

# Fachgebietsvorstellung 2010

**Professur VLSI-EDA**

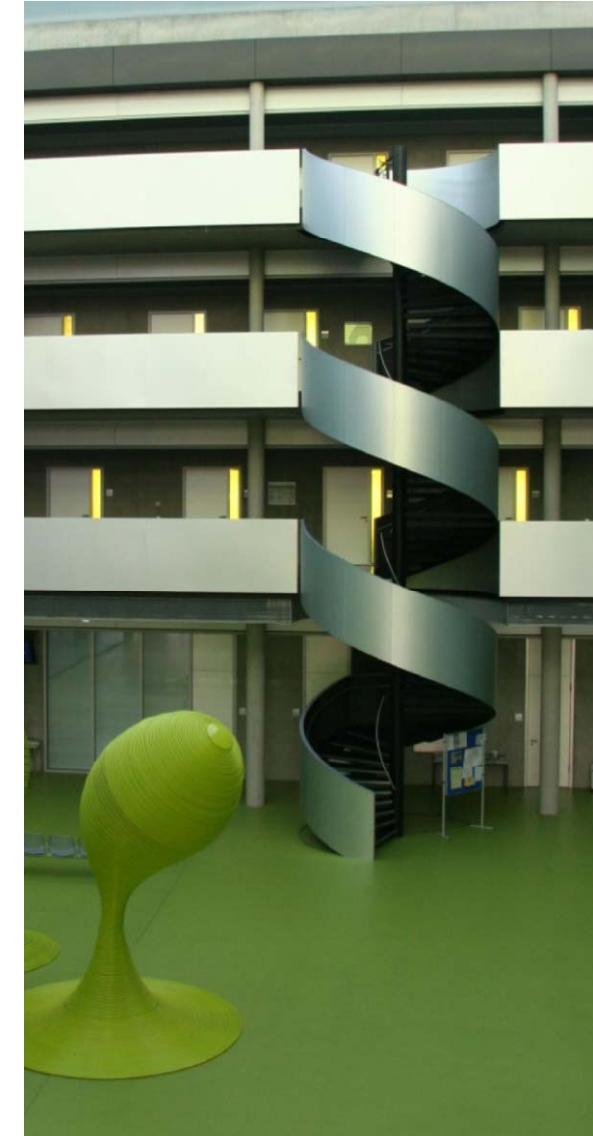
*Rainer G. Spallek*

TU Dresden, 28.07.2010



## Gliederung

- 1 Übersicht
- 2 Vorgehensweise
- 3 Lehrveranstaltungen
- 4 Forschungsschwerpunkte
- 5 Beleg- und Diplomarbeiten



# 1 Übersicht

## Was ist Technische Informatik?

### Quellen und Ursprung:

- Elektrotechnik, Elektronik, Digitaltechnik, Systemtheorie, Signaltheorie
- Mathematik (diskrete), Logik, Automatentheorie, formale Systeme
- Maschinenbau, Robotik, Sensorik, Aktuatorik, Automatisierung

### Vertritt die Gebiete:

- Rechnerarchitektur, HPC, Netzwerk- und Kommunikationstechnik
- Eingebettete Systeme -> Post-PC-Ära, System-on-a-Chip
- Systementwurf, Entwurfsmethodik, formale Beschreibungssprachen
- Modellierung und Simulation, Test und Diagnose, Verifikation, Validation

## Was ist Technische Informatik nicht?

## 2 Vorgehensweise

Studienordnung/Prüfungsordnung genau lesen!

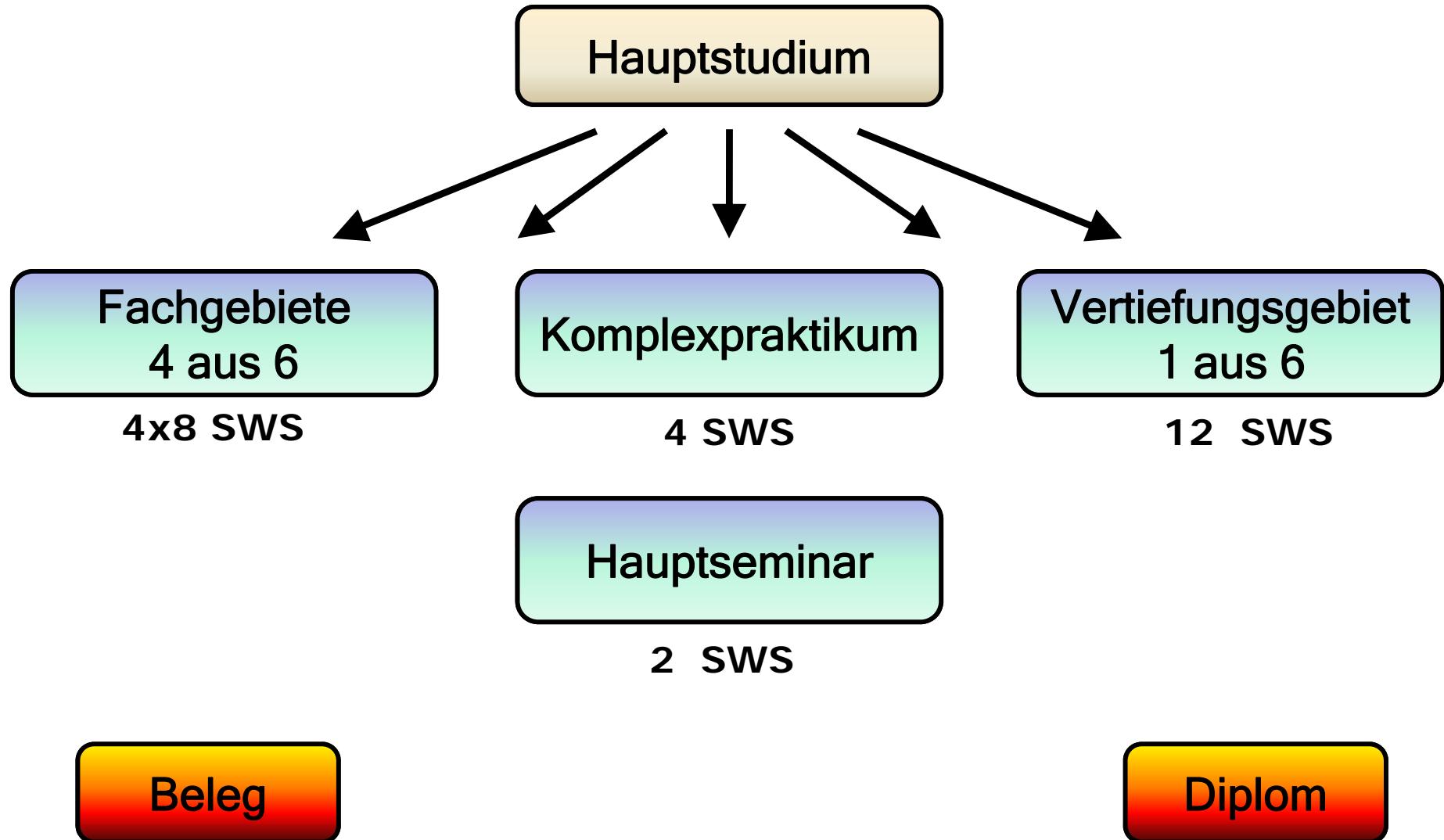
Studienplan für das Hauptstudium?

Verantwortlicher für das Fachgebiet/Vertiefungsgebiet

„Technische Informatik“:

Prof. Dr.-Ing. habil. Rainer G. Spallek

Fachgebietsprüfung/Vertiefungsprüfung (mündlich) dort vereinbaren,  
wo die meisten SWS liegen.



## 3 Lehrveranstaltungen

Lehrveranstaltung	kurz	VÜP	Sem.
Schaltkreis- und Systementwurf	SSE	202	SS
Programmierbare Schaltkreise	PLD	202	WS
Computertechnik	CT	220	SS
Computerarithmetik	CA	220	WS
Test und Diagnose digitaler Systeme	TDS	220	WS
Komplexpraktikum Prozessorentwurf	KPP	004	WS/SS
Hauptseminar Technische Informatik	HTI	020	WS/SS

Folgende Praktika auch ohne Vorlesung:

Lehrveranstaltung	kurz	SWS	Sem.
Schaltkreis- und Systementwurf	SSE	2	SS
Komplexpraktikum Prozessorentwurf	KPP	4, 2*	WS/SS

\* Komplexpraktikum auch mit 2 SWS möglich

Folgende Vorlesungen auch ohne Praktika:

Lehrveranstaltung	kurz	SWS	Sem.
Schaltkreis- und Systementwurf	SSE	2	SS
Programmierbare Schaltkreise	PLD	2	WS

5.

HTI

PLD

CA

TDS

WS

6.

HTI

SSE

CT

SS

7.

HTI

KPP

PLD

CA

TDS

WS

8.

HTI

KPP

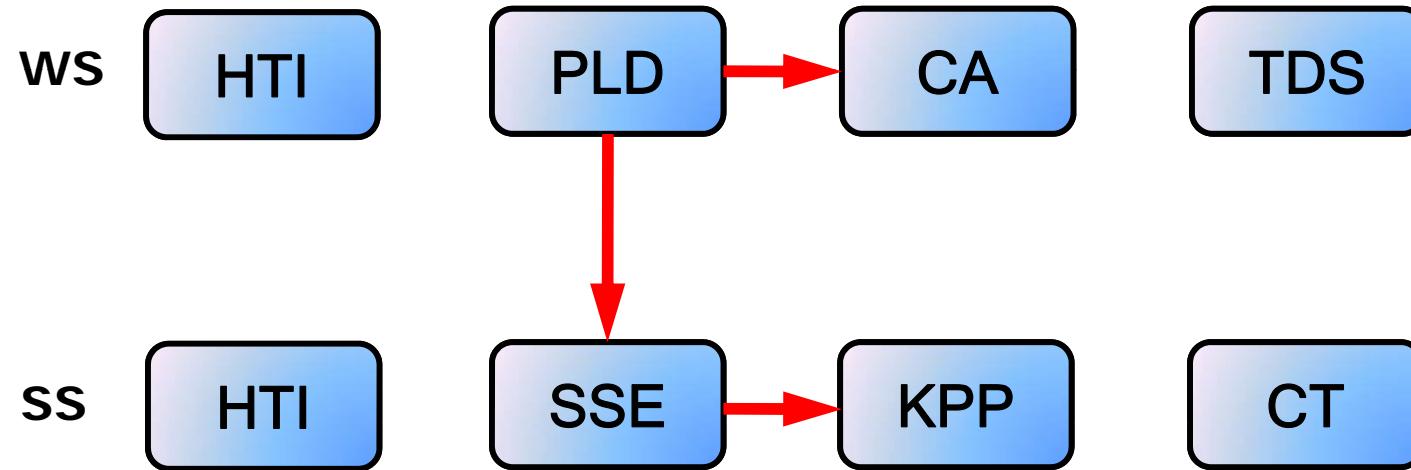
SSE

CT

SS

alternativ

## Lage und Abhängigkeiten



Abhängigkeiten: A → B    B nach oder gleichzeitig mit A

## 4 Forschungsschwerpunkte

### Prozessorarchitektur

- Entwurf und Applikation von Mikroprozessoren, SoC, MPSoC
- Robuste und zuverlässige Systeme, rekonfigurierbare Systeme

### Test und Diagnose

- Debugging von eingebetteten Systemen
- und Trace für Multi-Core SoC (MPSoC)

### Modellierung und Simulation von Prozessoren

- befehlsgenaue Simulation/Emulation
- taktzyklengenau auf Architekturebene

### Hochparalleles Rechnen in eingebetteten Systemen

- auf FPGA, GPU
- auf eingebetteten Multi-Core-Systemen

**Queens@TUD Weltrekord**<http://queens.inf.tu-dresden.de/>

**Q(26)=22.317.699.616.364.044** Möglichkeiten für die Platzierung von 26 sich nicht bedrohende Damen auf einem 26x26-Schachbrett (Laufzeit 10 Monate).  
→Lösung von 25.204.802 Teilaufgaben auf über 2000 Verarbeitungseinheiten

Das russische MC#-Projekt hat mit zwei Superrechnern aus der Top500-Liste (Juni 2009) unser Ergebnis bestätigt (Laufzeit 10 Monate).

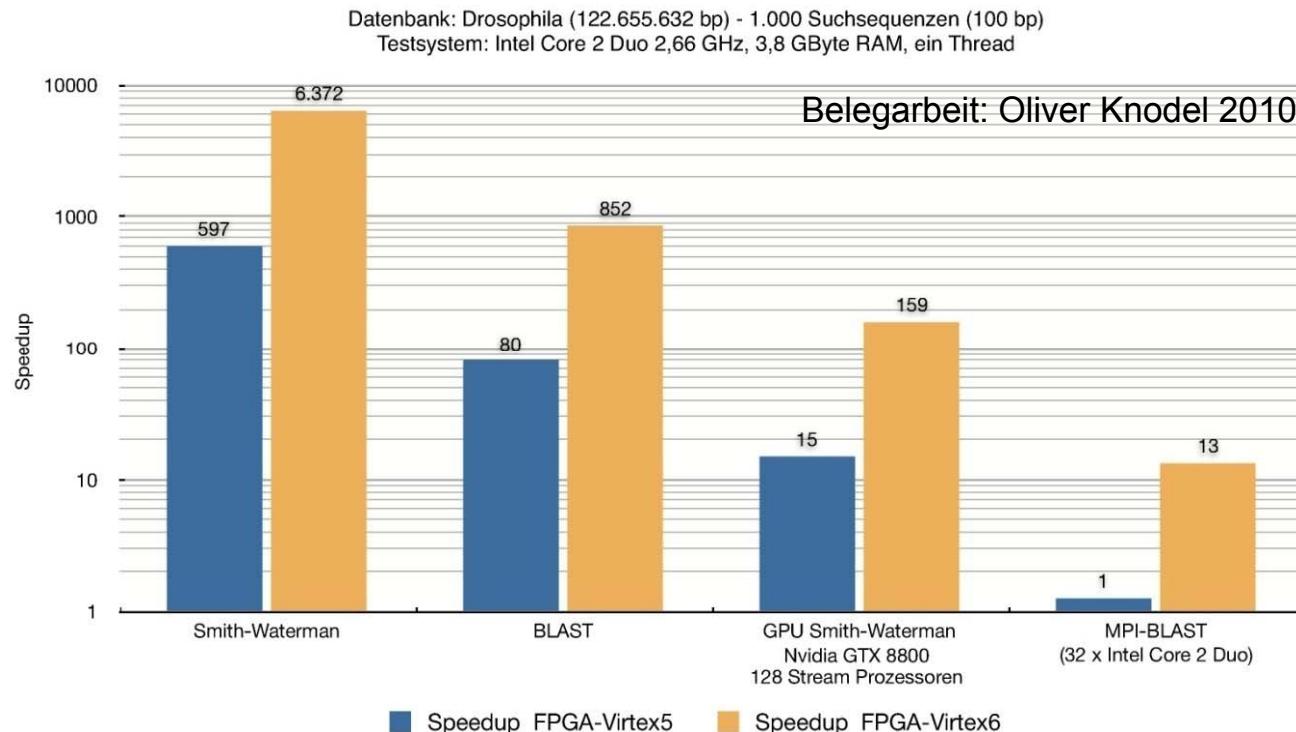
Das Projekt NQueens@Home versucht bereits seit Jahren zu einer Lösung zu kommen→offensichtlich aufgegeben.

**Q(27)=?**

## Hoch parallele Genom-Alignmentalgorithmen

Herausforderungen:

- Sequenzsuche in großen Datenbanken (z.B. Fruchtfliege:  $10^8$  Basenpaare)
- mehrere Millionen Suchsequenzen von 35 - 100 Basenpaaren
- Nichtübereinstimmungen gewisser Zahl können erlaubt sein



## Aufbereitung der Ereignismessungen des LHC

Large Hadron Collider: Teilchenbeschleuniger für Hadronen am  
Europäischen Kernforschungszentrum CERN (Genf)

Herausforderungen:

- 180000 Datenkanäle mit 16 Bit @40 MHz => 14,4 TByte/s
- Signalaufbereitung durch Filter, Klassifizierung und Selektion
- Tiefe DSP- und Datenpipelines, hoher Takt, hohe Parallelität

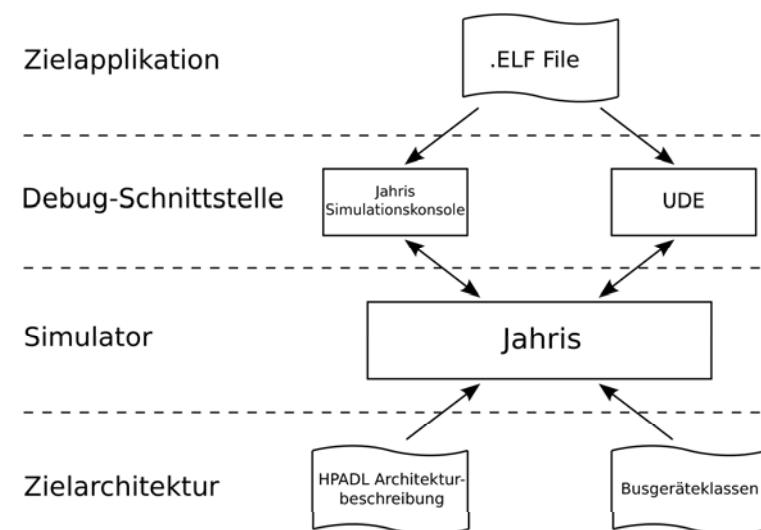
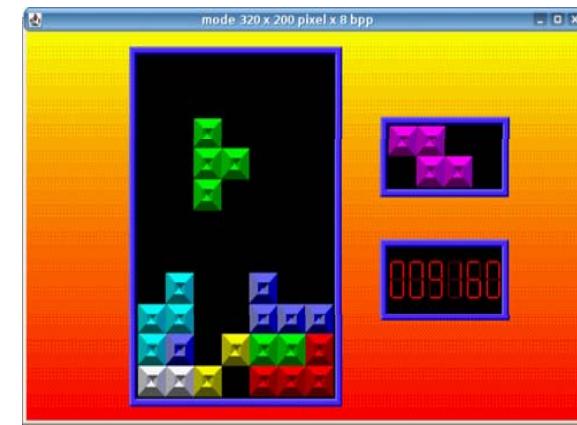
## Jahris – hochperformante ISA- und Full-System-Simulation

- befehlsgenaue Modellierung und Simulation
- retargierbar (Architekturbeschreibungssprache HPADL)
- Full-System-Simulation (ISA + IO-Geräte, Breakpoints, IRQs)
- Nutzung der JVM als Hostplattform für die Simulation  
(Plattformunabhängigkeit, Ausnutzung des Optimizer-Backends)

### Anwendungen

- ISA-Entwurf (Profiling, Design-Space-Exploration)
- Simulation eingebetteter Systeme in Echtzeit
- Testen und Debuggen von Software
- Backtrace, Kontext-Restaurierung

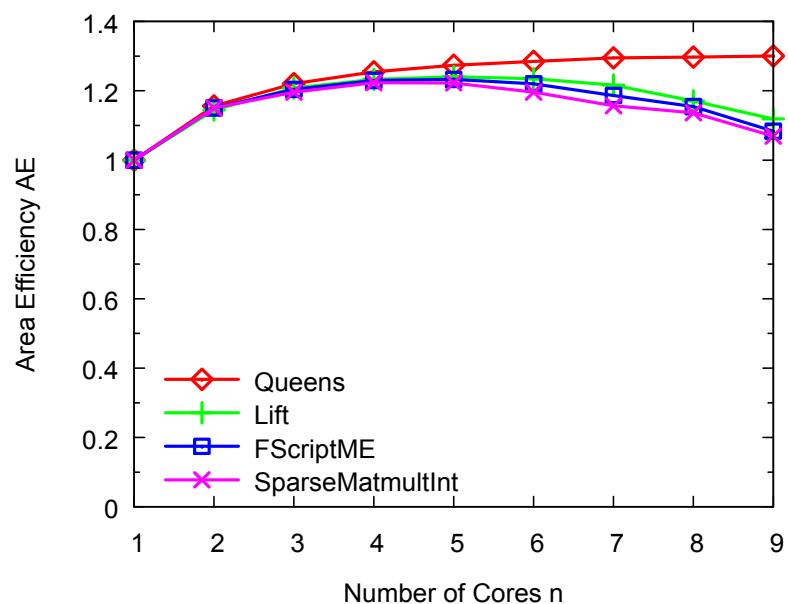
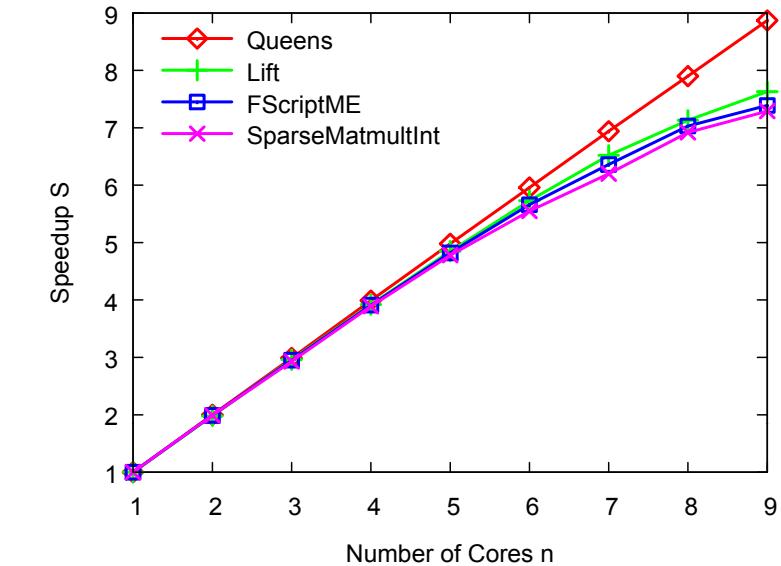
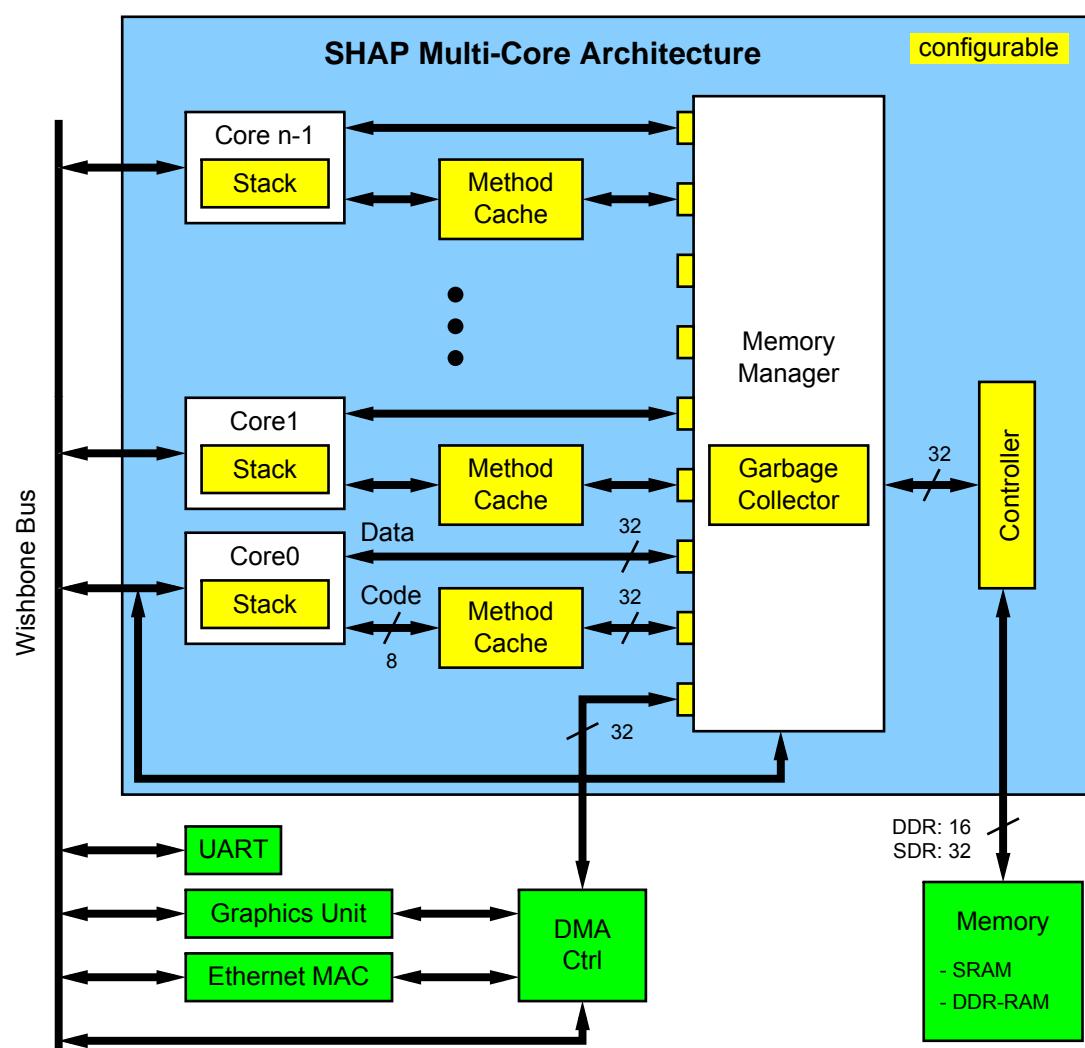
Jahris		LPC2106-B @40MHz		
	Zeit in s	MIPS	Zeit in s	MIPS
ts_sieve (2000)	1,192	302	13,881	25,97
ts_sieve (5000)	2,856	315	34,703	25,97
ts_sieve(10000)	5,570	323	69,405	25,97
ts_sieve(20000)	11,038	326	-	-



## SHAP Multi-Core Architecture (Java Bytecode Processor)

<http://shap.inf.tu-dresden.de/>

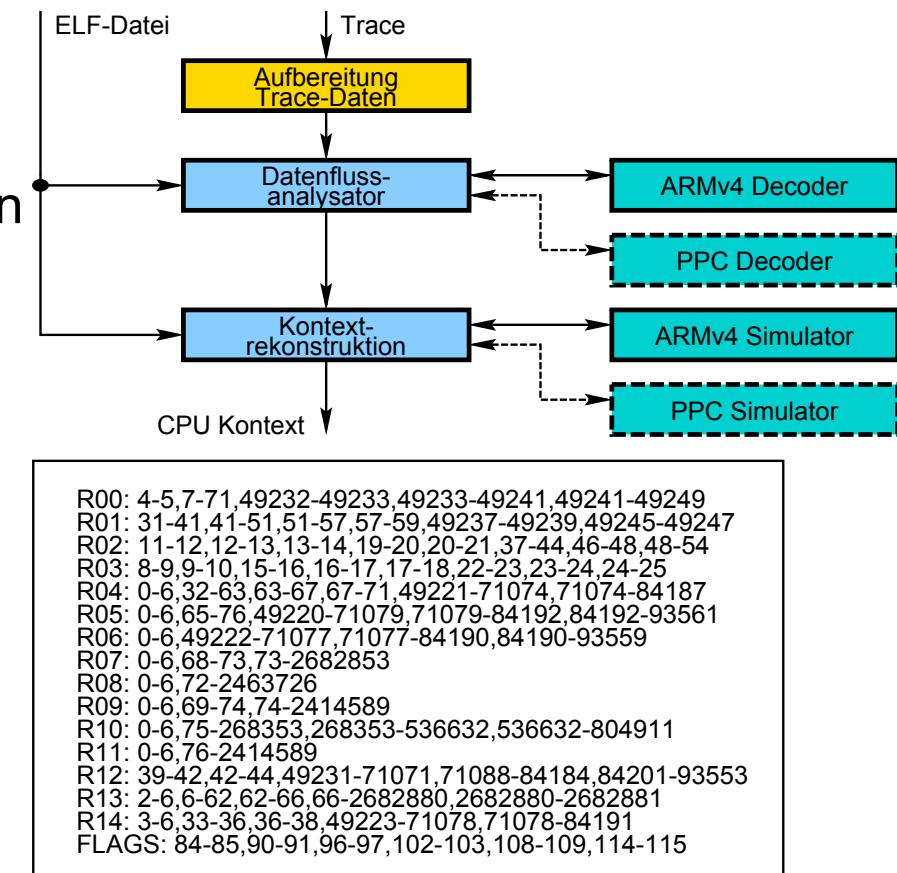
- Native execution of Java bytecode in hardware without underlying OS.
- Full-duplex memory bus with pipelined transactions for fast access to shared heap.
- Fast atomic operations for fully supported thread synchronization on a per-object basis.
- Multi-threaded cores with local on-chip stack and method cache.
- Exact and fully concurrent non-blocking garbage collector for automatic memory management.
- Round-Robin Scheduling.
- Load balancing implemented in Java.
- Design is synthesizable for a variable number of cores.
- Evaluation platform: ML505 Evaluation Board with Xilinx Virtex-5.



# Trace-basierte Rekonstruktion von Prozessorzuständen

## Systembeobachtung und Debugging

- nicht-beeinflussend (non-intrusive)
- unter Echtzeitbedingungen
- verschiedene Prozessorarchitekturen
- Multi-Core Systeme
- Simulator-Synchronisation



## 5 Beleg- und Diplomarbeiten

Beleg- und Diplomthemen zu den Forschungsschwerpunkten werden ständig vergeben (siehe auch WWW-Seiten der Professur bzw. durch persönliche Rücksprache mit den Themenbearbeitern).

Alternative Beleg- und Diplomthemen aus dem Bereich der Technischen Informatik sind ebenfalls möglich.

**Bitte versäumen Sie es nicht,  
uns auf aktuell verfügbare  
Beleg- und Diplomthemen anzusprechen!**