
Ein Ansatz für die verifikationsgerechte Verhaltensmodellierung für die semi-formale Verifikation von Mixed-Signal-Schaltungen

Martin Freibothe

Promotionsverteidigung

Überblick

- **Verifikationsansätze**
- **Verifikationsablauf für Mixed-Signal-Schaltungen**
- **Verifikationsgerechte Verhaltensmodellierung**
 - **Zeitdiskretisierung**
 - **Quantisierung**
 - **Endlicher Automat**
- **Anwendungsbeispiel**
- **Zusammenfassung und Ausblick**

Verifikationsansätze

- Ziel: Finden von Entwurfsfehlern

- Industrielle Praxis: vorrangig simulationsbasierte Validation

- Verifikation von Analog und Mixed-Signal-Schaltungen: aktueller Gegenstand der Forschung

Überprüfung eines digitalen Schaltungsentwurfs

Formale Verifikation

- Äquivalenzvergleich
- Modellüberprüfung, Eigenschaftsprüfung
- Formale Methoden, Korrektheitsnachweis
- Nachweis der Abwesenheit von Entwurfsfehlern

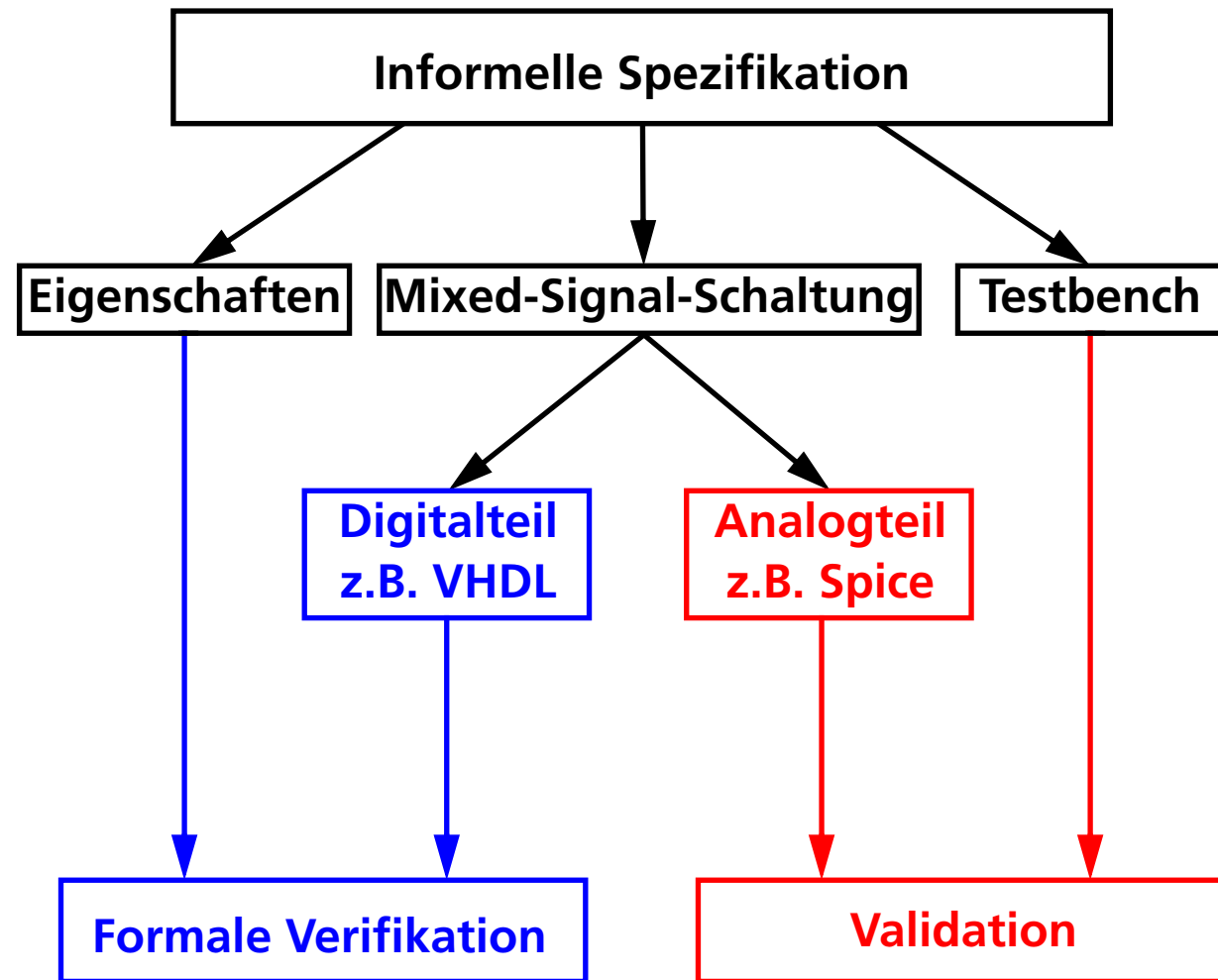
Validation

- Simulationsbasiert
- Testbenches
- Größere Schaltungen handhabbar
- Unzureichende funktionale Überdeckung

Verifikationsablauf für Mixed-Signal-Schaltungen (1)

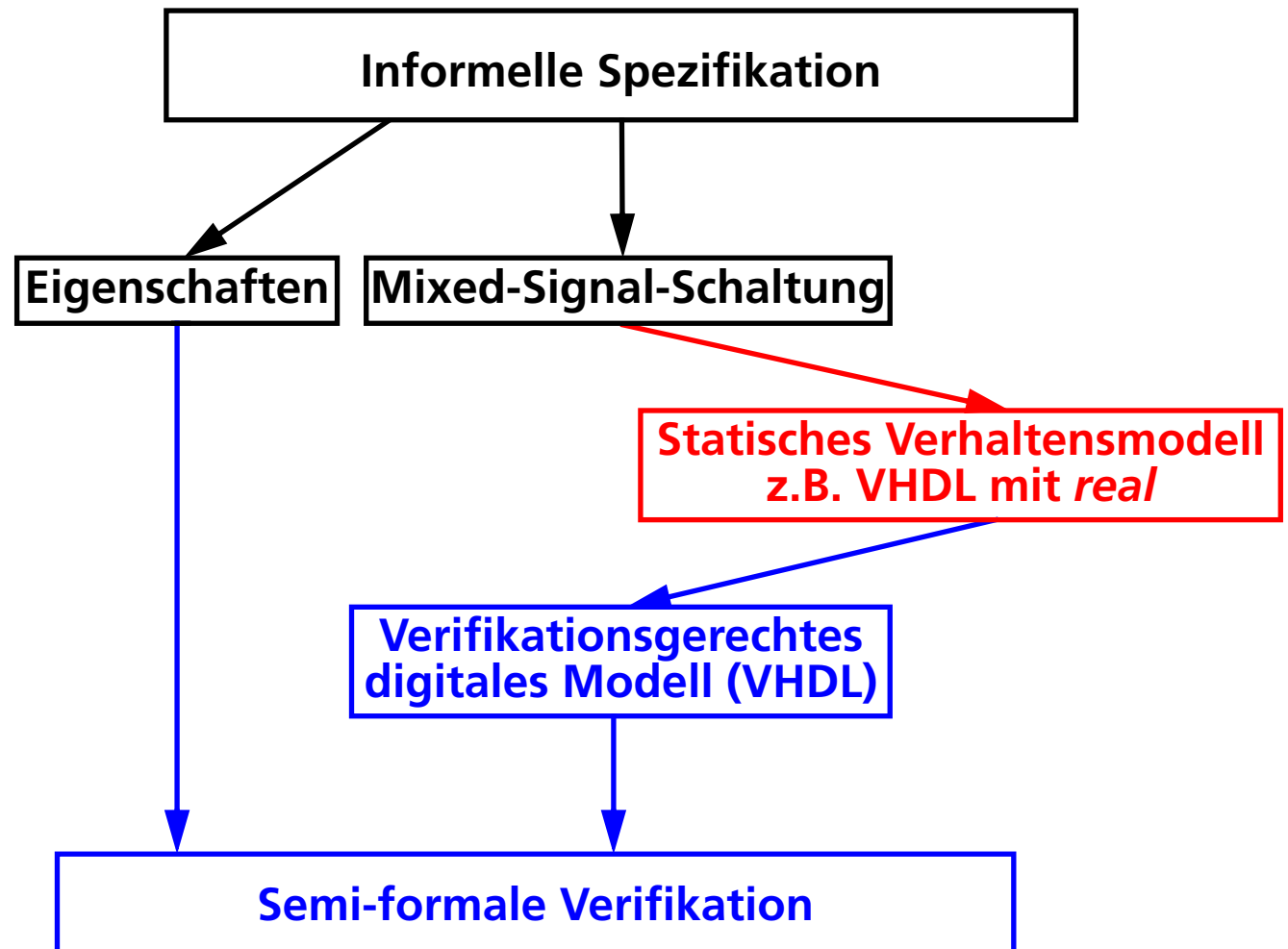
Mixed-Signal-Schaltung

- simulationsbasierte Validation
- Trennung der Analog- und Digitalteile
- Zusammenschaltung wird simuliert
- hoher manueller Aufwand
- funktionale Abdeckung unzureichend



Verifikationsablauf für Mixed-Signal-Schaltungen (2)

- **Digitales Verhaltensmodell**
 - **Quantisierung über endlichen Intervallen**
 - Quantisierungsfehler
 - Überläufe
- **Semi-formaler Ansatz**
- **Offene Probleme**
 - Herleitung des Verhaltensmodells
 - Überprüfung des dynamischen Verhaltens

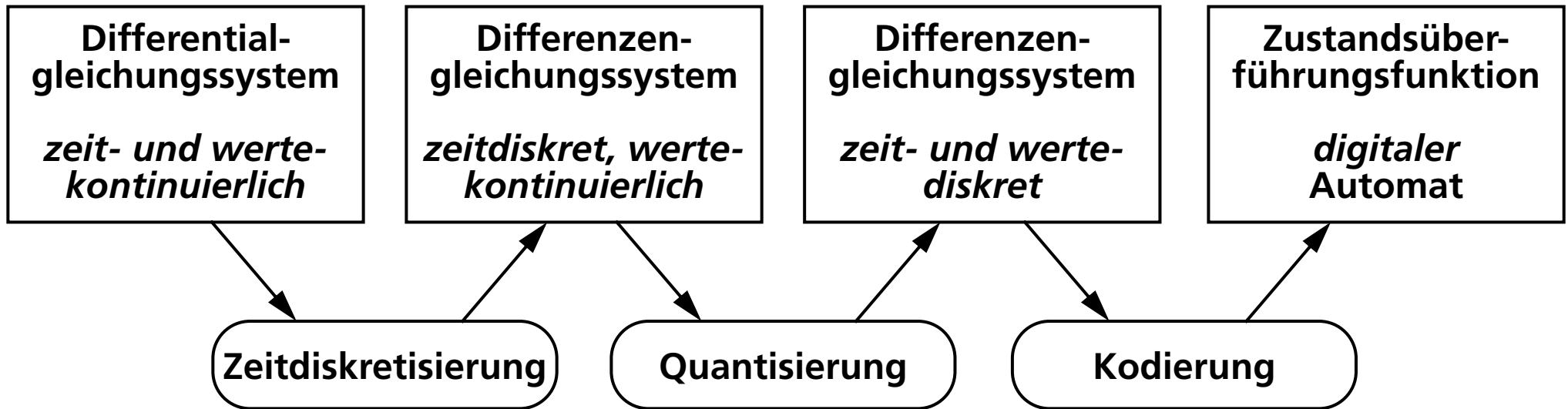


Digitale Verhaltensmodellierung analoger Komponenten

- **Ziele**
 - Modellierung des dynamischen analogen Verhaltens mit Hilfe von endlichen Automaten
 - Erstellen verifikationsgerechter Verhaltensmodelle (Implementierung z.B. in VHDL)
 - Verwendung etablierter Werkzeuge der digitalen formalen Verifikation zur Entwurfsüberprüfung

- **Probleme**
 - analoge Komponenten sind in Form elektrischer Netzwerke gegeben (nichtlineare Algebro-Differentialgleichungssysteme)
 - im Allgemeinen gibt es keine analytische Lösung im Vergleich zur Verhaltensbeschreibung digitaler Schaltungen auf der RT-Ebene

Modellierungsablauf (1)

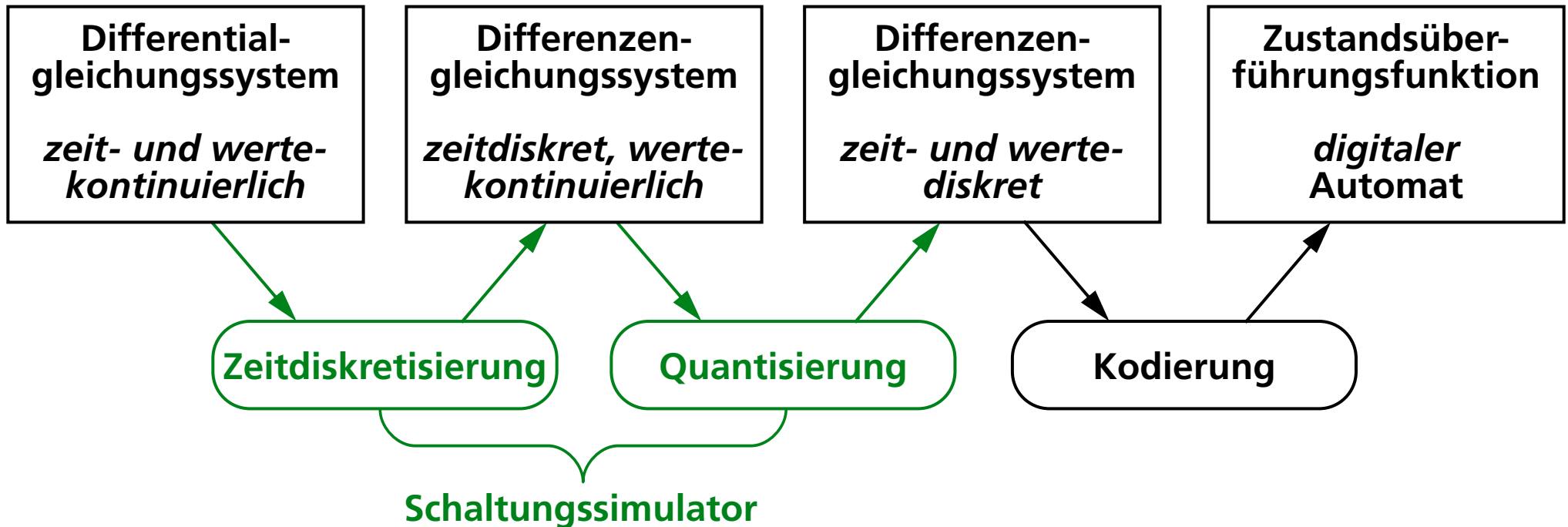


- Integrationsformeln
- Genauigkeit
- Komplexität

- Auflösung
- Genauigkeit
- Komplexität

- Endliche Wertebereiche
- Berechnung der Wertetabelle
- Komplexität

Modellierungsablauf (2) simulationsbasiert

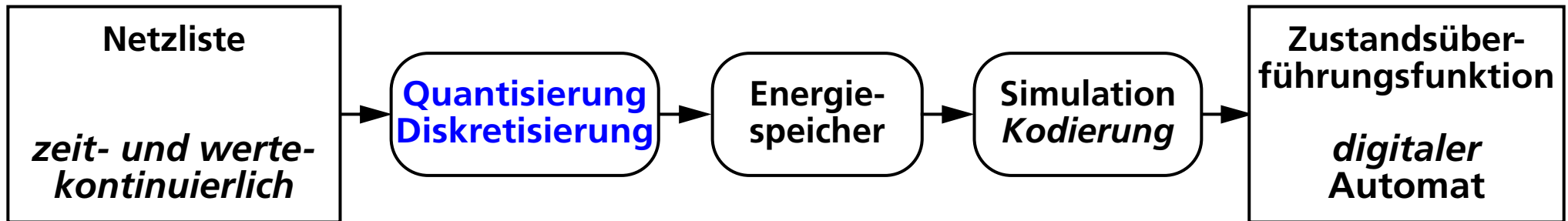


- Integrationsformeln
- Genauigkeit
- Komplexität

- Auflösung
- Genauigkeit
- Komplexität

- Endliche Wertebereiche
- Berechnung der Wertetabelle
- Komplexität

Modellierungsablauf (3) simulationsbasiert



- **Quantisierung**

- Quantisierung der analogen Werte über endlichen Intervallen; Parameter: Auflösung, Intervallgrenzen
- endliche Mengen für Eingabe-, Ausgabe- und Zustandssymbole → endlicher Automat
- Kompromiß zwischen Genauigkeit und Komplexität des Modells und der anschließenden Verifikation

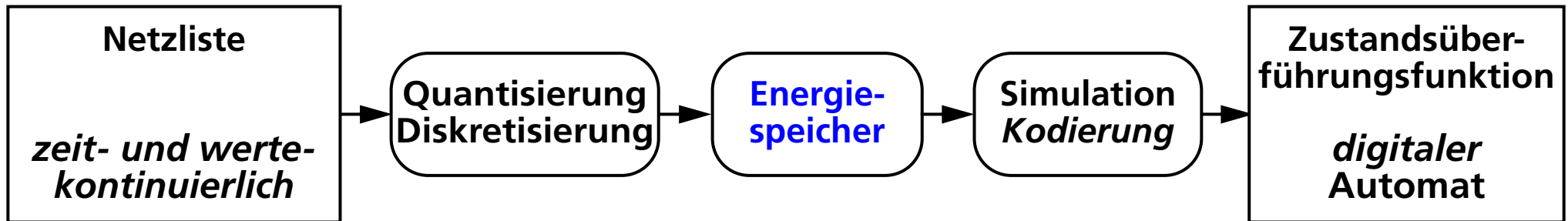
- **Zeitdiskretisierung**

- Festlegung des Simulationsintervalls $[0; t_s]$
- Zeitschritt t_s entspricht der Taktperiode des aufgestellten digitalen Automaten (*Ein-Schritt-Automat*)

Quantisierungsparameter und Simulationsintervall

- **Quantisierung**
 - Genauigkeit des Modells
 - Komplexität des Modells
 - Komplexität der anschließenden Verifikation
 - Auflösung der Quantisierung und Simulationsintervall $[0; t_s]$ müssen geeignet gewählt werden
 - Neues Quantisierungsintervall muß nach Simulation des Zeitschrittes t_s erreicht werden
- **Endliche Wertebereiche**
 - endliche Mengen für Eingabe-, Ausgabe- und Zustandssymbole des endlichen Automaten
 - Komplexität des Modells
 - Abdeckung größerer Wertebereiche mit mehreren Modellen — »Teile und Herrsche« —

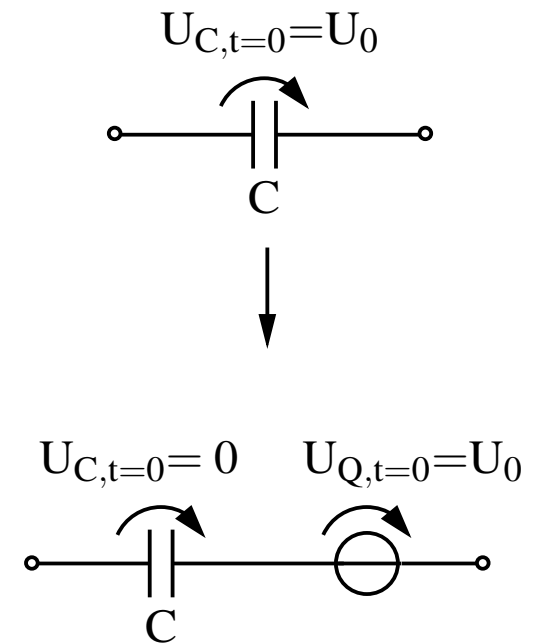
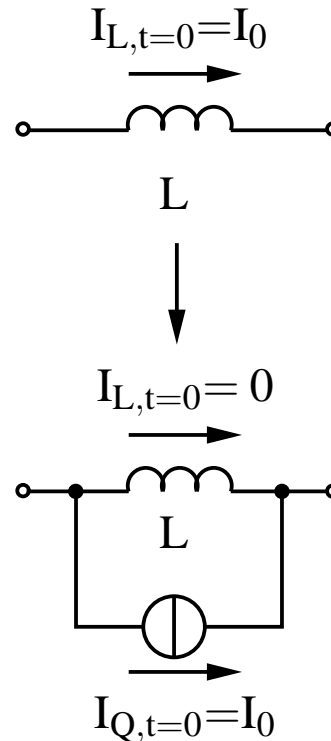
Modellierungsablauf (4) simulationsbasiert



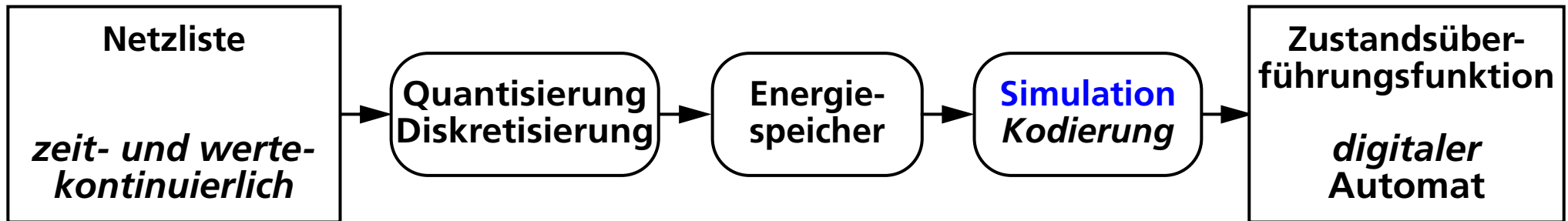
- **Energiespeichernde Elemente**
 - im Allgemeinen durch Kapazitäten und Induktivitäten gegeben
 - Komplexität des Ansatzes steigt exponentiell mit der Anzahl der energiespeichernden Elemente, die als Zustandsvariablen des Automaten modelliert werden
 - Bestimmung »relevanter« energiespeichernder Elemente
 - »relevante« Energiespeicher werden durch Ersatzschaltungen für die Transientensimulation ersetzt

Anpassungen der Netzliste für Energiespeicher

- »Relevante« energiespeichernde Elemente werden in der Netzliste ersetzt
- Äquivalente Ersetzung in Bezug auf das Klemmverhalten
- Anfangsbedingungen für die Simulation werden mit Hilfe der eingefügten Quellen I_Q und U_Q gesetzt



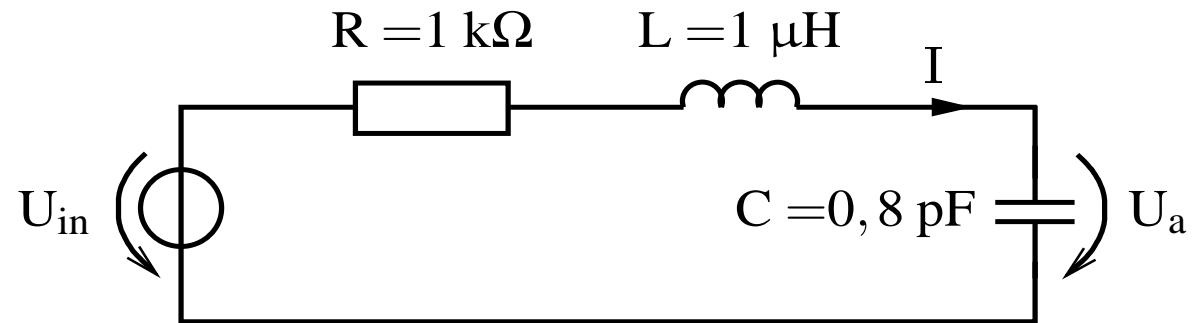
Modellierungsablauf (5) simulationsbasiert



- **Simulation**
 - ausschöpfende Simulation bzgl. der Quantisierung
 - Rundung der Simulationsergebnisse
 - jede mögliche Kombination von Belegungen der Eingänge und Anfangszustände
- **Vorteile der Verwendung eines Simulators**
 - Genauigkeit, Simulationszeit, numerische Stabilität
 - üblicherweise in industriellen Entwurfsabläufen verwendet
 - analoge Komponenten liegen i. A. als Netzlisten vor

Illustrierendes Beispiel

- RLC-Schaltung
 - Schaltbild



- Modellierungsparameter

- Zwei Speichervariablen:
 - Spannung über dem Kondensator $U_c = U_a$
 - Strom durch die Spule I
- Eingangsvariable U_{in}
- Ausgangsvariable U_a

Aufstellen des Automaten

- **Modellierungsparameter für die Beispielschaltung**
 - **Spannungen:**
-6,4 V . . . 6,0 V Auflösung: 0,4 V / bit
 - **Ströme:**
-2,0 mA . . . 1,875 mA Auflösung: 0,125 mA / bit
 - **Repräsentation mit Bitvektoren der Breite 5, Länge des Simulationsintervalls $[0; t_s] = 0,7 \text{ ns}$**

- **Zustandsüberföhrungsfunktion als Wertetabelle**
 - **Transientensimulation der angepaßten Netzliste für jede Kombination der Belegung der Eingänge und Anfangszustände**
 - **Berücksichtigung von technisch nicht sinnvollen Kombinationen**
 - **vollständige Verhaltensbeschreibung bezüglich der gewählten Quantisierung und Diskretisierung**

Wertetabelle der Funktionen λ und δ

- Endliche Anzahl von Eingangsfolgen, hier:

0,0 V \rightarrow 0,0 V

0,0 V \rightarrow 3,2 V

3,2 V \rightarrow 0,0 V

3,2 V \rightarrow 3,2 V

$U_{in}(k)$	$U_{in}(k+1)$	$I_S(k)$	$U_S(k)$	$I_S(k+1)$	$U_a(k+1)$
0,0	0,0	-1,250	0,8	-0,750	0,0
0,0	0,0	-1,250	1,2	-1,000	0,4
0,0	0,0	-1,375	2,8	-1,750	1,6
...
3,2	3,2	-1,375	2,4	-0,125	1,6

- **Bestimmung der Wertetabelle durch Simulation**

- **Beschreibung des Ein-Schritt-Automaten**
 - **Ausgabefunktion λ und**
 - **Zustandsübergangsfunktion δ**

Implementierung eines Ein-Schritt-Automaten

- **Synthetisierbare Untermenge von VHDL**
- **Voltage** und **current** mit Hilfe von Bitvektoren implementiert
- **Expliziter Takteingang**

```
entity rlc is
```

```
  port (
    -- Takteingang
    clk : in std_ulogic;

    -- Ein- und Ausgangs-
    -- spannung
    U_in : in voltage;
    U_a : out voltage);
```

```
end rlc;
```

Martin Freibothe

```
architecture behavior of rlc is
begin
```

```
  p1: process (clk, U_in)
    variable U_intern : voltage;
    variable I_intern : current;
```

```
  begin
```

```
    if clk'event and clk = '1' then
```

```
      if( U_in = voltage( 3.2 ) ) then
```

```
        if( I_intern = current( 0.001500 ) ) then
          if(U_intern = voltage( -2.400000 ) ) then
            I_intern := current( 0.001875 ) ;
            U_intern := voltage( -0.800000 ) ;
```

```
          end if;
```

```
        end if;
```

```
      end if;
```

```
      ...
```

```
    end process;
```

```
end behavior;
```

Promotionsverteidigung



Nachweis formaler Eigenschaften

- **Eigenschaft für das Einschwingen der Ausgangsspannung U_a**

$$\begin{aligned} (U_{in_0} &= Volt_{0_0} \vee U_{in_0} = Volt_{3_2}) \wedge \\ (U_{in_1} &= Volt_{0_0} \vee U_{in_1} = Volt_{3_2}) \wedge \\ (U_{in_{14}} &= U_{in_{13}} = U_{in_{12}} = \dots = U_{in_1}) \\ &\rightarrow \end{aligned}$$

- **Bounded interval model checking**

- endliches Zeitintervall
- hier: $[t; t+14]$

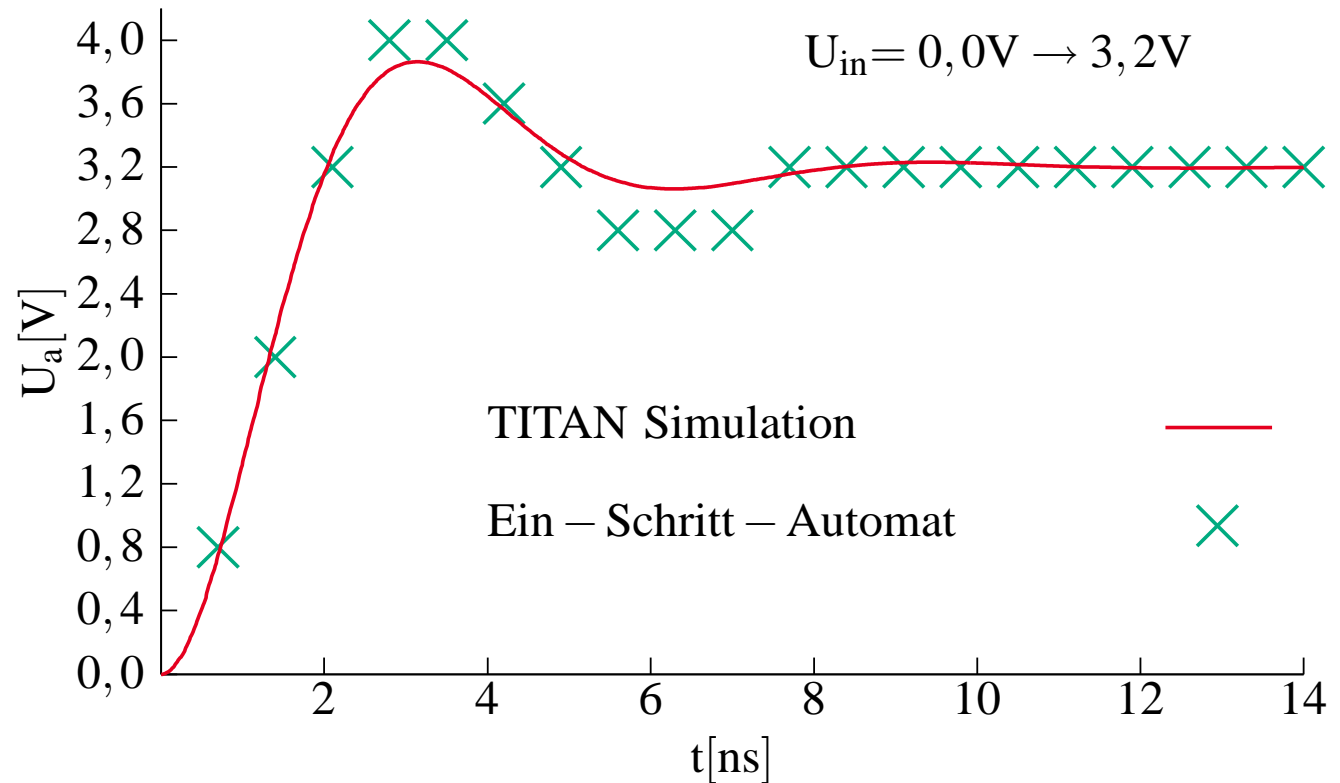
$$\begin{aligned} (U_{a_{14}} &= U_{a_{13}} \wedge \\ I_{intern_{14}} &= I_{intern_{13}}) \end{aligned}$$

- **Ressourcen**
 - 10 Minuten CPU
 - 750 MB RAM

- **Unabhängig vom Anfangszustand sind die Einschwingvorgänge nach 12 Takten abgeklungen**
- **12 Taktzyklen des Ein-Schritt-Automaten entsprechen einer Zeit von 8,4 ns**

Illustration der Approximationsfehler

- **Modell repräsentiert approximiertes Verhalten**
 - Diskretisierungsfehler
 - Quantisierungsfehler
- **Quantisierungsfehler muß in Eigenschaften berücksichtigt werden**



→ **Semi-formaler Verifikationsansatz**

TITAN von Qimonda/Infineon Technologies AG

N-Schritt-Automat

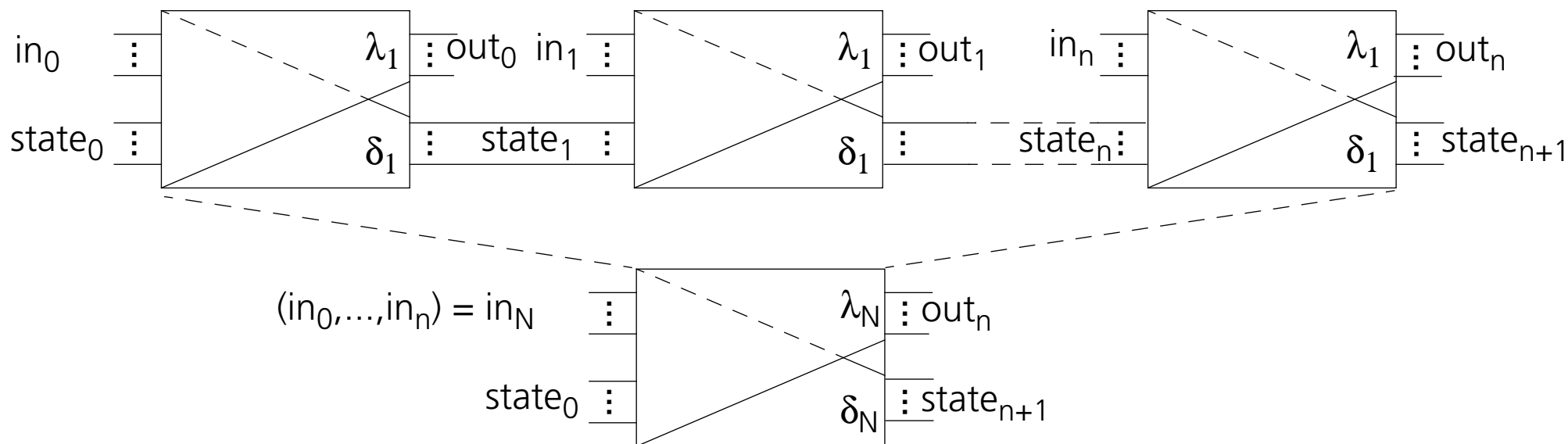
- **Ziel**

- digitales Modell für die Repräsentation des analogen Verhaltens in Bezug auf die digitale Umgebung
- digitales Modell ersetzt in der Zusammenschaltung das analoge Verhaltensmodell
 - **Digitales Verhaltensmodell der zugrundeliegenden Mixed-Signal-Schaltung**

- **Ansatz**

- weiterer Abstraktionsschritt ausgehend von einem Ein-Schritt-Automaten
- Aufstellen eines N-Schritt-Automaten
- Taktperiode P ist ganzzahliges Vielfaches der Länge des Simulationsintervalls t_s mit $P = n \cdot t_s$

Aufstellen eines N-Schritt-Automaten



- **N-Schritt-Automat**
 - **Ausgabefunktion:** λ_N
 - **Zustandsüberföhrungsfunktion:** δ_N
- $out_n = \lambda_1 (\delta_1 (\dots \delta_1 (\delta_1 (state_0, in_0) , in_1) , \dots , in_{n-1}) , in_n)$
 $= \lambda_N (state_0, (in_0, \dots, in_n))$
- $state_{n+1} = \delta_1 (\dots \delta_1 (\delta_1 (state_0, in_0) , in_1) , \dots , in_n)$
 $= \delta_N (state_0, (in_0, \dots, in_n))$

Verwendung der digitalen Verhaltensmodelle

- **Ein-Schritt-Automat**
 - Nachweis praktisch relevanter Eigenschaften über das dynamische Verhalten analoger Komponenten
 - Komposition von Ein-Schritt-Automaten
 - weitere Abstraktion durch das Aufstellen von N-Schritt-Automaten

- **N-Schritt-Automat**
 - Zusammenschaltung einer analogen Komponente mit der digitalen Umgebung
 - Komposition verschiedener N-Schritt-Automaten

Verbesserter Verifikationsablauf

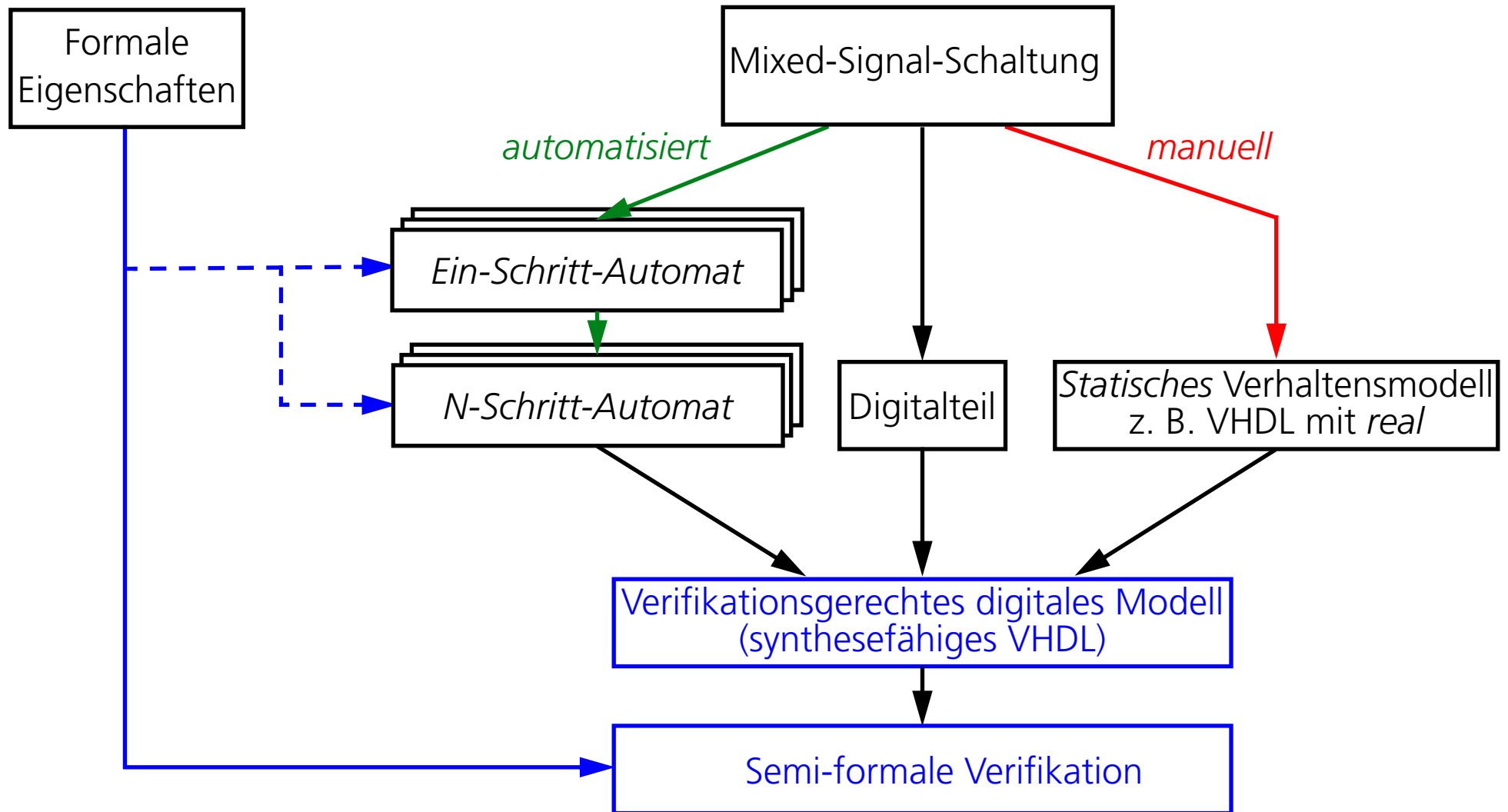
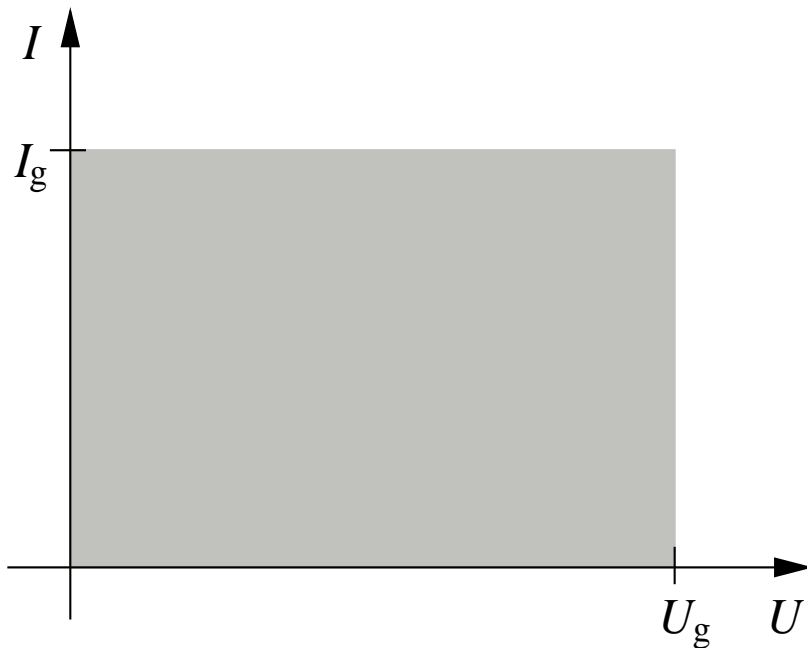


Illustration der Gültigkeit der Verhaltensmodelle

- Kontinuierliches Verhaltensmodell:
(elektrisches Netzwerk)



- Digitales Verhaltensmodell:
(endlicher Automat)

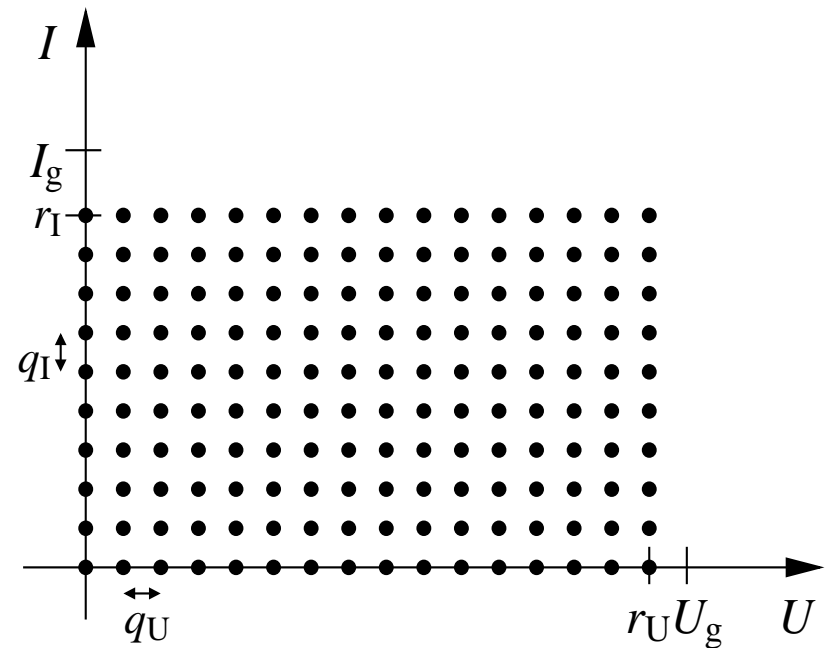
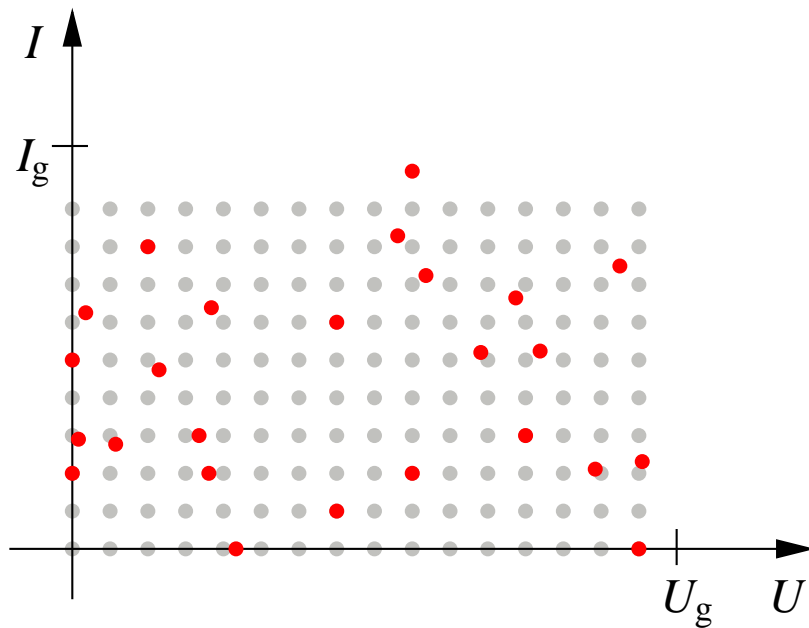
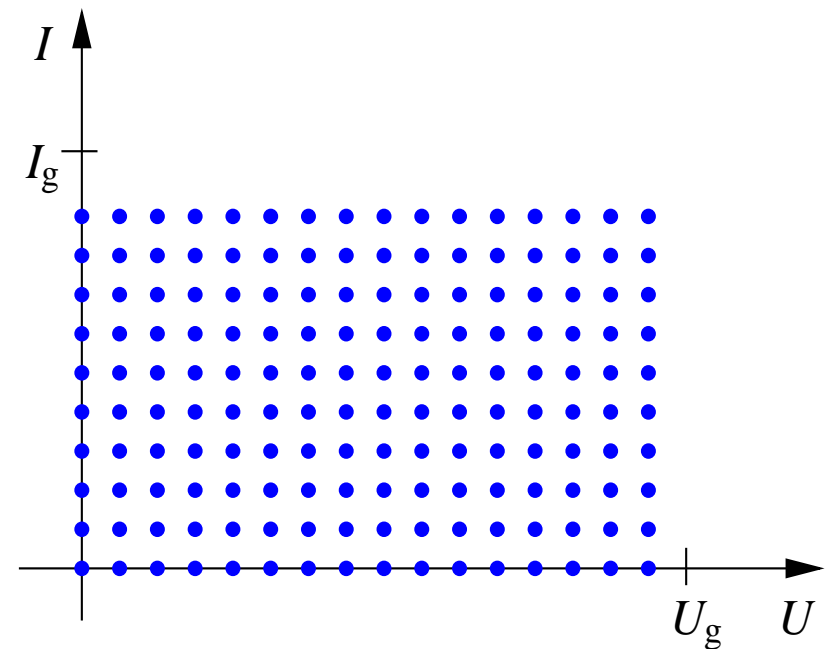


Illustration der funktionalen Abdeckung

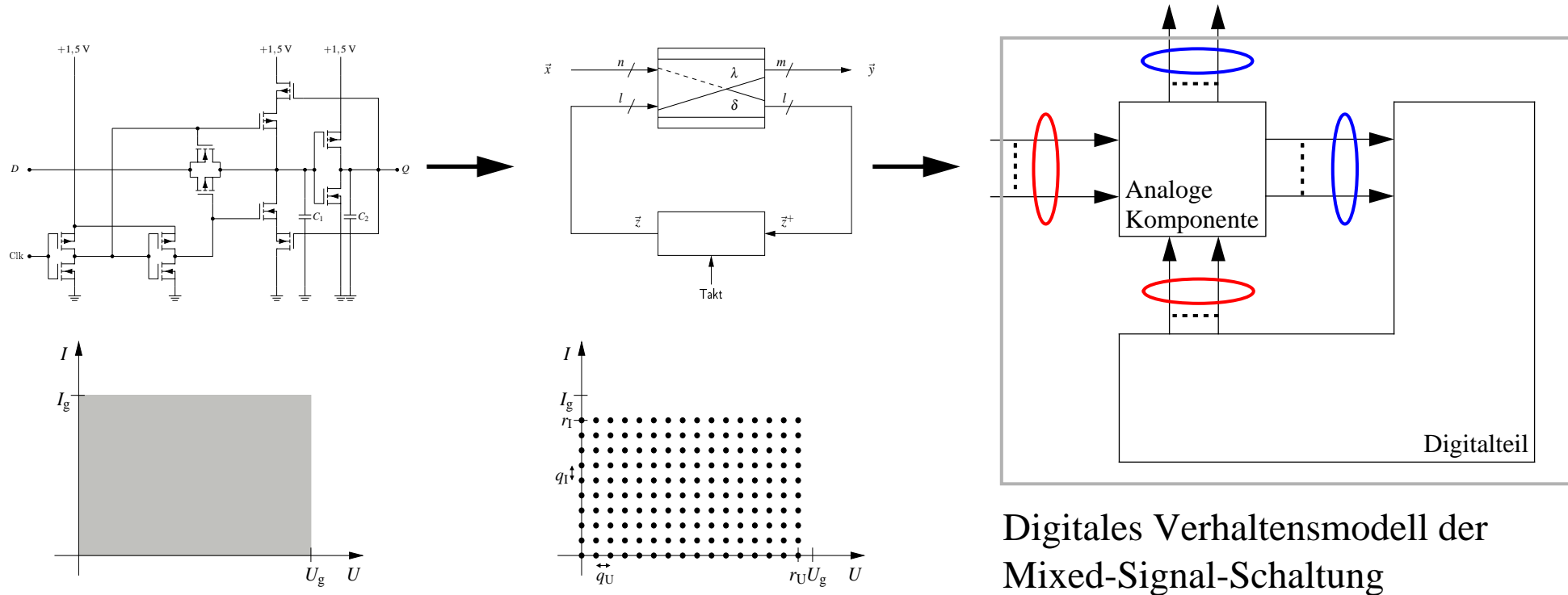
- Kontinuierliches Verhaltensmodell:
(elektrisches Netzwerk)



- Digitales Verhaltensmodell:
(endlicher Automat)



Einbetten einer analogen Komponente

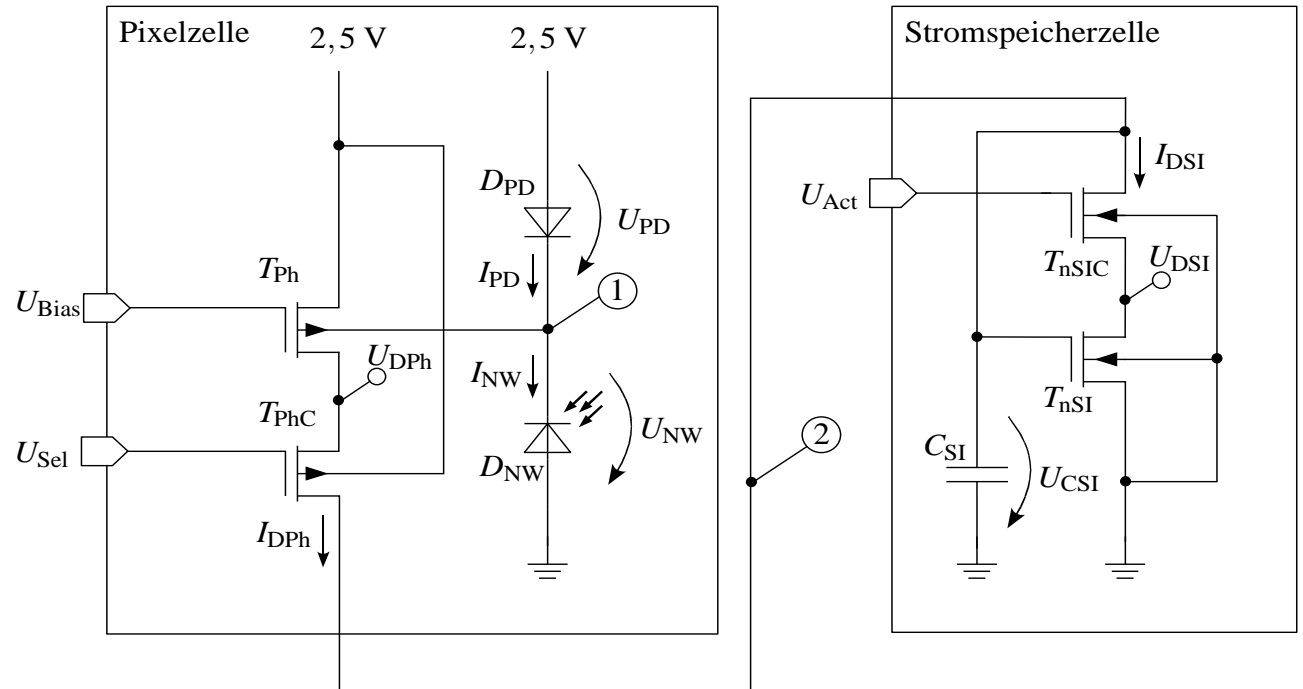


Verwendung des digitalen Verhaltensmodells:

- Konstanten für Grenzen der gültigen Wertebereiche
- **Einschränken** der Eingangsbelegungen
- **Überprüfen** der Ausgangsbelegungen

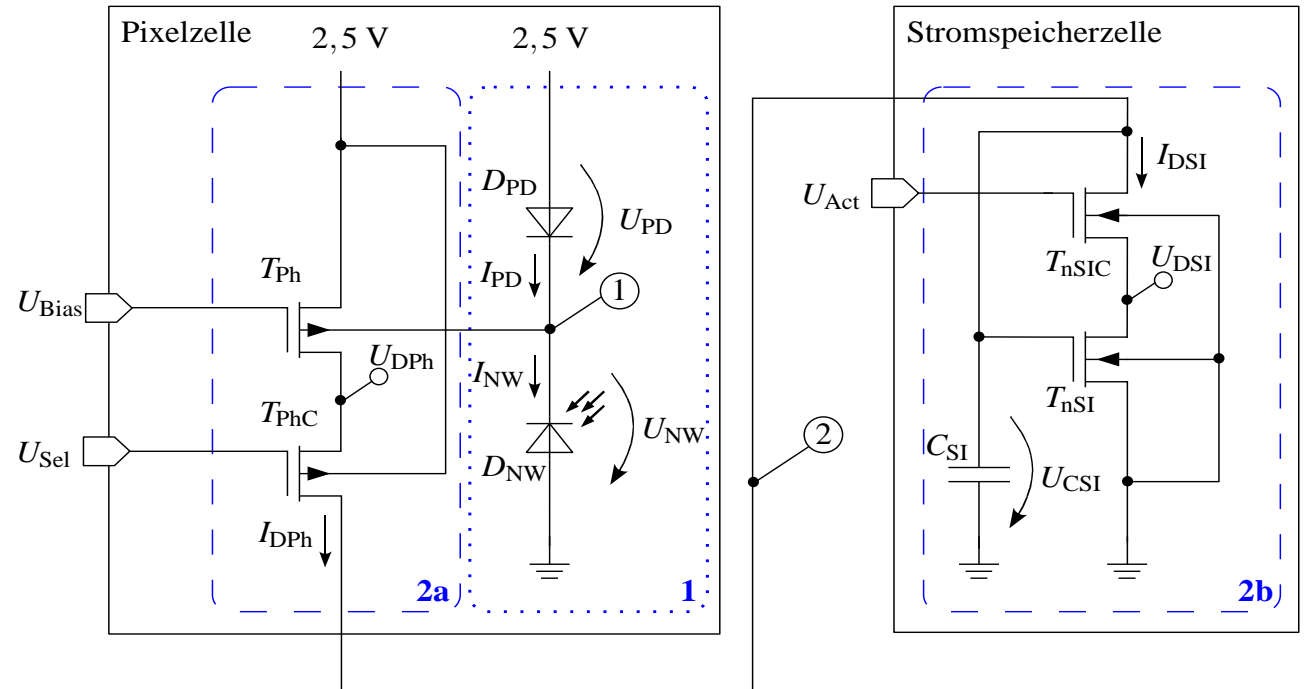
Anwendungsbeispiel: Pixelzelle (1)

- **Pixelzelle mit Stromspeicherzelle**
- **Zwei Modi:**
 - integrierend
 - logarithmierend
- **Eingangsvariable:**
 - Photostrom I_{NW}
- **Ausgangsvariable:**
 - Spannung über C_{SI}



Anwendungsbeispiel: Pixelzelle (2)

- **Partitionierung**
- **Knoten ①** wird als rückwirkungsfrei angenommen
- **Zwei Ein-Schritt-Automaten**
 - Partition 1
 - Partition 2(ab)



Anwendungsbeispiel: Pixelzelle (3)

- **Überprüfte Eigenschaften**
 - **Überschwingen: Nachweis, daß die Ausgangsspannung innerhalb der angegebenen Intervalle liegt**
 - **nachgeschaltete digitale Komponenten im Signalpfad übernehmen richtige Werte**
 - **Ausgangsspannung entspricht der Spezifikation**

Zusammenfassung

- **Anwendung formaler Verifikationsmethoden für eine größere Klasse von Mixed-Signal-Schaltungen**
- **Verwendung eines Simulators zur numerischen Lösung der Verhaltensbeschreibung in Form von DGLs**
- **Nachweis von praxisrelevanten Eigenschaften über das dynamische Zeitverhalten analoger Komponenten**
- **Bereitstellung von digitalen Verhaltensmodellen analoger Komponenten, die kontinuierliche Modelle in der Zusammenschaltung ersetzen**
- **Exponentieller Aufwand der Modellierung in der Anzahl der Eingangs- und Zustandsvariablen sowie der Anzahl der Quantisierungsstufen**
- **Verifikationsverfahren ist aufgrund der Modellierung trotz der Anwendung formaler Techniken semi-formal**

Offene Probleme und Ausblick

- **Beschreibung der Klasse von analogen Komponenten und Mixed-Signal-Schaltungen für die das Verfahren anwendbar ist**
- **Optimierung der Parameter für die Anwendung der Modellierung (Zeitdiskretisierung, Quantisierung)**
- **Verfahren/Ansätze für die Bestimmung »relevanter« energiespeichernder Elemente**
- **Modellierung basiert auf Nominalsimulationen, Parameterschwankungen werden nicht berücksichtigt**
- **Betrachtung des Verhaltens im Zeitbereich – Verhalten im Frequenzbereich (Frequenzgang) wird nicht überprüft**
- **Anwendung des Verfahrens ausgehend von weiteren Verhaltensbeschreibungen wie z.B. VHDL-AMS**