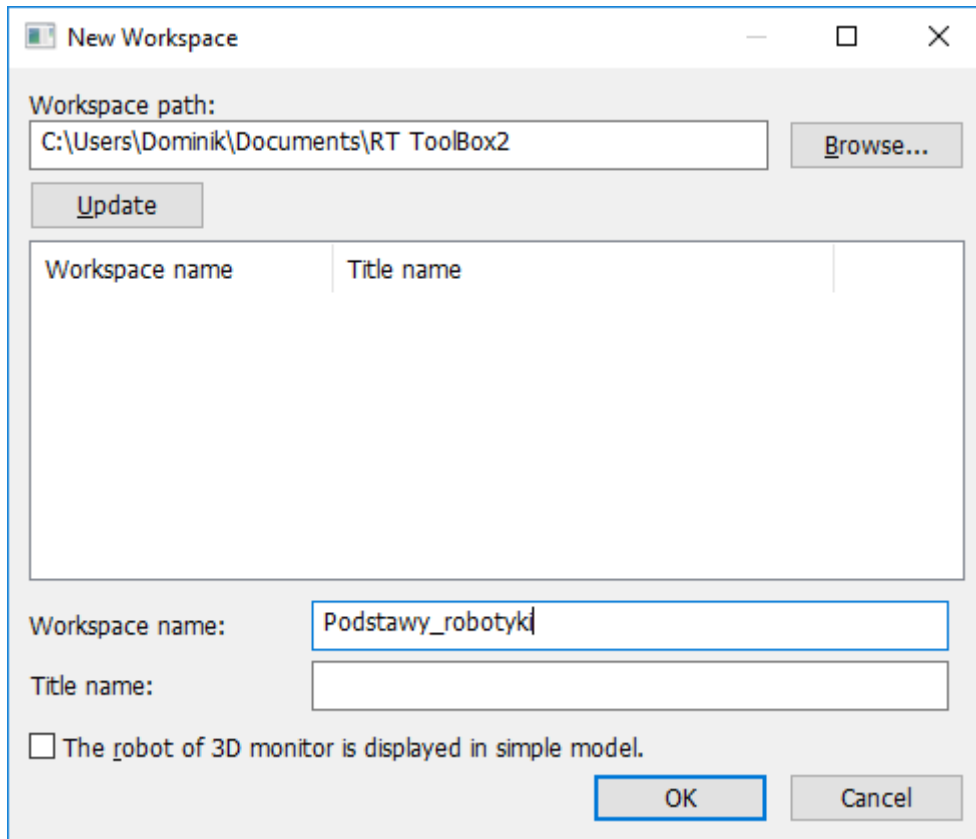


## Instrukcja na zajęcia laboratoryjne z przedmiotu „Podstawy robotyki”

### Zajęcia III - IV – „Wprowadzenie do obsługi programu RT ToolBox2”

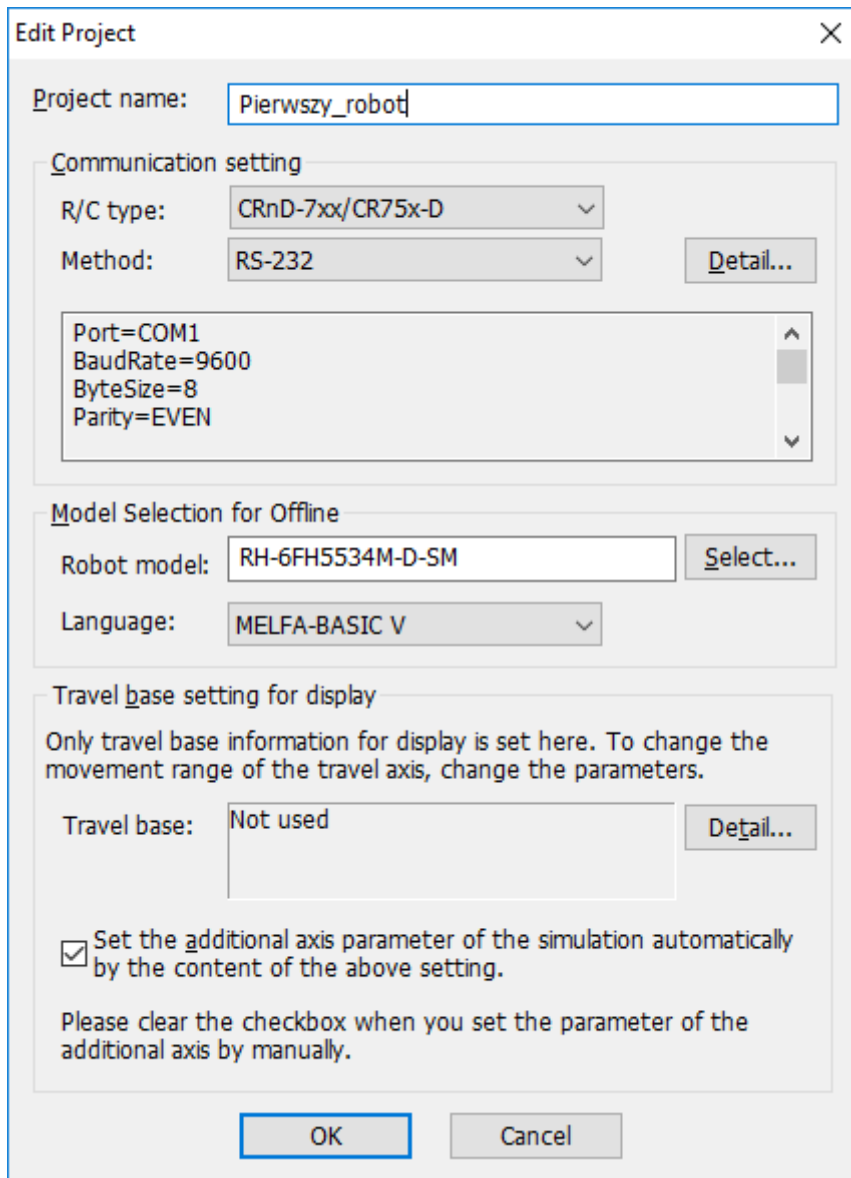
1. Utworzenie przestrzeni roboczej



Po uruchomieniu programu z menu „WorkSpace” wybrać polecenie „New”. Otworzy się okno umożliwiające utworzenie przestrzeni roboczej w wybranym katalogu.

Po utworzeniu przestrzeni pojawi się okno edycji projektu. Należy w nim podać nazwę projektu, a następnie w sekcji „Communication setting” wybrać rodzaj komunikacji pomiędzy fizycznym robotem, a komputerem. Jednak nie każdy model robota jest produkowany we wszystkich wariantach komunikacji. Komunikację tą należy również wybrać w sytuacji przygotowania wyłącznie symulacji działania robota, bez fizycznego połączenia z nim. Na zajęciach laboratoryjnych wybieramy opcję komunikacji przez port RS-232 tak jak pokazano na zrzucie ekranu poniżej.

W sekcji „Model Selection for Offline” można wybrać model programowanego robota. Po kliknięciu przycisku „Select” istnieje możliwość podglądu dostępnych modeli robota. Można również określić język w jakim będą zapisane instrukcje dla robota. W przypadku zajęć laboratoryjnych należy wybrać opcje tak jak zamieszczono na rysunku poniżej.



## 2. Pierwszy program

W drzewie przestrzeni roboczej po lewej stronie okna programu należy rozwinąć zakładkę „Offline”, kliknąć prawym przyciskiem myszy na „Program” oraz z rozwijanego menu wybrać „New”. Następnie należy podać nazwę programu. W opisywanym przykładzie będzie to „Pierwszy”. Nastąpi otwarcie okna pisania programu. Należy wpisać w nim poniższy prosty program:

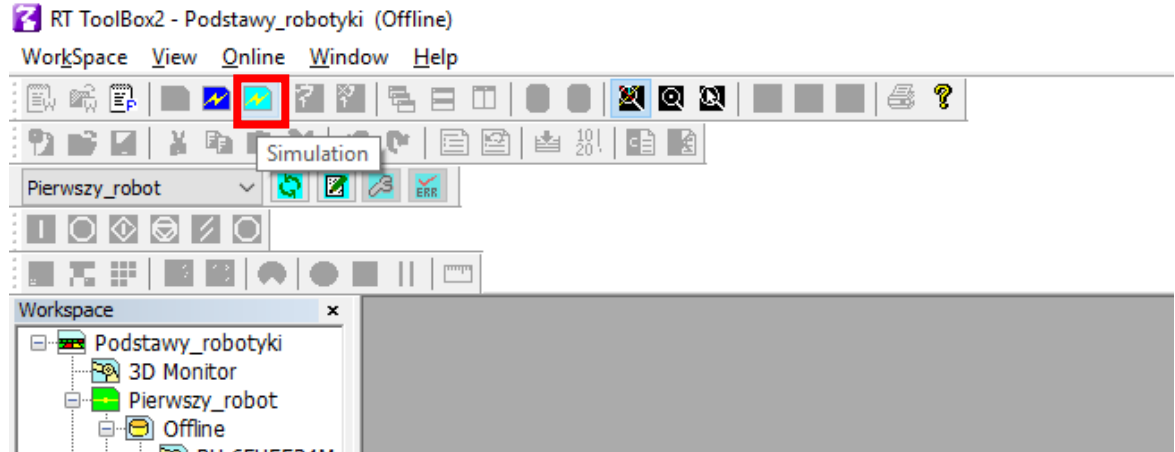
```
Mov PStart  
Mov PEnd  
End
```

Powyższy kod oznacza, że końcówka robocza manipulatora powinna przejechać ruchem prostoliniowym na pozycję punktu „PStart”, następnie na punkt „PEnd” oraz zakończyć działanie programu. Nazwy wszystkich punktów muszą zaczynać się od dużej litery P.

Po napisaniu programu należy zamknąć jego okno zapisując obecny jego stan.

### 3. Symulacja

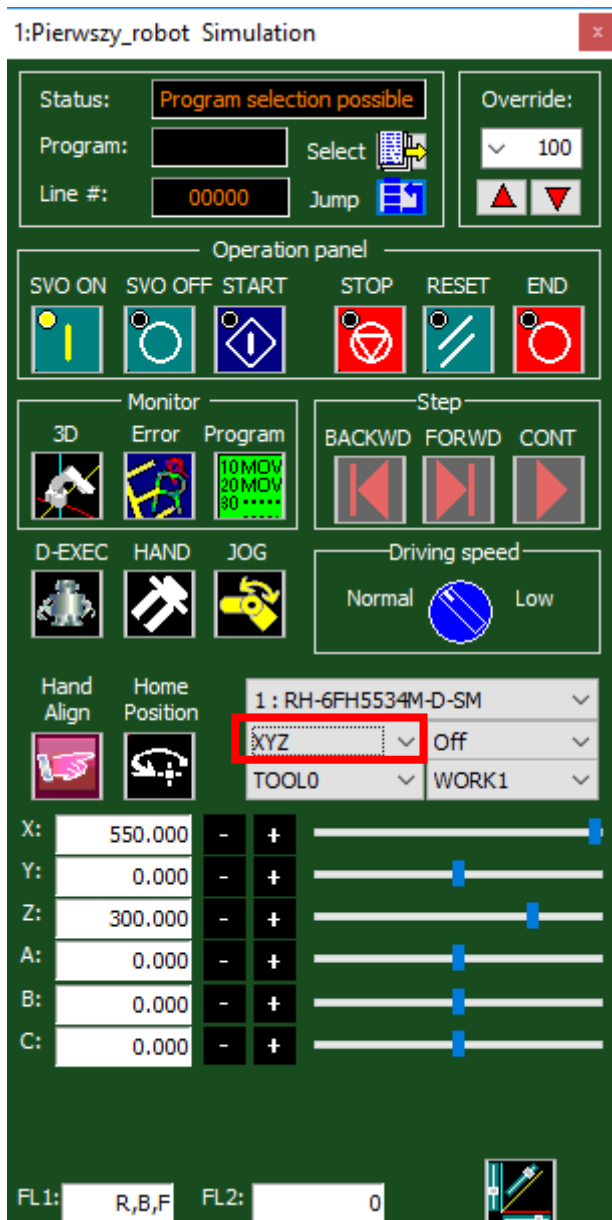
W programie należy wybrać z paska narzędzi jasnoniebieską ikonę „Simulation” oraz odczekać do momentu uruchomienia symulacji.



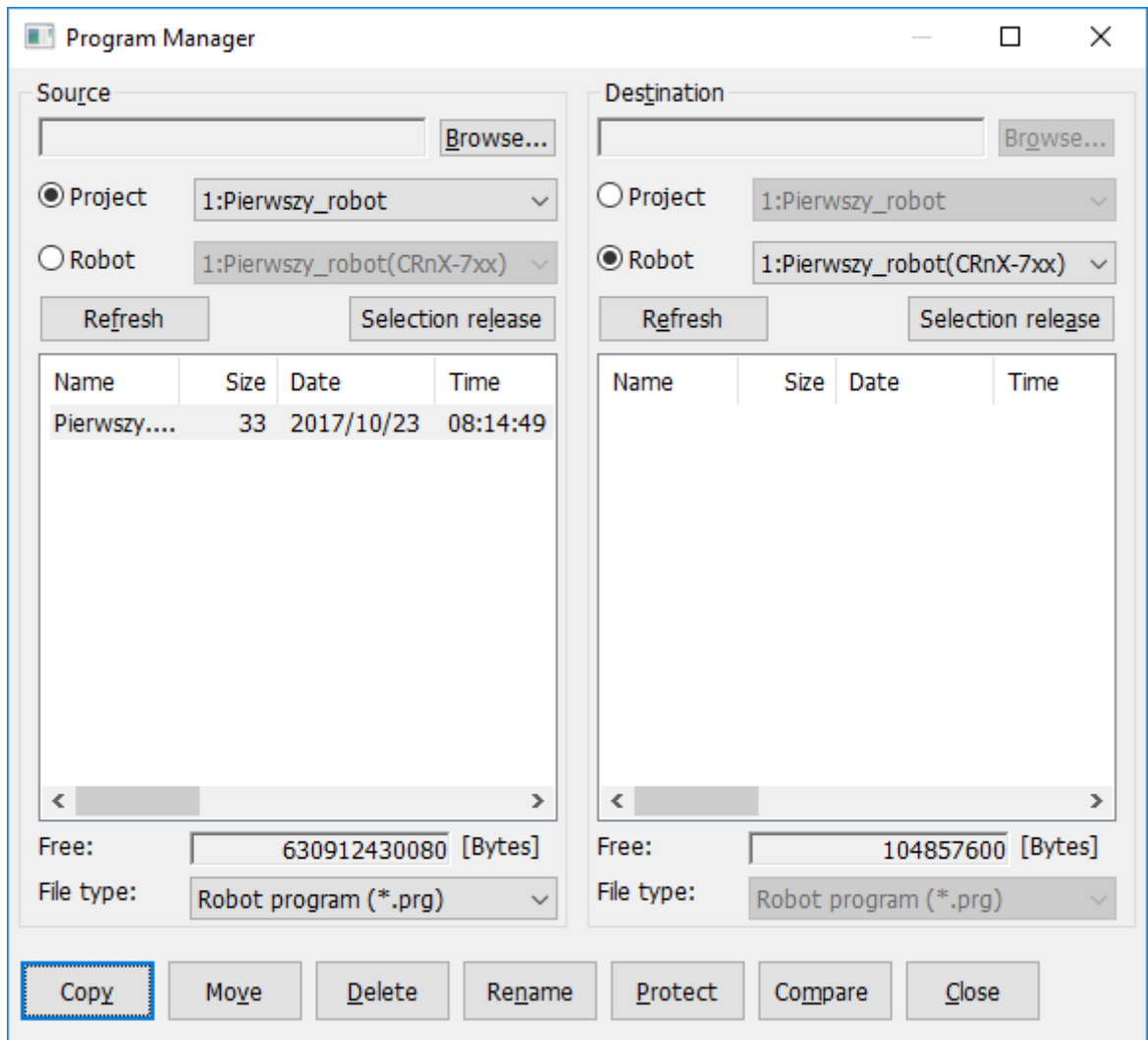
Po pojawieniu się panelu sterowania wybieramy opcję „JOG” umożliwiającą ręczne zadawanie położenia końcówki manipulatora.



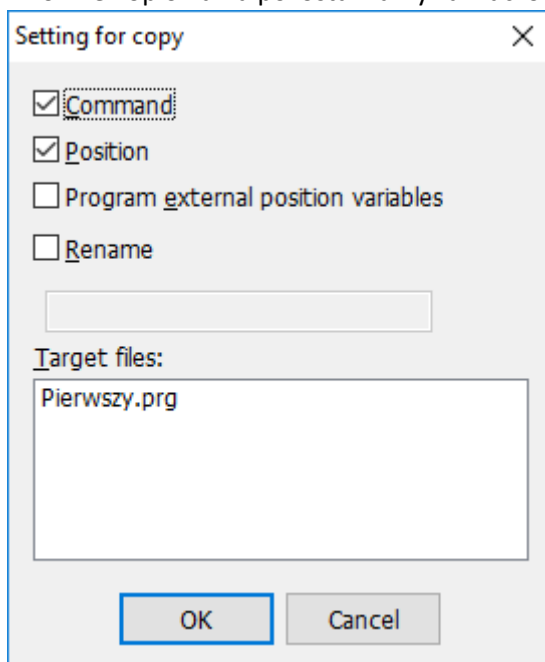
W rozwiniętej opcji domyślnie wybrane jest sterownie wartościami w przegubach „Joint”. Zmieniamy opcję na XYZ umożliwiającą podawanie współrzędnych w kartezjańskim układzie.



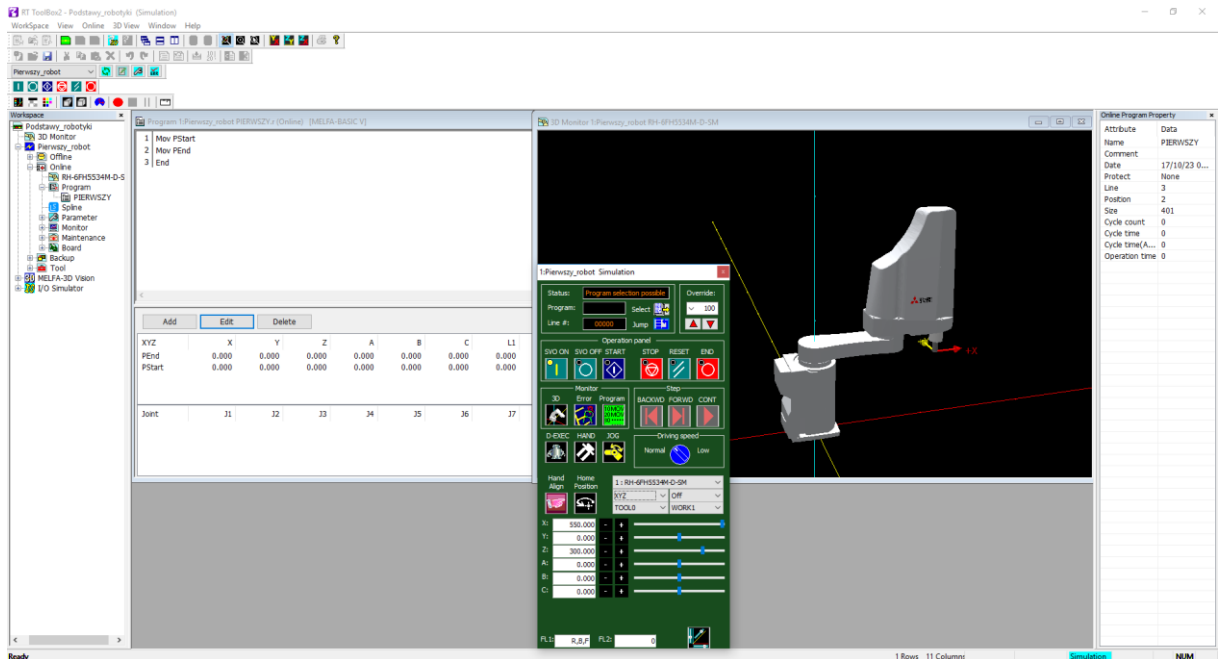
Następnie w drzewie projektu rozwijamy zakładkę „Online” oraz prawym przyciskiem myszy klikamy na „Program”. Z rozwiniętego menu wybieramy „Program Manager”. W nowym oknie sekcji „Source” wybieramy opcję „Project” oraz klikamy „Copy” w celu przekopiowania programu z trybu offline do symulacji.



W oknie kopiowania pozostawiamy zaznaczone dwie pierwsze pozycje.



Następnie w drzewie projektu wybieramy w zakładce „Online” okno programu klikając dwukrotnie na „Program”. Po otwarciu programu ustawiamy okna tak, aby mieć widoczność najważniejszych parametrów w każdym z nich.



Po ustaleniu współrzędnych punktu „PStart” w oknie symulacji zaznaczamy ten punkt w oknie programu oraz klikamy na przycisk „Edit”. W nowym oknie wybieramy „Get current position” oraz zatwierdzamy wybór „OK”.

**Uwaga!** Nie należy podawać pozycji skrajnych na największym zasięgu manipulatora. Może to skutkować wystąpieniem błędu symulatora.

Dialog box titled "Edit Position data" with a close button (X) in the top right corner.

**Name:** PStart

**Type:**

- XYZ
- Joint

X:	-36.130	<input checked="" type="checkbox"/>
Y:	-381.390	<input checked="" type="checkbox"/>
Z:	4.010	<input checked="" type="checkbox"/>
A:	0.000	<input checked="" type="checkbox"/>
B:	0.000	<input checked="" type="checkbox"/>
C:	-166.860	<input checked="" type="checkbox"/>
L1:	0.000	<input type="checkbox"/>
L2:	0.000	<input type="checkbox"/>

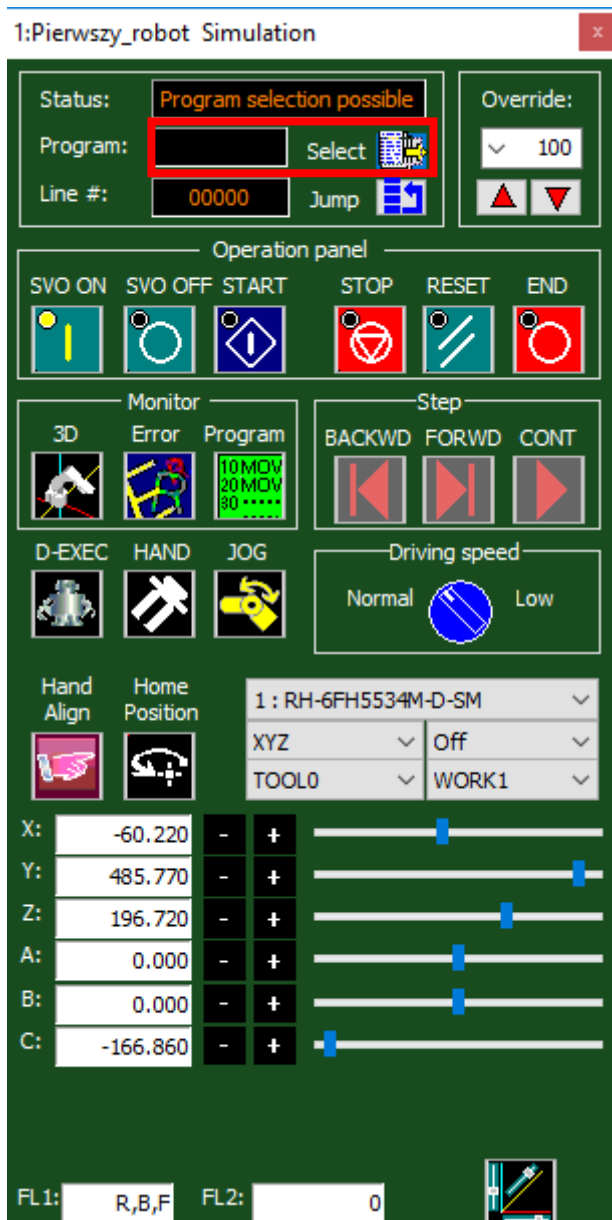
**Robot:** 1:RH-6FH5534M-D-SM

**FLG1:** R,B,F

**FLG2:** 0

Podobnie postępujemy dla punktu „PEnd”, a następnie zapisujemy odczytane punkty.

W panelu sterowania wczytujemy program „PIERWSZY”.



Teraz możemy uruchomić program klikając na niebieski przycisk START.

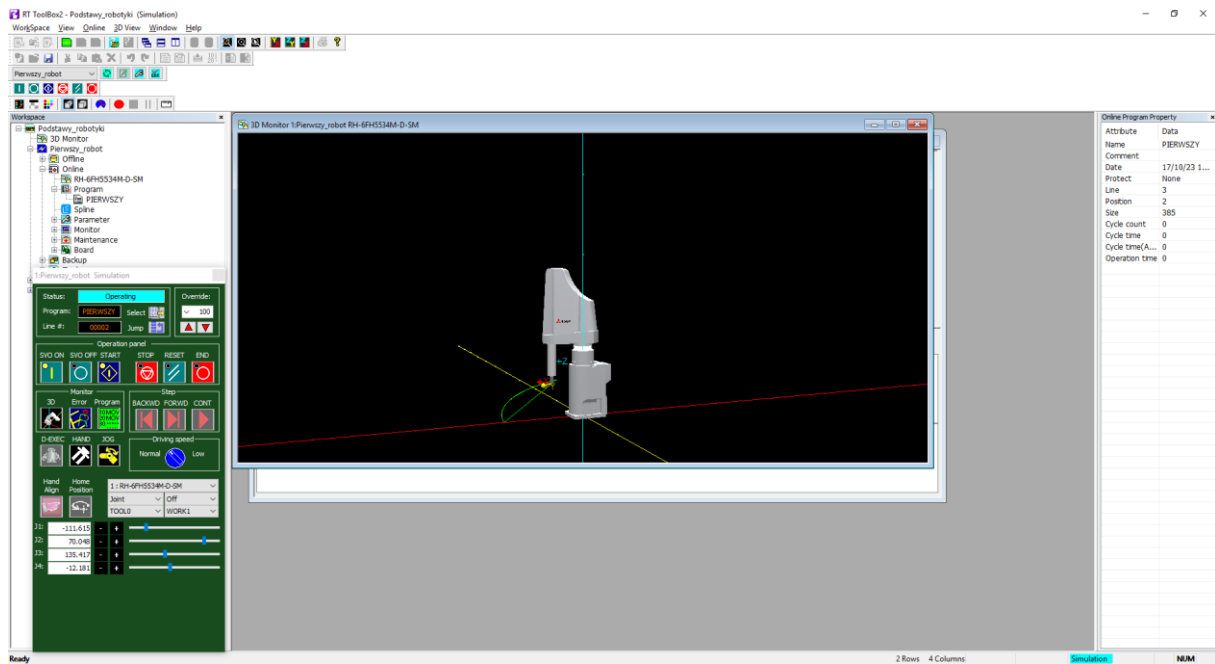
#### 4. Komendy programu

##### Zadanie 1:

Zmodyfikować powyższy kod, tak, aby punkt PStart znajdował się w pobliżu współrzędnych (495, 0, 0), natomiast punkt PEnd w pobliżu współrzędnych (0, 495, 150) – można skorzystać z punktów podpowiedź 1 oraz 2 (opisanych poniżej) lub wpisać wartości ręcznie. Następnie efektor powinien osiągać punkt PStart przy zastosowaniu komendy MOV, natomiast PEnd za pomocą komendy MVS.

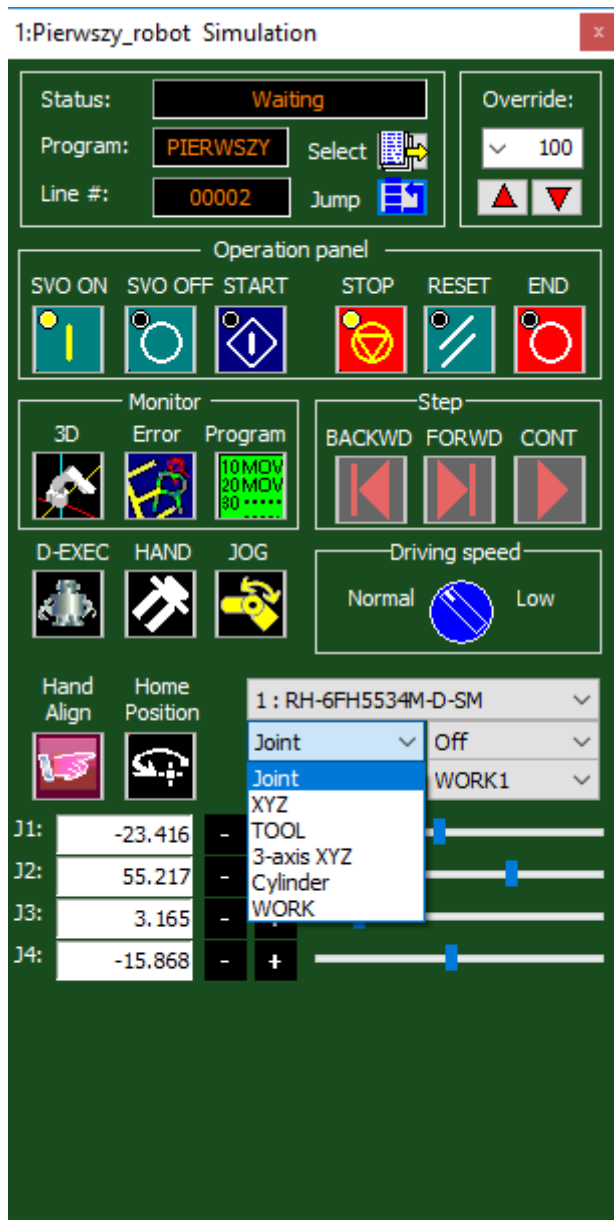
Należy włączyć rysowanie trajektorii efektora (patrz odpowiedź 3) oraz zapisać do sprawozdania zrzut ekranu z pracy manipulatora. W sprawozdaniu należy ponadto opisać różnice pomiędzy komendą MOV, a MVS. Przykładowy zrzut z poprawnie wykonanego programu poniżej:





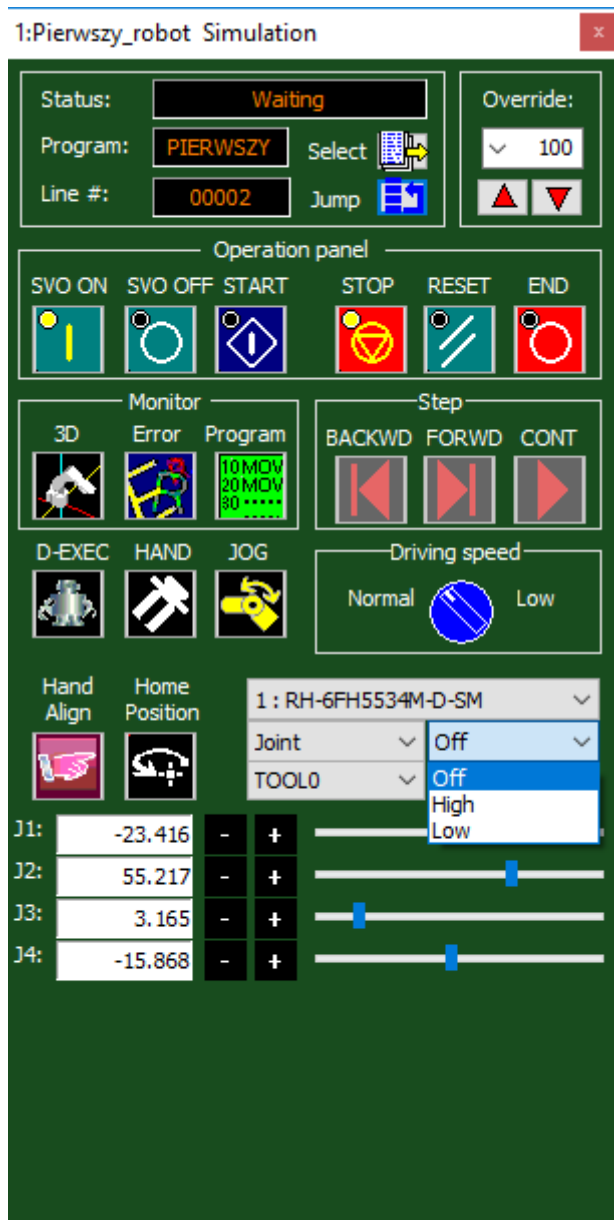
Podpowieź 1:

W celu osiągnięcia określonych współrzędnych przez efektor możliwa jest zmiana sterowania z opcji wskazywania współrzędnych kartezjańskich na sterowanie wartościami przegubów robota. W tym celu należy zmienić w panelu ręcznego sterowanie opcję „XYZ” przykładowo na „Joint” tak jak na obrazie poniżej.



Podpowiedź 2:

Aby dokonać bardziej dokładnego najazdu na interesujące współrzędne, można zwiększyć precyzję działania przycisków „-” oraz „+” dla każdego przegubu lub osi współrzędnych. Dokonuje się tego poprzez zmianę opcji precyzji „Off” na „High” lub jeszcze bardziej precyzyjne „Low” – jak na obrazie poniżej.



Podpowiedź 3:

Uruchomienie trajektorii ruchu efektora realizuje się wybierając z menu głównego programu następujące opcje:

3D View → Robot display option → Robot model → Robot path

Podstawowe komendy

Komenda	Przykład	Opis
MOV	MOV PHome	Ruch krzywoliniowy do punktu PHome
MVS	MVS PHome	Ruch prostoliniowy do punktu PHome
OVRD	Ovrd 50	Prędkość maksymalna robota w %

JOVRD	Jovrd 70	Ograniczenie prędkości do 70% wartości poprzedniej
SPD	Spd 100	Zmiana prędkości w interpolacji liniowej do 100mm/s
M_NSPD	M_Nspd	Przywraca domyślną prędkość ruchu z interpolacją liniową
M_NJOVRD	M_Njovrd	Przywraca domyślną prędkość ruchu z interpolacją przegubową
Dly	Dly 0.1	Odczekanie ok. 0.1 s
Wait	Wait m_svo=1	Odczekanie aż napędy zostaną włączone – komenda wait nie pozwala na przejście do kolejnej linii programu do momentu, aż warunek nie będzie spełniony
Fine P	Fine 0.1,P	Określenie tolerancji na poziomie 0.1 mm. Program nie wykona kolejnych instrukcji dopóki robot nie osiągnie pozycji z założoną tolerancją
Hopen/Hclose	Hopen 1/Hclose 1	Otwarcie/zamknięcie chwytaka
Servo	Servo on	Załączenie napędów
Loadset	Loadset 0,1	Przekazanie informacji do robota o tym, że w narzędziu nr 0 znajduje się przedmiot nr 1 o znanych parametrach
Gosub	Gosub *podprogram1	Skok do podprogramu
Return	Return	Kończy podprogram
End	End	Kończy program
'	' ruch do punktu początkowego	komentarz

## 5. Otoczenie

W celu dodania w otoczeniu robota innych obiektów 3D należy z menu głównego programu wybrać kolejno:

3D View → Layout → Add

W otwartym oknie istnieje możliwość wyboru elementarnej bryły (prostokątów, walec, kula) lub wczytać obiekt 3D przykładowo z rozszerzeniem \*.stl. Dla każdego obiektu można ustalić jego wymiary, pozycję oraz obrót w układzie współrzędnych.

Istnieje możliwość dodanie chwytaka do manipulatora, jednak nie będzie on wykonywał ruchów polegających na łapaniu oraz puszczeniu elementów, a jedynie będzie zawsze znajdował się w punkcie efektora manipulatora. Dodanie chwytaka odbywa się poprzez wybranie z menu głównego programu kolejno:

3D View → Hand → Add → Existing

Przykładowe obiekty 3D, które można dodać do programu zostały znajdują się w folderze Sample. Ścieżka dostępu dla modeli chwytaków:

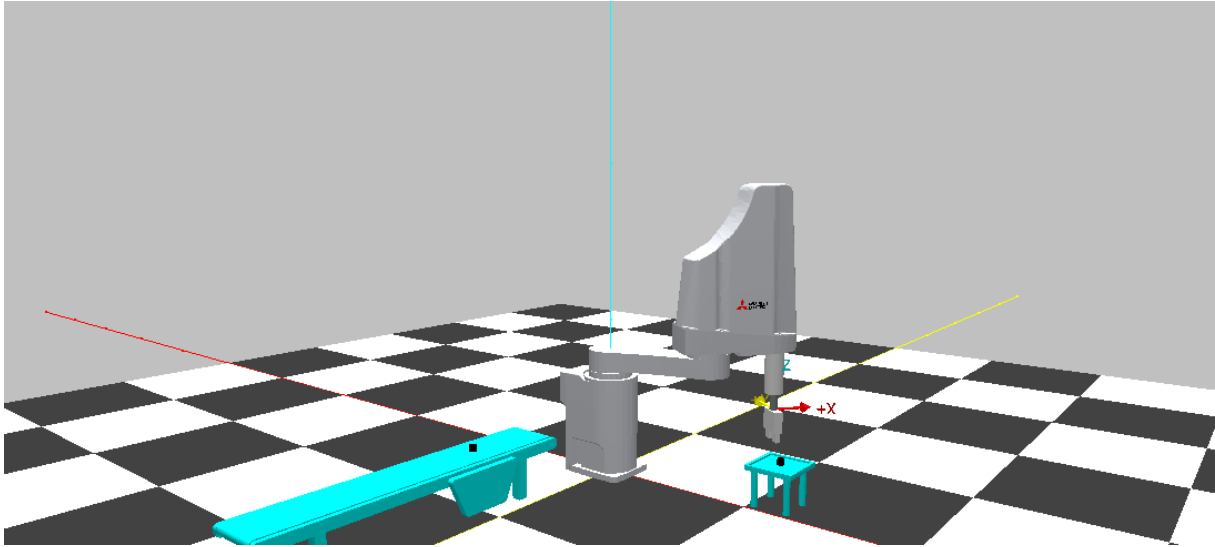
ToolBox2\_ENG\_3.60N/Sample/Hand/

Natomiast obiekty 3D znajdują się w folderze:

ToolBox2\_ENG\_3.60N/Sample/3DModel/

### Zadanie 2:

Wykorzystując modele taśmociągu (Convair.stl), stołu (Table.stl) oraz obiekt walca, utworzyć model jak na poniższym obrazie i napisać program według wytycznych podanych w dalszej części zadania.



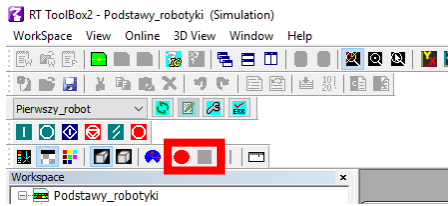
Wytyczne dotyczące otoczenia robota:

- Manipulator posiada chwytak
- Taśmociąg oraz stół znajdują się w niesąsiadujących ze sobą ćwiartkach globalnego układu współrzędnych (np. w I i III).
- Na obiektach 3D znajduje się model walca wyznaczający miejsce pobrania i odłożenia elementu przez uchwyt manipulatora. Wielkość walca powinna mieścić się w uchwytach chwytaka.
- Wysokość pomiędzy powierzchnia taśmociągu i stołu są różne.

Wytyczne dotyczące programu:

- Manipulator najeżdża nad element znajdujący się na taśmociągu (tak, aby nie strącić go chwytakiem), obniża chwytak i „bierze” element. Powoli przenosi go nad stół i opuszcza. Po otwarciu chwytaka podnosi go do góry (aby nie strącić elementu - ponownie).
- Ruch powrotny nad taśmociąg odbywa się szybciej niż w przeciwnym kierunku.
- Przed zamknięciem lub po otwarciu chwytaka należy odczekać ok. 0.2 sekundy czasu.

W sprawozdaniu wymagane jest zamieszczenie programu, opisanie jego działania oraz dołączenie zrzutu ekranu i nagrany proces pracy manipulatora. Nagrania dokonuje się poprzez wybranie przycisków „REC” oraz „Stop” na pasku narzędzi programu:



## 6. Instrukcje w kodzie sterującym

W programie sterującym możliwe jest stosowanie zmiennych, pętli oraz wyrażeń warunkowych. Poniższa tabela przedstawia wybrane z nich:

Komenda	Opis	Budowa komendy	Przykład
IF	Funkcja warunkowa. Jeżeli spełniony jest warunek, to wykonywany jest [proces_1], w przeciwnym wypadku [proces_2]	IF [warunek] THEN [proces_1] ELSE [proces_2]	IF M1=1 THEN MOV P1 MOV P2 ELSE MOV P1 MOV P3 ENDIF
FOR	Pętla, w której zmienna [licznik] jest zwiększana od wartości [w_początkowa] do wartości [w_koncowa] opcjonalnie co określoną wartość [skok]	FOR [licznik]=[w_początkowa] TO [w_koncowa] STEP [skok] [proces] NEXT	FOR M1=0 TO 10 P1.X=P1.X + M1 MOV P1 NEXT
WHILE	Pętla, która wykonuje [proces] dopóki spełniony pozostaje [warunek]	WHILE [warunek] [proces] WEND	WHILE M1<10 MOV ,+50 M!=M1+1 WEND
GOTO	Skok bezwarunkowy do określonej etykiety lub numeru linii [etykieta_lub_linia] w programie	GOTO [etykieta_lub_linia]	... 12 GOTO 24 ... 24 MOV P1
GOSUB	Skok bezwarunkowy do określonej etykiety lub numeru linii [etykieta_lub_linia] w programie. Po napotkaniu komendy RETURN nastąpi powrót do kolejnej linii po GOSUB	GOSUB [etykieta_lub_linia]	... GOSUB etykieta MOV P2 ... *etykieta ... MOV P1 ... RETURN

Poza opisanymi w powyższej tabeli komendami możliwy jest również skok do innego programu realizowany przez funkcję CALLP. Został on przedstawiony na poniższym przykładzie.

Należy utworzyć program główny o nazwie „Sterowanie” oraz o kodzie:

```

1  Krawedz = 50
2  PRuch = PStart
3  Mov PStart
4  GoSub *DolnaSciana
5  PRuch.Z = PRuch.Z + Krawedz
6  Mvs PRuch
7  GoSub *GornaSciana
8  Hlt
9  End
10 *DolnaSciana
11     PP_1 = PRuch
12     PP_1.X = PP_1.X + Krawedz
13     Mvs PP_1
14     PP_1.Y = PP_1.Y + Krawedz
15     Mvs PP_1
16     PP_1.X = PP_1.X - Krawedz
17     Mvs PP_1
18     Mvs PStart
19     Return
20 *GornaSciana
21     For M1 = 1 To 4
22         Select M1
23             Case 1
24                 PP_1 = PRuch
25                 PP_1.X = PP_1.X + Krawedz
26                 Mvs PP_1
27                 CALLP "Boczna", PP_1, Krawedz
28                 Break
29             Case 2
30                 PP_1 = PRuch
31                 PP_1.X = PP_1.X + Krawedz
32                 PP_1.Y = PP_1.Y + Krawedz
33                 Mvs PP_1
34                 CALLP "Boczna", PP_1, Krawedz
35                 Break
36             Case 3
37                 PP_1 = PRuch
38                 PP_1.Y = PP_1.Y + Krawedz
39                 Mvs PP_1
40                 CALLP "Boczna", PP_1, Krawedz
41                 Break
42             Case 4
43                 Mvs PRuch
44                 Break
45         End Select
46     Next
47     Return

```

Ponadto należy utworzyć program, do którego występują odwołania w powyższym kodzie. Będzie on posiadał nazwę „Boczna” oraz poniższy kod:

1	Fprm PP_1, Krawedz
2	PP_1.Z = PP_1.Z - Krawedz
3	Mvs PP_1
4	PP_1.Z = PP_1.Z + Krawedz
5	Mvs PP_1
6	End

Przed uruchomieniem programu „Sterowanie” należy podać współrzędne punktu PStart od którego robot rozpocznie rysowanie sześcianu. Przykładowe współrzędne punktu PStart to (450, 0, 0).

Zaprezentowany powyżej program jest odpowiedzialny za ruch efektora po wszystkich krawędziach sześcianu o długości boku zadeklarowanej w pierwszej linii pod zmienną "Krawedz". Następnie w linii 4 następuje przekierowanie wykonywania programu do etykiety o nazwie "DolnaSciana". Jest ona zadeklarowana w linii 10 oraz odpowiada za ruch efektora wzdłuż krawędzi podstawy sześcianu. Ruch ten realizowany jest przez zmianę współrzędnych X oraz Y punktu PP\_1, a następnie wykonywany jest ruch prostoliniowy do odpowiednio określonych wartości tego punktu. Po zakończeniu ruchu wzdłuż dolnych krawędzi i po osiągnięciu linii 19 programu, w której występuje komenda RETURN, następuje powrót do linii 5. Następnie modyfikowana jest współrzędna Z punktu PRuch i wykonywany jest ruch prostoliniowy, co skutkuje przemieszczeniem efektora wzdłuż pierwszej krawędzi bocznej sześcianu. W następnym kroku wykonywanie programu zostaje przeniesione do linii 20, gdzie zdefiniowana jest etykieta „GornaSciana”. W niej pętla FOR inkrementuje wartość zmiennej M1. Wewnątrz tej pętli wykonywana jest instrukcja warunkowa SELECT (w budowie podobna do instrukcji wyboru switch z języka C). Dla każdej wartości zmiennej M1 z przedziału od 1 do 4 warunek CASE odpowiada za narysowanie kolejnych krawędzi ściany górnej oraz za pomocą instrukcji CALLP wykonywany jest podprogram o nazwie „Boczna”, do którego jako parametry zostają przekazane wartości zmiennych PP\_1 oraz Krawedz. W pierwszej linii tego podprogramu (zdefiniowanego w pliku o tej samej nazwie) występuje instrukcja FPRM, która odpowiada za wczytanie przekazanych do niego zmiennych. Jej pominięcie uniemożliwi korzystanie z przekazanych wartości. Ponadto prosty w swojej budowie podprogram odpowiada za ruch efektora równoległy do osi Z wzdłuż krawędzi bocznych sześcianu oraz kończy się komendą END.

Po wykonaniu komendy RETURN z linii 47 programu głównego następuje powrót do linii nr 8. Komenda HLT odpowiada za zatrzymanie ramienia robota i czeka na kolejne naciśnięcie przycisku „Start”.

Wykonując opisany powyżej program poglądowy zaleca się włączyć opcji rysowania trajektorii ruchu efektora.



### Zadanie 3:

Wykorzystując layout stworzony w zadaniu 2 (można zmienić położenie modeli 3D) należy napisać program który z 10 klocków znajdujących się na taśmie produkcyjnej ułoży wieżę na stole. Manipulator powinien podnosić kolejne klocki z taśmy oraz odkładać je na odpowiedniej wysokości na stole. Przy czym powinny zostać zadeklarowane tylko dwa punkty:

- Współrzędne pierwszego klocka na linii produkcyjnej
- Współrzędne pierwszego (dolnego klocka) wieży.

Ponadto w programie określona powinna zostać odległość pomiędzy kolejnymi klockami na linii produkcyjnej, które będą znajdowały się w linii prostej.

Symulacja powinna zawierać 10 modeli klocków na linii produkcyjnej (miejsca z których manipulator będzie pobierał klocki) oraz wieżę na stole składającą się z 10 na przemian kolorowanych klocków. Budowa wieży pozwoli na określenie prawidłowości wysokości skrajnych położenia manipulatora podczas „upuszczania” kolejnych klocków, pomimo uproszczonego modelu symulacji, w którym nastąpi przenikanie uchwytu przez klocki, które jeszcze nie powinny znajdować się na wieży.

Wytyczne dotyczące programu oraz opisu w sprawozdaniu są takie same jak w zadaniu 2.

### Zadanie 4:

Wykorzystując modele 3D stołu, obrabiarki CNC oraz obiekt walca, utworzyć layout symulacji zawierający obrabiarkę CNC oraz dwa stoły wraz ze znajdującymi się w ich rogach czterema walcami stojącymi pionowo. Następnie dobrać z listy dostępnych manipulatorów model posiadający minimum 6 stopni swobody, który będzie mógł zrealizować niżej opisany scenariusz:

- Robot chwyta na pierwszym stole model walca od jego boku, przekręca, przenosi w pozycji poziomej nad obszar roboczy obrabiarki oraz wsuwa w jej uchwyt.
- Odjeżdża z obszaru roboczego obrabiarki i czeka 2 sekundy
- Chwyta element znajdujący się w uchwycie obrabiarki, wysuwa go oraz przenosi w jeden z rogów drugiego stołu
- Wraca po kolejny model walca na pierwszym stole i powtarza operacje dla wszystkich modeli.

Ruch związany z pobraniem i odłożeniem modelu walca na stole powinien być tak realizowany, aby robot nie strącił innych modeli, które w aktualnej fazie programu powinny tam pozostawać.

Po wykonaniu sekwencji ruchów dla 4 modeli, manipulator powinien się zatrzymać.

W programie należy zdefiniować punkty odpowiedzialne za określenie pozycji w globalnym układzie współrzędnych środków stołów oraz uchwytu obrabiarki.

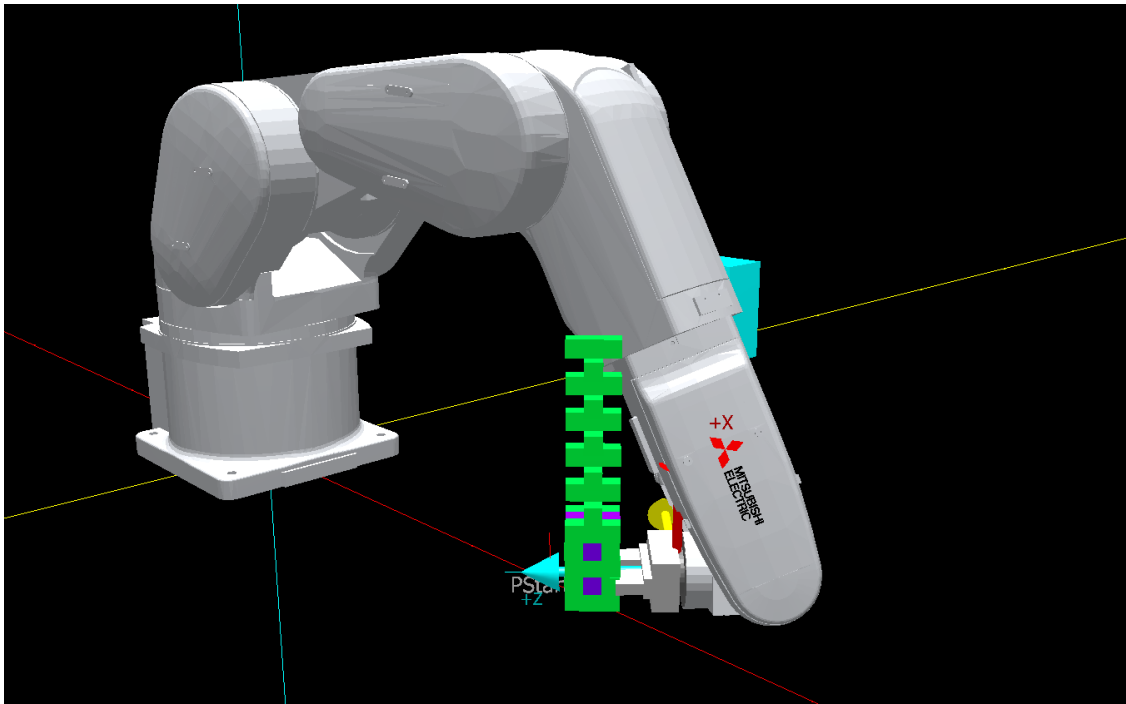
W sprawozdaniu powinny znaleźć się zrzuty ekranu z procesu wykonywania programu, kod sterowania manipulatorem, opis kodu oraz nagranie symulacji obejmujące pełny proces (dla 4 walców)

### Zadanie 5:

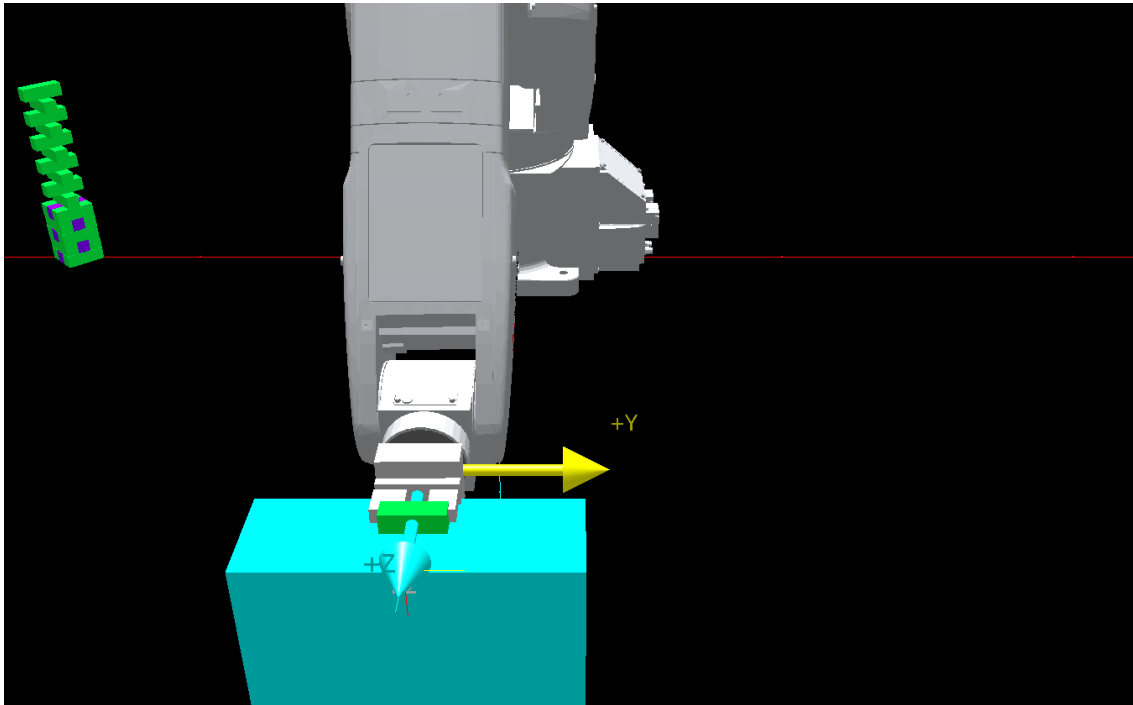
Dla wybranego manipulatora posiadającego minimum 7 stopni swobody napisać program, który zrealizuje ruchy robota w grze Jenga (przy założeniu, że wieża nigdy nie ulegnie zniszczeniu). Scenariusz powinien realizować:

- Wyciąganie kolejnych brzegowych klocków (dwa na każdym poziomie podstawowej wieży) ruchem prostopadłym do wieży chwytając za boczne ściany klocka (Rys. 1).
- Odłożenie wyciągniętych klocków z na dedykowanym podeście (Rys. 2).
- Zmiana sposobu trzymania klocka na „trzymanie od góry” (Rys. 3).
- Odłożenie klocka na szczycie wieży.
- Powrót do kolejnego klocka wyciąganego z podstawowej części wieży i wykonanie powyższych punktów od początku.

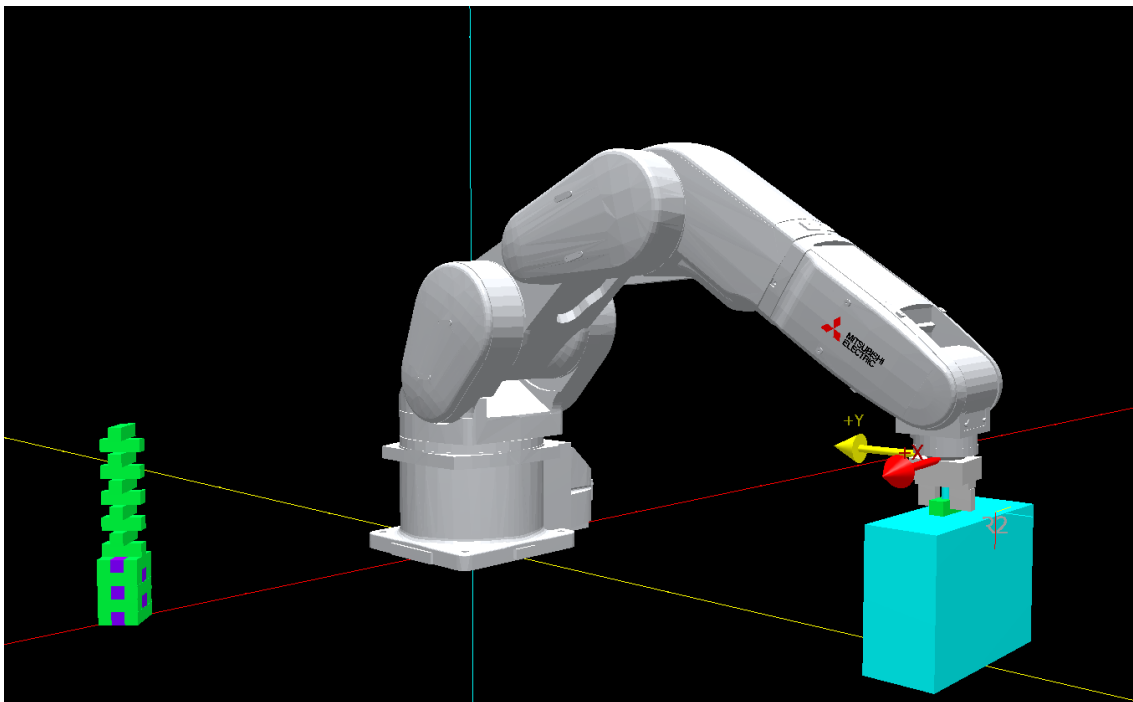
Wieża powinna być zbudowana z klocków w dwóch kolorach. Jeden odpowiadający za klocki wyciągane, drugi dla klocków pozostających w konstrukcji wieży po zwycięstwie. Przykładowe rozmiary klocka to 60x20x20 (X x Y x Z). Przykładowy chwytak zastosowany w manipulatorze to FT-2020 (należy przyjąć, że szczęki chwytaka otworzą się do rozmiaru umożliwiającego złapanie klocka). W rysunkach poniżej stosowano manipulator RV-7F-D.



Rysunek 1Faza pierwsza - wyciąganie klocka z podstawowej wieży



Rysunek 2 Faza druga - odłożenie wyciągniętego klocka



Rysunek 3 Faza trzecia - zmiana sposobu trzymania klocka

UWAGA: Sprawozdanie powinno zawierać:

- Na ocenę:
  - 3.0 poprawnie wykonane zadania 1-3 wraz z opisem,
  - 4.0 poprawnie wykonane zadania 1-4 wraz z opisem,
  - 5.0 poprawnie wykonane zadania 1-5 wraz z opisem.
- Sprawozdanie należy przesłać w formie elektronicznej na adres [dozog@kia.prz.edu.pl](mailto:dozog@kia.prz.edu.pl) jako załącznik do wiadomości e-mail. Temat wiadomości „PR Sprawozdanie II grupa LX”, gdzie X jest numerem grupy laboratoryjnej osób wykonujących sprawozdanie.
- Z racji na ograniczoną wielkość załączników do wiadomości e-mail, nagrania pracy robota można przesłać w formie linku do zewnętrznego źródła, w którym zostały zamieszczone lub przynieść na najbliższe zajęcia po przesłaniu sprawozdania oraz poinformować prowadzącego o zamiarze ich skopiowania na wskazany przez prowadzącego nośnik danych.
- Sprawozdanie wysyła 1 osoba z maksymalnie 3 osobowej grupy wykonującej sprawozdanie.
- Sprawozdanie powinno być wykonane na bazie formatki zamieszczonej na stronie z instrukcją.
- Jako poprawny sposób opisu zadania w sprawozdaniu uważa się zamieszczenie:
  - programu/programów odpowiedzialnych za sterowanie robota (w formie tekstu) oraz współrzędnych wykorzystywanych w programie punktów (wystarczy zrzut ekranu),
  - przykładowe zrzuty ekranu wykonane podczas wykonywania programu i pokazujące przygotowany Layout
  - szczegółowego opisu programu sterującego z wyjaśnieniem w jaki sposób zamieszczone w nim instrukcje rozwiązują problem
  - nagrania przynajmniej jednej pętli wykonywanego programu wykonanego przy użyciu funkcji Rec (nie dotyczy zadania 1)
- Czas nadsyłania prac: do godziny rozpoczęcia szóstych zajęć laboratoryjnych z przedmiotu „Podstawy robotyki” w semestrze (około 2 tygodnie)
- Opóźnienie w wysyłaniu sprawozdania o każdy rozpoczęty tydzień powoduje obniżenie oceny o 0,5 stopnia.