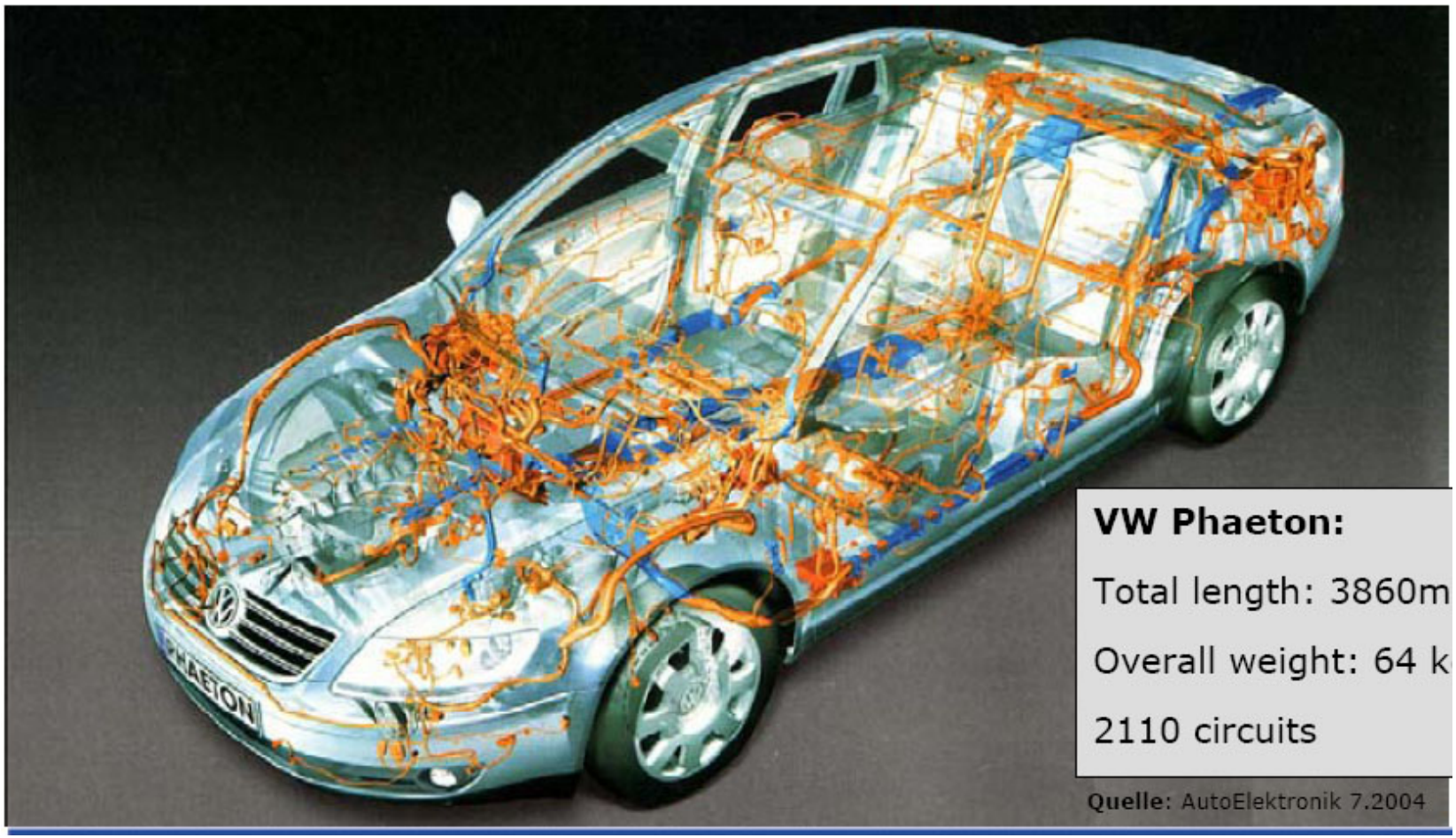
Seminarvortrag "Modellierung von hybriden ET und TT Systemen"

Yegor Yefremov

HAW-Hamburg

November 24, 2006

Motivation


VW Phaeton:
Total length: 3860m
Overall weight: 64 k
2110 circuits
Quelle: AutoElektronik 7.2004

Inhaltsverzeichnis

1. Einleitung
   - Problemstellung
   - Related work
2. Begriffe
   - Echtzeitsysteme
   - Ereignisgesteuerte Systeme
   - Zeitgesteuerte Systeme
3. Modellierung
   - Moore/Mealy Automaten
   - Hierarchische Automaten
   - AIRA
4. Vereinigung
5. Zusammenfassung
6. Literatur

Problemstellung

- Verteilung, Heterogenität
- kritische und nicht kritische Anwendungen
- ereignis- und zeitgesteuerte Anwendungen

Föderierte vs Integrierte Verteilte Systeme

Föderierte Systeme

Jede Anwendung (z.B. Autopilot, Brake-by-wire) hat eigenes Rechnersystem mit einer möglichen internen Redundanz.

- Gegenseitige hardwaremäßige Trennung von Anwendungsfunktionen
- Erhöhung der Transparenz von Interaktionen und Abhängigkeiten zwischen autonomen Systemen

Integrierte Systeme

Mehrere Anwendungen sind in einigen Rechnersystemen integriert.

- Kostenreduktion

Trend

Neues System, das Vorteile von föderierten und integrierten Systemen nutzt:

- Mehrfachnutzung der Hardware durch mehrere Anwendungen **(Kostenreduktion)**
- einheitliches Kommunikationssystem **(Transparenz)**
- Softwaremäßiges Abkapseln von kritischen und nicht kritischen Anwendungen, realisierbar durch ein geeignetes Framework
- hybrides Modellierungskonzept für ereignis- und zeitgesteuerte Systeme

Related work

Folgende Systeme für gemischte Anwendungen werden eingesetzt:

- MAFT (Multicomputer Architecture for Fault-Tolerance)
- IMA (Integrated Modular Avionics)
- OSEK/VDX und OSEKtime
- BASEMENT

Projekte:

- ARTIST2

Echtzeitsysteme

Echtzeitsysteme

Ein Echtzeitcomputersystem ist ein System, in dem die Korrektheit des Systemverhaltens nicht nur von logischen Ergebnissen der Berechnungen abhängt, sondern der Einhaltung von bestimmten Zeitgrenzen (deadlines) zur Ermittlung der logischen Ergebnissen. Der Ergebnis muss **rechtzeitig** vorliegen.

Weiche und harte Zeitgrenzen

Weiche Zeitgrenzen

Falls ein Ergebnis nutzbar bleibt, sogar nachdem die Zeitgrenze überschritten ist, so wird die Zeitgrenze als weich (soft) bezeichnet.

Harte Zeitgrenzen

Zeitgrenze ist hart (hard), falls sich die fatalen Folgen wegen der Überschreitung der Zeitgrenze ergeben können.

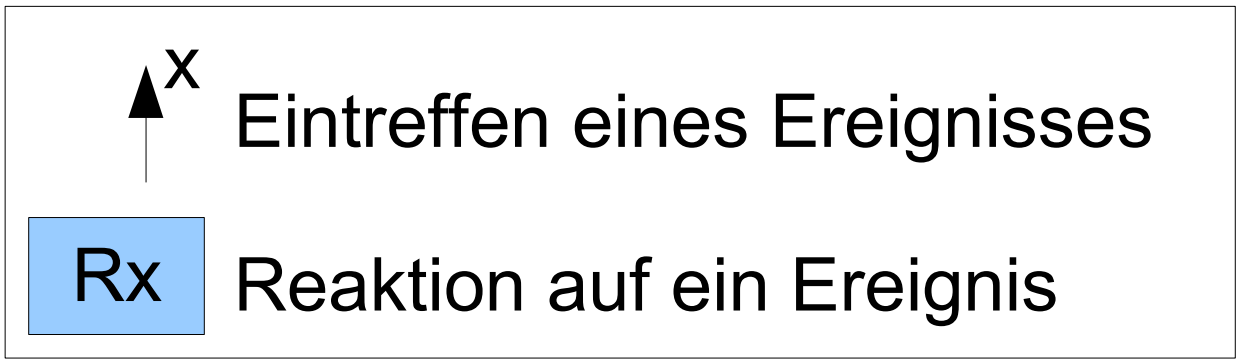
Ereignisgesteuerte Systeme

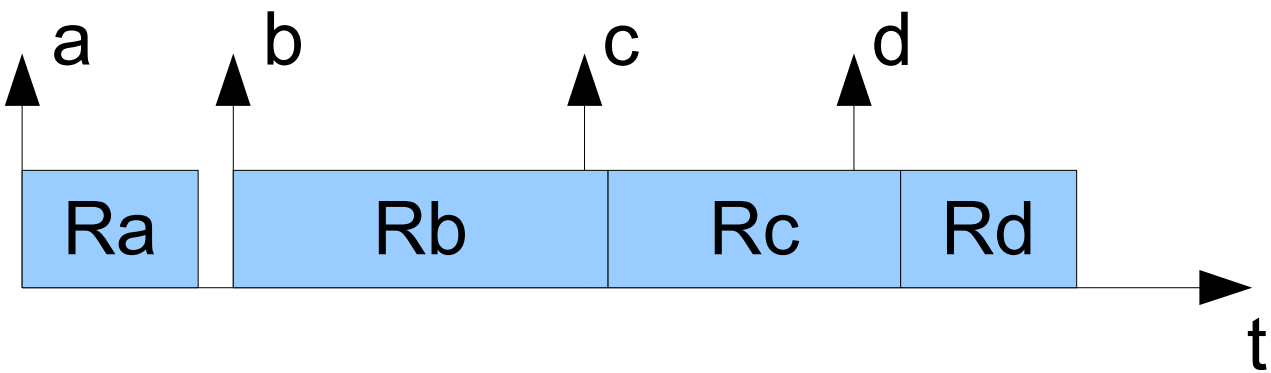
Bei der Ereignissteuerung wird auf ein von außen kommendes Ereignis reagiert, d. h. eine Verarbeitung gestartet.

Beispielhafte Anwendungen:

- diskrete Prozesse
- Benutzerinteraktionen
- Auftreten von Alarmen

Ereignisgesteuerte Systeme





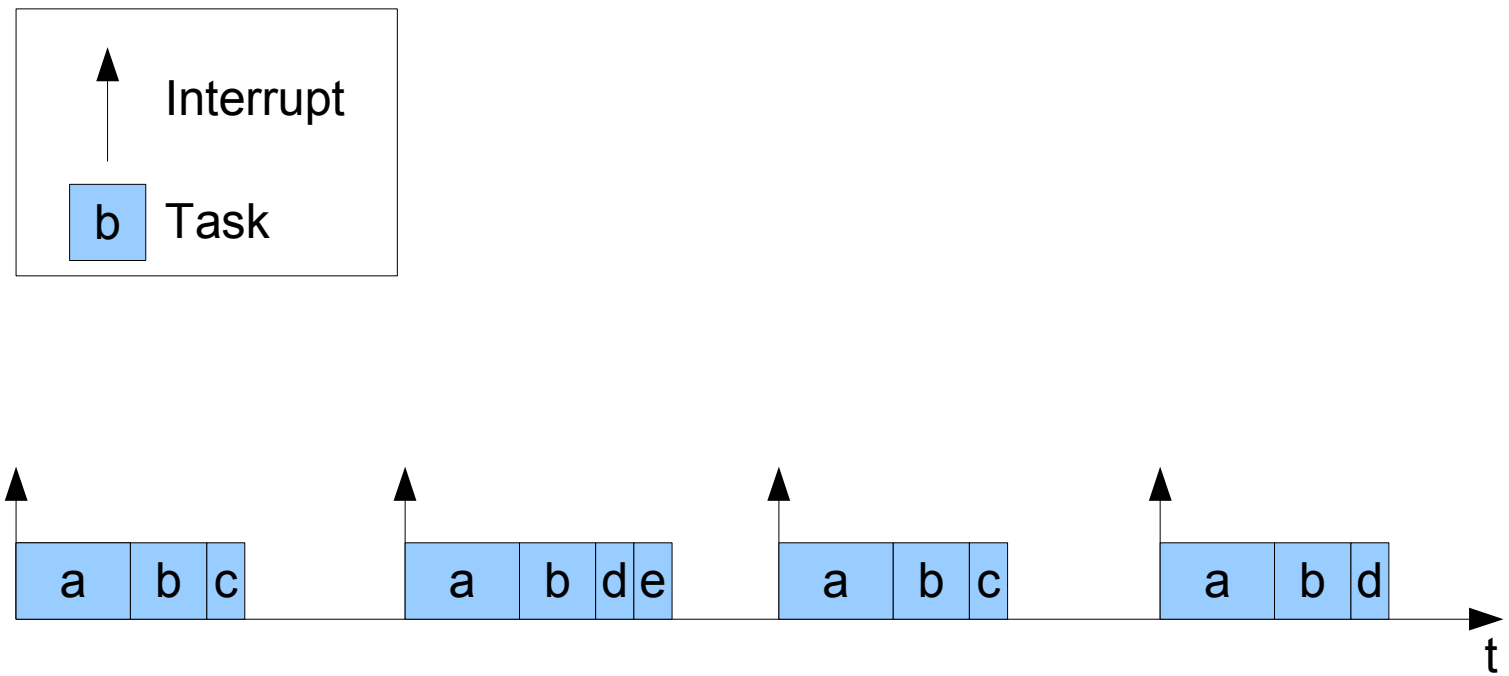
Zeitgesteuerte Systeme

Bei der Zeitsteuerung werden Verarbeitungen auf Grund eines vorher festgelegten Zeitplans gestartet.

Beispielhafte Anwendungen:

- zyklische Prozesse wie z.B. Regelung
- Ampelsteuerung

Zeitgesteuerte Systeme



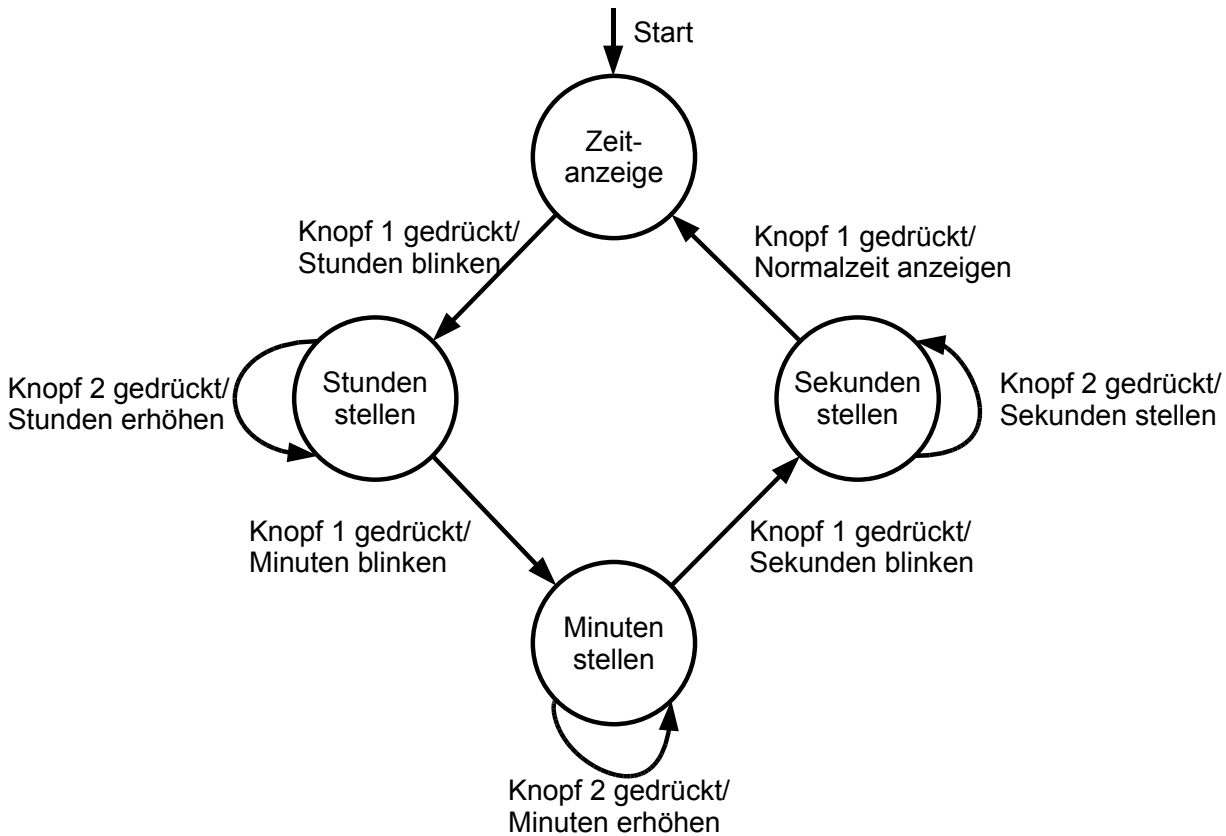
Moore/Mealy

Definition: Ein Moore/Mealy Automat ist ein 6-Tupel $A = (Q, \Sigma, T, \delta, \gamma, q_0)$, wobei gilt:

- Q ist eine nichtleere endliche Menge von Zuständen
- Σ ist ein endlicher Zeichenvorrat (also ein Alphabet)
- T ist das Ausgabealphabet
- $\delta : Q \times \Sigma \rightarrow Q$ ist eine Übergangsfunktion
  - $\delta(q, a)$ ist ein definierter Zustand für jeden Zustand q und Eingabe $a \in \Sigma$
- γ ist eine Ausgabefunktion:
  - **Moore** $\gamma : Q \rightarrow T$
  - **Maely** $\gamma : Q \times \Sigma \rightarrow T$
- $q_0 \in Q$ ist ein Anfangszustand

Beispiel: Mealy-Automat

- Ereignisse:
  - Knopf 1 gedrückt
  - Knopf 2 gedrückt
- Ausgaben:
  - Stunden blinken
  - Stunden erhöhen
  - Minuten blinken
  - Minuten erhöhen
  - Sekunden blinken
  - Sekunden stellen
  - Normalzeit anzeigen



Hierarchische Automaten

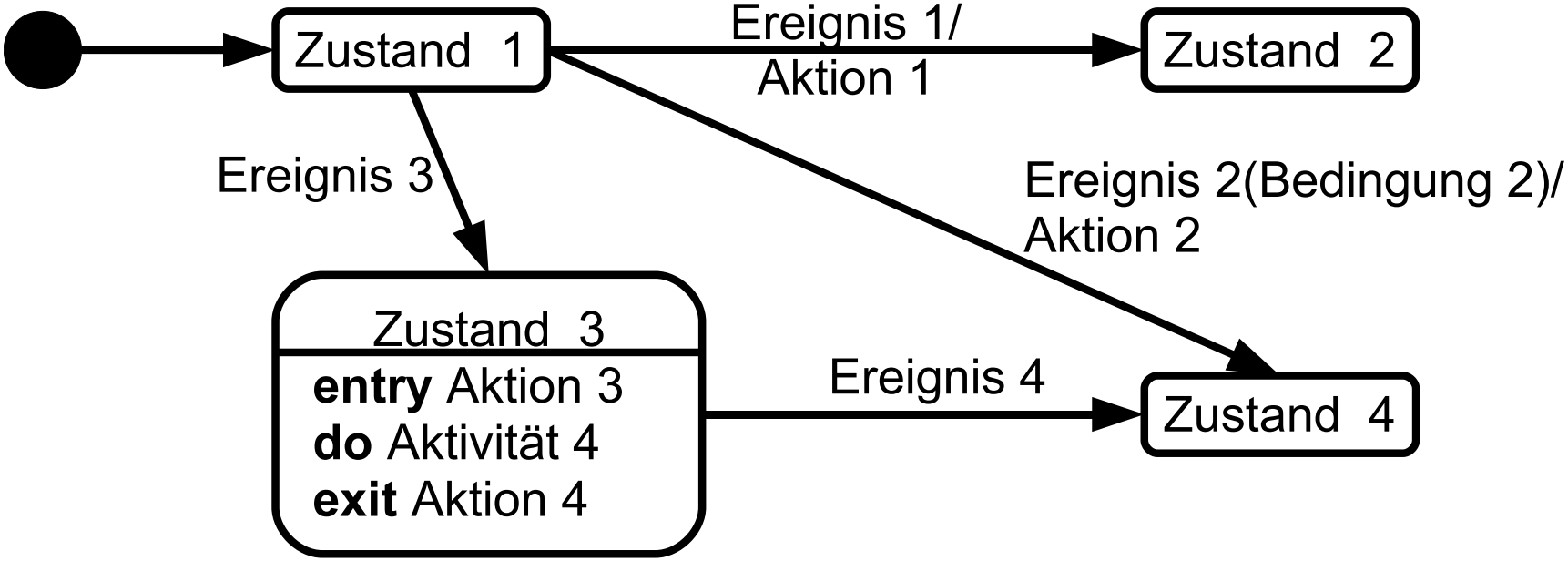
Informelle Definition:

- eine Zustandsmenge Q, die hierarchisch organisiert sein kann (durch Superzustände)
- eine Eingabemenge Σ von Ereignissen **(Nachrichten)**
- eine Menge von Prädikaten Π für Wächter
- eine Ausgabemenge T **(Nachrichten)**
- einen Anfangszustand q_0
- Zwei Funktionen mit Ausgabe:
  - Überführungsausgabe $\delta : Q \times \Sigma \times \Pi \rightarrow T \times Q$
  - Zustandsausgabe $\gamma : Q \times \Sigma \times \Pi \rightarrow T$
- Aktivierung von Subautomaten

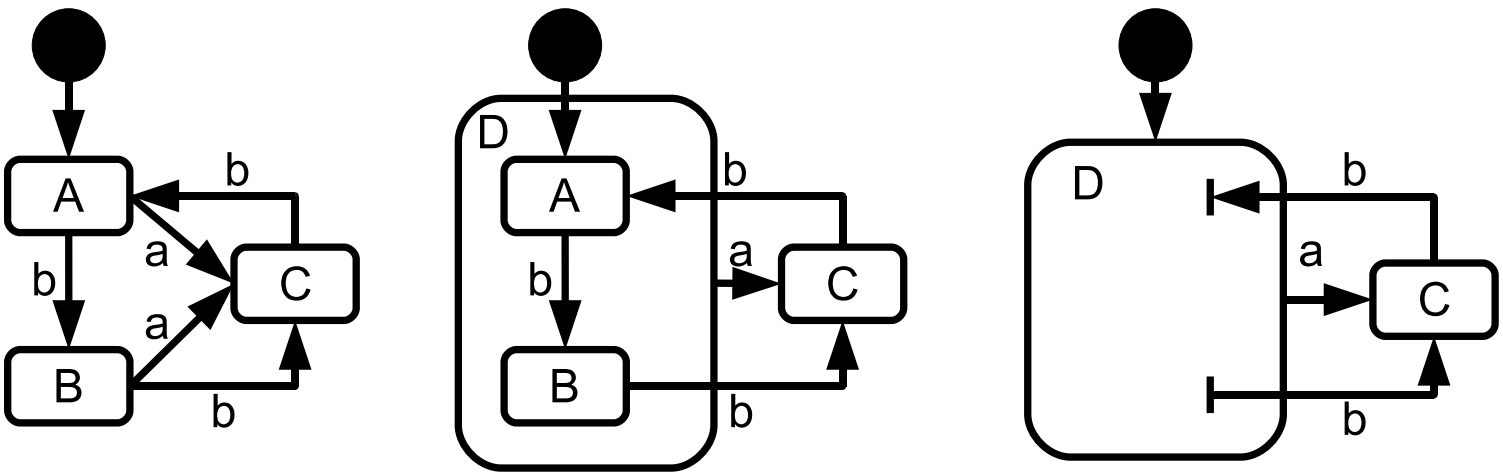
Eigenschaften von hierarchischen Automaten

- Hierarchische Zustandsebenen
- Orthogonale Subautomaten
- bewachte Reaktionen
- Vereinigung von Moore und Mealy Konzepten. Für Zustand wird optional ausgeführt:
  - **entry-Aktion:** ausgeführt beim Betreten des Zustands
  - **do-Aktivität:** ausgeführt solange im Zustand
  - **exit-Aktion:** ausgeführt beim Verlassen
  - zustandserhaltende und zustandsverändernde Reaktionen
- Zustände mit Wiederentrittsgedächtnis

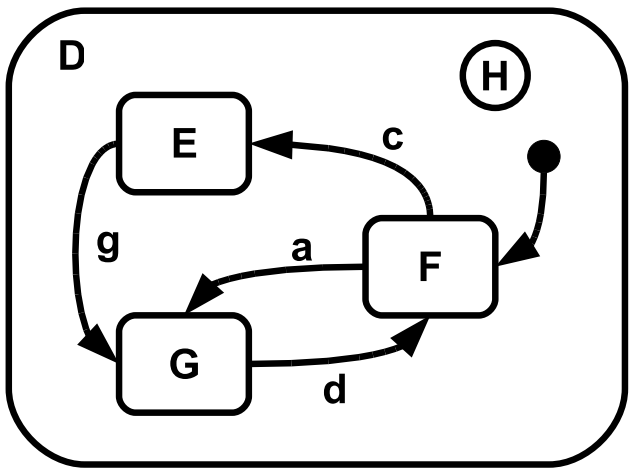
Beispiel: hierarchischer Automat



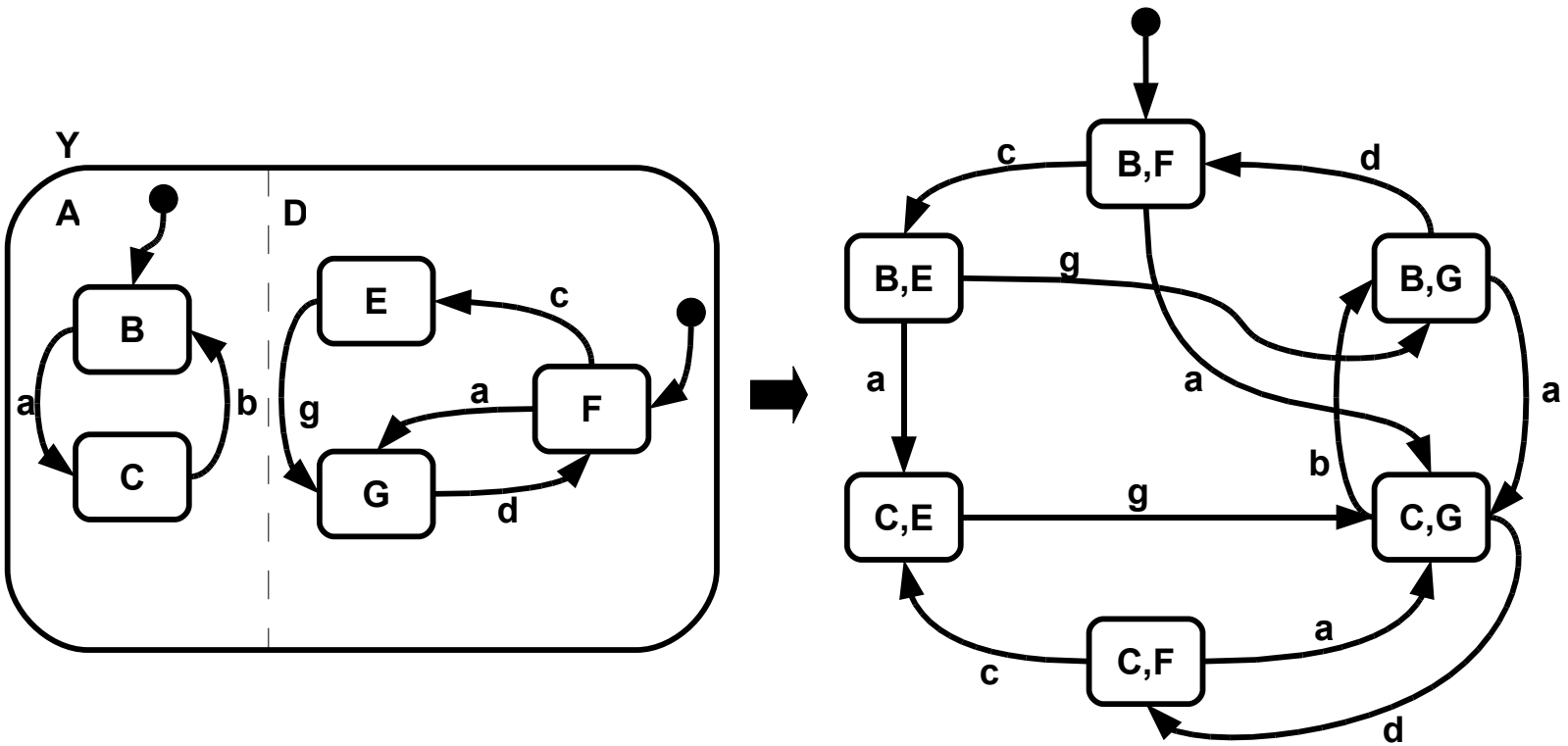
Beispiel: hierarchischer Automat



Beispiel: hierarchischer Automat mit Wiederentrittsgedächtnis



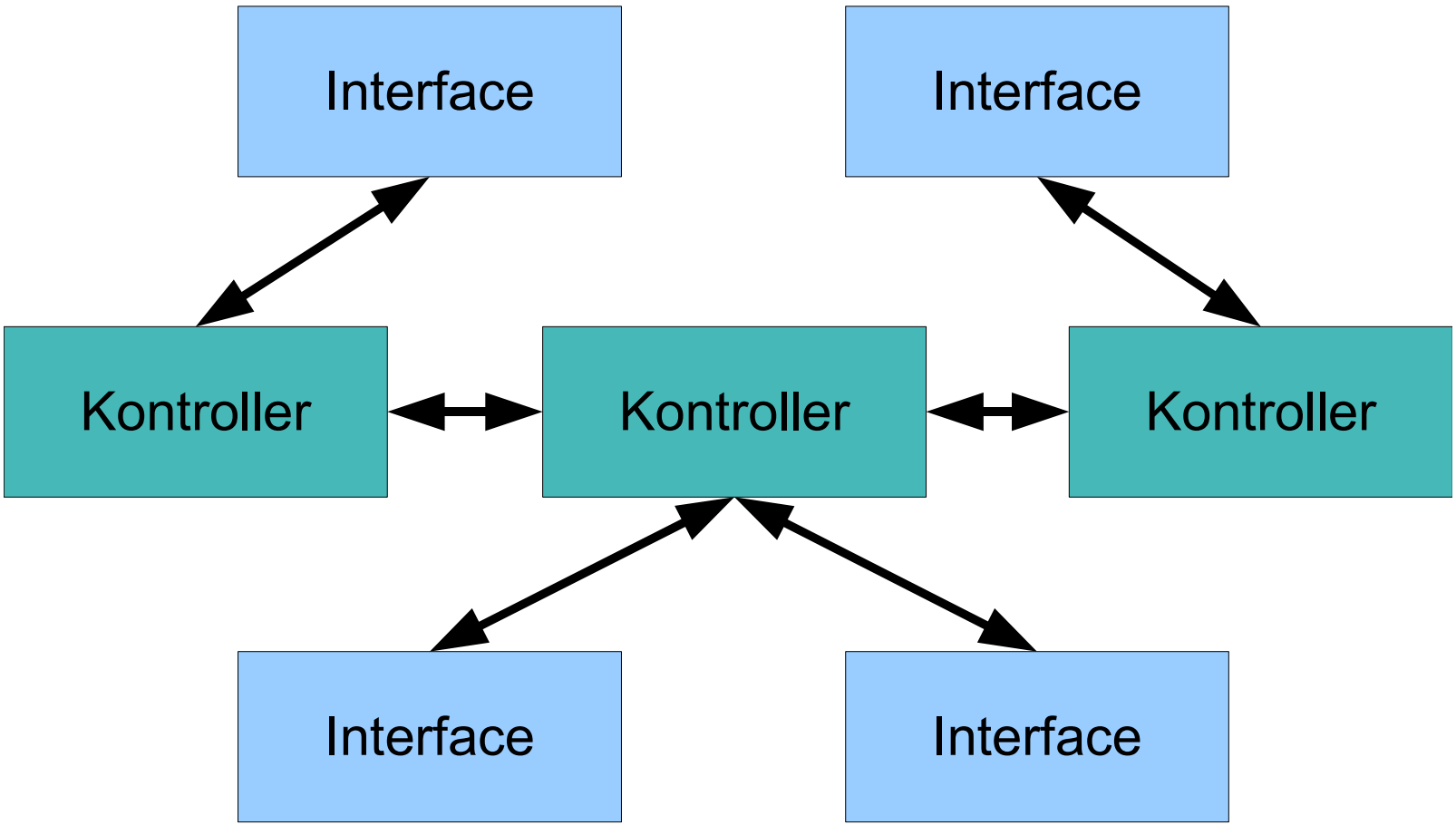
Beispiel: Orthogonaler (AND) Automat



Automaten in reaktiven Anwendungen (AIRA)

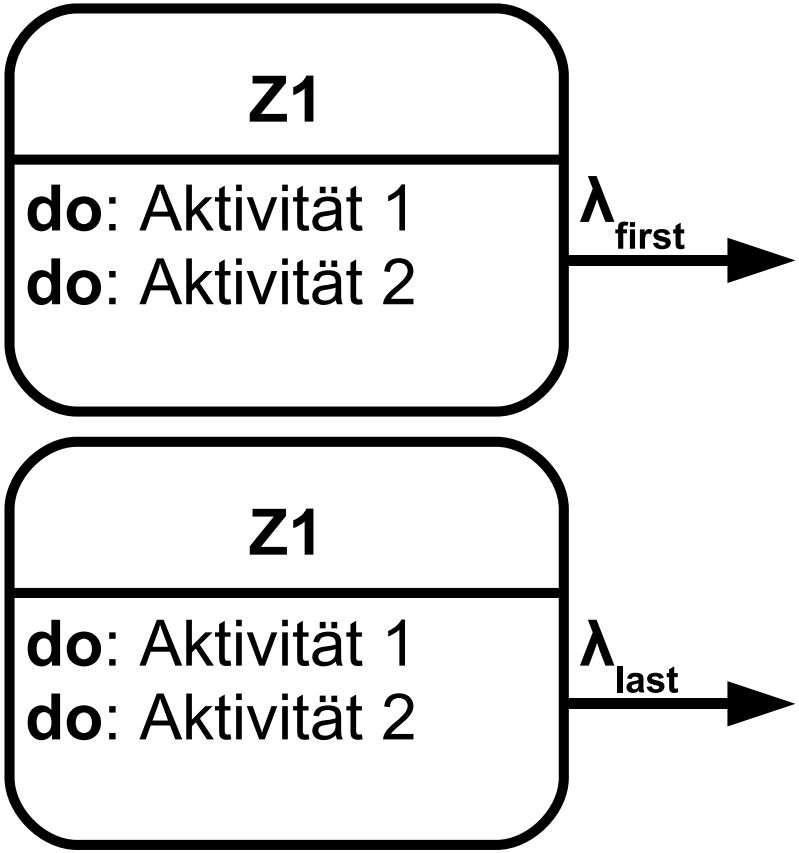
- Eine Modellierungssprache, die auf auf dem Konzept von hierarchischen Automaten basiert
- Beschreibung der Verteilung im System: Kontroller, Interface
- Trennung von Bearbeitungssoftware und Automaten
- Zeitsteuerung
- besondere λ-Übergänge
- Abschaffung von Wiederentrittsgedächtnis
- determinierende Semantik der Ausführungsreihenfolge

AIRA: Systemübersicht



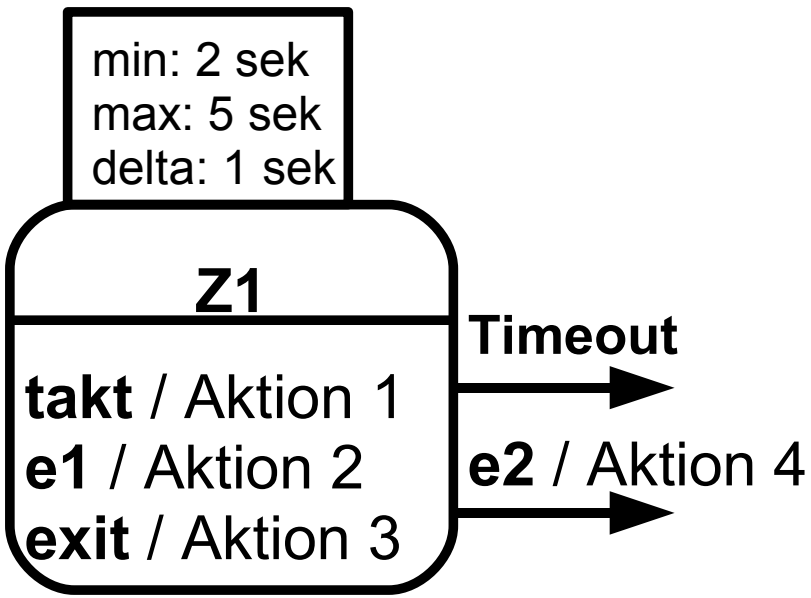
λ-Übergänge in AIRA

- λ_{first}: *Zustand Z1* wird verlassen, nachdem entweder *Aktivität 1* **oder** *Aktivität 2* beendet ist
- λ_{last}: *Zustand Z1* wird verlassen, nachdem *Aktivität 1* **und** *Aktivität 2* beendet ist



Zeit in AIRA

- **min**: zustandsändernde Reaktionen werden erst nach 2 Sekunden freigeschaltet
- **max**: *Aktion 3* wird nach 5 Sekunden ausgeführt
- **delta(takt)**: *Aktion 1* wird jede Sekunde ausgeführt



Semantik zur Reaktionsfindung

Nur ein Subautomat mit eingebettet

- unterste aktive Zustand wird gesucht und geprüft
  1. die internen Reaktionen werden geprüft. Wenn eine Reaktion möglich ist, hat das System reagiert
  2. externe Reaktionen prüfen. Wenn eine Reaktion möglich ist, hat das System reagiert
- wenn ein unterer Zustand nicht reagieren konnte, erfolgt die Prüfung eine Ebene höher

Mehrere Automaten sind eingebettet

- Baum der aktiven Zustände wird durchsucht
  - für jeden "Zustands"-Blatt wird nach obigem Verfahren gehandelt

OSEK/VDX

Standardisierung wichtiger Elemente einer fahrzeugweiten Elektronik-Infrastruktur

- Betriebsystem (OSEK/VDX-OS)
- Kommunikationssystem (OSEK/VDX-COM)
- Netzmanagement (OSEK/VDX-NM)
- OSEK Implementation Language (OSEK/VDX-OIL)

OSEKtime

OSEK/VDX: Nachteile

OSEK/VDX-OS Standard garantiert alleine keine globale Zeit für alle Steuergeräte

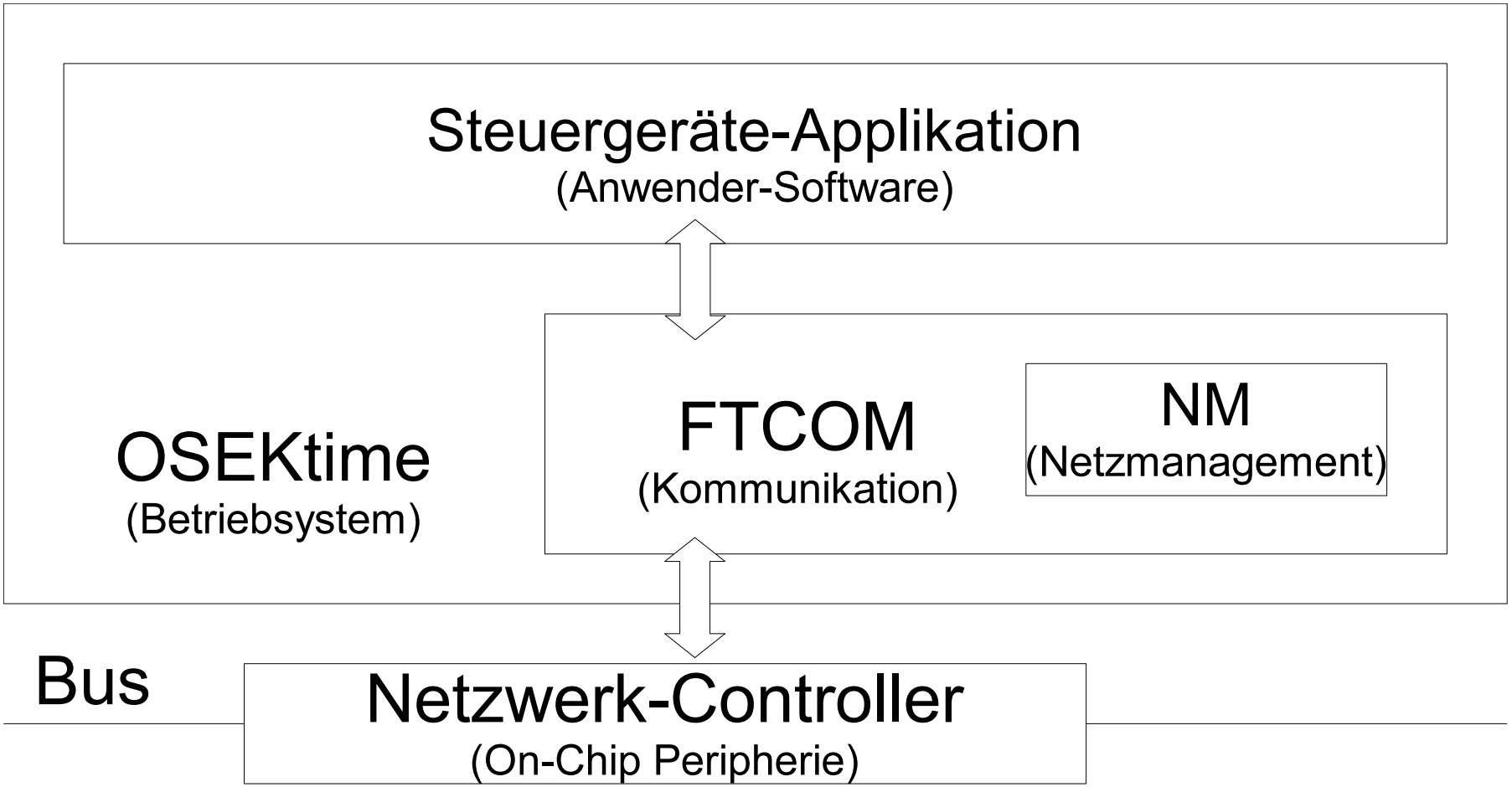
Daher

OSEKtime

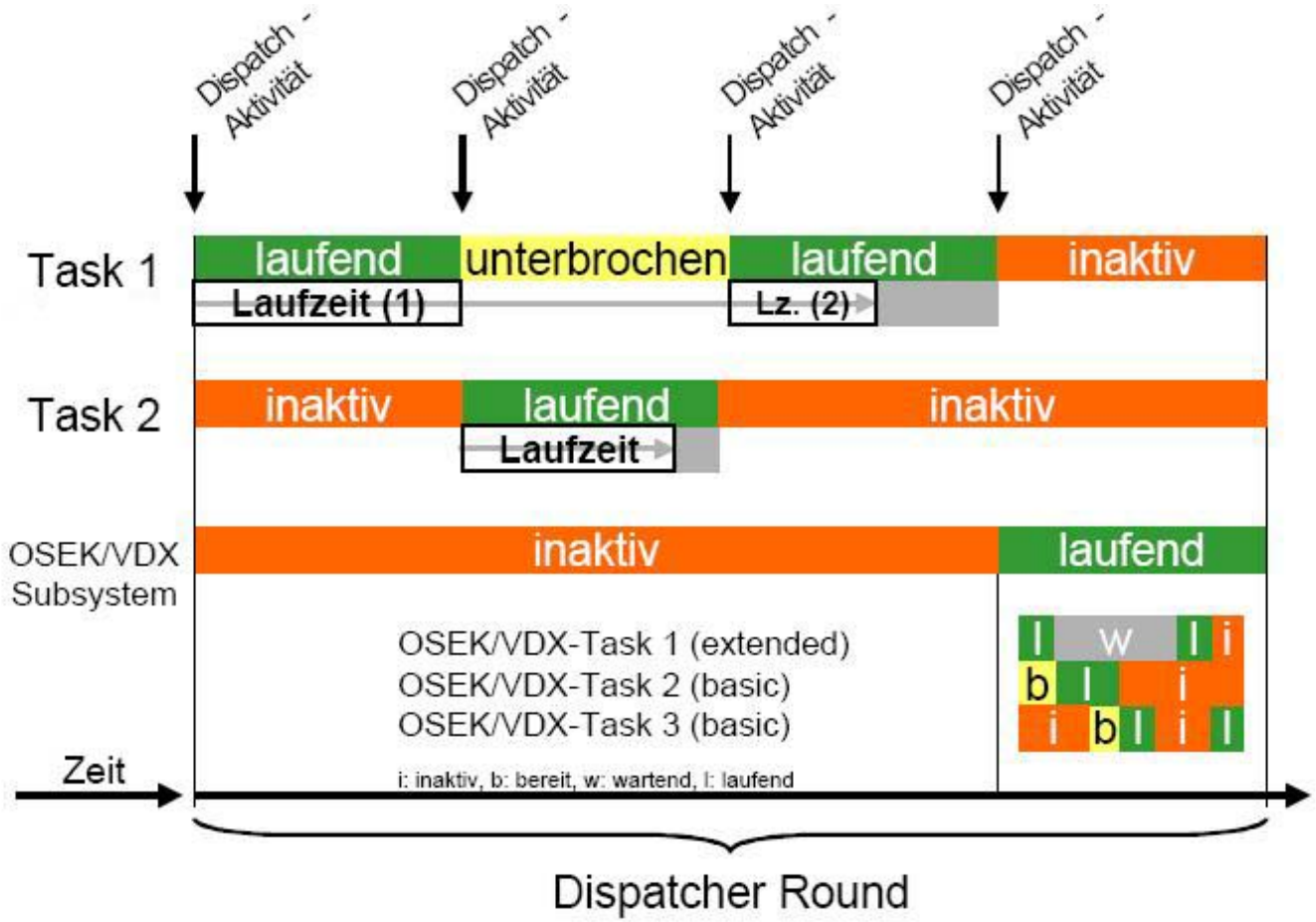
Zeitgesteuertes Betriebsystem als Ergänzung zu OSEK/VDX
OSEKtime besteht aus zwei Komponenten:

- Betriebsystemkern OSEKtime
- spezielles Kommunikationssystem FTCOM (Fault Tolerant COMunikation)

OSEKtime: Struktur



OSEKtime: TDMA



Möglichkeiten der Vereinigung von hybriden Systemen

Getrennte CPUs und getrennte Modellierung

Vorteile und Nachteile von föderierten Systemen

Ein CPU und getrennte Modellierung

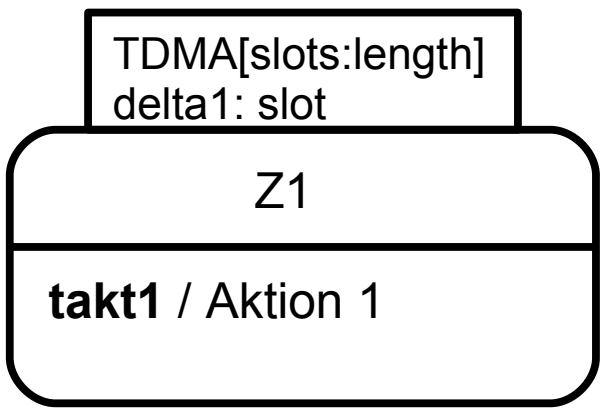
- OSEKtime
- Software SPS

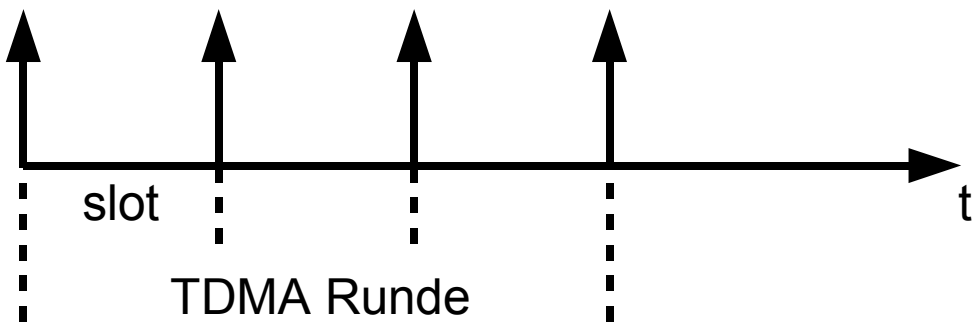
Ein CPU und eine Modellierungssprache

Erweiterung von AIRA Laufzeit, um zeitgesteuertes Konzept

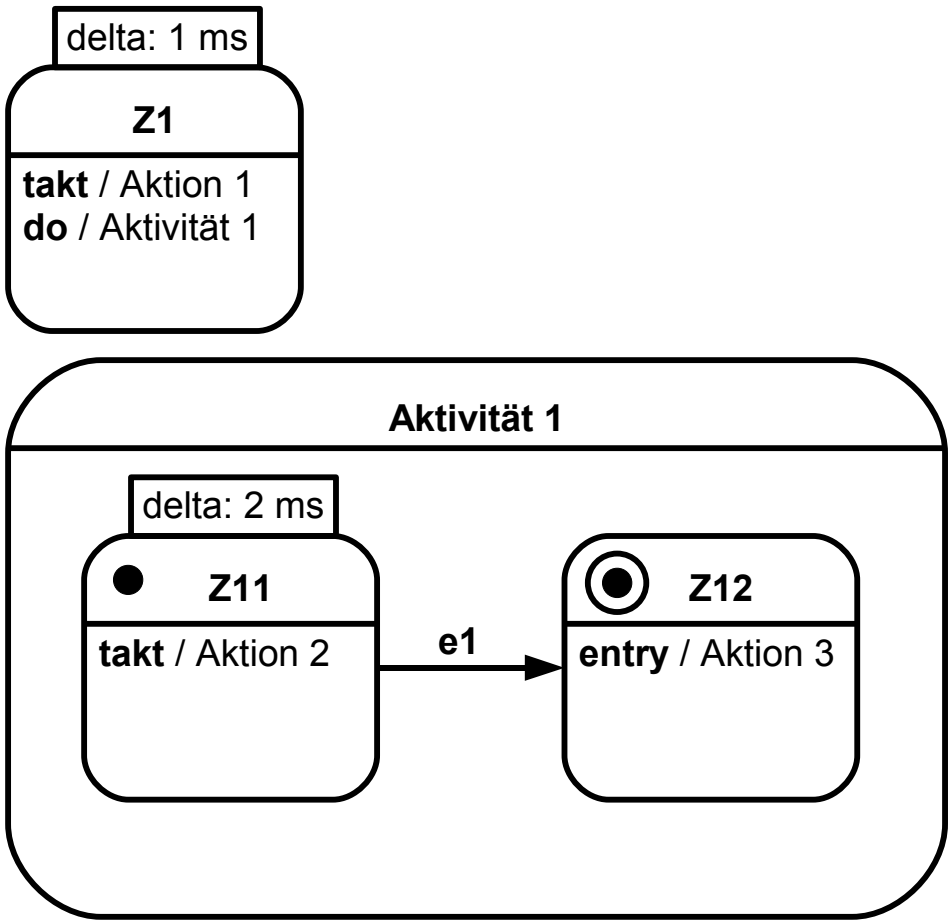
Zusammenfassung

- Modellierung: Erweiterung der AIRA-Sprache

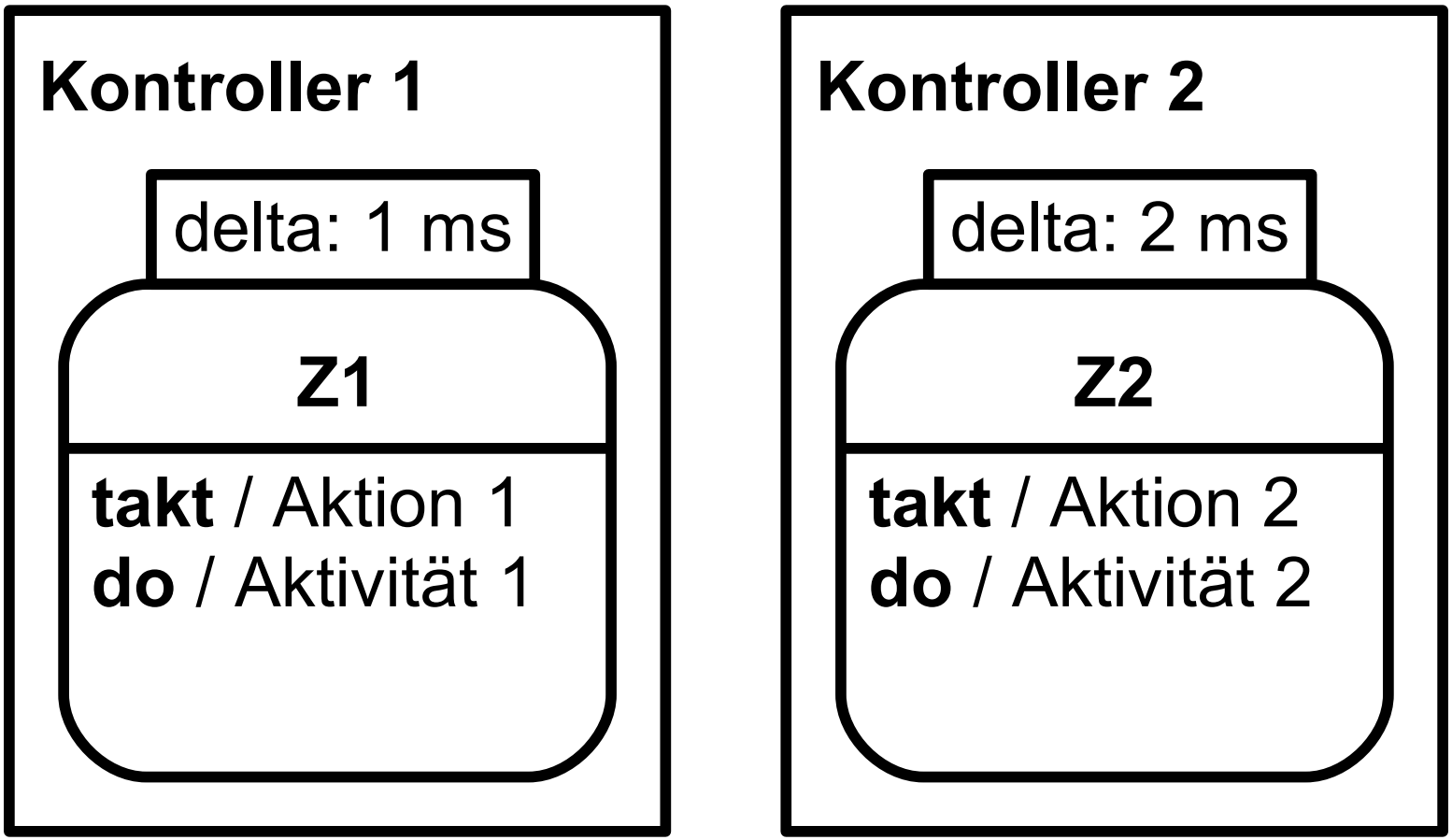




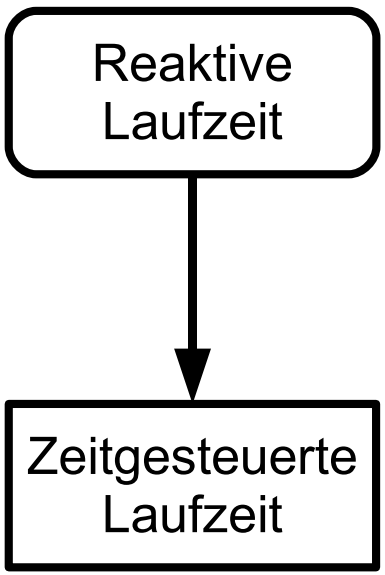
Zusammenfassung

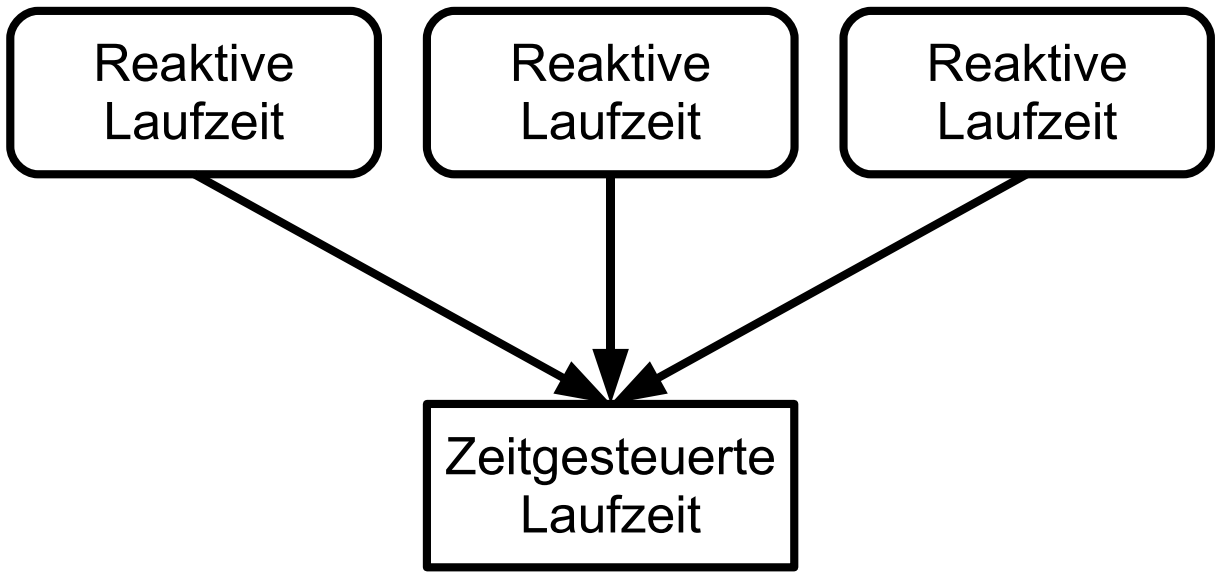


Zusammenfassung



Zusammenfassung





Literatur

Roman Obermeisser
Event-Triggered and Time-Triggered Control Paradigms.
Springer Science, 2005.

Jochen Schoof
OSEKtime - Standard für zeitgesteuerte Betriebsysteme.
3SOFT GmbH Erlangen, 2002.

David Harel
Statecharts: A Visual Formalism for Complex Systems.
Elsevier Science Publishers B.V., 1987.

Alexander Metzner, Ingo Stierand
Analyzing Mixed Event Triggered/Time Triggered Systems.
IEEE RTAS'06, 2006

Danke für Ihre Aufmerksamkeit!