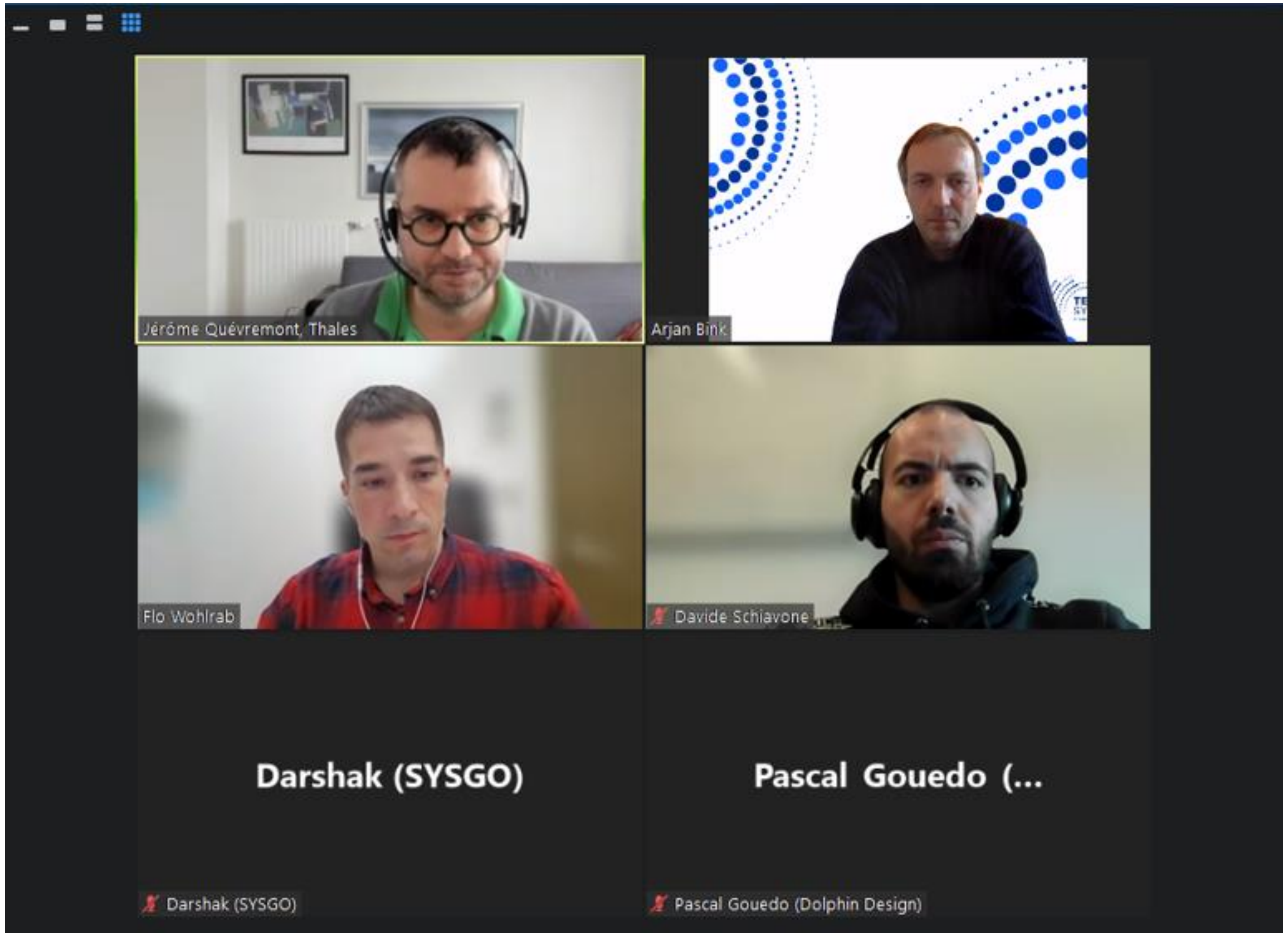
CORES TG – March 4 2024

Arjan Bink

Jérôme Quevremont

Davide Schiavone

Attendance


Jérôme Quévremont, Thales
Arjan Bink
Flo Wohlrab
Davide Schiavone
Darshak (SYSGO)
Pascal Gouedo (...
Darshak (SYSGO)
Pascal Gouedo (Dolphin Design)

Agenda

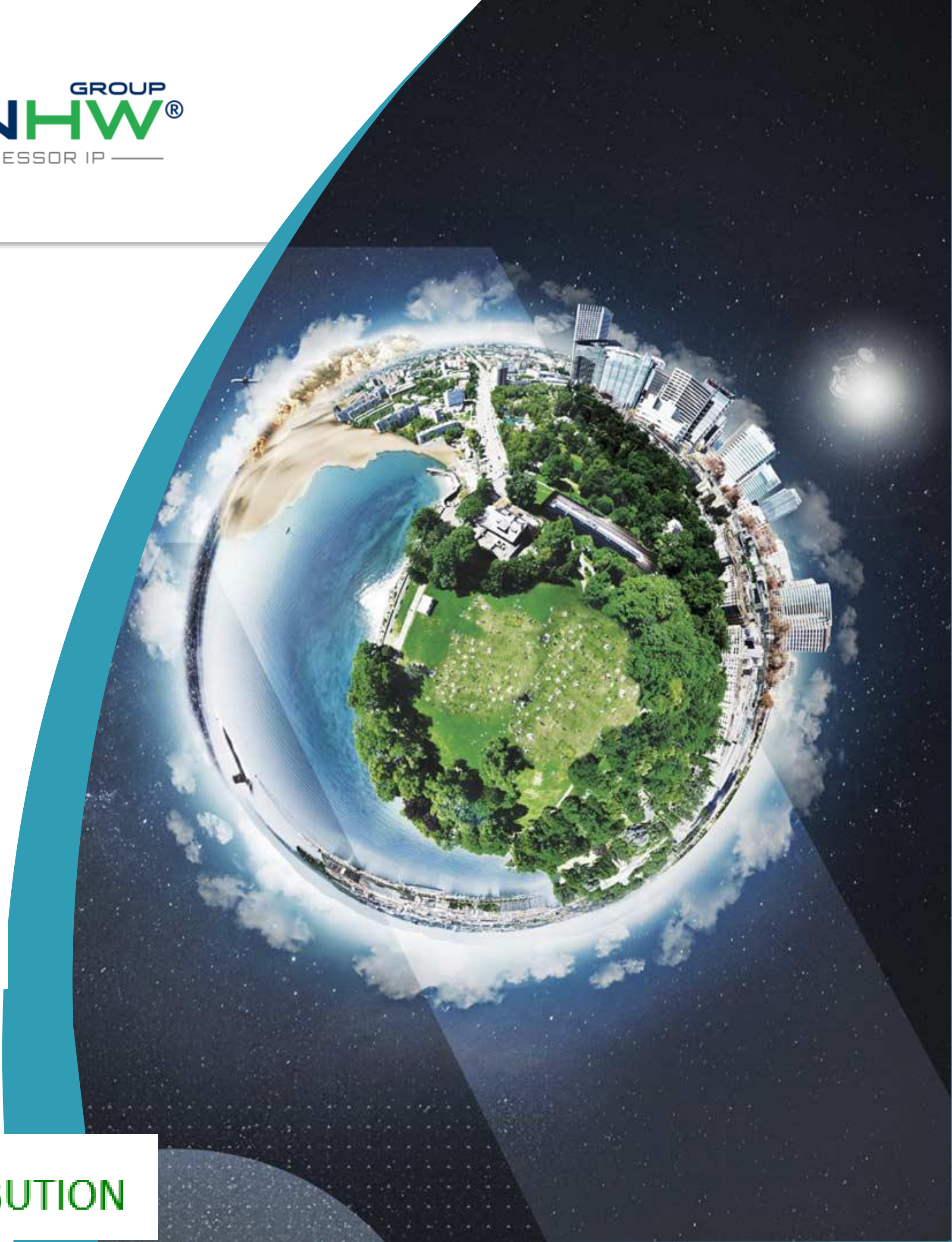
- CV-X-IF specification available for review
- Error detection & correction in CVA6 memories (Jérôme)

Core-V eXtension interface (CV-X-IF)

- CV-X-IF specification available for review
- https://github.com/openhwgroup/core-v-xif/releases/tag/v1.0.0-rc.1
- Please provide your comments until March 15th, 2024, AoE (Anywhere on Earth), either as issues or pull requests in the https://github.com/openhwgroup/core-v-xif repository.

Building a future we can all trust

Follow-up :
ECC implementation



Goals

Organization	contact	Goals / Needs
Axelera	Florian zaruba	Product requirements, bit flips are likely , execution protection is required. No certification required.
Bosch-FR	Nicolas.tribie	Use CVA6 in automotive applications : Reach ASIL B safety level in the context of ISO26262 standard, using memory protection + interconnect protections
CEA	Cesar.FUGUETTORTOLERO	Automotive and HPC need protection against bit flips, this is a request from industrial partners
Pulp ETH-Zurich, U Bologna	nils.wistoffMichael Rogenmoser	Leverage the CVA6 design for a reliable architecture targeting space applications. Protection against single-event upsets (SEUs, bitflips) throughout the entire SoC design. These are, of course, more research objectives. The design will be open-source and may be used in further projects
Thales DIS	jean-roch.coulon	Detection but no correction. Same needs. Redundant bits in buses and in caches. Use USER bits from AXI-4. WT caches. No correction mechansim. (left to user : SW, HW).
Thales BUs	Jerome Quevremont	Needs for error correction and detection (SECDED). Some applications need DO-254. Space applications as well. Some FIT requirements. No correction needed in WT mode.

Idea / context

- **Add ECC to Caches**
  - I$: legacy
    - Florian: Parity only ?
    - How to interface fault detection ? CSRs ?
  - D$: WT cache, WB cache, HPDCache ?
    - JR: if we evaluate HPDCache OK for CVA6 (area, perf), we'll select this one. TBC
    - Cesar: Can help to provide the proper parameters HPDcache
    - Bosch: likely HPDCache. Same cache for petit+grand Robert
    - Florian: WB cache, open for HPDCache.
    - Jerome : short terms needs for NeuroSoc : WT cache . Open to HPDCache. 3 yrs experience with the WT Cache .
    - Having a common cache eases application
    - ECC mechanism should be implemented in the cache to benefit from its pipelined implementation

    - Michael (ETH) : student project : protection of an icache for one of our cluster systems. Implementing error detection with parity bits and correction through invalidation and refetching proved very effective, unless error rates are far too high (unlikely for the space application) or there are different requirements on how static the data is (e.g. handshake disappearing if error happens during readout, also very unlikely).

  - MMU: Needs to be protected (shared TLB memory array between ITLB and DTLB). Shared TLB implemented in all configs by planV. Detection needed only, correction can be avoided by invalidation.
  - Write buffer + MSHR + Replay table (less sensitive (duration) needs protection (Cesar, HPD cache))

  - Scratchpads :
    - I+ D ? Yes, SECDED needed
    - Read/modify/write required to support byte access

The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Follow-up : ECC meeting 29/01/2024

- Writethrough : ECC fault => invalidate and reload later from DRAM
  - Not specified in DO-254 (no solution description) . ISO26262 ?

- Writeback :   ECC fault => correction required (SECDED) :
  - 32 bits, 7 bits ECC
  - 64 bits, 8 bits ECC
  - Depends on the algorithm
  - Might need different memory width (tags, cache line size Read/Modify/Write). Needs a customized algorithm
  - Hamming codes : https://en.wikipedia.org/wiki/Hamming_codeTAGs protection, data array protection , ECC = f (data, address)

- **Interconnect** :
  - reserve USER bits to carry ECC
  - Data needs to be reencoded in the cache
  - Bus ECC should be distinct from memories (cache, arrays, ..)
  - Needs Read/Modify/Write in the memory controller
  - No retry on AXI - available on CHI so parity can be enough , request can be retried
  - Interconnect can't be modified (IPs)
  - Florian doesn't need to secure the data channels – only memory arrays.

The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Follow-up : ECC meeting 29/01/2024

- **Error reporting** mechanism :

  - Data transfer (SLVERR) support required (currently ignored in axi_shim?) : Florian confirms : TBD
  - Cesar : changes done for VRP. Errors are reported for reads
  - Interrupt : Cesar (cacheable writes: asynchronous, uncacheable: synchronous (comes back with write ACK)) ,
  - Olivier : add a signal to report ECC errors.
  - HPDCache reports errors in CSR.
  - See RAS (error controller)
    - spec from RISC-V taskgroup : https://github.com/riscv-non-isa/riscv-ras-eri
    - See also U.Pisa Daniele Rossi : *RAS improvement via Error Logging and Reporting in VEC RISC-V Accelerator developed within EPI SGA-2 Project* - **Daniele Rossi** (University of Pisa)
    - https://www.researchgate.net/publication/374199416_HW-SW_Interface_Design_and_Implementation_for_Error_Logging_and_Reporting_for_RAS_in_RISC-V_Architectures

  - Florian : interrupts
  - Imprecise exception :
    - should we defined a new exception + new CSR ? Similar to a load exception
    - Cesar : implemented new exception

  - Michael (ETH) : We decided to handle bus errors not caught by CVA6 and implemented a hardware unit to track and log these errors. If configured and connected correctly, UNBENT (https://github.com/pulp-platform/unbent) will trigger an interrupt of the core when an error occurs on the bus, and subsequently also provide the information about the request (essentially Ax information from the AXI transaction) through a memory-mapped register. Furthermore, we are looking to develop some more advanced interconnect protection mechanisms to ensure tolerance to SEUs. However, these interconnects may not be 100% standard compliant, but still offering proper adapters to the standard pinout of the bus architectures (AXI).

The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Follow-up : ECC meeting 29/01/2024

WORKSHOP MEETING CVA6 TRUST BUT VERIFY 16-18 JANUARY 2024 AIX - MARSEILLE

- **Error reporting (follow up):**
  - **Precise exception** : need ? Feasability ?
    - Easy for reads,
    - More difficult for write: context needs to be saved
    - To be refined with SW dev (Florian : no concerns from SW team) , ISO-26262
    - Huge perf impact on writes
    - Could be done with reduced perf impact by merging the WB and the ScoreBoard.

Multicore and ECC :

- OpenPiton :
  - not fit for time critical applications . Latency impredictable so globally not fit for safety.
  - Need to check with J.Balkind
- Culsans : still WIP

Timelines :

- CEA : short and mid-term
- Thales :
  - short term need (EU project) 1Q24 with WT cache
  - long term (product) with HPDCache(?)
- Axelera : feature freeze. Feb, March ’24. RTL freeze : June’24
- Bosch FR:
  - RTL implementation idea 1Q24
  - Demonstrator 1Q25
  - ETH: (Michael) Furthermore, we are also looking to augment CVA6' protection by implementing a hybrid dual-core lockstep mechanism. To follow this up, we intend to look a bit into porting this mechanism to CVA6 icache, and launch a student project looking into how to protect the data cache of the system, although this is not 100% defined yet.

The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Follow-up : OpenHWGroup CVA6 meeting 14/02/2024

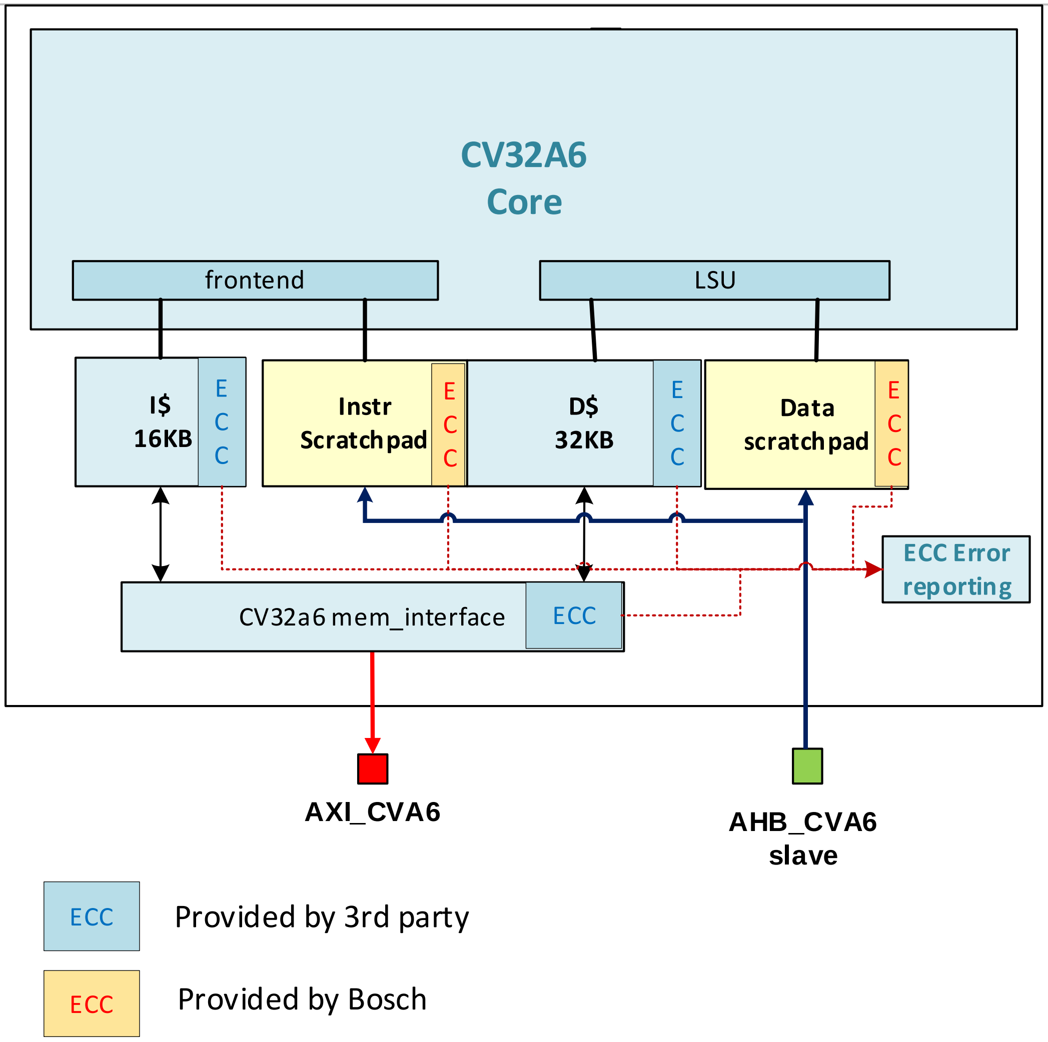
BOSCH FR implementation split proposal:

Bosch FR implements :

- ECC support in Scratchpads

ECC support integrated from partners :

- D$ (HPDCache) ?
- I$
- Memory bus



The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Follow-up : OpenHWGroup CVA6 meeting 14/02/2024

- **TBD D$ choice** : HPDcache, WT Cache, WB Cache ?
  - If WT : in case of error : invalidate, fetch again
  - If WB : correction required , local data only
  - HPDCache : WT/WB configurable : ?
    - Mixed criticality: critical data in WT , non critical in WB
    - WT is an advantage in single core where memory is not the bottleneck : invalidate => faster context switch
    - WB: correction could stall the core.
    - Multicore is possible in WT using OpenPiton, which provides L1.5 / L2.
  - HPDCache assessment needs to be done by partners.
    - Bosch – FR : ongoing
    - 10x : WB is the only option – easier fror verification. If timeline allows, HPDcache would be a good option
    - TDIS is evaluating the HPDCAche in WT in a small 2ways configuration. Working with Cesar to tune parameters to get a smaller area. Ongoing work, no definitive conclusion.

- **Memory bus** : do we need to protect in flight data ? Still need to identify the ressources to do it.
- **DCLS** : Coexistence between ECC and DCLS needs to be addressed.
- **HPDCache** : Cesar @ CEA: WT/WB mode and scratchpad mode have higher priority, but open to investigate ECC implementation as a lower priority.
- As an alternative due to timelines constraints, ECC will likely be implemented on WT/WB caches, and then on HPDCache.
- **I$ implementation** : might be done by Thales TRT. Similar to WT, only error detection is required.
- **Error handling mechanism** :
  - TBD : Exception, interrupt
  - spec from RISC-V taskgroup : https://github.com/riscv-non-isa/riscv-ras-eri

The TRISTAN project, nr. 101095947 is supported by Chips Joint Undertaking (CHIPS JU) and its members Austria, Belgium, Bulgaria, Croatia, Cyprus, Czechia, Germany, Denmark, Estonia, Greece, Spain, Finland, France, Hungary, Ireland, Israel, Iceland, Italy, Lithuania, Luxembourg, Latvia, Malta, Netherlands, Norway, Poland, Portugal, Romania, Sweden, Slovenia, Slovakia, Turkey .

Thank you!