

Wykład 2

Sterowniki programowalne (PLC, PAC), norma IEC61131, proces konfiguracji, programowania i uruchamiania sterowników.

1. **Sterowniki programowalne - wprowadzenie**
2. **Norma IEC 61131-3 - wprowadzenie**
3. **Konfiguracja i uruchomienie sterownika PLC/PAC na przykładzie sterownika serii CX firmy Beckhoff**
 - 3.1. Konfiguracja pakietu TwinCAT do współpracy ze sterownikiem serii CX
 - 3.2. Tworzenie prostego programu w języku ST – wprowadzenie
 - 3.3. Zdalny pulpit systemu Windows CE
4. **Literatura**

1. Sterowniki programowalne - wprowadzenie

Sterowniki programowalne są komputerami przemysłowymi, które pod kontrolą systemu operacyjnego czasu rzeczywistego:

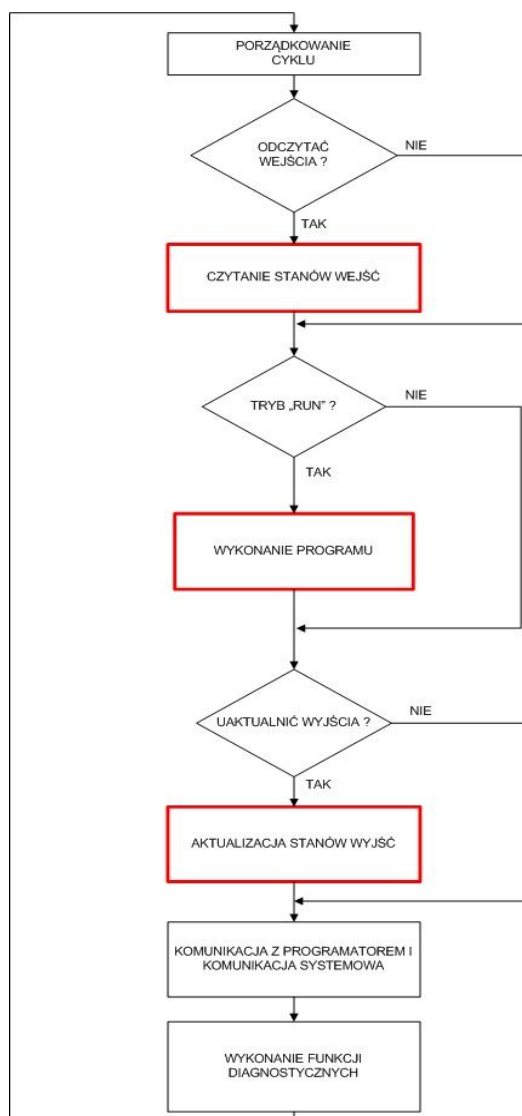
- *zbierają pomiary* za pomocą modułów wejściowych z cyfrowych i analogowych czujników oraz urządzeń pomiarowych;
- korzystając z uzyskanych danych o sterowanym procesie lub maszynie, *wykonują programy użytkownika*, zawierające zakodowane algorytmy sterowania i przetwarzania danych;
- *generują sygnały sterujące* odpowiednie do wyników obliczeń tych programów i przekazują je przez moduły wyjściowe do elementów i urządzeń wykonawczych [1].

PLC (*Programmable Logic Controller*) - *Programowalny Sterownik Logiczny* – uniwersalne urządzenie mikroprocesorowe przeznaczone do sterowania pracą maszyny lub urządzenia technologicznego [2]. Słowo *logic* (logiczny) ma dzisiaj znaczenie historyczne, współczesne sterowniki określane jako PLC oprócz typowych zadań sterowania logicznego (dwustanowego) wykonują również wiele innych działań, np. regulację, komunikację, przetwarzania danych.

PAC (*Programmable Automation Controller*) – *Programowalny Sterownik Automatyki*

Aby sterownik mógł skutecznie realizować zadania sterowania, musi pracować w tzw. *czasie rzeczywistym (Real Time)*. Oznacza to, że reakcja sterownika w postaci obliczonego sterowania w odpowiedzi na zdarzenie, które miało miejsce w obiekcie, musi wystąpić w określonym czasie, akceptowalnym z punktu widzenia wymagań stawianych temu sterowaniu [1].

Sterownik pracuje w cyklu, zwanym *cyklem programowym*, zazwyczaj posiadający stały czasem trwania, w którym wykonuje fazy przedstawione na poniższym rysunku [1].



2. Norma IEC 61131-3 – wprowadzenie

ISO (International Organization for Standardization - Międzynarodowa Organizacja Normalizacyjna) – wydaje normy międzynarodowe – www.iso.org

W dziedzinie elektrotechniki normalizacją kieruje

IEC (International Electrotechnical Commission - Międzynarodowa Komisja Elektrotechniczna) – www.iec.ch

Informacje na temat norm

PKN (Polski Komitet Normalizacyjny) – www.pkn.pl

Norma IEC 61131 – Sterowniki Programowalne (*Programmable Controllers*)

- pierwsza publikacja normy - październik 1993 r.

- do 1998 nazwa normy IEC 1131

- drugie poprawione wydanie normy IEC 61131-3 – 2003 r.

Elementy składowe normy

1. Postanowienia ogólne (*General Information*)
2. Wymagania i badania dotyczące sprzętu (*Equipment requirements and tests*)
3. Języki programowania (*Programming languages*)
4. Wytyczne dla użytkownika (*User guidelines*)
5. Wymiana informacji (*Communications*)
6. Komunikacja w sieciach polowych (*Communications via fieldbus*) - projekt
7. Programowanie sterowania z wykorzystaniem zbiorów rozmytych (*Fuzzy control programming*)
8. Wytyczne do implementacji języków programowania (*Guidelines for the application and implementation of programming languages*)
9. Single-drop digital communication interface for small sensors and actuators –SDCI - projekt

Cel opracowania normy IEC 61131

Wzrastająca złożoność systemów ze sterownikami PLC powodowała stały wzrost kosztów w obszarach:

- szkolenia programistów PLC;
- tworzenia coraz większych programów;
- tworzenia i wdrażania systemów o ciągle rosnącej złożoności.

Rosnące możliwości platform sprzętowych stosowanych w sterownikach PLC wymagały wsparcia w zakresie narzędzi programistycznych umożliwiających tworzenie zarówno prostych jak złożonych programów w celu obniżenia kosztów oraz zapewnienia jakości oprogramowania.

Wymagania stawiane narzędziom programistycznym stosowanym do programowania sterowników PLC:

- jednoczesne korzystanie z wielu języków programowania;
- modyfikacja programów PLC w trybie *on-line*;
- tworzenie dokumentacji istniejących programów (*reverse documentation*);
- wielokrotne wykorzystania elementów programów PLC;
- testowanie i symulacja programów w trybie *off-line*;
- zintegrowane narzędzia do konfiguracji i pomocy;
- zapewnienie jakości i procesu dokumentowania projektów;
- wykorzystanie systemów z interfejsami typu otwartego (*open interfaces*).

IEC 61131-3 Programming Languages

Główne cechy definiowane przez IEC 61131

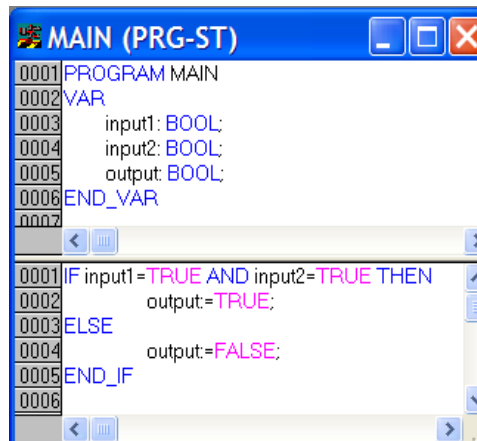
1. Wygoda i bezpieczeństwo odnośnie stosowania zmiennych:
 - globalne i lokalne zmienne zamiast adresów sprzętowych;
 - dostęp do danych PLC zorientowany na typ zmiennych;
 - jawna deklaracja typów wszystkich danych wraz z przypisaniem ich początkowych wartości (domyślnych bądź określonych przez programistę);
 - dostępność tablic i możliwość definiowania złożonych struktur danych;
 - jednolity sposób definiowania zmiennych dla wszystkich języków programowania.
2. Jednostki organizacyjne oprogramowania (POU) z rozbudowaną funkcjonalnością (POU – *Program Organization Unit* -funkcje, bloki funkcjonalne, programy):
 - możliwość wielokrotnego wykorzystywania POU;
 - ujednolicony sposób przekazywania danych pomiędzy POU;
 - zdefiniowane standardowe POU.
3. Nowy sposób konfiguracji PLC oraz ustalania parametrów działania programów:
 - zadania i programy są przypisywane do elementów sprzętowych na najwyższym poziomie projektu;
 - czas cyklu (*cycle time*) i priorytet programów definiowany podczas konfiguracji.
4. Ujednolicone języki programowania – **IEC 61131-3**

- 2 języki tekstowe

- IL (*Instruction List*) – lista rozkazów

```
0001 PROGRAM MAIN
0002 VAR
0003   input1: BOOL:
0004   input2: BOOL:
0005   output: BOOL:
0006 END_VAR
0007
0008 LD   input1
0009 EQ   TRUE
0010 AND  (input2
0011 EQ   TRUE
0012 )
0013 NOT
0014 JMP  else1_0
0015 LD   TRUE
0016 ST   output
0017 JMP  end1_0
0018 else1_0:
0019 LD   FALSE
0020 ST   output
0021 end1_0:
```

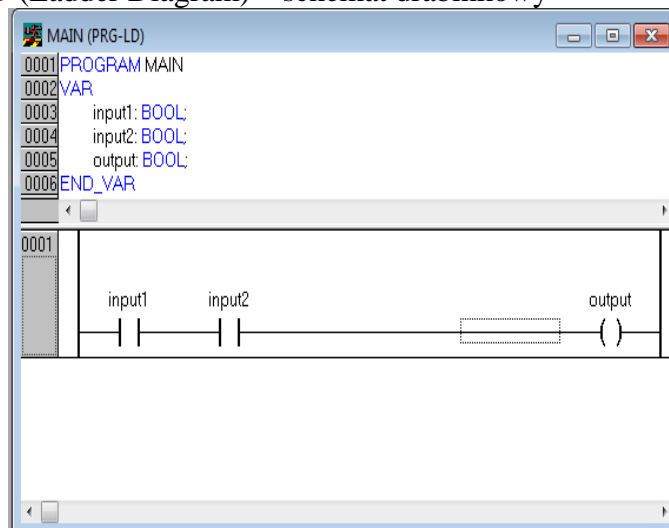
- ST (*Structured Text*) – tekst strukturalny



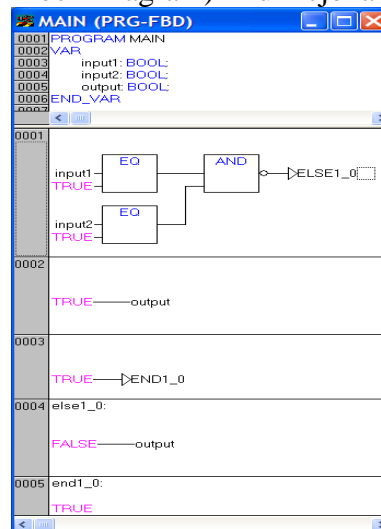
```
0001 PROGRAM MAIN
0002 VAR
0003   input1: BOOL;
0004   input2: BOOL;
0005   output: BOOL;
0006 END_VAR
0007
0001 IF input1=TRUE AND input2=TRUE THEN
0002   output:=TRUE;
0003 ELSE
0004   output:=FALSE;
0005 END_IF
0006
```

- 2 języki graficzne

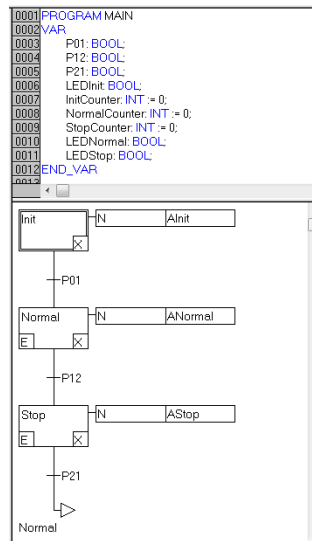
- LD (Ladder Diagram) – schemat drabinkowy



- FBD (Function Block Diagram) – funkcjonalny schemat blokowy



- 1 język do tworzenia sekwencyjnego schematu funkcjonalnego
- SFC – Sequential Function Chart



5. Strukturyzacja programów PLC:
 - definiowanie struktury programu (definiowanie danych, program główny);
 - struktura konfiguracji programowej i sprzętowej;
 - język SFC – strukturyzacja programu PLC.

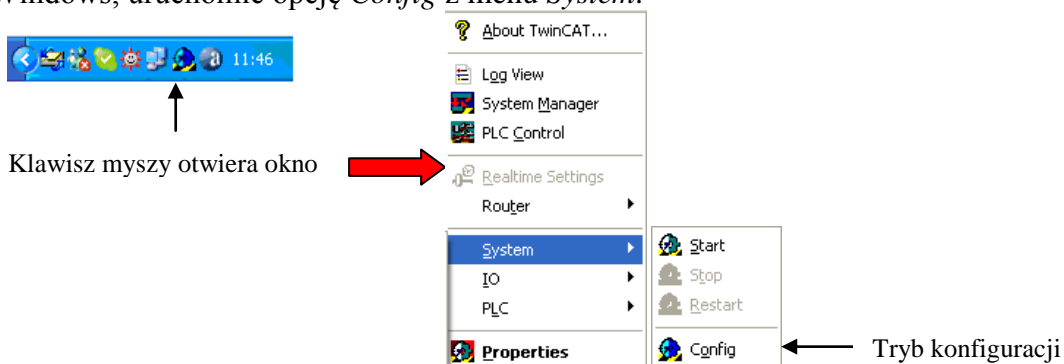
6. Trend dotyczący stosowania otwartych systemów PLC (Open PLC)
 - przenośność oprogramowania bez względu na producentów sprzętu

3. Konfiguracja i uruchomienie sterownika PLC/PAC na przykładzie urządzenia serii CX firmy Beckhoff

3.1. Konfiguracja pakietu TwinCAT do współpracy z sterownikiem CX

Uruchomienie systemu TwinCAT w trybie konfiguracji

Przy pomocy ikony pakietu TwinCAT  znajdującej się na pasku zadań systemu Windows, uruchomić opcję *Config* z menu *System*.



Tryby pracy TwinCAT są sygnalizowane odpowiednimi kolorami ikony programu:



- zielony – uruchomiony,



- czerwony – zatrzymany,



- niebieski - tryb konfiguracji,



- żółty - w trakcie uruchamiania.


Konfiguracja połączenia sieciowego

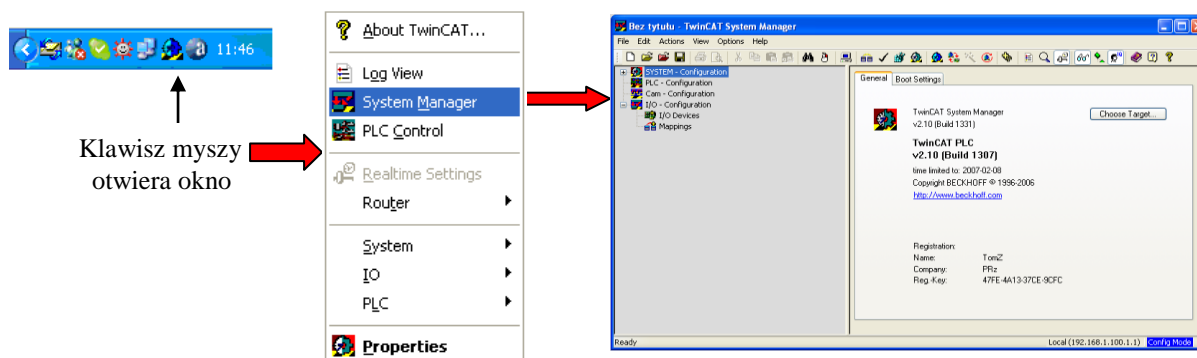
Komunikacja ze sterownikiem za pośrednictwem interfejsu Ethernet wymaga odpowiedniej konfiguracji sieci. Protokoły sieciowe sterownika oraz komputera PC z pakietem TwinCAT muszą być tak skonfigurowane, aby urządzenia znajdowały się w tej samej sieci (odpowiedni adres IP i maska podsieci – aby zweryfikować poprawność konfiguracji można wykorzystać instrukcję **ping**).

Jeżeli sterownik odpowiada na rozkaz ping, należy przejść do punktu **Konfiguracja połączenia w pakiecie TwinCAT**.

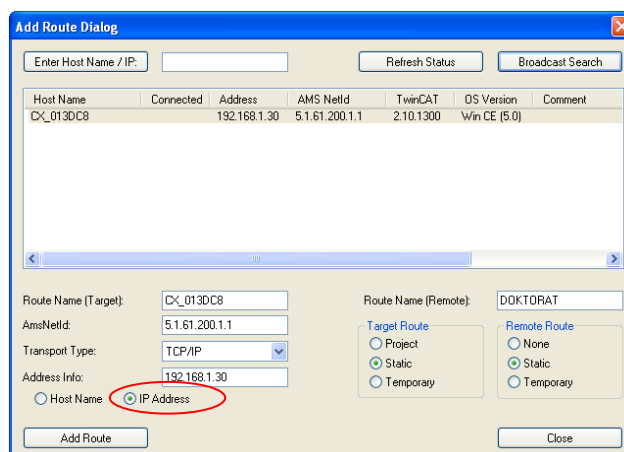
Domyślna nazwa sterownika, odczytywana po ustanowieniu połączenia, składa się z napisu CX_ oraz ostatnich trzech bajtów MAC-ID umieszczonego na sterowniku (np. CX_013DC8).

Konfiguracja połączenia w pakiecie TwinCAT

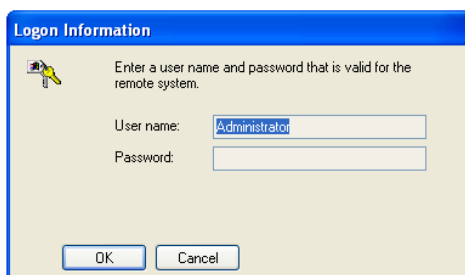
Przy pomocy ikony  znajdującej się na pasku zadań Windows, należy uruchomić pakiet **System Manager** a następnie wybrać opcję **New** z menu **File**.



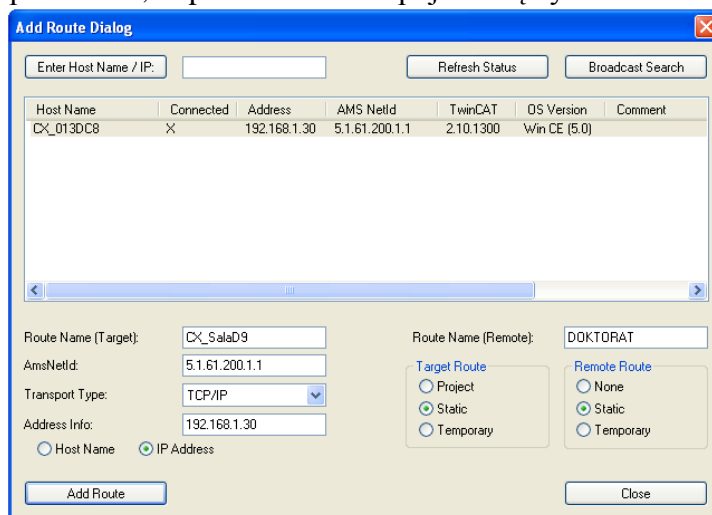
W celu zdefiniowania połączenia, w oknie **General** dla **SYSTEM-Configuration** wybrać opcję **Choose Target...**. Następnie uruchomić procedurę automatycznego wyszukiwania sterowników znajdujących się w sieci Ethernet – przycisk **Search (Ethernet)...**. W oknie **Add Route Dialog** określić typ warstwy transportowej (**Transport Type**) jako TCP/IP oraz zaznaczyć opcję IP Address (jak na poniższym rysunku) następnie uruchomić opcję wyszukiwania rozgłoszeniowego – przycisk **Broadcast Search**. Jeżeli sterownik zostanie odnaleziony, w oknie **Add Route Dialog** pojawi się odpowiedni wpis zawierający nazwę urządzenia (**Host Name**) oraz informacje o adresach IP, AMS oraz wersji urządzenia.



Brak symbolu X w polu *Connected* oznacza, że nie odbyło się logowanie do sterownika. W polu *Router Name (Target)* można dokonać zmiany nazwy sterownika (np. *CX_SalaD9*), która będzie w przyszłości identyfikowała skonfigurowane połączenie. W kolejnym kroku należy nacisnąć przycisk *Add Route*. W efekcie pojawi się okno logowania, w którym należy nacisnąć przycisk OK dla użytkownika *Administrator* bez hasła (jeżeli nie zostało wcześniej zdefiniowane).

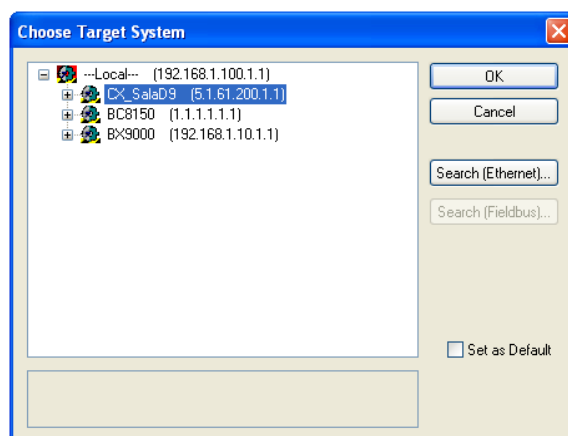


Gdy logowanie się powiedzie, w polu *Connected* pojawi się symbol X.

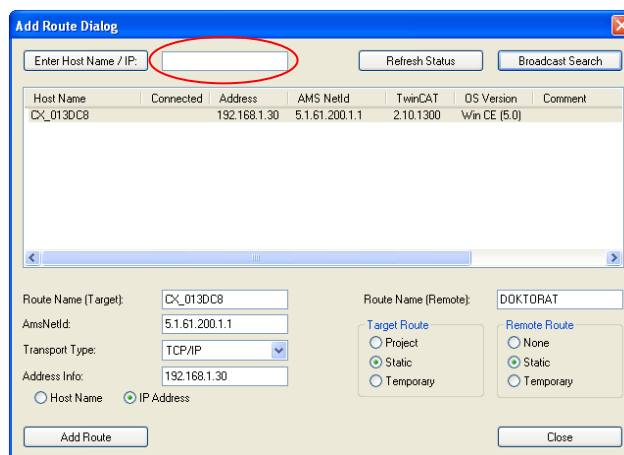


W kolejnym kroku należy zamknąć okno *Add Route Dialog* przy pomocy przycisku *Close* oraz wybrać zdefiniowane połączenie dla aktualnego projektu co spowoduje połączenie się *System Manager* ze sterownikiem. W tym celu należy zaznaczyć, przy pomocy myszki, nazwę zdefiniowanego połączenia i wybór zatwierdzić przyciskiem OK.

Od tego momentu okno pakietu *System Manager* pozwala sterować podsystemem PLC sterownika.




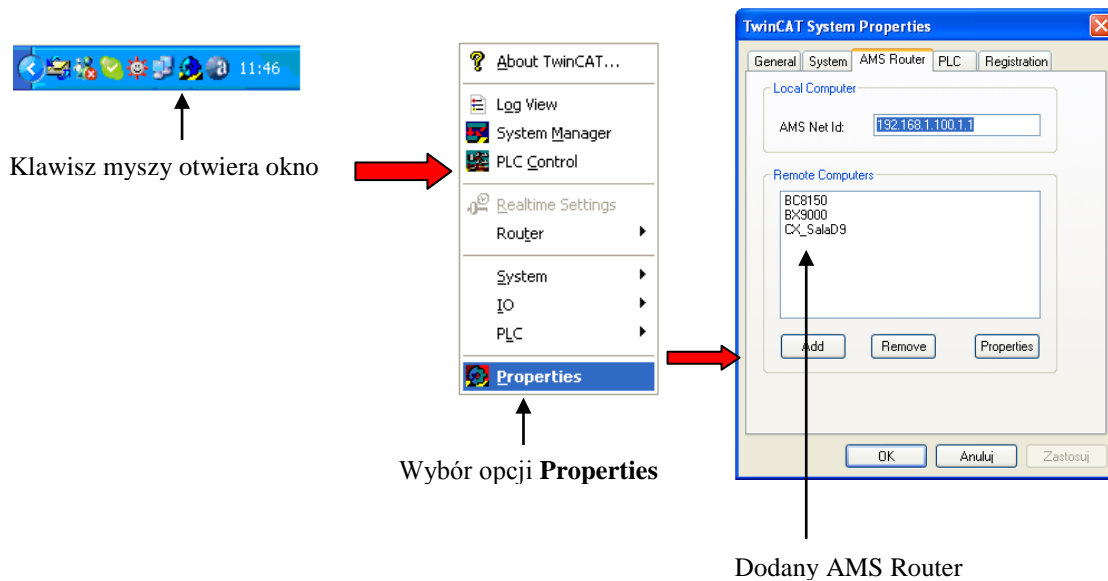
Jeżeli odpowiednie urządzenie nie zostanie wykryte, należy sprawdzić czy sterownik jest dostępny w sieci np. instrukcją *ping* i odpowiednio skonfigurować komputer PC (adresy IP, maska sieci). W przypadku gdy sterownik nie zostanie wyszukany w sieci, a odpowiada na instrukcję *ping*, możliwe jest wyszukanie sterownika bezpośrednio poprzez wpisanie jego adresu IP w polu zaznaczonym na poniższym rysunku i naciśnięciu przycisku *Enter Host Name/IP*:



Dalsze czynności należy wykonywać analogicznie jak opisano powyżej.

W przypadku, gdy komunikacja ze sterownikiem powiodła się, w systemie TwinCAT dodany zostanie obiekt AMS Router identyfikujący zdefiniowane połączenie.

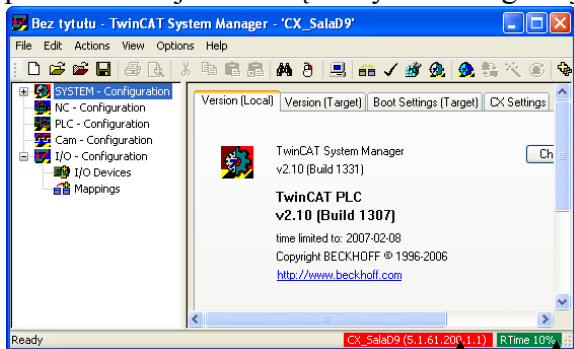
Informacje o aktualnie istniejących AMS Router, można uzyskać wykorzystując ikonę TwinCAT  znajdującą się na pasku zadań Windows i uruchamiając okno *TwinCAT System Properties*.



W oknie *Remote Computers* powinien znajdować się opis wpisany uprzednio jako nazwa sterownika. **Tak skonfigurowane połączenie może być wykorzystywane w przyszłości bez konieczności ponownego definiowania struktury komunikacyjnej.**

Po wykonaniu powyższych operacji system TwinCAT jest gotowy do współpracy ze sterownikiem CX. Aktualny tryb pracy systemu TwinCAT w sterowniku jest sygnalizowany w pasku statusu okna *TwinCAT System Manager*, jak pokazano na poniższych rysunkach.

W celu wyszukania urządzeń wchodzących w skład sterownika, system w sterowniku powinien znajdować się w trybie konfiguracji.



Nazwa połączenia

Stan połączenia (RTime – kolor zielony – połączenie aktywne)



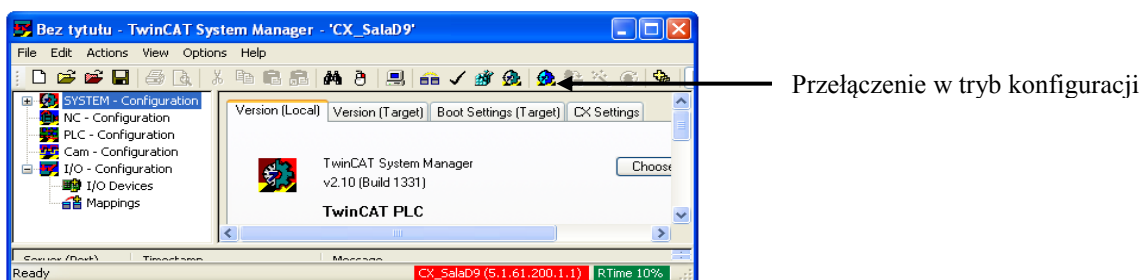
Stan połączenia (Timeout – kolor żółty – przerwa w komunikacji)



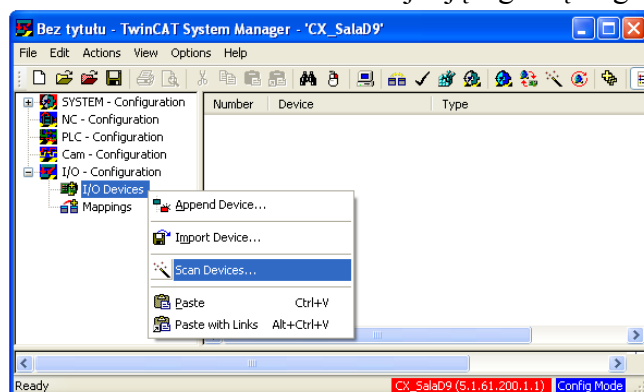
Stan połączenia (Config Mode – kolor niebieski – tryb konfiguracji)

Wykrywanie urządzeń połączonych ze sterownikiem

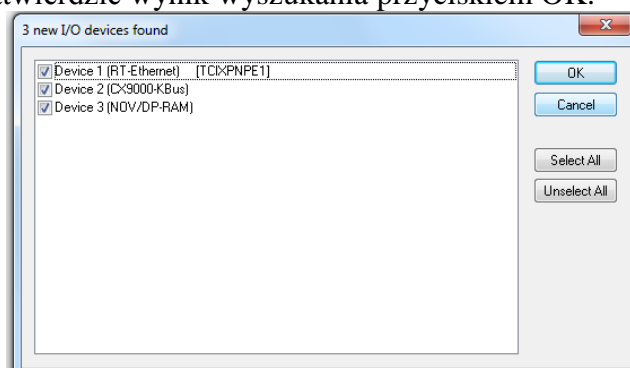
Automatyczne wykrywanie urządzeń możliwe jest w trybie konfiguracji, jeżeli *System Manager* jest w innym trybie należy dokonać przełączenia.



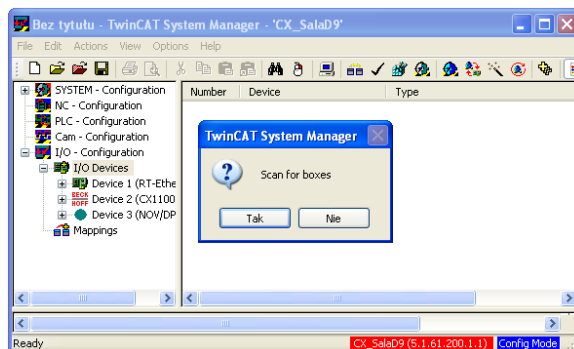
W celu wykrycia urządzeń i modułów wejścia/wyjścia, należy uruchomić (prawy klawisz myszy) funkcję *Scan Devices...* dla *I/O-Devices* znajdującego się w gałęzi *I/O-Configuration*.



Podstawowe urządzenia, które mogą zostać wykryte w zależności od typu i konfiguracji sprzętowej sterownika pokazano na poniższym rysunku. Aby wprowadzić wykryte urządzenia do projektu, należy zatwierdzić wynik wyszukania przyciskiem OK.

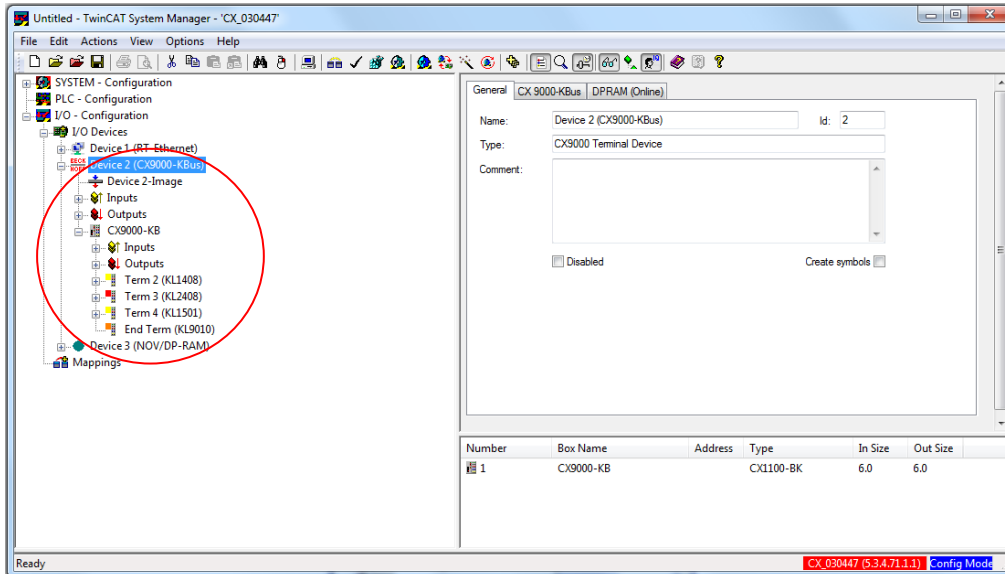


Wykryte urządzenia umieszczone zostaną w projekcie (jak pokazano na poniższym rysunku – *Device 1* do 3), dodatkowo wyświetlone zostanie okno umożliwiające wyszukanie modułów we/wy podłączonych do szyny E-bus (*Scan for boxes*). Naciśnięcie przycisku TAK rozpocznie proces wyszukiwania.

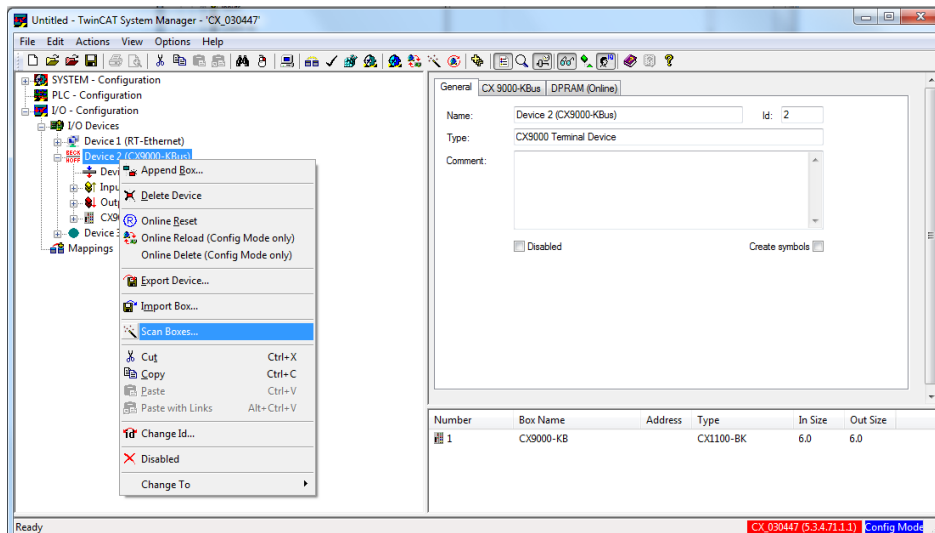


Moduły we/wy znalezione dla szyny E-bus bądź K-bus (w zależności od konfiguracji sprzętowej urządzenia) wyświetlone zostaną w odpowiedniej gałęzi projektu – przykład pokazany na poniższym rysunku.

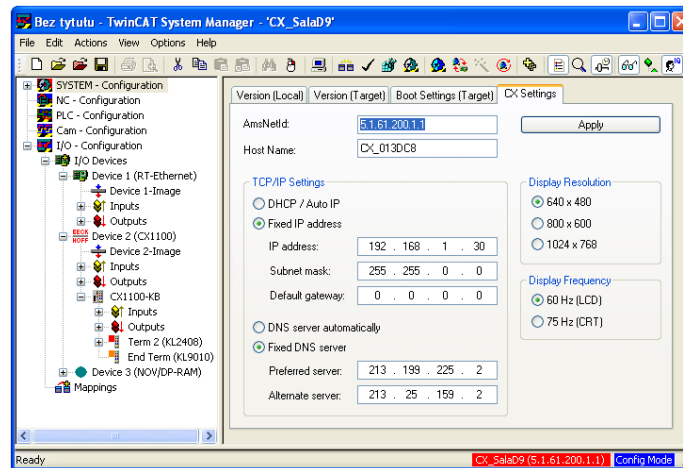
Przykładowa konfiguracja



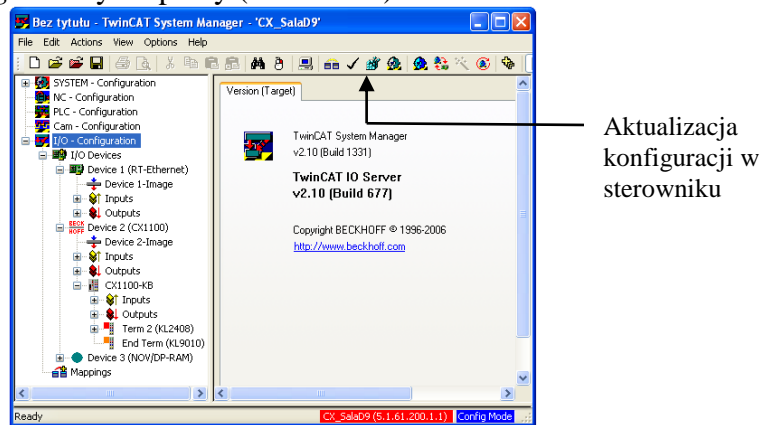
Jeżeli nie wszystkie fizycznie podłączone urządzenia zostały automatycznie wyszukane, należy uruchomić (prawy klawisz myszy) funkcję *Scan Boxes...* dla *Device 2* (np. *CX9000-EBus*) znajdującego w gałęzi *I/O Devices*.



Jeżeli zachodzi taka potrzeba, parametry połączenia sieciowego w sterowniku można zmodyfikować wybierając zakładkę *CX Settings* dla gałęzi *SYSTEM - Configuration*.

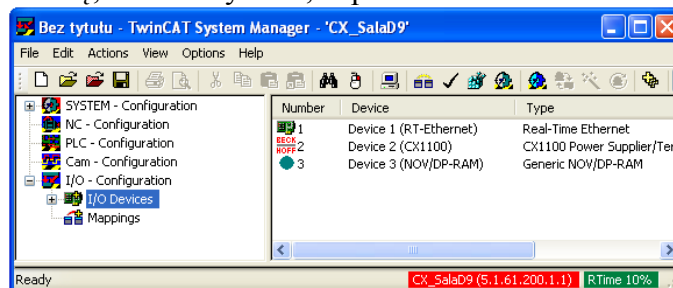


Gdy konfiguracja rzeczywista jest zgodna z automatycznie rozpoznaną, należy aktywować strukturę w sterowniku przy pomocy przycisku **Activate configuration** i uruchomić *System Manager* w trybie pracy (*Run-Mode*).



Aktualizacja konfiguracji w sterowniku

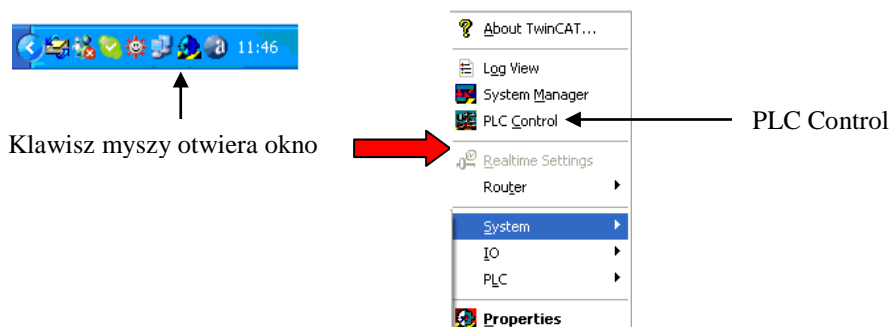
Jeżeli aktualizacja konfiguracji i uruchomienie System Manager w trybie pracy powiedzie się, na pasku statusu pojawi się, na zielonym tle, napis RTime.



W ostatnim kroku utworzony projekt należy zapisać na dysku pod wybraną nazwą. W przypadku gdy konfiguracja sprzętowa sterownika nie ulegnie zmianie, projekt może być wykorzystywany ponownie dla danego sterownika bez potrzeby powtarzania procesu konfiguracji.

3.2. Tworzenie prostego programu w języku ST – wprowadzenie

Aby utworzyć program należy uruchomić narzędzie *TwinCAT PLC Control*.



Wybór urządzenia docelowego

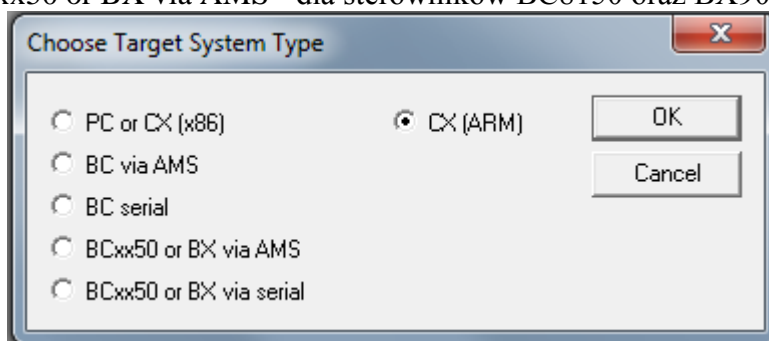
W pierwszym etapie należy określić sterownik, dla którego tworzony będzie program oraz sposób komunikacji z urządzeniem.

Z menu **File** wybrać opcję **New**, następnie określić docelową platformę w pojawiającym się oknie dialogowym.

W przypadku, gdy połączenie ze sterownikiem skonfigurowano jak w punkcie 1, jako platformę docelową należy wybrać CX(ARM).

Sprzęt dostępny w laboratoriach

- CX(ARM) dla sterowników serii CX9000 oraz panelu CP6607
- PC or CX (x86) – dla sterowników serii CX1000, CX1020
- BCxx50 or BX via AMS - dla sterowników BC8150 oraz BX9000



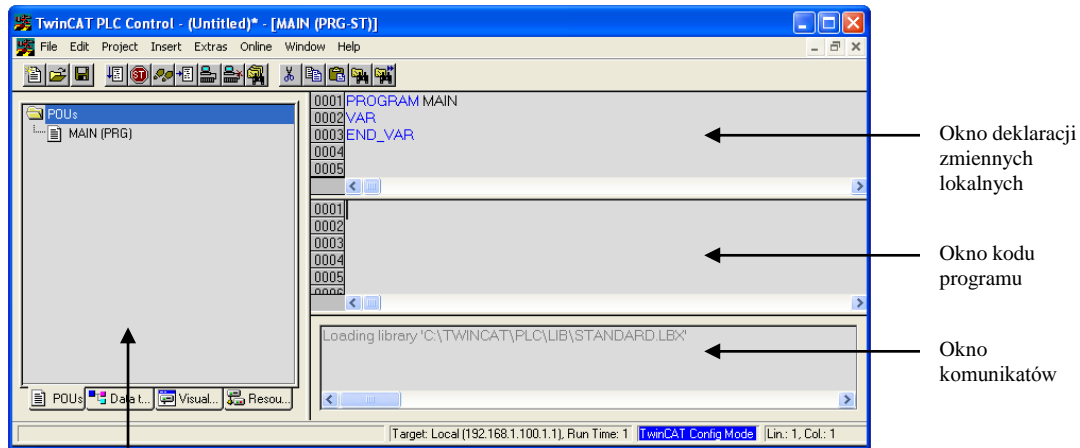
W kolejnym kroku należy określić język, w jakim będzie tworzony program. Pakiet *TwinCAT PLC Control* umożliwia programowanie w pięciu językach zgodnych ze standardem IEC 61131-3 oraz w dodatkowym języku CFC. Na tym etapie możliwe jest określenie typu obiektu (*Type of POU – Program organization unit* – jednostka organizacyjna programu) jaki będzie tworzony: program, blok funkcjonalny lub funkcja. Dla podstawowej wersji programu należy wybrać typ obiektu jako *Program*.

Przykład prostego programu w języku ST

ZADANIE. Zrealizować w języku *ST* funkcję AND dla dwóch symbolicznych zmiennych binarnych o nazwach *input1* oraz *input2* zaś wynik operacji umieścić w symbolicznej zmiennej binarnej *output*.

$$output = input1 \wedge input2$$

Okno *TwinCAT PLC Control* dla języka *ST* pokazano na poniższym rysunku.

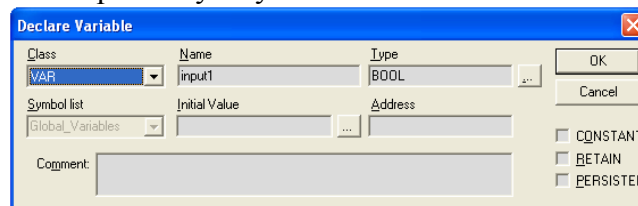


Okno organizacji projektu: programy, bloki funkcjonalne, funkcje

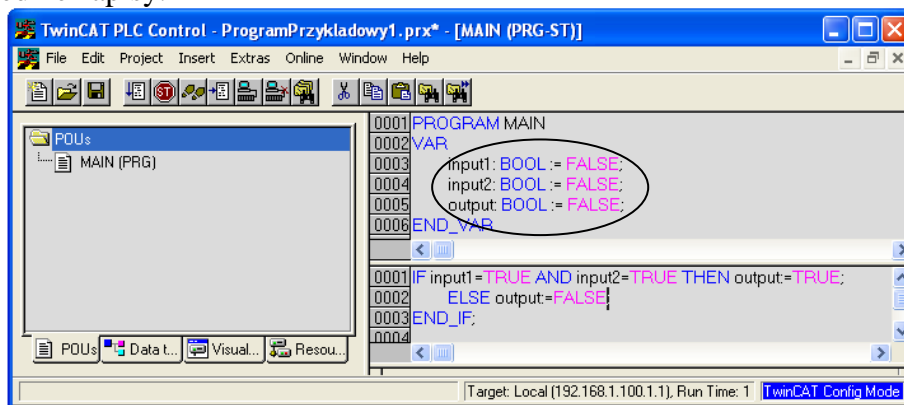
W niniejszym przykładzie wykorzystana zostanie instrukcja *IF* języka *ST*, operator koniunkcji *AND*, operator porównania '=' oraz operator przypisania ':='.
W oknie kodu programu należy wprowadzić program:

```
IF input1=TRUE AND input2=TRUE THEN output:=TRUE;
    ELSE output:=FALSE;
END_IF;
```

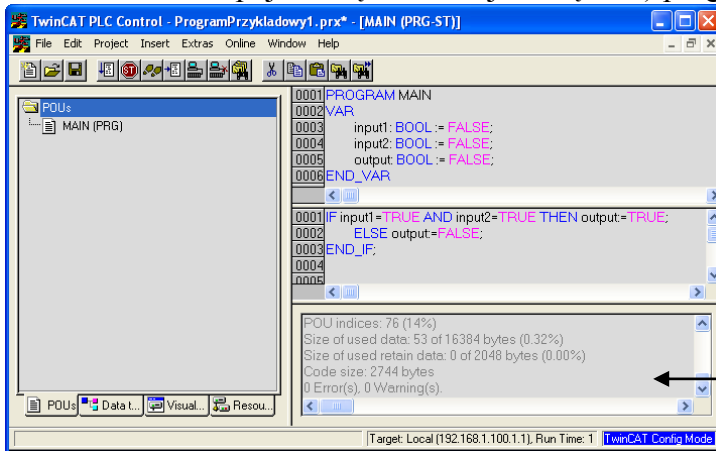
Ponieważ zmienne symboliczne *input1*, *input2* oraz *output* nie zostały wcześniej zadeklarowane, po wprowadzeniu pierwszej linii programu pojawi się okno deklaracji zmiennych pokazane na poniższym rysunku.



Dla potrzeb niniejszego przykładu dla wszystkich zmiennych należy wprowadzić *FALSE* jako wartość inicjalizującą (*Initial Value*) oraz zatwierdzić przyciskiem OK, pozostałe domyślne parametry. W obszarze deklaracji zmiennych lokalnych (rysunek poniżej) pojawi się zapis dotyczący trzech zmiennych binarnych (BOOL): *input1*, *input2* oraz *output*, którym przypisano początkowe wartości *FALSE*. Zmienne można również samodzielnie zdefiniować w oknie deklaracji zmiennych lokalnych umieszczając tam odpowiednie zapisy.

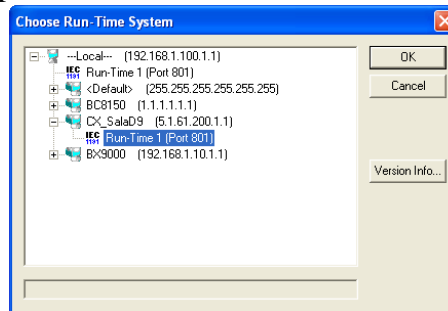


W kolejnym kroku należy zapisać program na dysku wykorzystując menu **File** i opcję **Save**. Po wprowadzeniu kodu programu należy przystąpić do jego kompilacji i konsolidacji wybierając z menu **Project** opcję **Build**. Gdy proces zakończy się sukcesem (w oknie komunikatów nie pojawi się informacja o błędach) program jest gotowy do testów.



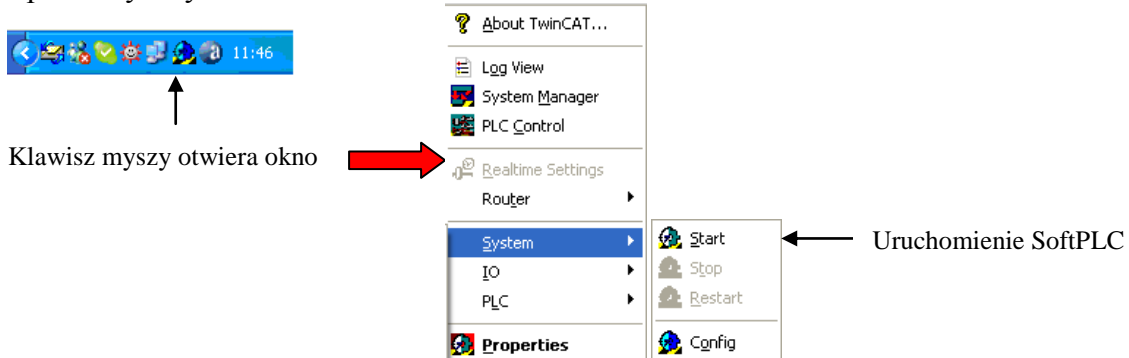
0 Error(s), 0 Warning(s)
Operacja Build zakończona sukcesem

W celu uruchomienia program, z menu **Online** wybrać platformę docelową – opcja **Choose Run-Time System...** i określić odpowiednie urządzenie. W przypadku, gdy połączenie skonfigurowano jak w punkcie 1, uruchomienie programu bezpośrednio w sterowniku CX wymaga zaznaczenia opcji np. CX-SalaD9.

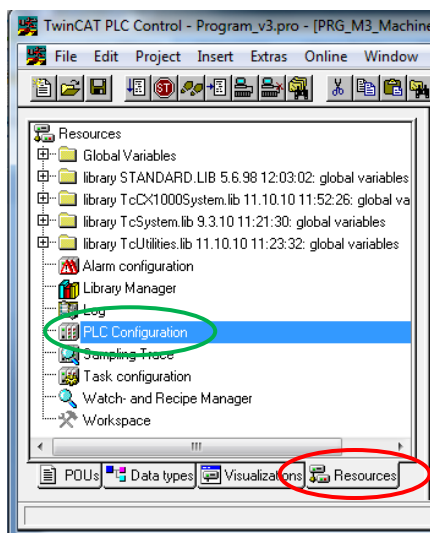


Symulacja

Jeżeli fizyczny sterownik nie jest dostępny, utworzony program może zostać przetestowany w trybie symulacji na komputerze PC przy pomocy symulatora danego typu sterownika dla BC8150 i BX9000 (pakiet *PLC Control*, pozycja menu *Online*, opcja *Simulation Mode*) bądź uruchomienia programu na komputerze PC na którym zainstalowany jest pakiet TwinCAT (*Choose Target System Type* – PC or CX (x86) i *Choose Run-Time System* - ---Local---). W celu uruchomienia systemu SoftPLC na komputerze PC należy wybrać opcję start pokazaną na poniższym rysunku.

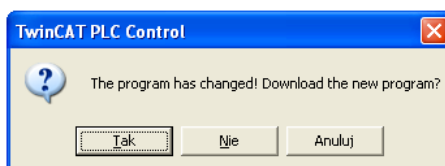


W przypadku sterowników serii CX9000 i panelu CP6607 aby uruchomić program w podsystemie SoftPLC należy zmienić platformę docelową (pakiet *PLC Control*, zakładka *Resources*, gałąź *PLC Configuration*) jak pokazano na poniższym rysunku i wykonać rekompilację projektu. Aby przetestowany program uruchomić na docelowym sterowniku należy analogicznie ponownie określić odpowiednią platformę docelową i wykonać rekompilację projektu.

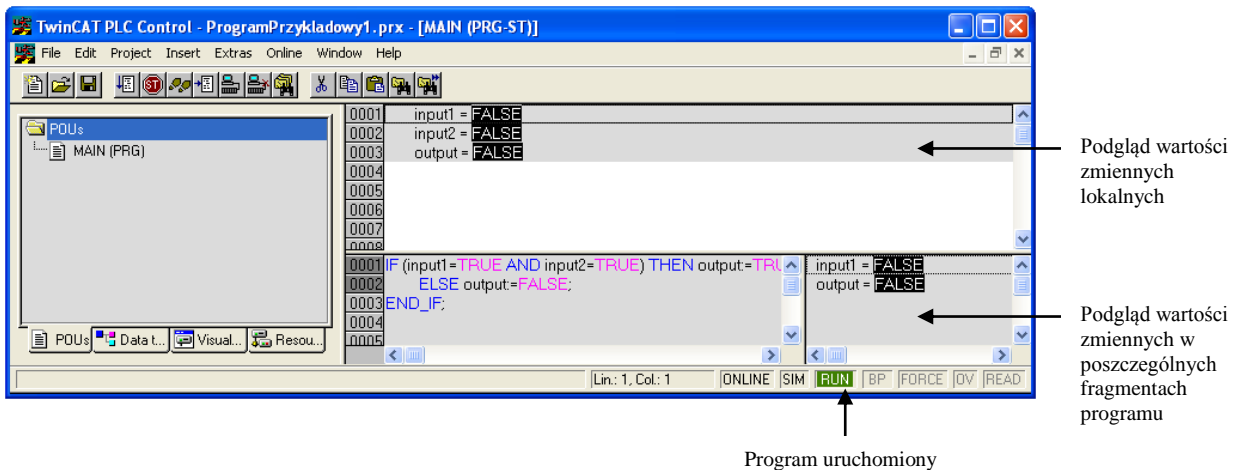


W celu zapewnienia prawidłowego działania podsystemu SoftPLC na niektórych komputerach, szczególnie typu laptop, wymagane jest uruchomienie programu *LowPrioProc.exe* udostępnianego przez firmę Beckhoff, który zapewnia dotrzymywanie parametrów czasowych sterowania.

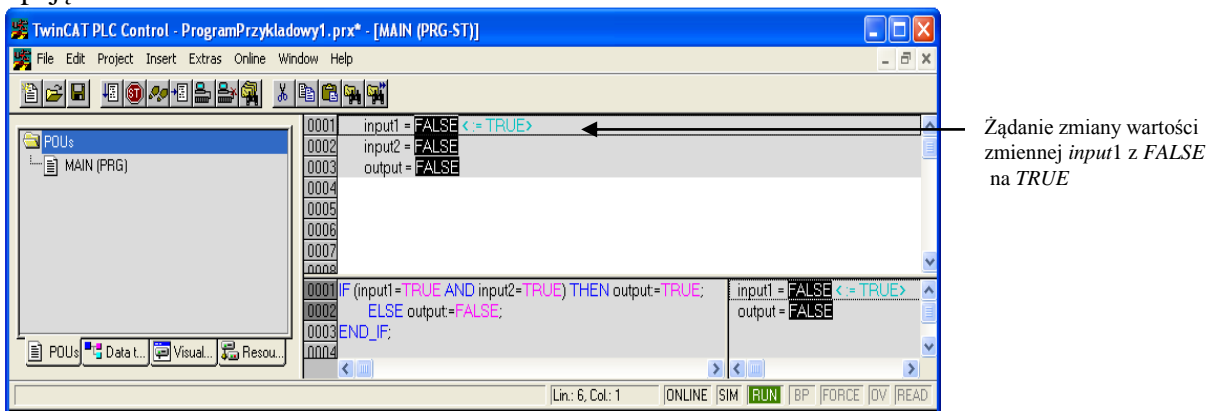
W kolejnym kroku należy zalogować się do sterownika bądź symulatora używając opcji **Login** z menu **Online**. Gdy program w sterowniku różni się od aktualnie uruchamianego, system wyświetli okno komunikatu umożliwiające zaprogramowanie sterownika nową wersją programu.



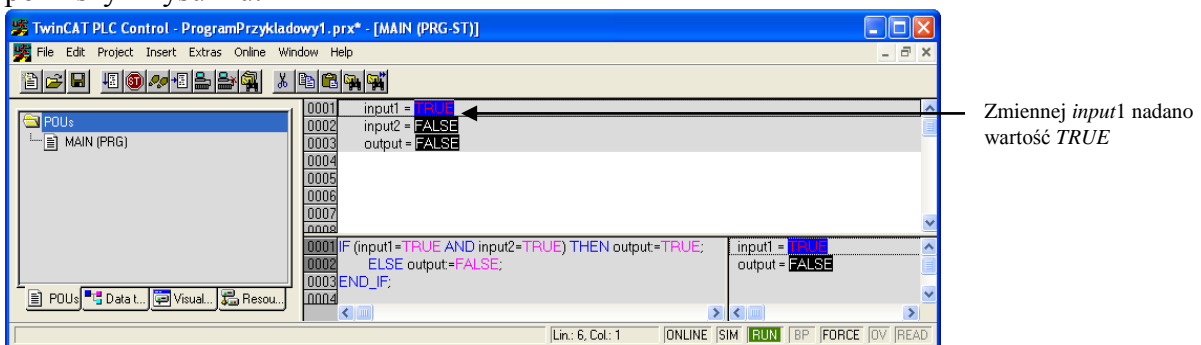
W wyniku naciśnięcia przycisku **Tak** nowy program zostanie przesłany do sterownika. W przypadku, gdy logowanie oraz przesłanie programu zakończy się sukcesem (dotyczy to zarówno pracy bezpośrednio ze sterownikiem jak i trybu symulacji) program *TwinCAT PLC Control* przechodzi do trybu podglądu działania programu. W menu **Online** uaktywniają się opcje takie jak np. **Logout** (pozwala wrócić do trybu edycji programu) oraz **Run** (uruchamia program). Aby obserwować działanie programu należy go uruchomić przy pomocy opcji **Run**, powodzenie operacji jest sygnalizowane pojawieniem się napisu *RUN* na zielonym tle w pasku statusu głównego okna *TwinCAT PLC Control*.



W celu testowania programu, wartości zmiennych mogą być modyfikowane podczas jego działania. Zmianę wartości zmiennej można zrealizować między innymi „klikając” dwukrotnie lewym przyciskiem myszki na nazwie zmiennej zarówno w oknie podglądu wartości zmiennych lokalnych, jak i w oknie podglądu wartości zmiennych w poszczególnych fragmentach kodu programu. W przypadku zmiennych logicznych (BOOL) zmiana wartości pomiędzy TRUE albo FALSE jest podpowiadana przez system automatycznie, w przypadku zmiennych innych typów pojawia się okno dialogowe umożliwiające wprowadzenie danych. Żądanie zmiany wartości zmiennej sygnalizowane jest ciągiem znaków <:=nowa wartość zmiennej> wyświetlanym obok jej nazwy. Aby wymusić wartość zmiennej należy uruchomić opcję *Force Values* z menu *Online*. Aby zapisać wartość dla jednego cyklu należy uruchomić opcję *Write Values* z menu *Online*.

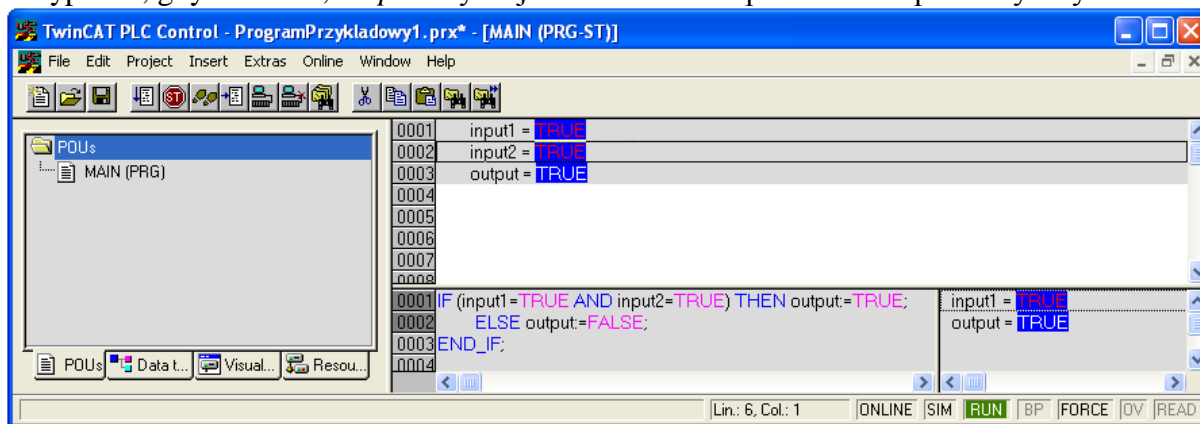


Po zatwierdzeniu zmiany (wykonanie *Force Values* bądź *Write Values*) aktualna wartość zmiennej wyświetlana jest w kolorze czerwonym na niebieskim tle jak pokazano na poniższym rysunku.



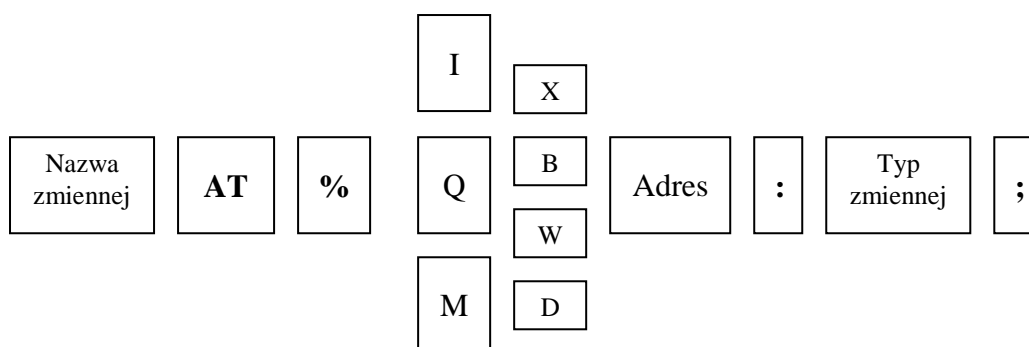
Aby sprawdzić działanie powyższego programu, wartości zmiennych *input1* i *input2* należy modyfikować zgodnie z tablicą prawdy dla koniunkcji logicznej i obserwować wartość zmiennej *output*.

Przypadek, gdy zmienna, *output* uzyskuje wartość *TRUE* pokazano na poniższym rysunku.



Powiązanie symbolicznych zmiennych z fizycznym obszarem pamięci urządzenia

Aby symboliczna zmienna z programu PLC mogła być powiązana z fizycznym wejściem, wyjściem bądź obszarem pamięci sterownika musi być zadeklarowana jako tzw. *zmienna adresowana*. W celu ulokowania zmiennej w odpowiednim obszarze pamięci (obszar wejść, wyjść, przestrzeń flag) w jej deklaracji należy użyć słowa kluczowego **AT**. Sposób deklarowania *zmiennej adresowanej* pokazano na poniższym schemacie



gdzie symbole I, Q, M określają obszar pamięci w jakim ma być ulokowana zmienna:

- I – obszar zmiennych wejściowych,
- Q – obszar zmiennych wyjściowych,
- M – obszar przestrzeni flag (fizyczny obszar pamięci sterownika o podanym adresie).

Modyfikatory X, B, W, D określają rozmiar zmiennej:

- X - bit (bit),
- B - bajt (byte - 8 bitów),
- W - słowo (word - 16 bitów),
- D - podwójne słowo (double word - 32 bity).

Adres określa fizyczną lokalizację zmiennej adresowanej w danym obszarze pamięci. W przypadku zmiennych bitowych adres jest tworzony jako ciąg liczb całkowitych bez znaku, oddzielonych kropkami np. 1.3 co oznacza bit o indeksie numer 3 (liczone od 0) w bajcie 1. Adresy zmiennych bitowych ulokowanych w przestrzeniach wejść (I) i wyjść (Q)

mogą być identyczne, gdyż są przechowywane w różnych obszarach pamięci. Dla zmiennych B, W, D - **Adres** jest reprezentowany przez liczbę całkowitą bez znaku.

Typ zmiennej – określa typ zmiennej np. BOOL, BYTE, INT, REAL, ARRAY pozwalający na prawidłową jej interpretację przez operatory danego języka programowania.

W typowych przypadkach, zmienne reprezentujące fizyczne wejścia albo wyjścia urządzenia powinny być ulokowane w odpowiednim obszarze pamięci bez dokładnego wyszczególnienia **Adresu**. Powiązanie danej zmiennej z fizycznym wyjściem/wyjściem jest realizowane w pakiecie *TwinCAT System Manager*. W omawianym przypadku **Adres** w deklaracji zmiennej zastępowany jest znakiem '*' np. `output AT %Q*:BOOL`; co oznacza, że zmienna *output* jest ulokowana w obszarze wyjść zaś jej szczegółowy adres zostanie określony w *TwinCAT System Manager*.

Aby powiązać zmienną *Output* (dla przykładowego programu opisanego powyżej) z fizycznym wyjściem urządzenia należy zmienić jej deklarację w następujący sposób

```
output AT %Q* : BOOL := FALSE;
```

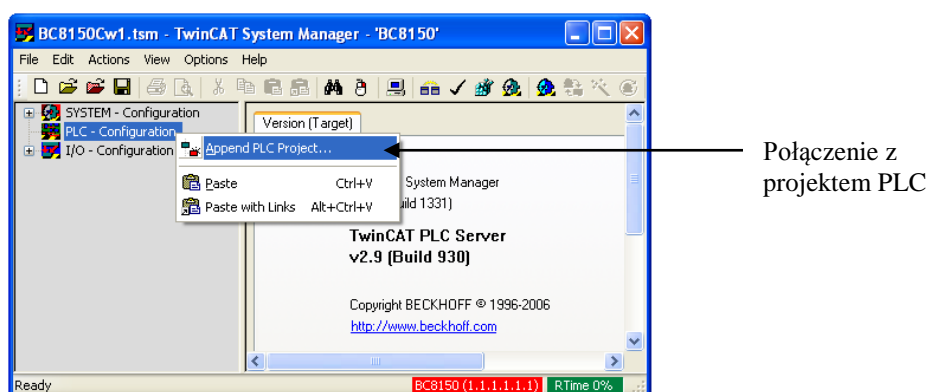
aby powiązać zmienne *input1* i *input2* z fizycznymi wejściami sterownika należy zmienić ich deklarację w następujący sposób

```
input1 AT %I* : BOOL := FALSE;
```

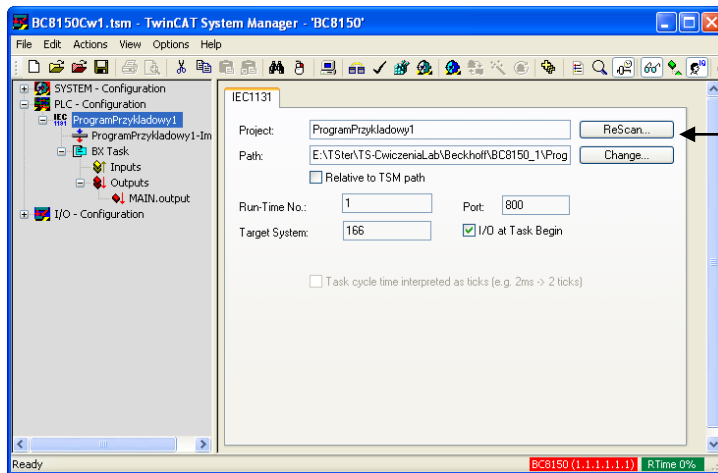
```
input2 AT %I* : BOOL := FALSE;
```

W kolejnym kroku należy dokonać ponownej kompilacji programu (menu **Project** opcja **Rebuild All**). W wyniku poprawnego utworzenia programu (w lokalizacji, w której zapisano projekt na dysku) zostanie utworzony plik z rozszerzeniem *tpy*.

Aby powiązać zmienną *output* z fizycznym wyjściem sterownika należy uruchomić projekt *TwinCAT System Manager* utworzony dla aktualnej konfiguracji sprzętowej urządzenia i połączyć z nim utworzony projekt PLC. Operacja ta jest realizowana przy pomocy opcji **Append PLC Project...**, dostępnej w *TwinCAT System Manager* dla gałęzi **PLC – Configuration** w menu pomocniczym wywoływanym przy pomocy prawego klawisza myszy.



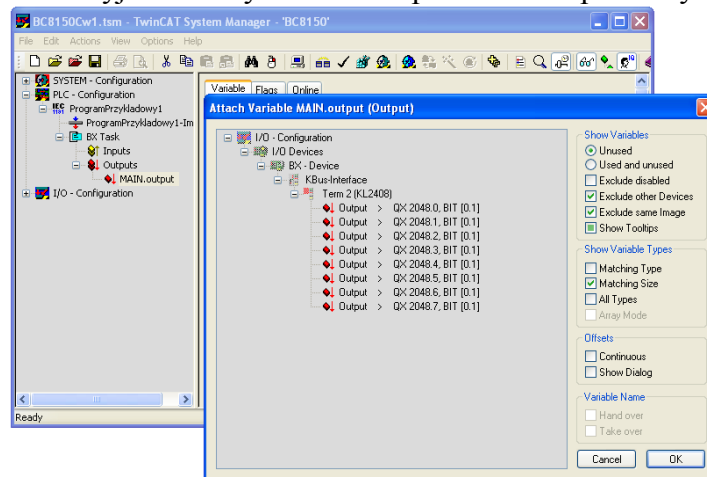
Wykonanie powyższej operacji umożliwi programowi *TwinCAT System Manager* dostęp do zmiennych adresowanych zadeklarowanych w projekcie PLC. Jak pokazano na poniższym rysunku dla rozważanego programu dostępna jest jedna zmienna o nazwie *MAIN.output*, gdzie *MAIN* określa nazwę programu w którym zmienna *output* została zadeklarowana. W przypadku modyfikacji deklaracji zmiennych adresowanych w projekcie PLC, należy w pakiecie *TwinCAT System Manager* zaktualizować dane dotyczące danego projektu PLC. Operacja ta jest realizowana przy pomocy funkcji **ReScan...**



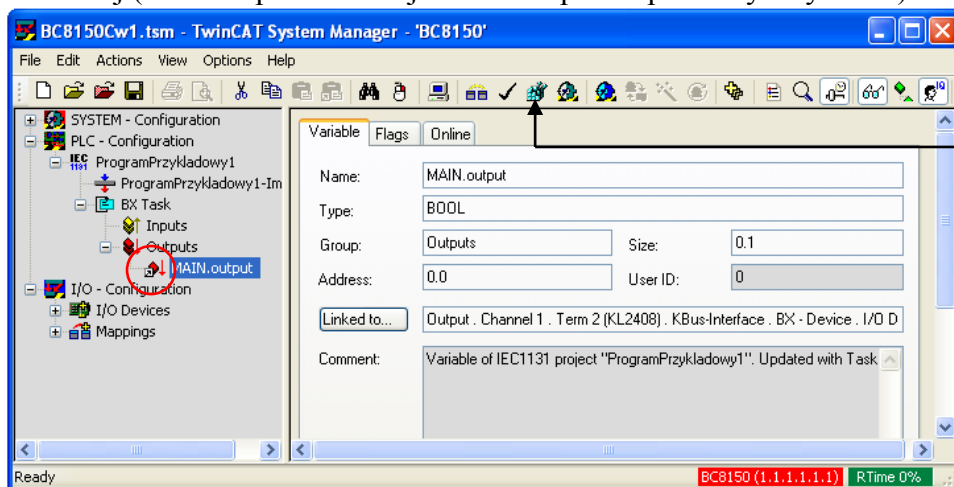
Aktualizacja projektu PLC

W wyniku dwukrotnego kliknięcia lewym klawiszem myszy na wybranej nazwie zmiennej, uzyskuje się dostęp do fizycznych wejść/wyjść urządzenia, które mogą być z daną zmienną skojarzone.

Przykładową możliwość dowiązania zmiennej *output* do fizycznych wyjść urządzenia wyposażonego w moduł wyjść binarnych KL2408 pokazano na poniższym rysunku.



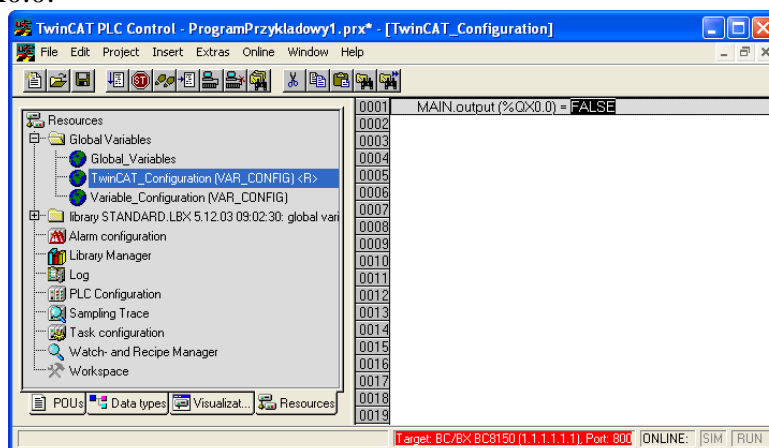
Klikając dwukrotnie lewym klawiszem myszy na wybranym wyjściu modułu KL2408 dokonujemy jego powiązania ze zmienną *output*. Informacja o powiązaniu zmiennej z fizycznym wyjściem jest sygnalizowana, między innymi, znakiem strzałki przy nazwie zmiennej (zobacz opis zmiennej *MAIN.output* na poniższym rysunku).



Aktualizacja konfiguracji w sterowniku

W kolejnym kroku utworzone powiązania należy przesłać do sterownika przy pomocy przycisku *Activate configuration*.

Po wykonaniu powyższych czynności należy powrócić do pakietu *TwinCAT PLC Control* i uruchomić program. Należy pamiętać, aby poprawnie określić system docelowy (**Choose Run-Time System...**). Po poprawnym przesłaniu programu do sterownika w zakładce *Resources* w folderze *Global_Variables* pojawi się pole *TwinCAT_Configuration*, które zawiera informacje o powiązaniach zmiennych adresowanych z fizyczną pamięcią urządzenia. W niniejszym przypadku zmienna *output* została ulokowana w przestrzeni wyjść pod adresem QX0.0.

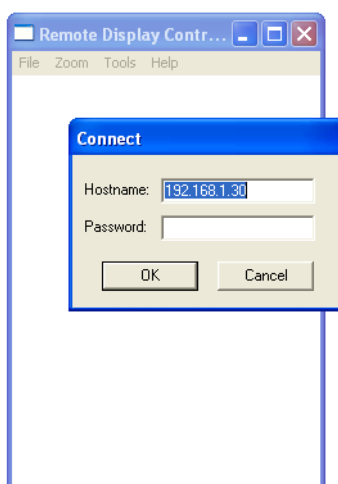


Uruchomienie i testowanie programu może być realizowane dla programu działającego w sterowniku w analogiczny sposób jak dla trybu symulacji.

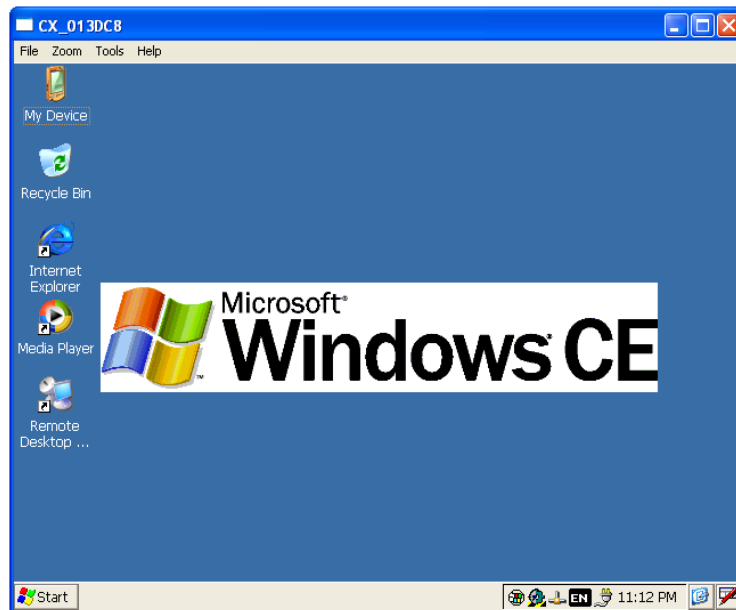
3.3. Zdalny pulpit systemu Windows CE

Program *CERHost.exe* dostępny między innymi w *Beckhoff Information System*, umożliwia podłączenie się za pomocą pulpitu zdalnego do systemu Windows CE sterownika CX.

Po uruchomieniu programu *CERHost.exe* należy wybrać opcję *Connect* z menu *File* i wprowadzić jako *Hostname* np. adres IP sterownika. Logowanie może wymagać wprowadzenia hasła.



Po poprawnym nawiązaniu komunikacji wyświetlony zostanie ekran zdalnego pulpitu sterownika CX.



Podłączając do modułu np. CX9xxx-N010 monitor (złącze DVI) oraz myszkę i klawiaturę (złącza USB) możliwe jest bezpośrednie konfigurowanie systemu Windows CE sterownika CX bez potrzeby wykorzystywania dodatkowego komputera PC i mechanizmu pulpitu zdalnego.

5. Literatura

- [1] J. Kasprzyk, *Programowanie sterowników przemysłowych*, ISBN 83-204-3109-3, WNT 2005
- [2] Wikipedia, http://pl.wikipedia.org/wiki/Sterownik_PLC
- [2]- T.Legierski, J.Kasprzyk, J.Wyrwał, J.Hajda, *Programowanie sterowników PLC*, Pracownia Komputerowa Jacka Skalmierskiego
- [3] materiały pomocnicze na stronie www.tomz.prz-rzeszow.pl (głównie: *Sterowniki_IEC61131-3.pdf*)
- [4] *Beckhoff Information System* – do pobrania ze strony www.beckhoff.pl bądź w wersji *on-line* infosys.beckhoff.com