Modellierung verteilter Systeme

Grundlagen der Programm und Systementwicklung

Sommersemester 2012

Prof. Dr. Dr. h.c. Manfred Broy

Unter Mitarbeit von Dr. M. Spichkova, J. Mund, P. Neubeck

Lehrstuhl Software & Systems Engineering

Nebenläufigkeit und Parallelität

Bisher: **Sequentielle** Systeme; Zustandsübergänge nacheinander

Jetzt: **Verteilte** Systeme; Zustandsübergänge zeitlich nebeneinander oder parallel

Modell: Komposition von Zustandsmaschine

gemeinsamer Zustandsraum durch Produkt/Summe

Produkt der Zustandsräume

Gegeben Zustandsmaschinen $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$
Produkt der Zustandsräume:

$\Sigma = \Sigma_1 \times \Sigma_2$ mit $\alpha = (\alpha_1, \alpha_2)$ und mit der Zustandsübergangsfunktion

$$\Delta: \Sigma \rightarrow \wp(\Sigma)$$

Komposition durch **Interleaving**:

$\Delta(\sigma_1, \sigma_2) = \{(\tau_1, \tau_2) \in \Sigma \mid (\tau_1 \in \Delta_1(\sigma_1) \wedge \tau_2 = \sigma_2)$ *** Fortschritt Z_1

$\vee \ (\tau_2 \in \Delta_2(\sigma_2) \wedge \tau_1 = \sigma_1) \ \}$ *** Fortschritt Z_2

Die Zustandsmaschine führen jeweils nur einen Schritt aus.

Produkt der Zustandsräume

Gegeben Zustandsmaschinen $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$

Produkt der Zustandsräume:

$\Sigma = \Sigma_1 \times \Sigma_2$ mit $\alpha = (\alpha_1, \alpha_2)$ und der Zustandsübergangsfunktion

$$\Delta: \Sigma \rightarrow \wp(\Sigma)$$

Komposition durch Interleaving mit **Stuttering:**

$$\Delta(\sigma_1, \sigma_2) = \{(\tau_1, \tau_2) \in \Sigma \mid (\tau_1 \in \Delta_1(\sigma_1) \wedge \tau_2 = \sigma_2)$$ *** Fortschritt Z_1

$$\vee \ (\tau_2 \in \Delta_2(\sigma_2) \wedge \tau_1 = \sigma_1)$$ *** Fortschritt Z_2

$$\vee \ (\tau_1 = \sigma_1 \wedge \tau_2 = \sigma_2)\}$$ *** kein Fortschritt

Zustandsmaschine führen jeweils keinen oder nur einen Schritt aus.

Produkt der Zustandsräume

Gegeben Zustandsmaschinen $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$

Produkt der Zustandsräume:

$\Sigma = \Sigma_1 \times \Sigma_2$ mit $\alpha = (\alpha_1, \alpha_2)$ und der Zustandsübergangsfunktion

$$\Delta: \Sigma \rightarrow \wp(\Sigma)$$

Komposition durch **synchrone Nebenläufigkeit** („und“-Parallelität)

$$\Delta(\sigma_1, \sigma_2) = \{(\tau_1, \tau_2) \in \Sigma \mid \tau_1 \in \Delta_1(\sigma_1) \wedge \tau_2 \in \Delta_2(\sigma_2)\}$$

Die Zustandsmaschine führen jeweils nur gemeinsam Schritte aus.

Produkt der Zustandsräume

Gegeben Zustandsmaschinen $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$

Produkt der Zustandsräume:

$\Sigma = \Sigma_1 \times \Sigma_2$ mit $\alpha = (\alpha_1, \alpha_2)$ und der Zustandsübergangsfunktion

$$\Delta: \Sigma \rightarrow \wp(\Sigma)$$

Komposition durch **Mischform** von Interleaving und Synchron

$$\Delta(\sigma_1, \sigma_2) = \{(\tau_1, \tau_2) \in \Sigma : (\tau_1 \in \Delta_1(\sigma_1) \wedge \tau_2 = \sigma_2)$$ *** Fortschritt Z_1

$$\vee\ (\tau_2 \in \Delta_2(\sigma_2) \wedge \tau_1 = \sigma_1)$$ *** Fortschritt Z_2

$$\vee\ (\tau_1 \in \Delta_1(\sigma_1) \wedge \tau_2 \in \Delta_2(\sigma_2))\ \}$$ *** Fortschritt Z_1 und Z_2

Zustandsmaschine führen jeweils gemeinsam oder einzelnen Schritte aus.

Summe disjunkter Zustandsräume

Zustandsmaschine $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$

Summe der Zustandsräume:

$\Sigma = \Sigma_1 \cup \Sigma_2$ mit Anfangszustand $\alpha = \alpha_1$ oder $\alpha = \alpha_2$

und der Zustandsübergangsfunktion $\Delta: \Sigma \rightarrow \wp(\Sigma)$ definiert ist durch

$$\Delta(\sigma) = \Delta_1(\sigma) \cup \Xi_1(\sigma) \qquad \text{falls } \sigma \in \Sigma_1$$

$$\Delta(\sigma) = \Delta_2(\sigma) \cup \Xi_2(\sigma) \qquad \text{falls } \sigma \in \Sigma_2$$

Dabei sind

$$\Xi_1: \Sigma_1 \rightarrow \wp(\Sigma_2) \text{ und } \Xi_2: \Sigma_2 \rightarrow \wp(\Sigma_1)$$

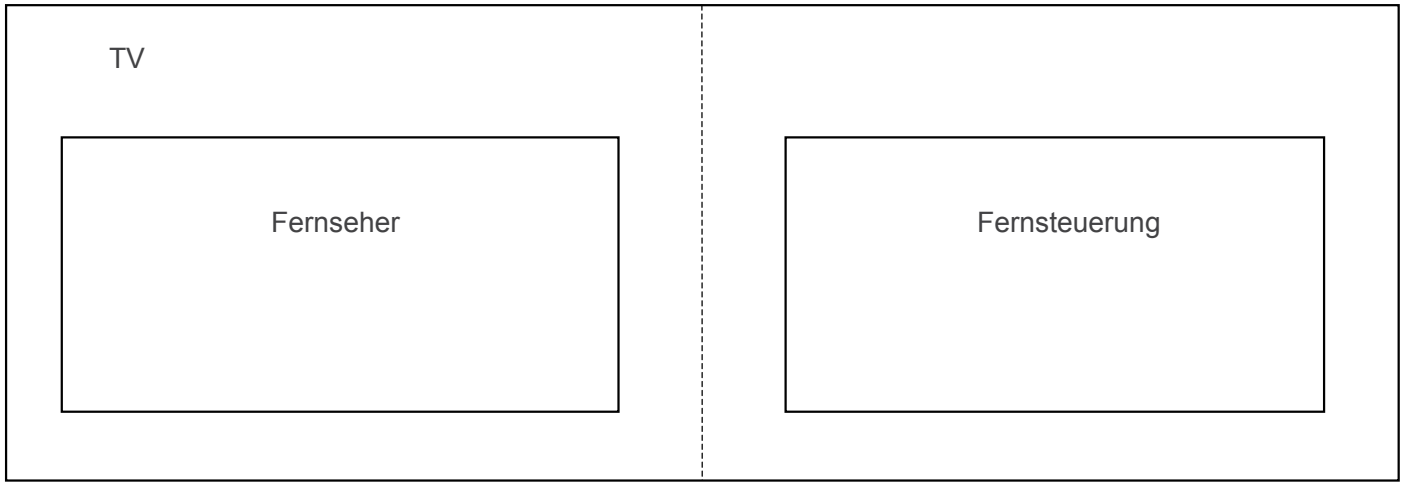
die Übergänge zwischen den Maschinen

Zustandsmaschine ist stets entweder in Zustand von Z_1 **oder** Z_2

Beispiel: Fernsehgerätsteuerung

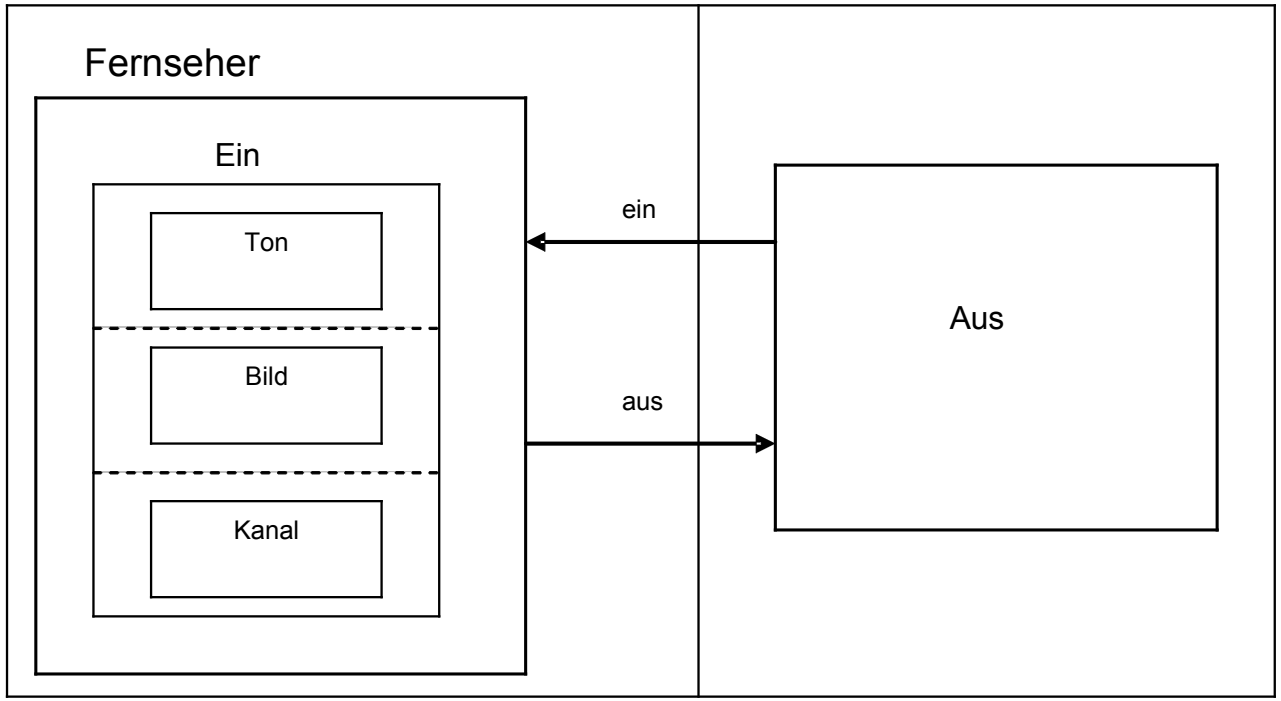
TV-System = Apparat und Fernbedienung

- 2 Zustandsmaschine nebenläufig zusammengesetzt

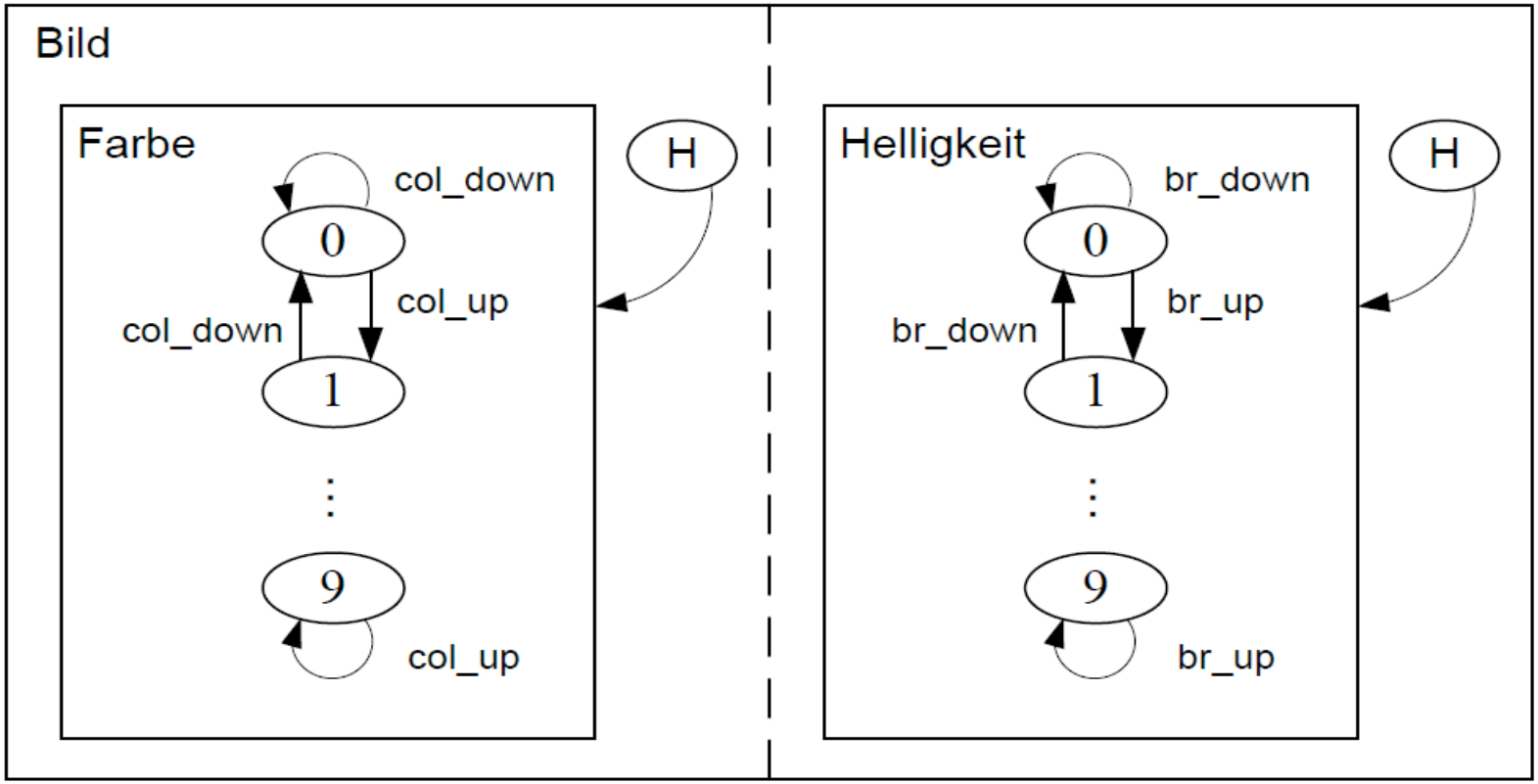


Beispiel: Fernsehgerätsteuerung

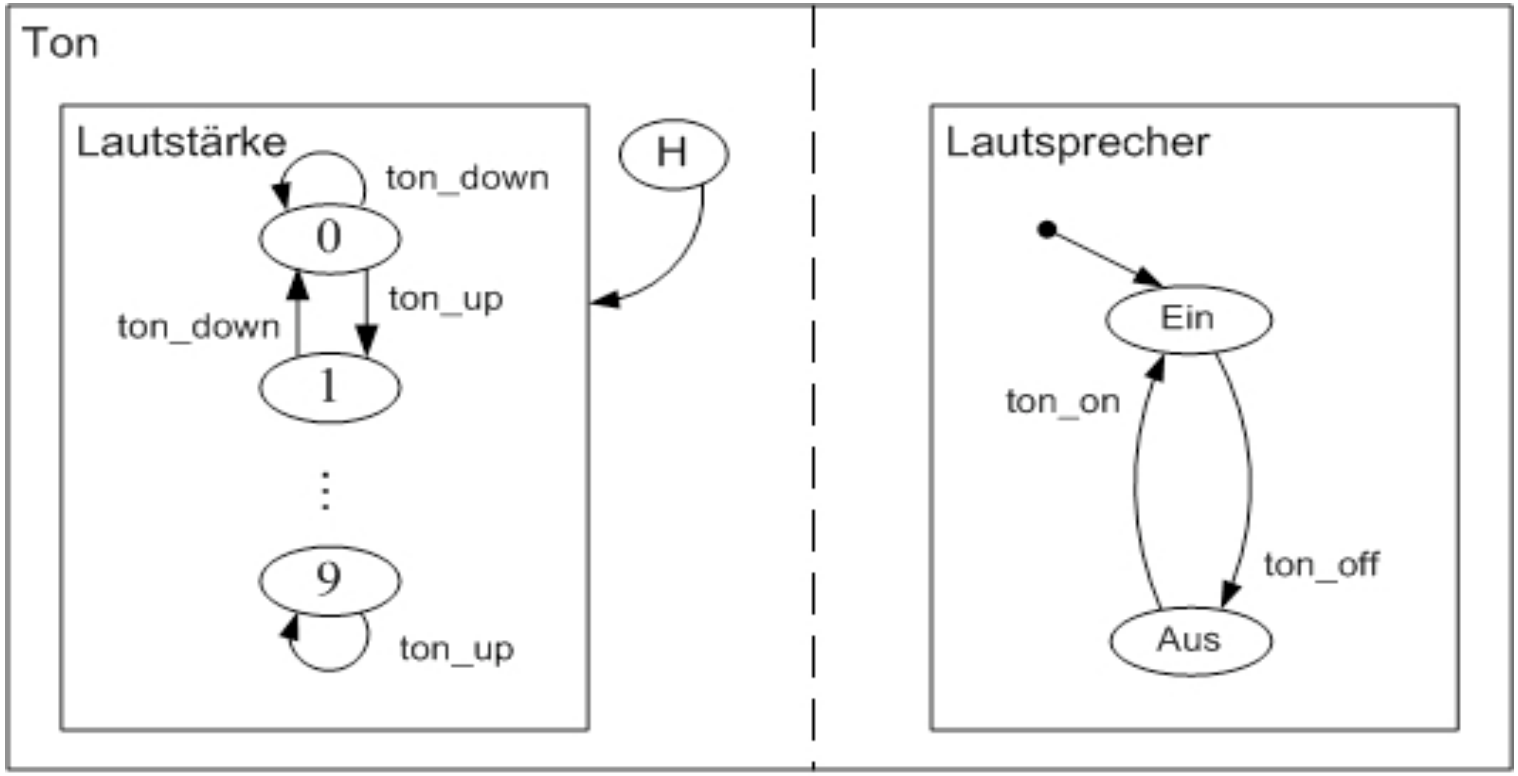
- Unterzustände bzw. parallele Zustandsmaschine (Ton, Bild, Kanal)
- Interaktion: Fernseher - Fernbedienung



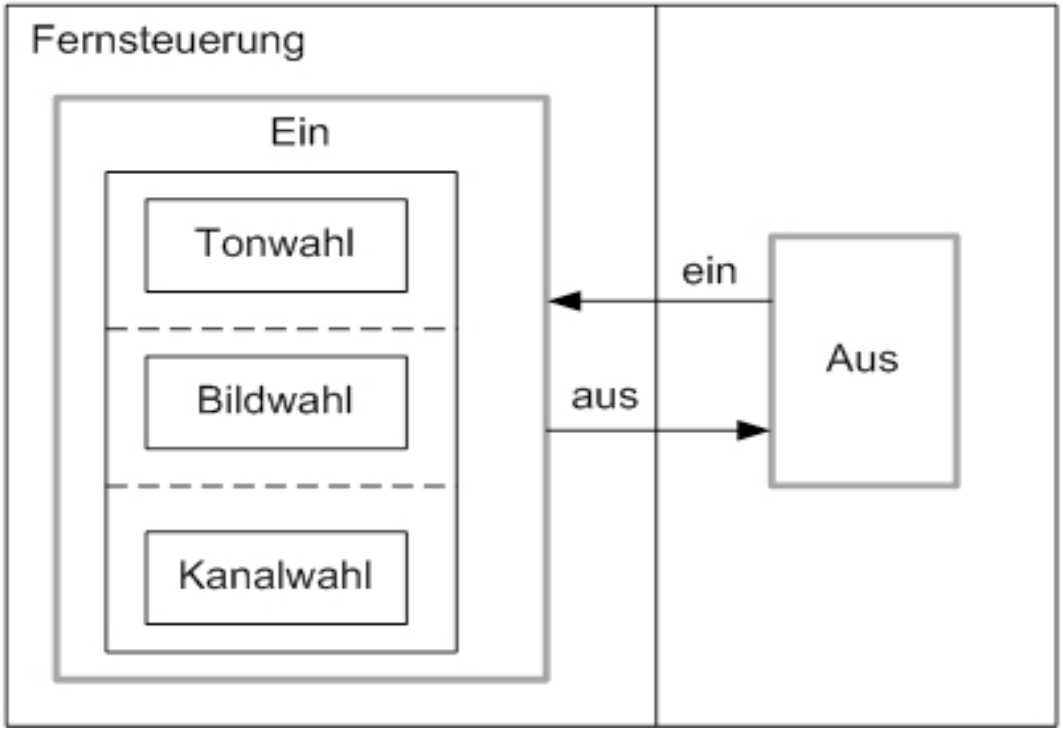
Beispiel: Fernsehgerätsteuerung



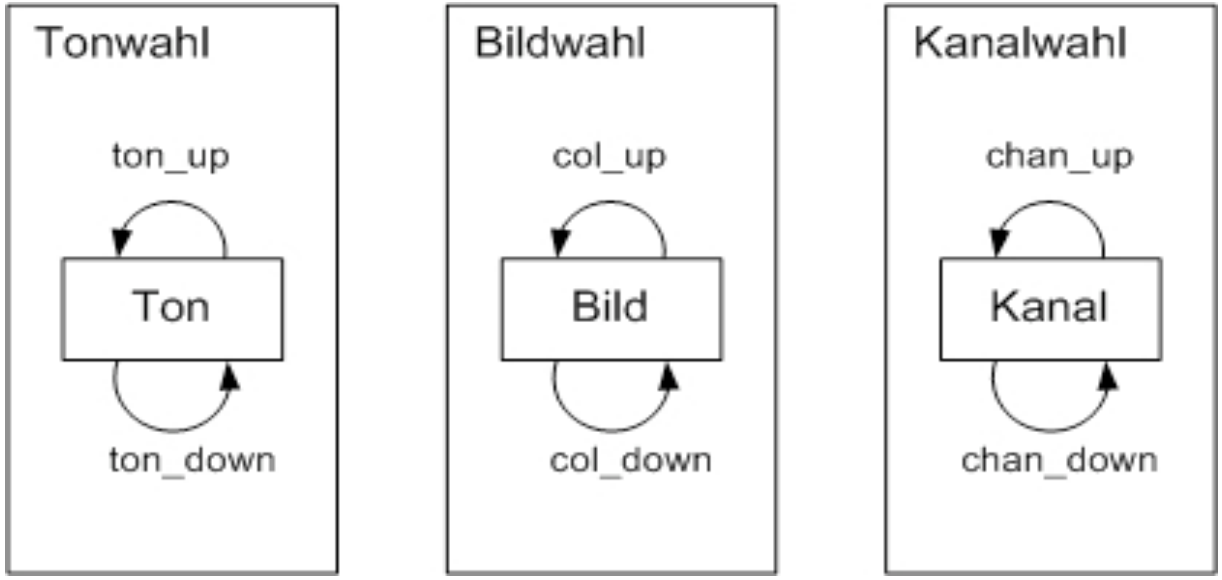
Beispiel: Fernsehgerätsteuerung



Beispiel: Fernsehgerätsteuerung



Beispiel: Fernsehgerätsteuerung: Fernbedienung



Beispiel: Fernsehgerätsteuerung

- TV-System = Apparat und Fernbedienung
  - 2 Zustandsmaschine nebenläufig zusammengesetzt

- Ausblick:
  - Komposition von Zustandsmaschine mit überlappenden Zustandsräumen

Parallele Programme mit gemeinsamen Variablen

Bisher: **Verteilte** Systeme; parallele Zustandsübergänge mit disjunkten ZR

Jetzt: **Verteilte** Systeme; parallele Zustandsübergänge mit gemeinsamen Attributen

Gemeinsame Variable

- Problem:
  - Konflikt bei synchronen Übergängen
  - Parallel ausgeführte Zustandsübergänge besetzen gemeinsame Attribute
- Gemeinsame Variable *(shared variable)*
  - Zustandsattribut
  - Zugriff durch unterschiedliche, parallel ablaufende Programme
  - Lesend und mindestens von einem schreibend
- Zweck:
  - Koordination, Synchronisation, Kommunikation

Komposition bei nicht-disjunkten ZR

Zustandsmaschinen: $Z_1 = (\Sigma_1, \alpha_1)$ und $Z_2 = (\Sigma_2, \alpha_2)$
mit $\Sigma_1 = \{V_1 \rightarrow M\}$ und $\Sigma_2 = \{V_2 \rightarrow M\}$
Menge gemeinsamer Variablen

$$V_0 = V_1 \cap V_2$$

nicht notwendigerweise leer.
Produkt der Zustandsräume: $\Sigma = \{V \rightarrow M\}$ mit $\Delta = \Delta_1 \parallel \Delta_2$ (parallele Komp.)
wobei $V = V_1 \cup V_2$

Die Zustandsübergangsfunktion $\Delta: \Sigma \rightarrow \wp(\Sigma)$ wird definiert als **Interleaving:**

$\Delta(\sigma) = \{\tau : (\tau|V_1 \in \Delta_1(\sigma|V_1) \wedge \tau|(V \backslash V_1) = \sigma\ (V \backslash V_1)\ )$ *** Z_1 mit Δ_1 auf V_1
$\vee\ (\tau|V_2 \in \Delta_2(\sigma|V_2) \wedge \tau|(V \backslash V_2) = \sigma\ (V \backslash V_2)\ )\ \}$ *** Z_2 mit Δ_2 auf V_2

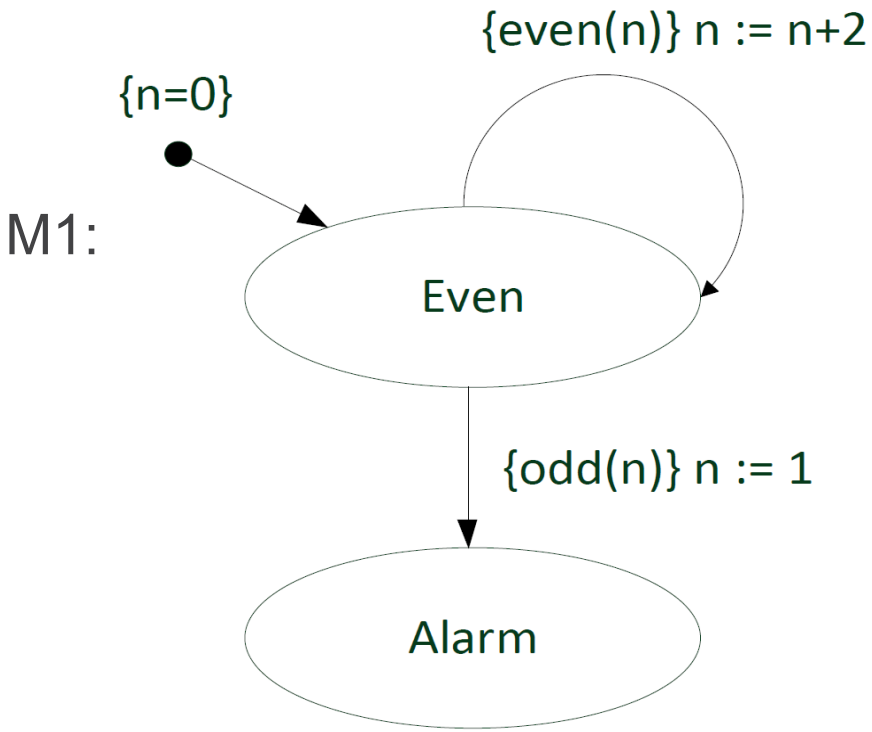
Beachte: synchrone Nebenläufigkeit nicht sinnvoll definierbar

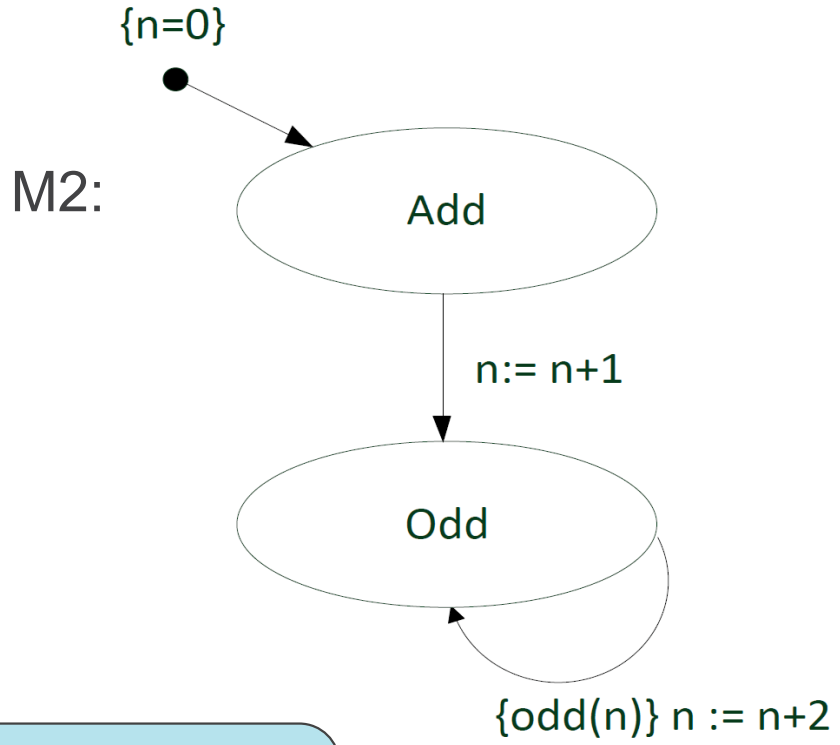
Komposition bei nicht-disjunkten ZR

- Menge erreichbarer Zustände hängt ab von:
  - **Granularität** der Transitionsschritte
  - Anzahl der Teilschritte
  - Zwischenzustände
- Wir erhalten neue Abläufe
  - Invarianten werden nicht notwendigerweise erhalten

Beispiel:
Granularität der Parallellen Komposition
mit gemeinsamen Variablen

Beispiel: Granularität – erreichbare Zustände



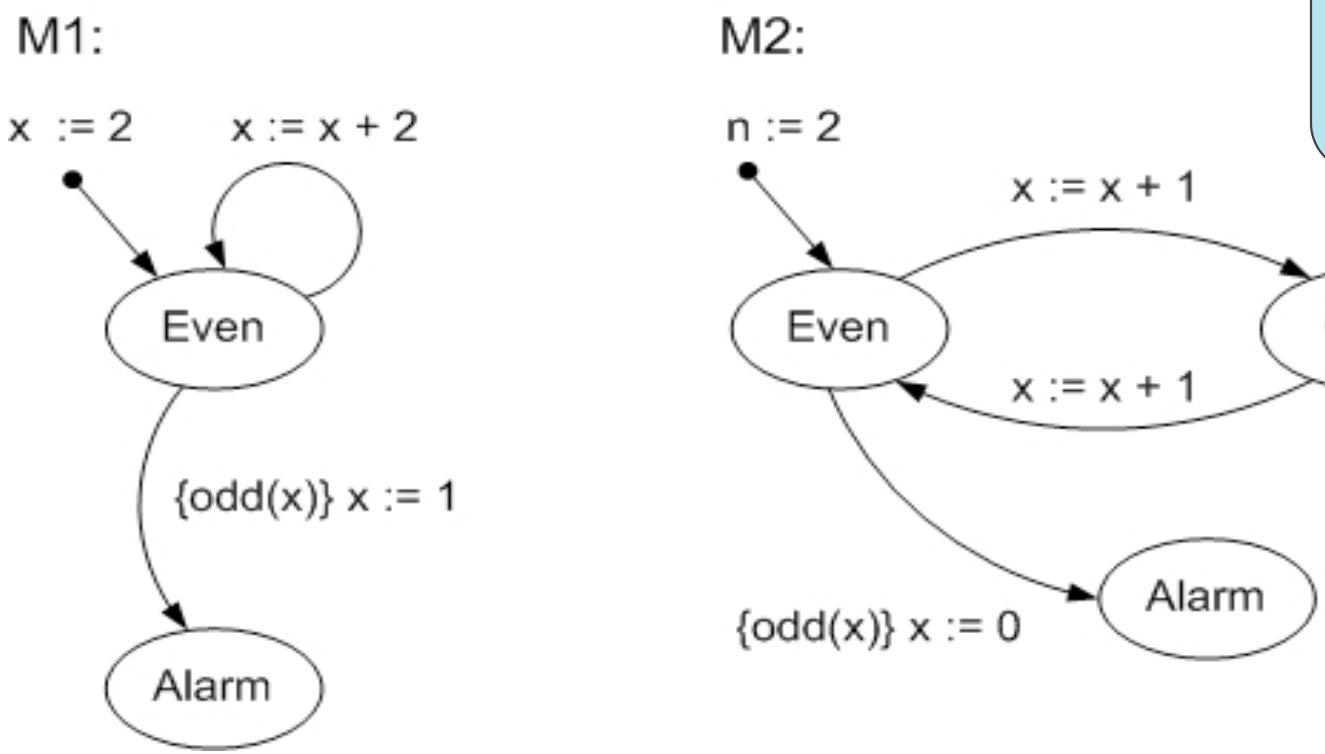


Zustand *Alarm* (n=1):
- für die Maschine M1||M2 erreichbar
- für die Maschinen M1 und M2 unerreichbar

Stabile Prädikate

- Voraussetzung: durch Interleaving komponierte Zustandsmaschine
  - **Stabilität:** stabil für beide Zustandsmaschine --> stabil für komponierte Zustandsmaschine
  - Nachweis entsprechend modular
  - **Achtung:** gilt nicht für Invarianten

Beispiel: Stabile Prädikate



M1 führt zwei Schritte der Maschine M2 in einem Schritt durch

Invariante *even(x):*
- *gilt für* M1 gilt die,
- gilt nicht für M2

M1 sowie M2: kann kein Alarm ausgelöst werden.

M1||M2: Alarm ist in der Menge der erreichbaren Zustände.

Abschließende Bemerkungen: Parallele Komposition von Zustandsmaschinen

- **Echte Parallelität**:

  Synchron: Beide Maschinen machen gleichzeitig einen Übergang

  Asynchron: Beide Maschinen machen unabhängig Übergänge - manchmal gleichzeitig

  - Vorteile: Modelliert echte Parallelität
  - Nachteile: Gemeinsame Variable schwierig zu handhaben: Im asynchronen Fall Fairness erforderlich um sicherzustellen, dass jede Maschine immer wieder zum Zug kommt

- **Interleaving**:

  Übergänge werden verschränkt hintereinander ausgeführt

  - Vorteile: Gemeinsame Variable problemlos zu handhaben
  - Nachteile: Parallelität wird nicht explizit modelliert: Fairness erforderlich um sicherzustellen, dass jede Maschine immer wieder zum Zug kommt