



Strukturierte objektorientierte Automatisierung

VL Prozessleittechnik I (SS 2011)

Professur für Prozessleittechnik

Prof. Dr.-Ing. Leon Urbas, Falk Doherr, Michael Obst

Inhalt

- Objekte in der Automatisierung
- Objektorientierte Leitechnik
- Herausforderungen an die Planung der Automatisierung
- Typicals von Funktionsbaustein
- Ausblick

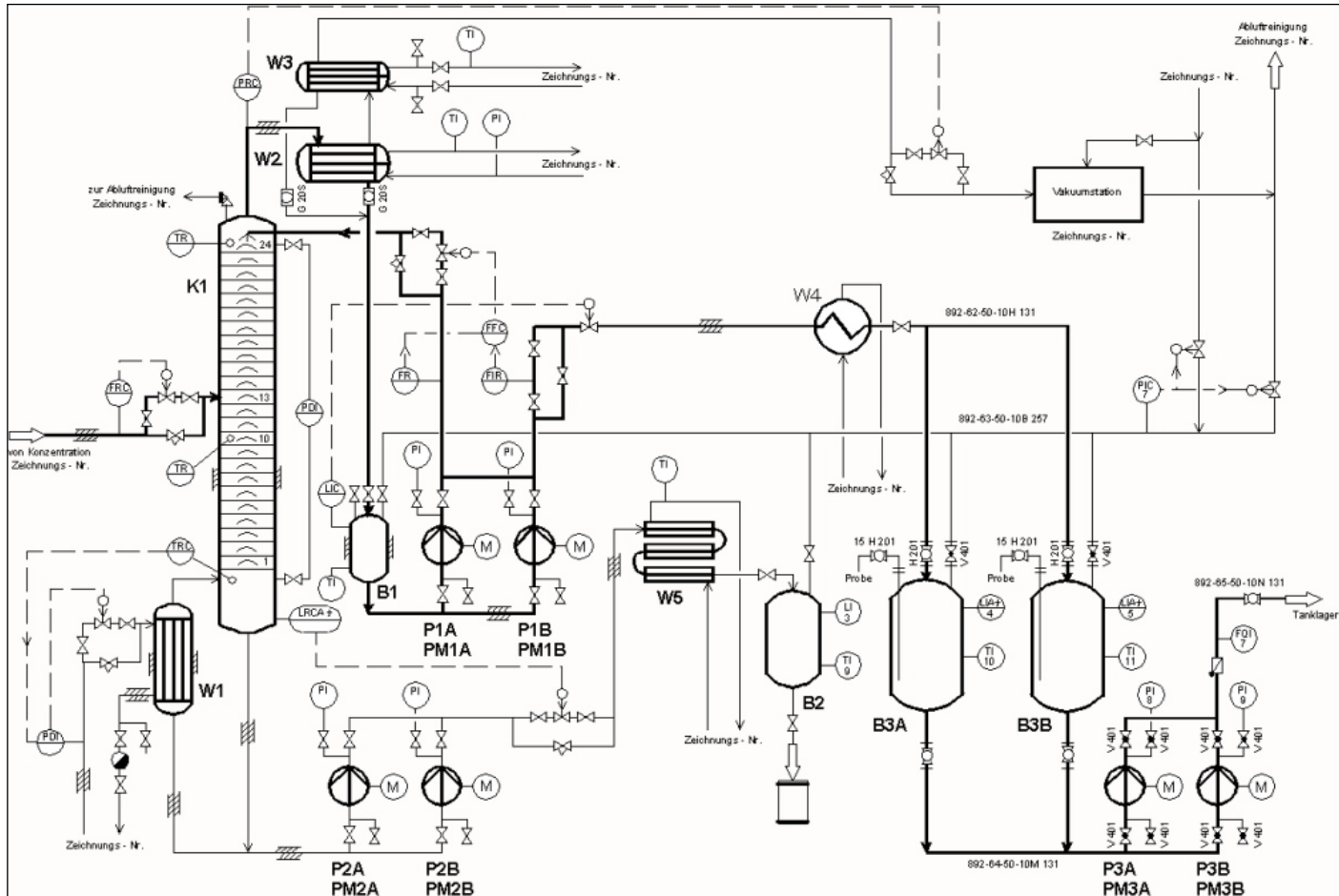
Objekte in der Automatisierung

- **Brainstorming:** Welche Objekte und Strukturen (Objektzusammenhänge) finden sich in einer automatisierten technischen Anlage?
 - Hardware
 - Software
- Fragen:
 - Wie groß ist Häufigkeit dieser Objekte und Strukturen in automatisierten technischen Anlagen?
 - Wie groß ist die funktionale Varianz dieser Objekte und Strukturen in automatisierten Anlagen?

Objekte in der Automatisierung

Objekt		Häufigkeit	fkt. Varianz
Aktoren	Motoren		
	Ventile		
Sensoren	digital		
	analog		
Signalanbindungen			
Hardware Informationsverarbeitung			
Software Informationsverarbeitung			
HMI (Human Machine Interface – Informationseingabe u.–ausgabe)			

Objekte in der Automatisierung

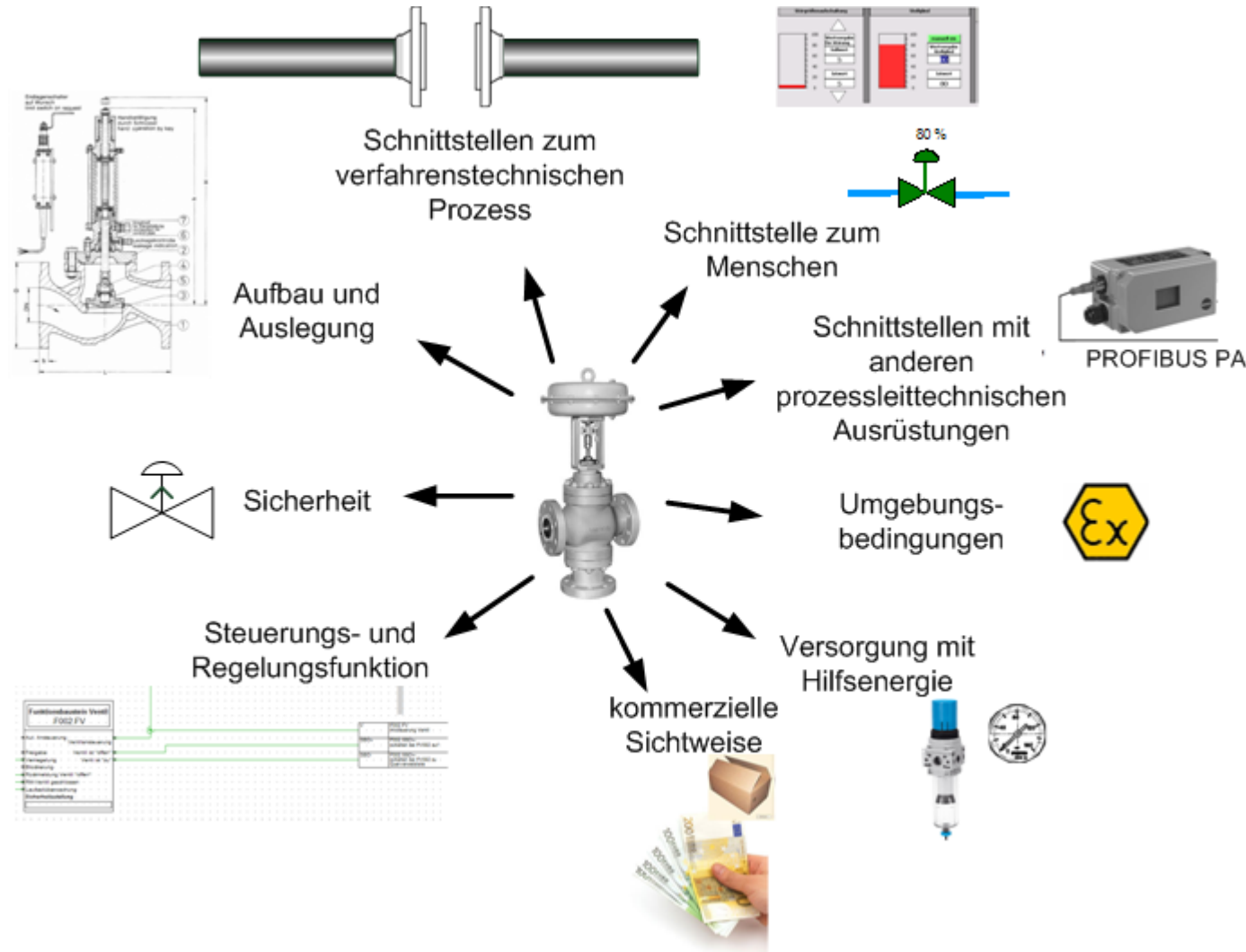


Objekte in der Automatisierung

Objekt		Häufigkeit	fkt. Varianz
Aktoren	Motoren	FT: sehr häufig VT: häufig	FT: niedrig VT: niedrig
	Ventile	FT: häufig (Ventilinseln) VT: sehr häufig	FT: sehr niedrig VT: niedrig
Sensoren	digital	FT: sehr häufig VT: häufig	FT: niedrig VT: niedrig
	analog	FT: gering VT: häufig	FT: sehr niedrig VT: normal
Signalanbindungen		sehr häufig	niedrig
Hardware Informationsverarbeitung		selten	normal
Software Informationsverarbeitung		häufig	niedrig (Regelung) sehr hoch (Sequenz)
HMI (Human Machine Interface – Informationseingabe u.–ausgabe)		selten	normal

Beispielobjekt – Ventil (Regelventil)

- ein Objekt mit verschiedenen Aspekten (Sichtweisen, Variablen) und Funktionen
- objektorientierte Automatisierung



Objektorientierte Leitechnik [Ahrens u. a.]

- „Was man sich von einer objekt-orientierten Entwurfsmethodik jedoch versprechen kann, ist der einfachere Zugang zu komplexen Systemen (über ihre Objekte), der modulare Aufbau und die Wiederverwendbarkeit der Entwurfsergebnisse.“
- Elemente der oo Entwurfsmethode:
 - objekt-zentrierte Sichtweise
 - Abstraktion, Klassenbildung, Instanzbildung
 - Taxonomien und Vererbungsmechanismen
 - Aggregation
 - Assoziation
 - ...

Objektorientierte Leitechnik [Ahrens u. a.]

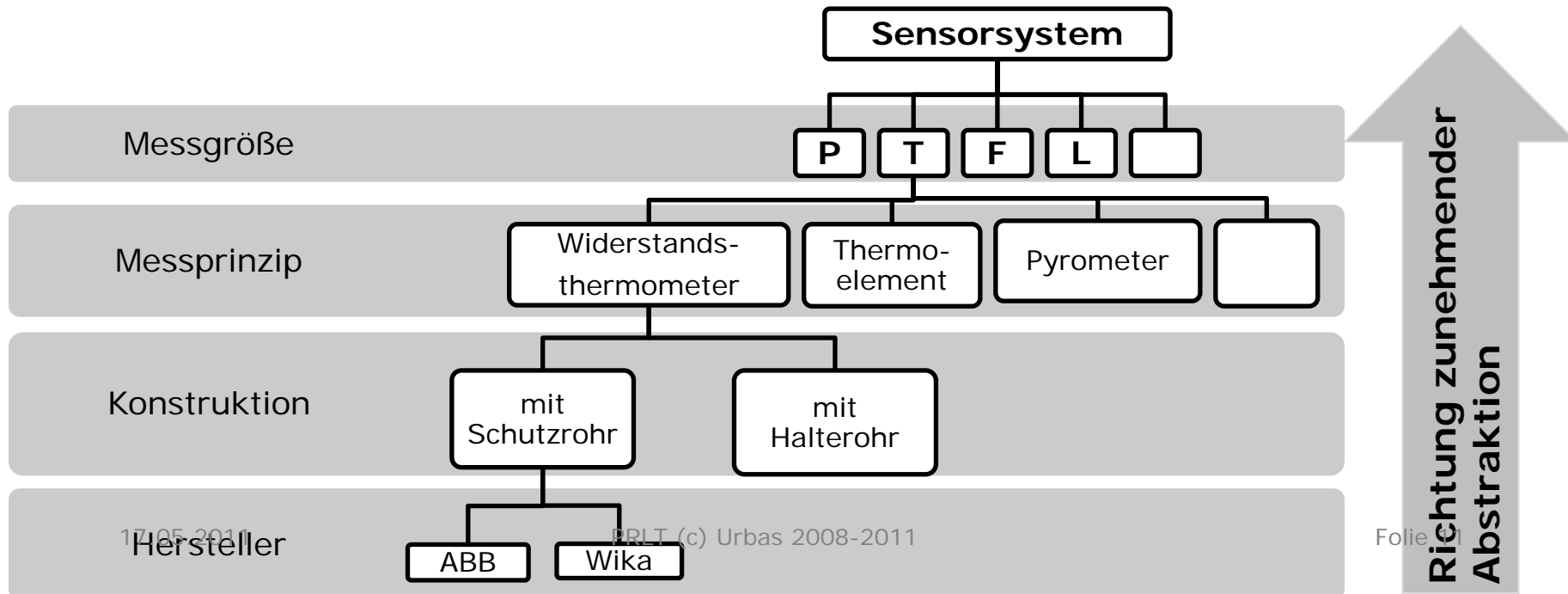
- objekt-zentrierte Sichtweise
 - Betrachtungsmittelpunkt sind nicht Funktionen der Anwendung, sondern die Objekte der Anwendungsdomäne
 - reale Objekte → Produkte, Geräte, Anlagen, Personen
 - Repräsentationen von Objekten → Pläne, Zeichnungen
 - Rollen von Objekten → Sicherheitsventil – Prozessventil
 - Ereignisse → Unfall, Meldung, Alarm, Fehler
 - Handlungsanweisung → Auftrag, Rezept, Stellwert
 - Konzepte → Ideen, Prinzipien

Objektorientierte Leitechnik [Ahrens u. a.]

- Abstraktion, Klassenbildung, Instanzbildung
 - Reduzierung der Komplexität der realen Welt durch Vernachlässigung von Details (Abstraktion) und Erkennung von Gemeinsamkeiten von Objekten, Situationen und Prozessen (Klassenbildung)
 - Individuen einer Klasse sind Instanzen (konkrete Objekte) und besitzen:
 - eine Identität (→ Kennzeichnungssystem)
 - eine oder mehrere Eigenschaften
 - Struktur (Zusammensetzung)
 - besitzen bestimmtes Verhalten (Funktionen)
 - haben eine Status

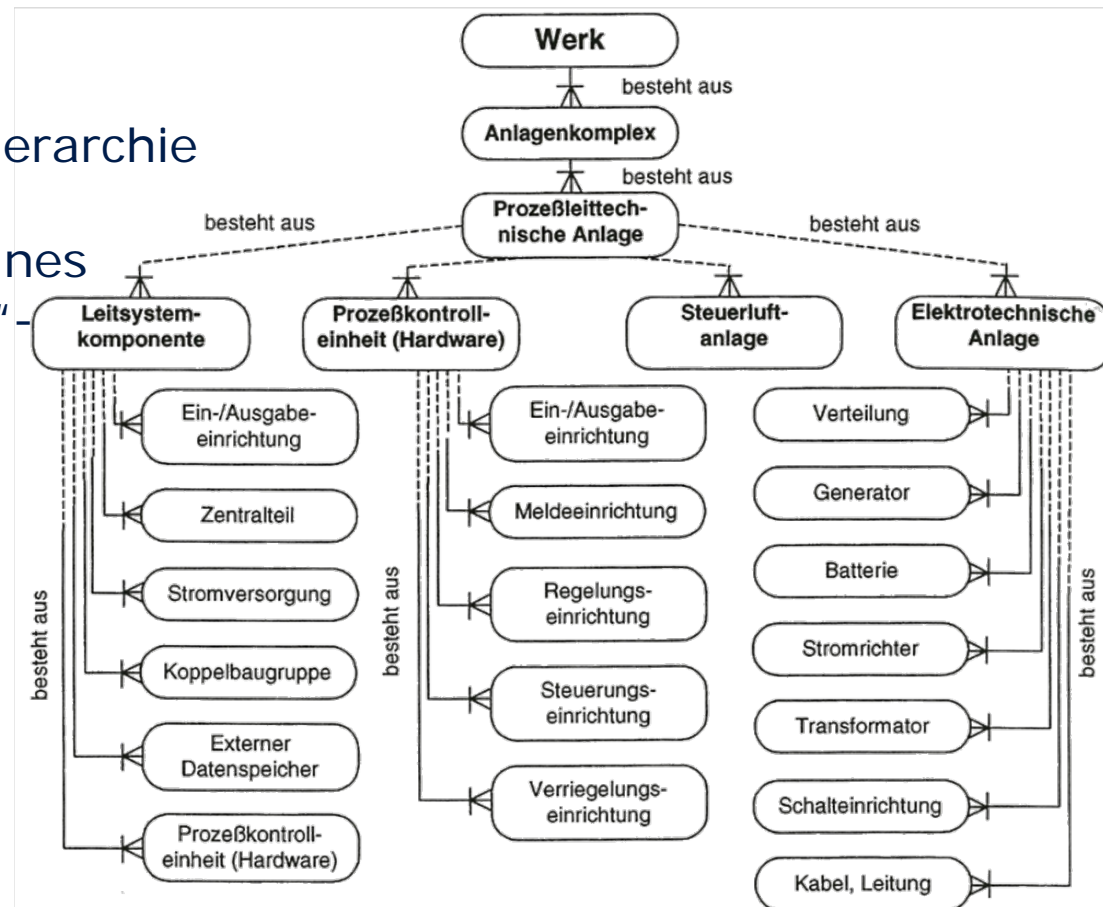
Objektorientierte Leittechnik [Ahrens u. a.]

- Taxonomien und Vererbungsmechanismen
 - Bildung von Klassenhierarchien durch Weglassen und Hinzufügen von Eigenschaften („is-a“-Hierarchie)
 - Abstraktionsrelation



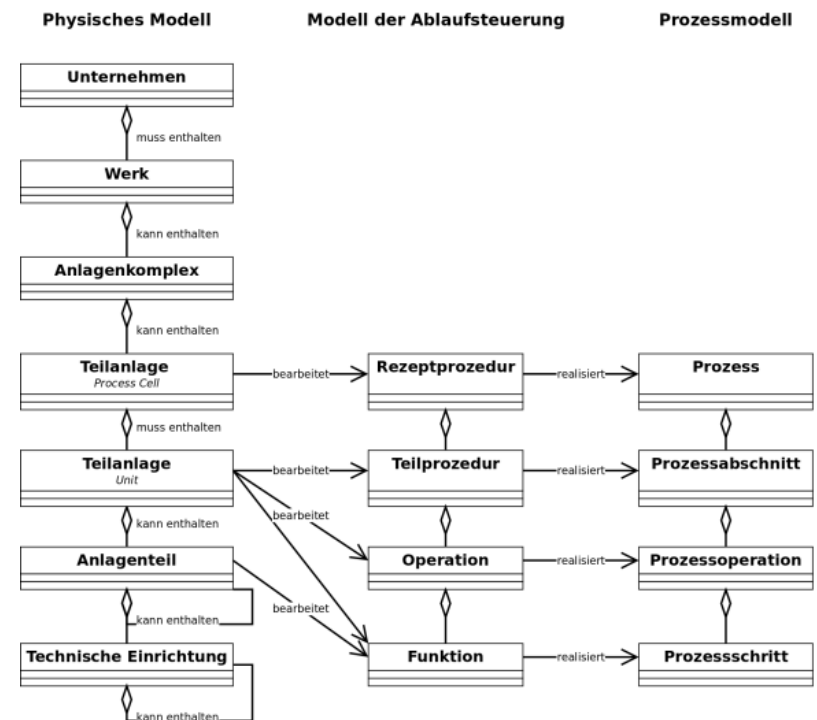
Objektorientierte Leittechnik [Ahrens u. a.]

- Aggregation
 - ist Teil-/Ganze Hierarchie
→ statische
Aufbaustruktur eines
System („part-of“
Beziehung)



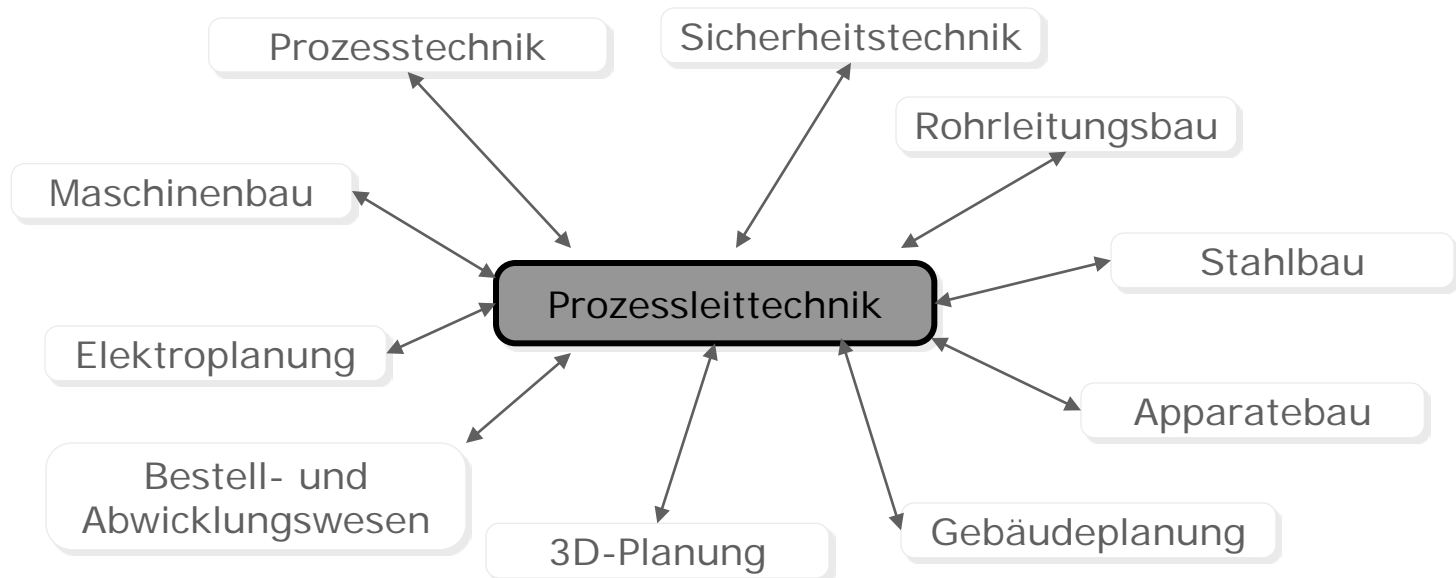
Objektorientierte Leittechnik

- IEC 61512 (ISA-88)
 - Prozessmodell: beschreibt VT Prozess als Hierarchie von Prozessabschnitten, operationen und schritten
 - > relationales Datenmodell mit vordefinierten Entitäten, Attributen und Relationen



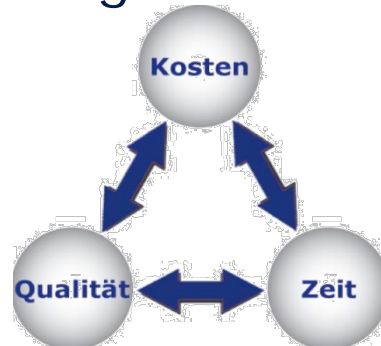
Herausforderungen an die Planung der Automatisierung

- starke Abhängigkeiten im Entwicklungsprozess mit anderen Gewerken (Verfahrenstechniker, Maschinenbauer,...)
 - klare Definition der Schnittstellen (Arbeitsteilung)
 - verschiedene Denk- und Sichtweisen



Herausforderungen an die Planung der Automatisierung

- Planungszeiten werden immer kürzer
 - die Qualitäts-, Sicherheits- und Umweltaanforderungen steigen stetig
 - Kostendruck wird immer stärker
- Effizienzsteigerung, d.h. Kosten- und Aufwandsminimierung bei gleichzeitiger Verbesserung der Arbeitsleistungen und Ergebnisse



Bewältigung der Herausforderungen

- objektorientiert Denkweise bei der Planung
 - häufige auftretende Objekte und Strukturen mittels Massenbearbeitungen verwalten und planen
 - Wiederverwendung bewährter Zusammenhänge und typischer Lösungen
 - Verringerung der Variantenvielfalt (Henry Ford: „Any customer can have a car painted any colour that he wants so long as it is black.“)
 - Instanziierung und Parametrierung von Klassenobjekten
- Entwicklung und Bereitstellung geeigneter Komponenten, die einen hohen Grad an Wiederverwendung erlauben → **Typicals**
- Automatisierung der Automatisierung

Typicals in der Prozessleittechnik

- Signalübertragung – Verdrahtung
- konstruktive Gestaltung (Aufbau; Montage) von Mess- und Stellgeräten, Schaltschränken und Verteilerkästen
- SIL-Nachweis und Ex-Berechnung
- Spannungsabfallberechnung
- Prüfprozeduren und –protokolle
- HMI-Darstellung im PLS (Fließbildsymbole, Faceplates)
- Informationsverarbeitung im PLS
 - typische Funktionsbausteine
 - typische Funktionsblockdiagramme (Funktionspläne)

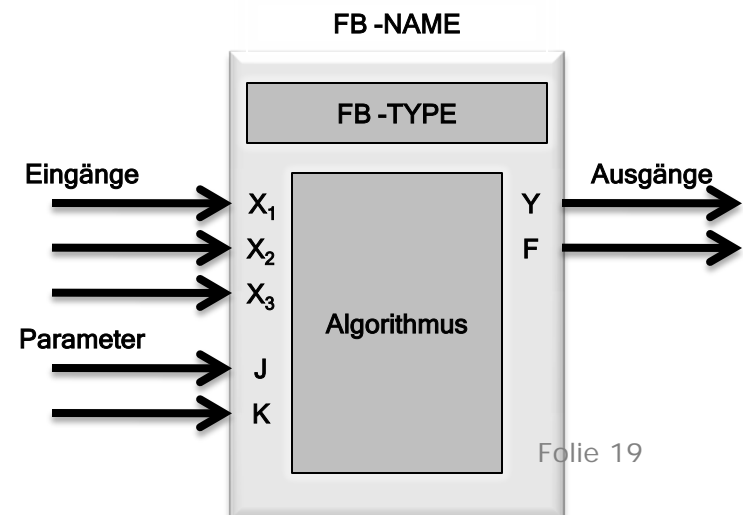


TECHNISCHE
UNIVERSITÄT
DRESDEN

Typicals von Funktionsbaustein

Typicals von Funktionsbaustein (FB)

- Anwender FB oder FB aus Spezialbibliotheken, die über die Standard FB (Zähler, Timer,...) hinausgehen
- Ziel: häufige Wiederverwendung im gesamten Programm → modulare Strukturierung
- Umsetzung mittels Programmiersprachen nach IEC 61131 (AWL, ST, FBS, KOP) bzw. FB-Verschachtelung
- Beispiele:
 1. Modellierung von Aktoren
 2. Analogwertverarbeitung
 3. Regler



Beispiel 1: Modellierung von Aktoren

- Individual Control Function (ICF) oder Einzelsteuerfunktion (ESF)
 - softwareseitige (PLS; SPS) Steuerung von häufig wiederkehrenden Aktortypen (Motoren und Ventile) und ihren verschiedenen Realisierungsvarianten
 - Umsetzung von Ansteuerungs-, Schutz-, Sicherheits- und B&B-Funktionen
 - Berücksichtigung von Verriegelungseingriffen (Schalt- bzw. Störsignale) und Betriebsartenumschaltungen (HAND – AUTO)

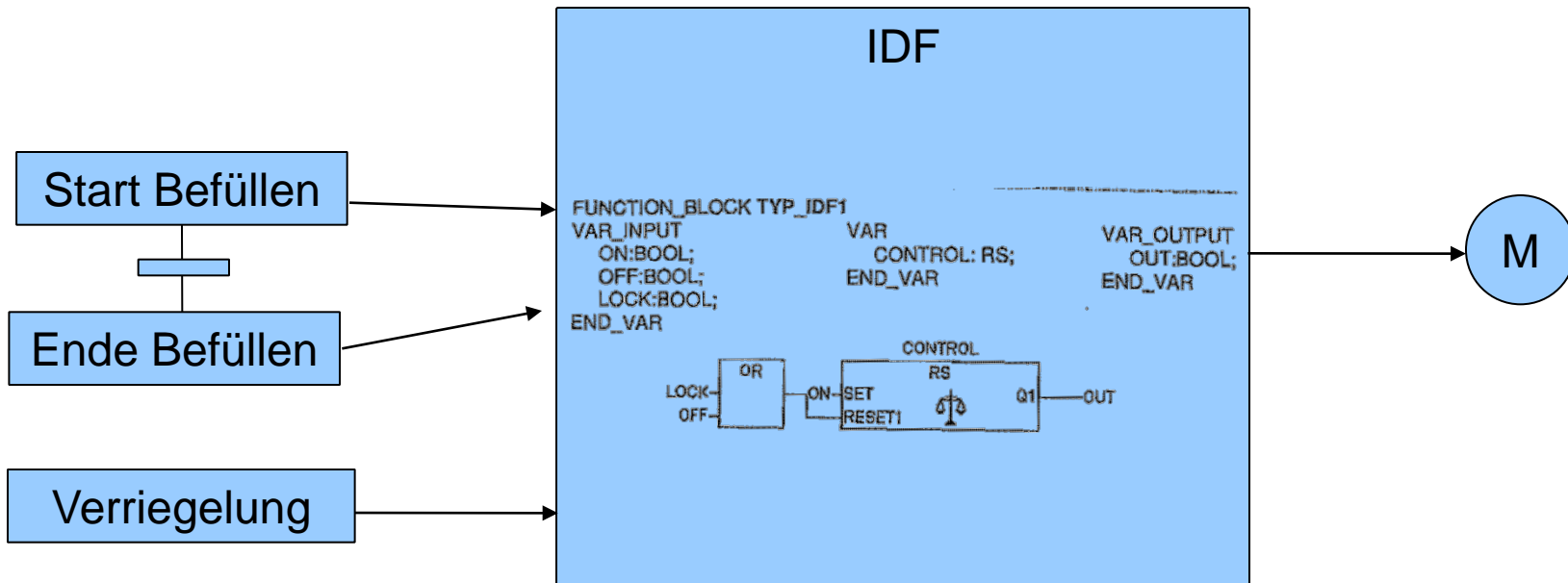
Typische Motorbausteine [Seitz2008]

- Motor mit einer Drehrichtung
 - Steuersignale: EIN, AUS ; Stellsignale: STELL
- Motor mit zwei Drehrichtungen
 - Steuersignale: LINKS, RECHTS, AUS; Stellsignale: STELL_L, STELL_R
- Schrittmotor
 - Steuersignale: LINKS, RECHTS, AUS, TAKTZEIT; Stellsignale: STELL, RECHTSLAUF, TAKT
- Polumschaltbarer Einrichtungsmotor
 - Steuersignale: SCHNELL, LANGSAM, AUS; Stellsignale: A_SCHNELL, A_LANGSAM
- Drehzahlveränderbarer Einrichtungsmotor mit Umrichter
 - Steuersignale: DREHZAL, EIN, AUS; Stellsignale: N_SOLL, STELL
- Drehzahlveränderbarer Zweirichtungsmotor mit Pulsweitenmodulation
 - Steuersignale: LINKS, RECHTS, AUS, PULS_PAUSEN; Stellsignale: TAKT, STELL_L, STELL_R

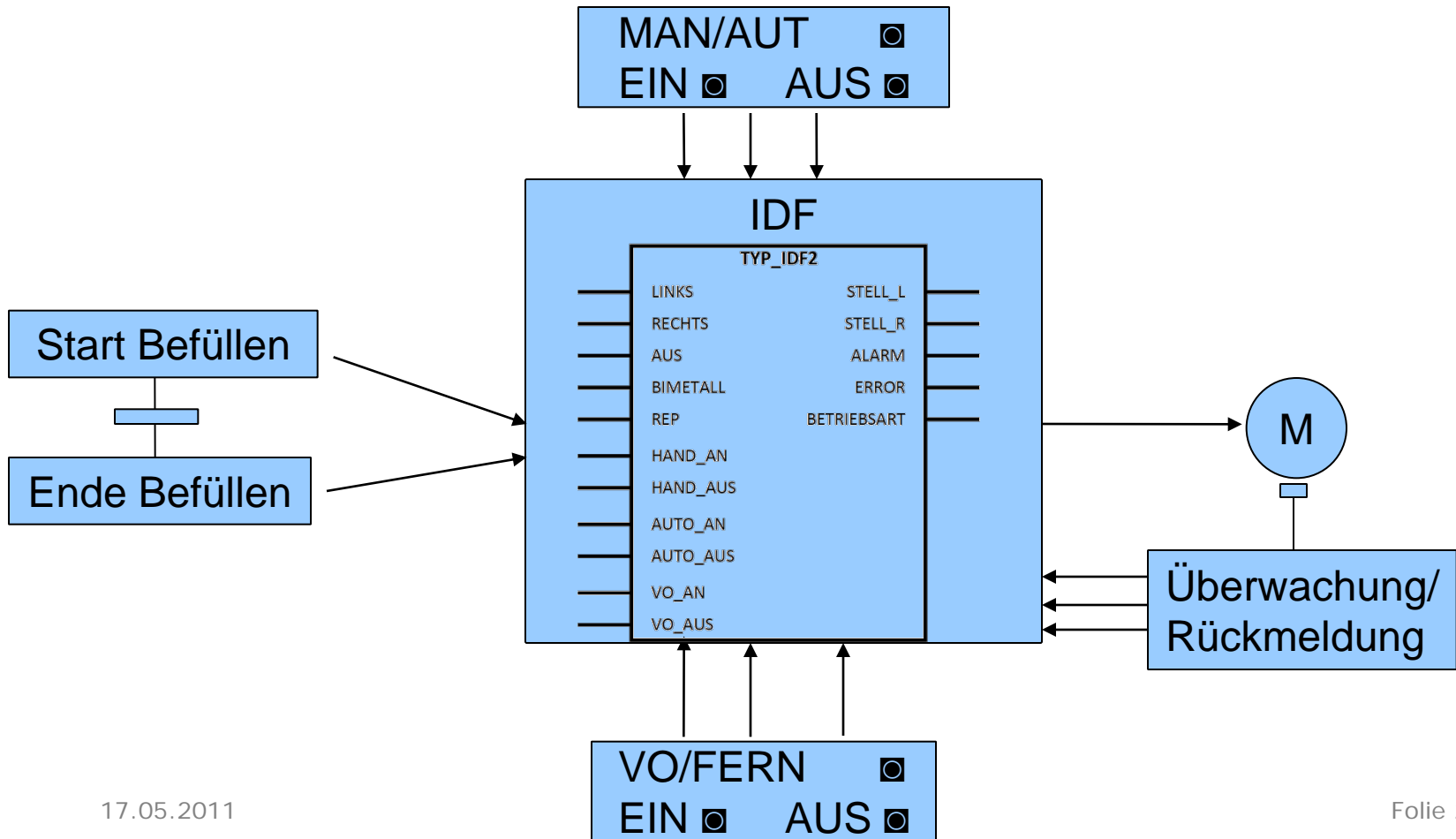
Einrichtungsmotor, IDF1

- IDF_Basic
 - ON, OFF → RS-Flipflop → OUT
 - Verriegelung / Überschreiben des Ein-Signals LOCK (Temporäres Ausschalten)
- IDF_Extended (Einbindung erweiterter Funktionen)
 - Betriebsartenumschaltung
 - Vorort-Bedienung, Manuell, Automatik, Außer Betrieb
 - Gleichberechtigung/Hierarchie abhängig von Betreiberphilosophie!
 - Lauf(richtungs)rückmeldung
 - Läuft, Geschwindigkeit, Stromaufnahme, ...

Beispiel: IDF_Basic Motor

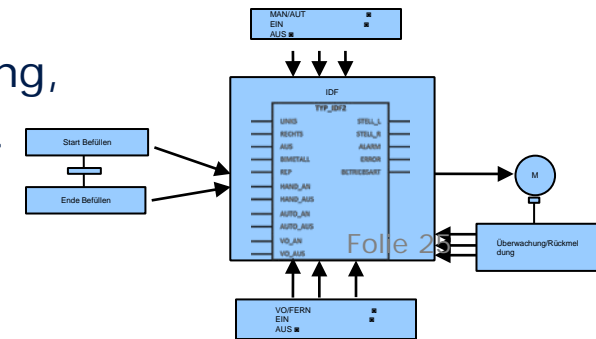


Beispiel: IDF_Extended Motor



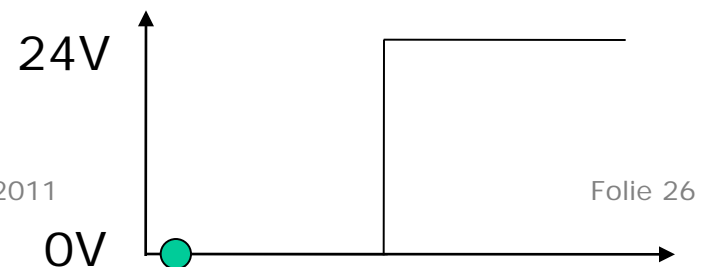
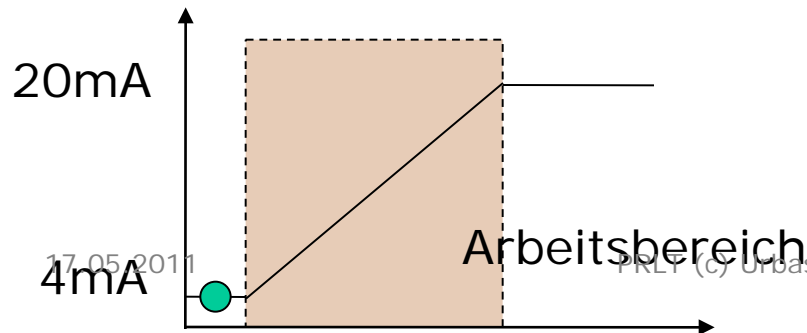
Betriebsartwahl (MANUEL – ORT – AUTOMATIK)

- Umschaltung zwischen 3 Betriebsarten (4 außer Betrieb) sollte für jede einzelne ICF möglich sein
 - (Außer Betrieb: Das Gerät ist nicht aktiv.)
 - Automatikbetrieb: Die Einzelsteuerfunktion wird von einem übergeordneten Programm automatisch angesteuert.
 - Handbetrieb: Die Einzelsteuerfunktion wird über eine Bediengraphik des Leitsystems direkt durch den Bediener angesteuert.
 - Vor-Ort-Betrieb: Der Bediener aktiviert das Gerät direkt vor Ort, zum Beispiel über ein Bedientableau.
 - Integration in den Baustein sinnvoll
- es darf nur eine Betriebsart aktiv sein und Hierarchien (Prioritäten) werden meist verlangt
 - z.B.: Die Vorort-Bedienung hat immer Vorrang, da Mitarbeiter Vorort „näher“ am Prozess ist.



Beispiel 2: Analogwertverarbeitung

- Physikalische Größen sind häufig kontinuierliche Größen: Druck, Temperatur, Geschwindigkeit, Drehzahl, pH-Wert, Abstand
 - Nutzung der analogen Messgröße im Programm (Regelung)
 - Generierung von binären Information aus analogen Messgrößen (Alarm- und Schaltwerte)
- Können von SPS nicht direkt verarbeitet werden, Ein/-Ausgabebaugruppen arbeiten nur mit elektrischen Signalen (Strom, Spannung) → **Messumformer**
- **Analoges Signal:** Kann innerhalb technischer **Grenzen** beliebige Werte annehmen.

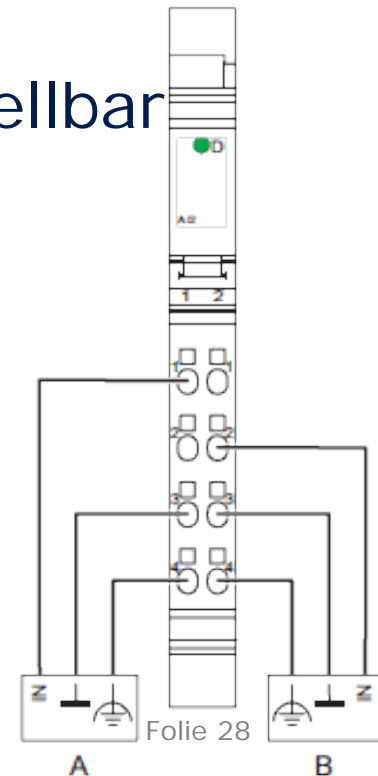


Analogwertdarstellung

- Umsetzung des analogen Prozesssignals in einen diskontinuierlich proportionalen Digitalwert mittel Analog-Digital-Umsetzer (ADU)
- Auflösung: Anzahl der Bits (8-15 & Vorzeichen)
- je nach Karte/Hersteller sind Auflösung, A/D-Zeit oder Störfrequenzunterdrückung parametrierbar

Messarten

- Verschiedene Messarten
 - Spannung, Strom (Widerstand, Temperatur)
 - verschiedene Messbereiche.
- Konfiguration herstellerabhängig einstellbar durch
 - Art der Verdrahtung,
 - Messbereichsmodul, und/oder
 - Parametrierung.



Messbereich	Stellung des Messbereichsmoduls
$\pm 1V$	A
$\pm 5V$	B
$\pm 10V$	B (Default)
1...5V	B

Messbereichsmodul (Siemens SM331)

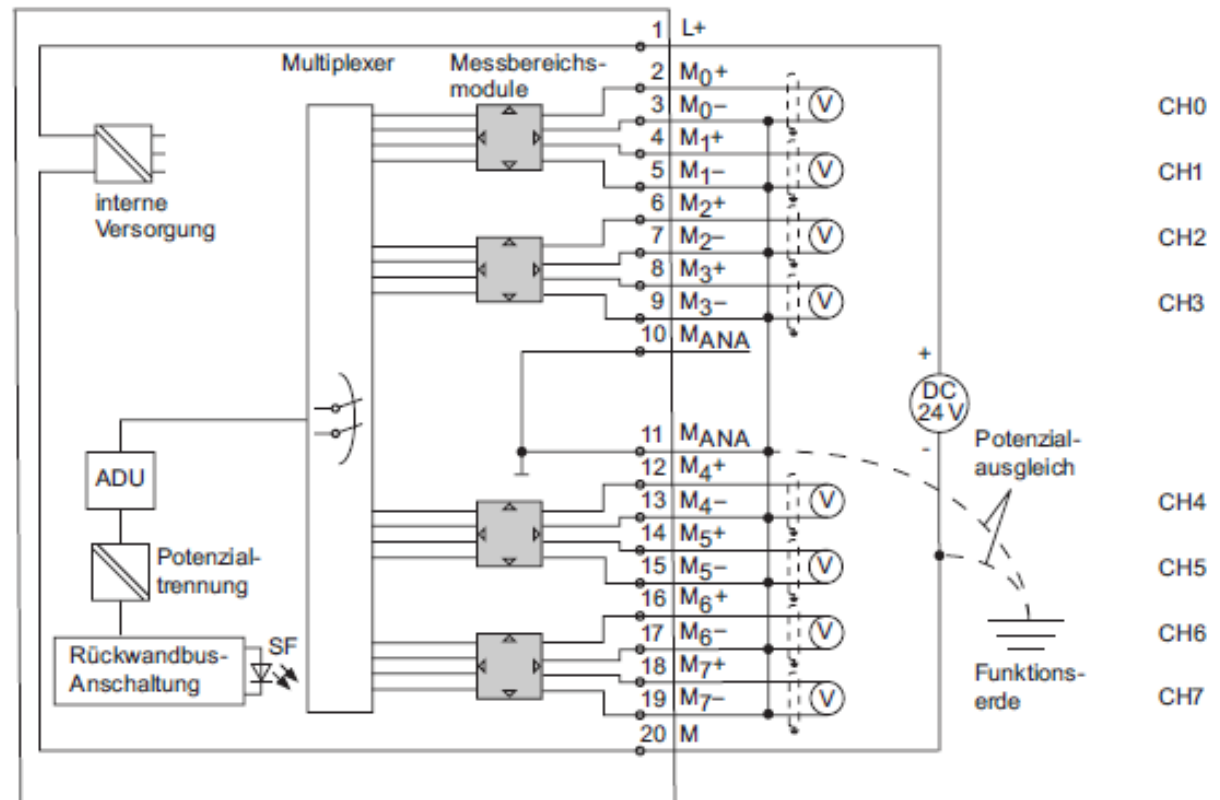


Bild 6-7 Prinzipschaltbild und Anschlussbild

Messbereiche Phoenix Contact IB IL AI 2/SF

Messbereich -20 mA bis +20 mA / -10 V bis +10 V

Eingangsdatenwort (Zweierkomplement)		-20 mA bis +20 mA I_{Eingang}	-10 V bis +10 V U_{Eingang}
hex	dez	mA	V
8001	Bereichs- überschreitung	> +21,6746	> +10,837
7F00	32512	+21,6746	+10,837
7530	30000	+20,0	+10,0
0001	1	+0,66667 μA	+333,33 μV
0000	-1	0	0
FFFF	0	-0,66667 μA	-333,33 μV
8AD0	-30000	-20,0	-10,0
8100	-32000	-21,6746	-10,837
8080	Bereichs- unterschreitung	< -21,6746	< -10,837

Code (hex)	Fehler
8001	Messbereich ve
8002	Drahtbruch
8004	Messwert ungüt
8010	Konfiguration ur
8040	Modul defekt
8080	Messbereich verlassen (unterschritten)

Messbereiche Siemens SM331

System		Spannungsmeßbereich				
dez.	hex.	±10 V	±5 V	±2,5 V	±1 V	
32767	7FFF	11,851 V	5,926 V	2,963 V	1,185 V	Überlauf
32512	7F00					
32511	7EFF	11,759 V	5,879 V	2,940 V	1,176 V	Übersteuerungsbereich
27649	6C01					
27648	6C00	10 V	5 V	2,5 V	1 V	Nennbereich
20736	5100	7,5 V	3,75 V	1,875 V	0,75 V	
1	1	361,7 µV	180,8 µV	90,4 µV	36,17 µV	
0	0	0 V	0 V	0 V	0 V	
-1	FFFF					
-20736	AF00	-7,5 V	-3,75 V	-1,875 V	-0,75 V	
-27648	9400	-10 V	-5 V	-2,5 V	-1 V	
-27649	93FF					
-32512	8100	-11,759 V	-5,879 V	-2,940 V	-1,176 V	Untersteuerungsbereich
-32513	80FF					Unterlauf
-32768	8000	-11,851 V	-5,926 V	-2,963 V	-1,185 V	

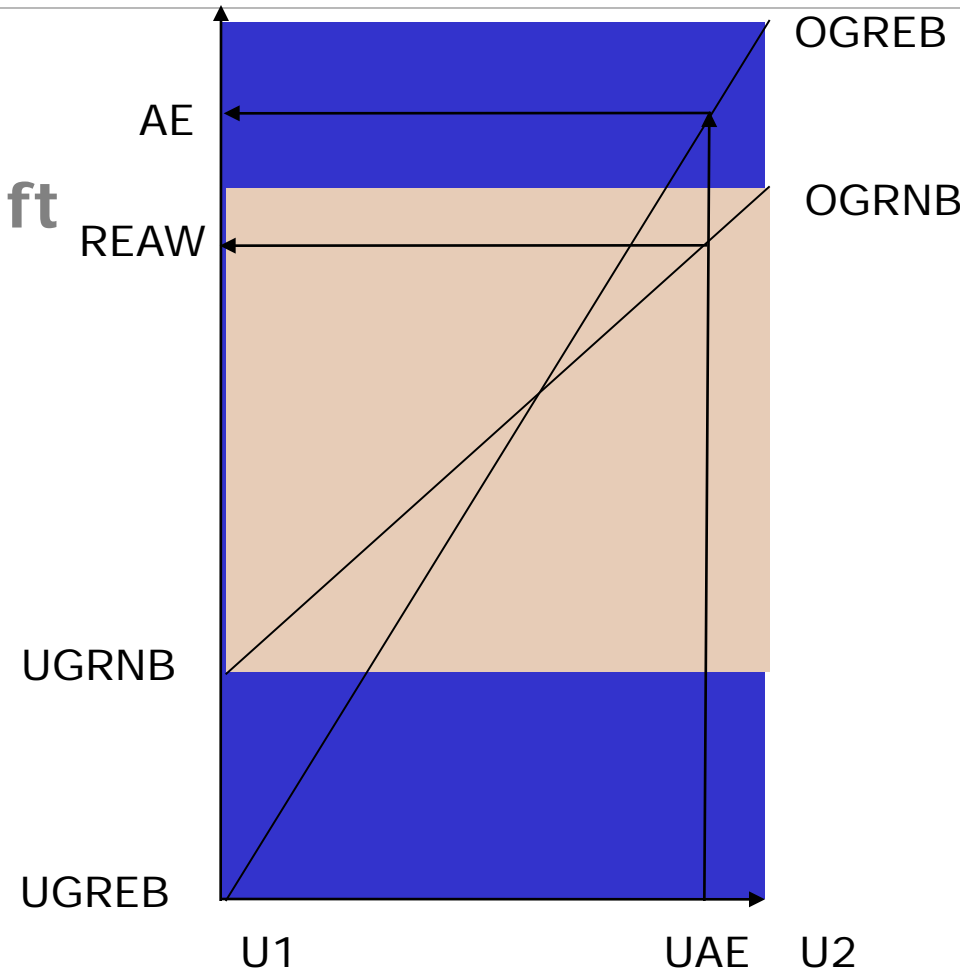
Analoge Messwerte einlesen und normieren

- Messwert steht als WORD im Prozessabbild
- Kann als 16-Bit Ganzzahl interpretiert werden
- Digitales Rohsignal, Normierung notwendig
- Beispiel: $AE_Norm = 500/27648 * AE_Nenn$

Physikalische Größe	Ausgang des Sensors	Digitaler Eingangsnennwert AE_Nenn	Normierter Wert AE_Norm
[m ³ /h]	[mA]		
0	4	0	0
500	20	27648	500

Allgemeine Normierungsvorschrift

- AE: Digitalisierter Eingangswert
- REAW: Normierter Analogeingabewert
- OGREB, UGREB: Grenzen Eingangsnennbereich
- OGRNB, UGRNB: Grenzen Normierungsbereich
- U1, U2: Spannungsmessbereich
- UAE: Anliegende Spannung



$$REAW = UGRNB + (AE - UGREB) \frac{\Delta GRNB}{\Delta GREB}$$

Realisierung als Funktion AEnorm in AWL (IL)

- Realisierung als IEC-61131 Funktion in AWL
- Ein-Address Maschine, alle binären Operationen ziehen verknüpfen den Akkumulator mit dem angegebenen Operanden und legen Verknüpfungsergebnis (VKE) dort wieder ab
- Sprache ist typsicher, d.h. nur Operatoren gleichen Typs können verknüpft werden
Wandlungsoperatoren *_TO_*

Benötigter Sprachumfang für diese AE

```
LD op (* vke := op *)
```

```
ST op (* op := vke *)
```

```
ADD op (* vke := vke + op *)
```

```
SUB op (* vke := vke - op *)
```

```
MUL op (* vke := vke * op *)
```

```
DIV op (* vke := vke / op *)
```

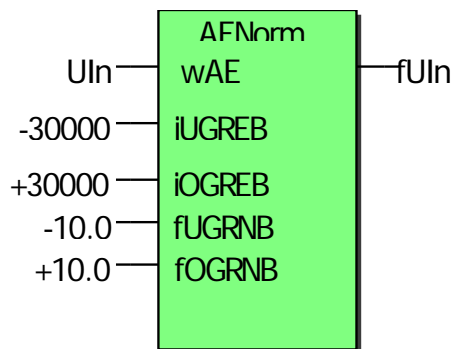
```
INT_TO_REAL (* vke :=  
float(vke) *)
```

```
WORD_TO_REAL (*vke   
float(vke) *)
```

AE ist als WORD Variable definiert

Normierungs- baustein in AWL

- Funktion (ohne output variablen)
- 3 lokale Variablen



17.05.2011

	Name	Typ	Verwendung	Beschreibung
Default				
	fUGRNB	REAL	VAR_INPUT	Untergrenze Normierungsber
	fOGRNB	REAL	VAR_INPUT	Obergrenze Normierungsber
	iOGREB	INT	VAR_INPUT	Obergrenze Eingangsnennbe
	iUGREB	INT	VAR_INPUT	Untergrenze Eingangsnennbe
	wAE	WORD	VAR_INPUT	Roher digitalisierter Analogw
	fDGRNB	REAL	VAR	Range Normierungsbereich
	fDGREB	REAL	VAR	Range Eingangs-Nennbereich
	fUGREB	REAL	VAR	Untergrenze Eingangs-Nennk

AENorm: AENorm - STD_CNF.STD_RES.FUNCTIONS.AENorm.AENorm				
1	LD iUGREB		(* Konvertiere iUGREB *)	
2	INT_TO_REAL			
3	ST fUGREB			
4				
5	LD iOGREB		(* Konvertiere iOGREB *)	
6	INT_TO_REAL			
7	SUB fUGREB		(* fDGREB <- (f)OGREB - fUGREB *)	
8	ST fDGREB			
9				
10	LD fOGRNB		(* fDGRNB <- fGRNBR - fGRNBR *)	
11	SUB fUGRNB			
12	ST fDGRNB			
13				
14			(* fUGRBNR + ((f)AE - fUGREB) * fDGNRB / fDGREB *)	
15	LD wAE		(* Konvertiere AE *)	
16	WORD_TO_REAL			
17	SUB fUGREB			
18	MUL fDGRNB			
19	DIV fDGREB			
20	ADD fUGRNB			
21	ST AENorm		(* Speichere Ergebnis (entspr. Return) *)	

Beispiel 3: weitere Typicals von FB - Regler

- kontinuierliche Regler (PID, PD)
- Split-Range-Regler
- Schaltende Regler (Zwei- und Dreipunktregler)

3	
SPLITR_P	OB35
Split ra	2/3
100.0 UHRANGE	QV1
0.0 ULRANGE	QV2
0.0 U	V1
50.0 NEUT_POS	V2
10.0 DEADB_W	
100.0 V1HRANGE	
0.0 V1LRANGE	
100.0 V2HRANGE	
0.0 V2LRANGE	

2	
CTRL_PID	OB35
PID Cont	2/2
0.0 DEADB_W	QSPEXTON
1.0 GAIN	QLMN_HLM
10.0 TN	QLMN_LLM
0.0 TV	QMAN_AUT
1.0 TM_LAG	QCAS_CUT
100.0 NM_LMNHR	LMN
0.0 NM_LMNL	QC_LMN
0.0 PV_IN	ER
16#80 QC_PV_IN	SP
100.0 NM_PVHR	
0.0 NM_PVLR	
0.0 DISV	
0 CSF	
No Track SP_TRK_0	
0.0 SP_EXT	
100.0 SPEXTHLM	
0.0 SPEXTLLM	
0 LIOP_INT	
0 SPEXON_L	
100.0 LMN_HLM	
0.0 LMN_LLM	
0.0 LMN_TRK	
0 LMN_SEL	
0 LIOP_MAN	
0 AUT_L	
0.0 LMNR_IN	
16#80 QC_LMNR	
16#80 QC_LMN_I	

Vom Typical zur Instanz

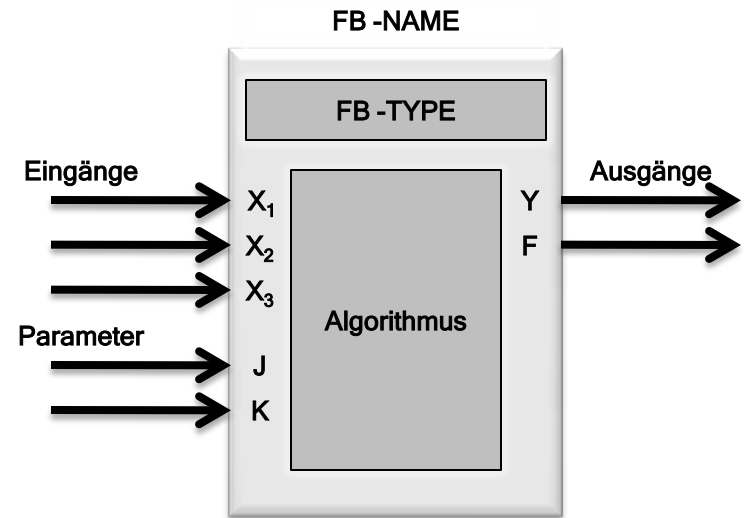
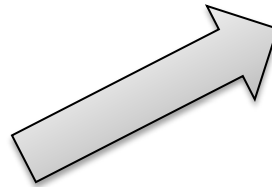
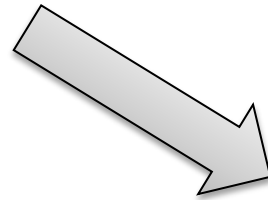
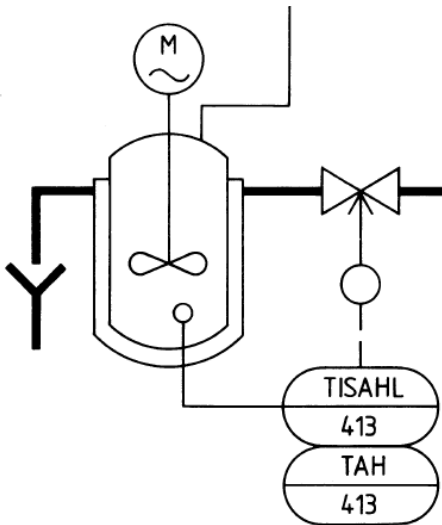
- Variante 1:
 - Analyse vorhandener Planungsunterlagen (z.B. R&I-Fließbild) und Identifizierung von wiederkehrenden Strukturen → Erstellung bzw. Zuordnung einer Funktionsbausteinklasse (Typical)
 - z.B.: alle Feldgerätetypen werden Typicals zugeordnet
 - Generierung der Funktionsbausteininstanzen aus Exportlisten der Planungsunterlagen (R&I-Fließbild) inkl. Typicalzuordnung
 - Parametrierung der Funktionsbausteininstanzen

Vom Typical zur Instanz

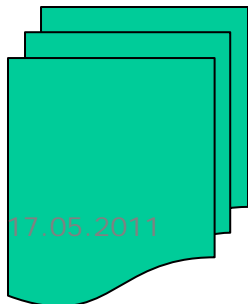
- Variante 2:
 - Entwicklung von Typickatalogen aus Erfahrungen von vergangenen Projekten (Kopplung: Funktionsbausteintypical \leftrightarrow Repräsentation auf Planungsunterlagen)
 - Erstellung der Planungsunterlagen (z.B. R&I-Fließbild) unter Verwendung des Typickatalogs
 - z.B.: alle Feldgerätetypen werden mit Typicals aus dem Katalog verarbeitet
 - Generierung und Parametrierung wie bei Variante 1

Vom Typical zur Instanz

Struktur
(Objekte,
Signale)



Parameter aus Spezifikationen
z.B.: Messbereich; Grenzwerte



17.05.2011

Zusammenfassung – Vorteile der strukturierten Automatisierung

- Verbesserung der Qualität der Planungsunterlagen
 - durch Wiederverwendung bewehrter und immer wiederkehrender ICFs lässt sich ein übergreifender „Standard“ etablieren
 - Test der ICF-Typicals zu frühen Planungszeitpunkten ermöglicht sehr frühe Fehlererkennung
- Verkürzung der Planungszeit
 - PLS-Softwaretest lässt sich extrem verkürzen, da nicht jede einzelne Funktion überprüft werden muss → Typicaltest (keine „Paste&Copy-Fehler“)
 - spätere Änderungen lassen sich auf Typicalebene durchführen

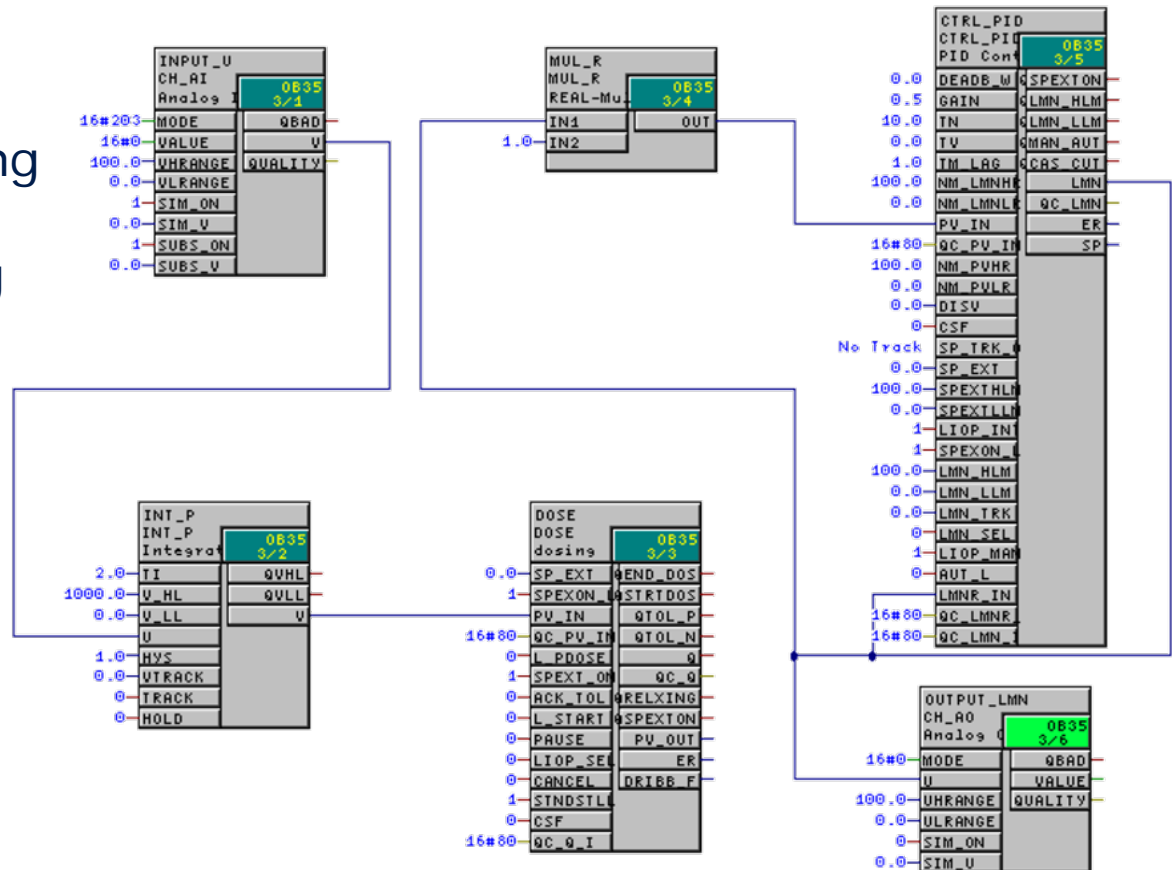


TECHNISCHE
UNIVERSITÄT
DRESDEN

Ausblick

Bildung typischer Funktionsplänen (CFC)

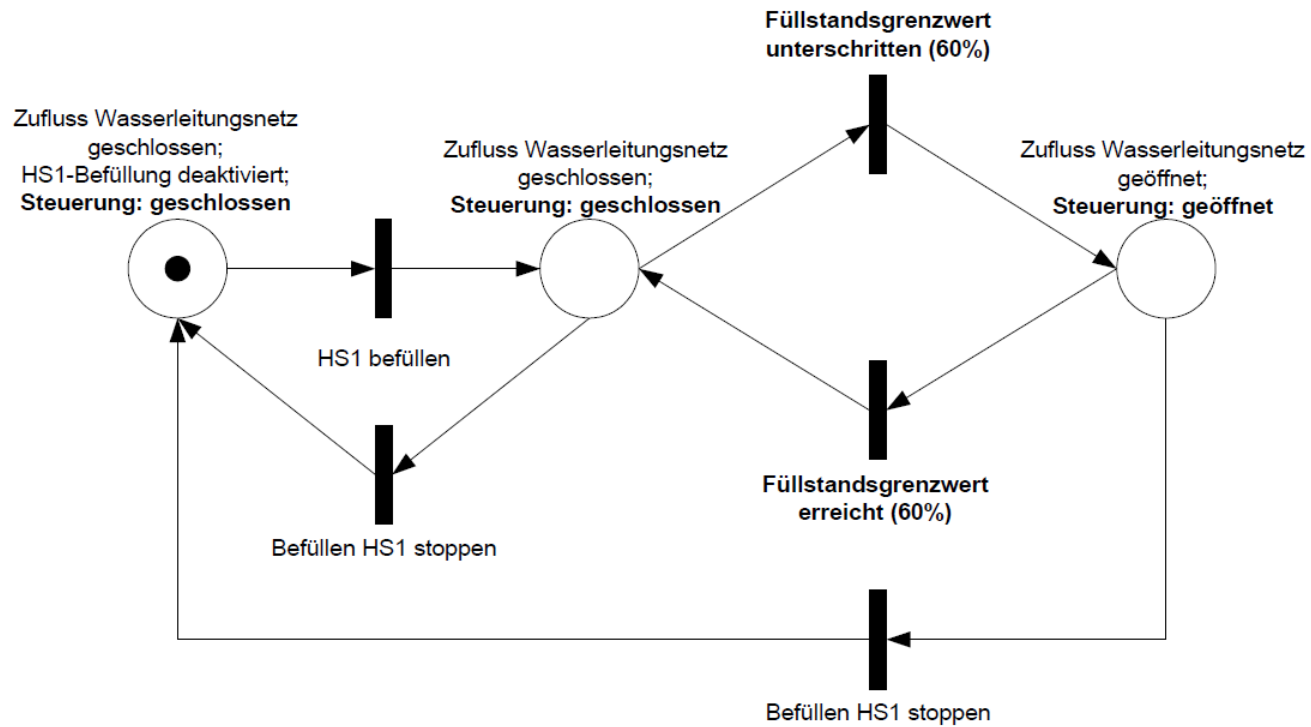
- z.B.: kompl. Regelkreis mit Messwertaufbereitung → Regler → Stellwertbegrenzung → Stellwertausgabe



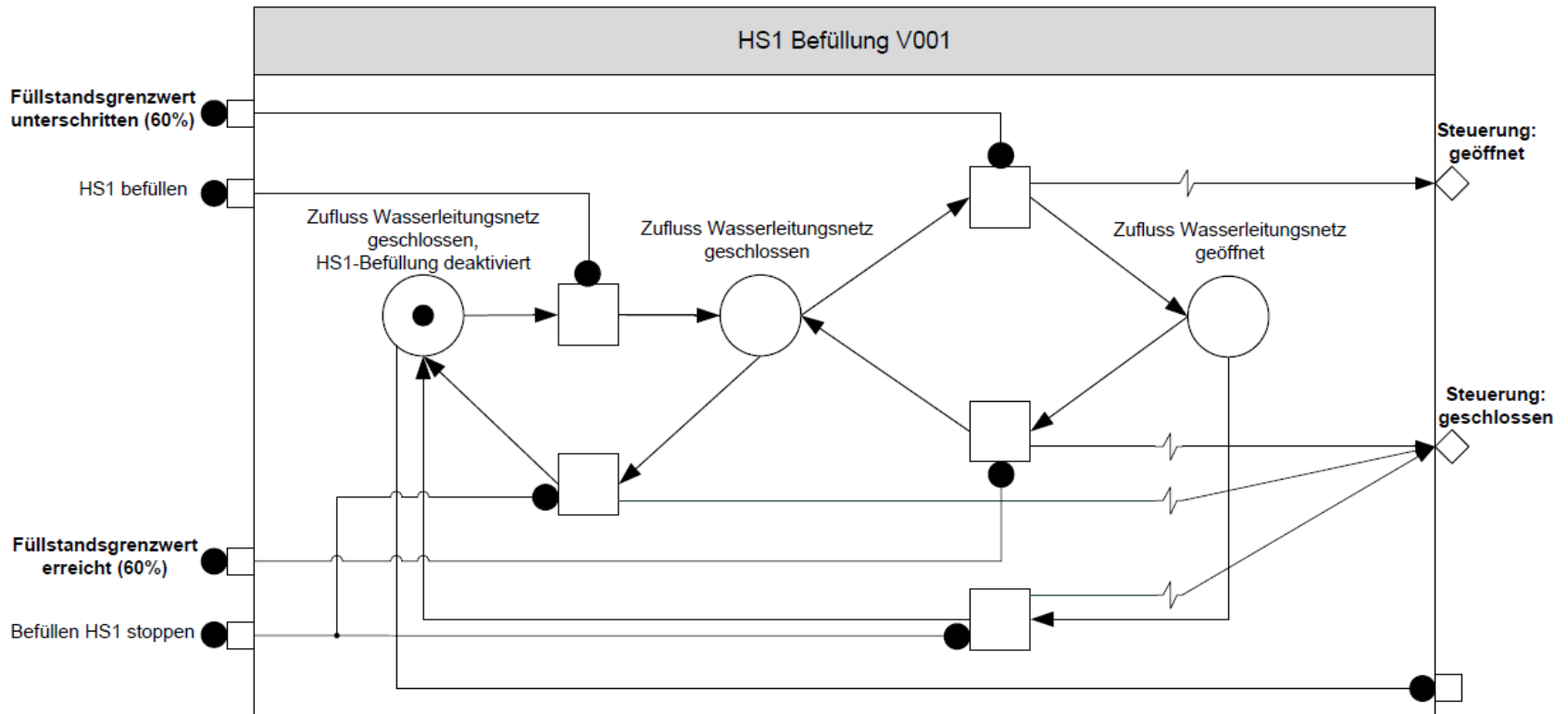
Moderne Beschreibungsmittel für den Steuerungsentwurf

- zustandsorientierte Beschreibungsmittel
 - Petri-Netze
 - Net Condition/Event Systeme (NCES)
 - Component Net (CNet)
 - GRAFCET
- objektorientierte Beschreibungsmittel
 - Unified Modeling Language (UML)
 - System Modeling Language (SysML)
 - Unified Modeling Language for Process Automation (UML-PA)
 - Formalisierte Prozessbeschreibungen VDI/VDE 3682

Beispiel - Petri-Netz

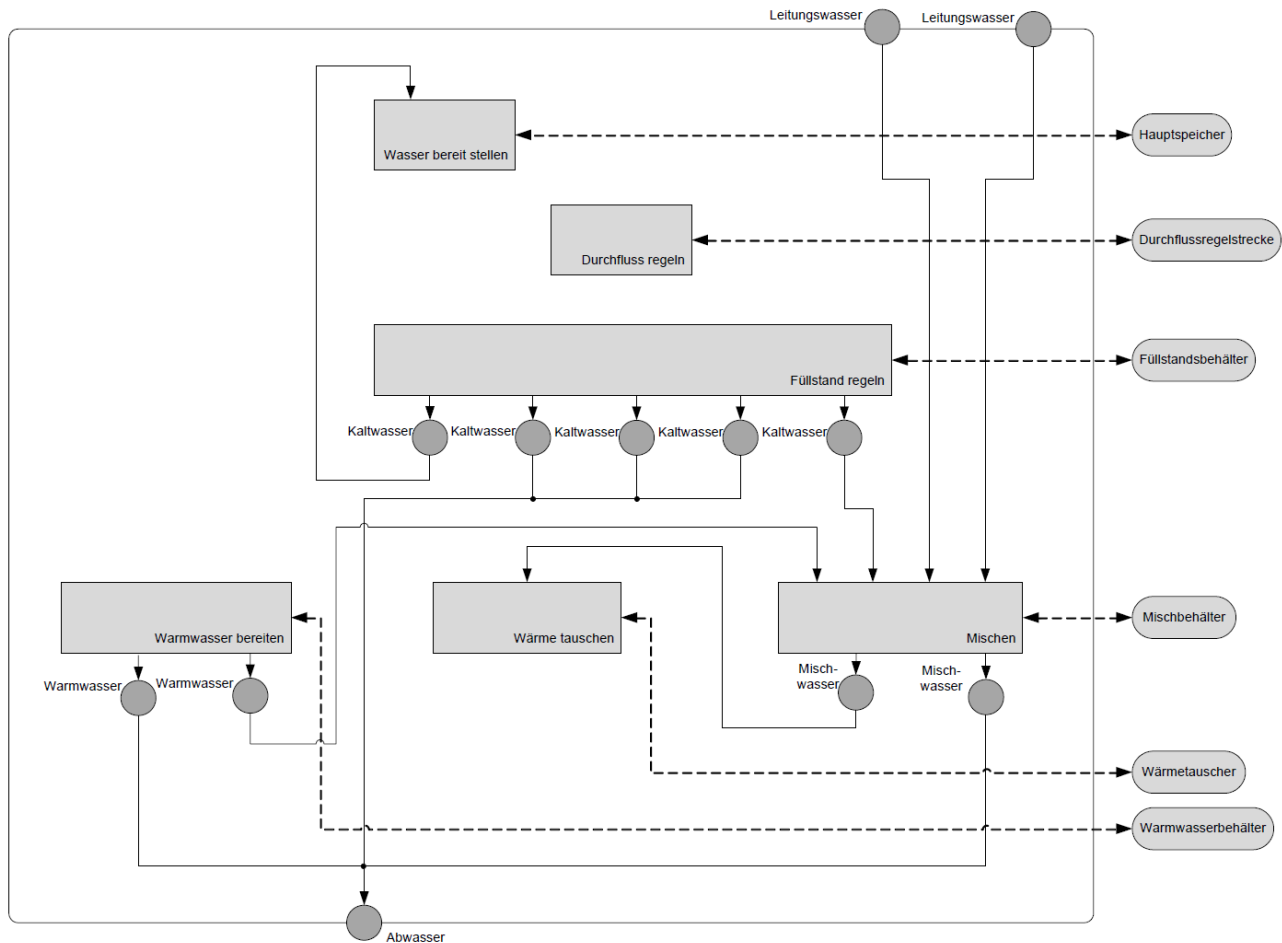


Beispiel - Net Condition/Event Systeme



Beispiel – VDI /VDE 3682

Formalisierte Prozessbeschreibung



Literatur

- [Ahrens u. a.] - Ahrens, W. ; Scheurlen, H.J. ; Spohr, G.U.: Informationsorientierte Leittechnik. München : Oldenbourg, 1997
- [Seitz2008] – Seitz, M.: Speicherprogrammierbare Steuerungen. München : Hanser, 2008
- [SIEMENS] SIMATIC Prozessleitsystem PCS 7 V7.0 SP1 PCS 7 - Getting Started Teil 1