Automating Fault Injection through CABA Simulation for Vulnerability Assessment

William Pensec, Vianney Lapôtre, Guy Gogniat

Automating Fault Injection through CABA Simulation for Vulnerability Assessment

William PENSEC, Vianney LAPÔTRE, Guy GOGNIAT
Lab-STICC, UMR 6285, Université Bretagne Sud, Lorient, France
firstname.lastname@univ-ubs.fr

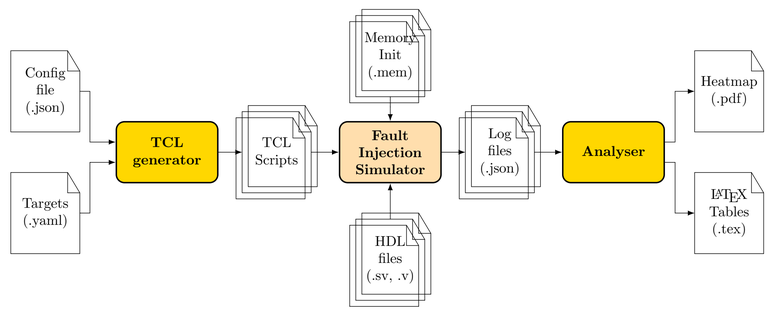
Context

Internet of Things (IoT) devices have revolutionised data collection and analysis, yet their proximity raises concerns about physical attacks like fault injection attacks (FIA). Numerous studies have highlighted critical system vulnerabilities to FIAs. This poster presents FISSA (Fault Injection Simulation for Security Assessment), automating circuit design robustness evaluation against FIA through early-stage simulations using existing HDL simulators.

FISSA - Fault Injection Simulation for Security Assessment

FISSA is an open-source [1] tool based on an HDL simulator such as Questasim/Modelsim for performing fault injection attack campaigns in simulation in order to assess the security of designs during the development phase.
Its main characteristics are:

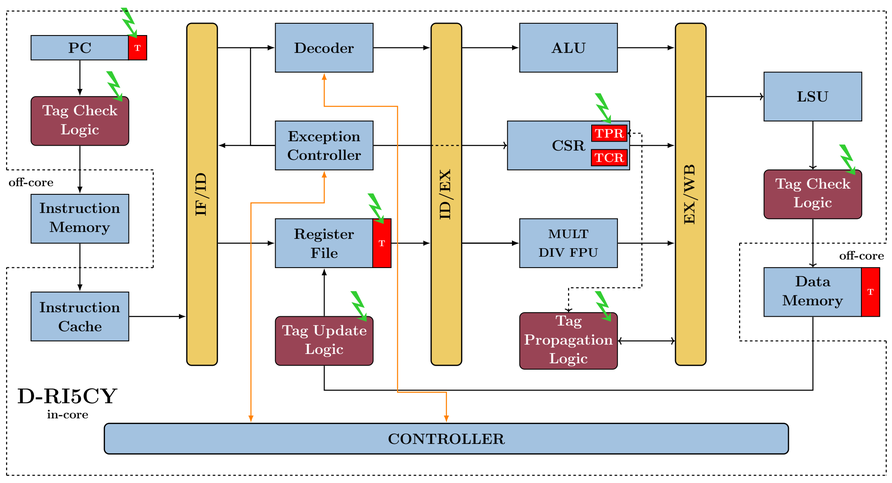
- Highly configurable
- Easy to integrate into a new design
- High control over fault scenarios
- Multiples fault scenarios already implemented



Software architecture of FISSA

D-RI5CY

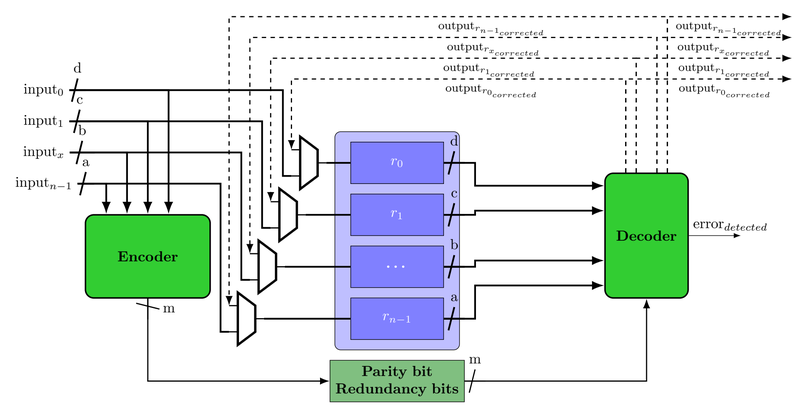
We study the D-RI5CY [2] which introduces a Dynamic Information Flow Tracking (DIFT) mechanism to protect the processor against software attacks such as buffer overflows, SQL injections, etc. DIFT-related elements are represented in red.



D-RI5CY processor architecture overview

Evaluated protection mechanism

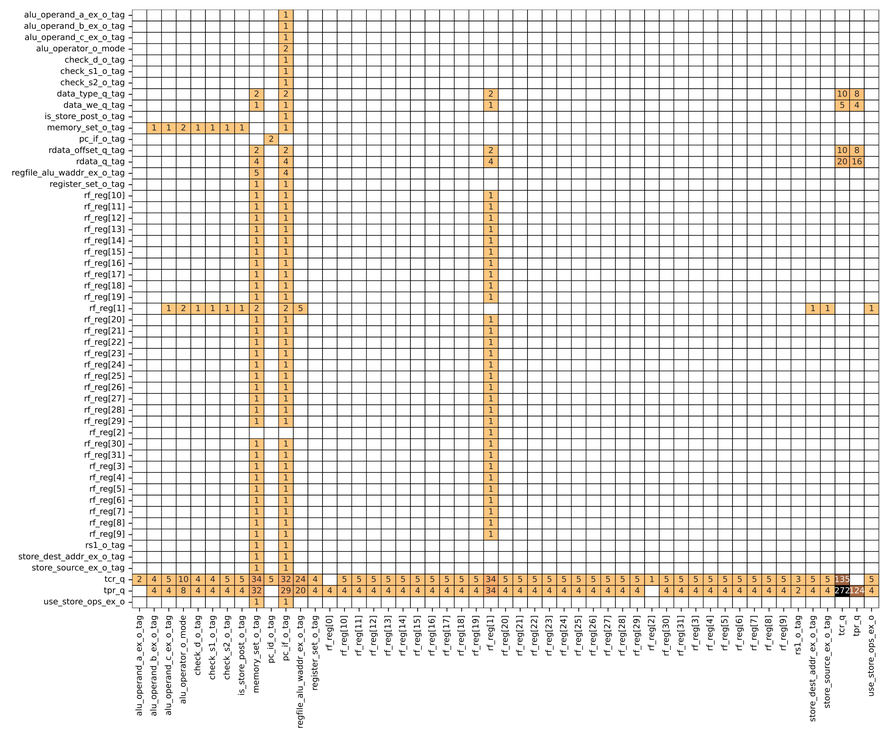
Hamming Codes are a class of linear error-correcting codes invented by Richard W. Hamming [3] in 1950. The main use of these codes is to detect and correct errors. They are mostly used in digital communication and data storage systems as error control codes.

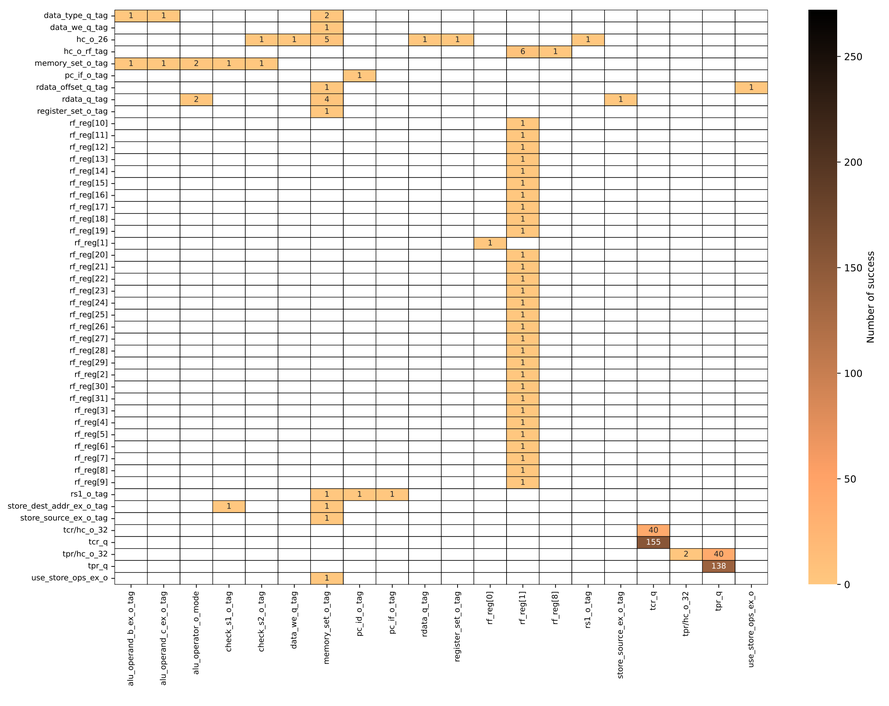


Proposed scheme for code-based protections of a set of independent registers

Experimental results

We relies on FISSA to assess D-RI5CY vulnerabilities against FIA. Moreover, we compare unprotected and Hamming Code protected versions considering a ROP attack based on a buffer overflow exploit detected by the DIFT mechanism in normal condition. However, a circuit designer may want to study the effect of FIA on such a mechanism.





Results for single bit-flip in two targets at same clock cycle for both versions

		Silent	Delay	Success	Total
Buffer overflow	No protection	45,097	1,503	1,406 (2.93%)	48,006
	Hamming Code	67,829	575	452 (0.66%)	68,856

Conclusion and Perspectives

In this work, we present a configurable open-source tool, FISSA, to automate fault injection campaigns simulation. FISSA can be used with well-known HDL simulators such as Questasim.
In future work, we plan to support new HDL simulators, extend the fault models supported and improve integration into the design workflow.

References

[1] W. Pensec, *Fault Injection Simulation for Security Assessment*. [Online]. Available: `https://github.com/WilliamPsc/FISSA`.

[2] C. Palmiero *et al.*, “Design and Implementation of a Dynamic Information Flow Tracking Architecture to Secure a RISC-V Core for IoT Applications,” in *High Performance Extreme Computing*, 2018. DOI: `10.1109/HPEC.2018.8547578`.

[3] R. W. Hamming, “Error detecting and error correcting codes,” *The Bell System Technical Journal*, 1950. DOI: `10.1002/j.1538-7305.1950.tb00463.x`.