

mgr inż. Maciej Pabel², mgr inż. Marcin Prokopiak², dr hab. inż. Piotr Skrzypczyński¹

¹)Instytut Automatyki i Inżynierii Informatycznej, Politechnika Poznańska, ul. Piotrowo 3A, 60-965 Poznań, ²) P.P.H. WObit, ul. Gruszkowa 4, 61-474 Poznań

ROBOT TYPU SCARA O OTWARTEJ ARCHITEKTURZE SPRZĘTOWO-PROGRAMOWEJ

W niniejszej pracy przedstawiono konstrukcję i oprogramowanie niewielkiego robota manipulacyjnego typu SCARA charakteryzującego się otwartością architektury sprzętowo-programowej oraz niewielkimi kosztami budowy. Robot ten, wyposażony w szeroką gamę interfejsów komunikacyjnych oraz graficzne, przyjazne dla użytkownika środowisko programowania przeznaczony jest głównie do celów edukacyjnych, lecz może być także użyty do realizacji prostych prac w przemyśle, zastępując nieporównywalnie droższe roboty w układzie SCARA dostępne obecnie na rynku.

A LOW-COST, OPEN ARCHITECTURE SCARA-TYPE ROBOT

In this paper the design and software of a low-cost SCARA-type robot are presented. This robot has an open, modular architecture of the controller which can be easily extended by adding both additional effectors (robot axis) and sensors. The user of this robot has a broad choice of the communication interfaces to connect the robot to a PC, and a simple, user-friendly graphical programming environment. The main purpose of the developed robot is education, however the robot can be also an attractive choice for small or medium enterprises seeking a low-cost robotic solution to their packaging, sorting or assembly tasks.

1. WSTĘP

Obserwując rynek robotów przemysłowych oraz ich aplikacje w Polsce można zauważyć, że brak jest prostych, bardzo tanich manipulatorów przeznaczonych dla małych i średnich przedsiębiorstw. Budżet większości z nich nie pozwala na wdrożenie względnie drogiego robota przemysłowego oferowanego przez dużych producentów. Pewne parametry takich robotów, a w szczególności szybkość działania i bardzo duża precyzja pozycjonowania pozostają niewykorzystane w wielu prostszych zastosowaniach, takich jak produkcja opakowań z tworzyw sztucznych (pakowanie, paletyzacja), czy też nanoszenie spoiwa (klejenie). Sytuacja ta skłoniła poznańską firmę WObit działającą głównie w branży napędów elektrycznych, sterowników i elementów automatyki, do zainicjowania projektu prostego robota manipulacyjnego, który mógłby zapłacić wspomnianą lukę rynkową. Inicjatywa ta zbiegła się z zainteresowaniem Instytutu Automatyki i Inżynierii Informatycznej (IAiI) Politechniki Poznańskiej możliwością rozwinięcia bazy posiadanych robotów edukacyjnych. Manipulator dydaktyczny, poza niskim kosztem, umożliwiającym wdrożenie większej liczby jednostek, powinien charakteryzować się także otwartością systemu sterowania oraz możliwością rozbudowy i modyfikacji (modułowością).

W ramach współpracy między IAiI a firmą WObit zaprojektowany został i wykonany prototyp robota manipulacyjnego spełniający najważniejsze wymagania postawione w obu obszarach aplikacji: robota użytkowego dla przedsiębiorstw o ograniczonych możliwościach finansowych oraz manipulatora dydaktycznego.

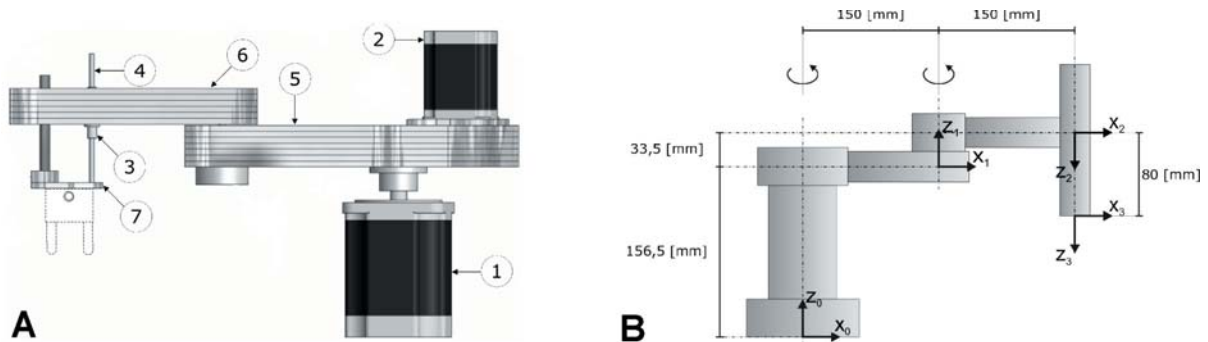
Spośród wielu możliwych struktur kinematycznych manipulatorów wybrano konfigurację SCARA (*Selective Compliant Articulated Robot for Assembly*)*. Manipulator typu SCARA został opracowany w końcu lat siedemdziesiątych ubiegłego wieku w zespole prof. Hiroshi Makino z Yamanashi University (Japonia) [9]. Manipulatory SCARA są klasyfikowane w grupie robotów monolitycznych o szeregowej strukturze kinematycznej. Posiadają dwie lub trzy obrotowe pary kinematyczne o osiach równoległych, które umożliwiają ruch w pewnej płaszczyźnie, oraz jedną parę przesuwną, realizującą ruch w kierunku prostopadłym do tej płaszczyzny [1]. Układ SCARA charakteryzuje się dużą sztywnością w płaszczyźnie ruchu pionowej osi przesuwnej oraz pewną podatnością w płaszczyźnie ruchu par obrotowych, stąd określane są mianem selektywnie podatnych (*selective compliant*). Układy napędowe par obrotowych nie muszą podtrzymywać ciężaru samego manipulatora czy też przenoszonego przedmiotu. Ponadto, silniki napędzające dwie pierwsze pary obrotowe manipulatora SCARA mogą zostać zabudowane w osi jego podstawy, a więc także nie obciążają ramienia. Powyższe cechy konstrukcyjne układu SCARA oraz względnie prosta konstrukcja mechaniczna ruchomych członów manipulatora (część obrotowa jest mechanizmem płaskim) sprawiają, że konfiguracja ta jest chętnie stosowana w przemyśle, a manipulatory SCARA mają w swej ofercie wszyscy główni producenci robotów przemysłowych. Manipulatory SCARA, ze względu na swoje cechy oraz kształt przestrzeni roboczej jest najbardziej przydatny do wykonywania zadań na płaszczyźnie. Roboty tego typu wykorzystywane są głównie przy montażu elementów i podzespołów, ale także przy powtarzalnym przenoszeniu detali oraz ich sortowaniu. Roboty takie są też chętnie używane w aplikacjach związanych z wytwarzaniem obwodów drukowanych.

Manipulatory typu SCARA mają też znaczny potencjał jako roboty edukacyjne. Są chętnie stosowane w dydaktyce związanej z przetwarzaniem obrazu, komputerowo zintegrowanym wytwarzaniem, mechatroniką [11] oraz telemanipulacją [4]. W wielu zastosowaniach edukacyjnych i badawczych szczególnie istotne wydają się takie cechy układu SCARA jak niewielka ilość miejsca zajmowanego przez robota przy relatywnie dużej przestrzeni roboczej oraz nieskomplikowana konstrukcja mechaniczna, która przyczynia się do dużej odporności na "złe traktowanie" przez użytkowników. Podstawowe założenia konstrukcyjne projektowanego robota zostały sformułowane następująco:

- zasięg ramienia porównywalny z konstrukcjami dostępnymi na rynku (od 300 mm do 350 mm);
- niski koszt wykonania robota;
- prosta konstrukcja mechaniczna, technologicznie przystosowana do wytwarzania małoseryjnego;
- modułowy układ sterowania umożliwiający dalszą rozbudowę robota (dodatkowe osie, czujniki);
- programowanie z komputera PC z możliwością sterowania z komputera oraz pracy samodzielnej po jego odłączeniu;
- szeroka gama interfejsów komunikacyjnych i możliwość wizualizacji stanów robota w czasie rzeczywistym;
- prosty, wizualny sposób programowania robota z nowoczesnym interfejsem graficznym.

Decyzją, która istotnie zaważyła na konstrukcji i parametrach zbudowanego prototypu manipulatora był wybór do jego napędu silników krokowych. Ten rodzaj silników elektrycznych ma wiele zalet, między innymi możliwość szybkiego rozbiegu, hamowania oraz zmiany kierunku a także precyzyjnego pozycjonowania, ma też jednak wady, takie jak trudności przy pracy z bardzo małymi prędkościami oraz możliwość pojawienia się rezonansu mechanicznego. Podjęta decyzja umotywowana była czynnikami związanymi z chęcią

maksymalnego ograniczenia kosztów robota: możliwością sterowania w otwartej pętli regulacji oraz dostępnością w ofercie firmy WObit silników krokowych charakteryzujących się łożyskami kulkowymi o znacznej średnicy, co umożliwiło osadzenie manipulatora bezpośrednio na osi silnika i uproszczenie konstrukcji mechanicznej. Także ze względu na ograniczenie kosztów zrezygnowano z trzeciej osi obrotowej, która w większości robotów SCARA realizuje zmianę orientacji efektora końcowego. Na rys. 1A przedstawiono ogólną strukturę ramienia robota *S08* zbudowanego we współpracy IAI i WObitu: 1 - silnik krokowy I członu, 2 - silnik krokowy II członu, 3 - aktuator liniowy III członu, 4 - śruba aktuatora liniowego, 5 - człon I, 6 - człon II, 7 - człon III. Rys. 1B przedstawia główne wymiary robota *S08*.



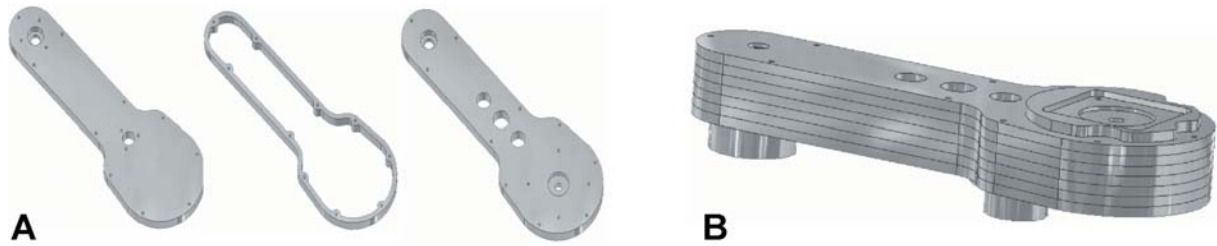
Rys. 1. Struktura robota *S08* (A) i jego podstawowe wymiary (B)

Do wiodących producentów manipulatorów SCARA należą m.in. Adept, Bosch, Epson, Kawasaki, KUKA, Toshiba, Yamaha. W ofertach większości z wymienionych producentów znajdują się manipulatory zbliżone wielkością do robota *S08*. Przykładami mogą być modele: *Adept Technology COBRA s350*, *Toshiba Machine TH350*, *Epson E2C351s* oraz *KUKA KR5 SCARA R350*. Wszystkie te roboty mają zasięg ramion około 350 mm i można zaliczyć je do tej samej klasy, do której należy *S08*. Pozwala to na wykorzystanie powstałego manipulatora w podobnych aplikacjach, co istniejące roboty. Należy zwrócić uwagę, że w porównaniu do prototypu *S08* wymienione tu roboty przemysłowe charakteryzują się lepszymi parametrami dotyczącymi udźwigu, prędkości oraz dokładności pozycjonowania końcówki roboczej, koszt ich zakupu jest jednak zdecydowanie większy niż koszt opisywanego tu rozwiązania.

2. KONSTRUKCJA MECHANICZNA I NAPĘDY

2.1. Struktura mechaniczna ramienia manipulatora

Konstrukcja mechaniczna manipulatora została zaprojektowana z uwzględnieniem możliwości wykonawczych, jakimi dysponowano. Charakter konstrukcji wykluczył zastosowanie technologii odlewu, która jest opłacalna jedynie przy produkcji wielkoseryjnej. Poszczególne człony manipulatora wykonano z płaskich elementów wyciętych z blachy przy wykorzystaniu maszyny CNC, co jest najtańszym rozwiązaniem dla produkcji pojedynczych egzemplarzy, a jednocześnie zapewnia odpowiednią jakość i powtarzalność wykonania [6]. Ponieważ *S08* ma spełniać także rolę robota edukacyjnego, jego ramię musiało posiadać zamkniętą konstrukcję, osłaniającą elementy wewnętrzne. Konstrukcję oparto więc o strukturę warstwową, łączoną śrubami, co zapewnia odpowiednią wytrzymałość i sztywność konstrukcji, a jednocześnie ułatwia montaż i serwis urządzenia. Jako materiału konstrukcyjnego użyto duraluminium PA6 (ENAW-2017A), wykorzystując arkusze blachy o grubości 5 mm.



Rys. 2. Konstrukcja I członu manipulatora: podstawowe elementy (A) i złożone ogniwo (B)

Na człon pierwszy składają się trzy podstawowe elementy oraz dodatkowe detale, niezbędne przy mocowaniu napędów i elektroniki robota (rys. 2). We wnętrzu tego członu umieszczono przekładnię pasową z pasem zębatym [12] przekazującą ruch obrotowy do drugiego członu robota. Konstrukcja mechaniczna drugiego członu jest bardzo zbliżona do budowy członu pierwszego. Ostatni człon manipulatora wykonuje ruch translacyjny. Do realizacji przesuwu wykorzystano aktuator liniowy, którego śruba jest mocowana ruchomo na detalu podstawowym tego członu. Wykorzystano łożysko liniowe (osadzone w członie drugim) oraz wał pokryty teflonem.

2.2. Napędy robota

Pierwszy człon manipulatora poruszany jest przy użyciu silnika krokowego 85BYGH450A, który ma moment 4,1 Nm i łożysko kulkowe o dużej średnicy, co umożliwiło osadzenie manipulatora bezpośrednio na jego osi. Duża masa (2,8 kg) silnika nie miała w tym przypadku znaczenia, ponieważ został on umieszczony w podstawie robota. Pierwsza oś może być przemieszczana o $\pm 150^\circ$ z maksymalną prędkością $360^\circ/\text{s}$ oraz pozycjonowana z rozdzielczością $0,1125^\circ$. Do napędu drugiego członu należało wykorzystać silnik o jak najmniejszej masie, ze względu na umiejscowienie go na ruchomym pierwszym członie manipulatora. Zdecydowano się na model 57BYGH803 o masie 1,1 kg, który cechuje się momentem 1,4 Nm. Zakres ruchu drugiej osi wynosi $\pm 140^\circ$, jej rozdzielczość kątowa $0,1125^\circ$, a maksymalna prędkość ruchu $360^\circ/\text{s}$. Ruch liniowy trzeciej osi zrealizowano przy użyciu niewielkiego silnika krokowego ze śrubą, typu 35BYZ-B01. Taki aktuator liniowy zapewnia dużą rozdzielczość przesuwu ($0,0254 \text{ mm}$) i posiada niewielką masę oraz małe wymiary, co pozwoliło na zamknięcie go wewnątrz drugiego członu manipulatora. Jego wadami są natomiast niska prędkość przesuwu (10 mm/s) i niewielkie wysunięcie maksymalne (do 70 mm). Wszystkie silniki wykorzystane w konstrukcji robota zostały wybrane z oferty firmy WObit [14].

3. ARCHITEKTURA i OPROGRAMOWANIE STEROWNIKA ROBOTA

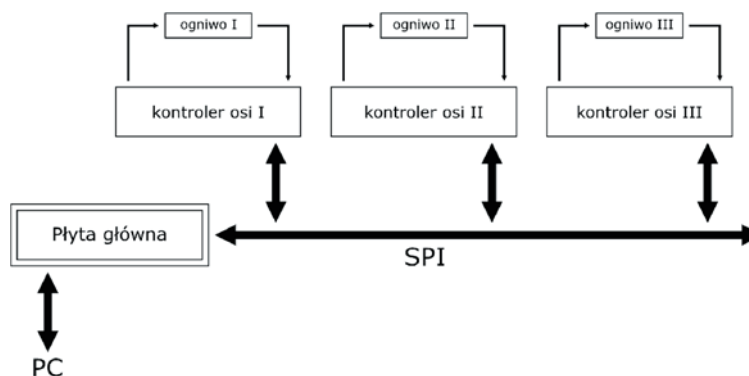
3.1. Sterownik nadrzędny

W robocie *S08* zastosowano modułową strukturę sterownika, który został podzielony na sterownik nadrzędny (tzw. płytę główną) oraz kontrolery poszczególnych osi. Każdy moduł posiada odrębny mikrokontroler, a komunikacja między nimi odbywa się po szeregowej magistrali SPI [5]. Przyjęcie takiej architektury sterownika uwarunkowane było znacznym stopniem komplikacji jego struktury, co wynika z obecności wielu urządzeń peryferyjnych, koniecznością wykonywania złożonych obliczeń w czasie rzeczywistym oraz chęcią uzyskania rozwiązania otwartego, podatnego na rozbudowę i modernizację (rys. 3).

Zaprojektowany sterownik umożliwia samodzielną pracę robota *S08* bez nadzoru komputera PC, po wcześniejszym zaprogramowaniu. Podstawowe funkcje wypełniane przez sterownik nadrzędny robota to:

- nadzór nad sterownikami osi poprzez system przerwań i cykliczne odpytywanie;
- rozwiązywanie zadań kinematyki prostej i odwrotnej;
- odtwarzanie zaprogramowanej uprzednio trajektorii;
- komunikacja z użytkownikiem.

Sterownik ten został zbudowany w oparciu o silny, 32-bitowy mikroprocesor AT91SAM7S256 firmy *Atmel* [10]. Jest to połączenie wydajnego rdzenia ARM7TDMI z dużą liczbą rozbudowanych urządzeń peryferyjnych. Sterownik został wyposażony w nieulotną pamięć Flash. Część pamięci jest zarezerwowana do przechowywania kodu programu napisanego przez użytkownika robota, reszta służy do przechowywania danych serwisowych. Znaczna pojemność pamięci (256 KB) umożliwia zapamiętanie przez robota złożonych sekwencji ruchów, a w połączeniu z zegarem czasu rzeczywistego umożliwia logowanie wszystkich stanów awaryjnych, co jest bardzo pomocne przy serwisowaniu urządzenia.



Rys. 3. Modułowa budowa sterownika robota S08

Sterownik nadrzędny został wyposażony w układy transmisyjne Ethernet, USB i RS-232 służące do komunikacji z komputerem PC. Pierwszy rodzaj transmisji zaimplementowano (przy użyciu układu ENC28J60 firmy *Microchip* [8]) jako główny sposób komunikacji manipulatora z komputerem PC. Duża szybkość przesyłania danych jest niezbędna między innymi przy wizualizacji stanu robota w czasie rzeczywistym. Magistrala RS-232 znajduje natomiast zastosowanie jako interfejs diagnostyczny robota lub środek komunikacji z innymi urządzeniami w jego otoczeniu. Łącze RS-232 (lub USB) umożliwia podłączenie prostego panelu sterującego, a także urządzenia podrzędnego – np. sterownika przenośnika taśmowego lub stolika pozycjonującego obrabiany element. Na płycie głównej znajduje się także gniazdo służące do podłączenia kolorowego, ciekłokrystalicznego wyświetlacza graficznego. Umożliwia on wizualizację podstawowych informacji o stanie robota bez użycia komputera PC.

Na płycie głównej umieszczone zostały cztery gniazda na kontrolery osi. W prototypie S08 wykorzystywane są trzy sterowniki osi. Wolne gniazdo umożliwia łatwą rozbudowę manipulatora o czwarty stopień swobody, bez konieczności modyfikacji sterownika nadrzędnego. W konstrukcji płyty sterownika przewidziano także możliwość podłączenia akcelerometru LIS3I02 firmy *STMicroelectronics* przy wykorzystaniu specjalnie przygotowanego złącza [3]. Rozbudowa robota o ten czujnik umożliwiający pomiar przyspieszenia efektora końcowego może być interesująca w zastosowaniach dydaktycznych. Możliwa jest także rozbudowa sterownika nadrzędnego o czujniki wykorzystujące magistralę jedнопроводową (*1-Wire*).

W celu zapewnienia bezpieczeństwa oraz niezawodności pracy robota, jego układ sterowania posiada odrębne zasilanie. Na płycie głównej został umieszczony czterostopniowy układ zasilania, który dostarcza energii do wszystkich komponentów elektronicznych robota.

3.2. Sterowniki osi

W robocie *S08* zastosowano 3 identyczne sterowniki osi. Każdy z nich, podobnie jak układ nadrzędny, został oparty na 32-bitowym mikrokontrolerze z rdzeniem ARM. Sterowniki wpinane są w płytę główną przy pomocy specjalnych gniazd, a komunikują się z resztą systemu przez magistralę SPI z częstotliwością 24 MHz. Główne zadania sterowników osi to:

- regulacja położenia osi;
- wyznaczanie i kontrola profilu prędkości osi;
- obsługa efektora końcowego (np. chwytaka) w przypadku kontrolera trzeciej osi.

W związku z użyciem silników krokowych dużej mocy konieczne było wykorzystanie wysokonapięciowych sterowników jako końcówek mocy. Zastosowane urządzenia, wybrane z oferty firmy WObit [15], pozwalają na realizację sterowania mikrokrokowego, które zapewnia bardziej płynną pracę silnika.

Po głębszej analizie rozwiązań konstrukcyjnych istniejących robotów SCARA zrezygnowano z początkowej koncepcji sterowania osi w otwartej pętli i w osiach obrotowych par kinematycznych zastosowano absolutne czujniki położenia kąтового wału silnika. Rozwiązanie takie zwalnia robota z konieczności pozycjonowania się zarówno przy starcie, jak i w trakcie cyklu roboczego, co ma istotny wpływ na jego wydajność. W prototypowym robocie *S08* użyto 12-bitowe enkodery magnetyczne z wyjściem szeregowym AS5045 firmy *Austria Microsystems AG* [2].

Każdy ze sterowników osi umożliwia także podłączenie dwustanowych czujników krańcowych informujących o napotkaniu przeszkody. W prototypie *S08* rozwiązanie to zostało wykorzystane tylko w sterowniku trzeciej osi, gdzie nie było możliwości montażu enkodera absolutnego.

3.3. Oprogramowanie sterownika robota

Oprogramowanie sterownika robota zostało napisane w języku wysokiego poziomu, zgodnym ze standardem języka C. Do przygotowania programów wykorzystano darmowe środowisko programistyczne *Eclipse* połączone z równie darmowym zestawem narzędzi *GNU ARM Toolchain*, w skład którego wchodzi między innymi kompilator języka C/C++ dla mikrokontrolerów ARM.

W programie głównym robota wyróżnić można podprogramy inicjalizujące system, wykonywane cyklicznie (w nieskończonej pętli) oraz wywoływane na skutek wystąpienia przerwania. Obsługa większości funkcji przerwania ogranicza się do ustawienia programowej flagi informującej o wystąpieniu zdarzenia. Jednakże funkcje takie jak komunikacja przez Ethernet, czy obsługa silników, są całkowicie wykonywane w funkcji obsługi danego przerwania. Aby uniknąć nieprzewidzianych zachowań systemu na czas obsługi jednego zdarzenia, zgłaszanie pozostałych przerw jest blokowane. Wyjątkiem od tej reguły są obsługa przerwania taktującego silniki krokowe oraz procedura kontroli transmisji przez Ethernet. Dzięki przydzielonemu wysokiemu priorytetowi, są one zawsze obsługiwane bez zwłoki.

Obsługa kontrolerów osi od strony układu nadrzędnego ogranicza się do wysłania informacji o nowej pozycji do osiągnięcia lub zmiany stanu efektora końcowego oraz odebrania informacji o aktualnej pozycji, w której znajduje się dana oś.

Oprogramowanie sterownika nadrzędnego realizuje także rozwiązanie zadań kinematyki prostej i odwrotnej manipulatora [1]. Ponieważ zadawanie pozycji wewnątrz robota (na linii układ nadrzędny – kontrolery osi) odbywa się z użyciem zmiennych przegubowych, algorytm obliczania kinematyki prostej pełni jedynie funkcje pomocnicze – jest on wykorzystywany

w procedurze wizualizacji stanu robota na graficznym wyświetlaczu LCD oraz komputerze PC. Implementacja algorytmu rozwiązywania zadania kinematyki odwrotnej była natomiast niezbędna, aby wyznaczyć położenia poszczególnych osi prowadzące do osiągnięcia pozycji końcówki roboczej zadanej we współrzędnych zewnętrznych. W celu uniknięcia nieprzewidzianych zachowań robota, w przypadku zadania przez użytkownika błędnej (nieosiągalnej) pozycji we współrzędnych zewnętrznych, do podprogramu rozwiązania zadania kinematyki odwrotnej wprowadzona została procedura sprawdzająca osiągalność danego punktu.

4. PROGRAMOWANIE ROBOTA W ŚRODOWISKU GRAFICZNYM

Znanych jest wiele sposobów programowania robotów przemysłowych [7]. Efektywną metodą, szczególnie dla prostszych zadań typu przenoszenie detali, jest programowanie off-line, gdzie program przygotowywany jest w całości na komputerze zewnętrznym w pewnym języku programowania, a następnie wysyłany do robota. Większość ze znajdujących się na rynku manipulatorów, w tym także typu SCARA, ma specyficzny język programowania. Przykładami mogą być język V+, używany w robotach firmy *Adept* oraz KRL koncernu *KUKA*, używany także w manipulatorach SCARA.



Rys. 4. Ekran graficznego środowiska programowania robota S08

Również dla robota S08 opracowany został prosty, blokowy język programowania oraz dedykowane środowisko graficzne umożliwiające intuicyjne zadawanie sekwencji ruchów. Szatę graficzną programu obsługi robota z poziomu komputera PC zaprojektowano z myślą o przejrzystym i przyjaznym wyglądzie interfejsu. Program został napisany przy użyciu *Borland C++ Builder* – środowiska programistycznego klasy RAD (*Rapid Application Development*) przeznaczonego dla systemów z rodziny *MS Windows*.

Po uruchomieniu aplikacji, okno główne programu zajmuje pełen ekran (rys. 4). Na głównym panelu programu zostały umieszczone podstawowe przyciski funkcyjne aplikacji. Ich aktywacja wywołuje okna programu pełniące konkretne funkcje.

- **Komunikacja** – okno konfiguracji połączenia komputera PC z robotem.
- **Programowanie** – panel graficznej edycji programów (sekwencji ruchów).
- **Sterowanie** – panel ręcznego sterowania robota.
- **Wizualizacja** – okno wizualizacji stanów robota.

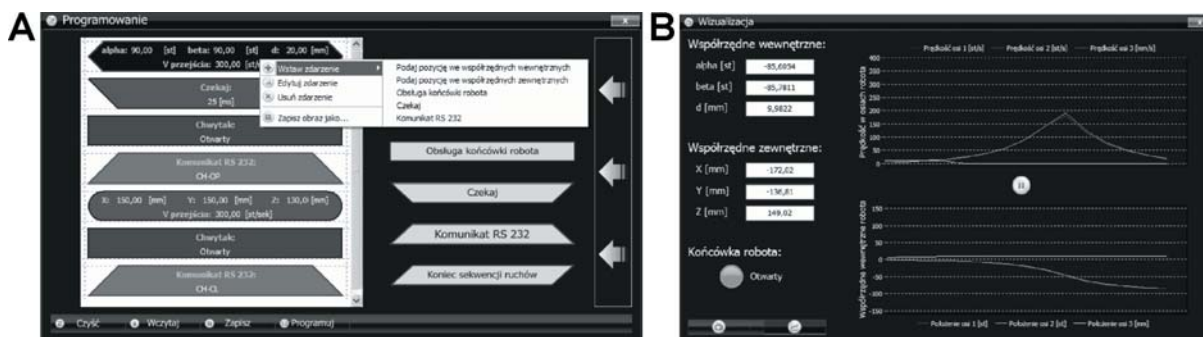
- **Podgląd** – panel podglądu tekstowego stanu robota oraz ramek wysyłanych i odbieranych przez Ethernet i RS-232.
- **Konfiguracja** – okno konfiguracji zegara RTC oraz ustawiania parametrów pomocniczych.
- **Pomoc** – wyświetla informacje niezbędne do obsługi robota.

Oprócz podstawowych przycisków aplikacji, na panelu głównym znajdują się również trzy "diody" sygnalizujące stan robota oraz status połączenia komputera PC z manipulatorem.

Głównym elementem panelu programowania (rys. 5A) jest okno podglądu kodu graficznego. Osobne okno komend umożliwia wstawianie bloków funkcyjnych do programu robota. Aktywowanie określonego rozkazu powoduje pojawienie się na ekranie jego okna ustawień. Poszczególne rozkazy są przedstawiane w postaci graficznych bloków, jeden pod drugim. Wewnątrz poszczególnych bloków, wyróżniających się kolorem i kształtem, zapisane są ustawione parametry rozkazów. Dostępnych jest 6 typów rozkazów.

- **Podaj pozycję we współrzędnych wewnętrznych.** Rozkaz umożliwia zadanie położenia robota we współrzędnych wewnętrznych oraz określenie prędkości przejścia do zadanej pozycji. Przed akceptacją rozkazu sprawdzana jest realizowalność zakresu ruchu ramienia oraz zadanej prędkości.
- **Podaj pozycję we współrzędnych zewnętrznych.** Rozkaz umożliwia zadanie pozycji końcówki manipulatora w globalnym układzie współrzędnych kartezjańskich. Należy także określić prędkość przejścia robota do podanej pozycji. Przy zatwierdzaniu rozkazu sprawdzana jest osiągalność pozycji i prędkości.
- **Obsługa końcówki robota.** Rozkaz umożliwia zamknięcie lub otwarcie chwytaka.
- **Czekaj.** Rozkaz czekaj pozwala na zatrzymanie pracy robota na określony czas, podany w milisekundach. Maksymalny czas opóźnienia to 99999 ms.
- **Komunikat RS-232.** Wysłanie komunikatu o maksymalnej długości 10 znaków do urządzenia zewnętrznego.
- **Koniec sekwencji ruchów.** Rozkaz zakończenia sekwencji operacji. Istnieje możliwość powtórzenia sekwencji określoną liczbę razy (do 99999 powtórzeń), lub powtarzania sekwencji aż do jej przerwania przez operatora.

Okno programowania posiada menu kontekstowe, które umożliwia wstawianie nowych rozkazów (zawsze przed wskazanym rozkazem), edycję wskazanej komendy lub jej usunięcie.



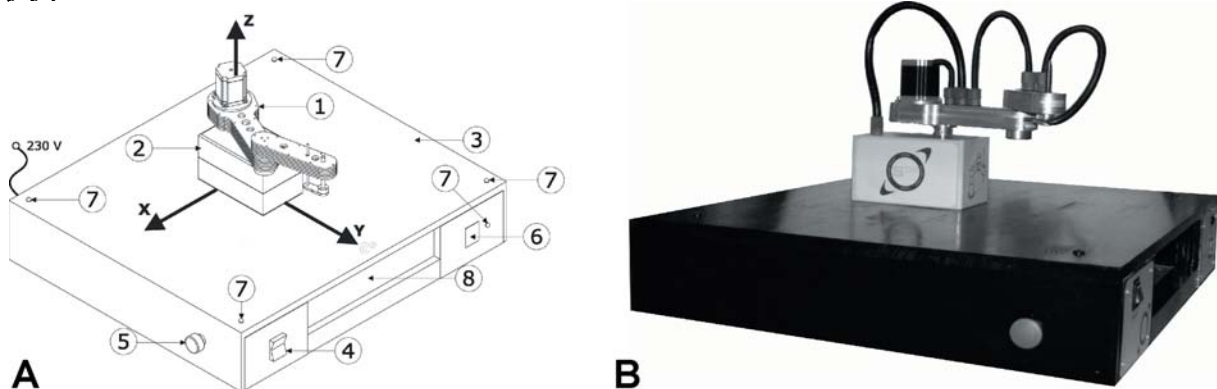
Rys. 5. Widok panelu programowania (A) i okna wizualizacji stanu z przebiegami prędkości i położenia (B)

Odrębne okno sterowania umożliwia użytkownikowi zadawanie pozycji robota, tak we współrzędnych wewnętrznych, jak i zewnętrznych. Sprawdzanie osiągalności w tym trybie sterowania odbywa się po stronie mikrokontrolera, a sterujący użytkownik nie ma możliwości przekroczenia wartości granicznych.

W oknie wizualizacji stanów robota wyświetlane są wartości współrzędnych wewnętrznych i zewnętrznych robota oraz prosta animacja bieżącej konfiguracji ramienia manipulatora. Animacja ta jest aktualizowana w czasie rzeczywistym z użyciem szybkiego protokołu UDP. Okno wizualizacji umożliwia również podgląd przebiegów prędkości i położenia w poszczególnych osiach robota w postaci wykresów (rys. 5B). Dodatkowo dostępny jest tekstowy podgląd stanów robota, informujący między innymi o starcie systemu, stanie połączenia z robotem oraz ewentualnych awariach.

5. STANOWISKO LABORATORYJNE

Ponieważ prototypowy manipulator *S08* przeznaczony jest do celów dydaktycznych, osadzono go na podstawie stanowiącej kompletne, zwarte stanowisko laboratoryjne. Ze względów bezpieczeństwa wszystkie komponenty elektroniczne i elektryczne zabudowano w podstawie, na której umieszczono też wyłącznik awaryjny. Podstawa ma wymiary 800 mm x 700 mm x 150 mm, a przestrzeń robocza manipulatora znajduje się całkowicie na jej powierzchni.



Rys. 6. Schemat (A) i widok (B) stanowiska laboratoryjnego z robotem *S08*

Na przednim panelu podstawy stanowiska laboratoryjnego umieszczono kolorowy wyświetlacz graficzny LCD. Obok współrzędnych wewnętrznych robota wyświetlane jest na nim też aktualne położenie końcówki robota we współrzędnych zewnętrznych (w milimetrach). Dodatkowo wyświetlacz informuje o stanie połączenia robota, poprzez Ethernet lub RS-232, z innymi urządzeniami (np. komputerem PC). Funkcję informacyjną pełnią też diody LED umieszczone w narożnikach górnej płyty stanowiska, które świecą się na czerwono w czasie ruchu robota, a na zielono, gdy manipulator znajduje się w spoczynku.

Rys. 6A przedstawia schemat rozmieszczenia elementów stanowiska laboratoryjnego: 1 - manipulator, 2 - obudowa silnika I członu, 3 - podstawa, 4 - włącznik zasilania, 5 - wyłącznik awaryjny, 6 - wyświetlacz LCD, 7 - diody LED, 8 - komponenty sterownika. Na rys. 6B przedstawiono widok kompletnego stanowiska.

6. PODSUMOWANIE

Przedstawiony w niniejszym artykule prototyp robota *S08* został wykonany w ramach pracy dyplomowej magisterskiej realizowanej w Instytucie Automatyki i Inżynierii Informatycznej Politechniki Poznańskiej [13]. W jego konstrukcji i oprogramowaniu udało się pogodzić większość wymagań stawianych robotowi dydaktycznemu oraz prostemu robotowi przemysłowemu, dostępnemu dla niewielkich przedsiębiorstw (koszty wykonania prototypu, łącznie ze stanowiskiem laboratoryjnym wyniosły około 5000 zł.). Realizacja tego projektu możliwa była dzięki współpracy z WObitem. Wiele komponentów robota wybrano z oferty tego przedsiębiorstwa, a jego zaplecze warsztatowe umożliwiło osiągnięcie założonych

standardów technologicznych wykonanego prototypu. Opracowanie robota *S08* stanowi także przykład nowoczesnego podejścia do kształcenia kadry inżynierskiej we współpracy między publiczną wyższą uczelnią techniczną a innowacyjnym przedsiębiorstwem prywatnym.

LITERATURA

- [1] Craig J., *Podstawy robotyki, mechanika i sterowanie*, WNT, 1993.
- [2] Czujnik położenia kąтового AS5045 Austria Microsystems AG, www.austriamicrosystems.com
- [3] Czujnik przyspieszenia STMicroelectronics LIS3l02, www.st.com
- [4] Goldberg K., Gentner S., Sutter C., Wiegley J., Farzin B., The Mercury Project: A Feasibility Study for Online Robots, w: *Beyond Webcams. An Introduction to Online Robots*, (K. Goldberg, R. Siegwart, eds.), MIT Press, 2002, s. 17–36.
- [5] Hadam P., *Projektowanie systemów mikroprocesorowych*, BTC, 2004
- [6] Heimann B., *Mechatronika: komponenty, metody, przykłady*, PWN, 2001
- [7] Honoczarenko J., *Roboty przemysłowe, budowa i zastosowanie*, WNT, 2004
- [8] Kontroler Ethernet Microchip ENC28J60, www.microchip.com
- [9] Makino H., Kato A., Yamazaki Y., Research and Commercialization of SCARA Robot – the Case of Industry-University Joint Research and Development, *Int. Journal of Automation Technology*, 1(1), 2007, s. 61–67.
- [10] Mikrokontroler Atmel AT91SAM7S, www.atmel.com
- [11] Nagchaudhuri A., Kuruganty S., Shakur A., Mechatronics Education Using an Industrial SCARA Robot, *Proc. 7th Mechatronics Forum International Conference*, Atlanta, 2000, CD-ROM.
- [12] Podzespoły napędowe firmy WHM, www.whm.pl
- [13] Pabel M., Prokopiak M., *Konstrukcja i oprogramowanie robota typu SCARA*, praca dyplomowa magisterska, Politechnika Poznańska, Wydział Elektryczny, 2008.
- [14] Silniki krokowe firmy WObit, www.silniki.pl
- [15] Sterowniki silników firmy WObit, www.sterowniki.pl