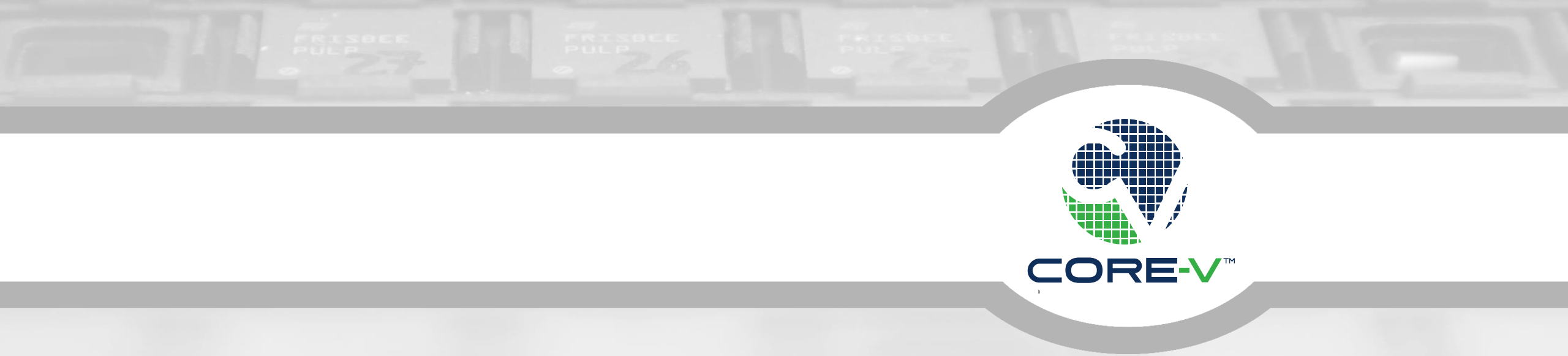


CORES TG – September 4 2023

Arjan Bink

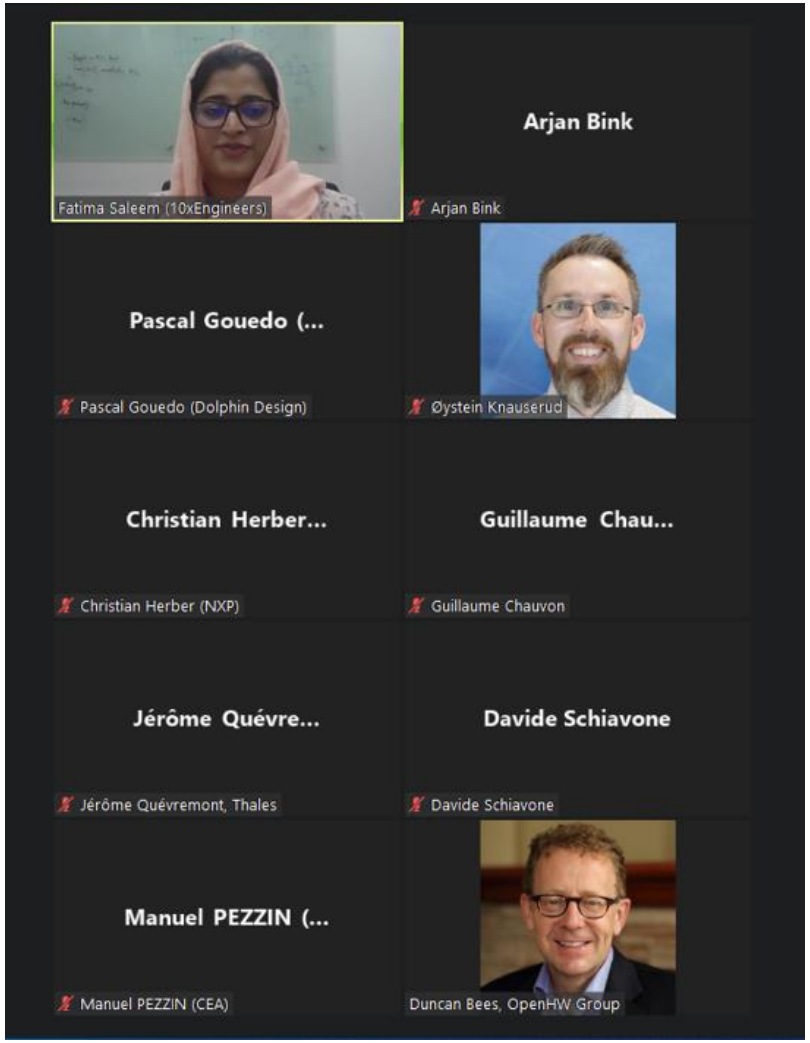
Jérôme Quevremont

Davide Schiavone

Agenda

- 10xEngineers roadmap on CVA6
- Thales design in on CVA6
- CV-X-IF project concept at Sept. 25th TWG meeting
- CV32E40X / CV32E40S status
- CV32E40P status
- Core overview

Participants



10xE CVA6 Roadmap

Fatima Saleem
Sr. Engineer

Open Source: Contributions & Experience

- Added support for Bit Manipulation extensions (Zba, Zbb, Zbc, Zbs) in CVA6 and verified it using the RISC-V Architectural Compliance tests (PR#878)
  - Added 43 instructions in total, and made the implementation parameterisable for XLEN 64 & 32 with clean synthesis
  - Updated the core-v-verif infrastructure to run the architectural compliance tests for bitmanip (PR#1254)
- Resolved over ~100 Lint warnings in CVA6 (PR#1150, PR#1268, PR#1280, PR#1297, PR#1303) (issue#1155)
- Developed MMU Architectural DV Plan and documented it in VPTOOL (PR#1789)

Open Source: Contributions & Experience (CVA6+CVVEC)

- Added support for 17 vector instruction to make ARA Compatible with RVV1.0
  - Vector Fixed Point — vsmul, vssra, vssrl, vnclup, vnclipu (PR#147)
  - Vector Mask — vmsbf.m, vmsif, vmsof.m, viota.m,vid, vpopc, vfirst (PR#149, #178)
  - Vector Permutations — vrgather, vcompress(PR#180)
  - Vector Floating Point — vfrec7.v, vfsqrt.v, vfncvt.rod.f.f.w (PR#184 #191 #201)
- Added support for single lane configuration and added support for shorter vector Lengths VLEN = 128,256,512 (PR#194)
- Added Design Documentation of ARA

Client work: Contributions & Experience

Integrated CVA6 in a system with a custom vector accelerator

- Enhanced CVA6 issue and load store logic to have the memory accesses consistency between scalar and vector loads and stores
- Shared CVA6 MMU for the address translation of vector load/store accesses
- Design and Verification of the RISC-V Vector Processor's Load/Store Unit
- Bringup of SV based verification environment for Questa/Vivado simulation
- Implemented co-simulation environment using SystemC for RTL vs an ISS model

Current Commitments

- Writing self-checking tests to execute MMU architectural testplan (Q3 2023)
- Resolve the remaining lint issues (Q3 2023)
  - Close the remaining open issues related to lint warnings
- Design Documentation of MMU (Q3 2023)
- Boot linux on ARA (CVA6+CVVEC) – (Q4 2023)
  - Add Virtual Memory Support by Sharing CVA6 MMU for address translation of vector load/store
  - Verify the support for exceptions/interrupts and shared MMU

Current Commitments

- SV/UVM Agent for Micro-architectural Verification of MMU (Q1 2024)
  - Plan is to test MMU mircro-arch using coverpoints and passive SV/UVM agent that would simply monitor the RTL and model transactions and broadcast it to the checker for comparison
  - Monitor would get the transactions from MMU and broadcast it to the scoreboard or checker
  - Model will contain tasks to mimic the MMU HW
  - Scoreboard/Checker would compare the transactions that it would get from the MMU model and MMU RTL

Potential Future Roadmap

- Adding support for extensions
  - Add support for Zicond (Conditional Ops) extension (Q3 2023)
    - Pending Verification
  - Add support for **Code Size Reduction** extensions (Zca, Zcb, Zcd, Zce, Zcf, Zcmp, Zcmt) (Q4 2023)
  - Add support for Bfloat16 Extensions (Zfbfmin, Zvfbfmin, Zvfbfwma) (TBD)
- Performance modeling and performance improvement
  - Enabling Dual issue, and renaming
  - Exploring the idea of updating the performance model for CVA6, which will help to identify the performance bottlenecks

THALES
Building a future we can all trust

Design-in: CVA6 embedded configuration

Jean-Roch COULON – THALES (TSS)

Design-in

- cv32a60x is a functional trade off between OpenHWGroup members
- *Recent Design-in*: THALES TSS plans to insert cva6 in its own products

Embedded configuration

Embedded configuration requests Design and Verification work

Main Challenges are (in order):

- **Power:** 32bits, smaller I and D caches, no FPU, only Machine mode, no perf counters, exception gate count reduction, AXI simplification, (Dual issue?)
- **Security:** PMP (out of MMU), no debug mode, CVXIF
- **Performance:** (Data scratch pad?), bare metal, (ZBitmanip and Zicond extensions?)
- **Code size:** C, Zc extensions

Features are RTL options

Start project for Dual Issue, will be presented in Aug 28th TWG

Next steps ?

KICKOFF MEETING CV32A6 TRUST BUT VERIFY

- cv32a60x verification (step1) is on-going, step2 and step3 were planned
- Thales TSS plans to use CV32A6_embedded for industrial product: design and verification tasks
- 10xEngineers presented its roadmap
- BOSCH, Axellera,… could join the project

CV-X-IF Project Concept

- The TRISTAN and ISOLDE European cooperative projects will work on the CV-X-IF specifications
  - E.g. addressing gaps…
  - Public work
  - Led by Christian Herber, NXP
- OpenHW members are welcome to join the initiative
- Next:
  - Project Concept at TWG Meeting
  - Monday 25th September, 16:00 CEST, 10:00 am EDT
  - Details in https://calendar.google.com/calendar/u/0/r?cid=meetings@openhwgroup.org

CV32E40X / CV32E40S status

Øystein Knauserud

CV32E40X and CV32E40S

- Cleanup
  - TODO removal
  - Lint cleaning
- Bugfixes
- Spec updates (CLIC and debug)
- Standardized way of killing multicyle instructions
  - Power saving in DIV/MUL
- User Manual updates

Timeline

- Both cores delayed until Q4 2023
  - Major unfinished work is the extension interface
    - Outstanding bugs, want to do a major rewrite of the implementation
    - Likely to prioritize other things above XIF, further delay expected.
  - Unratified CLIC and debug specifications

CV32E40Pv2 status

Pascal Gouédo

Yoann Pruvost

Xavier Aubert

Project

- Project
  - Design & Verification meeting
    - Wednesday 14:00 CET every 2 weeks(Ical)
  - Dedicated technical meetings when needed
  - Reporting to Cores TG

- Mattermost channels
  - TWG : Cores : CV32E4*P
  - TWG : Verification

- Resources
  - Pascal Gouédo
    - Specification, Design, Verification & Formal
  - Yoann Pruvost
    - Design, Verification & Formal
  - Xavier Aubert
    - Verification leader
  - Vaibhav Jain
    - Verification
  - Bao Shan Mak
    - Verification

- OpenHW staff
  - Mike Thompson
    - Verification support
  - Davide Schiavone
    - Architecture & Design support

User Manual

- Updates since cv32e40p_v1.3.0 creation
  - v1.3.1
    - HWloop start and end address registers 2 LSBs hardwired to 0
  - v1.3.2
    - Fixed some instructions ambiguity, corrected cv.avg emulation
    - Added pseudo-instructions section and re-ordered all sections
    - Added a note about post-incremented loads GPR write priority
    - Added a constraint about HWloop start, and and setup instructions alignment
    - Changed parameters values for Zfinx CSR presence
  - v1.4.0
    - Updated mimpid description
    - Renamed cv.slet/cv.sletu to cv.sle/cv.sleu
    - Aligned Post-Increment Load/Store instructions syntax with respect to proposals and final vote
    - Moved Synthesis Guidelines from Introduction section to Core Integration one
  - v1.4.1 (to create once PRs merged in master)
    - Added some additional information on HW loop constraints
    - Verification section elaborated (a little bit)

Formal Verification

- RV32IMC[F|Zfinx]X_Zicsr_Zifencei (all but DIV/SQRT results)
  - 18 bugs
    - F instructions decoding
    - F instructions result impact on pipeline
    - FFLAGS not correctly set
    - Missing MSTATUS.FS and SD (when Zfinx = 0)
    - FMUL.S wrong result
- Intensive debug and bug correction
  - All bugs corrections confirmed formally and pushed on git repo

Design

- CVFPU v0.8.0 release created
  - Integrated new single-precision divider from T-Head OpenE906
  - 2 bugs corrected (#86 #90)
- CVFPU v0.8.1 release created
  - Fix Underflow flag for MUL and DIV/SQRT operations (#94 #726 #729)
  - Fix for Float to Int conversion (#97 #83 #727)
- Core
  - Moved from PULP_ZFINX to v1.0 ZFINX, moved PULP_XPULP to COREV_PULP, PULP_CLUSTER to COREV_CLUSTER in all RTL files
  - Updated MISA value and make MSTATUS.FS/SD ZFINX dependent
  - Renamed cv32e40p_wrapper to cv32e40p_top, added a clock gating cell for FPU in cv32e40p_top
  - HWloop start and end address registers 2 LSBs hardwired to 0
  - Added new ZFINX CSR
  - Aligned uhartid, privlv and zfinx CSR addresses to v1.3.0 user manual
  - HW loop count decrementing to 0 after the end of body execution
  - Removed HW loop constraints hardware check
  - Vendorized CVFPU 0.8.0 and then 0.8.1
  - Corrections of 18 bugs found by formal verification

Tools

- SW toolchain deliveries:
  - 4 releases since April
  - Remaining
    - Automatic generation of HW loop instructions by GCC compiler

- Imperas
  - 5 Reference Model releases since April
  - 2 PULP riscvISACOV coverage files releases since April

Verification (1)

- Verification test plans
  - Mike & Mario feedback integration on-going
- core-v-verif environment
  - COREV-DV supporting all PULP instructions and Zfinx extension
    - Scenarios implemented to match DV Plan requirements (~2000 tests)
  - Switch to ImperasDV pushed in central repo
    - Switch to RVVI interface took way longer than planned but this work is behind us
  - Instructions and Functional Coverage
    - Standard & PULP instructions riscvISACOV files received from ImperasDV. Enabled in core-v-verif with ISS
    - Coverage collection will start in coming weeks
    - Additional PULP functional coverage will be added to ISACOV according to test plans
  - Non-regression
    - Addition of Siemens EDA Verification Run Manager scripts
    - Flow consolidation still needed to merge 20+ non-regressions on 9 configurations

Verification (2)

- Architectural tests using RISCOF framework
  - Integrated in core-v-verif (for cv32e40p only)

    Successfully generated RV32IMCF_Zicsr_Zifencei html reports with latest RTL

    236 Passed, 0 Failed
- Non-regression results
  - 3 non-regression ran on 7 configurations
    - Total of 20 non-regressions
  - 1608 tests run
  - 893 failing tests (~45% OK)

Core overview

Core	TRL	Privilege	ISA	Debug	Interrupts	Bus	Gate count (min, typ, max)	CoreMark / MHz (min, typ, max)	Target date
CV32E20	5	M, U (v1.11)	RV32I, RV32E, C, M, Zicsr	0.13.2	CLINT	OBI	14K, 19K, -	-, 2,47, -	2023 Q1 (per August 2022)
CV32E40P	5	M (v1.11)	RV32I, C, F, M, Xpulp, Zicsr, Zfinx, Zifencei	0.13.2	CLINT	OBI	-, -, -	-, 2.91, -	2023 Q1 (per October 2022)
CV32E40S	5	M, U (v1.12)	RV32I, RV32E, C, M, Xsecure, Zba, Zbb, Zbc, Zbkc, Zbs, Zca, Zcb, Zcmp, Zcmt, Zicsr, Zifencei, Zkt, Zmmul	1.00	CLINT, CLIC	OBI	-, -, -	-, 2.91, 3.12	2023 Q4 (per August 2023)
CV32E40X	5	M (v1.12)	RV32I, RV32E, A, C, M, Xif, Zba, Zbb, Zbc, Zbkc, Zbs, Zca, Zcb, Zcmp, Zcmt, Zicntr, Zihpm, Zicsr, Zifencei, Zkt, Zmmul	1.00	CLINT, CLIC	OBI	-, -, -	-, 2.91, 3.12	2023 Q4 (per August 2023)
CV32E41P	3	M (v1.11)	RV32I, C, F, M, Zca, Zcb, Zcmb, Zcmp, Zcmt, Zfinx	0.13.2	CLINT	OBI	-, -, -	-, 2.91, -	Not in active development
CVA5	3	M, S, U	RV32I, A, M				-, -, -	-, -, -	Not applicable
CVA6	5	M, S, U (v1.10)		0.13.2	CLINT	AXI4	-, -, -	-, 2.93, -	Unknown

Thank you!