POLITECHNIKA ŚWIĘTOKRZYSKA

Wydział Elektrotechniki, Automatyki i Informatyki

Katedra Elektrotechniki Przemysłowej i Automatyki Zakład Urządzeń i Systemów Automatyki

Urządzenia i Systemy Automatyki

Programowanie sterownika PLC Siemens S7-200

Instrukcja laboratoryjna

Paweł Strączyński

1 Sterownik PLC

Programowalny sterownik logiczny PLC (*ang. programmable logic controller*) to urządzenie pozwalające na sterowanie pracą maszyn i procesów technologicznych według określonego algorytmu. Sterownik programowalny składa się z jednostki centralnej (CPU), modułów wejść oraz wyjść. Sterownik PLC cechuje praca cykliczna. Na początku cyklu pracy sprawdzany jest stan wejść sterownika następnie przetwarzane są kolejne instrukcje programu. Na końcu obiegu pętli w wyniku przetwarzania programu aktualizowany zostaje stan wyjść. Ze względu na złożoność budowy sterowników programowalnych wyróżnić można różne grupy tych urządzeń m.in najprostsze przekaźniki programowalne, sterowniki kompaktowe oraz modułowe. Każdy z producentów oferujących sterowniki programowalne posiada w swej ofercie kilka różnych rodzin sterowników do różnych zastosowań.



Rysunek 1.1: Sterownik PLC róznych producentów

1.1 Parametry sterownika S7-200 z jednostka sterująca CPU-222

S7-200 to seria sterowników firmy Siemens przeznaczonych do sterowania małymi i średnimi maszynami. Zastępują one przekaźniki programowalne LOGO! w bardziej wymagających aplikacjach. Moduł podstawowy sterownika CPU może zostać rozszerzony poprzez zastosowanie modułów rozszerzeń: wejść i wyjść cyfrowych oraz analogowych, moduły komunikacji itp. Poniżej przedstawiono specyfikację sterownika S7-200 z jednostką sterującą CPU-222 który będzie wykorzystywany na zajęciach. Serie sterowników S7-200 można zaliczyć do sterowni-

Tabela 1.1: Parametry jednostki centralnej CPU222		
Zintegrowane wej./wyj. binarne	8 wejść/6 wyjść	
Binarne wej./wyj. max ilość	40/38/78	
Analogowe wej./wyj. max ilość	8/4/10	
Pamieć programu	4 kB	
Pamieć danych	2 kB	
Zabezpieczenia danych	Typ. 50 godz. za pomocą kondensatora	
Szybkie liczniki	$4 \ge 30 \text{ kHz}$	
Obsługiwane protokoły	PPI Master/Slave, MPI Slabe, Freeport	
Wymiary	90x80x62 mm	
Inne właściwości komunikacyjne	PROFIBUS-DP/ AS-Interface/Ethernet/Modem	
Zegar czasu rzeczywistego	Opcja	
Potencjometr analogowy	1 zintegrowany	
Interfejs komunikacyjny	$1 \ge RS-485$	



Rysunek 1.2: Sterownik PLC S7-200

ków kompaktowo-modułowych ze względu na wbudowane moduły wej./wyj. w jednostce CPU oraz możliwość rozszerzenia funkcjonalności sterownika przez moduły rozszerzeń. Wykorzy-stywany w laboratorium sterownik S7-200 posiada jednostkę CPU-222 DC/DC/DC. Skrót DC/DC/DC oznacza, że sterownik zasilany jest napięciem 24VDC, posiada wejścia 24 DC oraz wyjścia tranzystorowe 24 DC. W zależności od konkretnego modelu sterownik może być na przykład zasilany napięciem sieciowym 230 VAC oraz posiadać wyjścia przekaźnikowe - AC/DC/RLY. Na rysunku poniżej przedstawiony został rozkład poszczególnych elementów



Rysunek 1.3: Schemat wej./wyj. sterownika

jednostki centralnej sterownika PLC. Na panelu przednim sterownika znajduje się złącze DB9 komunikacji magistrali RS-485. Poprzez wykorzystanie interfejsu PPI/USB można programować sterownik z użyciem protokołu PPI. Ponadto panel czołowy zawiera diody statusu pracy sterownika oraz aktualnych stanów wejść i wyjść. Pod klapkami maskującymi znaleźć można przełącznik trybu pracy oraz złącza śrubowe z wyprowadzeniami sterownika. Sterownik może znajdować się w jednym z trzech trybów pracy:

- RUN sterownik cyklicznie wykonuje zdefiniowany program,
- STOP program nie jest wykonywany,
- w tej pozycji możliwa jest komunikacja sterownika z oprogramowaniem STEP7 Micro/WIN wraz z możliwością zatrzymywania i wznawiania pracy programu z poziomu komputera.



Rysunek 1.4: Jednostka centralna sterownika S7-200

1.2 Organizacja pamięci sterownika S7-200

Argumentami instrukcji programu sterownika PLC są głównie obszary pamięci urządzenia. Adres obszaru pamięci sterownika firmy Siemens z rodziny S7-200 składa się z identyfikatora typu pamięci, znacznika formatu zależnego od formy adresowania oraz numeru konkretnego obszaru pamięci. Poszczególne identyfikatory obszaru pamięci sterownika przedstawiono w poniższej tabeli.

Tahela	1.2.	Wybrane	ident	vfikatory	obszarów	namieci
Tabela	1.4.	wybrane	nuem	ymratory	ODSLatOW	pannęci

Ι	Wejścia cyfrowe
Q	Wyjścia cyfrowe
М	Znaczniki (markery)
V	Zmienne
Т	Liczniki
HC	Szybkie liczniki
AI	Wejścia analogowe
AQ	Wyjścia analogowe

1.3 Adresowanie obszarów pamięci sterowników PLC z rodziny S7

Adresowanie bitowe - umożliwia modyfikacje pojedynczego bitu w obrębie bajtu, słowa lub podwójnego słowa.

[obszar pamięci][numer bajtu].[numer bitu]

	Tabela 1.3: Adresowanie bitowe - przykłady
M100.2	drugi bit setnego bajtu obszaru pamięci znaczników
<i>I0.0</i>	zerowy bit zerowego bajtu pamięci wejść cyfrowych
Q1.7	siódmy bit pierwszego bajtu pamięci wyjść cyfrowych

W tabeli poniżej przedstawiono przykłady adresowania bitowego.

Adresowanie przez słowa, bajty i podwójne słowa - umożliwia modyfikacje całych bajtów (B) czyli 8 bitów na raz, całych słów (W) czyli 16 bitów i podwójnych słów (DW) czyli 32 bitów na raz.

[obszar pamięci][rozmiar].[numer]

Tabela poniżej zawiera przykłady adresowania umożliwiającego modyfikację wielu bitów jednocześnie. Dokonując adresowania poprzez bajty, słowa oraz podwójne słowa należy szczególna

	Tabela 1.4: Adresowanie bitowe - przykłady
MB100	setny bajtu obszaru pamięci znaczników
IWO	zerowe słowo pamięci wejść cyfrowych
QD1	pierwsze podwójne słowo pamięci wyjść cyfrowych

uwagę zwrócić na zależności występujące pomiędzy bajtem, słowem oraz podwójnym słowem. Poszczególne zależności przedstawiono na rysunku poniżej. Należy zwrócić uwagę na fakt, żę modyfikując obszar pamięci (wielkości słowa*MW100* zmianie ulegają dwa bajty *MB100* oraz *MB101*. Analogicznie dokonując modyfikacji podwójne słowa *MD100* zmianie ulegną bajty o adresach od *MB100* do *MB103*.



Rysunek 1.5: Organizacja bitów w bajcie, słowie i podwójnym słowie

2 Podstawowe instrukcje bitowe języka drabinkowego

Język drabinkowy LAD (*lader*) to jeden z graficznych języków programowania sterowników programowalnych PLC zdefiniowany w normie IEC-61131. Program w języku drabinkowym wykonuje się szczebel po szczeblu. Symbole umieszczane w gałęziach odpowiadają elementom z układów przekaźnikowych np. wejście sterownika odpowiada stykowi przekaźnika natomiast wyjście sterownika cewce. Podstawowe elementy schematu w języku drabinkowym przedstawiono poniżej. Styk realizuje operację mnożenia logicznego AND stanu z lewej strony styku

styk normalnie otwarty	negacja
\dashv \vdash	⊣™⊢
styk normalnie zamknięty	cewka
⊢/⊢	—()

Rysunek 2.1: Podstawowe elementy schematu LAD

oraz stanu zmiennej (w zależności od rodzaju styku z negacją lub bez) przypisanej do styku. Wynik tej operacji jest przekazywany do prawej strony styku. Cewka powoduje przypisanie stanu z jej lewej strony do zmiennej jej przypisanej np. bity wyjściowego lub markera. Na rysunku poniżej przedstawiono realizację prostych operacji logicznych z wykorzystaniem podstawowych elementów języka LAD.



Rysunek 2.2: Proste operacje logiczne w języku drabinkowym

3 Układy czasowe - timery

Jedną z najczęściej wykorzystywanych poza elementarnymi operacjami bitowymi grupą instrukcji w języku LAD są timery. Pozwalają one na wprowadzanie w programie opóźnień oraz różnego rodzaju zależności czasowych. Podstawowe timery w sterownikach serii S7-200 to:

- TON układ czasowy ustawiający stan swojego wyjścia po upływie zadanego czasu opóźnienia,
- TOF układ czasowy kasujący stan swojego wyjścia po upływie zadanego czasu opóźnienia.

Przykładowy program wykorzystujący timer typu TON przedstawiono na rysunku poniżej. Timer zacznie odliczać czas 100x100ms=10s, gdy użytkownik załączy wejście *I0.0*. Po upły-



Rysunek 3.1: Przykład wykorzystania timera TON

wie tego czasu ustawione zostanie wyjście **Q0.0**. Okres podstawowy timera (a wiec ten przez który mnożona jest wartość wpisana do PT) uzależniony jest od numeru timera - tabela poniżej.

Tabela 3.1: Okres podstawowy dla timer	ów TON i TOF
<i>T37–T63</i> , <i>T101–T255</i>	$100 \mathrm{ms}$
<i>T33-T36</i> , <i>T33-T36</i> , <i>T97-T100</i>	$10\mathrm{ms}$
<i>T32</i> , <i>T96</i>	$1 \mathrm{ms}$

4 Liczniki

Instrukcje dotyczące liczników są stosowane do zliczania wewnętrznych oraz zewnętrznych zdarzeń. Wyróżnić można 3 podstawowe typy liczników:

- CTU licznik zliczający w górę,
- CTD licznik zliczający w dół,
- CTUD licznik zliczający w górę i w dół.

Licznik CTU służy do zliczania impulsów (zbocze narastające) doprowadzonych do wejścia CU. Zakres licznika wynosi 32767 impulsów. Licznik może zostać zresetowany podaniem stanu wysokiego na wejście R. Przy zrównaniu się liczby zliczanych impulsów z wartością zadaną na wejście PV (*preset value*) licznik zostaje uaktywniony - bit wyjściowy licznika zostaje ustawiony w stan wysoki. W analogiczny sposób działają pozostałe dwa liczniki. Poza instrukcjami



Rysunek 4.1: Przykład wykorzystania licznika CTU

liczników przedstawionych powyżej w kategorii Intructions - > Counters znajdują się instrukcję odpowiedzialne za konfigurację oraz obsługę liczników o działaniu szybkim.

5 Tworzenie projektu w środowisku STEP7 Micro/WIN

Środowisko STEP 7-Micro/WIN służy do tworzenia oprogramowania dla sterowników PLC firmy Siemens serii S7-200. Okno główne programu przedstawiono na rysunku poniżej. W



Rysunek 5.1: Okno główne środowiska STEP 7 Micro/WIN

centralnej części okna znajduje się edytor w którym tworzy się kolejne szczeble programu. W drzewie projektu (po lewej stronie edytora programu), w zakładce *Instructions* znajdują się elementy wykorzystywane do budowy programu w języku LAD. Poza przedstawionymi wcześniej operacjami bitowymi oraz timerami znaleźć tu można szereg instrukcji porównujących oraz modyfikujących dane różnych typów. Instrukcje rozgałęziające oraz najczęściej używane operacje bitowe znajdują się powyżej edytora programu. Aby wgrać program do sterownika należy wybrać z menu głównego *File->Download*. Przed wgraniem programu do sterownika należy skonfigurować połączenie z urządzeniem. W tym celu z panelu *View* należy uruchomić okno *Communication* - rysunek poniżej. Jeżeli interfejs komunikacyjny został wcześniej

Communications	×
Address Local: 0 Remote: I I I I I I I I I I I I I I I I I I I	None> Address: 0 → O Double-Click to Refresh
Network Parameters Interface: <none> Protocol: Mode: Highest Station (HSA): Supports multiple masters</none>	
Transmission Rate Baud Rate: Search all baud rates	
Set PG/PC Interface	OK Cancel

Rysunek 5.2: Proste operacje logiczne w języku drabinkowym

poprawnie skonfigurowany po kliknięciu na *Double-Click to Refresh* pojawi się sterownik podłączony do komputera. W przypadku problemów należy sprawdzić poprawność konfiguracji w oknie *Set PG/PC Interface*. Więcej informacji znaleźć można w podręczniku użytkownika.

6 Zadania do wykonania

Zadanie 1 Napisać program w języku drabinkowym realizujący następujące operacje:

$$Q0.0 = I0.0 \cdot \overline{I0.1}$$
$$Q0.1 = \overline{Q0.0}$$
$$Q0.2 = I0.1 + I0.5$$
$$Q0.3 = (I0.2 + \overline{I0.6}) \cdot I0.4$$

Zadanie 2 Przy pomocy podstawowych elementów języka drabinkowego zrealizować układ sterowania rozruchem bezpośrednim silnika klatkowego. Schemat układu sterowania przedstawiono poniżej. Opisać działanie programu.



Zadanie 3 Zrealizuj dekoder naturalnego kodu binarnego na kod 1 z 4 z wykorzystaniem podstawowych elementów języka drabinkowego.

Zadanie 4 Zrealizuj z wykorzystaniem układów czasowych generator fali prostokątnej o wypełnieniu 50% oraz okresie 1s.

Zadanie 5 Zrealizuj układ sterowania oświetleniem klatki schodowej. Po wciśnięciu przycisku oświetlenie świeci przez 5s. Każde kolejne wciśniecie powoduje wydłużenie czasu świecenia o kolejne 5s.

Zadanie 6 Przedstawić i opisać działanie wybranych funkcji wykonujących operację na bajtach, słowach i podwójnych słowach.