Alain MERLE
CESTI LETI
CEA Grenoble
Alain.merle@cea.fr

Security testing of hardware product

Abstract

- « What are you doing in ITSEFs ? »
  - Testing, Security testing, Attacks, Evaluations, Common Criteria, Certification, …
- Security evaluations:
  - The French Certification Scheme
  - The Common Criteria
  - Smartcards evaluations
- Smartcard security testing
  - Strategy
  - Attacks

Common Criteria

The basic ideas

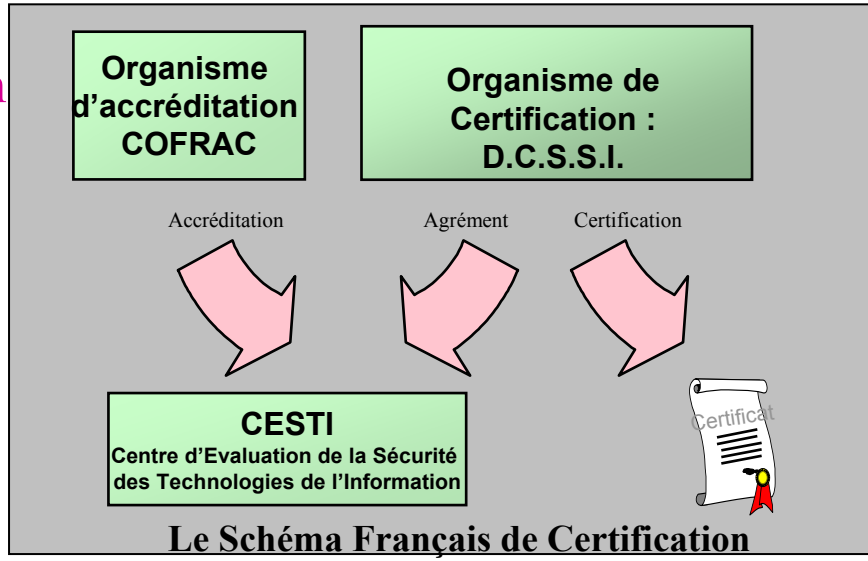
- Describe what is the security of a product
- Verify that the developer has done what it was supposed to do (and only that)
- Test (functional and attacks) the product
- Verify environmental constraints

- A standardized, objective and efficient Security Analysis Method (ISO IS 15408)
- An International Recognition through Mutual Recognition Arrangements.
- In Europe, mostly used for ***smartcards***
  - Integrated Circuits
  - IC with embedded software

CESTI LETI

Information Technology Security Evaluation Facilities

- ITSEF of the French Certification Scheme
- Area : hardware and embedded software
  - Smartcards
  - Security equipments
- Level: Up to EAL7
- Located in Grenoble
- Part of the biggest French Research center in Microelectronics



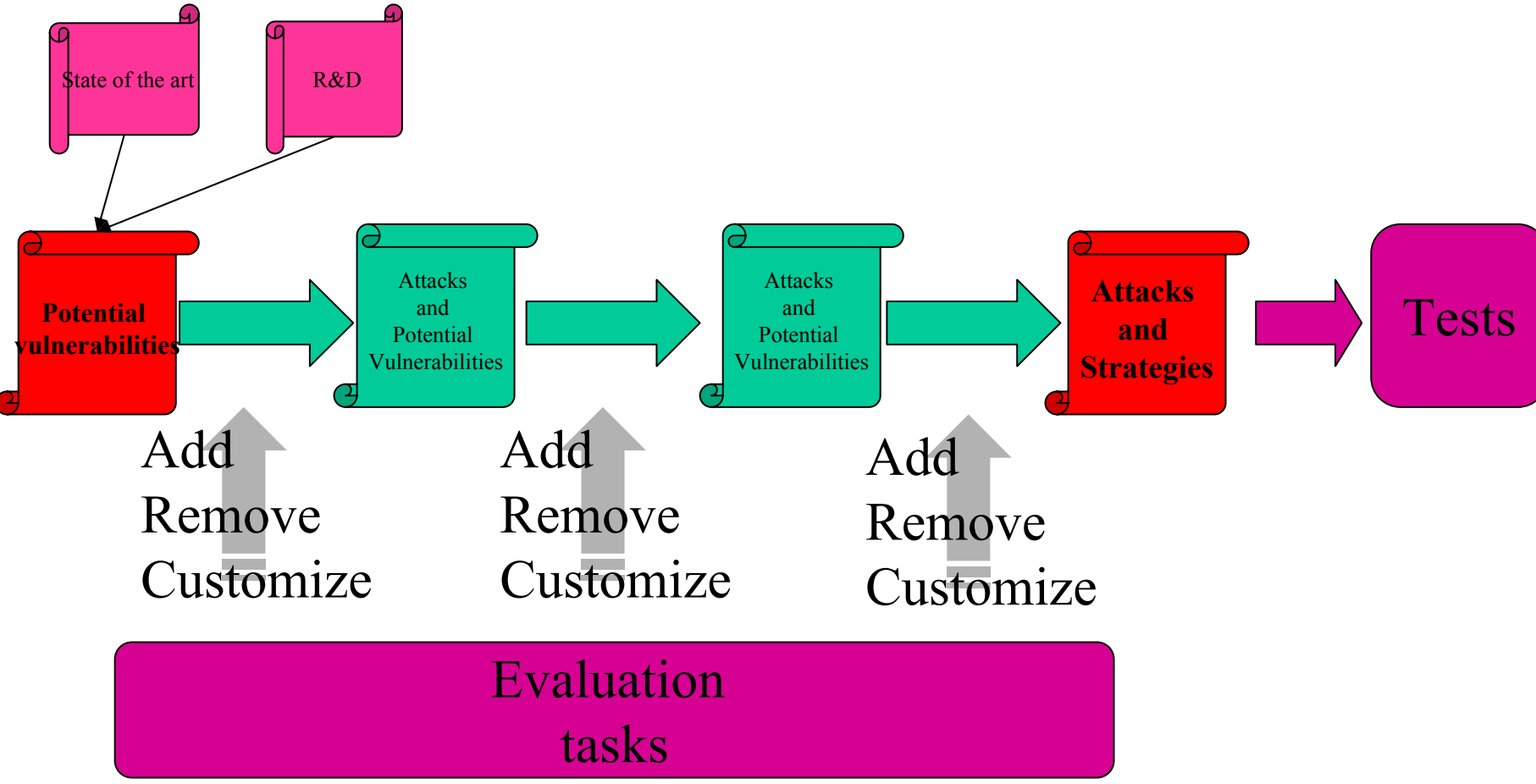
Smartcard evaluation

- Common Criteria, EAL4+ level
  - High Security level (banking applications)
  - White box evaluation
    - Design information
    - Source code
- A table defining the « attack potential »
  - Time, expertise, equipment, knowledge, …
  - The card must resist to the « maximum » (ie all realistic attacks)

What kind of testing ?

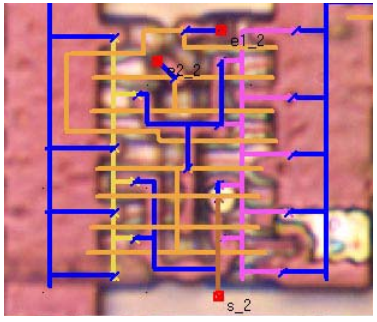
- Functional testing but security oriented
  - Are the Security Functions working as specified ?
- Attacks
  - Independent vulnerability analysis
  - Higher levels (VLA.4): adaptation of the classical "attack methods" to the specificities of the product

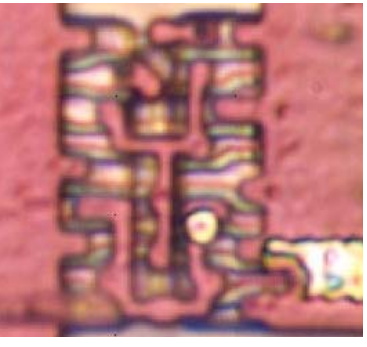
Test strategy (Attacks)

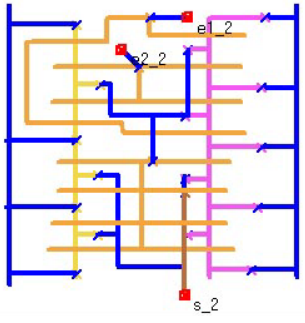


Attacks on smartcards

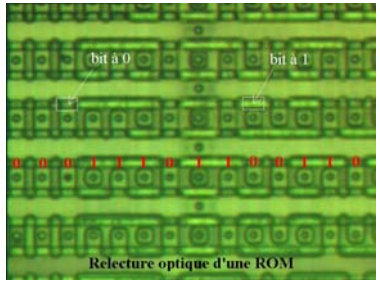
- Physical (Silicon related)
  - Memories
  - Access to internal signals (probing)
- Observation: Side Channel Analysis
  - SPA, EMA, DPA, DEMA
- Perturbations: inducing errors
  - Cryptography (DFA)
  - Generating errors
    - IO errors (reading, writing)
    - Program disruption (jump, skip, change instruction)
    - Dynamic rewriting of the code
- Specifications/implementation related attacks
  - Protocol, overflows, errors in programming, …

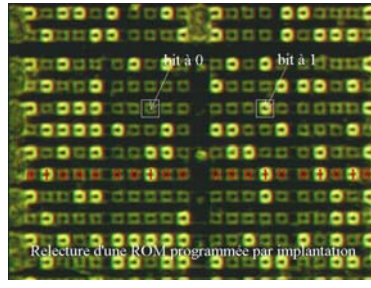






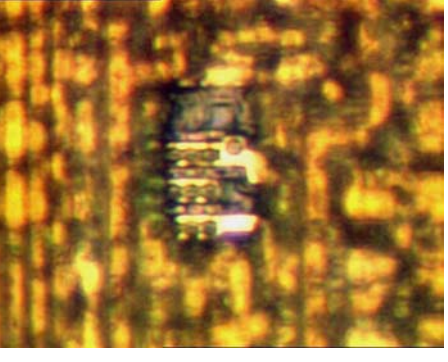
Reverse Engineering

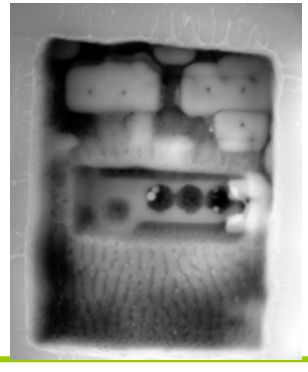




Optical reading of ROM

Probing : laser preparation





Probing : MEB

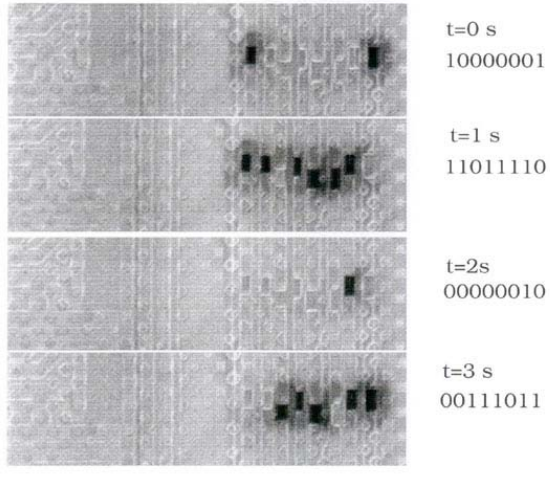
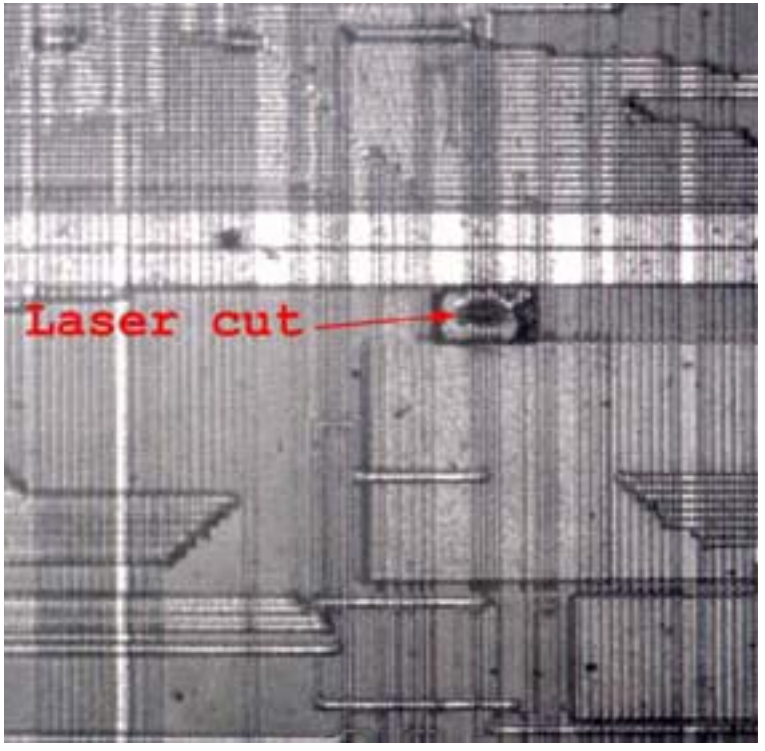
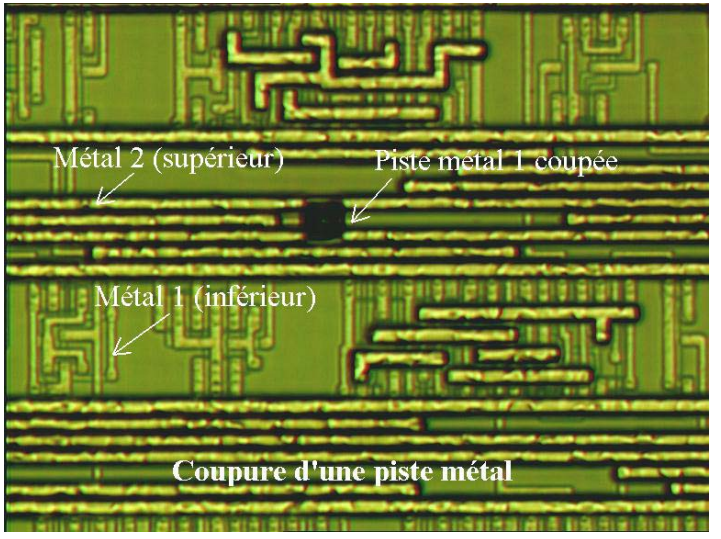


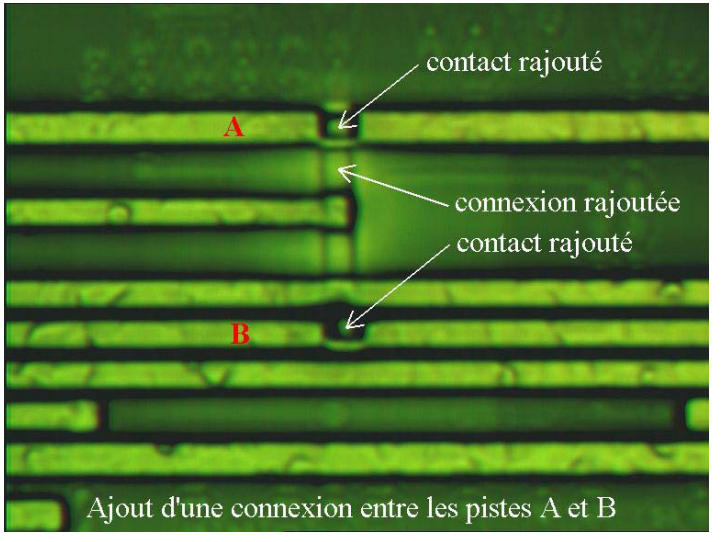
Figure 2: Image sous faisceau d'électrons en contraste de potentiel des états électriques des lignes du bus de données en fonction du temps.



Modification : Laser cut

Modification : FIB

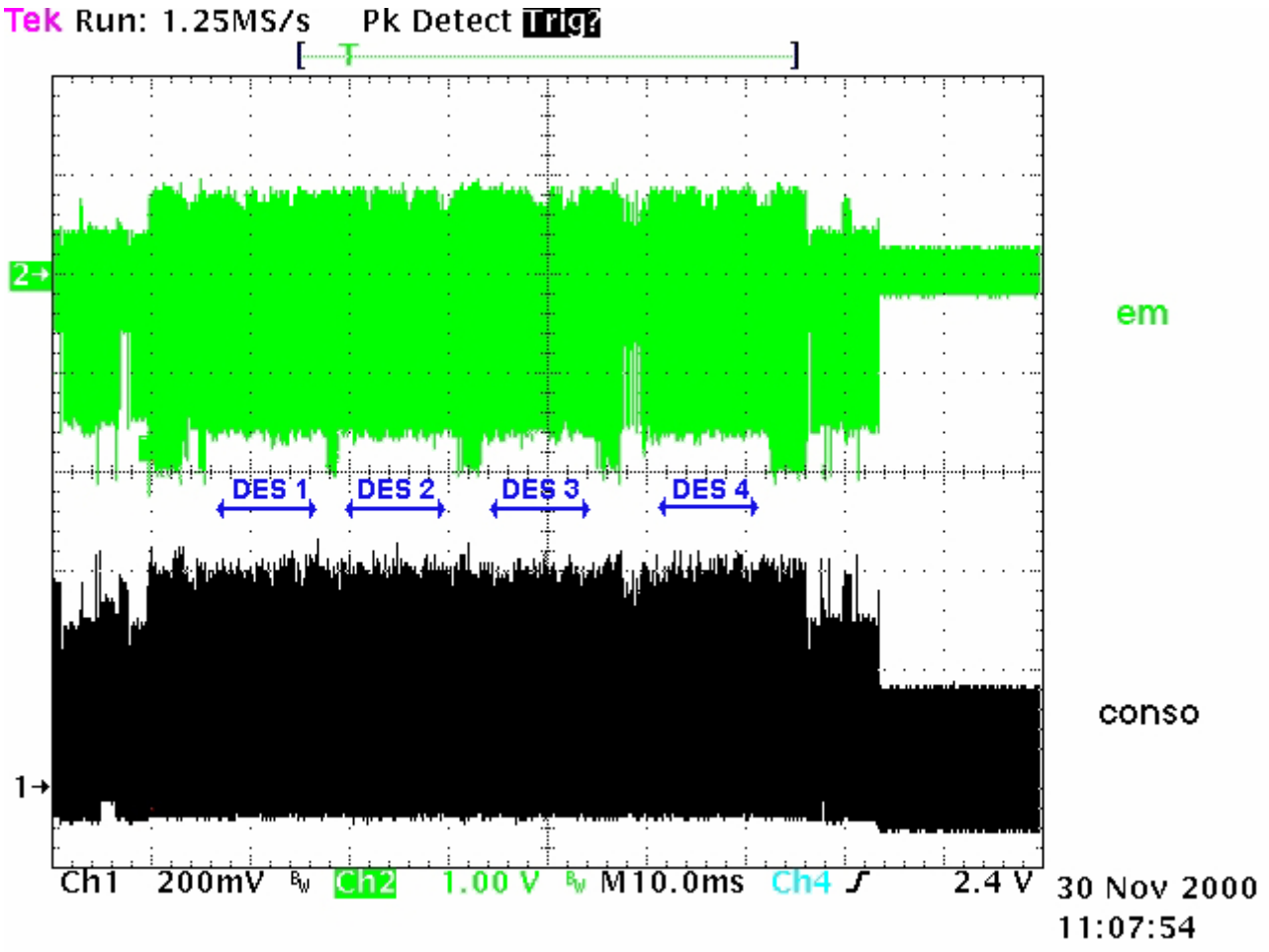




Basic attack strategy

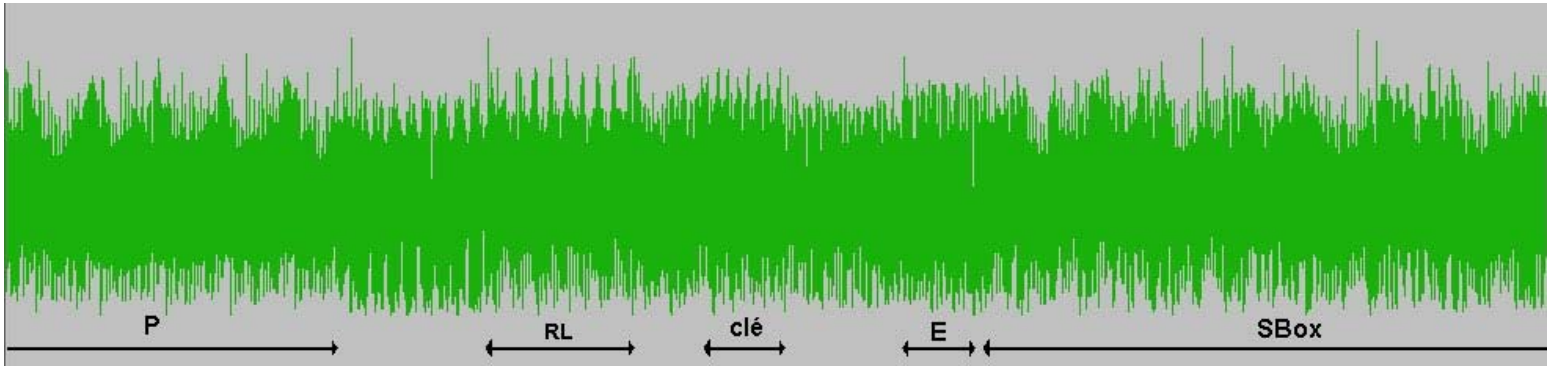
- Observation (SPA, EMA, Cartography)
  - Find an « interesting » location (time and space)
  - Synchronization
- Data acquisition or Perturbation
- For perturbation
  - Not a 100% predictable effect
  - Repetition required

EM signal analysis

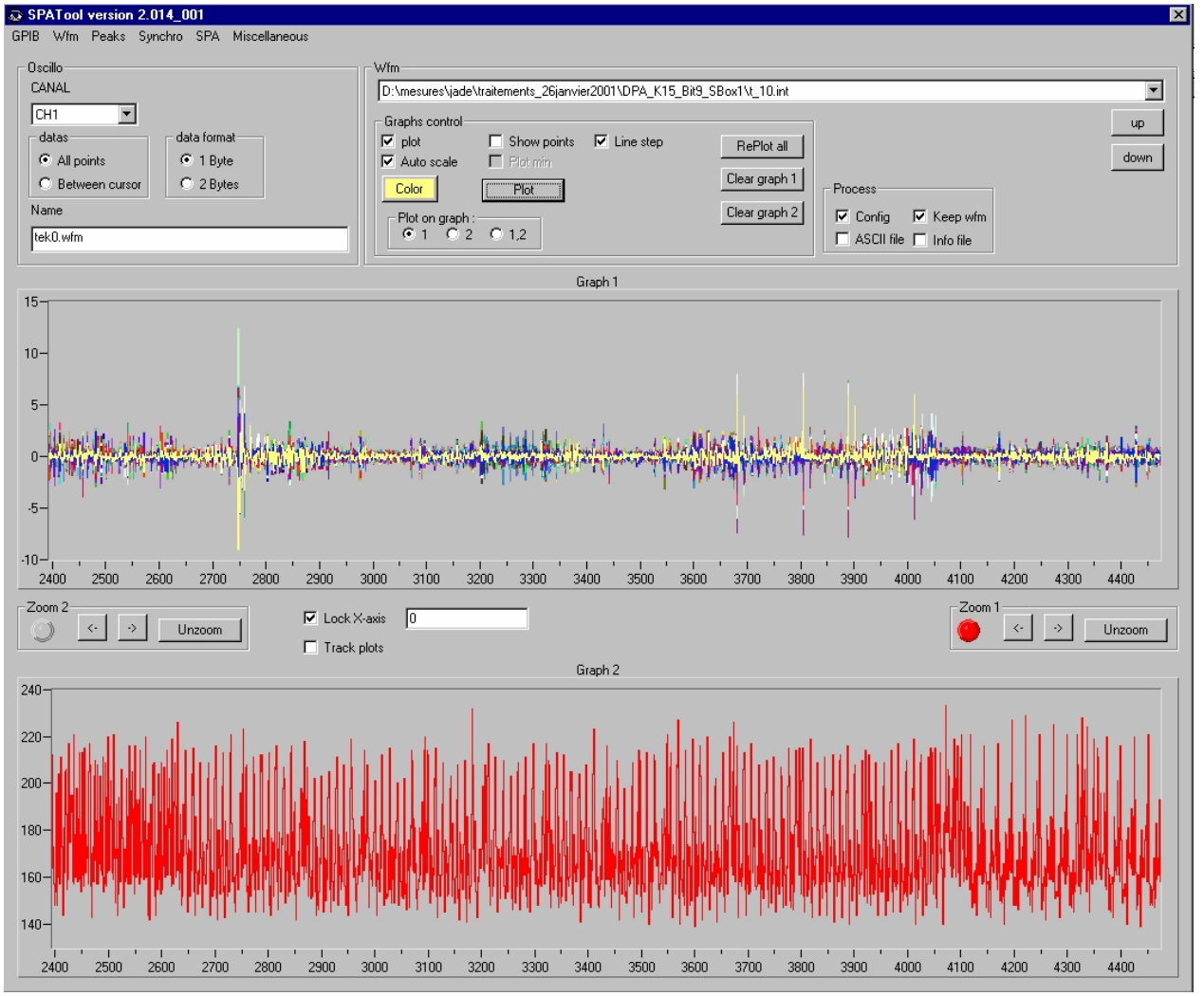


SPA/EMA Analysis

DES



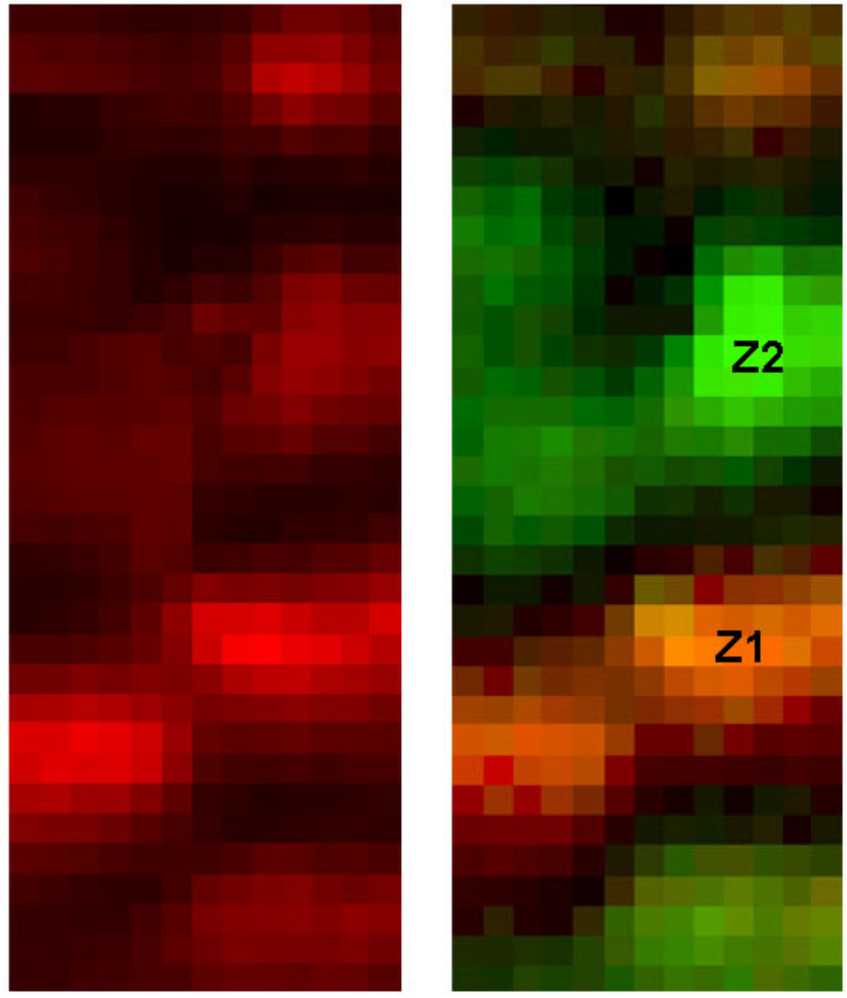
SPA/DPA analysis



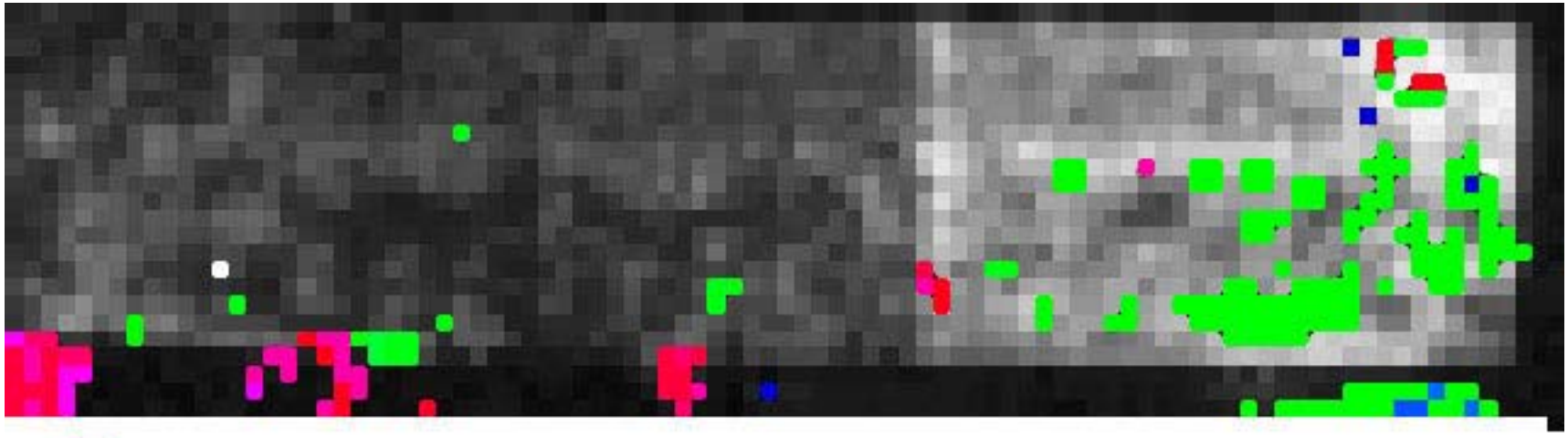
Cartography

Electro-magnetic signal during DES execution.

- Hardware DES
- Differential signal



Cartography



Light perturbation of Hardware DES

Perturbations examples

Initializations

valid = TRUE;

If got ^= expected then
valid = FALSE ;

If valid Then
    critical processing;

Branch on error

Non critical processing;
If not authorized then goto xxx;
Critical processing;

Re-reading after integrity checking

Memory integrity checking;
Non critical processing;
Data 1 reading;
Critical processing;
Data 2 reading;
Critical processing;

The race ...

- Challenge between
  - Attacks
  - Counter measures
- Today an attacks
  - Is based on an attacks method ex DPA, DFA
  - But is mainly attacking the counter measures
    - Signal processing, synchronization, anti suicide, safe errors, …

Examples (1)

DPA

- DPA theory
- De synchronization (internal clocks, random IT, fake code, …)
- Then signal processing
- Then masking techniques
- Then high order DPA
- Then smoothing the consumption signal
- Then EM based attacks
- …. to be continued

Example (2)

Perturbations

- Glitches based
- Then detectors and filters
- Then laser based
- Then integrity checking
- Then multiple perturbations
- …. To be continued

Example (3)

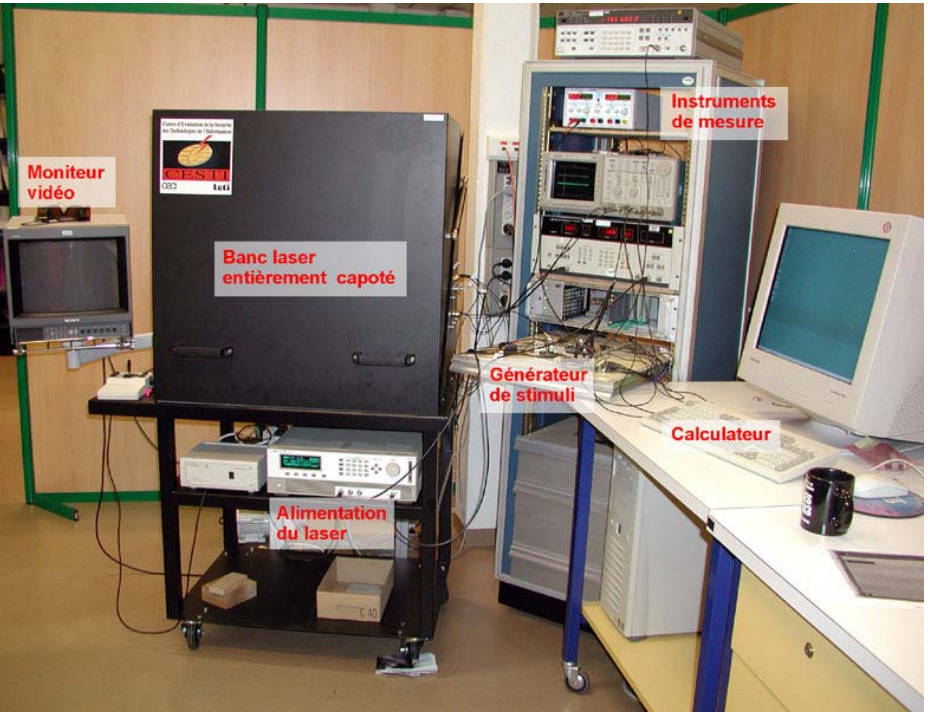
When counter measures induce vulnerabilities

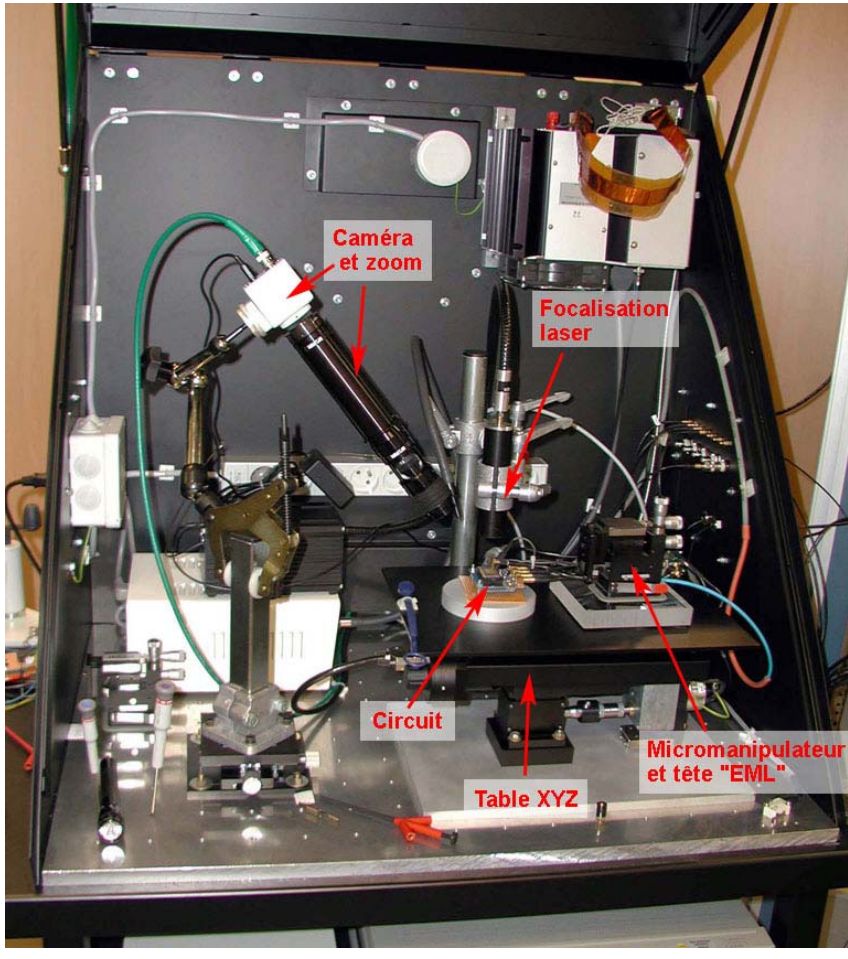
- Counter measure:
  - Performing the processing twice
  - If results are different then security reaction
- Attack
  - Generate a controlled error (setting a bit to 1)
  - If no reaction, then the value was 1

What is requested from a lab ?

- Good knowledge of the state of the art
  - Not always published
- Internal R&D on attacks
  - Equipment
  - Competences
- Multi-competences
  - Cryptography, microelectronics, signal processing, lasers, etc
- Competence areas defined in the French Scheme
  - Hardware (IC, IC with embedded software)
  - Software (Networks, OS, …)

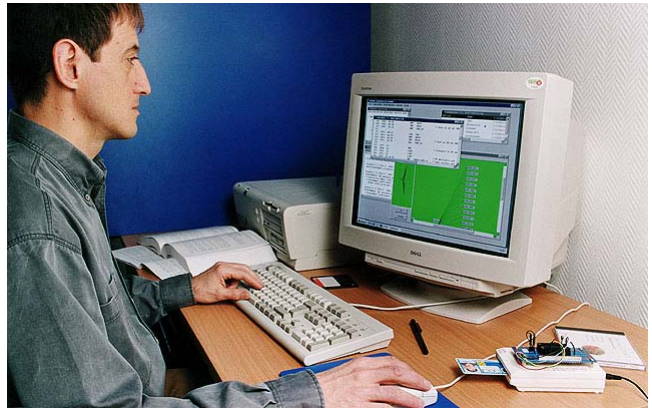
Test benches



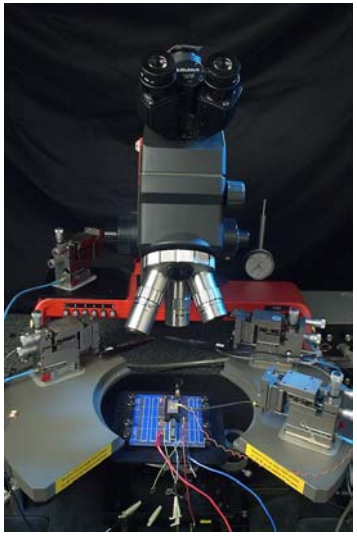


Competences

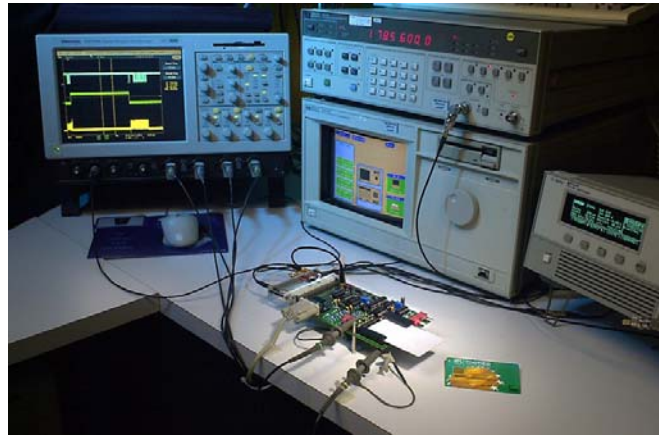
Microelectronic



Software



Testbenches



Some rules

- Security is the **whole product**: IC + software
- The IC must **hide itself**
  - **If you can see it, you can attack it !**
  - Critical processing,Sensitive data handling,Consistency checking, Memory access, …
- The IC must **control itself**
  - **Am I doing what I was supposed to do ?**
  - Consistency checking,Audits, log, …
- But attacks are now dedicated to counter-measures

CONCLUSION (1)

- Evaluation is
  - Rigorous & normalized process
  - But attacks require specific « human » skills
- Attack is
  - Gaining access to secret/forbidden operations
  - Free to « play » with abnormal conditions
  - An error is not an attack
    - But an error can often be used in attacks
    - An attack requires an “attack strategy”

CONCLUSION (2)

- The evaluation guarantees that
  - The product is working as specified
  - It has a “good” resistance level
    - At a specific time
    - Perfection as absolute security does not exist