

# Echtzeitsysteme

## Einleitung

### **Peter Ulbrich**

Lehrstuhl für Verteilte Systeme und Betriebssysteme  
Friedrich-Alexander-Universität Erlangen-Nürnberg  
[https://www4.cs.fau.de/Lehre/WS19/V\\_EZS/](https://www4.cs.fau.de/Lehre/WS19/V_EZS/)

14. Oktober 2019



# Das erste Echtzeitrechensystem

## ■ Whirlwind I

- **Zweck:** Flugsimulator  
(Ausbildung von Bomberbesatzungen)
- **Auftraggeber:** U.S. Navy
- **Auftragnehmer:** MIT
- **Laufzeit:** 1945 – 1952



(Quelle: Alex Handy from Oakland, Nmibia)

## ■ Technische Daten

- Digitalrechner, bit-parallele Operationen
- 5000 Röhren, 11000 Halbleiterdioden
- magnetischer Kernspeicher
- Röhrenmonitore mit Lichtgriffel



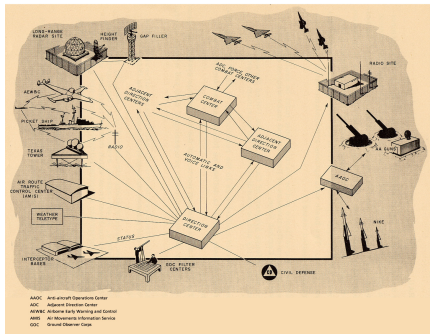
Spätere Nutzung in **SAGE** durch die U.S. Air Force



# SAGE – Semi-AutomatGround Environment

Erstes verteiltes Echtzeitrechnungssystem als Schöpfung des „Kalten Krieges“

## ■ Automatisiertes Kontroll- und Abwehrsystem gegen Bomber



- 27 Installationen
  - verteilt über die USA
  - *Nonstop*-Betrieb
  - 25 Jahre
- Kopplung durch Datenfernleitungen
  - Telefonleitungen
  - Internet-„Mutter“
- pro Installation...
  - 100 Konsolen
  - 500 KLOC Ass.



Entwicklung eines leistungsfähigeren Nachfolgers: Whirlwind II



- Der Nachfolger AN/FSQ-7 alias „Whirlwind II“:



(Quelle: Steve Jurvetson from Menlo Park, USA)

← SAGE Bedienstation

- Technische Daten

- **Auftraggeber:** U.S. Air Force
- **Auftragnehmer:** MIT, später IBM
- **Bauweise:** 55000 Röhren, 2000  $m^2$ , 275 t, 3 MW, 75 KIPS

- Betriebsdaten von SAGE:

- **Installation:** 22 - 23 Stationen im Zeitraum 1959 - 1963
- **Betrieb:** bis 1983 (Whirlwind I bis 1979)
- **Kosten:** 8–12 Milliarden \$ (1964)  $\rightsquigarrow$  ca. 97 Milliarden \$ (2019)
- **Nachfolger:** u.a. AWACS



# Moderne Echtzeitsysteme

Wo immer Rechensysteme mit ihrer physikalischen Umwelt interagieren ...



### CAN CLASS B

- 1 SAM5RB Fahrer
- 2 SAM5RB Beifahrer
- 3 SAM5RB Heck 1
- 4 SAM5RB Heck 2
- 5 Sitzsteuergerät Fahrer
- 6 Sitzsteuergerät Beifahrer
- 7 Sitzsteuergerät hinten links
- 8 Sitzsteuergerät hinten rechts
- 9 Türsteuergerät vorne Fahrerseite
- 10 Türsteuergerät vorne Beifahrerseite
- 11 Türsteuergerät hinten Fahrerseite
- 12 Türsteuergerät hinten Beifahrerseite
- 13 Steuergerät Tenorwand
- 14 Dachbodenantenne
- 15 Nachspeicher-Mitte (DKM)
- 16 Vorderer Boden-Feld (YBF)
- 17 Hinterer Boden-Feld (HBF)
- 18 Elektronisches Zündschloss (EZS)
- 19 Handbremse
- 20 Motorbremse
- 21 Motorbremse
- 22 Motorbremse
- 23 Motorbremse
- 24 Motorbremse
- 25 Motorbremse
- 26 Motorbremse
- 27 Motorbremse
- 28 Motorbremse
- 29 Motorbremse
- 30 Motorbremse
- 31 Motorbremse
- 32 Motorbremse
- 33 Motorbremse
- 34 Motorbremse
- 35 Motorbremse
- 36 Motorbremse
- 37 Motorbremse
- 38 Motorbremse
- 39 Motorbremse
- 40 Motorbremse
- 41 Motorbremse
- 42 Motorbremse
- 43 Motorbremse
- 44 Motorbremse
- 45 Motorbremse
- 46 Motorbremse
- 47 Motorbremse
- 48 Motorbremse
- 49 Motorbremse
- 50 Motorbremse
- 51 Motorbremse
- 52 Motorbremse
- 53 Motorbremse
- 54 Motorbremse
- 55 Motorbremse
- 56 Motorbremse
- 57 Motorbremse
- 58 Motorbremse
- 59 Motorbremse
- 60 Motorbremse
- 61 Motorbremse
- 62 Motorbremse
- 63 Motorbremse
- 64 Motorbremse
- 65 Motorbremse
- 66 Motorbremse
- 67 Motorbremse
- 68 Motorbremse
- 69 Motorbremse
- 70 Motorbremse
- 71 Motorbremse
- 72 Motorbremse
- 73 Motorbremse
- 74 Motorbremse
- 75 Motorbremse
- 76 Motorbremse
- 77 Motorbremse
- 78 Motorbremse
- 79 Motorbremse
- 80 Motorbremse
- 81 Motorbremse
- 82 Motorbremse
- 83 Motorbremse
- 84 Motorbremse
- 85 Motorbremse
- 86 Motorbremse
- 87 Motorbremse
- 88 Motorbremse
- 89 Motorbremse
- 90 Motorbremse
- 91 Motorbremse
- 92 Motorbremse
- 93 Motorbremse
- 94 Motorbremse
- 95 Motorbremse
- 96 Motorbremse
- 97 Motorbremse
- 98 Motorbremse
- 99 Motorbremse
- 100 Motorbremse

- 1 Parktronic (PTD)
- 2 Fahrerdruckkontrolle (FKC)
- 3 Pneumatische Steuerung (PSE)
- 4 Heckdielekterschließung/-öffnung
- 5 Zentrales Gateway
- 6 Elektronisches Wählhebelmodul
- 7 Airbag-EG (Armada)
- 8 Multifunktionssteuergerät (MSG)
- 9 Bonnetz-Steuergerät
- 10 Wandler Lenkschaltung
- 11 Motorbremse
- 12 Türschaltung hinten Fahrerseite
- 13 Türschaltung hinten Beifahrerseite

### CAN CLASS C

- 1 Elektronisches Zündschloss (EZS)
- 2 Komminstrument
- 3 Motorbremse
- 4 Zentrales Gateway
- 5 Elektronisches Wählhebelmodul
- 6 Luftleitung (SLF)
- 7 Thermo (DTR)
- 8 Lichtverlegeteilung
- 9 Motorbremse (ME)
- 10 Sensorisches Brake System (FSG)
- 11 Elektronische Getriebe-Steuerung

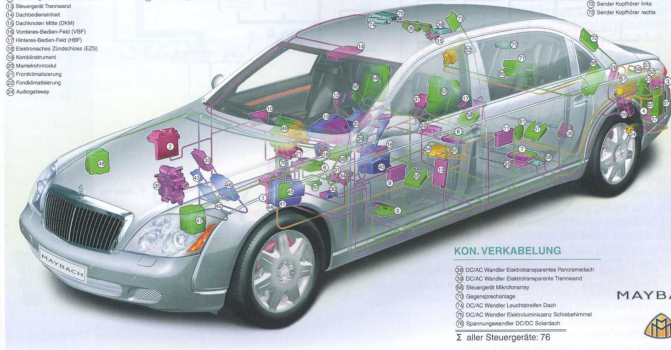
### MOST-BUS

- 1 Autogateway
- 2 Headunit
- 3 Steuergerät Sprachbedienung
- 4 TV-Tuner MOST
- 5 Soundverstärker
- 6 Navigationsrechner
- 7 Kommunikationssystem (CPI)

### PRIVATE-BUS

- 1 Sitzsteuergerät Fahrer
- 2 Sitzsteuergerät Beifahrer
- 3 Sitzsteuergerät hinten links
- 4 Sitzsteuergerät hinten rechts
- 5 TV-Tuner CAN
- 6 Dashboardinstrument
- 7 Sensorisches Brake System (FSG)
- 8 Sensorisches Brake System (ASG 1)
- 9 Sensorisches Brake System (ASG 2)
- 10 Multifunktions vorne links
- 11 Multifunktions vorne rechts
- 12 Multifunktions hinten links

- 13 Multifunktions hinten rechts
- 14 Keyless Go Innenraummodul
- 15 Keyless Go Tür hinten links
- 16 Keyless Go Tür hinten rechts
- 17 Drehinstrument
- 18 Farbdisplay hinten rechts
- 19 Kommunikationsplattform Ford (CPE)
- 20 Surround Amplifier
- 21 Audio Video Controller
- 22 CD-Wechsler
- 23 DVD-Spieler
- 24 Sender Kopfhörer links
- 25 Sender Kopfhörer rechts



### KON. VERKABELUNG

- 1 DC/AC Wandler Elektrontransparentes Perennschach
- 2 DC/AC Wandler Elektrontransparente Tenorwand
- 3 Steuergerät Mikrofon
- 4 Eingangsnetzbrücke
- 5 DC/AC Wandler Leuchtdioden-Dach
- 6 DC/AC Wandler Elektrontransparente Schiebhebel
- 7 Spannungswandler DC/DC Solarfach
- 8 aller Steuergeräte: 76

MAYBACH



(Quelle: DaimlerChrysler [1])



- 1 Historischer Bezug
  - Das erste Echtzeitrechensystem
  - SAGE – Der Nachfolger
  - Heutige Echtzeitsysteme
- 2 Echtzeitbetrieb
  - Definition
  - Realzeitbetrieb
  - Termine
  - Deterministische Ausführung
- 3 Aufbau und Abgrenzung
  - Struktur dieser Vorlesung
  - Abgrenzung
- 4 Zusammenfassung



*Echtzeitbetrieb ist ein Betrieb eines Rechensystems, bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind derart, dass die **Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind.***



*Echtzeitbetrieb ist ein Betrieb eines Rechensystems, bei dem Programme zur Verarbeitung anfallender Daten ständig betriebsbereit sind derart, dass die **Verarbeitungsergebnisse innerhalb einer vorgegebenen Zeitspanne verfügbar sind.***

*Die Daten können je nach Anwendungsfall nach einer zeitlich **zufälligen Verteilung** oder zu **vorbestimmten Zeitpunkten** anfallen.*





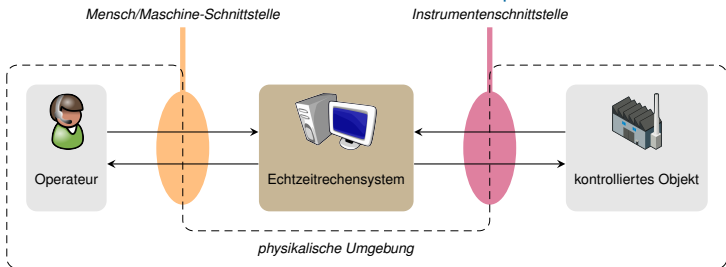
Operateur



Echtzeitrechensystem



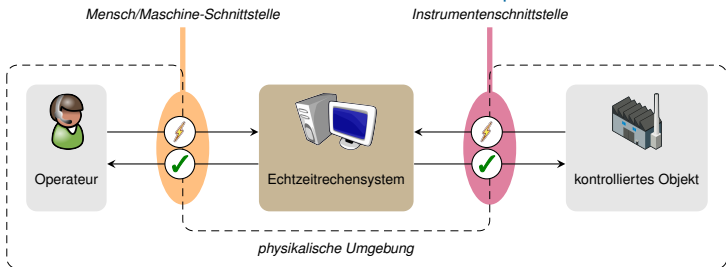
kontrolliertes Objekt



- Echtzeitrechnungssystem interagiert mit der **physischen Umwelt**

# Kopplung mit der (realen) Umwelt

## Komponenten eines Echtzeitsystems

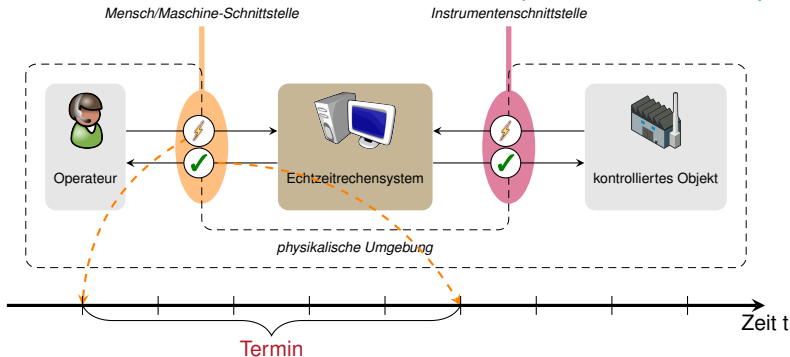


- Echtzeitrechnungssystem interagiert mit der **physikalischen Umwelt**
- Berechnet als Reaktion auf **Ereignisse** ⚡ (engl. *event*, Stimuli) der Umgebung **Ergebnisse** ✓ (engl. *result*)



# Kopplung mit der (realen) Umwelt

## Komponenten eines Echtzeitsystems

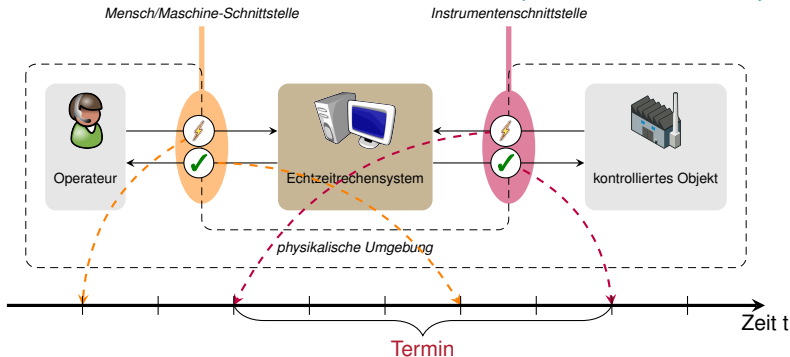


- Echtzeitrechnungssystem interagiert mit der **physikalischen Umwelt**
- Berechnet als Reaktion auf **Ereignisse** ⚡ (engl. *event*, Stimuli) der Umgebung **Ergebnisse** ✓ (engl. *result*)
- Zeitpunkt, zu dem ein Ergebnis vorliegen muss, wird als **Termin** oder **Frist** (engl. *deadline*) bezeichnet



# Kopplung mit der (realen) Umwelt

## Komponenten eines Echtzeitsystems



- Echtzeitrechnungssystem interagiert mit der **physikalischen Umwelt**
- Berechnet als Reaktion auf **Ereignisse** ⚡ (engl. *event*, Stimuli) der Umgebung **Ergebnisse** ✓ (engl. *result*)
- Zeitpunkt, zu dem ein Ergebnis vorliegen muss, wird als **Termin** oder **Frist** (engl. *deadline*) bezeichnet





Echtzeitbetrieb bedeutet **Rechtzeitigkeit**

- Funktionale Korrektheit reicht für korrektes Systemverhalten nicht aus
- **Rechtzeitige** Bereitstellung der Ergebnisse ist **entscheidend**





Echtzeitbetrieb bedeutet **Rechtzeitigkeit**

- Funktionale Korrektheit reicht für korrektes Systemverhalten nicht aus
- **Rechtzeitige** Bereitstellung der Ergebnisse ist **entscheidend**
  
- Den Rahmen stecken der **Eintrittspunkt** des Ereignisses und der entsprechende **Termin** ab





Echtzeitbetrieb bedeutet **Rechtzeitigkeit**

- Funktionale Korrektheit reicht für korrektes Systemverhalten nicht aus
- **Rechtzeitige** Bereitstellung der Ergebnisse ist **entscheidend**

- Den Rahmen stecken der **Eintrittspunkt** des Ereignisses und der entsprechende **Termin** ab

 Termine hängen dabei von der Anwendung ab

**wenige Mikrosekunden** z.B. Drehzahl- und Stromregelung bei der Ansteuerung von Elektromotoren

**einige Millisekunden** z.B. Multimedia-Anwendungen (Übertragung von Ton- und Video)

**Sekunden, Minuten, Stunden** z.B. Prozessanlagen (Erhitzen von Wasser)





**Geschwindigkeit ist keine Garantie** für die rechtzeitige Bereitstellung von Ergebnissen

- **Asynchrone Programmunterbrechungen** (engl. *interrupts*) können **unvorhersagbare Laufzeitvarianzen** verursachen
- **Schnelle Programmausführung** ist bestenfalls hinreichend für die rechtzeitige Bearbeitung einer Aufgabe





**Geschwindigkeit ist keine Garantie** für die rechtzeitige Bereitstellung von Ergebnissen

- **Asynchrone Programmunterbrechungen** (engl. *interrupts*) können **unvorhersagbare Laufzeitvarianzen** verursachen
- Schnelle Programmausführung ist bestenfalls hinreichend für die rechtzeitige Bearbeitung einer Aufgabe



**Zeit ist keine intrinsische Eigenschaft des Rechensystems**

- Die Zeitskala des Rechensystems muss nicht mit der durch die Umgebung vorgegebenen (Realzeit) übereinstimmen  $\leadsto$  Zeitgeber?
- Temporale Eigenschaften des kontrollierten (physikalischen) Objekts müssen im Rechenystem geeignet abgebildet werden





# Konsequenzen überschrittener Termine

Verbindlichkeit von Terminvorgaben





- **Weich** (engl. *soft*) auch „schwach“
  - **Ergebnis verliert** mit zunehmender Terminüberschreitung **an Wert** (z.B. Bildrate bei Multimediasystemen)
- Terminverletzung ist tolerierbar





- **Weich** (engl. *soft*) auch „schwach“
  - **Ergebnis verliert** mit zunehmender Terminüberschreitung **an Wert** (z.B. Bildrate bei Multimediasystemen)  
→ Terminverletzung ist tolerierbar
- **Fest** (engl. *firm*) auch „stark“
  - **Ergebnis wird** durch eine Terminüberschreitung **wertlos** und wird verworfen (z.B. Abgabetermin einer Übungsaufgabe)  
→ Terminverletzung ist tolerierbar, führt zum Arbeitsabbruch





# Konsequenzen überschrittener Termine

Verbindlichkeit von Terminvorgaben

- **Weich** (engl. *soft*) auch „schwach“
  - **Ergebnis verliert** mit zunehmender Terminüberschreitung **an Wert** (z.B. Bildrate bei Multimediasystemen)
  - Terminverletzung ist tolerierbar
- **Fest** (engl. *firm*) auch „stark“
  - **Ergebnis wird** durch eine Terminüberschreitung **wertlos** und wird verworfen (z.B. Abgabetermin einer Übungsaufgabe)
  - Terminverletzung ist tolerierbar, führt zum Arbeitsabbruch
- **Hart** (engl. *hard*) auch „strikt“
  - **Terminüberschreitung** kann zum **Systemversagen** führen und eine „Katastrophe“ hervorrufen (z.B. Airbag)
  - Terminverletzung ist keinesfalls tolerierbar





- **Fest/Hart**  $\mapsto$  Terminverletzung ist nicht ausgeschlossen<sup>1</sup>
  - Terminverletzung wird vom Betriebssystem erkannt
  - $\rightarrow$  Weiteres Vorgehen hängt von der Art des Termins ab

---

<sup>1</sup> Auch wenn Ablaufplan und Betriebssystem auf dem Blatt Papier Determinismus zeigen, kann das im Feld eingesetzte technische System von unbekanntem/unvermeidbarem Störeinflüssen betroffen sein!





- **Fest/Hart**  $\mapsto$  Terminverletzung ist nicht ausgeschlossen<sup>1</sup>
  - Terminverletzung wird vom Betriebssystem erkannt
  - $\rightarrow$  Weiteres Vorgehen hängt von der Art des Termins ab

**Fest**  $\rightsquigarrow$  plangemäß weiterarbeiten

- Betriebssystem bricht den Arbeitsauftrag ab
- Nächster Arbeitsauftrag wird (planmäßig) gestartet
- $\rightarrow$  Transparent für die Anwendung

---

<sup>1</sup> Auch wenn Ablaufplan und Betriebssystem auf dem Blatt Papier Determinismus zeigen, kann das im Feld eingesetzte technische System von unbekanntem/unvermeidbarem Störeinflüssen betroffen sein!





- **Fest/Hart**  $\mapsto$  Terminverletzung ist nicht ausgeschlossen<sup>1</sup>
  - Terminverletzung wird vom Betriebssystem erkannt
  - $\rightarrow$  Weiteres Vorgehen hängt von der Art des Termins ab

**Fest**  $\rightsquigarrow$  plangemäß weiterarbeiten

- Betriebssystem bricht den Arbeitsauftrag ab
- Nächster Arbeitsauftrag wird (planmäßig) gestartet
- $\rightarrow$  Transparent für die Anwendung

**hart**  $\rightsquigarrow$  sicheren Zustand finden

- Betriebssystem löst eine **Ausnahmesituation** aus
- Ausnahme ist **intransparent für die Anwendung**
- $\rightarrow$  **Anwendung** behandelt diese Ausnahme

<sup>1</sup> Auch wenn Ablaufplan und Betriebssystem auf dem Blatt Papier Determinismus zeigen, kann das im Feld eingesetzte technische System von unbekanntem/unvermeidbarem Störeinflüssen betroffen sein!



- **Hard real-time computer system**  
(dt. Hartes Echtzeitrechensystem)
  - Rechensystem mit mind. einem hartem Termin
  - Garantiert unter allen (spezifizierten) Last- und Fehlerbedingungen
  - Laufzeitverhalten ist ausnahmslos **deterministisch**
  - Typisch für **sicherheitskritische Echtzeitrechensysteme**
    - engl. *safety-critical real-time computer system*
    - Beispiel: Fluglageregelung, Airbag, ...

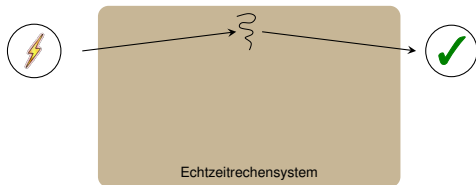


- **Hard real-time computer system**  
(dt. Hartes Echtzeitrechensystem)
  - Rechensystem mit mind. einem hartem Termin
  - Garantiert unter allen (spezifizierten) Last- und Fehlerbedingungen
  - Laufzeitverhalten ist ausnahmslos **deterministisch**
  - Typisch für **sicherheitskritische Echtzeitrechensysteme**
    - engl. *safety-critical real-time computer system*
    - Beispiel: Fluglageregelung, Airbag, ...
  
- **Soft real-time computer system**  
(dt. Weiches Echtzeitrechensystem)
  - Rechensystem welches keinen harten Termin erreichen muss
  - Termine können gelegentlich verpasst werden



# Herausforderung: Gewährleisten von Rechtzeitigkeit

Ereignisbehandlungen müssen termingerecht abgearbeitet werden

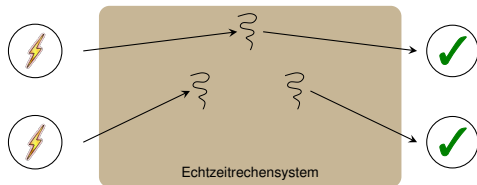


- Ereignisse aktivieren **Ereignisbehandlungen**
  - Wie viel Zeit benötigt die Ereignisbehandlung **maximal**?
  - Lösung des trivialen Falls ist (scheinbar) einfach, wenn man die **maximale Ausführungszeit** der Ereignisbehandlung kennt.



# Herausforderung: Gewährleisten von Rechtzeitigkeit

Ereignisbehandlungen müssen termingerecht abgearbeitet werden

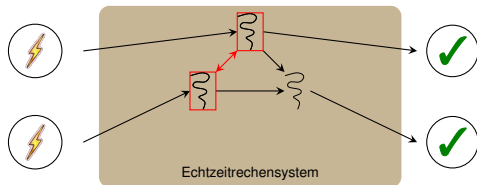


- Ereignisse aktivieren **Ereignisbehandlungen**
  - Wie viel Zeit benötigt die Ereignisbehandlung **maximal**?
  - Lösung des trivialen Falls ist (scheinbar) einfach, wenn man die **maximale Ausführungszeit** der Ereignisbehandlung kennt.
- Reale Echtzeitsysteme sind **komplex**
  - Mehrere Ereignisbehandlungen  $\leadsto$  Konkurrenz
    - Verwaltung gemeinsamer Betriebsmittel, allen voran die CPU.



# Herausforderung: Gewährleisten von Rechtzeitigkeit

Ereignisbehandlungen müssen termingerecht abgearbeitet werden



- Ereignisse aktivieren **Ereignisbehandlungen**
  - Wie viel Zeit benötigt die Ereignisbehandlung **maximal**?
  - Lösung des trivialen Falls ist (scheinbar) einfach, wenn man die **maximale Ausführungszeit** der Ereignisbehandlung kennt.
- Reale Echtzeitsysteme sind **komplex**
  - Mehrere Ereignisbehandlungen  $\leadsto$  Konkurrenz
    - Verwaltung gemeinsamer Betriebsmittel, allen voran die CPU.
  - Abhängigkeiten zwischen verschiedenen Ereignisbehandlungen



## Determiniertheit

*Bei identischen Eingaben sind verschiedene Abläufe zulässig, sie liefern jedoch stets das gleiche Resultat.*



## Determiniertheit

*Bei identischen Eingaben sind verschiedene Abläufe zulässig, sie liefern jedoch stets das gleiche Resultat.*



Im allgemeinen **unzureichend** für den Entwurf von Echtzeitsystemen



Transparenz von Programmunterbrechungen

- **Interrupts** verursachen vom normalen Ablauf abweichende **ausnahmebedingte Abläufe**



## Determiniertheit

*Bei identischen Eingaben sind verschiedene Abläufe zulässig, sie liefern jedoch stets das gleiche Resultat.*



Im allgemeinen **unzureichend** für den Entwurf von Echtzeitsystemen



Transparenz von Programmunterbrechungen

- **Interrupts** verursachen vom normalen Ablauf abweichende **ausnahmebedingte Abläufe**

## Determinismus

*Identische Eingaben führen zu identischen Abläufen. Zu jedem Zeitpunkt ist bestimmt, wie weitergefahren wird.*



## Determiniertheit

*Bei identischen Eingaben sind verschiedene Abläufe zulässig, sie liefern jedoch stets das gleiche Resultat.*



Im allgemeinen **unzureichend** für den Entwurf von Echtzeitsystemen



Transparenz von Programmunterbrechungen

- **Interrupts** verursachen vom normalen Ablauf abweichende **ausnahmebedingte Abläufe**

## Determinismus

*Identische Eingaben führen zu identischen Abläufen. Zu jedem Zeitpunkt ist bestimmt, wie weitergefahren wird.*



**Notwendig**, falls Termine einzuhalten sind

- Nur so lässt sich das Laufzeitverhalten verlässlich abschätzen



Echtzeitsysteme sind (schwach, stark oder strikt) deterministisch

## Vorhersagbarkeit

*Der Ablauf lässt sich zu jedem Zeitpunkt exakt angeben und hängt nicht von den aktuellen Eingaben oder vom aktuellen Zustand ab.*



## Vorhersagbarkeit

*Der Ablauf lässt sich zu jedem Zeitpunkt exakt angeben und hängt nicht von den aktuellen Eingaben oder vom aktuellen Zustand ab.*



### Vorteilhaft für zeitkritische Systeme

- Exakte Angaben zum zeitlichen Ablauf sind bereits à priori möglich
- Von Umgebung und Eingaben entkoppeltes Laufzeitverhalten
  - Aktivitäten folgen einem strikt vorgegebenem Stundenplan



Echtzeitsysteme sind (schwach, stark oder strikt) deterministisch

## Vorhersagbarkeit

*Der Ablauf lässt sich zu jedem Zeitpunkt exakt angeben und hängt nicht von den aktuellen Eingaben oder vom aktuellen Zustand ab.*



**Vorteilhaft** für zeitkritische Systeme

- Exakte Angaben zum zeitlichen Ablauf sind bereits à priori möglich
- Von Umgebung und Eingaben entkoppeltes Laufzeitverhalten
  - Aktivitäten folgen einem strikt vorgegebenem Stundenplan

Echtzeitsysteme müssen stets ein **deterministisches** oder besser **vorhersagbares** Laufzeitverhalten gewährleisten!

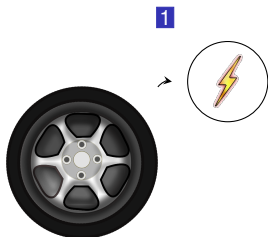
- Insbesondere beim **Zugriff auf gemeinsame Betriebsmittel**
  - CPU** → Umschaltung zwischen verschiedenen Aktivitäten
  - Kommunikationsmedium** → Versand von Nachrichten





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechensystems



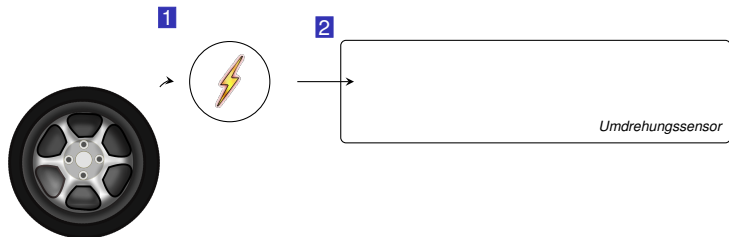
- ABS überwacht kontinuierlich Umdrehungszahl des Rads  
→ Messfühler erzeugt Signale (Ereignisse)





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechensystems



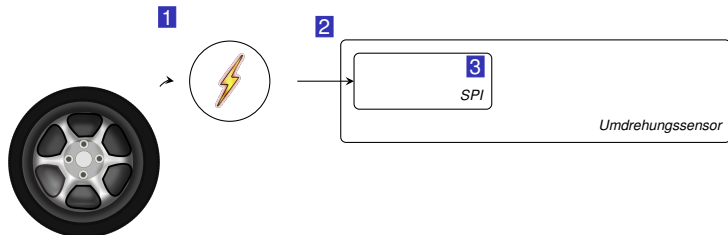
- ABS überwacht kontinuierlich Umdrehungszahl des Rads  
→ Messfühler erzeugt Signale (Ereignisse)
- **Intelligenter Sensor** (engl. *smart sensor*) führt Vorverarbeitung der Daten durch (erkennt z.B. Stillstand)





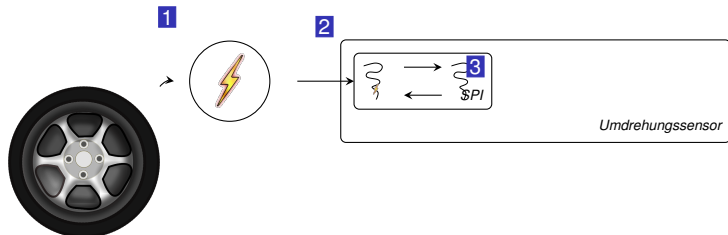
# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechensystems



- Meßfühlerdaten werden über den SPI-Bus entgegengenommen





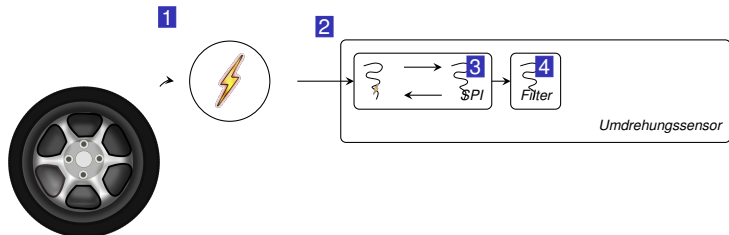
- Meßfühlerdaten werden über den SPI-Bus entgegengenommen
  - Buskommunikation erfordert eine ISR und einen Faden
    - Wann wird die ISR angesprungen? Sind Unterbrechungen gesperrt?
    - Wann wird der Faden eingeplant? Muss er auf Betriebsmittel warten?





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnungssystems



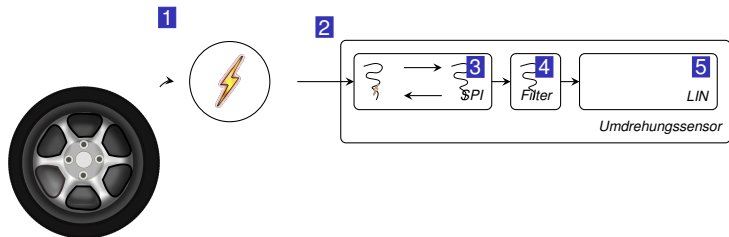
- Filter übernimmt die Signalvorverarbeitung
  - Angleichung diverser Abtastraten durch gesonderten Faden
    - der Filter verarbeitet immer mehrere Messwerte auf einmal
    - Wann wird der Faden eingeplant? Muss er auf Betriebsmittel warten?





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechensystems



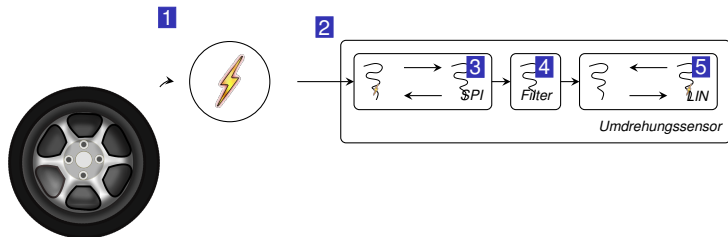
- Konsolidierte Messwerte werden an ABS-Steuergerät gesendet





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnensystems



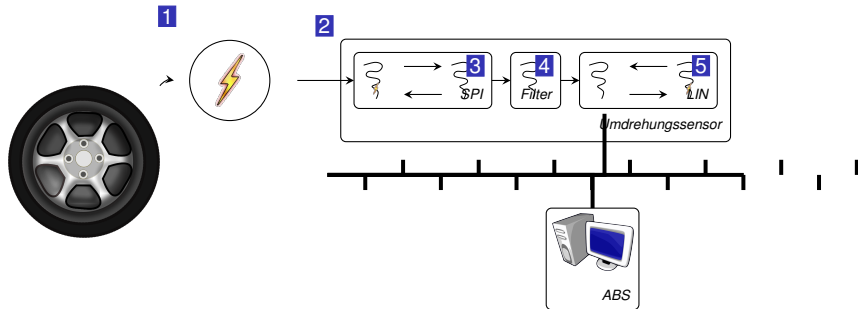
- Konsolidierte Messwerte werden an ABS-Steuergerät gesendet
  - Komplexer Gerätetreiber notwendig
    - Wann wird die ISR angesprochen? Sind Unterbrechungen gesperrt?
    - Wann wird der Faden eingeplant? Muss er auf Betriebsmittel warten?
    - Können alle Daten „auf einmal“ übertragen werden?





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnensystems



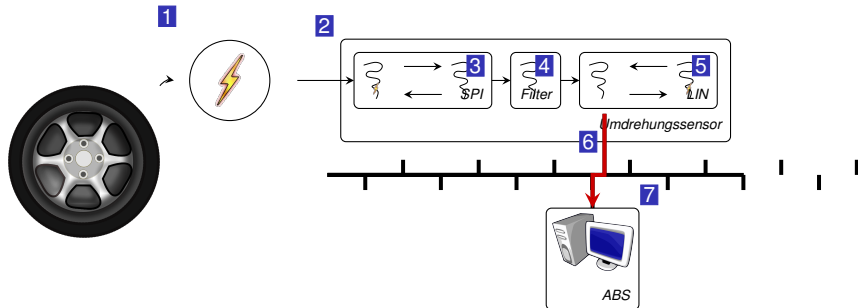
- Sensor und ABS-Steuergerät sind per LIN-Bus verbunden





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitsystems



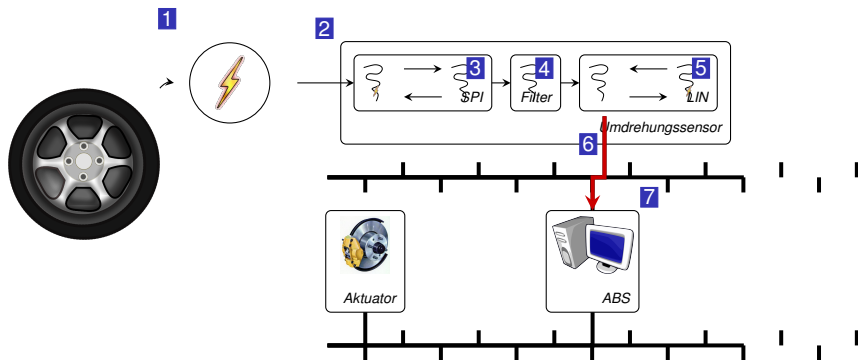
- Sensor und ABS-Steuergerät sind per LIN-Bus verbunden
  - Datenübertragung benötigt Zeit ...
    - Wie lange muss ich warten, bis ich auf das Medium zugreifen kann?
- ⚠ Vorgänge im ABS-Steuergerät sind noch deutlich komplexer





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnersystems



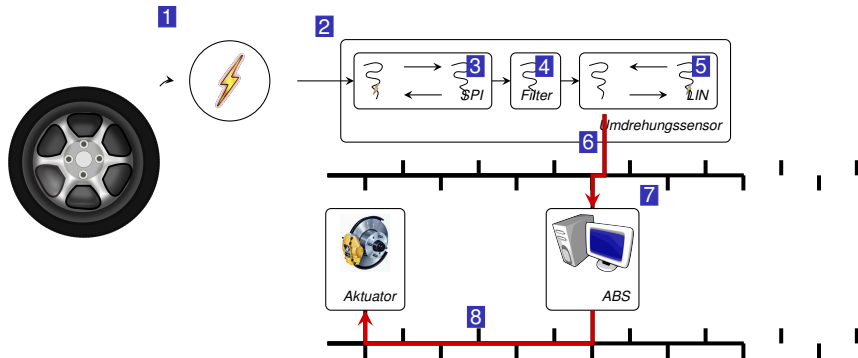
- Stellwert wird dem Aktor zugestellt
  - CAN-Bus verbindet ABS-Steuergerät und Aktor





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnersystems



## ■ Stellwert wird dem Aktor zugestellt

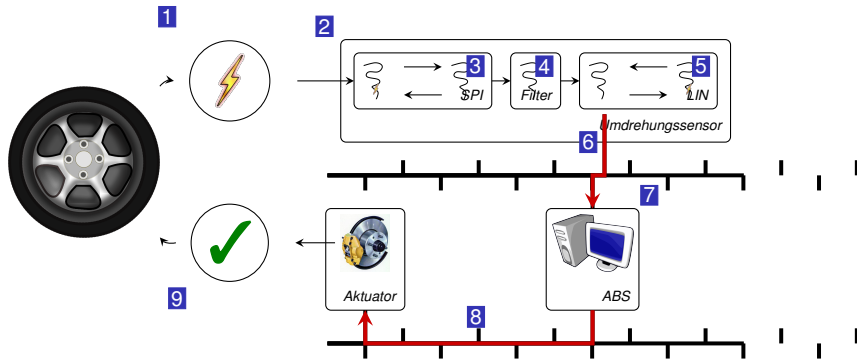
- CAN-Bus verbindet ABS-Steuergerät und Aktor
  - Wieviele Bytes schafft der Bus in einer bestimmten Zeit?
  - Wie lange muss ich warten, bis ich auf das Medium zugreifen kann?





# Beispiel: Ein (fiktives) Anti-Blockier-System

Funktion eines verteilten Echtzeitrechnersystems



- Stellwert wird dem Aktor zugestellt
  - CAN-Bus verbindet ABS-Steuergerät und Aktor
    - Wieviele Bytes schafft der Bus in einer bestimmten Zeit?
    - Wie lange muss ich warten, bis ich auf das Medium zugreifen kann?

☞ schließlich wird die Bremskraft geeignet beeinflusst





Wie lange dauert das ganze nun?



Die korrekte Funktion des ABS erfordert eine Reaktion auf eine Blockierung des Rades **innerhalb einer bestimmten Zeitspanne**





Wie lange dauert das ganze nun?



Die korrekte Funktion des ABS erfordert eine Reaktion auf eine Blockierung des Rades **innerhalb einer bestimmten Zeitspanne**

- Zu dieser Zeitspanne tragen zwei Komponenten bei:

**Aktive Zeitintervalle**  $\rightsquigarrow$  „Fortschritt“ im ABS

- Berechnungen benötigen Zeit  $\rightsquigarrow$  **maximale Ausführungszeit**
- Geschwindigkeit der Datenübertragung ist beschränkt





Wie lange dauert das ganze nun?



Die korrekte Funktion des ABS erfordert eine Reaktion auf eine Blockierung des Rades **innerhalb einer bestimmten Zeitspanne**

- Zu dieser Zeitspanne tragen zwei Komponenten bei:

**Aktive Zeitintervalle**  $\rightsquigarrow$  „Fortschritt“ im ABS

- Berechnungen benötigen Zeit  $\rightsquigarrow$  **maximale Ausführungszeit**
- Geschwindigkeit der Datenübertragung ist beschränkt

**Inaktive Zeitintervalle**  $\rightsquigarrow$  „Wartezeit“ für das ABS

- Fortschritt erfordert die Zuteilung von Betriebsmitteln
- z. B. CPU oder Kommunikationsmedium





Wie lange dauert das ganze nun?



Die korrekte Funktion des ABS erfordert eine Reaktion auf eine Blockierung des Rades **innerhalb einer bestimmten Zeitspanne**

- Zu dieser Zeitspanne tragen zwei Komponenten bei:

**Aktive Zeitintervalle**  $\leadsto$  „Fortschritt“ im ABS

- Berechnungen benötigen Zeit  $\leadsto$  **maximale Ausführungszeit**
- Geschwindigkeit der Datenübertragung ist beschränkt

**Inaktive Zeitintervalle**  $\leadsto$  „Wartezeit“ für das ABS

- Fortschritt erfordert die Zuteilung von Betriebsmitteln
- z. B. CPU oder Kommunikationsmedium



Die Frage ist, wie lange man auf die Zuteilung warten muss!

- **Determiniertheit** alleine reicht für die Beantwortung nicht aus!
- **Determinismus** erfordert die vollständige Kenntnis der Umgebung!
- **Vorhersagbarkeit** liefert die gewünschte Aussage zu dieser Frage!



 Deterministische Abarbeitung von Ereignisbehandlungen?



 Deterministische Abarbeitung von Ereignisbehandlungen?

- **Rein zyklisch**  $\rightsquigarrow$  periodische Ereignisbehandlungen, Abfrage-Betrieb
  - (Nahezu) konstanter Betriebsmittelbedarf von Periode zu Periode



 Deterministische Abarbeitung von Ereignisbehandlungen?

- **Rein zyklisch**  $\leadsto$  periodische Ereignisbehandlungen, Abfrage-Betrieb
  - (Nahezu) konstanter Betriebsmittelbedarf von Periode zu Periode
- **Meist zyklisch**  $\leadsto$  überwiegend periodische Ereignisbehandlungen
  - System muss auf externe Ereignisse reagieren können
  - Betriebsmittelbedarf schwankt bedingt von Periode zu Periode



## Deterministische Abarbeitung von Ereignisbehandlungen?

- **Rein zyklisch**  $\leadsto$  periodische Ereignisbehandlungen, Abfrage-Betrieb
  - (Nahezu) konstanter Betriebsmittelbedarf von Periode zu Periode
- **Meist zyklisch**  $\leadsto$  überwiegend periodische Ereignisbehandlungen
  - System muss auf externe Ereignisse reagieren können
  - Betriebsmittelbedarf schwankt bedingt von Periode zu Periode
- **Asynchron/vorhersagbar**  $\leadsto$  kaum periodische Ereignisbehandlungen
  - Aufeinanderfolgende Aktivierungen können zeitlich stark variieren
  - Zeitdifferenzen haben eine obere Grenze oder bekannte Statistik
  - Stark schwankender Betriebsmittelbedarf



## Deterministische Abarbeitung von Ereignisbehandlungen?

- **Rein zyklisch**  $\leadsto$  periodische Ereignisbehandlungen, Abfrage-Betrieb
  - (Nahezu) konstanter Betriebsmittelbedarf von Periode zu Periode
- **Meist zyklisch**  $\leadsto$  überwiegend periodische Ereignisbehandlungen
  - System muss auf externe Ereignisse reagieren können
  - Betriebsmittelbedarf schwankt bedingt von Periode zu Periode
- **Asynchron/vorhersagbar**  $\leadsto$  kaum periodische Ereignisbehandlungen
  - Aufeinanderfolgende Aktivierungen können zeitlich stark variieren
  - Zeitdifferenzen haben eine obere Grenze oder bekannte Statistik
  - Stark schwankender Betriebsmittelbedarf
- **Asynchron/nicht vorhersagbar**  $\leadsto$  aperiodische Ereignisbehandlungen
  - Ausschließlich externe Ereignisse
  - Hohe, nicht deterministische Laufzeitkomplexität einzelner Ereignisbehandlungen



- 1 Historischer Bezug
  - Das erste Echtzeitrechensystem
  - SAGE – Der Nachfolger
  - Heutige Echtzeitsysteme
- 2 Echtzeitbetrieb
  - Definition
  - Realzeitbetrieb
  - Termine
  - Deterministische Ausführung
- 3 Aufbau und Abgrenzung
  - Struktur dieser Vorlesung
  - Abgrenzung
- 4 Zusammenfassung



# Aufbau der Vorlesung

---

- Die Vorlesung orientiert sich vor allem . . .
  - an der Ausprägung des Spezialzweckbetriebs

Einleitung

Grundlagen



# Aufbau der Vorlesung

- Die Vorlesung orientiert sich vor allem ...
  - an der Ausprägung des Spezialzweckbetriebs

Einleitung

Grundlagen

vorranggesteuerte  
Systeme

taktgesteuerte  
Systeme

Analyse



# Aufbau der Vorlesung

- Die Vorlesung orientiert sich vor allem ...
  - an der Ausprägung des Spezialzweckbetriebs
  - und den Eigenschaften der Ereignisse und ihrer Behandlungen,

Einleitung

Grundlagen

vorranggesteuerte  
Systeme

taktgesteuerte  
Systeme

Analyse

periodische Echtzeitsysteme

nicht-periodische Echtzeitsysteme

Rangfolge

Zugriffskontrolle



# Aufbau der Vorlesung

- Die Vorlesung orientiert sich vor allem ...
  - an der Ausprägung des Spezialzweckbetriebs
  - und den Eigenschaften der Ereignisse und ihrer Behandlungen,
  - blickt aber auch über den Tellerrand.

Einleitung

Grundlagen

vorranggesteuerte  
Systeme

taktgesteuerte  
Systeme

Analyse

periodische Echtzeitsysteme

nicht-periodische Echtzeitsysteme

Rangfolge

Zugriffskontrolle

Aktuelle Forschungsthemen (Mehrkernrechner)

Aktuelle Forschungsthemen II / Industrievortrag (optional)



# Aufbau der Vorlesung

- Die Vorlesung orientiert sich vor allem ...
  - an der Ausprägung des Spezialzweckbetriebs
  - und den Eigenschaften der Ereignisse und ihrer Behandlungen,
  - blickt aber auch über den Tellerrand.

Einleitung

Grundlagen

vorranggesteuerte  
Systeme

taktgesteuerte  
Systeme

Analyse

periodische Echtzeitsysteme

nicht-periodische Echtzeitsysteme

Rangfolge

Zugriffskontrolle

Aktuelle Forschungsthemen (Mehrkernrechner)

Aktuelle Forschungsthemen II / Industrievortrag (optional)

Zusammenfassung und Ausblick

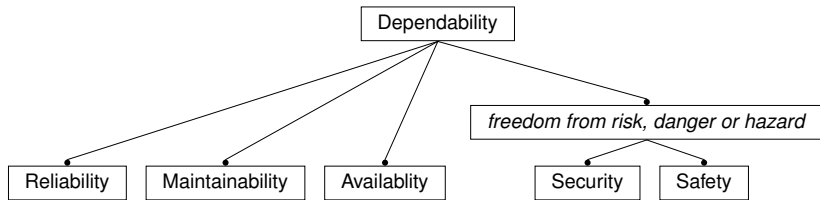


Echtzeitsysteme sind häufig **sicherheitskritische Systeme** und erfordern ein hohes Maß an **Verlässlichkeit**.



Echtzeitsysteme sind häufig **sicherheitskritische Systeme** und erfordern ein hohes Maß an **Verlässlichkeit**. Verlässlichkeit selbst hat viele Gesichter

...



*The trustworthiness of a computing system which allows reliance to be justifiably placed on the service it delivers. [3]*



Verlässlichkeit **erfordert** Rechtzeitigkeit!

- Verpasste Termine stellen Fehler dar
- Diese Fehler müssen ggf. erkannt oder maskiert werden





Verlässlichkeit **erfordert** Rechtzeitigkeit!

- Verpasste Termine stellen Fehler dar
- Diese Fehler müssen ggf. erkannt oder maskiert werden

■ **Andererseits:** Rechtzeitigkeit **erfordert** Verlässlichkeit!

- Fehler können zum Verpassen eines Termins führen
- Maskieren solcher Fehler hilft, die Rechtzeitigkeit zu gewährleisten





Verlässlichkeit **erfordert** Rechtzeitigkeit!

- Verpasste Termine stellen Fehler dar
- Diese Fehler müssen ggf. erkannt oder maskiert werden

■ **Andererseits:** Rechtzeitigkeit **erfordert** Verlässlichkeit!

- Fehler können zum Verpassen eines Termins führen
- Maskieren solcher Fehler hilft, die Rechtzeitigkeit zu gewährleisten

■ Betrachtung der Rechtzeitigkeit unter Annahme des *fehlerfreien Falls*

- Verletzte Termine werden auf einer höheren Ebene behandelt
- Toleranz gegenüber Fehlern dient der Verlässlichkeit





Verlässlichkeit **erfordert** Rechtzeitigkeit!

- Verpasste Termine stellen Fehler dar
- Diese Fehler müssen ggf. erkannt oder maskiert werden

■ **Andererseits:** Rechtzeitigkeit **erfordert** Verlässlichkeit!

- Fehler können zum Verpassen eines Termins führen
- Maskieren solcher Fehler hilft, die Rechtzeitigkeit zu gewährleisten

■ Betrachtung der Rechtzeitigkeit unter Annahme des *fehlerfreien Falls*

- Verletzte Termine werden auf einer höheren Ebene behandelt
- Toleranz gegenüber Fehlern dient der Verlässlichkeit



Das ist Thema der **Verlässlichen Echtzeitsystem** im SS



- 1 Historischer Bezug
  - Das erste Echtzeitrechensystem
  - SAGE – Der Nachfolger
  - Heutige Echtzeitsysteme
- 2 Echtzeitbetrieb
  - Definition
  - Realzeitbetrieb
  - Termine
  - Deterministische Ausführung
- 3 Aufbau und Abgrenzung
  - Struktur dieser Vorlesung
  - Abgrenzung
- 4 Zusammenfassung



- **Echtzeitbetrieb** eines Rechensystems in seiner Umgebung
  - Ereignis, Ereignisbehandlung, Ergebnis, Termin
- Komponenten eines Echtzeitsystems
  - Operateur, Echtzeitrechensystem, kontrolliertes Objekt
- **Weiche**, **feste** und **harte** Echtzeitbedingungen
- Determiniertheit, Determinismus, Vorhersagbarkeit
- Verhalten von Echtzeitanwendungen
  - Rein/meist zyklisch
  - Asynchron und irgendwie/nicht vorhersagbar



- **Echtzeitbetrieb** eines Rechensystems in seiner Umgebung
  - Ereignis, Ereignisbehandlung, Ergebnis, Termin
- Komponenten eines Echtzeitsystems
  - Operateur, Echtzeitrechensystem, kontrolliertes Objekt
- **Weiche**, **feste** und **harte** Echtzeitbedingungen
- Determiniertheit, Determinismus, Vorhersagbarkeit
- Verhalten von Echtzeitanwendungen
  - Rein/meist zyklisch
  - Asynchron und irgendwie/nicht vorhersagbar
- **Abgrenzung**: Fokus dieser Vorlesung liegt auf der **Rechtzeitigkeit**



- [1] DaimlerChrysler AG:  
Der neue Maybach.  
In: *ATZ/MTZ Sonderheft* (2002), Sept., S. 125
  
- [2] Deutsches Institut für Normung:  
*DIN 44300: Informationsverarbeitung — Begriffe.*  
Berlin, Köln : Beuth-Verlag, 1985
  
- [3] IFIP:  
*Working Group 10.4 on Dependable Computing and Fault Tolerance.*  
<http://www.dependability.org/wg10.4>, 2003
  
- [4] Liu, J. W. S.:  
*Real-Time Systems.*  
Englewood Cliffs, NJ, USA : Prentice Hall PTR, 2000. –  
ISBN 0-13-099651-3

