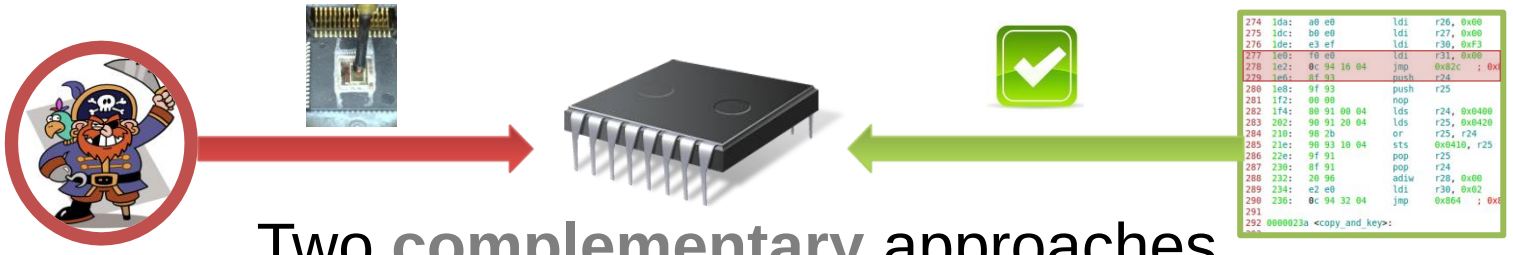
Security of assembly programs against faults

Target

Improve **security**, **availability** and **fault tolerance** of secure integrated circuits based on a microcontroller

Innovation

Design a **fault tolerant** structure for an **assembly code** on a standard up-to-date microcontroller versus fault injections from a realistic **fault model**



Two **complementary** approaches

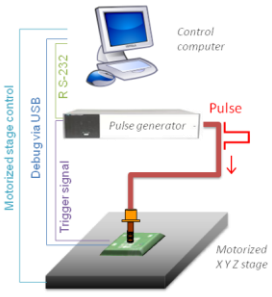
Fault model — 1

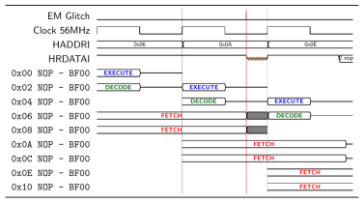
EM injections on µ-controllers (ATmega128 and STM32)

➔ Build a **realistic** set of attack possibilities so we can understand the fault model more clearly and give a list of possible **attack paths**

Current results:

- inject some faults on the **bus transfers**
- skip some assembly **instructions**
- replace some **instructions** by others
- change some **register** values





Work in collaboration with Amine Dehbaoui (ENSM.SE)

2 — Fault tolerance

Fault model made of **instruction skips** and data bus corruptions

➔ Propose a **fault-tolerant** code structure which could ensure a **correct execution** of the program even with possible instruction skips and bus corruptions

Current results:

- design a **duplication-based** countermeasure
- temporal redundancy in a **short time interval**
- could be easily adapted to **any compiled code**
- **formal proof** for the correctness with Vis

Standard code

```
ADD   R1, R1, #1
CMP   R1, #9
B     <label>
```

Fault tolerant code

```
ADD   R3, R1, R1
ADD   R3, R1, R1
MOV   R1, R3
MOV   R1, R3
CMP   R1, #9
CMP   R1, #9
B     <label>
B     <label>
```

Work in collaboration with Karine Heydemann (LIP6)

Current issues

- **Fault model:** perform a more precise characterization of the instruction replacement faults
- **Fault tolerance:** experimentally test our fault tolerance approach to evaluate its interest

Ph.D student: Nicolas Moro — *nicolas.moro@mines-stetienne.fr*
Joint advisor: Bruno Robisson — *bruno.robisson@cea.fr*
Advisor: Emmanuelle Encrenaz — *emmanuelle.encrenaz@lip6.fr*