Secondment report

TIRAMISU DC

Fellow: Germano Berto Girondi
Research project title: Virtual Prototyping as Hardware-Software Co-Design Methodology for Next Generation Automotive Edge-AI
From: Dumarey Softronix
To: Politecnico di Torino
Secondment period: 3rd May 2026 – 28th August 2026

Activities during the secondment

Scope and objectives

As modern vehicles increasingly rely on edge-AI for a growing number of applications, more rigorous architectural design choices are required to comply with the functional safety requirements defined by ISO 26262. AI accelerators, now critical components within electronic control units (ECUs), fundamental building blocks of modern vehicles, are central to enabling this edge-AI deployment. The goal of this project is to avoid the high cost of direct physical prototyping of different architectures by virtualizing the processors and accelerators, allowing the impact of architectural choices to be evaluated beforehand.

Regarding processor selection, two architectures were chosen: Arm, currently found in almost all vehicles worldwide, and RISC-V, which offers the benefit of an open-source instruction set architecture (ISA) and an ever-increasing range of applications. Once the architectures were defined, QEMU was selected as the target emulation and virtualization platform, given its open-source nature and its capability to support the integration of new processor models.

Before testing any real world applications, it was necessary to verify that the processor emulations were behaving as expected. For this purpose, the open-source soft-core RISC-V processor NEORV32 was selected, as it could later be validated against real hardware by running the processor on PYNQ-ZU, an AMD Xilinx Zynq UltraScale+ MPSoC. Another key motivation behind selecting this specific processor is that its interaction with an open source TPU had already been tested in prior work, offering a validated test case for TPU communication patterns similar to those used in more complex ADAS (Advanced Driver Assistance Systems) applications.

Activities

Different CPUs and machines were explored on QEMU, including the 32-bit and 64-bit RISC V targets, running both baremetal applications, such as search algorithms, and Linux, in order to map the emulator's capabilities in terms of component visibility, access level, state inspection, and data manipulation. Progress was made on integrating the NEORV32 as a viable CPU and machine option within QEMU, essential respectively for testing the ISA and processor functionalities, and for validating peripheral behavior. For this stage, a FreeRTOS port specifically adapted for the NEORV32 was used as the first application tested, enabling a direct comparison between the QEMU and FPGA environments. The benchmarks selected include Dijkstra, a matrix multiplication, Dhrystone, and the Thread-Metric suite, a modern and improved successor to the Rhealstone and Dhrystone suites. The peripherals required to support the adapted FreeRTOS, including UART, GPIO, and MTIME, were

The TIRAMISU Doctoral Network has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101169378.

implemented on the emulator, and their behavior was validated against tests conducted on the corresponding peripherals of the real hardware.

Building on the validated NEORV32 QEMU-FPGA environment, work progressed toward enabling comparable fault injection campaigns on both platforms, with the aim of assessing whether QEMU can serve as a reliable, low-cost proxy for FPGA level fault sensitivity analysis. To ensure that fault injection results are meaningful across platforms, the two environments were first aligned at the ISA and memory levels: the FPGA bitstream was re-synthesized to include the Zba, Zbb, Zbs, Zmmul, and Zicond extensions, matching the configuration already available in QEMU, and the heap size was reduced to 16 KB so that all benchmarks fit within a unified 32 KB data memory on both platforms. All four benchmarks were then recompiled from a single unified toolchain configuration, producing identical ELF binaries executable on both the FPGA and QEMU without modification.

Concurrently with the fault injection framework described above, a separate side project was developed addressing radiation hardened reconfigurable logic design. Motivated by a gap identified in existing FPGA radiation hardening literature, as prior work targets configuration memory cells, cluster level orchestration, or application specific selective redundancy, but no reviewed work redesigns the LUT and flip-flop fabric itself as a cohesive, integrated radiation hardened logic fabric, a mapping toolchain was developed to support this kind of design exploration. The tool accepts a structural gate level or RTL Verilog description together with a transistor level implementation of the target LUT and flip-flop primitives, and produces, from a single internal representation, both a flattened transistor level SPICE netlist for circuit level simulation and a structural RTL netlist built from the same LUT primitives for downstream layout generation. LTspice based simulation of clocked, sequential circuits have been carried out, including a comparison between standard and hardened cell variants that surfaced a clock margin sensitivity specific to the hardened design under test.

Main results achieved

With both platforms verified to produce identical, checksum matched golden outputs, a fault injection framework was developed for QEMU, capable of performing single bit flip campaigns, either targeted at a specific address and bit, or swept randomly across the .text and .data segments of each benchmark binary, and classifying the outcome of each run using a four category taxonomy: masked (no observable effect), silent data corruption (the application completes but produces incorrect output), and timeout (the application fails to terminate within the allotted time). Each fault is injected into an independent copy of the original binary prior to execution, ensuring that fault campaigns remain single fault and non-cumulative.

An initial sweep campaign across the three standalone benchmarks produced the following outcome distribution when testing 10000 fault injections in each application:

Benchmark	Masked	SDC	Halt
Dhrystone	81.0%	5.8%	13.2%
Multiply	54.2%	33.6%	12.2%
Dijkstra	76.9%	8.5%	14.7%

Table I: Preliminary FI results in emulated environment.

Multiply shows a markedly higher SDC rate than the other two benchmarks, consistent with its arithmetic and data intensive nature: a larger share of its instruction and data footprint directly participates in numerical computation, so a bit-flip is more likely to silently alter a computed value than to be masked by unused code paths or overwritten before use. Dhrystone and Dijkstra, both more control flow oriented, show a correspondingly higher share of masked faults. It should be noted that

The TIRAMISU Doctoral Network has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101169378.

the results presented in this work reflect a preliminary QEMU campaign. By the ongoing work described below, the central validation step of this project phase is the comparison between these QEMU based outcomes and an equivalent campaign executed on real FPGA hardware.

Next steps

Work is ongoing to extend an equivalent fault injection campaign to the FPGA, where faults will be introduced directly into the synthesized bitstream or BRAM contents during benchmark execution, with results classified using the same taxonomy. The resulting comparison between QEMU based and FPGA based fault injection outcomes constitutes the central validation step of this project phase, determining the extent to which QEMU based fault injection can substitute for costly and time consuming physical prototyping in early stage dependability assessment.

The inclusion of the TPU can also be evaluated while targeting tensor specific applications. In the emulation environment, the TPU can be included as a memory mapped object in the memory region that is used to connect peripherals, then containing its components: a register file, a unified buffer, and a weight buffer, along with the implementation of the systolic array computation. The inclusion of memory mapped objects was already conducted to observe how QEMU can handle different devices. As QEMU provides a very similar structure to the AXI-Lite concerning the callbacks of its memory regions, communication between the processor and the accelerator could bypass this step. But, as the idea is to replicate the behaviour in hardware, and, as it is possible to find examples of devices containing AXI-Lite register files in QEMU, an explicit bus model layer could be added to replicate the real environment more accurately once the communication without it is working properly.

Self-evaluation

Although progress is being made, and the near and long-term seem clear (if not by technical questions that naturally appear, and which is part of the project to solve), there was an underestimation of the time required to produce meaningful publishable results, which are an unavoidable requirement from the project's institution. There was also a lack of initiative in evaluating how the results produced by other ongoing projects could contribute to this one and vice versa, even if technical support was provided when demanded, which impacted on the research involvement. Other requirements, such as the total number of hard and soft skill credits, shall be completed by the end of the year, which seemed acceptable.

Overall score

5 - Agree

Optional comments

f.to: Luca Sterpone

Politecnico di Torino

e report approval by the supervisor:

Data: 28/07/2026 16:33:38 CEST

The TIRAMISU Doctoral Network has received funding from the European Union's Horizon Europe research and innovation programme under the Marie Skłodowska-Curie grant agreement No 101169378.