Hochschule
für angewandte Wissenschaften
Würzburg-Schweinfurt

CONTROL SYSTEMS

LAB REPORT

Group Number: 55

- MD Mazidur Rehman(4020084)
- Divyesh Jayswal(4021402)

Lab V5:Speed Control of a DC Motor

Controller Design

In this control scheme, two cascaded feedback loops were used to control the motor. The inner loop was used to bring the armature current to a desired value by adjusting the armature voltage. The outer loop was used to control the motor speed. The control signal for the outer loop is the desired value of the armature current, which served as set-point for the inner loop.

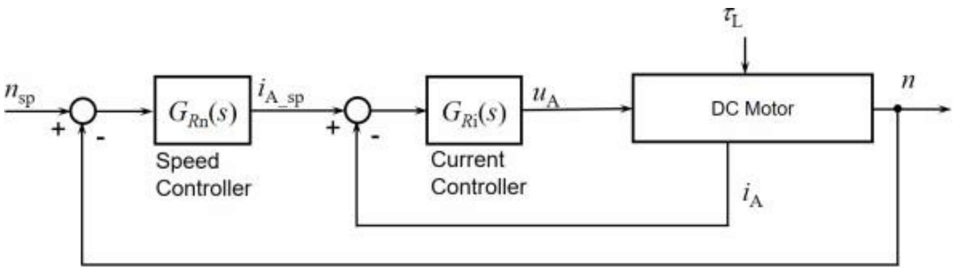

Figure 1: Cascade control scheme

1. Designing the current controller

A PI controller with the following transfer function was used to control the armature current:

$$G_{Ri}(s) = K_{Ri}\left(1 + \frac{1}{T_{Ii}s}\right)$$

The controller was designed using the method known as “modulus optimum”. During the design process, the dependency of induced emf e_M on the current itself was neglected (e_M is considered as an unknown disturbance).

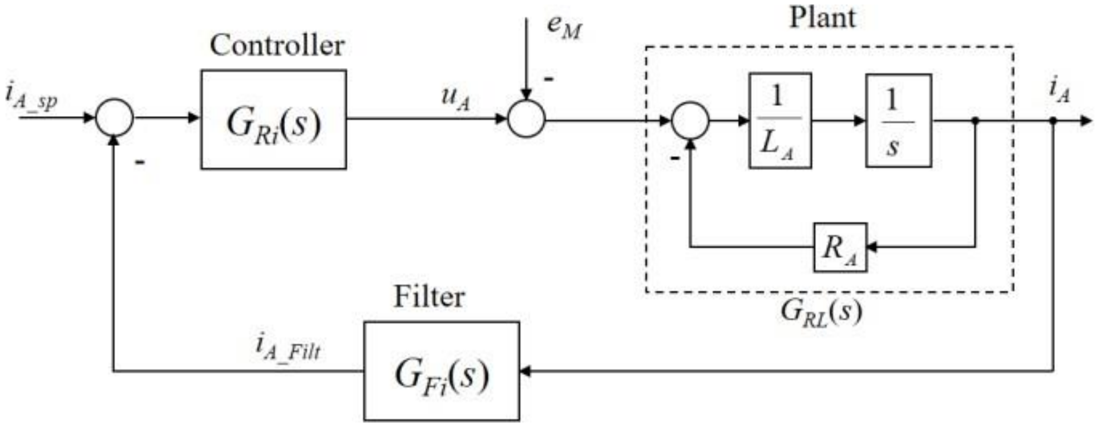

Figure 2: Current control loop

The following steps were executed to design the current controller:

1. Plant model $G_{RL}(s)$ was derived by neglecting the disturbance e_M:

$$G_{RL}(s) = \frac{I_A(s)}{U_A(s)} = \frac{1}{R_A + L_A s}$$

2. Plant model was multiplied with the filter transfer function $G_{F_i}(s) = \frac{1}{T_{Fi}s+1}$

$$G_{RLF}(s) = G_{RL}(s)G_{Fi}(s) = \frac{1}{L_A T_{Fi}s^2 + (R_A T_{Fi} + L_A)s + R_A}$$

3. The value for the parameter T_{Ii} was set equal to the time constant L_A/R_A of the electrical circuit.

$$T_{Ii} = \frac{L_A}{R_A} = \frac{5.01 \cdot 10^{-3}}{1.67} \approx 3\,ms$$

4. The controller was implemented on a digital computer with a sample time of T_S=0.001s. The dead-time caused by the sample and hold process was considered while calculating controller gain K_{Ri} according to the following expression:

$$K_{Ri} = \frac{T_{Ii}}{2K_{RL}(T_{Fi} + 0.5T_s)} = \frac{L_A}{2T_{Fi} + T_s} = \frac{5.01 \cdot 10^{-3}}{2 \times 0.002 + 0.001} = 1.002$$

2. Designing the speed controller

A PI controller was used to control the motor speed in the outer loop and was designed using the symmetrical optimum method. Consider the following controller transfer function:

$$G_{Rn} = K_{Rn}\left(1 + \frac{1}{T_{In}s}\right)$$

The plant model required to design this controller was derived from the block diagram given in Figure 3.

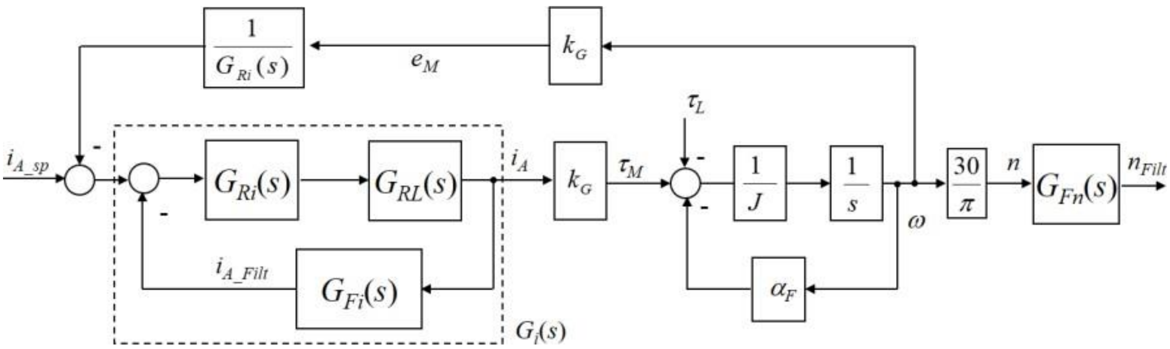

Figure 3: Plant model for outer loop

The speed controller was designed by performing the following steps:

1. Closed loop transfer function of the inner loop was calculated:

$$G_i(s) = \frac{I_A(s)}{I_{A\_sp}(s)} = \frac{G_{Ri}(s) \cdot G_{RL}(s)}{1 + G_{Fi}(s)G_{Ri}(s) \cdot G_{RL}(s)}$$

2. The transfer function of the entire model (without speed signal filter) was calculated:

$$G_n(s) = \frac{N(s)}{I_{A\_sp}(s)} = \frac{G_i(s) \cdot k_G \cdot \left(\frac{\frac{1}{Js}}{1 + \frac{\alpha_F}{Js}}\right)}{1 + G_i(s) \cdot k_G \cdot \left(\frac{\frac{1}{Js}}{1 + \frac{\alpha_F}{Js}}\right) \cdot \frac{k_G}{G_{Ri}(s)}} \cdot \frac{30}{\pi}$$

3. Considering the speed signal filter

$$G_{Sn}(s) = G_n(s) \cdot G_{Fn}(s) = G_n(s) \cdot \frac{1}{T_{Fn}s + 1}$$

4. The values of controller parameters K_{Rn} and T_{In} were determined using the symmetrical optimum method with a phase margin of ϕ_M = 37° using the Control System Designer app in MATLAB. This small value of the phase margin is helpful for disturbance rejection.

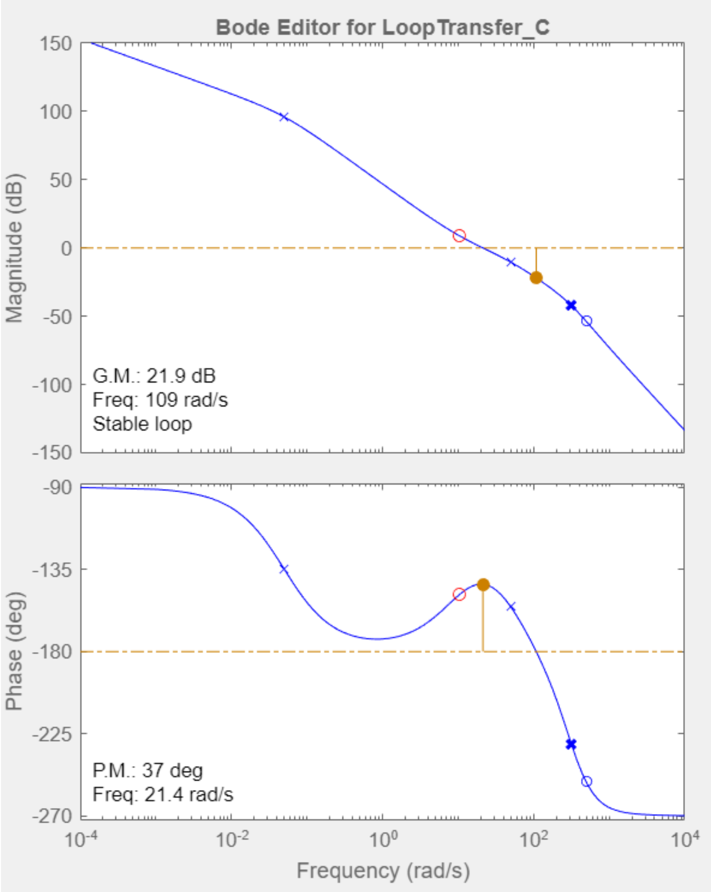

Figure 4: Control System Designer app screenshot

The controller transfer function of the speed controller was found from the Control System Designer app to be

$$G_{Rn}(s) = \frac{0.094267\ (s + 10.61)}{s}$$

From the above transfer function, we can infer that K_{Rn} = 0.094267 and T_{In} = 1/10.61 = 0.09425 s.

3. Simulation of the control system

After the calculation of the parameters we performed the simulation of the cascade control. The speed controller and current controller were implemented with anti-windup. The block diagrams of cascade control, current controller with anti-windup and speed controller with anti-windup are shown in Figure 5,6 and 7 respectively.

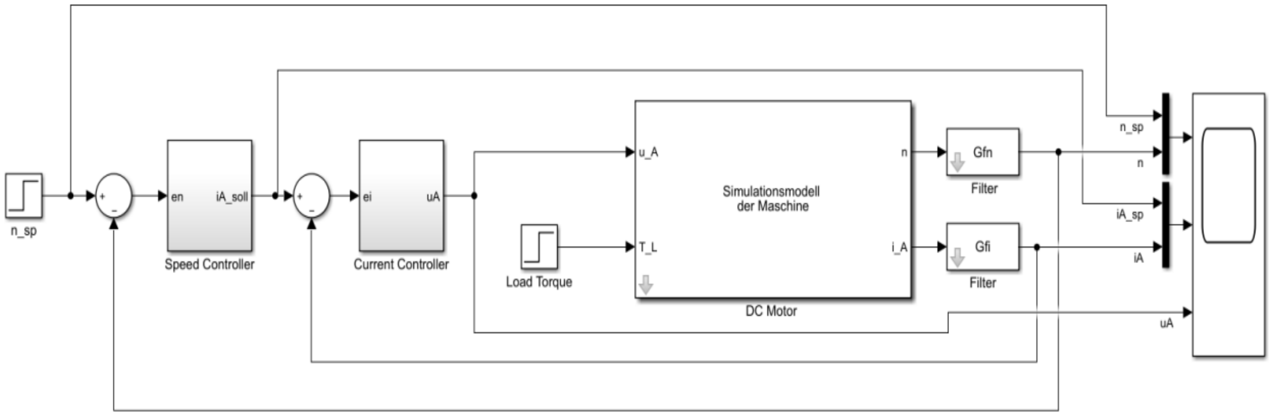

Figure 5: Block diagram for the simulation of cascade control

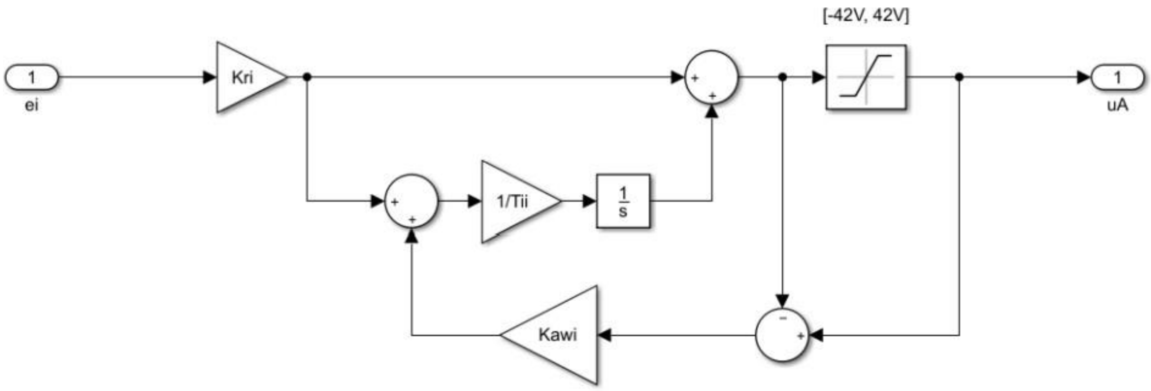

Figure 6: Current controller – a PI controller with Anti-Windup

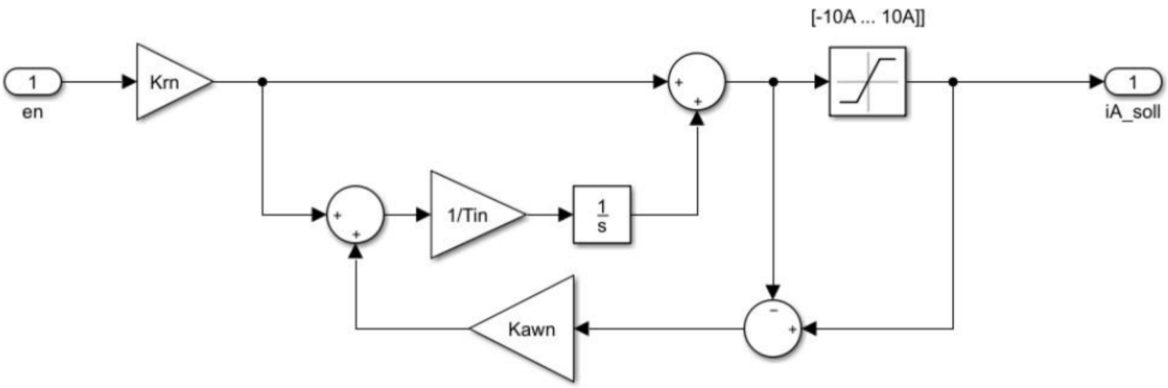

Figure 5: Speed controller – a PI controller with Anti-Windup

The simulation was performed for 10 seconds and the results are shown in the graphs below.

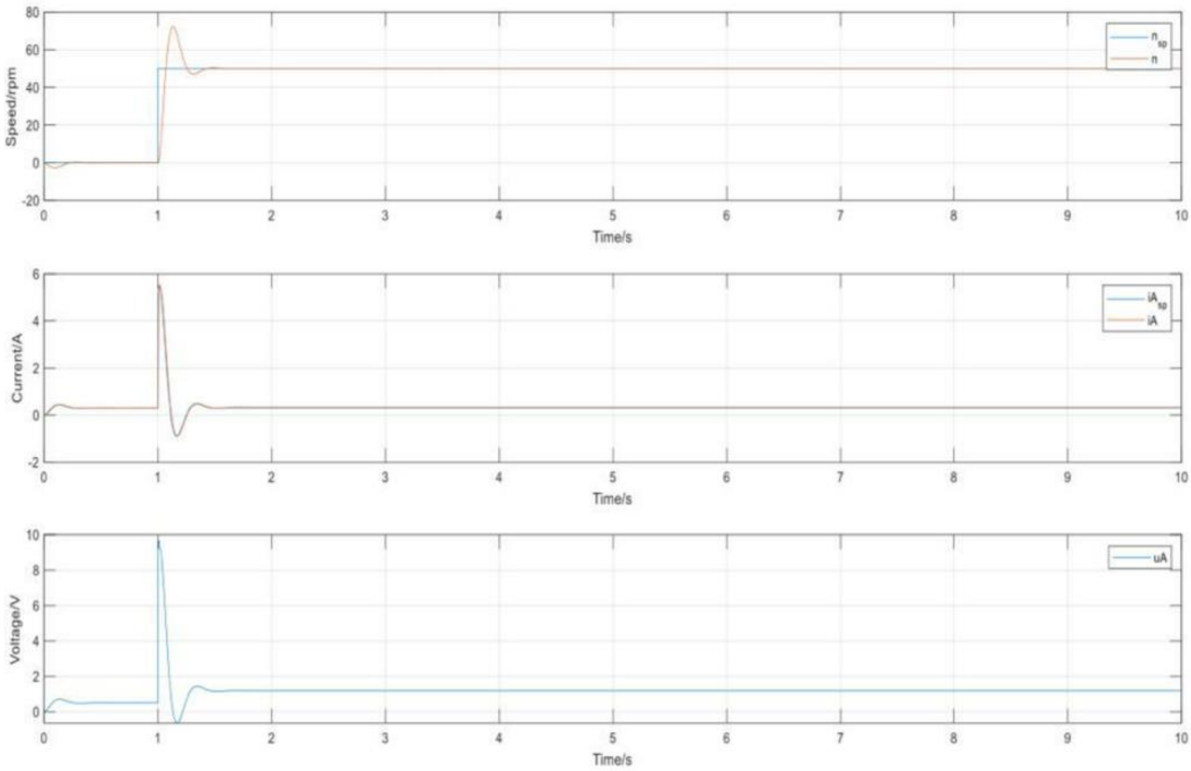

Figure 6: Simulation graph of n_{sp} = 50 $min^{-1}\cdot\varepsilon(t-1)$

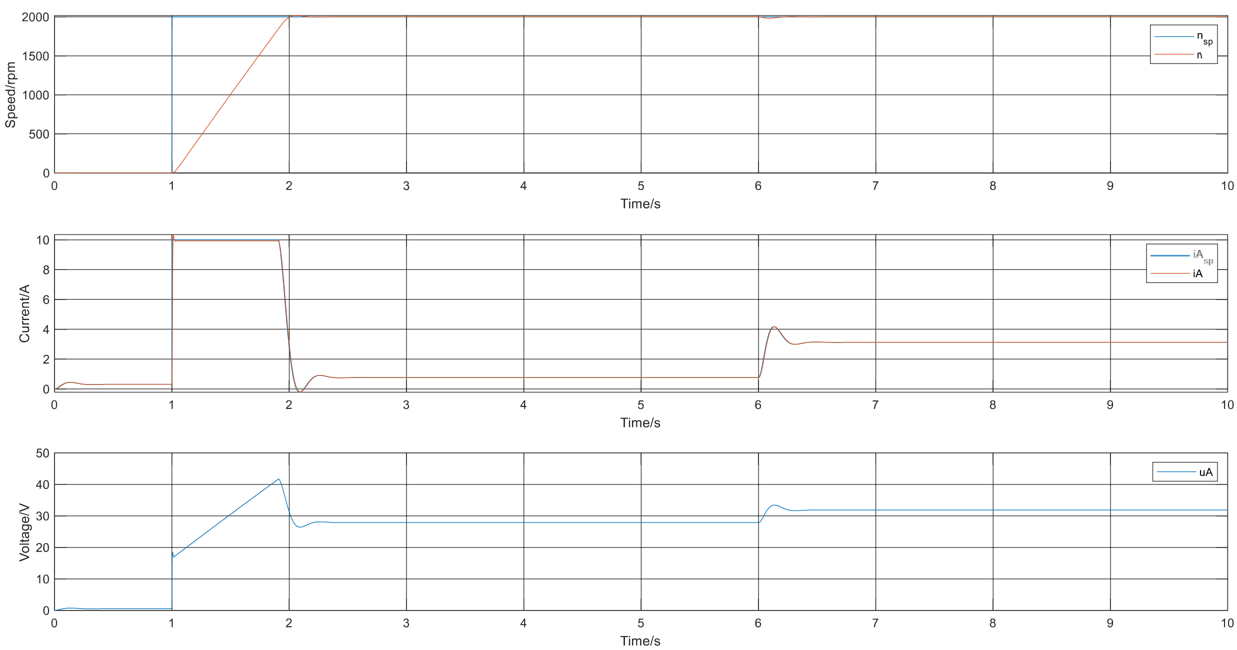

Figure 7: Simulation graph of n_{sp} = 2000 $min^{-1}\cdot\varepsilon(t-1)$ and $\tau_{load}(t)$=0.3Nm$\cdot\varepsilon(t-6)$

From figure 8, it can be seen that when the control deviation is positive, the graph of current and voltage is rising and as soon as the control deviation is negative, the graph is falling and after that it will achieve its steady state. From figure 9, it can be seen that as the speed is increasing, the current signal is also increasing but after some time, it is not increasing but remains steady at 10A because of the saturation we have set to[-10V,10V]. The speed control has large overshoot for smaller steps of set point as it reaches the small set point relatively quickly. But for large steps of set point there is almost no overshoot as the plant output reaches the set-point relatively slowly due to the actuator saturation. Also, there is some disturbance rejection after 6 seconds which is because of the torque load which was implemented. The disturbance rejection is very good as the set point is reached back in very less time.

4. Controller test on the real motor

After performing simulation of the control system, we performed the controller test on the real motor. All the necessary parameters were implemented as shown in the Simulink block diagram below.

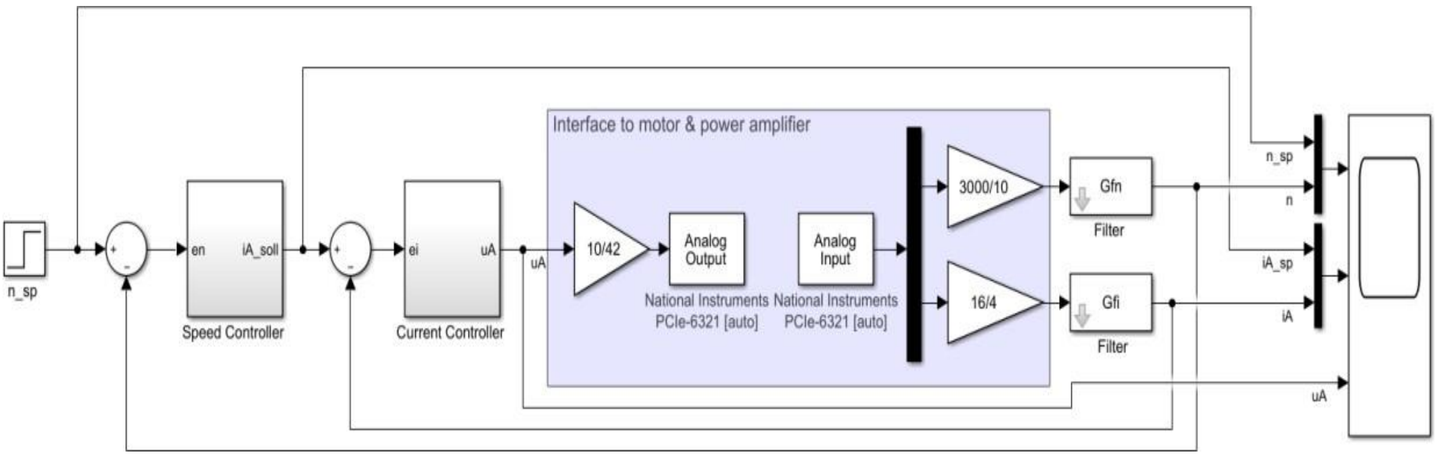

Figure 8: Simulink block diagram for the real-time control

The control loop response was measured using the following set-point trajectories:

a. $n_{sp}(t) = 50\text{min}^{-1} \cdot \epsilon(t-1)$
b. $n_{sp}(t) = 2000\text{min}^{-1} \cdot \epsilon(t-1)$ and switch on the load manually after about 6 seconds.

The result of control loop response is shown in graphs below:

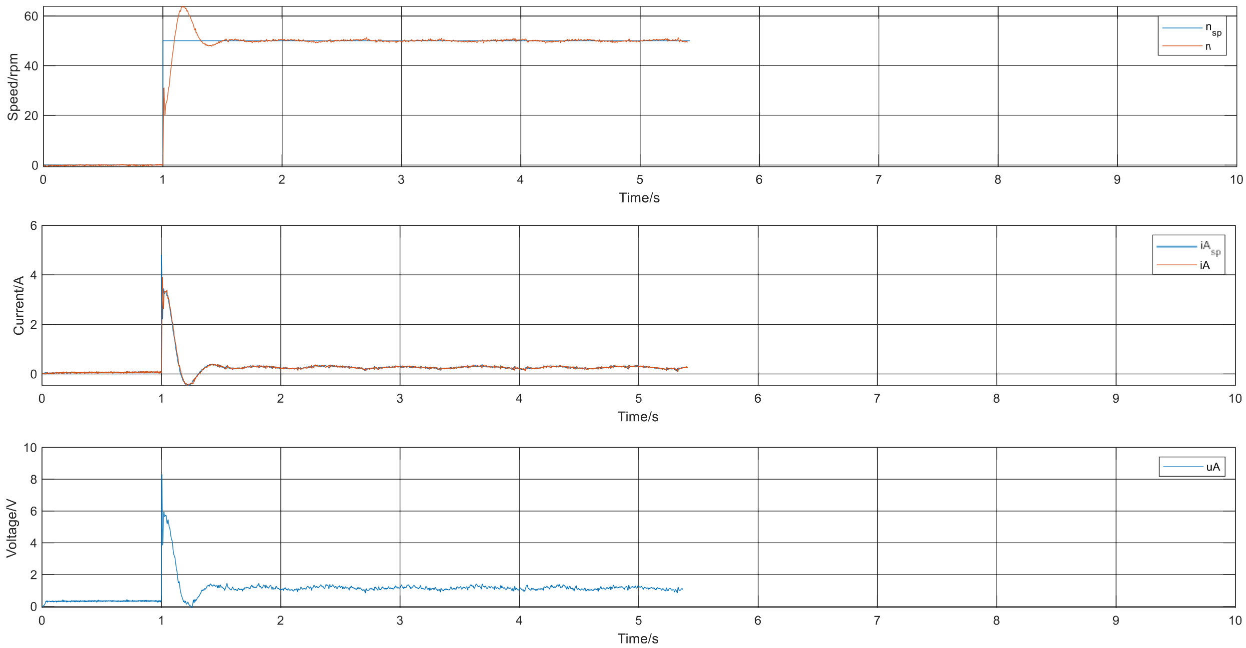

Figure 9:Graph of $n_{sp}(t) = 50\text{min}^{-1} \cdot \epsilon(t-1)$

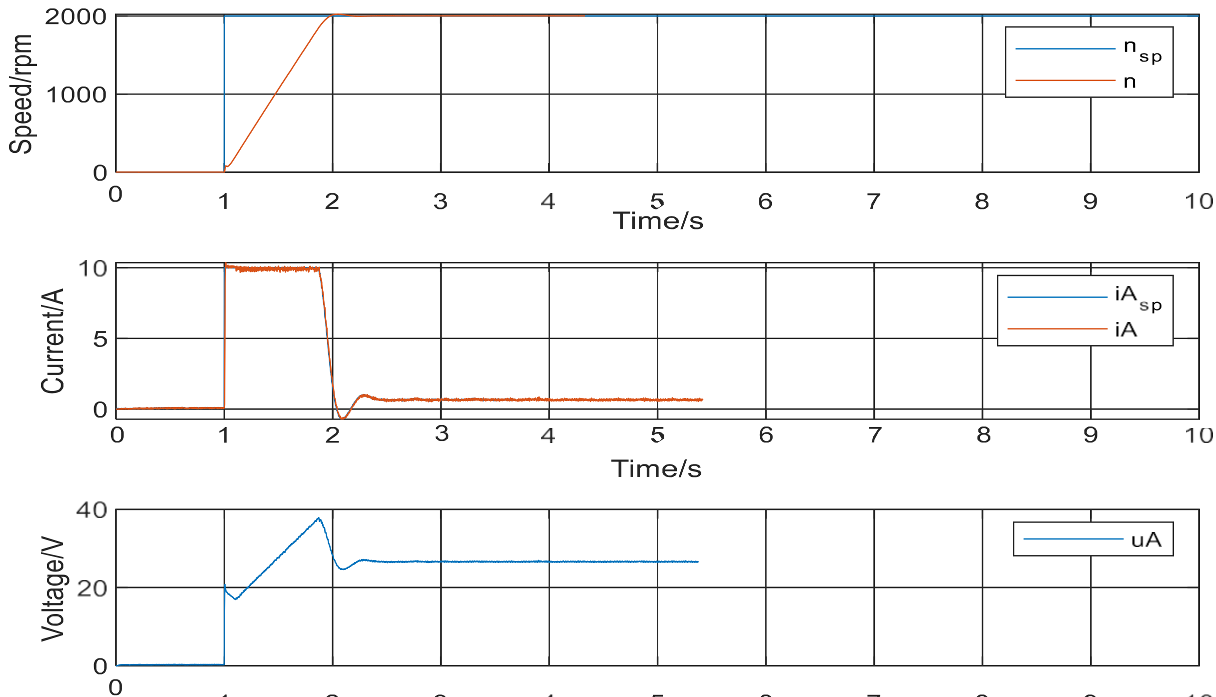

Figure 10:Graph of $n_{sp}(t) = 2000\text{min}^{-1} \cdot \epsilon(t-1)$

Conclusion:

From the figures 11 and 12 we can say the graphs of controller test on the real motor are very similar to the ones we obtained during simulation. Some noise can be seen in the graphs of real motor but it is tolerable. Hence, we can conclude that the controller parameters determined and simulation performed are accurate and satisfactory.