CONTROL SYSTEMS
ANALYSIS OF DC MOTOR

A COURSE PROJECT IN THE SUBJECT OF CONTROL SYSTEMS (IC2004) UNDERTAKEN

BY JOHNATHAN FERNANDES, SY Q - 35, GR 1710168

7 JANUARY 2019 – 20 ARIL 2019

UNDER THE GUIDANCE OF PROF. DR. (MRS.) SHILPA Y. SONDKAR

Consider the DC motor system given below:

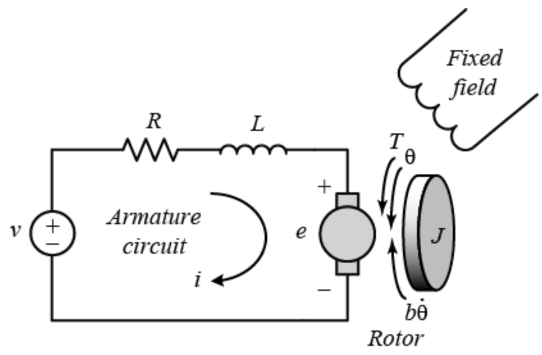


Fig 1: Circuit diagram of DC motor

The transfer function of a system is defined as the ratio of the system output to the system input, both in the s plane. For a DC motor, the input is voltage $V_s(s)$ and the output is angular velocity ω(s). This give us the open loop transfer function:

$$H(s) = \frac{\omega(s)}{V(s)}$$

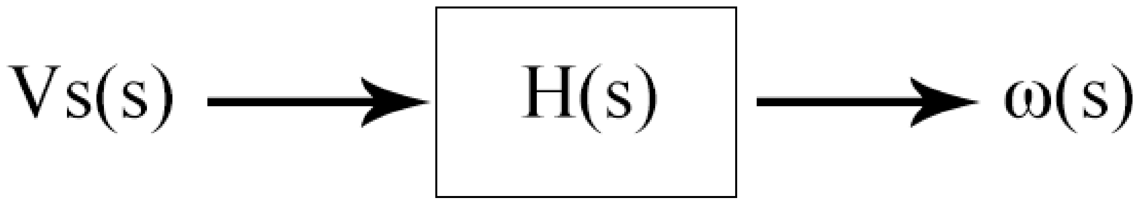


Fig 2: Block diagram of original system

The equations defining this system are:

1) $v_s(t) = R\, i(t) + L \frac{d}{dt}[i(t)] + K_B\, \omega(t)$
2) $\frac{d}{dt}\, \omega(t) = K_T\, i(t)\, I_L^{-1}$

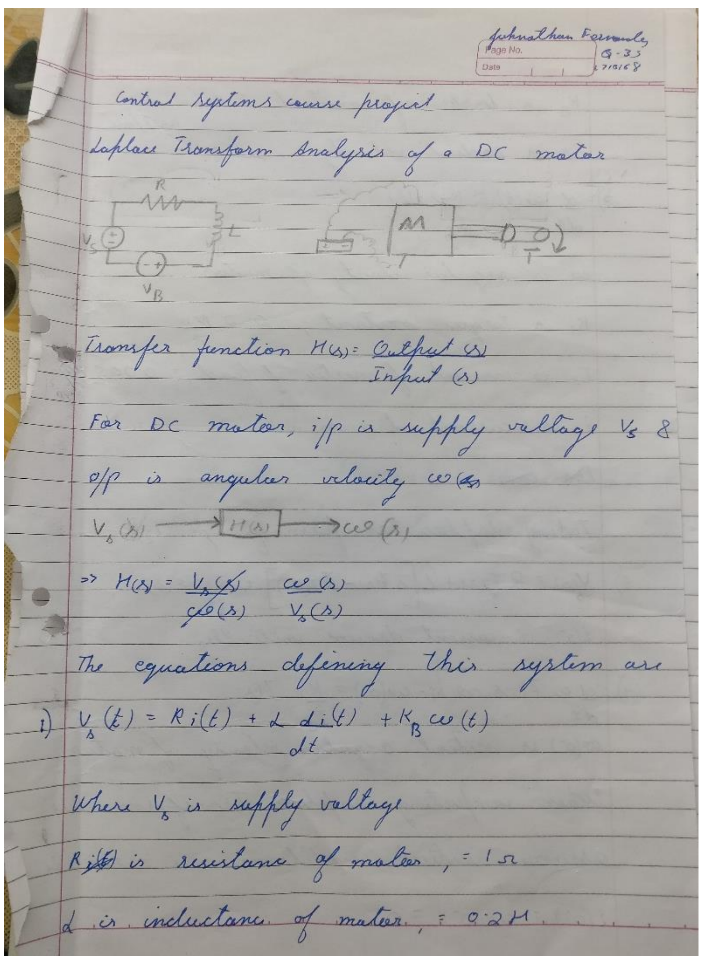
Where

- $V_S(t)$ = Supply voltage
- R = Motor resistance
- i(t) = Current drawn by motor
- L = Motor inductance
- K_B = Back e.m.f. generated
- ω(t) = Angular velocity
- K_T = Torque constant a.k.a. motor damping
- I_L = Moment of inertia of motor

Taking the Laplace transform of equations (1) and (2) :

$$3)\quad H(s) = \omega(s)\, v_s(s)^{-1} = [L\, I_L\, K_T^{-1}\, s^2 + R\, I_L\, K_T^{-1}\, s + K_B]^{-1}$$

Which is the final transfer function of the original system.



Control systems course project

Laplace Transform Analysis of a DC motor

Transfer function $H(s) = \frac{\text{Output}(s)}{\text{Input}(s)}$

For DC motor, i/p is supply voltage V_s &

o/p is angular velocity ω(s)

$V_s(s) \longrightarrow \boxed{H(s)} \longrightarrow \omega(s)$

$\Rightarrow H(s) = \cancel{\frac{V_s(s)}{\omega(s)}} \quad \frac{\omega(s)}{V_s(s)}$

The equations defining this system are

1) $V_s(t) = R\,i(t) + L\frac{di(t)}{dt} + K_B\,\omega(t)$

Where V_s is supply voltage

R is resistance of motor, = 1 Ω

L is inductance of motor, = 0.2 H

Fig 3.1: On paper derivation of transfer function

K_B is back emf constant, $= 0.22 \frac{V}{rad/sec}$

ω is radial velocity

2) $\frac{d}{dt}\omega(t) = \frac{K_T}{I_L} i(t)$

ω is angular velocity of motor

K_T is Torque constant, $= 0.2 \frac{Nm}{A}$

I_L is moment of inertia of load, $= 0.005 \frac{kg}{m^2}$

i is current drawn.

~~For com~~

Taking Laplace Transform of eqn 1 & 2

3) $V_s(s) = R I(s) + L\left[s I(s) - i(0^-)\right] + K_B \omega(s)$

$i(0^-)$ is current drawn initially

4) $\cancel{\frac{d}{dt}\omega(t)}\; s\,\omega(s) - \omega(0^-) = \frac{K_T}{I_L} I(s)$

$\omega(0^-)$ is initial angular velocity of motor

When computing Laplace Transform, we assume initial conditions are zero, ie

$i(0^-) = \omega(0^-) = 0$.

Fig 3.2: On paper derivation of transfer function

From eq 4, $I(s) = \frac{I_L}{K_T} s\,\omega(s)$

sub in 3

$V_s(s) = R\frac{I_L}{K_T} s\,\omega(s) + L s \frac{I_L}{K_T} s\,\omega(s) + K_B\,\omega(s)$

$V_s(s) = \omega(s)\left[\frac{R I_L}{K_T} s + L s^2 \frac{I_L}{K_T} + K_B\right]$

$\Rightarrow \frac{V_s(s)}{\omega(s)} = \left[\frac{R I_L}{K_T} s + L s^2 \frac{I_L}{K_T} + K_B\right]$

$\Rightarrow H(s) = \frac{\omega(s)}{V_s(s)} = \left[\frac{L I_L}{K_T} s^2 + \frac{R I_L}{K_T} s + K_B\right]^{-1}$

Fig 3.3: On paper derivation of transfer function

To analyze the system, we assume certain parameters of the DC motor:

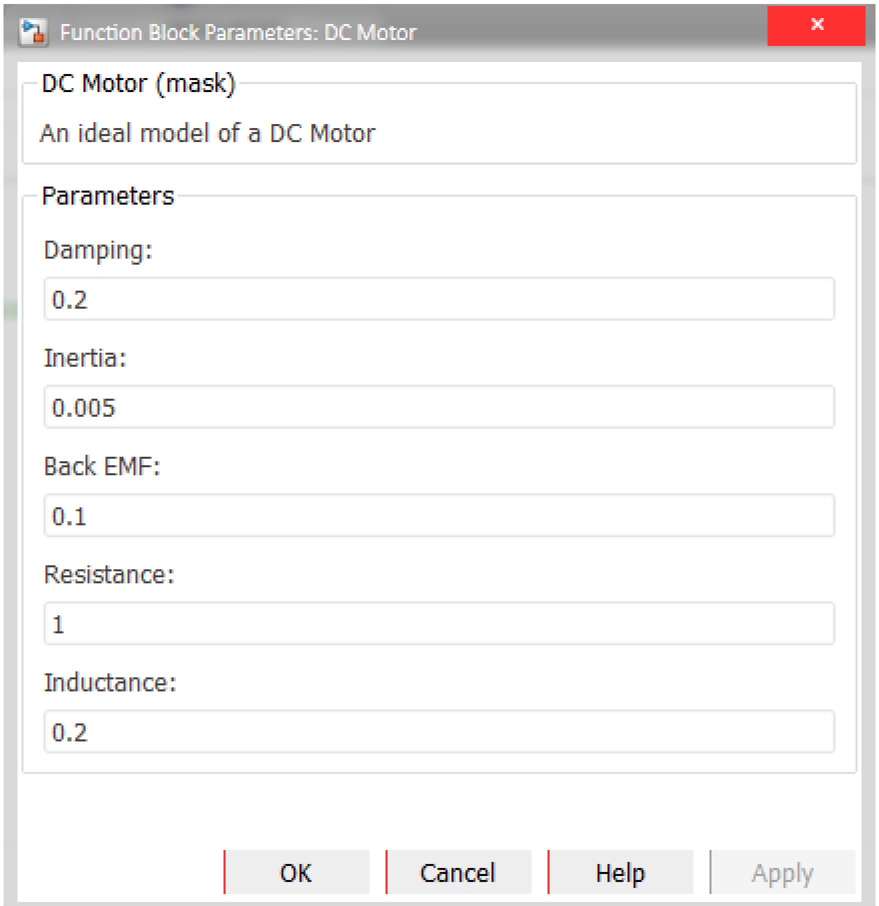


Fig 4: Assumed parameters of DC motor

Analyzing the system in MATLAB:

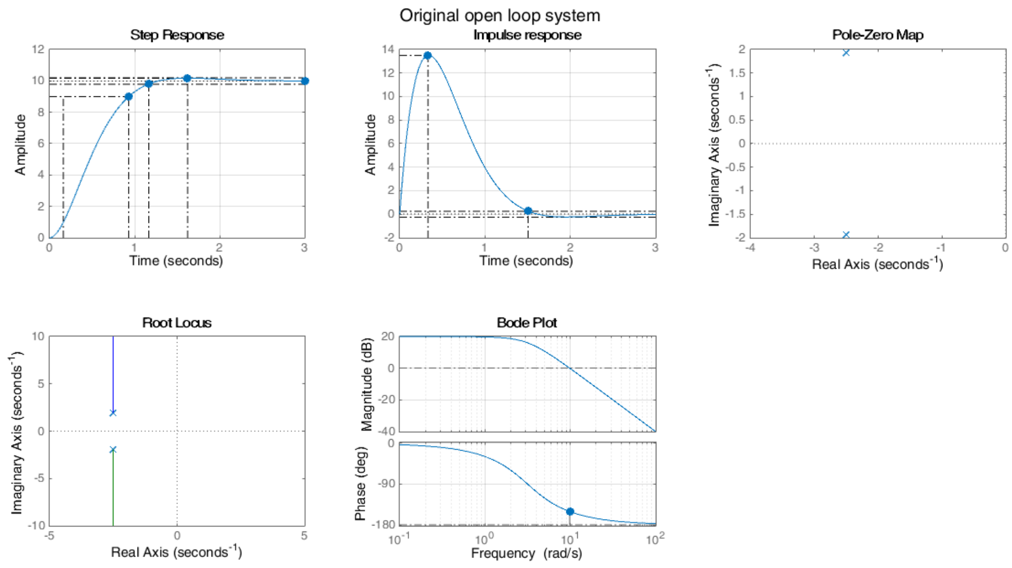


Fig 5: Analysis of original system in MATLAB

In order to improve stability of the system, we will add a PID controller.

A PID controller has a transfer function defined as:

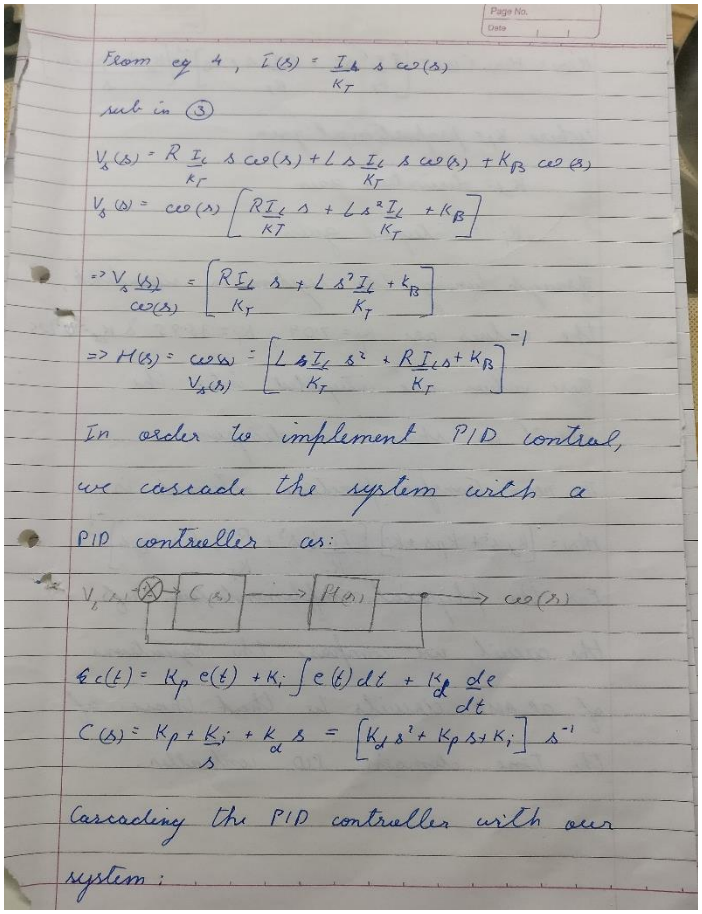
$$c(t) = K_P\, e(t) + K_I \int e(t)dt + K_D \frac{d}{dt} e(t)$$

in the time domain, which we apply a Laplace transform to to obtain:

$$C(s) = K_P + K_I\, s^{-1} + K_D\, s$$

Where

- K_P = Proportional gain constant
- K_I = Integral gain constant
- K_D = Derivative gain constant
- e(t) = error signal of system


From eq 4, $I(s) = \frac{I_L}{K_T} s\,\omega(s)$

sub in 3

$$V_s(s) = R\frac{I_L}{K_T} s\,\omega(s) + Ls\frac{I_L}{K_T} s\,\omega(s) + K_B\,\omega(s)$$

$$V_s(s) = \omega(s)\left[\frac{RI_L}{K_T}s + Ls^2\frac{I_L}{K_T} + K_B\right]$$

$$\Rightarrow \frac{V_s(s)}{\omega(s)} = \left[\frac{RI_L}{K_T}s + Ls^2\frac{I_L}{K_T} + K_B\right]$$

$$\Rightarrow H(s) = \frac{\omega(s)}{V_s(s)} = \left[\frac{LI_L}{K_T}s^2 + \frac{RI_L}{K_T}s + K_B\right]^{-1}$$

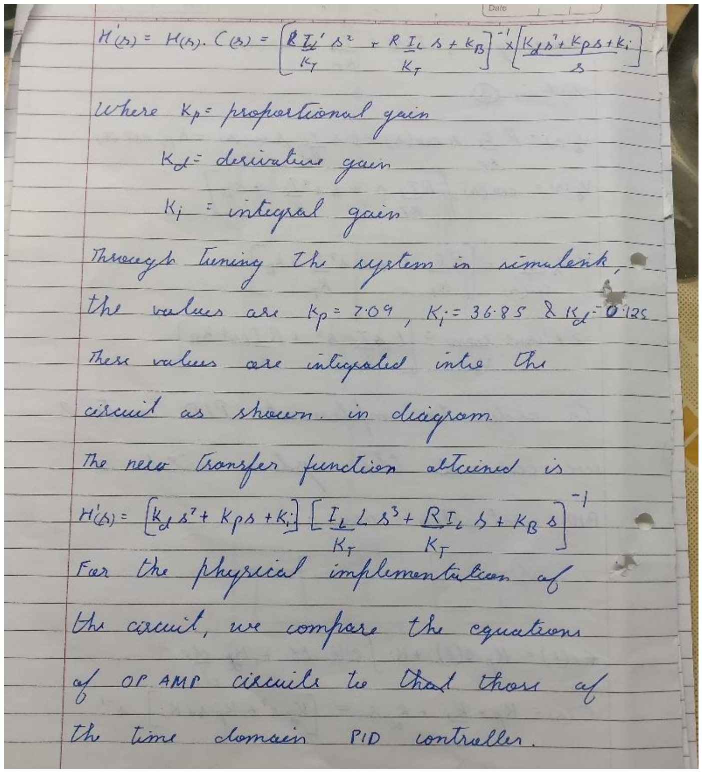
In order to implement PID control, we cascade the system with a PID controller as:

$$c(t) = K_p\,e(t) + K_i\int e(t)\,dt + K_d\frac{de}{dt}$$

$$C(s) = K_p + \frac{K_i}{s} + K_d s = \left[K_d s^2 + K_p s + K_i\right]s^{-1}$$

Cascading the PID controller with our system:

Fig 6.1: On paper derivation of cascaded transfer function



$$H'(s) = H(s).C(s) = \left[\frac{R I_L'}{K_T} s^2 + \frac{R I_L}{K_T} s + K_B\right]^{-1} \times \left[\frac{K_d s^2 + K_p s + K_i}{s}\right]$$

Where K_p = proportional gain

K_d = derivative gain

K_i = integral gain

Through tuning the system in simulink, the values are $K_p = 7.09$, $K_i = 36.85$ & $K_d = 0.125$

These values are integrated into the circuit as shown in diagram.

The new transfer function obtained is

$$H'(s) = [K_d s^2 + K_p s + K_i]\left[\frac{I_L}{K_T} L s^3 + \frac{R I_L}{K_T} s + K_B s\right]^{-1}$$

For the physical implementation of the circuit, we compare the equations of OP AMP circuits to ~~that~~ those of the time domain PID controller.

Fig 6.2: On paper derivation of cascaded transfer function

Cascading the PID controller with our system as shown:

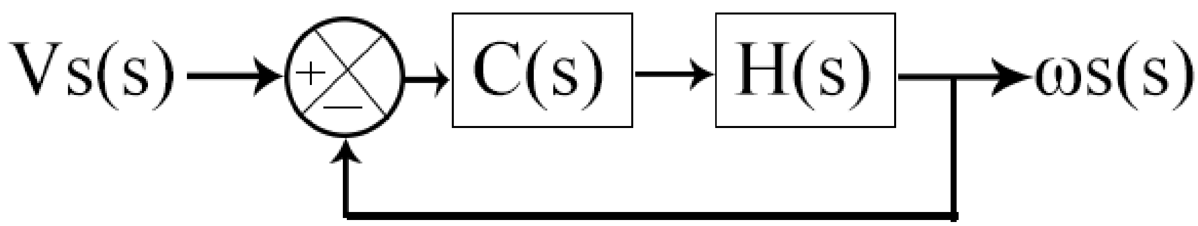


Fig 7: Block diagram of system with PID controller cascaded

The closed loop transfer function of this cascaded system is defined as:

$$H'(s) = [K_D s^2 + K_P s + K_I]\ [(L\ I_L\ K_T^{-1})$$

$$s^3 + (R\ I_L\ K_T^{-1} + K_D)\ s^2 + (K_B + K_P)s + K_I]^{-1}$$

To calculate the values of the gain constants, we construct a model using MATLAB Simulink:

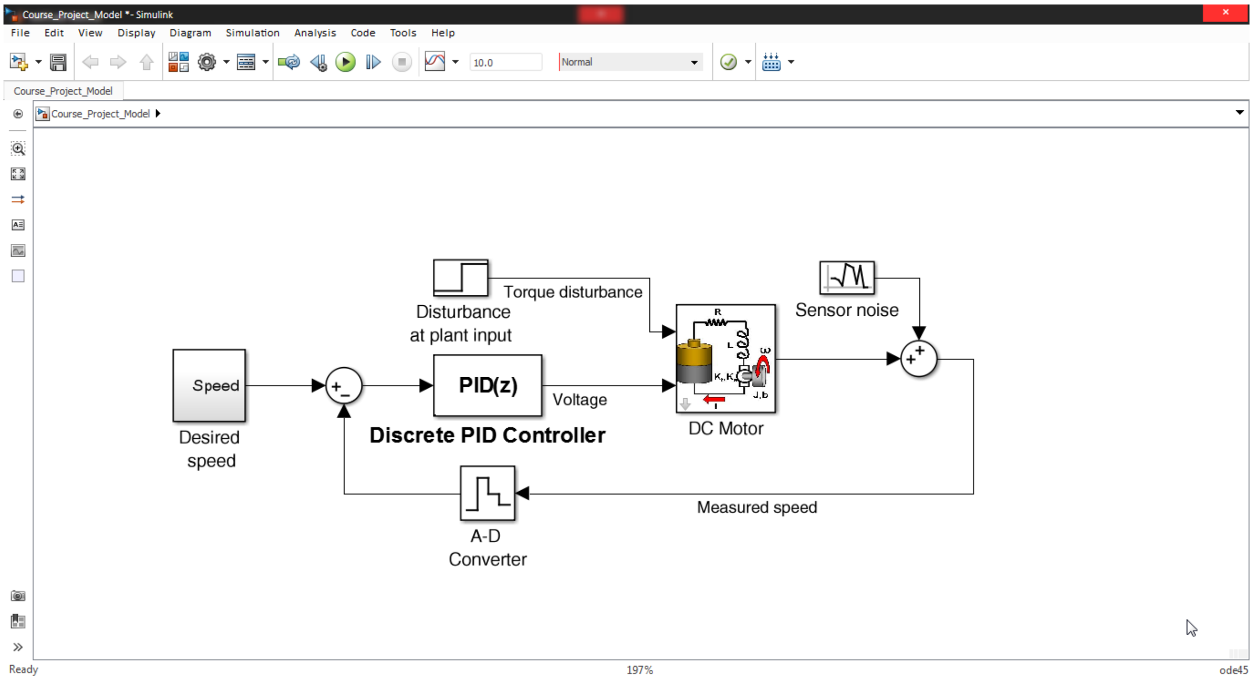


Fig 8: System constructed in MATLAB Simulink

Since the MATLAB Simulink is an ideal system, we artificially insert noise both at the sensor level and torque interference to get a realistic idea of a real world situation.

Using the MATLAB PID tuner tool, we obtain values for the gain constants:

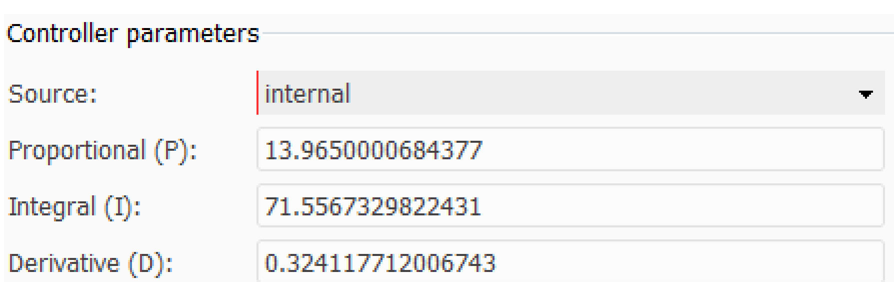


Fig 9: PID controller tuning using MATLAB Simulink

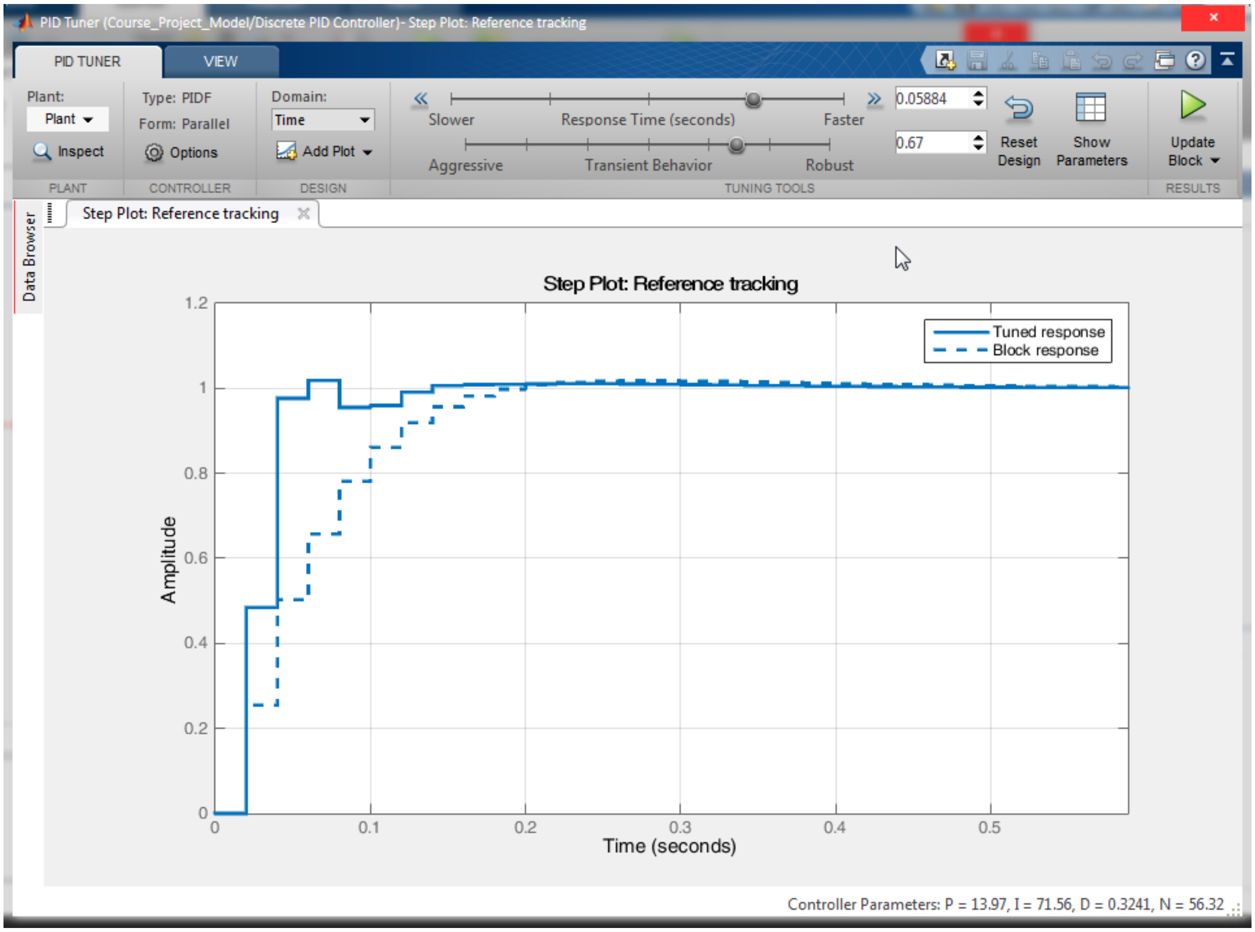


Fig 10: Transient analysis graph using tuned values vs original system

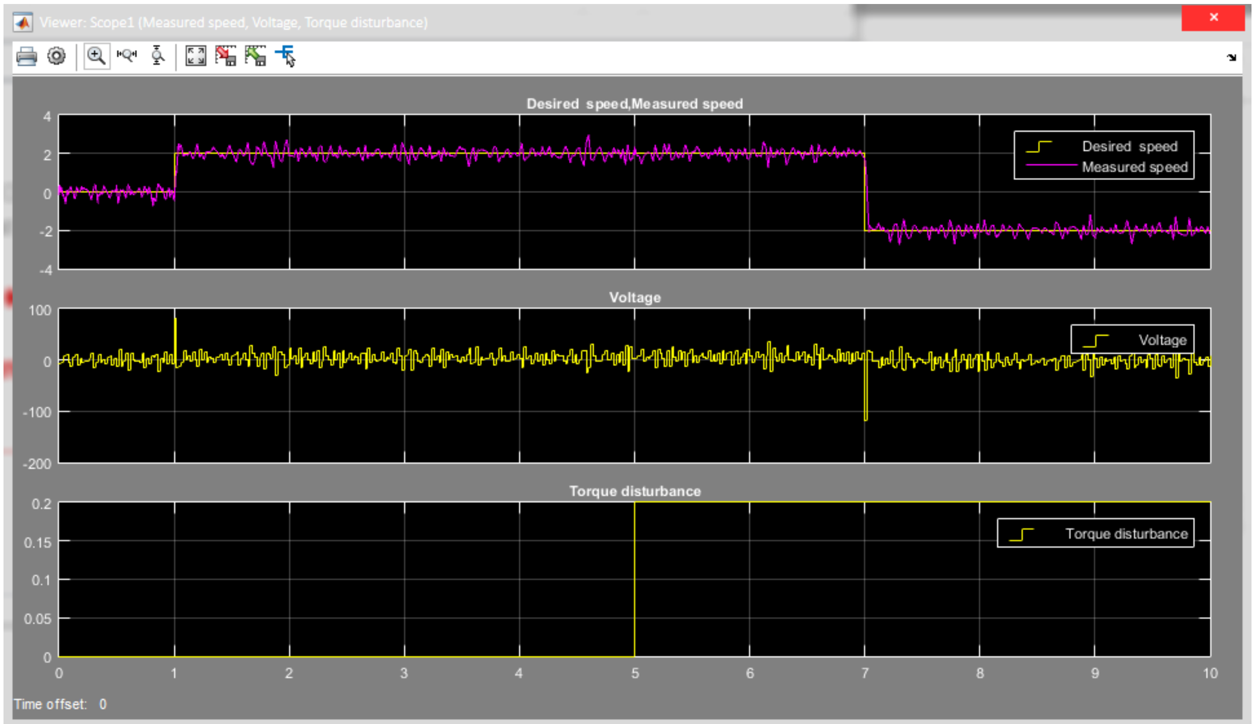


Fig 11: Plots of output showing desired speed, output speed, input voltage and torque disturbance

Using the gain constant values calculated in the MATLAB Simulink model, we carry out the same analysis performed in fig 4

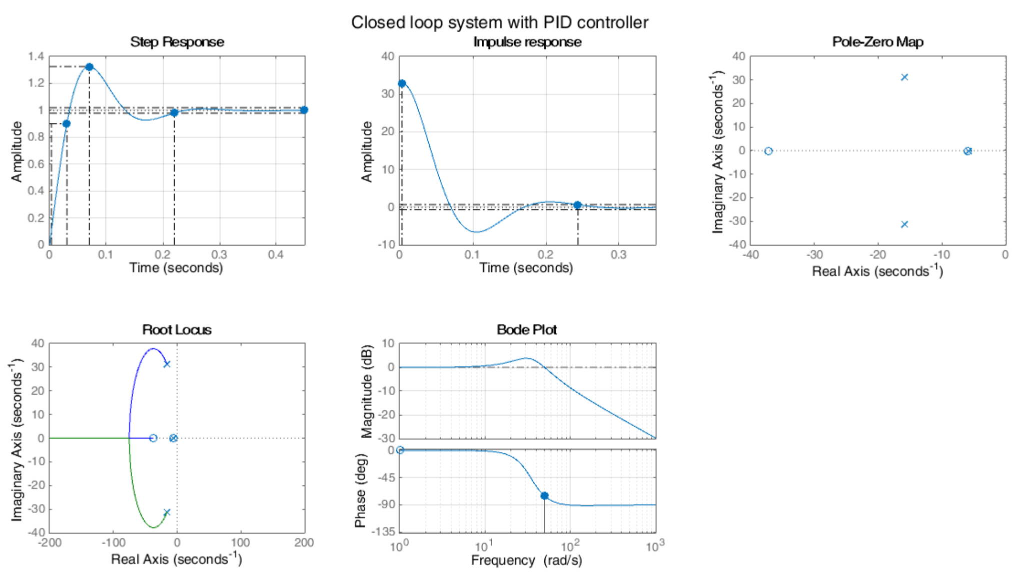


Fig 12: Analysis of modified system in MATLAB

Comparing the MATLAB analysis results of the system with and without the PID controller, we observe the following:

- Step response & Impulse response graphs: The settling time reduces (1.16 -> 0.22 seconds) when the PID controller is implemented.
- Pole – Zero Map & Root Locus: Addition of the PID controller shifts the poles and zeroes further left into the s-plane, hence moving the system from marginally to fully stable.
- Bode Plot: It is observed that the phase crossover frequency $\omega_{pc} = \infty$ while the gain crossover frequency ω_{gc} has a constant value, hence the system is stable. With the addition of the PID controller, the phase margin increases drastically (29.5 -> 104 degrees), thus increasing the stability of the system.

The above stated observations lead us to draw the conclusion that due to the inclusion of the PID controller, system stability has increased, and response time has improved.

Using the gain constant values calculated in MATLAB Simulink, we can construct a PID controller using OPAMP circuits:

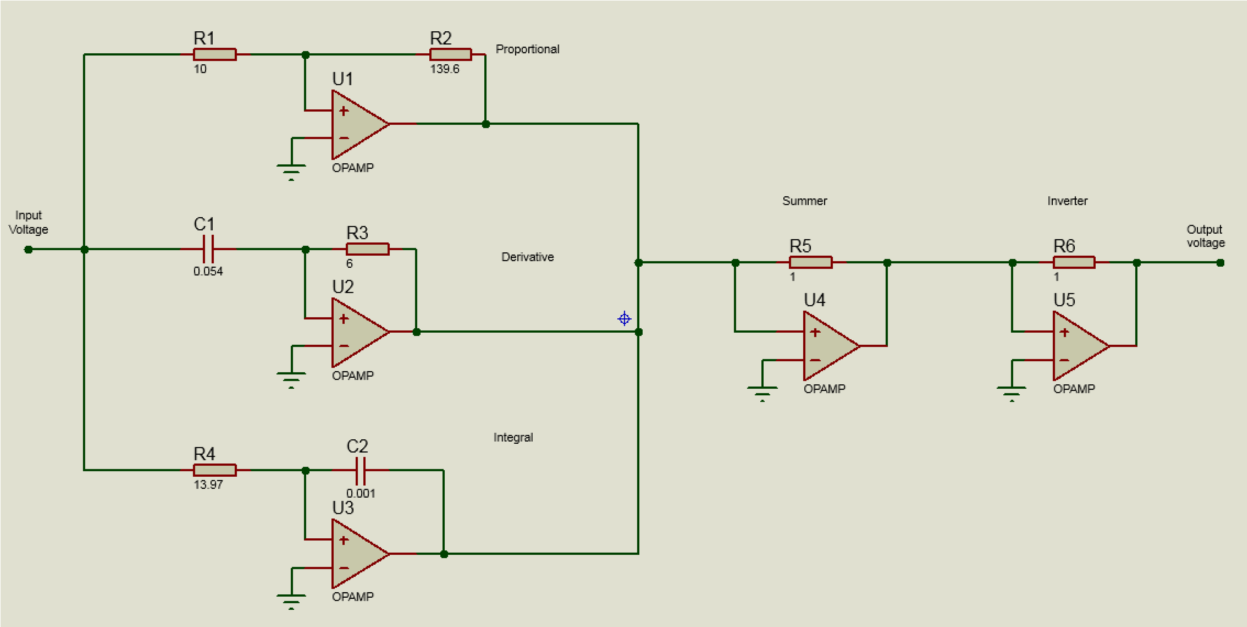


Fig 13: Circuit diagram of PID controller

We use an inverting amplifier for the proportional gain, a differentiator for derivative gain, and an integrator for integral gain. We the use a summing amplifier to combine the values of all 3, and an inverting amplifier with unity gain to correct the phase shift created in the initial step.

Comparing the formulae for the OPAMP circuits with those of the PID controller components:

OPAMP Configuration	OPAMP equation (V_i)	PID equation ($e(t)$)
Proportional	$-R_2 R_1^{-1} V_i$	$K_P\, e(t)$
Integral	$-[R_4 C_2]^{-1} \int V_i\, dt$	$K_I \int e(t)\, dt$
Derivative	$-R_3 C_1 \frac{d}{dt} V_i$	$K_D \frac{d}{dt} e(t)$

From these equations, we calculate values of resistors and capacitors required to obtain the gain values required.

Through this project, we understood how to analyze a system using various functions in MATLAB, PID controllers, their uses, importance, applications in systems and their physical implementation using OPAMP circuitry. We got the opportunity to observe the changes the controller had on the system side by side, and obtained a better understanding of a subject that is normally not covered in theory classes.

Bibliography:

- Prof. Dr. (Mrs.) Shilpa Y. Sondkar - Course professor, lab instructor & project guide
- http://www.ecircuitcenter.com - PID controller circuit concept and design
- https://in.mathworks.com - MATLAB related tutorials and guides
- https://www.electronics-tutorials.com - OPAMP circuitry and related equations
- https://www.youtube.com/user/DarrylMorrell - Instructional videos on the subject of Laplace transform analysis of DC motor
- MATLAB Simulink - System analysis & modelling
- Proteus – Electronic circuit design and simulation