

CHIST-ERA Projects Seminar RTCPS *SECODE Project*

Jean-Luc Danger

Paris, April 12th, 2018



Programme co-funded by the
EUROPEAN UNION



Secure Codes to thwart Cyber-physical Attacks

- ❖ To specify and design error correction codes suitable for an efficient protection of sensitive information in the context of Internet of Things (IoT) and connected objects.
- ❖ The codes can mitigate:
 - **passive attacks**, like side-channel analysis.
 - **active attacks**, like fault injection attacks.



SECODE GOALS

- ❖ The objective is to **leverage the codes** for protecting against cyberphysical attacks.
- ❖ The protections by codes will take advantage of **formal mathematical approaches** which can prove their effectiveness from both the information theory and the physical implementation level.
- ❖ An **automatic compiler** and a demonstrator will be performed on real platform.



chist-era

SECODEpartners

UCL

Université
catholique
de Louvain



LTCI



Inria

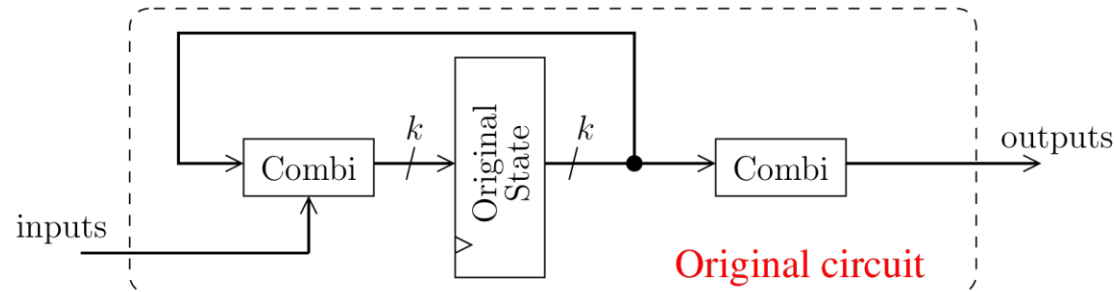
INVENTEURS DU MONDE NUMÉRIQUE

Sabancı
Universitesi

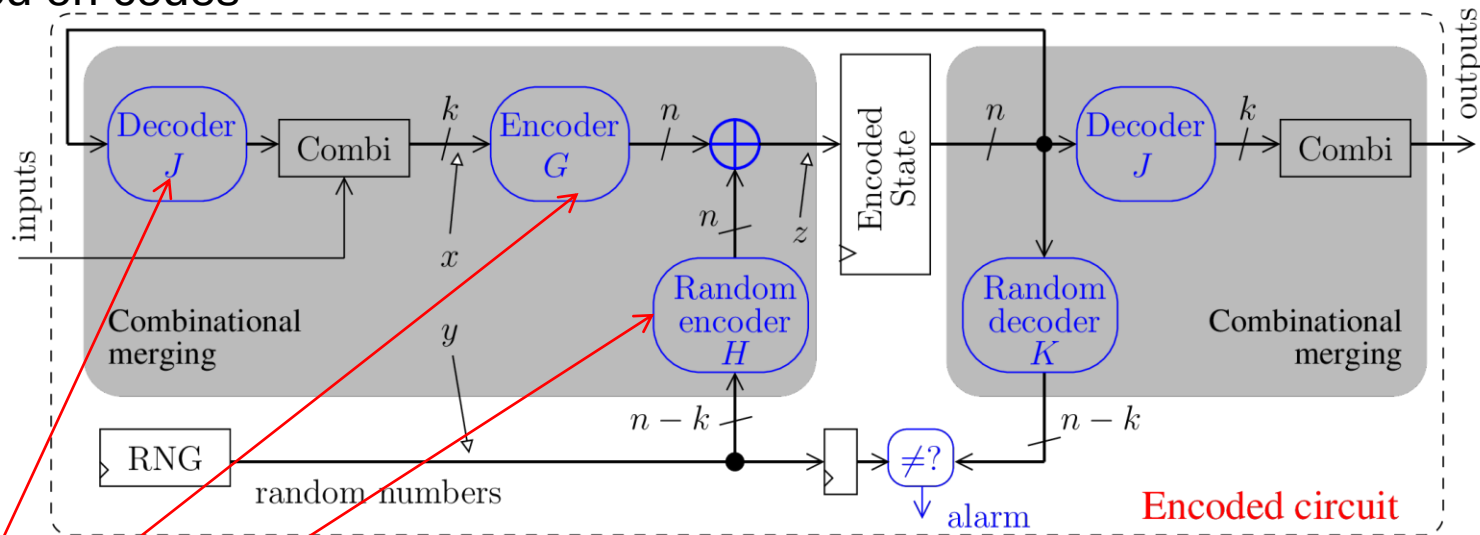
UNIVERSITÉ
PARIS8

VINCENNES-SAINT-DENIS

SECODE big picture



With masking protection
based on codes



What are the best Codes ?
LCD Code ?

What is the best implementation with maximum security ?
Can we implement the masking by codes automatically ?

SECODE challenge examples

❖ **Code specifications:**

- Design of "Linear Complementary Dual" LCD codes
- which are "Generalized Quasi Cyclic" (GQC).
- Relations between LCD codes and AG codes

❖ **Physical attacks:**

- Masking and Detection Multivariate secure with Codes.

❖ **Cyber physical protected platform:**

- Automatic insertion of protections based on codes during compilation.

Major achievements and output

❖ Code theory

- ✓ Generalized Quasi-Cyclic Codes $\Rightarrow$ LCD
- ✓ AG Codes $\Rightarrow$ LCD
- ✓ Any linear code with $q > 3$ can be equivalent to a LCD code
- ✓ Characterization of LCP codes $\Rightarrow$ equivalent to LCD

❖ Code security at implementation level

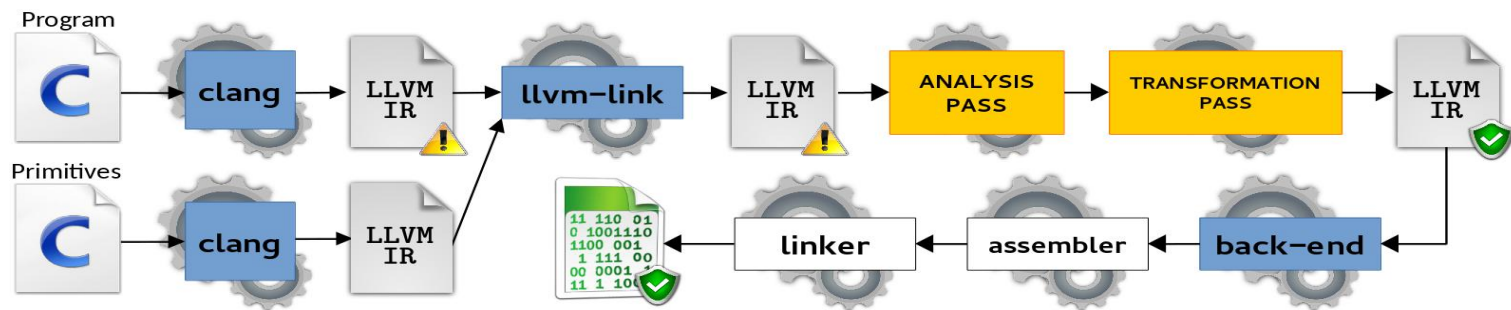
- ✓ Generic security parameters for both SW and HW implementation of masking
- ✓ Codes used for PUF

Major achievements and output

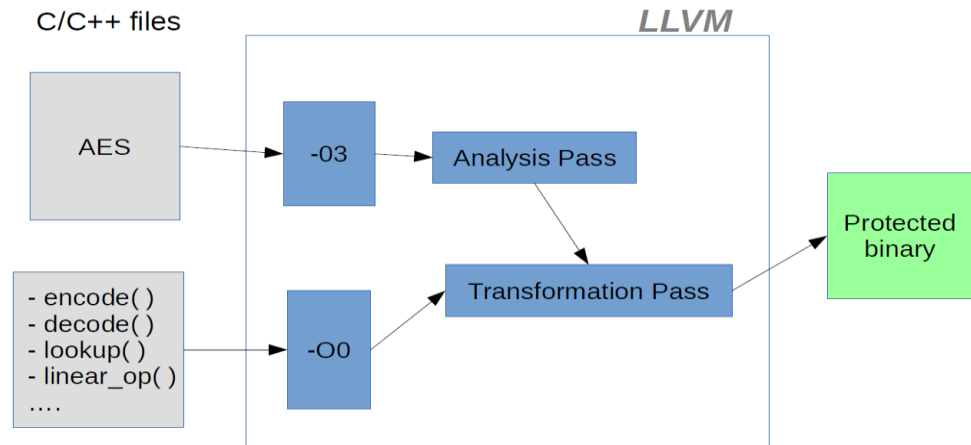
❖ Automatic compilation

✓ Demonstration with generic masking using LLVM

Flow of the method:



Demonstration with AES:





Upcoming challenges and needs

- ❖ To find codes to be robust against both SCA and FIA
- ❖ To refine code/implementation to reduce the physical leakage
- ❖ To optimize complexity and latency of the automatically compiled code



Questions ?