

## § 2 Automatisierungsgerätesysteme und -strukturen

- 2.1 Automatisierungs-Computer
- 2.2 Automatisierungs-Strukturen
- 2.3 Automatisierungs-Hierarchien
- 2.4 Verteilte Automatisierungssysteme
- 2.5 Automatisierungsstrukturen mit Redundanz

### Kapitel 2 - Lernziele

- Die unterschiedlichen Automatisierungscomputer kennen
- Wissen, was das besondere an der Arbeitsweise einer SPS ist
- Zwischen zentralen und dezentralen Strukturen unterscheiden können
- Kombinationen von Automatisierungsstrukturen erkennen können
- Automatisierungshierarchien und deren Anforderungen kennen
- Verstehen, was dezentrale Automatisierungssysteme sind
- Wissen, was die Grundstrukturen der Kommunikation sind
- Zwischen einem offenem und einem geschlossenem Kommunikationssystem unterscheiden können
- Wissen, was man unter Redundanz versteht
- Arten von Hardware-Redundanz kennen und charakterisieren können
- Erklären können, was Diversität ist

## § 2 Automatisierungsgerätesysteme und -strukturen

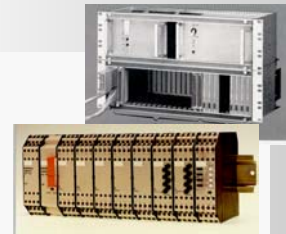
### 2.1 Automatisierungs-Computer

- 2.1.1 Speicherprogrammierbare Steuerungen (SPS)
- 2.1.2 Mikrocontroller
- 2.1.3 Industrie- PC
- 2.1.4 Prozessleitsysteme
- 2.2 Automatisierungs-Strukturen
- 2.3 Automatisierungs-Hierarchien
- 2.4 Verteilte Automatisierungssysteme
- 2.5 Automatisierungsstrukturen mit Redundanz

### 2.1.1 Speicherprogrammierbare Steuerungen (SPS)

#### Ausgangssituation (1)

- SPS seit 1972
- Von Bauteilen und Technologien geprägt
- Abhängig von der Aufgabe
- Einzug der SW



- ➔ Zunahme der Funktionalität und Leistungsfähigkeit von speicherprogrammierbaren Steuerungen

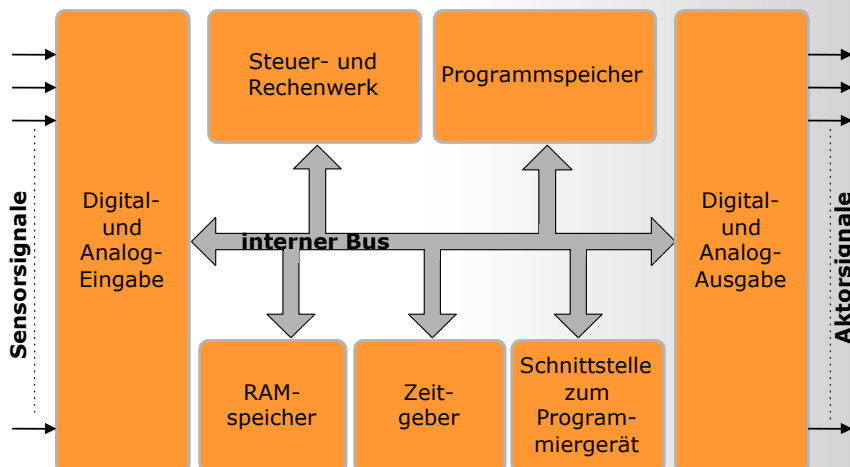
1980	:	2	kByte
1990	:	20	kByte
2000	:	2000	kByte



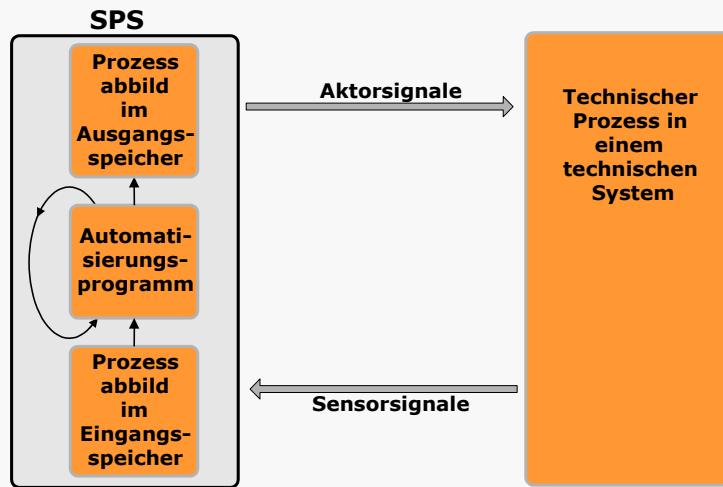
## Ausgangssituation (2)

- Zielgruppen
    - Anwender ohne Informatik-Studium
    - "Elektriker"
  - Zielsetzung für den SPS-Einsatz
    - Schütze/Relais ablösen
    - Zuverlässigkeit erhöhen
    - Kosten reduzieren
  - Zielsetzung für die SPS-Sprachen
    - Funktionen in bekannten Darstellungen beschreiben
      - Kontaktplan (abgeleitet aus dem Stromlaufplan)
      - Funktionsplan (abgeleitet aus dem Logikplan)
- ➔ Standardisierung der Entwicklung von SPS-Systemen
- IEC 1131  
DIN EN 61131

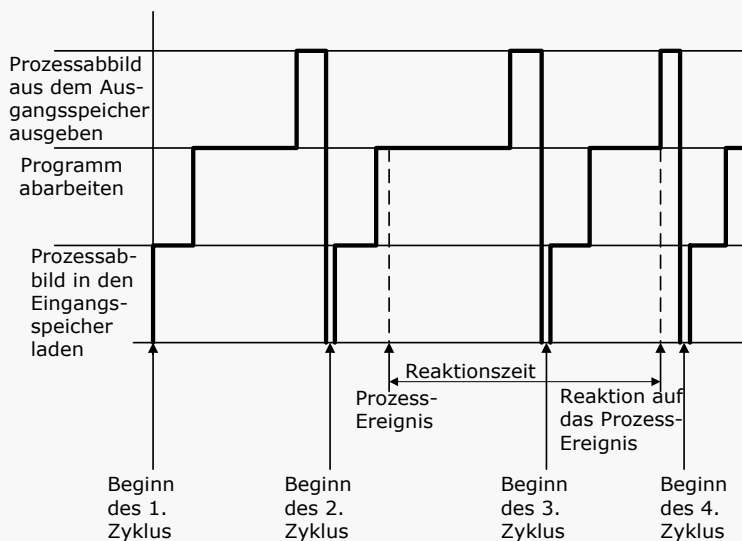
## Blockschaltbild der Hardwarestruktur einer SPS



## Zyklischer Betrieb beim Einsatz einer SPS



## Ablauf des zyklischen Programmbetriebes bei einer SPS



## Eigenschaften von SPS

**Vorteil:** einfache Programmierung durch zyklische Betriebsweise

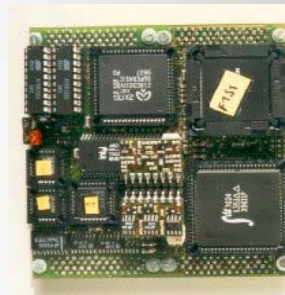
**Nachteil:** Reaktionszeit auf Ereignisse im technischen Prozess  
maximal zwei Programm-Zyklen

Programmarbeitungszeit:

- **Zykluszeit nicht konstant**
- **1 ms pro 1000 Anweisungen**

## Mikrocontroller (Ein-Chip-Computer)

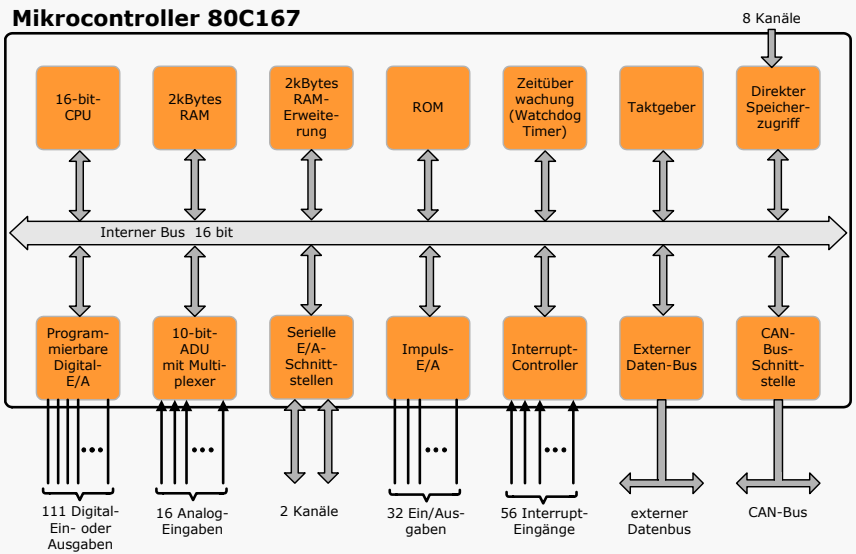
- Hochintegrierte Bausteine
- Einsatz für Massenprodukte
- Aufbau aus
  - Standard-Mikroprozessor
  - Datenspeicher/ Programmspeicher
  - Bus-Schnittstellen
  - Prozess-Signal-Schnittstellen
- Programmierung über Entwicklungssysteme
- kurze Wortlänge
- extrem niedriger Preis
- hohe Zuverlässigkeit und Lebensdauer
- hohe Anforderungen bezüglich Umgebungsbedingungen



**Beginnend bei 1 - 10 €**

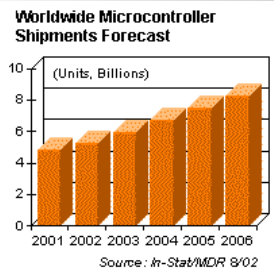
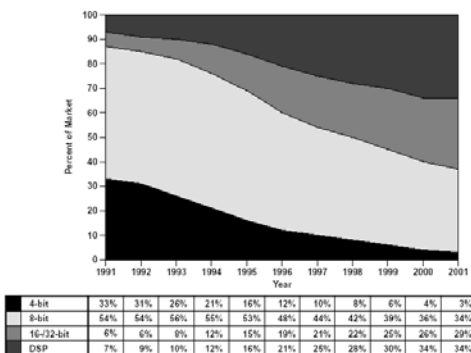
**Temperatur, Feuchtigkeit**

### Vereinfachtes Blockbild eines Mikrocontrollers



### Weltweiter Mikrocontroller-Markt

- Hersteller: Intel, Motorola, National Semiconductor, Toshiba, Infineon, Mitsubishi
- Weltweit wachsender Markt
- Zunehmende Tendenz zu 16-/32bit Mikrocontrollern und spezifischen Digitalen Signalprozessoren



## Unterscheidung

- Mikroprozessor  
Prozessor auf einem Mikroelektronik-Chip
  
- Mikrocomputer  
Alle Komponenten auf einem Mikroelektronik-Chip, d.h. Prozessor,  
Speicher, Schnittstellen zur Peripherie
  
- Mikrocontroller  
Automatisierungs-Computer bzw. ein Automatisierungs-  
Computersystem auf einem Chip

## Glossar

CPU = Central Processing Unit	<b>Mikroprozessor</b>
RAM = Random Access Memory	<b>Arbeitsspeicher</b>
EPROM/PROM/ROM = Erasable Programmable Read Only Memory	<b>Festwertspeicher</b>
I/O = parallele bzw. serielle Ein/Ausgabe- Bausteine	<b>Prozess- und Datenperipherie</b>
Counter/Timer = Taktgeber	
Interrupt Controller = Unterbrechungswerk	

### Industrie-PC (IPC)

- einsteckbare Leiterplatten zum Anschluss von
  - elektrischen Prozess-Signalen
  - optischen Prozess-Signalen
  - Bussystemen
- Programmierung in Hochsprache
- Einsatz von Echtzeit-Betriebssystemen
  - als einziges Betriebssystem
  - zusätzlich zu Standard-Betriebssystemen



### Einsatzgebiete von Industrie-PCs

- Prozess-Visualisierung
- Prozessauswertung und -überwachung
- übergeordnete Steuerungsaufgaben (Leitstandsaufgaben)

### (Eigenschaften) Umgebungsbedingungen

- Rauhe Umgebungsbedingungen
  - Temperaturschwankungen
  - Stöße und Erschütterungen
  - Staub und Feuchtigkeit
  - elektrische oder elektromagnetische Störungen
- Schutzvorrichtungen von Industrie-PCs (IPC)
  - schwingungsgedämpfte Laufwerke
  - hohe Güte der integrierten Bausteine
  - spezielles Schutzgehäuse



## Schutzart von Industrie-PCs mit IP-Index

1. Ziffer	Schutz gegen Festkörper	2. Ziffer	Schutz gegen Wassereinwirkung
0	kein Schutz	0	kein Schutz
1	Handkontakt unmöglich (50mm Objekte)	1	Schutz gegen vertikal fallende Tropfen/ Kondensation
2	Fingerkontakt unmöglich (12mm Objekte)	2	Schutz gegen Tropfen mit einem Fallwinkel von 15°
3	Drahtkontakt unmöglich (2.5mm Objekte)	3	Schutz gegen Regenfall bis zu 60°
4	Feiner Drahtkontakt unmöglich (1.0mm Objekte)	4	Schutz gegen Sprühwasser von allen Seiten
5	Schutz gegen schädlichen Staub	5	Schutz gegen Wasserstrahlen von allen Seiten
6	Komplett staubgeschützt	6	Schutz gegen Wasserfluten (bei schwerem Seegang)
		7	Schutz gegen Wassereinwirkung bei 1m Tiefe
		8	Schutz gegen lange Wassereinwirkung bei > 1m Tiefe

**IP =  
Ingress Protection**

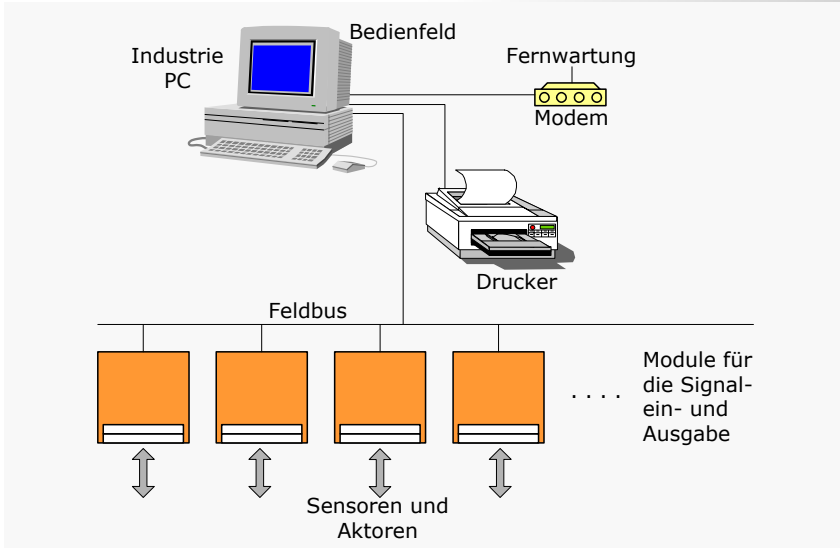
## Reine IPC-Lösung

- Erfassung von Prozessgrößen
- Steuerungsfunktion unter Echtzeit
- Schnittstelle zum Bediener

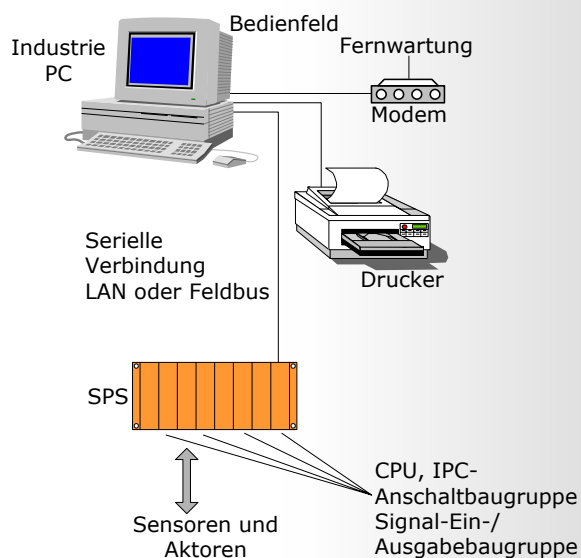
## Vorteile einer reinen IPC-Lösung

- bessere Skalierbarkeit der Hardware
- großes Angebot an Betriebssystemen
- großes Spektrum an Programmiersprachen
- offenes System zur Integration von fertigen Teillösungen

### Anordnung eines reinen IPC-Systems



### Anordnung eines SPS-IPC-Systems (Tafelanschrieb)



### Prozessleitsysteme (PLS)

- Verteilte, über Bus-Systeme verbundene Rechnersysteme
- Kopplung mit SPS-Rechnern
- Einsatz von vorkonfektionierten, vom Hersteller des PLS entwickelten Programmbausteinen
- Konfigurierung durch Anwender

### Komplettlösungen von einem Hersteller

- keine Kompatibilitätsprobleme
- einheitliche Bedienung und Beobachtung des Prozesses
- hohe Verfügbarkeit
- definierte Verantwortlichkeit
- lange Lebenszeit

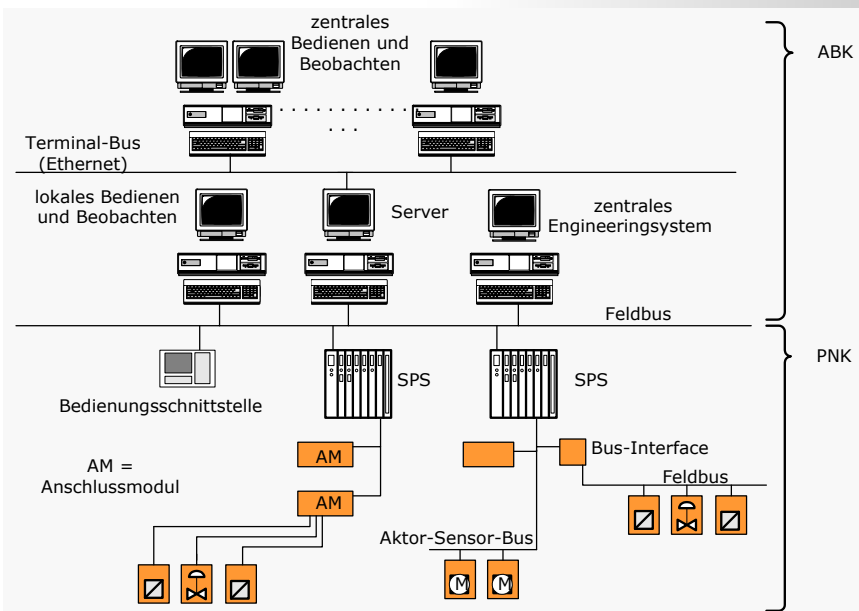
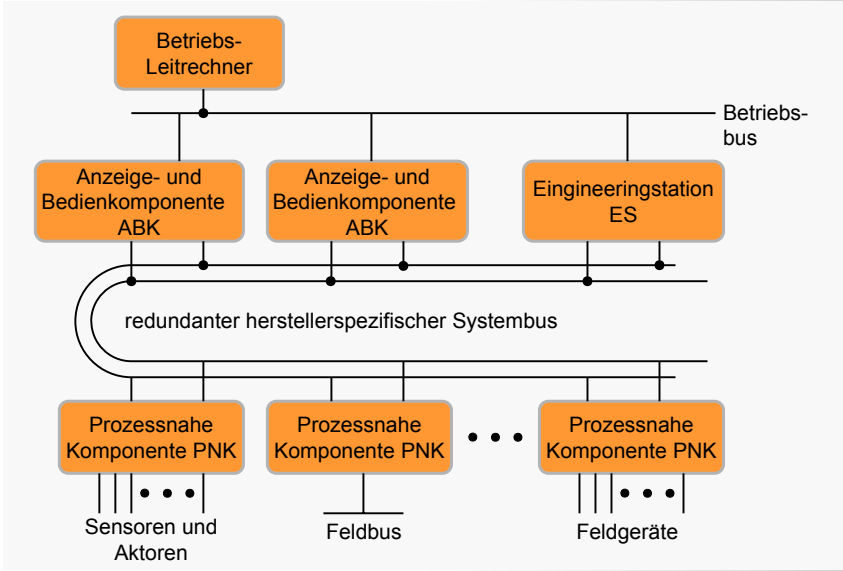
### Anwendungsgebiete von Prozessleitsystemen

- Kraftwerkstechnik
- Verfahrenstechnik
- Gebäudetechnik
- Fertigungstechnik

### Bestandteile eines PLS

- Anzeige- und Bedienkomponente (ABK)
- Prozessnahe Komponenten (PNK)
- Systemkommunikation
- Engineering-Tool

**Schematischer Aufbau**



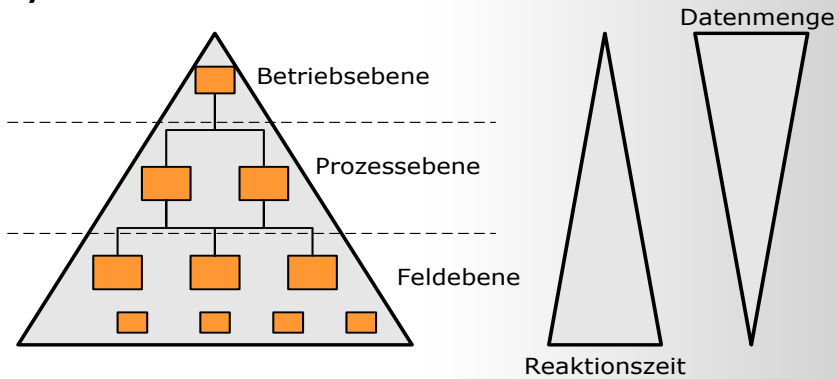
## Anzeige- und Bedienkomponente

- Funktionen
  - Rezepte/ Batch Abläufe erstellen/ modifizieren
  - aktuelle Werte ändern
  - Kommunikation mit dem Prozess
  - Alarmer/ Bedienanforderungen bearbeiten
  - Prozessvisualisierung
  - Schnittstelle zu Datenbanksystemen zur Prozessprotokollierung
  
- Realisierung
  - PC, IPC, Workstation
  - Windows 95, Windows NT, Unix

## Prozessnahe Komponente

- Eigenentwicklungen der Leitsystem-Hersteller
  
- Aufbau
  - Automatisierungscomputer: SPS, IPC
  - dezentrale Peripherie
  - Feldgeräte: Sensoren, Aktoren

## Systemkommunikation



### Bus-Systeme

Betriebsebene:	standardisierter Ethernet-Bus
Prozessebene:	Feldbus, z.B. Profibus, H1-Bus, Modbus, Interbus-S
unterste Feldebene:	schnelle Aktor-Sensor-Busse

## Engineering-Tool

- Aufgaben
  - Konfiguration
  - Programmierung
  - Wartung/Pflege
- Werkzeuge
  - graphische Werkzeuge (IEC 1131)
  - standardisierte Bibliotheken mit Komponenten
  - mächtige Editoren

### Leitsystemhersteller und deren Produkte

Hersteller	Produktname	Bemerkung
ABB	AdvantOCS	Firmenspezifischer Feldbus
	AdvaSoft	für kleine Anlagen
	Procontrol P	Kraftwerktechnik
EB Hartmann & Braun	Symphony	Unterstützt Anbindung an Betriebsebene, Remote I/O System
	Contronic E	Kraftwerktechnik, für große Anlagen
	Contronic P	Verfahrenstechnik, Ausdehnung bis zu 12km
Foxboro-Echardt	I/A Serie-System	Verfahrenstechnik, PNK mit PCMCIA-Technologie
Siemens	SIMATIC PCS 7	Verfahrenstechnik, ABK auf Basis von Windows 95 und Windows NT. Umfangreiches Hardwareangebot. Feldbus: Profibus. Anbindung an Betriebsebene möglich.
	Teleperm M	Verfahrenstechnik, weitverbreitet. Altes Bussystem (CS 275). Migration von Teleperm M nach SIMATIC PCS 7 möglich.
	Teleperm XP	Kraftwerkstechnik, offene Kommunikation, umfangreiches Hardwareangebot.
Honeywell	PlantScape	Verfahrenstechnik, offenes System, ABK auf Basis von Windows NT, unterstützt Remote I/O. Anbindung an Betriebsebene möglich.
	TDC 3000	Verfahrenstechnik, besitzt mehrere Prozessbusse mit unterschiedlicher Datenübertragung. MODBUS wird unterstützt.

## § 2 Automatisierungsgerätesysteme und -strukturen

- 2.1 Automatisierungs-Computer
- 2.2 Automatisierungs-Strukturen**
  - 2.2.1 Arten von Automatisierungsstrukturen
  - 2.2.2 Kombination von Automatisierungsstrukturen
- 2.3 Automatisierungs-Hierarchien
- 2.4 Verteilte Automatisierungssysteme
- 2.5 Automatisierungsstrukturen mit Redundanz

### Struktur des technischen Prozesses

- technischer Prozess als Einheit  
Bsp.: Bohrvorgang bei einer Bohrmaschine
- technischer Prozess, der aus Teilprozessen besteht  
Bsp.: Fertigung eines Getriebes

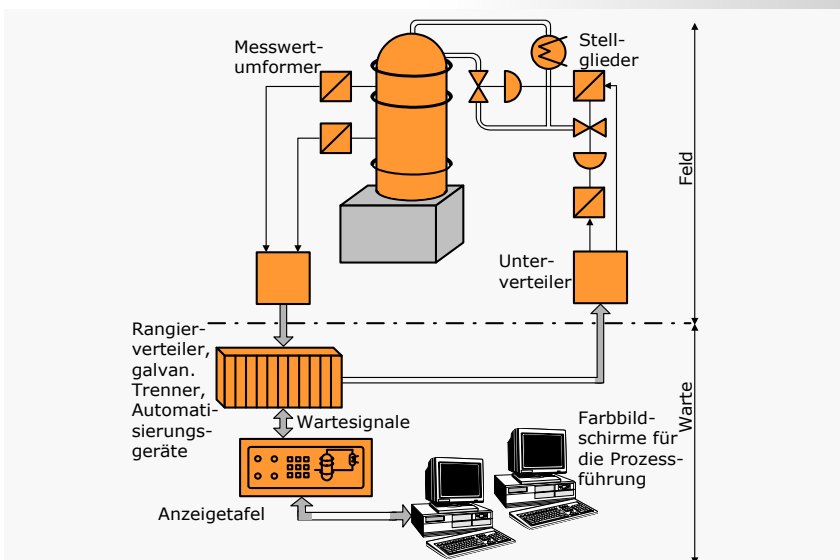
### Örtliche Struktur der Automatisierungsgeräte

- örtlich zentrale Anordnung
- örtlich dezentrale Anordnung

### Wirkungsmäßige Struktur des Automatisierungssystems (funktionelle Struktur)

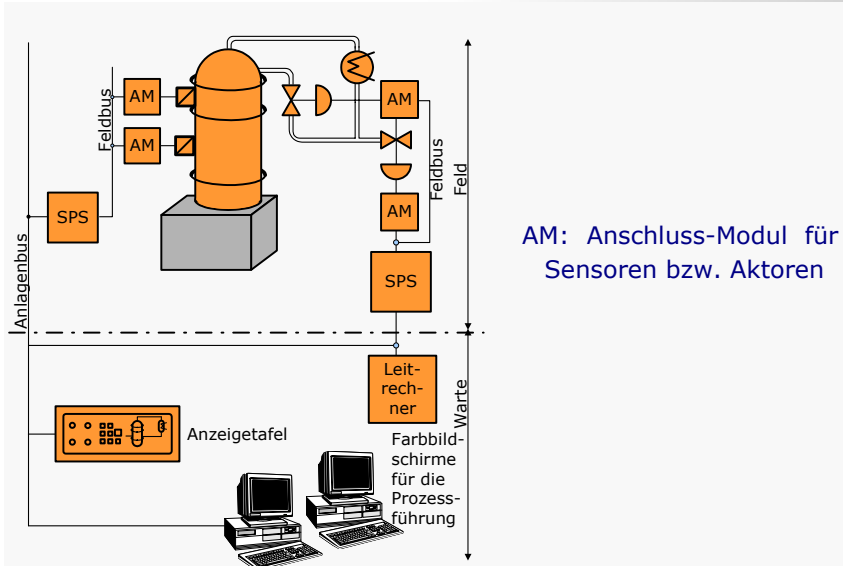
- = Aufteilung der Automatisierungsfunktionen auf die Automatisierungsgeräte
- wirkungsmäßig zentrale Grundstruktur
- wirkungsmäßig dezentrale Grundstruktur

### Örtlich zentrale Anordnung der Automatisierungsgeräte

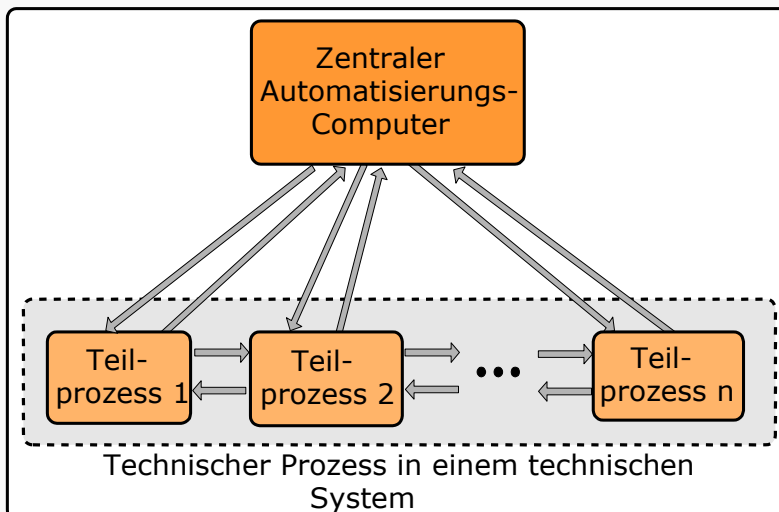




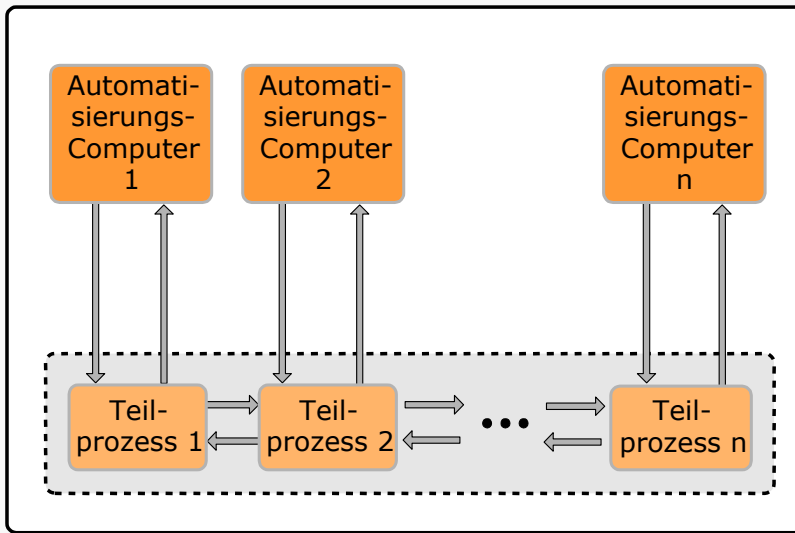
### Örtlich dezentrale Anordnung der Automatisierungsgeräte



### Wirkungsmäßig zentrale Automatisierungsstruktur



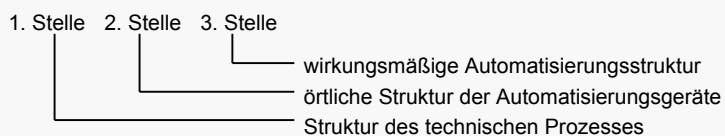
**Wirkungsmäßig dezentrale Automatisierungsstruktur**



**Unterschiedliche Kombination der Automatisierungsstrukturen**

		Technischer Prozess als eine Einheit betrachtet (zentrale Prozess-Struktur)		Technischer Prozess als Teilprozesse gegliedert (dezentrale Prozess-Struktur)	
		örtlich zentrale Anordnung der Automatisierungsgeräte	örtlich dezentrale Anordnung der Automatisierungsgeräte	örtlich zentrale Anordnung der Automatisierungsgeräte	örtlich dezentrale Anordnung der Automatisierungsgeräte
wirkungsmäßig Struktur	wirkungsmäßig zentrale Automatisierungsstruktur	Z Z Z	Z D Z	D Z Z	D D Z
	wirkungsmäßig dezentrale Automatisierungsstruktur	Z Z D	Z D D	D Z D	D D D

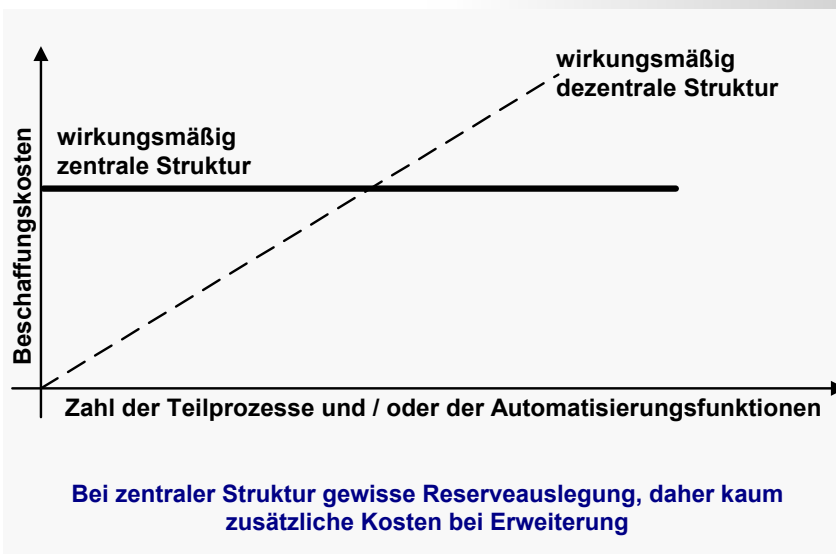
Kennzeichnung der verschiedenen Strukturen: Z = zentral/ D = dezentral



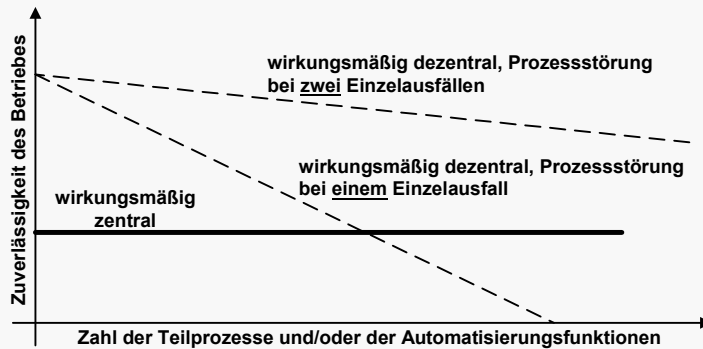
## Vergleichskriterien für Automatisierungsstrukturen

- die **Kosten** für die Geräte, die Verkabelung, die Software, die Pflege und die Wartung
- die **Teilverfügbarkeit** bei Hardware-Ausfällen oder bei Software-Fehlern
- die **Flexibilität** bei Änderungen
- die **Koordinierung** der Teilprozesse und die **Optimierung** des Gesamtprozesses
- die **Bedienbarkeit**

## Vergleich der Anschaffungskosten



## Vergleich der Zuverlässigkeit des Betriebes



Im Normalfall Verkopplung schwach:

- kein Totalausfall bei Ausfall von zwei oder mehr Automatisierungseinheiten
- Zuverlässigkeit des Betriebes bei einer wirkungsmäßig dezentralen Struktur höher als beim Einsatz eines zentralen Prozessrechners

## Dezentrale Struktur

- + Flexibilität bei Änderungen
- + Koordination der Teilprozesse
- + Optimierung des Gesamtprozesses
- zusätzlicher Aufwand zur Kommunikation der einzelnen Automatisierungseinheiten
- 0 Bedienbarkeit und Benutzerfreundlichkeit
- + Störfall-Lokalisierung
- + höhere Transparenz

## Bewertung von Automatisierungsstrukturen

Z = zentrale Struktur  
D = dezentrale Struktur

ZZZ	Typisch für Automatisierung kleinerer Geräte
ZDZ	Geringere Verkabelungskosten als bei ZZZ
DZZ	Ungünstig bzgl. Verfügbarkeit, Wartbarkeit, Verkabelungskosten
DDZ	Ungünstig bzgl. Verfügbarkeit und Flexibilität
ZZD	Günstig bzgl. Wartbarkeit und Flexibilität Ungünstig: Verkabelung
ZDD	Günstig: Flexibilität, Verfügbarkeit, Verkabelung Transparenz
DZD	Günstig: Verfügbarkeit, Wartbarkeit Ungünstig: Verkabelungskosten
DDD	Günstig: Flexibilität, Verfügbarkeit, Verkabelung, Transparenz

Produkt-  
automatisierung

Anlagen-  
automatisierung

Kfz-  
Elektronik

— wirkungsmäßige Struktur  
— örtliche Struktur der Automatisierungsgeräte  
— Struktur des technischen Prozesses

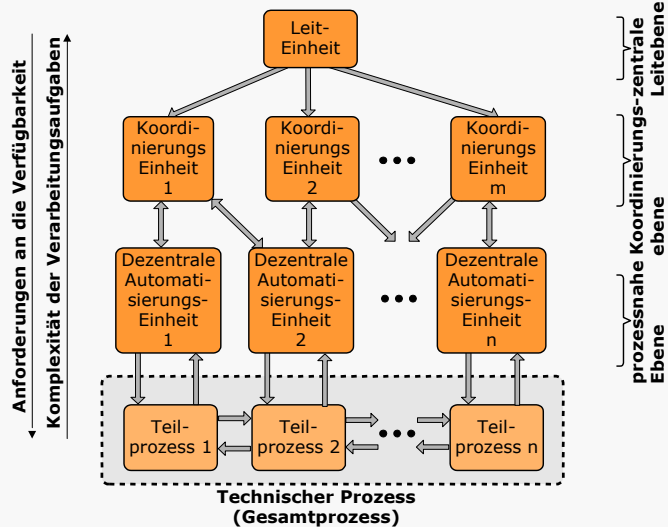
⇒ so dezentral wie möglich, so zentral wie nötig

## § 2 Automatisierungsgerätesysteme und -strukturen

- 2.1 Automatisierungs-Computer
- 2.2 Automatisierungs-Strukturen
- 2.3 Automatisierungs-Hierarchien**
- 2.4 Verteilte Automatisierungssysteme
- 2.5 Automatisierungsstrukturen mit Redundanz

## Kombination von zentraler und dezentraler Struktur

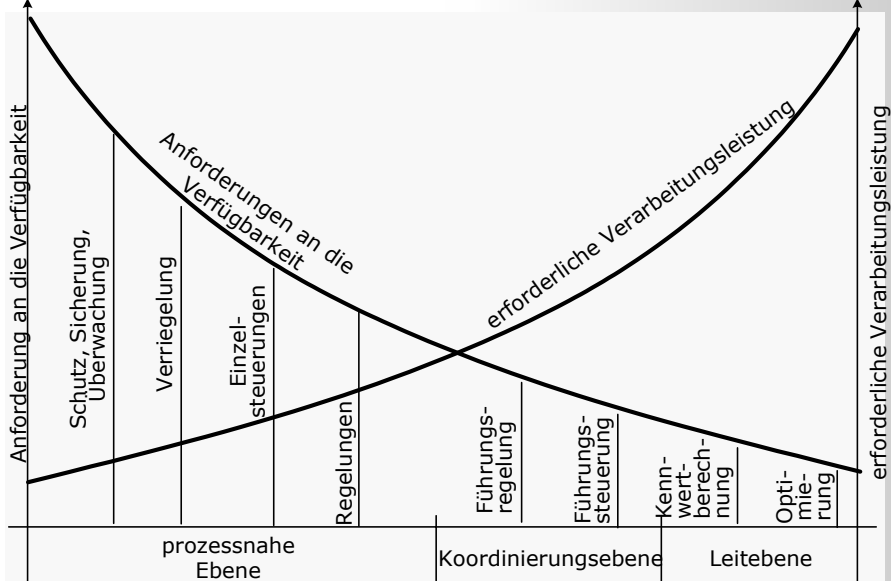
→ Einführung einer Hierarchie von Automatisierungseinheiten



## Zuordnung von Automatisierungsfunktionen zu den Prozessführungsebenen

- Dezentrale Automatisierungseinheiten erfüllen die Aufgaben der prozessnahen Ebene, hohe Anforderungen an die Verfügbarkeit
- Koordinierungs-Einheiten realisieren die Automatisierungsfunktionen der Koordinierungsebene, wie Koordinierung der Teilprozesse, Optimierung, Prozessüberwachung und -sicherung
- Leiteinheit realisiert die Aufgaben der Leitebene

### Anforderungen an Verfügbarkeit und Verarbeitungsleistung



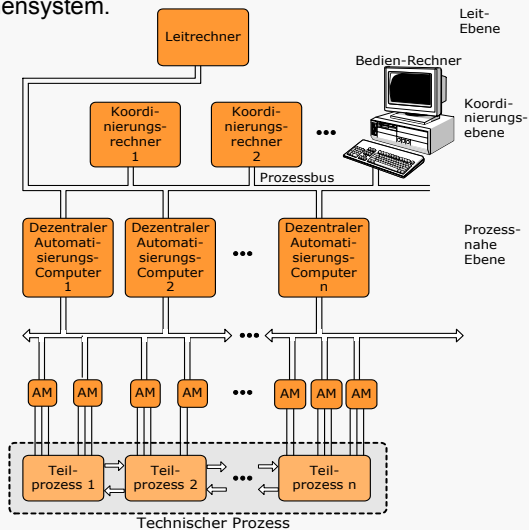
### Voraussetzung für die Realisierung

- Gliederung des technischen Prozesses in Teilprozesse
- Einsatz von intelligenten Automatisierungseinheiten
- Kommunikationssystem zwischen Automatisierungseinheiten

Je nach Größe des Unternehmens und Umfang des technischen Prozesses können Zwischenebenen eingeführt werden bzw. Ebenen zusammengezogen werden.

**Beispiel**

Realisierung einer Automatisierungs-Hierarchie mit einem busorientierten verteilten Prozessrechensystem.



## § 2 Automatisierungsgerätesysteme und -strukturen

- 2.1 Automatisierungs-Computer
- 2.2 Automatisierungs-Strukturen
- 2.3 Automatisierungs-Hierarchien
- 2.4 Verteilte Automatisierungssysteme**
  - 2.4.1 Zielsetzung
  - 2.4.2 Grundstrukturen der Kommunikation
  - 2.4.3 Bus-Systeme
  - 2.4.4 Arten von Kommunikationssystemen
- 2.5 Automatisierungsstrukturen mit Redundanz



### Zielsetzung bei verteilten Automatisierungssystemen

- Hohe Zuverlässigkeit durch Fehlertoleranz
  - Ausfall eines dezentralen Computers führt nicht zu Gesamtausfall
  - Eingrenzung des Fehlers durch Rekonfiguration
- Erhöhung der Verfügbarkeit durch schnelle Wartung und Instandsetzung
  - gegenseitige Überwachung mit Fehlerdiagnose
- Gegenseitige Aushilfe bei hoher Belastung
  - selbsttätige Anpassung der Aufgabenverteilung
  - Reduzierung der Reserven der einzelnen Einheiten
- Einfache Erweiterungsfähigkeit

### Realisierung bei verteilten Automatisierungssystemen

Verknüpfung von dezentralen Automatisierungseinheiten mit übergeordnetem Rechner über ein Kommunikationssystem

**keine Hierarchie**

### Unterschied zu Automatisierungs-Hierarchie

- Einheiten kommunizieren dort nur mit nächst höherer Ebene
- Nur prozessnahe Informationsaufgaben werden dort dezentral wahrgenommen

### Kriterien für die Wahl eines Kommunikationssystems (1)

- niedrige Verkabelungskosten
  - **Wesentliche Kosten: mehr Kabel bedeutet mehr Störungen auf den Leitungen**
- standardisierte Schnittstellen bezüglich
  - Stecker, Leitungen (mechanisch)
  - Spannungspegel (elektrisch)
  - Übertragungsprotokoll (logisch)
- Flexibilität bei Änderungen
- geringe Anforderungen an die Kommunikationspartner
  - Speicherplatzbedarf
  - Rechenzeitaufwand
- hohe Verfügbarkeit und Zuverlässigkeit

### Kriterien für die Wahl eines Kommunikationssystems (2)

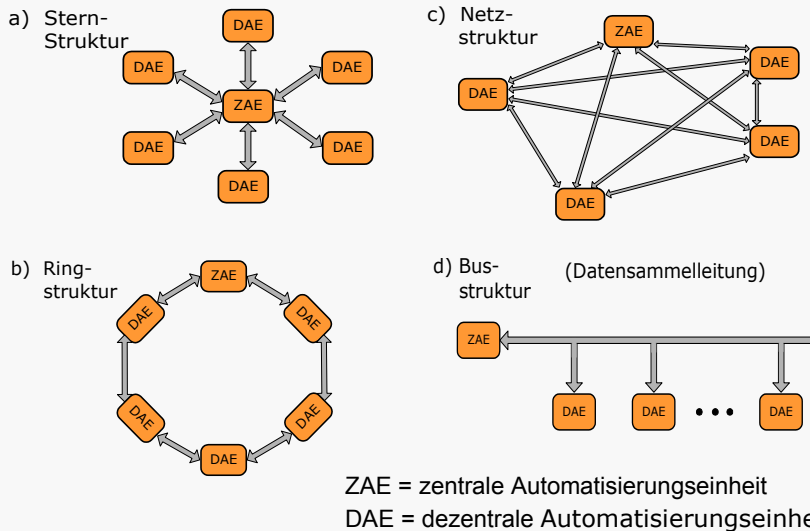
- sichere Übertragung der Informationen
  - Verwendung von Prüfbits
  - Bestätigung des korrekten Empfangs
- Realisierung hoher Datenübertragungsraten
- kurze Reaktionszeiten auf Übertragungsanforderungen
- Kopplung unterschiedlicher Kommunikationspartner

#### Problem:

Festlegung der Prioritäten bei der Realisierung der sich zum Teil widersprechenden Einzelkriterien?

**Bsp.:** hohe Verfügbarkeit und Zuverlässigkeit durch redundantes Bus-System bedeutet hohe Verkabelungskosten

## Grundstrukturen der Kommunikation



### Eigenschaften

#### Stern-Struktur

Ausfall der Zentraleinheit bedeutet Ausfall der Kommunikation

#### Ring-Struktur

Jede Einheit kann nur an direkte Nachbarn übertragen

#### Netz-Struktur

Parallele Informationsübertragung, kurze Reaktionszeit, viele Schnittstellen, hohe Verkabelungskosten

#### Bus-Struktur

Nur jeweils ein Teilnehmer kann senden, gleichzeitige Informationsaufnahme von allen Teilnehmern

### Parallele Busse

- Adressen, Daten und Steuersignale werden parallel übertragen
- Leitungsbündel

**Distanz: bis 20 m**

### Serielle Busse

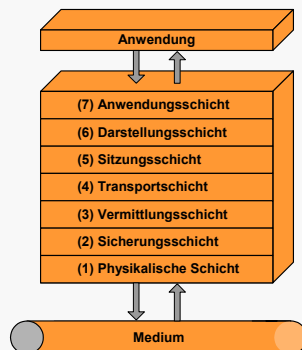
- Bits einer Nachricht zeitlich nacheinander
- Übertragungszeit länger als bei Parallelbus
- niedrigere Leitungskosten
- Erhöhung der Zuverlässigkeit
- Flexibilität bezüglich Leitungsprotokollen

**Distanz: 20 m bis 15 km**

### Arten von Kommunikationssystemen

- Offenes Kommunikationssystem

OSI = Open-System-Interconnection



**ISO/OSI-Basisreferenzmodell  
ISO IS 7498**

- Herstellerspezifisches Kommunikationssystem  
CSI = Closed-System-Interconnection

## Aufgaben der ISO/OSI- Schichten

Schicht 7:	<b>Anwendungsschicht</b> (Verarbeitungsschicht, Application Layer)	<ul style="list-style-type: none"> <li>– Grundvorrat an Diensten</li> <li>– Standardanwendungen</li> <li>– anwendungsspezifisch</li> </ul>
Schicht 6:	<b>Darstellungsschicht</b> (Presentation Layer)	<ul style="list-style-type: none"> <li>– Sprachanpassung (z.B. zwischen ASCII &amp; EBCDIC)</li> <li>– Datenverschlüsselung</li> <li>– Datenentschlüsselung</li> </ul>
Schicht 5:	<b>Sitzungsebene</b> (Kommunikationsstauerebene, Session Layer)	<ul style="list-style-type: none"> <li>– Eröffnung</li> <li>– Überwachung</li> <li>– Beendigung der Kommunikation</li> </ul>
Schicht 4:	<b>Transportebene</b> (Transport Layer)	<ul style="list-style-type: none"> <li>– Wegparallelisierung</li> <li>– Paketwiederholung</li> <li>– Paketsortierung</li> </ul>
Schicht 3:	<b>Netzebene</b> (Vermittlungsschicht, Network Layer)	<ul style="list-style-type: none"> <li>– Netzprotokolle</li> <li>– Datenadressierung</li> <li>– Datenvermittlung</li> <li>– Wegwahl</li> </ul>
Schicht 2:	<b>Verbindungsebene</b> (Sicherungsschicht, Data Link Layer)	<ul style="list-style-type: none"> <li>– Fehlerbehandlung/-erkennung</li> <li>– Zugriffsverfahren</li> <li>– Synchronisation</li> </ul>
Schicht 1:	<b>Physikalische Ebene</b> (Bitübertragungsschicht Physical Layer)	<ul style="list-style-type: none"> <li>– Übertragungsmedium</li> <li>– Codierungsart</li> <li>– Schnittstelle</li> <li>– Topologie</li> </ul>

## § 2 Automatisierungsgerätesysteme und -strukturen

- 2.1 Automatisierungs-Computer
- 2.2 Automatisierungs-Strukturen
- 2.3 Automatisierungs-Hierarchien
- 2.4 Verteilte Automatisierungssysteme

### 2.5 Automatisierungsstrukturen mit Redundanz

- 2.5.1 Formen der Redundanz
- 2.5.2 Fehlertolerante Strukturen
- 2.5.3 Hardware-Redundanz
- 2.5.4 Software-Redundanz

## Formen der Redundanz

Immer: Prozessbedienpersonal überwacht parallel zu einem Prozessrechner!

- Hardware-Redundanz
  - redundante Hardware
- Software-Redundanz
  - redundante Software
- Messwert-Redundanz
  - redundante Messgrößen
  - abhängige Messgrößen
- Zeit-Redundanz
  - mehrfache Abfrage des gleichen Messwertes in bestimmten Zeitabständen

**Bsp.: Geschwindigkeit,  
Beschleunigung**

Hardware- und Softwareredundanz bedeutet höherer Aufwand

- erhöhte Verfügbarkeit
- sicherheitsrelevante Systeme

## Ziel beim Einsatz fehlertoleranter Strukturen

Systeme so zu konstruieren, dass sie nach dem Auftreten von Fehlern in einzelnen Komponenten als Ganzes funktionsfähig sind.

## Stufen der Fehlertoleranz

- volle Fehlertoleranz **fail operational**
- verringerte Leistungsfähigkeit **fail soft, graceful degradation**
- Übergang in einen sicheren Zustand **fail safe**

### Prinzip der Fehlertoleranz

Aufbau eines Systems aus redundanten Modulen (Hardware- und Software), um bei Auftreten eines Fehlers die Funktionsfähigkeit des Systems zu erhalten.

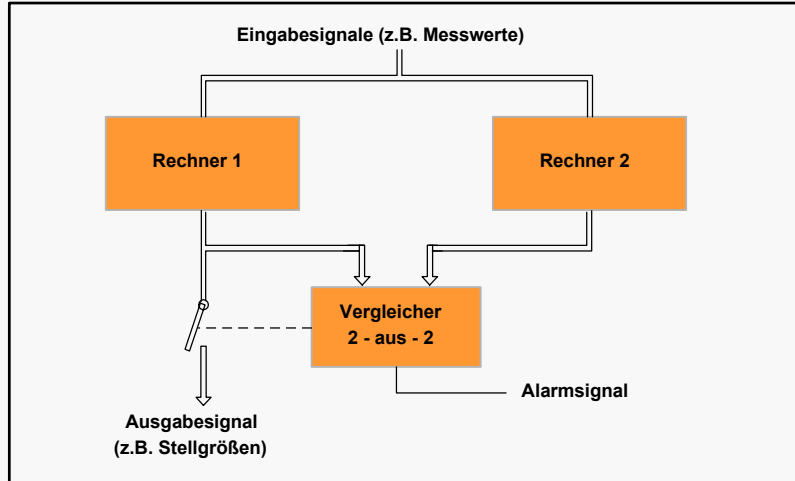
### Arten von Redundanz

- Statische Redundanz
  - alle redundanten Module ständig im Einsatz
- Dynamische Redundanz
  - redundante Module werden erst nach einem Ausfall eingesetzt
  - blinde Redundanz  
redundante Module sind im fehlerfreien Fall nicht tätig
  - funktionsbeteiligte Redundanz  
redundante Module führen im fehlerfreien Fall Stand-by-Funktionen durch

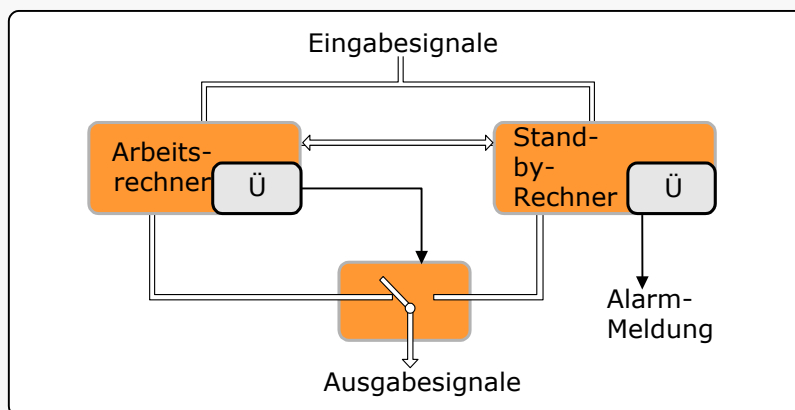
### Hardware-Redundanz

- Ziel:  
Erkennung von Ausfällen der Hardware
- Einsatzprinzip:  
m-von-n-Redundanz
  - Mehrheitsentscheid
  - Fehler erst bei Mehrfachdefekten
- Realisierung der Redundanz
  - Doppel-Rechner-Strukturen
  - Drei-Rechner-Strukturen

### Doppelrechner-Struktur mit statischer Redundanz



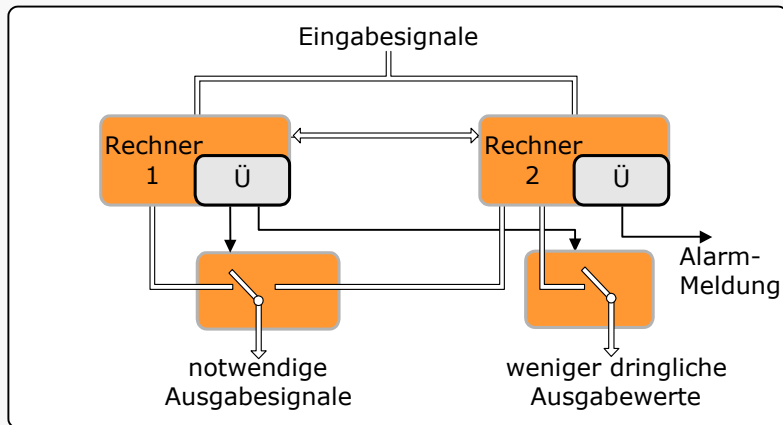
### Doppelrechner-Struktur mit dynamisch blinder Redundanz



Ü = Überwachungsprogramm

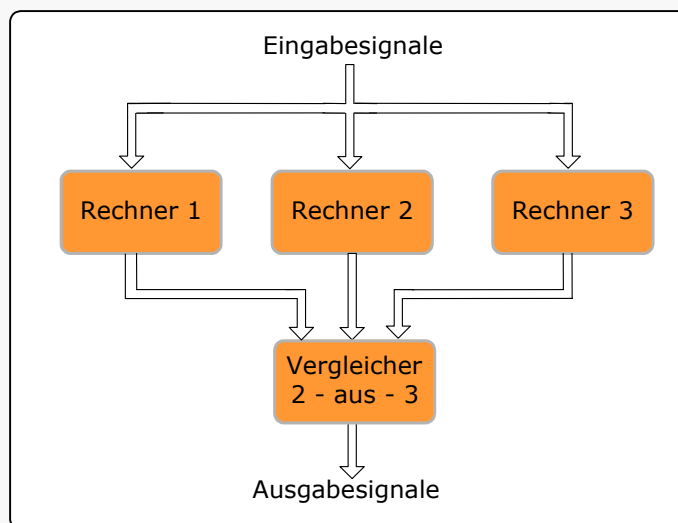


### Doppelrechner-Struktur mit dynamisch funktionsbeteiligter Redundanz



Ü = Überwachungsprogramm

### Drei-Rechner-Struktur mit statischer Redundanz



## Software-Redundanz

- Ziel:  
Erkennung von Fehlern in der Software
- Ausgangspunkt:  
**Software ist fehlerhaft**

Redundanzmaßnahmen bei Software:

Mehrfaches Vorhandensein der gleichen Software ist nicht sinnvoll, da nicht Ausfälle das Problem bei Software sind.

- verschiedenartiger Aufbau von Programmteilen
- gleiche Eingangsdaten müssen gleiche Ergebnisse liefern

## Diversitäre Software

**Diversität = Verschiedenartigkeit von Software bei gleicher Funktion**

- Unabhängige Entwicklerteams lösen dieselbe Aufgabe
- Gezielte Entwicklung verschiedener Strategien, Algorithmen und Software-Strukturen

### Einsatz bzw. Ausführung diversitärer Software-Teile

- Redundante Software-Alternativen werden nacheinander ausgeführt und über Entscheider (Voter) verglichen, nicht für Echtzeitsysteme mit hohen Zeitanforderungen
- Parallele Ausführung redundanter Software-Teile auf redundantem Mehrrechnersystem
- Zyklische Abwechslung der diversitären Teile

Vergleich schwierig

- Zwei Algorithmen mit unterschiedlichen Verarbeitungszeiten
- Beide Ergebnisse können korrekt sein, obwohl Werte unterschiedlich sind

**Frage zu Kapitel 2.1**

Ein Motor soll drehzahl geregelt werden. Dafür wird der Eingangsstrom mit einem induktiven Sensor und die Drehzahl an der Welle mit einer Lichtschranke erfasst. Im Falle eines zu starken Eingangsstroms soll der Motor über einen Schütz automatisch abgeschaltet werden.

Welche Arten von Prozess-Signalen können Sie identifizieren ?

**Antwort**

- Die Erfassung des Eingangsstroms mit einem induktiven Sensor liefert ein **analoges** Prozess-Signal.
- Die Erfassung der Drehzahl mittels Lichtschranke liefert ein **impulsförmiges** Prozess-Signal.
- Der Abschaltbefehl stellt ein **digitales** Prozess-Signal dar.

**Frage zu Kapitel 2.5**

Die einzelnen Module einer SPS sind untereinander mit einem herstellerspezifischen Bus verbunden.

Für die Kommunikation zwischen Feldgeräten wird oft ein Feldbus eingesetzt.

Erklären Sie an diesem Beispiel den Unterschied zwischen einem „offenen“ und einem „geschlossenen“ Kommunikationssystem.

**Antwort**

Bei dem SPS-Bussystem handelt es sich um ein **geschlossenes Kommunikationssystem**. Darunter versteht man ein herstellerspezifisches System, in welchem nur Geräte eines Herstellers kommunizieren können.

Ein Feldbussystem ist ein offenes Kommunikationssystem, bei dem Geräte unterschiedlicher Hersteller kommunizieren können.

### Frage zu Kapitel 2.5

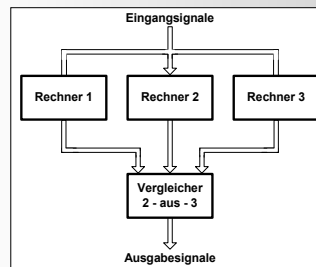
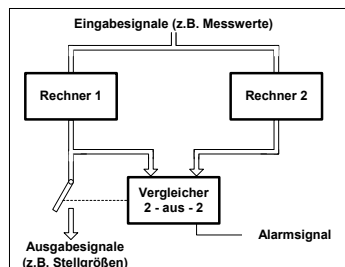
Es gibt eine Reihe unterschiedlicher Strukturen für ein Kommunikationssystem. Welchen der folgenden Aussagen würden Sie zustimmen?

### Antwort

- Eine Busstruktur verursacht den geringsten Verkabelungsaufwand
- Eine Busstruktur ist schneller als eine Netzstruktur
- Eine Netzstruktur kann eine größerer Datenmenge parallel übertragen als eine Busstruktur
- Eine Busstruktur hat kürzere Reaktionszeiten als eine Netzstruktur
- Eine Netzstruktur ist einfacher zu erweitern als eine Sternstruktur
- Eine Busstruktur eignet sich nur für dezentrale Systeme

### Frage zu Kapitel 2.6

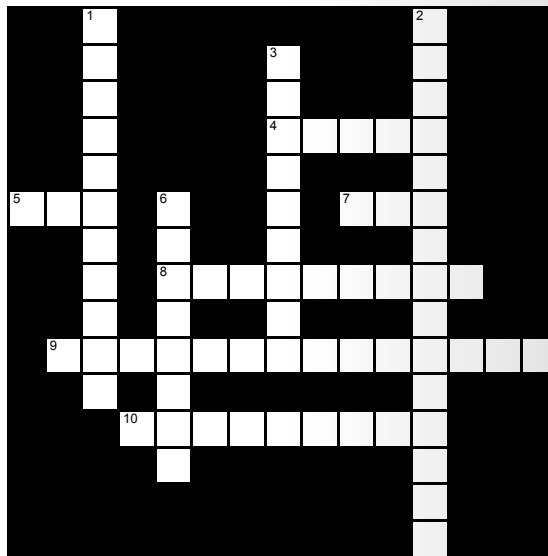
In einem Flugzeug werden wichtige Systemgrößen mehrfach berechnet. Würden Sie dafür ein Doppelrechnersystem oder ein Dreifachrechnersystem mit statischer Redundanz verwenden ?



### Antwort

Ein Doppelrechnersystem mit statischer Redundanz ist nicht fehlertolerant, da beim Auftreten eines Fehlers der Betrieb abgebrochen wird. Deshalb muss in diesem Fall ein Dreirechnersystem eingesetzt werden, da dort auf Grund der 2 aus 3 Entscheidung eine Fehlertoleranz vorhanden ist.

## Kreuzworträtsel zu Kapitel 2



## Kreuzworträtsel zu Kapitel 2

### Waagrecht

- 4 Taktgeber (5)
- 5 Kommunikations-Grundstruktur (3)
- 7 Abkürzung für Rechner, die speziell auf ein industrielles Einsatzfeld ausgelegt sind. (3)
- 8 Mehrfaches Vorhandensein von Hardware für die gleiche Aufgabe(9)
- 9 Erhalt der Funktionsfähigkeit trotz Auftreten eines Fehlers (14)
- 10 Örtlich verteilt (9)

### Senkrecht

- 1 Verschiedenartigkeit von Software bei gleicher Funktion (11)
- 2 Ein-Chip-Computer (15)
- 3 Unterbrechungsanforderung (9)
- 6 Gleichzeitige Datenübertragung auf mehreren Leitungen (8)