

PSDE5009

by Phillip Rowe

FILE	7557_PHILLIP_ROWЕ_PSDE5009_158265_286632943.DOCX		
TIME SUBMITTED	11-MAY-2018 12:28PM (UTC+0100)	WORD COUNT	2392
SUBMISSION ID	88973903	CHARACTER COUNT	12507

Phillip Rowe
Electrical and electronic engineering
PSDE 5009
Electrical Principles, Methods and
Simulation



Prifysgol Cymru
Y Drindod Dewi Sant
University of Wales
Trinity Saint David

Table of Contents

Introduction	3
First order systems.....	4
DC motor model analysis	9
Types of motor speed controllers – P controller	14
Types of motor speed controllers – PI controller	17
Types of motor speed controllers – PID controller	20
Types of motor speed controllers – P, PI and PID controller	22
Conclusions and recommendations.....	26

Introduction

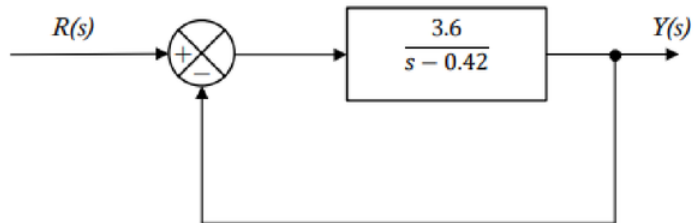
The main aims of this assignment are to enhance individual knowledge and also critical analytical skills while evaluating and analysing control systems with the use of simulation via developed models.

Some of the objectives of this assignment include:

- Being able to convert mathematical data from circuits into a format which is useable for matlab.
- Being able to develop mathematical models by using transfer functions and then use these in order to predict performance by using matlab.

First order systems

Shown below in figure 1 is the first order system which will be analysed in this task. Firstly the overall transfer function will be produced and the argand diagram will be plotted.



(Figure 1)

$$G(s)/1+G(s)H(s)$$

$$G(s) = 3.6/s-0.42$$

$$H(s)=1$$

$$T(s) = 3.6/(s-0.42)/1+3.6/(s-0.42)$$

Multiplying top and bottom by (s-0.42)

$$3.6/(s-0.42) * (s-0.42) / ((s-0.42)(1+3.6/(s-0.42)))$$

$$T(s) = 3.6/s-0.42+3.6$$

$$= 3.6/s+3.18$$

So the pole would be on the left hand side making it a stable system. If the s is taken away then without this feedback the system would be unstable.

Using the matlab code below in figure 2 a root locus has been generated based off the mathematical workings carried out.

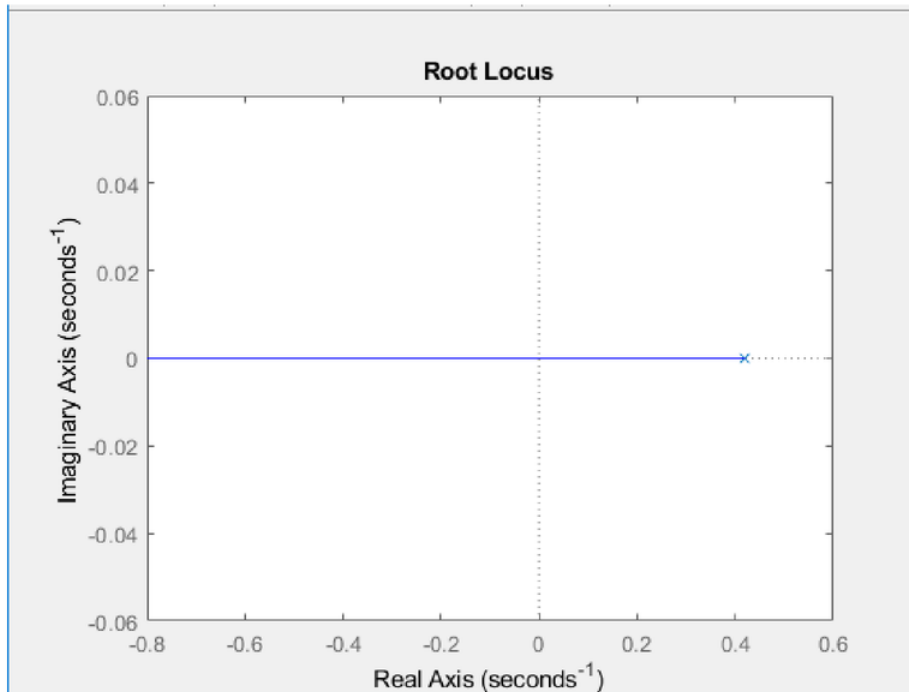
```

1
2 - num=[3.6];
3 - den=[1 -0.42];
4 - rlocus(num,den);
5

```

(Figure 2)

Below in figure 3 is the root locus based off the num and den above in figure 2.



(Figure 3)

Now using mathematical formulas the minimum value of K(gain) will be calculated.

$$T1(s) = 3.6/s - 0.42 + 3.6k$$

$$= 3.6/s + (3.6k - 0.42)$$

*explanation needed here, new block
diagram should be drawn*

$$\text{Pole is found at } s + 3.6k - 0.42 = 0$$

$$\text{For pole at } s=0$$

$$0 + 3.6k - 0.42 = 0$$

$$3.6k = 0.42$$

$$K = 0.117 \text{ So the minimum value of K is 0.117}$$

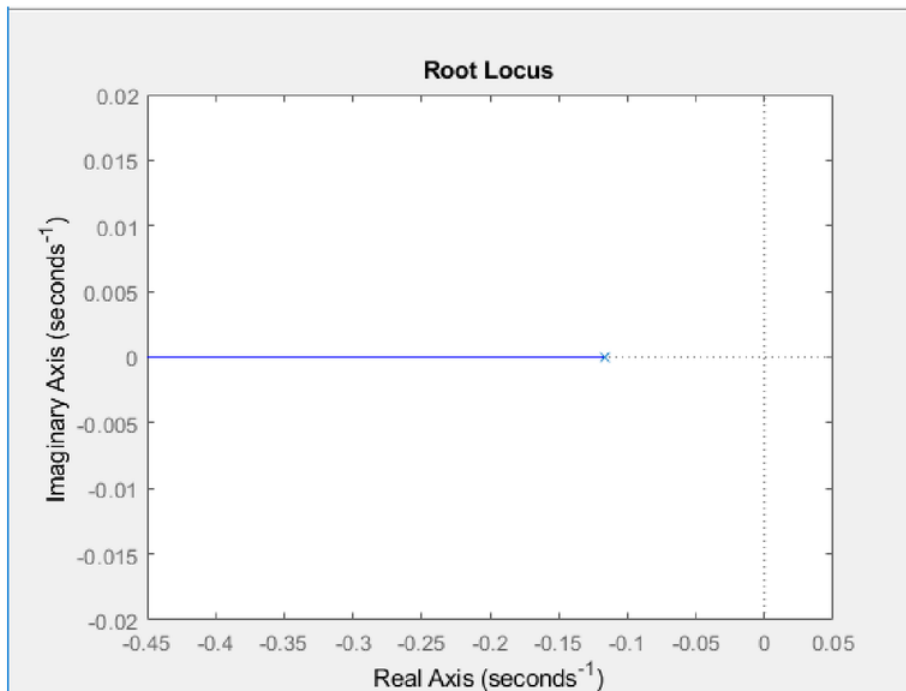
Now using the num and den below in figure 4 the root locus will be created using matlab.

```

1
2 - num=[3.6];
3 - den=[1 0.117];
4 - rlocus(num,den);
5
```

(Figure 4)

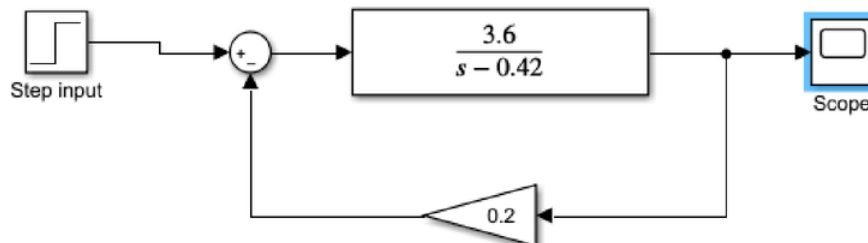
As shown below in figure 5 is the root locus of figure 4 above. In figure 5 it displays that the pole is on the left hand side which is the minimum value K needs to be for this to be stable.



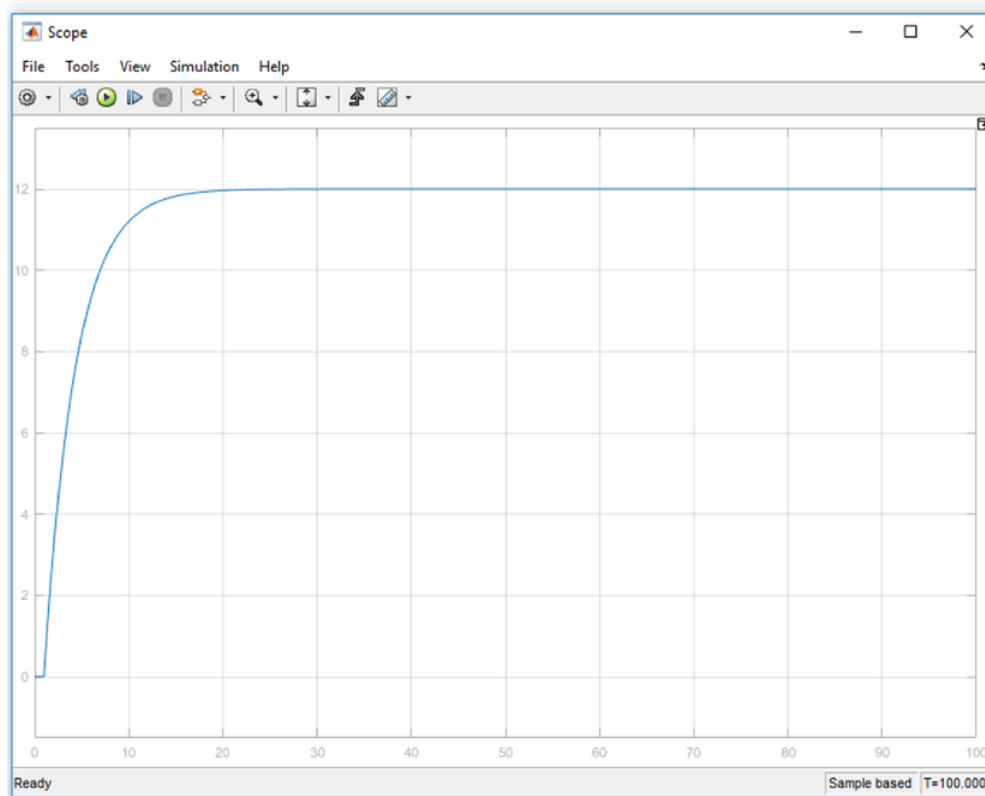
(Figure 5)

Now using Simulink a step input will be inputted into the circuit and the output will be analysed to see if the system is stable or unstable when changing the value of K.

Below in figure 6 the gain has been set to 0.2 which is above the minimum value of K and the response clearly shows in figure 7 that this system is a stable system.

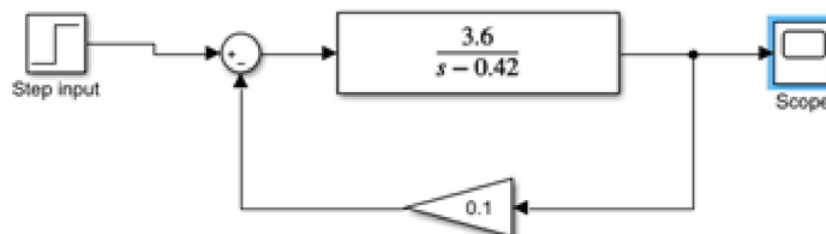


(Figure 6)

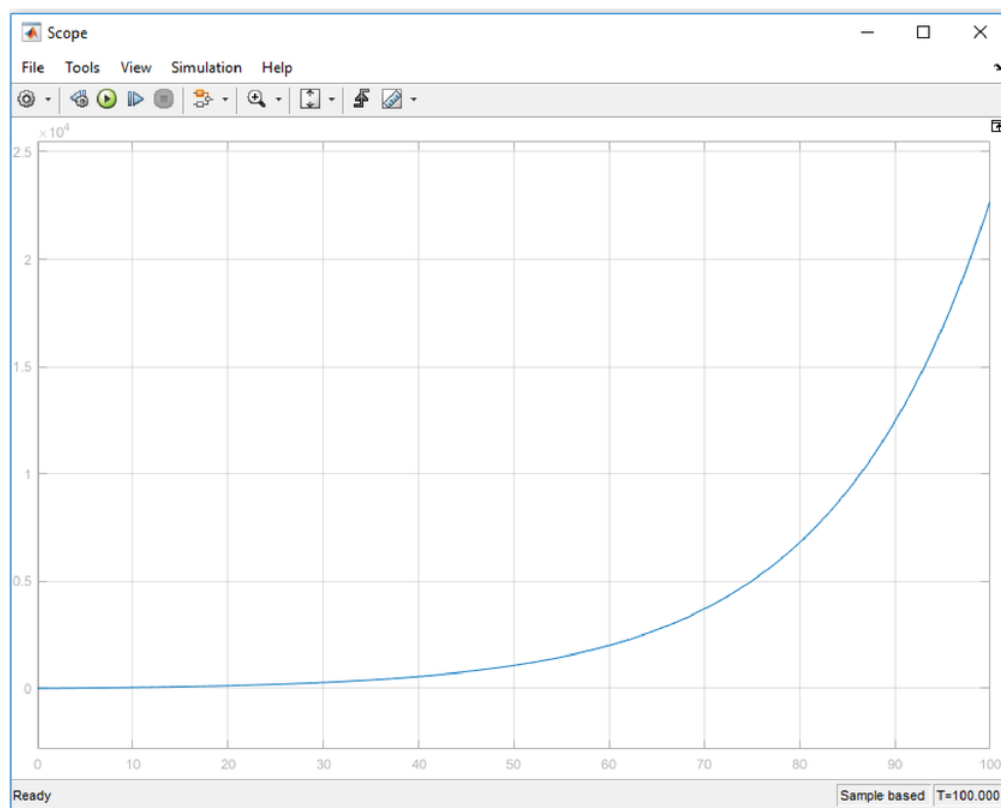


(Figure 7) *you could have calculated the overall gain and time constant for comparison*

Now that the response of an ideal system has been shown the value of K will be changed so that it makes the system unstable so that the response can be seen. So below in figure 8 is the updated system showing the value of K being changed from 0.2 to 0.1. The response of this can be seen in figure 9 and what this shows is a very unstable system.

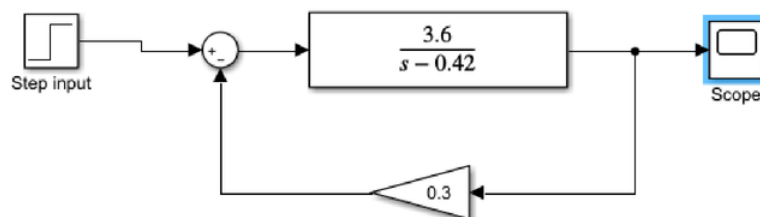


(Figure 8)

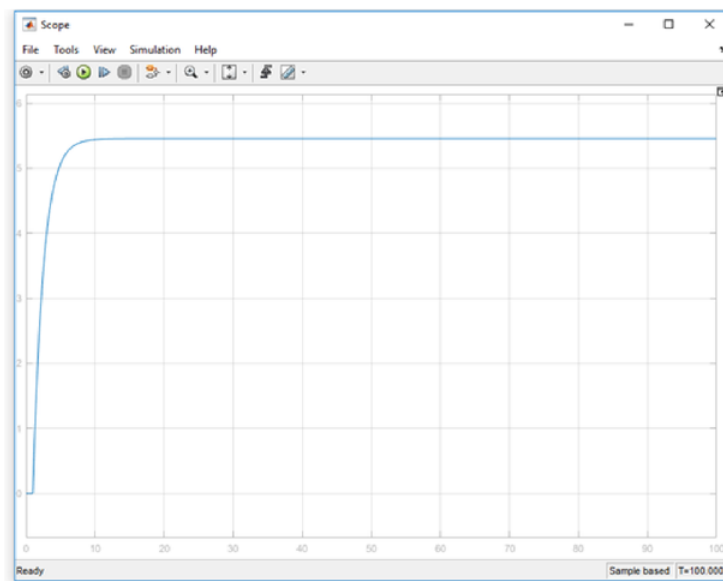


(Figure 9)

The third and final example of changing K can be seen below in figure 10. Figure 10 shows the value of K has been changed to 0.3. Looking at the scope response it shows an even more stable response compared to when the value was 0.2. This response can be shown in figure 11.



(Figure 10)



(Figure 11)

DC motor model analysis

Below for this task are all of the values needed to carry out this task.

- Armature resistance = 7.8 Ohms
- Armature inductance = 0.05H
- Rotor inertia = 0.000214 Kg m²
- Friction coefficient = 15.519 x 10⁻⁶ Nm/rad/s
- Motor voltage constant = 0.0983 V/rad/s
- Motor torque constant = 0.09 Nm/A
- Maximum armature voltage = 40V
- Load torque = 0.025 Nm

$G(s) = \frac{K_t[L_a]}{s^2 + s(b/J + R_a/L_a) + R_a b + K_t K_e/L_a}$ *equation is difficult to read where the equation needed for comes from*

Using the values above they will be put with their associated letters in the list below.

$R_a = 7.8 \text{ Ohms}$ $K_e = 0.0983$

$L_a = 0.05\text{H}$

$J = 0.000214$

$b = 15.519 \times 10^{-6}$

$K_t = 0.09$

Now using mathematical equations the values needed will be calculated such as the gain.

$$K_t/LaJ = 0.09/0.05 \times 0.000214 = 8411.215$$

$$b/J + Ra/La = (15.52 \times 10^{-6}/0.000214) + (7.8/0.05) = 156.07$$

$$Rab + K_t K_e/LaJ = (7.8 \times 15.519 \times 10^{-6} + 0.09 \times 0.0983/0.05 \times 0.000214) = 838.14$$

Using formula but rearranged to find ω_n

$$C_f = K_{wn}^2/s^2 + 2\zeta\omega_n + \omega_n$$

$$\omega_n = \sqrt{838.135} = 28.96 \text{ rads/s}$$

$$2\zeta\omega_n = 156.07$$

$$\zeta = 156.07/(2 \times 28.96) = 2.7$$

$$\text{The overall gain is } = 8411.25/838.135 = 10$$

Using the quadratic equation the poles will be found and a argand diagram will be sketched to show the poles.

$$A = 1$$

$$B = 156.073$$

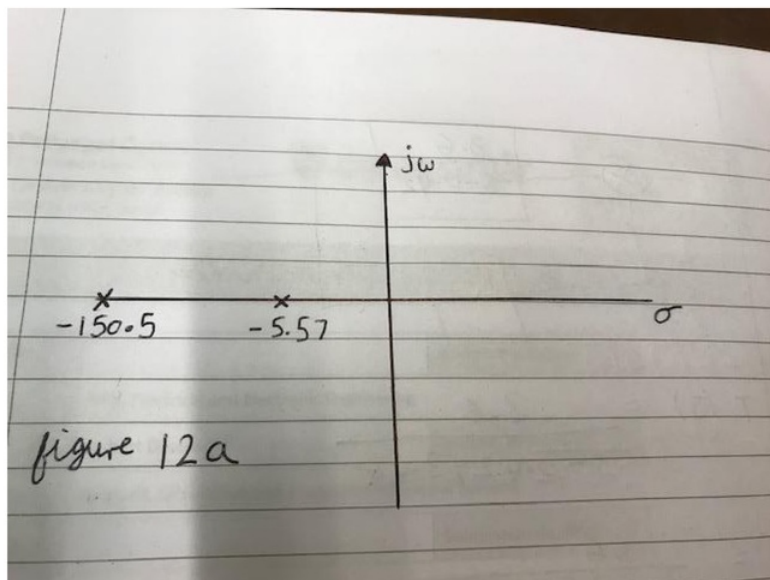
$$C = 838.14$$

$$S = -b \pm \sqrt{b^2 - 4ac} / 2a$$

$$S = -156.073 \pm \sqrt{156.073^2 - 4 \times 1 \times 838.14} / 2$$

$$= -5.6 \text{ and } -150.5$$

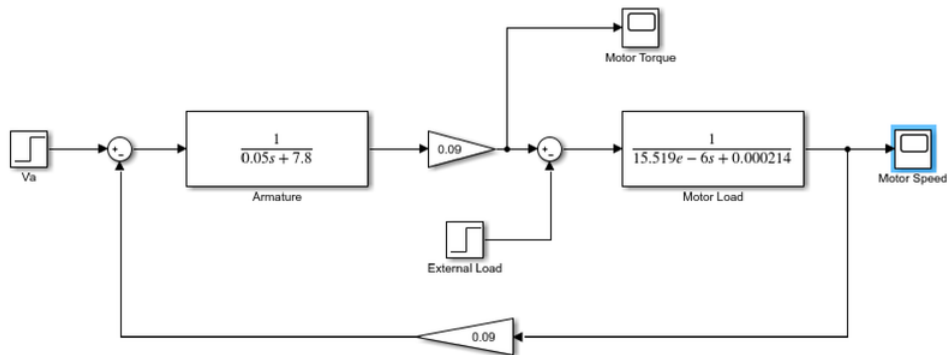
Below in figure 12a is the argand diagram.



(Figure 12a)

poor image, spoils the presentation of the report

The motor model is shown in figure 12b below.



(Figure 12b)

Before looking at the no load speed and the loaded speed of the motor should run at the speed with be calculated mathematically first.

No load speed.

$$\Omega_{NL}(b) = K_t + V_a / b R_a + K_t K_e$$

$$\Omega_{NL}(b) = 0.09 + 40 / (15.51 \times 10^{-6}) \times (7.8 + 0.09 \times 0.0983)$$

$$\Omega_{NL}(b) = 240.9 \text{ rads/s}$$

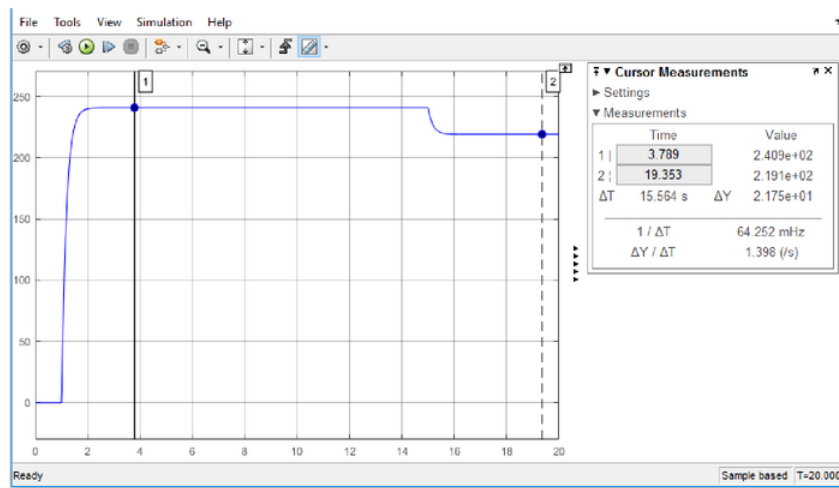
Loaded speed

$$\Omega_L = V_a - R_a T_l / K_t / R_a b / K_t + K_e$$

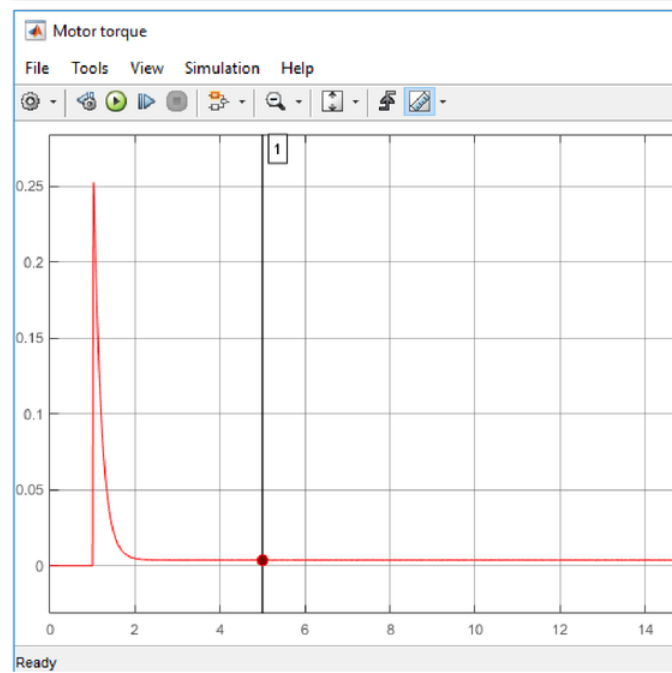
$$\Omega_L = 40 - 7.8 \times 0.025 / 0.09 / 7.8 (15.52 \times 10^{-6}) + 0.0983 / 0.09 = 220 \text{ rads/s}$$

Now that the mathematical values have been calculated the matlab values shall be simulated. Below in figure 13 is the simulated response for both no load speed and also loaded speed.

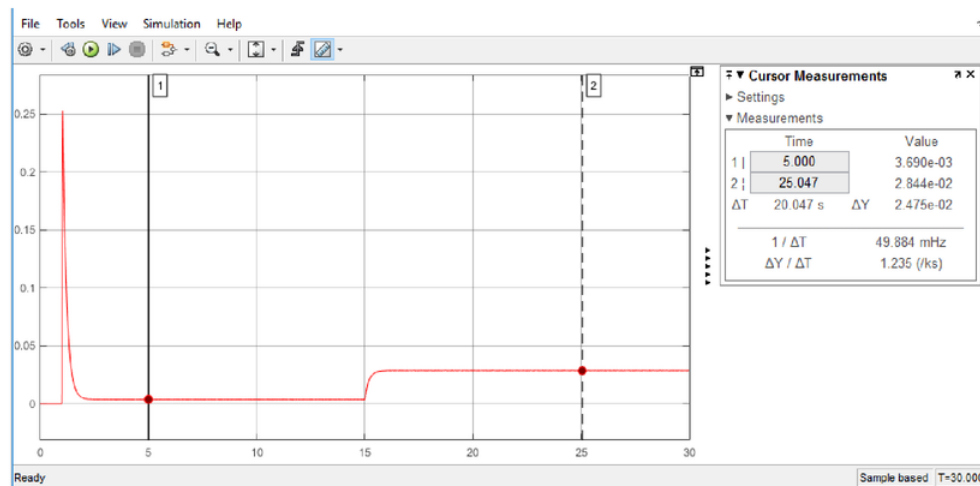
Extracting values from figure 13a the no load speed is 240.9 rads/s and the loaded speed is 219.1 rads/s. In figure 13b is the no load torque response also. In figure 13c shows the loaded torque response. As seen there is a difference between figure 13b and 13c which is expected since the disturbances has been added.



(Figure 13a)



(Figure 13b)



(Figure 13c)

Now that has been calculated the stall torque will now be worked out.

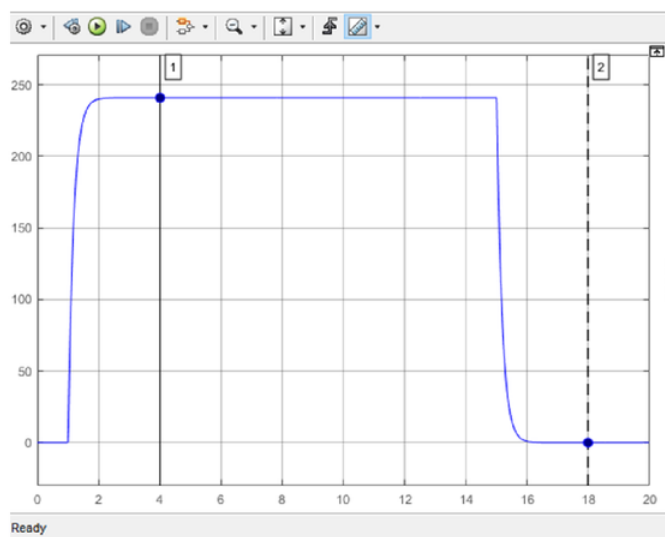
$$T_m = K_t V_a / R_a$$

$$T_m = 0.09 \times 40 / 7.8$$

$$T_m = 0.277$$

Now that this value has been calculated the mathematical model will also be simulated. This is displayed below in figure 14. The matlab value for the stall torque is 0.286.

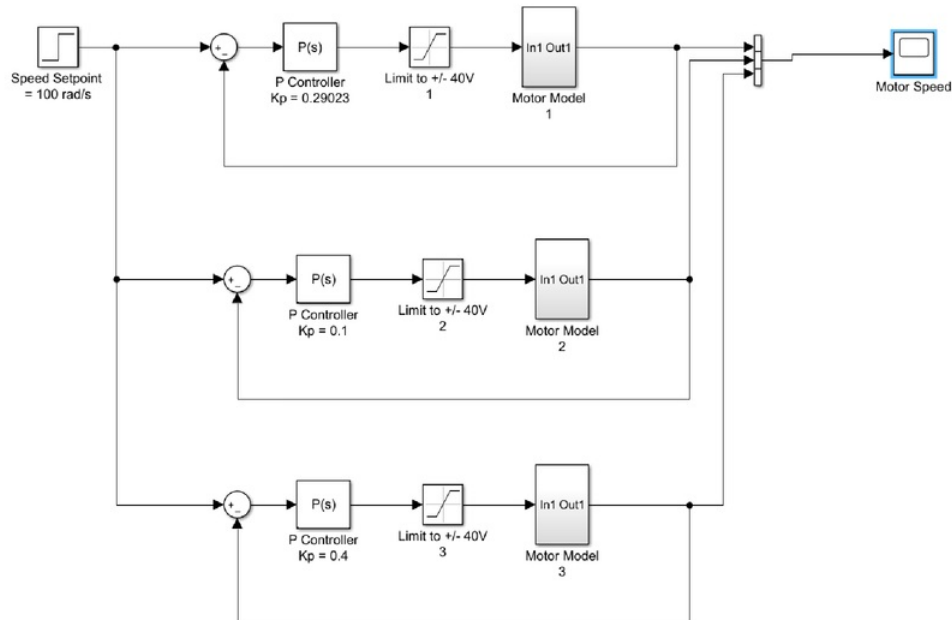
How come?



(Figure 14)

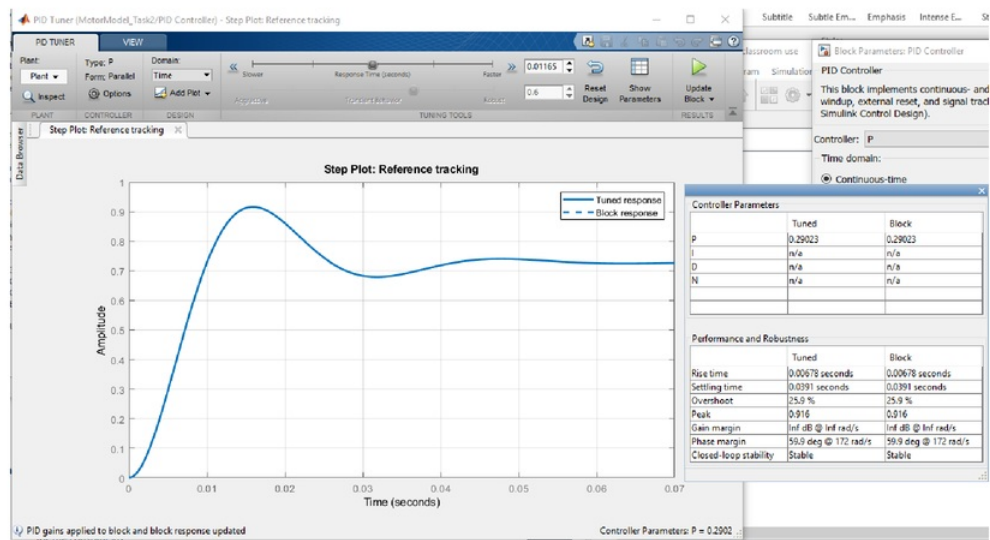
Types of motor speed controllers – P controller

For this task the motor model has been put into a subsystem, a P controller has been added with a summing junction also before it and there is a limiter set in the circuit to limit the armature voltage. This can be seen below in figure 15.



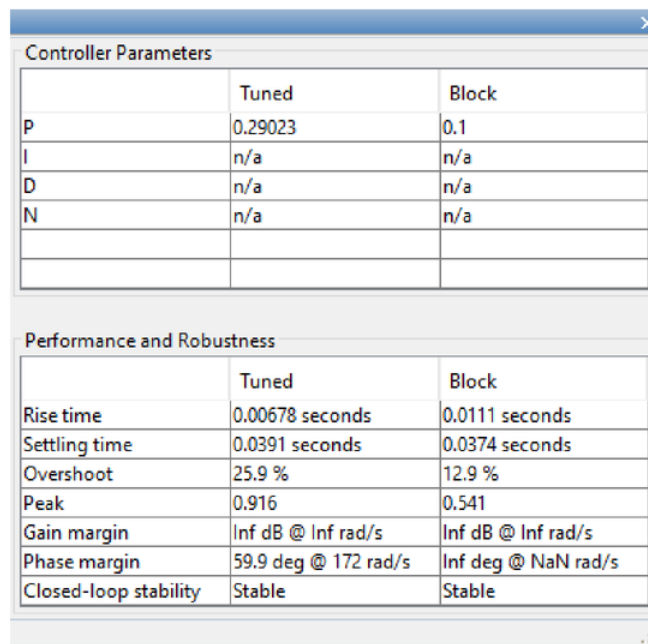
(Figure 15)

The first P controller was tuned to its ideal value which can be seen below in figure 16.



(Figure 16)

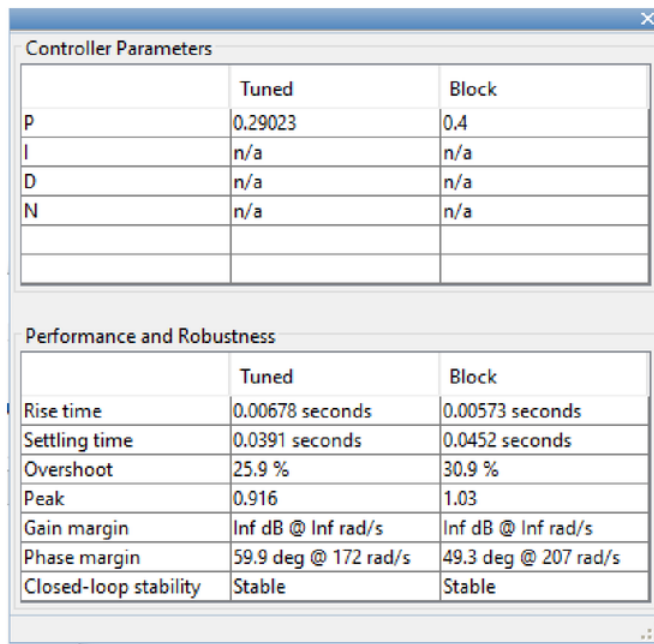
Since the first value of P was tuned to its perfect ideal value the other two were changed to show the difference in response. Below in figures 17 and 18 you can see the differences between the tuned and changed values. The 3 different values were 0.1, 0.29023 and 0.4. Using figures 17 and 18 the differences will be discussed and then summarised. The tuned value has a rise time of 0.00678 seconds compared to the other two which take 0.0111 seconds and 0.00573. This shows that the tuned controller is the second best on rise time however when looking at settling time the Kp value of 0.1 has the best settling time at 0.0374 compared to the tuned controller which takes 0.0391. The tuned controller also has the second best settling time at 25% compared to the 0.4 KP controller which has 30%. However after looking at all of these results looking at the graphical results in figure 19 it clearly shows that motor model 1 is the best since it is the only one. From this the conclusion can be made that the lower Kp is the less aggressive the response is and the greater Kp is then the harsher the initial response is and also the bigger the overshoot is. different values of KP, increasing KP leads to bigger overshoot but does get you closer to the 100 target.



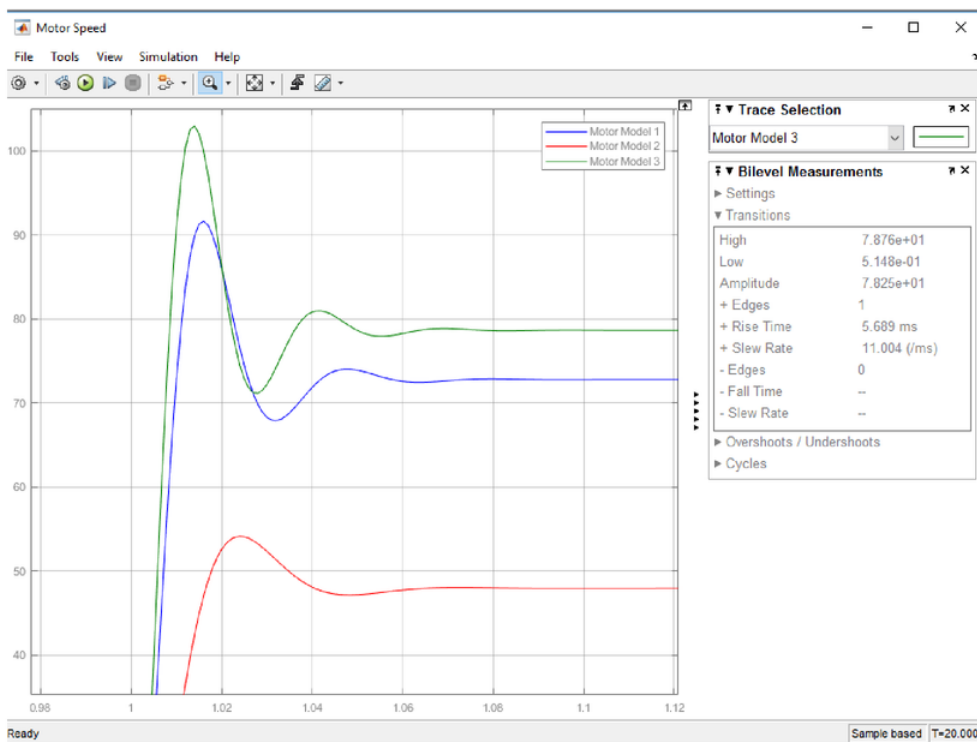
Controller Parameters		
	Tuned	Block
P	0.29023	0.1
I	n/a	n/a
D	n/a	n/a
N	n/a	n/a

Performance and Robustness		
	Tuned	Block
Rise time	0.00678 seconds	0.0111 seconds
Settling time	0.0391 seconds	0.0374 seconds
Overshoot	25.9 %	12.9 %
Peak	0.916	0.541
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	59.9 deg @ 172 rad/s	Inf deg @ NaN rad/s
Closed-loop stability	Stable	Stable

The steady state error should be your main aim to minimise the error (Figure 17)



(Figure 18)

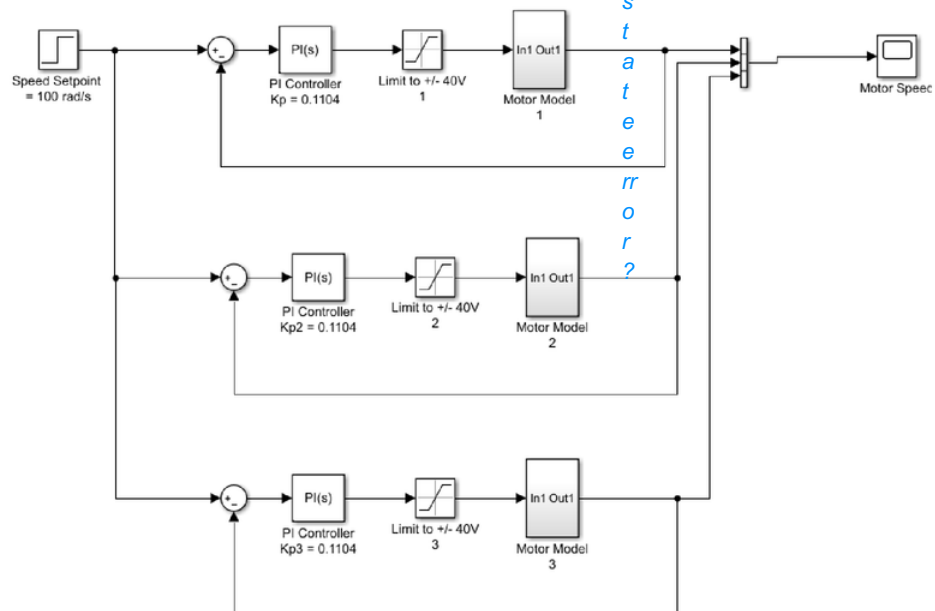


(Figure 19)

Kp value	Response
0.29	Good response, only 25% overshoot, quick rise time and good settling time.
0.1	Poor response, little overshoot due to poor response, quick settling time.
0.4	Quickest rise time, larger overshoot and poor settling time.

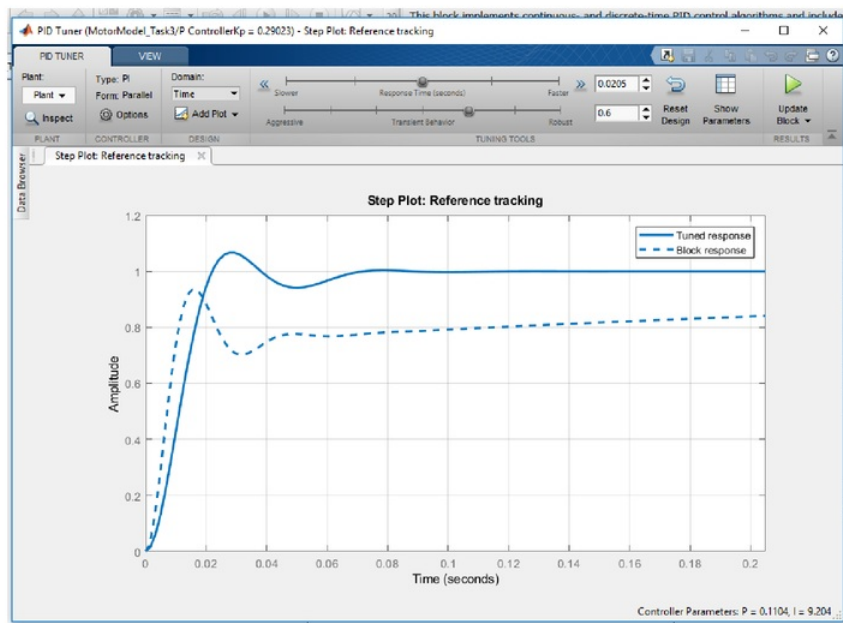
Types of motor speed controllers – PI controller

Following below in figure 20 is the updated version of the motor model with all PI controllers in them.



(Figure 20)

Shown below in figure 21 is the auto tuning of the first PI controller.



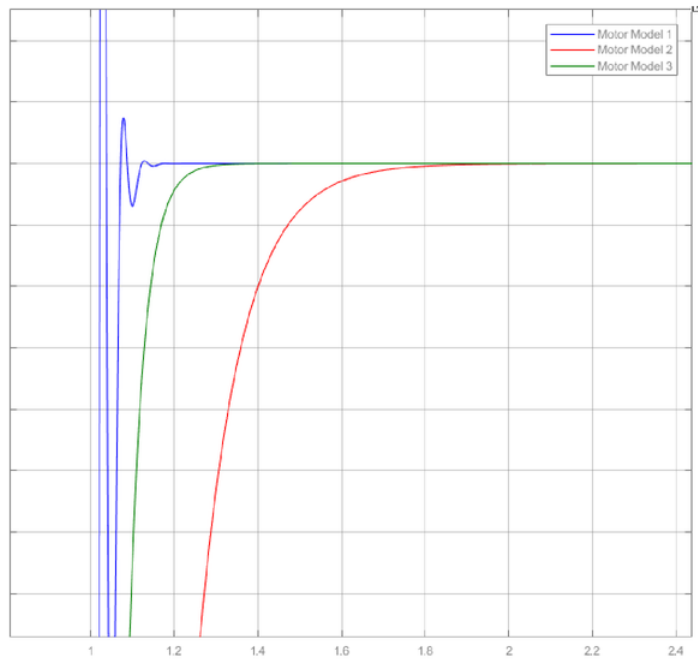
(Figure 21)

Below in figure 22 displays the tuned PI controller and also the changed PI controller. As seen in figure 22 the tuned controller has a better rise time but the settling time is worse and also the overshoot is higher.

Controller Parameters		
	Tuned	Block
P	0.11044	0.11044
I	9.2043	2
D	n/a	n/a
N	n/a	n/a
Performance and Robustness		
	Tuned	Block
Rise time	0.0145 seconds	0.159 seconds
Settling time	0.0638 seconds	0.329 seconds
Overshoot	6.71 %	0 %
Peak	1.07	0.998
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	60 deg @ 97.5 rad/s	116 deg @ 56.8 rad/s
Closed-loop stability	Stable	Stable

(Figure 22)

Shown below in figure 23 are the responses of the three different controllers. Looking at the graphs motor model 1 is the best it does overshoot more and needs some settling time however it still reaches its desired settling value before both of the other motor models. Motor model 2 is the worst due to its very slow rise time. Motor model 3 is the close second due to the fact that it takes longer to rise but settles a little after motor model 1.

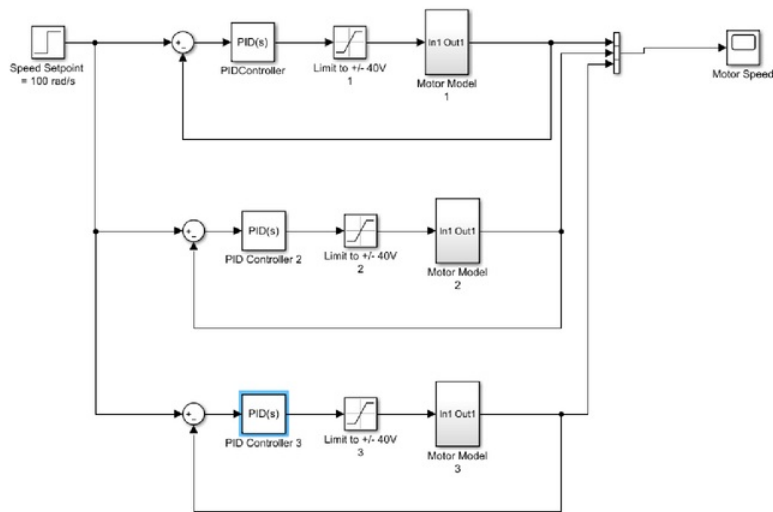


(Figure 23)

[Summary table for overshoot and rise time is needed here for different values of \$K_i\$ for comparison](#)

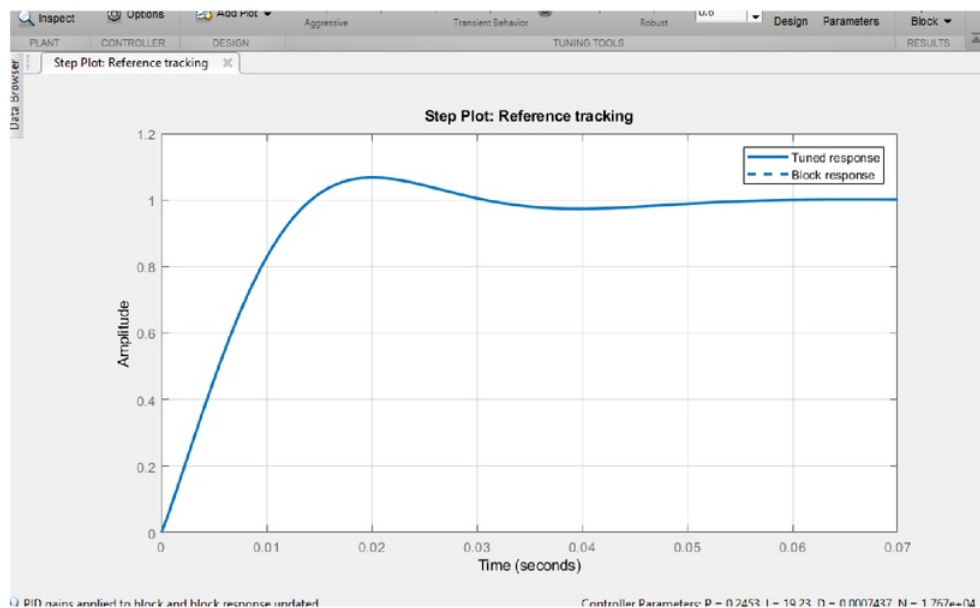
Types of motor speed controllers – PID controller

Below in figure 24 is the motor model for the PID controller. For this model all of the K_P and K_i values are kept the same but the K_d values are all changed.



(Figure 24)

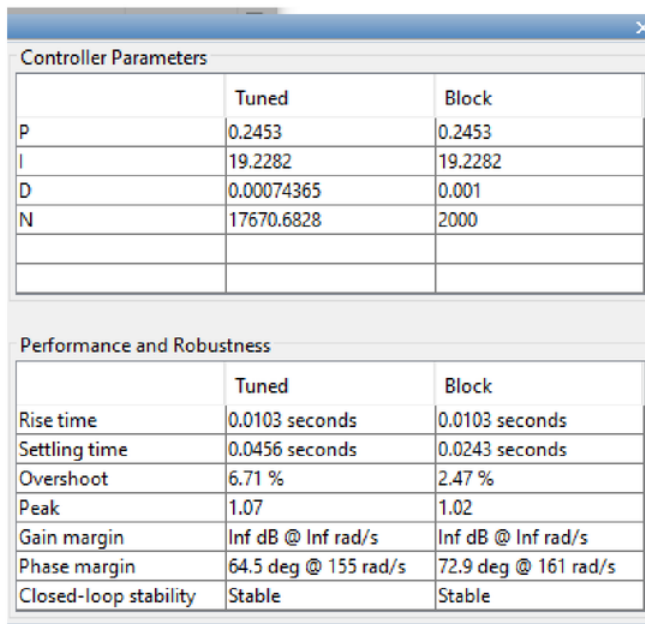
Below in figure 25 is the tuning of the first controller collecting the desired values.



(Figure 25)

Below in figure 27 shows the responses of the 3 different controllers. Looking at the responses of the motor models you can see that as the values of K_d decrease the settling time and overshoot. This is the case due to K_d was 0.0007 for motor model 1 and 0.007 for motor model 2 and 0.0001 for motor model 3. This shows a decay in over shoot and also the settling time. The settling time for motor mode 1 was 6.71% compared to motor model 3 which was 2.47%. Motor model 2 was the highest with 20% settling time. Below in figure 26 is a screenshot of two of the motor models responses.

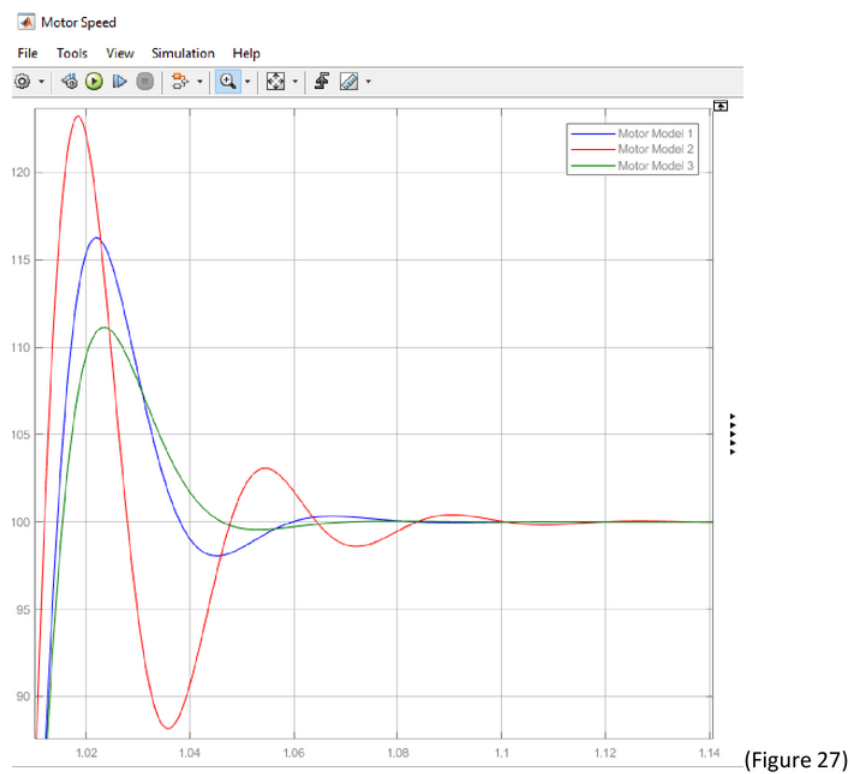
Kd value	Rise time	Settling time	Overshoot
0.0007	0.0103 seconds	0.0456 seconds	6.71%
0.007	0.0103 seconds	0.0012 seconds	20%
0.0001	0.0103 seconds	0.0243 seconds	2.47%



	Tuned	Block
P	0.2453	0.2453
I	19.2282	19.2282
D	0.00074365	0.001
N	17670.6828	2000

	Tuned	Block
Rise time	0.0103 seconds	0.0103 seconds
Settling time	0.0456 seconds	0.0243 seconds
Overshoot	6.71 %	2.47 %
Peak	1.07	1.02
Gain margin	Inf dB @ Inf rad/s	Inf dB @ Inf rad/s
Phase margin	64.5 deg @ 155 rad/s	72.9 deg @ 161 rad/s
Closed-loop stability	Stable	Stable

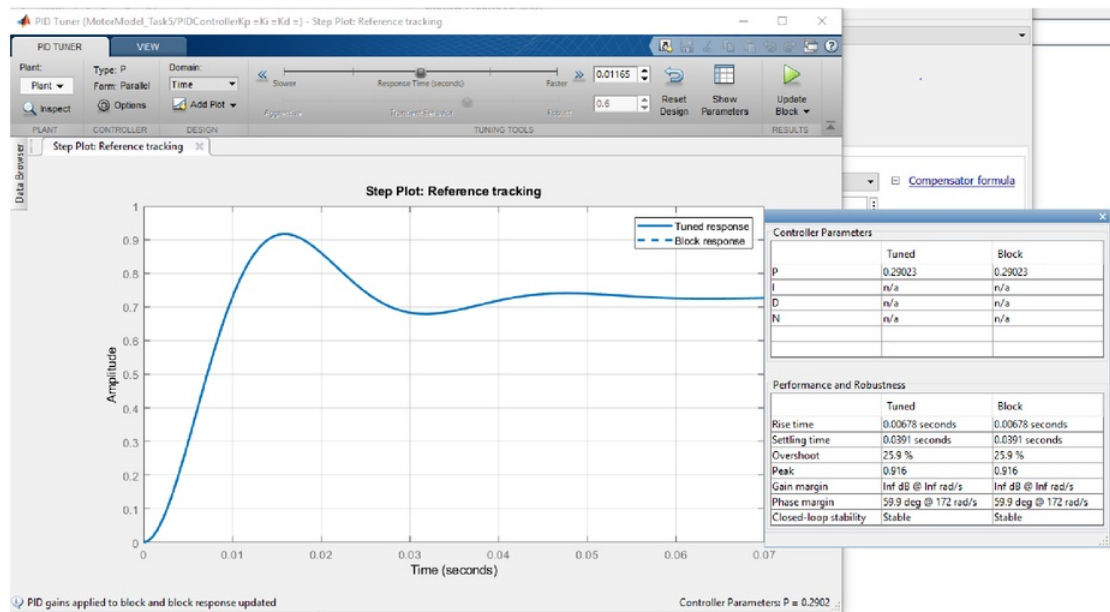
(Figure 26)



(Figure 27)

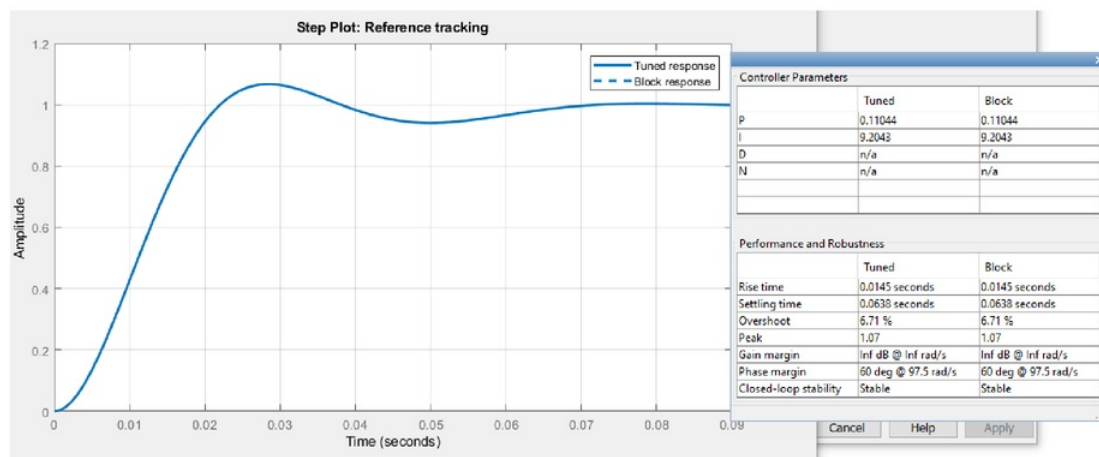
Types of motor speed controllers – P, PI and PID controller

In this task there are 3 different controllers the P, PI and PID controller. They are all tuned in order to give their best values and get their best response. Below in figure 28 is the tuning for the P controller.



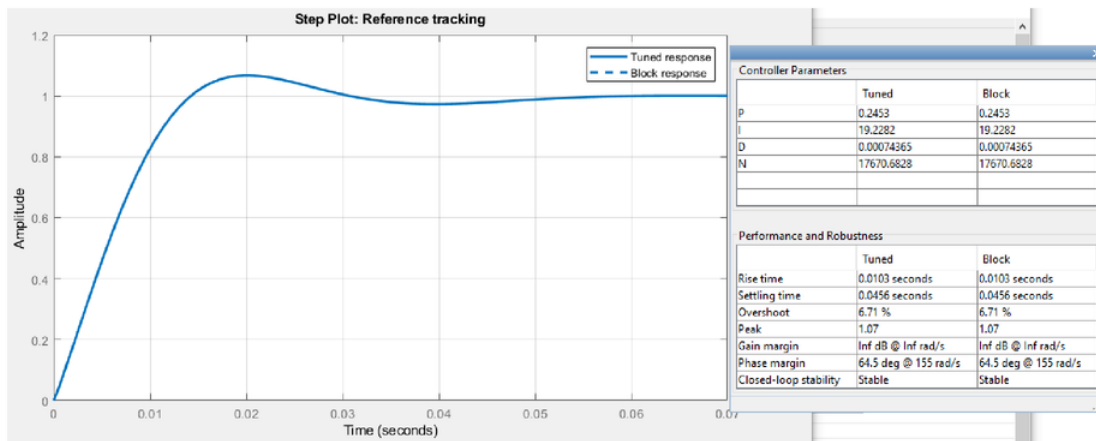
(Figure 28)

Following below in figure 29 is the tuned PI controller.



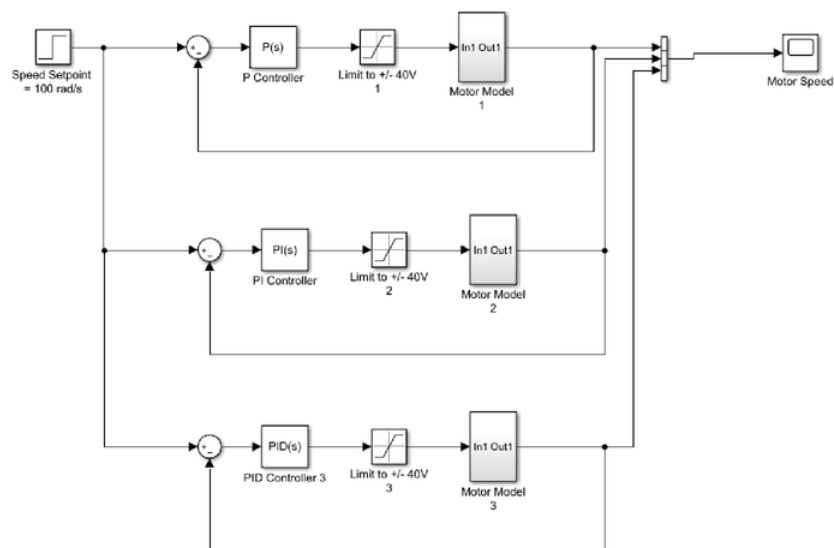
(Figure 29)

Below in figure 30 is the final tuned controller which is the tuned PID controller.



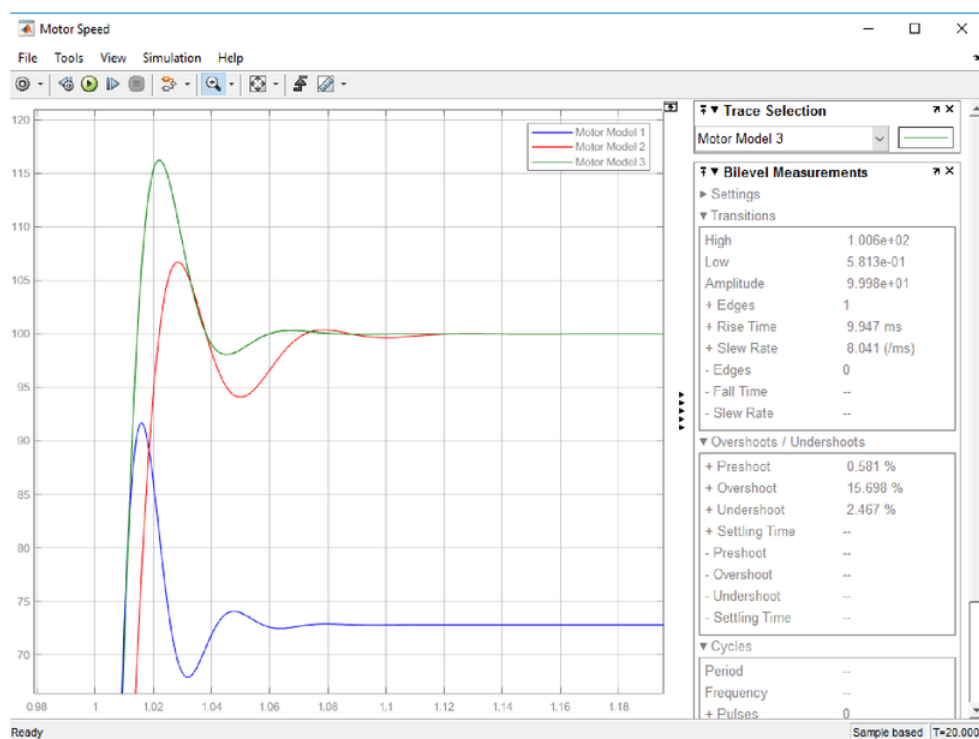
(Figure 30)

In figure 31 is the final motor model with all of the three different motor models in it. This motor model will now be simulated and each of the models will be individually evaluated giving reasons why some are better than others and any discrepancies explained.



(Figure 31)

Below in figure 32 is the 3 model response on the oscilloscope. Motor model 1 is the blue line which is the P controller. Motor model 2 is the red line which is the PI controller and finally motor model 3 is the green line and it is the PID controller. Firstly looking at the motor model one the response is very weak as it does not even get up to the desired running value it has a massive undershoot. Also with motor model 1 the settling time is very long as it oscillates much larger amounts compared to the other two controllers. This will be down to the fact that there is no K_i in the controller. Moving onto motor model 2 this is the PI controller. Already comparing it to motor model 1 it actually does reach the desired target but does have some overshoot. The settling time is a little bit better than motor model 1 however not as good as motor model 3. This is due to the fact that K_d reduces the settling time. Looking at motor model 3 this is the best motor model. It does have the largest overshoot however its settling time after the initial overshoot is better than both of the other settling times when looking at the graph. However sometimes this is not always the case. In some PID loops they only use PI due to the introduction to the differential part can cause the system to be less effective, however in this case the PID controller is the best controller.



(Figure 32)

Conclusions and recommendations

To conclude from this assignment a massive gain in understanding about first order systems has been gained. The method about how to solve mathematical problems to do with laplace equations has been understood and gained. Also a greater understanding has been gained about how to replicate a DC motor on matlab and find out real life predictions without having to do the practical and physical side of the work. The DC motor section was very good for self-development as with the use of oscilloscopes you can see the various characteristics of the motor such as the no load speed and torque. Also from the mathematical calculations things like rotor inertia and friction coefficient are known which can help in real life situations. A full understanding of the various different types of PID controllers has been collected due to the investigation into the 3 different aspects of PID control and what affects they have on a motor model. From this assignment the conclusion can be come to that a PID controller is the best form of controller for this particular assignment however it can be said that this is not true in every case as sometimes the differential can cause errors in the PID loop control and affects the performance of the controller.

Recommendations for this assignment would to possibly compare more data against each other for the PID controllers so from all of that data a more solid conclusion can be made. The more data that can be collected would also result in more comparisons being made which can also improve the accuracy of the data collected which will eliminate any discrepancies in results.

FINAL GRADE

GENERAL COMMENTS

Instructor

56/100

PAGE 1

PAGE 2

PAGE 3

PAGE 4

Text Comment. equations are difficult to read, should have used the equation editor

Text Comment. you could have drawn the argand diagram here

Text Comment. explanation needed here

PAGE 5

Text Comment. explanation needed here, new block diagram should be drawn

Text Comment. unnecessary to do this, you should have just placed a cursor on the above plot at zero

PAGE 6

PAGE 7

Text Comment. you could have calculated the overall gain and time constant for comparison

PAGE 8

PAGE 9

Text Comment. explanation needed for where the equation comes from

Text Comment. equation is difficult to read

PAGE 10

Text Comment. poor image, spoils the presentation of the report

PAGE 11

PAGE 12

PAGE 13

Text Comment. How come?

PAGE 14

PAGE 15

Text Comment. The steady state error should be your main point here

PAGE 16

PAGE 17

Text Comment. what about the steady state error?

PAGE 18

PAGE 19

Text Comment. Summary table for overshoot and rise time is needed here for different values of K_i for comparison

PAGE 20

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

Text Comment. do you mean steady state error?

PAGE 26

GRADING FORM: CSG-RUBIC

RESEARCH-INFORMED LITERATURE

 n/a

KNOWLEDGE AND UNDERSTANDING

 Section 1: Explanation of first order system – Briefly given but no argand diagram plotted and rlocus not needed the second time. (5/10) Section 2: Explanation of DC motor model – Some given, equations are difficult to follow, should have used equation editor. Good predicted values calculated. (6/10) Section 3: Explanation of types of speed controllers – Generally, very good. (6/10)

17

ANALYSIS

 Section 1: Testing of first order system – Good. However, you could have measured overall gain and time constants for each case. (5/10) Section 2: Testing of DC motor model – Very good evidence given, test results match theory. (6/10) Section 3: Testing of speed controllers – Good graphical evidence presented but not all results are summarized in a table for comparison. (10/20) Section 4: Conclusions and recommendations – Some conclusions offered but recommendation is a bit vague. (5/10)

26

PRACTICAL APPLICATION AND DEP.



Sections 1, 2 and 3: Use of Simulink and presentation quality of results - Good use of the software, good presentation quality. (7/10)

7

SKILLS FOR PROFESSIONAL PRAC.



All sections – Well structured but presentation could be improved by use of equation editor. Figures should be listed. (6/10)

6