

Echtzeitsysteme – Zusammenfassung SS97

Michael Aschke

Inhaltsverzeichnis:

1 EINLEITENDE ÜBERSICHT.....	1
1.1 DER BEGRIFF DES ECHTZEITSYSTEMS.....	1
1.1.1 Betrachtung des Sachverhalts.....	1
1.1.2 Zusammenhang mit dem System.....	2
1.1.3 Hierarchisch ereignisdiskretes System als Grundlage.....	2
1.1.4 Klassifizierung von Systemen.....	3
1.1.5 Echtzeitsysteme (Esys) und Echtzeitsystemtechnik (Esys).....	3
2 EREIGNISDISKRETE SYSTEME ALS GRUNDLAGE FÜR DAS SYSTEM-ENGINEERING ZUR MODELLIERUNG VON STRUKTUR UND VERHALTEN VON SACHVERHALTEN.....	3
2.1 VORGEHENSWEISEN (METHODENGEBÄUDE).....	3
2.2 MODELLIERUNGS- UND DARSTELLUNGSWEISEN DER ERGEBNISSE.....	3
3 EREIGNISDISKRETE SYSTEME ALS GRUNDLAGE FÜR DAS SYSTEM-ENGINEERING FÜR DEN FUNKTIONALEN ENTWURF VON ECHTZEITSYSTEMEN.....	4
3.1 GRUNDLAGEN FÜR DAS ENGINEERING.....	4
3.2 GRUNDLEGENDE BEGRIFFE.....	5
3.3 METHODENGEBÄUDE ZUM ENGINEERING.....	5
3.4 DARSTELLUNGSWEISEN FÜR ERGEBNISSE DES ENGINEERINGS FÜR DEN FUNKTIONALEN ENTWURF.....	5
4 SYSTEM-ENGINEERING FÜR DEN DIENSTORIENTIERTEN (OPERATIONALEN) ENTWURF VON ECHTZEITSYSTEMEN.....	6
4.1 GRUNDLAGEN, VORGEHENSWEISEN (METHODENGEBÄUDE) UND BEGRIFFE.....	6
4.2 MODELLIERUNG DER DIENSTKOOPERATION.....	6
4.3 FUNKTION UND DARSTELLUNG DER SYSTEMPLATTFORMDIENSTE (OSA++ FRAMEWORK).....	7
5 SYSTEM-ENGINEERING FÜR DIE PHYSIKALISCHE ARCHITEKTUR VON ECHTZEITSYSTEMEN...7	7
5.1 BEGRIFFE UND GRUNDLAGEN.....	7
5.2 ARCHITEKTUR DER SYSTEMPLATTFORMEN.....	8
5.3 SYSTEMPLATTFORMDIENSTE ZUR STRUKTURIERUNG VON SYSTEMPLATTFORMEN FÜR DIE REALISIERUNG VON AUTOMATISIERUNGSSYSTEMEN.....	8
6 DIE SYSTEMANALYSE VON PHYSIKALISCHEN, LINEAREN, KONTINUIERLICHEN DYNAMISCHEN ZU AUTOMATISIERENDEN SYSTEMEN.....	9
6.1 GRUNDLAGEN UND PRINZIPIEN ZUR BESCHREIBUNG.....	9
6.2 GRUNDLAGEN UND PRINZIPIEN ZUR BESCHREIBUNG DER SIGNALVERARBEITENDEN SYSTEME.....	9
6.3 GEWICHTSFUNKTION, SPRUNGFUNKTION UND ÜBERTRAGUNGSFUNKTION ZUR BESCHREIBUNG LINEARER SYSTEME.....	9
6.4 VERHALTEN VON MITTELS REIHENSCHALTUNG, PARALLELSCHALTUNG UND GEGENSCHALTUNG ZUSAMMENGESETZTER SYSTEME.....	10
6.5 BESCHREIBUNG DES SYSTEMVERHALTENS UND DER SIGNALE MIT HILFE DER FOURIER-TRANSFORMATION UND DER LAPLACE- TRANSFORMATION.....	10
7 MESSEN/REGELN/STELLEN IN DER KLASSISCHEN AUTOMATISIERUNGSTECHNIK.....	12
7.1 MESSEN.....	12
7.2 REGELUNGSTECHNIK.....	14
7.3 STELLEN.....	15

1 Einleitende Übersicht

1.1 Der Begriff des Echtzeitsystems

1.1.1 Betrachtung des Sachverhalts

- **Sachverhalt:** Reale u. abstrakte Dinge der Umwelt
- **Wahrnehmung:** Erkenntnisse über Sachverhalt
- **Objekt:** Gegenstand der Wahrnehmung
- **System:** Objektanordnungen (nach Regeln gebildet)

- **Modell:** Interpretierbare Repräsentation eines Systems

1.1.2 Zusammenhang mit dem System

- **System:** Kognitive, interpretierbare, nach Regeln gebildete Auffassung über Sachverhalt
- **System-Engineering:** Sachverhalte als System begreifen und damit umgehen
- **System-Technik:** Methodengebäude für das System-Engineering
- **System-Analyse:** System-Engineering zur Bildung von Systemauffassungen und -Modellen
- **Sicht:** Ausschnitt eines Sachverhalts
- **ECBS:** Engineering von Computer-basierten Systemen
- **Autonomes System:** nicht explizit eingebettet

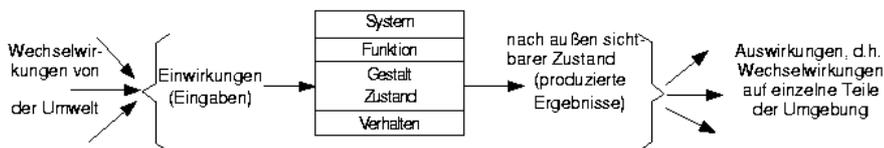
1.1.3 Hierarchisch ereignisdiskretes System als Grundlage

- **HDES:** zu automatisierende Systeme, automatisierte Systeme, automatisierende Echtzeitsysteme

1.1.3.1 System als Ganzes (Black-Box)

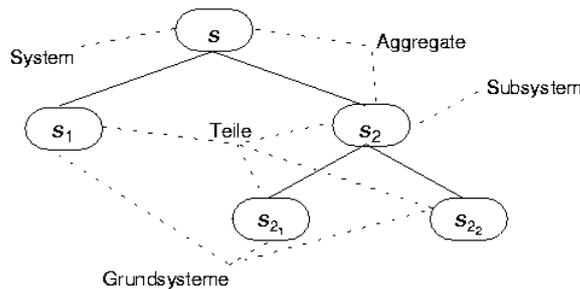
System wird in Beziehung mit nachfolgenden Eigenschaften/Objekten betrachtet:

- gleichbleibende Erscheinung (Gestalt)
- zeitinvariante Teile
- zustandsverändernde Einwirkungen



1.1.3.2 Systeme als Aggregate

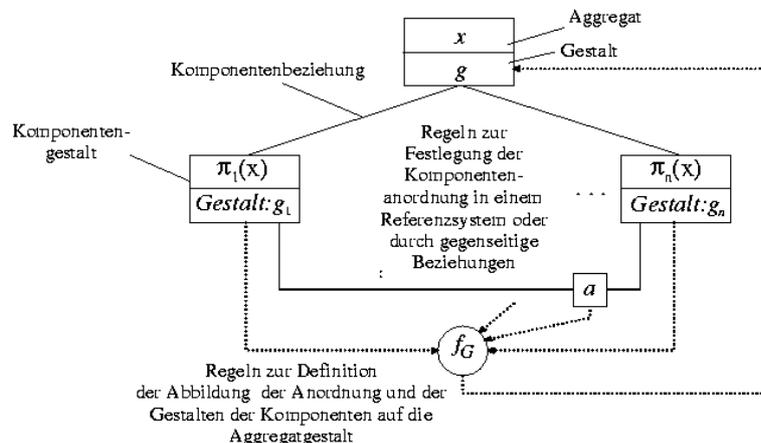
Systemhierarchie



Begriffe:

- **System:** ist nicht Komponente (Teil eines als Ganzes betrachteten, übergeordneten Systems)
- **Aggregat:** Gebilde, das selbst wieder Komponenten hat
- **Grundsystem:** "Blatt des Hierarchiebaumes" (quasi atomar)
- **Subsystem:** Aggregat, oder Grundsystem, das Teil eines Systems ist.

Systemstruktur (Architektur) eines Aggregats

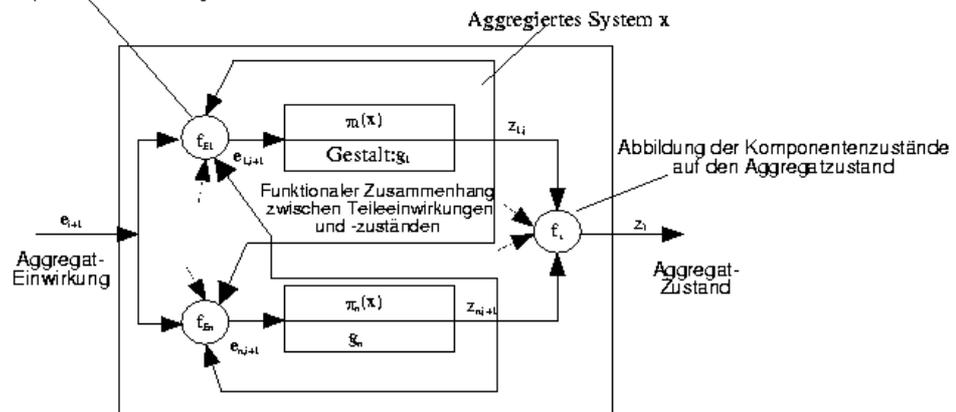


Mit diesen Begriffen, läßt sich eine "Blank-Box Betrachtung" durchführen

Verhalten eines Aggregats:

- funktionale Beziehungen
- Einwirkungen auf die Komponenten (Bild 1.6, echt1, Folie 22)

Abbildung der Aggregateinwirkung und der Komponentenzustände auf die Komponenteneinwirkungen



1.1.4 Klassifizierung von Systemen

- Echtzeitsysteme (Informatiksysteme, die zur Automatisierung schneller Prozesse dienen)
- Komplexe Systeme (tiefgestaffelte Hierarchie, rekursiv)
- Heterogene Systeme (die mit sonstigen Systemen zusammenwirken)

1.1.5 Echtzeitsysteme (Esys) und Echtzeitsystemtechnik (E syst)

Klassifikationsmerkmale:

- Informatiksysteme
- Esys lassen sich als HDES auffassen
- repräsentieren meist eingebettete Systeme
- dienen meist zur Automatisierung der Teile von technischen Anlagen
- Bei komplexen, hierarchisch geschichteten Automatisierungssystemen gehören nur prozeßnahe Systemteile zu den Echtzeitsystemen
- E syst umfaßt Methodengebäude (Vorgehensweise), Richtlinien (Ergebnisse), fertige Teilergebnisse (wiederverwendbare Komponenten)

2 Ereignisdiskrete Systeme als Grundlage für das System-Engineering zur Modellierung von Struktur und Verhalten von Sachverhalten

2.1 Vorgehensweisen (Methodengebäude)

- Systemanalyse
- Problemanalyse
- zugehörige automatisierende Systeme Echtzeitautomatisierungssysteme
- das Ganze automatisiertes System, Echtzeitanlage

2.2 Modellierungs- und Darstellungsweisen der Ergebnisse

- **Einwirkungen:** externe Ereignisse auf Systeme und Teilsysteme
- **Zustandsänderungen:** interne Ereignisse im System oder Aggregat
- funktionale Beziehung:

$$f_{sy}(einw_{i+1}, g, zu_i) = zu_{i+1} \quad \text{Gl.1}$$

$$f_v(einw, g, zu) = zu'$$

mit $einw \in \{einw \mid einw \text{ ist Element des Raums der Einwirkungsereignisse} \}$
 $zu, zu' \in \{zu \mid zu \text{ ist ein zugehöriges Element des Raums der Zustandsereignisse} \}$
 $g \in \{g \mid g \text{ ist Element des Raums der Gestaltsmerkmale} \}$

Gl.3

Modell der Funktionsspezifikation in Form eines HDES (White Box Betrachtung)

- Teilsysteme, deren Verhalten isoliert analysiert werden kann:

$$f_{xy:e}(e_{i+1}, z_{1,j_1}, \dots, z_{n,j_n}) = e_{s_{j_k+1}} \quad \text{Gl.5}$$

$$f_{xy:e}(\text{einw}_{xy}, z_{u_1}, \dots, z_{u_n}) = \text{einw}_{TS} \quad \text{Gl.6}$$
- Zustandsänderungen werden durch endliche Automaten definiert
- Auswirkung der Zustände:

$$f_{xy:z_u}(z_{u_1}, \dots, z_{u_n}, z_{\text{sgg},i}) = z_{\text{sgg},i+1}$$
- Systemklassenmodell Blockschaltbild
- Vollständige Darstellung erfordert explizite und implizite Definition der Funktionen
- Variablenbezeichner für Systeme durch reale Bezeichner ersetzen Entstehung von Blockschaltbildern für reale Systeme
- Wichtig ist die Interpretierbarkeit der Bezeichner
- Formales Modell:
 - $\text{mr}(s) = \langle\langle \text{OB}(s), \text{KB}(s), \text{VB}(s), \text{BB}(s), \text{FB}(s) \rangle, \text{SPEC}(\text{SIG}; \text{REG}) \rangle$
 - $\text{OB}(s)$ = Menge der Bezeichner für die Werte der Systeme, für Einwirkungen, Zustände, Gestaltungsparameter (OB = Objektbezeichner(menge))
 - $\text{KB}(s)$ = Klassenbezeichner, denen Werte (Objekte) angehören
 - $\text{VB}(s)$ = Variablenbezeichner
 - $\text{BB}(s)$ = Beziehungsbezeichner
 - $\text{FB}(s)$ = Funktionsbezeichner
- Gl 1–6 = Richtlinien für die Regelwerke

Skizze eines Regelwerks:

- Betrachtung des zugrundeliegenden Sachverhalts als Ganzes, sowohl Anlage als auch benötigtes Echtzeitsystem
- Je nach Komplexität Unterteilung des Systems
- Bei Dekomposition sollten Teile wieder selbst anschauliche Anlagenelemente repräsentieren
- Wenn immer möglich: Dekomposition auf bekannte und wiederverwendbare Teile
 - Falls nicht möglich: Entwurf von generalisierbaren Teilen, die sich vielfach wiederverwenden lassen
- Feststellung der systemgestaltenden, zeitinvarianten Merkmale, zeitinvarianten Zustände und Einwirkungen bei allen rekursiv gebildeten Teilen
- Funktionsspezifikation, funktionale Beziehungen festlegen
- Auf Zeitverhalten muß geachtet werden ... zwei Vorgehensweisen
 - Auswirkungen bei Funktionsspezifikation abschätzen und diese entsprechend gestalten (bei Erfahrungsmangel (bzgl des Systems) können nicht realisierbare Spezifikationen auftreten)
 - Vollständige Berücksichtigung der Einschränkungen und Aufnahme derselben in die Funktionsspezifikation (immer realisierbare Spezifikationen (bei umsichtiger Vorgehensweise))
- Funktionsspezifikation muß eindeutig und konsistent sein

3 Ereignisdiskrete Systeme als Grundlage für das System-Engineering für den funktionalen Entwurf von Echtzeitsystemen

3.1 Grundlagen für das Engineering

- Beim Engineering für den funktionalen Entwurf legt man das Verhalten in den erforderlichen Einzelheiten fest.
- Dabei muß ein Modell gebildet werden mit Teilen, die Teilautomatisierungsaufgaben lösen. Grundteile müssen als bekannt vorausgesetzt werden
- Meß-/Stell- und Regelsysteme werden a priori als bekannte Grundsysteme aufgefaßt
- Vorgehensweise zum Engineering beruht ebenfalls auf dem Konzept des HDES
 - **Erster Schritt:** Wenn bei Funktionsspezifikation noch keine Stellgrößen rekursive Erstellung eines vollständigen Modells durch fortlaufende Dekomposition
 - Grundsysteme: Stellsystem beinhaltet Stellglied und Stellgliedansteuerung, Meßsystem beinhaltet Sensor und Meßwerterfaßeinheit
 - Es wird vorausgesetzt, daß man entweder auf Zustandsereignisse sicher schließen kann, oder daß Meßsysteme existieren, aus welchen man die Zustände beobachten kann.
 - Das gewünschte Automatisierungssystem muß in Form des HDES dargestellt werden können
 - Nachprüfen, ob Werte der Zustandsänderung (erzwungen) mit den spezifischen Werten der Zustandsänderung übereinstimmen
 - Überprüfung während des Entwurfs Verifikation

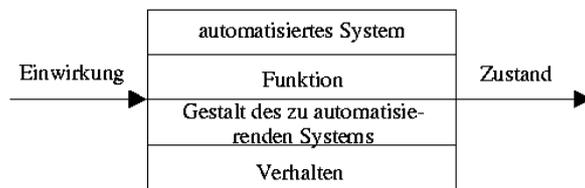
- Überprüfung während des Betriebs (maschinell) selbstständiger Diagnose
- Ablauf in 2 Schritten:
 - erster Schritt: Funktionsspezifikation, vollständiges Modell
 - zweiter Schritt: Einwirkungs-/Beobachtungsfunktionen Instanzen zuordnen.
- Das Instanzenmodell bildet das Ergebnis des Engineerings zur funktionalen Spezifikation
- Die Form des Instanzenmodells gemäß ISO/OSI-Schichtenmodell

3.2 Grundlegende Begriffe

- **steuerbar:** System, daß spezifische Zustände zu den damit assoziierten Zeitpunkten annimmt mit vorhandenen Stellgliedern zur Ansteuerung
- **beobachtbar:** wenn man entweder sicher auf die Zustandsgrößen schließen kann oder physikalische Größen messen und damit alle spezifizierten Zustände beobachten kann
- **Steuerung eines Systems: Wenn man** Steuerungsgrößen (Stellsignale, Stellglieder) nach dem Modell des automatisierten Systems bildet und physikalische Zustände nicht explizit mißt
- **Zustandsmessung und –Beobachtung eines Systems:** Wenn man bei Automatisierung nicht explizit mit Stellsignalen auf Stellglieder einwirkt, sondern nur Signale erfaßt und daraus auf Systemzustände zu assoziierten Zeitpunkten schließt
- **Regelung, (komplex, ereignisdiskret) rückgekoppelte Automatisierung:** Wenn man mit Stellsignalen auf Stellglieder einwirkt, Stellsignale nicht nur aufgrund der äußeren Einwirkung bildet, sondern auch aufgrund von erfaßten physikalischen und daraus beobachteten Zustandsgrößen ermittelt

3.3 Methodengebäude zum Engineering

Zur Anschauung:



- Bilde Modell (rekursiv)
- Modell stellt das "How" für das in Abschnitt 2 spezifizierte "What" dar
- Festlegung des Modells in Form des Instanzenmodells (auch Schichtenmodell genannt)

3.4 Darstellungsweisen für Ergebnisse des Engineerings für den funktionalen Entwurf

Form: Gl.3.1

$sy-zu_{i+1} = f_{eg}(sy-einw-er_{i+1}, g_{i+1}, zu_i) \leftrightarrow$ % Echtzeitsystemebene k
 % Eingang/Ausgangsbeziehung des automatisierten Systems (Spezifikation)

$tsyl-er_{i+1} = f_{teyl}(sy-einw-er_{i+1}, tsyl-zu_{i_1}, \dots, tsyn-zu_{i_n})$,
 % Einwirkungen auf das Teilsystem 1

$\wedge tsyl-zu_{i_1} = f_{teyl}(tsyl-er_{i_1}, g_{i_1}, tsyl-zu_{i_1})$
 % Ein-/Ausgangsbeziehung des Teilsystems 1

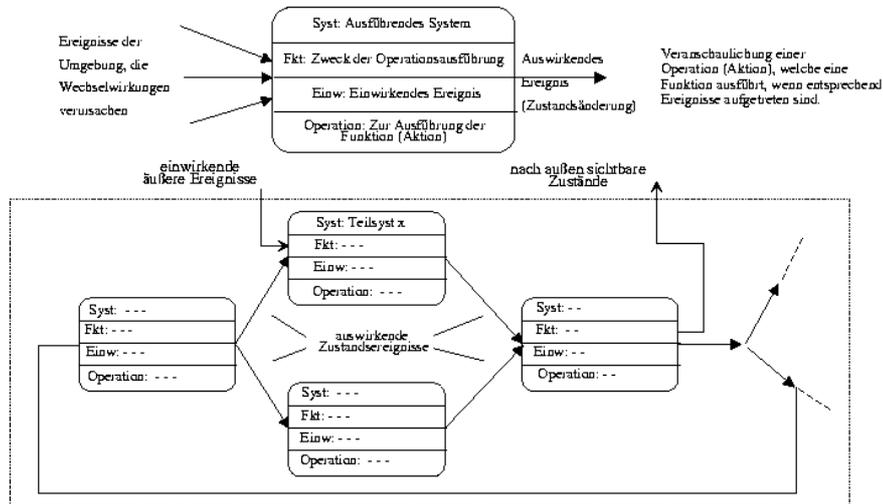
^ - - - - -
 $\wedge tsyn-er_{i_n+1} = f_{teyn}(sy-einw-er_{i+1}, tsyl-zu_{i_1}, \dots, tsyn-zu_{i_n})$
 % Einwirkungen auf das Teilsystem n

$\wedge tsyn_{i_n} = f_{teyn}(tsyn-er_{i_n+1}, g_{i_n}, tsyn-zu_{i_n})$
 % Eingangs/Ausgangsbeziehung des Teilsystems n

$\wedge sy-zu_{i+1} = f_{eg}(sy-zu_{i_1}, tsyl-zu_{i_1+1}, \dots, tsyn-zu_{i_n+1})$
 % Abbildung der Komponentenzustände auf den Systemzustand, d.h. auf den durch das Ereignis $sy-einw_{i+1}$ implizit bewirkte Zustand $sy-zu_{i+1}$. Hierdurch muß die Spezifikation erfüllt werden.

- Prädikatenkalkül erster Stufe
- Bei Instanzenmodell:
 - Rekursiv absteigend Sollvorgaben, Bildung der Stellsignale
 - Rekursiv aufsteigend Beobachtung der Aggregate und des Systemzustands
 - Signatur [mr(s)] muß um die Bezeichner der Instanzen erweitert werden

- Graphische Darstellung des funktionalen Ablaufs: Abb 3.6



4 System-Engineering für den dienstorientierten (operationalen) Entwurf von Echtzeitsystemen

4.1 Grundlagen, Vorgehensweisen (Methodengebäude) und Begriffe

- Hier "What" nicht "How"
- Zweckmäßig wird eine Zwischenphase zwischen Entwurf und Transformation des funktionalen Modells, welche **dienstorientier Entwurf** heißt und zu einem **Dienstmodell** führt.
- Transformation (Realisierung mit informationstechnischen Mitteln) des funktionalen Modells in ein Dienstmodell
- **Dienst:** abstrakte informationstechnische Einheit nach Vorschrift, kann mit anderen Diensten kooperieren
- Zum Dienst gehört eine **Ablage** für **Aufträge** (auch an untergeordnete Dienste) und **Ergebnisse**
- **Dienststeuerung:**
 - **Start**
 - **Unterbrechung**
 - **Fortführung**
 - **Beendigung**
- **Betriebsparameter**
- **Betriebsunterbrechung:** Auftrag, der in Bearbeitung ist, wird zuendegeführt, dann der Betrieb unterbrochen
- **Dienstbeendigung:** Alle bis dahin übergebenen Aufträge werden noch bearbeitet (vgl. Xerox)

Vorgehensweisen:

- Einzelnen Instanzen werden Dienste zur Ausführung zugeordnet
- Einem Dienst obliegt damit die Ausführung der Funktion
- Grundautomatisierungsinstanzen meist: Meß-/Stell- oder Regelsysteme werden meist autonom arbeitenden Diensten zugeordnet.
- Dienstbeauftragung, Ergebnismeldung
 - Ergebnisse werden entweder in der Ablage des auftraggebenden Dienstes hinterlegt, oder verbleiben in der Ablage des ausführenden Dienstes (bei letzterem: aktiver Zugriff darauf von anderen Diensten möglich)

4.2 Modellierung der Dienstkooperation

- Systemplattformdienste = Dienste zur Ausführung der Dienstkooperation
- Dienstarchitektur = Dienstmodell, welches präzise angibt, wie das spezifische, funktionale Verhalten auszuführen ist und Systemplattform, welche dieses Dienstmodell ausführen kann.
- Festlegung der Dienstarchitektur setzt Definition des Systemplattformdienste und Regeln zum Aufbau einer Dienstarchitektur voraus.
- OSA++ (Open System Architecture)
- CORBA (Common Object Request Broker Architecture) (von der OMG (Object Management Group) festgelegt) [Aussagen über einzuhaltende Funktionalität (aus Flexibilitätsgründen)]
- CORBA ist im Gegensatz zu OSA++ nicht für Echtzeitanforderungen gedacht.

4.3 Funktion und Darstellung der Systemplattformdienste (OSA++ Framework)

- zur Dienststeuerung
 - $\text{btr-steu}(di, \text{btr-inf}) = \text{btr-def}$ $\text{btr-def} \in \{\text{True}, \text{False}\}$
 - $\text{di-start}(di, \text{btr-inf}) = \text{start}$, $\text{start} \in \{\text{True}, \text{False}\}$
 - $\text{di-unterbr}(di) = \text{unterbr}$, $\text{unterbr} \in \{\text{True}, \text{False}\}$
 - $\text{di-fortf}(di) = \text{fortf}$, $\text{fortf} \in \{\text{True}, \text{False}\}$
 - $\text{di-beendg}(di) = \text{beendg}$, $\text{beendg} \in \{\text{True}, \text{False}\}$
- Informationsfunktion
 - $\text{betr-inf}(di) = \text{di-betr-inf}$
- Explizite Darstellung der Beauftragung
 - $\text{auftr-ert}(di, \text{auftr-inf}) = \text{vorg}$
- Der gesamte Prozeß für die Ausführung des Auftrags wird **Vorgang** genannt
- Explizite Ergebnismeldung
 - $\text{erg-meld}(di, \text{vorg}, \text{erg-inf}) = \text{vorg-abschl} \in \{\text{True}, \text{False}\}$
- Dienstkooperation
 - $\text{ert-auftr/erg}(di, \text{auftr-inf}) = \text{erg-inf}$
- Flexible Modellierung der Kooperation
 - $\text{erw-atr}(di) = \text{auftr-inf}$
- Tätigkeit solange unterbrechen, bis ein durch einen Vorgang gekennzeichnetes Ergebnis eintrifft
 - $\text{erw-erg}(di, \text{vorg}) = \text{erg-inf}$
- Modellierung (dienstablauf (Befehle beziehen sich auf Ablage))
 - $\text{exist-atr}(di, \text{auftr-inf}) = x$ $x \in \{\text{True}, \text{False}\}$
 - $\text{exist-erg}(di, \text{vorg}) = x$ $x \in \{\text{True}, \text{False}\}$
- Ablage
 - $\text{lese}(di, \text{cond}, \text{erg-inf}) = \text{inf}$
 - Angabe der Position
 - $\text{pos} \in \{\text{Erster}, \text{Letzter}, \text{Vorletzter}, \text{Nächster}\}$
 - Erster = zuerst erfolgter Eintrag
 - Nächstes = trivial
 - Letztes = zuletzt erfolgter Eintrag
 - Vorletztes = vorher erfolgter Eintrag
 - Zahl n, über die Mittelwert zu bilden ist
 - $\text{mittelwert} \in \langle n, \text{pos} \rangle$
 - n gemessene Werte, im Intervall $x_1 \geq x > x_2$
 - $\text{wertrelation} \in \{x_1, x_2, n\}$
 - Überprüfung, ob Ergebnis mit Wert übereinstimmt
 - $x = x_1$
 - $\text{schreibe}(di, \text{inf}, \text{pos}) = x$ $x \in \{\text{True}, \text{False}\}$
 - $\text{pos} \in \text{IN} \cup \{\text{zt} \mid \text{zt} = \text{auszuführender Zeitpunkt}\}$

5 System-Engineering für die physikalische Architektur von Echtzeitsystemen

5.1 Begriffe und Grundlagen

- **Systementwurf:** Transformation des dienstorientierten Modells in ein physikalisches Modell, Dienstbausteine festlegen und diese physikalischen, informationsverarbeitenden Einheiten zuzuordnen (Systemplattformen zu strukturieren)
- Es ist möglich einer Plattform mehrere Dienstbausteine zuzuordnen (paralleles Arbeiten)
- Systemplattformen müssen in beliebiger Weise miteinander vernetzt werden können
- Dienstbausteine müssen allozierbar und so die Systemplattformen strukturierbar sein
- Die Allokierung erfordert entweder eine feste Verdrahtung des Speichers, oder läuft programmgesteuert über einen programmierbaren Speicher (wird heutzutage meist gemacht)
- Im Speicher muß eine dienstspezifische Ablage eingerichtet werden

Vorgehensweise zur Bildung eines Modells für die physikalische Automatisierungsarchitektur

- erster Schritt: Festlegung der Systemplattformarchitektur aufgrund der Anforderungen der auszuführenden Dienste und nichtfunktionaler Anforderungen (z.B. Verteilung der Plattformen)
 - Dabei sind die Architekturen bzgl. Peripherie und Prozessor zu beachten, sowie die Art der Vernetzung

- zweiter Schritt: Festlegung der Dienstbausteine und Dienste und Bestimmung, auf welchen Systemplattformen die Dienstbausteine zu allokiert sind
- dritter Schritt: Erstellung einer Vorschrift, nach der nacheinander das Automatisierungssystem aufgebaut und strukturiert wird
 - heute: struktureller Aufbau meist softwaregesteuert

5.2 Architektur der Systemplattformen

- Nachrichtenaustausch über Standardschnittstellen (z.B. TCP/IP)
- Bei Echtzeitanforderungen ist ein entsprechendes Betriebssystem vonnöten
- Wichtig: Physikalische Schnittstellen, Eigenschaften der Betriebssystemebene und der Prozessorleistung
- Die Systemplattformarchitektur entspricht im wesentlichen dem ISO/OSI Schichtenmodell.
 - Nachrichtentransportschicht (Ebene 4)
 - Systemplattformschicht (Ebenen 5–7)
 - Schicht 7 nur nach außen zugänglich
- Alle Dienste, die auf einer Systemplattform konfiguriert sind, kommunizieren über diese. Die Systemplattform gewährleistet die Unabhängigkeit von Betriebssystem und Prozessoren
- Ein Echtzeitsystem ist **offen**, wenn es möglich ist, jederzeit zusätzliche Dienste hinzuzufügen oder wegzunehmen bzw. zu ändern
- OSA++ erlaubt im Gegensatz zu CORBA–ORB die Verwendung verschiedener Nachrichtentransportdienste (nicht nur TCP/IP). Beim Einsatz dieser muß auf die Echtzeiteigenschaften geachtet werden. Wenn Echtzeitanforderungen vorliegen, werden RS485 Schnittstellen oder Feldbusse eingesetzt, sonst Standard–Netzwerke

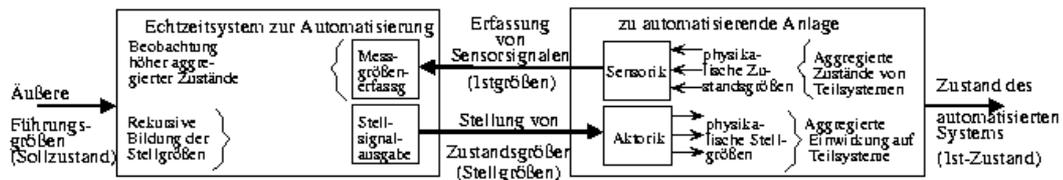
5.3 Systemplattfordienste zur Strukturierung von Systemplattformen für die Realisierung von Automatisierungssystemen

- Dienstbausteinallokierung (plf = Systemplattform)
 - create-service-brick(plf,brick-inf)=brick
 - delete-service-brick(plf,brick) ∈ {True, False}
- Einrichten von Diensten
 - config-service(brick,serv-inform)=service
 - deconfig-service(brick,service) ∈ {True, False}
- Verknüpfung von Diensten
 - define serv-access(serv,acc-inf)=acc-port (acc-port = Zugang/rechte)
 - define message-port(serv,message-def)=m-port (m-port Zugangsstelle für Ergebnisse)
- Dienste initialisieren
 - init-serv(serv,init-inform) ∈ {True, False}
 - start-serv(serv) ∈ {True, False}
 - stop-serv(serv) ∈ {True, False}
 - continue-serv(serv) ∈ {True, False}
 - end-serv(serv) ∈ {True, False}
- Dienstkooperation während des Betriebs (wie in Kap 4, nur auf "angelsächsisch")
 - place-order(serv,order-inf)=ref
 - await-order(serv,order-inf)=ref
 - send-result(serv,ref,result-inf) ∈ {True, False}
 - await-result(serv,ref)=result-inf
 - exist-order(serv,ref)=order-inf
 - exist-result(serv,ref)=result-inf
 - write-order-inf(serv,order-inf) ∈ {True, False}
 - read-result-inf(serv-cond)=result-inf

6 Die Systemanalyse von physikalischen, linearen, kontinuierlichen dynamischen zu automatisierenden Systemen

6.1 Grundlagen und Prinzipien zur Beschreibung

- Ist-Wert soll dem geforderten Sollwert nachgeführt werden (Abb. 6.1 echt6, Folie 5)



- Problemanalyse
- Systemanalyse

Lagrange'sche Zustandsfunktion

- $L(q', q) = I(q', q) - P(q)$
 - q' und q = generalisierte Koordinaten, welche den Energiezustand in einem Referenzsystem beschreiben und deren zeitliche Ableitungen
 - I = kinetische Energie, P = potentielle Energie
- Gl.6.2 beinhaltet die zeitliche Änderung der Energiezufuhr:

6.2 Grundlagen und Prinzipien zur Beschreibung der signalverarbeitenden Systeme

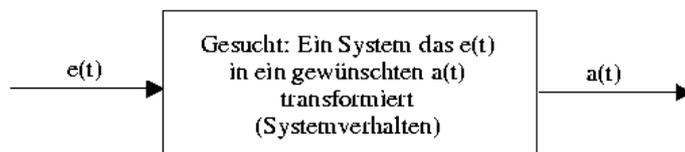
Der Begriff des Signals:

- Die rechte Seite der Differentialgleichungen, d.h. Kräfte, Momente und andere physikalische Größen sind Eingangssignale. Die sich durch Integration ergebenden physikalischen Größen stellen die Ausgangssignale dar

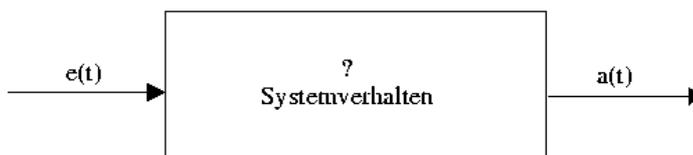
6.3 Gewichtsfunktion, Sprungfunktion und Übertragungsfunktion zur Beschreibung linearer Systeme

Prinzipielle Fragen:

- Systemsynthese (Abb. 6.8)

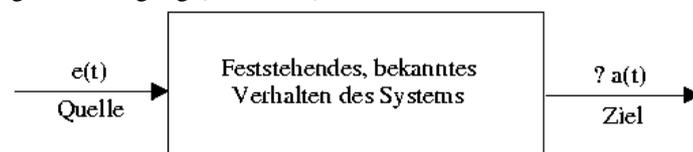


- Systemanalyse (Abb. 6.9)



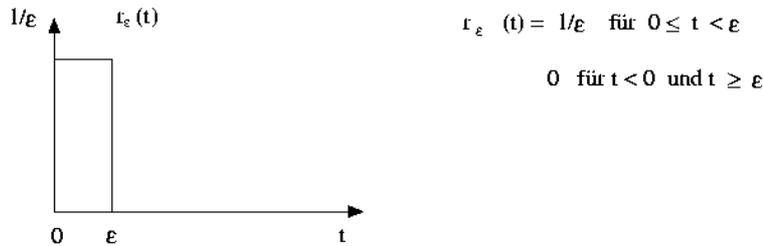
- Kenntnis wichtig, wie Eingangssignale bei der Übertragung verformt werden

- Signalübertragung (Abb. 6.10)

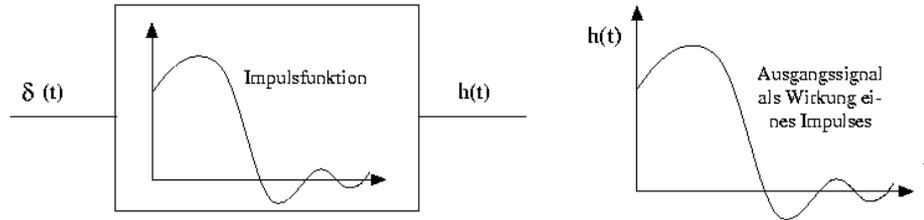


Impulsantwort von linearen kontinuierlichen dynamischen Systemen

- Abb 6.11 (Was ist eine Impulsfunktion?)



- Abb 6.12 (Was ist eine Impulsantwort?)



- Die dem Impuls innewohnende Energie ist per definitionem 1
- Voraussetzung für die Verknüpfung jedes Eingangssignals $e(t)$ mit dem dazugehörigen Ausgangssignal ist das Prinzip der Superposition (setzt Linearität der Systeme voraus)
- Faltungsintegral:
- Sprungfunktion (Amplitude nimmt konstant den Wert 1 an)

6.4 Verhalten von mittels Reihenschaltung, Parallelschaltung und Gegenschaltung zusammengesetzter Systeme

Hintereinanderschaltung

- Abb 6.15

$$a(t) = \int_{-\infty}^t h_2(t-\tau_1) \int_{-\infty}^{\tau_1} h_1(\tau_1-\tau_2) x(\tau_2) d\tau_2 d\tau_1$$

$$= \int_{-\infty}^t \int_{-\infty}^{\tau_1} h_2(t-\tau_1) h_1(\tau_1-\tau_2) x(\tau_2) d\tau_2 d\tau_1$$

Parallelschaltung

- Abb 6.16:

$$a(t) = \int_0^t h_1(t-\tau) e(\tau) d\tau + \int_0^t h_2(t-\tau) e(\tau) d\tau$$

Gegenschaltung

- Abb 6.17

$$a(t) = \int_{-\infty}^t h_1(t-\tau_1) e(\tau_1) d\tau_1 - \int_{-\infty}^t \int_{-\infty}^{\tau_1} h_1(t-\tau_1) h_2(\tau_1-\tau_2) a(\tau_2) d\tau_2 d\tau_1$$

6.5 Beschreibung des Systemverhaltens und der Signale mit Hilfe der Fourier-Transformation und der Laplace-Transformation

- $e(t) = e^{j\omega t}$ mit $\omega = 2\pi f$
- Fourier-Transformation der Impulsantwort:

$$H(j\omega) = \int_{-\infty}^{+\infty} h(\tau) e^{-j\omega\tau} d\tau$$

- Hierbei gilt:

$$A(j\omega) = \int_{-\infty}^{+\infty} a(t) e^{-j\omega t} dt$$

$$E(j\omega) = \int_{-\infty}^{+\infty} e(t) e^{-j\omega t} dt$$

$$H(j\omega) = \int_{-\infty}^{+\infty} h(t) e^{-j\omega t} dt$$

- Einfache Beschreibung des Eingangs-/Ausgangsverhalten:

$$\frac{A(j\omega)}{E(j\omega)} = H(j\omega)$$

- Bei instabilen Systemen zeigt $a(t)$ exponentiell anklingende Schwingungen auf. In diesem Fall ist die Beschreibung mit der Fourier-Transformation nicht mehr möglich, da sie bei $t \rightarrow \infty$ nicht konvergiert.

- In diesen Fällen wird links und rechts mit einem exponentiell abklingenden Signal $e^{j\omega - p t}$ multipliziert, so daß alle Schwingungen mit $t \rightarrow \infty$ gegen den Wert 0 konvergieren. Dann: $s = j\omega + p$

- $A(s), E(s), H(s)$ repräsentieren die Laplace-Transformierten:

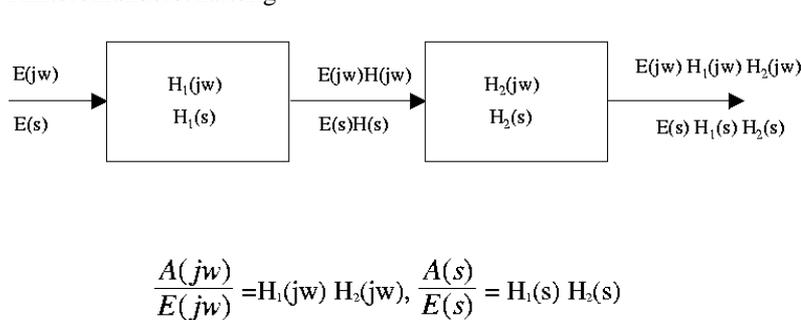
$$E(s) = \int_0^{\infty} e(t) e^{-st} dt$$

$$A(s) = \int_0^{\infty} a(t) e^{-st} dt$$

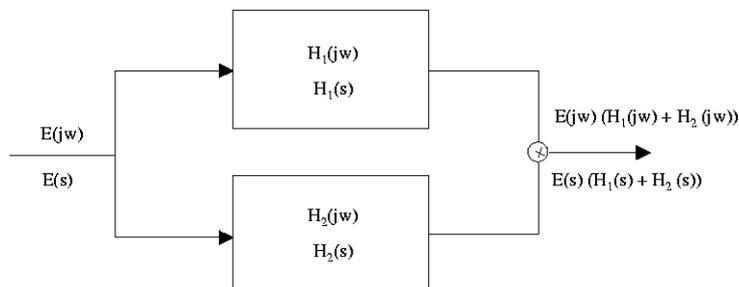
$$H(s) = \int_0^{\infty} h(t) e^{-st} dt$$

Die Laplace- und Fourier-Transformierten der Hintereinander-, Parallel- und Gegenschaltung von Systemkomponenten:

- Hintereinanderschaltung:

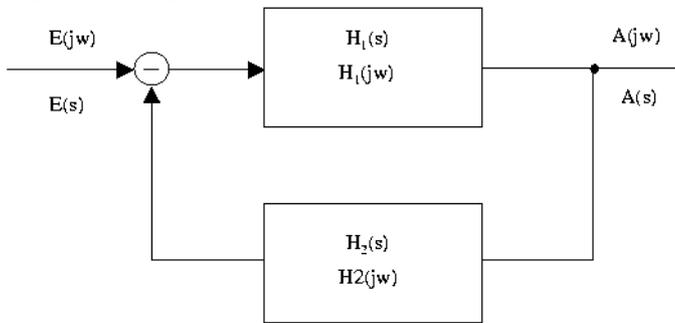


- Parallelschaltung:



$$\frac{A(j\omega)}{E(j\omega)} = H_1(s) + H_2(s), \quad \frac{A(s)}{E(s)} = H_1(s) + H_2(s)$$

- Gegenschaltung:



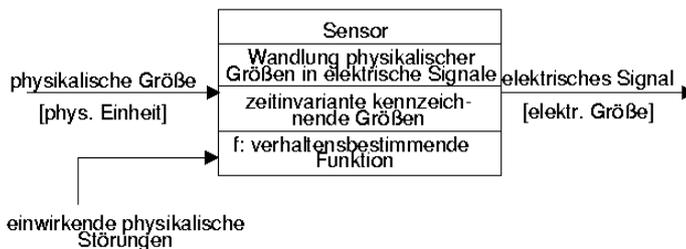
$$A(j\omega) = E(j\omega) H_1(j\omega) - A(j\omega) H_2(j\omega) H_1(j\omega)$$

$$\frac{A(j\omega)}{E(j\omega)} = \frac{H_1(j\omega)}{1 + H_1(j\omega)H_2(j\omega)}; \quad \frac{A(s)}{E(s)} = \frac{H_1(s)}{1 + H_1(s)H_2(s)}$$

7 Messen/Regeln/Stellen in der klassischen Automatisierungstechnik

7.1 Messen

- Umwandlung physikalischer Größen in elektrische Signale
- Illustration eines Sensors als signalverarbeitendes System:

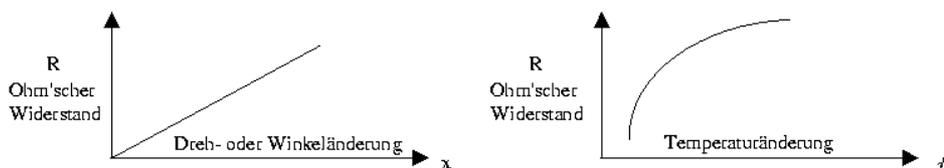


- Physikalische Größen:
 - Widerstand
 - Induktivität
 - Kapazität

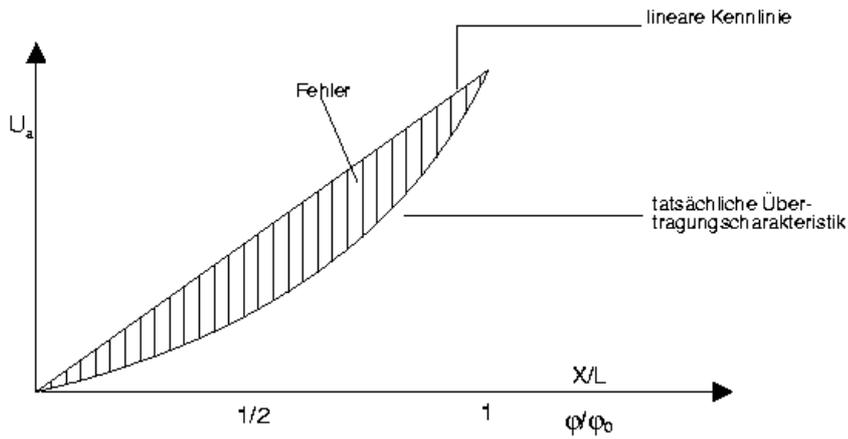
Beispiele für ohm'sche Meßgeber und für Meßfühleranordnungen

Ohm'scher Meßgeber	Potentiometer	Dehnmeßstreifen	Widerstandsthermometer	Fotowiderstand	Feldplatte
Meßgröße	Weg, Winkel	Dehnung, Kraft, Druck, Weg, Winkel, Thorsion	Temperatur, Strömung	Lichtgrößen, Weg, Winkel	Magnetische Größen, Weg, Winkel
sichveränderliche Größe	Ohm'scher Widerstand	Ohm'scher Widerstand	Ohm'scher Widerstand	Ohm'scher Widerstand	Ohm'scher Widerstand

- Übertragungseigenschaften von Meßfühlern und Feldplatten

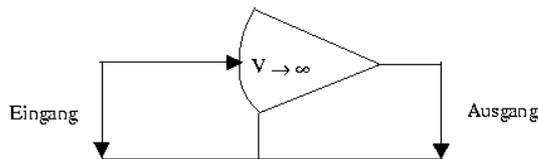


- Übertragungscharakteristik eines Potentiometers

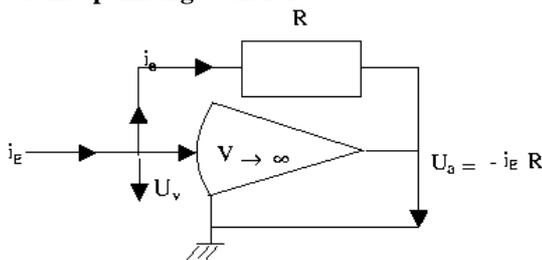


- **Meßwertaufbereitung:**

- Operationsverstärker: Hoher Widerstand und sehr hohe Verstärkung, die als ∞ betrachtet werden kann
- Symbol des Operationsverstärkers:



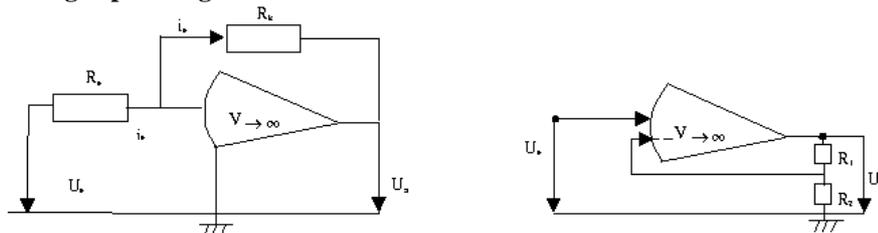
- **Strom/Spannungswandler**



$$U_a = i_E R$$

Stromspannungswandlung ist z.B. bei der Messung von Helligkeiten mit Hilfe von Photoelementen bedeutsam.

- **Spannungs/Spannungswandler**



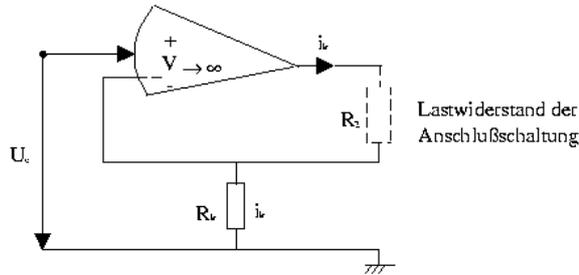
Bei der linken Schaltung:

$$i_e = \frac{U_e}{R_e}; \quad u_s = -i_e R_s = -u_e \frac{R_s}{R_e}$$

Bei der rechten Schaltung (annähernd):

$$V \left(U_e - \frac{U_a R_2}{R_1 + R_2} \right) = U_s \quad \text{mit } V \rightarrow \infty \quad U_s = U_e \left(1 + \frac{R_1}{R_2} \right)$$

- **Spannungs/Stromwandlung**



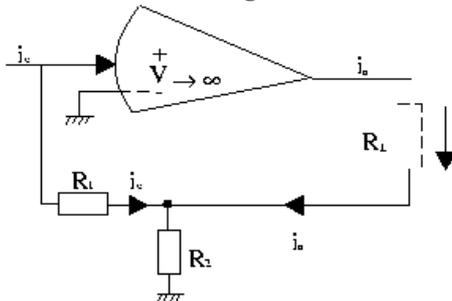
$$V (U_e - i_k \cdot R_s) = i_k (R_k + R_s)$$

mit $V \rightarrow \infty$

$$i_k = \frac{U_e}{R_s}$$

Eine Anwendung: Wenig fehlerhafte Gleichrichtung einer Wechselspannung

- **Strom/Stromwandlung**



$$V (i_e (R_1 + R_2) + i_a R_2)$$

$$= i_a R_L + (i_a + i_e) R_2$$

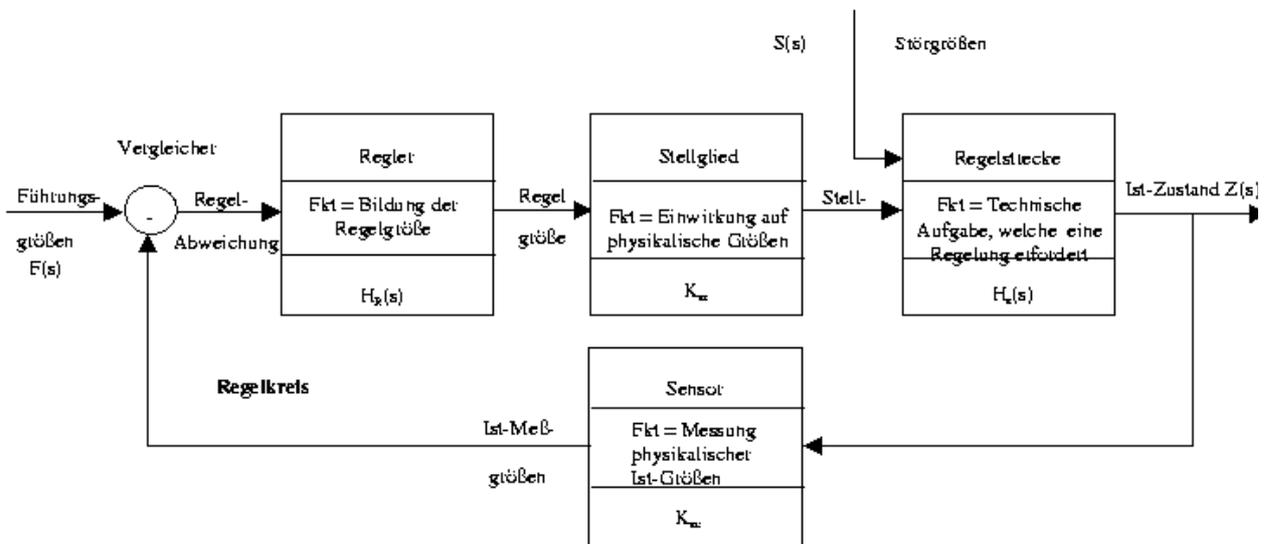
mit $V \rightarrow \infty$

$$i_a = i_e \left(1 + \frac{R_1}{R_2} \right)$$

7.2 Regelungstechnik

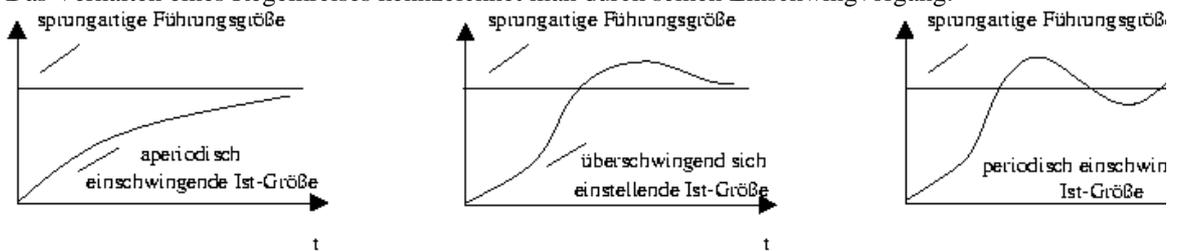
- Die Methodik zur selbständigen Regelung von physikalischen Systemen
- Regeln heißt, physikalische Ist-Zustände zu messen, sie mit vorgegebenen Führungsgrößen (Soll-Werte) zu vergleichen und bei Regelabweichungen durch einen Regler eine Regelgröße zu bilden. Damit muß über Stellglieder auf das physikalisch/technische System eingewirkt werden, daß sich die Ist-Zustandsgrößen den gewünschten Soll-Größen fortlaufend annähern.

Regelkreis:



- Bei Regelung einer physikalischen Größe (Fahrzeuggeschwindigkeit): **einschleifiger Regelkreis**
- Bei mehreren zu regelnden Größen (Flugzeug: v, h, Rollage, etc): **Mehrgrößenregelung** oder **gekoppelter Regelkreis**

- Wenn Ist-Größe konstant ist: **Haltepunktregelung**
- Das Verhalten eines Regelkreises kennzeichnet man durch seinen Einschwingvorgang:



- Wenn Regler, Regelstrecke, Stellglied und Sensor definiert sind und ihr Verhalten bekannt ist:

$$\frac{Z(s)}{F(s)} = \frac{H_R(s)K_S H_i(s)}{1 + H_R(s)K_S H_i(s)K_S}$$

- Störübertragungsgröße:

$$\frac{Z(s)}{S(s)} = \frac{H_S(s)}{1 + H_R(s)K_S H_i(s)K_S}$$

Einzelsschritte zur Lösung von Regelungsaufgaben:

- Zunächst 2 Schritte
 - Analyse des zeitlichen Verhaltens der zu regelnden Systems Übertragungsfunktion $H_S(s)$
 - Problemanalyse, welche die Spezifikation des zeitlichen Verhaltens des geregelten Systems beinhaltet.
- dritter Schritt: Festlegung der Übertragungsfunktion des Reglers (dabei ergibt sich:

$$H_R(s) = \frac{H_{RK}}{K_S H_i(s) (1 - H_{RK} K_S)}$$
- vierter Schritt: Abbildung der Übertragungsfunktion auf ein Netz hintereinandergeschalteter, parallelgeschalteter und gegengeschalteter elementarer Funktionsblöcke mit elementaren Übertragungsfunktionen. (Dieser Schritt entspricht dem Engineering für ein CBS Service-Modell von Abschnitt 4)
- fünfter Schritt: Entwurf der physikalischen Regler. Zur Realisierung werden in der Regel Operationsverstärker verwendet

7.3 Stellen

- Stellen bedeutet die Umwandlung von elektrischen Signalen in die zur Automatisierung erforderlichen physikalischen Größen
 - Hubmagnete, Antriebsmotoren, Ventile (Durchflüsse, Drücke in Behältern, Erzeugen von Impulsen)
 - Anstelle "Stellglied" sagt man auch "Aktor"
 - Wichtig für die Anwendung ist die Struktur und das Verhalten
 - Verhalten: Zusammenhang zwischen ansteuernder elektrischer Größe und resultierender physikalischer Größe (z.B. statisch)
 - Bei Stellgliedern muß häufig das zeitliche Verhalten bei dem Einsatz in einer Automatisierungsaufgabe mitbetrachtet werden
 - Wenn das Stellsystem selbst geregelt ist (meist wegen Überschaubarkeit), bezeichnet man es auch als **Servosystem**
 - Es kann erforderlich sein, elektrische Größen mehrfach in physikalische Größen zu wandeln (Pneumatik, Hydraulik) Solche Systeme sind u.a. in explosionsgeschützten Anlagen notwendig