Lecture 12: System-Level Design Tools

Andreas Gerstlauer  
Electrical and Computer Engineering  
University of Texas at Austin  
gerstl@ece.utexas.edu

Lecture 12: Outline

• Overview
  • System-level design landscape

• System-level design tools
  • Commercial tools
  • Academic tools

• SCE commercialization
  • ELEGANT environment
  • Specify-Explore-Refine (SER) tools
Electronic System-Level (ESL) Landscape

ESL Tools

- **Electronic System-Level (ESL) terminology misused**
  - Often single hardware unit only (high-level HW synthesis)

- **System-level has to span across hardware and software**
  - System-level frontend
  - Hardware and software synthesis backend

- **Commercial tools for modeling and simulation**
  - Algorithmic modeling (MoC) [UML, Matlab/Simulink, Labview]
  - Virtual system prototyping (TLM) [Coware, VaST, Virtutech]
  - Only horizontal integration across models / components

- **Academic tools for synthesis and verification**
  - MPSoC synthesis [SCE, Metropolis, SCD, PeaCE, Deadalus]
  - Vertical integration for path to implementation
Commercial Tools (1)

• **CoFluent (now Intel)**
  - SystemC-based modeling and simulation
    - Networks of timed processes
    - Communication through queues, events, variables
  - Early, high-level interactive design space exploration
    - Graphical application, architecture and mapping capture
    - Fast TLM simulation with estimated timing

• **Space Codesign**
  - Graphical application, architecture and mapping capture (Eclipse)
    - Process network with message-passing or shared-memory channels
  - SystemC TLM simulation
    - Annotated, host-compiled or cycle-accurate ISS models
  - FPGA-based prototyping
    - Cross-compilation and third-party hardware synthesis (Forte/Catapult)

Commercial Tools (2)

• **CoWare (now Synopsys)**
  - Virtual platforms
    - SystemC TLM capture, modeling and simulation
    - Extensive library of IP, processor and bus models
    - Application-specific processor ISS models (LISAtek acquisition)
  - Proprietary SystemC simulation framework
    - Optimized SystemC kernel
    - Graphical debugging, visualization and analysis capabilities

• **Soc Designer (Carbon Design Systems)**
  - Proprietary, C++ based modeling and simulation
    - Fast, statically scheduled cycle-accurate simulation
    - Special cycle-callable component models

• **VaST (now Synopsys), Simics (Virtutech, now Intel), OVP**
  - Proprietary SW-centric virtual platform modeling/simulation
    - Fast, cycle-approximate binary translated or compiled ISS + peripherals
    - SystemC wrappers
Academic Tools

- **Metropolis**
  - Platform-based design (PBD)
- **SystemCoDesigner**
  - Dynamic dataflow MoC
  - Automated design space exploration
- **Daedalus**
  - KPN MoC for streaming, multi-media applications
  - IP-based MPSoC assembly
- **PeaCE**
  - “Ptolemy extension as a Codesign Environment”
  - Recent extensions for software development (HoPES)
- **SCE**
  - SpecC-based “System-on-Chip Environment”
  - Successive, stepwise Specify-Explore-Refine methodology

Academic Tools: Metropolis

- **Platform-based**
  - Pre-defined target architecture
    - Reuse
- **Meet-in-the-middle**
  - Platform mapping and configuration
- **General, proprietary meta-modeling language**
  - Capture function, architecture and mapping
- **Modeling framework**
  - Built-in parsing and simulation
  - Back-end point tool integration
**Academic Tools: SystemCoDesigner**

- **SysteMoC input model**
  - Dynamic dataflow MoC (actors + FSMDs) in SystemC
- **Fully automatic, multi-objective design space exploration**
  - Multi-objective evolutionary algorithms (MOEAs)

**Academic Tools: Daedalus**

- **KPN input model**
- **System assembly and simulation**
  - Sequential application
  - Automatic Parallelization

**XML-based open infrastructure**

**Multi-processor System on Chip**
(Synthesizable VHDL and C/C++ code for processors)
Academic Tools: PeaCE

- **Ptolemy-based**
  - Heterogeneous SDF+FSM application MoC

- **Stepwise flow**
  - Application partitioning
  - Communication architecture exploration
  - Code and interface generation

- **Software extensions: HOPES**
  - Parallel programming API
  - Multi-processor code generation

System-On-Chip Environment (SCE)

- Specification
- System Design
  - Architecture Exploration
  - Scheduling Exploration
  - Network Exploration
  - Communication Synthesis
- PE/OS Models
- CE/Bus Models
- TLM_i
- Hardware Synthesis
- Software Synthesis
- SW DB
- Implementation Model
- RTL DB
- HW
- Synthesize target HW/SW
- Compile onto MPSoC platform

Design Decisions
Academic MPSoC Design Tools

<table>
<thead>
<tr>
<th>Approach</th>
<th>DSE</th>
<th>Comp. decision</th>
<th>Comm. decision</th>
<th>Comp. refine</th>
<th>Comm. refine</th>
</tr>
</thead>
<tbody>
<tr>
<td>Daedalus</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>○</td>
</tr>
<tr>
<td>Koski</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>○</td>
</tr>
<tr>
<td>Metropolis</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>○</td>
</tr>
<tr>
<td>PeaCE/HoPES</td>
<td>○</td>
<td>○</td>
<td>○</td>
<td>●</td>
<td>○</td>
</tr>
<tr>
<td>SCE</td>
<td></td>
<td></td>
<td></td>
<td>●</td>
<td>●</td>
</tr>
<tr>
<td>SystemCoDesigner</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
<td>●</td>
</tr>
</tbody>
</table>

Lecture 12: Outline

- **Overview**
  - System-level design landscape

- **System-level design tools**
  - Commercial tools
  - Academic tools

- **SCE commercialization**
  - ELEGANT environment
  - Specify-Explore-Refine (SER) tools
**ELEGANT Environment**

- Specification model simulation
- Venüs
- Formal verification
- Exploration and Refinement
- TLM
- Co-simulation
- Cycle-accurate Co-simulation
- Software synthesis
- High-level synthesis
- Software source code
- Hardware RTL
- ELEGANT : Electronic Design Guidance Tool for Space Use

Source: InterDesign Technologies, Inc. / Japanese Aerospace Exploration Agency (JAXA)

**ELEGANT SpaceWire Evaluation**

- **SpaceWire**: aerospace communication protocol standard
  - High-speed and high-reliability interconnection network
    - Asynchronous, fault-tolerance, topology agnostic
  - Automated SpaceWire design with ELEGANT tool set
    - From top-level specification model down to HW/SW
      - HW/SW partitioning and exploration of the architecture with SER
      - Synthesis down to SpaceCube prototyping platform

Source: Japanese Aerospace Exploration Agency (JAXA)
ELEGANT MPEG4 Decoder Evaluation

- MPEG4 decoder implementation
  - Third-party evaluation by JAXA and Applistar, Inc.

VLD: variable length decoding
DEQ: de-quantization
IDCT: inverse discrete cosine transform
MD: motion vector decoding
MC: motion compensation

Explore design alternatives by SER

1. HR5000
   - 200MIPS-class 64-bit MPU for space apps.
   - MIPS5kf core
   - Eureka ES510 system controller

2. HR5000
   - 200MIPS-class 64-bit MPU for space apps.
   - MIPS5kf core
   - Eureka ES510 system controller

3. HR5000
   - 200MIPS-class 64-bit MPU for space apps.
   - MIPS5kf core
   - Eureka ES510 system controller

4. HR5000
   - 200MIPS-class 64-bit MPU for space apps.
   - MIPS5kf core
   - Eureka ES510 system controller

ELEGANT MPEG4 Design Explorations

- Final implementation delays simulated using single testbench
  - Synthesis to RTL and cycle-accurate (CA) model
    - Synthesis from SER-generated pin-accurate communication model (PAM)
    - FPGA model is CA-SpecC model generated by NEC’s CyberWorkbench
  - Co-simulation with ELEGANT system model
    - HR5000 CPU model is generated by SER
    - Back-annotated timing with Fastveri for SW behaviors

- Comparison of design alternatives

  - Performance estimation
    - Target performance (30 frames/s) can been estimated/verified

Source: JAXA and Applistar, Inc.
Lecture 12: Summary

- **System-level design tools**
  - Commercial focus still only on modeling and simulation
  - Academic approaches towards true system-level design
  - Emerging commercial backend HW/SW synthesis
    - Complete, automated system design flow
    - From specification to implementation

- **ELEGANT environment**
  - Full industrial system-level design solution
  - Integrated tools for modeling, synthesis & verification
  - Deployed in, e.g. NEC Toshiba Space Systems