

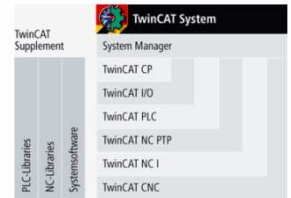


Pierwsze kroki w TwinCAT System Manager i TwinCAT PLC Control



TwinCAT System:

- ❑ TwinCAT System Manager
 - o TwinCAT CP
 - o TwinCAT I/O
 - o TwinCAT PLC
 - o TwinCAT NC PTP
 - o TwinCAT NC I
 - o TwinCAT CNC
- ❑ TwinCAT Supplement
 - o PLC – Libraries
 - o NC – Libraries
 - o Systemsoftware



TwinCAT System - kompletny system automatyki dla komputerów PC bazujący na systemie ADS, zmieniający komputer PC w kontroler czasu rzeczywistego, mogący uruchomić wiele systemów PLC, kontrolować osie NC. Umożliwia programowanie i obsługę urządzeń, pozwala odseparować programowanie od run-time systemu, pracuje w środowisku Windows (NT/2000/XP, NT/XP Embedded, CE). Obsługuje wszystkie standardy Fieldbus.

TwinCAT System Manager - centralne narzędzie konfiguracyjne i organizujące TwinCAT System (konfiguracja systemów PLC, konfiguracja osi oraz podłączonych kanałów we/wy i łączenie ich ze zmiennymi programowymi i zadaniami, zarządzanie urządzeniami podłączonymi przez Fieldbus)

TwinCAT CP - tworzy połączenie pomiędzy programami systemu Windows a funkcjami panelu kontrolnego, takimi jak: bezpośrednie przełączanie, sygnalizacja diodowa, wsparcie UPS

TwinCAT I/O - sterowanie w trybie rzeczywistym wejściami i wyjściami z poziomu systemu Windows NT/2000/XP/CE (łączenie wejść i wyjść z zadaniami, łączenie zadań z innymi zadaniami, najmniejsza jednostka – bit, obsługa wszystkich standardów Fieldbus)

TwinCAT PLC - tworzenie programów PLC (standard IEC 61131-3 {IL, LD, FBD/CFC, SFC, ST}, praca w środowisku Windows NT/2000/XP, programowanie lokalne, przez TCP/IP, przez Fieldbus, jedno narzędzie dla wszystkich platform {PC, BC, BX, CX}, rozbudowane biblioteki, wiele narzędzi pomocniczych)

TwinCAT NC PTP - sterowanie osiami w trybie point-to-point z pominięciem modułów pozycjonujących i kontrolerów NC (obliczenia wykonane przez procesor PC, połączenie z enkoderem, napędem i sterownikiem, specjalna struktura osi, zaawansowane algorytmy pozycjonowania)

TwinCAT NC I - system NC do interpolacji toru ruchu (interpolacja 3D, obsługa głównych standardów programowania CNC, sterowanie osiami z poziomu Windows NT/2000/XP, obliczenia wykonywane na procesorach PC, sterowanie osiami przez Fieldbus)

TwinCAT CNC - interpolacja osiowa w więcej niż 3 kierunkach (pełna funkcjonalność CNC z poziomu PC, praca w czasie rzeczywistym, rozszerzenie TwinCAT NC I o cechy CNC, obsługa do 32 osi, komunikacja przez wszystkie standardy Fieldbus)

.....
.....
.....
.....
.....
.....



Konfiguracja urządzeń

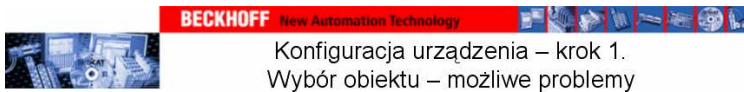
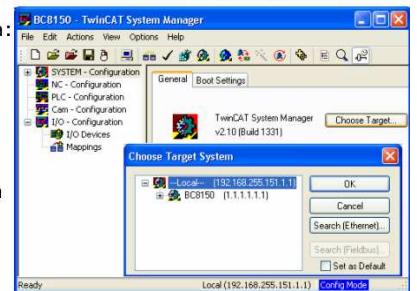
TwinCAT System Manager



Konfiguracja urządzenia – krok 1. Wybór obiektu

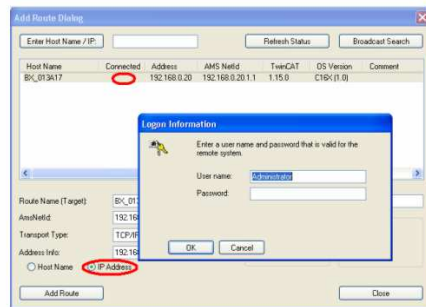
Wybór sterownika następuje w oknie Choose Target System

- Dostęp do okna wyboru sterownika:
 - ikona na pasku narzędzi
 - Actions\Choose Target System...
 - klawisz F8
 - SYSTEM-Configuration => General\Choose Target...
 - Search (Ethernet) – jeżeli nie ma szukanego sterownika



Konfiguracja urządzenia – krok 1. Wybór obiektu – możliwe problemy

- Broadcast Search – znalezienie w sieci wszystkich obiektów z uruchomionym systemem TwinCAT.
- wybieramy sterownik, z którym chcemy się połączyć.
- połączenie nawiązujemy komendą Add Route (IP Address).
- nawiązanie połączenia sygnalizuje Connected.



Powyższy przykład ilustruje wyszukiwanie sterowników komunikujących się przez połączenie **Ethernetowe** i znajdujący się w jednej sieci z komputerem, za pomocą którego łączymy się ze sterownikiem.

Jeżeli, po wciśnięciu opcji Broadcast Search, Twin CAT System Manager nie znalazł urządzenia, należy:

- sprawdzić czy sterownik jest zasilony prawidłowo,
- sprawdzić czy kabel służący do komunikacji nie jest uszkodzony,
- sprawdzić czy sterownik i komputer są w tej samej sieci (ustawienia TCP/IP)
- sprawdzić czy komputer ma wyłączoną zaporę Firewall

W przypadku urządzeń komunikujących się przez port **COM** należy ręcznie dodać urządzenie. W tym celu klikamy na ikonę **TwinCAT System Service** znajdującą się przy zegarku, wybieramy opcję **Properties**, zakładkę **AMS Router** i klikamy **Add**. Typowe ustawienia dla sterowników serii BC i BX to:

Name:	<input type="text" value="BX8000"/>	<input type="button" value="OK"/>
AMS Net Id:	<input type="text" value="1.1.1.1.1"/>	<input type="button" value="Cancel"/>
Address:	<input type="text" value="COM2:38400,e,8,1"/>	<input type="button" value="Browse"/>
Transport:	<input type="text" value="COM-PORT"/>	<input checked="" type="checkbox"/> Slow Connection



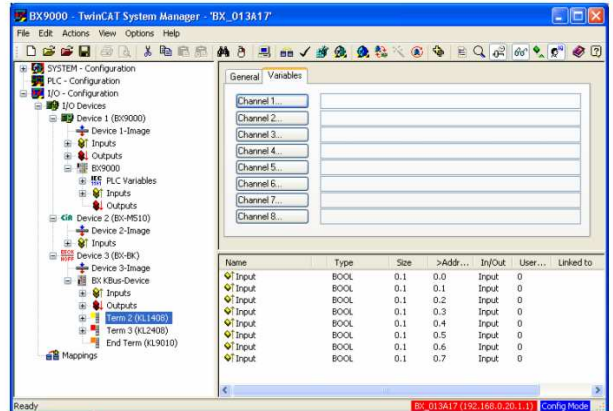
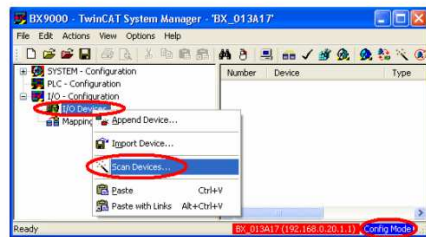
Konfiguracja urządzenia – krok 2.
Scan Devices



Konfiguracja urządzenia - zakończenie

Scan Devices:

- wykrywa urządzenia połączone ze sterownikiem
- uruchamiana po kliknięciu prawym przyciskiem myszy na I/O Devices
- aktywna w Config Mode



Luty 2007

TwinCAT PLC - Tworzenie prostego programu

6

Luty 2007

TwinCAT PLC - Tworzenie prostego programu

7

Polecenie **Scan Devices** uruchamia automatyczne wyszukiwanie urządzeń podłączonych do sterownika, na którym jesteśmy zalogowani (informację o sterowniku znajdziemy w dolnym prawym rogu, obok trybu pracy w jakim się ów sterownik znajduje). Zawsze po zakończeniu wyszukiwania należy sprawdzić zgodność konfiguracji z rzeczywistym zestawieniem sprzętowym. Drobne różnice mogą wynikać z faktu, że np. niektóre moduły zostały zastąpione przez nowsze modele, ale posiadają tę samą funkcjonalność i logikę. Aby zamienić wyszukany moduł klikamy na nim prawym przyciskiem myszy i wybieramy opcję **Change to Compatybile Type**. Jeżeli nie chcemy korzystać z wyszukiwania automatycznego, możemy zestawzić żadaną konfigurację sprzętową ręcznie poleceniami **Append Box** i **Instert Terminal Before**.

Standardowo wyszukiwane urządzenia to:

1. W przypadku sterowników serii BX:

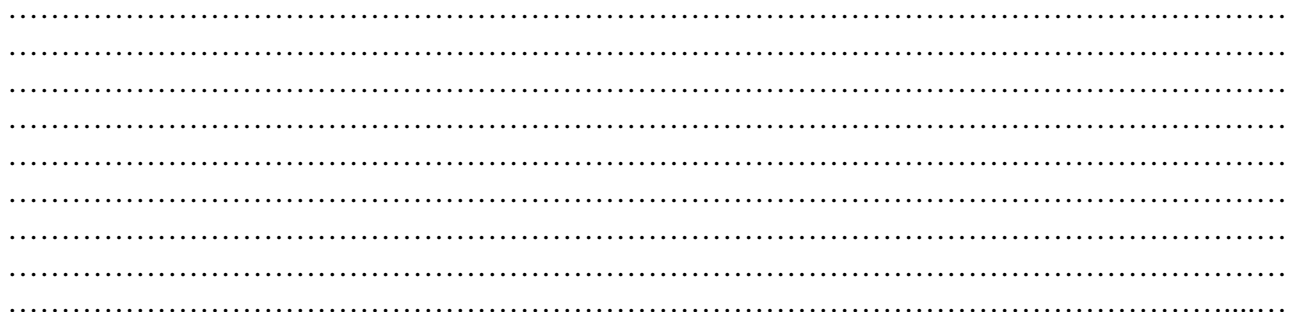
- BX – Device (BX-BK) – zasilacz z magistralą K-Bus, do którego dołączone są moduły
- Ethernet (BX9000) – interfejs ethernetowy, ustawienia protokołu TCP/IP są konfigurowalne
- SSB (BX-M510) – interfejs protokołu CANOpen

2. W przypadku sterowników serii CX:

- CX1100 - zasilacz z magistralą K-Bus, do którego dołączone są moduły
- NOV/DP-RAM – pamięć RAM przechowująca zmienne po zaniku zasilania
- RT – Ethernet – wymiana danych przez ethernet w czasie rzeczywistym

Skróconą informację o urządzeniu lub module znajdziemy klikając na interesujący nas obiekt i wybierając zakładkę **General**. Dla modułu KL 1408 wygląda ona następująco:

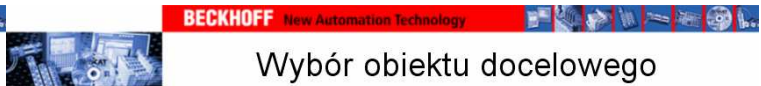
KL 1408, 8 Ch. Input (24V, 3.0ms)





TwinCAT PLC

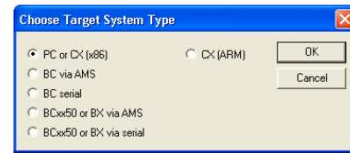
Tworzenie prostego programu



Wybór obiektu docelowego

Krok 1.

Do wyboru jest sześć typów programów, zależnych od obiektu docelowego i sposobu komunikacji.



Podstawowe biblioteki zostaną dołączone automatycznie.

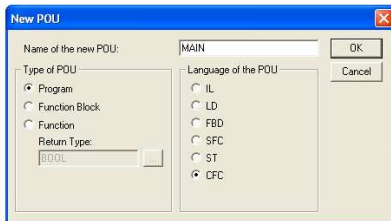


Wybór języka programowania



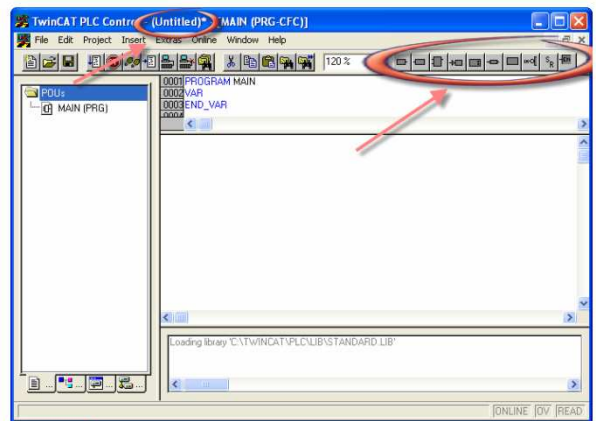
Okno główne

Krok 2.



Do wyboru jest sześć języków programowania, zgodnych ze standardem IEC 61131-3.

W oknie tym można wybrać również typ obiektu jaki zostanie stworzony: program, blok funkcyjny lub funkcja.



TwinCAT PLC pozwala na tworzenie aplikacji na wszystkie urządzenia firmy Beckhoff. Dodatkową zaletą jest fakt, że program napisany na sterowniki jednej rodziny pasuje na pozostałe, tzn. jeżeli tworząc program wybierzemy opcję PC or CX, stworzymy aplikację, a później istnieje potrzeba wgrania jej na sterownik z rodziny BX, to transformacji dokonujemy w prosty sposób. Oczywiście program będzie działał prawidłowo, jeżeli nowy sterownik ma nie mniejszą funkcjonalność. Zalecane jest jednak wybranie w oknie **Choose Target System Type** odpowiedniego obiektu docelowego.

Okno wyboru obiektu, który chcemy stworzyć pozwala wybrać:

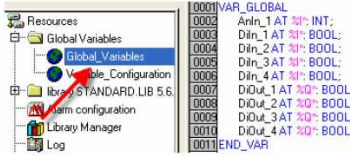
- **program** (wywołuje inne programy, funkcje i bloki funkcyjne, wywoływany przez task, przechowuje wartości zmiennych lokalnych do następnego cyklu PLC)
- **blok funkcyjny** (wywołuje inne bloki funkcyjne i funkcje, wartości zmiennych lokalnych przechowywane do następnego cyklu, może być wywołany wielokrotnie w jednym cyklu, dla każdego wywołania przypisany jest inny obszar pamięci)
- **funkcję** (nie przechowuje wartości zmiennych lokalnych, dokładnie jedno wyjście)

Deklaracja zmiennych

Krok 3. Wybór zakładki Resources



Krok 4. Przejście do Global_Variables



Krok 5. Deklaracja zmiennych:

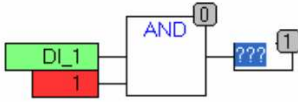
Nazwa AT %I*:Typ_Zmiennej; - zmienna wejściowa

Nazwa AT %Q*:Typ_Zmiennej; - zmienna wyjściowa

Luty 2007

Nazwanie zmiennych we/wy

Krok 7.



Na wejściu/wyjściu można wpisać:

- Zmienną globalną
- Zmienną lokalną
- Zmienną wcześniej niezadeklarowaną (domyślna autodeklaracja)
- Wartość stałą (nie da się jej zmienić podczas pracy programu!)
- Wyjście z innego bloku (np. Timer1.Q)
- Wyrażenie (np. WORD_TO_BOOL(Dane))

Luty 2007

Wybór obiektu docelowego

Krok 8.

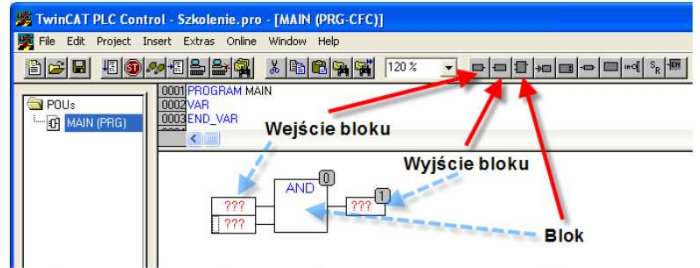


Local - program zostanie wgrany na lokalny komputer
Pozostałe obiekty to nazwy sterowników wraz z AMS Net Id

Luty 2007

Dodanie bloczka

Krok 6.



Dodawanie elementów metodą przeciągnij -> upuść

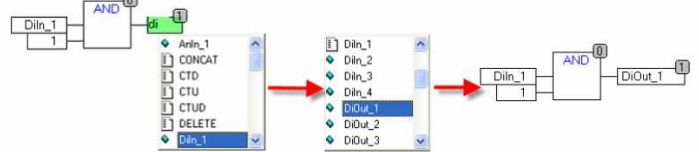
Domyślny typ bloczka to bramka AND

Widoczne wejścia/wyjścia bloczka

łatwe dodawanie wejść/wyjść

Luty 2007

Ctrl + Spacja



Skrót klawiszowy Ctrl + Spacja otwiera okno z zadeklarowanymi Zmiennymi, blokami funkcyjnymi i funkcjami

Luty 2007

Kompilacja projektu

Krok 9.



Warning 1990 -> brak obiektu docelowego dla zmiennej zaadresowanej

```
Warning 1990: No 'VAR_CONFIG' for 'DIn_2'
Warning 1990: No 'VAR_CONFIG' for 'DIn_3'
Warning 1990: No 'VAR_CONFIG' for 'DIn_4'
Warning 1990: No 'VAR_CONFIG' for 'DIOut_1'
Warning 1990: No 'VAR_CONFIG' for 'DIOut_2'
Warning 1990: No 'VAR_CONFIG' for 'DIOut_3'
Warning 1990: No 'VAR_CONFIG' for 'DIOut_4'
POU indices: 51 [2%]
Size of used data: 49 of 1048576 bytes [0.00%]
Size of used retain data: 0 of 32768 bytes [0.00%]
0 Error(s), 9 Warning(s)
0 Error(s), 0 Warning(s)
```



Luty 2007

Luty 2007

Podczas kompilacji projektu tworzony jest plik o takiej samej nazwie jak projekt, ale z rozszerzeniem *.tpy. Plik ten wczytywany jest w programie Twin CAT System Manager i to w nim linkowane są zmienne programowe ze zmiennymi we/wy.

Połączenie programu z urządzeniami

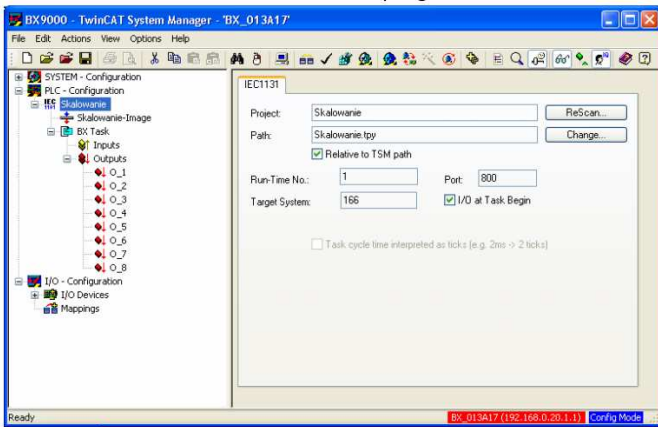
TwinCAT System Manager



Luty 2007

18

Konfiguracja urządzenia sterownik + program PLC



Luty 2007

20

Konfiguracja urządzenia – krok 5. aktywacja konfiguracji

Zakończenie konfiguracji sterownika:

- aktywowanie konfiguracji (Active configuration)
- przejście w tryb pracy (Run Mode) RTime 0%

Luty 2007

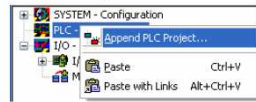
22

Zmienną z programu można zlinkować tylko z jedną zmienną wejścia/wyjścia.

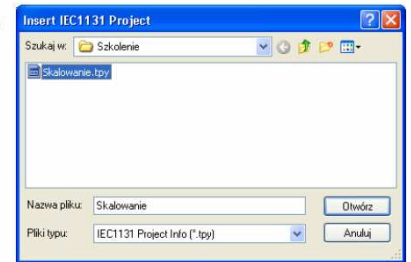
Zmienne w programie są połączone ze zmiennymi wejść/wyjść dopiero po aktywacji konfiguracji.

Konfiguracja urządzenia – krok 3. Dodanie projektu PLC

- wywołanie kliknięciem prawym przyciskiem myszy na PLC – Configuration



- otwieramy plik z rozszerzeniem *.tpy



Luty 2007

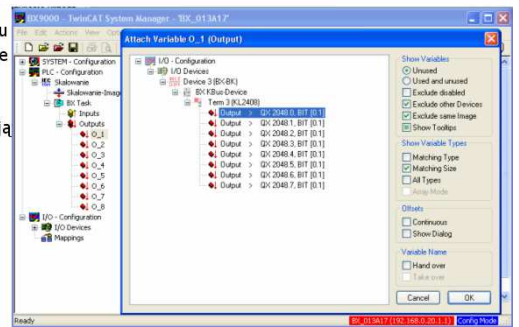
19

Konfiguracja urządzenia – krok 4. linkowanie zmiennych

- wywołanie 2-krotnym kliknięciem na zmienną we/wy w programie PLC lub kliknięciem prawym przyciskiem i wybranie opcji Change Link...

- linkowane mogą być zmienne tego samego typu
- linkowanie łączy zmienne programu PLC ze zmiennymi sprzętowymi
- zmienne zlinkowane mają symbol strzałki i są oznaczone „x”

Name	Type
I_O_1	X BOOL
I_O_2	X BOOL
I_O_3	X BOOL
I_O_4	X BOOL
I_O_5	X BOOL
I_O_6	BOOL
I_O_7	Variable
I_O_8	Variable



Luty 2007

21

Wgranie programu na sterownik

TwinCAT PLC



Logowanie/wylogowanie się

Krok 10.

Zalogowanie się: lub Login F11

Podczas logowania się w razie potrzeby wgrany jest program PLC.

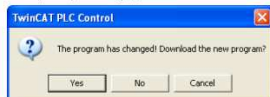
Wylogowanie się: lub Logout F12

Wgranie projektu na obiekt docelowy - komunikaty

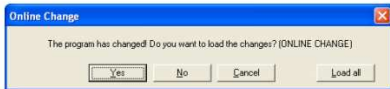
Na sterowniku nie ma programu:



Na sterowniku jest już program, jednak znacznie różni się od obecnego:



Na sterowniku jest już program, nieznacznie różniący się od obecnego, możliwa jest zmiana w trybie on-line (program PLC się nie zatrzymuje):



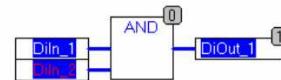
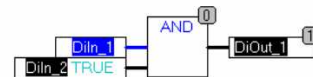
Praca w trybie on-line

Praca w trybie on-line

Uruchomienie programu (f5) lub

Target: CX 00F436 (5.0.244.54.1.1), Run Time: 1 | ONLINE: | SIM | RUN

Wpisywanie zmiennych (Ctrl + f7)



Podgląd wartości zmiennych adresowanych typu bool:

0001	AnIn_1 = 0
0002	Din_1 (%I*0.1) = TRUE
0003	Din_2 (%I*0.2) = TRUE
0004	Din_3 (%I*0.3) = FALSE
0005	Din_4 (%I*0.4) = FALSE
0006	DiOut_1 (%Q*0.0) = TRUE
0007	DiOut_2 (%Q*0.1) = FALSE
0008	DiOut_3 (%Q*0.2) = FALSE
0009	DiOut_4 (%Q*0.3) = FALSE

Nieadresowane, typ Int oraz String:

Cz_zapisu = 1
Cz_pliku = 0
Godzina = 15
Sciezka = 'c:\'
Naglowek = 'Kolumna 1'

Ważne jest, aby przed zalogowaniem się sprawdzić obiekt, na który zostanie wgrany program. Obiekt ten widoczny jest w prawym dolnym rogu okna programu TwinCAT PLC Control.

.....

Dodanie bloku

Krok 1.



Wybór typu obiektu i języka programowania



Luty 2007

Deklaracja zmiennych

Krok 2.

Var_Input - zmienne wejściowe bloczka
 Var_Output - zmienne wyjściowe bloczka
 Var - zmienne wewnętrzne bloczka

```

0000 FUNCTION BLOCK FB_Skalowanie
0002 VAR_INPUT
0003 Value : REAL;
0004 X0 : REAL;
0005 X1 : REAL;
0006 Y0 : REAL;
0007 Y1 : REAL;
0008 END_VAR
0009 VAR_OUTPUT
0010 OUT : REAL;
0011 END_VAR
0012 VAR
0013 IA : REAL;
0014 IB : REAL;
0015 END_VAR
    
```

Deklaracja zmiennych może odbywać się na bieżąco podczas pojawiania się nowych zmiennych w programie PLC

Luty 2007

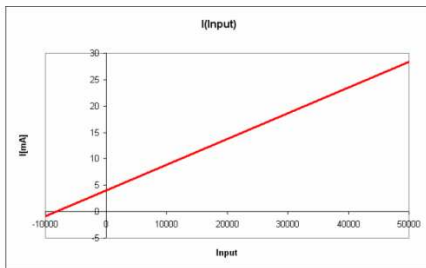
Pisanie programu (język ST)

Krok 3.

Kod:

```

fA := (Y1 - Y0) / (X1 - X0);
fB := Y1 - fA * X1;
OUT := fA * Value + fB;
    
```



WADA: Brak ograniczeń zakresu!

Luty 2007

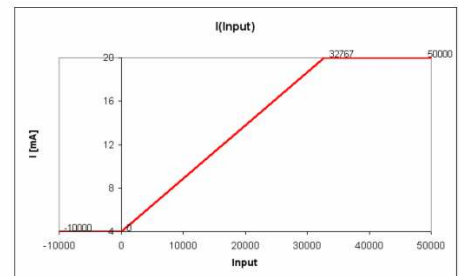
Pisanie programu (język ST)

Krok 4.

Kod:

```

0000 IF Value > X1 THEN
0001   OUT := Y1;
0002 ELSEIF Value <= X0 THEN
0003   OUT := Y0;
0004 ELSE
0005   IA := (Y1 - Y0) / (X1 - X0);
0006   IB := Y1 - IA * X1;
0007   OUT := IA * Value + IB;
0008 END_IF
    
```



WADA: Brak odporności na X1 = X2!

Luty 2007

Bloki funkcyjne posiadają zmienne:

- Var_Input - wejściowe, przenoszą wartości z obiektu, w którym blok jest wywołany, do wnętrza bloku,
- Var_Output - wyjściowe, wyprowadzają wartość z bloku do obiektu w którym jest on wywołany,
- Var_In_Out - wejścia-wyjścia, pozwalają na wykonanie operacji na zmiennych zewnętrznych bloku,
- Var - lokalne - wartość tej zmiennej nie może być odczytana poza blokiem.

Wywołanie bloczka

Krok 5.

1. Przechodzimy do programu głównego
 2. Wstawiamy bloczek (domyślnie AND)
 3. Zaznaczamy nazwę bloczka i:
- A) Wciskamy F2 i wybieramy interesujący nas bloczek



- B) Wpisujemy nazwę bloku funkcyjnego (FB_Skalowanie)



Wywołany jest dopiero blok funkcyjny, należy przypisać mu jeszcze nazwę

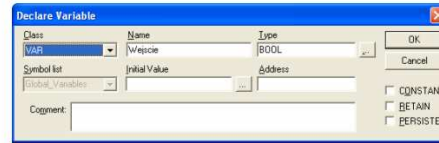
Luty 2007

35

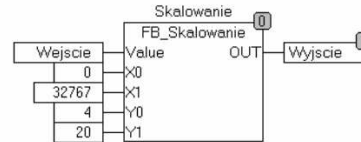
Przypisanie zmiennych we/wy

Krok 6.

Okno autodeklaracji



Gotowy bloczek skalujący zakres (0 - 32767) na (4 - 20)



Luty 2007

37

Uruchomienie programu

Krok 7.

Wpisanie nowej wartości



Wpisanie wartości – Ctrl + f7
Forsowanie wartości – f7



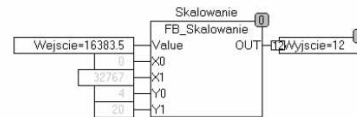
Luty 2007

38

Praca bloczka w trybie on-line

Krok 8.

Praca w trybie on-line



Wejście do bloczka:

Instances – podgląd wartości zmiennych w konkretnym bloczku

Implementation – podgląd kodu bloczka



Luty 2007

39

Podgląd wartości zmiennych w bloczku

Krok 9.

Wejście do bloczka pozwala na sprawdzenie aktualnych wartości zmiennych lokalnych w bloczku.

Wyświetlany jest również kod źródłowy bloczka.

```

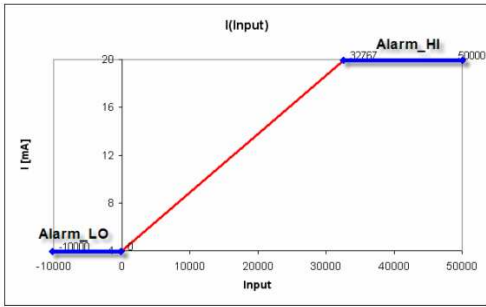
0001 IA = 4.882962e-004
0002 IB = 4
0003 Value = 16383.5
0004 X0 = 0
0005 X1 = 32767
0006 Y0 = 4
0007 Y1 = 20
0008 OUT = 12
0009
0010 IF (Value > X1) THEN
0011   OUT = Y1;
0012 ELSEIF (Value <= X0) THEN
0013   OUT = Y0;
0014 ELSE
0015   IA := (Y1 - Y0) / (X1 - X0);
0016   IB := Y1 - IA * X1;
0017   OUT := IA * Value + IB;
0018 END_IF
    
```

Luty 2007

40

Alarms

Krok 10.
Dodanie alarmu o przekroczeniu zakresu górnego lub dolnego.



Alarms

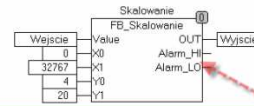
Kod:
Deklaracja zmiennych: Program:

```

0000 FUNCTION_BLOCK FB_Skalowanie
0001 VAR_INPUT
0002 Value : REAL;
0003 X0 : REAL;
0004 X1 : REAL;
0005 Y0 : REAL;
0006 Y1 : REAL;
0007 END_VAR
0008 VAR_OUTPUT
0009 OUT : REAL;
0010 Alarm_HI : BOOL;
0011 Alarm_LO : BOOL;
0012 END_VAR
0013 VAR
0014 IA : REAL;
0015 IB : REAL;
0016 END_VAR
0017 END_FUNCTION_BLOCK

0000 IF (Value > X1) THEN
0001   OUT := Y1;
0002   Alarm_HI := TRUE;
0003   Alarm_LO := FALSE;
0004 ELSEIF (Value <= X0) THEN
0005   OUT := Y0;
0006   Alarm_HI := FALSE;
0007   Alarm_LO := TRUE;
0008 ELSE
0009   IA := (Y1 - Y0) / (X1 - X0);
0010   IB := Y1 - IA * X1;
0011   OUT := IA * Value + IB;
0012   Alarm_HI := FALSE;
0013   Alarm_LO := FALSE;
0014 END_IF
    
```

Zmiany w bločku w programie głównym:



Alarms

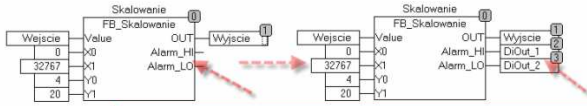
Kod:
Deklaracja zmiennych: Program:

```

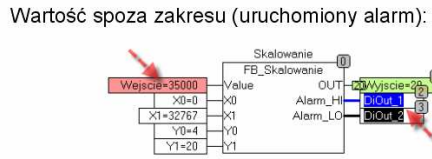
0000 FUNCTION_BLOCK FB_Skalowanie
0001 VAR_INPUT
0002 Value : REAL;
0003 X0 : REAL;
0004 X1 : REAL;
0005 Y0 : REAL;
0006 Y1 : REAL;
0007 END_VAR
0008 VAR_OUTPUT
0009 OUT : REAL;
0010 Alarm_HI : BOOL;
0011 Alarm_LO : BOOL;
0012 END_VAR
0013 VAR
0014 IA : REAL;
0015 IB : REAL;
0016 END_VAR
0017 END_FUNCTION_BLOCK

0000 IF (Value > X1) THEN
0001   OUT := Y1;
0002   Alarm_HI := TRUE;
0003   Alarm_LO := FALSE;
0004 ELSEIF (Value <= X0) THEN
0005   OUT := Y0;
0006   Alarm_HI := TRUE;
0007   Alarm_LO := FALSE;
0008 ELSE
0009   IA := (Y1 - Y0) / (X1 - X0);
0010   IB := Y1 - IA * X1;
0011   OUT := IA * Value + IB;
0012   Alarm_HI := FALSE;
0013   Alarm_LO := FALSE;
0014 END_IF
    
```

Zmiany w bločku w programie głównym (sygnalizacja diodowa):



Praca w trybie on-line



Podane jest też napięcie na odpowiednie wyjście cyfrowe (sygnalizowane jest to zapaleniem się diody)

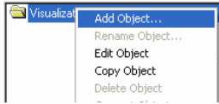
Zmienne wyjściowe z bloków funkcyjnych mogą nie zostać wykorzystane w programie, nie powoduje to wystąpienia błędu. Błędem jest nieprzypisanie zmiennej do wejścia bločka.

Wizualizacja

Obiekty wizualizacji znajdują się na zakładce Visualizations



Dodanie nowego obiektu



Nadanie nazwy



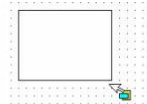
Luty 2007

Element wyświetlający wartość

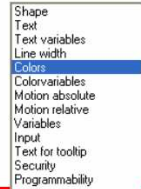
Wybór elementu



Wstawienie elementu



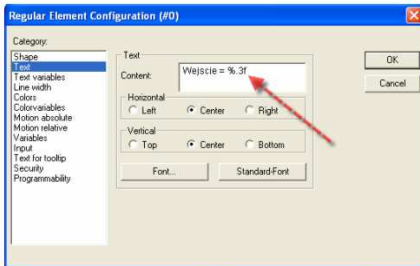
Ustawienie parametrów elementu



Luty 2007

Wstawienie tekstu

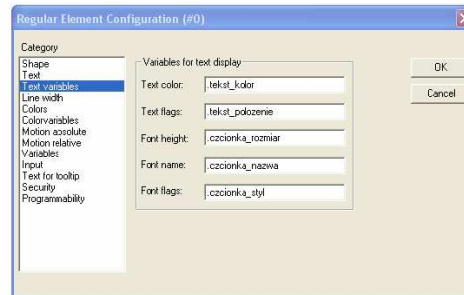
Tekst wyświetlany na elemencie (przycisku, polu, itp.)



%3f – wyświetlenie zmiennej typu Real z trzema miejscami po przecinku
%s – zmienna wyświetlana jako napis

Luty 2007

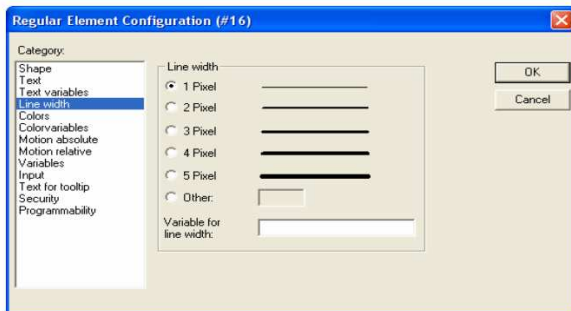
Edycja tekstu



Każda cecha wyświetlanego tekstu może być kontrolowana przez zmienną w programie

Luty 2007

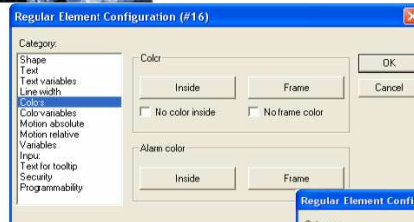
Edycja linii



Każda cecha wyświetlanego tekstu może być kontrolowana przez zmienną w programie

Luty 2007

Kolory



Kolory „statyczne” – nadawane tylko raz

Kolory dynamiczne – zależne od wartości zmiennych

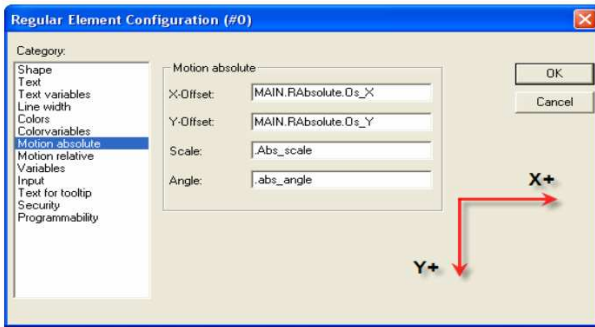


Luty 2007

Wizualizacja może być uruchomiona bezpośrednio na sterownikach serii CX z zainstalowanym Target Visu. Na wszystkich pozostałych sterownikach może być dokonywany podgląd aplikacji poprzez wizualizację uruchomioną na lokalnym komputerze.

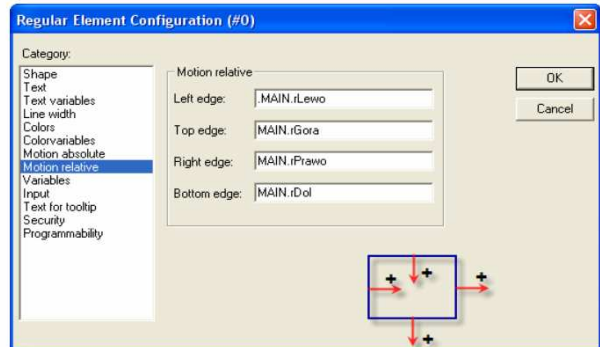
Wykorzystywane zmienne są zmiennymi z programu, nie trzeba ich specjalnie adresować lub deklarować.

Ruch elementu



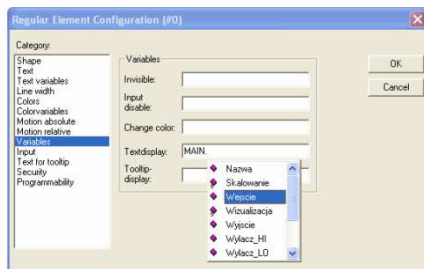
Wartość 0 to pozycja elementu w trybie offline.
Kontrola ruchu elementu może odbywać się poprzez zmienne typu Int
Dokładność 1 piksel.

Ruch krawędzi elementu



Dynamiczna zmiana położenia krawędzi pozwala na kontrolę wymiarów elementów w wizualizacji za pomocą zmiennych.

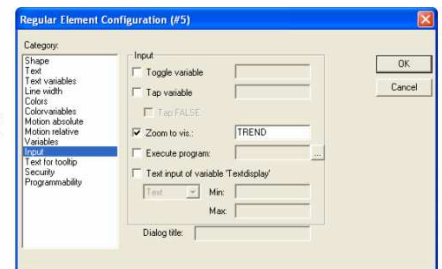
Własności elementu



- Ustawienie:
- znikania,
 - blokowania,
 - zmiany koloru,
 - wyświetlanego tekstu
 - podpowiedzi w trybie on-line

Main.Wejscie – odwołanie do zmiennej zadeklarowanej w programie głównym
.DiOut_1 – odwołanie do zmiennej globalnej

Akcje



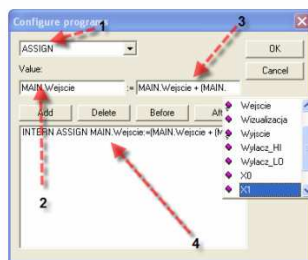
Akcja wykonuje się podczas naciśnięcia obiektu.

- Możliwe akcje:
- Przełączenie zmiennej
 - Chwilowe włączenie zmiennej
 - Zmiana wizualizacji
 - Wykonanie programu
 - Zmiana wartości

Main.Wejscie – odwołanie do zmiennej zadeklarowanej w programie głównym
.DiOut_1 – odwołanie do zmiennej globalnej

Możliwe jest ograniczenie zakresu wpisywanych wartości i wybór sposobu ich wpisywania.

Wykonanie programu

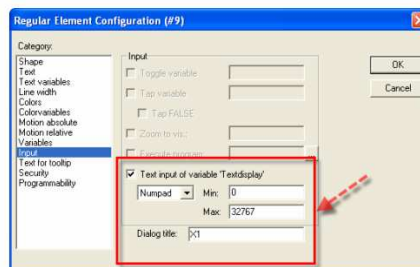


- W oknie wykonania programu:
- wybieramy typ programu
 - wybieramy zmienną, na której operacji dokonujemy
 - wpisujemy działanie jakie wykonujemy
 - dodajemy operację
 - wybieramy kolejną operację lub wracamy do okna głównego

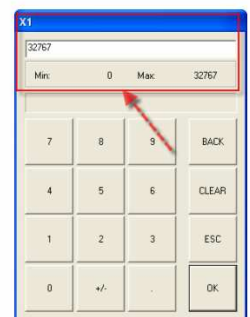
W programie możemy wykorzystywać zmienne wcześniej zdefiniowane, wartości stałe oraz wszystkie operacje matematyczne.

Wprowadzenie wartości

- Możliwy wybór:
- sposobu wprowadzenia danych,
 - zakresów,
 - nazwy dla pojawiającego się obiektu.



Widok on-line

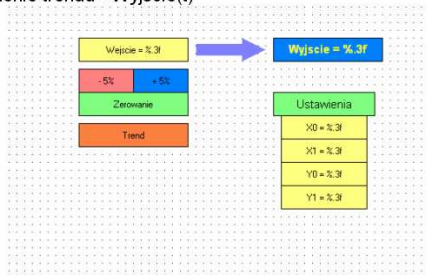


Dodanie akcji, zmiana własności elementów sprawia, że tworzone w wizualizacji elementy mogą się zmieniać podczas działania aplikacji, dzięki czemu możliwa jest m.in. animacja obiektów.

Wiz – strona 1 (off-line)

Wejscie -> Wyjscie:

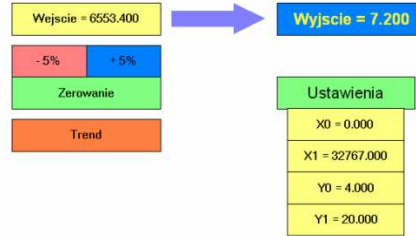
- Dodanie / odjęcie 5% wartości zakresu
- Zerowanie wejścia,
- Zmiana zakresów
- Wyświetlenie trendu - Wyjscie(t)



Wiz – strona 1 (on-line)

Wejscie -> Wyjscie:

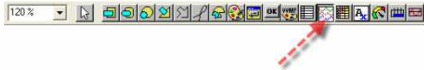
- Dodanie / odjęcie 5% wartości zakresu
- Zerowanie wejścia,
- Zmiana zakresów
- Wyświetlenie trendu - Wyjscie(t)





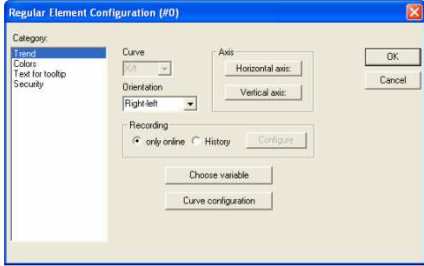
Trend – strona 2

Dodanie elementu:



Trend:

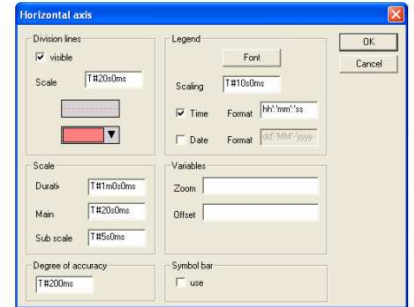
- Kierunek zmian,
- Konfiguracja obu osi,
- Wybór zmiennej
- Konfiguracja krzywej



Trend – strona 2

Oś pozioma:

- Jednostki - czas,
- Linie podziałowe,
- Skala,
- Odświeżanie,
- Opis osi.



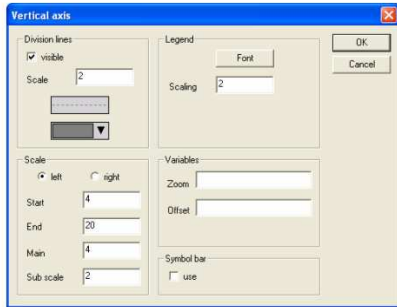
Trend – strona 2



Trend – strona 2

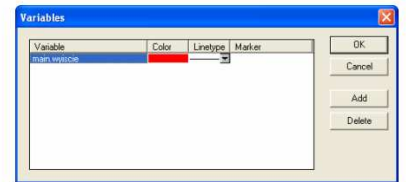
Oś pionowa:

- Jednostki - dowolne,
- Linie podziałowe,
- Skala,
- Opis osi.



Zmienne:

- Wybór zmiennej,
- Kolor na wykresie,
- Styl linii,
- Znacznik.



Trend – strona 2 (off-line)



Trend – strona 2 (on-line)

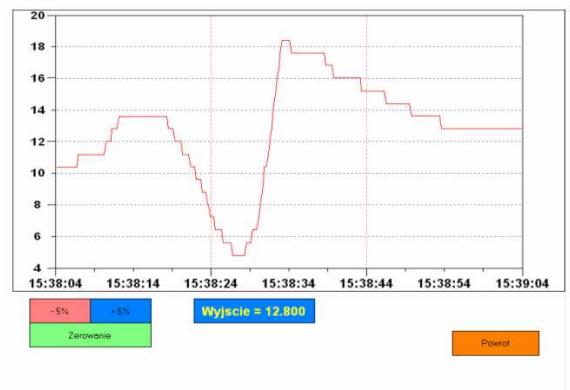
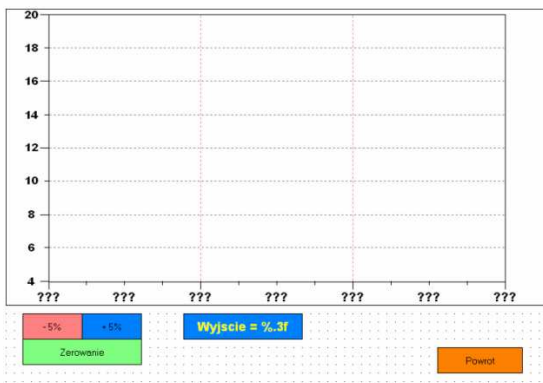


Tabela – wstawienie elementu

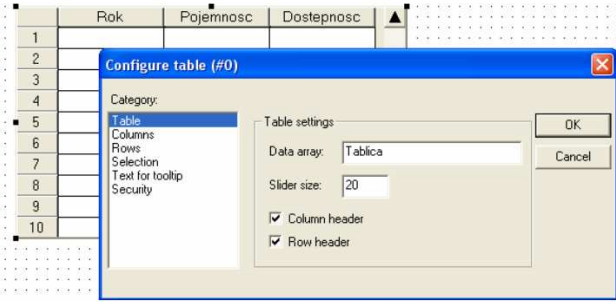
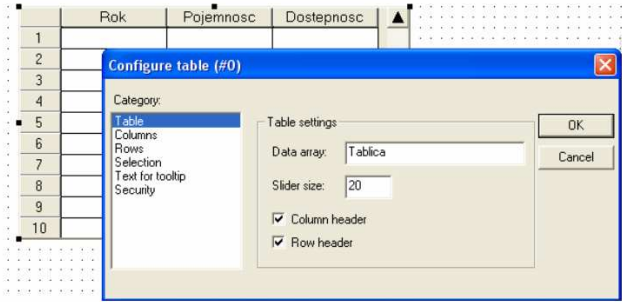


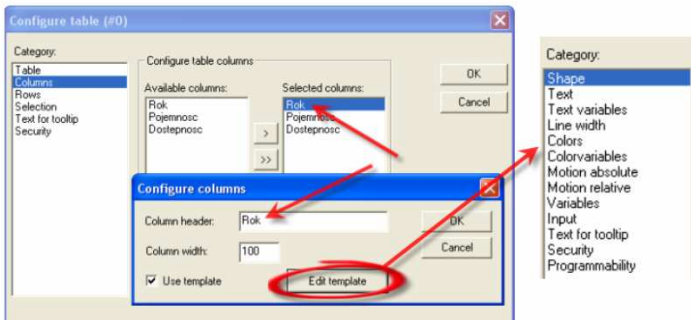
Tabela – wstawienie elementu



Luty 2007

65

Tabela – konfiguracja komórek



Parametry komórek mogą być konfigurowane tak samo jak każdego elementu w wizualizacji

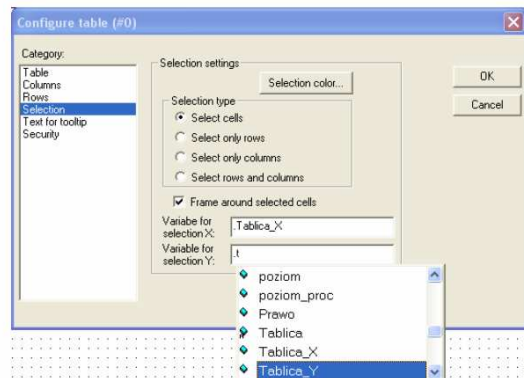
Luty 2007

66

Luty 2007

65

Tabela – zaznaczanie elementu



Zdefiniowanie zmiennych wskazujących na element tablicy pozwala np. na śledzenie obróbki

Luty 2007

67