

دانشگاه صنعتی خواجه نصیرالدین طوسی
دانشکده مهندسی برق - گروه مهندسی کنترل

رباتیک

پاسخ پروژه اول

ایمان گندمی	نام و نام خانوادگی
۴۰۲۲۳۷۰۴	شماره دانشجویی
علیرضا امیری	نام و نام خانوادگی
۴۰۲۰۲۴۱۴	شماره دانشجویی
آبان ماه ۱۴۰۳	تاریخ



فهرست مطالب

۳	۱	سوال اول - مشخصات فنی ربات
۴	۲	سوال دوم - روش دناویت-هارتنبرگ
۴	۳	سوال سوم - روش پیچ
۶	۴	سوال چهارم - سینماتیک وارون
۶	۵	سوال پنجم - صحت سنجی با SolidWorks
۶	۱.۵	محاسبه ی عددی
۷	۲.۵	محاسبه ی هندسی
۷	۶	سوال ششم - شبیه سازی
۱۶	۷	سوال هفتم - فرایند ساخت ربات



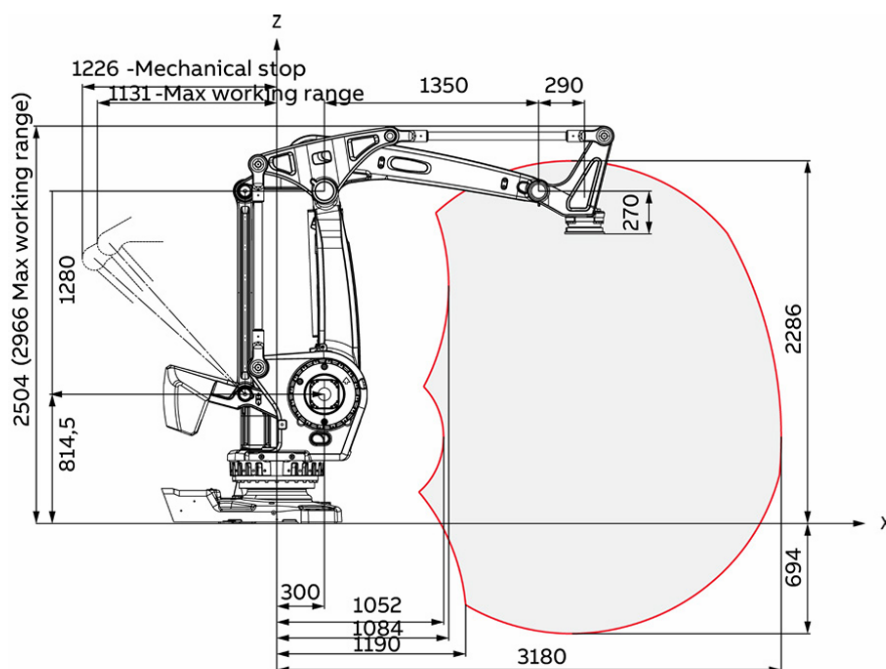
فهرست تصاویر

۱	فضای کاری ربات	۳
۲	محدودیت های مفاصل ربات	۴
۳	طرح رسم شده از ربات	۷
۴	طرح ربات در سالدورکس	۸
۵	طرح ربات در SimScape	۸
۶	اعمال ورودی به مدل ربات	۹
۷	مدل ربات در SimScape	۱۰
۸	نتیجه مدل، آزمایش ۱	۱۱
۹	نتیجه کد، آزمایش ۱	۱۲
۱۰	نتیجه مدل، آزمایش ۲	۱۳
۱۱	نتیجه کد، آزمایش ۲	۱۴
۱۲	نتیجه مدل، آزمایش ۳	۱۵
۱۳	نتیجه کد، آزمایش ۳	۱۶
۱۵	قطعه ۲	۱۷
۱۴	قطعه ۱	۱۷
۱۷	قطعه ۴	۱۷
۱۶	قطعه ۳	۱۷
۱۹	قطعه ۶	۱۸
۱۸	قطعه ۵	۱۸
۲۱	قطعه ۸	۱۸
۲۰	قطعه ۷	۱۸
۲۳	قطعه ۱۰	۱۹
۲۲	قطعه ۹	۱۹

۱ سوال اول - مشخصات فنی ربات

برای به دست آوردن مشخصات فنی ربات مورد استفاده در این پژوهش، با مراجعه به دیتاشیت ربات IRB ۷۶۰، می‌توان اطلاعات لازم از جمله ابعاد، محدودیت‌های کاری مفصل، فضای کاری، توان مصرفی، دمای کاری و سایر مشخصات مهم را استخراج کرد. این ربات چهار محوره با ظرفیت بار ۴۵۰ کیلوگرم و دسترسی ۱۸.۳ متر، برای کاربردهایی مانند پالت‌گذاری، دپالت‌گذاری، جابجایی مواد و پرس‌کاری طراحی شده است. ربات IRB ۷۶۰ با کنترلر OmniCore و نرم‌افزارهای QuickMoveTM و TrueMoveTM حرکات نرم و دقیقی را ارائه می‌دهد که برای جابجایی محصولات حساس نیز مناسب است.

این ربات دارای چهار محور است که به ترتیب در بازه‌های $\pm 180^\circ$ درجه (محور اول)، 85° تا 42° درجه (محور دوم)، 120° تا 20° درجه (محور سوم) و $\pm 300^\circ$ درجه (محور چهارم) حرکت می‌کنند و حداکثر سرعت محور چهارم به 160° درجه بر ثانیه می‌رسد. دقت تکرارپذیری آن 0.05 میلی‌متر و استاندارد حفاظت IP۶۷ را داراست که آن را برای محیط‌های گردوغبار و مرطوب مناسب می‌سازد. محدوده دمای کاری ربات بین 0° تا 50° درجه سانتی‌گراد است و برای دوره‌های کوتاه‌مدت تا 70° درجه را نیز تحمل می‌کند. با مصرف برق 75.2 کیلووات و وزن 2310 کیلوگرم، این ربات می‌تواند تا 88 چرخه در ساعت با بار کامل انجام دهد که برای کاربردهای سنگین صنعتی، به‌ویژه در خودروسازی، گزینه‌ای مناسب به شمار می‌رود.



IRB 760PT, working range

شکل ۱: فضای کاری ربات



IRB 760		
Axis movement	Working range	Axis max speed
Axis 1	+180° to -180°	85°/s
Axis 2	+85° to -42°	85°/s
Axis 3	+120° to -20°	85°/s
Axis 4*	+300° to -300°	160°/s

* +67 rev. to - 67 rev. max

شکل ۲: محدودیت های مفصل ربات

۲ سوال دوم - روش دناویت-هارتبرگ

برای حل مسئله ی سینماتیک مستقیم به روش دناویت-هارتبرگ، ابتدا پارامترهای ماتریس DH باید مشخص شوند. برای این منظور ماتریس های تبدیل را با استفاده از تابع تعریف شده چنین به دست می آوریم:

$$T_1 = \text{DH}(0, \frac{\pi}{2}, 0, \theta_1)$$

$$T_2 = \text{DH}(l_1, 0, 0, \theta_2)$$

$$T_3 = \text{DH}(l_5, 0, 0, \theta_3)$$

ماتریس تبدیل این جابه جایی از ضرب ماتریس های بالا به دست می آید. با جداسازی ماتریس دوران و انتقال از این ماتریس تبدیل در نهایت خواهیم داشت:

$$P_{\text{DH}} = \begin{pmatrix} 1.0 \cos(\theta_1) (l_5 \cos(\theta_2 + \theta_3) + l_1 \cos(\theta_2)) \\ 1.0 \sin(\theta_1) (l_5 \cos(\theta_2 + \theta_3) + l_1 \cos(\theta_2)) \\ 1.0 l_5 \sin(\theta_2 + \theta_3) + 1.0 l_1 \sin(\theta_2) \end{pmatrix}$$

$$R_{\text{DH}} = \begin{pmatrix} 1.0 \cos(\theta_2 + \theta_3) \cos(\theta_1) & -1.0 \sin(\theta_2 + \theta_3) \cos(\theta_1) & 1.0 \sin(\theta_1) \\ 1.0 \cos(\theta_2 + \theta_3) \sin(\theta_1) & -1.0 \sin(\theta_2 + \theta_3) \sin(\theta_1) & -1.0 \cos(\theta_1) \\ 1.0 \sin(\theta_2 + \theta_3) & 1.0 \cos(\theta_2 + \theta_3) & 0 \end{pmatrix}$$

۳ سوال سوم - روش پیچه

حل مسئله ی سینماتیک مستقیم این ربات، نیازمند مشخص کردن محورهای پیچه می باشد. در ادامه، با استفاده از موقعیت و جهت بردارها، می توان پارامترهای پیچه را استخراج کرد. با در نظر گرفتن ساختار این ربات، ماتریس های تبدیل در هر مفصل به صورت زیر به دست می آیند:

$$S_1 = SR(0, 0, 1, 0, 0, 0, \theta_1, 0);$$

$$S_2 = SR(0, -1, 0, 0, 0, 0, \theta_2, 0);$$

$$S_3 = SR(0, -1, 0, l_1, 0, 0, \theta_3, 0);$$

حاصل ضرب این ماتریس ها به طور مستقیم، ماتریس تبدیل نهایی را به وجود نمی آورد و نیاز است در گام بعد، ماتریس دوران و انتقال بین مختصات مرجع و مختصات پنجه ی ربات نیز لحاظ شود. در این قسمت، حاصل ضرب ماتریس های تبدیل بالا چنین به دست می آیند:

$$S = S_1 \cdot S_2 \cdot S_3 = \begin{pmatrix} \cos(\theta_2 + \theta_3) \cos(\theta_1) & -\sin(\theta_1) & -\sin(\theta_2 + \theta_3) \cos(\theta_1) & -l_1 \cos(\theta_1) \sigma_1 \\ \cos(\theta_2 + \theta_3) \sin(\theta_1) & \cos(\theta_1) & -\sin(\theta_2 + \theta_3) \sin(\theta_1) & -l_1 \sin(\theta_1) \sigma_1 \\ \sin(\theta_2 + \theta_3) & 0 & \cos(\theta_2 + \theta_3) & -l_1 (\sin(\theta_2 + \theta_3) - \sin(\theta_2)) \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

where

$$\sigma_1 = \cos(\theta_2 + \theta_3) - \cos(\theta_2)$$

حال برای تعریف ماتریس تبدیل از دستگاه مختصات مرجع به مبدا خواهیم داشت:

$$S_O = \begin{pmatrix} 1 & 0 & 0 & l_1 + l_5 \\ 0 & 0 & -1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$

در نهایت، ماتریس تبدیل برای این ربات از ضرب ماتریس S در ماتریس تبدیل دستگاه های مختصات مرجع و مبدا به محاسبه می شود. پس از جداسازی ماتریس دوران و انتقال این ماتریس تبدیل خواهیم داشت:

$$P_S = \begin{pmatrix} \cos(\theta_1) (l_5 \cos(\theta_2 + \theta_3) + l_1 \cos(\theta_2)) \\ \sin(\theta_1) (l_5 \cos(\theta_2 + \theta_3) + l_1 \cos(\theta_2)) \\ l_5 \sin(\theta_2 + \theta_3) + l_1 \sin(\theta_2) \end{pmatrix}$$

$$R_S = \begin{pmatrix} \cos(\theta_2 + \theta_3) \cos(\theta_1) & -\sin(\theta_2 + \theta_3) \cos(\theta_1) & \sin(\theta_1) \\ \cos(\theta_2 + \theta_3) \sin(\theta_1) & -\sin(\theta_2 + \theta_3) \sin(\theta_1) & -\cos(\theta_1) \\ \sin(\theta_2 + \theta_3) & \cos(\theta_2 + \theta_3) & 0 \end{pmatrix}$$

در نهایت، از مقایسه ی ماتریس های تبدیل به دست آمده از دو روش دناویت هارتنبرگ و پیچه، می توانیم صحت محاسبات انجام شده را به دست آوریم. برای این منظور، با محاسبه ی تفاضل ماتریس های به دست آمده از هر دو روش خواهیم داشت:

$$\text{Error} = T_{DH} - T_S = \begin{pmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

در نتیجه، صحت جواب های به دست آمده از این دو روش اثبات می شوند.



۴ سوال چهارم - سینماتیک وارون

در این بخش، با محاسبه ی روابط هندسی حاکم بر فضای سه بعدی، با دریافت ورودی های موقعیت ابزار و با معین بودن اندازه ی لینک ها خواهیم داشت:

$$\begin{aligned}\theta_1 &= \arctan\left(\frac{y}{x}\right) \\ r &= \sqrt{x^2 + y^2} \\ d &= \sqrt{r^2 + z^2} \\ \theta_3 &= \arccos\left(\frac{d^2 - l_1^2 - l_5^2}{2 \cdot l_1 \cdot l_5}\right) \\ \text{angle Total} &= \arctan\left(\frac{z}{r}\right) \\ \theta_2 &= \text{angle Total} - \theta_3\end{aligned}$$

با استفاده از روابط بالا، می توان زوایای مفصلی متناظر با هر موقعیت ابزار را به دست آورد.

۵ سوال پنجم - صحت سنجی با SolidWorks

در این بخش، ابتدا با جایگذاری مقادیر عددی داده شده مطابق با داده های سوال و محاسبه ی موقعیت ابزار ربات به وسیله ی ماتریس های تبدیل به دست آمده از دروش دناویت هارتبرگ و یا پیچه، پاسخی را به دست می آوریم. در ادامه با طراحی هندسی ربات و تنظیم زوایای آن، مختصات نقطه ی ابزار را اندازه می گیریم.

۱.۵ محاسبه ی عددی

پیش از آنکه مقادیر زاویه ها را در ماتریس های تبدیل قرار دهیم، لازم است ابتدا زوایای موجود در فایل طراحی ربات به زوایای مورد استفاده در ماتریس تبدیل تغییر کنند. برای این منظور، مقادیر θ_1^* ، θ_2^* ، θ_3^* که زوایای داده شده در فایل طراحی هستند با رابطه ی زیر تبدیل به روابط مورد استفاده در ماتریس تبدیل می شوند:

$$\theta_1 = \theta_1^*, \quad \theta_2 = \theta_3^*, \quad \theta_3 = \pi - \theta_3^* + \theta_2^*$$

با جایگذاری مقادیر داده شده در این رابطه خواهیم داشت:

$$\theta_1^* = 0, \quad \theta_2^* = \frac{\pi}{4}, \quad \theta_3^* = \frac{3\pi}{4} \implies \theta_1 = 0, \quad \theta_2 = \frac{3\pi}{4}, \quad \theta_3 = \frac{\pi}{4}$$

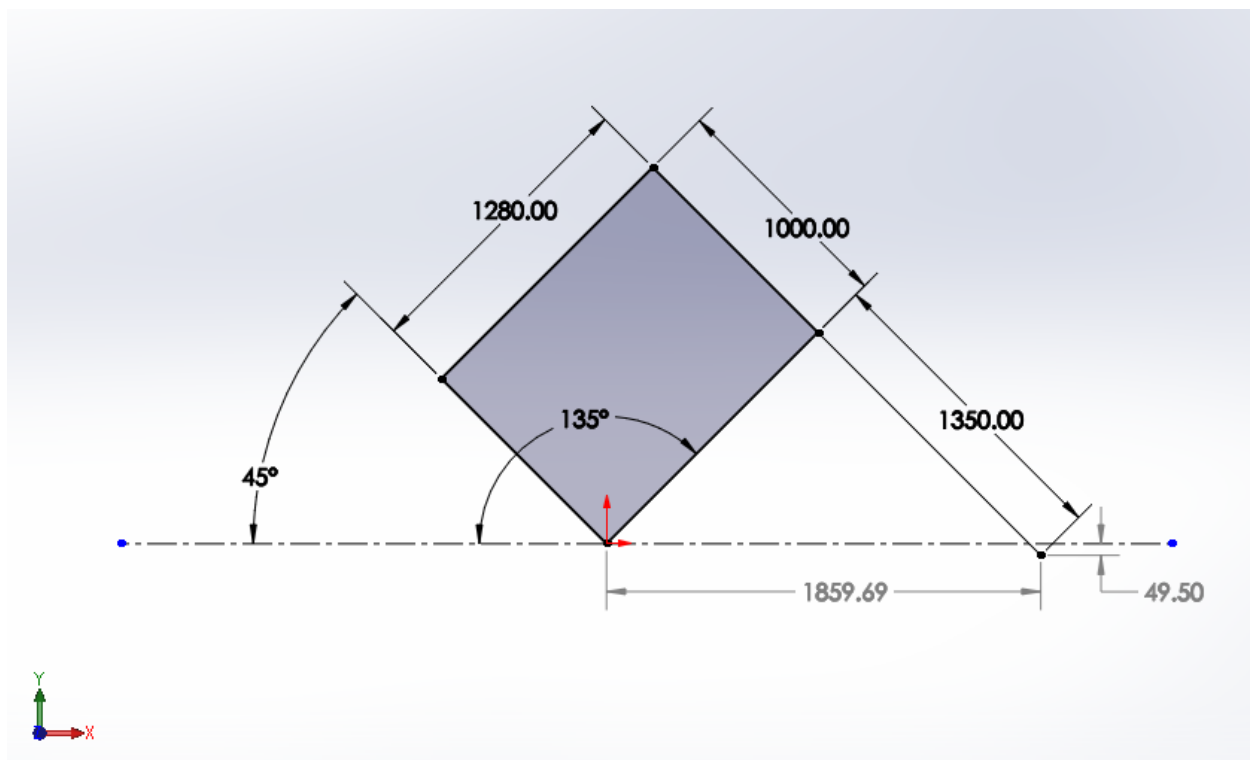
با استفاده از این مقادیر و محاسبه ی ماتریس تبدیل و حل مسئله ی سینماتیک مستقیم، موقعیت و جهت گیری نهایی ابزار به صورت زیر به دست می آید:

$$P_{\text{end-effector}} = \begin{pmatrix} -1.8597 \\ 0 \\ -0.0495 \end{pmatrix}$$

$$R_{\text{end-effector}} = \begin{pmatrix} -0.7071 & 0.7071 & 0 \\ 0 & 0 & -1.0000 \\ -0.7071 & -0.7071 & 0 \end{pmatrix}$$

۲.۵ محاسبه ی هندسی

در این روش، با رسم طرح ساده شده ی ربات و تنظیم اندازه ی لینک ها، محدودیت های اعمال شده بر آن و همچنین تنظیم زوایا بر اساس مقادیر داده شده، موقعیت نهایی ابزار را به دست می آوریم.

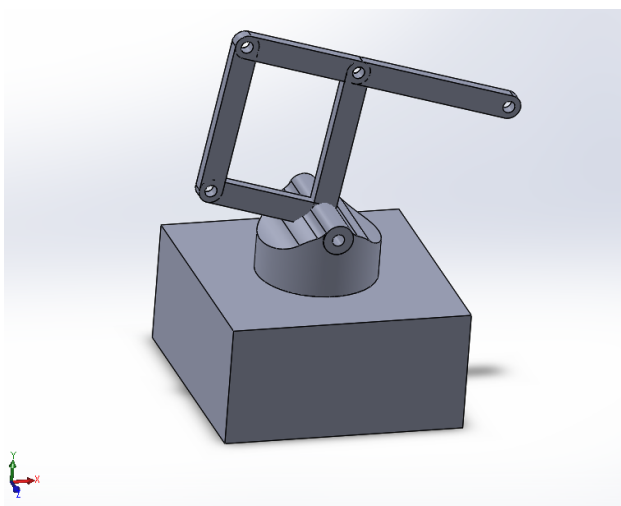


شکل ۳: طرح رسم شده از ربات

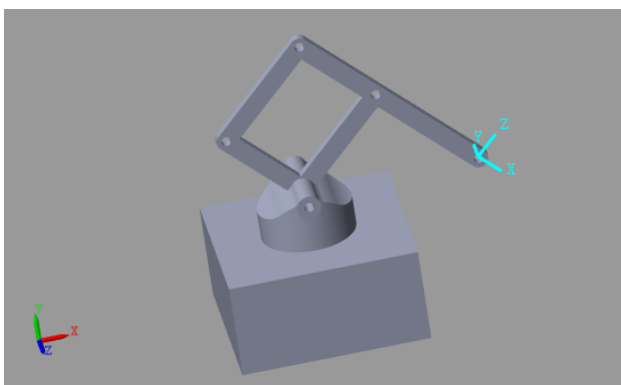
مشاهده می شود که موقعیت به دست آمده برای ابزار ربات در این روش، با موقعیت محاسبه شده به وسیله ی کد مطابقت دارد.

۶ سوال ششم - شبیه سازی

برای شبیه سازی عملکرد ربات مورد استفاده در این پژوهش، ساختار ربات به نرم افزار SimScape وارد شده است.



شکل ۴: طرح ربات در سالیدورکس

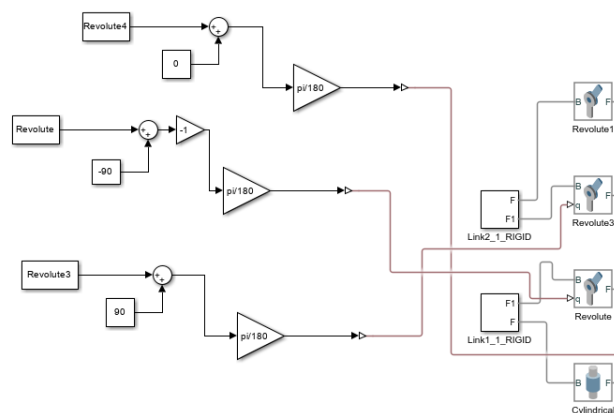


شکل ۵: طرح ربات در SimScape

پس از وارد کردن طراحی ربات به این نرم افزار، ساختار کلی مفاصل آن توسط سیمولینک ساخته می شود. برای اعمال ورودی های مختلف به این دستگاه و اندازه گیری موقعیت ابزار، لازم است بلوک های دیگری نیز به آن اضافه شوند که در ادامه به توضیح هر یک خواهیم پرداخت. برای اعمال ورودی زوایای مناسب به ربات، لازم است تغییراتی بر آن اعمال شود. در گام اول، برای یکسان سازی زاویه ی ۰ هر یک از محورهای دوران، با مقایسه ی طرح داده شده ی ربات و نتایج شبیه سازی، تغییرات زاویه برای هر یک از محور های دوران به دست آمد. تغییر زوایا با اضافه کردن مقدار ثابتی به هر یک از محورهای دوران صورت می گیرد. در شکل ۶ این بلوک ها نمایش داده شده اند. در نهایت، با ضرب مقدار عددی ثابت، واحد زوایای وارد شده به رادیان تبدیل می شود. پیش از اعمال ورودی ها به مدل، لازم است زوایای مورد استفاده در کد برنامه نویسی شده با مفاصل ایجاد شده در مدل ربات نگاشت شوند. بنابراین، در مدل استفاده شد از نگاشت های زیر برای تبدیل این زوایا استفاده می کنیم.

$$\text{Rev}4 = \theta_1$$

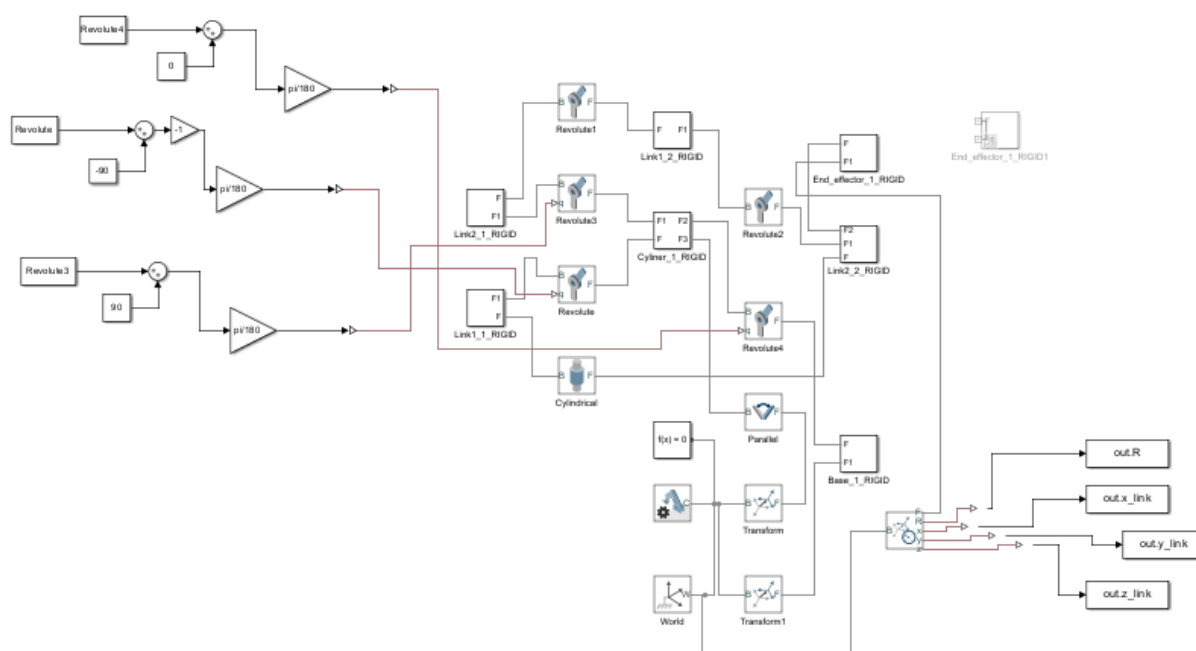
$$\text{Rev} = (\theta_3 - 90^\circ) \cdot (-1)$$



شکل ۶: اعمال ورودی به مدل ربات

$$\text{Rev}^3 = \theta_2 + 90^\circ$$

در ادامه، پس از تنظیم موقعیت دستگاه مختصات ربات بر روی ابزار آن و خاموش کردن وضعیت گرانش، مدل را ذخیره می‌کنیم. بخش انتهایی این مدل، شامل بلوکی برای دریافت مختصات مبدا جهانی و اعمال تبدیل‌های ناشی از مجموع لینک‌ها است. در نهایت، خروجی‌های این بلوک، به صورت Z, Y, X, R ، به دست می‌آیند. برای استفاده از این خروجی‌ها، آنها را به وسیله‌ی بلوک‌های **Workspace To** به فضای متلب وارد می‌کنیم. برای اعمال مقادیر ورودی‌ی زوایا به این مدل، با استفاده از کد متلب نوشته شده، مدل را فراخوانی کرده و در نهایت با اجرای کد، می‌توانیم نتایج را مشاهده کنیم. در ادامه، به صحت سنجی چهار داده‌ی ورودی خواهیم پرداخت. برای این کار، مقادیر داده شده به زوایا ابتدا به وسیله‌ی ماتریس تبدیل محاسبه شده و سپس نتایج را با نتایج به دست آمده از مدل ربات مقایسه می‌کنیم. بلوک کلی مورد استفاده در این مدل به صورت زیر می‌باشد:



شکل ۷: مدل ربات در SimScape

در این مقایسه، لازم به ذکر است که به دلیل تفاوت تعیین جهت دستگاه‌های مختصات در دوروش، علامت مقادیر موقعیت و همچنین ماتریس دوران تفاوت‌هایی در نمایش دارند. اما مقادیر عددی مطابقت دارند.

```

1      clc
2
3      Revolute = 135
4      Revolute3 = 45
5      Revolute4 = 0
6
7
8      sim("Final_Model.slx")
9
10
11     disp("x: ")
12     disp(out.x_link.Data(:, :, 1))
13
14     disp("y: ")
15     disp(out.y_link.Data(:, :, 1))
16
17     disp("z: ")
18     disp(out.z_link.Data(:, :, 1))
19
20     disp("R: ")
21     disp(out.R.Data(:, :, 1))

```

Command Window

```

x:
    185.9691

y:
    -4.9497

z:
    -9.2227e-15

R:
    0.7071    0.7071   -0.0000
   -0.7071    0.7071   -0.0000
   -0.0000    0.0000    1.0000

```

شکل ۸: نتیجه مدل، آزمایش ۱



Final End-Effector Position (numeric):

```
disp(P_DH_num*100)
```

```
-185.9691  
0  
-4.9497
```

```
fprintf('Rotation Matrix (numeric):\n')
```

Rotation Matrix (numeric):

```
disp(R_DH_num)
```

```
-0.7071    0.7071    0  
0          0      -1.0000  
-0.7071   -0.7071    0
```

شکل ۹: نتیجه کد، آزمایش ۱

با بررسی نتیجه ی به دست آمده از این قسمت، می توان مشاهده کرد که محور دوران θ_1 در مدل ربات، محور Z در نظر گرفته شده است، در حالی که با دستگاه های مختصات مورد استفاده در ماتریس های تبدیل که پیش از این مورد استفاده قرار گرفته اند، این محور Y در نظر گرفته شده است. به همین سبب، ماتریس دوران نیز حول محور های مختلفی لحاظ شده است همچنین، جهت محور X در راستای مخالف لحاظ شده است و به این دلیل، علامت آن در دو روش متفاوت است.



```
1      clc
2
3      Revolute = 90
4      Revolute3 = 25
5      Revolute4 = 0
6
7
8      sim("Final_Model.slx")
9
10
11     disp("x: ")
12     disp(out.x_link.Data(:, :, 1))
13
14     disp("y: ")
15     disp(out.y_link.Data(:, :, 1))
16
17     disp("z: ")
18     disp(out.z_link.Data(:, :, 1))
19
20     disp("R: ")
21     disp(out.R.Data(:, :, 1))
```

Command Window

```
x:
    122.3516

y:
    70.9465

z:
    9.5586e-14

R:
    0.9063    0.4226   -0.0000
   -0.4226    0.9063    0.0000
    0.0000   -0.0000    1.0000
```

شکل ۱۰: نتیجه مدل، آزمایش ۲



```
Final End-Effector Position (numeric):
```

```
disp(P_DH_num*100)
```

```
-122.3516  
0  
70.9465
```

```
fprintf('Rotation Matrix (numeric):\n')
```

```
Rotation Matrix (numeric):
```

```
disp(R_DH_num)
```

```
-0.9063    0.4226    0  
0          0      -1.0000  
-0.4226   -0.9063    0
```

شکل ۱۱: نتیجه کد، آزمایش ۲

در این قسمت، مجدداً مشاهده می‌کنیم که مقادیر عددی به دست آمده یکسان هستند، اما به دلیل تفاوت در دستگاه‌های مختصات که پیش از این توضیح داده شد در نمایش تفاوت دارند.



```
1   clc
2
3   Revolute = 60
4   Revolute3 = 10
5   Revolute4 = 20
6
7
8   sim("Final_Model.slx")
9
10
11  disp("x: ")
12  disp(out.x_link.Data(:, :, 1))
13
14  disp("y: ")
15  disp(out.y_link.Data(:, :, 1))
16
17  disp("z: ")
18  disp(out.z_link.Data(:, :, 1))
19
20  disp("R: ")
21  disp(out.R.Data(:, :, 1))
```

Command Window

```
x:
    64.7909

y:
    87.4087

z:
   -23.5820

R:
    0.9254    0.1632    0.3420
   -0.1736    0.9848   -0.0000
   -0.3368   -0.0594    0.9397
```

شکل ۱۲: نتیجه مدل، آزمایش ۳



```
Final End-Effector Position (numeric):
```

```
disp(P_DH_num*100)
```

```
-64.7909  
-23.5820  
87.4087
```

```
fprintf('Rotation Matrix (numeric):\n')
```

```
Rotation Matrix (numeric):
```

```
disp(R_DH_num)
```

```
-0.9254    0.1632    0.3420  
-0.3368    0.0594   -0.9397  
-0.1736   -0.9848     0
```

شکل ۱۳: نتیجه کد، آزمایش ۳

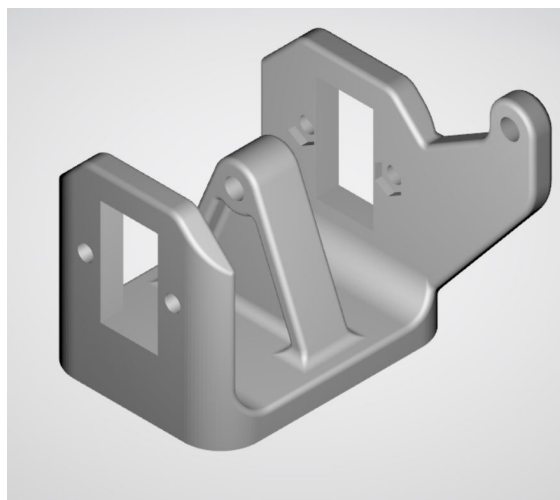
با آزمایش مدل به دست آمده با زوایای ورودی مختلف، می توانیم نتیجه بگیریم که روش های ارائه شده و مدل طراحی شده عملکرد مناسبی در شبیه سازی عملکرد ربات دارند.

۷ سوال هفتم - فرایند ساخت ربات

در بخش نهایی از این پروژه، برای ساخت قطعات از طرح های قرار داده شده در پایگاه گیت هاب ذکر شده در فایل سوالات استفاده شده است. طرح های داده شده توسط پرینتر سه بعدی چاپ شده و آماده ی استفاده برای مراحل بعد می باشد. با توجه به زمان بر بودن فرایند چاپ، ویدیو و گزارش های دقیق تر از فرایند ساخت پس از آماده شدن قطعات ارسال خواهد شد. برای چاپ این قطعات، با در نظر گرفتن استحکام مورد نیاز، از فیلامان با جنس PLA استفاده شده. همچنین، درصد Fill نیز برای بعضی از قطعات با ۴۰ درصد و برای بعضی دیگر با ۵۰ درصد لحاظ شده است. البته لازم به ذکر است که چهار لایه ی ابتدایی و انتهایی با fill کامل ساخته شده اند. در این قسمت، تصاویر قطعاتی که برای پرینت آماده هستند نمایش داده شده است.



شکل ۱۵: قطعه ۲



شکل ۱۴: قطعه ۱



شکل ۱۷: قطعه ۴



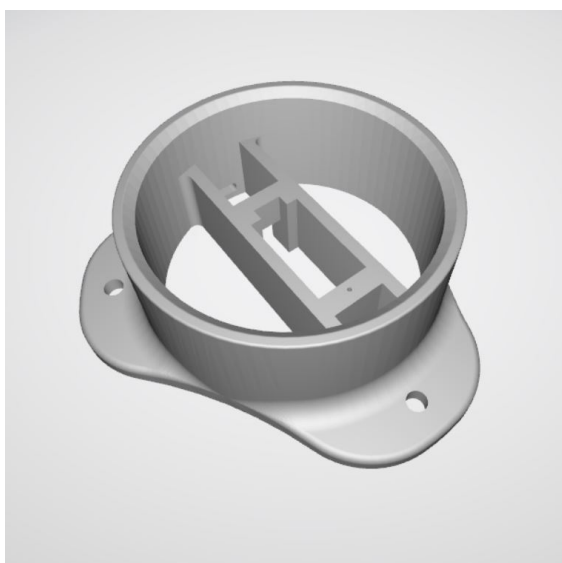
شکل ۱۶: قطعه ۳



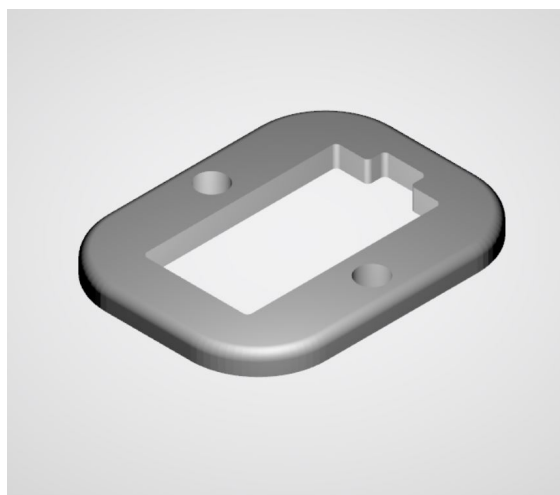
شکل ۱۹: قطعه ۶



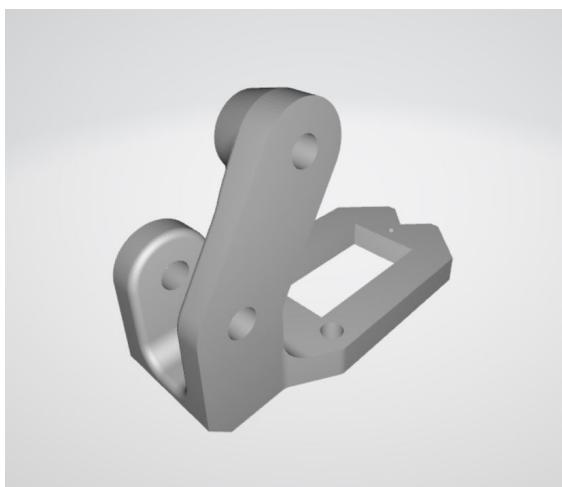
شکل ۱۸: قطعه ۵



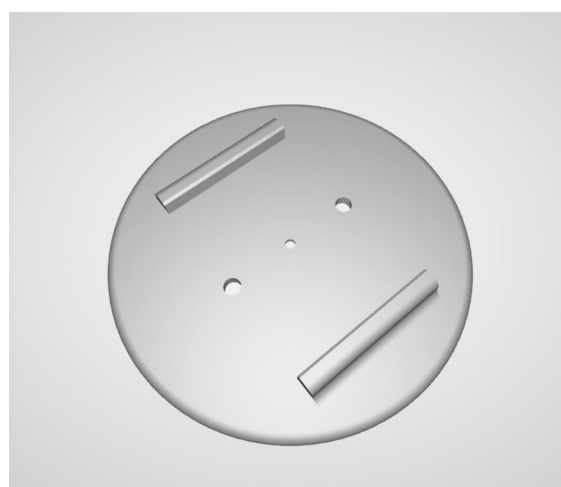
شکل ۲۱: قطعه ۸



شکل ۲۰: قطعه ۷



شکل ۲۳: قطعه ۱۰



شکل ۲۲: قطعه ۹