The RISC-V Klessydra Orbital Lab project

13th ESA ADCSS2019 Workshop, Noordwijk

SAPIENZA
UNIVERSITÀ DI ROMA

M. Olivieri, F. Menichelli, A. Mastrandrea,
A. Cheikh, F. Vigli, L. Blasi

DSI
Aerospace Technology

presenter:
Ms.C. Luigi Blasi,
Ph.D. Candidate, Sapienza University of Rome
FPGA Designer, DSI Aerospace Technologie GmbH

Presentation Outline

- Space Environment
  - Particle sources
  - Radiation Effects on FPGAs
- Fault-Tolerant design
  - Radiation Hardening strategies
- Klessydra RISC-V Core family
  - Overview
  - Development roadmap
  - Multithreading architecture
  - Fault-tolerant Architectures
  - Implementation Results
- The Klessydra Orbital Lab (KOL)
  - Motivation
  - Design Concept
  - Architecture and Mission Overview
- Results and Future Work

DSI Aerospace Technologie GmbH is an SME located in Bremen, Germany which provides following electronic equipment:

Platform & Instruments Computers

Payload Data Handling Units (incl. MMBs)

Data Processing Units

Test Systems (EGSE)

Engineering Services

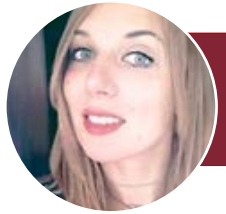
Aerospace Electronics

The Digital System Lab at Sapienza University of Rome

Mauro Olivieri
Associate Professor

Abdallah Cheikh
PhD candidate

Francesco Menichelli
Assistant Professor



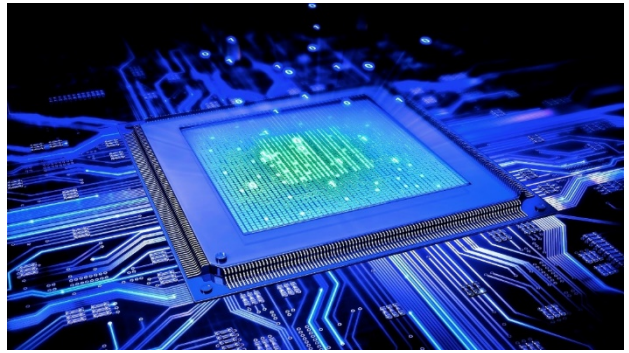
Giulia Stazi
PhD cand. @UTC inc.

Antonio Mastrandrea
Research Fellow

Luigi Blasi
PhD cand. @DSI Gmbh

Stefano Sordillo
PhD candidate

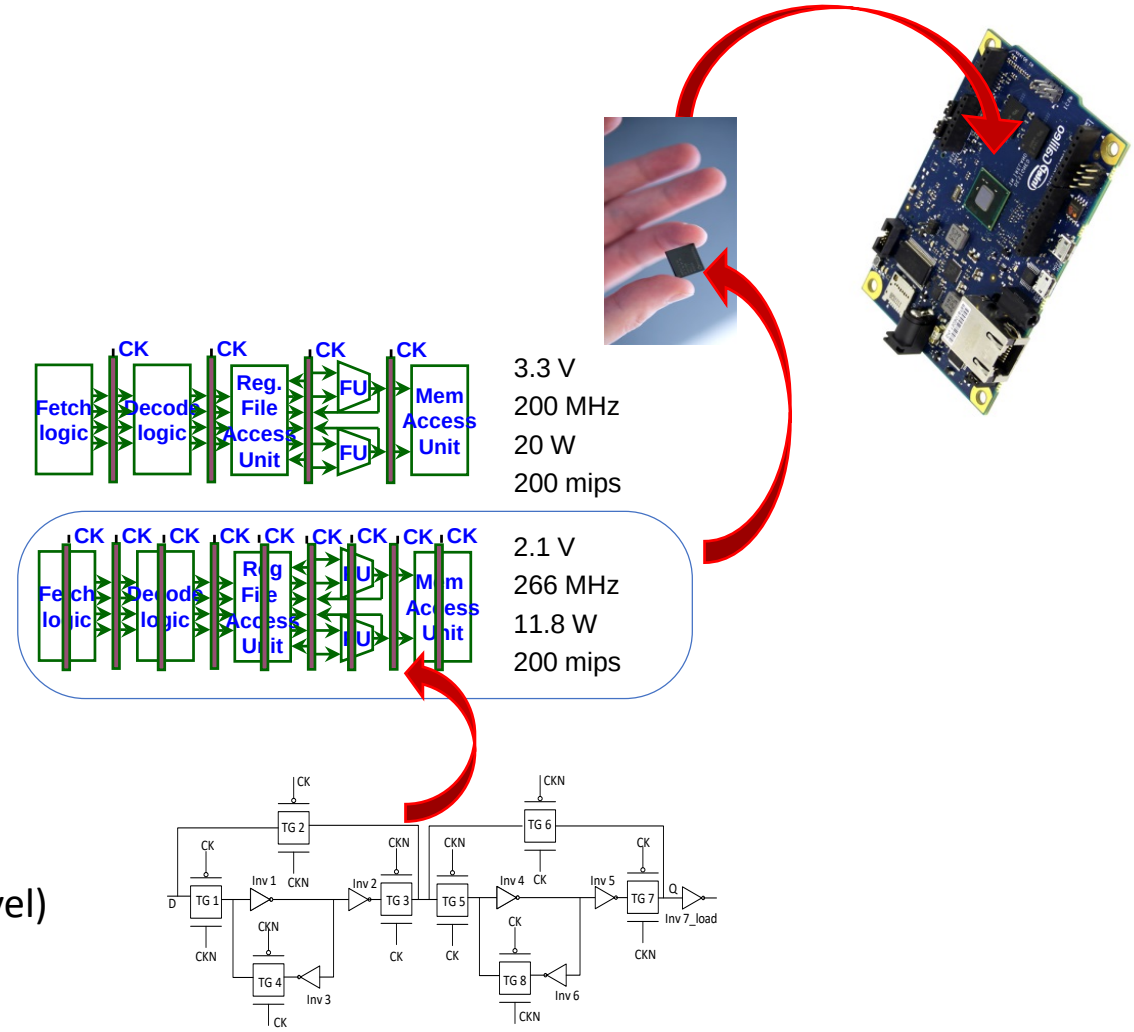
Francesco Vigli
PhD cand. @ ELT Spa



SAPIENZA
UNIVERSITÀ DI ROMA

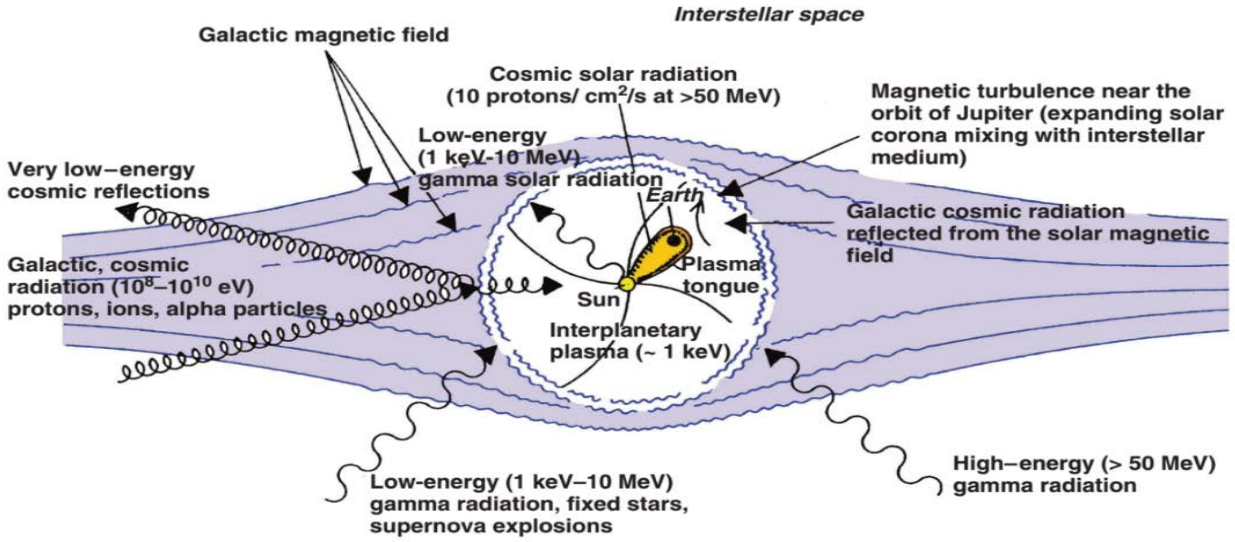
The Digital System Lab at Sapienza University of Rome

- **Advanced embedded HW/SW development**
  - Linux, Embedded Linux, kernel programming
  - ARM, PIC, STM32 system development
  - Zigbee, LoRaWAN for IoT
- **Architecture modeling/analysis** (single- & multi-core)
  - RISC-V, ARM, Leon, ST200, domain-specific (e.g. fuzzy, DSP, approximate computing)
  - Modeling approaches: Qemu, SystemC, C, VHDL
- **RTL semi-custom IP design**
  - RISC-V cores, arithmetic units, dedicated units, multithreading support, energy/speed tradeoff
  - FPGA and ASIC flow (primarily VHDL based)
- **Circuit and full custom design**
  - CMOS cell design, timing, power, reliability (SPICE level)
  - Self-timing, delay insensitive circuits



The Space Environment and Particle Sources

- The greatest challenge for electronics operating in space is ionizing radiations
  - A particle is **ionizing** if has the capability of dividing a stable atom into ions
- These particles can be classified in three major types:
  - Energetic charged particles (e.g. electrons, protons, heavy ions)
  - Electromagnetic radiation (X-rays, γ-rays, UV-rays)
  - Uncharged particles (neutrons)
- The main sources of energetic particles can be resumed as follows:
  - Protons and electrons trapped in the Van Allen belts
  - Heavy Ions trapped in the magnetosphere
  - Galactic Cosmic Rays (GCR) and Solar Particle Events (SPE)

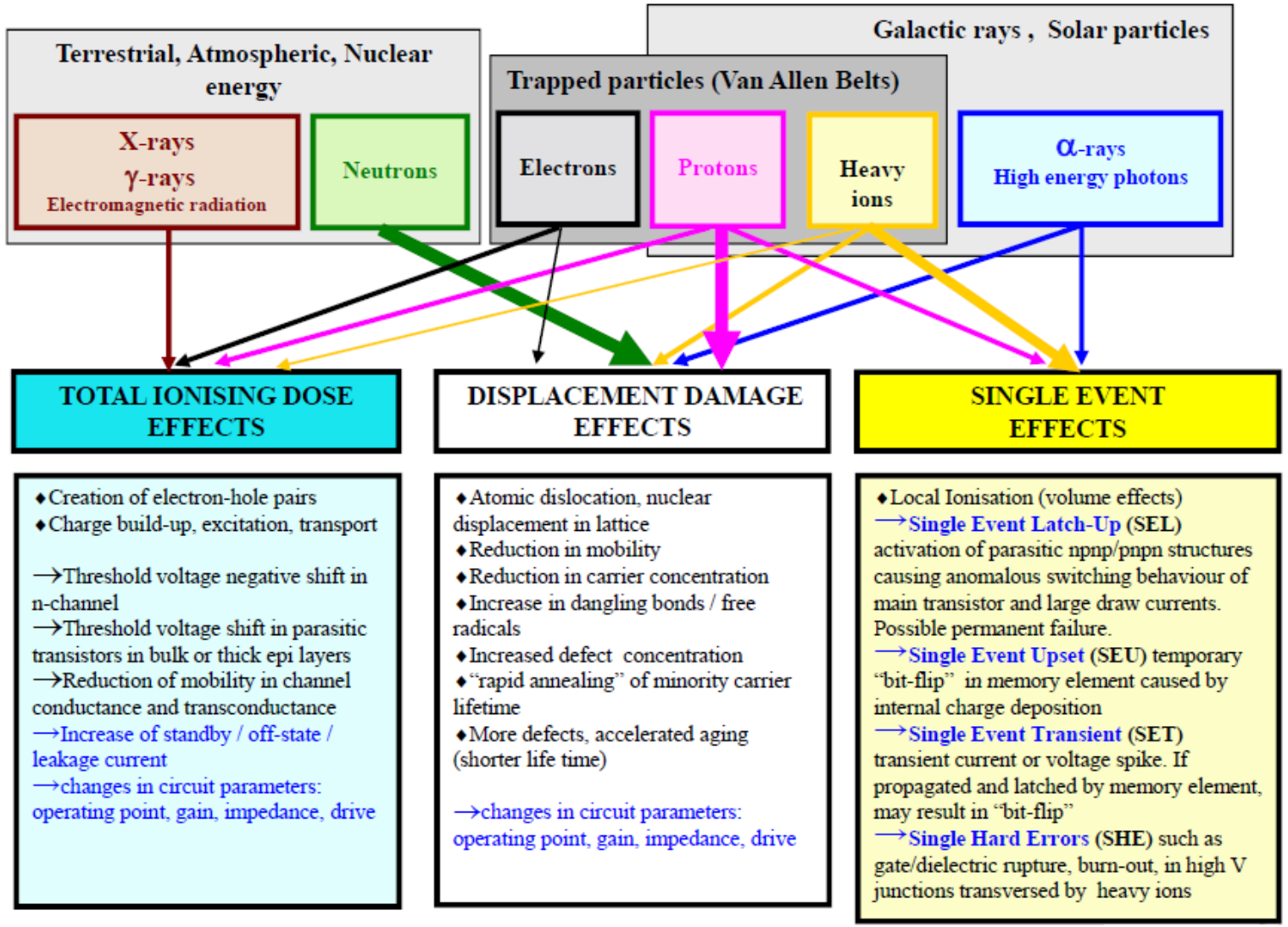


The Radiation Effects on FPGAs: An overview

Ionizing radiations can generate failures in electronic devices, since the deposited charge may perturb a transistor state. The charge may be deposited directly if the ionizing particles is charged (electrons, protons and heavy ions) or indirectly (neutrons).

- **Cumulative effects** cumulative damage of the semiconductor lattice caused by ionizing radiation over the exposition time. They causes slow gradual degradation of the device's performance and characteristics.
  - Total Ionizing Dose (TID)
  - Total Non-Ionizing Dose (TNID)
- **Single Event Effects (SEE)**
  - Disturbance to the normal operation of a circuit caused by the passage of a single ion (proton or heavy ion) through or near a sensitive node in a circuit.
    - Transient Effects (Soft Errors)
    - Permanent Effects (Hard Errors)

Radiation Effects on Electronics: resume picture

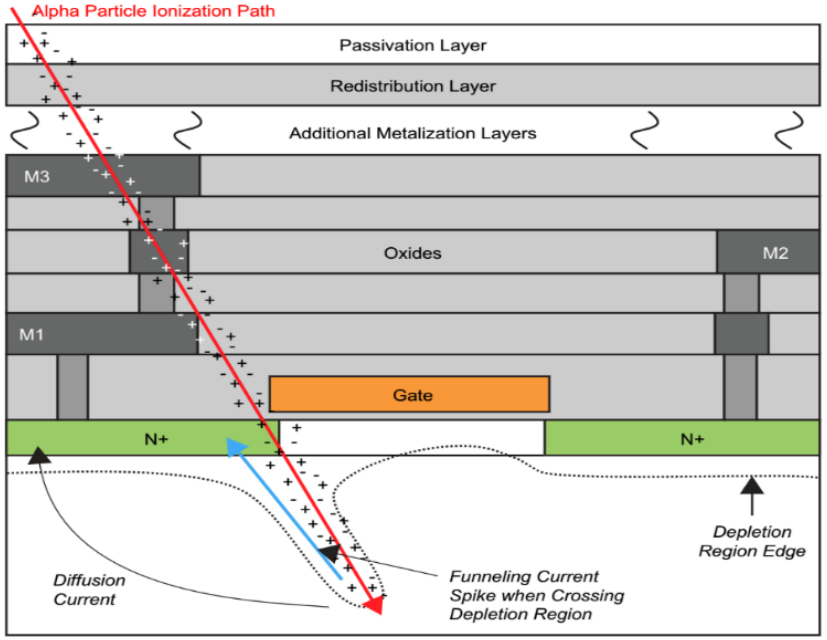


Courtesy of ESA

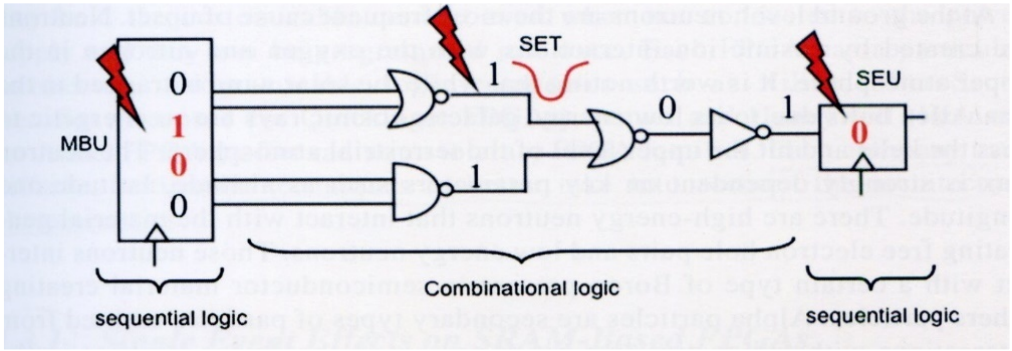
The Radiation Effects on FPGAs: SEE Soft Errors

Transient Effects (Soft Errors)

- Single Event Upset (SEU)
  - a change of state of a storage element induced ed by an energetic particle
- Single Event Transient (SET)
  - a current transient induced by passage of a particle, can propagate to cause output error in combinatorial logic
- Single Event Functional Interrupt (SEFI)
  - a condition where the device stops operating in its normal mode, and usually requires a power reset or other special sequence to resume normal operations.
- Multiple Bit Upset (MBU)
  - an event induced by a single energetic particle such as a cosmic ray that causes multiple upsets or transients during its path through a device



Alpha Particle: Helium Nucleus (2 Neutrons & 2 Protons)
Need a Charged Particle to Cause Ionization

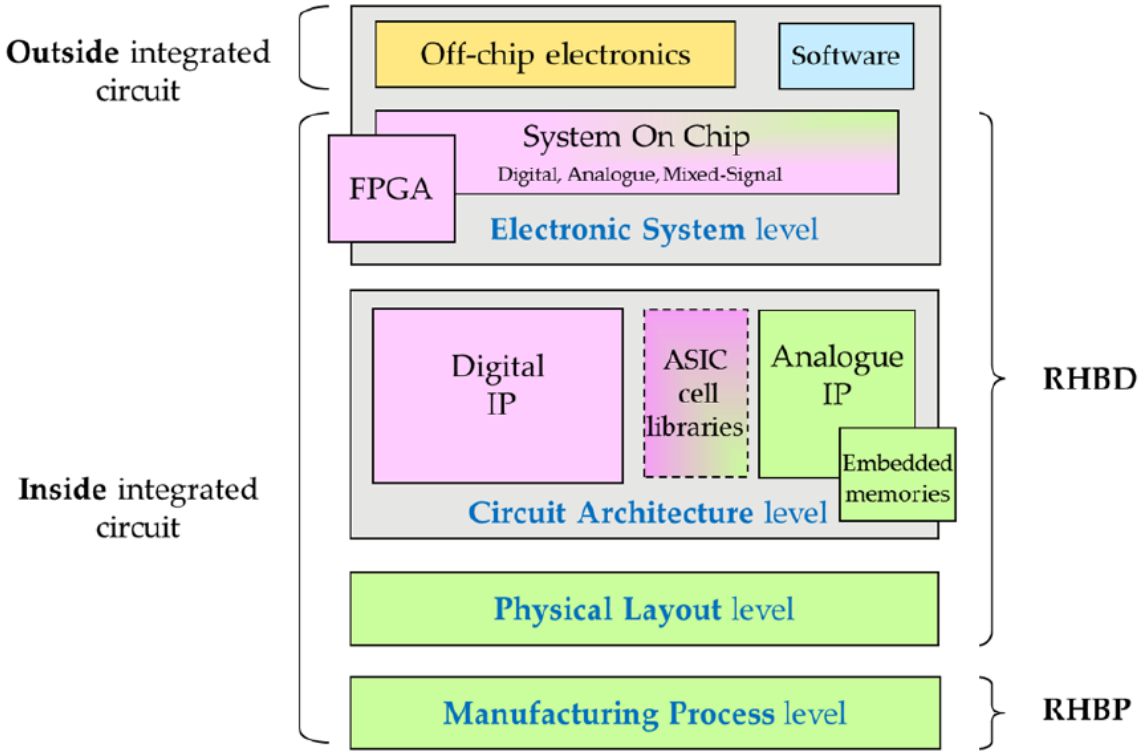


Fault-tolerant design: key concepts

- It is the property of a system to continue operating properly in the event of the failure of (or one or more faults within) some of its components.
- A fault-tolerant design enables a system to continue its operation, possibly at a reduced level, rather than failing completely, when some part of the system fails.
- A number of choices have to be examined to determine which components should be fault-tolerant
  - **How critical is the component?**
  - **How likely is the component to fail?**
  - **How expensive is it to make the component fault tolerant?**

Radiation Hardening strategy: RHBP vs RHBD

According with the **ECSS-Q-HB-60-02A** : Space Product Assurance " Techniques for radiation effects mitigation in ASICs and FPGAs handbook"



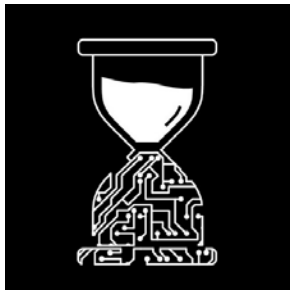
Courtesy of ESA

- **Choosing the best radiation hardening strategy** is a trade-off
  - technical requirements and goals
  - costs benefits
  - development time
  - resources constraints
- **Radiation hardening** design techniques
  - Radiation Hardening By Design (RHBD)
    - Techniques that can be applied at physical layout level in order to mitigate radiation effects
  - Radiation Hardening By Process (RHBP)
    - such modifications at IC manufacturing process level in order to reduce radiation impact on integrated circuits

Radiation Hardening Techniques Overview

- **Technology selection & process level mitigation**
- **Layout**
- **Embedded memories**
- **Radiation Hardened cell libraries**
- **Digital Circuits**
  - Spatial Redundancy (TMR,DMR)
  - Temporal Redundancy
  - Fail-safe dead-lock FSM
  - Selective use of logic cells, clock and reset lines hardening
- **System on Chip**
  - Error Detection and Correction Codes (EDAC)
  - Memory Blocks Mitigation (Bit Interleaving and Data Scrubbing)
  - Watch-dog Timers
- **Field Programmable Gate Arrays (FPGA)**
  - Local TMR
  - Global TMR
  - Large Grain TMR
  - Embedded user memory TMR
  - Embedded processor protection
    - Software-based redundancy (task, application)
    - Spatial redundancy (Lockstep)

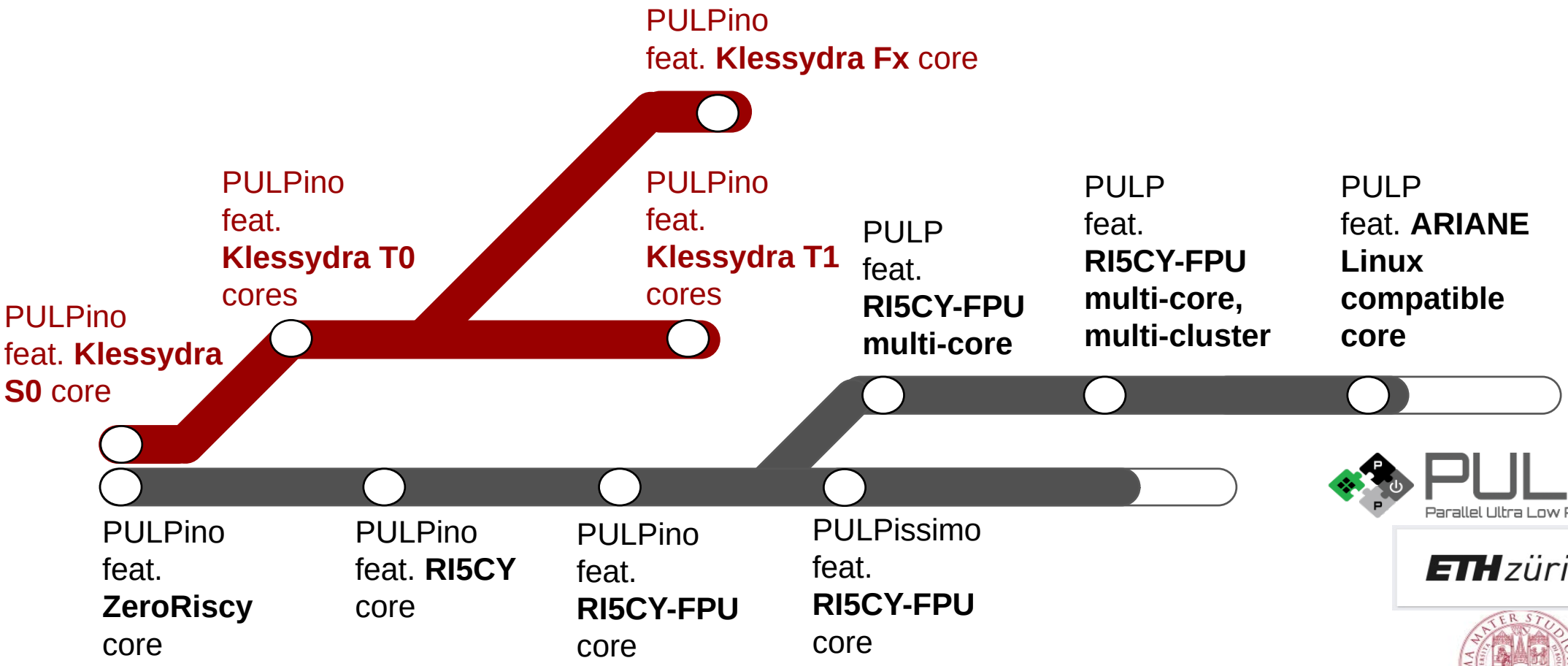
Klessydra: A RISC-V soft core family



- Soft IP core microprocessor compliant with RISC-V RV32I
  - Privileged ISA v1.10 (only M-Mode supported, NO OS)
  - Development toolchain
  - CSR subset (M-mode support)
  - Interrupt and Exceptions handling

- HW and SW compliance with the **PULPino Microcontroller SoC** hardware platform (ETH Zurich and University of Bologna)
- Processing based on an Interleaved Multithreading execution model with a 4-stages pipeline (F,D,E,WB) based on HARdware Threads (HARTs)
  - On every clock cycle a **new instruction is fetched** from a different HART, then decoded , executed and results are written back to the HART's register file.

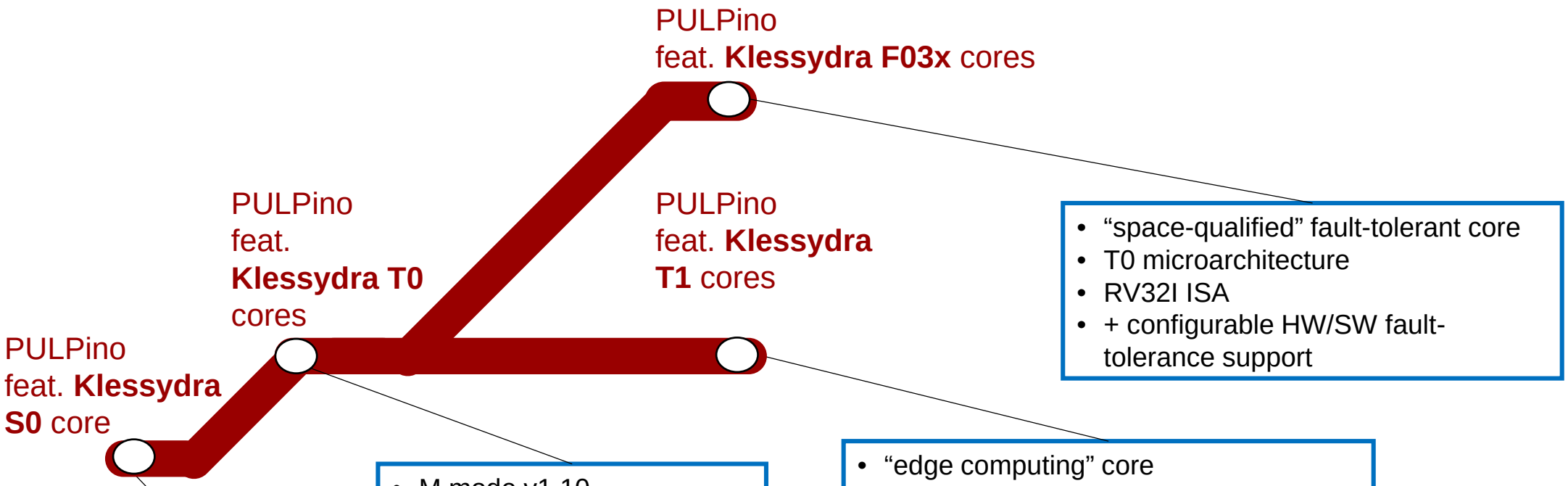
Klessydra Core Family: Development roadmap



PULP
Parallel Ultra Low Power

ETH zürich

Klessydra Core Family: Development roadmap (cont'd)

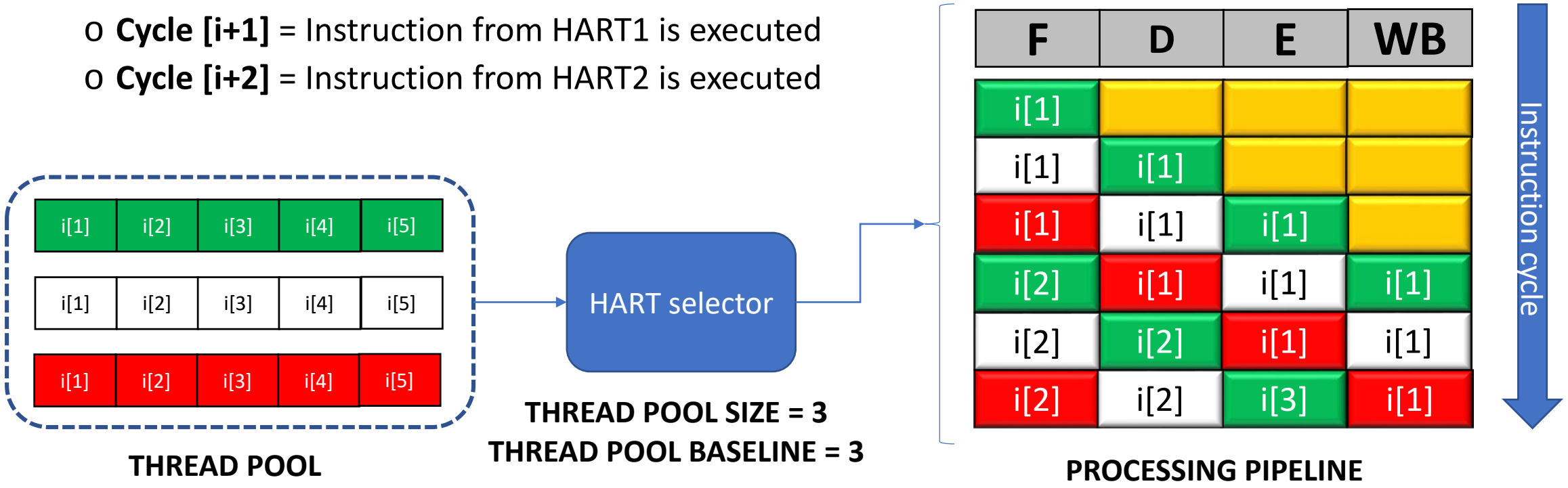


Klessydra Interleaved Multithreading : background concepts

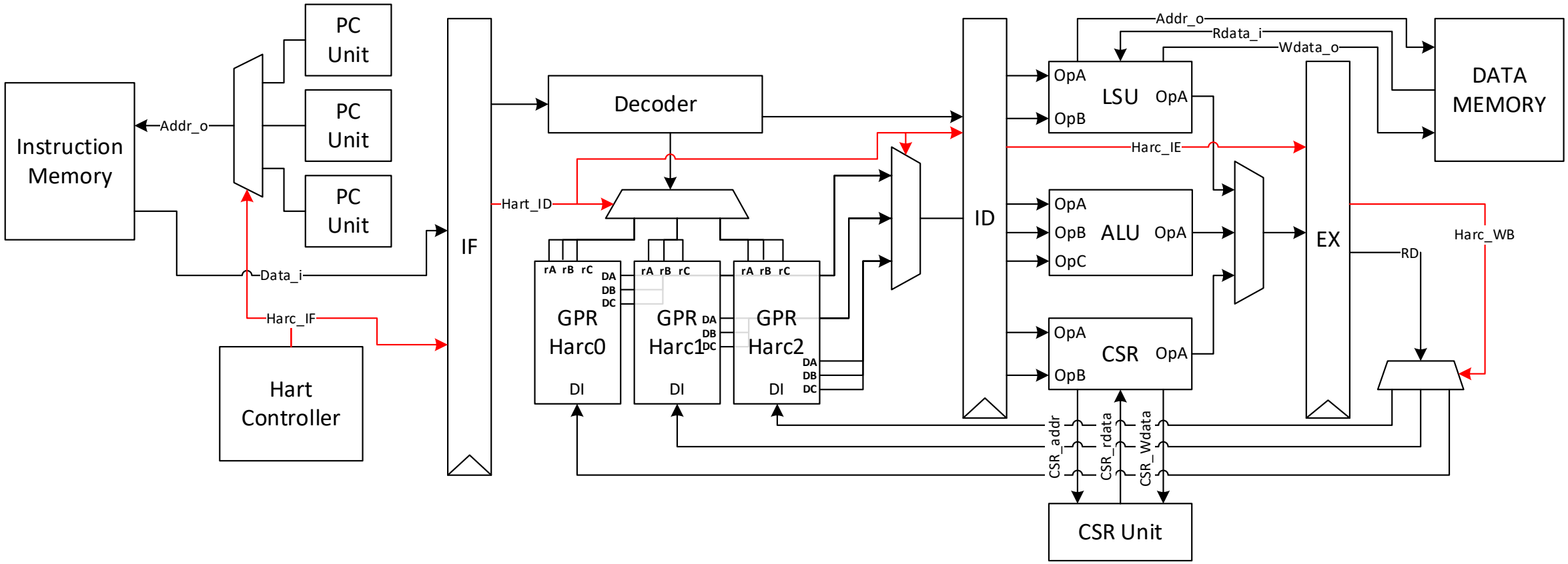
- **Thread**
  - A dispatchable unit of work within a **process**. It includes a processor context (Program Counter and Stack pointer) and its own data area for a stack (to enable subroutine branching). A thread **executes sequentially** and is interruptible so that the processor can switch to another thread.
- **Thread Pool**
  - A group of pre-instantiated, IDLE threads which stand ready to became ACTIVE.
- **HART (HARdware Thread)**
  - Thread with a dedicated hardware support (Register File and Program Counter)

Klessydra Interleaved Multithreading (cont'd)

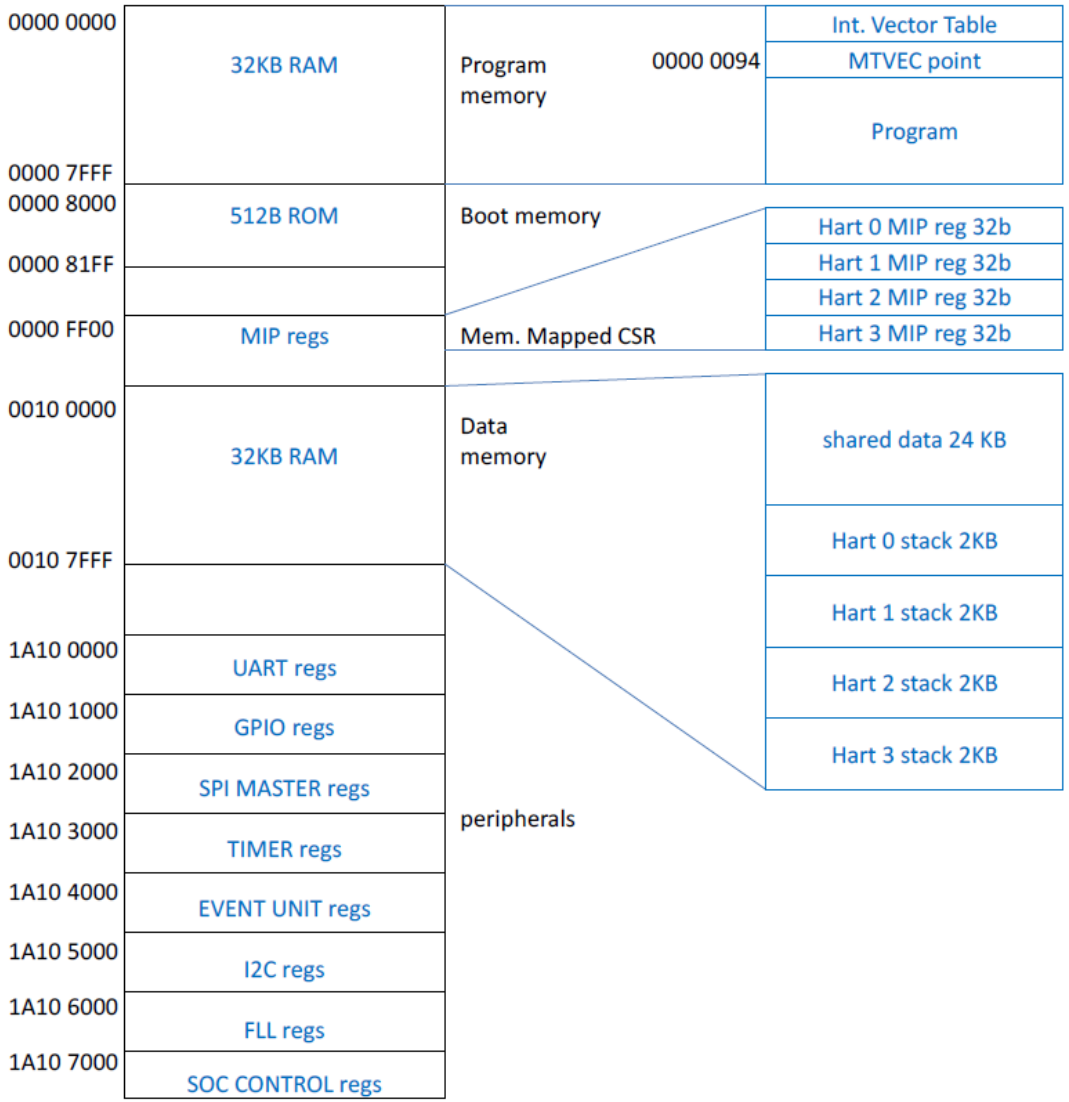
- **Key Concept:** removal of DATA DEPENDENCY, hardware simplification
  - Each thread is independent from other threads except for explicit shared memory accesses, so there is **no chance for an instruction** in decode stage needing an output from an older instruction still in the pipeline.
    - **Cycle [i+0]** = Instruction from HART0 is executed
    - **Cycle [i+1]** = Instruction from HART1 is executed
    - **Cycle [i+2]** = Instruction from HART2 is executed



The Klessydra Soft Cores: Microarchitecture Overview



The Klessydra Soft Cores: Memory Map



- **Each Hart** has its own register file which is accessible only by the related hart.
  - All HARTs share the same memory map (MM)
- **Each Hart has its own stack**, which has size and starting address customizable at SW level in the runtime system startup routine.
- **All Harts** can communicate using:
  - Inter-threads interrupts (MIP register)
  - Shared data memory

The Klessydra Soft Core: Pipeline Latency

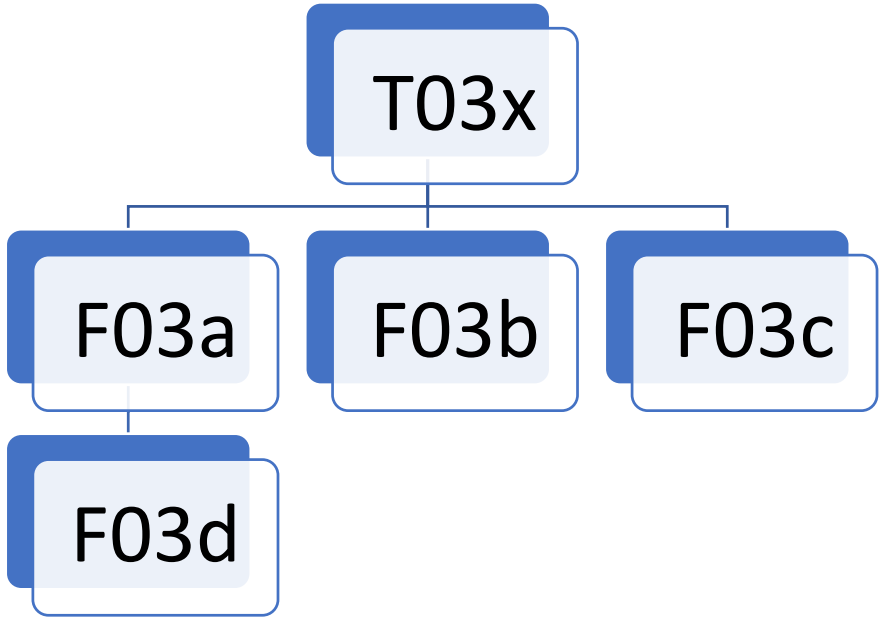
- Because of the **Interleaved Multithreading** architecture the BRANCH type instruction are considered always NOT TAKEN and they are executed in 3 clock cycles (delay slot)
  - Usually if a BRANCH is taken, the HW controller flushes all the instructions for a given thread already fetched inside the pipeline.
- In the **Klessydra T03x** version , the flush is **never performed** because of the interleaving factor is equal at least to 3, hence NO DATA HARZARDS occurs

	F	D	E	W
Load and store instructions	≥ 1	1	≥ 2	0
CSR instructions	≥ 1	1	≥ 2	0
Atomic memory operations	≥ 1	1	≥ 4	0
All other instructions	≥ 1	1	1	1

The Klessydra F03x Family: An Overview

- **Key features**
  - A fault-tolerant family derived from the **T03x** version, focused on RHBD HW and SW mitigation techniques (Circuit Architecture Level and Electronic System Level ref. ECSS-Q-HB-60-02A) to counteract the SEE which can affect the microcontroller core sequential and combinatorial elements.

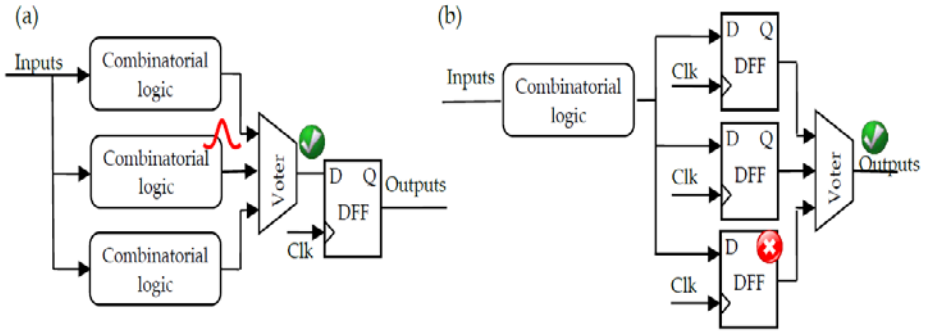
- **Implemented Architectures**
  - Klessydra F03a *TMR*
  - Klessydra F03b *CheckPoint Restore (CR)*
  - Klessydra F03c *Shadow Thread (ST)*
  - Klessydra F03d *Full-Weak TMR*

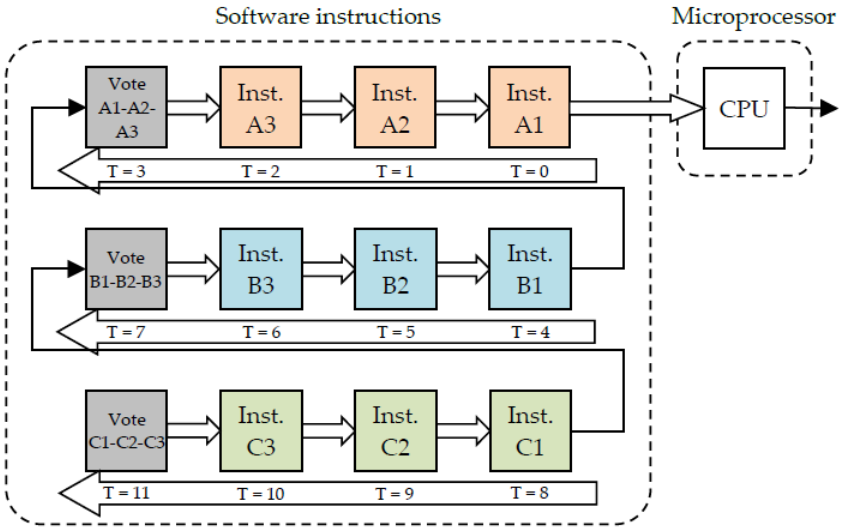


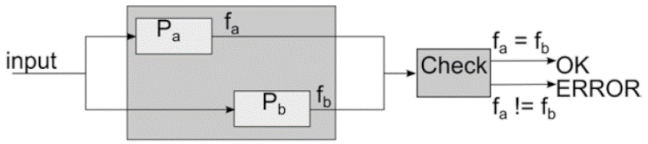
The Klessydra F03x family: fault-tolerant core design

Includes several fault-tolerant architectures using:

1. HW redundancy techniques (e.g. local-TMR)
2. SW redundancy techniques (e.g interleaved multithreading)
3. Dual Lockstep architecture using Check-point and Restore
4. Thread-Controlled Watchdog Timer module
5. Selective TMR protection (Full-Weak)





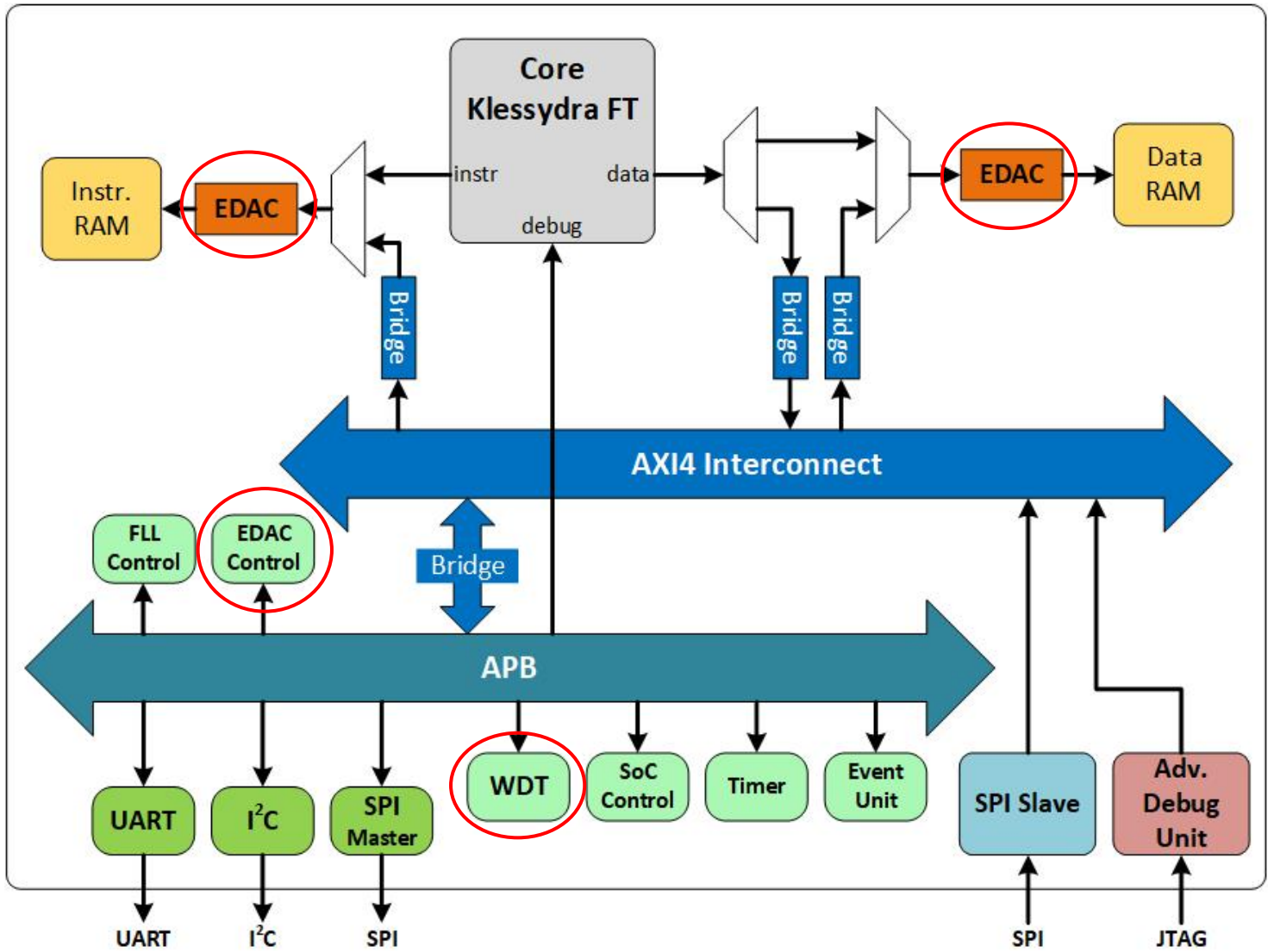



```
a0 = b0+c0;
a1 = b1+c1;
if (b0 != b1 || c0 != c1)
        error();
```

Adding fault tolerant features to PULPino-Klessydra SoC

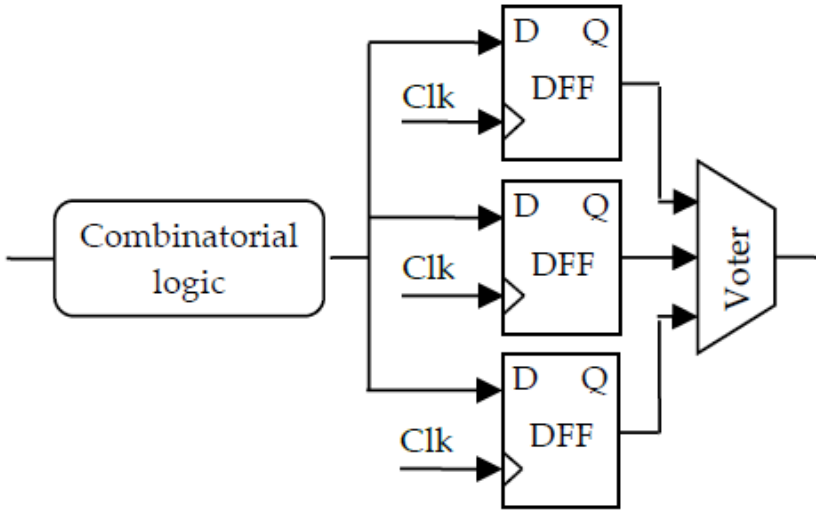
- **Data Memory** protection with a SEC-DED HAMMING(40,32) encoding scheme + Memory Scrubbing (
  - MM Scrubbing Control & Status registers access by APB I/F
- **Instruction Memory** protection with a SEC-DED HAMMING(40,32) encoding scheme + Memory Scrubbing
  - MM Scrubbing Control & Status registers access by APB I/F
- **Dedicated WDT** peripheral
  - MM Configuration register access by APB I/F
- **Bootloader Removal** in order to use it as a pure microcontroller (NO OS is supported)
- **Bitstream Protection** using
  - Xilinx® Bitstream Fallback
  - Xilinx® Readback CRC

The PULPino-Klessydra Fault Tolerant SoC Architecture

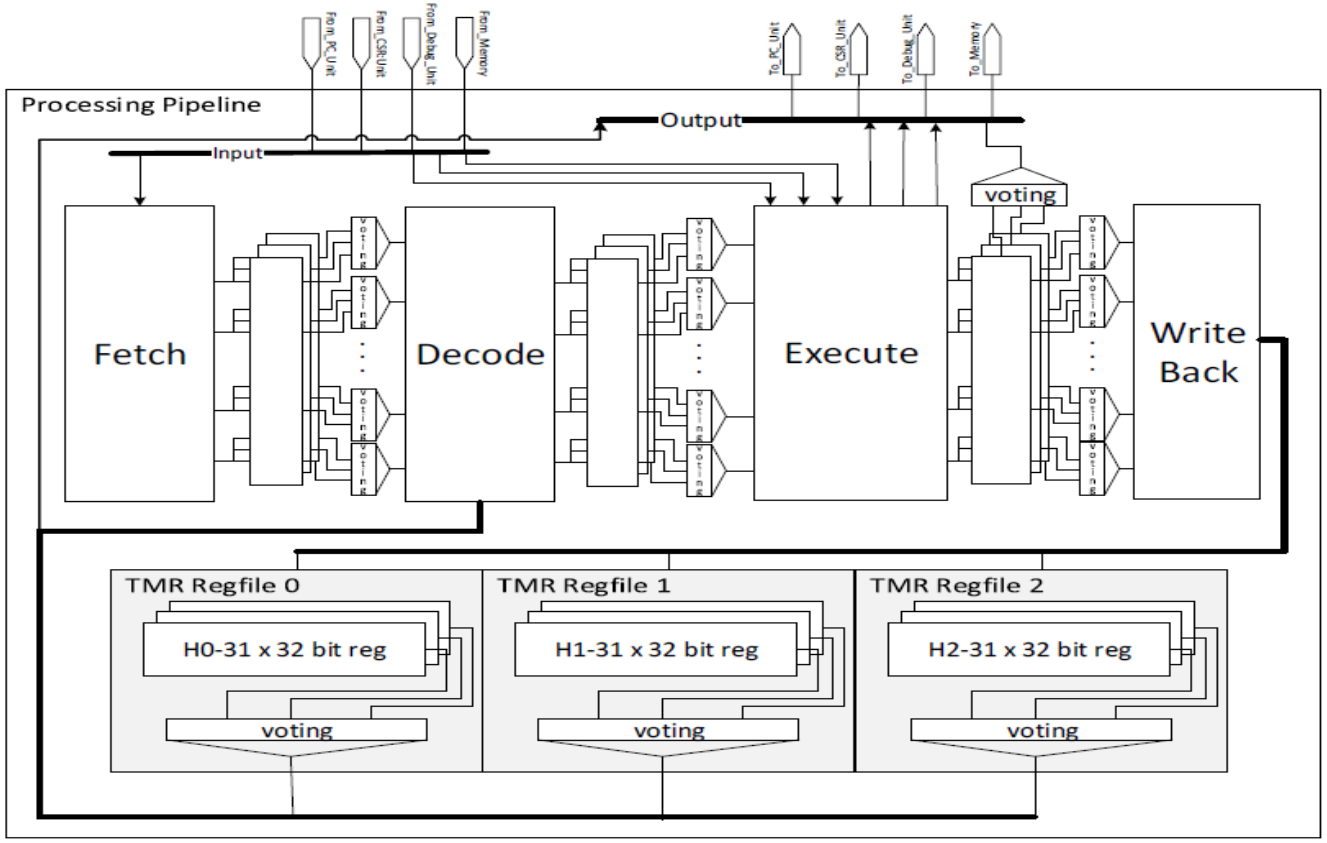


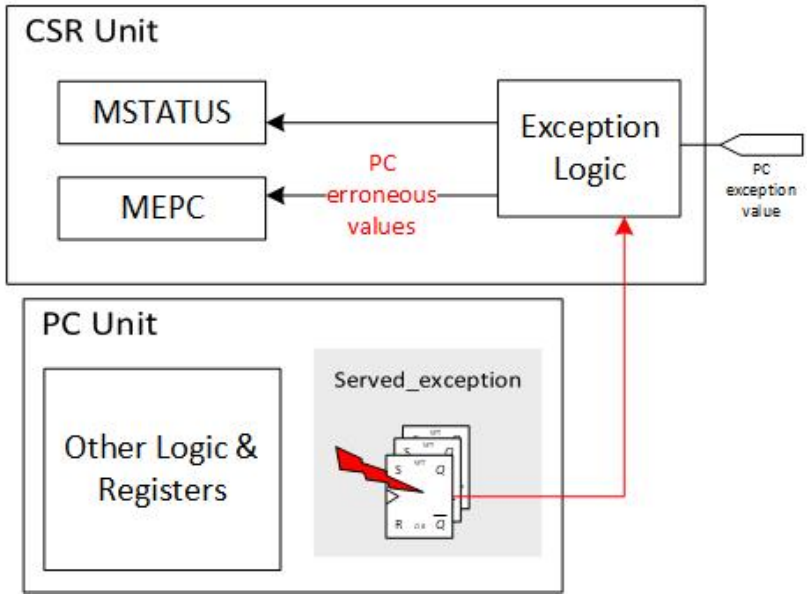
Klessydra F03a: Key Features

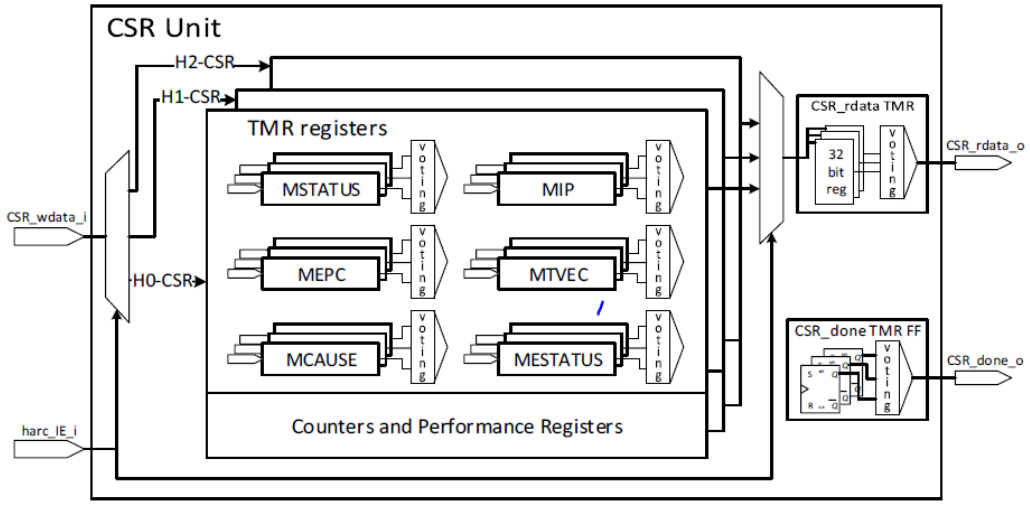
- **Local-TMR protection for the Control and Status Register unit (CSR)**
  - The CSR unit protection using local-TMR except for the Threads Counters and Performance registers which are protected using a SW redundancy
- **Local-TMR protection for the Processing Pipeline and Register Files**
  - All inter-stage registers (ISR) and Finite-State Machine registers (FSM) are protected
    - High overhead of hardware resources
    - Timing reduction due to combinational logic insertion
- **local-TMR protection for the PC unit**



Klessydra F03a: Microarchitecture



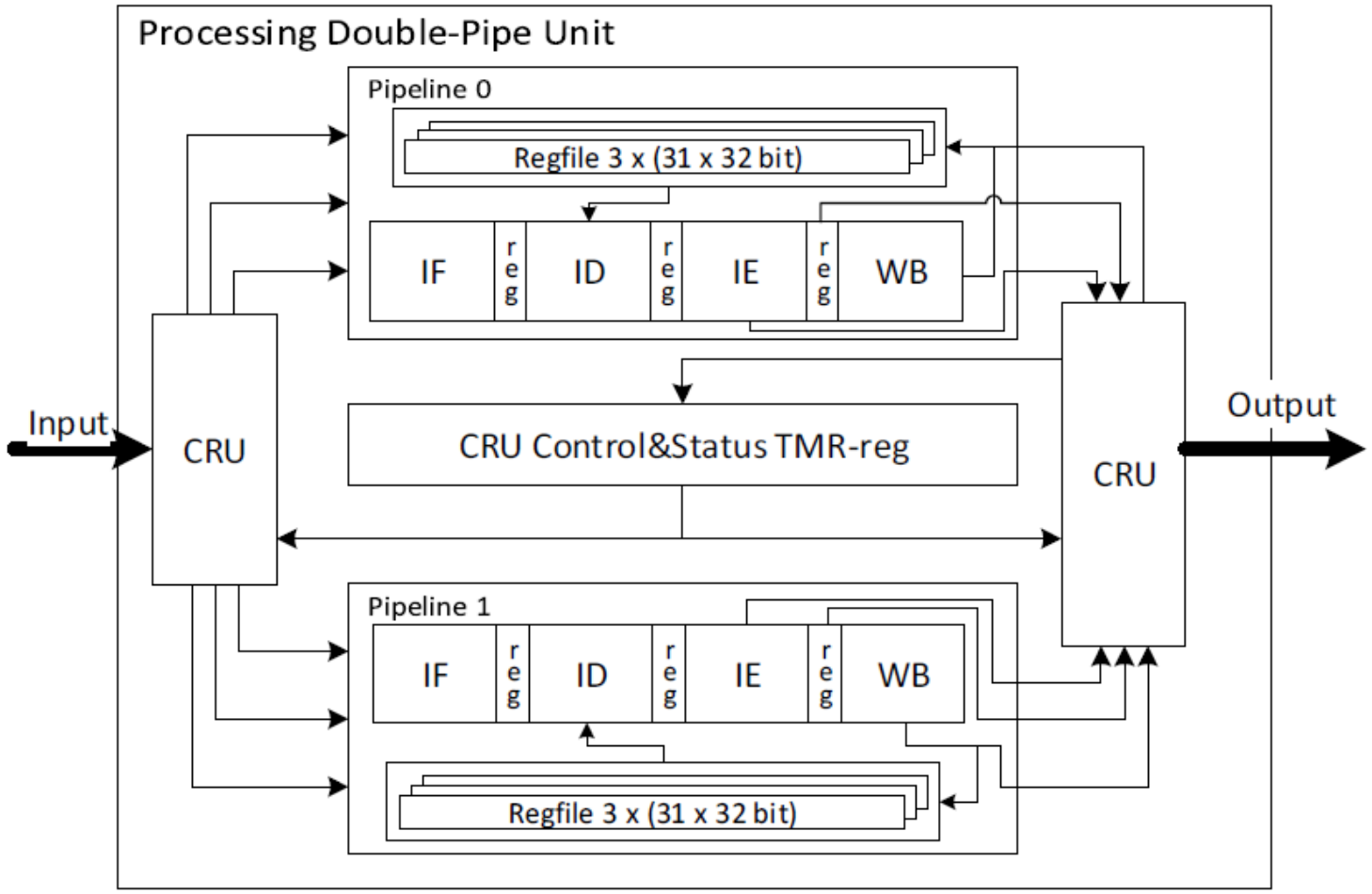




Klessydra F03b: Key Features

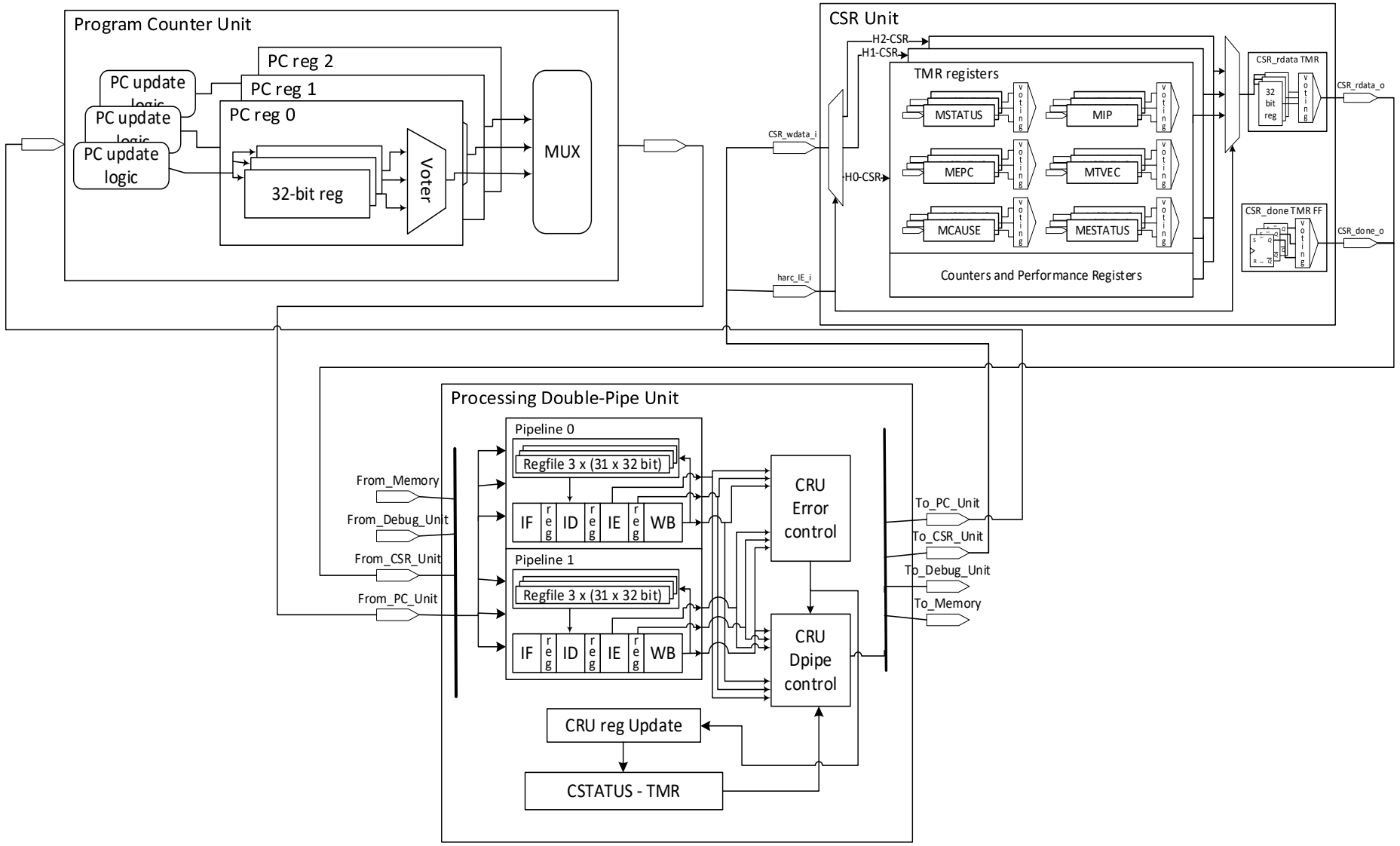
- **SW Checkpoint & Restore architecture**
  - saving register file data + PC within a dedicated Data Memory area before SW critical sections using a pseudo-code instruction
- **Local-TMR protection for CSR and PC (likely F03a)**
  - No TMR protection for Register Files and Processing Pipeline
- **Dual Pipeline Lockstep data elaboration (Double Processing Pipeline)**
  - Double Pipeline masked to the by the user software (only one is visible)
- **Check-Point and Restore Unit (CRU) unit for error detection and error handling and**
  - Error detection by checking the results of both pipelines
  - NO ERROR correction (only safe state recovery from a SW defined check-point)
  - Check-point data protection is provided both by a SEC-DED EDAC Hamming + Scrubbing architecture (for Register File data) and local-TMR (for CSR and PC data)
  - Error flags (R/W) and restore flags (R) are available from the user SW

Klessydra F03b: dual pipeline microarchitecture and CRU



- **Checkpoint enabled** by a dedicated SW pseudo instruction
- **Pipeline result check** is enabled (user hidden)
- **Checkpoint & Restore** in case of error detection from the previous stored checkpoint data
- **HW support** for the Restore procedure:
  - **Illegal Instruction** forcing (pipeline idle) and hardware exception handling to recover a safe state (checkpoint)

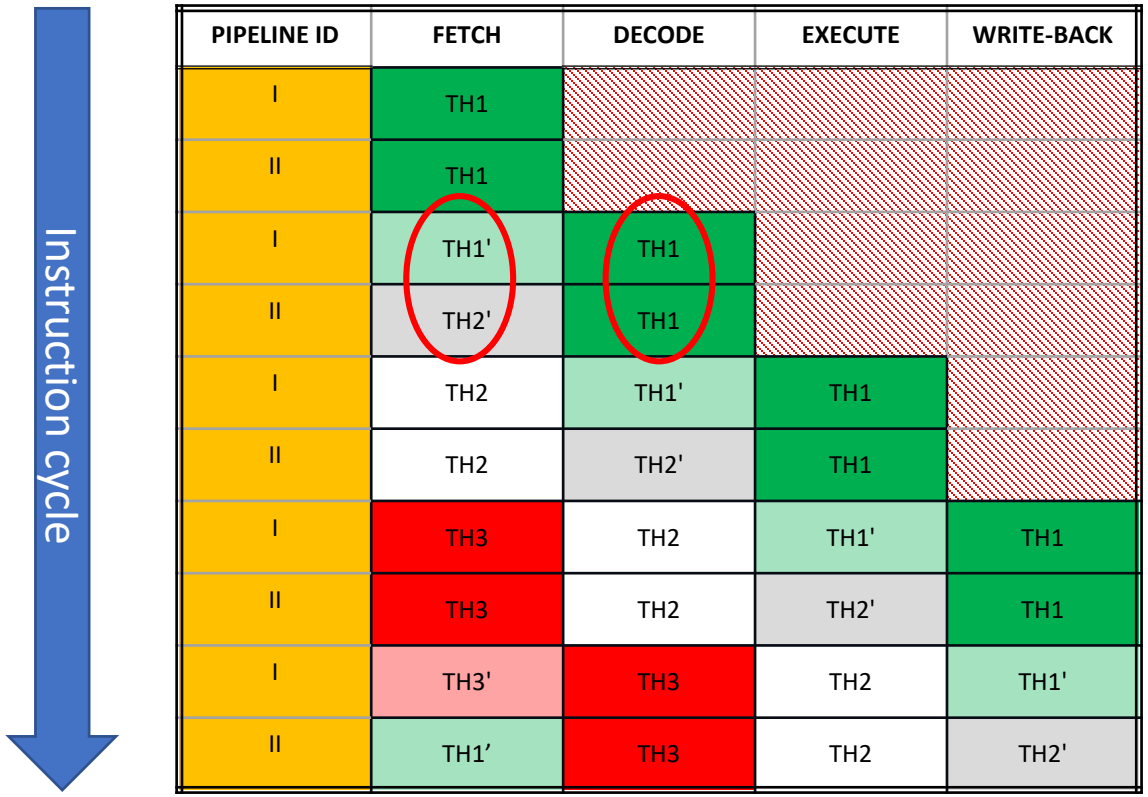
Klessydra F13b: complete microarchitecture



Klessydra F03c : Key Features

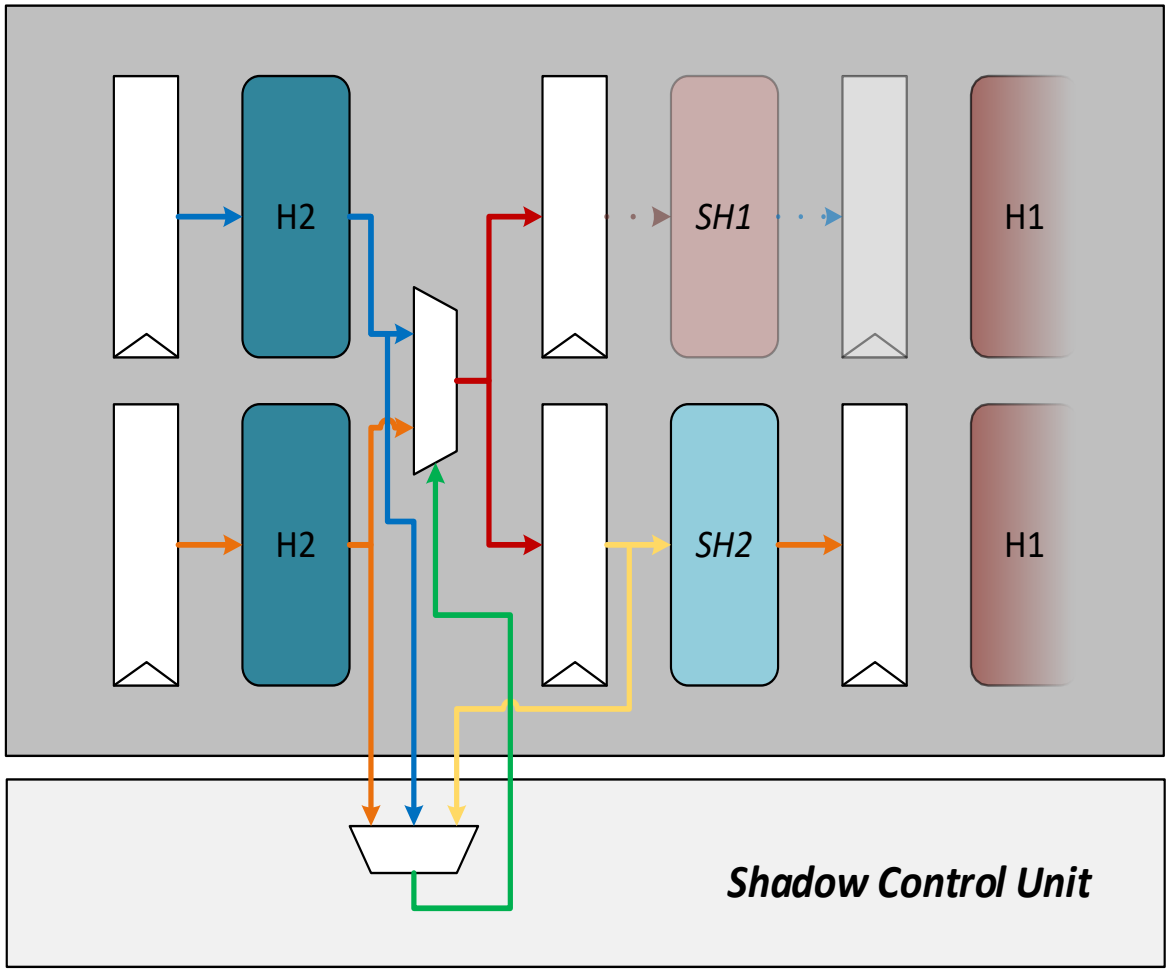
- **Hybrid Time-Space redundancy**
  - 2 pipelines + 3x instruction processing
  - duplicated registers files + copy inside the DATA memory.
- **Local-TMR protection for CSR and PC (likely F03a)**
  - No TMR protection for Register Files and Processing Pipeline
- **Dual Pipeline with shadow data elaboration (Shadow Double Processing Pipeline)**
  - Double Pipeline and Shadow Thread masked to the user software (only one is visible)
- **Shadow Control Unit**
  - It implements the voting system, manages the voted output of the processing pipeline and the voted input of the pipeline registers
- **Shadow Registers Unit**
  - Handles the register file errors check and restore during the instruction decode phase. The correction of a given register requires up to 2 clock cycles.

Klessydra F03c : Shadow Processing



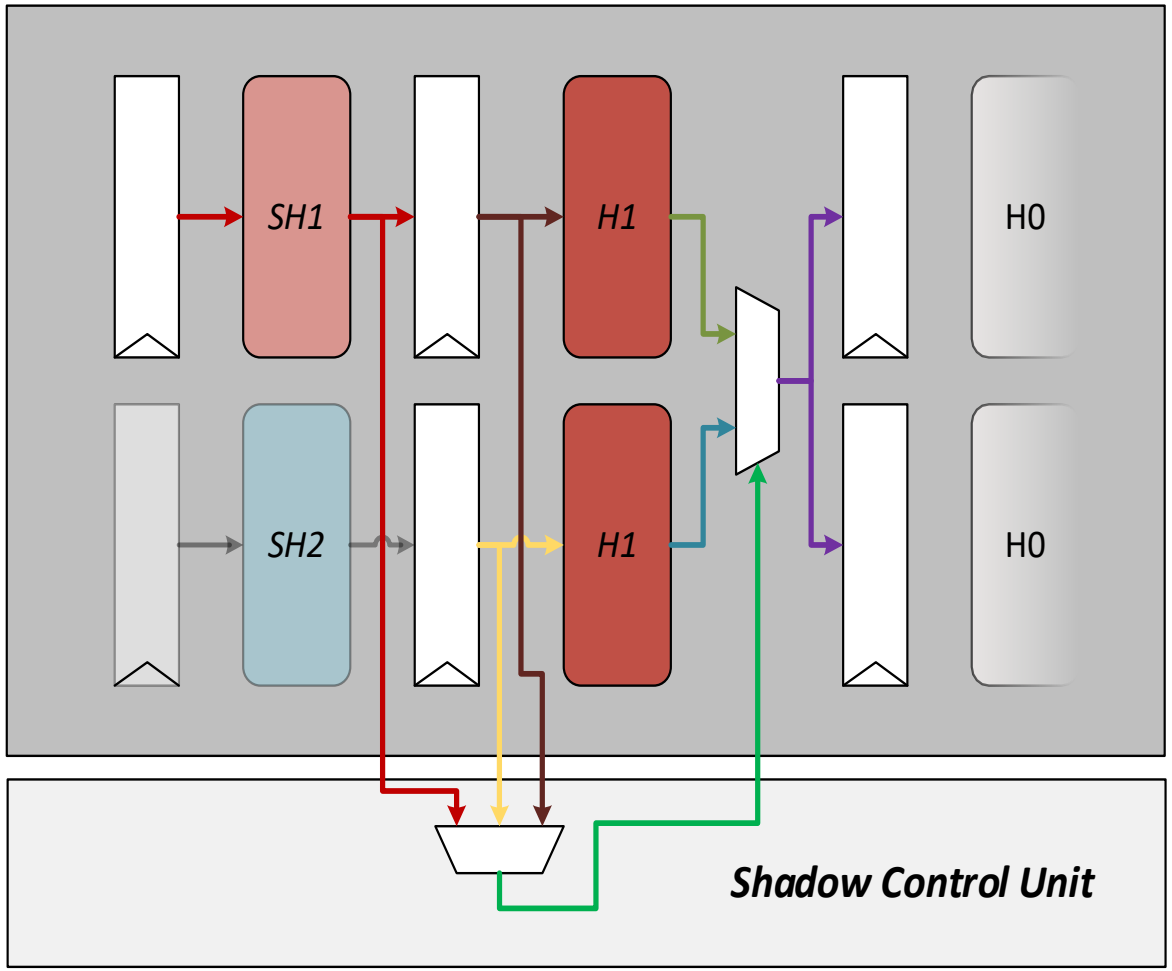
- **Main Processing**
  - The same thread instruction is paralleled issued on both pipelines. During this elaboration, the SCU handles the control of the result within every pipeline stage.
- **Shadow Processing**
  - the pipelines execute different thread instructions (shadow threads). These instructions are respectively copies of the thread instruction of the previous/following main elaboration.

Klessydra F03c : Pre-elaborated



- The **shadow control unit** compares the output value on the downstream register preloaded by the shadow phase with the values that the **main phase** wants to write to the register.
- If the values differ, the **SCU choose the correct value** for the input of the next pipeline register through a majority voting.

Klessydra F03c : Post-elaborated



- The **shadow control unit** compares the output values on the downstream register elaborated by the **main phase** with the values that the shadow phase wants to write to the register.
- If the values differ, the **SCU choose the value as the input** of the pipeline registers trough majority vote

Klessydra F03d: Key Features

- Derived from F03a
- Selective local-TMR protection ("*Full-Weak*") for Register Files
  - HART0 register file is protected with local-TMR **(FULL)**
  - HART1 and HART2 registers files are not protected **(WEAK)**
- Local-TMR for all the CSRs, PCs and Processing Pipelines
- Specific SW support required for HARTs error detection
  - HART1,2 perform not critical tasks
  - HART0 perform critical tasks and periodically checks the results of HART1,HART2
- Dedicated WDT driver SW support (Thread-Controlled WDT)
  - It can detect the WDT_RESET request from HART1,2
  - It can be reset only by the HART0
- Weak Threads fault detection by HART0 and WDT cooperation
  - If HART0 detects a mismatch between WDT_RESET request from HART1, HART2 it can:
    - Send a SW interrupt to the erroneous HART (which restarts from the address 0)
    - Allow an HARD reset to all the PULPino platform by the WDT

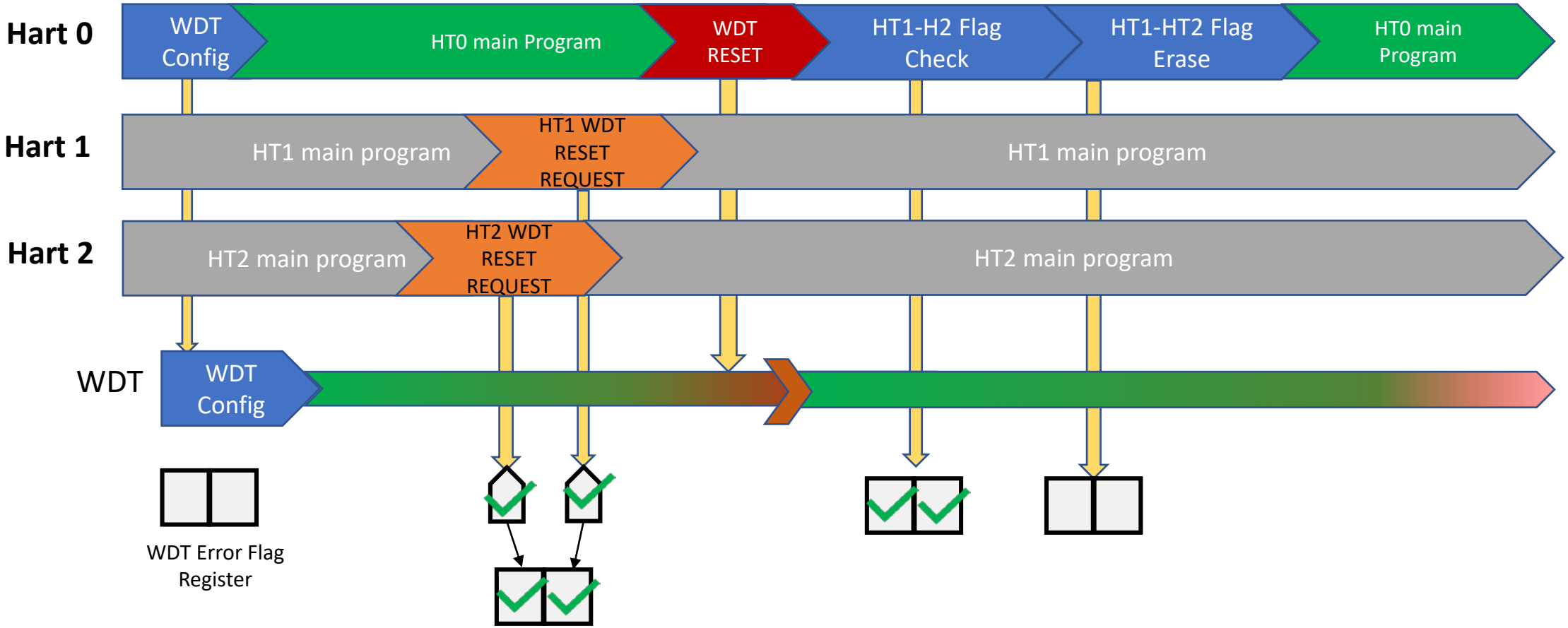
Thread-Controlled WDT: Key Features

- Designed to operate in multithread environments
- Provide support for F03d (compatible with all versions!!!)
- Perform an hard reset of the entire PULPino platform
  - NO single HART reset is possible
- Accessible only by the HART0 for set/reset operations using the APB interface
  - HART0 handles the WDT reset depending on the HART1,HART2 critical issue

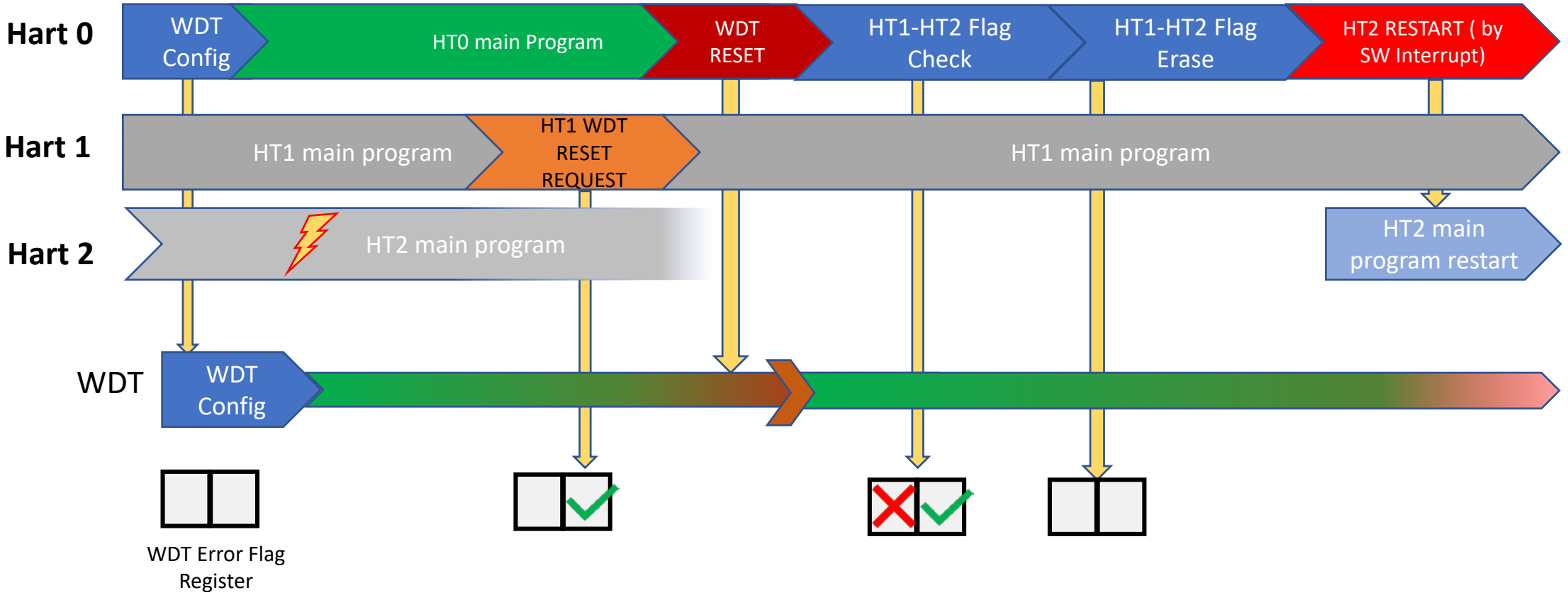
Klessydra F03d: HART0 and WDT

- WDT reset is handled by the HART0 (completely protected with local-TMR)
  - HART1 and HART2 set a WDT reset request
  - WDT enables the reset flags for HART1 and HART2
  - HART0 checks periodically the flags in order to detect mismatches
  - HART0 enable the reset request (if both weak HARTs have the same flag enabled)
- Fault-detection capability and fault-correction strategy is defined at SW level
  - A task critical issues analysis is mandatory in order to determine the best error handling strategy
- In case of mismatch between HART1 and HART2 WDT reset requests the HART0 can handle this error in several ways:
  - **Provide SW INTERRUPT reset to HART1 and/or HART2**
  - **Ignore the fault for HART1 and/or HART2**
  - **Let the WDT expire and reset the entire system**

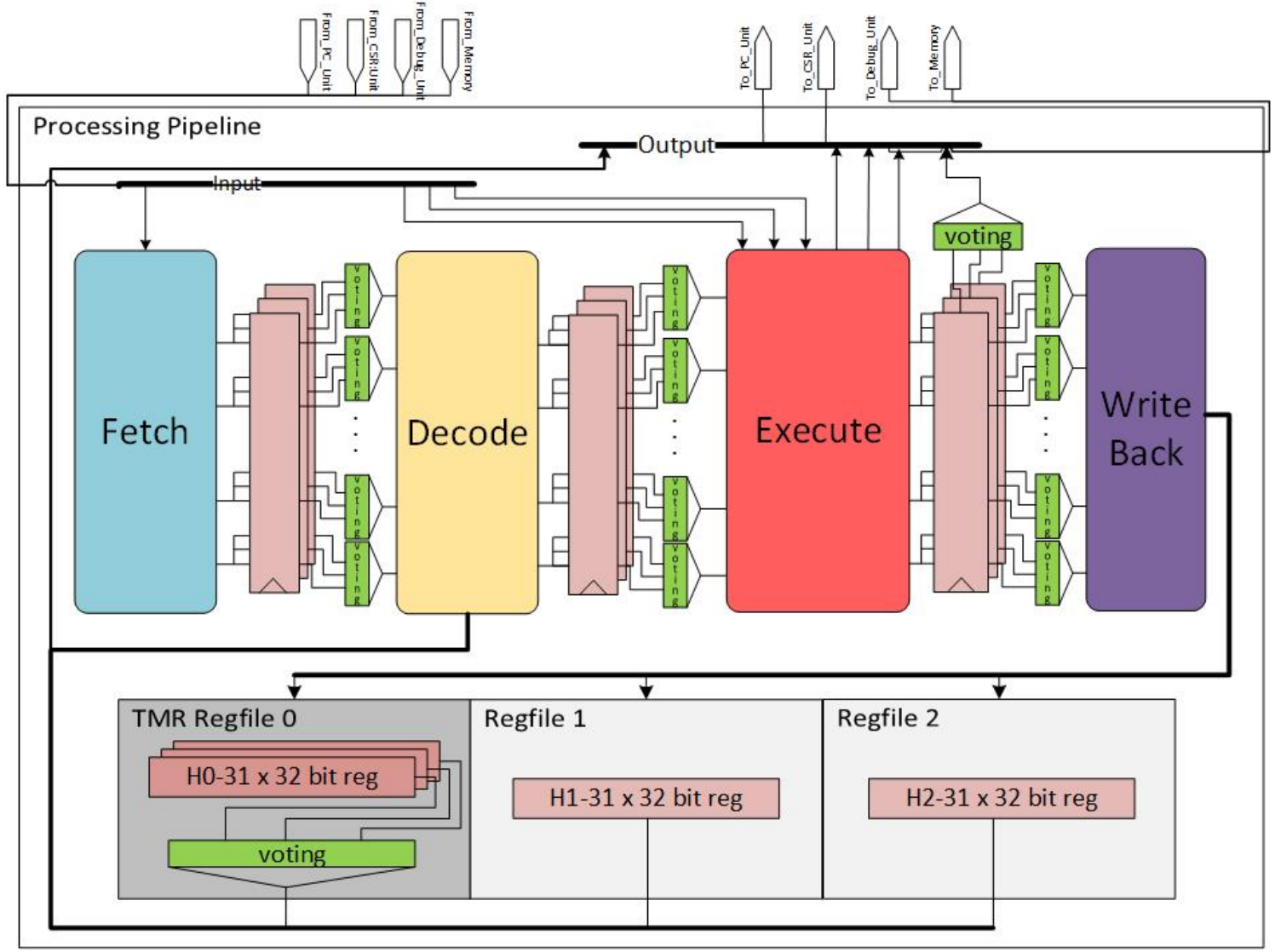
Klessydra F03d: No WDT REQ error detected



Klessydra F03d: WDT REQ error detected

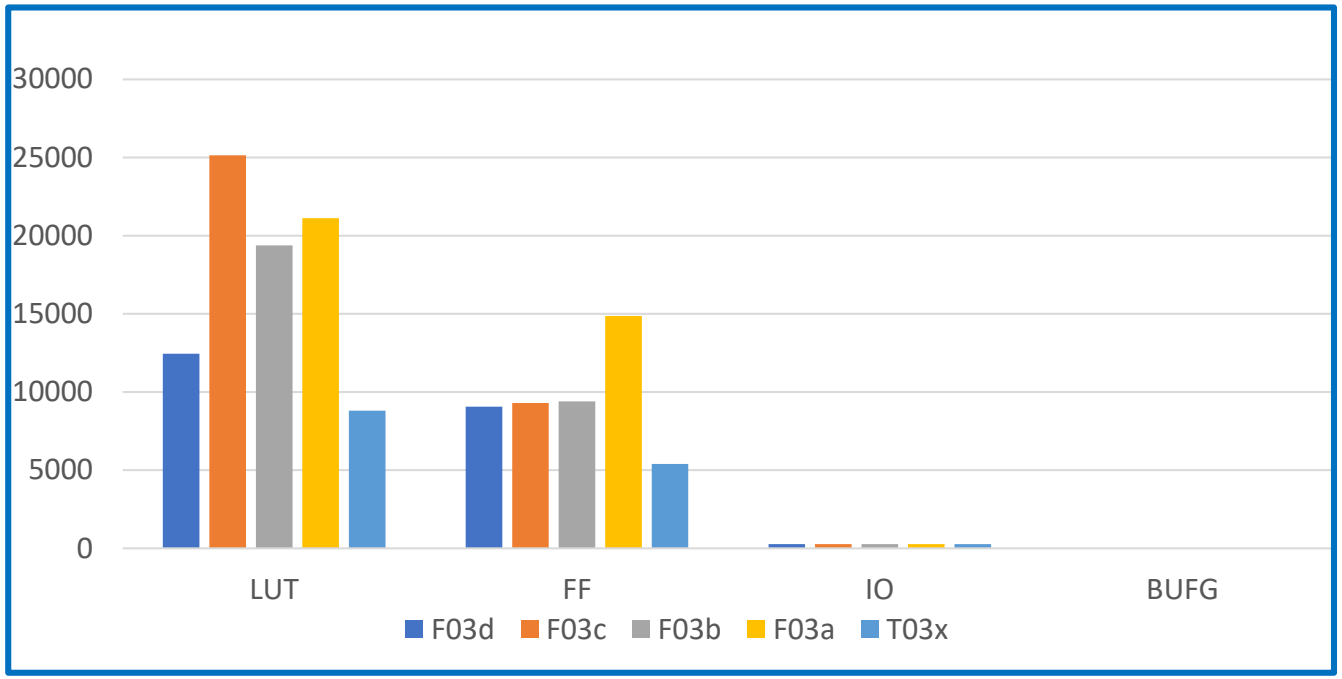


Klessydra F03d: microarchitecture



- AREA USAGE is less compared to the F03a
- Fault detection capabilities relies on a synergy between HART0 and WDT
- Fault correction capabilities relies on a dedicated SW implemented strategy
- A task critical issues analysis is required in order to execute the most critical task always by the HART0.

Klessydra F03x Family: Implementation Results



- **Results Analysis**
  - **F03a** requires an high amount of FFs
  - **F03c** requires an high amount of LUTs
  - **F03d** provides the best area effort between FFs usage (less than F03a) and LUTs usage (less than F03c)
  - **F03b** provides similar results to F03a without error correction capability

	F03d	F03c	F03b	F03a	T03x
LUT	12445	25138	19377	21117	8804
FF	9058	9299	9401	14863	5401
IO	269	269	269	269	269
BUFG	12	12	12	12	10

Klessydra F03x Family: Performance tests

RISC-V official test routines and Klessydra-specific test routines have been executed to compare the operation correctness and the performance between **F03x fault-tolerant cores** and **T03x standard** core.

TEST NAME	F03d	F03c	F03b	F03a	T03X
testALU	123131	146 679	123 413	123 131	123 135
testCSR	63098	77 396	63 380	63 098	63 098
testIRQ	316383	337 613	316 792	316 383	316 383
testException	43949	51 425	50 418	43 949	43 949
sw_irq_test	Not available	3 508	2 436	2 158	2 158
WFI_test	Not available	3 534	2 397	2 119	2 119
barrier_test	Not available	4 032	2 691	2 415	2 415

Clock cycles to complete each test routine

Klessydra F03x Family: HDL Fault-injection tests

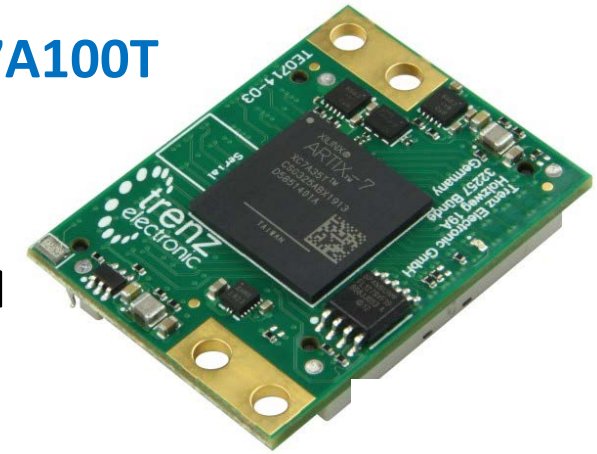
- **HDL Fault-injection campaign**
  - Simulate error injection over all the architecture internal registers (random)
  - Random single event upset for different upset rates
  - Worst-case (faster) fault injection rate for all architectures: **1 Upset/bit per 1µs**

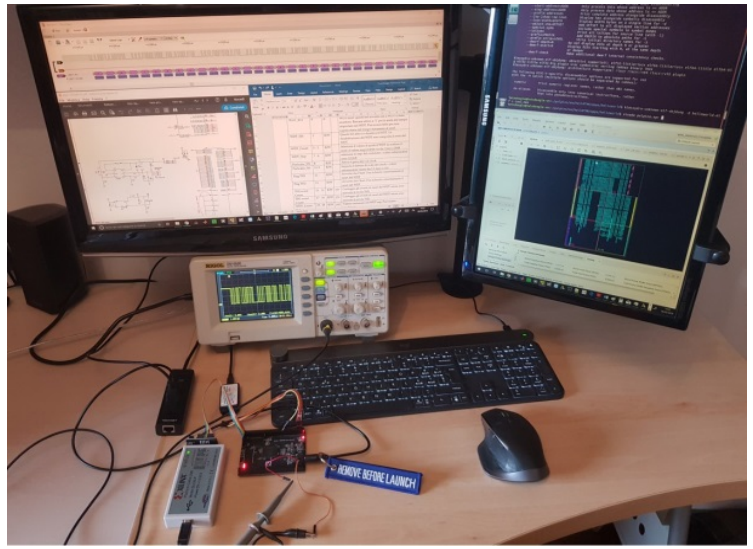
TEST NAME	F03d	F03c	F03b	F03a	T03X
FI_testALU_FX3X	63098	146 679	min 123 441 max 187 609	123 131	FAIL
FI_testCSR_FX3X	123131	77 398	min 63 618 max 121 748	63 098	FAIL
FI_testIRQ_FX3X	316383	337 641	min 380 035 max 756 406	316 383	FAIL

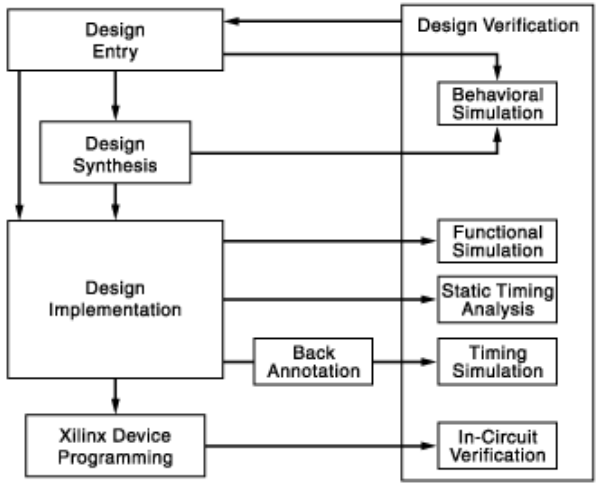
Clock cycles to complete each test routine

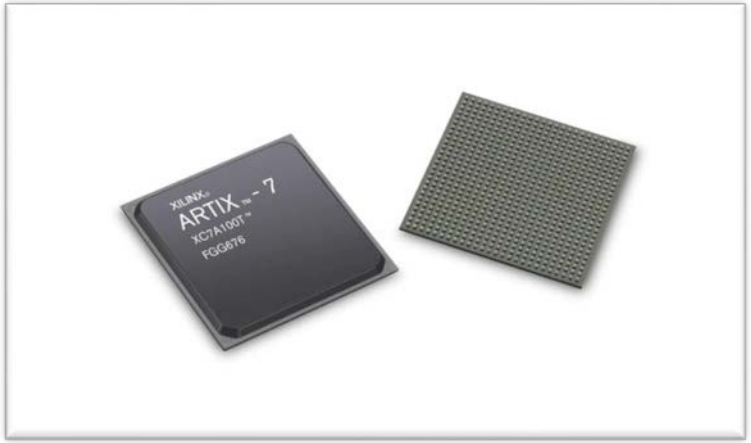
Klessydra F03x Family: Target FPGA for implementation

- **SRAM Target FPGA: Xilinx® Artix-7 XC7A100T**
  - 101440 Logic Cells
  - 15850 CLB Slices
  - 1188kb Maximum Distributed RAM
  - 4860Kb Total Block RAM
  - BER_{BRAM}= 3.73E-09 /bit /day
  - BER_{FF}= 3.78E-09 /bit /day
- **Tool for synthesis and implementation**
  - **VIVADO 2018.3**
    - (+ custom synthesis options)
  - Modelsim 10.6









The Klessydra Orbital Lab (KOL): Motivation

- **Nanosatellites** (e.g. CubeSat, PicoSat, PocketCube, etc.) allow academic institutions and small companies to afford space mission research.
- The little production volume demands the usage of **COTS components** in order to reduce the cost
- A **fault-tolerant HW architecture** is required in order to deal with the severe operating conditions of the space environment
- An **Open-source microarchitecture** design along with the exploitation of remotely **configurable devices** allow development support and design flexibility

What about sending a RISC-V microcontroller in space?

The Klessydra Orbital Lab (KOL): Design Concept

A Commercial Off-The-Shelf (COTS) SRAM FPGA-based In-orbit demonstrator (IOD) platform as a non-mission-critical on-board computer

- **Key Features**
  - HW reconfiguration capability using Over-the-air (OTA) bitstream uploads
  - HW reconfigurable FPGA-based architecture (soft IP-cores)
  - Open-source and flexible Instruction Set Architecture (ISA) support (RISC-V)
  - I/O interfaces with on-board sensors in order to collect data from the outside, process and store.
  - Test platform to evaluate several implemented Radiation Hardness By Design (RHBD) IP cores.
- **Project cooperation** with the School of Aerospace Engineering at Sapienza University, group led by Prof. Nascetti, for the satellite design

KOL: In Orbit Demonstrator Features and Mission description

IOD FEATURES

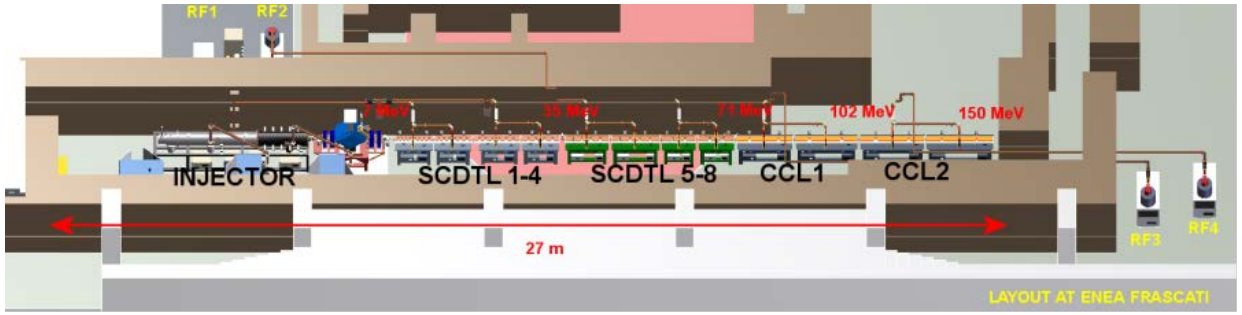
- **Payload** = PocketQube (50 x 50 x 300 mm)
- **Board** = 4 x 3 cm Xilinx TE0714 COTS FPGA development board (ARTIX-7 A50 FPGA)
- **External peripherals** = UART, I2C, SPI
- **Radio interface**
- **On-board sensors suite**
  - 3-axis accelerometer
  - Inertial Measurement Unit
  - Gyroscope
  - Magnetometer
  - Radiation sensors
  - Light sensors

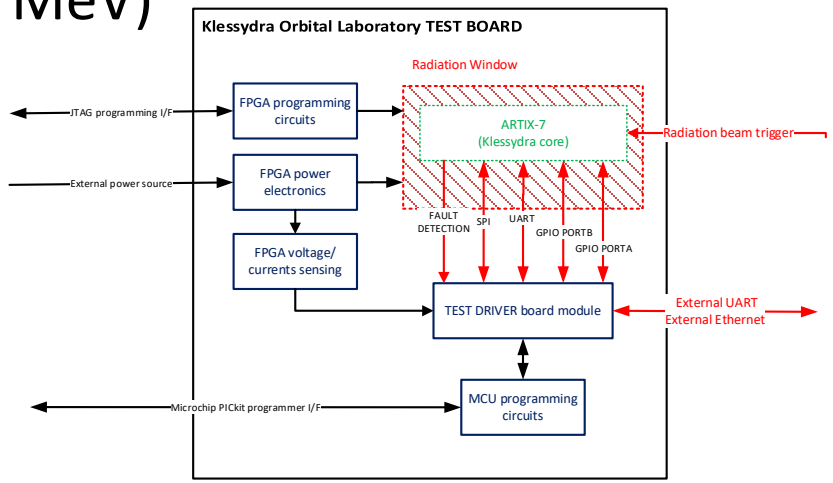
MISSION DESCRIPTION

- **Vector** = Soyuz
- **Primary Payload** = Unisat-7 (GAUSS srl)
- **Orbital Parameters**
  - Sun-synchronous LEO
  - 500-700 km
  - 97° inclination
  - Mission Duration (ext.) = 3-5 months
- **Radiation Environment**
  - TID ≈ 10KRad (1mm shielding)

Pre-mission radiation ground testing

- Work in progress
- Ionizing Particles beams irradiation test at TOP IMPLART LINAC
- Collaboration between Sapienza University and CNR-ENEA (Frascati)
- Testing the **KOL** soft IP core implementation on a Xilinx Artix-7 FPGA to evaluate the fault-detection capability for all the cores of the Klessydra F03x family
- Types of ionizing particles beams:
  - Low energy protons particle beams (from 21 MeV to 35 MeV)





Conclusions and Future work

- Present results on Klessydra F03x are based on fault injection simulation
- FPGA working prototypes of F03a, F03b, F03c, F03d available
- F03e design – pure time redundancy with hardware support – in progress
- Ground radiation test bed has been set-up
- Klessydra Orbital Lab satellite design and construction in progress
- Launch expected in spring 2020
- Future extension will be the inclusion of resilient re-configurable hardware acceleration unit for in-orbit data processing

Thanks for your attention

Questions?