

Cscape Instrukcja obsługi i programowania



Niniejsza instrukcja zawiera opis oprogramowania Cscape firmy Horner APG.

Wszelkie prawa zastrzeżone.

Żaden fragment niniejszego dokumentu nie może być kopiowany, przesyłany, kodowany, przechowywany w systemach wyszukiwania informacji, tłumaczony na inne języki za pomocą metod elektronicznych, mechanicznych, magnetycznych, optycznych, ręcznie lub w jakikolwiek inny sposób, bez uprzedniego otrzymania pisemnej zgody firmy ASTOR Sp. z o.o.

Oprogramowanie opisywane w niniejszej dokumentacji jest zastrzeżone prawami podanymi w Umowie Licencyjnej.

Informacje zawarte w niniejszym dokumencie mogą zostać zmienione bez wcześniejszego uprzedzenia i nie mogą być traktowane jako zobowiązanie ze strony firmy Horner APG.

Cscape, SmartStack i CsCAN to zastrzeżone nazwy towarowe Horner APG. DeviceNet to zastrzeżona nazwa towarowa Open DeviceNet Vendor Association (OVDA), Inc.

## **Oświadczenie Producenta oprogramowania: firmy Horner Electric APG o ograniczonej gwarancji i odpowiedzialności**

Horner APG, LLC. ("HE-APG") gwarantuje pierwotnemu nabywcy, że oprogramowanie Cscape nie posiada wad materiałowych i wykonawczych pod warunkiem, że będzie wykorzystywane i serwisowane zgodnie z zaleceniami

Zobowiązania HE-APG w ramach niniejszej gwarancji ograniczone są do naprawy lub wymiany elementu lub elementów, które uległy uszkodzeniu w trakcie normalnej eksploatacji lub serwisowania, w ciągu dwóch (2) lat od daty zakupu lub w ciągu osiemnastu (18) miesięcy od daty zainstalowania przez pierwotnego nabywcę, w zależności który z tych terminów jest wcześniejszy, pod warunkiem, że HE-APG zostanie poinformowany o takim uszkodzeniu i że zostanie ono zaakceptowane.

GWARANCJA TA ZASTĘPUJE WSZYSTKIE POPRZEDNIE GWARANCJE WYRAŻONE W FORMIE JAWNEJ LUB NIEJAWNEJ, WŁĄCZAJĄC W TO GWARANCJĘ SPRZEDAwalności I PRZYDATNOŚCI DO KONKRETNEGO CELU. HE-APG NIE PONOSI, ANI TEŻ NIE UPOWAŻNIA ŻADNEJ INNEJ STRONY TRZECIEJ DO ZOBOWIĄZYWANIA SIĘ W IMIENIU HE-APG DO PONOSZENIA JAKIEJKOLWIEK INNEGO TYPU ODPOWIEDZIALNOŚCI W ZWIĄZKU ZE SPRZEDAŻĄ OPROGRAMOWANIA CSCAPE. POWYŻSZA GWARANCJA TRACI WAŻNOŚĆ JEŻELI OPROGRAMOWANIE CSAPE LUB JEGO FRAGMENT ZOSTAŁY USZKODZONE, BYŁY NIEWŁAŚCIWIE UŻYWANE, MODYFIKOWANE LUB WYKORZYSTYWANE NIEZGODNIE Z PRZEZNACZENIEM. HE-APG NIE UDZIELA ŻADNEJ GWARANCJI NA DODATKOWE AKCESORIA LUB ICH ELEMENTY SKŁADOWE, NIE DOSTARCZANE PRZEZ HE-APG. TERMIN "PIERWOTNY NABYWCA", UŻYWANY PRZY UDZIELANIU NINIEJSZEJ GWARANCJI OZNACZA OSOBĘ, U KTÓREJ ZAINSTALOWANO PO RAZ PIERWSZY OPROGRAMOWANIE CSAPE PO JEGO DOSTARCZENIU.

W żadnym wypadku, bez względu na fakt, czy naruszone zostały postanowienia kontraktu, gwarancji lub z jakiegokolwiek innego tytułu, zarówno HE-APG jak i przedstawiciele tej firmy nie ponoszą jakiegokolwiek odpowiedzialności za celowe, powstałe w konsekwencji lub przypadkowe szkody, włączając w to między innymi, utratę zysków lub zmniejszenie przychodów, brak możliwości korzystania z innych produktów lub dowolnych, współpracujących z nimi urządzeń, uszkodzenie majątku trwałego, koszty produktów zastępczych, mediów, serwisowania lub wymiany zasilania, koszty przerw produkcyjnych lub żądania odszkodowania wysuwane przez klientów pierwotnego nabywcy.

Warunkiem uwzględnienia reklamacji gwarancyjnej jest dostarczenie produktu do dystrybutora wraz z opisem zaistniałej nieprawidłowości, dowodem zakupu i w odpowiednim opakowaniu, z opłaconymi kosztami transportu i ubezpieczenia.

### ***Uwagi odnośnie przykładowych programów***

Wszystkie przykładowe programy i fragmenty programów zamieszczone w niniejszej instrukcji lub dołączone na dyskietce zostały zamieszczone wyłącznie w celu lepszego zilustrowania omawianych zagadnień. Z uwagi na różnorodność czynników oraz wymagania występujące w przypadku konkretnej instalacji, Horner APG nie ponosi odpowiedzialności za skutki korzystania z zamieszczonych przykładów i schematów. Projektant systemu ponosi wyłączną odpowiedzialność za odpowiednie zaprojektowanie systemu, właściwe zintegrowanie oprogramowania Cscape oraz zapewnienie środków zabezpieczających dla urządzeń końcowych, stosownie do wymogów zawartych w odpowiednich normach branżowych i przepisach.

**Uwaga:**

Przykładowe programy sterujące zawarte w niniejszej instrukcji zamieszczono wyłącznie w celu lepszego zilustrowania omawianych zagadnień. Całkowitą odpowiedzialność za prawidłową pracę maszyny ponosi integrator systemu.

## Wprowadzenie

### Przeznaczenie instrukcji

---

Niniejsza instrukcja obsługi przeznaczona jest dla początkujących i średnio zaawansowanych programistów, korzystających z oprogramowania Cscape. Wymagane są podstawowe wiadomości z zakresu obsługi aplikacji Cscape, ponieważ niniejsza instrukcja nie zawiera szczegółowych opisów procedur obsługi. W przypadku dodatkowych pytań należy skorzystać z pomocy dostępnej w oprogramowaniu narzędziowym Cscape.

### Przegląd zawartości instrukcji

Cała instrukcja została w taki sposób przygotowana, aby ułatwić użytkownikowi programowanie oraz odszukiwanie żądanych informacji. Główne zagadnienia to:

- Podstawowe informacje o systemie Cscape (przegląd oprogramowania, wymagania, dystrybucja i instalowanie)
- Bloki funkcyjne języka drabinkowego (włączając w to bloki funkcyjne specjalne)
- Cscape Data Types Typy danych
- Zasoby systemowe sterowników
- Rejestry systemowe
- Schematy połączeń i rozmieszczenie styków
- Liczby zmiennoprzecinkowe
- Moduły SmartStack STP100
- Bloki PID
- Przetworniki analogowo-cyfrowe
- Moduł do czujników termoparowych/oporowych
- Aktualizowanie oprogramowania systemowego
- Skróty klawiszowe w Cscape
- Znaki tekstowe
- Korzystanie z edytora graficznego

## Podstawowe informacje o systemie Cscape

### Charakterystyka ogólna

Cscape to skrót od "Control Station Application Programming Environment" („Centralne środowisko do programowania aplikacji stacji sterujących"). Narzędzie Cscape pozwala na programowanie różnych sterowników PLC firmy Horner APG za pomocą jednego pakietu narzędziowego. W skład oprogramowania Cscape wchodzi:

- Edytor języka drabinkowego (Advanced Ladder Editor)
- Edytor języka IEC61131-1 (IEC 61131 Language Editor)
- Edytor języka drabinkowego z obsługą zmiennych symbolicznych (Advanced Ladder with Tag Editor)
- Edytor interfejsu operatorskiego (View/Edit Graphical Pages)
- Narzędzia do konfigurowania sterownika, włączając w to konfigurowanie wejść/wyjść (Hardware Configuration)
- Nawigator projektu, ułatwiający zarządzanie dużymi projektami (Project Navigator)
- Program śledzący pracujący w czasie rzeczywistym (Debugger)
- Rozbudowany system pomocy (Help)

Oprogramowanie Cscape wykorzystuje wszystkie możliwości systemu operacyjnego Windows i posiada łatwy w obsłudze intuicyjny interfejs. Większość zadań realizowana jest za pomocą myszy. W przypadku sterowników pracujących w sieci (CAN, DeviceNet, itp.), za pomocą oprogramowania Cscape można komunikować się (wysyłanie, odczyt danych) lub monitorować dowolny sterownik Horner APG pracujący w sieci. Funkcja Network Pass Through Connection pozwala oprogramowaniu Cscape na komunikowanie się z dowolnym sterownikiem, z jednego miejsca. Nie jest wymagane bezpośrednie, fizyczne podłączenie się do każdego z programowanych urządzeń. Cscape pozwala na logiczne łączenie się z każdym urządzeniem w sieci, z poziomu dowolnego innego urządzenia.

Oprogramowanie Cscape pozwala na konfigurowanie włączonych w sieć sterowników Horner APG. Funkcja Network Pass Through umożliwia zaprogramowanie dowolnego urządzenia, po fizycznym podłączeniu się do jakiegokolwiek innego sterownika występującego w tej sieci.

Każdy program w języku drabinkowym jest przed wysłaniem automatycznie sprawdzany w celu wykrycia ewentualnych błędów w składni. Fragment kodu źródłowego zawierający błędy można z łatwością odszukać poprzez dwukrotne kliknięcie na komunikacie.

Mechanizmy zabezpieczeń OEM uniemożliwiają nieupoważnionym osobom wyświetlanie lub edytowanie kodu źródłowego w języku drabinkowym. Szczegółowe drabiny oznaczone jako "Sekcje OEM" mogą być wyświetlane i edytowane wyłącznie przez pracowników posiadających odpowiednie uprawnienia.

Programy Cscape są samodokumentujące się, tzn. w sterowniku można zapisać także kod źródłowy, komentarze i nazwy bloków funkcyjnych. Pomimo, że rozwiązanie takie zajmuje więcej miejsca w pamięci sterownika, osoba posiadająca odpowiednie prawa dostępu i oprogramowanie Cscape, ma do dyspozycji kompletny program, obejmujący kod źródłowy, komentarze i nazwy, co pozwala na jego efektywne analizowanie. Nie jest konieczne zapisywanie archiwalnych plików na dysku.

Debugger oprogramowania Cscape pozwala na zlokalizowanie błędów fizycznych lub błędów powodowanych przez czynniki zewnętrzne. Umożliwia to łączenie się w czasie rzeczywistym ze wszystkimi nieprawidłowo pracującymi sterownikami. Użytkownik może obserwować punkty wejściowe i wyjściowe oraz wpływ jaki wywiera zmiana ich wartości.

Oprogramowanie Cscape obsługuje całą linię sterowników Horner APG. Można skonfigurować oprogramowanie Cscape dla określonego modelu sterownika i przygotować programy sterujące, zanim sam sterownik zostanie zakupiony. Po podłączeniu sterownika Cscape umożliwia automatyczne konfigurowanie sterowników.

Cscape pozwala na jednoczesne korzystanie z kilku plików zawierających programy w języku drabinkowym, umożliwia też jednoczesne monitorowanie z poziomu jednego komputera osobistego wszystkich sterowników Horner APG.

## Wymagania

---

Komputer pracujący pod systemem operacyjnym Microsoft Windows XP, 7 lub nowszym

- CPU co najmniej 1 GHz (do komfortowej obsługi narzędzia)
- 2GB pamięci RAM
- Mysz
- Wolny port szeregowy, USB lub Ethernet do podłączenie sterowników Horner APG
- 500MB wolnego miejsca na dysku.

Na dysku należy również przewidzieć miejsce do przechowywania napisanych programów w języku drabinkowym. Jeżeli mysz jest podpięta do portu szeregowego, komputer musi dysponować drugim, wolnym portem szeregowym. Nie ma możliwości zmiany parametrów konfiguracyjnych portu szeregowego wykorzystywanego przez oprogramowanie Cscape. Zawsze wykorzystywane są następujące parametry komunikacyjne: 9600 baud, 8 bitów danych, no parity i 1 bit stopu.

## Dystrybucja

---

Oprogramowanie Cscape można bezpłatnie pobrać z Platformy Internetowej ASTOR lub na życzenie może być wysłany dysk instalacyjny.

---

## Instalowanie

---

Płyta instalacyjna zawiera Kreatora Instalacji.

Należy uruchomić program SETUP.EXE, umieszczony na płycie CD-ROM.

Proces instalowania jest bardzo prosty. Należy jednak zachować szczególną uwagę przy wyborze katalogu, w którym zainstalowane zostanie oprogramowanie.

Domyślnie jest to katalog C:\Program Files\Cscape. Najczęściej wybierany "własny" katalog to C:\Cscape. Należy zawsze zapamiętać, w jakim katalogu "głównym" zostało zainstalowane oprogramowanie Cscape, bez względu czy jest to katalog C:\Program Files\Cscape, C:\Cscape czy też dowolny inny.

### Przebieg instalowania

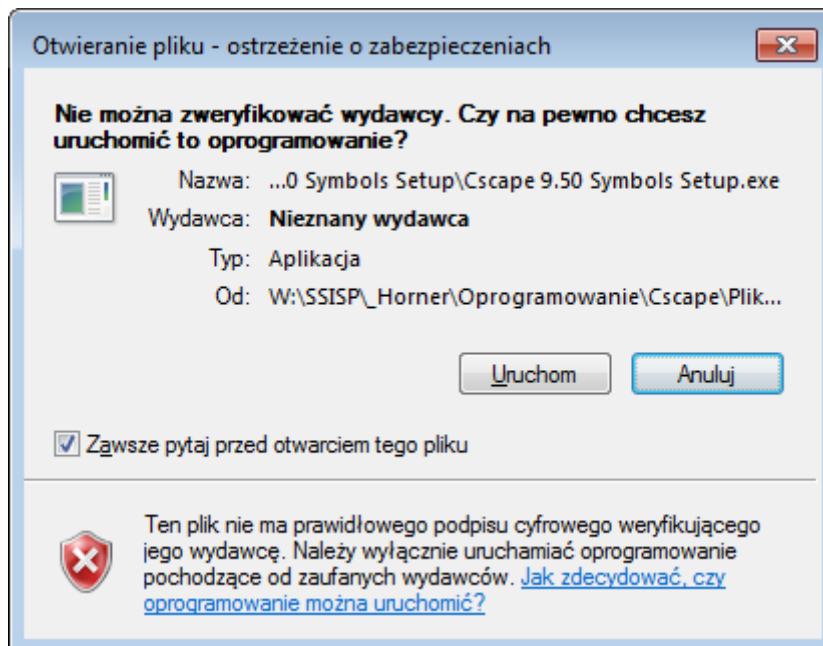
Instalowanie środowiska Cscape wymaga pobrania z Platformy Internetowej ASTOR pliku instalacyjnego. Ścieżka dostępu do katalogu w którym przechowywane są pliki instalacyjne jest następująca <http://platforma.astor.com.pl/grupa/Cscape>. Dostęp do pliku jest bezpłatny i nie wymaga logowania do serwisu w celu pobrania. Jeśli nie masz dostępu do Internetu aby pobrać oprogramowanie skontaktuj się z najbliższym oddziałem handlowym w celu otrzymania płyty DVD z oprogramowaniem narzędziowym Cscape.

**Uwaga.**

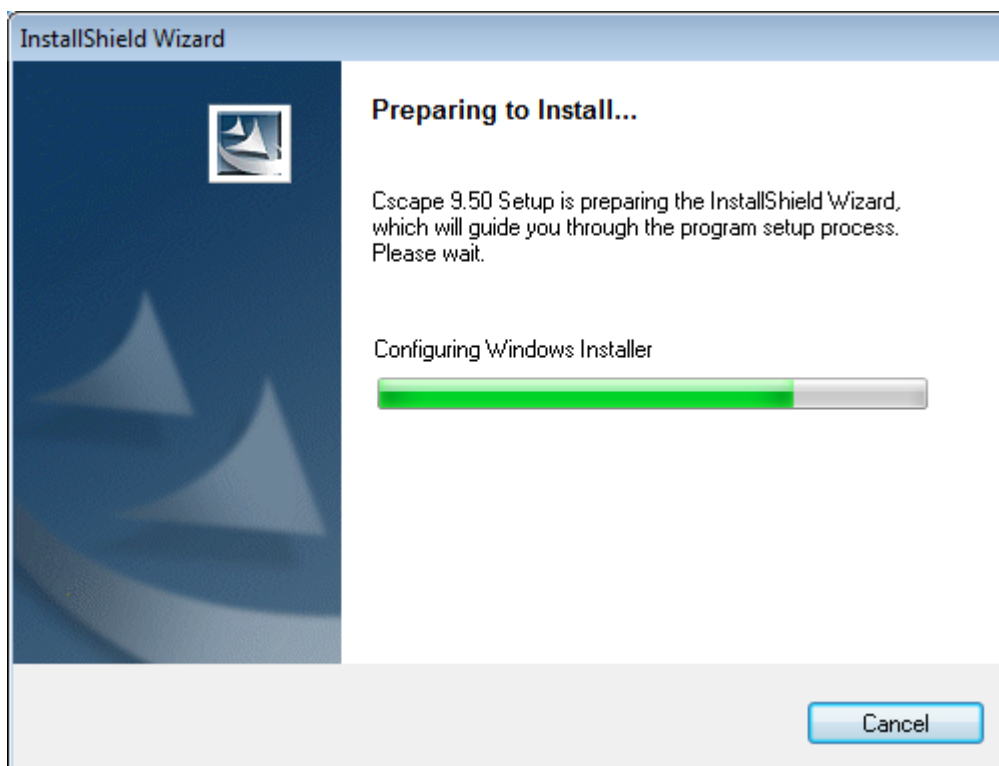
Firma ASTOR zaleca pobierania oprogramowania z Platformy Internetowej ASTOR – tylko wersje które udostępnione są na Platformie Internetowej ASTOR posiadają pełne wsparcie inżynierów ASTOR oraz producenta oprogramowania. Wersje narzędzia które pobrane zostały z innych źródeł nie posiadają wsparcia.

Uruchom plik instalacyjny i potwierdź jego uruchomienie

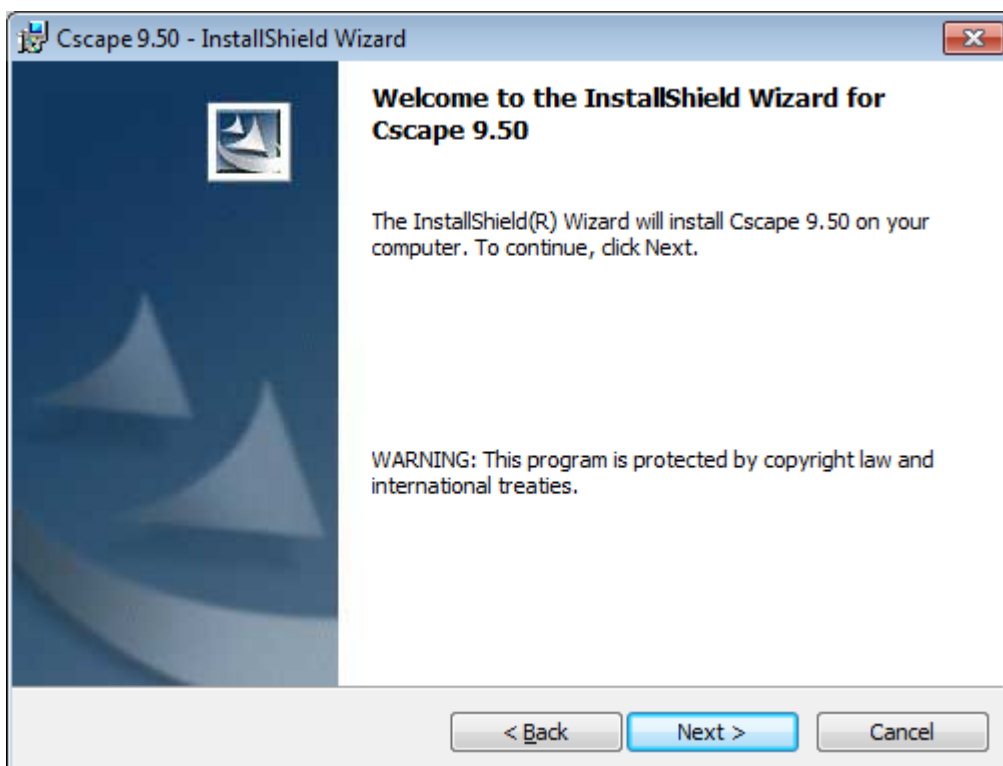




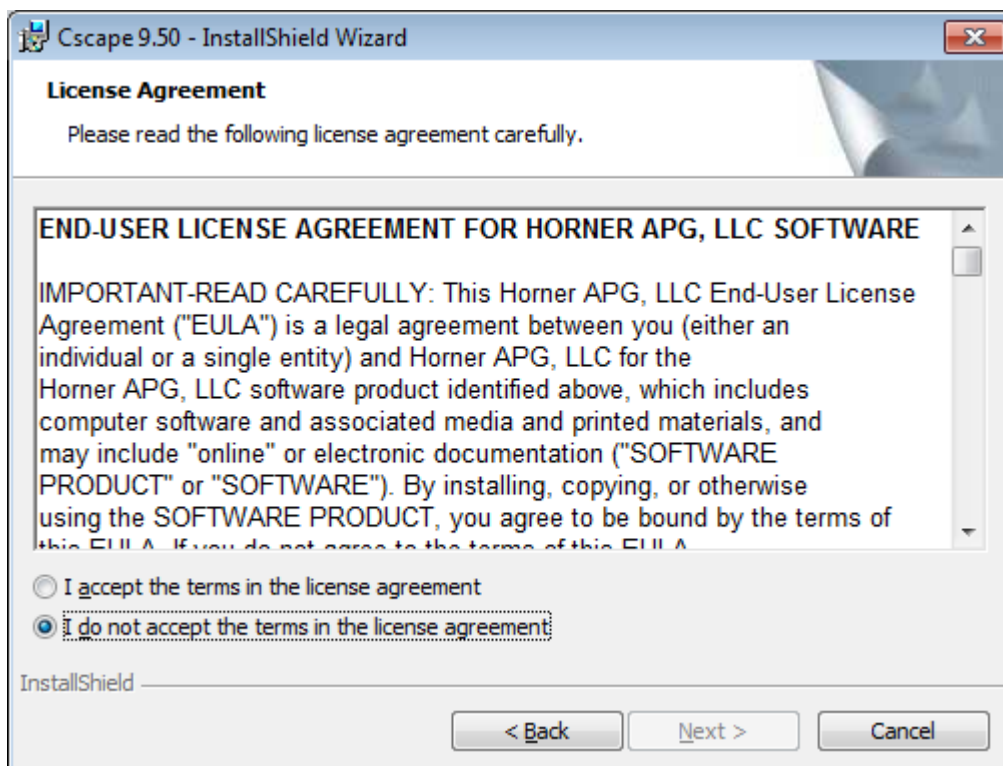
Oprogramowanie znaczenie się instalować



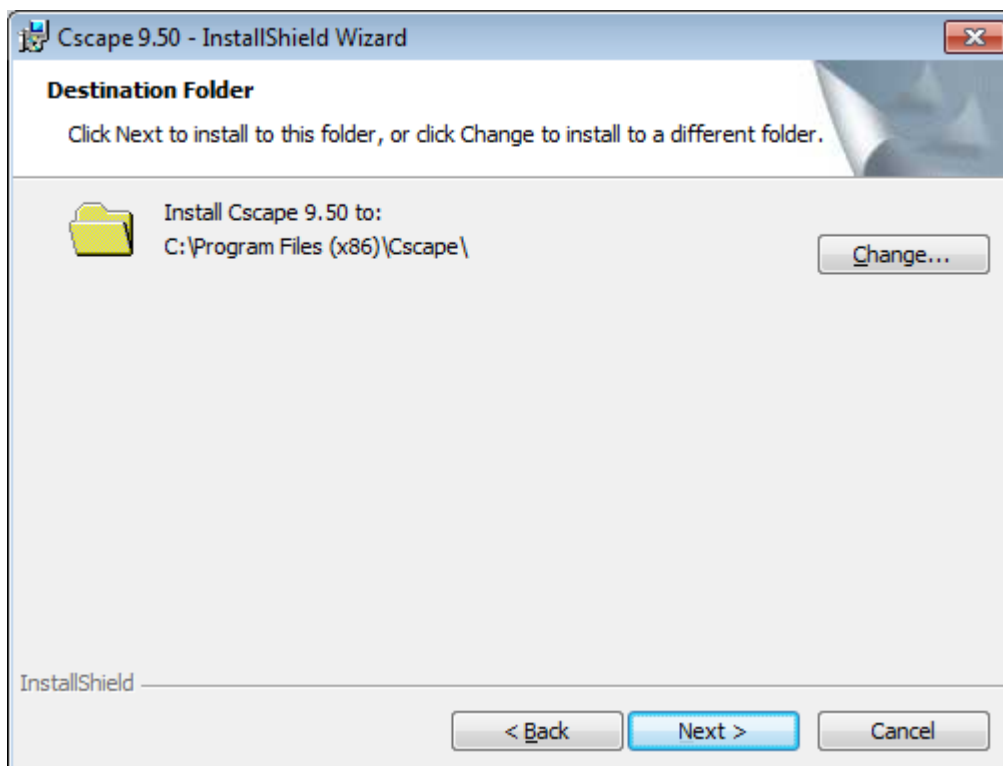
Postępuj zgodnie z procedurą instalacji, wybierz *Next*



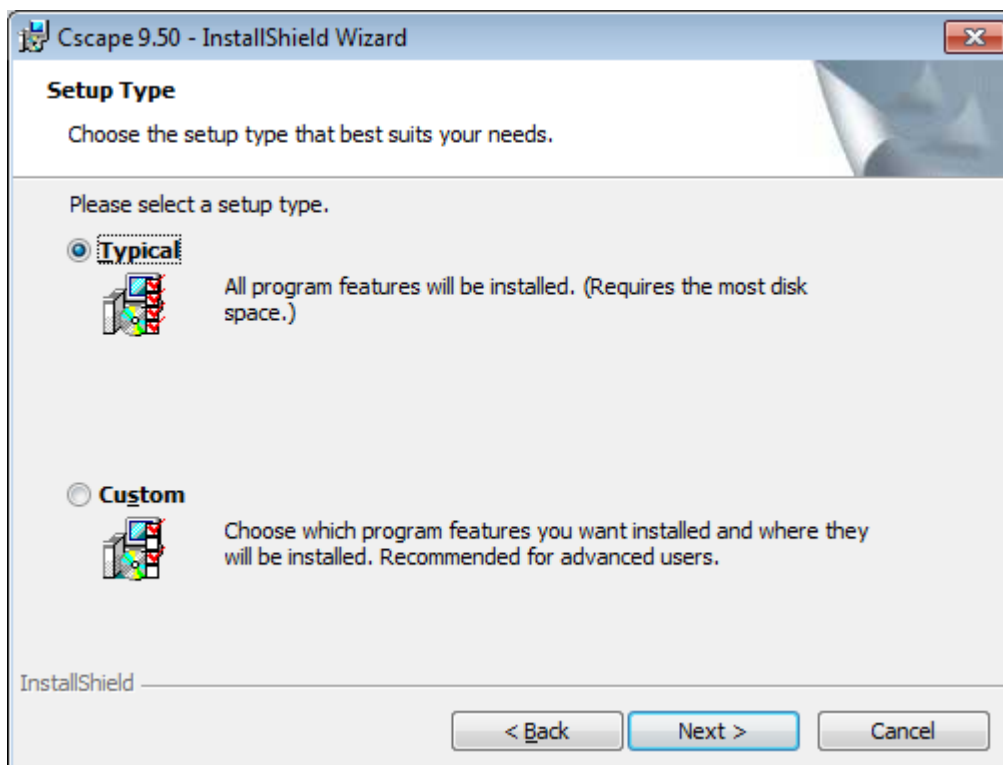
Zapoznaj się i zaakceptuj warunki umowy licencyjnej zaznaczając opcję *I accept the terms in the licence agreement* i klikając **Next**



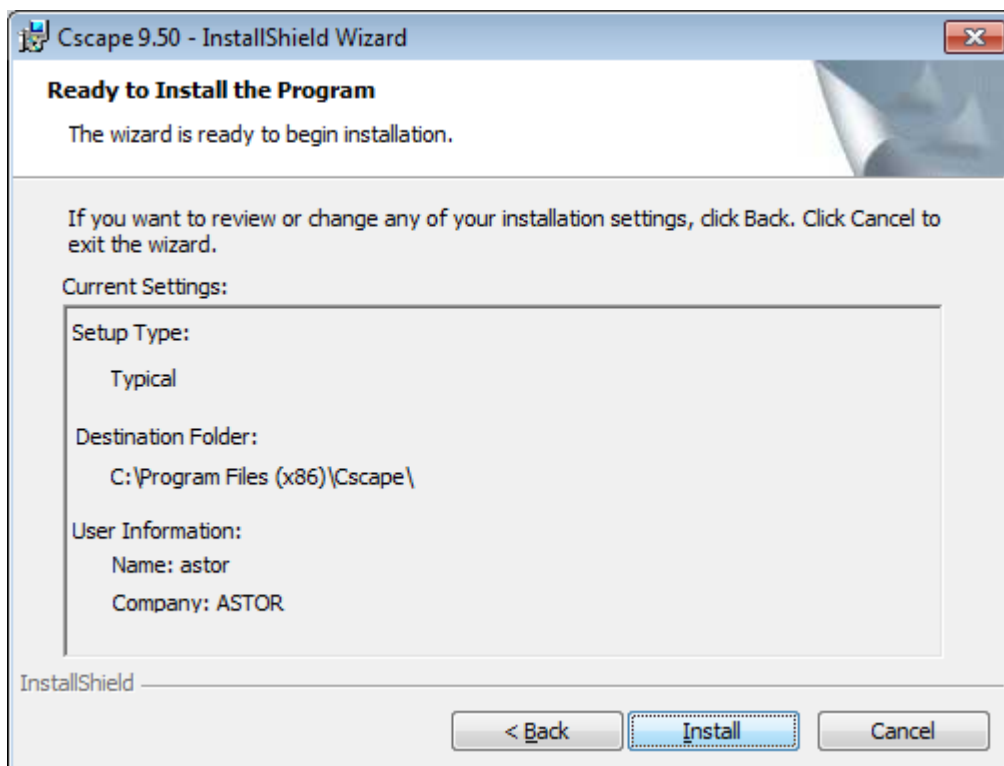
Wskaż miejsce na dysku gdzie zostanie zainstalowane oprogramowanie. Domyślny folder instalacji to *C:\Proram Files (x86)\Cscape\*. Aby go zmienić kliknij przycisk *Change*. Przejdź do kolejnego ekranu przyciskając *Next*



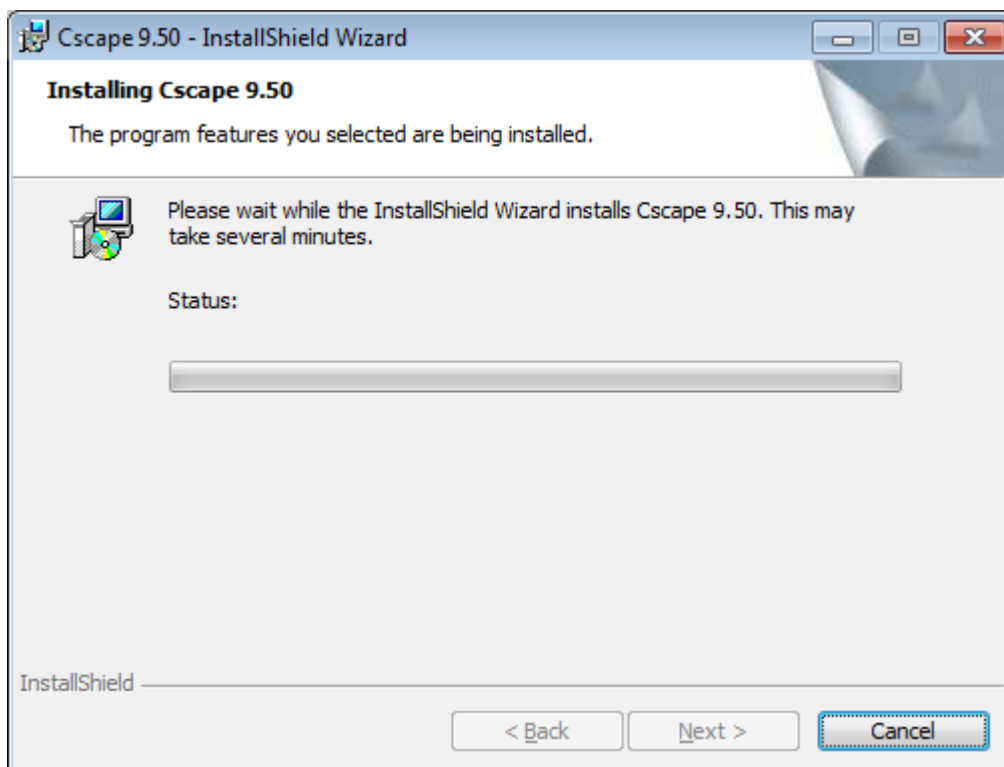
Wybierz domyślny tryb *Typical* instalacji i naciśnij *Next*



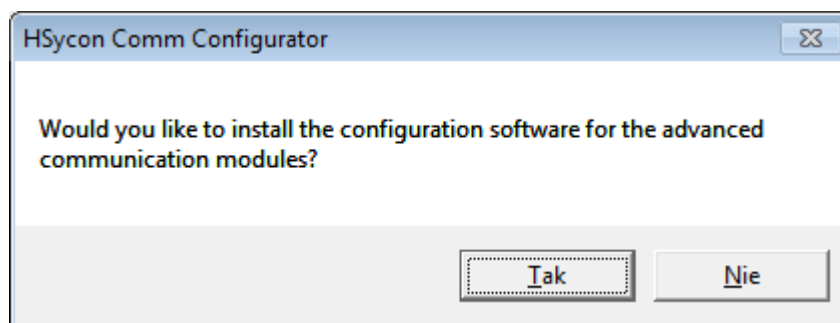
Instalator wyświetli podsumowanie konfiguracji instalacji. Przyciśnij *Install* aby rozpocząć instalację



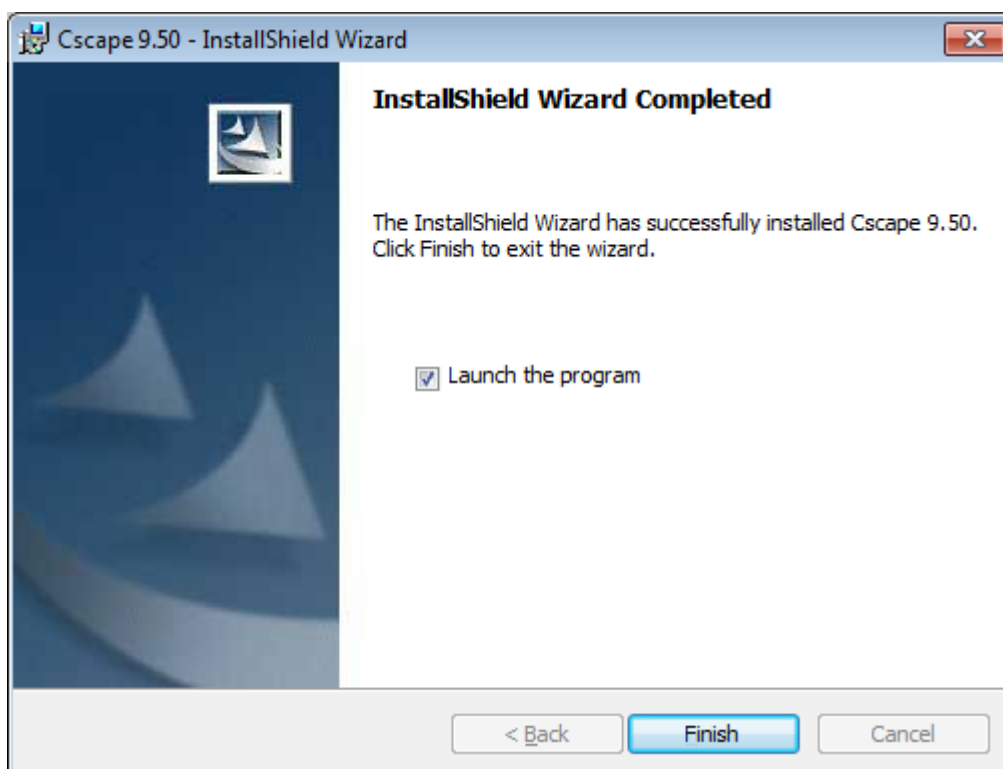
Rozpocznie się właściwy proces instalacji



Po zainstalowaniu oprogramowanie Cscape wyświetlone zostanie okno dialogowe czy chcemy zainstalować moduł do obsługi opcjonalnych kart komunikacyjnych *HeSysCon* (dla sterowników NX i QX). Jeśli nie chcemy go instalować przyciskamy *Nie*

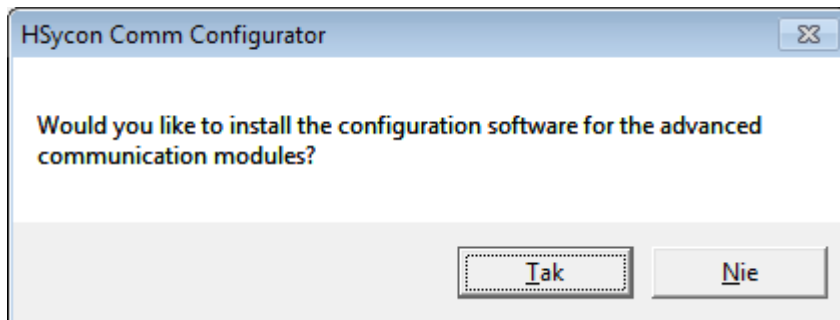


Oprogramowanie Cscape zostało zainstalowane o czym poinformuje okno dialogowe. Aby zakończyć naciśnij *Finish*. Zaznaczenie opcji *Launch the program* spowoduje uruchomienie oprogramowania

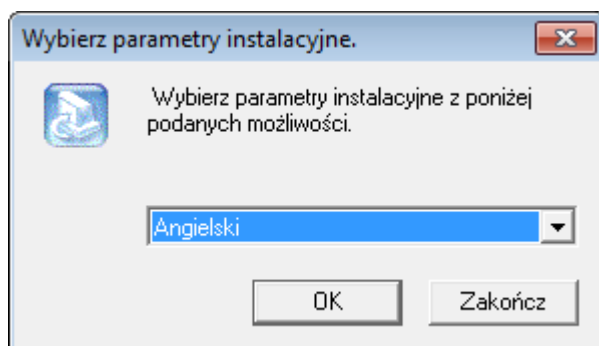


## Instalowanie środowiska HeSysCon

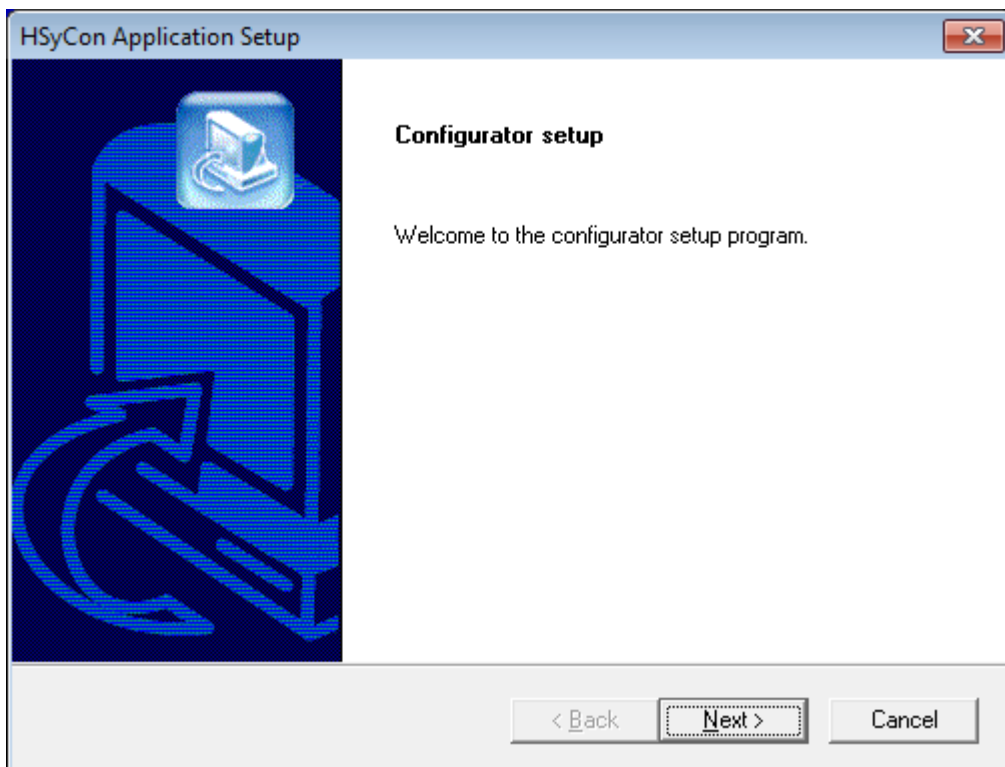
Jeśli chcemy zainstalować moduł do obsługi opcjonalnych kart komunikacyjnych *HeSysCon* klikamy *Tak*



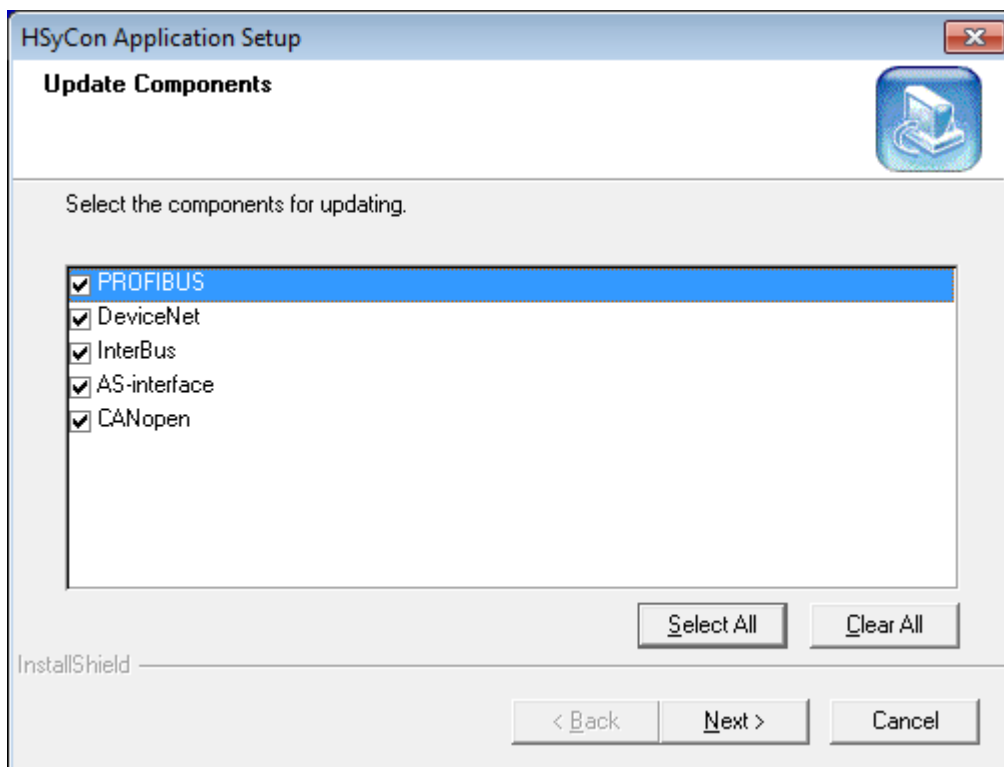
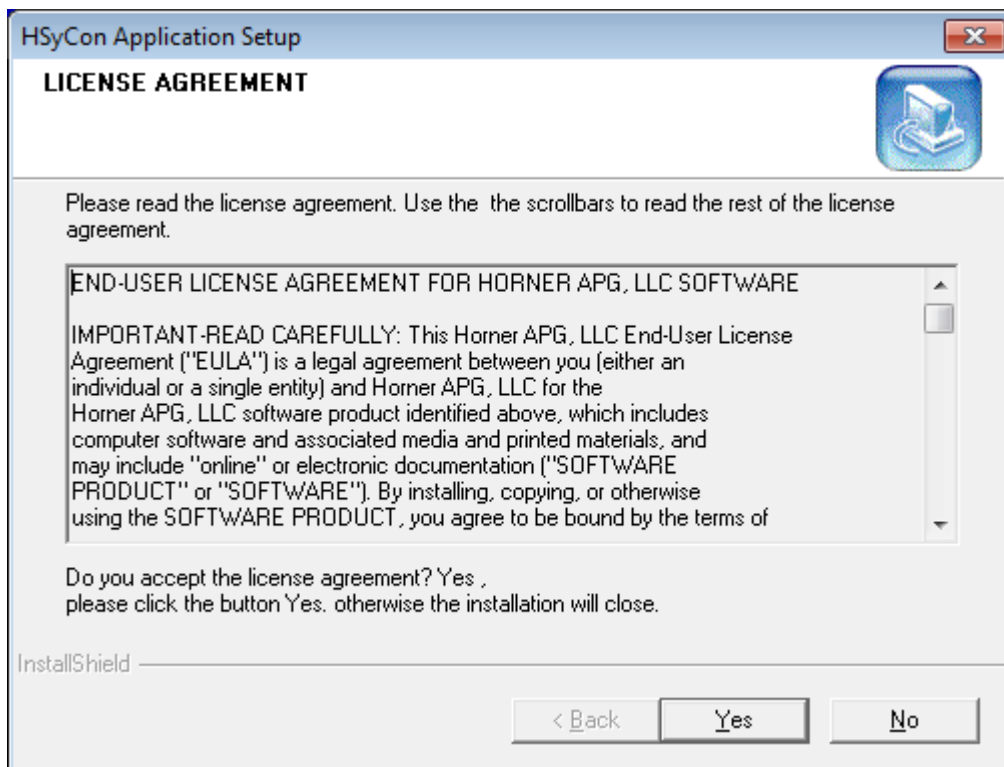
Wybieramy język instalacji modułu *HeSysCon* (domyślnie angielski)



Uruchomi się instalator modułu *HeSysCon*, Przyciśnij *Next* aby rozpocząć

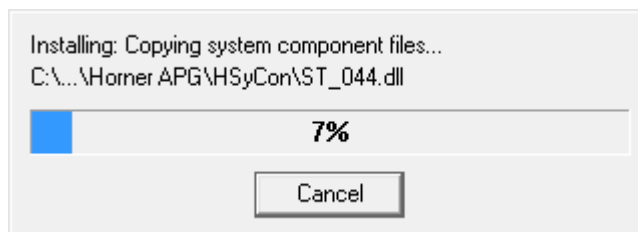


Zapoznaj się i zaakceptuj warunki umowy licencyjnej klikając przycisk Yes

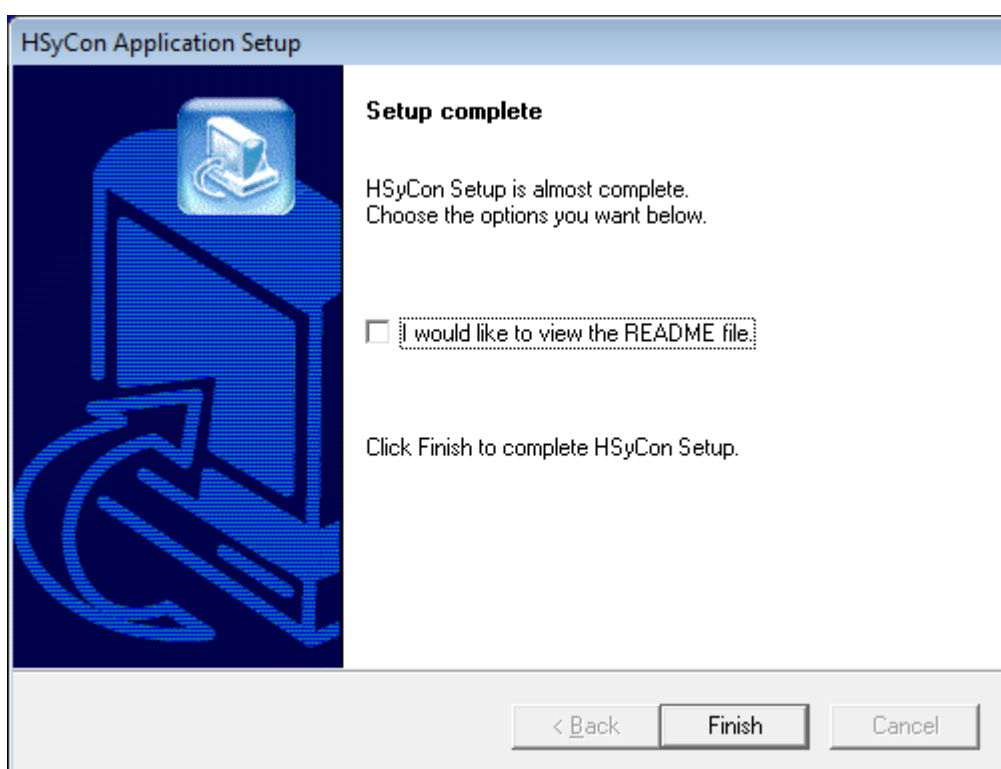




Rozpocznie się instalacja modułu



Po zakończeniu instalacji naciśnij przycisk *Finish*



Oprogramowanie zostało zainstalowane na komputerze.

## Wsparcie techniczne

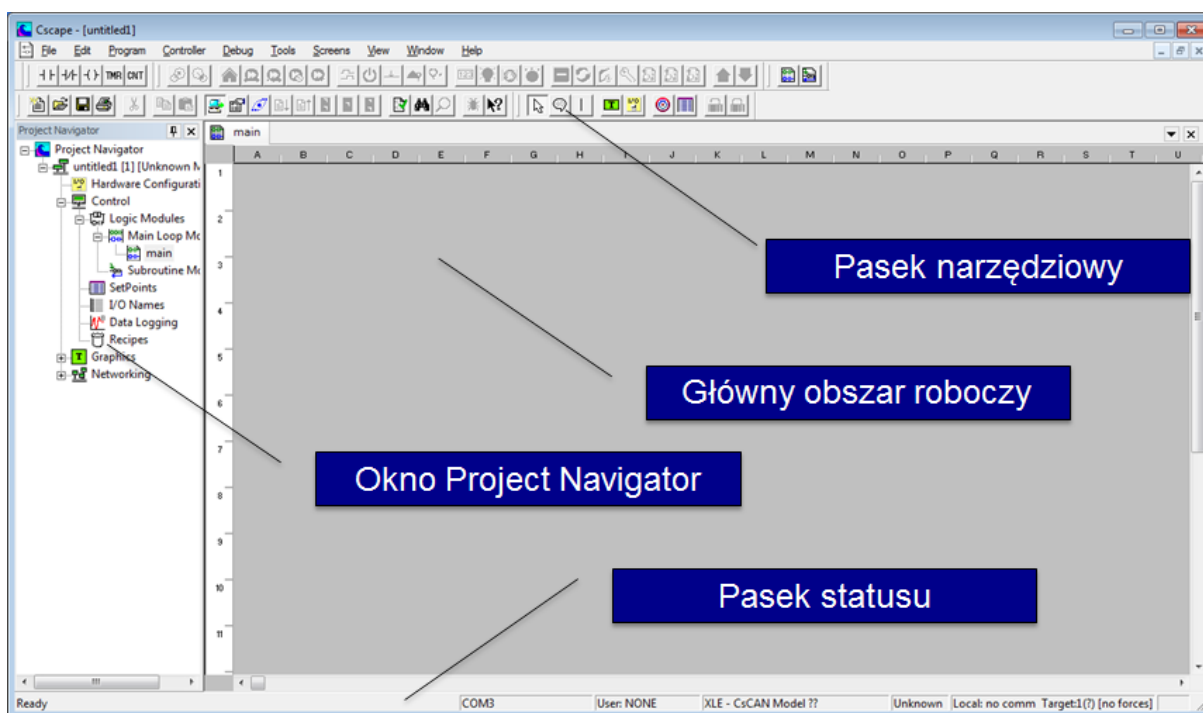
ASTOR Sp. z o.o.  
Ul. Smoleńsk 29  
31-112 Kraków  
tel. 12 428 63 00  
fax 12 428 63 09  
e-mail: [info@astor.com.pl](mailto:info@astor.com.pl)  
[www.astor.com.pl](http://www.astor.com.pl)

## Wygląd oprogramowania narzędziowego Cscape

Po uruchomieniu oprogramowania Cscape pojawi się ekran główny narzędzia. Oprogramowanie jest skonfigurowane tak, żeby zaraz po uruchomieniu wyświetlało konfigurator połączenia oraz konfigurator sterownika Horner APG których będzie programowany. Jeśli nie dysponujemy podłączonym sterownikiem ekran konfiguratora można zamknąć.

### Domyślny wygląd oprogramowania Cscape

Domyślny wygląd oprogramowania przedstawia poniższy rysunek i można na nim wskazać cztery główne obszary narzędzia.



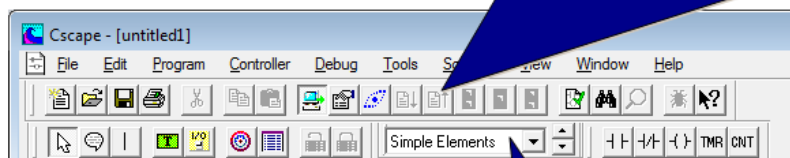
### Główny obszar roboczy

Zajmuje największą część ekranu i jest to obszar w który przygotowywany będzie program sterujący

### Pasek narzędziowy

Pasek narzędziowy to element pozwalający na szybki dostęp do podstawowych elementów wykorzystywanych podczas programowania.

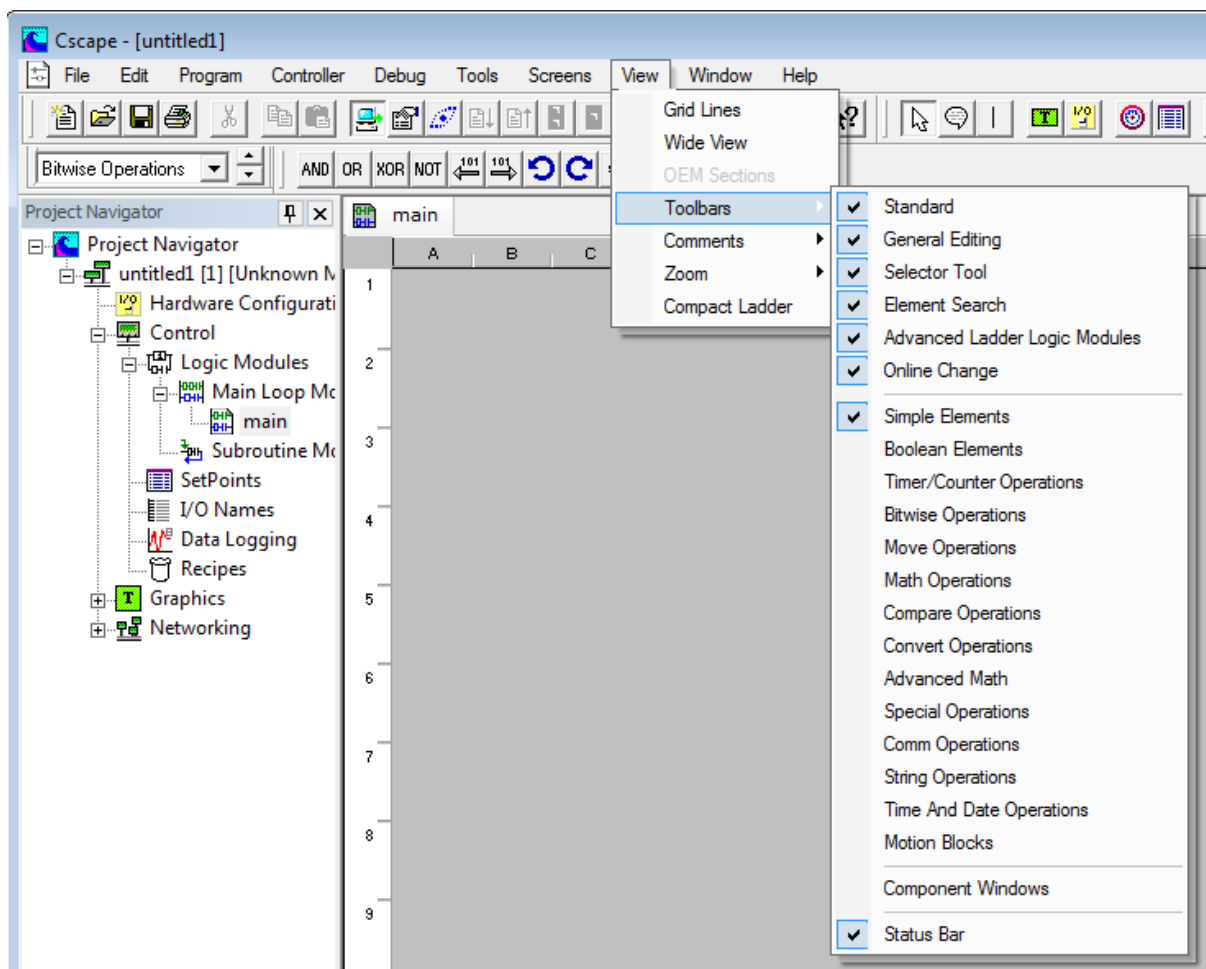
**Standard** – w pasku tym dostępne są podstawowe funkcje pozwalające na dokonywanie operacji na projekcie, czyli zapis, odczyt, przesłanie projektu do kontrolera, pobranie projektu z kontrolera, przełączniki trybu pracy sterownika, walidacja projektu.



**General Editing** – pasek pozwalający na konfigurację sprzętową kontrolera, zainstalowanych w systemie modułów, ustawienie adresu ID.

**Selektor Tool** – rozwijalna lista bibliotek

Wygląd paska użytkownik może skonfigurować indywidualnie od kątem własnych potrzeb wybierając paski narzędziowe które będą widoczne w oprogramowaniu. Konfiguracja realizowana je z poziomu menu *View/Toolbars* . Każdy pasek narzędziowy jest dokowalny w obszarze głównego paska i możemy określić w którym miejscu będzie widoczny.



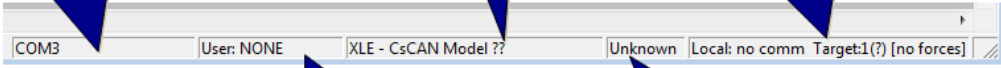
## Dostępne grupy bloków funkcyjnych

- Simple Elements – pasek zawierający podstawowe elementy niezbędne do tworzenia projektu, czyli styki normalnie otwarte i normalnie zamknięte oraz cewki
- Boolean Elements – pasek zawiera wszystkie typy styków oraz cewek
- Timer/Counter Operations – bloczki funkcyjne odpowiedzialne za operacje związane z licznikami i timerami
- Bitwise Operations – operacje bitowe AND, OR, XOR, NOT
- Move Operations – operacje związane przrzućaniem danych
- Math Operations – arytmetyczne operacje matematyczne
- Compare Operations – operacje związane z porównywaniem wartości liczbowych
- Convert Operations – bloczki pozwalające na konwertowanie typów wartości
- Advanced Math – zaawansowane funkcje matematyczne

- Special Operations – funkcje specjalne związane z ustawianiem zegara czasu rzeczywistego, konfiguracją alarmów, regulatorów PID oraz bloków odpowiedzialnych za obsługę wymiennych nośników danych
- Comm Operations – bloczki odpowiedzialne za komunikację z innymi urządzeniami, konfiguracji portów i protokołów
- String Operations – podstawowe operacje na danych typu string

## Pasek statusu

Pasek statusu zajmuje zawsze dolną część okna oprogramowania i udostępnia użytkownikowi podstawowych informacji dotyczących podłączonego do oprogramowania modelu sterownika oraz jako aktualnego statusu pracy.



The screenshot shows a status bar with the following fields: COMB, User: NONE, XLE - CsCAN Model ??, Unknown, Local: no comm, Target:1(?) [no forces].

**COMB** – pole informuje w jaki sposób skonfigurowane jest połączenie sterownika do komputera

**Model** – pole informujące o tym czy model kontrolera skonfigurowany w programie jest taki sam jak model fizycznie podłączony do komputera.

**Local and Target** – określają urządzenie o podanym ID, które jest podłączone do komputera. Target określa, w jakim trybie znajduje się kontroler: (R) – kontroler w trybie RUN, (I) – kontroler w trybie STOP lub IDLE, (D) – kontroler w trybie DO/IO, (B) – informuje, że kontroler jest zajęty (BUSY, inny komputer komunikuje się w tym czasie z kontrolerem).

**User** – pole to określa aktualnie zalogowanego użytkownika. Jeśli zabezpieczenia nie są skonfigurowane, lub nikt nie jest aktualnie zalogowany w polu pojawi się wartość NONE.

**Program Equality** – informuje użytkownika, czy program sterujący znajdujący się w jednostce sterującej i w oprogramowaniu Cscape jest taki sam.

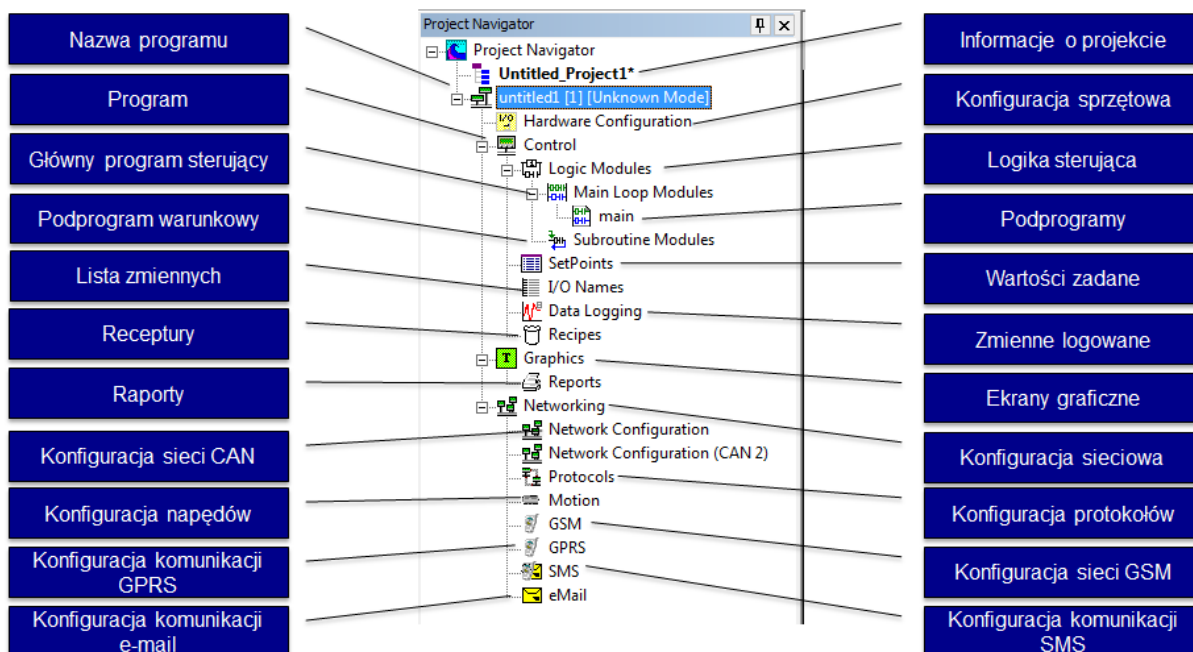
Pasek statusu to pasek na którym możemy obserwować wszystkie niezbędne informacje związania z podłączonym do Cscape urządzeniem oraz zgodności konfiguracji oraz programu pomiędzy projektem a programem znajdującym się w sterowniku

- Pole User - pole to określa aktualnie zalogowanego użytkownika. Jeśli zabezpieczenia nie są skonfigurowane, lub nikt nie jest aktualnie zalogowany w polu pojawi się wartość NONE.

- Pole Model - pole informujące o tym czy model kontrolera skonfigurowany w programie jest taki sam jak model fizycznie podłączony do komputera
- Pole Local and Target – określają urządzenie o podanym ID, które jest podłączone do komputera. Target określa, w jakim trybie znajduje się kontroler: (R) – kontroler w trybie RUN, (I) – kontroler w trybie STOP lub IDLE, (D) – kontroler w trybie DO/IO, (B) – informuje, że kontroler jest zajęty (BUSY, inny komputer komunikuje się w tym czasie z kontrolerem)
- Pole Program Equality – informuje użytkownika, czy program sterujący znajdujący się w jednostce sterującej i w oprogramowaniu Cscape jest taki sam.
- Pole COM - pole informuje w jaki sposób skonfigurowane jest połączenie sterownika do komputera

## Okno Project Navigator

### Okno Project Navigator



Okno Project Navigator to okno z poziomu którego mamy szybki dostęp do konfiguracji projektu, konfiguracji sprzętowej sterownika, logiki, ekranów operatorskich oraz komunikacji.

## Bloki funkcyjne języka drabinkowego

### Elementy programu sterującego opisane w niniejszej dokumentacji

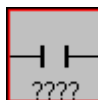
W czasie programowania można korzystać z następujących grup bloków funkcyjnych:

Styki i przekaźniki	Styk NO, przekaźnik NC, przekaźnik POS, ...
Elementy sterujące wyświetlaniem ekranów	Strony systemowe, strony alarmów, strony użytkownika)
Operacje logiczne	AND, NOT, XOR,...
Funkcje matematyczne	Add, Sub, Sin... i zaawansowane funkcje matematyczne
Równania matematyczne	%R1 = 1 + %R45,...
Operacje porównywania	>, <=, =...
Konwersja typów	INT, DINT, REAL
Liczniki i przekaźniki czasowe	On Delay, Count Up,...
Operacje bitowe	SHL, SHR, ROL,...
Przemieszczanie danych	MOV, BMV, IMV, Fill,...
Operacje na łańcuchach znaków	MOV STR, CMP STR
Bloki funkcyjne do komunikacji sieciowej	Net GetW, NetPutW, NetGet HB, NetPut HB
Komunikacja przez port szeregowy	Close, Send, RECV,...
Funkcje specjalne	Stepper Move, PID,...
Inne bloki funkcyjne	Komentarze, połączenia pionowe

### Styki i przekaźniki

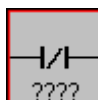
W czasie programowania można korzystać z następujących styków i przekaźników:

#### Styk otwarty



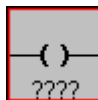
Przesyła sygnał, jeżeli związana z nim zmienna ma wartość 1 (ON).

#### Styk zamknięty



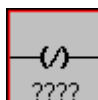
Przesyła sygnał, jeżeli związana z nim zmienna ma wartość 0 (OFF).

### Przełącznik chwilowy o stykach otwartych



Doprowadzenie sygnału do przełącznika powoduje ustawienie wartości związanej z nim zmiennej na 1.

### Przełącznik chwilowy o stykach zamkniętych



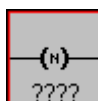
Doprowadzenie sygnału do przełącznika powoduje ustawienie wartości związanej z nim zmiennej na 0.

### Przełącznik uaktywniany zboczem narastającym sygnału



Jeżeli doływający sygnał do przełącznika zmieni stan z 0 na 1 to wartość zmiennej powiązanej z tym przełącznikiem jest ustawiana na 1 na okres jednego cyklu wykonywania programu sterującego.

### Przełącznik uaktywniany zboczem opadającym sygnału



Jeżeli doływający sygnał do przełącznika zmienia stan z 1 na 0 to wartość zmiennej powiązanej z tym przełącznikiem jest ustawiana na 1 na okres jednego cyklu wykonywania programu sterującego.

### Przełącznik-zatrząsk SET



Doprowadzenie sygnału do tego przełącznika powoduje ustawienie wartości związanej z nim zmiennej na 1. Zmienna zachowuje tę wartość do momentu jej zmiany za pomocą przełącznika Reset.

### Przełącznik-zatrząsk RESET



Doprowadzenie sygnału do tego przełącznika powoduje ustawienie wartości związanej z nim zmiennej na 0. Zmienna zachowuje tę wartość do momentu jej zmiany za pomocą przełącznika Set.



## Bloki funkcyjne do wyświetlania

### Wyświetlanie ekranów użytkownika

Z poziomu programu sterującego w języku drabinkowym można sterować wyświetlaniem stron.

Przełącznik powiązany z rejestrem %D staje się przełącznikiem do wyświetlania stron. Dzięki takiemu rozwiązaniu, z poziomu języka drabinkowego można z łatwością zmieniać numer strony pokazywanej na wyświetlaczu. W czasie konfigurowania tego przełącznika należy wybrać opcję **Force Screen** lub **Switch Screen**.

**Force Screen** - ekran jest pokazywany na wyświetlaczu tak długo, jak zmienna związana z przełącznikiem ma wartość 1. Powoduje to zasłonięcie wszystkich innych, wyświetlanych ekranów użytkownika. Jeżeli sygnał jest jednocześnie doprowadzany do kilku przełączników z ustawioną opcją Force Screen, na wyświetlaczu pokazywana jest ekran związany z przełącznikiem, który umieszczony jest w programie sterującym jako ostatni. Numer tego ekranu można odczytać ze zmiennej systemowej %SR2. Tryb Force Screen może posłużyć np. do budowy ekranów alarmowych.

**Switch Screen** - Opcja ta pozwala na przełączenie do określonego ekranu, ale nie wymusza, aby ekran ten pozostawał aktywny. Operator może przełączać się pomiędzy ekranami korzystając z różnych metod (menu, wywoływanie stron, przewijanie, ...). Jednocześnie aktywny powinien być wyłącznie jeden przełącznik z ustawioną opcją Switch Screen. Jeżeli zostanie aktywowany następny przełącznik tego typu, nie spowoduje to żadnych zmian na ekranie sterownika. Należy zwrócić uwagę, że ekran może jednak zostać zmieniony poprzez zapis do rejestru %SR1.

#### Uwaga

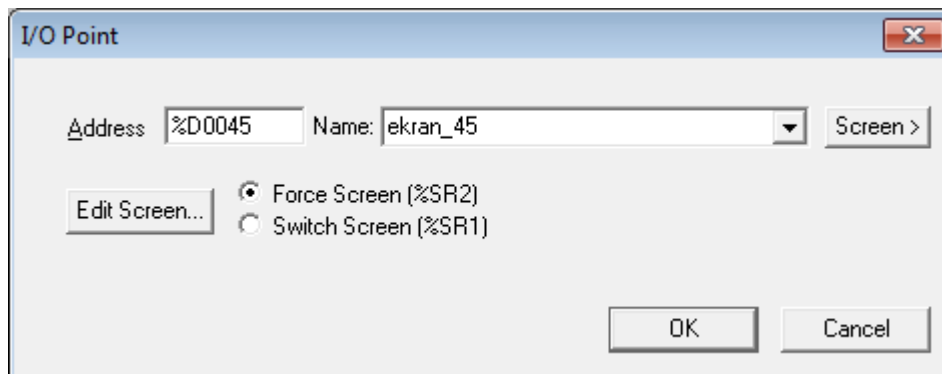
Sygnał nie przepływa przez przełącznik do wyświetlania ekranów (Proszę porównać z informacjami zawartymi w dalszej części)

### Konfigurowanie przełącznika do wyświetlania ekranów

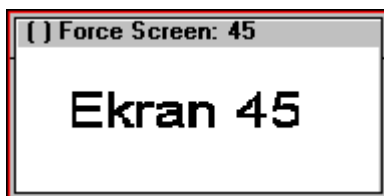
Pierwszy krok to utworzenie przełącznika:



Wpisać %D, a następnie numer ekranu lub wcisnąć przycisk "Screen >" w celu wybrania ekranu z listy.



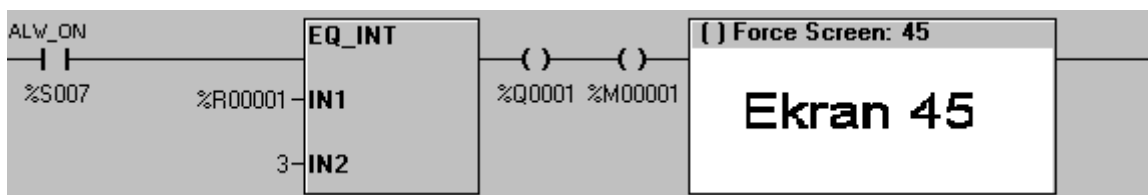
Zaznaczyć opcję Force Screen (%SR2) lub Switch Screen (%SR1), a następnie wcisnąć OK. Przekaznik zostanie wyświetlony w formie ekranu i pokazany zostanie podgląd ekranu, o ile jest on dostępny.



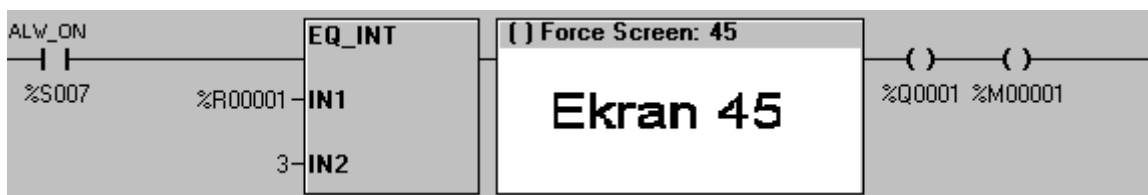
### Przepływ sygnału

Sygnał nie przepływa przez przekaznik do wyświetlania ekranów. Przekaznik do wyświetlania ekranów musi być zawsze umieszczony jako ostatni blok funkcyjny szczebla drabiny logicznej (skrajnie po prawej stronie).

### Prawidłowe użycie przekazywnika %D



### Nieprawidłowe użycie przekazywnika %D



## Rejestry systemowe z numerami wyświetlanej strony użytkownika

Sterownik posiada rejestry systemowe, które mogą być wykorzystane do odczytu i/lub zapisu numerów ekranów użytkownika, alarmowych i systemowych. Poniżej zamieszczono definicje tych trzech typów ekranów:

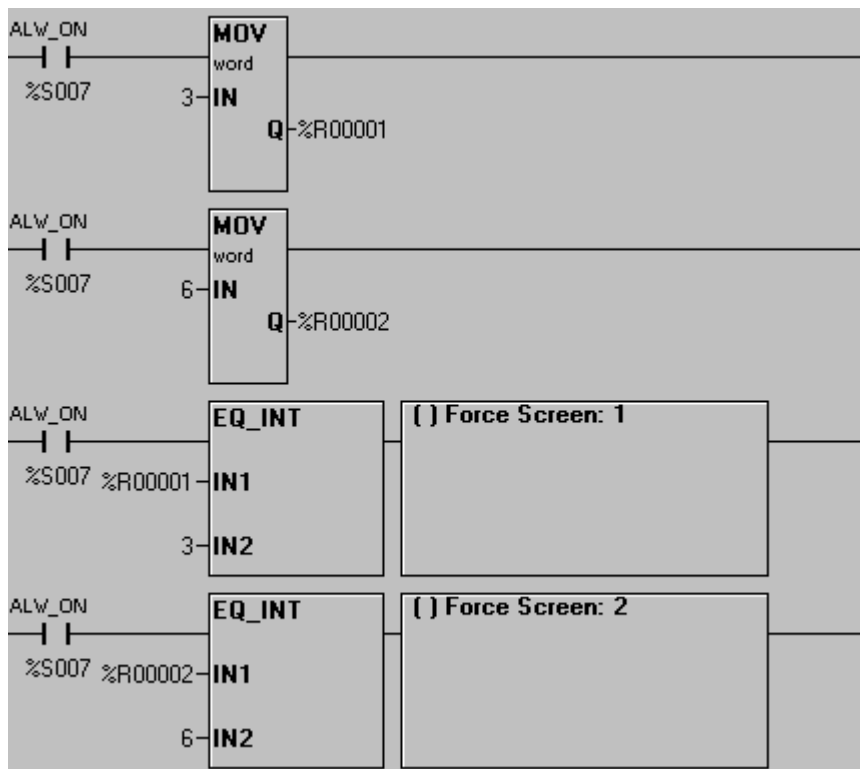
**Ekran systemowy** - predefiniowane ekrany, zawierające menu systemowe. Pierwszy z tych ekranów systemowych jest pokazywany po wejściu do menu system. Wybieranie pozycji z menu powoduje wyświetlanie kolejnych ekranów systemowych. Dowolny z ekranów systemowych można również wyświetlić poprzez wprowadzenie jego numeru do odpowiedniego rejestru systemowego. Ekran systemowy są zawsze pokazywane, nawet jeżeli aktywna jest ekran alarmów lub użytkownika.

**Ekran alarmów** - ekrany zdefiniowane przez programistę, wyświetlane za pomocą przekaźnika %D z poziomu programu sterującego. Numer aktualnie wyświetlanego ekranu można odczytać z rejestru systemowego ekranu alarmów. Zapisanie wartości do rejestru systemowego ekranów alarmowych nie powoduje żadnych zmian, ponieważ jednostka centralna oblicza i wyznacza numer ekranu alarmów po każdym cyklu pracy. Ekran te można w trakcie projektowania zdefiniować jako ekrany alarmów, niemniej jednak ekrany nie zdefiniowane jako typu Alarm, ale wywołane za pomocą przekaźnika %D, są traktowane w tym kontekście jako ekrany alarmów.

**Ekran użytkownika** - jeżeli nie jest wyświetlany ekran systemowy lub alarmów, pokazywany jest ekran użytkownika. Jeżeli dostępnych jest więcej niż jeden ekran użytkownika (nie zdefiniowany przez programistę jako typu Alarm), można przechodzić pomiędzy nimi za pomocą klawiszy sterownika UP i DOWN. Odczyt rejestru systemowego ekranów użytkownika pozwala na monitorowanie ekranów wybieranych przez operatora za pomocą klawiszy UP i DOWN. Zapisanie wartości do tego rejestru systemowego pozwala na bezpośrednie wywołanie ekranu użytkownika, który ma być pokazywany na wyświetlaczu.

## Jednoczesne wyświetlanie kilku ekranów

Jeżeli w czasie jednego cyklu wykonywania programu zostanie uaktywniony więcej niż jeden ekran, na wyświetlaczu pokazywany jest ostatni spośród uaktywnionych. Ma to miejsce w zamieszczonym poniżej programie sterującym:



*Jednoczesne wyświetlanie kilku ekranów*

W przedstawionym przykładzie aktywny jest zarówno ekran Screen 1, jak i Screen 2, ale na wyświetlaczu pokazywany jest ekran Screen 2, ponieważ został on aktywowany później.

Program ten nie zawiera żadnych błędów logicznych lub składniowych, a więc zarówno w czasie kompilowania, jak i pracy nie będą pokazywane żadne błędy. Odpowiedzialność za zaakceptowanie lub nie takiej sytuacji spoczywa wyłącznie na programiście, który może opracować mechanizm zapobiegający takim sytuacjom.

## Operacje logiczne

### Informacje ogólne

#### **Uwaga:**

Bloki funkcyjne do operacji logicznych (AND, OR, itp.) realizują działania na zmiennych typu WORDS (ciąg 16 bitów) lub typu DWORDS (ciąg 32 bitów).

Przy wykonywaniu operacji logicznych na stałych, należy je wprowadzić jako wartości 16 lub 32-bitowe, bez znaku. Operacje logiczne są przeprowadzane na ciągach bitów zapisanych w

rejestrach. Wynik operacji jest zapisywany w trzecim rejestrze, z którego korzysta blok funkcyjny. Żaden z argumentów operacji nie ulega zmianie.

### **Przeptyw sygnału**

Bloki te mają zawsze wartość TRUE. Sygnał zawsze przepływa przez te bloki funkcyjne, bez względu na wynik operacji.

### **Konfigurowanie**

W celu skonfigurowania bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wprowadzić adresy rejestrów z parametrami i rejestru z wynikiem. Warunkiem poprawnego wykonania tego typu bloków funkcyjnych jest podanie trzech (3) rejestrów. Należy podać IN1, IN2 i Q (wynik), za wyjątkiem funkcji NOT, która wymaga podania jednego rejestru z argumentem i rejestru z wynikiem.

Argumenty mogą być stałymi bez znaku typu Word lub DWord, 1, 23056, 45, itp. Rejestr Q (wynik) musi być określony poprzez podanie adresu.

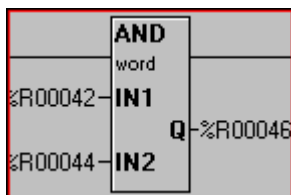
Argumentami tych bloków funkcyjnych mogą być wartości całkowite 16 lub 32-bitowe. Z rozwijalnej listy Type należy wybrać typ Word (16 bitów) lub DWord (32 bity). Obydwa argumenty muszą być jednakowego typu: Word lub DWord.

W większości przypadków, argumentami tych bloków funkcyjnych są rejestry mogące przechowywać zarówno wartości Word jak i DWord, przykładowo %R lub %AI. Możliwe jest jednak również korzystanie z rejestrów typu bitowego (Boolean) poprzez podanie odpowiedniego adresu, np. %Q17. Adres musi być wielokrotnością 16 bitów plus 1, np. można podać 1, 17, 33, itp.

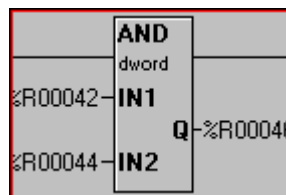
## Dostępne bloki logiczne

### Blok funkcyjny AND

Blok funkcyjny realizuje operację logiczną AND na dwóch rejestrach, umieszczając wynik w trzecim.



Operacja logiczna AND (Word)



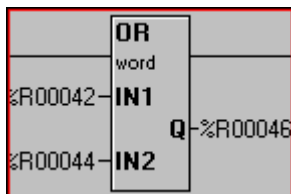
Operacja logiczna AND (DWord)

Przykładowo:

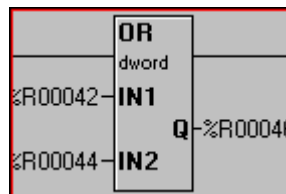
```
%R41 = 0000000000000111 ( 7 )
AND %R42 = 000000000001010 (10 )
RESULT %R43 = 000000000000010 ( 2 )
```

### Blok funkcyjny OR

Blok funkcyjny realizuje operację logiczną OR na dwóch rejestrach, umieszczając wynik w trzecim.



Operacja logiczna OR (Word)



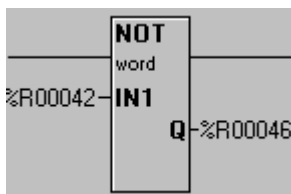
Operacja logiczna OR (DWord)

Przykładowo:

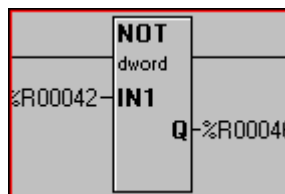
```
%R41 = 0000000000000111 ( 7 )
OR %R42 = 000000000001010 (10)
RESULT %R43 = 000000000001111 (15)
```

### Blok funkcyjny NOT

Blok funkcyjny realizuje operację logiczną NOT na jednym rejestrze, umieszczając wynik w drugim rejestrze.



*Operacja logiczna NOT (Word)*



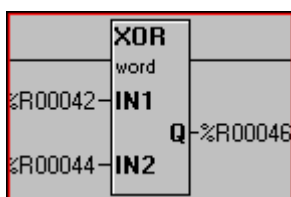
*Operacja logiczna NOT (DWord)*

Przykładowo:

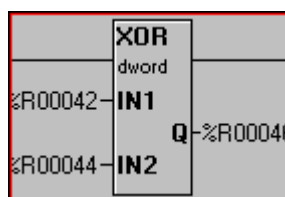
$\%R41 = 0000000000001010$  (10)  
 NOT  $\%R43 = 1111111111110101$  (65525 (bez znaku) lub -11 (ze znakiem))

### Blok funkcyjny Alternatywa wyłączająca OR (EXCLUSIVE OR)

Blok funkcyjny realizuje operację logiczną EXCLUSIVE OR na dwóch rejestrach, umieszczając wynik w trzecim.



*Alternatywa wyłączając OR (Word)*



*Alternatywa wyłączając OR (DWord)*

Przykładowo:

$\%R41 = 0000000011111111$  (255)  
 XOR  $\%R42 = 0000000010100101$  (165)  
 RESULT  $\%R43 = 000000001011010$  (90)

## Funkcje matematyczne

**Uwaga:**

Argumenty funkcji matematycznych mogą być wartościami całkowitymi ze znakiem INT (16-bitowych) lub DINT (32-bitowych), albo wartościami REAL (zmiennoprzecinkowymi).

### Zasada działania

Czasy wykonywania operacji na wartościach INT (16-bitowych) i DINT (32-bitowych) są bardzo zbliżone do siebie, w związku z czym typy te można wzajemnie zamieniać, bez żadnego zauważalnego wpływu na wydajność sterownika Horner APG.

Operacje na wartościach typu REAL (zmiennoprzecinkowych) są bardziej czasochłonne i znacznie wolniejsze, w porównaniu do tych samych operacji wykonywanych na wartościach INT lub DINT. Należy jak najczęściej i wszędzie tam, gdzie to jest możliwe, korzystać z wartości INT i DINT.

Przykładowo, temperatura mierzona przy pomocy termopary jest przekształcana na wartość cyfrową. Wartość wyjściowa z tego modułu jest zawsze typu INT (16-bitowa) pomimo, że reprezentuje nie całkowitą liczbę stopni. Nasuwający się od razu pomysł to przekształcenie tej wartości binarnej na wartość typu Real (stopnie i dziesiąte części stopnia). Niemniej jednak jest bardzo prawdopodobne, że operacje matematyczne będą mogły być zrealizowane na wartości INT, co pozwala na skrócenie czasu wykonywania operacji matematycznych.

Sygnał przepływający przez te bloki funkcyjne ma wartość ON (TRUE), pod warunkiem pomyślnego ich wykonania. W przypadku wystąpienia przekroczenia zakresu, nie doprowadzenia sygnału lub w przypadku próby dzielenia przez zero, sygnał ma wartość OFF (FALSE)

### Konfigurowanie

W celu skonfigurowania bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wprowadzić adresy rejestrów z parametrami oraz rejestru z wynikiem. Warunkiem poprawnego wykonania tego typu bloków funkcyjnych jest podanie trzech (3) poprawnych rejestrów. Jako parametr IN1 lub IN2 można podać stałą bez znaku. Parametr Q musi być zawsze rejestrem.

W polu Type można wybrać typ INT (16 bitów), DINT (32 bity) lub REAL (32 bity). W przypadku operacji na wartościach INT, wykorzystywane są wyłącznie pojedyncze rejestry 16-bitowe (%R43, %AI02, itp.). W przypadku operacji na wartościach DINT (32 bity) lub REAL (32 bity), wykorzystywane są pary 32-bitowe, np. %R43 i %R44, itp.

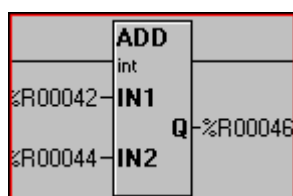
**Uwaga:**

Argumenty i wynik muszą być tego samego typu: INT, DINT lub REAL.

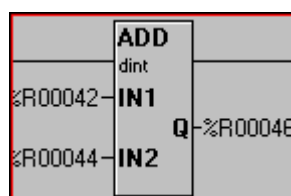


## Dostępne bloki funkcji matematycznych

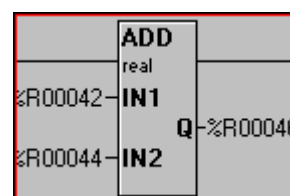
### Blok funkcyjny dodawania (Add)



*Dodawanie (Int)*



*Dodawanie (Dint)*

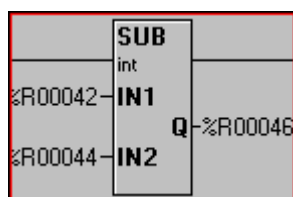


*Dodawanie (Real)*

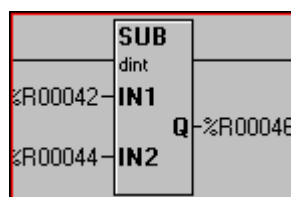
Blok funkcyjny dodaje parametry IN1 i IN2 a wynik zapisuje jako parametr Q.

$$Q = IN1 + IN2$$

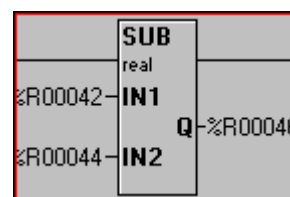
### Blok funkcyjny odejmowania (Sub)



*Odejmowanie (Int)*



*Odejmowanie (Dint)*

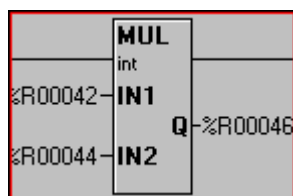


*Odejmowanie (Real)*

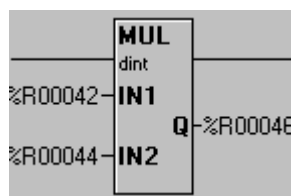
Blok ten odejmuje parametr IN2 od parametru IN1 a wynik zapisuje do parametru Q.

$$Q = IN1 - IN2$$

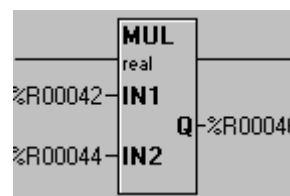
### Blok funkcyjny mnożenia (Mul)



*Mnożenie (Int)*



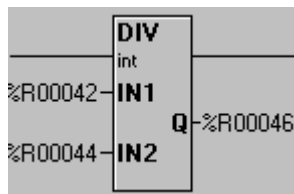
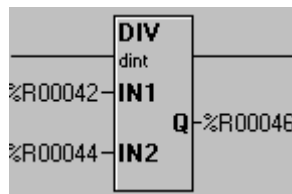
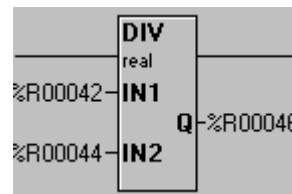
*Mnożenie (Dint)*



*Mnożenie (Real)*

Blok ten mnoży parametry IN1 i IN2 a wynik zapisuje do parametru Q.

$$Q = IN1 * IN2$$

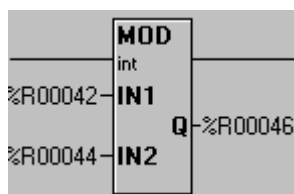
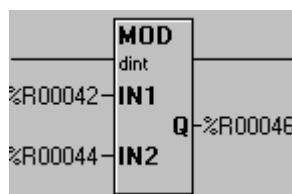
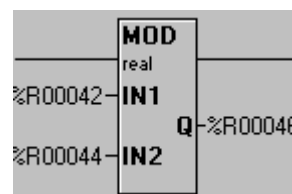
**Blok funkcyjny dzielenia (Div)**

*Dzielenie (Int)*

*Dzielenie (Dint)*

*Dzielenie (Real)*

Blok ten dzieli parametr IN1 przez IN2 a wynik zapisuje do parametru Q.

$$Q = IN1 / IN2$$

W przypadku wartości INT lub DINT, część ułamkowa wyniku jest odrzucana. W zamieszczonej poniżej tabeli podano wyniki dzielenia, dla parametru IN2 równego '5':

IN1	IN2	Q
10	5	2
11	5	2
12	5	2
13	5	2
14	5	2
15	5	3
16	5	3
17	5	3
18	5	3
19	5	3
20	5	4
21	5	4
22	5	4
23	5	4
24	5	4
25	5	5

**Blok funkcyjny dzielenia modulo (MOD)**

*Dzielenie modulo (Int)*

*Dzielenie modulo (Dint)*

*Dzielenie modulo (Real)*

Blok ten dzieli parametr IN1 przez IN2 a resztę z dzielenia zapisuje do parametru Q.

$Q = \text{część ułamkowa (IN1 / IN2)}$

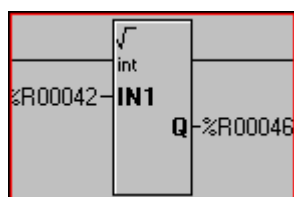
W zamieszczonej poniżej tabeli podano wyniki dzielenia modulo, dla parametru IN2 równego '5':

IN1	IN2	Q
0	5	0
1	5	1
2	5	2
3	5	3
4	5	4
5	5	0
6	5	1
7	5	2
8	5	3
9	5	4
10	5	0
11	5	1
12	5	2
13	5	3
14	5	4

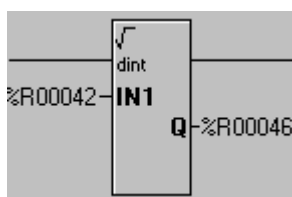
Wynik działania funkcji dzielenia modulo operującej na wartościach typu Real może być zaskakujący, jeżeli zasady działania tej funkcji nie są dostatecznie dobrze znane. Przykładowo  $3.12 \text{ MOD } 2.1 = 1.02$ . Sposób uzyskania tego wyniku przedstawiono na rysunku:

$$\begin{array}{r}
 \textcircled{1} \longleftarrow \text{Część całkowita z dzielenia} \\
 2.1 \overline{) 3.12} \\
 \underline{2.1} \\
 1.02 \\
 \textcircled{1.02} \longleftarrow \text{Wynik dzielenia modulo}
 \end{array}$$

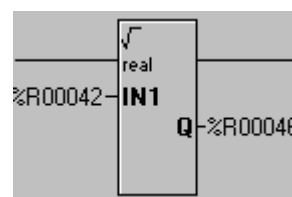
### Blok funkcyjny pierwiastek kwadratowy ( $\sqrt{\quad}$ )



Pierwiastek kwadratowy  
(Int)



Pierwiastek kwadratowy  
(Dint)



Pierwiastek kwadratowy  
(Real)

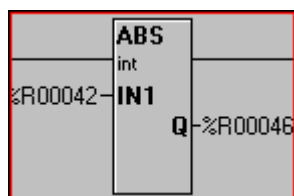
Blok ten oblicza pierwiastek kwadratowy z parametru IN1 a wynik zapisuje do parametru Q.

$$Q = \text{Pierwiastek Kwadratowy}(\text{IN1})$$

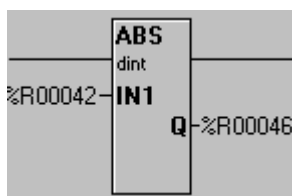
Argumentami tego bloku funkcyjnego są zwykle wartości typu REAL. Jako argument można podać wartości typu INT lub DINT, ale wynik działania tej funkcji jest rzadko liczbą całkowitą. W przypadku, gdy jako parametr Q podana zostanie wartość całkowita, część ułamkowa jest odrzucana.

IN1	Wynik rzeczywisty	Q
0	0	0
1	1	1
2	1.414	1
3	1.73	1
4	2	2
5	2.236	2
6	2.449	2
7	2.645	2
8	2.828	2
9	3	3
10	3.162	3
11	3.316	3
12	3.464	3
13	3.605	3
14	3.741	3
15	3.872	3
16	4	4

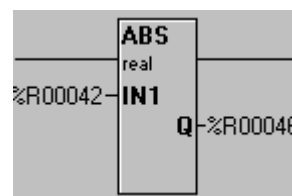
### Blok funkcyjny wartość bezwzględna (ABS)



Wartość bezwzględna (Int)



Wartość bezwzględna (Dint)



Wartość bezwzględna (Real)

Blok ten odrzuca znak parametru IN1 a wynik zapisuje do parametru Q.

$$Q = \text{Wartość Bezwzględna}(\text{IN1})$$

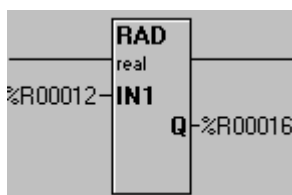
Wynik działania tego bloku funkcyjnego jest zawsze liczbą dodatnią.

## Dostępne bloki zaawansowanych funkcji matematycznych

### Uwaga:

Zaawansowane funkcje matematyczne korzystają wyłącznie z danych typu REAL (zmiennoprzecinkowych).

### Blok funkcyjny konwersja ze stopni na radiany (RAD)

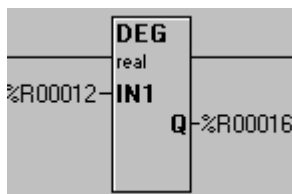


*Konwersja ze stopni na radiany (RAD)*

Wartość parametru IN1 jest przekształcana ze stopni na radiany, a wynik jest zapisywany do parametru Q. Parametr IN1 zawiera wartość w stopniach. Parametr Q zawiera wartość w radianach.

$$Q = \text{RAD}(\text{IN1})$$

### Blok funkcyjny Konwersja z radianów na stopnie (DEG)

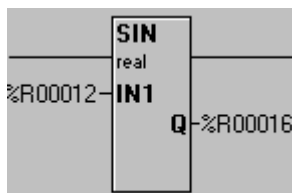


*Konwersja z radianów na stopnie (DEG)*

Wartość parametru IN1 jest przekształcana z radianów na stopnie, a wynik jest zapisywany do parametru Q. Parametr IN1 zawiera wartość w radianach. Parametr Q zawiera wartość w stopniach.

$$Q = \text{DEG}(\text{IN1})$$

### Blok funkcyjny Sinus (Sin)

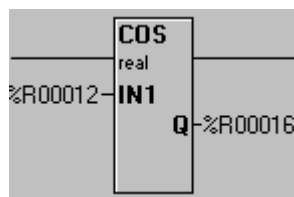


*Sinus*

Sinus z wartości IN1 jest zapisywany do parametru Q. Wartości parametru IN1 muszą być podane w radianach. Parametr Q ma zawsze wartość z przedziału -1 do +1.

$$Q = \text{SIN}(\text{IN1})$$

### Blok funkcyjny Cosinus (Cos)

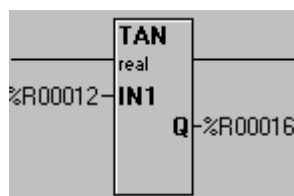


*Cosinus*

Cosinus z wartości IN1 jest zapisywany do parametru Q. Wartości parametru IN1 muszą być podane w radianach. Parametr Q ma zawsze wartość z przedziału -1 do +1.

$$Q = \text{COSIN}(IN1)$$

### Blok funkcyjny Tangens (Tan)

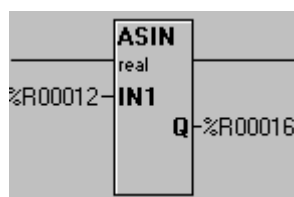


*Tangens*

Tangens z wartości IN1 jest zapisywany do parametru Q. Wartości parametru IN1 muszą być podane w radianach.

$$Q = \text{TAN}(IN1)$$

### Blok funkcyjny Arcus sinus (ASin)

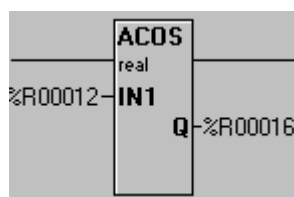


*Arcus Sinus*

Arcus sinus dla wartości IN1 jest zapisywany do parametru Q. Parametr IN1 musi mieć wartość z przedziału -1 do +1. Wartość parametru Q jest zawsze podawana w radianach.

$$Q = \text{ASIN}(IN1)$$

### Blok funkcyjny Arcus cosinus (ACos)

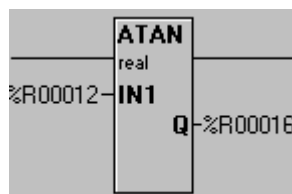


*Arcus cosinus*

Arcus cosinus dla wartości IN1 jest zapisywany do parametru Q. Parametr IN1 musi mieć wartość z przedziału -1 do +1. Wartość parametru Q jest zawsze podawana w radianach.

$$Q = \text{ACOSIN}(IN1)$$

### Blok funkcyjny Arcus tangens (ATan)

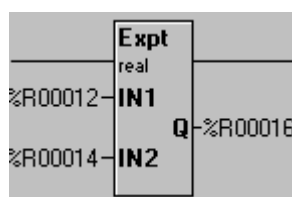


*Arcus tangens*

Arcus tangens z wartości IN1 jest zapisywany do parametru Q. Wartość parametru Q jest zawsze podawana w radianach.

$$Q = \text{ATAN}(IN1)$$

### Blok funkcyjny funkcja wykładnicza (EXPT)

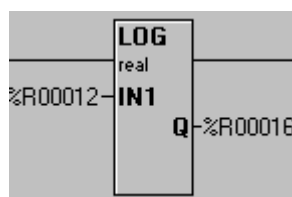


*Funkcja wykładnicza (EXPT)*

Funkcja ta podnosi parametr IN1 do potęgi IN2 a wynik zapisuje do parametru Q.

$$Q = IN1^{IN2}$$

### Blok funkcyjny logarytm dziesiętny (LOG)

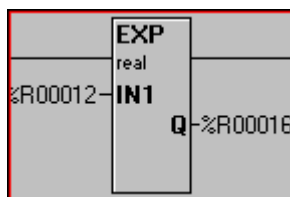


*Logarytm dziesiętny*

Funkcja ta wyznacza logarytm dziesiętny (o podstawie 10) z wartości IN1 a wynik przekazuje za pomocą parametru Q.

$$Q = \text{LOG}(IN1)$$

### Blok funkcyjny potęgowanie przy podstawie e (EXP)

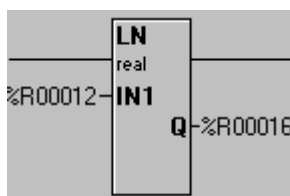


*Potęgowanie przy podstawie e*

Funkcja ta wyznacza wartość z liczby e (podstawa logarytmu naturalnego) podniesionej do potęgi IN1 a wynik umieszcza w parametrze Q.

$$Q = e^{IN1}$$

### Blok funkcyjny logarytm naturalny (LN)

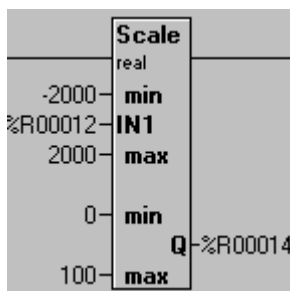


*Logarytm naturalny*

Funkcja ta wyznacza logarytm naturalny z IN1 a wynik zapisuje do parametru Q.

$$Q = \text{LN}(IN1)$$

### Blok funkcyjny skalowanie (SCALE)



*Skalowanie*

#### **Uwaga:**

Skalowanie może być przeprowadzane wyłącznie na wartościach typu REAL. W przypadku wartości całkowitych, wymagane jest ich przekształcenie na typ REAL, a później z reguły wynik musi być przekształcony ponownie na wartość typu całkowitego.

Potrzeba skalowania występuje często w przypadku konieczności wyrażenia wartości w innym zakresie. Przykładowo, może występować potrzeba przekształcenia wartości z czujnika poziomu na wartość w skali 0-100. Proces ten nazywa się **skalowaniem**.

W oknie dialogowym Scaling Element, w grupie Input, należy wpisać adres lub wybrać nazwę zmiennej.



**Uwaga:**

Musi to być zmienna typu REAL (32-bitowa), wykorzystywane są więc dwa sąsiadujące ze sobą rejestry 16-bitowe.

W grupie Input, w polach Minimum i Maximum należy podać spodziewaną wartość minimalną i maksymalną zmiennej wejściowej. W grupie Output Value wpisać adres lub wybrać nazwę zmiennej.

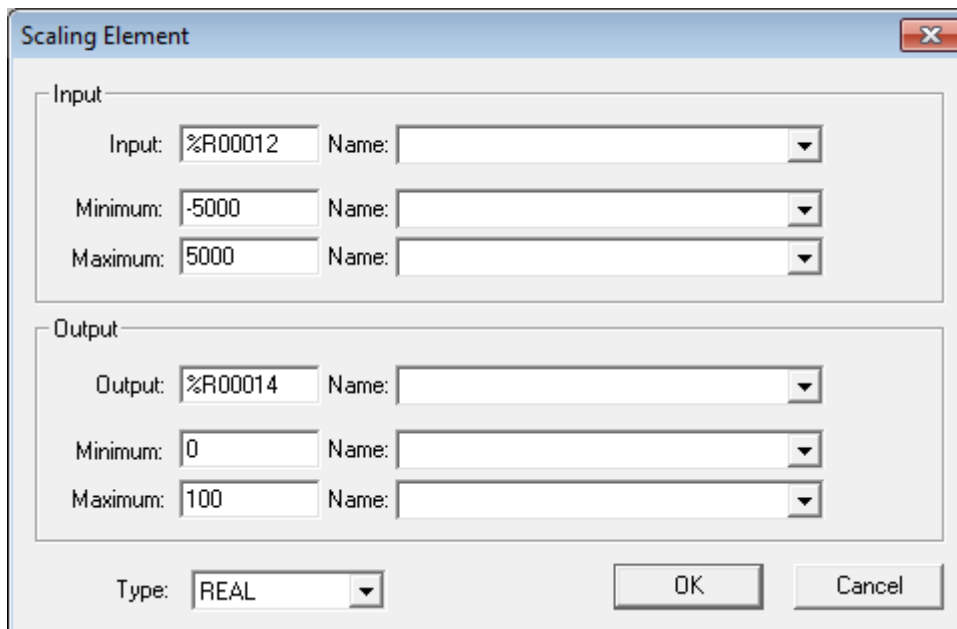
**Uwaga:**

Musi to być zmienna typu REAL (32-bitowa), wykorzystywane są więc dwa sąsiadujące ze sobą rejestry 16-bitowe.

Pola Minimum i Maximum w grupie Output określają zakres wartości, do którego konwertowana będzie zmienna wejściowa.

Przykładowo założymy, że monitorowany jest poziom napełnienia zbiornika lub poziom cieczy. Urządzenie monitorujące wysyła wartość z zakresu -5000 (pusty zbiornik) do +5000 (zbiornik wypełniony). Wartości te należy przekonwertować do przedziału od 0 (zero) do 100 procent. W tym celu, w oknie dialogowym Scaling Element należy wpisać następujące dane:

Przykład skalowania wartości



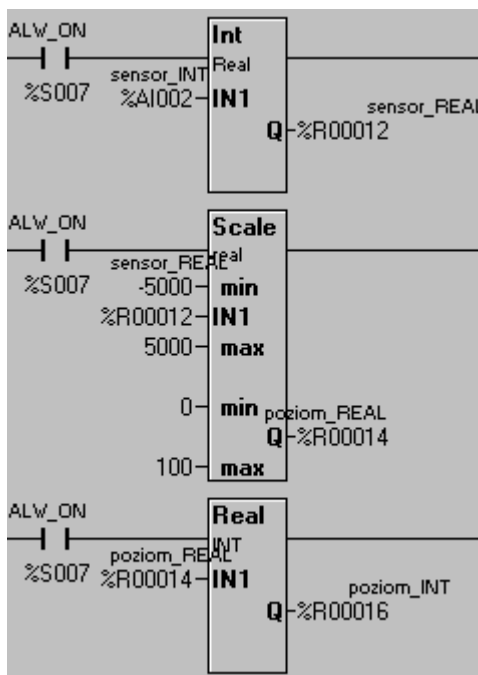
The screenshot shows a dialog box titled "Scaling Element". It contains the following fields and values:

- Input section:**
  - Input: %R00012
  - Minimum: -5000
  - Maximum: 5000
- Output section:**
  - Output: %R00014
  - Minimum: 0
  - Maximum: 100
- Type:** REAL
- Buttons:** OK, Cancel

**Uwaga:**

Blok funkcyjny Scale operuje na wartościach typu Real. Większość przetworników generuje wartości całkowite, a więc najpierw należy przekształcić argument na wartość typu Real.

Jeżeli wynik ma być liczbą całkowitą, wartość wyjściowa tego bloku funkcyjnego musi być przed jej wykorzystaniem przekształcona z typu Real na typ całkowity. W wielu przypadkach jednak akceptowane są wartości wyjściowe typu Real.



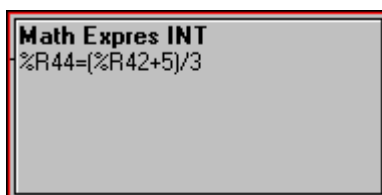
## Wyrażenia matematyczne

### Uwaga:

Blok funkcyjny Math Expres operuje na wartościach całkowitych ze znakiem (16 lub 32-bitowych) lub na wartościach typu Real.

## Złożone obliczenia matematyczne

Oprogramowanie Cscape zawiera blok funkcyjny, pozwalający na przeprowadzanie skomplikowanych operacji matematycznych przy zastosowaniu standardowej notacji matematycznej. Umożliwia to zredukowanie lub wyeliminowanie wielu szczebli programu, przez co program wynikowy jest prostszy i łatwiejszy do zrozumienia.



Wyrażenia matematyczne

## Przepływ sygnału

Po przejściu przez ten blok funkcyjny sygnał ma wartość ON (TRUE), jeżeli zostało poprawnie wyliczone wyrażenie matematyczne. W przypadku wystąpienia błędów (np. próba dzielenia przez zero), sygnał po przejściu przez blok funkcyjny będzie miał wartość OFF (FALSE).

W przypadku wystąpienia jakichkolwiek błędów matematycznych wartość umieszczona po lewej stronie równania jest niepoprawna.

## Konfigurowanie

W celu skonfigurowania tego bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wpisać równanie w następującym formacie:

[wynik] = [wyrażenie],	gdzie:
[wynik]	rejestr, zwykle jest to %R
=	OBOWIĄZKOWY znak równości
[wyrażenie]	formuła matematyczna.

SUMARYCZNA długość łańcucha znaków z wyrażenia nie może przekraczać 80 znaków. Przy obliczaniu tej długości należy uwzględnić adres rejestru z wynikiem, znak równości (%R5 =) i samo równanie.

Argumentami całego równania mogą być wartości zarówno INT (16-bitowe), DINT (32-bitowe) lub REAL. Należy zwrócić uwagę, że wszystkie zmienne w równaniu muszą być tego samego typu.

### Uwaga:

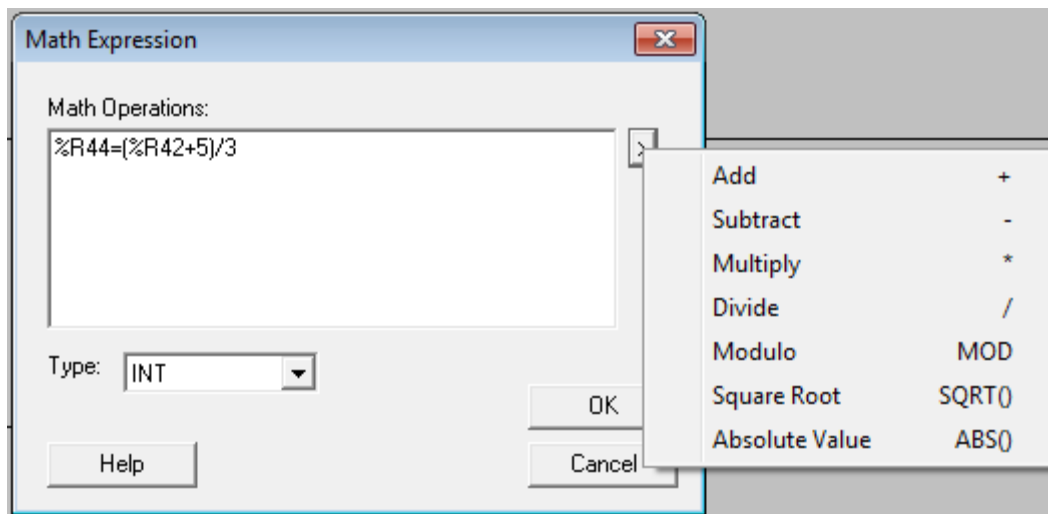
W jednym równaniu nie można korzystać z wartości różnych typów, np. INT, DINT lub Real.

### Uwaga:

Zawsze należy podawać znak mnożenia. Równanie  $\%R4 = 4(\%R1 + 4)$  ma NIEPOPRAWNĄ składnię. Należy je zapisać w postaci:  $\%R4 = 4 * (\%R1 + 4)$ .

## Szybkie wywoływanie funkcji matematycznych

W celu zmniejszenia czasochłonności oraz zredukowania możliwości popełnienia błędu w czasie wpisywania, dostępne funkcje matematyczne można wywoływać za pomocą menu More . W tym celu najpierw należy umieścić kursor w równaniu, w miejscu w którym ma być wstawiona funkcja, a następnie kliknąć przycisk More . Wyświetlone zostanie następujące menu:



MENU MORE

Po wstawieniu funkcji należy w razie potrzeby wprowadzić odpowiednie zmiany.

## Oznaczenia rejestrów

Wynik równania musi być zawsze zapisywany do rejestru. Zwykle jest do tego celu wykorzystywany rejestr %R, chociaż można również korzystać z innych rejestrów (np. %AI,...). Wielkość rejestru (16 lub 32 bity) jest określana przez pole TYPE. Można również korzystać z 16 lub 32-bitowych grup rejestrów typu Boolean (np. podając %Q17, a więc określając rejestr %Q17 - %Q32).

W celu określenia rejestru można podać jego predefiniowaną nazwę lub adres.

`Temp_Result = [wyrażenie]` może być równoważne: `%R10 = [wyrażenie]`

### Uwaga:

W celu odróżnienia wyrażenia od nazwy zmiennej adres musi być zawsze poprzedzony znakiem % (procent).

## Stałe numeryczne

Aby wprowadzić stałą numeryczną wystarczy ją wpisać:

(\* Blok funkcyjny konwertuje wartość ze skali stustopniowej na skalę Fahrenheita \*)

`%R22 = (%R15 * (9/5)) + 32`

### Uwaga:

W przypadku przeprowadzania operacji matematycznych na zmiennych typu INT lub DINT może nie zostać uzyskany oczekiwany wynik.

## Kolejność wykonywania działań matematycznych

Równania należy wpisywać zgodnie ze standardową notacją matematyczną. Stosowana jest następująca kolejność operacji:

ABS()	Najwyższy priorytet
SQRT()	
LOG()	
EXP()	
LN()	
SIN()	
COS()	
TAN()	
ASIN()	
ACOS()	
ATAN()	
DEG()	
RAD()	
EXPT	
* (mnożenie)	
/ (dzielenie)	
MOD	
+ (dodawanie)	
-	
(odejmowanie)	
=	Najniższy priorytet

Kolejność wykonywania operacji można zmienić za pomocą nawiasów ( ). Nawiasy można dowolnie zagnieżdżać pod warunkiem, że nie nastąpi przekroczenie dopuszczalnej długości równania, wynoszącej 80 znaków.

W celu zilustrowania, poniżej zamieszczono poprawne równanie matematyczne:

$$\%R22 = (\%R15 * \%R16) + (\%R15 / \%R16)$$

Jeżeli %R15 zawiera 25 a %R16 zawiera 5, to w rejestrze %R22 zapisana zostanie po wykonaniu równania wartość 130.

## Bloki funkcyjne do porównywania

### Uwaga:

Bloki funkcyjne do porównywania (EQ\_INT, itp.) operują na wartościach typu INT, liczbach całkowitych ze znakiem (16-bitowych), -32768 do +32768, wartościach całkowitych podwójnej długości ze znakiem (32-bitowych), -2147483648 do +2147483647 i wartościach typu Real (rzeczywistych), +/-3.40282e-38 do +/-3.40282e+38.

### Informacje ogólne

Bloki funkcyjne tego typu porównują dwie wartości typu INT ze znakiem (16-bitowe), DINT ze znakiem (32-bitowe) lub Real (32-bitowe).

### Przepływ sygnału

Jeżeli wynik porównania ma wartość True, na wyjście bloku funkcyjnego przesyłany jest sygnał, co można wykorzystać do ustawienia lub skasowania wartości przełącznika. Przykładowo:

Wartość		Operacja porównania	Przepływ sygnału
IN1=6	IN2=3	IN1>IN2	TRUE
IN1=6	IN2=3	IN1<IN2	FALSE
IN1=6	IN2=3	IN1=IN2	FALSE
IN1=3	IN2=6	IN1>IN2	FALSE
IN1=3	IN2=6	IN1<IN2	TRUE
IN1=3	IN2=6	IN1=IN2	FALSE

### Konfigurowanie

W celu skonfigurowania bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wprowadzić adresy rejestrów z parametrami.

Argumenty mogą być stałymi typu Integer, Double Integer lub Real (1, 1,23056, 45, itp.). Nawet obydwa argumenty wejściowe mogą być stałymi, ale wynik w takiej sytuacji byłby zawsze taki sam, TRUE (FALSE).

Z rozwijalnej listy Type należy wybrać typ zmiennej. Obydwa argumenty muszą być tego samego typu.

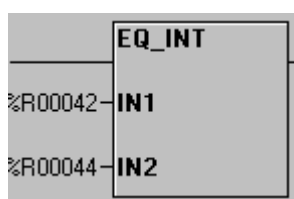
### Uwaga:

W jednej operacji porównania nie można korzystać z rejestrów różnych typów.

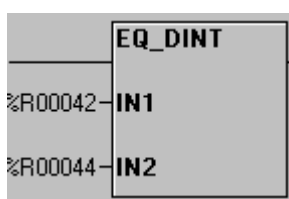
W większości przypadków, argumentami tych bloków funkcyjnych są rejestry mogące przechowywać zarówno wartości INT jak i DINT, przykładowo %R lub %AI. Możliwe jest jednak korzystanie z rejestrów typu bitowego (Boolean) poprzez podanie odpowiedniego adresu np. %Q17. Adres musi być wielokrotnością 16 bitów plus 1, np. 1, 17, 33, itp. Po zakończeniu konfigurowania, blok funkcyjny pokazuje jaki typ danych został wybrany.

## Dostępne bloki funkcyjne do porównania

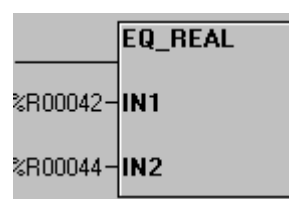
### Blok funkcyjny porównania czy równe (EQ)



*Równy (Int)*



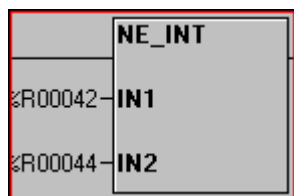
*Równy (Dint)*



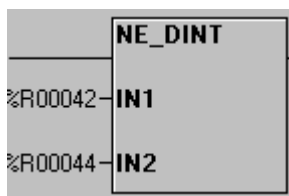
*Równy (Real)*

Blok funkcyjny EQ (EQUAL) porównuje dwie wartości i przesyła sygnał, jeżeli wartości te są sobie równe. Argumenty mogą być stałymi, można też podać adresy rejestrów.

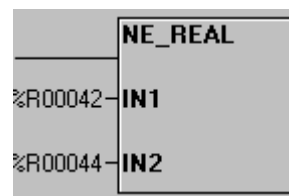
### Blok funkcyjny porównania czy różne (NE)



*Nie równy (Int)*



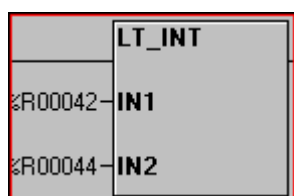
*Nie równy (Dint)*



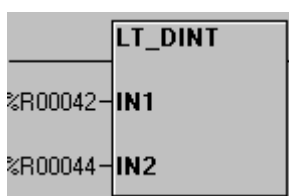
*Nie równy (Real)*

Blok funkcyjny NE (NOT EQUAL) porównuje dwie wartości i przesyła sygnał, jeżeli wartości te są różne od siebie. Argumenty mogą być stałymi, można też podać adresy rejestrów.

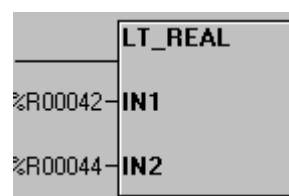
### Blok funkcyjny porównania czy mniejsze (LT)



*Mniejszy (Int)*



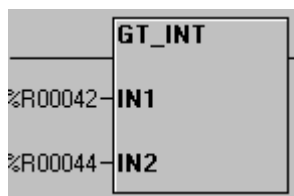
*Mniejszy (Dint)*



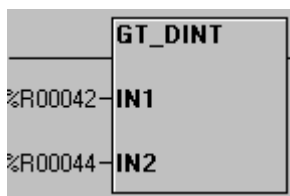
*Mniejszy (Real)*

Blok funkcyjny LT (LESS THAN) porównuje dwie wartości i przesyła sygnał, jeżeli wartość IN1 jest mniejsza od IN2. Argumenty mogą być stałymi, można też podać adresy rejestrów.

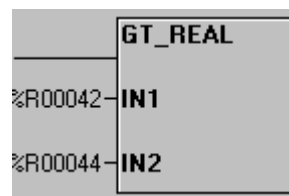
### Blok funkcyjny porównania czy większe (GT)



*Większy (Int)*



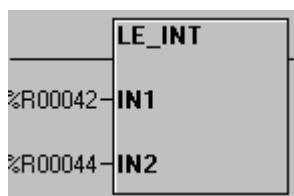
*Większy (Dint)*



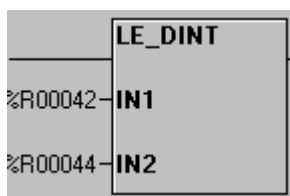
*Większy (Real)*

Blok funkcyjny GT (GREATER THAN) porównuje dwie wartości i przesyła sygnał, jeżeli wartość IN1 jest większa od IN2. Argumenty mogą być stałymi, można też podać adresy rejestrów.

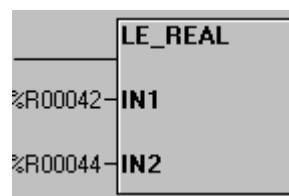
### Blok funkcyjny porównania mniejszy lub równy (LE)



*Mniejszy lub równy (Int)*



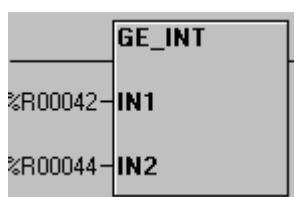
*Mniejszy lub równy (Dint)*



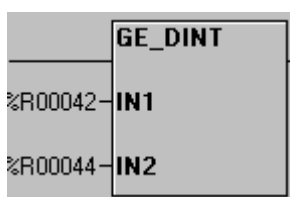
*Mniejszy lub równy (Real)*

Blok funkcyjny LE (LESS OR EQUAL) porównuje dwie wartości i przesyła sygnał, jeżeli wartość IN1 jest mniejsza lub równa wartości IN2. Argumenty mogą być stałymi, można też podać adresy rejestrów.

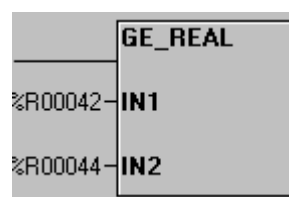
### Blok funkcyjny porównania większy lub równy GE



*Większy lub równy (Int)*



*Większy lub równy (Dint)*

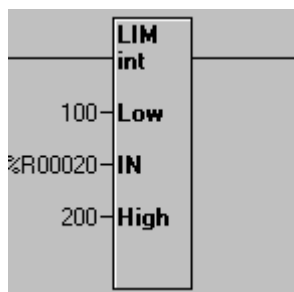


*Większy lub równy (Real)*

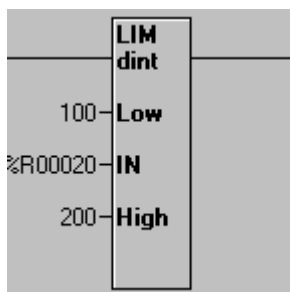
Blok funkcyjny GE (GREATER OR EQUAL) porównuje dwie wartości i przesyła sygnał, jeżeli wartość IN1 jest większa lub równa wartości IN2. Argumenty mogą być stałymi, można też podać adresy rejestrów.



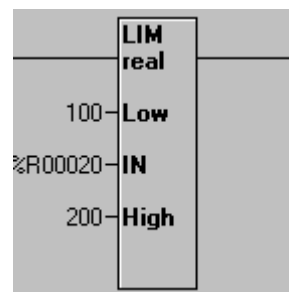
### Blok funkcyjny granicy (LIM)



*Granica (Int)*



*Granica (Dint)*

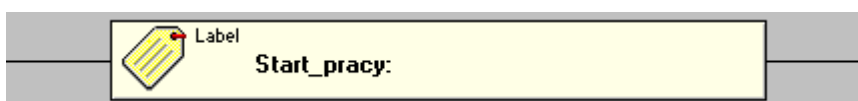


*Granica (Real)*

Blok funkcyjny LIM (Limens) porównuje czy wartość IN mieści się w zadanym zakresie Low i High. Argumenty mogą być stałymi, można też podać adresy rejestrów.

## Sterowanie wykonywaniem programu: Jump, Label, Call, Return i End

### Element Label

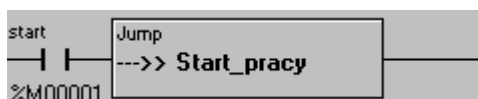


Blok ten pozwala na przypisanie nazwy do pozycji w drabince logicznej. Nazwa ta może być następnie podana w instrukcji JUMP lub CALL w celu określenia miejsca, do którego należy przenieść wykonywanie programu.

#### Uwaga:

W programie sterującym nie mogą wystąpić dwie takie same etykiety. Etykiety Label nie muszą posiadać odpowiadających im instrukcji skoku, ale każda instrukcja skoku musi posiadać odpowiadającą jej etykietę

### Blok funkcyjny Jump



Blok funkcyjny JUMP powoduje pominięcie fragmentu części logicznej programu sterującego, instrukcja JUMP może powodować przejście zarówno w kierunku do przodu, jak i do tyłu programu sterującego. Wykonywanie programu zostanie wznowione od miejsca oznaczonego etykietą LABEL.

Gdy do bloku funkcyjnego JUMP dożywa sygnał wejściowy, wszystkie przełączniki zawarte w obejmowanym przez tę instrukcję obszarze zachowują swój pierwotny stan, tzn. po instrukcji Jump nie wolno umieszczać żadnych bloków funkcyjnych. Po napotkaniu instrukcji JUMP, program jest dalej wykonywany od miejsca wskazywanego przez etykietę.

**Uwaga:**

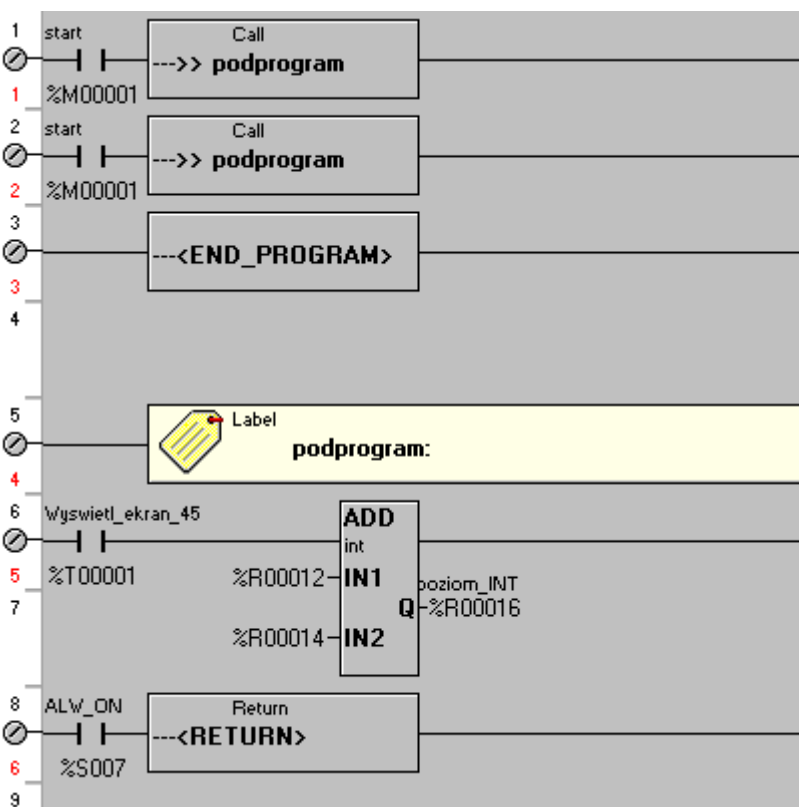
Aby uniknąć zapętlenia programu sterującego przy użyciu instrukcji skoku w przód i w tył, z instrukcją skoku w tył musi być związany warunek logiczny.

## Blok funkcyjny Call



Blok funkcyjny CALL pozwala na wywołanie podprogramu. Po doprowadzeniu sygnału do bloku funkcyjnego CALL, wykonywanie programu zostaje przeniesione do miejsca wskazywanego przez etykietę podaną w tej instrukcji. Następnie, po napotkaniu instrukcji RETURN, następuje przejście do szczebla, umieszczonego za instrukcją CALL. Podprogramy można zagnieżdżać (wywoływać podprogram z poziomu innego podprogramu) aż do 8 poziomów. W przypadku przekroczenia tej liczby, nastąpi zatrzymanie sterownika oraz zgłoszenie błędu. Za instrukcją CALL nie wolno umieszczać żadnych bloków funkcyjnych.

Przykład:



1. Rozpoczęcie wykonywania programu od szczebla 1, wywołanie podprogramu (CALL).
2. Wykonanie pierwszej linii podprogramu, szczebel 5 (szczebel 4 zawiera wyłącznie samą etykietę).
3. Wykonanie szczebla 6, napotkanie instrukcji RETURN i powrót do wykonywania szczebla umieszczonego po instrukcji CALL, czyli szczebla 2.
4. Wykonanie szczebla 2, ponowne wywołanie (CALL) podprogramu.
5. Wykonanie pierwszej linii podprogramu, szczebel 5 (szczebel 4 zawiera wyłącznie samą etykietę).
6. Wykonanie szczebla 6, napotkanie instrukcji RETURN i powrót do wykonywania szczebla umieszczonego po instrukcji CALL, czyli szczebla 3.
7. Wykonanie szczebla 3, napotkanie instrukcji END PROGRAM kończącej cykl wykonywania programu sterującego. Po uaktualnieniu wejść/wyjść i zrealizowaniu innych czynności, program sterujący będzie ponownie wykonywany od szczebla 1.

## Blok funkcyjny Return



Blok funkcyjny RETURN kończy podprogram. Po doprowadzeniu sygnału, następuje przejście do wykonywania części programu sterującego, umieszczonej po ostatnio wywoływanej instrukcji CALL.

W przypadku napotkania instrukcji RETURN, jeżeli wcześniej nie była wykonana instrukcja CALL, nastąpi zatrzymanie sterownika oraz zgłoszenie błędu. Po instrukcji RETURN nie wolno umieszczać żadnych bloków funkcyjnych.

## Blok funkcyjny End\_Program



Instrukcja ta informuje o końcu programu sterującego. Przed blokiem tym nie musi być umieszczany styk. Po napotkaniu tego bloku funkcyjnego następuje natychmiastowe przerwanie wykonywania programu sterującego, następnie odczytywany jest stan wejść/wyjść, wykonywane są pozostałe czynności realizowane w każdym cyklu pracy jednostki centralnej, po czym następuje ponowne rozpoczęcie wykonywania programu sterującego. Instrukcja ta pozwala na oddzielenie części głównej programu od podprogramów, tak jak to zrobiono w zamieszczonym powyżej przykładzie lub może być wykorzystana w czasie testowania do przejściowego anulowania części programu sterującego.

## Konwersja typów zmiennych

### Informacje ogólne

Bloki funkcyjne do konwersji pozwalają na zamianę typów danych. Główne typy danych to INT (16-bitowy), DINT (32-bitowy) i REAL (32-bitowy). Przykładowo konwersja danych jest wymagana jeżeli zmienna analogowa typu DINT musi być przekonwertowana na zmienną typu Real, przed jej skalowaniem.

**Uwaga:**

Parametry wejściowe i wyjściowe nie mogą być stałymi numerycznymi.

### Zasady konwersji typów

Operacja konwersji dotyczy wartości, a nie miejsca zajmowanego przez zmienną. Wszystkie wartości typu INT można konwertować na wartości typu DINT lub REAL. Niektóre z wartości DINT (32-bitowe) można z powodzeniem konwertować na wartości typu INT (16-bitowe). Podobnie, niektóre wartości Real (32-bitowe) można konwertować na wartości typu DINT (32-bitowe) lub INT (16-bitowe).

**Uwaga:**

Wartości po konwersji muszą mieścić się i być zgodne co do formatu określonego rejestru docelowego, całą odpowiedzialność za spełnienie tego wymogu ponosi programista.

W niektórych przypadkach może nastąpić zmniejszenie dokładności. Przykładowo, w czasie konwersji wartości typu DINT zawierającej 7 cyfr ("2123789") na REAL, wartość ta będzie reprezentowana z dokładnością do sześciu cyfr, pamiętanych w liczbach rzeczywistych ("2.12378E+06").

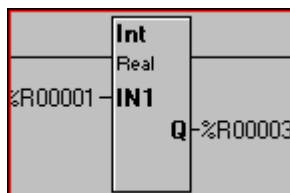
Może w takich sytuacjach nastąpić utrata części danych. Przy konwersji REAL na INT zaokrąglana jest część ułamkowa (REAL)1.23654E+02 = (INT)124.

Zmniejszenie dokładności w czasie konwersji NIE jest traktowane jako błąd. Blok funkcyjny nadal normalnie funkcjonuje, ale bloki funkcyjne umieszczone w dalszej części programu sterującego, korzystające z przekonwertowanych wartości, mogą dać nieoczekiwane wyniki.

Błąd jest natomiast sygnalizowany w przypadku podjęcia próby konwersji dużej wartości na typ, który nie jest w stanie reprezentować tej wartości. Sytuacja taka występuje najczęściej przy konwersji DINT na INT, ale może również wystąpić przy konwersji REAL na INT lub DINT. W takim przypadku sygnał nie przepływa przez blok funkcyjny, a wynik działania jest nieokreślony.

## Dostępne bloki do konwersji typów zmiennych

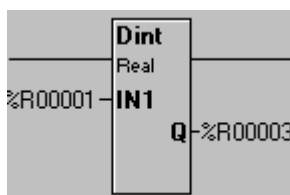
### Blok funkcyjny Konwersja INT na REAL



*Konwersja Int na Real*

Blok ten konwertuje wartość INT (16-bitową) zadaną parametrem IN1 na wartość REAL (32-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametr IN1 jest wartością 16-bitową, a parametr Q jest wartością 32-bitową.

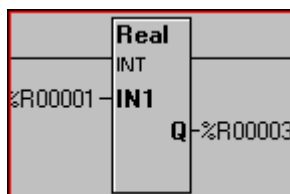
### Blok funkcyjny konwersja DINT na REAL



*Konwersja DINT na REAL*

Blok ten konwertuje wartość DINT (32-bitową) zadaną parametrem IN1 na wartość REAL (32-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametry IN1 oraz Q są wartościami 32-bitowymi.

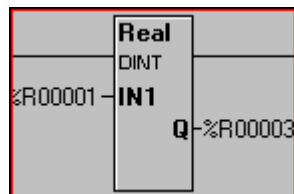
### Blok funkcyjny konwersja REAL na INT



*Konwersja Real na Int*

Blok ten konwertuje wartość REAL (32-bitową) zadaną parametrem IN1 na wartość INT (16-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametr IN1 jest wartością 32-bitową, a parametr Q jest wartością 16-bitową.

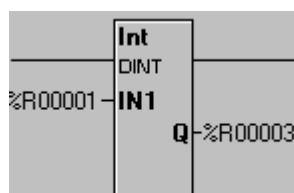
### Blok funkcyjny Konwersja REAL na DINT



*Konwersja Real na Dint*

Blok ten konwertuje wartość REAL (32-bitową) zadaną parametrem IN1 na wartość DINT (32-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametr IN1 oraz Q są wartościami 32-bitowymi.

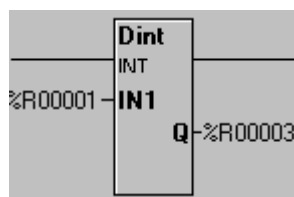
### Blok funkcyjny konwersja INT na DINT



*Konwersja Int na Dint*

Blok ten konwertuje wartość INT (32-bitową) zadaną parametrem IN1 na wartość DINT (32-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametr IN1 jest wartością 16-bitową, a parametr Q jest wartością 32-bitową R

### Blok funkcyjny konwersja DINT na INT



*Konwersja Dint na Int*

Blok ten konwertuje wartość DINT (32-bitową) zadaną parametrem IN1 na wartość INT (16-bitową), zapisywaną w parametrze Q. Należy zwrócić uwagę, że parametr IN1 jest wartością 32-bitową, a parametr Q jest wartością 16-bitową.

## Liczniki i przełączniki czasowe

Dane liczników/przełączników czasowych są przechowywane w dwóch (2) kolejnych rejestrach 16-bitowych.

<code>%[rejestr]</code>	=	Wartość akumulatora
<code>%[r%[rejestr + 1].15]</code>	=	Dalszy przepływ sygnału Jeżeli Bit 15 = 1, blok funkcyjny jest aktywowany.
<code>%[rejestr + 1].16</code>	=	Dalszy przepływ sygnału Jeżeli Bit 16 = 1, sygnał przepływa przez blok funkcyjny.

Przykładowo, jeżeli rejestr %R14 zostanie skonfigurowany jako licznik, %R14 jest akumulatorem zliczającym upływ czasu, %R15.16 jest bitem przepływu sygnału %R15.15 zawiera informacje czy licznik jest aktywny.

**Uwaga:**

Zarówno licznik jak i przełącznik czasowy mogą korzystać z tych samych lub pokrywających się rejestrów. Pomimo tego zalecane jest unikanie takiej sytuacji.

## Operacje bitowe

**Uwaga:**

Bloki funkcyjne do operacji bitowych wykorzystują argumenty całkowite typu WORD (16-bitowe) lub DWORD (32-bitowe)

## Informacje ogólne

Operacje logiczne są przeprowadzane na ciągach bitów zapisanych w rejestrach z argumentami. Wynik operacji jest zapisywany w trzecim rejestrze. Zawartość rejestrów zawierających argumenty nie ulega zmianie.

Przepływ sygnału przez bloki funkcyjne do przemieszczania słowa bitowego (SHIFT) zależy od ostatniego z bitów, usuwanego z rejestru. Przykładowo, jeżeli rejestr %R41 zawiera wartość 21770 (0101010100001010), a liczba przemieszczanych bitów wynosi 4:

Faza	Przeptyw sygnалу	Wartość	
0	0	010101010000101	bez przesunięcia
1	0	101010100001010	
2	0	010101000010100	
3	0	101010000101000	
4	1	010100001010000	

Przeptyw sygnалу jest określany po wykonaniu ostatniego przemieszczenia. Wszystkie poprzednie bity nie mają żadnego wpływu na przeptyw sygnалу i są tracone.

Sygnal zawsze przepływa przez bloki funkcyjne do przemieszczania słowa bitowego w obiegu zamkniętym (ROTATE), niezależnie od wartości ostatniego z bitów.

## Konfigurowanie

W celu skonfigurowania bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wprowadzić adres rejestru zawierającego argument, rejestru przechowującego wynik oraz liczbę przemieszczeń ("N").

Argument wejściowy i liczba przemieszczeń ("N") mogą być stałymi typu INT lub DINT. Parametr zawierający wynik ("Q") musi być zawsze rejestrem (%R, itp.).

Bloki te mogą wykorzystywać rejestry 16-bitowe lub 32-bitowe.

### Uwaga:

Jeżeli jako argument dla bloku funkcyjnego Shift podany zostanie rejestr Word, a liczba przemieszczeń (N) będzie większa od 15, wszystkie bity tego rejestru zostaną ustawione na 0.

W przypadku bloku funkcyjnego Rotate, dla liczby przemieszczeń N=16, w wyniku otrzymana zostanie oryginalna wartość.

Jeżeli jako argument dla bloku funkcyjnego Shift podany zostanie rejestr DWORD, a liczba przemieszczeń (N) będzie większa od 31, wszystkie bity tego rejestru zostaną ustawione na 0. W przypadku bloku funkcyjnego Rotate, dla liczby przemieszczeń N=32, w wyniku otrzymana zostanie oryginalna wartość.

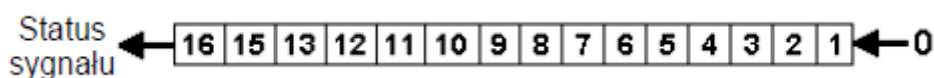
## Różnice pomiędzy blokami funkcyjnymi Shift i Rotate

Różnica pomiędzy tymi blokami funkcyjnymi polega na sposobie traktowania bitów wychodzących poza rejestr. W przypadku bloku funkcyjnego Shift, wysuwane poza rejestr bity są tracone, za wyjątkiem ostatniego bitu, który określa przeptyw sygnалу przez blok funkcyjny. Jeżeli ostatnim z



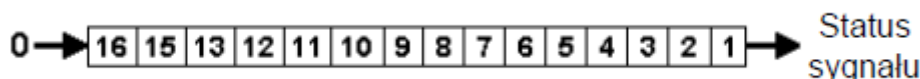
bitów wysuwanych poza rejestr jest bit 0, sygnał nie przepływa przez blok funkcyjny. Jeżeli ostatnim z bitów wysuwanych poza rejestr jest bit 1, sygnał przepływa przez blok funkcyjny.

### Przemieszczenie w lewo (SHL)



*Przemieszczenie w lewo (SHL)*

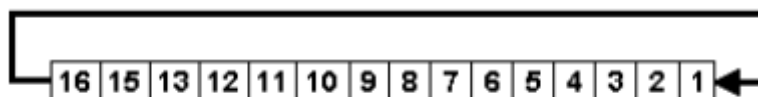
### Przemieszczenie w prawo (SHR)



*Przemieszczenie w prawo (SHR)*

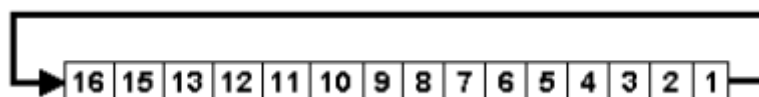
Funkcje Rotate wstawiają wysuwane poza rejestr bity na drugim końcu tego rejestru. Żaden bit nie jest tracony.

### Przemieszczenie w lewo w obiegu zamkniętym (ROL)



*Przemieszczenie w lewo w obiegu zamkniętym (ROL)*

### Przemieszczenie w prawo w obiegu zamkniętym (ROR)



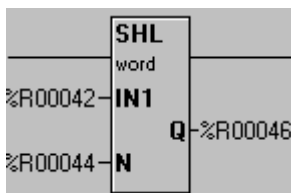
*Przemieszczenie w prawo w obiegu zamkniętym (ROR)*

## Dostępne bloki operacji bitowych

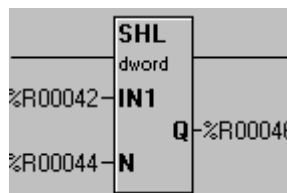
### Blok funkcyjny przemieszczenie w lewo (SHL)

Blok ten realizuje operację logiczną przemieszczenia bitowego w lewo na jednym rejestrze i umieszcza wynik w drugim rejestrze. Wysuwane poza rejestr bity nie są wstawiane w inne miejsce.

Liczba przemieszczeń jest określana za pomocą parametru N. Zarówno IN1 jak i N mogą być rejestrami (%R, %AI, itp.) lub wartościami całkowitymi (8, 23). Parametr Q musi być rejestrem.



Przemieszczenie bitowe w lewo (Word)

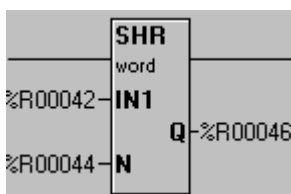


Przemieszczenie bitowe w lewo (Dword)

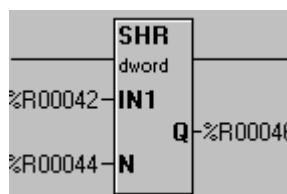
```
%R41 = 0000000100010000 (544)
SHL 8 %R43 = 0001000000000000 (4096)
Przepływ sygnału = ON (TRUE)
```

### Blok funkcyjny przemieszczenie w prawo (SHR)

Blok ten realizuje operację logiczną przemieszczenia bitowego w prawo na jednym rejestrze i umieszcza wynik w drugim rejestrze. Wsuwane poza rejestr bity nie są wstawiane w inne miejsce. Liczba przemieszczeń jest określana za pomocą parametru N. Zarówno IN1 jak i N mogą być rejestrami (%R, %AI, itp.) lub wartościami całkowitymi (8, 23). Parametr Q musi być rejestrem.



Przemieszczenie bitowe w prawo (Word)

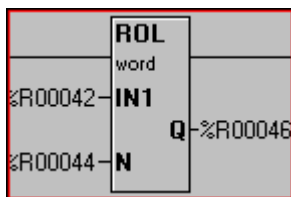


Przemieszczenie bitowe w prawo (Dword)

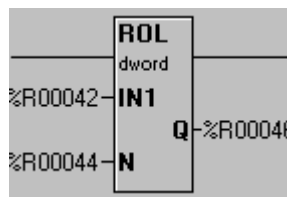
```
%R41 = 1000000000000000 (32768)
SHR 8 %R43 = 0000000010000000 (128)
Przepływ sygnału = OFF (FALSE)
```

### Blok funkcyjny przesunięcie bitowe w lewo w obiegu zamkniętym (ROL)

Blok ten realizuje operację logiczną przesunięcia bitowego w lewo w obiegu zamkniętym na jednym rejestrze i umieszcza wynik w drugim rejestrze. W czasie przemieszczania, bity wysuwane poza rejestr są wstawiane z prawej strony tego rejestru. Liczba przesunięć jest określana za pomocą parametru N. Zarówno IN1 jak i N mogą być rejestrami (%R, %AI, itp.) lub wartościami całkowitymi (8, 23). Parametr Q musi być rejestrem.



*Przesunięcie bitowe w lewo w obiegu zamkniętym (Word)*

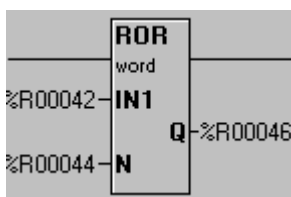


*Przesunięcie bitowe w lewo w obiegu zamkniętym (Dword)*

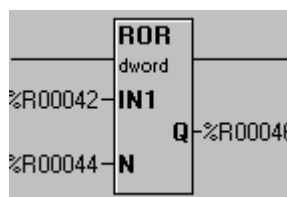
```
%R41 = 1010010100111100 (42300)
ROL 11 %R43 = 1110010100101001 (58665)
```

### Blok funkcyjny przesunięcie bitowe w prawo w obiegu zamkniętym (ROR)

Blok ten realizuje operację logiczną przesunięcia bitowego w prawo w obiegu zamkniętym na jednym rejestrze i umieszcza wynik w drugim rejestrze. W czasie przemieszczania, bity wysuwane poza rejestr są wstawiane z lewej strony tego rejestru. Liczba przesunięć jest określana za pomocą parametru N. Zarówno IN1 jak i N mogą być rejestrami (%R, %AI, itp.) lub wartościami całkowitymi (8, 23). Parametr Q musi być rejestrzem.



*Przesunięcie bitowe w prawo w obiegu zamkniętym (Word)*



*Przesunięcie bitowe w prawo w obiegu zamkniętym (Dword)*

```
%R41 = 1010010100111100 (42300)
ROR 11 %R43 = 1010011110010100 (42900)
```

## Przemieszczanie danych

### Przemieszczanie pojedynczych danych

Bloki funkcyjne do przemieszczania danych pozwalają na przemieszczanie danych pomiędzy rejestrami (np. odczytanie wejścia analogowego [%AI] i zapisanie go do rejestru %R. Bloki te można wykorzystać do wstawiania stałych do rejestrów, przemieszczania bloków danych do innego miejsca lub wypełniania bloków rejestrów tą samą wartością.

Wartości w rejestrze będącym parametrem wejściowym nie ulegają zmianie w przypadku bloków funkcyjnych do przemieszczania bloków (Block Move) lub wypełniania bloków (Block Fill), w

przypadku innych bloków funkcyjnych, wartości w rejestrze będącym parametrem wejściowym są zmieniane.

**a. Sprawdzanie typów**

Bloki te nie sprawdzają zgodności typów. Wartości przemieszczane są na poziomie bitowym bez względu na typ parametru wejściowego lub docelowego. Instrukcja do przemieszczania bloków (Block Move) pozwala na przemieszczanie wartości typu WORD, DWORD lub REAL. Należy zwrócić uwagę, że w przypadku wartości DWORD lub REAL przemieszczane są dwa (2) rejestry.

**b. Przepływ sygnału**

Po zakończeniu wykonywania tego typu instrukcji, sygnał ma zawsze wartość TRUE. Wyjątkiem jest blok funkcyjny Indirect Move. W jego przypadku, przesunięcie traktowane jest jako niepoprawne a sygnał ma wartość FALSE, jeżeli rejestr źródłowy lub docelowy zawiera 0 (zero) lub jeżeli długość przesunięcia przekracza liczbę bloków funkcyjnych dostępnych w sterowniku.

**c. Konfigurowanie**

W celu skonfigurowania bloku funkcyjnego należy na niego podwójnie kliknąć, a następnie wprowadzić adresy rejestrów z parametrami i rejestru wycinkowego.

W przypadku bloków funkcyjnych do przemieszczania rejestrów i wypełniania, jako parametr wejściowy można podać stałą numeryczną. Spowoduje to umieszczenie tej stałej numerycznej w rejestrze wyjściowym. Rejestr Q (wynik) musi być określony poprzez podanie adresu.

W przypadku bloku funkcyjnego do przemieszczania bloków zarówno parametry wejściowe jak i wyjściowy muszą być określone poprzez podanie adresu. Żaden z tych parametrów nie może być stałą numeryczną. Parametr Count określa liczbę przemieszczanych lub wypełnianych rejestrów w czasie pojedynczego wykonania bloku funkcyjnego. W przypadku bloku funkcyjnego do przemieszczania rejestru, parametr Count ma stałą wartość '1'.

W przypadku bloków funkcyjnych do przemieszczania lub wypełniania bloków, parametr COUNT może mieć wartość z przedziału od 0 (zero) do maksymalnej liczby rejestrów danego typu, dostępnych w sterowniku.

Dla rejestrów %R sterowników OCS, maksymalna wartość parametru COUNT wynosi 2048. Dla bloku funkcyjnego do przemieszczania danych typu Word należy wybrać typ danych WORD lub DWORD.

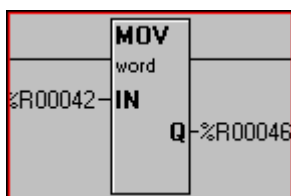
## Dostępne bloki do przemieszczania danych

### Blok funkcyjny przemieszczanie pojedynczego rejestru (Mov)

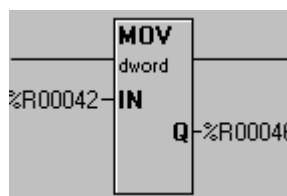
#### Uwaga:

Blok funkcyjny Move\_Word może operować na danych 16-bitowych lub 32-bitowych, w zależności od wyboru użytkownika.

Powoduje on przesunięcie rejestru, pary rejestrów (32 bitów) albo stałej wartości do następnego rejestru lub pary rejestrów (32 bitów).



a. MOVE



b. MOVE DWORD

W czasie konfigurowania, z rozwijalnej listy Type należy wybrać typ WORD (16 bitów) lub DWORD (32 bity).

Przykładowo, jeżeli przed wykonaniem bloku funkcyjnego w rejestrach znajdują się podane poniżej wartości:

%R42	34567
%R43	12
%R44	63
%R45	127
%R46	82

Po zakończeniu wykonywania bloku funkcyjnego typu INT, wartości te będą następujące:

%R42	34567
%R43	12
%R44	63
%R45	127
%R46	34567

Wartość IN może być również stałą numeryczną bez znaku. Przykładowo, jeżeli wartość parametru IN wynosi 1492, po wykonaniu instrukcji w rejestrze %R46 znajdować się będzie wartość 1492.

#### Uwaga:

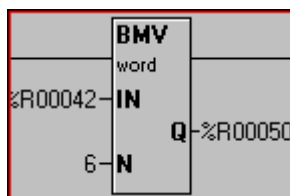
Jeżeli jako parametr IN wprowadzono stałą numeryczną ze znakiem, jest ona w czasie konfigurowania traktowana jako wartość bez znaku. Przykładowo, jeżeli parametr IN ma wartość -1, wykorzystywana jest wartość 65535.

W przypadku bloku funkcyjnego Move\_Dword, operacja jest wykonywana na dwóch kolejnych rejestrach. Przykładowo, jeżeli rejestry %R42 i %R43 będą przechowywać wartość 103582, po wykonaniu instrukcji, w rejestrach %R46 i %R47 będzie przechowywana wartość 103582.

### Blok funkcyjny przemieszczanie bloków rejestrów (BMV)

#### Uwaga:

Instrukcja Block\_Move operuje wyłącznie na rejestrach 16-bitowych. Instrukcja ta przemieszcza blok rejestrów z zadanego miejsca w inne.



*BLOCK MOVE (WORD)*

#### Ostrzeżenie:

Jeżeli parametr IN jest stałą numeryczną ze znakiem, w czasie konfigurowania jest on traktowany jako liczba bez znaku. Przykładowo, parametr IN równy -1 jest traktowany jako liczba 65535.

Przykładowo, jeżeli przed wykonaniem bloku funkcyjnego w rejestrach znajdują się podane poniżej wartości:

%R42	65535	%R50	0
%R43	65535	%R51	0
%R44	65535	%R52	0
%R45	65535	%R53	0
%R46	65535	%R54	0
%R47	65535	%R55	0
%R48	65535	%R56	0
%R49	65535	%R57	0

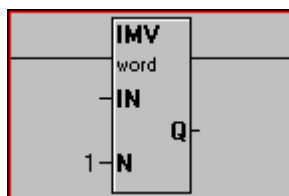
Wówczas po zakończeniu wykonywania, wartości te będą następujące:

%R42	65535	%R50	65535
%R43	65535	%R51	65535
%R44	65535	%R52	65535
%R45	65535	%R53	65535
%R46	65535	%R54	65535
%R47	65535	%R55	65535
%R48	65535	%R56	0
%R49	65535	%R57	0

## Blok funkcyjny Indirect Move (IMV)

### Uwaga:

Instrukcja Indirect\_Move operuje wyłącznie na rejestrach 16-bitowych. Działanie tej instrukcji jest podobne do działania instrukcji Block\_Move. Powoduje ona przemieszczenie danych z jednego miejsca pamięci do innego.



INDIRECT MOVE

Różnica polega na sposobie określania adresu bloku źródłowego (IN), bloku docelowego (Q) lub obydwu tych adresów. W przypadku tej instrukcji możliwe jest bezpośrednie określenie adresu poprzez podanie typu adresu lub pośrednie.

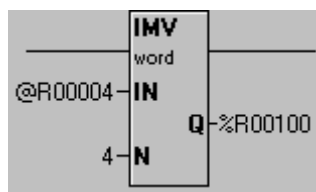
W przypadku formatu pośredniego wprowadzony rejestr zawiera przesunięcie pierwszego z przemieszczanych rejestrów %R. W przypadku adresu docelowego rejestr ten zawiera adres lub przesunięcie pierwszego z rejestrów %R w bloku docelowym.

### Uwaga:

Adresy pośrednie można określać wyłącznie dla rejestrów %R. Sam rejestr źródłowy lub docelowy może być dowolnego typu. W przypadku zaznaczenia opcji INDIRECT, można podać dowolny rejestr.

Adres pośredni lub bezpośredni można określić dla parametru IN, Q lub dla obydwu tych parametrów. Niemniej jednak, nie wolno wprowadzać stałych.

W celu zilustrowania, poniżej pokazano skonfigurowany blok funkcyjny, w którym jako parametr IN podano %A11, jako Q rejestr %R12, zaznaczono obydwie opcje Indirect. Blok ten powoduje przesunięcie czterech (4) słów.



INDIRECT MOVE

Należy zwrócić uwagę na specjalny znak @, informujący że stosowane jest adresowanie pośrednie.

Po doprowadzeniu sygnału do tego bloku funkcyjnego, następuje pobranie wartości z %A11, wykorzystywanej w dalszym ciągu jako wskaźnik do początku bloku w obszarze rejestrów %R. Wartość znajdująca się w rejestrze %R12 jest wykorzystywana jako wskaźnik bloku docelowego w obszarze rejestrów %R. W wyniku działania tego rejestru następuje przesunięcie czterech (4) słów 16-bitowych.

W niniejszym przykładzie założono, że rejestr %A1 zawiera "56", rejestr %R12 "100". Należy zwrócić uwagę, że modyfikowane są rejestry %R, ponieważ zaznaczono opcję Indirect zarówno dla parametru źródłowego jak i docelowego.

Przed rozpoczęciem wykonywania tego bloku funkcyjnego, zawartość rejestrów jest następująca:

%A1	56	%R12	100
%R55	123	%R99	1
%R56	45	%R100	2
%R57	28	%R101	3
%R58	20789	%R102	4
%R59	1	%R103	5
%R60	15	%R104	6

Po jego wykonaniu, wartości zapisane w rejestrach są następujące:

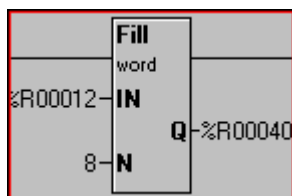
%A1	56	%R12	100
%R55	123	%R99	1
%R56	45	%R100	45
%R57	28	%R101	28
%R58	20789	%R102	20789
%R59	1	%R103	1
%R60	15	%R104	15

### Blok funkcyjny wypełnianie bloków (FILL)

#### Uwaga:

Blok funkcyjny Fill operuje wyłącznie na rejestrach 16-bitowych.

Blok ten powoduje zapisanie do bloku rejestrów określonej wartości. Parametr IN może być stałą numeryczną lub adresem rejestru.



Wypełnianie bloków



**Ostrzeżenie:**

Jeżeli parametr IN jest stałą numeryczną ze znakiem, w czasie konfigurowania jest on traktowany jako liczba bez znaku. Przykładowo, parametr IN równy -1 jest traktowany jako liczba 65535.

Przykładowo, jeżeli przed wykonaniem bloku funkcyjnego w rejestrach znajdowały się następujące wartości:

%R12	1234
%R40	3221
%R41	4632
%R42	252
%R43	584
%R44	9874
%R45	5212
%R46	32
%R47	14
%R48	216
%R49	8

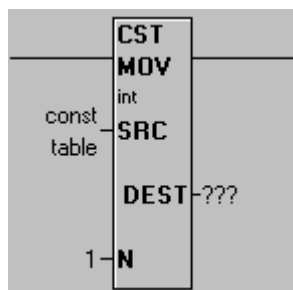
po zakończeniu wykonywania, wartości te będą następujące:

%R12	1234
%R40	1234
%R41	1234
%R42	1234
%R43	1234
%R44	1234
%R45	1234
%R46	1234
%R47	1234
%R48	216
%R49	8

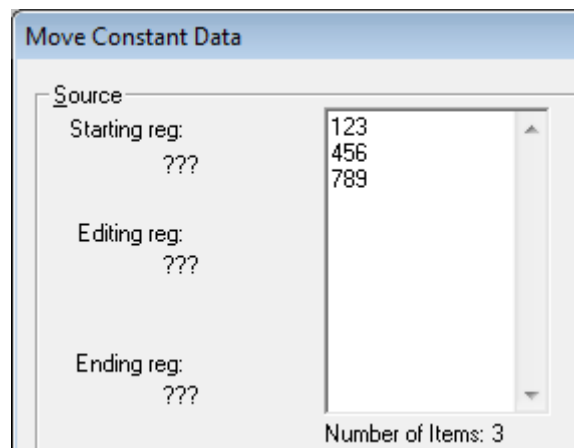
**Blok funkcyjny przemieszczanie bloków stałych (CST MOV)**

Blok funkcyjny Move Constant Data pozwala na zapisanie tabeli ze stałymi do grupy sąsiadujących rejestrów sterownika. Tabela stałych może zawierać wartości typu INT, UDINT, DINT, UDINT lub REAL. Wszystkie elementy tablicy muszą być tego samego typu.

Założmy, że istnieje następująca tabela z wartościami INT:



*Blok funkcyjny CONST DATA*



*Okno do edycji*

Pomocnicze informacje wyświetlane podczas wprowadzania tabeli:

- **Starting reg** – rejestr początku tabeli
- **Editing reg** – numer edytowanego elementu w tablicy
- **Ending reg** – rejestr końca tabeli

**Uwaga:**

Stałe można kopiować i wklejać pomiędzy aplikacjami Windows, włączając w to arkusz Microsoft Excel i edytor Microsoft Word.

**Uwaga:**

Dla liczb rzeczywistych mniejszych od 1 należy wpisać przed kropką zero (np. .999 - wartość niepoprawna, 0.999 - wartość poprawna).

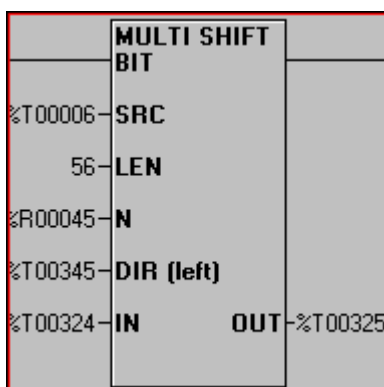
po zakończeniu wykonywania, wartości te będą następujące:

%R100	123
%R101	456
%R102	789

## Dostępne bloki funkcyjne do przemieszczania tablic

### Blok funkcyjny Multi Shift

Funkcja to pozwala na przesunięcie w lewo lub prawo o określoną liczbę razy tablicy z wartościami typu BITS, BYTES, WORD lub DWORDS.



*Przemieszczanie tablic z wartościami(BIT)*

#### a. Przepływ sygnału

Po doprowadzeniu sygnału do tego bloku funkcyjnego, w każdym cyklu następuje przemieszczenie danych, zgodnie z określonymi parametrami. Po załączeniu bloku funkcyjnego zawsze jest przesyłany sygnał wyjściowy.

#### b. Terminologia stosowania w funkcjach do przemieszczania tablic

**SRC** - Początek tablicy z danymi typu BIT, BYTE, WORD lub DWORD do przemieszczania.

Przesunięte dane są zapamiętywane w tablicy, rozpoczynającej się do tego miejsca. Tablica z danymi BIT może zaczynać się od dowolnego rejestru (%I1, %I6, %R1.1, %R4.7, ...). Tablice z danymi typu BYTE, WORD i DWORD muszą uwzględniać podział na słowa (%I1, %I17, %I33, %R1, %R2, ...).

**LEN** - Liczba bajtów (długość tablicy) w tablicy z danymi typu BITS, BYTES WORDS lub DWORDS. Musi to być stała z zakresu 1 do 32767.

**N** - Liczba elementów w tablicy do przesunięcia. Parametr ten może być stałą lub zmienną WORD.

**DIR** - Kierunek przemieszczania. Jeżeli parametr ten ma wartość 1, dane są przemieszczane w lewo. Jeżeli parametr ten ma wartość 0, dane są przemieszczane w prawo.

Przykłady:

- Przesunięcie wartości typu BIT w lewo o 1: %T2 <- %T1 %T3 <- %T2...
- Przesunięcie wartości typu BYTE w lewo o 1: %R1(starszy bajt) <- %R1(młodszy bajt) %R2(młodszy bajt) <- %R1(starszy bajt)...
- Przesunięcie wartości typu WORD w lewo o 1: %R2 <- R1 %R3 <- %R2...

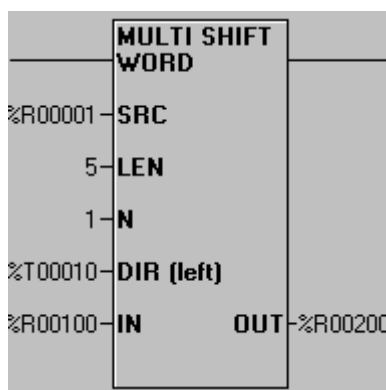
- Przesunięcie wartości typu DWORD w lewo o 1: %R3, %R4 <- %R1, %R2 %R5, %R6 <- %R3, %R4
- Przesunięcie wartości typu BIT w prawo o 1: %T2 -> %T1 %T3 -> %T2...
- Przesunięcie wartości typu BYTE w prawo o 1: %R1(starszy bajt) -> %R1(młodszy bajt) %R2(młodszy bajt) -> %R1(starszy bajt)...
- Przesunięcie wartości typu WORD w prawo o 1: %R2 -> %R1 %R3 -> %R2...
- Przesunięcie wartości typu DWORD w prawo o 1: %R3, %R4 -> %R1, %R2 %R5, %R6 -> %R3, %R4

**IN** - Początek wartości typu BIT, BYTE, WORD lub DWORD wstawiane do tablicy.

**OUT**- Ostatnia z usuniętych z tablicy wartości typu BIT, BYTE, WORD lub DWORD.

### Przykłady instrukcji do przemieszczania tabel z wartościami typu Word

Pokazany poniżej blok funkcyjny wykorzystywany jest w zamieszczonych, w dalszej części przykładach



*MULTI SHIFT WORD*

Przykład 1: Na początku rejestry mają następujące wartości:

%R1	1
%R2	2
%R3	3
%R4	4
%R5	5
%T10	TRUE
%R100	123
%R200	0

Po pierwszym cyklu, wartości są następujące:

%R1	123
%R2	1

%R3	2
%R4	3
%R5	4
%T10	TRUE
%R100	123
%R200	5

Po drugim cyklu:

%R1	123
%R2	123
%R3	1
%R4	2
%R5	3
%T10	TRUE
%R100	123
%R200	4

Należy zwrócić uwagę na przemieszczanie danych w tabeli WORD (%R1 do %R5) i ostatecznie na wyjście.

Przykład 2:

Na początku rejestry mają następujące wartości:

%R1	1
%R2	2
%R3	3
%R4	4
%R5	5
%T10	TRUE
%R100	123
%R200	0

Po pierwszym cyklu przeniesiono też parametr wejściowy (%R100) na 456.

%R1	123
%R2	1
%R3	2
%R4	3
%R5	4
%T10	TRUE
%R100	456
%R200	5

Po drugim cyklu:

%R1	456
%R2	123
%R3	1
%R4	2
%R5	3
%T10	TRUE
%R100	456
%R200	4

Przykład 3:

Na początku rejestry mają następujące wartości (należy zwrócić uwagę, że zmieniono parametr DIRECTION, a więc wartości będą przemieszczane w prawo):

%R1	1
%R2	2
%R3	3
%R4	4
%R5	5
%T10	FALSE
%R100	123
%R200	0

Po pierwszym cyklu, wartości są następujące:

%R1	2
%R2	3
%R3	4
%R4	5
%R5	123
%T10	FALSE
%R100	123
%R200	1

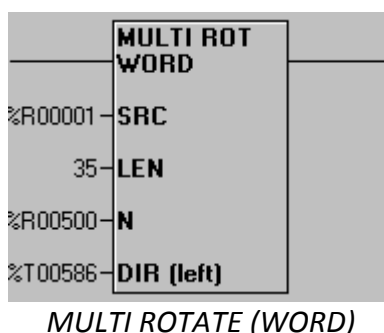
Po drugim cyklu:

%R1	3
%R2	4
%R3	5
%R4	123
%R5	123
%T10	FALSE
%R100	123
%R200	2

Należy zwrócić uwagę na przemieszczania danych w tabeli WORDS (%R5 do %R1) i ostatecznie na wyjście.

### Blok funkcyjny Multi Rotate

Przemieszczanie tablic danych w obiegu zamkniętym. Funkcja to pozwala na przesunięcie w lewo lub prawo w obiegu zamkniętym wartości typu BITS, BYTES, WORD lub DWORDS o określoną liczbę razy.



#### a. Przepływ sygnału

Po doprowadzeniu sygnału do tego bloku funkcyjnego, w każdym cyklu pracy jednostki centralnej następuje przemieszczenie danych w obiegu zamkniętym, zgodnie z określonymi parametrami. Funkcja ta nie reaguje na zmianę wartości sygnału. Sygnał wyjściowy jest zawsze przesyłany.

#### b. Terminologia stosowana w przemieszczaniu tabel danych w obiegu zamkniętym

**SRC** - Początek tablicy z danymi typu BIT, BYTE, WORD lub DWORD do przemieszczania w obiegu zamkniętym. Przemieszczane w obiegu zamkniętym dane są zapamiętywane w tablicy, rozpoczynającej się od tego miejsca. Tablica z danymi BIT może zaczynać się od dowolnego rejestru (%I1, %I6, %R1.1, %R4.7, ...). Tablice z danymi typu BYTE, WORD i DWORD muszą uwzględniać podział na słowa (%I1, %I17, %I33, %R1, %R2, ...).

**LEN** - Liczba bajtów w tablicy z danymi typu BITS, BYTES WORDS lub DWORDS. Musi to być stała z zakresu 1 do 32767.

**N** - Liczba elementów w tablicy do przesunięcia w obiegu zamkniętym. Parametr ten może być stałą lub zmienną WORD.

**DIR** - Kierunek przemieszczania w obiegu zamkniętym. Jeżeli parametr ten ma wartość 1, dane są przemieszczane w obiegu zamkniętym w lewo. Jeżeli parametr ten ma wartość 0, dane są przemieszczane w obiegu zamkniętym w prawo.

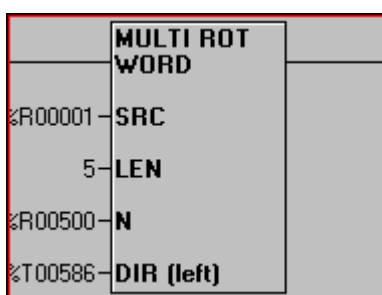
Przykłady:

- Przesunięcie wartości typu BIT w lewo o 1: %T2 <- %T1 %T3 <- %T2...
- Przesunięcie wartości typu BYTE w lewo o 1: %R1(starszy bajt) <- %R1(młodszy bajt) %R2(młodszy bajt) <- %R1(starszy bajt)...
- Przesunięcie wartości typu WORD w lewo o 1: %R2 <- %R1 %R3 <- %R2...
- Przesunięcie wartości typu DWORD w lewo o 1: %R3, %R4 <- %R1, %R2 %R5, %R6 <- %R3, %R4
- Przesunięcie wartości typu BIT w prawo o 1: %T2 -> %T1 %T3 -> %T2...

- Przesunięcie wartości typu BYTE w prawo o 1: %R1(starszy bajt) -> %R1(młodszy bajt) %R2(młodszy bajt) -> %R1(starszy bajt)...
- Przesunięcie wartości typu WORD w prawo o 1: %R2 -> %R1 %R3 -> %R2...
- Przesunięcie wartości typu DWORD w prawo o 1: %R3, %R4 -> %R1, %R2 %R5, %R6 -> %R3, %R4

**Przykłady instrukcji do przemieszczania tabel w obiegu zamkniętym z wartościami typu Word**

Pokazany poniżej blok funkcyjny wykorzystywany jest w zamieszczonych w dalszej części przykładach.



*MULTI ROTATE WORD*

**Przykład 1**

Na początku rejestry mają następujące wartości:

%R1	1
%R2	2
%R3	3
%R4	4
%R5	5
%T586	TRUE
%R500	1

Po pierwszym cyklu, wartości są następujące:

%R1	5
%R2	1
%R3	2
%R4	3
%R5	4
%T586	TRUE
%R500	1



Po drugim cyklu:

%R1	4
%R2	5
%R3	1
%R4	2
%R5	3
%T586	TRUE
%R500	1

Przykład 2

Na początku rejestry mają następujące wartości (należy zwrócić uwagę, że zmieniono parametr DIRECTION, a więc wartości będą przemieszczane w prawo):

%R1	1
%R2	2
%R3	3
%R4	4
%R5	5
%T586	FALSE
%R500	1

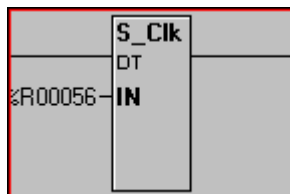
Po pierwszym cyklu, wartości są następujące:

%R1	2
%R2	3
%R3	4
%R4	5
%R5	1
%T586	FALSE
%R500	1

Po drugim cyklu:

%R1	3
%R2	4
%R3	5
%R4	1
%R5	2
%T586	FALSE
%R500	1

## Blok funkcyjny ustawiania zegara czasu rzeczywistego (S\_Clk)



Blok funkcyjny S\_Clk pozwala na ustawienie z poziomu programu sterującego zegara czasu rzeczywistego. Umożliwia to synchronizację różnych urządzeń pracujących w sieci lub zmianę czasu w oparciu o zaimplementowany algorytm.

Parametry wejściowe tego bloku funkcyjnego są zapisane w sześciu kolejnych rejestrach. Format tych rejestrów jest następujący:

- Rejestr 1 Nowa wartość sekund
- Rejestr 2 Nowa wartość minut
- Rejestr 3 Nowa godzina (w formacie 24-godzinnym)
- Rejestr 4 Nowa data
- Rejestr 5 Nowy miesiąc (1 = styczeń)
- Rejestr 6 Nowy rok (wyrażony za pomocą 4 cyfr)

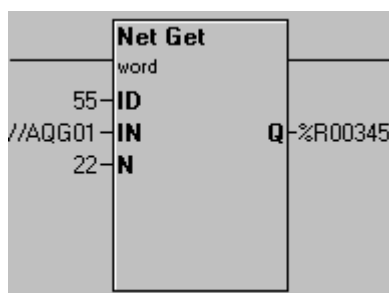
Blok ten przesyła sygnał pod warunkiem, że rejestry określające nowy czas i datę mają poprawne wartości. Błędne wartości to na przykład wpisanie '50' do rejestru określającego godzinę, czy '100' do rejestru określającego miesiąc.

Dzień tygodnia jest automatycznie obliczany i uaktualniany w zegarze czasu rzeczywistego (%RSR50).

## Bloki funkcyjne do komunikacji sieciowej

### Dostępne bloki do komunikacji sieciowe

#### Blok funkcyjny odczyt wartości typu Word z sieci (NetGet)



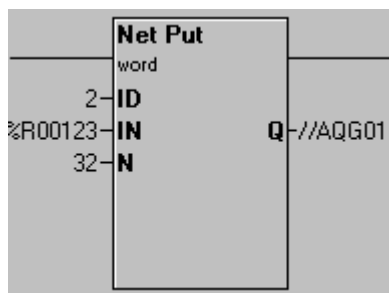
*Network Get Words*

Blok ten pozwala na odczyt danych globalnych z dowolnego urządzenia przyłączonego do sieci i skopiowanie ich do rejestrów. Jeżeli urządzenie określone za pomocą parametru ID nie wysła danych, blok funkcyjny nie przesyła sygnału i wysyła żądanie wysłania danych. Po otrzymaniu żądanych danych, następuje przesłanie sygnału przez ten blok funkcyjny.

Z funkcji tej można korzystać w sieciach CsCAN i DeviceNet.

- **ID** - Rejestr lub stała definiująca źródło danych globalnych. W przypadku nieprawidłowej wartości tego parametru, funkcja nie realizuje żadnych działań i nie przesyła sygnału.
- **IN** - Parametr ten definiuje adres początkowy żądanych danych globalnych. Może to być rejestr %AG lub %QG. Należy zwrócić uwagę, że rejestry %QG muszą uwzględniać podział na słowa (1, 17, 33, itp.). Jest to rejestr sieciowy, przypisany do rejestru i utworzony poprzez przesłanie parametru ID.
- **N** - Liczba słów do odczytu z urządzenia wskazanego za pomocą parametru ID. Dopuszczalny przedział wartości to 1 do 32.
- **Q** - Adres początkowy, określający miejsce zapisywania odczytanych danych. Jest to rejestr sterownika.

#### Blok funkcyjny zapis wartości typu Word do urządzenia sieciowego (NetPut)



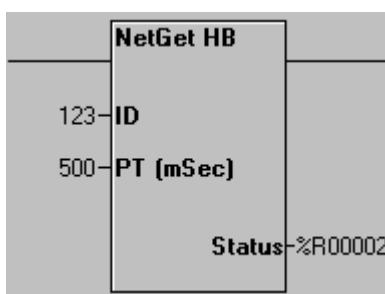
*Network Put Words*

Blok ten pozwala na wysłanie danych globalnych do określonego urządzenia sieciowego. Funkcja ta nie reaguje na zmianę poziomu sygnału, za każdym razem po doprowadzeniu sygnału kopiuje ona dane z rejestrów źródłowych i próbuje je przesłać. Z funkcji tej można korzystać w sieciach CSCAN.

Funkcja przesyła sygnał pod warunkiem podania poprawnej wartości parametru ID, mieszczącego się w przedziale określonym przez identyfikator sieciowy oraz liczbę identyfikatorów ID przypisanych do danego węzła.

- **ID** - Rejestr lub stała podająca identyfikator wykorzystywany do wysyłania danych w sieci. Parametr ten musi mieścić się w przedziale określonym przez główny identyfikator sieciowy oraz sumaryczną liczbę identyfikatorów dla danego węzła.
- **IN** - Pierwszy z rejestrów wysyłanych poprzez sieć. Jest to lokalny rejestr sterownika.
- **N** - Liczba słów wysyłanych siecią.
- **Q** - Parametr ten definiuje rejestr początkowy, określający miejsce zapisywania przesyłanych danych. Należy zwrócić uwagę, że rejestry %QG muszą uwzględniać podział na słowa (1, 17, 33, itp.). Jest to rejestr sieciowy, przypisany do identyfikatora sieciowego ID.

### Blok funkcyjny sprawdzanie pracy urządzenia sieciowego (NetGet HB)



*Network Get Heart Beat*

Funkcja ta pozwala na sprawdzenie z poziomu jednego urządzenia, czy inne urządzenie pracuje, nie generuje ona jednak żadnego ruchu w sieci. Można z niej korzystać wyłącznie w sieciach CSCAN.

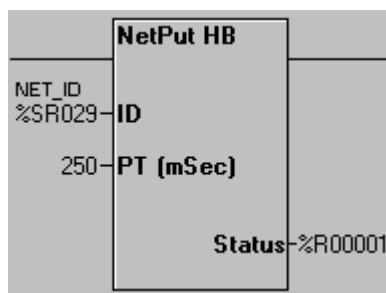
Funkcja ta nie przesyła sygnału w przypadku nieprawidłowej wartości parametru ID lub jeżeli monitorowane urządzenie nie wysyła w czasie określonym za pomocą parametru PT komunikatu informującego o jego pracy.

- **ID** - Rejestr lub stała podająca identyfikator monitorowanego urządzenia.
- **PT** - Maksymalny czas oczekiwania na komunikat informujący o pracy monitorowanego urządzenia. Czas ten powinien być większy od odwrotności częstotliwości wysyłania przez monitorowane urządzenie komunikatów informujących o pracy. (W zależności od natężenia ruchu w sieci oraz czasu trwania cyklu pracy jednostki centralnej, czas „timeout” funkcji GET powinien być większy 10 do 1000 milisekund od czasu „timeout” funkcji PUT).

Należy wybrać wartość z przedziału 1 do 6553 milisekund. Rozdział 2. Bloki funkcyjne języka drabinkowego

- **Status** - Rejestr ten jest obecnie wykorzystywany do celów wewnętrznych. Nie wolno zapisywać do niego żadnych wartości.

### Blok funkcyjny wysyłanie komunikatu informującego o pracy urządzenia sieciowego (NetPutW)



*Network Put Heart Beat*

Dzięki tej funkcji urządzenie może wysłać komunikat, informujący pozostałe włączone urządzenia, że jest ono włączone i normalnie pracuje. Funkcja ta generuje ruch w sieci. Tworzenie takiego komunikatu zwykle nie powoduje zmniejszenia szerokości pasma przenoszenia pod warunkiem, że komunikaty takie nie będą generowane bardzo często i przez dużą liczbę urządzeń.

Funkcja ta przesyła sygnał pod warunkiem, że podana zostanie poprawna wartość parametru ID. Można z niej korzystać wyłącznie w sieciach CscAN.

- **ID** - Rejestr lub stała będąca głównym identyfikatorem sieciowym (%SR29). Parametr ten musi mieścić się w przedziale wyznaczonym przez główny identyfikator sieciowy oraz liczbę identyfikatorów przypisanych do urządzenia.
- **PT** - Częstotliwość wysyłania komunikatów, w milisekundach. Dopuszczalny przedział wartości to 1 do 6553.
- **Status** - Rejestr ten jest obecnie wykorzystywany do celów wewnętrznych. Nie wolno zapisywać do niego żadnych wartości.

## Operacje na łańcuchach znaków

---

### Wprowadzenie

Łańcuchy znaków w oprogramowaniu Cscape muszą być ujęte w apostrofy ( ' ). Mogą one mieć zerową długość. Poniżej podano przykłady łańcuchów znaków, poprawnych z punktu widzenia oprogramowania Cscape. Należy zwrócić uwagę, że są one ujęte w apostrofy.

'Hot' Długość = 3

' ' Długość = 0

' ' jedna SPACJA Długość = 1

Elementem łańcucha nie musi być znak ASCII, może to być dowolna, 8-bitowa wartość. Niemniej jednak należy pamiętać, że wyświetlacz posiada ograniczone możliwości w pokazywaniu znaków innych niż ASCII.

### Znaki specjalne

Ponieważ znak apostrofu jest wykorzystywany do zaznaczenia początku i końca łańcucha znaków, nie można z niego bezpośrednio korzystać. W celu wstawienia do łańcucha znaków znaku apostrofu należy podać dwa znaki. Pierwszy z nich to znak dolara '\$'. Metoda ta pozwala na wprowadzenie szeregu innych, użytecznych znaków.

Kombinacja znaków	Interpretacja przy wydruku
\$\$	Znak Dolara
'	Pojedynczy apostrof
\$L lub \$l	Przejdźcie do nowej linii
\$N lub \$n	Nowe linia
\$P lub \$p	Przejdźcie do nowej strony
\$R lub \$r	Powrót karetki
\$T lub \$t	Tabulator

### Liczby heksadecymalne

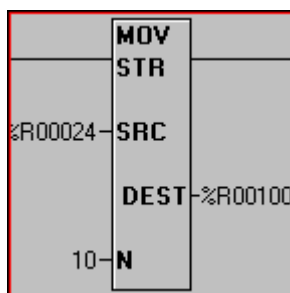
Można również podawać liczby heksadecymalne. W celu wpisania liczby heksadecymalnej należy wprowadzić znak dolara a następnie dwa znaki heksadecymalne.

Kombinacja znaków	Interpretacja przy wyświetlaniu
\$0D	Powrót karetki (działanie identyczne jak \$R)
\$0D\$0A	Sekwencja Powrót karetki/Przejsie do nowej linii (działanie identyczne jak \$R\$L)
\$00	Znak pusty
\$FF	Wartość binarna 255

Numer heksadecymalny musi składać się z dokładnie dwóch (2) znaków. Dopuszczalne znaki to '0' - '9', 'A' - 'F' i 'a' - 'f'. Nie jest rozróżniana wielkość liter.

Jeżeli numer można przedstawić za pomocą jednego znaku heksadecymalnego (np. '\$a'), łańcuch znaków musi zawierać na początku dodatkowe 0 (np. '\$0a').

### Blok funkcyjny przemieszczanie znaków w łańcuchu tekstowym (MOV STR)



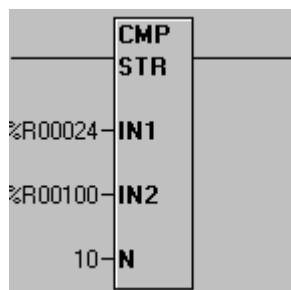
*Przemieszczanie znaków w łańcuchu tekstowym*

Po doprowadzeniu sygnału do bloku funkcyjnego, określona liczba znaków jest przemieszczana z miejsca określonego parametrem SRC do miejsca określonego parametrem DEST.

- Parametr **STR** może być stałą lub rejestrem. Stałe tekstowe muszą być ujęte w apostrofy ( ' ), np. 'To jest test'.
- Parametr **DEST** może być stałą lub zmienną.
- **N** określa liczbę znaków do przesunięcia i musi to być wartość dziesiętna.

Jeżeli parametr SRC jest stałą tekstową (tzn. pierwszy znak to apostrof), następuje dezaktywacja pola N, wartość tego pola jest równa liczbie znaków. Należy zwrócić uwagę, że sekwencja w formacie heksadecymalnym ('\$0A') umieszczana na początku stałej tekstowej jest traktowana jako jeden znak.

### Blok funkcyjny porównywanie łańcuchów znaków (CMP STR)



*Porównanie ciągu znaków*

Po doprowadzeniu sygnału do bloku funkcyjnego, następuje porównanie łańcuchów znaków zadanych za pomocą parametrów IN1 i IN2. Jeżeli wynik porównania ma wartość TRUE (obydwa łańcuchy znaków są identyczne) następuje przesłanie sygnału na wyjście.

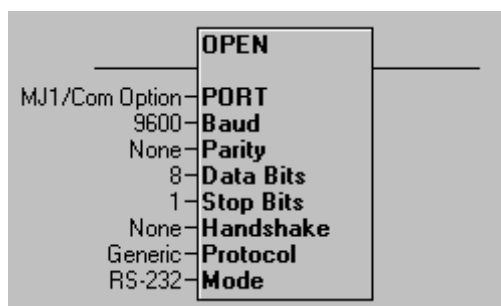
Jako parametr IN1 można podać stałą lub zmienną. Stałe tekstowe muszą być ujęte w apostrofy ( ' ), np. 'To jest test'. Jako parametr IN2 można podać stałą lub zmienną.

N określa liczbę porównywanych znaków i musi to być wartość dziesiętna. Jeżeli parametr IN1 jest stałą tekstową (tzn. pierwszy znak to apostrof), następuje dezaktywacja pola N, wartość tego pola jest równa liczbie wpisanych znaków. Należy zwrócić uwagę, że sekwencja w formacie heksadecymalnym ('\$0A') umieszczana na początku stałej tekstowej jest traktowana jako jeden znak.

## Komunikacja przez port szeregowy

### Dostępne bloki funkcyjne do komunikacji przez port szeregowy

#### Blok funkcyjny ptwarcie portu komunikacyjnego (OPEN)



*Otwarcie portu szeregowego*



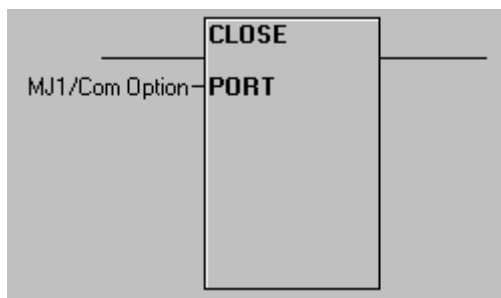
Blok OPEN tworzy kanał do komunikacji z podanym portem. Umożliwia również zdefiniowanie parametrów konfiguracyjnych tego portu (prędkość przesyłania danych, itp.). Kanał pozostaje otwarty do momentu jego zamknięcia za pomocą bloku funkcyjnego Close lub wyłączenia trybu pracy RUN w sterowniku.

Okno dialogowe do konfiguracji zawiera szereg parametrów. Należy wybrać z rozwijalnych list odpowiednie wartości.

Sygnał przepływa przez ten blok funkcyjny, jeżeli zostanie on poprawnie wykonany lub jeżeli port jest już otwarty. W przypadku podjęcia próby otwarcia nie istniejącego portu, sygnał nie przepływa przez blok funkcyjny.

Jeżeli port został otwarty przy pomocy omawianego bloku funkcyjnego, nie może być wykorzystywany do programowania w protokole CsCAN. Sterowniki Horner APG wyposażone w jeden port komunikacyjny można nadal programować z innego urządzenia za pomocą funkcji Pass Through Connection.

## Blok funkcyjny zamknięcie portu komunikacyjnego (CLOSE)



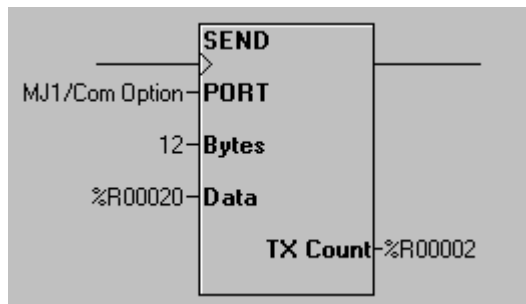
*Zamknięcie portu szeregowego (CLOSE)*

Blok ten zamyka kanał do komunikacji z danym portem. Nie wymaga on określenia jakichkolwiek parametrów, za wyjątkiem numeru portu (PORT). Parametr ten musi być stałą dziesiętną.

Sygnał przepływa przez ten blok funkcyjny, jeżeli zostanie on poprawnie wykonany lub jeżeli port został już wcześniej zamknięty. W przypadku podjęcia próby zamknięcia nie istniejącego portu sygnał nie przepływa przez blok funkcyjny.

Jeżeli wybrany port był poprzednio wykorzystywany jako port do programowania CsCAN, można nadal korzystać z tej funkcji.

### Blok funkcyjny wysyłanie danych poprzez port (SEND)



*Wysyłanie danych poprzez port szeregowy*

Jeżeli port został już otwarty, funkcja ta pozwala na wysłanie określonej liczby bajtów ze sterownika (za pośrednictwem wewnętrznego bufora wysyłania wybranego portu komunikacyjnego).

- Parametr **PORT** to numer uprzednio otwartego portu komunikacyjnego.
- Parametr **BYTES** może być stałą lub zmienną. Podaje on liczbę wysyłanych bajtów.
- Parametr **DATA** zawiera adres początkowy wysyłanych danych. Musi on być określony poprzez podanie adresu rejestru.
- Parametr **TX COUNT** zawiera rzeczywistą liczbę bajtów przesłanych do wewnętrznego bufora portu (lub -1 w przypadku nieprawidłowego wykonania funkcji). Musi on być określony poprzez podanie adresu rejestru.

Po doprowadzeniu sygnału do tego bloku funkcyjnego, rejestr TX Count zawiera liczbę faktycznie wysłanych danych do bufora portu komunikacyjnego. W przypadku nie doprowadzenia sygnału do tego bloku funkcyjnego, rejestr ten zawiera wartość -1 (minus jeden).

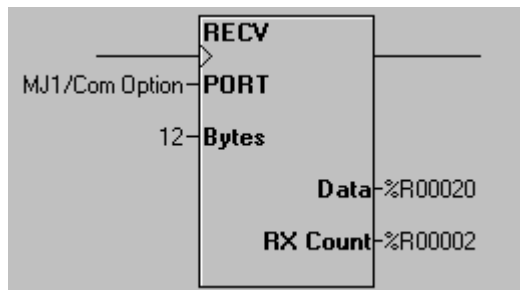
Blok ten nie przesyła sygnału do momentu wysłania do wewnętrznego bufora wysyłania portu komunikacyjnego odpowiedniej liczby znaków (następuje wtedy przesłanie sygnału). Możliwe jest wysłanie wszystkich danych w ciągu jednego cyklu wykonywania programu sterującego.

Jeżeli port nie został otwarty, blok ten nie realizuje żadnych działań i nie przepływa przez niego sygnał.

Jeżeli wartość parametru BYTES jest większa od 255, blok funkcyjny nie realizuje żadnych działań i nie przepływa przez niego sygnał.

Parametr Bytes może być stałą dziesiętną lub zmienną. Maksymalna, dopuszczalna wartość wynosi 255. W przypadku gdy parametr ten jest zmienną i w rejestrze zapisana jest wartość mniejsza od 0 (zero) lub większa od 255, blok funkcyjny nie realizuje żadnych działań i nie przepływa przez niego sygnał.

### Blok funkcyjny odbieranie danych z portu (RECV)



*Odbieranie danych z portu szeregowego*

Jeżeli port został otwarty, funkcja ta pozwala na odebranie określonej liczby bajtów z określonego portu komunikacyjnego.

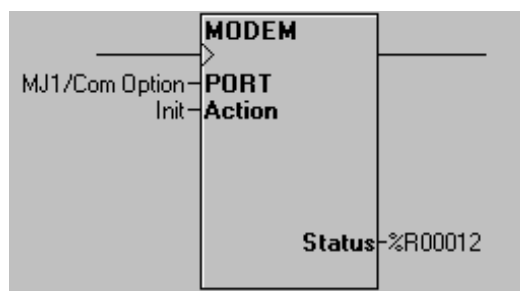
- Parametr **PORT** to numer uprzednio otwartego portu komunikacyjnego.
- Parametr **BYTES** może być zmienną lub stałą dziesiętną. Podaje on maksymalną liczbę znaków do odebrania.
- Parametr **DATA** określa miejsce zapisywania odebranych danych. Musi on być określony poprzez podanie adresu rejestru.
- Parametr **RX COUNT** zawiera rzeczywistą liczbę bajtów skopiowanych z bufora portu do rejestrów określonych za pomocą parametru DATA (lub -1 w przypadku nieprawidłowego wykonania funkcji).

Jeżeli port nie został otwarty, blok funkcyjny nie realizuje żadnych działań i nie przepływa przez niego sygnał.

Blok ten nie przesyła sygnału do momentu odebrania z bufora portu komunikacyjnego odpowiedniej liczby znaków (następuje wtedy przesłanie sygnału). Blok ten może nie być w stanie przesać wszystkich danych w ciągu jednego cyklu wykonywania programu sterującego, zwłaszcza przy mniejszych prędkościach transmisji.

Parametr BYTES może być zmienną. Maksymalna, dopuszczalna wartość wynosi 255. W przypadku gdy parametr jest zmienną i w rejestrze zapisana jest wartość mniejsza od 0 (zero) lub większa od 255, blok ten nie realizuje żadnych działań i nie przepływa przez niego sygnał.

### Blok funkcyjny sterowanie modemem (MODEM)



*Sterowanie modemem*

Blok ten pozwala sterownikowi OCS kontrolować podłączony modem. Parametr PORT określa numer portu, do którego podłączony jest modem.

Parametr Status to rejestr typu WORD (16-bitowy) określany poprzez podanie adresu. Zawiera on wynik wykonywania bloku funkcyjnego.

Parametr ten może w trakcie pracy przyjmować następujące wartości:

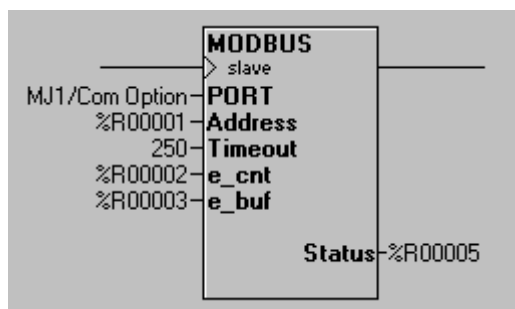
Wartość	Status
0	Pomyślne zrealizowanie operacji
1	Pomyślnie nawiązanie połączenia z modemem
2	Modem wykrył sygnał dzwonienia połączenia przychodzącego
3	Modem zgubił sygnał (nośną) lub podczas próby dzwonienia nie wykrył nośnej
4	Modem zasygnalizował wystąpienie błędu (należy sprawdzić wybieranie numeru, lub łańcuch znaków do inicjalizacji)
-1	Funkcja nie aktywa
-2	Wykonanie funkcji zostało rozpoczęte ale nie zakończone
-3	Przekroczenie czasu (modem nie odpowiada)

Parametr ACTION określa działanie, które ma być wykonane. Z rozwijalnej listy można wybrać jedno z działań:

- **INITIALIZE** - Wysłanie do modemu określonego łańcucha znaków do inicjalizacji.
- **AUTO DIAL** - Wymuszenie wybrania przez modem numeru telefonicznego określoną metodą (tonowo lub za pomocą impulsów).
- **AUTO ANSWER ON** - Włączenie funkcji modemu do automatycznego odpowiadania (jeżeli modem dysponuje taką funkcją) i ustawienie liczby sygnałów dźwiękowych.

Po nawiązaniu połączenia poprzez modem za pomocą AUTO DIAL lub AUTO ANSWER, można rozpocząć wymianę danych za pomocą funkcji Send, Receive, Modbus slave i Modbus master. Jeżeli nawiązano połączenie poprzez modem, a wymagane jest nawiązanie komunikacji poprzez sieć CsCAN, można zamknąć port szeregowy i przełączyć się ponownie do trybu CsCAN.

### Blok funkcyjny MODBUS SLAVE



*Modbus Slave*

- Parametr **PORT** określa port komunikacyjny otwarty wcześniej z programu sterującego, dla którego parametr Protocol został ustawiony na Modbus ASCII lub Modbus RTU.
- Parametr **ADDRESS** może być zmienną lub stałą dziesiętną z przedziału 1 do 247. Określa on adres Modbus wykorzystywany przez sterownik do wysyłania odpowiedzi na żądanie nadesłane przez Modbus.
- Parametr **Timeout** może być zmienną lub stałą dziesiętną z przedziału 0 do 1023. Określa on okres czasu od momentu wysyłania żądania przez master, do momentu ustawienia bitu przekroczenia czasu oczekiwania w słowie statusu. Wartość tego parametru podawana jest jako wielokrotność czasu 100 milisekund (tzn. 100 jednostek = 10.0 sekund).
- Parametr **e\_cnt** (wymagany wyłącznie w przypadku włączenia obsługi komunikatów Exception Message) jest określany poprzez podanie adresu rejestru. Zawiera on liczbę bajtów do wysłania w buforze Message Data. Przejście z wartości zerowej do niezerowej powoduje rozpoczęcie transmisji komunikatu Exception Message.
- Parametr **e\_buf** (wymagany wyłącznie w przypadku włączenia obsługi komunikatów Exception Message) jest określany poprzez podanie adresu rejestru. Jest to numer pierwszego rejestru tablicy, zawierającej komunikat Exception Message (pierwszy bajt komunikatu jest zawarty w młodszej bajcie podanego rejestru).
- Parametr **STATUS** to rejestr typu WORD (16-bitowy) określany poprzez podanie typu i przesunięcia. Zawiera on wynik wykonywania bloku funkcyjnego.

Interpretacja bitów w słowie statusu:

Numer bitu	Status
1	Przekroczenie czasu pozostawiania w bezczynności (timeout)
4	Otrzymanie poprawnego komunikatu (informacje typu „toggle”)
5	Błąd parzystości (pojedyncze ustawienie bitu)
6	Błąd ramki (pojedyncze ustawienie bitu)
7	Przepełnienie (pojedyncze ustawienie bitu)
8	Błąd sumy kontrolnej (pojedyncze ustawienie bitu)
9	Wysłanie komunikatu Exception Message (bit zerowany jeżeli e_cnt = 0)
10	Wielkość komunikatu Exception Message przekracza rozmiar bufora wysyłania (bit zerowany jeżeli e_cnt = 0)
11	Próba wysłania komunikatu Exception Message w czasie transmisji (bit zerowany jeżeli e_cnt=0)

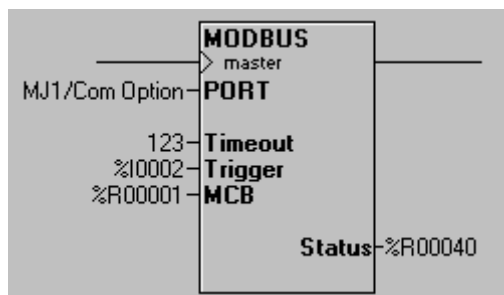
## Mapowanie master

W celu uzyskania dostępu do punktu sterownika (zmiennej w pamięci sterownika) w połączeniu Modbus urządzenie, master musi być skonfigurowane poprzez podanie typu i adresu punktu. Zwykle jest to realizowane za pomocą jednej z dwóch metod. Pierwsza metoda polega na

skorzystaniu z tradycyjnego schematu adresowania, w którym najstarsza cyfra reprezentuje typ punktu a najmłodsza cyfra reprezentuje przesunięcie punktu (począwszy od punktu 1). Ponieważ sposób ten pozwala na reprezentowanie tylko czterech typów, funkcja Modbus pakuje szereg tabel danych sterownika w pojedynczą tabelę typów punktów.

Kolumna tradycyjne rejestry RTG w tabeli zamieszczonej poniżej podaje adresy początkowe dla każdej tabeli sterownika. Druga metoda wymaga skonfigurowania urządzenia master poprzez podanie określonej komendy Modbus i przesunięcia. Obsługiwane polecenia Modbus oraz odpowiadające im przesunięcia podano również w poniższej tabeli.

Zmienna sterownika	Maksymalny zakres	Tradycyjna zmienna Modbus	Polecenie (-a) Modbus	Przesunięcie Modbus (offset)
%I1	2048	10001	Odczyt stanu cewki (2)	00000
%IG1	256	13001		03000
%S1	256	14001		04000
%K1	256	15001		05000
%Q1	2048	00001	Odczyt stanu przekaźnika (1) Wymuszenie stanu przekaźnika (5) Wymuszenie stanu wielu przekaźników (15)	0000
%M1	2048	03001		03000
%T1	2048	06001		06000
%QG1	256	09001		09000
%AI1	512	30001	Odczyt rejestru wejściowego (4)	00000
%AIG1	32	33001		03000
%SR1	32	34001		04000
%AQ1	512	40001	Odczyt rejestru Hold Register (3) Ustawienie rejestru (6) Ustawienie wielu rejestrów (16)	00000
%R1	2048	43001		03000
%AQG1	32	46001		06000

**Blok funkcyjny MODBUS MASTER**

*Modbus Master*

- Parametr **PORT** określa port komunikacyjny otwarty wcześniej z programu sterującego, dla którego parametr Protocol został ustawiony na Modbus ASCII lub Modbus RTU.
- Parametr **Timeout** może być zmienną lub stałą dziesiętną z przedziału 0 do 1023. Określa on dopuszczalny czas oczekiwania pomiędzy poleceniem Modbus a odpowiedzią na to polecenie. Wartość tego parametru podawana jest w dziesiątych częściach sekundy (tzn. 100 = 10.0 sekund).
- Parametr **Trigger** musi być zmienną typu binarnego. Po zmianie wartości tego bitu z 0 na 1 blok wysyła komunikat Modbus zdefiniowany w bloku sterowania komunikatami (MCB). Jeżeli parametr ten ma wartość 0, następuje skasowanie słowa statusu.
- Parametr **MCB** (Message Control Block) musi być określony poprzez podanie adresu rejestru. Rejestr ten jest pierwszym z sześciu (6) rejestrów zawierających informacje sterujące bloku.
  - Słowo 1 **Slave ID** - wartość z przedziału 1 do 247, określająca urządzenie do którego kierowany jest komunikat
  - Słowo 2 **Modbus Command** - Polecenie Modbus wysyłane do urządzenia slave (proszę porównać z listą obsługiwanych poleceń)
  - Słowo 3 **Slave Offset** - Adres początku danych zapisywanych lub odczytywanych z urządzenia Modbus Slave
  - Słowo 4 **Data Length** - Ilość danych do odczytu lub zapisu
  - Słowo 5 **Controller Reference type** - Numerowana lista typów rejestrów sterownika (proszę porównać z informacjami odnośnie numerowania typów rejestrów)
  - Słowo 6 **Controller Reference Offset** - Numer rejestru sterownika - 1

W zamieszczonym poniżej przykładzie odczytywane są 32 bity danych, począwszy od bitu 17 z urządzenia slave ID 34. Odczytane dane są umieszczane w rejestrach sterownika, począwszy od %R425:

Słowo 1 = 34 (Identyfikator ID urządzenia slave = 34)  
 Słowo 2 = 1 (Polecenie Modbus #1 = Odczyt statusu przekaźnika)  
 Słowo 3 = 16 (Rozpocznij od 16-go bitu)  
 Słowo 4 = 32 (Odczytaj 32 bity)  
 Słowo 5 = 8 (Typ docelowej zmiennej - %R)  
 Słowo 6 = 424 (Przesunięcie rejestru docelowego - 424 [425-1])

Parametr Status to rejestr typu WORD (16-bitowy) określany poprzez podanie adresu. Zawiera on wynik wykonywania bloku funkcyjnego.

Interpretacja bitów w słowie statusu:

Numer bitu	Status
1	Żądanie pomyślnie wykonane (OK)
2	Żądanie nie wykonane (proszę porównać z informacjami o błędach)
3	Wartość identyfikatora ID poza zakresem
4	Przekroczenie długości ramki Modbus
5	Nie obsługiwane polecenie
6	Niewłaściwa zmienna sterownika
7	Zarezerwowany
8	Zarezerwowany
9	Przekroczenie czasu oczekiwania (timeout)
10	Błąd ramki lub parzystości
11	Niewłaściwa suma kontrola z urządzenia slave
12	Niewłaściwy format z urządzenia slave
13	Odrzucenie polecenia przez urządzenie slave
14	Odrzucenie adresu przez urządzenie slave
15	Odrzucenie danych przez urządzenie slave
16	Błąd w urządzeniu slave

Funkcja przesyła sygnał jeżeli wykorzystywany port jest otwarty i gotowy do komunikacji.

## Bloki funkcyjne specjalne

---

### Wprowadzenie

Bloki funkcyjne specjalne to bloki, nie mieszczące się w standardowej klasyfikacji bloków funkcyjnych. Są to bloki Stepper Move i PID.

#### Blok funkcyjny Stepper Move

Blok funkcyjny Stepper Move zapewnia interfejs pomiędzy Cscape a modułem SmartStack jednoosiowego sterownika silnika krokowego STP100. Moduły SmartStack to moduły rozszerzeń dla sterowników Horner APG serii OCS, NX oraz QX.



Moduł STP100 wymaga podania siedmiu (7) lub czternastu (14) kolejnych rejestrów wyjść analogowych (%AQ). W celu zaprogramowania modułu STP100 należy przesunąć odpowiednie dane do rejestrów %AQ.

Blok funkcyjny Stepper Move jest centralnym punktem, zbierającym informacje z różnych miejsc programu sterującego i przesyłającym te dane do modułu STP100 za pomocą pojedynczej instrukcji.

Dodatkowo, blok funkcyjny Stepper Move posiada wbudowany kalkulator StepCalc, obliczający graf profitów przemieszczeń w oparciu o wartości wybrane przez użytkownika.

## Konfiguracja

### Uwaga:

Przed rozpoczęciem konfigurowania tego bloku funkcyjnego należy sprawdzić konfigurację modułu SmartStack.

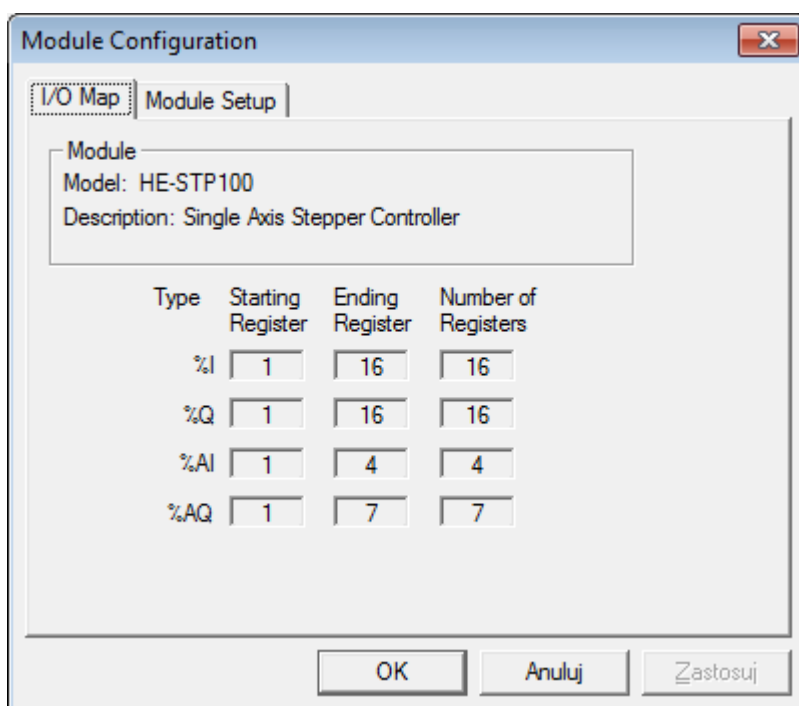
Programista musi określić szereg parametrów. Parametry te mogą być stałymi numerycznymi, rejestrami określanymi poprzez podanie adresu lub rejestrami określanymi poprzez podanie nazwy.

- **INDEXED MOVE** - Opcję tę należy zaznaczyć w celu włączenia funkcji Indexed Move tego bloku funkcyjnego. Również moduł SmartStack musi być skonfigurowany do korzystania z funkcji Indexed Move. Po włączeniu funkcji Indexed Move wymagane jest określenie siedmiu (7) dodatkowych rejestrów (w sumie 14).
- **STEPPER STARTING %AG** - Adres pierwszego rejestru %AQ przypisanego do modułu Stepper SmartStack. Informację tę można odczytać z konfiguracji modułów SmartStack Stepper Module.
- **DESTINATION POSITION** - Rejestr 32-bitowy. Rejestr ten zawiera pozycję, w której ruch ma zostać zakończony. Wartość musi mieścić się w przedziale od -8,388,608 to +8,388,607.
- **VELOCITY POSITION** - Rejestr 16-bitowy. Zakres wartości z przedziału 20 do +65535.
- **BASE VELOCITY** - Rejestr 16-bitowy. Zakres wartości z przedziału 1 do +8190.
- **RUNNING VELOCITY** - Rejestr 16-bitowy. Zakres wartości z przedziału 2 do +8191. A
- **CCELERATION POSITION** - Rejestr 16-bitowy. Czas podawany w milisekundach (ms). Zakres wartości z przedziału 1 do +27300.
- **DECELERATION TIME** - Rejestr 16-bitowy. Czas podawany w milisekundach (ms). Zakres wartości z przedziału 0 do +27300. Wymienione poniżej rejestry są wykorzystywane wyłącznie w przypadku korzystania z funkcji Indexed Move.
- **INDEX DESTINATION POSITION** - Rejestr 32-bitowy.
- **INDEX DECELERATION** - Rejestr 16-bitowy.
- **INDEX WINDOW OPEN** - Rejestr 32-bitowy.
- **INDEX WINDOW CLOSED** - Rejestr 32-bitowy.

### Działanie

Omawiany blok funkcyjny zbiera wszystkie dane, a następnie zapisuje je jako grupę do modułu Stepper Controller SmartStack Module. Patrząc od strony technicznej, sama operacja zapisu jest realizowana w następnym cyklu obsługi wejść/wyjść. Blok ten znacznie ułatwia programowanie, ponieważ normalnie wymagane by było wprowadzenie sześciu lub dziesięciu pojedynczych bloków funkcyjnych. Instrukcja Stepper Move pozwala na realizację tego zadania za pomocą tylko jednego bloku funkcyjnego.

Rejestry przypisane do modułu Stepper Controller SmartStack Module są przypisywane domyślnie, w czasie konfigurowania sterownika. Dokładna pozycji w module obszarów %I, %Q, %AI i %AQ jest określana na podstawie liczby modułów SmartStack podłączonych do sterownika oraz na podstawie fizycznego pozycji modułu HE800STP100 na stosie. Jest to typowy sposób konfigurowania, w którym STP100 jest pierwszym (lub jedynym) modułem SmartStack i nie jest wykorzystywana funkcja Indexed Move.



*Okno dialogowe do konfigurowania bloku funkcyjnego Stepper Move*

Należy zwrócić uwagę na parametr STARTING REGISTER tego modułu, a zwłaszcza dla %AQ. Informacja ta jest wykorzystywana w oknie dialogowym do konfigurowania. W przykładzie tym, sterownik silnika krokowego umieszczony jest pod adresem %AQ01 i wymaga podania siedmiu (7) kolejnych rejestrów. Dane te należą do pola Stepper Starting %AQ okna dialogowego do konfiguracji bloku funkcyjnego.

**Uwaga:**

Jeżeli zarówno moduł jak i blok funkcyjny skonfigurowane są do korzystania z funkcji Indexed Move, należy podać czternaście (14) kolejnych rejestrów %AQ.

Najpierw należy upewnić się, czy moduł SmartStack może pracować, poprzez sprawdzenie Bitów Statusu, %I1 do %I16. Jeżeli dowolny z Bitów błędów ma wartość 1, należy usunąć błąd i wysłać polecenie CLEAR ERRORS. Stan Bitów statusu zależy od poprzednio wysłanego polecenia. Nie należy wysyłać nowego polecenia (za wyjątkiem polecenia IMMEDIATE STOP lub DECELERATE AND STOP), do momentu zakończenia wykonywania poprzednio wysłanego polecenia.

Po doprowadzeniu sygnału do tego bloku funkcyjnego skonfigurowane parametry są przesyłane do modułu STP100. Patrząc od strony technicznej, sama operacja zapisu jest realizowana w następnym cyklu obsługi wejść/wyjść.

**Uwaga:**

NIE WOLNO wykonywać instrukcji Stepper Move do momentu zakończenia wykonywania poprzedniego polecenia.

Polecenia są wysyłane poprzez ustawienie odpowiedniego bitu polecenia w przestrzeni adresowej %Q modułu Stepper, po zakończeniu wykonywania instrukcji Stepper Move.

**Kalkulator StepCalc**

Oprogramowanie Cscape posiada wbudowany kalkulator StepCalc.

**Uwaga:**

Kalkulator StepCalc nie zmienia zawartości rejestrów sterowników Horner APG

W czasie pracy można wprowadzić eksperymentalne wartości do tego kalkulatora. Kalkulator sprawdzi, czy wprowadzone dane są poprawne z punktu widzenia instrukcji Stepper Move, a następnie przedstawione zostaną kluczowe wartości w celu sprawdzenia, czy ruch może być zrealizowany przez fizyczny silnik krokowy. Ostatecznie można też wyświetlić wykres lub profil planowanego ruchu.

Wartościami, które są umieszczone w odpowiednich Rejestrach sterownika krokowego za pośrednictwem instrukcji Stepper Move, są **Velocity Resolution, Base Velocity, Running Velocity, Acceleration Time i Deceleration Time**. Kalkulator StepCalc wymaga wprowadzenia tych wartości.

Parametr **Number of Pulses** jest opcjonalny. Wartość tego parametru jest wykorzystywana w przypadku profilowania ruchu trapezowego. Dopuszczalne wartości tego parametru to 0 do +16777215. Wartość 0 (zero) jest wykorzystywana do profilowania ruchów typu JOG i TRIANGULAR.

Pozostałe parametry są dopasowywane i obliczany jest następnie profil. Maksymalne dopuszczalne wartości to:

- VELOCITY RESOLUTION – Wartość z zakresu 20 do 65535.
- BASE VELOCITY – Wartość z zakresu 1 do 8190.
- RUNNING VELOCITY -- Wartość z zakresu 2 do 8191. Musi być większa od BASE VELOCITY.
- ACCELERATION TIME – Wartość z zakresu 1 do 27300.
- DECELERATION TIME – Wartość z zakresu 0 do 27300.

Wartości te mogą być dodatkowo ograniczane przez inne wartości, wprowadzane do kalkulatora. StepCalc generuje ostrzeżenie w przypadku przekroczenia dopuszczalnych wartości. Po wprowadzeniu odpowiednich wartości należy kliknąć przycisk UPDATE. Powoduje to aktualizację wartości wyświetlanej w oknie dialogowym. W celu wyświetlenia profilu ruchu, należy wcisnąć przycisk Calc/Graph.

## Bloki funkcyjne PID

### Informacje ogólne

Cscape zawiera dwa (2) bloki funkcyjne PID (Proportional/Integral/Derivative) (proporcjonalno/całkowo/różniczkowe): PID-ISA i PID o niezależnych wyrazach. Różnica pomiędzy tymi blokami funkcyjnymi polega na wpływie, jaki ma współczynnik wzmocnienia proporcjonalnego na wynik końcowy. Poniżej przedstawiono odpowiadające tym blokom funkcyjnym równania:

#### Blok funkcyjny PID o niezależnych wyrazach

Sygnał sterujący regulatora =  $K_p * U_{chyb} + K_i * U_{chyb} * dt + K_d * pochodna + odchyłka regulacji$

#### Blok funkcyjny PID-ISA

Sygnał sterujący regulatora =  $K_p * (U_{chyb} + (U_{chyb} * dt/T_i) + T_d * pochodna) + odchylenie regulacji$

Gdzie:

- $dt$  = Przyrost czasu
- $Pochodna = (U_{chyb} - Poprzedni\ uchyb)/dt$

lub

- $Pochodna = (Wartość\ sygnału\ sterowanego - Poprzednia\ wartość\ sygnału\ sterowanego)/dt$

[wybierane przez użytkownika w czasie konfigurowania].

- $T_i$  = Czas całkowania
- $T_d$  = Czas różniczkowania

Równania te pokazują, że w regulatorze PID o niezależnych wyrazach, wartość  $K_p$  jest wykorzystywana samodzielnie, natomiast w regulatorze PID-ISA, wartość  $K_p$  jest wykorzystywana w połączeniu z wartościami  $K_i$  i  $K_d$ .

Regulator PID o niezależnych wyrazach to standardowa funkcja, ale dostrajanie regulatora PID-SA jest łatwiejsze.

## Rejestry bloku funkcyjnego PID

Każdy blok funkcyjny PID korzysta z tablicy zawierającej piętnaście (15) rejestrów WORD (16-bitowych). Są to zwykle rejestry typu %R. Razem tworzą tzw. Tabelę parametrów.

Pozycja	Parametr	Jednostki	Przedział	Opis
0	Sample Period (Okres próbkowania)	10 ms	0 do 65535	Najmniejszy odstęp czasowy (mierzony w setnych częściach sekundy) pomiędzy dwoma kolejnymi wykonaniami bloków funkcyjnych PID.
1	Dead Band (Górna granica strefy nieczułości)	Jednostki bezwym. PV	0 do 32000	Definiuje górną i dolną granice strefy nieczułości w jednostkach bezwymiarowych PV. Parametry te należy ustawić na 0 (zero) jeżeli nie są wymagane strefy nieczułości. W czasie procesu strojenia parametrów regulatora PID parametry te powinny być ustawione na 0 (zero). Zdefiniowanie granic strefy nieczułości, po dostrojeniu regulatora PID, pozwala na uniknięcie małych zmian sygnału ustawiającego CV, powodowanych przez nieznaczące zmiany uchybu.
2	Dead Band (Dolna granica strefy nieczułości)	Jednostki bezwym. PV	0 do 32000	

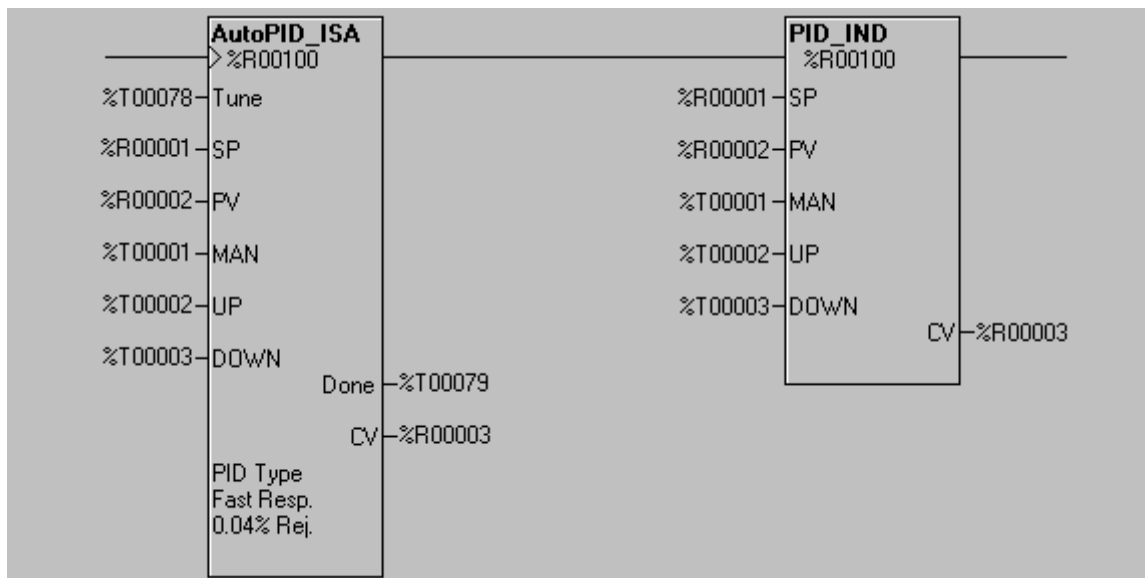
3	Proportional Gain – Kp (Współczynnik wzmocnienia proporcjonalnego Kp) (Kp)	Procenty	0 do 327.67 %	Wartość współczynnika wzmocnienia proporcjonalnego (Kp) należy podać w procentach. Wartość 100 odpowiada wzmocnieniu jednostkowemu (wzmocnienie 1).
4	Derivative Gain – Kd (Współczynnik wzmocnienia różniczkowego Kd) (Kd)	10 ms	0 do 327.67 sek.	Wprowadzana w setnych częściach sekundy. W równaniu PID, wartość ta jest wykorzystywana w następujący sposób: $Kd * \text{Przyrost Uchybu} / \text{Przyrost Czasu}$ .
5	Integral Rate – Ki (Współczynnik wzmocnienia całkowego Ki) (Ki)	1000 sekund	0 do 32.767 powtórzeń na sekundę	Liczba powtórzeń na sekundę - efektywnie jest to natężenie całkowania. W równaniu PID, wartość ta jest wykorzystywana w następujący sposób: $Kd * \text{Uchyb} * \text{Przyrost Czasu}$ .
6	CV Bias	Jednostki bezwym. parametru CV	-32000 do +32000	Odchyłka regulacji (przesunięcie punktu pracy) - wartość całkowita w jednostkach bezwymiarowych CV, dodawana do sygnału wyjściowego bloku PID, przed zastosowaniem wartości granicznych prędkości zmian i amplitudy.
7	Upper Clamp (Górna granica wartości sygnału wyjściowego)	Jednostki bezwym. parametru CV	-32000 do +32000	Wartości określające największą i najmniejszą wartość sygnału wyjściowego. Parametr ten musi mieć większą wartość od parametru CV Lower Clamp.
8	Lower Clamp (Dolna granica wartości sygnału wyjściowego)	Jednostki bezwym. parametru CV	-32000 do +32000	
9	Minimum Slew Time (Minimalny czas narastania sygnału wyjściowego)	sekundy pełnego przemieszczenia	0 do 32000 sek. Do przesunięcia o 32000 CV	Maksymalna prędkość narastania sygnału wyjściowego CV.

10	Config Word - Parametr konfiguracyjny	brak	brak	Do użytku wewnętrznego - nie wolno modyfikować tej wartości
11	Manual Command (Sygnał sterujący w trybie ręcznym)	Jednostki bezwym. parametru CV	Nadąża za parametrem CV w trybie automatycznym lub określa wartość CV w trybie ręcznym	W trybie automatycznym, rejestr ten nadąża za wartością sygnału wyjściowego CV. W trybie ręcznym, rejestr ten zawiera wartość wykorzystywaną do wyznaczenia wartości CV w obrębie przedziału wyznaczonego przez Górną i Dolną granicy wartości sygnału sterującego oraz Minimalny czas narastania sygnału.
12	Internal SP (Punkt pracy regulatora SP)	Wykorz. przez sterownik PLC	brak	Odwzorowuje SP.
13	Internal PV (Wartość wielkości regulowanej PV)	Wykorz. przez sterownik PLC	brak	Odwzorowuje PV.
14	Internal CV (Sygnał ustawiający CV)	Wykorz. przez sterownik PLC	brak	Nadąża za parametrem CV.

Każdy blok funkcyjny PID musi korzystać z oddzielnej Tablicy parametrów, nawet jeżeli wartości w poszczególnych tablicach są identyczne. Bloki funkcyjne PID nie mogą zajmować tego samego obszaru pamięci.

Przed wywołaniem bloku funkcyjnego PID muszą być podane rejestry tablicy o numerach 0 do 9. Najłatwiejsza metoda realizacji tego zadania to skorzystanie z funkcji Tune okna dialogowego oprogramowania Cscape do konfigurowania bloków funkcyjnych PID. Można jednak również wpisywać do tych rejestrów wartości z poziomu programu sterującego.

## Regulator PID o niezależnych wyrazach i regulator PID-ISA



*Blok funkcyjny PID z automatycznym dostrojeniem i bez automatycznego dostrajania*

Jako parametry wejściowe, do bloku tego wpisywana jest wartość pięciu (5) zmiennych zewnętrznych - dwie z nich to wartości typu Word (16-bitowe), a trzy (3) to wartości binarne (1-bitowe). Parametry wyjściowe tego bloku funkcyjnego to jedna zmienna typu Word (16-bitowa) oraz jeden pojedynczy bit.

Każdy blok funkcyjny PID korzysta z tablicy zawierającej piętnaście (15) rejestrów WORD (16-bitowych), najczęściej typu %R. Jest to tzw. Tablica zmiennych (proszę porównać z opisem poniżej).

Jeżeli po doprowadzeniu sygnału do tego bloku funkcyjnego nie jest doprowadzany sygnał do parametru wejściowego MAN (Manual Input), blok funkcyjny jest automatycznie przełączany do trybu automatycznego. Najpierw sprawdzane jest, czy od momentu ostatniego wywołania tego bloku funkcyjnego upłynął minimalny czas, określony za pomocą parametru Sample Period. Jeżeli czas ten upłynął, następuje wykonanie algorytmu PID i zaktualizowanie sygnału regulowanego (CV).

Jeżeli do bloku funkcyjnego doprowadzony zostanie sygnał oraz sygnał doprowadzony zostanie również na wejście MAN, blok funkcyjny jest przełączany do trybu ręcznego. Wartość zmiennej sterującej CV jest wyznaczana z wykorzystaniem parametru Manual Command, przechowywanego w tablicy zmiennych. Jeżeli dodatkowo doprowadzany jest sygnał do wejść UP lub DOWN, wartość zmiennej sterującej jest zwiększana lub zmniejszana o 1, za każdym wywołaniem algorytmu PID.

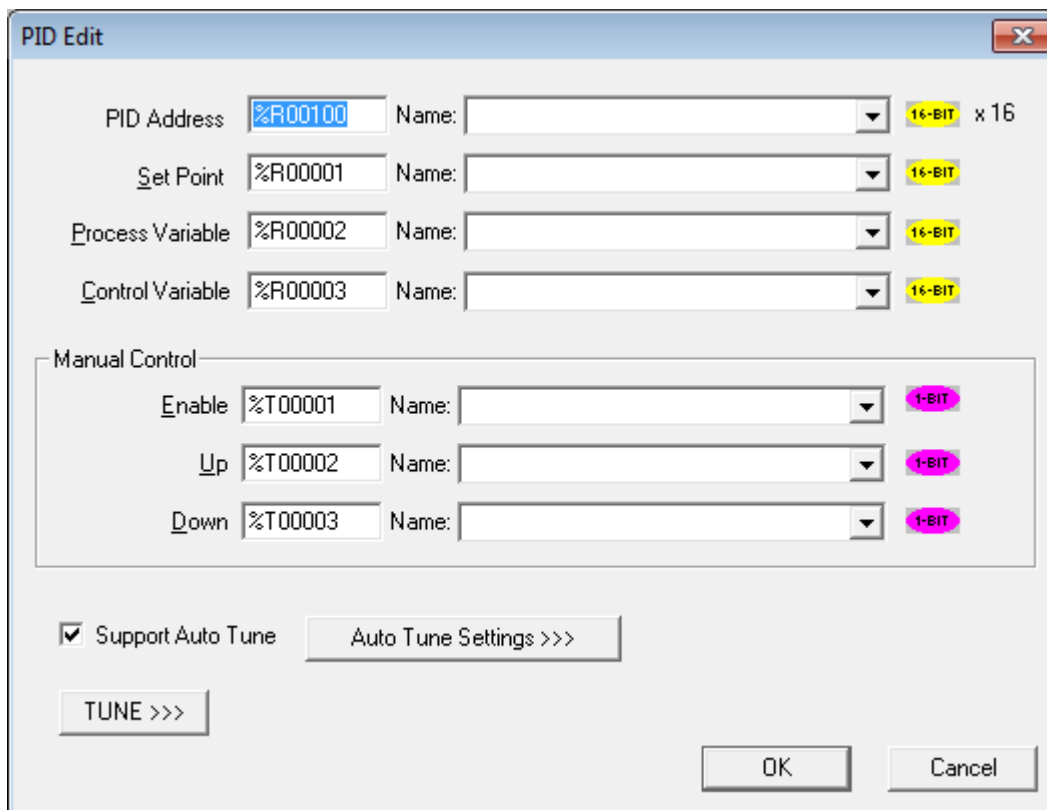


Zarówno w trybie ręcznym jak i automatycznym, sygnał sterujący CV jest ograniczony przez wartości graniczne CV Lower Clamp i CV Upper Clamp jak i przez minimalną prędkość narastania sygnału Minimal Slew Time. Jeżeli wartość sygnału sterującego przekroczy jedną z wartości granicznych lub prędkość narastania sygnału, jest modyfikowana stosownie do tych granic. Sygnał sterujący CV zmienia się w kierunku przeciwnym do wartości granicznej, jeżeli wewnętrzna zmienna Internal CV spadnie poniżej tej wartości granicznej lub czas narastania sygnału spadnie poniżej wartości granicznej CV Slew Rate. Jest to mechanizm zapobiegający przekraczaniu wartości granicznych oraz umożliwiające płynne przejście pomiędzy trybem automatycznym i ręcznym.

Po doprowadzeniu sygnału do bloku funkcyjnego PID, jest on wykonywany wyłącznie pod warunkiem, że upłynął pewien czas (określony za pomocą parametru Sample Time) od momentu ostatniego wykonywania tego bloku funkcyjnego. Po ustawieniu parametru Sample Time Period na 0, blok funkcyjny PID jest wykonywany w każdym cyklu jednostki centralnej. W żadnym wypadku nie może on jednak być wykonywany częściej niż co 10 milisekund. Przykładowo, jeżeli cykl pracy jednostki centralnej sterownika OCS wynosi 9 milisekund, a parametr Sample Period został ustawiony na 0 (zero), blok funkcyjny jest wykonywany co drugi cykl pracy jednostki centralnej, czyli co 18 ms. Blok funkcyjny nie może być wykonywany w każdym cyklu, ponieważ czas trwania cyklu jest krótszy od minimalnego czasu pomiędzy kolejnymi wywołaniami bloku funkcyjnego PID (10 milisekund). Blok funkcyjny może więc być wykonany dopiero w następnym cyklu, czyli 9 milisekund później.

## Konfigurowanie bloku funkcyjnego PID

Bloki funkcyjne PID są konfigurowane za pomocą okna dialogowego PID Edit.



**PID Edit**

PID Address:  Name:  16-BIT x 16

Set Point:  Name:  16-BIT

Process Variable:  Name:  16-BIT

Control Variable:  Name:  16-BIT

Manual Control:

Enable:  Name:  1-BIT

Up:  Name:  1-BIT

Down:  Name:  1-BIT

Support Auto Tune    Auto Tune Settings >>>

TUNE >>>

OK    Cancel

- **PID ADDRESS** - Należy wprowadzić adres rejestru lub wybrać nazwany rejestr. Jest to adres początkowy piętnastu (15) kolejnych rejestrów WORD (26-bitowych), wykorzystywanych przez blok funkcyjny PID do przechowywania parametrów. Parametr ten nie może być stałą dziesiętną.
- **SETPOINT** - Należy wprowadzić adres rejestru lub wybrać nazwany rejestr. Parametr ten określa zdefiniowaną przez użytkownika wartość sygnału nastawiającego (Process Setpoint). Można również wprowadzić 16-bitową stałą dziesiętną ze znakiem.
- **PROCESS VARIABLE** - Należy wprowadzić adres rejestru lub wybrać nazwany rejestr. Parametr ten (zwykle %AI) określa wielkość regulowaną (Process Variable), odczytywaną z procesu. Parametr ten NIE MOŻE być stałą dziesiętną.
- **CONTROL VARIABLE** - Należy wprowadzić adres rejestru lub wybrać nazwany rejestr. Parametr ten (zwykle %AQ) określa wielkość sterującą (Control Variable), wysyланą do sterowanego procesu. Parametr ten NIE MOŻE być stałą dziesiętną.
- **MANUAL INPUT** - Należy wprowadzić adres rejestru lub wybrać z listy nazwany rejestr, będący parametrem wejściowym MAN ustawianym przez użytkownika, decydującym o przełączeniu na tryb ręcznego sterowania. Jest to rejestr logiczny (1-bitowy), zwykle %T.
- **UP INPUT** - Należy wprowadzić adres rejestru lub wybrać z listy nazwany rejestr, będący parametrem wejściowym UP sterowanym przez użytkownika. Jest to rejestr logiczny (1-bitowy), zwykle %T.
- **DOWN INPUT** - Należy wprowadzić adres rejestru lub wybrać z listy nazwany rejestr, będący parametrem wejściowym DOWN sterowanym przez użytkownika. Jest to rejestr logiczny (1-bitowy), zwykle %T.
- **TUNE** - Należy kliknąć na przycisk
- **TUNE** w celu wywołania okna dialogowego PID Element Tuning Dialog.

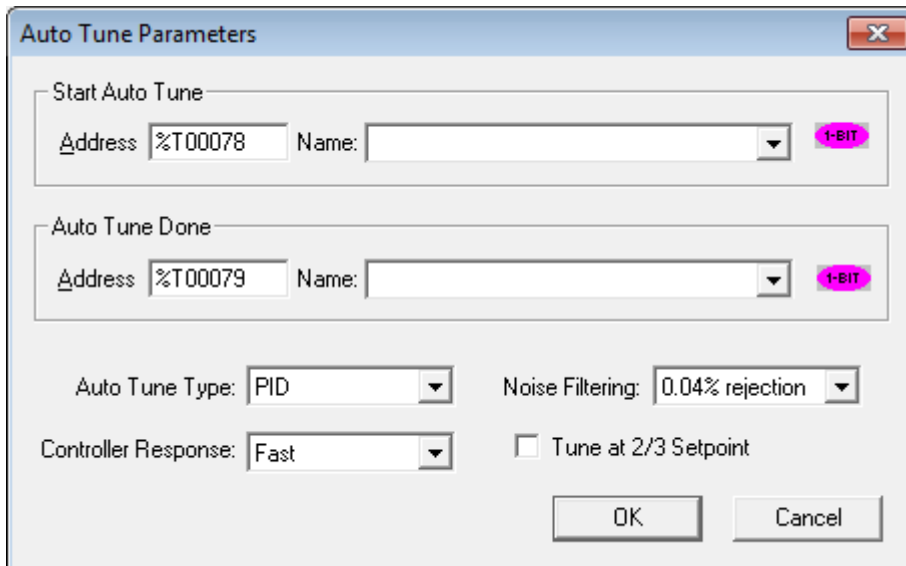
## Automatyczne dostrajanie bloku funkcyjnego PID

Automatyczne dostrajanie bloków PID wnosi szereg nowych możliwości. Cykl automatycznego dostrajania jest rozpoczynany po ustawieniu parametru wejściowego AUTOTUNE. Parametr ten musi mieć wartość 1 przez cały cykl automatycznego dostrajania. Zmiana jego wartości na 0 powoduje przerwanie cyklu automatycznego dostrajania i powrót do poprzednich wartości parametrów. Wynikiem działania cyklu AUTOTUNE jest zmiana wartości określonych współczynników sterownika oraz ustawienie wartości parametru wyjściowego AUTOTUNE DONE. Blok PID powraca do poprzedniego stanu, automatycznie lub ręcznie. Zaprzestanie doprowadzania sygnału na wejście AUTOTUNE powoduje ustawienie sygnału wyjściowego AUTOTUNE DONE. Blok jest wtedy gotowy do rozpoczęcia następnego cyklu automatycznego dostrajania. Nowe wartości współczynników do dostrajania są zapisywane do odpowiednich rejestrów.

Poza funkcją automatycznego dostrajania nie ma żadnych różnic pomiędzy blokiem PID z automatycznym dostrajaniem i bez automatycznego dostrajania.

Funkcja automatycznego dostrajania pozwala na dokonanie pewnego eksperymentu na procesie i wykorzystanie otrzymanych w ten sposób wyników do obliczenia parametrów bloku funkcyjnego PID, dostosowanych do wymogów procesu i żadanego działania bloku funkcyjnego PID.

Po zaznaczeniu opcji Support Auto Tune, wyświetlane jest okno dialogowe, pozwalające na wprowadzenie parametrów automatycznego dostrajania:



- **Start Auto Tune** - Rejestr określony za pomocą adresu, będący parametrem wejściowym decydującym o rozpoczęciu procesu automatycznego dostrajania.
- **Auto Tune Done** - Rejestr określony za pomocą adresu, będący parametrem wyjściowym, ustawianym po zakończeniu automatycznego dostrajania.
- **Auto Tune Type** - Pole to pozwala na wybranie wyrazów automatycznie dostrajanych: PID, PI lub P.
- **Controller Response** - Pole to pozwala na zdefiniowanie względnej prędkości pętli PID po dostrojeniu.
- **Noise Filtering** - Pole to pozwala na zdefiniowanie dolnej i górnej granicy sygnału zadającego w czasie realizacji próby automatycznego dostrojenia. Dla procesów odznaczających się większym poziomem zakłóceń należy wprowadzić większą wartość procentową.
- **Tune at 2/3 Setpoint** - Po zaznaczeniu tej opcji, w czasie próby automatycznego dostrojenia sygnał wyjściowy będzie zmieniany dla 2/3 sygnału zadającego. Opcję tę należy zaznaczyć, jeżeli w czasie próby automatycznego dostrajania nie powinna być przekroczona wartość sygnału zadającego.

### Korzystanie z funkcji automatycznego dostrajania

Przed rozpoczęciem automatycznego dostrajania wymagane jest częściowe zdefiniowanie parametrów bloku PID. Dotyczy to zwłaszcza parametrów Setpoint, Sample Period, Upper and Lower Clamp, Error Term i Output Polarity. Poprzednie wartości współczynników dla wyrazów proporcjonalnego, całkowego i różniczkowego nie mają żadnego wpływu na działanie funkcji automatycznego dostrajania.

Przy pozostawieniu domyślnych wartości parametrów, funkcja automatycznego dostrajania oblicza współczynniki wyrazów proporcjonalnego, całkowego i różniczkowego, za pomocą metody Zieglera i Nicholasa. Metoda ta dobrze się sprawdza w przypadku wielu typowych procesów z opóźnieniem lub dwiema inercjami oraz z wielkościami regulowanymi pozbawionymi zakłóceń.

W pewnych sytuacjach, w celu zwiększenia efektywności tej funkcji, można zmienić domyślne wartości parametrów. Zmiana taka ma wpływ wyłącznie na obliczanie współczynników automatycznego dostrajania. Domyślnie w czasie wykonywania funkcji automatycznego dostrajania obliczane są współczynniki dla wyrazów proporcjonalnego/całkowego/różniczkowego (PID), ale można także wybrać obliczanie wyłącznie dla wyrazów proporcjonalno/całkowego (PI) lub tylko dla wyrazu proporcjonalnego (P). Sterowanie PI jest zwykle bardziej stabilne w przypadku procesów bez jakichkolwiek opóźnień, ale z inercją. Pełne sterowanie PID daje lepsze rezultaty w przypadku procesów z opóźnieniem. Wszystkie reguły dostrajania PID zakładają, że w procesie występują znacznie opóźnienia i mogą nie być przydatne dla innych procesów. Algorytm automatycznego dostrajania uwzględnia te tryby poprzez ustawienie nie wykorzystywanych współczynników na zero. Można później ręcznie zwiększyć ich wartość w celu aktywowania pozostałych trybów.

Pole Controller Responser pozwala na zwiększenie tłumienia sterownika w celu zmniejszenia przeregulowania. W przypadku typowego procesu Zieglera i Nicholasa, domyślna wartość FAST parametru Controller Response generuje pewne przeregulowanie oraz zanikające w stosunku 4:1 dudnienia. Wybranie wartości MEDIUM powoduje nieznaczne przeregulowanie. Wartość SLOW nie powoduje przeregulowania. W przypadku procesów wykraczających poza optymalny zakres reguł Zieglera i Nicholasa, w celu uzyskania odpowiedniej odpowiedzi może być konieczne ustawienie wartości VERY SLOW.

W czasie automatycznego dostrajania algorytm wykrywa przechodzenie procesu powyżej i poniżej dolnej granicy aktywnej sygnału zadającego. W odniesieniu do sygnału zadającego stosowana jest histereza, mająca na celu uniknięcie fałszywych wskazań, powodowanych przez zakłócenia procesu. Domyślne pasmo histerezy wynosi 0.04% pełnego zakresu skali. W przypadku procesów z dużymi zakłóceniami, należy tę wartość zwiększyć (parametr NOISE FILTERING).

Po wprowadzeniu wyższych wartości w polu Noise Filtering, algorytm automatycznego dostrajania będzie dobierał nieco wolniejsze i stabilniejsze współczynniki. W przypadku procesów z zakłóceniami, zalecane jest wybieranie sterowania PI, a nie sterowania PID.

### **Algorytm pracy automatycznego dostrajania**

Funkcja automatycznego dostrajania eksperymentuje na sterowanym procesie, wykorzystując otrzymane wyniki do obliczenia optymalnych współczynników PID. W czasie jej wykonywania, sygnał wyjściowy jest zmieniany w przedziale określonym przez górną i dolną granicą wartości dopuszczalnej. Zerowane są czasy przechodzenia procesu od procentowego przekroczenia sygnału zadającego (obliczanego w oparciu o filtrowanie zakłóceń) w górę i w dół, jak również przeregulowania i nie doregulowania. Po zakończeniu eksperymentu, otrzymane dane są wykorzystywane do obliczenia nowych współczynników regulatora PID.

## Inne bloki funkcyjne

Grupa ta obejmuje komentarze i połączenia pionowe.

### Blok funkcyjny komentarze

Komentarze pozwalają na wpisywanie do programu tekstów objaśniających. Można je przesyłać do sterownika. Nie mają one żadnego wpływu na wykonywanie programu sterującego, ale zajmują pamięć w sterowniku.

### Dodawanie połączenia pionowego

W celu dodania połączenia pionowego należy kliknąć na przycisk Vertical Branch . Należy przemieścić kursor do miejsca, w którym ma być umieszczone połączenie pionowe, a następnie jeden raz kliknąć myszą.

### Usuwanie połączenia pionowego

W celu usunięcia połączenia pionowego należy kliknąć przycisk Vertical Branch. Ustawić kursor nad połączeniem, które ma być usunięte. W momencie, gdy kursor ma kształt gumki do wymazywania, kliknąć na połączenie do usunięcia.

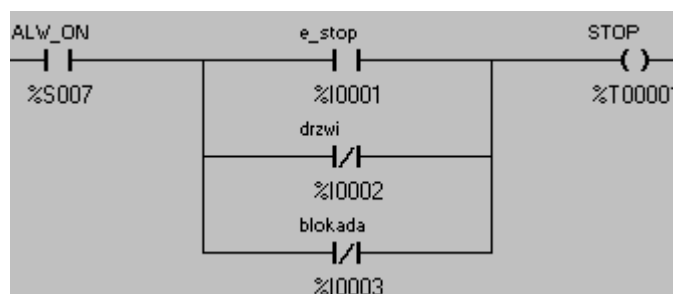
#### Ostrzeżenie:

Usunięcie połączenia pionowego może spowodować rozłączenie bloków funkcyjnych. Bloki te należy ponownie połączyć przed przestaniem programu sterującego.

#### Przykład

Zamieszczony poniżej fragment programu może pełnić funkcję blokady zabezpieczającej. Należy zwrócić uwagę, że zastosowanie pionowych połączeń pozwoliło na połączenie trzech przełączników za pomocą spójnika logicznego OR.

%I01 to przycisk zawsze otwarty, uruchamiany ręcznie. %I02 i %I03 reprezentują przełączniki zabezpieczające i są ustawiane na wartość jeden po zamknięciu drzwi.



*Gałąź pionowa*

Tak więc wciśnięcie przycisku e\_stop LUB otwarcie przełącznika drzwi LUB zadziałanie blokada powoduje ustawienie bit STOP.

## Typy danych stosowane w Cscape

### Wprowadzenie

Cscape pozwala na przechowywanie danych w wielu różnych formatach. Format musi być dobrany do sposobu interpretowania informacji. Typowe interpretacje wartości to ciągi bitów, liczby bez znaku, liczby ze znakiem, liczby rzeczywiste i ciągi znaków.

### Formaty danych

Typ	Nazwa	Opis
BOOL	Boolean	Pojedynczy bit. Może on posiadać tylko wartość '0' lub '1'.
BYTE	Byte	Ciąg 8 kolejnych bitów. Wartości typu bajt są stosowane, jeżeli ustawienia poszczególnych bitów mają większe znaczenie niż sama wartość.
WORD	Word	Ciąg 16 kolejnych bitów. Wartości typu Word są stosowane, jeżeli ustawienia poszczególnych bitów mają większe znaczenie niż sama wartość.
DWORD	Double Word	Ciąg 32 kolejnych bitów. Wartości typu Dword są stosowane, jeżeli ustawienia poszczególnych bitów mają większe znaczenie niż sama wartość.
INT	Integer	Wartość 16-bitowa ze znakiem. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale -32768 do +32768.
SINT	Short Integer	Wartość 8-bitowa ze znakiem. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale 128 do +127.
DINT	Double Integer	Wartość 32-bitowa ze znakiem. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale -2,147,483,648 do +2,147,483,647.
UINT	Unsigned Integer	Wartość 16-bitowa bez znaku. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale od 0 do 65 535.
USINT	Unsigned Short Integer	Wartość 8-bitowa ze znakiem. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale 0 (zero) do +25.

UDINT	Unsigned Double Integer	Wartość 32-bitowa bez znaku. Typ ten jest stosowany, jeżeli przewidywane jest, że wartość będzie mieścić się w przedziale 0 (zero) do 4,294,967,296.
REAL	Floating Point	Wartość 32-bitowa. Wartości te są przechowywane i przetwarzane w formacie IEEE pojedynczej precyzji (sześć cyfr). Wartość musi mieścić się w przedziale od -3.40282E+38 do +3.40282E+3
STRING	String	Ciąg znaków o zmiennej długości. Każdy ze znaków jest reprezentowany przez jeden bajt.

Zwykle, dowolny typ danych może korzystać z dowolnego rejestru sterownika. Przykładowo, wartość DINT można zapisać zarówno do rejestru Word (%R) lub rejestru Boolean (%I). Istnieje jednak ograniczenie w przypadku korzystania z rejestrów Boolean. Wartości mogą być do nich zapisywane wyłącznie od odpowiedniego bitu. Przykładowo, wartości DWORD, DINT lub UINT mogą być zapisywane do rejestrów Boolean wyłącznie w granicach wyznaczonych przez słowa (16-bitowe), np. do bitu 1, 17, 33, itp.

Należy również zachować uwagę przy zapisie danych typu innego niż Boolean do rejestrów %I i %Q. Przykładowo, przy zapisie danych typu WORD do rejestrów %I, w czasie cyklu uaktualniania wejść/wyjść, sterownik zmieni wartość zapisaną do tego typu rejestrów przez program sterujący. W celu uniemożliwienia wystąpienia takiej sytuacji, programista powinien zadbać, aby z rejestrów %I wyłącznie odczytywać dane (dane zapisane do tych rejestrów zostaną zmienione w czasie cyklu obsługi wejść/wyjść) lub korzystać z rejestrów %I, którym nie są przypisane żadne, fizyczne punkty wejść/wyjść.

Bity w rejestrach typu Word można traktować jako wartości typu Boolean. Wtedy stosowana jest konwencja adresowania bitów, w której podawany jest typ rejestru, przesunięcie oraz przesunięcie bitu.

## Kolejność zapisywania

Wartości 32-bitowe (DWORD, DINT, UDINT) zajmują kolejne 32 bity lub dwa (16-bitowe) rejestry. Przykładowo, wartość DINT przechowywana w rejestrze %R43 zajmuje rejestry %R43 i %R44.

Wartości 32-bitowych są zapisywane na zasadzie „najpierw słowo młodsze. Przykładowo, dla wartości DINT równej "65540" (0000000000000001 000000000000100), zapisywanej w rejestrze %R43, rejestr %R43 zawiera "4" a rejestr %R44 zawiera "1".

Wartości bajtowe (jak na przykład dane typu STRING) są zapisywane na zasadzie „najpierw bajt starszy”. Przykładowo, w celu zapisania łańcucha znaków "31" w rejestrze %R43, należy zapisać wartość heksadecymalną 3133 (dziesiętnie 12595).

## Zasoby systemowe sterowników

### Wprowadzenie

W niniejszym rozdziale opisano zasoby systemowe rodziny sterowników Horner APG.

### Tabela zasobów systemowych

**Uwaga:**

Podane w tym punkcie informacje mają charakter ilustracyjny i zostały zamieszczone w celach porównawczych, mogą one ulec zmianie bez uprzedzenia. Kompletne i aktualne informacje podane są w instrukcji obsługi dostarczonej wraz z zakupionym sterownikiem.

**Tabela dla sterowników Horner APG serii OCS i miniOCS**

Zasób	Opis	MiniOCS	OCS100	OCS200	OCS250	RCS210
Rejestry %I	wejścia dyskretne	2048				
Rejestry %Q	wyjścia dyskretne	2048				
Rejestry %AI	wejścia analogowe	512				
Rejestry %AQ	wyjścia analogowe	512				
Rejestry %IG	globalne wejścia dyskretne pochodzące z urządzeń na sieci CsCAN	64 dla węzła				
Rejestry %QG	globalne wyjścia dyskretne pochodzące z urządzeń na sieci CsCAN	64				
Rejestry %AIG	globalne wejścia analogowe pochodzące z urządzeń na sieci CsCAN	32				
Rejestry %AQG	globalne wyjścia analogowe pochodzące z urządzeń na sieci CsCAN	32				
Rejestry %T	bity tymczasowe	2048				



Rejestry %M	bity z pamięcią stanu	2048				
Rejestry %R	rejestry z pamięcią	2048				
Rejestry %K	bity klawiatury	10	10	10	12	0
Rejestry %D	bity ekranów	200	200	300	200	0
Rejestry %S	bity systemowe	8				
Rejestry %SR	słowa systemowe	64				
Pamięć programu w formacie drabinkowym		64K				
Wyświetlacz		LCD 2x20	LCD 2x20	LCD 4x20	LCD z grafiką 240x128	brak
Klawisze		16	17	32	32	32 wirtualne
Strony tekstowe		200	200	200	300	200
Liczba pól na ekranie		10	10	10	50	10
Tabele tekstowe		200	200	200	200	200
Liczba elementów w jednej tabeli		20	20	20	20	20

Model RCS210 nie posiada fizycznego ekranu i klawiatury, ale obsługuje zdalne teksty oprogramowania Cscape.

#### Tabela dla sterowników Horner APG serii XLe, XLt oraz XL4e

Zasób	Opis	XLe	XLt	XL4e
Rejestry %I	wejścia dyskretne	2048		
Rejestry %Q	wyjścia dyskretne	2048		
Rejestry %AI	wejścia analogowe	512		
Rejestry %AQ	wyjścia analogowe	512		
Rejestry %IG	globalne wejścia dyskretne pochodzące z urządzeń na sieci CsCAN	64 dla węzła		

Rejestry %QG	globalne wyjścia dyskretne pochodzące z urządzeń na sieci CsCAN	64		
Rejestry %AIG	globalne wejścia analogowe pochodzące z urządzeń na sieci CsCAN	32		
Rejestry %AQG	globalne wyjścia analogowe pochodzące z urządzeń na sieci CsCAN	32		
Rejestry %T	bity tymczasowe	2048		
Rejestry %M	bity z pamięcią stanu	2048		
Rejestry %R	rejestry z pamięcią	9999	9999	50000
Rejestry %K	bity klawiatury	10	10	10
Rejestry %D	bity ekranów	1040	1040	1040
Rejestry %S	bity systemowe	8		
Rejestry %SR	słowa systemowe	202		
Pamięć programu w formacie drabinkowym		128 kB	128kB	1 MB
Wyświetlacz		LCD 128x64	LCD 160x128	LCD kolor 320x240
Klawisze		19	5	5
Strony tekstowe		1023	1023	1023
Liczba pól na ekranie		50	50	50
Tabele tekstowe		200	200	200
Liczba elementów w jednej tabeli		20	20	20

**Tabela dla sterowników Horner APG serii XL6/XL6e, XL7e oraz XL10e**

Zasób	Opis	XL6e	XL7e	XL10e
Rejestry %I	wejścia dyskretne	2048		
Rejestry %Q	wyjścia dyskretne	2048		
Rejestry %AI	wejścia analogowe	512		
Rejestry %AQ	wyjścia analogowe	512		

Rejestry %IG	globalne wejścia dyskretnie pochodzące z urządzeń na sieci CsCAN	64 dla węzła		
Rejestry %QG	globalne wyjścia dyskretnie pochodzące z urządzeń na sieci CsCAN	64		
Rejestry %AIG	globalne wejścia analogowe pochodzące z urządzeń na sieci CsCAN	32		
Rejestry %AQG	globalne wyjścia analogowe pochodzące z urządzeń na sieci CsCAN	32		
Rejestry %T	bity tymczasowe	2048		
Rejestry %M	bity z pamięcią stanu	2048		
Rejestry %R	rejestry z pamięcią	9999	50000	50000
Rejestry %K	bity klawiatury	10	10	10
Rejestry %D	bity ekranów	1040	1040	1040
Rejestry %S	bity systemowe	8		
Rejestry %SR	słowa systemowe	202		
Pamięć programu w formacie drabinkowym		128 kB	1 MB	1 MB
Wyświetlacz		LCD 320x240	LCD 800x480	LCD kolor 800x600
Klawisze		6	6	7
Strony tekstowe		1023	1023	1023
Liczba pól na ekranie		50	50	50
Tabele tekstowe		200	200	200
Liczba elementów w jednej tabeli		20	20	20

## Rejestry systemowe

### Informacje ogólne

Rejestry systemowe to specjalne rejestry informujące o operacjach wykonywanych przez sterownik i/lub sterujące ich wykonywaniem.

### Tabela rejestrów systemowych

SR#	Nazwa	Opis	Min	Max
1	User Screen Numer	Numer ekranu użytkownika	0	200
2	Alarm Screen Number	Numer ekranu alarmowego		
3	System Screen Number	Numer ekranu systemowego		
4	Self Test Result	Wynik autotestu		
5	Controller Mode (RUN..)	Stan kontrolera		
6	Scan Rate AVG	Średni czas skanu		
7	Reserved	Zarezerwowane		
8	Reserved	Zarezerwowane		
9	Edit Buffer Low			
10	Edit Buffer High			
11	Ladder Size Low			
12	Ladder Size High			
13	User Text Size Low			
14	User Text Size High			
15	System Text Size Low			
16	System Text Size High			
17	I/O Config Size Low			
18	I/O Config Size High			
19	Net Config Size Low			
20	Net Config Size High			
21	Security Data Size Low			
22	Security Data Size High			
23	Ladder CRC			
24	User Text CRC			

25	System Test CRC			
26	I/O Config CRC			
27	Net Config CRC			
28	Security Data CRC			
29	Network ID Low	ID urządzenia w sieci	1	253
30	Network Baud Rate	Prędkość transmisji	0	3
31	Network Required		0	1
32	LCD Contrast	Kontrast wyświetlacza	1	40
33	Key Toggle Mode		0	1
34	Serial Protocol	Protokół szeregowy		
35	Serial Numer Low			
36	Serial Numer High			
37	Model Numer	Numer modelu		
38	Engine Version	Wersja firmware		
39	BIOS Version	Wersja BIOSu		
40	FPGA Version	Wersja płyty głównej		
41	LCD Columns	Liczba kolumn		
42	LCD Rows	Liczba wierszy		
43	Keypad Type			
44	RTC Seconds	Sekundy	0	59
45	RTC Minutes	Minuty	0	59
46	RTC Hours	Godziny	0	23
47	RTC Day of Month	Dzień miesiąca	1	31
48	RTC Month	Miesiąc	1	12
49	RTC Year	Rok	1996	2095
50	RTC Day of Week	Dzień tygodnia	1	7
51	Network Error Count	Licznik błędów sieci		
52-55	Reserved	Zarezerwowane		
56	Last Key	Ostatni klawisz		
57	LCD Backlight	Podświetlenie		
58	User Leds	Diody użytkownika		
59-60	Reserved	Zarezerwowane		
61	Num Ids			
62-174	Reserved	Zarezerwowane		
175	CF Status	Status karty CF		
176	CF Free Low	Pozostała ilość wolnej pamięci		
177	CF Free High	Pozostała ilość wolnej pamięci		
178	CF Total Low	Całkowita pamięć CF		

179	CF Total High	Całkowita pamięć CF		
180	Reserved	Zarezerwowane		
181	Alarms Unacknowledged	Alarmy niepotwierdzone		
182	Alarms Active	Alarmy aktywne		
183	System Beep	Dźwięk systemowy	0	1
184	User Beep	Dźwięk użytkownika	0	1
185	Screen Saver	Wygaszasz ekranu	0	1
186	Screen Saver Time	Czas wygaszacza	0	1200
187	Network Usage (Avg)	Średnie użycie sieci	0	1000
188	Network Usage (Min)	Minimalne użycie sieci	0	1000
189	Network Usage (Max)	Maksymalne użycie sieci	0	1000
190	Network TX Use (Avg)	Średnie użycie TX	0	1000
191	Network TX Use (Min)	Minimalne użycie TX	0	1000
192	Network TX Use (Max)	Maksymalne użycie TX	0	1000
193	Online change			
194-199	Reserved			
200	InitRD Version			
201-205	Kernel Version			

### SR01

Numer strony użytkownika

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Min.: 0 Maks.: 200 (dla modelu OCS 200/100 w momencie drukowania)

Rejestr ten podaje/steruje numerem aktualnie pokazywanej strony użytkownika. Po wpisaniu do tego rejestru wartości 0, na wyświetlaczu nie jest pokazywana żadna strona.

### SR02

Numer strony alarmów

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Min.: 0 Maks.: 200 (dla modelu OCS 200/100 w momencie drukowania)

Rejestr ten podaje/zmienia numer aktualnie wyświetlanej strony alarmów.

### SR03

Numer strony systemowej

Język drabinkowy: Zapis/Odczyt  
Tekst: Zapis/Odczyt

Min.: 0 Maks.: 10 (dla modelu OCS 200/100 w momencie drukowania)  
Rejestr ten podaje/steruje numerem aktualnie pokazywanej strony systemowej. Po wpisaniu do tego rejestru wartości 0 na wyświetlaczu nie jest pokazywana żadna strona.

**SR04**

Wynik działania procedury samoczynnego testowania

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten zawiera wynik działania procedury samoczynnego testowania, wyrażony za pomocą ciągu bitów.

**SR05**

Tryb sterownika

Tekst: Zapis/Odczyt

Rejestr ten podaje/przełącza sterownik pomiędzy trybami RUN. DO I/O lub IDLE.

0 = IDLE  
1 = DO I/O  
2 = RUN

**SR06**

Średni czas trwania cyklu sterownika

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten podaje średni czas trwania cyklu sterownika w dziesiątych częściach milisekund (123 = 12.3 ms).

**SR07 do SR08**

ZAREZERWOWANE

**SR09 + SR10**

Bajt młodszy i starszy bufora edycji

Język drabinkowy: Odczyt  
Tekst: Odczyt

Ten 32-bitowy rejestr zawiera tymczasową, 1-, 8- lub 32-bitową wartość edytowaną na stronie tekstowej.

**SR11 + SR12**

Bajt młodszy i starszy wielkości pamięci zajmowanej przez program

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie wczytany program.

**SR13 + SR14**

Bajt młodszy i starszy wielkości pamięci zajmowanej przez teksty użytkownika

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie załadowane strony tekstowe i tabele tekstowe użytkownika.

**SR15 + SR16**

Bajt młodszy i starszy wielkości pamięci zajmowanej przez teksty systemowe

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie załadowane strony i teksty systemowe.

**SR17 + SR18**

Młodszy i starszy bajt wielkości pamięci zajmowanej przez konfigurację wejść/wyjść

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie załadowaną konfigurację wejść/wyjść.

**SR19 + SR20**

Młodszy i starszy bajt wielkości pamięci zajmowanej przez konfigurację sieciową

Język drabinkowy: Odczyt

Tekst: Odczyt



Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie wczytaną konfigurację sieciową.

**SR21 + SR22**

Młodszy i starszy bajt wielkości pamięci zajmowanej przez dane zabezpieczające

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o liczbie bajtów zajmowanych przez aktualnie załadowane dane zabezpieczające

**SR23**

Suma kontrolna programu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanym programie sterującym.

**SR24**

Suma kontrolna tekstów użytkownika

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanych stronach, tekstach i tabelach użytkownika.

**SR25**

Suma kontrolna ekranów tekstowych systemu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanych stronach, tekstach i tabelach systemowych.

**SR26**

Suma kontrolna konfiguracji wejść/wyjść

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanej konfiguracji wejść/wyjść.

**SR27**

Suma kontrolna konfiguracji sieciowej

Język drabinkowy: Odczyt

Tekst: Odczyta

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanej konfiguracji sieciowej.

**SR28**

Suma kontrola danych zabezpieczających

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wartości sumy kontrolnej wykorzystywanej do wykrywania błędów w aktualnie załadowanych danych zabezpieczających.

**SR29**

Identyfikator sieciowy ID

Język drabinkowy: Odczyt

Tekst: Zapis/Odczyt

Min.: 1 Maks.: 253 (dla modelu OCS 200/100 w trybie CsCAN, w momencie drukowania)

Rejestr ten podaje lub ustawia identyfikator sieciowy ID sterownika.

**SR30**

ZAREZERWOWANY

**SR31**

Wymagana sieć

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten zawiera status bitu Network Required. Wartość "1" informuje, że sieć jest wymagana i sterownik rejestruje wszystkie błędy sieciowe. Wartość "0" informuje, że sieć nie jest wymagana, w związku z czym sterownik nie rejestruje błędów sieciowych.

**SR32**

Kontrast wyświetlacza LCD

Język drabinkowy: Odczyt

Tekst: Zapis/Odczyt

Min.: 1 Maks.: 40 (dla modelu OCS 200/100 w momencie drukowania)

Rejestr ten pozwala na wyświetlenie lub zmodyfikowanie wartości określającej kontrast wyświetlacza ciekłokrystalicznego. Ma on zastosowanie wyłącznie w odniesieniu do sterowników z wyświetlaczami LCD.

**SR33**

Tryb przełączania klawiszem

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Min.: 0 Maks.: 1 (dla modelu OCS 200/100 w momencie drukowania)

Rejestr ten podaje lub ustawia tryb pracy klawiatury. Wartość 0 oznacza tryb chwilowy. Wartość 1 informuje o trybie trwały (toggle). Rejestr ten ma zastosowanie wyłącznie w odniesieniu do sterowników z klawiaturą.

**SR34**

Protokół szeregowy

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o aktualnie wybranym protokole dla PORTU 1 sterownika.

- 0 Aktualizacja oprogramowania systemowego
- 1 CsCAN
- 2 Protokół serial I/O
- 3 Modbus RTU
- 4 Modbus ASCII

**SR35 + SR36**

Bajt młodszy i starszy numeru seryjnego

Język drabinkowy: Odczyt

Tekst: Odczyt

Te 32-bitowe rejestry informują o elektronicznym numerze seryjnym sterownika Numer ten jest inny od numeru podanego na opakowaniu i etykiecie przyklejonej do sterownika

**SR37**

Numer modelu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten podaje binarny numer modelu sterownika. Przykładowo, sterowniki OCS100, OCS200 czy RCS210 posiadają różne numery binarne.

**SR38**

Wersja oprogramowania systemowego

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wersji oprogramowania systemowego. Przyjmuje się, że kropka znajduje się po drugiej cyfrze (12345 = 123.45).

**SR39**

Wersja BIOS

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o wersji BIOS-u oprogramowania systemowego. Przyjmuje się, że kropka znajduje się po drugiej cyfrze (12345 = 123.45).

**SR40**

Wersja FPGA

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o numerze wersji FPGA (dodatkowe urządzenia programowane z poziomu oprogramowania, występujące w większości sterowników). Przyjmuje się, że kropka znajduje się po drugiej cyfrze (12345 = 123.45).

**SR41**

Liczba kolumn wyświetlacza LCD

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o liczbie kolumn wyświetlacza tekstowego LCD lub wyświetlacza wirtualnego.

**SR42**

Liczba wierszy wyświetlacza LCD

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o liczbie wierszy wyświetlacza tekstowego LCD lub wyświetlacza wirtualnego.

**SR43**

Typ klawiatury

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o typie klawiatury.

**SR44**

Sekundy zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 0 Maks.: 59

Rejestr ten podaje sekundy zegara czasu rzeczywistego.

**SR45**

Minuty zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 0 Maks.: 59

Rejestr ten podaje minuty zegara czasu rzeczywistego.

**SR46**

Godziny zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 0 Maks.: 23

Rejestr ten podaje godziny zegara czasu rzeczywistego.

**SR47**

Dzień miesiąca zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 1 Maks.: 31

Rejestr ten podaje dzień miesiąca zegara czasu rzeczywistego.

#### **SR48**

Miesiąc zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 1 Maks.: 12

Rejestr ten podaje miesiąc zegara czasu rzeczywistego: 1 = styczeń ... 12 = grudzień

#### **SR49**

Rok zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 1996 Maks.: 2095

Rejestr ten podaje rok zegara czasu rzeczywistego, w formacie czterocyfrowym. Jest to rozwiązanie kompatybilne z Problemem Roku 2000.

#### **SR50**

Dzień tygodnia zegara czasu rzeczywistego

Język drabinkowy: Odczyt

Tekst: Odczyt

Min.: 1 Maks.: 7

Rejestr ten podaje dzień tygodnia zegara czasu rzeczywistego: 1= niedziela, 2 = poniedziałek 7 = niedziela

#### **SR51**

Licznik błędów sieciowych

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten informuje o liczbie zarejestrowanych błędów sieciowych.

**SR52 do SR55**  
ZAREZERWOWANE**SR56**  
Ostatni klawiszJęzyk drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten informuje, który klawisz został ostatnio wciśnięty na klawiaturze. Zamieszczona poniżej tabela zawiera kody generowane w czasie wciskania różnych klawiszy.

Uwaga: Nie wszystkie klawiatury posiadają wymienione poniżej klawisze.

Klawisz	Kod
Nie wciśnięty żaden klawisz	0
F1	1
F2	2
F3	3
F4	4
F5	5
F6	6
F7	7
F8	8
F9	9
F10	10
F11	11
F12	12
Enter	13
+/-	14
. (kropka)	15
0	16
1	17
2	18
3	19
4	20
5	21
6	22
7	23
8	24
9	25
System	26
Escape	27

W lewo	28
W prawo	29
W górę	30
W dół	31
Shift	32
Klawisz programowalny 1	34
Klawisz programowalny 2	35
Klawisz programowalny 3	36
Klawisz programowalny 4	37
Klawisz programowalny 5	38
Klawisz programowalny 6	39
Klawisz programowalny 7	40
Klawisz programowalny 8	41
Klawisz nie wciśnięty	255

**SR57**

Podświetlenie wyświetlacza LCD

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Rejestr ten podaje/steruje podświetlaniem wyświetlacza ciekłokrystalicznego:

- Dla sterowników OCS250 i OCS3xx 0 = Podświetlenie wyłączone wartość różna od zera = Podświetlenie włączone
- Dla pozostałych sterowników 0 = podświetlenie wyłączone 100 = pełne podświetlenie

**SR58**

Diody użytkownika

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Rejestr ten steruje diodami LED klawiatury w modelu OCS250. Ustawienie pierwszego bitu powoduje włączenie diody umieszczonej poniżej klawisza F1, ustawienie drugiego bitu powoduje włączenie diody umieszczonej pod klawiszem F2.

**SR59**

Numer build kontrolera

Język drabinkowy: Odczyt

Tekst: Odczyt



Rejestr zawiera numer buildu kontrolera. W rejestrze przechowywane są ostatnie trzy wartości dziesiętne. Np. Jeśli firmware sterownika jest 12.75.762 wówczas w %SR 59 jest wartość "762". %SR 59 może być wykorzystywane tylko w sterownikach o 12.75.762 lub wyższym

**SR60**

Typ Buildu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr zawiera typ buildu

- Build Test = 0
- Build Beta = 1
- Build Product = 2

%SR 60 może być wykorzystywane tylko w sterownikach o 12.75.762 lub wyższym

**SR61**

Ilość numerów ID

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje liczbę interfejsów CsCAN ID dostępnych w urządzeniu

**SR62**

Rejestr testu portów

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Rejestr ten może być wykorzystany do diagnostyki portu szeregowego oraz jego funkcjonalności.

**SR63 do SR135**

Zarezerwowane

**SR136 - SR137**

Timeout komunikacji

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten przechowuje wartość timeout dla komunikacji. Wartość w rejestrze zmienia się w zależności od skonfigurowanego Time Out. Np. Dla komunikacji w której timeout ustawiony jest na 1000 mS, %SR136 pokazuje wartość +19500 a %SR137 pokazuje +500.

**SR138 to SR163**

Zarezerwowane

**SR164**

Fail Safe &amp; Clone

- %SR164.3 Wykonanie automatycznego przywracanie - Tylko do odczytu
- %SR164.4 Wykonanie zadania Backupu rejestrów - Tylko do odczytu
- %SR164.5 Ustawienie opcji AutoRun jako „Tak”(1) lub Nie „0” – Odczyt/zapis
- %SR164.6 Ustawienie opcji AutoLoad jako „Tak”(1) lub Nie „0” – Odczyt/zapis
- %SR164.7 Bit wyzwalający robienie Backupu wszystkich danych rejestrowych – Odczyt/zapis
- %SR164.8 Bit zerujący robienie Backupu wszystkich danych rejestrowych (jeśli poprzednio wywołano %SR164.7) – Odczyt/zapis
- %SR164.9 Bit wyzwalający wykonanie MAKE\_CLONE (jeśli karat pamięci jest zainstalowana w sterowniku) – Odczyt/zapis
- %SR164.10 Bit wyzwalający wykonanie LOAD\_CLONE (jeśli karat pamięci jest zainstalowana w sterowniku) – Odczyt/zapis
- %SR164.11 Błąd wykonania operacji MAKE\_CLONE – Odczyt/zapis
- %SR164.12 Błąd wykonania operacji LOAD\_CLONE – Odczyt/zapis

**SR165 to SR166**

Zarezerwowane

**SR167**

Czas odświeżania ekranu

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Rejestr ten wykorzystywany jest do modyfikacji czasu odświeżania ekranu w ms. Wartość minimalna wynosi 2ms i maksymalna wartość 50 ms

**SR168 to SR174**

Zarezerwowane

**SR175**Status karty CF

---

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten pokazuje aktualny status karty pamięci zainstalowanej w sterowniku.

#### **SR176 do SR177**

Ilość wolnej pamięci karty CF

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten pokazuje ilość w bajtach wolnego miejsca na włożonej karcie CF. W sterownikach ZX oraz XL4e rejestr pokazuje ilość w kilobajtów.

#### **SR178 to SR179**

Całkowita pojemność karty CF

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten pokazuje w bajtach całkowitą ilość miejsca na włożonej karcie CF. W sterownikach ZX oraz XL4e rejestr pokazuje ilość w kilobajtów.

#### **SR181**

Alarmy niepotwierdzone

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten jest wskaźnikiem dla zaawansowanego menedżera alarmów w sterownikach Horner. Każdy bit słowa wskazuje, czy dana grupa ma niepotwierdzony alarm. Na przykład, jeżeli bit 1 jest ON oznacza, że w sterowniku niepotwierdzony jest alarm z grupy pierwszej.

#### **SR182**

Aktywne alarmy

Język drabinkowy: Odczyt  
Tekst: Odczyt

Rejestr ten jest wskaźnikiem dla zaawansowanego menedżera alarmów w sterownikach Horner. Każdy bit słowa wskazuje, czy dana grupa ma aktywny alarm. Na przykład, jeżeli bit 1 jest ON oznacza, że w sterowniku aktywny jest alarm z grupy pierwszej.

#### **SR183**

---

Brzęczek systemowy

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje, czy brzęczyk systemowy jest włączony.

#### **SR184**

Brzęczek użytkownika

Język drabinkowy: Zapis/Odczyt

Tekst: Zapis/Odczyt

Rejestr ten uruchamia brzęczek użytkownika.

- 0 = Beeper OFF
- 1 = Beeper ON

#### **SR185**

Wygaszacz ekranu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten pokazuje czy wygaszacz ekranu w sterowniku jest aktywny

0 = wygaszacz ekranu jest nieaktywny

1 = wygaszacz ekranu jest aktywny

#### **SR186**

Opóźnienie wygaszacza ekranu

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten określa limit czasu dla wygaszacza ekranu w minutach. Jeśli wygaszacz ekranu nie jest włączony (Zobacz SR185), rejestr ten nie jest używany.

#### **SR187**

Średnie wykorzystanie sieci

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje średnie całkowite wykorzystanie sieci CAN. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

**SR188**

Minimalne wykorzystanie sieci

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje minimalne całkowite wykorzystanie sieci CAN. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

**SR189**

Maksymalne wykorzystanie sieci

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje maksymalne całkowite wykorzystanie sieci CAN. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

**SR190**

Średnia ilość danych wysyłanych na sieć

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje średnie wykorzystanie sieci CAN podczas transmisji dla tego urządzenia transmitowane przez to urządzenie. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

**SR191**

Minimalna ilość danych wysyłanych na sieć

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje minimalne wykorzystanie sieci CAN podczas transmisji dla tego urządzenia transmitowane przez to urządzenie. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

**SR192**

---

Maksymalna ilość danych wysyłanych na sieć

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje maksymalne wykorzystanie sieci CAN podczas transmisji dla tego urządzenia transmitowane przez to urządzenie. Wartość podana jest w dziesiątych częściach procenta. Np.: 25 oznacza całkowite wykorzystanie przepustowości sieci na poziomie 2.5%

### **SR193**

Zmiany na ruchu

- %SR193.1 Wartość "1" jeśli są dwa programy zapisane w urządzeniu docelowym
- %SR193.2 Wyzwolenie przełączenia programu z aktualnie wykonywanego na program wgrany na ruchu. Sterownik zeruje bit jeśli procedura zakończyła się powodzeniem
- %SR193.3 Wartość "1" jeśli aktualnie wykonywany program to program wgrany na ruchu (który został załadowany przez on-line programming)
- %SR193.4 Wartość "1" podczas ostatniego cyklu sterownika w którym wykonywany jest stary program
- %SR193.5 Wartość "1" podczas pierwszego cyklu sterownika w którym wykonywany jest nowy program wgrany na ruchu
- %SR193.6 Wyzwolenie powrotu do poprzedniej wersji program (zapisanej w pamięci Flash) oraz wykasowanie programu wgranego na ruchu z pamięci RAM. Sterownik zeruje bit jeśli procedura zakończyła się powodzeniem
- %SR193.9 Wartość "1" jeśli w programie wgranym na ruchu występuje błąd.

### **SR200**

Wersja InitRD

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje wersję initrd dla modeli ZX.

Np: %SR200 = 101 oznacza wersję initrd 1.01

### **SR201 - 205**

Wersja Kernel

Język drabinkowy: Odczyt

Tekst: Odczyt

Rejestr ten wskazuje wersję Kernel dla modeli ZX.

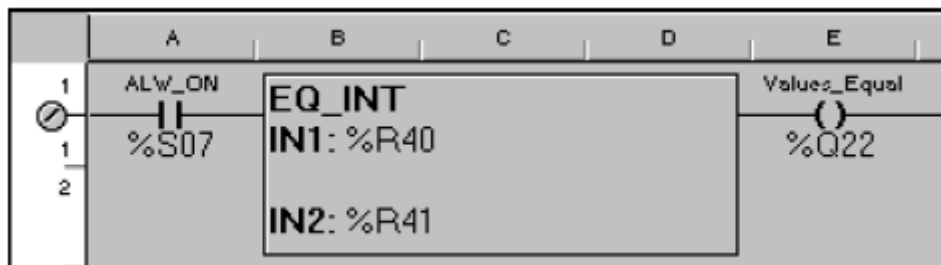
### Predefiniowane zmienne dwustawowe

Pewne punkty wejść/wyjść (adresy) w pamięci zostały predefiniowane. Mogą one być bezpośrednio wykorzystywane przez program sterujący.

Bit	Nazwa	Funkcja
%S01	FST_SCN	Sygnalizuje pierwszy cykl pracy sterownika
%S02	NET_OK	Sieć działa prawidłowo
%S03	T_10MS	Podstawa generatora czasu 10ms
%S04	T_100MS	Podstawa generatora czasu 100 ms
%S05	T_1SEC	Podstawa generatora 1 sekunda
%S06	IO_OK	Wejścia/wyjścia OK.
%S07	ALW_ON	Styk zawsze zwarty
%S08	ALW_OFF	Styk zawsze rozwartry
%S09	PAUSING_SCN	
%S10	RESUMED_SCN	
%S11	FORCE	Wejścia/wyjścia są sforcowne
%S12	FORCE_EN	Forsownie zmiennych jest aktywne
%K01	F1_Key	Klawisz funkcyjny 1
%K02	F2_Key	Klawisz funkcyjny 2
%K03	F3_Key	Klawisz funkcyjny 3
%K04	F4_Key	Klawisz funkcyjny 4
%K05	F5_Key	Klawisz funkcyjny 5
.	.	.
.	.	.
%K0x	Fx_Key	Klawisz funkcyjny x

Przykład 1:


W programie można wywoływać w każdym cyklu pracy jednostki centralnej wiele bloków funkcyjnych, bez względu na stan wejść. W tym celu należy skorzystać z punktu ALW\_ON.



Przykład korzystania z punktu ALW\_ON

Przykład 2:

Klawisze funkcyjne pozwalają na łatwe wprowadzanie danych przez operatora. W zamieszczonym poniżej programie, każde wciśnięcie klawisza funkcyjnego 4 powoduje wyświetlenie strony.

	A	B	C	D	E
1	F4_KEY	( ) Screen: 4 Function Key 4 is pressed			
1					
2	%K04				

*Przykład obsługi klawisza funkcyjnego 4*



## Schematy połączeń i rozmieszczenie styków

W rozdziale tym zamieszczono schematy połączeń oraz opisano rozmieszczenie styków w sterownikach OCS i współpracujących z nimi urządzeniach.

### Złącza zasilania

---

#### Złącze zasilania OCS

Tego typu złącze zasilania stosowane jest w sterownikach OCS100, OCS200 i OCS250. Masę ochronną (PF) należy podłączyć do metalowej obudowy sterownika.



*Złącze zasilające OCS*

#### Złącze zasilania MiniOCS

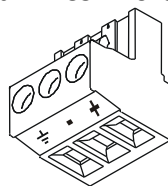
Złącze zasilania tego typu stosowane jest w sterownikach MiniOCS.



*Złącze zasilania MiniOCS*

#### Złącze zasilania RCS i serii XL

Złącze zasilania tego typu stosowane jest w RCS21 oraz wszystkich sterownikach serii NX i XL



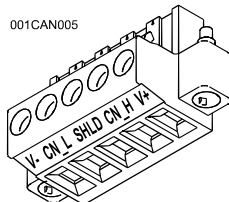
*Złącze zasilania RCS*

### Złącza sieciowe

---

#### Złącze sieciowe CAN

Złącze tego typu stosowane jest przy podłączaniu do szybkiej sieci CsCAN, DeviceNet, J1939 oraz CANopen.



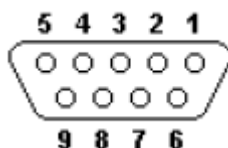
*Złącze sieciowe OCS (widok z tyłu sterownika)*

Jest to złącze żeńskie. Funkcje poszczególnych styków są następujące:

Numer styku	Funkcja
1	Napięcie – (zasilanie)
2	CAN_L
3	Ekran
4	CAN_H
5	Napięcie + (zasilanie)

### Port szeregowy/port do programowania

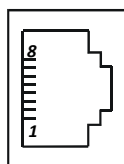
Port ten jest zwykle wykorzystywany do podłączenia komputera, na którym uruchomione jest oprogramowanie Cscape. Port ten może być również tymczasowo wykorzystywany jako port szeregowy z parametrami definiowanymi przez użytkownika z poziomu programu sterującego w języku drabinkowym. W trybie takim nie można jednak wykorzystywać tego portu do programowania za pomocą oprogramowania Cscape.



*Rozmieszczenie styków w porcie szeregowym OCS (widok z tyłu sterownika)*

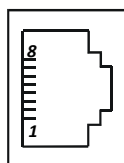
Jest to złącze żeńskie. Funkcje poszczególnych styków są następujące:

Numer styku	Funkcja	Kierunek przesyłania względem OCS	Opis
1	DCD	Wyjściowy	Data Carrier Detect (zawsze ON)
2	TXD	Wejściowy	Receive Data
3	RXD	Wyjściowy	Transmit Data
4	DSR	Wejściowy	Data Terminal Ready
5	GND	-	masa
6	DTR	Wyjściowy	Data Set Ready
7	CTS	Wyjściowy	Request to Send
8	RTS	Wejściowy	Clear to Send
9	RI	Wejściowy	Ring Indicator (zawsze OFF)



*MJ1 - Rozmieszczenie styków w porcie szeregowym sterowników serii XL  
(widok z tyłu sterownika)*

Numer styku	Funkcja	Kierunek przesyłania względem OCS	Opis
1	RX+ / TX+	Wejściowy / Wyjściowy	Transmit Data / Receive Data
2	RX- / TX-	Wejściowy / Wyjściowy	Transmit Data / Receive Data
3	CTS	Wyjściowy	Request to Send
4	RTS	Wejściowy	Clear to Send
5	+5V 60mA	Wyjściowy	napięcie 5VDC
6	0V	-	masa
7	RXD	Wyjściowy	Transmit Data
8	TXD	Wejściowy	Receive Data



*MJ2 - Rozmieszczenie styków w porcie szeregowym sterowników serii XL  
(widok z tyłu sterownika)*

Numer styku	Funkcja	Kierunek przesyłania względem OCS	Opis
1	RX+	Wejściowy	Receive Data +
2	RX-	Wejściowy	Receive Data -
3	TX+	Wyjściowy	Transmit Data +
4	TX-	Wyjściowy	Transmit Data -
5	+5V 60mA	Wyjściowy	napięcie 5VDC
6	0V	-	masa
7	RXD	Wyjściowy	Transmit Data
8	TXD	Wejściowy	Receive Data

## Liczby zmiennoprzecinkowe

Liczby rzeczywiste, zwane również zmiennoprzecinkowymi, to liczby zawierające symbol dziesiętny. Liczby te nazywane są "rzeczywistymi", ponieważ odwzorowują one rzeczywistą wartość wyniku pomiaru (dokładność systemu) w postaci wartości całkowitej i części ułamkowych, bez sztucznego obcinania do formatu o mniejszej precyzji, jak to ma miejsce w przypadku konwersji na liczby całkowite.

Położenie znaku części dziesiętnej (wyznaczającego części całkowite oraz ułamkowe) jest określone w samej liczbie. Ponieważ znak części dziesiętnej może znajdować się w różnych miejscach liczby, są one z tego powodu często określane jako "zmiennoprzecinkowe". W Cscape terminy "liczba rzeczywista" i "liczba zmiennoprzecinkowa" są używane naprzemiennie.

## Format liczb rzeczywistych

Liczby rzeczywiste są zwykle wprowadzane i wyświetlane za pomocą sześciu cyfr.

```
3.12159
654321
```

Jeżeli wartość jest zbyt duża lub zbyt mała, aby ją można było przedstawić za pomocą sześciu cyfr, stosowany jest format w którym podawane jest sześć cyfr plus wykładnik.

```
1.03647e+12
9.73157e-22
```

Przy wyświetlaniu liczb rzeczywistych pokazywanych jest sześć cyfr ze znakiem części dziesiętnej oraz opcjonalnie wykładnik. Jeżeli wartość może być pokazana za pomocą sześciu lub mniej cyfr, wykładnik jest pomijany:

+3.14159  
-654321  
12  
.001357  
-.000032

Znak '+' lub '-' jest opcjonalny. Jeżeli znak ten nie jest podany, domyślnie przyjmuje się znak '+'. Liczby posiadające dużą liczbę miejsc dziesiętnych wyświetlane są w notacji naukowej. W formacie tym pokazywanych jest sześć cyfr ze znakiem części dziesiętnej plus wykładnik. Wykładnik składa się z literki 'E' lub 'e', znaku ('+' lub '-') oraz dwucyfrowej liczby. Przykładowo:

.0000000004567 = 4.567e-103  
143286945 = 3.14329e+09

Należy zwrócić uwagę, że w drugim przykładzie następuje utrata precyzji, ponieważ w liczbach rzeczywistych dostępne jest wyłącznie sześć cyfr znaczących.

Reprezentacja wewnętrzna liczb zmiennoprzecinkowych to format IEEE 32-bitowy, pojedynczej precyzji. W formacie tym wykorzystywana jest 23-bitowa mantysa, 8-bitowy wykładnik oraz jeden bit znaku.

Należy zwrócić uwagę, że każda liczba rzeczywista jest reprezentowana za pomocą 32 bitów. Sterownik OCS wymaga więc w tym celu dwóch (2) kolejnych rejestrów 16-bitowych, zwykle %R.

### Zakres wartości

32-bitowy format IEEE pojedynczej precyzji pozwala na reprezentowanie wartości od +/- 3.40282E+38 (bardzo mała wartość ułamkowa) do +/- 3.40282E+38 (bardzo duża wartość całkowita).

### Cyfry znaczące

W reprezentacji liczb rzeczywistych wykorzystywanych jest sześć (6) cyfr znaczących. Jeżeli wyświetlanych jest więcej niż sześć (6) cyfr, wyłącznie pierwszych sześć może być brane pod uwagę przy wyznaczaniu dokładności.

3.14159265 = 3.14159  
2535.45 = 2535

### Wprowadzanie wartości rzeczywistych

Format ten musi być stosowany dla wszystkich liczb rzeczywistych.

W przypadku podawania wykładnika, mantysa musi zawierać znak części dziesiętnej. Należy zwrócić uwagę, że jeżeli stosowany jest format różny od formatu x.yyy, następuje automatyczne przesunięcie znaku części dziesiętnej oraz obliczenie odpowiedniego wykładnika.

$123.456e+3 = 123456$  [Wartość może być wyświetlona za pomocą sześciu cyfr, bez wykładnika]

$143.643E-12 = 1.43643E-10$  [Przesunięcie znaku części dziesiętnej i obliczenie odpowiedniej wartości wykładnika]

Należy zawsze podawać znak części dziesiętnej, w celu uniknięcia niejednoznaczności w interpretowaniu wartości. Przykładowo, wartość 123e10 można wprowadzić jako 123.0e10 lub lepiej jako 1.23E12 (Cscape automatycznie przekonwertuje wartość do tego formatu).

Zarówno mantysa, jak i wykładnik nie mogą zawierać spacji. Wartości "123 45e-12" czy "4.3256e - 23" nie zostaną poprawnie zinterpretowane, ponieważ zawierają spację.

Zarówno mantysa jak i wykładnik może zawierać znak, '+' lub '-' np. "-1.3245e+12" lub "4.243e-8". Jeżeli znak nie jest podany, przyjmuje się że jest to znak dodatni "1.2345e10".

### **Błędy**

Najczęściej występujący błąd to PRZEKROCZENIE ZAKRESU. Błąd ten występuje jeżeli wynik operacji na liczbach rzeczywistych jest większy od +3.40282E+38 lub mniejszy od -3.40282E-38. Przykładowo, błąd ten wystąpi przy obliczaniu wyniku działania  $1.2345E-20 * 2.3456E-20$ .

### **Nieskończoność**

W przypadku wystąpienia błędu przekroczenia zakresu, sygnał nie przepływa przez blok funkcyjny, a wynik ma wartość nieskończenie dużą dodatnią (jeżeli wartość jest większa od +3.40282E+38) lub nieskończenie dużą ujemną (jeżeli wynik jest mniejszy od -3.40282E+38).

### **Wartość nieokreślona**

Jeżeli wartość nieskończenie duża zostanie podana jako argument następnego działania matematycznego, wynik tego działania będzie nieokreślony. Wynik taki określany jest terminem NaN (Not a Number - Nie liczba).

W przypadku wyniku równego NAN, sygnał nie przepływa przez dany blok funkcyjny.

Jeżeli wartość NAN zostanie wprowadzona jako parametr następnego bloku funkcyjnego, jest ona przesyłana do tego bloku funkcyjnego.

## Moduł SmartStack STP 100

### Informacje ogólne

**Uwaga:**

Rozdział ten zawiera informacje ogólne. Szczegółowe informacje podane są w Instrukcji obsługi (SUP0270), dostarczanej wraz z tym modułem. Zawiera ona szczegółowy opis korzystania i programowania tego modułu.

Moduł SmartStack STP100 pozwala na Absolutne, Względne i Indeksowane (Indexed Move) sterowanie ruchami pojedynczej osi.

Moduł ten posiada szesnaście (16) wejść cyfrowych (%I), szesnaście (16) wyjść cyfrowych (%Q), cztery (4) wejścia analogowe (%AI) oraz siedem (7) lub czternaście (14) wyjść analogowych (%AQ), w zależności od trybu pracy.

W celu ułatwienia programowania modułu STP100, oprogramowanie Cscape zawiera specjalnych blok funkcyjny Stepper Move. Blok ten przesyła stałe lub wartości zmiennych do rejestrów %AQ modułu STP100.

### Bity poleceń

Szesnaście (16) wyjść cyfrowych (%Q) wykorzystywanych jest jako bity poleceń.

Punkt	Opis
%Q1	Reserved (Zarezerwowany)
%Q2	Reserved (Zarezerwowany)
%Q3	Reserved (Zarezerwowany)
%Q4	Find Origin Up
%Q5	Find Origin Down
%Q6	Jog Up (Przesunięcie w górę)
%Q7	Jog Down (Przesunięcie w dół)
%Q8	Move Relative
%Q9	Move Absolute
%Q10	Resume Move
%Q11	Move Indexed
%Q12	Reserved (Zarezerwowany)
%Q13	Set Current Position (Ustawienie bieżącej pozycji)
%Q14	Clear Error(s) (Kasowanie błędów)
%Q15	Decelerate and Stop (Hamowanie i zatrzymanie)
%Q16	Immediate Stop (Natychmiastowe zatrzymanie)

Jednocześnie może być ustawiony tylko jeden bit. Jeżeli jednocześnie jest ustawiony więcej niż jeden bit, większy priorytet ma bit o wyższym numerze. Należy zwrócić uwagę, że największy priorytet ma bit NATYCHMIASTOWE ZATRZYMANIE.

Natychmiast po włączeniu zasilania, ustawiany jest bit statusu Błąd - włączenia zasilania/resetowanie. Pierwszym wysyłanym poleceniem musi być polecenie KASOWANIE BŁĘDÓW.

W przypadku ustawienia dowolnego z bitów błędów, wszystkie wysyłane polecenia są ignorowane. Wszystkie bity poleceń reagują na pozytywną zmianę sygnału (z OFF na ON). Polecenia JOG UP i JOG DOWN reagują również na negatywną zmianę sygnału (z ON na OFF), ponieważ wymagają one zarówno sygnału początku jak i końca.

### Uwaga:

Przed wysyłaniem jakichkolwiek poleceń musi być wysłane polecenie KASOWANIE BŁĘDÓW. Jest to mechanizm zaimplementowany w celu zwiększenia bezpieczeństwa.

Nie wszystkie polecenia są przez cały czas dostępne. Przykładowo, jeżeli realizowane jest polecenie nakazujące ruch, można wysyłać wyłącznie polecenia ZAHAMUJ I ZATRZYMAJ lub NATYCHMIASTOWE ZATRZYMANIE.

## Bity statusu

Szesnaście (16) wejść cyfrowych (%I) wykorzystywanych jest jako bity statusu.

Punkt	Opis
%I1	Błąd - awaryjne zatrzymanie
%I2	Błąd - zatrzymanie z powodu przekroczenia dolnej wartości granicznej
%I3	Błąd - zatrzymanie z powodu przekroczenia górnej wartości granicznej
%I4	Błąd - niedopuszczalny ruch
%I5	Błąd - zatrzymanie silnika
%I6	Zarezerwowany do wykorzystania w przyszłości
%I7	Zarezerwowany do wykorzystania w przyszłości
%I8	Błąd - włączenie zasilania/resetowanie
%I9	Możliwość wznowienia pierwotnego ruchu
%I10	Bieżąca pozycja poprawna
%I11	Zarezerwowany do wykorzystania w przyszłości
%I12	Zarezerwowany do wykorzystania w przyszłości
%I13	Pozycja bazowa
%I14	Przyśpieszenie
%I15	Hamowanie
%I16	Ruch



Bit 1 do 8 to bity informujące o błędach. Ustawienie wartości tych bitów sygnalizuje wystąpienie błędu.

W przypadku wystąpienia jednego lub więcej błędów, moduł nie będzie pracował do momentu ich wykasowania. Natychmiast po włączeniu zasilania lub przekroczeniu czasu zegara wyłączającego ustawiany jest bit 8 - Błąd - włączenie zasilania/resetowanie, co zapobiega pracy modułu, do momentu wysłania polecenia KASOWANIE BŁĘDÓW.

Z tego powodu, pierwszym wysylnym poleceniem musi być polecenie KASOWANIE BŁĘDÓW. W przypadku ustawienia dowolnego z bitów błędów, wszystkie wysyłane polecenia są ignorowane.

Bit 0 do 16 to bity statusu. Status tych bitów (0 lub 1) informuje o stanie modułu. NIE są to błędy, a więc w przypadku ustawienia dowolnego z tych bitów moduł nadal normalnie pracuje. Bity te NIE SĄ zerowane po wysłaniu polecenia KASOWANIE BŁĘDÓW.

## Rejestry sprzężenia zwrotnego położeniowego

Cztery (4) wejścia analogowe (%AI) są wykorzystywane jako dwa (2) rejestry DINT (32-bitowe). Pierwsze dwa punkty są łączone w jeden rejestr 32-bitowy, a dwa następne punkty w drugi rejestr 32-bitowy.

### Uwaga:

W oprogramowaniu Cscape, te pary rejestrów są traktowane jako typu DINT.

Adres	Opis	Przedział
%AI1	Młodszy bajt pozycji silnika	-8,388,608 - +8,388,607
%AI2	Starszy bajt pozycji silnika	
%AI3	Młodszy bajt pozycji dekodera	-8,388,608 - +8,388,607
%AI4	Starszy bajt pozycji dekodera	

Natychmiast po włączeniu zasilania sterownika do rejestrów tych zapisywana jest wartość (zero) 0, co jest sygnalizowane jako błąd poprzez wyzerowanie Bitu statusu - Bieżąca pozycja poprawna.

Wartość - Pozycja silnika jest sygnalizowana jako błędna do momentu wysłania polecenia FIND HOME lub SET CURRENT POSITION.

Jeżeli bit statusu - Bieżąca pozycja poprawna ma wartość zero, nie jest akceptowane polecenie Ruch absolutny. Oprogramowanie Cscape – Instrukcja obsługi i programowania

## Dane wykorzystywane przez polecenia

Dane wykorzystywane przez polecenia zawarte są w następujących rejestrach.

Punkt	Typ danych	Opis	Wartość
%AQ1	32 bity	Młodszy bajt pozycji docelowej	-8,388,608 - +8,388,607
%AQ2		Starszy bajt pozycji docelowej	
%AQ3	16 bitów	Dzielnik prędkości	20 - 65,535
%AQ4	16 bitów	Prędkość bazowa	1 - 8,190
%AQ5	16 bitów	Prędkość robocza	2 - 8,191
%AQ6	16 bitów	Czas przyspieszania (ms)	1 - 27,300
%AQ7	16 bitów	Czas hamowania (ms)	0 - 27,300

Pierwsze dwa punkty są łączone w pojedynczy, 32-bitowy rejestr. Zawiera on pozycję, na której ma się zatrzymać silnik krokowy. W zależności od rodzaju wysłanej instrukcji, pozycja ta może być pozycją bezwzględną, liczoną od pozycji bazowej lub pozycją względną, obliczaną w stosunku do aktualnej pozycji.

- Rejestr **Dzielnik prędkości** określa rozdzielczość Prędkości bazowej i Prędkości roboczej. Zagadnienie to omówiono szczegółowo w instrukcji obsługi modułu STP100.
- Rejestr **Prędkość bazowa** określa pierwszą prędkość, wykorzystywaną w czasie rozpoczynania ruchu oraz ostatnią prędkość w czasie kończenia ruchu.
- Rejestr **Prędkość robocza** jest maksymalną prędkością ruchu, dopuszczalną chwilowo.

W trakcie normalnej pracy, ruch jest rozpoczynany z **Prędkością bazową**, po czym następuje przyspieszenie do Prędkości roboczej, hamowanie do Prędkości bazowej i zatrzymanie. Bity statusu Przyspieszanie, Hamowanie i Ruch odzwierciedlają bieżący status ruchu.

- **Czas przyspieszania** to okres czasu przeznaczony na przyspieszenie od Prędkości bazowej do Prędkości roboczej.
- **Czas hamowania** to okres czasu przeznaczony na hamowanie od Prędkości roboczej do Prędkości bazowej. Jeżeli wpisana zostanie wartość (zero) 0, moduł automatycznie korzysta z parametru Czas przyspieszania.

## Funkcja Indexed Move

Moduł STP100 pozwala na korzystanie z funkcji Indexed Move. W tym celu moduł SmartStock należy skonfigurować do korzystania z zewnętrznego parametru Index Input i odpowiednio skonfigurować blok funkcyjny Stepper Move.

### Uwaga:

Przy korzystaniu z funkcji Indexed Move wszystkie ruchy są ruchami względnymi.

Blok funkcyjny Stepper Move korzysta wtedy z siedmiu (7) dodatkowych rejestrów, sześć sposobów nich jest łączonych w celu otrzymania trzech (3) rejestrów 32-bitowych, a jeden (1) jest wykorzystywany jako 16-bitowy rejestr bez znaku.

Punkt	Typ danych	Opis	Wartość
%AQ8	32 bity	Młodszy bajt pozycji docelowej	1 – 16,777,215
%AQ9		Starszy bajt pozycji docelowej	
%AQ10	16 bitów	Czas przyspieszenia	0 - 27,300
%AQ11	32 bity	Młodszy bajt pozycji początkowej	1 – 16,777,215
%AQ12		Młodszy bajt pozycji początkowej	
%AQ13	32 bity	Młodszy bajt pozycji końcowej	1 – 16,777,215
%AQ14		Młodszy bajt pozycji końcowej	

Blok funkcyjny Stepper Move korzysta z zewnętrznego wejścia o nazwie INDEX. Wejście to powinno być połączone z przełącznikiem lub innym elektromechanicznym urządzeniem (optycznym, magnetycznym, itp.). Wejście to jest aktywne w stanie niskim. Jeżeli moduł stwierdzi, że wejście INDEX ma stan niski w oknie, następuje przemieszczenie silnika do innej pozycji.

Okno jest definiowane poprzez podanie Indexed Window Begin Position i Indexed Window End Position. Wejście INDEX jest akceptowane wyłącznie pod warunkiem, że Sterownik silnika krokowego znajduje się w dopuszczalnym zakresie.

### Uwaga:

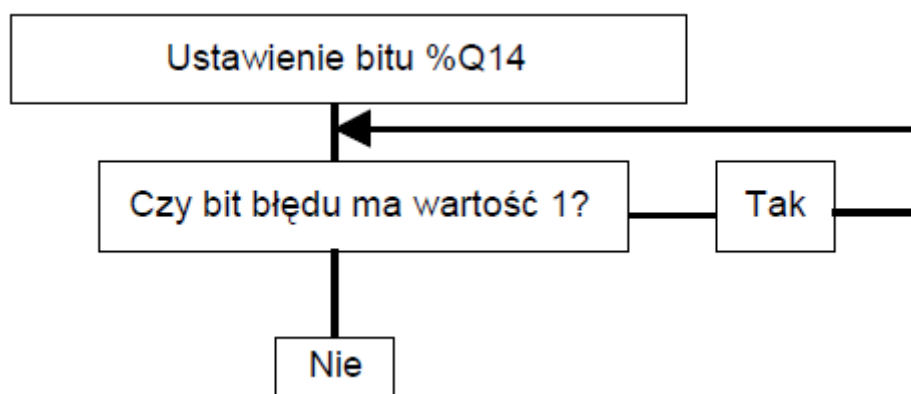
Okres okna jest dodatkowo ograniczony do czasu, w ciągu którego silnik krokowy musi uzyskać Prędkość roboczą. Jeżeli okno jest tak zdefiniowane, że następuje próba otwarcia okna w czasie przyspieszania, okno nie jest otwierane do momentu uzyskania Prędkości roboczej. Okno jest również automatycznie zamykane, w momencie rozpoczęcia hamowania. Jeżeli silnik krokowy nie jest w stanie dojść do Prędkości roboczej, Okno indeksu nie jest otwierane.

Jeżeli sygnał INDEX zostanie doprowadzony w czasie wyświetlania okna, sterownik silnika krokowego zmienia pozycję docelową ruchu na określoną przez Indexed Destination Position (%AQ8/%AQ9) liczoną względem Bieżącej pozycji silnika (%AI1, %AI2) w momencie przełączenia sygnału INDEX w stan aktywny. Czas hamowania jest określany przez rejestr Indexed Deceleration Time.

## Wysyłanie poleceń

Pierwszy krok polega na sprawdzeniu, czy nie ma błędów. Natychmiast po włączeniu zasilania lub zresetowaniu ustawiany jest bit Błąd - Włączenie zasilania/resetowanie, a więc najpierw należy wysłać polecenie KASOWANIE BŁĘDÓW.

Uproszczony sposób wykonywania polecenia KASOWANIE BŁĘDÓW przedstawiono na rysunku poniżej.



*Sposób realizacji polecenia KASOWANIE BŁĘDÓW*

Po włączeniu zasilania następuje wyzerowanie rejestrów (%AI1 - %AI4) oraz bitu statusu Bieżąca pozycja poprawna (%I9). Jeżeli bit %I9 ma wartość zero (0), nie jest akceptowane polecenie Absolute Move, ponieważ rzeczywista pozycja absolutna nie jest znana. W celu ustalenia tej pozycji należy wysłać polecenie FIND Origin UP, FIND Origin DOWN lub SET CURRENT POSITION FIND Origin UP, FIND Origin DOWN, lub SET CURRENT POSITION.

Bit Bieżąca pozycja poprawna jest również ustawiany na zero (0) w przypadku nagłego zatrzymania silnika. Przyczyną może być zatrzymanie awaryjne, zatrzymanie z powodu przekroczenia dolnej wartości granicznej, zatrzymanie z powodu przekroczenia górnej wartości granicznej, zatrzymanie silnika lub wysłanie polecenia Natychmiastowe zatrzymanie.

Pozostałe polecenia wysyłane są w podobny sposób.

- W przypadku występowania błędów należy usunąć przyczyny ich powstania, po czym wysłać polecenie KASOWANIE BŁĘDÓW.
- Ustawić parametry bloku funkcyjnego Stepper Move, a następnie doprowadzić sygnał do tego bloku.

- Ustawić odpowiedni bit polecenia na 1.
- Sprawdzić odpowiedni bit statusu dla polecenia.
- Nie wysyłać następnego polecenia do momentu pomyślnego zakończenia wykonywania poprzedniego polecenia lub wystąpienia błędów.

## Przetworniki analogowo-cyfrowe

### Wprowadzenie

---

Wiele programów sterujących musi realizować bardziej złożone operacje niż proste sterowanie z wykorzystaniem zmiennych typu binarnego. Programy takie często korzystają z temperatury, natężenia przepływu, stanu napełnienia czy innych wartości ciągłych, zmieniających się w pewnych granicach. Wielkości takie są mierzone przy użyciu czujników, a następnie zamieniane na napięcie lub natężenie prądu. Napięcie lub natężenie prądu odzwierciedla mierzoną wartość.

Procesory urządzeń cyfrowych (np. sterowników Horner APG lub komputerów osobistych) nie mogą bezpośrednio przetwarzać przykładowo napięcia. Urządzenia te mogą bezpośrednio rozpoznawać tylko dwa poziomy napięcia: 0 (zero) lub +V. Poziomy te mogą oczywiście zmieniać się w ustalonych granicach. Wartości analogowe napięcia lub prądu muszą zostać przetworzone przed doprowadzeniem do komputera, za pomocą przetwornika analogowo-cyfrowego. Przetwornik taki zamienia napięcie lub prąd na wartość binarną, która może być wysłana do urządzenia cyfrowego.

W przypadku sterowników Horner APG dostępne są dwa rodzaje modułów z przetwornikami: moduł wejść analogowych oraz moduł do czujników termoparowych/oporowych. Sposób działania modułu wejść analogowych jest zgodny z przedstawionym powyżej opisem. Do modułu tego doprowadzane jest napięcie lub prąd, reprezentujące wyniki pomiarów. Moduł ten następnie przekształca wartość analogową na wartość binarną, wykorzystywaną przez sterownik Horner APG. Wartość ta może reprezentować różne mierzalne wielkości - natężenie przepływu, ciężar, poziom, itp.

Moduł do czujników termoparowych/oporowych wykorzystywany jest przy pomiarach temperatury. Pomimo, że zasada działania tych modułów jest bardzo podobna, moduł do czujników termoparowych/oporowych podaje wynik w stopniach i dziesiątych częściach stopnia, a więc jest bardziej wyspecjalizowany.

### Konwersja wartości analogowych

---

Sygnal analogowy zmienia się w sposób ciągły pomiędzy dwoma wartościami. Ujmując to matematycznie, sygnał analogowy składa się z nieskończenie dużej liczby punktów dyskretnych pomiędzy wartościami A i B.

W celu wprowadzenia takiej wartości do komputera, sygnał analogowy musi być skwantowany do pewnej liczby dyskretnych poziomów. Liczba tych poziomów zależy od możliwości przetwornika analogowo- cyfrowego.

Istnieje szereg metod kwantowania sygnału analogowego na wartości dyskretne. Powszechnie stosowana jest metoda kolejnych przybliżeń, ponieważ jest ona kompromisowym rozwiązaniem pomiędzy prędkością a kosztem. Znacznie większą prędkość zapewniają przetworniki typu Flash, ale są one o wiele droższe. Przetworniki z podwójnym nachyleniem są bardzo dokładne, wolne i jednocześnie trochę droższe.

Bez względu na stosowaną metodę, przetwornik analogowo-cyfrowy zamienia sygnał analogowy na serie dyskretnych wartości. Wartość analogowa mieszcząca się w dopuszczalnym zakresie wartości jest zamieniana na pojedynczą wartość cyfrową. Przykładowo, przetwornik może być zaprojektowany tak, aby napięcie mieszczące się w przedziale 9.9975 V - 1.0025 V było zamieniane na wartość cyfrową odpowiadającą 1.000 V. Wartość analogowa z przedziału 9.9925 V - 9.9975 V jest przekształcana na wartość cyfrową odpowiadającą 9.995 V. Wartość cyfrowa reprezentująca napięcie jest doprowadzana do urządzenia cyfrowego (procesora sterownika Horner APG).

## Rozdzielczość

Liczba poziomów kwantyzacji zależy od rozdzielczości przetwornika. Rozdzielczość uzależniona jest od liczby bitów wykorzystywanych do reprezentowania wartości wyjściowej przetwornika. Powszechnie stosowane są przetworniki 10-, 12- i 14-bitowe. 12-bitowy przetwornik analogowo-cyfrowy daje na wyjściu 12-bitowe liczby, odczytywane przez sterownik Horner APG. Sterownik udostępnia je jako liczby 16-bitowe, (rozdzielczość pomiaru wynosi nadal 12-bitów).

Znając liczbę bitów rozdzielczości, można wyznaczyć liczbę poziomów kwantyzacji, korzystając z wzoru  $2$  do potęgi  $N$ , gdzie " $N$ " jest liczbą bitów rozdzielczości. Tak więc przetwornik 10-bitowy, posiada 1024 dyskretne poziomy, przetwornik 12-bitowy - 4096 poziomów, a przetwornik 14-bitowy - 16384 poziomy.

Należy zwrócić uwagę, że rozdzielczość nie określa dokładności. Dokładność zależy od sposobu w jaki przetwornik przekształca wartość analogową na wartość cyfrową. 8-bitowy przetwornik (256 dyskretne poziomy) może być znacznie bardziej dokładny niż źle zaprojektowany lub uszkodzony przetwornik 16-bitowy.

## Wielkość kroku kwantowania

Rozdzielczość informuje o wielkości kroku kwantowania. Inaczej ujmując informuje, jak mała zmiana sygnału analogowego może być zmierzona przez przetwornik. Wcześniej, na podstawie znanego zakresu zmian wielkości analogowej należy wybrać w module przetwornika odpowiedni ZAKRES. Należy wybrać ZAKRES obejmujący wszystkie możliwe wartości wejściowe, za wyjątkiem wartości generowanej po wystąpieniu błędu w mierzonym procesie.

W celu zapewnienia spójnych wyników pomiarów, większość modułu korzysta z jednego z następujących zakresów:

Przedział	Min. przedział:	Maks. przedział
0 V do +10 V	0 V	+ 10.24 V
+/-10 V	- 10.24 V	+ 10.24 V
0 V do +5 V	0 V	+ 5.12 V
+/-5 V	- 5.12 V	+ 5.12 V
0 do 20 mA	0 mA	+20.48 mA
+/-20 mA	-20.48 mA	+20.48 mA

Większość modułów przetworników ADC pozwala na skonfigurowanie dwóch lub więcej zakresów.

Po wyznaczeniu zakresu zmian wielkości analogowej, można obliczyć wielkość kroku kwantowania za pomocą przedstawionego poniżej, prostego równania matematycznego:

Wielkość kroku kwantowania = (Maksymalna wartość - Minimalna wartość) /  
Rozdzielczość

Najlepiej ilustruje to zagadnienie przykład. Dla przetwornika ADC o rozdzielczości 12-bitów i zakresu +/- 10 VDC:

Wielkość kroku kwantowania =  $(+10.24 - (-10.24)) / 4096$

Wielkość kroku kwantowania =  $20.48 / 4096$

Wielkość kroku kwantowania =  $0.005 \text{ V} = 5 \text{ mV}$

lub dla zakresu 0 do +5V:

Wielkość kroku kwantowania =  $(+5.12 - 0) / 4096$

Wielkość kroku kwantowania =  $5.12 / 4096$

Wielkość kroku kwantowania =  $0.00125 \text{ V} = 1.25 \text{ mV}$

**Uwaga:**

W czasie obliczeń należy uwzględnić fakt, że moduły pozwalają na przekraczanie zakresu.

W pierwszym z przykładów, zmiana wejściowego sygnału analogowego o co najmniej 5 mV powinna powodować zmianę wartości cyfrowej. Jest to najmniejsza zmiana sygnału, dla której należy oczekiwać zmiany wartości cyfrowej.

Nie oznacza to, że mniejsza zmiana nie spowoduje zmiany wartości cyfrowej. Zmiana taka może być uzależniona od poprzedniej i następnej wartości analogowej. W celu zagwarantowania zmiany wartości cyfrowej, sygnał musi się zmienić o co najmniej 5 mV.

Wartość ta nazywana jest również "1 LSB" (1 mniej znaczący bit). Zmiana wartości binarnej o 1 reprezentuje zmianę sygnału analogowego o 5 mV.

Innym często używanym terminem jest "1/2 LSB" (1/2 mniej znaczącego). W przedstawianym przykładzie jest to wartość 2.5 mV lub 1/2 "1 LSB".



## Wartości kwantowane

Jak już wcześniej wspomniano, moduł ADC konwertuje ciągłą wartość analogową na serię dyskretnych wartości kwantowanych. Należy więc mieć świadomość jakie wartości dyskretne mogą być otrzymane dla danego sygnału analogowego.

Ponownie należy rozważyć zakres i rozdzielczość modułu ADC. Dla zakresu +/-10 V (w rzeczywistości +/-10.24 V po uwzględnieniu możliwości przekraczania zakresu w module) oraz 12-bitowej rozdzielczości (4096 poziomów), w przykładzie obliczono, że krok kwantowania wynosi 5 mV.

Z uwagi na działanie procedur korygujących, najmniejsza wartość analogowa (w naszym przypadku -10.24 V) jest przekształcana na wartość cyfrową "000" (zero). Punkty konwersji są przesunięte o 1/2 LSB, a więc po doprowadzeniu wartości analogowej -10.2375 uzyskana zostanie wartość binarna "0000". Wprowadzenie wartości binarnej z zakresu -10.2375 do -10.2325 daje wartość cyfrową "0001", itp., aż do wartości analogowej +10.2375, której odpowiada wartość cyfrowa "4095".

Jeżeli ulegnie zmianie rozdzielczość, ale nie zakres, otrzymana wartość cyfrowa będzie inna. Przykładowo, po zamianie modułu SmartStack na 14-bitowy (16384 poziomy), wielkość kroku zostanie zmieniona na 1.25 mV. Przy zakresie wynoszącym +/-10.24 V, napięcie analogowe -10.24 V zostanie przekształcone na "0". Napięcie analogowe +10.24 V zostanie przekształcone na "16383", a napięcie 0.000 V na "8182".

## Znormalizowane wartości analogowe

Jest oczywiste, że zmiana modułu ADC wymaga wprowadzenia zmian w programie sterującym, w celu uwzględnienia zmian w sposobie przekształcania wartości analogowych na cyfrowe.

Oprogramowanie Cscape i sterowniki Horner APG posiadają rozwiązania, dzięki którym z modułu SmartStack do programu sterującego przesyłane są wartości znormalizowane. Wartości znormalizowane to wartości przekształcone z standardowej wartości cyfrowej, zależnej od rozdzielczości, na zakres -32000 do +32000 (dla sygnałów analogowych przyjmujących zarówno wartości dodatnie jak i ujemne). Wartość znormalizowana jest tak dobrana, aby jej znak był identyczny ze znakiem sygnału analogowego (tzn. dodatnie napięcia są przekształcane na wartości znormalizowane dodatnie, a ujemne napięcia są przekształcane na wartości znormalizowane ujemne). Wejściowy sygnał analogowy o wartości wynoszącej dokładnie 0.000 V jest przekształcany na wartość cyfrową "000".

W czasie normalizacji, do najmniejszej wartości znormalizowanej przypisywany jest najmniejsza, ujemna wartość analogowa, a do największej wartości znormalizowanej przypisywana jest

największa, dodatnia wartość analogowa. Dla przedziału +/-10 V, "-32768" reprezentuje -10.24 V, a "+32767" reprezentuje +10.24 V.

Pomimo, że operacje takie można też zrealizować za pomocą programu sterującego zawierającego zaledwie kilka szczebli, zadanie normalizacji jest realizowane przez sterownik OCS, przed odwoływaniem się do przekonwertowanych wartości cyfrowych z poziomu programu sterującego. Proces normalizacji jest całkowicie niewidoczny z poziomu programu sterowania.

Mając wejściową wartość analogową, można z łatwością obliczyć oczekiwaną wartość cyfrową:

Przekonwertowana wartość = (Wartość wejściowa / Maksymalny zakres) \* 32768

Gdzie:

- Maksymalny zakres - Maksymalna dopuszczalna wartość wejściowa (z uwzględnieniem możliwości przekroczenia zakresu) dla skonfigurowanego zakresu. +/-10 V = "10.24"; +/-5 = "5.12", +/-20 mA = "20.48".

Przykładowo założmy, że na wejście analogowe został doprowadzony sygnał 4.23 V. Dla zakresu +/-10 V:

- Przekonwertowana wartość =  $(4.23 / 10.24) * 32768$
- Przekonwertowana wartość = 41308 \* 32768
- Przekonwertowana wartość = 13536

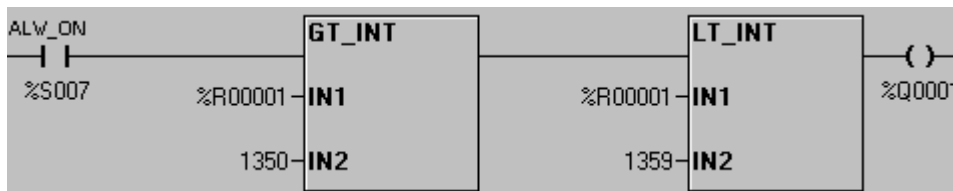
Ale dla zakresu +/-5 V:

- Przekonwertowana wartość =  $(4.23 / 5.12) * 32768$
- Przekonwertowana wartość = 826127 \* 32768
- Przekonwertowana wartość = 27072

Nie wszystkie wartości z przedziału -32768 do +32767 są dostępne. Jest to spowodowane przez rozdzielczość przetwornika, dzielącą cały przedział znormalizowanych wartości cyfrowych na 2N kroków. Przykładowo, dla przetwornika 12-bitowego i zakresu +/-10 V, cały przedział znormalizowanych wartości (65535) jest dzielony na 4096 kroków, o wartości 16 jednostek każdy. Każdy krok reprezentuje 5 mV.

Tak więc dla napięcia 0.000 V otrzymywana jest znormalizowana wartość "0" (zero), a dla napięcia dodatniego 5 mV - "0016", dla napięcia dodatniego 10 mV - "0032", dla napięcia ujemnego 20 mV - "-0064", itp. Z procesem normalizacji związane są też pewne błędy matematyczne. Wielkość kroku nie wynosi dokładnie 16 jednostek (w rozważanym przykładzie). Z tego powodu, znormalizowana wartość może być mniejsza lub większa o 1 jednostkę.

W rzeczywistości jednak często nie są wymagane wszystkie wartości cyfrowe. W większości przypadków, znormalizowane wartości są dalej przekształcane np. na trzy zakresy Niski/Średni/Wysoki lub wyrażane w procentach. W nielicznych sytuacjach, gdzie wymagana jest dokładna wartość, jest ona kwalifikowana do przedziałów za pomocą programu drabinkowego.



*Przykład kwalifikowania wartości temperatury do przedziału*

W przedstawionym powyżej programie sterującym, każda wartość znormalizowana należąca do przedziału wyznaczonego przez liczby "1350" i "1359" powoduje ustawienie zmiennej przekaźnika. W przedziale tym mieści się spodziewana wartość 1356+/-1.

## Sygnaly bez polaryzacji

Sygnaly, które przyjmują wyłącznie dodatnie (lub ujemne) wartości są zwane sygnałami bez polaryzacji (bez znaku). Sygnaly, które przyjmują wyłącznie dodatnie (lub ujemne) wartości są zwane sygnałami z polaryzacją (ze znakiem).

Sygnaly bez polaryzacji są przetwarzane tak samo jak sygnaly z polaryzacją. Niemniej jednak, jeżeli moduł SmartStack jest skonfigurowany dla zakresu bez polaryzacji (przykładowo 0 do +10 V), cała rozdzielczość obowiązuje dla tego zakresu. Rozdzielczość pozorna jest więc dwa razy większa, ponieważ zakres fizycznych wartości wejściowych jest o połowę mniejszy w porównaniu do zakresu z polaryzacją (+/-10 V). Tak więc, w przypadku sygnałów bez polaryzacji, może być rozpoznana mniejsza zmiana sygnału wejściowego.

Nie wpływa to jednak generalnie na zmianę oczekiwanych wartości znormalizowanych. Jeżeli jednak w programie sterującym zaimplementowano kwalifikowanie wartości do przedziałów, należy uwzględnić mniejszą wartość kroku kwantowania w zmniejszonym przedziale wartości.

## Stabilność obwodów analogowych

Parametry układów analogowych nie są stałe, zmieniają się wraz z temperaturą oraz mają tendencje do samoczynnej, powolnej zmiany wartości (jest to jedna z przyczyn dużej popularności urządzeń cyfrowych takich jak odtwarzacze CD, syntetyzatory muzyczne i telefony komórkowe). Dokładne dostrojenie obwodu analogowego jest bardzo trudne. Z tego powodu wartości po konwersji mogą nieco odbiegać od wartości wyznaczonych na drodze matematycznej.

Zalecane jest przeprowadzenie eksperymentu z użyciem znanych wartości w celu przekonania się, jak na prawdę realizowana jest konwersja. Przekonwertowane wartości powinny być bardzo bliskie teoretycznym, mogą się jednak różnić o kilka jednostek. Program sterujący należy zawsze pisać dla tych rzeczywistych wartości, a nie dla teoretycznych.

Wartości analogowe mogą zmieniać się wraz z upływem czasu i temperaturą. Moduły SmartStack odznaczają się bardzo dobrą stabilnością parametrów, natomiast znacznie bardziej może się zmieniać doprowadzany sygnał analogowy w funkcji czasu lub temperatury. Jeżeli otrzymywane wartości wykazują tendencję do odchylenia od wartości wyznaczonych eksperymentalnie, prawdopodobną przyczyną jest źródło sygnału wejściowego, a nie moduł SmartStack.

## Zakłócenia

W obwodach analogowych dużą rolę odgrywają zakłócenia. Zakłócenia należy zredukować do absolutnego minimum, zwłaszcza przy korzystaniu z przetworników 14- i 16-bitowych. Przyczyną zakłóceń może być tętnienie źródła zasilania, przydźwięk, trzaski komutacyjne, zakłócenia elektromagnetyczne oraz zakłócenia powodowane przez uziemienie. Większość zakłóceń powstaje w źródle sygnału wejściowego, a nie w przetworniku ADC.

Im większa jest rozdzielczość przetwornika ADC, tym mniejszy poziom zakłóceń jest dopuszczalny. Dla zakresu wartości  $\pm 10$  V oraz dla przetwornika 10-bitowego poziom zakłóceń może wynosić 0.02 V (20 mV), ale już dla przetwornika 12-bitowego poziom zakłóceń powinien być mniejszy od 5 mV.

Nie ma źródła sygnału pozbawionego całkowicie zakłóceń. Niektóre zakłócenia przedostają się do systemu, nawet pomimo podjęcia wszelkich starań w celu ich wyeliminowania. Wpływ zakłóceń na uzyskiwaną po konwersji wartość cyfrową (a więc także na wartość znormalizowaną) zależy od rozdzielczości przetwornika oraz od wartości wejściowego sygnału analogowego. Poziom zakłóceń w systemie musi być znany i zredukowany do akceptowalnego poziomu.

Przykładowo, założmy że punkty kwantyzacji przetwornika skonfigurowano tak, że dla sygnału wejściowego mniejszego od  $-0.0025$  V uzyskiwana jest wartość cyfrowa "2047", dla sygnału  $+0.0025$  V - wartość "2049, a dla sygnału z przedziału  $-0.0025$  do  $+0.0025$  V - wartość "2048" (ma to miejsce w przypadku przetwornika 12-bitowego, dla którego wybrano zakres  $\pm 10$  V).

W pierwszym przykładzie założmy, że napięcie wejściowe ma wartość dokładnie 0.000 V, a poziom zakłóceń wynosi 4 mV. Jest to wartość łączna dla wszystkich możliwych źródeł zakłóceń. Taki poziom zakłóceń oznacza, że przy dowolnej, stałej wartości napięcia wejściowego, wartość ta może się zmieniać w granicach  $-0.002$  do  $+0.002$  V. Dla tych danych, wszystkie możliwe poziomy napięć zostaną poprawnie przekształcane na wartość cyfrową "2048".

W drugim przykładzie założmy, że napięcie wejściowe ma wartość dokładnie  $+0.001\text{ V}$ , a poziom zakłóceń wynosi  $4\text{ mV}$ . Taki poziom zakłóceń oznacza, że przy dowolnej, stałej wartości napięcia wejściowego, wartość ta może się zmieniać w granicach  $-0.0001$  do  $+0.003\text{ V}$ . Sygnał  $+0.03\text{ V}$  jest konwertowany na wartość cyfrową "2049". W zależności od momentu dokonywania konwersji, przekonwertowana wartość cyfrowa może wynosić "2048" lub "2049".

W trzecim przykładzie założmy, że napięcie wejściowe ma wartość dokładnie  $0.000\text{ V}$ , a poziom zakłóceń wynosi  $6\text{ mV}$ . W tym przypadku, wartości chwilowe napięcia mogą wynosić  $-0.003$  do  $+0.003\text{ V}$ . Przekonwertowana wartość cyfrowa może wynosić "2047", "2048" lub "2049" w zależności od momentu, w którym przeprowadzana jest konwersja.

Zjawisko samoczynnej zmiany wielkości jest określane często jako "fluktuacja". Mając świadomość występowania zakłóceń w systemie, można uwzględnić i zaakceptować pewien poziom fluktuacji. Należy dążyć do tego, aby na wskutek fluktuacji oczekiwana wartość nie zmieniała się więcej niż  $\pm 1$  jednostkę. W powyższym przykładzie będzie to powodować poziom zakłóceń mniejszy od 1 LSB, lub mniejszy od  $5\text{ mV}$ . W pewnych systemach dopuszczalny jest większy poziom zakłóceń (a więc i fluktuacja).

Fluktuacja konwertowanych wartości jest zjawiskiem normalnym. W zamieszczonym powyżej przykładzie, fluktuacja wartości binarnej nie przekracza 16 jednostek w skali  $-32000$   $+32000$  ( $\pm 1$  jednostka w skali przetwornika) znormalizowanej wartości. W przykładach powyżej, każda wartość cyfrowa "2048" jest normalizowana do "0000  $\pm 1$ ", wartość cyfrowa "2047" jest normalizowana do "-0016  $\pm 1$ ", a wartość cyfrowa "2049" jest normalizowana do "0016  $\pm 1$ ". W celu uwzględnienia w programie sterującym tego poziomu zakłóceń, wartość  $0.000\text{ V}$  (prawdopodobnie "0000" powinna być kwalifikowana na podstawie przedziału "-0017" do "+0017").

Oczywiście, dopuszczalny poziom zakłóceń jest w pewnym stopniu wyznaczany na podstawie rozdzielczości przetwornika ADC. Zastosowanie przetwornika 10-bitowego w miejsce 12-bitowego pozwala na czterokrotne zwiększenie dopuszczalnego poziomu zakłóceń.

Dodatkowe informacje na temat przekształcania wartości analogowych na cyfrowe podane są w następnym rozdziale: "Moduł do czujników termoparowych i oporowych".

## Moduł do czujników termoparowych i oporowych

### Informacje ogólne

Moduł do czujników termoparowych i oporowych pracuje na tej samej zasadzie co zwykle moduły ADC, ale wartość cyfrowa jest w nich konwertowana na stopnie, a nie do skali -32000 do +32000. Dodatkowo, moduły te linearyzują programowo wartości, natomiast moduły ADC wyłącznie konwertują otrzymane wartości i poddają je standardowej normalizacji.

### Czujniki oporowe do pomiaru temperatury

Czujnik oporowy do pomiaru temperatury (RTD – Resistance Temperature Device) zachowuje się jak rezystor, zmieniający swoją oporność w zależności od temperatury. Element mierzący jest wykonany z platyny, a krzywa zależności temperatury od rezystancji jest powszechnie znana. Ogólnie ujmując, czujniki RTD są zalecane dla zakresu temperatury  $-200\text{ }^{\circ}\text{C}$  do  $+600\text{ }^{\circ}\text{C}$ . W standardowych czujnikach RTD, dla temperatury  $0\text{ }^{\circ}\text{C}$  rezystancja wynosi  $100.00\text{ ohm}$ . Dostępne są również urządzenia, dla których ta rezystancja wynosi  $200\text{ ohm}$ ,  $500\text{ ohm}$  lub  $1000\text{ ohm}$ .

Często dla czujników RTD podawany jest również parametr alpha (zwany również współczynnikiem temperaturowym rezystancji - TCR). Jest to miara krzywej temperaturowej urządzenia. Współczynnik alfa jest miarą czystości platyny wykorzystywanej w tych urządzeniach. Im większa wartość tego współczynnika, tym większy stopień czystości platyny. Zwykle współczynnik ten wynosi od  $0.00375$  do  $0.003927$ . W większości urządzeń współczynnik alpha wynosi  $0.00385$ . Najistotniejsze jest jednak, aby przy wymianie zastosować czujnik RTD o takim samym współczynnikiem alpha.

Zmiana rezystancji w czujniku RTD jest mierzona za pomocą Mostka Wheatstone'a. Mostek Wheatstone'a składa się z czterech rezystorów, jednym z nich jest czujnik RTD. Jakkolwiek zmiana rezystancji czujnika RTD powoduje zakłócenie równowagi mostka, wynikający stąd przepływ prądu można zmierzyć i zamienić na napięcie, które jest następnie wysyłane do przetwornika analogowo cyfrowego ADC. Przetwornik ADC przekształca wartość napięcia na wartość binarną, wykorzystywaną przez urządzenie cyfrowe.

Zmiana temperatury powoduje bardzo małą zmianę rezystancji czujnika RTD. Zmiana rezystancji jest tak mała, że często jest ona mniejsza od rezystancji przewodów łączących czujnik RTD z modułem dla czujników oporowych. Z tego powodu, bardzo często stosowane są dwa specjalne rozwiązania czujników RTD. Są to tzw. czujniki RTD "trójprzewodowe" i "czteroprzewodowe". W rozwiązaniu takim rezystancje przewodów w obwodzie pomiarowym znoszą się wzajemnie, a czynnikiem decydującym o wyniku pomiaru jest sama rezystancja czujnika RTD.

## Termopary (THM)

Bardzo upraszczając, termopara składa się z dwóch przewodów wykonanych z różnych metali wzajemnie skręconych. Czujniki termoparowe wykorzystują zjawisko Seebecka, polegające na generowaniu napięcia przez dwa splecione przewody wykonane z różnych materiałów, po podgrzaniu ich do wyższej temperatury.

Wielkość generowanego napięcia uzależniona jest od rodzaju metali oraz od wzrostu temperatury przewodów. Jest to tzw. współczynnik Seebecka. Współczynnik Seebecka ma jednak bardzo małą wartość (zwykle wynosi ona kilka mikrowoltów na stopień zmiany temperatury a otrzymywany sygnał wyjściowy, w większości przypadków ma jedynie kilka miliwoltów). Dodatkowo krzywa obrazująca zależność napięcia od temperatury ma tendencje do krzywoliniowego przebiegu.

Szereg eksperymentów przeprowadzonych w ciągu 170 lat zaowocował opracowaniem kombinacji metali i stopów, zapewniających stosunkowo wysoki poziom napięcia wyjściowego (nadal jednak jest ono rzędu kilku mikrowoltów na stopień) oraz odporność na szereg czynników środowiskowych, takich jak temperatura, korozyjność i radioaktywność atmosfery.

Poniżej podano powszechnie stosowane w termoparach metale i stopy:

- Miedź                    100% czystej miedzi
- Żelazo                    100% czystego żelaza
- Platyna                    100% czystej platyny
- 15% niklu, 55% miedzi
- Chromel                    90% niklu, 10% chromu
- Alumel                    95 niklu, 2% aluminium, 2% manganu, 1% silikonu
- Nikrosil                    84.6% niklu, 14% chromu, 1.4% silikonu
- Nisil                        95.6% niklu, 1.4% silikonu

W ciągu lat opracowano szereg różnych kombinacji przewodów z tych materiałów. Zostały one zatwierdzone przez American National Standard Institute (ANSI) z uwagi na wykazywanie powtarzalności krzywej obrazującej zależność temperatury od napięcia.

Typ	Przewód dodatni	Przewód ujemny	Zakres pomiarowy	Cechy specjalne
C	5% renu	26% renu	-4200 °F	Słabe utlenianie. Czujnik musi być używany w atmosferze próżniowej, wodorowej lub gazów obojętnych.
E	Chromel	Konstantan	-300 - 1650 °F	Może występować fluktuacja w zakresie 600-1100 °F
J	100% żelaza	Konstantan	0 - 1500 °F	Stabilny. Brak fluktuacji w zakresie 700-1000 °F

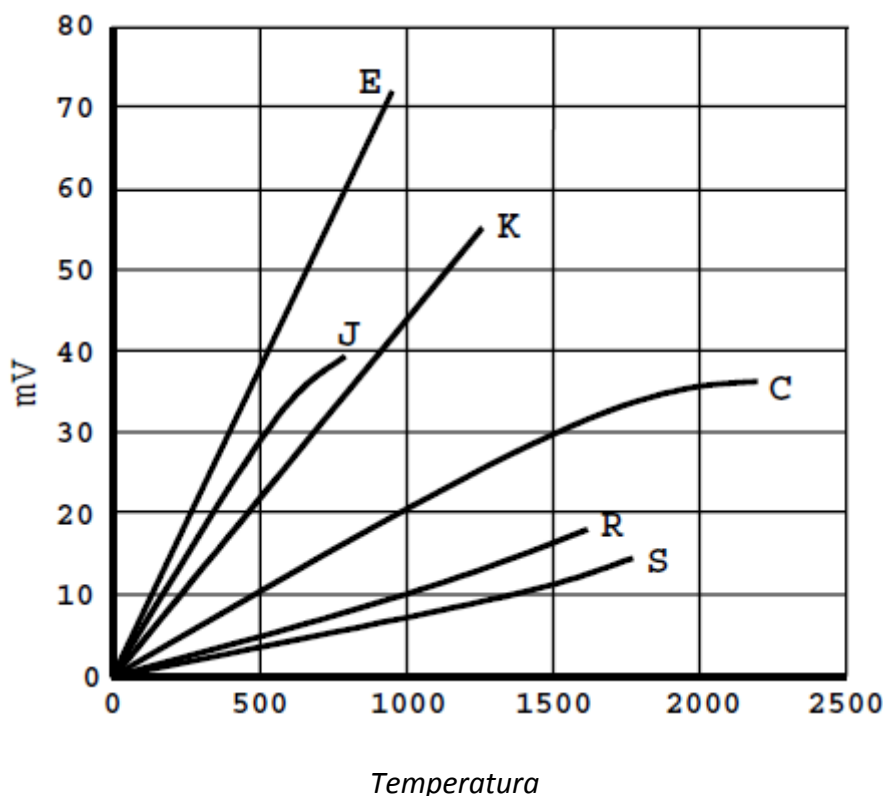
K	Chromel	Alumel	-300 - 2300 °F	Może występować fluktuacja w zakresie 600-1000 °F
T	100% miedzi	Konstantan	-300 - 660 °F	Bardzo stabilny w zakresie niskich temperatur
N	Nikrosil	Nisil	0 - 2300 °F	Bardziej stabilny w zakresie 600-1100 °F Większa odporność na promieniowanie cząstek elementarnych w porównaniu do typu K
R	13% rod	100% platyny	0 - 2700 °F	Wysoka odporność na utlenianie i korozję. Może ulec zanieczyszczeniu w czasie kontaktu z oparami wodoru lub węgla
S	10% rod	100% platyny	0 - 2700 °F	Może ulec zanieczyszczeniu w czasie kontaktu z oparami wodoru lub węgla
B	30% rod	6% rod	1600 - 3100 °F	Wysoka odporność na utlenianie i korozję. Może ulec zanieczyszczeniu w czasie kontaktu z oparami wodoru lub węgla

**Uwaga:**

Nie wszystkie typy czujników termoparowych mogą współpracować z modułami SmartStack. Wartości zakresów są przybliżone. Czujniki dostarczane przez konkretnych producentów mogą mieć inne zakresy. Typ 'C' nie jest objęty normą ANSI

Zamieszczony poniżej rysunek zawiera charakterystyki powszechnie spotykanych czujników termoparowych.





Pomimo że napięcie wyjściowe z czujnika termoparowego jest bardzo małe, można je przekształcić na wartość cyfrową za pomocą przetwornika ADC. Krzywe charakterystyczne termopar są nieliniowe, jeżeli jednak znany jest typ ANSI, można wynik zlinearyzować, korzystając z odpowiedniego oprogramowania. Należy zwrócić uwagę, aby w module SmartStack skonfigurować odpowiedni typ czujnika termoparowego, który będzie faktycznie zainstalowany oraz aby w czasie wymiany czujnika termoparowego na inny odpowiednio przekonfigurować moduł SmartStack.

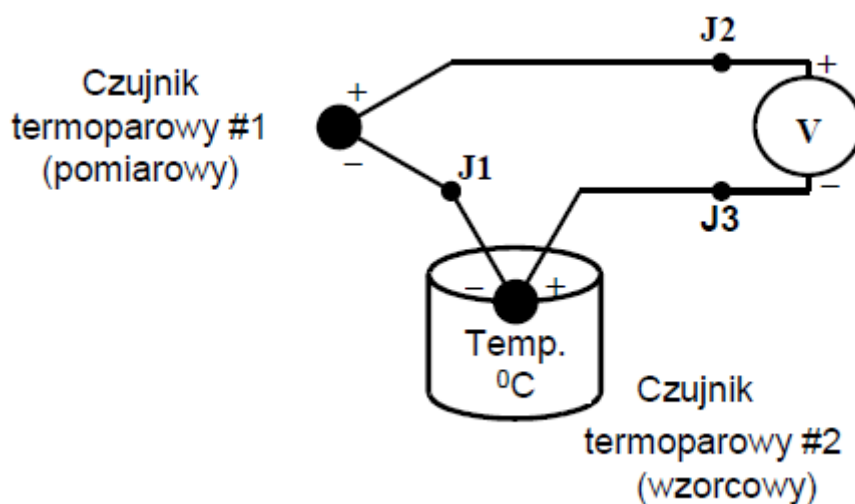
Zjawisko Seebecka jest także przyczyną głównej wady czujników termoparowych. Jakikolwiek dwa różne metale generują po zetknięciu napięcie. Oznacza to, że nie można wykorzystać standardowych przewodów miedzianych do podłączania czujnika termoparowego (za wyjątkiem przewodu dodatniego czujnika termoparowego typu K), ponieważ na styku miedzi i innego metalu, z powodu zjawiska Seebecka, wygenerowane zostanie napięcie, wprowadzając błąd do pomiaru.

Czujnik termoparowy należy podłączać za pomocą specjalnych przewodów. W rzeczywistości, wszystkie połączenia elementów metalowych w instalacji, jak na przykład gniazdka, wtyczki, tablice połączeń, itp., muszą być z tych samych materiałów co czujnik termoparowy. Może to powodować znaczny wzrost kosztów, zwłaszcza gdy czujnik termoparowy umieszczony jest w znacznej odległości od modułu SmartStack. Przewody przyłączeniowe czujnika termoparowego są wykonane z dokładnie takich samych materiałów jak czujnik termoparowy (mogą pracować jednak w znacznie mniejszym zakresie temperatur), dzięki czemu są tańsze. Można również stosować przewody kompensacyjne ze stopów. W przewodach kompensacyjnych również występuje zjawisko Seebecka, identyczne jak w czujnikach termoparowych, ale są one znacznie tańsze.

## Kompensacja metodą zimnych końców

Nie można całkowicie uniknąć łączenia różnych metali. Przykładowo, ma to miejsce w punkcie, w którym czujnik termoparowy (lub jego przewody) są wprowadzane do modułu SmartStack czujnika termoparowego. W punkcie tym generowane jest napięcie Seebecka, wprowadzając błąd do systemu. Błąd ten można jednak zmniejszyć stosując kompensację metodą zimnych końców.

Klasyczna kompensacja metodą zimnych końców polega na szeregowym włączeniu do obwodu pomiarowego identycznego czujnika termoparowego. Dodatkowy czujnik termoparowy jest umieszczany w stałej temperaturze, najlepiej w  $0^{\circ}\text{F}$ , skąd zresztą wzięta się nazwa tej metody.



Obydwa przewody ujemne czujników termoparowych są połączone w punkcie "J1". Ponieważ przewody te są wykonane z tych samych materiałów, w punkcie tym nie jest generowane napięcie Seebecka. Dwa pozostałe przewody są doprowadzane do gniazd modułu do czujników termoparowych. W gniazdach tych generowane jest napięcie Seebecka, ale jest ono identyczne w obydwu gniazdach. Ponieważ te dwa napięcia są "po przeciwnych stronach", wzajemnie się znoszą.

W przedstawionym powyżej schemacie napięcie na czujniku termoparowym #2 jest odejmowane od napięcia na czujnik termoparowym #1. Czujnik termoparowy #2 jest przechowywany w stałej temperaturze  $0^{\circ}\text{C}$ , a więc występujące na nim napięcie można z łatwością wyznaczyć posługując się powszechnie znanymi normami ANSI. Ten stały błąd pomiaru można z łatwością skorygować w oprogramowaniu, poprzez odjęcie odpowiedniej wartości.

Utrzymywanie stałej temperatury  $0^{\circ}\text{C}$  jest jednak bardzo kosztowne i czasochłonne. Moduły SmartStack do czujników termoparowych umożliwiają korzystanie z zarówno kompensacji wewnętrznej, jak i kompensacji zdalnej metodą zimnych końców. W przypadku kompensacji wewnętrznej w pobliżu gniazd wejściowych modułu SmartStack umieszczany jest półprzewodnikowy element do pomiaru temperatury. Pozwala to na pomiar temperatury na

gniazdach wejściowych modułu SmartStack oraz na matematyczne skorygowanie napięcia odczytanego z czujnika termoparowego. W przypadku modułu SmartStack, kompensacja taka jest przeprowadzana automatycznie. Należy jednak zwrócić uwagę, że pomiędzy modułem SmartStack a czujnikiem termoparowym muszą być stosowane odpowiednie przewody przyłączeniowe czujnika termoparowego.

W przypadku zdalnej kompensacji metodą zimnego połączenia, półprzewodnikowy element do pomiaru temperatury jest umieszczany w oddalonym "bloku kompensacyjnym". Do bloku tego podłączany jest również czujnik termoparowy, a więc zjawisko Seebecka powstaje w tym bloku kompensacyjnym. Blok ten może być umieszczony w miejscu oddalonym od komputera, zwykle jednak nie jest umieszczany w środowisku, w którym dokonywany jest pomiar temperatury. Sygnały z czujnika termoparowego i z elementu półprzewodnikowego do pomiaru temperatury są przesyłane do modułu SmartStack standardowymi (taniami) przewodami miedzianymi, podłączonymi do tego bloku kompensacyjnego.

Dzięki takiemu rozwiązaniu, moduł SmartStack może mierzyć temperaturę w "bloku kompensacyjnym" i drogą matematyczną korygować napięcie odczytane z czujnika termoparowego. W przypadku kompensacji tą metodą, funkcja automatycznej kompensacji wewnętrznej jest samoczynnie wyłączana.

## Wartości pomiarowe uzyskane z modułu SmartStack

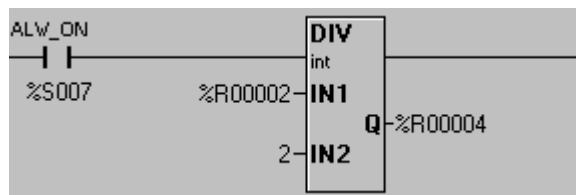
---

Sygnały pomiarowe uzyskane z modułu SmartStack do czujników termoparowych i oporowych są już skalibrowane w stopniach. Mogą one być wyrażone zarówno w skali Celsjusza jak i Fahrenheita. Rozdzielczość zwykle wynosi  $0.05^{\circ}$ ,  $0.1^{\circ}$  lub  $0.5^{\circ}$ .

Zwracana wartość to liczba całkowita ze znakiem reprezentująca temperaturę wyrażoną w dziesiątych częściach stopnia. Przykładowo, dla temperatury  $100^{\circ}\text{C}$  i rozdzielczości ustawionej na  $0.5^{\circ}\text{C}$ , zwracana wartość wynosi 200. Oznacza to, że temperatura składa się z 200 jednostek, o wartości  $0.5^{\circ}\text{C}$  każda. Taka sama temperatura jest przy rozdzielczości  $0.10^{\circ}\text{C}$  reprezentowana przez wartość 1000 lub przy rozdzielczości  $0.05^{\circ}\text{C}$  przez wartość 2000.

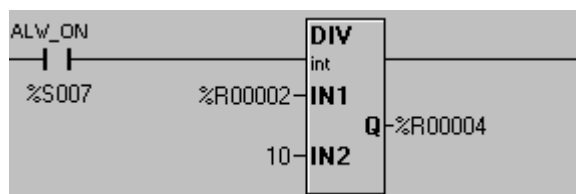
Zawsze, ilekroć to jest możliwe, wartość odczytana z modułu do czujników termoparowych i oporowych powinna być przechowywana i przetwarzana w formacie całkowitym. Ma to na celu uniknięcie wykonywania czasochłonnych operacji na liczbach rzeczywistych. Jeżeli części dziesiątne stopnia nie są wykorzystywane, wartość można zamienić bezpośrednio na stopnie za pomocą prostych operacji matematycznych.

Jeżeli rozdzielczość wynosi  $0.5^{\circ}\text{C}$ :



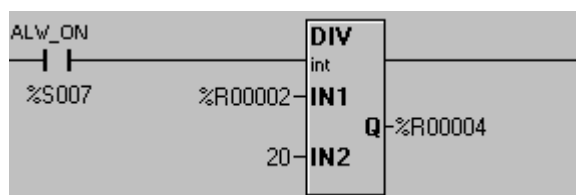
*Przykład dzielenia przez 2*

Jeżeli rozdzielczość wynosi 0.1 °C:



*Przykład dzielenia przez 10*

Jeżeli rozdzielczość wynosi 0.05 °C:



*Przykład dzielenia przez 20*

## Bloki funkcyjne PID

### Terminologia

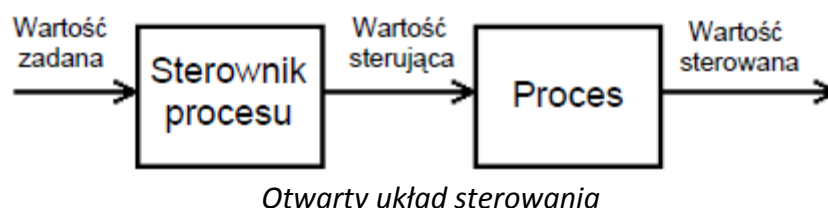
Przed rozpoczęciem opisywania działania bloków PID zdefiniowane zostaną podstawowe terminy, niezbędne do zrozumienia omawianych zagadnień.

PID  
 Wartość zadana  
 Wartość sterująca (wynikowa)  
 Przesunięcie punktu pracy  
 Odchyłka regulacji (Bias)  
 Sterowanie proporcjonalne  
 Sterowanie całkowe  
 Sterowanie różniczkowe

- **K** - Wzmocnienie w otwartym układzie sterowania, wyznaczone przez stosunek zmiany wielkości regulowanej/zmiany wielkości sterującej.
- **Kp** - Współczynnik wzmocnienia proporcjonalnego. Jest to miara zmiany wartości sterowanej odpowiadająca zmianie uchybu. Współczynnik ten nazywany jest też współczynnikiem wzmocnienia regulatora (Kc).
- **Ki** - Współczynnik wzmocnienia całkowego. Określa okres czasu określający częstotliwość "całkowania" uchybu. Krótszy okres czasu odpowiada większemu wzmocnieniu wyrazu całkowego.
- **Kd** - Współczynnik wzmocnienia różniczkowego. Współczynnik ten określa natężenie wprowadzania zmian do systemu.
- **SP** - Wartość zadana. Wartość, którą należy uzyskać i utrzymać w procesie.
- **PV** - Wartość regulowana. Wartość regulowana w procesie, np. temperatura, ciśnienie, itp.
- **CV** - Wartość sterująca. Wartość wyjściowa funkcji PID. Na wartość tą mają wpływ części: proporcjonalna, całkowa, różniczkowa i odchyłka regulacji.

### Wprowadzenie

W typowym procesie przemysłowym bardzo często pewne parametry tego procesu, jak temperatura czy ciśnienie, muszą być sterowane. Zadanie to może być realizowane w układzie otwartego sterowania:



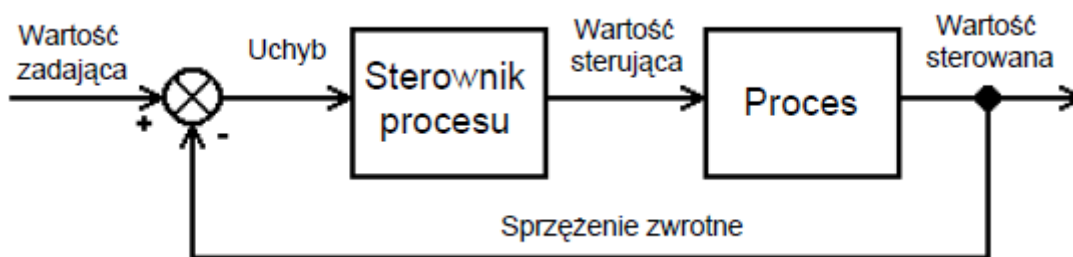
W systemie takim, użytkownik procesu wprowadza do sterownika wartość. Wartość ta jest nazywana wartością zadaną. Sterownik procesu generuje wartość przesyłaną do procesu, czyli tzw. wartość sterującą. W odpowiedzi na wysłanie wartości sterującej, ulega zmianie wartość regulowana.

Wadą takiego rozwiązania jest fakt, że sterownik procesu nie ma żadnych informacji, czy uzyskana została w procesie żądana wartość regulowana. Sterownik procesu musi przyjąć założenie, że proces przebiega bardzo szybko i dokładnie z zamierzeniami. W wielu przypadkach, takie założenie jest zbyt dużym uproszczeniem.

Uproszczenie to powoduje duże niedokładności. Przyczyną może być po prostu sama niedokładność procesu. Przykładowo, zamiast do żądanej temperatury  $350^{\circ}\text{C}$ , podgrzewacz może podgrzać materiał do  $400^{\circ}\text{C}$ . Istnieje również prawdopodobieństwo, że w samym procesie wystąpią pewne zmiany powodujące błędy. Także dodanie półfabrykatów o niskiej lub wysokiej temperaturze z pewnością wpłynie na temperaturę procesu.

Jakakolwiek zmiana w systemie powodująca w efekcie zmianę wielkości regulowanej jest nazywana wymuszeniem. Wymuszenie można wywołać celowo poprzez zmianę wartości zadanej lub też może być efektem ubocznym realizacji procesu, jak to ma przykładowo miejsce przy doprowadzaniu lub usuwaniu pewnych materiałów z procesu. System sterowania musi reagować zarówno na wymuszenia, jak i na zmiany powodujące obniżenie lub wzrost wartości regulowanej.

Większość procesów korzysta do tego celu ze sprzężenia zwrotnego. Jest to tzw. sterowanie ze sprzężeniem zwrotnym. W sterowaniu takim dokonywany jest pomiar wartości regulowanej, która następnie jest przesyłana do sterownika procesu.



*Sterowanie ze sprzężeniem zwrotnym*

Proces taki może reagować zarówno na zmianę wielkości zadanej, jak i na zmiany w procesie lub obciążeniu. Po zmianie wartości zadanej, sterownik będzie tak sterował procesem, aby dojść do tej nowej wartości. Jeżeli wartość regulowana ulegnie zmianie, sterownik będzie dążył aby sprowadzić tę wartość do wartości zadanej.

Należy rozważyć, jakie możliwości daje sprzężenie zwrotne. W większości zastosowań, wartość z pętli sprzężenia zwrotnego jest odejmowana od wartości zadanej, w wyniku czego otrzymujemy

Wartość zadana    Wartość sterująca    Wartość sterowana    Sterownik procesu    Proces    Wartość

zadająca Wartość sterująca Wartość sterowana Sterownik procesu Proces Uchyb Sprężenie  
 zwrotne jest uchyb. Wielkość uchybu określa różnicę pomiędzy wartością zadanej a wartością regulowaną. Przykładowo, wartość zadana może wynosić  $350^{\circ}\text{C}$ . Załóżmy, że odczytana wartość regulowana wynosi  $200^{\circ}\text{C}$ . Tak więc, uchyb będzie wynosił  $150^{\circ}\text{C}$ .

Pierwszym nasuwającym się pomysłem jest dodanie uchybu do wartości zadającej i przesłanie wyniku tej operacji do procesu, w celu uzyskania żądanej wartości regulowanej w jak najkrótszym czasie. W rozważanym przykładzie, po dodaniu wartości zadającej  $350^{\circ}\text{C}$  do uchybu wynoszącego  $150^{\circ}\text{C}$ , podjęta zostanie próba doprowadzenia temperatury w procesie do  $500^{\circ}\text{C}$ , w efekcie czego nagrzewanie będzie znacznie bardziej intensywne.

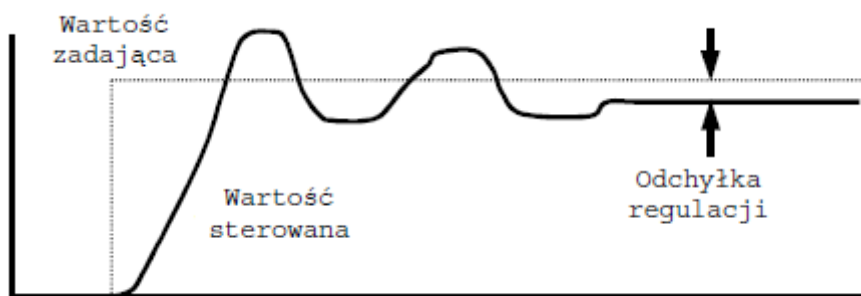
W następnym cyklu, niech przykładowo wartość regulowana wynosi  $250^{\circ}\text{C}$ , a więc uchyb ulegnie zmniejszeniu do  $100^{\circ}\text{C}$ . Sterownik doda uchyb do wartości zadającej i będzie dążył do uzyskania w procesie temperatury  $450^{\circ}\text{C}$ . W czasie kolejnych cykli sterowania, temperatura w procesie będzie coraz wyższa. Uchyb będzie mniejszy, a temperatura procesu będzie coraz bliższa żądanej.

Wadą takiego rozwiązania jest fakt, że proces może przebiegać za szybko, aby sterownik mógł reagować na zachodzące w nim zmiany. W rozważanym przykładzie, temperatura może wzrosnąć do  $400^{\circ}\text{C}$  lub więcej stopni, zanim sterownik zdąży ją obniżyć. Może wystąpić również sytuacja odwrotna, temperatura może znacznie spaść, zanim sterownik podejmie odpowiednie działania.

## Sterowanie proporcjonalne

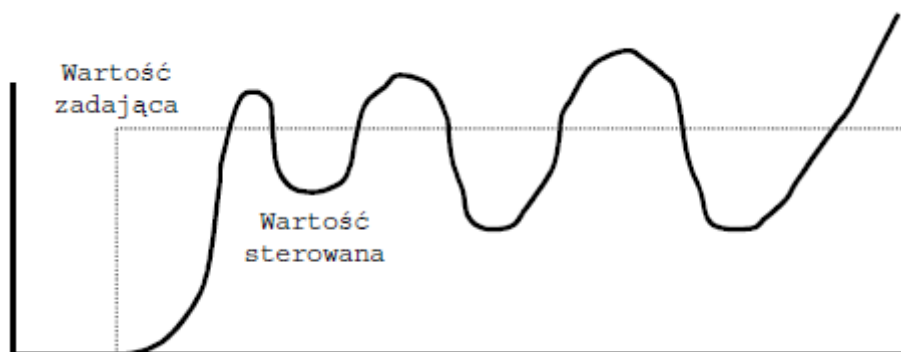
To zadanie sterowania realizowane jest przez regulator proporcjonalny. W praktyce uchyb stanowi w rzeczywistości część (często wyrażoną w procentach) pełnego uchybu. W rozważanym przykładzie, jeżeli obliczony zostanie uchyb o wartości  $150^{\circ}\text{C}$ , sterownik może być tak zaprogramowany, aby dodawać tylko 20-30% tej wartości uchybu. Zmiany w procesie są wolniejsze, co jest powodowane przez mniejsze wahania wartości sterującej, a jednocześnie całe sterowanie jest bardziej dokładne.

Poniżej zamieszczono przykład typowego procesu sterowanego proporcjonalnie.



W sterowaniu proporcjonalnym często podawany jest współczynnik przesunięcia. Proces bowiem prawie nigdy nie ma zerowego uchybu. Jest to powodowane przez szereg przyczyn, leżących poza możliwościami oddziaływania sterowania proporcjonalnego.

Z drugiej strony, przy zbyt dużym udziale sterowania proporcjonalnego, mogą wystąpić oscylacje wartości regulowanej i utrata sterowności procesu.



W procesie prawie zawsze występuje zwłoka lub opóźnienie. Większość wartości regulowanych nie może natychmiast zmienić. Dotyczy to zwłaszcza procesów cieplnych. Zmiany w procesie cieplnym przebiegają bardzo wolno. Podobne opóźnienia mogą również wystąpić przy sterowaniu ciśnieniem i natężeniem przepływu. Przyczynami są zachodzące w systemie zjawiska fizyczne, będące poza strefą oddziaływania sterownika procesu.

## Przesunięcie punktu pracy (BIAS)

ODCHYŁKA REGULACJI to stały uchyb, który można w prosty sposób usunąć poprzez dodanie takiej samej wartości ale z przeciwnym znakiem. Jest to stała wartość, wyznaczana przez użytkownika, nie zmieniana przez funkcję PID. Stosując sterowanie proporcjonalne i niewielkie przesunięcie punktu pracy można sterować wieloma procesami.

## Sterowanie całkowite

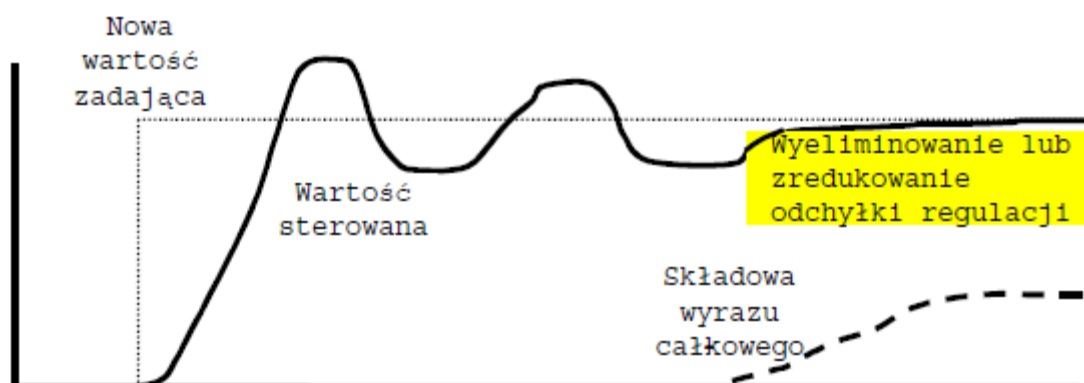
Funkcje całkowania zostały wprowadzone w celu zredukowania wielkości błędu przesunięcia. Funkcja całkowania mierzy czas trwania uchybu oraz generuje odpowiednią składową tego uchybu. Wartość jest dobierana w taki sposób, że następuje prawie całkowite wyeliminowanie błędu odchyłki regulacji.

Zbieranie i wygładzanie wartości w czasie jest nazywane całkowaniem. Z uwagi na właściwości całkowania, wyraz całkowity sterowania jest uwzględniany dopiero w momencie dochodzenia wartości sterowanej do stanu stabilnego (tzn. przy mniejszych wartościach uchybu). Całkowanie powoduje "wygładzenie" szybkich zmian uchybu i ma mniejszy wpływ na proces. Wraz z



dochodzeniem wartości sterownej do stanu stabilnego, uchyb całkowania ma coraz większe znaczenie, co powoduje zniesienie przesunięcia wprowadzonego przez sterowanie proporcjonalne.

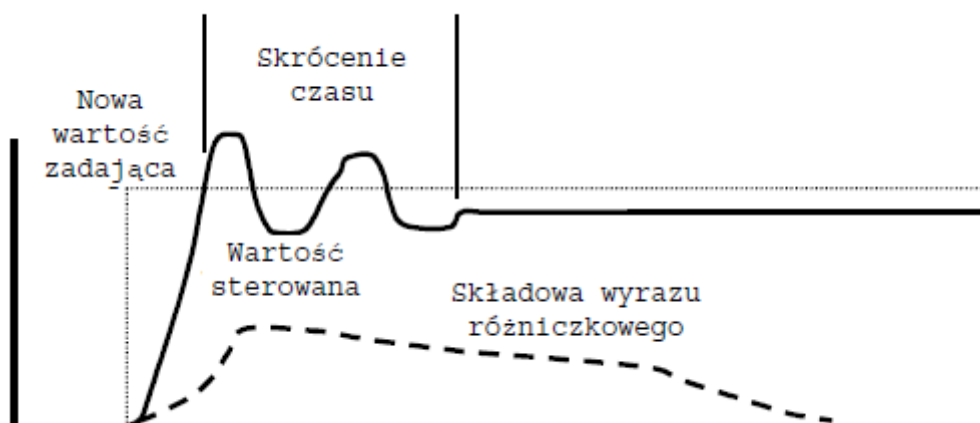
Wadą sterowania całkowego jest niewłaściwa reakcja na szybkie zmiany, zarówno wartości zadanej jak i wartości regulowanej. Pomimo, że sterowanie proporcjonalne pozwala na utrzymanie wartości zadającej, w przypadku szybkich zmian tej wartości zadanej lub wartości regulowanej, możliwości sterowania całkowego są bardzo ograniczone.



Wiele procesów daje się natomiast dobrze sterować za pomocą sterowania proporcjonalno całkowego. W tym przypadku odchyłka regulacji jest redukowana prawie do zera.

## Sterowanie różniczkowe

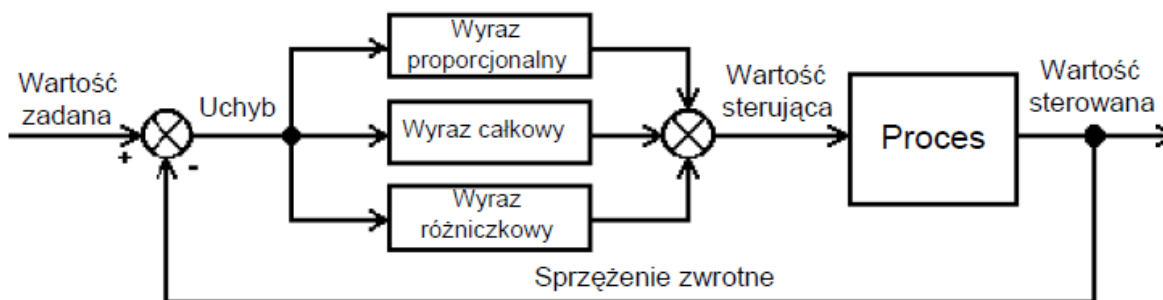
Sterowanie różniczkowe ma na celu uwzględnienie szybko zachodzących zmian w procesie. Sterowanie to generuje następny sygnał uchybu, zależny od nachylenia uchybu, czyli szybkości zmian wartości uchybu w czasie. W przypadku żądania pierwszej zmiany nachylenie uchybu jest bardzo strome a udział sterowania różniczkowego w uchybie znaczny. Wraz z dochodzeniem procesu do stanu stabilnego, szybkość zmian uchybu w czasie zmniejsza się, a tym samym maleje udział sterowania różniczkowego.



## PID

W przypadku dużej liczby procesów wystarcza zastosowanie sterowania proporcjonalnego. Sterowanie różniczkowe lub całkowe nie są samodzielnie wykorzystywane do sterowania procesami. Zarówno sterowanie całkowe jak i różniczkowe są stosowane jako elementy pomocnicze, umożliwiające lepsze reagowanie na zmieniający się stan procesu.

**PID** to skrót od **Proportional Integral Derivative** (Regulator Proporcjonalno-CałkowoRóżniczkowe). PID to funkcja wykorzystująca wszystkie trzy opisywane powyżej metody sterowania jednocześnie w celu wygenerowania odpowiedniej wartości sterującej. Funkcja ta nie tylko uwzględnia samą wartość uchybu (sterowanie proporcjonalne), ale również czas jego trwania (sterowanie proporcjonalne) i szybkość zmian uchybu w czasie (sterowanie różniczkowe).



W momencie wystąpienia uchybu w procesie najpierw podejmowane są próby skorygowania go za pomocą wyrazu proporcjonalnego. Wyraz różniczkowy mierzy wielkość tych zmian i zwiększa wartość uchybu, co ma na celu szybsze uzyskanie zdefiniowanej wartości zadanej. Udział wyrazu całkowego jest bardzo mały, ponieważ wartości uchybu zmieniają się w dużych granicach.

Wraz ze wzrastającą sterownością procesu wielkość uchybu maleje. Wyraz proporcjonalny nadal dąży do uzyskania wartości zadającej, ale zmiany uchybu są coraz mniejsze i coraz bardziej maleje udział wyrazu różniczkowego. Wyraz całkowy wykrywa, że wartość uchybu zbliża się do stałej wartości, podejmuje działania mające na celu zredukowanie uchybu powodowanego przez odchyłkę regulacji.

Po dojściu procesu do stanu stabilnego, wyraz proporcjonalny generuje uchyb o bardzo małej wartości, starając się zrównoważyć wartość odchyłki regulacji. Wyraz całkowy mierzy jak długi okres uchyb pozostaje bez zmian i generuje swój własny sygnał kompensacyjny. Ponieważ natężenie zmian uchybu jest bardzo małe, udział wyrazu różniczkowego jest prawie zerowy.

Powszechnie stosowane są dwie metody implementowania sterowania PID - metoda o niezależnych wyrazach i metoda ISA.

### **Blok funkcyjny PID o niezależnych wyrazach**

Sygnał sterujący regulatora =  $K_p * U_{chyb} + K_i * U_{chyb} * dt + K_d * \text{pochodna} + \text{odchyłka regulacji}$

### **Blok funkcyjny PID-ISA**

Sygnał sterujący regulatora =  $K_p * (U_{chyb} + (U_{chyb} * dt/T_i) + T_d * \text{pochodna}) + \text{odchylenie regulacji}$

Gdzie:

dt = Przyrost czasu

Pochodna =  $(U_{chyb} - \text{Poprzedni uchyb})/dt$

lub

Pochodna =  $(\text{Wartość sygnału sterowanego} - \text{Poprzednia wartość sygnału sterowanego})/dt$

[wybierane przez użytkownika w czasie konfigurowania]

T<sub>i</sub> = Czas całkowania

T<sub>d</sub> = Czas różniczkowania

Jako standard przyjęty jest algorytm o niezależnych wyrazach. Pomimo że obydwa rodzaje bloków PID zapewniają te same wyniki, blok PID-ISA jest łatwiejszy do dostrojenia.

Składnik określający przesunięcie punktu pracy (CVBias) jest dodatkowym składnikiem, niezależnym od pozostałych bloków regulatora PID. Jest on najczęściej wykorzystywany w przypadku stosowania samego bloku proporcjonalnego (K<sub>p</sub>) - regulator proporcjonalny. W efekcie, jeżeli wartość sterowana (PV) jest równa wartości zadanej (SP), wartość sterująca (CV) ma pewną, niezerową wartość. Współczynnik CVBias zwykle nie jest wykorzystywany (ma wartość 0) w przypadku wykorzystywania bloku całkowego.

## **Dostrajanie regulatora PID**

---

Zadaniem regulatora PID jest generowanie odpowiedniej wartości sterującej CV, na podstawie wartości zadanej SP i wartości sterowanej PV tak, aby wartość PV była jak najbliższa lub równa wartości SP. Zadanie to ma być zrealizowane w jak najkrótszym czasie oraz z minimalnymi wahaniami sterowanej wartości.

W celu uzyskania tak określonych celów, regulator PID musi być wcześniej dostrojony. Proces ten polega na dobraniu takich wartości K<sub>p</sub>, K<sub>i</sub>, i K<sub>d</sub>, aby każdy uchyb w procesie był likwidowany w możliwie najkrótszym czasie i możliwie jak najdokładniej. Te dwa cele zwykle wzajemnie się wykluczają. Można szybciej sterować procesem, ale z mniejszą dokładnością, lub wolniej, ale z

większą dokładnością. Wybór odpowiedniego kompromisu oraz odpowiednie dostrojenie bloku PID należy do inżyniera odpowiedzialnego za proces.

Dostrajanie regulatora PID to trudne zadanie. Często stosowana jest metoda "prób i błędów". Po ustawieniu wartości parametrów  $K_p$ ,  $K_i$  i  $K_d$  należy obserwować zachowywanie się procesu po wystąpieniu następnego zakłócenia. Jeżeli proces jest sterowany w sposób spełniający postawione wymagania, należy zakończyć dostrajanie. W przeciwnym wypadku należy zmienić wartości tych parametrów i przeprowadzić następną próbę. Ogólnie ujmując, procedura taka jest bardzo czasochłonna.

Doświadczony inżynier zwykle dobrze zna właściwości procesu, co znacznie pomaga w wyznaczaniu wartości parametrów regulatora PID. Jest to jeszcze łatwiejsze, po poznaniu zależności pomiędzy parametrami  $K_p$ ,  $K_i$  i  $K_d$ .

Upraszczając, głównym czynnikiem decydującym o sterowaniu jest współczynnik wzmocnienia proporcjonalnego  $K_p$ . Wiele procesów można sterować korzystając z samego regulatora proporcjonalnego. Pierwszy krok polega na wyłączeniu sterowania całkowego i różniczkowego i podjęciu próby sterowania procesem za pomocą samego regulatora proporcjonalnego. Stosowanie samego regulatora proporcjonalnego powoduje powstanie odchyłki regulacji, tzn. niewielkiej, względnie stałej różnicy pomiędzy rzeczywistą wartością sterowaną a wartością zadaną. Jeżeli odchyłka regulacji jest niewielka i stała, można ją anulować za pomocą parametru  $CVBias$ . W przeciwnym wypadku, należy ustawić parametr  $CVBias$  na 0 (zero) i podjąć próbę skorygowania różnicy za pomocą sterowania całkowego.

Sterowanie całkowite, ma na celu zlikwidowanie uchybu poprzez dobranie odpowiedniej wartości przesunięcia, w zależności od czasu trwania uchybu. Im dłużej trwa uchyb, tym sterowanie to jest efektywniejsze. Należy zwiększać wartość parametru  $K_i$  (współczynnik wzmocnienia proporcjonalnego), do momentu zminimalizowania lub nawet całkowitego wyeliminowania uchybu.

Stosując te dwa rodzaje sterowania, proporcjonalne i całkowite, można uzyskać zadawalające wyniki w przypadku większości procesów. Niemniej jednak, można czasem napotkać zjawisko "wahania" się wartości sterowanej PV względem żądanego poziomu. Zjawisko to określane jest terminem tłumionych oscylacji. W takim przypadku  $K_p$  należy ustawić tuż pod punktem, w którym zaczynają występować oscylacje procesu i utrata sterowności.

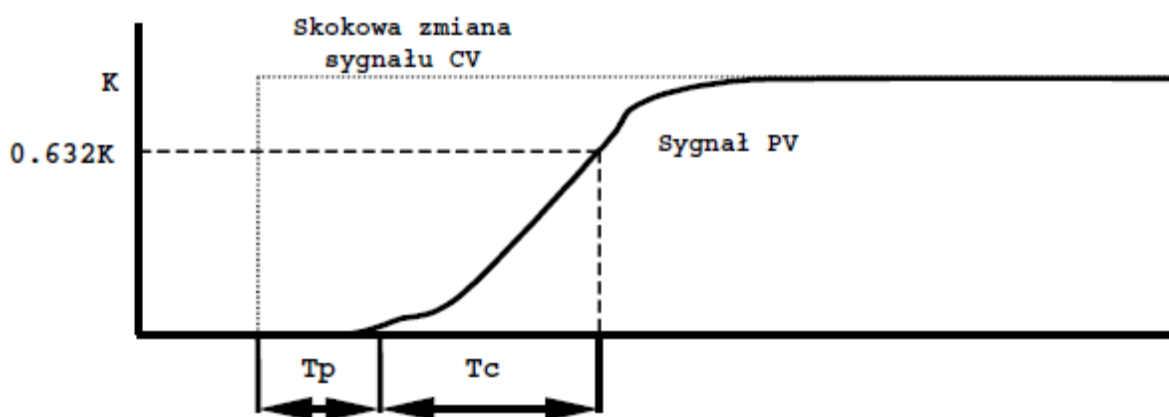
Oscylacje te można w niektórych przypadkach jeszcze bardziej stłumić za pomocą sterowania różniczkowego. Sterowanie różniczkowe uwzględnia szybkość zmian wartości sterowanej PV (a tym samym wynikowego uchybu). Maksymalne natężenie zmian występuje tuż po zakłóceniach, tzn. również w czasie oscylowania sterowania proporcjonalnego. Poprzez zwiększenie wartości  $K_d$  (współczynnik wzmocnienia sterowania różniczkowego) można efektywniej tłumić oscylacje i szybko doprowadzić sterowany proces do żądanego stanu.

Procedura dostrajania bloku PID zależy od wiedzy użytkownika na temat sterowanego procesu. Wartości współczynników  $K_p$ ,  $K_i$  i  $K_d$  zależą od charakterystyk procesu, które należy dokładnie poznać przed rozpoczęciem dostrajania.

Należy zawsze poznać dwie charakterystyki procesu:

Pierwsza z nich określa wielkość zmiany wartości sterowanej PV przy zmianie wartości sterującej CV o stałą wartość.

Druga, określa prędkość zmiany wartości sterowanej PV po zmianie wartości sterującej CV.



Zmiana wartości sterowanej PV jest po prostu mierzona. Dysponując tymi charakterystykami można wyznaczyć na podstawie prostego wzoru wzmocnienie otwartego układu sterowania ( $K$ ):

Wzmocnienie otwartego układu sterowania ( $K$ ) = zmiana wielkości sterowanej PV / zmiana wielkości sterującej CV

Jeżeli zmiana wielkości sterującej CV powoduje identyczną zmianę wielkości sterowanej PV, wzmocnienie otwartego układu sterowania wynosi jeden. Jeżeli zmiana wielkości sterującej CV o pewną wartość powoduje mniejszą od tej wartości zmianę wielkości sterowanej PV, wzmocnienie otwartego układu sterowania  $K$  jest mniejsze od 1. Jeżeli zmiana wielkości sterującej CV o pewną wartość powoduje większą od tej wartości zmianę wielkości sterowanej PV, wzmocnienie otwartego układu sterowania  $K$  jest większe od 1.

W przypadku zdecydowanej większości procesów, wartość sterowana PV jest zmieniana dopiero po upływie pewnego okresu czasu od momentu zmiany wartości sterującej CV. Jest to tzw. opóźnienie procesu  $T_p$  lub czas martwy (nie należy mylić tego parametru z parametrem Dead band (strefa nieczułości)). Stała czasowa procesu  $T_c$  to czas do momentu, kiedy wielkość regulowana PV dochodzi do 63.2% wartości końcowej, przy skokowej zmianie sygnału ustawiającego CV.

### Szukanie K i Tc

W celu uzyskania tych wartości może być konieczne przeprowadzenie eksperymentów. W tym celu należy przełączyć blok PID do trybu ręcznego sterowania (MANUAL), powoli zmieniać wartość sterującą CV i jednocześnie rejestrować zmiany wartości sterowanej PV. W przypadku wolnych procesów czynność ta może być wykonywana ręcznie, natomiast w przypadku procesów szybszych zachodzi potrzeba skorzystania z rejestratora danych lub komputera z odpowiednim oprogramowaniem.

Wielkość zmiany wartości sterującej CV powinna być dostatecznie duża, aby powodowała dającą się zaobserwować zmianę wartości wielkości sterowanej PV, ale nie na tyle duża, aby doprowadzić do przerwania mierzonego procesu.

Uzyskany w ten sposób wykres jest bardzo zbliżony do pokazanego wcześniej. Na jego podstawie można z łatwością wyznaczyć parametry K, Tc i Tp.

### Dostrajanie procesu

Jeżeli znane są wartości parametrów K, Tc i Tp, za pomocą przedstawionych poniżej równań można oszacować wartości początkowe parametrów Kp, Ki i Kd bloku PID.

$$K_p = (1.2 * T_c) / (K * T_p)$$

$$K_i = (0.6 * T_c) / (K * T_p * T_p)$$

$$K_d = (0.6 * T_c) / K$$

Tc i Tp są wyrażone w jednostkach czasu. Należy zwrócić uwagę, aby jednostki te były takie same (np. milisekundy, sekundy, godziny lub inne jednostki, w zależności od sterowanego procesu). W oknie dialogowym oprogramowania Cscape, wartości te muszą być wyrażone jako wielokrotność 10 ms. (np. czas dla „100” jednostek = 10ms \* 100 = 1 sekunda). Jeżeli stosowane będzie wyłącznie sterowanie proporcjonalne (Ki i Kd = 0), należy postąpić następującym równaniem:

$$K_p = T_c / (K * T_p)$$

natomiast dla sterowania proporcjonalno/całkowego (Kd =0), skorzystać z równań:

$$K_p = 0.9 * T_c / (K * T_p)$$

$$K_i = 0.3 * K_p / T_p$$

Równania te znane są jako metoda dostrajania Zieglera i Nicholasa.

## Aktualizowanie oprogramowania systemowego

### Informacje ogólne

W sterownikach Horner APG oprogramowanie systemowe zapisane jest w pamięci flash. Oprogramowanie to można przesłać z poziomu aplikacji Cscape. Mechanizm ten w połączeniu z przesyłaniem plików Internetem lub pocztą elektroniczną pozwala na prawie natychmiastowe uaktualnianie.

Aby zaktualizować system operacyjny należy wybrać z menu *File* polecenie *Update Firmware...* Wyświetlony wtedy zostanie komunikat z pytaniem czy sterownik OCS ma być zatrzymany i przygotowany do rozpoczęcia aktualizacji oprogramowania systemowego. W celu kontynuowania należy kliknąć na przycisk Yes, natomiast w celu przerwania kliknąć na przycisk No.

#### Ostrzeżenie

W czasie aktualizacji oprogramowania systemowego może zostać usunięty program sterujący załadowany do sterownika. Należy się upewnić, czy dostępna jest kopia tego programu.

Jeżeli wciśnięto przycisk Yes, wyświetlone zostanie okno dialogowe Firmware Update. W polu edycyjnym Select Firmware File należy wpisać pełną ścieżkę dostępu do pliku zawierającego nową wersję oprogramowania systemowego. Plik ten można również odnaleźć za pomocą okna dialogowego, wyświetlanego po wciśnięciu przycisku Browse .

#### Uwaga:

Większość plików z oprogramowaniem systemowym dostarczana jest w formacie ASCII-HEX i posiada rozszerzenie .Hex.

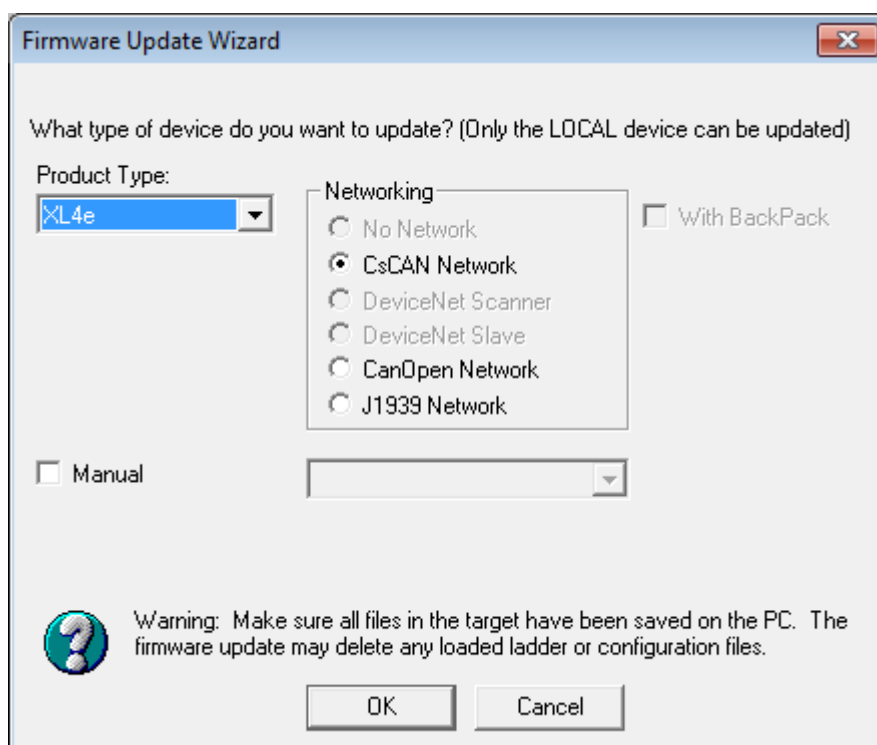
Po wybraniu odpowiedniego pliku należy kliknąć na przycisk Send w celu rozpoczęcia aktualizacji. Jeżeli oprogramowanie systemowe w sterowniku OCS/RCS jest w wersji 7.16 lub nowszej i nie ma zakłóceń komunikacyjnych pomiędzy sterownikiem a oprogramowaniem Cscape, procedura aktualizacji zostanie zrealizowana automatycznie.

Po zakończeniu aktualizacji sterownik zostanie automatycznie zresetowany w celu przygotowania do pracy z nową wersją oprogramowania systemowego.

## Wizard do aktualizacji oprogramowania systemowego

### Korzystanie z wizarda do aktualizacji oprogramowania systemowego

Należy podłączyć sterownik do portu szeregowego komputera. Wybrać z menu File polecenie Update Firmware.... Wyświetlone zostanie następujące okno dialogowe:



*Wizard do aktualizacji oprogramowania systemowego*

W polu Product type powinien zostać pokazany typ sterownika, dla którego uaktualniane będzie oprogramowanie systemowe. Jeżeli tak nie jest, należy wybrać odpowiedni model z rozwijalnej listy, a w grupie Networking wybrać odpowiedni typ sieci. Należy kliknąć na OK.

#### **Uwaga:**

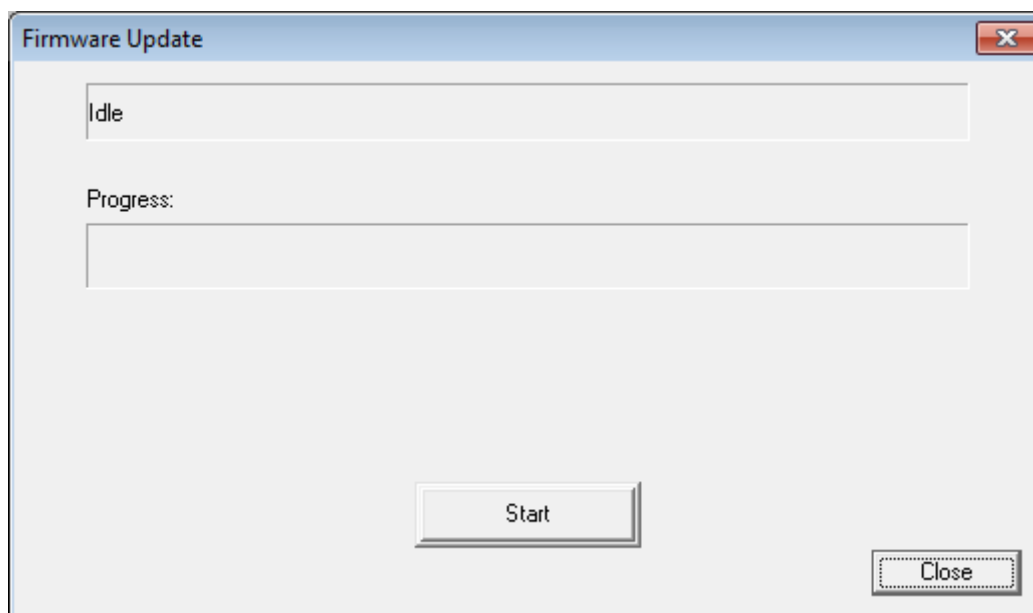
Jeżeli sterownik został zakupiony w wersji bez sieci CsCAN, nie będzie możliwe przesłanie oprogramowania systemowego obsługującego sieć. Jeżeli sterownik został zakupiony bez sieci CsCAN, nie wolno przysyłać oprogramowania systemowego obsługującego sieć.

#### **Uwaga:**

Wizard do aktualizacji oprogramowania systemowego automatycznie wybiera podkatalog Firmware katalogu Cscape. Jeżeli jednak użytkownik przechowuje plik z oprogramowaniem systemowym w innym katalogu, należy kliknąć na przycisk Browse, w celu odszukania tego pliku.



W celu rozpoczęcia procesu aktualizacji należy kliknąć przycisk Send i odczekać do momentu zasygnalizowania zakończenia procesu aktualizacji.



*Wizard do aktualizacji oprogramowania systemowego*

## Ręczna aktualizacja oprogramowania systemowego

Należy wybrać z menu File polecenie Update Firmware.... Wyświetlony zostanie komunikat z zapytaniem, czy sterownik OCS ma być zatrzymany i przygotowany do rozpoczęcia aktualizacji oprogramowania systemowego. W celu rozpoczęcia aktualizacji należy kliknąć na przycisk Yes, natomiast w celu przerwania na przycisk No.

### **Ostrzeżenie:**

W czasie aktualizacji oprogramowania systemowego może zostać usunięty program sterujący załadowany do sterownika. Należy się upewnić, czy dostępna jest kopia tego programu.

Jeżeli wciśnięto przycisk Yes, wyświetlone zostanie okno dialogowe Firmware Update. W polu edycyjnym Select Firmware File należy wpisać pełną ścieżkę dostępu do pliku z oprogramowaniem systemowym. Plik ten można również odnaleźć za pomocą okna dialogowego, wyświetlanego po wciśnięciu przycisku Browse.

### **Uwaga:**

Większość plików z oprogramowaniem systemowym dostarczana jest w formacie ASCII-HEX i posiada rozszerzenie .HEX.

Po wybraniu odpowiedniego pliku kliknąć na przycisk Send w celu rozpoczęcia aktualizacji.

Jeżeli oprogramowanie systemowe w sterowniku OCS/RCS jest w wersji 7.16 lub nowszej i nie ma zakłóceń komunikacyjnych pomiędzy sterownikiem a oprogramowaniem Cscape, procedura aktualizacji zostanie zrealizowana automatycznie.

Po zakończeniu aktualizacji sterownik zostanie automatycznie zresetowany w celu wczytania nowej wersji oprogramowania systemowego.

**Uwaga**

Odpowiedzialność za przesyłanie odpowiedniej wersji oprogramowania systemowego ponosi użytkownik.

### Aktualizacja oprogramowania systemowego w sterowniku OCS250

Sterownik OCS250 posiada dwie jednostki centralne, pracujące równolegle. Z tego powodu należy zaktualizować oprogramowanie systemowe dla każdej z tych jednostek centralnych. Należy postępować zgodnie z zamieszczoną poniżej procedurą:

1. Należy uruchomić wizar do aktualizacji oprogramowania systemowego, postępując zgodnie z zamieszczonym wcześniej opisem należy wybrać model sterownika OCS250, a następnie kliknąć na OK.
2. W oknie dialogowym Firmware Update zaznaczyć opcję Graphics Processor i kliknąć przycisk SEND.
3. Po zakończeniu aktualizacji należy ponownie uruchomić wizar, ponownie wybrać model OCS250 i kliknąć na OK.
4. Tym razem jednak **NIE ZAZNACZAĆ** opcji Graphics Processor, ale od razu kliknąć na przycisk SEND.

## Edytor graficzny

### Wprowadzenie

Jeżeli przy tworzeniu projektu wybrany zostanie sterownik z graficznym ekranem operatorskim, udostępniane jest specjalne środowisko graficzne do projektowania ekranów, różniące się od środowiska do projektowania ekranów w sterownikach z wyświetlaczem tekstowym. Duża rozdzielczość ekranów graficznych pozwala na jednoczesne wyświetlanie większej ilości informacji, w porównaniu z ekranami tekstowymi. Użytkownik posiada do dyspozycji szereg narzędzi, pozwalających na tworzenie dopracowanych, komunikatywnych i intuicyjnych graficznie stron. Można również korzystać ze standardowych obiektów graficznych i ich domyślnych konfiguracji, co pozwala na znaczne zmniejszenie czasochłonności projektowania ekranów graficznych.

W zamieszczonych w dalszej części punktach opisano obiekty i narzędzia do projektowania ekranów graficznych.

### Obiekty

Obiekt to element graficzny pokazywany na ekranie wyświetlacza, zawierający informacje dla operatora i opcjonalnie pozwalający na zmianę tej informacji. Informacje mogą być przedstawiane w formie tekstów (wyświetlanych za pomocą różnych czcionek i różnych kolorach) lub w postaci animowanych ikon, np. rysunku przełącznika. Oprogramowanie zawiera szereg gotowych do użytku obiektów. W czasie projektowania na jednej stronie można umieścić do 50 różnych obiektów, za wyjątkiem niektórych typów obiektów, których liczba jest ograniczona.

Obiekty dostępne w oprogramowaniu narzędziowym dzielimy na obiekty statyczne oraz dynamiczne (animowane). Obiekty statyczne to elementarne obiekty rysunkowe takie jak linie, prostokąty, elipsy i teksty, nie animowane (nie zmieniające się) w czasie wyświetlania na ekranie. Obiekty dynamiczne, jak animowane ikony, mapy bitowe i wartości tekstowe zmieniają swój wygląd oraz zawartość w celu odzwierciedlenia bieżącego statusu wejść/wyjść.

Każdy obiekt posiada szereg parametrów konfiguracyjnych, definiujących funkcjonalność oraz sposób wyświetlania. Przykładowo, obiekt reprezentujący przełącznik, w zależności od ustawienia jego parametrów konfiguracyjnych może reprezentować przełącznik zwierny, przełącznik do zmiany stanu na przeciwny, albo przełącznik do załączenia lub wyłączenia. Dodatkowo, przełącznik ten może być wyświetlany jako wciskany lub przesuwany, można też skonfigurować wyświetlanie krawędzi i opisu.

## Tworzenie i prosta edycja obiektu

W niniejszym punkcie opisano tworzenie, zmianę rozmiarów oraz usuwanie obiektów.

### Tworzenie obiektu

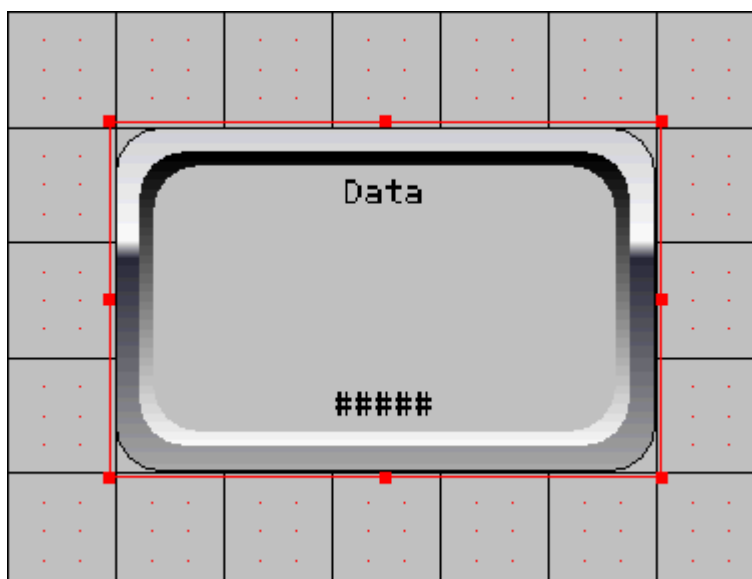
Po otwarciu okna edytora graficznego wyświetlana jest pierwsza strona, do której można od razu wstawiać nowe obiekty. W celu wybrania i umieszczenia obiektu na stronie należy:

- Kliknąć na żądany obiekt na pasku narzędziowym obiektów.
- Kliknąć i trzymając wciśnięty lewy przycisk myszy ustawić kursor w miejscu, w którym ma znajdować się górny, lewy narożnik obiektu.
- Przeciągnąć myszkę w celu zaznaczenia żądanych rozmiarów i kształtu obiektu, po czym zwolnić lewy przycisk myszy.

Obiekt zostanie wyświetlony na ekranie. Należy zwrócić uwagę, że w przypadku włączenia opcji przyciągania do siatki, obiekt może zostać „przyciągnięty” do najbliższego węzła siatki, a więc nie będzie umieszczony dokładnie w miejscu, w którym kliknięto myszą.

### Zaznaczanie obiektu

Po utworzeniu obiektu na stronie jest on automatycznie „zaznaczony”. Zaznaczenie obiektu jest sygnalizowane poprzez umieszczenie go w CZERWONEJ ramce, posiadającej znaczki do zmiany rozmiarów w narożnikach oraz z boku. Ramka ta zostanie usunięta po utworzeniu nowego obiektu lub kliknięciu na inny obiekt.



*Ramka pokazywana dla zaznaczonego obiektu*

Ramka ta sygnalizuje obszar zajmowany przez obiekt. Po umieszczeniu na stronie, domyślnie wokół obiektu wyświetlany jest prostokąt zaznaczający jego obrys. W celu zaznaczenia obiektu do edycji należy kliknąć w dowolnym miejscu wewnątrz tego prostokąta. W przypadku obiektów nakładających się na siebie, należy kliknąć na „widzialną” część obiektu. Jeżeli obiekt jest całkowicie zakryty, można go tymczasowo przenieść na pierwszy plan (aby był widoczny), a następnie zaznaczyć (proszę porównać z punktem Opis narzędzi: To Back).

Wyjątkiem jest zaznaczanie elementarnych obiektów graficznych takich jak okręgi czy prostokąty. Jeżeli taki elementarny obiekt posiada wypełnienie, należy postępować zgodnie z zamieszczonym powyżej opisem. Jeżeli obiekt posiada 'przezroczyste' wypełnienie, użytkownik w celu zaznaczenia musi kliknąć bezpośrednio na krawędzi wyznaczającej obrys obiektu.

## Przesuwanie obiektu

W celu przesuwania obiektów należy:

- Zaznaczyć obiekt.
- Umieścić kursor nad ramką otaczającą obiekt (kursor przybierze kształt czterech strzałek połączonych w jednym punkcie i skierowanych w różne strony).
- Wcisnąć i trzymać lewy przycisk myszy.
- Przesunąć obiekt do nowego miejsca, a następnie zwolnić lewy przycisk myszy.

Zmiana rozmiarów obiektu

W celu zmiany rozmiarów obiektów należy:

- Zaznaczyć obiekt.
- Umieścić kursor nad jednym ze znaczników do zmiany rozmiarów (kursor przyjmie kształt linii zakończonej strzałkami z obu stron).
- Wcisnąć i trzymać lewy przycisk myszy.
- Przeciągnąć punkt do nowego miejsca, a następnie zwolnić przycisk myszy.

## Pokrywanie się obiektów

Obiekty dynamiczne zwykle nie są przezroczyste, a umieszczenie ich na stronie może spowodować zasłonięcie uprzednio wstawionych obiektów. W niektórych przypadkach jest to pożądane, przykładowo przy wyświetlaniu wartości numerycznej nad obiektem reprezentującym miernik. Kolejność, w której obiekty są umieszczane jest taka sama jak kolejność ich tworzenia. Kolejność ta może być zmieniona przez użytkownika (proszę porównać z punktem Opis pasków narzędziowych).

- W celu zmiany tej kolejności należy kliknąć prawym przyciskiem myszy na obiekcie, a następnie wybrać z menu polecenie To Front lub To Back.

## Usuwanie obiektu

W celu usunięcia obiektów należy:

- Zaznaczyć obiekt, a następnie wcisnąć klawisz DEL.

Lub

- Kliknąć prawym przyciskiem myszy w obrębie jego zarysu, a następnie wybrać z menu polecenie DELETE.

## Grupowanie obiektów

---

Obiekty można łączyć w grupy, co pozwala na traktowanie ich jako pojedynczy element. Utworzony w ten sposób element można za pomocą jednego polecenia usunąć, wyciąć, skopiować lub zapisać do pliku. Dodatkowo, wszystkie obiekty w grupie można wyrównać do dowolnie wybranej strony, albo wyrównać ich środki w kierunku poziomym lub pionowym.

### Tymczasowe grupowanie obiektów

W celu tymczasowego grupowania obiektów należy:

- Kliknąć na przycisk Group Selector na pasku narzędziowym Tools.
- Kliknąć lewym przyciskiem i ciągnąć kursor myszy tak, aby zaznaczyć obiekty które mają być połączone w grupę.

Obiekty zostaną tymczasowo połączone w grupę, można je rozłączyć po wybraniu odpowiedniego polecenia z menu rozwijanego po kliknięciu prawym przyciskiem myszy.

### Grupowanie obiektów na stałe

W celu grupowania obiektów należy:

- Utworzyć tymczasową grupę obiektów
- Kliknąć prawym przyciskiem myszy nad tą grupą, a następnie wybrać z menu polecenie Group.

Jeżeli obiekty są zgrupowane na stałe, wybranie dowolnego z obiektów składowych powoduje zaznaczenie całej grupy. Zaznaczona grupa jest otaczana ramką w kolorze czerwonym. W celu rozróżniania grupy od pojedynczego obiektu, ramka ta jest rysowana przerywaną linią.

### Rozłączanie obiektów

W celu rozłączenia obiektów należy:

- Zaznaczyć grupę.
- Kliknąć prawym przyciskiem myszy nad tą grupą, a następnie wybrać z menu polecenie Ungroup.

### **Przesuwanie grupy**

W celu przesuwania grupy obiektów należy:

- Zaznaczyć grupę
- Umieścić kursor w środku grupy tak, aby zmienić kształt na ikonę przesuwania.
- Kliknąć i trzymając wciśnięty lewy przycisk myszy przeciągnąć grupę do nowego miejsca, po czym zwolnić przycisk.

### **Wycinanie, kopiowanie i wklejanie grup**

W celu wycięcia, kopiowania i wklejania obiektów należy:

- Zaznaczyć grupę.
- Kliknąć prawym przyciskiem myszy nad grupą, a następnie wybrać z menu polecenie Cut lub Copy.

Po skopiowaniu grupy do schowka Windows można ją wkleić do dowolnej ze stron, wybierając polecenie Paste z menu rozwijanego po kliknięciu prawym przyciskiem myszy.

### **Usuwanie grupy**

W celu usunięcia obiektów należy:

- Zaznaczyć grupę.
- Wcisnąć klawisz DEL.

Lub

- Kliknąć prawym przyciskiem myszy nad tą grupą, a następnie wybrać z menu polecenie Delete.

### **Zapisywanie grupy**

W celu zapisania grupy obiektów należy:

- Zaznaczyć grupę.
- Kliknąć prawym przyciskiem myszy na tą grupą, a następnie wybrać z menu polecenie Save.
- Wybrać katalog i wpisać nazwę w oknie dialogowym Save As.

Po zapisaniu grupy do pliku można ją wczytać do dowolnego projektu za pomocą polecenia Grouping/Import Group.

### **Importowanie grupy**

W celu zaimportowania grupy obiektów należy:

- Z menu Grouping wybrać Import Group.
- Wskazać plik, który ma być zaimportowany, a następnie kliknąć na przycisk OK.
- Grupa zostanie wstawiona do strony. Jeżeli powinna być umieszczona w innym miejscu, należy na niej kliknąć i przesunąć do nowego miejsca.

Oprogramowanie CsCape dostarczane jest wraz z pewną biblioteką grup. Biblioteka ta zawiera szablony (wirtualne menu, wirtualne panele sterowania) i animowane obiekty (rurociągi, zawory,

pompy, zbiorniki), które można importować, umieszczać w dowolnym miejscu na stronie oraz zmieniać rozmiary. Obiekty te umieszczone są w odpowiednich podkatalogach katalogu otwieranego po wybraniu polecenia Import Group z menu Grouping.

### **Wyrównywanie obiektów w grupie**

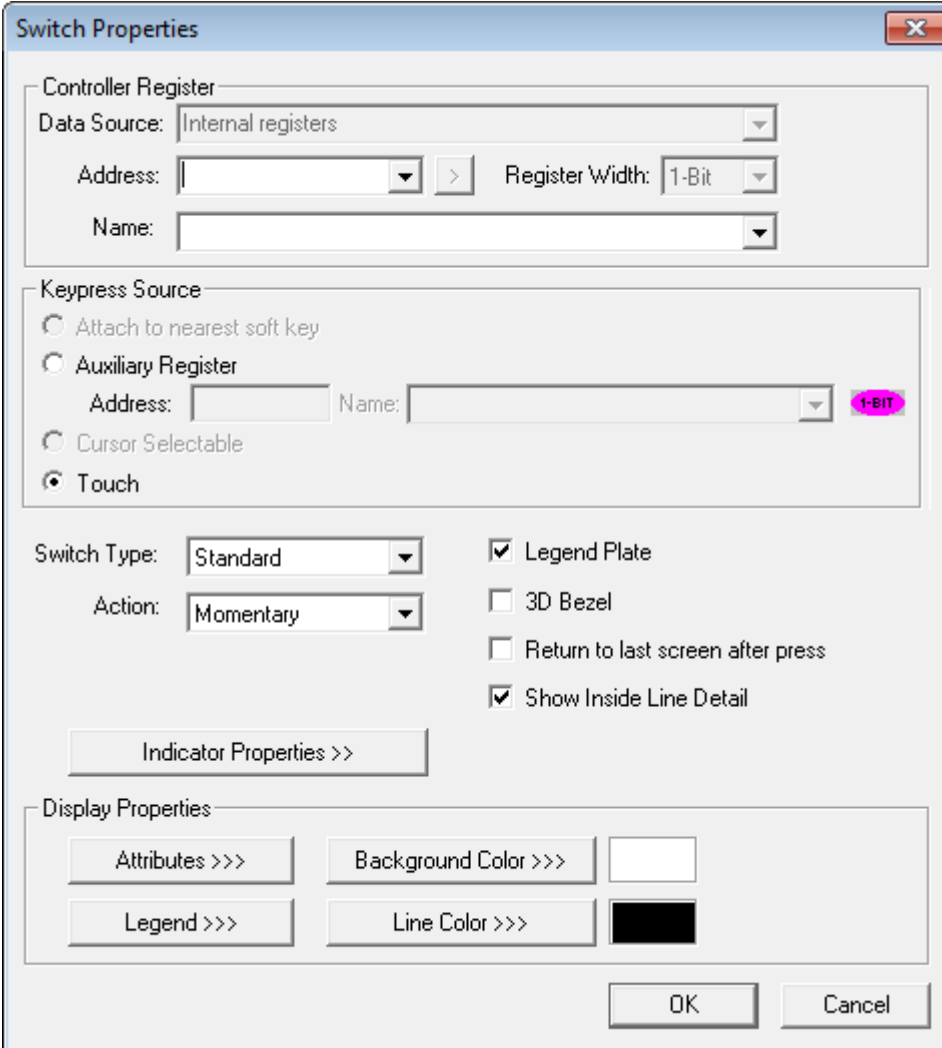
W celu wyrównywania obiektów w grupie należy:

- Zaznaczyć grupę.
- Kliknąć prawym przyciskiem myszy na tą grupę, a następnie wybrać z menu polecenie Align.
- Wybrać z rozwiniętego menu odpowiednie polecenie do wyrównywania.



## Właściwości obiektów

Po wstawieniu obiektu na stronę należy skonfigurować jego właściwości. Właściwości te definiują funkcjonalność oraz format wyświetlania obiektu. Właściwości obiektu są konfigurowane z poziomu okna dialogowego rozwijanego po podwójnym kliknięciu na obiekt.



*Przykład okna dialogowego do definiowania właściwości obiektu*

W zamieszczonych w dalszej części punktach omówiono konfigurowanie właściwości, wspólnych dla wszystkich obiektów. Szczegółowy opis parametrów konfiguracyjnych obiektów zamieszczono w dalszej części podręcznika.

### a. Grupa Controller Register

W grupie tej należy wybrać rejestr sterownika, powiązany z danym obiektem. W zależności od obiektu, rejestr ten może być wykorzystywany do odczytu, zapisu lub zarówno do odczytu, jak i do zapisu. Grupa ta może zawierać maksymalnie trzy pola.

- Źródło danych.  
W naszym przypadku źródłem danych może być pamięć naszego sterownika (Internal registers) lub inne urządzenie podłączone do portu komunikacyjnego sterownika PLC (Driver komunikacyjny skonfigurowany w sterowniku).
- Adres zmiennej adres rejestru (np. %R12).
- Nazwa zmiennej (Np. start\_maszyny)

### b. Grupa Keypress Source

Grupa ta jest wyświetlana dla obiektów, których stan jest sterowany przez operatora (np. poprzez przełączniki). Można wybrać programowalne klawisze ekranowe albo rejestr sterownika (klawisz funkcyjny lub przełącznik zewnętrzny).

Sterowniki nie wyposażone w dotykowy ekran operatorski do nawigacji po ekranie wykorzystują przyciski ekranowe. Programowalne klawisze ekranowe i klawisze oznaczone znakami '>' i '<' umieszczone są z boku wyświetlacza. Po wybraniu opcji, Grupa Keypress Source obiekt zostanie powiązany ze znajdującym się najbliżej programowalnym klawiszem ekranowym. Dodatkowo, obiekt zawiera wskaźnik graficzny, informujący operatora o powiązaniu z konkretnym programowalnym klawiszem ekranowym.

Jeden programowalny klawisz ekranowy może być przypisany na ekranie wyłącznie do jednego obiektu. Z programowalnego klawisza ekranowego można korzystać wyłącznie przy projektowaniu ekranów graficznych.

Alternatywne rozwiązanie polega na powiązaniu obiektu z rejestrem (rejestrami) sterownika, jak na przykład z klawiszami funkcyjnymi (F1-F10) [adresy %K1 - %K10], umieszczonymi poniżej wyświetlacza. Ponieważ jednak klawisze funkcyjne mogą być również wykorzystywane przez program sterujący, należy zwrócić uwagę, aby nie doprowadzić do kolizji.

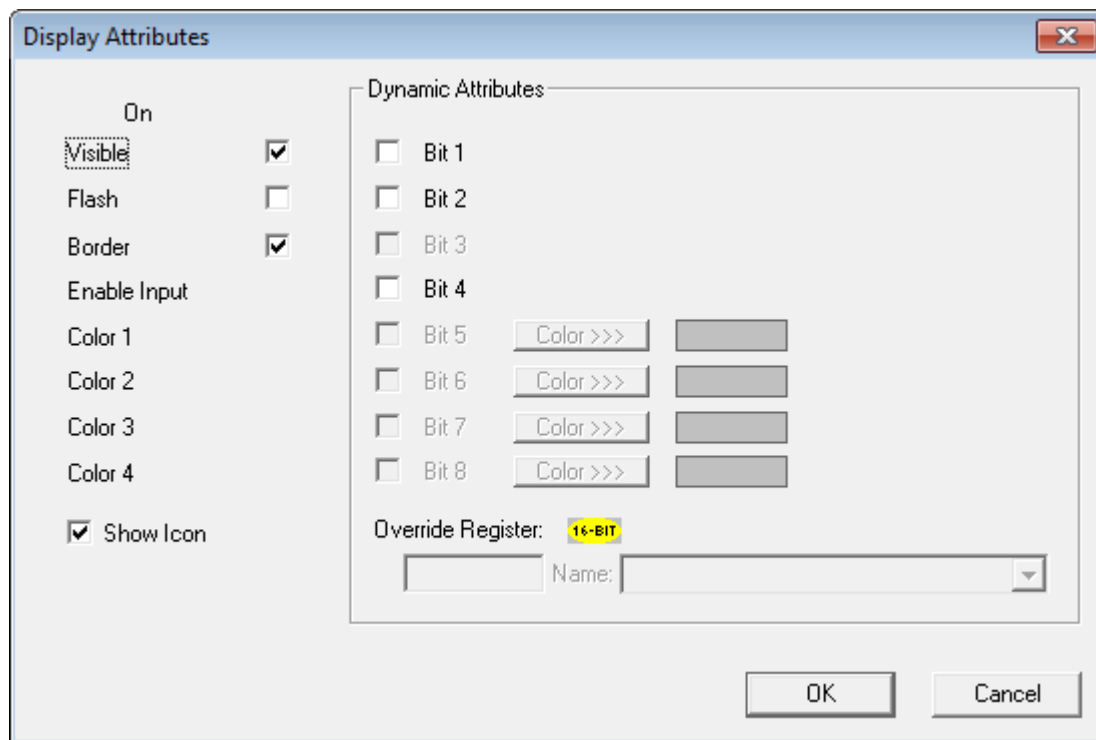
W przypadku niektórych obiektów (np. Screen Jump) można również zaznaczyć opcję „Cursor selectable”. Po zaznaczeniu tej opcji, operator może wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter w celu wykonania działania przypisanego do obiektu. Wybranie obiektu jest sygnalizowane poprzez otoczenie go prostokątem rysowanym przerywaną linią.

### c. Grupa Display Properties

Grupa ta pozwala na skonfigurowanie parametrów określających format wyświetlania, takich jak legenda, rysowanie krawędzi, migotanie czy kolor linii i wypełnienia.

Pewne atrybuty obiektów, takie jak błyskanie czy krawędzie mogą być konfigurowane przez użytkownika, w zależności od jego preferencji. Dodatkowo, niektóre z tych atrybutów mogą być

zmieniane dynamicznie w zależności od wartości zapisanej w podanym rejestrze sterownika lub statycznie, zgodnie z wartością podaną w czasie projektowania. Po kliknięciu przycisku Attributes>>>, na ekranie wyświetlane jest pokazane poniżej okno dialogowe, umożliwiające zmianę parametrów wyświetlania.



*Atrybuty wyświetlania*

### Visible

Obecnie wszystkie obiekty są zawsze widoczne, wskutek czego odpowiednia opcja w grupie atrybutów statycznych i dynamicznych jest zawsze nieaktywna

### Flash

Po zaznaczeniu opcji Flash wyświetlane dane lub ikona animacji będą migotać przez cały czas, gdy przypisany rejestr sterujący ma wartość 1. Po zaznaczeniu opcji w kolumnie Dynamic, można zdefiniować trzy stany wyświetlania: wyłączony, włączony i migotający, wybierane w zależności od stanu rejestru sterującego i rejestru Override.

### Border

Jest to atrybut statyczny, sterujący wyświetlaniem dekoracyjnej ramki (prostokąta) wokół obiektu. Zwykle opcja ta nie jest zaznaczana, bowiem rysowana jest bardziej dekoracyjna ramka za pomocą podstawowych elementów graficznych lub ramka w ogóle nie jest rysowana.

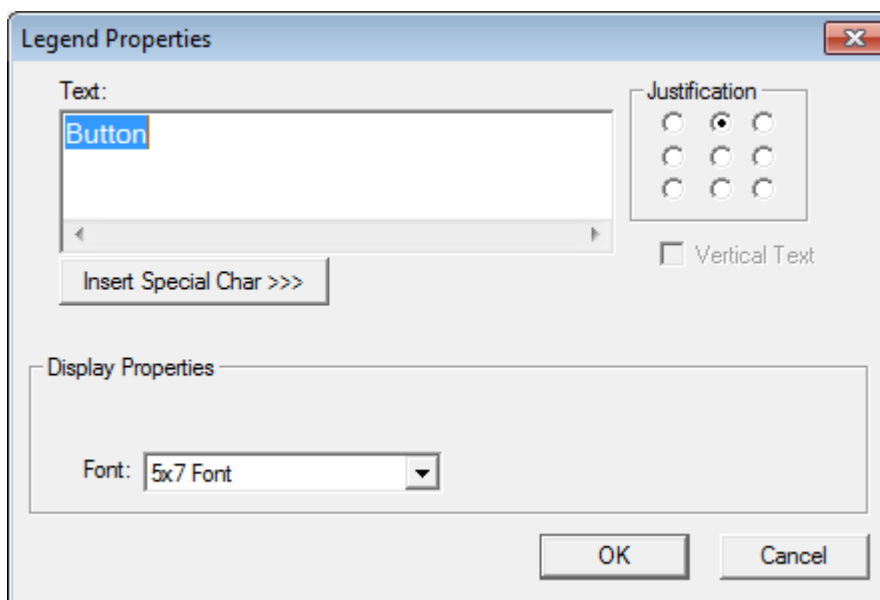
### Enable Input

Jest to atrybut dynamiczny, po jego włączeniu ignorowane są wszelkie polecenia klawiszowe kierowanie do obiektu. Przy pomocy tego atrybutu można zaimplementować mechanizm,

ograniczający prawa do obsługi danego obiektu. Atrybut ten pozwala na ograniczenie uprawnień operatora lub zablokowanie modyfikowania w przypadku ruchu obiektu. Jeżeli opcja ta nie zostanie zaznaczona, będzie możliwe wprowadzenie danych do sterownika za pomocą tego obiektu.

### Show Icon

Jest to atrybut statyczny, dostępny w niektórych obiektach, jak na przykład Switch czy Screen Jump. Po zaznaczeniu tej opcji wyświetlana będzie ikona. Większość obiektów pozwala na zdefiniowanie opisowego tekstu (legendy) wyświetlanego w granicach tego obiektu. Po kliknięciu przycisku Legend>>>, na ekranie wyświetlane jest pokazane poniżej okno dialogowe, umożliwiające zdefiniowanie opisu.



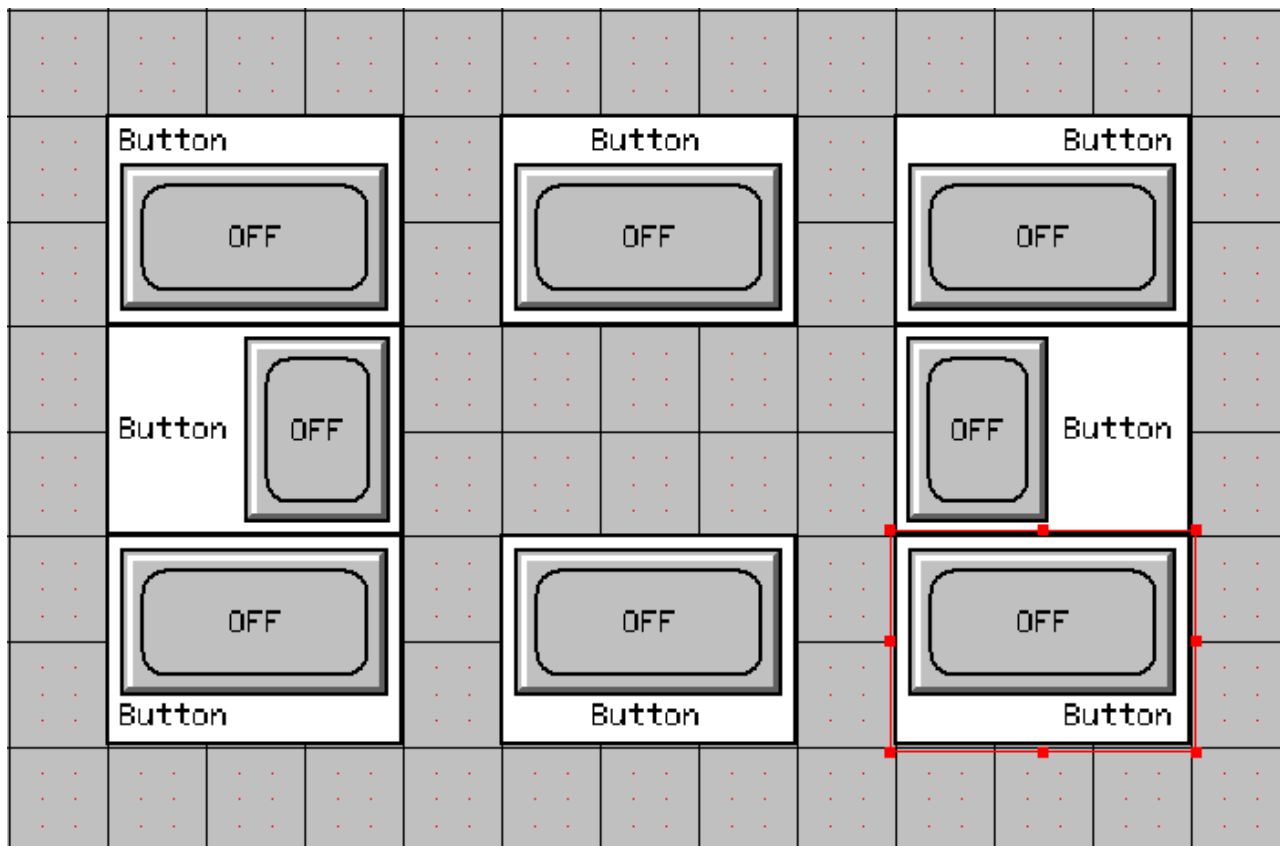
*Definiowanie legendy*

### Text

W polu tym należy wpisać tekst opisu. Wciśnięcie klawisza Enter powoduje przejście do nowej linii. W przypadku tekstów, których długość przekracza szerokość obiektu, tekst jest automatycznie dzielony na linie (nawet jeżeli nie był podzielony poprzez wciskanie klawisza Enter). W przypadku, gdy tekst nie mieści się, jest on obcinany.

### Justification

Grupa ta pozwala na zdefiniowanie położenia tekstu legendy w obrębie obiektu. Poniżej pokazano wygląd przełącznika po wybraniu różnych opcji wyrównania. Jedyna różnica pomiędzy pokazywanymi obiektami polega na ustawieniu opcji wyrównania.



*Przykłady przełączników z różnymi ustawieniami opcji wyrównywania legendy*

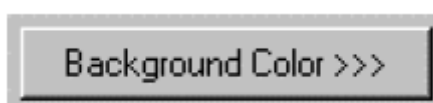
Należy zwrócić uwagę, że animowana część obiektu zmienia rozmiary oraz miejsce w zależności od wybranej opcji wyrównania opisu. Wyrównanie opisu należy dobrać stosownie do przeznaczenia obiektu (np. w obiektach wyświetlających zawartość rejestru sterownika opis powinien być umieszczony na górze lub na dole, w obiektach służących do wprowadzania wartości przez operatora, opis powinien być umieszczony po lewej stronie, a jeżeli opis wykorzystywany jest do wyświetlania jednostek, powinien być umieszczony po prawej stronie).

### Insert Special Chars

Po wciśnięciu tego przycisku wyświetlane jest okno dialogowe ze znakami specjalnymi dla aktualnie wybranej czcionki. Pozwala to na wprowadzenie znaków specjalnych, np. znaku ° (stopnie).

### Font Type

Czcionka tekstu opisu może być inna niż czcionki dla pozostałych tekstów wyświetlanych przez obiekt.



Przycisk ten pozwala na wybranie koloru tła (wypełnienia) obiektu (sterownik OCS250 obsługuje tylko dwa kolory: czarny i biały).



Przycisk ten pozwala na wybranie koloru „pierwszego planu” obiektu tzn. kolor legendy, ramki (o ile jest rysowana) oraz linii animacji (tekstu).

## Mechanizm wyświetlania stron

---

Dla sterownika OCS250 można zaprojektować do 300 ekranów. W danej chwili może być wyświetlany jeden ekran. Sterownik OCS250 posiada mechanizmy pozwalające operatorowi na zmianę aktualnie wyświetlanego ekranu. Mechanizm ten pozwala na przełączanie ekranów za pomocą panelu operatorskiego, programu sterującego lub sieci komunikacyjnej. Dodatkowo dostępne są trzy poziomy priorytetów wyświetlania ekranów, co pozwala na przerwanie wyświetlania aktualnego ekranu użytkownika i przejście do wyświetlania ekranu systemowego lub ekranu alarmów. W niniejszym punkcie opisano numerowanie, atrybuty oraz mechanizmy zmiany ekranów.

### Ekran startowy

Każdy ekran użytkownika ma przypisany numer. Numer ten wykorzystywany jest zarówno do identyfikacji strony w trakcie edycji jak i w trakcie pracy sterownika (pośrednie odwoływanie się do strony). Strony są numerowane od 1 do 300. Pierwsza strona, wyświetlana po włączeniu zasilania oraz po każdym przejściu do trybu pracy RUN jest nazywana stroną startową. Ekran startowy jest również pokazywany w przypadku próby wyświetlenia ekranu nie zdefiniowanego lub podania nieprawidłowego numeru ekranu (klawisz ESC). Domyślnie jako ekran startowy przyjmowany jest ekran o numerze 1, ale można go zmienić na dowolny ekran poprzez wybranie z menu Screens lub z menu rozwijanego po kliknięciu prawym przyciskiem myszy polecenia Set Initial Screen.

### Parametry konfiguracyjne strony

Każdy ekran posiada dwa konfigurowalne parametry: kolor tła i komentarze. Domyślnie jako kolor tła ustawiany jest kolor biały, ale można go zmienić na czarny poprzez wybranie z menu Screens lub z menu rozwijanego po kliknięciu prawym przyciskiem myszy na polecenie Set background. Można również zmienić kolor tła lub kolor pierwszoplanowy poszczególnych obiektów, w celu dostosowania ich do przyjętego schematu kolorów. Dla każdego z ekranów można również wprowadzić „ukryty” komentarz. Komentarz „ukryty” może zawierać uwagi programisty lub być wykorzystywany w formie dokumentacji. W celu wywołania okna dialogowego edytora komentarzy należy z menu Screens wybrać polecenie Comments lub kliknąć na odpowiedni przycisk na pasku narzędziowym.

### Priorytety stron

W mechanizmie wyświetlania ekranów zdefiniowane są trzy poziomy priorytetów: EKRANY UŻYTKOWNIKA, EKRANY ALARMÓW I EKRANY SYSTEMOWE. EKRAN SYSTEMOWY posiada najwyższy priorytet, dzięki czemu wciśnięcie klawisza System zawsze powoduje przerwanie wyświetlania aktualnego ekranu. EKRAN ALARMÓW posiada priorytet o jeden poziom niższy. Z

poziomu programu sterującego można przerwać wyświetlanie ekranu użytkownika za pomocą przekaźnika do wyświetlania ekranu z ustawioną opcją Force Screen. Jeżeli nie jest aktywny EKRAN SYSTEMOWY lub EKRAN ALARMÓW, wyświetlany jest aktualnie EKRAN UŻYTKOWNIKA.

Do wyświetlania ekranów wykorzystywane są 3 rejestry systemowe sterownika (%SR1-3). Rejestr %SR1 steruje ekranami o najniższym priorytecie, czyli EKRANAMI UŻYTKOWNIKA i zawsze powinien zawierać liczbę z zakresu 1 do 300. Zawartość zapisana w rejestrze %SR1 jest modyfikowana przez obiekt Screen Jump oraz przez przekaźnik do wyświetlania ekranów z ustawioną opcją Switch Screen. Po włączeniu zasilania lub po przejściu do trybu pracy RUN, do rejestru tego wpisywany jest numer ekranu startowego. Rejestr %SR2 steruje ekranami o wyższym priorytecie (EKRANAMI ALARMÓW) i powinien zawsze zawierać liczbę z zakresu 1 do 300. Zawartość tego rejestru %SR2 jest modyfikowana przez przekaźnik do wyświetlania ekranów z ustawioną opcją Force Screen. Ekran ALARMÓW przestaje być wyświetlany jeżeli przez powiązany z nią przekaźnik do wyświetlania ekranów nie przepływa sygnał (do rejestru %SR2 wpisywana jest wartość zero). Rejestr %SR3 steruje ekranami o najwyższym priorytecie (EKRANAMI SYSTEMOWYMI) i musi zawsze zawierać wartość 0 lub 1. Zawartość rejestru %SR3 jest modyfikowana po wciśnięciu klawiszy 'systemowych' na panelu w celu wyświetlenia menu systemowego. EKRAN SYSTEMOWY jest usuwany z ekranu po wciśnięciu klawisza ESC (do rejestru %SR3 wpisywana jest wartość zero).

Ogólnie ujmując, rejestry %SR1-3 nie są modyfikowane bezpośrednio przez operatora, ale można je wykorzystywać do monitorowania aktualnie wyświetlanego ekranu. Pozwala to przykładowo na zaimplementowanie w programie sterującym mechanizmu zezwalającego na dostęp do pewnych ekranów wyłącznie po podaniu hasła (poprzez doprowadzanie sygnału do przekaźnika do wyświetlania ekranów, jeżeli w rejestrze %SR1 zapisany jest jeden z numerów blokowanych ekranów).

### **Sterowanie stronami za pomocą obiektu Screen Jump**

Z punktu widzenia operatora, zmiana w sposobie wybierania ekranu do wyświetlania w sterowniku OCS250 jest znaczna, ponieważ nie jest on zmuszony do przewijania listy ekranów za pomocą klawiszy strzałek. Oprogramowanie Cscape – Instrukcja obsługi i programowania pomocą klawiszy kursora. Sterownik OCS250 posiada obiekt Screen Jump, który można powiązać z określonym klawiszem lub punktem wejść/wyjść w celu przejścia do wyświetlania żądanego ekranu. Zapewnia to możliwość szybszego i bezpośredniego przechodzenia do określonego ekranu. Obiekty Screen Jump mogą być wyświetlane pojedynczo lub też być połączone w grupy, co pozwala na utworzenie 'wirtualnego' menu. Menu takie pozwala operatorowi na przewijanie za pomocą klawiszy kursora listy ekranów i wybranie żądanego ekranu poprzez wciśnięcie klawisza Edit/Enter. Przy konfigurowaniu obiektu Screen Jump można podać numer ekranu lub odczytać ten numer z rejestru sterownika, co pozwala na utworzenie mechanizmu wyświetlającego odpowiedni ekran w zależności od kontekstu.

Priorytet ekranu (ekran UŻYTKOWNIKA lub ALARMÓW) nie jest określany w czasie projektowania w edytorze. Priorytet jest ustalany w czasie pracy aplikacji, na podstawie zawartości rejestrów %SR(1-3). Zwykle wszystkie obiekty znajdujące się na ekranie pracują tak samo, bez względu na priorytet ekranu (ekran UŻYTKOWNIKA lub ALARMÓW). Wyjątkiem jest obiekt Screen Jump, który

powoduje zmianę ekranu wyłącznie wówczas jeżeli aktualnie wyświetlany jest ekran UŻYTKOWNIKA. Więcej informacji o obiekcie Screen Jump podano w dalszej części instrukcji.

### **Kolejkowanie ekranów**

Mechanizm zmiany ekranów pozwala na ich kolejkowanie. Przed wykonaniem przejścia do wyświetlania nowego ekranu, obiekt Screen Jump może zapisać swój numer do kolejki. Jeżeli operator zakończy przeglądanie nowo otwartego ekranu i wciśnie klawisz ESC, nastąpi powrót do poprzednio wyświetlanej ekranu. Pozwala to na zamykanie za pomocą klawisza ESC ekranów z dodatkowymi informacjami, komunikatami ostrzegawczymi lub sekwencji 'wirtualnych' menu.

Do kolejki można wstawić maksymalnie 16 numerów ekranów. Po opróżnieniu kolejki, wciskanie klawisza ESC nie powoduje żadnych zmian. Wciśnięcie klawisza ESC powoduje powrót do poprzedniego ekranu wyłącznie w przypadku, gdy ekran ma priorytet ekran UŻYTKOWNIKA. Wciśnięcie klawisza ESC na ekranie posiadającym priorytet ekran ALARMÓW, nie powoduje zmiany ekranu ani nie zmienia zawartości kolejki ekranów. Należy również pamiętać, że kolejka jest automatycznie odświeżana w przypadku zdalnej zmiany rejestru ekranów UŻYTKOWNIKA %SR1 (z poziomu programu sterującego za pomocą przekaźnika do wyświetlania ekranów lub poprzez sieć).

### **Korzystanie z kursora (w trakcie pracy sterownika)**

Dla obiektów Data (Text) zawierających edytowalne dane tekstowe oraz dla obiektach binarnych dla których zaznaczono opcję Cursor selectable wyświetlany jest kursor obiektu. Kursor ten ma postać przerywanej linii rysowanej wokół prostokąta obejmującego aktualnie zaznaczony obiekt. Na ekranie startowym automatycznie wybierany jest obiekt umieszczony w górnym, lewym rogu. Operator może za pomocą klawiszy kursora wybrać inny, wybieralny obiekt. Po wybraniu obiektu, w celu wywołania edytora obiektu należy wcisnąć dodatkowy klawisz. W przypadku obiektów Data (Text), edytor obiektów jest wywoływany po wciśnięciu klawisza alfanumerycznego. Natomiast w przypadku obiektów binarnych należy wcisnąć klawisz Edit/Enter.

Kierunek wybierania obiektów zależy od wciśniętego klawisza kursora. Sterownik OCS250 próbuje wybrać obiekt położony najbliżej w zadanym kierunku. Przy wyznaczaniu obiektu do wybrania, rozważane są wyłącznie te obiekty, których środki znajdują się we wnętrzu kąta  $\pm 45$  stopni, prowadzonego od środka aktualnie zaznaczonego obiektu (w kierunku wskazywanym przez klawisz kursora). Metoda ta pozwala na wybranie dowolnego obiektu, bez względu na jego położenie na ekranie. Niemniej jednak zalecane jest, aby umieszczać obiekty w jednej linii, w kierunku poziomym lub pionowym, w celu uproszczenia operatorowi obsługi. Przy wyborze rozmieszczenia obiektów wybieranych przez operatora należy również pamiętać o odpowiednim dobraniu ich rozmiarów, ponieważ obiekty są wybierane na podstawie położenia ich środków.

Jeżeli obiekty pokrywają się całkowicie lub częściowo, wybrany aktualnie obiekt jest umieszczany na samej górze. Jeżeli w kierunku wskazywanym przez klawisz kursora nie znajduje się żaden obiekt, kursor nie zmienia się. Na ekranach na których nie ma wybieralnych obiektów, kursor ekranu nie jest pokazywany.



## Opis pasków narzędziowych

Edytor posiada 3 różne paski narzędziowe (Tools, Objects i Drawing Primitives). Każdy pasek narzędziowy może być zablokowany lub swobodnie przemieszczany. Pasek narzędziowy można wywołać na ekran i zablokować poprzez wybranie z menu View polecenia Restore. Poniżej zamieszczono krótkie opisy poszczególnych przycisków pasków narzędziowych. Bardziej szczegółowe opisy zamieszczono w dalszych punktach niniejszej instrukcji.

### Pasek narzędziowy Tools (dotyczy programowania modelu OCS250)



*Tabela - przyciski paska narzędziowego Tools*

#### Kursor wybierania:

Kursor wybierania to domyślnie wybierane narzędzie, jeżeli nie został wciśnięty żaden inny przycisk. Kursor wybierania używany jest do zaznaczania, przesuwania i zmiany rozmiarów obiektów.

#### Poprzedni ekran:

Przejdźcie do poprzedniej strony.

#### Kursor wybierania grupy:

Kursor wybierania grupy służy do zaznaczania grupy obiektów. Po zgrupowaniu obiekt (obiekty) można przesuwać, wyrównywać oraz usuwać tak, jakby był to pojedynczy element.

#### Podgląd kilku ekranów (przejdźcie do strony):

Po wciśnięciu tego przycisku na ekranie pokazanych jest 20 pomniejszonych stron. Kliknięcie na jeden z ekranów powoduje otwarcie go. Okno zawierające ekrany można przewijać, co pozwala na wybranie dowolnego z 300 dostępnych ekranów.

#### Symulator operatora:

Po wciśnięciu tego przycisku na ekranie wyświetlana jest ikona „Symulator operatora”. W celu animowania obiektów, włączając w to przyciski, wskaźniki i mierniki, należy przeciągnąć ikonę „Symulator operatora” dożądanego obiektu i kliknąć lewym przyciskiem myszy. W niektórych przypadkach kliknięcie lewym przyciskiem myszy powoduje zmianę stanu obiektu. W innych przypadkach każde kliknięcie lewym przyciskiem myszy powoduje wyświetlenie serii map statycznych, klatka po klatce, co daje w efekcie złudzenie ruchu.

#### Następny ekran:

Przejdźcie do następnej strony.

**Wycinanie/kopiowanie/wklejanie:**

Polecenia tej pozwalają na wycinanie, kopiowanie i wklejanie obiektów lub grup do bieżącej lub nowej strony.

**Podgląd/edycja komentarzy do ekranów:**

Funkcja ta pozwala na wprowadzanie uwag i pytań odnośnie aktualnego ekranu, z przeznaczeniem wyłącznie dla programisty. Komentarze te nie są ani drukowane ani wyświetlane w sterowniku OCS250. Jeżeli dla aktualnego ekranu wprowadzono komentarz, przycisk ten jest animowany (zaczyna migotać), w celu zwrócenia uwagi.

**Przenieś na wierzch:**

W przypadku pokrywania się obiektów funkcja umożliwi przeniesienie zaznaczonego obiektu na wierzch, czyli obiekt taki rysowany jest jako ostatni.

**Sprawdzenie poprawności:**

Zainicjalizowanie procesu sprawdzania, w trakcie którego generowane są ostrzeżenia oraz informacje o błędach.

**Przenieś pod spód:**

W przypadku pokrywania się obiektów, funkcja to umożliwi przeniesienie zaznaczonego obiektu pod spód, czyli obiekt taki rysowany jest jako pierwszy.

**Wyrównywanie do siatki głównej:**

Po wciśnięciu tego przycisku, wstawiony lub przesuwany obiekt jest wyrównywany do linii siatki głównej. Przycisk ten nie dotyczy obiektów „Static Text” i elementarnych obiektów graficznych.

**Powiększanie:**

Powiększenie pokazywanej na ekranie strony sterownika OCS. Na przycisk ten należy klikać do momentu powiększenia do żądanej wielkości.

**Wyrównywanie do siatki pomocniczej:**

Po wciśnięciu tego przycisku wstawiony lub przesuwany obiekt „Static Text” lub element obiekt graficzny są wyrównywane do linii siatki pomocniczej. Jeżeli nie jest wciśnięty przycisk wyrównania do siatki głównej, przycisk ten powoduje również wyrównanie do siatki pomocniczej pozostałych obiektów.

**Pomniejszanie:**

Przycisk ten powoduje pomniejszenie pokazywanej na ekranie strony sterownika OCS. Na przycisk ten należy klikać do momentu pomniejszenia do żądanej wielkości.

## Pasek narzędziowy Object (dotyczy programowania modelu OCS250)



*Tabela - przyciski paska narzędziowego Object*

### **Static Text (Tekst statyczny):**

Obiekt ten pozwala na wyświetlanie tekstu w dowolnym miejscu na ekranie. Tekst wyświetlany jest w sterowniku dokładnie tak, jak na ekranie komputera.

### **Selector Object (Przycisk wielopozycyjny):**

Wyświetla i formatuje wielopozycyjny przycisk, powiązany z rejestrem do zapisywania wartości. Przy pomocy tego obiektu można utworzyć przycisk jedno, dwu, trzy lub cztero pozycyjny. Przyciski tego typu można połączyć z programowalnym klawiszem ekranowym sterownika OCS lub z rejestrem sterownika.

### **Numeric Data (Dane numeryczne):**

Formatuje dane numeryczne odczytywane z określonego rejestru i/lub zapisuje dane do rejestru.

### **Screen Jump (Zmiana strony):**

Obiekt pozwalający na przejście do określonej strony. Obiekt ten można połączyć z programowalnym klawiszem ekranowym sterownika OCS lub z rejestrem sterownika.

### **Time Data (Godzina/data):**

Wyświetlanie godziny/ daty odczytywanej z rejestrów i/lub zapis tych danych do rejestrów.

### **Bar Graph (Wykres słupkowy):**

Wykres słupkowy powiązany z określonym rejestrem. Zakres skali jest wybierany przez użytkownika.

### **Password Data (Hasło):**

Formatowanie hasła zapisywanego do rejestru.

### **Meter (Miernik):**

Miernik powiązany z określonym rejestrem. Zakres skali jest wybierany przez użytkownika.

### **Text Table (Tabela tekstowa):**

Obiekt do tworzenia tabeli tekstowej i formatowania tekstu odczytywanego z rejestru i/lub zapisywania do rejestru.

### **Static Bitmap (Statyczna mapa bitowa):**

Obiekt ten pozwala na wybieranie, kopiowanie i wklejanie mapy bitowej z pliku lub dodanie mapy bitowej do domyślnej kartoteki map bitowych. Umożliwia również edycję map bitowych za pomocą wybranego przez użytkownika edytora map bitowych.

#### **ASCII Data (Dane ASCII):**

Formatowanie tekstu odczytywanego z rejestru i/lub zapis do rejestru.

#### **Animation Object (Obiekt do animowania):**

Obiekt ten umożliwi użytkownikowi skopiowanie i wklejenie 2 lub więcej statycznych map bitowych do serii klatek. Za pomocą takiej operacji, mapy bitowe mogą być animowane w celu sprawienia wrażenia ruchu lub w celu zasygnalizowania zmian statusu. Mapy bitowe nie są animowane za pomocą tego obiektu.

#### **Indicator (Lamp) (Lampka sygnalizacyjna):**

Obiekt w formie lampki sygnalizacyjnej, powiązany z rejestrem. W zależności od ustawienia, na stronie może być pokazywana lampka okrągła, kwadratowa lub żarówka.

#### **Data Trend Box (Wykres trendów):**

Obiekt o tworzenia wykresu trendów, pokazującego przebieg zmian w czasie wartości jednej lub kilku zmiennych. Dostępne są cztery rodzaje trendów. Każdy trend może pokazywać do czterech linii (rejestrów). Dla każdego wykresu trendów należy określić rejestr, wykorzystywany do rozpoczęcia rysowania trendu (Trigger Address).

#### **Switch (Przełącznik):**

Obiekt to tworzenia przycisku powiązanego z rejestrem do zapisu wartości. Pozwala na utworzenie przełącznika standardowego, okrągłego, kwadratowego lub klawiszowego. Przyciski tego typu można połączyć z programowalnym klawiszem ekranowym sterownika OCS lub z rejestrem sterownika.

#### **X-Y Graph (Wykres X-Y):**

Obiekt ten pozwala na utworzenie wykresu X-Y, przedstawiającego przebieg zmian zmiennej względem zmian jednej lub więcej zmiennych. Na wykresie można rysować szereg wartości przy pomocy współrzędnych x-y. Każdy wykres może pokazywać do czterech linii (rejestrów). W czasie konfigurowania należy podać adres Trigger Address, wykorzystywany do zerowania rejestrów i wznowiania rysowania wykresu.

### **Pasek narzędziowy Drawing Primitives (dotyczy programowania modelu OCS250)**



*Tabela – przyciski posk narzędziowego Drawing Primitives*

**Prostokąt:**

Rysowanie prostokątów i kwadratów

**Linia: Rysowanie linii****Elipsa: Rysowanie elips i okręgów.****Zaokrąglony prostokąt:**

Rysowanie prostokątów o zaokrąglonych narożnikach.

**Szczegółowy opis paska narzędziowego Tools**

---

**Wskaźnik wybierania:**

Kursor wybierania to domyślnie wybierane narzędzie, jeżeli nie został wciśnięty żaden inny przycisk. Kursor ten pozwala na wybieranie, przesuwanie i zmianę rozmiarów pojedynczego obiektu lub grupy obiektów.

Po kliknięciu lewym przyciskiem. Wyświetlana jest ramka ze znacznikami do zmiany rozmiarów. Kliknięcie prawym przyciskiem myszy na obiekcie powoduje wyświetlenie na ekranie menu pokazanego poniżej.

Kliknięcie prawym przyciskiem myszy poza obiektem powoduje wyświetlenie na ekranie menu pokazanego poniżej.

W celu przesunięcia obiektu należy kliknąć na nim lewym przyciskiem myszy i trzymając wciśnięty ten przycisk przesunąć obiekt do nowego miejsca.

Kursor wybierania

**Kursor wybierania grupy:**

Kursor wybierania grupy służy do zaznaczania i przesuwania kilku obiektów.

W celu wybrania kilku obiektów należy kliknąć lewym przyciskiem myszy i przeciągnąć kursor tak, aby zaznaczyć obszar obejmujący żądane obiekty. Wokół zaznaczonych obiektów wyświetlona zostanie ramka. W celu usunięcia tej ramki należy kliknąć lewym przyciskiem myszy w dowolnym miejscu na ekranie.

Kliknięcie prawym przyciskiem myszy na obiekcie powoduje wyświetlenie na ekranie menu pokazanego poniżej.

W celu przesunięcia grupy zaznaczonych obiektów należy kliknąć lewym przyciskiem myszy na grupie obiektów i trzymając wciśnięty przycisk należy przesunąć grupę do nowego miejsca.

Kliknięcie prawym przyciskiem myszy poza obiektem powoduje wyświetlenie na ekranie menu pokazanego poniżej.

**Symulator operatora:**

Po wciśnięciu tego przycisku na ekranie wyświetlana jest ikona „Symulator operatora”. W celu animowania obiektów, włączając w to przyciski, wskaźniki i mierniki należy przeciągnąć ikonę „Symulator Operatora” do żądanego obiektu i kliknąć lewym przyciskiem myszy. W niektórych

przypadkach kliknięcie lewym przyciskiem myszy powoduje zmianę stanu obiektu. W innych, każde kliknięcie lewym przyciskiem myszy powoduje wyświetlenie serii map statycznych, klatka po klatce, co daje w efekcie złudzenie ruchu.

**Przenieś na wierzch:**

Po wciśnięciu tego przycisku wybrany obiekt zostanie przeniesiony na wierzch innych obiektów, a więc będzie rysowany jako ostatni. Jeżeli obiekt ten był wcześniej zakrywany przez inne obiekty, zostanie on umieszczony przed nimi.

**Przenieś pod spód:**

Po wciśnięciu tego przycisku wybrany obiekt będzie przeniesiony pod spód innych obiektów, a więc będzie rysowany jako pierwszy. Jeżeli inne obiekty zajmują to samo miejsce na ekranie, może on zostać częściowo lub całkowicie zakryty. Mimo że obiekty zostały przesunięty pod spód, to w czasie jego edycji (po zaznaczeniu go) będzie chwilowo narysowany na wierzchu (nad pozostałymi elementami).

**Powiększanie:**

Przycisk ten powoduje powiększenie pokazywanego na monitorze ekranu sterownika OCS. Na przycisku tym należy klikać do momentu powiększenia do żądanej wielkości.

**Pomniejszanie:**

Przycisk ten powoduje pomniejszenie pokazywanego na monitorze ekranu sterownika OCS. Na przycisku tym należy klikać do momentu pomniejszenia do żądanej wielkości.

**Poprzedni ekran:**

Przejdźcie do poprzedniego ekranu.

**Podgląd kilku ekranów:**

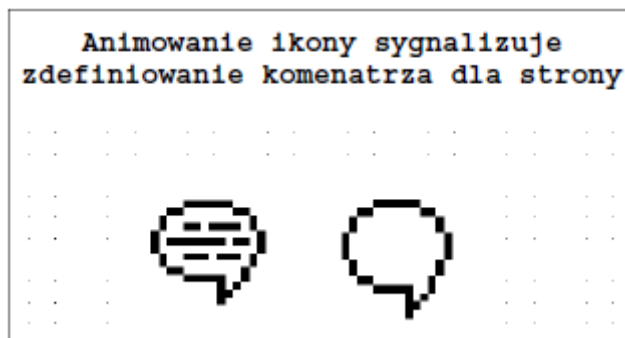
Po wciśnięciu tego przycisku na ekranie pokazywanych jest 20 pomniejszonych ekranów. Kliknięcie jeden z ekranów powoduje przejście do niego. Okno zawierające ekran można przewijać, co pozwala na wybranie dowolnego z 300 dostępnych ekranów.

**Następny ekran:**

Przejdźcie do następnego ekranu.

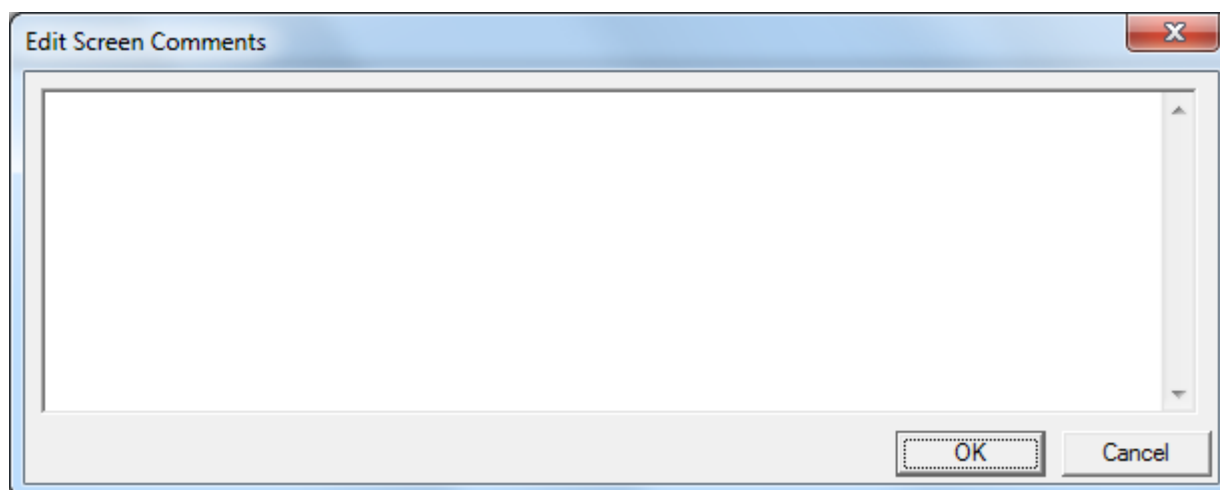
**Podgląd/edycja komentarzy do ekranów:**

Przycisk ten pozwala na wprowadzanie uwag odnośnie aktualnie projektowanego ekranu, przeznaczonych wyłącznie dla programisty. Komentarze nie są nigdy wyświetlane na ekranie i są niewidoczne dla operatorów. Po wprowadzeniu komentarzy, ikona przycisku jest animowana (migocze). Sygnalizuje to, że dla danej strony wprowadzono komentarz.



*Przycisk Podgląd/edycja komentarzy*

Po wciśnięciu przycisku podglądu/edycji komentarzy na ekranie wyświetlane jest pokazane poniżej okno dialogowe. W celu usunięcia komentarzy należy usunąć w tym oknie cały wprowadzony tekst.



*Okno dialogowe Edit Screen Comments*

### **Sprawdzenie poprawności:**

Przycisk ten pozwala na sprawdzenie poprawności parametrów konfiguracyjnych obiektów graficznych, bez wychodzenia z edytora graficznego.

### **Wyrównanie do siatki głównej:**

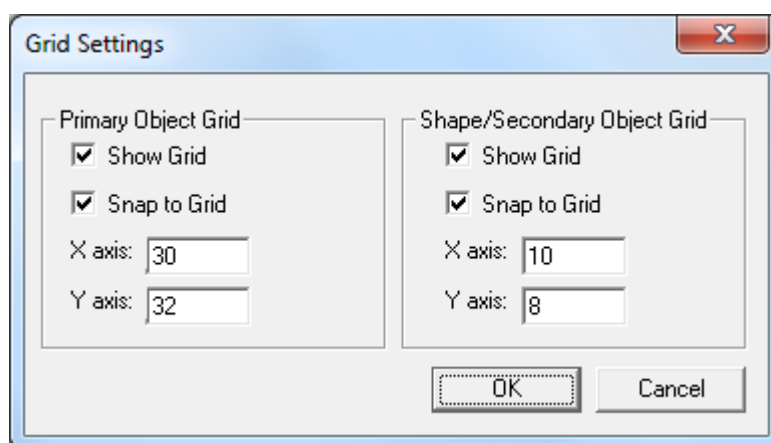
Po wciśnięciu tego przycisku, wstawiony lub wyrównywany obiekt jest przesuwany do linii siatki głównej. Nie dotyczy to obiektów „Static Text” oraz podstawowych elementów graficznych. Domyślna wielkość oczek siatki głównej jest dobrana proporcjonalnie do programowalnych klawiszy ekranowych umieszczonych z boku wyświetlacza.

### **Wyrównanie do siatki pomocniczej:**

Opcja ta dotyczy obiektów „Static Text” i podstawowych elementów graficznych. Po wciśnięciu tego przycisku wstawiony lub przesuwany obiekt „Static Text” lub podstawowy element graficzny

jest wyrównywany do linii siatki pomocniczej. Jeżeli nie jest wciśnięty przycisk wyrównywania do siatki głównej, przycisk ten powoduje również wyrównywanie do siatki pomocniczej pozostałych obiektów.

Przyciski „Wyrównywanie do siatki głównej” i „Wyrównywanie do siatki pomocniczej” ułatwiają dokładne umieszczanie obiektów. Działanie tych opcji zależy od ustawionych parametrów siatki. Obiekty są wyrównywane do górnej i lewej linii siatki. W celu zmodyfikowania parametrów siatki głównej i pomocniczej należy z menu View wybrać polecenie Grid Settings.



*Okno dialogowe do modyfikowania parametrów siatki Grid Settings*

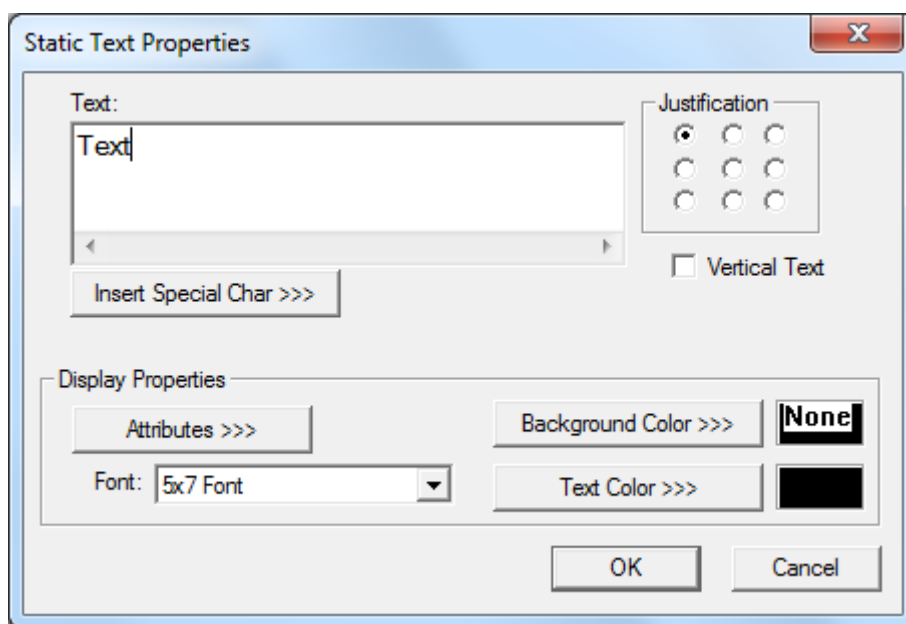
Siatka główna jest zaznaczana na ekranie za pomocą linii, a siatka pomocnicza za pomocą punktów. Pola „X axis” i „Y axis” określają wielkość oczek siatki w pikselach (bez uwzględniania początkowego piksela siatki). Opcja „Show grid” pozwala na wyświetlenie/ukrycie siatki na ekranie. Działanie opcji „Snap to grid” jest identyczne jak omawianych przycisków wyrównywania do siatki.



## Szczegółowy opis obiektów

### **Static Text (dotyczy modelu OCS250):**

Obiekt ten pozwala na wyświetlanie tekstu w dowolnym miejscu na ekranie.



*Właściwości obiektu Static Text*

Opis wybranych parametrów:

- **Text**

Tekst wyświetlany na stronie. Jest on automatycznie dzielony na linie. W przypadku, gdy rozmiary obiektu są zbyt małe, część tekstu będzie niewidoczna.

- **Justification**

Opcje te określają wyrównanie tekstu we wnętrzu obiektu. Działanie tych opcji jest analogiczne do działania w przypadku wyrównywania legendy.

- **Vertical Text**

Po zaznaczeniu tej opcji, tekst zostanie umieszczony w pionie (każdy znak umieszczany jest w osobnej linii).

- **Insert Special Char**

Przycisk ten pozwala na wstawianie znaków specjalnych, dostępnych w aktualnie wybranej czcionce, nie wchodzących w skład zestawu znaków ANSI.

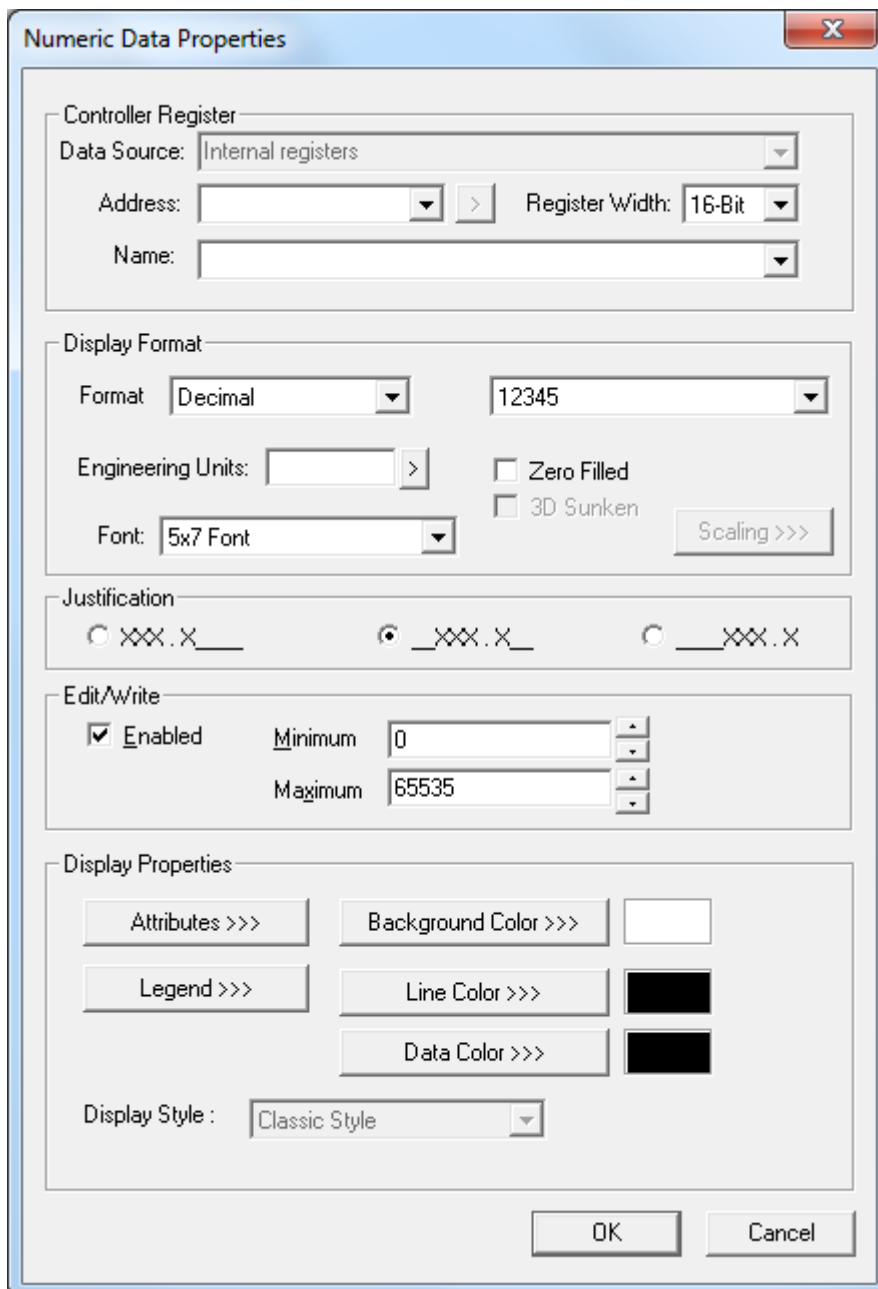
- **Background Color**

Po kliknięciu na ten przycisk, dla obiektów „Static Text” wyświetlany jest dodatkowy kolor, oznaczony symbolem 'Tran' (Transparent-przeźroczysty). Po zaznaczeniu tej opcji i umieszczeniu

obiektu tekstowego nad innymi obiektami, na stronie rysowany będzie sam tekst. Cały obiekt NIE BĘDZIE więc wypełniany kolorem tła.

***Numeric Data (Dane numeryczne) (dotyczy modelu OCS250):***

Formatowanie danych numerycznych odczytywanych z określonego rejestru i/lub zapisywanie danych do rejestru.



*Właściwości obiektu Numeric Data*

Opis wybranych parametrów:

- **Justification**

Grupa ta służy do określenia wyrównania wyświetlanej wartości numerycznej w obszarze zajmowanym przez obiekt. Liczba wyświetlanych cyfr powinna być mniejsza od liczby cyfr zarezerwowanych do tego celu. W zależności od wybranej opcji, cyfry mogą być ustawiane na środku lub wyrównywane do lewej krawędzi ekranu.

- **Font**

Pole to pozwala na wybór czcionki do wyświetlania wartości numerycznej.

- **Format**

Pole to określa format wyświetlania wartości numerycznej.

- Binary - każda cyfra reprezentuje stan binarny odpowiedniego bitu wartości wejściowej.
- Decimal - zawartość rejestru jest wyświetlana jako wartość bez znaku.
- Signed Decimal - zawartość rejestru jest wyświetlana jako wartość ze znakiem.
- Hexadecimal - zawartość rejestru jest wyświetlana w formacie heksadecymalnym [0- F]
- Real/Floating Point - zawartość rejestru jest wyświetlana jako liczba zmiennoprzecinkowa [xx.xx] (musi to być rejestr 32-bitowy, zawierający liczbę w formacie zmiennoprzecinkowym IEEE).
- Scientific Notation - zawartość rejestru jest wyświetlana w notacji naukowej [x.xxe±xx].

- **Digits**

Pole to określa maksymalną liczbę wyświetlanych cyfr. Jeżeli liczba cyfr wyświetlanej wartości jest większa od podanej w tym polu, pokazany zostanie omówiony poniżej wskaźnik przekroczenia zakresu. Należy zwrócić uwagę na dobranie odpowiednio dużych rozmiarów obiektu, tak aby mogła być wyświetlana określona liczba cyfr. Wielkość obiektu można wyznaczyć porównując liczbę wyświetlanych w edytorze znaków '#' z określoną w tym polu liczbą cyfr.

- **Decimal Place**

Po wybraniu formatu dziesiętnego wartość wyświetlana jest wraz z symbolem dziesiętnym (.) (nie dotyczy wyświetlania wartości numerycznych innych niż „real” i „floating point” oraz „Scientific Notation”).

W przypadku liczb zmiennoprzecinkowych, pole to określa liczbę wyświetlanych cyfr po symbolu Liczba wyświetlanych cyfr części całkowitej jest równa liczbie wprowadzonej w polu „Digits” minus liczba cyfr części dziesiętnej. W przypadku liczb w notacji naukowej, pole to określa liczbę wyświetlanych cyfr po symbolu.

Maksymalna liczba cyfr, którą można wprowadzić musi być mniejsza od liczby wprowadzonej w polu „Digits” minus liczba cyfr części dziesiętnej.

Liczba wyświetlanych cyfr liczby całkowitej jest zawsze równa 1. Wykładnik zawsze składa się z dwóch cyfr.

- **Zero Fill**

Po zaznaczeniu tej opcji, wyświetlona wartość zostanie poprzedzona odpowiednią liczbą zer.

- **Editable**

Po zaznaczeniu tej opcji, za pomocą obiektu będzie można zmieniać wartość numeryczną zmiennej wyświetlanej na ekranie. Pola Min i Max i pozwalają na określenie przedziału, w którym musi się zawierać wartość wprowadzana przez użytkownika.

- **Engineering units**

W polu tym można wprowadzić jedną linię tekstu podającą stosowane jednostki dla danej zmiennej (np. o C, mV, kg).

Charakterystyka działania obiektu:

- **Controller register**

Obiekt ten może być powiązany z rejestrem dowolnego typu i o dowolnej wielkości. Dla rejestrów binarnych (%I, %Q, itp.), w polu „Register Width” musi być wybrana pozycja 1-bit. Pole „Register Width” określa liczbę bitów wykorzystywanych do wyświetlania wartości analogowej. 15  
Oprogramowanie Cscape – Instrukcja obsługi i programowania

- **Funkcjonalność**

Zawartość rejestru sterownika OCS jest odczytywana i aktualizowana na ekranie co 150ms. Jeżeli zaznaczono opcję „Editable”, wartość wprowadzona przez użytkownika jest zapisywana do rejestru sterownika. Jeżeli wartość nie może być wyświetlona zgodnie z określonym formatem, w miejscocyfr wyświetlane są następujące znaki:

- ‘+’ Jeżeli wartość posiada większą liczbę cyfr niż zdefiniowana.
- ‘>’ Jeżeli wartość jest większa od maksymalnej (dotyczy wyłącznie obiektów dla których zaznaczono opcję Editable).
- ‘<’ Jeżeli wartość jest mniejsza od minimalnej (dotyczy wyłącznie obiektów dla których zaznaczono opcję Editable).
- ‘e’ Jeżeli wartość rzeczywista jest nieskończenie duża lub równa NAN (nieokreślona).

- **Object Editor (po zaznaczeniu opcji Editable)**

Obiekt ten obsługuje dwa tryby edycji wartości: tryb WSTAWIANIA i tryb NADPISYWANIA. W celu wywołania edytora obiektu:

Należy wybrać obiekt za pomocą klawiszy kursora (zostanie on otoczony ramką rysowaną przerywaną linią), a następnie wcisnąć klawisz Edit/Enter. Niewykorzystywane cyfry zostaną zamienione na zera, a całe pole będzie podświetlone (tryb WSTAWIANIA).

Wciśnięcie dowolnego klawisza w trybie WSTAWIANIE spowoduje usunięcie aktualnej wartości i zastąpienie jej przez wartość przypisaną do wciśniętego klawisza. Każda następną wprowadzana cyfra spowoduje przesunięcie poprzednich cyfr w lewo.

W celu przejścia do trybu NADPISYWANIE należy wcisnąć klawisz kursora w lewo lub w prawo. Nie będzie wtedy podświetlana cała wartość, ale wyłącznie pojedynczy znak.

W trybie NADPISYWANIE podświetloną cyfrę można zwiększać/zmniejszać za pomocą klawiszy kursora w górę/w dół lub zmienić poprzez wciśnięcie odpowiedniego klawisza numerycznego. Zmniejszanie/zwiększanie cyfry może spowodować zmianę cyfr poprzedzających.

W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter, niezależnie od trybu edycji. W celu anulowania zmiany należy wcisnąć klawisz ESC, co spowoduje przywrócenie poprzedniej wartości. Bez względu na wciśnięty klawisz, nastąpi wyjście z trybu edycji i zaprzestanie podświetlania wartości.

#### **Uwagi odnośnie edycji wartości heksadecymalnych:**

Wszystkie klawisze, za wyjątkiem klawiszy '2ABC' i '3DEF' są wykorzystywane zgodnie z opisem przedstawionym powyżej. Każde wciśnięcie klawisza '2ABC' lub '3DEF' powoduje natomiast wyświetlanie kolejnych znaków przypisanych do klawisza (np. 2-A-B-C).

Po wybraniu za pomocą klawisza '2ABC' lub '3DEF' odpowiedniej cyfry, w celu przejścia do następnego pozycji, należy wcisnąć klawisz z kropką.

#### **Uwagi odnośnie edycji liczb zmiennoprzecinkowych oraz liczb w formacie naukowym:**

W przypadku tych zmiennych obsługiwany jest wyłącznie tryb WSTAWIANIE. W żadnym z pól (część całkowita, dziesiętna i wykładnik), operator nie ma możliwości wprowadzenia za dużej liczby cyfr.

W celu wpisania liczby zmiennoprzecinkowej należy:

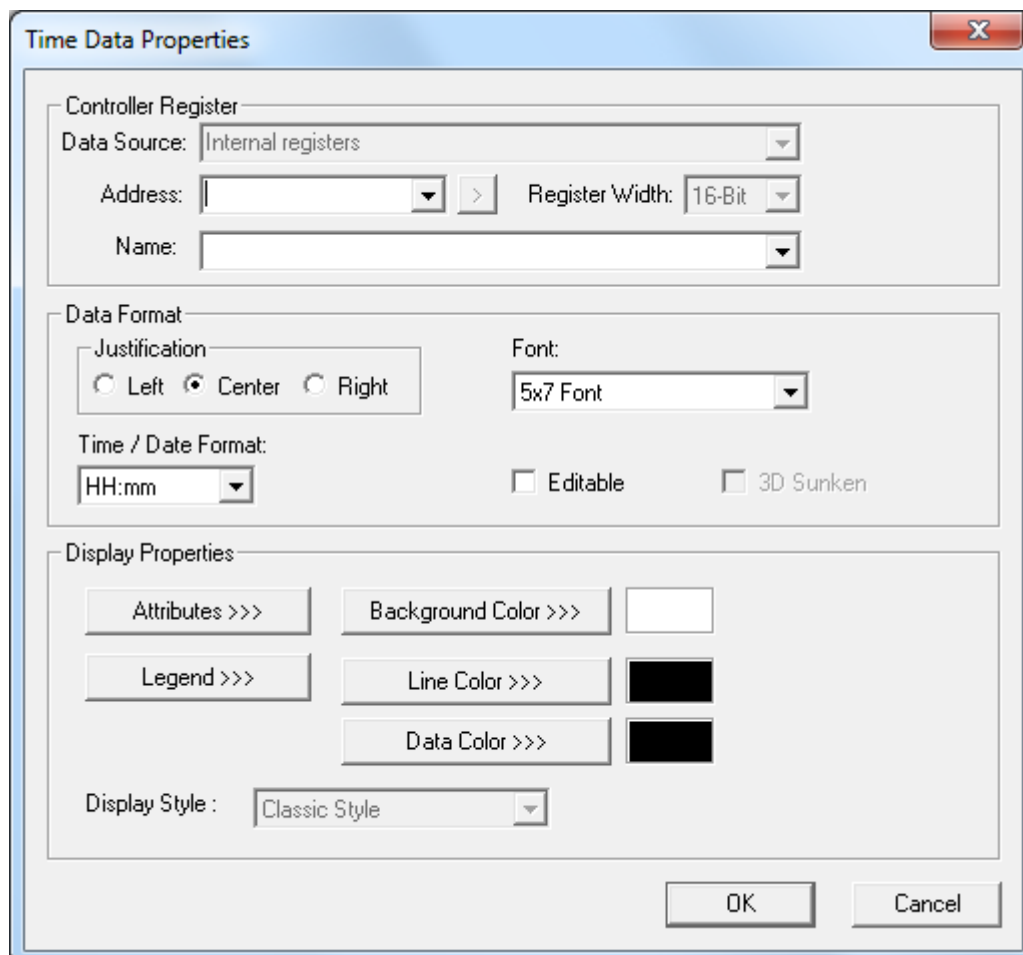
- wpisać cyfry części całkowitej,
- wpisać znak części dziesiętnej,
- wpisać cyfry części dziesiętnej.

W celu wpisania liczby w notacji naukowej należy:

- wpisać cyfry części całkowitej,
- wpisać znak części dziesiętnej,
- wpisać cyfry części dziesiętnej,
- wcisnąć ponownie klawisz z kropką (.) w celu wyświetlenia symbolu 'E',
- wpisać cyfry wykładnika.

Opcjonalne atrybuty wyświetlania:

- Border - statyczny (otoczka elementu),
- Flash (statyczny lub dynamiczny - migotanie),
- Enable Input - (dynamiczny - możliwość edycji tej zmiennej przez operatora).

**Time Data (Godzina/data): Formatowanie czasu zapisywanego do rejestru.**


*Właściwości obiektu Time (/Data)*

Opis wybranych parametrów:

- **Justification**

Grupa służy do określenia wyrównania wyświetlanego czasu lub daty w obszarze zajmowanym przez obiekt.

- **Font**

Pole to pozwala na wybór czcionki do wyświetlania czasu lub daty.

- **Data Format**

Grupa ta zawiera opcje definiujące sposób wyświetlania czasu lub daty. Z rozwijalnej listy Time/Date Format można wybrać jeden z formatów. Poniżej opisano skrótowe oznaczenia:

HH	= Godzina (w trybie 24 - godzinnym)	mmm	= miesiąc [Jan - Dec]
Hh	= Godzina (w trybie 12 - godzinnym)	mm	= miesiąc [01 - 12]
Mm	= Minuty	dd	= dzień [01 - 31]
Ss	= Sekundy	yy	= rok [96 - 95]
xM	= wskaźnik pory dnia AM/PM	yyyy	= rok [1996 - 2095]

- **Editable**

Po zaznaczeniu tej opcji, obiekt będzie można wybierać w trakcie pracy sterownika i zmieniać czas lub datę.

Charakterystyka działania obiektu:

- **Controller register (adres początkowy 3 kolejnych rejestrów)**

Jako adres można podać jeden z rejestrów systemowych (%SR44 - zegar systemowy, %SR47 – data systemowa) lub dowolny adres będący wielokrotnością 16 bitów plus 1, określający obszar, w którym kolejne 48 bitów jest wykorzystywanych do przechowywania daty/czasu.

	Format dla czasu:	Format dla daty
rejestr 16+ 0:	sekundy	dzień miesiąca
rejestr 16+ 1:	minuty	miesiąc
rejestr 16+ 2:	godziny	rok

- **Funkcjonalność**

Zawartość rejestrów z czasem/datą jest odczytywana co 150 ms i wyświetlana na ekranie. Jeżeli zaznaczono opcję 'Editable', wartość wprowadzona przez użytkownika może być zapisywana do rejestru sterownika po wejściu operatora w tryb edycji.

- **Object Editor (po zaznaczeniu opcji Editable)**

Należy wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter.

Podświetlone zostanie pierwsze z pól (tryb WYBIERANIE).

Za pomocą klawiszy kursora w lewo i w prawo wybrać pole, które ma być zmodyfikowane (np. godziny, minuty, itp.).

Za pomocą klawiszy kursora w górę/w dół można zmniejszyć lub zwiększyć wartość (NIE JEST możliwe bezpośrednie wpisanie wartości).

W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter, niezależnie od trybu edycji. W celu anulowania wprowadzonych zmian należy wcisnąć klawisz ESC, spowoduje to przywrócenie poprzedniej wartości.

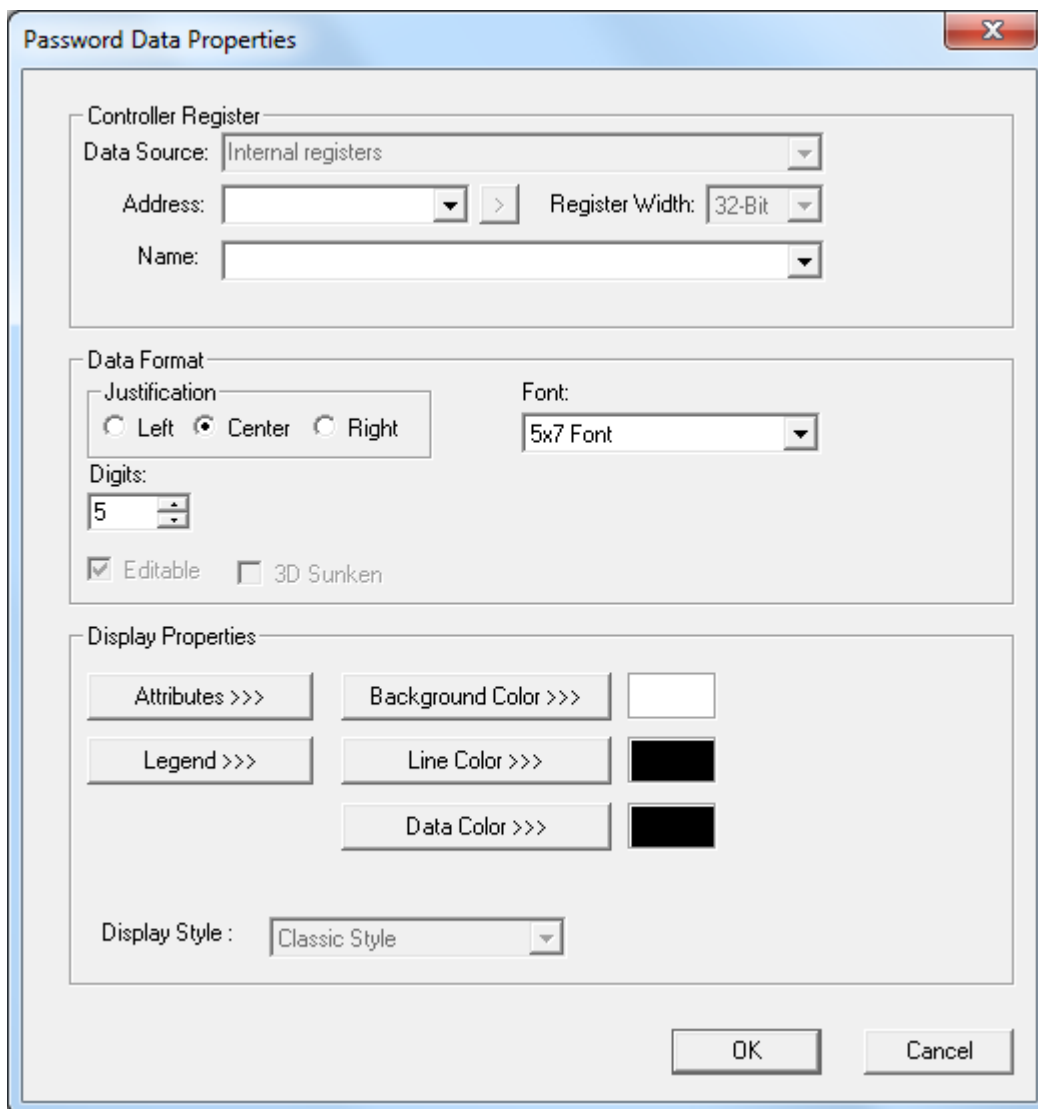
Bez względu na wciśnięty klawisz, nastąpi wyjście z trybu edycji i usunięcie podświetlenia bieżącego czasu lub daty.

**Opcjonalne atrybuty wyświetlania:**

- Border (ramka) - statyczny
- Błyskanie - statyczny lub dynamiczny
- Enable Input - dynamiczny

**Password Data (Hasło):**

Obiekt ten pozwala na wpisanie 32-bitowej wartości do rejestru sterownika, z maskowaniem znaków (tzn. podczas wprowadzania hasła, będzie ono niewidoczne na ekranie panelu).



*Właściwości obiektu Password Data*



Opis wybranych parametrów:

- **Justification**

Grupa służy do określenia wyrównania hasła w obszarze zajmowanym przez obiekt.

- **Font**

Pole to pozwala na wybranie czcionki używanej do wyświetlania znaku maskowania '\*' (gwiazdki).

- **Digits**

Pole to określa maksymalną, dopuszczalną liczbę znaków hasła.

Charakterystyka działania obiektu:

- **Controller Register**

Adres pod który będzie wprowadzane hasło musi być wielokrotnością 16 bitów plus 1. Obiekt ten wykorzystuje 32 bity.

- **Funkcjonalność**

Obiekt ten pozwala na wprowadzenie hasła w postaci 32-bitowej wartości całkowitej bez znaku, z maskowaniem wprowadzanych cyfr. W czasie wpisywania wyświetlane są znaki '\*'.

Po zakończeniu edycji w miejsce wszystkich cyfr pokazywane są znaki '\*', chyba że nastąpiło przekroczenie dopuszczalnej liczby cyfr. W przypadku przekroczenia dopuszczalnej liczby cyfr, wyświetlane są znaki '+'.  
W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter. W celu anulowania zmiany należy wcisnąć klawisz ESC, co spowoduje przywrócenie poprzedniej wartości. Bez względu na wciśnięty klawisz, nastąpi wyjście z trybu edycji i zaprzestanie podświetlania wartości.

- **Sposób wprowadzenia hasła do sterownika**

Aby wprowadzić hasło na panelu należy wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter.

Podświetlone zostanie całe pole (tryb WSTAWIANIE).

Wpisać nową wartość za pomocą klawiszy numerycznych (w miejsce wprowadzanych cyfr będą pokazywane znaki '\*').

W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter. W celu anulowania zmiany należy wcisnąć klawisz ESC, co spowoduje przywrócenie poprzedniej wartości. Bez względu na wciśnięty klawisz, nastąpi wyjście z trybu edycji i zaprzestanie podświetlania wartości.

**Opcjonalne atrybuty wyświetlania:**

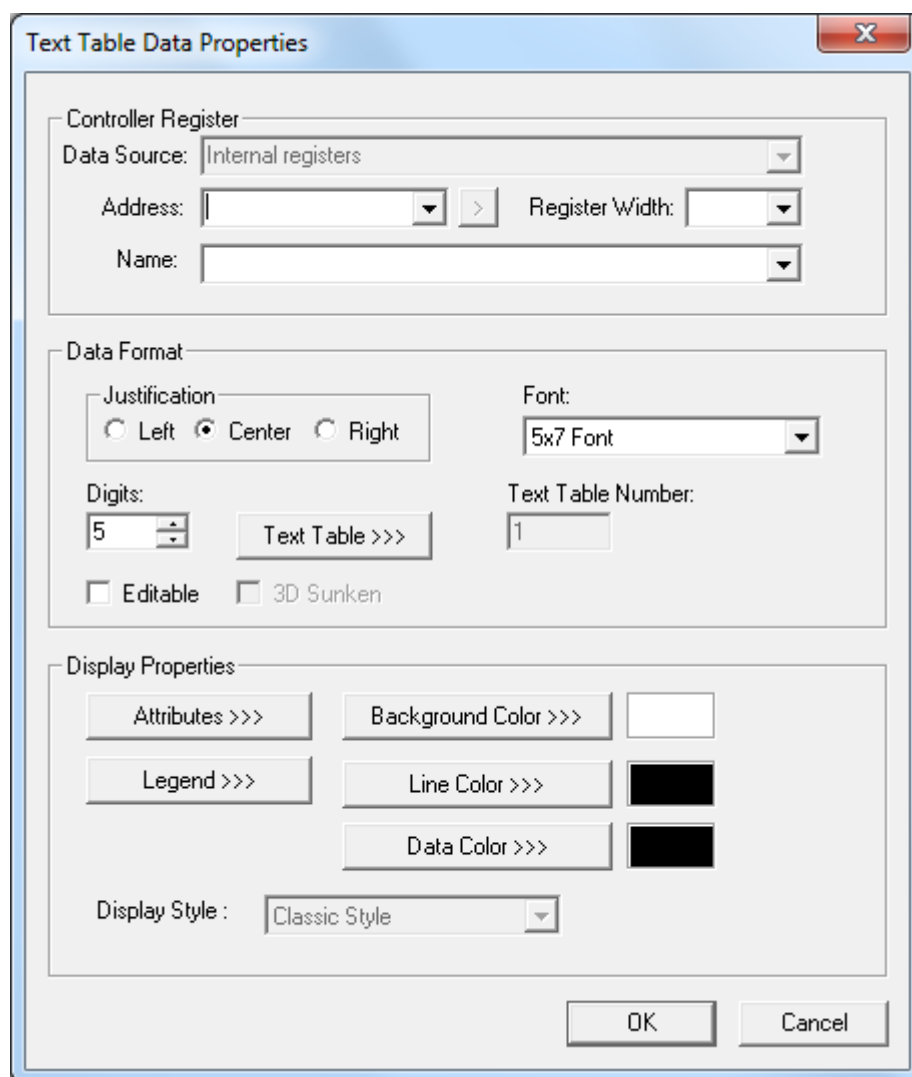
Border (ramka) - statyczny

Błyskanie - statyczny lub dynamiczny

Enable Input – dynamiczny

**Text Table Data:**

Obiekt do tworzenia tabeli tekstowej i formatowania tekstu odczytywanego z rejestru i/lub zapisywania do rejestru.



*Właściwości obiektu Text Table Data*

Opis wybranych parametrów:

- **Justification**

Grupa tych opcji służy do określenia wyrównania pól tekstowych w obszarze zajmowanym przez obiekt. Jeżeli długość pól tekstowych będzie mniejsza od liczby określonej w polu 'Digits', pola te zostaną odpowiednio przesunięte, stosownie do zaznaczonej opcji wyrównywania. Jeżeli długość pól tekstowych będzie większa od liczby określonej w polu 'Digits', część znaków z prawej strony zostanie zignorowana.

- **Font**

Pole to pozwala na wybór czcionki wykorzystywanej do wyświetlania pól tekstowych.

- **Digits**

Pole to określa maksymalną, dopuszczalną liczbę znaków pól tekstowych.

- **Text Table**

Wciśnięcie tego przycisku powoduje wywołanie edytora, pozwalającego na dodanie, usunięcie lub zmodyfikowanie dowolnej tablicy tekstowej.

- **Editable**

Zaznaczenie tej opcji pozwala na zmianę wartości zmiennej za pomocą klawiszy kursora.

Charakterystyka działania obiektu:

- **Controller Register**

Obiekt ten może być powiązany z rejestrem dowolnego typu i o dowolnej wielkości. Dla rejestrów binarnych, (%I, %Q, itp.) w polu 'Register Width' musi być wybrana pozycja 1-bit. Pole 'Register Width' określa liczbę bitów wykorzystywanych do wyświetlania wartości analogowej.

- **Funkcjonalność**

W zależności od wartości zapisanej w rejestrze sterownika wyświetlany jest odpowiedni tekst. Po każdej zmianie zawartości rejestru, obiekt przeszukuje tabelę tekstową (od wartości najmniejszej do największej) do momentu odnalezienia wartości większej lub równej od zapisanej w rejestrze sterownika. Następnie obiekt wyświetla tekst przypisany do tej wartości, z uwzględnieniem określonej, maksymalnej liczby znaków oraz wybranej opcji wyrównywania. W czasie przeszukiwania zawartości tabeli zarówno rejestr sterownika, jak i same wartości zapisane w tabeli są traktowane jako liczby całkowite bez znaku.

Jeżeli zaznaczona została opcja 'Editable', operator może po wybraniu obiektu wybrać jeden z tekstów tabeli i zapisać do rejestru sterownika wartość przypisaną do tego tekstu.

- **Sposób zmiany wartości w sterowniku za pomocą tablic tekstowych (po zaznaczeniu opcji Editable)**

Aby wprowadzić zmianę wartości w sterowniku należy wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter.

Podświetlone zostanie całe pole oraz włączony zostanie tryb WYBIERANIE.

Przewijać listę z tekstami za pomocą klawiszy kursora w górę/w dół.

W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter. W celu anulowania zmiany należy wcisnąć klawisz ESC, co spowoduje przywrócenie poprzedniej wartości. Bez względu na wciśnięty klawisz, nastąpi wyjście z trybu edycji i zaprzestanie podświetlania wartości.

### **Opcjonalne atrybuty wyświetlania**

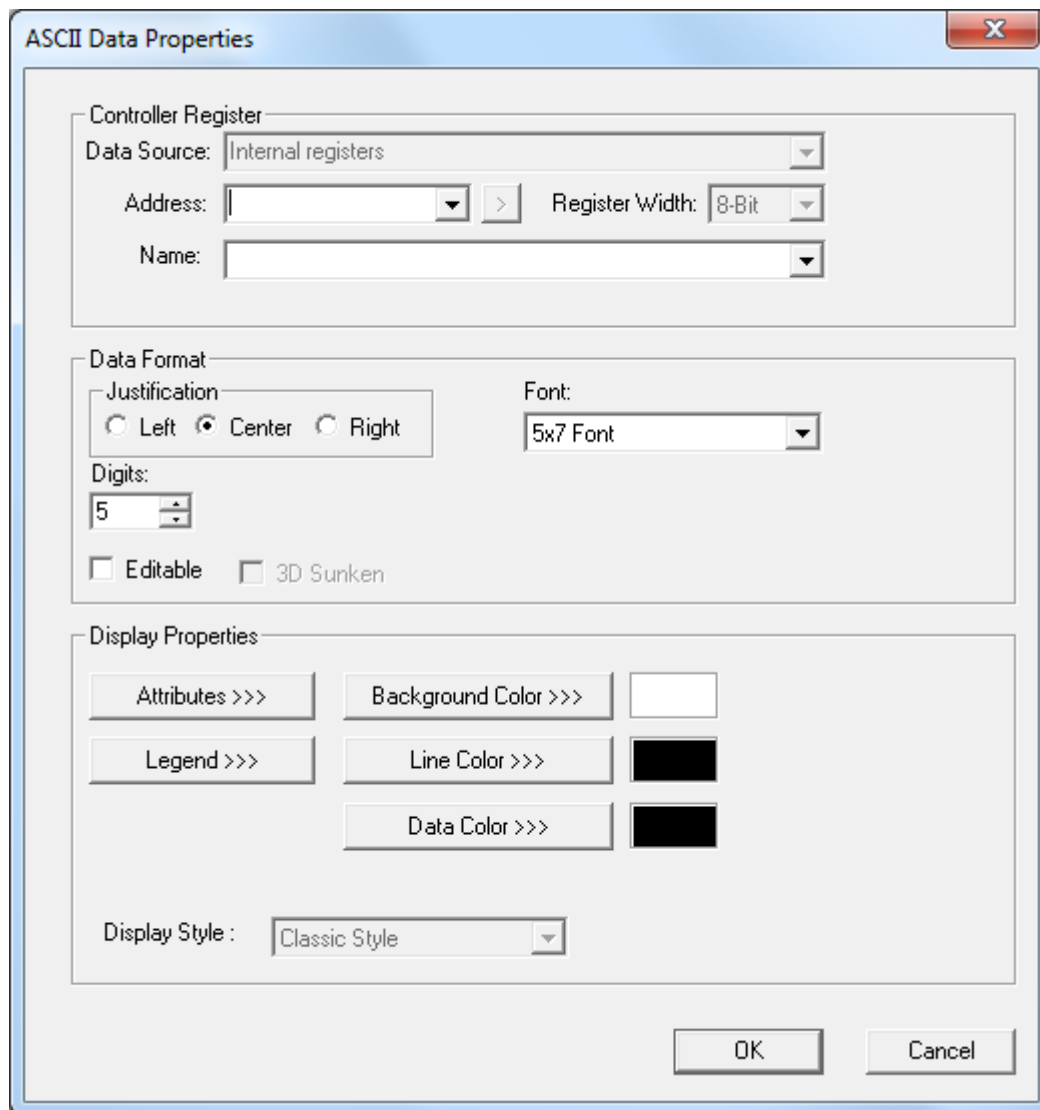
Border (ramka) - statyczny

Błyskanie - statyczny lub dynamiczny

Enable Input – dynamiczny

**ASCII Data (Dane ASCII):**

Formatowanie tekstu odczytywanego z rejestru i/lub zapis do rejestru.



*Właściwości obiektu ASCII Data*

Opis wybranych parametrów:

- **Justification**

Grupa tych opcji służy do określenia wyrównania tekstu w obszarze zajmowanym przez obiekt.

- **Font**

Pole to pozwala na wybór czcionki wykorzystywanej do wyświetlania tekstu.

- **Digits**

Pole to określa maksymalną, dopuszczalną liczbę znaków tekstu.

Charakterystyka działania obiektu:

- **Controller Register**

Adres musi być wielokrotnością 16 bitów plus 1 i musi wskazywać na grupę zawierającą dwa 8-bitowe znaki ASCII w jednym słowie 16-bitowym.

- **Funkcjonalność**

Obiekt ten wyświetla co 150 ms na ekranie tekst, którego początek znajduje się w podanym rejestrze sterownika.

Tekst powinien składać się z 8-bitowych znaków ANSI (dwa w każdym rejestrze 16-bitowym), przy czym znak z lewej strony powinien być zapisany w młodszych 8-bitach rejestru sterownika. Pole 'Digits' określa długość tekstu. Należy zwrócić uwagę, że koniec tekstu NIE JEST znaczonej za pomocą 8-bitowego znaku NULL (zero). Jeżeli w czasie odczytu tekstu zostanie napotkana 8-bitowa wartość NULL, zostanie ona potraktowana jako znak spacji ( ' ') i nadal będą odczytywane kolejne znaki, stosownie do liczby podanej w polu 'Digits'.

Jeżeli zaznaczono opcję 'Editable', wartość wprowadzona przez użytkownika zostanie zapisana do rejestru (rejestrów) sterownika.

- **Sposób wprowadzania do sterownika ciągów znaków ANSCI (po zaznaczeniu opcji Editable)**

Aby wprowadzić ciąg znaków do sterownika należy wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter.

Podświetlone zostanie całe pole oraz włączony zostanie tryb WSTAWIANIE.

Wciśnięcie dowolnego klawisza alfanumerycznego spowoduje wykasowanie całego pola, wpisanie wciśniętego klawisza na pierwszym miejscu z lewej strony, wypełnienie pozostałych miejsc znakami spacji ' ' i przejście do trybu NADPISYWANIE.

W trybie NADPISYWANIE można wybrać miejsce, do którego ma być wprowadzony znak za pomocą klawiszy kursora w lewo/ w prawo.

W edytorze obiektu, do każdego klawisza numerycznego przypisanych jest kilka znaków. Każde wciśnięcie klawisza powoduje wybieranie kolejnego znaku. Jeżeli wciskany jest ten sam klawisz, kursor nie przechodzi do następnego miejsca. W celu przejścia do następnego miejsca należy wcisnąć inny klawisz lub, jeżeli następny znak ma być taki sam, przejść za pomocą klawiszy kursora.

W celu potwierdzenia nowej wartości należy wcisnąć klawisz Edit/Enter. Aby anulować zmianę, należy wcisnąć klawisz Esc, co spowoduje przywrócenie poprzedniej wartości. Bez względu na wciśnięty klawisz nastąpi wyjście z trybu edycji i zaprzestanie podświetlania wpisanego tekstu.

- **Znaki przypisane do klawiszy numerycznych**

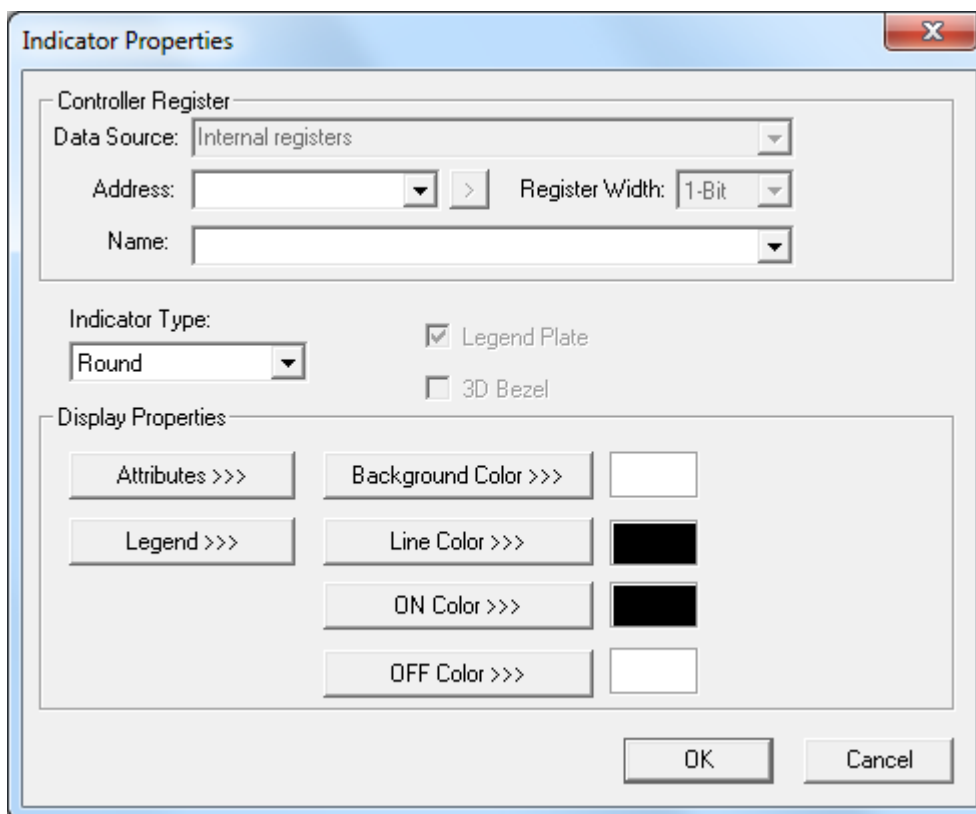
- [1] - ' ', Q, Z, q, z, 1
- [2] - A, B, C, a, b, c, 2
- [3] - D, E, F, d, e, f, 3
- [4] - G, H, I, g, h, i, 4
- [5] - J, K, L, j, k, l, 5
- [6] - M, N, O, m, n, o, 6
- [7] - P, R, S, p, r, s, '7
- [8] - T, U, V, t, u, v, 8
- [9] - W, X, Y, 'w, x, y, 9
- [0] - 0
- [+/-] - +, -, \*, /, =, (, ), }
- [.] - ., ?, :, ; (,), ('), (")

**Opcjonalne atrybuty wyświetlania:**

- Border (ramka) - statyczny
- Błyskanie - statyczny lub dynamiczny
- Enable Input – dynamiczny

**Indicator (Lamp) (Lampka sygnalizacyjna):**

Jest to obiekt w formie lampki sygnalizacyjnej, powiązany z rejestrem. W zależności od ustawienia na stronie może być pokazywana lampka okrągła, kwadratowa lub żarówka.



*Właściwości obiektu Indicator*

Opis wybranych parametrów:

- **Indicator Type**

Typ wyświetlanej (animowanej) ikony (okrągła, kwadratowa, żarówka)

Charakterystyka działania obiektu:

- **Controller Register**

W polu tym można wpisać adres dowolnego rejestru będący wielokrotnością 16 bitów plus 1.

- **Funkcjonalność**

Ikona tego obiektu odzwierciedla zawartość rejestru określonego w polu Controller Register. W przypadku ikony okrągłej i kwadratowej, jeżeli bit zapisany w rejestrze ma wartość 1, wewnątrz jest wypełniane kolorem linii, natomiast dla wartości 0, wewnątrz tego obiektu nie jest wypełniane. W przypadku ikony żarówki dla wartości 1 rysowane są promienie.

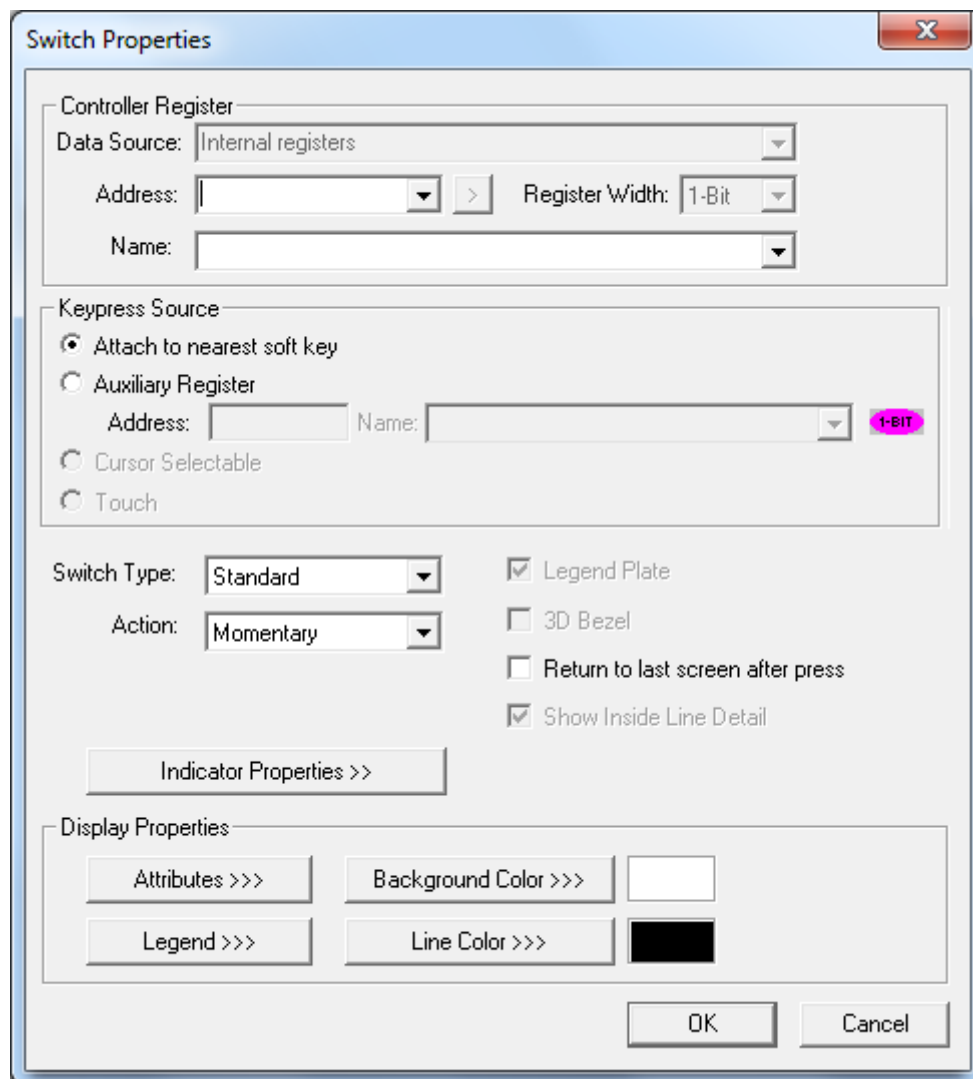
**Opcjonalne atrybuty wyświetlania:**

Border (ramka) - statyczny

Błyskanie - statyczny lub dynamiczny

### Switch (Przełącznik):

Obiekt do tworzenia przycisku powiązanego z rejestrem do zapisu wartości.



*Właściwości obiektu Switch*

Opis wybranych parametrów:

- **Keypress Source**

Grupa ta określa element, wizualizowany za pomocą przełącznika. Można podać programowalny klawisz ekranowy lub rejestr zewnętrzny.

- **Switch Type**

Typ wyświetlanej (animowanej) ikony (przełącznik prostokątny, okrągły, kwadratowy).



- **Action**

Pole to umożliwia wybór działania związanego ze zmianą stanu przełącznika. Momentary Przez cały czas wciśnięcia klawisza, w rejestrze będzie przechowywana wartość 1. Po zwolnieniu klawisza, wartość ta zostanie zastąpiona przez 0.

**ON** Wciśnięcie klawisza spowoduje zapisanie do rejestru wartości 1.

**OFF** Wciśnięcie klawisza spowoduje zapisanie do rejestru wartości 0.

**Toggle** Każde wciśnięcie klawisza spowoduje zmianę wartości na przeciwną.

- **Show On/Off state caption**

Po zaznaczeniu tej opcji, animowana ikona będzie wyświetlana wraz z jednym z dwóch tekstów.

- **State Properties (przycisk)**

Przycisk ten jest aktywowany po zaznaczeniu opcji 'Show On/Off state caption'. Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego umożliwiającego zdefiniowanie nowych tekstów do sygnalizacji stanów przełącznika oraz określenie, czy stan przełącznika ma być uzależniony od rejestru sterownika podanego dla tego obiektu czy od zdefiniowanego rejestru zewnętrznego.

- **Return to last screen after press**

Po zaznaczeniu tej opcji, natychmiast po zmianie stanu, następuje powrót do poprzednio wyświetlanej strony (działanie podobne jak klawisza ESC). Jeżeli za pomocą obiektu Screen Jump została utworzona kolejka stron, wyświetlona zostanie poprzednia strona. W przypadku, gdy nie ma takiej kolejki, ekran nie ulegnie zmianie. Opcja ta pozwala na tworzenie okien dialogowych z przyciskami 'OK' i 'Cancel', w których po wykonaniu działania następuje powrót do poprzedniej strony.

Charakterystyka działania obiektu:

- **Controller Register**

W polu tym można wpisać adres dowolnego rejestru, będący wielokrotnością 16 bitów plus 1.

- **Funkcjonalność**

Zawartość rejestru sterownika jest zmieniana po każdej zmianie stanu źródła sterowania w przypadku wybrania działania 'Toggle' lub po każdym przejściu ze stanu niskiego do wysokiego źródła sterowania dla pozostałych działań. Jeżeli w grupie 'Keypress Source' wybrano opcję 'Auxilliary Register', zmiana jest dokonywana dopiero po pierwszym przejściu ze stanu niskiego do wysokiego. Mechanizm taki zapobiega nieprawidłowej zmianie wartości w sytuacji, gdy w polu 'Keypress Source' wybrano opcję 'Attach to nearest soft key', a klawisze funkcyjne (F1-F10) są wykorzystywane do zmiany stron.

W przypadku ikony standardowej, okrągłej i prostokątnej, stan ON lub OFF jest sygnalizowany poprzez zmianę wypełnienia i koloru linii ikony. W stanie ON wewnątrz ikony jest wypełniane kolorem linii, natomiast w stanie OFF wewnątrz pozostawiane jest bez wypełnienia. Należy zwrócić uwagę, że ikona odzwierciedla stan rejestru sterownika a nie wciśniętego klawisza (lub rejestru określony w polu 'Keypress Source'). Ikona odzwierciedli więc fakt zmiany zawartości rejestru sterownika przez sieć albo z poziomu programu sterującego.

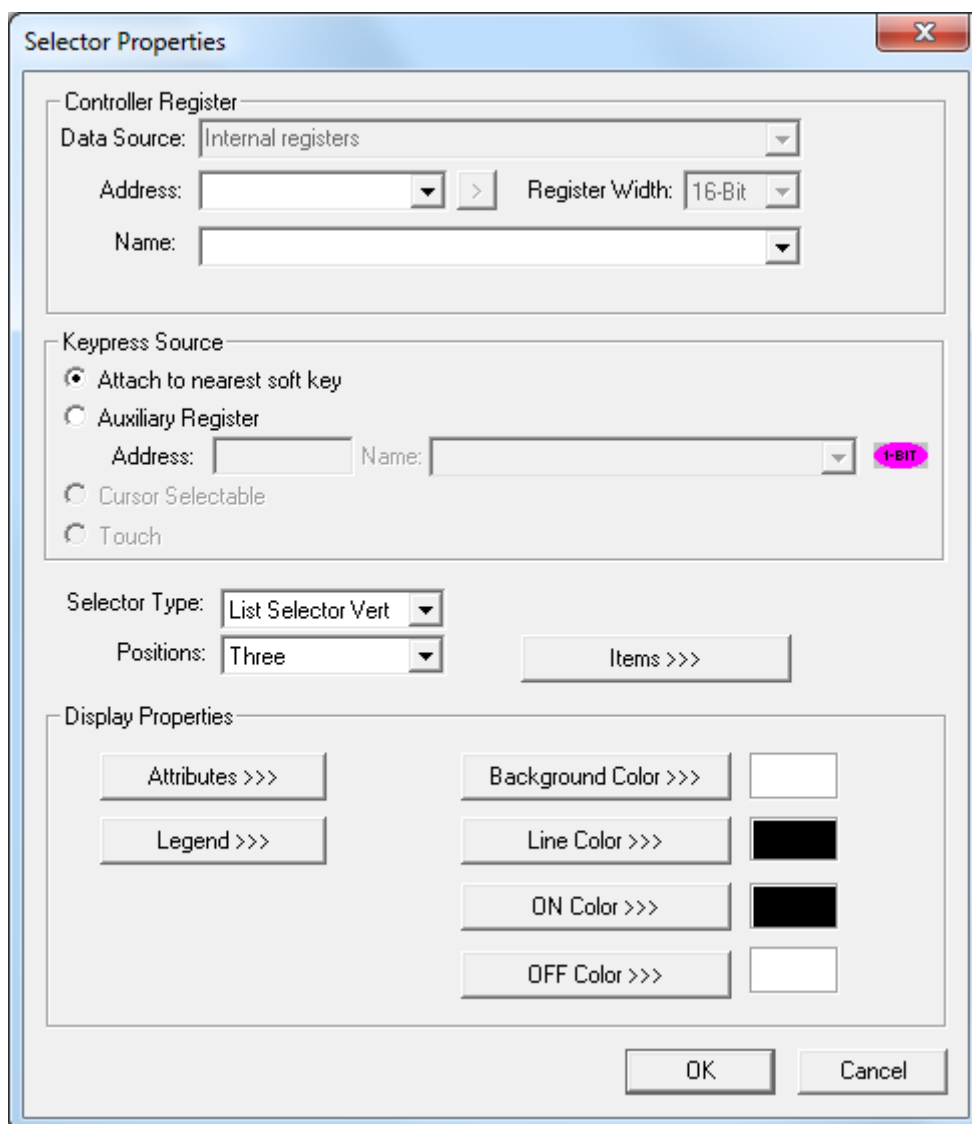
Opcjonalnie, we wnętrzu obiektu mogą być wyświetlane teksty dla stanu ON i stanu OFF. Obydwa te teksty mogą być zdefiniowane przez użytkownika i zmieniać się w zależności od stanu przełącznika (rejestr sterownika) lub urządzenia docelowego (rejestr zewnętrzny), które może być sterowane nie tylko za pomocą tego rejestru przełącznika.

**Opcjonalne atrybuty wyświetlania:**

- Border (ramka) - statyczny
- Enable Input - dynamiczny
- Show ICON – statyczny

**Przełącznik wielopozycyjny:**

Wyświetla i formatuje wielopozycyjny przełącznik, powiązany z rejestrem do zapisywania wartości.



*Właściwości obiektu Selector*

Opis wybranych parametrów:

- **Positions**

Liczba pozycji przełącznika. Można wybrać wartość z zakresu 1 do 4.

- **Items**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego, pozwalającego na zdefiniowanie tekstu dla każdej pozycji przełącznika oraz sumarycznej liczby w przypadku, gdy zdefiniowana jest tylko jedna pozycja.

Charakterystyka działania obiektu:

- **Controller Register**

Należy podać adres dowolnego rejestru będący wielokrotnością 16 bitów plus 1. Obiekt ten wykorzystuje 16 bitów (słowo).

- **Funkcjonalność**

Obiekt ten symuluje jedno lub wielopozycyjny przełącznik. Każda pozycja tego przełącznika jest powiązana z odpowiednim źródłem sterowania (programowalnym klawiszem ekranowym lub zewnętrznym rejestrem sterownika). Liczba wykorzystywanych programowalnych klawiszy ekranowych zależy od wartości wybranej w polu 'Positions'. W przypadku powiązania obiektu z rejestrem zewnętrznym, obiekt ten wykorzystuje odpowiednią liczbę sąsiadujących ze sobą bitów rejestru, zależną od wartości ustawionej w polu 'Positions' (adres musi być wielokrotnością 16 bitów plus 1).

Wystąpienie zdarzenia w źródle sterowania (przejście ze stanu niskiego do wysokiego) powoduje zapisanie pozycji do określonego rejestru sterownika. Pozycja położona najbliżej górnej części ekranu jest powiązana z pierwszym źródłem sterowania. Jeżeli naraz zostanie zmieniona wartość kilku bitów źródła sterowania (jest to możliwe wyłącznie po połączeniu tego rejestru z obiektem zewnętrznym), najwyższy priorytet ma bit o najmniejszym numerze.

W przełączniku jednopozycyjnym, obiekt jest sterowany za pomocą jednego klawisza lub jednego rejestru zewnętrznego. Każde zdarzenie źródła sterowania (wciśnięcie programowalnego klawisza ekranowego lub zmiana rejestru zewnętrznego) polegające na przejściu ze stanu niskiego do wysokiego powoduje inkrementację liczby określającej pozycję przełącznika z zakresu [0;(Liczba pozycji-1)] i zapisanie tej liczby do rejestru sterownika. Obiekt podświetla ostatnio wybraną pozycję przełącznika. Należy zwrócić uwagę, że ikona odzwierciedla stan rejestru sterownika, a nie wciśnięty klawisz (lub rejestr określony w polu 'Keypress Source'). Ikona odzwierciedli więc fakt zmiany zawartości rejestru sterownika przez sieć albo z poziomu programu sterującego.

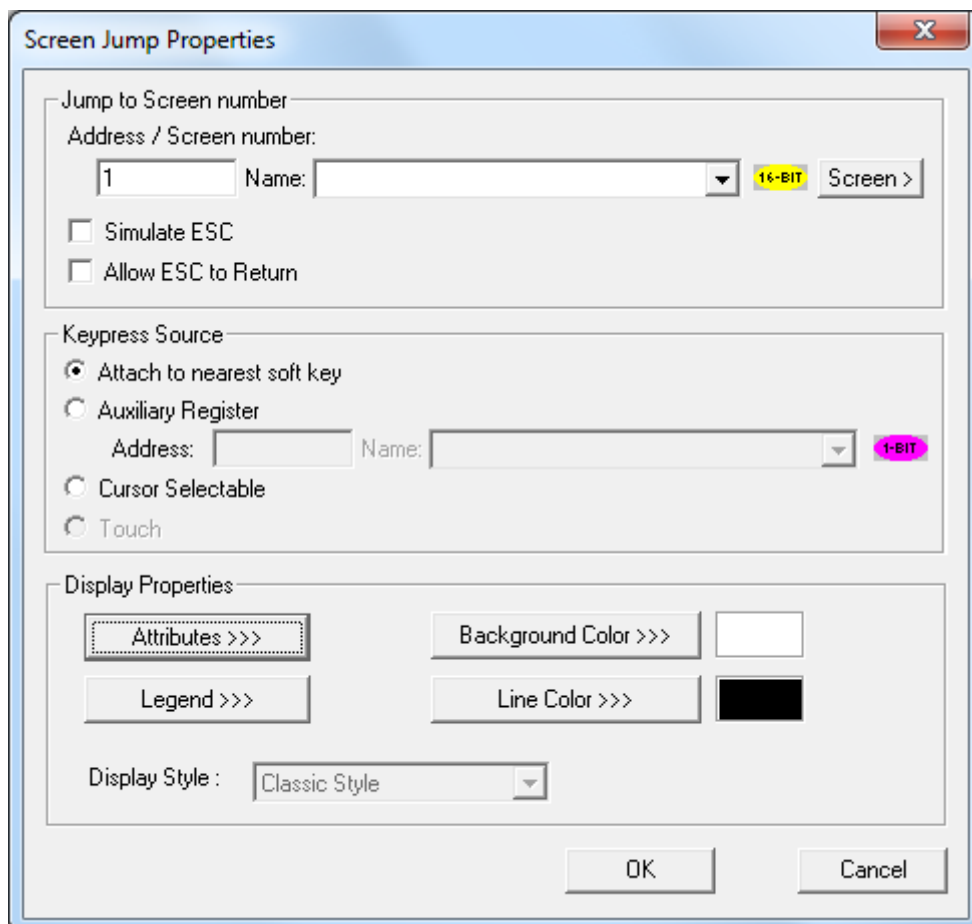
**Opcjonalne atrybuty wyświetlania:**

Border (ramka) - statyczny

Enable Input – dynamiczny

### Screen Jump (Zmiana ekranu):

Obiekt pozwalający na przejście do określonej strony.



Właściwości obiektu Screen Jump

Opis wybranych parametrów:

- **Address/Number**

**W polu tym należy podać numer strony, który ma być wyświetlony (bezpośrednio) lub za pomocą rejestru sterownika OCS. Numer strony może być również obliczony poprzez dodanie lub odjęcie odpowiedniej wartości od numeru aktualnej strony. W tym celu wpisaną wartość należy poprzedzić znakiem '+' lub '-'.**

- **Allow ESC to Return**

Zaznaczenie tej opcji powoduje wstawianie ekranów do kolejki. Po przejściu do strony wskazywanej przez obiekt i wciśnięciu klawisza ESC nastąpi powrót do strony, z poziomu której nastąpiło wywołanie. W kolejce można zapisać do 16 numerów stron. Jeżeli kolejka jest pusta, wciśnięcie klawisza ESC nie powoduje żadnych zmian na ekranie.

Charakterystyka działania obiektu:

- **Funkcjonalność**

Obiekt ten może być sterowany przez jedno z trzech źródeł sterowania (programowalny klawisz ekranowy, rejestr zewnętrzny lub za pomocą kursora). Po zaistnieniu odpowiedniego zdarzenia, następuje przejście do wyświetlania nowej strony. Jeżeli zaznaczona jest opcja 'Allow ESC to Return', przed przejściem do nowej stron, numer aktualnej strony zapisywany jest do kolejki.

Jeżeli obiekt reaguje na wciśnięcie programowalnego klawisza ekranowego lub zmianę rejestru zewnętrznego, przejście do innej strony następuje po zmianie stanu z niskiego na wysoki. W przypadku wyboru za pomocą klawiszy kursora, w celu przejścia do nowej strony należy najpierw wybrać obiekt za pomocą klawiszy kursora, a następnie wcisnąć klawisz Edit/Enter.

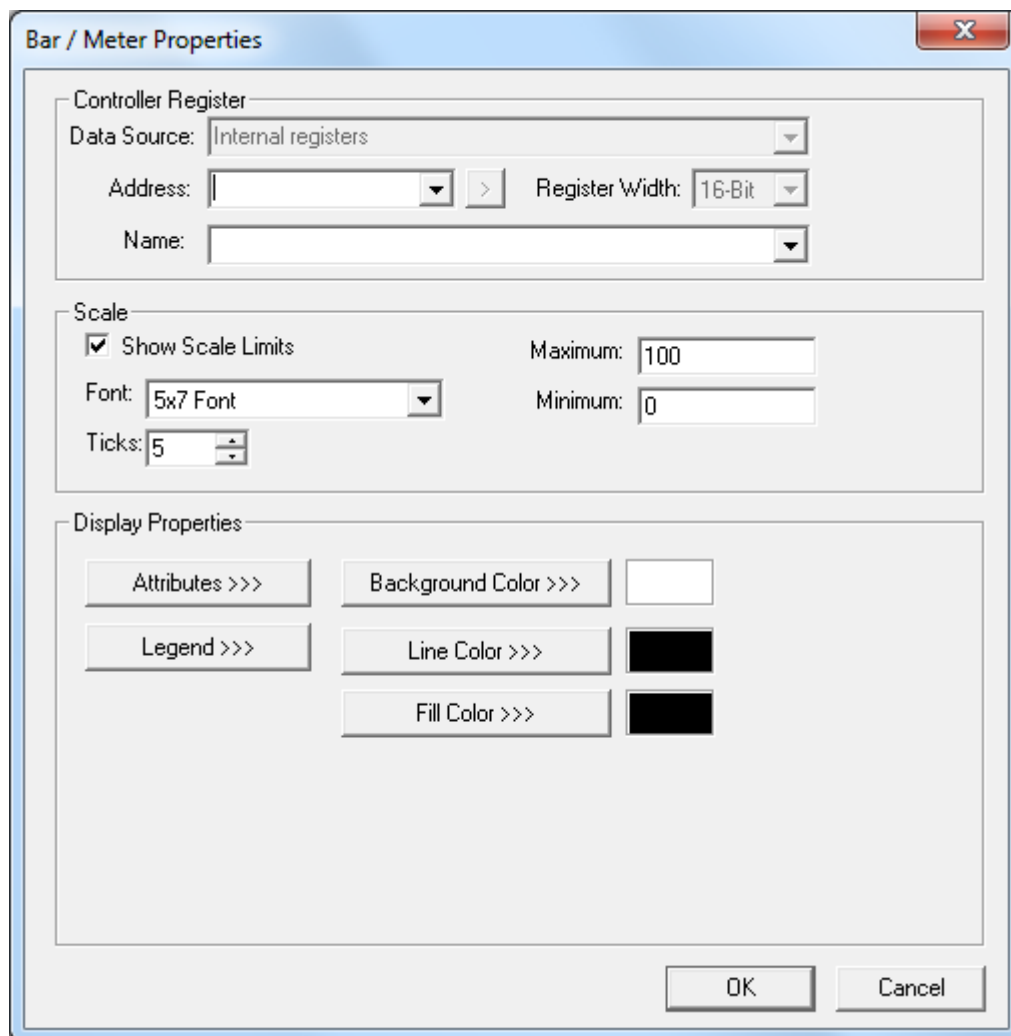
Przejście do nowej strony jest realizowane wyłącznie jeżeli aktualnie wyświetlany jest ekran UŻYTKOWNIKA (%SR2=0). Jeżeli aktualnie wyświetlany jest ekran ALARMÓW (%SR2<>0), nie nastąpi przejście do nowej strony.

**Opcjonalne atrybuty wyświetlania**

- Border (ramka) - statyczny
- Enable Input - dynamiczny
- Show ICON – statyczny

### **Bar Graph (Wykres słupkowy):**

Wykres słupkowy powiązany z określonym rejestrem.



*Właściwości obiektu Bar Graph*

Opis wybranych parametrów:

- **Show scale limits**

Po zaznaczeniu tej opcji, będą pokazywane wartości graniczne skali.

- **Font**

Pole to pozwala na wybór czcionki do wyświetlania wartości granicznych skali.

- **Maximum/Minimum**

Pola te pozwalają na zdefiniowanie zakresu skali miernika.

- **Ticks**

Liczba działek skali.

Charakterystyka działania obiektu:

- **Controller Register**

Adres musi być wielokrotnością 16 bitów plus 1. Wartość zapisana w rejestrze jest traktowana jako liczba 16-bitowa ze znakiem.

- **Funkcjonalność**

Obiekt przez cały czas monitoruje wskazany rejestr. Monitorowana wielkość jest skalowana w przedziale określonym przez dolną wartość graniczną (0% wypełnienia) i górną wartość graniczną (100% wypełnienia).

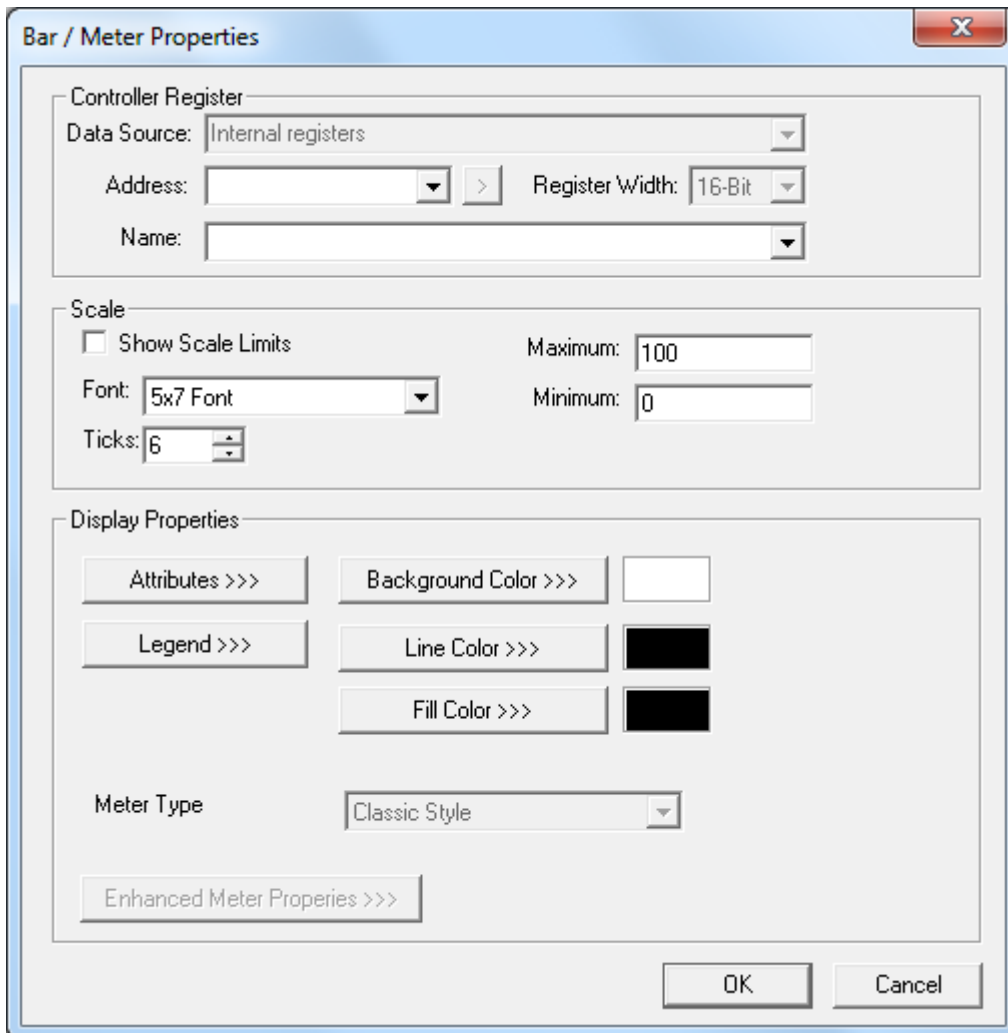
Kierunek wypełniania zależy od rozmiaru obiektu. Jeżeli szerokość jest większa od wysokości, skala jest rysowana w kierunku poziomym. W przeciwnym wypadku, skala jest rysowana w kierunku pionowym.

**Opcjonalne atrybuty wyświetlania:**

Brak

### **Meter (Miernik):**

Miernik powiązany z określonym rejestrem.



*Właściwości obiektu Meter*

Opis wybranych parametrów:

- **Show scale limits**

Po zaznaczeniu tej opcji na skali tarczowej pokazywane będą wartości graniczne.

- **Font (grupa Scale)**

Pole to pozwala na wybór czcionki do wyświetlania wartości granicznych skali.

- **Maximum/Minimum (grupa Scale)**

Pola te pozwalają na zdefiniowanie zakresu skali miernika.

- **Ticks**

Liczba działek skali wyświetlanych na tarczy.



Charakterystyka działania obiektu:

- **Controller Register**

Adres musi być wielokrotnością 16 bitów plus 1. Wartość zapisana w rejestrze jest traktowana jako liczba 16-bitowa ze znakiem.

- **Funkcjonalność**

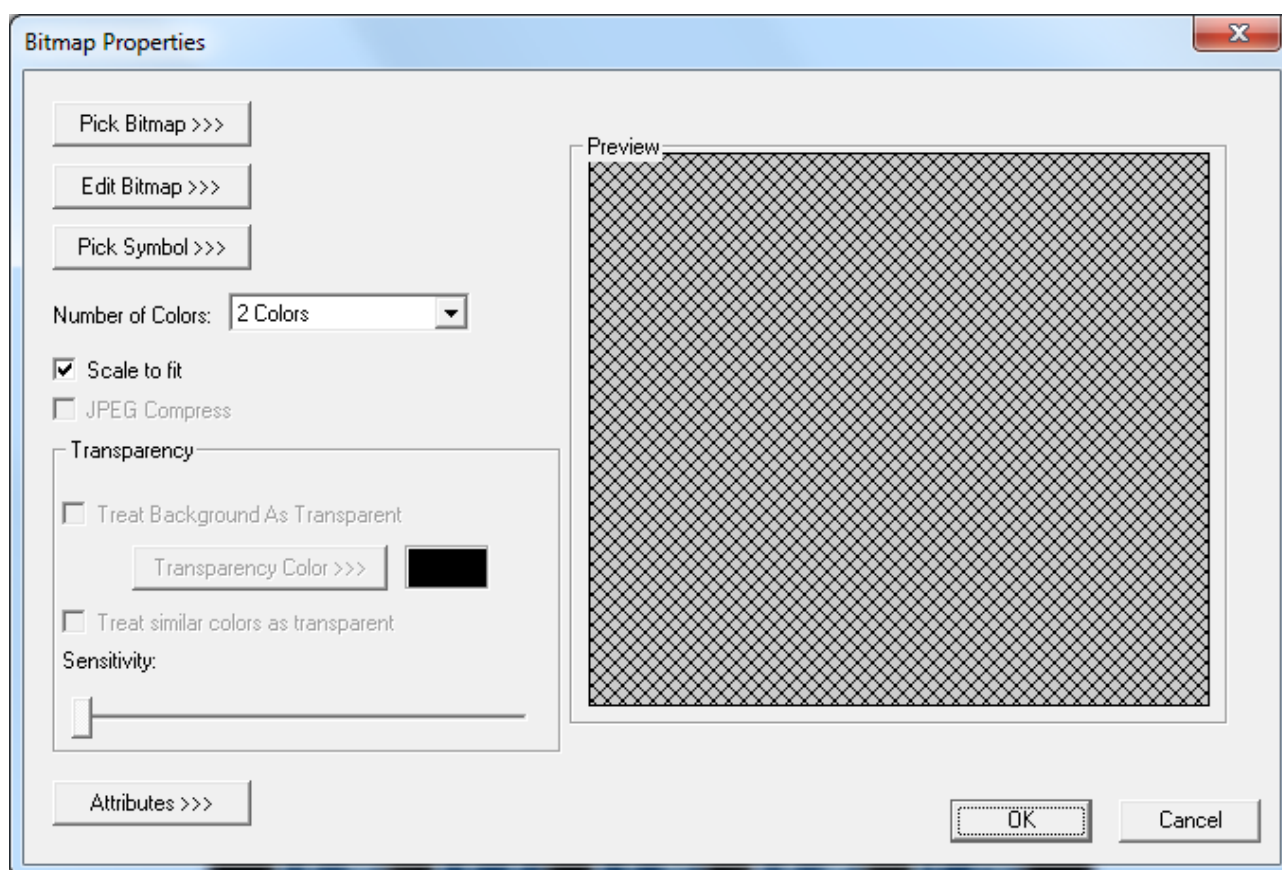
Obiekt przez cały czas monitoruje wskazany rejestr sterownika. Monitorowana wielkość jest skalowana w przedziale określonym przez dolną wartość graniczną (wskazówka obrócona maksymalnie w lewo) i górną wartość graniczną (wskazówka obrócona maksymalnie w prawo).

**Atrybuty wyświetlania**

Brak

**Static Bitmap (Statyczna mapa bitowa):**

Obiekt ten wyświetla pojedynczą mapę bitową.



*Właściwości obiektu Bitmap*

Opis wybranych parametrów:

- **Pick Bitmap**

Kliknięcie na ten przycisk powoduje wyświetlenie okna dialogowego do wyboru pliku z mapą bitową.

- **Edit Bitmap**

Kliknięcie tego przycisku powoduje otwarcie edytora map bitowego i wczytanie uprzednio wybranej mapy bitowej. Należy pamiętać, aby wcześniej wskazać położenie edytora map bitowych poprzez wybranie z menu 'Tools' polecenia 'Set Bmp Editor'. Najczęściej wskazywana jest ścieżka dostępu do edytora MS Paint, dostarczanego wraz z systemem operacyjnym Windows.

- **Scale to Fit**

Zaznaczenie tej opcji powoduje powiększenie importowanej mapy bitowej do rozmiarów obiektu. Jeżeli opcja ta nie zostanie zaznaczona, rozmiary obiektu zostaną automatycznie zmienione do rozmiarów mapy bitowej, poprzez zmianę położenia prawego, dolnego narożnika. Jeżeli mapa bitowa jest większa od ekranu, zostanie ona obcięta.

Charakterystyka działania obiektu:

- **Funkcjonalność**

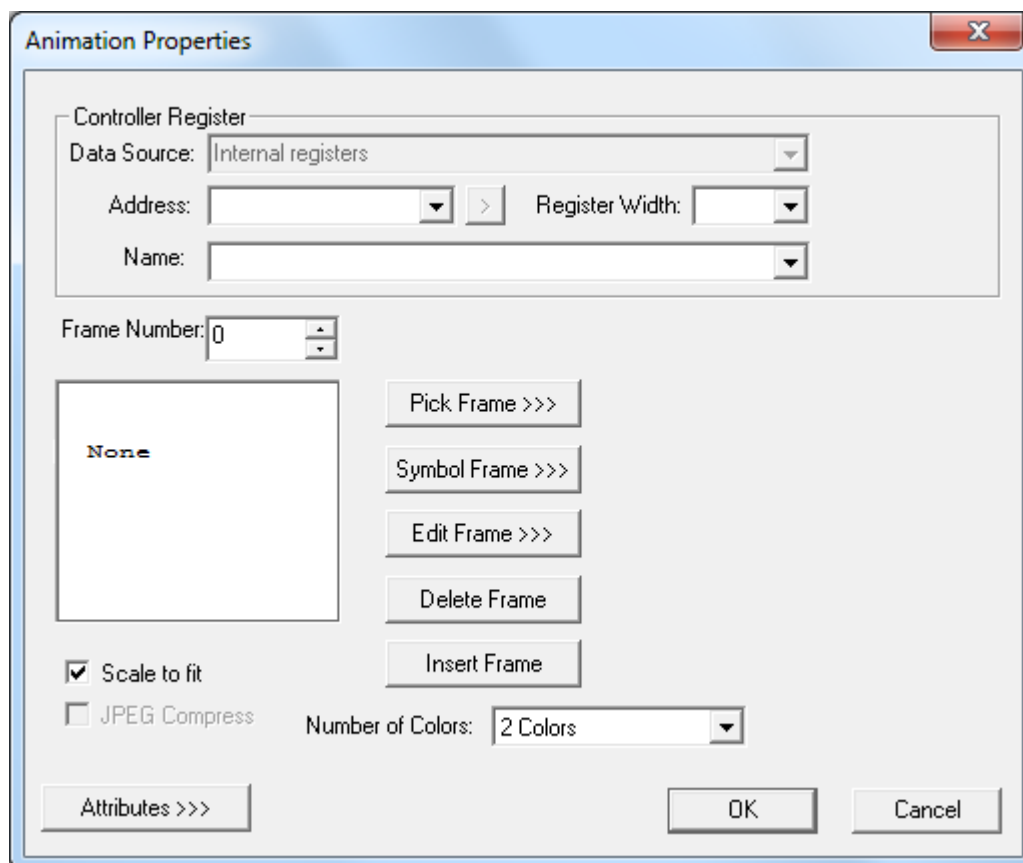
Obiekt ten pozwala na wstawienie mapy bitowej (plik z rozszerzeniem .BMP), która następnie jest wyświetlana na ekranie. Mapa bitowa jest rysowana jeden raz, w czasie inicjalizowania wyświetlania strony.

**Atrybuty wyświetlania:**

Brak

### **Animation (Animacja):**

Obiekt ten wyświetla odpowiednią mapę bitową (ramkę), w zależności od zawartości rejestru sterownika.



*Właściwości obiektu Animation*

Opis wybranych parametrów:

- **Frame number**

Numer klatki dla której wybierana jest mapa bitowa.

- **Pick Frame**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego do wyboru pliku z mapą bitową.

- **Edit Frame**

Kliknięcie na ten przycisk powoduje otworenie edytora map bitowego i wczytanie uprzednio wybranej mapy bitowej. Należy pamiętać, aby wcześniej wskazać położenie edytora map bitowych poprzez wybranie z menu 'Tools' polecenia 'Set Bmp Editor'. Najczęściej wskazywana jest ścieżka dostępu do edytora MS Paint, dostarczanego wraz z systemem operacyjnym Windows.

- **Delete Frame**

Usunięcie mapy bitowej zdefiniowanej dla aktualnej ramki i przesunięcie wszystkich map bitowych klatek o wyższych numerach jeden poziom niżej. Wciśnięcie tego przycisku powoduje więc usunięcie mapy bitowej z sekwencji klatek.

- **Insert Frame**

Przesunięcie wszystkich map bitowych, o numerach wyższych od numeru aktualnej klatki jeden poziom wyżej. Pozwala to na wstawienie do sekwencji nowej mapy bitowej.

- **Scale to Fit**

Zaznaczenie tej opcji powoduje powiększenie importowanej mapy bitowej do rozmiarów obiektu. Jeżeli opcja ta nie zostanie zaznaczona, rozmiary obiektu zostaną automatycznie zmienione do rozmiarów mapy bitowej 'pierwszej klatki', poprzez zmianę położenia prawego, dolnego narożnika. Jeżeli mapa bitowa jest większa od ekranu lub od mapy bitowej 'pierwszej klatki', jest ona obcinana.

Charakterystyka działania obiektu:

- **Controller Register**

Obiekt ten może być powiązany z rejestrem dowolnego typu i o dowolnej wielkości. Dla rejestrów binarnych (%I, %Q, itp.), w polu 'Register Width' musi być wybrana pozycja 1-bit. Pole 'Register Width' określa liczbę bitów wykorzystywanych do wyświetlania wartości analogowej.

- **Funkcjonalność**

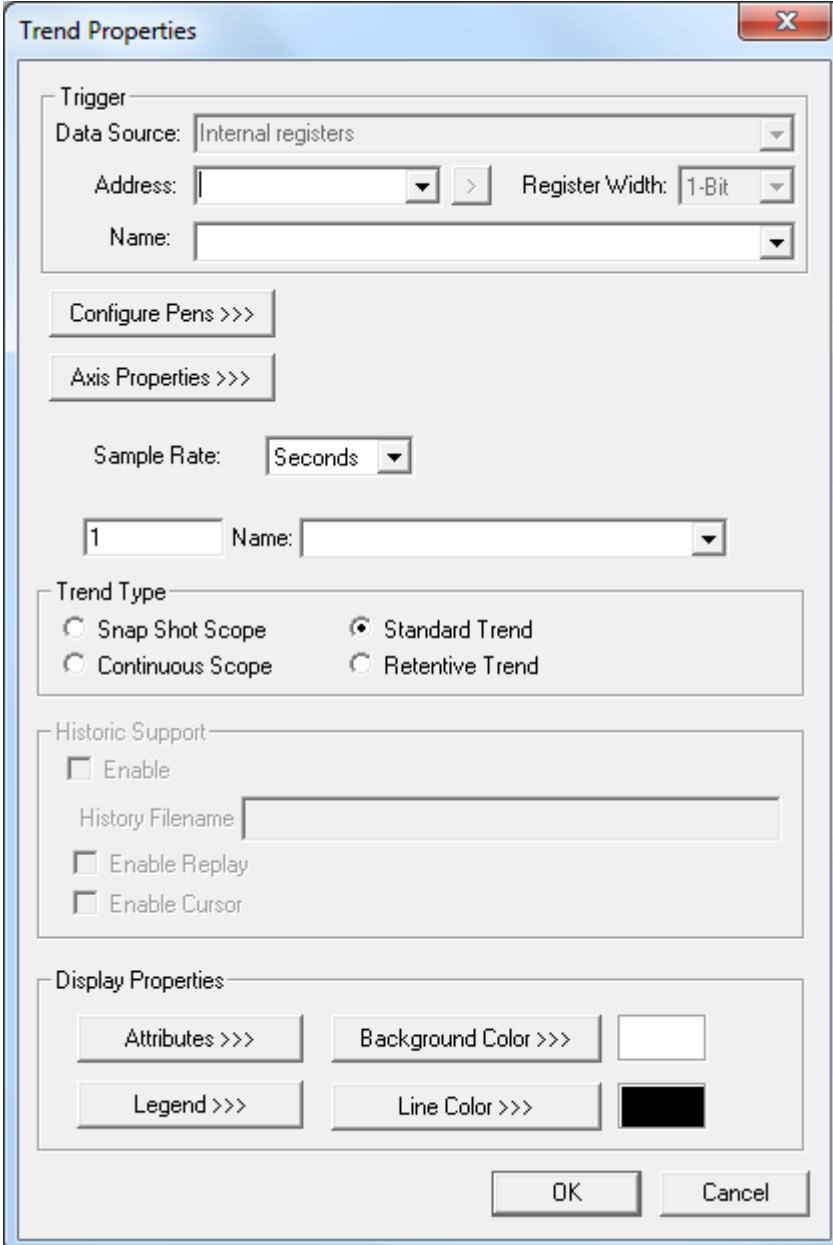
Obiekt ten pozwala na wstawienie szeregu map bitowych (pliki z rozszerzeniem .BMP), wyświetlanych w zależności od zawartości rejestru sterownika. Rejestr sterownika jest przez cały czas monitorowany i traktowany jako wartość bez znaku. W przypadku zmiany zawartości tego rejestru, na ekranie pokazywana jest klatka o numerze równym monitorowanej wartości. Jeżeli monitorowana wartość jest większa od najwyższego numeru klatki, wyświetlana jest mapa bitowa zdefiniowana dla klatki o najwyższym numerze.

### **Atrybuty wyświetlania**

Brak

### **Data Trend (Trend danych):**

Obiekt to tworzenia wykresu trendów, pokazującego przebieg zmian w czasie wartości jednej lub więcej zmiennych. Dostępne są cztery rodzaje trendów. Każdy trend może pokazywać do czterech linii (wartości z rejestrów). Dla każdego wykresu trendów należy określić rejestr, wykorzystywany do rozpoczęcia rysowania trendu (Trigger Address).



*Właściwości obiektu Trend*

Opis wybranych parametrów:

- **Sample Rate**

Czas pomiędzy pobieraniem próbek. Ogólnie ujmując, można wybrać wartość z przedziału 1-999. W polu 'Hours' należy wybrać żadaną jednostkę czasu. Rodzaj dostępnych jednostek czasu zależy od typu trendu. W przypadku trendów typu 'Standard Trend' lub 'Retentive Trend', jednostkami mogą być sekundy, minuty lub godziny. W przypadku pozostałych typów trendów, jednostkami mogą być wyłącznie milisekundy.

- **Configure Pens**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego, w którym definiowana jest liczba linii oraz związane z nimi rejestry. Jednocześnie może być rysowanych do czterech linii, w jednym z trzech stylów (linia pełna przerywana, lub kropkowana).

- **Axis Properties**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego do definiowania etykiety osi, wartości granicznych (skalowanie w osi Y) i liczby działek na osi.

- **Trend Type**

Grupa ta pozwala na wybranie typu trendu.

- **Trigger**

W grupie tej należy podać rejestr sterownika OCS (1-bitowy) sterujący tym obiektem (dane rysowane, nie rysowane, usuwanie danych z trendu).

Charakterystyka działania obiektu:

- **Funkcjonalność**

Po zainicjalizowaniu odczytywane są rejestry sterownika zdefiniowane dla poszczególnych pisaków, a zapisane w nich wartości są rysowane na wykresie. Dane na trendzie są rysowane wyłącznie, gdy w rejestrze wyzwalania zapisana jest wartość 1 i sterownik znajduje się w trybie pracy RUN. Jeżeli zawartość tego rejestru zostanie zmieniona na 0, rysowanie zostaje wstrzymane. Po wykryciu zmiany zawartości tego rejestru z 0 na 1, przed rozpoczęciem rysowania poprzednie dane są kasowane. W przypadku zmiany trybu RUN-STOP-RUN, poprzednie dane NIE SĄ kasowane i trend jest nadal kontynuowany pod warunkiem, że zawartość rejestru wyzwalania nie została zmieniona w czasie tej zmiany trybów.

Każdy rejestr sterownika jest traktowany jako 16-bitowa wartość bez znaku, a jego zawartość jest skalowana w osi pionowej (i obcinana) do wartości Y-Min i Y-Max, zdefiniowanych w oknie dialogowym 'Axis Dialog'.

**Snap shot scope (Trendy szybkie)** - tryb ten pozwala na rysowanie wykresu, którego szerokość nie przekracza szerokości obiektu. Obiekt posiada tylko jeden stan, aktywny a dane są pamiętane wyłącznie jeżeli strona na której umieszczono ten obiekt jest wyświetlana. Minimalny czas próbkowania wynosi 10 ms.

**Continuous scope (Trend szybki)** - tryb ten pozwala na ciągłe uaktualnianie danych po wyzwoleniu (wykres jest przewijany). Obiekt posiada tylko jeden stan, aktywny, a dane są

pamiętane wyłącznie jeżeli ekran na którym umieszczono obiekt jest wyświetlany. Minimalne natężenie próbkowania wynosi 50 ms.

Standard Trend - typ ten pozwala na ciągłe uaktualnianie danych po wyzwoleniu, bez względu na fakt, czy ekran, na którym umieszczono obiekt jest, wyświetlany. Dodatkowo, obiekt ten może być niezależnie sterowany, bez względu na fakt, czy strona na której go umieszczono jest wyświetlana. Dane mogą być zawsze oglądane, o ile widoczna jest strona zawierająca ten obiekt. Jeżeli trend jest kontynuowany po zmianie trybu RUN-STOP-RUN, dane są oddzielane za pomocą pionowej, kreskowanej linii.

**Retentive trend** - trend ten zachowuje się identycznie jak opisywany poprzednio. Jedną różnicą to fakt zapamiętywania w podtrzymywanej bateryjnie pamięci danych (o długości równej szerokości obiektu) i ich wczytywanie do obiektu po włączeniu zasilania. Jeżeli trend jest kontynuowany po zmianie trybu RUN-STOP-RUN, dane są oddzielane za pomocą pionowej, kreskowanej linii.

Każda próbka zajmuje na osi poziomej jeden piksel. W edytorze graficznym, na obiekcie pokazywane są w białym prostokącie szerokość i wysokość wykresu. Na podstawie pokazywanej szerokości użytkownik może wyznaczyć liczbę próbek, jednocześnie pokazywanych na wykresie.

Maksymalna liczba obiektów Data Trend jest ograniczona i zależy od wybranego typu trendu. Ograniczenie to faktycznie wynika z liczby skonfigurowanych pisaków. Można skonfigurować maksymalnie 16 pisaków dla trendów typu Retentive i 32 dla dowolnej kombinacji typów Standard i Retentive.

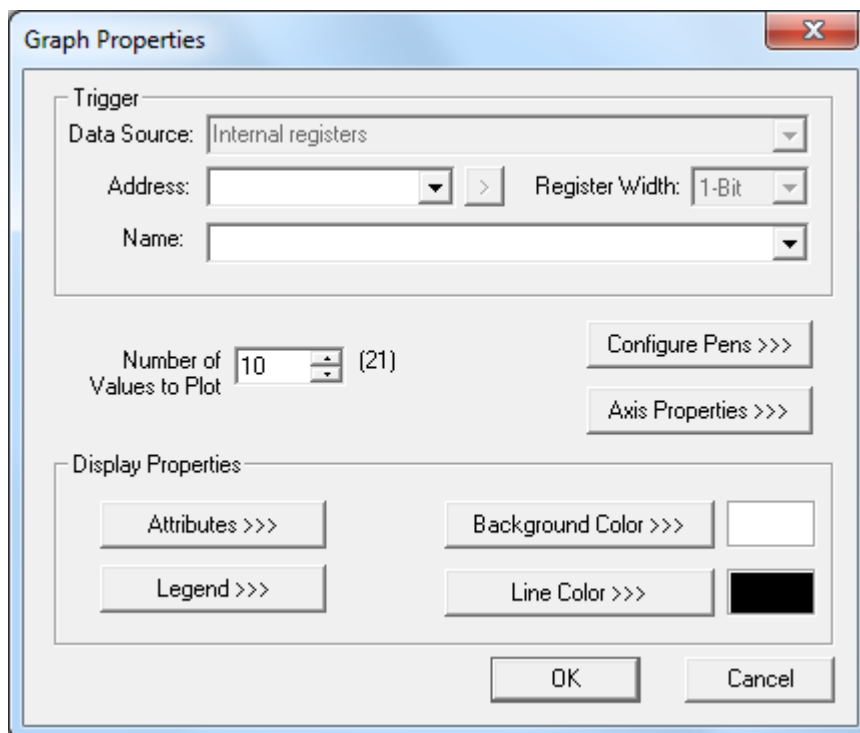
### Atrybuty wyświetlania

Border

### ***X-Y Graph (Wykres X-Y):***

Obiekt ten pozwala na utworzenie wykresu X-Y, przedstawiającego przebieg zmian zmiennej względem jednej lub więcej zmiennych. Na wykresie można rysować lub umieszczać szereg wartości przy pomocy współrzędnych X-Y. Każdy wykres może pokazywać do czterech linii (rejestrów).

W czasie konfigurowania należy podać adres Trigger Adres, wykorzystywany do zerowania rejestrów i wznawiania rysowania wykresu.



*Właściwości obiektu X-Y Graph*

Opis wybranych parametrów:

- **Number of values to plot**

W polu tym należy określić liczbę sąsiadujących rejestrów, wykorzystywanych do tworzenia wykresu. Wprowadzona liczba musi być mniejsza od wartości maksymalnej (szerokości wykresu), pokazywanej w nawiasach z prawej strony tego pola.

- **Configure Pens**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego, w którym definiowana jest liczba pisaków oraz związane z nimi rejestry. Jednocześnie może być rysowanych do czterech linii, w jednym z trzech stylów (linia pełna przerywana, lub kropkowana).

- **Axis Properties**

Wciśnięcie tego przycisku powoduje wyświetlenie okna dialogowego do definiowania etykiety osi, wartości granicznych (skalowanie w osi Y) i liczby działek na osi.

- **Trigger**

W grupie tej należy podać rejestr sterownika OCS (1-bitowy), sterujący tym obiektem (obliczanie danych, wyświetlanie).

Charakterystyka działania obiektu:

- **Funkcjonalność**

Po wyzwoleniu, obiekt ten rysuje dane dla każdego ze skonfigurowanych pisaków, wykorzystując do tego celu liczbę rejestrów podanych w polu 'Number of values to plot' i rozpoczynając od



zadanego rejestru. Rysowanie jest rozpoczynane na osi Y, a kolejne punkty są proporcjonalnie rozstawione i łączone.

Każdy rejestr sterownika jest traktowany jako 16-bitowa wartość bez znaku, a jego zawartość jest skalowana w osi pionowej (i obcinana) do wartości Y-Min i Y-Max, zdefiniowanych w oknie dialogowym 'Axis Dialog'.

W edytorze graficznym, na obiekcie pokazywane są w białym prostokącie szerokość i wysokość wykresu. Na podstawie pokazywanej szerokości użytkownik może wyznaczyć liczbę próbek, jednocześnie pokazywanych na wykresie.

### Opcjonalne atrybuty wyświetlania

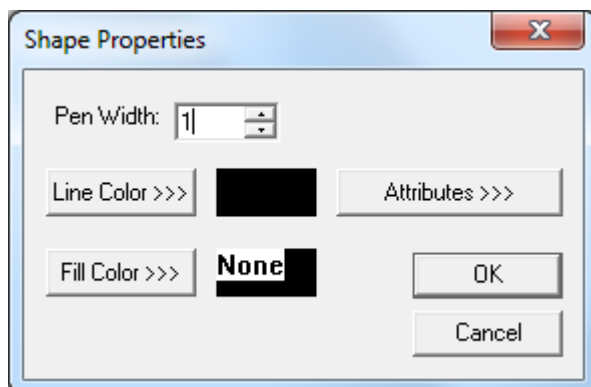
Border

## Pasek narzędziowy Drawing Primitives

Przy tworzeniu dekoracyjnego tła, ramek i kształtów, użytkownik może korzystać z następujących podstawowych elementów graficznych:

- Prostokąt
- Okrąg
- Zaokrąglony prostokąt
- Linia

Obiekty te można układać w warstwach i są one aktualizowane w momencie aktualizacji dynamicznego parametru wyświetlania obiektu umieszczonego pod spodem. Należy zwrócić uwagę, że każdy z tych kształtów jest wliczany do liczby obiektów na stronie.



### Właściwości kształtu

- Dla każdego z tych obiektów można konfigurować szerokość pisaka oraz kolor rysowania. W przypadku obiektów posiadających wnętrze, można definiować kolor wypełnienia.
- Jeżeli dla obiektu wybrano wypełnienie przezroczyste (Tran), nie można takiego obiektu zaznaczyć poprzez kliknięcie w jego wnętrzu. W celu zaznaczenia, należy ustawić kursor na krawędzi linii obiektu.

---

## Przykładowa kolejność czynności przy projektowaniu ekranów graficznych

---

Poniżej zamieszczono przykładową kolejność przy projektowaniu ekranów graficznych.

1. Po pierwsze należy wybrać cele nadrzędne realizowane przez system wizualizacji graficznej.
2. Należy wskazać procesy i zdarzenia, które powinny być przedstawiane w formie graficznej. (Które ze zautomatyzowanych urządzeń wejść/wyjść przesyłają informacje zwrotne?)
3. Wyznaczyć dane potrzebne operatorowi oraz zdarzenia, które powinny być monitorowane. (Jakie dane będą prawdopodobnie wymagane przez operatora? Kiedy? Jakie zdarzenia muszą być monitorowane i/lub wymagają reakcji ze strony operatora?)
4. Wybrać urządzenia wejść/wyjść oraz przypisać do każdego z nich nazwę.
5. W tym momencie programista może zacząć pisać program sterujący za pomocą oprogramowania Cscape. Fazy pisania programu sterującego i projektowania ekranów graficznych mogą być realizowane jednocześnie. Należy się zastanowić: (Jakie informacje i urządzenia i jakie dane muszą być wyświetlane na każdym z ekranów? Czy zadaniem strony jest obrazowanie zdarzenia zaistniałego w procesie (np. strona alarmów sygnalizująca przerwanie pracy przez maszynę)? Czy strona jest projektowana w celu wprowadzenia danych przez operatora? Czy strona ma na celu wymuszenie pewnych działań operatora, np. zatwierdzenie alarmu?).