

Framework für den SoC Entwurf

Armin Jeyrani Mamegani
Betreuer: Prof. Dr. B. Schwarz

HAW Hamburg
Department Informatik

10.06.2008

Einleitung

- Sensor/Aktor Plattform
 - Regelung
 - z.B. Fahrzeugsteuerung



Einleitung

- **Sensor/Aktor Plattform**

- Entwicklung neuer Fahrzeuge, FAUST

- ➔ Verwenden einer einzigen Plattform für versch. Aufgaben

- nicht alles neu entwickeln

- ➔ Wiederverwendung von Modulen aus Libraries

- Echtzeitanforderungen müssen eingehalten werden

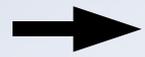
- ➔ Auslagerung einiger Tasks in parallele HW

- Ein System aus SW und HW

Einleitung

- **Handys/Smartphones**

- niedrige Performance, vor allem bei Grafik/Video



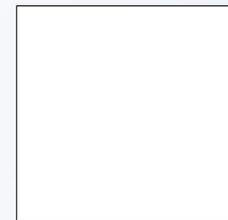
Auslagerung einiger Tasks in andere HW

- HW-Decoder, Grafik Prozessoren, ...

- geringes Bauvolumen



Alles auf einem Chip



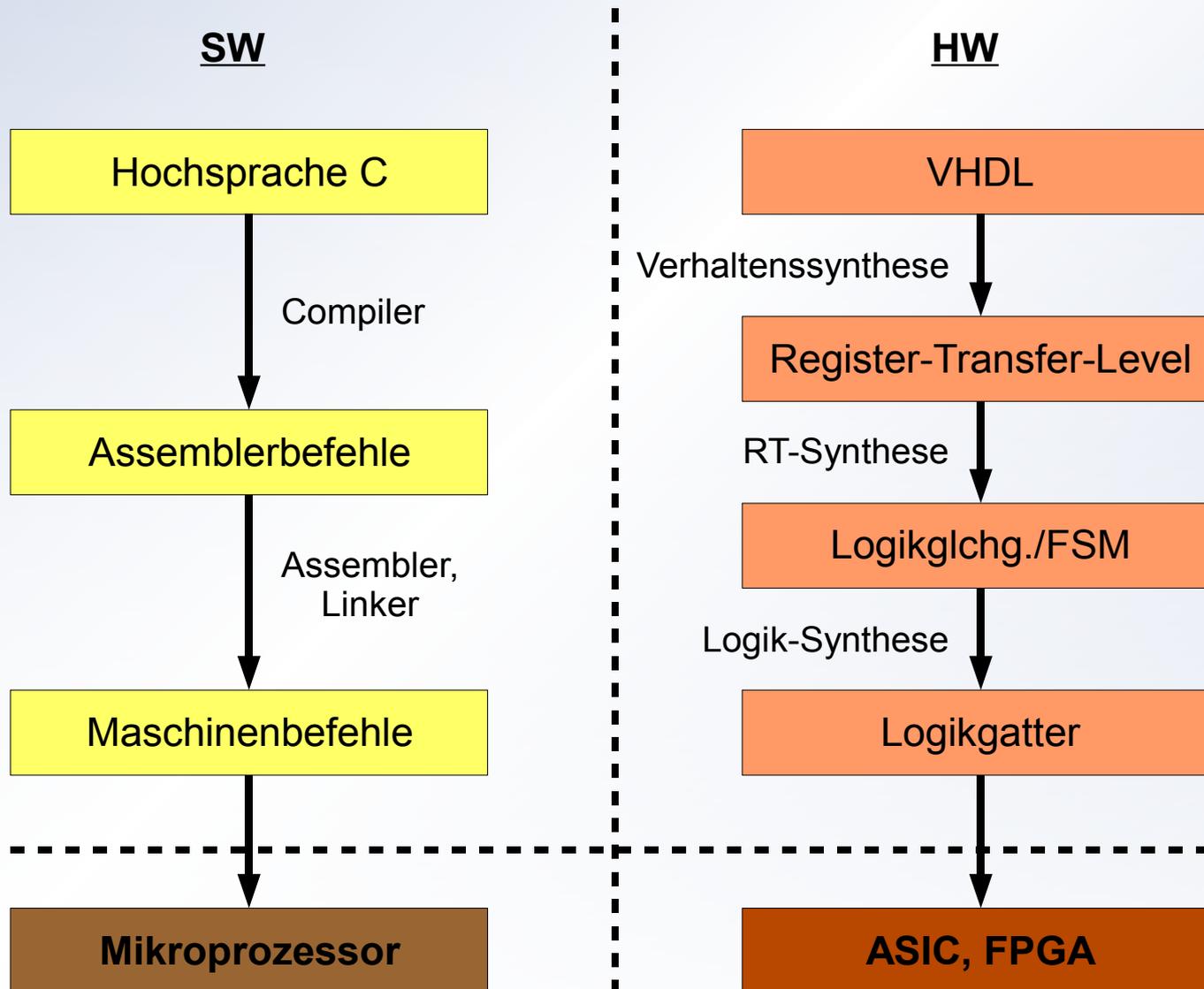
- FPGA oder ASIC

- **HW/SW System auf einem einzigen Chip**



Einleitung

- HW/SW Co-design



Einleitung

- **Entwicklung von eingebetteten Systemen**

- **reine SW-Entwicklung**

- SW muss sich der HW anpassen
- Overhead durch Scheduling/OS
- je höher der Abstraktionsgrad der Programmiersprache, desto geringer die HW-Kontrolle
- schnelle Anpassung an neue Anforderungen
- **sehr flexibel**

- **reine HW-Entwicklung**

- knappe Ressourcen
- höhere Entwicklungskosten
- lange Entwicklungszeiten
- keine Änderungen bei ASICs, nur bei FPGAs
- echte Parallelität
- **sehr performant**

Einleitung

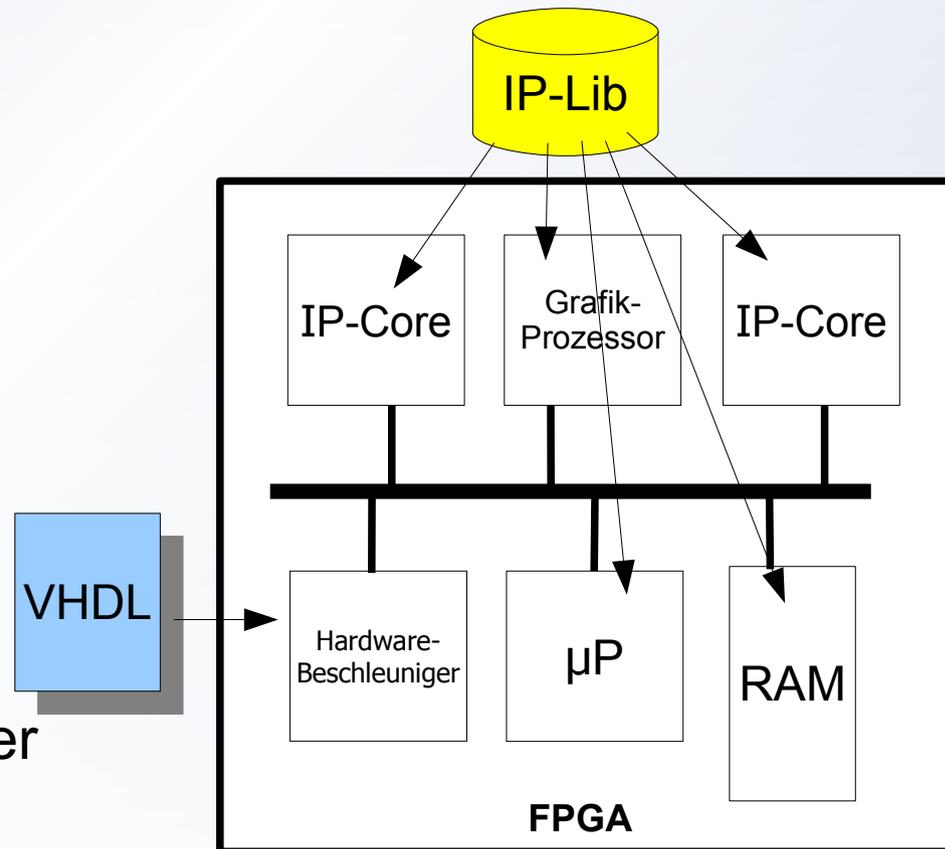
- Was ist SoC?

- Eine Ein-Chip-Lösung eines kompletten, digitalen Systems
- Prozessor + Peripherie + spez. HW auf **einem** Chip

- System on
“a programmable”
Chip

- FPGA

- Bussystem
- Softcore-Prozessor
- Speicher
- Hardware-Beschleuniger
- IP-Cores



Einleitung

- Warum So(ap)C?

- Hardware ist performant
- Software ist flexibel
- HW/SW Co-design vereint Hardware und Software
 - SoC ist ein Teilbereich von HW/SW Co-design
- schnelle Implementierung und Integration von Hardware-Beschleunigern durch SoC
- weniger Platzbedarf, mobile Geräte
- Nutzung der selben Plattform für verschiedene Aufgaben
 - dennoch **Domain-spezifische Flexibilität** (Kamera, Handy,...)

- Herausforderung

- Modellierung
- HW/SW Partitionierung

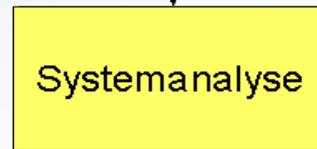
Einleitung

- HW/SW design flow

SysML
UML Profile for SoC



SystemC

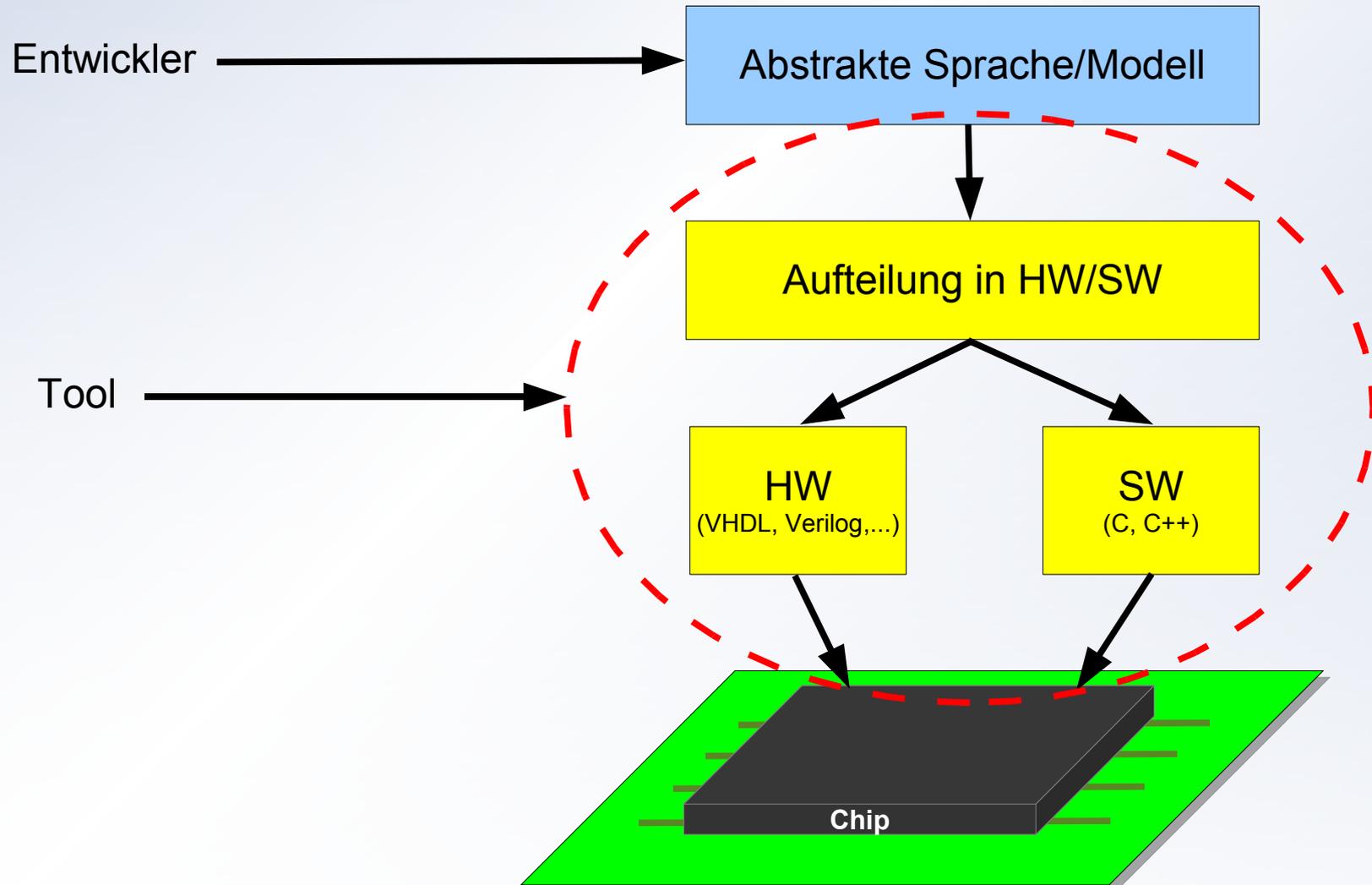


HW/SW Partitionierung



Einleitung

- Der Traum



Inhalt

- Einleitung
- SoC Designmethoden
 - SysML
 - UML Profile for SoC
- SystemC
- HW/SW Partitionierung
- Zusammenfassung
- Ausblick
- Literaturverzeichnis
- Glossar

Designmethoden

- **Framework**

- Modellierungsgerüst für einen bestimmten Anwendungsbereich, d.h. Methoden und Elemente, die zur Lösung eines bestimmten Problems benötigt werden.

- **UML**

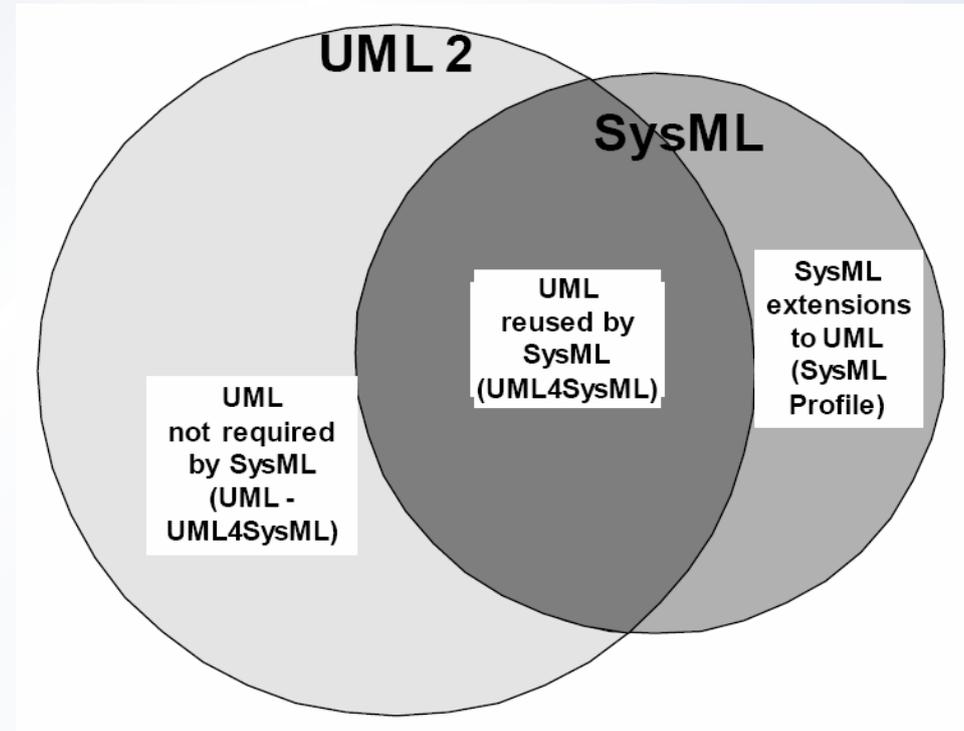
- Standard
- bewährte Methodik
- Toolportabilität (XMI)
- erweiterbar (Profile, Stereotypen)
- transparenter als Code
- SW-Entwickler verstehen UML
 - HW-Entwickler eher nicht, da zu **Softwarelastig**

Designmethoden

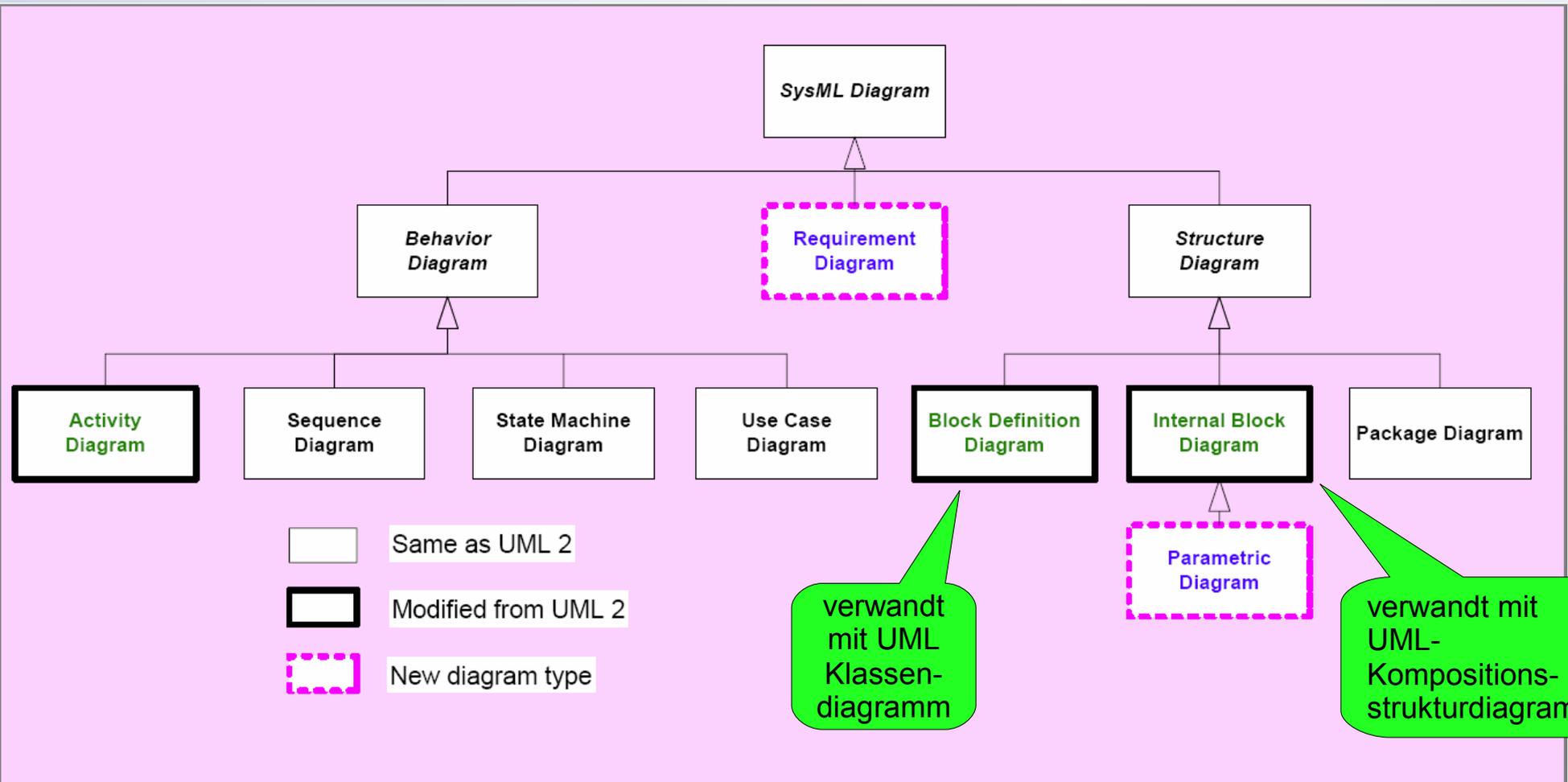
- **Anwendungsbereich SoC**
 - besteht aus HW und SW
 - benötigt einheitliche Sicht
- **also Erweiterung von UML für SoC über UML Profile**
 - Ein Profil ist ein Satz von spezialisierten Notationen, die die Basiselemente und Muster eines bestimmten Einsatzgebietes repräsentieren, um mehrere Interpretationen des selben Modells zu vermeiden.

SysML

- eine graphische Modellierungssprache als Antwort zu UML für das Systems Engineering
- SysML 1.0, Mai 2006, OMG + INCOSE
- Spezifikation, Analyse, Design, Verifikation und Validieren von komplexen Systemen
- **unabhängig von der Disziplin (SW, HW, Mechanik, ...)**
- UML Profil, das eine Untermenge von UML 2 mit Erweiterungen darstellt

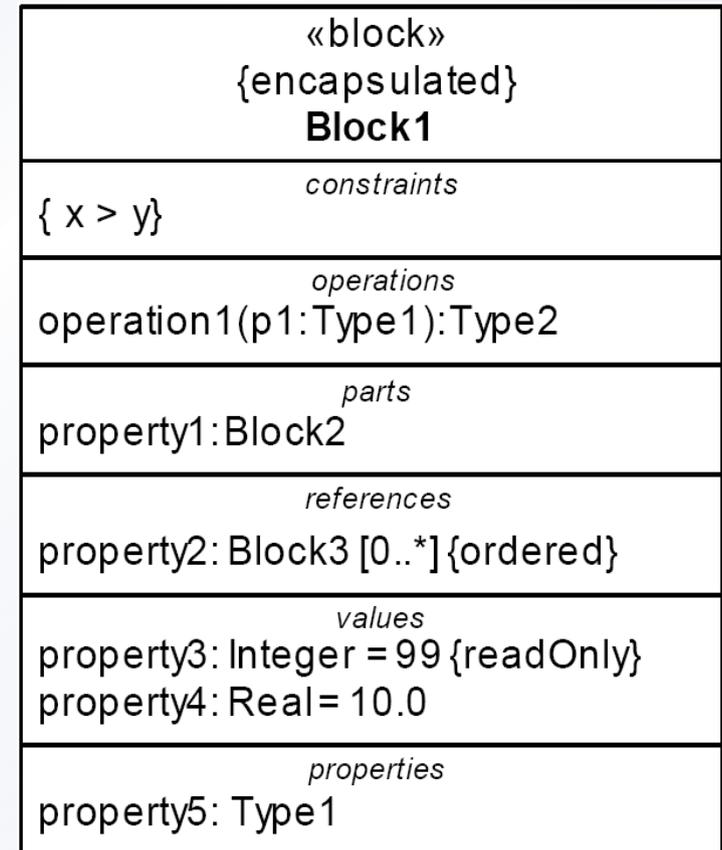


• Diagramm Überblick



• Blocks

- basieren auf UML Klassen
- spezifizieren Hierarchie und Abhängigkeiten
- Blöcken kann Verhalten zugewiesen werden (Aktivitäten)
- „Block definition diagram“ beschreibt Beziehungen zwischen Blöcken (Klassendiagramm)
- „Internal block diagram“ beschreibt die interne Struktur eines Blocks bezüglich seiner Eigenschaften und Verbindungen

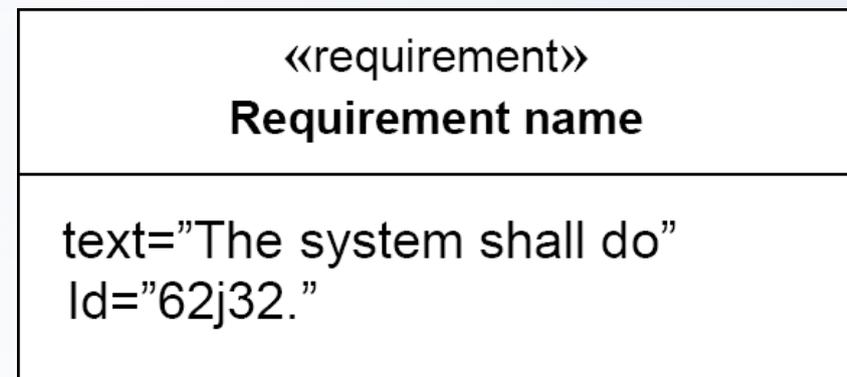


● Requirement Diagram

- Requirements (stereotypisierte Klasse) in Beziehung zu einander
- erweitert um ID und erläuterndem Text
- keine Attribute, keine Operationen
- nähere Beschreibung durch Anwendungsfälle oder Schnittstellen
- stets abstrakt, da reine Spezifikationselemente
- Beschreibung nicht-funktionaler Anforderungen

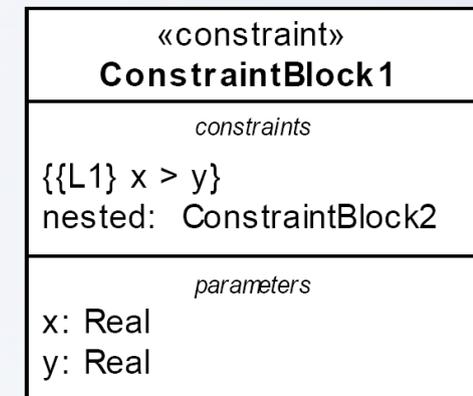
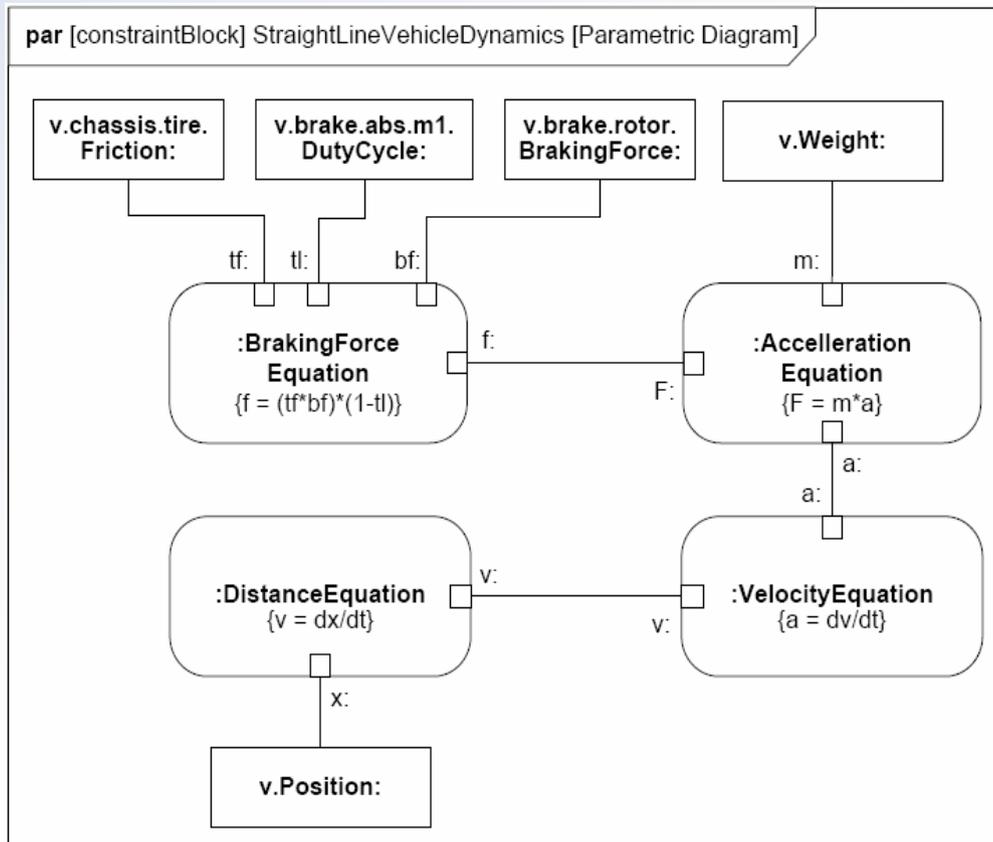
● Beziehungen

- Ableitungsbeziehung
- Enthältbeziehung
- Verfolgungsbeziehung
- ...



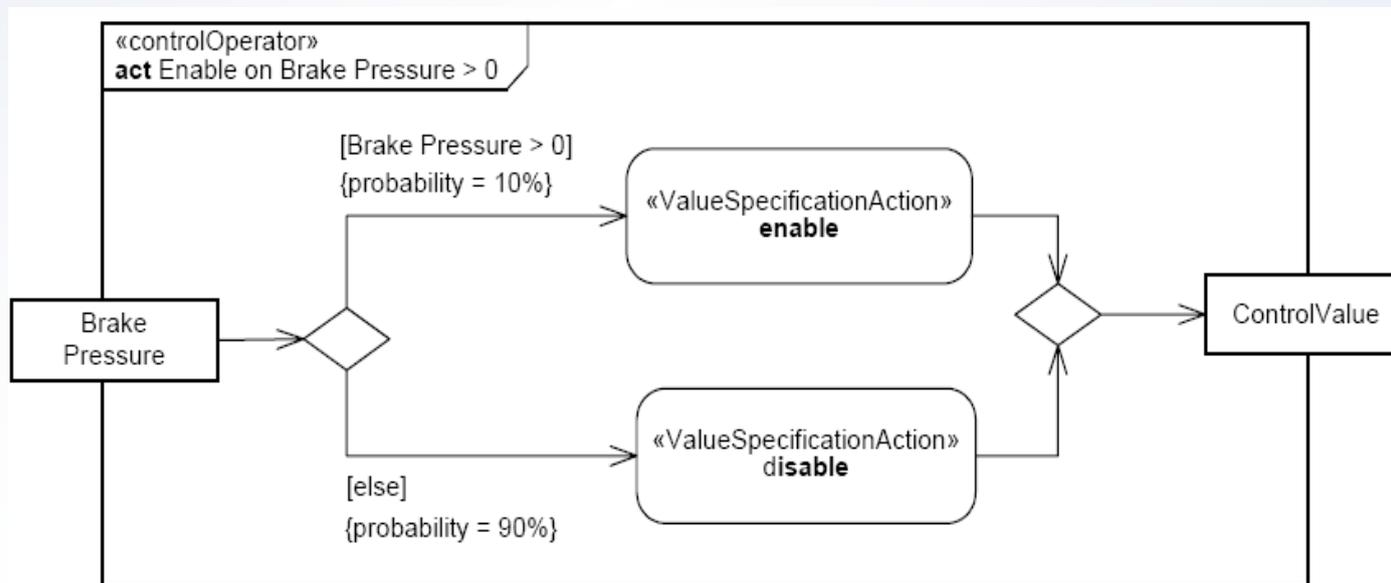
● Parametric Diagram

- Zusammenhänge zwischen Eigenschaften der Systembausteine lassen sich beschreiben und festlegen
- Integration von Performanz- oder Zuverlässigkeitsmodellen



Activities

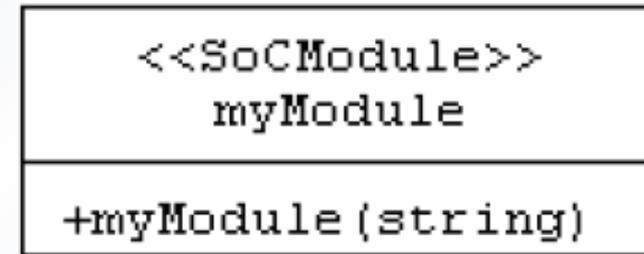
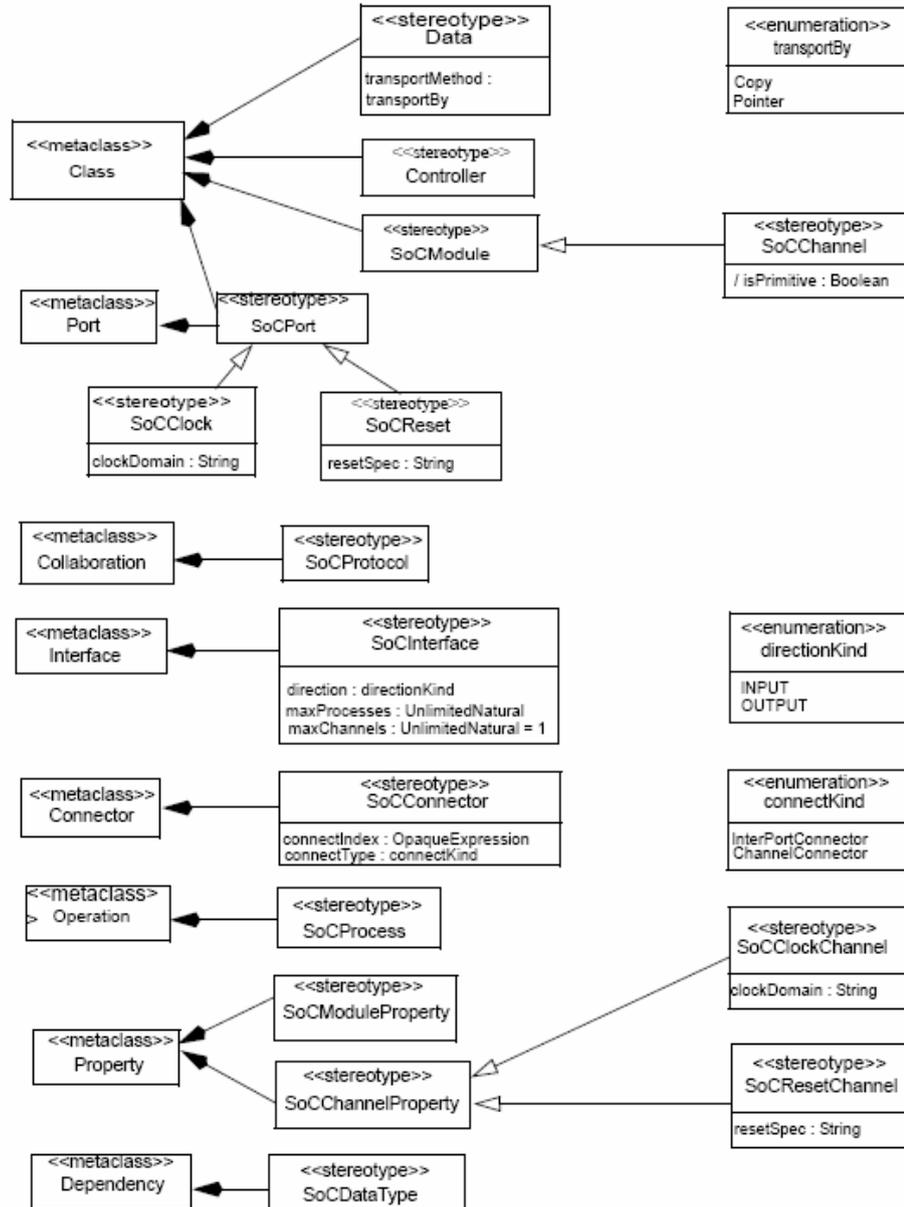
- Spezifikation von Abläufen sowie Ein- und Ausgabedaten
- erweiterter Kontrollfluss mit Zusatzinformationen, um Aktionen zu stoppen
- Unterstützung der Modellierung kontinuierlicher Systeme (kontinuierlicher oder diskreter Objektfluss)
- Wahrscheinlichkeit von Abläufen



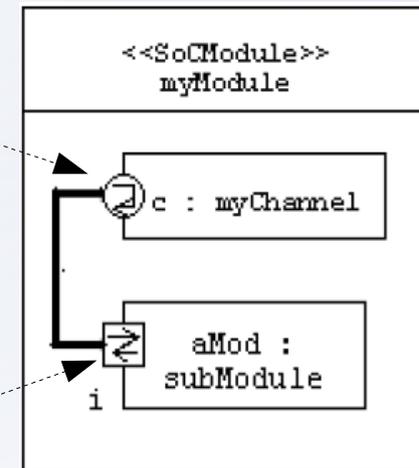
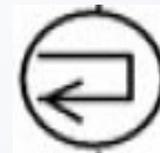
UML Profile for SoC

- „UML Profile for System on a Chip“ 1.0, Juni 2006, OMG + USoCF
- UML Konstrukte nicht ausreichend, um SoC Elemente zu repräsentieren
 - Satz neuer Stereotypen benötigt
- direkte Umsetzung in SoC Elemente
- keine Abhängigkeit von HDLs
- SystemC orientiert

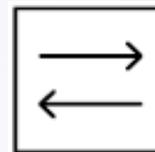
UML Profile for SoC



Protokoll Interface



Struktur Diagramm



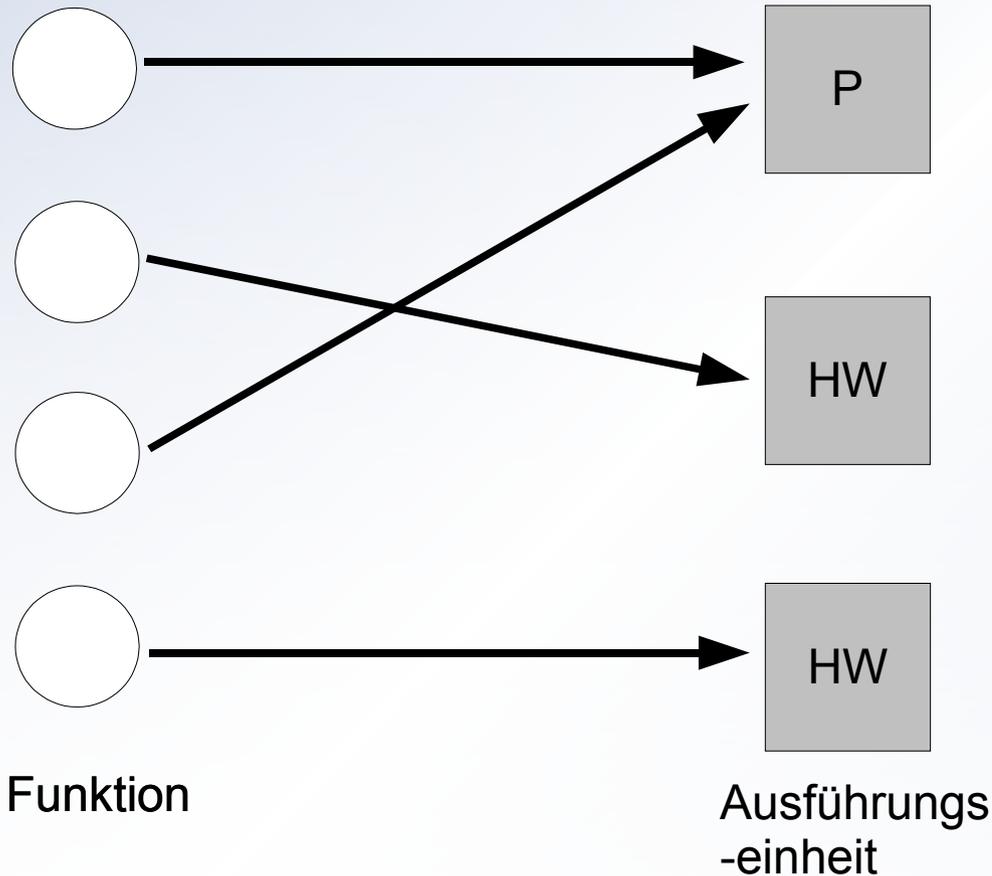
Port Interface

SystemC

- keine Sprache, C++ Klassenbibl.
- Open Source
- SystemC 2.2, OSCI, März 2007
- 2005 standardisiert durch IEEE
- anfänglich nur eingeschr. Simulator und RTL-Sprache
- erweitert zu System Level Sprache, d.h. HW und SW
- Konstrukte zur Modellierung von:
 - Zeitverhalten
 - Parallelität
 - synchronen und asynchronen Prozessen
 - Hierarchien
 - Hardware-Datentypen
- Simulation des Gesamtsystems

HW/SW Partitionierung

- Zuordnung von Funktionen entweder zu HW oder SW
- Programmausführungszeit verringern, durch Verlagerung von Teilen zwischen SW und HW



Funktion wird abgebildet auf HW



Funktionen werden als SW ausgeführt auf einem Proz.



HW/SW Partitionierung

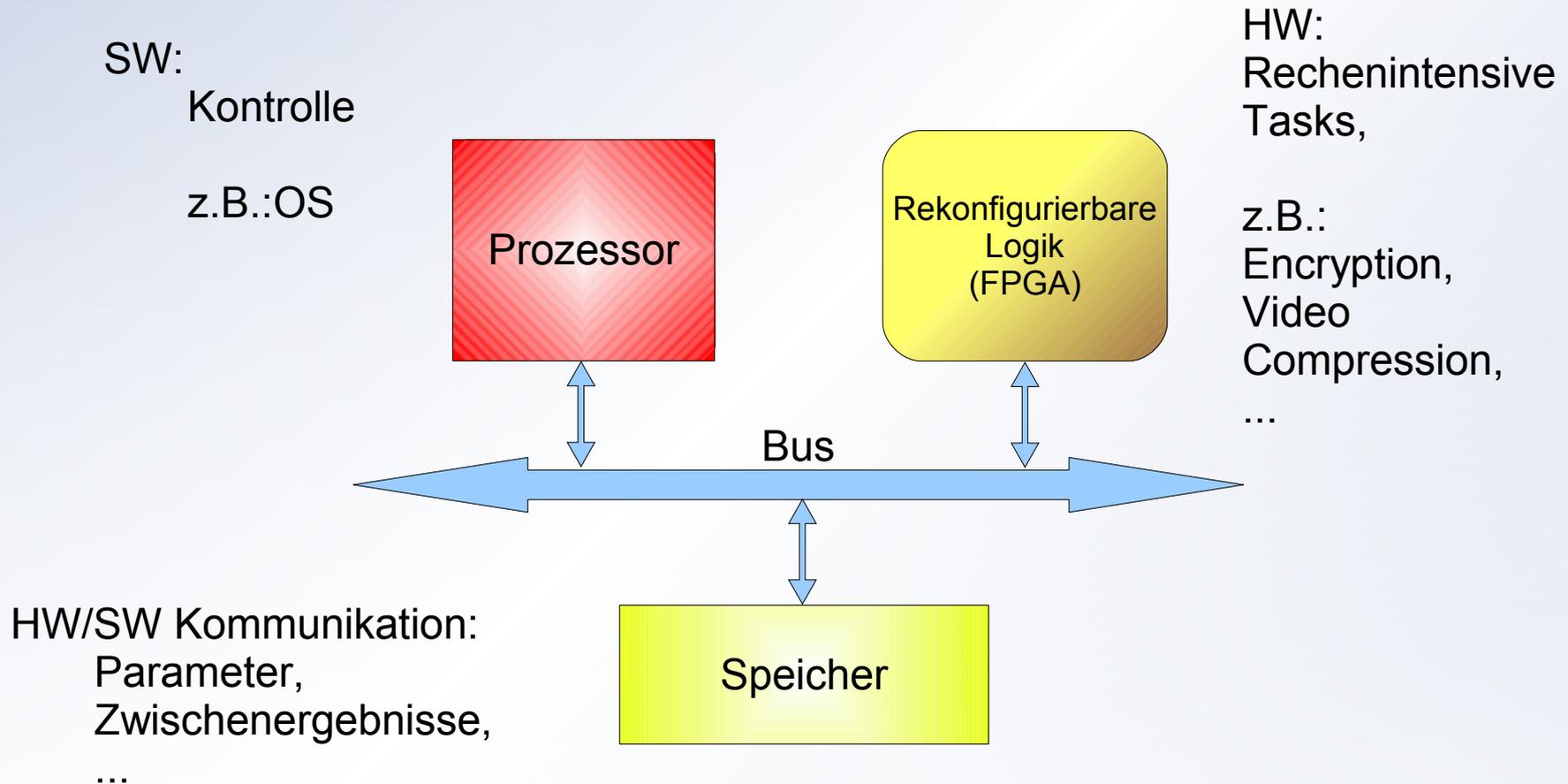
- Softwareorientierte Partitionierung
 - alle Funktionen in SW
 - Umlagerung von SW in HW, wenn
 - SW nicht schnell genug
- Hardwareorientierte Partitionierung
 - alle Funktionen in HW
 - Umlagerung von HW in SW, wenn
 - HW-Ausführung komplexer
 - Verbrauch von zu vielen Ressourcen
 - SW den Zeitanforderungen genügt
- viele Verfahren, Suche nach geeigneten Tools

Zusammenfassung

- SysML für abstrakte, einheitliche Modellierung von HW/SW Systemen
- „UML Profile for SoC“ besser geeignet für HW/SW Systeme im SoC Bereich
 - konzeptnah zu SystemC
 - automatische Umsetzung in SystemC
- SystemC
 - bekannte C++ Syntax
 - Simulation auf hohem Level
 - automatische Umsetzung in HW und SW
- HW/SW Partitionierung
 - viele Verfahren, keine Etablierten
 - Recherchebedarf, Algorithmen und Tools

Ausblick

- Handy Plattform, Dynamic Reconfigurable Computing



Ausblick

- Modellierungswerkzeuge für SysML, „UML Profile for SoC“
 - AW2, Seminar
- Recherche HW/SW Partitionierung
 - AW2, Seminar
- MPSoC/NoC
 - AW2, Seminar
- SysML Modell CaroloCup
 - Projekt bis Master
- FAUST Fahrzeug SoC Entwurf
 - Projekt bis Master
- ...

Literaturverzeichnis

Chris Rowen

Engineering the Complex SOC, Pearson Education, 2. Auflage, 2004

Jürgen Teich, Christian Haubelt

Digitale Hardware/Software-Systeme, Springer, 2. Auflage, 2007

Peter Marwedel

Eingebettete Systeme, Springer, 1. Auflage, 2007

Wayne Wolf

High Performance Embedded Computing, Elsevier(MK), 1. Auflage, 2007

David C. Black, Jack Donovan

SystemC: From The Ground Up, Springer, 1. Auflage, 2004

OMG Systems Modeling Language (OMG SysML™)

OMG Available Specification, V1.0, 07-09-01

OMG UML Profile for System on a Chip (SoC)

OMG Available Specification, V1.0, 06-06-01

Carsten Bieser

Konzept einer bibliotheksbasiert konfigurierbaren Hardware-Testeinrichtung für eingebettete elektronische Systeme, Dissertation, Uni Karlsruhe, 2007

Grant Martin, Wolfgang Müller

UML for SOC Design, Springer, 1. Auflage, 2005

Tim Weilkiens

Systems Engineering mit SysML/UML, dpunkt.verlag, 1. Auflage, 2006

Ralf Gessler, Thomas Mahr

Hardware-Software-Codesign, Vieweg, 1. Auflage, 2007

Glossar

- SoC – System on Chip
- FAUST – Fahrerassistenz- und Autonome Systeme
- FPGA – Field Programmable Gate Array
- ASIC – Application Specific Integrated Circuit
- IP-Core – Intellectual Property Core
- XMI – XML Metadata Interchange
- INCOSE – International Council On Systems Engineering
- USoCF – UML SoC Forum
- HDL – Hardware Description Language
- OSCI – Open SystemC Initiative
- RTL – Register Transfer Level