

Prototypowanie sterownika dla robota IRp-6

1. Opis techniczny robota IRp-6.

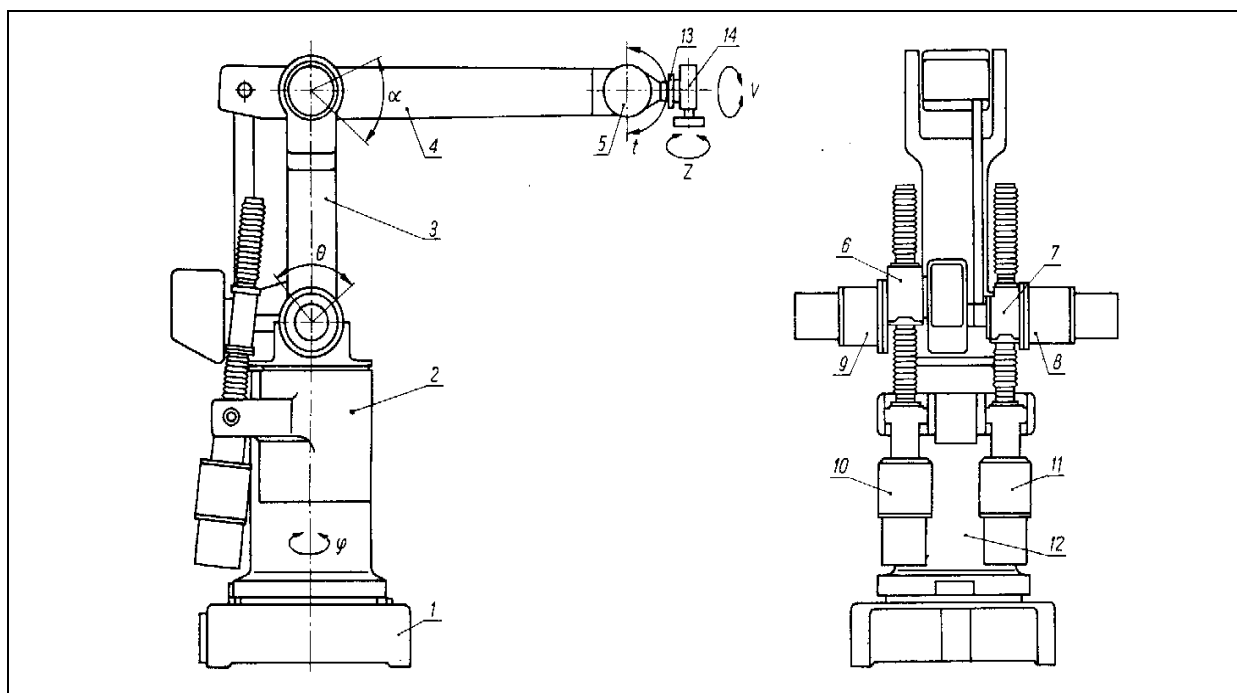
Robot IRp-6 jest robotem elektrycznym wykorzystującym silniki prądu stałego do realizacji ruchu poszczególnych osi (ramion). Łańcuch kinematyczny IRp-6 składa się z pięciu stopni swobody. Do pomiaru rzeczywistej prędkości obrotowej wału każdy silnik robota został wyposażony w prądnicę tachometryczną. Do pomiaru położenia wałów poszczególnych silników użyto resolver'ów

Część manipulacyjna robota IRp-6.

Część manipulacyjna robota obejmuje elementy mechaniczne, elementy napędowe oraz elementy pomiarowe. Składa się ona z podstawy, korpusu, ramienia dolnego, górnego oraz zespołu przegubu zakończony końcówką kołnierkową, służącą do zamocowania urządzeń wykonawczych.

Opis rysunku 1

1.podstawa robota	6.przekładnia śrubowa toczna ruchu θ	11.napęd ruchu α
2.korpus robota	7.przekładnia śrubowa toczna ruchu α	12.napęd ruchu φ
3.ramię dolne	8.napęd ruchu v	13. końcówka kołnierkowa
4.ramię górne	9.napęd ruchu t	14. oś z
5.zespół przegubu	10.napęd ruchu θ	



Rys.1 Część manipulacyjna robota IRp-6.

Realizacja poszczególnych ruchów części manipulacyjnej dokonywana jest przez tzw. zespoły silników i odpowiednie przekładnie mechaniczne. W skład każdego zespołu napędowego wchodzi :

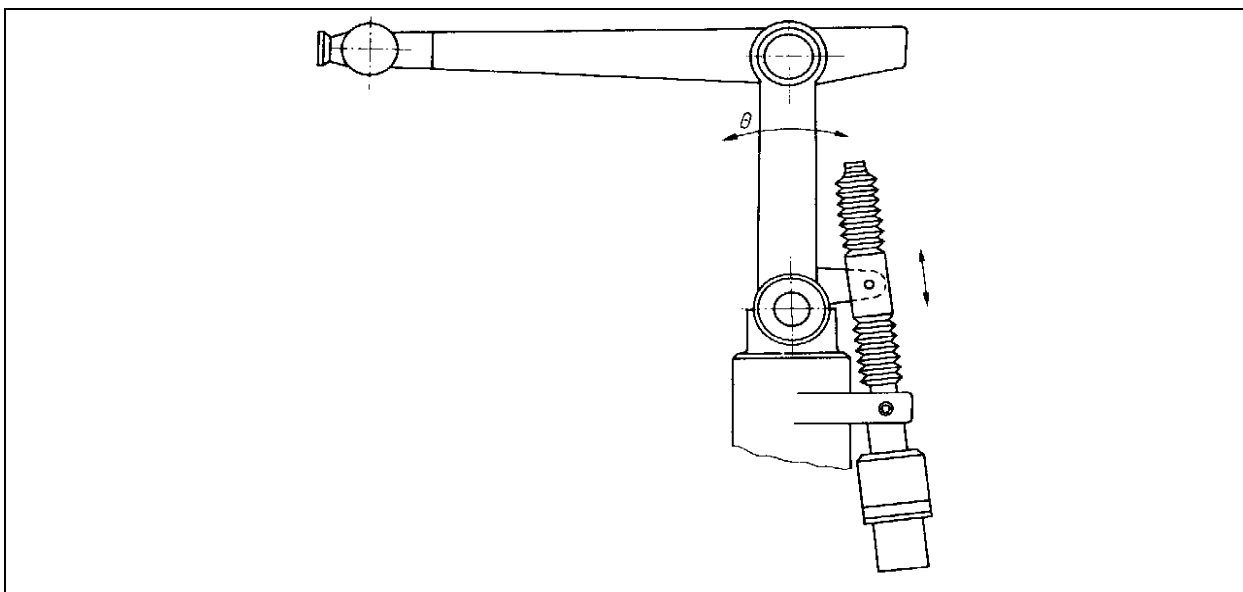
silnik prądu stałego

transformator położenia kąтового (resolver)

prądnica tachometryczna

Na korpusie są zamocowane napędy ruchów $\theta(10)$ i $\alpha(11)$ dolnego i górnego ramienia, składające się z zespołu silnika i przekładni śrubowej tocznej (6,7) oraz napędy ruchów przegubu v(8) i t(9) złożone z zespołu silnika i przekładni falowej. W górnej części podstawy jest umieszczony zespół silnika i przekładni falowa nadająca korpusowi ruch obrotowy $\varphi(12)$. Ramię dolne jest łożyskowane obrotowo we wsporniku korpusu.

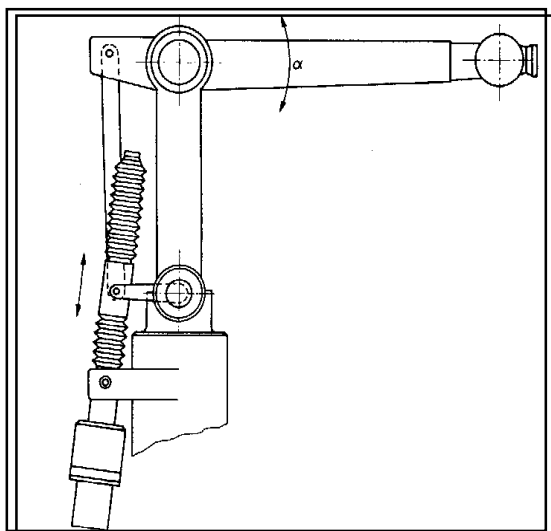
Ruch przekładni falowej jest przekazywany do dolnego ramienia za pomocą dźwigni połączonej z nakrętką tej przekładni.



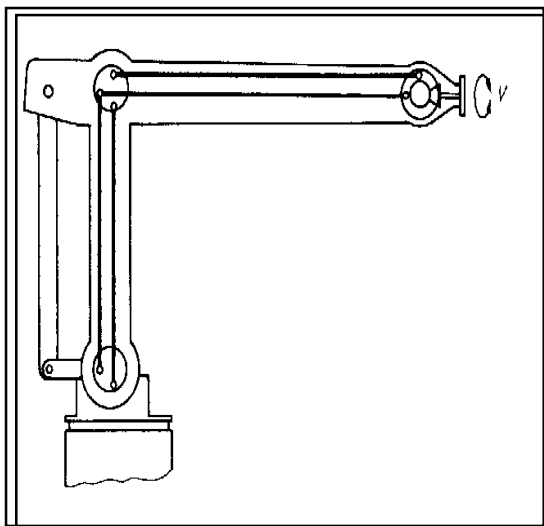
Rys.2. Realizacja ruchu θ dolnego ramienia.

Ramię górne łożyskowane jest obrotowo w górnej części ramienia dolnego. Ruch obrotowy silnika jest przenoszony za pomocą przekładni śrubowej tocznej na dwa pręty zamocowane przegubowo do nakrętki przekładni i tworzące razem z dolnym i górnym ramieniem równoległobok, co wywołuje ruch górnego ramienia (Rys. 3).

Ruch silników osi v i t jest przekazywany do końcówki kołnierzonej zespołu przegubu za pośrednictwem przekładni falowych i zespołów złożonych z tarcz obrotowych i cięgien, umieszczonych po obu stronach zespołu ramion. Przy realizacji ruchu ostatnim członem nadającym ruch końcówce kołnierzonej jest przekładnia stożkowa (Rys.4). Mechanizm realizacji ruchu t, umieszczony z drugiej strony ramion, takiej przekładni nie zawiera. Do końcówki kołnierzonej przegubu jest mocowany chwytak, narzędzie lub inne urządzenie wykonawcze. Przewody elektryczne i przewody sprężonego powietrza do sterowania chwytaka są poprowadzone wewnątrz górnego ramienia. Gniazdo do przyłączenia przewodu sprężonego powietrza znajduje się w tylnej części tego ramienia, a wyjście sprężonego powietrza i zaciski wyjść elektrycznych są umieszczone w jego przedniej części.



Rys. 3. Realizacja ruchu α górnego ramienia.



Rys. 4. Realizacja ruchu v końcówki kołnierzowej zespołu przegubu.

W sterowaniu robotem konieczne staje się przetworzenie sygnału położenia osi. W robocie IRp-6 jako przetwornik położenia osi wykorzystywany jest resolver, czyli transformator położenia kąтового. W skład resolvera wchodzi dwie uzwojone części: stojan i wirnik.

Wirnik jest sprzężony z ruchomym wałem silnika napędzającego oś robota.

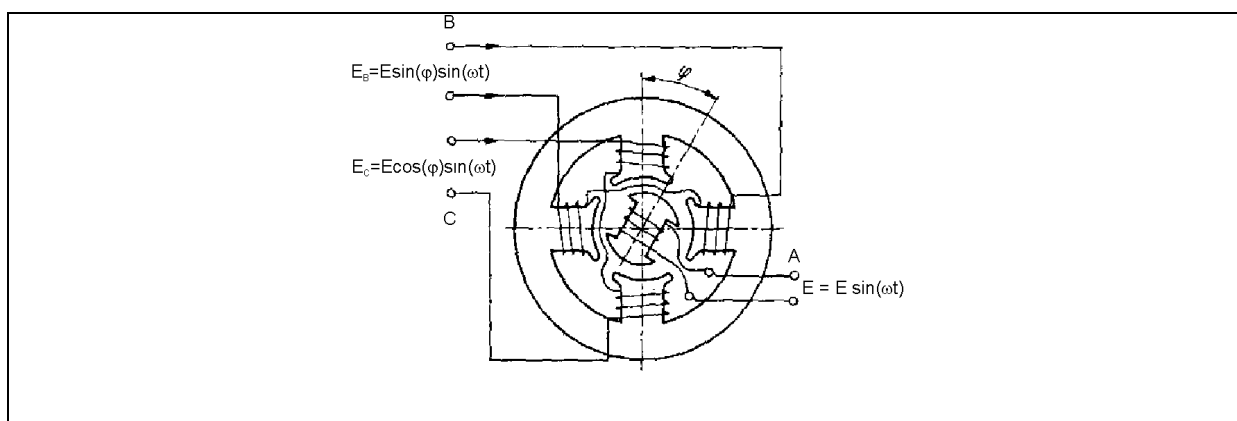
Uzwojenie wirnika zasilane jest sygnałem referencyjnym postaci:

$$EA = \sin(\omega t)$$

o częstotliwości 5kHz. W uzwojeniach stojana indukują się następujące napięcia:

$$EB = E \sin \varphi \sin(\omega t) \text{ oraz } EC = E \cos \varphi \sin(\omega t),$$

gdzie φ jest miarą położenia kąтового wirnika resolvera.



Rys.5 Resolver

Sygnałami pomiarowymi położenia są więc dwa sygnały sinusoidalne modulowane amplitudowo sinusem i cosinusem kąta obrotu wału silnika, z których układ sterowania wyznacza (po uwzględnieniu przekładni) kąt obrotu ramienia.

Dla właściwego sterowania ruchem robota nie wystarczy tylko znajomość położenia (w tym przypadku z resolver'ów). Oprócz sprzężenia od położenia istnieje również sprzężenie od prędkości ruchu osi.

Rzeczywista prędkość jest mierzona za pomocą prądnicy tachometrycznej sprzężonej z wałem silnika napędowego robota. Dostarcza ona do sterownika mocy napięcie stałe proporcjonalne do prędkości ruchu.

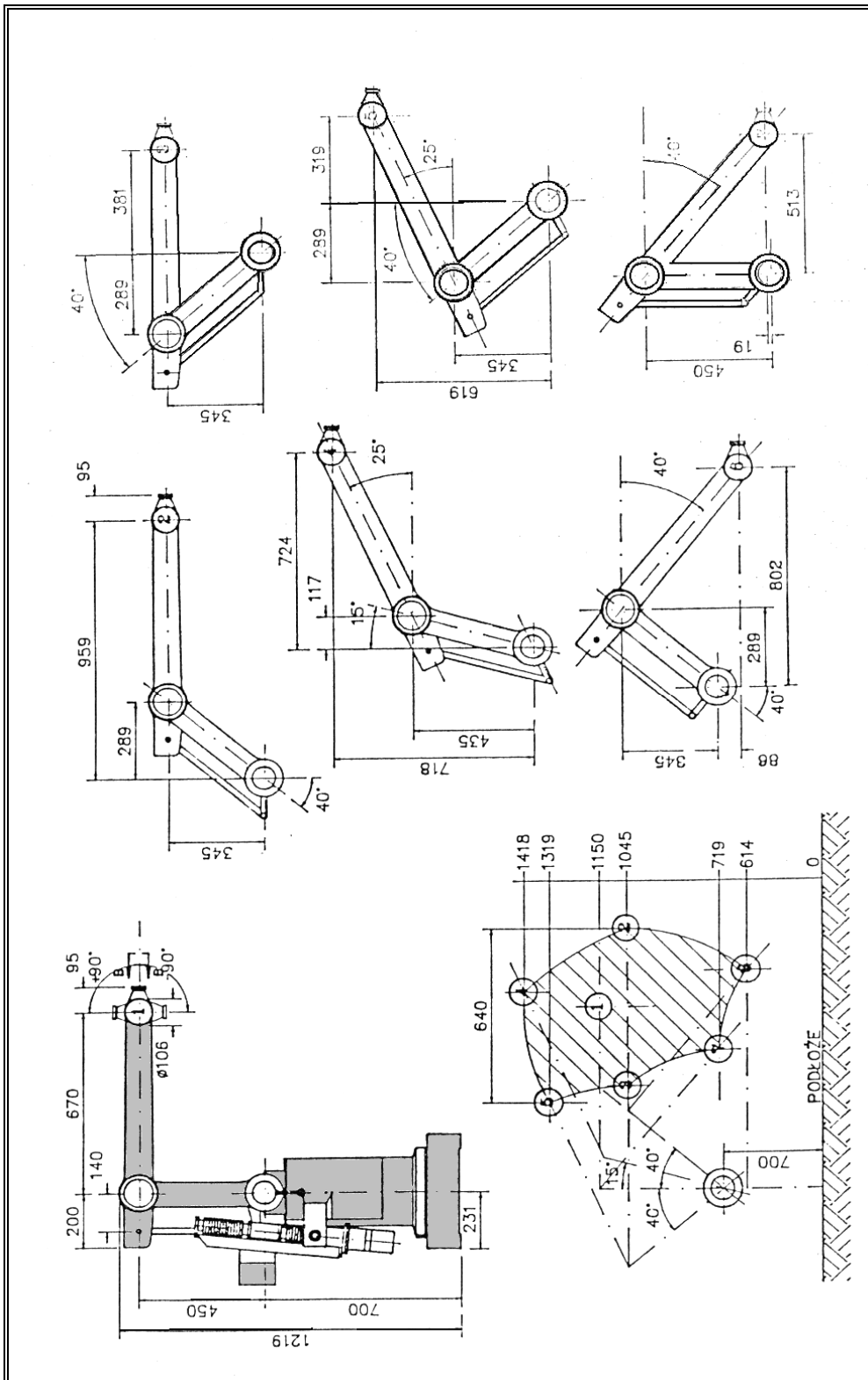
Sygnal odniesienia prędkości V_c w postaci analogowej dochodzi do sterownika prędkości znajdującego się w sterowniku mocy, gdzie porównywany jest z sygnałem prędkości rzeczywistej z prądnicy tachometrycznej. Na podstawie różnicy między tymi wartościami jest wytwarzany sygnał zadający prądu twornika silnika. Sygnał rzeczywistej wartości prądu jest uzyskiwany z rezystora pomiarowego. Różnica pomiędzy wymaganą a rzeczywistą wartością prądu jest wzmacniana i dostarczana do modulatora impulsów. Szeregowo z silnikiem jest włączony dławik wygładzający impulsy.

Poniżej zamieszczono zestaw wybranych parametrów robota IRp-6 uwzględniający kątowy zakres ruchu poszczególnych osi.

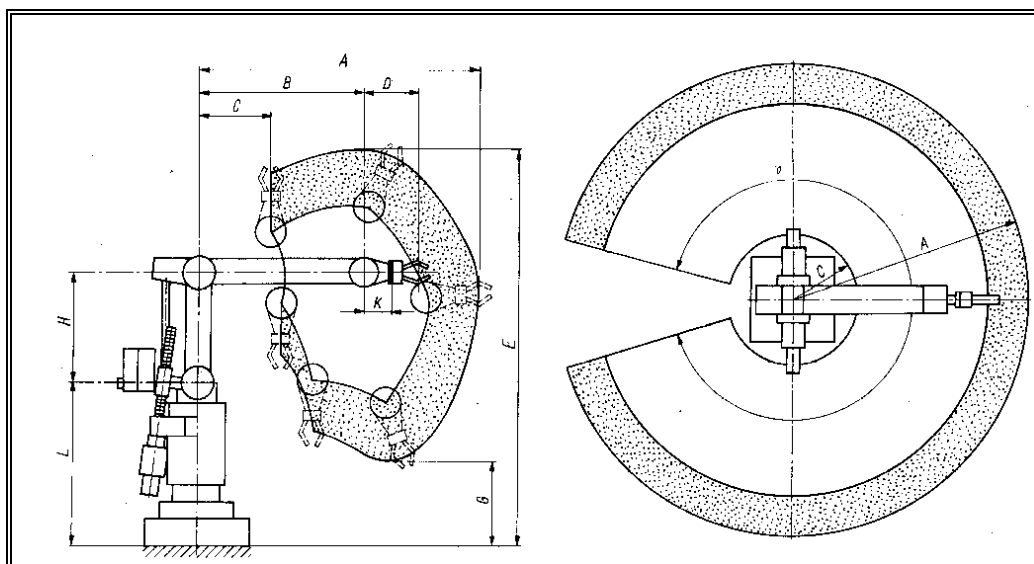
Nr osi	Przełożenie	Prędkość [°/s]	Zakres ruchu [°]
1	$6.33 \cdot 10^{-3}$	60	± 160
2	$6.37 \cdot 10^{-3}$	60	± 40
3	$6.33 \cdot 10^{-3}$	60	-25..+40
4	$7.87 \cdot 10^{-3}$	75	-25 ...+120
5	$1.32 \cdot 10^{-2}$	125	-25....+150

Tab.1. Wykaz wielkości mechanicznych opisujących poszczególne osie robota IRp-6.

Poniżej zamieszczono dwa rysunki ilustrujące zakres roboczy IRp-6.



Rys.6. Przestrzeń robocza robota IRp-6. Przykładowe osiągalne pozycje robota.



Rys.7. Przestrzeń robocza robota IRp-6.

Zespoły napędowe poszczególnych osi obejmujące silnik prądu stałego, przekładnię oraz resolver do pomiaru położenia kąтового (umieszczony przed przekładnią) połączone są z komputerem poprzez kartę procesorową **DS1005** oraz kartę pomiarową **DS2201** systemu dSPACE (wejścia analogowe karty wykorzystane do pomiaru położenia, wyjścia przetwornika DA do wysyłania sygnału sterującego do zespołów napędowych, wejścia cyfrowe do pomiaru stanu czujników bazowych).

Należy uruchomić **Matlab R2015a**. Następnie uruchomić środowisko **Simulink**.

W środowisku Simulink zrealizować w postaci schematu blokowego sterownik dla 5 osi robota (serwomechanizmy z regulatorem P), przy wykorzystaniu bloków z biblioteki Simulink-a oraz bloków związanych z kartą **DS2201** (biblioteka **dSPACE RTI1005/DS2201**).

Bloki realizujące funkcje pomiarowe i sterujące karty **DS2201** potrzebne do realizacji schematu to:

- DS2201ADC_B1** - pomiar sygnałów z resolverów
- DS2201DAC_B1** - wysyłanie sygnału sterującego
- DS2201IN_B1** - pomiar sygnałów z czujników bazowych.

Dla uzyskania pomiaru położenia osi robota należy na wyjściu bloku **DS2201ADC_B1** umieścić bloki realizujące operację przeliczającą sygnał z resolvera na stopnie, uwzględniając przekładnię osi. Gotowe podsystemy realizujące tą operację dla poszczególnych osi znajdują się w podbibliotece **irplib** (uruchamianie biblioteki poprzez wpisanie nazwy **irplib** w Comand Window).

W tworzonym schemacie blokowym, regulator w serwomechanizmach dla poszczególnych osi robota, ustawić na działanie proporcjonalne o wzmacnieniu 0.5.

Sygnał sterujący wyliczony przez regulator powinien być, przed podaniem na wyjście, ograniczony do ± 3 (wykorzystać blok *Saturation*) oraz wprowadzony na blok *Rate Limiter* (parametry: 50, -50).

Przed wejściem na blok *DS2201DAC_B1* należy umieścić blok *Gain* o wzmacnieniu 0.1 (wymaganie karty DS2201 związane z normowaniem sygnału).

1. Wykorzystując narzędzie **Real-Time Interface** wygenerować i załadować do karty DS1005 aplikację czasu rzeczywistego odpowiadającą zrealizowanemu schematowi blokowemu:
 - w oknie roboczym wybrać: *Code/C/C++ Code/Code Generation Options*
 - w zakładce *Solver* sprawdzić ustawienie parametrów
Stop time : inf
Solver options: Type: Fixed-step oraz metodę całkowania (Euler)w podzakładce *Additional Options*
 - Fixed step size*: 0.001 (jest to krok próbkowania)
 - powrócić do zakładki *Code/C/C++ Code* po czym wybrać opcję *Build Model (Ctrl+B)* ; wybór tej opcji powoduje automatyczną generację kodu C ze schematu blokowego, kompilację, linkowanie i załadowanie aplikacji do karty DS1005, która przejmie wykonywanie programu.
2. Przy pomocy pakietu **ControlDesk 6.0** zbudować pulpit operatorski umożliwiający komunikację z procesem realizowanym na karcie DS1005, tzn. zmiany i monitorowanie parametrów procesu oraz obserwacje i rejestrowanie przebiegów zmiennych:
 - po otwarciu *ControlDesk* wybrać *File / New* a następnie *Create New Project and Experiment*
 - nadać nazwę dla tworzonego projektu oraz ustawić właściwą kartotekę, po czym wybrać *Next>*
 - nadać nazwę dla eksperymentu, wybrać *Next>*
 - w kolejnym oknie zaakceptować platformę *DS1005 PPC Board*, wcisnąć *NEXT>*
 - korzystając z opcji *Import from file* wczytać z odpowiedniej kartoteki zbiór *nazwa_aplikacji.sdf* zawierający informacje o zmiennych realizowanej aplikacji, wcisnąć *Finish*
 - w polu roboczym, przy pomocy dostępnych wirtualnych narzędzi (zakładka *Instrument Selector*), utworzyć panel operatorski
 - przypisać zmienne do wirtualnych narzędzi na pulpicie
 - uruchomienie utworzonego panelu (uruchomienie komunikacji z aplikacją czasu rzeczywistego) poprzez wybranie przycisku *Go Online*.

Funkcje jakie ma realizować wykonany prototyp sterownika:

1. Sterowanie napędów: serwomechanizmy dla poszczególnych osi z uwzględnieniem ograniczeń dla wartości zadanych kątów
2. Bazowanie robota
3. Pozycjonowanie w układzie współrzędnych konfiguracyjnych - tryb pracy „ręcznej” (niezależne ruszanie osiami) i tryb „wyzwalany” (ustawianie wartości zadanych dla osi i uruchomienie ruchu)
4. Zadawanie prędkości roboczej ruchu
5. Koordynacja prędkości ruchu
6. Pozycjonowanie w układzie kartezjańskim: implementacja prostego zadania kinematyki pozycji