

Vorlesung Automatisierungstechnik

für Prozess- und Biotechnik

HS Furtwangen University, Fachbereich MLS, Prof. Dr. Stefan von Weber

Version März 2015 (Skript 1333)

Inhaltsverzeichnis	Content
1. Einleitung	Introduction
2. Automatisierungskonzepte	Concepts of automation
2.1 Gerätesteuerung mit μ -Controller	Device control with a μ -controller
2.2 Anlagenautomatisierung mit SPS	Facility automation with a SPC
2.3 Automatisierung mit PC	Automation with a PC
3. Kommunikation	Communication
3.1 Signaltransport und Verarbeitung	Signals - transport and processing
3.2 Netze	Networks
4. Pneumatik	Pneumatics
4.1 Logik- und Schaltzeichen	Logic and switching symbols
4.2 Elektropneumatische Ventile	Electro pneumatic control valves
4.3 Pneumatikplan	Pneumatic circuit diagram
4.4 Sensoren und Aktoren	Sensors and actuators
5. Regeln	Feedback control
5.1 Einführung	Introduction
5.2 Regelstrecken	Controlled members
5.3 Kontinuierliche Regler	Continuous controllers
5.4 Regeln mit der SPS	Control with SPC
5.5 Fuzzy Control	Fuzzy control
5.6 Neuronale Netze	Neuronal networks
5.7 Mehrgrößenregelung	Multivariable control loops
5.8 Stabilität von Regelkreisen, Gütemaße, Optimierung	Stability of control loops Measures of quality, optimization
6. Übungsklausuren	Written exams
7. Übungsliste	list of exercises
8. Dictionary	dictionary

Literatur

References

- M. Horn / N. Dourdoumas (2004) : Regelungstechnik, Pearson Studium, München**
 Töpfer, Besch (1990): Grundlagen der Automatisierungstechnik
Wellenreuther/Zastrow (2001): Automatisieren mit SPS, Vieweg
 Haug (1991): Pneumatische Steuerungstechnik
 Dorf/Bishop (2005): Modern Control Systems, Pearson Prentice Hall
G.Stormann (1990): Messtechnik im Chemiebetrieb, Oldenbourg
 Zakharian u. a. (1998): Neuronale Netze für Ingenieure, Vieweg

1. Einleitung

Vergleicht man ein Automatisierungssystem mit dem menschlichen Körper, dann sind die Rechner (PC, SPS, μ -Controller) das Gehirn, die Signalleitungen die Nervenbahnen, die Sensoren die Augen und Ohren, die Aktoren die Muskeln, die Starkstromleitungen die Adern. Alle Teile sind wichtig und müssen zueinander passen.

2. Automatisierungskonzepte

Das **Konzept** hängt von vielen Gesichtspunkten ab, z.B.

- den geforderten Funktionen (Steuern, Regeln, Daten sammeln, ...)
- der Größe der Anlage (Gerät, Anlage, Fabrik, Verbund, ...)
- der Fahrweise (Batch, kontinuierlicher Prozess)
- der geforderten Sicherheit (normal, explosionsgeschützt)

2.1 Gerätesteuerung mit μ -Controller

Ein μ -Controller ist ein 1-Platinenrechner von der Größe eines Handys oder kleiner. Ein μ -Controller kann rechnen, Daten speichern, Analogwerte in Zahlen wandeln und zurückwandeln. Man wendet ihn in vielen Geräten an, z.B. in Messgeräten, Analysegeräten, Beschriftungssysteme, Erkennungssysteme. Die μ -Controller werden sehr oft in der Programmiersprache C programmiert.

2.1.1 Kurze Einführung in die wichtigsten Instruktionen von C

Ein C-Programm besteht aus einem Hauptprogramm (main) und beliebig vielen Subroutinen, z.B. mathematische Funktionen oder Unterprogrammen, die Steuerfunktionen haben oder der Datenkommunikation dienen usw.

int main(...)	Ein Main-Programm beginnt mit dieser Anweisung. Das „int“ sagt, dass es bei Beendigung einen ganzzahligen Rückkehrcode abliefert, der Auskunft über den Erfolg oder Misserfolg der Programmmission gibt.
------------------------	--

double x, y, A[100], M[30][20];	Reserviert Speicherplatz für die Variablen x und y, für den Vektor A mit den Elementen A_0, \dots, A_{99} und für die Matrix $M_{0,0}, M_{0,1}, \dots, M_{0,19}, \dots, M_{29,19}$. Jeder Wert dieser Größen wird als 8-Byte Gleitkommazahl (ca. 15 Stellen Genauigkeit) abgespeichert. Mit double Zahlen rechnet man.
int i, j, N, kvec[10000];	Reserviert Speicherplatz für die Variablen i, j, und N und für den Vektor kvec mit den Elementen $kvec_0, \dots, kvec_{9999}$. Jeder Wert dieser Größen wird als 4-Byte integer-Zahl (ganzzahlige Zahl) abgespeichert. Mit int-Zahlen zählt man.

Eine **Programmverzweigung** erlaubt unterschiedliche Programmstücke zu durchlaufen.

if (x>0) y = 3.14;	Wenn der augenblickliche Wert der Variablen x größer null ist, dann und nur dann setze die Variable y auf den Wert 3.14.
if(x<=5) { y = x*x; z = sin(x); }	Bei mehreren Anweisungen in der Verzweigung sind diese mit geschweiften Klammern zu verbinden.
if(x>0) { y=x/3.14; z=log(x); } else { y=0; z=3.14; }	Im Ja-Fall (folgt immer unmittelbar dem if) wird $y = x/3.14$ und $z = \ln(x)$ ausgeführt, im Nein-Fall $y=0$ und $z=3.14$.

Schleifen gestatten es, Programmstücke beliebig oft zu durchlaufen. Bei der **for-Schleife** ist die Zahl der Durchläufe von vornherein bekannt

N=5;	Die ganzzahlige Zählgröße N wird 5 gesetzt.
for(i=0; i<N; i++) { A[i]= i*x; }	Der Index i durchläuft die Zahlen 0, 1, 2, 3, 4 und es werden im Schleifenkörper die Aktionen $A_0=0*x, A_1=1*x, A_2=2*x, \dots, A_4=4*x$ ausgeführt, d.h. 5 Vektorelemente von A werden belegt. Das Kürzel i++ sagt, dass jeweils nach einem Durchlauf der Laufindex i um den Wert eins zu erhöhen ist.

Bei der **do-while-Schleife** wird der Schleifenkörper mindestens einmal durchlaufen und dann erst auf eine Wiederholung der Schleife getestet. Die Zahl der Durchläufe muss vorher nicht bekannt sein.

i=0; do{ A[i]=i*x; i=i+1; } while ((A[i]<800) && (i<99)); N=i;	Setze Laufindex i auf Startwert 0. Start der do-while-Schleife Setze Vektorelement A_i gleich i mal x.. Erhöhe Index i um eins (geht auch mit i++). Wenn der Wert von Vektorelement A_i kleiner als 800 ist und der Wert von i kleiner als 99 ist, dann wird die Schleife wiederholt. Andernfalls wird sie verlassen. Merke auf Zählgröße N die Zahl der belegten Elemente von Vektor A (Zahl der ausgeführten Durchläufe).
--	---

Bei der **while-Schleife** wird vor Eintritt in den Schleifenkörper getestet, ob er durchlaufen werden soll. Falls nicht, dann wirkt die Schleife wie eine Leeranweisung.

double PI = 3.14159265;	Setze Größe PI gleich π
while(x > 2*PI) { x = x - 2*PI; }	Wenn x größer 2π ist, dann erniedrige x um 2π bis der Wert von x kleiner oder gleich 2π ist. Dann verlasse die Schleife.

2.1.2 Verfahrenstechnischer Beispielprozess - Herstellung einer Lotion im Rührkessel

Die Abkürzungen bedeuten: Mot= Motor, Kl=Klappe, ES=Endschalter „Klappe zu“,

P0, P1, P2 = Pegelmelder, V1, V2,...=Ventile, Pu1, Pu2 = Pumpen.

Motor und Pumpen laufen bei Signal 1, stehen bei 0.

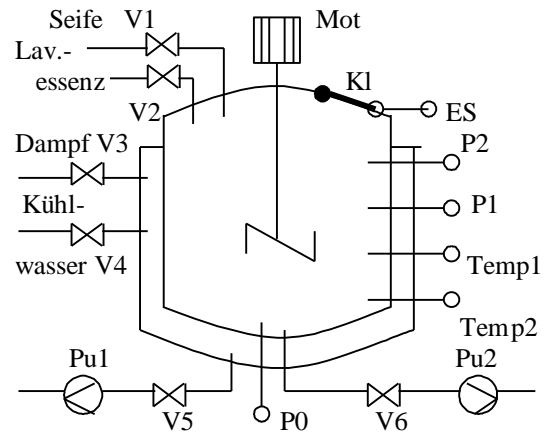
Ventile öffnen bei Signal 1, schließen bei 0.

Pegelmelder und Endschalter melden Signal 1, wenn berührt, sonst 0.

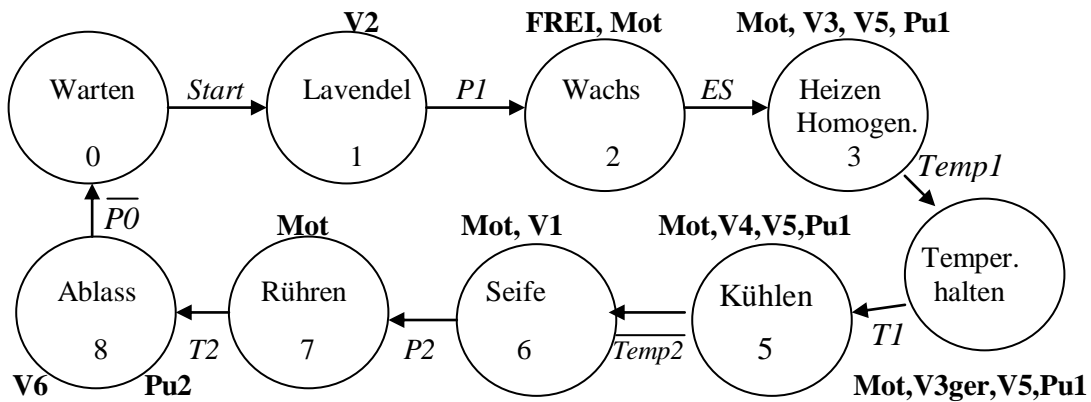
Temperaturmelder melden Signal 1, wenn die eingestellte Temperatur erreicht oder überschritten ist, sonst 0.

Die 9 Prozessschritte sind:

0. Warten auf Start,
1. Befüllen Lavendelessenz bis Pegelmelder P1,
2. Klappenfreigabe mit Signal *FREI* (Klappe öffnet sich, Arbeiter befüllt mit Wachspenlen, drückt Klappe zu bis sich Endschalter ES meldet),
3. Homogenisieren und Heizen bis Temperaturmelder Temp1 anspricht,
4. 15 Min Temperatur halten und rühren,
5. Wasserkühlung bis Temperaturmelder Temp2 nicht mehr anspricht,
6. Flüssigseife bis Pegel P2 befüllen,
7. 10 min mischen,
8. Abpumpen und zurück zum Warten.



Ein **Mealy-Graphen** stellt einen Prozess als Schrittfolge graphisch dar. Die Knoten werden als nummerierte und beschriftete Kreise dargestellt. Jeder Knoten ist ein Prozessschritt. Außen am Kreis vermerkt man die Aktionen dieses Prozessschritts (z.B. Ventil öffnen, Rühren, Heizen, Pumpen, Leuchtsignal ausgeben, ...). An die Verbindungspfeile (Kanten) des Graphen schreibt man die Signale oder Signalkombinationen, die den Übergang von einem Prozessschritt zu seinem Nachfolger auslösen (z.B. Startsignal, Pegelmelder, Temperaturmelder, Timersignal, ...). Falls das Ausbleiben eines Signals den Übergang auslösen soll, z.B. beim Abpumpen das Ausbleiben des Pegelmeldersignals P0, dann schreibt man $\overline{P0}$, d.h. man setzt einen Negationsstrich über das Signal. Bei geregelten Aktoren setzt man ein „geregelt“ oder kurz „ger“ hinzu, z.B. beim Halten der Temperatur wird mit dem Dampfventil V3 geregelt (V3ger).



Die Schalttafel enthält hier:

Den Starttaster,
den Betriebsschalter mit LED,
den Netzschalter,

und 10 LEDs, die wir im Programm
mit BetriebsLED, LED0 für Warten,
LED1 für Lavendel usw. bis LED8 für
Ablassen bezeichnen.

Start Warten



Betrieb



Netz

Lavendel

Wachs

Heizen

Halten

Kühlen

Seife

Rühren

Ablassen

2.1.3 Entwurf eines Steuerprogramms in C für einen μ -Controller

Kommentare im Programm werden durch zwei Schrägstriche eingeleitet. Kommentare sind nur informativ für den Leser, nicht programmrelevant.

```
int main(...) // Der Inhalt der Klammer hängt vom Betriebssystem ab
{ double T1, T2; // Timer arbeiten mit gebrochenen Zahlen, also double-Zahlen verwenden
  int Start, Betrieb, P0, P1, P2, Mot, FREI, V1, V2, V3, V4, V5, V6, Pu1, Pu2;
  int BetriebsLED, LED0, LED1, LED2, LED3, LED4, LED5, LED6, LED7, LED8;

  // Setze alle Variablen (außer Variable Betrieb) auf Startwert 0
  T1= T2= Start= P0= P1= P2= Mot= FREI= V1= V2= V3= V4= V5= V6, Pu1= Pu2= 0;
  BetriebsLED= LED0= LED1= LED2= LED3= LED4= LED5= LED6= LED7= LED8= 0;

  // Warten darauf, dass der Betriebsschalter in die Stellung „Ein“ geschoben wird (Signal 1)
  do{ }while( !Betrieb); // Warten auf das Signal 1, d.h. rotiere solange Signal Betrieb = 0 ist
  BetriebsLED=1; // Schalte die BetriebsLED ein.
```

```

// do-while-Schleife über alle 9 Prozesszustände von Warten bis Ablassen
do{
    LED0=1;           // Schalte LED „Warten“ ein.
    do{ }while( !Start); // Warten auf das Startsignal, d.h. rotiere solange Start= 0 ist
    LED0=0;           // Schalte LED aus

    // Befüllen mit Lavendelessenz über Ventil V2
    do{ V2=1; LED1=1;} while ( !P1); // Öffne Ventil V2 solange P1=0 ist
    V2=0; LED1=0;           // Schließe das Ventil V2, Schalte LED aus

    Mot=1;               // Schalte den Rührer einmalig ein für Prozessschritte 2 bis 7

    // Klappe freigeben über Leuchtsignal „FREI“ bis der Arbeiter die Klappe
    // schließt und sich der Klappenendschalter ES mit Signal 1 meldet
    do{ FREI=1; LED2=1; } while( !ES );
    FREI=0; LED2=0;

    // Heizen mit Dampf sowie Abdampf und Kondenswasser abpumpen
    do{ V3=1; V5=1; Pu1=1; LED3=1;} while (!Temp1);
    V3=0; V5=0; Pu1=0; LED3=0;

    // Temperatur mit einem Zweipunktregler 15 min = 900 sec halten. Dazu wird
    // über die Funktion time() die augenblickliche Zeit geholt und Timer T1 gestellt.
    // Unsere Funktion time() liefere die abgelaufenen Sekunden seit Start des Controllers.
    // Der Zweipunktregler if ( Temp1) ... schließt das Dampfventil V3, wenn der
    // Temperaturfühler Temp1=1 meldet (Solltemperatur ist erreicht bzw. überschritten)
    //
    T1 = time() + 900; LED4=1; // Bis zur Zeit T1 ist geregelt zu heizen
    do{ if (Temp1) { V3=0; V5=0; Pu1=0; } else { V3=1; V5=1; Pu1=1; }
        } while ( time() < T1 );
    V3=0; V5=0; Pu1=0; LED4=0;

    // Kühlen bis der Temperaturfühler Temp2 Signal 0 meldet (Temperatur unterschritten)
    do{ V4=1; V5=1; Pu1=1; LED5=1;} while ( Temp2 );
    V4=0; V5=0; Pu1=0; LED5=0;

    // Mit Flüssigseife über Ventil V1 bis Pegel P2 befüllen
    do{ V1=1; LED6=1;} while ( !P2 );
    V1=0; LED6=0;

    // 10 min = 600 sec Rühren. Der Rührer läuft noch immer seit Prozessschritt 2.
    T2 = time() + 600; LED7=1; // Bis zur Zeit T2 ist zu rühren
    do{ } while ( time() < T2 );
    LED7=0;

    Mot=0; // Schalte den Rührer ab

```

```

// Ablassen der Lotion. Sie wird zur Abfüllanlage gepumpt. Wenn der Pegelmelder
// P0 das Signal 0 meldet, ist der Rührkessel leer und das Abpumpen beendet.
do{ V6=1; Pu2=1; LED8=1;} while ( P0 );    // Rotiere solange P0 eine 1 meldet
V6=0; Pu2=0; LED8=0;

} while (Betrieb);    // Verlassen der Schleife bei Betrieb = 0, ansonsten zurück zum Warten

BetriebsLED=0;    // Schalte die BetriebsLED wieder aus.
return (0);    // Gib Rückkehrcode 0 an das Betriebssystem zurück (alles okay)
}

```

2.1.4 Allgemeine Hinweise zu Steuerprogrammen

Das wichtigste Programmierelement der Steuerung und Regelung ist die zentrale Steuerschleife. Meist nimmt man dafür eine do-while-Schleife. Ein typisches Steuerprogramm besteht aus den folgenden 3 Abschnitten:

Setze Startwerte , wie z.B. Startzeit festhalten oder Stückzähler auf Null setzen Eröffne Dateien für Protokoll und Daten
// Zentrale Steuerschleife do { Lies Signale von Sensoren, z.B. Lichtschranken, Thermometer, Füllstand, ... Berechne mit logischen Funktionen UND, ODER, NEGATION die Schaltstellungen für digital angesteuerte Aktoren (EIN oder AUS), wie z. B. Füllventile, Ablassventile, Deckelöffner, geschaltete Heizungen, Signallampen usw. Berechne nach DIN-Algorithmen die Stellgrößen für geregelte Aktoren, z.B. Regelventile, geregelte RührerMotoren, Dosierpumpen, geregelte Heizungen Gehe in eine Warteschleife, bis die festgelegte Zeit (Tastzeit) für einen Schleifendurchlauf um ist, z.B. TA=0,1 Sekunde }while (kein Stoppsignal und kein Fehler vorliegt)
Aufräumarbeiten , z.B. Endstände von Zählern abspeichern, Dateien abschließen (close), Ventile schließen, Motore abschalten, usw.

Einige Dinge, die es beim μ -Controller zu beachten gilt, sind z.B.:

- Man braucht ein Netzgerät mit mehreren Spannungspegeln (5 V, 12 V, ...)
- Man braucht eine Entwicklungsumgebung (einen PC mit C-Compiler, Bibliotheken)
- Man braucht ein Gerät zum Übertragen des Programms in den μ -Controller (Brenner)
- Der μ -Controller lässt sich nach einem Löschvorgang (erase) beliebig oft neu brennen
- Die Spannungen und Stromstärken der Ausgangssignale sind schwach (<5 Volt, <1 mA)
- Der μ -Controller ist empfindlich für Überspannungen
- Der μ -Controller rechnet langsamer als ein PC

Siemens Logo! Als Beispiel einer Zwischenstufe zwischen μ -Controller und SPS. Dieses kleine Bauteil ersetzt Schütze und Relais im Schaltschrank (z.B. für Verriegelung von Schutzeinrichtungen), kann kleinere 1-phasige Lasten schalten, spart dadurch Verdrahtung im Schaltschrank. Hat Analogausgänge und sogar PI-Regler, z.B. für Heizung, Druck, Füllstand.

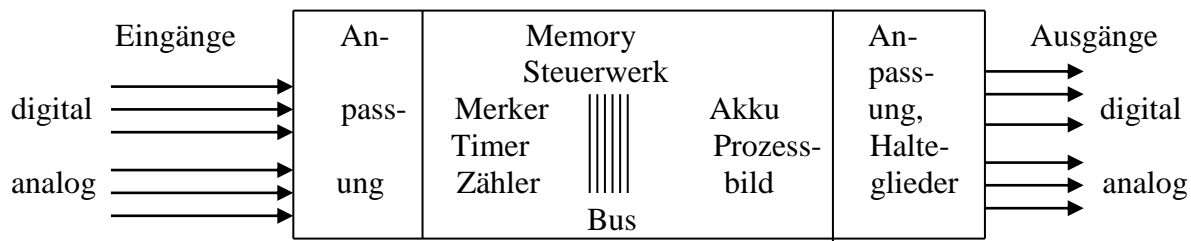
2.2 Anlagenautomatisierung mit SPS (Nur Steuerung)

Die SPS ist ein μ -Controller in robuster Ausführung mit einer sehr leistungsfähigen eingebauten Signalkonditionierung. Die SPS kann **steuern**, regeln und kommunizieren.

2.2.1 Allgemeines zur SPS

Die SPS ist das Arbeitspferd der Automatisierer. Sie besteht aus einem robusten Rechner und sehr leistungsfähigen eingebauten Signalkonditionierern. Der externe Spannungspegel für digitale Signale (0-1-Signale) ist 24 Volt. Die Stromstärke kann bei der Ausgabe bis zu 1 Ampere betragen. Analoge Daten werden ebenfalls verarbeitet im Spannungsbereich -10 bis +10 Volt (bzw. bei Stromsignalen 4 bis 20 mA Industriennorm). Die SPS ist entweder in einem Schaltkasten vor Ort direkt neben der Anlage untergebracht, oder aber in einem Schaltschrank in einer separaten Schaltwarte. Die Prozesssteuerung erfolgt zumeist über ein Touchpad.

Aufbau einer SPS



Anpassung: Bei der digitalen Eingabe (0/1-Werte) wird der externe 24-Volt-Pegel auf den internen 2-3 Volt-Pegel transformiert und eine galvanische Trennung realisiert. Bei Analog-Eingängen (-10 Volt bis +10 Volt) wird auf den 50 mV-Bereich des AD-Wandlers transformiert, bei analogen Stromeingängen (4-20 mA) ebenfalls. Bei der digitalen Ausgabe muss der Spannungspegel wieder auf 24 Volt erhöht werden, mit einem Halteglied eine Tastzeit lang gehalten werden, und notfalls Stromstärken bis zu 1 A geliefert werden. Die analoge Ausgabe arbeitet mit Pegel -10 bis +10 Volt bzw. Stromstärken von 4-20 mA.

Memory: Speicher für das Programm und für Datenblöcke

Steuerwerk: Macht die Befehlsverarbeitung, d.h. die logischen Operationen, Zahlenrechnungen und die Kommunikation (Drucker, Bildschirm, Internet)

Akku (Akkumulator): Speicher- und Rechenregister, bestehend aus zwei 16-Bit-Worten (Akku1 und Akku2). Vor dem Laden eines Wertes nach Akku1 wird der alte Inhalt nach Akku2 verschoben, dessen Inhalt dabei verloren geht.

Merker: Extra Datenspeicher zusätzlich zum Memory. Merker sind adressierbar als Worte (16 Bit), als Bytes (8 Bit) und als einzelnes Bit (1 Bit). Ein 1-Bit-Merker kann sich eine 0 oder eine 1 merken.

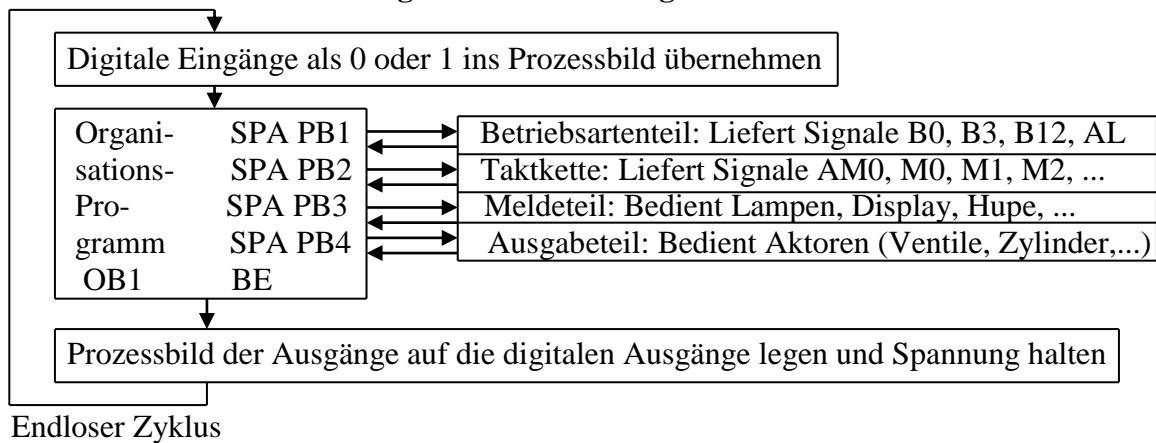
Prozessbild: Speicher mit je einem Bit für jeden digitalen Eingang bzw. Ausgang. Die Augenblickliche Spannungsführung (0 bzw. 24 Volt) wird hier abgebildet als 0 oder 1.

Timer, Zähler: Extra Speicherworte für Zeitdaten bzw. Zählerstände.

Bus: SPS-interne Daten- und Steuerleitung mit Bus-Treibern an jedem Ende.

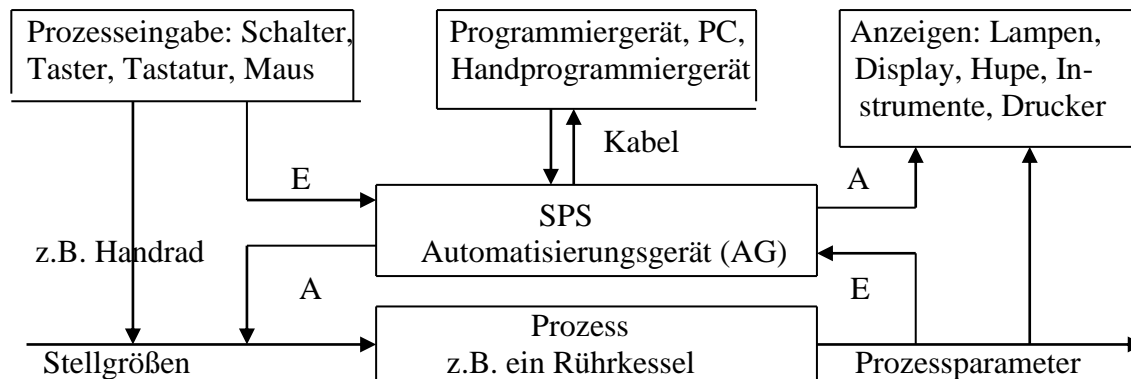
Als **Programmiergerät** wird heute fast ausschließlich ein PC (Laptop) genommen. Es gibt neben den Herstellerprogrammen auch eigenständige Softwarefirmen. Z.B. WinSPS-S7 gestattet nicht nur das Programmieren, sondern auch Trockentests ohne SPS und ermöglicht beim Zusammenspiel mit der SPS z.B. die Beobachtung von Speicherinhalten (wichtig bei Fehlersuche im Programm)

Programmverarbeitung einer SPS



<i>SPA PB1</i>	Maschinenbefehl, bedeutet <i>Sprung absolut zum Anwenderprogramm PB1</i>
<i>BE</i>	Maschinenbefehl und heißt <i>Baustein-Ende</i> (der RETURN-Befehl)
<i>B0</i>	Richtimpuls, entsteht beim ersten Drücken der Starttaste
<i>B3</i>	Startimpuls, entsteht beim zweiten Drücken der Starttaste
<i>B12</i>	Stoppperker (wird beim Start 0 gesetzt, mit der Stopptaste auf 1 gesetzt)
<i>AL</i>	Signal <i>Automatik läuft</i> (AL=1) als Dauersignal im Automatikbetrieb (Alternative ist der Handbetrieb mit AL=0)

Zusammenspiel Prozess mit SPS



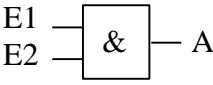
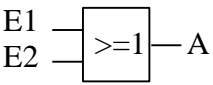
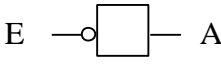
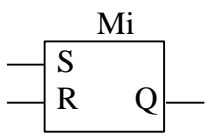
2.2.2 FUP-Programmierung (Funktionsplan)

Es gibt 3 Programmierarten, um ein SPS-Programm zu erstellen:

- AWL oder Anweisungsliste: Man schreibt Maschinenbefehle (Profis arbeiten so)
- FUP oder Funktionsplan: Man vernetzt Boxen (logische oder andere Operationen)

- KOP oder Kontaktplan: Sieht aus wie eine Stromschaltung mit Relais (in den USA üblich)

Ein **Programm** besteht aus **Bausteinen** (Teilprogrammen), diese aus **Netzwerken**, diese aus Linien und Boxen. Die Logik arbeitet mit 0/1. Einige Boxen sind:

	UND-Box: Der Ausgang wird nur 1, wenn <u>alle</u> Eingänge 1 sind
	ODER-Box: Der Ausgang wird nur 0, wenn <u>alle</u> Eingänge 0 sind
	Negation: Ist der Eingang 1, wird der Ausgang 0 Ist der Eingang 0, wird der Ausgang 1
	Flipflop (Set-Reset-Box): Eine 0-1-Flanke am Set-Eingang setzt das Flipflop (Merker Mi und Schaltausgang Q) auf den Wert 1. Der Wert bleibt, auch wenn das Signal am Set-Eingang auf 0 zurückfällt. Eine 1 am Reset-Eingang setzt Mi und Q auf 0 zurück. Merker Mi kann auch aus anderen Netzwerken abgefragt werden.

Timer sind eine Art Kurzzeitwecker. Die Einschaltverzögerung z.B. meldet sich mit einem 1-Signal, wenn die eingestellte Zeit abgelaufen ist. Die Laufzeit wird bei der alten Serie S5 von Siemens-SPS-en als Timerwort in Form einer Zeitkonstanten eingegeben.

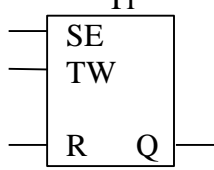
Beispiel: **KT 500.1** verschlüsselt die Zeit 50 Sekunden (500 mal 0,1 Sekunden)

KT steht für Konstante Timer

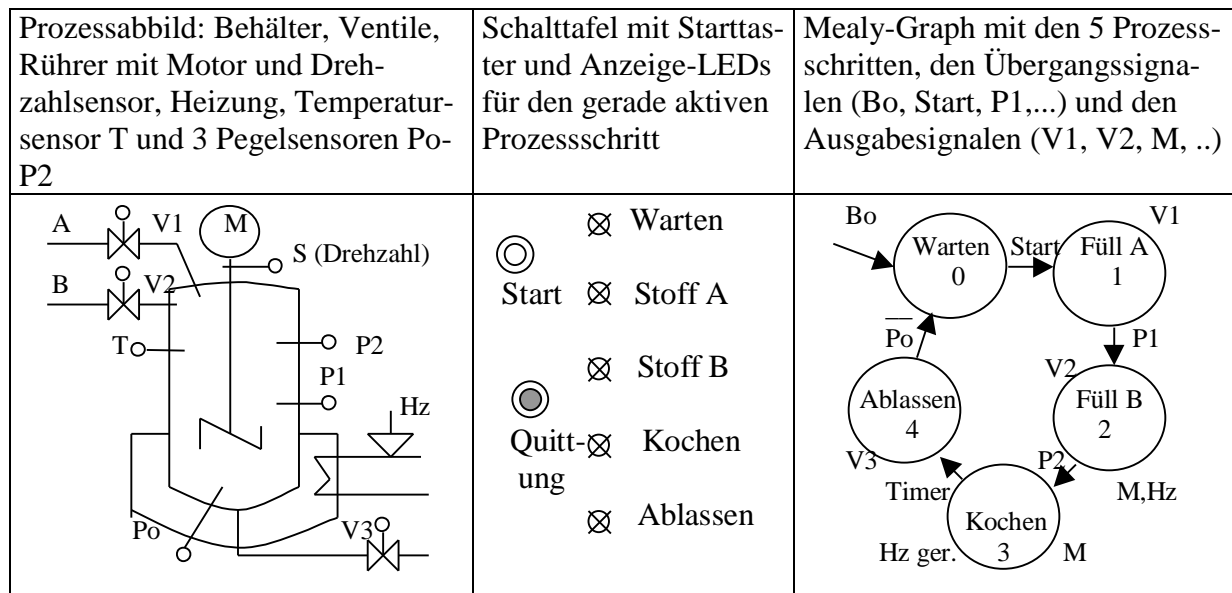
500 ist der Zeitfaktor (Werte von 1–999 sind möglich)

.1 ist das Zeitraster (**.0**=0,01 Sekunde **.1**=0,1 s **.2**=1 s **.3**=10 s)

Wichtig: Das Zeitraster möglichst klein wählen, damit die Zeit genau eingehalten wird (KT005.3 z.B. hätte die fatale Folge, dass jede Zeit zwischen 40–50 Sekunden auftreten kann)

	Die Einschaltverzögerung (eine von 5 Timerarten): Eine 0-1-Flanke am SE-Eingang startet den Timer. Q und Timerbit Ti sind anfänglich 0. Das SE-Signal muss mindestens die Zeit TW auf 1 bleiben, denn erst dann schalten Q und Ti auf den Wert 1. Eine 1 am Reset-Eingang stoppt den Timer, d.h. Q und Ti werden bzw. bleiben 0. Geht das SE-Signal auf 0 zurück, wirkt das wie ein Reset (Q=Ti=0). Timerbit Ti kann aus anderen Netzwerken abgefragt werden (es ist immer Q=Ti)
---	--

2.2.2.1 Unser Beispiel sei ein Prozess, bei dem aus Stoff A (Fruchtsirup) und Stoff B (Pektinlösung) Gelee gekocht wird. Prozessbild, Schalttafel und Zustandsgraph sind:

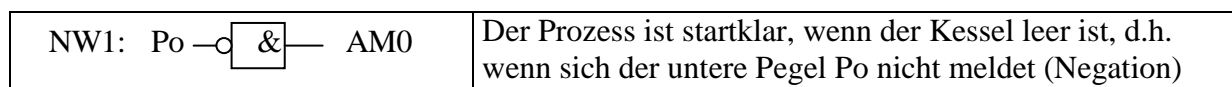


Ein typisches SPS-Steuerprogramm (der S5-Serie) besteht aus den 5 Programmbausteinen:

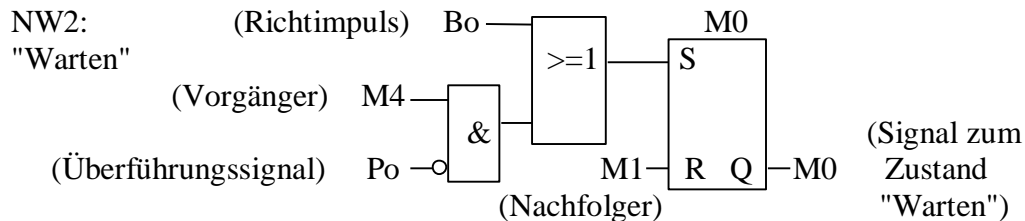
1. PB1 (**Betriebsartenteil**, erzeugt Signale wie Richtimpuls Bo, Startsignal B3, Stoppmerker B12. Dieser Teil ist unveränderlich und wird unbeschadet aus alten Programmen übernommen)
2. PB2 (**Taktkette**. Das erste Netzwerk, AM0 oder Starterlaubnis, gehört eigentlich in den Betriebsartenteil. Da aber prozessspezifische Signale ausgewertet werden, steht es immer in der Taktkette. Die weiteren Netzwerke setzen den Zustandsmerker M0, M1, M2, ... des gerade aktiven Zustandes, und schalten den Merker wieder auf den Wert 0 zurück, wenn der Prozessschritt abgelaufen ist.
3. PB3 (**Meldeteil**. Hier werden Lampen, Hupen, Warnleuchten, Drucker- und Bildschirmausgaben, Kommunikation mit einem Leitsystem usw. programmiert)
4. PB4 (**Ausgabeteil**. Alle Aktoren, wie Ventile, Motore, Stellzylinder, Heizungen usw. werden durch jeweils genau ein Netzwerk angesteuert)
5. OB1 (**Organisationsbaustein 1** enthält die Aufrufe der 4 Bausteine PB1 – PB4. OB1 selbst wird zyklisch vom Betriebssystem der SPS aufgerufen. Wie oft in der Sekunde, das hängt von der Programmlänge und der Leistungsfähigkeit der CPU ab)

2.2.2.2 Programmbaustein PB2 (Taktkette)

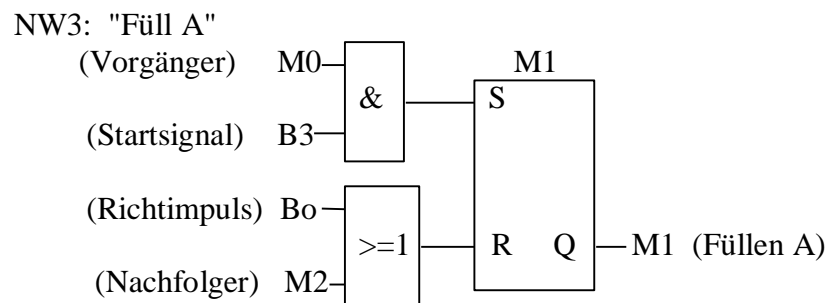
Der Baustein liefert die Signale AM0, M0, M1, M2, ... Signal AM0=1 heißt, *der Prozess ist startklar*. Dabei wird z.B. gecheckt, ob der Deckel geschlossen ist, ob der Behälter leer ist, ob Druck anliegt usw. Ein einfaches Beispiel für Netzwerk 1, das AM0-Netzwerk, ist



Die restlichen Netzwerke von PB2 realisieren die Taktkette (so etwas wie eine Schaltwalze, die einen Prozessschritt nach dem anderen aktiviert und wieder deaktiviert): Der aktive Zustand M_i liefert ein 1-Signal. Wird Zustand M_i (Prozessschritt M_i) beendet, dann reicht er die Stafette an seinen Nachfolger weiter. Dieser wird aktiv (und beendet das Aktivsignal seines Vorgängers. (Alle im folgenden Text gezeigten Netzwerke sind nur für Automatikbetrieb. Die Kopplung Hand/Automatik ist etwas aufwendiger zu programmieren.)

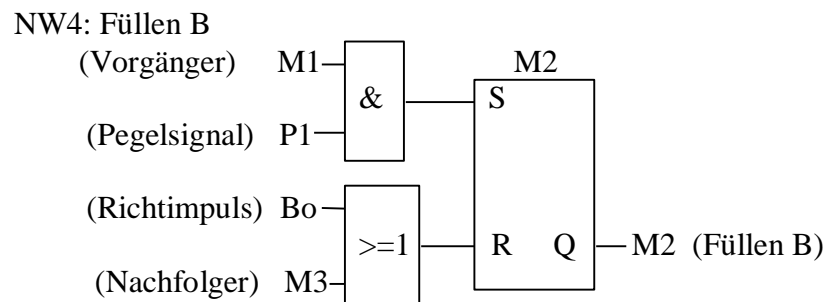


Zustand "Warten" (Merker $M0$) wird entweder durch den anfänglichen Richtimpuls Bo beim Start der SPS eingeschaltet oder aber, wenn sich im Zustand $M4$ (Prozessschritt Ablassen) der untere Pegel nicht mehr meldet, d.h. wenn der Kessel leergelaufen ist. Der Nachfolgezustand $M1$ (Befüllen mit Stoff A) schaltet den Wartenmerker $M0$ ab ($M0=0$).

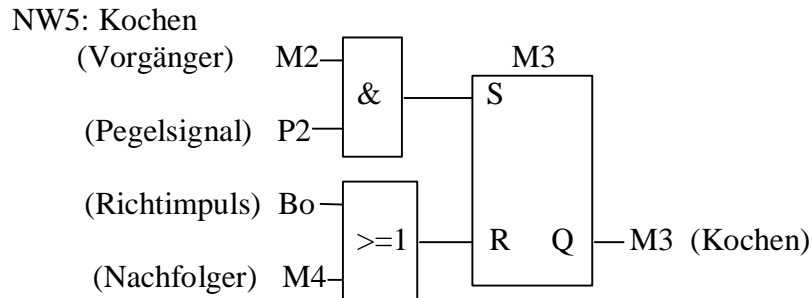


Wenn Zustand Warten aktiv ist ($M0=1$) und die Starttaste gedrückt wird (Signal $B3=1$), wird das Flipflop für Zustand 1 gesetzt (Merker $M1=1$). $M1$ wird Null gesetzt durch den anfänglichen Richtimpuls, oder aber durch den Nachfolgezustand $M2$.

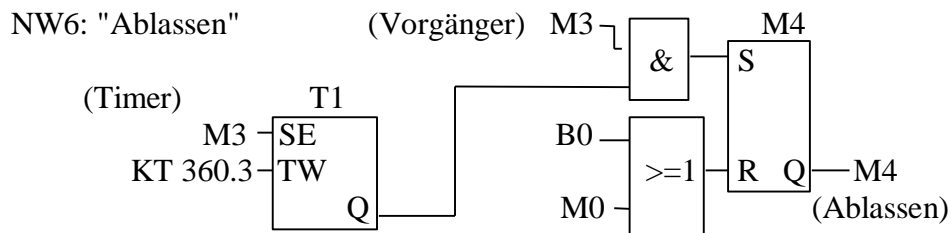
Das Netzwerk für den nächsten Takt ("Füllen B") sieht fast gleich aus: Wenn Zustand "Füllen Stoff A" aktiv ist ($M1=1$) und sich Pegelmelder $P1$ meldet (Signal $P1=1$), wird das Flipflop für Zustand 2 gesetzt (Merker $M2=1$). $M2$ wird Null gesetzt durch den anfänglichen Richtimpuls, oder aber durch den Nachfolgezustand $M3$.



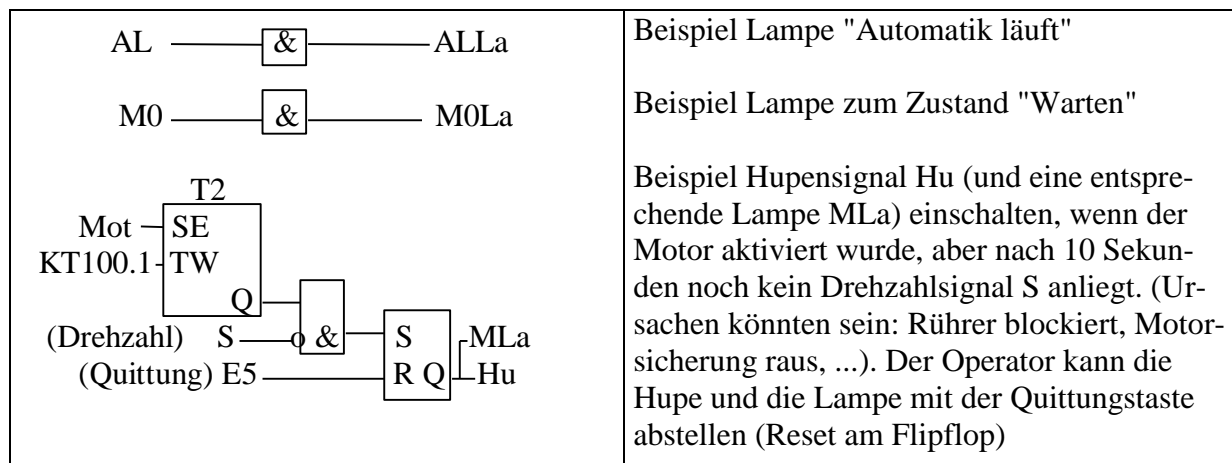
Zustand Kochen schaltet sich ein, wenn Pegelmelder $P2$ meldet, dass beide Stoffe befüllt sind.



Zustand Ablassen, der letzte Zustand der Taktkette, wird durch die abgelaufene Kochzeit T1 eingeschaltet. Die Timerkonstante KT360.3 entspricht einer Stunde Timerlaufzeit. Vom Zustand Ablassen geht es zurück zum Warten (Zustand M0, siehe Netzwerk 2).

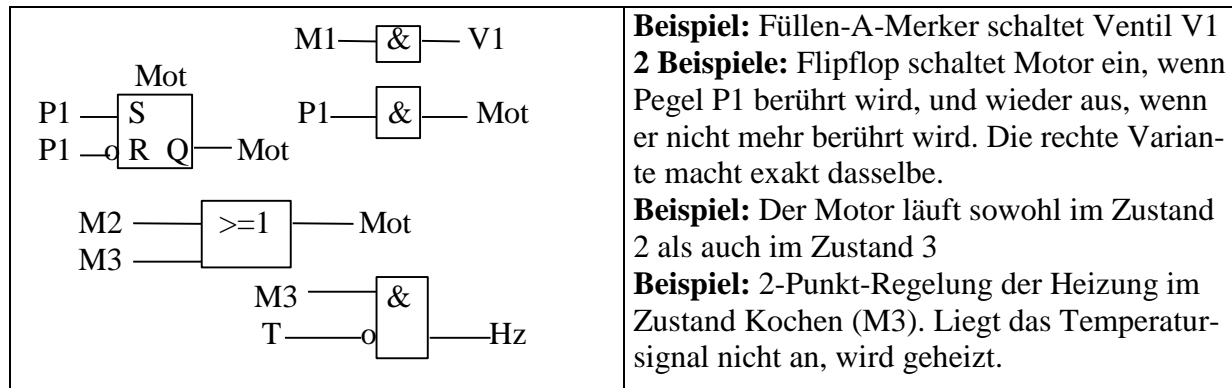


2.2.2.3 Meldeteil PB 3: Lampen, Hupe. Es wird Quittungstaste E5 verwendet (Schalttafel)



2.2.2.3 Ausgabeteil PB4: Ventile, Heizung, Rührermotor

Für jeden Aktor darf nur ein Ausgang programmiert werden (in der Regel ein Netzwerk für jeden Aktor). Oft gibt es mehrere Programmiermöglichkeiten, die gleichwertig sind.

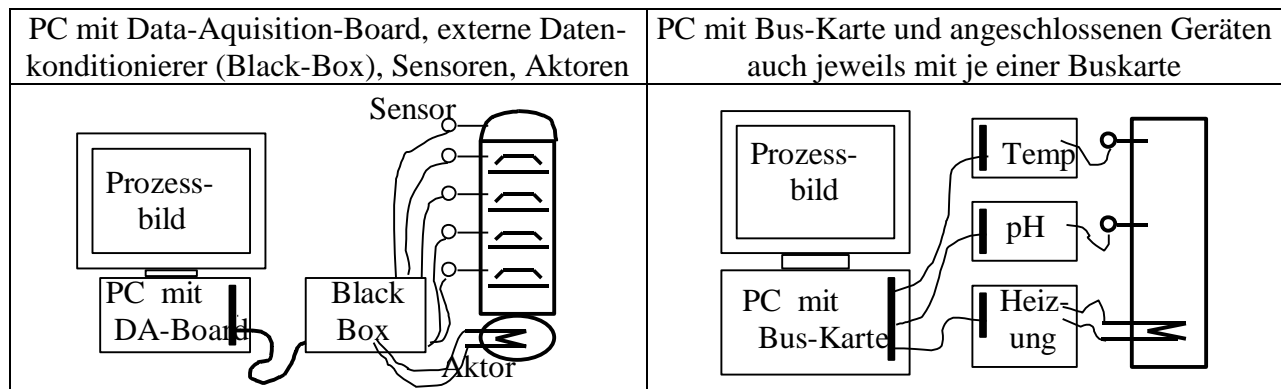


Organisationsbaustein OB1 ist oben unter Programmverarbeitung einer SPS erklärt worden. Er wird nicht als FUP, sondern in AWL, d.h. in der Maschinensprache, programmiert.

2.3 Automatisieren mit PC

2.3.1 Laborautomatisierung mit PC

Für eine Laborautomatisierung werden fast ausschließlich aufgerüstete PCs verwendet. Um Daten von und zu Geräten oder Versuchsaufbauten zu transportieren, ist zusätzliche Hardware nötig. Übliche Varianten sind z.B.:

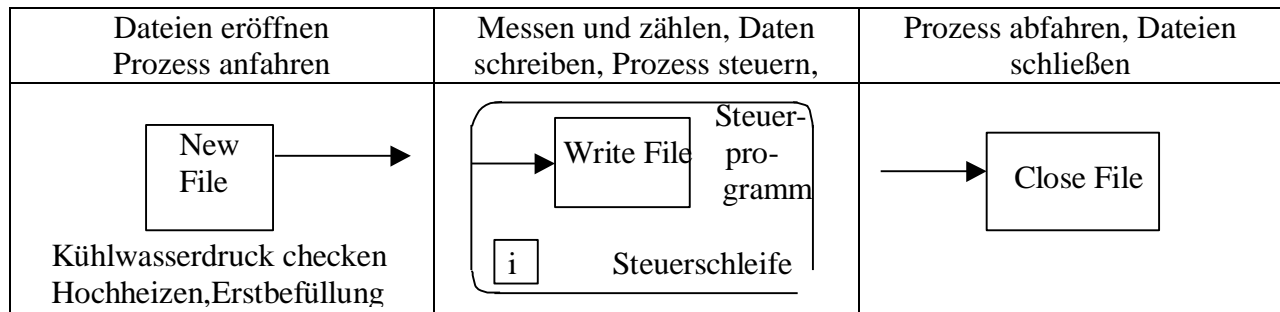


Es gibt zahlreiche Steuerprogramme, die auf den PCs laufen. Sie kommunizieren über das Data-Aquisition-Board oder über USB oder über eine Bus-Karte mit dem Prozess. Immer dazwischen geschaltet sind Signalkonditionierer, die verschiedene Funktionen haben, z.B.

- Verstärkung schwacher Signale bei Eingängen, Relais oder Verstärker bei Ausgängen
- Glättung und Filterung der Signale gegen Rauschen oder Spannungsspitzen
- Schutz vor Überspannungen
- Netzteile für die Stromversorgung

Eine typische Anwendung besteht aus einem Prozessbild auf dem Bildschirm des PC (der Bedienoberfläche) mit eingeblendeten Zahlenangaben, Graphiken von wichtigen Messwerten, Signallampen, Schaltern, Schieberegler usw., und einem Steuerprogramm im Hintergrund, das sowohl mit dem Prozessbild kommuniziert, als auch mit dem Prozess (Signale von Sensoren liest

und Aktoren ansteuert). Viele dieser Programme werden graphisch programmiert. Die Grundstruktur eines solchen Programms ist jedoch wieder eine Dreiteilung:



Industrie-PC sind heute voll gekapselt, haben keine beweglichen Teile (Lüfter, Laufwerke) und max. 10 Watt Leistungsaufnahme (z.B. mit Prozessor Intel ®Atom™ Z530 mit 1,1 GHz Taktfrequenz, 2,5 Watt Leistung) und damit sehr robust. Betriebssysteme sind Windows, CE, XP.

2.3.2 Beobachtermodelle

Sind Größen zu regeln, die man messtechnisch nur sehr schwer oder nur in großen Intervallen oder nur mit großem Zeitverzug erfassen kann, dann lässt man parallel zum realen Prozess ein Beobachtermodell mitlaufen (Luenberger 1964). Das Modell vertritt im PC zeitweise den realen Prozess. Es erhält laufend die leicht erfassbaren Messdaten des realen Prozesses und von Zeit zu Zeit die schwer erfassbaren. In den Messintervallen interpoliert das Modell alle Kurvenverläufe. Meist sind es Systeme gekoppelter DGL, die dabei berechnet werden. Im Beobachtermodell sind somit alle Messdaten laufend verfügbar, auch die Daten, die schwer messbar sind. Natürlich weichen die Daten des Beobachtermodells nach einer gewissen Zeit von denen des realen Prozesses ab. Ursachen sind Störungen im realen Prozess, Nichtkonformitäten von Modell und Realität und Rechenungenauigkeiten. Die ständige Minimierung dieser Fehler ist eine der Hauptaufgaben beim Betrieb eines Beobachtermodells.

2.3.3 Leitsysteme

Leitsysteme sind eine hierarchische Kombination aus einem Leitrechner (meist ein PC), untergeordneten Steuerrechnern (meist SPSen) und oder Industrieregler. Das Wesentliche am Leitsystem ist, dass auf dem Bildschirm oder den Bildschirmen des Leitrechners alle wichtigen Informationen zusammenlaufen. Der Prozess, z.B. eine Brauerei, ist auf verschiedenen Prozessbildern dargestellt, z.B. das Sudhaus, der Gärkeller, der Lagerkeller, die Abfüllanlagen, das Auslieferungslager, das Leergutlager, das Qualitätslabor. Per Mausklick kann der Betriebsleiter sich nicht nur alle Prozessstände ansehen (Füllstände, Temperaturen, Lagerbestände, Zahl der aktiven Gärtanks), sondern der Braumeister kann vom Leitrechner aus Batchprozesse starten, wie das Befüllen der Sudpfanne mit Wasser und Würze und das anschließende Kochen und Klären.

3. Kommunikation

Ein immer mehr wachsender Anteil an Automatisierungskosten fällt in das Gebiet der Kommunikation. Kommunikation beginnt bereits im Gerät, z.B. einem Messumformer direkt an einem Thermometer, wo die einzelnen elektronischen Bauelemente, wie Prozessor, Speicher, Display, durch interne Busverbindungen (Flachkabel mit Bustreibern an jedem Ende) verbunden werden.

Die nächste Stufe der Kommunikation ist die Anbindung von Sensoren und Aktoren, z.B. des Messumformers eines Thermometers oder des Stellungsreglers eines Ventils, über sogenannte Feldbusse an das nächstgelegene Steuergerät, zumeist eine SPS. Bis hierhin spricht man von der **Feldebene** (Anlagennaher Datenaustausch).

Eine weitere Stufe ist die **Kopplung SPS-Leitrechner**. Hier werden hauptsächlich Zahlen und Text ausgetauscht. Die Busse sind für die Übertragung großer Datenmengen ausgelegt. Die SPS meldet Zählerstände, Temperaturen, Ventilstellungen usw. an den Leitrechner. Dieser gibt Rezepturnummern, neue Sollwerte oder Startbefehle für Teilprozesse an die SPS. Diese Ebene wird oft als **MES-Ebene** (Manufacturing Execution System) bezeichnet. Von hier aus werden aber auch Produktionszahlen bzw. Ressourcenanforderungen an die ERP-Ebene gesandt und Aufträge entgegengenommen. Die **ERP-Ebene** (Enterprise Resource Planning) ist in der Geschäftsleitung angesiedelt. Dort werden Maschinenauslastung, Auftragsdurchlauf, Transport- und Lagerauslastung geplant.

3.1 Signale - Transport und Verarbeitung

Signalformen: Analoge Signale für sich kontinuierlich verändernde Größen, wie Temperatur, Druck, Drehzahl, pH, Trübung, Füllstand, Position, Konzentration, Leitfähigkeit usw.

Strom	meist 4-20 mA Industrienorm, geeignet für Entfernungen bis 500 m, Spannung schwankend 5–10 V
Spannung	meist –10 bis +10 V Industrienorm, geeignet für Entfernungen bis 10 m (Abschirmung ist wichtig), Stromstärke <1 mA
Druck	0,2 bis 1,0 bar Überdruck Industrienorm, nur noch in alten pneumatischen Anlagen zu finden

Digitale Einzelsignale (Binäre Signale) für mechanische, elektrische oder elektronische Elemente, die nur zwei Werte liefern bzw. annehmen können, wie z.B. Schalter, Taster, Lichtschranken, Stellung eines Absperrventils, Signallampen, Pegelmelder usw.

Strom	0 oder 20 mA Industrienorm
Spannung	0 oder 24 V Industrienorm

Digitale Signale für beliebige Größen und Nachrichten. In Form von Zahlen und Texten, die binär verschlüsselt übertragen werden, können alle oben genannten Informationen übertragen werden. Zusätzlich können in Textform noch Befehle an sogenannte intelligente Messumformer oder Industrieregler übertragen werden, um z.B. Alarmpegel oder Sollwerte zu verändern, oder ein intelligenter Sensor liefert eine Fehlermeldung im Klartext, z.B. "Hilfsspannung U4 ausgefallen".

Wandler: Analoge Signale werden in den Messumformern, μ -Controllern, SPSen oder im Industrieregler bei der Eingabe in Zahlen umgewandelt bzw. bei der Ausgabe aus einer Zahl rückgewandelt. Wir unterscheiden zwei grundsätzlich verschiedene Wandlerarten:

1. Das Wägeverfahren speichert mit einem elektronischen Halteglied den Augenblickswert U_x eines Strom- oder Spannungssignals als Aufladung eines kleinen Kondensators. Dann beginnt eine elektronische Schaltung nacheinander Vergleichsspannungen zu erzeugen, z.B. $U_7=5$ V. Ist die Spannung U_x im Kondensator größer oder gleich U_7 , wird Bit 7 der zu erzeugenden Binärzahl 1 gesetzt, sonst 0. Im nächsten Schritt wird in Abhängigkeit von der Bitbelegung die nächste Vergleichsspannung erzeugt, $U_6=2,5$ oder $U_6=7,5$ V. Jetzt wird wieder mit U_x verglichen und das Bit 6 gesetzt usw. bis alle 8 Bits gesetzt sind. Standard ist 8 oder 12 Bit, selten mehr bei den Wägeverfahren. Wägeverfahren arbeiten sehr schnell. Man erreicht leicht 10.000 Wandlungen pro Sekunde. Der Nachteil ist, dass Rauschen oder Spannungsspitzen die Messung beeinträchtigen, da Augenblickswerte gewandelt werden.
2. Das integrierende Verfahren lädt einen etwas größeren Kondensator mit der Messspannung U_x auf. Welligkeiten, Rauschen, Spannungsspitzen werden integriert und damit gemittelt. Über eine kalibrierte Gegenspannung wird der Kondensator über einen definierten Entlade-widerstand entladen. Die Zeit bis zum Nulldurchgang der Spannung wird mit einem Schwingquarz gemessen. Die Zeit ist proportional zum Messwert. Die integrierenden Verfahren sind genauer (bis 20 Bit), aber dafür auch deutlich langsamer (10 – 100 Messungen pro Sekunde).

Zahlen: Treten als Konstanten, Merkmalswerte oder Signalwerte in den Rechnern auf. Intern werden Binärzahlen verwendet, extern werden Zeichenketten verwendet. Binäre Zahlen haben die Basis 2 im Gegensatz zum Dezimalsystem mit Basis 10.

Beispiel: $137 = 128 + 8 + 1 = 1 \cdot 2^7 + 0 \cdot 2^6 + 0 \cdot 2^5 + 0 \cdot 2^4 + 1 \cdot 2^3 + 0 \cdot 2^2 + 0 \cdot 2^1 + 1 \cdot 2^0$
 oder binär 1 0 0 0 1 0 0 1

Bei negativen Zahlen benutzt man zumeist das 2-er-Komplement. Um die Zahl -137 darzustellen, beginnt man mit der Binärdarstellung der $+137$. Wir benutzen ein 16-Bit-Wort (2 Byte)

Binärdarstellung der positiven Zahl	0 0 0 0 0 0 0 0 1 0 0 0 1 0 0 1
Negation jedes einzelnen Bits	1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 0
Addition einer 1 nach den Regeln des binären Rechnens	1 1 1 1 1 1 1 1 0 1 1 1 0 1 1 1

Gleitkommazahlen, d.h. solche mit Dezimalstellen und/oder einem Exponenten, werden ebenfalls intern binär verschlüsselt. Hier werden jedoch die eigentliche Zahl (Mantisse) und der Exponent getrennt als Binärzahlen behandelt.

Für die Ausgabe auf ein Display müssen die Zahlen von der Binärdarstellung wieder in die Dezimalform gebracht werden. Hierbei wird jede Ziffer und jedes Zeichen der Dezimalzahl einzeln als druckbares Zeichen mit einem Zeichencode verschlüsselt (Z.B. der ASCII-Code (American Standard Code for Information Interchange)). Es entstehen Zahlen-Zeichenketten wie:

3 -77 3.14 27E+03 (lies $27 \cdot 10^3$) -3.14e-22 (lies $-3.14 \cdot 10^{-22}$)

Signaltransport: Die Transportmöglichkeiten für Signale unterscheiden sich u.a. im Preis, Übertragungsrate, Reichweite und Anfälligkeit gegen Fehler

Transportart	Vorteile	Nachteile
2-Draht-Stromleiter verdreht	billig, bis 500 m, wenig Fehler	niedrige Rate
abgeschirmtes Spannungskabel	billig, bis 10 m, öfters Fehler	niedrige Rate
Hochfrequenz-Kabel (HF)	hohe Rate, bis 100 m, wenig Fehler	teuer
Lichtleiter	hohe Rate, bis 1000 m, kaum Fehler	teuer
Funkverbindung	hohe Rate, bis 10 km, wenig Fehler	teuer

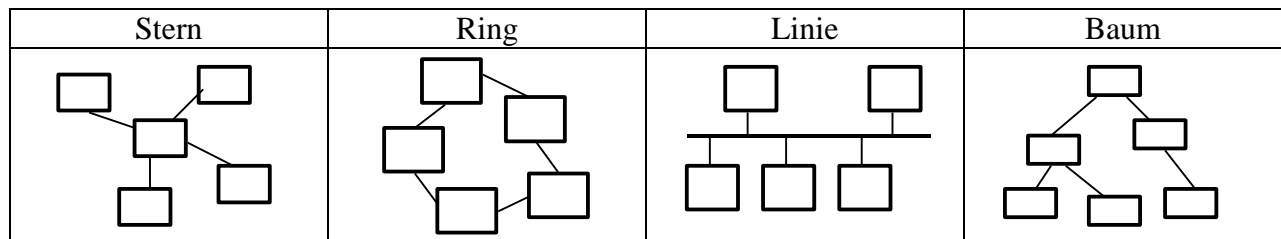
3.2 Netze

Netze dienen den Verbindungen Rechner - Sensor, Rechner - Aktor, und Rechner - Rechner.

Ein **LAN** (Local Area Network) darf rechtlich nur auf einem Grundstück arbeiten.

Ein **WAN** (Wide Area Network) darf weltweit agieren unter Beachtung der Posthoheit der betroffenen Länder.

Netztopologien sind Stern, Ring, Linie und Baum. Es gibt aber auch Mischformen.



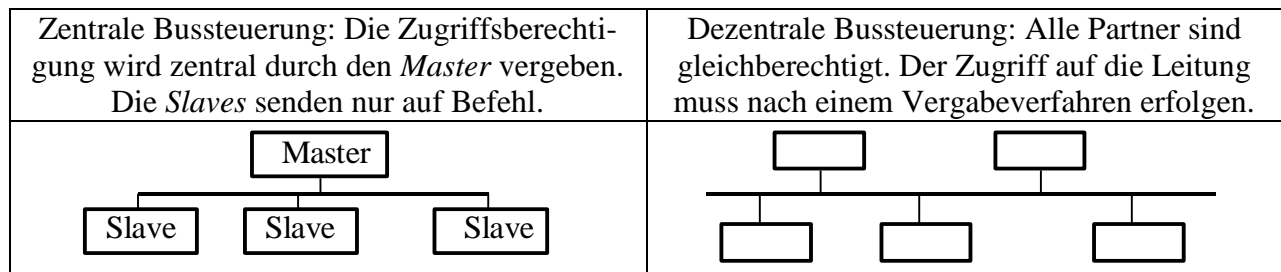
Übertragungsverfahren sagen, wie die Signale im Leiter übertragen werden (3 Beispiele):

Basisband: 0-1-Signale werden als Spannungspegel übermittelt, z.B. 0 V - 5 V

Trägerband: 0-1-Signale werden als 2 unterschiedliche Frequenzen ω_0 und ω_1 übertragen

Breitband: 0-1-Signale werden als 2 Frequenzen $\omega_0^{(i)}$, $\omega_1^{(i)}$ aus Frequenzband $i=1, \dots, n$ übertragen, d.h. die Leitung kann mehrere Kanäle gleichzeitig übertragen.

Zugriffsverfahren: Eine Leitung, an der viele Geräte hängen, bedarf einer exakten Regelung, wann welches Gerät senden darf.



Zugriffsverfahren (Vergabe der Zugriffsrechte auf die Leitung) (Beispiele):

1. Zentrale Steuerung: Eine Umlaufliste und eine Alarmliste im Masterrechner legen die Reihenfolge der Zugriffsberechtigungen fest. Diese werden in festen Zeittakten vergeben. Ein Slave darf aber in der Liste mehrfach auftreten, was seinen Zugriff verbessert
2. Demand Priority (Basiert auf der DPAM=Demand Priority Access Methode). Alle Geräte sind am Master angeschlossen. Will ein Slave senden, muss es erst eine Anfrage an den Master senden. Der vergibt daraufhin das Zugriffsrecht. Treten mehrere Verbindungsaufträge gleichzeitig auf, werden sie nach Reihenfolge und Priorität des Eingangs vergeben.
3. CSMA/CD (Carrier Sense Multiple Access with Collision Detection): Jeder darf senden, wenn die Leitung frei ist. Bei gleichzeitigem Start zweier Geräte wird die Kollision entdeckt. Jedes Gerät wartet dann eine Zufallszeitspanne lang auf seinen Neustart.
4. CSMA/CA : Baut auf dem CSMA/CD auf, versucht aber Kollisionen zu vermeiden. Sendewillige Geräte schicken Zeichen ins Netz, die von den anderen Geräten beachtet werden. Auch hier spielt die Priorität eine Rolle, z.B. Gerät 001 kommt vor 002.
5. Token Passing (Newhall-Verfahren): Ein Bitmuster (Token genannt) wird von Rechner zu Rechner weitergereicht. Nur der darf senden, der gerade das Token hat. Die zu sendende Information und die Adressen des Zielrechners und des Absenders (d.h. ihre Rechnernummern) werden an das Token angehängen. Der Empfänger setzt in die Nachricht ein Quittungsbit und reicht das Token mit seinem Anhang weiter. Der Absender löscht seine quitierte Nachricht im Tokenanhang, wenn sie wieder bei ihm vorbei kommt.

Die zentrale Steuerung wird bei hoher Netzauslastung empfohlen.

CSMA/CD wird bei niedriger Netzauslastung empfohlen.

Token Passing wird bei mittlerer Netzauslastung empfohlen

Begriffe aus der Netzwelt:

Repeater	sind in die Leitung eingebaute Signalverstärker bei langen Leitungen
Bridges	sind Verbindungsknoten (Rechner) zwischen unterschiedlichen Zugriffsverfahren, z.B. Token-Passing und Zentrale Steuerung
Gateways	sind Verbindungsknoten (Rechner) zwischen unterschiedlichen Protokollen (z.B. unterschiedliche Spannungspegel oder Signalfolgen beim Handshake)
Router	arbeiten wie Bridges, suchen aber über Adresslisten zusätzlich den Weg zum Zielrechner in einem vermaschten Netz, auch in WAN-Netzen (z.B. WWW)
Protokolle	sind Signalfolgen in einem Dialog zwischen Rechnern, die zur Aufnahme, Durchführung und Beendigung einer Datenübertragung notwendig sind.
Busse	sind Datenleitungen mit Bus-Controllern an jedem Ende
Feldbusse	sind serielle Busse zur Ankopplung von Sensoren und Aktoren an übergeordnete Rechner (z.B. an SPSen oder Leitrechner). Als Übertragungsmedium dient oft nur eine verdrehte elektrische Doppelleitung.
Master	sind aktive Buskomponenten, die auf den Bus zugreifen.
Slaves	sind passive Buskomponenten, die vom Master gesteuert werden.
Profibus	ist ein langsamer Kommunikationsbus. Es werden Texte, Befehle, Zahlen übermittelt. Der Profibus erfordert "intelligente" Slaves, d.h. solche mit eigener Rechenleistung. Es sind viele Stationen mit mehreren Mastern möglich.
Interbus-S	ist ein schnellerer zyklischer Abtastbus mit einem Master und vielen Slaves

CAN-Bus	ist asynchron, seriell, besonders im Automobil vertreten, benutzt CSMA/CA, Kupfer- oder Glasfaser, Multimaster-Architektur mit Übertragungsraten von ca. 1 MBit/s bei Entfernungen bis 500 m, Linientopologie bevorzugt.
ASI-Bus	(Actuator-Sensor Interface), ein Bus für die unterste Automatisierungsebene, Twisted-Pair-Kabel, bis zu 31 Slaves. Es gibt Master-Einschubkarten für SPSen und PCs. Diese fragen die Slaves zyklisch ab. Sternarchitektur bevorzugt.
Ethernet	Kabelgebundenes LAN-Netzwerk. Datenaustausch in Form von Datenrahmen. Festlegung von Kabeltypen, Steckern, Paketformate, Protokollen in Standards.
IGB	Integrated Bus ermöglicht zusätzlich zum Datenaustausch noch Internetzugang und Telefonverbindung für Mails und Nachrichten.

Wide Area Automation Controller bieten über verschiedene Kanäle Möglichkeiten der vernetzten Automation. Auf der Feldebene stehen die RTUs (Remote Terminal Unit) mit digitalen und analogen Eingängen, Relaisausgängen für Steuerung vor Ort. Die zentrale Steuereinheit (PC) enthält Ethernet-LAN, ISDN, GSM (Global System for Mobile Communications), USB, RS485 bzw. RS422, Web-Server (Internetzugang), FTP-Server (File Transfer Protocoll), File System (Datenbank)

4. Pneumatik

Pneumatik ist die Verwendung von Druckluft als Signalmedium und für Energietransport. Die Anfänge der Pneumatik reichen 150 Jahre zurück (Rohrpost 1853, pneumatische Bohrmaschinen für Tunnelbau und Stahlbau um 1870, Niethammer um 1900). Die Blütezeit der pneumatischen Steuerung und Regelung war zwischen 1900 und 1980, zuletzt hauptsächlich im Explosionsschutz von Anlagen. Heute werden fast keine pneumatischen Steuerungen mehr gebaut, wohl aber pneumatische Antriebe und Sensoren. Bei den Antrieben ist es die robuste und billige Bauart, bei den Sensoren der Selbstreinigungseffekt und die Unempfindlichkeit gegen elektromagnetische Störfelder, die auch heute noch eine Verwendung sinnvoll machen.

Hauptanwendungsgebiete der Pneumatik sind heute:

- Pneumatische Stellantriebe für Ventile und Klappen, Spannvorrichtungen, Auswerfer, Vorschubeinrichtungen z.B. an Bohrmaschinen
- Düsenspritzgeräte wie Sandstrahlen, Farbsprühen, Späne wegblasen
- Luftdruckhammer
- Pneumatische Regler werden im Fahrzeugbau z.T. noch eingesetzt z.B. zur Drehzahlregelung bei Dieselmotoren und als Stellungsregler in pneumatischen Ventilantrieben.
- Pneumatische Endtaster, Luftschranken, Ultraschallschranken

Für die Verfahrenstechnik und Biotechnologie ist die elektropneumatische Steuerung wichtig. Eine SPS oder ein Rechner erzeugt die elektrischen Steuersignale. Diese schalten elektropneumatische Ventile, und diese wieder die rein pneumatischen Aktoren

Die Vorteile der Pneumatik sind:

- Die a-priori gegebene Überlastsicherung bei Arbeitszylindern, denn nur die Kraft $F = A \cdot P$ (Fläche mal Druck) kann maximal wirken. Getriebe z.B. sind da gefährlicher.
- Geringe Wartung durch neue verschleißarme Werkstoffkombinationen (Metall - Kunststoff)

- Verfügbarkeit von standardisierten Bauelementen von vielen Herstellern, die auch von Laien ausgewechselt werden dürfen im Gegensatz zur Starkstromelektrotechnik
- Hohe Bewegungsbeschleunigung bei Arbeitszylindern
- Leichte Kraftregulierung durch Drosselventile in den Zu- oder Abluftleitungen

Nachteile der Pneumatik sind:

- Die hohen Luftkosten (Verdichtung, Reinigung, Transport durch Rohrleitungen)
- Der Geräuschpegel beim Luftauslass

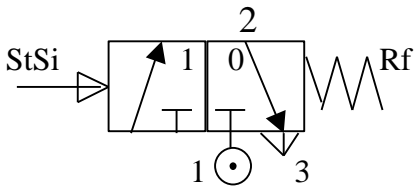
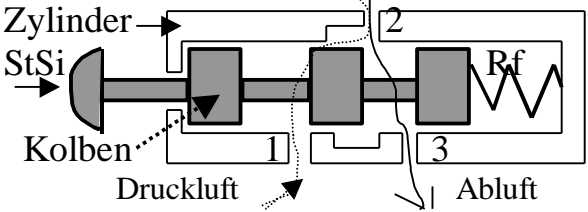
Die Luft muss auf **6 bis 8 bar** verdichtet werden und von Rost, Zunder, Staub, Wasser und Öl gereinigt werden. Man benutzt einen Zyklon und danach Filter mit Porendurchmesser 5-50 µm. Der Druckabfall im Filter liegt bei 0,2 bis 0,3 bar. In alten Anlagen werden noch **Öler** eingesetzt, die die Luft mit einem Ölnebel versetzen. Da Öl giftig ist, arbeiten neuere Anlagen ölfrei.

Leitungen sind entweder starr aus Metall oder flexibel aus Kunststoff. Starre Verbindungen sind verschraubt (Fittings), geflanscht, gelötet, geschweißt oder geklebt. Flexible Verbindungen haben fast immer lösbare Schnellkupplungen (Plug and go).

Ventile sind das wichtigste Steuerelement der Pneumatik:

- Als Regelventil für den Arbeitsdruck in den Druckluftleitungen der Pneumatik
- Als Sperr- oder Entlüftungsventil für einen Schnellstopp der Anlage
- Als Steuerventil, das je nach Schaltstellung unterschiedliche Luftwege durchschaltet
- Als Leistungsventil, das für große Aktoren große Luftmengen steuert

Beispiel eines Steuerventils: Monostabiler 3/2-Wegekolbenschieber. Der Name sagt, dass das Ventil 3 Luftanschlüsse (1, 2, 3) hat und zwei Schaltstellungen (0 und 1) hat. In der Schaltstellung 0 ist die Entlüftung (3) an die Arbeitsleitung (2) angeschlossen, in Schaltstellung 1 hingegen die Druckluft (1). Die Schaltzeit bei elektromagnetischem Antrieb ist 8–25 ms. Das Spiel des Kolbens im Zylinder beträgt 0,002 – 0,004 mm.

<p>Symbol: Steuersignal StSi, Druckluft 1, Arbeitsleitung 2, Abluft 3, Rückstellfeder Rf</p>	<p>Mögliche Konstruktion: Im Nullzustand ist Arbeitsleitung 2 mit Abluft 3 verbunden. Bei gedrücktem Knopf ist Druckluft 1 mit der Arbeitsleitung 2 verbunden.</p>
	

Betätigungsarten des Ventils (Steuerantrieb einzeln oder kombiniert)

Vorsteuer- oder Pilotventile ermöglichen es mit kleinen Leistungen größere Arbeitsventile zu schalten. Tasterrollen hingegen fahren auf einer Schablone entlang und steuern so das Ventil.

Durch Luftdruck vorgesteuert	Durch Luftdruck direkt	Elektromagnet stoßend	Elektromagnet stoßend + ziehend	Rückstellen mit Federkraft

Muskelkraft	Schlagknopf	Tasterrolle	Motor

4.1 Logik- und Schaltzeichen (Pneumatische Schaltzeichen nach DIN-ISO 1219)

Ruhestellung hat Nummer 0 Fehlt definierte Ruhestellung, dann definiert man die Ausgangsstellung Pro Schaltstellung zeichnet man einen Kasten Die Anschlüsse werden am Kasten 0 gezeichnet	
---	--

Verschiedene Steuerventile:

2 / 2	3 / 2	4 / 2	5 / 2

Bezeichnung der Leitungen

Art der Leitung	Symbol	Nummer ISO 5599 (alte Bez.)
Druckluft		1 (P)
Arbeitsleitungen		2, 4 (A, B)
Entlüftungen		3, 5 (R, S)
Steuerleitungen		12, 14 (z, y)

Steuersignal 12 (sprich eins-zwei) verbindet Anschluss 1 mit Anschluss 2

Steuersignal 14 (sprich eins-vier) verbindet Anschluss 1 mit Anschluss 4

Pneumatische Verknüpfungen (Auswahl)

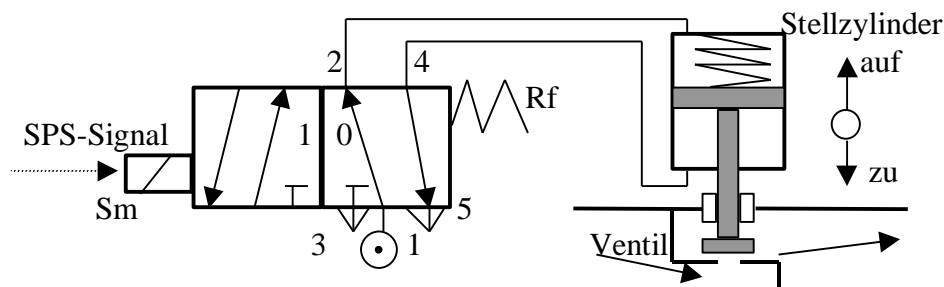
Wir unterscheiden logische Schaltzeichen nach DIN 40900 und pneumatische Schaltzeichen nach DIN-ISO 1219. Steuerventile für logische Aufgaben werden heute nur noch dort eingesetzt, wo die Elektropneumatik teurer wäre und keine Zeitvorteile bringt. Bei der Konjunktion (logisches UND) zeigt das pneumatische Schaltzeichen eine bewegliche Sperre aus 2 Scheiben, mit einem Abstandshalter. Bei der Disjunktion (logisches ODER) rollt im pneumatischen Schaltzeichen eine Absperrkugel in einen der Trichter, oder flattert in der Mitte. Identität und Haftspeicher sind 3/2-Steuerventile. Signale x , x_1 , x_2 , y sind Steuersignale (passiv), A ein Arbeitssignal (aktiv).

Funktion	DIN 40900	DIN-ISO 1219
$x = A$ (Identität) Steuersignal x als Leistungssignal A durchgeschaltet		
$x_1 \wedge x_2 = y$ (Konjunktion oder logisches UND) erzeugt ein Steuersignal (schwach)		
$x_1 \vee x_2 = y$ (Disjunktion oder logisches ODER) erzeugt ein Steuersignal (schwach)		
Haftspeicher (Flipflop) schaltet ein starkes Arbeitssignal A (x_1 =Set- und x_2 =Resetsignal)		

4.2 Elektropneumatische Ventilsteuerung

Digital gesteuerte Ventile (Schließer- oder Öffnerventile) sperren eine Rohrleitung völlig ab oder aber öffnen sie zu 100%. Ist der Stellantrieb des Ventils ein pneumatischer Zylinder, dann verwendet man als Bindeglied zwischen SPS und Ventilantrieb gern elektropneumatische Steuerventile. Diese Steuerventile haben einen Magneten, der das Steuerventil stellt. Den Strom dafür liefert die SPS als 24-Volt-Signal.

Die Bezeichnung *5/2-elektropneumatisches Steuerventil* bedeutet 5 Luftanschlüsse am Ventilkörper und 2 Schaltzustände. Ein 3/2-Steuerventil hat demnach nur 3 Luftanschlüsse. Bei einem **doppelt wirkenden Stellzylinder** mit zwei Luftanschlüssen verwendet man gerne das **5/2-Ventil**.

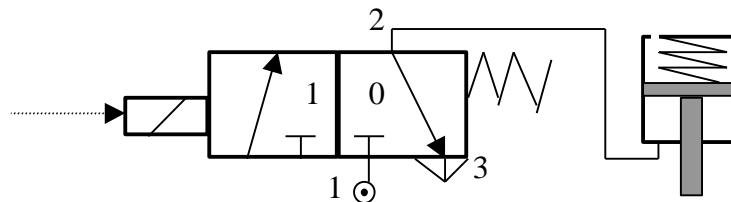


Sm ist der Stoßmagnet, der die Schaltkörper 0 und 1 mittels Rückholfeder Rf horizontal bewegt. Die Luftanschlüsse 1-5 behalten dabei ihre alte Position. Im Schaltzustand 0 ist die Druckluft (Anschluss 1) über Arbeitsleitung 2 mit dem oberen Einlass des Stellzylinders verbunden. Der Kolben geht in Richtung "zu" und schließt das Ventil. Die Luft unter dem Kolben kann über Arbeitsleitung 4 und Entlüftung 5 entweichen.

Kommt ein Signal von der SPS, dann wird Schaltkörper 1 vom Stoßmagneten zwischen die Luftanschlüsse geschoben. Jetzt strömt die Druckluft über Arbeitsleitung 4 in den unteren Teil des Stellzylinders und drückt den Kolben in Richtung "auf". Die Luft oberhalb des Kolbens entweicht über Arbeitsleitung 2 und Entlüftung 3. Die Feder im Stellzylinder sorgt bei Ausfall der Druckluft für eine sichere Ventilstellung (hier wird das Ventil geschlossen).

Endet das SPS-Signal, schiebt die Rückholfeder Rf den Schaltkörper 0 wieder zurück zwischen die Luftanschlüsse, und das Ventil fährt automatisch zu (Nullstellung).

Bei einem **einfach wirkenden Stellzylinder** mit nur einem Luftanschluss und Schließfeder verwendet man gerne das **3/2-Steuerventil**:



In Schaltstellung 0 drückt die Feder im Stellzylinder das Ventil zu. Die Luft unter dem Kolben entweicht über die Arbeitsleitung (2) und Entlüftung (3). In der Schaltstellung 1 geht die Druckluft (1) über die Arbeitsleitung (2) zum Zylinder und drückt ihn gegen die Federkraft nach oben. Die Luft über dem Kolben entweicht durch eine Bohrung.

4.3 Pneumatikplan

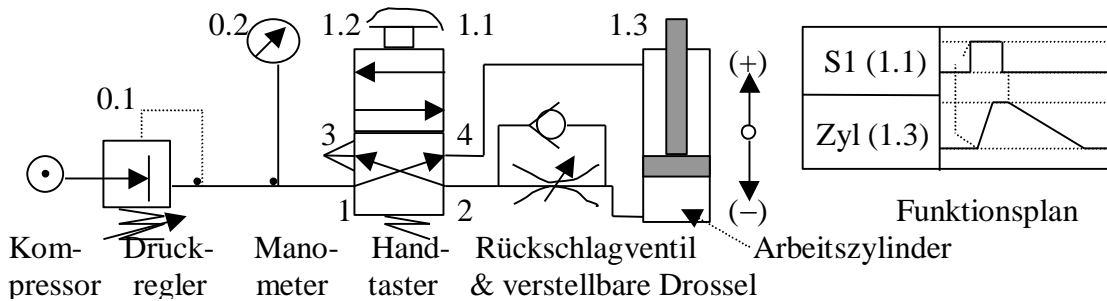
Logische Schaltungen

<p>Zylinder Z1 soll nur ausgefahren werden, wenn Signal 1.1 <u>und</u> mindestens eines der Signale 1.2 und/<u>oder</u> 1.3 und/oder 1.4 anliegt.</p>	
---	--

Der Pneumatikplan enthält die pneumatischen Schaltzeichen (siehe oben), zusätzlich auch Elemente der Luftbereitstellung, -kontrolle und -freigabe (Kompressor, Druckregler, Manometer, Freigabeventil). In der Nähe von Arbeitszylindern werden Funktionsdiagramme angeordnet, die die Dynamik des Ein- und Ausfahrens des Zylinders graphisch darstellen (Steuersignal über der Zeit aufgetragen, Weg über der Zeit aufgetragen).

Beispiel: Bei Betätigung des Handtasters 1.1 am Steuerventil 1.2 fährt Zylinder 1.3 im Eilgang in die Plus-Lage (+). Nach dem Loslassen des Tasters fährt der Zylinder langsam zurück in die Minus-Lage (-). Das langsame Zurückfahren wird durch eine verstellbare Drossel in der Leitung 2 reguliert. Das Rückschlagventil verhindert das schnelle Abströmen der Abluft. Der Funktionsplan

zeigt diese Dynamik. (Oben das Handtastersignal, unten der Weg-Zeit-Plan. Die auslösenden Signalfanken sind mit dem Weg-Zeit-Diagramm durch feine gestrichelte Pfeile verbunden.)



Übliche Steuerungssysteme in der Pneumatik

- Manuelle Steuerung (Hand- und Fußbetätigung von Ventilen)
- Zeitabhängige Folgesteuerung (berücksichtigt auch Wartezeiten, z.B. startet der Endschalter eines Zylinders nach einer Wartezeit von 1 Sekunde den nächsten Vorgang)
- Programmschaltwerksteuerung (Nockenwelle mit konstanter Drehung betätigt Ventile)
- Schrittschaltwerksteuerung (Nockenwelle dreht erst auf ein auslösendes Signal weiter)
- Wegabhängige Folgesteuerung (Tasterrollen laufen auf Schablonen und steuern Ventile)
- Taktkettensteuerung (von Zustand zu Zustand, wird heute kaum noch pneumatisch realisiert)
- Elektropneumatische Steuerung (Eine SPS steuert den Einsatz der pneumatischen Aktoren)

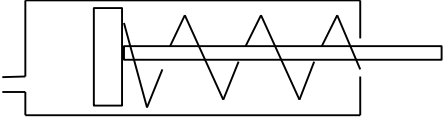
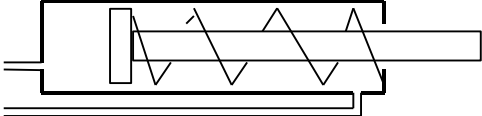
4.4 Pneumatische Sensoren und Aktoren (Auswahl)

Pneumatische Sensoren haben einen Selbstreinigungseffekt bei Holzspänen oder Staub. Sie sind unempfindlich gegen Hitze, elektromagnetische Felder oder Vibrationen.

<p>Luftschränke bis zu einem Abstand $A=50$ mm einsetzbar um z.B. das Beladen einer Hobelmaschine mit einem Brett zu kontrollieren. Der Luftstrom aus der Strahldüse S staut sich im Trichter des Empfängers E. Das auswertbare Drucksignal x wird durch einen Turbulenzverstärker (TV) weiter verstärkt</p>	
<p>Näherungsgeber: Bei großer Entfernung des Körpers K zur Düse ist das Signal x ein Unterdruck. Die aus der Düse strömende Luft erzeugt ihn. Bei Annäherung auf 6-8 mm staut sich die Luft. Signal x schlägt um in Überdruck. Die Messgenauigkeit ist 0,1 mm. Ein pneumatischer Membranverstärker erzeugt dann aus x ein verwertbares 0-1-Drucksignal.</p>	
<p>Ultraschallschränke bis 1,5 m. Die Pfeife links erzeugt Ultraschall. Bei freiem Durchgang verstärkt sich der Schalldruck im Empfänger, einmal durch den Trichter, und zum anderen durch den Turbulenzverstärker. Das Drucksignal x wird weiter in ein 0-1-Signal verstärkt.</p>	

Aktoren : Das Hauptanwendungsgebiet der Pneumatik sind heute die Aktoren. Sie sind robust, langlebig, wartungsarm, leicht zu ersetzen.

Der Zylinder ist das häufigste Arbeitselement der Pneumatik. Die Kolbengeschwindigkeiten liegen zwischen 0,05 bis 2 m/s. Der verwendete Luftdruck liegt zwischen 4 und 8 bar. Die Durchmesser liegen zwischen 8 mm und 350 mm. Große pneumatische Zylinder heben bis zu 8 t. Endlagendämpfer aus Gummi verhindern ein hartes Aufschlagen des Kolbens, erzeugen aber Schwingungen. Die dynamische Kraft eines Zylinders während des Ausfahrens oder Einfahrens beträgt ca 80% der statischen Kraft $F=P \cdot A$ (Druck mal Querschnitt). Dieser Kraftverlust beruht auf dem Druckabfall in den Zuluftleitungen und einer Druckzunahme in den Abluftleitungen während die Luft strömt. Der Wirkungsgrad von Zylindern liegt bei $\eta=0,5$.

Einfach wirkende Zylinder benötigen nur eine Arbeitsleitung, dafür aber eine Rückstellfeder	Doppelt wirkende Zylinder fahren mit Druck ein und aus. Die Feder wirkt bei Druckausfall.
	

Es gibt zahlreiche **Sonderbauarten von Zylindern** (eine Auswahl):

- 3- und 4-Stellungszyylinder statt mit nur zwei Arbeitsstellungen
- Tandemzylinder: (mehrere parallel wirkende Zylinder vereinen ihre Kraft. Der Vorteil ist die geringe Bauhöhe des einzelnen Zylinders)
- Schlagzylinder (Presslufthammer, Niethammer, Schmiedehammer)
- Seilzylinder (Ein flaches, flexibles Metallband wird durch kleine Rollen so geführt, dass es außen am Zylindermantel anliegt. Es ist durch luftdichte Schlitze mit der Bewegung des Kolben im Zylinder verbunden. Man erreicht dadurch eine extrem kurze Baulänge.)
- Drehzylinder (eine am Kolben befestigte Zahnstange erzeugt die Drehung)
- Drehflügelzylinder (kann bis 3/4 Drehung ausführen, z.B. bei Ventilkappen eingesetzt)
- Membranzylinder (werden sehr oft als Stellantrieb bei Ventilen verwendet. Die Membran ist gummiertes Leinen zwischen zwei Stahlplatten, deren Durchmesser etwas kleiner ist, als der des Zylinders.. Die Stahlplatten nehmen den Druck der Feder oder der Federn auf. Die Bewegung beträgt meist nur wenige Millimeter oder Zentimeter)

Druckluftmotore arbeiten wie umgekehrte Kompressoren. Mit Druckluft werden die Kolben bewegt. Die Kurbelwelle setzt diese Bewegung in Drehbewegung um. Der Wirkungsgrad ist etwa $\eta=0,1$. Man baut sie bis zu Leistungen von 100 KW. **Luftdruckturbinen** haben meist eine geringe Leistung (ca. 1 W), erreichen dafür aber 350.000 U/min (Zahnbohrer).

Düsengeräte werden beim Versprühen von Farbe, zum Ausblasen von Spänen und zum Sandstrahlen benutzt.

Der Schwebetisch besteht aus zwei schweren planen Metallplatten, zwischen die durch Düsen in der unteren Platte Luft gepresst wird. Die obere Platte schwimmt auf einem Luftpolster von wenigen Zehntel mm auf und kann Lasten bis zu 300 t tragen. Diese Lasten lassen sich relativ mühelos in eine gewünschte Arbeitsposition verschieben. Stellt man die Luft ab, setzt sich die obere Platte und das schwere Bauteil wird in seiner neuen Position fixiert.

Pneumatikfinger gibt es in unterschiedlichen Ausführungen. Sauger heben z.B. mit Unterdruck Papierbögen von einem Stapel an, um sie einer Druckmaschine zuzuführen. Bestimmte Arten von **Roboterfingern** bestehen z.B. aus luftdichten Nachbildungen von Händen. Diese "Handschuhe" sind neben den Eisenfingern des Roboters zusätzlich mit kleinen Kugeln gefüllt, die einen weichen Griff des Roboters bewirken. Saugt man die Luft aus dem Handschuh, werden die kleinen Kugeln starr aneinander gepresst, und der Handschuh erstarrt. Der Roboter kann jetzt ein empfindliches Elektronikteil heben, ohne dass es aus der "Hand" fällt und ohne dass es zerbricht.

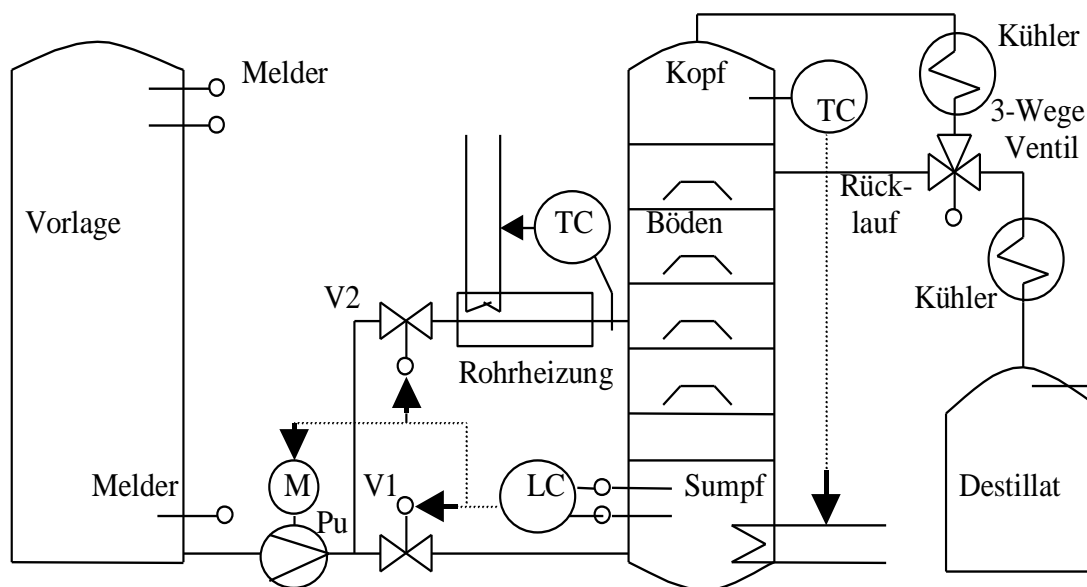
5. Regeln

Regeln ist eine wichtige Teilaufgabe der Automatisierung. Regeln heißt die Regelgröße ständig messen, mit dem Sollwert vergleichen und bei Abweichungen die Stellgröße so ändern, dass die Regelgröße sich wieder dem Sollwert annähert. Wichtige **Regelaufgaben** sind:

1. Konstantregelung: (Der Sollwert bleibt konstant)	Die Regelung soll Störungen von der Regelgröße fernhalten, z.B. eine konstante Temperatur einhalten, oder einen konstanten pH
2. Führungsregelung: (Der Sollwert ändert sich, entweder durch einen Oberregelkreis oder durch sich ändernde Prozessparameter)	Die Regelung hat hier die Aufgabe, den Istwert (die Regelgröße) möglichst schnell und sauber dem Sollwert nachzuführen.

5.1 Kurze Einführung in die Regelung

5.1.1 Beispiel Destille



Für den reinen Destillationsvorgang reicht der Vorlagetank, Pumpe Pu mit Motor M, die Destille mit den Böden, die Sumpfheizung, die beiden Kühler, das 3-Wege-Ventil und der Destillattank. Aber ohne Automatisierung müsste ein Destillenarbeiter ständig die Kopftemperatur mittels der Sumpfheizung nachregeln, die Tankfüllhöhen überwachen, den Sumpfpegel überwachen. Er wäre damit fast pausenlos beschäftigt.

Steuerungsaufgaben sind hier:

- Meldungen anzeigen (Füllstände, Betriebszustand, ...)
- Level-Control im Sumpf (über Pumpe, Ventile V1, V2)
- 3-Wege-Ventil für Vorlauf(Rücklauf schalten (wenn es keine eigene Zeitschaltung hat)
- Betriebszustände ein-/ausschalten (Anfahren, Dauerbetrieb, Abfahren)

Regelungsaufgaben sind hier:

- Temperature Control der Kopftemperatur mittels Sumpfheizung
- Temperature Control des Zulaufs mittels Rohrheizung

5.1.2 RI-Diagramme (Rohrleitungs- und Instrumentierungsfließbild nach DIN 19227)

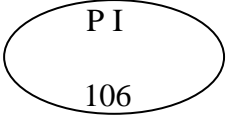

Wir unterscheiden drei Fließbildstufen:

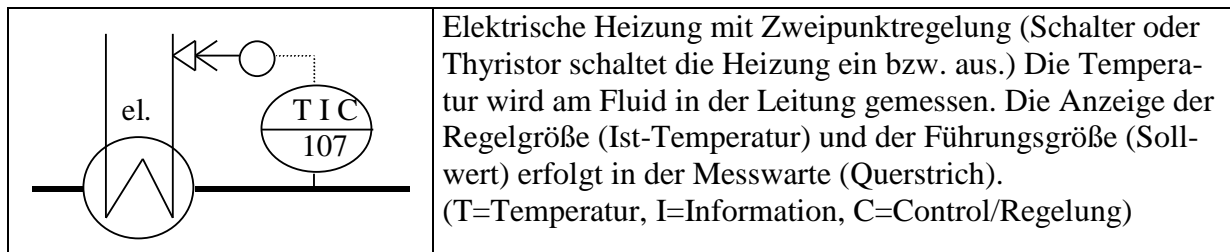
- Das Grundfließbild besteht aus Rechtecken (z.B. "Säuresynthese oder "Fermentation", die ganze Verfahren darstellen. Die Rechtecke sind durch Massen- und Energieströme verknüpft mit Wertangaben an den Verbindungslinien, z.B. "Prozessdampf 50 t/h" oder "elektrische Heizung 400 KW".
- Das Verfahrensließbild enthält symbolhaft Apparate und Armaturen in der Grundform, Rohrleitungen und wichtige MSR-Aufgaben. Das Fließbild zur "Destille" in der Einführung ist ein Beispiel.
- Das RI-Fließbild enthält Apparate in speziellen Ausführungen (z.B. unterschiedliche Symbole für Rohrbündel- oder Plattenwärmetauscher), alle MSR-Aufgaben, Angaben zu Material, Isolierung, Nennweiten usw.

MSR-Stellenkreis (Messen-Steuern-Regeln Kreis)

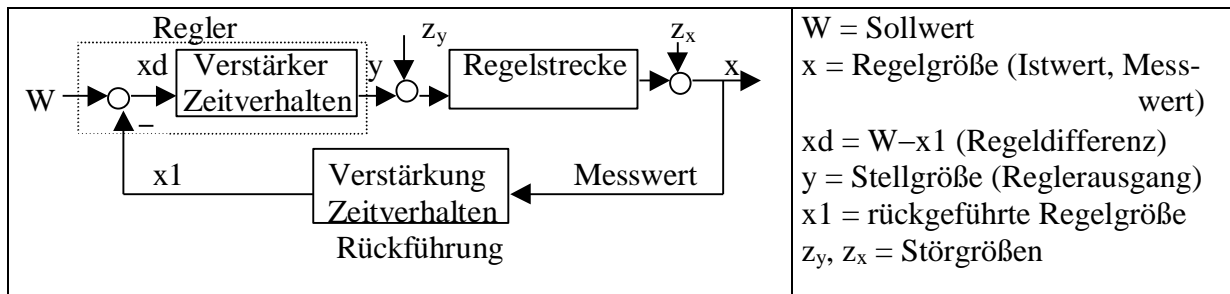
Die MSR-Stellenkreise vertreten symbolisch die Mess-, Steuer- und Regelaufgaben. Man bringt sie am Ort der Messung an. Bei Steuerung und Regelung (Kennbuchstabe C) zieht man eine gestrichelte Linie zum Stellort, z.B. zum Ventil. Einige Messvariable und ihre Kennbuchstaben: **T**emperature, **P**ressure, **F**low, **L**evel, **E**lectricity, **D**ensity, **W**eight, **M**oisture, **V**iscosity.

Einige Aktionen und ihre Kennbuchstaben: **C**ontrol, **R**egistration, **I**nformation, **A**larm.

	Druckanzeige örtlich (P=Pressure/Druck, I= Information, kein Querstrich zwischen PI und Messstellenummer bedeutet örtliche Anzeige, z.B. einfach ein Druckmesser am Rohr)
	Durchfluss (F), Registration (R) und Control (C). Control kann Steuerung und/oder Regelung heißen. Das Signal wird zentral in der Messwerte verarbeitet. Die Messstellenkennzeichnung darf auch Buchstaben enthalten



5.1.3 Die Abbildung zeigt das **Blockschaltbild** eines **Einfachregelkreises**.



Die Regelstrecke muss immer aus der Sicht der Regelgröße gesehen werden, denn ein und derselbe Rührkessel kann sich in Hinsicht auf die Temperaturregelung ganz anders verhalten (z.B. viel langsamer auf die Heizung reagieren) als bezüglich des O₂-Gehalts (indem die O₂-Konzentration z.B. schnell auf eine Rührerdrehzahländerung reagiert).

Eine Temperaturregelstrecke ist z.B. das Temperaturverhalten eines Rührkessels als Antwort auf eine Hubänderung am Dampfventil der Heizung. Eine pH-Regelstrecke ist z.B. das Verhalten des pH-Wertes im selben Rührkessel als Antwort auf das Einschalten einer Dosierpumpe, die Natronlauge dosiert. Ein und derselbe Apparat kann demnach mehrere Regelstrecken mit unterschiedlichen Eigenschaften aufweisen.

Als **erweiterte Regelstrecke** bezeichnet man den gesamten Regelkreis mit Ausnahme des Reglers, d.h. die Regelstrecke, den Aktor mit Stellantrieb, und den Messvorgang mit Messumformer (Rückführung).

Regler können analog arbeiten (elektronische Schaltung) oder digital (μ-Prozessor). Ein Regler besteht aus den 4 Teilen: Vergleich, Verstärker, Integration, Differenziation

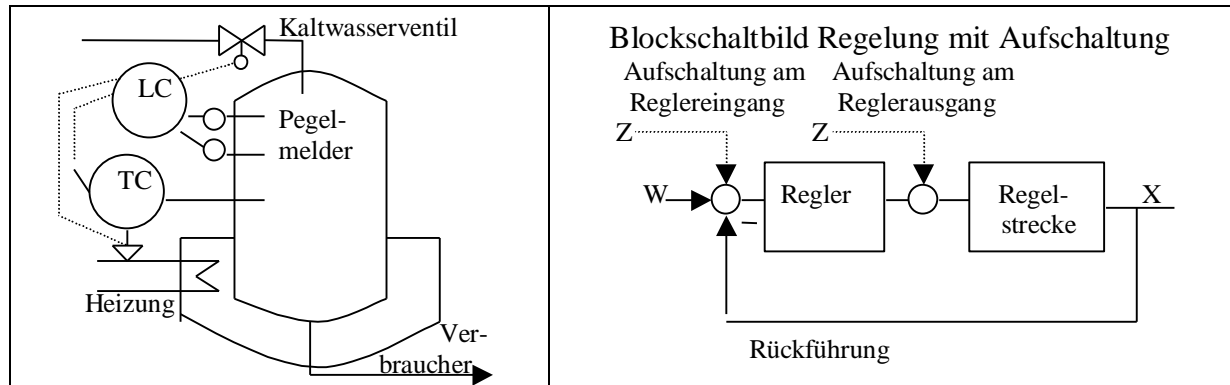
Vergleicher (Additions- bzw. Subtraktionsglied)	Bildet die Regeldifferenz x_d aus Sollwert minus Istwert x_1	$x_d(t) = W(t) - x_1(t)$
Verstärker	Bildet P-Anteil der Stellgröße	$y_p(t) = K_p \cdot x_d(t)$
Integrierendes Zeitverhalten: Das Integral über die Regeldifferenz (Fläche unter der Kurve) multipliziert mit einer Konstanten K_I ist der I-Anteil		$y_I(t) = K_I \cdot \int_0^t x_d(u) du$ <p>Eine anwachsende Fläche deutet auf länger anhaltende Abweichungen x_d hin</p>

5.1.5 Regelschaltungen

Von den vielen möglichen Regelschaltungen sind die folgenden 4 für uns interessant:

- Der Einfachregelkreis (siehe Kap. 5.1.3)
- Der Regelkreis mit Aufschaltung eines zweiten Messwertes (Hauptstörgröße)
- Die Kaskadenregelung
- Die Mehrgrößenregelung

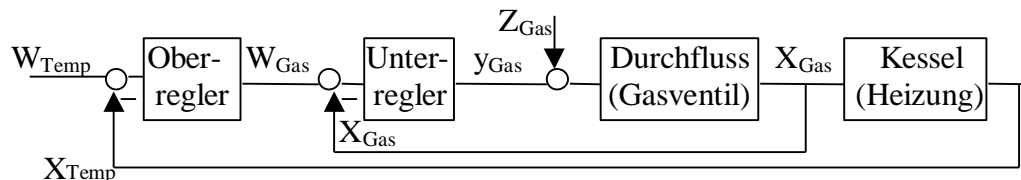
Regelkreis mit Aufschaltung: Eine messbare Hauptstörgröße z liefert ein Signal, das den



Aktor (z.B. das Ventil der Heizung) sofort ansteuert. Der Regler übernimmt dann nur noch das Nachregulieren. Der Vorteil ist der Zeitgewinn. Der einsetzenden Störung wird sofort begegnet, noch bevor ein normaler Regler reagieren könnte. Das Beispiel zeigt einen Warmwasserspeicher für $W=85^\circ\text{C}$ Solltemperatur. Wenn die Niveauregelung LC das Kaltwasserventil öffnet, wird das Ventilsignal (20 mA) gleichzeitig an die Temperaturregelung gegeben.

- Aufschaltung am Reglereingang wirkt wie eine Sollwertänderung von $W=85^\circ\text{C}$ auf das Ende des Messbereichs (z.B. auf 120°C).
- Aufschaltung am Reglerausgang wirkt wie das Signal "100 % Heizen".

Kaskadenschaltung: Ein Oberregelkreis liefert einem oder mehreren Unterregelkreisen den Sollwert. Auch mehr als 2 Stufen sind möglich. *Wichtig: Der Oberregelkreis muss langsam regeln, der unterste Regelkreis am schnellsten, sonst entstehen gefährliche Schwingungen.* Das Beispiel zeigt eine Temperaturregelung, bei der der Oberregelkreis den Sollgasdurchfluss W_{Gas} vorgibt, den der Unterregelkreis gegen Störungen aus dem Gasnetz halten soll. Solche Störungen können z.B. Druckschwankungen im Netz sein.



Es bedeuten: W_{Temp} , X_{Temp} sind Soll- und Ist-Temperatur
 W_{Gas} , X_{Gas} sind Soll- und Ist-Gasdurchfluss für den Brenner
 Z_{Gas} ist eine Störung im Gasnetz, z.B. eine Druckschwankung

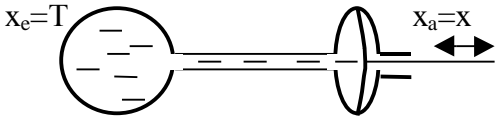
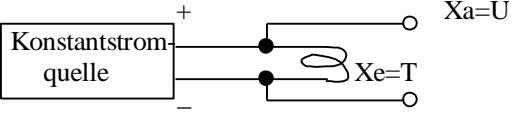
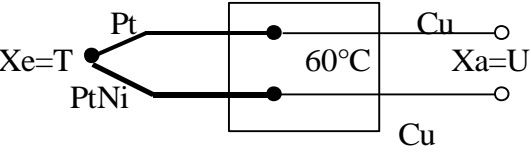
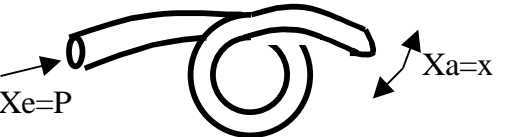
Mehrgrößenregelung: Stören sich Regelkreise massiv, z.B. eine Temperaturerhöhung reduziert massiv die O_2 -Konzentration im Rührkessel bzw. eine Erhöhung der Rührerdrehzahl trägt massiv

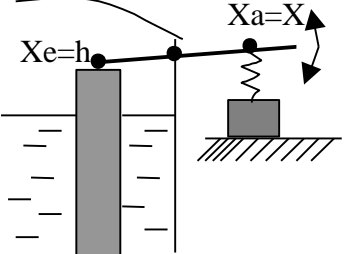
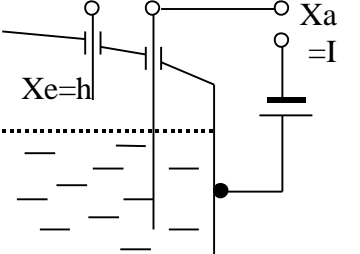
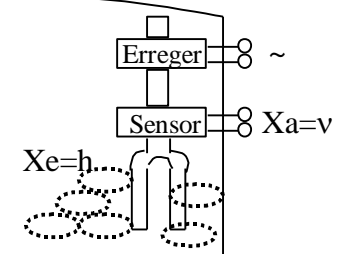
Wärme ein, dann spricht man von *gekoppelten Regelkreisen* und benötigt eine Mehrgrößenregelung. Es gibt viele Lösungen, von denen zwei genannt werden sollen:

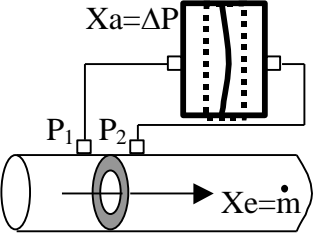
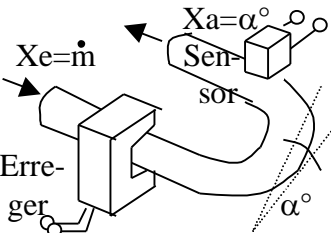
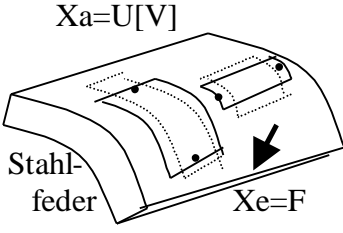
- Entkopplung der Regelkreise über Zusatzregler
- Fuzzy Control (Auswählen und Ausführen von Aktionen, die in einer Wissensbasis von Experten zusammengestellt wurden)

5.1.6 Bauteile

Messwerterfassung: Sensoren wandeln die physikalische Messgröße in ein elektrisches Signal. Die folgende Tabelle zeigt einige Sensoren, die in der Biotechnologie und Verfahrenstechnik eingesetzt werden:

	<p>Das <i>Flüssigkeitsthermometer</i> besteht aus einer gefüllten temperaturfesten Kugel. Der Inhalt dehnt sich bei Erwärmung und verbiegt eine Membran. Die Auslenkung erzeugt das Signal</p>
	<p><i>PT100</i> (Platinthermometer mit Widerstand 100 Ω) wird von konstantem Strom (meist 1 mA) durchflossen. Der Widerstand ist temperaturabhängig und damit auch Spannung U.</p>
	<p><i>Thermoelement</i>: Zwei unterschiedliche Metalllegierungen erzeugen an der Lötstelle eine temperaturabhängige Thermospannung U. Die Lötstellen zum Kupferdraht müssen konstante Temperatur haben (40, 50, 60°C üblich)</p>
	<p><i>Bourdonrohr (Rohrfeder)</i> zum Messen sehr hoher Drücke. Das Rohr aus massivem Stahl "streckt" sich bei Druckbeaufschlagung. Die Auslenkung X ist dem Druck proportional.</p>

<p>Pegelsensor: Ein Auftriebskörper (Keramik) ändert mit der Füllhöhe sein Gewicht und die Federauslenkung</p>	<p>Pegelmelder: Drahtsonden tauchen in die (leitende) Flüssigkeit. Stromkontakt meldet den Pegelstand der Sonde</p>	<p>Pegelmelder für lockeres Schüttgut: Die Dämpfung D der Stimmgabel und damit die Frequenz ν hängt von h ab</p>
		

<p>Durchflusssensor Lochblende: Massendurchfluss \dot{m} hängt funktional vom Differenzdruck ΔP vor und nach der Lochblende im Dampfrohr ab</p>	<p>Coriolissensor: Massendurchfluss \dot{m} hängt proportional vom Winkel α° der Schwingungsebene des schwingenden Rohrbogens ab</p>	<p>Kraftsensor: Kraft F ist proportional der Spannung U, die aus den Widerstandsänderungen der Dehnungsmessstreifen resultiert (Wheatstone-Brücke)</p>
		

Messumformer sind elektronische Bauteile, zumeist direkt am Sensor angeordnet und zumeist mit einem μ -Prozessor bestückt. Messumformer haben folgende Aufgaben:

- Verstärkung des Messsignals (Sensoren liefern oft Spannungen im μV -Bereich.) Die Verstärkung legt auch die Ausdehnung des Messbereichs 0 bis 100% (z.B. 20-80°C) fest.
- Nullpunkteinstellung, z.B. 0% --> 20°C
- Digitalisierung des analogen Messsignals mittels AD-Wandler
- Linearisierung, wenn die Sensorkennlinie nichtlinear ist. Die Umrechnung wird über eine Tabelle von Eichpunkten mittels linearer oder nichtlinearer Interpolation durchgeführt.
- Filterung (entfernen von Spannungsspitzen), Glätten (Reduktion von Rauschen)
- Betrieb der Datenleitung zur Leitwarte (Bustreiber für den Feldbus)
- Alarmkontakte schalten (Überschreitung, Unterschreitung von gesetzten Grenzwerten)
- Galvanische Trennung von Prozess und Leitwarte (Schutz vor Überspannungen)
- Anzeige vor Ort mittels Display oder Zeigerinstrument
- Stromversorgung des Sensors

Trennverstärker sind die billige Variante zum Messumformer: Sie verstärken das Messsignal und machen eine galvanische Trennung. Man hat meist 10 oder 20 nebeneinander.

Ein Bus ist eine Datenleitung (analog oder digital) mit *Bustreibern* an beiden Enden, die sich über *Protokolle* (festgelegte Signalfolgen) miteinander verständigen (Siehe Kapitel 3).

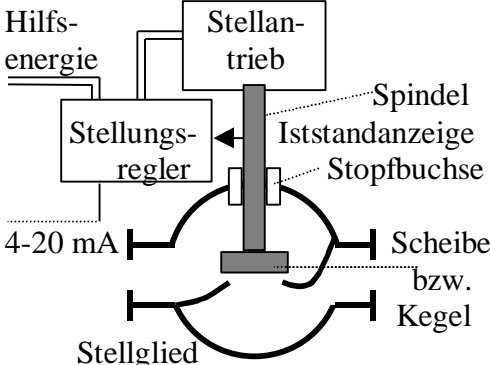
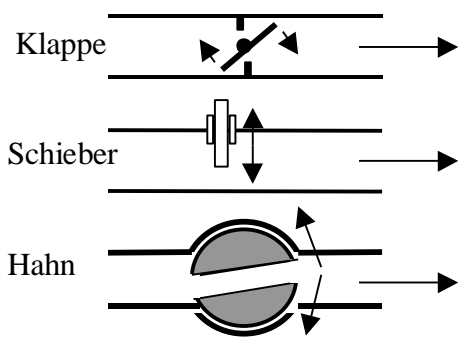
Aktoren beeinflussen Massen- und Energieströme in einem Prozess. Es gibt eine Vielzahl von Aktoren: Ventile Klappen, Schieber, Hähne, Motore, Pumpen, Verdichter, Arbeitszylinder usw.

Ventile: Wir unterscheiden Absperrventile (stoppen Durchfluss), Regelventile (machen Durchfluss regelbar), Sicherheitsventile (öffnen bei Überdruck), Rückschlagventile (schließen bei Unterdruck), Reduzierventile (an Gasflaschen), Nadelventile (für kleine und kleinste Durchflüsse), Membranventile (völlig dicht nach außen, sterilisierbar).

Klappen verwendet man alternativ zu Ventilen bei nicht zu hohen Drücken.

Schieber verwendet man bei großen Rohrdurchmessern

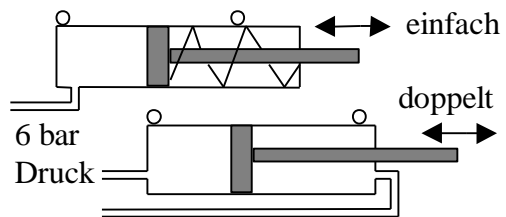
Hähne verwendet man alternativ zu bestimmten Ventilformen, z.B. zu Absperrventilen

<p>Regelventil: Hilfsenergie für den Stellantrieb ist 380 V Drehstrom oder Druckluft 6-8 bar. Das Stellsignal hat 4-20 mA</p>	<p>Klappe wird oft statt Ventil verwendet Schieber bei großen Rohrdurchmessern Hähne bei Glasapparaturen, als Absperrhähne</p>
	

Motore dienen zum Antrieb von Apparaten, Förderbändern, Pumpen, Verdichtern usw.

- Drehstrom- und Gleichstrommotore arbeiten asynchron zur Netzfrequenz von 50 Hz
- 1-Phasen-Wechselstrommotore mit Hilfsphase arbeiten meist synchron zur Netzfrequenz
- Schrittmotoren arbeiten synchron zu den eingespeisten Stromimpulsen der Steuerung
- Gasmotore arbeiten wie Benzin-/Dieselmotore und liefern Wärme und mech. Energie
- Druckluftmotore arbeiten wie ein umgekehrter Kompressor (geringer Wirkungsgrad)

Zylinder werden als Linearantriebe benutzt z.B. für Ventile, Klappen, Pressen, zum Festklemmen von Werkstücken, zum Nieten, Hämmern usw.

<p>Ein Drehstrommotor ist belastbar und robust. Die 3 Spulen des Stators erzeugen ein magnetisches Drehfeld, das den Anker mitnimmt</p>	<p>Einfachzylinder benötigt Rückstellfeder und 1 Luftleitung. Der doppelt wirkende Zylinder benötigt 2 Luftleitungen</p>
<p>Zylinder werden als Linearantriebe benutzt z.B. für Ventile, Klappen, Pressen, zum Festklemmen von Werkstücken, zum Nieten, Hämmern usw. Einfachzylinder benötigt Rückstellfeder und 1 Luftleitung. Der doppelt wirkende Zylinder benötigt 2 Luftleitungen.</p>	

Pumpen fördern Flüssigkeiten bzw. Gase durch Rohrleitungen. Bei Flüssigkeiten spielt die Viskosität eine Rolle bzw. Verschmutzungen oder Beimengungen (Schlämme, Sedimente, Sand), die die Ventile beeinträchtigen können. Fördermenge und erzeugter Pumpendruck sind weitere Auswahlkriterien. **Verdichter** fördern und komprimieren Luft bzw. Gase. Radialverdichter arbeiten wie die Kreiselpumpe, Axialverdichter wie ein Propeller (mehrstufig als Turbine mit Schaufeln), Kolbenverdichter wie eine Kolbenpumpe, Membranverdichter wie die Membranpumpe, Schraubenverdichter wie eine Zahnradpumpe.

<p>Bei der Kreiselpumpe schleudern die Schaufeln das Wasser nach außen, wo es austritt. keine Ventile große Fördermenge niedriger bis mittlerer Druck kleine Viskosität erforderlich</p>	<p>Die Kolbenpumpe hat ein Ein- und ein Auslassventil. Der Kolben wird von einem Motor hin und her bewegt kleine - mittlere Fördermenge mittlerer - hoher Druck hohe Viskosität möglich</p>	<p>Die Membranpumpe arbeitet wie eine Kolbenpumpe, ist aber absolut dicht und sterilisierbar. Es sind nur kleine bis mittlere Drücke möglich, kleine bis mittlere Fördermengen, hohe Viskosität.</p>
<p>In der Zahnradpumpe nehmen die Zähne außen das Fluid mit nach unten. In der Mitte kann es nicht zurück. hoher Druck kleine Fördermenge hohe Viskosität möglich</p>	<p>Die Schlauchpumpe schiebt mit Quetschrollen das Fluid im Schlauch weiter. kleine Fördermenge niedriger Druck steril hohe Viskosität möglich</p>	<p>Die Dampfstrahlpumpe arbeitet als Vakuumpumpe. Der Dampfstrahl reißt das Gas mit und es entsteht ein Unterdruck im Gasrohr, der weiteres Gas nachzieht, bis ein Vakuum entstanden ist.</p>

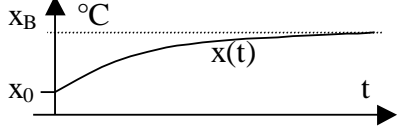
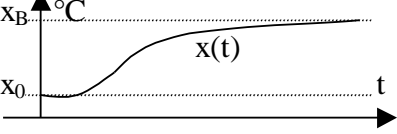

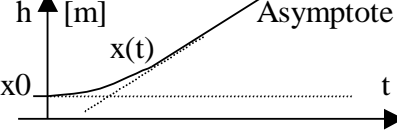
Verdichter fördern und komprimieren Luft bzw. Gase. Radialverdichter arbeiten wie die Kreiselpumpe, Axialverdichter wie ein Propeller (mehrstufig als Turbine mit Schaufeln), Kolbenverdichter wie eine Kolbenpumpe, Membranverdichter wie die Membranpumpe, Schraubenverdichter wie eine Zahnradpumpe.

5.2 Regelstrecken

Regelstrecken sind Prozesse, bei denen Prozessparameter (z.B. Temperatur, Druck, pH, Trübung, Konzentration, Durchfluss usw.) geregelt werden. Wichtig ist die Reaktion des zu regelnden Prozessparameters auf seine Stellgröße, z.B. wie stark und wie schnell reagiert die Temperatur auf eine definierte Hubänderung am Heizventil

5.2.1 Klassifikation der Regelstrecken: Regelstrecken werden nach der **Sprungantwort** klassifiziert. Eine Sprungantwort ist der Verlauf der Regelgröße x nach einer sprungartigen Erhöhung der Stellgröße y um einen definierten Betrag $\Delta y = U_0$.

	<p>Die Graphik zeigt einen Stellgrößensprung, z.B. das Einschalten einer Zusatzheizung. Die Größe U_0 muss keine Spannung sein, sondern ist die Änderung einer beliebigen Stellgröße (Heizleistung, Rührerdrehzahl, Stromstärke, ..)</p>
--	--

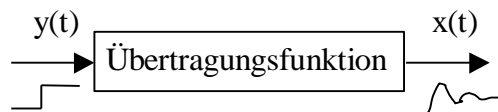
	<p>PT1-Sprungantwort (Ein-Speicher-Modell): Die Regelgröße steigt sofort nach dem Stellgrößensprung vom alten Beharrungswert x_0 und nähert sich asymptotisch als e-Kurve dem neuen Beharrungswert x_B. Beispiel: Temperatur im Rührkessel nach Heizsprung</p>
	<p>PTn-Sprungantwort (n-Speicher-Modell): Die Regelgröße steigt erst nach einer Verzugszeit vom alten Beharrungswert x_0 zum Wendepunkt und von da asymptotisch zum neuen Beharrungswert x_B. Beispiel: Temperatur am Duschkopf einer Dusche.</p>
	<p>I-Sprungantwort (einfache integrierende Strecke): Die Regelgröße steigt sofort vom alten Stand x_0 in Form einer Geraden ohne definierten Endpunkt. Beispiel: Pegel in einem Tank nach Start der Pumpe</p>
	<p>ITn-Sprungantwort (integrierende Strecke mit n Verzögerungsgliedern): Die Kurve $x(t)$ startet waagrecht und geht dann asymptotisch in eine Gerade (Asymptote) über. Beispiel: Pegelstand in einem Wasserbehälter mit vorge-schaltetem Sandfilter</p>

PT1- und PTn-Strecken heißen Strecken **mit Ausgleich**, da sie einem neuen Beharrungswert zustreben, wenn sich die Stellgröße ändert (engl. selfregulating members). I- und ITn-Strecken sind Strecken **ohne Ausgleich**, da sie keinem bestimmten Wert zustreben.

5.2.2 Übertragungsglieder der Regelungstechnik

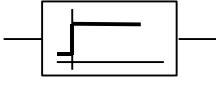
Regelungstechnik ist Signalverarbeitung. Diese wird gern als **Blockschaltbild** dargestellt. Das Blockschaltbild besteht aus **Boxen**, **Linien** und **Additionspunkten**. Boxen sind Symbole für Übertragungsglieder. Übertragungsglieder stehen für Verhaltensmodelle realer Prozesse. Der reale Prozess wird durch ein **mathematisches Modell** approximiert.


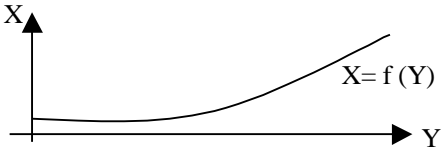
Ein Übertragungsglied hat einen Signaleingang $y(t)$ und einen Signalausgang $x(t)$. Der Quotient $x(t)/y(t)$ heißt **Übertragungsfunktion**. Ist $y(t)$ die normierte Sprungfunktion $1(t)$, dann ist $x(t)$ identisch mit der Übertragungsfunktion (da Division durch 1 nichts verändert).

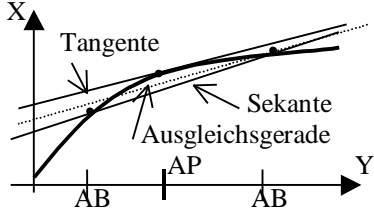


Man unterscheidet die beiden Klassen **statische** und **dynamische** Übertragungsglieder:

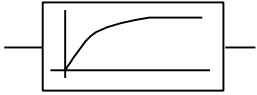
- *Statische Glieder* haben eine funktionelle Abhängigkeit $x=f(y)$, in der die Zeit nicht auftritt, d.h. der Ausgang x hängt nur über eine Kennlinie vom Eingang y ab, unabhängig von der Zeit t .
- *Dynamische Glieder* haben einen Zeitverlauf $x(t)$ am Ausgang, d.h., dass wir das Verhalten durch Differentialgleichungen beschreiben müssen mit zeitabhängigen Lösungen $x(t)$. Aus historischen Gründen versucht man, möglichst Differentialgleichungen mit konstanten Koeffizienten zu verwenden, z.B. $a_2 \ddot{x}(t) + a_1 \dot{x}(t) + a_0 x(t) = b_0 y(t)$. Jedes Modell (beschrieben durch seine DGL) hat seine charakteristische Sprungantwort, d.h. eine charakteristische Kurvenform der Lösungskurve.

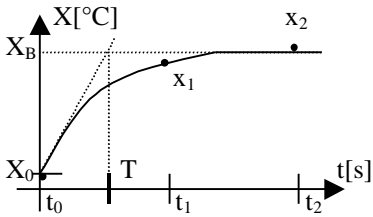
Das P- oder Proportionalglied modelliert statische Glieder mit Kennlinie, wie z.B. Ventile, Pumpen, Verstärker usw. Das Boxensymbol zeigt die Sprungantwort, die wieder eine Sprungfunktion ist. Das Modell hat nur einen Parameter, die Verstärkung K_P .	
---	---

Kennlinien können Gerade durch den Ursprung sein, wie z.B. $x = K_P \cdot y$ oder beliebige Gerade, wie z.B. $x = x_0 + K_P \cdot y$. Die Übertragungsfunktion ist dann $F(p)=K_P$.	
In den meisten Fällen sind Kennlinien jedoch nichtlineare Funktionen, bei denen das Ausgangssignal x über einen funktionalen Zusammenhang $x=f(y)$ vom Eingangssignal y abhängt.	

Aus Gründen, die mehr historisch sind und der Vereinfachung der mathematischen Behandlung komplexer Schaltungen dienen, werden nichtlineare Kennlinien gern linearisiert. Drei übliche Methoden sind: 1. Tangente im Arbeitspunkt AP 2. Sekante über dem Arbeitsbereich AB 3. Ausgleichsgerade über dem Arbeitsbereich AB	
--	--

In allen drei Fällen der Linearisierung entsteht die allgemeine Gerade $x = x_0 + K_P y$ mit der sehr einfachen Übertragungsfunktion $F(p)=K_P$.

Das PT1-Glied (auch 1-Speicher-Modell genannt) modelliert Prozesse wie zum Beispiel das Aufheizen oder Abkühlen eines Rührkessels. Das Symbol rechts zeigt die Sprungantwort - eine e-Kurve, die gegen einen Beharrungswert strebt. Das Modell hat 2 Parameter, K_P und T .	
--	---

Die Sprungantwort auf einen Eingangssprung der Höhe $\Delta y=U_0$ (z.B. das Einschalten einer Heizung mit 200 W) zeigt die Graphik rechts. Die Temperatur steigt anfänglich stark. Durch die mit steigender Temperatur proportional steigende Wärmeabgabe verlangsamt sich der Anstieg und endet, wenn Wärmeeintrag und Wärmeabtrag im Gleichgewicht sind.	
---	--

T = Zeitkonstante x_0 = alter Beharrungswert x_B = neuer Beharrungswert
 t_i = Zeitpunkte x_i = Messwerte K_P = Streckenverstärkung

Große Zeitkonstante T bedeutet einen großen Speicher (braucht lange zum Hochheizen bzw. zum Abkühlen). Eine Faustformel sagt: *nach 5 Zeitkonstanten ist der neue Beharrungswert erreicht (zu 99%)*. Die *Streckenverstärkung* K_P gibt an, wie stark die Strecke auf eine Eingangsänderung U_0 reagiert:

$$K_p = \frac{x_B - x_0}{U_0} \left[\frac{\dim x}{\dim y} \right] \quad \text{z.B.} \quad \left[\frac{K}{W} \right]$$

Die Zeitkonstante T kann man graphisch bestimmen:

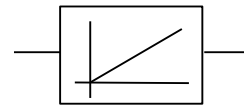
Zur graphischen Bestimmung zeichnet man im Punkt x_0 die Tangente an die Antwortkurve (siehe Graphik oben). Der Zeitwert des Schnittpunkts der Tangente mit der x_B -Linie ist die Zeitkonstante T.

Die **Identifikation einer Strecke** erfordert zwei Anstrengungen:

1. Man muss das zutreffende Modell (den Streckentyp) erkennen, z.B., ob eine PT₁- oder eine PTn- oder eine I-Strecke oder sonst ein Typ vorliegt. Dazu vergleicht man die Sprungantwort der Strecke mit den Sprungantworten verschiedener Modelle und sucht sich das passende Modell für die Strecke heraus.
2. Man bestimmt die Modellparameter des ausgewählten Modells, z.B. die zwei Parameter K_p und T im Falle des PT₁-Modells.

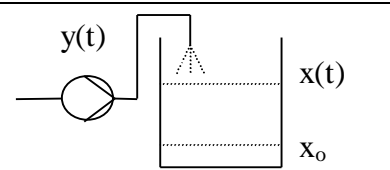
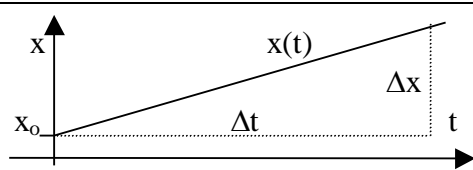
- Die Kenntnis des Streckentyps ist wichtig für die Reglerauswahl.
- Die Kenntnis der Modellparameter ist wichtig für Simulationsrechnungen.

Das **I-Glied** (Integrator) nimmt man z.B. zum Modellieren von Tankfüllständen in Abhängigkeit von Zufluss und Abfluss, aber auch bei Positionen in Abhängigkeit von der Drehzahl eines Positioniersystems, bei Salzkonzentrationen in Abhängigkeit von der Dosierpumpe. Das Symbol zeigt die Sprungantwort des Integrators, eine Gerade.



Das Modell hat nur einen Parameter, die Integrierkonstante K_I .

Sprungantwort $x(t)$ auf einen Leistungssprung $\Delta y = U_0$ z.B. einer Pumpe.



Wird die Pumpe eingeschaltet, dann steigt der Pegel x im Behälter kontinuierlich an (vorausgesetzt ein konstanter Querschnitt des Behälters). Die Identifikation erfordert nur die Berechnung der Integrierkonstanten K_I .

Integrierkonstante:

$$K_I = \frac{\Delta x}{U_0 \Delta t} \left[\frac{\dim x}{\dim y \cdot \text{Zeit}} \right] \quad (\text{siehe Graphik})$$

Zusammengesetzte Übertragungsglieder

Mit P-, PT₁-, DT₁- und I-Gliedern lassen sich komplexere Übertragungsverhalten modellieren. (Das DT₁-Modell wird hier nicht behandelt. Es tritt fast nur als Bestandteil von Reglern auf). Die 3 wichtigsten Schaltungen sind:

Reihenschaltung (am Beispiel von 2 Gliedern)	
Parallelschaltung (am Beispiel von 2 Gliedern)	
Kreisschaltung: Wellengenerator hat das "+" Regelkreis hat das "-" F ₂ heißt Rückführung	

Das PTn-Glied oder n-Speicher-Modell

Man modelliert damit größere Verzögerungen, z.B. die Temperatur am Duschkopf, wenn das heiße Wasser den weiten Weg aus einer Heizanlage im Keller nimmt, oder eine Kraftwerkskesselheizung mit einem vorgeschalteten Förderband (hier gilt eigentlich das sehr ähnliche **PT₁T_t-Modell**, das aber mathematisch schwerer zu behandeln ist, als das PTn-Modell).

Das PTn-Modell ist eine Reihenschaltung von n PT ₁ -Gliedern. Man nimmt meist die gleiche Zeitkonstante T für alle Glieder. Die Gesamtverstärkung K_{PS} ist das Produkt aller Einzel-K _P .	
---	--

Das PTn-Modell hat 3 Parameter: **K_{PS}**, **n**, **T** (Verstärkung, Speicherzahl, Zeitkonstante eines einzelnen Speichers). Die zwei konkurrierenden Modelle PTn und PT₁T_t unterscheiden sich nur minimal in der Sprungantwort (**T_t** heißt hier Totzeit):

Eingangssprung Δy	Sprungantwort PTn-Modell	Sprungantwort PT ₁ T _t -Modell

Auf Grund der Ähnlichkeit der beiden Modelle behandelt man das PT₁T_t-Modell wie ein PTn-Modell, wobei die **Totzeit T_t** gleich **Verzugszeit T_u** gesetzt wird, und die Zeitkonstante T₁ des einzigen Speichers des PT₁T_t-Modells gleich der **Ausgleichszeit T_g** des PTn-Modells gesetzt wird, d.h. **T_t=T_u** und **T₁=T_g**

Die Identifikation des PTn-Modells erfordert die Bestimmung von K_{PS}, n und T. Die Verzugszeit **T_u** bestimmt man aus dem Schnittpunkt der Wendepunkt tangente mit der Geraden x₀. Die Zeitsumme Tu+T_g bestimmt man aus dem Schnittpunkt der Wendepunkt tangente mit der Geraden x_B. Dann ist die Ausgleichszeit **T_g = (Tu+T_g) – Tu**.

Gesamtverstärkung der Strecke

$$K_{PS} = \frac{x_B - x_0}{U_0} \left[\frac{\dim x}{\dim y} \right]$$

Schwierigkeitsgrad der Strecke
(ab $S=0,3$ heißt eine Strecke *schwierig*)

$$S = T_u / T_g$$

Speicherzahl des Modells
(auf nächstliegende ganze Zahl runden)

$$n = [10 \cdot S + 1]_{\text{runden}}$$

Für die Bestimmung der Zeitkonstantensumme ΣT_i benutzt man die Tatsache, dass alle Sprungantworten der Modelle PT_1 , PT_2 , PT_3 , ..., PT_n im Zeitpunkt $t = \Sigma T_i$ fast identische Werte haben. Der Mittelwert der Höhen liegt bei 57,5% des Abstands $x_B - x_0$. Deshalb berechnet man die Höhe x_{57} und findet so den Zeitpunkt $t = \Sigma T_i$.

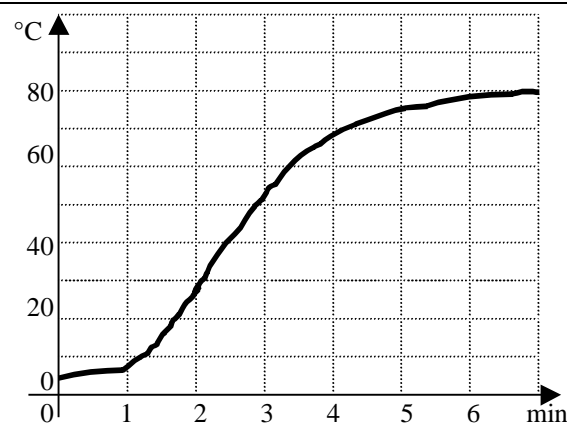
$$x_{57} = x_0 + (x_B - x_0) \cdot 0,575$$

Den Wert $t = \Sigma T_i$ findet man am Schnittpunkt der Geraden x_{57} mit der Kurve $x(t)$.

Zahlenbeispiel

Identifizieren Sie die rechts stehende Antwortkurve auf einen Stellgrößensprung der Höhe $U_0=5$ KW Heizleistung (Typ und alle Parameter der Strecke):

1. Der Typ ist PT_n
2. Wir zeichnen Wendepunkt WP ein
3. Wir schätzen den Beharrungswert x_B
4. Wir berechnen Gesamtverstärkung K_{PS}
5. Wir zeichnen die Tangente und bestimmen T_u , T_u+T_g und damit T_g



6. Wir berechnen Schwierigkeitsgrad S und daraus Speicherzahl n des Modells
7. Wir berechnen die Hilfsgröße x_{57} und bestimmen damit die Zeitkonstantensumme ΣT_i .
8. Wir bestimmen T nach der Formel $T = \Sigma T_i / n$.

$x_B = 80,8^\circ\text{C}$ aus der Graphik

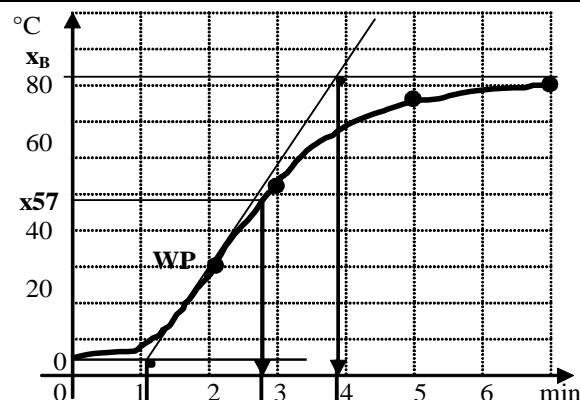
$$K_{PS} = \frac{x_B - x_0}{U_0} = \frac{80,8 - 5}{5} = 15,16 \left[\frac{^\circ\text{K}}{\text{KW}} \right]$$

$T_u = 1,1$ [min] aus der Graphik

$T_u + T_g = 3,9$ [min] aus der Graphik

$$T_g = (T_u + T_g) - T_u = 3,9 - 1,1 = 2,8 \text{ [min]}$$

$$S = T_u / T_g = 1,1 / 2,8 = 0,39$$



$$n = [10 \cdot S + 1]_{\text{runden}} = [4,9]_{\text{runden}} = 5 \quad (\text{unser Streckenmodell sollte 5 Speicher haben})$$

$$x_{57} = x_0 + (x_B - x_0) \cdot 0,575 = 5 + (80,8 - 5) \cdot 0,575 = 48,6 \text{ [}^\circ\text{C]}$$

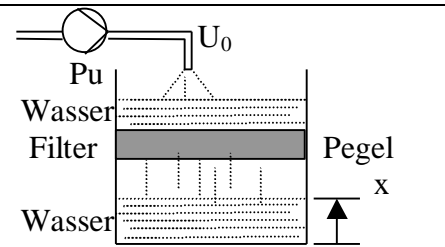
Wir finden mit dem Schnittpunkt x_{57} und Kurve $x(t)$ den Wert $\Sigma T_i = 2,8 \text{ [min]}$ für die Zeitkonstantensumme. Daraus berechnen wir die Zeitkonstante eines einzelnen Speichers:

$$T = \Sigma T_i / n = 2,8 / 5 = 0,56 \text{ [min]}.$$

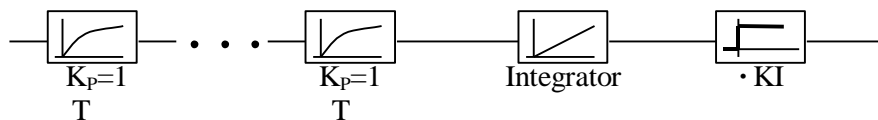
Das ITn-Glied oder n-Speicher-Modell mit Integrator

Das Modell besteht aus einer Reihenschaltung eines PTn-Gliedes mit einem Integrator (I-Glied). Das PTn-Glied ist verantwortlich für eine Verzögerung des Prozesses. Das Modell hat 3 Parameter: Integrierkonstante K_I , Speicherzahl n , Zeitkonstante T eines einzelnen Speichers

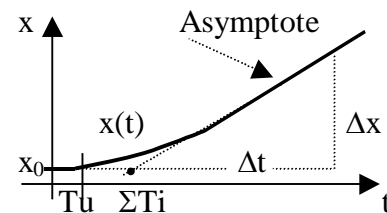
Ein Beispiel für ein ITn-Glied ist eine Pumpe mit Sandfilter und Wassertank. Beginnt die Pumpe zu arbeiten, dann kommt die ersten Sekunden kein Wasser aus dem Filter. Dann tröpfelt es und erst nach einer gewissen Zeit, wenn sich über dem Sandfilter genug Wasser angesammelt hat, das Druck erzeugt, dann ist Zufluss gleich Abfluss im Sandfilter, und der Pegel x steigt linear.



Das Blockschaltbild (Symbol) zeigt die Reihenschaltung. (Manche Programmierer von Simulationsaufgaben lassen die letzte Box mit der Multiplikation mit K_I weg, und setzen dafür eine der Verstärkungen K_P der n Verzögerungsglieder gleich K_I , zumeist das letzte K_P .)



Die **Sprungantwort** des ITn-Gliedes auf einen Eingangssprung der Höhe $\Delta y = U_0$ zeigt die Graphik rechts. Nach der anfänglichen Verzugszeit T_u schwenkt die Kurve $x(t)$ in die Asymptote ein. Der Schnittpunkt der Asymptote mit der Geraden x_0 liefert die Zeitkonstantensumme ΣT_i . Der Anstieg $\Delta x / \Delta t$ der Asymptoten geht in die Berechnung der Integrierkonstanten K_I ein.



Identifikation des ITn-Gliedes heißt Bestimmung der 3 Parameter K_I , n , T

Integrierkonstante

$$K_I = \frac{\Delta x}{U_0 \cdot \Delta t} \left[\frac{\dim x}{\dim y \cdot \text{Zeit}} \right]$$

Verzugszeit T_u und Zeitkonstantensumme ΣT_i werden aus der Graphik abgelesen (siehe oben). Wir berechnen die Hilfsgröße

$$H = T_u / \Sigma T_i$$

In Abhängigkeit von Hilfsgröße H bestimmen wir die Ausgleichszeit T_g

Ausgleichszeit ist
$$T_g = \begin{cases} \Sigma T_i, & \text{wenn } H < 0,43 \\ (\Sigma T_i - T_u) / 0,57, & \text{wenn } H \geq 0,43 \end{cases}$$

Schwierigkeitsgrad $S = T_u / T_g$

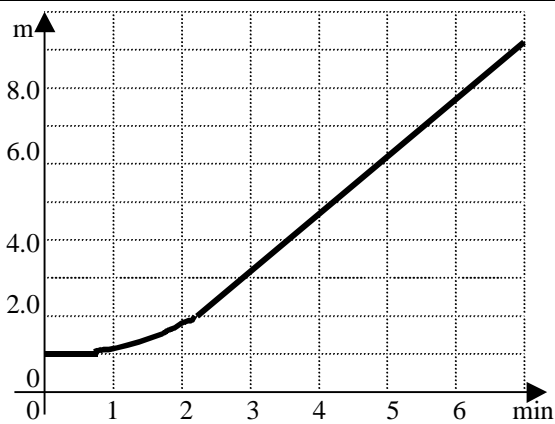
Speicherzahl $n = [10 \cdot S + 1]_{\text{runden}}$

Zeitkonstante eines Speichers $T = \Sigma T_i / n$

Zahlenbeispiel:

Identifizieren Sie die rechts stehende Antwortkurve auf einen Stellgrößensprung der Höhe $U_0=4$ KW Pumpenleistung (Typ und alle Parameter).

1. Steigungsdreieck Δx , Δt zeichnen und Δx und Δt bestimmen
2. Integrierkonstante K_I berechnen
3. Verzugszeit T_u ist die Zeit für das horizontale Startstück der $x(t)$ -Kurve
4. Zeitkonstantensumme ΣT_i wird durch den Schnittpunkt der Asymptoten mit der Geraden x_0 gefunden.



5. Hilfsgröße H, Ausgleichszeit T_g berechnen
6. Schwierigkeitsgrad S, Speicherzahl n, und Zeitkonstante T eines Speichers berechnen.

$\Delta x = 8$ [m], $\Delta t = 5,5$ [min] (aus Graphik)

$$K_I = \frac{\Delta x}{U_0 \cdot \Delta t} = \frac{8}{4 \cdot 5,5} = 0,364 \left[\frac{m}{KW \cdot min} \right]$$

$T_u = 0,7$ [min] (aus Graphik)

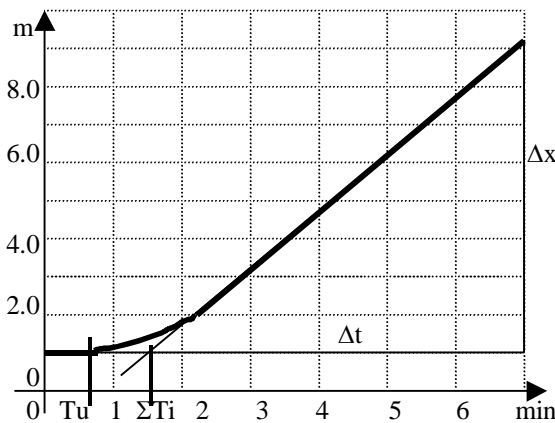
$\Sigma T_i = 1,5$ [min] (aus Graphik)

$$H = T_u / \Sigma T_i = 0,7 / 1,5 = 0,47$$

da $H > 0,43$ ist, berechnen wir

$$T_g = (\Sigma T_i - T_u) / 0,57 = (1,5 - 0,7) / 0,57$$

$$T_g = 1,4$$
 [min]



$$S = T_u / T_g = 0,7 / 1,4 = 0,5$$

$$n = [10 \cdot S + 1]_{\text{runden}} = [10 \cdot 0,5 + 1]_{\text{runden}} = 6$$

$$T = \Sigma T_i / n = 1,5 / 6 = 0,25$$
 [min]

5.3 Kontinuierliche Regler

Kontinuierliche Regler arbeiten entweder analog oder digital. Die Taktzeit τ eines digitalen kontinuierlichen Reglers muss klein sein gegenüber der Zeitkonstantensumme ΣT_i der Regelstrecke ($\tau \leq 0,1 \cdot \Sigma T_i$). Allen hier behandelten Reglern gemeinsam ist die Reglerverstärkung K_{PR} . Sie ist dimensionsbehaftet. Im Zähler steht die Dimension der Stellgröße y , im Nenner die der Regelgröße x . Beide Größen können mit absoluten Dimensionen auftreten ($\text{dim } y$, $\text{dim } x$) oder mit relativen ($\% x$, $\% y$). Die 4 möglichen Dimensionspaarungen sind:

$$K_{PR} \left[\frac{\text{dim } y}{\text{dim } x} \right] \quad \text{oder} \quad K_{PR} \left[\frac{\% y}{\% x} \right] \quad \text{oder} \quad K_{PR} \left[\frac{\text{dim } y}{\% x} \right] \quad \text{oder} \quad K_{PR} \left[\frac{\% y}{\text{dim } x} \right].$$

Prozentzahlen bei der Stellgröße beziehen sich auf den Stellbereich des Aktors, bei der Regelgröße auf den eingestellten Messbereich des Messumformers. Die Dimension der Reglerverstärkung K_{PR} ist immer invers zur Streckenverstärkung K_{PS} . Das Produkt heißt **dimensionslose Kreisverstärkung** V_0 .

$$V_0 = K_{PR} \cdot K_{PS}$$

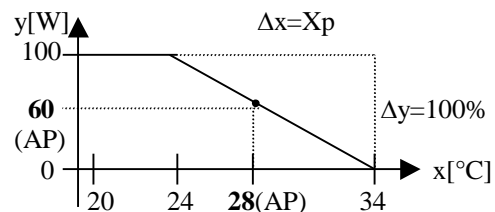
5.3.1 P-Regler

Der P-Regler ist ein einfacher Verstärker. Als Analogregler ist er ein Operationsverstärker oder ein Transistor. Digital ist er eine einfache Multiplikation. Der P-Regler ist billig, robust, aber in manchen Anwendungen ungenau. Die Stellgröße y berechnet sich nach der Geradengleichung:

$$y = Y_0 + K_{PR} \cdot x_d = Y_0 + K_{PR} (W - x) = Y_0 + \Delta y$$

y	Stellgröße am Reglerausgang, die den Aktor ansteuert, z.B. ein Ventil
Y_0	Arbeitspunkt der Stellgröße. Der P-Regler addiert bzw. subtrahiert lediglich ein proportional von x_d abhängiges Δy zum bzw. vom Arbeitspunkt Y_0
$x_d = W - x$	Regeldifferenz zwischen Soll- und Istwert der Regelgröße
W	Sollwert, der von der Regelgröße eingehalten werden soll
x	gemessener Istwert der Regelgröße
Δy	Änderung der Stellgröße, Abweichung vom Arbeitspunkt
X_p	Proportionalbereich: Der Abschnitt der x -Achse, in dem eine Änderung Δx der Regelgröße x zu einer proportionalen Änderung Δy der Stellgröße y führt
AP	Arbeitspunkt des P-Reglers mit den Werten $W_0 = X_0$ und Y_0 . Man sucht die Stellgröße Y_0 , die bei ungestörtem Betrieb die Regelgröße x genau auf dem Arbeitspunktsollwert W_0 hält ($W_0 = X_0$).

Beispiel: Eine Aquarium-Heizung von $Y_0 = 60$ Watt hält gerade die Temperatur x auf der gewünschten Solltemperatur von $W_0 = X_0 = 28^\circ\text{C}$ (vorausgesetzt ein normal temperiertes Wohnzimmer). Unter 24°C Wassertemperatur läuft die Heizung maximal, über 34°C schaltet sie ab. Der Bereich $24\text{--}34^\circ\text{C}$ ist der Proportionalbereich X_p .



P-Regler arbeiten unterschiedlich gut mit unterschiedlichen Typen von Regelstrecken zusammen. **Die Kombination P-Regler mit I-Strecken ist ein gutes Gespann.** Die Kombination P-Regler mit P-Strecken ist nur zu empfehlen, wenn der P-Regler aus Preisgründen genommen werden muss (Billigregler)

Wir stellen uns einen P-Regler im Arbeitspunkt vor ($W=W_0$ und $y=Y_0$). Liegen keine Störungen vor, dann stellt sich $x=X_0=W_0$ ein, d.h. der Regler arbeitet genau. Es gilt $xd=W-x=0$ im Arbeitspunkt bei fehlenden Störungen. Bei einer Störung z (angegeben in der Dimension von y) oder einer gewünschten Sollwertänderung ΔW (Abweichung vom Arbeitspunktsollwert W_0) ergeben sich folgende **ständige Regelabweichungen xd des P-Reglers**:

P-Regler mit I-Strecke bzw. ITn-Strecke	$xd = z / K_{PR}$	Abweichungen treten nur bei Störung z auf
P-Regler mit PTn- oder PT ₁ Tt-Strecke	$xd = \frac{1}{1+V_0} \Delta W + \frac{K_{PS}}{1+V_0} z$	Abweichungen bei Störung z und bei Sollwertänderung ΔW

5.3.2 PID-Regler (Proportional + Integrierend + Differenzierend)

Der PID-Regler ist der Standardregler der Industrie, aus dem sich leicht auch PI- oder PD-Regler konfigurieren lassen, indem man einen Part weglässt, d.h. ausschaltet. Die Stellgröße setzt sich aus 3 Anteilen zusammen, die addiert werden - der P-, der I- und der D-Anteil. Man unterscheidet zwei Typen von PID-Reglern, den *idealen PID-Regler* und den *realen PID-Regler* (auch PIDT₁-Regler genannt).

Der reale PID-Regler baut den differenzierenden Anteil der Stellgröße y e-kurvenförmig ab, gesteuert durch die Zeitkonstante T_1 .

T_1	Reglerparameter, Zeitkonstante des D-Anteils, der sich mit e^{-t/T_1} abbaut
y	berechnete Stellgröße, d.h. das Ausgangssignal des Reglers
K_{PR}	Verstärkung, Reglerparameter, steuert den P-Anteil, aber auch I- und D-Anteil
T_N	Nachstellzeit , Reglerparameter, steuert den I-Anteil,
T_V	Vorhaltezeit , Reglerparameter, steuert den D-Anteil,
$xd=W-x$	Regeldifferenz zwischen Soll- und Istwert
t	die ablaufende Zeit

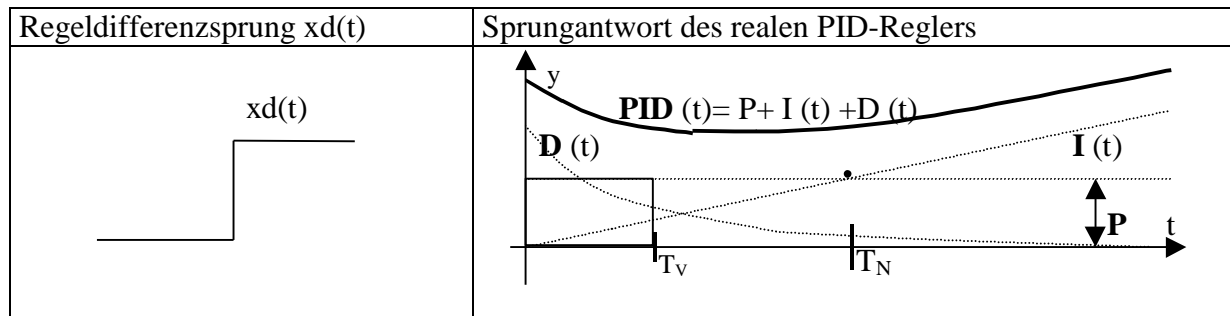
(Bei manchen Reglerherstellern wird nicht T_1 als Reglerparameter des realen PID-Reglers benutzt, sondern die **Vorhalteverstärkung** $VV = T_V / T_1$.)

Der DIN-Algorithmus **des idealen PID-Reglers** ist eine in der Regelung häufig benutzte digitale Berechnungsformel für die Stellgröße y . Der neue Wert der Stellgröße, y_{i+1} , berechnet sich aus dem alten Wert y_i durch Addition eines Inkrements Δy :

$$y_{i+1} = y_i + K_{PR} \left[(xd_{i+1} - xd_i) + \left(\frac{T_A}{T_N} xd_i \right) + \frac{T_V}{T_A} (xd_{i+1} - 2xd_i + xd_{i-1}) \right]$$

i ist eine fiktive Taktzeitnummerierung $i = 0, 1, 2, 3, \dots, \infty$

T_A Taktzeit, d.h. die Zeit von einer Messung bis zur nächsten Messung der Regelgröße



Der I-Anteil benötigt die **Nachstellzeit** T_N , um auf denselben Wert anzuwachsen, den der P-Anteil sofort und konstant liefert (siehe Graphik). Die *Wirkung* des D-Anteils insgesamt von $t=0$ bis $t \rightarrow \infty$ ist gleich der Wirkung des P-Anteils über die **Vorhaltezeit** T_V . Unter Wirkung (action) versteht man hier das Integral. Die Fläche unter der $D(t)$ -Kurve ist gleich der Fläche des Rechtecks mit Höhe P und Breite T_V (siehe links unten in der Graphik).

P-Anteil des PID-Reglers	$P = K_{PR} \cdot x_d$	konstant, wenn x_d konstant
I-Anteil des PID-Reglers	$I(t) = \frac{K_{PR} \cdot x_d}{T_N} \cdot t$	Eine Gerade, wenn x_d konstant ist
D-Anteil des realen PID-Reglers	$D(t) = \frac{K_{PR} \cdot T_V}{T_1} e^{-t/T_1}$	DT ₁ -Kurve: zuerst ein Sprung und dann ein exponentielles Absinken

5.3.3 Reglerauswahl und Reglereinstellung

Reglerauswahl: Strecken mit I-Anteil bzw. ohne I-Anteil verlangen unterschiedliche Regler (**I** heißt sich mit **I**). Bei rauen Messwerten der Regelgröße (z.B. Rauschen oder häufige kleine Störungen) verzichtet man besser auf den differenzierenden Anteil. Nimm deshalb

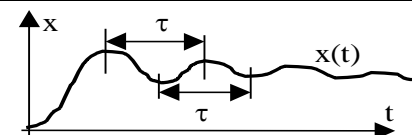
- den PD-Regler für I- oder IT_n-Strecken bei glattem Messwertverlauf
- den P-Regler für I- oder IT_n-Strecken bei rauem Messwertverlauf
- den PID-Regler für P-, PT_n- oder PT₁T_t-Strecken bei glattem Messwertverlauf
- den PI-Regler für P-, PT_n- oder PT₁T_t-Strecken bei rauem Messwertverlauf

Reglereinstellung nach Ziegler-Nichols

Bei P-, PT_n-, PT₁T_t- und IT_n-Strecken, d.h. bei so ziemlich allen Regelstrecken, ist dieses Verfahren anwendbar. Man setzt einen P-Regler in den Regelkreis ein. Dann gibt man eine Störung auf, um den Regelkreis in Schwingung zu versetzen. Die Schwingung sollte um den Arbeitspunkt oder im Arbeitsbereich erfolgen. Eine Störung kann sein:

- eine plötzliche sprungartige Sollwertänderung
- oder eine von außen angreifende sprungartige Störung, z.B. das Öffnen eines Ventils.

Man erhöht die Reglerverstärkung K_{PR} des eingebauten P-Reglers, bis die Störantwort $x(t)$ 3 bis 4 deutliche Schwingungen zeigt. Die Verstärkung K_{PR} , bei der das eintritt, wird kritische Verstärkung $K_{PR_{kr}}$ genannt, die Periode τ_{kr} .



Ziegler-Nichols geben folgende Tabelle zur Berechnung der Reglerparameter an:

Reglerart	K_{PR}	T_N	T_V	für den Streckentyp
P	$K_{PR\ kr} / 2,0$	∞	0	I- oder ITn-Strecken
PI	$K_{PR\ kr} / 2,2$	$\tau_{kr} \cdot 0,85$	0	P-, PTn-, PT ₁ Tt-Strecken
PID	$K_{PR\ kr} / 1,7$	$\tau_{kr} \cdot 0,50$	$\tau_{kr} \cdot 0,125$	P-, PTn-, PT ₁ Tt-Strecken
PD	$K_{PR\ kr} \cdot 0,788$	∞	$\tau_{kr} \cdot 0,068$	I- oder ITn-Strecken

Beispiel: $K_{PR\ kr} = 11,8$ [W/K], $\tau_{kr} = 27$ [s]. Wir wollen einen PID-Regler konfigurieren.

$$K_{PR} = K_{PR\ kr} / 1,7 = 11,8 / 1,7 = 6,94 \text{ [W/K]}$$

$$T_N = \tau_{kr} \cdot 0,50 = 27 \cdot 0,5 = 13,5 \text{ [s]}$$

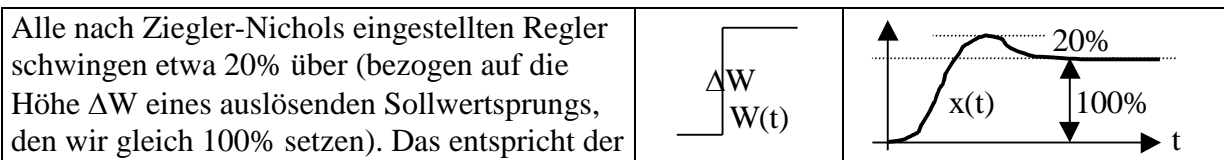
$$T_V = \tau_{kr} \cdot 0,125 = 27 \cdot 0,125 = 3,3 \text{ [s]}$$

Wir wollen einen PD-Regler konfigurieren.

$$K_{PR} = K_{PR\ kr} \cdot 0,788 = 11,8 \cdot 0,788 = 9,30 \text{ [W/K]}$$

$$T_N = \infty \quad (\text{Unendlich ist bei Industrieregeln meist } T_N = 10000 \text{ Sekunden})$$

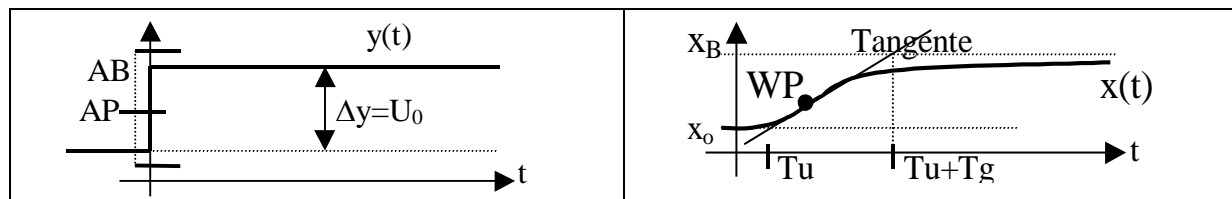
$$T_V = \tau_{kr} \cdot 0,068 = 27 \cdot 0,068 = 1,8 \text{ [s]}$$



gerade noch zulässigen dimensionslosen Dämpfung $D=0,45$ (kleinere Dämpfungen sind nicht erwünscht). Durch Verkleinern des K_{PR} des Reglers (z.B. auf die Hälfte) kann die Höhe des Überschwingens stark vermindert werden, aber auf Kosten der Dynamik. Der Regler arbeitet jetzt etwas langsamer.

Reglereinstellung nach Chien, Hrones und Reswick

Diese Art der Reglereinstellung funktioniert nur bei Strecken mit Ausgleich (P-, PTn-, PT₁Tt-Strecken), und zusätzlich muss der Ausgleich im Arbeitsbereich bzw. in der Nähe des Arbeitspunktes erfolgen. Man nimmt die Regelstrecke aus dem Regelkreis heraus, d.h. man betreibt sie ohne Regler. Man gibt einen Stellgrößensprung $\Delta y = U_0$ auf, der den Arbeitspunkt AP bzw. Arbeitsbereich AB abdeckt, und nimmt die Sprungantwort als Messkurve $x(t)$ auf.



U_0 ist der Stellgrößensprung, z.B. Erhöhung der Heizleistung um $U_0=500$ [W]

x_0 ist der alte Beharrungswert, z.B. 43 [°C] bei der alten Heizleistung

x_B ist der neue Beharrungswert, z.B. 84 [°C] bei der neuen Heizleistung

WP	ist der Wendepunkt der $x(t)$ -Kurve mit der eingezeichneten Tangente
T_u	ist die Verzugszeit, z.B. 11 [s] am Schnittpunkt Tangente und x_0
$T_u + T_g$	ist die Zeit am Schnittpunkt Tangente und x_B
T_g	ist die Ausgleichszeit, z.B. 89 [s]. Es gilt $T_g = (T_u + T_g) - T_u$

$$K_{PS} = \frac{x_B - x_0}{U_0} = \frac{84 - 43}{500} = 0,082 \left[\frac{K}{W} \right] \quad \text{ist die Streckenverstärkung}$$

$$V_0 = K_{PS} \cdot K_{PR} = 0,082 \left[\frac{K}{W} \right] \cdot K_{PR} \left[\frac{W}{K} \right] \quad \text{ist die dimensionslose Kreisverstärkung}$$

Chien, Hrones und Reswick geben eine Tabelle vor, in der für einen P-, PI- oder PID-Regler Werte für die Größen " $V_0 T_u / T_g$ ", " T_N " und " T_V " vorgegeben werden. " T_N " und " T_V " sind bereits Reglerparameter. Den dritten Reglerparameter, die Reglerverstärkung K_{PR} , berechnet man dann aus dem Tabellenwert von " $V_0 T_u / T_g$ ".

Weiterhin unterscheiden die Autoren die Regleranwendung und die Überschwingweite:

- **Führungsregelung** heißt, dass hauptsächlich einem sich ändernden Sollwert nachgefahren werden soll
- **Konstantregelung** heißt, dass bei vorwiegend konstantem Sollwert Störungen ausgeregelt werden sollen.
- **0% Überschwingen** bedeutet einen langsamen Regler ohne Überschwingen (hohe Dämpfung von etwa $D=0,8$)
- **20% Überschwingen** bedeutet einen schnellen Regler mit etwa 20% Überschwingen, bezogen auf die Höhe einer sprungartigen Sollwertänderung (niedrige Dämpfung von etwa $D=0,45$)

Tabelle von Chien, Hrones und Reswick zur Reglereinstellung

Regler		0% Überschwingen		20% Überschwingen	
Typ	Parameter	Konstantreg.	Führungsreg.	Konstantreg.	Führungsreg.
P	$V_0 T_u / T_g$	0,3	0,3	0,7	0,7
PI	$V_0 T_u / T_g$	0,6	0,35	0,7	0,6
	T_N	$4 \cdot T_u$	$1,2 \cdot T_g$	$2,3 \cdot T_u$	$1,0 \cdot T_g$
PID	$V_0 T_u / T_g$	0,95	0,6	1,2	0,95
	T_N	$2,4 \cdot T_u$	$1,0 \cdot T_g$	$2,0 \cdot T_u$	$1,35 \cdot T_g$
	T_V	$0,42 \cdot T_u$	$0,5 \cdot T_u$	$0,42 \cdot T_u$	$0,47 \cdot T_u$

Beispiel: Wir wollen einen PID-Regler für Führungsregelung und ohne Überschwingen konfigurieren (parametrieren). Die Werte für K_{PS} , T_g und T_u siehe weiter oben:

$$K_{PR} = \frac{(V_0 T_u / T_g) \cdot T_g}{K_{PS} \cdot T_u} = \frac{0,6 \cdot 89}{0,082 \cdot 11} = 59,2 \left[\frac{W}{K} \right] \quad \text{Reglerverstärkung (Wert 0,6 aus Tabelle)}$$

$$T_N = 1,0 \cdot T_g = 89 \text{ [s]} \quad \text{Nachstellzeit (Wert 1,0 aus Tabelle)}$$

$$T_V = 0,5 \cdot T_u = 0,5 \cdot 11 = 5,5 \text{ [s]} \quad \text{Vorhaltezeit (Wert 0,5 aus Tabelle)}$$

Sonderfall: Eine PT_1 -ähnliche Regelstrecke hat entweder eine Verzugszeit $T_u=0$ oder zumindest eine sehr kleine Verzugszeit T_u . Das würde nach obiger Formel für K_{PR} zu einer unsinnig hohen Reglerverstärkung führen, die die Regelung instabil macht. Es ist deshalb angebracht, die Reglerverstärkung nach oben zu begrenzen.

Ist $T_u < 0,1 \cdot T_g$, dann liegt dieser Sonderfall vor, und wir berechnen K_{PR} nach der rechts stehenden Formel	$K_{PR} = \frac{(V_0 T_u / T_g) \cdot 10}{K_{PS}}$
---	--

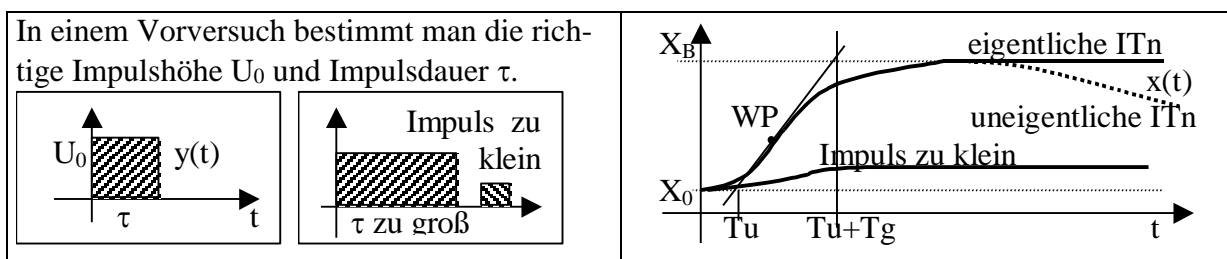
Beispiel zum Sonderfall: $T_u = 6$ [s], $T_g = 89$ [s], $K_{PS} = 0,082$ [K/W]. Wir wollen den gleichen PID-Regler wie oben parametrieren:

$$K_{PR} = \frac{0,6 \cdot 10}{0,082} = 73,7 \left[\frac{W}{K} \right], \quad T_N = 1,0 \cdot T_g = 89 \text{ [s]}, \quad T_V = 0,5 \cdot T_u = 0,5 \cdot 6 = 3 \text{ [s]}.$$

Reglereinstellung nach Impulsantwort

I-Strecken regelt man bei glattem Messwertverlauf am besten mit einem PD-Regler. Die Impulsantwort einer echten IT_n -Strecke ähnelt der Sprungantwort einer PT_n -Strecke. Es gibt auch unechte IT_n -Strecken (eigentlich PT_n -Strecken mit zu stark wirkenden Aktoren, so dass sich im Messbereich kein Beharrungswert einstellen kann). Auch diese regelt man wie echte IT_n -Strecken.

Das Aufnehmen der Impulsantwort erfolgt bei abgekoppelter Regelstrecke. Die Impulsantwort sollte im Arbeitsbereich liegen bzw. der Arbeitspunkt etwa auf halber Höhe der Impulsantwort. Man gibt einen Stellgrößenimpuls auf und registriert die Antwortkurve der Regelstrecke, z.B. schaltet man 1,5 Sekunden lang ($\tau = 1,5$ s) eine Pumpe mit der Leistung 250 W ein ($U_0 = 250$ W) und misst die Pegelerhöhung im Tank. Die Impulsdauer darf maximal $1/20$ der Zeitkonstantensumme betragen. Eine gute Näherung ist $\tau \leq (T_u + T_g)/20$. Da man die Summe aus Verzugszeit und Ausgleichszeit $T_u + T_g$ erst kennen muss, bedarf es einiger Vorversuche, bis man die richtige Impulshöhe und Impulsdauer gefunden hat.



Hat man den richtigen Impuls ermittelt, dann registriert man die Impulsantwort und bestimmt aus der Graphik x_0 , x_B , T_u , T_g . Dabei ist:

U_0	die Impulshöhe, z.B. 250 W Pumpenleistung
t	die Impulsdauer, z.B. 1,5 Sekunden
$U_0 \cdot t$	die Impulsfläche (in der Graphik oben links schraffiert), im Beispiel 375 Ws
x_0, x_B	alter und neuer Beharrungswert (bzw. Maximum bei einer uneigentlichen IT_n), z.B. $x_0 = 15,0$ cm, $x_B = 18,3$ cm

Tu Verzugszeit, z.B. 2,6 Sekunden (Schnittpunkt Wendepunkt tangente mit x_0)
 Tg Ausgleichszeit, z.B. 32,5 s (aus Schnittpunkt Wendepunkt tangente mit x_B gewinnt man $Tu+Tg$, dann minus Tu liefert Tg)

Berechne $K_I = \frac{x_B - x_0}{U_0} = \frac{18,3 - 15,0}{375} = 0,0088 \left[\frac{cm}{Ws} \right]$ Integrierkonstante

Die Tabelle von Hall und v. Weber liefert für PD-Regler die beiden Reglerparameter K_{PR} und T_V einmal ohne Überspringen (0%) und einmal mit 20% Überspringen.

	0 % Überspringen	20 % Überspringen
K_{PR}	$\frac{1,17 \left(1 - (Tu/Tg)^2 0,347 \right)}{K_I (Tu + Tg)}$	$\frac{1,3 + (Tg/Tu^*)^2 0,006}{K_I (Tu + Tg)}$
T_V	$(Tu + Tg)(1 - 0,132 (Tu/Tg))(0,4 + (Tg/Tu^*)^2 0,00096)$	$(Tu + Tg) 0,228$

$$Tu^* = \begin{cases} Tu, & \text{falls } Tu \geq 0,05 Tg \\ 0,05 Tg & \text{sonst} \end{cases} \quad \text{ist für den Sonderfall eines zu kleinen } Tu\text{-Wertes zu nehmen}$$

Beispiel: Ein PD-Regler mit 0% Überspringen soll für die obigen Beispielzahlen parametrisiert werden.

$$K_{PR} = \frac{1,17 (1 - (2,6/32,5)^2 0,347)}{0,0088 \cdot 35,1} = 3,78 \left[\frac{W}{cm} \right] \quad \text{Reglerverstärkung}$$

$$T_V = 35,1 (1 - 0,132 (2,6/32,5))(0,4 + (32,5/2,6)^2 0,00096) = 19,08 [s] \quad \text{Vorhaltezeit}$$

Dieser PD-Regler kann jetzt den Pegelstand im Tank auf seinem Sollwert halten

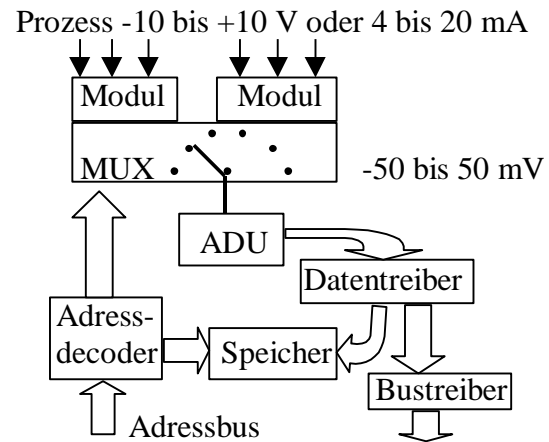
5.4 Regeln mit der SPS (Beispielhaft für die S5-Serie aus dem AT-Praktikum)

Wie man mit einer SPS steuert, wurde im Kapitel 2.2 behandelt. **Steuern** heißt, binäre Signale auszuwerten und mit den Mitteln der Logik und anderen binären Schaltelementen (Flipflop, Timer, Zähler) Prozessschritte zu starten und zu beenden, und innerhalb der Prozessschritte die richtigen Sensoren zu befragen und die richtigen Aktoren zu stellen. **Regeln** dagegen heißt, eine Regelgröße auf einem Sollwert zu halten trotz Störungen oder Sollwertänderungen.

5.4.1 Lesen und Wandeln von Analogsignalen allgemein

Analogsignale sind z.B. Stromsignale im Bereich 4-20 mA (Industrienorm) oder Spannungssignale im Bereich -10 bis +10 V (Industrienorm). Beim Einlesen in den Rechner werden die Signale in Zahlen umgewandelt.

Vom Prozess kommen analoge Signale. Ein **Modul** wandelt sie in Spannungen (-50 bis +50 mV). Der **Multiplexer** (MUX) ist ein Kanalschalter, der vom Adressbus gesteuert wird. Er wählt einen Messkanal und leitet den Wert an den **Analog-Digital-Umwandler** (ADU). Der erzeugt ein 16-Bit- oder 32-Bit-Zahlenwort mit dem Wert der analogen Größe. Der Datentreiber bringt das Wort zum Speicher, wo für jeden Kanal ein Speicherwort für eine Zwischenspeicherung bereit steht, und auch zum Bustreiber, falls das SPS-Programm gerade diesen Wert haben möchte. Die Kanaladresse wird im Adressdecoder entschlüsselt.



- Ein **Kanal** ist die Nummer einer Signalklemme für eine Messleitung, meist pro Baugruppe von 0 bis 7 nummeriert
- Eine **Baugruppe** hat meist 8 Kanäle. Die **Baugruppennummer** ist eigentlich eine Speicheradresse der SPS im Portbereich für eine 8-er-Gruppe von Speicherworten. Der Portbereich ist ein Speicherbereich, auf den sowohl externe Geräte (z.B. der Bustreiber der Baugruppe) als auch die CPU (Central Processing Unit, d.h. der eigentliche μ -Controller in der SPS) zugreifen können, vergleichbar einem Bücherregal, das von beiden Seiten begehbar ist.
- Eine Messung heißt **potentialgetrennt**, wenn keine der beiden Zuleitungen geerdet ist. Messgröße ist dann die Differenzspannung. Jede Strippe wird gegen Erde gemessen und dann rechnerisch intern die Differenz gebildet, d.h. beide Messleitungen müssen sich nach dem Anpassungsmodul im Messbereich -50 bis +50 mV bewegen.
- Eine Messung heißt **potentialgebunden**, wenn nur eine spannungs- oder stromführende Messleitung benutzt wird und die zweite Leitung an der Sammelerde der Analogeingabe angeklemmt ist.
- Ein **Shunt** (eine elektrische Hilfsverbindung, d.h. ein definierter Widerstand zwischen Messleitung und Erde, meist als Spannungsteiler aus zwei Widerständen aufgebaut) wandelt bei Stromeingängen das Stromsignal in das geforderte Spannungssignal im Bereich 0-50 mV für den Wandler (ADU).
- **Drahtbruchmeldung** ist ein SPS-Service, der nur bei Stromeingängen angeboten wird. Vor jeder Messung wird 1,6 ms lang ein Konstantstrom auf die Messleitung aufgegeben. Der angeschlossene Messumformer muss das akzeptieren können, ohne Schaden zu nehmen. Steigt dabei die Spannung in der Messleitung auf den höchsten erreichbaren Wert des Konstantstromgebers (ca. 12 Volt), dann nimmt man Drahtbruch an, und eine entsprechende Fehlermeldung wird erzeugt. Zusätzlich muss man in der Baugruppe für diesen Kanal einen Jumper stecken (Jumper sind kleine Schalter oder Kurzschlussstecker, mit denen man bestimmte Bedingungen setzen kann. Die Elektronik fragt die Schalterstellung dann ab)
- Als **Abtastung** bezeichnet man das Messen eines Messwertes. Für die Regelung ist eine zyklische Abtastung in gleichen Zeitabständen wichtig (**Tastzeit TA**). Werden alle 8 Kanäle einer Baugruppe zyklisch abgetastet, dann ist bei einer Wandelzeit von ca. 60 ms/Kanal ein bestimmter Kanal nur alle 480 ms dran. Das ist etwa alle halbe Sekunde.
- **Einzelabtastung** eines Kanals heißt, dass der Multiplexer nur diesen einen Kanal wählt und der ADU nur diesen Kanal wandelt. Auf diese Weise kann man die 8-fache Abtastrate erreichen.

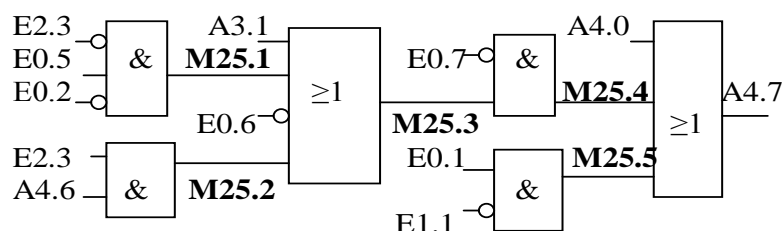
5.4.2 Die Programmiersprache AWL (Anweisungsliste)

Viele SPS-Programme werden auch heute noch in der Maschinensprache geschrieben. Das trifft insbesondere auf kompliziertere Kommunikationsprogramme, aber auch auf die Analogwertverarbeitung zu. Eine Maschinensprache verwendet abgekürzte Bezeichnungen für die Befehle und für die Daten. Ein Befehl macht immer nur sehr wenig, so dass ein Programmstück aus sehr vielen einzelnen Befehlen bestehen kann.

Liste einiger Befehle und Konstanten:

Beschreibung des Befehls oder der Konstanten	Beispiel
Sprung absolut zu einem Programmbaustein, z.B. PB1	SPA PB 1
Sprung absolut zu einer Zielmarke, z.B. zur Marke M007	SPA =M007
Bedingter Sprung zu einer Zielmarke. Dieser Sprung wird nur ausgeführt, wenn im 1-Bit-Vergleicherregister (VG) eine 1 steht	SPB =M007
Aufruf eines Datenblocks, z.B. DB 10. Die SPS merkt sich diese Nummer in einem Register. DB 10 wird zum <i>aktiven</i> Datenblock	A DB 10
Laden eines 2-Byte-Wortes (Wort 22 z.B.) aus dem aktiven Datenblock in das Akkumulatorregister Akku1 (der alte Inhalt wandert vorher nach Akku2. Dessen alter Inhalt verschwindet)	L DW 22
Laden eines Bytes vom Merkerspeicher nach Akku1	L MB 488
Transferieren (Umspeichern) des Akku1-Inhalts in den Speicher	T MB 555
Größenvergleich der beiden Akkuinhalte, z.B. Akku2 > Akku1 Das Ergebnis erscheint im Vergleich (Ja: VG=1, Nein: VG=0)	F>
Laden eines Merkerbits in den Vergleich. Das kann mit einem UND-Befehl oder mit einem ODER-Befehl geschehen	U M 25.3 O M 25.3
Negiertes Laden eines Merkerbits (z.B. eine 0 wird als 1 geladen)	UN M 25.3
Setzen einer 1 auf ein Merkerbit wenn der Vergleich VG=1 ist	S M 25.3
Setzen einer 0 auf ein Merkerbit wenn der Vergleich VG=1 ist	R M 25.3
Abspeichern des Vergleichereinhalts auf ein Merkerbit	= M 25.3
Festkommakonstante, z.B. eine Baugruppenadresse	KF +192
Doppelkonstante (2 Zahlen zu je einem Byte Speicherplatz)	KY 4, 7
Bitmaskenkonstante mit einer Bitbelegung eines 16-Bit-Wortes	KM 00110011 11110000

Beispiel einer logischen Schaltung aus UND- und ODER-Bausteinen. Kleine Kreise deuten negierte Eingänge an (aus 1 wird 0, aus 0 wird 1). Ein UND-Baustein (&) liefert am Ausgang nur 1, wenn alle Eingänge 1 sind (nach einer eventuellen Negation). Ein ODER-Baustein (≥ 1) liefert am Ausgang nur 0, wenn alle Eingänge 0 sind (nach einer eventuellen Negation).



Das entsprechende AWL-Programm benutzt die Merker (1-Bit-Speicher) M25.1 bis M25.5.

: UN E2.3	: = M25.2	: UN E0.7	: = M25.5
: U E0.5	: O A3.1	: U M25.3	: O A4.0
: UN E0.2	: O M25.1	: = M25.4	: O M25.4
: = M25.1	: ON E0.6	: U E0.1	: O M25.5
: U E2.3	: = M25.3 →	: UN E1.1 →	: = A4.7
: U A4.6 →			

Kleine Beispiele für die Konstruktion logischer Netzwerke aus UND- und ODER-Bausteinen:

Von den Anzeige-LEDs der Prozessschritte ist immer nur eine eingeschaltet (Signal 1). Der Rührer A4.3 soll in den Prozessschritten LED1, LED3 und LED4 immer laufen, im Schritt LED7 aber nur, wenn Pegelmelder P1 berührt ist.	
Heizung A4.7 soll in den Prozessschritten LED2 und LED4 über Temperaturfühler Temp3 zwei-punkt-geregelt laufen, im Schritt LED3 konstant.	
Leuchttafel A3.0 mit Aufschrift „Magazin nachfüllen“ soll leuchten, wenn die Lichtschranke E1.1 Durchgang des Laserstrahls (Signal 1) meldet.	

5.4.3 Datenbaustein für die PID-Regelung (bei der S5-Serie von SPSen benötigt)

Für jeden Regelkreis benötigt man einen eigenen Datenbaustein. Datenbausteine werden nummeriert, z.B. DB 1, DB 2, ..., DB 255. Der Datenbaustein einer PID-Regelung umfasst genau 49 Datenworte a 16 Bit (2 Byte). In diesem Datenbaustein werden alle wichtigen Informationen zusammengetragen, damit der PID-Algorithmus (vom Hersteller der SPS mitgeliefert) arbeiten kann. Der Datenbaustein wird vom Programmierer als *neuer Baustein* angelegt und dann ausgefüllt. Im laufenden Betrieb ändern sich dann nur noch der Sollwert W, die Istgröße x und die Stellgröße y. Die Tastzeit TA taucht hier nur indirekt bei der Berechnung von KI und KD auf.

Die wichtigsten Eintragungen im PID-Baustein der S5-Serie sind:

Wort	Inhalt	Beschreibung
DW1	K	Reglerverstärkung, z.B. KF +17053 bei $K_{PR}=17,053$ (Faktor 1000)
DW3	R	Zusatzfaktor im P-Anteil. Die Gesamtverstärkung ist hier $K \cdot R$ (Faktor 1000)
DW5	TI	=TA/TN Integrierkonstante, z.B. KF +44 bei TI=43,59 (Faktor 1000)
DW7	TD	=TV/TA Differenzierkonstante, z.B. KF +7 bei TD=6,51 (Faktor 1)
DW9	W	Sollwert im Zahlenbereich von -2047 bis +2047
DW11	STEU	Steuerwort mit Steuerinformation (siehe unten)

DW12	yH	Handwert der Stellgröße y bei Umschaltung in Handbetrieb
DW14	BGOG	Obere Begrenzung der Stellgröße y im Zahlenbereich von -2047 bis +2047
DW16	BGUG	Untere Begrenzung der Stellgröße y im Zahlenbereich von -2047 bis +2047
DW22	X	Istwert der Regelgröße im Zahlenbereich von -2047 bis +2047
DW24	Z	Hauptstörgröße bei Aufschaltung im Zahlenbereich von -2047 bis +2047
DW48	y	Ausgabewert der Stellgröße y im Zahlenbereich von -2047 bis +2047

Das **Steuerwort** besteht aus 2 Bytes. Das high Byte wird vom Programmierer mit Nullen belegt und darf im Betrieb nur vom PID-Baustein selbst verändert werden. Die Bits 0 bis 5 des low Byte haben jeweils eine Steuerfunktion, z.B.

Bit0 = 0 heißt Handbetrieb	Bit0 = 1 heißt Automatikbetrieb
Bit2 = 0 heißt Regelbaustein arbeitet	Bit2 = 1 heißt Regelbaustein arbeitet nicht
Bit3 = 0 heißt K-Regler	Bit3 = 1 heißt S-Regler
Bit5 = 0 heißt ohne Aufschaltung	Bit5 = 1 heißt mit Aufschaltung

5.4.4 Lesen von Analogsignalen mit FB 250 (bei der S5-Serie von SPS-en)

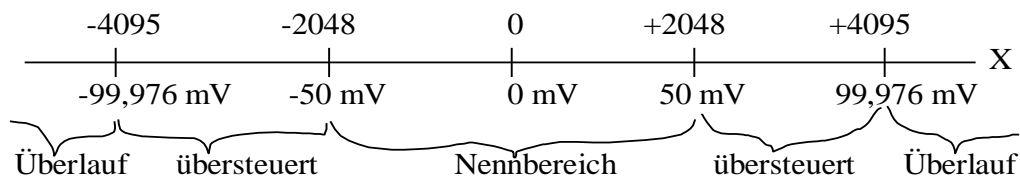
Es ist nicht üblich, die gewandelten Zahlenwerte direkt aus den Portspeichern zu lesen. Dafür gibt es einige Gründe. Der AWL-Programmierer bedient sich viel mehr eines vom Hersteller mitgelieferten Programms (Funktionsbaustein FB 250 bei der S5-Serie). Das Eingabewort der SPS hat 16-Bit und den folgenden Aufbau:

Byteadressen	n								n+1							
Bitnummern im Byte																
Bedeutung der Bits																
Bedeutung des Bytes	high Byte								low Byte							

Die Bits 2^{12} bis 2^0 enthalten den gewandelten Zahlenwert in binärer Form. Sie sind mit Null oder Eins belegt je nach Zahlenwert. Der Zahlenwert kann auch negativ sein (siehe 2-er-Komplement). Dabei enthält das *high Byte* die höherwertigen Dualstellen, das *low Byte* die niederwertigen. Die Byteadresse des high Bytes ist jedoch niedriger als die des low Byte. Beide Bytes liegen im Speicher unmittelbar hintereinander. Die drei letzten Bits (T, F, Ü) im low Byte sind für Eingabefehlermeldungen reserviert. Tritt kein Fehler auf, sind sie 0.

Bitbelegung T=1 meldet Tätigkeit des Wandlers (z.B. Einzelabtastung für einen anderen Kanal)
 Bitbelegung F=1 meldet einen Drahtbruch (nur bei Stromeingang und gesetztem Jumper)
 Bitbelegung Ü=1 meldet Überlauf. Der Wert der analogen Größe ist zu groß.

Intern wird die analoge Größe auf eine Zahlengerade X abgebildet:



Die Ausdehnung des Nennbereichs kann jedoch durch die Angabe neuer Grenzen noch abgeändert werden, z.B. so, dass der positive Bereich von 0 bis 50 mV auf die Zahlen 0 bis 1000 abgebildet wird. Einen Merker mit ständiger Bitbelegung 0 bezeichnet man als **Nullmerker**.

Sobald der Programmierer am Programmiergerät den Befehl "**SPA FB 250**" eintippt und die ENTER-Taste betätigt, erscheint nach dem Befehl eine farbige Maske zum Ausfüllen (fette Schrift markiert die Maske):

	: SPA FB	250	Aufruf der Analogeingabe, z.B. eines Sollwertes
Name	: RLG:AE		Diese Zeile sollten Sie nicht ändern
BG	: KF	+192	Baugruppe
KNKT	: KY	0, 4	Kanalnummer, Kanaltyp (Der Kanaltyp ist codiert)
OGR	: KF	+1000	Obere Grenze Nennbereichsumrechnung (2048—>1000)
UGR	: KF	+0	Untere Grenze Nennbereichsumrechnung (0 —> 0)
EINZ	: M	12.0	Falls M 12.0 ein Nullmerker ist, dann keine Einzelabtastung
XA	: DW	9	Zielort der Eingabe ist hier Datenwort 9 im aktuellen DB
FB	: M	25.0	Zielmerker für <i>Drahtbruchmeldung</i> ist hier Merker 25.0
BU	: M	25.1	Zielmerker für Meldung <i>Bereichsüberschreitung</i>
TBIT	: M	25.2	Zielmerker für Meldung <i>ADU anderweitig beschäftigt</i>

Einige Kanaltypen für die Eingabe

Typ	Beschreibung
3	Stromeingabe 4 bis 20 mA
4	Spannungseingabe unipolar, d.h. nur positive Spannungen von 0 bis 10 V
6	Spannungseingabe bipolar, d.h. alle Spannungen von -10 bis +10 V

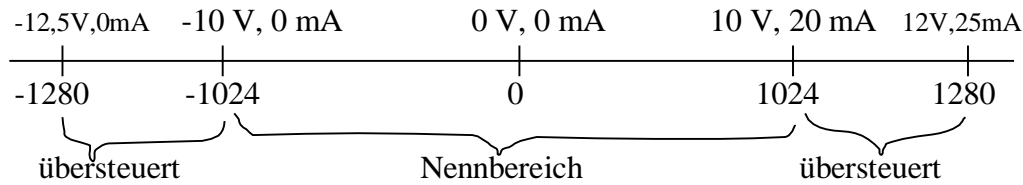
5.4.5 Ausgeben von Analogsignalen mit FB 251 (bei der S5-Serie von SPS-en)

Eine Analogausgabe benötigt man z.B., um den Hub eines Regelventils einzustellen oder die Rührerdrehzahl im Fermenter einzustellen.

Das Ausgabewort der SPS hat ebenfalls 16-Bit und den folgenden Aufbau:

Byteadressen	n								n+1							
Bitnummern im Byte	7	6	5	4	3	2	1	0	7	6	5	4	3	2	1	0
Bedeutung der Bits	2^{11}	2^{10}	2^9	2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	X	X	X	X
Bedeutung des Bytes	high Byte								low Byte							

Die Bits 2^{11} bis 2^0 enthalten den Zahlenwert in binärer Form, der als Analogwert, z.B. als Strom- oder Spannungssignal ausgegeben werden soll. Die 4 mit X markierten Bits sind beliebig, jedoch belegen sie die meisten Programmierer mit Nullen. Um diese Details bemüht sich jedoch der Funktionsbaustein FB 251. Extern wird die Zahlengerade X entweder auf einen Stromwert oder auf einen Spannungswert abgebildet (es gibt jedoch keine negativen Ströme):



Sobald der Programmierer am Programmiergerät den Befehl "**SPA FB 251**" eintippt und die ENTER-Taste betätigt, erscheint nach dem Befehl eine farbige Maske zum Ausfüllen (fette Schrift markiert die Maske):

	: SPA FB 251	Aufruf der Analogausgabe, z.B. Stellsignal für ein Ventil
Name	: RLR:AA	Diese Zeile sollten Sie nicht ändern
XE	: DW 48	Herkunft der auszugebenden Zahl (hier aus aktuellem DB)
BG	: KF +224	Baugruppe
KNKT	: KY 0, 0	Kanalnummer, Kanaltyp (Der Kanaltyp ist codiert)
OGR	: KF +1000	Obere Grenze Nennbereichsumrechnung (1000—>1024)
UGR	: KF +0	Untere Grenze Nennbereichsumrechnung (0 —> 0)
FEH	: M 26.0	Zielmerker für Fehlermeldung $UGR \geq OGR$ ist hier M 26.0
BU	: M 26.1	Zielmerker <i>Bereichsüberschreitung</i> $XE < UGR, XE > OGR$

OGR und UGR dürfen hier nicht willkürlich gesetzt werden, wie bei der Analogeingabe, sondern müssen identisch sein mit den Grenzvorgaben für die Stellgröße im Datenblock des Regelkreises.

Achtung ! An einem Spannungsausgang müssen mindestens **3,3 K Ω** Widerstand hängen, da sonst die Stromstärke zu hoch wird
 An einem Stromausgang dürfen maximal **300 Ω** Widerstand hängen, da sonst die erforderliche Spannung zu hoch wird

Kanaltypen der Ausgabe:

Z.B. ist Kanaltyp 0 ein unipolarer Spannungsausgang mit dem Ausgabebereich 0 bis 10 Volt

Z.B. ist Kanaltyp 1 ein bipolarer Spannungsausgang mit dem Ausgabebereich -10 bis 10 Volt

5.4.6 Der PID-Baustein OB251 (bei den S5-Serien)

Für die Regelung wichtig ist eine konstante Tastzeit T_A , die beim Start des Programms festgelegt wird. Aus der Gleichung des idealen PID-Reglers mit den Bestandteilen **P**roportional, **I**ntegrierend, **D**ifferenzierend

$$y = K_{PR} \left[R \cdot xd + \frac{1}{T_N} \int_0^t xd(u) du + T_V \dot{xd} \right]$$

mit $xd = W - x$ ergibt sich nach einer Differenziation

$$\dot{y} = K_{PR} \left[R \cdot \dot{xd} + \frac{1}{T_N} xd + T_V \ddot{xd} \right]$$

und mit näherungsweise

$$\dot{y}_n = \frac{y_n - y_{n-1}}{TA}, \quad \dot{x}d_n = \frac{x d_n - x d_{n-1}}{TA} \quad \text{und} \quad \ddot{x}d_n = \frac{\dot{x}d_n - \dot{x}d_{n-1}}{TA} = \frac{x d_n - 2x d_{n-1} + x d_{n-2}}{TA^2}$$

die Differenzengleichung des idealen PID-Algorithmus (DIN-Algorithmus)

$$y_n = y_{n-1} + K_{PR} \left[R(x d_n - x d_{n-1}) + \frac{TA}{T_N} x d_n + \frac{T_V}{TA} (x d_n - 2x d_{n-1} + x d_{n-2}) \right]$$

Hierbei ist:

- n eine fiktive Nummerierung 1, 2, 3, ... der Tastzeiten, die im Rechner nicht wirklich erfolgt
- y_n der Stellgrößenwert nach der n-ten Abtastung berechnet
- K_{PR} der Teil der Reglerverstärkung, der auf alle 3 PID-Anteile wirken soll (P-, I-, D-Anteil)
- R der Teil der Reglerverstärkung, der zusätzlich den P-Anteil steuern soll
- x_d_n = W-x die Regeldifferenz Sollwert minus Istwert im n-ten Tastschritt
- x_d_{n-1} die Regeldifferenz des vorangehenden Tastschritts
- x_d_{n-2} die Regeldifferenz zwei Tastschritte davor
- T_N die Nachstellzeit des PID-Reglers mit Faktor K_{PR}TA/T_N als Faktor des I-Anteils
- T_V die Vorhaltezeit des PID-Reglers mit Faktor K_{PR}T_V/TA als Faktor des D-Anteils
- Der I-Anteil wird 0, wenn Faktor TA/T_N Null gesetzt wird
- Der D-Anteil wird 0, wenn Faktor T_V/TA Null gesetzt wird

Ein **S-Regler** (Schritt- oder Geschwindigkeitsalgorithmus) gibt nicht y_n aus, sondern die Differenz Δy = y_n - y_{n-1}. Die Summation erfolgt im Stellglied. Beispiel sei ein Regelventil, das durch einen Motor weiter auf- bzw. weiter zu gemacht werden soll. Ein positives Δy bewirkt Rechtslauf des Motors und damit ein Auffahren des Ventils, ein negatives Δy hingegen Linkslauf und Zufahren. Der Betrag |Δy| ist der Drehzahl des Motors proportional. Der Regelkreis benötigt eine Rückmeldung über die genaue Stellung des Stellgliedes, aber keinen Stellungsregler.

Ein **K-Regler** (kontinuierlicher Regler) gibt y_n aus. Die genaue Einstellung des Stellgliedes übernimmt ein Stellungsregler im Stellantrieb. Dafür spart man die Rückmeldung über die Stellung des Stellgliedes ein, die beim S-Regler notwendig ist.

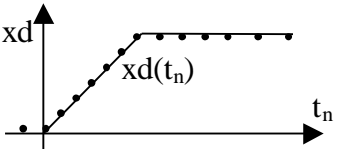
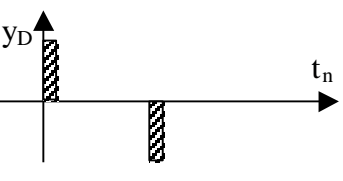
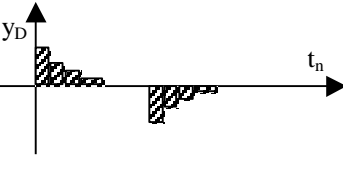
Der PID-Algorithmus ist als Organisationsbaustein OB 251 (S5-Serie) bereits in das Betriebssystem der SPS integriert. Man ruft ihn einfach auf. Der Aufruf muss jedoch zyklisch zeitgesteuert erfolgen, damit die Tastzeit TA peinlich genau eingehalten wird. Der Aufruf wird durch eine in die SPS integrierte Uhr ausgelöst, deren Zeitimpulse einen Interrupt in der CPU auslösen. Ist die Tastzeit eines Regelkreises abgelaufen, wird z.B. Organisationsbaustein OB13 aufgerufen, der dann die eigentliche PID-Regelung anstößt (Siehe Organigramm unten).

4 verschiedene **Tastzeiten** sind möglich: TA₁, TA₂, TA₃ und TA₄. Ist TA₁ abgelaufen, wird OB13 per Interrupt aufgerufen, ist TA₂ abgelaufen, dann OB12, ist TA₃ abgelaufen, dann OB11, ist TA₄ abgelaufen, dann OB10. Laufen mehrere Tastzeiten gleichzeitig ab, gilt eine Vorrangregelung. Wozu braucht man unterschiedliche Tastzeiten? Z.B. ändert sich die Temperatur in einem großen Fermenter nur langsam. Hier reichen Tastzeiten von 10 Sekunden und mehr vollkommen aus für eine Regelung. Ist es ein Membranreaktor, dann kann sich jedoch der Membrandruck in Minuten dramatisch ändern, und man regelt die Pumpenleistung besser mit einer kürzeren Tastzeit, z.B. 1 Sekunde.

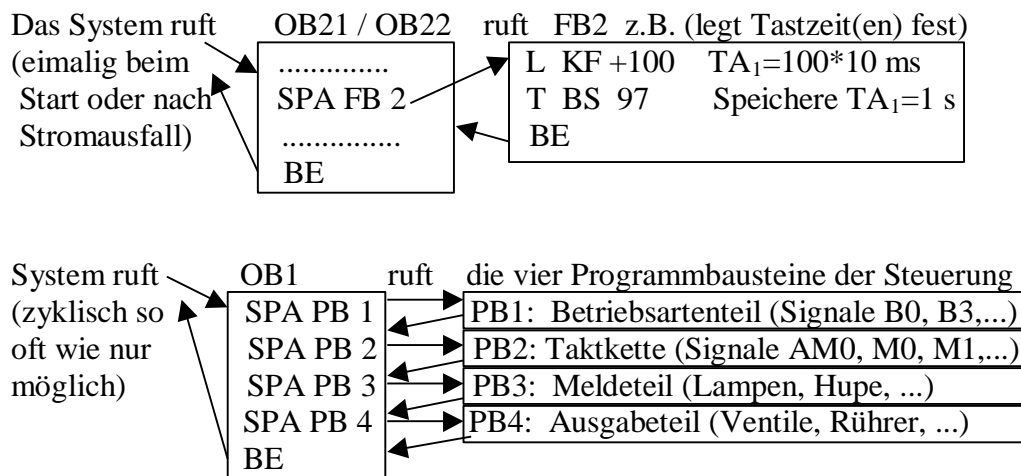
Der **ideale** und der **reale PID-Regler** (letzterer auch **PIDT₁**-Regler genannt) unterscheiden sich in der Behandlung des differenzierenden Anteils (D-Anteil). Der reale PID-Regler hat einen zusätzlichen Reglerparameter **VV** (Vorhalteverstärkung) bzw. **T₁** (Zeitkonstante des D-Anteils) je nach Hersteller. Die beiden Größen sind über die Formel $VV = T_V / T_1$ verknüpft. Die DGL des realen PID-Reglers enthält auf der linken Seite zusätzlich die Ableitung des D-Anteils nach der Zeit:

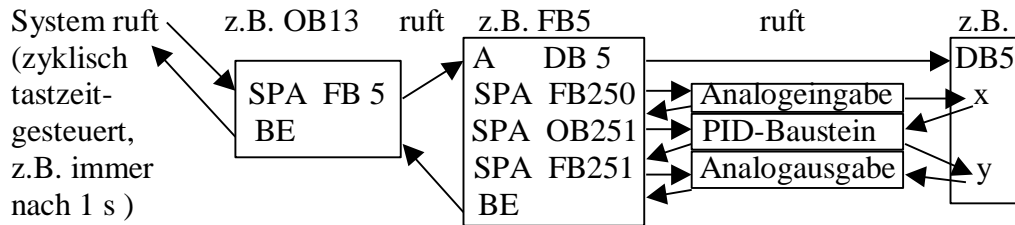
$$T_1 \dot{y}_D + y = K_{PR} \left[R \cdot x_d + \frac{1}{T_N} \int_0^t x_d(u) du + T_V \dot{x}_d \right]$$

Den Unterschied in der Wirkung zeigen die folgenden kleinen Graphiken:

	<p>Eine Sollwertrampe $W(t)$ führt über die Gleichung $x_d = W - x$ zu einer Rampe der Regeldifferenz x_d, wenn die Regelung nur langsam genug reagiert. Die Punkte stellen die Tastzeitpunkte dar, an denen der Istwert x gemessen wird.</p>
	<p>Der ideale PID-Regler erzeugt nur an den Knickstellen der x_d-Kurve y_D-Impulse (die aber von beträchtlicher Größe), denn nur dort ist eine Krümmung (zweite Ableitung) der x_d-Kurve vorhanden. Der Gesamt-y-Wert kann auf diese Weise negativ werden oder über 100% klettern, so dass die Wirkung verpufft.</p>
	<p>Der reale PID-Regler verteilt die starken y_D-Impulse auf eine Treppe von Impulsen mit abnehmender Höhe, aber derselben Fläche. Dadurch hält die Wirkung länger an, und die Gefahr sinkt, dass das Gesamt-y negativ wird oder über 100% steigt.</p>

Organigramm eines SPS-Steuer- und Regelprogramms (S5-Serie aus dem AT-Praktikum)





5.4.7 Graphische Reglerprogrammierung (Ausblick auf neuere SPS-en)

Die S5-Serie ist veraltet und wird im Skript nur deshalb so ausführlich behandelt, weil

1. sie im AT-Praktikum von UV/BT immer noch verwendet wird
2. die Grundprinzipien sich kaum geändert haben

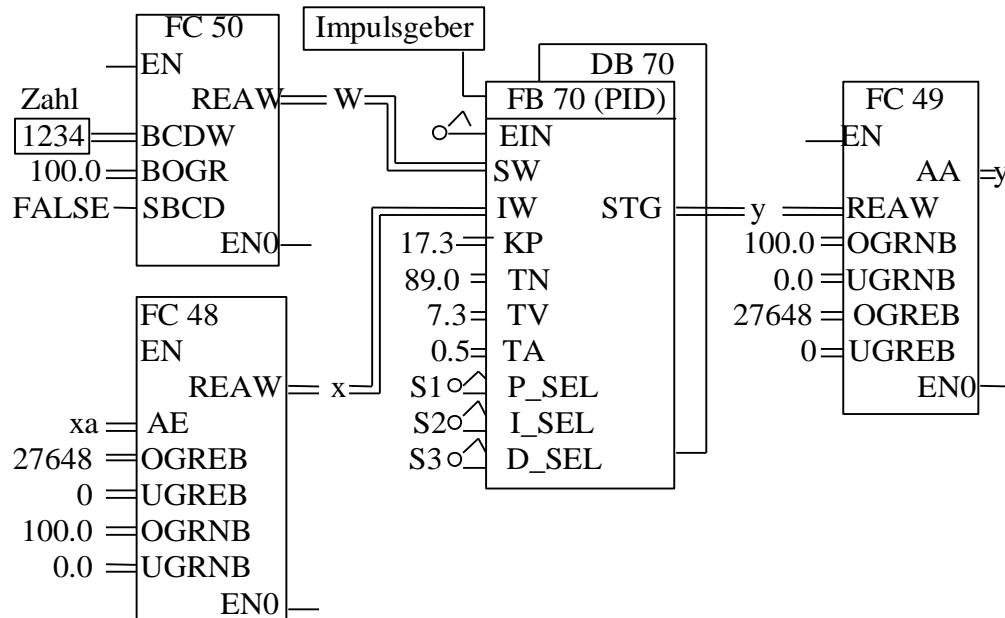
Neuere SPSen haben Gleitkommaprozessoren, wesentlich mehr Programm- und Datenspeicher, eine höhere Geschwindigkeit und bessere Kommunikationsmöglichkeiten. Augenfällig in der Programmierung ist, dass viele Standard-Funktionsbausteine neben ihrer AWL-Aufrufmaske auch eine graphische Aufruf-Repräsentation in Boxenform haben. Mit diesen Boxen können Datenflusspläne (Regelungsprogramme) ähnlich den Funktionsplänen der digitalen Steuerung gezeichnet werden. Für viele Standardaufgaben reicht diese graphische Programmiermethode vollkommen aus. Zudem zwingt sie zur Verwendung vorgefertigter ausgetesteter Funktionsbausteine und vermindert so die Zahl der Programmierfehler. Für tiefergehende Studien oder praktische Anwendungen wird die Literatur empfohlen, z.B. Wellenreuther / Zastrow. Gegenüber der S5-Serie sind z.B. bei der S7-Serie unter anderem neue Bausteinnummern eingeführt worden und z.T. neue Datenblockkonzepte.

Die folgende Graphik (an Wellenreuther / Zastrow angelehnt) zeigt die Eingabe eines Sollwertes über eine Zifferneingabetastatur und die Wandlung der Dezimalzahl mit der Funktion FC 50 in eine REAL-Zahl (Gleitkommaformat intern), die Analogeingabe des Istwertes mit Funktion FC 48 und die Normierung der gelesenen REAL-Zahl auf den Bereich 0 bis 100, den Aufruf des PID-Bausteins (FB 70), und die Stellgrößenausgabe mit Funktion FC 49. Neu ist z.B. die Vorgabe der Tastzeit TA zusammen mit den Reglerparametern. Für die Speicherung solcher unveränderlichen Konstanten benutzt die S7-Philosophie sogenannte Instanz-Datenbausteine. Jede Reglerbox ist z.B. eine Instanz (eine Kopie) der allgemeinen PID-Aufrufmaske und hat einen eigenen Konstantenspeicher, eben diesen Instanz-Datenbaustein. Zusätzlich sind aber auch allgemein zugängliche Datenbausteine möglich, wie wir sie von der S5-Serie her kennen.

In der folgenden Graphik bedeutet:

REAW	ein Datenwort vom Typ REAL
BCDW	ein Datenwort im BCD-Code (einzeln verschlüsselte Dezimalziffern)
BOGR	obere Begrenzung der gewandelten Zahl
FALSE	logischer Wert "false" (z.B. ein Nullmerker)
SBCD	Selektor BCD, d.h. sind auch die Pseudoziffern A, B, ..., F erlaubt?
DB 70	Der Instanzdatenblock zu gerade diesem PID-Aufruf
SW, IW, KP, TN	Sollwert, Istwert, Reglerverstärkung, Regler-Nachstellzeit
TV, TA	Regler-Vorhaltezeit, Tastzeit
S1, P_SEL	Schalter 1, Selektor für den P-Anteil aus der Stellgröße $y=P+I+D$

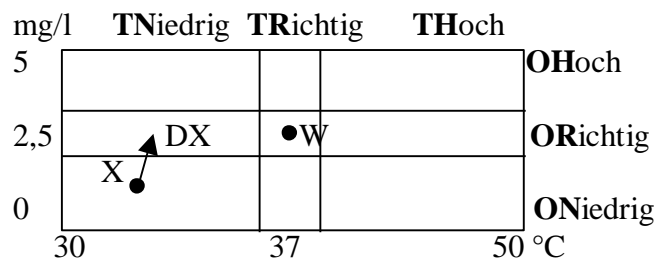
OGRNB, UGRNB Obere und untere Begrenzung des Nennbereichs
 OGREB, UGREB Obere und untere Begrenzung der internen Festkommandarstellung von y



Für die Anlagenumrüstung, z.B. von der veralteten S5-Serie auf die modernere S7-Serie stehen **Interface-Module** zur Verfügung, die es ermöglichen, einzelne SPS-en ohne großen Verdrahtungsaufwand und Stillstand auszutauschen.

5.5 Fuzzy Control

Lotfi Zadeh führte 1965 den Begriff der unscharfen Logik ein (Fuzzy Logic). Statt **wahr** und **falsch** sind jetzt auch Grautöne zugelassen. Meist wird false=0 und wahr=1 gesetzt. Die Fuzzy Logic rechnet demnach mit allen Zahlen aus dem Intervall [0 , 1]. Das folgende Beispiel orientiert sich an der gemeinsamen Regelung von Temperatur **T** und O₂-Konzentration **O**. Wir stellen uns eine kleine Graphik auf dem Bildschirm des Leitstandes vor:



Der Sollpunkt **W** mit $T=37^{\circ}\text{C}$ und $O_2=2,5 \text{ mg/l}$ liegt genau in der Mitte. Die Ordinate bzw. y-Richtung wird von der O₂-Konzentration **O** gebildet, die Abszisse oder x-Richtung von der Temperatur **T**. Der Ist-Punkt **X** zeigt, dass beide Werte, T und O, zu niedrig sind. Der Geschwindigkeitspfeil **DX** zeigt, dass sich der X-Punkt in der letzten Minute mit erheblicher Geschwindigkeit

in die angezeigte Richtung bewegt hat. DX setzt sich aus den Komponenten Temperaturänderung pro Zeiteinheit **DT** [K/min] und O₂-Änderung pro Zeiteinheit **DO** [mg/(l·min)] zusammen. DT und DO werden im Programm mit geeigneten Faktoren multipliziert und als Wanderungsgeschwindigkeit DX des X-Punktes auf dem Bildschirm als Pfeil dargestellt in [mm/min].

Fuzzification

Fuzzification ist die Zuordnung von Zahlenwerten zu den **Prädikaten** unserer vier **linguistischen Eingangsvariablen** über **Zuordnungsfunktionen**.

Die 4 linguistischen Eingangsvariablen sind

T DT O DO

Die 4·3=12 Prädikate bzw. Zuordnungsfunktionen sind

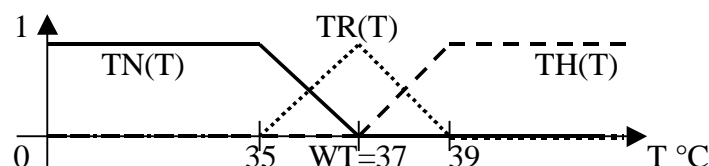
T Niedrig	TR ichtig	TH och
DT Negativ	DT unerheblich	DT ositiv
ON iedrig	OR ichtig	OH och
DO Negativ	DO unerheblich	DO ositiv

Temperatur und O₂-Konzentration können demnach mehr oder weniger niedrig, richtig oder hoch sein. Die Änderungsgeschwindigkeiten DT bzw. DO können mehr oder weniger negativ, unerheblich oder positiv sein. Steigt die Temperatur schnell, ist DT positiv. Steigt die Temperatur langsam oder sinkt sie langsam, ist DT unerheblich, usw.

Eine **Zuordnungsfunktion** gibt einem Prädikat einen Wert, z.B. die Funktion $TN(T)$ wertet die Temperatur T aus und vergibt danach einen Prädikatwert zwischen 0 und 1, je nachdem wie viel *zu niedrig* die Temperatur ist.:

$$TN(T) = \begin{cases} 1, & \text{wenn } T \leq W_T - 2 \text{ } ^\circ\text{C} \\ 0, & \text{wenn } T \geq W_T \\ (W_T - T) / 2, & \text{sonst} \end{cases}$$

W_T ist der Temperatursollwert, im Beispiel hier 37 °C. Die Breite des Richtigbereichs der Temperatur sei hier $\pm 2 \text{ } ^\circ\text{C}$. Die unterste Zeile der Funktionsdefinition von $TN(T)$ ist eine fallende Gerade mit der Steigung $-1/2$. Die folgende Graphik zeigt die drei Zuordnungsfunktionen $TN(T)$ durchgezogen, $TR(T)$ gepunktet und $TH(T)$ gestrichelt. Wie man sieht, liefert die Funktion **TR**ichtig(T) nur den Wert 1, wenn die Temperatur genau im Sollwert 37°C ist. Die Funktion **TH**och dagegen liefert den Wert 1 dagegen erst, wenn die Temperatur T über 39 °C steigt. Zwischen 37 und 39 °C ist die Temperatur *mehr oder weniger hoch*, d.h., es wird ein Wert zwischen 0 und 1 geliefert.



In ähnlicher Weise werden alle 12 Zuordnungsfunktionen programmiert. Natürlich sind auch andere Funktionsformen als die dargestellten Rampen möglich, nur sollte die Summe der Zuordnungsfunktionen zu einer Variablen, z.B. $TN(T)+TR(T)+TH(T)$ zur Variablen T, immer möglichst genau den Wert 1 ergeben. Hier bei den Rampen ist diese Bedingung immer exakt erfüllt.

Beispiel: $T=35,64\text{ °C}$
 Solltemperatur $W_T=37\text{ °C}$
 $TN(T)=(37 - 35,64)/2= 0,68$ d.h., zu etwa 2/3 niedrig
 $TR(T)=(35,64 - 35)/2= 0,32$ d.h., zu 1/3 richtig
 $TH(T)=0$ d.h., kein bisschen zu hoch
 $TN+TR+TH=1$

Inference

Inference (oder Schlussfolgerung) heißt die Abfrage von **Expertenregeln** aus einer **Wissensbasis** nach einer bestimmten Methode, z.B. der **Min-Max-Methode**. In diesem Schritt bekommen die **Ergebnisfunktionen** ihre Werte.

Expertenregeln gewinnt man aus der Befragung von Experten. So sagt z.B. Meister Schröder in einer bestimmten Regelsituation: "Wenn die Temperatur **Niedrig** **UND** die Temperaturänderungsgeschwindigkeit **DT** Unerheblich ist, dann musst du **HeiZen**."

Es gibt zahlreiche Rechenregeln für die unscharfe Logik, die mehr oder weniger kompliziert sind. Die Min-Max-Methode z.B. ersetzt die zweiwertige dyadische logische Operation *Konjunktion* ($a \wedge b$ bzw. "a UND b") durch die Funktion Minimum(a, b), die *Disjunktion* ($a \vee b$ bzw. Alternative, Adjunktion, inklusives "a ODER b") durch die Funktion Maximum, und die monadische zweiwertige logische Operation *Negation* ($\neg a$) durch die Funktion "Rest zu Eins". Mit false=0 und true=1 ist

Funktion	scharfe Logik	unscharfe Logik
a UND b	$1 \wedge 1 = 1$ $1 \wedge 0 = 0$ $0 \wedge 1 = 0$ $0 \wedge 0 = 0$	Minimum(a, b)
a ODER b	$1 \vee 1 = 1$ $1 \vee 0 = 1$ $0 \vee 1 = 1$ $0 \vee 0 = 0$	Maximum(a, b)
Nicht a	$\neg 1 = 0$ $\neg 0 = 1$	$1 - a$

Die 5 Ergebnisfunktionen in unserem Beispiel könnten sein:

HZ	für eine digitale Ansteuerung der Heizung (Ein / Aus)
RN Rühre nicht	für eine kontinuierliche Drehzahlregelung des Rührers geeignet
RW Rühre wenig	
RM Rühre mittelstark	
RS Rühre stark	

Da besonders bei Befragung mehrerer Experten zu einer Ergebnisfunktion mehrere Regeln mit unterschiedlichen Werten zu einer Ergebnisfunktion führen können, muss man die Werte irgend wie verknüpfen. Zwei landläufige Methoden sind die **Mittelung** der Werte zu einer Ergebnisfunktion oder die **Maximumbildung**, was einer ODER-Verknüpfung der Einzelwerte entspricht. Bei beiden Verfahren werden alle Ergebnisfunktionen vor dem Start der Inference auf den Wert 0 ge-

setzt, z.B. $HZ=0$, $RN=0$, ..., $RS=0$,. Bei der Mittelwertbildung muss zusätzlich noch die Anzahl n der addierten Werte 0 gesetzt werden, z.B. $nHZ=0$, $nRN=0$, ..., $nRS=0$.

Benutzen wir die Maximumbildung als Verknüpfung mehrerer Regeln zu einer Ergebnisfunktion und C als Programmiersprache, dann würde obige Beispielregel von Meister Schröder etwa so umgesetzt werden:

$$HZ = \text{Maximum} (HZ, \text{Minimum} (TN, DTU))$$

Die Funktion Maximum realisiert hier die Verknüpfung mehrerer Regeln zum Heizen, die Funktion Minimum realisiert das UND aus Meister Schröders Heizregel. Der neue Wert von Ergebnisfunktion HZ wird wieder auf seinen alten Platz zurück gespeichert

Defuzzification

Die Defuzzification berechnet Stellgrößen aus den Werten der Ergebnisfunktionen. Bei Schaltern reicht als Ergebnis ein Ein/Aus. Bei kontinuierlichen Größen können beliebige Werte herauskommen. Manchmal muss man sie jedoch normieren bzw. beschränken, z.B. auf ein Intervall $[0, 1]$ oder ein Intervall $[0, 100\%]$.

Bei digitalen zweiwertigen Stellgrößen (Ein-Aus-Größen) arbeitet man zumeist mit einem willkürlich wählbaren Schwellenwert aus dem Intervall $[0, 1]$, z.B. 0,5 in diesem Beispiel für die Schalterstellung H der Heizung.

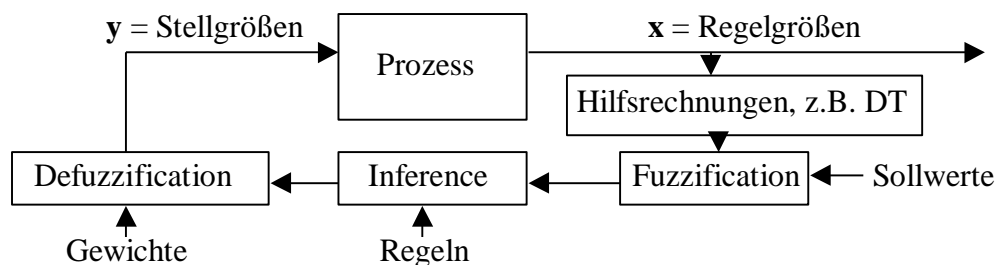
$$H = \begin{cases} 1, & \text{wenn } HZ > 0,5 \\ 0, & \text{wenn } HZ \leq 0,5 \end{cases}$$

Bei kontinuierlichen Stellgrößen ohne Normierung benutzt man z.B. die COM-Methode (Center Of Mass oder Schwerpunktmethod), wie z.B. zur Berechnung der Rührerdrehzahl R:

$$R = RN \cdot 0 + RW \cdot 30 + RM \cdot 60 + RS \cdot 120$$

Die gewählten Gewichte 0, 30, 60, 120 bestimmen den Stellgrößenverlauf. Die maximal mögliche theoretische Rührerdrehzahl ist hier 120 U/min.

Blockschaltbild der Fuzzy Control



Wir sehen einen geschlossenen Regelkreis. Die Regelgrößen x bilden einen Vektor (da es mehrere sind), ebenso die Stellgrößen y . Die Sollwerte gehen in die Fuzzification ein und ändern dort

die Zuordnungsfunktionen. Die Regeln gehen in die Inference ein. Die Gewichte gehen in die Berechnung der Ergebnisfunktionen ein. Insgesamt hat man in der Fuzzy Control mehr Möglichkeiten der Einflussnahme, als bei einer Regelung mit herkömmlichen PID-Reglern, und kann auf diese Weise auch nichtlineare und/oder gekoppelte Systeme regeln.

Beispiel einer Hilfsrechnung, z.B. DT aus T berechnen

Zur Ausführung der eigentlichen Regelaufgaben sind oft Hilfsrechnungen notwendig. So werden z.B. Temperaturänderungsgeschwindigkeiten fast nie gemessen, sondern berechnet. DT ist die Änderungsgeschwindigkeit der Temperatur. Ist **dt** die Dauer eines Tastschritts, dann gilt ungefähr

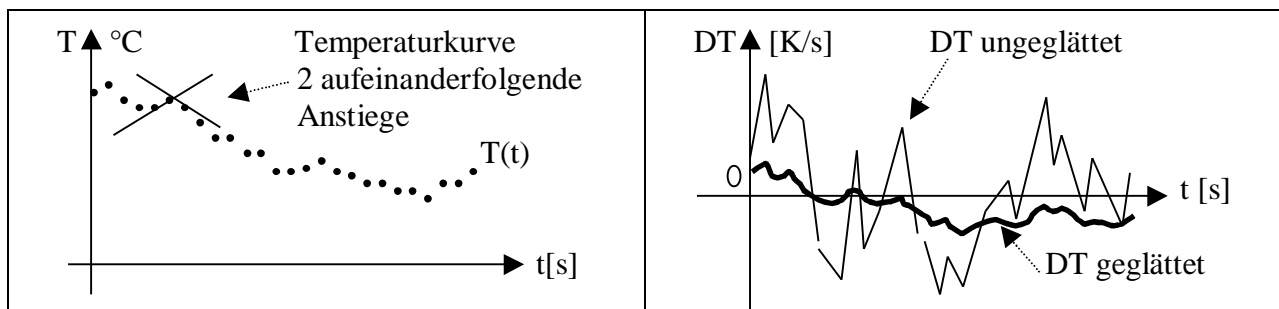
$$DT = \frac{\Delta T}{dt} = \frac{T_j - T_{j-1}}{dt}$$

T_{j-1} und T_j sind zwei aufeinanderfolgende Temperaturmessungen im Zeitabstand dt . Da differenzierte Kurven eine starke Neigung zur Aufrauung haben, muss man sie glätten. Das einfachste Glättungsverfahren ist die exponentielle Glättung (siehe auch Kap. 2.1 Gerätesteuerung mit μ -Controller). So werden z.B. vom alten DT-Wert 90% verwendet, vom neuen DT-Wert 10%. Störungen in den berechneten DT-Werten werden so auf 10% geschrumpft. Nachteil ist hier eine Zeitverzögerung von $10 \cdot dt$.

Vor der Steuerschleife:
`DT = 0; // Startwert für DT`
`C = 0.9; // Vergessensfaktor der exponentiellen Glättung`
`dt = 0.1; // Tastzeit Sekunden`
`T_alt = Messen(Kanal1); // Temperaturmessung`
`delay(100); // Warte 0.1 Sekunde`

In der Steuerschleife:
`T = Messen(Kanal1); // Neuen Temperaturwert messen`
`DT = C * DT + (1-C) * (T-T_alt)/dt; // DT berechnen und glätten`
`T_alt = T; // Mache aktuellen Messwert zum alten Messwert`
`delay(100); // Warte 0.1 Sekunde`

Die folgenden beiden Graphiken zeigen eine etwas raue T-Kurve, eine ungeglättete DT-Kurve und eine geglättete DT-Kurve. Ziel der Glättung sollte es sein, dass die DT-Kurve nicht rauer als die T-Kurve ist. Liegt der Vergessensfaktor C der exponentiellen Glättung sehr nahe bei 1, z.B. $C=0,999$, dann wird zwar die Glättung immer besser, aber es tritt ein hoher Zeitverzug von $dt/(1-C)=1000 \cdot dt=100$ Sekunden zwischen T-Kurve und geglätteter DT-Kurve auf .



5.6 Neuronale Netze

Literatur: S. Zakharian, P. Ladewig-Riebler, S. Thoer (1998): Neuronale Netze für Ingenieure, Vieweg. Aus diesem Buch stammt auch das Beispiel zur Regelungstechnik.

Wozu neuronale Netze?

- Zur Approximation von Funktionen ohne explizites mathematisches Modell, z.B. Darstellung von Kennlinien oder von Stabilitätsgrenzen von Reglern
- Für eine Klassifikation von Objekten oder Situationen, z.B. von Regelsituationen
- Zur Optimierung, z.B. einer adaptiven Regleranpassung an sich ständig ändernde Regelstrecken
- In der Bild-, Schrift- und Spracherkennung (Mustererkennung)

Welche Vorteile sieht man z.B. gegenüber Fuzzy-Control?

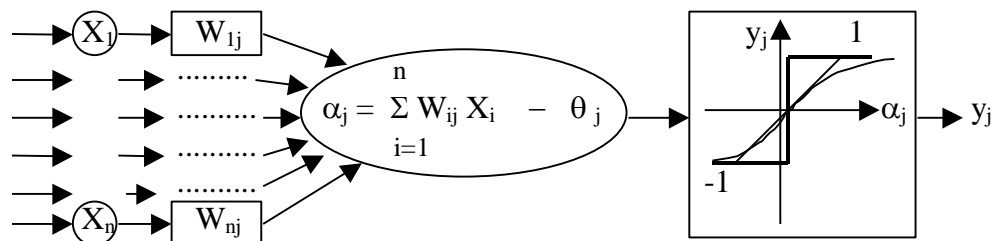
Es gibt nur Standardregeln, d.h., man muss keine Expertenregeln erfragen und notieren. Das neuronale Netz passt in einem Lernvorgang das Netz an die Lerndaten an.

Welche Nachteile gibt es?

Lange Rechenzeiten für das Lernen trotz schneller Prozessoren. Oftmals fehlt beim sogenannten "überwachten Lernen" das Ziel, d.h., es ist in der Regelungstechnik nicht immer klar, was eine *gute Regelung* ist bzw. wer genügend gute Vorlagen liefert.

5.6.1 Das Einzelschicht-Perzeptron nach F. Rosenblatt (1957)

Das Einzelschicht-Perzeptron besteht aus einer Schicht von k mathematischen Nachbildungen y_1, y_2, \dots, y_k einer **Nervenzelle**. Das Modell einer einzelnen Nervenzelle j funktioniert folgendermaßen: Es werden die **Eingangssignale** x_1, x_2, \dots, x_n des Neurons j mit den **Gewichten** w_{ij} gewichtet und addiert. Überschreitet die Gewichtssumme $\sum w_{ij} x_j$ einen festgelegten **Schwellwert** θ_j des Neurons, dann "feuert" das Neuron ein Signal y_j nach draußen. Die Signalhöhe kann in manchen Modellen mit ganz bestimmten **Aktivierungsfunktionen** von der Gewichtssumme $\sum w_{ij} x_j$ abhängen. Das folgende Bild zeigt das Modell eines einzelnen Neurons y_j :



Rechts im Bild sind drei Beispiele von Aktivierungsfunktionen $y(\alpha)$ eingezeichnet:

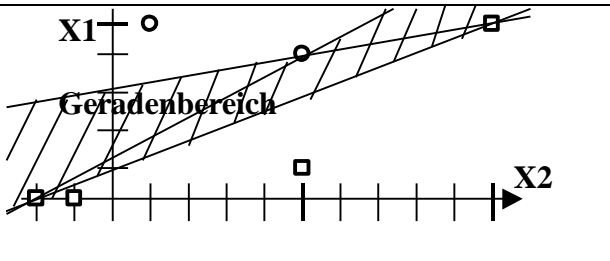
1. Die Sprungfunktion, die bei $\alpha=0$ einen Sprung von -1 nach $+1$ macht
2. Die Rampe, die in einem Toleranzbereich um $\alpha=0$ herum linear von -1 nach $+1$ steigt
3. Eine S-Funktion, die stetig und s-förmig von -1 nach $+1$ steigt

Das Einzelschicht-Perzeptron besteht aus k Neuronen y_1, y_2, \dots, y_k , die alle dieselben Eingangssignale x_1, x_2, \dots, x_n benutzen, aber schon ein einzelnes Neuron kann z.B. abhängig von der Zahl n der Eingangssignale verschieden komplizierte Klassifikationsaufgaben lösen, z.B. bei

- $n=1$ eine Zahlengerade durch einen Punkt in zwei Klassen teilen
- $n=2$ eine Ebene durch eine Gerade in zwei Klassen teilen
- $n=3$ einen Raum durch eine Ebene in zwei Klassen teilen

Hat man K Klassen zu unterscheiden, dann benötigt man mindestens $K-1$ Neuronen in der Einzelschicht. Überschneiden sich jedoch die Geraden in der Ebene, dann benötigt man sogar mehrere Schichten (siehe Mehrschicht-Perzeptron). Im folgenden Beispiel wird von einem Neuron eine Trenngerade gesucht, die eine Ebene in zwei Klassen teilt. Auf der Ebene sind zwei Arten von Objekten angesiedelt (hier 2 Kreise und 4 Quadrate), die durch die Gerade voneinander abgeteilt werden sollen. Es gibt die zwei Koordinaten x_1 und x_2 . Die folgende kleine Tabelle zeigt die Klasse und die Koordinaten, die Graphik zeigt die Ebene, die Objekte und den zulässigen Bereich für die Trenngerade (schraffiert).

Objekt	Klasse	x_1	x_2
Kreis	$d=1$	1	5
Kreis	$d=1$	5	4
Quadrat	$d=-1$	-2	0
Quadrat	$d=-1$	-1	0
Quadrat	$d=-1$	5	1
Quadrat	$d=-1$	10	5

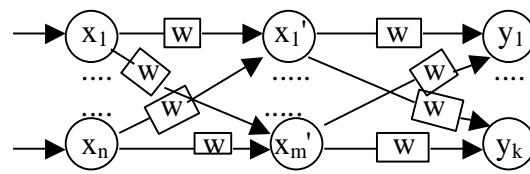


Der **Lernalgorithmus** ist eine **Optimierungsaufgabe**, bei der die **Summe der Fehlerquadrate** minimiert wird. Der Algorithmus könnte also auch Lösungen für solche Aufgaben finden, bei denen keine saubere Trennung der Objekte möglich ist. In diesem Falle würde der Algorithmus den *Klassifikationsfehler minimieren*. Lerndatenumfang ist hier $l=6$, **Lerndaten** sind unsere 6 Koordinatenpaare und die 6 Klassennummern $d=1$ bzw. $d=-1$. Viele Autoren benutzen noch einen **Lernfaktor** η , der die Konvergenz der Minimumsuche beschleunigen soll. Wenn man nicht weiß, welcher Wert von η optimal ist, nimmt man einfach $\eta=1$. Die folgende kleine Tabelle beschreibt den Minimierungsalgorithmus für die Fehlerquadratsumme.

Zeile	Beschreibung	Algorithmus
1	Startgewichte, Schwellenwert, Lernfaktor, j	$w_1 = w_2 = 1, \theta = 2, \eta = 1, j=1$
2	Nimm Zeile j der Lerndaten ($j=1, 2, \dots, l$)	$x_{1j} = -2, x_{2j} = 0, d_j = 1$
3	Bilde $\alpha = \sum w_i x_i - \theta$	$\alpha = 1 \cdot (-2) + 1 \cdot 0 - 2 = -4$
4	Bilde y wenn $\alpha < 0$, dann $y = -1$ wenn $\alpha = 0$, dann $y = 0$ wenn $\alpha > 0$, dann $y = +1$	$\alpha = -4 \longrightarrow y(\alpha) = -1$
5	Berechne den Fehler $\Delta = d - y(\alpha)$	$\Delta = 1 - (-1) = 2$
6	Aktualisiere $w_1 \longleftarrow w_1 + \eta \cdot \Delta \cdot x_1$ die Gewichte $w_2 \longleftarrow w_2 + \eta \cdot \Delta \cdot x_2$	$1 + 1 \cdot 2 \cdot (-2) \longrightarrow w_1 = -3$ $1 + 1 \cdot 2 \cdot 0 \longrightarrow w_2 = 1$
7	Wiederhole Zeile 2 bis 6 immer wieder für alle Zeilen der Lerndaten, bis $\sum \Delta_j^2 = \text{Minimum}$ ist	

5.6.2 Mehrschicht-Perzeptron

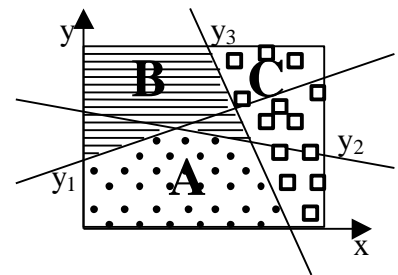
Das Mehrschicht-Perzeptron kann pro Ausgangssignal z.B. eine Gerade bilden (bei $n=2$ Eingangssignalen x_1, x_2) und logische Funktionen, wie AND, OR oder XOR realisieren, d.h. auch komplizierter geformte Klassen erkennen.



Die Zahl der Neuronen in jeder Schicht kann unterschiedlich sein. Dabei wird jedes Neuron der verborgenen ersten Schicht (hidden layer) x_1', x_2', \dots, x_m' über Gewichte w_{ij} mit allen n Eingangssignalen x_1, \dots, x_n und über andere Gewichte w_{jq} mit allen k Neuronen y_1, \dots, y_k der zweiten Schicht verbunden, so dass man insgesamt $n \cdot m + m \cdot k$ verschiedene Gewichte w benötigt. Deren Berechnung aus zufälligen Startgewichten ist eine anspruchsvolle Optimierungsaufgabe, die

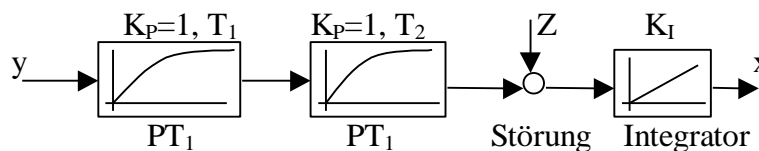
1. nicht immer das absolute Minimum findet
2. oft sehr viel Rechenzeit in Anspruch nimmt

Die Graphik rechts zeigt drei unterschiedlich geformte Objektklassen A, B, C, die durch 3 Geradengleichungen der Form $g_j = w_{1j} \cdot x_1 + w_{2j} \cdot x_2$ abgegrenzt werden. $g_1 = w_{11} \cdot x_1 + w_{21} \cdot x_2$ lässt sich umformen in die übliche Geradengleichung $x_2 = B_{10} + B_{11} \cdot x_1$ bzw. mit $y = x_2, x = x_1$ in $y_1 = B_{10} + B_{11} \cdot x$. Ebenso erhalten wir $y_2 = B_{20} + B_{21} \cdot x$ und $y_3 = B_{30} + B_{31} \cdot x$. Objekte von A finden wir mit $y < y_1$ UND $y < y_2$ UND $y < y_3$. Objekte von B finden wir mit $(y \geq y_1$ ODER $y \geq y_2)$ UND $y < y_3$. Objekte von C finden wir mit $y \geq y_3$.



5.6.3 Ein Beispiel aus der Regelungstechnik

Ein Roboterarm hebt 4 Sorten unterschiedlich schwerer Teile aus einem Behälter auf ein Band, und das möglichst ohne Überschwingen wegen der Stauch- und Kollisionsgefahr. Die Regelstrecke *Roboterarm* ist in diesem Beispiel hier eine IT_2 -Strecke mit dem Blockschaltbild



Die IT_2 -Strecke wurde für die 4 unterschiedlich schweren Teile P1, P2, P3, P4 jeweils identifiziert, d.h. die Streckenparameter T_1, T_2 und K_I bestimmt. Da die Integrierkonstante K_I nur vom Übersetzungsverhältnis des Motorgetriebes des Roboterarms abhängt, und somit konstant ist, ergaben sich lediglich unterschiedliche Verzögerungszeiten T_1 und T_2 , wie die nachfolgende kleine Tabelle zeigt.

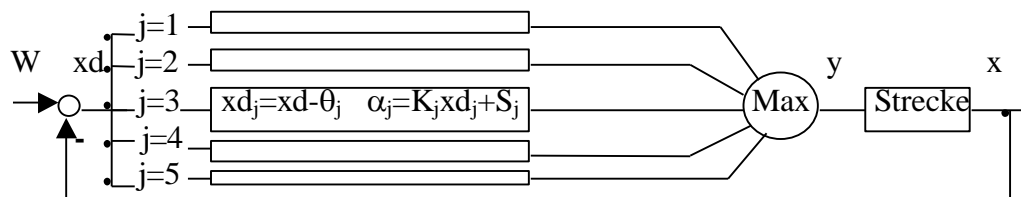
	P1	P2	P3	P4
T_1	0,10	0,220	0,700	1,44
T_2	0,19	0,135	0,105	0,10

I-Strecken regelt man mit PD-Reglern, wenn das Messsignal rauschfrei verfügbar ist. Das Messsignal ist hier die Rückmeldung über die Position des Teils. Für jedes unterschiedliche Teil müsste man eigentlich mit einem speziell auf dieses Teil abgestimmten PD-Regler arbeiten. Die optimalen Parameter (optimale Reglerverstärkung K_{PR} und optimale Vorhaltezeit T_V des D-Anteils) für die 4 Regelsituationen zeigt die folgende kleine Tabelle:

	P1	P2	P3	P4
K_{PR}	5,26	7,41	9,52	10,0
T_V	0,19	0,135	0,105	0,1

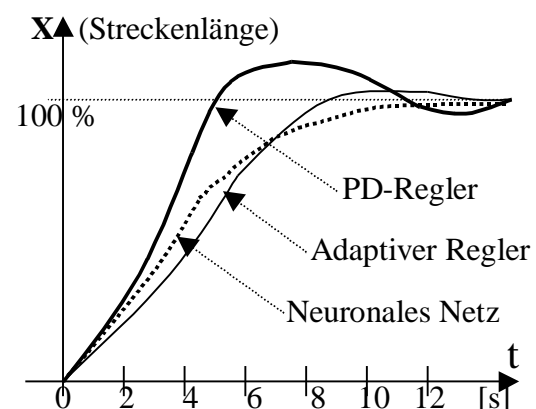
Das Problem bei dieser Aufgabe ist, dass der Regler nicht den Typ des Teils weiß, das am Robotergreifer hängt. Ist das Teil leicht, schwingt der Arm über und knallt das Teil aufs Band. Ist das Teil schwer, schafft der Roboter die Zeitvorgabe nicht.

Das Blockschaltbild der Neuronalen-Netz-Steuerung zeigt die folgende Graphik:



Sollwert W ist der Zielpunkt auf dem Band, d.h. ein ganz bestimmter Wert von x . Die Regeldifferenz $xd = W - x$ wird in 5 Signale aufgesplittet und es werden parallel 5 Alpha-Signale $\alpha_1, \alpha_2, \dots, \alpha_5$ berechnet. Dafür werden 2·5 Gewichte K_j und S_j und 5 Schwellenwerte θ_j benötigt, die in einer Lernphase vor dem ersten Einsatz des Roboters bestimmt werden müssen. Die Zahl 5 ist eher pragmatisch zu sehen. Es hätten auch 6 oder 4 sein können. Die Maximumbildung ist eine Anleihe aus der Fuzzy Logic und entspricht einer ODER-Verknüpfung der 5 Stellsignale α_j .

1. Für einen einfachen PD-Regler muss man einen Kompromiss eingehen. Die Weg-Zeitkurve des PD-Reglerkompromisses für ein leichtes Teil zeigt dann auch ein deutliches Überspringen.
2. Ein adaptiver Regler berechnet aus der Weg-Zeit-Kurve der ersten 0,1 Sekunden T_1 und T_2 und daraus ein angepasstes K_{PR} und T_V . Aber auch er schwingt noch leicht über.
3. Der von einem neuronalen Netz gesteuerte Roboterarm erkennt die Bewegungsdynamik und reagiert wie ein geschulter Kranfahrer



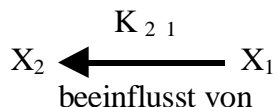
5.7 Mehrgrößenregelung

Für jede Regelgröße in einer Anlage, z.B. in einem Fermenter, muss eine eigene Stellgröße existieren. Die Temperatur regelt man mit Heizung/Kühlung, die O₂-Konzentration mit der Rührerdrehzahl, den pH-Wert mit der Drehzahl der Dosierpumpe, den Druck mit dem Hub des Ablassventils usw. Auf diese Weise existieren in einer Anlage mehrere Regelkreise, die im Normalfall nebeneinander arbeiten, ohne sich gegenseitig zu stören. Hier spricht man von Einzelregelkreisen,

Sind jedoch starke Wechselwirkungen (Kopplungen) zwischen den einzelnen Regelkreisen vorhanden, dann ist *Mehrgrößenregelung* angesagt. Die Mehrgrößenregelung befasst sich mit der Regelung stark gekoppelter Regelkreise.

Beispiel: Heizen erniedrigt die O₂-Konzentration im Fermenter, da warmes Wasser weniger Sauerstoff lösen kann, als kaltes. Erhöht man jetzt die Rührerdrehzahl, wird zwar mehr Sauerstoff eingetragen, aber auch zusätzliche Reibungswärme. Etwa 90% der Rührermotorleistung bleiben zuerst einmal als Wärme im Fermenter. (Ob die Kopplung in diesem Beispiel wirklich so stark ist, dass man Mehrgrößenregelung anwenden muss, wird im Einzelfall entschieden.)

Ein Koppelmaß K_{ij} gibt an, wie stark eine Änderung der Regelgröße X_j die Regelgröße X_i beeinflusst. Dabei gilt, dass allgemein $K_{ij} \neq K_{ji}$ ist.

$K_{ij} = \frac{\Delta X_i}{\Delta X_j} \frac{D_j}{D_i}$	
--	--

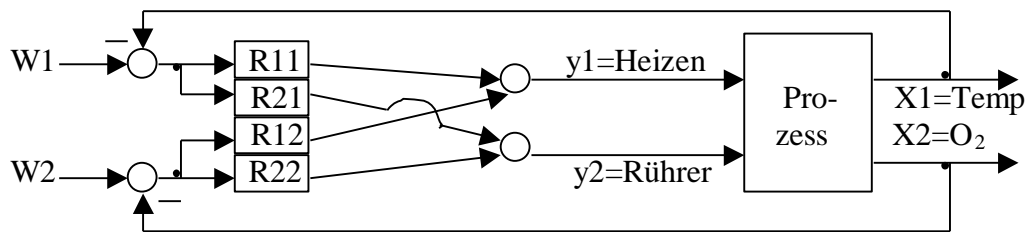
ΔX_i und ΔX_j sind Änderungen der Regelgrößen X_i und X_j nach einem aufgegebenen y_j -Sprung. Eigentlich sollte durch diesen Stellgrößensprung $U_{0j} = \Delta y_j$ nur die Regelgröße X_j beeinflusst werden, aber durch die Kopplung ändert sich Regelgröße X_i ungewollt mit. D_i und D_j sind die Toleranzen für die Regelgrößen X_i und X_j . Man nimmt die gesamte Breite des Toleranzbandes. Z.B. sei bei einer Temperaturregelung die zulässige Toleranz $\pm 2^\circ\text{C}$, dann ist $D = 4^\circ\text{C}$. Ab einem Koppelmaß $K_{ij} > 0.5$ spricht man von starker Kopplung und sollte eine Mehrgrößenregelung in Erwägung ziehen.

Es gibt zahlreiche Verfahren der Mehrgrößenregelung, z.B.

- Stationäre Entkopplung
- Dynamische Entkopplung
- Fuzzy Control
- Beobachtermodelle usw.

Die *stationäre Entkopplung* benutzt für stark gekoppelte Regelkreise zusätzliche Entkopplungsregler. Wie der Name sagt, funktioniert die Entkopplung am besten im stationären Zustand (wenig Störungen, wenig Sollwertänderungen). Bei Sollwertänderungen oder starken Störungen können kurzzeitig Schwingungen oder ein Überspringen auftreten. Ein gutes Beispiel ist die stationäre Entkopplung nach Boksenboom und Hood. Die nachfolgende Graphik zeigt zwei Ein-

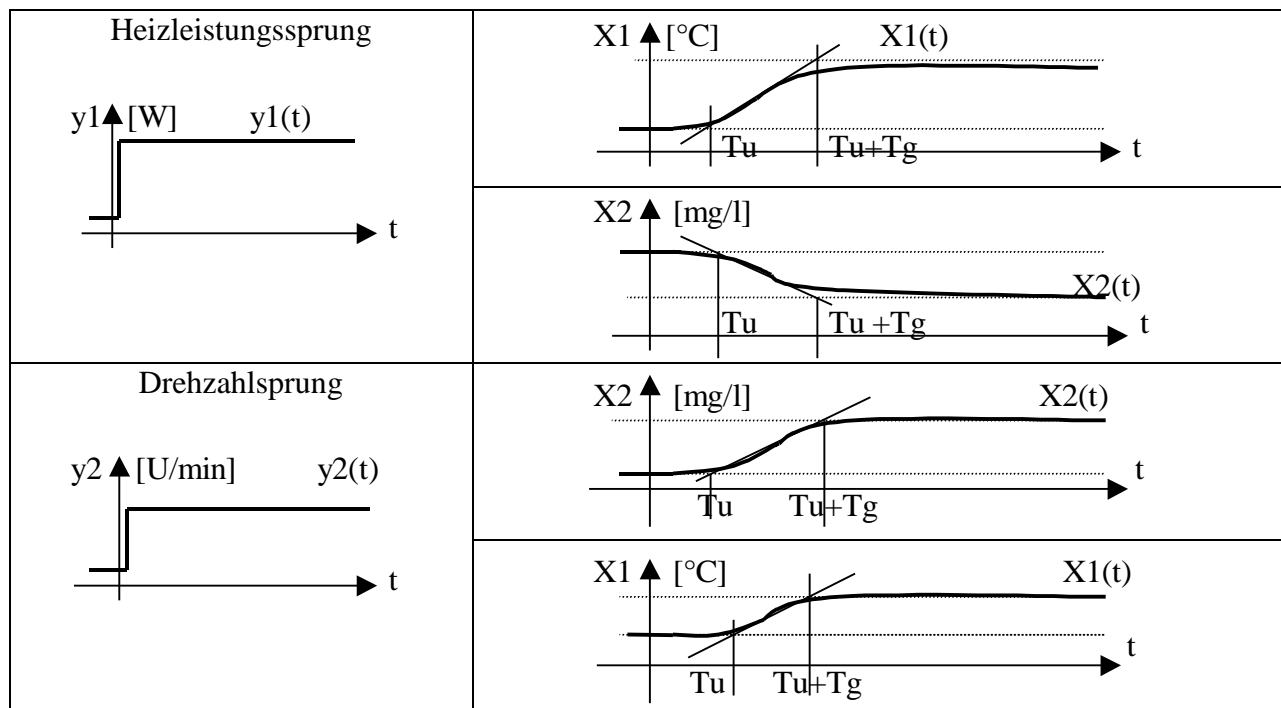
zelregelkreise für Temperatur und O_2 -Konzentration, die durch zwei Zusatzregler entkoppelt werden.



$W1, W2$ sind die Sollwerte für Temperatur und O_2 -Konzentration
 $X1, X2$ sind die Istwerte für Temperatur und O_2 -Konzentration
 $y1$ ist die Stellgröße für die Heizung (Heizleistung)
 $y2$ ist die Stellgröße für die O_2 -Konzentration (Rührerdrehzahl)
 $R11, R22$ sind die Hauptregler des Temperatur- und des O_2 -Regelkreises
 $R12, R21$ sind die beiden Entkopplungsregler

Die Berechnung der Haupt- und Entkopplungsregler erfolgt in 4 Schritten:

Schritt 1: Ausführung von zwei Sprungversuchen und Registrierung der 4 Sprungantworten und daraus die Bestimmung der K_{PS} -Matrix der Streckenverstärkungen.



Die K_{PS} -Matrix wird aus den 4 Verstärkungen der 4 Sprungantworten gebildet und sei

$$K_{PS} = \begin{bmatrix} 0,09 \left[\frac{K}{W} \right] & 0,008 \left[\frac{K}{U / \min} \right] \\ -0,0045 \left[\frac{mg / l}{W} \right] & 0,170 \left[\frac{mg / l}{U / \min} \right] \end{bmatrix}.$$

Schritt 2: Auslegung von P-Reglern für die beiden Einzelregelkreise so, als wäre keine Kopplung vorhanden (z.B. nach der Reglerparametrierung von Chien, Hrones und Reswick aus Kap. 5.3.3). Die beiden berechneten Verstärkungen K_{P11} und K_{P22} werden als Diagonalmatrix D_R geschrieben. Diese sei

$$D_R = \begin{bmatrix} 1,7 \left[\frac{W}{K} \right] & 0 \\ 0 & 4,1 \left[\frac{U / \min}{mg / l} \right] \end{bmatrix}$$

Schritt 3: Berechne die Reglermatrix nach der Matrix-Formel $R = K_{PS}^{-1} D_R D_K$.

Hierbei ist K_{PS}^{-1} die inverse Matrix zu K_{PS} . Man erhält sie mit EXCEL oder einem geeigneten Taschenrechner. D_K ist die Diagonale von K_{PS} als Diagonalmatrix geschrieben. Die numerische Berechnung erfordert zwei Matrixmultiplikationen und ist

$$R = \begin{pmatrix} 11,085 & -0,5216 \\ 0,2934 & 5,8685 \end{pmatrix} \cdot \begin{pmatrix} 1,7 & 0 \\ 0 & 4,1 \end{pmatrix} \cdot \begin{pmatrix} 0,09 & 0 \\ 0 & 0,17 \end{pmatrix}$$

bzw.

$$R = \begin{pmatrix} 1,70 \left[\frac{W}{K} \right] & -0,36 \left[\frac{W}{mg / l} \right] \\ 0,04 \left[\frac{U / \min}{K} \right] & 4,1 \left[\frac{U / \min}{mg / l} \right] \end{pmatrix}$$

Wie man sieht, haben sich die Hauptregler R11 und R22 nicht verändert, aber es wurden zwei Entkopplungsregler berechnet, von denen R12 eine negative Verstärkung aufweist, d.h. der Ausgang dieses Reglers muss subtrahiert werden, nicht addiert.

Schritt 4: Ersetze die 4 P-Regler $R11=1.70$, $R12=-0.36$, $R21=0.04$ und $R22=4.1$ durch PI- bzw. PID-Regler. Ein Beispiel soll die Vorgehensweise demonstrieren. Die Reglertabelle von Chien, Hrones und Reswick (siehe Kap. 5.3.3) hat die zwei Spalten "Führungs-" bzw. "Konstantregelung", und diese wieder unterteilt in je zwei Spalten "0% Überschwingen" und "20% Überschwingen". Wir benutzen die Spalte "Führungsregelung mit 0% Überschwingen".

Laut Tabelle gilt für P-Regler $V_0 T_u / T_g = 0,3$

Laut Tabelle gilt für PI-Regler $V_0 T_u / T_g = 0,6$ und $T_n = 4 T_u$

Für den Hauptregler $R11 = 1.70 [W/K]$ sei aus der $y1$ -Sprungantwort der Temperatur (siehe Graphik oben) eine Verzugszeit $T_{u11} = 7 [s]$ bestimmt worden. Wir berechnen

$$K_{p11} = \frac{R11}{0,3} \cdot 0,6 = \frac{1,70}{0,3} \cdot 0,6 = 3,40 \left[\frac{W}{K} \right] \quad \text{Reglerverstärkung}$$

$$T_{n11} = 4 \cdot T_{u11} = 28 [s] \quad \text{Nachstellzeit des I-Anteils}$$

Für einen der Koppelregler, z.B. $R12 = -0.36 [W/(mg/l)]$, machen wir eine ähnliche Rechnung. Hier wird die Verzugszeit aus der Temperatur-Sprungantwort zum Rührerdrehzahlsprung $y_2(t)$ abgelesen (siehe Graphik oben) und sei $T_{u12} = 9 [s]$.

$$K_{p12} = \frac{R12}{0,3} \cdot 0,6 = \frac{-0,36}{0,3} \cdot 0,6 = -0,72 \left[\frac{W}{mg/l} \right] \quad \text{Reglerverstärkung}$$

$$T_{n12} = 4 \cdot T_{u12} = 36 [s] \quad \text{Nachstellzeit des I-Anteils}$$

Ist ein Koppelkoeffizient $K_{ij} < 0.5$, dann kann auf den Entkopplungsregler R_{ij} verzichtet werden. Bei einer Softwarelösung ist die Einsparung unerheblich. Werden jedoch Hardware-Regler benutzt, dann lohnt sich diese Überlegung.

5.8 Stabilität und Güte von Regelkreisen

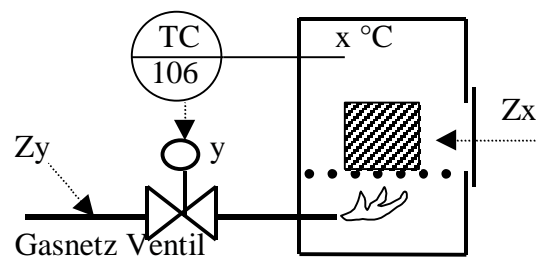
Stabilität in der Regelung heißt, dass keine Dauerschwingungen auftreten, und auch kein Aufschaukeln von Schwingungen möglich ist.

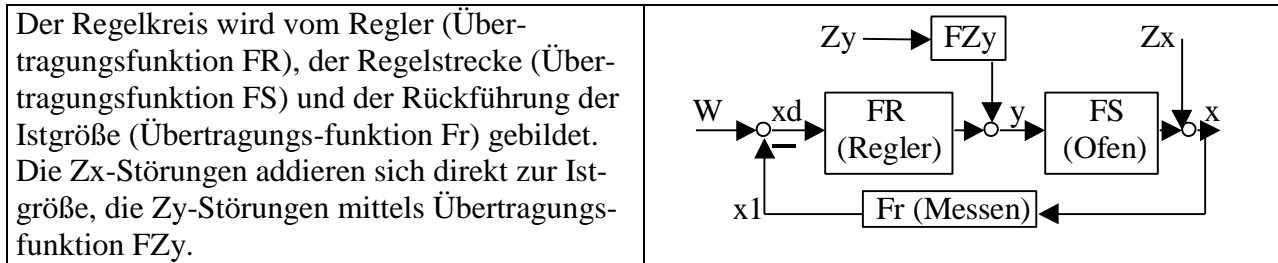
Güte einer Regelung heißt Einhalten des Toleranzbandes für den Istwert, und ein schneller Übergang ohne Schwingungen bei Sollwertänderungen.

Stabilität und Güte widersprechen sich teilweise, zumindest was Geschwindigkeit und Schwingungsfreiheit anbetrifft. Der Ingenieur muss hier einen Kompromiss suchen.

5.8.1 Beispiel Temperaturregelung eines Industrieofens

Die Graphik zeigt das RI-Fließbild des Ofens: Im Inneren herrscht die Temperatur $x \text{ } ^\circ\text{C}$. Der Regler stellt ein Gasventil, um die Ist-Temperatur auf dem Sollwert zu halten. Stellgröße ist der Hub y am Gasventil. Störungen Z_x können durch die geöffnete Ofentür und kaltes Material entstehen, Störungen Z_y durch Druckschwankungen im Gasnetz.





W	Sollwert, z.B. 1000 °C Schmiedetemperatur für Stahlrohlinge
x	tatsächliche Temperatur im Ofen
x1	Istwert nach dem Messvorgang, z.B. mit einem Thermoelement NiCrNi gemessen
$x_d = W - x_1$	Regeldifferenz Sollwert minus Istwert
y	Stellgröße, z.B. der Hub des Gasventils in % von 100
Zx	Störgröße, z.B. die Temperaturabsenkung in °C durch eine offene Ofentür oder einen kalten Stahlblock
Zy	Störgröße, z.B. eine Druckminderung im Gasnetz. Z.B. könnten sich 10% Minderdruck wie 15% weniger Hub am Gasventil auswirken. Deshalb rechnet die Übertragungsfunktion FZy Druckschwankungen [mbar] in y-Werte [% Hub] um.
FR	Übertragungsfunktion des Reglers, z.B. eines PID-Reglers
FS	Übertragungsfunktion der Regelstrecke einschließlich Stellantrieb (hier der Ofen und das Regelventil), modelliert als PT ₂ -Strecke
Fr	Übertragungsfunktion der Rückführung, z.B. die Verzögerung der Temperaturmessung durch die Keramikummantelung des Thermoelements modelliert als PT ₁ -Strecke

<p>Zur Erinnerung: Die Übertragungsfunktion eines Gliedes ist der Quotient von Ausgangssignal x und Eingangssignal y.</p>	$F(p) = \frac{x(p)}{y(p)}$	
---	----------------------------	--

An dieser Stelle soll aus dem Blockschaltbild die **Kreisgleichung** abgeleitet werden. Liegen Übertragungsglieder in Reihe, dann multiplizieren sich die Übertragungsfunktionen. Wir arbeiten im Frequenz-bereich, d.h. wir schreiben $x(p)$ statt $x(t)$, $W(p)$ statt $W(t)$ usw. Laut obigem Blockschaltbild ist

$$x(p) = Zx(p) + FZy(p) \cdot FS(p) \cdot Zy(p) + FR(p) \cdot FS(p) \cdot x_d(p) .$$

Das Signal Zy muss erst durch die Box FZy und dann durch die Box FS hindurch, bevor es zu einem Bestandteil von x wird. Das Signal xd muss erst durch die Box FR und dann durch die Box FS hindurch, bevor es zu einem Bestandteil von x wird. Lediglich die Störung Zx ist direkter Bestandteil von x. Wir ersetzen $x_d(p)$ durch $W(p) - x_1(p)$ und $x_1(p)$ durch $Fr(p) \cdot x(p)$, da die reale Temperatur x durch die Box Fr hindurch muss, um zur gemessenen Istgröße x1 zu werden, und wir erhalten die Gleichung

$$x(p) = Zx(p) + FZy(p) \cdot FS(p) \cdot Zy(p) + FR(p) \cdot FS(p) \cdot [W(p) - Fr(p) \cdot x(p)] .$$

Wir lösen die eckige Klammer auf, bringen den Term $FR(p) \cdot FS(p) \cdot Fr(p) \cdot x(p)$ zum $x(p)$ auf die linke Seite und klammern dort $x(p)$ aus, und erhalten so

$$x(p) \cdot [1 + FR(p) \cdot FS(p) \cdot Fr(p)] = Zx(p) + FZy(p) \cdot FS(p) \cdot Zy(p) + FR(p) \cdot FS(p) \cdot W(p)$$

Nun dividieren wir beide Seiten der Gleichung durch die Klammer $[1 + FR(p) \cdot FS(p) \cdot Fr(p)]$ und erhalten die Kreisgleichung für unseren gestörten Regelkreis

$$x(p) = \frac{FR(p)FS(p)}{1 + FR(p)FS(p)Fr(p)} W(p) + \frac{FZy(p)FS(p)}{1 + FR(p)FS(p)Fr(p)} Zy(p) + \frac{Zx(p)}{1 + FR(p)FS(p)Fr(p)}$$

Die 3 Terme der rechten Seite beschreiben die Einflüsse des Sollwertverlaufs und die Einflüsse von Zy - und Zx -Störungen auf die Temperatur x im Ofen.

5.8.2 Zahlenbeispiel für den Einfluss von Sollwertänderungen

PID-Regler mit $K_{PR} = 10,0$ [% / K] Reglerverstärkung in Hub pro Grad Abweichung
 $T_N = 17,5$ [s] Nachstellzeit des I-Anteils des PID-Reglers
 $T_V = 3,5$ [s] Vorhaltezeit des D-Anteils des PID-Reglers
 und der Übertragungsfunktion

$$FR(p) = K_{PR} \left[1 + \frac{1}{T_N p} + T_V p \right] = 10,0 \left[1 + \frac{1}{17,5 p} + 3,5 p \right]$$

Ofen mit PT_2 -Verhalten $K_{PS} = 16,2$ [K / %] Streckenverstärkung in Grad pro Ventilhub
 $T = 8,5$ [s] Zeitkonstante eines Verzögerungsgliedes
 und der Übertragungsfunktion

$$FS(p) = K_{PS} \left(\frac{1}{1 + pT} \right)^n = 16,2 \left(\frac{1}{1 + 8,2 \cdot p} \right)^2$$

Messung mit PT_1 -Verhalten $K_P = 1,0$ [K/K] Verstärkungsfaktor
 $T = 2,7$ [s] Zeitkonstante der Messverzögerung
 und der Übertragungsfunktion

$$Fr(p) = \frac{K_P}{1 + Tp} = \frac{1}{1 + 2,7 p}$$

Damit ist die Übertragungsfunktion $F(p)$ des Kreises bezüglich einer Sollwertänderung auch numerisch berechenbar:

$$x(p) = \frac{FR(p)FS(p)}{1 + FR(p)FS(p)Fr(p)} W(p) = F(p) W(p) .$$

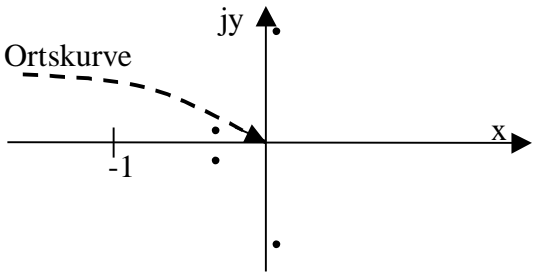
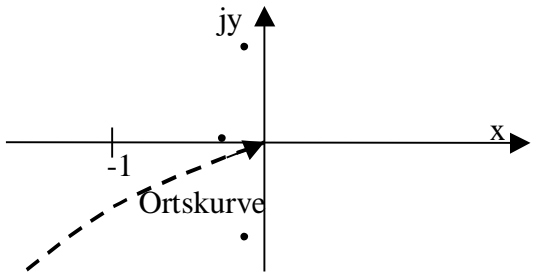
5.8.3 Zwei gleichwertige Stabilitätskriterien

Für die Beurteilung der Stabilität eines Regelkreises ist vor allen Dingen der Nenner der Übertragungsfunktion, $1 + FR(p) \cdot FS(p) \cdot Fr(p)$, wichtig. Das Produkt $F_0(p) = FR(p) \cdot FS(p) \cdot Fr(p)$ heißt *Übertragungsfunktion des aufgeschnittenen Regelkreises* und spielt eine wichtige Rolle. $F_0(p)$

beschreibt die Reihenschaltung von Regler, Regelstrecke und Rückführung (siehe Blockschaltbild oben).

Das **Wurzelort-Kriterium** der Stabilität (Root Locus Criterion): Sind die Realteile aller Wurzeln (Nullstellen) der Gleichung $1+F_0(p)$ negativ, dann treten nur gedämpfte Schwingungen auf, und die Regelung ist stabil.

Das **Ortskurven-Kriterium** (auch Nyquist- oder Routh-Hurwitz Criterion) : Lässt die Ortskurve $F_0(j\omega)$ für $\omega \rightarrow \infty$ den Punkt "-1" auf der reellen Achse "links liegen", dann ist keine Selbstverstärkung möglich und die Regelung ist stabil.

<p>Eine Lösung mit den PID-Parametern $K_{PR}=30$, $T_N=8$ und $T_V=1,5$ zeigt zwei konjugiert komplexe Wurzeln im positiven Bereich bei $0,03 \pm 1,93j$ und zwei konjugiert komplexe Wurzeln im negativen Bereich bei $-0,33 \pm 0,15j$. Die Ortskurve lässt den Punkt "-1" rechts liegen. Beide Kriterien sagen instabil.</p>	<p>Eine Lösung mit den PID-Parametern $K_{PR}=10$, $T_N=17,5$ und $T_V=3,5$ zeigt zwei konjugiert komplexe Wurzeln im negativen Bereich bei $0,16 \pm 1,70j$ und eine reelle Wurzel im negativen Bereich bei $-0,21$. Die Ortskurve lässt den Punkt "-1" links liegen. Beide Kriterien sagen stabil.</p>
	

Wie berechnet man die Wurzeln (Nullstellen) einer komplexen Funktion $F(p)$? Systeme wie Mathcad oder Maple bieten da Lösungen an. Wer es selbst einmal versuchen will, geht z.B. folgendermaßen vor. Man nimmt den Bildschirm als Gaußsche komplexe Zahlenebene und steckt den Zahlenbereich für das komplexe Argument $p=\delta+\omega j$ ab, z.B. $-3 \leq \delta \leq 3$ und $-3 \leq \omega \leq 3$. Auf diese Weise kann man jedem Pixel einen komplexen Wert zuordnen. Den setzt man in die komplexe Funktion $F(p)$ ein, berechnet deren Wert $z = x+yj$. Dann bildet man den Betrag

$$|F(p)| = |z| = \sqrt{x^2 + y^2}.$$

Den Betrag kann man jetzt zur Färbung in einer Grauskala desjenigen Pixels benutzen, zu dem das p gehört. Kleine Beträge färbt man dunkel, denn das sind die Pixel auf oder direkt neben einer Nullstelle. Je größer der Betrag, desto heller färbt man die Pixel.

Wie berechnet man die Ortskurve einer Übertragungsfunktion? Die Ortskurve einer komplexen Funktion $F(p)$ entsteht, wenn man nacheinander p -Werte in die Funktion einsetzt, die auf der imaginären Achse der komplexen Zahlenebene liegen, d.h. $p=0+\omega j$. Lässt man z.B. ω Werte von 0,0001 bis 10,0 in Schritten von 0,0001 durchlaufen, entstehen 100.000 komplexe $F(p)$ -Werte $z = x+yj$. Die Kurve wird sichtbar, wenn man die Pixel, auf die z zeigt, schwarz färbt.

5.8.4 Gütemaße der Regelung, Optimierung

Einziges Gütemaß im eingeschwungenen Zustand ist die **Regelabweichung** $x_d = W - x$. Diese tritt aber nur bei Reglern ohne I-Anteil auf, z.B. bei einem P-Regler.

Gütemaße für den Einschwingvorgang gibt es reichlich. Schnellen Reglern gestattet man, dass sie bis zu 20% überschwingen, bezogen auf die Höhe eines Sollwertsprungs. Regler ohne Überschwingen arbeiten langsamer.

Bei schnellen Reglern minimiert man eines der folgenden Gütemaße T_{an} , I oder I^2 :

T_{an} = Anregelzeit (Zeit bis das neue Toleranzband $W_{neu} \pm Tol$ erreicht ist)

$I = \int_0^{TM} |x_d(t)| dt$ (Regelfläche absolut innerhalb einer festgelegten Messzeit TM)

$I^2 = \int_0^{TM} x_d^2(t) dt$ (Quadratische Regelfläche innerhalb einer festgelegten Messzeit TM)

unter Einhaltung einer der folgenden Bedingungen

$\delta \geq 0,45$ (Dimensionslose Dämpfung, $\delta = 0,45$ gibt gerade 20% Überschwingen)

$X_m \leq 20\%$ (Überschwingweite bezogen auf die Höhe des Sollwertsprungs)

Bei langsamen Reglern ohne Überschwingen minimiert man eines der folgenden Gütemaße:

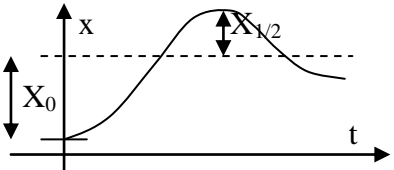
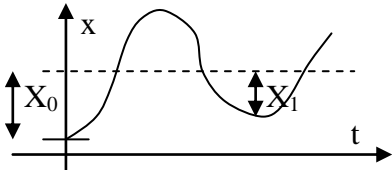
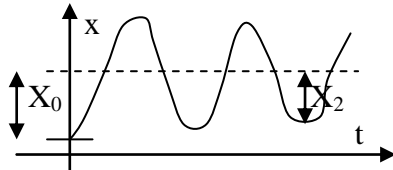
T_{aus} = Ausregelzeit (ab T_{aus} liegt der Istwert nur noch im Toleranzband $W_{neu} \pm Tol$ um den neuen Sollwert)

$It = \int_0^{TM} t \cdot |x_d(t)| dt$ (Zeitgewichtete absolute Regelfläche innerhalb einer festgelegten Messzeit TM. Spätere Abweichungen werden hier härter bestraft.)

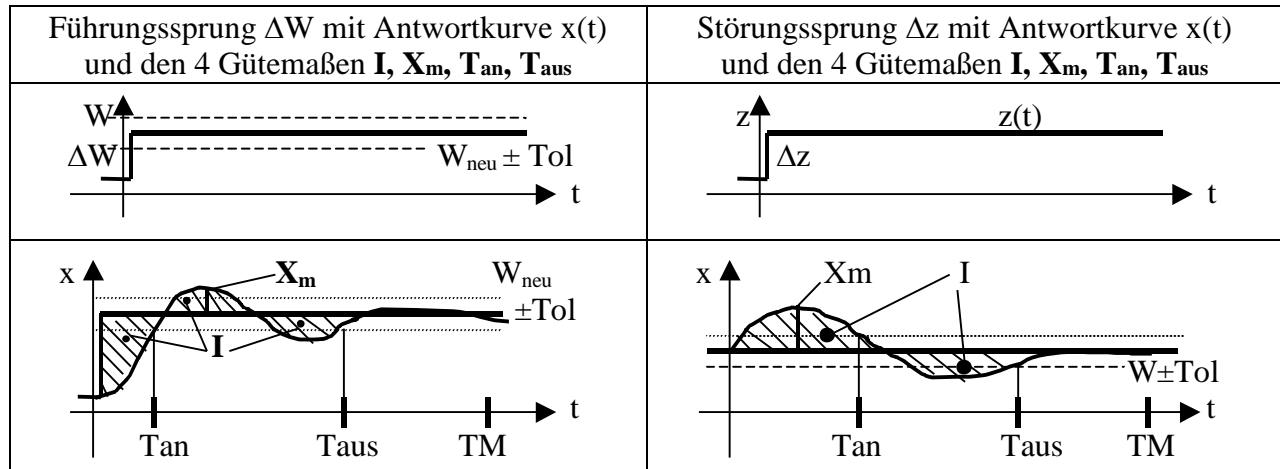
unter Einhaltung von $X_m \leq Tol$ (Kein Überschwingen über das Toleranzband hinaus).

Die dimensionslose Dämpfung δ der Schwingung ist $\delta = \frac{\lambda}{\sqrt{4\pi^2 + \lambda^2}}$.

Hier ist λ das logarithmische Dekrement, d.h., die logarithmische Abnahme der Schwingungsamplituden X .

Bei großer Dämpfung $\lambda = 2 \ln X_0 / X_{1/2} $	Bei mittlerer Dämpfung $\lambda = \ln X_0 / X_1 $	Bei kleiner Dämpfung $\lambda = 0,5 \cdot \ln X_0 / X_2 $
		

Die folgenden 4 kleinen Graphiken zeigen einen Sollwertsprung ΔW (Führungssprung) mit der Antwortkurve $x(t)$ des Regelkreises und eingezeichneten Gütemaßen, sowie einen Störungssprung Δz (z.B. in der Sommerhitze ein Laborfenster aufgerissen) mit der Antwortkurve und eingezeichneten Gütemaßen.



Optimierung

Optimierung heißt entweder Verbesserung der Regeleigenschaften oder Erhöhung der Stabilität.

Man optimiert **am laufenden Regelkreis** durch vorsichtige Veränderung der Regelparameter und Messung eines der obigen Gütemaße. Beim PID-Regler und zu starkem Überschwingen verringert man die Verstärkung K_{PR} und vergrößert parallel dazu die Nachstellzeit T_N . Beim PD-Regler und zu starkem Überschwingen verringert man nur die Verstärkung K_{PR} . Schleicht der Regler sich zu langsam an den neuen Sollwert heran oder braucht er zu lange, um eine Störung wegzuregeln, dann erhöht man K_{PR} und verringert parallel dazu die Nachstellzeit T_N .

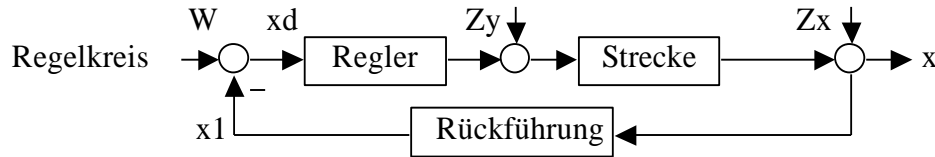
PID-Regler nimmt man für P- und PTn-Strecken bei glattem Messwertverlauf. PI-Regler nimmt man statt PID, wenn das Messsignal gestört ist. In diesem Falle darf nicht differenziert werden.

PD-Regler nimmt man für I- und ITn-Strecken bei glattem Messwertverlauf. Den reinen P-Regler für diese Strecken bei gestörten Messwerten. Bei der Vorhaltezeit T_v sollte man sich ungefähr am Richtwert $0,25 \Sigma T_i$ orientieren, wobei ΣT_i die Zeitkonstantensumme der Regelstrecke ITn sind. P- und PD-Regler sollte man immer im Arbeitspunkt betreiben. Das ist der Wert y_0 der Stellgröße, bei dem der Prozess auch ohne Regler lief, vorausgesetzt, dass keine Störungen und keine Sollwertänderungen anliegen.

Man kann einen Regelkreis auch **am PC optimieren**, wenn man alle Glieder des Regelkreises durch ein Modell beschreiben kann, z.B. die Regelstrecke einschließlich des Aktors durch ein PT₂-Glieder, die Rückführung (Sensor, Messumformer) durch ein PT₁-Glieder. Natürlich müssen die echten Glieder dafür identifiziert werden. Alle Verstärkungen und Zeitkonstanten müssen bekannt sein. Man nimmt dann ein Simulationspaket für Regelschaltungen, z.B. IAS oder LabView, programmiert das Regelkreismodell durch Aneinanderklicken der Bausteine, parametriert sie mit den identifizierten Verstärkungen und Zeitkonstanten, und kann dann in aller Ruhe den geeigneten

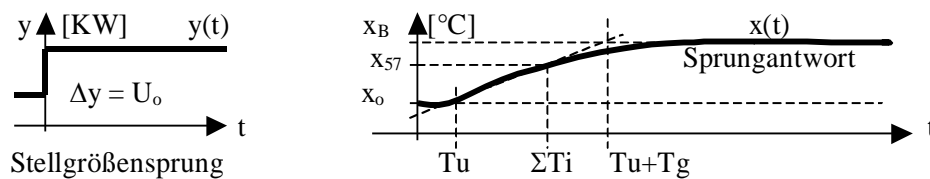
ten Regler ausprobieren und die Antwortkurven des Regelkreises optimieren, indem man die Reglerparameter geeignet wählt.

Beispiel Temperaturregelung



Blockschaltbild des Regelkreises

Um Typ und Parameter der Strecke zu bestimmen, macht man mit der vom Regelkreis abgekoppelten Regelstrecke einen Sprungversuch:



Die aufgezeichnete Sprungantwort ist typisch für PTn-Strecken, was man durch Vergleich mit den Sprungantworten anderer Streckentypen im Kapitel 5.2.2 erkennt. Das PTn-Modell hat 3 Parameter: K_{PS} , n , T (Verstärkung, Speicherzahl, Zeitkonstante eines einzelnen Speichers). Die DGL z. B. des nicht schwingenden PT2-Gliedes als Beispiel eines PTn-Gliedes ist

$$T^2 \ddot{x}(t) + 2T\dot{x}(t) + x(t) = K_{PS} y(t)$$

Die komplexe Übertragungsfunktion ist
$$F(p) = K_{PS} \frac{1}{(1 + Tp)^n}$$

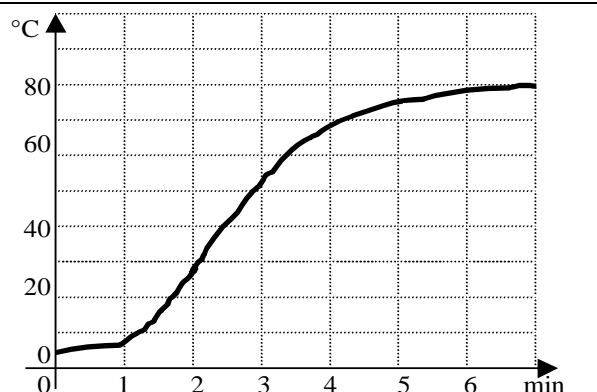
Die Sprungantwort des Modells auf einen Eingangssprung der Höhe $\Delta y = U_0$ ist

$$x(t) = x_0 + K_{PS} \cdot U_0 \cdot (1 - e^{-t/T})^n$$

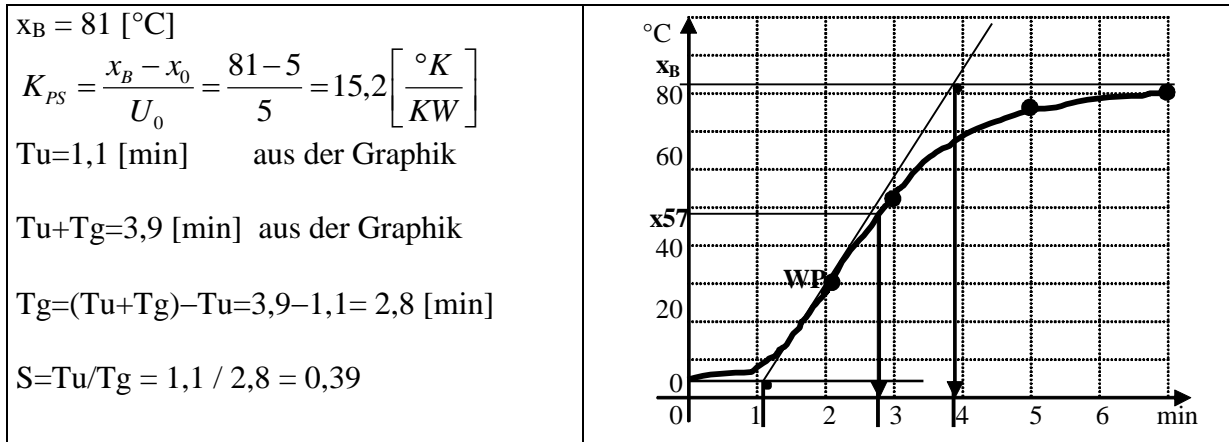
Die Identifikation des PTn-Modells erfordert die Bestimmung von K_{PS} , n und T .

Wir **identifizieren** die rechts stehende Antwortkurve auf einen Stellgrößensprung der Höhe $U_0 = 5$ KW Heizleistung (Typ und alle Parameter der Strecke):

1. Der Typ ist PTn (durch Vergleich)
2. Wir zeichnen Wendepunkt WP ein
3. Wir schätzen den Beharrungswert x_B
4. Wir berechnen Gesamtverstärkung K_{PS}
5. Wir zeichnen die Tangente und bestimmen T_u , $T_u + T_g$ und damit T_g



6. Wir berechnen Schwierigkeitsgrad S und daraus Speicherzahl n des Modells
7. Wir berechnen die Hilfsgröße x_{57} und bestimmen damit die Zeitkonstantensumme ΣT_i .
8. Wir bestimmen T nach der Formel $T = \Sigma T_i / n$.



$$n = [10 \cdot S + 1]_{\text{runden}} = [4,9]_{\text{runden}} = 5 \quad (\text{unser Streckenmodell sollte 5 Speicher haben})$$

$$x_{57} = x_0 + (x_B - x_0) \cdot 0,575 = 5 + (81 - 5) \cdot 0,575 = 48,7 [^{\circ}\text{C}]$$

Wir finden mit dem Schnittpunkt x_{57} und Kurve $x(t)$ den Wert $\Sigma T_i = 2,8 [\text{min}]$ für die Zeitkonstantensumme. Daraus berechnen wir die Zeitkonstante eines einzelnen Speichers:

$$T = \Sigma T_i / n = 2,8 / 5 = 0,56 [\text{min}].$$

Wir wollen einen einzelnen Wert einer Sprungantwort dieser Strecke berechnen für eine Eingangssprunghöhe $U_0 = 3 [\text{KW}]$, Starttemperatur $x_0 = 16 [^{\circ}\text{C}]$ und für Zeitpunkt $t = 2 [\text{min}]$:

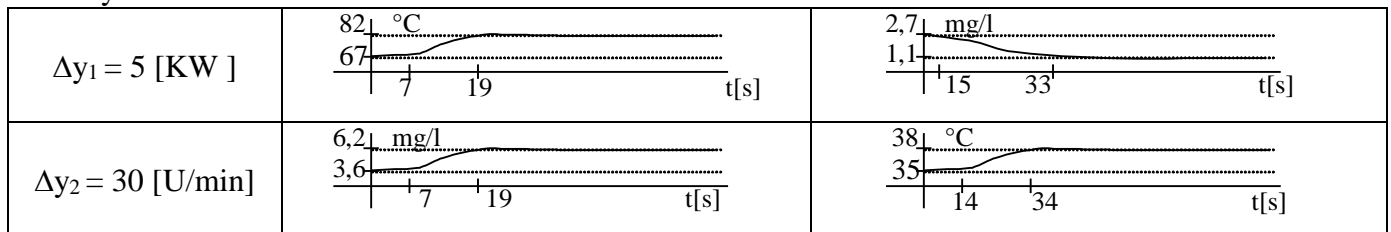
$$x(t) = x_0 + K_{PS} \cdot U_0 \cdot (1 - e^{-t/T})^n = 16 + 3 \cdot 15,2 \cdot (1 - e^{-2/0,56})^5 = 55,436 [^{\circ}\text{C}]$$

Mit dem so an den realen Prozess angepassten Modell unserer Regelstrecke lassen sich im PC Simulationsrechnungen durchführen, um Regel- und Steueraufgaben optimieren zu können.

6. Alte Klausuren

AT-Klausur SS 2012

1. (15 P) Berechnen Sie nach dem Beispiel von Übung 21 in 3 Schritten die Reglermatrix R (Schritt 4 nicht rechnen). Gegeben sind die 4 Sprungantworten zu zwei Stellgrößensprünge Δy_1 und Δy_2 :



Schritt 1: Kps-Matrix berechnen und Dimensionen angeben

Schritt 2: Die beiden Haupt-P-Regler D_{11} und D_{22} mit $V_o T_u / T_g = 0.6$ berechnen

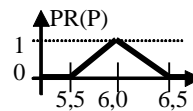
Schritt 3: Die inverse Kps-Matrix berechnen und daraus Reglermatrix R der P-Regler

2. (15 P) Entwerfen Sie den Mealy-Graphen mit den Schritten 0 bis 9 für folgenden Prozess:

(0) Warten, (1) Befüllen des Reaktionsgefäßes mit Leinenabfällen bis der Arbeiter Signalknopf "voll" drückt, (2) Befüllen mit Sulfonsäure über Ventil V1 bis Pegelmelder P1, (3) Befüllen mit Salzsäure über Ventil V2 bis Pegelmelder P2, (4) 12 Minuten rühren (Rü) und Reaktionszeit abwarten (Timer T1) bei gleichzeitiger Kühlung (Ventil V3) über den Mantel, (5) Ablassen der Flüssigkeit über Ventil V4 (Pegelmelder P0 abfragen), (6) Wasserzulauf über Ventil V5 bis Pegel P2, (7) 15 Minuten einweichen lassen unter Rühren (Timer T2), (8) Ablassen des Waschwassers über Ventil V6 (Pegelmelder P0 abfragen), (9) Leuchtsignal "Leeren" für den Arbeiter als Hinweis zum Entleeren, zurück ins Warten wenn von diesem Signalknopf "leer" gedrückt wird.

3. (8 P) Zeichnen Sie das Symbol eines 5/2-Steuerventils mit Stoßmagnet und Rückholfeder und einen doppelt wirkenden Zylinder, der ungedrosselt ausgefahren wird und nach dem Ende des SPS-Signals am Stoßmagnet gedrosselt eingeholt wird (Drossel mit Rückschlagventil in eine der Luftleitungen einzeichnen). Schreiben Sie die Anschlussnummern 1, 2, 3, 4, 5 richtig an das 5/2-Ventil.

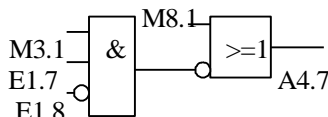
4. (10 P) In einer Fuzzy-Regelung soll eine Zuordnungsfunktion mit Nahmen $PR(P)$ in Form einer Auswahlklammer geschrieben werden (siehe ganz rechts). Die Daten entnehmen Sie der Graphik.



$$PR(P) = \begin{cases} \dots, & \text{wenn} \\ \dots, & \text{wenn} \\ \dots & \text{sonst} \end{cases}$$

5. (10 P) Schreiben Sie das AWL-Programm zu folgenden beiden Aufgabenstellungen. In der Aufgabe a) nehmen Sie als Zwischenmerker M25.0 als Ausgang der UND-Box.

a) Stelle auf M20.1 einen Einsmerker her. Realisiere gleich danach das folgende Schaltbild:



b) Schreiben Sie den Aufruf von FB250 (Analogeingabe) für eine Sollwerteingabe, davor Aufruf Datenblock 5, Herstellen eines Nullmerkers M20.0. Die Angaben zur Eingabe sind: Baugruppe 192, Kanal 4, Kanaltyp 3, Zielintervall [0;2000], keine Einzelabtastung,

6. (10 P) Der eingestellte Messbereich in einem Messumformer sei 40 - 90 °C. Im Fermenter sei augenblicklich eine Temperatur von $T = 56,75$ °C.

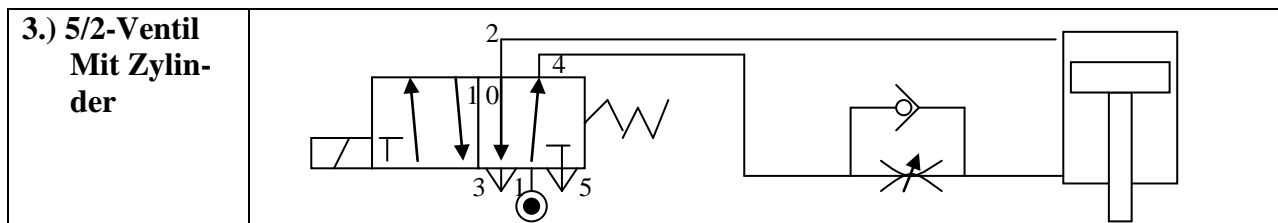
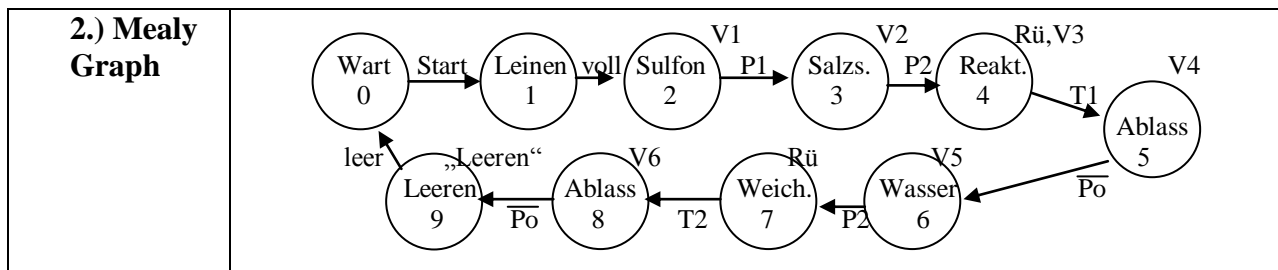
- Berechnen Sie das Messsignal in % und in mA für einen (4 - 20)- mA-Ausgang.
- Wir nehmen einen 12-Bit-Wandler an mit 4096 möglichen Stufen (0 bis 4095). Bilden Sie das Messsignal auf die Stufen ab (0 % entspricht Stufe 0, 100 % entsprechen Stufe 4095). Runden Sie die Stufennummer zur nächstgelegenen ganzen Zahl.
- Berechnen Sie den Dualwert der Zahl 1729. Geben Sie das Ergebnis auch hexadezimal an.

Lösungen zur Klausur SS 2012

1.) **Reglermatrix** berechnen: $K_{ps} = \begin{bmatrix} 3 \text{ U / KW} & 0,1^\circ \text{K / (U / min)} \\ -0,32 (\text{mg / l}) / \text{KW} & 0,0867 (\text{mg / l}) / (\text{U / min}) \end{bmatrix}$

$K_{ps}^{-1} = \begin{bmatrix} 0,297 & -0,342 \\ 1,096 & 10,27 \end{bmatrix}$ und mit $V_o T_u / T_g = 0,6$ wird Matrix $D = \begin{bmatrix} 0,3429 & 0 \\ 0 & 11,868 \end{bmatrix}$

$K_{ps}^{-1} D = \begin{bmatrix} 0,102 & -4,064 \\ 0,376 & 121,93 \end{bmatrix}$ und Ergebnis $R = K_{ps}^{-1} D K = \begin{bmatrix} 0,305 & -0,352 \\ 1,127 & 10,567 \end{bmatrix}$



1. Zurdnungs-funktion	$a+5,5m=0$ $a+6m=1$	$a+6m=1$ $a+6,5m=0$	$PR(P) = \begin{cases} -11+2P & 5,5 < P < 6 \\ 13-2P & 6 < P < 6,5 \\ 0 & \text{sonst} \end{cases}$
------------------------------	------------------------	------------------------	---

5. AWL	a) UN M 20.1 S M 20.1 U M 3.1 U E 1.7 UN E 1.8 = M 25.0	b) : A DB 5 : U M 20.0 : R M 20.0 : SPA FB250 Name: RLG : AE BG : KF +192	EINZ: M 20.0 XA: DW 9 FB: M 17.1 BU: M 17.2 TBIT: M 17.3
---------------	--	--	--

	O M 8.1 ON M 25.0 = A 4.7	KNKT: KY 4, 3 OGR: KF +2000 UGR: KF +0
--	---------------------------------	--

- 10a) $MB=40$ bis $90^{\circ}\text{C} = 50$ $MBA=40^{\circ}\text{C}$ $MBE=90^{\circ}\text{C}$ $\text{Prozent} = ((56,75-40)/50)*100 = 33,5\%$
 $mA = ((56,75-40)/50)*16+4 = 9,36 \text{ mA}$
 b) auf Stufe abbilden $n = ((56,75)/50)*4095 = 1371,8$ gerundet 1372
 c) Die Dezimalzahl 1729 ist binär 0110 1100 0001 hexadezimal 6C1

AT-Klausur WS 2012/13

<p>1. (12 P) Mealy-Graph einer Lotionproduktion: Mot= Motor, Kl=Klappe, ES=Endschalter Klappe, P0, P1,P2 = Pegelmelder, V1, V2,...=Ventile, Pu1, Pu2 = Pumpen. Die 9 Prozessschritte sind: 0. Warten auf Start, 1. Befüllen Lavendelessenz bis P1, 2. Klappenfreigabe mit Signal <i>FREI</i> (Klappe öffnet sich, Arbeiter befüllt mit Wachspenlen, drückt Klappe zu bis ES), 3. Homogenisieren und Heizen bis Temperaturmelder Temp1 anspricht, 4. 15 Min Temperatur halten, rühren, 5. Wasserkühlung bis Temperaturmelder Temp2 nicht mehr anspricht, 6. Seife bis P2 befüllen 7. 10 min mischen, 8. Abpumpen und zurück zum Warten.</p>	
--	--

Zeichnen Sie den Mealy-Graphen mit allen Übergangssignalen und allen Ausgabesignalen. Achten Sie auf den Einsatz des Rührers in den Prozessschritten 2-7.

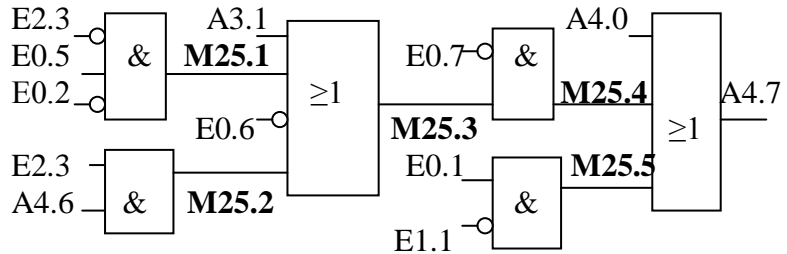
<p>2. (6 P) In einer pH-Fuzzy-Regelung mit Sollwert $W_{pH}=6$ soll eine Zuordnungsfunktion mit Nahmen $p_{HH}(pH)$ in Form siehe ganz rechts geschrieben werden. Die Daten entnehmen Sie der Graphik.</p>		$p_{HH}(pH) = \begin{cases} \dots, & \text{wenn} \\ \dots, & \text{wenn} \\ \dots & \text{sonst} \end{cases}$
---	--	---

3. (7 P) Der eingestellte Messbereich in einem Druck-Messumformer sei 0 – 10 bar. Im Fernerter sei augenblicklich ein Druck von $P = 3,71$ bar.

- Berechnen Sie das Messsignal in % und in mA für einen (4 - 20)- mA-Ausgang.
- Wir nehmen einen 10-Bit-Wandler an mit 1024 möglichen Stufen (0 bis 1023). Bilden Sie das Messsignal auf die Stufen ab (0 % entspricht Stufe 0, 100 % entsprechen Stufe 1023). Runden Sie die Stufennummer zur nächstgelegenen ganzen Zahl.
- Berechnen Sie den Dualwert der Zahl **-911** in 2-er-Komplement-Darstellung für negative Zahlen

4. (8 P)

Programmieren Sie in AWL das rechts abgebildete Netzwerk. Die fettgedruckten Merker **M25.1** bis **M25.5** benutzen Sie bitte als Zwischenspeicher.

**5. (6 P) Luftverbrauch eines Arbeitszylinders:**

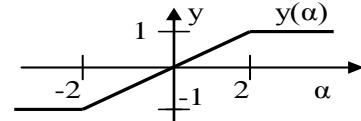
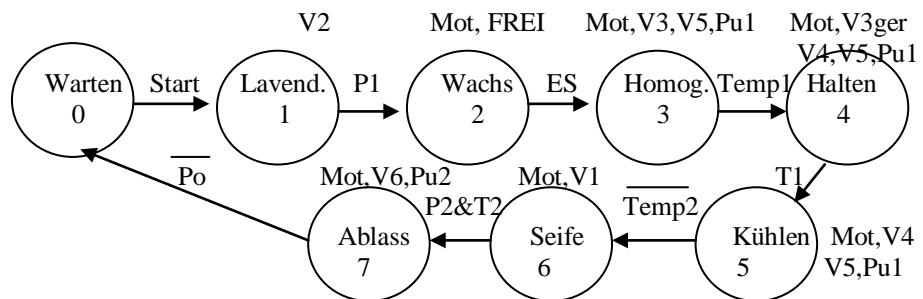
Durchmesser innen 125 mm, Hub 450 mm.

- Wie viele Hübe schafft man mit 80 m³ Druckluft?
- Welche Energie in KWh ist in 80 m³ Druckluft mit 10 bar Überdruck gespeichert?



6. (8 P) Berechnen Sie Bit 11 bis Bit 8 der Ausgabe eines 12-Bit-AD-Wandlers nach dem Wä-geverfahren. Die Messspannung ist $U_e = 7.13$ [V], der Messbereich geht von 0 bis 10 [V].

7. (5 P) Berechnen Sie zuerst die Aktivierung α eines Neurons mit den Eingangssignalen $X_1=0.3$, $X_2=-0.7$, $X_3=2.6$, den Gewichten $W_1=2$, $W_2=0.5$, $W_3=1.5$, und Schwellwert $\theta=5$, dann das Ausgangssignal y über die rechts abgebildete Aktivierungsfunktion $y(\alpha)$.

**Lösungen zur AT-Klausur WS 2012/13****1. Mealy Graph**

2. Zuordnungsfunktion

$$pHH(pH) = \begin{cases} 0 & \text{wenn } pH < 6 \\ 2pH - 12 & \text{wenn } 6 < pH < 6,5 \\ 1 & \text{wenn } pH > 6,5 \end{cases}$$

- 3. Messwerte umrechnen**
- 37,1% 9,936 mA
 - 1024 Stufen n=380
 - 911 im 2-er-Komplement ist 1111 1100 0111 0001

4. AWL UN E 2.3 U E 2.3 O A 3.1 UN E 0.7 U E 0.1 O A 4.0

$$\begin{array}{llll}
 U & E & 0.5 & U & A & 4.6 & O & M & 25.1 & U & M & 25.3 & U & N & E & 1.1 & O & M & 25.4 \\
 UN & E & 0.2 & = & M & 25.2 & ON & E & 0.6 & = & M & 25.4 & = & M & 25.5 & O & M & 25.5 \\
 = & M & 25.1 & & & & O & M & 25.2 & & & & & & & = & A & 4.7 \\
 & & & & & & = & M & 25.3 & & & & & & & & &
 \end{array}$$

5.a) Zahl N der Hube mit 80 m³ Druckluft $N = 80 / V_{Hub} = 14486$ Hube

b) Energie $E = 10 \cdot 10^5 \cdot 80 = 8 \cdot 10^7$ Ws = 22,22 KWh

6. 12-Bit-AD-Wandler Wägeverfahren

Bit 11 $7,13 \geq 5,0$ Bit 11 wird 1 gesetzt Neue Vergleichsspannung ist $5 + 10 / 4 = 7,5$ V
 Bit 10 $7,13 < 7,5$ Bit 10 wird 0 gesetzt Neue Vergleichsspannung ist $7,5 - 10 / 8 = 6,25$ V
 Bit 9 $7,13 \geq 6,25$ Bit 9 wird 1 gesetzt Neue Vergleichsspannung ist $6,25 + 10 / 16 = 6,875$ V
 Bit 8 $7,13 \geq 6,875$ Bit 8 wird 1 gesetzt

6. Aktivierung eines Neurons: $\alpha = 0,3 \cdot 2 - 0,7 \cdot 0,5 + 2,6 \cdot 1,5 - 5 = -0,85$

$$y(\alpha) = \begin{cases} -1 & \text{wenn } \alpha < -2 \\ 0,5 \cdot \alpha & \text{wenn } -2 \leq \alpha \leq 2 \\ 1 & \text{wenn } \alpha > 2 \end{cases} = 0,5 \cdot (-0,85) = -0,425$$

AT-Klausur SS 2013

1. (12 P) Berechnen Sie nach dem Beispiel von Übung 21 in 3 Schritten die Reglermatrix R (Schritt 4 nicht rechnen). Gegeben sind die 4 Sprungantworten zu zwei Stellgrößensprünge Δy_1 und Δy_2 :

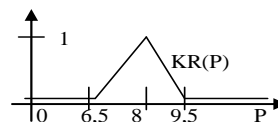
$\Delta y_1 = 500$ [W]		
$\Delta y_2 = 30$ [mbar]		

Schritt 1: Kps-Matrix berechnen und Dimensionen angeben

Schritt 2: Die beiden Haupt-P-Regler D_{11} und D_{22} mit $V_o T_u / T_g = 0.6$ berechnen

Schritt 3: Die inverse Kps-Matrix berechnen und daraus Reglermatrix R der P-Regler berechnen

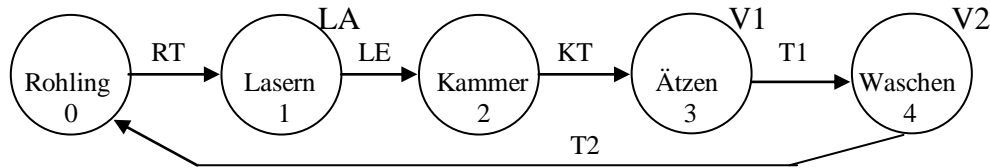
2. (6 P) In einer Fuzzy-Regelung soll eine Zuordnungsfunktion mit Namen $KR(P)$ in Form einer Auswahlklammer geschrieben werden (siehe ganz rechts). Die Daten entnehmen Sie der Graphik.



$$KR(P) = \begin{cases} \dots, & \text{wenn} \\ \dots, & \text{wenn} \\ \dots & \text{sonst} \end{cases}$$

3. (4 P) Der eingestellte Messbereich in einem Messumformer sei 10 - 50 °C. Im Fermenter sei augenblicklich eine Temperatur von $T = 28,37$ °C. Berechnen Sie das Messsignal in % und in mA für einen (4 - 20)- mA-Ausgang.

4. (10 P) Gegeben ist ein Mealy-Graph als Prozessbeschreibung für eine typographische Ätzanlage zur Herstellung von beschrifteten Messing- bzw. Edelstahlschildern:



Die Schritte sind: **Betriebstaste BT** drücken **0. Rohling** per Hand positionieren. Der Bediener drückt Taste RT. **1. Lasern** zur Zerstörung der Lackschicht. Die Lasersteuerung gibt am Ende das Signal LE aus. **2. Kammer** manuell mit dem gelaserten Rohling beschicken. Der Bediener drückt Taste KT. **3. Ätzen** mit Ätzflüssigkeit aus Ventil V1 über eine Zeit von 100 Sekunden. **4. Waschen** mit Lackentferner aus Ventil V2 über eine Zeit von 60 Sekunden. Zurück zu Schritt 0 (Schleifenende, wenn BT=0).

- Schreiben Sie ein C-Programm nach der Vorlage von Übung 1.
- Zeichnen Sie eine Schalttafel mit den LEDs (auch BetriebsLED) und Tastern

5. (4 P)	Berechnen Sie die statische Kraft F [N], mit der der Kolben eines doppelt wirkenden Zylinders auf die Fläche A drückt. $P_1=0$ bar, $P_2=10$ bar, Federkraft im Punkt A ist 250 [N]. Innendurchmesser Zylinder 50 mm, Durchmesser Kolbenstange 15 mm.	
-----------------	---	--

6. (8 P) Zeichnen Sie den Mealy-Graph mit den Zuständen **0, 1,...,6** mit allen Überführungs- und Ausgabesignalen zu folgendem Prozess:

0. Warten auf Startsignal. **1. Befüllen VE:** Mischtank über Ventil V1 bis Pegel P1 mit VE-Wasser befüllen. **2. Befüllen Alk.:** Mischtank über Ventil V2 bis Pegel P2 mit Alkohol befüllen. **3. Fett:** Leuchtschrift L02 einschalten („100 Kg Fettwürfel einwerfen, Luke verriegeln und quittieren“) und warten, bis Lukensignal E08 **und** Quittungssignal E09 des Arbeiters beide anliegen. **4. Erhitzen:** Rühren (R) und heizen (Hz), bis das Temperatursignal Tk erscheint. **5. Homogenisieren:** 8 Minuten rühren. **6. Abpumpen:** Abpumpen (Pu), bis sich Pegelmelder Po nicht mehr meldet. Dann zurück ins Warten.

Lösungen zur Klausur SS 2013

1. (12P) Aufgabe: Reglermatrix berechnen : Kps-Matrix (4P), D+K-Matrix (4P), R-Matrix (4P)

kps-Matrix	$\begin{bmatrix} 0,02 & 0,2 \\ -0,0044 & 0,04 \end{bmatrix}$	$\begin{matrix} ^\circ\text{C} \\ \text{mg/l} \end{matrix}$	$\begin{matrix} \text{Kps}^{-1} \\ \text{Watt} \end{matrix}$	$\begin{bmatrix} 23,8 & -119 \\ 2,62 & 11,9 \end{bmatrix}$	$\begin{matrix} \\ \text{mbar} \end{matrix}$
D-Matrix	$\begin{bmatrix} 42,5 & 0 \\ 0 & 26,8 \end{bmatrix}$	K-Matrix	$\begin{bmatrix} 0,02 & 0 \\ 0 & 0,04 \end{bmatrix}$		
Kps*D	$\begin{bmatrix} 1011 & -3189 \\ 111 & 318,9 \end{bmatrix}$	R-Matrix	$\begin{bmatrix} 20,2 & -127,6 \\ 2,2 & 12,8 \end{bmatrix}$	$\begin{matrix} \text{Watt} \\ \text{mbar} \end{matrix}$	
	$^\circ\text{C}$	mg/l			

2. (6 P) In einer Fuzzy-Regelung: Bild insgesamt 1P, 1.Z. 2P, 2.Z. 2P, 3.Z. 1P

$$KR(P) = \begin{cases} -4,33 + 0,67 P & \text{wenn } 6,5 < P < 8 \\ 6,33 - 0,67 P & \text{wenn } 8 < P < 9,5 \\ \text{sonst} & \end{cases}$$

3. (4 P) Der eingestellte Messbereich:

T=	28,37 °C
MBA=	10 °C
MBE=	50 °C
mA=	11,348 mA
P%=	45,925 %

4. (10 P) Gegeben ist ein Mealy-Graph:

a) Schreiben Sie ein C-Programm nach der Vorlage von Übung 1.(Pro Fehler ein Punkt Abzug)

```
int main(...)           // Der Inhalt der Klammer hängt vom Betriebssystem ab
{ double T1, T2,...;    // Timer arbeiten mit gebrochenen Zahlen, also double-Zahlen verwenden
  int LED0, LED1, LED2, ...V1, V2, .....BT;
```

```
    BT=1;  T1= T2= .....= V1=...= 0; //alle Variablen (außer Betriebstaste BT) auf Startwert
0
```

```
do{           // do-while-Schleife über alle Prozesszustände von Positionieren bis Ablassen
```

```
    LED0=1;           // Schalte LED1 „Warten auf Taste RT“ ein.
    do{ }while( !RT); // Warten auf das RT-signal, d.h. rotiere solange RT= 0 ist
    LED0=0;           // Schalte LED aus
```

```
    LED1=1;           // Schalte LED1 „Lasern“ ein.
    do{ L=1;} while( !LE); // Warten auf das LE-signal, d.h. rotiere solange LE= 0 ist
    L=LED1=0;         // Schalte Laser und LED aus
```

```
    LED2=1;           // Schalte LED2 „Kammer beschicken“ ein.
    do{ }while( !KT); // Warten auf das KT-signal, d.h. rotiere solange KT= 0 ist
    LED2=0;           // Schalte LED aus
```

```
    T1 = time() + 100000; LED3=1; // Bis zur Zeit T1 ist zu ätzen (Zeit in ms)
    do{ V1=1;} while ( time() < T1 );
    V1=LED3=0; // Schließe Ventil und schalte LED aus
```










```
    T2 = time() + 60000; LED3=1; // Bis zur Zeit T2 ist zu waschen
    do{ V2=1;} while ( time() < T2 );
    V2=LED4=0; // Schließe Ventil und schalte LED aus
```

```
} while (BT); // Verlassen der Schleife bei Betrieb = 0, ansonsten zurück zum Positionieren
```

```

BetriebsLED=0;    // Schalte die BetriebsLED wieder aus.
return (0);       // Gib Rückkehrcode 0 an das Betriebssystem zurück (alles okey)
}

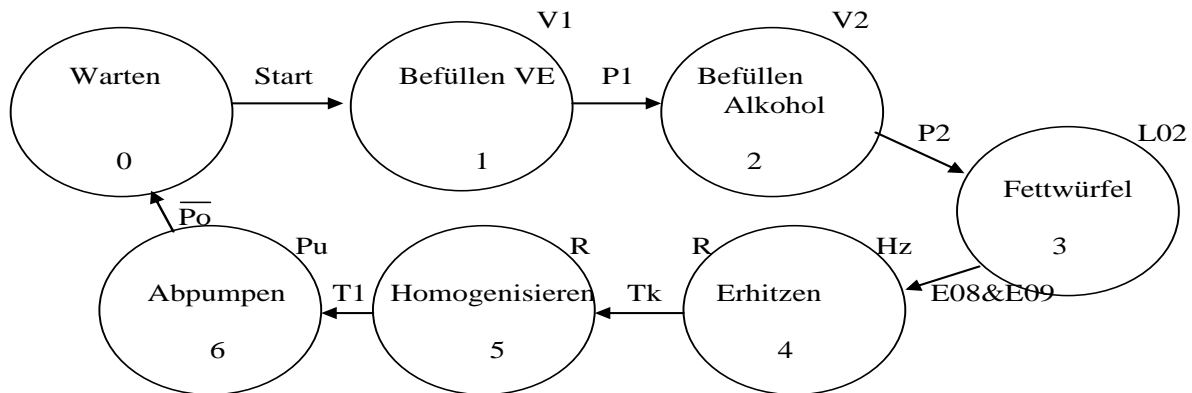
```

b) Zeichnen Sie eine Schalttafel mit den wichtigen LEDs und Tastern (2 Punkte)	<div style="display: flex; flex-wrap: wrap;"> <div style="width: 50%;"> <p> BT</p> <p> RT</p> <p> KT</p> </div> <div style="width: 50%;"> <p> Betrieb</p> <p> Positionieren</p> <p> Lasern</p> <p> Kammer besch.</p> <p> Ätzen</p> <p> Waschen</p> </div> </div>
---	---

5. (4 P) Berechnen Sie die statische Kraft F [N] :

Feder	250 N
Durchm	
Zyl	0,05 m
Durchm	
KSt	0,015 m
Druck p	1,00E+06 Pa
F=	2036,7808 N

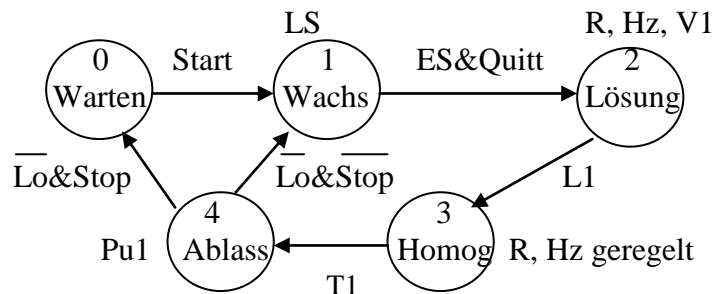
6. (8 P) Zeichnen Sie den Mealy-Graph mit den Zuständen **0, 1,...,6** (Pro Fehler ½ Punkt Abzug)



7. Liste der Übungen aus der Vorlesung

7.1 Programmierung einer kleinen Prozesssteuerung in C (Kap. 2.1)

Das Programm besteht aus 2 großen geschachtelten Schleifen und fünf Inneren. Ein Zweipunktregler schaltet die Heizung aus im Zustand 3, wenn das Temperatursignal TEMP anliegt. Die Timerzeit T1=10 min. Die Stopp-taste rastet ein beim Drücken. ES=Endschalter, R=Rühren, LS=Lichtsignal, HZ=Heizen, V1=Ventil, Li=Pegelmelder, Pu1=Pumpe.

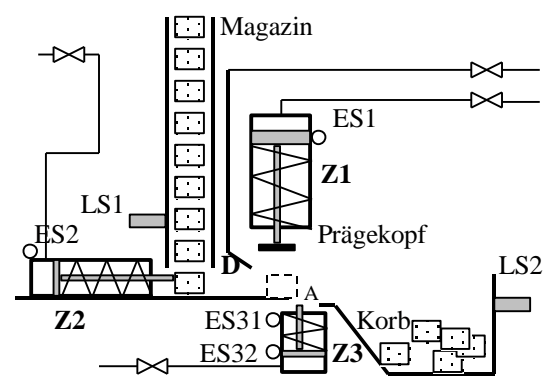


7.2 Einen Mealy-Graph nach einer Prozessbeschreibung zeichnen (Kap. 2.1)

7.2a) Als Übung zu Hause: Entwerfen Sie den Mealy-Graphen mit seinen Prozessschritten für folgenden chemischen Prozess: Warten, Befüllen des Reaktionsgefäßes mit Baumwolle bis der Arbeiter Signalknopf "voll" drückt, Befüllen mit Schwefelsäure bis Pegelmelder P1, Befüllen mit Salpetersäure bis P2, 10 Minuten Reaktionszeit abwarten (Timer T1) bei gleichzeitiger Kühlung über den Mantel, Ablassen der Säure (P0 abfragen), Wasserzulauf bis Pegel P2, 5 Minuten einweichen lassen (Timer T2), Ablassen des Waschwassers (P0 abfragen), Leuchtsignal "Leeren" für den Arbeiter als Start zum Entleeren, zurück ins Warten wenn von diesem Signalknopf "leer" gedrückt wird

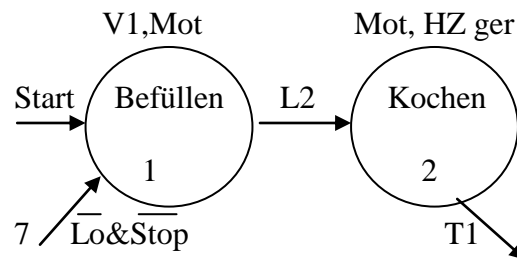
- Machen Sie eine Anlagenskizze mit Ventilen V1, V2, ..., Kessel, Kühlmantel, Pegelmeldern
- Zeichnen Sie den Mealy-Graphen mit Überführungs- und Ausgabesignalen

7.2 b) Zeichnen Sie den Mealy-Graphen der Prozessschritte *Warten, Anschlag ausfahren, Schieben, Prägen, Freigeben, Ausblasen* mit allen Überführungssignalen (Endschalter, Timer usw.) an den Pfeilen und allen Ausgabesignalen (Z1, Z2, ...) an den Zuständen (Kreisen). Zyl. Z2 fördert die Teile zum Anschlag A, der von Zyl. Z3 ausgefahren wird. Dann prägt Zyl. Z1 das Teil. Wenn Anschlag **und** Zyl1 **und** Zyl2 zurückgefahren sind, bläst Düse D das Teil in den Korb. Ein neuer Zyklus beginnt. Wenn Lichtschranke LS1 sich meldet **oder** LS2 nicht, dann gehen Sie in den Zustand *Warten*. ES1, ES2, ... sind Endschalersignale. Wo Endschalter fehlen, nehmen Sie Timer T1, T2, ...



7.3 FUP-Programmierung einer SPS: Taktkette, Meldeteil, Ausgabeteil (Kap. 2.2)

Gezeigt wird ein Ausschnitt aus einem Mealy-Graphen. **Taktkette** 2 Netzwerke (Zustand 1 und 2), **Meldeteil** 2 Netzwerke LED1 und LED2), **Ausgabeteil** 3 Netzwerke (V1, Mot, HZ geregelt mit Zweipunktregler bei Nutzung von Temperatursignal TEMP), *Ausgabesignale* sind: V1=Füllventil, Mot=Rührermotor, HZ=Heizung, *Steuersignale* sind: Start, Stop,



T1=Timer (25 min), *Sensorsignale* sind: Lo,L2=Pegelmelder (melden bei Berührung mit Flüssigkeit Signal 1). Befüllen (1) wird aus den Zuständen Warten (0) und Ablassen (7) erricht.

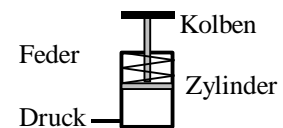
7.4 Umrechnung von Einheiten, Binärzahlen, Wägeverfahren beim ADW (Kap. 3.1)

Der eingestellte Messbereich in einem Messumformer sei 20 - 80 °C. Im Fermenter sei augenblicklich eine Temperatur von $T=36,25$ °C.

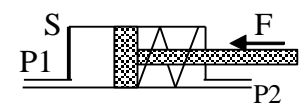
- Berechnen Sie das Messsignal in % und in mA für einen (4 - 20)- mA-Ausgang. Berechnen Sie die anliegende Temperatur T °C und T% zum Signaleingang 17,1 mA auf 3 Dezimalen.
- Wir nehmen einen 12-Bit-Wandler an mit 4096 möglichen Stufen (0 bis 4095). Bilden Sie das Messsignal auf die Stufen ab (0 % entspricht Stufe 0, 100 % entsprechen Stufe 4095). Runden Sie die Stufennummer zur nächstgelegenen ganzen Zahl.
- Berechnen Sie den Dualwert der gerundeten Stufennummer. (Falls Sie Aufgabe b) auslassen möchten, dann von der beliebigen Stufennummer 972). Geben Sie das Ergebnis auch hexadezimal an.
- Berechnen Sie Bit 11, 10, 9, 8 der Ausgabe eines AD-Wandlers für die Spannung 3,79 Volt nach dem Wägeverfahren.

7.5 Energiekosten und Kraftentfaltung beim Pneumatikzylinder (Kap. 4)

a) Welcher Druck P sollte am Pneumatikzylinder anliegen, der mit seinem Kolben einen Motorblock der Masse 175 Kg anheben kann. Die maximale Rückstellkraft der Feder ist 200 [N]. Der Durchmesser ist $D=6$ [cm].



b) Berechnen Sie die statische Kraft F [N], mit der der Kolben eines doppelt wirkenden Zylinders auf die Stirnfläche S drückt. $P_1=0$ bar, $P_2=6$ bar, Federkraft im Punkt S ist 120 [N]. Innendurchmesser des Zylinders 40 mm, Durchmesser Kolbenstange 12 mm.



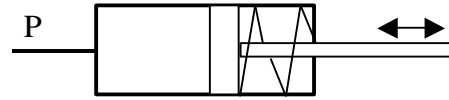
c) Berechnen Sie die dynamische Kraft bei Linksbewegung und Mittelstellung des Kolbens unter Berücksichtigung der Leitungsdruckverluste von beidseitig je 1 bar, der mittigen Federkraft von 550 [N] und der durch die Kolbenstange verminderten Fläche ($P_1=0$ bar, $P_2=6$ bar ohne Leitungsdruckverluste)

Luftverbrauch eines Arbeitszylinders: Durchmesser innen 50 mm, Hub 270 mm.

d) Wieviel Hube schafft man mit 1 m^3 Druckluft?

e) Welche Energie in KWh ist in 20 m^3 Druckluft mit 8 bar Überdruck gespeichert? Welche Kosten fallen an bei 0,25 €/KWh ?

f) Wieviel Frischluft wird dafür angesaugt?



7.6 Details aus einem RI-Fließbild (Kap.5.1.2)

a) Zeichne eine Heizschlange, die über einen Wärmetauscher Wärme an einen Querstrom abgibt. Zeichne MSR-Stellenkreis K701 für Temperatur, Regeln, Anzeigen in Messwarte, Registrieren.

b) Zeichne MSR-Stellenkreis T306 für eine Pegelregelung (Level) zwischen Pegeln L1 und L2 in einem Tank mittels einer Pumpe (nur örtliche Anzeige).

c) Zeichne MSR-Stellenkreis D408 für eine Bodendruckmessung (Pressure) in einem Tank (um in der Messwarte daraus den Füllstand berechnen zu können), d.h. Anzeige in Messwarte.

7.7 Identifikation und Blockschaltbild einer Regelstrecke (Kap. 5.2.2)

<p>Als Übung: $U_0=7,5 \text{ [KW]}$ Bestimme den Streckentyp Bestimme alle Parameter des Modells Zeichne das Blockschaltbild</p>	<p>Als Hausaufgabe: $U_0=0,250 \text{ [KW]}$ Bestimme den Streckentyp Bestimme alle Parameter des Modells Zeichne das Blockschaltbild</p>

7.8 Reglerparametrierung nach Ziegler/Nichols und nach Chien et al. (Kap. 5.3.3)

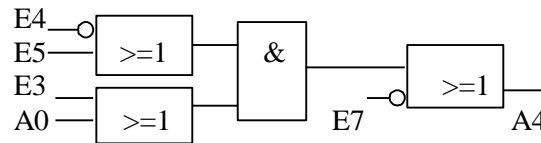
<p>Reglerparametrierung nach Ziegler/Nichols: Kritische Verstärkung ist $K_{PR\ kr} = 6,85 \text{ [°/KW]}$ Die Schwingungsperiode τ_{kr} bestimmen Sie aus der nebenstehenden Graphik a) Für einen PI-Regler b) für einen PD-Regler</p>	
---	--

c) Reglerparametrierung nach **Chien et al.**: Die Regelstrecke ist vom Typ PTn mit $K_{PS}=39,44 \text{ [°/KW]}$, $T_u=17 \text{ [s]}$, $T_g=89 \text{ [s]}$. Gesucht ist ein PID-Regler für 0% Überschwingen und Führungsregelung.

7.9 Kleine Aufgaben der AWL-Programmierung einer SPS (Kap. 5.4.2)

a) Logikschaltplan aus UND- und ODER-Boxen, eventuell mit negierten Eingängen zu folgendem Problem zeichnen: Ventil V4 soll geöffnet werden, wenn Pumpe Pu7 läuft und Pegelmelder P3 sich meldet, oder wenn Pumpe Pu6 läuft und Pegel P2 sich meldet. Wenn Pegelmelder P4 sich meldet, blockiert er die Ausgabe des V4-Signals (Negierter Eingang in einer UND-Box).

b) Programmieren Sie den rechts abgebildeten logischen Schaltplan in AWL-Sprache. Als Zwischenspeicher benutzen Sie die Merker M77.0, M77.4 und M77.5



c) Programmieren Sie in AWL einen 2-Punkt-Regler (Schalter) für Ausgang A3.5. Der Sollwert steht auf DW9, der Istwert auf DW22 im Datenblock 16.

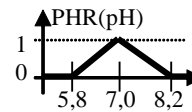
7.10 Umwandlung einer Differentialgleichung in eine Differenzengleichung (Kap. 5.4.6)

Wandeln Sie die DGL in eine Differenzengleichung um und lösen Sie sie nach x_n auf:

$$T^2 \ddot{x}(t) + 2T \dot{x}(t) + x(t) = b_o y(t)$$

7.11 Zuordnungsfunktion und Regel programmieren in der Fuzzy-Control (Kap. 5.5)

7.11 a) In einer Fuzzy-Regelung soll eine Zuordnungsfunktion mit Namen PHR(pH) in Form einer Auswahlklammer geschrieben werden (siehe ganz rechts). Die Daten entnehmen Sie der Graphik.



$$PHR(pH) = \begin{cases} \dots, & \text{wenn} \\ \dots, & \text{wenn} \\ \dots & \text{sonst} \end{cases}$$

7.11 b) Programmieren Sie mit der Min-Max-Methode folgende Regel: „Wenn der Druck hoch ist und die Temperatur richtig, oder wenn der Druck richtig ist und die Temperatur richtig ist und die Temperaturänderung positiv ist, dann betätige das Abblasventil V7. Berechnen Sie den Wert der Ergebnisfunktion V7 für folgende Werte der Zuordnungsfunktionen der Prädikate und dem alten Wert von V7: $V7_{alt}=0.3$, $PH=0.7$, $TR=0.9$, $PR=0.85$, und $DTP=0.16$.

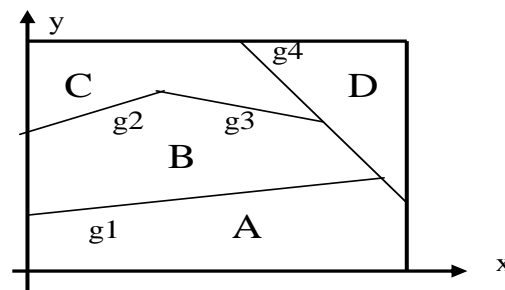
7.12 Ausgangssignal eines Neurons (Kap. 5.6.1)

Berechnen Sie das Ausgabesignal y eines Neurons. Die 5 Eingangssignale sind 0.2, 0.7, 1.2, 1.9, 0.3. Ihre Gewichte sind 2.3, 1.8, -0.6, 0.77, -0.2, Schwellwert ist $\theta = 1.7$. Aktivierungsfunktion siehe rechts \rightarrow

$$y(\alpha) = \begin{cases} 1 - e^{-0.5\alpha} & \text{wenn } \alpha \geq 0 \\ -1 + e^{0.5\alpha} & \text{sonst} \end{cases}$$

7.13 Logische Bedingungen bei der multivariaten Klassifikation (Kap.5.6.2)

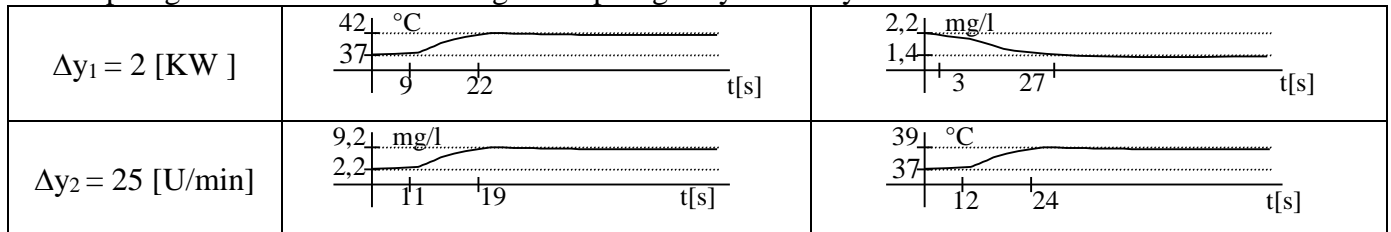
Legen Sie durch UND und ODER und \geq bzw. $<$ die Klassifikation eines Objekts mit den Koordinaten (y, x) in den Klassen A, B, C, bzw. D fest. Die Trenngeraden sind $g1(x)$, $g2(x)$, ... (Klammern Sie, um Vorrangregeln zu umgehen.) Beispiel: Klasse D: $y \geq g4(x)$



7.14 Koppelmaß und Kopplungsregler berechnen nach Boksenbom und Hood (Kap. 5.7)

7.14 a) Berechnen Sie das Koppelmaß K_{12} mit X_1 =Temperatur, X_2 =Sauerstoff, X_1 steigt von 37 auf 39 °C, X_2 von 2,7 auf 9,2 mg/l, und den Toleranzen $D_1 = \pm 1^\circ\text{C}$ bzw. $D_2 = \pm 0,3 \text{ mg/l}$.

7.14 b) Berechnen Sie in 3 Schritten die Reglermatrix R (Schritt 4 nicht rechnen). Gegeben sind die 4 Sprungantworten auf zwei Stellgrößensprünge Δy_1 und Δy_2 :



Schritt 1: Kps-Matrix berechnen und Dimensionen angeben

Schritt 2: Die beiden Haupt-P-Regler D_{11} und D_{22} mit $V_o T_u / T_g = 0.6$ berechnen

Schritt 3: Die inverse Kps-Matrix berechnen und daraus Reglermatrix R der P-Regler

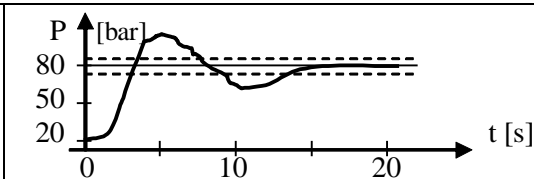
7.15 Parametrierung von einfachen Übertragungsfunktionen (Kap. 5.8.2)

Schreiben Sie die Übertragungsfunktionen der folgenden Bauteile einmal allgemein und dann mit eingesetzten Zahlenwerten auf:

PID-Regler mit $K_{pr} = 19,3 \text{ [KW/K]}$, $T_N = 32 \text{ [s]}$, $T_V = 4,7 \text{ [s]}$
 Rührkessel Typ PT3 mit $K_{ps} = 0,77 \text{ [K/KW]}$, $n = 3$, $T = 12 \text{ [s]}$
 Rückführung Typ PT1 mit $K_p = 1$, $T = 2,3 \text{ [s]}$

7.16 Gütemaße der Regelung (T_{an} , T_{aus} , X_{max} , Dämpfung (Kap. 5.8.4)

Gegeben ist die Sprungantwort eines Regelkreises auf einen Sollwertsprung von $p = 20$ auf $p = 80$ bar. Die neuen Toleranzgrenzen sind gestrichelt. Bestimmen Sie die Anregelzeit T_{an} , die Ausregelzeit T_{aus} , das maximale Überschwingen $X_{m,absolut}$ und $X_{m\%}$ und die Dämpfung δ .

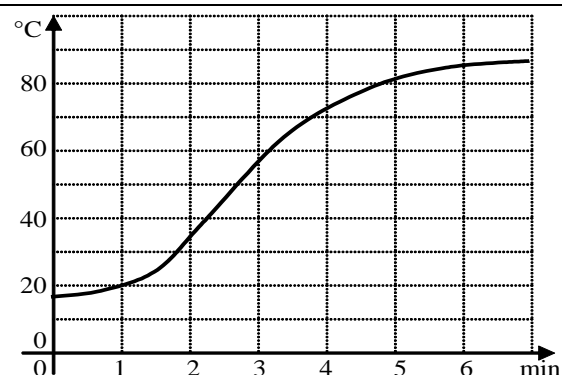


7.17 Identifikation einer Regelstrecke und Berechnung der Sprungantwort (Kap. 5.8.4)

Identifizieren Sie die Regelstrecke:

Die Sprungantwort auf einen Stellgrößensprung von $U_o = 2,5 \text{ KW}$ ist rechts gezeigt.
a) Bestimmen Sie alle Parameter des Streckenmodells.

b) Berechnen Sie die Temperatur nach einem Stellgrößensprung $U_o = 1,6 \text{ KW}$ und Starttemperatur $x_o = 32^\circ\text{C}$ für die Zeit $t = 3,5 \text{ min}$.



7.17 Beispiele für Verständnisfragen zum Skript

Anlagenautomatisierung mit SPS: Was ist der Akku?

Kommunikation: Was ist ein Router?

Unstetige Regler: Was ist ein Dreipunktregler?

Fuzzy Control: Was macht eine Zuordnungsfunktion?

8. Wörter der Automatisierungstechnik - Dictionary of Automation

Abbruch	abort, program stop	Messumformer	measuring transformer
abfahren (Prozess)	shut-down	Nachstellzeit (Regler)	reset time
Ablaufsteuerung	sequencing control	Netzwerk	ladder contact network
Abtastzeit	sampling time	Nocke (Nockenscheibe)	cam
Aktor	actuator	Pneumatikplan	Pneumatic circuit diagram
Analogwertverarbeitung	analog value processing	pneumatische Steuerung	pneumatic control
anfahren (Prozess)	start, start-up	Programmiergerät	programmer
Anpassung (SPS)adapter block		Prozessabbild	process I/O image
Anschluss (Schlauch)	connection	Pumpe	pump
Anweisung (Programm)	instruction, statement	Rauchmelder	smoke detector
Anweisungsliste	statement list	Regelgröße	controlled variable
Aufbau	construction	Regelkreis	closed (control) loop
Ausfall	failure	Regelstrecke	controlled member
Ausschaltverzögerung	release delay	Reglervverstärkung	gain
Automatikbetriebautomatic mode		Relais	electrical relay
Automatisierungseinrichtung	automation system	Richtimpuls	initializing pulse
Automatisier.-funktion	automation function	Rückschlagventil	non-return valve
Automatisierungsgerät	programmable controller	Schalldämpfer	silencer, muffler
Baustein (Programm)	module, block	Schaltbild	connection diagram
Befehl	instruction,command	Schalter	switch
Befehlskette	sequence	Schaltstellung	position
Beharrungswert	steady-state value	Schaltwarte	control room, - centre
Bereichüberschreitung	limit error, overflow	Schaltzeichen	switching symbol
Betriebsartenteil	operating mode module	Schlauch(kupplung)	tube (coupler)
Blockschaltbild	block diagram	Schütz (Relais)	contactor
Dämpfung	damping	SPC	(storage programmable control).
Drossel(ventil)	throttle (valve)	SPS	SPC stor. progr.control
Druckregler	pressure controller	Speicherverhalten	storage properties
Düse	nozzle, jet	Spindel (Ventil)	spindle, stem, rod
Einschaltverzögerung	closing delay	S-Regler	step controller
elektropneumatisch	electropneumatic	Starttaste	start button, start key
Endlagenschalter	limit switch	Staudruck	impact pressure
Exschutz	explosion protected	Stellgröße	control variable
feststellen (Spannvorr.)	chuck, clamp, gripp	Steuerungsaufgabe	control function
Flankenbewertung	pulse edge evaluation	Steuerwort	control -/command word
Freigabe Signal, -Luft	release sig., enable air	Stoßmagnet	push-type electromagnet
Führungsregelung	pilot control, setpoint c.	Taktkette (Ablaufkette)	sequence cascade
FUP Funktionsplan	function Diagram	Totzeit	delay time
Getriebe	gear, transmission	Turbulenzverstärker	turbulence intensifier
Grundstellung (-zustand)	initial position (- state)	Überführung (Graph)	transition
Handbetrieb	manual mode	Überlastsicherung	overload protection
Kante (Graph)	edge	Ventil	valve
Klemme, Kontakt	clamp, contact	Verriegelung	interlocking
		Verstärker	amplifier

Knoten (Graph)	node, state	Vorhaltezeit (Regler)	rate time
Kolben(stange)	piston (rod)	Weiterschaltbedingung	stepping conditions
Kontaktplan	ladder diagram	Wirkungsgrad	operating ratio
K-Regler	continuous controller	Zähler	counter
LED (Leuchtdiode)	light emitting diode	Zeitfunktion	timing function
Leitstand	control room	Zeitglied	timer
Leitsystem	control system	Zeitkonstante	time constant
Leitung (Strom,Luft)	electric line, air pipe	Zeitplansteuerung	time-program control
Lichtschranke	light barrier	Zeitverhalten	time response,specified t.
logische Verknüpfung	logic operation	zuführen	supply, feed
lösen (Spannvorrichtung)	release, loose	Zugmagnet	pull-type electromagnet
Luftaufbereitung	air conditioning	Zustand (Graph)	state (of graph)
Merker	flag, marker	Zylinder	cylinder