

Politechnika Świętokrzyska
Katedra Urządzeń i Systemów Automatyki

Sterowniki PLC Instrukcja Laboratoryjna

Ćwiczenie 3

Programowanie Sterowników Simatic S7 – 1200

Układy kombinacyjne, liczniki, timery, komparatory

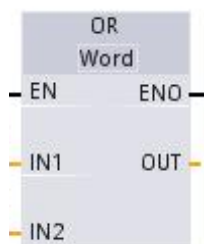
Robert Kazała
Tomasz Kwaśniewski
Michał Łaskawski
Miroslaw Wciślik
Paweł Zagniński
Karol Suchenia

Instrukcje AND, OR i XOR

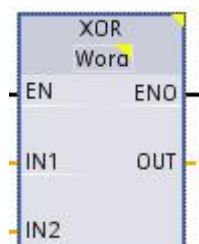
- AND: logiczna operacja AND dla danych typu BYTE, WORD i DWORD
- OR: logiczna operacja OR dla danych typu BYTE, WORD i DWORD
- XOR: logiczna operacja XOR dla danych typu BYTE, WORD i DWORD



IN1	IN2	OUT
0	0	0
0	1	0
1	0	0
1	1	1



IN1	IN2	OUT
0	0	0
0	1	1
1	0	1
1	1	1



IN1	IN2	OUT
0	0	0
0	1	1
1	0	1
1	1	0

Parametr	Typ danych	Opis
IN1, IN2	BYTE, WORD, DWORD	Wejścia logiczne
OUT	BYTE, WORD, DWORD	Wyjście logiczne

Podczas wyboru typu danych jest ustawiany taki sam typ danych parametrów IN1, IN2 i OUT. Odpowiadające sobie bity IN1 i IN2 są argumentami operacji logicznej, której wynik jest wpisywany do OUT. Po zakończeniu wykonywania powyższych instrukcji, ENO ma zawsze wartość TRUE.

Przykład 1

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.2 wartość logiczną „1” i poprawnym wykonaniu operacji logicznej AND dla rejestrów 100 i 102.

IN1=01000101 IN2=00011100

OUT=00000100

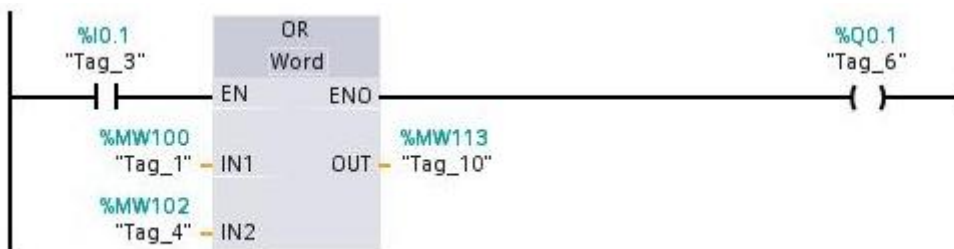


Przykład 2

Wyjście logiczne Q0.1 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.1 wartość logiczną „1” i poprawnym wykonaniu operacji logicznej OR dla rejestrów 100 i 102.

IN1=01000101 IN2=00011100

OUT=01011101

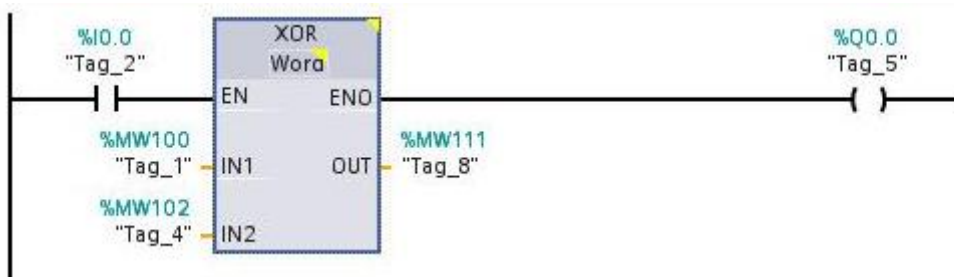


Przykład 3

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.0 wartość logiczną „1” i poprawnym wykonaniu operacji logicznej XOR dla rejestrów 100 i 102.

IN1=01000101 IN2=00011100

OUT=01011001



Instrukcja inwersji

Instrukcja INV jest stosowana do wyznaczenia dwójkowego uzupełnienia do jedności parametru IN. Uzupełnienie do jedności jest wykonywane poprzez inwersję każdego bitu parametru IN (zamianę każdego 0 na 1 i 1 na 0). Po zakończeniu wykonywania instrukcji, ENO ma zawsze wartość TRUE.

IN	OUT
0	1
1	0



Parametr	Typ danych	Opis
IN	SINT, INT, DINT, USINT, UINT, UDINT, BYTE, WORD, DWORD	Element podlegający inwersji
OUT	SINT, INT, DINT, USINT, UINT, UDINT, BYTE, WORD, DWORD	Wyjście po inwersji

Przykład 4

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.3 wartość logiczną „1” i poprawnym wykonaniu operacji inwersji rejestru 100.

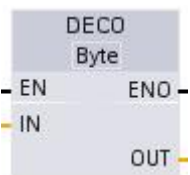
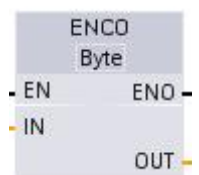
IN=01000101

OUT=10111010



Instrukcje enkodowania i dekodowania

- ENCO koduje ciąg bitów na liczbę dwójkową.
- DECO dekoduje liczbę dwójkową na ciąg bitów.



Parametr	Typ danych	Opis
IN	ENCO: BYTE, WORD, DWORD DECO: UINT	ENCO: ciąg bitów do zakodowania DECO: wartość do dekodowania
OUT	ENCO: INT DECO: BYTE, WORD, DWORD	ENCO: wartość zakodowana DECO: ciąg bitów zdekodowany

Instrukcja ENCO dokonuje konwersji parametru IN na liczbę dwójkową odpowiadającą ciągowi bitów znajdujących się na najmniej znaczącej pozycji w IN i zapisuje wynik jako parametr OUT. Jeżeli parametr IN jest równy 0000 0001 albo 0000 0000, to do OUT jest wpisywane 0. Jeśli parametr IN ma wartość 0000 0000 to ENO przyjmuje wartość FALSE.

Instrukcja DECO dekoduje liczbę dwójkową z parametru IN poprzez ustawianie bitów znajdujących się na odpowiadających pozycjach OUT na 1 (wszystkie pozostałe bity są ustawione na 0). Po zakończeniu wykonywania instrukcji, ENO ma zawsze wartość TRUE.

Przykład 5

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.3 wartość logiczną „1” i poprawnym wykonaniu operacji dekodowania.

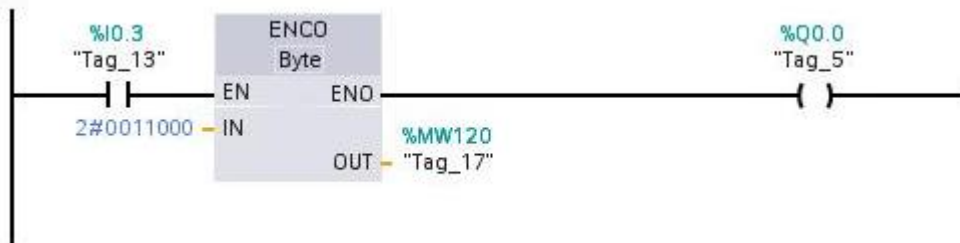
IN=111 OUT=10000000



Przykład 6

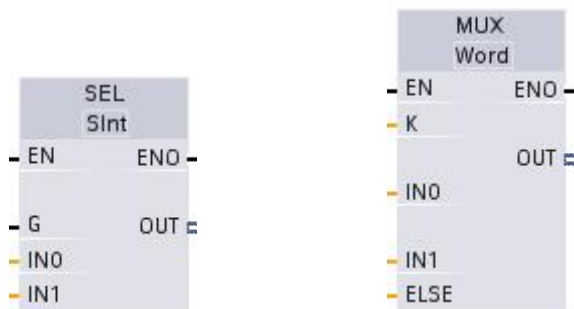
Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki po ustawieniu na wejściu I0.3 wartość logiczną „1” i poprawnym wykonaniu operacji enkodowania.

IN=0011000 OUT=3



Instrukcje wyboru (SEL) i multipleksowania (MUX)

- SEL, w zależności od wartości parametru G, przypisuje jedną z dwóch wartości wejściowych parametrowi OUT.
- MUX, w zależności od wartości parametru K, przypisuje jedną z wielu wartości wejściowych parametrowi OUT. Jeżeli parametr K wykracza poza dozwolony zakres, to parametrowi OUT jest przypisywana wartość parametru ELSE.



Parametry dla SEL

Parametr	Typ danych	Opis
G	BOOL	Przełącznik selektora: FALSE dla IN0, TRUE dla IN1

IN1, IN2	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wejścia
OUT	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wyjście

Parametry dla MUX

Parametr	Typ danych	Opis
K	UINT	Przełącznik selektora: 0 dla IN0, 1 dla IN1
IN0, IN1 ...	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wejścia
ELSE	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wartość wejściowa dla podstawienia (opcjonalnie)
OUT	SINT, INT, DINT, USINT, UINT, UDINT, REAL, BYTE, WORD, DWORD, TIME, CHAR	Wyjście

Zmienne wejściowe i zmienna wyjściowa muszą być tego samego typu danych.

- Instrukcja SEL zawsze wybiera pomiędzy dwoma wartościami wejściowymi.
- Instrukcja MUX wybrana po raz pierwszy w edytorze programu ma dwa parametry wejściowe IN, ale można ją rozszerzyć dodając więcej parametrów IN. Do dodawania i usuwania parametrów wejściowych instrukcji MUX służy następująca metoda:
 - W celu dodania wejścia należy kliknąć prawym klawiszem myszy na końcówkę wejściową jednego z istniejących parametrów IN i wybrać komendę „insert input”.
 - W celu usunięcia wejścia należy kliknąć prawym klawiszem myszy na końcówkę wejściową jednego z istniejących parametrów IN (jeżeli jest więcej wejść niż oryginalne dwa) i wybrać komendę „Delete”.

Przykład 7

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki, po ustawieniu na wejściu I0.3 wartość logiczną „1”. Na wyjściu OUT w zależności od stanu G pojawi się IN0 dla stanu niskiego, IN1 dla stanu wysokiego.

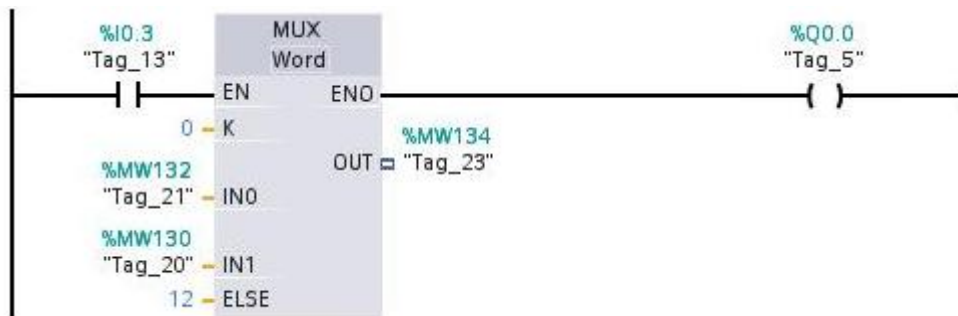


Przykład 8

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki, po ustawieniu na wejściu I0.3 wartość logiczną „1”.

K=0 IN0=10 IN1=20

OUT=20



Instrukcja Shift

Instrukcja przesunięcia (shift) jest stosowana bitów parametru IN. Wynik jest przypisany parametrowi OUT. Parametr N określa o ile pozycji bitów ma nastąpić przesunięcie.

- SHR: przesunięcie bitów w prawo.
- SHL: przesunięcie bitów w lewo.



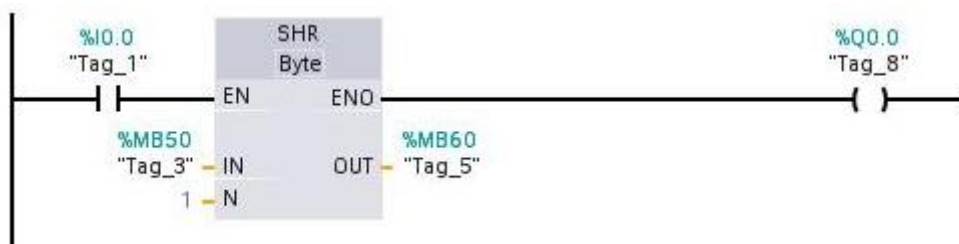
Parametr	Typ danych	Opis
IN	BYTE, WORD, DWORD	Bity do przesunięcia
N	UINT	Liczba pozycji bitów do przesunięcia
OUT	BYTE, WORD, DWORD	Bity po operacji przesunięcia

- Dla N = 0 nie jest wykonywane przesunięcie i do OUT jest przypisywana wartość IN.
- Na pozycje opróżnione podczas przesuwania są wpisywane zera.
- Jeżeli liczba pozycji do przesunięcia (N) przekracza liczbę bitów wartości docelowej (8 dla BYTE, 16 dla WORD i 32 dla DWORD), to oryginalna wartość zniknie i zostanie zastąpiona przez zera (do OUT zostaną wpisane same zera).
- Dla operacji przesuwania, ENO ma zawsze wartość TRUE.

Przykład 9

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki, a zawartość rejestru MB50 zostanie przesunięta o 1 bit w prawo.

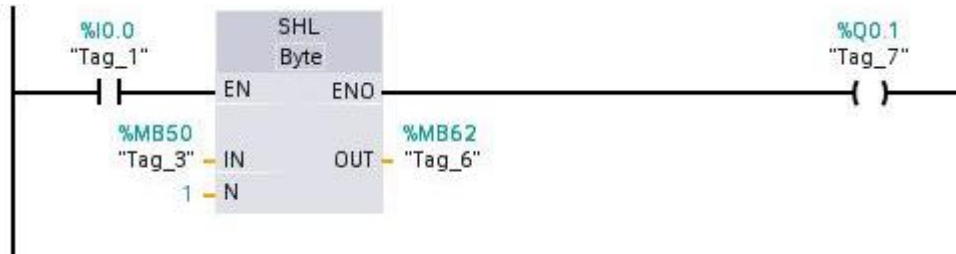
IN=10010101 OUT=01001010



Przykład 10

Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki, a zawartość rejestru MB50 zostanie przesunięta o 1 bit w lewo.

IN=10010101 OUT=00101010



Instrukcja Rotate

Instrukcje obrotu są stosowane do cyklicznego przesuwania bitów parametru IN. Wynik jest przypisywany do parametru OUT. Parametr N określa o ile pozycji bitów ma nastąpić obrót.

- ROR: obrót bitów w prawo.
- ROL: obrót bitów w lewo.



Parametr	Typ danych	Opis
IN	BYTE, WORD, DWORD	Bity do obrotu
N	UINT	Liczba pozycji bitów do obrotu
OUT	BYTE, WORD, DWORD	Bity po operacji obrotu

- Dla N = 0 nie jest wykonywany obrót i do OUT jest przypisywana wartość IN.
- Podczas obrotu bity wysuwane z jednej strony trafiają na pozycje opróżniane z drugiej strony parametru docelowego; zatem żaden oryginalny bit nie jest tracony.
- Jeżeli liczba pozycji do przesunięcia (N) przekracza liczbę bitów wartości docelowej (8 dla BYTE, 16 dla WORD i 32 dla DWORD), to obrót jest nadal wykonywany.
- Dla operacji obrotu, ENO ma zawsze wartość TRUE.

Przykład 11

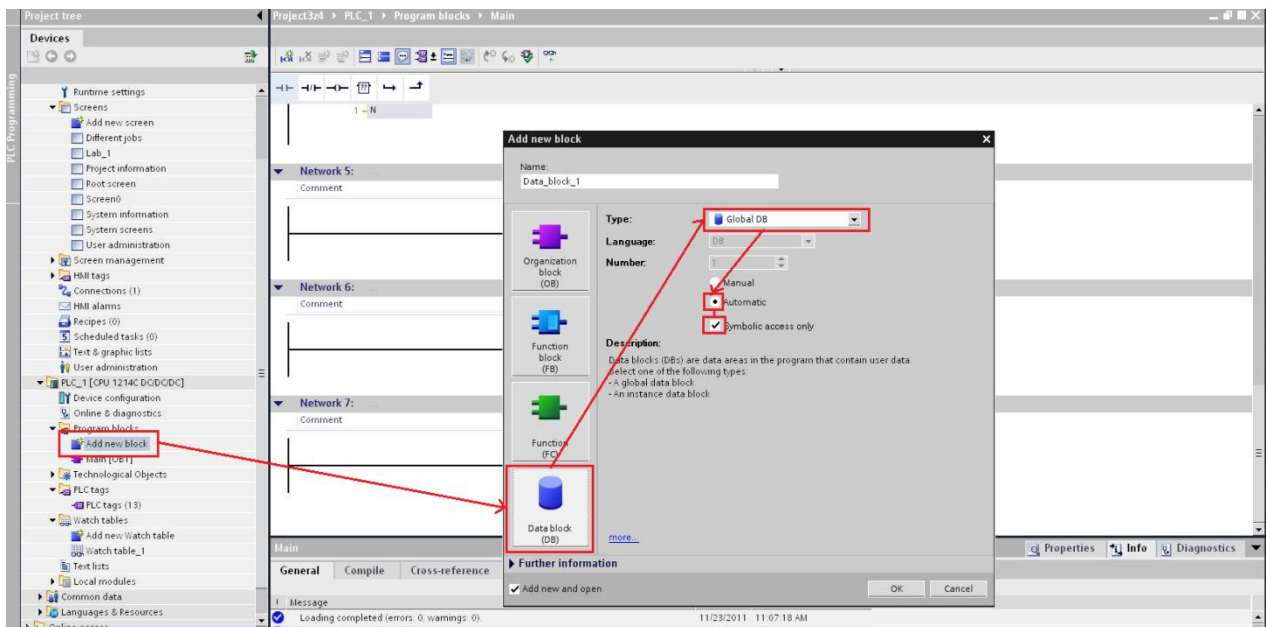
Wyjście logiczne Q0.0 zostanie ustawione w stan wysoki, a zawartość rejestru MB50 zostanie przesunięta o 1 bit w prawo.

IN=10010101

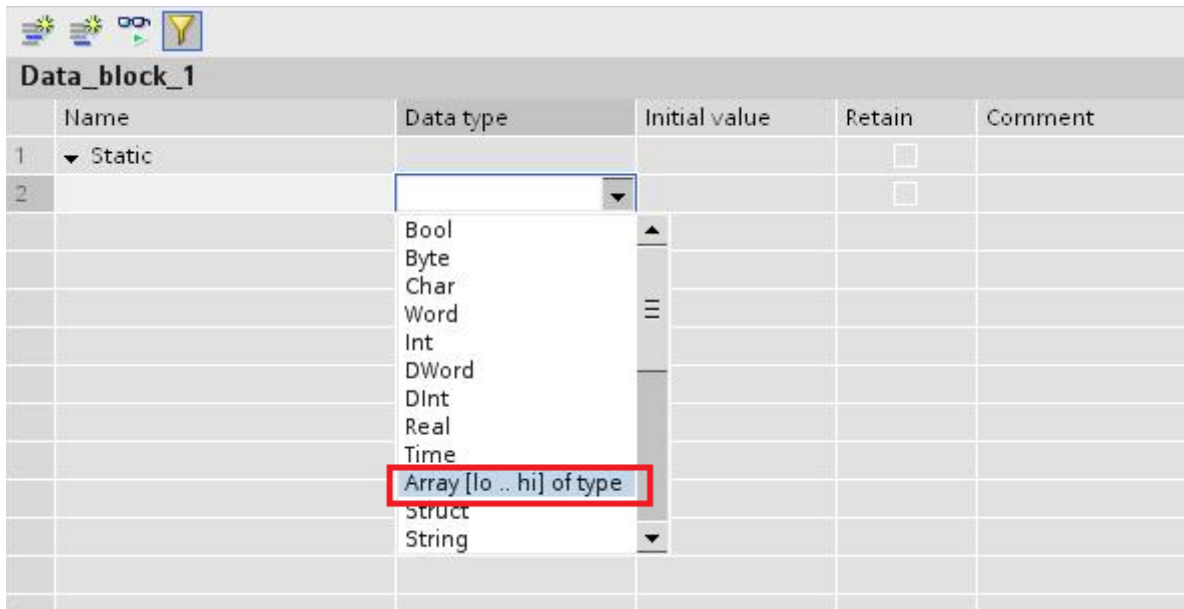
OUT=11001010



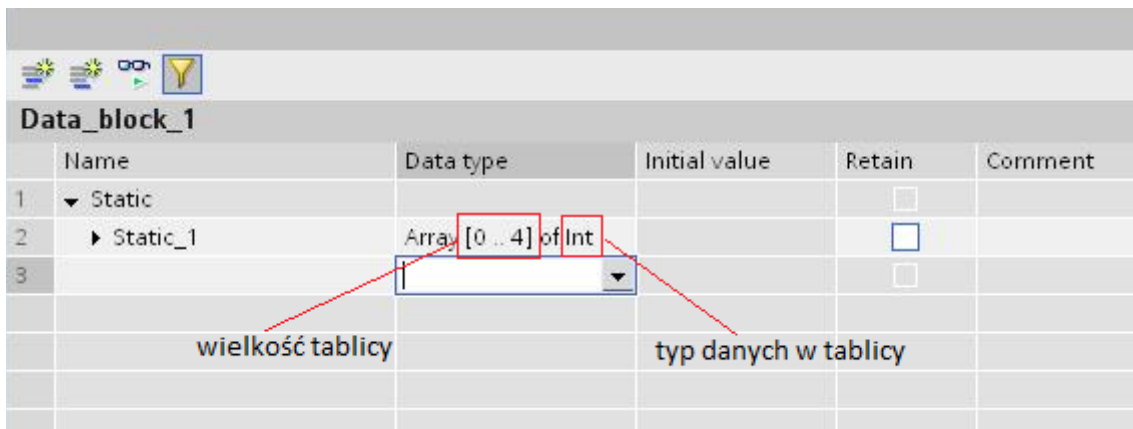
Implementacja tablicy



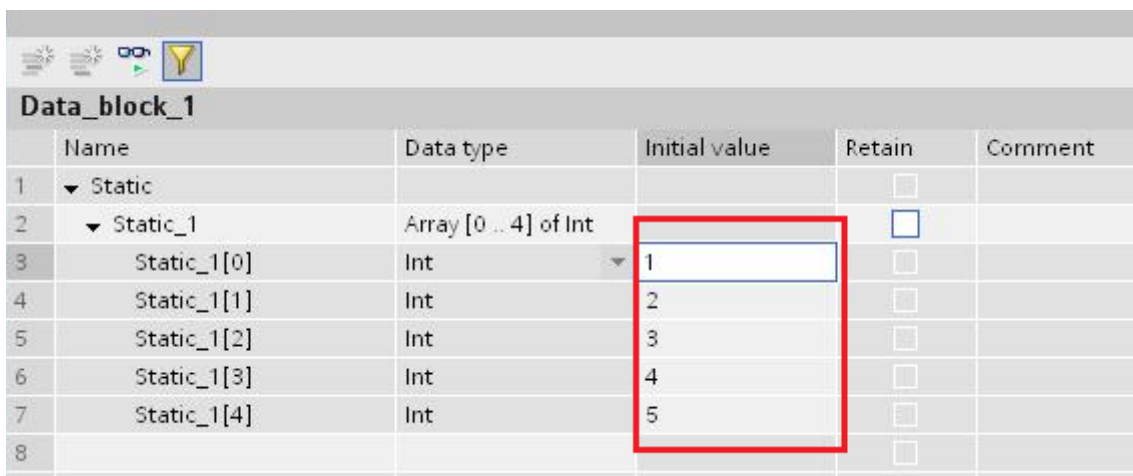
- kliknij 'Add new block'
- kliknij 'Data block (DB)'
- wybierz 'Global DB'
- zaznacz 'Automatic' i 'Symbolic access only'



- wybierz typ danych 'Array [lo .. hi] of type'



- wpisz wielkość tablicy i typ danych



- uzupełnić tabelę wartościami

PRZEBIEG ĆWICZENIA

Zadanie 1.

Uruchomić przykłady znajdujące się w instrukcji.

Zadanie 2.

Zrealizować rotacje bitów prawo-lewo poprzez zmianę stanu wejść.

Zadanie 3.

Zrealizuj dekodery 2/4 (2 wejścia 4 wyjścia) za pomocą styków.

Wejścia		Wyjścia							
A	B	1 z 4				NOT (1 z 4)			
		3	2	1	0	3	2	1	0
0	0	0	0	0	1	1	1	1	0
1	0	0	0	1	0	1	1	0	1
0	1	0	1	0	0	1	0	1	1
1	1	1	0	0	0	0	1	1	1

Zadanie 4.

Zrealizuj daną tabelę za pomocą multipleksera i tablic.

Zadanie 5.

Napisać program krokowego wpisywania do tablicy wartości pozycji bitu stanu wysokiego podanego rejestru.

Literatura

[1] Simatic S7-1200 – Programowalny sterownik S7-1200, Warszawa 2010