

## 4. MATERIAŁ NAUCZANIA

### 4.1. Układy regulacji i sterowania

#### 4.1.1. Materiał nauczania

Układy regulacji i sterowania realizują zadania związane z uruchomieniem, oddziaływaniem, kontrolą i zakończeniem pracy urządzeń automatyki.

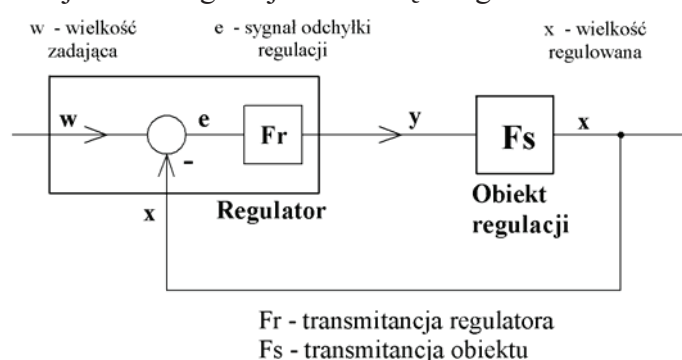
#### Regulacja. Układy regulacji.

Regulacja polega na celowym oddziaływaniu na wielkość regulowaną mierzona na bieżąco, aby była podobna do wielkości zadanej. W procesie regulacji występują trzy podstawowe sygnały: wielkość zadana, odchyłka regulacji i wielkość regulowana.

Rozróżniamy regulację ręczną i automatyczną. Regulacja automatyczna może być:

- stałowartościowa – utrzymuje daną wartości wielkości regulowanej,
- nadążna – zapewnia nadążanie wielkości regulowanej za zmianami wartości zadanej.

Proces regulacji odbywa się w układzie zamkniętym, z ujemnym sprzężeniem zwrotnym od wielkości regulowanej. Układ regulacji składa się z regulatora i obiektu regulacji.



Rys.1. Schemat blokowy układu regulacji [9, s. 208]

Ze względu na sposób przetwarzania sygnałów rozróżniamy układy regulacji:

- ciągłej, przy przetwarzaniu analogowym sygnałów,
- dwustanowej (binarnej), przy przetwarzaniu binarnym sygnałów,
- cyfrowej, przy przetwarzaniu cyfrowym sygnałów.

#### Regulacja ciągła.

Wielkość sterująca obiektem w układach regulacji ciągłej może przyjmować dowolne wartości z dopuszczalnego obszaru zmienności.

Regulatory ciągłe najczęściej budowane są ze elektronicznych wzmacniaczy operacyjnych.

#### Regulacja nieliniowa.

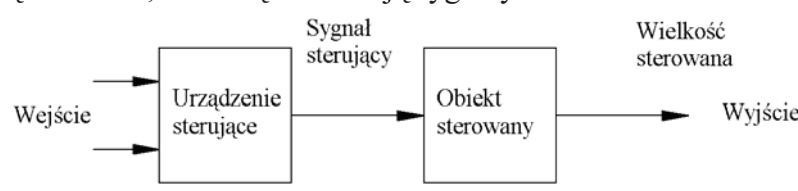
W układach regulacji nieliniowej wartość wielkości podawanej do obiektu może przyjmować tylko dwie lub kilka wybranych wartości. Najczęściej spotykane metody regulacji nieliniowej to regulacja dwustanowa i trójstanowa.

Regulatory dwustanowe charakteryzują się dwoma jednoznacznie określonymi stanami sygnału wejściowego. Typowym przykładem ich zastosowania są układy regulacji temperatury, ciśnienia czy poziomu napełnienia zbiorników. Parametrem charakterystycznym regulatorów dwustanowych jest histereza przełączania określająca różnicę wartości stanów sygnału wyjściowego.

Regulatory trójstanowe charakteryzują się trzema jednoznacznie określonymi stanami sygnału wejściowego.

### Sterowanie. Układy sterowania.

Sterowanie to proces celowego oddziaływania na sygnałów sterujących na przyrządy, urządzenia technologiczne lub maszyny robocze. Sterowanie odbywa się w układzie otwartym. Sygnały sterujące działają na obiekt bez bieżących pomiarów i korekcji procesu. Układ sterowania składa się z obiektu sterowania i urządzenia sterującego. Obiekt sterowania to część układu, na którą oddziałują sygnały sterownicze.



Rys. 2. Schemat blokowy układu sterowania

Urządzenie sterujące to część układu, która przez człon wykonawczy oddziałuje na obiekt sterowania. Urządzenie sterujące składa się z:

- urządzenia wejścia, są to różnego rodzaju czujniki, łączniki przyciski elektryczne,
- członu realizującego funkcje logiczne przetwarzającego sygnały wejściowe według zadanego programu, wypracowując sygnały sterujące,
- urządzenia wyjścia są to człony wykonawcze oddziałujące bezpośrednio na obiekt sterowania np. przekaźniki, styczniki, tyrystory, zawory hydrauliczne i pneumatyczne.

Ze względu na postać sygnału rozróżnia się sterowanie:

- analogowe, gdy stosowane są sygnały ciągłe,
- binarne, gdy stosowane są sygnały dwuwartościowe,
- cyfrowe, gdy stosowane są sygnały w postaci cyfrowej.

Ze względu na sposób przetwarzania i wykorzystania sygnałów rozróżnia się sterowanie:

- kombinacyjne – sygnał sterujący powstaje z powiązania kilku sygnałów,
- sekwencyjne – kolejne czynności sterujące wykonywane są krok po kroku.

Ze względu na sposób realizacji i zapisu programu sygnału rozróżnia się sterowanie:

- stałoprogramowe w układach stykowo-przekaźnikowych,
- programowalne w układach ze sterownikami mikroprocesorowymi.

Ze względu na rodzaj sygnału rozróżnia się sterowanie: mechaniczne, elektryczne, pneumatyczne, hydrauliczne.

### 4.1.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń

1. Na czym polega proces regulacji?
2. Jakie znasz rodzaje regulacji?
3. W jakim układzie może odbywać się proces regulacji?
4. Z jakich elementów składa się układ regulacji?
5. Jakie sygnały mogą występować w układach regulacji?
6. Czym charakteryzuje się regulacja ciągła?
7. Czym charakteryzuje się regulacja nieliniowa?
8. Na czym polega proces sterowania?
9. Jakie znasz rodzaje sterowania?
10. Z jakich elementów składa się układ sterowania?
11. Z jakich bloków składa się urządzenie sterujące?

### 4.1.3. Ćwiczenia

#### Ćwiczenie 1

Wybierz, korzystając z katalogu, czujniki, które można zastosować w układzie sterowania bramą garażową. Sterowanie odbywa się za pośrednictwem pilota: brama może otworzyć się do połowy lub na całą wysokość.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) dobrać rodzaje czujników,
- 3) zaproponować ilość i miejsce instalacji czujników,
- 4) uzasadnić przyjęte rozwiązanie
- 5) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- katalogi elementów automatyki,
- zeszyt
- literatura uzupełniająca zgodna z punktem 6.

#### Ćwiczenie 2

Podaj dwa przykłady układów regulacji stosowanych w znanych Ci urządzeniach elektrycznych i elektronicznych. Określ, jaki rodzaj regulacji automatycznej został wykorzystany.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować treść zadania,
- 2) zapoznać się z przygotowaną dokumentacją techniczną wybranych urządzeń elektronicznych,
- 3) narysować blokowe schematy poglądowe układów regulacji zastosowanych w analizowanych urządzeniach,
- 4) wyodrębnić elementy układów regulacji,
- 5) określić rodzaj regulacji,
- 6) uzasadnić sformułowane wnioski,
- 7) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna wybranych urządzeń elektronicznych,
- zeszyt,
- literatura uzupełniająca zgodna z punktem 6.

#### 4.1.4 Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) narysować schemat blokowy układu regulacji?	<input type="checkbox"/>	<input type="checkbox"/>
2) podać przykłady zastosowania regulacji automatycznej?	<input type="checkbox"/>	<input type="checkbox"/>
3) określić rodzaj regulacji wykorzystywanej w urządzeniu?	<input type="checkbox"/>	<input type="checkbox"/>
4) opisać działanie regulatora ciągłego?	<input type="checkbox"/>	<input type="checkbox"/>
5) opisać działanie regulatora dwustanowego?	<input type="checkbox"/>	<input type="checkbox"/>
6) opisać działanie regulatora trójstanowego?	<input type="checkbox"/>	<input type="checkbox"/>
7) określić, który element układu regulacji jest regulatorem, a który obiektem?	<input type="checkbox"/>	<input type="checkbox"/>
8) narysować schemat blokowy układu sterowania?	<input type="checkbox"/>	<input type="checkbox"/>
9) określić rodzaj sterowania wykorzystywany w urządzeniu?	<input type="checkbox"/>	<input type="checkbox"/>
10) określić człony składowe urządzenia sterującego?	<input type="checkbox"/>	<input type="checkbox"/>

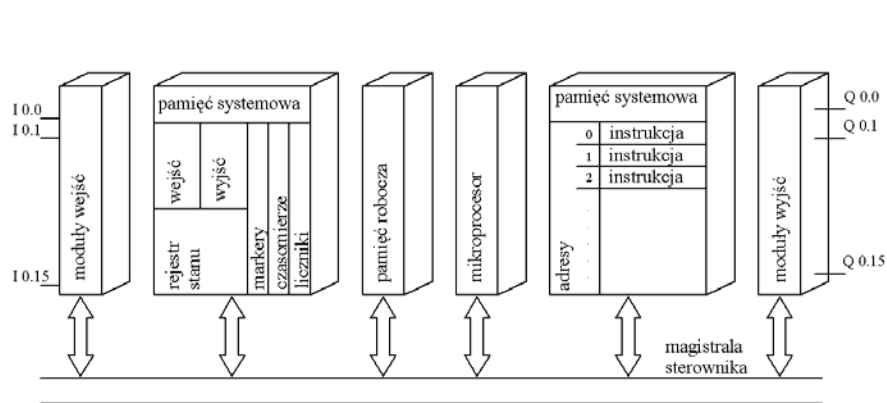
## 4.2. Sterowniki programowalne PLC

### 4.2.1. Materiał nauczania

#### Budowa sterownika PLC

Sterowniki programowalne PLC mogą być wykonane w postaci kompaktowej lub modułowej, która pozwala na modernizację z rozszerzenia możliwości urządzenia.

Sterownik PLC zbudowany jest z: układu zasilania napięciowego (zasilacza), modułu sygnałów wejściowych, jednostki centralnej z mikroprocesorem (CPU), bloku pamięci, modułu sygnałów wyjściowych. Układ zasilania napięciowego to zazwyczaj zasilacz napięcia stałego +24V.



Rys. 3. Schemat blokowy sterownika PLC [9, s. 168]

Blok pamięci zawiera następujące obszary połączone wewnętrzną magistralą sterownika:

- pamięć danych typu RAM lub EPROM, przechowuje dane i instrukcje programu użytkownika, może być rozszerzana za pomocą dodatkowych kart lub modułów,
- pamięć robocza – szybka pamięć typu RAM, w trakcie przetwarzania programu przez użytkownika kopiowane są do niej dane,
- pamięć systemowa zawiera zmienne, na których wykonywane są operacje programu.

Moduł sygnałów wejściowych zawiera układy elektroniczne zamieniające sygnały pochodzące z urządzeń zewnętrznych na sygnały logiczne akceptowane przez sterownik. Układy stosowane w tym module to: dzielniki napięcia, filtry RC tłumiące zakłócenia, diody chroniące, układy prostownicze, transoptory izolujące obwody wejściowe i magistralę sterownika. Sterowniki zawierają zazwyczaj moduły 8, 16, lub 32 wejść binarnych, które są multipleksowane. Ich stan sygnalizowany jest diodą LED.

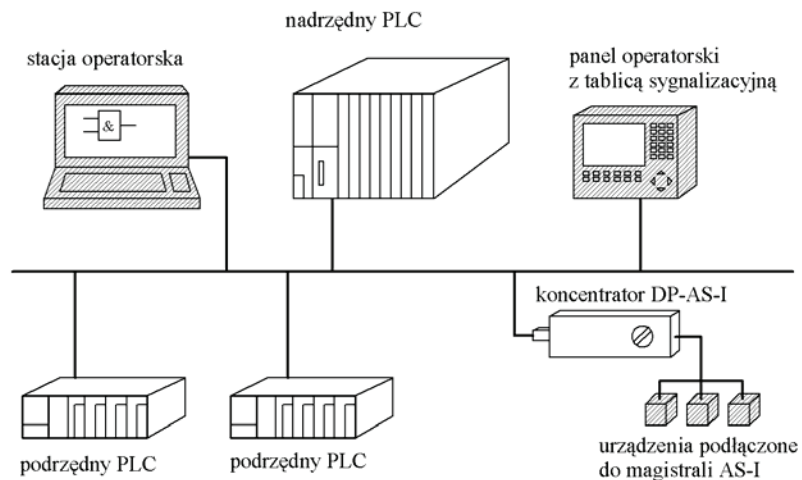
Moduł sygnałów wyjściowych zawierają zazwyczaj moduły 8, 16, lub 32 wyjść binarnych trzech rodzajów:

- wyjścia przekaźnikowe zapewniające całkowitą separację galwaniczną wewnętrznych układów sterownika i obwodów wejściowych,
- wyjścia z triakami stosowane dla zasilania odbiorników prądu przemiennego,
- wyjścia tranzystorowe stosowane dla zasilania odbiorników napięcia stałego.

#### Współpraca sterowników w sieciach przemysłowych.

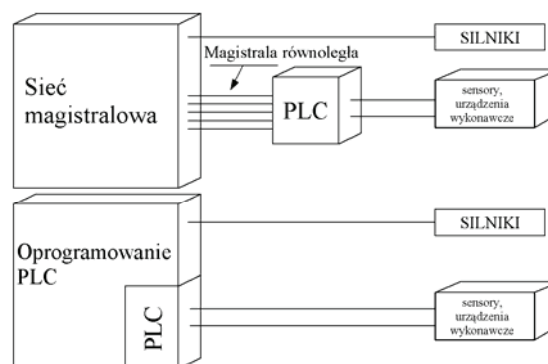
Dla obsługi skomplikowanych procesów przemysłowych zazwyczaj nie wystarczy jeden sterownik. Poszczególne jednostki produkcyjne są wówczas sterowane poprzez sterowniki lokalne, podrzędne (Slave), połączone przy pomocy sieci np. PROFIBUS-DP ze sterownikiem nadrzędnym (Master), który zarządza całym systemem. Do magistrali może

być podłączonych wiele innych urządzeń jak np. stacje i panele operatorskie, inne sieci np. pozwalające na współpracę z urządzeniami pomiarowymi i wykonawczymi.



Rys. 4. Schemat blokowy sieci sterowników. [9, s. 166]

Sterownik PLC może też być zintegrowany z innym układem sterowania cyfrowego. Często integruje się tylko wybrane moduły sterownika z systemem operacyjnym układu sterowania cyfrowego.



Rys. 5. Schemat blokowy układu zintegrowanego sterownika. [9, s. 167]

### Rodzaje sygnałów

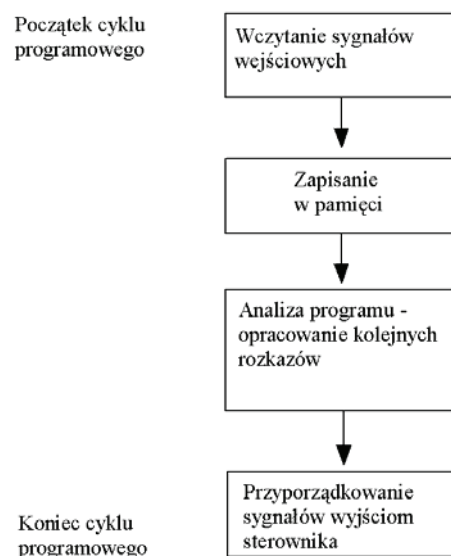
Sterowniki są urządzeniami cyfrowymi, zatem przetwarzają sygnały cyfrowe. Najczęściej sygnałowi cyfrowemu 1 odpowiada stałe napięcie +24V (dopuszczalny jest zakres zmian od 16V do 36V), natomiast sygnałowi 0 odpowiada napięcie 0V.

W układach sterowania występują różnego rodzaju czujniki wytwarzające sygnały analogowe, które niosą określone informacje. Również sygnały wyjściowe sterowników muszą w określonych sytuacjach sterować elementami i urządzeniami analogowymi. Sterowniki wyposażone są w wejścia i wyjścia analogowe o zakresie zmian napięcia zazwyczaj od 0V do 10V. Wejściowe sygnały analogowe są w sterowniku przed obróbką przez program przetwarzane na sygnały cyfrowe. Podobnie cyfrowe sygnały będące wynikiem przetwarzania programowego mogą być zamienione w końcowej fazie cyklu na wyjściowe sygnały analogowe.

Dokładne informacje na temat typów i poziomów sygnałów wejściowych i wyjściowych sterownika znajdują się w jego dokumentacji technicznej.

## Zasada działania sterownika PLC

Sterownik PLC pracuje sekwencyjnie. Program sporządzony za pomocą programatora (zazwyczaj to komputer PC) jest zapamiętany w pamięci sterownika. Na początku każdego programu sprawdzane są stany operandów (np. wejść, wyjść) i zapamiętywane w rejestrze pośrednim. Następnie wykonywane są rozkazy zawarte w programie kolejno jeden po drugim (operandy wiązane są ze sobą zgodnie z zapisanymi w programie funkcjami). Wyniki operacji zapisywane są w akumulatorze i mogą być dalej przetwarzane. Po opracowaniu wszystkich rozkazów programu końcowe wyniki operacji przesyłane są do rejestru wyjściowego i dalej przyporządkowane do odpowiednich wyjść sterownika, które oddziałują poprzez człony wykonawcze na obiekt sterowania. Sterownik pracuje cyklicznie co oznacza, że proces opracowania programu jest ciągle powtarzany. Czas przetwarzania jednego programu nazywany jest czasem cyklu i zależy od szybkości działania CPU, długości programu i operacji, jakie są w nim zawarte.



Rys. 6. Schemat blokowy przetwarzania programu użytkownika

## Oprogramowanie narzędziowe

Programowanie sterownika odbywa się poprzez programator, jakim zazwyczaj jest komputer PC. Zainstalowane na nim jest oprogramowanie narzędziowe odpowiednie dla danego typu sterownika. Posiada ono szereg funkcji umożliwiających tworzenie i edycję programów dla sterownika oraz dokumentacji, komunikację ze sterownikiem, diagnostykę trybu jego pracy, wymuszanie stanów wyjściowych. Niektóre wersje oprogramowania narzędziowego umożliwiają symulacje działania sterownika, co pozwala na testowanie tworzonych programów.

Komunikacja programatora z komputerem odbywa się poprzez interfejs szeregowy RS 232C.

### 4.2.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Z jakich modułów składa się sterownik PLC?
2. Jakiego rodzaju układy elektroniczne składają się na moduł wejściowy?
3. Jakie są stosowane rodzaje wyjść sterownika PLC?
4. Z jakich bloków składa się pamięć sterownika?



5. Co to znaczy, że sterownik pracuje sekwencyjnie?
6. Co to znaczy, że sterownik pracuje cyklicznie?
7. Do czego służy oprogramowanie narzędziowe?
8. W jaki sposób odbywa się komunikacja programatora i sterownika?

### 4.2.3. Ćwiczenia

#### Ćwiczenie 1

Określ rodzaj wyjść sterownika, oraz parametry sygnałów wyjściowych na podstawie dokumentacji technicznej. Jakimi urządzeniami może on sterować?

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować treść zadania,
- 2) zapoznać się z przygotowaną dokumentacją techniczną sterownika,
- 3) wypisać dane techniczne,
- 4) wyjaśnić do sterowania jakimi urządzeniami może służyć ten typ sterownika.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika,
- zeszyt,
- literatura uzupełniająca zgodna z punktem 6.

#### Ćwiczenie 2

Zbadaj tryb pracy sterownika na podstawie dokumentacji technicznej i oprogramowania narzędziowego. Wykorzystaj funkcje obsługujące (dodatkowe) dostępne w menu podstawowym.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować treść zadania,
- 2) zapoznać się z przygotowaną dokumentacją techniczną sterownika,
- 3) uruchomić oprogramowanie narzędziowe sterownika,
- 4) uruchomić funkcje obsługujące z poziomu menu podstawowego,
- 5) zdiagnozować tryb pracy sterownika.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika,
- programator sterownika (komputer osobisty PC) z zainstalowanym oprogramowaniem narzędziowym,
- sterownik PLC,
- literatura uzupełniająca zgodna z punktem 6.



#### 4.2.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) opisać budowę sterownika PLC?	<input type="checkbox"/>	<input type="checkbox"/>
2) określić wielkość i rodzaj pamięci sterownika, na podstawie dokumentacji technicznej?	<input type="checkbox"/>	<input type="checkbox"/>
3) określić ilość i rodzaj wejść sterownika, na podstawie dokumentacji technicznej?	<input type="checkbox"/>	<input type="checkbox"/>
4) określić ilość i rodzaj wyjść sterownika, na podstawie dokumentacji technicznej?	<input type="checkbox"/>	<input type="checkbox"/>
5) określić poziomy sygnałów wejściowych sterownika, na podstawie dokumentacji technicznej?	<input type="checkbox"/>	<input type="checkbox"/>
6) określić poziomy sygnałów wyjściowych sterownika, na podstawie dokumentacji technicznej?	<input type="checkbox"/>	<input type="checkbox"/>
7) zdiagnozować trybu pracy sterownika, posługując oprogramowaniem narzędziowym?	<input type="checkbox"/>	<input type="checkbox"/>
8) zdiagnozować parametry sterownika, posługując się oprogramowaniem narzędziowym?	<input type="checkbox"/>	<input type="checkbox"/>
9) opisać budowę sieci sterownikowej?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.3. Podstawy programowania sterowników PLC

### 4.3.1. Materiał nauczania

#### Projektowanie systemów automatyki

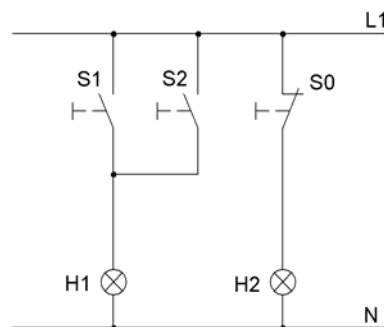
Podstawą do wykonania projektu związanego z automatyzacją jest dokładny opis procesu który powinien zawierać warunki konieczne do realizacji zadania. Wraz z opisem zadania podawany jest często schemat technologiczny. Zawiera on budowę i zasadę działania urządzenia, informacje dotyczące procesu technologicznego oraz liczby i miejsca zainstalowania urządzeń zabezpieczających.

Realizację projektu można podzielić na 2 etapy: tworzenie struktury sprzętowej układu sterowania i wybór struktury programu użytkownika. Pierwszy etap rozpoczyna się od ustalenia powiązań sieciowych. W dalszej kolejności następuje wybór sprzętu, jego kompletacja i na koniec konfiguracja.

Wybór struktury programu użytkownika, to określenie, z jakich bloków programowych będzie się on składał.

#### Schemat rozwinięty układu sterowania

To schemat ideowy przedstawiający połączenie elementów wejściowych i wyjściowych stycznikowego układu sterowania. Stanowi często podstawę do realizacji projektu sterowania z wykorzystaniem sterownika PLC.



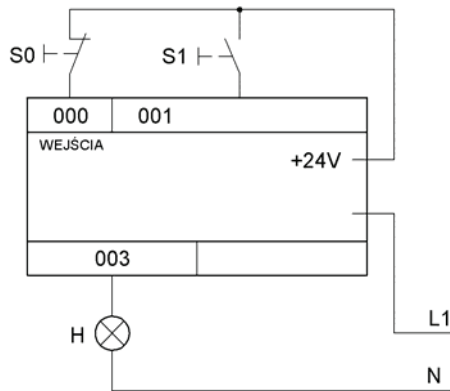
Rys. 7. Przykładowy schemat rozwinięty sterowania stycznikowego

#### Schemat połączenia zacisków sterownika

Schemat połączenia zacisków sterownika przedstawia sposób podłączenia urządzeń zewnętrznych ze sterownikiem. Jest on niezbędny do ustalenia adresów zmiennych wejściowych i wyjściowych używanych w programie. Symbole podłączanych urządzeń są znormalizowane, takie same, jak w schematach sterowania stycznikowego.

Układy sterowania ze sterownikiem PLC muszą posiadać:

- obwody zasilania sterownika,
- obwody zasilania napięciem stałym wyjść sterownika,
- obwody zasilania urządzeń podłączonych do wyjść sterownika.



Rys. 8. Przykładowy schemat połączeń zacisków sterownika.

## Lista przyporządkowująca

Lista przyporządkowująca zawiera symbole wszystkich zmiennych zwanych operandami (sygnałów wejściowych, wyjściowych, merkerów) używanych w programie sterowania. Mogą one być opatrzone komentarzami np. nazwą urządzenia, jakie podłączono do wejścia sterownika. Umieszczone w liście przyporządkowującej symbole i komentarze pojawiają się automatycznie po wywołaniu w programie zadeklarowanej zmiennej.

Lista przyporządkowująca jest bardzo przydatna przy tworzeniu dokumentacji technicznej układu sterowania, ułatwia również analizę programu w przypadku modyfikacji lub awarii. Umożliwia ona określenie: urządzeń przyłączonych do wejść i wyjść sterownika oraz funkcji wewnętrznych sterownika użytych w programie i ich funkcji w procesie sterowania.

wiersz	operand	symbol	komentarz do operandu
0000	E 0.00	start S1	przycisk załączający zwierny
0001	E 0.01	bezp S0	wyłącznik bezpieczeństwa rozwierny
0002	M 0.00	czas T0	wyjście modułu czasowego opóźnienia
0003	A 0.00	Lampka H1	wyjście sygnalizacyjne

Rys. 9. Przykład listy przyporządkowującej

## Budowa programu użytkownika

Tworzenie programu rozpoczyna się od wybrania struktury, program jest podzielony na oddzielne bloki programowe. Program może zawierać bloki organizacyjne, programowe, funkcyjne, sekwencyjne i bloki danych.

Bloki organizacyjne (OB) są połączeniem pomiędzy systemem operacyjnym a programem użytkownika, mogą to być blok wywoływane:

- cyklicznie przez system operacyjny (np. blok z programem głównym),
- przez system operacyjny po wystąpieniu określonych zdarzeń (np. przerwania),
- w programie użytkownika.

Bloki programowe wykorzystywane są do strukturyzacji programu użytkownika.

Bloki funkcjonalne są wywoływane z argumentami, pozwalają na wykorzystanie całego zbioru dostępnych instrukcji procesora. Dzięki temu można programować wielokrotnie powtarzające się sekwencje z różnymi wartościami parametrów.

Bloki sekwencyjne wykorzystuje się w programowaniu zadań sekwencyjnych czyli takich które występują w ustalonej kolejności jeden po drugim. Blok sekwencyjny składa się z zadań i warunków przejścia.

Bloki danych przechowują stałe i zmienne dane wykorzystywane w programie. Często korzysta się z adresowania symbolicznego zmiennych. Po zadeklarowaniu zmiennej jako zmiennej globalnej jej symboliczna nazwa jest znana we wszystkich blokach programu. Zmienna zadeklarowana w danym bloku jest zmienną lokalną znaną tylko w tym bloku.

W tworzeniu nowego programu możliwe jest wykorzystanie istniejących programów źródłowych i typowych bloków programowych napisanych w ramach innych projektów.

### 4.3.2. Pytania sprawdzające

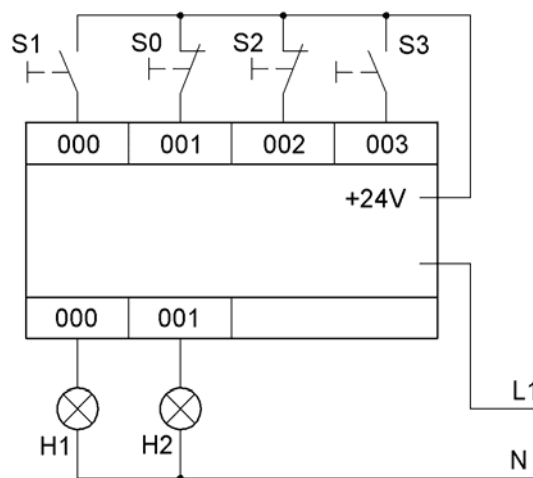
Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Co przedstawia schemat rozwinięty sterowania stycznikowego?
2. Co przedstawia schemat połączenia zacisków sterownika?
3. Jakie obwody zasilania zawiera układ ze sterownikiem PLC?
4. Jakie informacje zawiera lista przyporządkowująca?
5. Jakie bloki może zawierać program użytkownika?
6. Do jakich celów wykorzystuje się bloki programowe?
7. Jakie bloki umożliwiają programowanie wielokrotnie powtarzających sekwencji?
8. W jaki sposób programuje się zadania sterowania sekwencyjnego?

### 4.3.3. Ćwiczenia

#### Ćwiczenie 1

Wykonaj połączenia pomiędzy sterownikiem i urządzeniami zewnętrznymi na podstawie schematu połączeń zacisków sterownika.



Schemat połączeń zacisków sterownika.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) zanalizować treść zadania,
- 2) zapoznać się z przygotowaną dokumentacją techniczną sterownika,
- 3) wykonać wskazane połączenia,
- 4) zaprezentować wyniki swojej pracy.

- Wyposażenie stanowiska pracy:
- dokumentacja techniczna sterownika,
  - sterownik PLC,
  - schemat połączeń zacisków sterownika,
  - 2 przyciski zwierne, 2 przyciski rozwierny,
  - 2 lampki sygnalizacyjne,
  - katalogi elementów automatyki.

## Ćwiczenie 2

W celu załączenia silnika należy wcisnąć jednocześnie dwa przyciski zwierne S1 i S2 . Układ wyposażony jest w wyłącznik bezpieczeństwa S0 rozwierny. Pracę silnika sygnalizuje lampka sygnalizacyjna H1. Do sterowania układu należy wykorzystać sterownik PLC. Narysuj schemat połączeń zacisków sterownika dla opisanego układu i sporządź listę przyporządkowującą.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować treść zadania,
- 2) zapoznać się z przygotowaną dokumentacją techniczną sterownika,
- 3) narysować schemat połączeń zacisków sterownika dla opisanego układu,
- 4) uruchomić oprogramowanie użytkowe sterownika,
- 5) sporządzić listę przyporządkowującą,
- 6) zaprezentować wyniki swojej.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika,
- sterownik PLC,
- programator sterownika wraz z oprogramowaniem narzędziowym,
- katalogi elementów automatyki,
- zeszyt,
- literatura uzupełniająca zgodna z punktem 6.

### 4.3.4. Sprawdzenie postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) opisać działanie układu na podstawie schematu rozwiniętego sterowania?	<input type="checkbox"/>	<input type="checkbox"/>
2) wykonać schemat połączenia zacisków sterownika na podstawie schematu technologicznego?	<input type="checkbox"/>	<input type="checkbox"/>
3) wykonać połączenia między sterownikiem a zewnętrznymi elementami na podstawie schematu połączenia zacisków sterownika?	<input type="checkbox"/>	<input type="checkbox"/>
4) wykonać schemat połączenia zacisków sterownika na podstawie opisu działania układu?	<input type="checkbox"/>	<input type="checkbox"/>
5) sporządzić listę przyporządkowującą dla danego układu sterowania?	<input type="checkbox"/>	<input type="checkbox"/>
6) określić, z jakich bloków składa się program sterowniczy?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.4. Języki programowania sterowników PLC

### 4.4.1. Materiał nauczania

#### Rodzaje języków programowania sterowników PLC

Oprogramowanie narzędziowe umożliwia tworzenie programów dla sterownika w językach tekstowych i językach graficznych odpowiednich dla każdego typu sterownika. Języki tekstowe możemy podzielić na:

- języki list instrukcji,
- języki strukturalne.

Języki graficzne to:

- schematy drabinkowe ,
- schematów blokowych .

Najczęściej wykorzystywane są listy instrukcji, schematy drabinkowe i blokowe.

#### Schemat stykowy KOP

Programowanie schematów drabinkowych jest bardzo podobne do tworzenia schematów stykowo-przełącznikowych układów sterowania elektrycznego. Często schemat rozwinięty sterowania stycznikowego przetwarzany jest na schemat stykowy KOP. Można również dokonać konwersji, za pomocą odpowiednich funkcji oprogramowania użytkowego sterownika, programu napisanego w języku instrukcji na schemat stykowy KOP.

W tab. 1 przedstawiono symbole używane w schemacie stykowym KOP.

**Tabela 1** Podstawowe symbole używane w schemacie stykowym KOP.

symbol schematu stykowego	sygnał
--][--	niezanegowany wejściowy
--]/[--	zanegowany wejściowy
--()--	niezanegowany wyjściowy

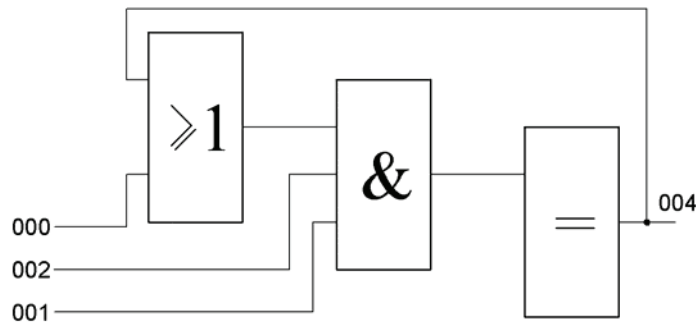
Schemat stykowy KOP przedstawia poziomo poszczególne gałęzie prądowe układu zawierające odpowiednio połączone ze sobą symbole obrazujące sygnały. Połączenie symboli odwzorowuje ich powiązanie w funkcji logicznej. Na początku każdej nowej gałęzi (lewa strona) jest podana informacja ile sygnałów wejściowych i wyjściowych sterownika zostało powiązanych ze sobą funkcjami logicznymi. Symbole posiadają adresy sygnałów (wejściowych, wyjściowych, merkerów).



**Rys. 10.** Przykładowy fragment schematu stykowego.

#### Języki schematów blokowych

Języki schematów blokowych posiadają symbole podobne do symboli używanych w technice cyfrowej. Przedstawiają one wzajemne powiązania sygnałów za pomocą funkcji logicznych. Schematy blokowe (zwane też funkcyjnymi lub logicznymi) umożliwiają szybką analizę działania skomplikowanych układów sterowania.



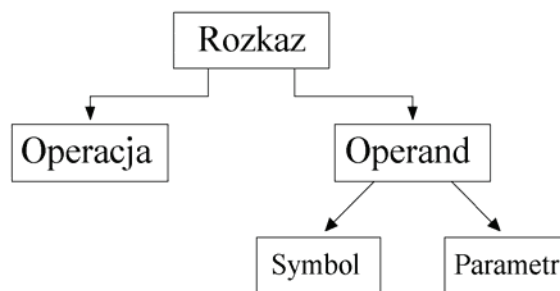
Rys. 11. Przykładowy schemat funkcjonalny.

### Języki list instrukcji

Najbardziej uniwersalne są języki list instrukcji. Każdy język posiada zbiór instrukcji (rozkażów). Program napisany w języku list instrukcji jest zapisem symbolicznym, stanowiącym ciąg kolejno ułożonych jedna pod drugą instrukcji. Jest to zapis zwarty operujące na skrótach literowych symbolizujących np.: wejścia lub wyjścia sterownika, operacje logiczne, bloki funkcyjne. Wszystkie instrukcje (rozkaży) programu są tłumaczone przez kompilator na kod maszynowy zrozumiały dla sterownika.

Instrukcja (rozkaż) zazwyczaj składa się z operatora i operandu – argumentu, niektóre rozkaży posiadają jedynie operator. Każda instrukcja posiada swój adres w programie użytkownika. Operator określa działanie jakie ma być wykonane. Argument to stała lub zmienna poddana temu działaniu. Argument składa się z symbolu i parametru. Symbol określa typ zmiennej parametr podaje jej adres. Operandami mogą być:

- zmienne wejściowe,
- zmienne wyjściowe,
- merkery, czyli zmienne wewnętrzne informujące o stanie przetwarzania,
- zmienne licznikowe przetwarzane przez bloki funkcyjne realizujące liczenie,
- zmienne czasowe przetwarzane przez bloki funkcyjne realizujące odmierzenie czasu.



Rys. 12. Struktura rozkaży.

Rozkaży sterownika możemy podzielić na bitowe i organizacyjne. Rozkaży bitowe działają na operandach, to np. rozkaży :

- ładowania operandu,
- funkcji logicznej,
- przypisania,
- zapisania przerzutnika RS,
- zerowania przerzutnika RS itp.



Rozkazy organizacyjne odnoszą się do struktury programu i są to np. rozkazy:

- końca programu,
- skoku warunkowego,
- skoku bezwarunkowego,
- końca podprogramu,
- komentarza itp.

Program napisany w języku list instrukcji jest zapisem symbolicznym, stanowiącym ciąg kolejno ułożonych, jedna pod drugą instrukcji. Jest to zapis zwarty operujące na skrótach literowych symbolizujących np.: wejścia lub wyjścia sterownika, operacje logiczne, bloki funkcyjne. Każdy program kończy się odpowiednim dla danego języka rozkazem końca programu.

#### 4.4.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Jakie znasz rodzaje języków programowania sterowników PLC?
2. Jak dzielimy języki tekstowe?
3. Jak dzielimy języki graficzne?
4. Czym charakteryzuje się schemat stykowy KOP?
5. Jaki informacje zawiera schemat blokowy?
6. Jaka jest struktura rozkazu języka list instrukcji?
7. Jak można podzielić rozkazy języka list instrukcji?
8. Jaki rozkaz kończy każdy program napisany w języku tekstowym?

#### 4.4.3. Ćwiczenia

##### Ćwiczenie 1

Wypisz, z przygotowanego programu, po dwa przykłady rozkazów bitowych i organizacyjnych. W rozkazach bitowych określ rodzaj operandu oraz jego adres.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) zapoznać się z listą rozkazów sterownika,
- 2) zanalizować przygotowany przez nauczyciela program dla sterownika,
- 3) wypisać dwa przykłady rozkazów organizacyjnych,
- 4) wypisać dwa przykłady rozkazów bitowych,
- 5) określić rodzaj i adres operandu w podanych przykładach rozkazów organizacyjnych,
- 6) zaprezentować wyniki pracy.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika wraz z listą rozkazów,
- wydruk programu dla sterownika napisanego w języku list instrukcji,
- zeszyt,
- literatura uzupełniająca zgodna z punktem 6.

## Ćwiczenie 2

Wykonaj konwersję programu napisanego w języku list instrukcji na schemat stykowy KOP. Znajdź trzy fragmenty w jednej i drugiej postaci programu, w których użyto tych samych operandów.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) uruchomić oprogramowanie użytkowe sterownika,
- 2) dokonać konwersji przygotowanego przez nauczyciela programu napisanego w języku list instrukcji na schemat stykowy KOP, za pomocą odpowiednich funkcji oprogramowania użytkowego,
- 3) wydrukować program w postaci schematu stykowego KOP,
- 4) porównać przygotowany przez nauczyciela program oraz schemat stykowy KOP,
- 5) zaznaczyć na obu wydrukach trzy fragmenty, w których użyto tych samych operandów,
- 6) zaprezentować wyniki pracy.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika wraz z lista rozkazów,
- programator sterownika wraz oprogramowaniem narzędziowym,
- drukarka,
- program dla sterownika napisany w języku list instrukcji w postaci źródłowej,
- wydruk programu dla sterownika napisanego w języku list instrukcji,
- literatura uzupełniająca zgodna z punktem 6.

## Ćwiczenie 3

Wykonaj konwersję schematu funkcjonalnego na program napisany w języku list instrukcji. Znajdź w obu postaciach programu funkcje przypisania.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) uruchomić oprogramowaniem użytkowe sterownika,
- 2) uruchomić funkcje obsługujące
- 3) dokonać konwersji przygotowanego przez nauczyciela schematu funkcjonalnego na program w postaci listy instrukcji,
- 4) wydrukować program w postaci listy instrukcji,
- 5) przeanalizować i porównać wydruki schematu funkcyjnego programu i listy instrukcji,
- 6) znaleźć i zaznaczyć na obu wydrukach te fragmenty, które zawierają funkcje przypisania,
- 7) zaprezentować wyniki pracy.

Wyposażenie stanowiska pracy:

- dokumentacja techniczna sterownika wraz z lista rozkazów,
- programator sterownika wraz oprogramowaniem narzędziowym,
- drukarka,
- program dla sterownika w postaci schematu funkcjonalnego,
- wydruk programu dla sterownika w postaci schematu funkcjonalnego,
- literatura uzupełniająca zgodna z punktem 6.

#### 4.4.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) rozpoznać program dla sterownika napisany w języku list instrukcji?	<input type="checkbox"/>	<input type="checkbox"/>
2) rozpoznać program dla sterownika w postaci schematu stykowego KOP?	<input type="checkbox"/>	<input type="checkbox"/>
3) rozpoznać program dla sterownika w postaci schematu funkcjonalnego?	<input type="checkbox"/>	<input type="checkbox"/>
4) rozróżnić w rozkazie języka list instrukcji operand i operację?	<input type="checkbox"/>	<input type="checkbox"/>
5) rozróżnić w rozkazie języka list instrukcji adres operandu?	<input type="checkbox"/>	<input type="checkbox"/>
6) zidentyfikować w rozkazie języka list instrukcji rodzaj operandu?	<input type="checkbox"/>	<input type="checkbox"/>
7) wykonać konwersję programu napisanego w języku list instrukcji na schemat stykowy KOP?	<input type="checkbox"/>	<input type="checkbox"/>
8) wykonać konwersję schematu stykowego KOP na program napisanego w języku list instrukcji?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.5. Podstawowe funkcje w programach sterowniczych

### 4.5.1. Materiał nauczania

Operatory funkcji posiadają znormalizowane oznaczenia niemieckie i angielskie oraz matematyczne, natomiast producenci sterowników stosują własne oznaczenia odpowiednie dla języka instrukcji danego sterownika. Każdy operator funkcji posiada również własny symbol graficzny stosowany w schematach funkcyjnych.

#### Funkcja „ładuj” (load)

Funkcja „ładuj” (load) może być wykorzystywana w programie wielokrotnie. Wprowadza ona do pamięci roboczej sterownika zmienną. Operacja „ładuj” zawsze poprzedza funkcje logiczne, matematyczne oraz bloki funkcyjne.

Tabela 2. Oznaczenia operacji „ładuj”

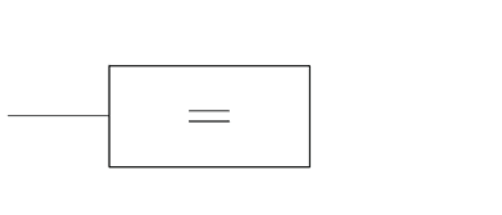
znormalizowane oznaczenia operacji „ładuj”			przykłady oznaczeń operacji „ładuj” stosowane przez różnych producentów			
niemieckie	angielskie	matematyczne	U	O	LD	!
L	L	!				

#### Funkcja „=” przypisania (przyporządkowania, przekazania)

Funkcja „=” przypisania (przyporządkowania, przekazania) może być wykorzystywana w programie wielokrotnie. Kończy (zamyka) ona określone fragmenty programu np. zawierający funkcję logiczną, której wynik zostaje przypisany operandowi np. wyjściu sterownika.

Funkcja przypisania posiada jedno powszechnie stosowane oznaczenie „=”.

Na schemacie funkcjonalny reprezentowana jest przez symbol graficzny przedstawiony na rys. 13.



Rys. 13. Symbol graficzny funkcji przypisania [9, s. 170]

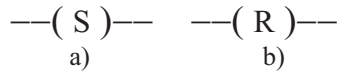
#### Przerzutnik SR i RS

Przerzutniki SR i RS służą do zapamiętania krótko trwającego sygnału. W przerzutniku RS pierwsza w kolejności pojawia się instrukcja zawierająca operator S, która ustawia argument lub merker na wartość 1. Po niej w programie występują inne instrukcje działające na tym operandzie. Następnie druga pojawia się instrukcja z operatorem R, ustawiająca dany operator na wartość 0. Przerzutnik RS zwany jest też przerzutnikiem z dominującym wejściem zerującym (wejściem R).

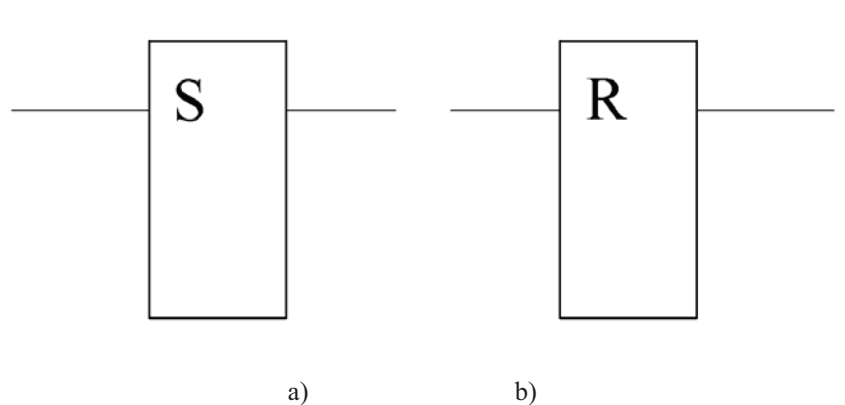
W przypadku przerzutnika SR pierwsza w kolejności pojawia się instrukcja zawierająca operator R, która ustawia argument lub merker na wartość 0. Po niej w programie występują inne instrukcje działające na tym operatorze. Następnie druga pojawia się instrukcja

z operatorem S ustawiająca dany operator na wartość 1. Przerzutnik SR zwany jest też przerzutnikiem z dominującym wejściem ustawiającym (wejściem S).

Rozkazy z operatorami S i R są przedstawiane w schematach stykowych KOP oraz funkcyjnych za pomocą odpowiednich symboli graficznych.



Rys. 14. Symbole graficzne rozkazu a) ustawienia S, b) zerowania R, używane w schematach stykowych KOP [9, s. 170]



Rys 15. Symbole graficzne rozkazu a) ustawienia S, b) zerowania R, używane w schematach funkcyjnych [9, s. 170]

#### 4.5.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Jakie znasz podstawowe funkcje wykorzystywane w programach sterowniczych?
2. Jak działa funkcja ładowania?
3. Jak działa funkcja przypisania?
4. W jakim celu wykorzystuje się przerzutnik SR (RS)?
5. Jak działa rozkaz ustawiania?
6. Jak działa rozkaz zerowania?
7. Na jakich zmiennych operują funkcje logiczne?

#### 4.5.3. Ćwiczenia

##### Ćwiczenie 1

Napisz program sterowniczy, w wyniku którego, lampka sygnalizacyjna świeci się przy wciśniętym zwiernym przycisku sterowniczym S1. Po zwolnieniu przycisku gaśnie.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,

- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

## Ćwiczenie 2

Napisz program sterowniczy, w wyniku którego, lampka sygnalizacyjna zaświeci się po wciśnięciu zwiernego przycisku sterowniczego S1 i świeci nadal po jego zwolnieniu. Wciśnięcie przycisku S2 powoduje zgaszenie lampki.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

### 4.5.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) określić, kiedy używa się w programie sterowniczym funkcji ładowania?	<input type="checkbox"/>	<input type="checkbox"/>
2) określić, kiedy używa się w programie sterowniczym funkcji przypisania?	<input type="checkbox"/>	<input type="checkbox"/>
3) określić, kiedy stosuje się przerzutnik z dominującym wejściem ustawiającym?	<input type="checkbox"/>	<input type="checkbox"/>
4) określić, kiedy stosuje się przerzutnik z dominującym wejściem zerującym ?	<input type="checkbox"/>	<input type="checkbox"/>
5) rozpoznać na schemacie stykowym KOP symbol graficzny określający wejście ustawiające przerzutnika RS?	<input type="checkbox"/>	<input type="checkbox"/>
6) rozpoznać na schemacie stykowym KOP symbol graficzny określający wejście zerujące przerzutnika RS?	<input type="checkbox"/>	<input type="checkbox"/>
7) rozpoznać na schemacie funkcyjnym symbol graficzny przerzutnika RS ?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.6. Podstawowe funkcje logiczne

### 4.6.1. Materiał nauczania

Programowanie każdej funkcji logicznej rozpoczyna się rozkazem ładowania, natomiast kończy rozkazem przypisania. Sygnał wyjściowy może być zaprogramowany tylko jeden raz. Operand będący wynikiem funkcji logicznej może sterować dowolną ilością wyjść sterownika.

Operatory funkcji logicznych posiadają znormalizowane oznaczenia niemieckie i angielskie oraz matematyczne, natomiast producenci sterowników stosują własne oznaczenia odpowiednie dla języka instrukcji danego sterownika. Każdy operator funkcji logicznej posiada również własny symbol graficzny stosowany w schematach funkcyjnych.

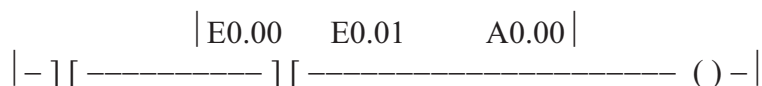
#### Iloczyn logiczny I (koniunkcja)

Wyjście iloczynu logicznego przyjmuje wartość 1 jeżeli wszystkie operandy jakie wiąże miały wartość 1.

**Tabela 3.** Oznaczenia iloczynu logicznego

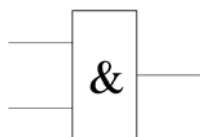
znormalizowane oznaczenia iloczynu logicznego I			przykłady oznaczeń iloczynu logicznego I stosowane w listach instrukcji przez różnych producentów			
niemieckie	angielskie	matematyczne	UND	AND	&	U
U	A	&				

Na schemacie stykowym KOP iloczyn logiczny przedstawiany jako połączenie szeregowe operandów.



**Rys. 16.** Przykład realizacji funkcji iloczynu logicznego w schemacie stykowym KOP

Iloczyn logiczny posiada swój symbol graficzny używany w schematach funkcyjnych. Operator iloczynu logicznego może mieć więcej niż dwa wejścia.



**Rys. 17.** Symbol graficzny iloczynu logicznego używany w schematach funkcyjnych [9, s. 170]

#### Suma logiczna LUB

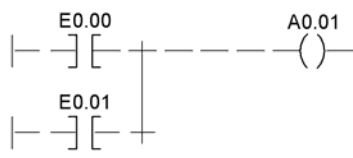
Wyjście sumy logicznej LUB przyjmuje wartość 1, jeżeli przynajmniej jeden z operandów jakie wiąże ma wartość 1.



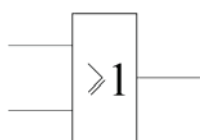
**Tabela 4.** Oznaczenia sumy logicznej

znormalizowane oznaczenia sumy logicznej LUB			przykłady oznaczeń iloczynu logicznego I stosowane w listach instrukcji przez różnych producentów			
niemieckie	angielskie	matematyczne	OR	ODER	/	O
O	O	/				

Na schemacie stykowym KOP suma logiczna przedstawiana jest jako połączenie równoległe operandów.



**Rys. 18.** Przykład realizacji funkcji sumy logicznej w schemacie stykowym KOP. Suma logiczna posiada swój symbol graficzny używany w schematach funkcyjnych. Operator sumy logicznej może mieć więcej niż dwa wejścia.

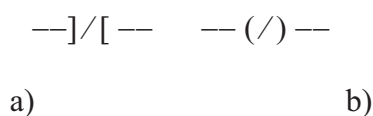


**Rys. 19.** Symbol graficzny sumy logicznej używany w schematach funkcyjnych [9, s. 170]

### Negacja NIE

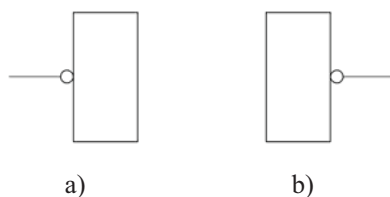
Negacja jest funkcją jednoelementową zmieniającą wartość binarną operandu na przeciwną. Jeżeli operand miał wartość 1 logicznej to po negacji będzie miał wartość 0 logicznego. Natomiast jeżeli operand miał wartość 0 logicznego to po negacji będzie miał wartość 1 logicznej. Negacji mogą być poddane wszystkie operandy bitowe.

W schemacie stykowym KOP zanegowane operandy posiadają swoje symbole graficzne.



**Rys. 20.** Symbole graficzne używane w schemacie KOP a) zanegowane wejście, b) zanegowane wyjście.

W schemacie funkcyjnym zanegowane operandy posiadają swoje symbole graficzne.

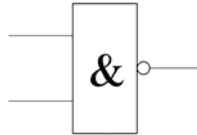


**Rys. 21.** Symbole graficzne używane na schemacie funkcyjnym a) zanegowane wejście, b) zanegowane wyjście [9, s. 170]

Negacja jest również łączona sumą logiczną i iloczynem logicznym.

### Zanegowany iloczyn logiczny

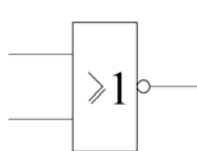
Zanegowany iloczyn logiczny przyjmuje wartość 0, jeżeli wszystkie operandy jakie wiąże miały wartość 1.



Rys. 22. Symbol graficzny zanegowanego iloczynu logicznego używany w schematach funkcyjnych [9, s. 170]

### Zanegowana suma logiczna

Zanegowana suma logiczna przyjmuje wartość 0, jeżeli przynajmniej jeden z operandów operandy jakie wiąże ma wartość 1.



Rys. 2. Symbol graficzny sumy logicznej używany w schematach funkcyjnych [9, s. 170]

W schematach stykowych KOP stosuje się negacje operandów tworzących funkcje logiczne.

Tabela 5. Oznaczenia negacji

znormalizowane oznaczenia negacji			przykłady oznaczeń negacji stosowane w listach instrukcji przez różnych producentów			
niemieckie	angielskie	matematyczne	NOT	N	E dopisane przed rozkazem	I dopisane do rozkazu
N	N					

### Merkery

W przypadku skomplikowanych struktur logicznych stosuje się merkery, czyli komórki pamięci przechowujące wyniki pośrednich funkcji logicznych. Merker programowany jest jak wyjście sterownika, lecz nie może on oddziaływać bezpośrednio na urządzenie podłączone do sterownika. Oddziałuje on natomiast na końcowy wynik operacji logicznych, czyli pośrednio również na urządzenia podłączone do wyjścia sterownika.

Istnieją dwa rodzaje merkerów: nieremanentne i remanentne. Merkery nieremanentne tracą zapisaną informację przy zaniku napięcia zasilającego sterownik. Natomiast merkery remanentne nie tracą informacji przy zaniku napięcia zasilającego sterownik. Remanentność merkera może być ustawiana za pomocą specjalnych funkcji oprogramowania użytkowego sterownika.

## 4.6.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Kiedy iloczyn logiczny przyjmuje wartość 1?
2. Kiedy zanegowany iloczyn logiczny przyjmuje wartość 0?
3. Kiedy suma logiczna przyjmuje wartość 1?
4. Kiedy zanegowana suma logiczna przyjmuje wartość 0?
5. Ile argumentów może mieć funkcja negacji?
6. Kiedy wynik negacji jest równy 1?
7. Jak wygląda symbol graficzny zanegowanego sygnału wyjściowego używany na schematach stykowych KOP?
8. Jakie jest przeznaczenie merkerów?
9. Co oznacza remanentność merkera?

## 4.6.3. Ćwiczenia

### Ćwiczenie 1

Napisz program sterowniczy wykorzystujący funkcje logiczne, w wyniku którego, lampka sygnalizacyjna świeci się przy wciśnięciu przycisków sterowniczych zwiernych S1 i S2. Po zwolnieniu jednego lub obu przycisków świeci nadal.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

### Ćwiczenie 2

Napisz program sterowniczy wykorzystujący funkcje logiczne, w wyniku którego, lampka sygnalizacyjna może być zaświecana po wciśnięciu zwiernego przycisku sterowniczego S3 lub S4 i świeci nadal po ich zwolnieniu. Wciśnięcie przycisku S5 powoduje zgaszenie lampki.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,

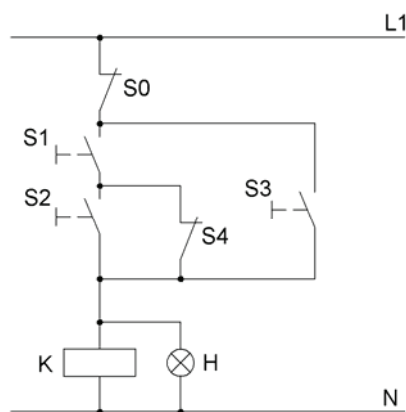
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- sterownik
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

### Ćwiczenie 3

Napisz program realizujący przedstawiony na rysunku schemat rozwinięty sterowania stycznikowego. Symbole S1, S2, S3 oznaczają przyciski zwierne, S0 i S4 przyciski rozwierne, K stycznik, H lampkę sygnalizacyjną.



Schemat rozwinięty sterowania stycznikowego

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy

Wyposażenie stanowiska pracy:

- sterownik ,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

#### 4.6.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) określić, kiedy używa się w programie sterowniczym funkcji iloczynu logicznego?	<input type="checkbox"/>	<input type="checkbox"/>
2) określić, kiedy używa się w programie sterowniczym funkcji sumy logicznej?	<input type="checkbox"/>	<input type="checkbox"/>
3) stworzyć fragment schematu stykowego KOP zawierający sumę logiczną?	<input type="checkbox"/>	<input type="checkbox"/>
4) stworzyć fragment schematu stykowego KOP zawierający iloczyn logiczny?	<input type="checkbox"/>	<input type="checkbox"/>
5) rozpoznać na schemacie stykowym KOP symbol graficzny określający zanegowany operand?	<input type="checkbox"/>	<input type="checkbox"/>
6) określić, kiedy zanegowana suma logiczna przyjmuje wartość 0?	<input type="checkbox"/>	<input type="checkbox"/>
7) określić, kiedy zanegowany iloczyn logiczny przyjmuje wartość 0?	<input type="checkbox"/>	<input type="checkbox"/>
8) określić, kiedy stosuje się merkery?	<input type="checkbox"/>	<input type="checkbox"/>
9) rozpoznać na schemacie funkcyjnym poszczególne symbole funkcji logicznej?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.7. Podstawowe moduły sterowników PLC

### 4.7.1. Materiał nauczania

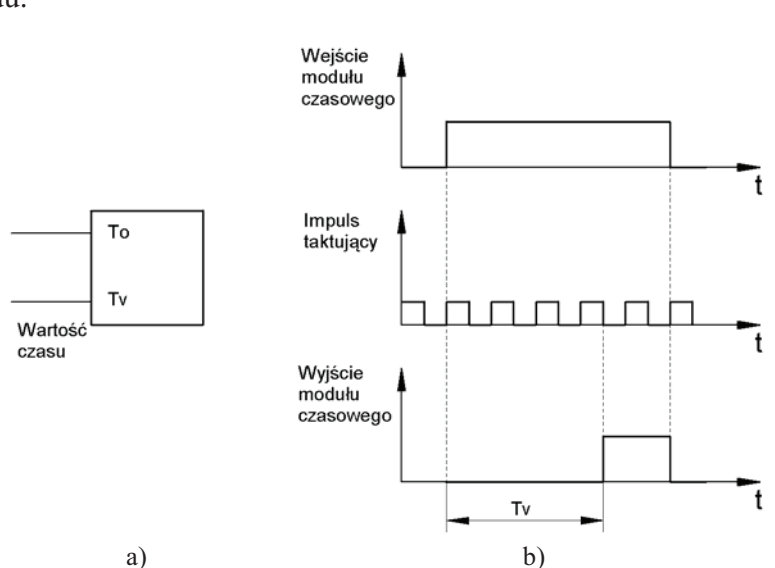
Producenci sterowników posługują się różnymi oznaczeniami graficznymi modułów używanymi w schematach funkcyjnych i stykowych. Oznaczenia modułów programowych sterownika używane w listach instrukcji zależą również od producenta.

#### Moduł czasowy

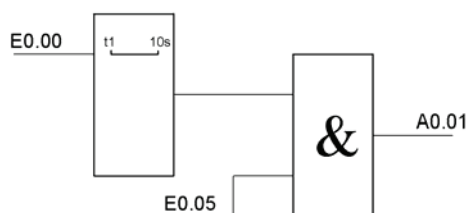
Moduły czasowe w sterownikach PLC służą do realizacji działań czasowych np. odmierzenia czasu opóźnienia, wytwarzania impulsu o określonym czasie trwania, wytwarzania ciągu impulsów.

Po wywołaniu w programie modułu czasowego pojawia się odpowiednia maska umożliwiająca zaprogramowanie jego wejść i wyjścia. Moduł czasowy uruchamiany jest poprzez operand startu, który może być sygnałem wejściowym sterownika lub wynikiem realizacji funkcji logicznej. Wartość odmierzanego czasu  $T_v$  ustawiana jest programowo. Inicjalizacja czasomierza powoduje uruchomienie licznika zliczającego w tył impulsy wewnętrznego sygnału taktującego, sterującego pracą mikroprocesora sterownika PLC. Gdy stan licznika osiągnie wartość zero, moduł czasowy sygnalizuje to na wyjściu. Opisany sposób odmierzenia czasu jest bardzo dokładny ze względu na wykorzystanie sygnału sterującego pracą mikroprocesora.

W zależności od typu sterownika czasomierze posiadają dodatkowe funkcje np. umożliwiające odczytanie aktualnej wartości odliczanego czasu zapisanej w odpowiednio zaadresowanych komórkach pamięci, zatrzymanie odliczania czasu przy określonej wartości jakiegoś operandu.



Rys.24. a) Przykładowy symbol modułu czasowego używany w schemacie funkcyjnym. b) Przebiegi czasowe obrazujące działanie modułu czasowego [11, s. 62]



Rys.25. Przykład użycia modułu czasowego w schemacie funkcyjnym [11, s. 62]



Rys. 26. Przykład użycia modułu czasowego w schemacie stykowym KOP

Ilość modułów czasowych, jakimi dysponuje sterownik jest zależna od jego typu i producenta.

## Licznik

Licznik umożliwia określenie ilości poprzez zliczanie impulsów. Źródłem zliczanych impulsów mogą być zewnętrzne czujniki sygnalizujące pojawienie się obiektu w polu ich oddziaływania. Po wywołaniu w programie licznika pojawia się odpowiednia maska umożliwiająca jego zaprogramowanie.

Liczniki posiadają następujące wejścia:

- zliczające na które podawane są poprzez odpowiedni operand bitowy zliczane impulsy (stan operandu może się również zmieniać po wykonaniu określonej funkcji logicznej),
- ustawiające stan początkowy licznika S – jeśli operand na tym wejściu zmieni stan z 0 na 1 do licznika zostanie wprowadzona jego wartość początkowa,
- zerujące R – podanie na niego operandu o wartości 1 powoduje wyzerowanie stanu licznika.

Stan początkowy licznika ustawiany jest programowo.

Liczniki mogą posiadać następujące wyjścia:

- umożliwiający odczytanie aktualnego stanu licznika który jest zapisywany w odpowiednio zaadresowanych komórkach pamięci,
- statusu licznika na którym utrzymuje się wartość 1 jeśli aktualny stan licznika jest różny od zera,
- informujący, że stan licznika osiągnął wartość zero.

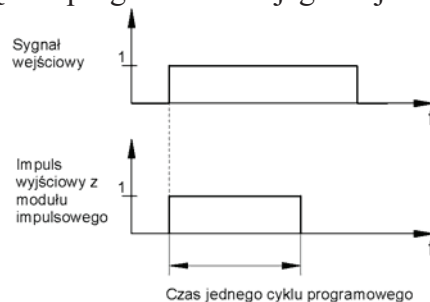
Licznik może liczyć

- w górę – jest to licznik dodający, którego wartość zwiększa się o jeden po zmianie z 0 na 1 wartości operandu podanego na wejście zliczające,
- w dół – jest to licznik odejmujący, którego wartość zmniejsza się o jeden po zmianie z 0 na 1 wartości operandu podanego na wejście zliczające.

Sterowniki zależnie od typu posiadają różną ilość liczników mogących pracować niezależnie.

## Moduł impulsowy

Moduł impulsowy przetwarza sygnał ciągły na pojedynczy impuls. Posiada on jedno wejście na które podawany jest sygnał o wartości 1 logicznej. Zbocze narastające tego sygnału inicjuje pojawienie się na wyjściu modułu impulsowego impulsu trwającego jeden cykl programowy sterownika. wywołaniu w programie modułu impulsowego pojawia się odpowiednia maska umożliwiająca zaprogramowanie jego wejścia i wyjścia.



Rys. 27. Przebiegi czasowe sygnałów na wejściu i wyjściu modułu impulsowego



Moduł impulsowy może być wykorzystywany w sterowaniu wszelkiego rodzaju automatów. Sterownik może posiadać dowolną ilość modułów impulsowych pracujących niezależnie.

#### 4.7.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Jakie znasz moduły programowe używane w sterownikach PLC?
2. Jak działa moduł czasowy?
3. Jakiego zastosowania licznik?
4. Jakiego wejścia posiada licznik?
5. Jak działa moduł impulsowy?

#### 4.7.3. Ćwiczenia

##### Ćwiczenie 1

Napisz program sterujący załączaniem oświetlenia z najniższego poziomu budynku. Oświetlenie na czterech dolnych poziomach klatki schodowej zostaje załączone po naciśnięciu łącznika przyciskowego zwiernego S1, po 2 minutach załączone zostaje oświetlenie na następnych czterech poziomach. Oświetlenie całej klatki schodowej gaśnie po kolejnych 3 minutach lub może być wyłączone za pomocą przycisku rozwiernego S2. Jako elementy symulujące oświetlenie na poszczególnych czterech poziomach wykorzystaj dwie lampki sygnalizacyjne.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

##### Ćwiczenie 2

Napisz program kontrolujący ilość elementów w pojemniku. Elementy z taśmociągu dostarczane są do pojemnika przez wejście w jego górnej części zaopatrzone w czujnik optyczny. Po naciśnięciu przez pracownika przycisku zwiernego S1 otwierane jest wyjście umieszczone w dolnej części pojemnika przez, które wydostaje się na zewnątrz jeden element. W pojemniku mieści się 50 elementów. Jeśli pojemnik jest całkowicie wypełniony

powinna zaświecić się lampka czerwona. Wolne miejsca w pojemniku sygnalizuje lampka zielona. Urządzenie załączane jest przyciskiem zwiernym S0.

#### Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) narysować schemat funkcyjny programu,
- 4) sporządzić listę przyporządkowującą,
- 5) napisać program realizujący opisane sterowanie,
- 6) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 7) uruchomić i sprawdzić poprawność działania napisanego programu,
- 8) zaprezentować wyniki swojej pracy

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

#### 4.7.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) określić, w jaki sposób należy zaprogramować moduł czasowy sterownika tak by odliczał określony czas?	<input type="checkbox"/>	<input type="checkbox"/>
2) określić, w jaki sposób należy zaprogramować licznik sterownika tak by liczył do przodu?	<input type="checkbox"/>	<input type="checkbox"/>
3) określić, w jaki sposób należy zaprogramować licznik sterownika tak by liczył w tył?	<input type="checkbox"/>	<input type="checkbox"/>
4) wyjaśnić, jak napisać program sterowniczy w wyniku, którego generowany będzie impuls o czasie trwania jednego cyklu programowego?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.8. Słowa informacyjne

### 4.8.1. Materiał nauczania

Informacja zawarta w operandach bitowych o wartościach 1 lub 0 często nie jest wystarczająca do rozwiązywania skomplikowanych problemów sterowania. Stosuje się wówczas słowa informacyjne składające się zazwyczaj z 8 lub 16 bitów. Bit najmniej znaczący zajmuje skrajną pozycję z prawej strony słowa (pozycja nr 0), natomiast najbardziej znaczący skrajną pozycję z lewej strony. Informacje jaką zawiera słowo zapisana jest w kodzie dwójkowym.

Wartość dziesiętną słowa bitowego oblicza się tak jak w kodzie binarnym. Każdej pozycji przypisana jest waga równa  $2^k$  gdzie  $k$  określa numer pozycji bitu. 16-bitowe słowo informacyjne dzielone jest na dwa bajty starszy i młodszy.

**Tabela 6.** 16-bitowe słowo informacyjne: rozmieszczenie bitów, odpowiadające im wartości dziesiętne, podział na bajty.

15	14	13	12	11	10	09	08	07	06	05	04	03	02	01	00
$2^{15}$	$2^{14}$	$2^{13}$	$2^{12}$	$2^{11}$	$2^{10}$	$2^9$	$2^8$	$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$
starszy bajt								młodszy bajt							

**Tabela 7.** 8-bitowe słowo informacyjne: rozmieszczenie bitów, odpowiadające im wartości dziesiętne.

07	06	05	04	03	02	01	00
$2^7$	$2^6$	$2^5$	$2^4$	$2^3$	$2^2$	$2^1$	$2^0$

Słowa informacyjne podlegają operacjom logicznym przeprowadzanym na odpowiadających sobie w każdym słowie bitach. Wynikiem takich operacji jest również słowo informacyjne.

Przykład

Wynikiem sumy logicznej merkera słownego MW1= 01100101 oraz merkera słownego MW2=10011001 jest merker słowny o wartości równej 11111101.

Wynikiem iloczynu logicznego tych samych merkerów słownych MW1 i MW2 jest merker słowny o wartości równej 00000001.

### Moduły programowe operujące na słowach informacyjnych.

Sterowniki posiadają moduły programowe, których operandami są słowa informacyjne.

### Moduły funkcyjne

Na słowach informacyjnych zapisanych w rejestrach danych można przeprowadzać działania arytmetyczne, które realizowane są przez moduły funkcyjne.

Moduły funkcyjne są gotowymi podprogramami zbudowanymi z merkerów zawierających funkcje specjalne takie jak: dodawanie, odejmowanie, mnożenie, czy porównywanie dwóch słów informacyjnych. Moduły te korzystają czasami dodatkowych merkerów pomocniczych których przeznaczenie jest podobne do bitów rejestru flagowego. Moduły funkcyjne są przechowywane w pamięci ROM i mogą być wywoływane podobnie jak moduły czasowe impulsowe, czasowe czy liczniki. Dane przetwarzane przez moduły funkcyjne są najpierw ładowane do rejestrów danych by mogły być w każdej chwili dostępne dla programu sterowniczego.

Bardzo rozbudowanymi modułami funkcyjnymi są regulatory PID.

## Licznik impulsów wyższej częstotliwości

Niektóre sterowniki posiadają licznik impulsów wyższej częstotliwości wykorzystywany do zliczania impulsów o częstotliwości rzędu 10kHz. Licznik taki może liczyć jedynie do przodu. Zazwyczaj ma przypisane konkretne wejście bitowe sterownika z którego zlicza impulsy. Moduł ten posiada wejście zerujące oraz wyjście gdzie w postaci słowa informacyjnego zapisany jest stan licznika.

## Moduł rejestru przesuwnego

Rejestr przesuwany używany jest do przesuwania informacji bitowej w prawo lub w lewo w wyniku czego zawiera on słowo informacyjne.

Po wywołaniu w programie modułu rejestru pojawia się odpowiednia maska umożliwiająca jego zaprogramowanie. Moduł ten posiada wejścia:

- danych – po podaniu na nie aktywnego operandu bitowego (zbocza narastającego lub opadającego) następuje przejście przez rejestr lewego lub prawego bitu informacji,
- przesuwające - po podaniu na nie aktywnego operandu bitowego (zbocza narastającego lub opadającego) następuje przesunięcie informacji zawartej w rejestrze o jedno miejsce w prawo lub lewo,
- zerujące zawartość rejestru.

### 4.8.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. Jak zbudowane jest słowo informacyjne?
2. W jakim kodzie zapisana jest informacja podana w słowie informacyjnym?
3. Jakie operacje można wykonywać na słowach informacyjnych?
4. W jaki sposób przeprowadza się operacje logiczne na słowach informacyjnych?
5. W jakim celu używany jest rejestr przesuwany?
6. Jaka jest różnica pomiędzy licznikiem impulsów wyższej częstotliwości a modułem programowym licznika w sterownikach PLC?

### 4.8.3. Ćwiczenia

#### Ćwiczenie 1

Wykonaj operację zanegowanej sumy logicznej i zanegowanego iloczynu logicznego na merkerach słownych MW1=0011001111010101 oraz MW2=1010110100110101.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować treść zadania,
- 2) wykonać operację zanegowanej sumy logicznej,
- 3) wykonać operację zanegowanego iloczynu logicznego,
- 4) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- zeszyt,
- literatura uzupełniająca zgodna z punktem 6.

## Ćwiczenie 2

Napisz program w postaci listy rozkazów, sterujący oświetleniem korytarza łączącego halę produkcyjną z magazynem. W korytarzu znajduje się 8 punktów świetlnych sterowanych wyjściami sterownika, które powinny zapalać się i gasnąć kolejno. Każdy punkt świetlny powinien świecić przez 30s. Oświetlenie uruchamiane jest przyciskiem zwiernym, umieszczonym przy wejściu do korytarza, natomiast przycisk rozwierny S10 wyłącza sterowanie.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) sporządzić listę przyporządkowującą,
- 4) napisać program realizujący opisane sterowanie,
- 5) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 6) uruchomić i sprawdzić poprawność działania napisanego programu,
- 7) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

### 4.8.4. Sprawdzian postępów

**Czy potrafisz:**

	<b>Tak</b>	<b>Nie</b>
1) wykonać sumę logiczną na merkerach słownych?	<input type="checkbox"/>	<input type="checkbox"/>
2) wykonać zanegowaną sumę logiczną na merkerach słownych?	<input type="checkbox"/>	<input type="checkbox"/>
3) wykonać iloczyn logiczny na merkerach słownych?	<input type="checkbox"/>	<input type="checkbox"/>
4) wykonać zanegowany iloczyn logiczny na merkerach słownych?	<input type="checkbox"/>	<input type="checkbox"/>
5) opisać działanie licznika impulsów wyższej częstotliwości?	<input type="checkbox"/>	<input type="checkbox"/>
6) opisać działanie rejestru przesuwającego?	<input type="checkbox"/>	<input type="checkbox"/>

## 4.9. Obróbka sygnału analogowego w sterownikach PLC

### 4.9.1. Materiał nauczania

Sterowniki PLC posiadają zazwyczaj moduł, umożliwiający obróbkę sygnałów analogowych pochodzących np. z czujników temperatury czy ciśnienia. Napięcia analogowe jakie mogą być przetwarzane przez sterowniki mieszczą się zazwyczaj w zakresie od 0V do 10V, ponieważ takie wartości sygnałów są typowe dla układów automatyzacji. Obróbka sygnałów analogowych rozpoczyna się od przetworzenia ich w przetworniku A/C na przeważnie ośmiobitowy sygnał binarny. W takim przypadku napięciom z przedziału 0V÷10V odpowiada 0÷255 sygnałów binarnych. Przetwarzanie napięcia analogowego U na ośmiobitową wartość binarna X odbywa się zgodnie z zależnością:

$$X = U \cdot \frac{255}{10}$$

Sterowniki PLC posiadają zazwyczaj kilka wejść analogowych i przynajmniej jedno wyjście.

#### Komparator

Obróbka sygnału analogowego odbywa się najczęściej w komparatorze. Jest to moduł programowy wywoływany podobnie jak moduł czasowy czy licznik. Posiada dwa wejścia na które podawane są porównywane sygnały analogowe np. na jedno z wejść E1 podany jest rzeczywisty sygnał z czujnika ciśnienia natomiast na drugie E2 wartość zadana. Komparator w sposób ciągły porównuje te wartości sprawdzając która z poniższych zależności jest spełniona:

$$E1 > E2 \qquad E1 = E2 \qquad E1 < E2$$

Komparatory wykorzystywane są najczęściej w układach regulacji. Jeśli dany sygnał musi zawierać się pomiędzy wartością minimalną a maksymalną to taką pętlę histerezy realizuje się za pomocą dwóch komparatorów: jeden porównuje wartość badanego sygnału z zadaną wartością minimalną, drugi z zadaną wartością maksymalną.

### 4.9.2. Pytania sprawdzające

Odpowiadając na pytania, sprawdzisz, czy jesteś przygotowany do wykonania ćwiczeń.

1. W jaki sposób sterownik PLC opracowuje sygnał analogowy?
2. Na czym polega przetwarzanie ośmiobitowe?
3. Jaki moduł programowy służy do obróbki sygnału analogowego?
4. W jakim celu stosuje się moduł komparatora?
5. Jakie informacje uzyskiwane są na wyjściach komparatora?

### 4.9.3. Ćwiczenia

#### Ćwiczenie 1

Oblicz cyfrową wartość odpowiadającą napięcia  $U=6V$  w przetwarzaniu ośmiobitowym.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) zanalizować treść zadania,

- 2) skorzystać z zależności:  $X = U \cdot \frac{255}{10}$
- 3) wynik tej operacji zapisać binarnie.
- 4) zaprezentować wyniki swojej pracy.

Wyposażenie stanowiska pracy:

- zeszyt,
- kalkulator,
- literatura uzupełniająca zgodna z punktem 6.

## Ćwiczenie 2

Napisz program w postaci listy instrukcji sterujący utrzymaniem poziomu wilgoci w pokoju na poziomie 50% z tolerancją 10%. W pomieszczeniu mierzony jest poziom wilgoci odpowiednim czujnikiem dającym na wyjściu sygnał 5V dla poziomu wilgotności 50%, 4,5V dla poziomu wilgotności 40%, 5,5V dla poziomu wilgotności 60%. Napięcie to podawane jest na wejście analogowe sterownika. Zbyt małą wilgotność sygnalizuje lampka H1, zbyt dużą lampka H2. Sterowanie nawilżaniem odbywa się za pomocą nawilżacza sterowanego sygnałem z wyjścia bitowego sterownika.

Sposób wykonania ćwiczenia

Aby wykonać ćwiczenie powinieneś:

- 1) przeanalizować opisaną sytuację,
- 2) narysować schemat połączenia wejść i wyjść sterownika,
- 3) sporządzić listę przyporządkowującą,
- 4) napisać program realizujący opisane sterowanie,
- 5) podłączyć zewnętrzne elementy wykonawcze do sterownika,
- 6) uruchomić i sprawdzić poprawność działania napisanego programu,
- 7) zaprezentować wyniki swojej pracy

Wyposażenie stanowiska pracy:

- sterownik,
- elementy zewnętrzne (przyciski, lampki sygnalizacyjne),
- regulowany zasilacz napięcia stałego + 10V
- programator wraz z oprogramowaniem narzędziowym,
- dokumentacja techniczna sterownika,
- literatura uzupełniająca zgodna z punktem 6.

### 4.9.4. Sprawdzian postępów

<b>Czy potrafisz:</b>	<b>Tak</b>	<b>Nie</b>
1) obliczyć wartość cyfrową odpowiadającą napięciu analogowemu w przetwarzaniu ośmiobitowym?	<input type="checkbox"/>	<input type="checkbox"/>
2) opisać przetwarzanie sygnału analogowego przez sterownik?	<input type="checkbox"/>	<input type="checkbox"/>
3) opisać działanie modułu komparatora?	<input type="checkbox"/>	<input type="checkbox"/>
4) zaprogramować komparator by porównywał dwa sygnały analogowe?	<input type="checkbox"/>	<input type="checkbox"/>



## 6. LITERATURA

1. Gerlach M., Janas R.: Automatyka dla liceum technicznego. WSiP, Warszawa 1999
2. Jabłoński W., Płoszajski G.: Elektrotechnika z automatyką. WSiP, Warszawa 2002
3. Kordowicz-Sot A.: Automatyka i robotyka. Robotyka. WSiP, Warszawa 1999
4. Kordowicz-Sot A.: Automatyka i robotyka. Układy regulacji automatycznej. WSiP, Warszawa 1999
5. Kostro J.: Elementy, urządzenia i układy automatyki. WSiP, Warszawa 1998
6. Kostro J.: Pracownia automatyki. WSiP, Warszawa 1996
7. Płoszajski G.: Automatyka. WSiP, Warszawa 1995
8. Pochopień B.: Automatyzacja procesów przemysłowych. WSiP, Warszawa 1993
9. Schmidt D., Baumann A., Kaufmann H., Paetzold H., Zippel B.: Mechatronika REA Warszawa 2002
10. Siemieniako F., Gawrysiak M.: Automatyka i robotyka. WSiP, Warszawa 1996
11. Technika sterowników z programowalną pamięcią. Praca zbiorowa. WSiP, Warszawa 1998