Design Space Exploration of Systems-on-Chip: DIPLODOCUS

Ludovic Apvrille

Telecom ParisTech
ludovic.apvrille@telecom-paristech.fr

May, 2011
Outline

Introduction
   Context
   Design Space Exploration

The DIPLODOCUS Approach
   DIPLODOCUS in a Nutshell
   Methodology
   TTool

Demo

Outlook
Outline

Introduction
Context
Design Space Exploration

The DIPLODOCUS Approach

Demo

Outlook
Systems-on-Chip

- A System-On-Chip = set of SW and HW components intended to perform a predefined set of functions for a given market
- Constraints
  - Right market window
  - Performance and costs
Design Challenges

**Complexity**
- Very high software complexity
- Very high hardware complexity

**Problem**
How to decide whether a function should be implemented in SW or in HW, or both?

**Solution**
Design Space Exploration!
Design Space Exploration

- Analyzing various functionally equivalent implementation alternatives
- Find an optimal solution

Important key design parameters

- Speed
- Power Consumption
- Silicon area
- Generation of heat
- Development effort
Level of Abstraction

Problematic

- Designers struggle with the complexity of today’s circuits
- Cost of late re-engineering
  - Right decisions should be taken as soon as possible ...
  - And quickly (time to market issue), and so, simulations must be fast

→ System Level Design Space Exploration

- Reusable models, fast simulations / formal analysis, prototyping can start without all functions to be implemented

But: high-level models must be closely defined so as to take the right decisions
DIPLODOCUS in a Nutshell

DIPLODOCUS = UML Profile
- System-level Design Space Exploration
- Y-Methodology
- MARTE compliant

Main features
- Data are abstracted
- Formal semantics
- Very fast simulation support
- Fully supported by an open-source toolkit
  - TTool
The Y-Methodology

- Simulation
- Static analysis

Application modeling

Architecture modeling

mapping

DSE

Simulation
Static analysis
Application Modeling

- Simulation
- Static analysis
- Application modeling
- Architecture modeling
- Functions are first modeled independently from the architecture
- DSE
- Mapping
- Simulation
- Static analysis
Architecture Modeling

Simulation
Static analysis

Application modeling

Architecture modeling

Then, architecture is modeled based on generic hardware components: microprocessors, buses, memories, bridges, etc.

Simulation
Static analysis

DSE

map
Mapping

Simulation
Static analysis

Application modeling

Architecture modeling

mapping

DSE

Functions are then associated to architecture components

Static analysis
Browsing the DIPLODOCUS Methodology

- Application structure
- Application behavior
- Formal verification

Simulation
Static analysis

Application modeling

Architecture modeling

DSE

Mapping model
Simulation
Formal verification
Figure: Activity Diagram of the SmartCard component
Formal Verification at Application Level

- No assumption on the underlying architecture
- All possible interleavings between actions are considered
- Formal verification is based on LOTOS or UPPAAL
  - Press-button approach
Given in terms of parameterized nodes

- CPU, HWA, Bus, Memory, Bridge, etc.
- CPU parameters: scheduling policy, cache miss ratio, miss-branching prediction, pipeline size, etc.
- Task are mapped on execution nodes (e.g., CPUs, HWAs)
- Channels are mapped on communication and storage nodes
After-Mapping Simulation

- TTool Built-in simulator
- Extremely fast
- Diagram animation
- Step-by-step execution, breakpoints, etc.
After-Mapping Formal Verification

- TTool built-in simulator can compute all possible execution paths
- Graph analysis and visualization
TTool: Main Features

- Open-source UML toolkit
- Meant to support UML2 profiles
  - 8 UML2 profiles are currently supported
- Mostly programmed in Java
  - Editor, interfaces with external tools
  - Simulators and model-checkers are programmed in C++ or SystemC
- Formal verification and simulation features
  - Hides formal verification and simulation complexity to modelers
  - Press-button approach
Outline

Introduction

The DIPLODOCUS Approach

Demo

Outlook
Outline

Introduction

The DIPLODOCUS Approach

Demo

Outlook
Results

Fully integrated environment for the fast design of Systems-on-Chip

- Based on UML
- Open-source toolkit

Support

- evita
- Texas Instruments
- Freescale
A Few Case Studies ...

- MPEG coders and decoders (Texas Instruments)
- LTE SoC (Freescale)
- Partitioning in vehicle embedded systems (EVITA project)
To Go Further ...

TTool and DIPLODOCUS

- ttool.telecom-paristech.fr
  - Type *TTool UML* under *google*
  - And click on the *I am lucky* button!