

Mikroarchitekturgewahre Analyse des Ressourcenverbrauchs unter Berücksichtigung des Gesamtsystems

Echtzeit 2019, 21. November 2019

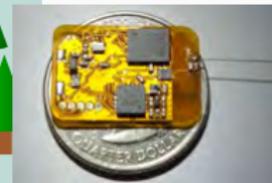
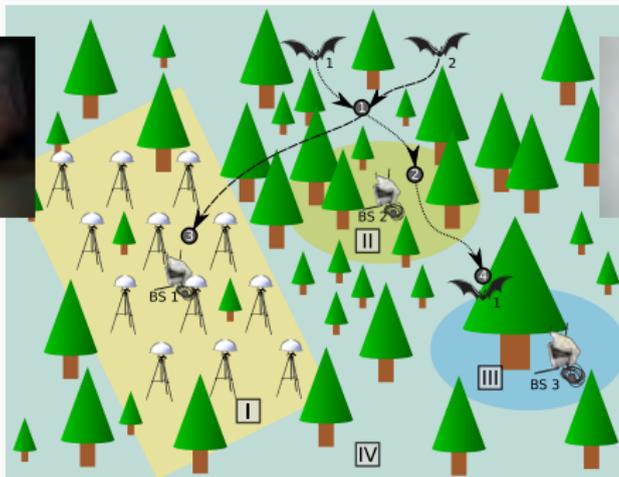
Phillip Raffeck

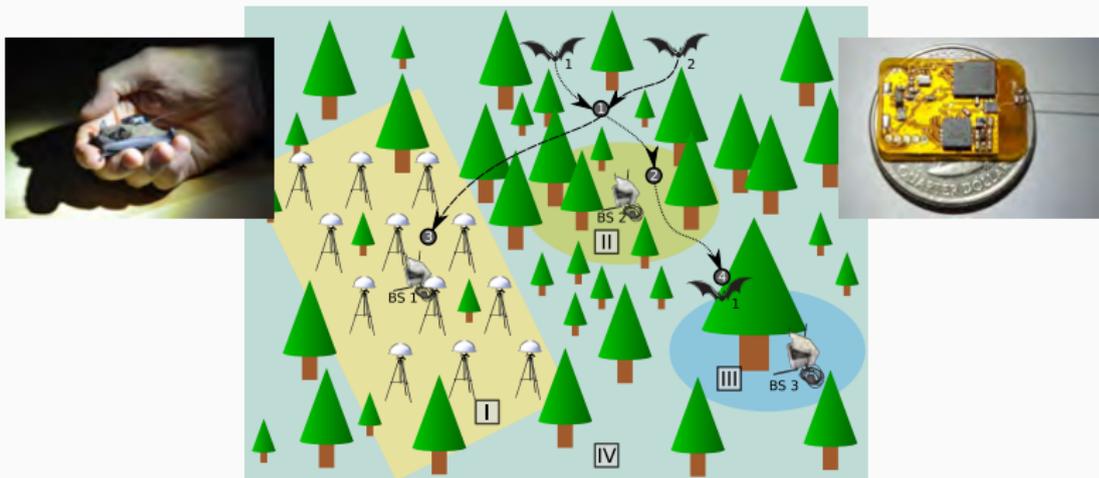
Friedrich-Alexander-Universität Erlangen-Nürnberg



EU EFRE funds
0704/883 25

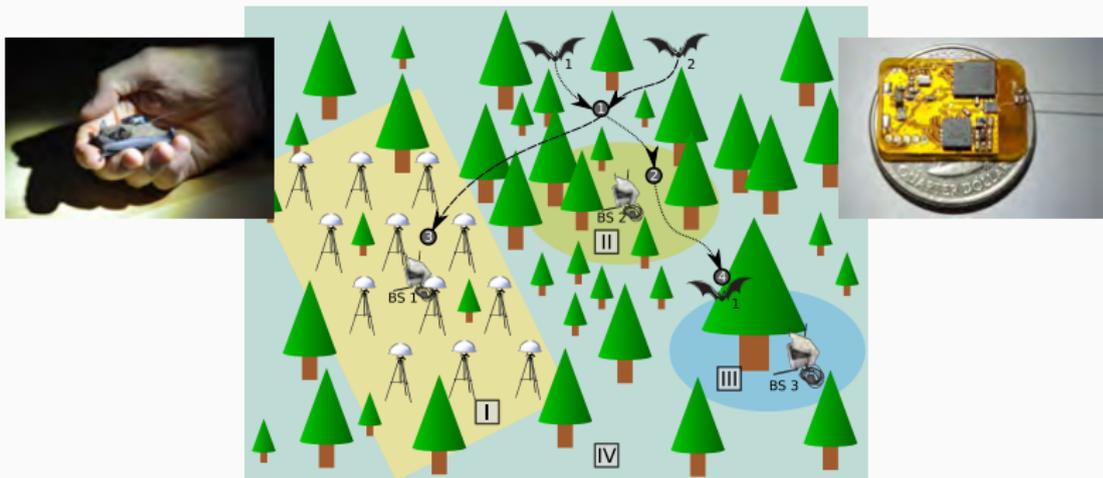






Überwachung von Fledermäusen

- Kommunikation \rightsquigarrow **Zeitanforderungen**
- Energieversorgung \rightsquigarrow **Energieanforderungen**



Überwachung von Fledermäusen

- Kommunikation \rightsquigarrow **Zeitanforderungen**
- Energieversorgung \rightsquigarrow **Energieanforderungen**
- ☞ **Garantiertes** Beachten der Anforderungen
- ☞ Analyse des schlimmstmöglichen Zeit- und Energiebedarfs



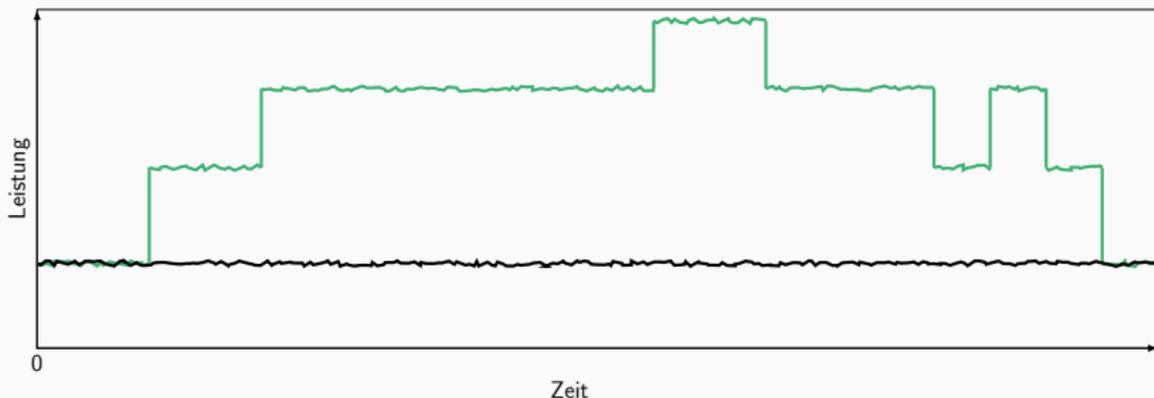
Annahmen

- vollpräemptives System
- Einkernprozessor
- Aufgaben aktivieren Peripheriegeräte

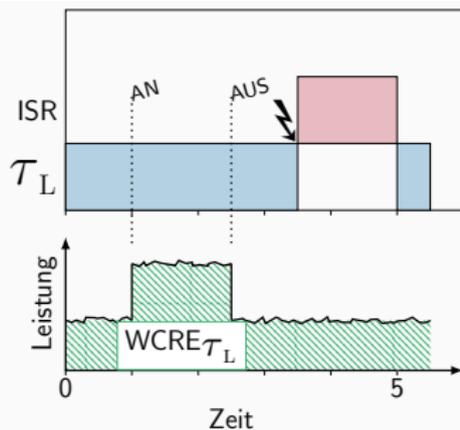
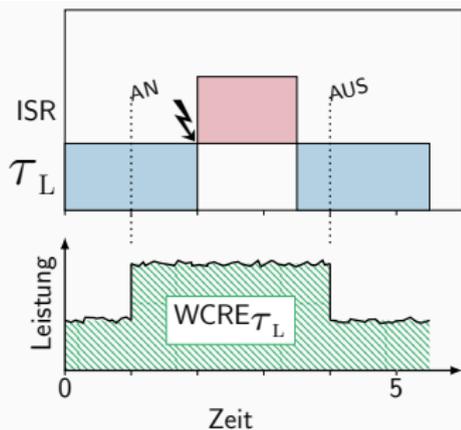


Annahmen

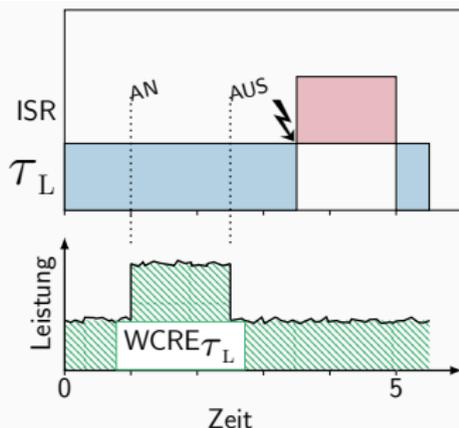
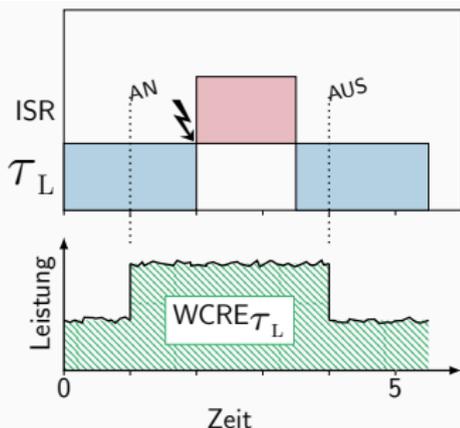
- vollpräemptives System
- Einkernprozessor
- Aufgaben aktivieren Peripheriegeräte



```
t_low_func() {  
    f();  
    gerät_an();  
    g();  
    gerät_aus();  
    h();  
}
```

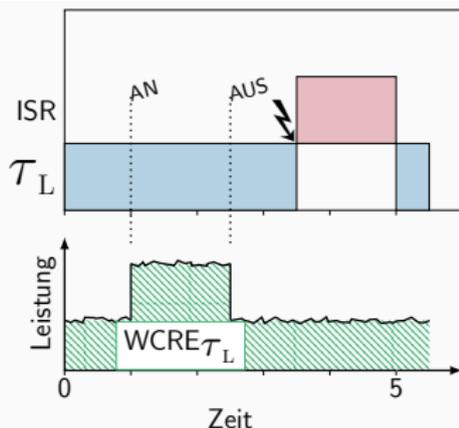
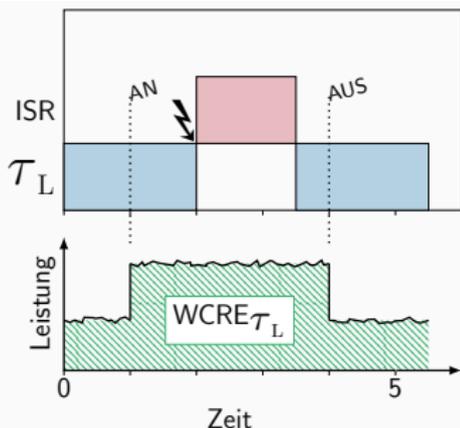


```
t_low_func() {  
  f();  
  gerät_an();  
  g();  
  gerät_aus();  
  h();  
}
```



- Temporär aktive Geräte dominieren Leistungsaufnahme
- Beeinflussen maximalen Energieverbrauch
 - bis zum **Erhalt einer Antwort**
 - Worst-Case Response Energy Consumption (WCRE)

```
t_low_func() {  
  f();  
  gerät_an();  
  g();  
  gerät_aus();  
  h();  
}
```



- Temporär aktive Geräte dominieren Leistungsaufnahme
- Beeinflussen maximalen Energieverbrauch
 - bis zum **Erhalt einer Antwort**
 - Worst-Case Response Energy Consumption (WCRE)
- **kontextsensitives** Beachten aller Systemaktivitäten

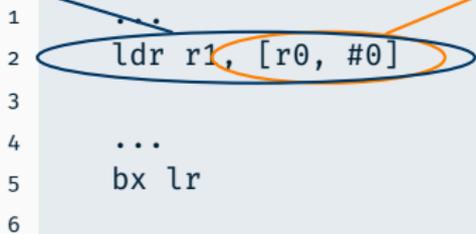
```
1   ...  
2   ldr r1, [r0, #0]  
3  
4   ...  
5   bx lr  
6
```

Im Datencache?

```
1  ...  
2  ldr r1, [r0, #0]  
3  
4  ...  
5  bx lr  
6
```

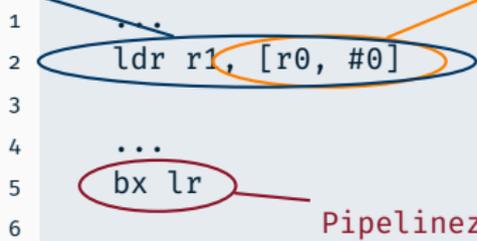
Im Befehls-cache?

Im Data-cache?



Im Befehls-cache?

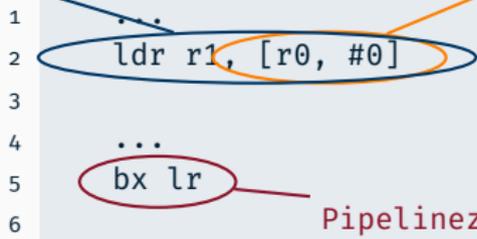
Im Data-cache?



Pipelinezustand verwerfen?

Im Befehls-cache?

Im Data-cache?



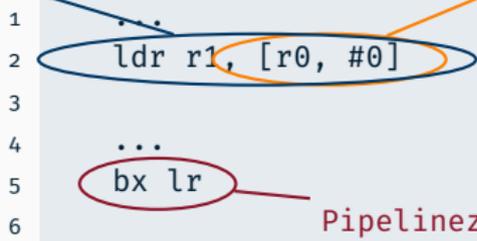
Pipelinezustand verwerfen?

Pessimistische Annahmen über die Mikroarchitektur

- Fehlendes Wissen über Pipeline
 - ☞ nichtüberlappende Ausführung
- Fehlendes Wissen über Cache
 - ☞ jeden Befehl aus Speicher laden

Im Befehls-cache?

Im Data-cache?

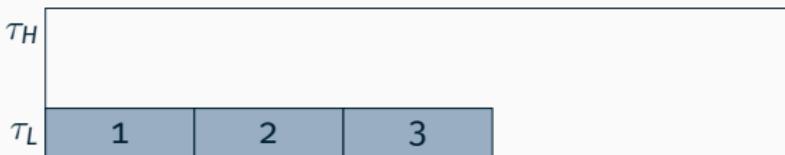


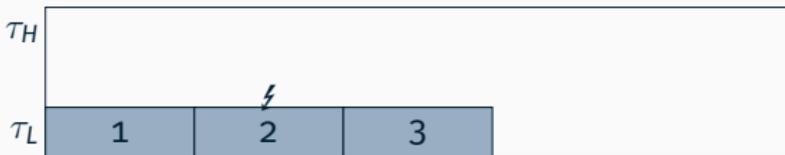
Pipelinezustand verwerfen?

Pessimistische Annahmen über die Mikroarchitektur

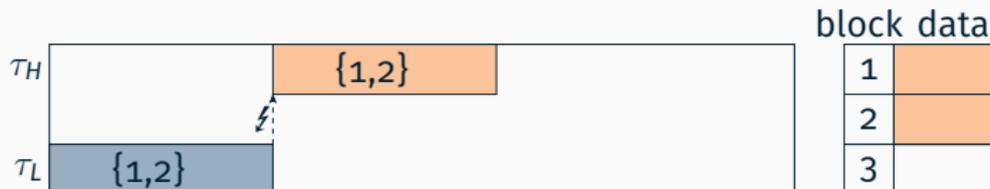
- Fehlendes Wissen über Pipeline
 - ☞ nichtüberlappende Ausführung
- Fehlendes Wissen über Cache
 - ☞ jeden Befehl aus Speicher laden

☞ **Unnötiger Pessimismus**



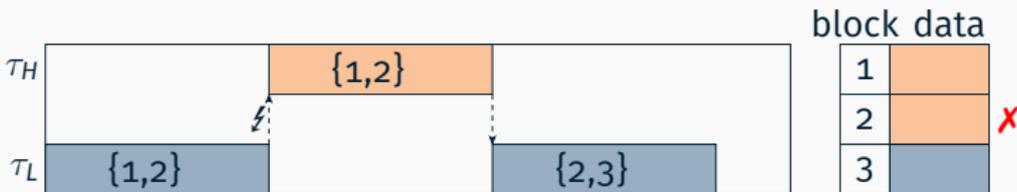






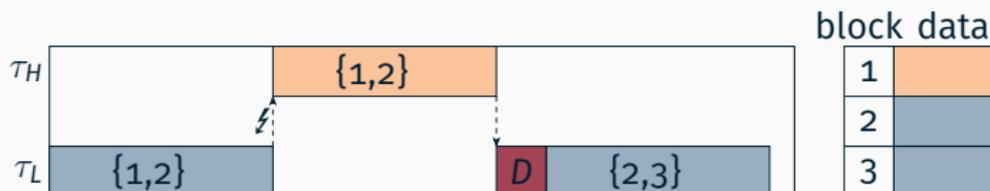
Cachebezogene Verzögerung

- τ_H verdrängt Daten von τ_L in Block 2



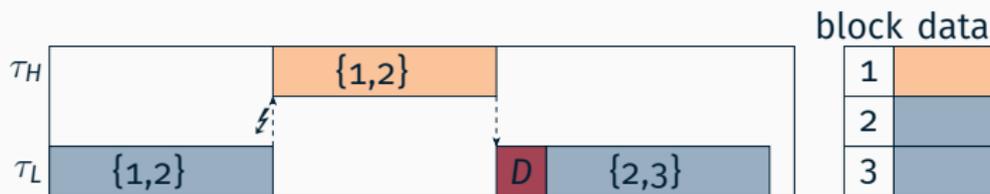
Cachebezogene Verzögerung

- τ_H verdrängt Daten von τ_L in Block 2



Cachebezogene Verzögerung

- τ_H verdrängt Daten von τ_L in Block 2
- *D* Delay: Nachladezeit von Block 2



Cachebezogene Verzögerung

- τ_H verdrängt Daten von τ_L in Block 2
- *D* Delay: Nachladezeit von Block 2
- **Verzögerung** aufgrund von Verdrängung

Probleme

- **Einfluss von Peripherie** auf Leistungsbedarf
- Einfluss der **Mikroarchitektur**
 - Mikroarchitekturzustand, Verdrängungen

Probleme

- **Einfluss von Peripherie** auf Leistungsbedarf
- Einfluss der **Mikroarchitektur**
 - Mikroarchitekturzustand, Verdrängungen

Ansatz

- Analyse des Ressourcenbedarfs
 - **Zeit**: Modellierung des **Mikroarchitekturzustands**
 - **Energie**: **Indirektion** über Zeit- und Leistungsanalyse
- Dekomposition des Systems in Leistungszustände
 - abschnittsweise Obergrenzen für Leistungsaufnahme
 - kontextgewahre Gesamtsystemanalyse

Motivation

Mikroarchitekturanalyse

Mikroarchitekturgewahre Gesamtsystemanalyse

Evaluation

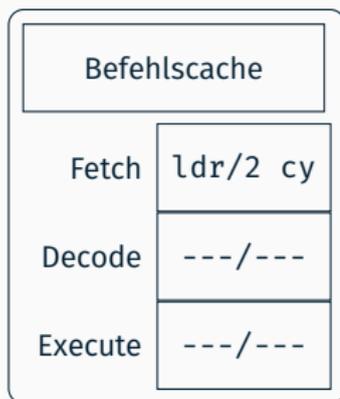
Fazit

Microarchitecture Execution Graph (MEG) [1]

- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen

Microarchitecture Execution Graph (MEG) [1]

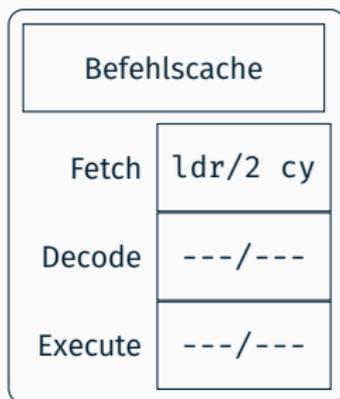
- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen



[1] I. Stein: ILP-based path analysis on abstract pipeline state graphs. Doktorarbeit. 2010

Microarchitecture Execution Graph (MEG) [1]

- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen



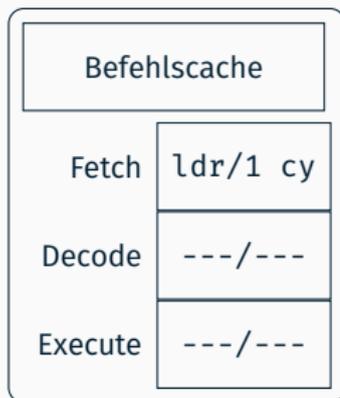
Ausführungszeit

- bekannter Startzustand
- Auswirkungen pro Prozessortakt
- Akkumulation der Übergänge

[1] I. Stein: ILP-based path analysis on abstract pipeline state graphs. Doktorarbeit. 2010

Microarchitecture Execution Graph (MEG) [1]

- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen



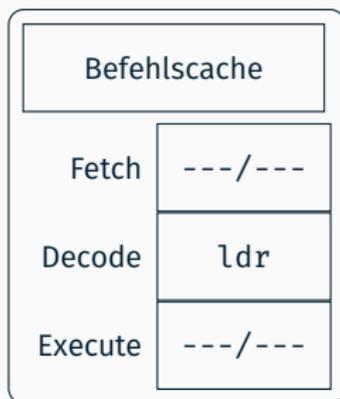
Ausführungszeit

- bekannter Startzustand
- Auswirkungen pro Prozessortakt
- Akkumulation der Übergänge

[1] I. Stein: ILP-based path analysis on abstract pipeline state graphs. Doktorarbeit. 2010

Microarchitecture Execution Graph (MEG) [1]

- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen



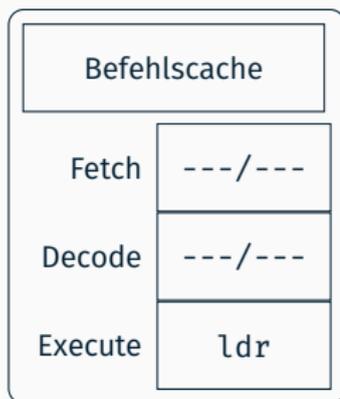
Ausführungszeit

- bekannter Startzustand
- Auswirkungen pro Prozessortakt
- Akkumulation der Übergänge

[1] I. Stein: ILP-based path analysis on abstract pipeline state graphs. Doktorarbeit. 2010

Microarchitecture Execution Graph (MEG) [1]

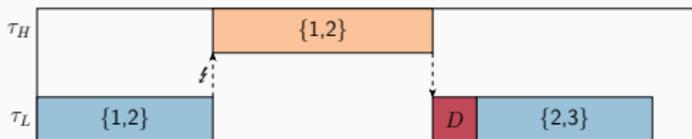
- gerichteter Graph
- Knoten: (abstrakter) Mikroarchitekturzustand
- Kanten: mögliche Übergänge zwischen Zuständen



Ausführungszeit

- bekannter Startzustand
- Auswirkungen pro Prozessortakt
- Akkumulation der Übergänge

[1] I. Stein: ILP-based path analysis on abstract pipeline state graphs. Doktorarbeit. 2010

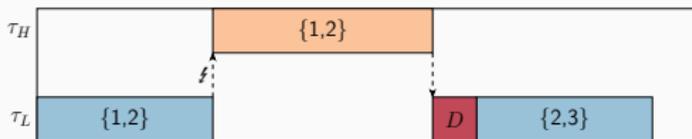


Plattformspezifische Kosten

- bei Verdrängung und Wiedereinlastung
- Cachebezogen [2]
- Pipelinebezogen [3]

[2] J. Busquets-Mataix et al.: Adding instruction cache effect to schedulability analysis of preemptive real-time systems. RTAS 1996

[3] J. Schneider: Cache and Pipeline Sensitive Fixed Priority Scheduling for Preemptive Real-Time Systems. RTSS 2000



Plattformspezifische Kosten

- bei Verdrängung und Wiedereinlastung
- Cachebezogen [2]
- Pipelinebezogen [3]
- Hardware-Modell nötig

[2] J. Busquets-Mataix et al.: Adding instruction cache effect to schedulability analysis of preemptive real-time systems. RTAS 1996

[3] J. Schneider: Cache and Pipeline Sensitive Fixed Priority Scheduling for Preemptive Real-Time Systems. RTSS 2000

Mikroarchitekturgewahre Gesamtsystemanalyse

WCRE-Abschätzung mittels SysWCEC [4]

- Aufzählung aller Systemzustände
 - ↔ Systemaufrufe, Verdrängungen, Unterbrechungen
- Unterteilung des Systems in atomare Bereiche
 - **konstante, kontextsensitive Leistungsaufnahme**
 - **Trennung bei Systemaufrufe**

[4] P. Wagemann et al.: Whole-System Worst-Case Energy-Consumption Analysis for Energy-Constrained Real-Time Systems. ECRTS 2018

WCRE-Abschätzung mittels SysWCEC [4]

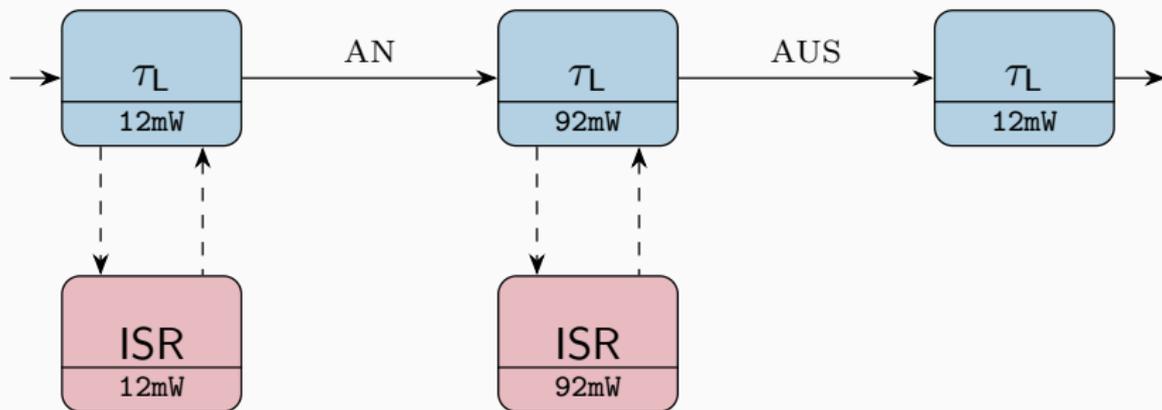
- Aufzählung aller Systemzustände
 - ↔ Systemaufrufe, Verdrängungen, Unterbrechungen
- Unterteilung des Systems in atomare Bereiche
 - **konstante, kontextsensitive Leistungsaufnahme**
 - **Trennung bei Systemaufrufe**
- Power-State-Transition Graph (PSTG)

[4] P. Wägemann et al.: Whole-System Worst-Case Energy-Consumption Analysis for Energy-Constrained Real-Time Systems. ECRTS 2018

WCRE-Abschätzung mittels SysWCEC [4]

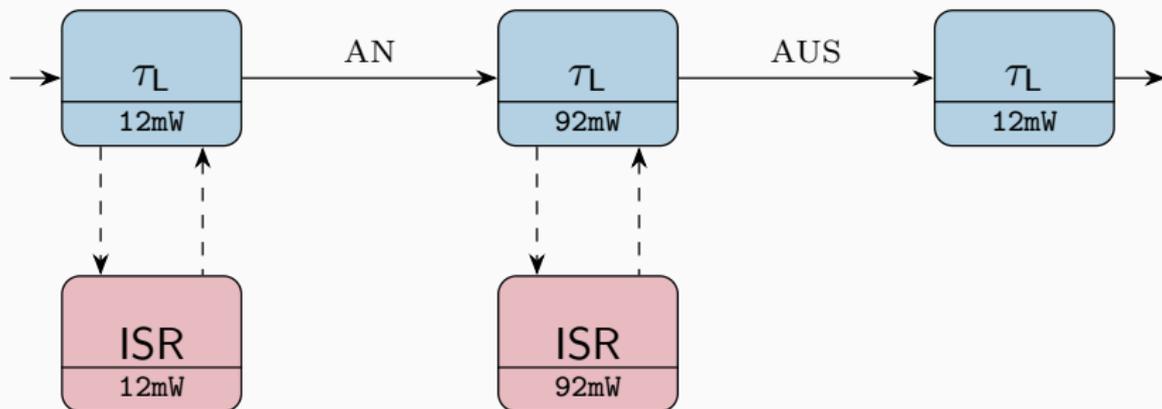
- Aufzählung aller Systemzustände
 - ↔ Systemaufrufe, Verdrängungen, Unterbrechungen
- Unterteilung des Systems in atomare Bereiche
 - **konstante, kontextsensitive Leistungsaufnahme**
 - **Trennung bei Systemaufrufe**
- Power-State-Transition Graph (PSTG)
- PSTG \mapsto Optimierungsproblem (ILP)

[4] P. Wägemann et al.: Whole-System Worst-Case Energy-Consumption Analysis for Energy-Constrained Real-Time Systems. ECRTS 2018



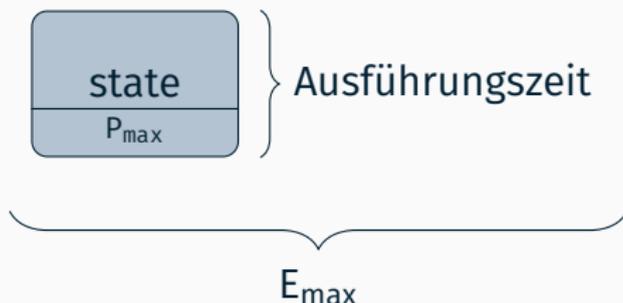
Power-State-Transition Graph

- Übergänge für Systemaufrufe und Unterbrechungen
- kontextsensitive maximale Leistungsaufnahme



Power-State-Transition Graph

- Übergänge für Systemaufrufe und Unterbrechungen
- kontextsensitive maximale Leistungsaufnahme
- 👉 **Globaler leistungsgewahrer Kontrollflussgraph**



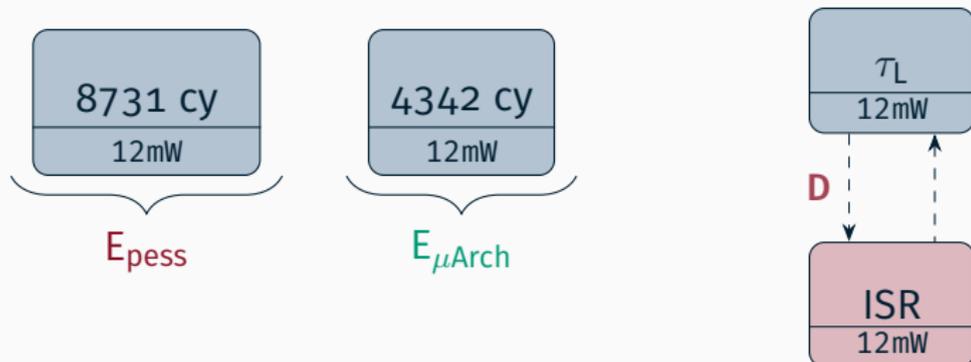
WCRE-Abschätzung

- maximale Ausführungszeit pro Zustand
- Kombination mit maximaler Leistungsaufnahme
- Übertragung in Optimierungsproblem
- Optimierungsziel: **WCRE**



Beachtung der Mikroarchitektur

- mikroarchitekturgewahre Kosten pro atomarer Region
 - ✓ Microarchitecture Execution Graph



Beachtung der Mikroarchitektur

- mikroarchitekturgewahre Kosten pro atomarer Region
 - ✓ Microarchitecture Execution Graph
- Beachtung von Verdrängungen
 - ✓ pipeline- & cachebezogene Verzögerungen
 - ✓ Integration in Optimierungsproblem

Evaluation

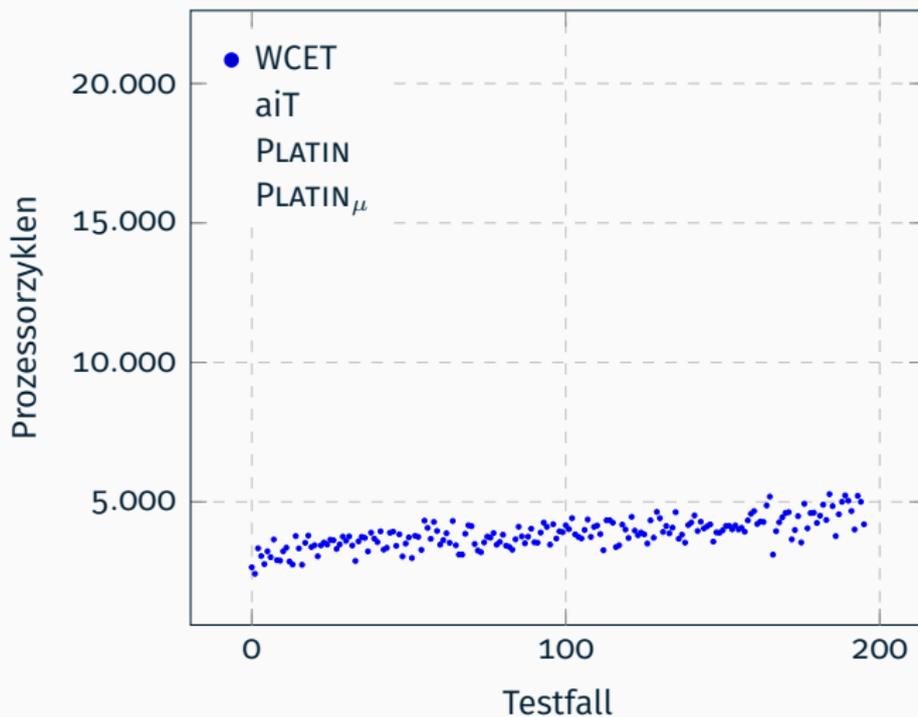
Versuchsaufbau #1

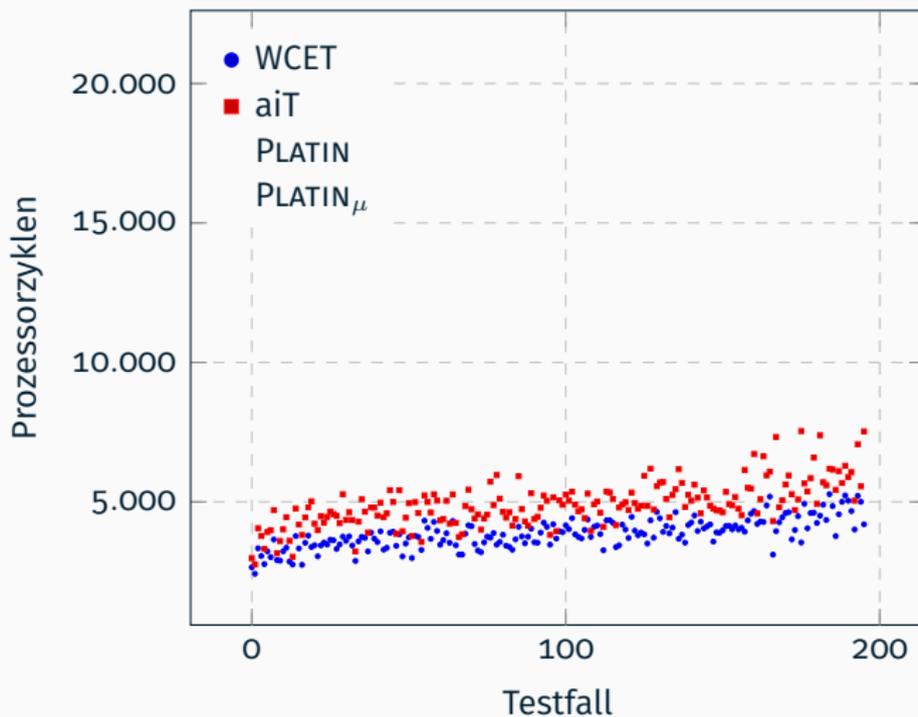
- Zielplattform: Infineon XMC4500
 - ARM Cortex-M4 Prozessor
- Generierte Testprogramme mit **bekanntem WCET**
 - werkzeuggestützt [5, 6]
- Vergleich der WCET-Abschätzungen
 - aiT
 - PLATIN [7]
 - Mikroarchitekturgewahres PLATIN_μ

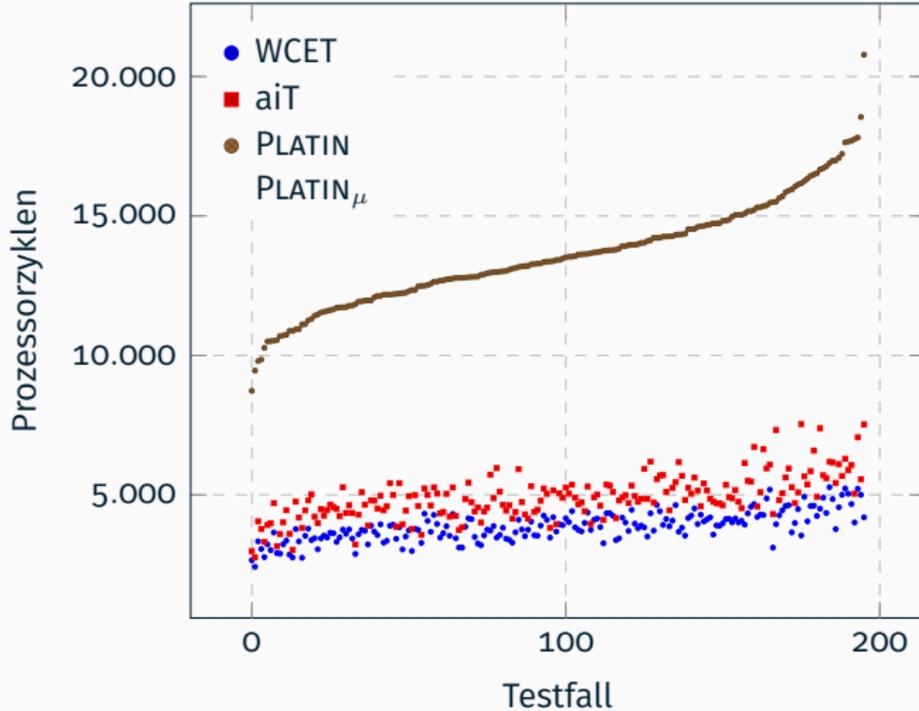
[5] C. Eichler et al.: Demo Abstract: Tooling Support for Benchmarking Timing Analysis. RTAS 2017

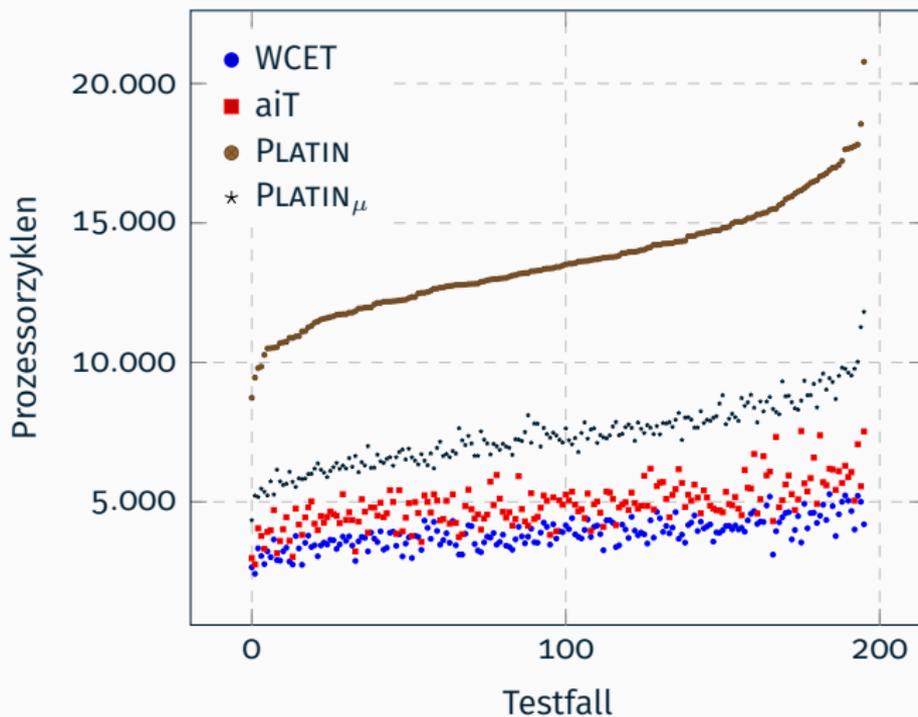
[6] P. Wagemann et al.: Benchmark Generation for Timing Analysis. RTAS 2017

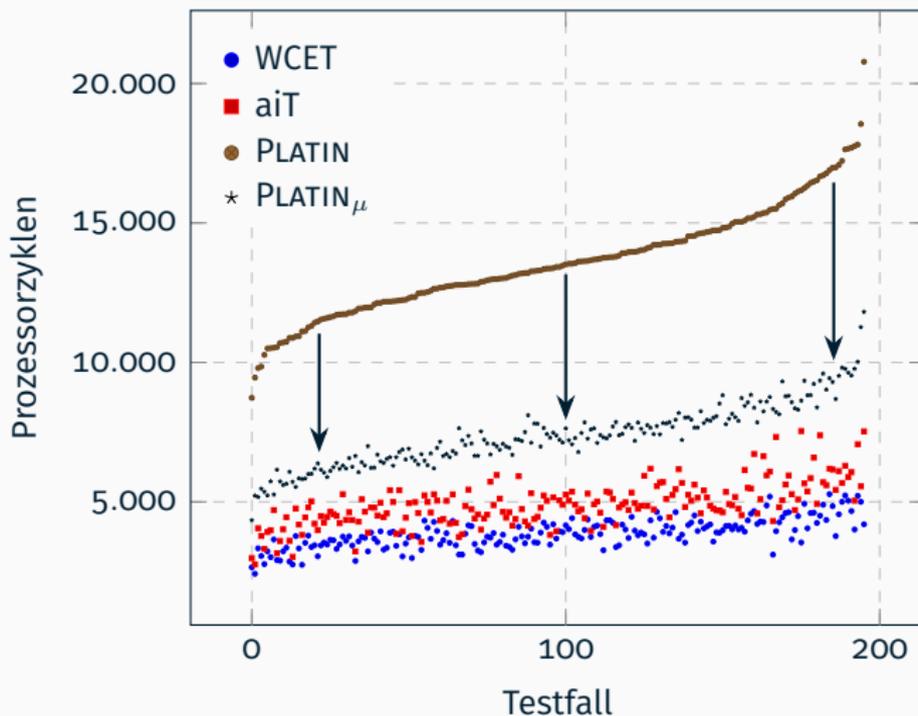
[7] P. Puschner et al.: The T-CREST Approach of Compiler and WCET-analysis Integration. SEUS 2013









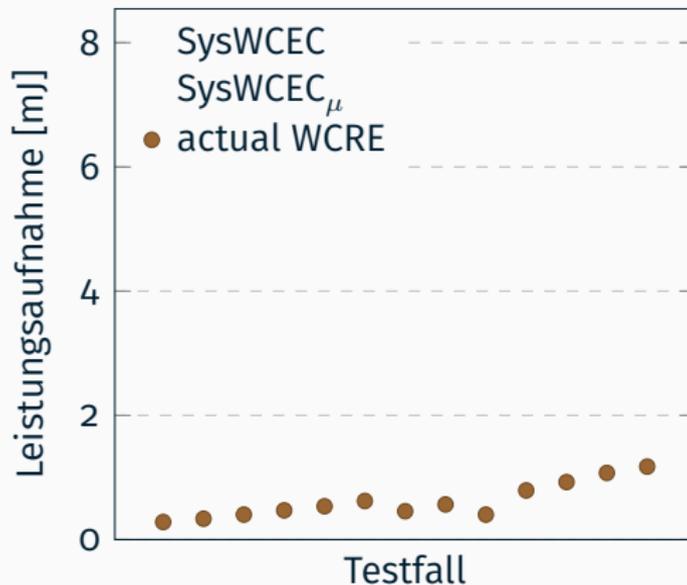


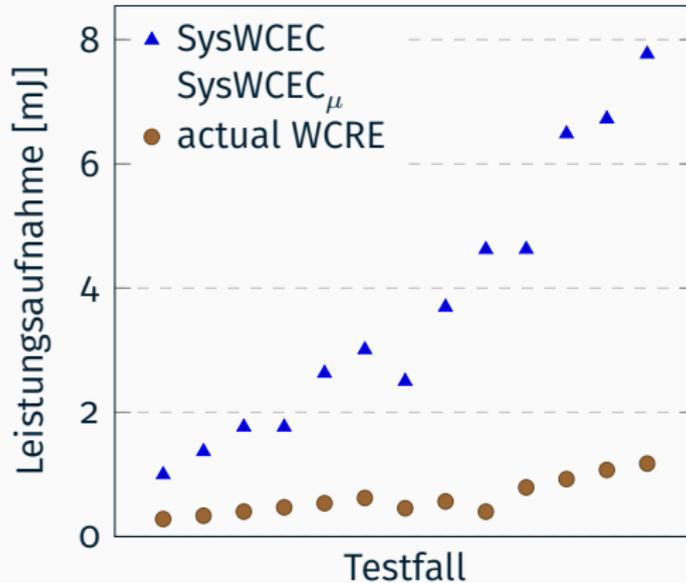
38-54% kleinere WCET-Abschätzungen

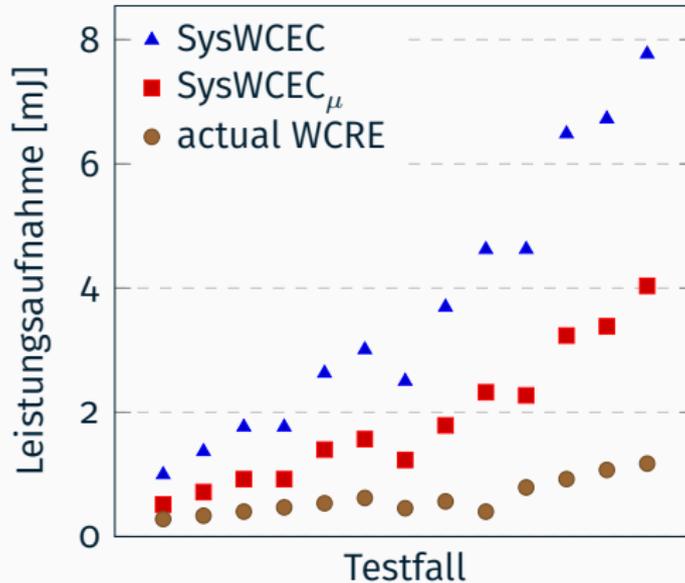
Versuchsaufbau #2

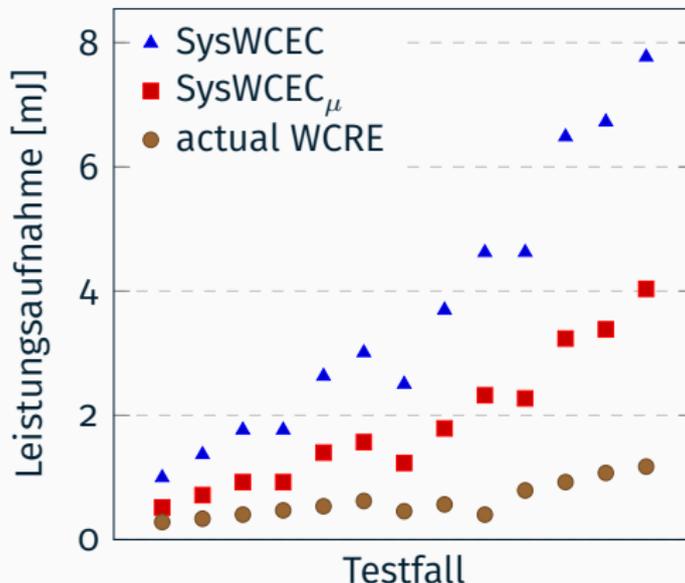
- Zielplattform: Infineon XMC4500
 - ARM Cortex-M4 Prozessor
- Generierte Testsysteme mit **bekannter WCRE**
 - werkzeuggestützt [8]
- Vergleich der WCRE-Abschätzungen von SysWCEC
 - Pessimistisch
 - Mikroarchitekturgewahr (SysWCEC_{μ})

[8] C. Eichler et al.: GenEE: A Benchmark Generator for Static Analysis Tools of Energy-Constrained Cyber-Physical Systems. CPS-IoTBench 2019









- Überabschätzung von 170% (Median)
- 46-51% kleinere WCRE-Abschätzungen

Fazit

- ✗ Abschätzung Ressourcenbedarf
- ✗ Berücksichtigung des gesamten Systems
 - Kombination verschiedener Analysetechniken
 - Integration in Gesamtsystemanalyse



```
gitlab.cs.fau.de/  
syswcec-uarch
```

- ✓ Mikroarchitekturgewahr
- ✓ Systemgewahr
- ✓ Peripheriegewahr
- ✓ **Zeit und Energie**

- ✗ Abschätzung Ressourcenbedarf
- ✗ Berücksichtigung des gesamten Systems
 - Kombination verschiedener Analysetechniken
 - Integration in Gesamtsystemanalyse



```
gitlab.cs.fau.de/  
syswcec-uarch
```

- ✓ Mikroarchitekturgewahr
- ✓ Systemgewahr
- ✓ Peripheriegewahr
- ✓ **Zeit** und **Energie**

Fragen?