

mgr inż. Łukasz Dworzak
prof. dr hab. inż. Tadeusz Mikulczyński
Politechnika Wroclawska

SYNTEZA SEKWENCYJNYCH ALGORYTMÓW STEROWANIA ZAWIERAJĄCYCH KROKI CZASOWE

W pracy zaprezentowano zastosowanie metody Grafpol do syntezy sekwencyjnych algorytmów sterowania, których realizacja kroków jest zależna od warunków logicznych i czasowych. Omówiono zawarty w normie PN-EN 61131-3 licznik czasu typu TP. Skróceniwo przybliżono również zasady określania warunków zapisu i kasowania komórek pamięci oraz wyznaczania postaci funkcji zmiennych wyjściowych bez konieczności analizy przebiegu sygnałów wejściowych układu sterowania.

SYNTHESIS OF SEQUENTIAL CONTROL ALGORITHMS WITH TIME STEPS

The paper presents a method for the synthesis of sequential control algorithms using Grafpol method which realization of steps depends on the logical and temporal conditions. Discussed the type of timer TP contained in the PN-EN 61131-3. Briefly brought closer the rules for determining the conditions for setting and resetting memory cells and determining a function of output variables without having to analyse the course of the input signal of control system.

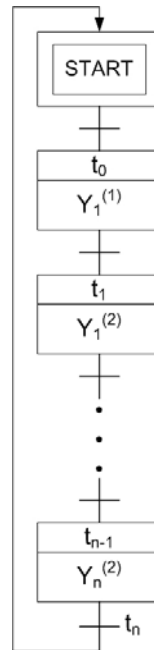
1. WSTĘP

W projektowanych algorytmach sterowania różnego rodzaju procedurami jedną z kluczowych roli odgrywają warunki, które decydują o realizacji kroków algorytmu sterowania. Najczęściej spotykanymi warunkami są warunki logiczne, jednak nie można zapominać również o spotykanych warunkach czasowych. Opracowana w ITMiA Politechniki Wroclawskiej metoda Grafpol [1] modelowania i programowania procedur sekwencyjnych, pozwala na proste i szybkie wyznaczenie równania schematu. Równanie to stanowi uniwersalną podstawę do implementacji opracowanego algorytmu sterowania w układzie sterowania (np. sterownik PLC). Zaletą metody Grafpol jest możliwość realizacji pamięci procedury sekwencyjnej bez konieczności analizowania przebiegu zewnętrznych sygnałów układu sterowania [2].

Zastosowanie metody Grafpol w odniesieniu do algorytmów, w których realizacja kroków określana jest przez warunki logiczne i czasowe, wymaga jednak jej drobnych modyfikacji. Szczegółową prezentację tych modyfikacji przedstawiono w niniejszej pracy.

2. REALIZACJA PAMIĘCI ORAZ WYZNACZANIE POSTACI ZMIENNYCH WYJŚCIOWYCH SEKWENCYJNYCH ALGORYTMÓW STEROWANIA

Zarówno pamięć, jak i postać zmiennych wyjściowych sekwencyjnych algorytmów sterowania, są określone w metodzie Grafpol na podstawie sieci Grafpol GS – sieci działania (rys. 1).



Rys. 1. Przykładowy algorytm sterowania reprezentowany za pomocą sieci Grafcop GS, $Y_i^{(1)}$ – zmienna wyjściowa sterująca ruchem i-tego napędu pneumatycznego z pozycji wyjściowej, $Y_i^{(2)}$ – zmienna wyjściowa sterująca ruchem i-tego napędu pneumatycznego do pozycji wyjściowej

Na podstawie tej sieci jest możliwa synteza programu użytkowego sterownika PLC zgodnie z następującymi zasadami:

Zasada 1

Zapis komórek pamięci następuje po zakończeniu wykonania kroków sterujących realizacją pierwszych etapów realizowanych przez poszczególne napędy pneumatyczne lub hydrauliczne. Stany te opisują tranzycje t_i , które sygnalizują zakończenie wykonania kroku S_i . W tranzycjach t_i nie uwzględnia się niezależnych od procesu warunków logicznych.

Zależności opisujące zapis elementarnych komórek pamięci mają następujące postaci:

$$\begin{aligned} M_1(S) &= t_1, \\ M_j(S) &= t_i \cdot m_{j-1} \cdot t_{i-1}, \quad \text{dla } j=2 \dots L, i=2 \dots n-1, \end{aligned} \quad (1)$$

gdzie:

M_1 – pierwsza elementarna komórka pamięci

M_j – j-ta elementarna komórka pamięci

m_j – sygnał wyjściowy j-tej elementarnej komórki pamięci.

Zasada 2

Kasowanie wszystkich elementarnych komórek pamięci następuje w ostatnim stanie algorytmu sterowania. Zależność opisująca kasowanie wszystkich elementarnych komórek pamięci ma postać:

$$M_{1..L}(R) = t_n \cdot m_L, \quad (2)$$

Zasada 3

Do wyznaczenia funkcji zmiennej wyjściowej algorytmu sterowania konieczne jest określenie zależności t_i^* , czyli tranzycji algorytmu sterowania z uwzględnionymi pamięciami. Tranzycja t_i^* określa rozpoczęcie i-tego kroku algorytmu sterowania. Stanowi ją iloczyn:

- warunku t_i , wynikającego z sieci Grafcop GS
- sygnału pamięci (m_j) lub zanegowanego sygnału pamięci (\bar{m}_j).

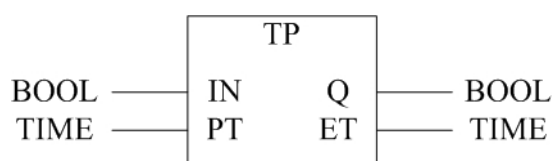
Użyta w tranzycji t_i^* postać sygnału pamięci M_j jest określana w następujący sposób:

- zanegowane sygnały pamięci \bar{M}_j występują we wszystkich tranzycjach poprzedzających tranzycję, w której nastąpił zapis pamięci M_j , do tranzycji t_0 lub tranzycji, w której zapisywana jest poprzednia elementarna komórka pamięci (M_{j-1}) włącznie
- niezanegowane sygnały pamięci M_j występują we wszystkich tranzycjach następujących po tranzycji, w której nastąpił zapis pamięci M_j włącznie, do tranzycji t_{n-1} lub tranzycji, w której następuje zapis kolejnej elementarnej komórki pamięci (M_{j+1}) włącznie.

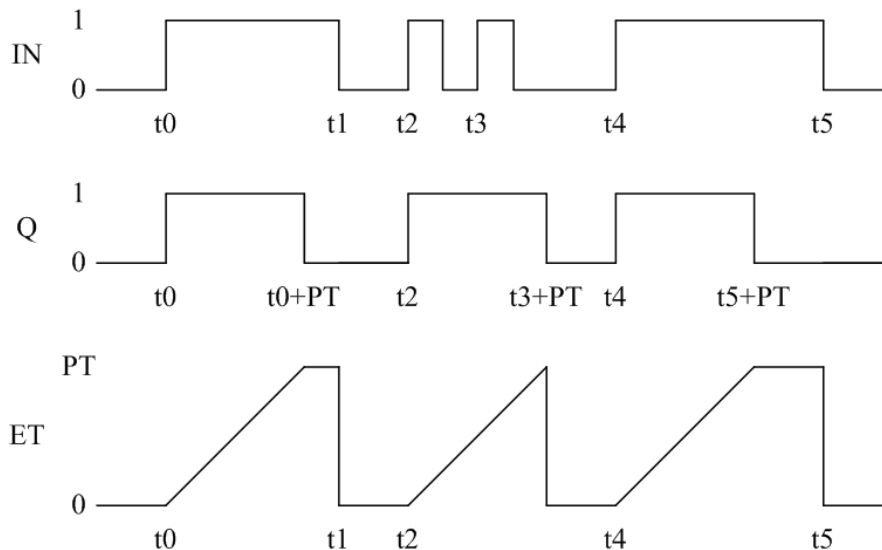
3. SYNTEZA SEKWENCYJNYCH ALGORYTMÓW STEROWANIA, W KTÓRYCH REALIZACJĘ KROKÓW OKREŚLAJĄ WARUNKI LOGICZNE I CZASOWE

3.1. Licznik czasu w normie PN-EN 61131-3:2004

Norma PN-EN 61131-3, określająca języki programowania sterowników PLC, definiuje trzy rodzaje liczników czasu. Jednym z nich jest licznik TP (*ang. Pulse Timer*), którego schemat przedstawiono na rys. 2. Licznik ten inicjowany jest przez impuls i działa w następujący sposób. Podanie na wejście IN sygnału o zboczu narastającym (zmieniającego wartość logiczną z 0 na 1) powoduje inicjalizację licznika, skutkiem czego zwiększaniu ulega wartość zmiennej skojarzonej z ET (*ang. Elapsed Time* – czas, który upłynął) licznika czasu. Z chwilą, gdy zajdzie zależność $ET \geq PT$ (*ang. Preset Time* – zadany czas) wyjście Q (*ang. Quit*) zmienia swoją wartość logiczną z 1 na 0. Przypisanie zmiennej skojarzonej z ET wartości 0, tzw. wyzerowanie licznika czasu, następuje w chwili gdy IN i Q mają wartość logiczną 0. Przykładowy diagram, ukazujący stan wyjścia (Q) licznika czasu w zależności od stanu jego wejścia (IN), przedstawiono na rys. 3.



Rys. 2. Schemat bloku licznika czasu TP [3]



Rys. 3. Diagram funkcjonalny licznika czasu TP [3]

3.2. Zasady zastosowania licznika czasu w metodzie Grafpol

W odniesieniu do procedur, których poprawna realizacja określana jest przez warunki logiczne i czasowe, metoda Grafpol charakteryzuje się kilkoma zmianami wynikającymi ze specyfiki kroków czasowych. Zmiany te zawarto w następujących zasadach:

Zasada 1

W krokach czasowych stosowany jest licznik czasu typu TP.

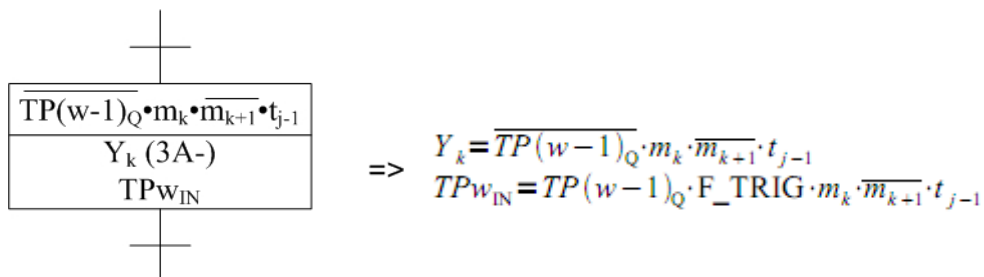
Zasada 2

Tranzycja występująca po kroku w którym nastąpiła inicjacja licznika czasu (timera) ma postać iloczynu zanegowanego sygnału wyjściowy tego licznika czasu i tranzycji poprzedzającej ten krok.

$$t_j = \overline{TPk_Q} \cdot t_{j-1} \tag{3}$$

Zasada 3

Jeśli tranzycja poprzedzająca krok w którym następuje inicjacja w-tego licznika czasu (timera) składa się zanegowanego sygnału (w-1)-tego licznika czasu to w-ty licznik czasowy inicjowany jest przez zbocze opadające (w-1)-tego licznika czasu.



Rys. 4. Ilustracja zasady 3

Zasada 4

Jeśli z sieci Grafpol GS wynika, że komórka pamięci powinna być zapisana przez zanegowany sygnał w-tego licznika czasu, to należy ją zapisać przez zbocze opadające

wyjścia w-tego licznika czasu (4) -zgodnie z PN-EN 61131-3 stosujemy funkcję F_TRIG.

$$\begin{aligned} \text{ŹLE: } M_k(S) &= \overline{TP_{wQ}} \cdot M_{k-1} \cdot \overline{TP^{(w-1)}_Q} \\ \text{DOBRZE: } M_k(S) &= TP_{wQ} \cdot F_TRIG \cdot M_{k-1} \cdot \overline{TP^{(w-1)}_Q} \end{aligned} \quad (4)$$

3.3. Przykład zastosowania

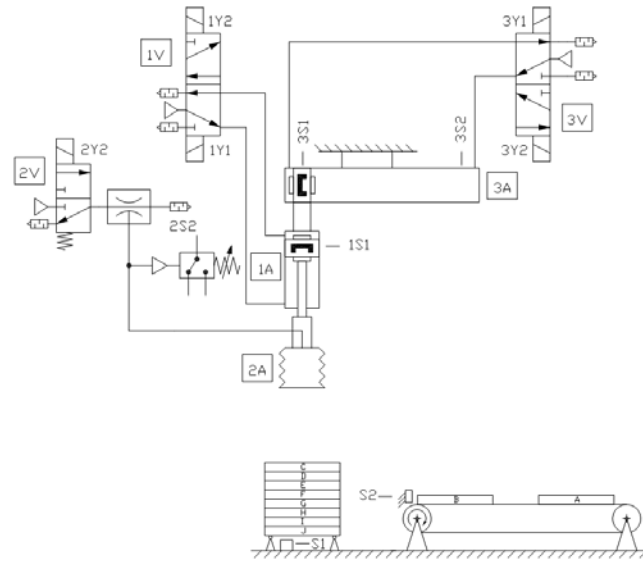
Na rys. 5 przedstawiono, w stanie początkowym, przykładowy schemat procesu do sterowania którym konieczne jest użycie licznika czasu. Proces realizowany jest przez urządzenie, które ma za zadanie podawanie blach ze stosu na przenośnik taśmowy. Ze względu na zmienną odległość pomiędzy chwytakiem podciśnieniowym a blachą znajdującą się na szczycie stosu, tłoczysko siłownika 1A wysuwa się w każdym cyklu na różną odległość. Dlatego wartość wysuwu określana jest, w algorytmie sterowania, przez czas wysuwu, równy czasowi potrzebnemu do maksymalnego wysuwu. Analogiczna sytuacja występuje przy odkładaniu blach na podajnik taśmowy, ze względu na wymagany różny wysuw tłoczyska w zależności od grubości blachy. Proces podawania blach uzależniony jest od ich obecności na stosie (czujnik S1) oraz możliwości odłożenia na przenośnik taśmowy (czujnik S2).

Sposób funkcjonowania urządzenia można zapisać w postaci opisu słownego poszczególnych etapów:

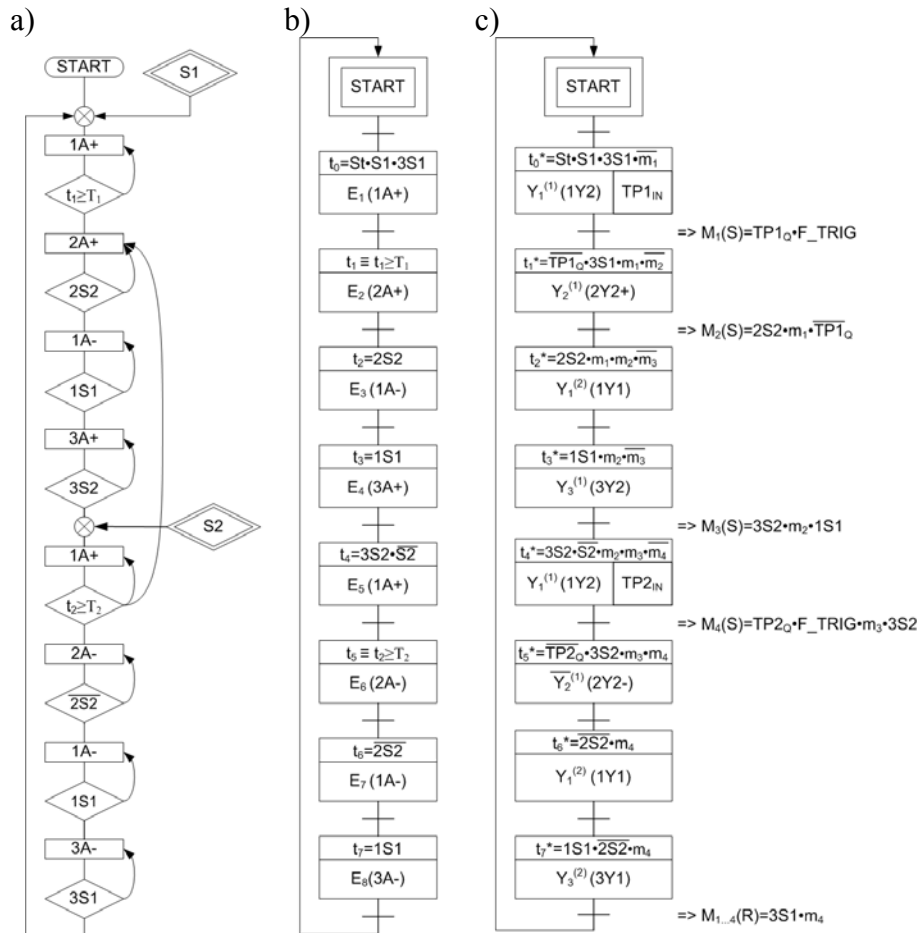
ETAP E1: *opuszczenie chwytaka*	ETAP E5 *opuszczenie chwytaka*
REALIZACJA: 1A+ (1Y2)	REALIZACJA: 1A+ (1Y2)
SYGNALIZACJA: $t_1 \geq T_1$	SYGNALIZACJA $t_2 \geq T_2$
ETAP E2: *chwycenie blachy*	ETAP E6: *zwolnienie blachy*
REALIZACJA: 2A+ (2Y2+)	REALIZACJA: 2A- (2Y2-),
SYGNALIZACJA: 2S2	SYGNALIZACJA: 2S2,
ETAP E3: *uniesienie chwytaka*	ETAP E7: *uniesienie chwytaka*
REALIZACJA: 1A- (1Y1)	REALIZACJA: 1A- (1Y1)
SYGNALIZACJA: 1S1	SYGNALIZACJA: 1S1
ETAP E4: *przesuw chwytaka do taśmociągu*	ETAP E8: *przesuw chwytaka od taśmociągu*
REALIZACJA: 3A+	REALIZACJA: 3A-
SYGNALIZACJA: 3Y2	SYGNALIZACJA: 3Y1

W powyższym opisie słownym etapy E1 i E5 są na poziomie algorytmu sterowania realizowane przez kroki czasowe. W algorytmie procesu t_1 to czas odmierzony przez licznik czasu nr 1, a T_1 to czas zadany licznika czasu nr 1.

Na podstawie schematu funkcjonalnego i opisu słownego stworzono algorytm procesu, który przedstawiono na rys. 6a i 6b. Na rys. 6c zaprezentowano, za pomocą sieci Grafcop GS, algorytm sterowania uzupełniony o zrealizowaną pamięć, wraz z warunkami jej zapisu i kasowania. Algorytm ten opracowano stosując wyżej przedstawione zasady. Jego algebraiczne ujęcie stanowi równanie schematowe (5). Równanie to stanowi podstawę do implementacji w sterowniku za pomocą dowolnego języka programowania opisanego w normie PN-EN 61131-3. Jego aplikacyjną reprezentację, za pomocą języka FBD, przedstawiono na rys. 7.

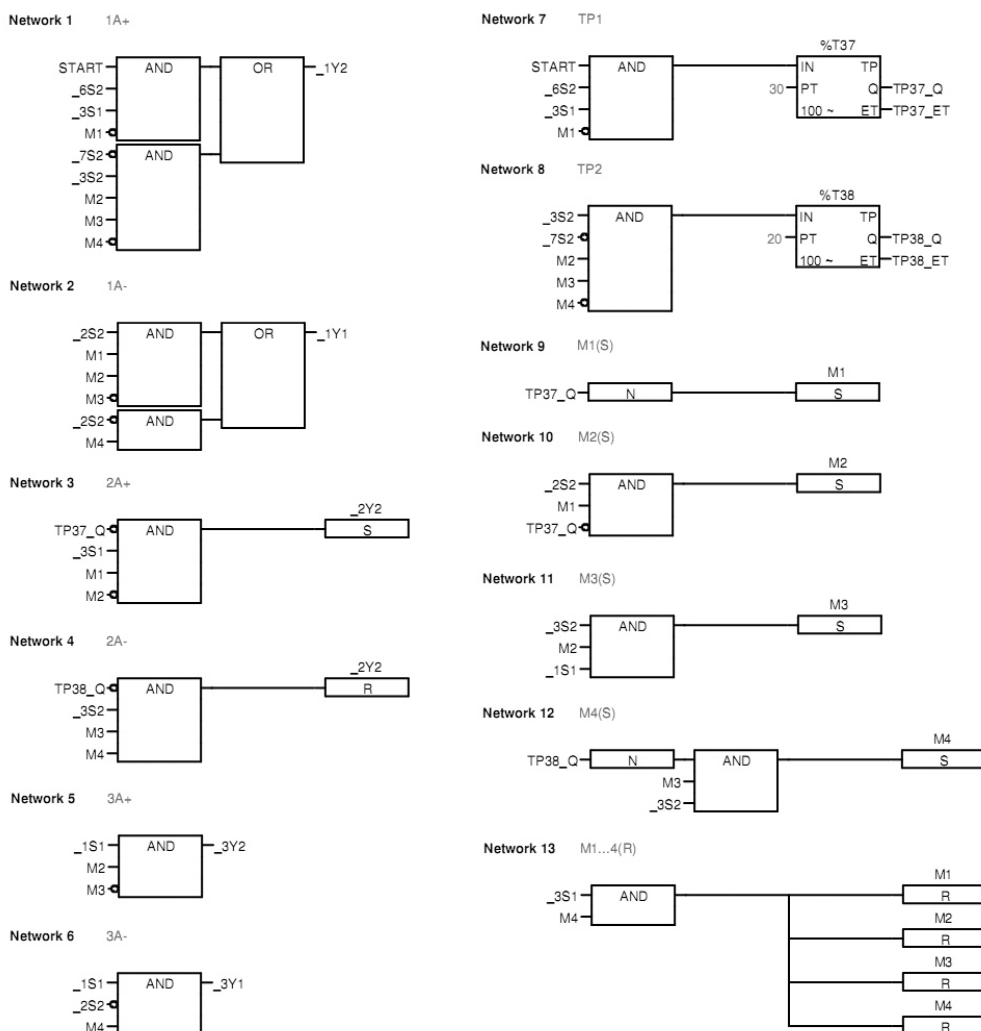


Rys. 5. Schemat funkcjonalny przykładowej aplikacji w której realizację kroków określają warunki logiczne i czasowe



Rys. 6. Algorytm procesu zapisany za pomocą: a) sieci operacyjnej, b) sieci Grafcop GP i stworzony na jego podstawie c) algorytm sterowania z określonymi warunkami zapisu i kasowania komórek pamięci oraz określonymi postaciami zmiennych wyjściowych Y_i

$$F(Y, M) = \sum \left[\begin{aligned}
 &St \cdot S1 \cdot 3S1 \cdot \overline{m_1} \cdot (Y_1^{(1)} + TP1_{IN}), \\
 &\overline{TP1_Q} \cdot 3S1 \cdot m_1 \cdot \overline{m_2} \cdot Y_2^{(1)}(S), \\
 &2S2 \cdot m_1 \cdot m_2 \cdot \overline{m_3} \cdot Y_1^{(2)}, \\
 &1S1 \cdot m_2 \cdot \overline{m_3} \cdot Y_3^{(1)}, \\
 &3S2 \cdot \overline{S2} \cdot m_2 \cdot m_3 \cdot \overline{m_4} \cdot (Y_1^{(1)} + TP2_{IN}), \\
 &\overline{TP2_Q} \cdot 3S2 \cdot m_3 \cdot m_4 \cdot Y_2^{(1)}(R), \\
 &\overline{2S2} \cdot m_4 \cdot Y_1^{(2)}, \\
 &1S1 \cdot \overline{2S2} \cdot m_4 \cdot Y_3^{(2)}, \\
 &TP1_Q \cdot F_TRIG \cdot M_1(S), \\
 &2S2 \cdot m_1 \cdot TP1_Q \cdot M_2(S), \\
 &3S2 \cdot m_2 \cdot 1S1 \cdot M_3(S), \\
 &TP2_Q \cdot F_TRIG \cdot m_3 \cdot 3S2 \cdot M_4(S), \\
 &3S1 \cdot m_4 \cdot [M_1(R) + M_2(R) + M_3(R) + M_4(R)]
 \end{aligned} \right] \quad (5)$$



Rys. 7. Zapis programu użytkowego sterownika PLC (Siemens S7-200) za pomocą języka FBD

4. PODSUMOWANIE

Opracowane zasady syntezy sekwencyjnych algorytmów sterowania, w których realizację kroków określają warunki logiczne i czasowe to kolejny krok na drodze rozwoju metody Grafpol, bez konieczności analizowania przebiegu sygnałów wejściowych i wyjściowych układu sterowania. Dzięki temu zwiększa się uniwersalność jej stosowania, a możliwość zapisu opracowanych algorytmów sterowania w dowolnym języku programowania zdefiniowanym w normie PN-EN 61131-3 gwarantuje zgodność z większością sterowników PLC.

5. LITERATURA

- [1] Mikulczyński T.: *Automatyzacja procesów produkcyjnych*, WNT, Warszawa 2006.
- [2] Dworzak Ł., Ciskowski S., Mikulczyński T.: *Synteza metodą Grafpol sekwencyjnych algorytmów sterowania*. Pomiary Automatyka Robotyka nr 2/2009.
- [3] Norma PN-EN 61131-3:2004 - Sterowniki programowalne - Część 3: Języki programowania.