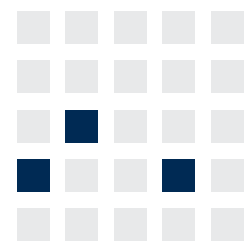


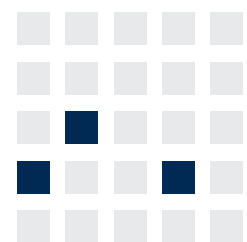


# Internet of Things / Industrial Internet

## Automatisierungstechnik II



Lehrstuhl für Wirtschaftsinformatik  
Prozesse und Systeme  
*Universität Potsdam*



Chair of Business Informatics  
Processes and Systems  
*University of Potsdam*

Univ.-Prof. Dr.-Ing. habil. Norbert Gronau  
*Lehrstuhlinhaber | Chairholder*

August-Bebel-Str. 89 | 14482 Potsdam | Germany

*Tel* +49 331 977 3322

*Fax* +49 331 977 3406

*E-Mail* [ngronau@lswi.de](mailto:ngronau@lswi.de)

*Web* [lswi.de](http://lswi.de)



Kommunikationssysteme

Echtzeitprogrammiersysteme

Speicherprogrammierbare Steuerungen SPS

Computerized Numerical Control CNC

# Lernziele

---

- Grundlegendes Verständnis über bestehende Kommunikationsstrukturen
- Besonderheiten und Charakteristika von Echtzeitsystemen
- Wesentliche Steuerungsverfahren und NC-Programmabläufe



## **Kommunikationssysteme**

Echtzeitprogrammiersysteme

Speicherprogrammierbare Steuerungen SPS

Computerized Numerical Control CNC

# Anforderungen

---

## Allgemeine Anforderungen

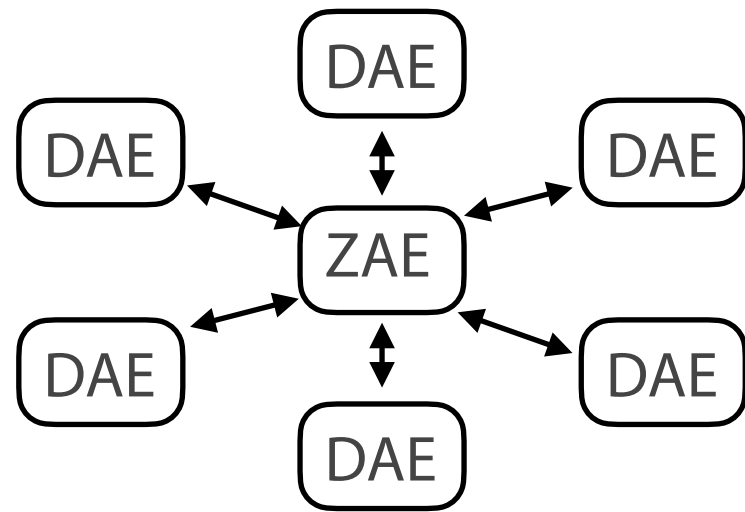
- Zuverlässige Kommunikation unter allen Umgebungsbedingungen
- Hohe Übertragungsrates
- Einfache Handhabung durch Instandhaltungspersonal
- Einfache und robuste Anschlusstechnik
- Eigensicherheit in explosionsgefährdeten Bereichen
- Robustheit gegenüber elektrostatischen, mechanischen, physikalischen und chemischen Einflüssen
- Safety / Datensicherheit

## Anwendungsbezogene Anforderungen

- Anzahl der Ein- und Ausgangssignale
- Komplexität
- Granularität
- Anlagenausdehnung
- Echtzeitanforderungen

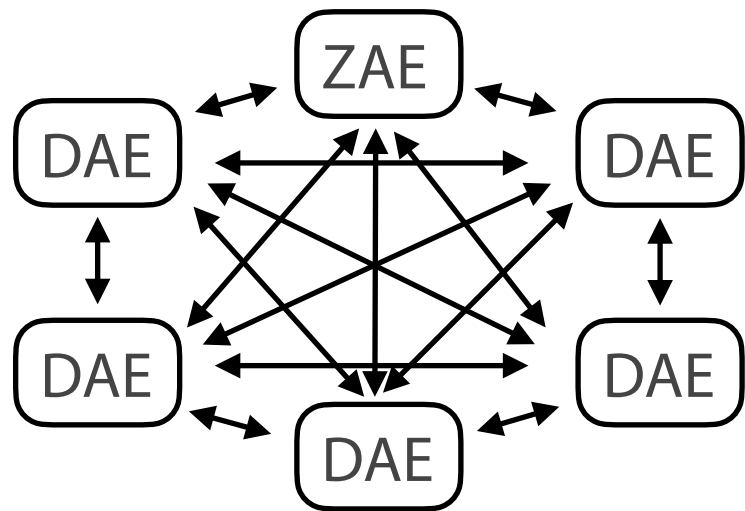
**Durch den Einsatz serieller Bussysteme für die Kommunikation zwischen Aktoren bzw. Sensoren und dem Automatisierungs-Computersystem können Projektierungs- und Installationskosten gesenkt werden.**

# Grundstrukturen der Kommunikation



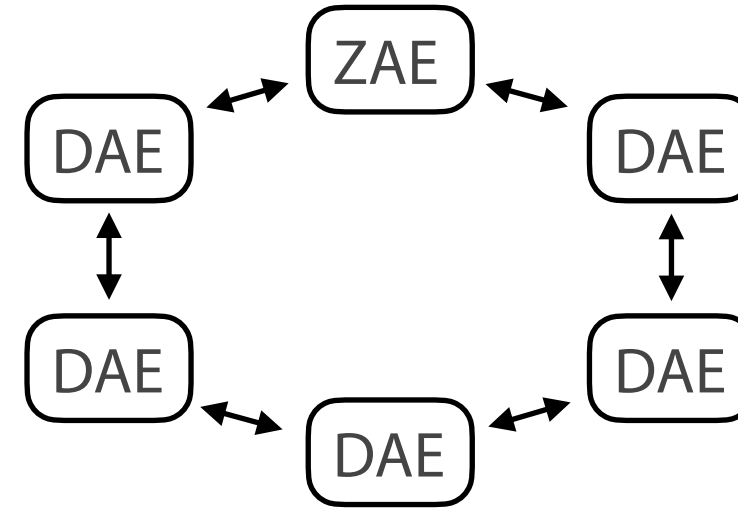
## Sternstruktur

- Ausfall der Zentraleinheit verursacht Ausfall der Kommunikation



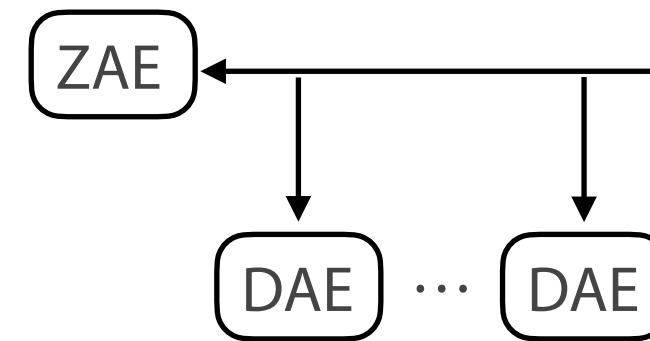
## Netzstruktur

- Parallele Informationsübertragung
- Kurze Reaktionszeiten
- Viele Schnittstellen
- Hohe Verkabelungskosten



## Ringstruktur

- Übertragung nur an direkten Nachbarn möglich

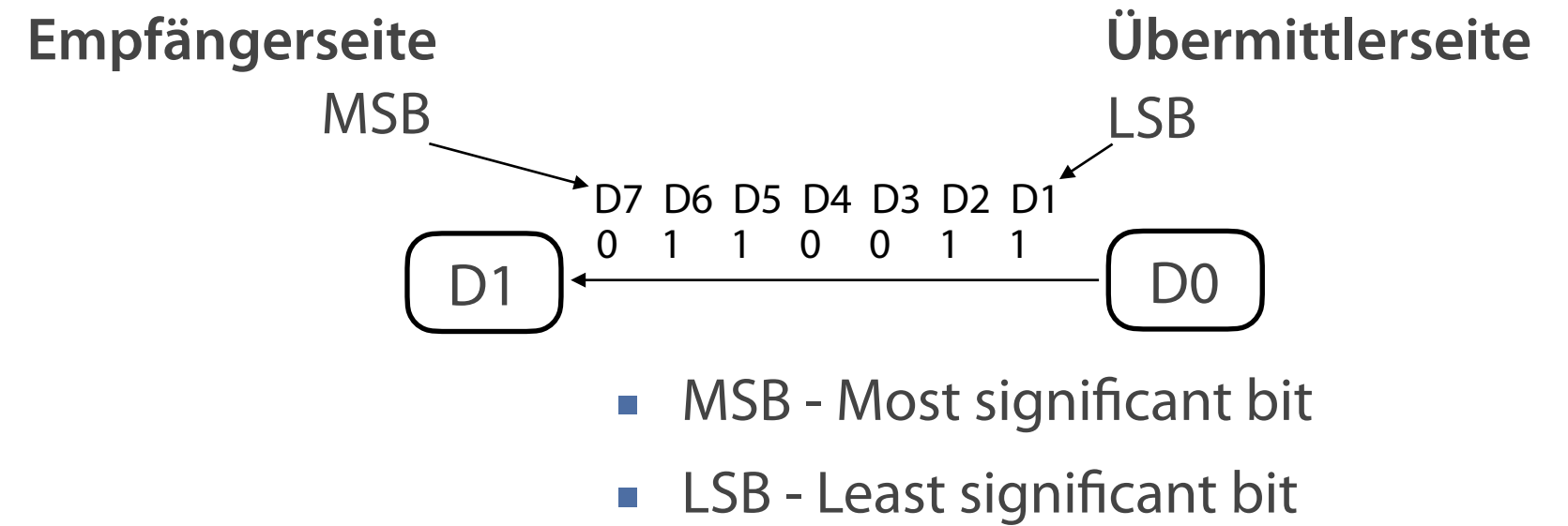
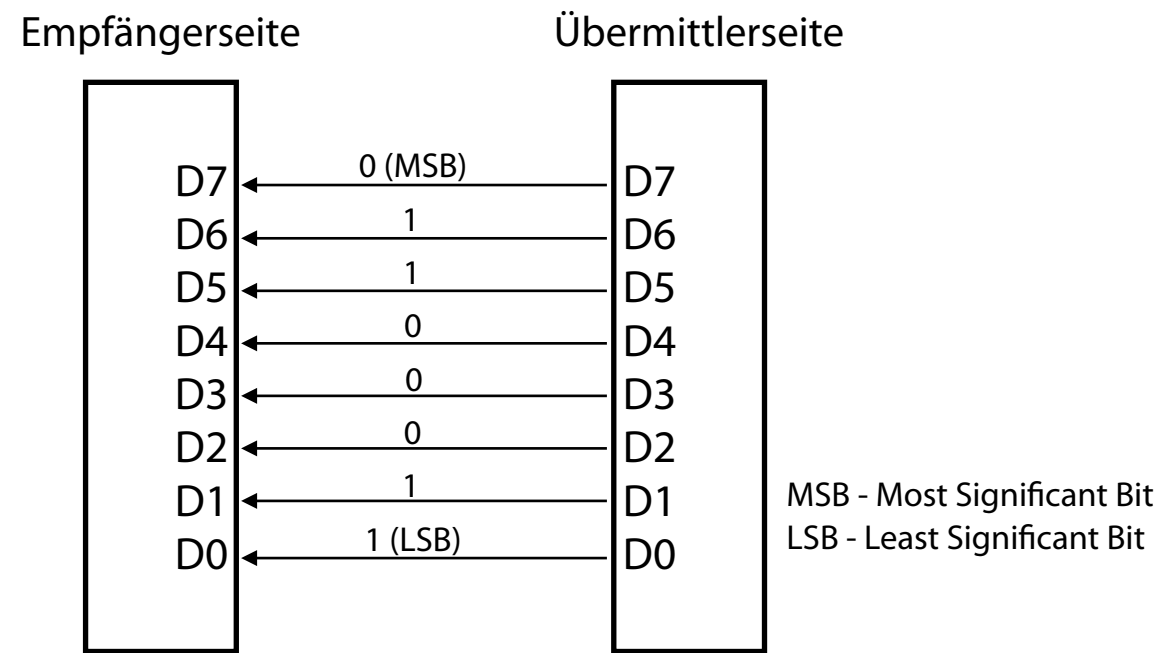


## Busstruktur

- Nur ein Teilnehmer kann jeweils senden
- Gleichzeitige Informationsaufnahme von allen Teilnehmern

Die Wahl der Struktur muss nach den spezifischen Anforderungen erfolgen.

# Arten von Bussystemen



## Parallele Busse

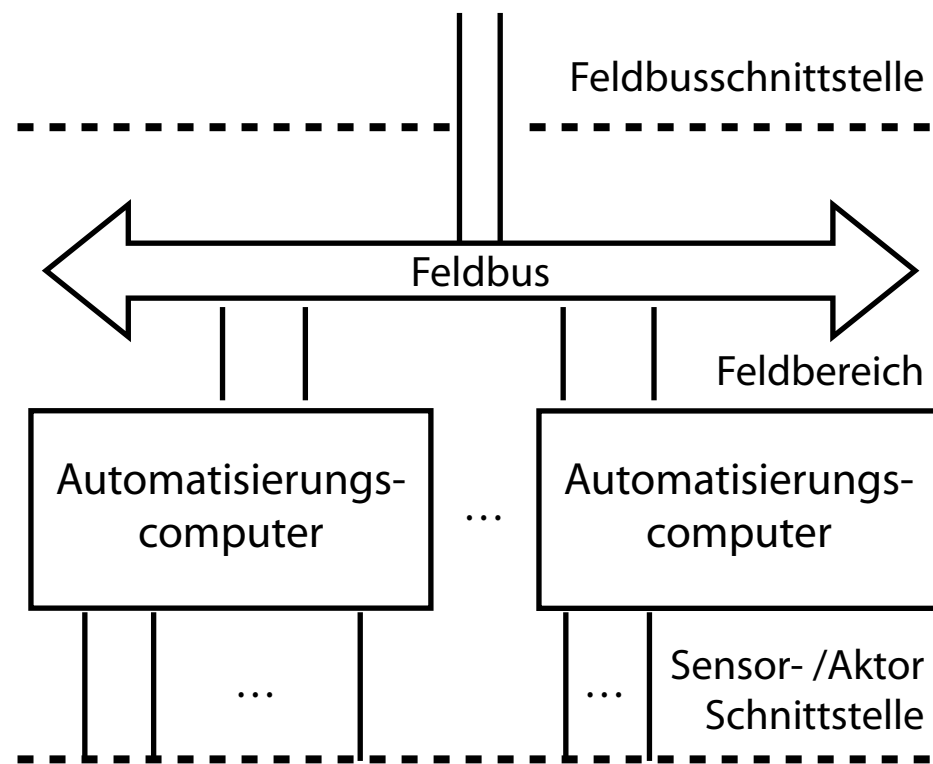
- Parallele Übertragung von Adressen, Daten und Steuersignalen
- Leitungsbündel mittels aufwendiger Verbindungen (Klemmtechnik)
- Ggf. Schwierigkeiten bei hohen Übertragungsraten und unterschiedlicher Kabellängen durch Laufzeitunterschiede

## Serielle Busse

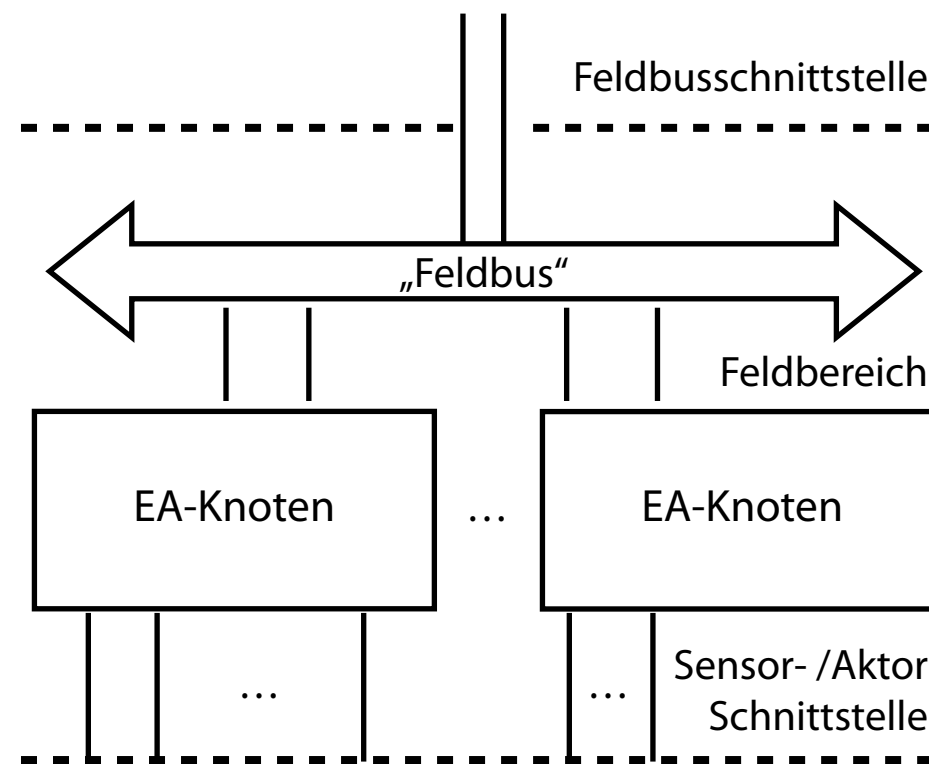
- Bits einer Nachricht werden zeitlich hintereinander übertragen
- Niedrige Leitungskosten
- Hohe Zuverlässigkeit
- Flexibilität bezüglich Leitungsprotokollen

# Feldbuskonzepte

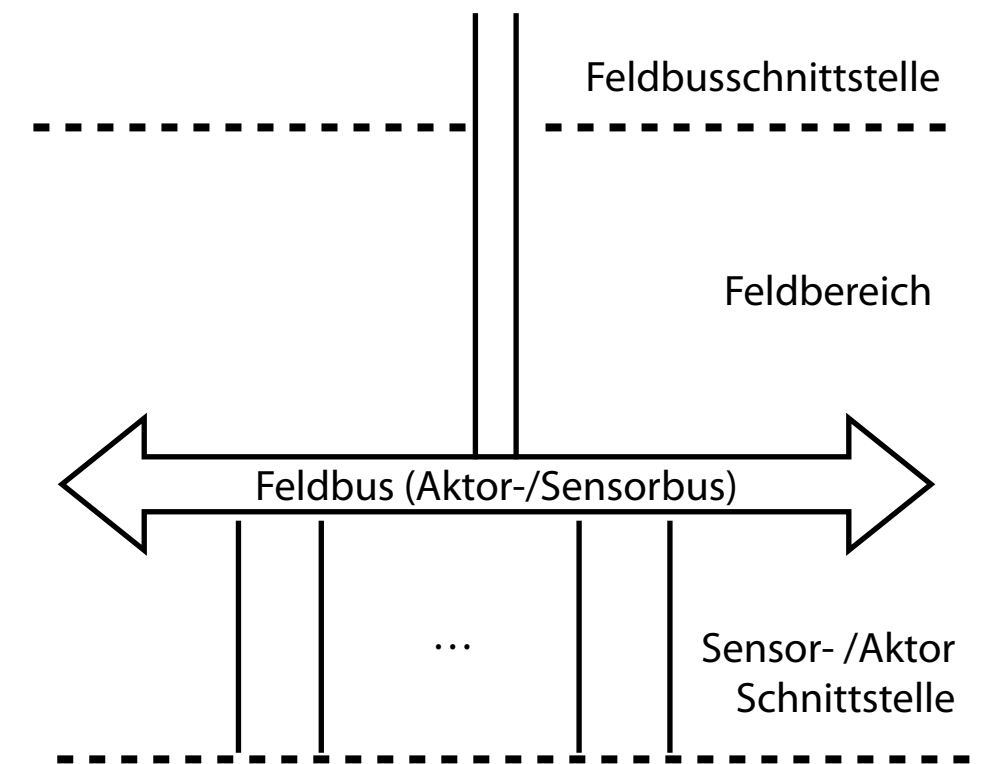
Vernetzung von dezentralen, im Feld untergebrachten Automatisierungs-Computern



Vernetzung von E/A-Knoten

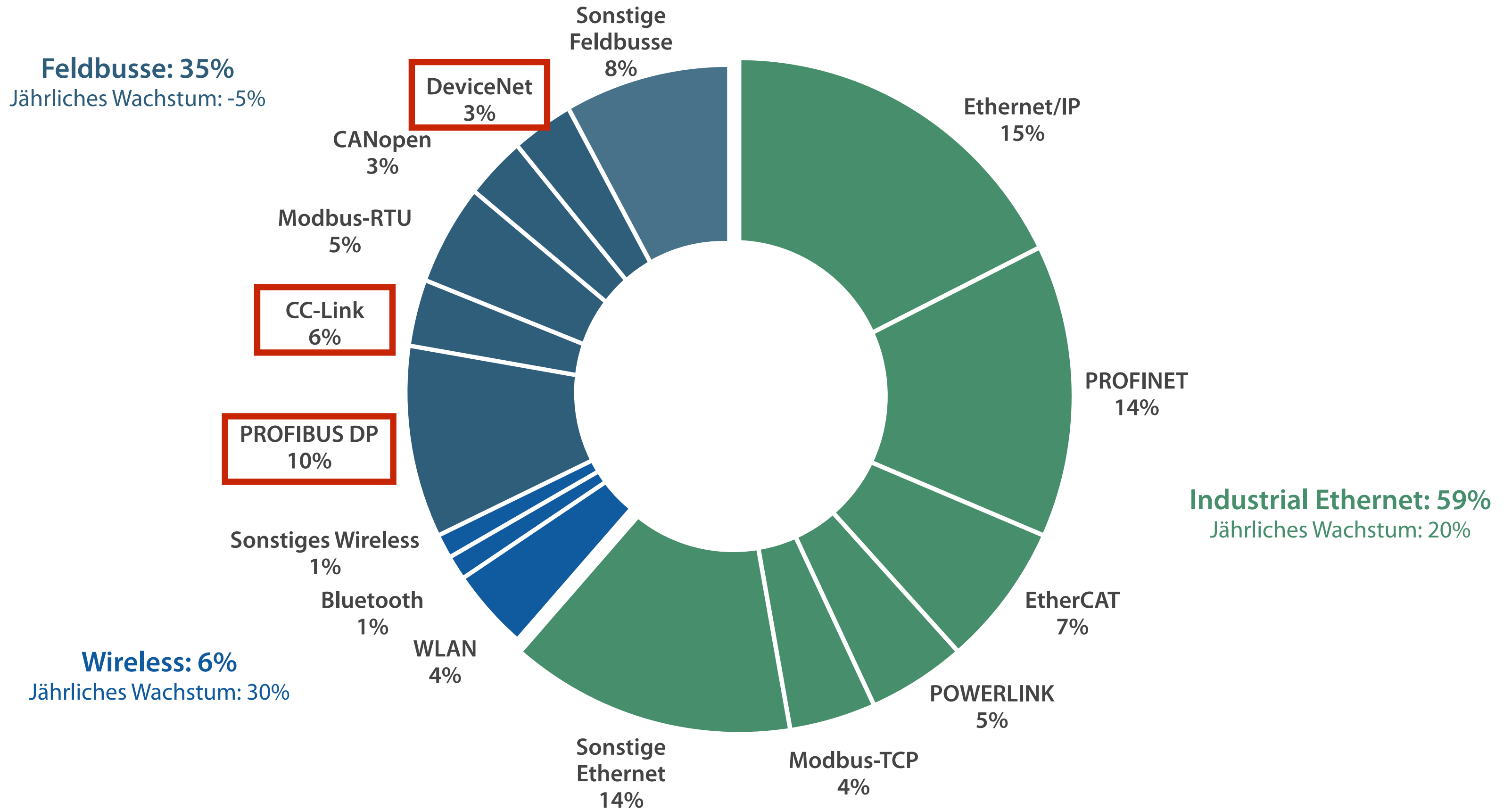


Vernetzung von Sensoren und Aktoren über Busschnittstellen





# Marktanteile von Feldbussen und Industrial Ethernet



# Vergleich ausgewählter Feldbusse

	<b>Profibus</b>	<b>DeviceNet</b>	<b>CC-Link</b>
<b>Anwendungsbereiche</b>	Feldebene	Feldebene	Feldebene
<b>Reaktionszeiten</b>	mittel	mittel	mittel
<b>Topologie</b>	Bus	Bus	Bus, Stern, T-Branch
<b>Übertragungsmedium</b>	Kabel (2-adrig, geschirmt, verdrillt), Lichtwellenleiter	Kabel (4-adrig, geschirmt, verdrillt)	Kabel (3-adrig, geschirmt, verdrillt), Kabel (5-adrig, geschirmt, verdrillt, inkl. Spannungsversorgung)
<b>Übertragungsverfahren</b>	RS-485, LWL oder IEC 61158-2 (MBP)	CAN	RS-485
<b>Übliche Anschlusstechniken</b>	Sub-D (9-polig), M12 für IP65	Klemme, Mini- und Micro-Style	Klemme, Sub-D (9-polig)
<b>Max. Teilnehmerzahl</b>	126	64	64
<b>Max. Ausdehnung je Segment</b>	100 m (12 Mbit/s), 1200 m (9,6 kbit/s)	100 m (500 kbit/s), 1000 m (62,5 kbit/s)	100 m (10 Mbit/s), 1200 m (156 kbit/s)
<b>Energieversorgung der Teilnehmer über den Bus</b>	DP: nein PA: optional	ja, 24 Volt	optional 24 Volt
<b>Einsatz im Exbereich</b>	DP: nein PA: ja	nein	nein

# Industrial Ethernet

---

## Zielsetzung

- Nutzung von Ethernet auf der Feldebene
- Durchgängiger Einsatz des Kommunikationssystems von strategischer bis operativer Ebene in Produkt und Anlagenautomatisierung

## Herausforderungen

- Lange Latenzzeit durch Netzwerktopologie
- Nicht deterministischer Buszugriff
- Geringe mechanische Robustheit

## Vorteile des Industrial Ethernet

- **Höhere Datenübertragungsraten:**  
Schnelle Übertragung von Daten, was insbesondere in der industriellen Automatisierung wichtig ist (Echtzeitdatenverarbeitung und -übertragung)
- **Erhöhte Flexibilität:**  
Mit Industrial Ethernet können Geräte und Systeme flexibler miteinander verbunden und vernetzt werden, was zu einer höheren Effizienz und Produktivität führt
- **Geringere Kosten:**
  - Industrial Ethernet nutzt Standard-Netzwerktechnologien die kosteneffizienter sind als proprietäre Systeme
  - Ethernet-basierte Netzwerke können leicht erweitert werden, ohne dass zusätzliche Hardware erforderlich ist

**Für die Bewältigung der Herausforderungen und die Realisierung der Benefits gibt es verschiedene technische Entwicklungen, wie die Entwicklung einer erweiterten Netzwerk-Topologie oder Protokollanpassungen.**



Kommunikationssysteme

**Echtzeitprogrammiersysteme**

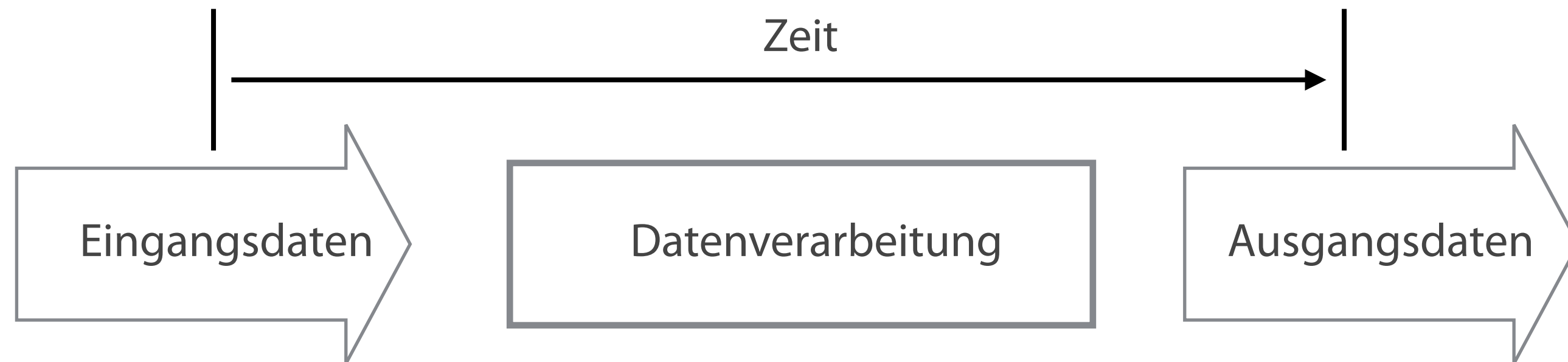
Speicherprogrammierbare Steuerungen SPS

Computerized Numerical Control CNC

# Typische Anwendungsbeispiele mit Zeitbedingungen

	<b>Beispiele unter</b>	
	<b>Absolutzeit-Bedingungen</b>	<b>Relativzeit-Bedingungen</b>
Ausführung einer Funktion zu festen Zeitpunkten	Kennfeldaufnahme an Prüfständen	Analyse von Stoffen in der Chemie
Ausführung einer Funktion in einem Toleranz-Zeitintervall	Erfassung von Regelgrößen	Messwertüberwachung auf gleitende Grenzen
Ausführung einer Funktion in einem Zeitintervall bis zu einem spätesten Zeitpunkt	Erfassung von Datentelegrammen	Erfassung von Stückgutkennungen
Ausführung einer Funktion in einem Zeitintervall von einem frühesten Zeitpunkt an	Folgesteuerung bei Chargenprozessen	Erfassung von Signalen einer Lichtschranke

Für die Umsetzung sind Echtzeit-Systeme notwendig.



**Die Datenverarbeitung liefert ihre Ergebnisse garantiert innerhalb eines dem steuernden Prozess angemessenen Zeitraums.**

# Anforderungen an Echtzeitsysteme

---

## Rechtzeitigkeit

- System muss zur richtigen Zeit reagieren

## Gleichzeitigkeit

- System muss auf mehrere Dinge gleichzeitig reagieren

## Verlässlichkeit

- System muss zuverlässig, sicher und verfügbar sein

## Vorhersagbarkeit

- Alle Reaktionen müssen planbar und deterministisch sein

# Unterschiede zwischen Informations- und Echtzeitsystemen

<b>Informationssysteme, z.B. MES</b>	<b>Echtzeitsysteme, z.B. Roboter</b>
datengesteuert	ereignis-/zeitgesteuert
komplexe Datenstrukturen	einfache Datenstrukturen
große Menge an Eingabedaten	kleine Menge an Eingabedaten
E/A-intensiv	rechenintensiv
hardwareunabhängig	hardwareabhängig



# Echtzeitprogrammierung

---

- Erfassung der Eingabedaten
- Verarbeitung der Daten
- Ausgabe der Ausgabedaten

## Harte Echtzeit-Systeme

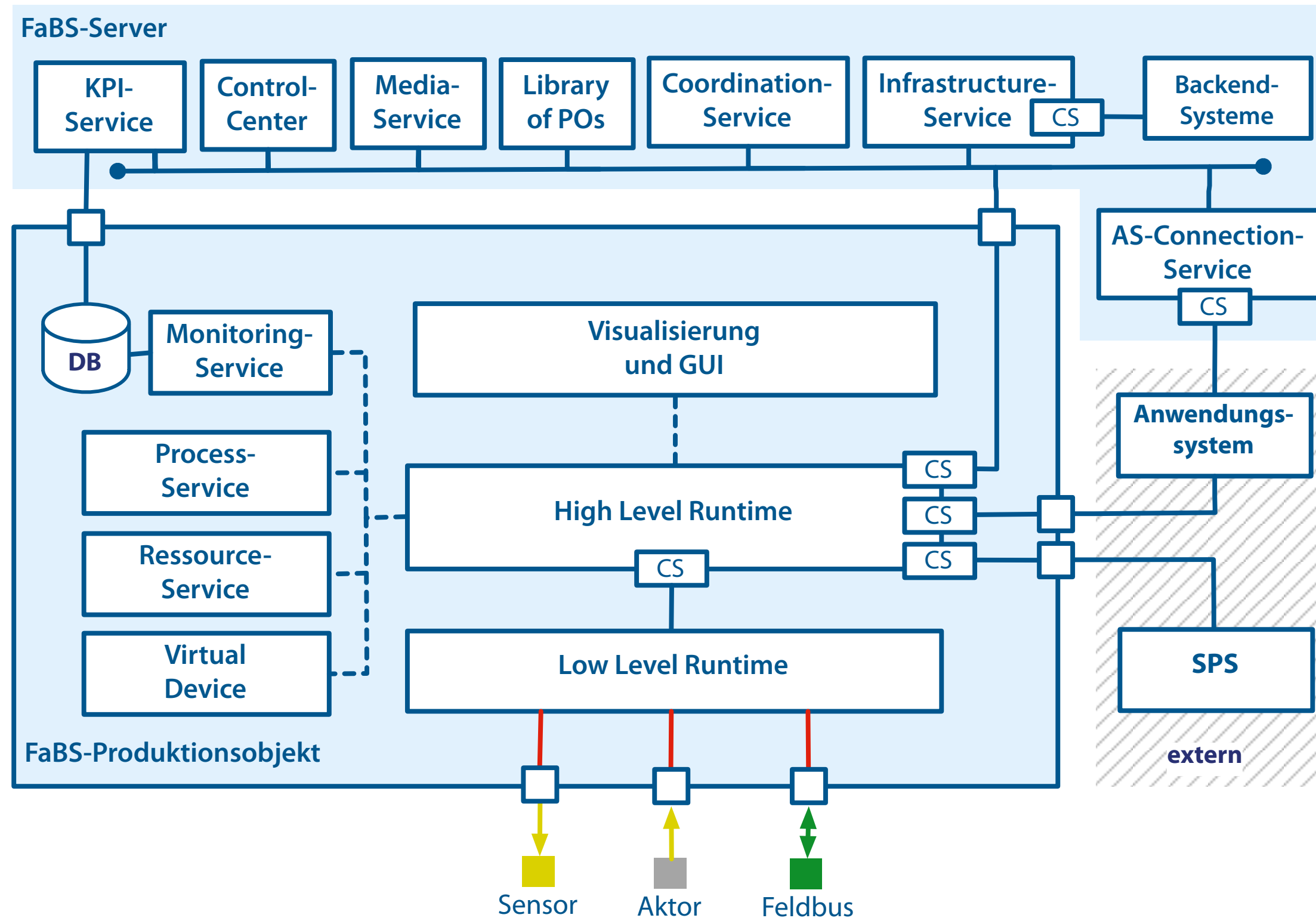
- Einhaltung von strengen Zeitschranken unabdingbar

## Weiche Echtzeit-Systeme

- Gewisse Verletzung von Zeitschranken ist tolerierbar

**Bei der Echtzeitprogrammierung werden synchrone und asynchrone Programmierverfahren unterschieden.**

# Perspektive Software Fabrikbetriebssystem (FabOS)



# Synchrones Programm

---

## Vorgehen

- Synchronisierung der zyklisch auszuführenden Teilprogramme mit einem Zeitraster
- Zeitraster über Echtzeit-Uhr, Unterbrechungssignal zum Aufruf über Teilprogramme
- fest vorgegebene Reihenfolge des Ablaufs der Teilprogramme

## Eignung

- Gut geeignet für Echtzeit-Systeme mit zyklischen Programmabläufen
- Ungeeignet für die Reaktion auf zeitlich nicht vorhersehbare (asynchrone) Ereignisse

## Eigenschaften

- Näherungsweise Erfüllung der Rechtzeitigkeit
- Erfüllung der Gleichzeitigkeit, wenn Zykluszeit  $T$  klein gegenüber den Zeitabläufen im technischen Prozess ist
- Im Normalfall deterministisches Verhalten
- kein komplexes Organisationsprogramm
- etwas aufwändigere Planung (Entwicklung)

## Nachteile

- Änderung der Aufgabenstellung bedeutet Änderung der gesamten Programmstruktur

**Bei einem synchronen Programm wird die Planung des zeitlichen Verhaltens zyklisch auszuführender Teilprogramme vor der Ausführung durchgeführt.**

# Asynchrones Programm (Parallelprogramm)

---

## Vorgehen

- Aufruf der Teilprogramme, wenn Zeitbedingungen erfüllt sind
- Gleichzeitige Ausführung wird durch die Zuordnung von Prioritätsnummern sequenzialisiert

## Eignung

- Gut geeignet für die Reaktion auf zeitlich nicht vorhersehbare (asynchrone) Ereignisse

## Eigenschaften

- Forderung nach Rechtzeitigkeit nur näherungsweise erfüllt
- Zeitbedingungen umso besser erfüllt, je höher die Priorität des jeweiligen Teilprogramms
- Ist-Zeitablauf kann sich gegenüber Soll-Zeitablauf stark verschieben

## Nachteile

- Aufeinanderfolge der Teilprogramme nicht deterministisch, stattdessen dynamische Einstellung
- Komplexität im Verwaltungsprogramm
- Programmablauf schwer durchschaubar

**Bei einem asynchronen Programm steuert ein Organisationsprogramm (Echtzeitbetriebssystem) den zeitlichen Aufruf der Teilprogramme während des Ablaufes.**

# Unterprogramm vs. Rechenprozess

---

## Unterprogramm

- Ausführung des ausführenden Programms wird unterbrochen
- Ausführung des Unterprogramm (einmalige Ausführung der Befehlsfolge)
- Fortsetzung des ausführenden Programms

## Rechenprozesse

- Vom Echtzeit-Betriebssystem gesteuerter Vorgang der Abarbeitung eines sequenziellen Programms
- Gleichzeitige Ausführung des aufrufenden Programms und des aufgerufenen Rechenprozesses
- Beginnt mit Eintrag in eine Liste des Echtzeit-Betriebssystems und endet mit dem Löschen aus dieser Liste
- Existiert nicht nur während der Ausführung der Befehle, sondern auch während geplanter oder erzwungener Wartezeiten
- Unterscheidung nach parallelen, sequentiellen, nebenläufigen und simultanen Aktionen von Rechenprozessen

# Aktionen von Rechenprozessen

---

## Sequentielle Aktionen

- Aktionen in Rechenprozessen sind in bestimmter Reihenfolge angeordnet
- Bsp.: Ein Computer führt nacheinander verschiedene Rechenoperationen aus

## Parallele Aktionen

- Aktionen in Rechenprozessen können gleichzeitig ablaufen
- Aufspaltung in nebenläufige und simultane Aktionen
- Bsp.: Ein Computer führt parallel verschiedene Rechenoperationen aus

## Nebenläufige Aktionen (äußere Parallelität)

- Zwei Aktionen aus zwei verschiedenen Rechenprozessen können gleichzeitig ablaufen
- Bsp.: Ein Computer führt parallel verschiedene Rechenoperationen unterschiedlicher Module aus

## Simultane Aktionen (innere Parallelität)

- Zwei Aktionen aus einem Rechenprozess können gleichzeitig ablaufen
- Bsp.: Ein Computer führt parallel verschiedene Rechenoperationen aus einem Modul aus



# Task vs. Thread

---

<b>Task</b>	<b>Thread</b>
Besitzer von Betriebsmitteln	Kann außer Prozessor selbst keine Betriebsmittel besitzen; verfügt über alle Betriebsmittel der Task, der er angehört
Eigener Adressraum	Adressraum der Task, der er angehört
Enthält einen oder mehrere Threads	Element einer Task
Kommunikation über die Taskgrenzen hinaus, bevorzugt über Botschaften	Kommunikation zwischen den Threads, bevorzugt über gleiche Daten

# Bewertung der Echtzeit-Architekturen

---

## Ereignisgesteuerte Architekturen

- Alle Aktivitäten als Folge von Ereignissen
  - Aktivierung von Tasks
  - Senden von Nachrichten
- Unterstützung durch Echtzeitbetriebssysteme
- Nicht-deterministisches Verhalten
- Flexibel bezüglich Veränderungen

## Zeitgesteuerte Architekturen

- Periodischer Durchführung aller Tasks und Kommunikationsaktionen
- Abtastung externer Zustandsgrößen zu festgelegten Zeitpunkten
- Wenig flexibel bei Änderungen
- Einfach analysierbar



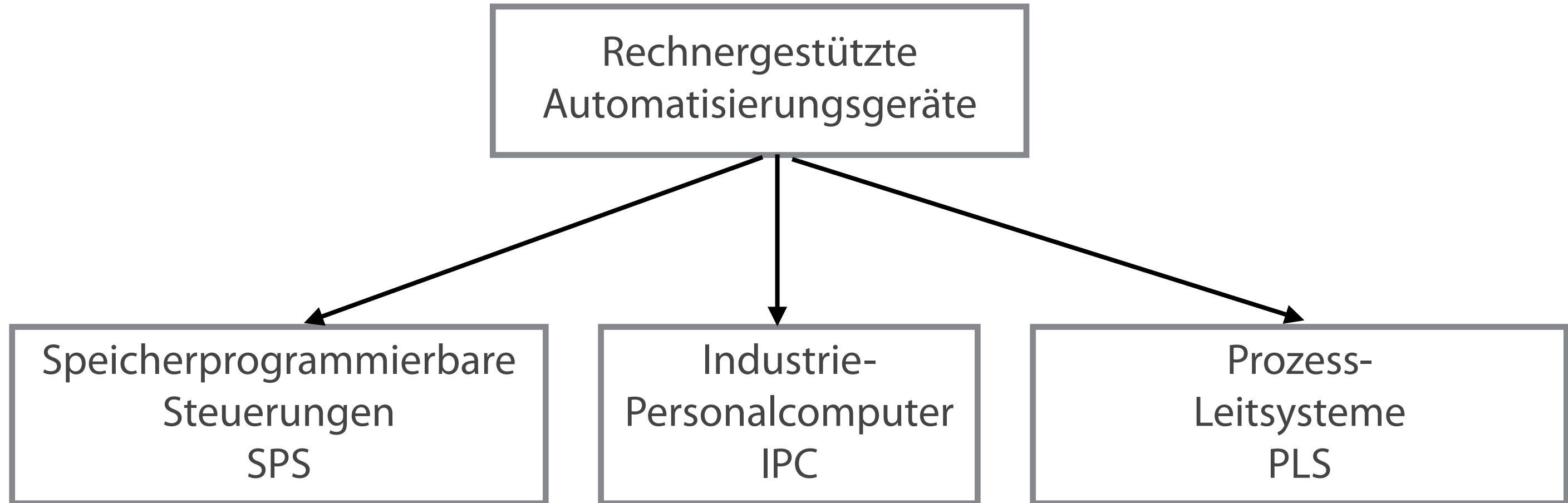


Kommunikationssysteme

Echtzeitprogrammiersysteme

**Speicherprogrammierbare Steuerungen SPS**

Computerized Numerical Control CNC



# Automatisierungscomputer

---

## Speicherprogrammierbare Steuerung SPS

- Einsatz von Geräten mit Zertifizierungen
- Proprietäre Hardware mit oft langfristigen Zusagen für die Ersatzteillieferung
- Verknüpfung binärer Signale in einfacher Darstellungen

## Personal Computer PC / Industrial Personal Computer IPC

- Programmierung in Hochsprache
- Einsatz von Echtzeit-Betriebssystem als Stand Alone oder als Ergänzung möglich
- Mögliche Einsatzgebiete: Prozess-Visualisierung, -Auswertung, -Überwachung, Leitstandsaufgaben

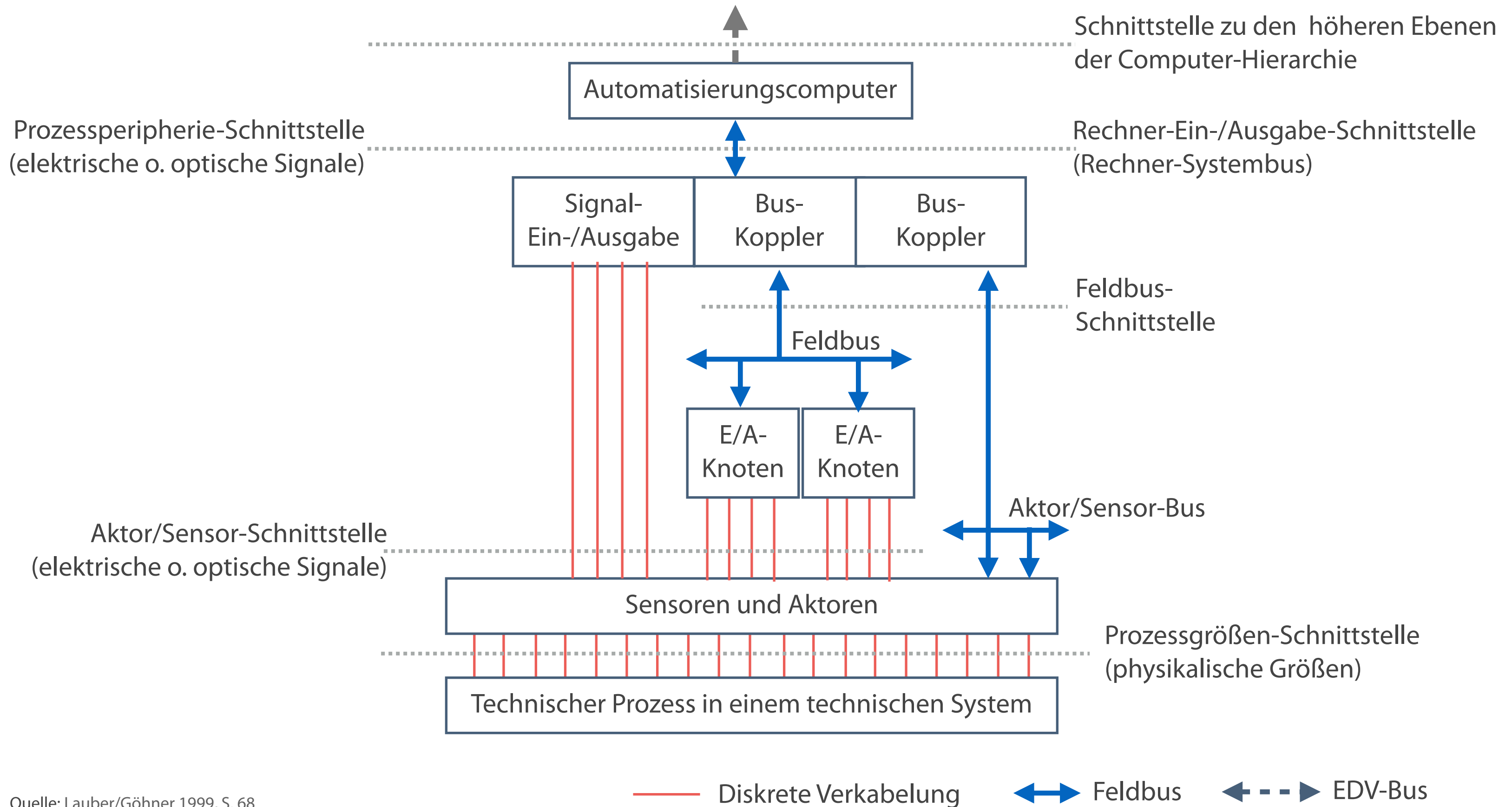
## Mikrocontroller

- „Ein-Chip-Computer“ mit niedrigem Preis
- Hochintegrierte Bausteine
- Verwendung für Massenprodukte
- Hohe Zuverlässigkeit und Lebensdauer

## Prozessleitsystem PLS

- Verteiltes, über Bus-Systeme verbundenes Rechnersystem
- Kopplung mit SPS-Rechnern
- Einsatz vorkonfigurierter, vom Hersteller des PLS entwickelten Programmbausteinen
- Konfigurierung durch Anwender

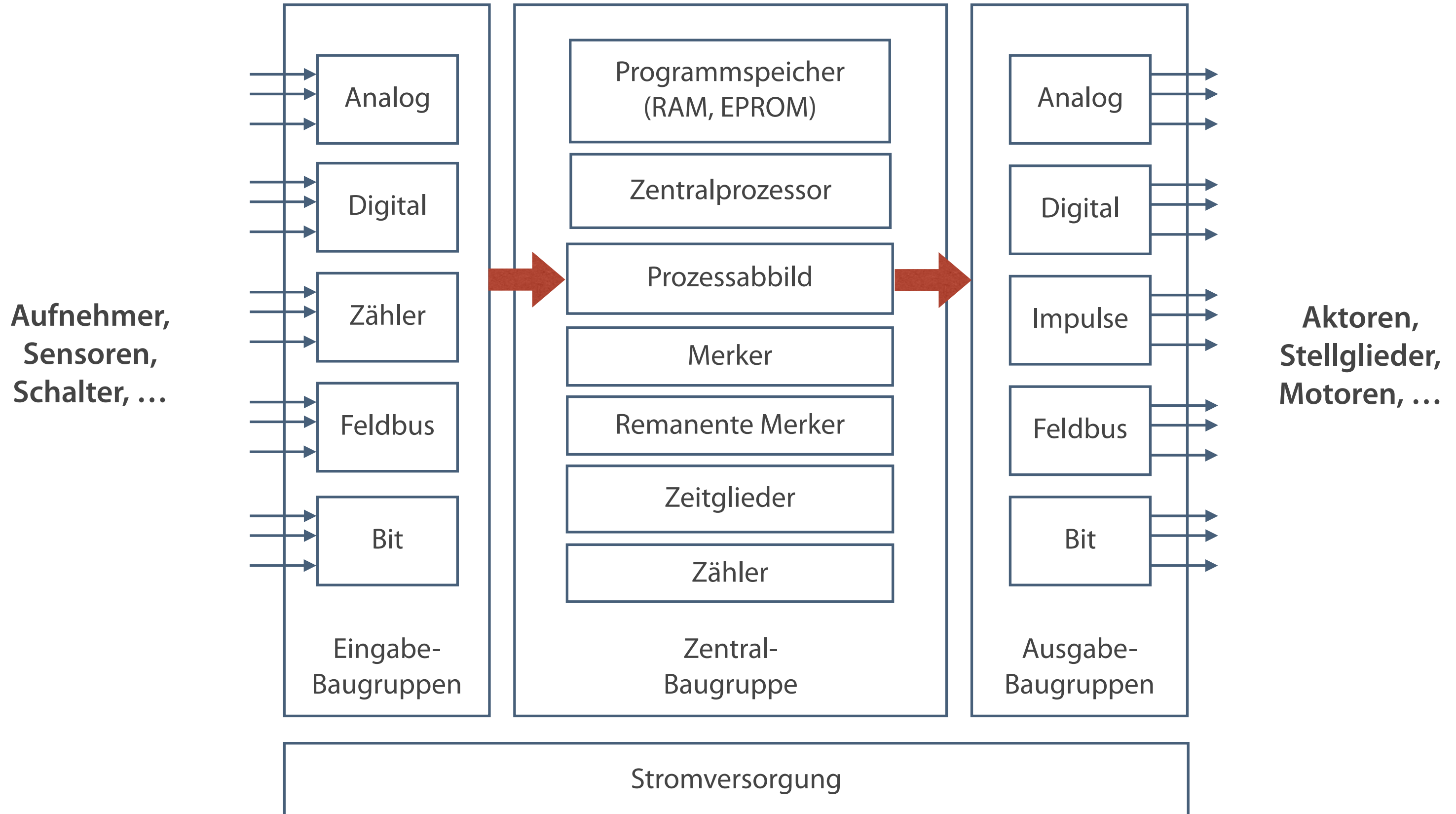
# Kopplung Automatisierungscomputer mit technischem Prozess und zugehörigen Schnittstellen



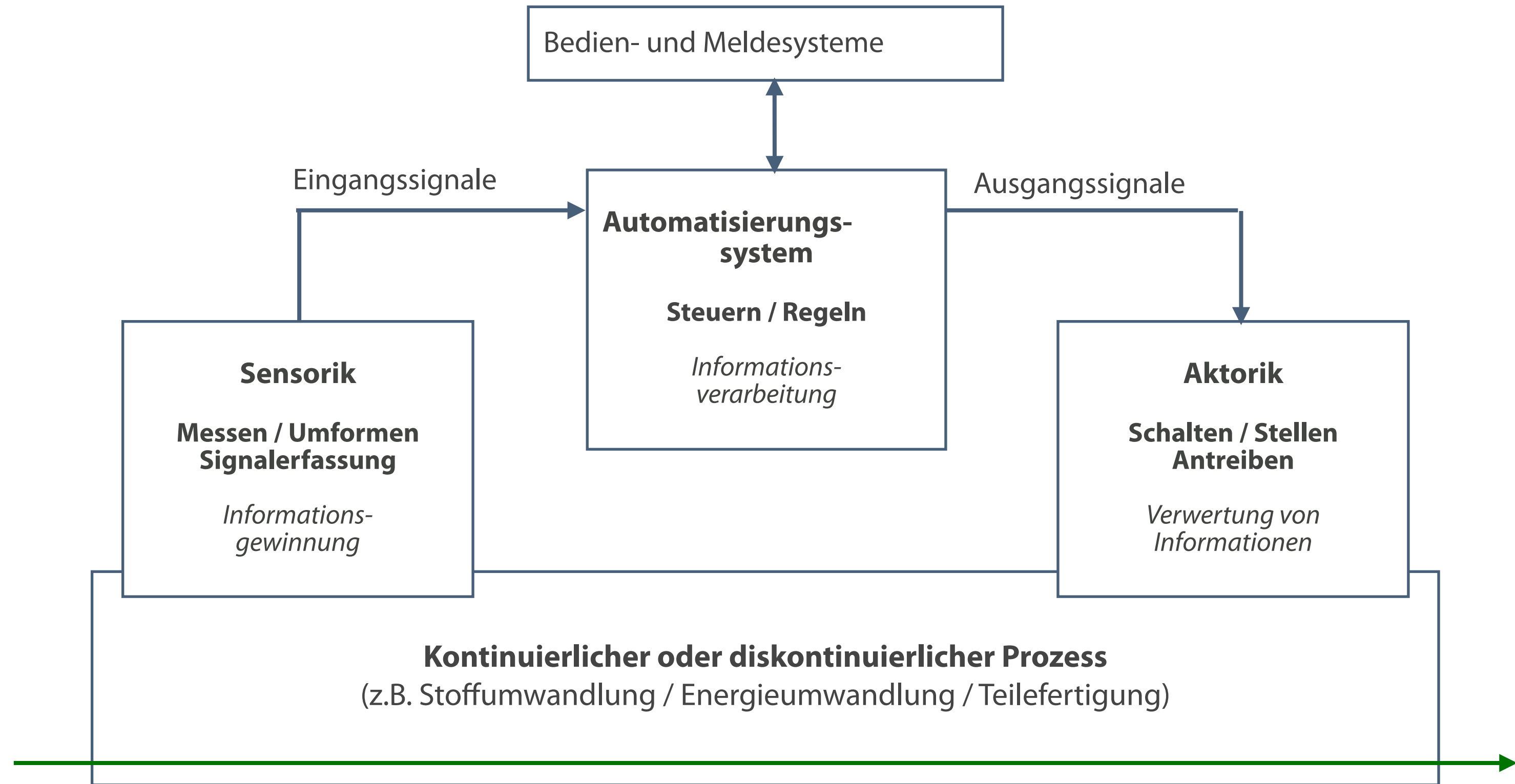
# Abgrenzung

	SPS	IPC	PLS
<b>Aufbau</b>	<ul style="list-style-type: none"> <li>• speziell für die Steuerungstechnik konzipierte Rechnersysteme</li> <li>• breiten Palette an Ein-/Ausgabebaugruppen</li> </ul>	<ul style="list-style-type: none"> <li>• PC mit industrietauglichem elektrischen und mechanischen Aufbau</li> <li>• Prozesskopplung über Ein-/Ausgabebaugruppen der standardisierten Rechnerbusschnittstellen</li> </ul>	<ul style="list-style-type: none"> <li>• dezentralisierte und hierarchisch aufgebaute Rechnersysteme</li> <li>• Aufbau mit SPS- und/oder IPC-Systemen</li> </ul>
<b>Verwendung</b>	<ul style="list-style-type: none"> <li>• speziell für Steuerungsaufgaben</li> <li>• Zusatzbaugruppen können auch Regelungsaufgaben erfüllen.</li> </ul>	<ul style="list-style-type: none"> <li>• Steuerungs- und Regelungsaufgaben</li> <li>• Visualisierung der automatisierten Prozessen</li> </ul>	<ul style="list-style-type: none"> <li>• Führung, Steuerung, Regelung und Überwachung komplexer Maschinen und Anlagen</li> </ul>
<b>Programmierung</b>	<ul style="list-style-type: none"> <li>• Fachsprachen: AWL, KOP, FUP, ST, AS</li> <li>• Programmentwicklung mit Entwicklungsrechner (z. B. PCs)</li> </ul>	<ul style="list-style-type: none"> <li>• Standardisierte Hochsprachen wie C/C++, Pascal, Java u. a.</li> <li>• komfortable Programmsysteme für Automatisierungsaufgaben</li> </ul>	<ul style="list-style-type: none"> <li>• einfache Konfigurierung der Programme für die Prozesskontrolle</li> <li>• komfortable Softwaresysteme für die Prozessleittechnik</li> </ul>

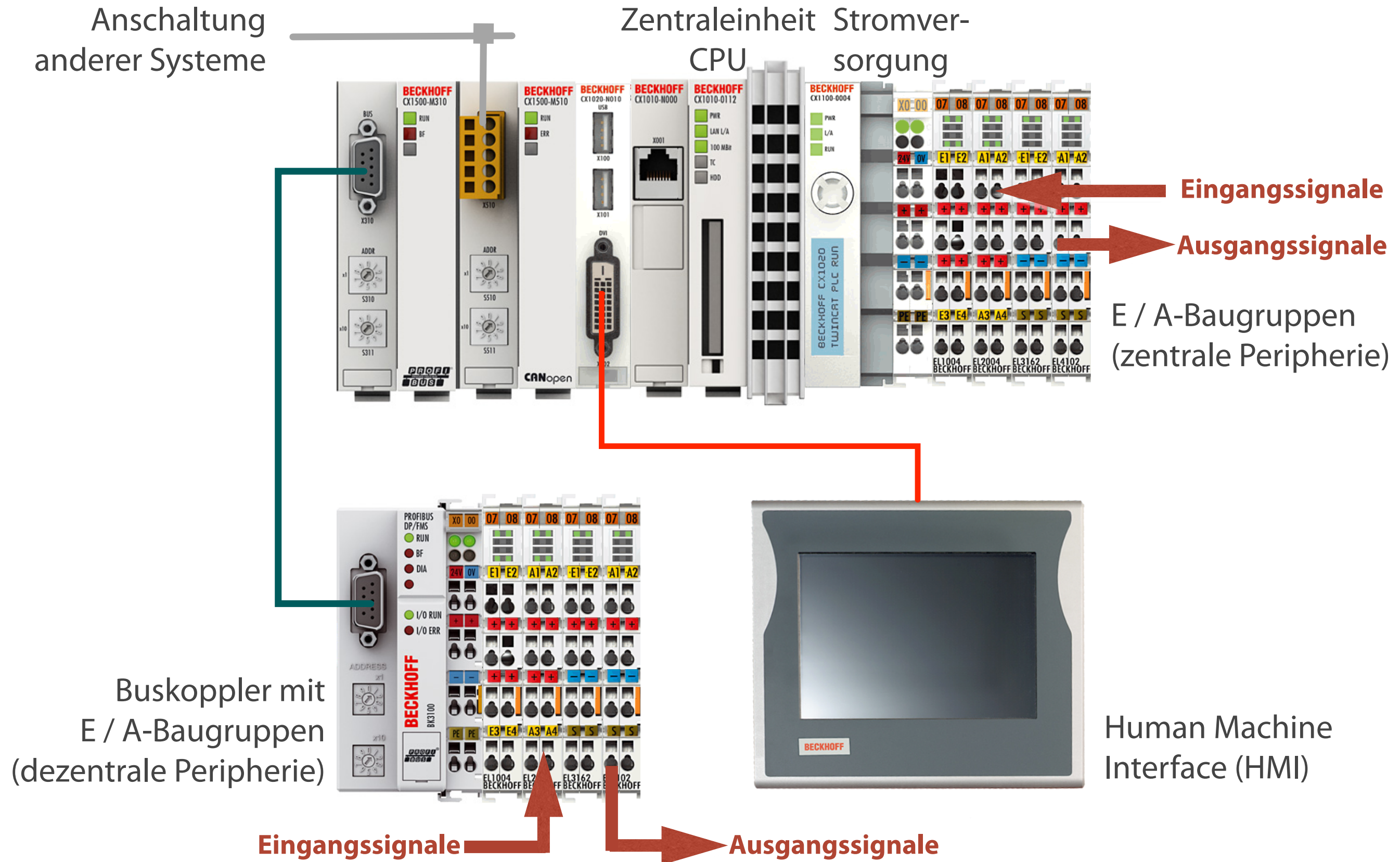
# Aufbau einer SPS



# Ausgangspunkt

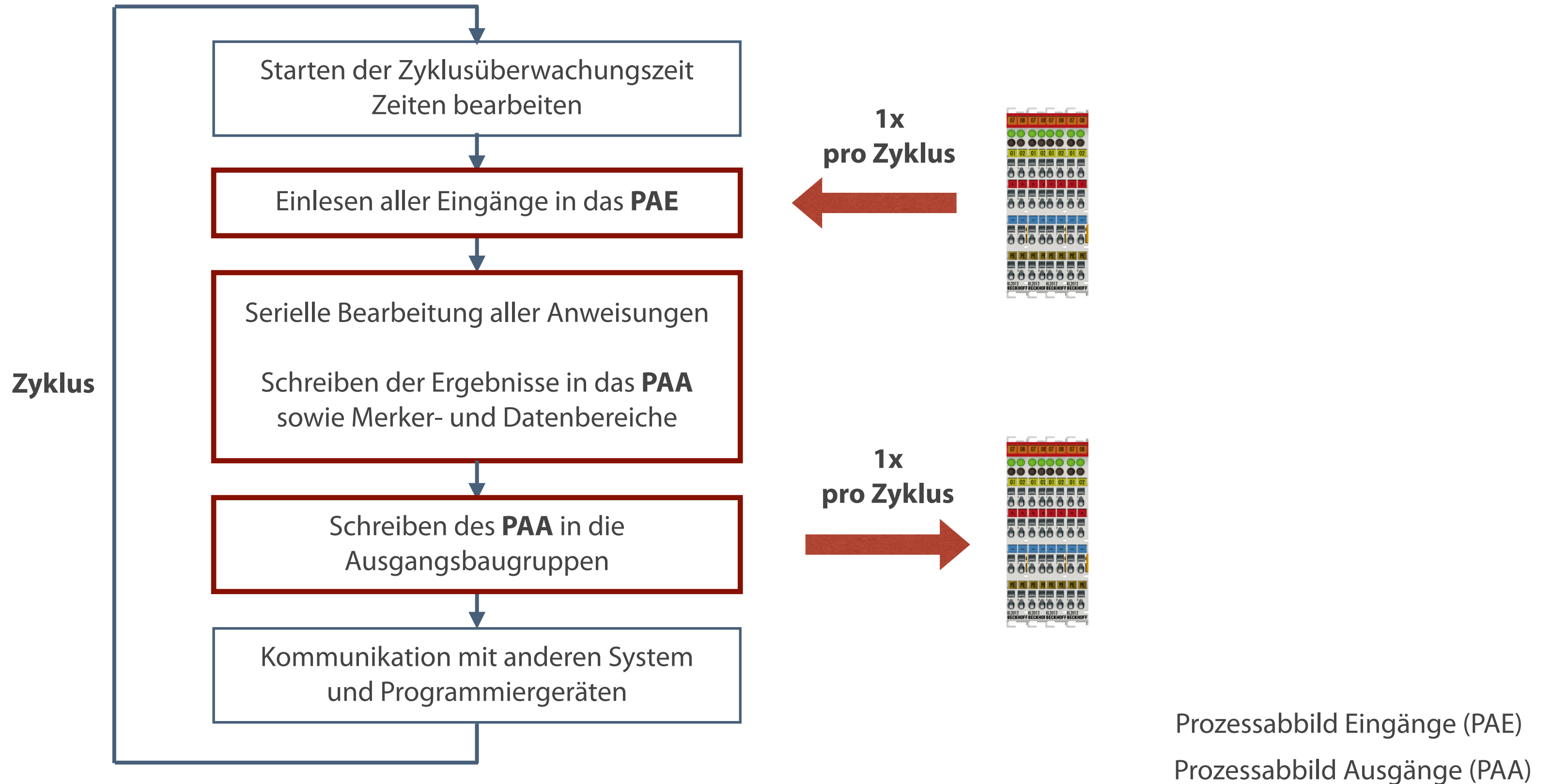


# Aufbau in der Praxis

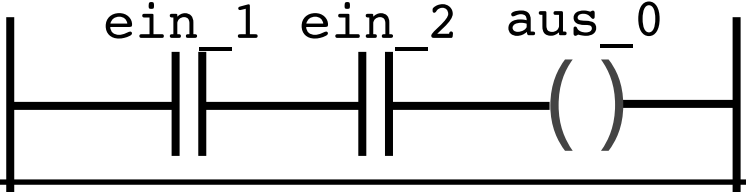
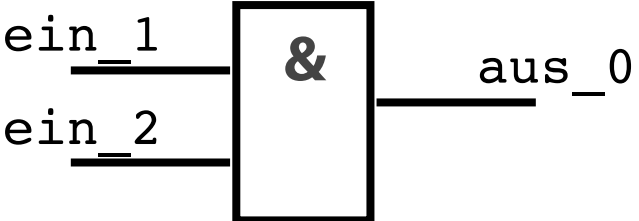
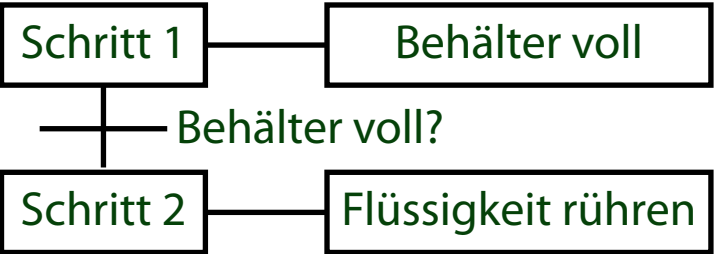




# Zyklische Abarbeitung



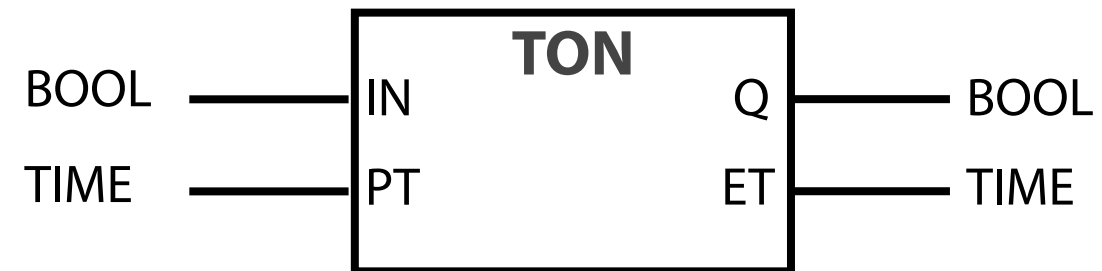
# Sprachen für die Programmierung von Speicherprogrammierbaren Steuerungen

ISO DIN EN 61131-3		DIN 19239 bzw. VDI 2880
<ul style="list-style-type: none"> <li>• AWL - Anweisungsliste</li> <li>• IL - Instruction List</li> </ul>	<pre>LD  eingang_1 AND eingang_2 ST  ausgang_0</pre>	<ul style="list-style-type: none"> <li>• AWL - Anweisungsliste</li> </ul>
<ul style="list-style-type: none"> <li>• ST - Strukturierter Text</li> <li>• ST - Structured Text</li> </ul>	<pre>C := A AND NOT B</pre>	
<ul style="list-style-type: none"> <li>• KOP - Kontaktplan</li> <li>• LD - Ladder Diagram</li> </ul>		<ul style="list-style-type: none"> <li>• KOP - Kontaktplan</li> </ul>
<ul style="list-style-type: none"> <li>• FBS - Funktionsbausteinsprache</li> <li>• FBD - Function Block Diagram</li> </ul>		<ul style="list-style-type: none"> <li>• FUP - Funktionsplan</li> </ul>
<ul style="list-style-type: none"> <li>• AS - Ablaufsprache</li> <li>• SFC - Sequential Function Chart</li> </ul>		<ul style="list-style-type: none"> <li>• Funktionsplan nach DIN 40719, Teil 6</li> </ul>

# Programmierung

## Beispiele

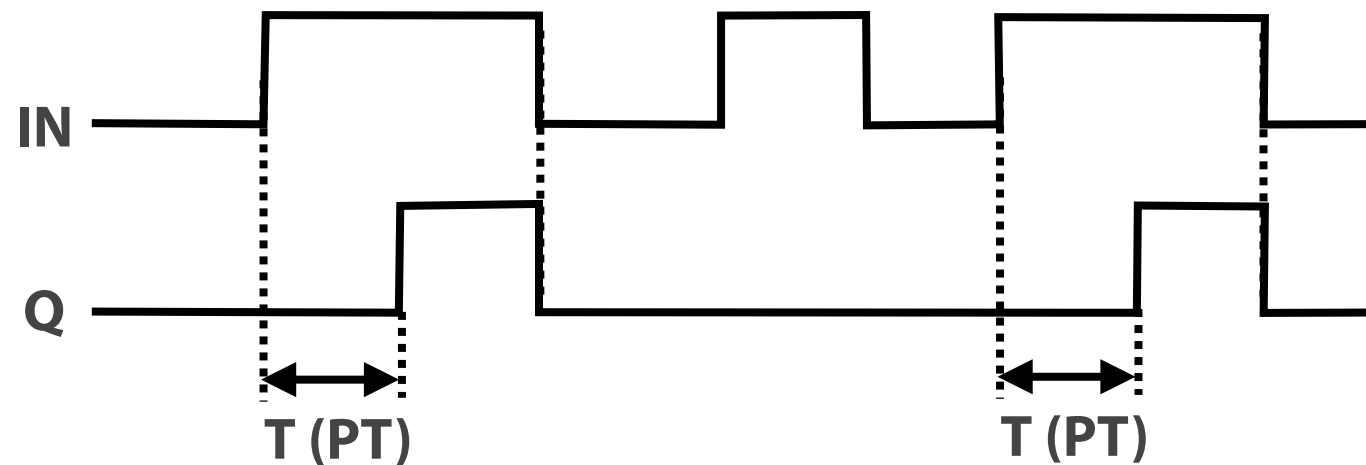
### Prototyp:



### Operanden:

- IN Startbedingung
- PT Zeitwertvorgabe
- Q Binärer Zustand des Zeitgebers
- ET Aktueller Zeitwert

### Beschreibung:



### Beispiel: um 777ms verzögert einschalten

```
PROGRAM zeit777ein

VAR
  Timer2 : TON;
  Start AT %IN0.0.0.0.0 : BOOL;
  Zeitdauer : TIME := T%777ms;
  Ausgang AT %Q0.0.0.0.0 : BOOL;
  Istzeit : TIME;
END_VAR

CAL Timer2(IN := Start,
           PT := Zeitdauer)

LD Timer2.Q
ST Ausgang
LD Timer2.ET
ST Istzeit

END_PROGRAM
```

# Programmiererstellung Ablauf

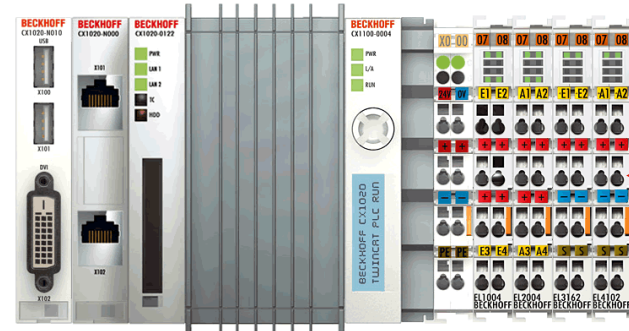
## Programmiersystem



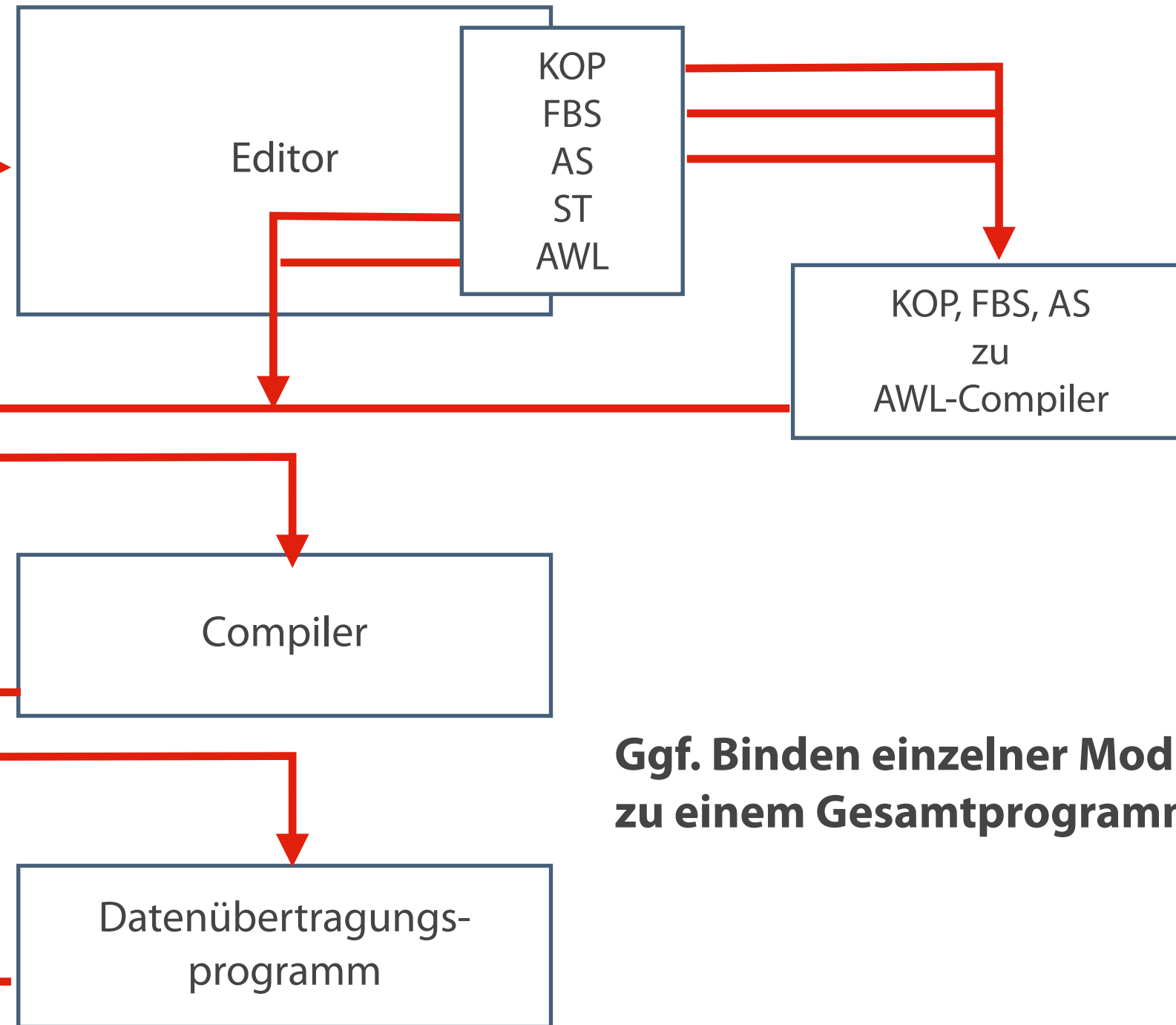
## Datenträger



## Datenträger



## SPS



**Ggf. Binden einzelner Module zu einem Gesamtprogramm**



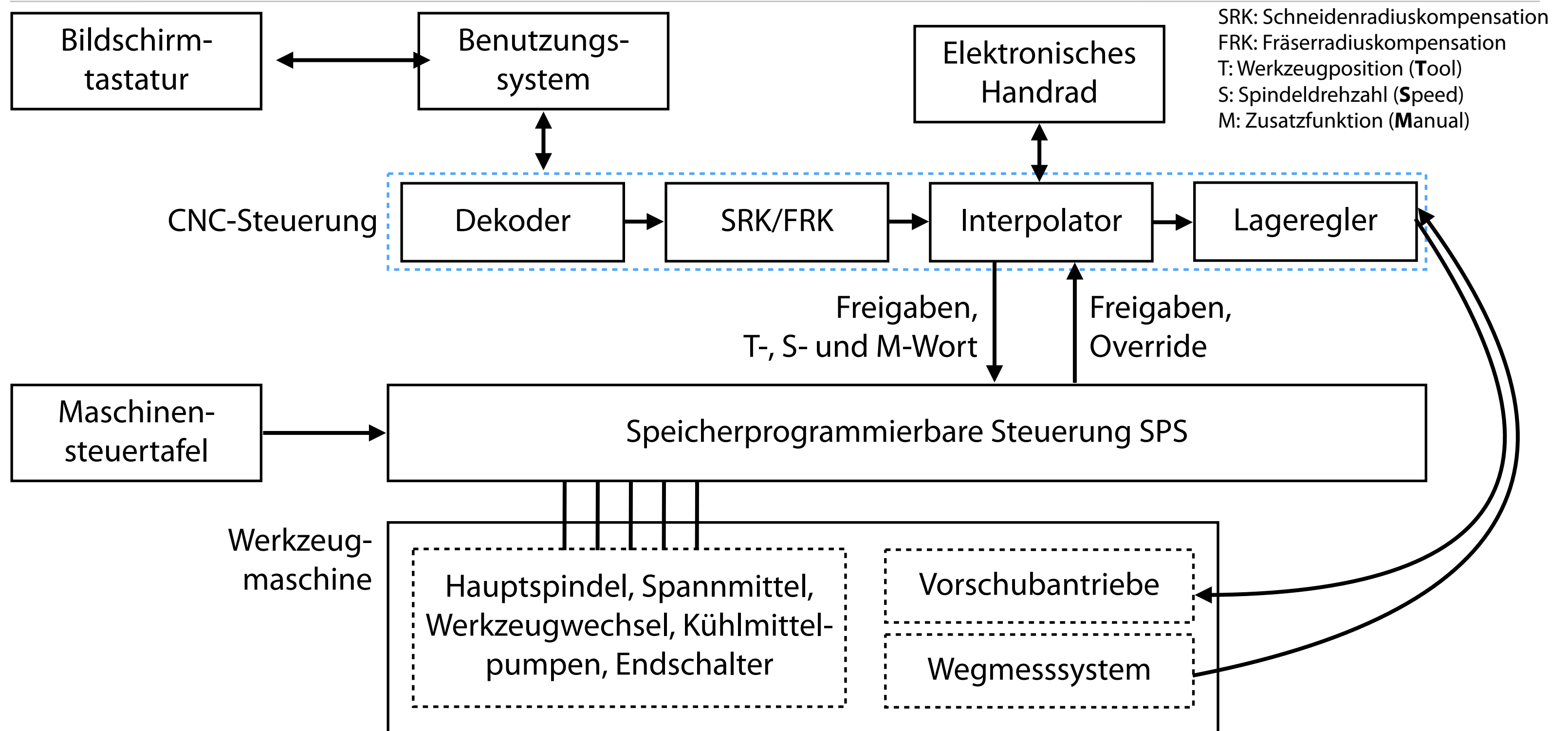
Kommunikationssysteme

Echtzeitprogrammiersysteme

Speicherprogrammierbare Steuerungen SPS

**Computerized Numerical Control CNC**

# Steuerung einer CNC-Werkzeugmaschine



CNC-Werkzeugmaschinen ermöglichen durch den Einsatz moderner Steuerungstechnik die automatische Herstellung von Werkstücken komplexer Formen mit einer hohen Präzision.

# Koordinatensystem einer CNC-Werkzeugmaschine

---

## Bewegungsachsen

- Drei Bewegungsachsen bilden ein rechtshändiges, rechtwinkliges Koordinatensystem
- Normung nach DIN 66217
  - Zuordnung eines Koordinatensystems zu Bewegungsachsen einer numerisch gesteuerten Arbeitsmaschine
  - Vereinheitlichung der Programmierung von Werkzeugmaschinen

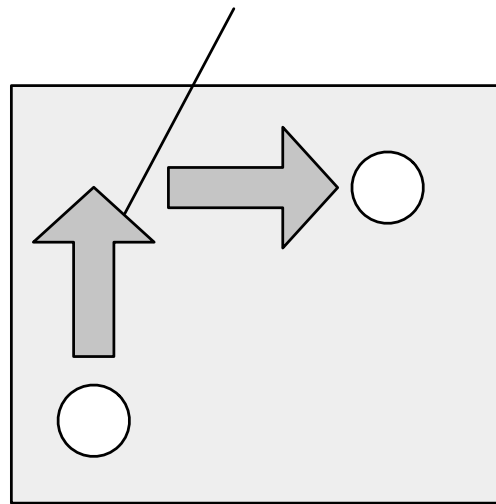
## Referenzpunkt

- Festlegung eines Bezugspunktes mit bekannten, vorher ausgemessenen Koordinaten
- Zweck: Initialisierung der Achspositionen

## Maschinennullpunkt

- Maschinenkoordinatensystem: Bezugskordinatensystem für numerisch-gesteuerte Werkzeugmaschine
- Maschinennullpunkt: Ursprung des Maschinenkoordinatensystems

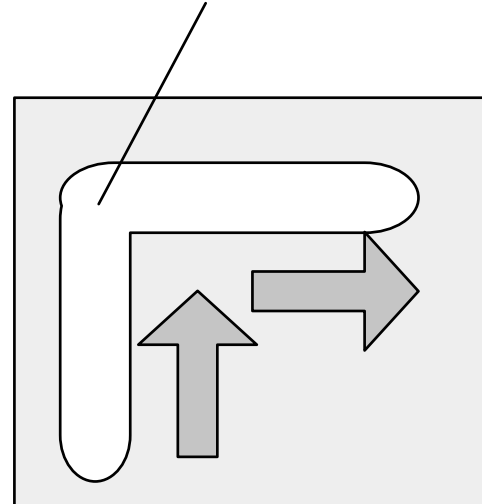
Werkzeug nicht im Eingriff



## Punktsteuerung

- Für punktuelle Bearbeitung des Werkstückes (z. B. Punktschweißen)
- Kein funktionsmäßiger Zusammenhang zwischen den Bewegungen in Richtung verschiedener Koordinatenachsen
- Werkzeug bei Bewegung nicht im Eingriff

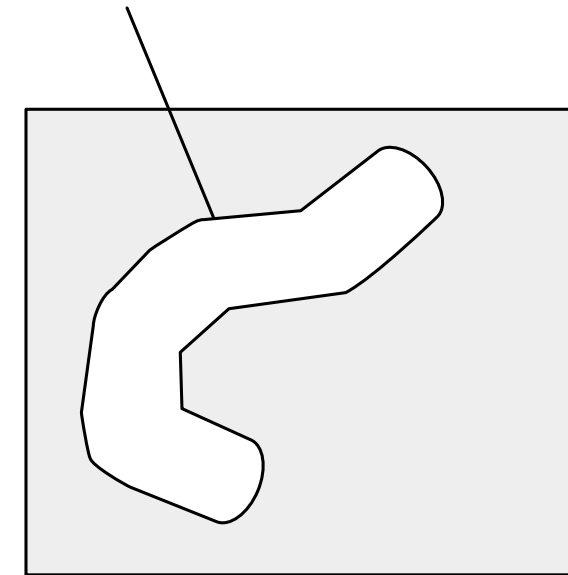
Werkzeug im Eingriff



## Streckensteuerung

- Für Dreh- und Fräsmaschinen
- Vorschubbewegung immer nur entlang einer Achse
- z. B. Nutenfräsen, Drehen zylindrischer Wellen

Werkzeug im Eingriff



## Bahnsteuerung

- Bearbeitung entlang beliebiger Kurven
- Bewegung in mehreren Koordinaten gleichzeitig
- Funktionsmäßiger Zusammenhang der Achsen
- Geraden- und Kreisinterpolation



# Interpolation

---

## Geradeninterpolation

- Kürzeste Verbindung zweier Punkte
- Räumliche Durchführung möglich

## Kreisinterpolation

- Durchführung meist nur in einer Ebene möglich
- Angabe von Eckpunktkoordinaten, Mittelpunktkoordinaten und Drehrichtung der Kreisbewegung

**Interpolation bezeichnet die Berechnung einzelner Stützpunkte eines Kurvenzugs. Sie dient der Reduzierung der zur Definition der Raumkurve (bei der Bahnsteuerung) notwendigen Datenmenge.**

# NC-Programme

---

Besteht aus Folge von Anweisungen (Sätzen), die eine NC-Maschine veranlassen eine bestimmte Bearbeitungsaufgabe durchzuführen

## Notwendige Informationen

- Erforderliche Weginformationen
- Zusätzliche Schaltinformationen
- Hilfsbefehle

## Sätze beinhalten

- Vollständige Arbeitsanweisung
- Programmmanfangszeichen
- Spezielle Funktion für das Programmende



# Aufbau von NC-Sätzen

---

## Wegbedingung G

- Anweisungen, wie die nachfolgenden Daten zu verarbeiten sind (z .B. absolute vs. inkrementelle Bemaßung, Werkzeugkorrekturaufruf, Spiegelung der Achsen)

## Werkzeugauswahl T

- Aufruf des für die Bearbeitungsart erforderlichen Werkzeug am Revolver
- Gleichzeitige Berücksichtigung aller – im Werkzeugdatenspeicher abgelegten – zugehörigen Werkzeugdaten

## Koordinatenwerte X,Y,Z

- Eingaben erfolgen mit oder ohne Vorzeichen in Vielfachen der kleinsten Einheit

## Vorschubgeschwindigkeit F

- Angabe in mm/min

## Drehzahl S

- Zweistellige Angabe nach VDI 3252 oder analoge Drehzahlausgabe bei V-konstant

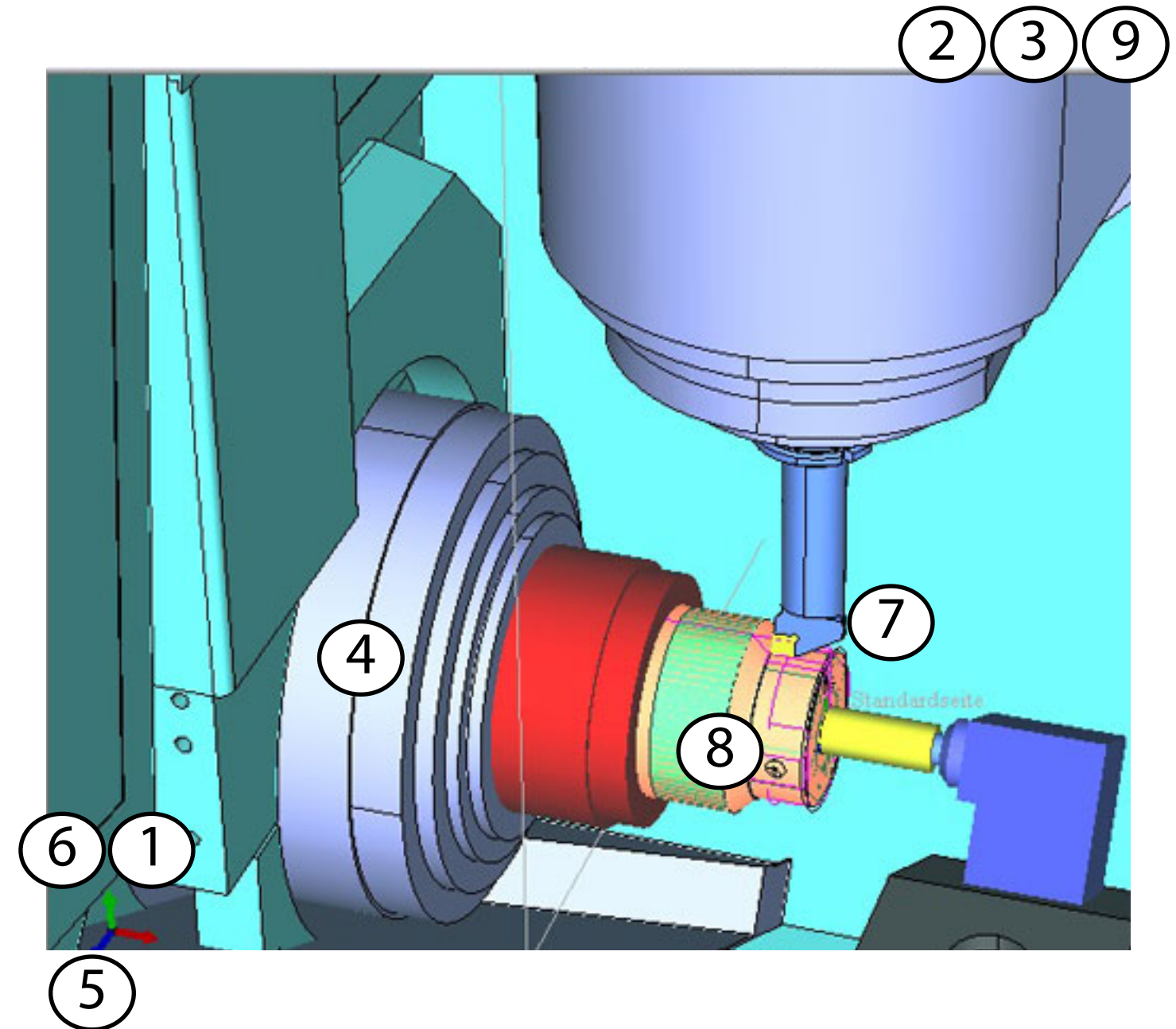
## Hilfsfunktion

- Koordinierung von Schaltfunktionen außerhalb der NC-Steuerung (z. B. Drehrichtung, Werkzeugwechseln, Kühlmittelzufuhr)

**Jeder Satz eines Maschinensteuerprogrammes beginnt mit einer (meist dreistelligen) Satznummer (N 001 bis N 999)**

# Beispiel NC-Programm

Schritt	01	Programm
1	N1 G59 Z60	Nullpunktverschiebung in Z programmiert
2	N2 G53 X150 Z200	Werkzeugwechsel anfahren (Eilgang) bezogen auf Maschinennullpunkt
3	N3 T0101	Werkzeugaufruf Werkzeug/WZ-Korrekturspeicher
4	N4 G96 V180 M03	Hauptspindel mit C=180 m/min (konst. Schnittgeschwindigkeit); Spindeldrehung rechts
5	N5 G0 X80	X zustellen mit Eilgang
6	N6 Z101	Z zustellen mit Eilgang (modaler Befehl G0)
7	N7 G1 Z50 F0.2 M8	Längsdrehen mit Vorschub (F= 0,2mm/U); Kühlmittelpumpe ein
8	N8 X100	Querdrehen mit Vorschub
9	N9 G53 X150 Z200 M9	Werkzeugwechsellpunkt anfahren (Eilgang); Kühlmittelpumpe aus
-	N10 M30	Programmende



# Arten der CNC-Programmerstellung

---

## Externe Programmerstellung

- Maschinelle, rechnergestützte Programmierung an separatem Rechner mit entsprechender Software
- Programm prüft auf Korrektheit, führt Berechnungen geometrischer und technologischer Art durch

## Programmbibliothek beinhaltet u. a.

- Arithmetische Funktionen zur Berechnung komplizierter Strukturen
- Schnittstelle zu CAD-System
- Technologiedatenbank
- Dynamische Werkstückänderung

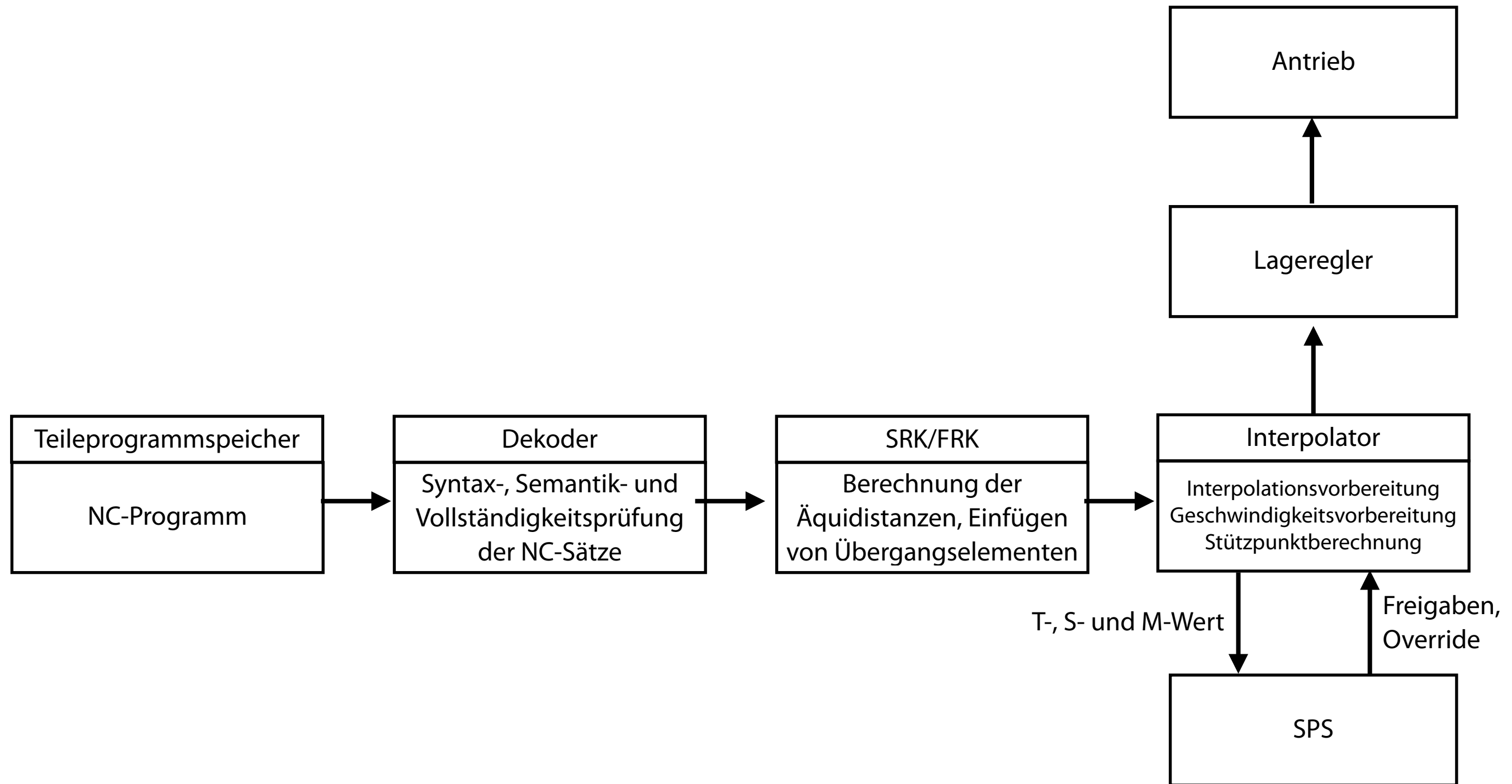
## Handeingabe

- Direkte Eingabe der Steuerbefehle über Bedientastatur an Maschine
- Hohe Fehlergefahr

## Sonderform: Werkstattprogrammierung

- In die CNC-Steuerung integriertes Programmiersystem
- Bedienung im Dialogbetrieb
- Für einfache Werkstücke geeignet

# Informationsverarbeitung in einer CNC-Steuerung



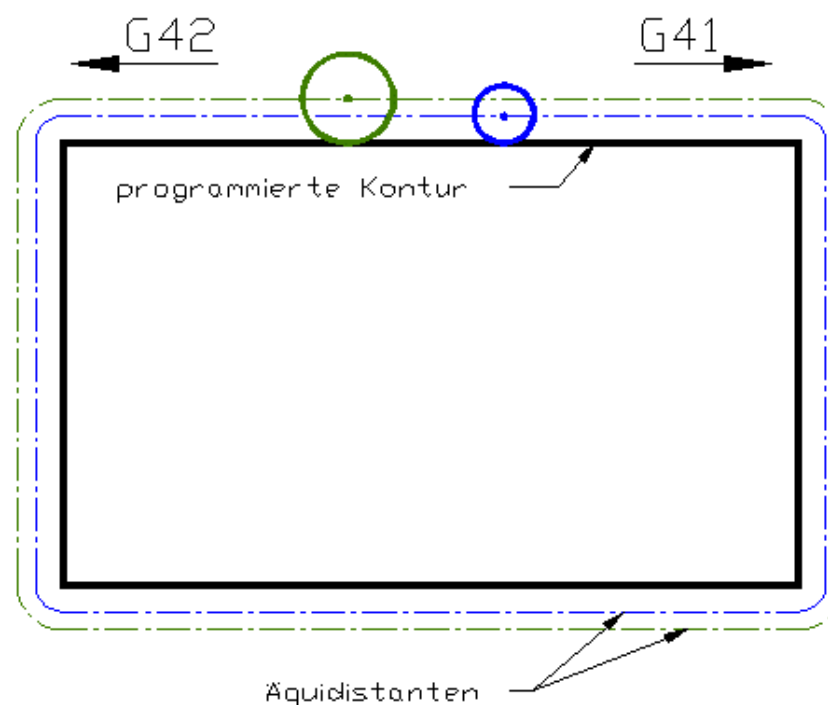
# Korrekturmechanismen

## Nullpunktverschiebung

- Ziel: vom Maschinenkoordinatensysteme unabhängige Programmerstellung
- Programmerstellung soll sich auf den Werkstücknullpunkt beziehen
- Achsparallele Verschiebung zwischen dem Maschinen- und dem Werkstückkoordinatensystem am Anfang des NC-Programms

## Werkzeugkorrekturen

- Werkzeuglängenkorrektur: Ausgleich von Verschleiß, Voreinstellungstoleranzen durch Maßabweichung in Richtung der Werkzeugachsen
- Werkzeugradiuskorrektur: Berücksichtigung des Werkzeugradiuswertes (Werkzeugmittelpunkt-bahn):



## Spindelsteigungsfehlerkompensation

- Notwendig, wenn Werkzeugmaschine über ein indirektes Wegmeßsystem (rotatorisches Meßsystem) verfügt
- Vorgehen
  - 1) Vermessung der Spindel
  - 2) Kompensation des mechanischen Fehlers
  - 3) Fehler wird im Interpolator korrigiert

**Die Korrekturen dienen der Umrechnung der vorgegebenen Koordinatenwerte in die Koordinaten für die korrekte Ansteuerung der NC-Achsen.**

# Literatur

---

Becker, U., Grundlagen der Automatisierungstechnik I: Einführung in die Automatisierungstechnik; Fachzentrum für Automatisierungstechnik im BTZ Rohr-Kloster 2007

HMS Industrial Networks GmbH: Markttrends: Industrielle Kommunikation 2019: Marktanteile industrieller Netzwerke

Lauber, R.; Göhner, P.: Prozessautomatisierung I, Springer-Verlag Berlin-Heidelberg, 1999.

Tröster, Fritz. Steuerungs-und Regelungstechnik für Ingenieure. Oldenbourg Verlag, 2011.

Uhlmann, E.: Skript: Übungen im Versuchsfeld – Grundlagen numerischer Steuerung, Sommersemester 2008.

Weyrich, M.: 2015 Skript Automatisierungstechnik I,, Universität Stuttgart, Institut für Automatisierungs- und Softwaretechnik, Sommersemester 2015.