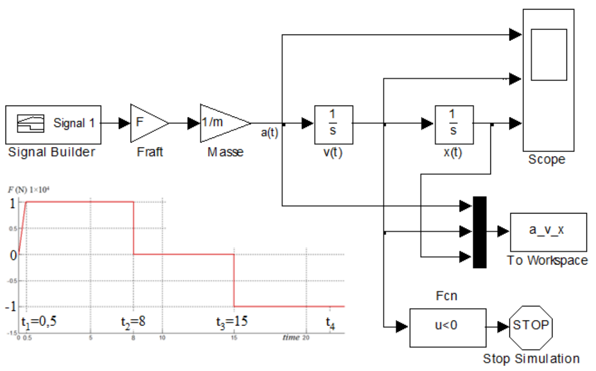
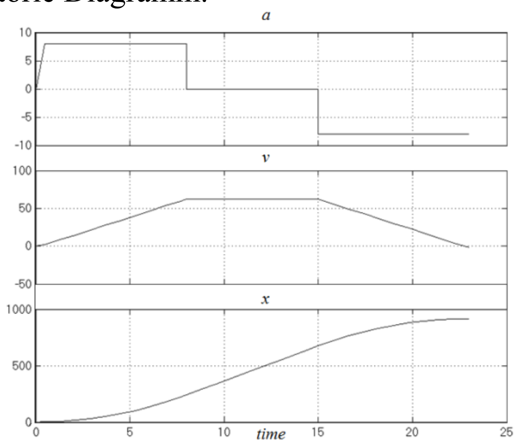
Musterlösungen Übung 6 – Modelbildung

1. Ein m = 1250 kg Fahrzeug wird wie im F(t) – t Diagramm dargestellte Kraft angetrieben.

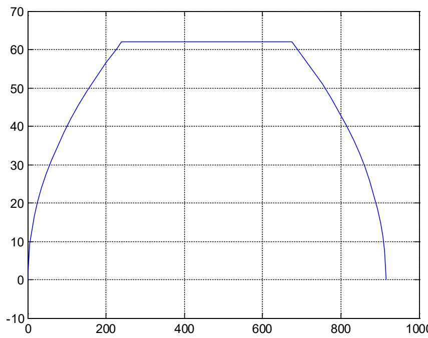
a) Erstellen Sie ein SIMULINK Modell, Simulieren Sie Beschleunigung a(t), Geschwindigkeit v(t), Weg x(t) und x-v Trajektorie Diagramm.

```
>> F = 10000; m= 1250;
```






```
>> plot(a_v_x.Data(:,3),a_v_x.Data(:,2));
```

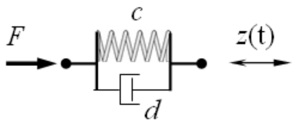


x-v Trajektorie Diagramm

b) Berechnen Sie die gesamt zurückgelegte Zeit t_4 zum Stillstand und Strecke $x_4(t)$.

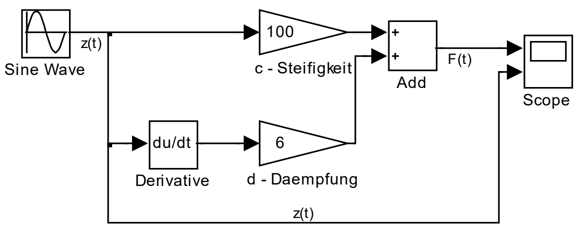
time	$0 \sim t_1$	$t_1 \sim t_2$	$t_2 \sim t_3$	$t_3 \sim t_4$
a(t)	$\frac{F}{m} \cdot \frac{t}{t_1}$	$\frac{F}{m}$	0	$-\frac{F}{m}$
$v(t) = \int_{ti}^{tj} a(t)dt$	$v_1 = \frac{F}{m} \frac{t^2}{2 \cdot t_1}\Big\vert_0^{0,5} = 2$	$v_2 = v_1 + \frac{F}{m} t\Big\vert_{0,5}^{8} = 2 + 8 \cdot 7,5$ $= 62$	$v_3 = v_2 = 62$	$v_4 = v_3 + \frac{F}{m}(-t)\Big\vert_{15}^{t_4} = 62 - 8t_4 + 120 = 0$ $t_4 = 22,75$
$x(t) = \int_{ti}^{tj} v(t)dt$	$x_1 = 8 \cdot \frac{t^3}{6 \cdot t_1}\Big\vert_0^{0,5} = \frac{1}{3}$	$x_2 = \left(v_1 t + 8 \cdot \frac{(t-t_1)^2}{2} \right)\Bigg\vert_{0,5}^{8}$ $= 240,33$	$x_3 = x_2 +$ $v_2(t_3 - t_2)$ $= 674,33$	$x_4 = x_3 + \left(v_3 t - 8 \cdot \frac{(t-t_3)^2}{2} \right)\Bigg\vert_{15}^{22,75}$ $= 914,583$

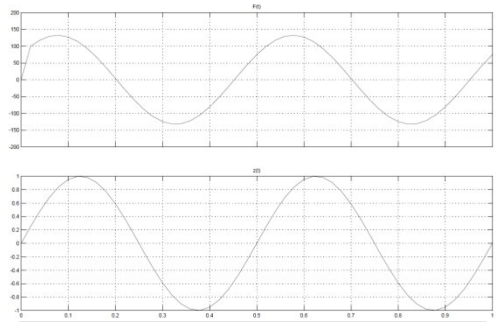
2. Ein Kelvin-Voigt-Modell mit c = 100 N/mm, d = 6 Ns/mm werde durch z(t) = 1 *sin*(2π·2·t) angeregt, die folgende Aufgabe werden simuliert:



a) Zeitverlauf der z(t) und F(t).
b) Hysteresis-Diagramm F vs. z
c) Dynamische Steifigkeit & Verlustwinkel abhängig von Frequenz (Hz). (using MATLAB m-file)

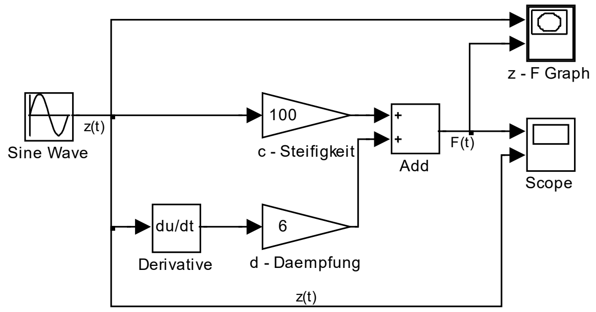
a)

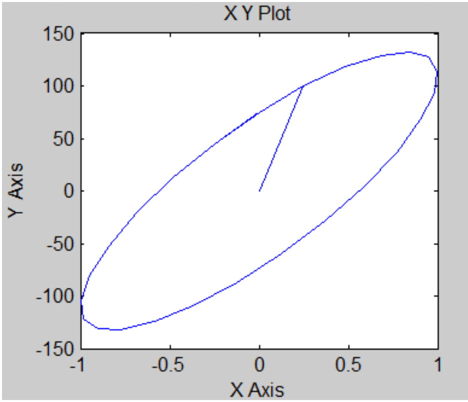




Ue5_KelvinVoigt_1a.mdl

b)



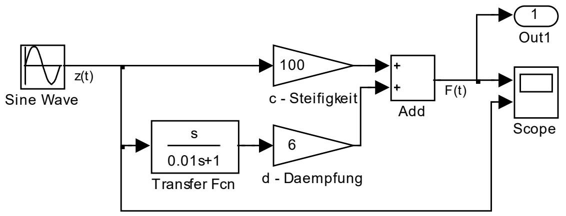


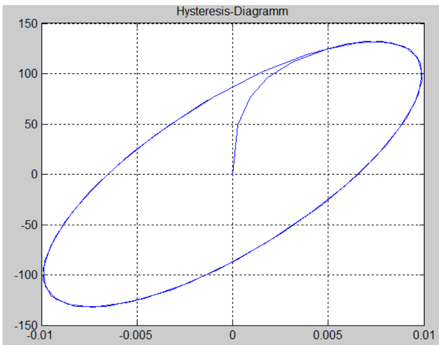
Ue5_KelvinVoigt_1b.mdl

Der Matlab-Befehl sim

Falls die Variablen in einem Script File definiert werden, lässt sich das Simulink-Programm mit dem Befehl **sim** direkt vom Script File aus starten. Es ist es sinnvoll, auch das Plotting in Matlab durchzuführen.

```
>> help sim
>> [t,z,F]=sim('Ue5_KelvinVoigt_1b2')
>> plot(z,F); grid on; title('Hysteresis-Diagramm');
```

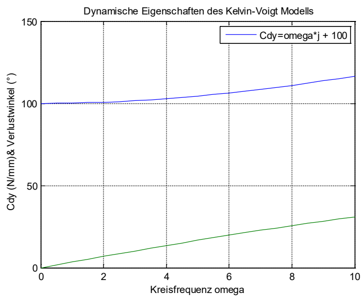




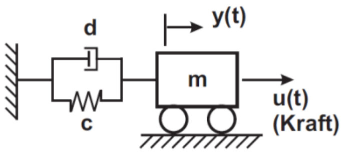
Ue5_KelvinVoigt_1b2.mdl

c) Ue5_KelvinVoigt_1c.m

```
Clear all; clc;
omega=0:0.5:10
p=[6 100];                          % Polynom deklarieren p=6*omega*j+100
Cdy =polyval(p,omega*sqrt(-1)); % Berechnet die Ordinaten von p für alle x-Werte
Amplitude=abs(Cdy);
winkel= angle(Cdy)*180/pi;
plot(omega,Amplitude,omega,winkel)
axis([0 10 0 150]); grid on;
xlabel('Kreisfrequenz omega');
ylabel('Cdy (N/mm)& Verlustwinkel (°)');
legend('Cdy=omega*j + 100');
title('Dynamische Eigenschaften des Kelvin-Voigt Modells');
```



3. Betrachten Sie das Feder-Masse-Dampfer System,
Die Parameter lauten: , m = 100 kg , c = 100 N/mm , d = 200 Ns/m, $u(t) = U_0 \sin(\omega\, t)$
U0 = 1500 N, $\omega = 6\pi$, Die Anfangsbedingungen: y(0) = 0.1m , y´(0) = -0,2m/s



a) Stellen Sie die Bewegungsgleichung des Systems auf, berechnen Sie die ungedämpfte Eigenkreisfrequenz ω_0 und Dämpfungsgrad ξ.

Bewegungsgleichung:

$$m \cdot \ddot{y} + d \cdot \dot{y} + c \cdot y = U_0 \sin(\omega \cdot t)$$

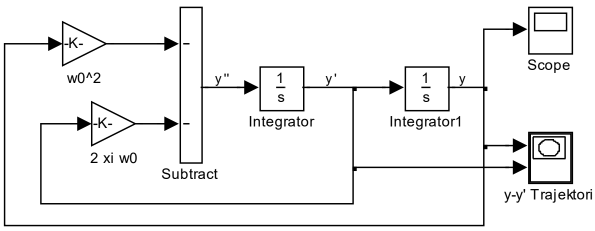
$$\ddot{y} + 2\xi\omega_0 \cdot \dot{y} + \omega_0^2 \cdot y = U_0/m \cdot \sin(\omega \cdot t)$$

mit Parameter:

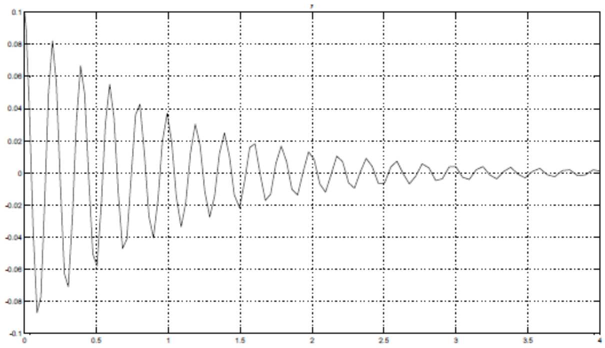
$$\omega_0 = \sqrt{c/m} = \sqrt{100 \times 1000/100} = 31{,}62$$

$$\xi = d/2\sqrt{cm} = 200/(2\sqrt{100 \times 1000 \times 100}) = 0{,}0316$$

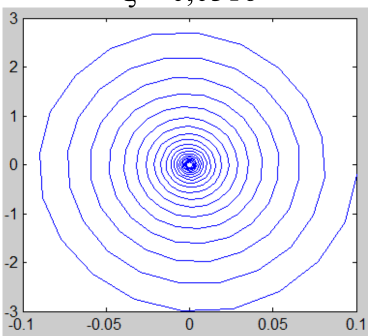
b) Erstellen Sie ein SIMULINK Modell, Simulieren Sie den Zeitverlauf y(t) der homogenen Gleichung mit Anfangsbedingungen und y'-y Trajektorie Diagramm mit $\xi < 0$, $\xi = 0$ und $0 < \xi < 1$.



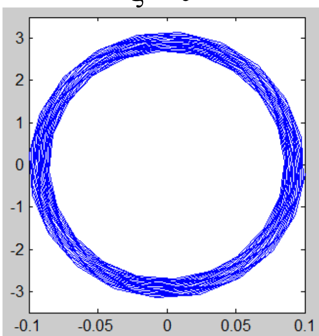
Ue6_FreiSchwingung_3b.mdl



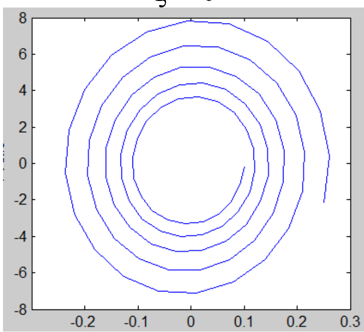
$\xi = 0{,}0316$



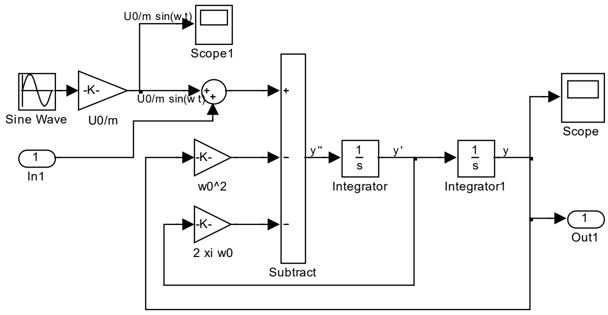
$\xi = 0$



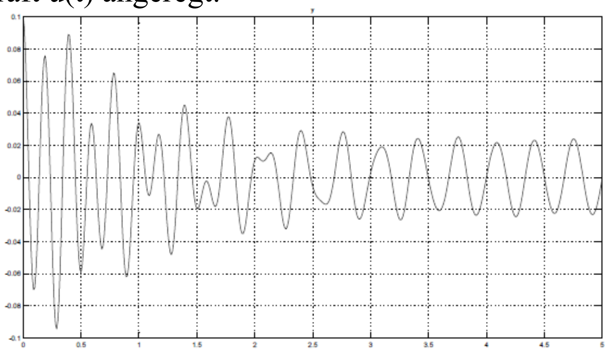
$\xi < 0$



c) Simulieren Sie den Zeitverlauf y(t) der inhomogenen Gleichung mit Anfangsbedingungen, Zurzeit t = 0 werde das System durch die Kraft u(t) angeregt.

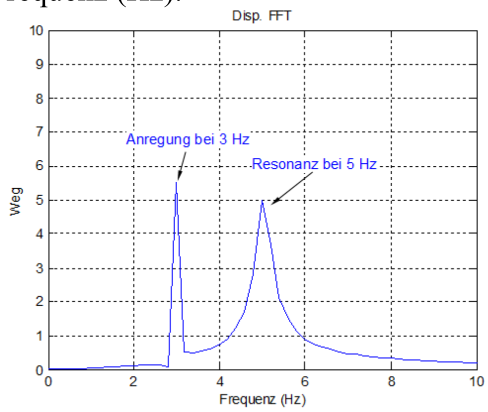


Ue6_ErzwungSchwingung_3c.mdl



d) Erzeugen Sie eine Graphik der FFT abhängig von Frequenz (Hz).

```
% Ue6_FFT_3d.m
[t,out]=sim('Ue6_ErzwungSchwing_3c');
yt=out(:,1);
Yfft=fft(yt);
Ymag=abs(Yfft);
frq=(0:length(Yfft)-1)'*100/length(Yfft);
plot(frq,Ymag),axis([0 10 0 10]);
xlabel('Frequenz (Hz)');
ylabel('Weg');
title('Disp. FFT'); grid on;
```



e) Konvertieren Sie das SIMULINK Modell zu A, B, C, und D Matrizen und berechnen Sie die gedämpfte und ungedämpfte Eigenfrequenz (Hz), sowie Dämpfungsgrad ξ.

```
>>[A,B,C,D]=linmod('Ue6_ErzwungSchwing_3c')
A =
         0    1.0000
 -999.8244   -1.9996
B =
     0
     1
C =
     1     0
D =
     0
>>eig(A)
ans =
  -0.9998 +31.6042i
  -0.9998 -31.6042i
```

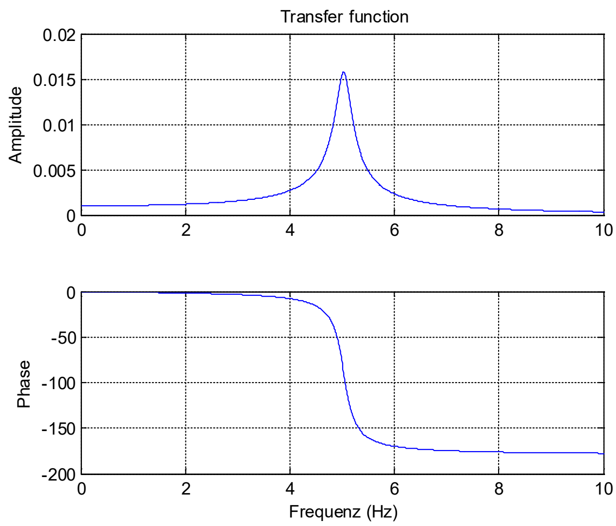
```
>> eig(A)/2/pi
ans =
  -0.1591 + 5.0300i
  -0.1591 - 5.0300i
>> abs(-0.1591 + 5.0300i)
ans =
    5.0325
>> 0.1594/5.0325
ans =
    0.0317
```

f) Erzeugen Sie eine Graphik der Übertragungsfunktion G(*f*) abhängig von Frequenz (Hz).

```
% Ue6_Transferfun_3f.m
[A,B,C,D]=linmod('Ue6_ErzwungSchwing_3c');
x=0:0.01:10; % Frequenz Hz
[fq,Gf]=uef(A,B,C,D,x);
subplot(2,1,1);
plot(fq,abs(Gf)); grid on;
ylabel('Amplitude');
title('Transfer function');
subplot(2,1,2);
plot(fq,angle(Gf)*180/pi);grid on;
xlabel('Frequenz (Hz)');
ylabel('Phase');
```

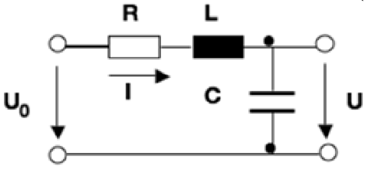
$$G(f)=\frac{Y(f)}{U(f)}=C\,((2\pi f)\,j\,I-A)^{-1}B\,;$$

```
% uef.m
function[fq,Gfj]= uef(A,B,C,D,f)
Aa=A; [Na,Ma]=size(Aa);
Bb=B; [Nb,Mb]=size(Bb);
Cc=C; [Nc,Mc]=size(Cc);
Dd=D; [Nd,Md]=size(Dd);
I=eye(Na); [Nf,Mf]=size(f);
  for n=1:Mf
         fq(n)=f(n);
         fjI_A=(2*pi*f(n)*sqrt(-1))*I-Aa;
         Gfj(n)=Cc*inv(fjI_A)*Bb + D;
  end
```



4. Nachstehend ist der Schaltplan und Differentialgleichung eines RLC- Gliedes abgebildet.

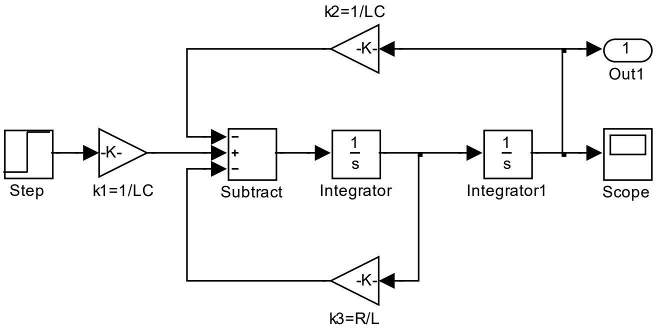
a) Simulieren Sie den Zeitverlauf U(t) mit einer sprungförmigen Spannung u(t) = 10 Volt.



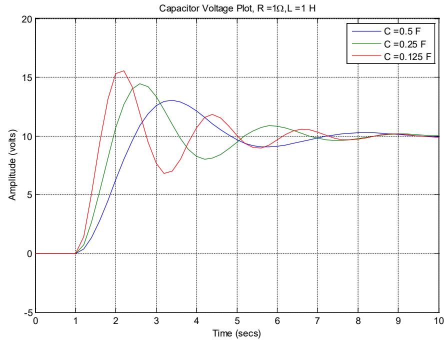
$$LC \cdot \ddot{U}(t) + RC \cdot \dot{U}(t) + U(t) = U_0(t)$$

R = 1 W; L = 1 H; C = 0,25F

a)



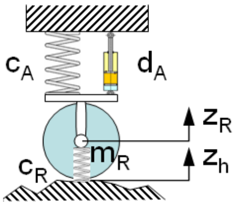
b)



b) Erzeugen Sie eine M- Datei, mit den die Zeitverlauf abhängige Parameter C = 0,5F, C = 0,25; C = 0,125 simulieren

```
%' M-File skript zu plot simulation Daten '
R = 1; % set R value;
L = 1; % set L
Cp = [1/2 1/4 1/8]; % set simulation Werte
für C
for k = 1:3
C = Cp(k); % Kapazitor Werte für simulation
sim('Ue5_RLC_3a'); % run simulation
vc(k,:) = yout; % save Kapaiitor Voltage Daten
end
plot(tout,vc(1,:),tout,vc(2,:),tout,vc(3,:)); % Plot Daten
xlabel('Time (sec)'); grid;
ylabel('Amplitude (volts)');
st1 = 'C = ';
st2 = num2str(Cp(1));
st3 = ' F';
sl1 = strcat(st1,st2,st3); % legend 1
st2 = num2str(Cp(2));
sl2 = strcat(st1,st2,st3); % legend 2
st2 = num2str(Cp(3));
sl3 = strcat(st1,st2,st3); % legend 3
st1 ='Capacitor Voltage Plot, R = ';
st2 = num2str(R); st3 = '\Omega';
st4 = ',L = '; st5 = num2str(L);
st6 = ' H';
stitle = strcat(st1,st2,st3,st4,st5,st6);
title(stitle)
legend(sl1,sl2,sl3); % plot legend
axis([0 10 -5 20]); % set axis für Voltage range -5 bis 20
```

5. Das Bild zeigt eines Rad Modell mit Radaufhängung



DGL:

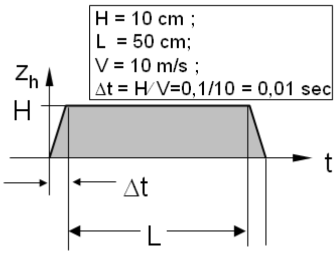
$$m_R \cdot \ddot{z}_R + d_A \cdot \dot{z}_R + (c_A + c_R) \cdot z_R = c_R \cdot z_h$$

m_R = 31 kg

c_A = 20,2 N/mm

c_R = 128 N/mm

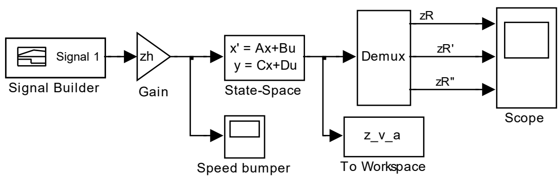
d_A = 0,4 Ns/mm

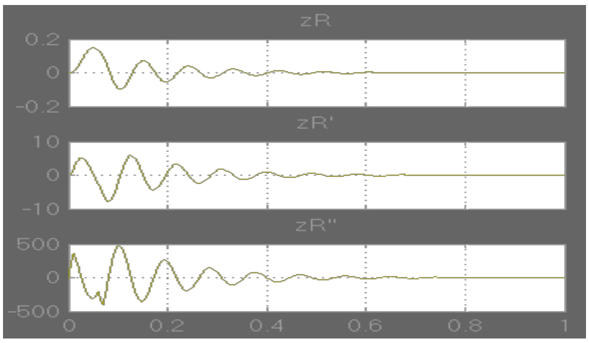


a) Berechen Sie die ungedämpfte und gedämpfte Eigenfrequenz des Rades f_R , f_{Rd}. Schreiben Sie die Zustandsmatrizen A, B, C und D.

```
%Simulation Rad mit Radaufhängung using Matlab/simulink
Clear all; clc;
mR=31;              % Radmasse kg
cA=20.2*1000;   % Aufbaufedersteifigkeit N/m
cR=128*1000;    % Radsteifigkeit N/m
dA=0.4*1000;    % Dämpfungskoeffizient Ns/m
zh=0.1;              % Height of the Speed bumper m
omegaR=sqrt((cA+cR)/mR);  % Radeigenkreisfrequenz
fR=omegaR/(2*pi);        % Eigenfrequenz Hz
xiR=dA/(2*mR*omegaR);  % Dämpfungsgrad
omegaRd = omegaR*sqrt(1-xiR^2);
fRd=omegaRd/(2*pi);   % Eigenfrequenz Hz
%----- Zustandsmatrizen ----------
omegaRR=sqrt(cR/mR);     %
A = [0 1; -omegaR^2   -2*xiR*omegaR] ;
B = [0; omegaRR^2];
C = [1 0; 0 1; -omegaR^2   -2*xiR*omegaR];
D = [0 ;  0;  omegaRR^2];
```

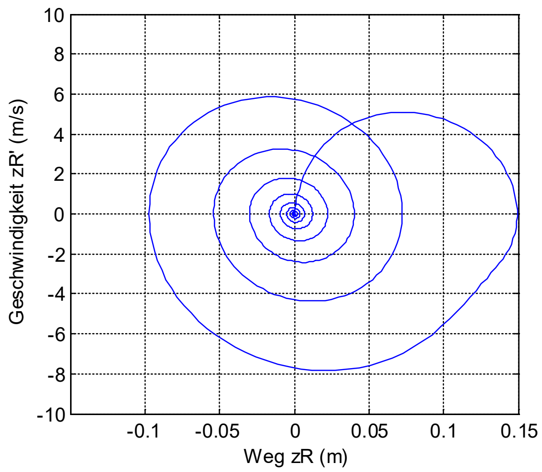
b) Erstellen ein SIMULINK Modell, Simulation die Antworten des Radweges, der Radgeschwindigkeit und Radbeschleunigung für gegebene Straße.





z_R'-z_R Trajektorie Diagramm

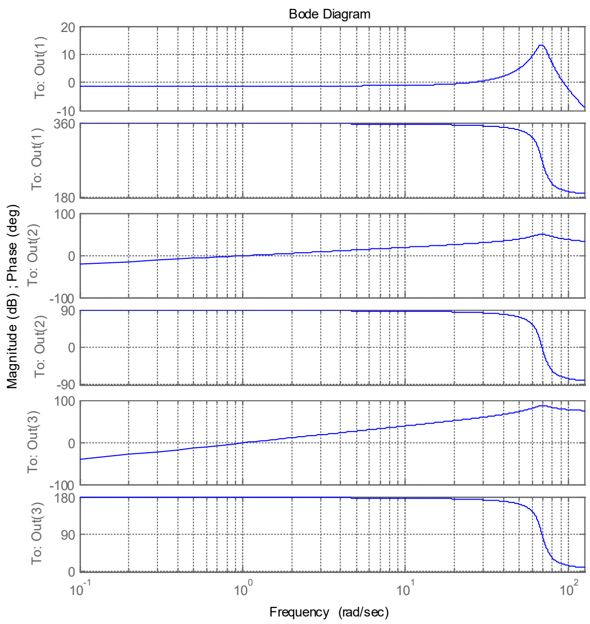
```
>> plot(z_v_a(:,1),z_v_a(:,2)); grid on;
```

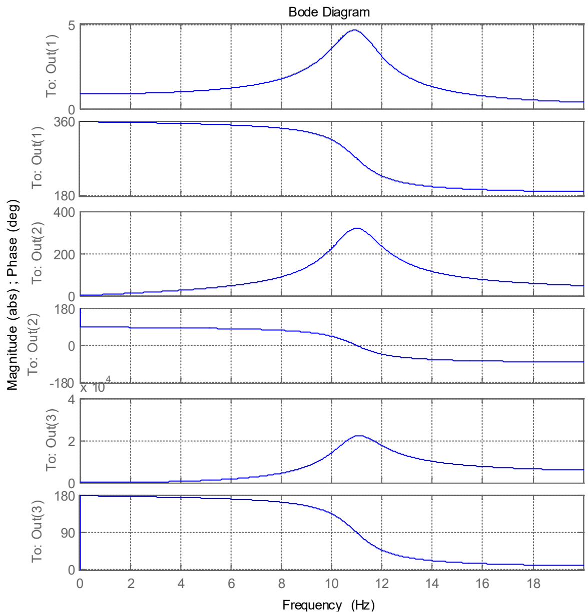


c) Plottern Sie die Übertragungsfunktionen: G(f)=Z_R/Z_h, G(f)=Z'_R/Z_h und G(f) = Z''_R/Z_h

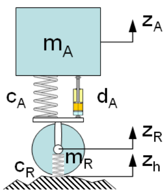
```
% Plot der Übertragungsfunktion
>> [Z,N]=ss2tf(A,B,C,D);

>> Z =
  1.0e+003 *
         0    -0.0000    4.1290
         0     4.1290   -0.0000
    4.1290     0.0000    0.0000
>> N =
  1.0e+003 *
    0.0010    0.0129    4.7806
>> fhz=0:0.1:20*2*Pi; % input frequenz
>> bode(Z,N,fhz);grid; % Bode Diagramm
```

You can use the Units pane to change units in your response plot. The contents of this pane depend on the response plot associated with the editor.





6. Das skizzierte Viertelfahrzeugmodell bewegt sich über eine unebene Fahrbahn



DGL:

$$m_A \cdot \ddot{z}_A = -d_A(\dot{z}_A - \dot{z}_R) - c_A(z_A - z_R)$$

$$m_R \cdot \ddot{z}_R = d_A(\dot{z}_A - \dot{z}_R) + c_A(z_A - z_R) + c_R(z_h - z_R)$$

$m_R = 256$ kg
$m_R = 31$ kg
$c_A = 20{,}2$ N/mm
$c_R = 128$ N/mm
$d_A = 0{,}4$ Ns/mm

a) Erstellen ein SIMULINK Modell, Ausgänge: der relative Weg zwischen Aufbau und Rad, und die Aufbaubeschleunigung. Straßeprofil:
(I). Signal Builder wie Ü6.5.
(II) Chrip Signal: Initial frequency (Hz), 0.01; Target time (secs), 20; frequency at target time (Hz), 20

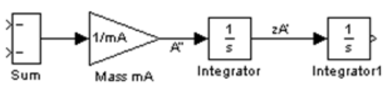
Schnitt 1:

```
%m-file Viertelkfzdaten.m
%-----------------
mA=265;        % Aufbaumasse(kg)
mR=31;         % Radmasse   (kg)
cA=20.2*1000 % Aufbausteifigkeit(N/m)
dA=0.4*1000  % Aufbaudaempfung(Ns/m)
cR=128*1000  % Radsteifigkeit(N/m)
Zh=0.1       % Fahrbahnamplitude(m)
```

Schnitt 2:

Diese Gleichungen werden mit Gain Blocks (für $1/m_A$ & $1/m_R$) und 2 sum Blocks repräsentiert. Es gibt 2 Kräfte, die auf die Masse m_A wirken (1 Feder und 1 Dämpfer) und 3 Kräfte, die auf die Masse m_R wirken (1 Aufbaufeder, 1 Aufbaudämpfer und 1 Radfeder)

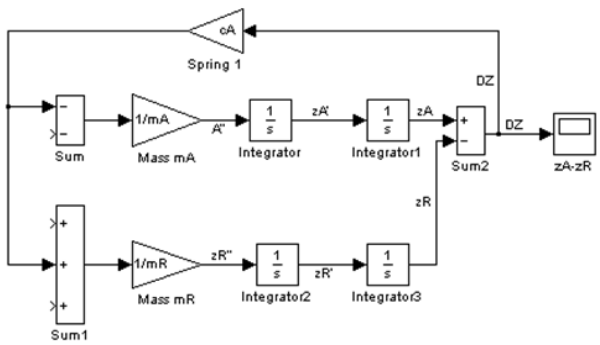
$$\ddot{z}_A = \frac{1}{m_A}\sum F$$



$$\ddot{z}_R = \frac{1}{m_R}\sum F$$

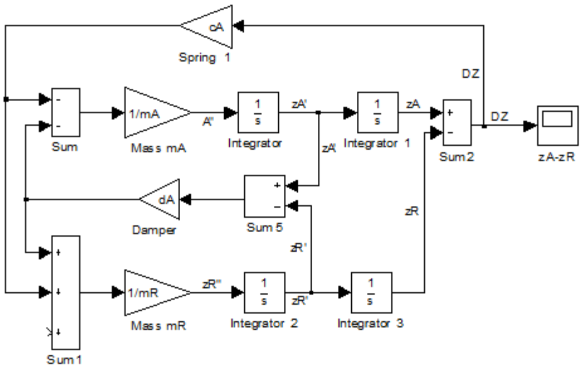
Schnitt 3:

Die Aufbaufederkraft wird auf die Masse m_A addiert. Dies gleich c_A mal die relative Bewegung z_A - z_R.



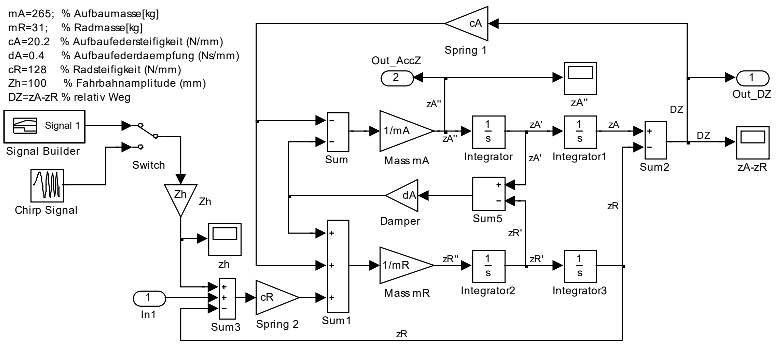
Schnitt 4:

Die Aufbaudämpferkraft wird auf die Masse m_A addiert. Dies gleich d_A mal die relative Geschwindigkeit $\dot{z}_A - \dot{z}_R$



Schnitt 5: (Ue6_Viertelfahrzeug_a.mdl)

Die Radfederkraft wird auf die Masse m_R addiert. Dies gleich c_R multipliziert die relative Bewegung $z_h - z_R$. Gleichzeitig werden die Straßensignale mit manul switch aufgebaut.



Chirp Signal:

Initial frequency (Hz)

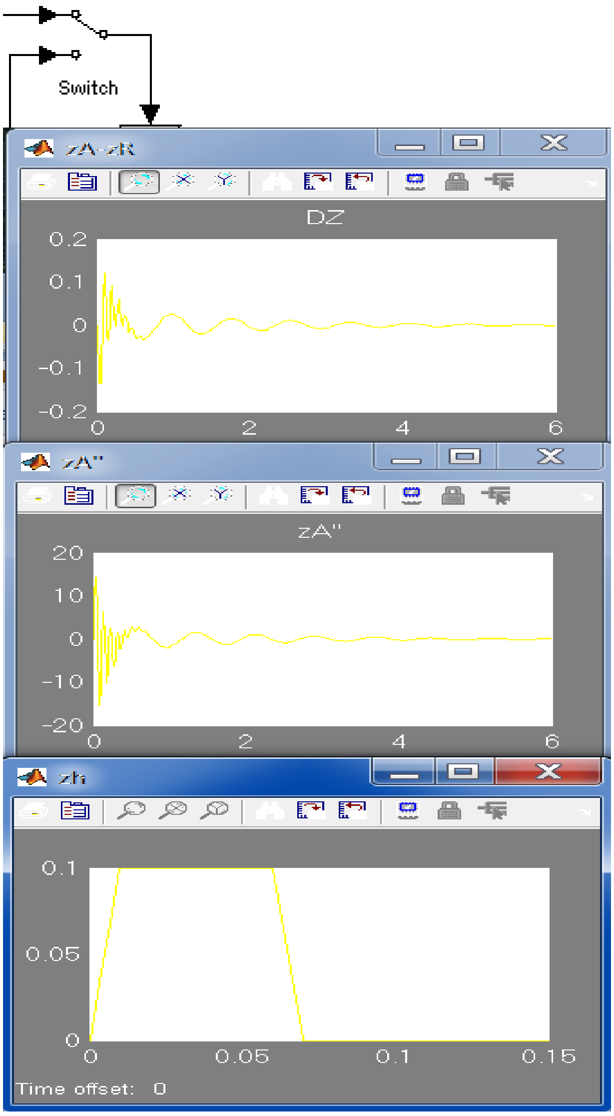
0.01

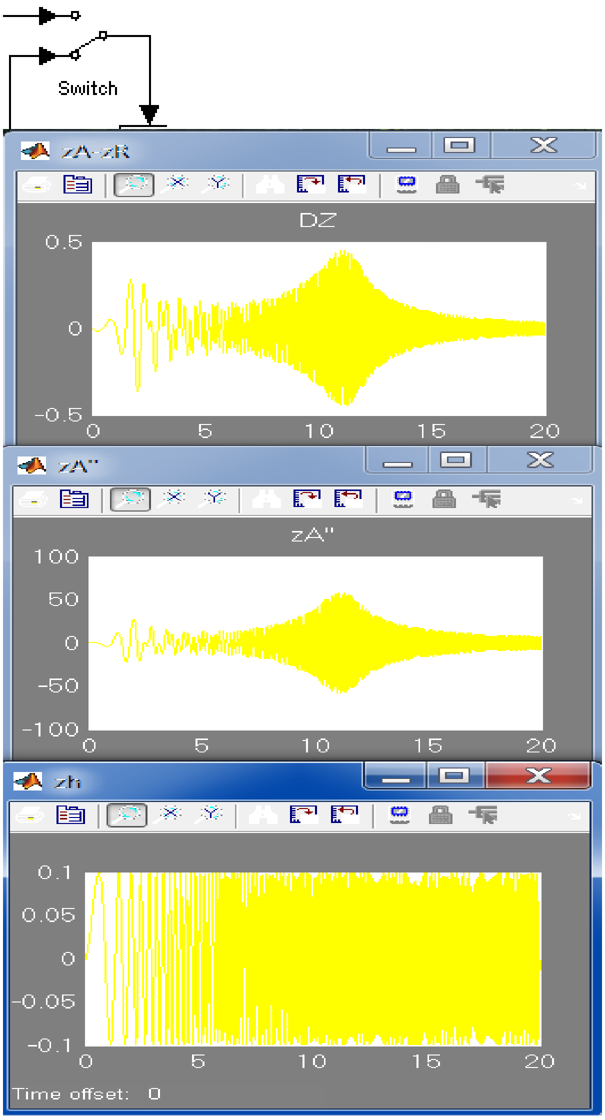
Target time (secs)

20

Frequency at target time (Hz)

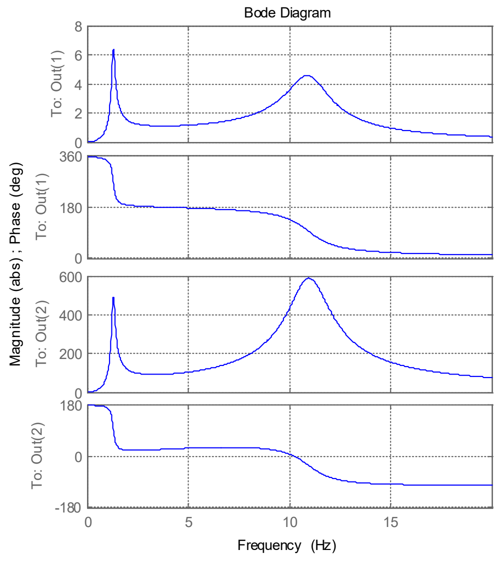
20





b) Berechen die Systemeigenschaften: ungedämpfte und gedämpfte Eigenfrequenzen, so wie Dämpfungsgrad. (Bearbeitung in m.file)

```
%Ue6_b
[A,B,C,D]=linmod('Ue6_Viertelfahrzeug_a');
Eigenfrqcplx=eigs(A)/(2*pi);
ungedampfq=abs(Eigenfrqcplx);
gedampfq=imag(Eigenfrqcplx);
dampgrad=abs(real(Eigenfrqcplx)./ungedampfq);
```



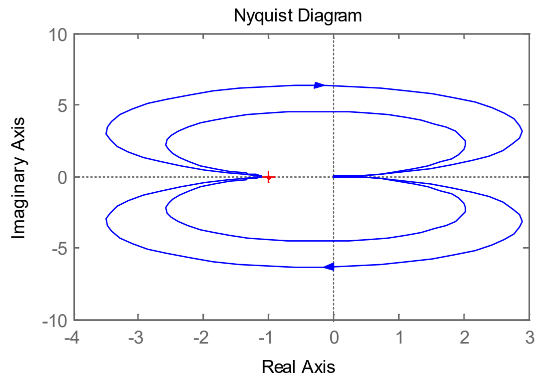
c) Plottern Sie die Übertragungsfunktionen:
$G(f)=DZ/Z_h$ und $G(f) = Z''_A/Z_h$

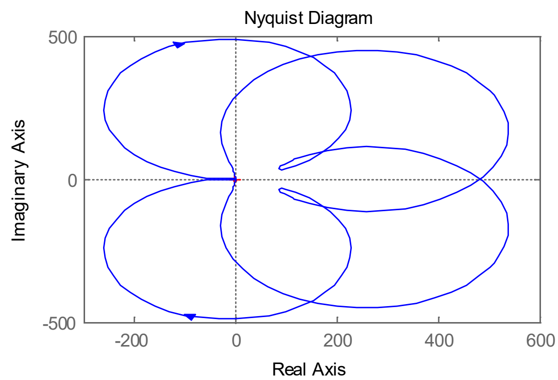
```
% Ue6_c: Plot der Übertragungsfunktion
[Z,N]=ss2tf(A,B,C,D);
fhz=0:0.1:20*2*Pi;  % input frequenz
bode(Z,N,fhz);grid; % Bode Diagramm
```

d) Stellen Sie die Bewegungsgleichungen in die 1. DGL Form der Zustandraumgleichung um, mit dieser Gleichung erstellen Sie ein Simulink-Modell. Vergleichen Sie die Ergebnisse vom a) erzeugten Modell. Plottern Sie die nyquist-Diagramme der Übertragungsfunktionen:
$G(f)=DZ_R/Z_h$ und $G(f) = Z''_A/Z_h$

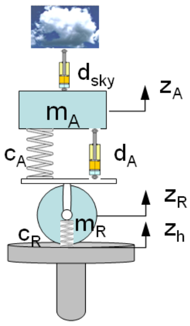
Zustandsgrößen: $\begin{bmatrix} z_A & z_R & \dot{z}_A & \dot{z}_R \end{bmatrix}^T = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T$

```
% Ue6_d: Zustandsmatrizen
A=[0 0 1 0;
   0 0 0 1;
  -cA/mA cA/mA -dA/mA dA/mA;
   cA/mR -(cA+cR)/mR dA/mR -dA/mR];
B=[0; 0; 0; cR/mR];
C=[1 -1 0 0;-cA/mA cA/mA -dA/mA dA/mA];
D=[0; 0];
[Z,N]=ss2tf(A,B,C,D);
subplot(1,2,1);
G=tf(Z(1,:),N); nyquist(G);
subplot(1,2,2);
G=tf(Z(2,:),N); nyquist(G);
```





7. Das Viertelfahrzeugmodell mit Skyhook Controller bewegt sich über ein Stempel eines Straßensimulators



DGL:

$$m_A \cdot \ddot{z}_A = -F_d - c_A(z_A - z_R)$$

$$m_R \cdot \ddot{z}_R = F_d + c_A(z_A - z_R) + c_R(z_h - z_R)$$

$$F_d = d_A(\dot{z}_A - \dot{z}_R) + d_{sky}\dot{z}_A$$

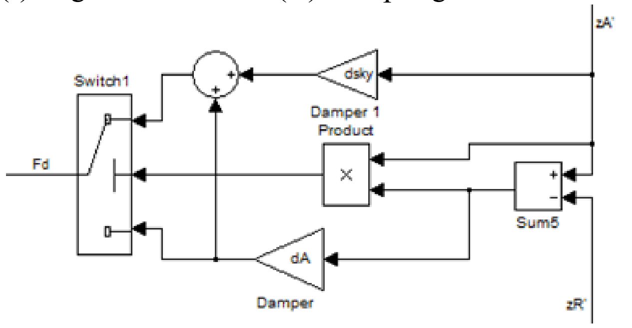
$$F_d = d_A(\dot{z}_A - \dot{z}_R)$$

$m_A = 256$ kg
$m_R = 31$ kg
$c_A = 20{,}2$ N/mm
$c_R = 128$ N/mm
$d_A = 0{,}4$ Ns/mm
$d_{sky} = 4 d_A$

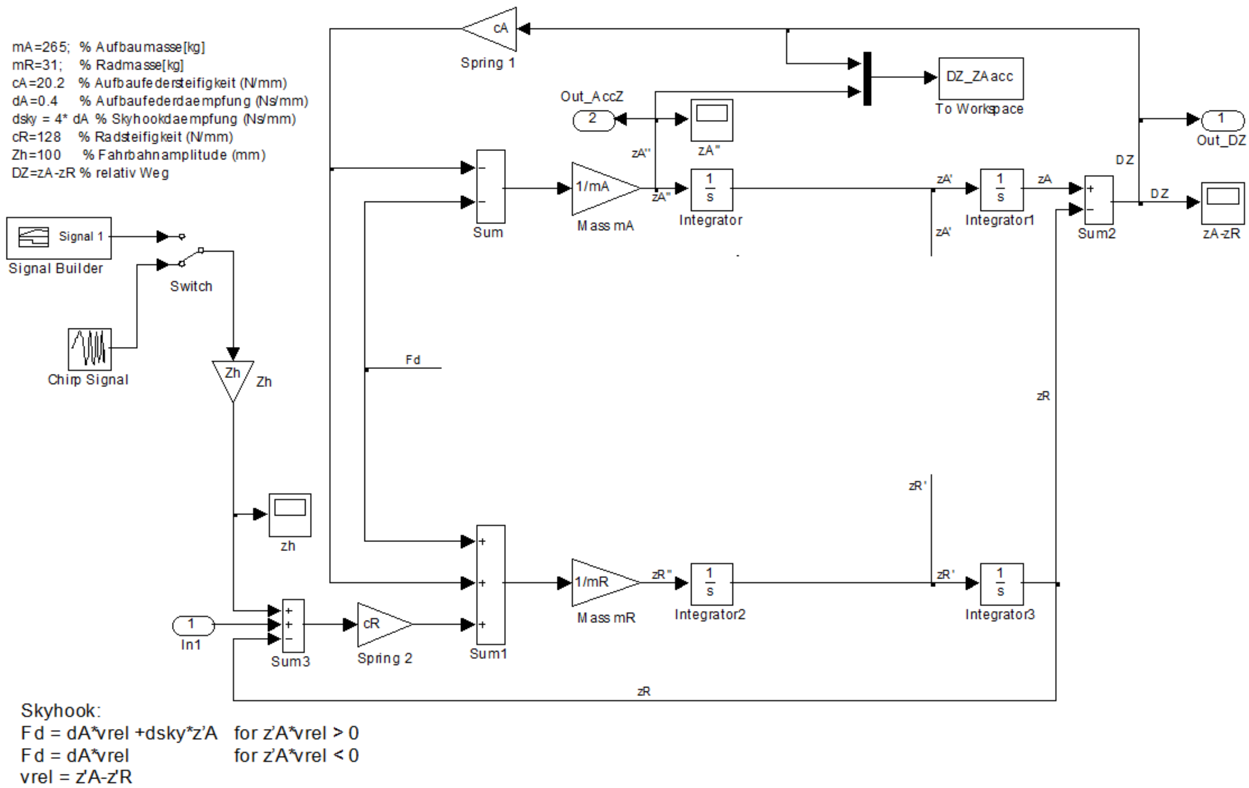
a) Erstellen Sie ein SIMULINK Modell, Ausgänge: der relative Weg zwischen Aufbau und Rad, und die Aufbaubeschleunigung. Straßeprofil (I). Signal Builder und (II) Chrip Signal wie Ü6-6.

Schnitt 1: Erstellen eine m.file wie Übung 6

```
%m-file Ue7_a:
mA=265;      % Aufbaumasse(kg)
mR=31;       % Radmasse    (kg)
cA=20200;  % Aufbausteifigkeit(N/m)
dA=400;     % Aufbaudaempfung(Ns/m)
dsky=4*dA;     % Skydaempfung(Ns/m)
cR=128*1000;   % Radsteifigkeit(N/m)
Zh=0.1        % Fahrbahnamplitude(m)
```

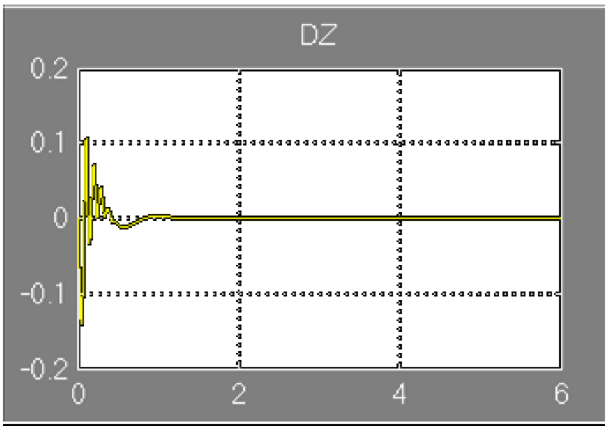


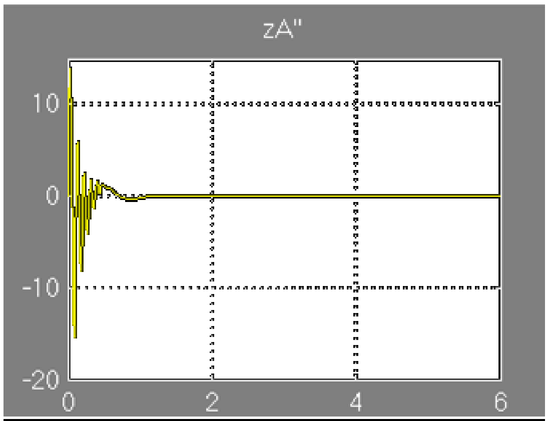
Schnitt 2: Das Modell in Übung6 wird umgebaut.



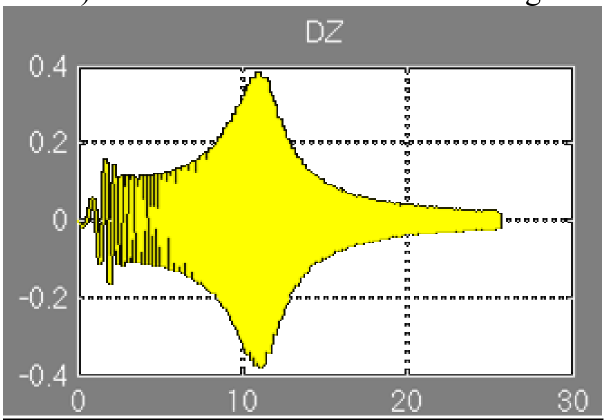
Schnitt 3: Ein Skyhook Modell wird erstellt.

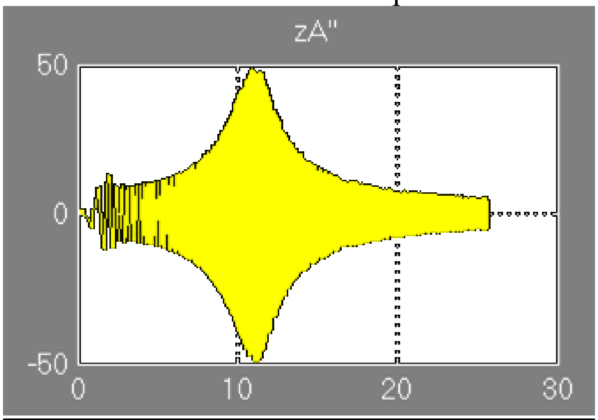
Schnitt 4: Ein Skyhook Modell wird ins Hautmodell eingebaut.
Schnitt 5: I) Simulieren Sie das Modell mit Signal 1: Run Time 6 sec mit fixed step size 0.01





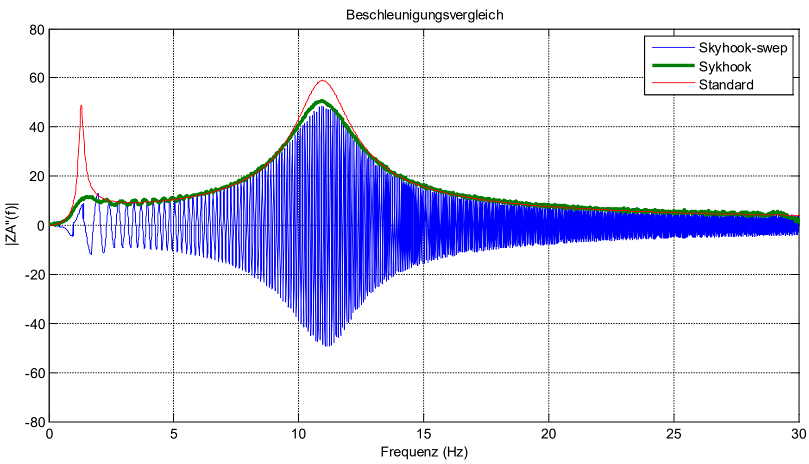
II) Simulieren Sie das Modell mit Signal 2: Run Time.25.6 sec mit fixed step size 0.01



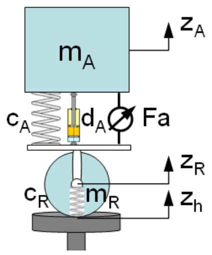


b) Vergleichen Sie die Ergebnisse der Aufbaubeschleunigung $|z''_A|$ im Frequenzbereich (Hz) des konventionalen und Skyhook- Dämpfers. (Bearbeitung in m.file)

```
% Ue6_viertelfahrzeugSkyhook_7b.m
% Ue6_7a:
mA=265;           % Aufbaumasse[kg]
mR=31;            % Radmasse    [kg]
cA=20.2*1000;     % Aufbaufedersteifigkeit (N/m)
dA=0.4*1000;      % Aufbaudaempfung (Ns/m)
dsky= 4*dA;
cR=128*1000;      % Radsteifigkeit (N/m)
Zh=0.1;           % Fahrbahnamplitude (m)
% Ue6_7b:
T = 0.01;         % Sample time = Fixed-step-size Configuration parameter in Simulink
Fs = 1/T;         % Sampling frequency
Tfinal=30;        % Simulationsendzeit
L = Tfinal*Fs;    % Length of signal = gesamte punkte.
t = (0:L-1)*T;    % Time vector = von t=0 bis vorletzte Zeitpunkt: t=(30000-1)*0.01=29.99
% zA"
[zeit,x]=sim('Ue6_ViertelfahrzeugSkyhook_7',[0 Tfinal]);% Zustandsvek.: x1=zA, x2=zR,x3=zA', x4=zR'.
NFFT = 2^nextpow2(L);   % Next power of 2 from length of L
f = Fs/2*linspace(0,1,NFFT/2+1); % Frequency Vector
ZAacc=DZ_ZAacc(:,2);           % Beschleunigung ZA" aus Simulink Modell
ZAaccfft = fft(ZAacc,NFFT)/L; %/L- Das ist eine Normierung auf die Anzahl der Signalwerte und
                              % wird wohl nur zu Anzeigezwecken gemacht
ZAmag=2*abs(ZAaccfft(1:NFFT/2+1))*Tfinal; % Amplitude der ZAacc
% zA" aus Ue6_d => Zustandsmatrizen:
A=[0 0 1 0;
   0 0 0 1;
  -cA/mA cA/mA -dA/mA dA/mA;
   cA/mR -(cA+cR)/mR dA/mR -dA/mR];
B=[0; 0; 0; cR/mR];
C=[1 -1 0 0;-cA/mA cA/mA -dA/mA dA/mA]; D=[0; 0];
[Z,N]=ss2tf(A,B,C,D);
ZAbsys = tf(Z(2,:),N);                 %Man kann auch Bode-Diagramm plottern: "bode(sys)" ;
ZAbsysg = frd(ZAbsys,f,'Units','Hz'); % frd -> Matlab Frequency-Response Data (FRD) Models
frHz=ZAbsysg.Freq;                     % Frequenz in Hz
% Plot single-sided amplitude spectrum.
plot(zeit,ZAacc,f,ZAmag,frHz,Zh*abs(ZAbsysg.Resp(:))); grid on; axis([0 30 -80 80])
title('Beschleunigungsvergleich);
xlabel('Frequenz (Hz)'); ylabel('|ZA"(f)|'); legend('Skyhook-swep','Sykhook','Standard');
```



8. Das Viertelfahrzeugmodell mit PID Controller bewegt sich über ein Stempel eines Straßensimulators



DGL:

$$m_A \cdot \ddot{z}_A = -d_A(\dot{z}_A - \dot{z}_R) - c_A(z_A - z_R) + Fa$$

$$m_R \cdot \ddot{z}_R = d_A(\dot{z}_A - \dot{z}_R) + c_A(z_A - z_R) + c_R(z_h - z_R) - Fa$$

m_A = 256 kg	c_R = 128 N/mm	K_d=7429,5;
m_R = 31 kg	d_A = 0,4 Ns/mm	K_p=29717,8;
c_A = 20,2 N/mm	Z_h=0.1; m	K_i=22288,4

Erstellen Sie ein SIMULINK Modell mit Subsystem, Ausgänge: der relative Weg zwischen Aufbau und Rad, und Kontrolle Kraft Fa. Straßenprofil (I). Signal Builder und (II) Chrip Signal wie Ü6-7.

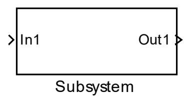
Simulieren Sie das Modell 5 s mit open /closed loop für Straßenprofil (I),

Simulieren Sie das Modell 30 s mit open /closed loop für Straßenprofil (II),

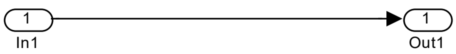
Fixed-step size 1/128.

Schnitt 1:

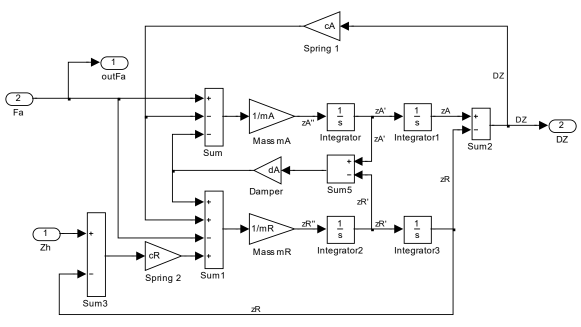
Create a new model window and Drag a Subsystem block →



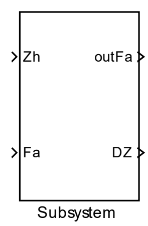
→Double click →



→Open Model 'Ue6_viertelfahrzeug_6a.mdl'→ Copy the model in the Subsystem → Modify the model like left figure:

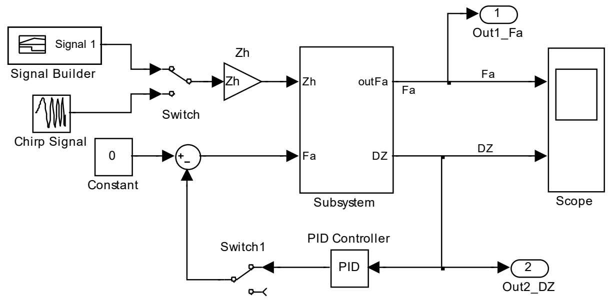


Close the Subsystem window, back to main model, save the model with name this block "Ue6_Subskfz_8a".



Schnitt 2:

Drag the model like right figure:



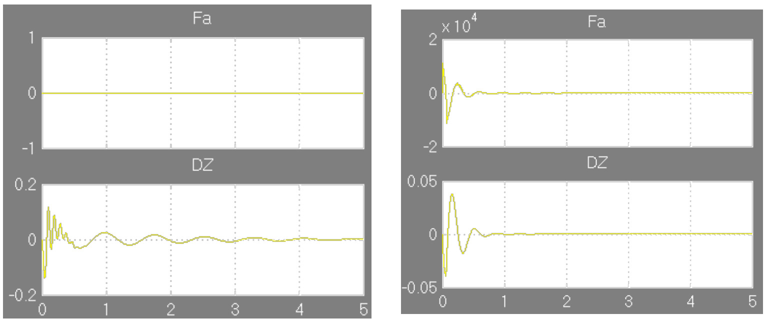
Schnitt 3: simulation

open loop | closed loop

Run model 5 s with open loop:

Run model 5 s with closed loop:

- Fixed-step size 1/128
- Switch Signal 1



Run model 30 s with open loop:

Run model 30 s with closed loop:

- Fixed-step size 1/128
- Switch Chrip Signal