ECE311 Introduction to Linear Control System

Lab 1: Introduction to LTI Systems in Matlab and Simulink
Group number: Team 15
Group member: Weizhou Wang 1004421262
Shuang Wu 1004728119
Lingwei Sun 1004723276
Naize Yang 1005274579

Video link:
https://drive.google.com/file/d/1fboZ8dO2JUY8Zp3Nod-G70rXDzjtA8rE/view?usp=sharing

3. LTI System Representations and Conversions

As we define the motor model within Matlab, we generate the following outputs for Lab1 part3.

Output 1.

1. The following pictures are the printed outputs for the transfer functions *G_motor* and *G_motor_simplified*.

G_motor transfer function:

```
G_motor =

           833.3
   ---------------------
   s^2 + 150.2 s + 33.33

Continuous-time transfer function.
```

G_motor_simplified, simplified version of the transfer function:

```
G_motor_simplified =

     5.556
   ----------
   s + 0.2222

Continuous-time transfer function.
```

2. The following picture is the printed output for the transfer function in ZPK form as named *zpk_motor*.

Transfer function in ZPK form:

```
zpk_motor =

          833.33
  ------------------
  (s+0.2223) (s+149.9)

Continuous-time zero/pole/gain model.
```

3. The following picture is the printed output for the poles of the transfer function *G_motor,* along with the comments on the location of the poles and expected motor behavior.

Poles of the transfer function *G_motor*:

```
pole =

 -149.9444
   -0.2223
```

Comments:
As we observed from the printed output for the poles of the transfer function G_motor, we find out that the location of the two poles are on the negative real axis and are separated apart. One pole has the magnitude of 0.2223 which is much smaller than the other pole's magnitude, which is 149.9444.

The expected behaviour based on the location of the poles when the input voltage is a unit step is the two negative value poles contribute exponential terms that decay with time. So, when the input voltage is a unit step, the output time response can be found as exponential functions before it reaches the steady state. The output will first with a short transient contributed by term exp(-149.9444t), then, the response will be dominated by a slower exponential exp(-0.2223t), before it reaches the steady state.

4. The following picture is the printed output for the numerator and denominator arrays of the transfer function *G_motor* and *G_motor_simplified*.

Numerator array of the transfer function G_motor:

```
num =

         0         0  833.3333
```

Denominator array of the transfer function G_motor:

```
den =

    1.0000  150.1667   33.3333
```

Numerator array of the transfer function G_motor_simplified:

```
num1 =

         0    5.5556
```

Denominator array of the transfer function G_motor_simplied:

```
den1 =

    1.0000    0.2222
```

4. Numerical Simulation of LTI Systems

Output 2.

1. Output figures for the corresponding three parts.

Figure 1.1: The figure below consists of three sub-figures with respect to 1000 time samples between 0 to 30 seconds. The first sub-figure is the step response of the state space model motor with respect to time; the second sub-figure is the step response of the simplified state space model motor_simplified with respect to time; the third sub-figure is the difference of step response between state space model motor with respect to time.

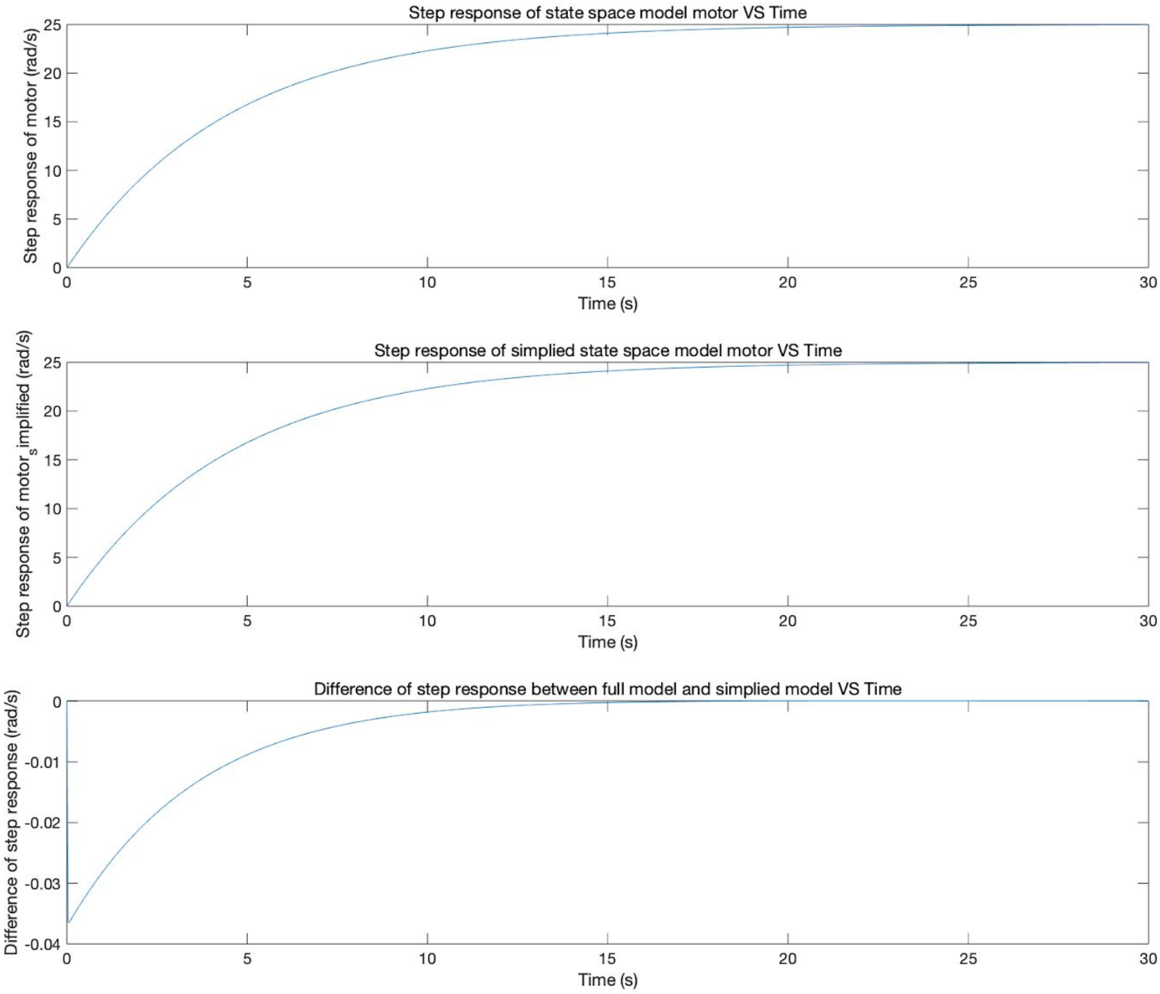


Figure 1.2: The figure below is the armature current of the state space model motor with respect to time samples.

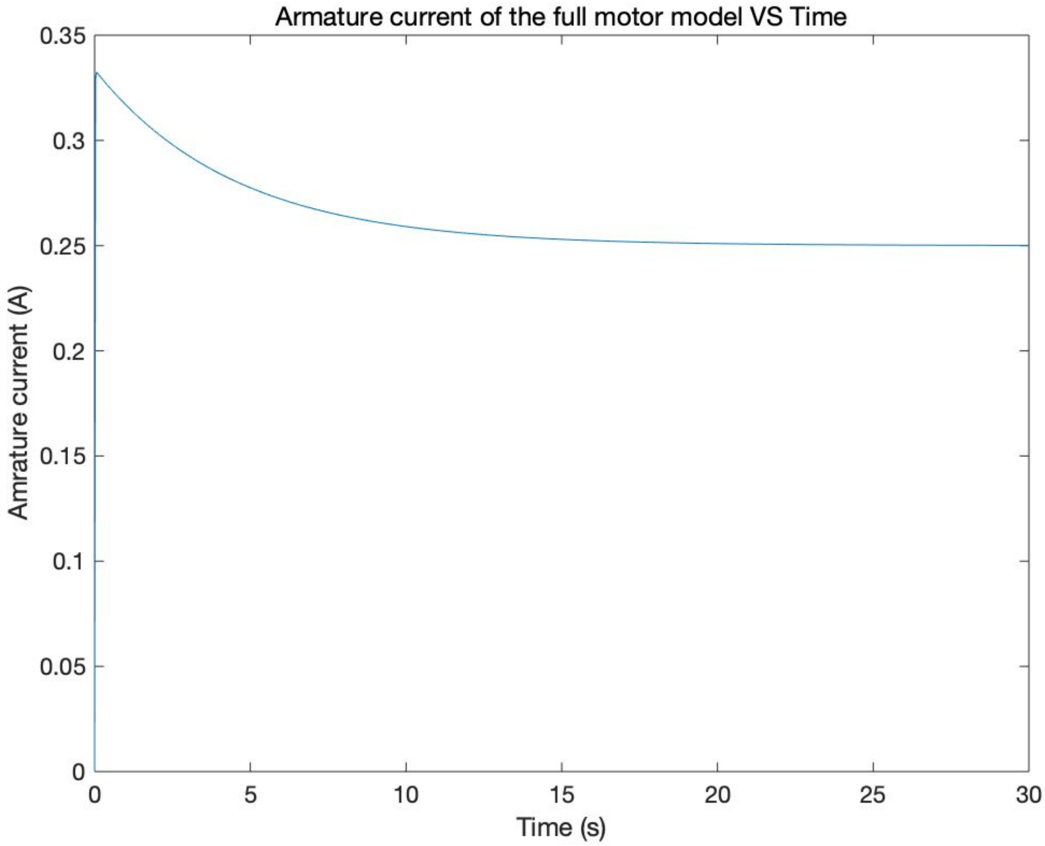
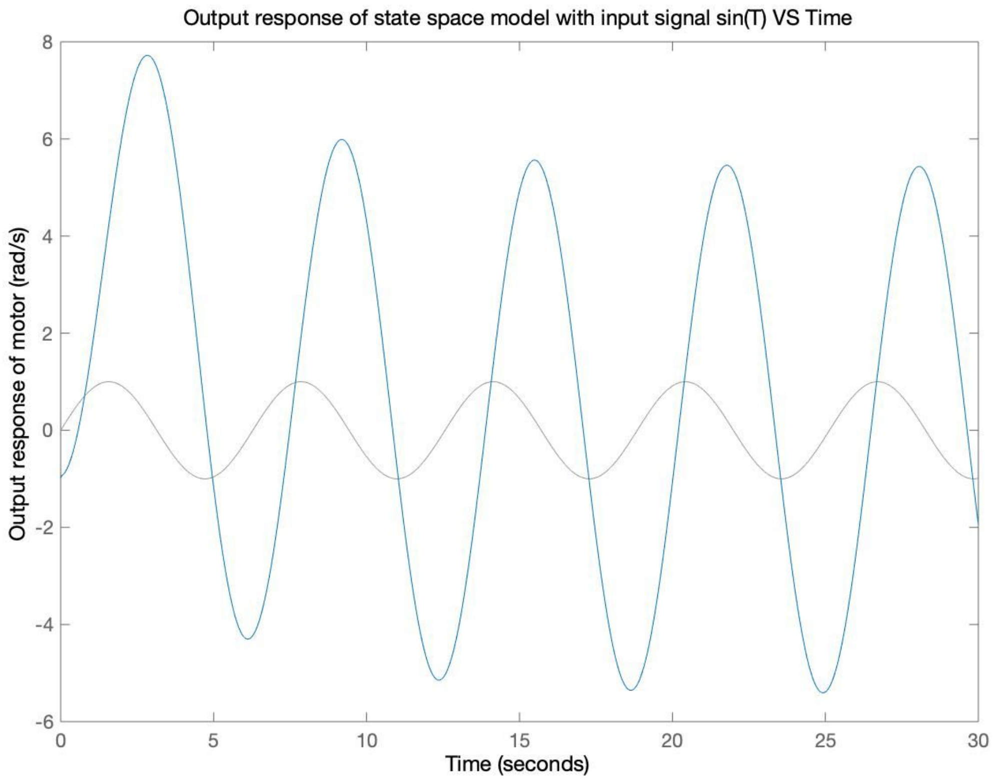


Figure 1.3: The figure below is the output response of the state space model motor with respect to time samples in T, and the input function is sin(T).



2. The simplified version and the actual full motor model has a very close output. The difference between the waveforms is very little. The maximum error between motor and motor_simplified is around 0.0365 rad/s.

3. The picture below is the approximate asymptotic value of the motor speed, as a response to a unit step.

```
approximate_asymptotic =

    24.9682
```

4. The picture below is the theoretical asymptotic value of the motor speed, as a response to a unit step. The result is 83330 / 3333 = 25.0015, which is very close to the approximate asymptotic value.

```
theoretical_asymptotic =

83330/3333
```

5. The picture below is the theoretical amplitude of oscillation of the motor speed in response to input sin(T).

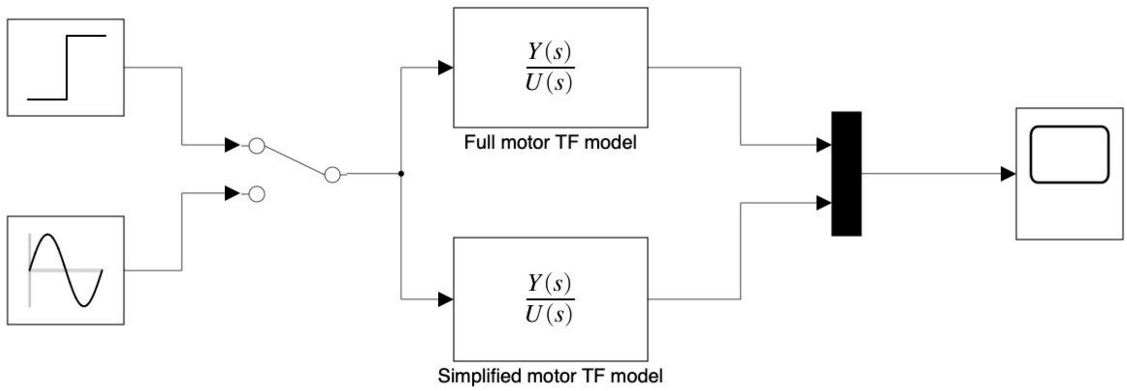
```
approximate_amplitude =

    5.4201
```

6. The picture below is the theoretical amplitude of oscillation of the motor speed in response to input sin(T). The magnitude of the theoretical amplitude is sqrt(1.1419^2 + 5.3035^2) = 5.4250, which is very close to the approximate amplitude value.

```
theoretical_amplitude =

   1.1419 - 5.3035i

theoretical_amplitude =

    5.4251
```

5. Definition of LTI System Blocks in Simulink and Numerical Simulation

The Simulink diagram for Lab1 part5.



Output 3.

1. Two Outputs of the motor models.

Figure 1.1: Below is the output of the motor models when the input is a unit step. The output from the full motor TF model is represented by a yellow line, and the output from the simplified motor is represented by a blue line:

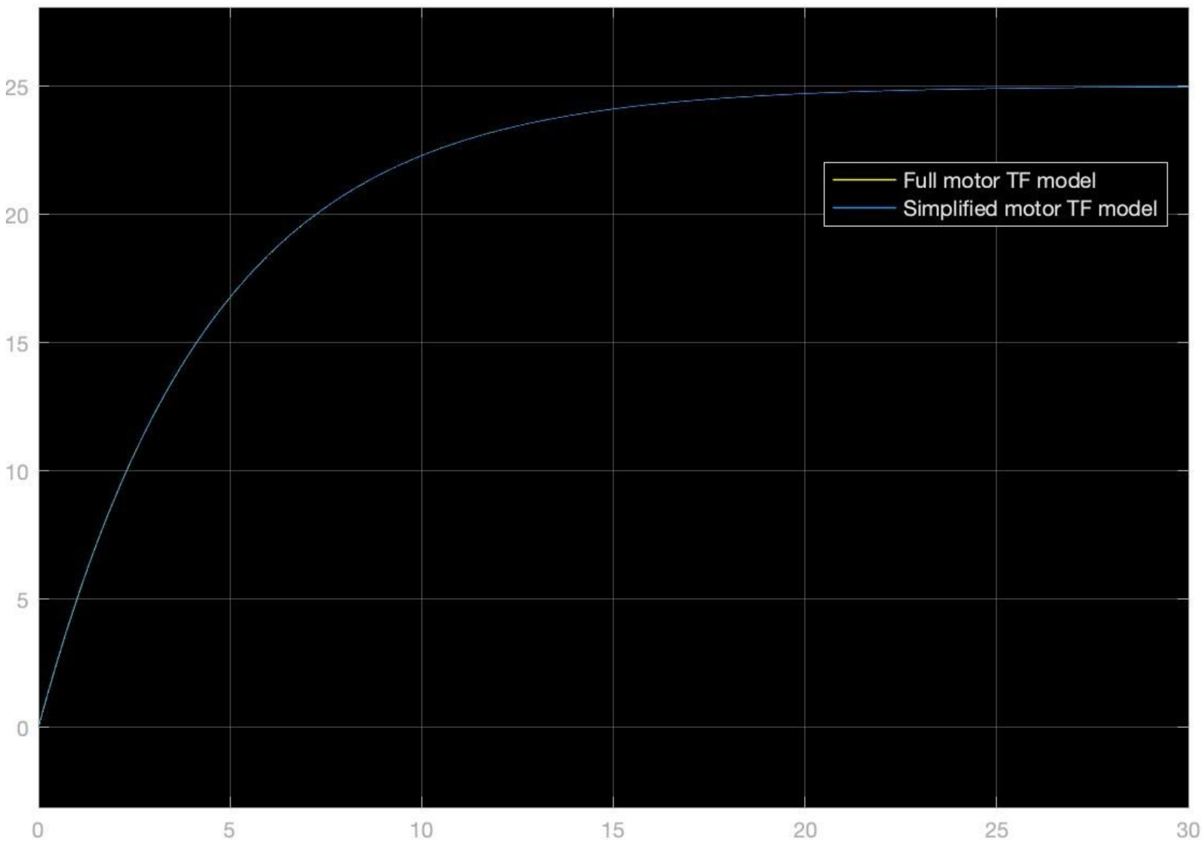
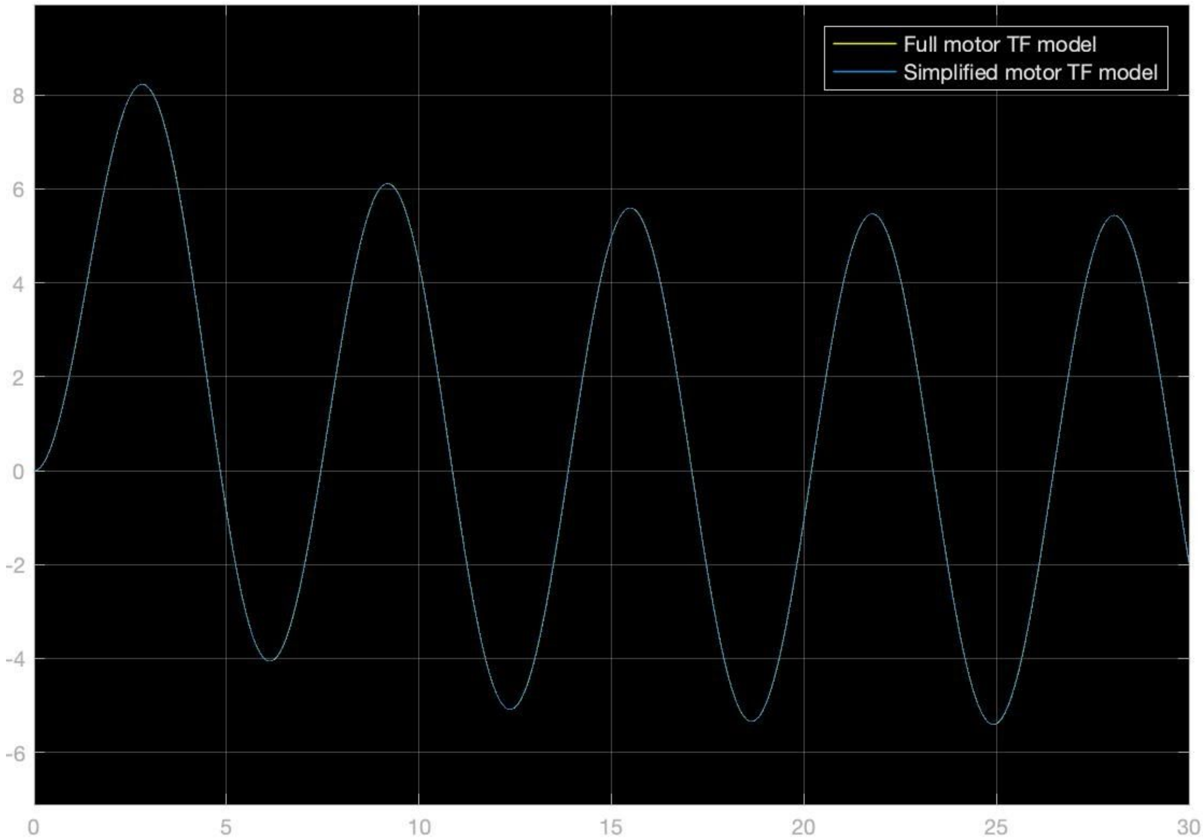


Figure 1.2: Below is the output of the motor models when the input is a sine wave. Same as above, the output from the full motor is the yellow line, and the output from the simplified motor is the blue line:



2. Comment on how well the simplified transfer function model approximates the full transfer function model for the two given input signals.

   From the overall view of these two plots, the blue and yellow lines are basically the same so that we cannot distinguish the two lines from their overall views. However, if we zoom in on the plots a lot, we find that there is a small difference between the output values from the full motor and the simplified motor:

Figure 2.1: Below is the zoom-in view of the output when the input is a unit step.

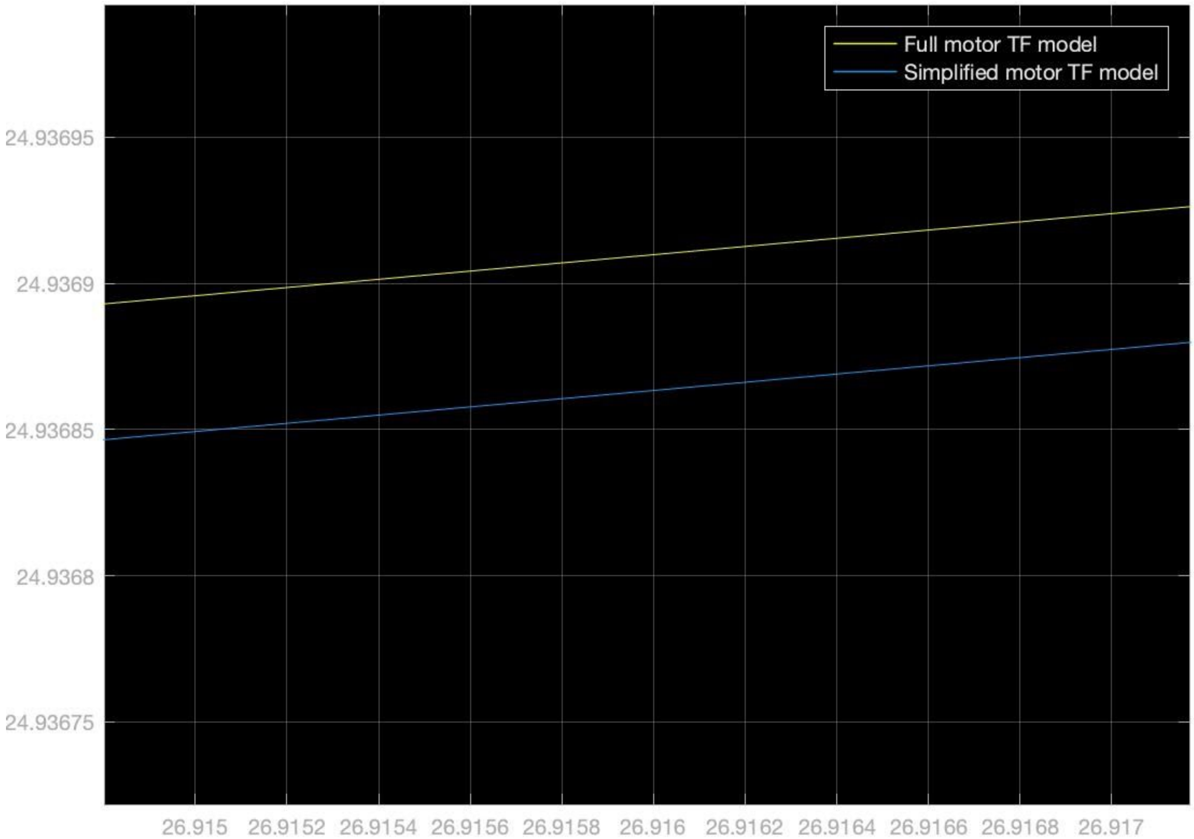
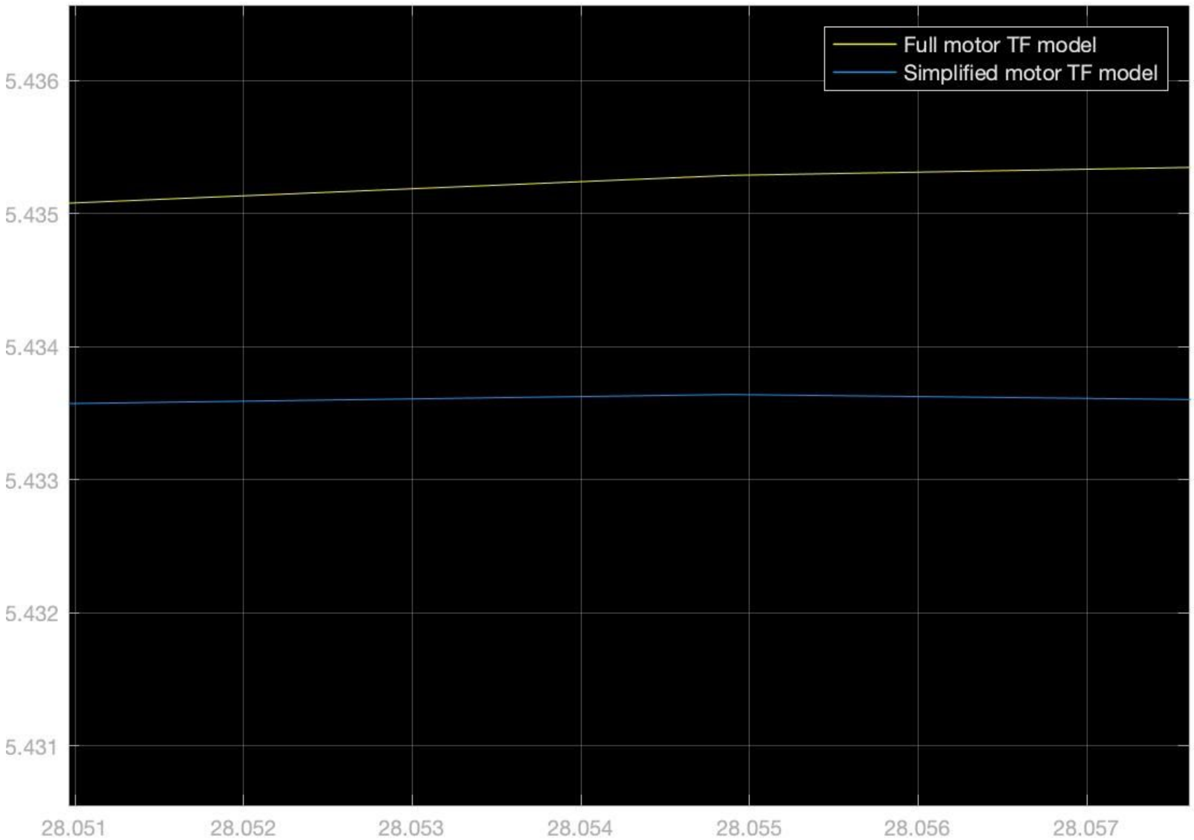


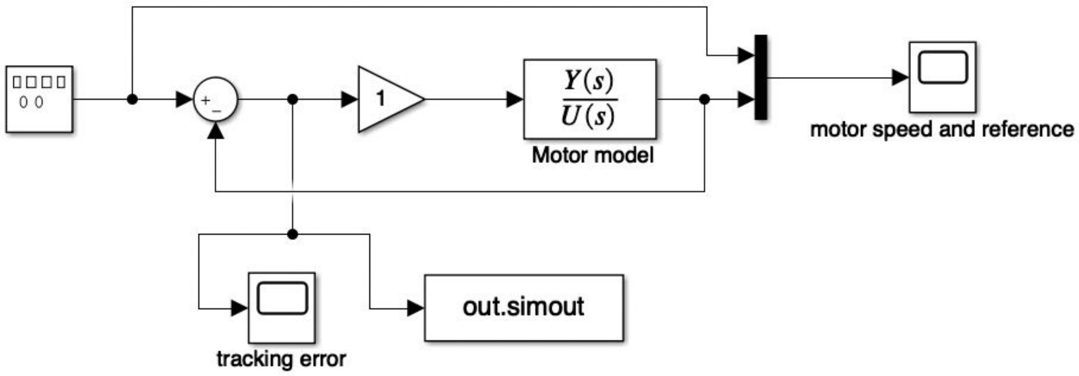
Figure 2.2: Below is the zoom-in view of the output when the input is a sine wave:



Notice that the difference between the full motor's output and the simplified motor's output is small, which is negligible in practice. Therefore, we conclude that the simplified motor transfer function model approximates the full transfer function model really well.

6. Proportional Control of the Permanent Magnet DC Motor

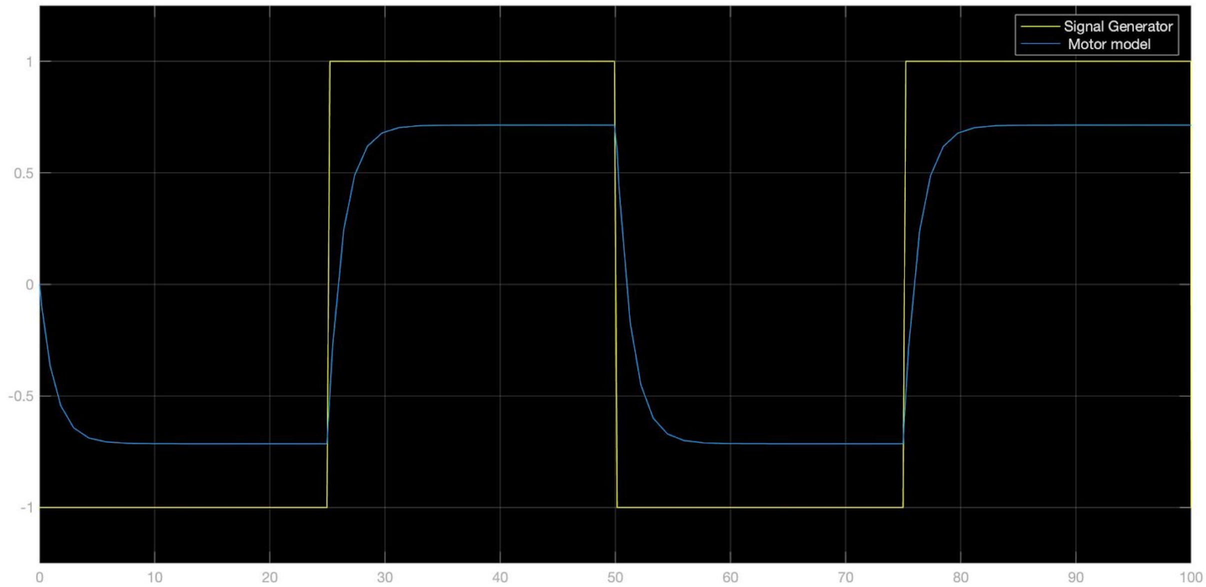
The following is the schematics that we created for simulating the outputs for Lab1 part6.



Output 4.

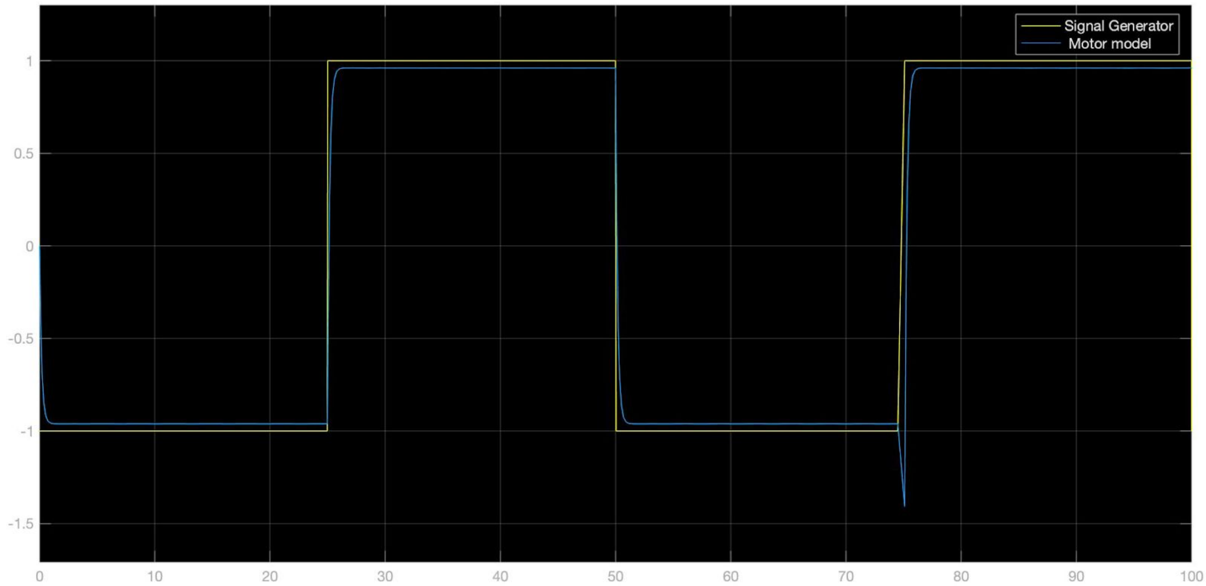
1. The following are the two figures displaying the motor speed and the reference signal for the two cases of K=0.1 and K=1.

Figure 1.1 The motor speed and reference signal for K=0.1



The figure1.1 demonstrates the input and output signals when the K in the proportional controller equals 0.1. The input signal generated by the signal generator is plotted in yellow. The output of the motor model is plotted in blue. The total experimenting time is 100 seconds.

Figure1.2 The motor speed and reference signal for K=1



The figure1.2 demonstrates the input and output signals when the K in the proportional controller equals 1. The input signal generated by the signal generator is plotted in yellow. The output of the motor model is plotted in blue. The total experimenting time is 100 seconds.

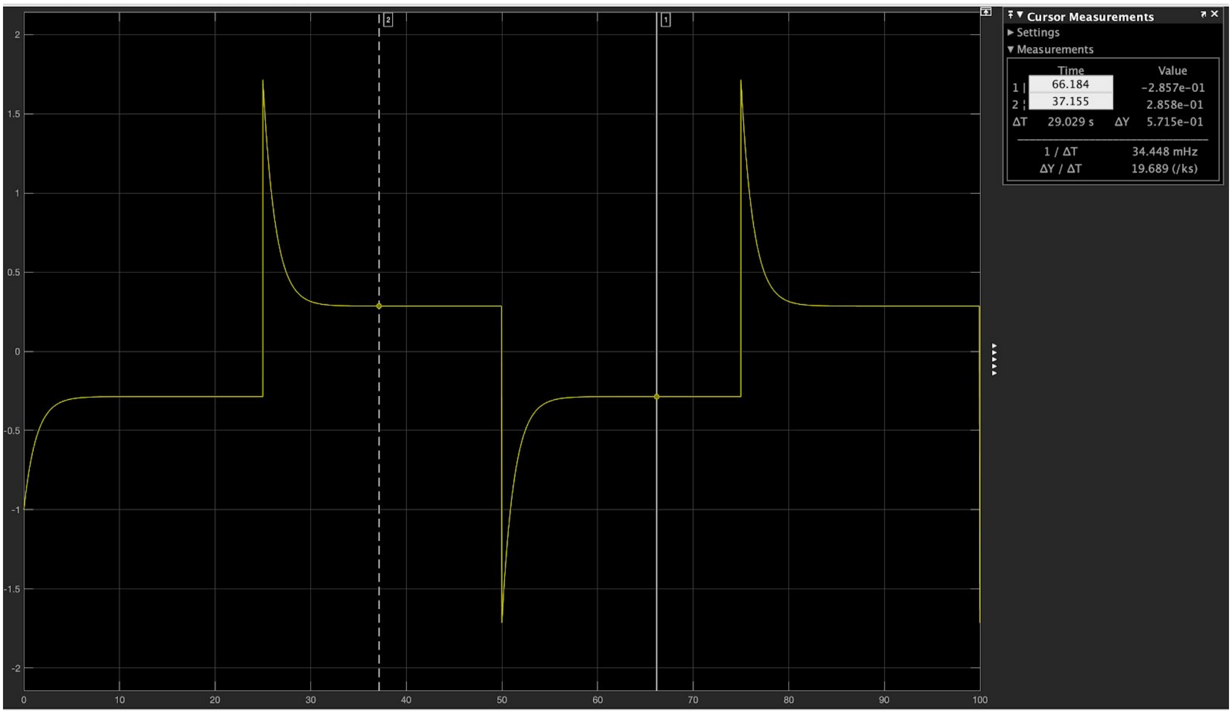
2. Compare and comment on the results in the two figures.

By comparing the output of the two figures with the same process but different in the proportional controller K, we discovered that the greater the control gian K is, the faster the output reaches the steady state and a higher the corresponding output value it converges to.

3. Determine the approximated asymptotic value of the tracking error in the two cases *K=0.1* and *K=1*.

Calculate the tracking error when K=0.1:

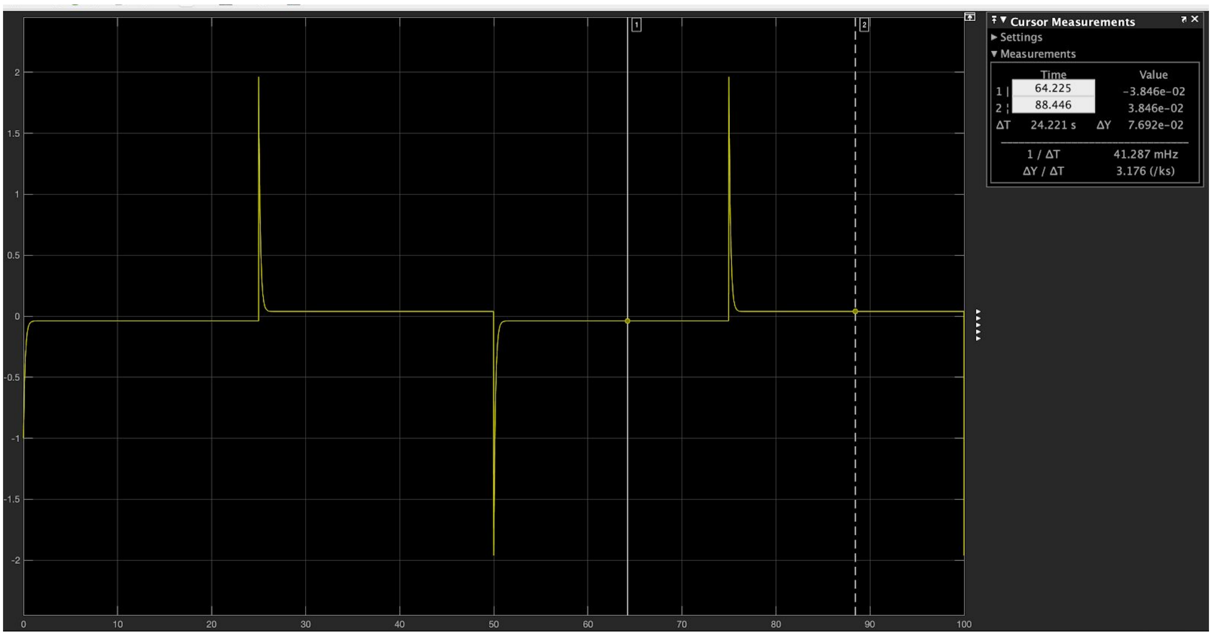
Figure 3.1: The figure below is the tracking error output waveform when K=0.1, with the cursor measuring the approximated asymptotic value.



As we measure from the graph, the asymptotic tracking errors from the graph are -2.857e-01 and 2.858e-01 for a square wave input. So, the absolute value of the error that it achieves when it reaches a steady state is approximately 2.8575e-01.

Calculate the tracking error when K=1:

Figure 3.2: The figure below is the tracking error output waveform when K=1, with the cursor measuring the approximated asymptotic value.



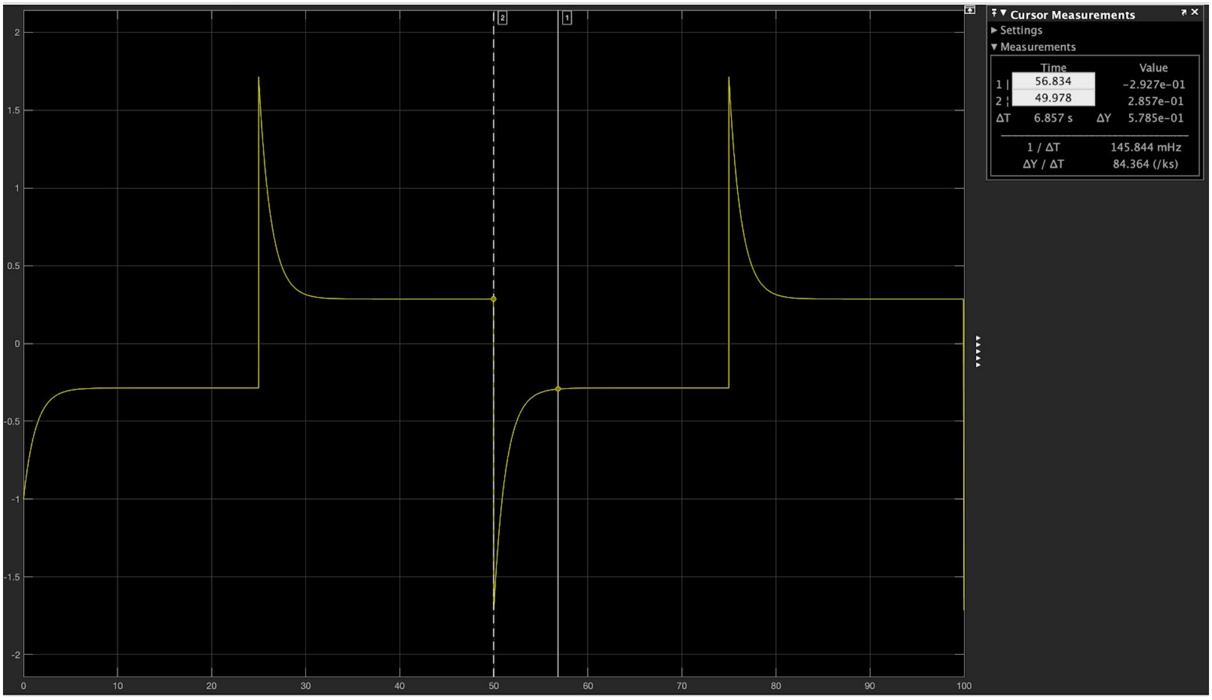
As we measure from the graph, the asymptotic tracking errors from the graph are -3.846e-02 and 3.846e-02 for a square wave input. So, the absolute value of the error that it achieves when it reaches a steady state is approximately 3.846e-02.

By comparing the tracking error when K=0.1 and K=1, we discovered that, when the control gain K increases, the absolute value of the tracking error decreases.

4. Determine the rate of convergence in the two cases *K=0.1* and *K=1*.

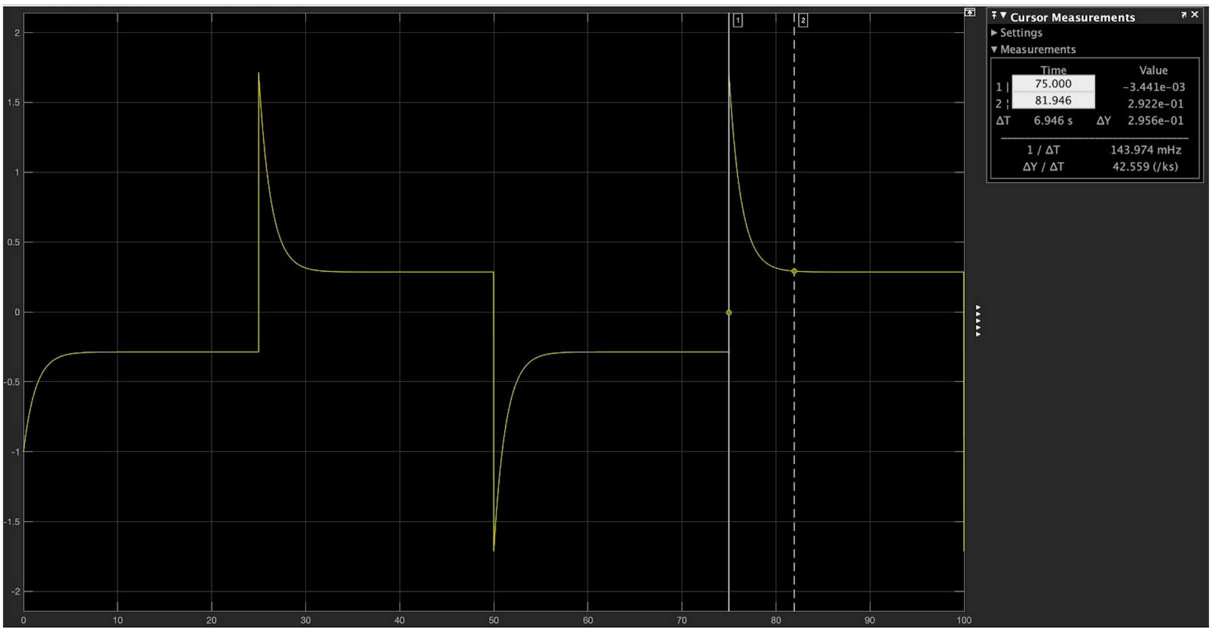
Calculate the tracking error when K=0.1:

Figure 4.1: The figure below is the tracking error output waveform when K=0.1, with the cursor measuring the approximated time for it to reach the steady state when input is negative.



As we measured from the output waveform, it takes 56.834s - 49.978s = 6.856s.

Figure 4.2: The figure below is the tracking error output waveform when K=0.1, with the cursor measuring the approximated time for it to reach the steady state when input is positive.



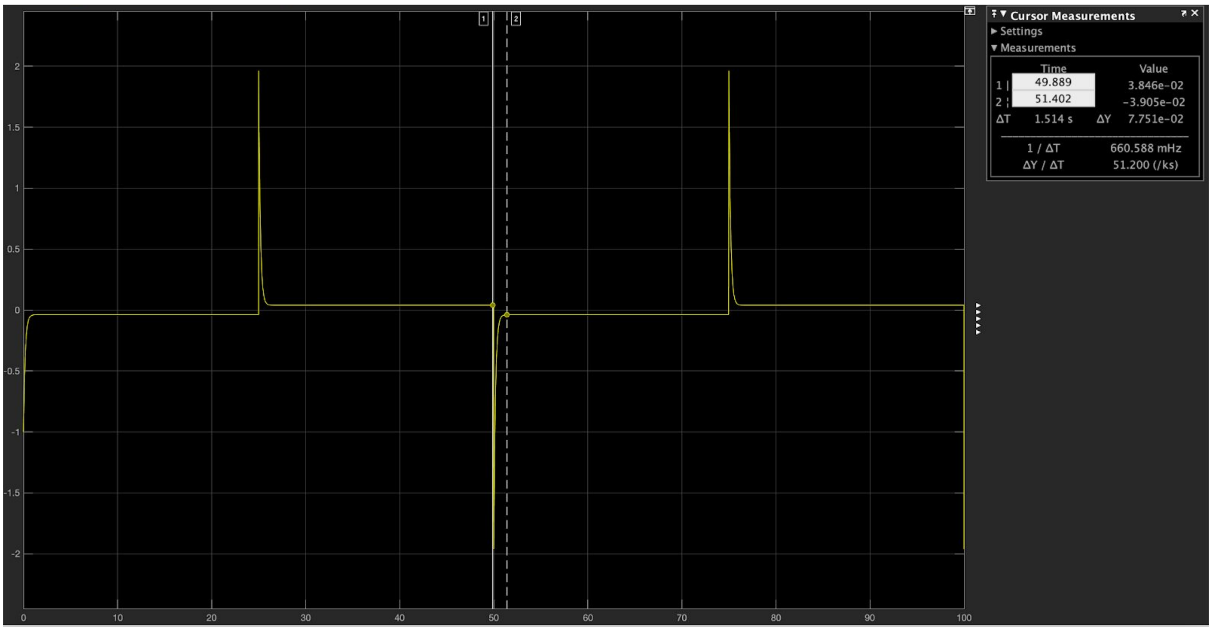
As we measured from the output waveform, it takes 81.946s - 75.000s = 6.946s.

Calculate the rate of convergence:

$$rate\_of\_convergence = \frac{time\ for\ one\ period - time\ of\ non\ convergence}{time\ for\ one\ period} = \frac{50s - 6.946s - 6.856s}{50s} = 0.72396$$

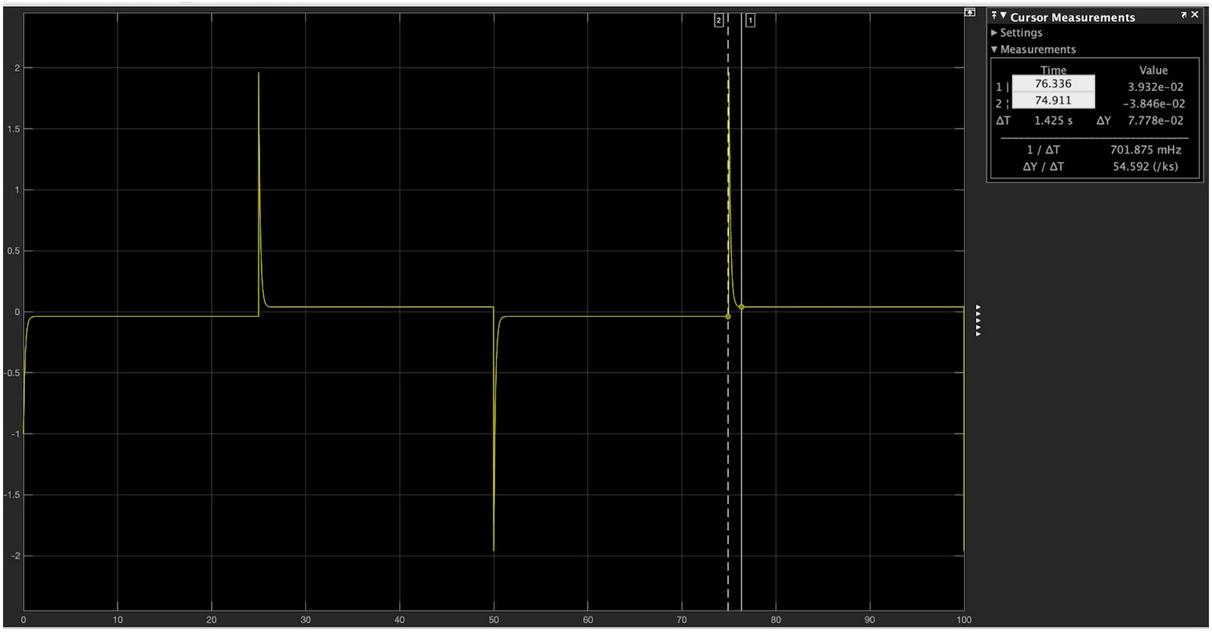
Calculate the tracking error when K=1:

Figure 4.3: The figure below is the tracking error output waveform when K=1, with the cursor measuring the approximated time for it to reach the steady state when input is negative.



As we measured from the output waveform, it takes 51.402s - 49.889s = 1.513s.

Figure 4.4: The figure below is the tracking error output waveform when K=1, with the cursor measuring the approximated time for it to reach the steady state when input is positive.



As we measured from the output waveform, it takes 76.336s - 74.911s = 1.425s.

Calculate the rate of convergence:

$$rate\_of\_convergence = \frac{time\ for\ one\ period - time\ of\ non\ convergence}{time\ for\ one\ period} = \frac{50s - 1.513s - 1.425s}{50s} = 0.94124$$

By comparing the rate of error when K=0.1 and K=1, we discovered that, when the control gain K increases, the rate of convergence increases.

5. Make concluding remarks. Do you think that proportional control is an adequate means to regulate the speed of a DC motor? Do you have any ideas for improvement?

From the calculation and comparison between the tracking error and rate of convergence from the previous part between the K=0.1 and K=1 process, we discovered that, as the control gain K increases, the tracking error decreases and the rate of convergences increases. So, the proportional control is an adequate means to regulate the speed of a DC motor. As we tried to continue increasing the control gain to a larger value as K=2 and K=10, we discovered that the output is even better.

Figure 5.1 The motor speed and reference signal for K=2

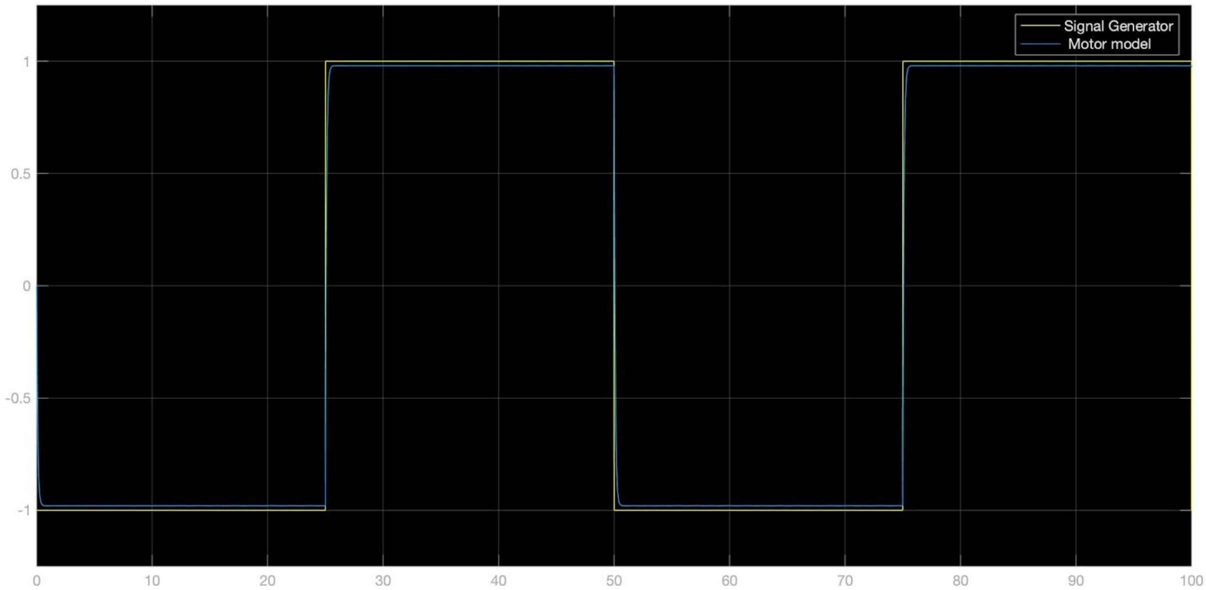


Figure 5.2: The figure below is the tracking error output waveform when K=2, with the cursor measuring the approximated asymptotic value.

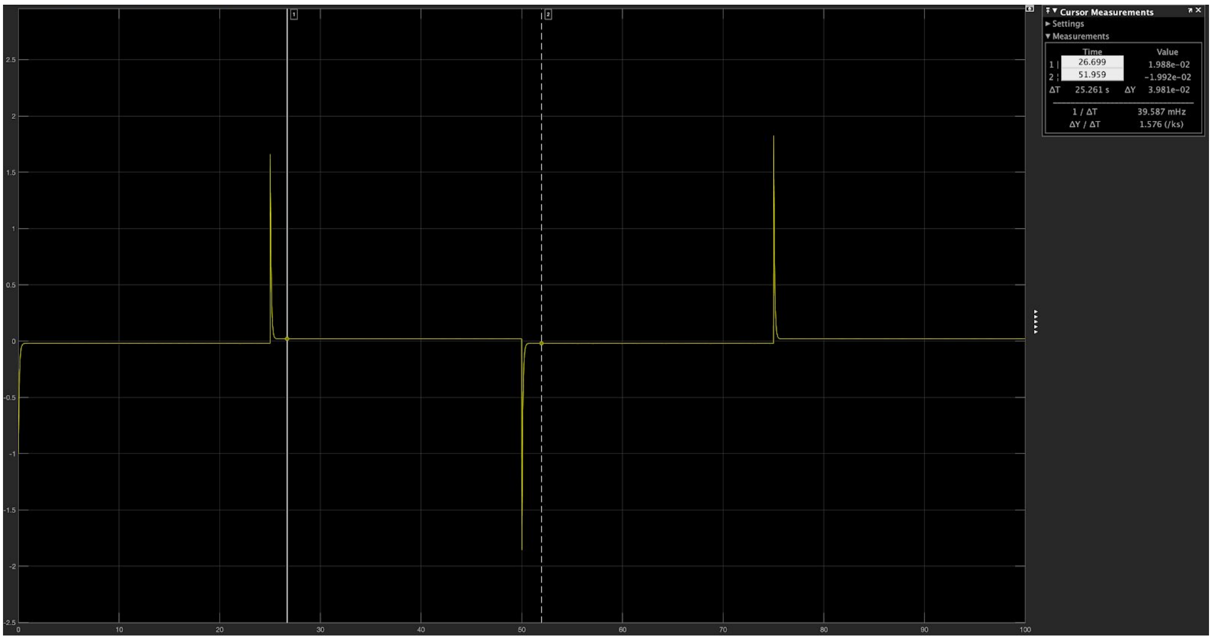


Figure 5.3: The figure below is the tracking error output waveform when K=2, with the cursor measuring the approximated time for it to reach the steady state when input is negative.

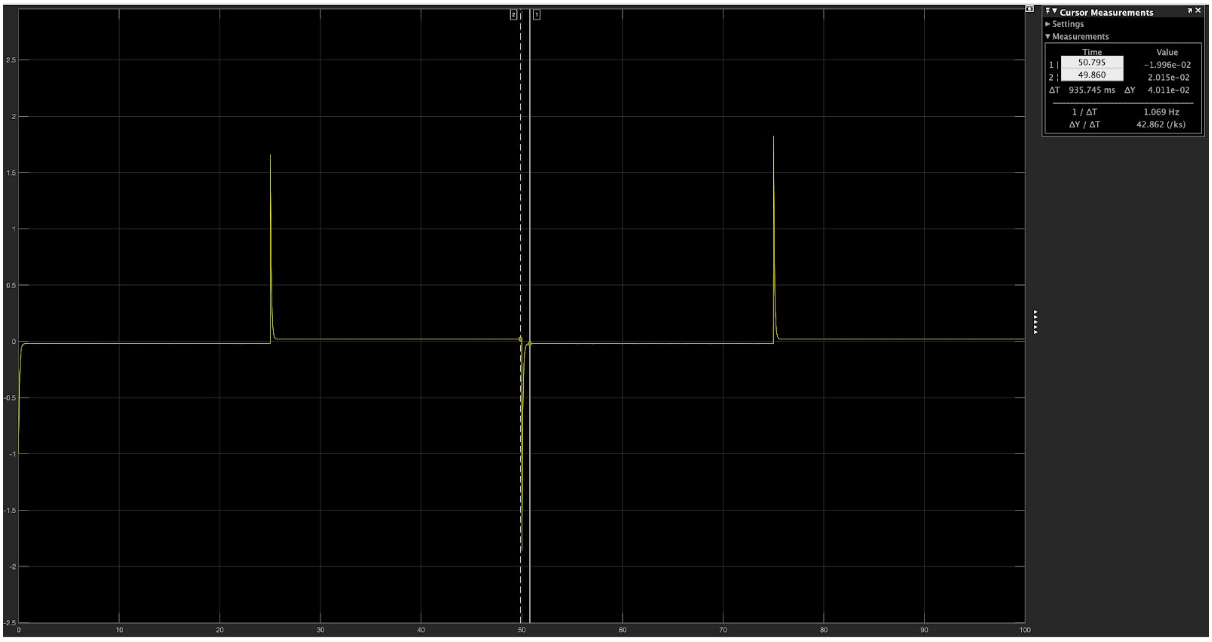
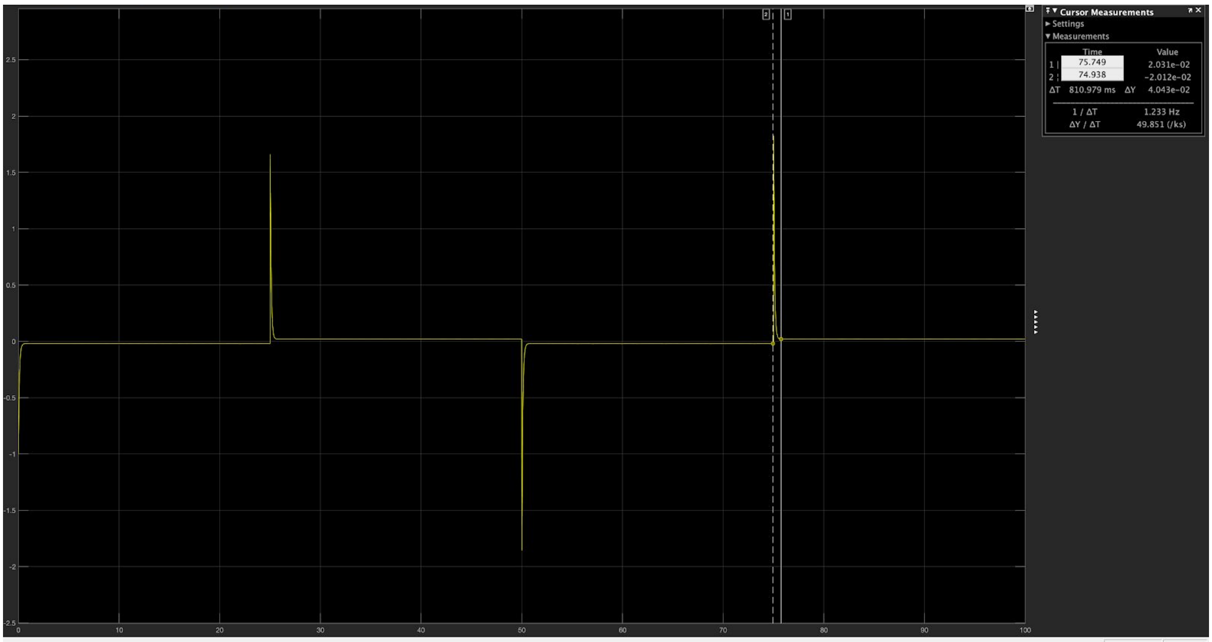


Figure 5.4: The figure below is the tracking error output waveform when K=2, with the cursor measuring the approximated time for it to reach the steady state when input is positive.



Calculations when K=2:

As we measure from the graph, the asymptotic tracking errors from the graph are -1.992e-02 and 1.998e-02 for a square wave input. So, the absolute value of the error that it achieves when it reaches a steady state is approximately 1.995e-02.

As we measured from the error waveform, it takes 50.795s - 49.860s = 0.935s and 75.749s - 74.938s = 0.811s.

$$rate\_of\_convergence = \frac{time\ for\ one\ period - time\ of\ non\ convergence}{time\ for\ one\ period} = \frac{50s - 0.935s - 0.811s}{50s} = 0.96508$$

According to the calculation above, when K=2, the error is minimized to an absolute value of 0.02, and the rate of convergence reaches 0.965, which are all better than the values when K=1.

Figure 5.5 The motor speed and reference signal for K=10

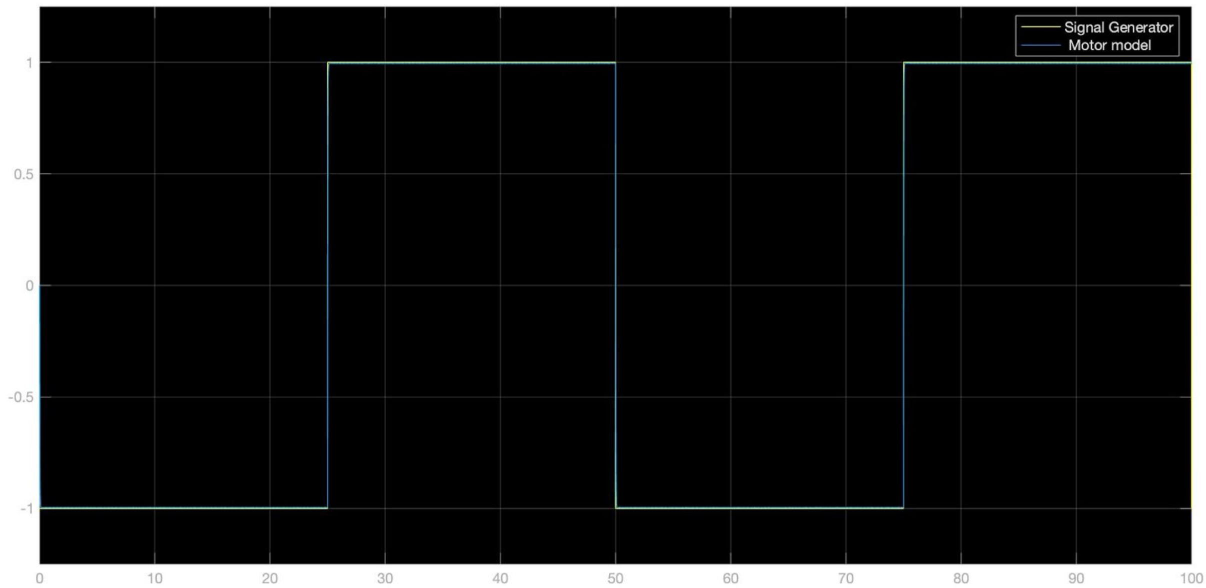


Figure 5.6: The figure below is the tracking error output waveform when K=10, with the cursor measuring the approximated asymptotic value.

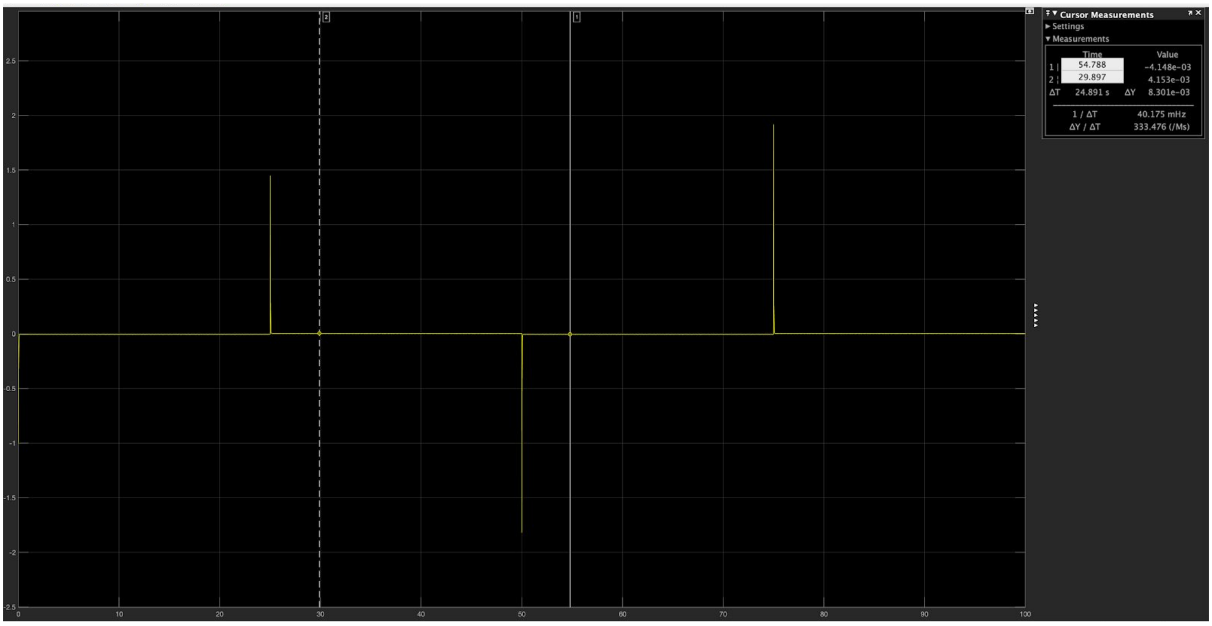


Figure 5.7: The figure below is the tracking error output waveform when K=10, with the cursor measuring the approximated time for it to reach the steady state when input is negative.

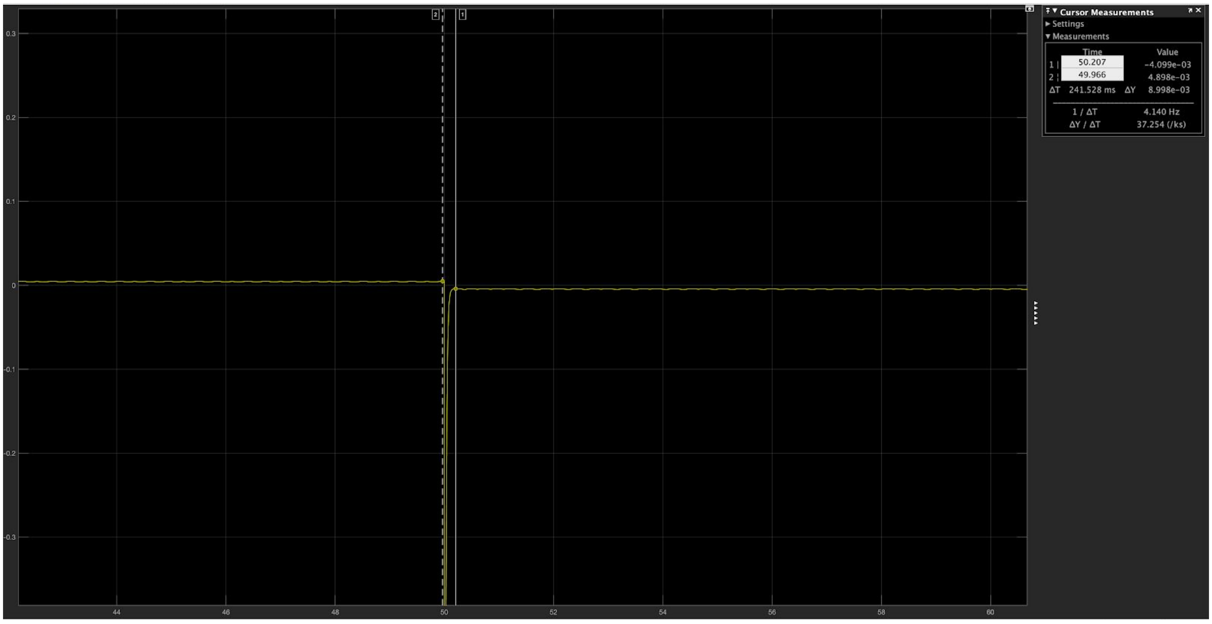
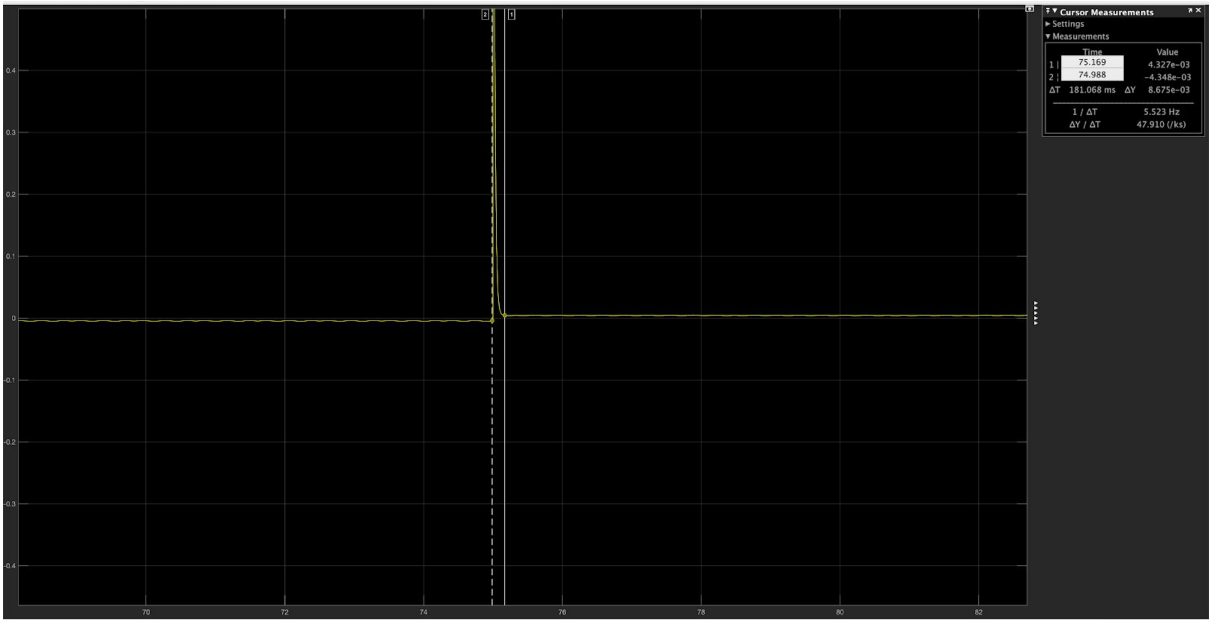


Figure 5.8: The figure below is the tracking error output waveform when K=10, with the cursor measuring the approximated time for it to reach the steady state when input is positive.



Calculations when K=10:

As we measure from the graph, the asymptotic tracking errors from the graph are -4.148e-03 and 4.153e-03 for a square wave input. So, the absolute value of the error that it achieves when it reaches a steady state is approximately 4.150e-03.

As we measured from the error waveform, it takes 50.207s - 49.966s = 0.241s and 75.169s - 74.988s = 0.181s.

$$rate\_of\_convergence = \frac{time\ for\ one\ period - time\ of\ non\ convergence}{time\ for\ one\ period} = \frac{50s - 0.241s - 0.181s}{50s} = 0.99156$$

According to the calculation above, when K=10, the error is minimized to an absolute value of 0.004, which is negligible and the rate of convergence reaches 0.99, which is very high. As a result, in order to keep improving the output, we can increase the control gain K to a corresponding higher value, which will result in a lower tracking error, and higher rate of convergence. So, by increasing the control gain K, the output will reach the steady state faster and with a closer final value towards the input, which is a corresponding higher speed of the motor as the output.