به نام خدا

دانشگاه تهران

پردیس دانشکده‌های فنی

دانشکده مهندسی برق و کامپیوتر

**آزمایشگاه سیستم­های کنترل خطی**

**آزمایش شماره 2**

محیا شهشهانی -- شیرین جمشیدی

810199598 -- 810199570

فروردین ماه 1402

**فهرست**

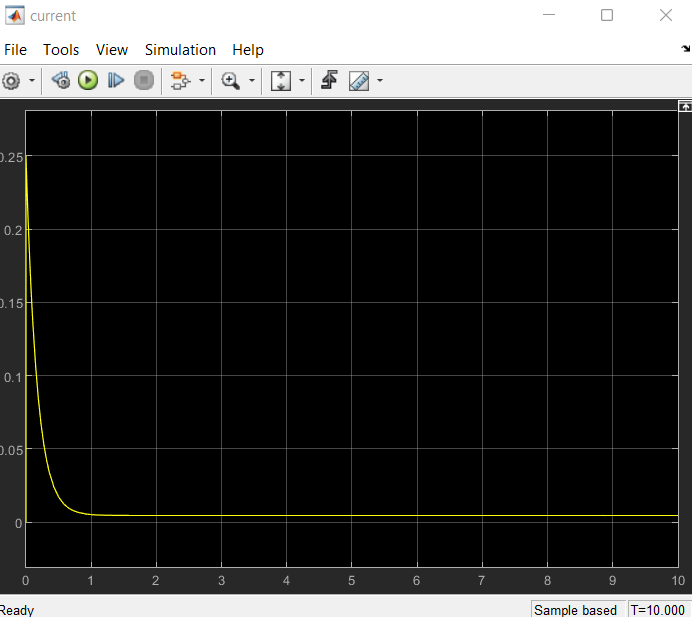
|  |  |
| --- | --- |
| **عنوان** | **شماره صفحه** |
| چکیده | ۳ |
| بخش 2 | 4 |
| بخش 3 | 6 |
| بخش 4 | 9 |
| بخش 5 | 12 |
| پیوست | 22 |
|  |  |

**چکیده**

در بخش‌های یک تا چهار به تحلیل و شبیه سازی موتور DC با استفاده از روش های مختلف میپردازیم: حل معادله دیفرانسیل با استفاده از MATLAB، شبیه سازی به کمک Simulink، شبیه سازی به کمک SimMechanics، بدست اوردن تابع تبدیل و رسم پاسخ پله. در بخش 5 گزارشکار به شبیه سازی سیستم های مکانیکی به وسیله‌ی SimMechanics پرداختیم. و سپس با استفاده از tune، یک کنترلر pid برای سیستم اخر طراحی کردیم. بخش اول صورت گزارش، به توضیح بیشتر درباره‌ی روند کار تخصیص داده شده که چیزی برای انجام و گزارش نداشت. به همین سبب گزارشکار از بخش دوم نوشته شده است که نتایج شبیه سازی‌های مربوطه در ان قرار گرفته است.

# بخش 2) شبیه‌سازی معادلات دیفرانسیل در محیط simulink

خروجی جریان آرمیچر:



شکل 1-2

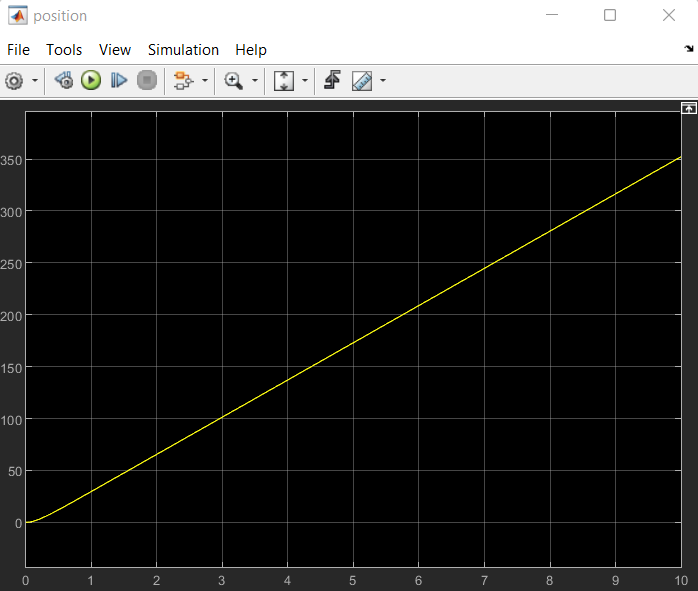
خروجی سرعت موتور:

Graphical user interface

Description automatically generated with low confidence

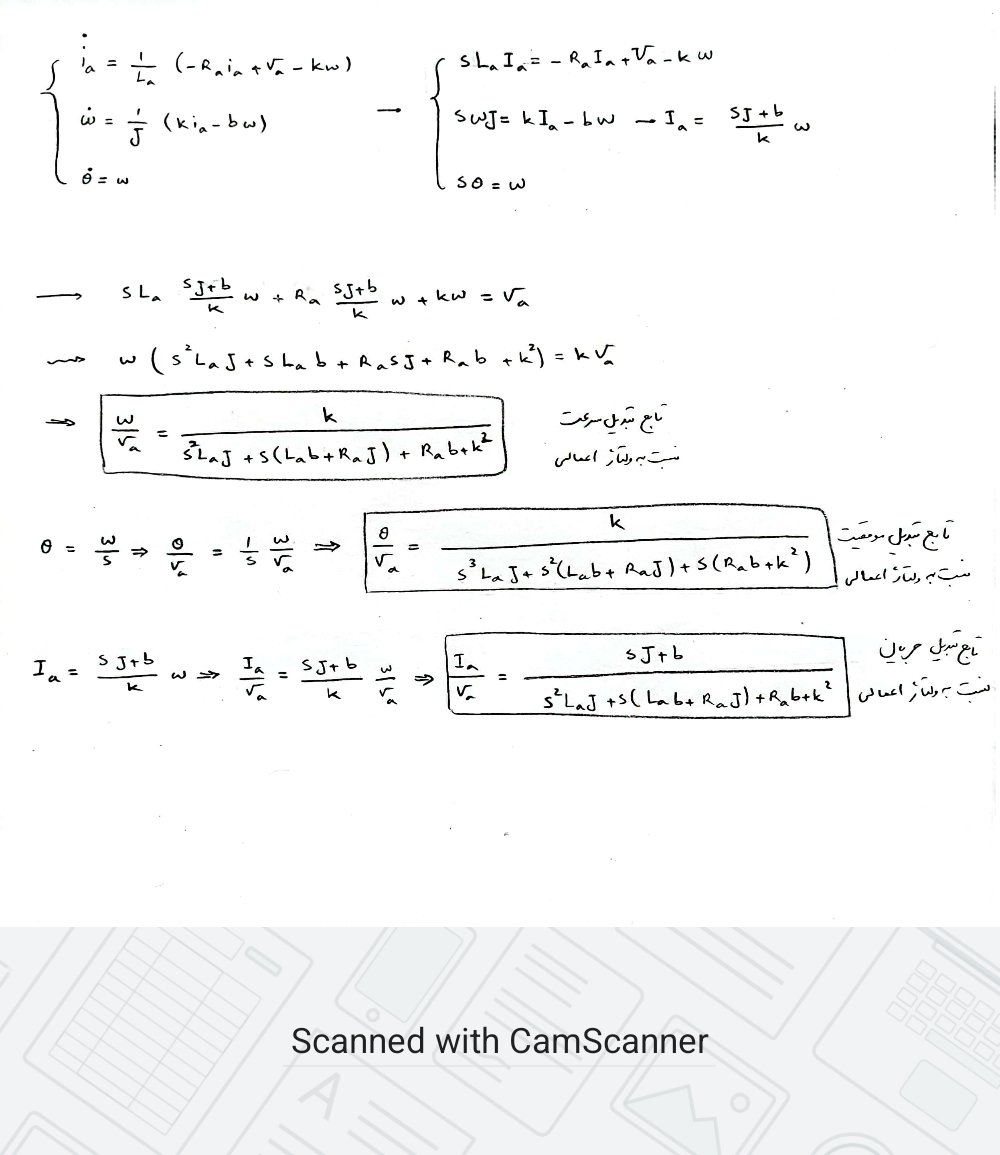
شکل 2-2

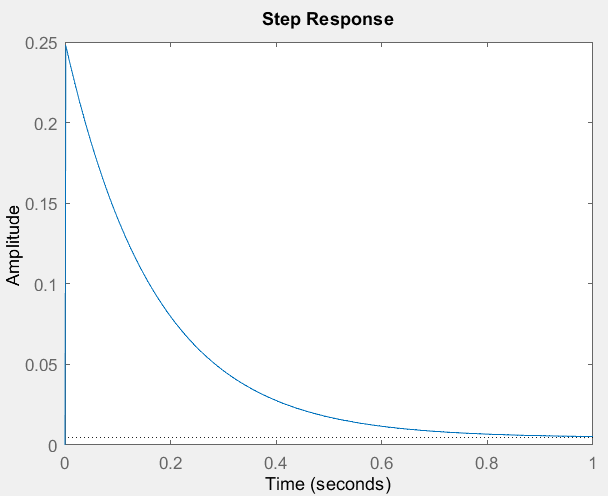
خروجی موقعیت موتور:



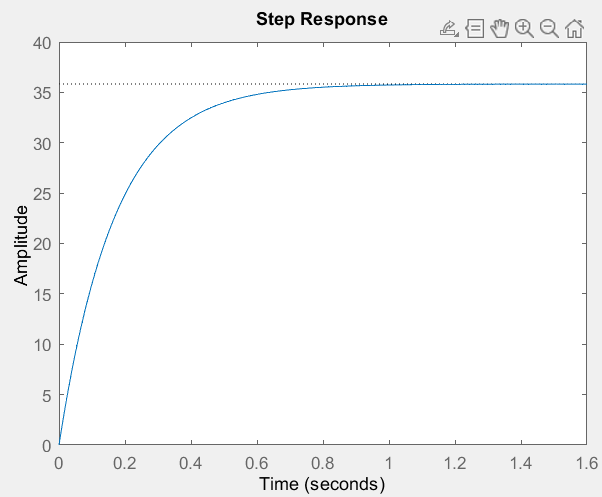
شکل 3-2

# بخش 3) شبیه‌سازی معادلات دیفرانسیل در محیط editor(mFile)

* شبیه سازی به کمک تابع تبدیل:

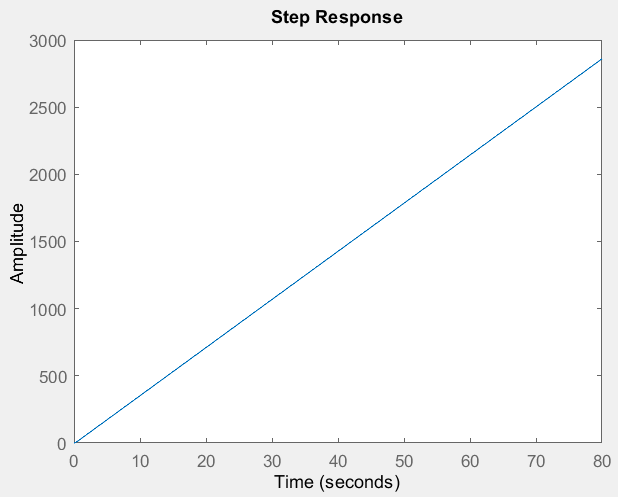
پاسخ پله‌ی تابع تبدیل جریان موتور:

شکل 1-3

پاسخ پله‌ی تابع تبدیل سرعت موتور:

شکل 2-3

پاسخ پله‌ی تابع تبدیل موقعیت موتور:



شکل 3-3

میبینیم که نتایج بدست امده در این بخش، تطابق کاملی با بخش قبلی دارد.

* Graphical user interface, application

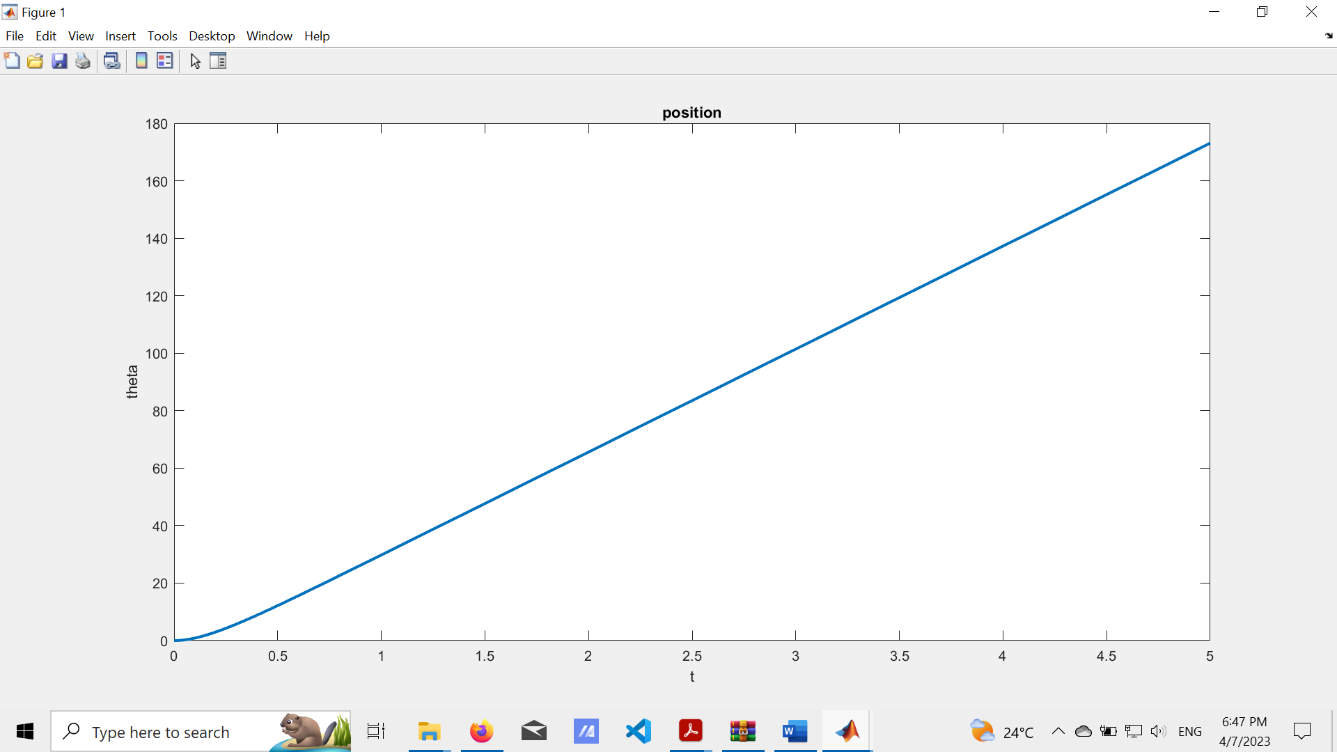
  Description automatically generatedشبیه سازی به کمک معادله دیفرانسیل:

شکل 4-3

Graphical user interface, application

Description automatically generated

شکل 5-3



شکل 6-3

# بخش 4) مدل کردن مدل DC با استفاده از بخش‌های مختلف simscape

شکل موتور DC ساخته شده:

Graphical user interface

Description automatically generated with medium confidence

شکل 1-4

Diagram

Description automatically generatedشکل سیستم:

شکل 2-4

A screenshot of a computer

Description automatically generatedسیگنال جریان آرمیچر:

شکل 3-4

A screenshot of a computer

Description automatically generatedسیگنال موقعیت موتور:

شکل 4-4

A screenshot of a computer

Description automatically generatedسیگنال سرعت موتور:

شکل 5-4

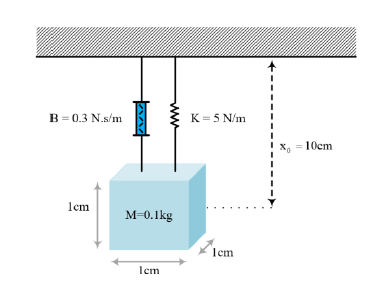
میبینیم که هر سه شکل موج خروجی با تمامی بخش های قبلی اعم از سیمولینک و محاسبه با تابع تبدیل و حل معادله دیفرانسیل تطابق دارند.

در هر نمودار **سرعت** مشاهده میکنیم که پس از عبور از حالت گذرا، سرعت چرخش موتور به مقدار ثابتی میرسد که باتوجه به پارامترهای داده شده به سیستم در تمامی نمودارها این عدد برابر با 35.827 رادیان بر ثانیه است.

در هر نمودار **موقعیت**، همانطور که انتظار داریم مشاهده میکنیم که پس از حالت ناپایدار اولیه، موقعیت با شیب ثابتی پیوسته در حال افزایش است. میدانیم که در موتور DC منظور از موقعیت، زاویه قرارگیری است که برحسب رادیان در تمامی نمودارها رسم شده. هنگامی که سرعت به مقدار ثابت خود میرسد، زاویه نیز با شیب ثابت(خطی) افزایش می‌یابد.

در هر نمودار **جریان** مشاهده میکنیم که با گذر زمان و عبور از حالت گذرا، مقدار جریان آرمیچر به صفر میرسد زیرا در حالت پایدار سرعت موتور ثابت شده و حال که سرعت چرخش ثابت است، تغییرات شار نسبت به زمان صفر بوده در نتیجه دیگر ولتاژ القایی تولید نمیشود و همین امر باعث کاهش پیوسته جریان میشود تا زمانی که به مقدار صفر برسد.

# بخش 5) کار با SimMechanics

**سیستم اول:**

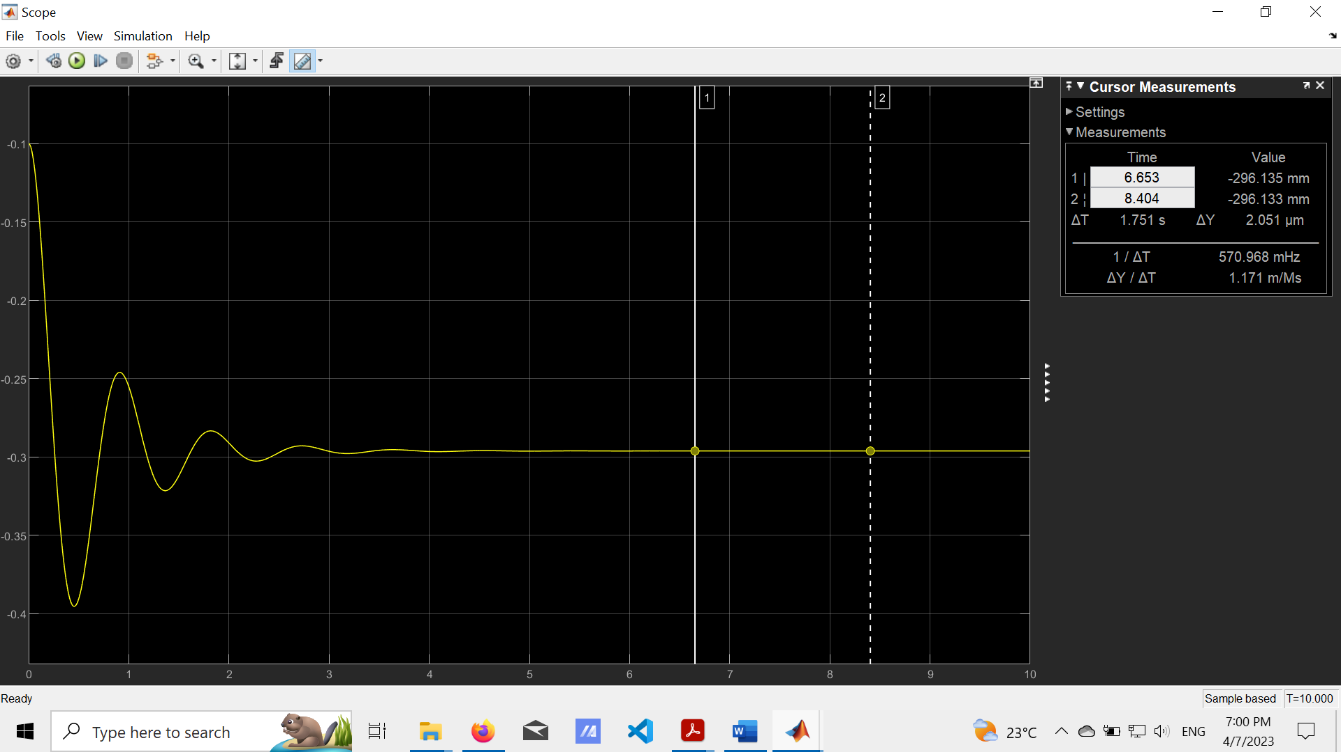
شکل 1-5

به ازای l0=0.1(m) داریم:

برای محاسبه نقطه تعادل t را به بی‌نهایت میل می‌دهیم و خواهیم داشت:

\*در محاسبه جهت مثبت را به سمت پایین گرفتیم به این دلیل عدد حاصل مثبت می‌باشد.

می‌بینیم که پاسخ حالت تئوری کاملا با شبیه‌سازی تطابق دارد.

نتیجه شبیه سازی:

شکل 2-5

به ازای l0=0.05(m) داریم:

برای محاسبه نقطه تعادل t را به بی‌نهایت میل می‌دهیم و خواهیم داشت:

\*در محاسبه جهت مثبت را به سمت پایین گرفتیم به این دلیل عدد حاصل مثبت می‌باشد.

A screenshot of a computer

Description automatically generatedمی‌بینیم که پاسخ حالت تئوری کاملا با شبیه‌سازی تطابق دارد.

شکل 3-5

Graphical user interface

Description automatically generated with medium confidence**سیستم دوم:**

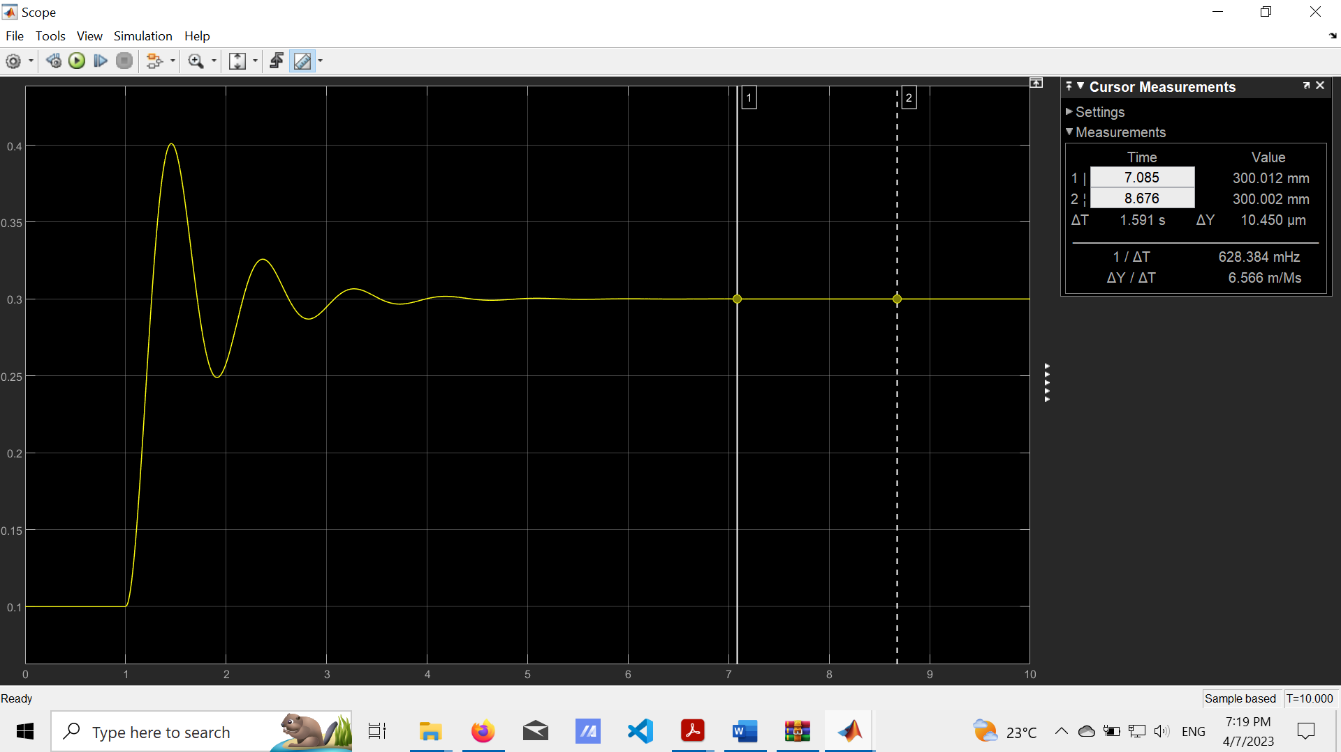
شکل 4-5

به ازای l0=0.1(m) داریم:

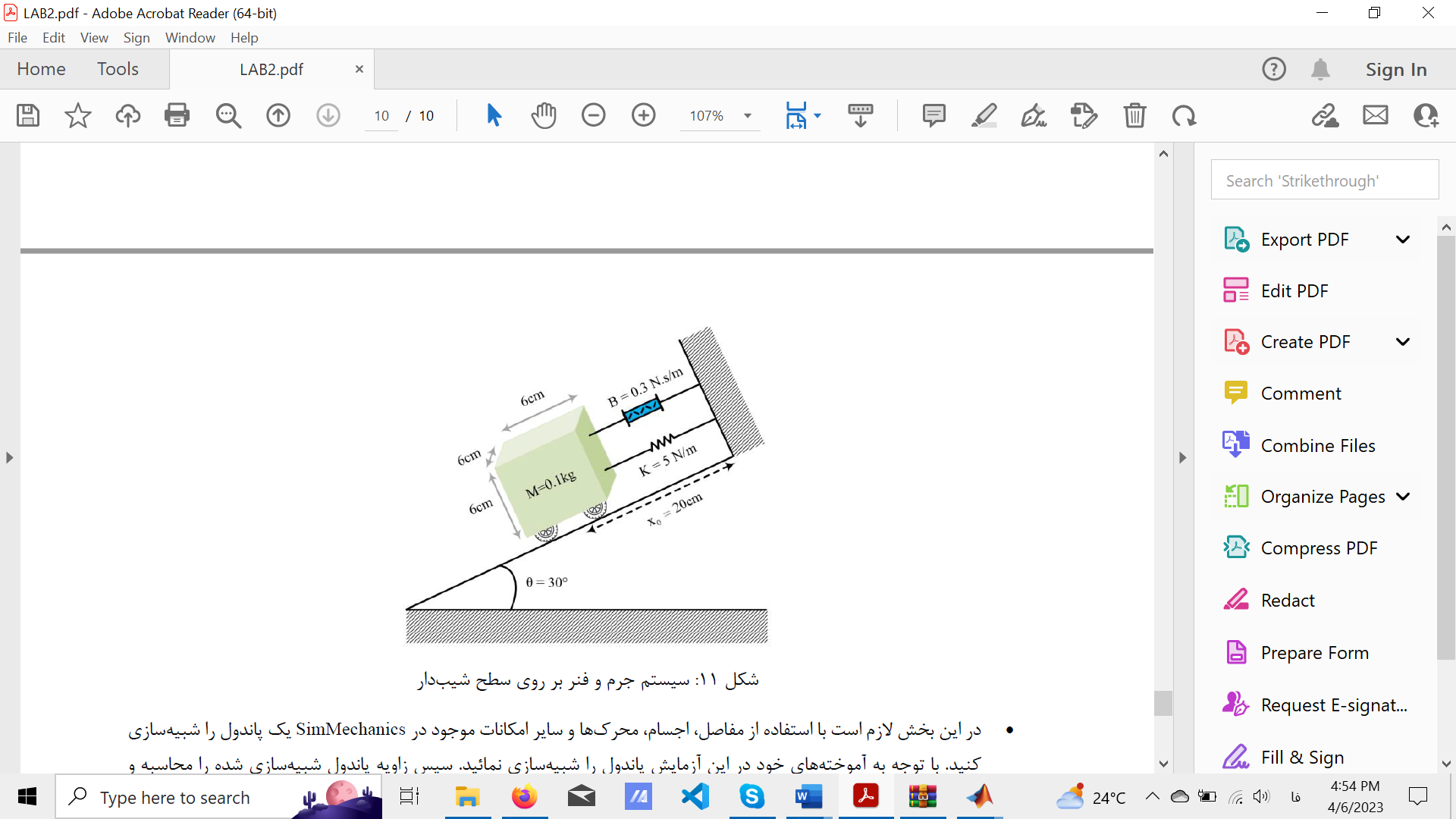
برای محاسبه نقطه تعادل t را به بی‌نهایت میل می‌دهیم و خواهیم داشت:

\*در محاسبه و شبیه سازی جهت مثبت را به سمت چپ دیوار گرفتیم به این دلیل عدد حاصل مثبت می‌باشد.

می‌بینیم که پاسخ حالت تئوری کاملا با شبیه‌سازی تطابق دارد.

نتیجه شبیه سازی:

شکل 5-5

**سیستم سوم:**

شکل 6-5

به ازای l0=0.15(m) داریم:

برای محاسبه نقطه تعادل t را به بی‌نهایت میل می‌دهیم و خواهیم داشت:

\*در محاسبه و شبیه سازی جهت مثبت را به سمت چپ دیوار گرفتیم به این دلیل عدد حاصل مثبت می‌باشد.

می‌بینیم که پاسخ حالت تئوری کاملا با شبیه‌سازی تطابق دارد.

A screenshot of a computer

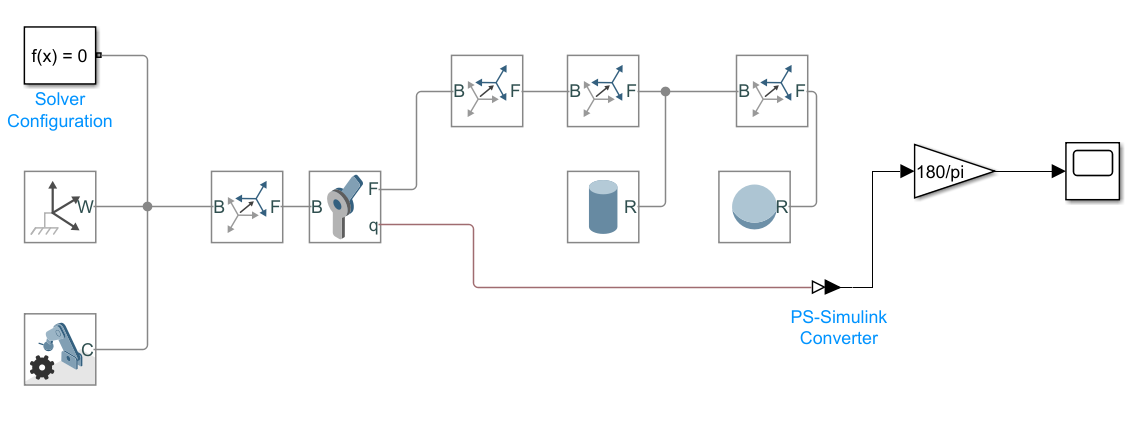
Description automatically generated with medium confidenceپاسخ شبیه سازی:

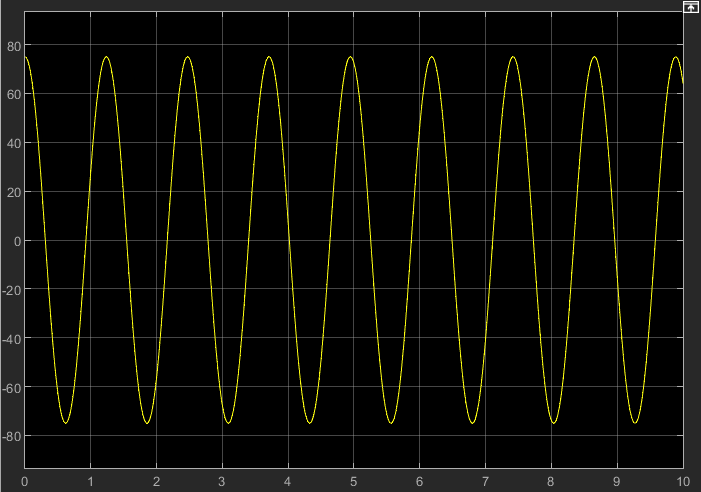
شکل 7-5

**سیستم چهارم:**

* پیش از طراحی کنترلر:

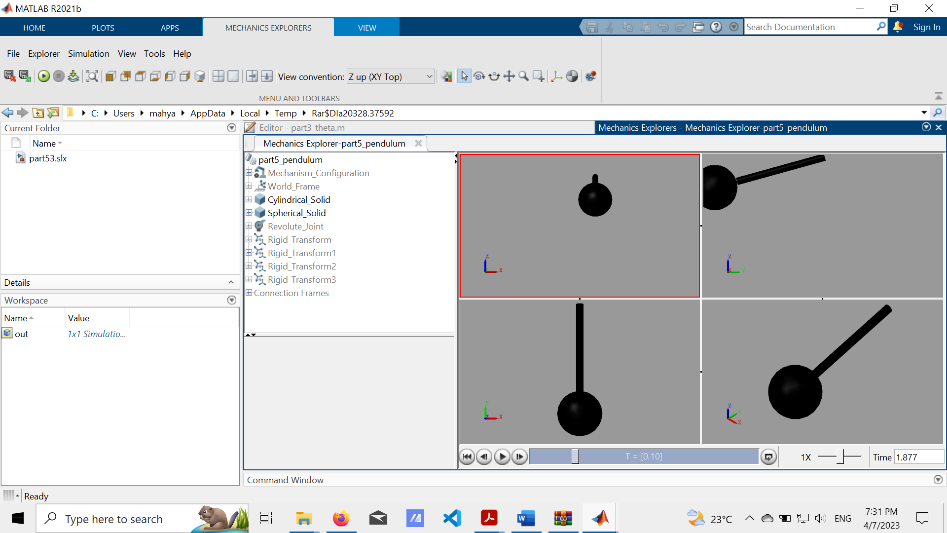
شکل 8-5

خروجی اسکوپ:



شکل 9-5

همانطور که انتظار داشتیم(به دلیل عدم وجود اصطکاک)، پاندول از زاویه +75 درجه تا -75 درجه به صورت سینوسی نوسان میکند. همین نتیجه را نیز در محیط SimMechanics مشاهده کردیم.

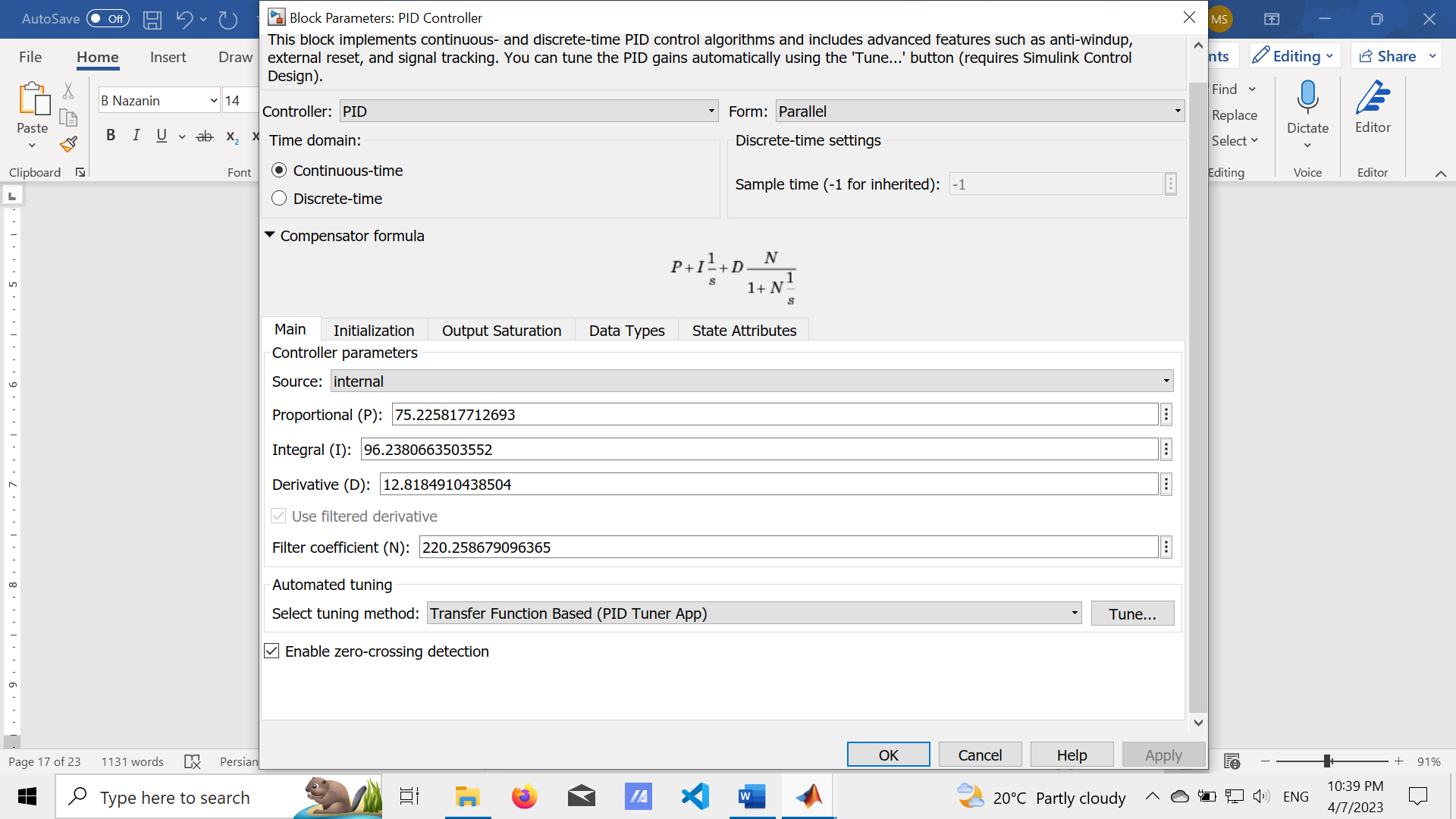


شکل 10-5

* Diagram

  Description automatically generated with medium confidenceپس از طراحی کنترلر:

شکل 11-5

برای طراحی کنترلر، از حلقه‌ی فیدبک استفاده میکنیم که هر سری با تفاضل زاویه خروجی با مقدار 30 درجه تصمیم میگیرد چه نیرویی به سیستم وارد کند.

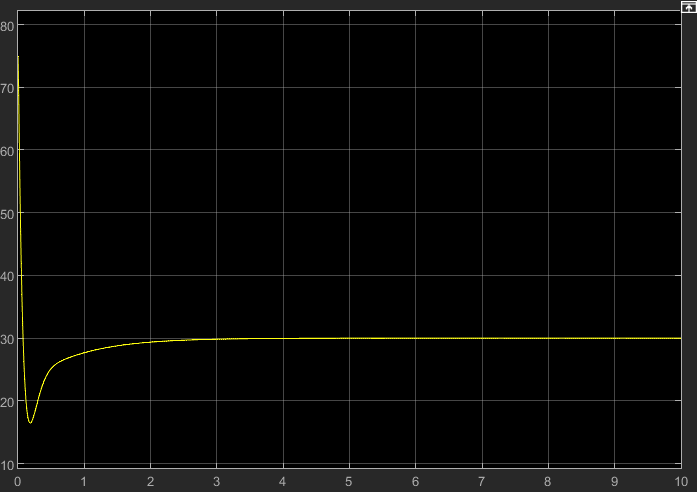
با استفاده از قابلیت tune کنترلرPID، ضرایب کنترل‌کننده‌های تناسبی، مشتق‌گیر و انتگرال‌گیر به گونه‌ای محاسبه شده است که ما را به مطلوب مسئله‌مان رسانده تا در زاویه 30 درجه پایدار شود.

Graphical user interface, text

Description automatically generated

شکل 12-5

خروجی اسکوپ در نهایت به شکل زیر میباشد:



شکل 13-5

که همانطور که انتظار داشتیم از زاویه 75 درجه که در ان رها شده است، undershoot داشته و سپس در زاویه 30 درجه به تعادل میرسد.

نکته: تمام زوایا را برای SimMechanics باید به رادیان تبدیل کنیم و برای Simulink، باید به درجه تبدیل کنیم.

Graphical user interface, text, application

Description automatically generated**پاندول وارونه:**

شکل 13-5

اگر منظور این بوده است که گوی در ابتدا بالای بالا باشد (یعنی با محور زاویه 180 درجه داشته باشد)، میبینیم که سیستم در تعادل بوده و میماند. پس زاویه ابتدایی را برابر 179 درجه قرار میدهیم و سپس شکل موج مشاهده شده به صورت زیر خواهد بود:

A screenshot of a computer

Description automatically generated with medium confidence

شکل 14-5

همانطور که انتظار داشتیم، پاندول از +179 تا -179 نوسان میکند.

Graphical user interface

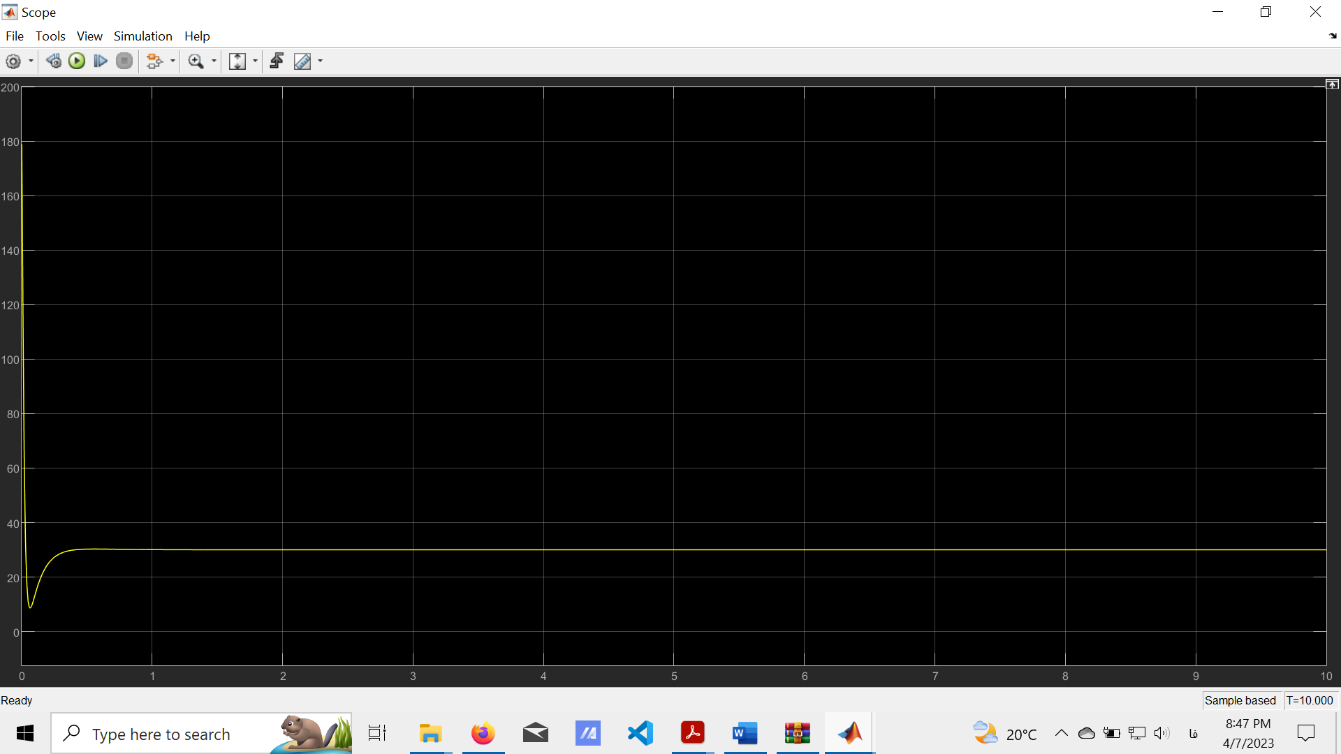
Description automatically generated

شکل 15-5

حال برای این سیستم با همان روش قبلی کنترلری طراحی میکنیم تا سیستم را در زاویه 30 درجه ثابت نگه دارد.

Graphical user interface, text, application

Description automatically generatedضرایب عبارتند از:

شکل موج خروجی:

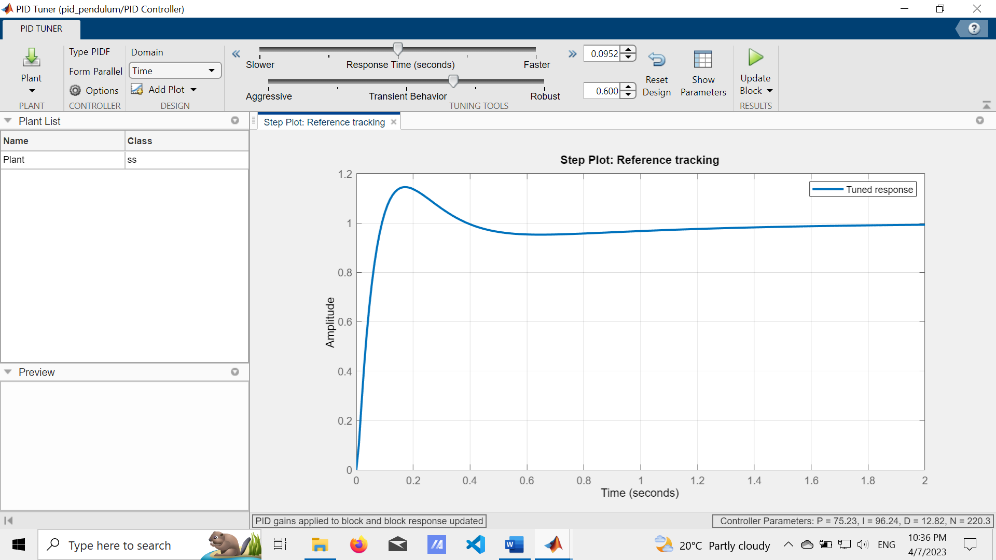
شکل 16-5

همانطور که انتظار داشتیم از زاویه 179 درجه که در ان رها شده است، undershoot داشته و سپس در زاویه 30 درجه به تعادل میرسد.

Graphical user interface, text, application

Description automatically generatedشکل سیستم با کنترلر:

شکل 17-5

Graphical user interface, application

Description automatically generated

شکل 19-5

شکل 18-5

دو شکل بالا، پاسخ پله کنترل کننده طراحی شده هستند. میبینیم که برای پاندول وارونه که در بالای بالا قرار دارد، ضریب تناسبی در حدود 5 برابر بیشتر، ضریب انتگرالگیر بیش از 10 برابر و ضریب مشتق گیر حدودا 3 برابر حالتی است که پاندول از زاویه 75 درجه رها شود. این افزایش ضرایب با شهود ما نیز سازگار است چرا که وقتی پاندول از فاصله بالاتری رها شده، در زاویه 30 درجه انرژی بیشتری در آن ذخیره شده و نیز سرعت بیشتری دارد. کنترل کننده طراحی شده ما که با فیدبک کار میکند، به مقایسه پیاپی گشتاور سیستم میپردازد و سپس بر اساس انرژی سیستم گشتاور(نیرویی) اعمال میشود تا پاندول از حرکت بازایستد. هنگامی که خود پاندول در هر دو نقطه مورد بررسی، نسبت به حالت قبل، انرژی بیشتری دارد، کنترل کننده نیز به گشتاور (نیروی) بیشتری برای ثابت نگه داشتن آن نیاز پیدا میکند که همین منجر به افزایش مقدار ضرایب کنترل کننده می شود که آن را مشاهده کردیم.

# پیوست: روند اجرای برنامه

فایل‌های مربوط به هر بخش داخل فولدری با شماره همان بخش موجود است.

* بخش دو: اجرای فایل شبیه سازی سیمولینک به همراه فایل متلب متغیرها که منجر به ذخیره در workspace میشود.
* بخش سه:
* قسمت اول: سه فایل تابع تبدیل به همراه فایل متلب متغیرها
* قسمت دوم: فایل حل دیفرانسیلی معادلات
* بخش چهارم: فایل شبیه سازی SimMechanics
* بخش پنجم:
* سیستم اول: دو فایل شبیه‌سازی سیستم که تفاوت آن‌ها در طول آزاد فنر است.
* سیستم دوم: فایل شبیه سازی SimMechanics
* سیستم سوم: فایل شبیه سازی SimMechanics
* سیستم چهارم:

پاندول معمولی: دو فایل شبیه سازی، یکی خود پاندول و دیگری با کنترلر

پاندول وارونه: دو فایل شبیه سازی، یکی خود پاندول و دیگری با کنترلر