



Institut für Mikroelektronische Systeme



Leibniz
Universität
Hannover

Methoden zur applikationsspezifischen Verlustleistungsoptimierung für eingebettete Prozessoren

Sebastian Hesselbarth, Holger Blume

Institut für Mikroelektronische Systeme, Leibniz Universität Hannover
{hesselbarth, blume}@ims.uni-hannover.de



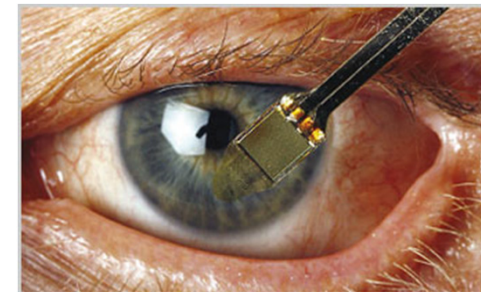
15. ITG-Fachtagung für Elektronische Medien, 26.-27. Februar 2013, Dortmund

Überblick

- Motivation
- Modellierung der Verlustleistung
 - Hybrid Functional Level/Instruction Level Power Analysis
 - Erweiterung der Hybrid FLPA/ILPA
- Modellierung für Prozessoren im Entwurfsstadium
 - Vorgehensweise
 - Identifikation von Einflussfaktoren auf Verlustleistung
- Emulation der Verlustleistungsmodelle
 - Integration und Aufbau des Moduls zur Bestimmung der Verlustleistung
- Evaluation der Verlustleistungsmodelle
- Zusammenfassung

Anforderungen an mobile Anwendungen

- Steigende Leistungsanforderungen an mobile Anwendungen
 - Sinkende Strukturgrößen erlauben leistungsfähige Multiprozessor-Chips
- Batteriebetrieb begrenzt Laufzeit drastisch
- Verlustleistung muss stärker beachtet werden
 - Optimierung von Verlustleistung und Energieverbrauch
- Anwendungsfelder
 - Mobile Endgeräte
 - Hörgeräte
 - Medizinische Implantate



Entwicklung für mobile Anwendungen

- Optimierung der Leistungsfähigkeit während der Entwicklung
 - HW/SW-Codesign mit funktionaler Emulation auf FPGA üblich
 - Abbildung der *Funktion* des Prozessors auf FPGA mit geringerer Taktfrequenz (Faktor ca. 0,1-0,5)
 - Gleichzeitige Entwicklung von Hardware und Software
 - Vergrößert Optimierungspotential
 - Verkürzung der Time-to-Market

Entwicklung für mobile Anwendungen

- Optimierung der Leistungsfähigkeit während der Entwicklung
 - HW/SW-Codesign mit funktionaler Emulation auf FPGA üblich
 - Abbildung der *Funktion* des Prozessors auf FPGA mit geringerer Taktfrequenz (Faktor ca. 0,1-0,5)
 - Gleichzeitige Entwicklung von Hardware und Software
 - Vergrößert Optimierungspotential
 - Verkürzung der Time-to-Market
- Optimierung der Verlustleistung erst nach Fertigung des ASIC
 - Keine Messung am Chip möglich
 - Nur sequentielle Optimierung von Hardware und Software
 - Verringertes Optimierungspotential
 - Verlängerung der Time-to-Market

Verlustleistungsbestimmung während des Entwurfs

- Dynamische Verlustleistung

- Elektrische Kapazität C
- Versorgungsspannung V
- Betriebsfrequenz f
- Schaltaktivität α (Häufigkeit der Umladevorgänge)

$$P_{dyn} = \alpha \cdot C \cdot V^2 \cdot f$$

Verlustleistungsbestimmung während des Entwurfs

- Dynamische Verlustleistung
 - Elektrische Kapazität C
 - Versorgungsspannung V
 - Betriebsfrequenz f
 - Schaltaktivität α (Häufigkeit der Umladevorgänge)
- Simulation der technologieabhängigen Gatter-Netzliste (ASIC)
 - Extraktion der Schaltaktivität
 - Verlustleistungssimulation
 - Sehr langsame Simulationsgeschwindigkeit (10-50Hz)
- Ermittlung der Verlustleistung ganzer Anwendungen unmöglich
 - Kaum Beachtung realer Eingangsdaten

$$P_{dyn} = \alpha \cdot C \cdot V^2 \cdot f$$

Verlustleistungsbestimmung während des Entwurfs

- Dynamische Verlustleistung
 - Elektrische Kapazität C
 - Versorgungsspannung V
 - Betriebsfrequenz f
 - Schaltaktivität α (Häufigkeit der Umladevorgänge)
 - Simulation der technologieabhängigen Gatter-Netzliste (ASIC)
 - Extraktion der Schaltaktivität
 - Verlustleistungssimulation
 - Sehr langsame Simulationsgeschwindigkeit (10-50Hz)
 - Ermittlung der Verlustleistung ganzer Anwendungen unmöglich
 - Kaum Beachtung realer Eingangsdaten
- **Abstraktion von der zeitintensiven Gatter-Ebene erforderlich**

$$P_{dyn} = \alpha \cdot C \cdot V^2 \cdot f$$

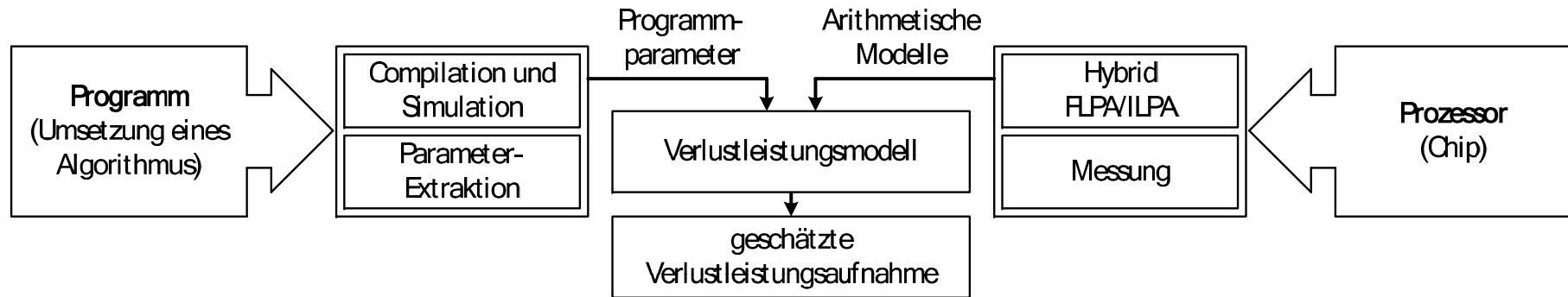
Modellierung der Verlustleistung

- Differenzierung auf funktionaler Ebene (FLPA)
 - Funktion der Einheit bestimmt Verlustleistung
 - z.B. Anzahl der ausgeführten Instruktionen, Cache-Misses, Speicherzugriffe
- Differenzierung auf Instruktionsebene (ILPA)
 - Ausgeführte Instruktion bestimmt Verlustleistung
 - z.B. Addition/Subtraktion, Multiplikation, Laden/Speichern

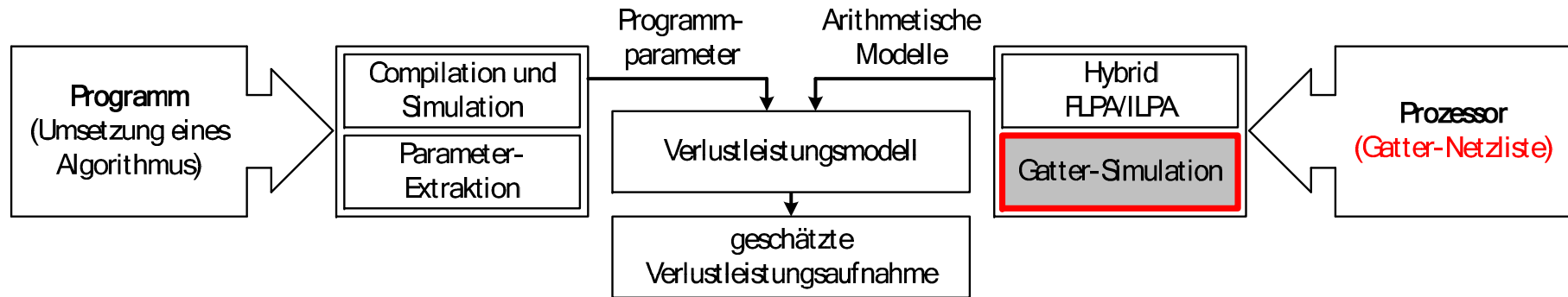
Modellierung der Verlustleistung

- Differenzierung auf funktionaler Ebene (FLPA)
 - Funktion der Einheit bestimmt Verlustleistung
 - z.B. Anzahl der ausgeführten Instruktionen, Cache-Misses, Speicherzugriffe
- Differenzierung auf Instruktionsebene (ILPA)
 - Ausgeführte Instruktion bestimmt Verlustleistung
 - z.B. Addition/Subtraktion, Multiplikation, Laden/Speichern
- Hybrid Functional Level/Instruction Level Power Analysis
 - Hybride Modellierung vereint FLPA und ILPA
 - Modellbildung durch *Messung* der Verlustleistung am *realisierten Chip* durch gezielte Anregung des Prozessors

Erweiterung der Hybrid FLPA/ILPA

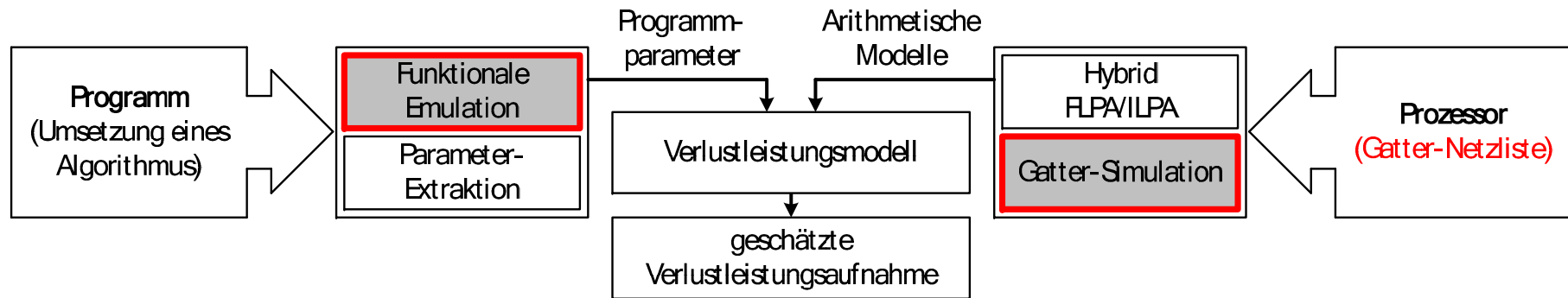


Erweiterung der Hybrid FLPA/ILPA



- Bildung der arithmetischen Modelle
 - Messung am noch nicht realisierten Chip unmöglich
 - Gatter-Simulation mit kurzen Programmsequenzen
 - Extraktion instruktionsspezifischer Schaltaktivität einzelner Module

Erweiterung der Hybrid FLPA/ILPA

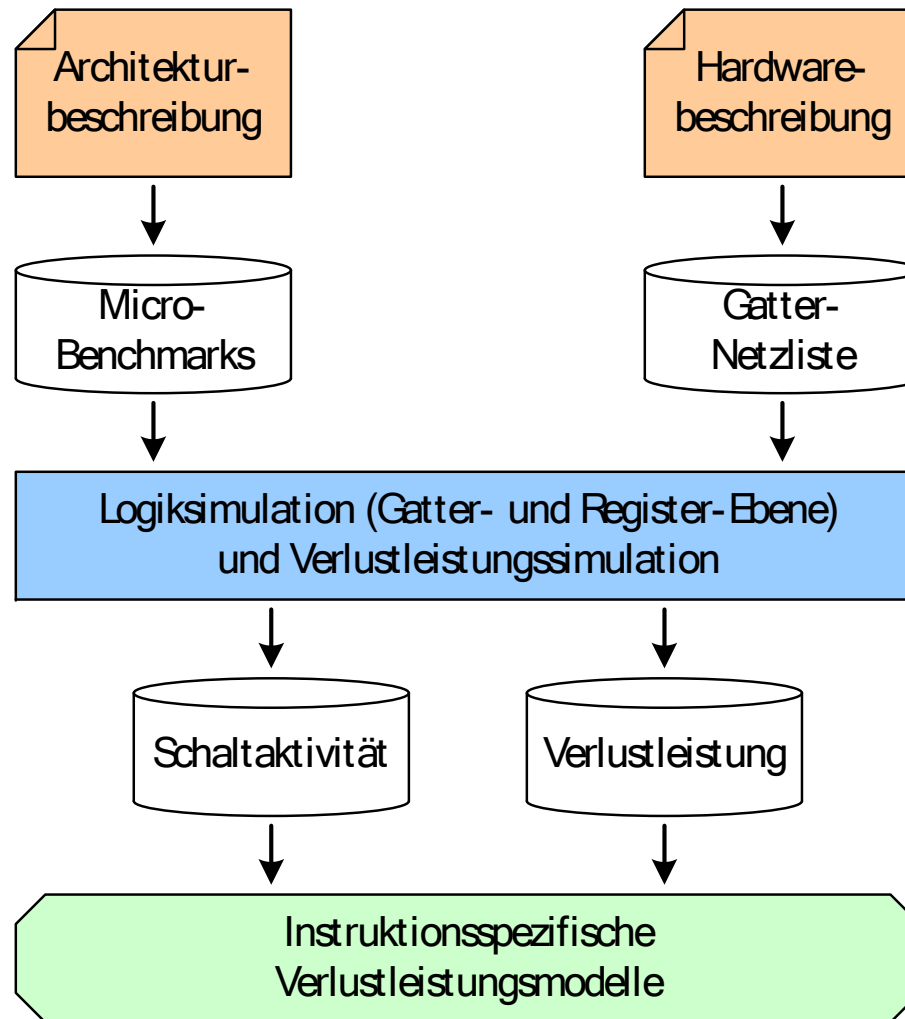


- Bildung der arithmetischen Modelle
 - Messung am noch nicht realisierten Chip unmöglich
 - Gatter-Simulation mit kurzen Programmsequenzen
 - Extraktion instruktionsspezifischer Schaltaktivität einzelner Module
- Integration in die funktionale Emulation auf FPGAs
 - Nutzung der Emulation zur Anwendung der arithmetischen Modelle
 - Großer Geschwindigkeitsgewinn (50-150MHz)
 - Ermöglicht Untersuchung ganzer Anwendungen

Überblick

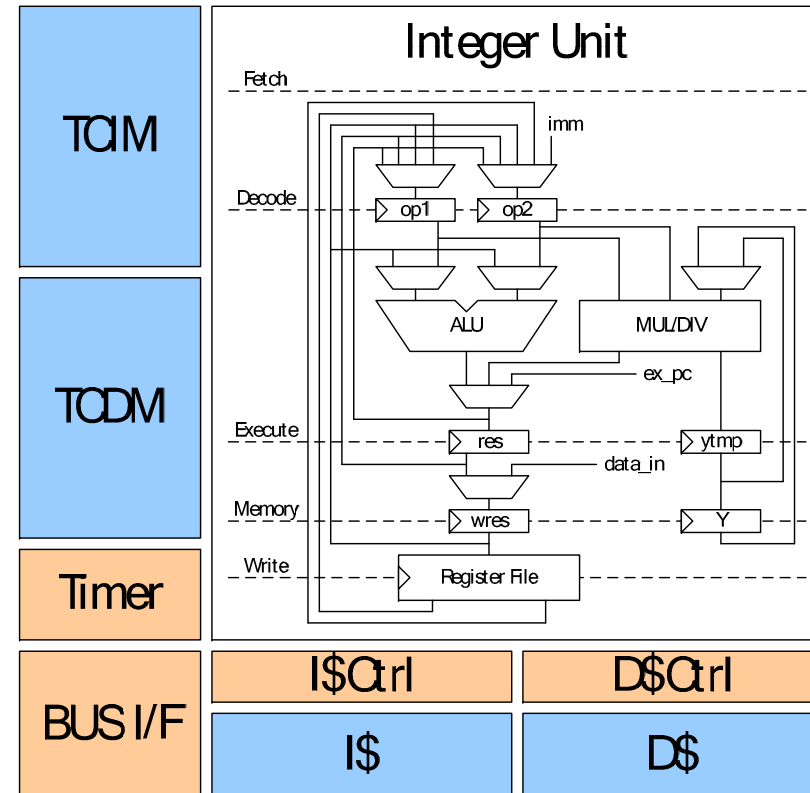
- Motivation
- Modellierung der Verlustleistung
 - Hybrid Functional Level/Instruction Level Power Analysis
 - Erweiterung der Hybrid FLPA/ILPA
- Modellierung für Prozessoren im Entwurfsstadium
 - Vorgehensweise
 - Identifikation von Einflussfaktoren auf Verlustleistung
- Emulation der Verlustleistungsmodelle
 - Integration und Aufbau des Moduls zur Bestimmung der Verlustleistung
- Evaluation der Verlustleistungsmodelle
- Zusammenfassung

Methodik zur Verlustleistungsmodellierung



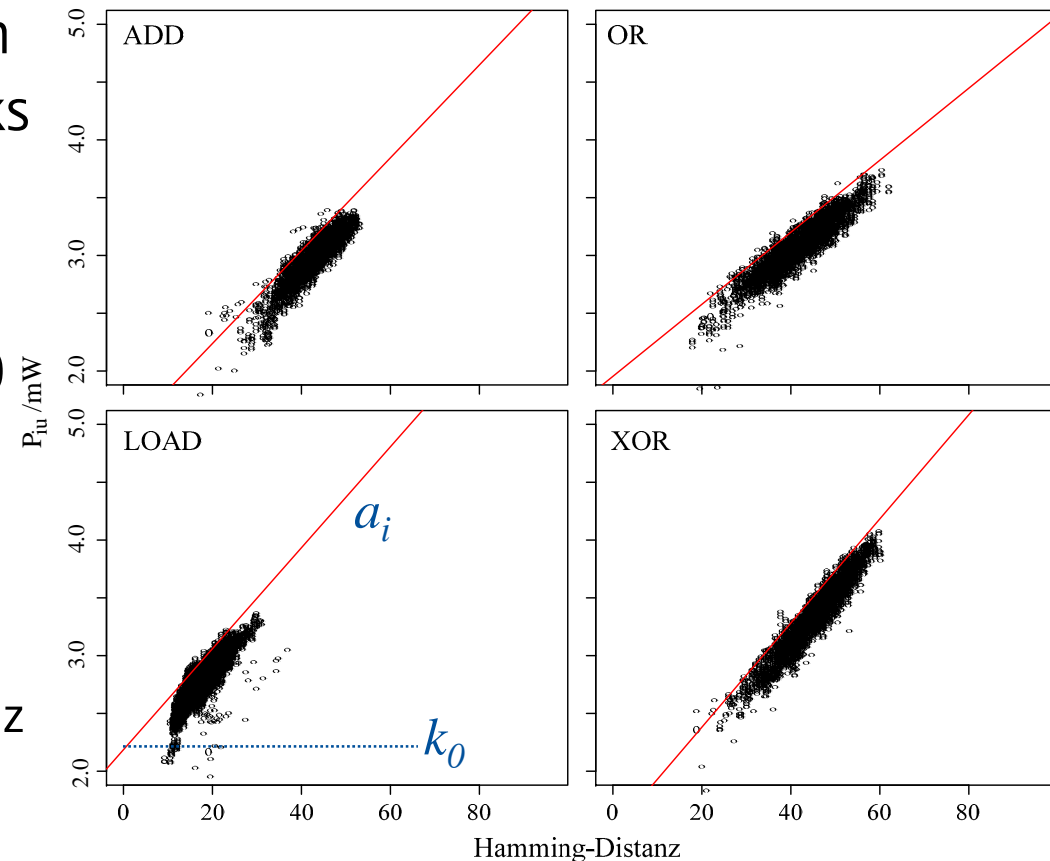
LEON-2 RISC-Prozessor

- Gaisler Research (GPL)
- Sparc v8 Architektur
- 5-stufige Pipeline
- Instruktions-/Daten-Cache
- Lokale Speicher für Instruktionen und Daten
- FPU, Coprozessor (optional)
- 40nm LP Standardzell-Bibliothek für 500 MHz



Instruktionspezifische Verlustleistung

- Einmalige Simulation von kurzen Mikro-Benchmarks
- Wiederholte Ausführung einer Instruktion
- Ausführungszeit ca. 5-10 Minuten/Instruktion
- Gut parallelisierbar
- Maß für Schaltaktivität
 - zeitliche Hamming-Distanz
- Modellbildung durch Lineare Regressions-Analyse

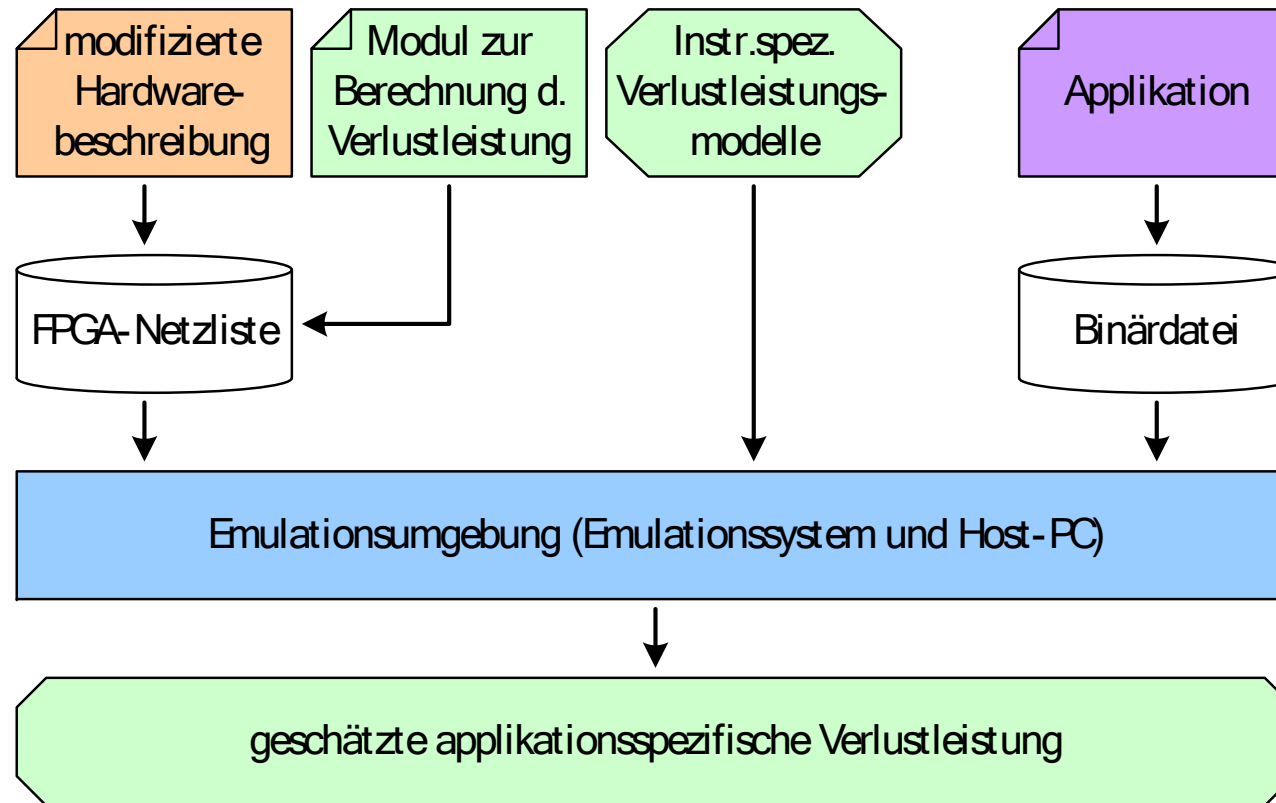


$$\tilde{P}_{IU,I}(t) = k_0 + \sum_i a_i \cdot HD(\mathbf{x}_i, t)$$

Überblick

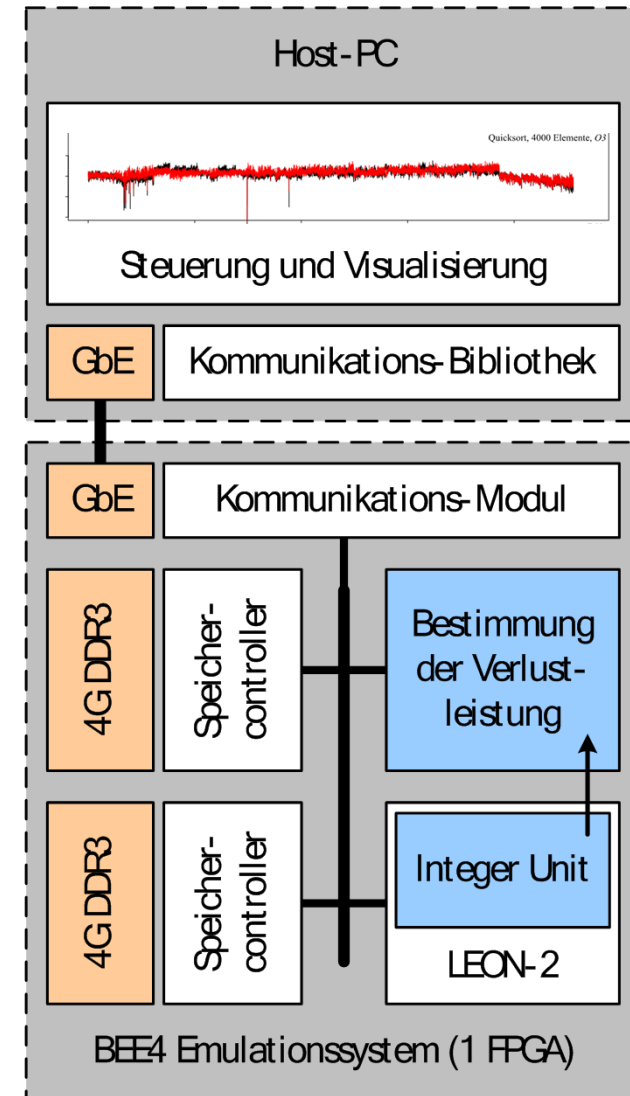
- Motivation
- Modellierung der Verlustleistung
 - Hybrid Functional Level/Instruction Level Power Analysis
 - Erweiterung der Hybrid FLPA/ILPA
- Modellierung für Prozessoren im Entwurfsstadium
 - Vorgehensweise
 - Identifikation von Einflussfaktoren auf Verlustleistung
- **Emulation der Verlustleistungsmodelle**
 - Integration und Aufbau des Moduls zur Bestimmung der Verlustleistung
- Evaluation der Verlustleistungsmodelle
- Zusammenfassung

Anwendung der Verlustleistungsmodelle



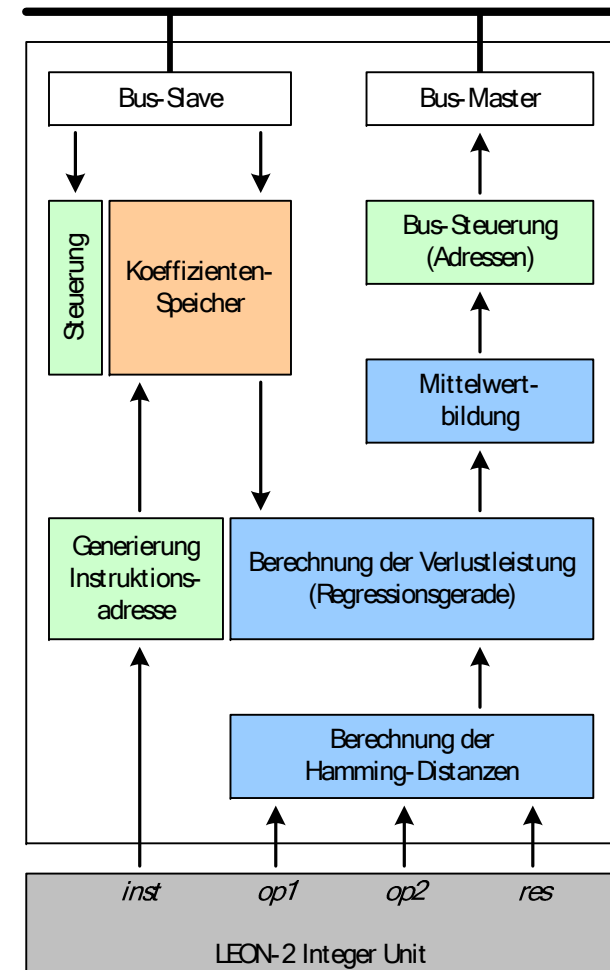
Aufbau der Emulationsumgebung

- BEEcube BEE4 Emulationssystem
 - 4x Xilinx Virtex-6 LX550T
 - Je 2x 4GB DDR3 SDRAM-Module
 - Je 1x Gigabit Ethernet (GbE)
- FPGA
 - LEON-2 Prozessor
 - Modul zur Bestimmung der Verlustleistung
 - Kommunikationsmodul
- Standard Host-PC
 - Datentransfer
 - Emulations-Steuerung
 - Visualisierung des zeitlichen Verlaufes der Verlustleistung



Modul zur Berechnung der Verlustleistung

- Durchführen aller Berechnungen
 - Geringfügige Modifikation der LEON-2 Hardware-Beschreibung erforderlich
 - Zugriff auf interne Signale des Prozessors
- Speicher für Regressions-Koeffizienten
- Speichern der Ergebnisse in SDRAM
- Geringer Ressourcenbedarf
 - 1,4% der LUTs des FPGAs bzw. 11,6% der LUTs des Prozessorkerns
 - 2,5% der BlockRAMs des FPGAs
- Kein Einfluss auf erreichbare Emulationsgeschwindigkeit



Überblick

- Motivation
- Modellierung der Verlustleistung
 - Hybrid Functional Level/Instruction Level Power Analysis
 - Erweiterung der Hybrid FLPA/ILPA
- Modellierung für Prozessoren im Entwurfsstadium
 - Vorgehensweise
 - Identifikation von Einflussfaktoren auf Verlustleistung
- Emulation der Verlustleistungsmodelle
 - Integration und Aufbau des Moduls zur Bestimmung der Verlustleistung
- Evaluation der Verlustleistungsmodelle
- Zusammenfassung

Anwendung der Verlustleistungsmodelle

- Auswahl typischer Algorithmen für eingebettete Systeme
- Einfluss von Entscheidungen auf Applikationsebene
 - Algorithmische Varianten und Compiler-Optimierungen

Benchmark	#Zyklen	%MAE	NRMSE
Quicksort, <i>O</i> _s	9.401.600	3,01%	6,89%
Quicksort, <i>O</i> ₃	9.111.040	2,15%	4,68%
Lowpass 3x3, Variante A, <i>O</i> _s	2.063.360	3,92%	10,30%
Lowpass 3x3, Variante A, <i>O</i> ₃	1.700.480	4,36%	6,75%
Lowpass 3x3, Variante B, <i>O</i> ₃	1.547.520	3,82%	9,50%
<i>Durchschnitt (alle Benchmarks)</i>		3,86%	7,88%

Anwendung der Verlustleistungsmodelle

- Auswahl typischer Algorithmen für eingebettete Systeme
- Einfluss von Entscheidungen auf Applikationsebene
 - Algorithmische Varianten und Compiler-Optimierungen

Benchmark	#Zyklen	%MAE	NRMSE
Quicksort, <i>O</i> _s	9.401.600	3,01%	6,89%
Quicksort, <i>O</i> ₃	9.111.040	2,15%	4,68%
Lowpass 3x3, Variante A, <i>O</i> _s	2.063.360	3,92%	10,30%
Lowpass 3x3, Variante A, <i>O</i> ₃	1.700.480	4,36%	6,75%
Lowpass 3x3, Variante B, <i>O</i> ₃	1.547.520	3,82%	9,50%
<i>Durchschnitt (alle Benchmarks)</i>		3,86%	7,88%

Anwendung der Verlustleistungsmodelle

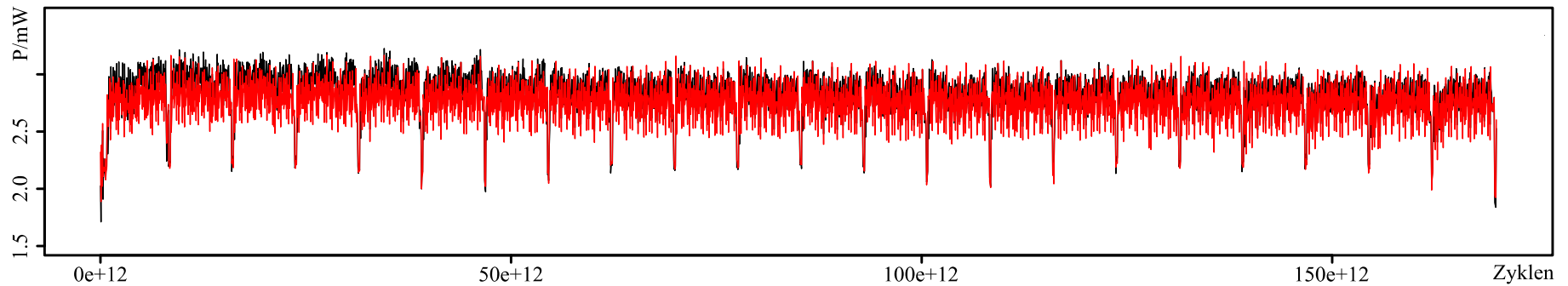
- Auswahl typischer Algorithmen für eingebettete Systeme
- Einfluss von Entscheidungen auf Applikationsebene
 - Algorithmische Varianten und Compiler-Optimierungen

Benchmark	#Zyklen	%MAE	NRMSE
Quicksort, 0s	9.401.600	3,01%	6,89%
Quicksort, 03	9.111.040	2,15%	4,68%
Lowpass 3x3, Variante A, 0s	2.063.360	3,92%	10,30%
Lowpass 3x3, Variante A, 03	1.700.480	4,36%	6,75%
Lowpass 3x3, Variante B, 03	1.547.520	3,82%	9,50%
<i>Durchschnitt (alle Benchmarks)</i>		3,86%	7,88%

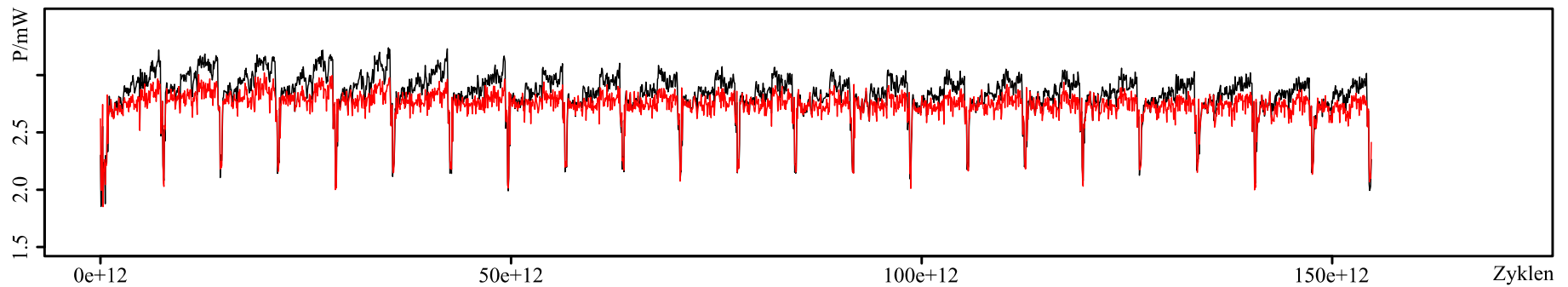
-17% ↓
-9% ↓

Ergebnisse: Einfluss von algorithmischen Varianten

- Beispiel: Lowpass-Filter 3x3
- Variante A: Rechenlastig (Shift)

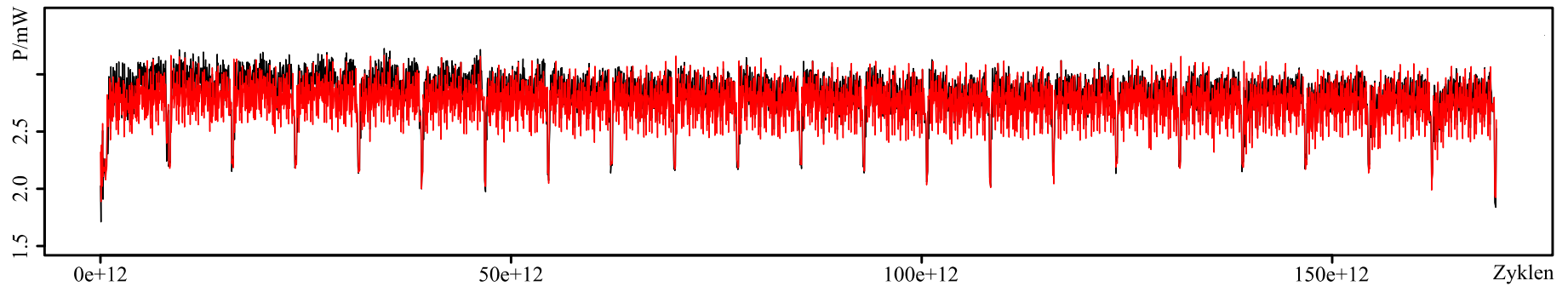


- Variante B: Speicherlastig (Load Byte)

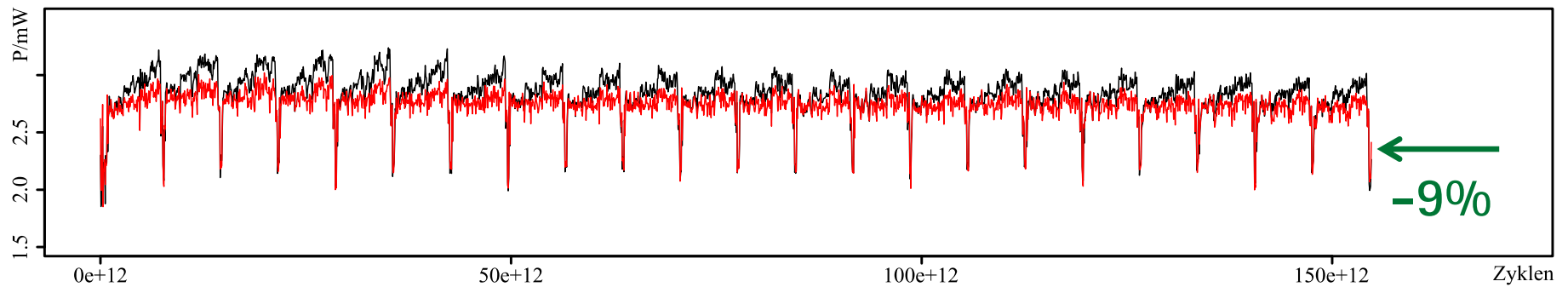


Ergebnisse: Einfluss von algorithmischen Varianten

- Beispiel: Lowpass-Filter 3x3
- Variante A: Rechenlastig (Shift)



- Variante B: Speicherlastig (Load Byte)



Zusammenfassung

- Notwendigkeit zur Optimierung der Verlustleistung
- Methodik zur Verlustleistungsmodellierung
 - Erweiterung der Hybrid FLPA/ILPA
 - Prozessoren im Entwurfsstadium
 - Integration in die funktionale Emulation
- Evaluation der Methodik an realem Beispiel
 - Signifikanter Unterschied im Energiebedarf