Exploring the Security of an Accelerator integrated with Core-V eXtention InterFace (CV-X-IF)

Alessandra Dolmeta[1], Behnam Farnaghinejad[2], Davide Bellizia[3],
Guido Masera[1],Maurizio Martina[1], Stefano Di Carlo[2], Alessandro Savino[2], and Ernesto Sanchez[2]

1 Politecnico di Torino, DET - Dipartimento di Elettronica e Telecomunicazioni , Turin, Italy, 2 Politecnico di Torino, DAUIN - Dipartimento di Elettronica e Telecomunicazioni , Turin, Italy 3 Telsy S.p.A., Rome, Italy

Finanziato dall'Unione europea

Introduction

- The rise of connected embedded systems has intensified the need for secure microcontrollers, especially against side-channel attacks.
- Hardware accelerators improve performance but introduce new vulnerabilities.
- The RISC-V-based Core-V-Extension Interface (**CV-X-IF)** [1] enables clean accelerator integration without core modifications, enhancing design flexibility. However, its side-channel resistance remains largely unexplored, raising critical questions for security-sensitive applications.

CV-X-IF

- **Tightly-Coupled Integration in RISC-V.** In a tightly-coupled approach, coprocessors are integrated directly with the processor pipeline for low-latency communication. Typically, this requires modifying elements such as the decoder and register file. However, the **CV-X-IF interface** introduces a **dispatcher** that enables the integration of custom instructions and coprocessors **without modifying the existing core components**, preserving the original decoder, ALU, and RF (*Figure 1*).

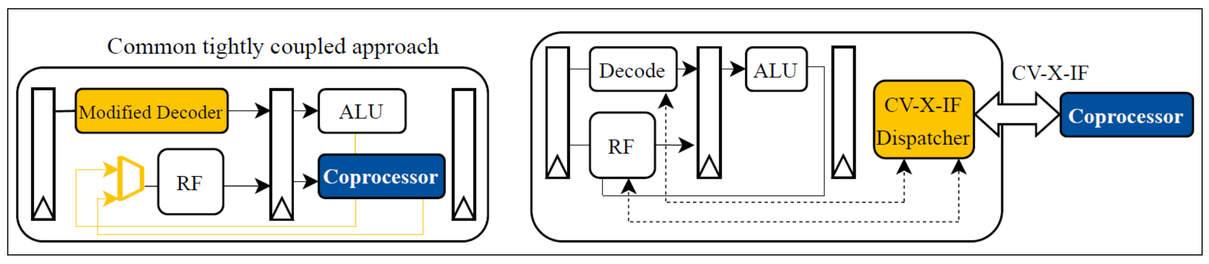

***Figure 1**. Common tightly approach vs. CV-X-IF.*

- CV-X-IF **adds custom instructions** exploiting unused opcodes to trigger the coprocessor. It ensures:
  - **low-latency register access**
  - **external extension support**
  - **synchronous execution**.

Architecture

- This work compares two implementations of the XOR operation on a RISC-V platform:
  - **Native RISC-V Instruction**: The standard xor instruction executed directly on the CVA6 core (Ariane RISC-V CPU [2]). This serves as the baseline for performance and power evaluation.
  - **Custom Accelerator**: A hardware-optimized XOR module implemented as a tightly-coupled coprocessor. The integration is performed via the CV-X-IF interface, allowing seamless extension of the CVA6 pipeline without modifying its core components.

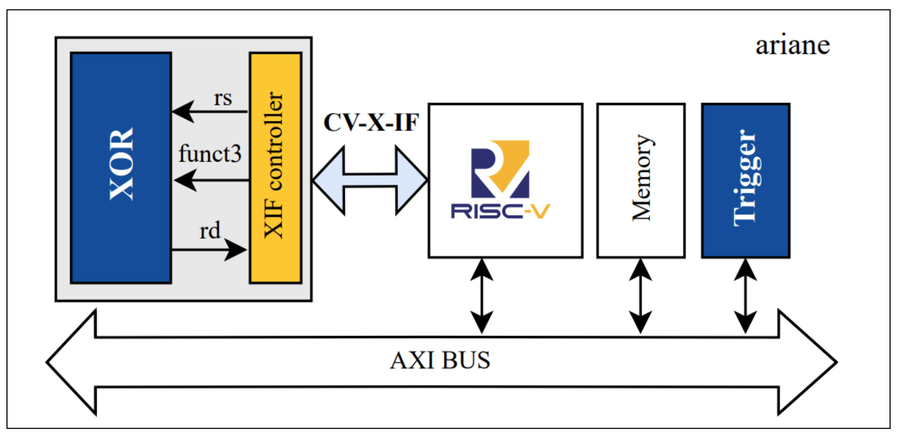

***Figure .2** Architecture of the tested RISC-V SoC.*

- **Target Hardware**: The CVA6 core, extended with the CV-X-IF XOR accelerator, is deployed on an AMD Xilinx Artix-7 FPGA and hosted on a CW305 board [3]. The system operates at 5 MHz with a core voltage of 1 V. A **trigger** GPIO is added, to provide a precise trigger for isolating the relevant execution window in power traces.

Methodologies

- To evaluate the side-channel resilience of the CV-X-IF-integrated accelerator, we employed a **Test Vector Leakage Assessment** (TVLA) framework. Instead of the standard fixed-vs-random approach, we analyzed power traces while executing the same XOR operation on identical input data across two implementations.
- This method isolates differences in leakage purely due to the execution path rather than input variations.
- **Measurement Tools**: a Picoscope 6404D oscilloscope for capturing power traces at 1.25 GS/s, synchronized via a trigger signal.

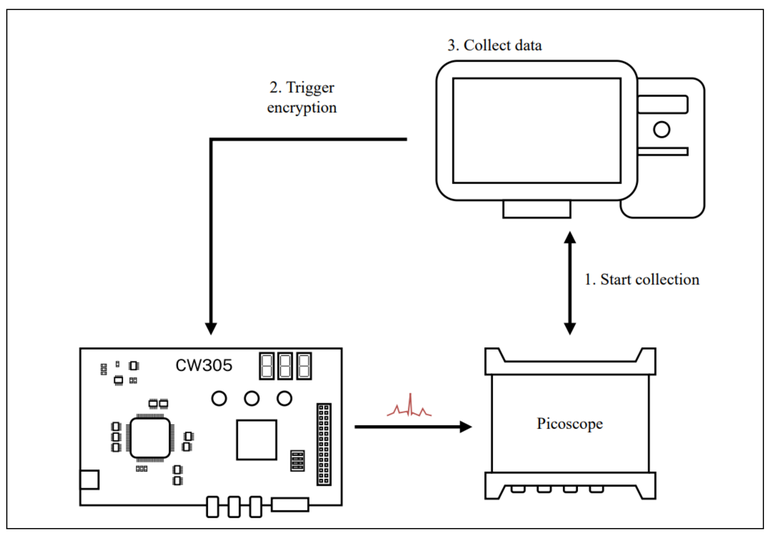

***Figure 3.** Setup environment.*

- **Control Interface**: a UART-based communication protocol between the FPGA and a host PC, enabling:
  - ***PRNG Initialization***: A cryptographically secure pseudorandom number generator (AES-CBC with a 128-bit seed) to randomize the selection between the two XOR implementations, preventing timing or execution-pattern biases
  - ***Trace Configuration***: Dynamic adjustment of the number of traces (10,000–50,000 per test) to ensure statistical significance.

Results

- We employed the TVLA methodology to compare the side-channel leakage of the XOR instruction when executed natively on the CVA6 core and via the CV-X-IF accelerator. The plots show:
  - **Raw power traces** for both implementations.
  - **Averaged traces**, highlighting their distinct temporal profiles.
  - The resulting **t-test scores**, which **exceed the ±4.5 threshold,** indicating a statistically significant difference in leakage between the two implementations.

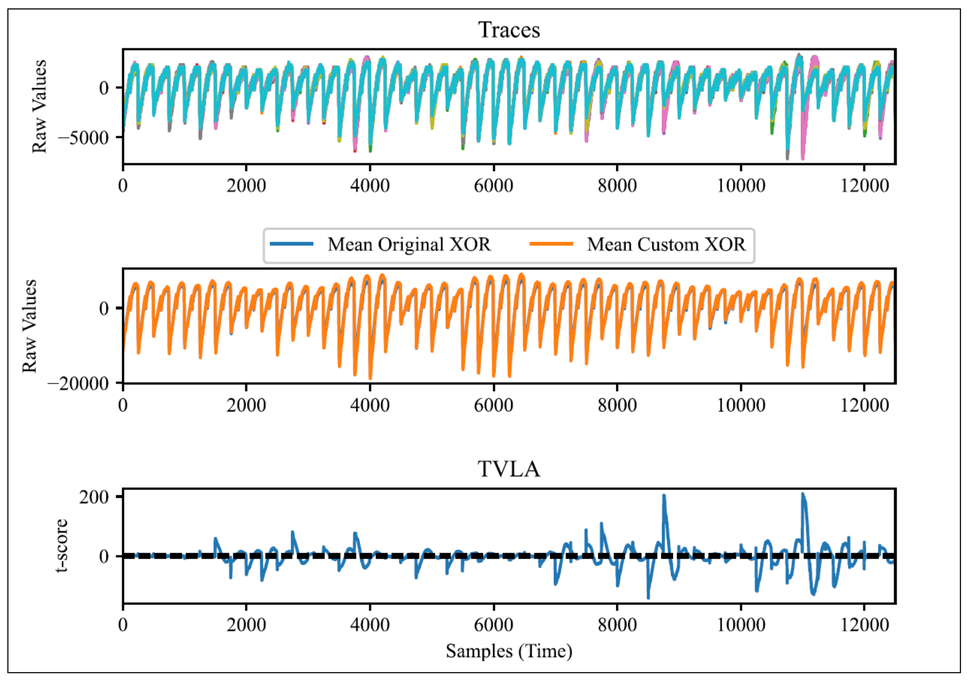

***Figure 4.** Test results.*

- These results demonstrate that the CV-X-IF-based accelerator exhibits distinct side-channel behavior, confirming that execution paths are distinguishable from the native pipeline.

Reference

[1] https://github.com/openhwgroup/core-v-xif/tree/main
[2] https://github.com/openhwgroup/cva6
[3] https://www.newae.com/products/nae-cw305