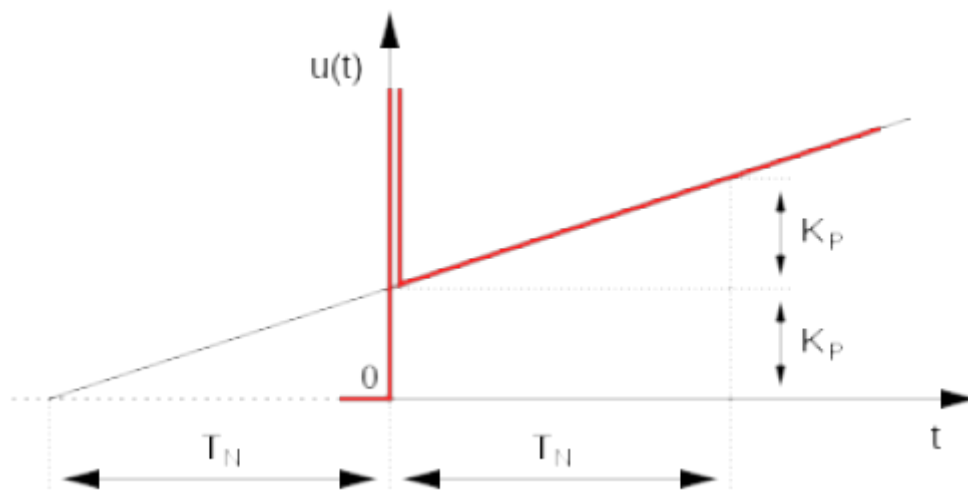


Anwendung der iCon-L PID-Regler

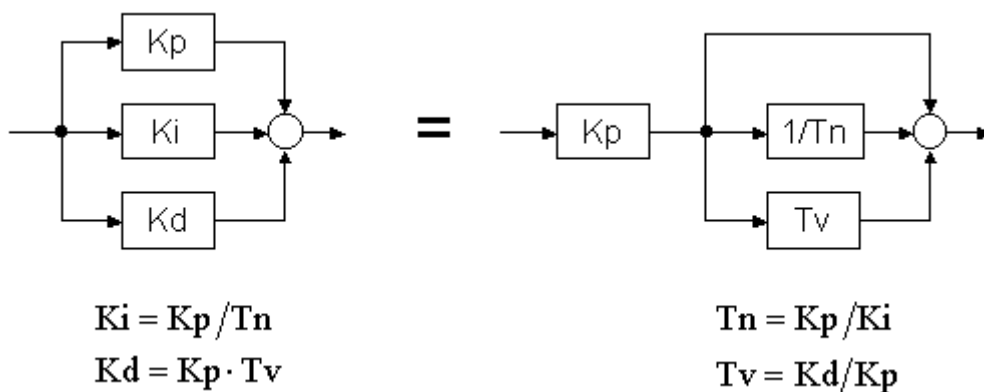
Ein wenig Theorie

Um die iCon-L PID-Regler besser zu verstehen, sollen an dieser Stelle ein paar theoretische Betrachtungen angestellt werden

Reglerstrukturen (Darstellungsmöglichkeiten)



Es gibt 2 Darstellungsmöglichkeiten, die inhaltlich identisch sind. In der analogen Regeltechnik ist noch die Darstellung mit der Nachstellzeit T_n und der Vorhaltezeit T_v üblich. Die Umrechnung zwischen den beiden Strukturen ist mit den angegebenen Formeln möglich.



In Abb 1. sehen Sie die ideale Sprungantwort eines PID-Reglers. Aus dem Bild wird die Bedeutung der einzelnen Parameter deutlich. Insbesondere die Bedeutung T_n als maßgeblicher Parameter für den I-Anteil ist gut zu erkennen. Hier wird auch deutlich, warum häufig T_n und nicht K_i als Parameter

genutzt wird. Den Parameter T_n als Zeit kann man sich im realen Prozess besser vorstellen.

In iCon-L werden beide Reglerstrukturen genutzt. Die Darstellung mit T_n und T_v wird häufig in Deutschland verwendet. Die Darstellung mit K_i und K_d ist im Angelsächsischen verbreitet.
 In iCon-L existieren 3 unterschiedliche PID-Reglerimplementierungen. Im Abschnitt [Sprungantwort der iCon-L PID-Regler](#) kann man das Verhalten der Regler gut vergleichen.

Was versteht man unter Nachstellzeit (T_n) ?

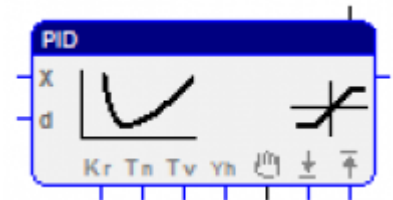
Der I-Anteil eines Reglerausgangssignales sorgt für eine ständige Veränderung des Reglerstellgrades, bis der Istwert den Sollwert erreicht hat. Solange eine Regelabweichung besteht, wird der Stellgrad auf- bzw. abintegriert. Je länger eine Regelabweichung an einem Regler ansteht, desto größer wird der integrale Einfluss auf den Stellgrad. Je größer die Regelabweichung und je kleiner die Nachstellzeit ist, desto stärker (schneller) ist die Wirkung des I-Anteils. Der I-Anteil sorgt dafür, dass ohne bleibende Regelabweichung ausgeregelt werden kann. Die Nachstellzeit ist ein Maß dafür, wie stark die zeitliche Dauer der Regelabweichung in die Regelung eingeht. Eine große Nachstellzeit bedeutet einen geringen Einfluss des I-Anteils und umgekehrt. In der angegebenen Zeit T_n (in sec.) wird die Stellgrößenänderung, die der P-Anteil (x_p oder p_b) bewirkt, noch einmal aufaddiert. Somit besteht ein festes Verhältnis zwischen dem P- und I-Anteil. Wird der P-Anteil (x_p) geändert, so bedeutet das auch ein geändertes Zeitverhalten, bei einem gleichbleibenden Wert von T_n .

Was versteht man unter Vorhaltezeit (T_v)

Über die Vorhaltezeit wird die Intensität des D-Anteils (Differentieller Anteil) eingestellt. Der D-Anteil eines Reglers, mit PID oder PD-Struktur, reagiert auf die Änderungsgeschwindigkeit des Istwertes. Beim Anfahren an den Sollwert „bremst“ der D-Anteil und kann damit ein Überschwingen der Regelgröße über den Sollwert vermeiden. Im Prinzip hat der D-Anteil folgende Auswirkungen: Sobald sich die Regelgröße verändert, wirkt der D-Anteil dieser Änderung entgegen. Für einen Regler mit einem inversen Wirksinn (also z. B. Heizen) würde dies beispielsweise bedeuten:

verkleinert sich die Regelgröße aufgrund einer Störung in der Regelstrecke, bildet der D-Anteil einen positiven Stellgrad, welche der Verkleinerung der Regelgröße entgegen wirkt. vergrößert sich die Regelgröße aufgrund einer Störung in der Regelstrecke, bildet der D-Anteil einen negativen Stellgrad, welche der Vergrößerung der Regelgröße entgegen wirkt. Je größer die Einstellung für die Vorhaltezeit (Wert in Sekunden) vorgenommen wird, umso größer ist das dämpfende Verhalten.

Kraftwerksregler (Implementierung 1)



Der Regler wird in diesem Dokument als Implementierung 1 (I1) geführt, da er historisch gesehen der erste in iCon-L verfügbare PID-Regler war.

- Bibliothek Controller (Float) (V1.1)
 - PID-Controller - (nutzt die Tn/Tv-Darstellung)

Dieser Regler wurde maßgeblich durch das Zentrum für Forschung und Technologie (ZFT) im VEB EAW Berlin entwickelt und als Kraftwerksregler genutzt.

Im Wesentlichen zeichnet sich der Regler durch die spezielle Implementierung des D-Anteils aus.

- Für die Aufschaltung des D-Anteils existiert ein extra Eingang (x). Im Normalfall wird der Eingang x auf das gleiche Signal wie der d-Eingang geschaltet. Experten können hier allerdings auch gezielt mit dem D-Anteil arbeiten.
- Der D-Anteil ist als „realer“ D-Anteil mit einer Verzögerung implementiert. siehe [Idealer und realer PID-Regler](#)

Weitere Besonderheiten des Kraftwerksreglers

- stoßfreie Hand/Automatikumschaltung
- stoßfreie Parameteränderung.

Dieser Regler arbeitet nach folgendem Algorithmus.

```

dYP=d - Wh
DYI=(d+Wh) *Ta / (2*Tn)
dYD=(X - Xh) * (10 * exp(Ta / (Tv/10))) + YD * (exp(Ta / (Tv/10)) - 1)
YD=YD+dYD
Xh=X
Wh=d
dY=(dYP+dYD+dYI) *Kr
Su=Su+dY;
Y=Su;

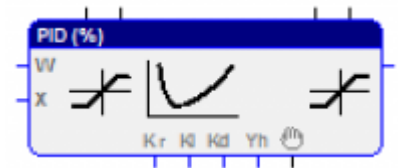
```

Hinweis : Der dargestellte Algorithmus entspricht nicht 1:1 dem C-Quellcode des Reglers, sondern ist eine vereinfachte Darstellung. Besonderheiten hinsichtlich Umschaltverhalten wind-up usw. sind nicht enthalten.

Wichtiger Hinweis zum PID-Demo-Projekt in früheren iCon-L Versionen

In den iCon-L PID-Demo Applikationen wurde der Eingang x an die Regelgröße angeschlossen. Das ist 'fałsch'. An den Eingang X muss in diesem Beispiel die Regeldifferenz (gleiches Signal wie an d) angeschlossen werden.

Universal- und Prozent-Regler (Implementierung 2 und 3)

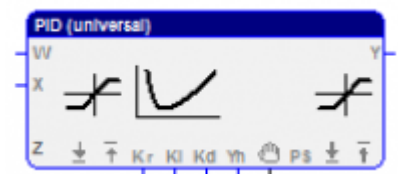


- Bibliothek PID-Controller (Float)(V3.0)
 - PID-Controller (Percent)
 - PID-Controller (Universal)

Die beiden Regler (Prozent und Universal) basieren auf dem gleichen mathematischen Basisalgorithmus

```
esum = esum + e
y = Kp * e + Ki * Ta * esum + Kd * (e - ealt)/Ta
ealt = e
```

Hinweis : Der dargestellte Algorithmus entspricht nicht 1:1 dem C-Quellcode des Reglers, sondern ist eine vereinfachte Darstellung. Besonderheiten hinsichtlich Umschaltverhalten wind-up usw. sind nicht enthalten.



Sprungantwort der iCon-L PID-Regler im Vergleich

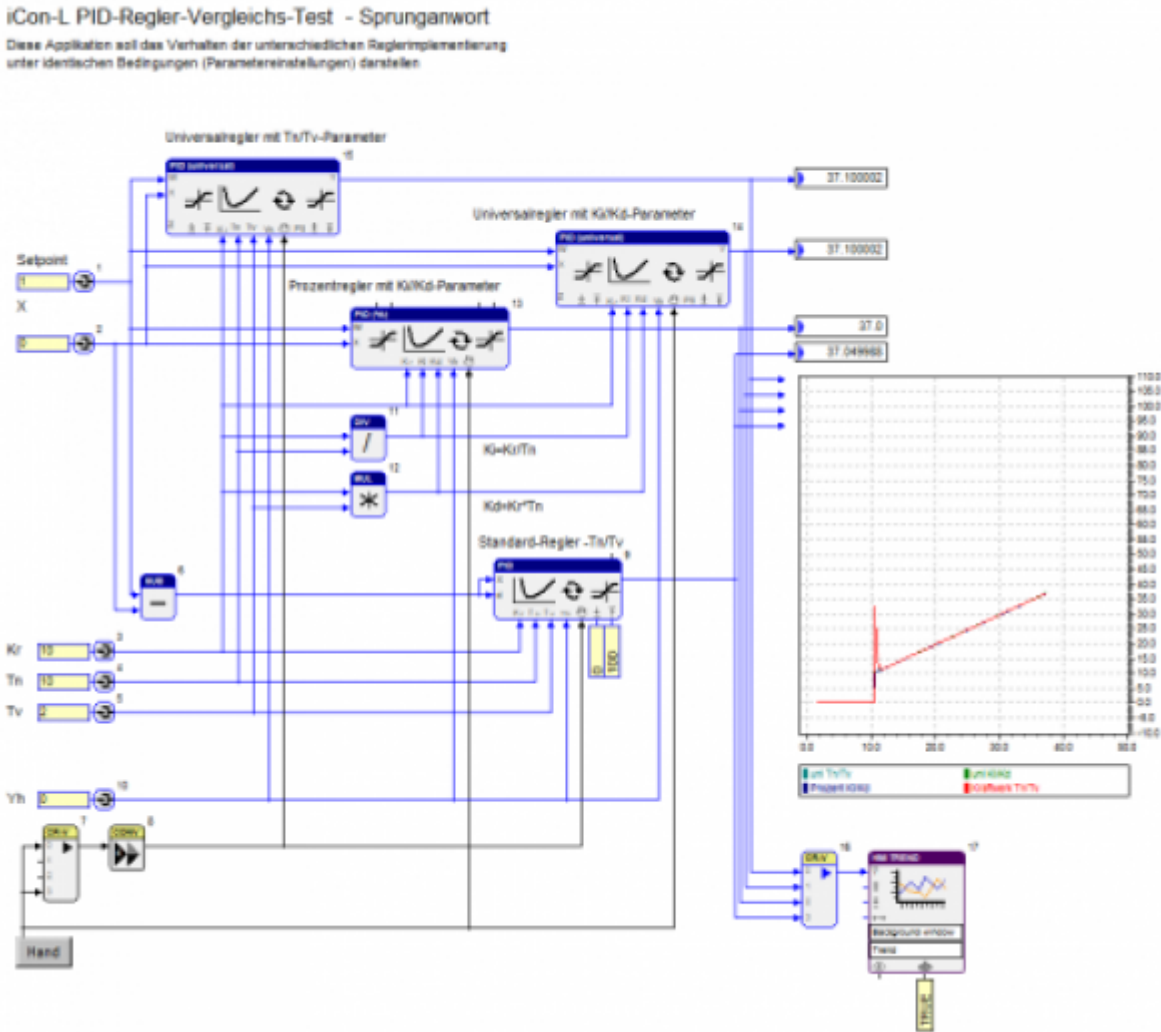


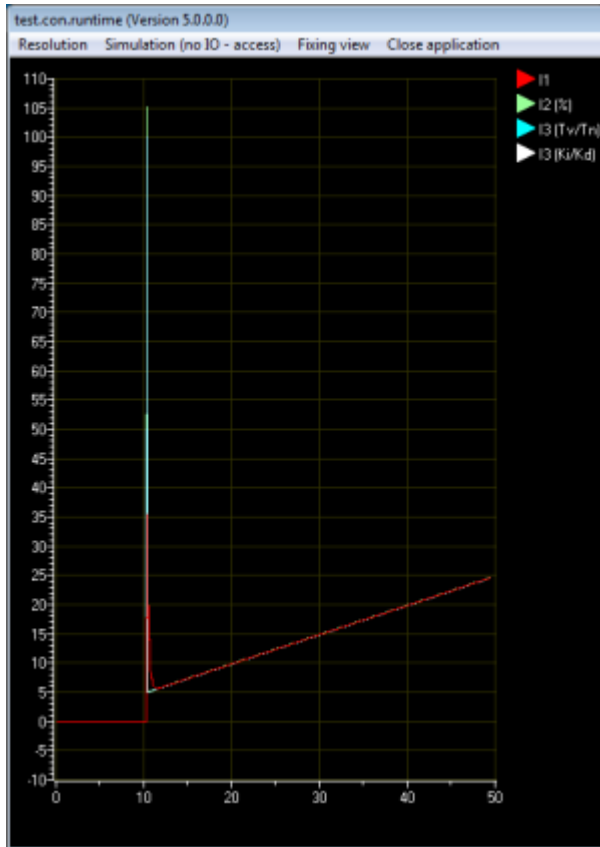
Abb.5 zeigt ein iCon-L Anwenderprogramm, welches die Sprungantwort der unterschiedlichen iCon-L Reglerimplementierungen im Vergleich darstellt. Für alle Regler werden identischen Parameter genutzt. (Umrechnung der Parameter in die unterschiedlichen Darstellungsformen $T_n \rightarrow K_i$; $T_v \rightarrow K_d$)

Kr = 5
 Tn = 10 (Sekunden)
 Tv = 2

Testverlauf

- Das Programm wird mit den konstanten Werten \rightarrow Sollwert $W=0$ und Instwert $X=0$ in den Controller geladen.
- Nach 10 Sekunden wird der Sollwert W auf 1 gesetzt (positives Sprungsignal).

Ergebniss



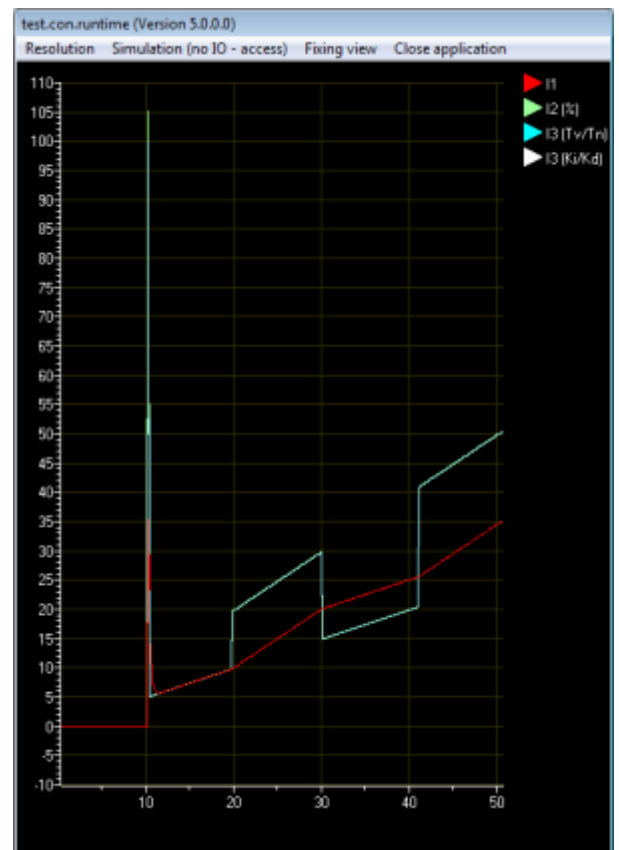
Wie zu erwarten war, zeigen die Diagramme (Abb.5 und Abb.6) ähnliche Kurven für alle Regler. Der Kraftwerksregler (I1 -rote Linie) verhält sich im D-Anteil etwas anders. Der Regler zeigt keinen idealen Sprung, sondern fällt in einer e-Funktion ab. Der Regler arbeitet also wie ein [realer PID-Regler](#).

Verleich mit Abb.1 - ideale Antwort

Die Regler zeigt den erwarteten Signalverlauf.

1. Nach dem Eingangssignalsprung reagiert der Regler mit einem sehr starken Impuls.
2. Dann fällt der Impult bis auf den Kr-Wert
3. Der Wert steigt dann alle 10 Sekunden (T_n) um den Kr-Wert

Parameteränderung während der Laufzeit



Während sich die 3 Implementierungen beim Sprung fast identisch verhalten, ist dies bei Parameteränderungen während der Laufzeit nicht mehr so. Sehr deutlich wird dies, wenn man die sich die Sprungantwort mit einer Parmeteränderung (K_r) ansieht.

1. K_r wird nach 20 Sekunden von 5 auf 10 geändert
2. Nach 30 Sekunden wird K_r wieder auf 5 gesetzt
3. Nach ca 40 Sekunden wird K_r wieder auf 10 gesetzt

In einem zweiten Versuch wurde der Parameter T_n während der Laufzeit geändert. Das Bild ähnelt sehr stark dem Signalverlauf bei der Änderung von K_r .

1. T_n wird nach 20 Sekunden von 10 auf 5 geändert
2. Nach 30 Sekunden wird T_n wieder auf 10 gesetzt
3. Nach 40 Sekunden wird T_n wieder auf 5 gesetzt

Ein Änderung von T_v hat keine Auswirkungen, da sich ja auch die Diffrenz zwischen W und X sich nicht ändert.

Ergebnis

Kraftwerksregler (Implementierung 1) mit stoßfreier Parameterumschaltung

In Abb. 7 sehen wir den Signalverlauf der Sprungantwort. Die rote Linie zeigt den Signalverlauf des Kraftwerkreglers (I_1). Es ist zu erkennen, dass die Parameteränderung (K_r) am Ausgang zu keinem

Signalstoß führt. Der Regler arbeitet im Geschwindigkeitsmodus, sodass sich der neue Parameter nur auf die Änderung der Differenz zwischen Soll und Istwert auswirkt.

Regler (I2 und I3) mit stoßbehaftete Parameterumschaltung

Anders verhält es sich bei den Reglern (I2 und I3). Beide Regler verhalten sich identisch, daher auch nur eine Kurve für beide Regler. Die Parameteränderung hat eine unmittelbare Auswirkung auf das Ausgangssignal und führt zu einem Stoß im System. Dabei ist es egal, ob K_r oder T_n geändert wird. Entsprechend des Algorithmus haben beide Parameteränderungen ähnliche Auswirkungen. Y ist unmittelbar von K_i und dem Integral der Differenz abhängig. Anders als in der Implementierung des Kraftwerksreglers wirkt der Parameter K_i auf das gesamte Integral des I-Gliedes. So ist auch die stärkere Wirkung der Parameteränderung nach längerer Zeit zu erklären. Da in K_i unmittelbar auch K_r steckt, wirkt sich die Änderung von K_r auch unmittelbar auf das I-Verhalten aus.

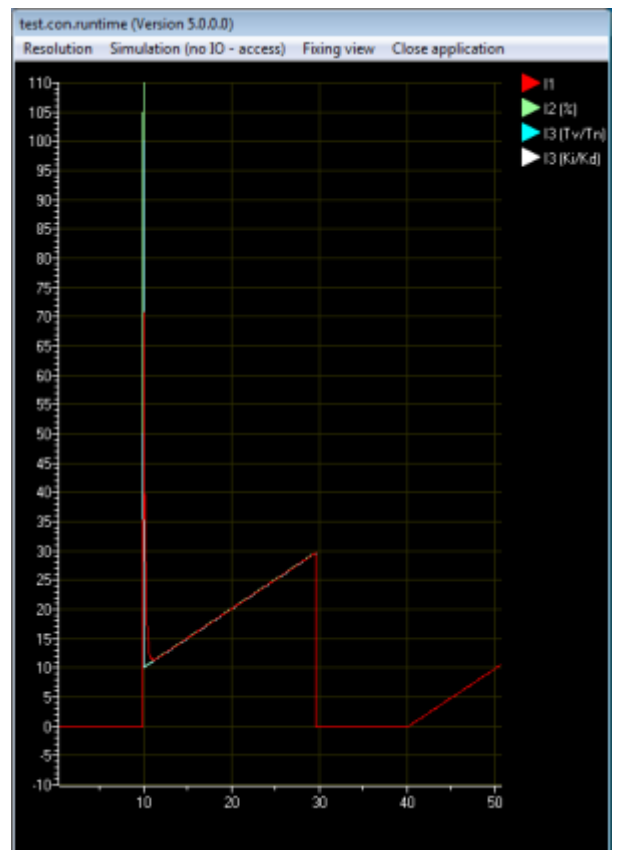
$$K_i = K_r / T_n$$
$$y = K_r * e + K_i * T_a * e_{sum} + K_d * (e - e_{alt}) / T_a$$

Eine Änderung von T_v hat keine Auswirkungen, da sich Differenz bei einem Sprungsignal nicht ändert.

Warum diese unterschiedlichen Implementierungen?

Obwohl die stoßbehaftete Parameteränderung in realen Systemen zu Problemen führen kann, entspricht diese Implementierung der Erwartung des Anwenders. Wenn sich ein Parameter verdoppelt, erwartet der Anwender eine unmittelbare und heftige Reaktion des Reglers, da er die vereinfachte mathematische Formel im Kopf hat.

Hand/Automatikumschaltung



Testverlauf

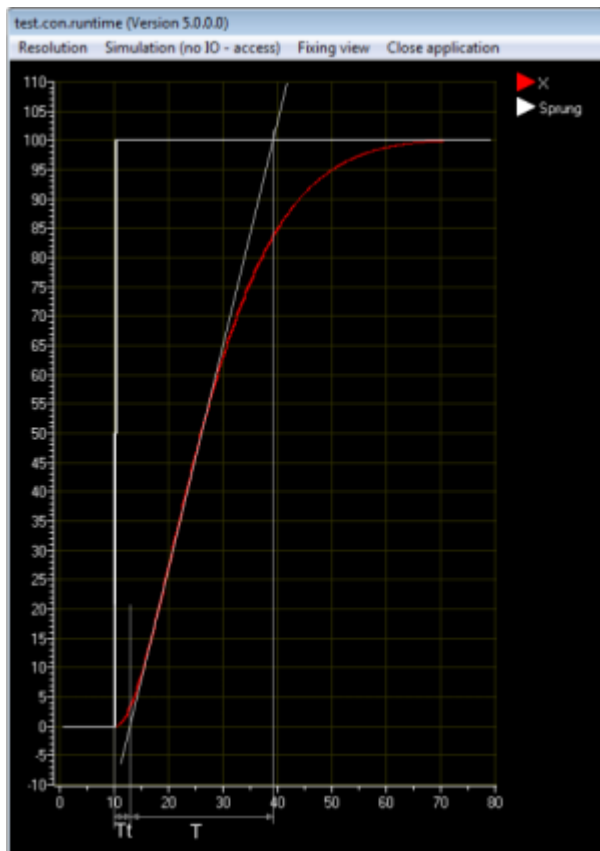
- Nach 10 Sekunden wird ein Sprungsignal (1) auf den offenen PID-Regler gegeben.
- Nach 30 Sekunden werden die Regler auf Hand geschaltet ($Y_h = 0$)
- Nach 40 Sekunden wird der Regler wieder auf Automatik geschaltet.

Ergebnis - Stoßfreie Hand/Automatikumschaltung

Alle drei Implementierungen verhalten sich bei der einfachen Hand/Automatikumschaltung identisch.(siehe Abb. 8)

- Bei der Umschaltung auf Hand wird das Signal $Y_h = 0$ auf den Ausgang Y gegeben.
- Zugleich wird die interne Summe der Regler auf den Y_h -Wert abzüglich des Wertes aus dem P-Anteil gesetzt.
- Bei der Umschaltung auf den Automatikbetrieb startet der Regler auf dem voreingestellten Y_h -Wert am Ausgang
- Im Automatikbetrieb wirkt zunächst nur der I-Anteil, da der P-Anteil aus der internen Summe herausgerechnet wurde.

Der geschlossene Regelkreis



Nun soll das Verhalten der einzelnen Reglerimplementierungen im geschlossenen Regelkreis betrachtet werden. Schauen Sie sich hierzu auch die Wikipediaseite mit den Faustregeln zum Einstellen an. http://de.wikipedia.org/wiki/Faustformelverfahren_%28Automatisierungstechnik%29 Zur einfachen Parameterbestimmung soll das Ziegler und Nichols -Verfahren genutzt werden.

Die Anwendung des Ziegler und Nichols - Verfahren soll hier nicht als Empfehlung wahrgenommen werden. Es ist nicht unbedingt das optimale Verfahren (siehe auch Faustformelverfahren (wikipedia)). Das Verfahren dient nur zur Darstellung des Verhaltens der Regler.

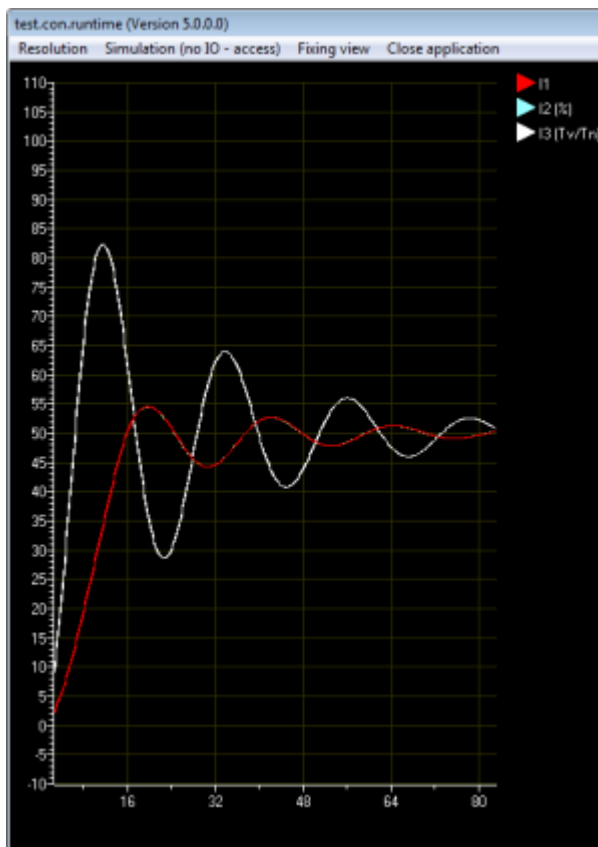
Sprungantwort der PT2-Strecke

In Abb. 9 sehen Sie die Sprungantwort der PT2-Strecke (PT2-Funktionsbaustein - TC=10; A=0.9). Der Kurvenverlauf wird entsprechend des gewählten Verfahrens ausgewertet.

Ergebnisse

$$T_t = 3 \text{ Sec.}$$
$$T = 26 \text{ Sec.}$$

Führungsverhalten als PI-Regler



Entsprechend der Faustformel wurden die Parameter für PI-Regler ermittelt und eingetragen.

$$K_r = 0.9/K * T/T_t = 7.8$$

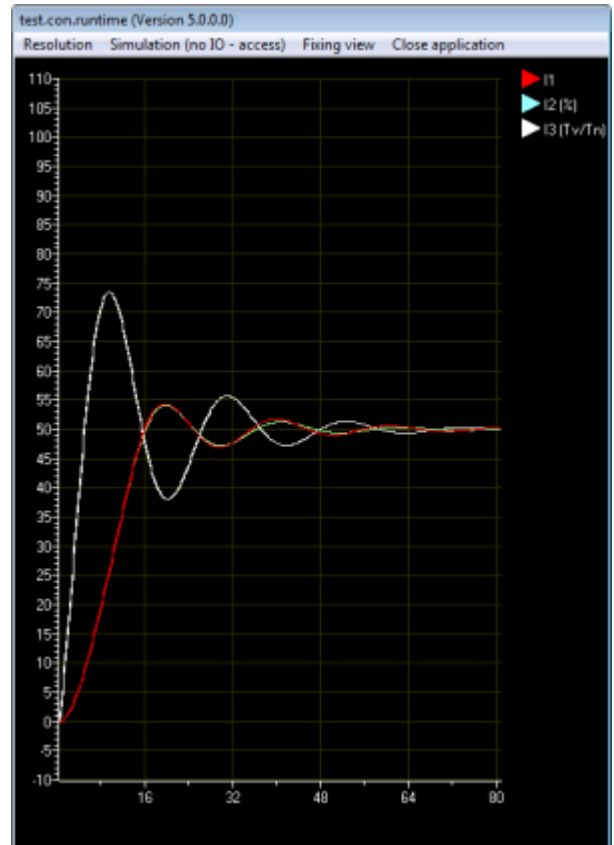
$$T_n = 3.33 * T_t = 9.99$$

- Abb. 10 zeigt das Verhalten der Regler
 - Der Universalregler (I3) (Abb. 10 - weiße Linie) arbeiten ohne Bgrenzung des Ausgangs - normalerweise wird der Ausgang entsprechend der realen Möglichkeiten des Stellgliedes begrenzt.
 - Der Kraftwerkregler (I1) und der Prozentregler (Abb. 10 I1= rote Linie und I2 = grüne Linie) ist am Ausgang auf 0-100% begrenzt und kann daher nicht das volle Potenzial ausspielen.

Alle 3 Implementierungen (I1 bis I3) würden sich grundsätzlich identisch verhalten, wenn I3 ebenfalls auf 0-100 begrenzt werden würde. Die Regler schwingen weit über, da das Einstell-Verfahren nach ZN für ein Störverhalten und nicht für das Führungsverhalten genutzt wird.

siehe [http://de.wikipedia.org/wiki/Regelkreis - Führungsverhalten und Störverhalten](http://de.wikipedia.org/wiki/Regelkreis_-_Führungsverhalten_und_Störverhalten)

Führungsverhalten als PID-Regler



Entsprechend der Faustformel nach ZN wurden die Parameter für die PID-Regler ermittelt und eingetragen.

$$K_r = 1.2/K * T/T_t = 10.3$$

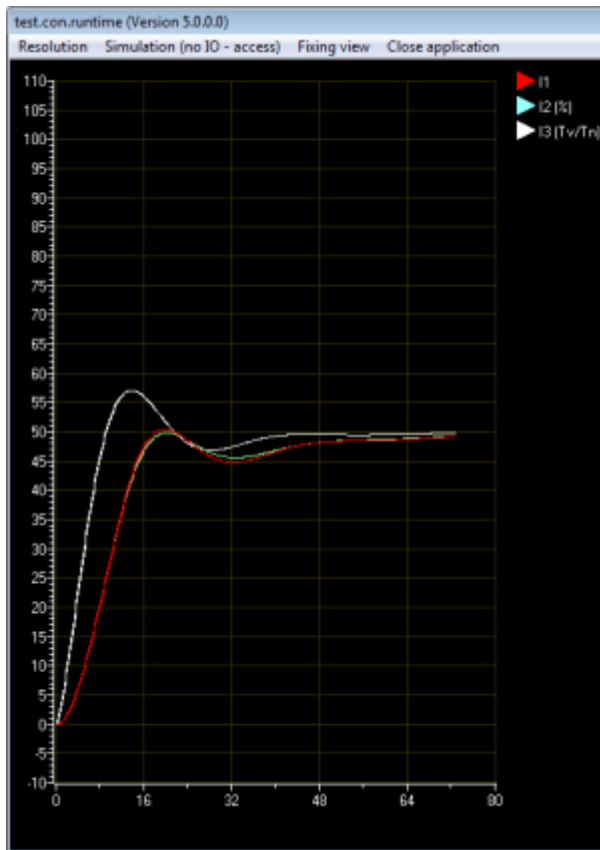
$$T_n = 2 * T_t = 6$$

$$T_v = 0.5 * T_t = 1.5$$

- Abb. 12 zeigt das Verhalten der Regler
 - Der Universalregler (I3) (Abb. 12 - weiße Linie) arbeiten ohne Bgrenzung des Ausgangs - normalerweise wird der Ausgang entsprechend der realen Möglichkeiten des Stellgliedes begrenzt.
 - Der Kraftwerksregler (I1) und der Prozentregler (I2) (Abb.12 - I1 = rote Linie, I2 = grüne Linie) ist am Ausgang auf 0-100% gegrenzt und kann daher nicht das volle Potenzial ausspielen.

Der PID-Regler schwingen schon nicht mehr so weit über wie der PI-Regler, da der D-Anteil dämpfend wirkt, obwohl K_r erhöht werden konnte und der Regler insgesamt schneller arbeitet.

Einstellregeln nach Chien, Hrones und Reswick



Ein besseres Führungsverhalten erreicht man, wenn man die Einstellregeln nach Chien, Hrones und Reswick nutzt.

siehe Abb.13.

| Regler | Aperiodischer Regelverlauf | | Regelverlauf mit 20 % Überschwängen | |
|--------|----------------------------------|------------------------------|-------------------------------------|-------------------------------|
| | Störung | Führung | Störung | Führung |
| P | Kr $0.3 \cdot (T/Tt \cdot K)$ | $0.3 \cdot (T/Tt \cdot K)$ | $0.7 \cdot (T/Tt \cdot K)$ | $0.7 \cdot (T/Tt \cdot K)$ |
| PI | Kr $0.6 \cdot (T/Tt \cdot K)$ | $0.35 \cdot (T/Tt \cdot K)$ | $0.7 \cdot (T/Tt \cdot K)$ | $0.6 \cdot (T/Tt \cdot K)$ |
| | Tn $4 \cdot Tt$ | $1.2 \cdot T$ | $2.3 \cdot Tt$ | $1 \cdot T$ |
| PID | Kr $0.95 \cdot (T/(Tt \cdot K))$ | $0.6 \cdot (T/(Tt \cdot K))$ | $1.2 \cdot (T/(Tt \cdot K))$ | $0.95 \cdot (T/(Tt \cdot K))$ |
| | Tn $2.4 \cdot Tt$ | $1 \cdot T$ | $2 \cdot Tt$ | $1.35 \cdot T$ |
| | Tv $0.42 \cdot Tt$ | $0.5 \cdot Tt$ | $0.42 \cdot Tt$ | $0.47 \cdot Tt$ |

Kr = 5.2
 Tn = 26
 Tv = 1.5

Störverhalten der Regler im Vergleich

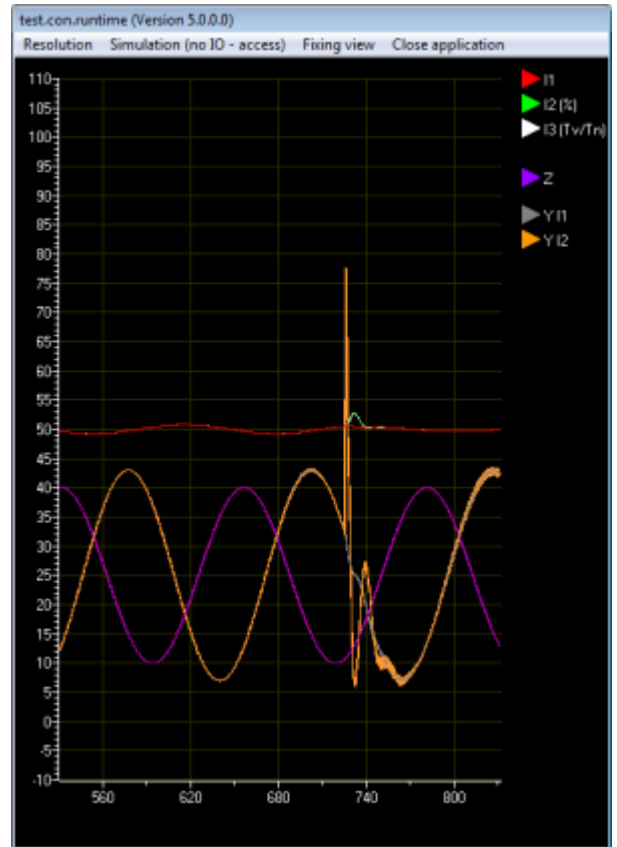
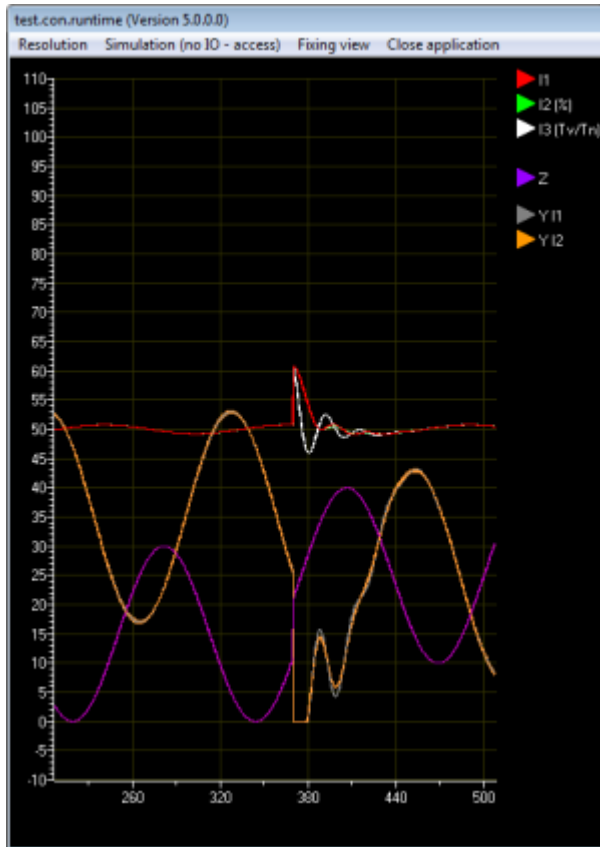
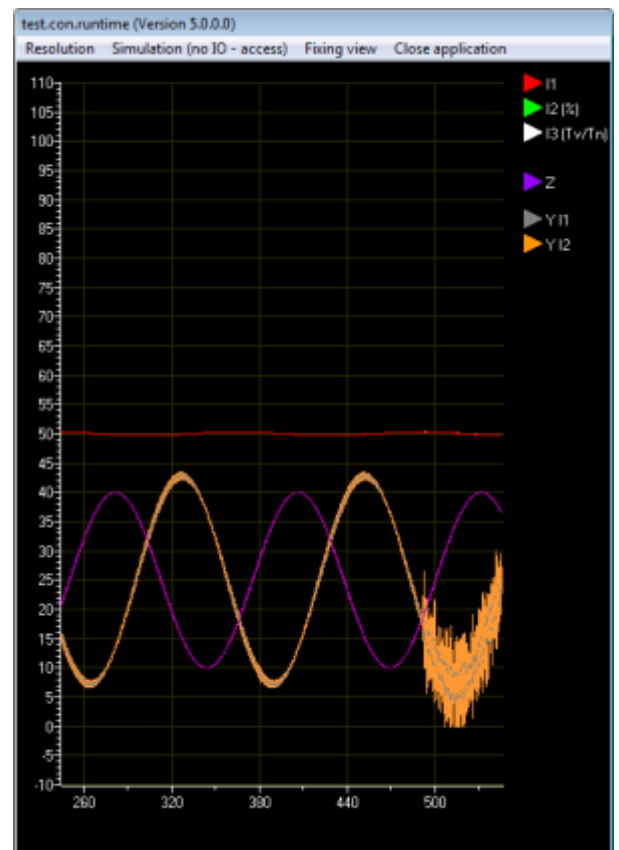
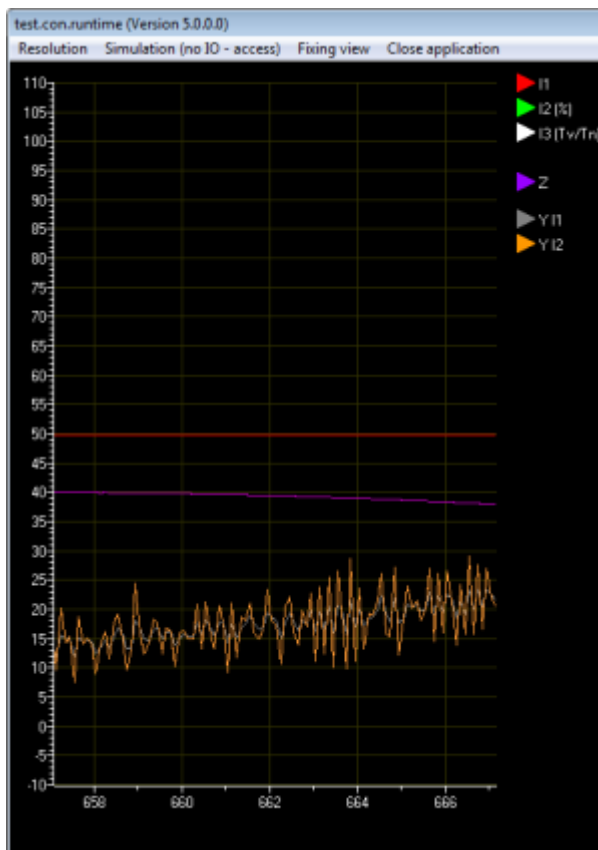


Abb. 13 zeigt das Störverhalten der Regler bei einer niederfrequenten sinusförmige Störung. Die lila-Linie zeigt das Störsignal, welches unmittelbar auf die Regelgröße aufgeprägt wird. Die überdeckende rote, weiße und grüne Linie zeigt die ausgeregelte Regelgröße. Die orange und überdeckte graue Linie zeigt das Stellsignal der Regler I2 und I1. Der heftige Ausschlag der Regelgröße rührt von Sprung des Störsignal um 25%. Alle drei Regler regeln den Störungssprung ähnlich gut aus.

Abb. 14 zeigt das Verhalten der Regler bei einer Parameteränderung. Während der Universalregler und der Prozentregler auf Parameteränderungen mit einem heftigen Ausschlag reagieren, macht sich eine Parameteränderung am Kraftwerksregler nicht bemerkbar.

Idealer und realer PID-Regler



Bei einem idealen Regler hat der D-Anteil keine Verzögerung. Ein solches Verhalten ist in der Praxis aber nicht realisierbar und auch nicht erstrebenswert, da die Regeleinrichtung dann schon bei höherfrequenten Störungen kleiner Amplitude, die der Regelgröße überlagert sein können, zur Übersteuerung neigen würde. Die Implementierungen des Universalreglers (I3) und des Prozentregler (I2) kommen dem Verhalten eines idealen Reglers allerdings weit näher als der Kraftwerksregler, da diese Implementierungen im Algorithmus selbst keinen Verzögerung des D-Anteils besitzen. Selbstverständlich reagieren diese Regler in der Kombination mit dem Stellglied ebenfalls verzögert. Die Abbildungen. 15 und 16 zeigen, dass der Universalregler (I3) und der Prozentregler (I2) auf höherfrequente Störungen relativ „nervös“ reagieren, während der Kraftwerksregler aufgrund seiner eigenen Verzögerung des D-Anteils wesentlich sanfter auf höherfrequente Störungen reagiert.

From:
<https://wiki.pro-sign.de/psdokuwiki/> - ProSign DokuWiki

Permanent link:
https://wiki.pro-sign.de/psdokuwiki/doku.php/de:public:user_s_documents:icon-l:applications:pid

Last update: 2014/01/28 07:45

