Introduction to Fault Attacks

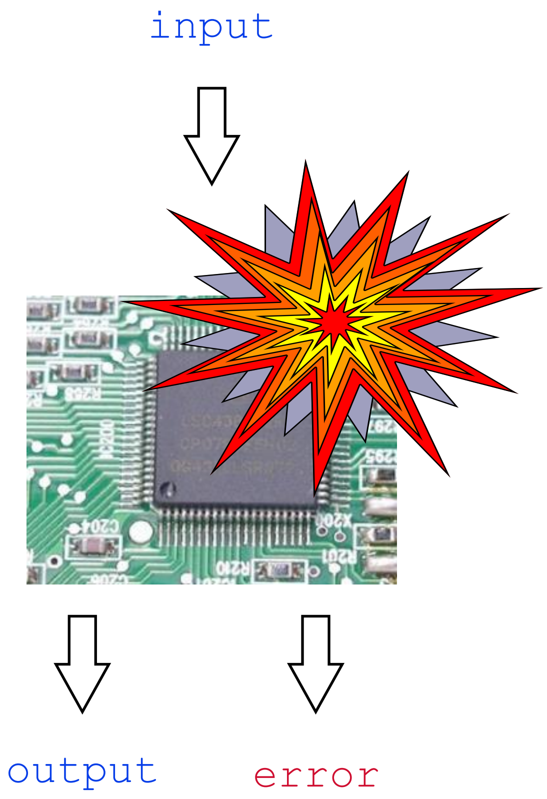
Josep Balasch
KU Leuven ESAT / COSIC

IACR Summer School 2015

Chia Laguna, Sardinia (Italy)
19 October 2015

What are fault attacks?

- **Active** attacks against cryptographic implementations
- Electronic devices are subject to (usually) rare faults
  - Software
  - Hardware
- Reason: combination of strange events
- A fault can cause errors
- An errors can be exploited to expose secrets



History

- Single Event Upsets (SEU)
  - Random bit flips occurring in storage elements

1950s | 1960s | 1970s | 1980s

Ground nuclear testing

anomalies in electronic monitoring equipment

Aerospace industry

problems in space electronics: soft-fails

IBM research

effects of alpha particles on semiconductor electronics

[ZL79]

From *accidental* faults to *intentional* faults

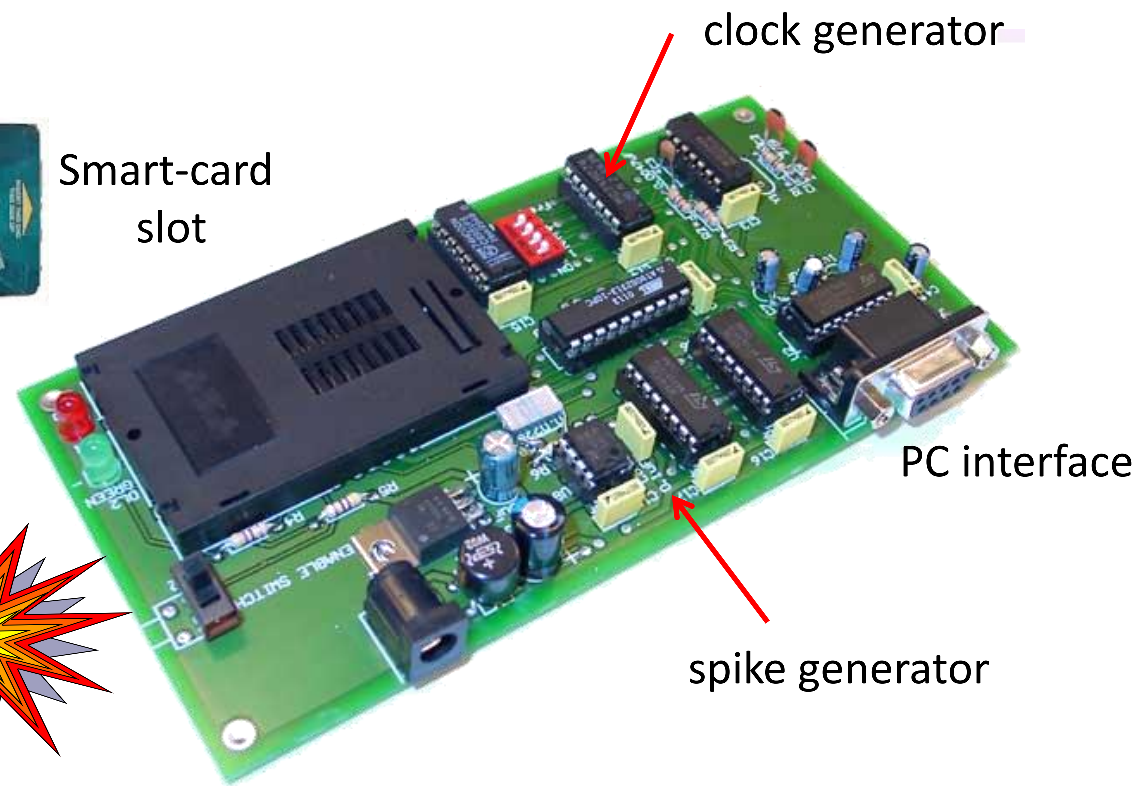
- #1: Hacking community vs. DirecTV (late 90s)
  - PayTV technology, broadcast only
  - Smart-card based subscription model
  - Phone line to communicate with provider

- Hacking community:
  - Read/write access to smart cards
  - Change to unlimited subscription model
- Reply from DirecTV
  - Possibility to update cards through broadcast channel
  - Disable hacked cards by inserting an inifinite loop

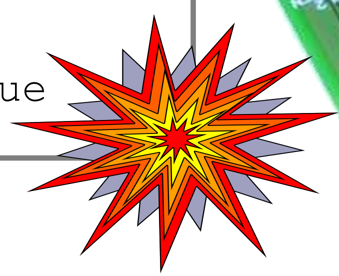
```
…          // booting
inf_loop:
 JMP inf_loop
…          // continue
```

From *accidental* faults to *intentional* faults

- Reply from the hacker community
  - Unlooper: device that was able to unlock the card




```
…         // booting
inf_loop:
 JMP inf_loop
…         // continue
```



From *accidental* faults to *intentional* faults

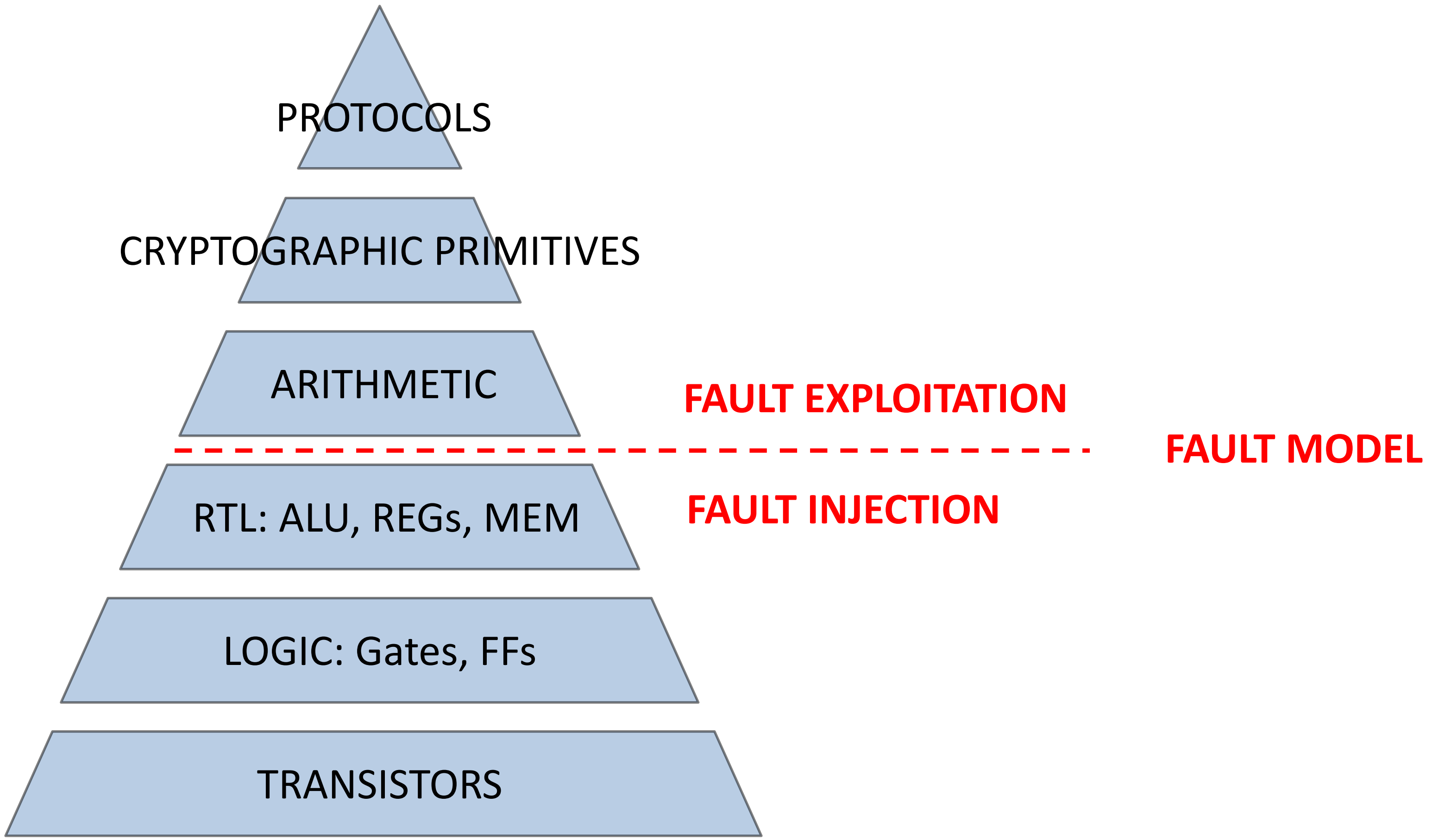
- #2: The Bellcore Attack [BDL97]
  - Target: implementations of RSA with CRT
    - Main operation: $s = m^d \bmod n$, where d is private key
    - Security of RSA: intractability of factoring large integers ($n = p \cdot q$)
    - RSA-CRT allows to speed-up computations:

$$s_p = m_p^{dP} \bmod p$$
$$s_q = m_q^{dQ} \bmod q$$
$$s = (((s_q - s_p) \cdot p_{inv}) \bmod q) \cdot p + s_p$$

  - Attack steps:
    1. Input m, collect s
    2. Input m, inject any fault on s_p or s_q, collect $\hat{s}$
    3. Compute $\gcd(s - \hat{s}, n)$ to factorize RSA modulus
  - Devastating effects
  - Today countermeasures extensively studied and deployed

The fault attack jungle

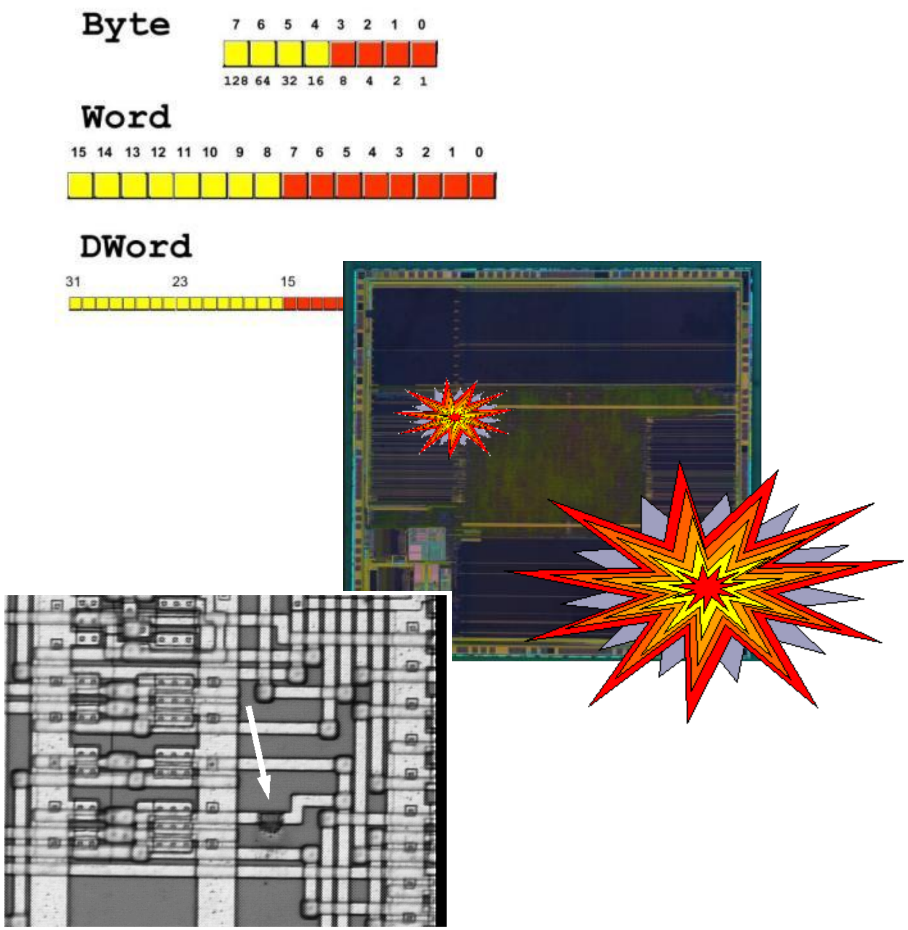
- The embedded design space



[VKS11]

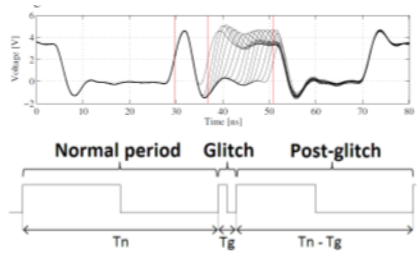
The fault model

1. Granularity: how many bits dare affected by the fault?
    1. Single bit
    2. Few bits
    3. Word
2. Modification (aka fault type)
    1. Stuck-at, e.g. zero or one
    2. Flip
    3. Random
3. Control: on the fault location <u>and</u> on timing
    1. Precise
    2. Loose
    3. None
4. Duration or effect of the fault
    1. Transient
    2. Permanent
    3. Destructive



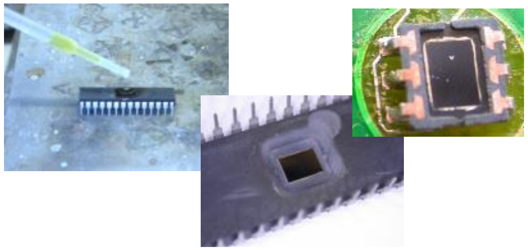
Categories of fault injection

- <u>Non-invasive</u>
  - No physical damage to device
  - Modify working conditions
  - Moderate knowledge/equipment



- <u>Semi-invasive</u>
  - Chip decapsulation
  - Milling, etching, cleaning
  - Affordable equipment

src: AirClean Systems



src: Dr. Sergei Skobogoratov

- <u>Invasive</u>
  - Establish electrical contact to chip
  - Modification, destruction, ...
  - Expensive equipment, e.g. semiconductor diagnostics

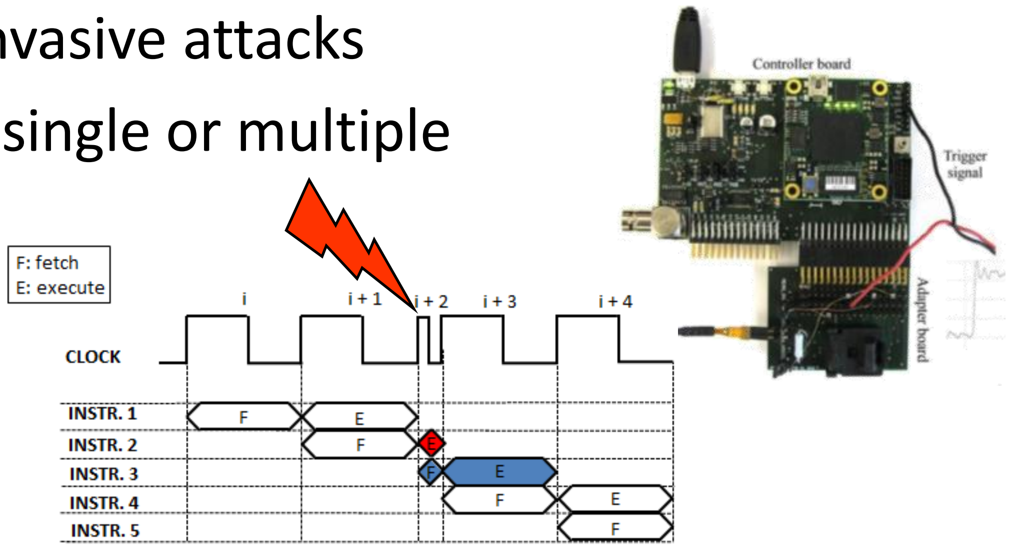
src: ZEISS

src: Bridge Technology

Glitches and spikes

- Most popular form of non-invasive attacks
- Both precise timing control, single or multiple

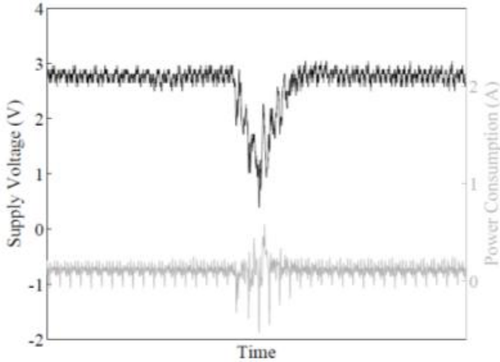
- Clock glitches
  - Temporal overclocking
  - Critical path violations



[BGV11]

- Voltage spikes
  - Temporal switch to higher (or lower) voltages

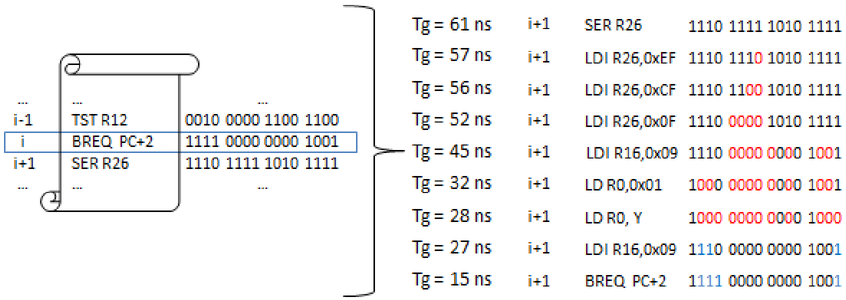
[KQ07]



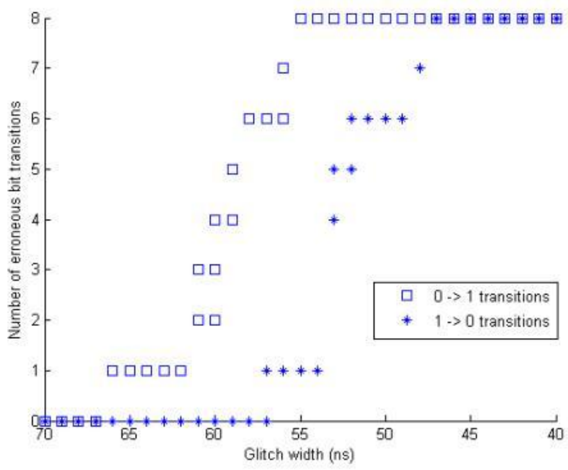
[SH08]

Glitches and spikes

- Effects on program flow
  - Replacement of instructions (sometimes skipping)
  - Tampering with loops and conditional statements
  - Change of program counter



- Effects on data flow
  - Computation errors
  - Corrupted memory pointers
  - No bit transitions on data bus

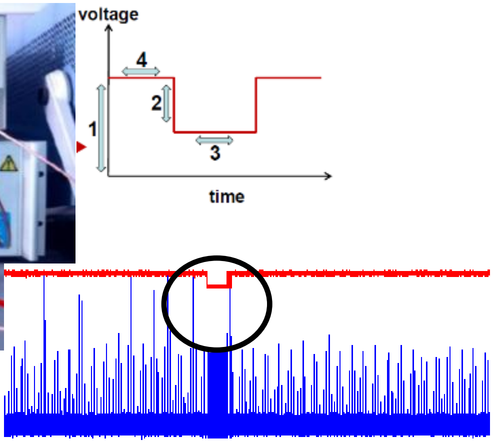


[BGV11]

Other Non-invasive Methods

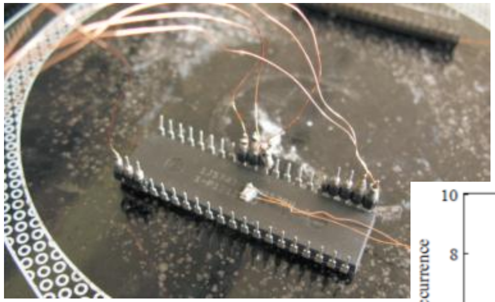
- Underpowering
  - Reduce supply voltage
  - Transient vs. Permanent
  - Increase propagation delay of combinational logic

src:: EMSE

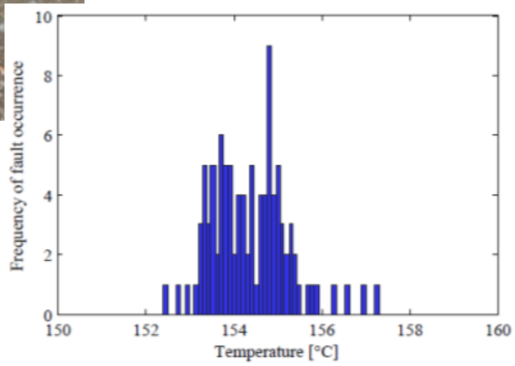


[BGVLV12]

- Temperature
  - Device on heating plate
  - Errors appear for a short window
    - Low-controlability
    - Low-frequency
  - Cooling: data retention

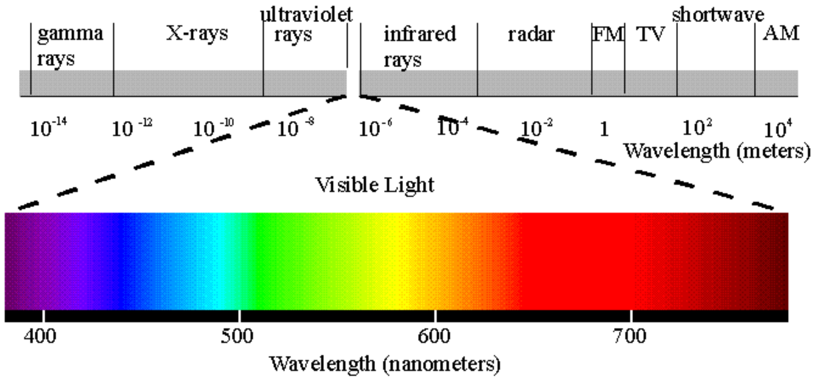


[HS13]

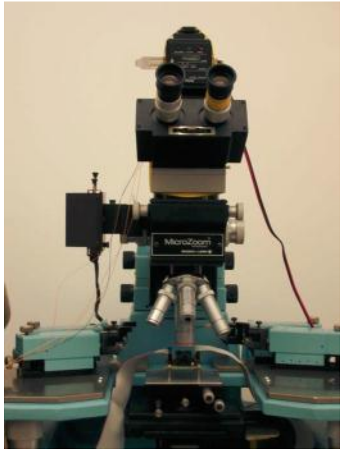


Optical Fault Injection

- Semiconductors are inherently sensitive to light
- Effect of optical pulses
  - Switching a transistor
- The chip die needs to be exposed
  - Semi-invasive method



- Example of fault injection setups:
  - Photo flash in micro-probing station
  - Laser beam on XY table, with microscope view and camera

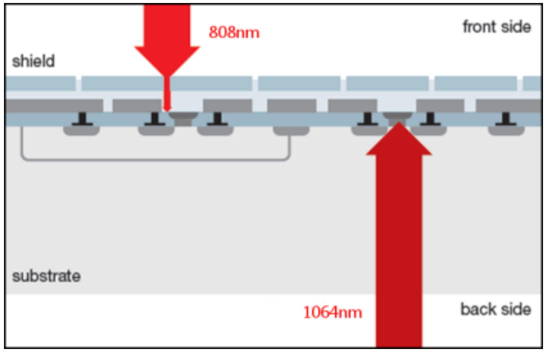


[SA02]

src: Opto

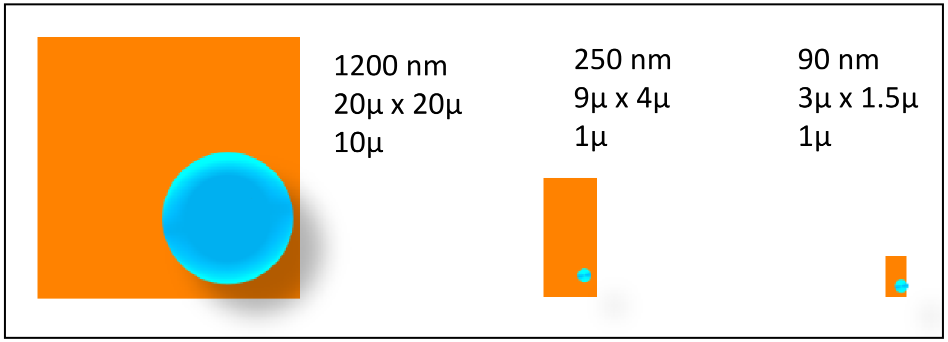
Optical Fault Injection

- Many configurable parameters
  - Position (X,Y coordinates)
  - Wavelength
  - Spot size
  - Energy / Peak power
  - Pulse vs. Continuous
  - Repetition rate
  - ...

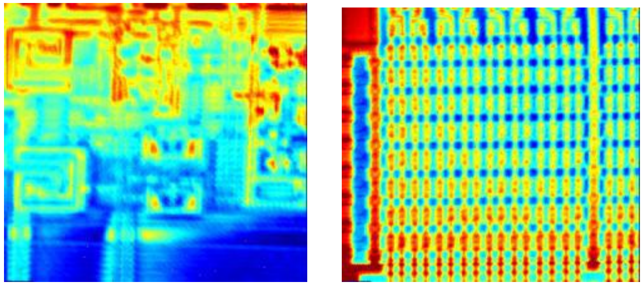


[WWM11]

[CLFT14]



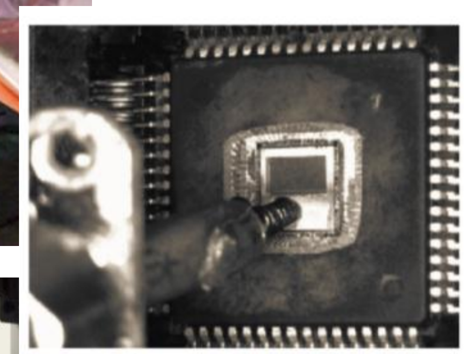
- Search space grows exponentially !
- Many fault models possible

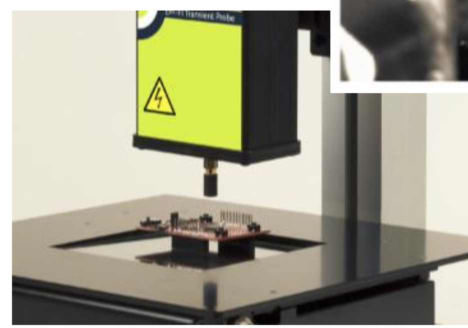


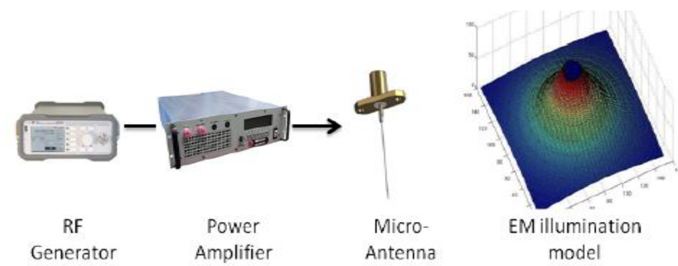
src: Dr. Sergei Skobogoratov, Semi-invasive attakcs, page 98

EM Fault Injection

- Injection of faults via the EM channel [QS3]
  - Induction of Eddy current
    - Camera flash-gun connected to an active probe
    - Spark-gap transmitter
    - EM Pulses with micro probes
  - Effects:
    - Switching transistors
    - Critical path violations
  - (Non-) and semi- invasive approach

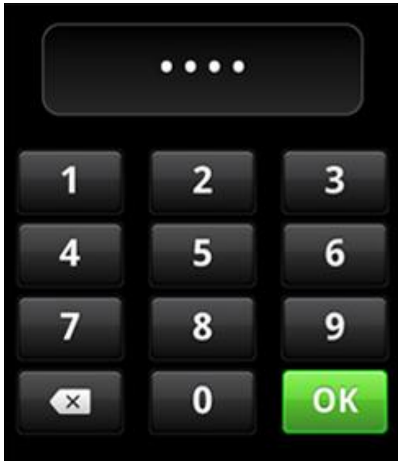






Back to the PIN example

- Assume the function *check(…)* runs in constant time

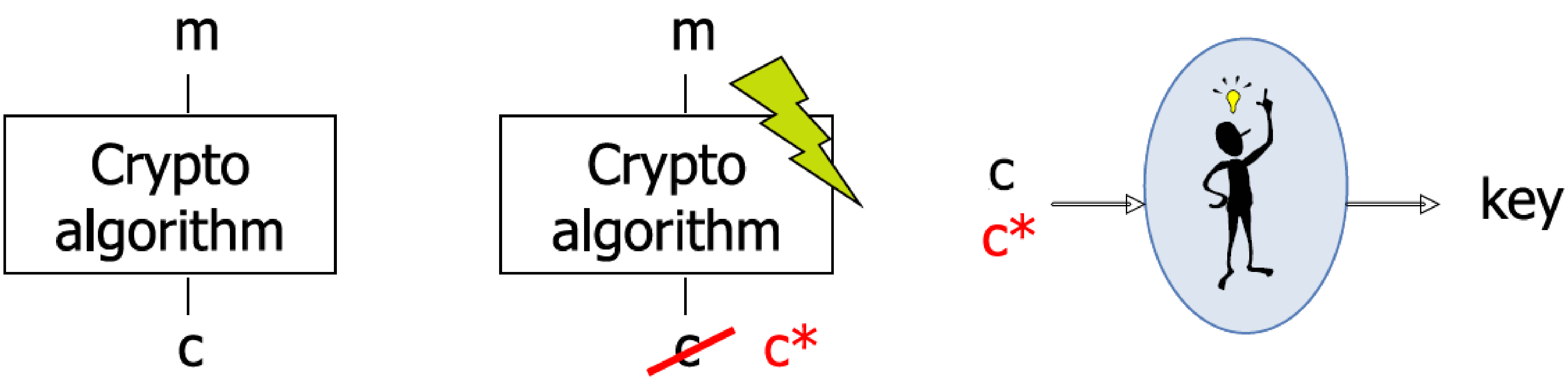



```
MAIN FUNCTION
  …
  IF check(…) == -1
    COUNTER++
  ELSE
    COUNTER = 0
  …
```

- Attacker can target the main function with an active attack
  - "Skip" conditional statement
    - E.g. by glitches/spikes during condition check
  - Prevent the counter increase
    - E.g. by disconnecting power supply
  - …

Differential Fault Analysis

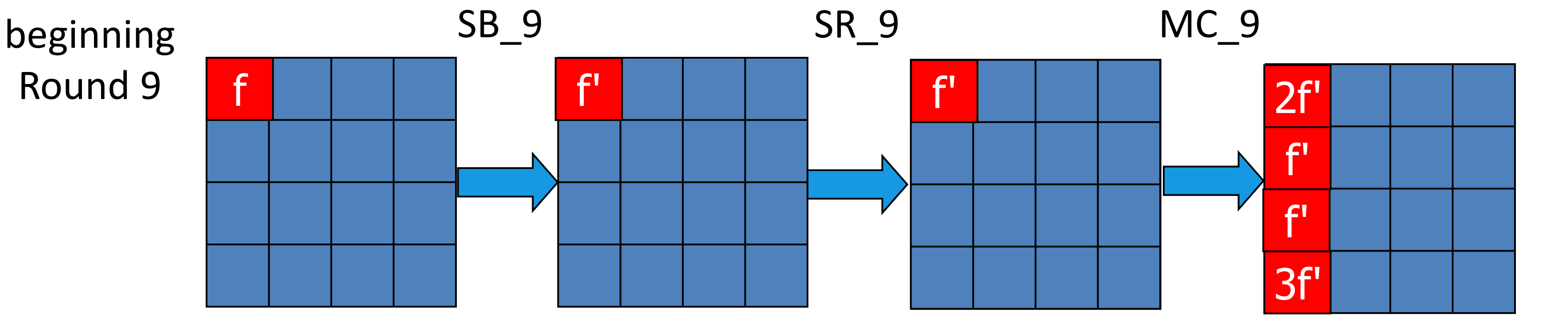
- Ask for a cryptographic computation twice
  - With any input and no fault (reference)
  - With the same input and fault injection
- Infer information about the key from the output differential

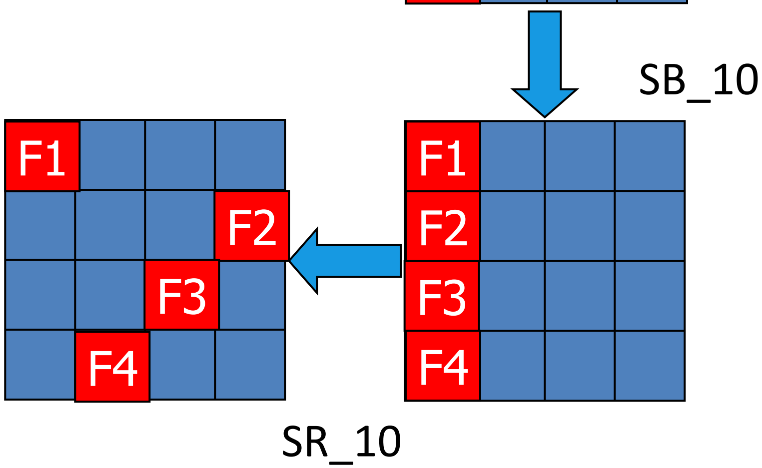


- Sometimes a single fault injection is enough!
  - Recall #2: Bellcore attack

Fault analysis on block ciphers

- DFA – Differential Fault Analysis [BS97]
  - Similar to classical differential cryptanalysis





$$ISB(x_1+K_1)+ISB(x_1+F_1+K_1)=2[ISB(x_2+K_2)+ISB(x_2+F_2+K_2)]$$

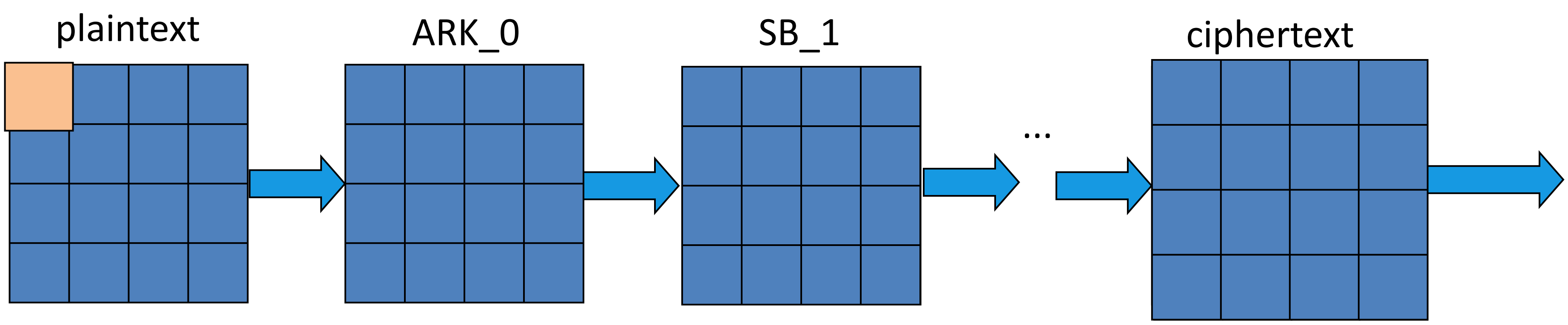
$$ISB(x_2+K_2)+ISB(x_2+F_2+K_2)=ISB(x_3+K_3)+ISB(x_3+F_3+K_3)$$

$$ISB(x_4+K_4)+ISB(x_4+F_4+K_4)=3[ISB(x_2+K_2)+ISB(x_2+F_2+K_2)]$$

- 2/3 faulty encryptions, 4 key bytes, 2^{16} complexity

Fault analysis on block ciphers

- CFA – Collision Fault Analysis [H04]

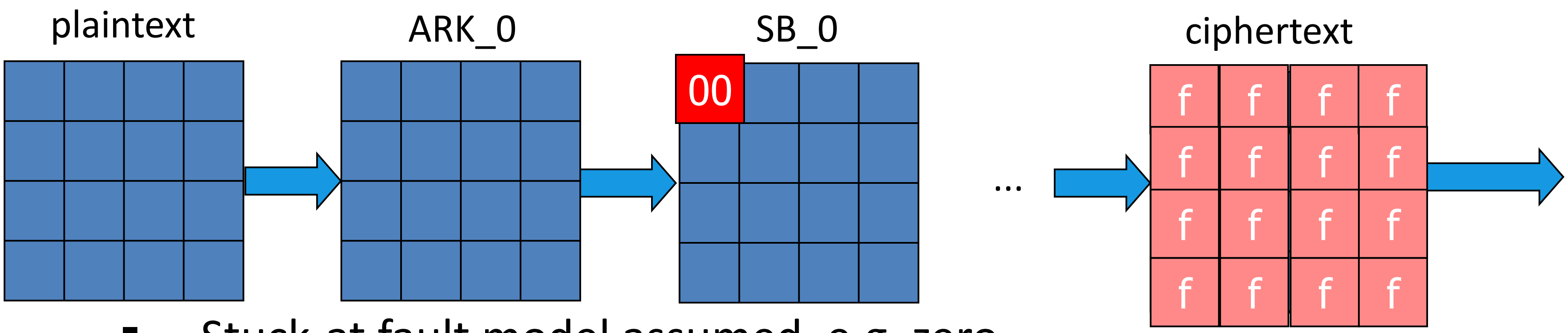


  - Stuck-at fault model assumed, e.g. zero
  - Target operations in first round(s)
  - Attack steps:
    1. Random plaintext, fault @SB_1: ciphertext Ĉ
    2. Random plaintext, no faults: ciphertext C
    3. When Ĉ == C, recover key byte:

$$SB(P1 \text{ xor } K1\_11) = 0x00$$

Fault analysis on block ciphers

- IFA – Ineffective Fault Analysis [BS03] [C07]



  - Stuck-at fault model assumed, e.g. zero
  - Target operations in first round(s)

1. Random plaintext, no faults: ciphertext C
2. Same plaintext, fault @SB_1: ciphertext Ĉ
3. When Ĉ == C, recover key byte:

SB(P1 xor K1_11) = 0x00

  - Differences with CFA:
    - Larger number of faults, not required to know the ciphertext !

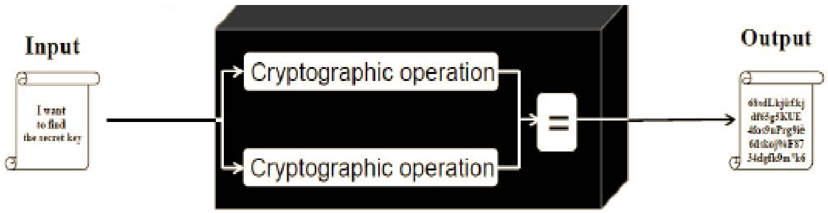
Countermeasures

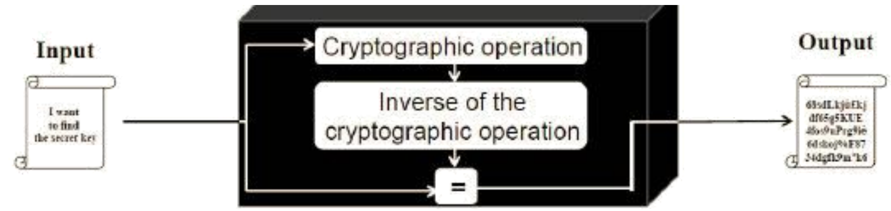
You **cannot** prevent the adversary from trying to mount an attack

- But you can try to make it more difficult !
- Typical countermeasures against fault attacks:
  - *Hardening* hardware:
    - "Hide" sensitive parts of the chip:
      - glue logic, bus scrambling, memory encryption, ...
      - metal layers (passive shielding)
    - Add filters and/or security sensors:
      - power, clock
      - light, temperature, wire mesh (active shielding)

Countermeasures

- *Hardening* computations:
  - Information redundancy
    - Addition of parities, linear codes
    - Ring embeddings
    - Infective computations
  - Hiding countermeasures
  - Branchless implementations
  - Parallel execution or inverse execution





... but second-order fault attacks are possible

Conclusions

- Fault attacks are a very powerful tool
  - Specialized equipment available to wider class of adversaries
- There is no 100% protection
  - With enough resources and time, attacks can be mounted
- Arms-race attacks vs. countermeasures

Bibliography

[BDL97] D. Boneh, R. DeMillo, and R. Lipton, "On the importance of checking cryptographic protocols for faults", CRYPTO, 1997.

[BGV11] J. Balasch, B. Gierlichs, and I. Verbauwhede, "An In-depth and Black-box Characterization of the Effects of Clock Glitches on 8-bit MCUs", FDTC, 2011.

[BGVLV12] J. Balasch, B. Gierlichs, R. Verdult, L. Batina, I. Verbauwhede, "Power Analysis of Atmel CryptoMemory - Recovering Keys from Secure EEPROMs", CT-RSA, 2012.

[BS97] E. Biham and A. Shamir, "Differential Fault Analysis of Secret Key Cryptosystems", CRYPTO, 1997.

[BS03] J. Blömer and J.-P. Seifert, "Fault Based Cryptanalysis of the Advanced Encryption Standard (AES)", FC, 2003.

[C07] C. Clavier, "Secret External Encodings Do Not Prevent Transient Fault Analysis", CHES, 2007.

[CLFT14] F. Courbon, P. Loubet-Moundi, J. Fournier, A. Tria, "Adjusting laser injections for fully controlled faults", COSADE, 2014.

Bibliography

[HS13] M. Hutter, J.-M. Schmidt, “The Temperature Side Channel and Heating Fault Attacks”, CARDIS, 2013.

[HSP08] M. Hutter, J.-M. Schmidt, T.Plos, “RFID and its Vulnerability to Faults”, CHES, 2008.

[H04] L. Hemme, “A Differential Fault Attack Against Early Rounds of (Triple-) DES”, CHES, 2004.

[KQ07] C. H. Kim and J.-J. Quisquater, “Fault attacks for CRT based RSA: new attacks, new results, and new countermeasures”, WISTP, 2007.

[QS03] J.-J. Quisquater and D. Samyde, “Eddy current for Magnetic Analysis with Active Sensor”, *Esmart,* 2002.

[SH08] J.-M. Schmidt and C. Herbst, “A Practical Fault Attack on Square and Multiply”, FDTC, 2008.

[SA02] S. Skorobogatov, R. Anderson, “Optical Fault Induction Attacks”, CHES, 2002.

[WWM11] J.van Woudenberg, M. Witteman and F. Menarini, “Practical optical fault injection on secure microcontrollers”, FDTC, 2011.

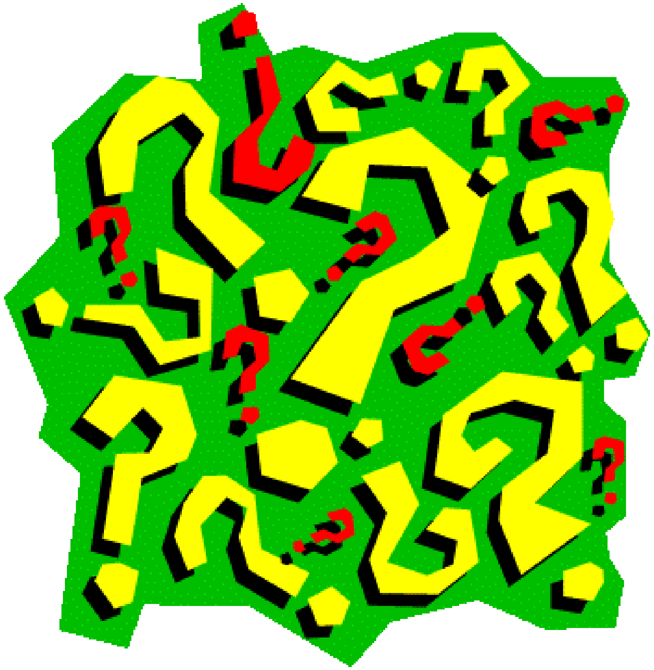
Bibliography

[VKS11] I. Verbauwhede, D. Karaklajić, and J.-M. Schmidt, "The Fault Attack Jungle - A Classification Model to Guide You", FDTC, 2011.

[ZL79] J.F. Ziegler and W.A. Landford, "Effect of cosmic rays on computer memories", Science, 1979.

Thanks for your attention!

QUESTIONS ?



Josep Balasch: josep.balasch@esat.kuleuven.be