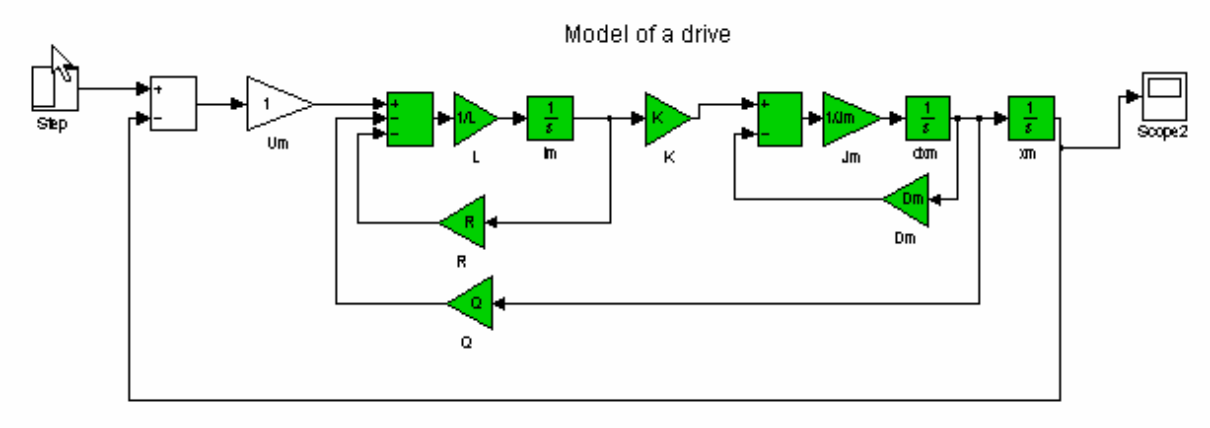
Exercise 7:

Controller design for a position control

In Exercise 5 a model of a position control with a P-controller was simulated with Simulink. The control will now be improved using the control-toolbox of Matlab for controller design. For each controller following steps must be done:

First the model of the controlled system will be exported directly from Simulink. Delete the controller and use the input and output points to define the input and the output of the system. Next step a controller is designed using the control toolbox. Finally the controller will be tested inside Simulink. Examine in Simulink the influence of a limited controller output and a limited motor current.



Simulink model of a drive

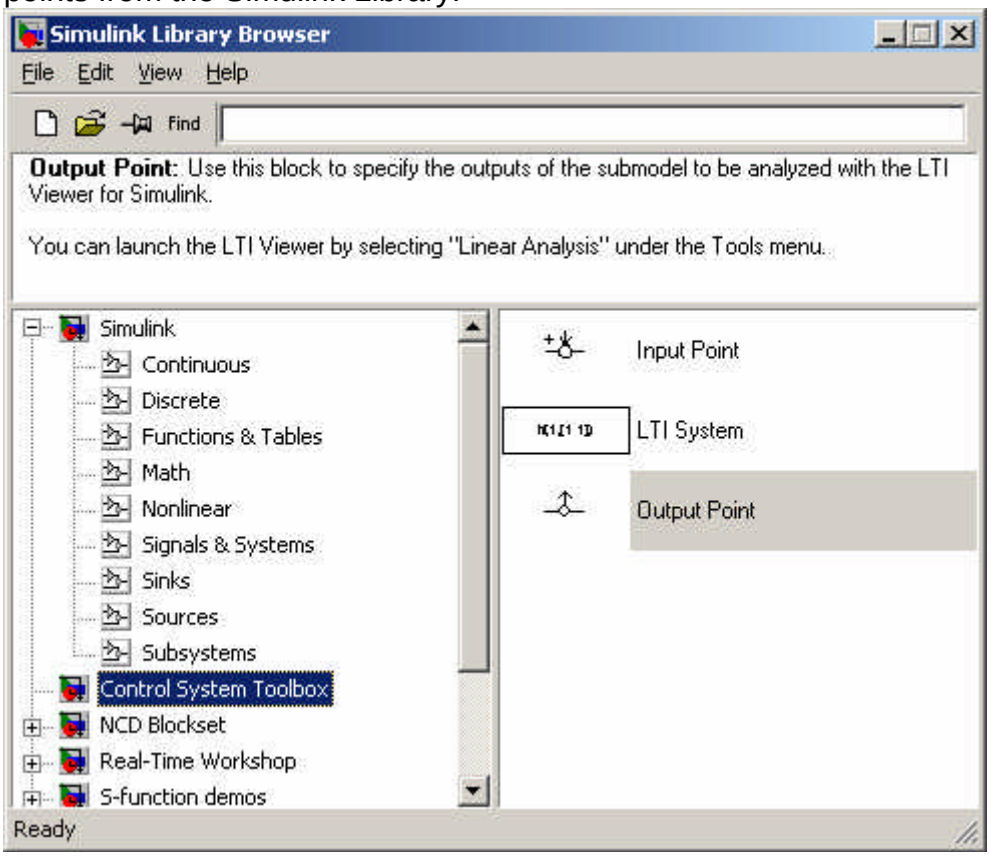
Task:

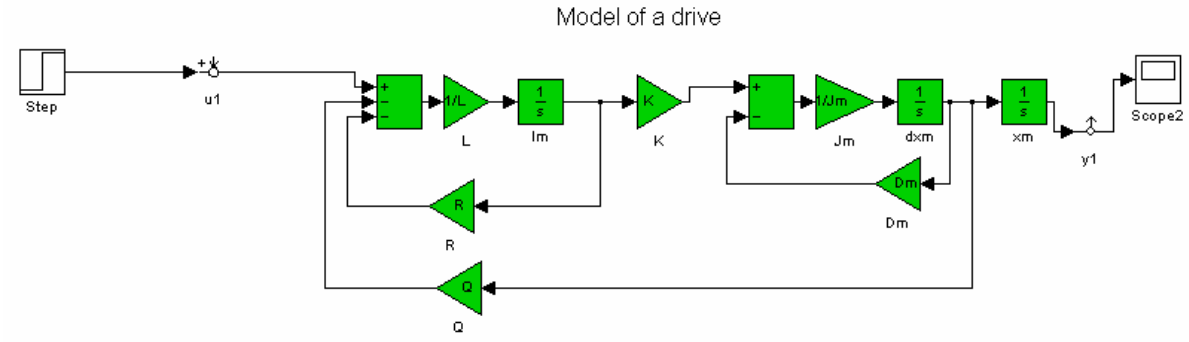
a) Replace the P controller with a PID-Controller. Compare the results with the P controller.

b) Build a cascade controller for the position control. The inner loops are the motor current, and the velocity. Compare the result with the P and PID controller.

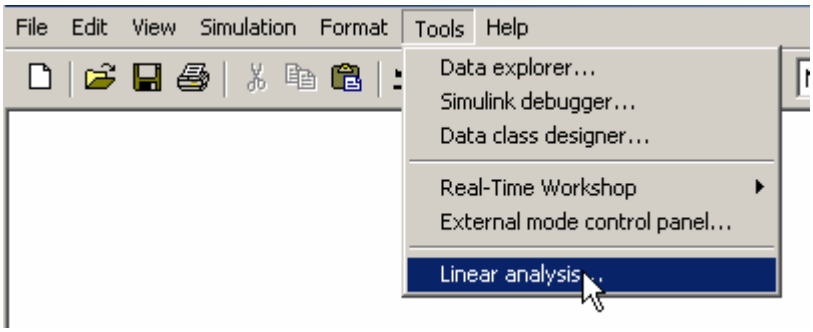
Document and print your work using word.

Add the output and input points to the open loop model using the Input and Output points from the Simulink Library.

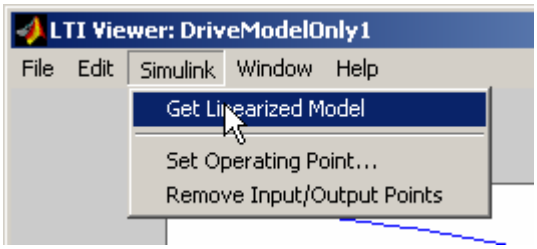




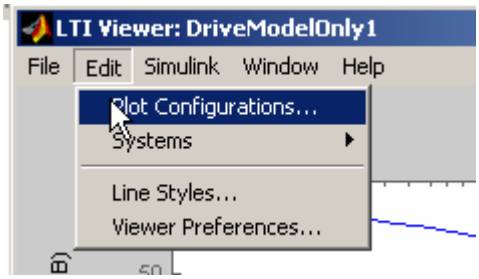
Call the Lti-Viewer



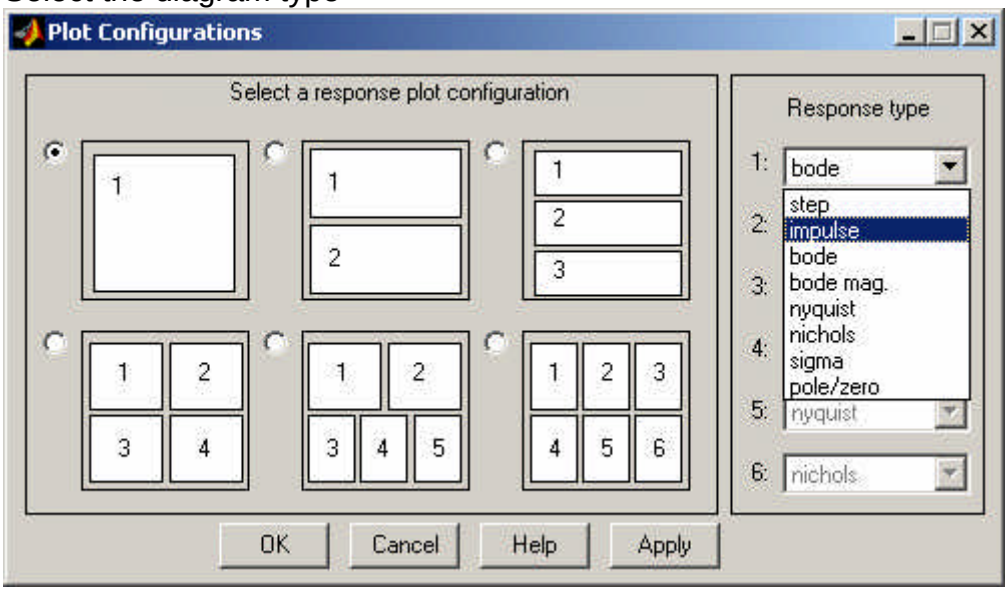
Import the model from Simulink



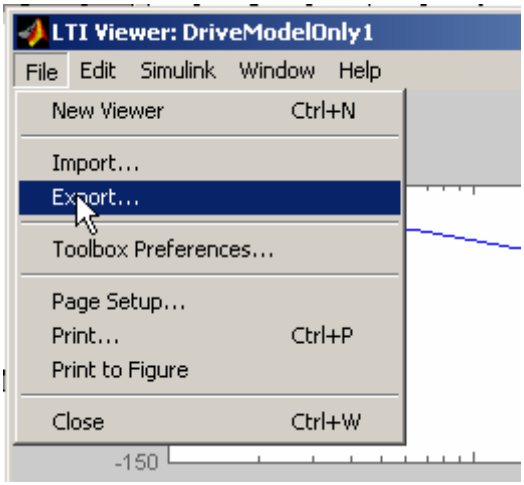
Check the step-response and the bode diagram in the LTI-Viewer



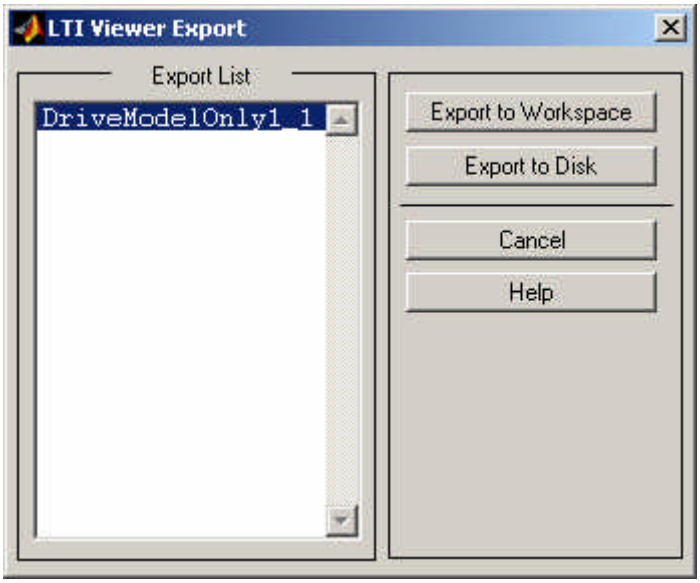
Select the diagram type



Export the model to Matlab-Workspace



Select model and Export to Workspace



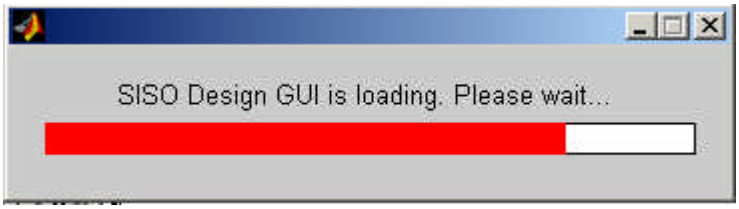
Inside the workspace the model will be transfered to a transfer function:

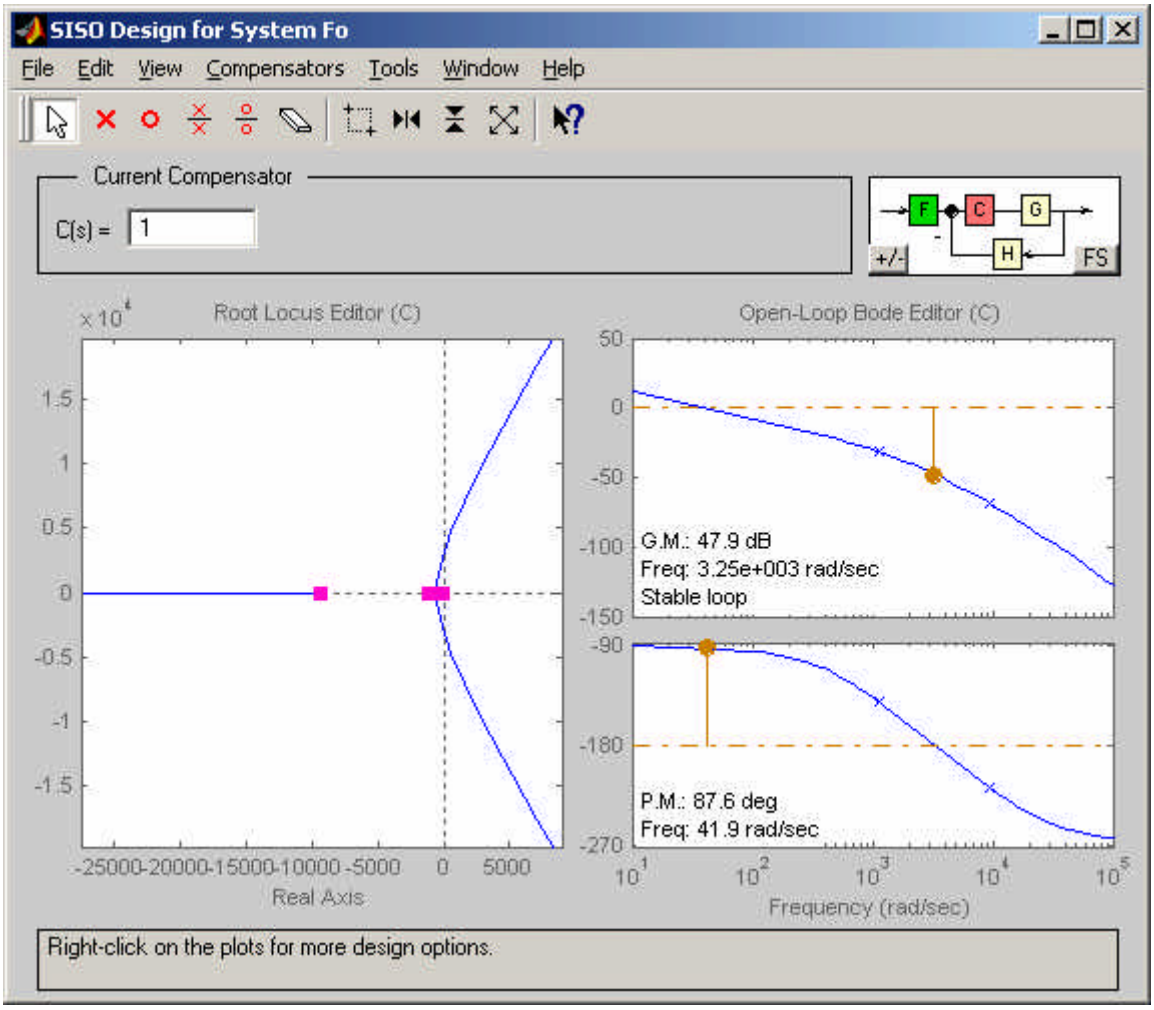
```
>> Fo=tf(DriveModelOnly1_1)

Transfer function from input "u1" to output "y1":
            4.423e008
---------------------------------
s^3 + 1.045e004 s^2 + 1.054e007 s
```

Now the controller is designed with the control toolbox. Here we use the “sisotool”

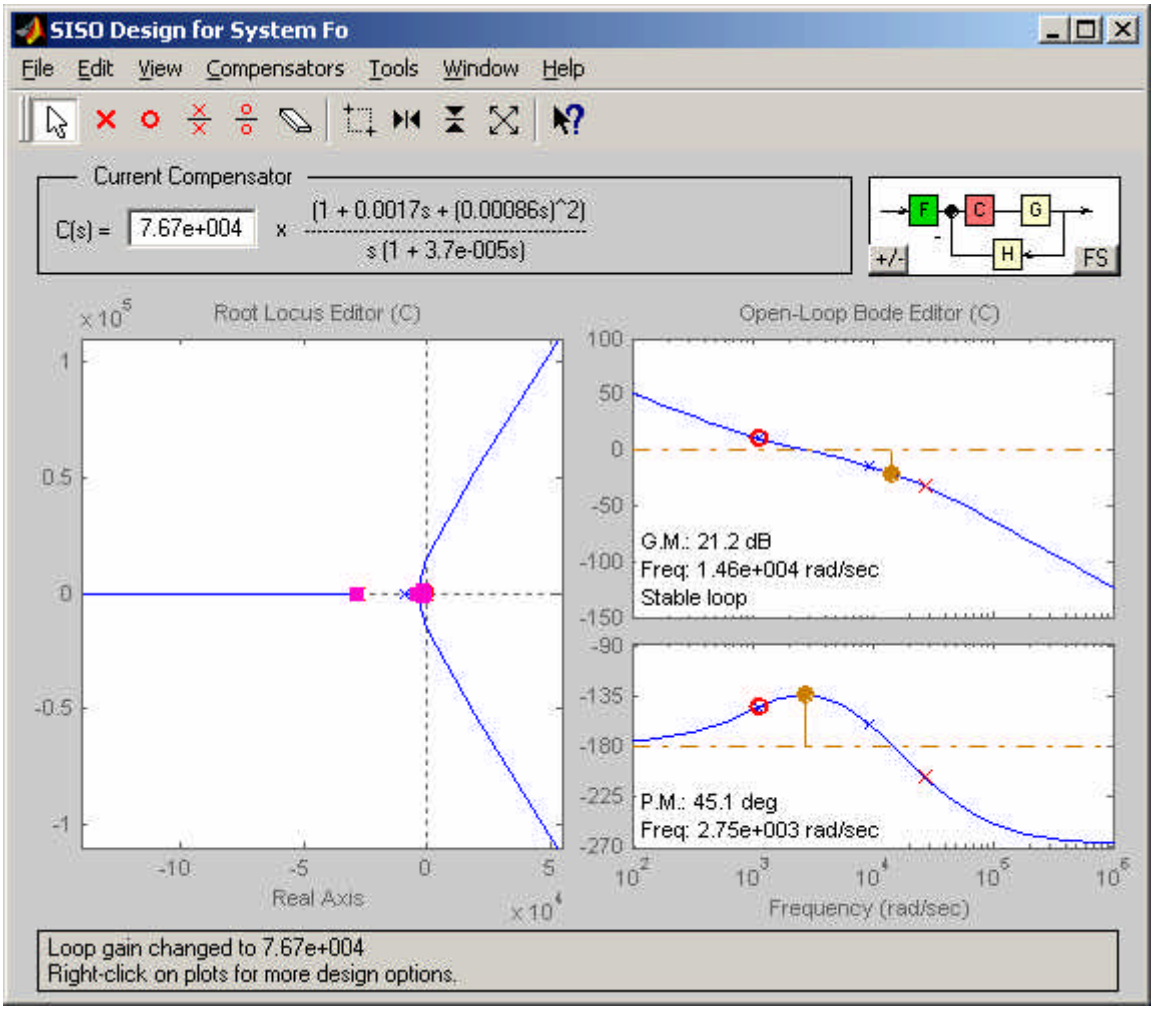
```
>>
>> sisotool(Fo)
```

The sisotool starts:

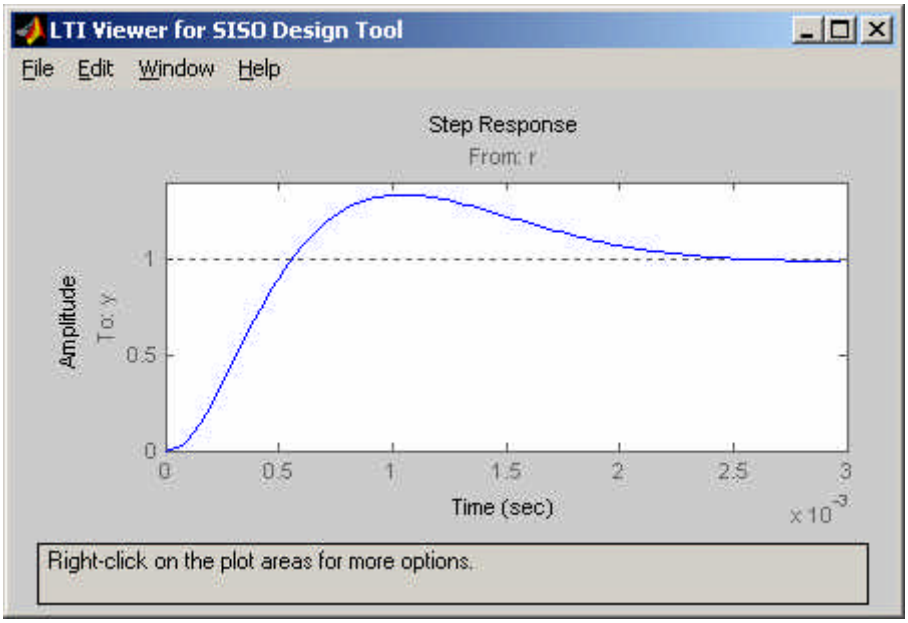




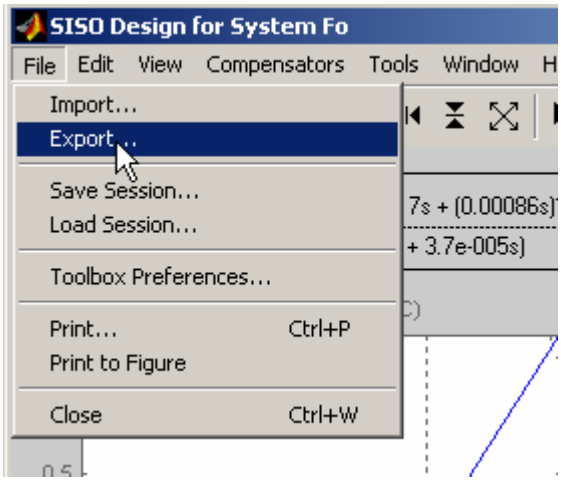
Now the PID controller is designed inside the sisotool using the root locus and the open-loop bode diagram. Design a real, not an ideal PID controller. That means that the order of the numinator should not be greater than the donominator.



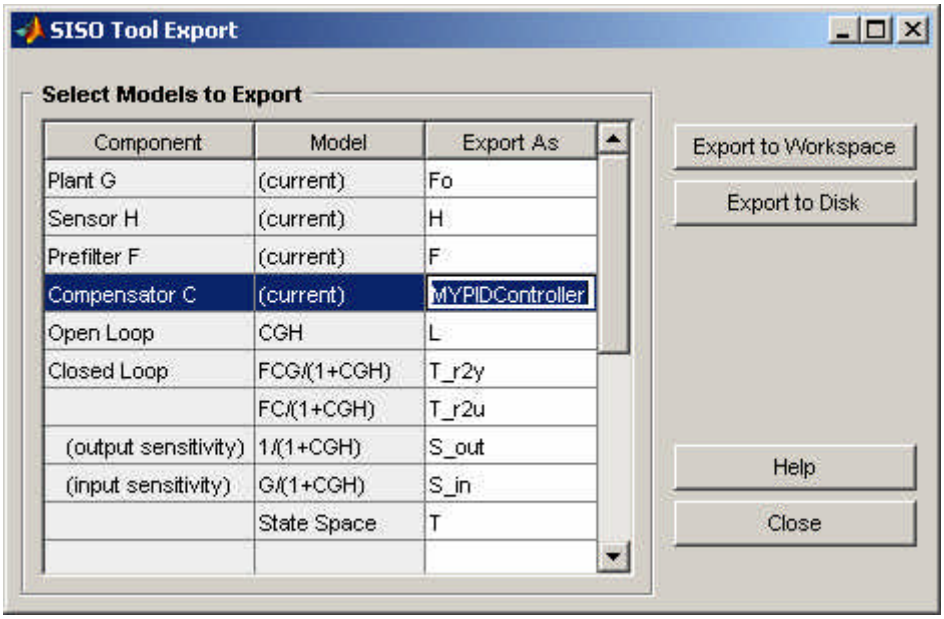
We get e. g. the closed loop step response.




```
This controller is exported to the workspace:
```



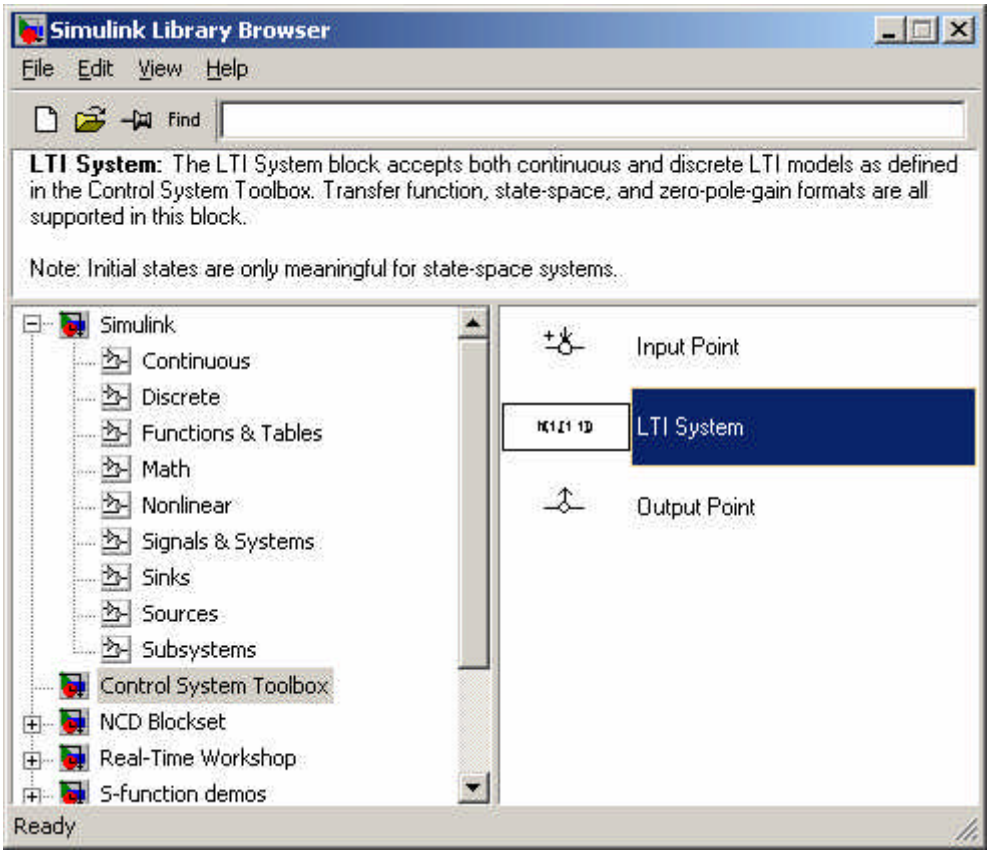
Inside the export dialog the component must be selected and the name may be changed to MYPIDController.



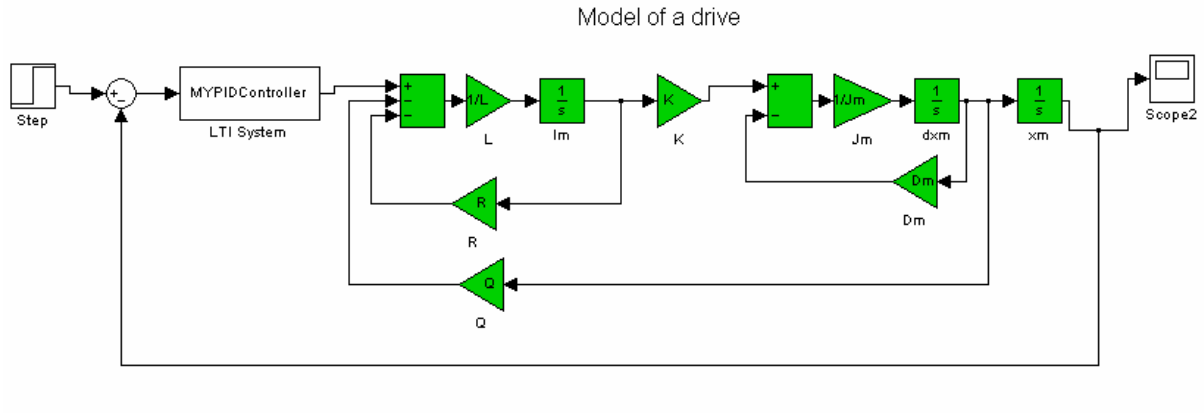
Now the transfer-function of the contoller is available as a transfer function inside the Matlab workspace.

```
>> MYPIDController

Zero/pole/gain:
1548.8576 (s^2 + 2265s + 1.356e006)
-----------------------------------
          s (s+2.739e004)

>>
```

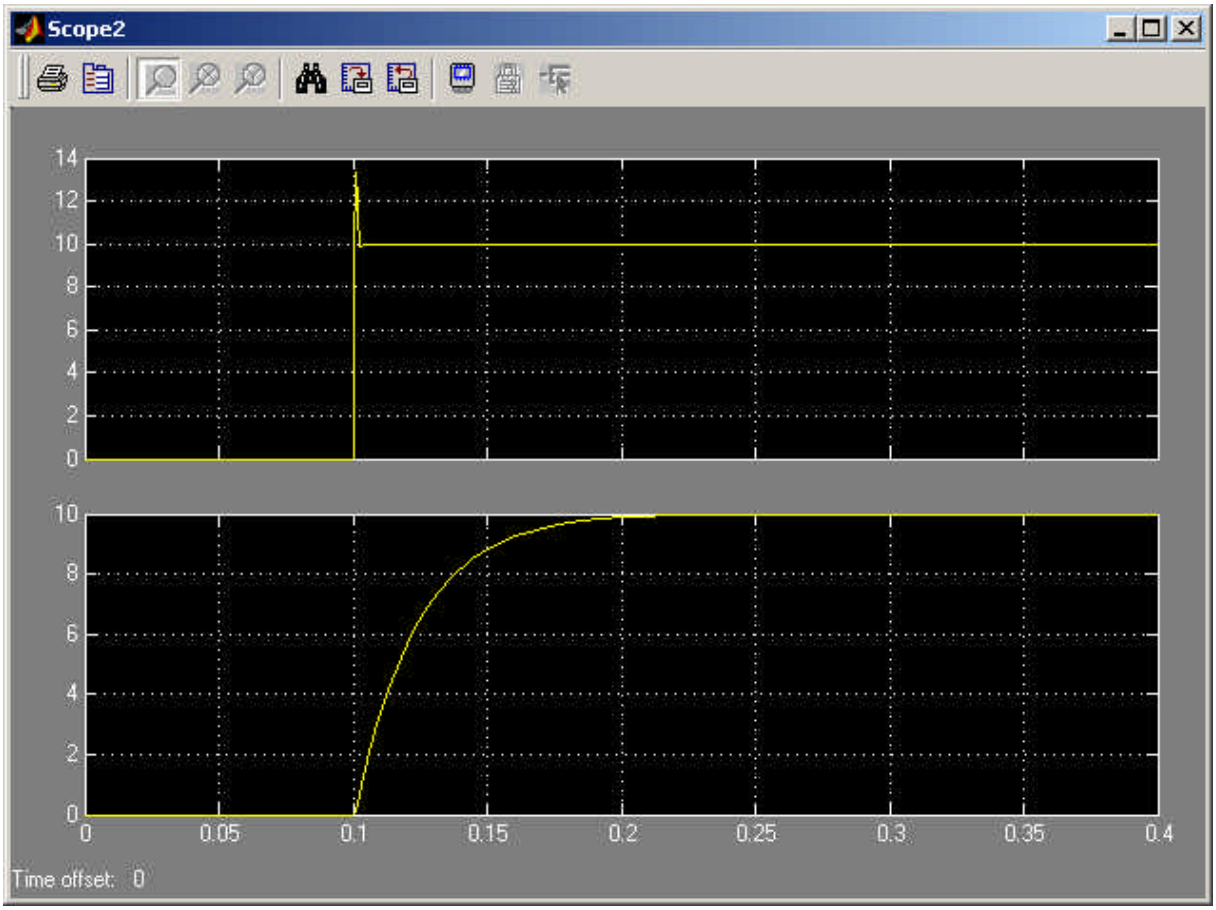
Inside Simulink this controller can directly be used by adding the LTI-System from the Simulink library



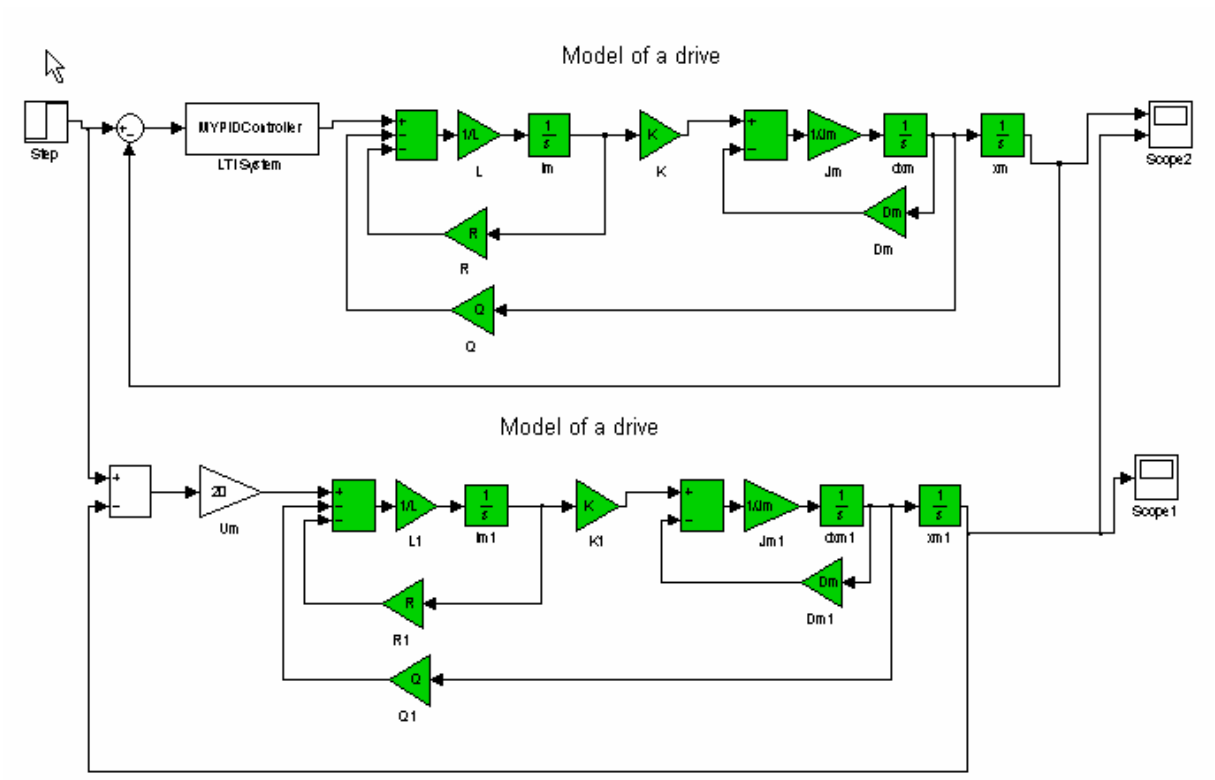
This block is added to the model. The transfer-function of the contoller is used a block parameter.



The model is simulated and afterward compared with the P-Controller. The new controller is much faster when the old one.



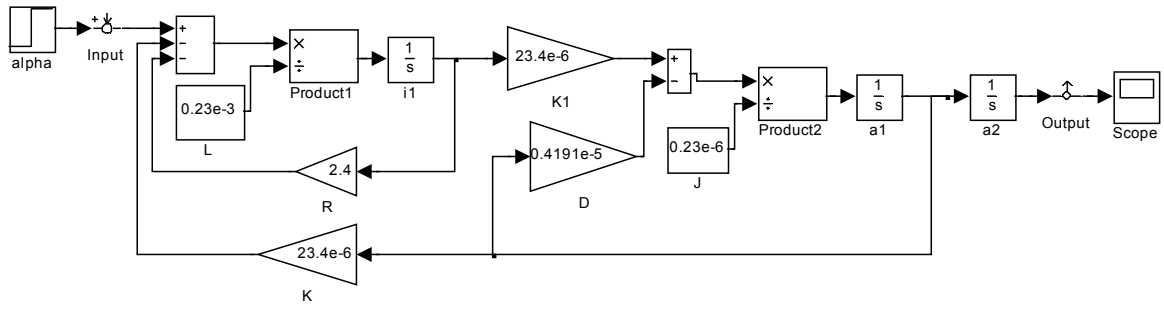
The comparsion is done using this structure:



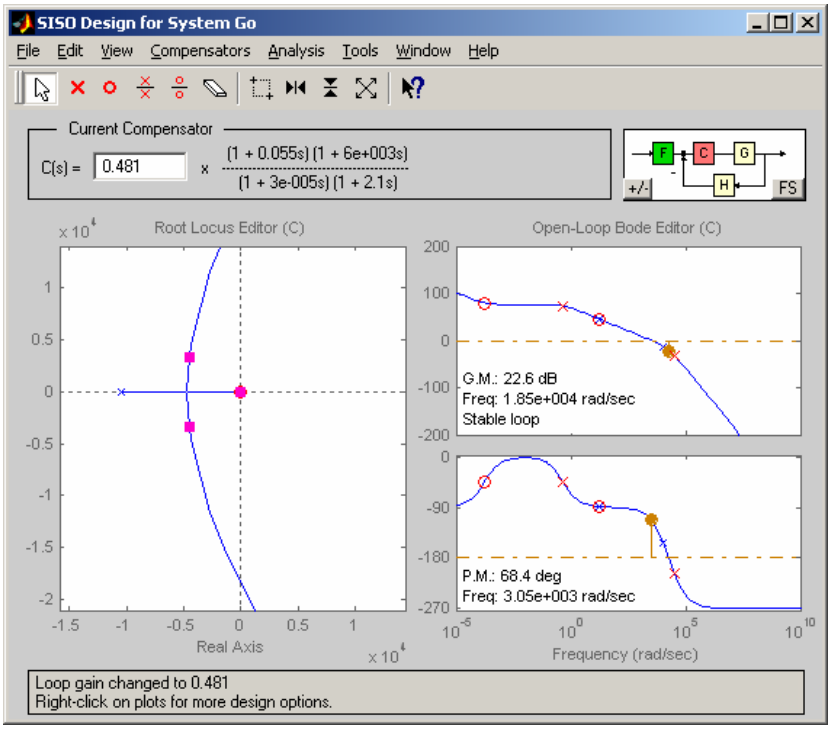
Exercise 7:

Controller design for a position control

a)
- Löschen des vorhanden P- Reglers und der Rückkopplung (offener RK!!)
- Einfügen von Input und Output - Point in das vorhandene Model (aus Control System Toolbox)
- Aufruf des LTI-Viewer (Tools / Linear analyses)
- Aufruf des Linearisierten Models (in LTI-Viewer Simulink / Get Linearized Model)
- Auswählen der Plotfenster (Edit / Plot Configuration)
- Exportieren des Models in Matlab-Workspace (File / Export) Die exportierte Variable ist die Zustandsraumdarstellung
- Umwandlung der Zustandsraumdarstellung in eine Transferfunktion Go=tf(variablenname)
- Aufruf von sisotool(Go)

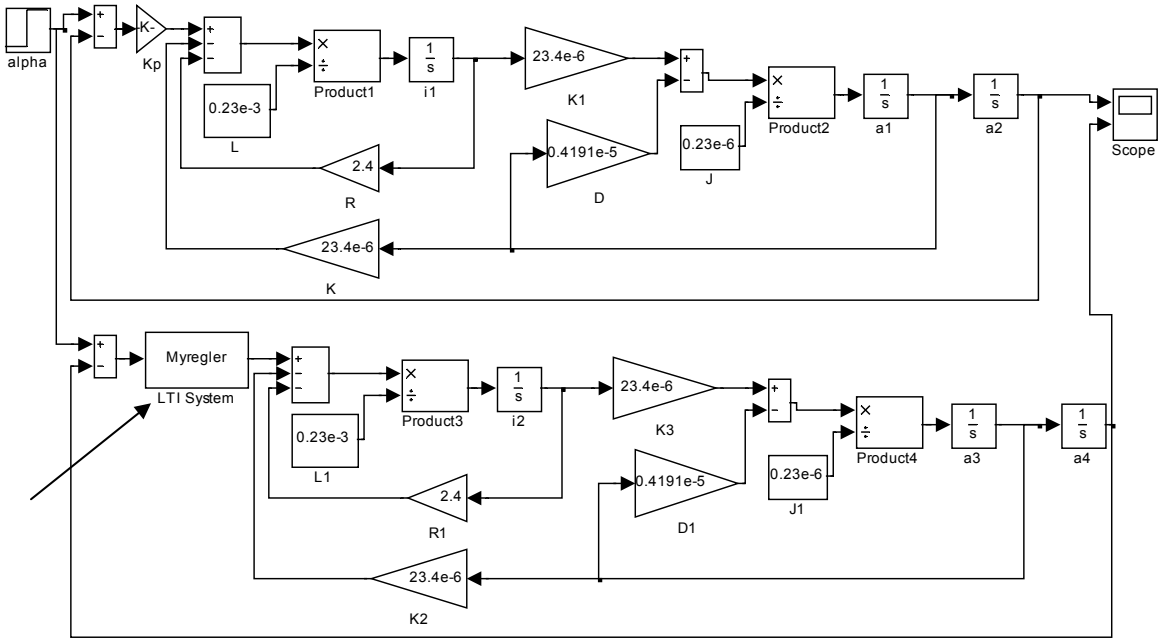


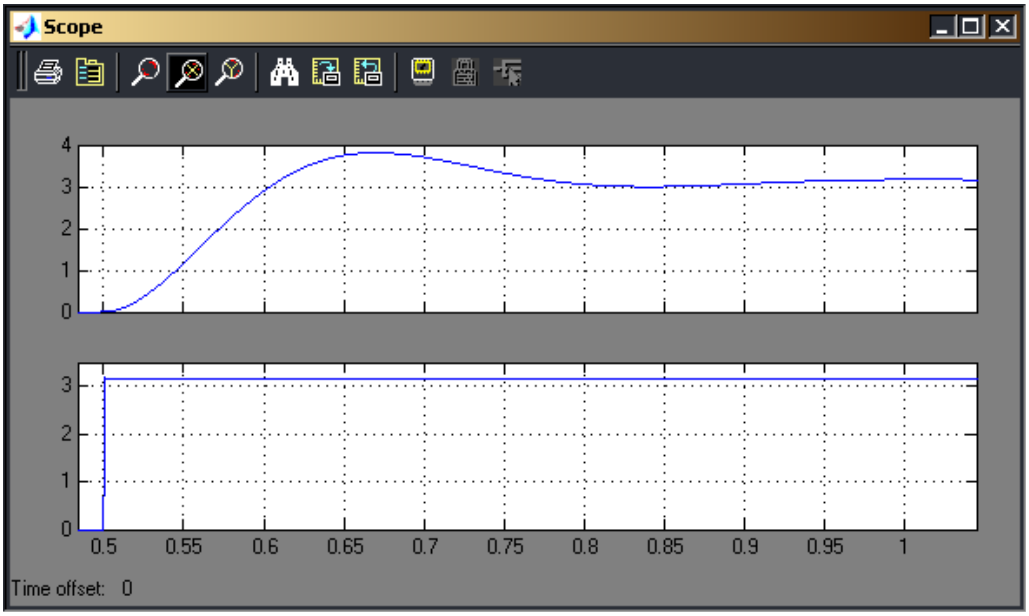
- Reglerentwurf mit Sisotool



- Exportieren des entworfenen Reglers nach Matlab-Workspace (File / Export) Achtung: sinnvollen Namen geben
  Exportierter Regler ist in Transferfunktion
- Einfügen des entworfenen Reglers in Simulink mittels LTI Block

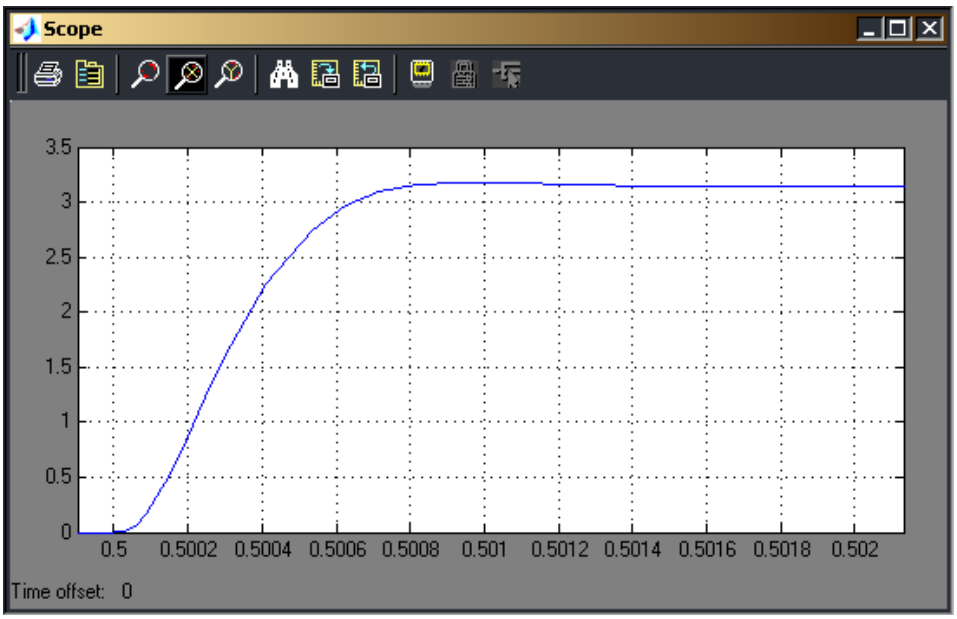
Vergleich des PDI-Reglers mit dem P-Regler:





P-Regler:
Kp=8

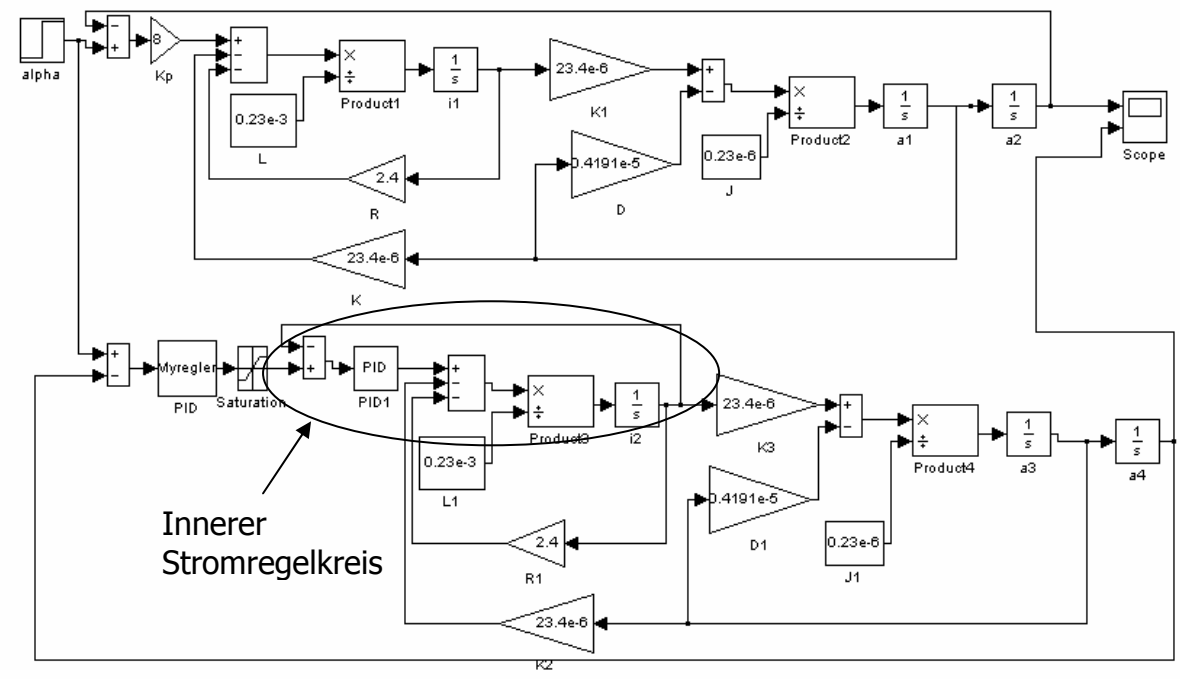
PID-Regler:
Kp=0.481



PID-Regler:
Kp=0.481

b) Entwurf einer Kaskadenregelung

b) Kaskadenreglung



<u>Bedingung für eine Kaskaskadenregelung :</u>

→ Der Innere Regelkreis sollte etwa 10 mal so schnell sein wie der Äußere Regelkreis

ps:

(Der Reglerentwurf wird nach den Prüfungen durchgeführt)