

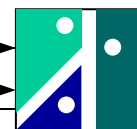
Modellbildung und Simulation SS 2017
 Prof. Dr. W. Kröber

Zur Bewertung der Aufgaben muss der gesamte Lösungsweg ersichtlich sein.

- Bearbeitungszeit : 90 min
- Erlaubte Hilfsmittel :
 - selbstständige Lösung
 - keine weiteren Einschränkungen

Note : _____

Aufgabe	erreichte Punkte
1	
2	
3	
4	
5	
6	
Summe	



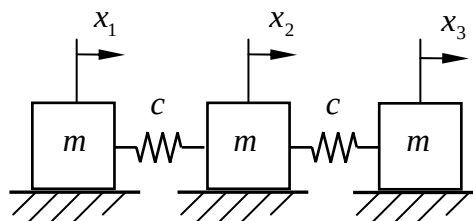
HS Koblenz
 FB Ingenieurwesen
 FR Maschinenbau
 © Prof. Dr. Kröber
 Modellbildung und Simulation
 Prüfung 12.07.2017

Hinweis zum Abspeichern: Die abzuspeichernden Ergebnisdateien beginnen mit den sechs Buchstaben *nnnvvv*. Dabei steht *nnn* für die ersten drei Buchstaben des Nachnamens und *vvv* für die ersten drei Buchstaben des Vornamens.

Aufgabe 1 (17P) Ergebnisdateien → *nnnvvv1.m* und *nnnvvv1_sl.slx*

Das abgebildete Schwingungssystem hat die Freiheitsgrade x_1 , x_2 und x_3 . Die Differentialgleichungen zur Beschreibung des Bewegungsablaufes lauten:

$$\begin{aligned} m \cdot \ddot{x}_1 &= -c \cdot (x_1 - x_2) \\ m \cdot \ddot{x}_2 &= -c \cdot (x_2 - x_1) - c \cdot (x_2 - x_3) \\ m \cdot \ddot{x}_3 &= -c \cdot (x_3 - x_2) \end{aligned}$$

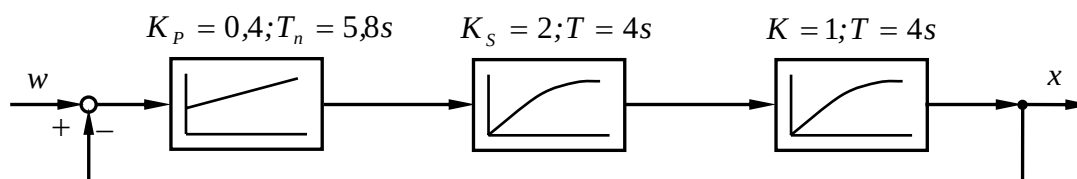


Die Anfangsauslenkungen seien $x_1(t=0) = x_0$, $x_2(t=0) = 0$ und $x_3(t=0) = -x_0$. Nach Vorgabe der Daten in MATLAB soll die Lösung in Simulink erfolgen. Die graphische Darstellung soll wieder in MATLAB realisiert werden. Die Simulationsdauer und feste Schrittweite werden in MATLAB vorgegeben. In einem Plotbefehl sollen x_1 , x_2 und x_3 in Abhängigkeit der Zeit dargestellt werden. Wählen Sie als Lösungsverfahren zur Lösung der Differentialgleichungen "ode3 Bogacki-Shampine".

Zahlenwerte: $m = 10 \text{ kg}$; $c = 200 \text{ N/m}$; $x_0 = 2 \text{ mm}$; $t_{\text{sim}} = 5 \text{ s}$; $\Delta t = 0,001 \text{ s}$

Aufgabe 2 (17P) Ergebnisdateien → *nnnvvv2.m* und *nnnvvv2_sl.slx*

Der Wirkungsplan zeigt einen Regelkreis mit einer Regelstrecke 2. Ordnung und einem PI-Regler. Untersucht werden soll ein Sollwertsprung. Zunächst seien alle Variablen gleich Null.

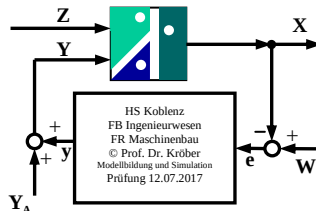


Der Sollwertsprung zum Zeitpunkt $t_1 = 1 \text{ s}$ beträgt $w = w_\infty = 5$. Geben Sie die Daten in MATLAB vor und führen Sie die Berechnung in SIMULINK durch. Die graphische Darstellung soll wieder in MATLAB realisiert werden. Hier soll die Führungsgröße und die Regelgröße in Funktion der Zeit dargestellt werden. Die Simulationsdauer und feste Schrittweite werden in MATLAB vorgegeben ($t_{\text{sim}} = 40 \text{ s}$; $\Delta t = 0,01 \text{ s}$). Wählen Sie als Lösungsverfahren zur Lösung der Differentialgleichungen "ode3 Bogacki-Shampine".

Aufgabe 3 (17P) Ergebnisdateien → *nnnvvv3.m* und *nnnvvv3_sl.slx*

Es wird eine Excel-Datei mit dem Dateinamen "daten.xlsx" zur Verfügung gestellt. In der Datei befinden sich in Spalte 1 Zahlenwerte, die mit einem zu erstellenden MATLAB-Programm eingelesen werden sollen. Insgesamt sind dies 2001 Zahlenwerte in 2001 Zeilen. Die Zeitschrittweite ist in der Exceldatei nicht aufgeführt, beträgt aber stets $\Delta t = 0,01$ s. Somit ergibt sich eine Simulationsdauer $t_{\text{sim}} = 20$ s. Die Daten sollen nach SIMULINK übergeben werden. Dort wirken diese Daten als Eingangsgröße u auf einen Tiefpassfilter 1. Ordnung. Die Ausgangsdaten des Tiefpassfilters v sollen wieder nach MATLAB zurückgegeben werden. In einem Plotbefehl sollen die Eingangs- und Ausgangsgrößen des Tiefpassfilters (also $u(t)$ und $v(t)$) graphisch dargestellt werden. Der Tiefpassfilter hat die Anfangsbedingung $v(t = 0) = 5$. Die Zeitkonstante sei $T = 0,5$ s. Wählen Sie als Lösungsverfahren zur Lösung der Differentialgleichungen "ode3 Bogacki-Shampine"!

Tiefpassfilter: $G = \frac{v}{u} = \frac{1}{1 + j\omega T}$



Hinweis für spätere Semester:

In Aufgabe 4 ist auch das Makro angegeben, mit dem die Werte prinzipiell erzeugt wurden.

Aufgabe 4 (16P) Ergebnisdatei → *nnnvvv4.xlsm*

Der nebenstehende Teil eines/des Makros ist bereits erstellt. Das Makro wird mit der Tastenkombination „Strg y“ gestartet. Es schreibt in Spalte 1 des Excel-Sheets die Zeitachse und in Spalte 2 die Eingangswerte $u(t)$.

Das Programm ist so zu ergänzen, dass die Eingangswerte $u(t)$ tiefpassgefiltert werden. Der Tiefpassfilter hat die Anfangsbedingung $v(t = 0) = 5$. Die tiefpassgefilterten Werte $v(t)$ sollen in die Spalte 3 des Excel-Sheets eingetragen werden. Es soll eine Graphik "u und v über der Zeit" erstellt werden.

Rekursionsalgorithmus für den Tiefpassfilter:

$$v_{i+1} = \left(1 - \frac{\Delta t}{T}\right) \cdot v_i + \left(\frac{\Delta t}{T}\right) \cdot u_i$$

Zahlenwerte: $\Delta t = 0,01$ s ; $T = 0,5$ s

Auszug (oben und unten) aus der Excel-Datei:

	A	B
1	0	
2	0,01	
3	0	
4	0,02	
5	0	
6	0,03	
7	0	
8	0,04	
9	0	
10	0,05	
11	0	
12	0,06	

	A	B
1993	0	
1994	9,97	
1995	0	
1996	9,98	
1997	0	
1998	9,99	
1999	0	
2000	10	
2001	0	
2002		

```
Sub Makro1()
'
' Makro1 Makro
'
' Tastenkombination: Strg+y
'
    Dim u(2000) As Double
    dt = 0.01
    T = 0.5

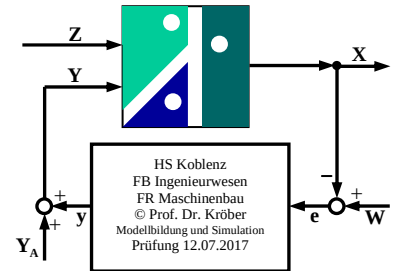
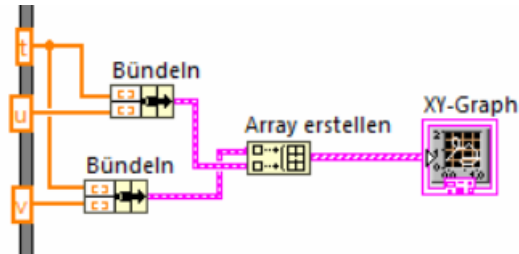
' Eingangsgrößenverlauf festlegen
    For i = 0 To 999
        u(2 * i + 1) = 0.01 * (i + 1)
    Next i
    For i = 0 To 1000
        u(2 * i) = 0
    Next i

' Ausgabe in Excel-Sheet
    For i = 0 To 2000
        Cells(i + 1, 1) = i * dt
        Cells(i + 1, 2) = u(i)
    Next i
End Sub
```

Aufgabe 5 (16P) Ergebnisdatei → *nnnvvv5.vi*

Grundsätzlich soll hier das gleiche Problem wie in Aufgabe 4 bearbeitet werden. Die Lösung soll in einem Formelknoten in LabVIEW gelöst werden. Der Kurvenverläufe $u(t)$ und $v(t)$ sollen hier in einem XY-Schrieb ausgegeben werden.

Hilfestellung: Der Graphikteil könnte prinzipiell so aussehen

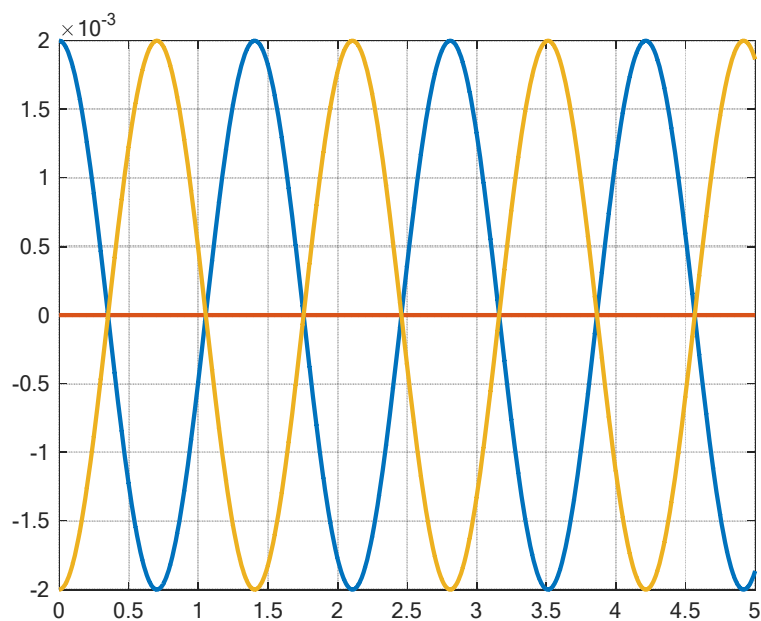
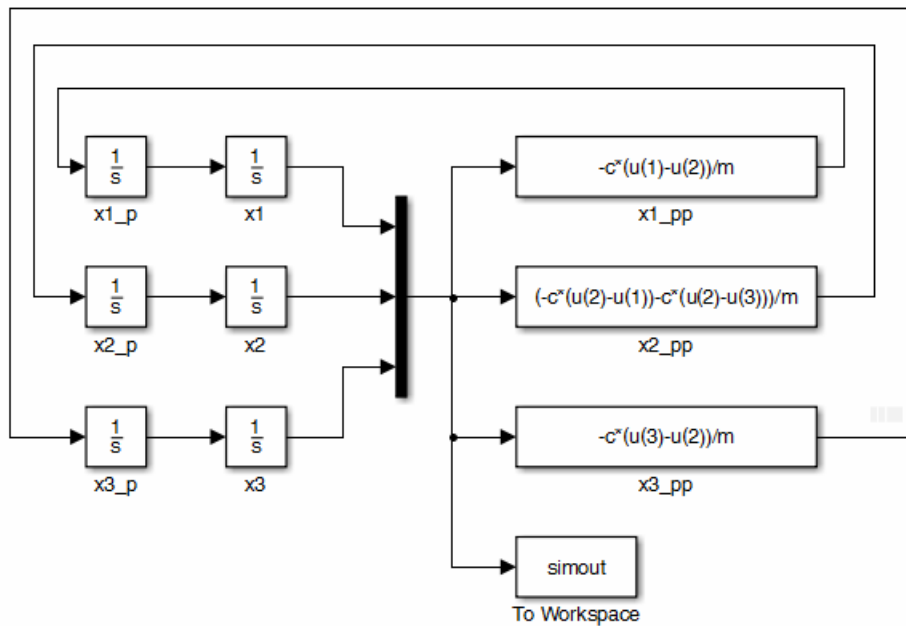
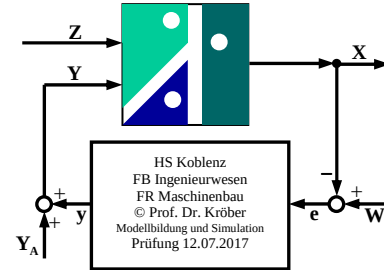


Aufgabe 6 (17P) Ergebnisdatei → *nnnvvv6.vi*

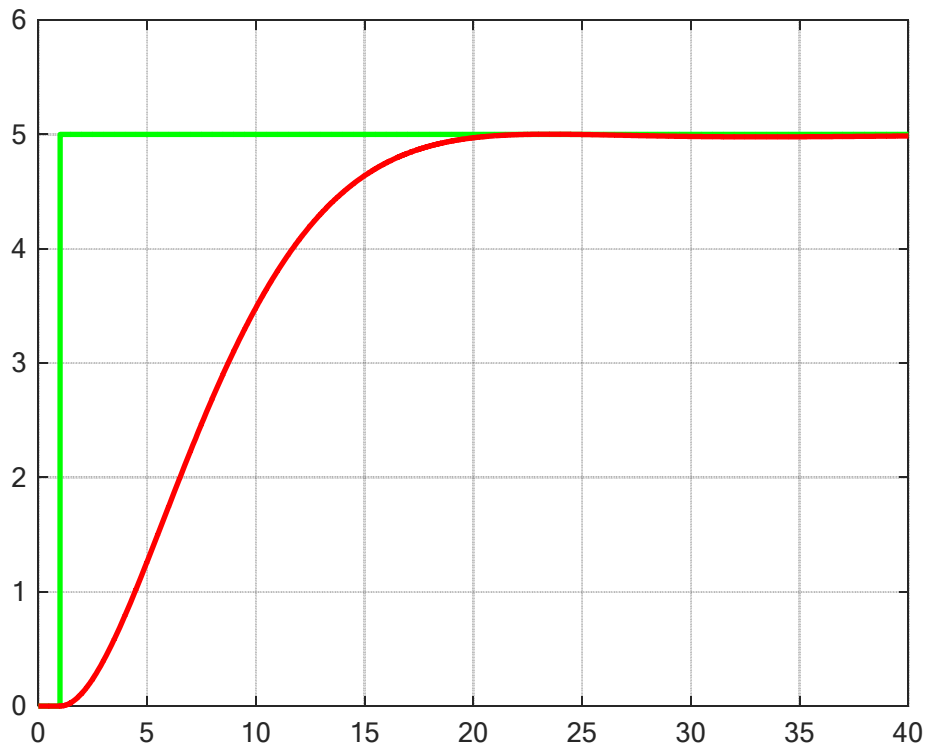
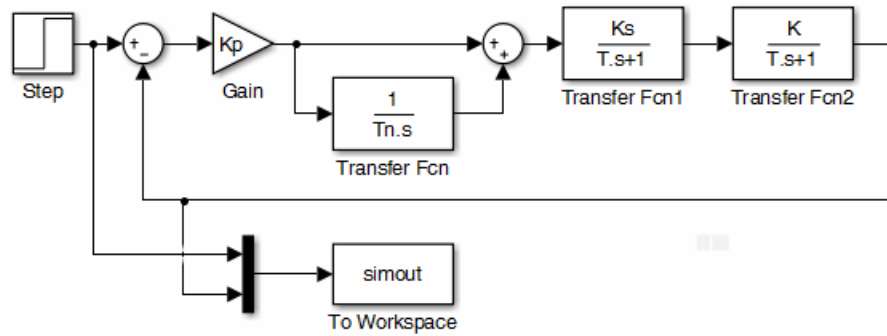
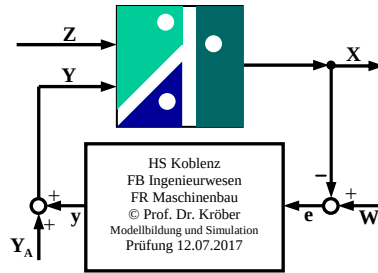
Mit dem USB6008 ist durch das Steckmodul ein NTC-Widerstand zur Bestimmung der Temperatur angeschlossen. Als Ausgangspunkt kann ein vorhandenes Programmbeispiel verwendet werden, bei dem die Temperatur in Funktion der Zeit angezeigt wird.

Das Programm ist dahingehend zu ergänzen: Es soll die minimale Temperatur ϑ_{min} und maximale Temperatur ϑ_{max} seit Programmstart ermittelt werden. In einem numerischen Ausgabefenster soll die Differenz $\vartheta_{max} - \vartheta_{min}$ ausgegeben werden. Somit ist die maximale Temperaturspanne seit Programmstart ersichtlich.

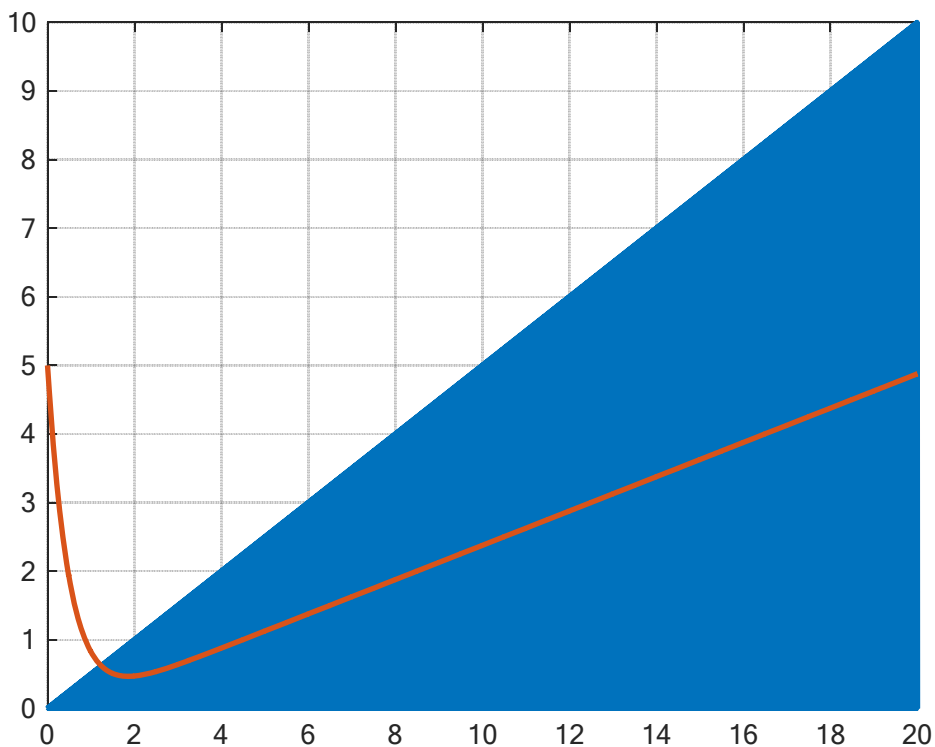
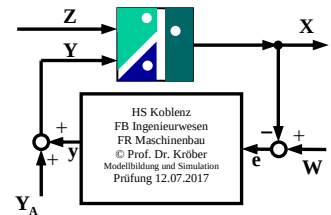
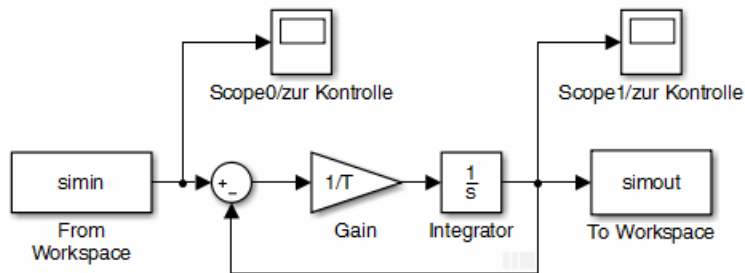
```
clear;clc;
m=10;
c=200;
x0=0.002;
tsim=5;
dt=0.001;
sim('krowol1_sl');
t=simout.time;
x1=simout.signals.values(:,1);
x2=simout.signals.values(:,2);
x3=simout.signals.values(:,3);
plot(t,x1,t,x2,t,x3,'Linewidth',2);grid;
```



```
clear;clc;
Kp=0.4;
Tn=5.8;
Ks=2;
T=4;
K=1;
t1=1;
wun=5;
tsim=40;
dt=0.01;
sim('krowol2_sl');
t=simout.time;
w=simout.signals.values(:,1);
x=simout.signals.values(:,2);
plot(t,w,'g',t,x,'r','Linewidth',2);grid;
```



```
clear;clc;
u=xlsread('c:\mus\Prfg_12_07_2017\daten','A1:A2001');
dt=0.01;
tsim=20;
T=0.5;
AB=5;
t(:,1)=0:dt:2000*dt; % Zeitvektor
simin(:,1)=t(:,1); % Vorbereitung der Übergabe an Simulink
simin(:,2)=u(:,1);
sim('krowol3_sl');
plot(t,u,t,simout.signals.values,'Linewidth',2);grid;
```



```

Sub Makrol1()
'
' Makrol Makro
'
' Tastenkombination: Strg+y
'

    Dim u(2000), v(2000) As Double
    dt = 0.01
    T = 0.5

' Eingangsgrößenverlauf festlegen
    For i = 0 To 999
        u(2 * i + 1) = 0.01 * (i + 1)
    Next i
    For i = 0 To 1000
        u(2 * i) = 0
    Next i

' Ausgangsgrößenverlauf berechnen
    v(0) = 5
    For i = 0 To 1999
        v(i + 1) = v(i) * (1 - dt / T) + u(i) * dt / T
    Next i

' Ausgabe in Excel-Sheet
    For i = 0 To 2000
        Cells(i + 1, 1) = i * dt
        Cells(i + 1, 2) = u(i)
        Cells(i + 1, 3) = v(i)
    Next i

End Sub

```

