A Flexible Simulation Environment for RISC-V

Karsten Emrich¹, Conrad Foik¹, Johannes Kappes², Sebastian Prebeck², Daniel Mueller-Gritschneder¹, Wolfgang Ecker², Ulf Schlichtmann¹
¹ Chair of Electronic Design Automation, Technical University of Munich
² Infineon Technologies AG

ETISS: Extendable Translating Instruction Set Simulator

Key Features [1]:
- ISA-independent instruction set simulator core
- Fast dynamic binary translation execution engine, based on a C-language intermediate format
- Easy integration into SystemC / TLM VPs
- Open Source with permissive license model (BSD 3-clause)
- Plugin mechanism with full access to various parts of the execution loop and memory interfaces -> easy integration of tracing, debugging, timing and memory models without needing to touch the actual simulation loop
- Dhrystone MIPS: ~110 for bare RV32GC model

M2-ISA-R: ISA Meta-Model and Code Generator

- M2-ISA-R provides a Python-based metamodel for describing arbitrary ISAs
- A two-step framework is provided for fast creation and generation of ISA extensions or new ISAs:
  - A front-end or producer produces a M2-ISA-R model hierarchy
  - Back-end or consumer tools to generate e.g. simulation models
- A reference front-end for the ISA description DSL CoreDSL 2 as well as a model generator backend for ETISS are provided

CoreDSL2 [2]:
- DSL for modeling complex ISAs
- Features / Support:
  - Parameterization of Cores (accounting for e.g. bit-width)
  - Consumer-independent descriptions
- Cross Verification:
  - Existing EDA tools can output CoreDSL files directly if Python-based output is infeasible
  - CoreDSL output allows broader possibilities of code reuse, also in other tools
  - Coherent descriptions for verification during HW/SW-Codesign

Future Work:
- Goal: Generate datasheet-accurate simulation models from a single CoreDSL 2 description
- Missing pieces:
  - Support of exception and interrupt behavior
  - Core-close peripherals (MMU, MPU, interrupt controller, SysTick)

References

Experimental Results

<table>
<thead>
<tr>
<th>Architecture</th>
<th>Forwarding</th>
<th>Branch prediction</th>
<th>Est. target perf. (CPI)</th>
<th>Simulation speed (MIPS)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Harvard</td>
<td>No</td>
<td>Static</td>
<td>1.39</td>
<td>15.07</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Dynamic</td>
<td>1.48</td>
<td>12.27</td>
</tr>
<tr>
<td>Vbn Neumann</td>
<td>No</td>
<td>Static</td>
<td>1.68</td>
<td>6.77</td>
</tr>
<tr>
<td></td>
<td>Yes</td>
<td>Dynamic</td>
<td>1.57</td>
<td>6.12</td>
</tr>
</tbody>
</table>

Future Work:
- Support of advance pipeline features (multi-issue, buffer, compressed instructions, etc.)
- Increased simulation speed
- Integration / adaptation to other simulators

Contact:
- wolfgang.ecker@infineon.com
Open Source:
- https://github.com/tum-ei-eda/m2-isa-r
- https://github.com/tum-ei-eda/etiss
- https://github.com/tum-ei-eda/CoreDSL