Energiegewahre Optimierung eingebetteter Echtzeitsysteme

Jahrestagung der Fachgruppe Frauen und Informatik 2025

03. Mai 2025

Eva Dengler

Friedrich-Alexander-Universität Erlangen-Nürnberg

Lehrstuhl für Verteilte Systeme
und Betriebssysteme

Friedrich-Alexander-Universität
Technische Fakultät

Über mich

Eva Dengler
Promotionsstudentin @ FAU Erlangen-Nürnberg seit Nov. 2022
Lehrstuhl für Informatik 4 (Systemsoftware)
Eingebettete Systemsoftware
Fokus auf energiebeschränkte Echtzeitsysteme

Preis WITA 2025 für die beste Masterarbeit
Clock-Tree–Aware Resource-Consumption Models for Embedded SoC Platforms

Echtzeit

Wodurch kennzeichnet sich ein *Echtzeitsystem*?

- Interaktion mit Umgebung (z.B. Sensoren)
- Rechtzeitigkeit (Termineinhaltung)

Echtzeit

Wodurch kennzeichnet sich ein *Echtzeitsystem*?

- Interaktion mit Umgebung (z.B. Sensoren)
- Rechtzeitigkeit (Termineinhaltung)
  $\Rightarrow$ EZS zeichnen sich nicht dadurch aus, dass sie besonders schnell sind

Echtzeit

Wodurch kennzeichnet sich ein *Echtzeitsystem*?

- Interaktion mit Umgebung (z.B. Sensoren)
- Rechtzeitigkeit (Termineinhaltung)
  $\Rightarrow$ EZS zeichnen sich nicht dadurch aus, dass sie besonders schnell sind

Taskbeschreibung einzelner periodischer Aufgabe:

- Periode P
- max. Ausführungszeit (WCET) e
- relativer Termin / Deadline D

Echtzeit

Wodurch kennzeichnet sich ein *Echtzeitsystem*?

- Interaktion mit Umgebung (z.B. Sensoren)
- Rechtzeitigkeit (Termineinhaltung)
  $\Rightarrow$ EZS zeichnen sich nicht dadurch aus, dass sie besonders schnell sind

Taskbeschreibung einzelner periodischer Aufgabe:

- Periode P
- max. Ausführungszeit (WCET) e
- relativer Termin / Deadline D

$\exists$ verschiedene Strategien zur *Planung* des EZS, um *Garantien* geben zu können

Eingebettete Echtzeitsysteme

Eingebettete Echtzeitsysteme

- Eingebettete Systeme haben Geräte

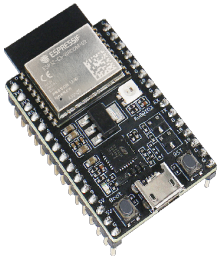
Eingebettete Echtzeitsysteme

- Eingebettete Systeme haben Geräte , die Energie benötigen

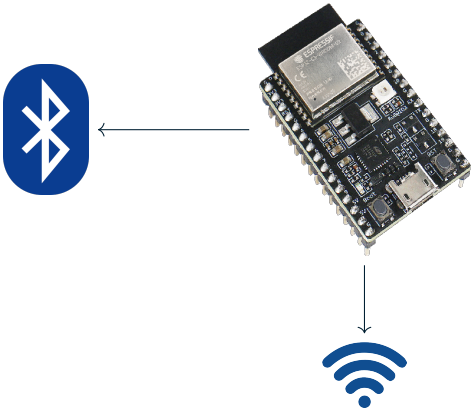
Eingebettete Echtzeitsysteme

- Eingebettete Systeme haben Geräte , die Energie benötigen
- Frage: Wie kann man *Energie sparen*, um möglichst lang zu agieren, während man nach wie vor *Echtzeitgarantien* geben kann?

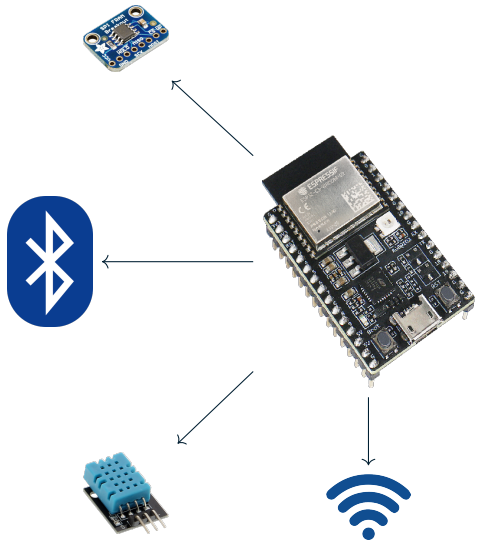
Geräte in eingebetteten Systemen



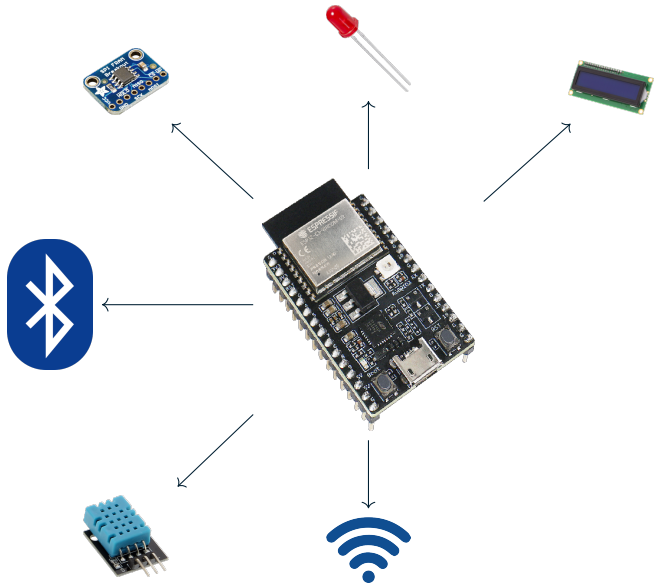
Geräte in eingebetteten Systemen



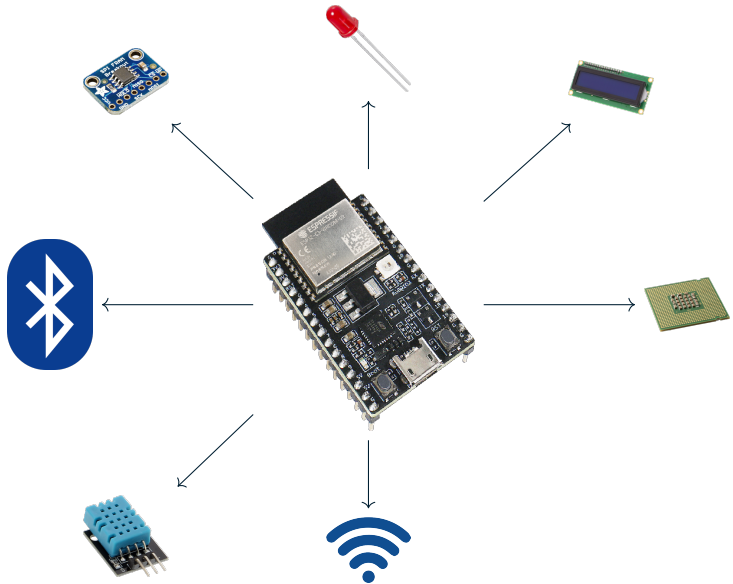
Geräte in eingebetteten Systemen



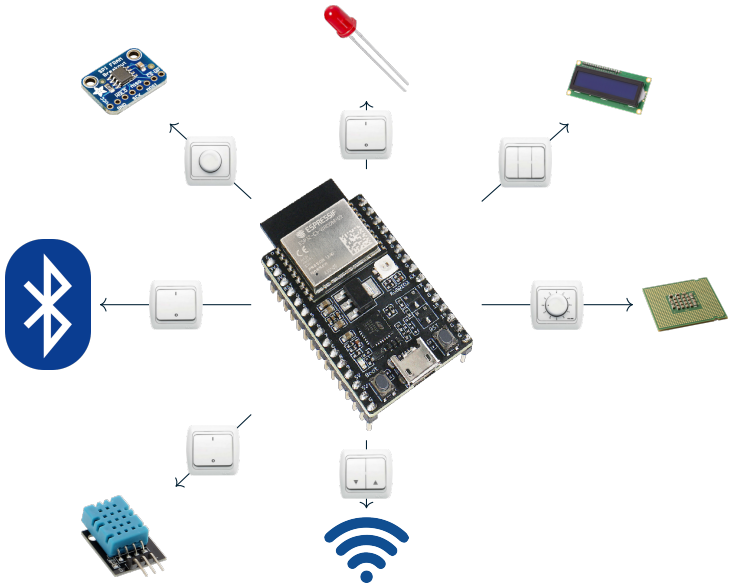
Geräte in eingebetteten Systemen



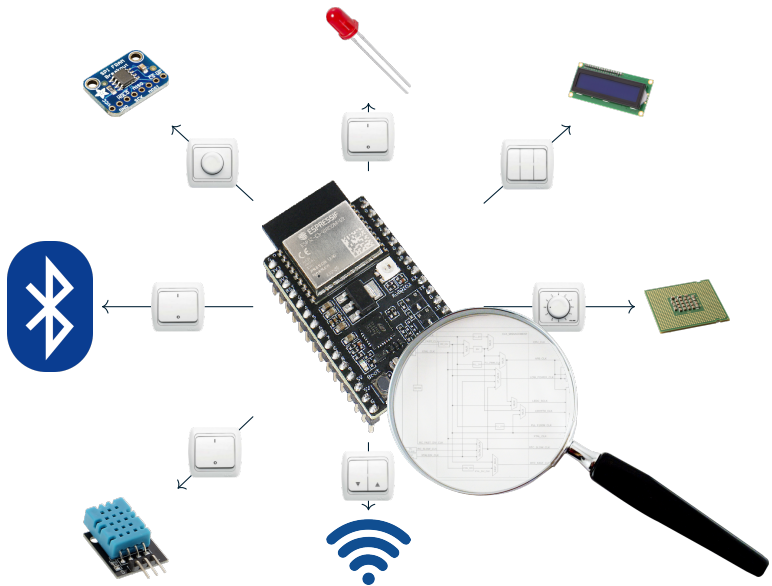
Geräte in eingebetteten Systemen



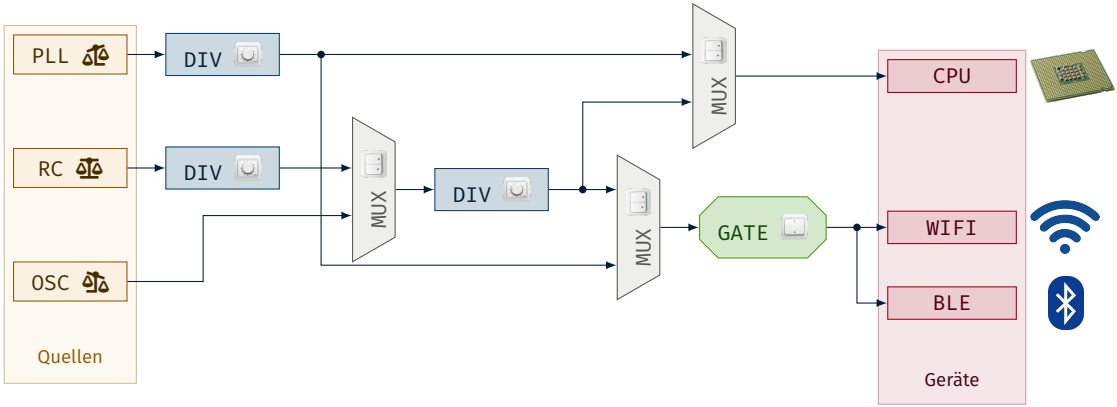
Geräte in eingebetteten Systemen



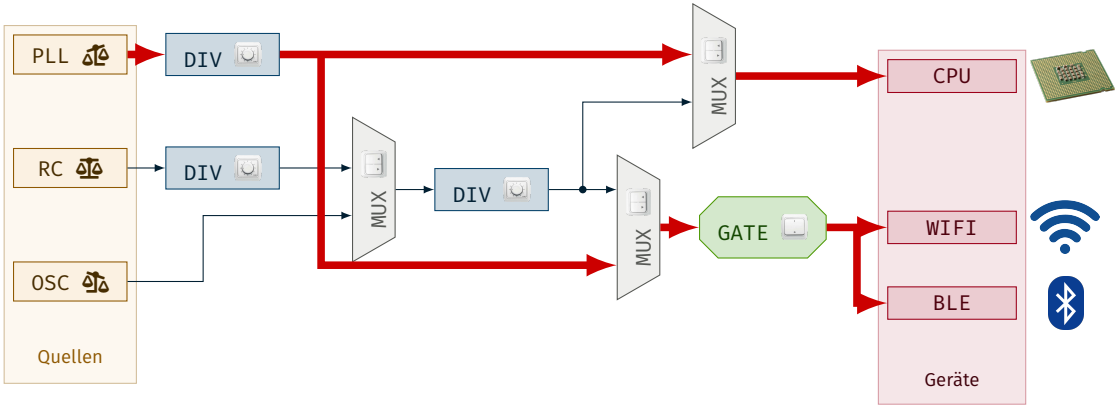
Geräte in eingebetteten Systemen

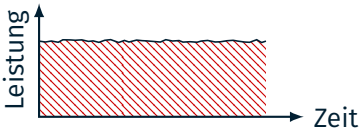


Der *Clock Tree*



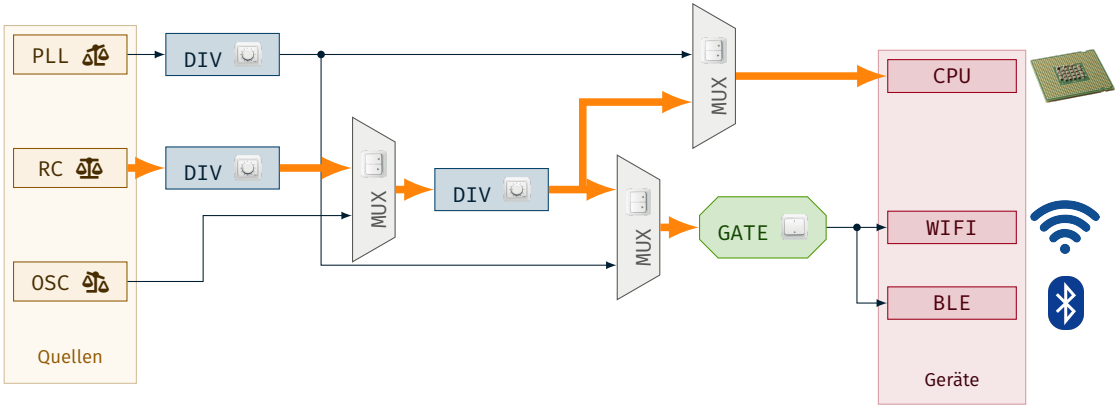
Der *Clock Tree*

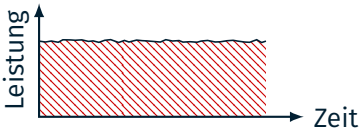




Konfiguration *HIGH*

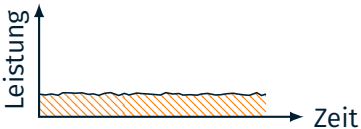
Der *Clock Tree*





Konfiguration *HIGH*

v.s.



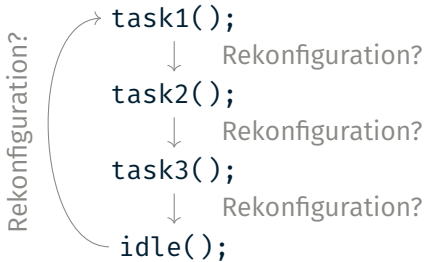
Konfiguration *LOW*

Aufgabenmodell

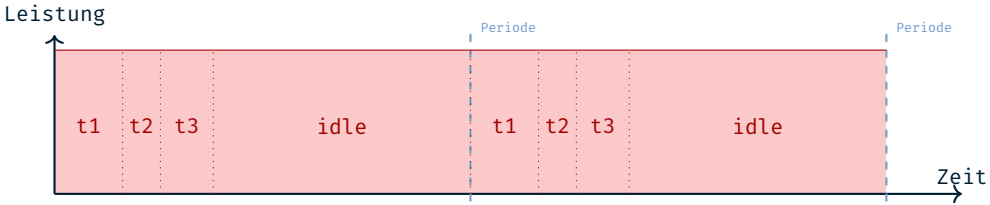
```
main:
      while (true):
                task1();
                task2();
                task3();
                 idle();
```

Aufgabenmodell

Aufgabenmodell

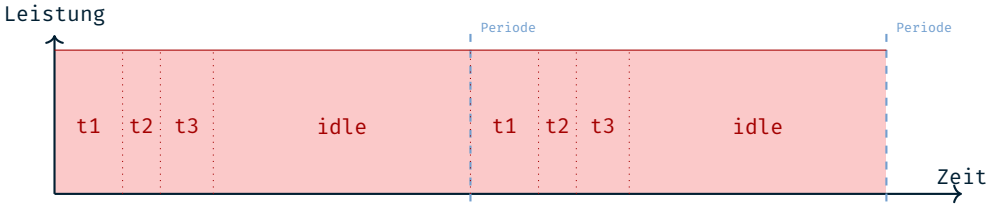


Problemanalyse



all-always-on Ansatz

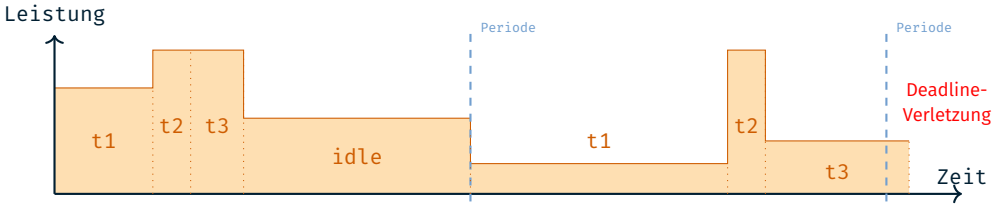
Problemanalyse



all-always-on Ansatz

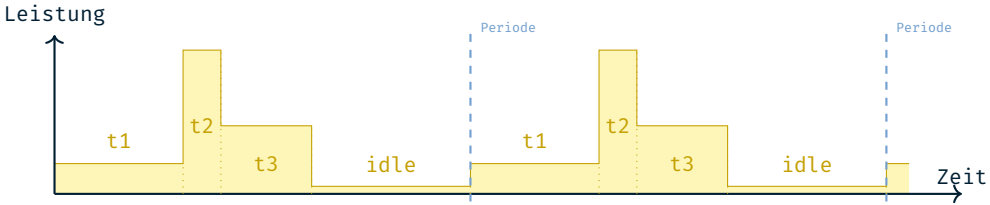
× Minimerung des Energiebedarfs

Problemanalyse



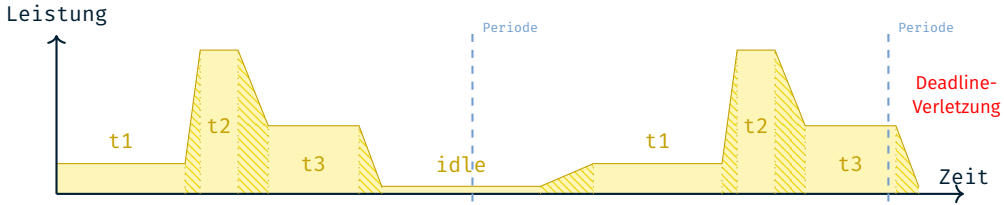
Feedback-basierter Ansatz: Rekonfigurationen während der Ausführung

- Minimerung des Energiebedarfs
- × Echtzeitgarantien



Statischer Ansatz: Analyse vor der Ausführung

- Minimerung des Energiebedarfs
- Echtzeitgarantien



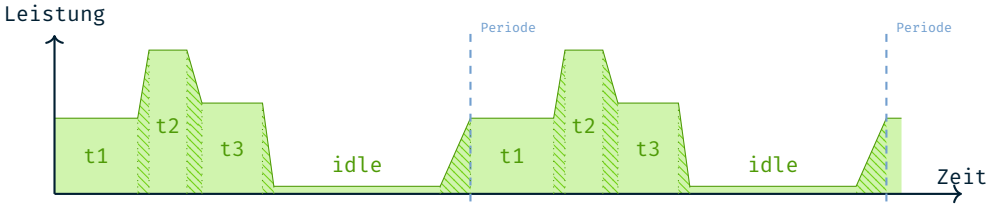
Statischer Ansatz ohne Rekonfigurationskosten

- Minimerung des Energiebedarfs
- × Echtzeitgarantien
- × Einbeziehen der Rekonfigurationskosten

Probleme

1. Ansätze, die nur die CPU betrachten, …
   - missachten den Energiebedarf von **Geräten**
   - ignorieren **Abhängigkeiten der Geräte** und des **Clock Tree**s
2. **Keine Garantien** feedback-basierender Ansätze
3. fehlende **Rekonfigurationskosten**

Konzept

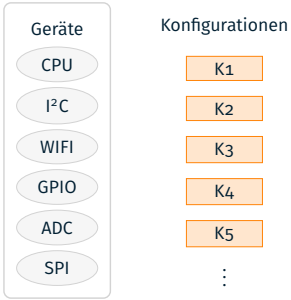


Statischer Ansatz mit Rekonfigurationskosten

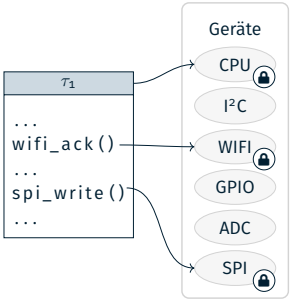
- ✓ Minimerung des Energiebedarfs
- ✓ Echtzeitgarantien
- ✓ Einbeziehen der Rekonfigurationskosten

Umsetzung

Umsetzung


Geräte
CPU
I²C
WIFI
GPIO
ADC
SPI
Konfigurationen
K1
K2
K3
K4
K5
⋮

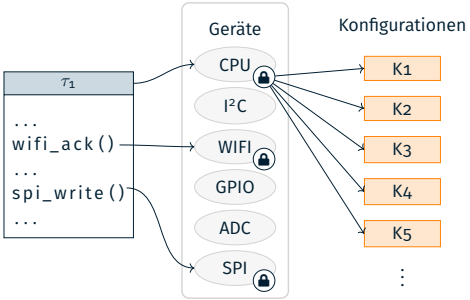
Umsetzung



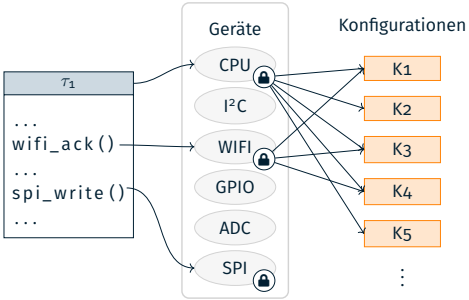
Konfigurationen

- K1
- K2
- K3
- K4
- K5
- ⋮

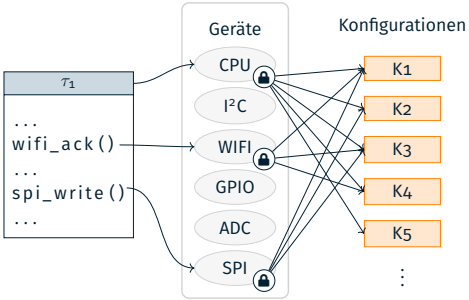
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
⋮

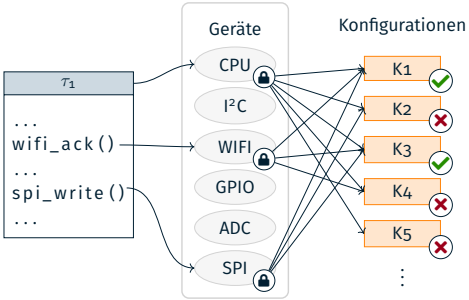
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
⋮

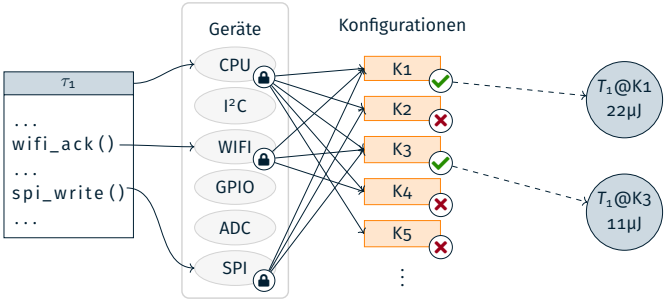
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5

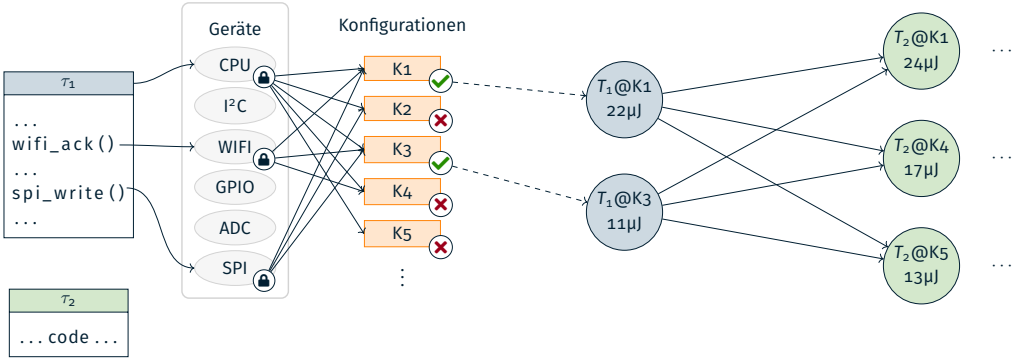
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5

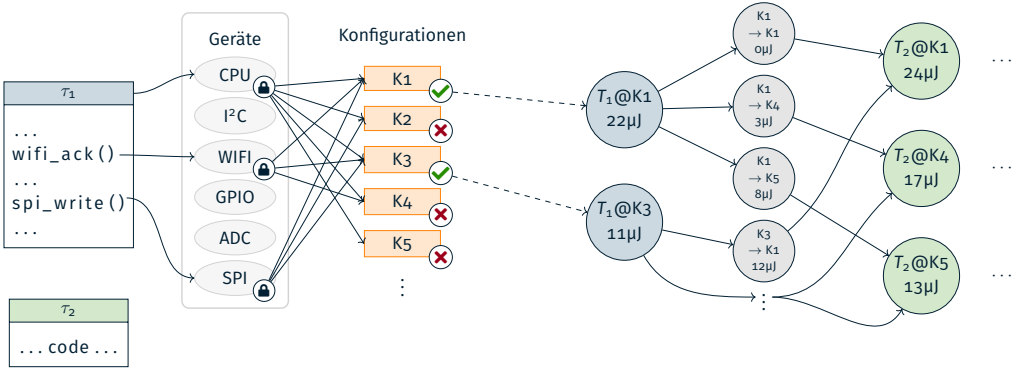
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
T_1@K1
22µJ
T_1@K3
11µJ

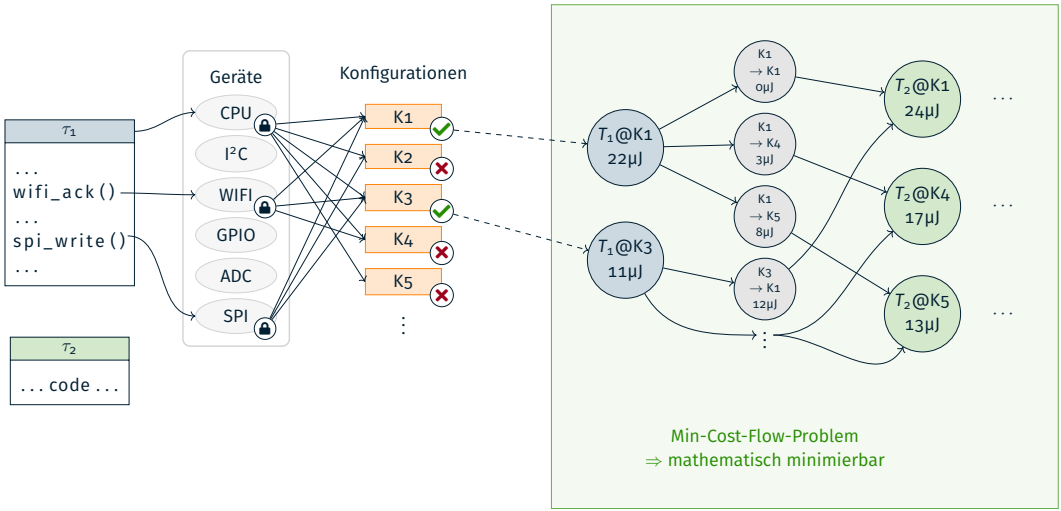
Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
τ_2
... code ...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
T_1@K1
22µJ
T_1@K3
11µJ
T_2@K1
24µJ
T_2@K4
17µJ
T_2@K5
13µJ

Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
T_1@K1
22µJ
T_1@K3
11µJ
K1 → K1
0µJ
K1 → K4
3µJ
K1 → K5
8µJ
K3 → K1
12µJ
T_2@K1
24µJ
T_2@K4
17µJ
T_2@K5
13µJ
τ_2
... code ...

Umsetzung


Geräte
Konfigurationen
τ_1
...
wifi_ack()
...
spi_write()
...
CPU
I²C
WIFI
GPIO
ADC
SPI
K1
K2
K3
K4
K5
τ_2
... code ...
T_1@K1 22µJ
T_1@K3 11µJ
K1 → K1 0µJ
K1 → K4 3µJ
K1 → K5 8µJ
K3 → K1 12µJ
T_2@K1 24µJ
T_2@K4 17µJ
T_2@K5 13µJ
Min-Cost-Flow-Problem
⇒ mathematisch minimierbar

Formalisierung

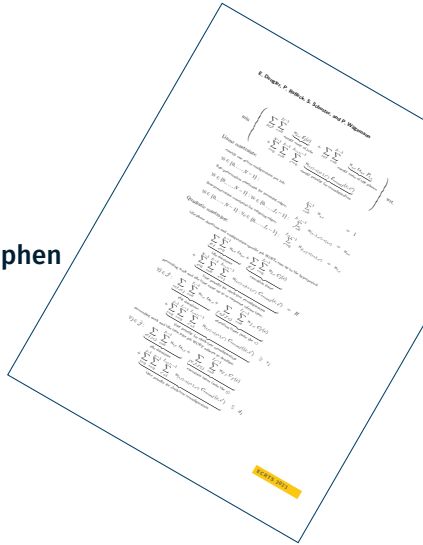
min **Energiekosten** aller Knoten
+ **Energiekosten** der Rekonfiguration

w.r.t.
Flusseinschränkungen im **Clock-Tree-Rekonfigurationsgraphen**
alle Zeiten aufsummiert ergeben verfügbare **Gesamtzeit**
jeder Task ist bis zu seiner **Deadline** beendet
...

Formalisierung

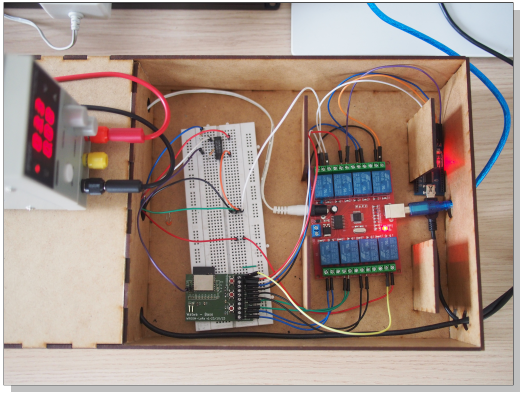
min **Energiekosten** aller Knoten
+ **Energiekosten** der Rekonfiguration

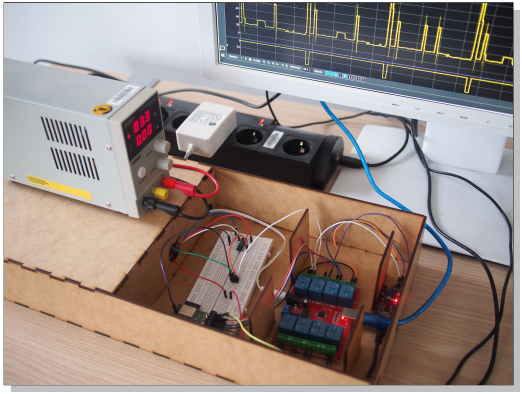
w.r.t.
Flusseinschränkungen im **Clock-Tree-Rekonfigurationsgraphen**
alle Zeiten aufsummiert ergeben verfügbare **Gesamtzeit**
jeder Task ist bis zu seiner **Deadline** beendet
...



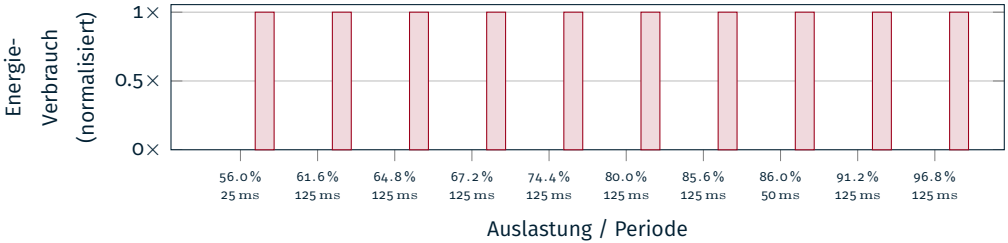
Evaluation

Entwicklung eines eigenen Energiemesssetups:



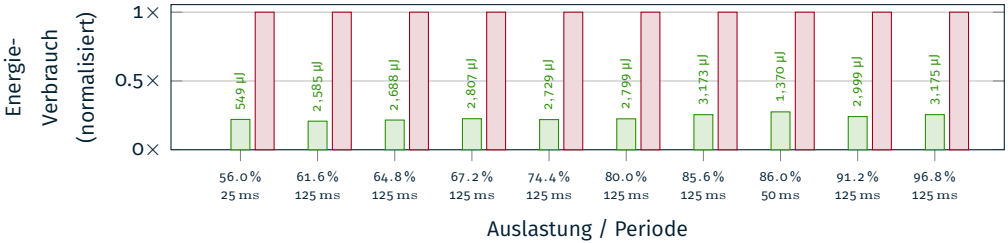


Evaluation: Vergleich generierter Aufgabensets



Anwendung ohne Optimierungen

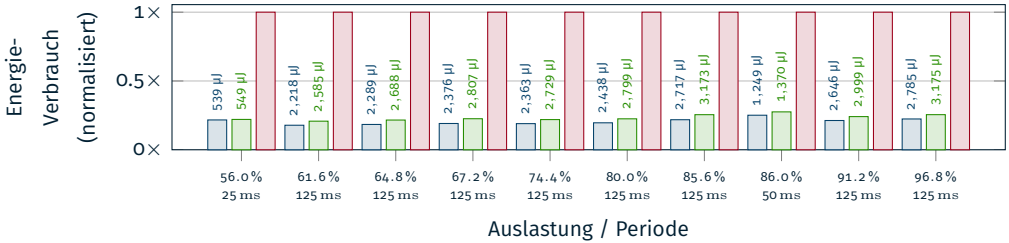
Evaluation: Vergleich generierter Aufgabensets



Vorhersage des Energieverbrauchs

Anwendung ohne Optimierungen

Evaluation: Vergleich generierter Aufgabensets

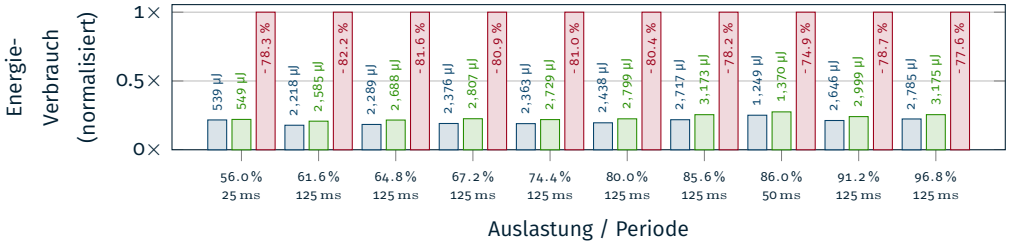


Anwendung mit Optimierungen

Vorhersage des Energieverbrauchs

Anwendung ohne Optimierungen

Evaluation: Vergleich generierter Aufgabensets



Anwendung mit Optimierungen

Vorhersage des Energieverbrauchs

Anwendung ohne Optimierungen

Zusammenfassung

1. Ansätze, die nur die CPU betrachten, …
   - missachten den Energiebedarf von **Geräten**
   - ignorieren **Abhängigkeiten der Geräte** und des **Clock Tree**s
2. **Keine Garantien** feedback-basierender Ansätze
3. fehlende **Rekonfigurationskosten**

Zusammenfassung

1. Ansätze, die nur die CPU betrachten, …
   - missachten den Energiebedarf von **Geräten**
   - ignorieren **Abhängigkeiten der Geräte** und des **Clock Tree**s

   ✓ **Modell, das Gerätezustände abhängig vom Clock Tree abbildet**
2. **Keine Garantien** feedback-basierender Ansätze

3. fehlende **Rekonfigurationskosten**

Zusammenfassung

1. Ansätze, die nur die CPU betrachten, …
   - missachten den Energiebedarf von **Geräten**
   - ignorieren **Abhängigkeiten der Geräte** und des **Clock Tree**s

   ✓ **Modell, das Gerätezustände abhängig vom Clock Tree abbildet**
2. **Keine Garantien** feedback-basierender Ansätze

   ✓ **Garantien durch statischen Ansatz**
3. fehlende **Rekonfigurationskosten**

Zusammenfassung

1. Ansätze, die nur die CPU betrachten, …
   - missachten den Energiebedarf von **Geräten**
   - ignorieren **Abhängigkeiten der Geräte** und des **Clock Tree**s

   ✓ **Modell, das Gerätezustände abhängig vom Clock Tree abbildet**
2. **Keine Garantien** feedback-basierender Ansätze

   ✓ **Garantien durch statischen Ansatz**
3. fehlende **Rekonfigurationskosten**

   ✓ **Rekonfigurationskosten des Clock Trees Teil der Optimierung**

Weiterführende Forschung

- Tasks mit variabler Release-Zeit / Deadline
  Paper @ ECRTS'23 [2]
- Einbeziehung von Unterbrechungen
  Best Paper @ ECRTS'24 [1]

Weiterführende Forschung

- Tasks mit variabler Release-Zeit / Deadline
  Paper @ ECRTS'23 [2]
- Einbeziehung von Unterbrechungen
  Best Paper @ ECRTS'24 [1]
- *Aktuelle Forschung*:
  - Umsortierung von Aufgaben
  - Energiemodellierung / Hardware-nahe Untersuchungen

Fragen?

Meine Lehrstuhl-Website: `https://sys.cs.fau.de/~dengler`

Publikationen und Source Code

Eva Dengler et al. **“FusionClock: Energy-Optimal Clock-Tree Reconfigurations for Energy-Constrained Real-Time Systems”.** In: *ECRTS ’23*. DOI: 10.4230/LIPIcs.ECRTS.2023.6

Quellcode & Artefakt: `https://gitos.rrze.fau.de/fusionclock`

Eva Dengler and Peter Wägemann. **“Crêpe: Clock-Reconfiguration–Aware Preemption Control in Real-Time Systems with Devices”.** In: *ECRTS ’24*. DOI: 10.4230/LIPIcs.ECRTS.2024.10

Quellcode & Artefakt: `https://gitos.rrze.fau.de/crepe`