

Robert Salvador

salvador@htlinn.ac.at

Numerische Simulation analoger linearer und nichtlinearer Regelkreise



- **Mathematische / Fachliche Inhalte in Stichworten**
Regelungstechnik, mathematische Modellbildung, Numerik von Differenzialgleichungen, Simulation dynamischer Systeme
- **Kurzzusammenfassung**
**Für verschiedene analoge Regelkreisglieder werden zunächst mathematische Modelle entwickelt. Dazu gehört die PT_n-Strecke (bei Bedarf auch mit Totzeit), klassische P-, I-, PI-, PID-Regler und nichtlineare Regler (Zweipunktregler ohne und mit Hysterese).
Diese Einzelsysteme werden dann zu einem Regelkreis "verschaltet", und das Verhalten des Gesamtsystems wird iterativ berechnet.**
- **Lehrplanbezug (bzw. Gegenstand / Abteilung / Jahrgang)**
Regelungstechnik bzw. Industrielle Elektronik, Angewandte Mathematik (Differenzialgleichungen), eventuell auch angewandte Informatik (Modellbildung, Simulation, Iteration)
- **Mathcad-Version**
Mathcad 11
- **Hinweis:**



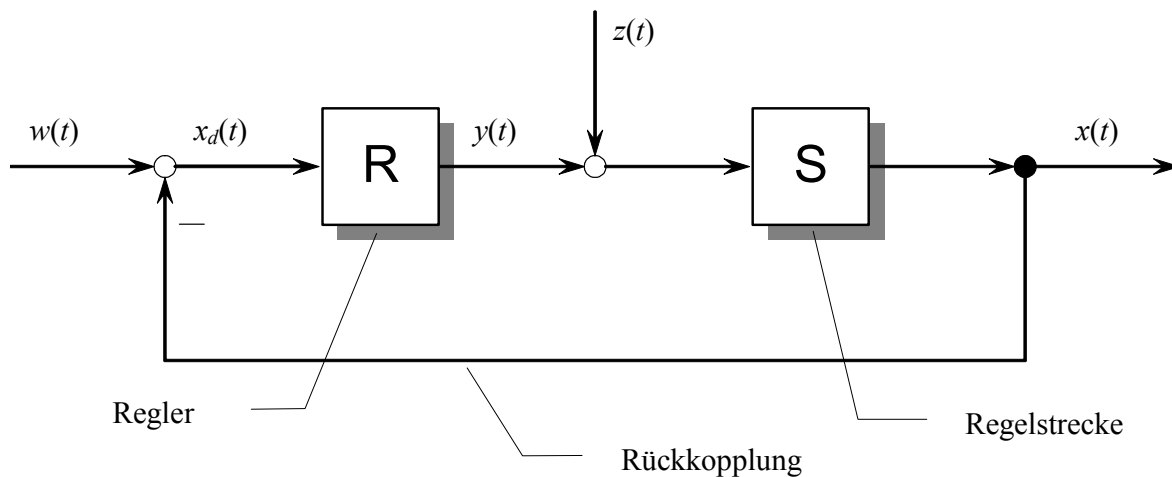
INHALT (gewünschten Bereich anklicken!):

- [Allgemeines](#)
- [Simulationsmodelle](#)
- [Simulationbeispiele](#)

Allgemeines**zum Anfang****Simulationsmodelle**

Das folgende Bild zeigt das Schema des Standardregelkreises:

Von der vorgegebenen *Führungsgröße* ("Sollwert") $w(t)$ wird die *Regelgröße* $x(t)$ (also jene Größe, die der Führungsgröße möglichst schnell und genau angeglichen werden soll) subtrahiert, was die *Regeldifferenz* $x_d(t)$ ergibt. Auf diese Regeldifferenz reagiert nun der Regler und erzeugt die sogenannte *Stellgröße* $y(t)$ (z. B. Ventilstellung in einer Heizanlage, Spannung an einem Motor, ...), mit der die Regelstrecke angesteuert wird. Das "Ausgangssignal" der Regelstrecke ist die schon erwähnte Regelgröße (z. B. die tatsächliche Raumtemperatur, Motordrehzahl, ...), die in der Realität natürlich messtechnisch erfasst und aufbereitet wird (dazu müssen unter Umständen noch weitere Regelkreisglieder mit einbezogen werden). An jeder beliebigen Stelle des Kreises können außerdem *Störgrößen* $z(t)$ eingreifen, deren Einfluss auf die Regelgröße ausgeregelt werden muss.



Wie in der Kurzzusammenfassung bereits erwähnt, geht es vorerst darum, für die einzelnen Teile des Regelkreises mathematische Modelle zu entwickeln. Zu diesen Teilen zählen außer verschiedenen Regler- und Streckentypen auch Subtraktionsstelle (zur Bildung der Regeldifferenz) und Additionsstelle (dort, wo z. B. eine Störgröße in den Regelkreis eingreift).

Jedes Modell eines Regelkreisgliedes wird durch eine entsprechende Funktion beschrieben, die einerseits die Modellparameter (z. B. Verstärkung, Zeitkonstante, ...) und andererseits einspaltige Matrizen (bzw. Vektoren) als Eingabegrößen erhält.

Diese Vektoren stehen für die Eingangs- und das Ausgangssignale des betreffenden Regelkreisgliedes. Da die Simulation mit einer diskretisierten Zeitvariablen arbeitet, stehen die Einträge in den einzelnen Zeilen der Eingangs- und Ausgangsvektoren für die Signalwerte zu den entsprechenden diskreten Zeitpunkten.

Damit sich das Übertragungssystem dynamisch weiterentwickeln kann, wird aus den momentanen und eventuell in der Vergangenheit liegenden Signalwerten jeweils ein Schritt in die Zukunft gerechnet. Das eigentliche Simulationsprogramm besteht nur aus einer Schleife, die diesen Schritt nach vorn für jedes Regelkreisglied berechnet und alle Zeitpunkte durchläuft.

Am Ende der Simulation liegen alle im Regelkreis vorkommenden Signale in Form von Vektoren vor. Um auf die interessanten Signale zugreifen zu können, werden die entsprechenden Vektoren "nebeneinander" gehängt; das Simulationsprogramm liefert die dabei entstehende Matrix als Rückgabewert.

Simulationsmodelle**zum Anfang****Simulationsbeispiele****Summenberechnung**

```

Sum_2(sum1, sum2, sum, n) := | sum_{n+1} ← sum1_n + sum2_n
                           | sum

```

Die Funktion *Sum_2* verlangt in der Variablenliste zuerst die beiden Vektoren jener Signale, die addiert werden sollen (Eingänge), dann den Vektor, in welchem das Summensignal (Ausgang) abgespeichert wird und am Schluss die aktuelle Zeilennummer, die dem momentanen Zeitpunkt entspricht.

Im folgenden wird ein Anwendungsfall gezeigt:

```

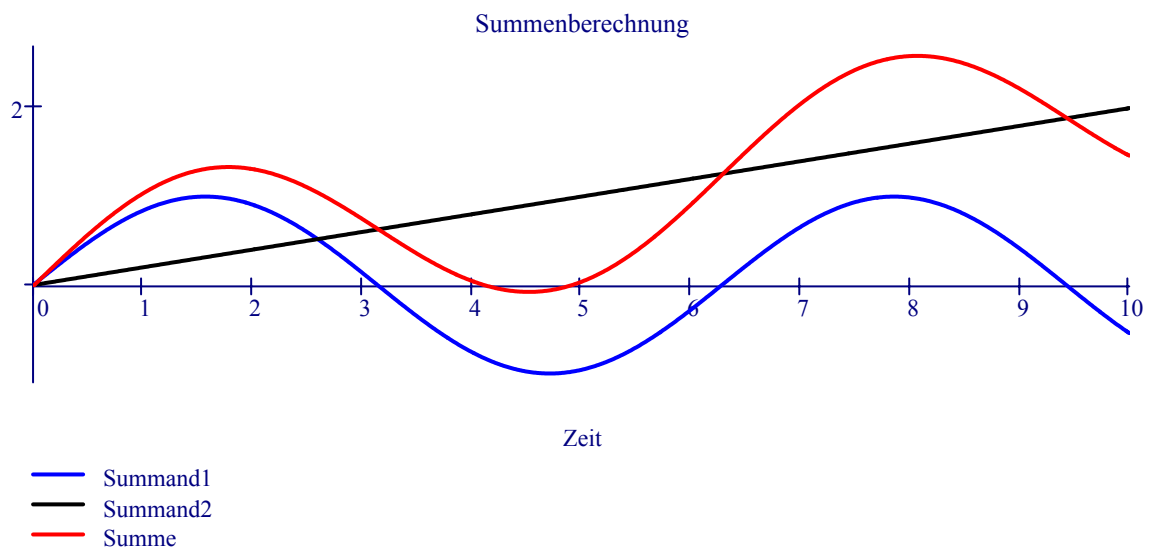
Zeit := 10      N := 1000      Δt :=  $\frac{\text{Zeit}}{N}$ 
k := 0..N      sum1_k := sin(k · Δt)      sum2_k :=  $\frac{k \cdot \Delta t}{5}$       sum_k := 0

```

```

sum := | for j ∈ 0..N
        |   feld_j ← 0
        | for k ∈ 0..N - 1
        |   feld ← Sum_2(sum1, sum2, feld, k)
        | feld

```



PT1-System

$$PT_1(K, T, in, out, \Delta t, n) := \begin{array}{l} d1 \leftarrow \Delta t \cdot \left(\frac{K \cdot in_n - out_n}{T} \right) \\ d2 \leftarrow \frac{\Delta t}{2 \cdot T} \cdot \left[\left(K \cdot in_n - out_n \right) - \left(K \cdot in_{\max(n-1, 0)} - out_{\max(n-1, 0)} \right) \right] \\ out_{n+1} \leftarrow out_n + d1 + d2 \\ out \end{array}$$

Die Funktion PT_1 beschreibt das Verhalten eines PT_1 -Systems. Dazu wurde die PT_1 -Differentialgleichung

$$T \cdot \frac{d}{dt} out(t) + out(t) = K \cdot in(t)$$

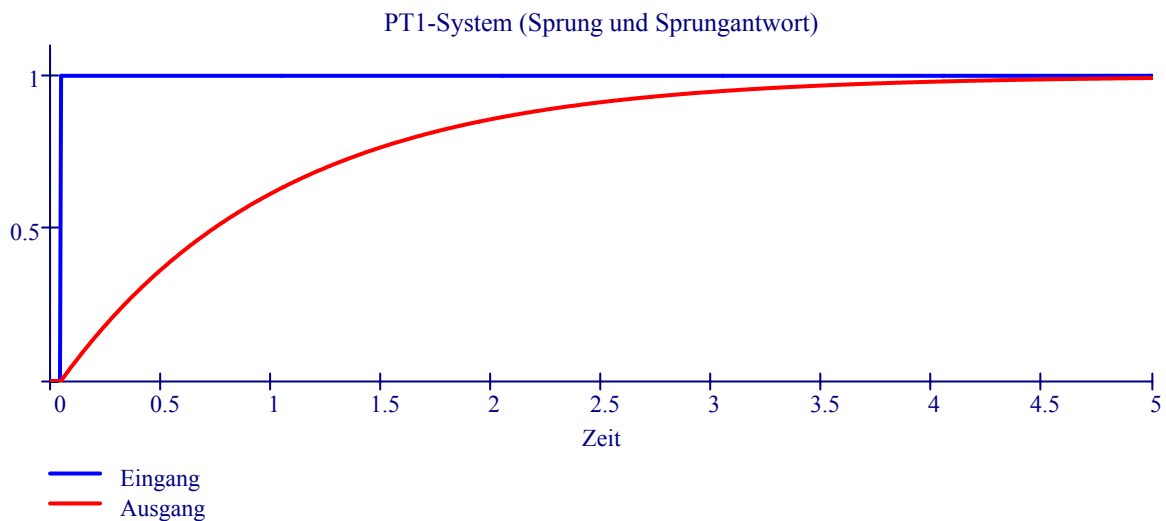
mittels Taylorverfahren (2. Ordnung) diskretisiert. Die Schrittweite der Zeitdiskretisierung Δt wird mit in die Variablenliste aufgenommen.

Durch Hintereinandersetzen mehrerer PT_1 -Glieder entstehen beliebige PT_n -Glieder. Es existiert allerdings ein eigenes Modell für ein PT_2 -Übertragungssystem.

Anwendungsbeispiel (Sprungantwort):

$$\text{Zeit} := 5 \quad N := 1000 \quad \Delta t := \frac{\text{Zeit}}{N} \quad k := 0..N \quad in_k := \Phi(k - 10)$$

```
out :=
  for j ∈ 0..N
    feld_j ← 0
  for k ∈ 0..N - 1
    feld ← PT1(1, 1, in, feld, Δt, k)
  feld
```



Weitere Simulationsmodelle mit einfachen Demonstrationen findet man hier:

▼ Weitere Simulationsmodelle

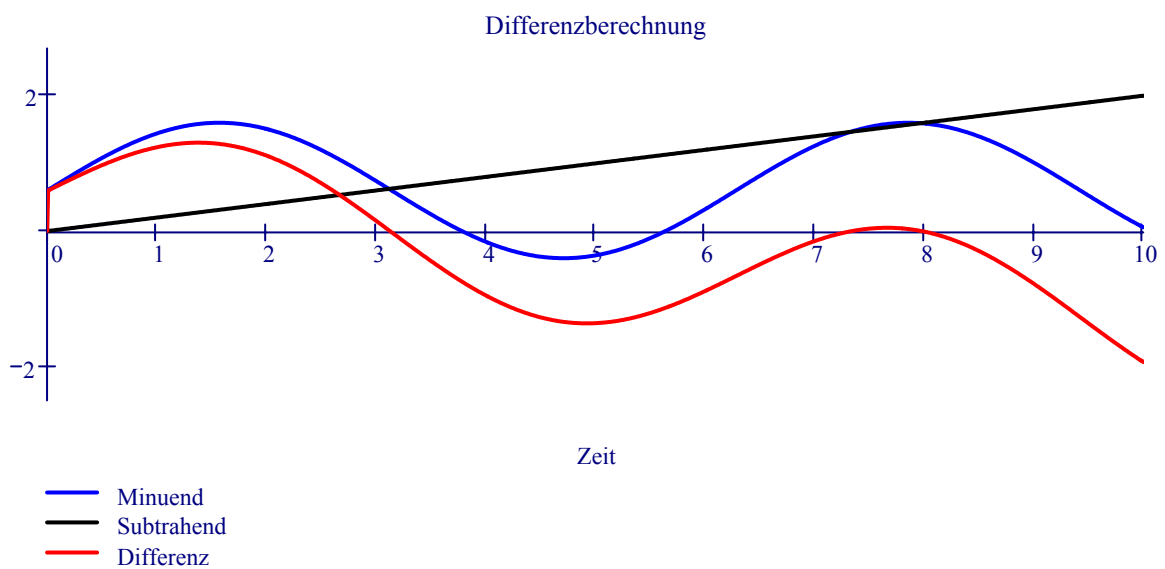
Differenzberechnung

$$\text{Diff_2}(\text{min}, \text{sub}, \text{diff}, n) := \begin{cases} \text{diff}_{n+1} \leftarrow \text{min}_n - \text{sub}_n \\ \text{diff} \end{cases}$$

$$\text{Zeit} := 10 \quad N := 1000 \quad \Delta t := \frac{\text{Zeit}}{N}$$

$$k := 0 .. N \quad \text{minu}_k := \sin(k \cdot \Delta t) + 0.6 \quad \text{sub}_k := \frac{k \cdot \Delta t}{5} \quad \text{diff}_k := 0$$

```
diff :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← Diff_2(minu, sub, feld, k)
  feld
```



PT2-System

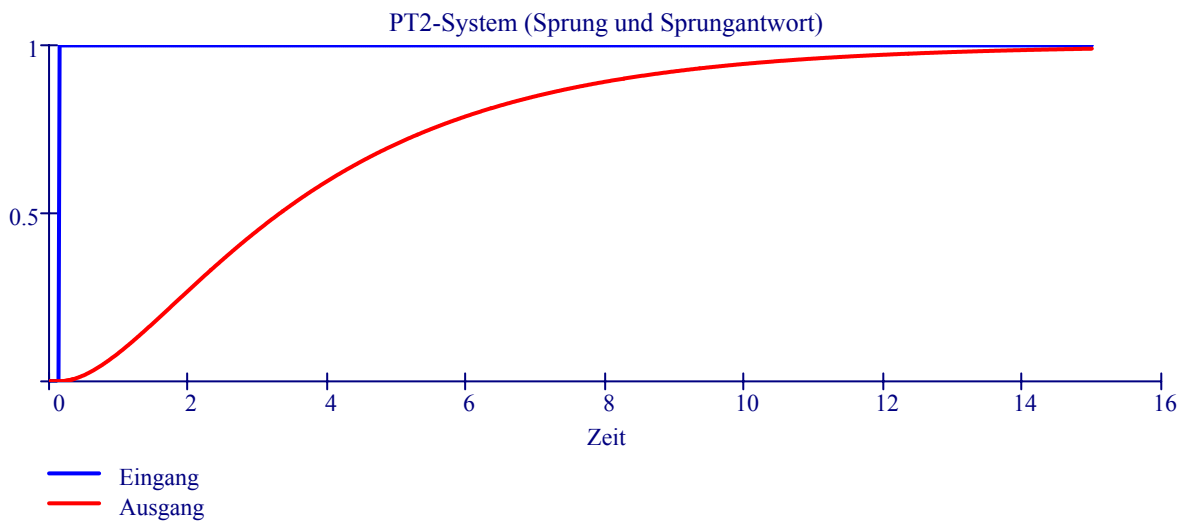
$$PT_2(K, T_1, T_2, in, out, \Delta t, n) := \begin{cases} T_{12} \leftarrow T_1 \cdot T_2 \\ out_{n+1} \leftarrow out_n + \frac{K \cdot \Delta t^2 \cdot in_n + out_n \cdot (T_{12} - \Delta t^2) - T_{12} \cdot out_{\max(n-1, 0)}}{(T_1 + T_2) \cdot \Delta t + T_{12}} \\ out \end{cases}$$

$$Zeit := 15 \quad N := 1000 \quad \Delta t := \frac{Zeit}{N} \quad k := 0 .. N \quad in_k := \Phi(k - 10)$$

```

out :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← PT2(1, 1, 3, in, feld, Δt, k)
  feld

```



Totzeitglied

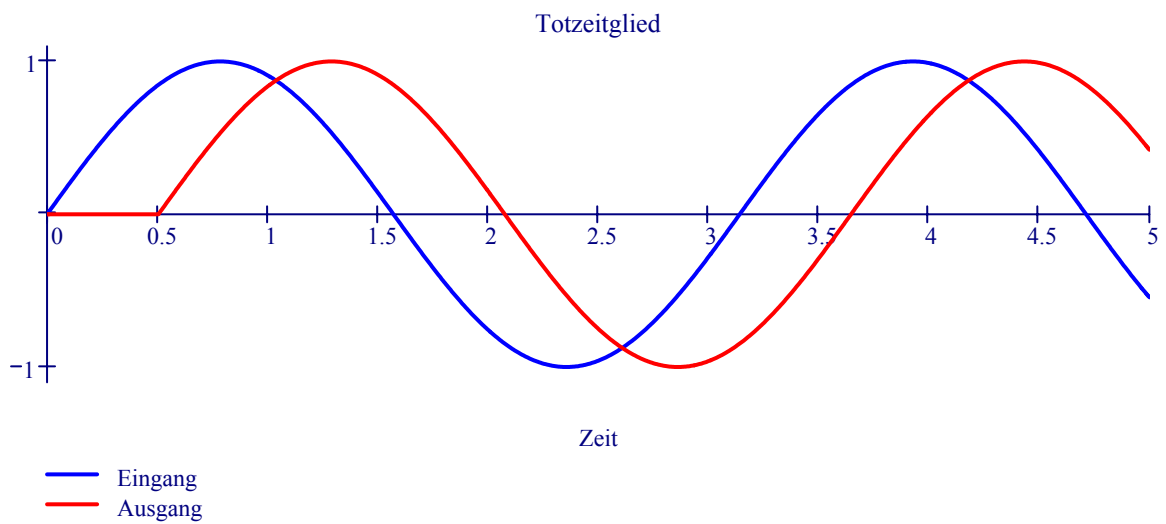
$$\text{Totzeit}(T_t, in, out, \Delta t, n) := \begin{array}{l} out_{n+1} \leftarrow in \\ \quad \quad \quad n - \min\left(n, \text{ceil}\left(\frac{T_t}{\Delta t}\right)\right) \\ out \end{array}$$

$$\text{Zeit} := 5 \quad N := 1000 \quad \Delta t := \frac{\text{Zeit}}{N} \quad k := 0 .. N \quad in_k := \sin(2 \cdot k \cdot \Delta t)$$

```

out :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← Totzeit(0.5, in, feld, Δt, k)
  feld

```



P-Regler (Proportionalverstärker)

```

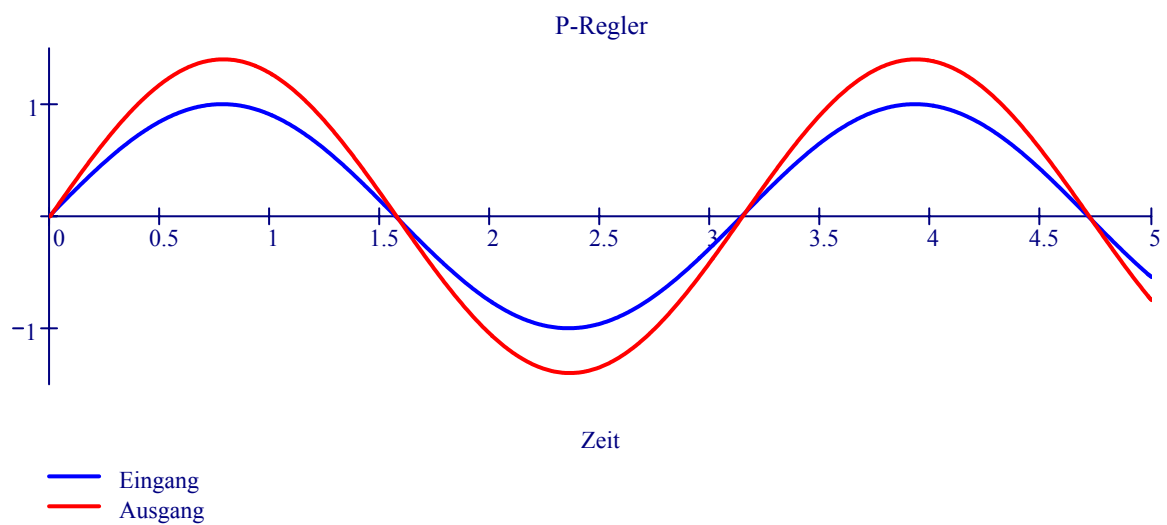
P(K, in, out, n) := | outn+1 ← K · inn
                    | out
  
```

```

Zeit := 5           N := 1000           Δt :=  $\frac{\text{Zeit}}{N}$            k := 0 .. N           ink := sin(2 · k · Δt)
  
```

```

out := | for j ∈ 0 .. N
        |   feldj ← 0
        | for k ∈ 0 .. N - 1
        |   feld ← P(1.4, in, feld, k)
        | feld
  
```

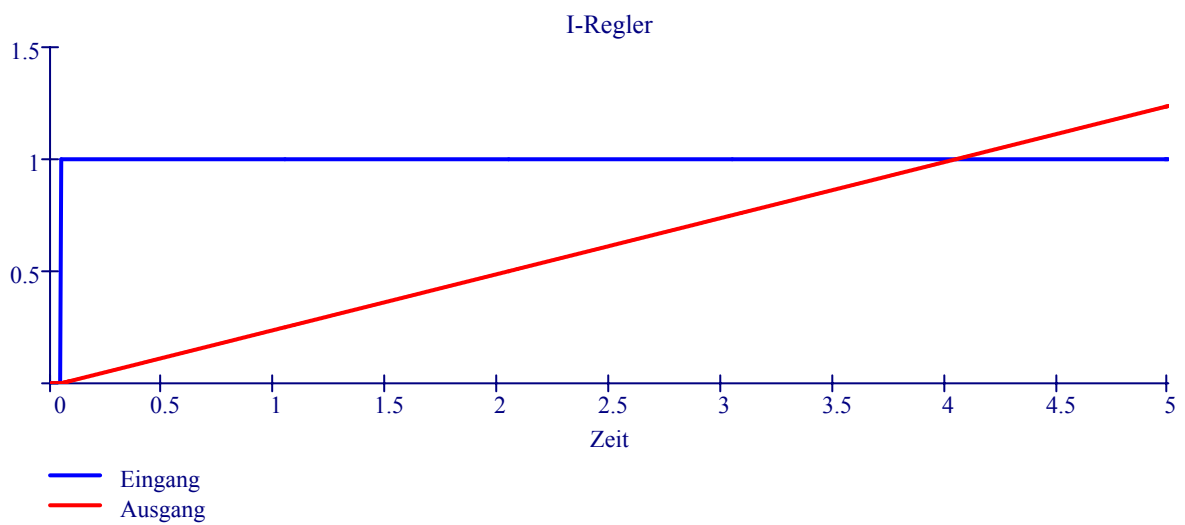


I-Regler (Integrierer)

$$I(Ti, in, out, \Delta t, n) := \begin{array}{l} out_{n+1} \leftarrow out_n + \frac{\Delta t}{Ti} \cdot in_n \\ out \end{array}$$

Zeit := 5 N := 1000 $\Delta t := \frac{Zeit}{N}$ k := 0 .. N $in_k := \Phi(k - 10)$

```
out :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← I(4, in, feld, Δt, k)
  feld
```



PI-Regler

$$PI(K, T_i, in, out, \Delta t, n) := \begin{array}{l} out_{n+1} \leftarrow out_n + K \cdot \left[in_n \cdot \left(1 + \frac{\Delta t}{T_i} \right) - in_{\max(n-1, 0)} \right] \\ out \end{array}$$

Zeit := 5

N := 1000

 $\Delta t := \frac{Zeit}{N}$

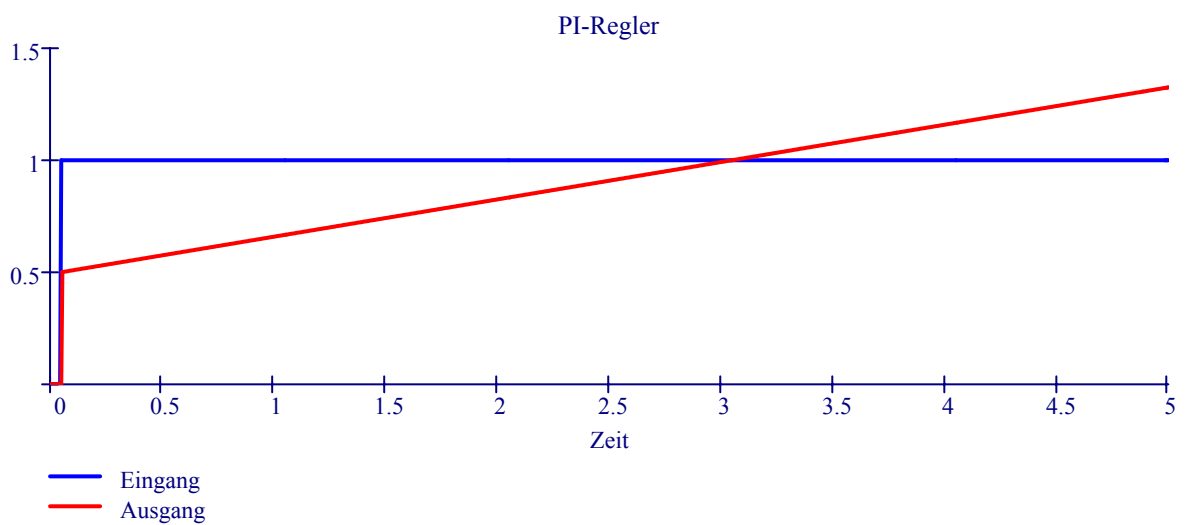
k := 0 .. N

 $in_k := \Phi(k - 10)$

```

out :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← PI(0.5, 3, in, feld, Δt, k)
  feld

```

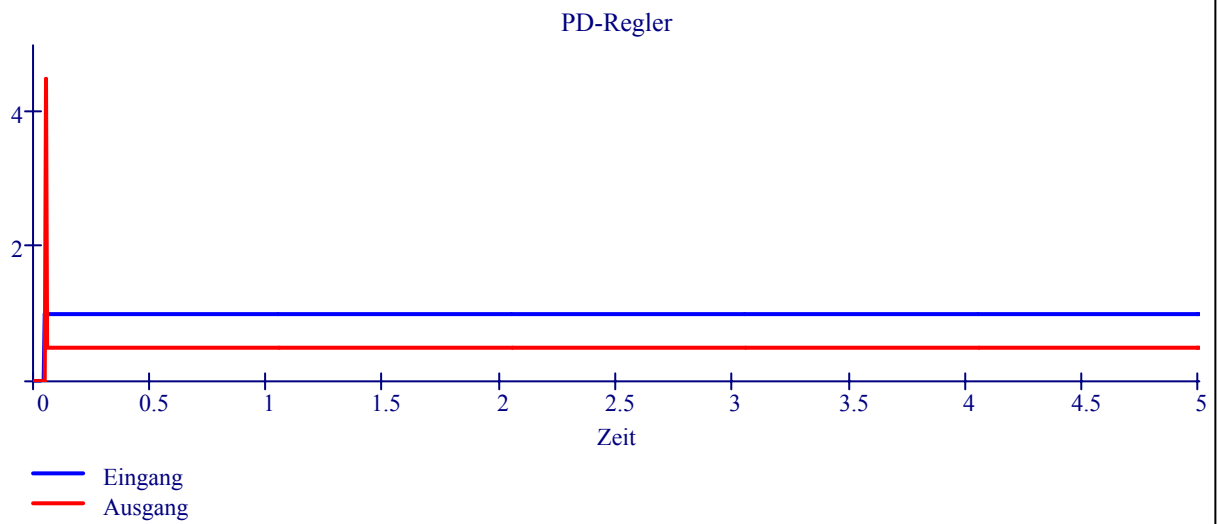


PD-Regler

$$PD(K, Td, in, out, \Delta t, n) := \left[\begin{array}{l} out_{n+1} \leftarrow K \cdot \left[in_n \cdot \left(1 + \frac{Td}{\Delta t} \right) - Td \cdot \frac{in_{\max(n-1, 0)}}{\Delta t} \right] \\ out \end{array} \right]$$

Zeit := 5 N := 1000 $\Delta t := \frac{Zeit}{N}$ k := 0 .. N $in_k := \Phi(k - 10)$

```
out :=
  for j ∈ 0 .. N
    feld_j ← 0
  for k ∈ 0 .. N - 1
    feld ← PD(0.5, .04, in, feld, Δt, k)
  feld
```



PID-Regler

$$\begin{aligned}
 PID(K, T_i, T_d, in, out, \Delta t, n) := & \quad f_1 \leftarrow 1 + \frac{\Delta t}{T_i} + \frac{T_d}{\Delta t} \\
 & \quad f_2 \leftarrow 1 + \frac{2 \cdot T_d}{\Delta t} \\
 & \quad out_{n+1} \leftarrow out_n + K \cdot \left(in_n \cdot f_1 - in_{\max(n-1, 0)} \cdot f_2 \right) \text{ if } n < 2 \\
 & \quad out_{n+1} \leftarrow out_n + K \cdot \left(in_n \cdot f_1 - in_{\max(n-1, 0)} \cdot f_2 + in_{\max(n-2, 0)} \cdot \frac{T_d}{\Delta t} \right) \text{ otherwise} \\
 & \quad out
 \end{aligned}$$

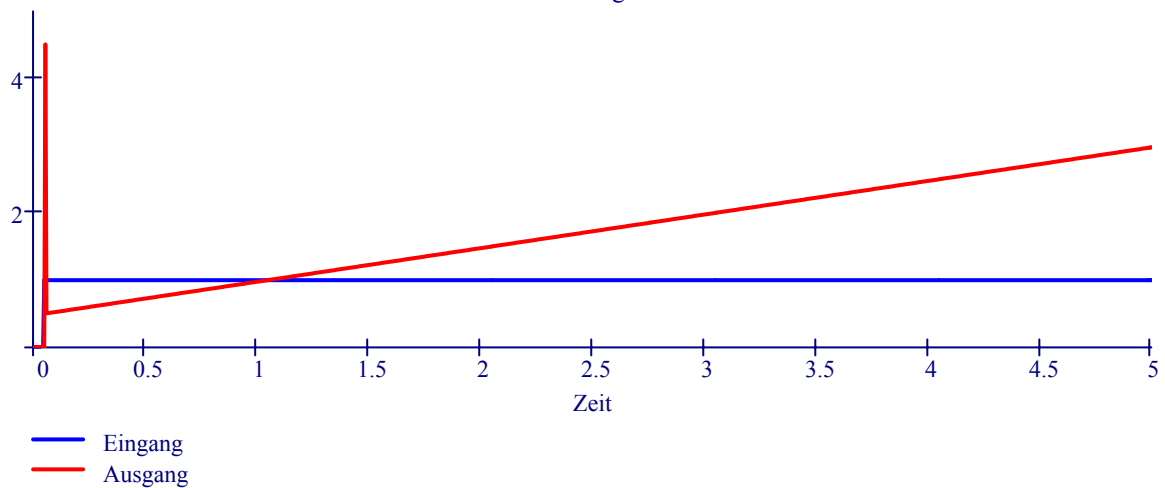
$$\begin{aligned}
 Zeit := 5 & \quad N := 1000 & \quad \Delta t := \frac{Zeit}{N} & \quad k := 0..N & \quad in_k := \Phi(k - 10)
 \end{aligned}$$

```

out :=
  for j ∈ 0..N
    feld_j ← 0
  for k ∈ 0..N-1
    feld ← PID(0.5, 1, 0.04, in, feld, Δt, k)
  feld

```

PID-Regler



Zweipunktregler (nichtinvertierend)

$$\text{Zweipunkt_ninv}(tr_on, tr_off, on, off, in, out, n) := \begin{cases} out_{n+1} \leftarrow off & \text{if } \left[(in_n \leq tr_on) \cdot (in_{n-1} < in_n) + (in_n < tr_off) \right] \\ out_{n+1} \leftarrow on & \text{if } \left[(in_n \geq tr_off) \cdot (in_n < in_{n-1}) + (in_n \geq tr_on) \right] \\ out & \end{cases}$$

$$N := 50 \quad k := 0..N \quad \Delta := 6 \quad Abstand(k) := \frac{\sqrt{k} - 0}{\sqrt{N} - 0} \cdot \Delta$$

$$in1_k := \frac{-\Delta}{2} + Abstand(k) \quad in2_k := \frac{\Delta}{2} - Abstand(k)$$

$$trigger_on := 1 \quad trigger_off := -1 \quad on := 5 \quad off := -2$$

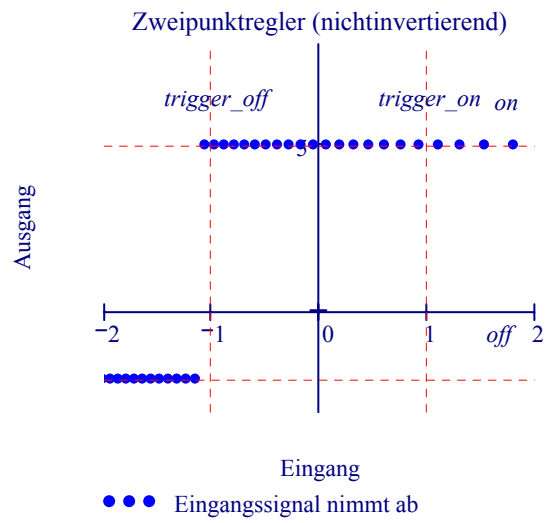
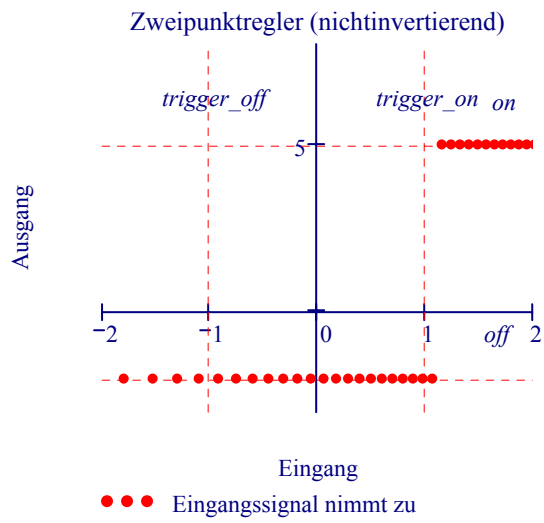
$$out1 := \begin{cases} \text{for } j \in 0..N \\ \quad feld_j \leftarrow 0 \\ \text{for } k \in 1..N-1 \\ \quad feld \leftarrow \text{Zweipunkt_ninv}(trigger_on, trigger_off, on, off, in1, feld, k) \\ \quad feld \end{cases}$$

$$out2 := \begin{cases} \text{for } j \in 0..N \\ \quad feld_j \leftarrow 0 \\ \text{for } k \in 1..N-1 \\ \quad feld \leftarrow \text{Zweipunkt_ninv}(trigger_on, trigger_off, on, off, in2, feld, k) \\ \quad feld \end{cases}$$

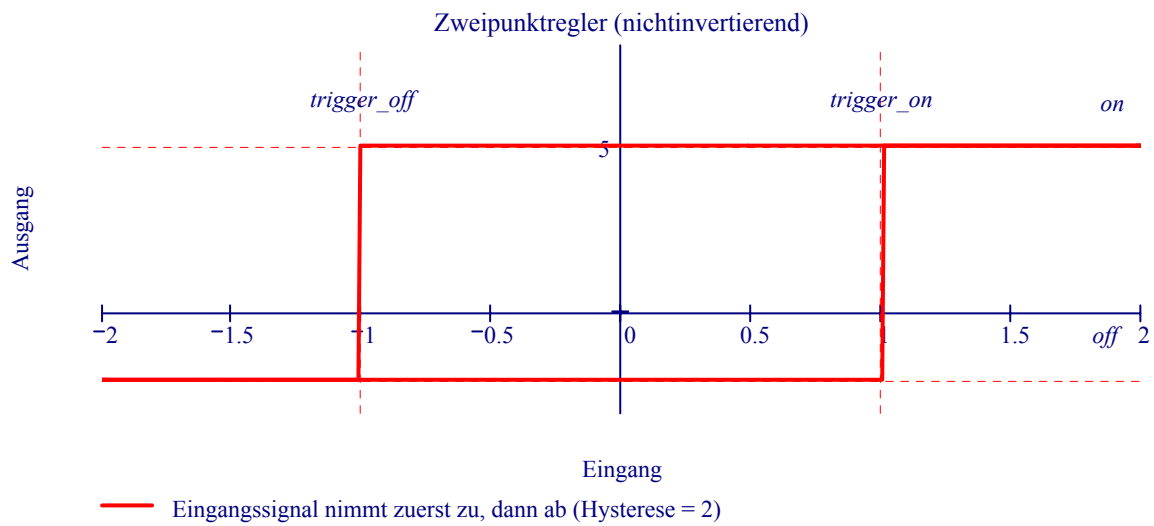
$$M := 800 \quad j := 0..M \quad in_{auf_j} := \frac{-\Delta}{2} + j \cdot \frac{\Delta}{M} \quad in_{ab_j} := \frac{\Delta}{2} - j \cdot \frac{\Delta}{M} \quad in := \text{stapeln}(in_{auf}, in_{ab})$$

$$out := \begin{cases} \text{for } j \in 0..2 \cdot M + 1 \\ \quad feld_j \leftarrow 0 \\ \text{for } k \in 1..2 \cdot M - 1 \\ \quad feld \leftarrow \text{Zweipunkt_ninv}(trigger_on, trigger_off, on, off, in, feld, k) \\ \quad feld \end{cases}$$

In den folgenden Diagrammen ändert sich das Eingangssignal jeweils in Richtung der dichter liegenden Punkte.



$$k := 0..2 \cdot M + 1$$



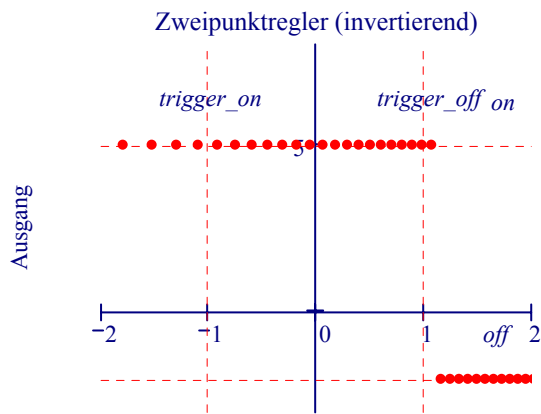
Zweipunktregler (invertierend)

$$\text{Zweipunkt_inv}(tr_on, tr_off, on, off, in, out, n) := \begin{cases} out_{n+1} \leftarrow off & \text{if } [(in_n \geq tr_on) \cdot (in_{n-1} \geq in_n) + (in_n \geq tr_off)] \\ out_{n+1} \leftarrow on & \text{if } [(in_n < tr_off) \cdot (in_n > in_{n-1}) + (in_n < tr_on)] \\ out & \end{cases}$$

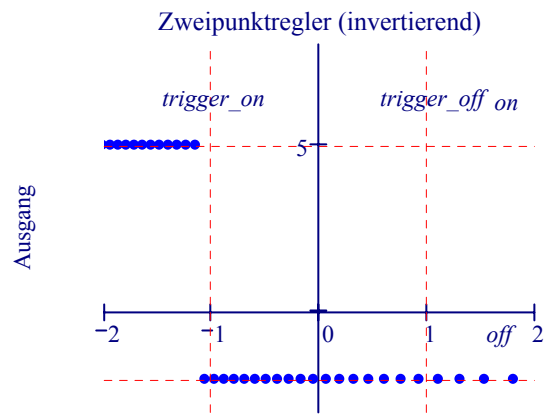
$trigger_on := -1$ $trigger_off := 1$ $on := 5$ $off := -2$

```
out1 := | for j ∈ 0..N
        |   feld_j ← 0
        | for k ∈ 1..N-1
        |   feld ← Zweipunkt_inv(trigger_on, trigger_off, on, off, in1, feld, k)
        | feld
```

```
out2 := | for j ∈ 0..N
        |   feld_j ← 0
        | for k ∈ 1..N-1
        |   feld ← Zweipunkt_inv(trigger_on, trigger_off, on, off, in2, feld, k)
        | feld
```



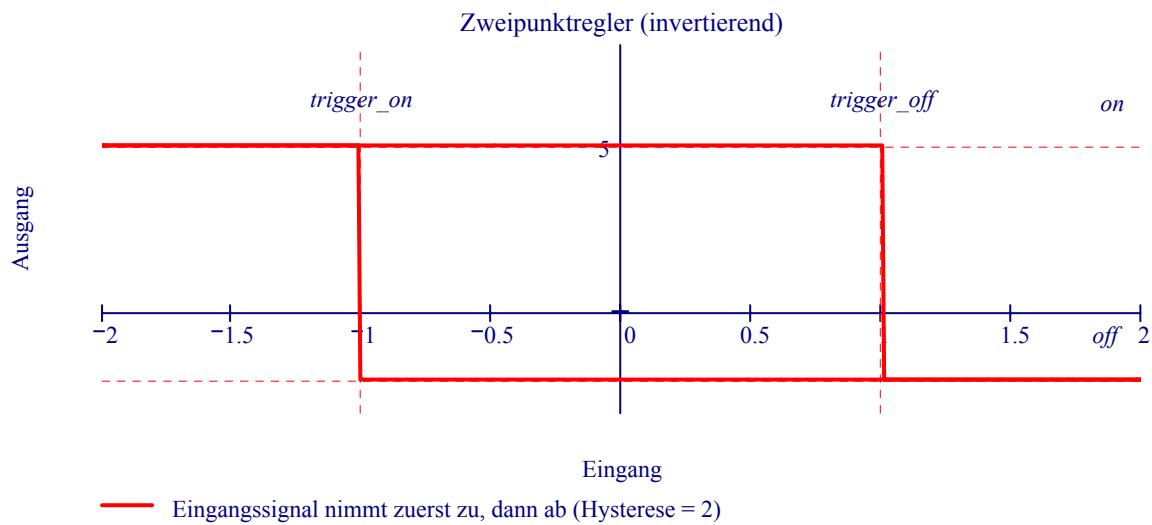
Eingang
●●● Eingangssignal nimmt zu



Eingang
●●● Eingangssignal nimmt ab

```
out := | for j ∈ 0..2 · M + 1
        |   feld_j ← 0
        | for k ∈ 1..2 · M - 1
        |   feld ← Zweipunkt_inv(trigger_on, trigger_off, on, off, in, feld, k)
        | feld
```

$$k := 0..2 \cdot M + 1$$



▲ Weitere Simulationsmodelle

Simulationsbeispiele

Bemerkung:

Bei den vorgestellten Regelungsbeispielen geht es nur um die Demonstration der Vorgangsweise bei der Simulation. Die Reglerparameter wurden "gefühlsmäßig" gewählt und nicht nach irgendeinem Verfahren berechnet oder gar optimiert.

Beispiel 1: [zum Anfang](#) [Simulationsmodelle](#) [zum Beispiel 2](#)

Standardregelkreis, bestehend aus PT₁-Strecke und PI-Regler
 Berechnet wird das Regelkreisverhalten bei Sprung der Führungsgröße $w(t)$

Gesamtzeitintervall: Zeitintervall := 20

Zahl der Iterationen: Iterationen := 3000

Rechenschrittweite: $\Delta t := \frac{\text{Zeitintervall}}{\text{Iterationen}}$ $\Delta t = 6.667 \times 10^{-3}$

PT₁-Strecke: K_S := 1 T_I := 4

PI-Regler: K_R := 2 T_i := 2

Definition der im Regelkreis vorkommenden Signale in Form von Vektoren:

$$k := 0..Iterationen$$

Führungsgröße: $wI_k := \Phi\left(k - \text{floor}\left(\frac{1}{\Delta t}\right)\right)$

Regelgröße: $xI_k := 0$

Regeldifferenz: $xdI_k := 0$

Stellgröße: $yI_k := 0$

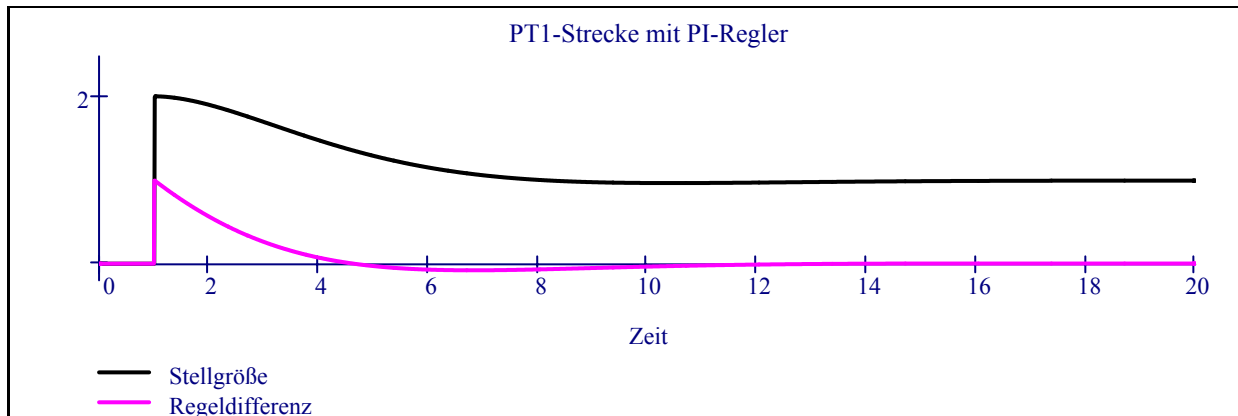
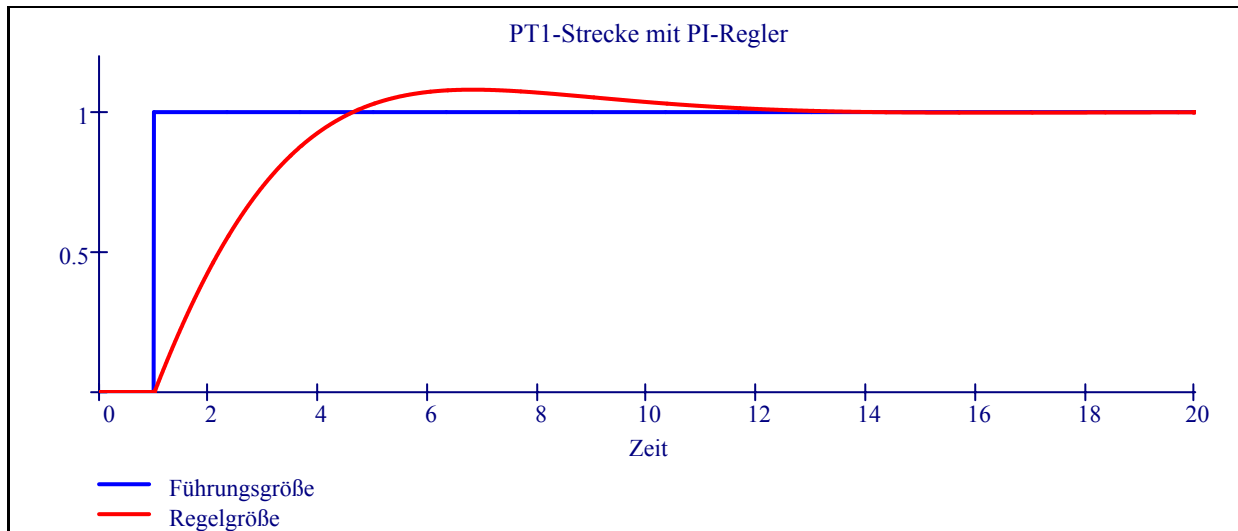
Zeitvektor: $tI_k := k \cdot \Delta t$

Die Funktion *Regelkreis* beinhaltet die Schleifenanweisung für die iterative Berechnung der Vektorkomponenten. Jedes Regelkreisglied wird durch eine Programmzeile beschrieben, die aus dem Eingangsvektor (den Eingangsvektoren) den Vektor des Ausgangssignals erzeugt. Die Reihenfolge der Regelkreisglieder in der Schleife ist belanglos.

$$\text{Regelkreis}(wI, xdl, xI, yI) := \left| \begin{array}{l} \text{for } n \in 1 \dots \text{Iterationen} - 1 \\ \quad xdl \leftarrow \text{Diff_2}(wI, xI, xdl, n) \\ \quad yI \leftarrow \text{PI}(K_R, T_i, xdl, yI, \Delta t, n) \\ \quad xI \leftarrow \text{PTI}(K_S, T_I, yI, xI, \Delta t, n) \\ \text{erweitern}(wI, xdl, xI, yI) \end{array} \right.$$

Regelmatrix := *Regelkreis*(*wI*, *xdl*, *xI*, *yI*)

Die Matrix *Regelmatrix* enthält nun nebeneinander die einzelnen interessierenden Signalvektoren, die als Spalten extrahiert werden müssen:

$w1 := \text{Regelmatrix}^{(0)}$ $x1 := \text{Regelmatrix}^{(2)}$ $xd1 := \text{Regelmatrix}^{(1)}$ $y1 := \text{Regelmatrix}^{(3)}$ 

Die exakte Berechnung der Regelgröße über die Laplace-Transformation und den Vergleich mit der Näherungslösung von vorher zeigt der folgende kurze Abschnitt. Man beachte, dass dabei ausschließlich die Regelgröße berechnet wird (für die anderen Signale sind natürlich Rechnungen mit entsprechenden Übertragungsfunktionen ebenfalls möglich).

► Vergleich mit Laplace

Beispiel 2:zum Beispiel 1zum Beispiel 3Standardregelkreis, bestehend aus PT_2 -Strecke und PI-ReglerBerechnet wird das Regelkreisverhalten bei Sprung der Führungsgröße $w(t)$

- ohne zusätzlichen Störgrößeneinfluss
- bei zusätzlichem rechteckigem Eingriff der Störgröße $z(t)$

Gesamtzeitintervall: **Zeitintervall := 35**Zahl der Iterationen: **Iterationen := 5000**Rechenschrittweite: $\Delta t := \frac{\text{Zeitintervall}}{\text{Iterationen}}$ $\Delta t = 7 \times 10^{-3}$ **PT₂-Strecke:** **$K_{S1} := 1$** **$T_1 := 2$** **$K_{S2} := 1$** **$T_2 := 3$** **PI-Regler:** **$K_R := 1.5$** **$T_i := 4$**

Definition der im Regelkreis vorkommenden Signale in Form von Vektoren:

 $k := 0 \dots \text{Iterationen}$ Führungsgröße: $w2_k := \Phi\left(k - \text{floor}\left(\frac{1}{\Delta t}\right)\right)$ Regelgröße: $x2_k := 0$ Regeldifferenz: $xd2_k := 0$ Zusätzliches Signal zwischen den beiden Streckenteilen: $temp2_k := 0$ Störgröße: $z2_k := 0$ Stellgröße: $y2_k := 0$ Eingangssignal der Strecke: $yy2_k := 0$ Zeitvektor: $t2_k := k \cdot \Delta t$

```

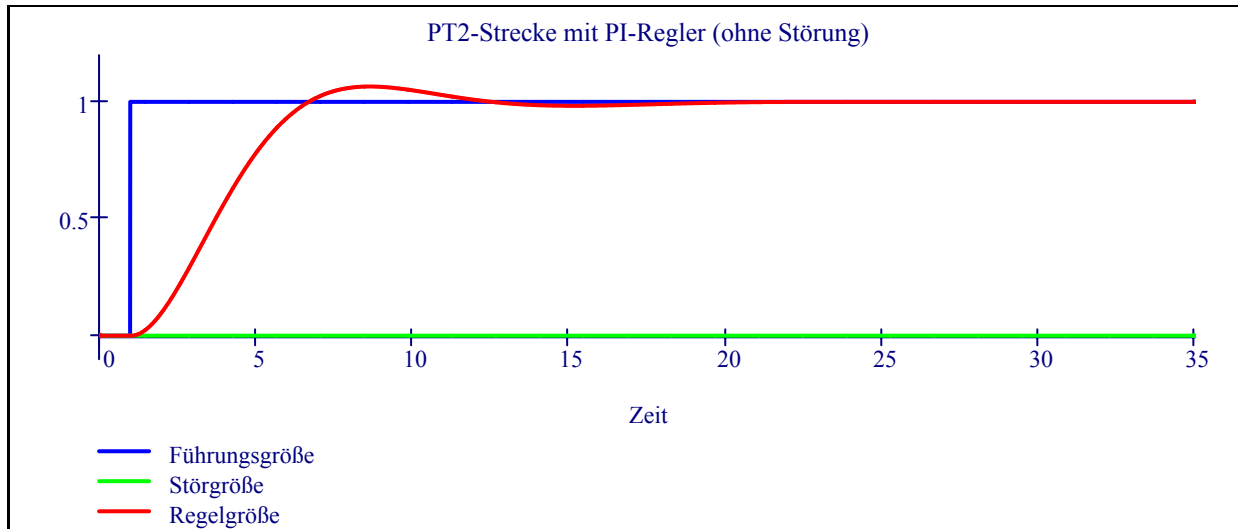
Regelkreis(w2,xd2,z2,x2,y2,yy2,temp2) :=
  for n ∈ 1 .. Iterationen - 1
    xd2 ← Diff_2(w2,x2,xd2,n)
    y2 ← PI(K_R,T_i,xd2,y2,Δt,n)
    yy2 ← Sum_2(y2,z2,yy2,n)
    temp2 ← PT1(K_S1,T_1,yy2,temp2,Δt,n)
    x2 ← PT1(K_S2,T_2,temp2,x2,Δt,n)
  erweitern(w2,xd2,z2,x2,y2,yy2,temp2)

```

$Regelmatrix := Regelkreis(w2, xd2, z2, x2, y2, yy2, temp2)$

$w2 := Regelmatrix^{(0)}$ $z2 := Regelmatrix^{(2)}$ $y2 := Regelmatrix^{(4)}$

$xd2 := Regelmatrix^{(1)}$ $x2 := Regelmatrix^{(3)}$



Bei den verwendeten Reglerparametern zeigt sich ein Überschwingen der Regelgröße über den Sollwert von etwa 6% und eine Einschwingdauer von insgesamt ungefähr 20.

Nun soll zusätzlich zum Führungssprung ein etwas später auftretender rechteckiger Einfluss der Störgröße $z(t)$ am Streckeneingang berücksichtigt werden:

$k := 0 \dots \text{Iterationen}$

Führungsgröße: $w2_k := \Phi\left(k - \text{floor}\left(\frac{1}{\Delta t}\right)\right)$

Regelgröße: $x2_k := 0$

Regeldifferenz: $xd2_k := 0$

Zusätzliche Größe zwischen den beiden Streckenteilen: $temp_k := 0$

Störgröße: $z2_k := \frac{1}{2} \cdot \left(\Phi\left(k - \text{floor}\left(\frac{15}{\Delta t}\right)\right) - \Phi\left(k - \text{floor}\left(\frac{20}{\Delta t}\right)\right) \right)$

Stellgröße: $y2_k := 0$

Eingangssignal der Strecke: $yy2_k := 0$

Zeitvektor: $t2_k := k \cdot \Delta t$

```

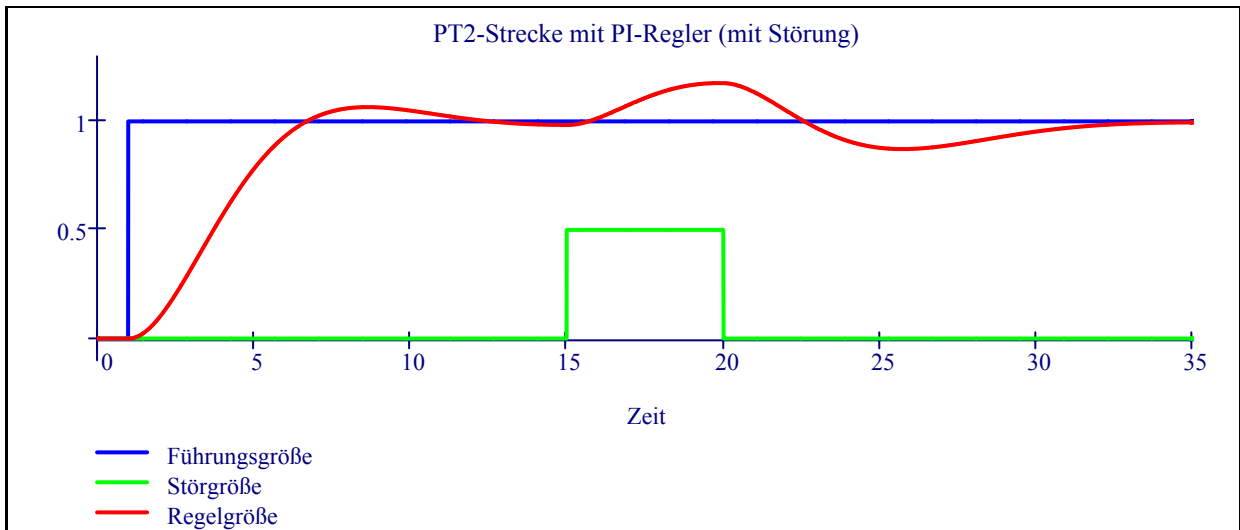
Regelkreis(w2,xd2,z2,x2,y2,yy2,temp2) :=
  for n ∈ 1 .. Iterationen - 1
    |
    |   xd2 ← Diff_2(w2,x2,xd2,n)
    |   y2 ← PI(KR,Ti,xd2,y2,Δt,n)
    |   yy2 ← Sum_2(y2,z2,yy2,n)
    |   temp2 ← PT1(KS1,T1,yy2,temp2,Δt,n)
    |   x2 ← PT1(KS2,T2,temp2,x2,Δt,n)
    |   erweitern(w2,xd2,z2,x2,y2,yy2,temp2)
  
```

```

Regelmatrix := Regelkreis(w2,xd2,z2,x2,y2,yy2,temp2)
  
```

```

w2 := Regelmatrix<0>           z2 := Regelmatrix<2>           y2 := Regelmatrix<4>
xd2 := Regelmatrix<1>        x2 := Regelmatrix<3>
  
```



Knapp bevor sich die Regelgröße der Führungsgröße angepasst hat wird sie durch eine Störung kurzzeitig "aus der Bahn geworfen". Der Störgrößeneinfluss kann jedoch vollständig ausgeregelt werden. Die dafür benötigte Zeit ist ungefähr gleich lang wie bei der Anpassung an die Führungsgröße ohne Störung.

Beispiel 3: Raumheizung[zum Beispiel 2](#)[zum Beispiel 4](#)

Standardregelkreis, bestehend aus PT_1 - T_t -Strecke (PT_1 -Strecke mit Totzeit T_t) und Schaltregler mit Hysterese
Berechnet wird das Regelkreisverhalten bei Sprung der Führungsgröße

Gesamtzeitintervall: **Zeitintervall := 15**Zahl der Iterationen: **Iterationen := 3000**Rechenschrittweite: $\Delta t := \frac{\text{Zeitintervall}}{\text{Iterationen}}$ $\Delta t = 5 \times 10^{-3}$ **PT_1 -Strecke mit Totzeit:** **$K_S := 1$** **$T_I := 3$** **$T_t := 0.2$** **Schaltregler (Zweipunktregler mit Hysterese):**Schaltschwellen des Reglers: **tron := 0.05** **troff := -0.05**Schaltzustände des Reglers: **ein := 2** **aus := 0**

Definition der im Regelkreis vorkommenden Signale in Form von Vektoren:

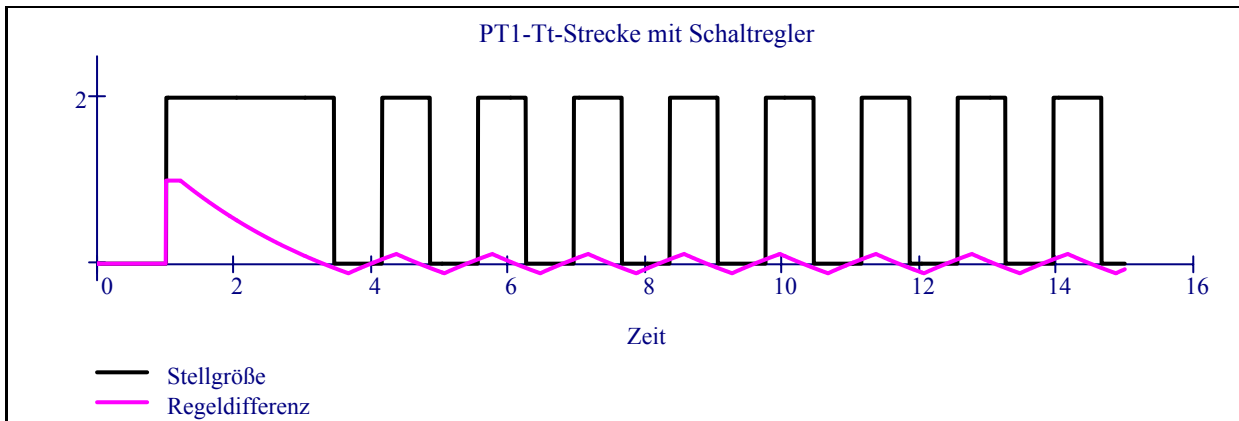
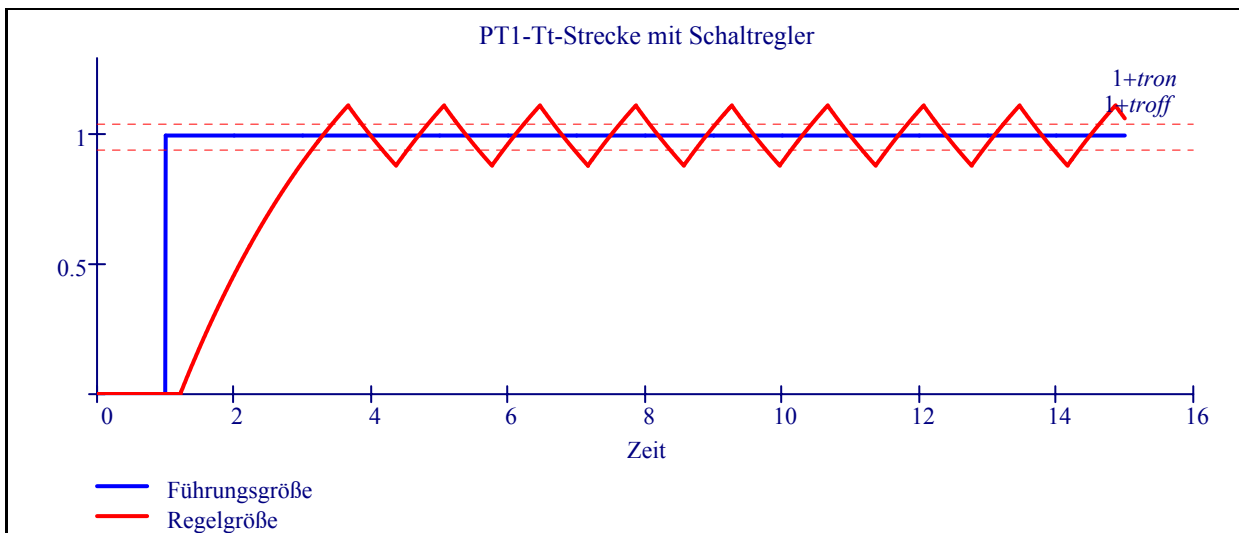
 $k := 0 \dots \text{Iterationen}$ Führungsgröße: $w3_k := \Phi \left(k - \text{floor} \left(\frac{1}{\Delta t} \right) \right)$ Regelgröße: $x3_k := 0$ Regeldifferenz: $xd3_k := 0$ Stellgröße: $y3_k := 0$ Zusätzliche Größe zwischen den beiden Streckenteilen: $temp3_k := 0$ Zeitvektor: $t3_k := k \cdot \Delta t$

```

Regelkreis(w3,xd3,x3,y3,temp3) := for n ∈ 1 .. Iterationen - 1
    |
    |   xd3 ← Diff_2(w3,x3,xd3,n)
    |   y3 ← Zweipunkt_ninv(tron,troff,ein,aus,xd3,y3,n)
    |   temp3 ← PT1(1,TI,y3,temp3,Δt,n)
    |   x3 ← Totzeit(Tt,temp3,x3,Δt,n)
    |
    |   erweitern(w3,xd3,x3,y3)

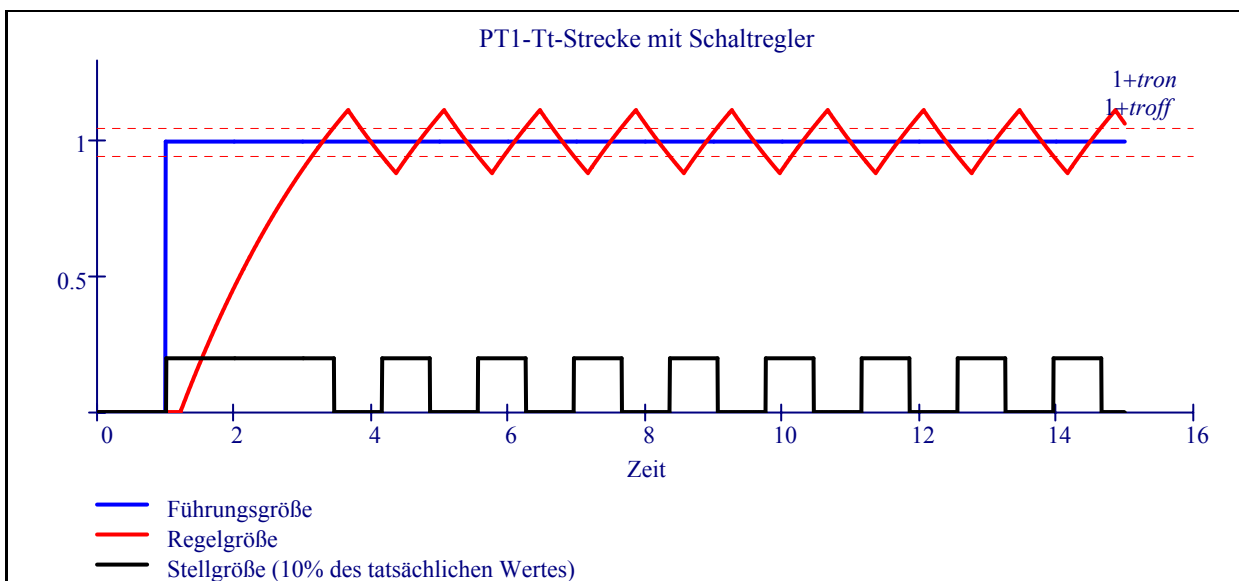
```

 $\text{Regelmatrix} := \text{Regelkreis}(w3,xd3,x3,y3,temp3)$ $w3 := \text{Regelmatrix}^{\langle 0 \rangle}$ $x3 := \text{Regelmatrix}^{\langle 2 \rangle}$ $xd3 := \text{Regelmatrix}^{\langle 1 \rangle}$ $y3 := \text{Regelmatrix}^{\langle 3 \rangle}$



Die Regelgröße führt die erwartete Schwingbewegung ("Arbeitsbewegung") um den Sollwert aus, wobei die Reaktionen wegen der Totzeit verzögert sind. Die möglichen Stellgrößenwerte sind 2 ("volle Heizleistung" - damit ließe sich asymptotisch eine Regelgröße 2 erreichen) und 0 ("Heizung aus").

Um den zeitlichen Zusammenhang zwischen Arbeitsbewegung und Stellgrößenverlauf besser demonstrieren zu können, werden Führungsgröße, Regelgröße und Stellgröße ($y/10$) in einem gemeinsamen Diagramm dargestellt. Wegen der Totzeit steigt die Regelgröße - obwohl der Regler bereits abgeschaltet hat - noch weiter an. Auch nach dem Wieder-Einschalten etwas später sinkt die Regelgröße noch eine Zeit T_t lang weiter.



Um die Amplitude der Arbeitsbewegung zu verringern gibt es - bei vorgegebener Strecke - verschiedene Möglichkeiten, unter anderem **Schalten mit Grundlast**. Dabei wird die Heizleistung nicht vollständig abgeschaltet, sondern nur zurückgenommen. Die Heizleistung für den EIN-Zustand wird verringert. Die Regelung wird durch diese Maßnahmen allerdings träger.

Schaltregler (Zweipunktregler mit Hysterese):

Schaltsschwellen des Reglers: $tron := 0.05$ $troff := -0.05$
 Schaltzustände des Reglers: $ein := 1.2$ $aus := 0.8$

Definition der im Regelkreis vorkommenden Signale in Form von Vektoren:

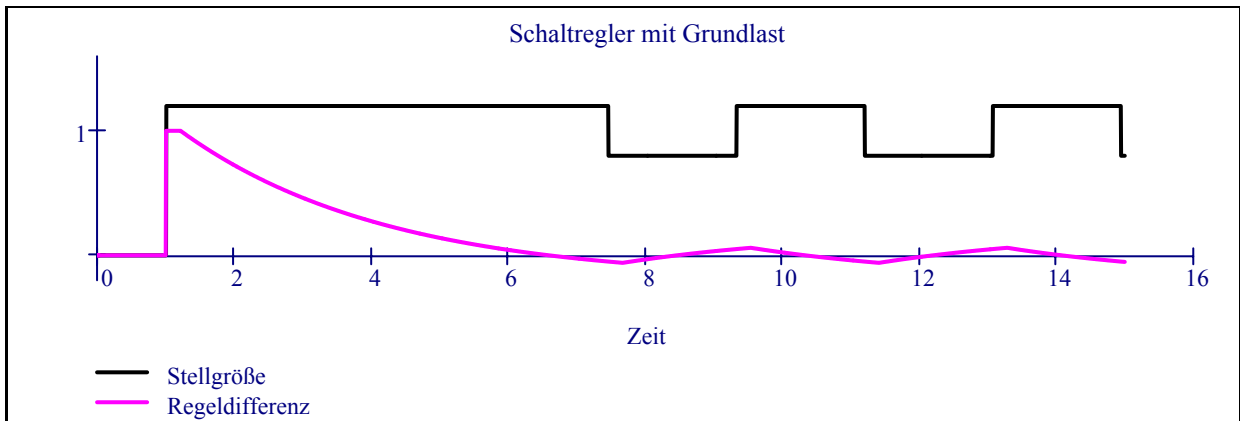
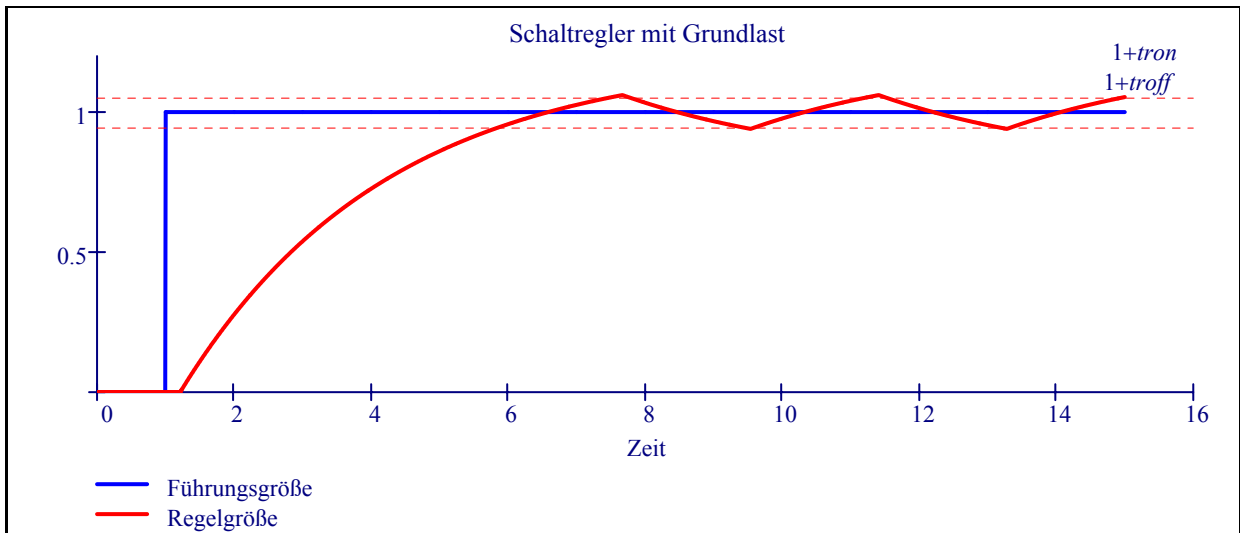
$k := 0 .. Iterationen$

Führungsgröße: $w3_k := \Phi\left(k - \text{floor}\left(\frac{1}{\Delta t}\right)\right)$
 Regelgröße: $x3_k := 0$
 Regeldifferenz: $xd3_k := 0$
 Stellgröße: $y3_k := 0$
 Zusätzliche Größe zwischen den beiden Streckenteilen: $temp3_k := 0$
 Zeitvektor: $t3_k := k \cdot \Delta t$

$Regelkreis(w3, xd3, x3, y3, temp3) :=$ $\left\{ \begin{array}{l} \text{for } n \in 1 .. Iterationen - 1 \\ \quad xd3 \leftarrow \text{Diff_2}(w3, x3, xd3, n) \\ \quad y3 \leftarrow \text{Zweipunkt_niv}(tron, troff, ein, aus, xd3, y3, n) \\ \quad temp3 \leftarrow \text{PT1}(1, T1, y3, temp3, \Delta t, n) \\ \quad x3 \leftarrow \text{Totzeit}(Tt, temp3, x3, \Delta t, n) \\ \text{erweitern}(w3, xd3, x3, y3) \end{array} \right.$

$Regelmatrix := Regelkreis(w3, xd3, x3, y3, temp3)$

$w3 := Regelmatrix^{(0)}$ $x3 := Regelmatrix^{(2)}$
 $xd3 := Regelmatrix^{(1)}$ $y3 := Regelmatrix^{(3)}$



Beispiel 4: Zweischleifiger Regelkreis**zum Beispiel 3**

Der Regelkreis besteht aus einer PT_2 -Strecke und einem Schaltregler mit verzögerter Rückführung

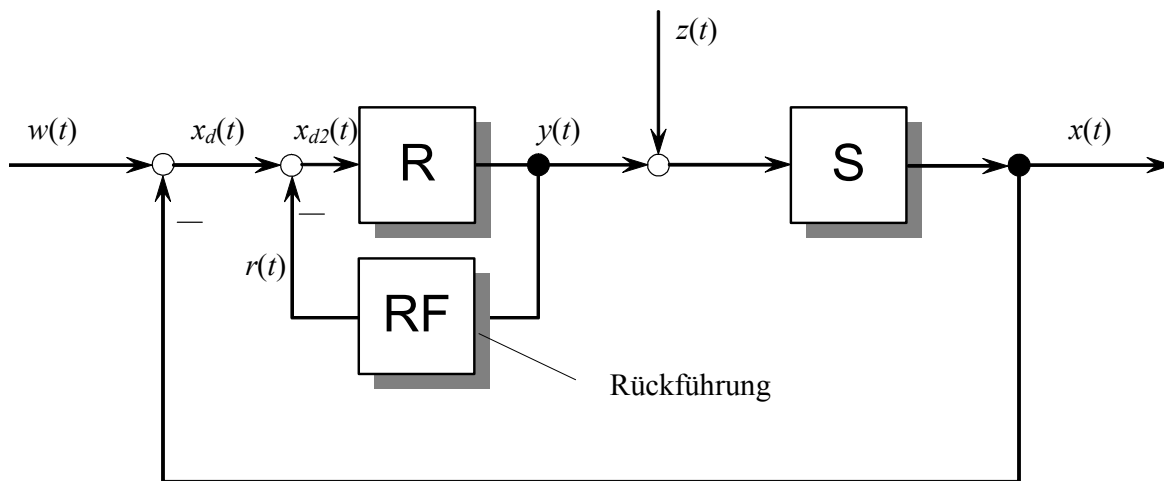
Das Stellgrößensignal wird über ein passendes Übertragungsglied RF (in unserem Fall PT_1) auf den Reglereingang zurückgekoppelt.

Wegen $x_{d2}(t) = w(t) - x(t) - r(t) = w(t) - (x(t) + r(t))$ wird dem Regler ein Regelgrößenwert vorgespielt, der um $r(t)$ größer ist als der tatsächliche. Das veranlasst den Regler, früher umzuschalten, was wiederum eine Verringerung der Amplitude der Arbeitsbewegung zur Folge hat. Die PT_1 -Rückführung muss dazu deutlich schneller reagieren als die Strecke selbst. Wie aus dem Verlauf der Regelgröße unten hervorgeht, ergibt sich unter günstigen Bedingungen ein Regelgrößenverlauf, welchen man sich eher von einem Regelkreis mit stetigem Regler statt einem Schaltregler erwarten würde ("quasistetige Regelung").

Als nachteilig zu vermerken sind:

- die drastische Erhöhung der Schaltfrequenz des Reglers. Der sanfte Verlauf der Regelgröße kommt natürlich durch die fein dosierte Energiezufuhr (wenn man etwa an eine Heizungsregelung denkt) zustande.
- die bleibende Regeldifferenz, die durch die Einseitigkeit der Rückführung bedingt ist. Bei zweiseitiger Rückführung kann man eine bleibende Regeldifferenz von 0 erreichen.

Berechnet wird das Regelkreisverhalten bei Sprung der Führungsgröße



Gesamtzeitintervall: **Zeitintervall := 5**

Zahl der Iterationen: **Iterationen := 3000**

Rechenschrittweite: $\Delta t := \frac{\text{Zeitintervall}}{\text{Iterationen}}$ $\Delta t = 1.667 \times 10^{-3}$

PT_2 -Strecke: **$K_S := 1$** **$T_I := 1$** **$T_2 := 1$**

Schaltregler (Zweipunktregler mit Hysterese):

Schalterswellen des Reglers: **$tron := 0.05$** **$troff := -0.05$**

Schaltzustände des Reglers: **$ein := 2$** **$aus := 0$**

PT_1 -Rückführung: **$K_{rück} := 0.1$** **$T_{rück} := 0.1$**

Definition der im Regelkreis vorkommenden Signale in Form von Vektoren:

$k := 0 \dots \text{Iterationen}$

Führungsgröße:	$w_k^4 := \Phi \left(k - \text{floor} \left(\frac{.1}{\Delta t} \right) \right)$
Regelgröße:	$x_k^4 := 0$
Regeldifferenz:	$xd_k^4 := 0$
Reglereingangssignal:	$xxd_k^4 := 0$
Rückführsignal:	$r_k^4 := 0$
Stellgröße:	$y_k^4 := 0$
Zusätzliche Größe zwischen den beiden Streckenteilen:	$temp_k^4 := 0$
Zeitvektor:	$t_k^4 := k \cdot \Delta t$

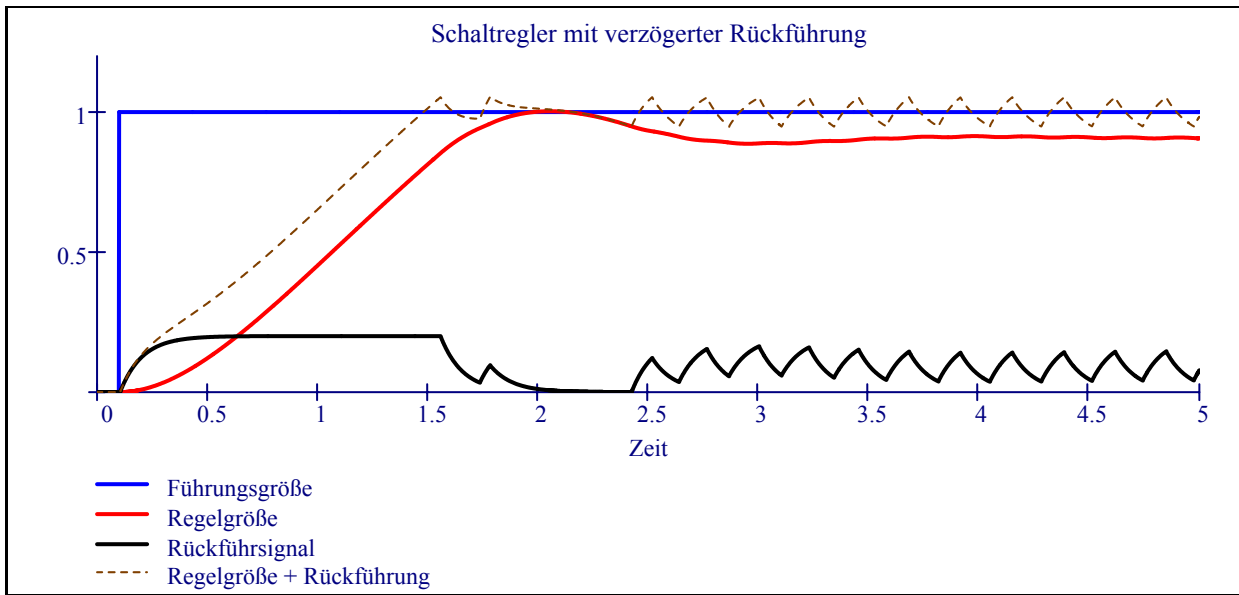
Regelkreis($w4, xd4, xxd4, r4, x4, y4, temp4$) := $\left| \begin{array}{l} \text{for } n \in 1 \dots \text{Iterationen} - 1 \\ \quad \left| \begin{array}{l} xd4 \leftarrow \text{Diff_2}(w4, x4, xd4, n) \\ xxd4 \leftarrow \text{Diff_2}(xd4, r4, xxd4, n) \\ r4 \leftarrow \text{PT}_I(K_{\text{rück}}, T_{\text{rück}}, y4, r4, \Delta t, n) \\ y4 \leftarrow \text{Zweipunkt_niv}(\text{tron}, \text{troff}, \text{ein}, \text{aus}, xxd4, y4, n) \\ temp4 \leftarrow \text{PT}_I(1, T_1, y4, temp4, \Delta t, n) \\ x4 \leftarrow \text{PT}_I(1, T_2, temp4, x4, \Delta t, n) \end{array} \right. \\ \text{erweitern}(w4, xd4, xxd4, r4, x4, y4) \end{array} \right.$

Regelmatrix := Regelkreis($w4, xd4, xxd4, r4, x4, y4, temp4$)

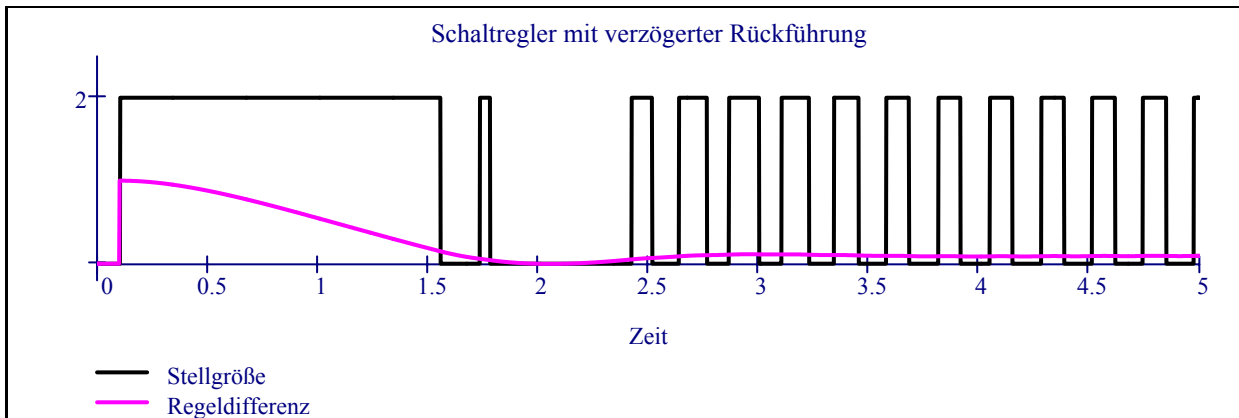
$w4 := \text{Regelmatrix}^{\langle 0 \rangle}$ $r4 := \text{Regelmatrix}^{\langle 3 \rangle}$

$xd4 := \text{Regelmatrix}^{\langle 1 \rangle}$ $x4 := \text{Regelmatrix}^{\langle 4 \rangle}$

$xxd4 := \text{Regelmatrix}^{\langle 2 \rangle}$ $y4 := \text{Regelmatrix}^{\langle 5 \rangle}$



Die Arbeitsbewegung wird in diesem Fall vom Summensignal (Regelgröße + Rückführung) ausgeführt. Das ist verständlich, da ja dem Regler genau dieses Signal als Regelgröße vorgetäuscht wird.



[zum Anfang](#) [zum Beispiel 4](#)