

5. Sterowniki PLC – budowa i zasada działania

PLC (ang. Programmable Logic Controller) to wyspecjalizowane urządzenie mikroprocesorowe wyposażone w programowalną pamięć, realizujące zadanie sterowania poprzez cykliczne wykonanie programu umieszczonego w jego pamięci.

W latach 90. XXw. Międzynarodowa Komisja Elektrotechniki (IEC, ang. International Electrotechnical Commission) opracowała normę IEC 61131 poświęconą sterownikom PLC. Norma ta powstała w odpowiedzi na potrzebę wprowadzenia pewnych standardów zwłaszcza w zakresie metod programowania sterowników (ze względu na dużą różnorodność rozwiązań stosowanych przez producentów). Kolejne części normy zawierają ([15], [16], [17], [18]):

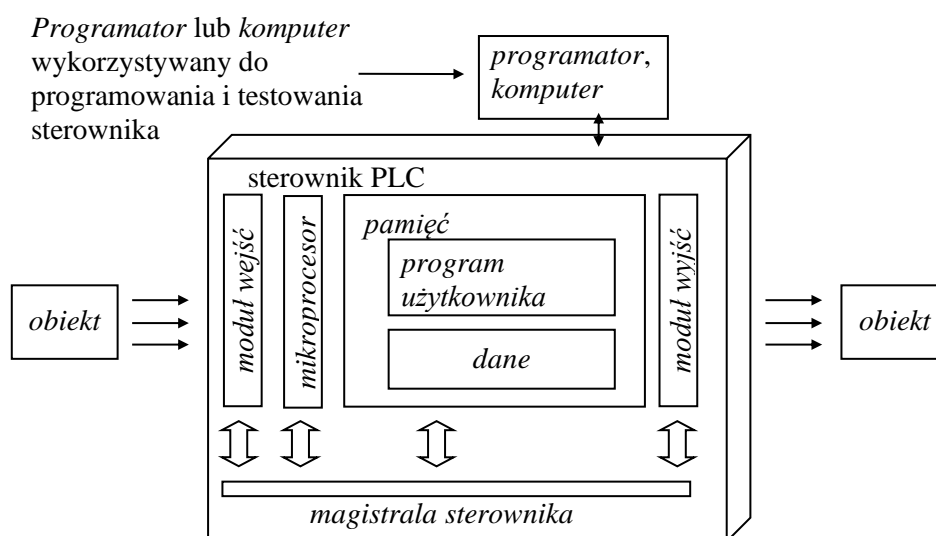
- część 1 – informacje ogólne: definicje pojęć i własności funkcjonalne PLC,
- część 2 – wymagania sprzętowe oraz badania, testy i warunki użytkowania,
- część 3 – metody programowania (omówione języki graficzne i tekstowe),
- część 4 – wytyczne wspomagające proces projektowania, eksploatacji i konserwacji,
- część 5 – zbiór zasad dotyczących komunikacji pomiędzy sterownikami różnych rodzin oraz pomiędzy sterownikami i innymi urządzeniami.

Zgodnie z normą sterownik programowalny to:

„cyfrowy system elektroniczny do stosowania w środowisku przemysłowym, który posługuje się pamięcią programowalną do przechowywania zorientowanych na użytkownika instrukcji w celu sterowania przez cyfrowe lub analogowe wejścia i wyjścia szeroką gamą maszyn i procesów”.

5.1. Budowa

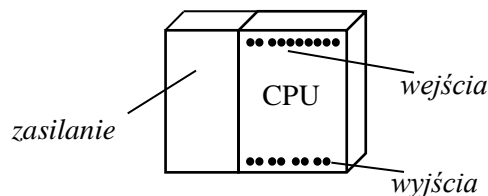
Już przedstawione powyżej definicje pozwalają na wyodrębnienie podstawowych modułów sterownika. Schemat ideowy architektury sterownika został przedstawiony na rys. 5.1.



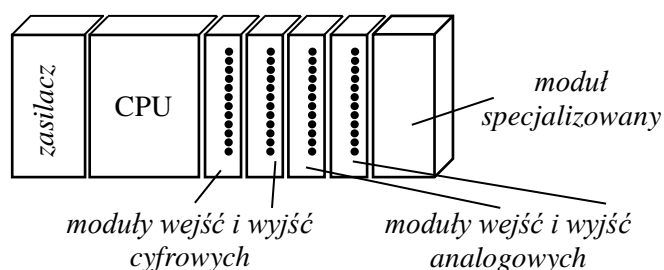
Rys. 5.1. Schemat ideowy sterownika PLC

Ze schematu przedstawionego na rys. 5.1 można wywnioskować, że moduły wejściowe i wyjściowe znajdują się w jednej obudowie z pozostałymi elementami sterownika. W rzeczywistości sterowniki PLC budowane są jako :

- sterowniki kompaktowe (rys. 5.2) – mają tzw. sztywną architekturę: w jednej obudowie umieszczone są zasilacz, jednostka centralna CPU (w sterownikach przyjmuje się, że CPU to mikroprocesor + pamięć), moduły wejść i wyjść (zwykle cyfrowych),
- sterowniki modułowe (rys. 5.3) – mają architekturę elastyczną: składają się z modułów, moduły montowane są na płytach (nazywanych kasetami) lub szynach montażowych,
- sterowniki kompaktowo-modułowe – sterowniki kompaktowe, które mogą być rozbudowywane.



Rys.5.2. Sterownik kompaktowy



Rys.5.3. Sterownik modułowy

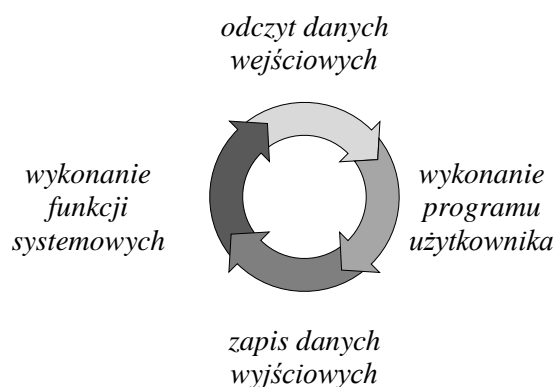
5.2. Zasada działania

Ogólnie zasadę działania sterownika można opisać w sposób następujący:

- sygnały z czujników, urządzeń pomiarowych kontrolowanego obiektu (procesu) zamieniane są w *modułach wejściowych* na sygnały cyfrowe akceptowane przez sterownik i zapamiętywane w obszarze *pamięci* nazywanej *pamięcią danych wejściowych*,
- *mikroprocesor* przetwarza program użytkownika, obliczając na podstawie danych wejściowych wartości sygnałów wyjściowych, wartości te zapamiętywane są w *pamięci danych wyjściowych*,
- *moduły wyjściowe*, korzystając z danych wyjściowych generują sygnały dla urządzeń wykonawczych.

Działania te wykonywane są w sposób *cykliczny* i tworzą tzw. *cykl programowy* sterownika. Upraszczając, w cyklu sterownika można wyróżnić cztery podstawowe fazy (zob. rys. 5.4):

- 1) *odczyt danych wejściowych*
dane wejściowe z modułów wejściowych wprowadzane są do pamięci sterownika,
- 2) *wykonanie programu użytkownika*
realizowany jest jeden przebieg programu,
- 3) *zapis danych wyjściowych*
dane wyjściowe wysyłane są do modułów wyjściowych,
- 4) *wykonanie funkcji systemu operacyjnego sterownika*
wykonywane są pozostałe czynności niezbędne do funkcjonowania sterownika: kontrola konfiguracji, diagnostyka, zarządzanie pamięcią i wykonaniem programu, komunikacja z urządzeniami zewnętrznymi.



Rys.5.4. Uproszczony schemat cyklu programowego sterownika PLC

Funkcje systemowe wykonywane są przed, po i w trakcie realizacji programu użytkownika. Do czynności wykonywanych na starcie należą np.: obliczanie czasu cyklu, uaktualnianie tablicy błędów i zerowanie zegara *watchdog*. Po zakończeniu programu i zapisie danych wyjściowych przetwarzane są żądania nadesłane przez programator i moduły komunikacyjne realizujące dwukierunkową komunikację z innymi urządzeniami w sieci. Na końcu wykonywane są również czynności diagnostyczne: przeprowadzana jest kontrola poprawności działania sterownika i programu użytkownika. W trakcie realizacji programu użytkownika monitorowany jest np. aktualny czas cyklu.

Cykl programowy sterownika może być rozpisany dokładniej, np. w podręczniku sterowników GE Fanuc Automation serii 90-20, 90-30, Micro PLC [3] w cyklu programowym wyodrębnianych jest 7 faz cyklu:

- 1) *porządkowanie cyklu* (obliczanie czasu trwania cyklu, planowanie rozpoczęcia następnego cyklu, określenie trybu pracy sterownika w następnym cyklu, uaktualnianie tablicy błędów i zerowanie zegara *watchdog*),
- 2) *odczyt danych wejściowych*,
- 3) *wykonanie programu użytkownika*,
- 4) *zapis danych wyjściowych*,
- 5) *komunikacja z programatorem*,
- 6) *komunikacja systemowa*,
- 7) *diagnostyka*.

Podręcznik do sterownika SIMATIC S7-1200 firmy Siemens [19] wymienia 5 podstawowych faz cyklu:

- 1) *zapis danych wyjściowych* (zgodnie z dokumentacją, w tej fazie jednostka centralna sterownika CPU przepisuje stan wyjść z obszaru pamięci sterownika do wyjść fizycznych),
- 2) *odczyt danych wejściowych* (CPU odczytuje fizyczne wejścia i zapamiętuje ich wartości w obszarze pamięci sterownika),
- 3) *wykonanie programu użytkownika* (CPU wykonuje zadania użytkownika, aktualizując wartości wyjściowe w pamięci, nie zmieniając stanu fizycznych wyjść),
- 4) *autodiagnostyka* (przeprowadzana jest kontrola systemu i sprawdzanie stanu modułów wejściowych i wyjściowych).

Dodatkowo w tle uruchomiona jest 5 faza, która obsługuje przerwania i komunikację z urządzeniami zewnętrznymi. Przerwania mogą być wywoływane np. zdarzeniami będącymi wynikiem błędów (diagnostycznego lub czasowego) czy zdarzeniami sprzętowymi.

Wykonanie części faz cyklu jest uzależnione od *trybu pracy* sterownika. Podstawowe tryby pracy to:

tryb **RUN** – tryb, w którym wykonywane są wszystkie fazy cyklu programowego,

tryb **STOP** – tryb zatrzymania, przeznaczony do programowania i konfigurowania sterownika (zazwyczaj w tym trybie pomijane są fazy odczytu danych wejściowych, wykonania programu i zapisu danych wyjściowych).

W sterownikach firmy GE Fanuc Automation (90-20, 90-30 i Micro) wykonanie faz odczytu i zapisu danych w trybie zatrzymania zależy od parametrów konfiguracyjnych (możliwy jest tryb pracy z nieaktywnymi wejściami i wyjściami oraz tryb z odczytywaniem wejść i ustawianiem wyjść). Podobnie, zmieniając odpowiedni parametr konfiguracyjny, można w trybie zatrzymania umożliwić zapis wyjść sterownika SIMATIC S7-1200.

Cykl programowy zwykle trwa od ułamków do kilkadziesiąt milisekund. W przypadku przekroczenia dopuszczalnego czasu trwania cyklu (zwykle od 100 ms do 500 ms) sterownik przechodzi z trybu RUN w tryb STOP, a błąd przekroczenia czasu cyklu rejestrowany jest w buforze diagnostycznym.

Szczegóły dotyczące przebiegu cyklu programowego, jak i czynności wykonywanych podczas pracy w różnych trybach omawiają instrukcje obsługi poszczególnych sterowników. W niektórych sterownikach wyodrębniane są dodatkowe tryby pracy. Sterownik SIMATIC S7-1200 może być uruchomiony w trybie STARTUP. W trybie tym kasowany jest obszar pamięci, w którym przechowywane są dane wejściowe, inicjalizowany jest obszar pamięci dla danych wyjściowych, następnie wykonywane są programy startowe, odczytywane są wejścia i ustawiane jest pozwolenie na zapis danych wyjściowych. Tryb STARTUP w sterownikach tego typu jest wymuszany podczas przejścia z trybu STOP do trybu RUN. Sterowniki mogą posiadać również dodatkowe tryby przeznaczone do testowania umożliwiające np. wykonanie jednego cyklu programowego, po którym sterownik automatycznie przechodzi w tryb STOP.