4 years of Open Source RISC-V at Thales

RISC-V Summit Europe, Barcelona, June 8, 2023

Thierry Collette, Ph.D.

Thales Research and Technology
Open HW : New research opportunities

CVA6 pipeline architecture & ISA

CV32A6: RV32IMA[F][C]_Zicsr_Zifencei M/S/U [Sv32]

CV64A6 - RV64IMA[F][D][C]_Zicsr_Zifencei M/S/U[H] [Sv39]

SYMPATI2 : SIMD Processor Architecture
(T. Collette Ph.D.)
CONTINUOUS & TRUSTABLE COMPUTING SOLUTIONS

- Autonomy
- Low Power
- Safety
- Cybersecurity
- Interconnection
- Cloud-Edge-IoT
Exemple of AI R&T Challenges
Exemple of R&T challenges for IT – OT convergence

Before

Simulation
Symbolic IA
Machine Learning

Embedded/OT

HPC

Continuum computing

In progress

QPU?

Simulation
Digital twin
Machine Learning
Symbolic AI
Optimisation
Decision

Preprocessing
Sensor Data Fusion
Embedded Inference (AI)

Preprocessing
Sensor Data Fusion
Embedded Inference (AI)

HPC

Edge

Emb. /OT

Off Line

Configuration

Symbolic data

Iconic data

Sensors

On Line

On Line

Off Line

Off Line

Emb. /OT

HPC

Simulation
Symbolic IA
Machine Learning

Machine Learning
Symbolic AI
Optimisation
Decision

Preprocessing
Sensor Data Fusion
Embedded Inference (AI)
Why Thales invests in RISC-V? Main differentiators:

- **No vendor-locking & Sovereignty & Share cost instead of purchasing IP**
  - Open-source community
  - Business opportunities for support, customization…
  - Possible commercial exploitation without export constraints
  - Enable strong EU investment

- **SWaP & customization & Safety & Security**
  - Exact fit between features and application needs
  - A fully auditable processor
  - No black-box
Enablers of RISC-V wave

Software and service ecosystem

- Academic and Industrial
- across implementations

Performance

- State-of-the-art processor

Minimum mass production

Active RISC-V projects at Thales TRT-Fr

It took 15 years to Linux for massive adoption. RISC-V has started in 2010
Our RISC-V communities

RISC-V International (“the Foundation”)

- Specifies the open RISC-V instruction set
  - Simple & modular
  - 32- or 64-bit
  - Custom extensions
  - Covers a wide range of needs, from MCU to HPC
- Currently specifying upcoming optional extensions
  - E.g. bit manipulation, crypto…
- Hosts several special interest groups (SIG)
- Does not deliver implementations

OpenHW Group

- Not-for-profit corporation steered by its members
- Goal: deliver open-source IP for production SoCs
  - RISC-V compatible cores
  - SoC IP blocks
  - Verification environment
  - Supporting SW and tools
- Permissive, open-source, export-friendly license
Open-source is suited for industrial use

### OpenHW governance
- Not-profit organization steered by its members
- Based on Eclipse Foundation’s processes
- Target industrial-grade quality

### Value generation
- Share cost instead of purchasing proprietary IIP
- Customize for own application
- Increase control on your solutions
- Easier white box certification

### Participation is encouraged:
- Share IP development costs
- Influence technical content
- Get recognized as a contributor

### Business models
- Commercial SW/HW/tooling add-ons
- Maintenance and support offers

### Apache / Solderpad permissive licences
- Freely use, modify, integrate in proprietary solutions
- No need to publish modifications, no viral effect

### Reduced supplier / export risks
- Ability to fork; no end-of-life
- Significantly lower exposition to export control

---

This document may not be reproduced, modified, adapted, published, translated, in any way, in whole or in part or disclosed to a third party without the prior written consent of Thales. © Thales 2018 All rights reserved.
CVA6 core

Open-source RISC-V application core
- Supports rich OSes like Linux

Common source code, two flavors:
- CV64A6
  - 64-bit
  - ARIANE donated by ETH Zürich to OpenHW
- CV32A6
  - 32-bit
  - Compact version designed by Thales
## HW implementations

<table>
<thead>
<tr>
<th>CV32A6</th>
<th>FPGA</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Frequency</strong></td>
<td>125 MHz</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>2.25 CoreMark/MHz 281 CoreMark</td>
</tr>
<tr>
<td><strong>Resources</strong></td>
<td>10,416 LUT</td>
</tr>
<tr>
<td><strong>Technology</strong></td>
<td>Zynq UltraScale+ -3</td>
</tr>
<tr>
<td><strong>Configuration</strong></td>
<td>RV32IMA, 8K D$ + 8K I$, noFPU, MMU</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CV32A6</th>
<th>ASIC</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Frequency</strong></td>
<td>900 MHz</td>
</tr>
<tr>
<td><strong>Performance</strong></td>
<td>2.5 CoreMark/MHz 2250 CoreMark</td>
</tr>
<tr>
<td><strong>Resources</strong></td>
<td>80 kgates</td>
</tr>
<tr>
<td><strong>Technology</strong></td>
<td>28 nm (worst case corner)</td>
</tr>
<tr>
<td><strong>Configuration</strong></td>
<td>RV32IMA, 8K I$ + no D$, noFPU, MMU</td>
</tr>
</tbody>
</table>

Ongoing work.
More optimizations are coming!
## CVA6 2022 results: FPGA optimization

![Diagram showing FPGA optimization results](image)

<table>
<thead>
<tr>
<th></th>
<th>Original CV32A6</th>
<th>Optimized version</th>
<th>Evolution</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Look-up tables</strong></td>
<td>18,103</td>
<td>8,077</td>
<td>−55%</td>
</tr>
<tr>
<td><strong>Flip-flops</strong></td>
<td>11,484</td>
<td>4,403</td>
<td>−61%</td>
</tr>
<tr>
<td><strong>DSP blocks</strong></td>
<td>4</td>
<td>4</td>
<td>-</td>
</tr>
<tr>
<td><strong>Block RAM</strong></td>
<td>36</td>
<td>12</td>
<td>−67%</td>
</tr>
<tr>
<td><strong>Max. freq.</strong></td>
<td>100 MHz</td>
<td>140 MHz</td>
<td>+40%</td>
</tr>
<tr>
<td><strong>CoreMark/MHz</strong></td>
<td>2.8</td>
<td>2.8</td>
<td>-</td>
</tr>
<tr>
<td><strong>CoreMark</strong></td>
<td>280</td>
<td>392</td>
<td>+40%</td>
</tr>
</tbody>
</table>

(Xilinx Kintex7)

Some optimizations are also beneficial for ASIC.
Multi-sourcing

- For **ASIC targets** (32/64 bit)
- For any **FPGAs** (32 bit)

Leverage your investment:
reuse your HW/SW architectures throughout your product range
(multi-sourcing: any ASIC and FPGA vendors)
CVA6 2022 results: Linux Yocto available

Up-to-date Yocto embedded Linux released:

- The most popular distribution generator for embedded systems
- 32- and 64-bit support
- Built upon U-Boot and OpenSBI
- [https://github.com/openhwgroup/meta-cva6-yocto](https://github.com/openhwgroup/meta-cva6-yocto)

<table>
<thead>
<tr>
<th>File</th>
<th>Edition</th>
<th>Architecture</th>
<th>Terminal</th>
<th>Options</th>
<th>Aide</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>0.0973456</td>
<td>x86_64</td>
<td>U-Boot x86 Enhanced</td>
<td>Host Controller (EHCI) Driver</td>
<td></td>
</tr>
</tbody>
</table>
Demonstrator – Doom on CV32A6 soft-core

### Demonstrates CV32A6 softcore performance

- Support of high-level software applications
- Implementation on the Genesys 2 FPGA board
  - Kintex 7 (XC7K325T-2FFG900C)
  - DDR3
  - PMOD-VGA connected to a monitor
- Yocto Linux OS running
- VGA framebuffer integration
  - SVGA 800x600 at 60FPS
- Chocolate-doom
  - 15 FPS

### Ready for products

- Augmented reality
- Embedded HMI
RISC-V student contest

Organized by
> Prizes sponsored by Thales

Goals
> Promote RISC-V and computer architecture in French education
> Extend RISC-V and OpenHW communities
> Strengthen industry-academy connections

2020-2021: Improve CV32A6 FPGA performance
> 13 teams from 10 universities
> Awarded: Télécom Paris, U. Toulouse III

2021-2022: Improve CV32A6 energy efficiency
> 12 teams from 7 universities
> Bernhard Quendt, Thales CTO, gave out the awards
> Awarded: U. Strasbourg (2 teams), IMT Atlantique

2022-2023: Focus on CV32A6 security
SW ecosystem

**Boot and FW**
- U-Boot
- OpenSBI

**OS support**
- **Linux**: 32 & 64 bit
- **Yocto** honister, Buildroot 2021.08
- **FreeRTOS**: 32 & 64 bit
- CVA6 compatible with many others

**Compiler**
- Standard **GCC** (11.2)
- Libraries: **glibc** (2.70) & others
- LLVM on the roadmap

**Debug**
- HW and baremetal: JTAG probe, OpenOCD, GDB
- Linux-based: GDB server, GDB/Eclipse IDE

Full open-source software ecosystem

Protect your HW investments
CVA6 verification

- Continuous integration (CI)
- Leverages Google open-source components and OpenHW methodology

Next steps:
- Complete UVM testbench
- New test sequences
- 100% functional coverage (UVM-based)

Verification artifacts will be available as open-source.
Target is 100% verification coverage.
An extendable core

CV-X-IF interface to extend the CVA6 instruction set
- Current or future RISC-V extensions
- Custom extensions (crypto, DSP, AI…)

CV-X-IF specified by OpenHW Group
- Open specification, can be used off OpenHW
- Reuse coprocessors between CORE-V cores (CVA6, CV32E40X, CVE2)

Compiler support
- Seamless for RISC-V standard extensions
- LLVM should ease the support of custom extensions
- Inline ASM possible for specific processing

Speed up your application with a custom accelerator
Add extensions without fully re-validating the core
Soft-core roadmap

An industrial roadmap for an open-source soft-core

- CVA6 initially designed for ASIC & in 64 bits
- **2020**
  - 32-bit CVA6 version (ASIC & FPGA)
  - First optimized FPGA soft-core
  - Support of Linux
  - Dual-core version running Linux SMP

- **2021**
  - Optimizations (+50% freq. and -30% area)
  - U-Boot, OpenSBI, Yocto

- **2022**

- **202x**
  - More optimizations to come!
Conclusion

Join the development of an industrial-grade open-source processor

- Get funding from the HE, KDT JU and ESA for IoT and Edge and EuroHPC JU for HPC and Edge
  - EU institutions are strongly supporting RISC-V sovereign technologies
- Influence next developments for your future products

Participate in the next open-source revolution

- The next “Linux” for hardware CPUs

Use the CVA6 in your future products

- For ASIC and FPGAs
- With a strong open SW ecosystem

And we will success with a sustainable Ecosystem and Mass Production
Thanks to the Thales Teams:

- DIS/INVIA,
- Thales INDIA,
- TRT-Fr.