

Systemy automatyki firmy Allen-Bradley cz. 2

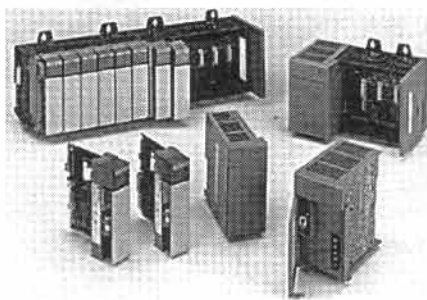
Sterowniki SLC-500

W artykule zaprezentowane zostały sterowniki programowalne rodziny SLC-500, ze szczególnym uwzględnieniem zastosowań w sterowaniu procesów ciągłych.

Omówiono w nim sposoby dołączania sterownika do obiektu, realizacji funkcji regulacyjnych wraz z zestawieniem oraz możliwości budowy systemów hierarchicznych i kaskadowych.

W poprzednim artykule mieliśmy możliwość zapoznania się z rodziną mikrosterowników MicroLogix 1000. Podstawowym ich zastosowaniem były różnorodne systemy dyskretnie, ograniczone jednak, z uwagi na budowę sterownika, do 32 punktów. Współczesne systemy automatyki wymagają od sprzętu znacznie większych możliwości: obsługi setek wejść/wyjść zarówno cyfrowych jak i analogowych, realizacji układów regulacji, obsługi przerw itp. Te właśnie cechy, oraz wiele innych posiadają sterowniki serii SLC-500, reprezentujące kolejny, po MicroLogix, poziom automatyzacji. Zanim przejdziemy do analizy typowych zastosowań warto zapoznać się z podstawowymi parametrami najpopularniejszych procesorów 5/03 i 5/04, przedstawionymi dla porównania razem z MicroLogix:

Dzięki modułowej konstrukcji sterownika (fot. 1) projektant ma możliwość elastycznego doboru konfiguracji systemu do włas-



nych potrzeb. Szeroka gama modułów zarówno analogowych ($\pm 20\text{mA}$, $\pm 10\text{V}$, termopary, RTD) jak i cyfrowych (TTL, 24VDC, 220VAC) pozwala na uniknięcie jakichkolwiek ograniczeń sprzętowych. Ale różnorodność modułów pozwala jedynie na realizację sprzęgu sterownika z dowolnym obiektem, udostępniającym swoje parametry za pomocą wymienionych sygnałów. Pozostałe funkcje: obróbkę danych oraz sterowanie musi już przejąć procesor sterownika. Wymagania stawiane nowoczesnym sterownikom, a takim jest SLC-500, są bardzo wysokie i nie ograniczają się tylko do prostej akwizycji pomiarów z procesu i generowania sygnałów sterujących.

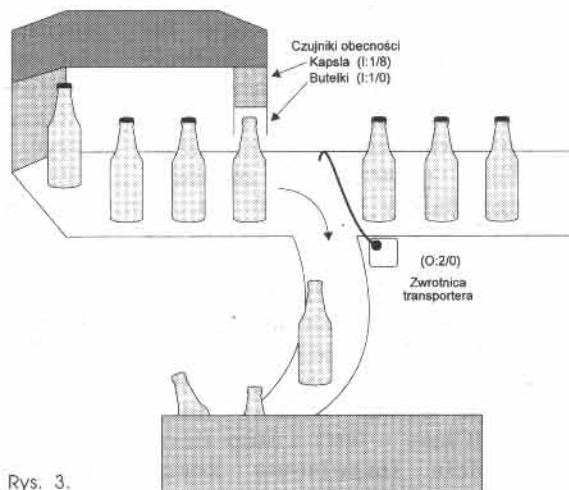
Jednym z typowych problemów jakie można napotkać w przemyśle jest sprawdzanie bardzo szybko przemieszczających się partii produktów, np. butelek. Poniższy przykład ilustruje rozwiązanie problemu testowania obecności kapsła na szklanej butelce. Każdy tego typu proces cechuje się dużą prędkością ruchu oraz nieciągłością występowania produktów na taśmie. Zatem konieczne jest stosowanie szybkich modułów cyfrowych, w celu eliminacji opóźnień sprzętowych oraz zastosowanie takiego sposobu odczytu wejść, by uniknąć sytuacji pominięcia butelki. O ile w wypadku elektroniki sprawa jest dosyć prosta - specjalne szybkie moduły ITB16 oraz ITV16 o czasie reakcji rzędu 0.1 ms pozwalają na spełnienie wymagań, to z odświeżaniem tablicy wejściowej sprawa jest trudniejsza. Rysunek 2 pokazuje typowy cykl pracy każdego procesora, w którym najistotniejszy jest fakt, że wejścia fizyczne są odczytywane tylko raz, w pierwszym bloku. Pozostałe bazują już na ustalonej informacji o stanie wejść, aż do następnego cyklu. W przypadku typowych, złożonych programów może to być czas nawet rzędu 10-20 ms. W wypadku linii butelkowania pokazanej na rysunku 3 sterownik kontroluje zaró-

Parametr	5/03	5/04	MicroLogix
Budowa	Modułowa	Modułowa	Scalona
Ilość cyfrowych punktów I/O	960	960	32
Czas wykonania programu (500 instr)	1 ms	0.9 ms	1.5 ms
Obsługa sygn. analogowych i PID	Tak (120 I/O)	Tak (120 I/O)	Nie
Komunikacja	DH-485, RS-232	DH+, DH-485, RS-232	Nie
Przerwania	Tak	Tak	Tylko czasowe

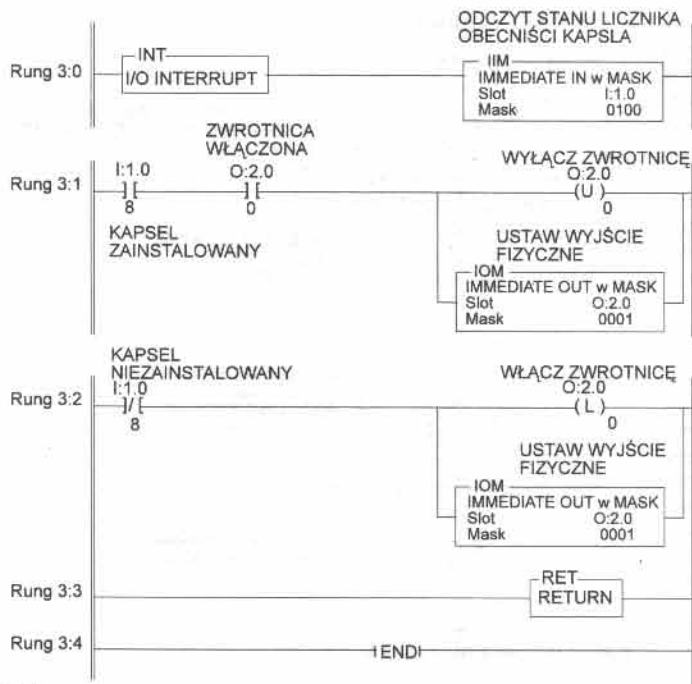
wno proces napełniania, jak i kapslowania oraz ostatecznego pakowania. Przy obecnym stopniu złożoności takich linii czas wykonania jednego cyklu może być podobny do podanego. Przy prędkości 5-10 butelek w ciągu sekundy i czasie trwania pierwszego bloku rzędu 0.1ms jest prawdopodobne, że czujnik I:1/O zasygnalizuje obecność butelki impulsem 5ms w trakcie trwania np. bloku programowego. Przy wymienionych czasach sterownik tego sygnału nie zauważy. Zatem należy tak zmienić cykl sterownika, by można było wykonywać obsługę testu kapsła w każdym momencie. Do tego właśnie celu służy mechanizm przerwań dyskretnych DII (ang. Discret Input Interrupt). Pozwala on na sprawdzanie co każde obrabiane słowo (zarówno rozkazowe jak i danych) stanu 8 wejść zadeklarowanego w statusie modułu. Wpisanie do statusu następującej sekwencji: Slot : 1, Maska : 00000001, Porównanie : 00000001, Plik 3, powoduje wykrywanie stanu wysokiego na wejściu 0 w slotie 1 i wykonywanie pliku 3. Plik ten, zwany procedurą obsługi przerwania DII, jest odpowiedzialny za wykonanie akcji po wykryciu butelki, czyli odzucenie (ustawienie bitu wyjściowego O:2/



Rys. 2.



Rys. 3.



Rys. 4.

0) gdy jest brak kapsła (stan 0 czujnika I:1/8) lub przepuszczenie (zerowanie O:2/0) gdy kapsel jest poprawny (1 sygnalizowana przez I:1/8). Wyjście O:2/0 jest połączone ze zwrotnicą transportera. Zawartość procedury DII przedstawia rysunek 4.

Obsługa szybkich procesów impulsowych, w których informacja o przebiegu jest generowana przez enkodery impulsowe, np. proces precyzyjnego cięcia taśmy stalowej, wymaga zastosowania specjalnego licznika sprzętowego HSCE. Cykl pracy takiej krajarki (rys 5) polega na: załączeniu sprzęgła i zadziałaniu silnika z transporterem, łagodnym rozpędzeniu, przejściu do normalnej prędkości, hamowaniu oraz cięciu które jest związane z odłączeniem sprzęgła i zadziałaniem hamulców taśmy. Stan procesu jest określany jako przedziały wartości zliczonych impulsów i na tej podstawie moduł ustawia swoje wyjścia połączone z silnikiem (szybko/wolno), sprzęgłem, hamulcem oraz mechanizmem noża. Moduł HSCE jest programowany za pomocą specjalnych tablic, w których zawarta jest informacja o sposobie pracy, a więc:

- zakresy wartości powodujące ustawienie wyjść,
- kombinacje wyjść odpowiadające podanym zakresom,
- sposób zliczania (w górę, w dół, tylko wejście A, A+B).

Ponieważ moduł ten posiada własny licznik i sam podejmuje odpowiednie decyzje to nie jest wymagane stosowanie mechanizmu DII, a jedynie konieczne jest ustawienie konfiguracji i jej ewentualna korekta w trakcie pracy krajarki. Przedstawione powyżej działania sterownika mogą zostać uzupełnione o przedstawione w poprzednim artykule rozkazy MicroLogix 1000.

Zastosowania SLC-500 w systemach dyskretnych to jedynie część bogatych możliwości tej serii. Kolejnym miejscem, w którym mogą być stosowane, są procesy ciągłe charakteryzujące się analogowym sprzężen-

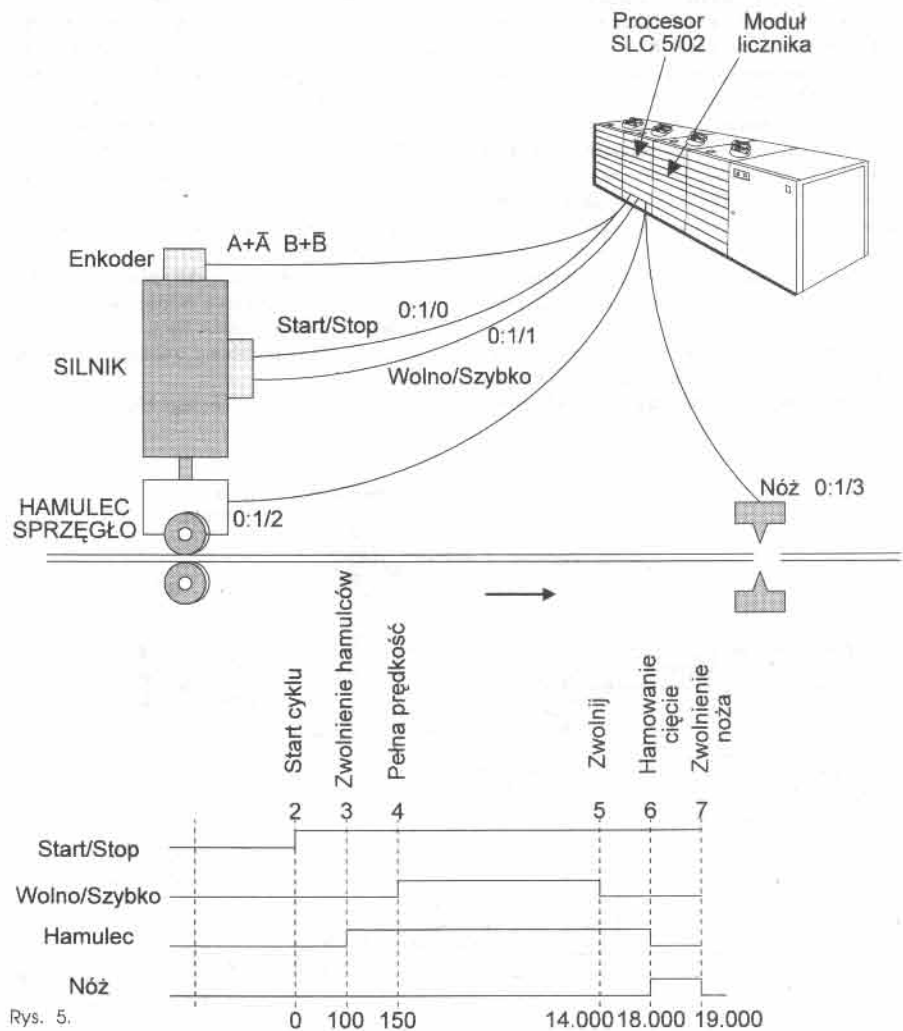
niem sterownika z obiektem. Od strony procesora linia technologiczna jest widziana jako licznik z przedziału 3277-16384, zaś sygnał wyjściowy 4-20mA jako licznik z przedziału 6242-31208. Regulatory PID implementowane w sterownikach SLC-500, bazują na danych z przedziału 0-16383, zatem ogólny schemat regulacji wraz z konwersją ma postać przedstawioną na rys.6, gdzie bloki SCL są rozkazami skalowania. Analogowe moduły obiektów przeznaczonych dla układów wykonawczych. Analogowe moduły obiektów przeznaczonych dla obsługi różnego typu sygnałów, co pociąga za sobą zastosowanie różnych przetworników A/

ność ich przeskalowania przed dalszą obróbką, gdyż np. sygnał wejściowy 4-20mA jest widziany jako licznik z przedziału 3277-16384, zaś sygnał wyjściowy 4-20mA jako licznik z przedziału 6242-31208. Regulatory PID implementowane w sterownikach SLC-500, bazują na danych z przedziału 0-16383, zatem ogólny schemat regulacji wraz z konwersją ma postać przedstawioną na rys.6, gdzie bloki SCL są rozkazami skalowania. Po dopasowaniu danych przychodzi konieczność zastosowania odpowiedniego regulatora. Standardowo implementowany jest typowy regulator PID, dla którego można ustawić następujące parametry:

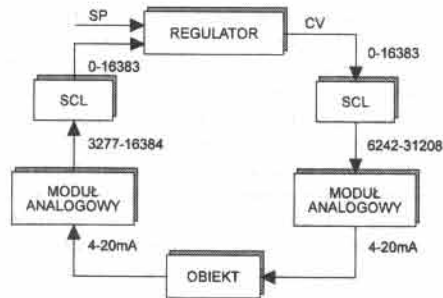
- wartości współczynników K, T_i, T_d ,
- strefę nieczułości na zmiany wartości procesowej PV wokół wartości zadanej SP,
- ograniczenia wartości wyjściowej CV, pozwalające na realizację nasyczeń wyjścia regulatora, a także sygnalizowanie stanów alarmowych,
- wartość okresu uaktywniania regulatora (ang. loop update) T_{LU} , która połączona z umieszczeniem regulatora w procedurze STI (uruchamianej co stały okres czasu), pozwala na precyzyjne określenie czasu pomiędzy kolejnymi aktywnościami.

Proces strojenia jest zależny od sterowanego obiektu; może odbywać się na drodze teoretycznej lub o ile jest to możliwe eksperymentalnej, polegającej na wzbudzeniu

C i C/A. Konsekwencją tych różnic są także inne zakresy zmian liczb, reprezentujących w tablicach wejściowej i wyjściowej poziom sygnału. Z faktem tym wiąże się koniecz-



Rys. 5.



Rys. 6.

obiektu przez zmianę wzmocnienia K aż do uzyskania oscylacji. Kolejno rejestruje się okres oscylacji, wzmocnienie K_0 i wyznacza pozostałe parametry zgodnie z zależnościami:

$$K = 0.5 * K_0; T_i = \dots; T_D = T_i / 8; T_{LU} = \dots / 10$$

Dzięki szerokiemu zakresowi zmian parametrów regulatora (czasy: 0.01 - 327.67 min, wzmocnienie 0.01-327.67) mogą one być stosowane zarówno w układach wymagających szybkiej reakcji, jak i procesów wolnych. Istotną zaletą sterowników regulatorów cyfrowych jest możliwość bardzo prostej realizacji układów kaskadowych - rys.7.

W takim układzie przy adresowaniu poszczególnych regulatorów wykorzystuje się odpowiednią wartość bloku poprzedniego. Uproszczony program realizujący sterowanie kaskadowe zbiornikiem z rysunku 7 przedstawia rysunek 8.

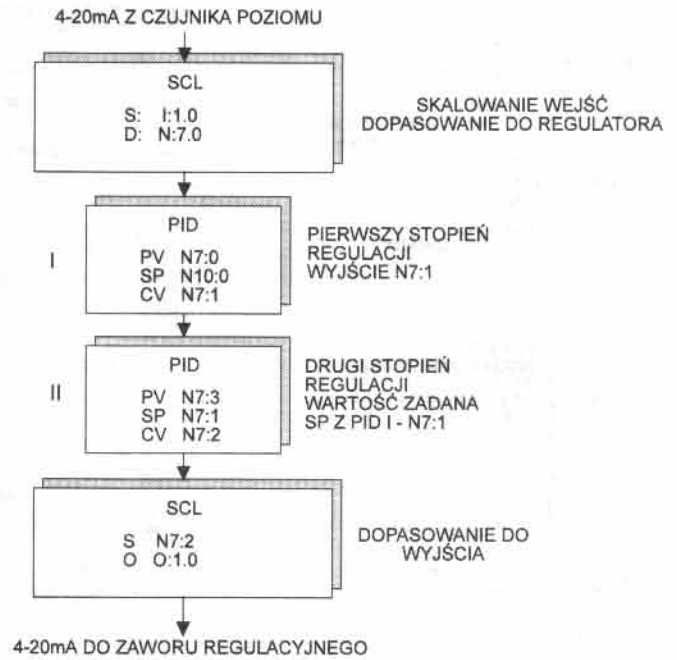
Przedstawione powyżej dwa rodzaje pracy znalazły szerokie zastosowanie zarówno w dużych systemach przemysłowych, takich jak rafinerie, zakłady chemiczne czy papiernie jak i w niewielkich oczyszczalniach ścieków. Wiele z istniejących aplikacji wymagało jednak nie tylko prostego regulatora. Konieczne było zapewnienie możliwości autokorekcji parametrów poprzez program sterownika lub poprawek prowadzonych przez operatora.

Każdy z regulatorów (ich liczba jest ograniczona jedynie pojemnością pamięci i ilością pętli analogowych) bazuje na 23 słowo-

wym bloku typu całkowego, w którym przechowuje wszystkie swoje parametry. Stąd wynika bezpośrednia możliwość dostępu do tych danych, zarówno przez program jak i operatora. Dzięki szerokiemu wachlarzowi instrukcji matematycznych oraz dostępności typu zmiennoprzecinkowego istnieje możliwość wpisania procedury dokonującej korekty na podstawie obserwacji procesu (np. średniej z n pomiarów, odchyień, wartości wskaźników jakości regulacji). Analogicznie wygląda procedura zmian przez operatora - dokonuje on jedynie modyfikacji odpowiednich rejestrów. Tu pojawia się pytanie - w jaki sposób zabezpieczyć system przed podaniem niewłaściwych parametrów. W wypadku terminali sprawa jest prosta: oprogramowanie narzędziowe ma możliwość określenia zakresów wpisywanych wartości. Ale przy złożonych programach trudno jest przewidzieć wystąpienie specyficznych wartości np. przepełnień. Stąd zaimplementowany został mechanizm obsługi błędów sterownika. Z całej liczby ok. 60 błędów 40% to błędy, które sterownik jest w stanie sam poprawić. Należą do nich:

- przekroczenia zakresów odpowiednich typów danych,
- błędne wartości (np. SP dla regulatora),
- przekroczenie zakresu tablic (w wypadku kolejek, sekwenserów itp.),
- różnego typu problemy związane z modułami.

Szerzej procedury obsługi błędów zostaną



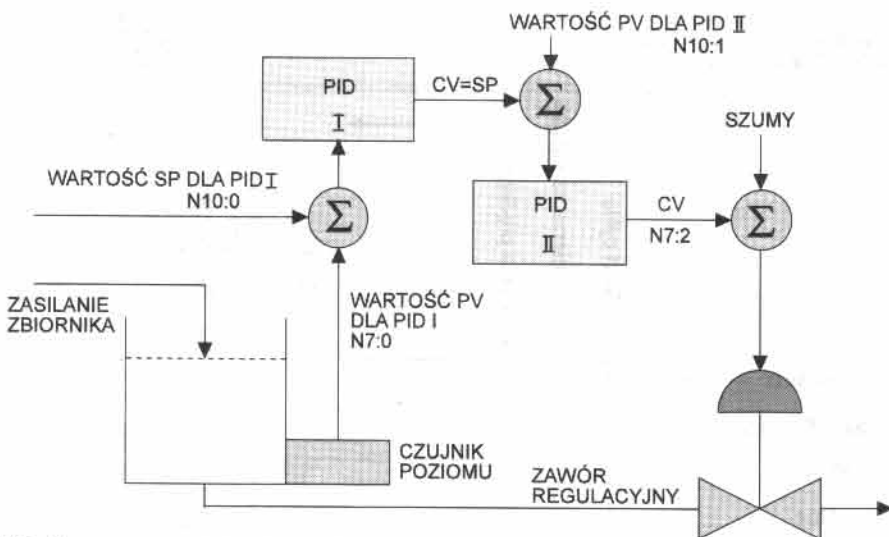
Rys. 8.

omówione w następnym artykule, dotyczącym sterowników PLC-5.

Przedstawione przykłady zastosowań pokazują, że rodzina sterowników SLC-500 jest w stanie zapewnić pełne sterowanie obiektów na małym i średnim poziomie automatyzacji. Jest oczywiste, że sterowniki muszą zostać obudowane systemami komunikacji pomiędzy sobą oraz operatorem na różnych poziomach. Jak zostanie to zaprezentowane w dalszej części cyklu, jest to również proste jak realizacja podstawowych funkcji sterujących.

Rafał Tutaj

Autor jest pracownikiem działu Allen-Bradley firmy Elmark



Rys. 7.