

## Modernizacja układu sterowania elastyczną linią obróbkową

*Streszczenie: W referacie przedstawiono przykład zastosowania oryginalnej koncepcji otwartego układu sterowania produkcją zautomatyzowaną do modernizacji układu sterowania systemem TOR. Omówiono zakres wykonanych prac, napotkane problemy i uzyskane wyniki. Zastosowana metoda sterowania produkcją oparta jest na macierzowym modelu realizowanego procesu wytwarzania.*

### Modernisation of flexible machining line control system

*Summary: The paper presents the exemplary application of original open control system concept. Example concerns the modernisation of machining line TOR control system. The range of accomplished tasks, appearing problems and achieved results are described. Applied method of production control is based on matrix model of manufacturing process.*

#### 1. WPROWADZENIE

W referacie przedstawiono przykład zastosowania oryginalnej koncepcji otwartego układu sterowania zautomatyzowanym wytwarzaniem, opartej na macierzowym modelu realizowanego procesu [3], do modernizacji układu sterowania systemem TOR, w którym wykorzystano urządzenie Centrum Produkcyjnego TOR1 (CPTOR1) zainstalowanego w Instytucie Technologii Maszyn i Automatykacji Produkcji Politechniki Krakowskiej.

CPTOR1 zaprojektowano i zbudowano w końcu lat 70-tych w Centrum Badawczo-Konstrukcyjnym Obrabiarek w Pruszkowie. Z powodu braku zainteresowania przemysłu i zawodności układu sterowania, centrum to pracowało krótko. Potem system przekazano Politechnice Krakowskiej i zainstalowano w laboratorium ITMiAP, gdzie był on wykorzystywany w dydaktyce, praktycznie przez cały czas pozostając nieruchomym eksponatem.

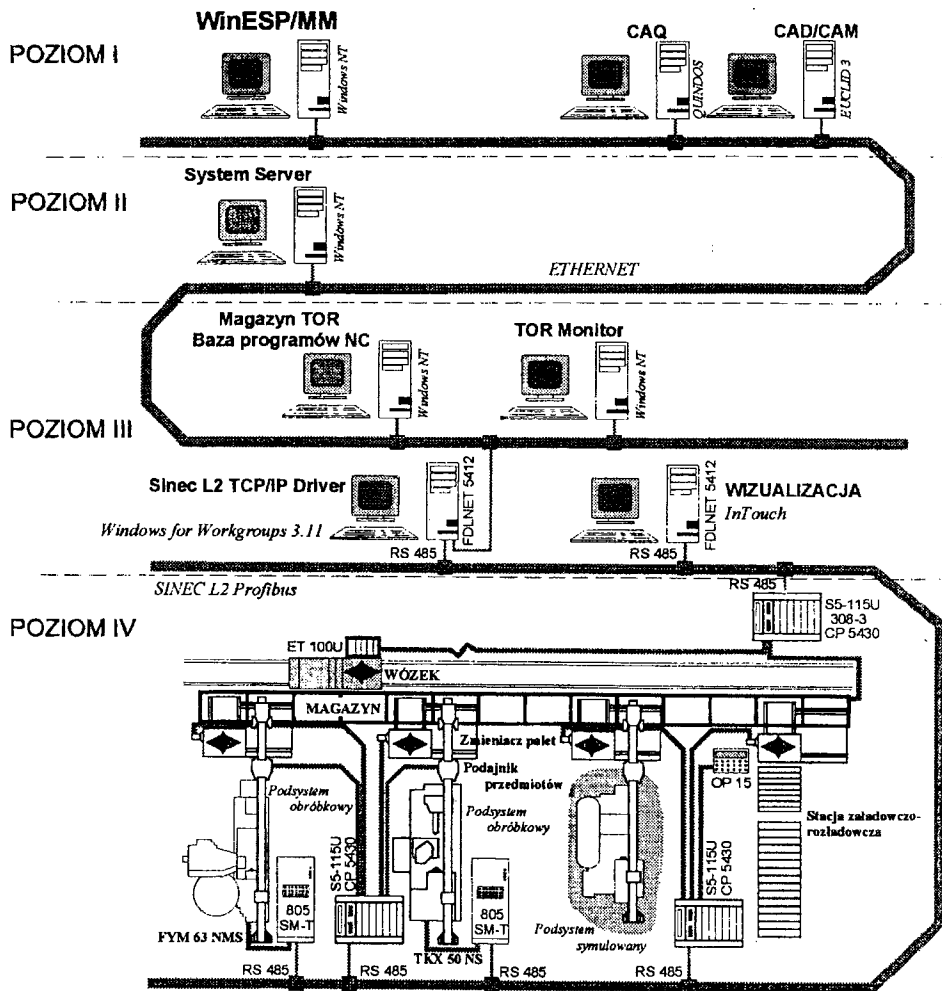
Centrum Produkcyjne TOR1 było elastyczną linią obróbkową do obróbki wałków o średnicach 25-80 mm, długości 100-800 mm i masie do 25 kg. Główne moduły CPTOR1, to:

- stacja załadowczo-rozładowcza (SZR);
- centrum obróbkowe poziome WCF-40NMS, tokarka TKX50NS, centrum obróbkowe pionowe FYM63NMS, wraz ze zmieniaczami palet i podajnikami przedmiotów;
- komplet palet z chwytakami sprężynowymi do wałków (1, 2 lub trzy rzędy, po 8 w każdym rzędzie);
- magazyn regałowy palet i wózek szynowy przemieszczający palety;
- układ sterowania.

Moduły posiadały własne (lokalne) układy sterowania, w których zastosowano:

- sterowniki NUMS 320 i NUMS 331 wraz z UDS oraz regulatorami napędów firmy GETYS wykonanymi w technice TTL;

- układy sterowania cyklami pracy: zmiennicy palet i podajników przedmiotów wykonane w technice LOGISTER;
- układ sterowania wózkiem wykonany w technice LOGISTER;
- układy wejściowe z przycisków, krańcówek, zderzaków itp. oraz wyjściowo-wykonawcze (TTL, LOGISTER).



Rys. 1. Układ sterowania systemem TOR

W systemie zastosowano sterowanie dyspozytorskie, w którym wykorzystano ponadto:

- pulpit dyspozytorski, umożliwiający podawanie rozkazów dla wózka i zmiennicy palet przy stacjach, komunikowanie się dyspozytora z operatorami stacji oraz przedstawianie stanu magazynu, wykonany w technice TTL;
- tzw. monitory, umieszczone na poszczególnych stacjach, służące do komunikowania się operatora z dyspozytorem, wykonane w technice TTL;
- układy dopasowujące transmitowane sygnały elektryczne;
- czujniki obecności palety w regałach magazynu.

Prace prowadzone w latach 1992–1997 w Katedrze Systemów Wytwarzania PK doprowadziły do gruntownego zmodernizowania układu sterowania systemem TOR. Jego obecną strukturę przedstawia rys. 1. Należy podkreślić, że całość przedsięwzięcia wykonano we własnym zakresie. Zostaną krótko omówione wyniki najważniejszych jego etapów.

## 2. MACIERZOWY MODEL PROCESU WYTWARZANIA W SYSTEMIE TOR

Opracowanie modelu macierzowego systemu TOR wymagało dyskretyzacji procesu wytwarzania, tzn. wyodrębnienia obiektów systemu oraz czynności elementarnych przez nie realizowanych. Obiekty, to: stacja załadowczo-rozładowcza (SZR), w której realizowane jest ładowanie półfabrykatów na palety i rozładowywanie gotowych wyrobów, frezarka – nakiełczarka WCF40NMS (WCF- obiekt symulowany), stacja tokarska TKX50NS (TKX) oraz stacja frezarska FYM63NMS (FYM). Każda z wymienionych stacji współpracuje ze zmieniającym palet (odpowiednio  $ZP\ 1 \div ZP\ 4$ ), który pobiera palety z modułu pobrania w magazynie (odpowiednio  $IN\ 1 \div IN\ 4$ ) i dostarcza do stacji obróbkowej, a następnie zwraca palety po obróbce do modułu oddania (odpowiednio  $OUT\ 1 \div OUT\ 4$ ). Wózek szynowy ( $W$ ) przemieszcza palety pomiędzy stacjami. Przyjęto, że w systemie będą produkowane trzy rodzaje przedmiotów:  $PO1$ ,  $PO2$  i  $PO3$ , przy czym pierwsze dwa są transportowane (po 4 sztuki) na paletach jednorzędowych ( $PAL1$ ), a trzeci – na paletach dwurzędowych ( $PAL2$ ), po 8 sztuk na każdej. Palety przechowywane są w magazynie regałowym ( $M$ ). Łącznie w modelu systemu TOR wyodrębniono 23 obiekty i 127 czynności elementarnych. Dla każdej z czynności zdefiniowane zostały tzw. zbiory następników, przypisane obiektom w niej uczestniczącym, tzn. czynności, do których gotowy jest dany obiekt po zakończeniu czynności rozpatrywanej (szczegółowe informacje o modelu macierzowym można znaleźć w monografii [1]). Tabela 1 zawiera przykłady zbiorów następników dla czynności związanych z obróbką przedmiotu  $PO3$  na stacji frezarskiej  $FYM$  i wybranych obiektów. Dla wyjaśnienia: czynność  $PAL2+PO3/3j$ ,  $M \rightarrow W$  oznacza pobranie palety  $PAL2$  z przedmiotami typu  $PO3$  w stanie 3 z magazynu  $M$  na wózek  $W$ ; czynność  $PAL2+PO3/3j$ ,  $FYM+ZP\ 4 *$  oznacza obróbkę tych przedmiotów na stacji frezarskiej  $FYM$  z udziałem zmieniaacza palet  $ZP\ 4$ .

Tabela 1. Zbiory następników dla wybranych czynności w modelu systemu TOR

Czynności ↓	Obiekty →	W	IN4	OUT4	ZP4	FYM	PAL2	PO3
		2	6	10	14	18	20	23
$PAL2+PO3/3j$ , $OUT3 \rightarrow W$	103	104, 105					104, 105	104, 105
$PAL2+PO3/3j$ , $W \rightarrow M$	104	3					106	106
$PAL2+PO3/3j$ , $W \rightarrow IN4$	105	3	107				107	107
$PAL2+PO3/3j$ , $M \rightarrow W$	106	105					105	105
$PAL2+PO3/3j$ , $IN4 \rightarrow ZP4$	107		7		108		108	108
$PAL2+PO3/3j$ , $FYM+ZP\ 4 *$	108				109	19	109	109
$PAL2+PO3/4j$ , $ZP4 \rightarrow OUT4$	109			110	15		110	110
$PAL2+PO3/4j$ , $OUT4 \rightarrow W$	110	111, 112		11			111, 112	111, 112
$PAL2+PO3/4j$ , $W \rightarrow M$	111	3					113	113

W opracowanym modelu, zadania każdej ze stacji podzielono na trzy czynności elementarne: zabranie palety z modułu pobrania na zmieniaacz palet, obróbkę przedmiotów dostarczonych na palecie (w przypadku stacji SZR – załadowanie lub rozładowanie palety) z udziałem właściwego zmieniaacza palet oraz zwrot palety ze zmieniaacza palet do modułu oddania. Wskutek tego, potencjalnie jest możliwe, że w stacji znajdują się w jednej chwili trzy palety: pierwsza oczekująca na obróbkę, druga w trakcie obróbki, trzecia – gotowa – oczekująca na zabranie jej przez wózek. Umożliwia to skrócenie czasów oczekiwania, np. obrabiarki na dostarczenie kolejnej palety do obróbki. Zadania transportowe wykonywane przez wózek

podzielono na dwa rodzaje czynności elementarnych: pobranie palety z magazynu oraz jazda wózka połączona z oddaniem palety. Wyodrębniono ponadto dodatkowe czynności dojazdu wózka do modułów oddania. Dzięki temu wózek może oczekiwać przy module oddania na pojawienie się palety zwróconej ze stacji. Innymi dodatkowymi czynnościami są czynności sprawdzania gotowości poszczególnych obiektów do pracy w trybie automatycznym.

Opracowany model poddano licznym eksperymentom symulacyjnym, które umożliwiły analizę poprawności działania systemu TOR oraz ocenę kryteriów jego efektywności.

### 3. MODERNIZACJA UKŁADÓW STEROWANIA URZĄDZENIAMI

Realizacja tego etapu modernizacji wymagała dokonania przeglądu urządzeń i układów mechanicznych, hydraulicznych, elektrycznych, oraz podjęcia decyzji, które elementy starego układu sterowania należy usunąć, a które pozostawić. Ze starego sterowania postanowiono pozostawić wszystkie czujniki, przyciski, krańcówki, wyłączniki itp., oraz układy mocy. Dokonano również przeglądu i identyfikacji (często „z natury”) układów wejściowych i wyjściowych oraz połączeń, co było niezbędne dla doboru i podłączenia sterowników CNC i PLC. Były to żmudne i trudne prace, głównie ze względu na duże braki w dokumentacji.

Kolejnym krokiem było wyspecyfikowanie typów i konfiguracji sterowników oraz dodatkowych elementów (np. złączy, interfejsów, oprogramowania narzędziowego), zaprojektowanie oraz wykonanie połączeń wejść i wyjść sterowników z odpowiednimi elementami sterowanych urządzeń, opracowanie oprogramowania sterowników do realizacji funkcji, ruchów elementarnych i sekwencji czynności sterowanych urządzeń. Zdecydowano się na wykorzystanie sterowników firmy *Siemens: PLC Simatic S5-115U* (do sterownia wózkiem, zmiennaczami palet oraz podajnikami przedmiotów) oraz *CNC SINUMERIK 805SM-TW* (do sterowania obrabiarkami). Oprogramowanie sterowników PLC napisano w języku STEP5.

Pokonując wiele, zgoła nieprzewidywalnych trudności wynikających z wieloletniego unieruchomienia systemu, usuwając podczas prób liczne usterki, których z tego samego powodu nie można było wcześniej zauważyć, doprowadzono do poprawnej pracy każdego z urządzeń.

### 4. KOMUNIKACJA POMIĘDZY MODUŁAMI UKŁADU STEROWANIA

Aby komputerowy układ sterowania mógł pracować efektywnie, konieczne jest zastosowanie właściwego sposobu przekazywania informacji między poszczególnymi jego modułami. Na dwóch najwyższych poziomach układu sterowania [3], na których pracują aplikacje **WinESP/MM** oraz **System Server**, wykorzystywana jest sieć *Ethernet* i protokół *TCP/IP* (środowisko *Windows NT*). Dobór rozwiązania komunikacyjnego na poziomie sterowników lokalnych (PLC i CNC), zależy od konkretnych urządzeń pracujących w systemie. W przypadku modernizacji systemu TOR zdecydowano się na wykorzystanie lokalnej sieci przemysłowej *Sinec L2* firmy *Siemens*. Warto podkreślić, że wykonana instalacja była pierwszą w Polsce instalacją sieci *Sinec L2* zawierającą sterowniki *SINUMERIK 805SM-TW*.

Przyjęto, że wszelkie informacje przesyłane w układzie sterowania (tzn. polecenia, zapytania o gotowość oraz potwierdzenia wykonania poleceń) będą zawarte w blokach danych o stałej długości. Struktura takiego bloku (tabela 2) została tak zaprojektowana, aby umożliwiał on przesłanie każdego typu informacji między dowolnymi dwoma modułami systemu.

O rodzaju informacji przekazywanej w bloku danych decyduje zawartość pola *Block header*. Wartość 0 oznacza polecenie lub potwierdzenie prawidłowego wykonania polecenia, wartość 51328 oznacza, że blok zawiera informację o błędzie. Inne wartości oznaczają polecenia sterowników kierowane do nadrzędnego układu sterowania (może to być np. polecenie przesłania programu numerycznego) oraz bloki informacyjne.

Jeśli blok jest poleceniem, pole *Act\_code* zawiera kod polecenia. Tabela 3 zawiera (jako przykład) kody poleceń wydawanych przez nadrzędny układ sterowania sterownikowi wózka.

Tabela 2. Format bloku danych wykorzystywany w układzie sterowania systemem TOR

Nr	Nazwa pola	Informacja zawarta w polu
1	Block_header	Nagłówek informujący o zawartości bloku
2	Act_number	Numer wykonywanej czynności
3	Act_code	Kod polecenia
4	Act_status	Kod błędu lub 0, jeśli polecenie wykonane poprawnie
5	Pal_num	Numer palety
6	Pal_type	Typ palety
7	Wp_num	Numer przedmiotu obrabianego
8	Wp_state	Stan przedmiotu obrabianego
9	Start_place	Początkowy adres ruchu wózka
10	End_place	Końcowy adres ruchu wózka
11	Mach_program	Numer programu obróbki
12	Wp_length	Długość przedmiotu obrabianego

Tabela 3. Kody poleceń realizowanych przez sterownik wózka

Kod	Polecenie	Opis
0	Sprawdź gotowość	Sprawdzenie gotowości wózka do pracy automatycznej
1	Dojeżdż do modułu	Dojazd wózka do określonego modułu w magazynie
2	Pobierz paletę	Pobranie palety z magazynu (w miejscu postoju wózka)
3	Oddaj paletę	Oddanie palety do magazynu (w miejscu postoju wózka)

Rozkazy inne niż *Sprawdź gotowość* realizowane są jedynie wówczas, gdy warunek gotowości w chwili otrzymania rozkazu jest spełniony. Brak gotowości wymaga zwykle ingerencji operatora. Polecenia potwierdzane są tylko wtedy, gdy zostały prawidłowo wykonane. Jeśli np. na skutek awarii polecenie nie zostało poprawnie zakończone i konieczna była ręczna ingerencja operatora, można je potwierdzić jedynie w serwerze systemowym.

Należy zwrócić uwagę, że nadrzędny układ sterowania nie kieruje polecenia obróbki bezpośrednio do sterownika obrabiarki, lecz do sterownika współpracujących z nią zmiennicza palet i podajnika przedmiotów. Na tym właśnie sterowniku spoczywa zadanie koordynacji współpracy wszystkich modułów stacji obróbkowej (tzn. zmiennicza, podajnika i obrabiarki), a także wydawanie poleceń obrabiarce. Dla zilustrowania przebiegu takiego procesu przedstawiono przykładową wymianę poleceń i informacji, która ma miejsce w przypadku obróbki na stacji tokarskiej. W tabeli 4 *Serwer* oznacza serwer systemowy, *Zmiennicz* oznacza sterownik zmiennicza palet w stacji tokarskiej, natomiast *TKX* oznacza sterownik tokarki TKX50NS. Warto zauważyć, że pomiędzy przesyłaniem kolejnych bloków sterownik zmiennicza i podajnika może wykonywać różne czynności (np. między otwarciem a zamknięciem konika podajnik przedmiotów dokonuje wymiany wałków). Sekwencja bloków 12 ÷ 19 w tabeli 4 przesyłana jest tyle razy, ile jest wałków na paletcie.

Wszystkie moduły zostały wyposażone w stosowne oprogramowanie obsługujące sieć *Sinec L2*. W celu umożliwienia wymiany informacji z wyższymi poziomami układu sterowania opracowano ponadto aplikację *Sinec L2 TCP/IP Driver*. Pracuje ona na komputerze wyposażonym w dwie karty sieciowe: *Ethernet*, oraz *Sinec L2* i pełni funkcję połączenia (*gateway*) pomiędzy tymi dwiema sieciami. Wszystkie bloki przekazywane z wyższych poziomów sterowania (konkretnie: z serwera systemowego) są przekazywane do właściwych stacji sieci *Sinec L2* i odwrotnie - bloki wysyłane przez sterowniki są odsyłane do serwera systemowego. W tabeli 5 wymieniono wszystkie stacje obecnej konfiguracji sieci *Sinec L2* w systemie TOR.

Tabela 4. Bloki przesyłane podczas obróbki przedmiotów na stacji tokarskiej

Nr	Nadawca	Odbiorca	Przesyłany blok
1	Serwer	Zmieniacz	Polecenie <i>Obrabiaj paletę</i>
2	Zmieniacz	TKX	Polecenie <i>Przygotuj obrabiarkę</i>
3	TKX	Serwer	Polecenie <i>Prześlij program numeryczny</i>
4	Serwer	TKX	Informacja o rozpoczęciu przesyłania programu
5	Serwer	TKX	Kolejne bloki programu numerycznego
6	Serwer	TKX	Informacja o zakończeniu przesyłania programu
7	TKX	Zmieniacz	Potwierdzenie polecenia <i>Przygotuj obrabiarkę</i>
8	Zmieniacz	TKX	Polecenie <i>Otwórz osłonę</i>
9	TKX	Zmieniacz	Potwierdzenie polecenia <i>Otwórz osłonę</i>
10	Zmieniacz	TKX	Polecenie <i>Otwórz konik</i>
11	TKX	Zmieniacz	Potwierdzenie polecenia <i>Otwórz konik</i>
12	Zmieniacz	TKX	Polecenie <i>Zamknij konik</i>
13	TKX	Zmieniacz	Potwierdzenie polecenia <i>Zamknij konik</i>
14	Zmieniacz	TKX	Polecenie <i>Obrabiaj</i>
15	TKX	Zmieniacz	Potwierdzenie polecenia <i>Obrabiaj</i>
16	Zmieniacz	TKX	Polecenie <i>Otwórz osłonę</i>
17	TKX	Zmieniacz	Potwierdzenie polecenia <i>Otwórz osłonę</i>
18	Zmieniacz	TKX	Polecenie <i>Otwórz konik</i>
19	TKX	Zmieniacz	Potwierdzenie polecenia <i>Otwórz konik</i>
20	Zmieniacz	Serwer	Potwierdzenie polecenia <i>Obrabiaj paletę</i>

Tabela 4. Stacje sieci Sinec L2 w układzie sterowania systemem TOR

Nr	Sprzęt	Funkcje
1	SINUMERIK 805SM-TW z modułem sieciowym	Sterowanie stacją tokarską <i>TKX</i>
2	Simatic S5-115U z kartą CP5430	Sterowanie wózkiem i zmienia-czem palet przy stacji <i>WCF</i>
3	Simatic S5-115U z kartą CP5430	Sterowanie podajnikami i zmieni-aczami w stacjach <i>TKX</i> i <i>FYM</i>
4	Komputer PC z kartą CP5412	Połączenie ( <i>gateway</i> ) pomiędzy sieciami <i>Ethernet</i> a <i>SINEC L2</i>
5	Simatic S5-115U z kartą CP5430	Sterowanie zmienia-czem palet w stacji załadowczo - rozładowczej
6	SINUMERIK 805SM-TW z modułem sieciowym	Sterowanie stacją frezarską <i>FYM</i>
7	Komputer PC z kartą CP5412	Wizualizacja systemu TOR

## 5. OPROGRAMOWANIE UKŁADU STEROWANIA SYSTEMEM TOR

Ponieważ programy pracujące na dwóch najwyższych poziomach sterowania, tzn. nadrzędny program sterujący **WinESP/MM** (przetwarzający model macierzowy realizowanego procesu wytwarzania) oraz **System Server** (zarządzający systemem), są uniwersalne (mogą być wykorzystane do sterowania każdym systemem), niezbędne było jedynie przygotowanie dla nich odpowiednich danych. W przypadku programu **WinESP/MM** dane te stanowił model

macierzowy (zob. rozdział 2), natomiast dla aplikacji **System Server** opracowano program skryptowy opisujący sposób zarządzania systemem TOR i obejmujący:

- definicję formatu bloku danych wykorzystywanego w układzie sterowania;
- skrypty dla 20 szablonów czynności, opisujące operacje, jakie należy wykonywać dla poszczególnych czynności modelu macierzowego;
- skrypty obsługujące błędy oraz polecenia odbierane od sterowników;
- definicje 30 identyfikatorów wykorzystywanych w skryptach;
- definicje pięciu tymczasowych miejsc przechowywania palet, tzn. miejsc poza magazynem, w których mogą znajdować się palety (zmiennicze palet oraz wózki);
- definicje identyfikatorów adresów sieciowych, pod które będą wysyłane bloki danych;
- definicje komunikatów błędów.

Poniżej przedstawiono dwa przykłady skryptów. Pierwszym jest skrypt *errors* obsługujący informacje o nieprawidłowym wykonaniu polecenia:

<code>Error = @Act_status</code>	przypisz zmiennej <i>Error</i> wartość z pola <i>Act_status</i> bloku danych
<code>Activity = _GetActivityNumber()</code>	przypisz zmiennej <i>Activity</i> aktualny numer czynności
<code>SendItem = _GetItem()</code>	przypisz zmiennej <i>SendItem</i> adres sieciowy, skąd nadeszło błędne potwierdzenie
<code>@Block_header = 0</code>	przypisz polom <i>Block_header</i> i <i>Act_status</i> wartości 0, tak aby
<code>@Act_status = 0</code>	blok nie zawierał już błędu
<code>_AddError(Error,Activity,SendItem)</code>	dodaj blok do listy błędów

Drugi przykład to skrypt dla szablonu \$1+\$2[#3], \$4+ZP #5\* (np. PAL2+PO3[3], FYM+ZP 4 \*):

<code>@Act_code = 1</code>	wpisz do pola <i>Act_code</i> wartość 1 (kod obróbki)
<code>@Pal_num = _GetObjectPaletteNum([ZP #5])</code>	wpisz do pola <i>Pal_num</i> numer palety na zmienniczu ZP #5 (ZP 4)
<code>@Pal_type = [\$1]</code>	wpisz do pola <i>Pal_type</i> typ palety - wartość stałej identyfikowanej przez \$1 (PAL2)
<code>@Wp_num = [\$2]</code>	wpisz do pola <i>Wp_num</i> numer przedmiotu - wartość stałej o nazwie \$2 (PO3)
<code>@Wp_state = #3</code>	wpisz do pola <i>Wp_state</i> wartość stałej #3 (2)
<code>@Mach_program = [NC\$2_\$4]</code>	wpisz do pola <i>Mach_program</i> wartość stałej o nazwie NC\$2_\$4 (NCPO3_FYM)
<code>@Wp_length = [L\$2]</code>	wpisz do pola <i>Wp_length</i> wartość stałej identyfikowanej przez L\$2 (LPO3)
<code>@Act_number = _GetActivityNumber()</code>	wpisz do pola <i>Act_number</i> numer aktualnej czynności
<code>_Send("ZP #5")</code>	wyślij blok pod adres identyfikowany przez ZP #5 (ZP 4)
<b>CONFIRM</b>	
<code>State = _Increment(@Wp_state)</code>	zwiększ stan przedmiotu na paletce o 1
<code>_SetObjectState([ZP #5],@Pal_num,@Pal_type,@Wp_num,State)</code>	umieść paletę na ZP #5 (ZP 4)

Dla układu sterowania systemem TOR opracowano dwie aplikacje dodatkowe, pracujące na III poziomie sterowania. Pierwszą z nich jest program obsługi magazynu **Magazyn TOR**. Jego zadaniem jest przechowywanie i udostępnianie informacji o stanie magazynu. Program obsługi magazynu współpracuje z serwerem systemowym, odpowiadając na jego zapytania (np. o miejsce, w którym znajduje się poszukiwana paleta), oraz uwzględniając przekazywane przezeń informacje (np. o tym, gdzie położono określoną paletę).

Drugą niezbędną w układzie sterowania systemem TOR aplikacją pomocniczą jest **Baza programów NC SINUMERIK 805**. Jej zadaniem jest przesyłanie wymaganych programów NC (przechowywanych na twardym dysku komputera) do pamięci sterownika SINUMERIK 805SM-TW. Układ sterowania stacji obróbkowej (tokarskiej lub frezarskiej) przed rozpoczęciem operacji obróbki zwraca się do serwera systemowego z żądaniem przesłania stosownego programu. Żądanie to jest kierowane do bazy programów numerycznych, która przesyła program w formacie właściwym dla sterownika SINUMERIK 805SM-TW.

Kolejnym etapem modernizacji układu sterowania było zapewnienie możliwości uwzględniania informacji mających istotny wpływ na przebieg procesu wytwarzania. Są one wprowadzane do nadrzędnego programu sterującego za pośrednictwem *macierzy warunków lokalnych* [3]. Konieczne stało się opracowanie aplikacji integrującej informacje o warunkach realizowalności czynności i przekazującej je do tej macierzy. Zadanie to spełnia aplikacja **Tor Monitor**, napisana w języku *Access Basic* i wykorzystująca bazę danych *Microsoft Access*.

## 6. URUCHOMIENIE ZMODERNIZOWANEGO UKŁADU STEROWANIA

System sterowania procesem wytwarzania oparty na jego modelu macierzowym ma dwie, bardzo istotne z punktu widzenia jego uruchamiania, zalety. Po pierwsze: istnieje możliwość wykonania dowolnej liczby eksperymentów symulacyjnych z wykorzystaniem tych samych programów i danych, które później będą stosowane do sterowania. Daje to możliwość łatwego, taniego i szybkiego sprawdzenia prawidłowości i efektywności działania sterowanego systemu. Po drugie: system może być uruchamiany stopniowo, poprzez dołączanie poszczególnych jego modułów i symulowanie działania pozostałych. Moduły wyizolowane w ten sposób łatwiej obserwować, szybciej można stwierdzić prawidłowość ich działania lub wykryć usterki i błędy. Dla przyspieszenia testów niektóre czynności (np. obróbka) mogą być pozorowane (np. „skrawanie powietrza”), a warunki prowadzące do blokowania czynności z powodu ich nierealizowalności, sztucznie wywoływane.

Ujawnione na tym etapie nieprawidłowości działania układu sterowania systemem TOR wynikały prawie wyłącznie z błędów w przesyłanych blokach danych lub w ich interpretacji.

## 7. ZAKOŃCZENIE

Zakres merytoryczny wykonanych prac jest obszerny, toteż w referacie jedynie w zarysie omówiono napotkane trudności, problemy oraz uzyskane wyniki, ograniczając się do najistotniejszych zagadnień. Na tej podstawie można sobie wyrobić pogląd na skalę zadań stojących przed projektantami i wykonawcami tego typu przedsięwzięć. Więcej szczegółów można znaleźć w opracowaniu [2]. Z całością osiągniętych rezultatów można zapoznać się w Katedrze Systemów Wytwarzania Politechniki Krakowskiej, podczas prezentacji działania zmodernizowanego systemu TOR.

## LITERATURA

- [1] J. Cyklis, W. Pierzchała: Modelowanie procesów dyskretnych w elastycznych systemach produkcyjnych. Zeszyty Naukowe Politechniki Krakowskiej, Mechanika z. 77, Monografia nr 3, Kraków 1995.
- [2] J. Cyklis, W. Pierzchała, J. Zając i in.: Opracowanie dyskretnego modelu przepływu informacji w systemie sterowania produkcją zautomatyzowaną. Raport z realizacji projektu badawczego Nr 7 T07D 026 08, ITMiAP, Politechnika Krakowska, Kraków 1997.
- [3] Pierzchała W.: Otwarty układ sterowania zautomatyzowanym wytwarzaniem. Materiały Konferencji AUTOMATION'98.