**Thesis subject**: Formal leakage-free analysis and modelling of microarchitectural sources of leakage

**Host team**: LIP6 laboratory, ALSOC team

**Director**: Emmanuelle.Encrenaz, emmanuelle.encrenaz@lip6.fr
**Co-supervisor**: Quentin Meunier, quentin.meunier@lip6.fr

## Context

Physical attacks named side channel attacks (SCA) consist in measuring physical quantities of a computing system (power consumption, electromagnetic radiations) during the execution of a cryptographic algorithm, and using them in order to retrieve the encryption key [1, 2]. These attacks target more specifically embedded devices, such as payment cards, for which the encryption key must not be known to the user.

SCA have grown in popularity since the early 2000s. To limit their impact, two main hardening techniques have appeared: hiding [3] and masking [4]. The latter consists in breaking down the secret into several parts (shares) using masks (variables following a uniform random distribution between several executions), so that only the recomposition of the different shares makes it possible to deduce information on the original secret. The algorithm must be modified to apply an independent treatment to each of the shares.

Various works have studied how to automate the security proofs of such masking schemes [6, 7, 8, 9]: the goal of resulting tools is, from a description of the program, to verify for all the expressions manipulated by the program that their distributions are statistically independent of the secrets (typically the encryption key). Such security proofs are based on a leakage model (e.g value-based leakage model) that abstracts the real leakage. Moreover they are often applied on a high level representation of a masked program that is too far from the compiled code running on a given hardware target. To guarantee the absence of leakage when running a masked code, it is necessary to perform a leakage-free analysis on the assembly code (or binary) of the program with a sufficiently precise model of the processor [5], typically taking into account its micro-architecture and its sources of leakage.

## Subject

Recently, the framework Armistice [10] has shown the relevance of modelling the processor micro-architecture for the formal verification of masked program, using the Arm Cortex-M3 core as an example. One objective of the thesis will be to extend this work by proposing a modelling methodology starting from an RTL description. Another important part will concern the design or extension of a formal leakage-free analysis, or on how to combine a formal analysis with simulated power traces. The thesis will also include an experimental part in order to validate the different proposals (model and security analysis).

The work of this thesis will be related to the ANR IDROMEL project that has started in 2021. The objectives of this project are to analyze the micro-architectural sources of leakage and to propose different methods and tools to strengthen the software and hardware security of embedded systems against SCA. The consortium brings together Arm France, CEA, IRISA, LAAS and LIP6. The PhD candidate will interact with all partners, in particular with Arm France.
We are looking for a candidate with some of the following skills:
– understanding of processor micro-architecture, side-channel attacks based on power consumption, HDL (Verilog)
– advanced programming level in at least one language such as Python, C, C++
– code analysis including assembly
– reasonable background in mathematics and statistical analysis

## References

[1] S. Mangard, E. Oswald, and T. Popp. Power analysis attacks: Revealing the secrets of smart cards,Vol. 31. Springer Science & Business Media, 2008.

[2] Q. L. Meunier. FastCPA: Efficient Correlation Power Analysis Computation with a Large Number of Traces, in CS2'19, 2019, Valencia, Spain

[3] N. Belleville, D. Couroussé, K. Heydemann and H.-P. Charles. Automated software protection for the masses against side-channel attacks, ACM Transactions on Architecture and Code Optimization (TACO) 15.4 (2018): 47.

[4] Y. Ishai, A. Sahai, D. Wagner. Private circuits: Securing hardware against probing attacks, in Annual International Cryptology Conference. Springer, Berlin, Heidelberg, 2003.

[5] D. Zoni, A. Barenghi, G. Pelosi, W. Fornaciari. A Comprehensive Side-Channel Information Leakage Analysis of an In-Order RISC CPU Microarchitecture, ACM Transactions on Design Automation of Electronic Systems (TODAES) 23.5 (2018): 57.

[6] G. Barthe, S. Belaïd, G. Cassiers, P.-A. Fouque, B. Grégoire, F.-X. Standaert. maskVerif: Automated Verification of Higher-Order Masking in Presence of Physical Defaults, European Symposium on Research in Computer Security. Springer, Cham, 2019.

[7] Q. L. Meunier, E. Pons, and K. Heydemann. "LeakageVerif: Efficient and Scalable Formal Verification of Leakage in Symbolic Expressions," in IEEE Transactions on Software Engineering, 2023.

[8] I. Ben El Ouahma, Q. L. Meunier, K. Heydemann, E. Encrenaz. Side-channel robustness analysis of masked assembly codes using a symbolic approach, Journal of Cryptographic Engineering (JCEN), March 2019

[9] Pengfei, G., Hongyi, X., Sun, P., Zhang, J., Song, F., & Chen, T. (2020). Formal Verification of Masking Countermeasures for Arithmetic Programs, IEEE Transactions on Software Engineering.

[10] A. d. Grandmaison, K. Heydemann and Q. L. Meunier, "ARMISTICE: Microarchitectural Leakage Modeling for Masked Software Formal Verification," in IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 11, pp. 3733-3744, Nov. 2022