# CoDeSys 3 – programowanie w języku drabinkowym LD

Spis treści



Powering Business Worldwide

# 1. Wstęp

Aplikacja CoDeSys 3 jest nowoczesnym środowiskiem służącym do programowania sterowników firmy Eaton Electric.

Notatka ta ma na celu przedstawienie programowania w języku CFC jednym z 7 języków zgodnych z normą IEC-61131-3. Podstawy programowania przedstawiono w notatkach NA03001PL oraz NA03002PL.

## 1.1. Wymagania programowe

Przedstawiana notatka opiera się na oprogramowaniu narzędziowym Xsoft-CoDeSys V3.5.7 (Build 3152).

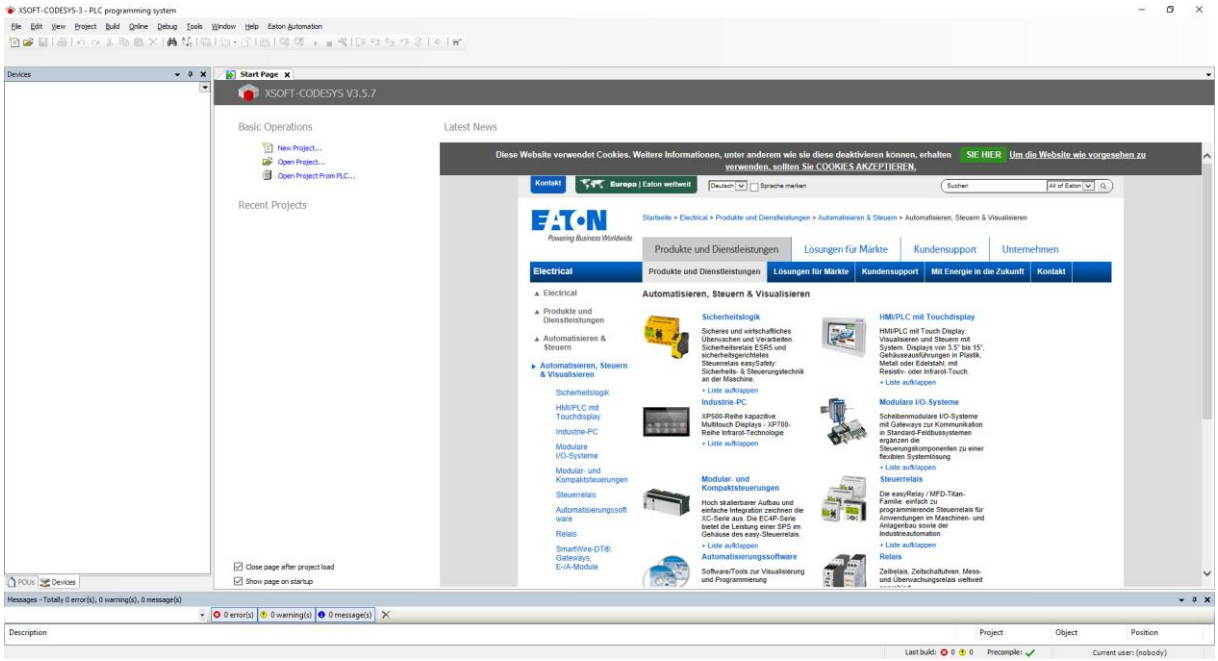
Oprogramowanie przetestowane jest dla systemu Windows XP/7/8 (32,64bit), pracuje również poprawnie w Windows 10, ale nie było to szczegółowo testowane i użytkownik robi to na własną odpowiedzialność.

Aktualne wersje demonstracyjne oprogramowania Eaton można pobrać z Download Centre pod adresem:
http://applications.eaton.eu/sdlc

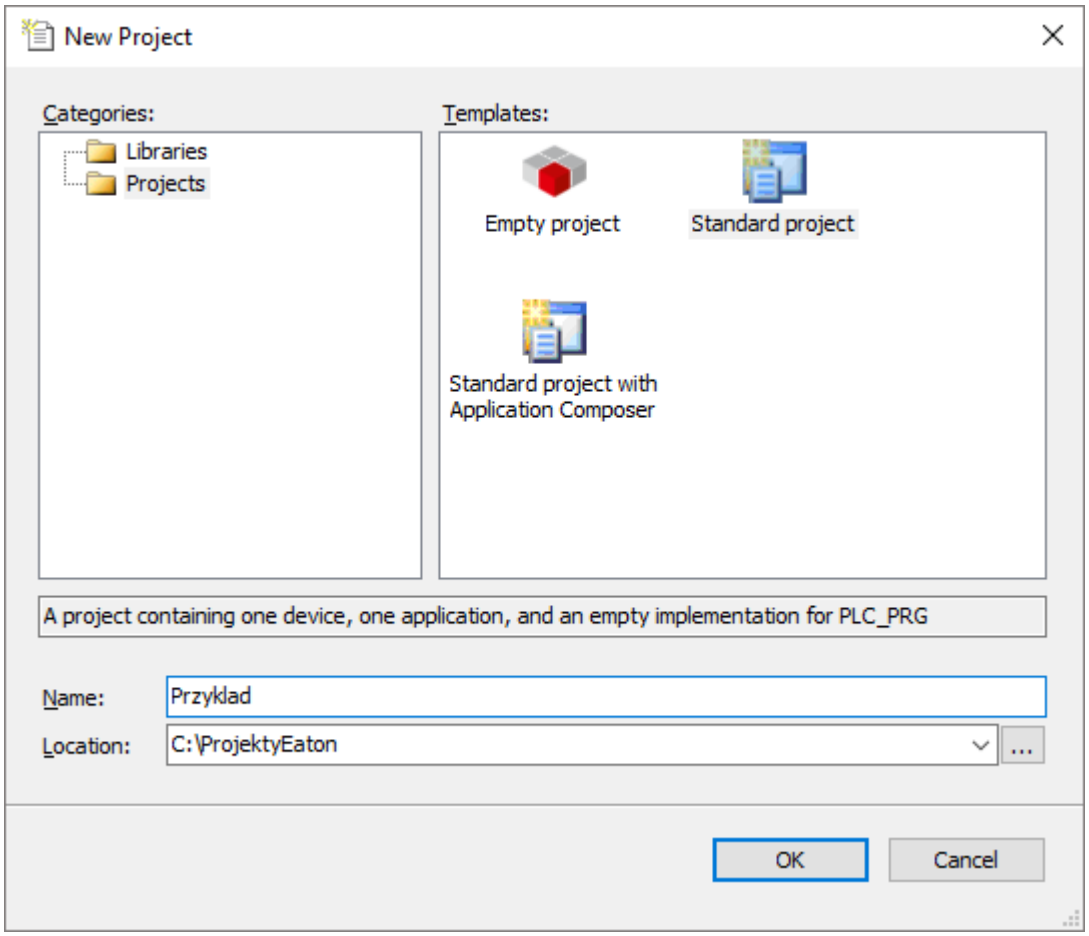
Uwaga. Z poziomu CoDeSys 3 nie ma możliwości zaprogramowania sterowników EC4P, XC-CPU101, XC-CPU201, XN-PLC, XC-CPU121. Jednostki te można programować jedynie z poziomu CoDeSys 2.

# 2. Tworzenie projektu i dodawanie programu w języku LD

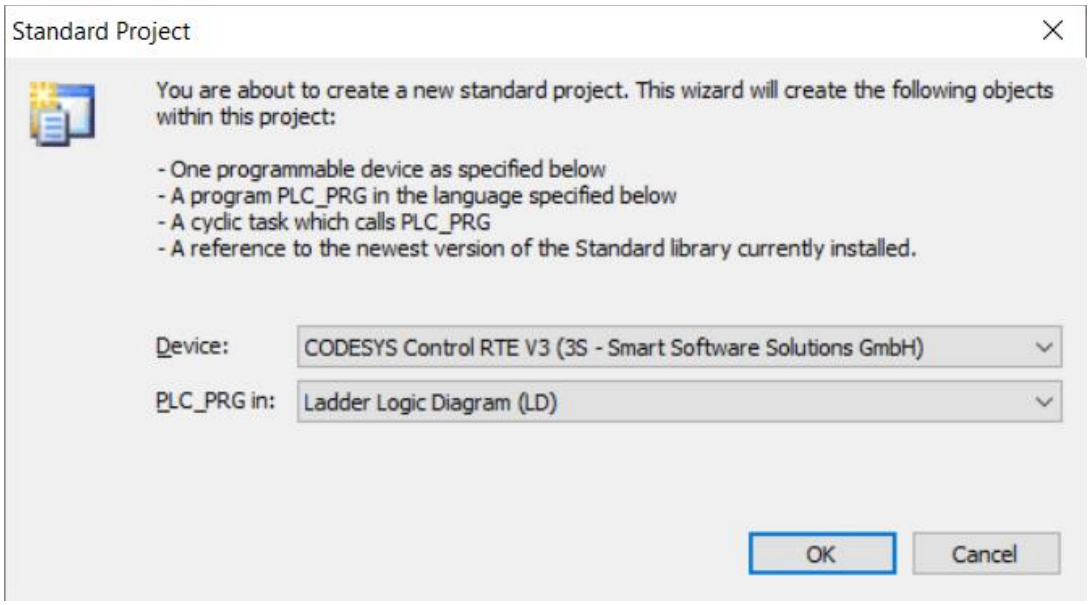
Uruchamiamy CoDeSys 3 i w oknie **Start Page** wybieramy opcję **New Project**:



W okienku **New Project** wybieramy **Standard project** – wtedy




W kolejnym oknie **Standard Project** wybieramy typ sterownika oraz język programowania programu głównego. Mamy do dyspozycji 7 języków programowania zgodnych z IEC. W obrębie projektu programy mogą być napisane w różnych językach. Na potrzeby tej notatki wybieramy **XV300** oraz język najwyższego poziomu **CFC.**



# 3. Organizacja okien dla języka LD

Po kliknięciu na program LD np. **PLC_PRG** ekran wyświetla się jak poniżej. Jest podzielony na pięć części:

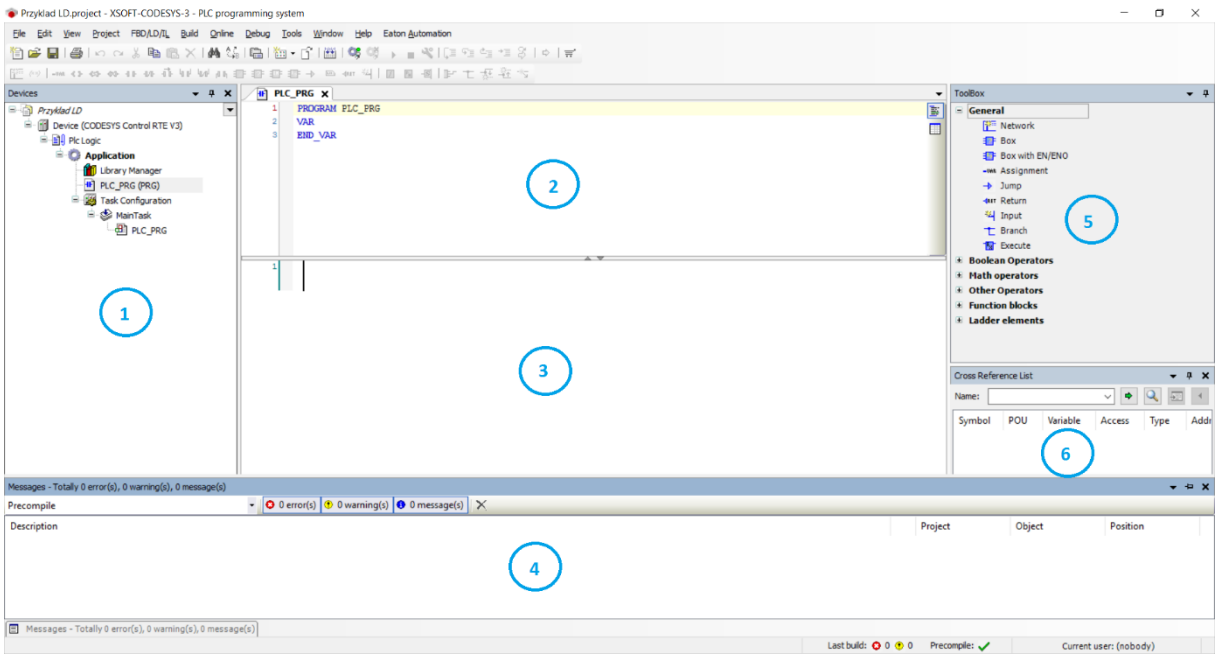
1 – Struktura projektu;

2 – Okno zmiennych lokalnych;

3 - Okno w którym piszemy program;

4 – Okno przebiegu kompilacji projektu;

5 – Toolbox - elementy programowania dla LD.



# 4. Składnia języka LD – Toolbox

Gdy wybrany jest język LD w narzędziu toolbox znajdują się następujące elementy tego języka:

Elementy dedykowane dla języka drabinkowego

 - nowa linia programu. Analogia do fizycznego przewodu prowadzonego od źródła znajdującego się z lewej strony poprzez styki do cewki.

 - styk - przewodzi sygnał od lewej do prawej gdy zmienna bool do niego przypisana jest w stanie wysokim. Analogia do styku przekaźnika fizycznego. Po wstawieniu styk możemy zanegować klikając na niego prawym przyciskiem i wybierając Negation

 - styk zanegowany - przewodzi sygnał od lewej do prawej gdy zmienna bool do niego przypisana jest w stanie niskim

 - wstawianie styku równolegle do innych. Umożliwia wykonanie sumy logicznej OR

 - jak wyżej tylko styk jest zanegowany

 - cewka – przeniesienie sygnału z linii na zmienną do niej przypisaną. Analogia do cewki przekaźnika fizycznego. Po wstawieniu cewkę możemy zanegować klikając na nią prawym przyciskiem i wybierając Negation

 - cewka ustawiająca – gdy na linii przed tą cewką pojawi się choć na chwilę stan wysoki to zmienna do niej przypisana zostaje ustawiona w stan wysoki i w nim pozostaje

 - cewka resetująca – gdy na linii przed tą cewką pojawi się choć na chwilę stan wysoki to zmienna do niej przypisana zostaje ustawiona w stan niski i w nim pozostaje

 - opóźnione załączenie – gdy na wejściu IN stan wysoki to po czasie z wejścia PT na wyjściu Q pojawi się stan wysoki. Na wyjściu ET wyprowadzony jest czas od przejścia IN w stan wysoki. Gdy wejście IN przejdzie w stan niski to blok się zeruje

 - opóźnione wyłączanie - gdy na wejściu IN stan wysoki zmieni się na niski to na wyjściu stan zmieni się na niski, ale po czasie z wejścia PT. Na wyjściu ET wyprowadzony jest czas od przejścia IN w stan niski

 - zliczanie w górę – z każdym przejściem wejścia CU ze stanu niskiego w wysoki wartość na wyjściu CV zwiększana jest o 1 aż do osiągnięcia wartości z wejścia PV wtedy na wyjściu Q pojawia się stan wysoki. Wejście RESET zeruje licznik

 - zliczanie w dół - z każdym przejściem wejścia CU ze stanu niskiego w wysoki wartość na wyjściu CV zmniejszana jest o 1 począwszy od wartości z wejścia PV. Gdy CV=0 wtedy na wyjściu Q pojawia się stan wysoki. Wejście RESET zeruje licznik

 - przepisanie zmiennej wejściowej na wyjściową jeżeli na wejściu EN jest stan wysoki

 - skok do linii z etykietą (label) o takiej samej nazwie

 - jeśli na wejściu stan wysoki program dalej się nie wykonuje, wykonywany jest następny program

 - odgałęzienie linii programu

 - odgałęzienie od punktu do punktu. Przenosimy na punkt gdzie linia ma się odgałęzić a następnie drugi raz w punkt gdzie ma nastąpić powrót

Ogólne

 - nowa linia programu

 - nowy blok funkcyjny, funkcja lub program

 - jak wyżej tylko z wejściem enable

 -

 - skok do linii z etykietą (label) o takiej samej nazwie

 - jeśli na wejściu stan wysoki program dalej się nie wykonuje, wykonywany jest następny program

 - nowe wejście do bloku funkcyjnego

 - odgałęzienie linii programu

 - umożliwia wykonanie kodu ST, który pisze się wewnątrz tego elementu

Operatory logiczne

 - iloczyn logiczny AND z dwoma wejściami

 - iloczyn logiczny AND z trzema wejściami

 - suma logiczna OR z dwoma wejściami

 - suma logiczna OR z trzema wejściami

 - alternatywa wykluczająca XOR z dwoma wejściami

Operatory matematyczne

 - dodawanie dwóch zmiennych

 - dodawanie trzech zmiennych

 - odejmowanie dwóch zmiennych

 - mnożenie dwóch zmiennych

 - dzielenie dwóch zmiennych – uwaga w przypadku dzielenia przez zero może nastąpić zatrzymanie sterownika PLC

 - sprawdzenie czy zmienne wejściowe są sobie równe wtedy na wyjściu stan wysoki

 - jeśli na wejściu zmienne różne to na wyjściu stan wysoki

 - jeśli zmienna na wejściu 1 mniejsza od zmiennej na wejściu 2 wtedy stan wysoki na wyjściu

 - jeśli zmienna na wejściu 1 mniejsza lub równa zmiennej na wejściu 2 wtedy stan wysoki na wyjściu

 - zmienna na wejściu 1 większa od zmiennej na wejściu 2 wtedy stan wysoki na wyjściu

 - jeśli zmienna na wejściu 1 większa lub równa zmiennej na wejściu 2 wtedy stan wysoki na wyjściu

Pozostałe operatory

 - gdy na wejściu G jest stan niski to na wyjście zostaje przepisana wartość z IN0 a jeśli wysoki to z IN1

 - przepisanie na wyjście wartości z wejścia o numerze równym wartości zmiennej na wejściu K

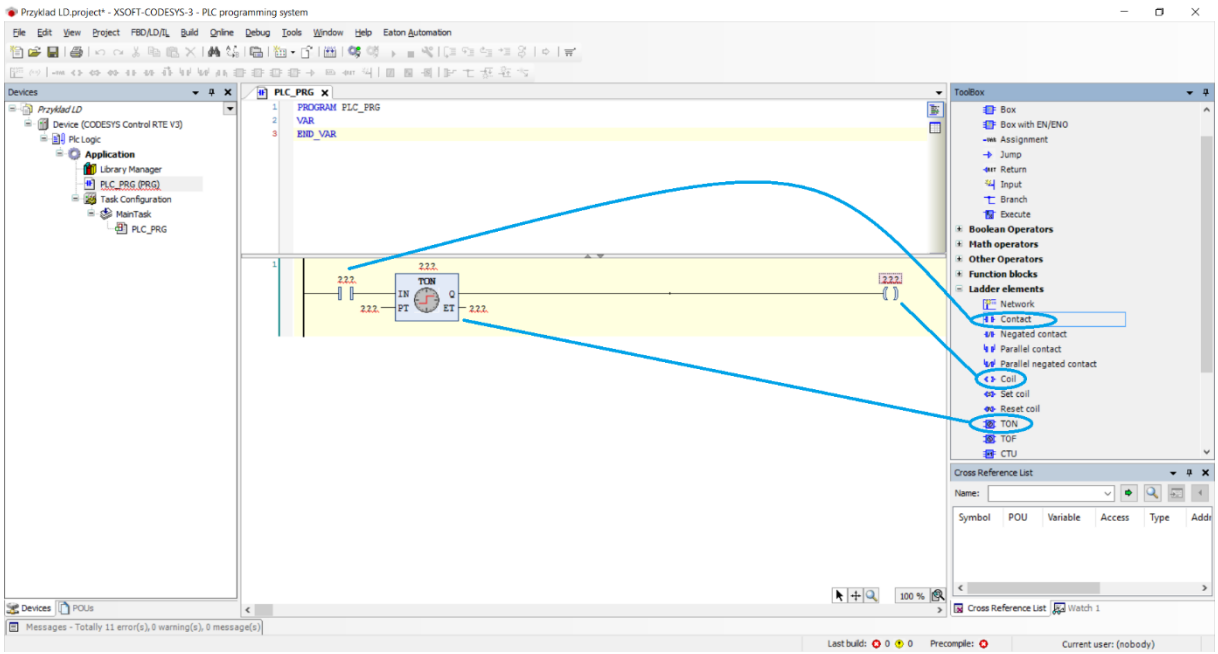
 - ograniczenie - na wyjście przepisywana jest wartość wejściowa, ale nie mniejsza niż wartość MN i nie większa niż MX

 - przepisanie zmiennej wejściowej na wyjściową jeżeli na wejściu EN jest stan wysoki

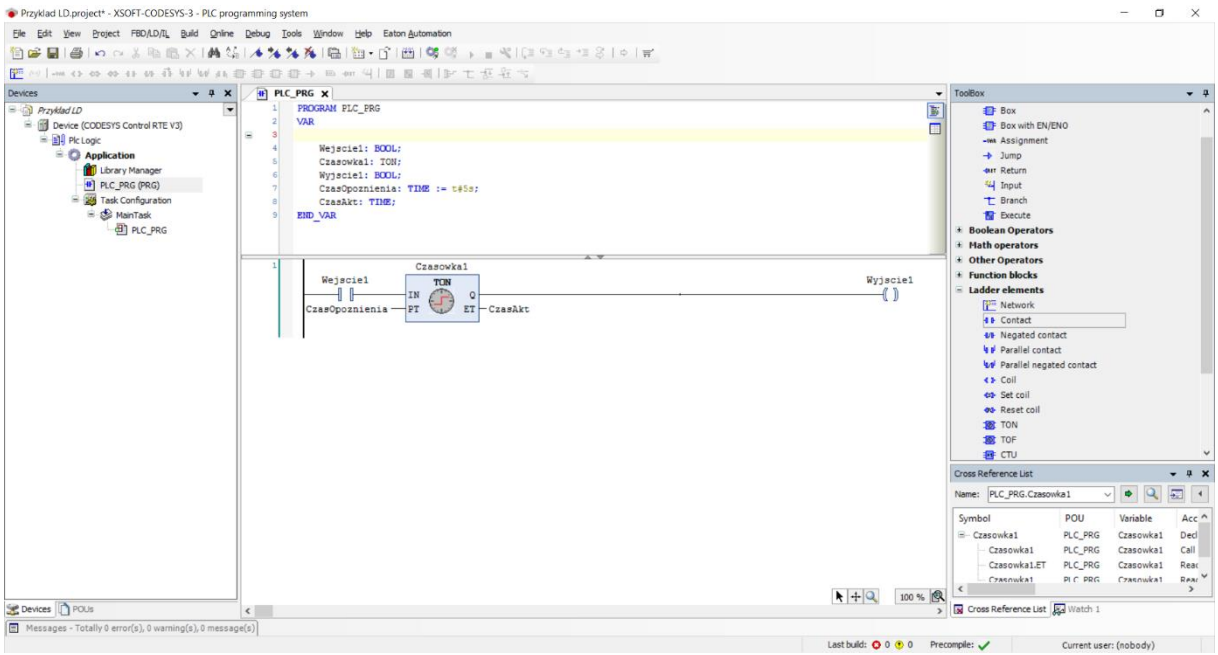
 - konwersja typu zmiennej wejściowej na typ zmiennej wyjściowej. Jeżeli wejście jest typu WORD a wyjście typu INT nazwa bloku uzupełniona jest do WORD_TO_INT. Możemy wpisać lub skorzystać z asystenta deklaracji

# 5. Przykładowy program

Przedstawimy tutaj bardzo prosty przykład opóźnionego załączania wyjścia. W tym celu do pierwszej linii przeciągamy styk, blok TON oraz cewkę.




Deklarujemy nazwy zmiennych i bloku korzystając z Asystenta Deklaracji.




Po poprawnej kompilacji (przycisk  ), program możemy wgrać do sterownika.