

# PRZEMYSŁOWY INSTYTUT AUTOMATYKI I POMIARÓW

OH

PIAP

A

Al. Jerozolimskie 202

02-486 Warszawa

Zespół Zrobotyzowanych Systemów Inteligentnych ZSI

Główny wykonawca: mgr inż. Zbigniew Pilat

Wykonawcy: doc. dr inż. Piotr Jabłoński  
mgr inż. Marek Pachuta  
mgr inż. Zbigniew Wawerek

Nr zlecenia: 1504C i 9635C

Tytuł pracy: Opracowanie i uruchomienie produkcji uniwersalnego robota sześcioposiowego o udźwigu 120/150 kg, dużym obszarze pracy i dużych prędkościach, dostosowanego do pracy w zautomatyzowanych gniazdach, liniach produkcyjnych i systemach ESP oraz CIM.

Numer i tytuł etapu Etap 3b. Oprogramowanie i badania układu sterowania URP wraz z łączem V24 (RS232C)

Zlecniodawca: Projekt celowy dofinansowywany ze środków Komitetu Badań Naukowych

Pracę rozpoczęto dnia: 01.1995

zakończono dnia: 05.1995

Kierownik Ośrodka

mgr inż. <sup>1</sup>Zbigniew Pilat

Dyrektor Pionu

dr inż. Jan Jabłkowski

Praca zawiera:

Rozdzielnik - ilość egz.:

stron 6  
rysunków --  
fotografii --  
tabel --  
tablic --  
załączników 9

Egz. 1 ZSI  
Egz. 2 ZSS  
Egz. 3 OME  
Egz. 4 OIN  
Egz. 5  
Egz. 6

Nr rej. 7347

**Analiza deskryptorowa:**

**ROBOTY PRZEMYSŁOWE, STEROWANIE, OPROGRAMOWANIE**

**Analiza dokumentacyjna:**

Sprawozdanie zawiera opis prac zrealizowanych w ramach uruchamiania, oprogramowywania i wstępnych prób układu sterowania URP przeznaczonego do robota o udźwigu 120/150 kg.

**Tytuły poprzednich sprawozdań:**

Niniejsze sprawozdanie jest pierwszym dokumentem przedstawiającym przebieg realizacji pracy nad robotem 120/150 kg w zakresie oprogramowania.

## 1. Wstęp

Robot o udźwigu 120/150 kg ma być oferowany w dwóch wersjach sterowania:

- z układem sterowania rho3 - będzie on kupowany bezpośrednio u producenta, w firmie Bosch (Niemcy),
- z układem sterowania URP - opracowany i produkowany w PIAP.

Układ URP ma konstrukcję modułową. Dzięki temu może być zestawiany w różnych konfiguracjach, stosownie do typu robota, a ściślej jego części mechanicznej (manipulatora) oraz do przewidywanych urządzeń zewnętrznych, z którymi sterownik ten ma współpracować. We wcześniejszych pracach prowadzonych w PIAP były wykorzystywane sterowania URP w wersji pięcio- i siedmioosiowej. Konfiguracja sześćoosiowa, aczkolwiek zestawiana przy okazji robota 120/150 kg po raz pierwszy, nie jest więc czymś zupełnie nowym. Niesie ze sobą jednak pewne szczególne trudności wynikające z mocy napędów tego robota. Przy udźwigu 120/150 kg trzeba było zastosować w części manipulacyjnej nowe silniki, znacznie silniejsze niż wykorzystywane wcześniej w robotach o udźwigu 60 kg. Można więc było oczekiwać pewnych trudności w obszarze stopni końcowych mocy układu sterowania. Osobnym problemem jest oprogramowanie podstawowe. Bazowy program sterujący robotów URP ma również budowę modułową. Przygotowując program dla konkretnego robota nie wystarczy jednak proste zestawienie odpowiednich bloków programowych, konieczne jest również wprowadzenie pewnych danych, charakterystycznych dla manipulatora. W efekcie otrzymujemy osobną wersję programu, przeznaczoną dla konkretnego manipulatora. Musi być ona odpowiednio sprawdzona i przetestowana w zakresie wszystkich funkcji, szczególnie dotyczy to funkcji ruchu. Do prób tego typu niezbędne jest posiadanie możliwości monitorowania stanu jednostki centralnej, formowania, przesyłania i gromadzenia komunikatów o sposobie wykonywania programu. Do tego celu najlepiej jest wykorzystać kanał transmisji szeregowej RS232C, przewidziany konstrukcyjnie w jednostce centralnej układu sterowania URP. W zamyśle projektantów robota 120/150 kg kanał ten jest przewidziany jako łącze informacyjne do wyższej warstwy sterowania. Jego sprawdzenie jest więc niezbędne już na etapie wstępnych prac uruchomieniowych układu sterowania. Dlatego też prace niniejszego etapu objęły uruchomienie docelowej szafy sterowniczej URP oraz jej wstępne oprogramowanie w zakresie obsługi kanału RS232C i posadowienia prototypowej wersji podstawowego programu sterującego robota.

## 2. Zadania zrealizowane w ramach uruchamiania, oprogramowywania i wstępnych prób układu sterowania URP przeznaczonego do robota o udźwigu 120/150 kg.

W ramach prac omawianego etapu zlecenia zrealizowano następujące zadania:

- 1) Zestawienie i uruchomienie w zakresie sprzętu układu sterowania URP-120/150.

Część cyfrowa układu sterowania została zestawiona w następującej konfiguracji:

- jednostka centralna MV-52,
- pamięć masowa MV-62,
- dwa bloki we/wy dwustanowych MV-12 - jeden systemowy i jeden obiektowy,
- zasilacz rezolwerów MV-21 w wykonaniu sześciokanałowym,
- sześć modułów cyfrowych sterowników osi (CSPO) MV-20.

W szafie sterowniczej zamontowano sześć modułów stopni końcowych mocy, w których zastosowano specjalne tranzystory mocy najnowszej generacji - tranzystory MOSFET typu IRFP-360.

- 2) Testy części napędowej i obiektowej układu sterowania URP-120/150.

Sprawdzono podstawowe obwody układu sterowania robota: zasilanie, zespół transformatorów, rozgałęzienie zasilania, prostownik i zasilacze niskonapięciowe, zespół bezpieczników i styczników oraz obwód STOP AWARYJNY. Przetestowano bloki we/wy zarówno systemowy jak i obiektowy. Działanie torów napędowych sprawdzono kolejno we współpracy z pojedynczym silnikiem robotowym - w tych testach CSPO pracowały pod kontrolą specjalnego oprogramowania uruchomieniowego.

- 3) Uruchomienie kanału transmisji szeregowej RS232C - posadowienie programu MONITOR OPERATORSKI w jednostce centralnej układu sterowania URP-120/150.

Jednostka centralna MV-52 posiada dwa kanały RS232C. Pierwszy jest przeznaczony do dołączenia panelu programowania. Drugi będzie wykorzystany do komunikacji w komputerem zewnętrznym. W załączniku A przedstawiono spis komend programu MONITOR OPERATORSKI. Jego posadowienie w jednostce centralnej umożliwiło prowadzenie dalszych testów.

- 4) Badanie łącza RS232C podczas pracy układu sterowania robota URP-120/150 pod kontrolą programu MONITOR OPERATORSKI.

Sprawdzono działanie poszczególnych komend programu MONITOR OPERATORSKI. W szczególności przebadano ładowanie zbiorów typu INTELHEX z komputera zewnętrznego do pamięci RAM jednostki centralnej oraz przesyłanie danych z jednostki do komputera i zapamiętywanie ich w formie pliku dyskowego. Przetestowano również współpracę komputera zewnętrznego z CSPO poprzez kanał RS232C i jednostkę centralną MV-52.

- 5) Posadowienie prototypowej wersji podstawowego programu sterującego robota URP-120/150.

Jako bazę wykorzystano standardowe oprogramowanie dla robota przemysłowego URP-6. Zostało ono wstępnie uruchomione w pamięci RAM jednostki centralnej, a po uzyskaniu poprawnie pracującej wersji przygotowano wzorce pamięci EPROM. Dograno do nich również MONITOR OPERATORSKI.

- 6) Testowanie funkcjonalne układu sterowania robota URP-120/150 pracującego pod kontrolą prototypowej wersji podstawowego programu sterującego robota URP-120/150.

Ponieważ jako prototypowy wykorzystano program dla robota pięcioosiowego, nie umożliwił on sprawdzenia torów napędowych, co jednak zrealizowano wcześniej, w p. 2). Przy pomocy tego programu sprawdzono funkcjonowanie komunikacji jednostki centralnej układu sterowania URP-120/150 z panelem programowania - poprzez łącze RS232C oraz z innymi modułami części cyfrowej - poprzez magistralę kasety. Sprawdzono wszystkie funkcje nie związane z ruchem robota, m.in. edycja programu użytkowego i jego ładowanie do/z komputera zewnętrznego poprzez łącze RS232C.

### 3. Wyniki przeprowadzonych prób i wnioski.

W efekcie prowadzonych w niniejszym etapie prac uzyskano następujące wyniki:

- sprawdzono działanie układu sterowania robota URP-120/150 - układ jest gotowy do połączenia z częścią manipulacyjną,
- sprawdzono działanie łącza RS232C w komunikacji jednostki centralnej układu sterowania robota URP-120/150 z komputerem zewnętrznym - kanał jest sprawny, umożliwia komunikację on-line oraz przesyłanie większych porcji informacji w formie zbiorów danych, w szczególności mogą to być programy użytkowe robota,

W dalszych pracach należy wykonać:

- połączenie i uruchomienie kompletnego robota URP-120/150 w zakresie sprzętowym,
- opracować, uruchomić i posadowić w postaci wzorców pamięci EPROM program sterujący robota URP-120/150,
- przebadać prototypowy robot URP-120/150 według przygotowanego programu prób,
- zweryfikować dokumentację układu sterowania URP-120/150 w zakresie sprzętu i oprogramowania.

## Zał. A Zestawienie funkcji programu MONITOR OPERATORSKI

WYWOŁANIE	NAZWA FUNKCJI	WYKONYWANA CZYNNOŚĆ:
C	CLEAR_SCREEN	kasowanie zawartości ekranu konsoli
D	DUMP	wyświetlenie zawartości określonego obszaru pamięci
F	FILL	wypełnianie pamięci określoną wartością
G	GO	start programu użyt. z możliwością ustawiania pułapki
H	HELP	inf. o dostępnych funkcjach i sposobie ich wywoływania
IB	INPUT_BYTE	odczyt wartości jednobajtowych z danego portu I/O
IW	INPUT_WORD	odczyt wartości dwubajtowych z danego portu I/O
K	KALKULATOR	arytmetyka hexadecymalna
L	LOAD	wpisywanie do pamięci informacji odbieranej w formacie INTELHEX z wybranego kanału transmisyjnego
SB	MEMORY_BYTE	wyświetlanie i zmiana zawartości pojed. bajtu pamięci
SW	MEMORY_WORD	wyświetlanie i zmiana zawartości pojed. słowa pamięci
M	MOVE	przepisanie zawartości jednego obszaru pamięci do innego obszaru
OB	OUTPUT_BYTE	wysyłanie wartości jednobajtowych do danego portu I/O
OW	OUTPUT_WORD	wysyłanie wartości dwubajtowych do danego portu I/O
P	PARAMETERS_OF_TRANSMISSION	odczyt i zmiana parametrów transmisji łącza szeregowego
Q	QUIT	skok do instr. znajdującej się pod adresem FFFF:0000
X	REGISTER	wyświetlanie i zmiana zawartości rejestrów użytkownika

<b>R</b>	<b>RETRANSMISSION</b>	wzajemna transmisja pomiędzy wybranymi kanałami bezpośrednio obsługiwanymi przez program MONITOR
<b>AB</b>	<b>SEARCH_BYTE</b>	poszukiwanie komórek pamięci o określonej zawartości
<b>AW</b>	<b>SEARCH_WORD</b>	poszukiwanie słów pamięci o określonej zawartości
<b>E</b>	<b>SEND_FROM_MEMORY</b>	cykliczne wysyłanie zawartości pamięci do jednego z kanałów transmisyjnych
<b>Y</b>	<b>SET_INTERRUPT_VECTORS</b>	ustawianie wektora przerw
<b>T</b>	<b>TRAP</b>	wykonanie pojedynczej instrukcji programu użytkownika
<b>U</b>	<b>UNASSEMBLE</b>	deasemblacja instr. mikroproc. 80186 i koproc. 8087
<b>V</b>	<b>VERIFY</b>	porównywanie zapisanej w pamięci informacji z danymi w formacie INTELHEX odbieranymi z wybranego kanału transmisyjnego
<b>W</b>	<b>WRITE_TO_MEMORY</b>	zapis do pamięci informacji odbieranej z jednego z kanałów transmisyjnych, bezpośrednio obsługiwanych przez program MONITOR