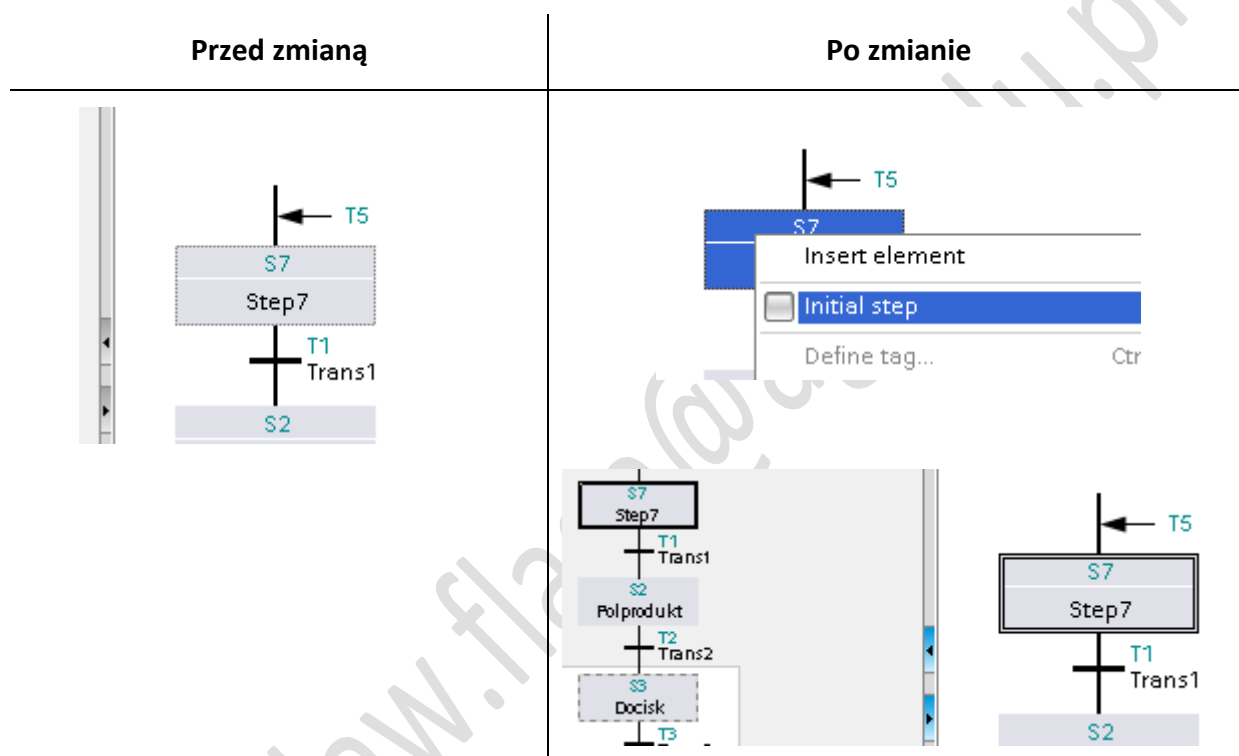


Initial Step – krok inicjujący sekwenser

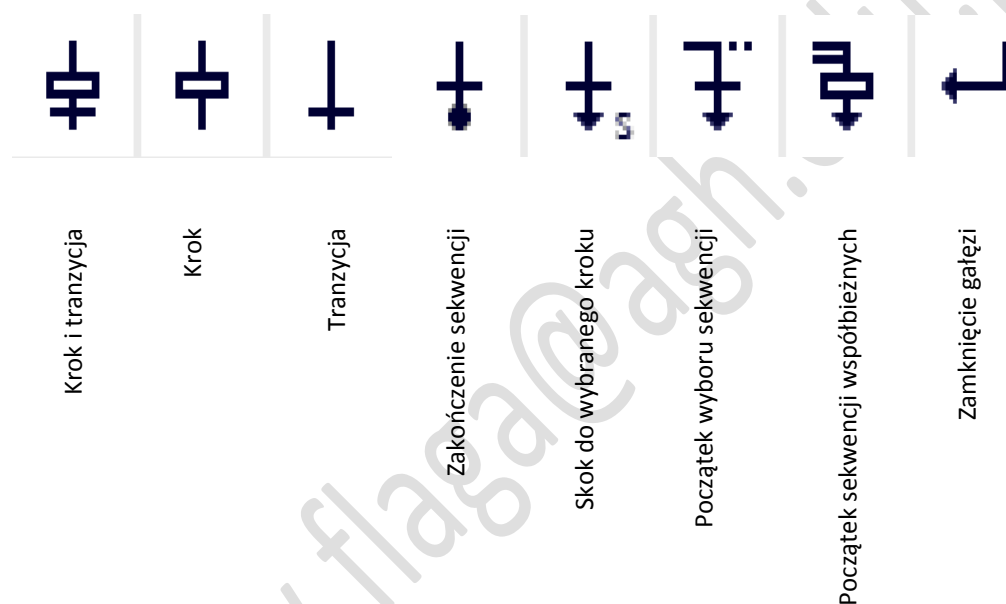
W ferworze walki czasem usuniemy krok inicjujący (po rozpoczęciu FB z GRAPH jest on standardowo oznaczony S1). Skutkuje to tym, że wszystko wygląda dobrze, a sekwenser nie działa – nie inicjuje się. Należy wówczas wybranemu krokowi nadać status kroku inicjującego – nie musi to być pierwszy krok w sekwencji – pojawi się podwójna ramka, którą czasem przy dużych rozdzielczościach niezbyt dokładnie widać.



SFC

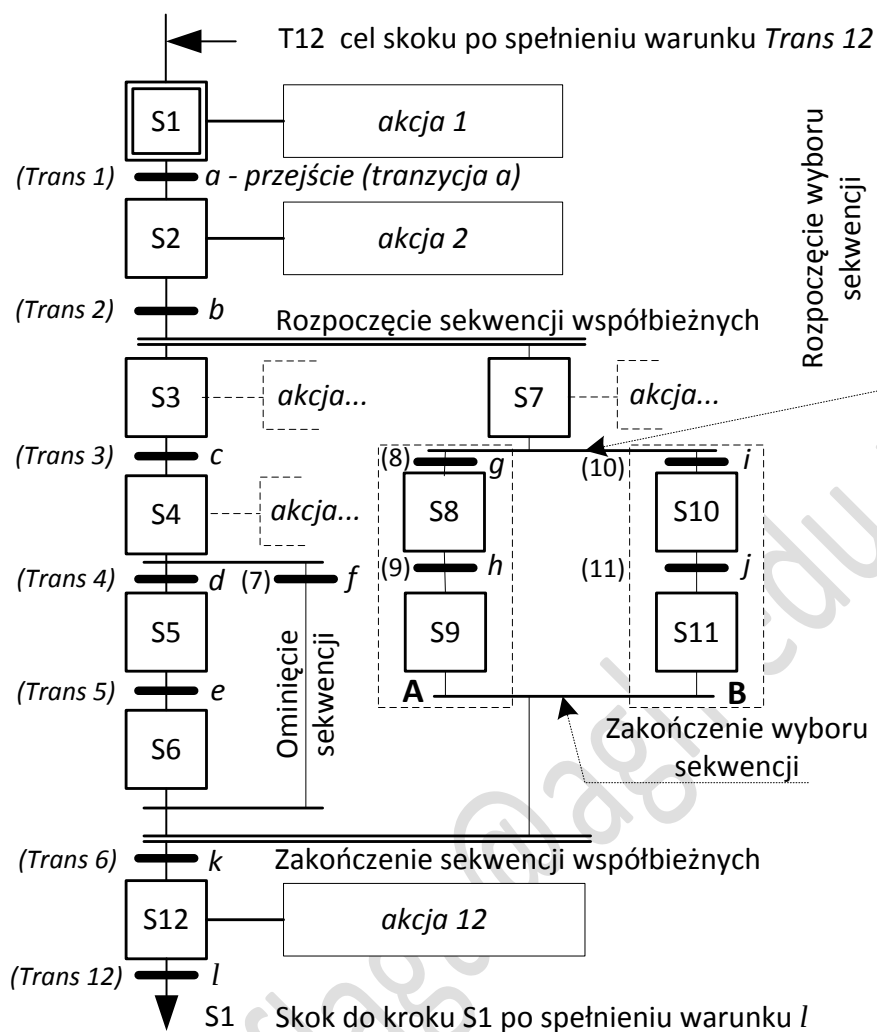
Graf sekwencji SFC zakłada dekompozycję procesu sekwencyjnego na rozdzielne operacje wykonywane w kolejnych etapach (krokach). Kolejny krok może zostać wykonany, jeśli prawdziwy jest warunek przejścia (tranzycja) i wykonał się krok poprzedni. Implementację SFC w postaci narzędzia graficznego możemy znaleźć np. w oprogramowaniu firmy Siemens pod nazwą GRAPH7.

Do programowanie w GRAPH7 wykorzystujemy następujące symbole:



Na Rys. 1 przedstawiono podstawowe symbole i budowane z nich struktury. Kwadraty z liczbą poprzedzoną literą **S** w środku (tutaj od S1 do S12) nazywane są krokami lub etapami. Obrazują one zbiory stanów lub akcji do wykonania. Dokładniejszy opis akcji, skojarzonych z każdym etapem, może znajdować się w prostokącie po prawej stronie kwadratu z numerem etapu (tutaj *akcja ...*).

Każda sieć SFC powinna posiadać tylko jeden element określony zbiorem etapów aktywnych w chwili początkowej oraz początkowymi wartościami jego stanów wewnętrznych i wyjść. Krok aktywny w chwili początkowej wyróżniony jest przez kwadrat rysowany podwójną kreską (ta konwencja nie zawsze jest zachowywana). Na omawianym rysunku jest to krok z numerem S1.



Rys. 1. Przykład grafu SFC

Połączenia pomiędzy kolejnymi krokami w postaci graficznej przedstawiamy jako pionowe odcinki łączące kolejne kroki (etapy). Kierunek przepływu sterowania w odcinkach łączących przyjmowany jest domyślnie jako z góry (wejście) do dołu (wyjście).

Przejścia nazywane tranzycjami (ang. transition) – zobrazowane graficznie krótkimi poziomymi, grubymi odcinkami, prostopadłymi do linii łączącej kolejne kroki, ponumerowane zostały liczbami, poprzedzonymi słówkiem *Trans* (konwencja notacji Siemens), umieszczonymi w nawiasach po lewej stronie znaku. Są one aktywne, jeśli skojarzone z nimi warunki logiczne (małe litery alfabetu od *a* do *k*, po prawej stronie kreski obrazującej warunek przejścia) przyjmą stan ON. Oprócz tego na rysunku widzimy rozpoczęcie i zakończenie sekwencji procesów współbieżnych (podwójne poziome kreski) oraz wyboru sekwencji (pojedyncze poziome kreski).

Na Rys. 1 pokazano również skok, w tym przypadku powrót do kroku S1, w celu zapętlenia procesu. Przy strzałce w dół poniżej kroku S12

Opis przykładu z rysunku 1

Opis przykładowego systemu przedstawionego na Rys. 1 jest następujący.

W chwili początkowej wykonywany jest krok S1 aż do chwili, w której zostanie spełniony warunek $a = 1$. Spełnienie warunku przejścia (*Trans 1*) spowoduje uruchomienie kroku S2, krok S1 przestaje być aktywny. Kolejne wykonywanie kroków analogiczne do przejścia pomiędzy krokami S1 i S2 nazywamy sekwencją prostą.

Aktywowanie przejścia (*Trans 2*) poprzez nadanie wartości 1 zmiennej b spowoduje rozpoczęcie wykonywania sekwencji współbieżnych, które rozpoczynają się poniżej podwójnej poziomej kreski, jedna od kroku S3, a druga od kroku S7. Prawa sekwencja rozpoczyna się od kroku S7, po jego wykonaniu dochodzi do struktury nazywanej wyborem sekwencji (pozioma kreska). W tym przypadku dochodzi do wyboru sekwencji A lub B. Wykonana zostanie ta sekwencja, której tranzycja stanie się aktywna. Implementacja tej sekwencji musi być tak zorganizowana, aby nie można było dokonać więcej niż jednego wyboru. Zakładając, że zmienna $i = 1$, wykonana zostanie sekwencja prosta kroków S10 i S11, co spowoduje dojście tej gałęzi do zakończenia sekwencji współbieżnych.

Wracając do kroku S3 można zauważyć, że przy aktywnym przejściu (*Trans 3*) wykonana się sekwencja prosta kroków S3 i S4, a następnie rozpocznie się szczególny przypadek wyboru sekwencji nazywany ominięciem sekwencji. Cechą charakterystyczną tego przypadku jest to, że jedna z gałęzi nie zawiera żadnych kroków. W pokazanym przypadku, jeśli $f = 1$, to kroki S5 i S6 nie zostaną wykonane.

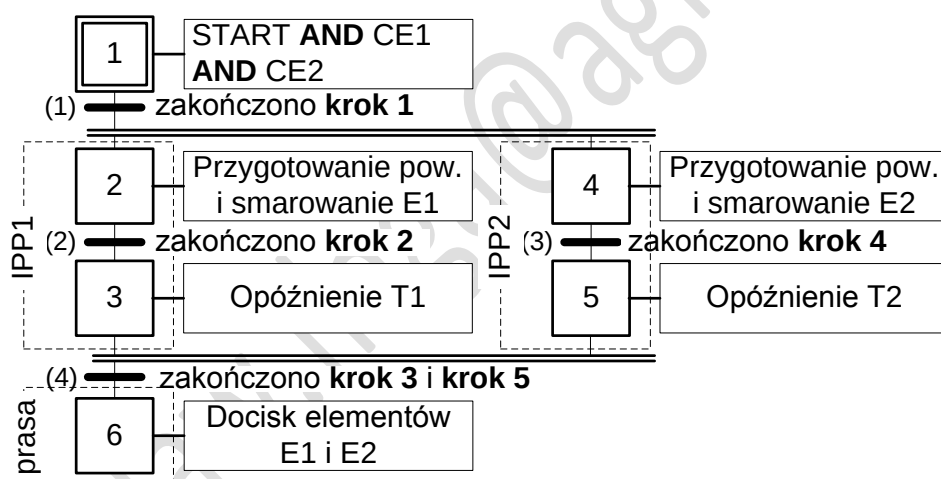
Jeśli obie sekwencje współbieżne wykonają swoje zadania i zajdzie warunek $k = 1$, to nastąpi wykonanie kroku S12, po którym, jeżeli zajdzie warunek $l = 1$ (aktywność przejścia *Trans 12*), system powinien wykonać skok do kroku S1.

Omówienie przykładu rysunkowego ma posłużyć zademonstrowaniu najczęściej spotykanych struktur przy modelowaniu systemu sekwencyjnego z użyciem SFC.

Opis przykładu z rysunku 2

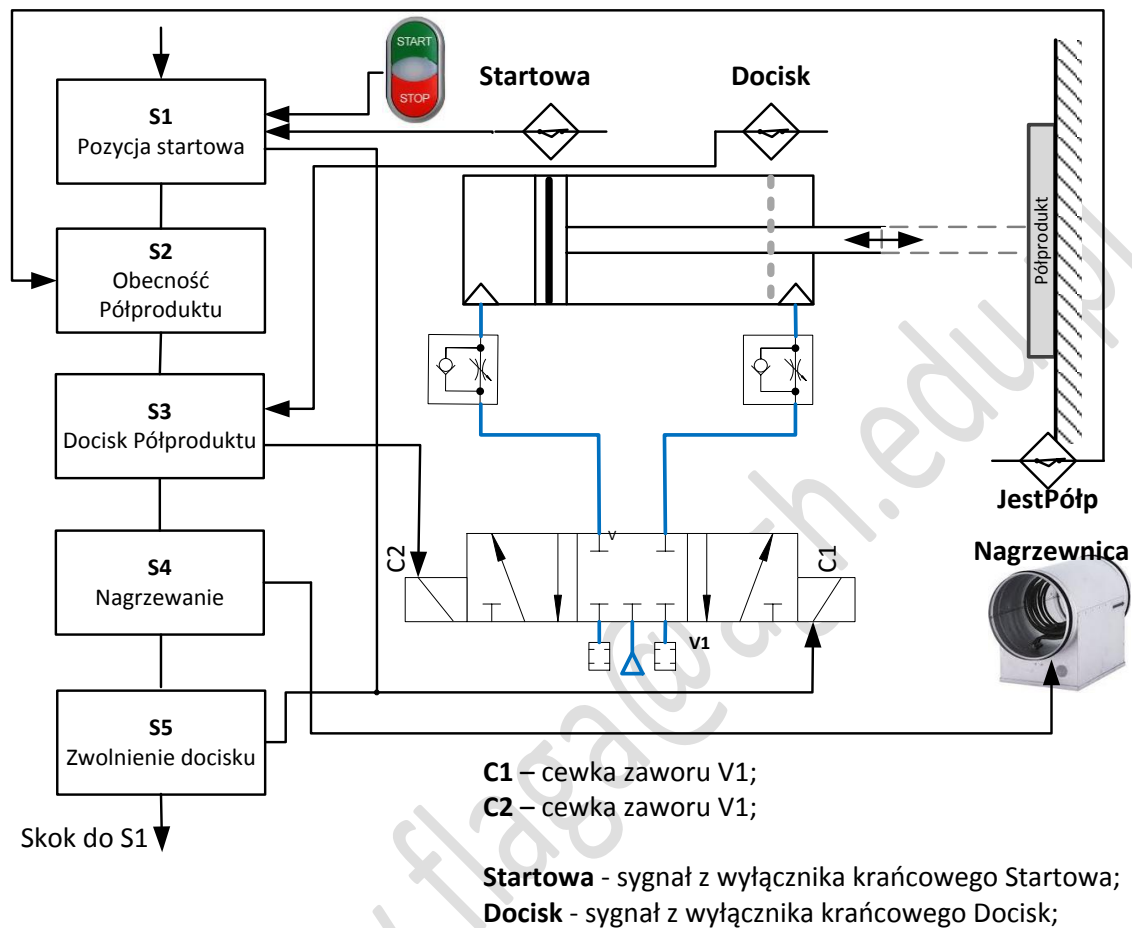
Na Rys. 2 przedstawiony został graf SFC pewnego procesu.

Zadaniem postawionym przed tym przykładowym procesem jest równoczesne przygotowanie powierzchni dwóch elementów E1 i E2, naniesienie na nie warstwy kleju, odczekanie zadanego interwału czasu T1 i T2 potrzebnego na odparowanie składowych lotnych kleju i dociśnięcie sklejanych powierzchni. Warunki początkowe pozwalające na rozpoczęcie procesu określone są przez sygnały pochodzące z czujników obecności elementów CE1 i CE2 w instalacjach przygotowania powierzchni i nanoszenia warstwy kleju (IPP1, IPP2) oraz pojawieniu się impulsu pochodzącego od przycisku sterowniczego START umieszczonego na konsoli operatora. Przykład jest mocno strywializowany przez pominięcie wielu czujników i układów wykonawczych, czy choćby przycisku sterowniczego odpowiadającego za zatrzymanie awaryjne. Jest to jednak konieczne ze względu na czytelność i objętość przykładu.



Rys. 2. Graf SFC przykładowego procesu

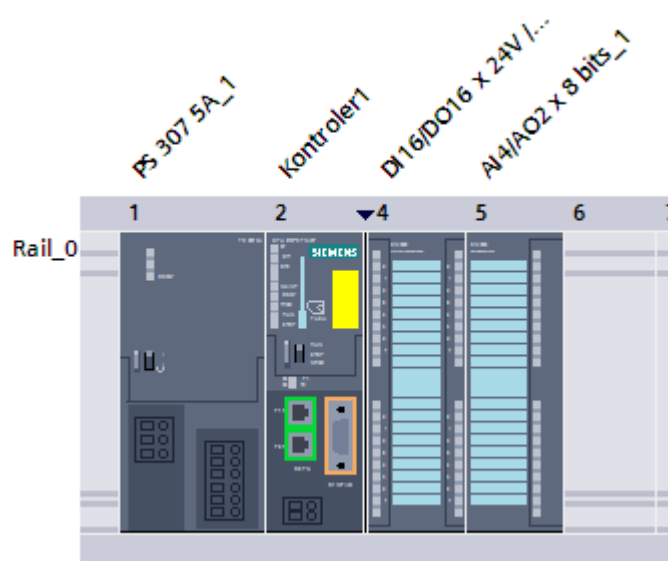
Realizacja przykładowego programu z użyciem SFC i oprogramowania TIA 12



Rys. 3. Przykłada do realizacji

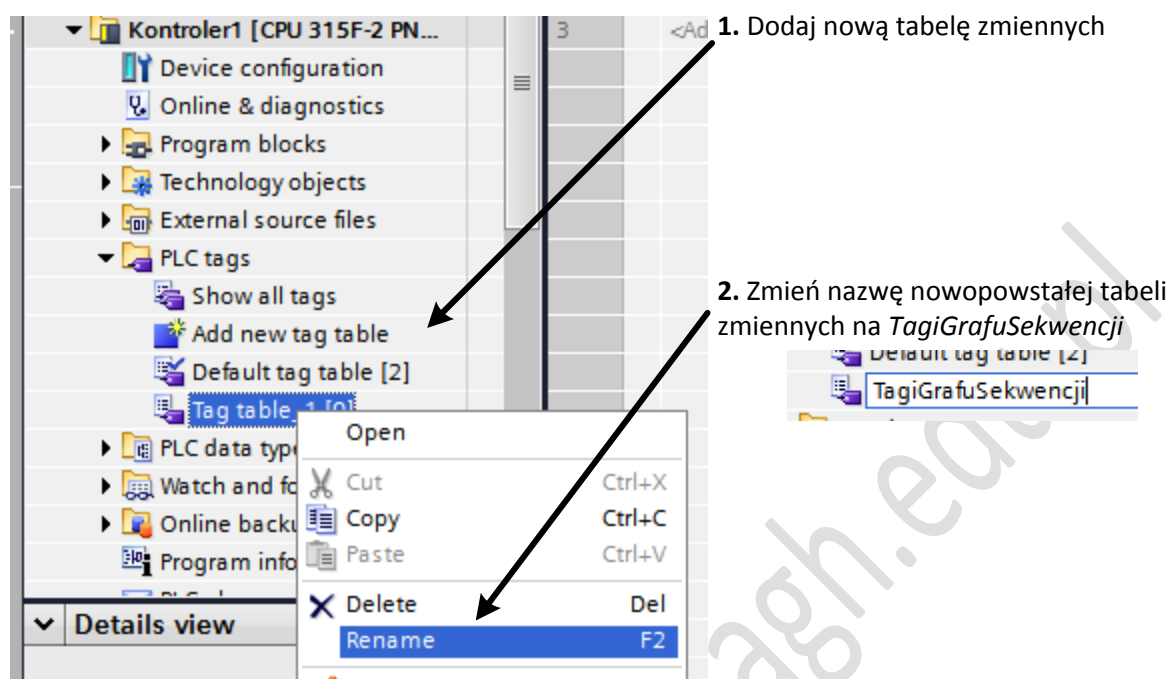
Krok	Opis	Nazwa
S1	Pierwszym krokiem jest zawsze krok startowy (inicjujący) – tutaj stan wysoki markera informującego o pojawieniu się stanu wysokiego zmiennej START. Sprawdzenie czy siłownik znajduje się w położeniu Starowym – jeżeli nie to ustawienie siłownika w tym położeniu	Home
S2	Sprawdzanie obecności półproduktu	Półprodukt
S3	Uruchomienie siłownika dociskającego półprodukt i załączenie nagrzewnicy.	Docisk
S4	Nagrzewanie przez określony czas (np. 10 s)	Nagrzewanie
S5	Zwolnienie docisku – powrót siłownika do położenia początkowego. Zakończenie procesu i powrót do pozycji Startowej	Zwolnienie

Konfiguracja sterownika S7-300



Module	Rack	Slot	I address	Q address	Type	Order no.
PS 307 5A_1	0	1			PS 307 5A	6ES7 307-1EA01-0AA0
▼ Kontroler1	0	2			CPU 315F-2 PN/DP	6ES7 315-2FJ14-0AB0
MPI/DP interface_1	0	2 X1	2047*		MPI/DP interface	
► PROFINET interface_1	0	2 X2	2046*		PROFINET interface	
	0	3				
DI16/DO16 x 24V / 0.5A_1	0	4	0...1	0...1	DI16/DO16 x 24V / ...	6ES7 323-1BL00-0AA0
AI4/AO2 x 8 bits_1	0	5	272...279	272...275	AI4/AO2 x 8 bits	6ES7 334-0CE01-0AA0

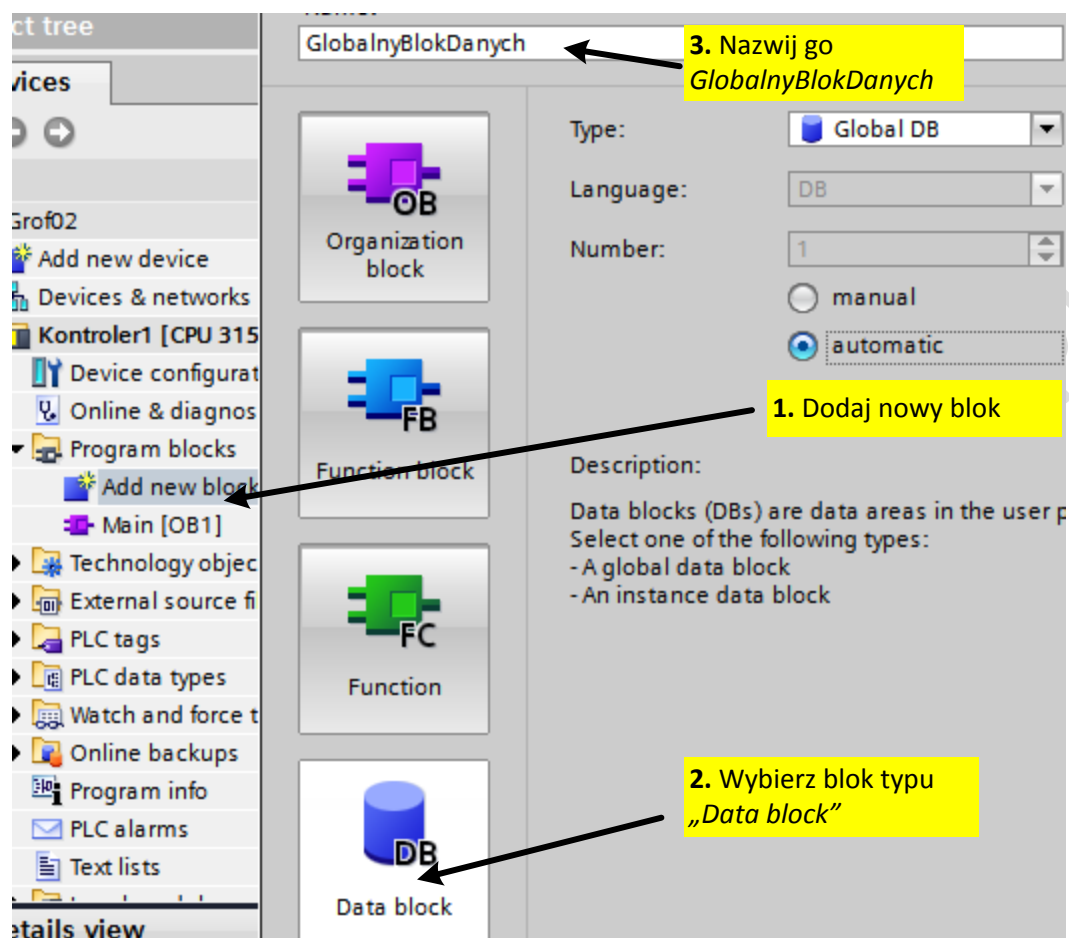
Tworzenie tabeli zmiennych



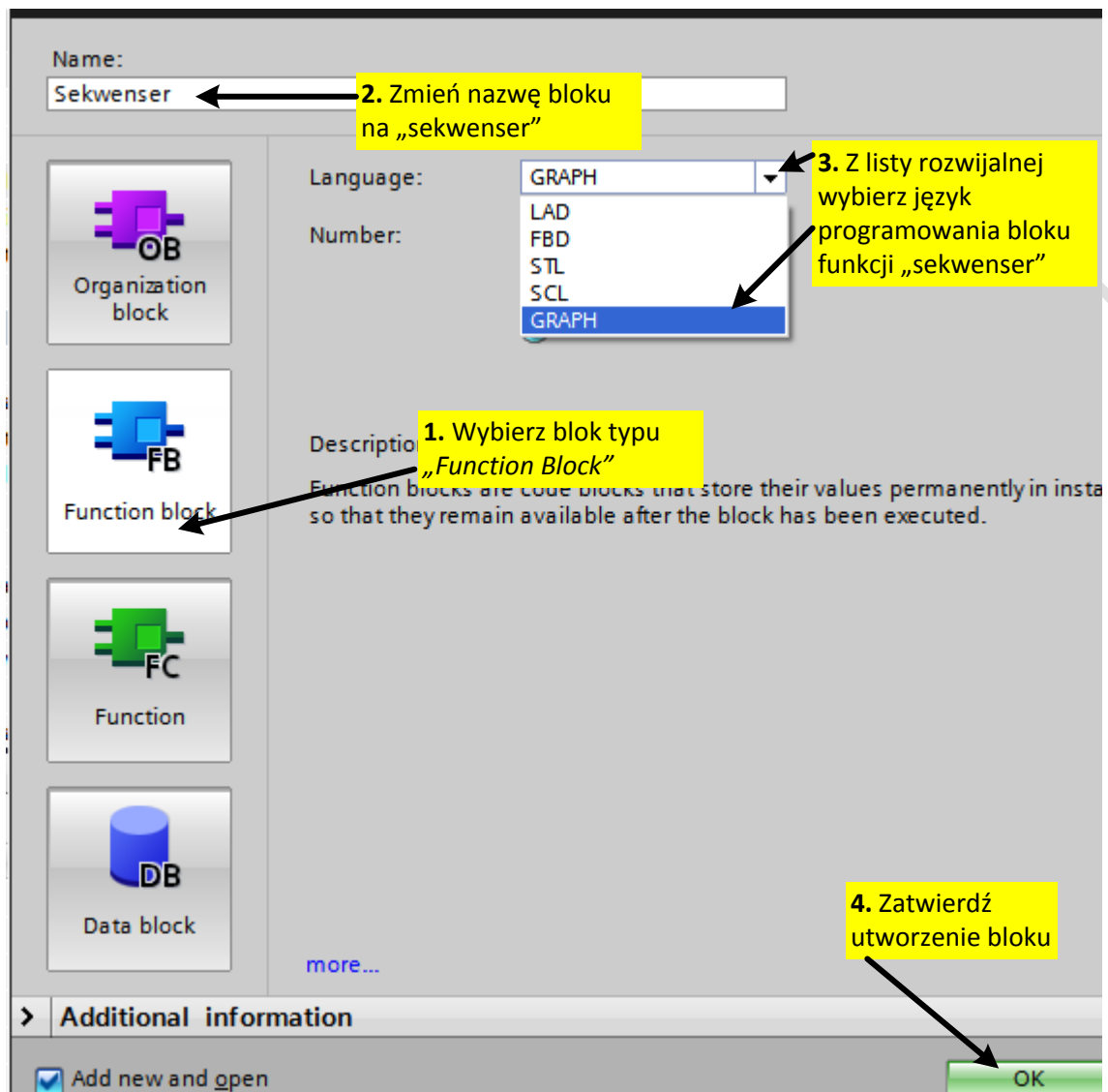
Dodawanie Tagów

	Name	Data type	Address ▲
1	Start	Bool	%I0.0
2	Stop	Bool	%I0.1
3	Startowa	Bool	%I0.2
4	Docisk	Bool	%I0.3
5	JestPolprodukt	Bool	%I0.4
6	DoStartowej	Bool	%Q0.1
7	DoDocisk	Bool	%Q0.2
8	Nagrzewnica	Bool	%Q0.3
9	BylStart	Bool	%M0.0
10	GRAPH_blad_globalny	Bool	%M0.1
11	GRAPH_licznik_docisk	Word	%MW2

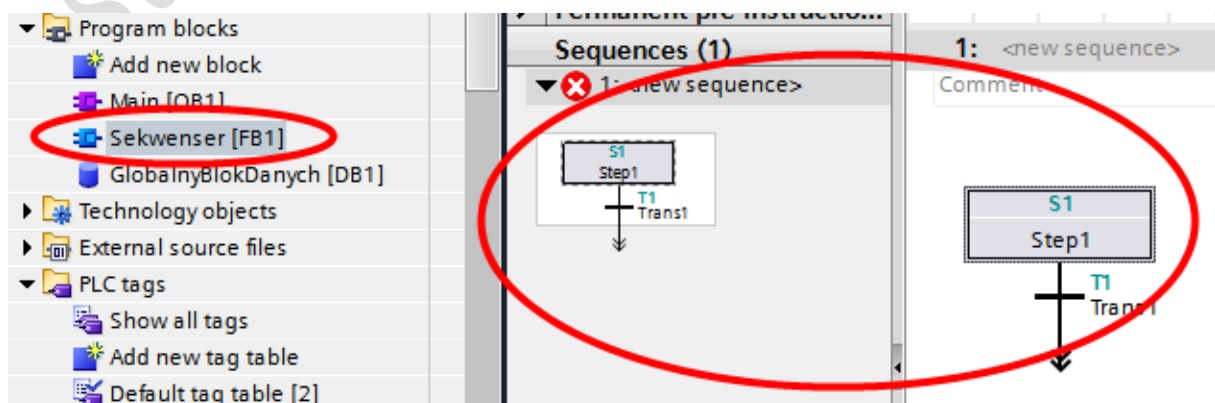
Tworzenie globalnego blok danych



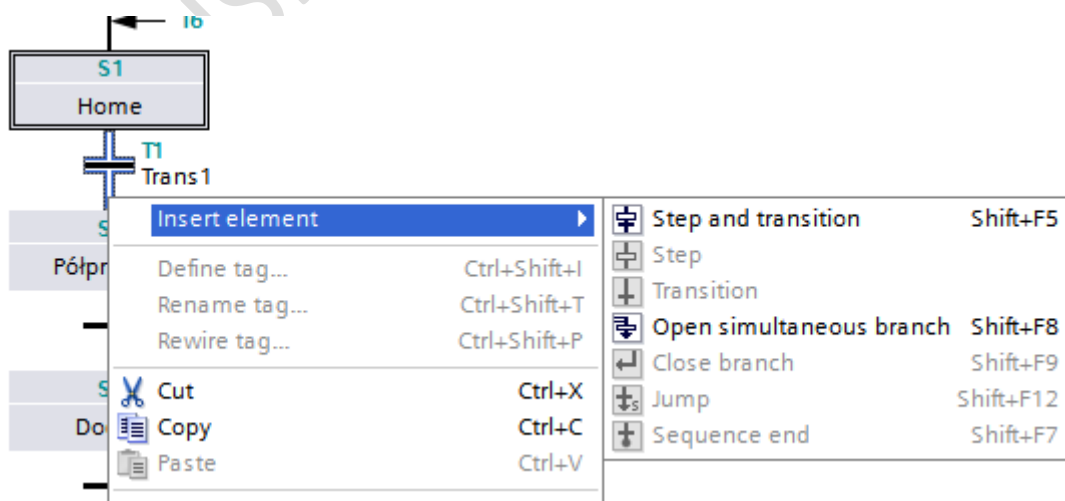
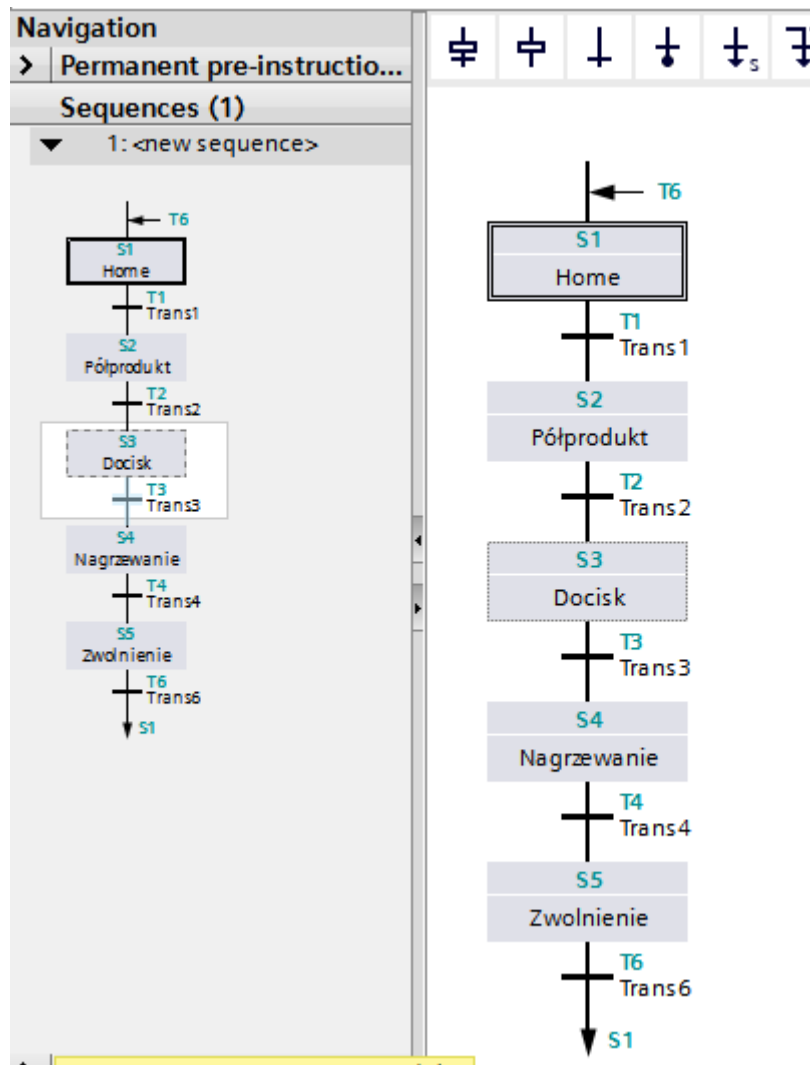
Tworzenie blok funkcji sekwensera



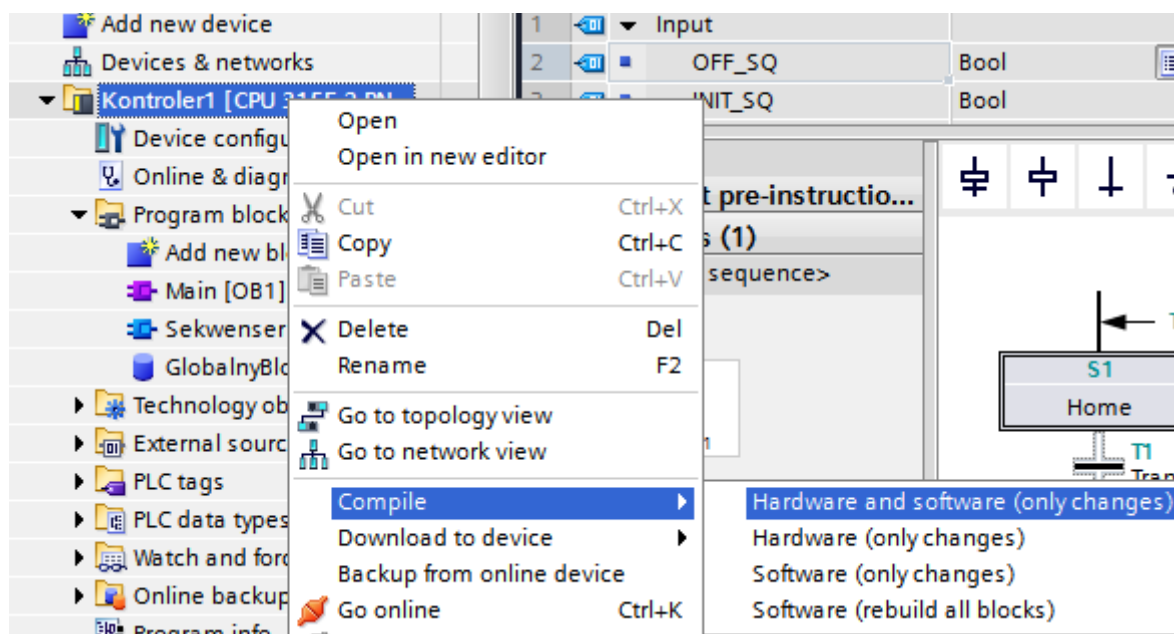
W efekcie otrzymano



Struktura przykładowego sekwensera

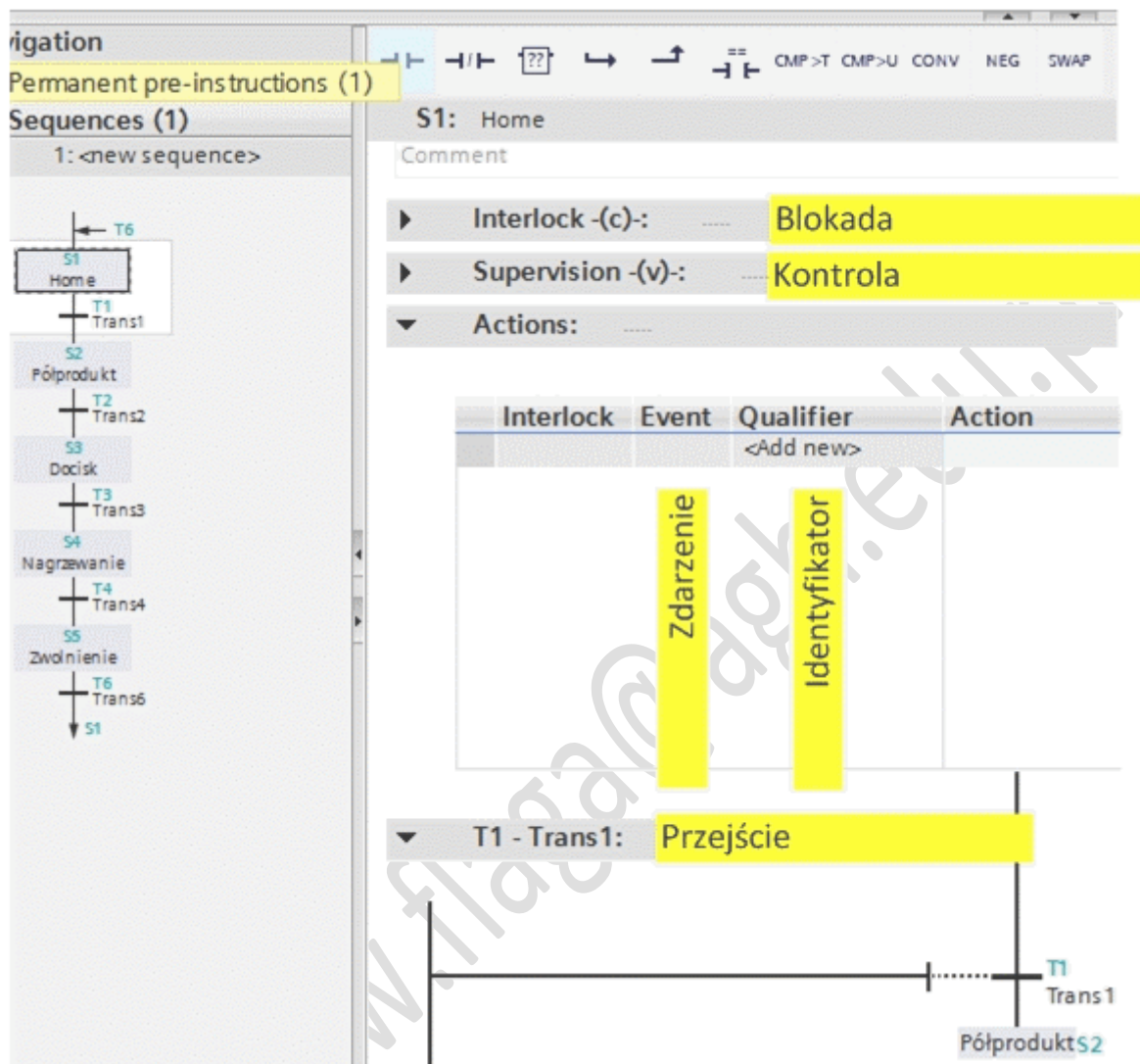


Kompilacja projektu



General		Cross-references	Compile	Syntax		
Compiling completed (errors: 0; warnings: 10)						
!	Path	Description	Go to	?	Errors	Warnings
⚠	▼ Kontroler1		↗		0	10
✓	Hardware configuration		↗		0	0
⚠	▼ Program blocks		↗		0	10
✓	GraphStep (UDT)	The data type was successfully updated.	↗		0	0
✓	GlobalnyBlokDanych (D...	Block was successfully compiled.	↗		0	0
✓	GraphTransition (UDT)	The data type was successfully updated.	↗		0	0
⚠	▼ Sekwenser (FB1)		↗		0	10
⚠	Sequence 1	Step Home does not contain actions.	↗	?	0	1
⚠	Sequence 1	Step Półprodukt does not contain actions.	↗	?	0	1
⚠	Sequence 1	Step Docisk does not contain actions.	↗	?	0	1
⚠	Sequence 1	Step Nagrzewanie does not contain actions.	↗	?	0	1
⚠	Sequence 1	Step Zwolnienie does not contain actions.	↗	?	0	1
⚠	Sequence 1	Transition Trans1 does not contain conditions.	↗	?	0	1
⚠	Sequence 1	Transition Trans2 does not contain conditions.	↗	?	0	1
⚠	Sequence 1	Transition Trans3 does not contain conditions.	↗	?	0	1
⚠	Sequence 1	Transition Trans4 does not contain conditions.	↗	?	0	1
⚠	Sequence 1	Transition Trans6 does not contain conditions.	↗	?	0	1
i		Block was successfully compiled.			0	0
⚠	Compiling completed (errors: 0; warnings: 10)				0	1

Kroki



Blokada (Interlock):

Blokada jest programowym warunkiem wewnątrz kroku. W zakładce Interlock umieszczone są instrukcje, których wykonanie uruchamia blokadę. Może to być wykorzystane w przypadku wystąpienia alarmów w programie użytkownika lub stanów zagrożeń maszyny albo jeśli chcemy zablokować aktywację danej akcji.

- Jeżeli warunki blokady są spełnione i blokada jest aktywowana, sekwenser nie jest zatrzymywany, aktywowana jest też akcja, do której została przypisana blokada.
- Jeżeli warunków blokady nie są spełnione i blokada nie jest aktywowana, sekwenser nie jest zatrzymywany, nie jest wykonywana akcja, do której została przypisana blokada.

Kontrola (Supervision):

Kontrola (monitorowanie pracy kroku) jest programowym warunkiem wewnątrz kroku. Jeśli warunek kontroli nie został spełniony i kontrola nie została aktywowana, oznacza to, że nie wystąpiły stany awaryjne w sekwencji.

- Jeśli warunki kontroli są spełnione i kontrola jest aktywowana, sekwencer jest zatrzymywany w kroku, gdzie wystąpiła aktywacja kontroli, blokowane jest przejście do kolejnego kroku. Możliwe jest skonfigurowanie meldunków informacyjnych o aktualnym stanie pracy sekwencera. Odblokowanie kontroli należy wykonać aktywując wejście ACK_EF bloku FB sekwencera.
- Jeśli warunki kontroli nie są spełnione, sekwencer nie jest zatrzymywany. Wykonywany jest dany krok i kroki następne sekwencera.

Akcje (Actions):

Akcja zawiera bieżące instrukcje procesu sterowania. Wykonanie instrukcji można uzależnić od blokady lub wystąpienia innych zdefiniowanych zdarzeń. ID akcji jest używane do zdefiniowania typu akcji przeznaczonej do wykonania. W akcji można zaprogramować instrukcje takie jak przypisanie wartości, wywołanie bloku lub licznika.

Zdarzenie (Event):

Zdarzenie jest to zmiana stanu sygnału w kroku, kontroli lub w blokadzie lub potwierdzenie komunikatu lub otrzymanie rejestracji. Zdarzenie może być zapisane i przetwarzane w akcji.

Identyfikator (Qualifier):

Identyfikator jest używany do określenia typu akcji, w której krok GRAPH będzie wykonywany. Podczas wybierania niektórych standardowych akcji są automatycznie tworzone symbole zastępcze (np. podczas wywoływania licznika).

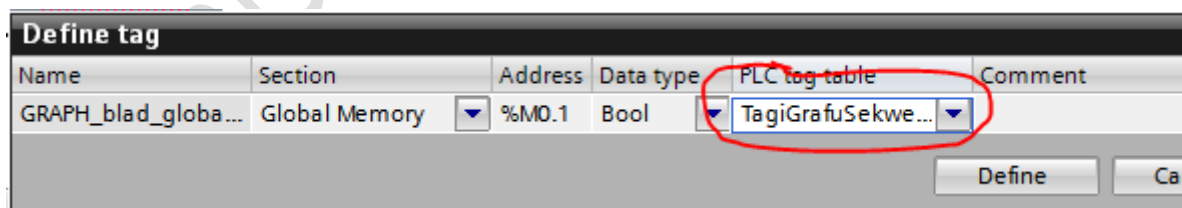
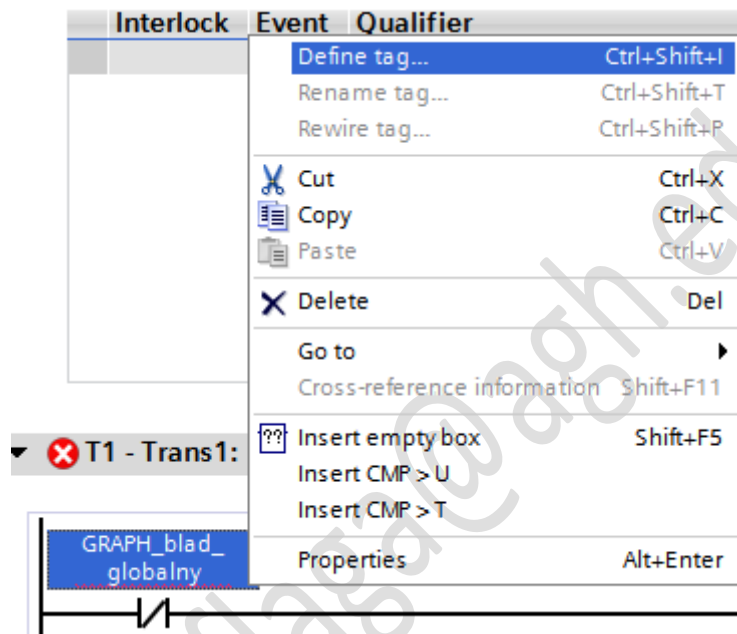
W przykładowym projekcie użyto identyfikatora "N", za pomocą którego można przypisać zmienną do wartości, gdy krok jest aktywny.

Przejście (Transitions):

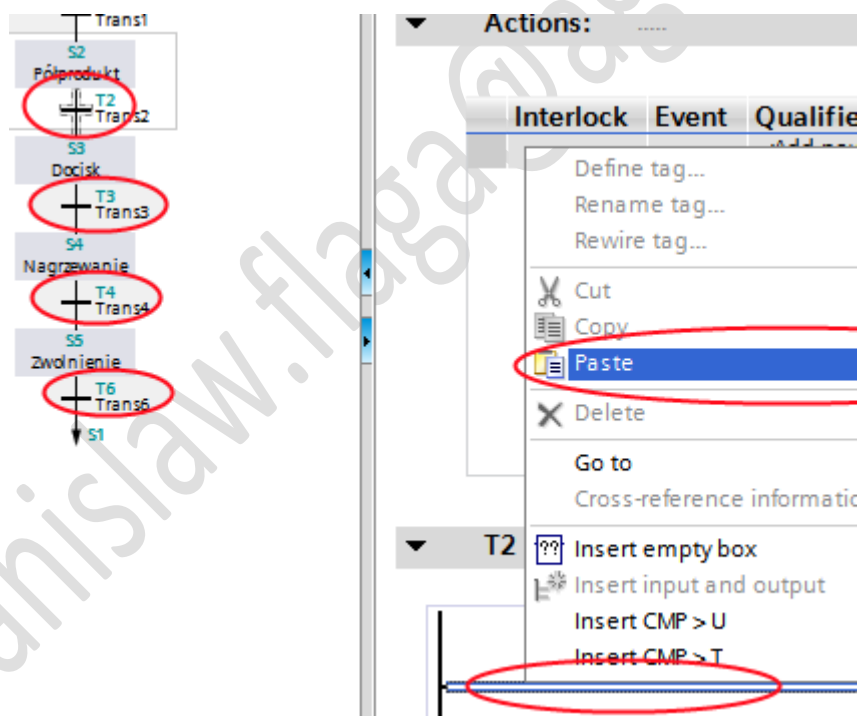
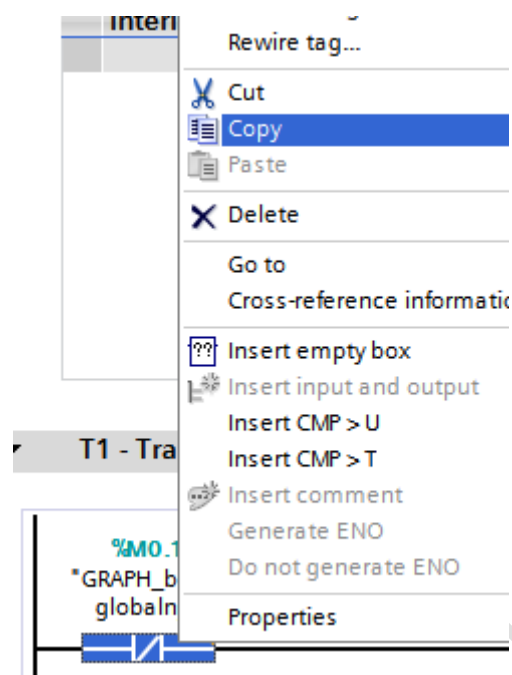
Przejścia zawierają warunki do przełączenia na następny krok. Gdy warunki przejścia są spełnione, proces przełącza się na następny krok.

Wielokrotne warunki przejścia

Używa się go w celu zatrzymania sekwensera w przypadku wystąpienia błędu globalnego.



Skopiować GRAPH_blad_globany do wszystkich tranzycji



S1 – T1

ock	Event	Qualifier	Action
	S1	N - Set as long as step is active	"GRAPH_licznik_dociak" := 0
		<Add new>	Open
			Define tag... Ctrl+Shift+I
			Rename tag... Ctrl+Shift+T
			Remove tag... Ctrl+Shift+R

Define tag					
Name	Section	Address	Data type	PLC tag table	Comment
GRAPH_licznik_doc...	Global Memory	%MW2	Word	TagiGrafuSekwe...	
					Define

Navigation

> Permanent pre-instructio...

Sequences (1)

1: <new sequence>

No condition defined.

S1: Home

Comment

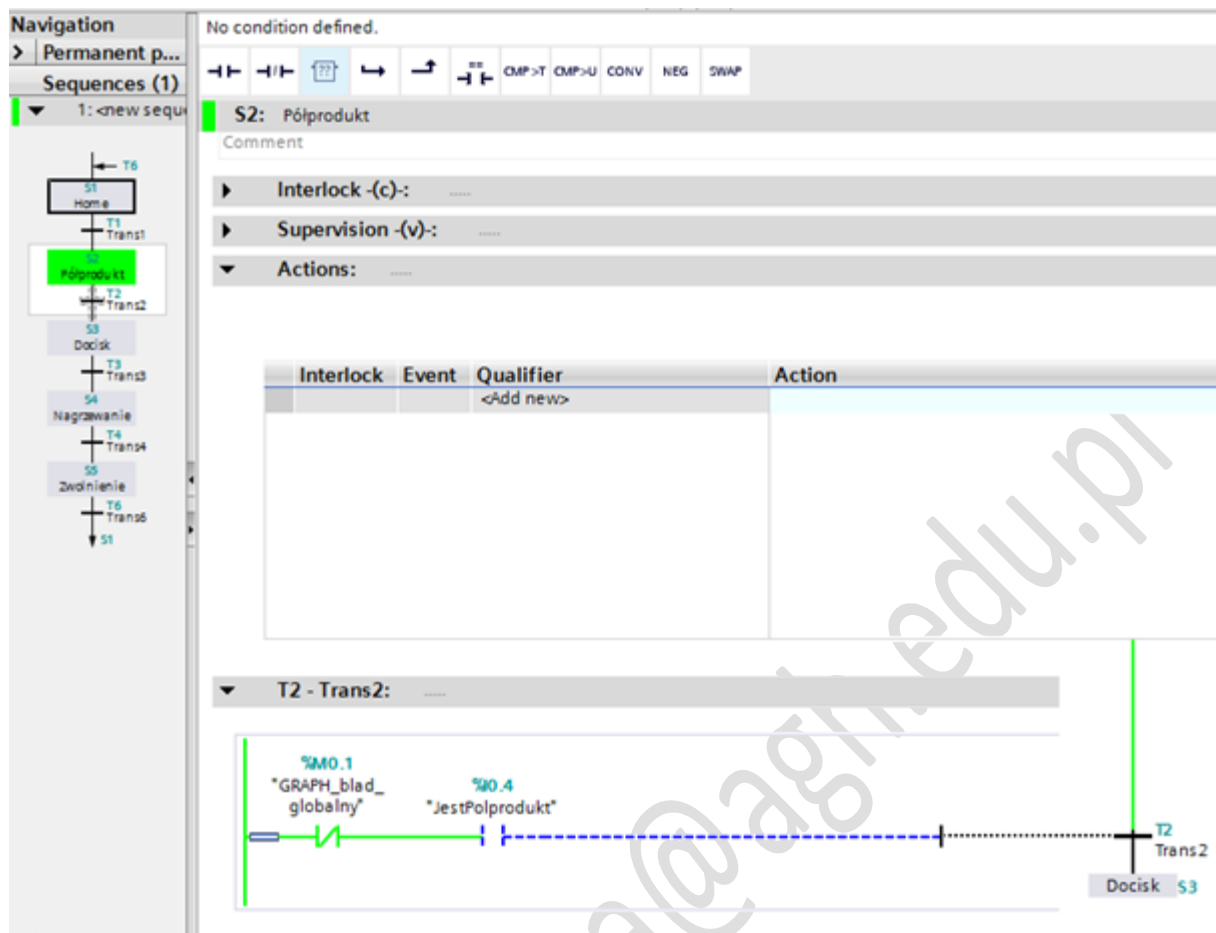
Interlock -(c)-:

Supervision -(v)-:

Actions:

Interlock	Event	Qualifier	Action
	S1	N - Set as long as step is active	"GRAPH_licznik_dociak" := 0
	S1	S - Set to 1	"DoStartowej"
		<Add new>	

T1 - Trans1:

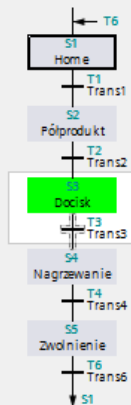


Navigation

Permanent p...

Sequences (1)

1: <new sequ...



No condition defined.

← ← [??] → → [CMP>T] [CMP>U] [CONV] [NEG] [SWAP]

S3: Docisk

Comment

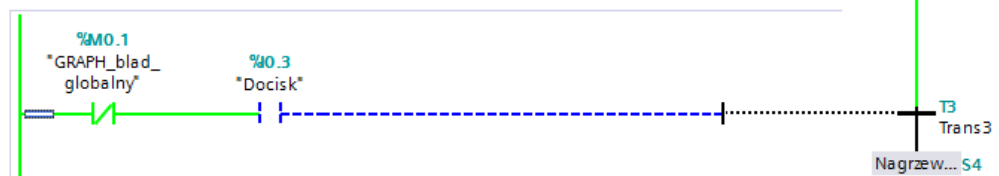
► Interlock -(c)-:

► Supervision -(v)-:

▼ Actions:

Interlock	Event	Qualifier	Action
	S1	R - Set to 0	"DoStartowej"
	S1	S - Set to 1	"DoDocisk"
		<Add new>	

▼ T3 - Trans3:



Navigation

Permanent p...

Sequences (1)

1: <new sequ...

Home

Trans1

Półprodukt

Trans2

Docisk

Trans3

Nagrzewanie

Trans4

Zwolnienie

Trans6

S1

No condition defined.

←

→

??

↩

↪

↔

CMP>T

CMP>U

CONV

NEG

SWAP

S5: Zwolnienie

Comment

Interlock -(c)-:

Supervision -(v)-:

Actions:

Interlock	Event	Qualifier	Action
	S1	R - Set to 0	"DoDocisk"
		S - Set to 1	"DoStartowej"
		<Add new>	

T6 - Trans6:

%M0.1

"GRAPH_blad_

globalny"

↯

.....

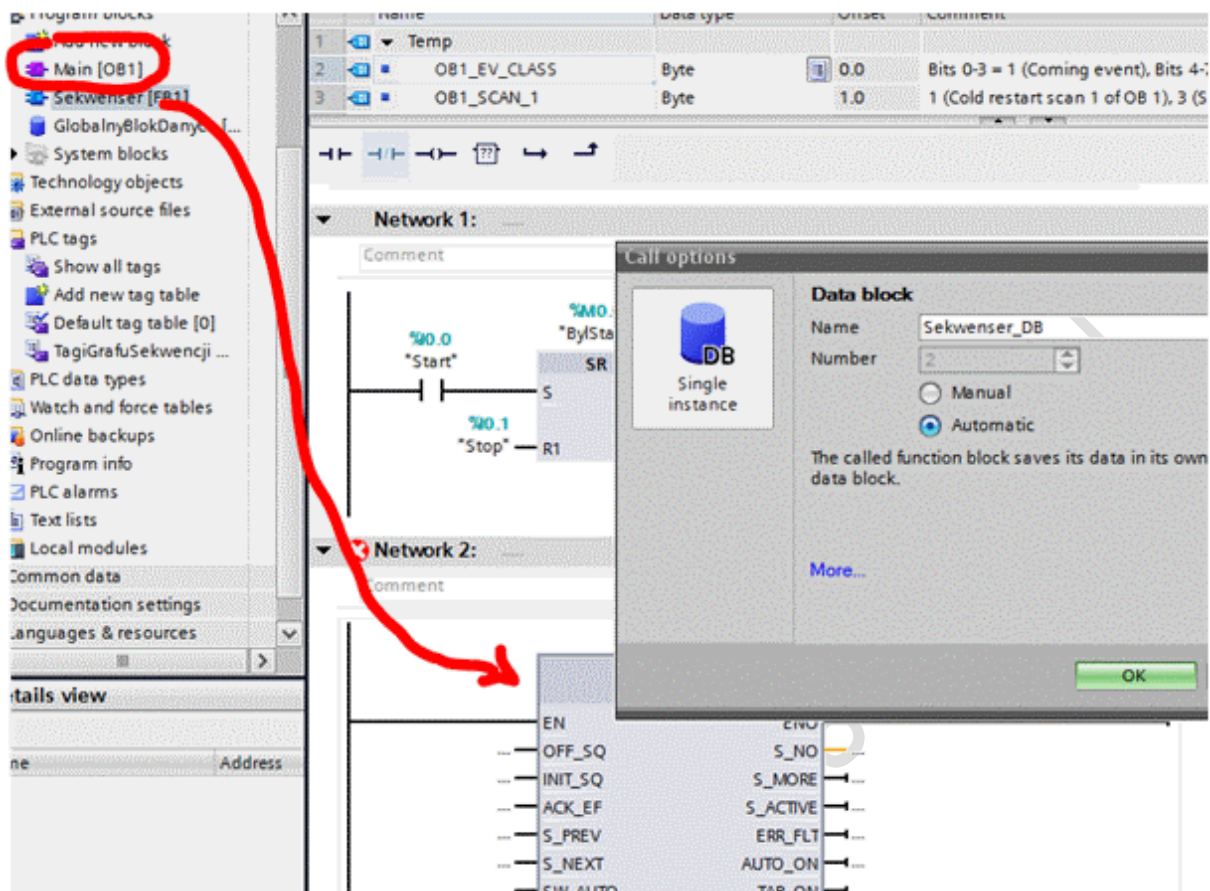
↯

T6

Trans6

stanislaw.flaga

Wywołanie sekwensera



Permanent pre i post instruction

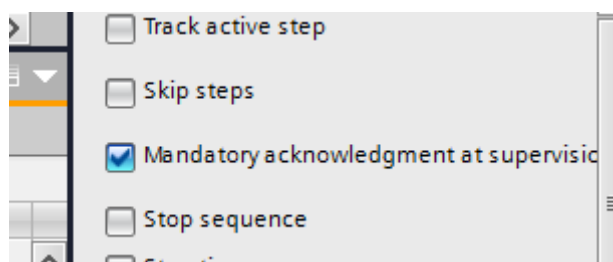
Miejsce na umieszczenie instrukcji wykonywanych przed i po sekwencji.

UWAGA 1: jeżeli FB sekwensera jest wywołane to pre/post instrukcje wykonywane są w każdym cyklu niezależnie od stanu sekwensera

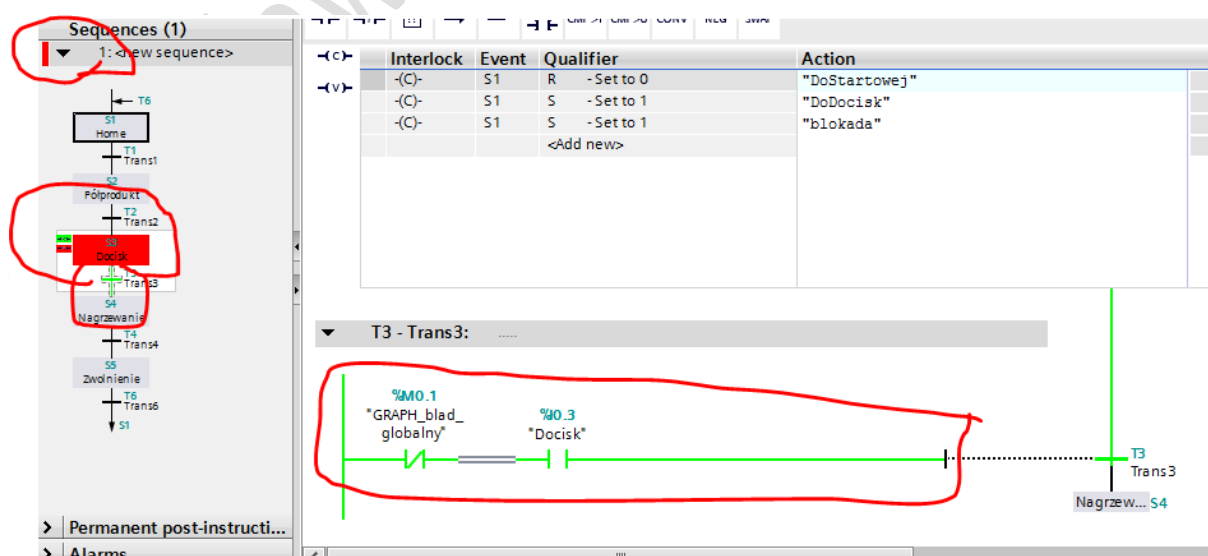
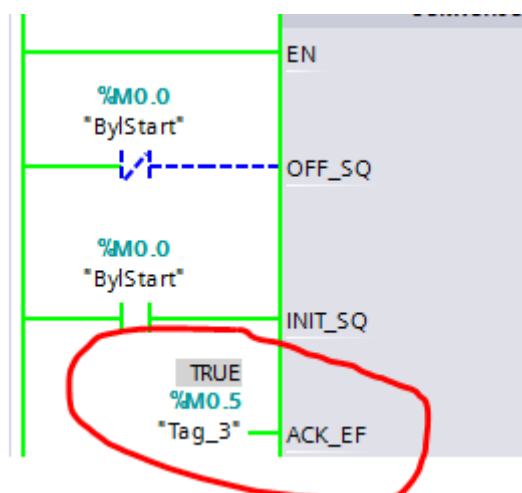
Supervision – korzystanie z kontroli kroku

UWAGA 1: zablokowanie przejścia do kolejnego kroku jeżeli $-(v) = 1$

Jeżeli ustawiono kwitowanie blokowania przejścia



Należy po spełnieniu warunku przejścia ustawić 1 na ACK_EF w celu skwitowania



Interlock – korzystanie z blokady

UWAGA 1: nie są wykonywane zablokowane akcje jeżeli $\neg(c) = 0$

