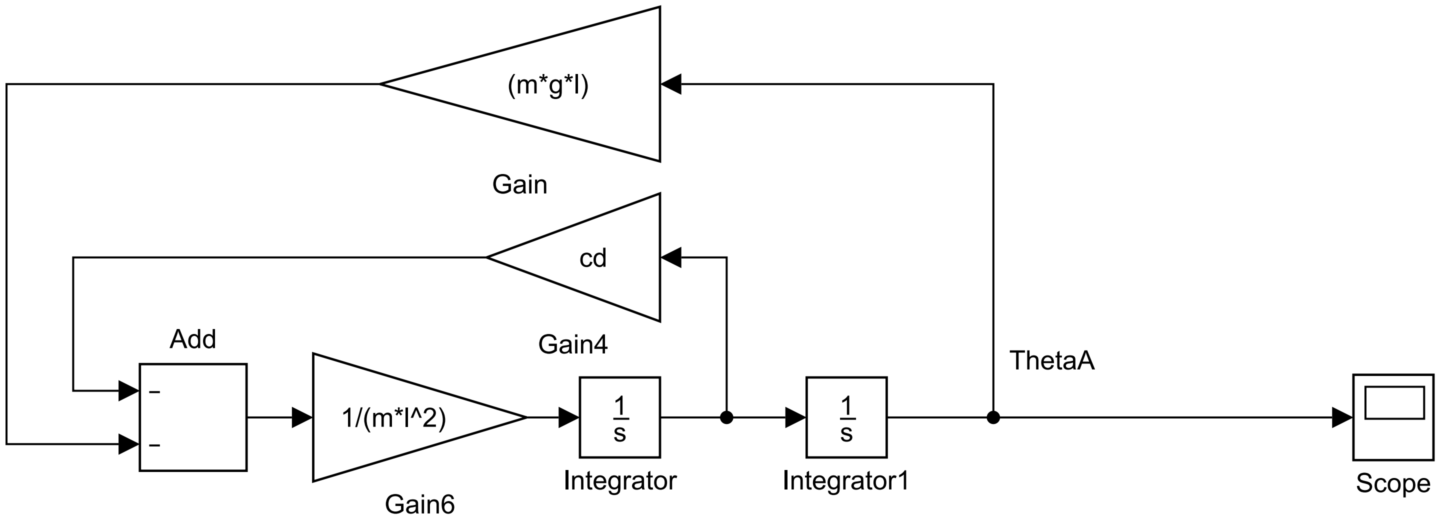
```
% SiSy Lab5 single pendulum  _ HS18 dqtm
%  (C) RUN SIMULINK SIMULATION
% ===============================
clc, clear all, close all;

% Parameters for Model
cd = 0.05;  % Dampfer constant in [N.m/rad.s]
m = 1.5;        % Mass in [kg]
l = 0.7;        % Length Pendel Rod in [m]
g = 9.8;        % Gravitation (acceleration) in [m/s^2]

thetaA_0 = pi/8;    % initial condition

tfinal = 60;   % Simulation time in [s]

% Open & Run Simulink model
open_system('single_pendulum')

% Parameters for Simulink
set_param(0,'solver','ode23',...
            'solvermode','auto',...
            'starttime','0.0',...
            'stoptime','tfinal')

sim('single_pendulum')

%%  (E) Checking numerical solution of Diff Equation

syms s
eqn_s = (m*l^2*s^2) + (cd*s) + (m*g*l) == 0
sol_s = solve(eqn_s,s)

% visualize the result from Matlab
t = 0:1e-1:60;
A = thetaA_0;
theta_t = A/2*( exp(sol_s(1)*t) + exp(sol_s(2)*t) );

figure(1)
plot(t,theta_t);


%% (F) Compare Simulink and Diff-Equation numerical solution

tsimu = Theta_single_simulink.time;
thetaAsimu = Theta_single_simulink.signals.values;

figure(2)
plot(tsimu,thetaAsimu,'k',...
     t,theta_t,'m'),grid
    legend({'Simulink','DiffEqu'})
```


(m*g*l)
Gain
cd
Gain4
Add
1/(m*l^2)
Gain6
1
s
Integrator
1
s
Integrator1
ThetaA
Scope

(d) Another possibility is to solve this differential equation directly in Matlab. We will do this by testing an exponential function as solution. The exponential function is specially adequate, as we will see once it is replaced in the differential equation.

Solution hypothesis: $\theta(t) = A \cdot e^{s \cdot t}$

$\dot{\theta}(t) = s \cdot A \cdot e^{st}$

$\ddot{\theta}(t) = s^2 \cdot A \cdot e^{st}$

Replacing in the differential equation:

$$A \cdot e^{st}\left[ml^2 \cdot s^2 + C_D \cdot s + mgl\right] = 0$$

non-trivial solution (w/ $A \neq 0$)

Resulting algebraic equation:

$$ml^2 s^2 + C_D \cdot s + mgl = 0 \quad \Rightarrow$$

2nd order equation w/ 2 roots

$s_{1,2} = -\delta \pm j\omega$

→ oscillatio

Back to the solution hypothesis:

$$\theta(t) = \frac{A}{2} \cdot \left[e^{s_1 t} + e^{s_2 t}\right] \quad ; \text{ w/ } A = \theta(0)$$

→ envelope exponential decay

(e) Let us calculate numerically in Matlab the solution of the 2nd order algebraic equation. And plot the corresponding solution: $\theta(t)$ time function.

Work in the Matlab script in the section
"`Checking numerical solution of Diff Equation`"

(f) We also want to compare both solution methods (Simulink and Matlab- via algebraic equation). In order to do so, import the solution from Simulink in the Matlab workspace and plot both solutions in the same graphic.

Work in the Matlab script in the section
"`Compare Simulink and Diff-Equation numerical solution`"

(g) What could be the cause for the differences you can observe?
How are the time steps specified in Simulink?
Force a smaller time steps in Simulink and test the comparison once more.
In Simulink the time steps are defined by a maximum value and a accuracy tolerance.

(h) How does this method of the exponential solution (and conversion from a differential equation to an algebraic equation) relates to the representation with Fourier Transformation?
This method is analog to the derivation property of the FT. But by supposing a complex s value, we are also able to consider harmonic responses with decaying or growing amplitude.