Verwenden Sie als Grundlage für diese Aufgabe das Modell des Lorenz Attraktors. Die Differentialgleichungen dazu sind

$\dot{x} = \sigma(y - x)$

$\dot{y} = -xz + \rho x - y$

$\dot{z} = xy - \beta z$

mit

$\sigma = 3$

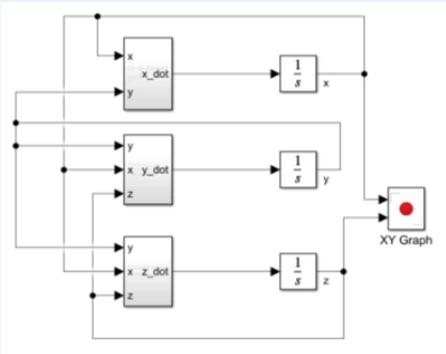
$\rho = 26.3$

$\beta = 1$

und den Anfangsbedingungen $x(0) = 5$, $y(0) = 5$ und $z(0) = 5$.

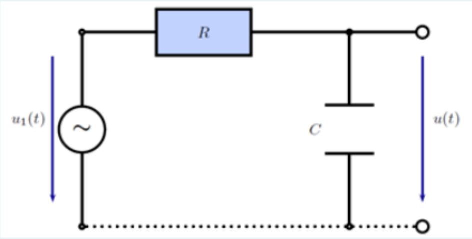
Vereinfachen Sie das Modell mit Hilfe von drei MATLAB Function-Blöcken für die drei Differentialgleichungen (d.h. je eine für die Terme auf der linken Seite des Gleichzeichens).

Verwenden Sie als Grundlage für diese Aufgabe wieder das Modell des Lorenz Attraktors. Vereinfachen Sie das Modell mit Hilfe von drei Subsystemen für die drei Differentialgleichungen (d.h. je eine für die Terme auf der linken Seite des Gleichzeichens, siehe Bild). Geben Sie den Ein- und Ausgängen Ihrer Subsysteme geeignete Namen.



Hinweis:
Sie können dazu den Block Subsystem aus der Bibliothek unter Ports & Subsystems verwenden oder mit selektieren der relevanten Blöcke und dann über das Kontext-Menü ein Subsystem erstellen.

Erstellen Sie ein Simulink-Modell für die RC-Kombination für das gilt:



$\frac{d}{dx}u(t) = -\frac{1}{RC}u(t) + \frac{1}{RC}u_1(t)$

Berechnen Sie den zeitlichen Verlauf für $t = 0...2.5$ s der Ausgangsspannung $u(t)$. Verwenden Sie dazu

- Widerstand: *R* = 10 kΩ
- Kondensator: *C* = 47e-6 F

Verwenden Sie für die Eingangsspannung einen «From Workspace» Block und für die Ausgangsspannung den «To Workspace» Block. Definieren Sie die Konstanten und die Eingangsspannung des Modells in einem Script namens *uebung_4_init.m*. Die Eingangsspannung ist folgende Funktion:

$$u_1(t) = \begin{cases} 2t, & \text{if } t \leq 1 \\ 2, & \text{otherwise} \end{cases}$$

Geben Sie unten die beiden Dateien ab.

Hinweis:

Lesen Sie in den Eigenschaften des «From Workspace» Blocks, wie in welcher Form die Eingangsspannung Simulink übergeben werden soll. Verwenden Sie dazu das Format «Matrix».

Erzeugen Sie zwei neue MATLAB-Dateien, *pendgl.m* und *pendgl_lin.m*. Beide Dateien sind MATLAB-Funktionen die unten abgegeben werden sollen. Die beiden Funktionen sind:

- *pendgl.m:* Implementation der nicht-lineare Pendelgleichung aus der Vorlesung, $\ddot{\alpha}(t) = -\frac{g}{l}\sin(\alpha(t))$
- *pendgl_lin.m:* Implementation der linearisierten Pendelgleichung. Diese ist $\ddot{\alpha}(t) = -\frac{g}{l}\alpha(t)$

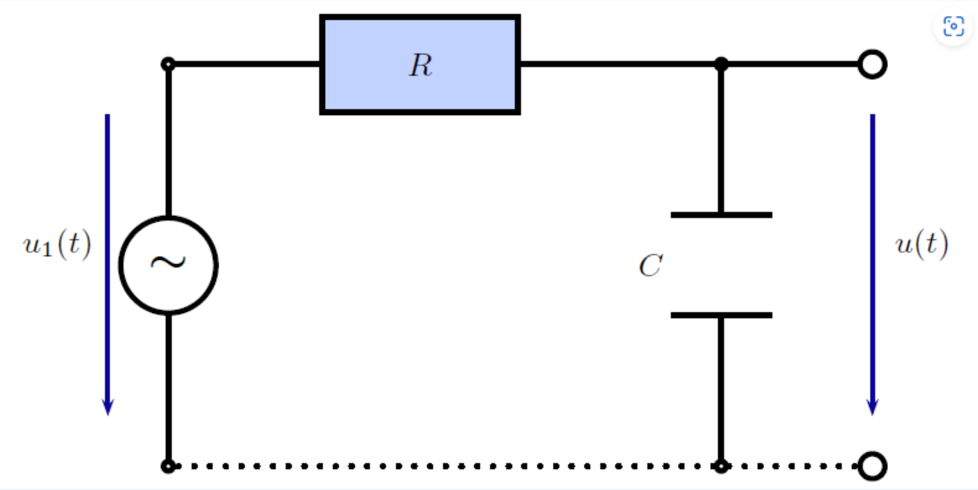
Verwenden Sie die Wert $l = 10m$ und $g = 9.81m/s^2$.

Entwickeln Sie die beiden Funktionen, damit die Differentialgleichungen mit dem vordefinierten Code unten für die Anfangswerte $\alpha(t=0) = \frac{19\pi}{20}$ und $\dot{\alpha}(t=0) = 0$ berechnet wird.

Hinweise:
Der vordefinierte Code wird zum Lösen der beiden Systeme verwendet und Sie können ihn auch zum Testen und Plotten Ihrer Funktionen in Ihrem lokalen MATLAB verwenden.

Weitere Fragestellung:
Vergleichen Sie die beiden Funktionen miteinander. Welche Funktion beschreibt das System besser?

Bestimmen Sie die Lösung für die Differentialgleichung einer RC-Kombination in MATLAB:



$\frac{d}{dt}u(t) = -\frac{1}{RC}u(t) + \frac{1}{RC}u_1(t)$

Berechnen Sie den zeitlichen Verlauf für $t = 0, 0.1, \ldots, 2.5$ Sekunden der Ausgangsspannung $u(t)$. Verwenden Sie dazu

- Widerstand: R =10'000 Ohm
- Kondensator: C =47e-6 Farad
- Eingangsspannung: $u_1(t) = \begin{cases} 1V, & \text{if } t \geq 1 \\ 0V, & \text{otherwise} \end{cases}$

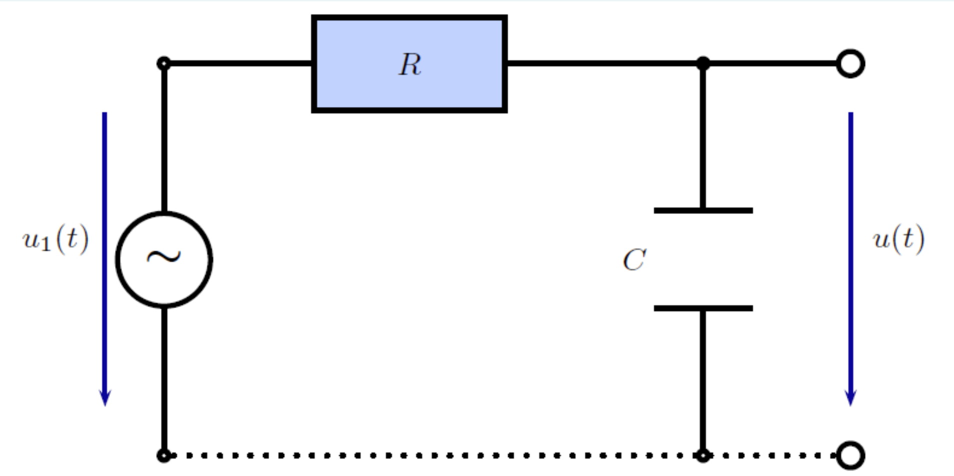
Geben Sie die MATLAB-Funktion *rckomb(t, u)* für die Differentialgleichung (d.h. $\frac{d}{dt}u(t)$) und die MATLAB-Funktion *u1_sprung(t)* für die Eingangsspannung (d.h. $u_1(t)$) als Dateien ab. Erweitern Sie den Code unten so, dass die Lösung der Differentialgleichung für einen Anfagnswert von $u = 0$ V berechnet wird.

Hinweis:

Die Funktion für $u_1(t)$ finden Sie in den Folien der Vorlesung.

Diese Aufgabe basiert auf der Lösung von Aufgabe 9-1 Aufgabe 2.

Bestimmen Sie die Lösung für die Differentialgleichung einer RC-Kombination aus der vorhergehenden Aufgabe in MATLAB, aber dieses Mal mit einer **sinusförmigen Eingangsspannung** $u(t)_1$ mit einer **Frequenz von** $f = 3$ **Hz**.



$\frac{d}{dt}u(t) = -\frac{1}{RC}u(t) + \frac{1}{RC}u_1(t)$

Berechnen Sie den zeitlichen Verlauf für $t = 0, 0.01, \ldots, 2.5$ Sekunden der Ausgangsspannung $u(t)$. Verwenden Sie dazu

- Widerstand: R =10'000 Ohm
- Kondensator: C =47*1e-6 Farad
- Eingangsspannung: $u_1(t) = \sin(2\pi \cdot f \cdot t)$

Geben Sie die MATLAB-Funktion *rckomb2(t, u)* für die Differentialgleichung und die MATLAB-Funktion *u1_sinus(t)* für die Eingangsspannung als Dateien ab. Erweitern Sie den Code unten so, dass die Lösung der Differentialgleichung für einen Anfagnswert von $u = 0$ V mit dem Solver *ode45* berechnet wird.

Lösen Sie mit Hilfe von MATLAB die folgende Differentialgleichung eines Wachstumsprozesses:

$$\dot{P}(t) = \alpha(P(t))^{\beta}$$

mit $\alpha = 2.2$, $\beta = 1.0015$ und $P(t = 0) = 100$. Berechnen Sie die Lösung für das Interval $t = 0, \ldots, 2.5$, wobei Sie den Zeitschritt nicht explizit angeben, sondern ihrn von MATLAB automatisch bestimmen lassen.

Geben Sie die Funktion *wachstum(t, P)* als MATLAB-Datei unten ab und passen Sie den Code so an, dass Ihre Funktion für die Berechung des Anfangswertproblems mit dem Solver ode23 verwendet wird.

Lösen Sie mit Hilfe von MATLAB das Differentialgleichungssystem

$$\begin{cases} \dot{y}_1(t) = -2y_1(t) - y_2(t) \\ \dot{y}_2(t) = 4y_1(t) - y_2(t) \end{cases}$$

mit

$$\begin{cases} \dot{y}_1(t = 0) = 1 \\ \dot{y}_2(t = 0) = 1 \end{cases}$$

und einem Zeitintervall von $t = 0, 0.1, \ldots, 5$ Sekunden.

Implementieren Sie die Differentialgleichung in der MATLB-Funktion *dgl6* und geben Sie die Datei unten ab.

Lösen Sie mit Hilfe von MATLAB das Differentialgleichungssystem

$$\dddot{y} + 3\ddot{y} + 3\dot{y} + y = 0$$

mit

$$\begin{cases} y(0) = 0 \\ \dot{y}(0) = 1 \\ \ddot{y}(0) = 1 \end{cases}$$

und einem Zeitintervall von $t = 0, 0.1, \ldots, 5$ Sekunden.

Geben Sie die Funktion *dgl7* als MATLAB-Datei unten ab und passen Sie den Code so an, dass Ihre Funktion für die Berechung des Anfangswertproblems mit dem Solver ode23 verwendet wird.

Lösen Sie mit Hilfe von MATLAB die Differentialgleichung des mathematischen Pendels aus den Folien für verschiedene Pendellängen. Die jeweilige Pendellänge soll dabei bei Aufruf des Lösungsalgorithmus (ode23) übergeben werden können. Plotten Sie anschliessend die Lösungen für den Anfangswert $\alpha(t = 0) = \frac{\pi}{4}$, das Zeitintervall $t = 0, 0.1, \ldots, 5$ Sekunden und Pendellängen von $l_1 = 20$ und $l_2 = 5$.

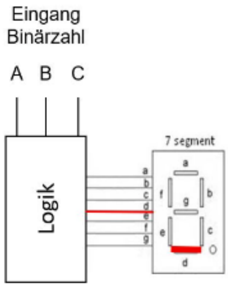
Geben Sie die MATLB-Funktion pendgl_ext als Datei unten ab.

Hinweis:

Verwenden Sie die MATLAB-Hilfe, um zu ermitteln, auf welche Weise zusätzliche Parameter an die Definitionsfunktion der Differentialgleichung übergeben werden können. Als Start können Sie diesen Link in der MATLAB Onlinehilfe oder diesen Link auf MATLAB Central verwenden.

1. Geben Sie die symbolische Logikfunktion an, wann die Anzeige **d NICHT** leuchtet. Weisen Sie diese Funktion der Variablen D zu.
2. Vereinfachen Sie D und weisen Sie die neue Funktion der Variablen Z zu.

Zahl	A	B	C	Segment d
0	0	0	0	1
1	0	0	1	0
2	0	1	0	1
3	0	1	1	1
4	1	0	0	0
5	1	0	1	1
6	1	1	0	1
7	1	1	1	0



1. Geben Sie die symbolische Logikfunktion an, wenn das Licht der Ampel auf **Gelb schaltet**. Weisen Sie diese Funktion der Variablen D zu.
2. Vereinfachen Sie D und weisen Sie die neue Funktion der Variablen Z zu.

A	B	C	Grün	Gelb	Rot
0	0	0	1	0	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	0	0	1

1. Geben Sie die symbolische Logikfunktion an, wenn das Licht der Ampel auf **Rot schaltet**. Weisen Sie diese Funktion der Variablen D zu.
2. Vereinfachen Sie D und weisen Sie die neue Funktion der Variablen Z zu.

A	B	C	Grün	Gelb	Rot
0	0	0	1	0	0
0	0	1	1	0	0
0	1	0	1	0	0
0	1	1	0	1	0
1	0	0	0	0	1
1	0	1	0	0	1
1	1	0	0	0	1
1	1	1	0	0	1

Gegeben sind Beobachtungen von blauen Sonnenbarschen, einer Fischspezies. Bei diesen Beobachtungen wurde die Gösse und das Alter der Fische aufgezeichntet und Sie finden die Daten in dieser Datei.

Passen Sie ein lineares Regressionsmodell den Daten an. Das Modell soll die Form

$Groesse = \beta_0 + \beta_1 \cdot Alter$

haben. Verwenden Sie für das Modell die Matrixform aus der Vorlesung

$\mathbf{y} = \mathbf{X}\mathbf{b}$

Erstellen Sie den Vektor **y** und die Matrize **X**. Berechnen Sie anschliessend die Werte für die Unbekanten und speichern Sie das Resultat in der Variable **b**.

Zusatzaufgabe:

Visualisieren Sie die Beobachtungen und den Fit in einem Plot in MATLAB. **Geben Sie den Code** für die Visualisierung **nicht** im Feld unten **ab**. Die Musterlösung für den Plot wird für Sie sichtbar, wenn Sie den Test abschliessen.

Gegeben ist ein Datensatz mit dem Gewicht und dem Verbrauch von Autos aus den Jahren 1970, 1976 und 1982. Die Daten sind in dieser Datei zu finden und das Gewicht ist in kg und der Verbrach in Liter/100km.

Passen Sie ein lineares Regressionsmodell den Daten an. Das Modell soll die Form

$Verbrauch = \beta_0 + \beta_1 \cdot Gewicht$

haben. Verwenden Sie für das Modell die Matrixform aus der Vorlesung

$\mathbf{y} = \mathbf{X}\mathbf{b}$

Erstellen Sie den Vektor **y** und die Matrize **X**. Berechnen Sie anschliessend die Werte für die Unbekanten und speichern Sie das Resultat in der Variable **b**.

Machen Sie zusätzlich eine Vorhersage für den Verbrauch bei einem Gewicht von 2250kg. Speichern Sie das Resultat der Vorhersage in der Variable **y_pred**. Wie gross ist der vorhergesagte Verbrauch?

Zusatzaufgabe:

Visualisieren Sie die Beobachtungen, den Fit und die Vorhersage in einem Plot in MATLAB. **Geben Sie den Code** für die Visualisierung **nicht** im Feld unten **ab**. Die Musterlösung für den Plot wird für Sie sichtbar, wenn Sie den Test abschliessen.

Erstellen Sie ein neues Simulink-Modell für den senkrechten Wurf (siehe Folien), welches die folgende Funktion modelliert:

$\ddot{y} = g$ mit $g = -9.81$

Führen Sie das Simulink-Modell aus (Start Time 0.0, End Time 10.0) und überprüfen Sie das Resultat graphisch mit der Lösung auf den Folien in der Vorlesung. Verwenden Sie dazu die Anfangsbedingungen

$v(0) = 0$ und $y(0) = 0$

Hinweis:

- Die Anfangsbedingungen können in den Eigenschaften eines Integrators eingestellt werden.
- Die Simulationsdauer kann entweder in den Einstellungen des Solvers oder im Menü *SIMULATION* im Feld *Stop Time* eingestellt werden.

Verwenden Sie für diese Aufgabe aufgabe_5.slx als Vorlage. Verbinden Sie bei Ihrer Lösung den Eingang des bestehenden Multiplexer in der Vorlage mit Ihrer Lösung $y(t)$ und $v(t) = \dot{y}$. Dies generiert nach dem Ausführen des Simulink-Modells eine Datei namens *loes5.mat*. Laden Sie die Datei *loes5.mat* und *aufgabe_5.slx* anschliessend hoch und überprüfen Sie das Resultat.

Schreiben Sie ein Simulink-Modell, das folgenden Satz von nichtlinearen Differentialgleichungen löst:

$\dot{x} = \sigma(y - x)$
$\dot{y} = -xz + \rho x - y$
$\dot{z} = xy - \beta z$

mit

$\sigma = 3$

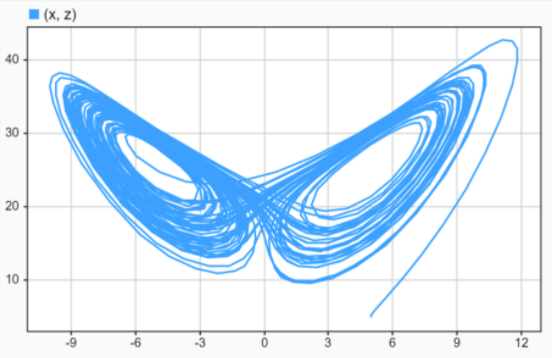
$\rho = 26.3$

$\beta = 1$

Zeichnen Sie x über z in einem *XY Graph* für eine Simulationsdauer von 0 bis 100 Sekunden auf. Verwenden Sie als Anfangsbedingungen

$x(0) = 5$, $y(0) = 5$ und $z(0) = 5$

Das Resultat sollte so aussehen:



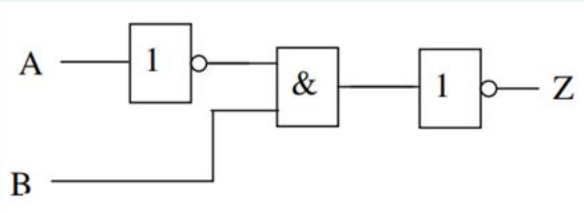
Hinweis:

Die drei Differentialgleichungen beschreiben den aus der Mathematik und Physik bekannten Lorenz Attraktor. Sie stellen ein einfaches Modell atmosphärischer Konvektionsvorgänge dar und wurden 1961 von Edward Lorenz zur Wettervoraussage aufgestellt. Für bestimmte Kombinationen von σ, ρ und β entsteht ein chaotisches Systemverhalten, das um zwei Gleichgewichtspunkte im Raum schwingt.

Es ist keine Abgabe dieser Aufgabe nötig.

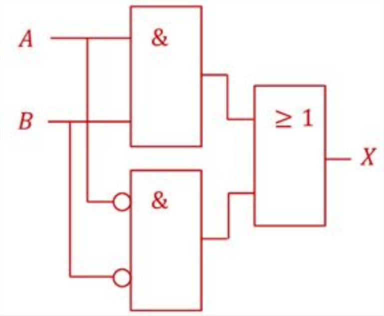
Geben Sie für die nachfolgende Schaltung die boolesche Gleichung für die Variable Z an.

Geben Sie nur eine Formel für Z ab.



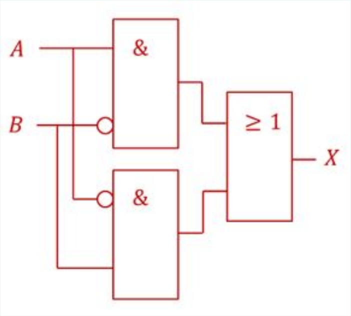
Geben Sie für die nachfolgende Schaltung die boolesche Gleichung für die Variable X an.

Geben Sie nur eine Formel für X ab.



Geben Sie die Wahrheitstabelle für die folgende Schaltung an:

(Hinweis: Vervollständigen Sie die Wahrheitstabellen für die booleschen Ausdrücke. Erstellen Sie zunächst alle möglichen Kombinationen für die Eingänge mit Hilfe einer MATLAB-Funktion (siehe Funktion truth_table aus den Folien) und speichern Sie die Vektoren in den Variablen A und B. Vervollständigen Sie dann die Tabelle, indem Sie den Ausdruck für die Variable X definieren.



A	B	X
0	0	
0	1	
1	0	
1	1	

Definieren Sie den symbolischen Ausdruck $x1 = (4t^2 - 2t + 8)^3$ und erweitern Sie das Polynom mit dem Befehl "expand".

Speichern Sie das Ergebnis in der Variablen x2.

Definieren Sie den symbolischen Ausdruck

$$y1 = \frac{2t^5 e^{3t-3}}{t^2+2t}.$$

Vereinfachen Sie y1 und speichern Sie das Ergebnis in y2.

Definieren Sie die symbolische Funktion $f(x) = \sqrt{5x^2 - 7x + 4}$ als Variable f.

Leiten Sie mit der Matlab Funktion diff die symbolische Funktion f einmal ab.

Speichern sie das Ergebnis in der Variablen df.

Mit der Funktion diff können partielle Ableitungen von symbolisch definierten Funktionen berechnet werden. Bestimmen Sie die Summe der beiden ersten partiellen Ableitungen der Funktion $g = xy + x^2$ und speichern Sie das Ergebnis in der Funktion g_s.

In dieser Übung sollen Sie einen Messbericht für ein Praktikum mit einen MATLAB Live Script erstellen. Verwenden Sie das Template *Messbericht.mlx* mit den Messdaten *i_mess.mat*

Fügen Sie die folgenden Punkte dazu und achten Sie auf die entsprechende Formatierung (Formel):

- im Abschnitt *Praktikumsbeschrieb*: In diesem Praktikum wurden zwei verschiedene Widerstände auf ihren Wert überprüft. Dafür wurde je zehnmal der Strom bei einer angelegten Spannung 24 V von gemessen.
- im Abschnitt *Berechnung der Widerstände*: $R = \frac{U}{i}$

Setzen Sie den Messbericht in den '*Hide code*' Modus und führen Sie nun das *MATLAB Live Script* aus.

Geben Sie den Messbericht als Live Script - und als PDF-Datei ab.