

# ReCoNets – Entwurfsmethodik für eingebettete Systeme bestehend aus kleinen Netzwerken hardwarerekonfigurierbarer Knoten und -verbindungen

Ch. Haubelt, D. Koch, T. Streichert, J. Teich

Hardware-Software-Co-Design

Universität Erlangen-Nürnberg

{haubelt, dirk.koch, streichert, teich}@informatik.uni-erlangen.de

# Gliederung

---

- Motivation
- Ziele
- Arbeitsprogramm:
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - Implementierung
- Ausblick

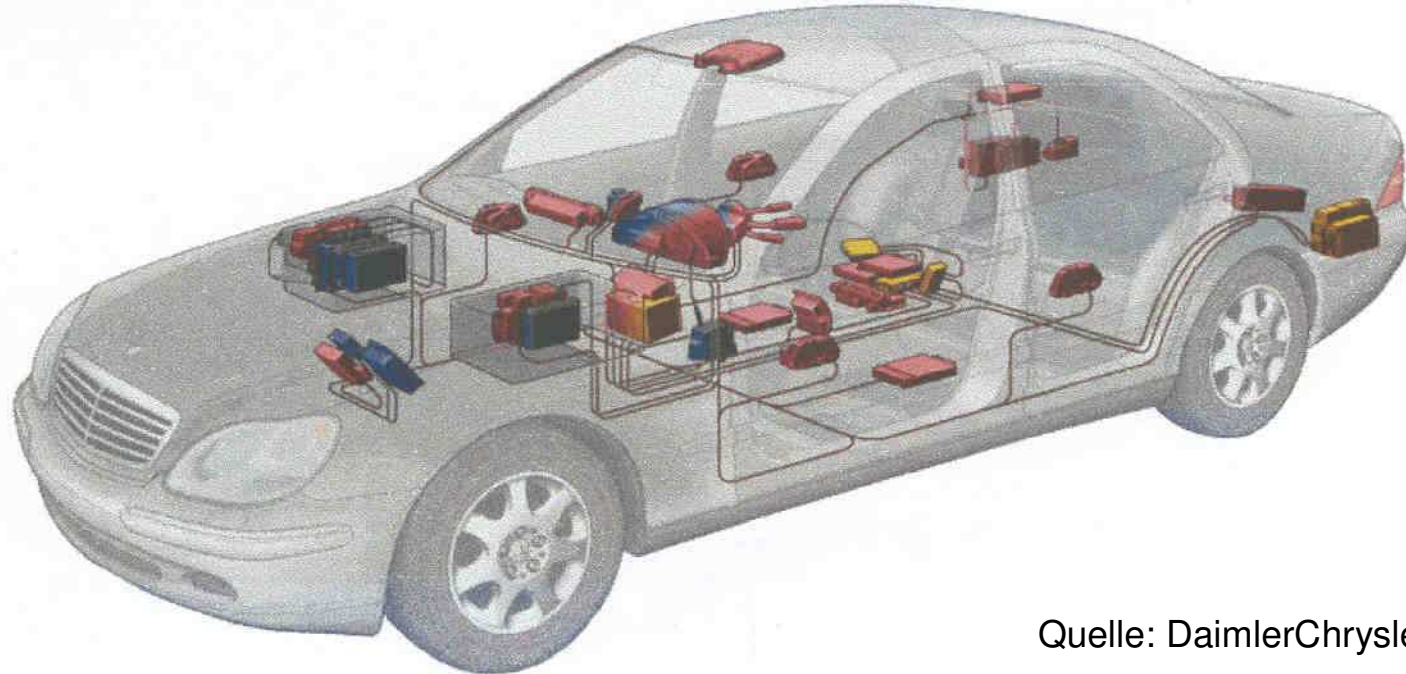
# Gliederung

---

- Motivation
- Ziele
- Arbeitsprogramm:
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - Implementierung
- Ausblick

# Beispiel: Automobil

---

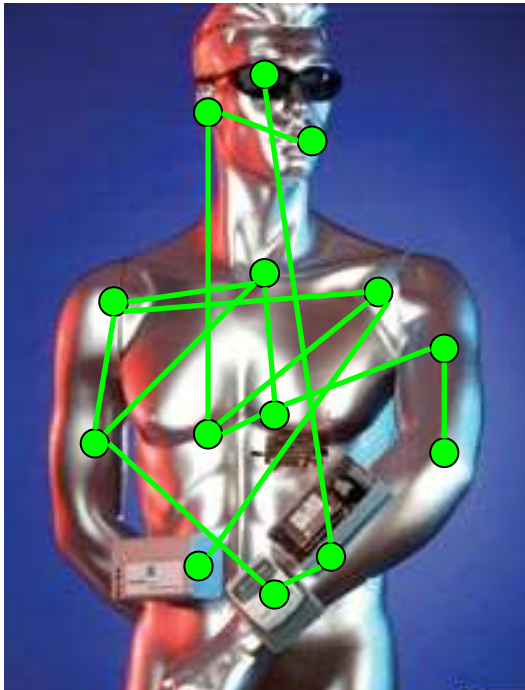


Quelle: DaimlerChrysler AG

Beispiele: Antriebsstrang, Komfort, Anzeigesysteme,  
Telematik, Diagnose, Diebstahlschutz, zukünftige  
Fahrfunktionen

# Beispiel: Body-Area-Netzwerke

---



Quelle: Fraunhofer Gesellschaft

- geographisch feste Knoten
- hohe Spezialisiertheit
- schlechte Ressourcenauslastung
- kaum Flexibilität
- geringe Ausfallsicherheit
- ...

# Gliederung

---

- Motivation
- Ziele
- Arbeitsprogramm:
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - Implementierung
- Ausblick

# Ziele (1/4)

---

Entwurfsmethodik für eingebettete Systeme bestehend aus **kleinen Netzwerken** hardwarerekonfigurierbarer Knoten und -verbindungen

- < 100 Knoten
- Zum jedem Zeitpunkt ist jedem Knoten die funktionsfähige Netztopologie und die Prozessverteilung bekannt

# Ziele (2/4)

---

Entwurfsmethodik für **eingebettete Systeme** bestehend aus kleinen Netzwerken hardwarerekonfigurierbarer Knoten und -verbindungen

- Spezialisiertes System
- Dynamische Nutzungsszenarien erfordern die Anpassbarkeit des Systems und der einzelnen Knoten an unterschiedliche Anforderungen
  - Energieaspekte
  - Rechenleistung
  - ...
- Nur durch Hardwarerekonfiguration zu erreichen



# Ziele (3/4)

---

Entwurfsmethodik für eingebettete Systeme bestehend aus kleinen Netzwerken **hardwarerekonfigurierbarer Knoten und -verbindungen**

- Knoten sind elektrisch miteinander verbunden (Dies steht im Gegensatz zu Modellen und Verfahren von Funk- und optischen Netzen)
- Knoten sind an geographisch festen Orten gebunden
- Flexibilität des Systems wird allein durch Hardwarerekonfiguration erreicht

# Ziele (4/4)

---

Entwurfsmethodik für eingebettete Systeme bestehend  
aus kleinen Netzwerken hardwarerekonfigurierbarer  
Knoten und -verbindungen

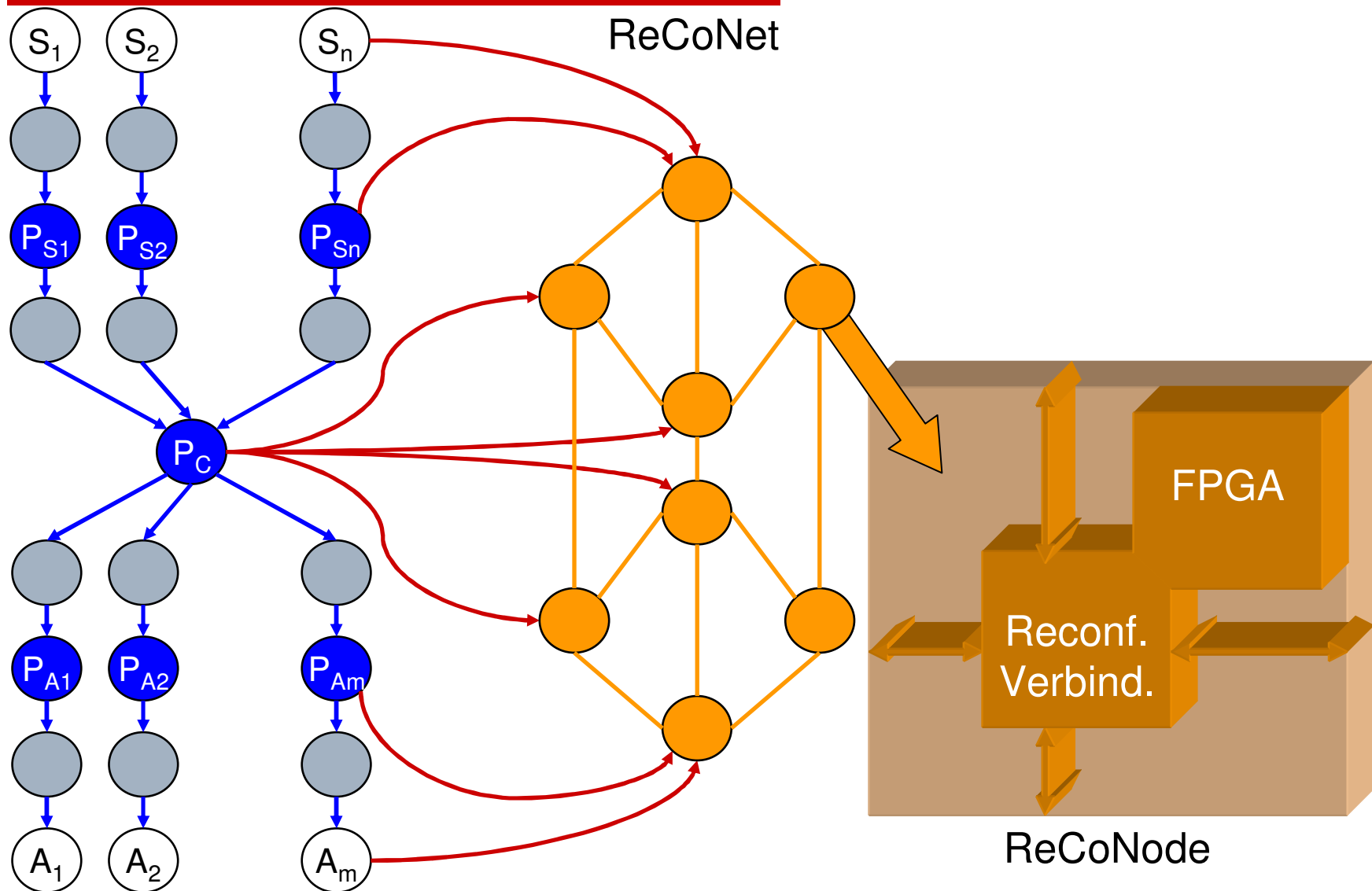
- Es fehlt an Entwurfsmethodik für diese Systeme!

# Gliederung

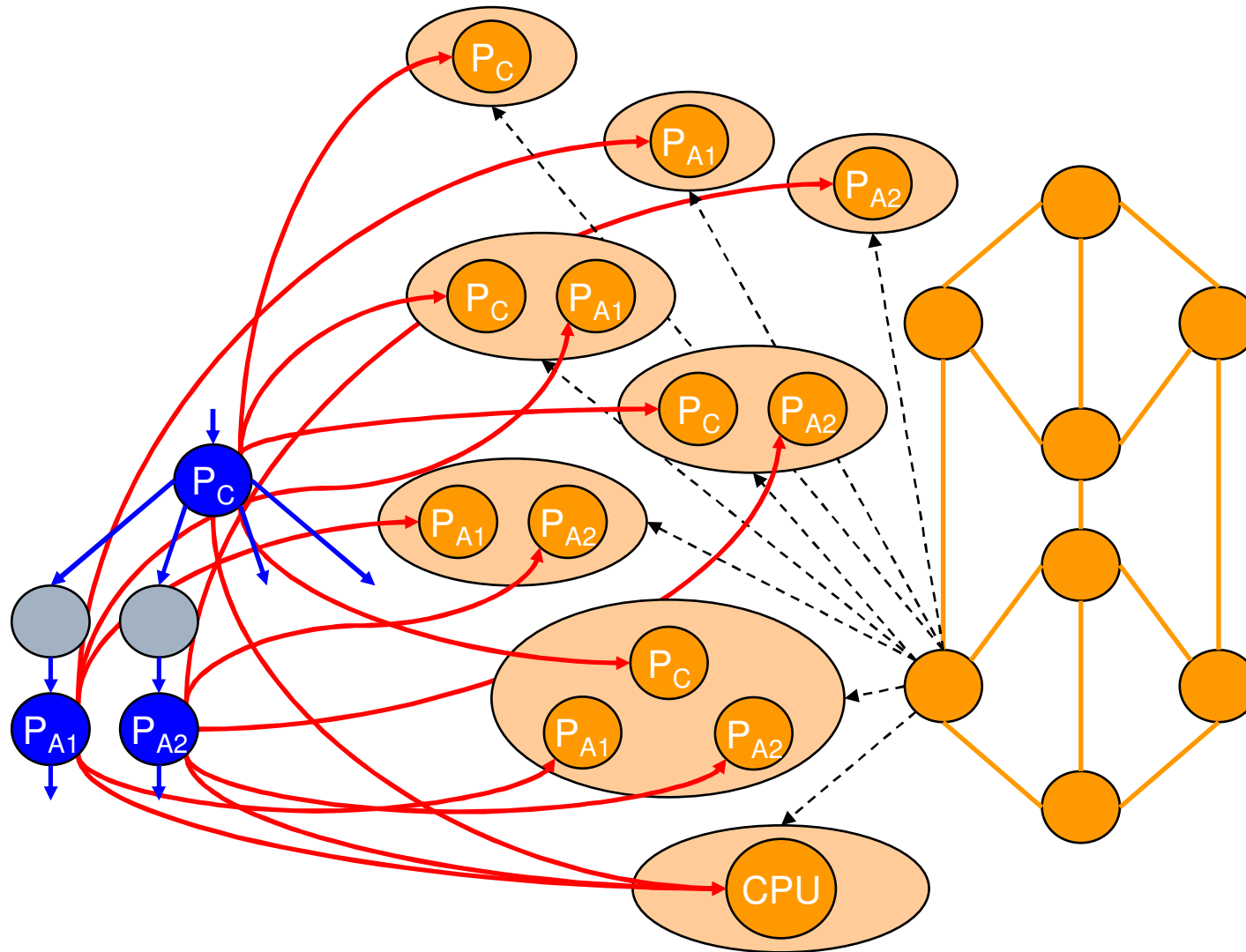
---

- Motivation
- Ziele
- **Arbeitsprogramm:**
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - Implementierung
- Ausblick

# Sensor-Controller-Aktuator-Kette



# Rekonfigurierbare Hardware



# Zusammenfassung: Modellierung

---

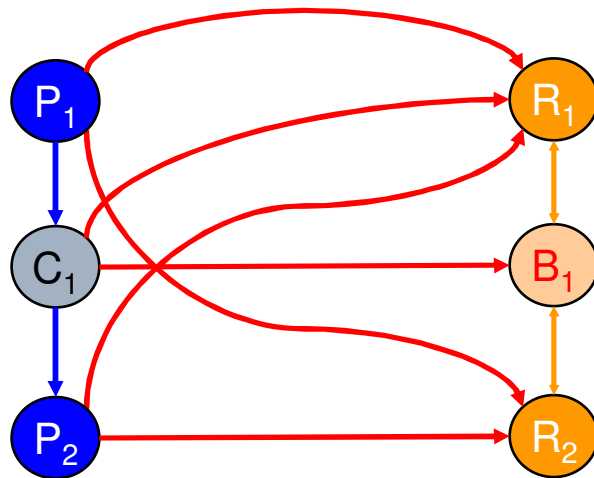
- Modellierung und Architekturabbildung von sog. Sensor-Controller-Aktuator-Ketten
- Modellierung und Spezifikation zeitvarianter Anwendungen und Architekturen:
  - Zeitvariante Allokation  $\alpha(t)$ :  
Welche Komponenten (Konfigurationen) stehen zum Zeitpunkt  $t$  zur Verfügung?
  - Zeitvariante Bindung  $\beta(t)$ :  
Auf welcher Komponente wird ein Prozess zum Zeitpunkt  $t$  ausgeführt?

# Gliederung

---

- Motivation
- Ziele
- **Arbeitsprogramm:**
  - Modellierung
  - **Analyse**
  - Synthese und Optimierung
  - Implementierung
- Ausblick

# Ausfallsicherheit und Fehlertoleranz



1-Fehler:

R <sub>1</sub>	B <sub>1</sub>	R <sub>2</sub>
✓	✓	✓

2-Fehler:

R <sub>1</sub> B <sub>1</sub>	R <sub>1</sub> R <sub>2</sub>	B <sub>1</sub> R <sub>2</sub>
✗	✗	✓



# Ausfallsicherheit

---

- Erste Vorarbeiten:
- SAT-basierte Verfahren zur System-Synthese
- Auch einsetzbar zur Analyse der Ausfallsicherheit zur Entwurfszeit → Topologieoptimierung

# Detektion von Ressourcendefekten

---

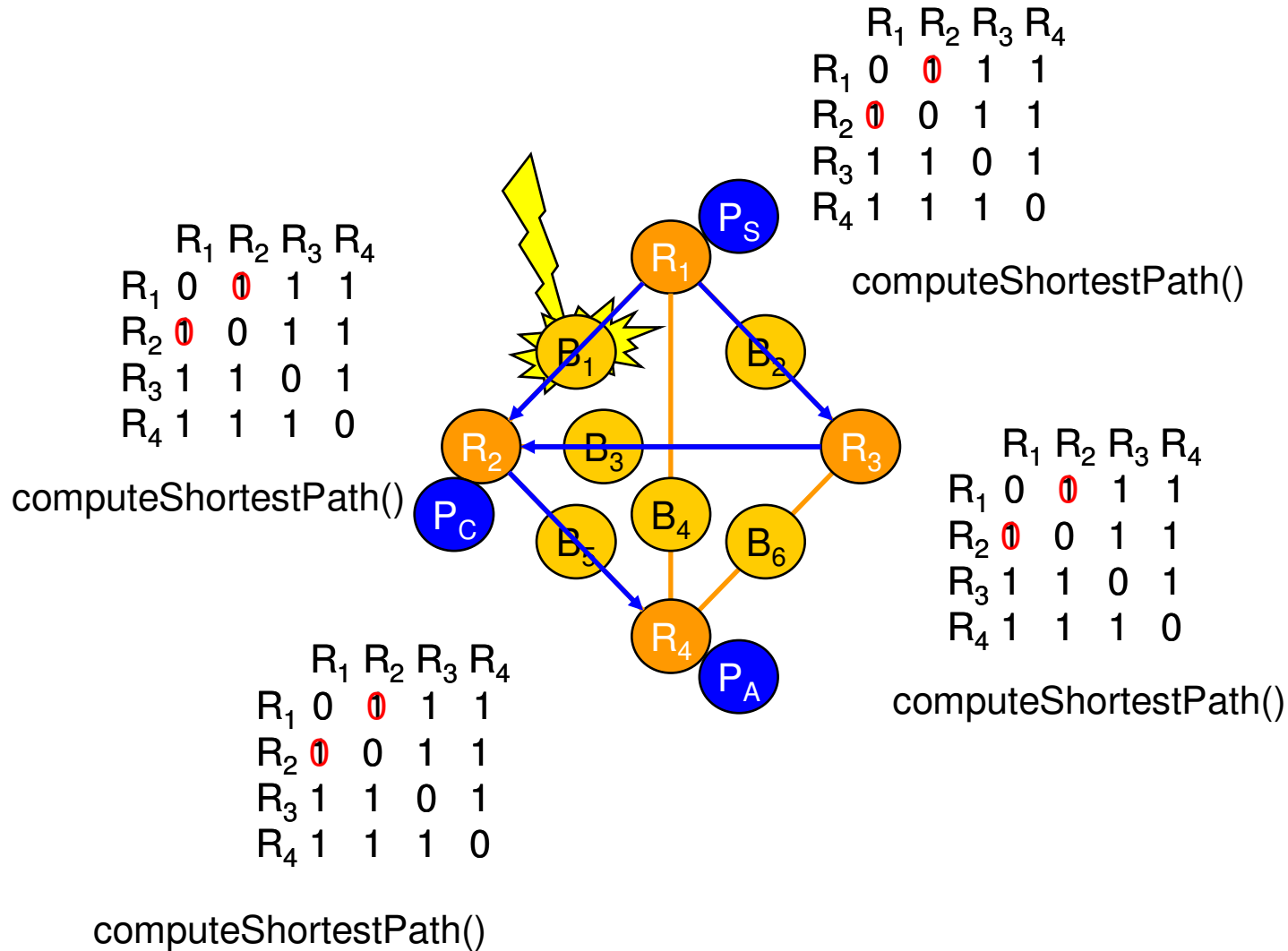
- Erkennung von Fehlern auf Leitungen z.B. durch paketorientierte Kommunikation
- Einsatz bekannter Techniken zur Kanalcodierung und Fehlerkorrektur als Hilfswerkzeuge möglich
- Knotendetektion kann z.B. von Leitungsdetektion abgeleitet werden

# Gliederung

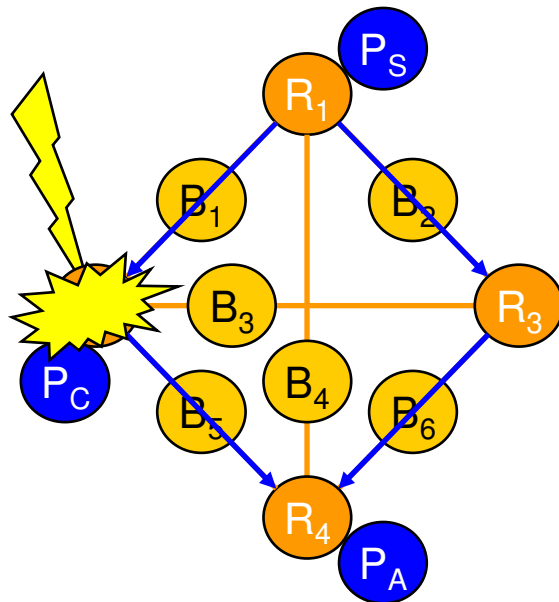
---

- Motivation
- Ziele
- **Arbeitsprogramm:**
  - Modellierung
  - Analyse
  - **Synthese und Optimierung**
  - Implementierung
- Ausblick

# Rerouting

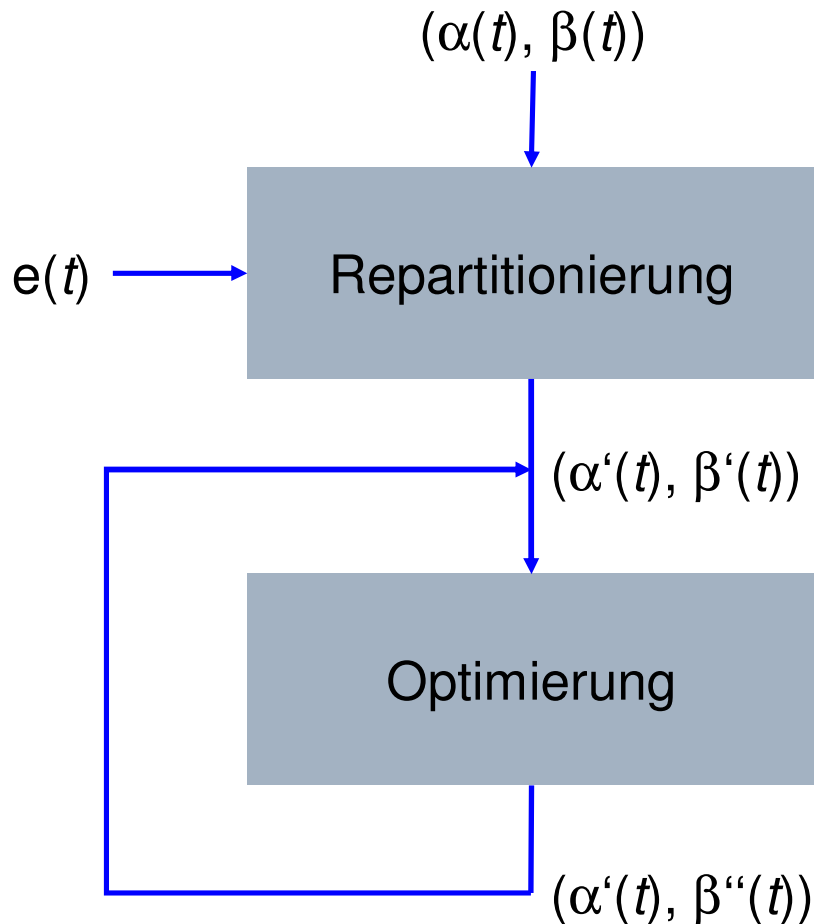


# Online-HW/SW-Partitionierung (1/2)



- Aber: Wohin soll ein Prozess migriert werden?
  - Minimierung der Anzahl der zu rekonfigurierenden Knoten?
  - Ausgeglichene Ressourcenauslastung?
  - Minimierung der mittleren Kommunikationsstrecke?
  - ...
- Soll ein Prozess nach der Migration in HW oder SW ausgeführt werden?
- Managementfunktion zentral oder dezentral realisieren?

# Online-HW/SW-Partitionierung (2/2)



- Ereignis  $e(t)$  bewirkt Allokationsänderung
- Repartitionierung bindet "kritische" Prozesse neu
- Optimierungsphase verteilt Prozesse neu
- Mögliche Zielgrößen:
  - Flächenbedarf für Hardware-Prozesse je Knoten
  - Anzahl der Prozesse pro Knoten

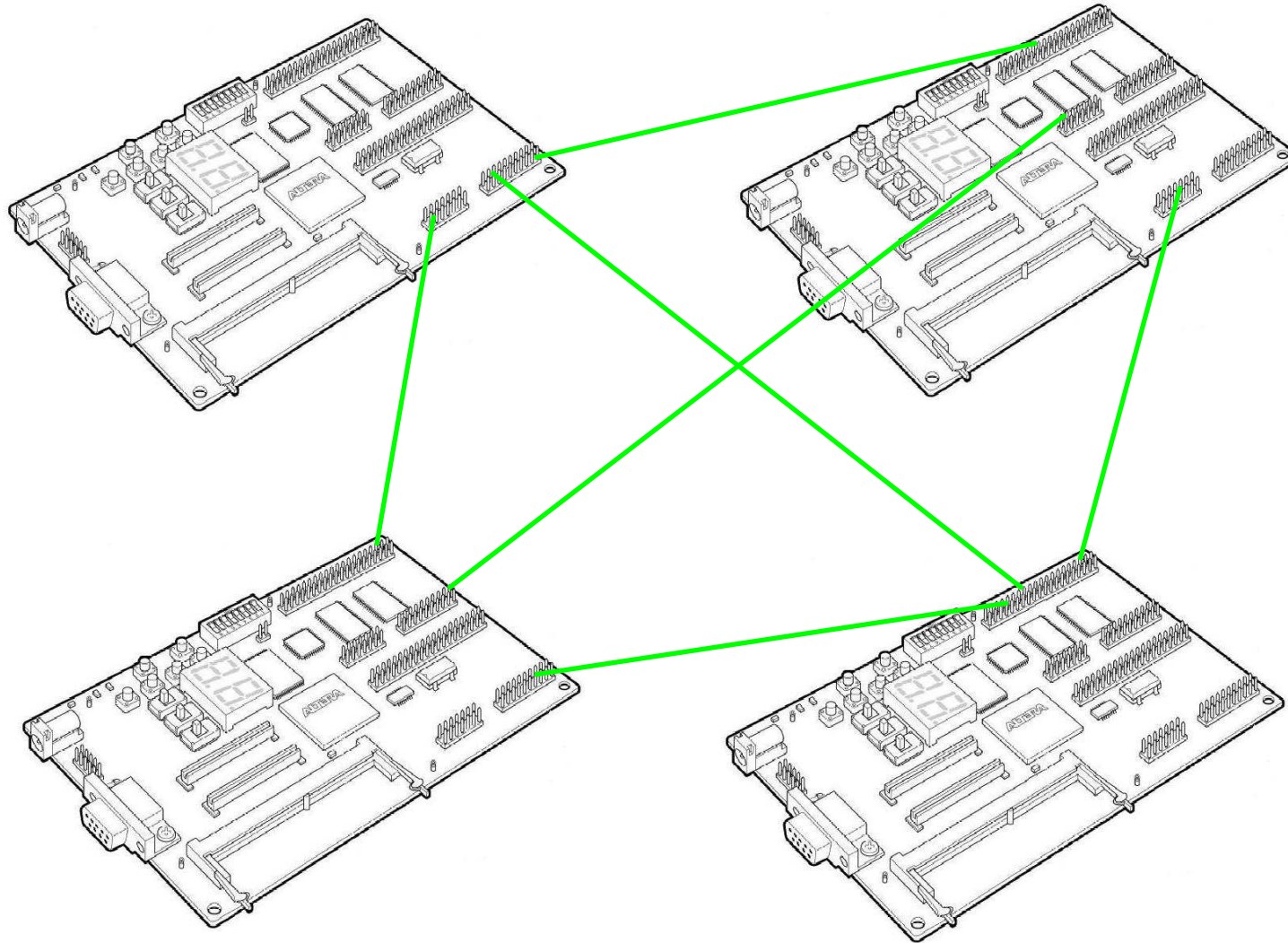
# Gliederung

---

- Motivation
- Ziele
- **Arbeitsprogramm:**
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - **Implementierung**
- Ausblick

# Implementierung

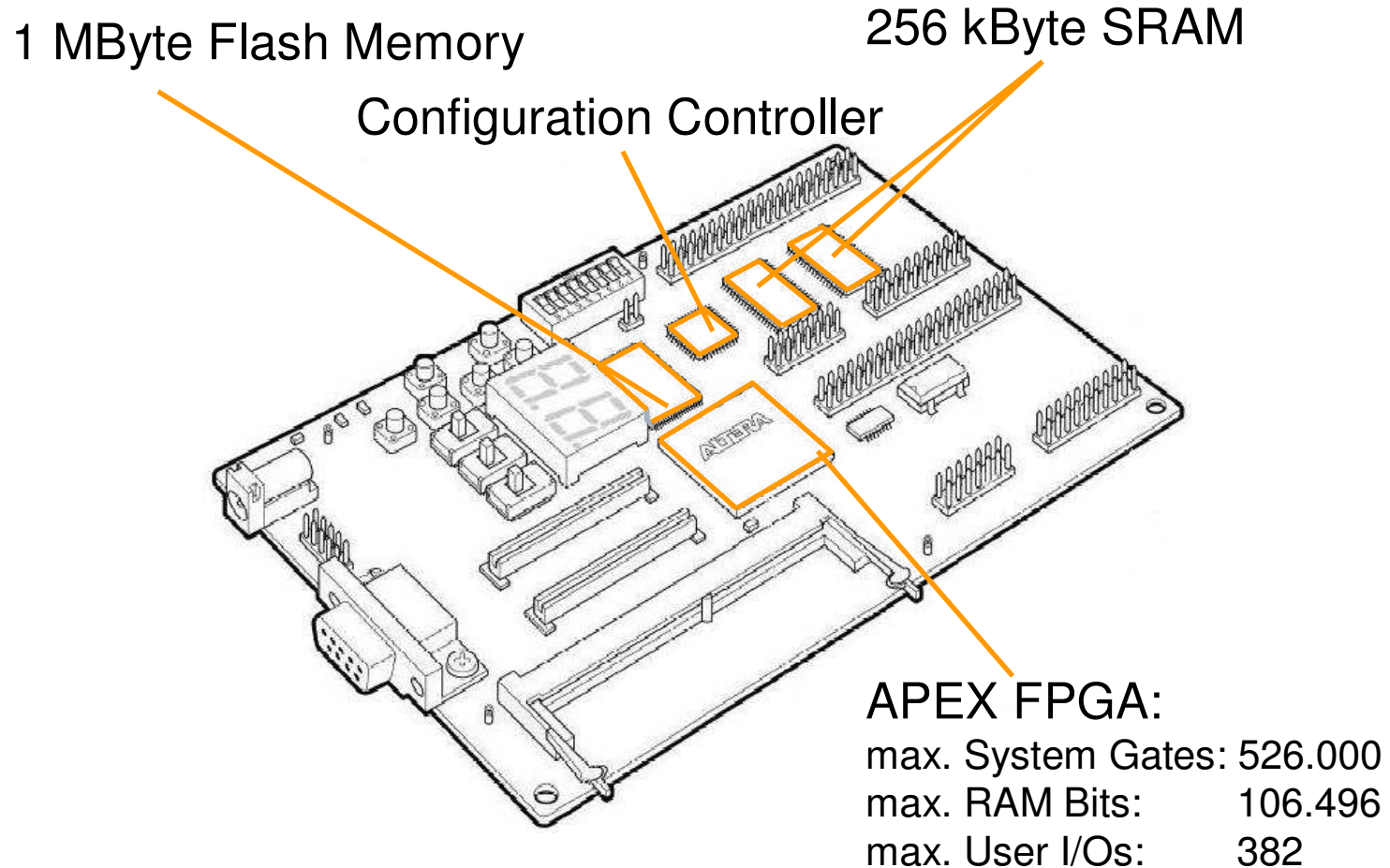
---



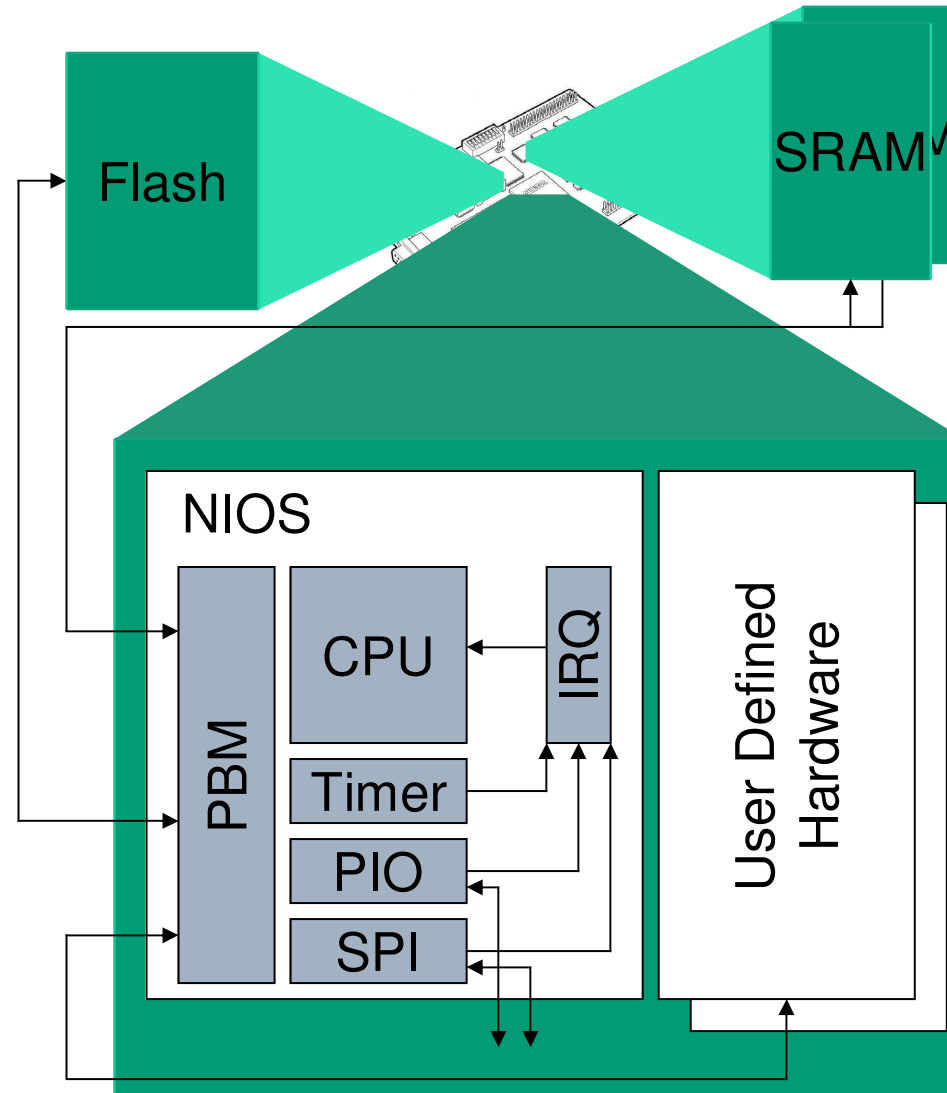


# Excalibur-Board

---



# SoC-Realisierung



# Arbeitsplan (1/2)

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
<b>Modellierung</b>								
AP1	60%	60%	40%					
AP2			30%	60%	40%	10%	10%	10%
<b>Analyse</b>								
AP3	40%	40%	10%					
AP4	10%	20%	30%	30%	30%	30%	20%	20%
AP5	10%	10%	10%	20%	20%	20%	10%	
<b>Synthese</b>								
AP6	40%	30%	30%	30%	10%			
AP7	40%	40%	40%	40%	30%	10%		
<b>Implement.</b>								
AP8			10%	10%	10%	10%	20%	20%
AP9				10%	10%	20%	20%	20%
AP10					20%	30%	30%	30%
AP11					20%	40%	50%	20%
AP12					10%	30%	40%	80%

# Arbeitsplan (2/2)

	Q1	Q2	Q3	Q4	Q5	Q6	Q7	Q8
Modellierung AP1 AP2	Modellierung zeitvarianter Anwendungen und Architekturen Definition zeitvarianter Allokationen und Bindungen							
Analyse AP3 AP4 AP5	Analyse von Spezifikationen mit zeitv. Allokation und Bindung Analyse von Ausfallsicherheit und Fehlertoleranz Erkennung von Leitungs- und Knotenstörungen und -ausfällen							
Synthese AP6 AP7	Mechanismen und Zielfunktionen zum Rerouting in ReCoNets Algorithmen zur Online-Hardware/Software-Partitionierung							
Implement. AP8 AP9 AP10 AP11 AP12	Analyseverfahren zur Bestimmung der Ausfallsicherheit Routingverfahren bei fehlerhaften Leitungen und Knoten Simulative Implementierung der Online-HW/SW-Partitionierung Aufbau des Demonstrators Test des Demonstrators an Beispielen							

# Gliederung

---

- Motivation
- Ziele
- Arbeitsprogramm:
  - Modellierung
  - Analyse
  - Synthese und Optimierung
  - Implementierung
- **Ausblick**

# Ausblick (1. + 2. Phase)

---

- 1. Phase terminiert in einem ersten einfachen Prototypen:
  - Automatisches Erkennen von Leitungs- und Knotenausfällen
  - Automatisches Rerouting bzw. Repartitionierung
  
- 2. Phase: Betriebssystemdienste:
  - Verwaltung von HW- und SW-Prozessen
  - Kombination der Entwurfsmethodiken auf Mikro- und Makroebene (ReCoNets & ReCoNodes)

# Ausblick (3. Phase)

---

- 3. Phase:
  - Optimierung der Netztopologie
  - Sicherheit und Zugang zu ReCoNets
  - Echtzeitverhalten in ReCoNets