

# Reliability in High-Performance Computing: Insights from a RISC-V Vector Processor



Marcello Barbirota, Francesco Minervini, Carlos Rojas Morales, Adrian Cristal, Osman Unsal and Mauro Olivieri

Sapienza University of Rome, Italy - Barcelona Supercomputing Center, Spain  
[marcello.barbirota@uniroma1.it](mailto:marcello.barbirota@uniroma1.it), [mauro.olivieri@uniroma1.it](mailto:mauro.olivieri@uniroma1.it)

## Challenge

High-Performance Computing (HPC) systems are designed for large-scale processing and complex dataset analysis, often integrating specialized hardware structures such as Vector Processing Units (VPUs). As these systems have grown in complexity and scale, their vulnerability to errors and failures has become an important and complex issue in the HPC world.

Our research addresses this challenge by exploring and implementing advanced fault tolerance techniques inside the Vitruvius+ architecture, a partial out-of-order Vector Processing Unit developed by a consortium of European companies and universities within the European Processor Initiative (EPI) research program [1].

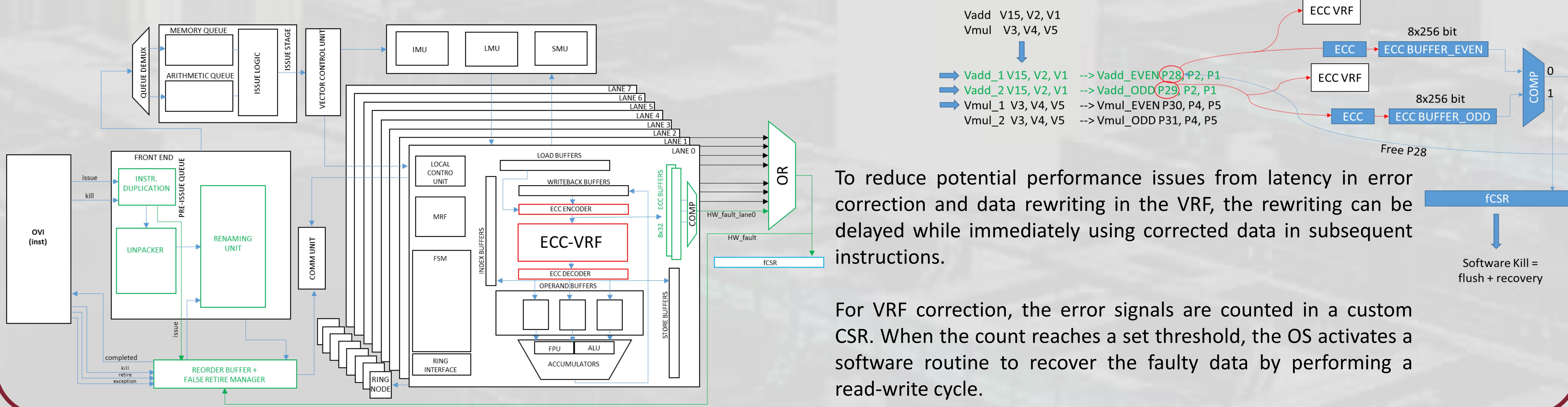
## Starting Point

Compliant with RVV version 1.0, the Vitruvius+ Architecture [2] is a decoupled vector accelerator with lightweight out-of-order execution capabilities boosted by vector register renaming and concurrent execution of memory and arithmetic instructions. Vitruvius+ mainly targets HPC applications characterized by long vectors; it supports up to 16384 bits in a vector register, meaning that it is capable of managing any vector length of up to 256 Double-Precision (DP) 64-bit floating-point elements.

## Methodology

The Fault Tolerant Vitruvius+ Architecture [3], depicted on the left, maintains the same vector characteristics of the original version, plus some modifications (green and red blocks). The approach is based on temporal redundancy (green blocks) in conjunction with ECC (red blocks). Generally speaking, the main advantage of temporal redundancy (instruction replication) techniques is related to the almost null hardware overhead at the expense of time overhead.

- Every replicated instruction (called EVEN and ODD) writes inside the ECC buffer the code word produced by the ECC logic (Figure right).
- If the ECC comparison reports a mismatch, the instruction is not committed, and a kill / roll-back hardware procedure is activated to recover the status of the VPU before the execution of the faulty instruction.
- The ECC calculation (red blocks in Figure left) gives the advantage of protecting the VRF and comparing the calculated ECC values of each replicated instruction result before writing it in the VRF rather than comparing the whole replicated result vectors.



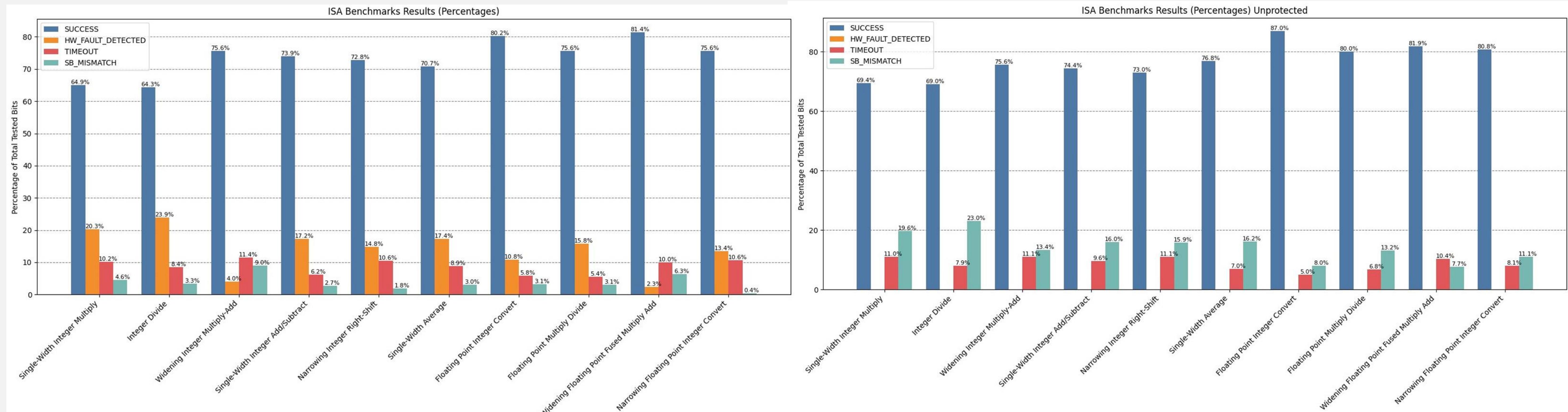
To reduce potential performance issues from latency in error correction and data rewriting in the VRF, the rewriting can be delayed while immediately using corrected data in subsequent instructions.

For VRF correction, the error signals are counted in a custom CSR. When the count reaches a set threshold, the OS activates a software routine to recover the faulty data by performing a read-write cycle.

## Fault Injection Results

The fault injection test campaign was done using a UVM environment in conjunction with the Spike simulator. It was chosen to analyze only the 0-index bit of all the signals belonging to Lane 0, for a total of approximately 3400 faulty bits per simulation run, with an average of 1 fault every 300 clock cycles, to avoid multiple faults hitting two copies (EVEN and ODD).

The introduced features reduce by 75% the occurrence of non-silent undetected faults that would result in application failure.



## Hardware Occupation Results

The entire VPU was synthesized for the GF22FDX technology using Cadence Genus Synthesis Solution 19.11, successfully satisfying the timing constraints, achieving an estimated maximum frequency of 1.47 GHz with the critical path set in the Register File due to the newly introduced ECC mechanisms, compared to 1.51 GHz reached from the original Vitruvius+ design.

	VPU Area	VPU Frequency (typ)	VPU Frequency (slow)
No ECC – No HW-ID	1.49 $\mu\text{m}^2$	1.51 GHz	1.08 GHz
ECC – HW-ID	1.61 $\mu\text{m}^2$	1.47 GHz	1.04 GHz

## References

- [1] M. Kovačić, "European processor initiative: the industrial cornerstone of eurohpc for exascale era," in Proceedings of the 16th ACM International Conference on Computing Frontiers, ser. CF '19. New York, NY, USA: Association for Computing Machinery, 2019, p. 319. [Online]. Available: <https://doi.org/10.1145/3310273.3323432>
- [2] F. Minervini, O. Palomar, O. Unsal, E. Reggiani, J. Quiroga, J. Marimon, C. Rojas, R. Figueras, A. Ruiz, A. Gonzalez et al., "Vitruvius+: an area-efficient risc-v decoupled vector coprocessor for high performance computing applications," ACM Transactions on Architecture and Code Optimization, vol. 20, no. 2, pp. 1–25, 2023.
- [3] M. Barbirota, F. Minervini, C. R. Morales, A. Cristal, O. Unsal and M. Olivieri, "Enhancing Fault Tolerance in High-Performance Computing: A Real Hardware Case Study on a RISC-V Vector Processing Unit," in IEEE Open Journal of the Computer Society, vol. 5, pp. 553–565, 2024, doi: 10.1109/OJCS.2024.3468895.