. Przykład programowania w języku FBD

Ćwiczenie 1

Dodaj i skonfiguruj sterownik PLC

Ćwiczenie 2

Zrealizuj zatrask pokazany w części teoretycznej formatki.

Ćwiczenie 3

Dodaj dwa tagi – *czujnik* (I0.2) oraz *lampkaPrawa* (q0.1). Zrealizuj zatrask na *lampkaPrawa* z użyciem *czujnik* oraz *czerwony*. Zatrask powinien być zrealizowany za pomocą języka FBD.

Ćwiczenie 4

Zdeklaruj tag *Liczba*, typu INT, o adresie MW10. Korzystając z matematycznej instrukcji *ADD* zrealizuj program, w którym po każdorazowym naciśnięciu przycisku zielonego wartość *Liczba* wzrasta o 2. Zwróć uwagę na sposób działania aplikacji, jeśli nie zostanie użyte zbcze narastające dla styku *zielony*.