Building commercially relevant open source silicon

The many aspects of Ibex

Greg Chadwick, Digital Design Lead, lowRISC
gac@lowrisc.org
OpenTitan

- OpenTitan - An open-source silicon root of trust for industry
  - Multiple security/cryptography related accelerators (such as AES, OTBN, KMAC)
  - Various peripherals such as SPI, USB, I2C, UART
  - Discrete chip (RTL frozen May 2023 in preparation for tapeout) and Integrated variants (under development, using Discrete chip as a base)
    - [www.github.com/lowrisc/opentitan](http://www.github.com/lowrisc/opentitan)
    - [www.opentitan.org](http://www.opentitan.org)
- And OpenTitan needed a secure CPU…
Ibex

- lowRISC takes responsibility for zero-riscy from ETH Zürich and renames it Ibex
- An open-source RV32IMCB core, w/ additional extensions for Security, IoT and embedded applications
  - ePMP, debug, instruction cache, 2 or 3 stage pipeline
  - Dual core lockstep for fault detection and a variety of other security features (for use in OpenTitan)
  - Highly configurable
  - [www.github.com/lowrisc/ibex](http://www.github.com/lowrisc/ibex)
- Comprehensive documentation and DV
So What?
So What?

- RTL is only one aspect of what is required for designs viable for commercial usage
- Comprehensive DV, with full code and functional coverage is vital
- RTL needs to be consumable
  - Usable by a variety of EDA tools
  - Uniform style and adherence to SystemVerilog best practices for synthesis
  - Lint clean
- DV needs to be credible
  - Complete test plans and coverage plans
  - Nightly regressions with published results
- Project must be active and maintained
  - Code review and CI employed to ensure standards are met
- Silicon Commons framework
DV Strategy

- Major focus of recent Ibex work
- Randomized programs generated by RISC-V DV [https://github.com/chipsalliance/riscv-dv](https://github.com/chipsalliance/riscv-dv)
- UVM testbench with configurable sequences for constrained randomised stimulus
  - Instruction and data memory
  - Interrupts and debug requests
- Co-simulation with Spike ISS for checking
  - Includes checking of all data memory accesses
- Functional coverage divided into two
  - **Architectural coverage** Instructions executed and different cases of those instructions
  - **Micro-architectural coverage**
    - Specific Ibex behaviour, such as stalls and hazards
    - Doesn’t look at specific instructions, but instruction categories (Such as ALU or memory)
    - Cross coverage for combinations
    - PMP configurations and behaviour including corner cases
- Test plan specifying tests focussed on different scenarios enabling us to achieve full code coverage and functional coverage
- Directed tests used for hard to hit scenarios (mostly PMP related)
Current Status and Nightly Regression

- Regular regressions run by lowRISC
- Vital to demonstrate project health and life
- Monitored with failures triaged and fixed
- A key part of Silicon Commons working practices
- [https://ibex.reports.lowrisc.org/opentitan/latest/report.html](https://ibex.reports.lowrisc.org/opentitan/latest/report.html)

Ibex Regression Results

Date/Time run: Thursday 25 May 2023 04:10 UTC

Git Commit: a31c043

<table>
<thead>
<tr>
<th>Test Name</th>
<th>Passing</th>
<th>Total</th>
<th>Pass Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>riscv arithmetic basic test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv machine mode rand test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv rand instr test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_rand_jump_test</td>
<td>9</td>
<td>10</td>
<td>90.0%</td>
</tr>
<tr>
<td>riscv_jump_stress_test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_loop test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_mmu_stress_test</td>
<td>9</td>
<td>10</td>
<td>90.0%</td>
</tr>
<tr>
<td>riscv illegal_inst_test</td>
<td>15</td>
<td>15</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_hint_inst_test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_ebreak_test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv debug_basic_test</td>
<td>9</td>
<td>10</td>
<td>90.0%</td>
</tr>
<tr>
<td>riscv debug_triggers_test</td>
<td>5</td>
<td>5</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv debug_stress_test</td>
<td>15</td>
<td>15</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv debug_branch_jump_test</td>
<td>7</td>
<td>10</td>
<td>70.0%</td>
</tr>
<tr>
<td>riscv_debug_inst_test</td>
<td>24</td>
<td>25</td>
<td>96.0%</td>
</tr>
<tr>
<td>riscv debug_wfi_test</td>
<td>10</td>
<td>10</td>
<td>100.0%</td>
</tr>
<tr>
<td>riscv_dbrt_test</td>
<td>3</td>
<td>5</td>
<td>60.0%</td>
</tr>
<tr>
<td>riscv_debug_ebreak_test</td>
<td>14</td>
<td>15</td>
<td>93.3%</td>
</tr>
<tr>
<td>riscv_debug_ebreakmu_test</td>
<td>11</td>
<td>15</td>
<td>73.3%</td>
</tr>
</tbody>
</table>

---

**Coverage**

<table>
<thead>
<tr>
<th>Functional</th>
<th>Block</th>
<th>Branch</th>
<th>Statement</th>
<th>Expression</th>
<th>Toggle</th>
<th>FSM</th>
<th>Assertion</th>
</tr>
</thead>
<tbody>
<tr>
<td>94.6%</td>
<td>94.5%</td>
<td>94.6%</td>
<td>94.6%</td>
<td>97.1%</td>
<td>100.0%</td>
<td>94.1%</td>
<td></td>
</tr>
</tbody>
</table>

**Test Failure Details**

```plaintext
riscv_rand_jump_test:2565

binary: test.bin
rft_log: rft_sim.log
rft_trace: trace_core_00000000.log
iss_cosim_trace: spike_cosim_trace_core_00000000.log

[FAILURE] Simulation ended gracefully due to timeout [1000ms].
```

---

**OpenTitan**

- [https://ibex.reports.lowrisc.org/opentitan/latest/report.html](https://ibex.reports.lowrisc.org/opentitan/latest/report.html)

---

RISC-V Summit Europe June 2023

---

lowRISC
Free Fully-Verified IP? Fantastic!
Free Fully-Verified IP?
Fantastic!
Yes - but there’s more to it
Why Open Source?

- It’s a mistake to just see it as ‘Free IP’, it’s a different development model with distinct advantages
- Tweakable and customizable, can close DV rapidly on modified design provided existing testbench and collateral, like test plans and coverage plans, are open too
- Multiple contributing partners can build a better design with less resource compared to doing it all in-house
- No commercial pressure to gate-keep improvements behind yearly releases
- Freedom to configure as you want, no marketing and sales need for lots of model numbers and bundled features
  - It’s not Ibex-A478-S or Ibex-A480-B, just Ibex, with many possibilities
- Contributing partners are vital, open source isn’t free!
- Using Ibex already? Interested in supporting our work or helping drive development?
  - Get in touch - info@lowrisc.org
Thank you

info@lowrisc.org - General lowRISC enquiries
gac@lowrisc.org - Presenter