

assignment 2 5009

by Sean Jones

FILE	10202_SEAN_JONES_ASSIGNMENT_2_5009_158265_329781126.DOCX		
TIME SUBMITTED	15-MAY-2018 03:55PM (UTC+0100)	WORD COUNT	2138
SUBMISSION ID	89272981	CHARACTER COUNT	14525

Assignment2

Sean Jones

Analysis, Simulation and
Evaluation of Electrical Circuits
and Electronic Systems

HNC electrical and electronic
engineering

PSDE5009

Submission date: 15/05/2018

Contents

1. First Order Control System.....	3
2. DC Motor Model Analysis	10
3. Types of Motor Speed Controller.....	15
4. Conclusion.....	26

Figure 1 First order circuit.....	3
Figure 2 model of first order circuit	4
Figure 3 first order no gain simulation.....	5
Figure 4 first order argand diagram	7
Figure 5 0.118 gain first order step response	8
Figure 6 unstable gain circuit.....	8
Figure 7 0.115 gain used on first order step response.....	9
Figure 8 motor model	11
Figure 9 no load step response.....	12
Figure 10 no load torque	12
Figure 11 load speed.....	13
Figure 12 stall torque	14
Figure 13 p controller circuit.....	15
Figure 14 tuned Kp value	16
Figure 16 low Kp value	17
Figure 15 high Kp value	17
Figure 17 p controller comparison.....	18
Figure 18 pi controller tuned	19
Figure 20 pi low Ki value	20
Figure 19 high Ki value	20
Figure 21 pi comparison circuit.....	21
Figure 22 Pi response comparison.....	21
Figure 23 0.000001 KD response	22
Figure 24 KD response	23
Figure 25 0.01 KD response	23
Figure 26 PID comparison graph.....	24
Figure 27 PID comparison	24
Figure 28 comparing p, pi & PID	25
Figure 29 p pi & pid comparison simulation	25

this gap should be removed

1. First Order Control System

Below in Figure 1 shows the first order circuit. This circuit currently has feedback with no gain. It will

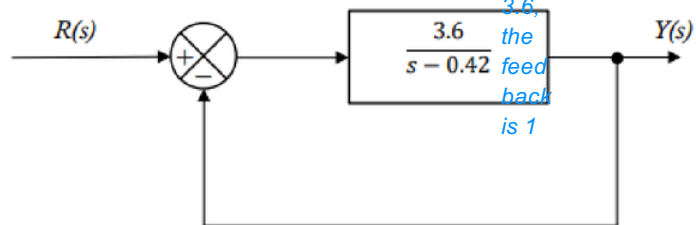


Figure 1 First order circuit

need to be calculated and tested to see how stable this system would be without any gain in the feedback

this is the open loop transfer function

$$Gs = \frac{3.6}{s - 0.42}$$
$$Hs = 1$$
$$s - 0.42 = 0$$
$$s = 0.42$$

The calculation suggests that this system would be unstable as the value would be far over on the positive side of the x axis on an argand diagram. This system but wouldn't have a sharp response that is desired. The gain on a feedback multiplies the current feedback by the gain, theoretically the gain is 1 when no gain is added. A model has been created on Simulink to simulate how this system would react on a step response. Below in Figure 2 is the model that's been created.

This circuit has a step response added to the input and a virtual scope added to the output to give a virtual waveform output of the behaviour of the output of this model

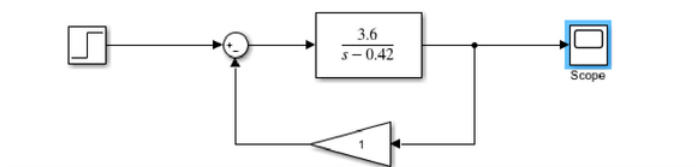


Figure 2 model of first order circuit

Below in Figure 3 is the step response from the no feedback model. The slow response shows the over damping issue that was expected during calculation.

Damping is a term relating to second order systems whereas this is a first order system and the term does not apply. You should be explaining about time constants and overall gain

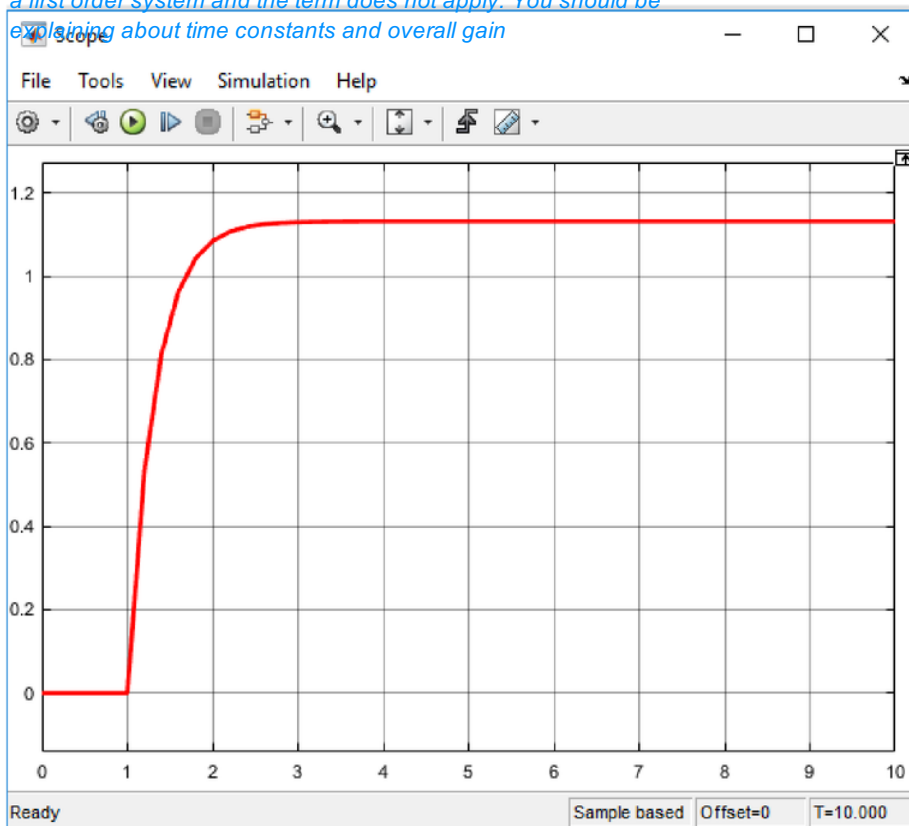


Figure 3 first order no gain simulation

The correct gain is going to be calculated to give the circuit critical damping and a minimum overshoot. *incorrect*

$$T(s) = \frac{3.6/(s - 0.42)}{1 + 3.6/(s - 0.42)}$$

Multiply top and bottom by (s-0.42)

$$T(s) = \frac{3.6/(s - 0.42) * (s - 0.42)}{1 + 3.6/(s - 0.42) * (s - 0.42)}$$

$$T(s) = \frac{3.6}{s - 0.42 + 3.6}$$

$$T(s) = \frac{3.6}{s + 3.18}$$

Root of the denominator

$$s + 3.18 = 0$$

$$s = -3.18$$

$$H(s) = K$$

$$t1(s) = \frac{\frac{3.6}{(s-0.42)}}{1 + \frac{3.6K}{(s-0.42)}}$$

$$t1(s) = \frac{3.6}{s - 0.42 + 3.6k}$$

$$t1(s) = \frac{3.6}{s + (3.6k - 0.42)}$$

Pole is found when

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

For pole at s=0

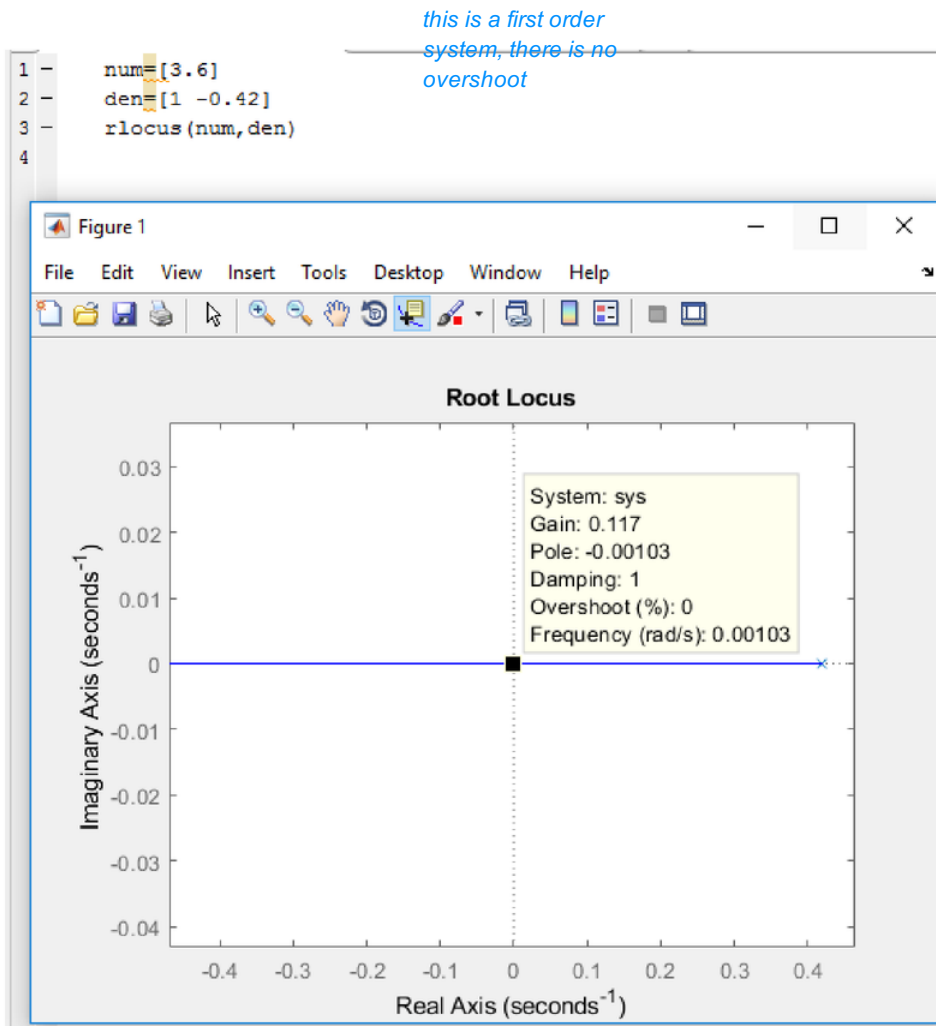
$$0 + 3.6k - 0.42 = 0$$

$$3.6k = 0.42$$

$$K = 0.117$$

This is Kmin

When the calculated gain is plotted on an argand diagram it shows it should behave very stable as it is in the negative on the x axis and would have no overshoot. Figure 4.



no applicable The critical damping gain value is placed onto the Simulink model that was created previously. Figure 5. The value of 0.117 wasn't used as this value is the dead minimum value that can be used before the system goes unstable. This shows a sharper response which takes almost half the time to reach the maximum value. This model does have a large steady state error which isn't ideal but is better than having overshoots and ringing.

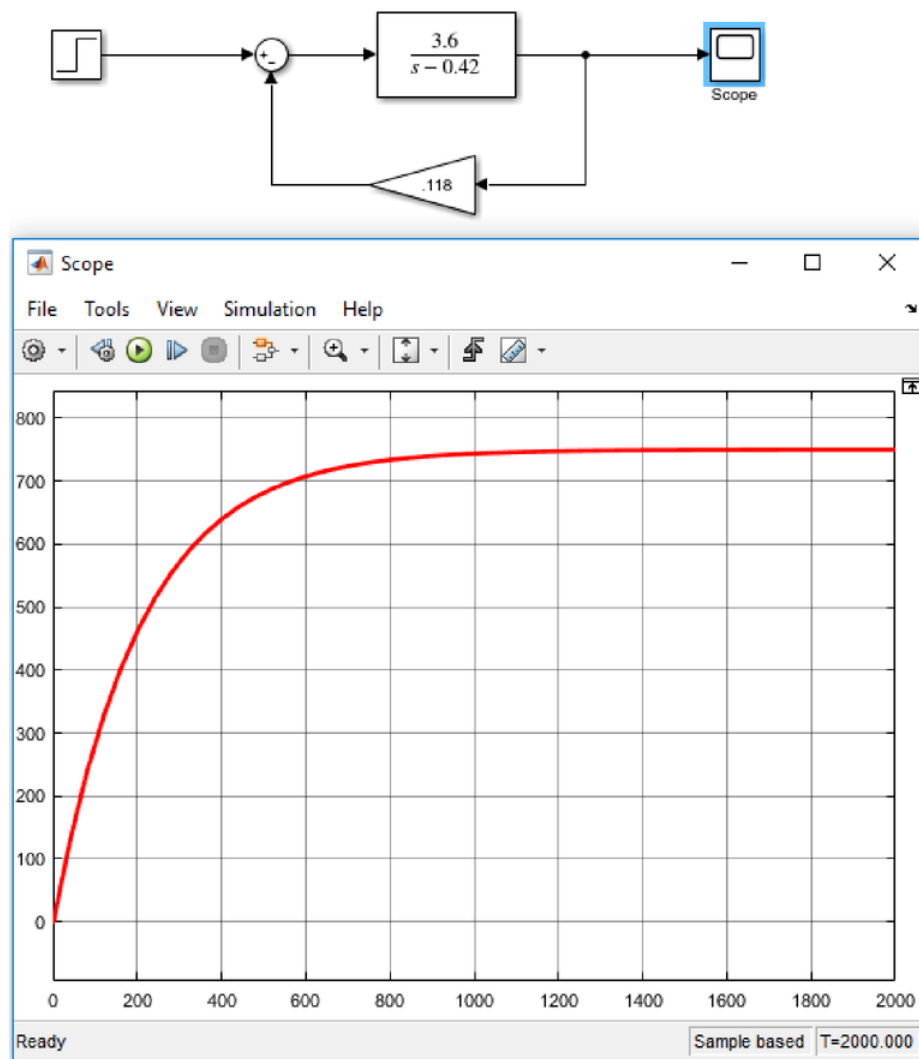


Figure 5 0.118 gain first order step response

To prove the system would go unstable, the gain 0.115 was used which should make the model unstable according to the argand diagram. Figure 6

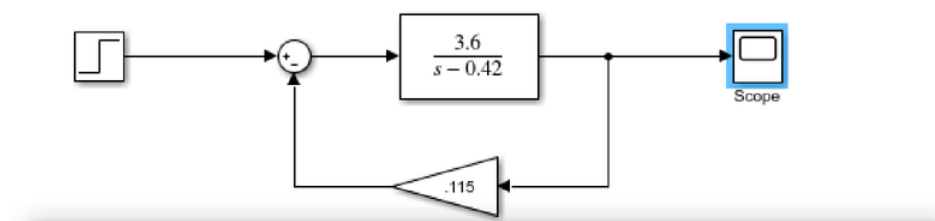


Figure 6 unstable gain circuit

Figure 7 shows the circuit going unstable with a gain of 0.115. this proves the calculations are correct and that the perfect gain value can be calculated to give the best step response on a model

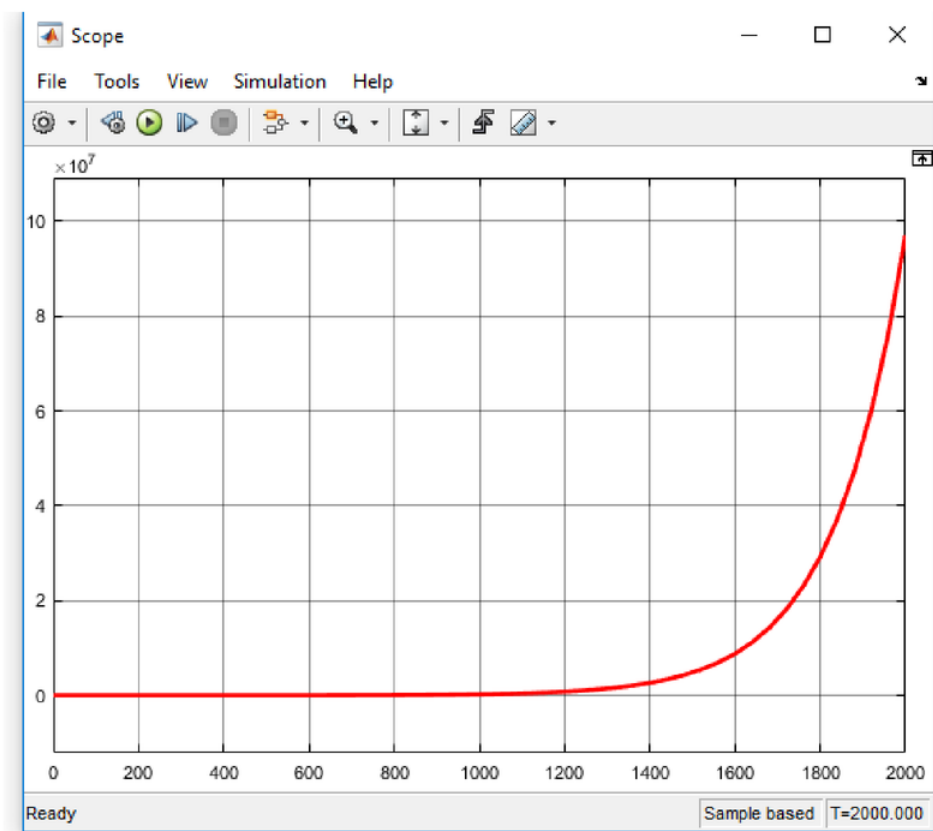


Figure 7 0.115 gain used on first order step response

2. DC Motor Model Analysis

A motor need to be modelled on Simulink to perform simulation on to see how to make it stable have a critical step response and to see how it behaves under load. The transfer function will firstly need to be derived from the motor data which is below.

- 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

you should have started with a block diagram first.

The equation for the transfer function is below.

$$G(s) = \frac{Kt/LaJ}{s^2 + s\left(\frac{b}{J} + \frac{Ra}{La}\right) + \frac{Rab + KtKe}{LaJ}}$$

Below the values of each part or shown.

$$Ra = 7.8 \text{ Ohms}$$

$$La = 0.05\text{H}$$

$$J = 0.000214$$

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

$$Kt = 0.09$$

$$Ke = 0.0983$$

This equation is then calculated in parts then the put back into the equation to simplify it to reduce the chance of a miss calculation.

$$\frac{Kt}{LaJ} = \frac{0.09}{0.05 \times 0.000214} = 8411.215$$

$$\frac{b}{J} + \frac{Ra}{La} = \left(\frac{15.519 \times 10^{-6}}{0.000214}\right) + \left(\frac{7.8}{0.05}\right) = 0.0725 + 156 = 156.0725$$

$$\frac{Rab + KtKe}{LaJ} = \frac{(7.8 \times 15.519 \times 10^{-6} + 0.09 \times 0.0983)}{0.05 \times 0.000214} = 838.135$$

$$G(s) = \frac{8411.215}{s^2 + 156.0725s + 838.135}$$

$$cf = \frac{If}{s^2 + 2\zeta\omega_n + \omega_n^2}$$

$$\omega_n^2 = 838.135$$

$$\omega_n = \sqrt{838.135} = 28.95 \text{ rad/s}$$

$$2\zeta\omega_n = 156.0725$$

$$\zeta = \frac{156.0725}{(2 \times 28.95)} = 2.7$$

Overall gain

$$K = \frac{8411.25}{838.135} \approx 10$$

The circuit below [Figure 8] shows the final circuit presented in Matlab this is where the tests are made to simulate the circuit to prove the calculations. The motor speed, load speed and stall speed are calculated and compared against the simulated values

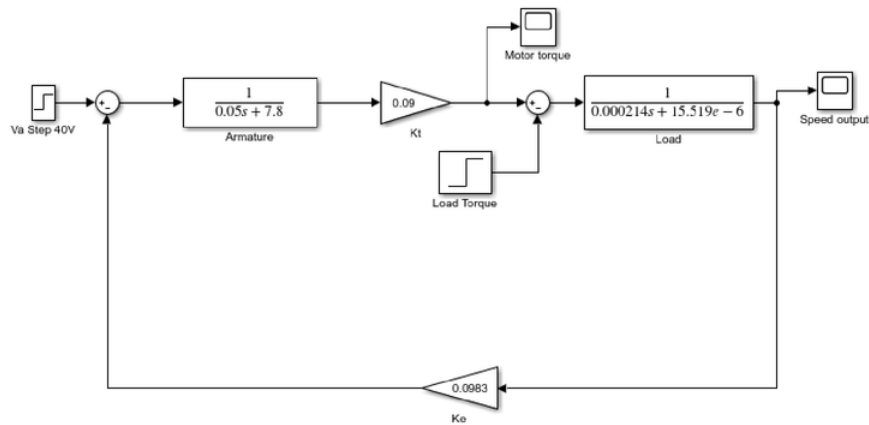


Figure 8 motor model

The first value to be calculated is the speed in which the motor will run under no load. This is calculated using the equation below.

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

$$\Omega_{NL}(b) = \frac{0.09 \cdot 40}{(15.519 \cdot 10^{-6}) \cdot 7.8 + 0.09 \cdot 0.0983}$$

$$\Omega_{NL}(b) = 240.86(\text{rad/s})$$

Below shows the speed of the motor under no load. The probe results on the side show a value of

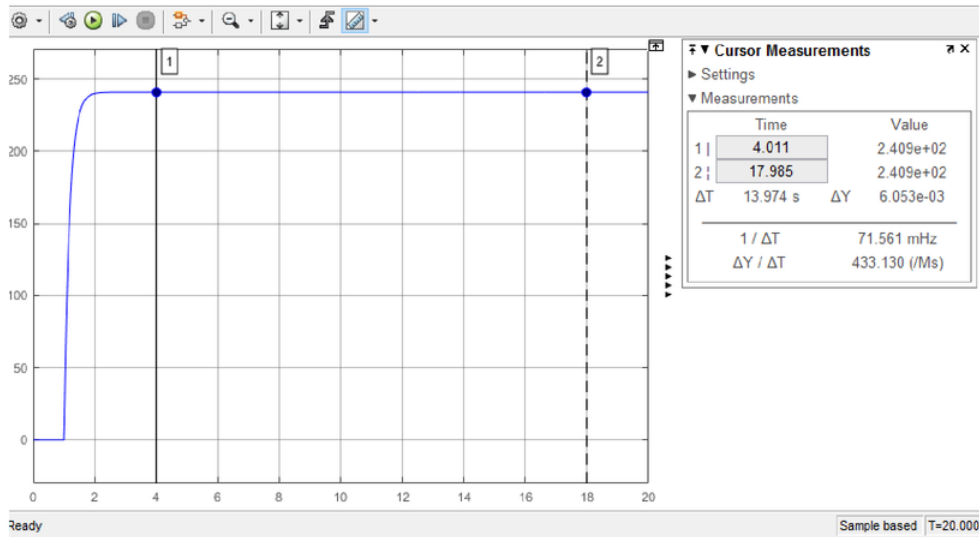


Figure 9 no load step response

240.9 rad/s which is very close to the calculations . this is more than likely due to a rounding issue with the Simulink software. The torque of the motor is shown below in Figure 10

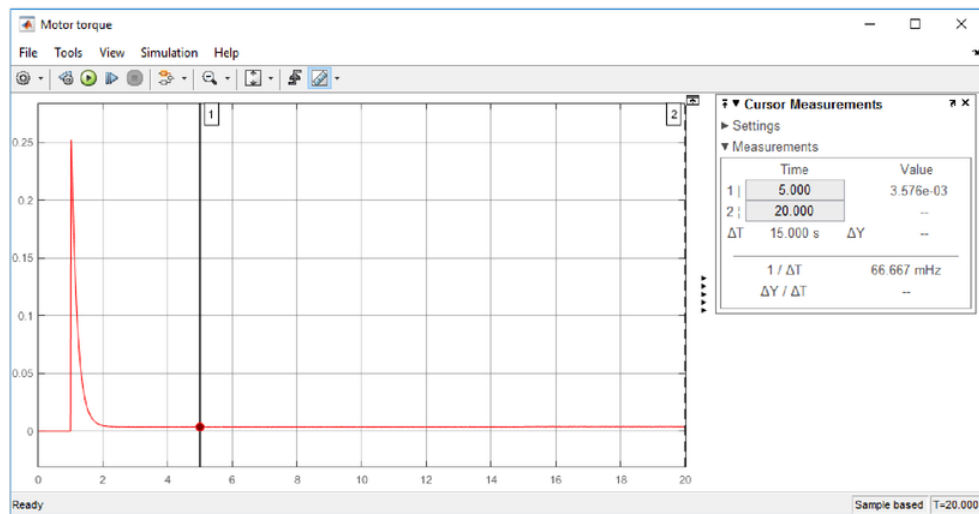


Figure 10 no load torque

The next part of the motor to be calculated is the loaded speed. Below shows how this is calculated.

$$\Omega L = \frac{Va - \frac{RaTl}{Kt}}{\frac{Rab}{Kt} + Ke}$$
$$\Omega L = \frac{40 - \frac{7.8 \times 0.025}{0.09}}{\frac{7.8(15.519 \times 10^{-6})}{0.09} + Ke}$$

Below shows the speed of the simulated model with load added after 10 seconds. The speed of the motor is 219.7rad/s which is what was calculated. when the load is added the motor slows down by 21.2 rad/s.

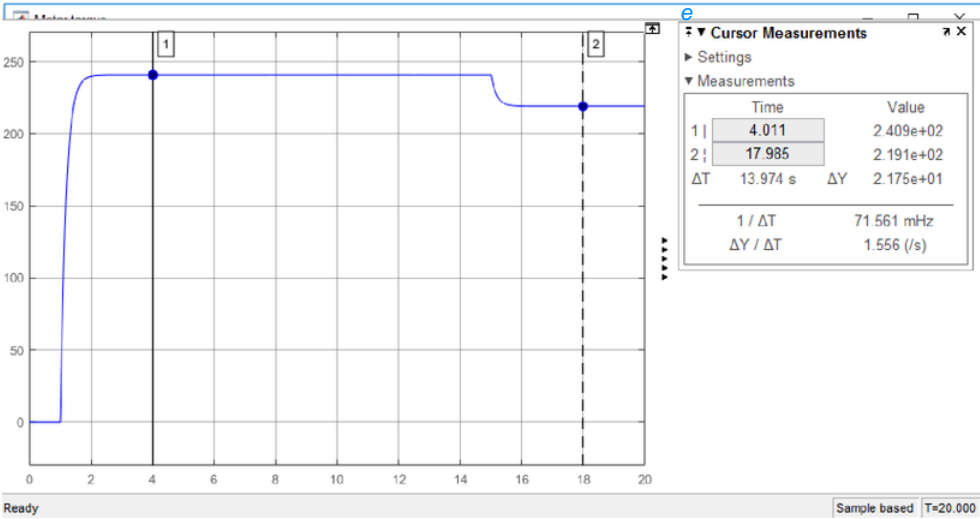


Figure 11 load speed

The stall torque is now calculated. This is the torque that is required to stall the motor. This is firstly calculated.

$$Tm(stall) = \frac{Kt Va}{Ra}$$

$$Tm(stall) = \frac{0.09 * 40}{7.8}$$

$$Tm(stall) = 0.276923$$

the calculated torque is then inputted into the torque of the motor model and shown below on the speed response of the motor. Figure 12 the motor runs at full speed for 10s and when the torque is added the motor stops, proving the calculations are correct.

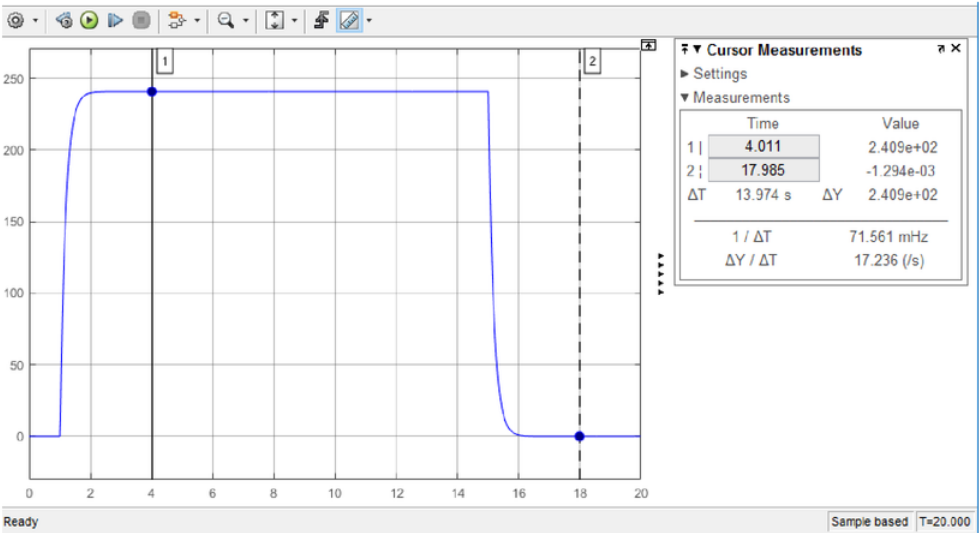


Figure 12 stall torque

	Calculation	Simulation
No load speed	240.86rad/s	240.9rad/s
Load speed		219.7 rad/s
Stall load torque	0.276923	Stalled at this load

The calculations matched the simulated values very close suggesting slight rounding errors somewhere affecting the final values. This proves the calculations are correct. The simulation software performs the calculations that were done also so the final number are expected to be reasonable close if correct.

3. Types of Motor Speed Controller

The motor model previously made is reduced down into a sub circuit to make it easy to be used multiple times. A 100rad/s step input is put into a summing block which subtracts the motor output from it. This then goes into a proportional controller. The same disturbance of 0.025Nm will still be used. The proportional control block will need to sense the drop in speed from the motor after the extra torque is added and act appropriately to get the motor back to the running speed. The proportional control will also control the acceleration of the motor getting to the full running speed. There is a limiter of 40v passed before the motor model to prevent the motor rotating too fast. *exceed its rating*

The following tests show the effect of changing K_p on the proportional control from the tuned suggested value, to value much higher than the suggested value and to a value much lower. These 3 different values should give clear indication on the effect of K_p .

Below in Figure 3 shows the simulink model.

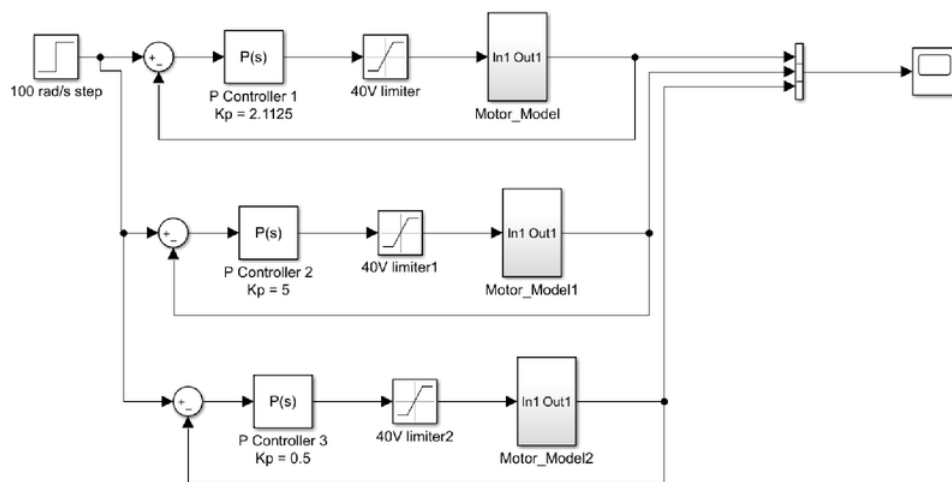


Figure 13 p controller circuit

the first value was given by the tune function on simulink and then input that value as K_p . In the table below shows the characteristics when changing the value of K_p . It shows the steady state error is increased as K_p is decreased but the time to get to the steady state is increased. Also with a high K_p a high overshoot is given. When tuning the K_p a compromise has to have a equal balanced response where little steady state error is wanted along with a quick response time and little overshoot.

K_p	E_{ss} (rad/s)	T_r (seconds)	$O\%$
2.1125	4.5	0.0131	11.2
5	2	0.0069	27.8
0.5	16.7	0.0503	0

Below shows the response on the auto tune feature.

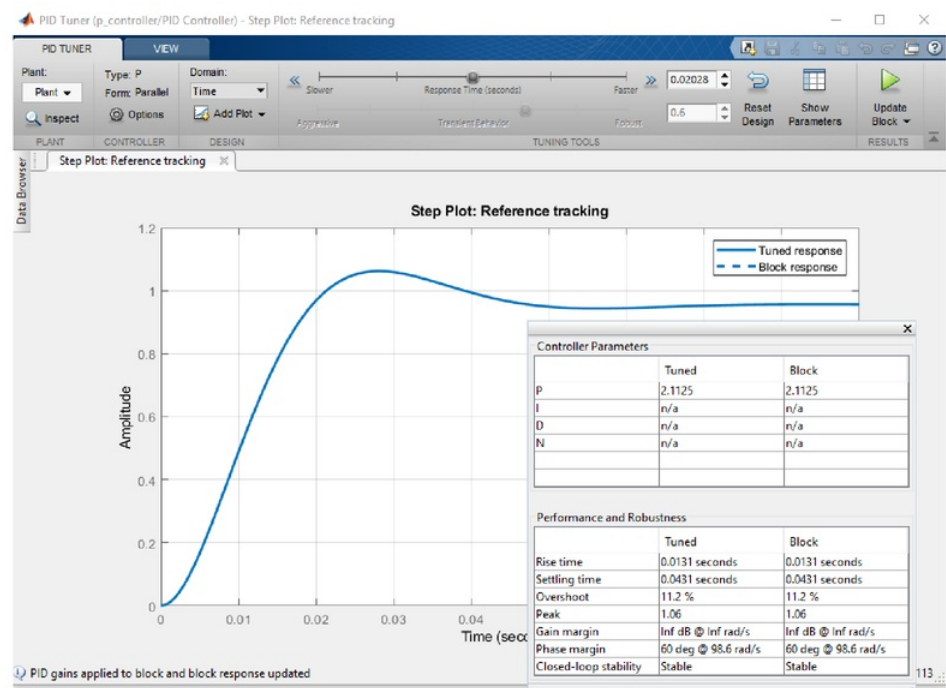


Figure 14 tuned Kp value

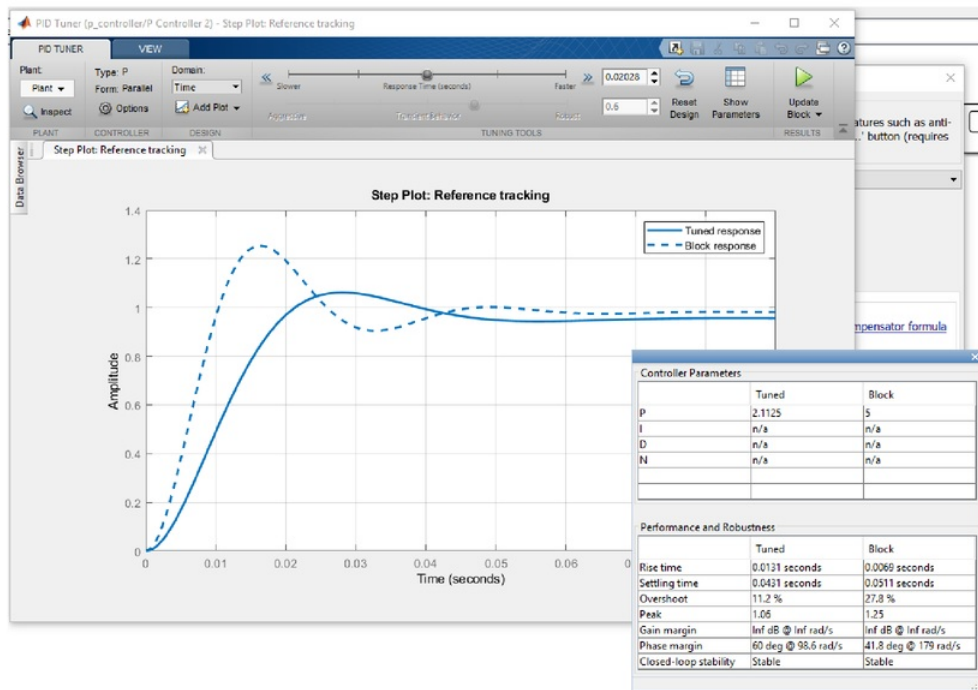


Figure 15 high Kp value

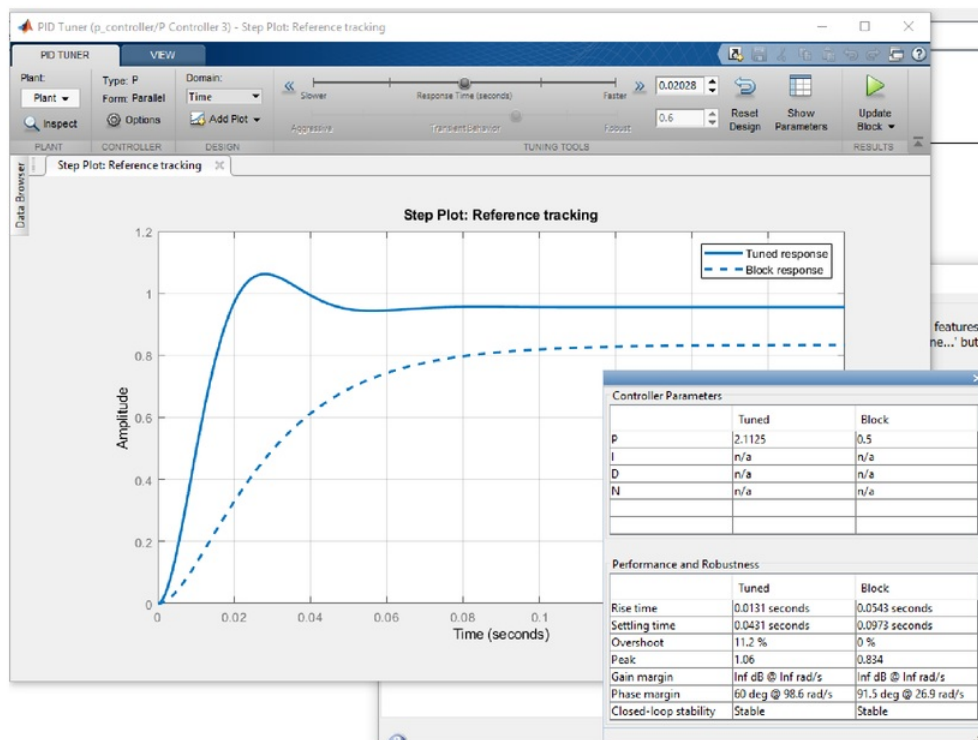


Figure 16 low Kp value

below in Figure 17 shows the 3 systems attached to the same input scope showing the speed of the motor with different Kp values. There is a clear difference in the response of the motor with overshoot, response time and steady state error. To summarise the proportional controller, increasing Kp gives a better steady state error a better response time but a large overshoot.

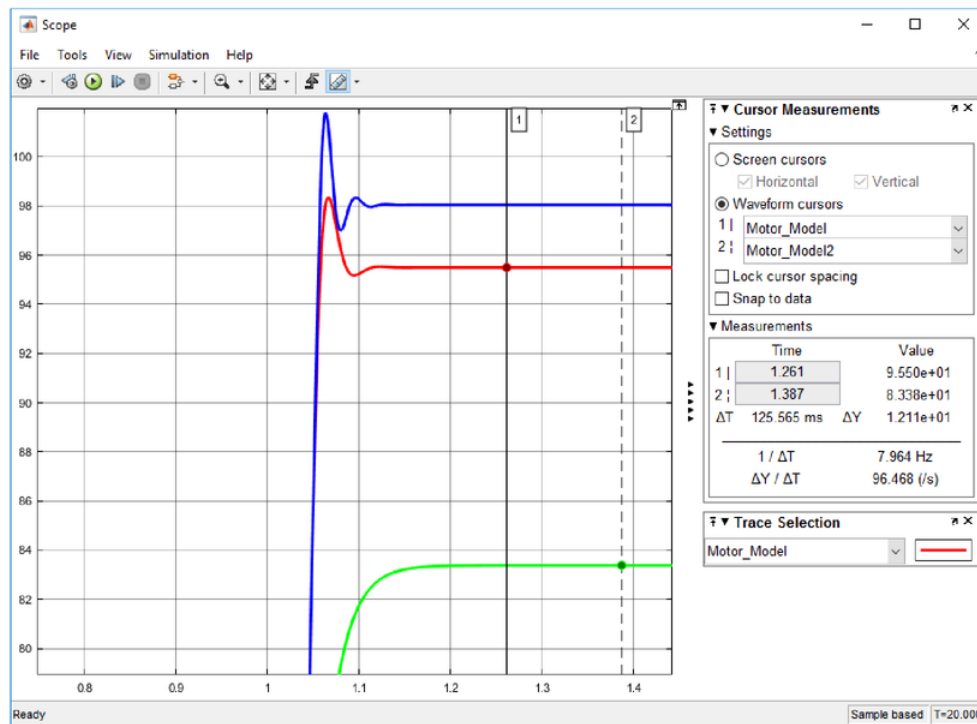


Figure 17 p controller comparison

you could have summarized you results in a table for comparison

Pi Controller

The same simulation is used but with PI controllers used. The K_p value used will be the tuned value from the P controller and the K_i value will be varied to see the effect of changing the K_i .

The ideal K_i value will be found first of all using the auto tuning feature on simulink. Figure 18 shows the auto tuning of the pi controller.

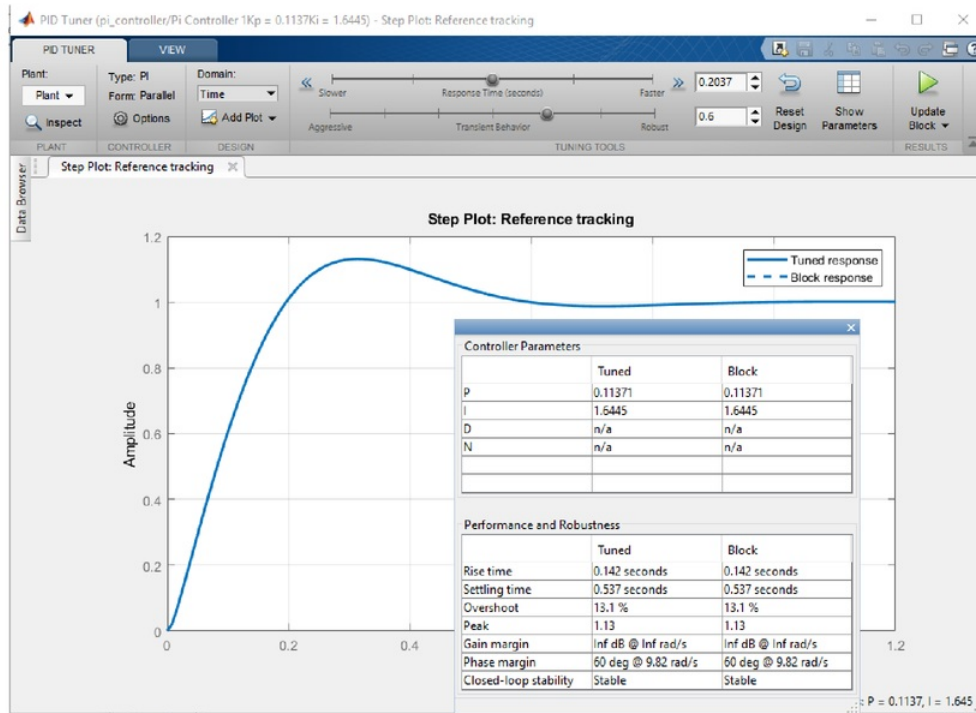


Figure 18 pi controller tuned

a higher and lower value of K_i was then inputted into the pi controllers and tested on the auto tune function to obtain the characteristics in Figure 19 and Figure 20.

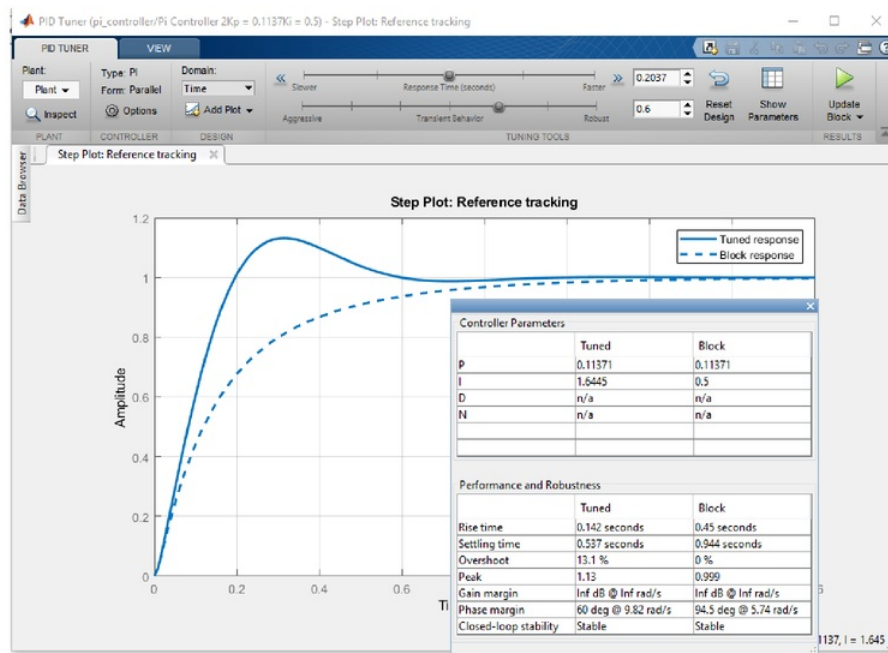


Figure 20 pi low Ki value

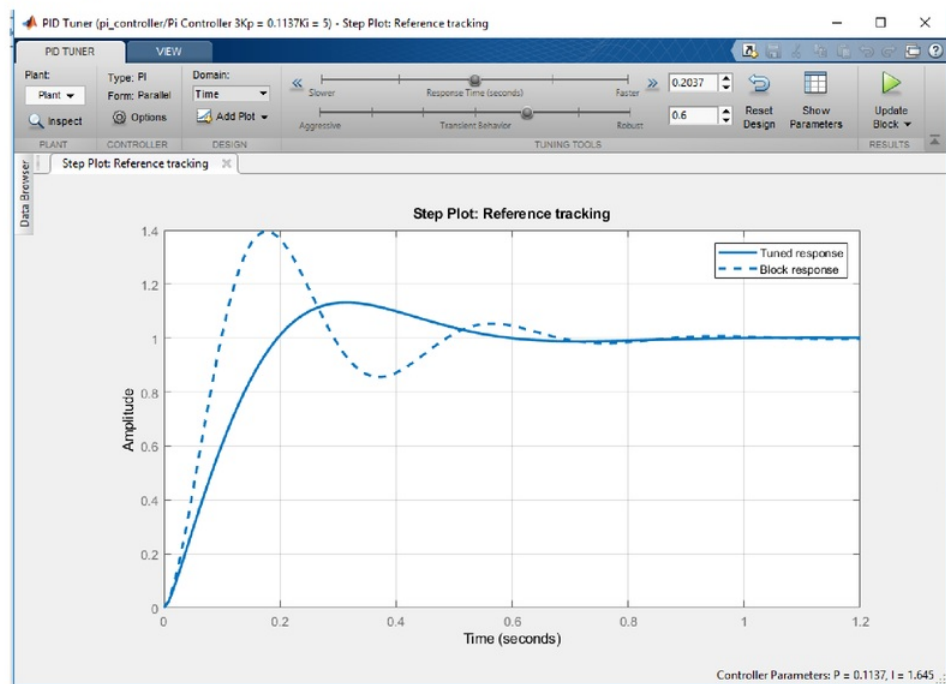


Figure 19 high Ki value

results could have been tabulated for comparison

all three controllers are placed onto the same model to compare them Figure 21.

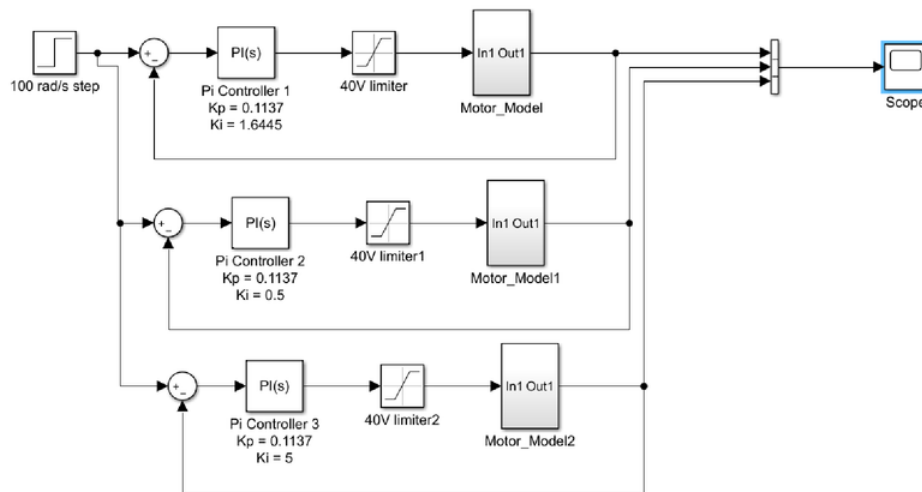


Figure 21 pi comparison circuit

When this circuit is tested on the scope there is a obvious difference in the PI controller. The integral takes away the steady state error but adds overshoot and a ringing affect on the wave form after the overshoot. The integral can cause oscillations if an unstable circuit is used. Increasing the K_i

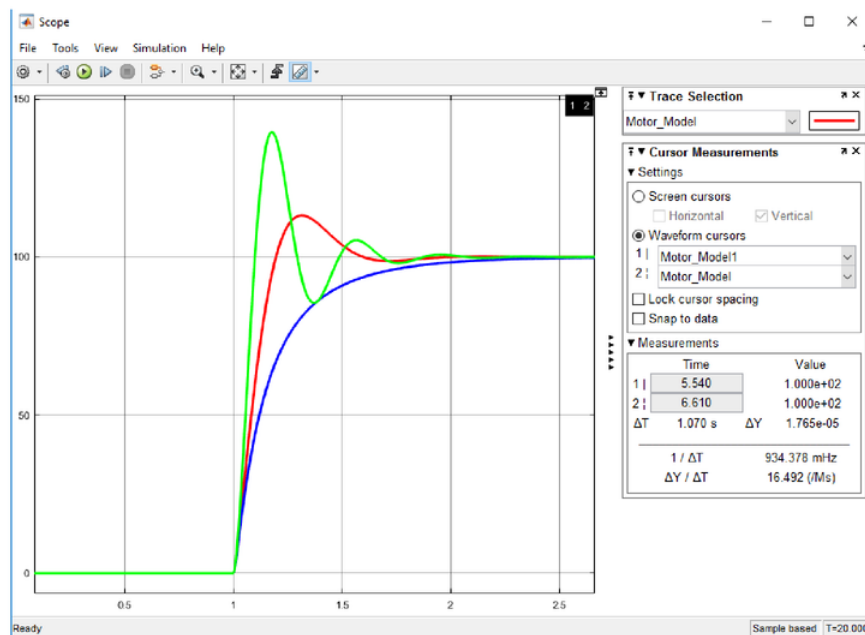


Figure 22 Pi response comparison

PID controller

The circuit will be used again but accommodating PID controllers to control the speed of the circuit. The K_P and K_i will remain a constant whilst the K_D is varied to see the effect of the differential.

$K_P = 0.1$

$K_i = 1$

The first value that will be used will be 0.000001. The auto tune feature suggests changing the differential to 0 as this circuit would work better without a differential but to show the effect of it will have a number very close to 0 hence why 0.000001 was used. The response for this shown in Figure 23 shows a response very similar to the tuned pi circuit. Figure 25 and Figure 24 shows the effect of increasing K_D .

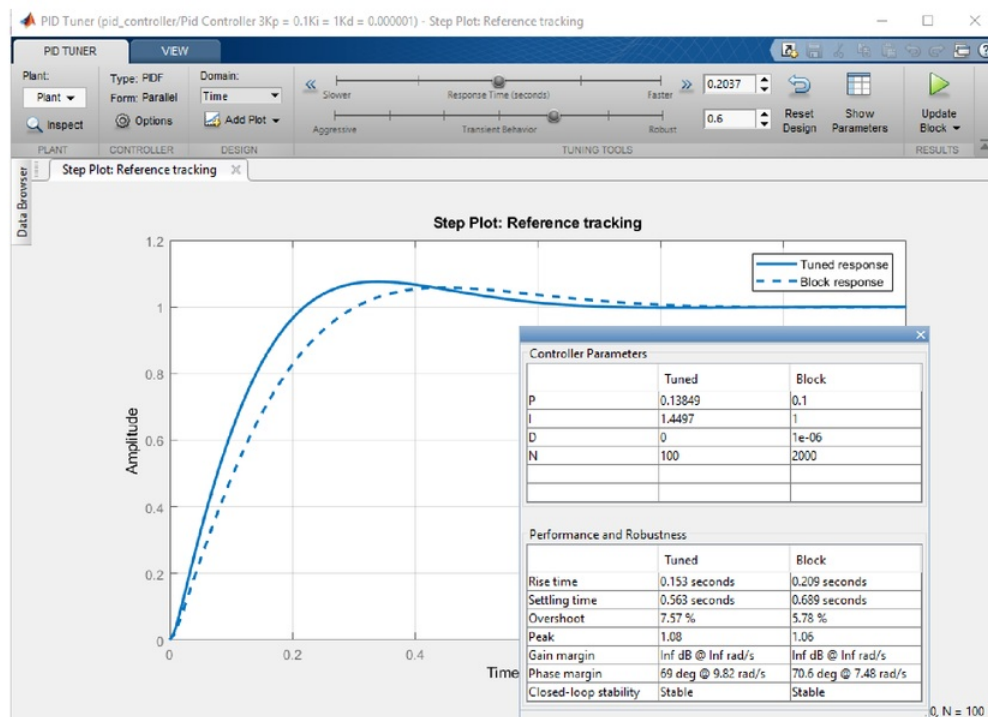


Figure 23 0.000001 K_D response

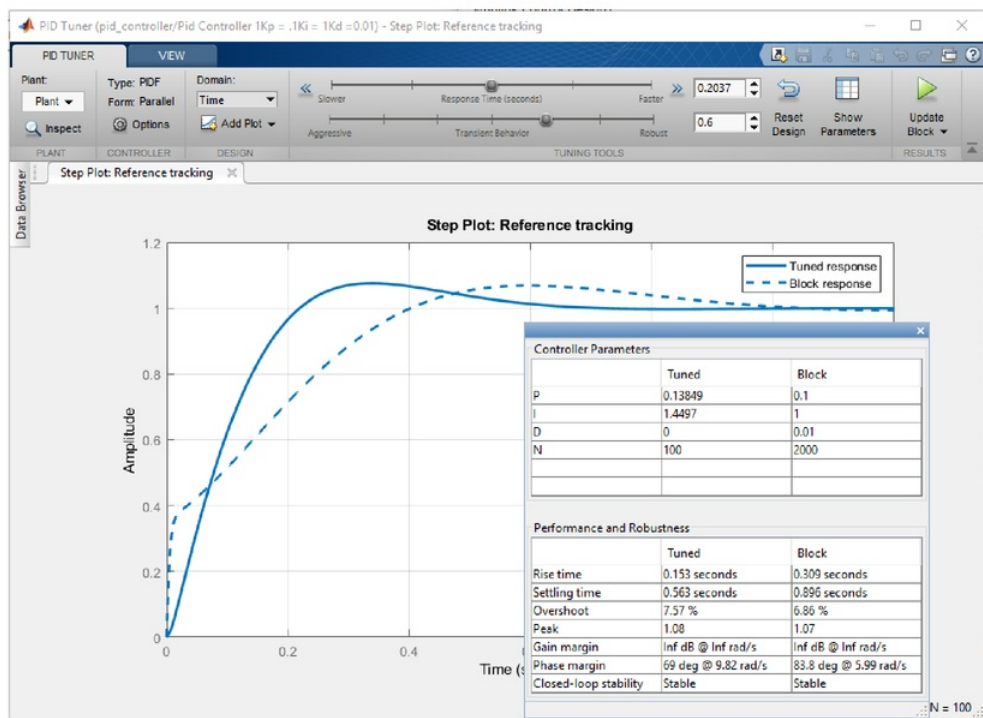


Figure 25 0.01 KD response

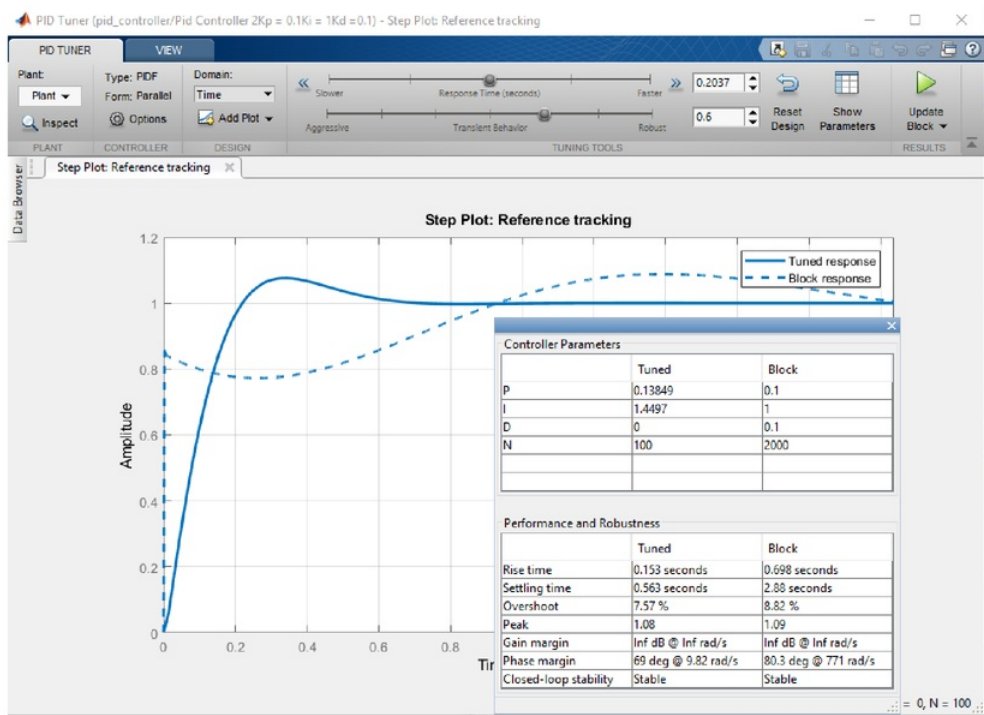


Figure 24 KD response

To easily visually compare the 3 values and the effect of changing KD the 3 controllers are all placed on Simulink sinking into the same scope so that they all will be shown on the same graph. Figure 27

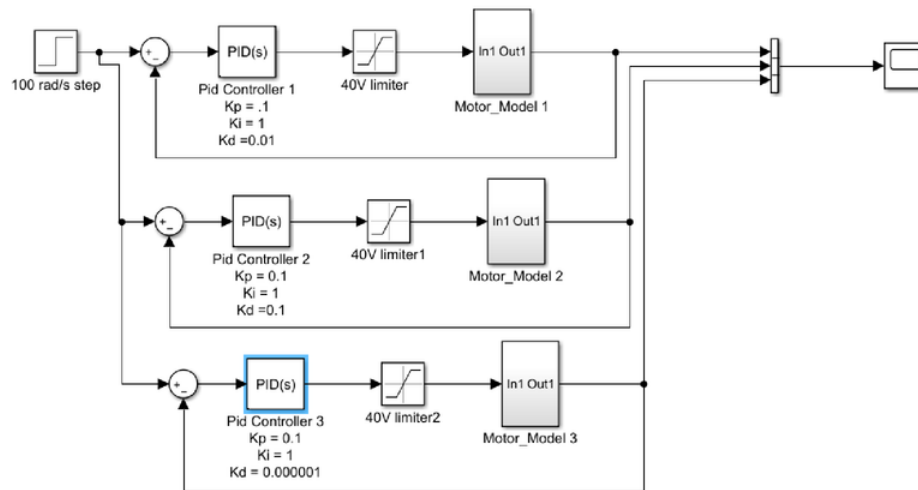


Figure 27 PID comparison

The graph in Figure 26 shows that increasing KD doesn't improve the speed response but doesn't make the circuit unstable.

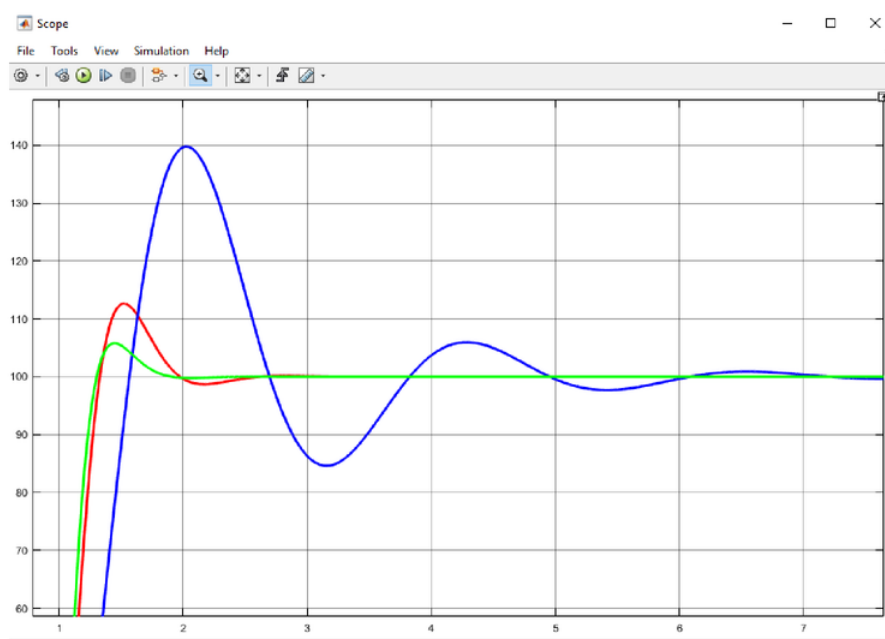


Figure 26 PID comparison graph

Adjusting P I and D in a circuit adjusts the characteristics in different ways so to compare and contrast the 3 the best tuned model of each controller is put into 1 circuit and simulated to see the difference. The circuit is shown below in Figure 28.

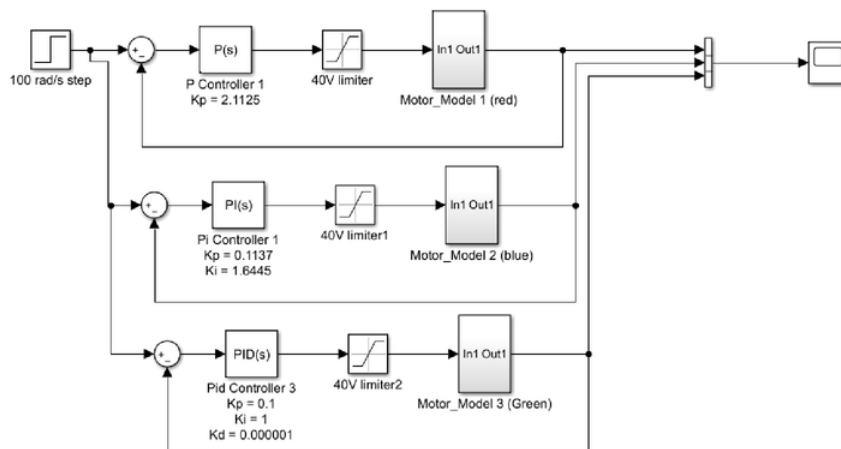


Figure 28 comparing p, pi & PID

The circuit is then simulate and the result is shown below. It shows that for response time the best controller is p but the down side of this is a large steady state error so its suggested that PID is more suited for the motor speed control as it holds the correct speed but with a les sharp reaction and a greater overshoot. If this was a positioning instead of speed then the overshoot would be critical. Figure 29

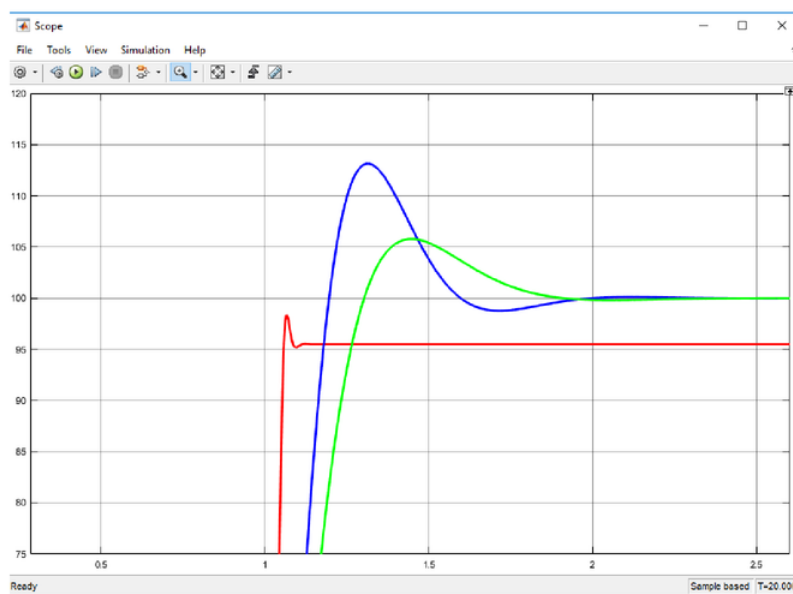


Figure 29 p pi & pid comparison simulation

4. Conclusion

The report highlights the affect of using different speed controllers and also the importance of feed back to make an unstable system stable. The structure of the report helped show what each type of speed controller does and how to tune it as well as effect tuning the controller does to the response of the circuit.

The calculations were all proved to be correct using the simulating feature on simulink. It was established that the best speed controller to be used was the PID controller as it had little overshoot, a reasonable time to get to the steady state and no steady state error.

assignment 2 5009

GRADEMARK REPORT

FINAL GRADE

GENERAL COMMENTS

Instructor

57 / 100

PAGE 1

PAGE 2

PAGE 3

Text Comment. this gap should be removed

Text Comment. the open loop gain is 3.6, the feedback is 1

PAGE 4

Text Comment. without any feedback at all!

Text Comment. this is the open loop transfer function

Text Comment. you could have drawn the argand diagram here

QM

Spelling

Spelling error

PAGE 5

Text Comment. Damping is a term relating to second order systems whereas this is a first order system and the term does not apply. You should be explaining about time constants and overall gain

Text Comment. incorrect

Text Comment. feedback

PAGE 6

PAGE 7

Text Comment. this is a first order system, there is no overshoot

Text Comment. no applicable

PAGE 8

PAGE 9

PAGE 10

Text Comment. you should have started with a block diagram first.

PAGE 11

PAGE 12

PAGE 13

Text Comment. incomplete

PAGE 14

PAGE 15

Text Comment. exceed its rating

PAGE 16

PAGE 17

PAGE 18

Text Comment. you could have summarized you results in a table for comparison

PAGE 19

PAGE 20

Text Comment. results could have been tabulated for comparison

PAGE 21

PAGE 22

PAGE 23

PAGE 24

PAGE 25

PAGE 26

SEAN JONES**57****RESEARCH-INFORMED LITERATURE**

n/a

KNOWLEDGE AND UNDERSTANDING

Section 1: Explanation of first order system – Some given but shows an incorrect understanding as you are mixing up first and second systems by referring to damping and overshoot. The initial explanation should be referring to the open loop response with no feedback. (4/10) Section 2: Explanation of DC motor model – Good presentation of equations and good predicted values. However, not all parts are completed and you could have started with the block diagram, explained each part and developed the equations from them. (6/10) Section 3: Explanation of types of speed controllers – Good explanations given. (6/10)

16**ANALYSIS**

Section 1: Testing of first order system – Results are good but analysis is not always accurate. You could have measured the time constants and overall gains for comparison. (5/10) Section 2: Testing of

28

DC motor model – Good evidence of testing, results match theory and are compared well. (8/10) Section 3: Testing of speed controllers – Good evidence of testing but results should be tabulated for comparison. (12/20) Section 4: Conclusions and recommendations – Some offered but no recommendations. (3/10)

PRACTICAL APPLICATION AND DEP.



Sections 1, 2 and 3: Use of Simulink and presentation quality of results - Good use of simulation software including auto-tuning. (8/10)

8

SKILLS FOR PROFESSIONAL PRAC.



All sections – Generally good, figures are listed correctly but there are spelling, grammar errors and some words are missing and there is a large space on page 3. (5/10)

5