Attacks on Smart Cards

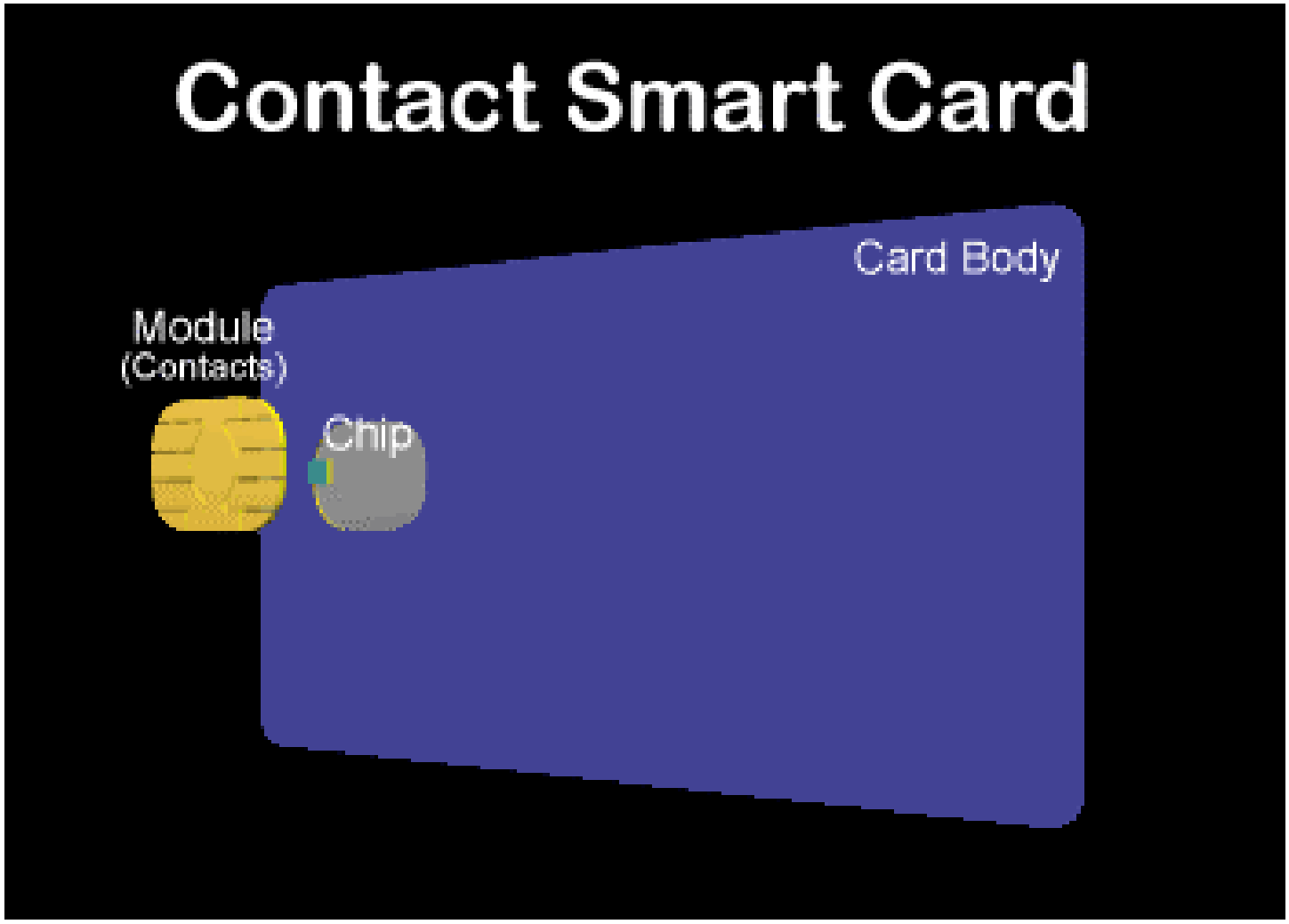
Michael Tunstall
michael.tunstall@gemplus.com

GEMPLUS

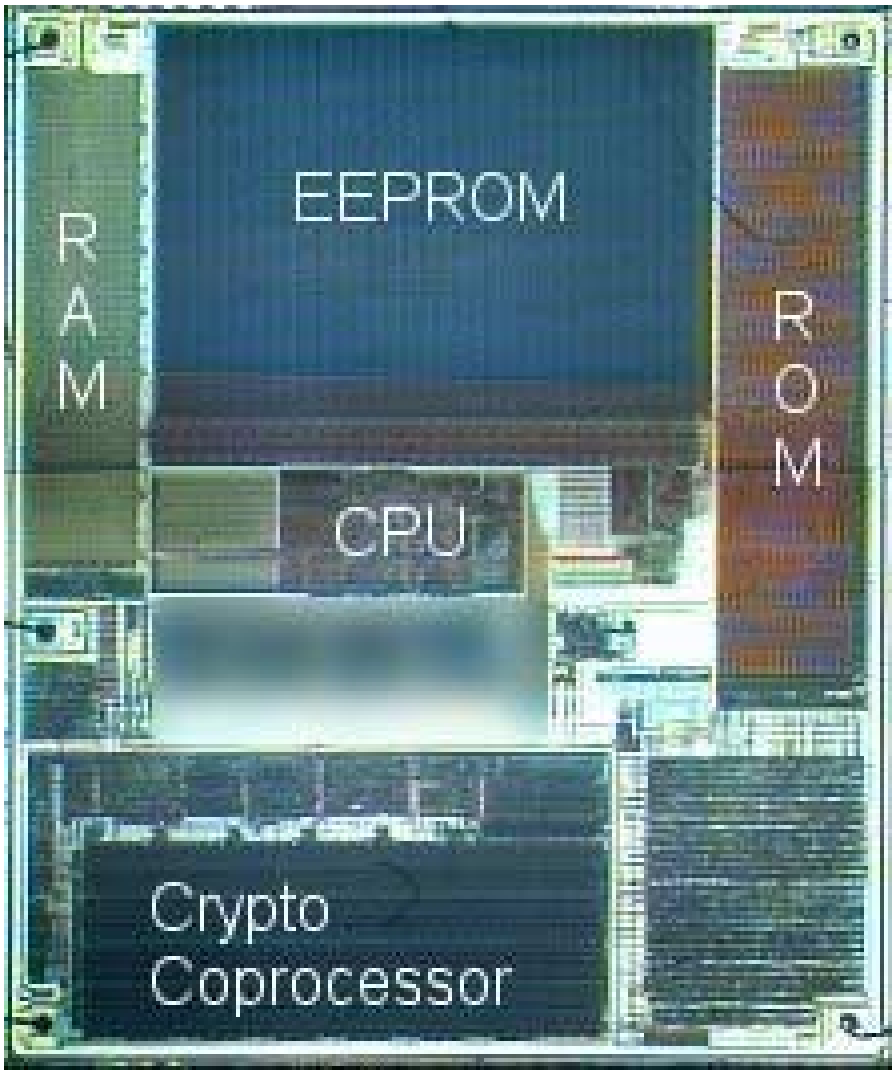
Outline

- **Smart Cards: What and Why**
- **Attacks on cards**
  - **Physical**
  - **Timing, spa, dpa, dfa**
- **Attacks on systems using Smart Cards**
- **Examples**

What is a Smart Card



Chip Structure

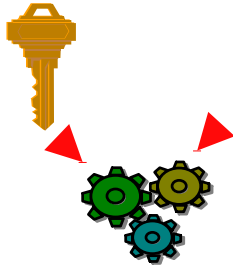


Why Use Smart Cards?

- Tamper resistance
  - Storage

- Portability

- Tamper resistance
  - Processing



- Ease of use
- Onboard key generation
- Cost

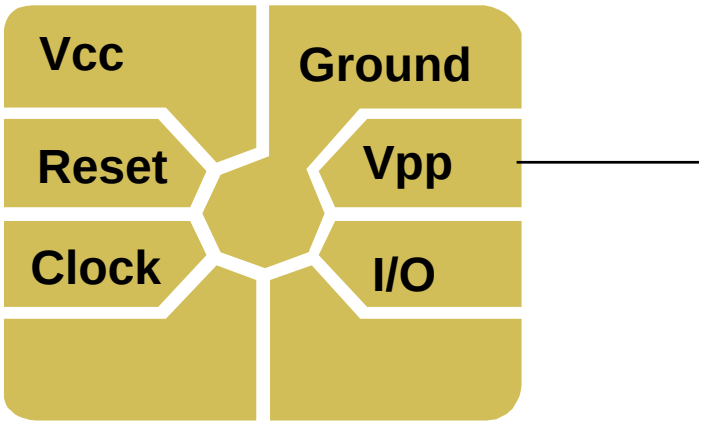
Outline

- **Smart Cards: What and Why**
- **Attacks on cards**
  - **Physical**
  - **Timing, spa, dpa, dfa**
- **Attacks on systems using smart cards**
- **Examples**

Classes of Attacks

- **Physical**
- **Side-channel**
- **Software**
- **Environment**

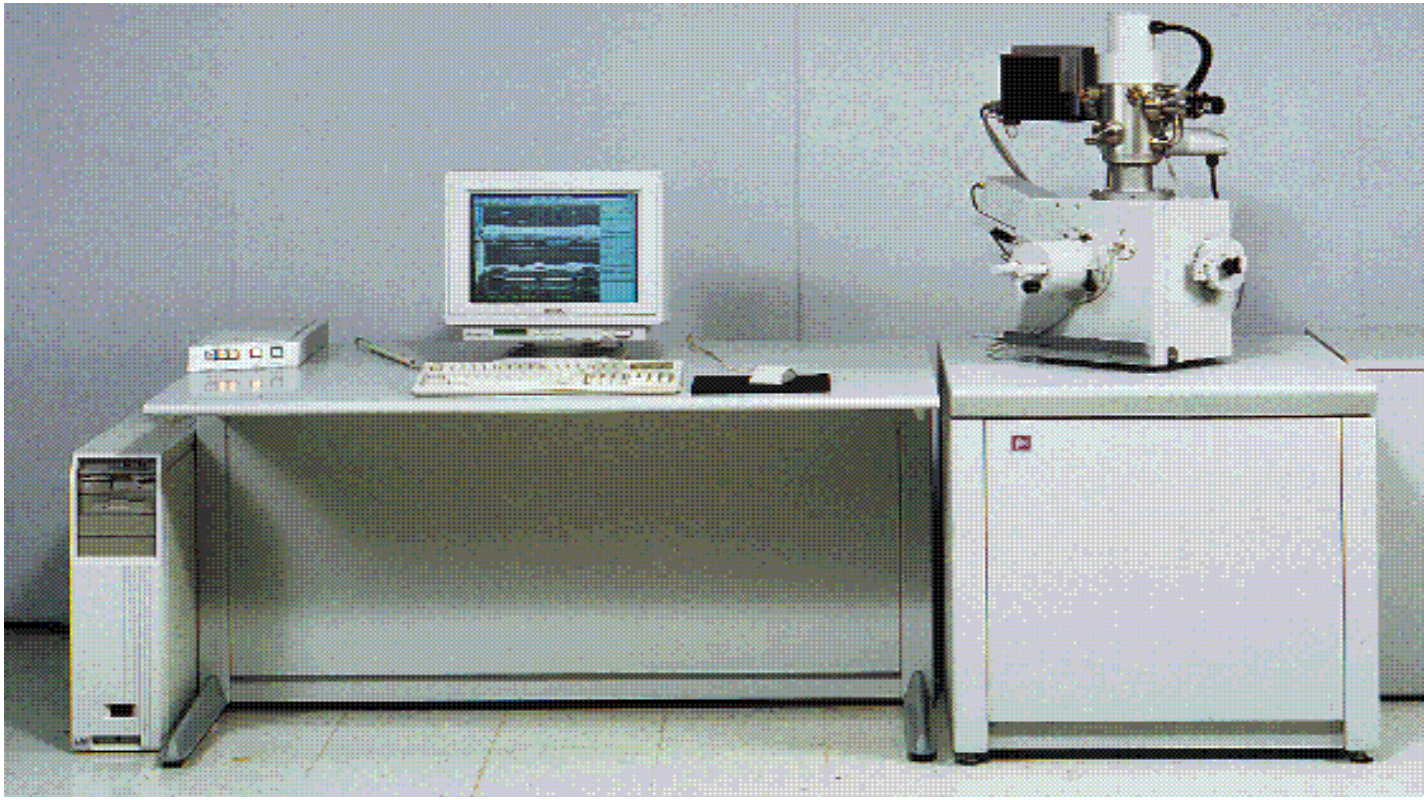
Power Outage



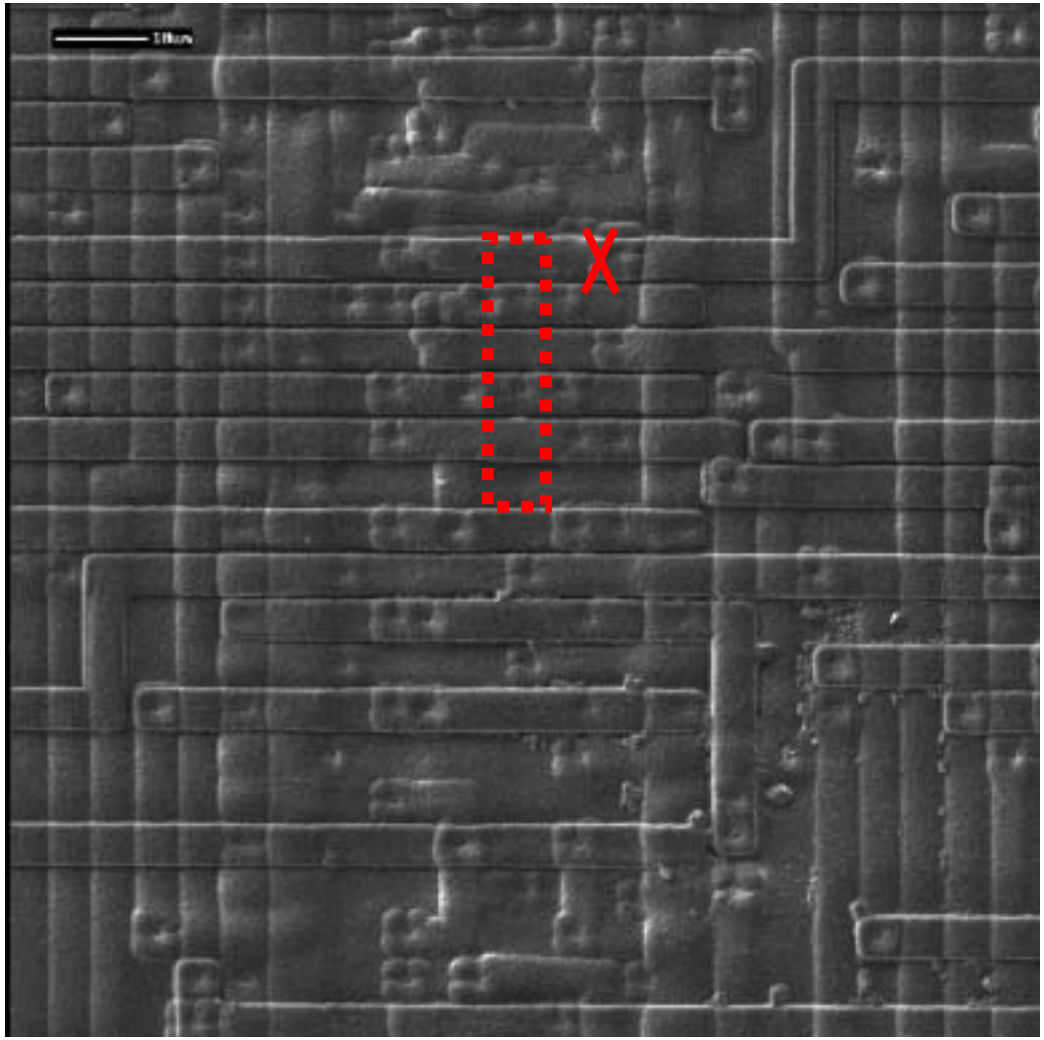
- Attack on VPP
- Using nail polish
- Card not debited...

Probe Stations, F.I.B.

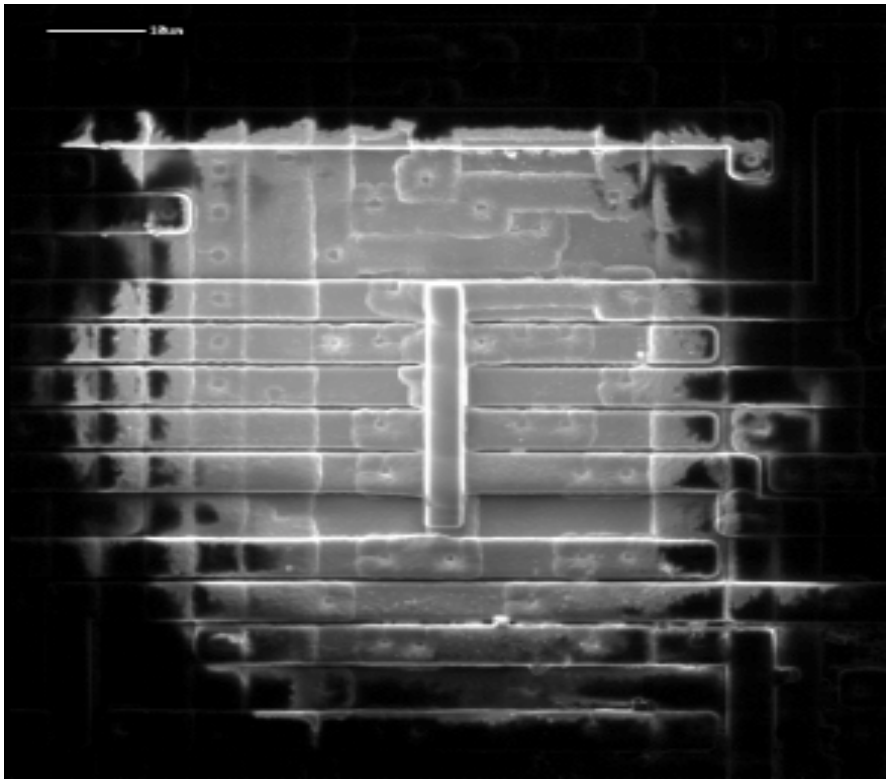
- **If you have more money or if you are a student.**



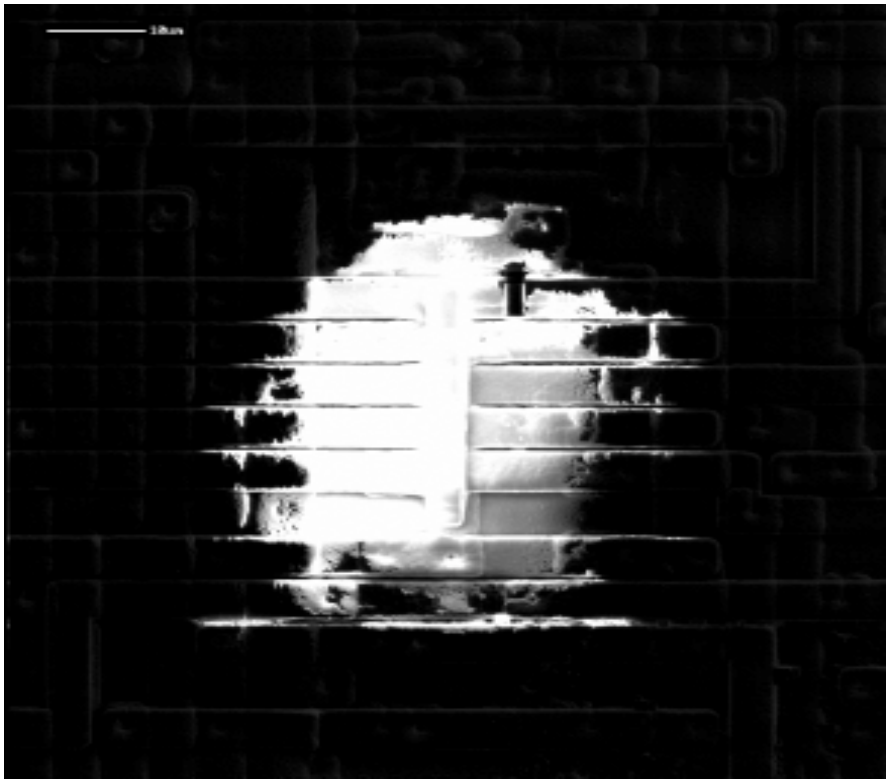
Chip Re-Wiring



Addition of a Track

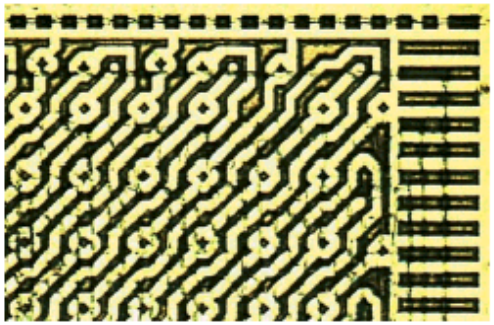


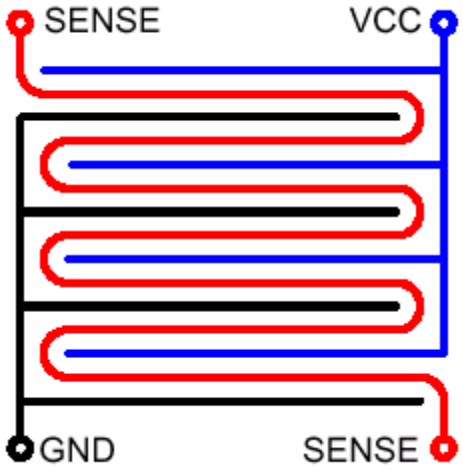
Cutting of a Track



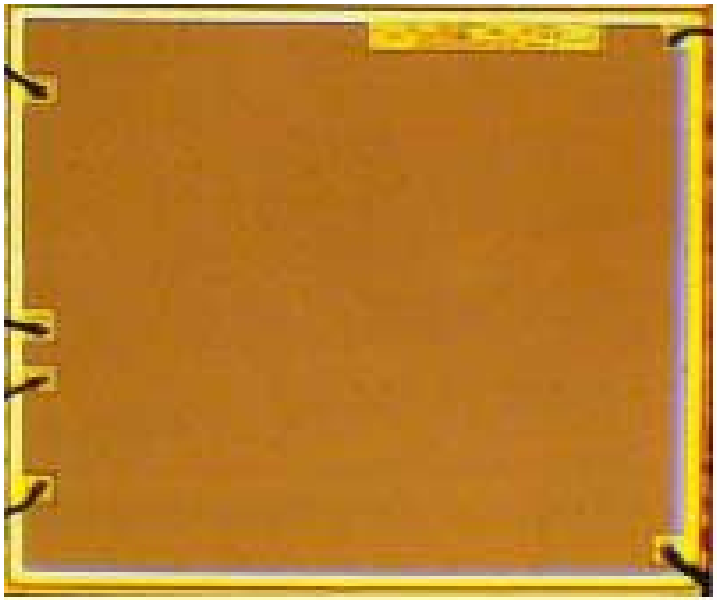
Countermeasures

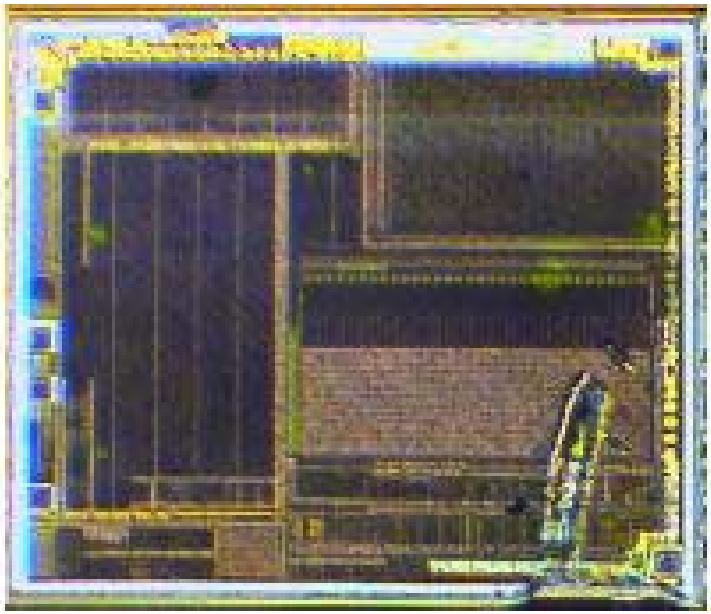
- Metal layers
- Bus scrambling
- Onboard sensors
  - Temperature, light, frequency
  - Integrity :





Chip Structure





Physical Attacks Summary

- **Difficult to defeat completely**
- **Expensive**
- **Destructive**
- **Target dependant**
- **Time consuming**

Classes of Attacks

- **Physical**
- **Side-channel**
- **Software**
- **Environment**

Side Channel Attacks

- **Exploit information on secret data leaked byt the card.**
  - **Time (Timing Attacks)**
  - **Power (SPA, DPA)**
  - **Radiation (Electromagnetic SPA/DPA)**

Timing Attacks

- **You put $28 in one of the pots and $10 in the other:**

- **Question: Compute**
  - **Blue * 10 + Red *7**
  - **Tell me if the result is odd or even.**

- **Is your answer enough to reveal what's in each pot?**

Timing Attacks

- **Well, normally not :**

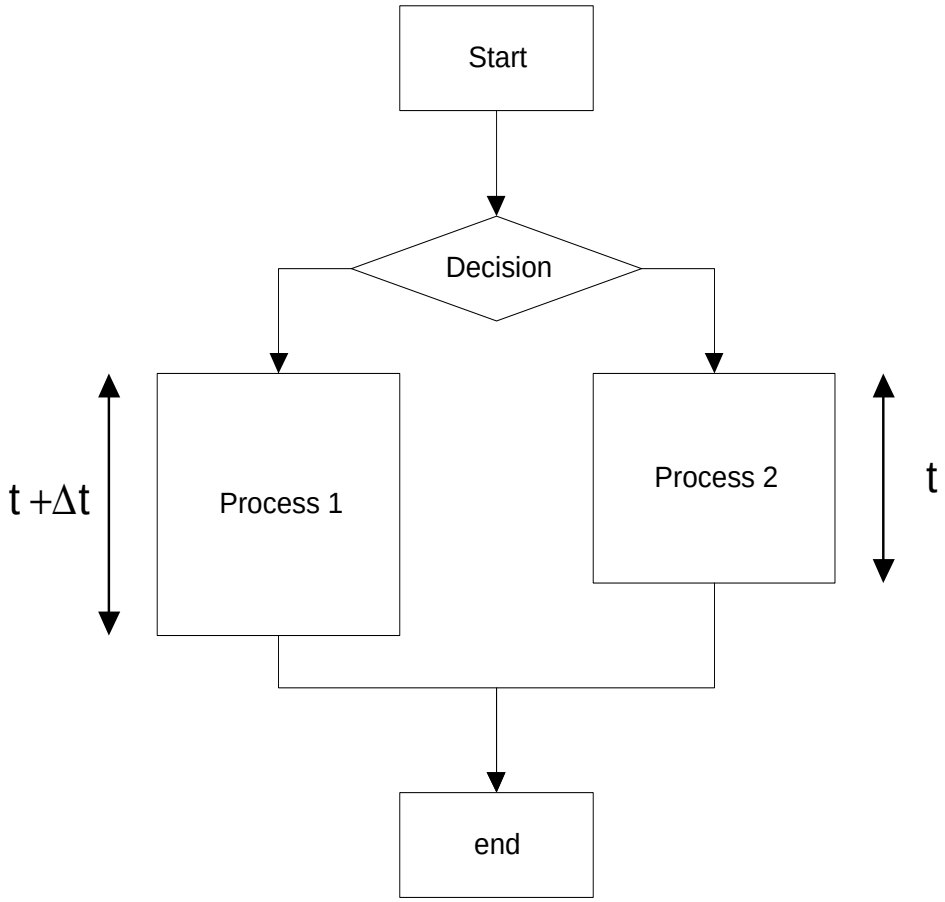
  28* 7 + 10 * 10 = 296 **is an even number**

  and

  10 * 7 + 28 * 10 = 350 **is also even...**

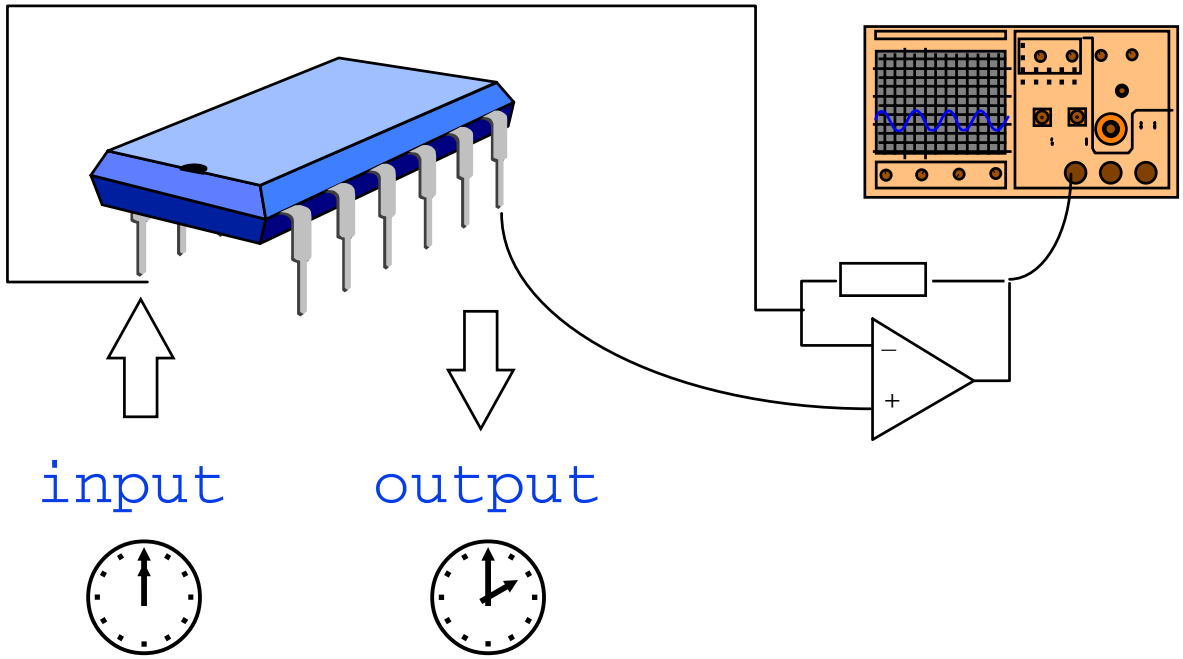
- **However, just by monitoring the time it takes to give the answer one can tell where each amount is!**

Timing Attack on a Smart Card



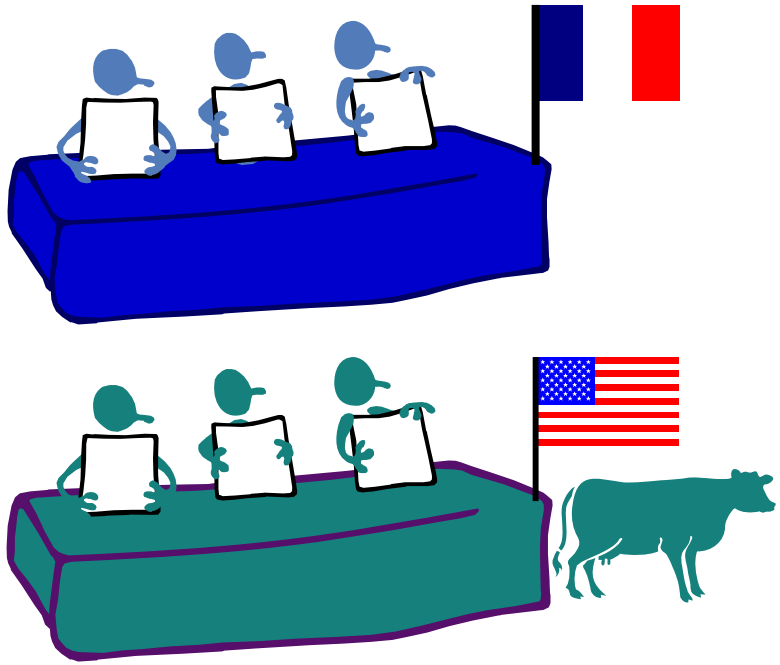
Power Attacks

- **Measure the circuit's processing time and current consumption to infer what is going on inside it.**



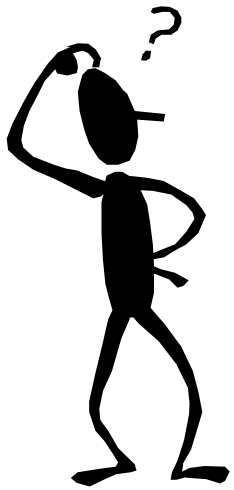
Power Attacks

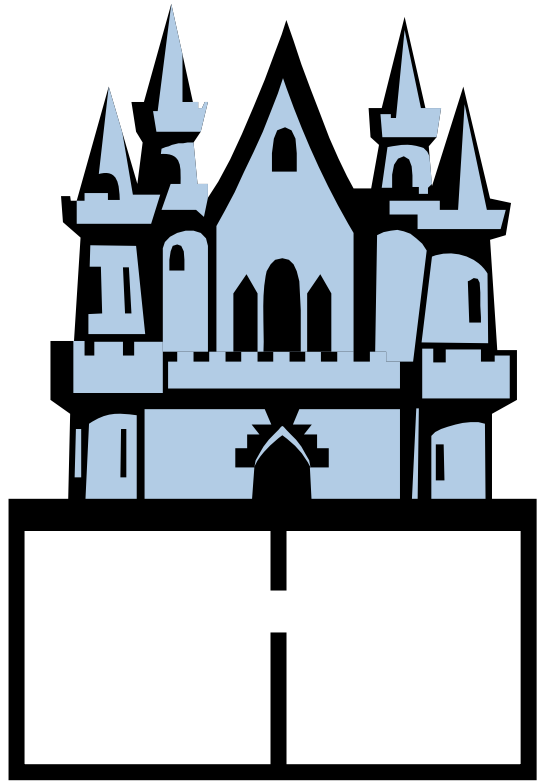
- **Seattle, 1999.**
- **US and French delegates negotiate under which conditions beef could be imported to France. «The Sun» sends a journalist to investigate:**



Power Attacks

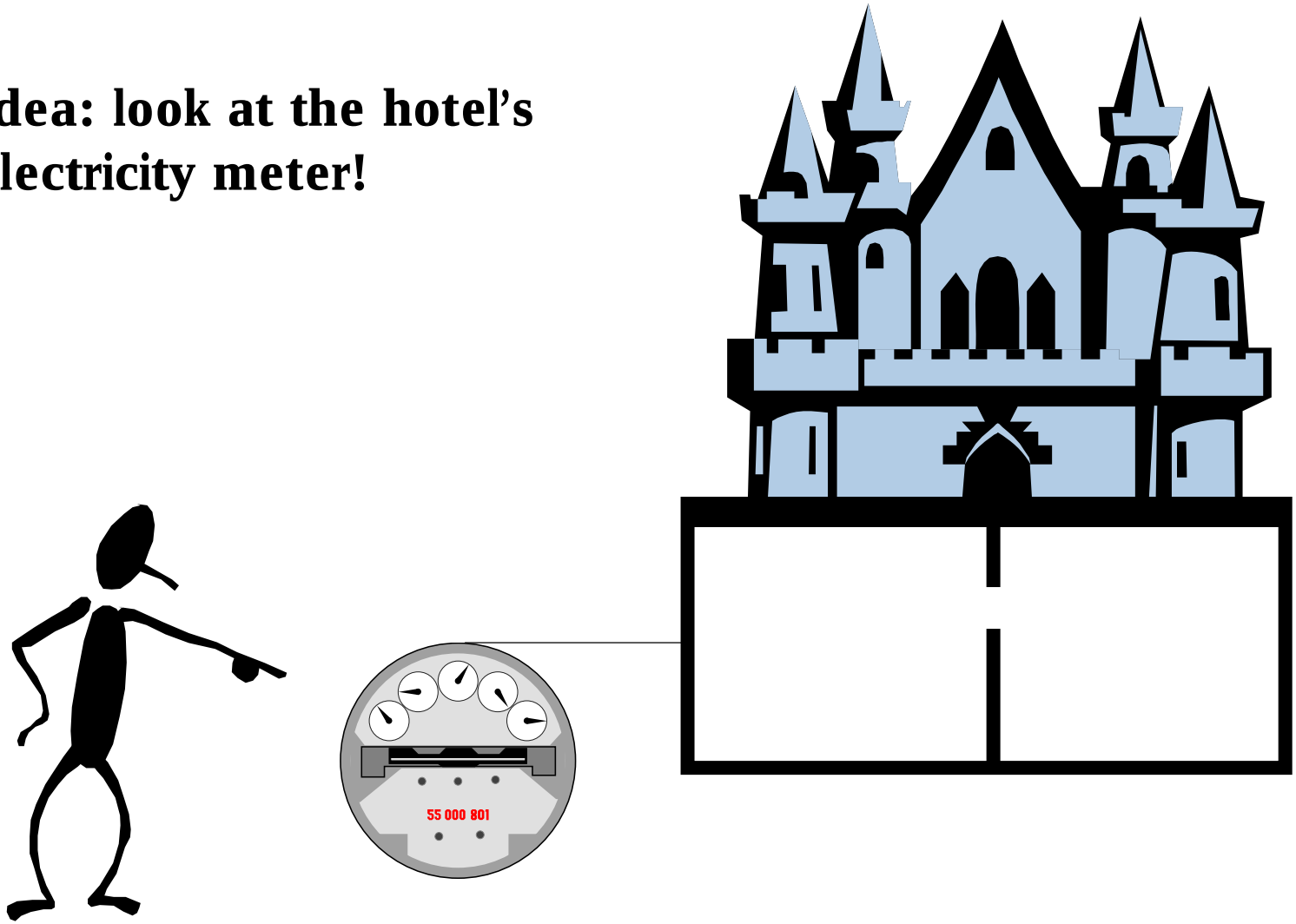
- **But there is a technical problem: negotiations take place in a hotel which windows are opaque.**





Power Attacks

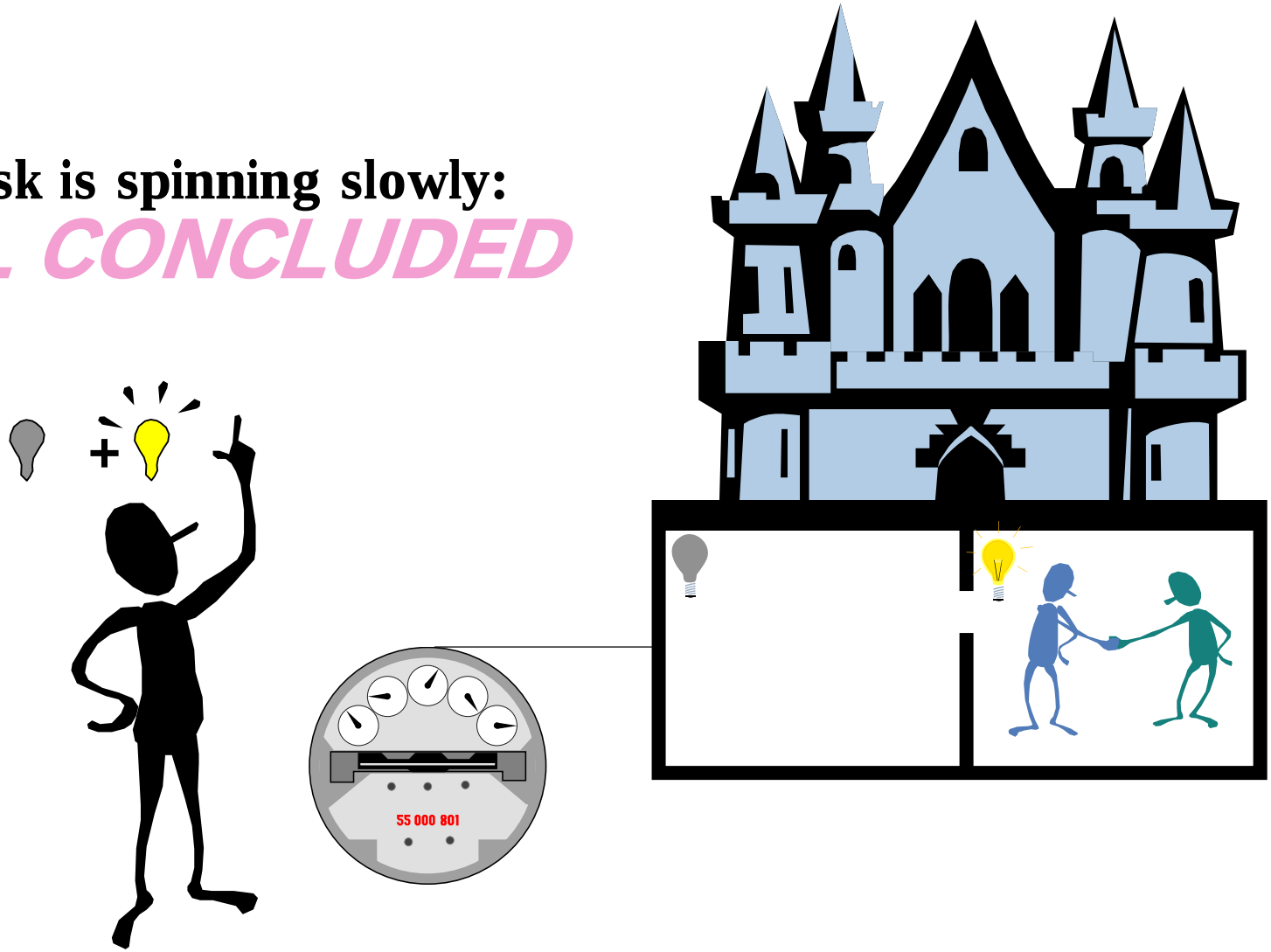
- **Idea: look at the hotel's electricity meter!**



Power Attacks

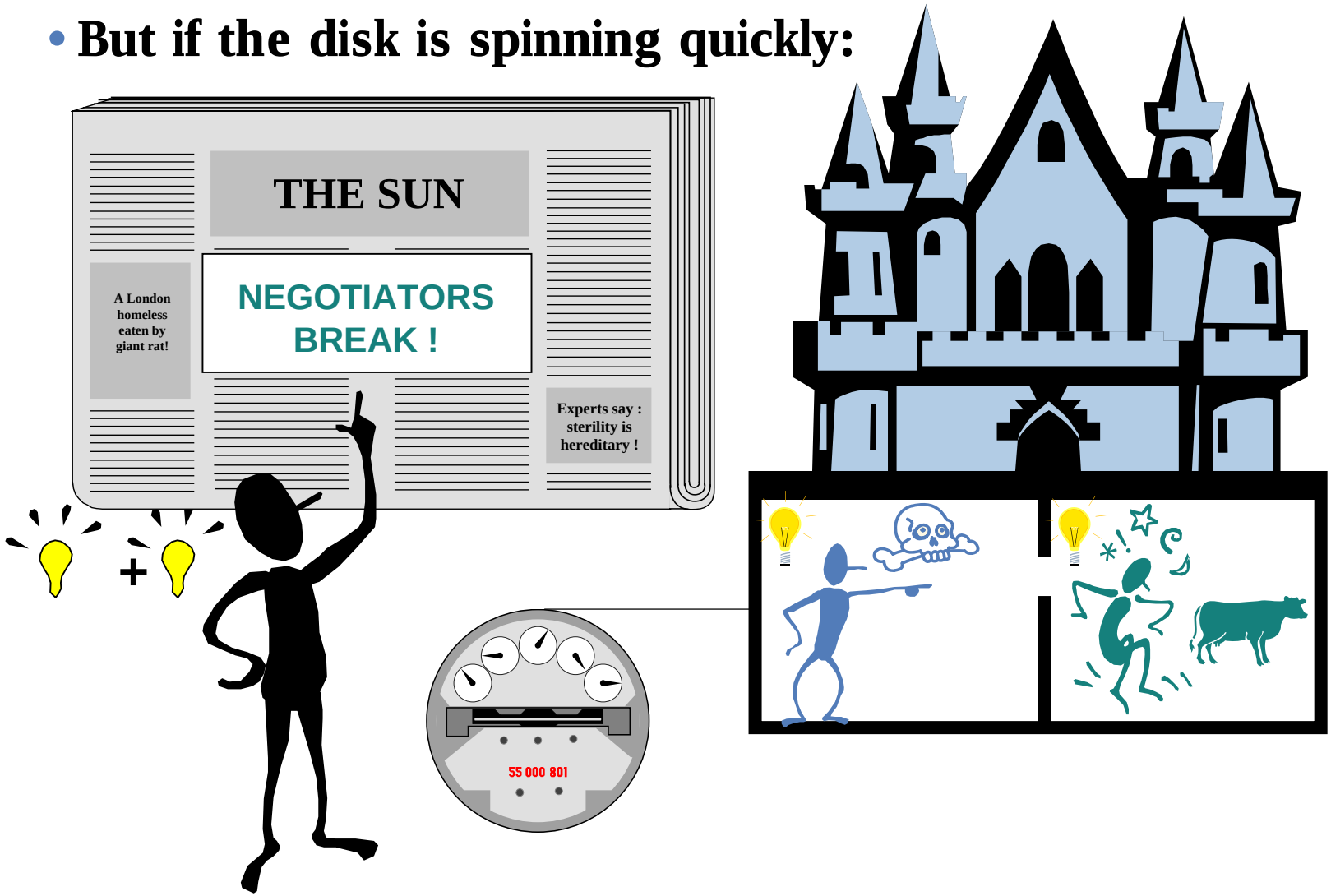
- Disk is spinning slowly:

DEAL CONCLUDED



Power Attacks

- **But if the disk is spinning quickly:**



SPA attack on RSA

- **SPA against RSA private exponentiation**

$$s = m^d \bmod n$$

  - **n large modulus, say 1024 bits**
  - **m message**
  - **s signature**
  - **d private exponent**

- **The attacker aims at retrieving d**

SPA attack on RSA

- **Implementation (assumed known hereafter)**
  - **basic "square and multiply" algorithm**
  - **exponent bits scanned from MSB to LSB (left to right)**

```
Let k = bitsize of d

Let s = m

For i = k-2 down to 0

  Let s = s*s mod n (SQUARE)

  If (bit i of d) is 1 then
   Let s = s*m mod n (MULTIPLY)
  End if

End for
```

Example : $s = m^9 = m^{1001b}$

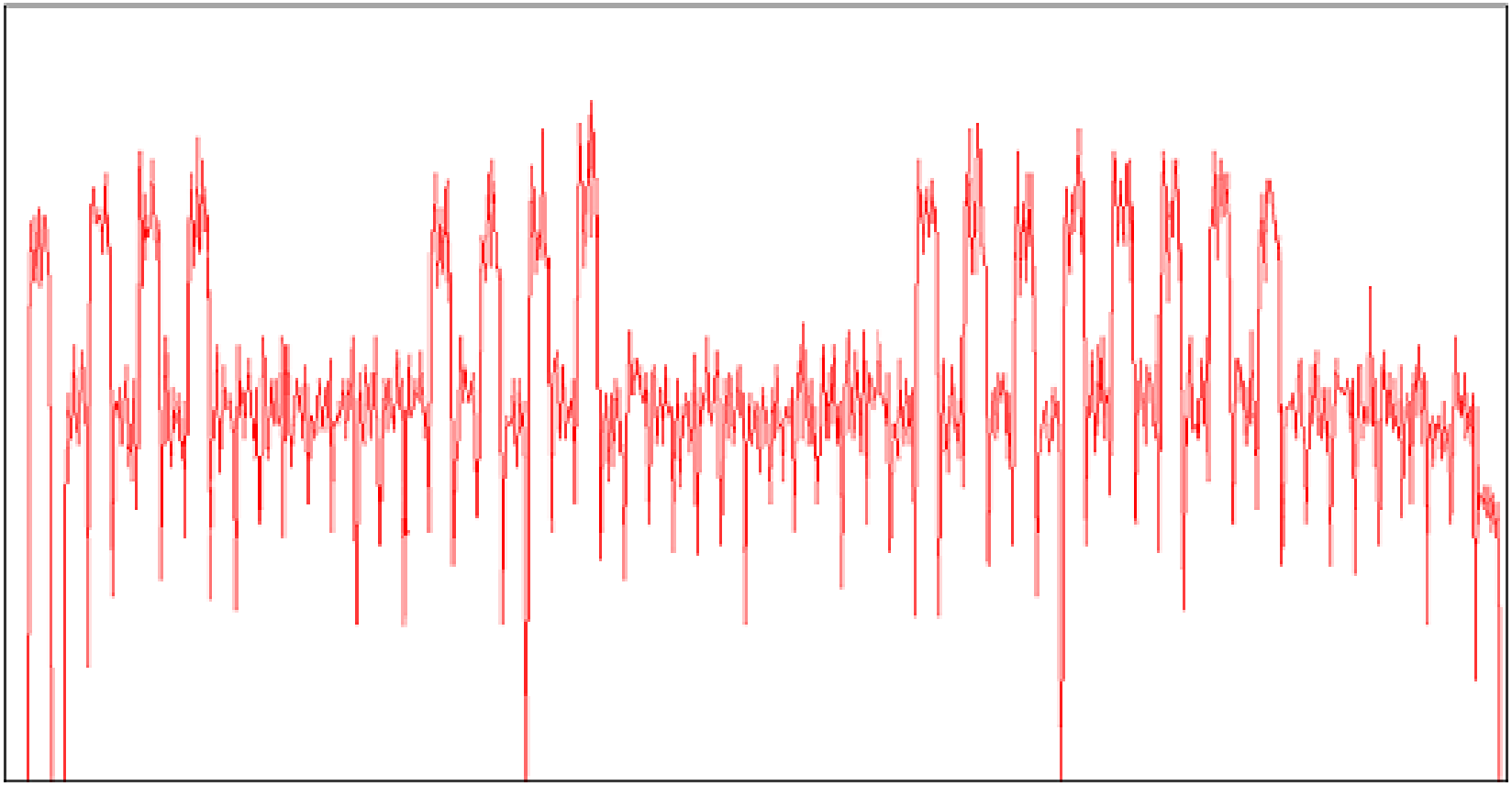
init (MSB 1) $s = m$

round 2 (bit 0) $s = m^2$

round 1 (bit 0) $s = (m^2)^2 = m^4$

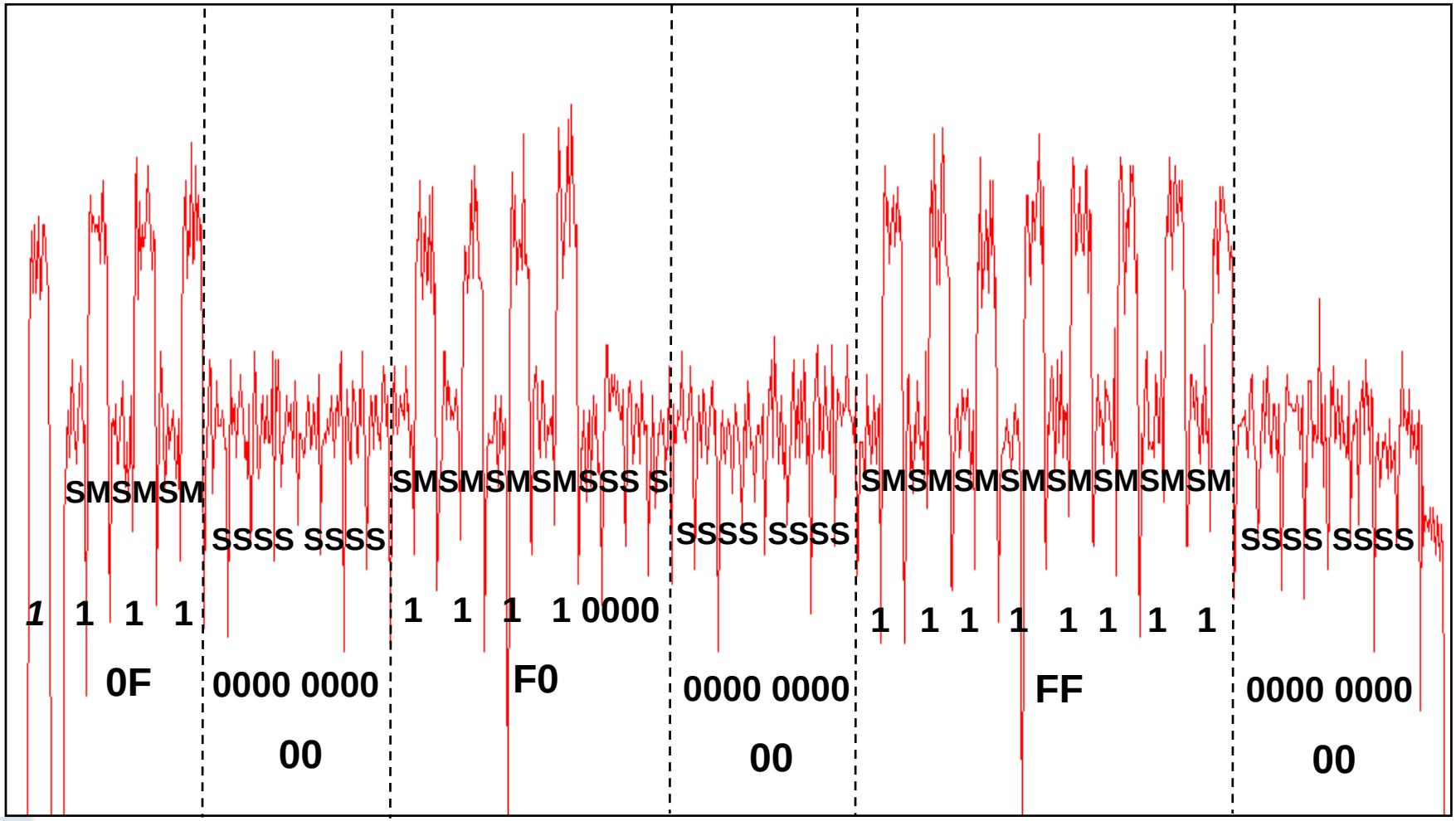
round 0 (bit 1) $s = (m^4)^2 * m = m^9$

SPA attack on RSA



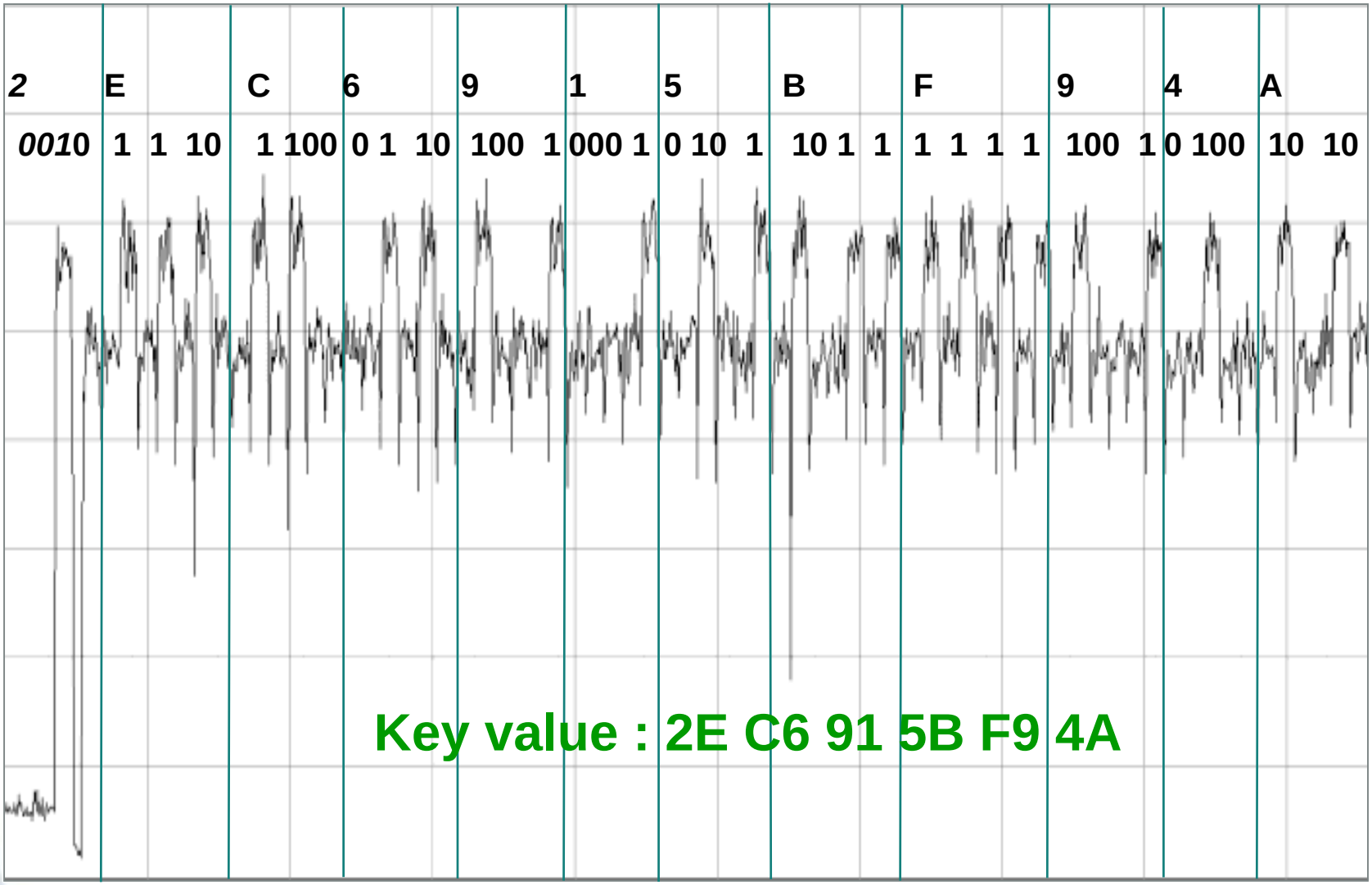
SPA attack on RSA

Test key value : 0F 00 F0 00 FF 00



GEMPLUS

SPA attack on RSA



GEMPLUS

Randomising RSA

- RSA signature: $y = x^{d} \bmod N$

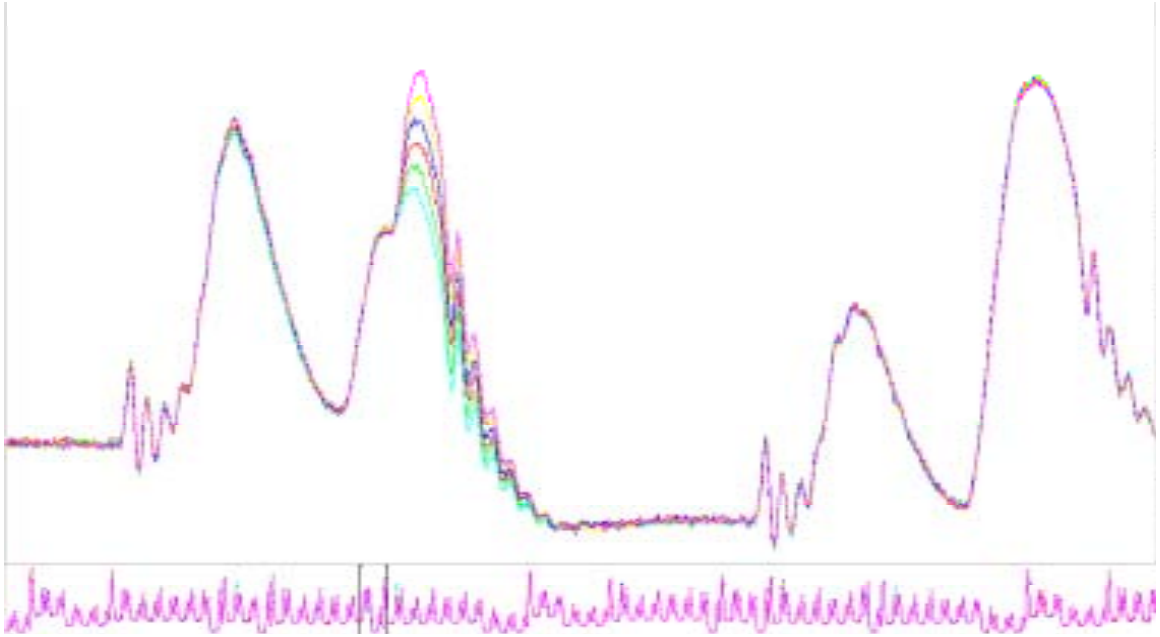
1. choose 3 randoms r_1, r_2 and r_3
2. set $x' = x + r_1 N$
3. set $d' = d + r_2\,\phi(N)$
4. set $N' = r_3\,N$
5. compute $y' = (x')^{d'} \bmod N'$
6. return $y = y' \bmod N$

Summary on SPA

- SPA uses implementation related patterns
- SPA strategy
  - algorithm knowledge
  - reverse engineering phase (signature location)
  - representation tuning (height of view, zoom, visualisation)
  - playing with implementation assumptions…
- SPA is always specific due to
  - the algorithm implementation
  - the applicative constraints
  - the chip's technology (electrical properties)
  - possible counter-measures…

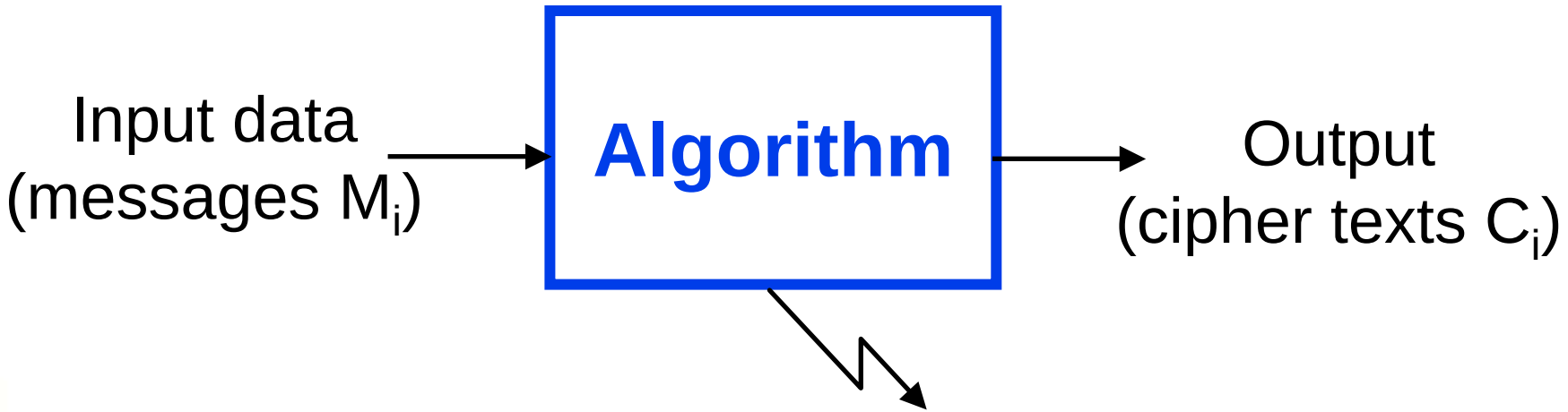
DPA

- **Based on SPA**
  Adding the power of statistics to separate signal from noise



DPA Hypothesis

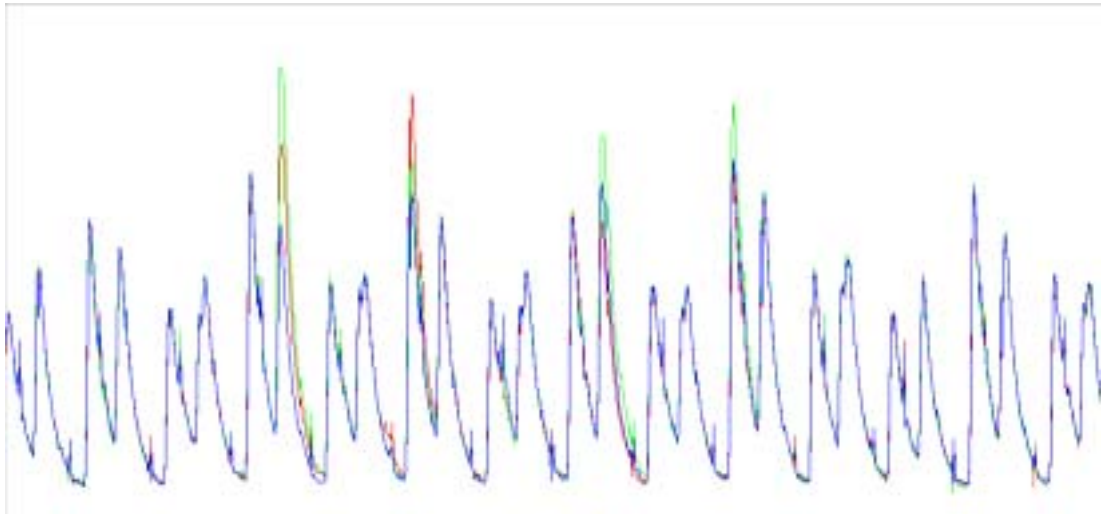
Play the algorithm N times
(100 < N < 100000)



Acquisition procedure

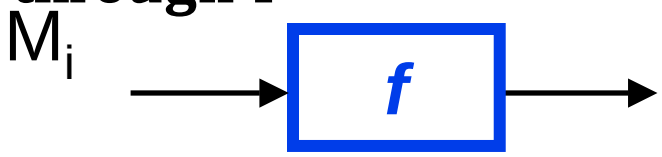
- **After data collection, what is available ?**
  - **N plain and/or cipher random texts**

    00 B688EE57BB63E03E
    01 185D04D77509F36F
    02 C031A0392DC881E6 …
  - **N corresponding power consumption waveforms**



Selection & Prediction

- Assume the data are processed by a known deterministic function f (transfer, permutation…)
- Knowing the data, one can re-compute off line its image through f

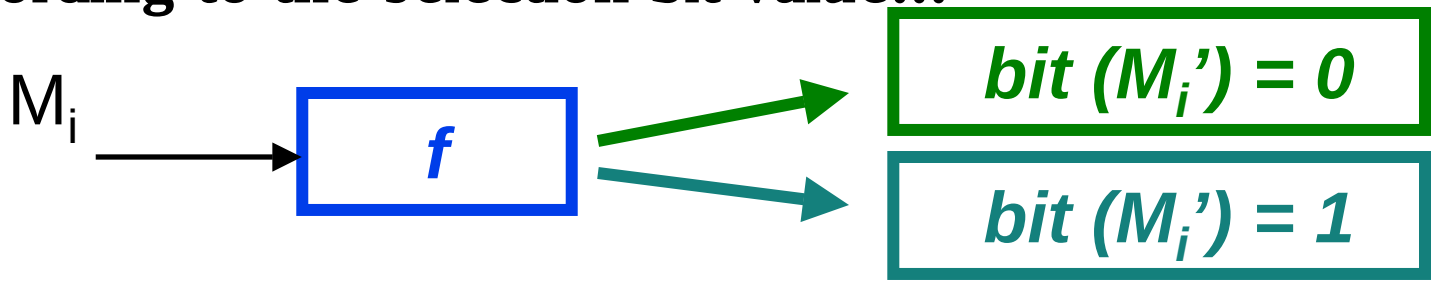


- Now select a single bit among M' bits (in M' buffer)
- One can predict the true story of its variations

i	Message	bit	
0	B688EE57BB63E03E	1	
1	185D04D77509F36F	0	
2	C031A0392DC881E6	1	….

DPA Operator

- **Partition the data and related curves into two packs, according to the selection bit value…**



- **… and assign -1 to pack 0 and +1 to pack 1**

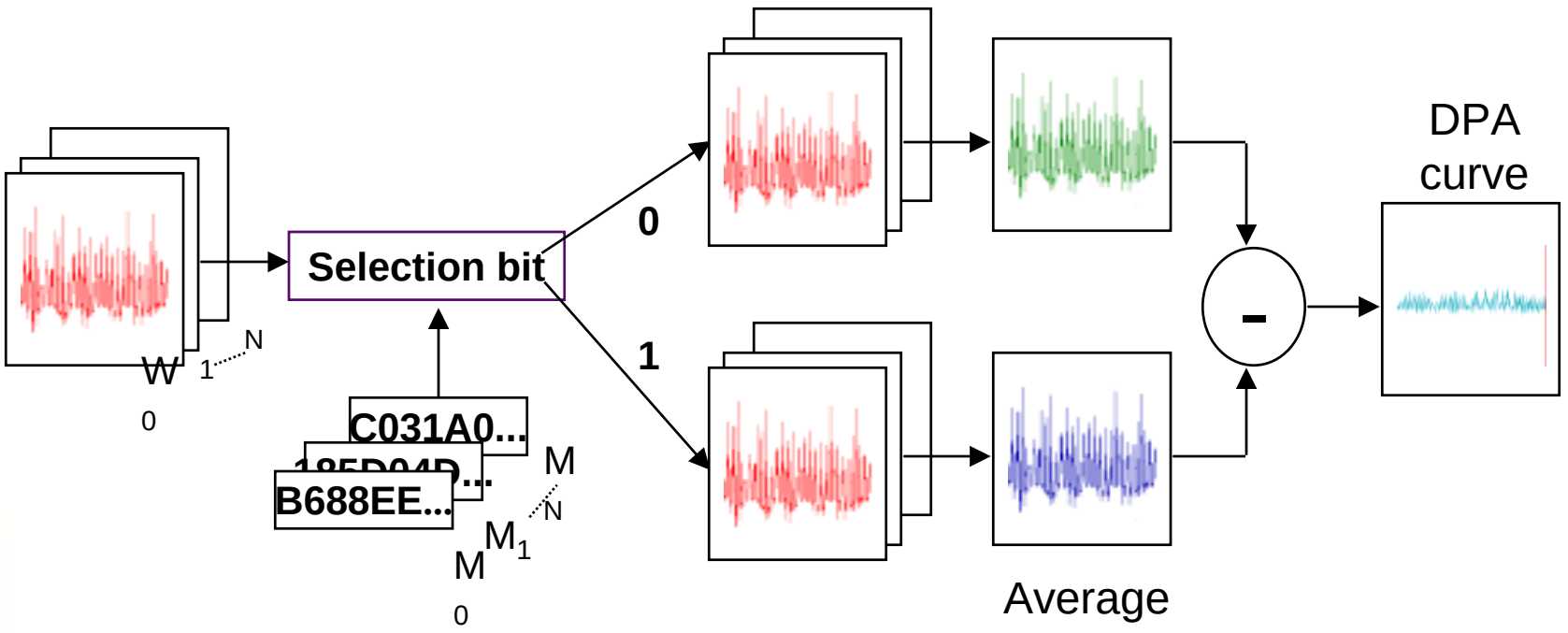
0	B688EE57BB63E03E	1	+1	
1	185D04D77509F36F	0	-1	
2	C031A0392DC881E6	1	+1	…

- **Sum the signed consumption curves and normalise**
- **<=> Difference of averages**

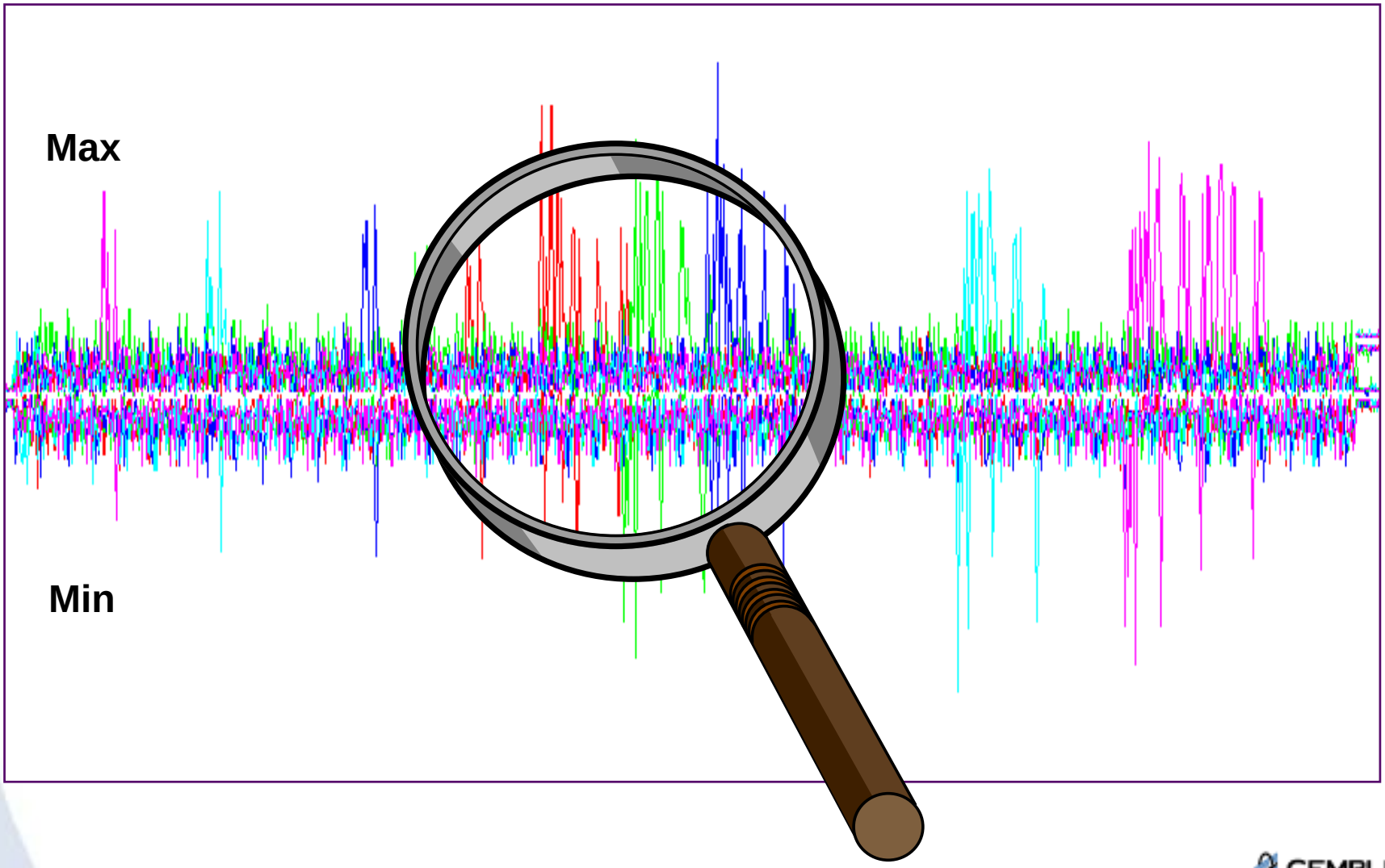
  ($N_0 + N_1 = N$)

$$DPA = \frac{\sum W_1}{N_1} - \frac{\sum W_0}{N_0}$$

DPA Curve Construction

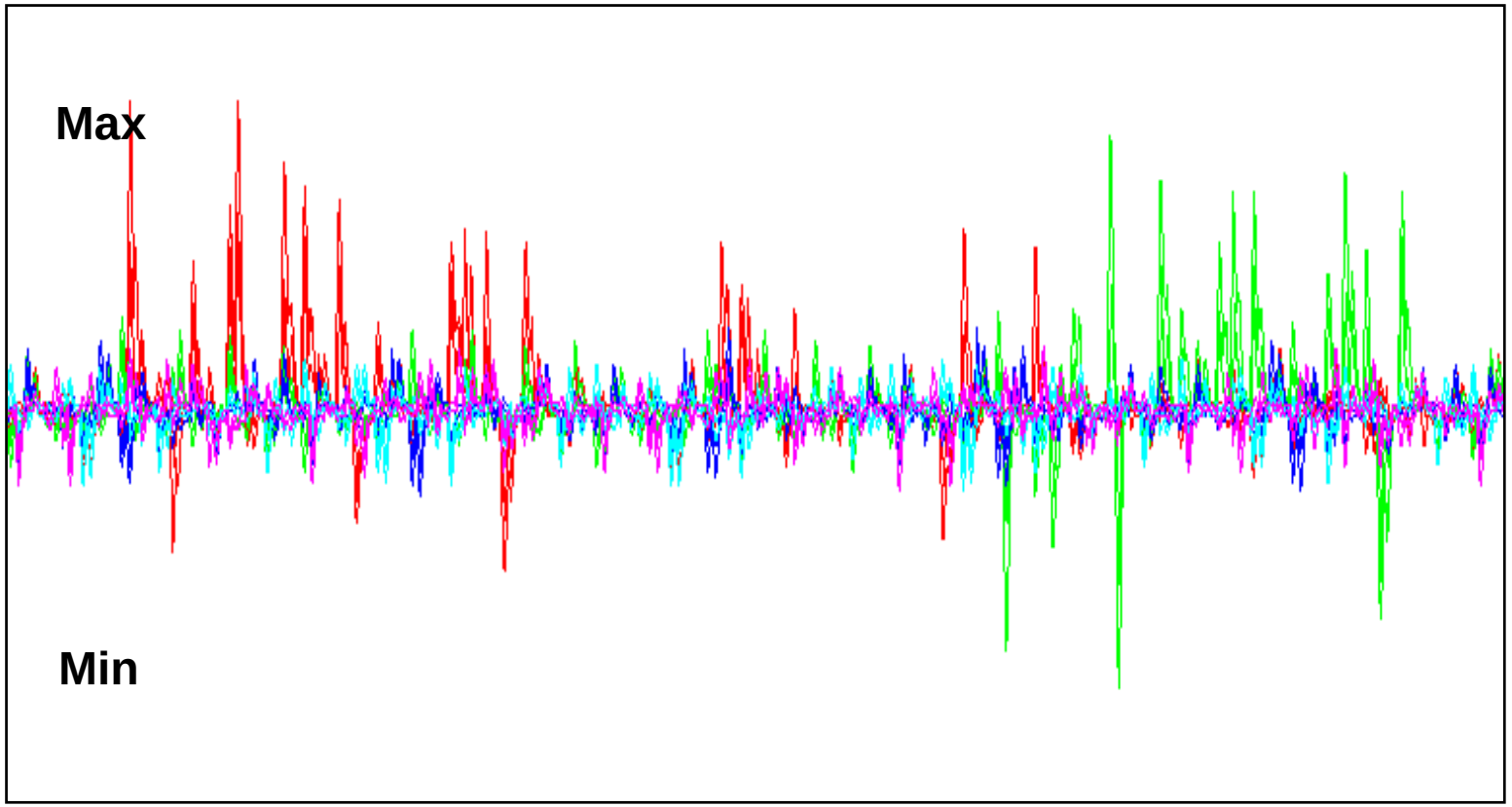


DPA Curve Example



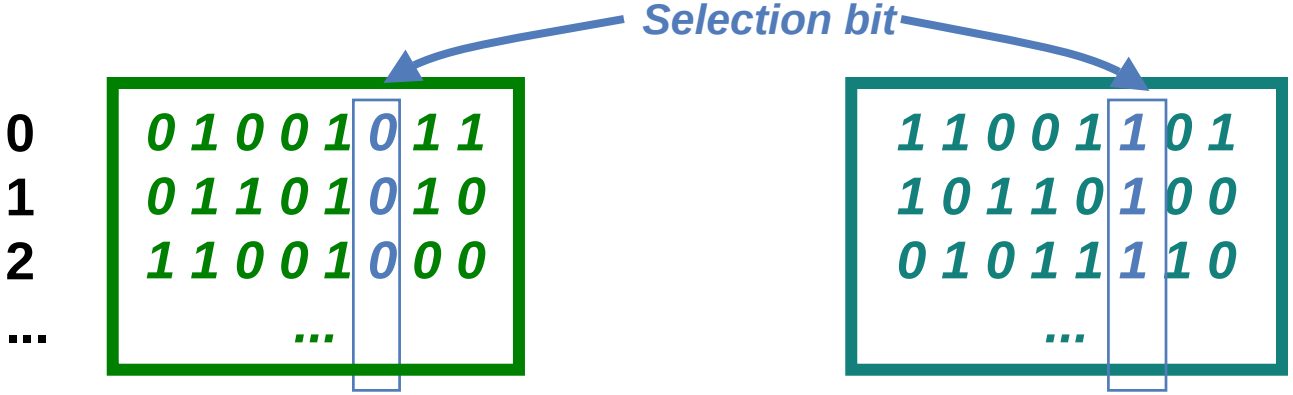
DPA Curve Example

- **Peaks are present when selection bits are handled**



DPA operator & curve

- **Spikes explanation : Hamming Weight of the bit's byte**



Average = E [HW_0] = 0 + 3.5

Average = E [HW_1] = 1 + 3.5

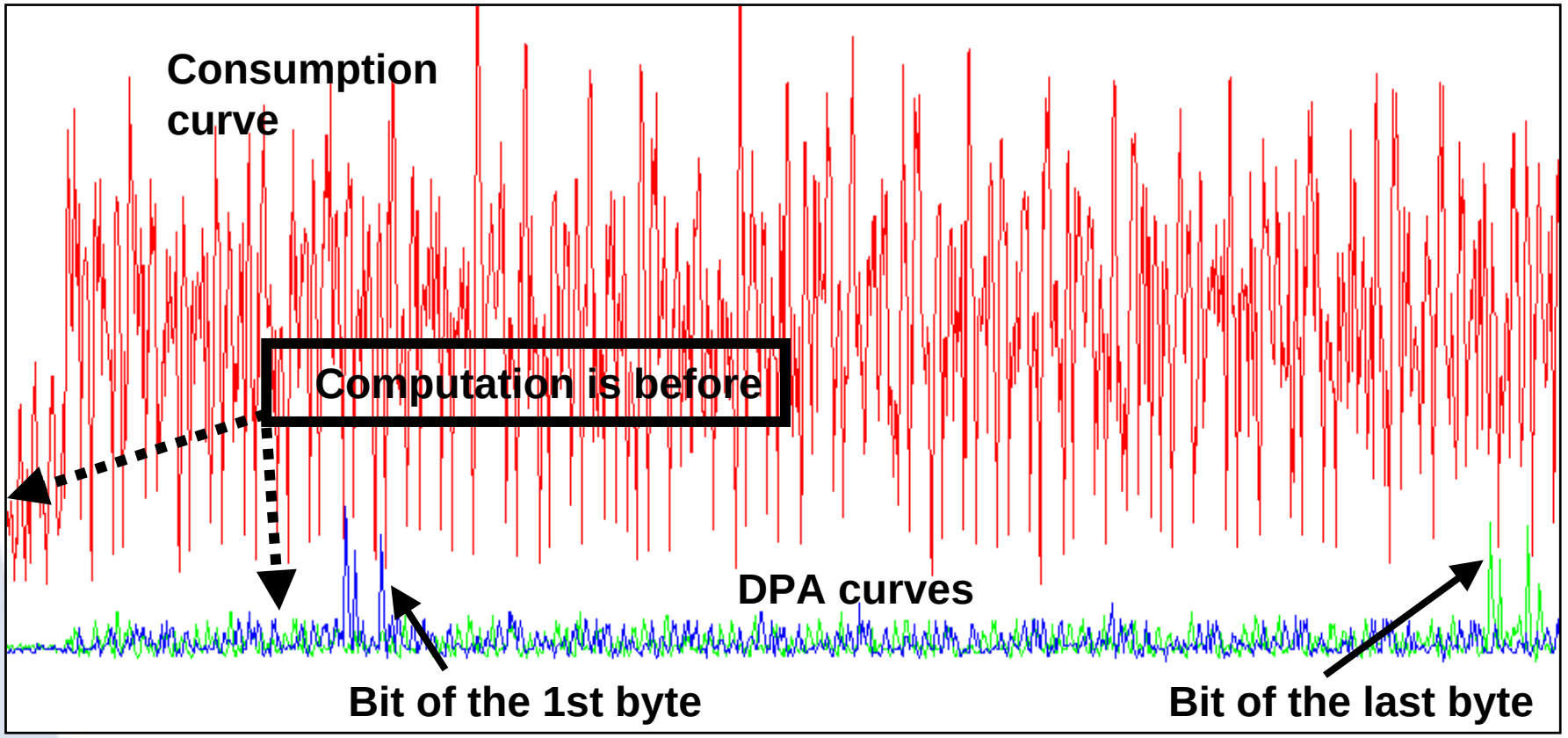
$$\Delta = E[HW_1] - E[HW_0] = 1$$

- **Contrast (peak height) proportional to $N^{1/2}$ (evaluation criterion)**
- **If prediction was wrong : selection bit would be random**

$$E[HW0] = E[HW1] = 4 \Rightarrow \quad \Delta = 0$$

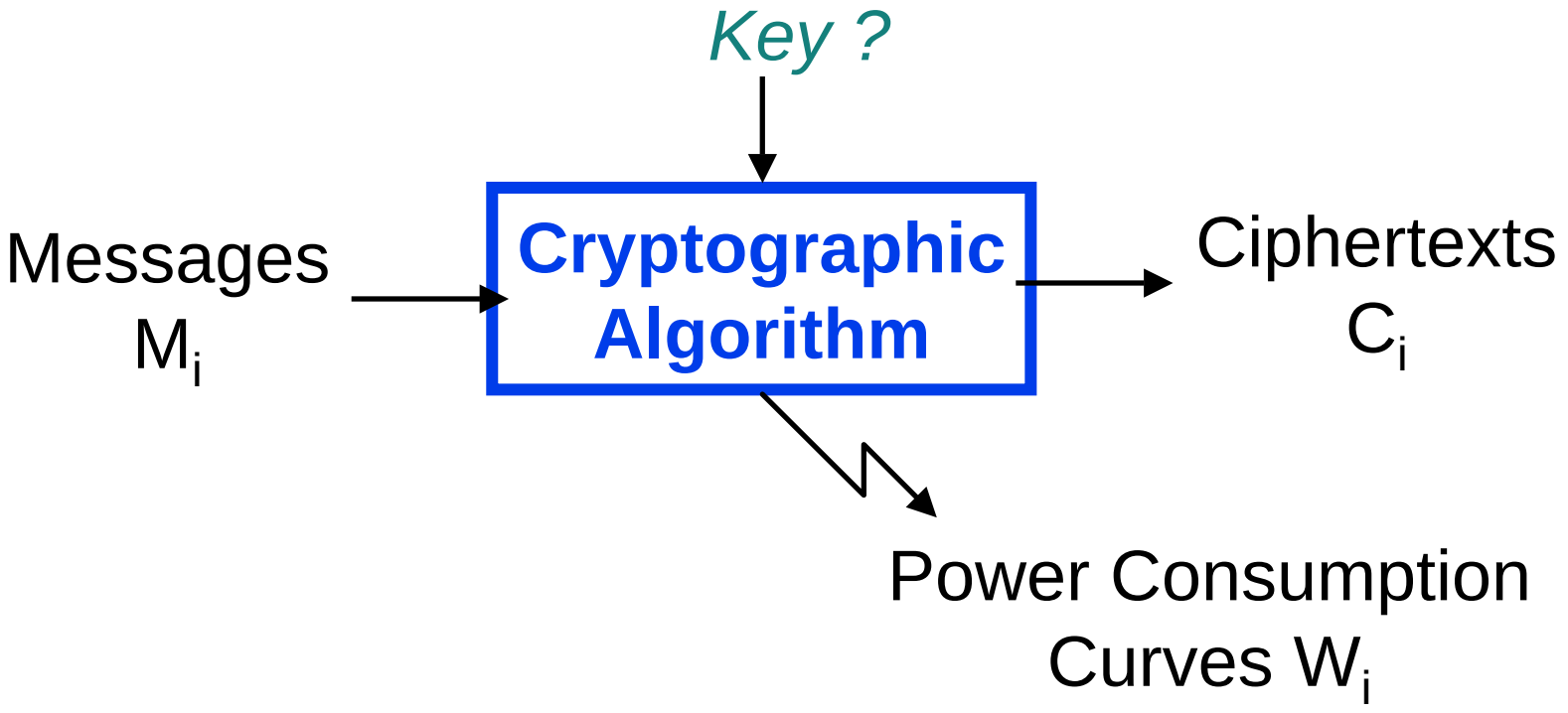
Reverse engineering using DPA

- Use DPA to locate when predictable things occur
- Example : locate an algo trace by targeting its output (ciphertext transfer to RAM, ciphertext is given)



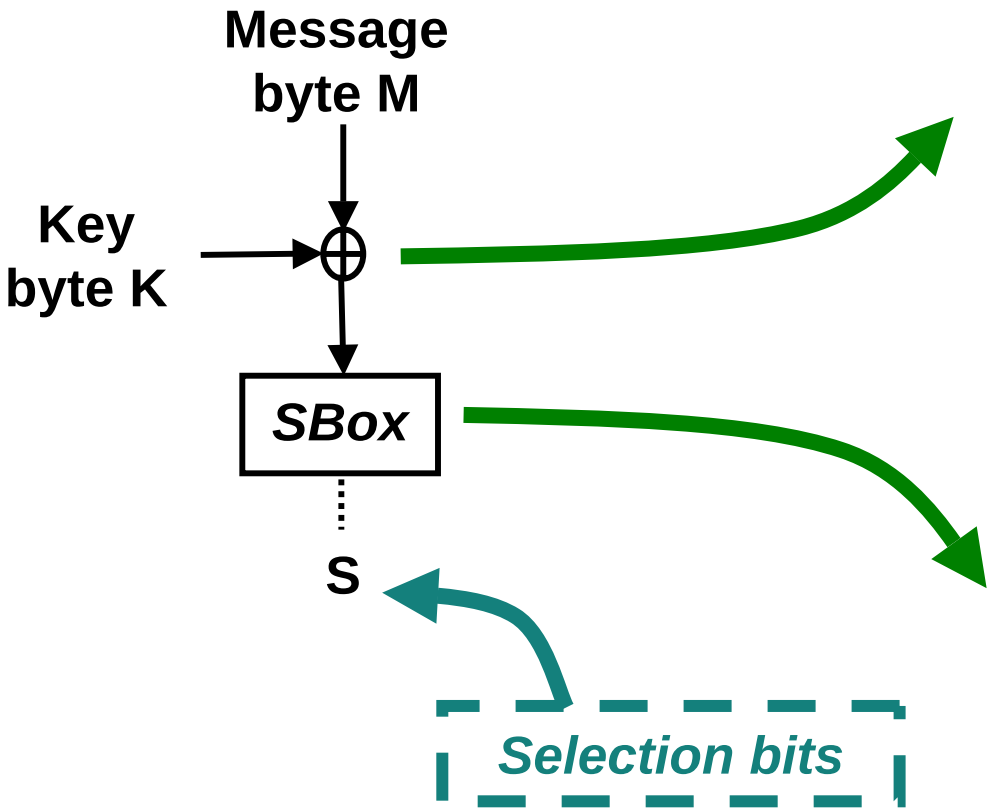
Attacking a Secret Key Algorithm

- **DPA works thanks to the perfect prediction of the selection bit**
- **How to break a key ?**



DPA: typical target

- **Basic mechanism in Secret Key algorithms (AES, DES…)**



Exclusive OR

Non-linear substitution

by table look-up :

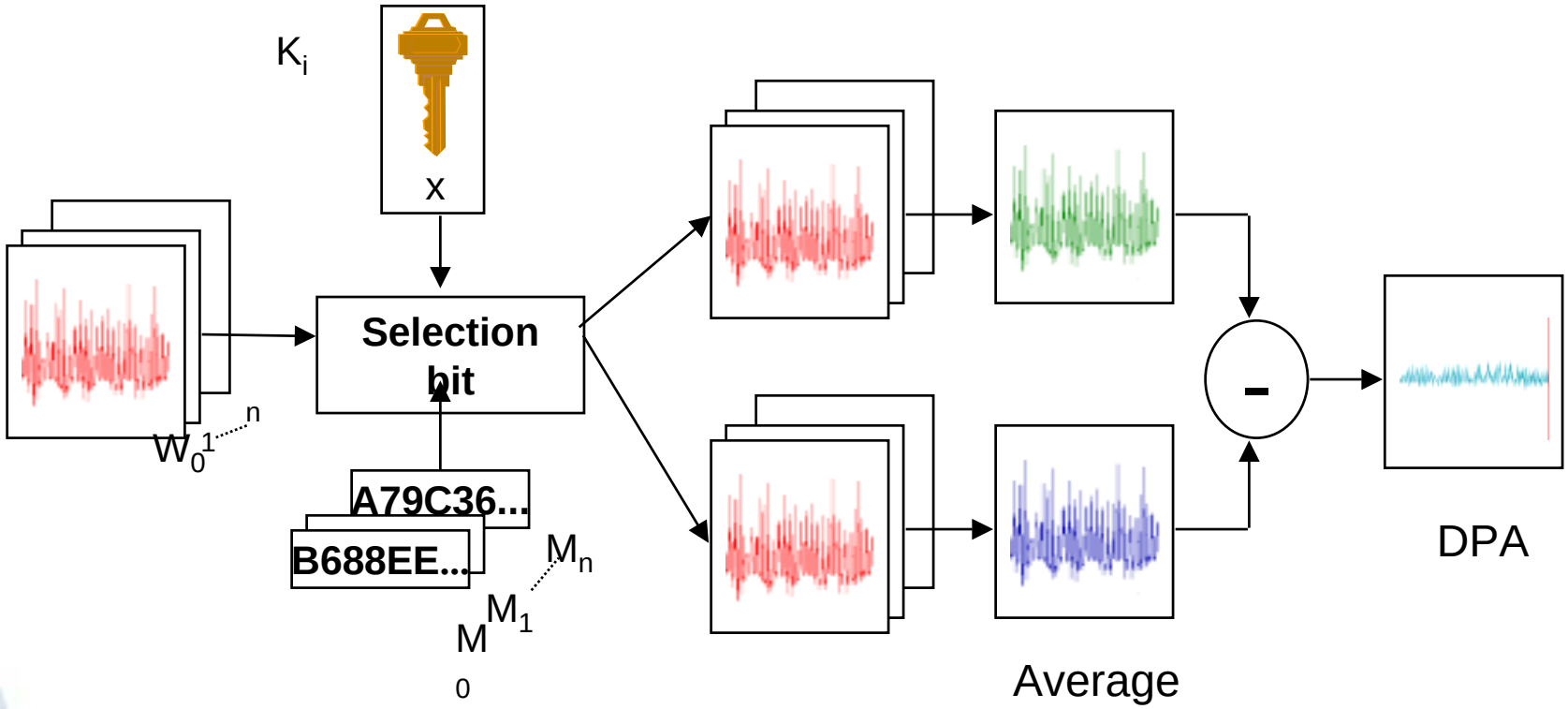
S = SB(K⊕M)

Attacking a Secret Key Agorithm

- **Try different keys a valid them with DPA**
- **Isn't it like cryptographic exhaustive search ?**
- **Not exactly ...**
- **... because the research space is drastically reduced !**

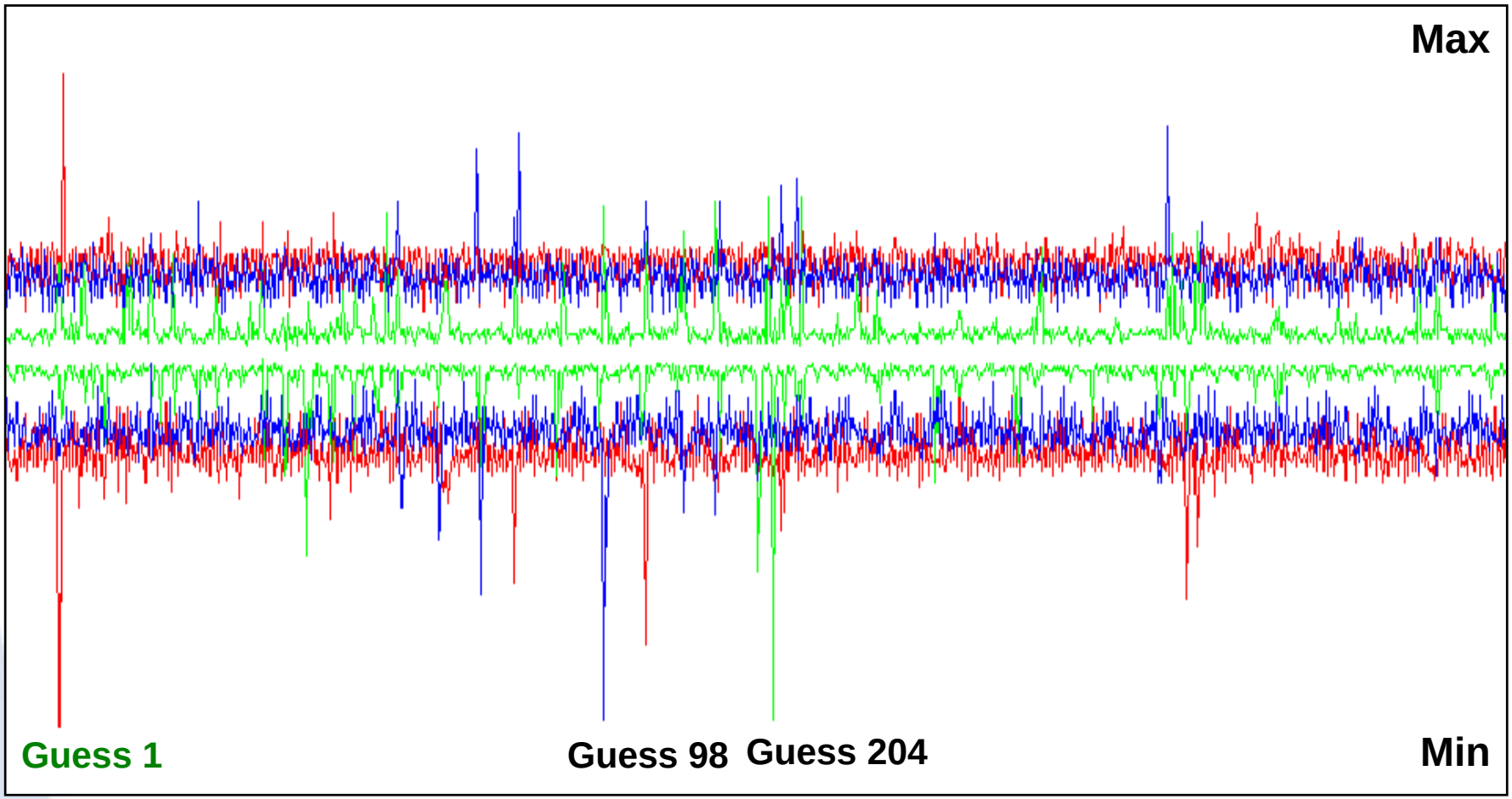
Hypothesis Testing (guess)

- **Example : AES 128 bits key = 16 bytes K_i (i = 1 to 16)**
  - **Test 256 guesses per K_i with 256 DPA**
  - **128 key bits disclosed with 16 x 256 = 4096 DPA (<< 2^{128} !)**



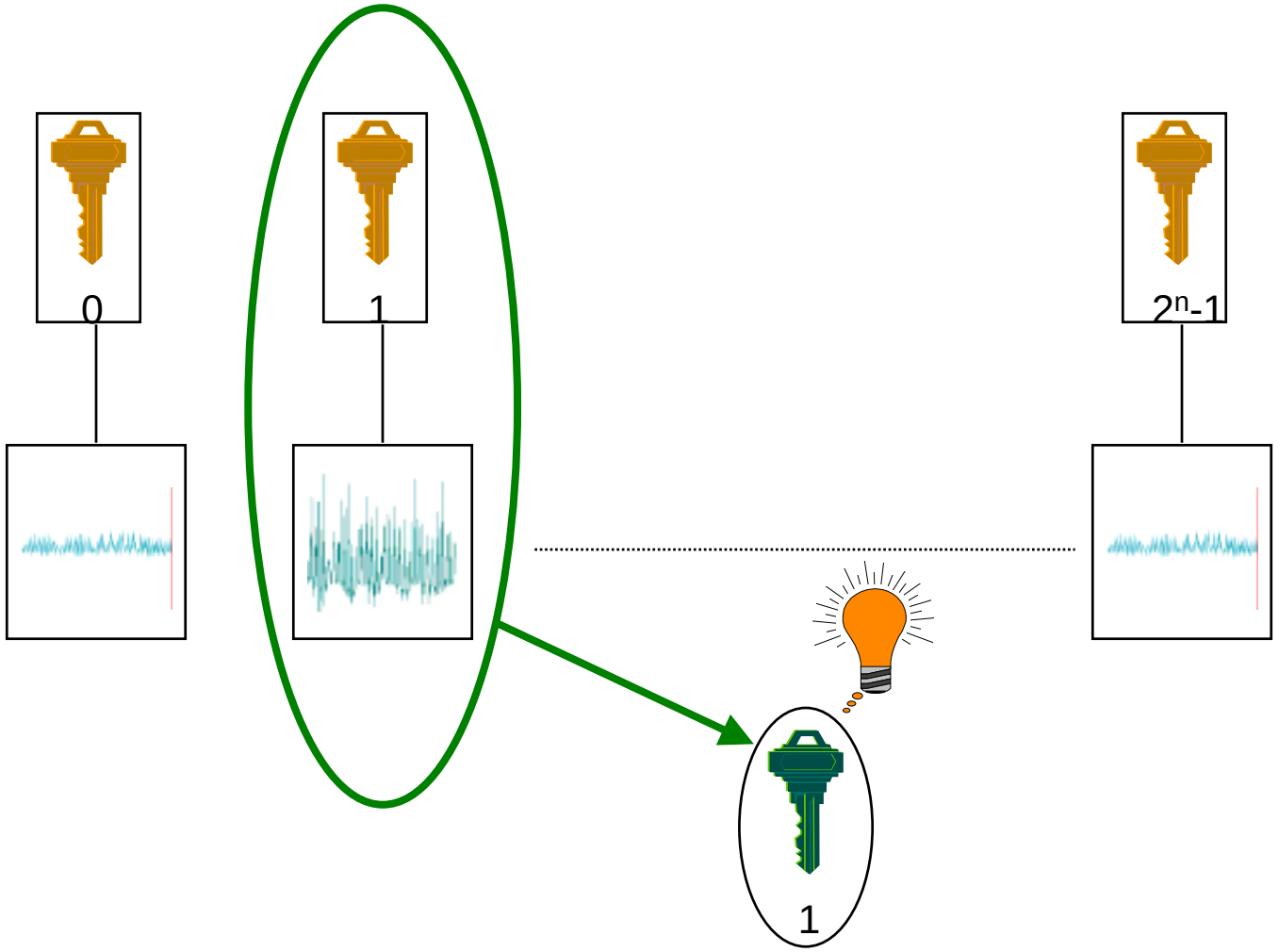
Hypothesis Testing (guess)

DPA on AES : 1st round and 1st byte (right guess = 1)



Hypothesis Testing (guess)

- **The right guess provides the highest spikes !**



Hypothesis Testing (right guess)

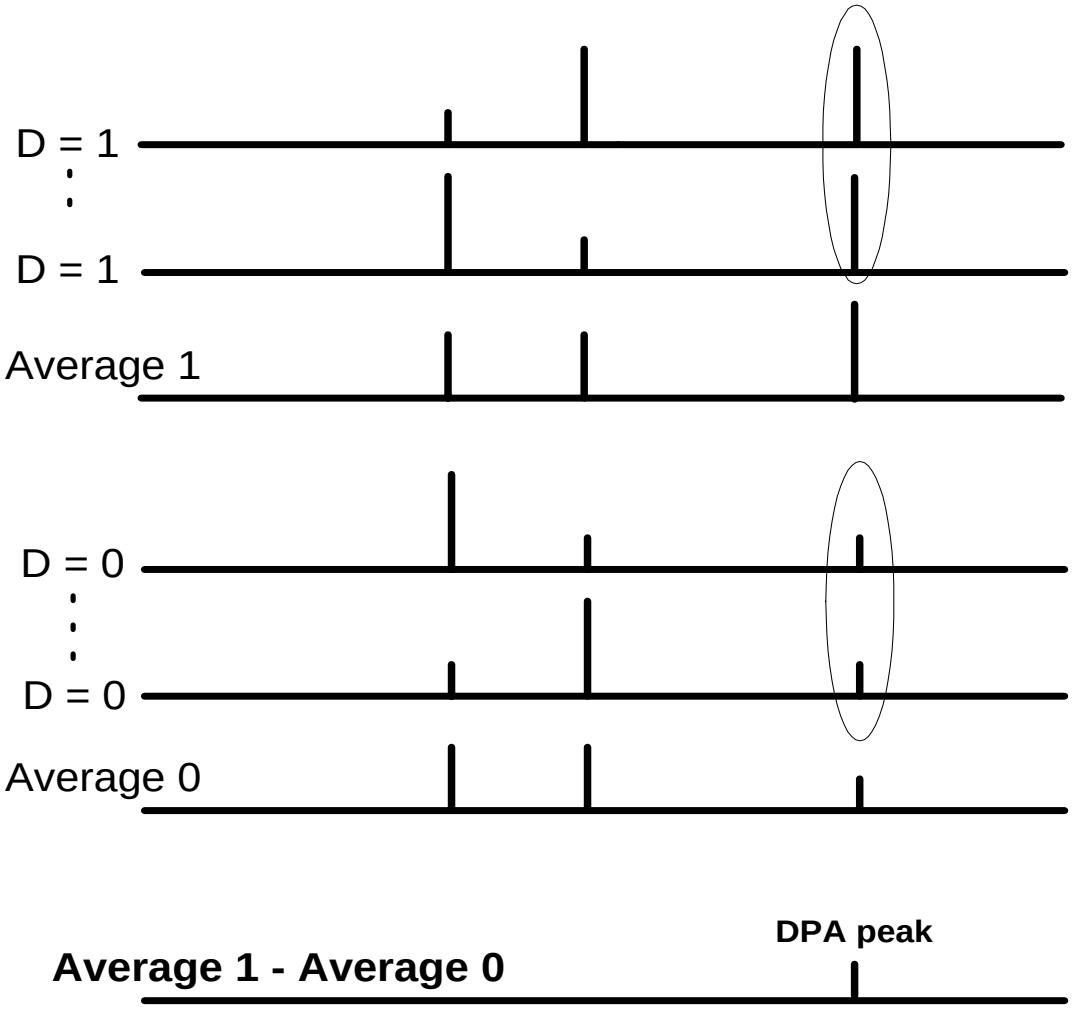
Right guess

Exact prediction of the selection bit

0	B688EE57BB63E03E	1	1
1	185D04D77509F36F	0	0
2	C031A0392DC881E6	1	1
...			

Real

Predicted



Hypothesis Testing (wrong guess)

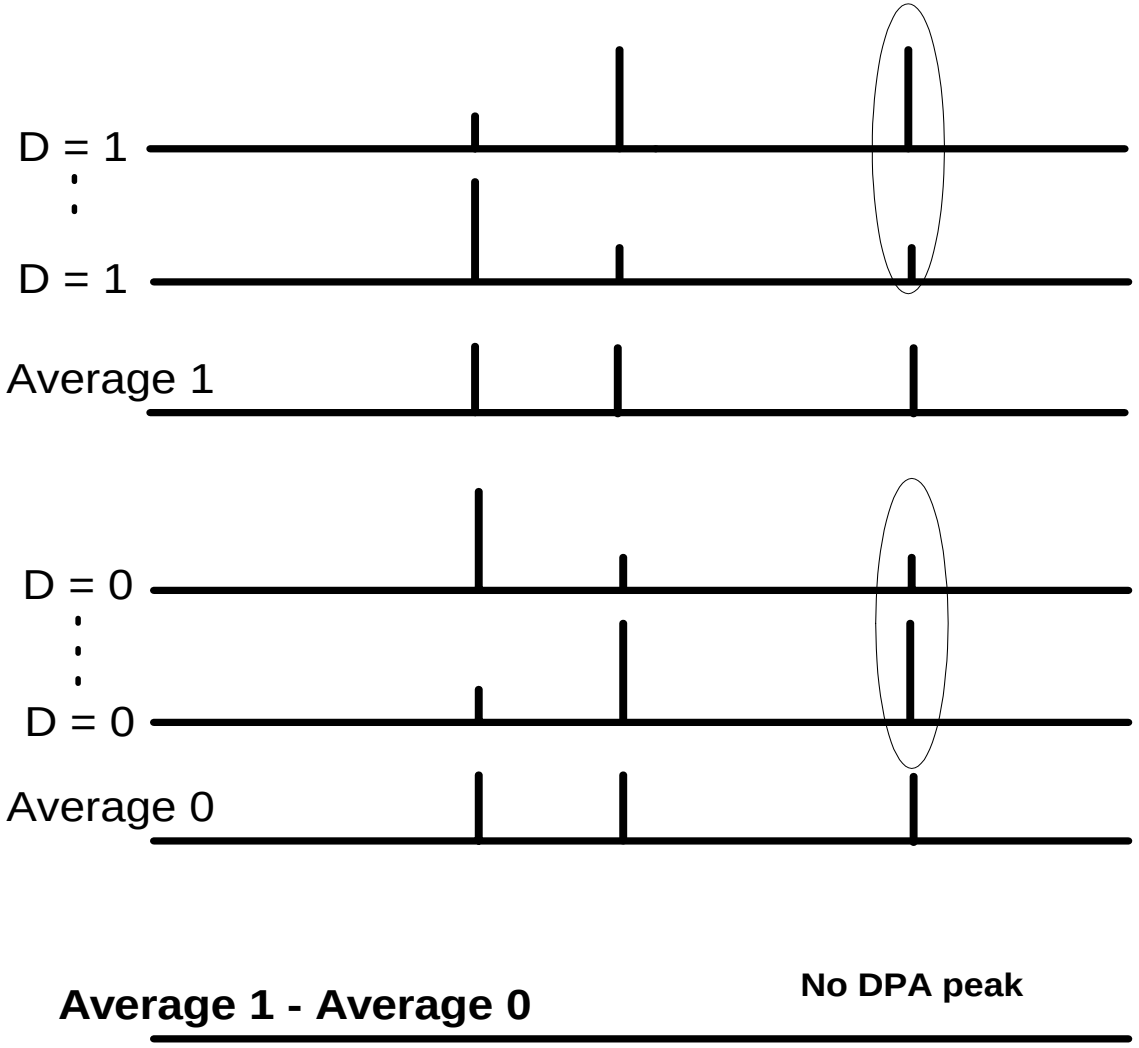
Wrong guess

Wrong prediction of the selection bit

0	B688EE57BB63E03E	1	0
1	185D04D77509F36F	0	1
2	C031A0392DC881E6	1	1
...			

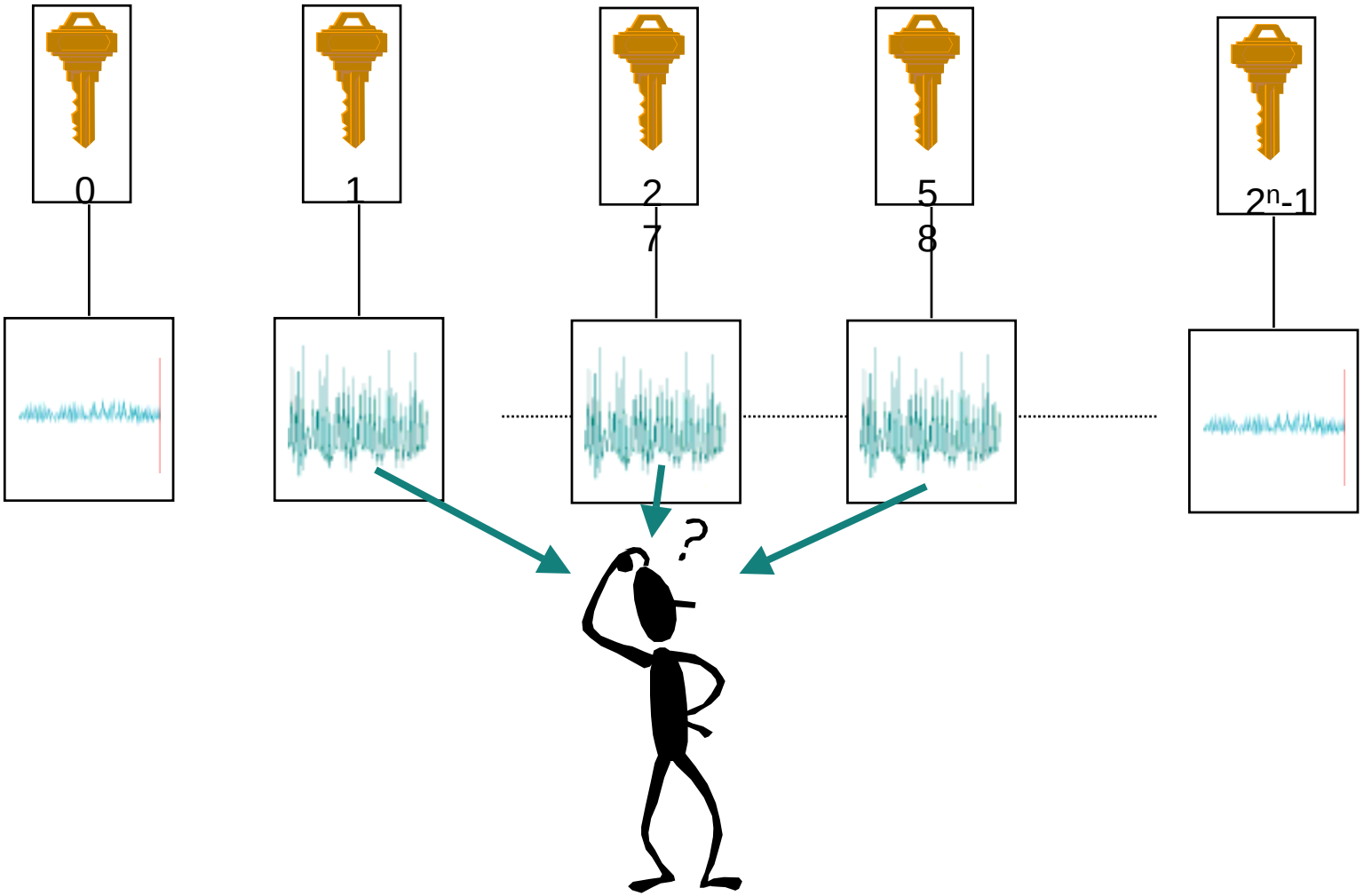
Real

Predicted



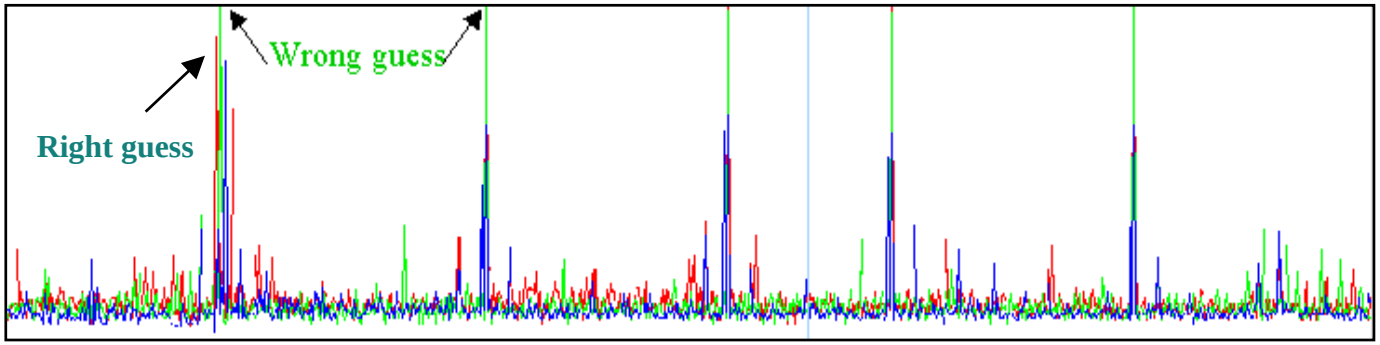
Hypothesis Testing (guess)

- **Wrong guesses may provide higher DPA peaks !**



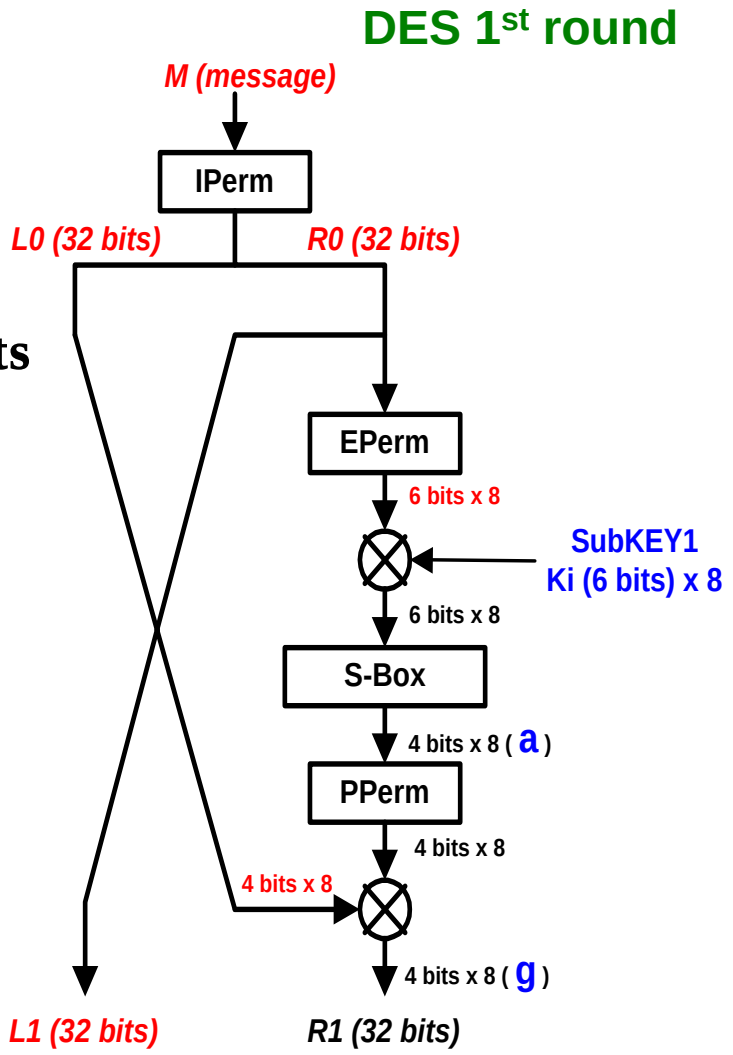
Hypothesis Testing

- Typical difficulties
  - low contrast between the guesses
  - wrong guesses leading to higher peaks (false alarm)
- Possible explanations
  - physical : lack of correlation between data & signals
  - cryptographic : algorithmic noise (implementation model)
- Practical solutions
  - try other selection bits (but they not necessarily agree !)
  - complementary exhaustive search on gleaned information



DPA on other algorithms

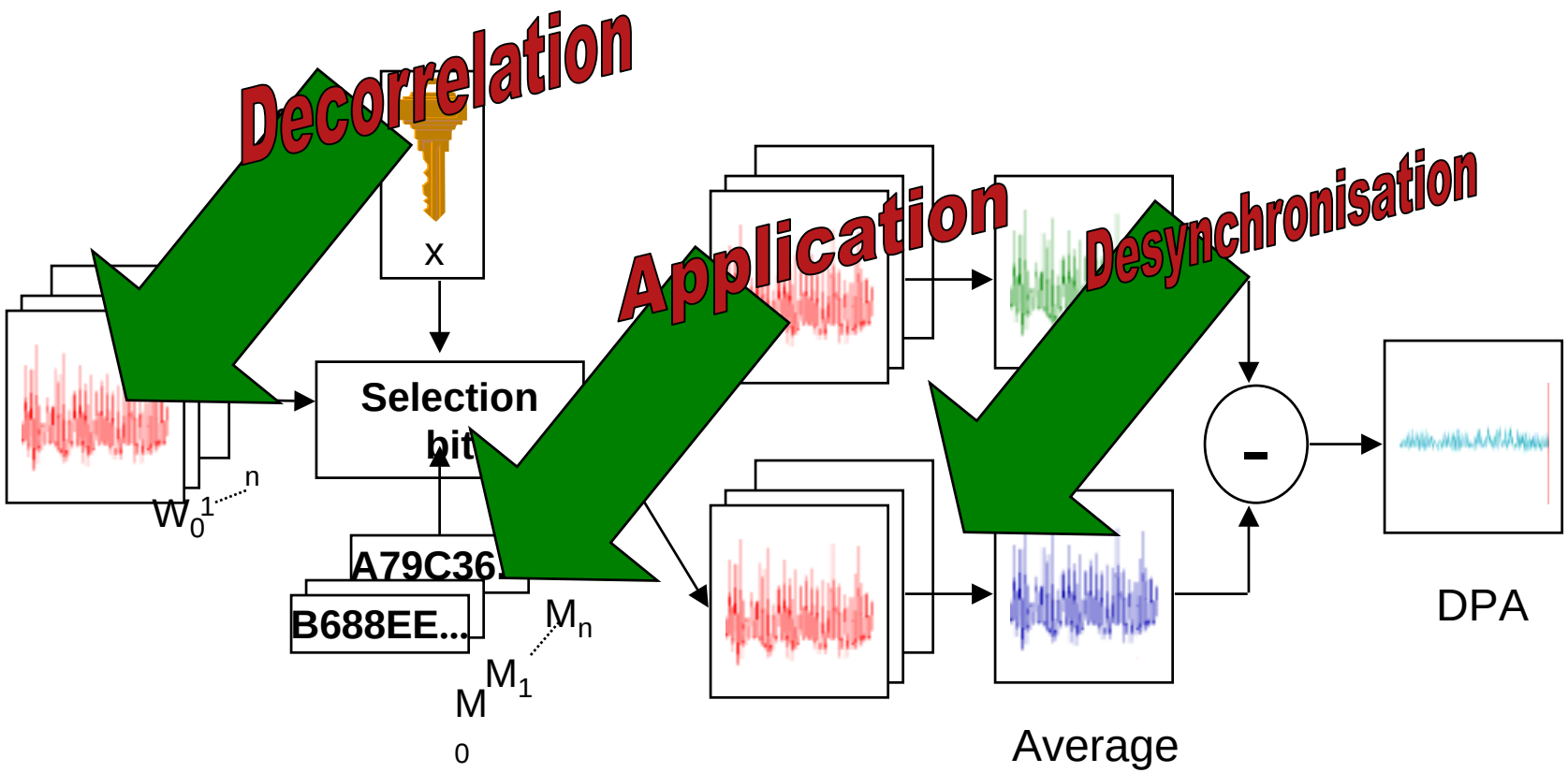
- DES (64 - 8 = 56 key bits)
  - historically the 1st target of DPA
  - more complicated
    - several possible targets
    - 8 Sboxes: 6 input bits, 4 output bits
    - key schedule (subkeys)
  - 8 x 64 = 512 guesses
  - only 48 bits on 1st round
  - 2nd round attack for the whole key
  - possibility of last round DPA



DPA on other Algorithms

- **Other SK algorithms**
  - **AES**
  - **3-DES**
  - **Comp 128**
  - **Hash MAC**
  - **modular arithmetic (modulo 256, 257)**
  - **proprietary (GSM)**
- **RSA modular exponentiation**
  - **No key schedule => prediction more difficult**
  - **The key is not entirely handled from the beginning, but progressively introduced**
  - **Prediction by time slices : next bit inference requires the previous bit to be broken**

DPA Countermeasures

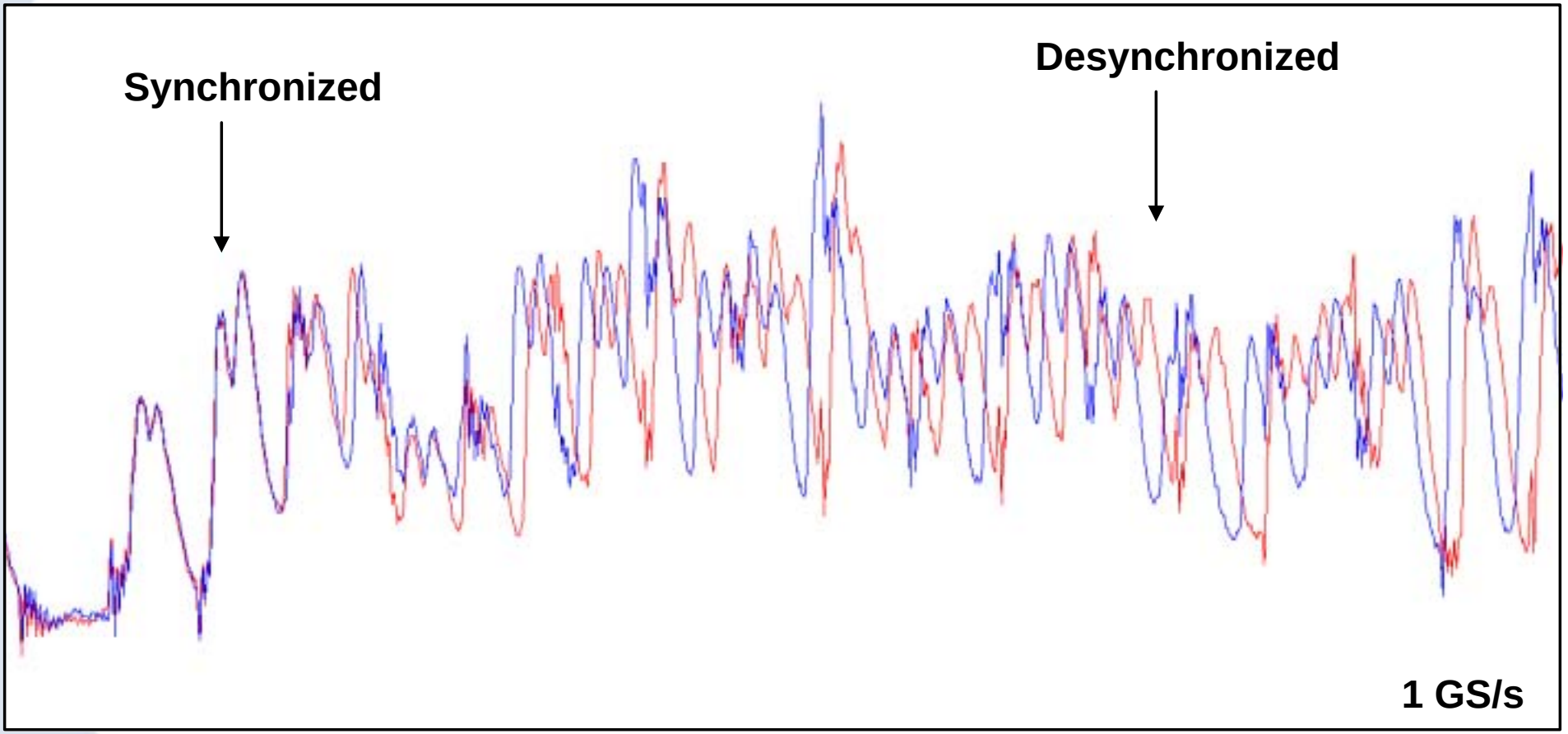


Anti-DPA counter-measures

- **Applicative counter-measures : make message free randomization impossible !**
  - **Fix some message bytes**
  - **Constrain the variable bytes (ex : transaction counter)**
- **Decorrelate power curves from data**
  - **by hardware : current scramblers (additive noise)**
  - **by software : data whitening**
- **Desynchronise the N traces (curves misalignment)**
  - **software random delays**
  - **software random orders (ex : SBoxes in random order)**
  - **hardware wait states (dummy cycles randomly added by the CPU)**
  - **hardware unstable internal clock (phase shift)**
- **DPA is powerful, generic (to many algorithms) and robust (to model errors)...**
- **... but there are counter-measures !**

Anti-DPA counter-measures

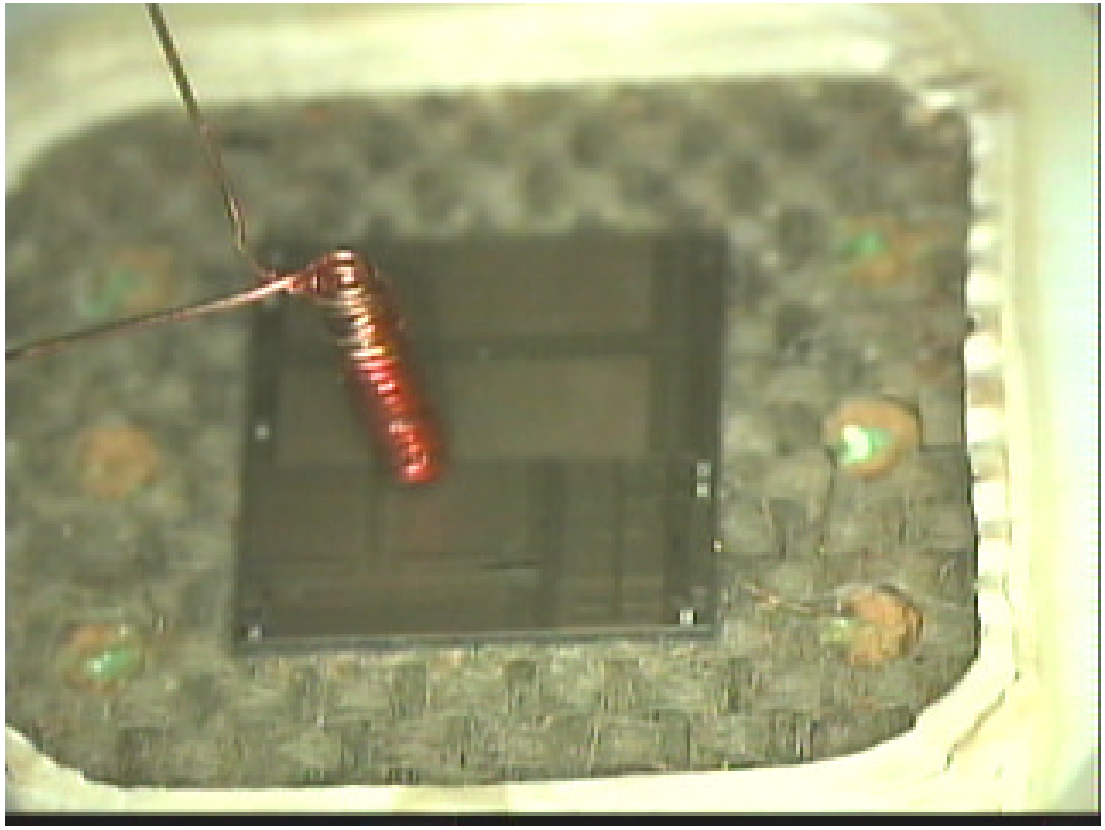
- **Internal clock effects (phase shift)**



DPA Summary

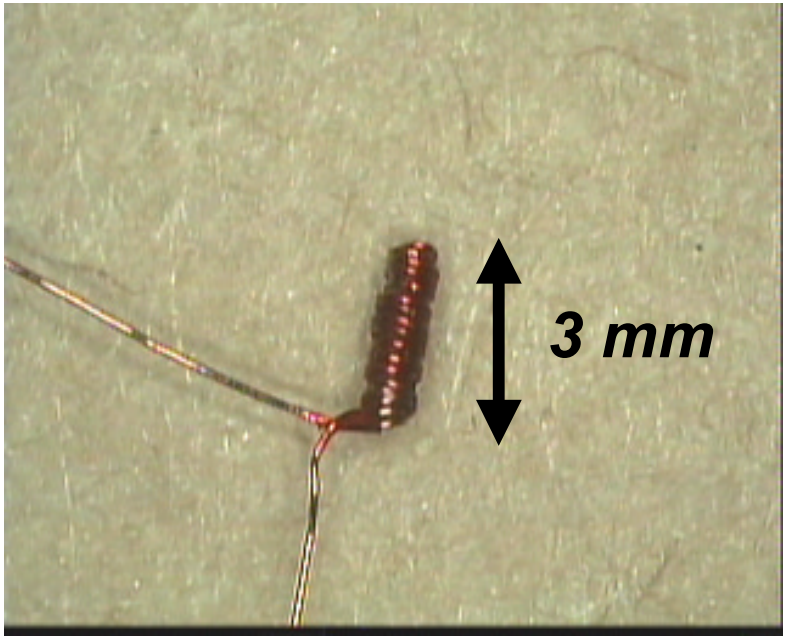
- Very effective
- Target independent
- Can be automated
- Does not require expensive hardware
- Effective countermeasures exist
- Still need to get a hold of the card

Electromagnetic Power Analysis



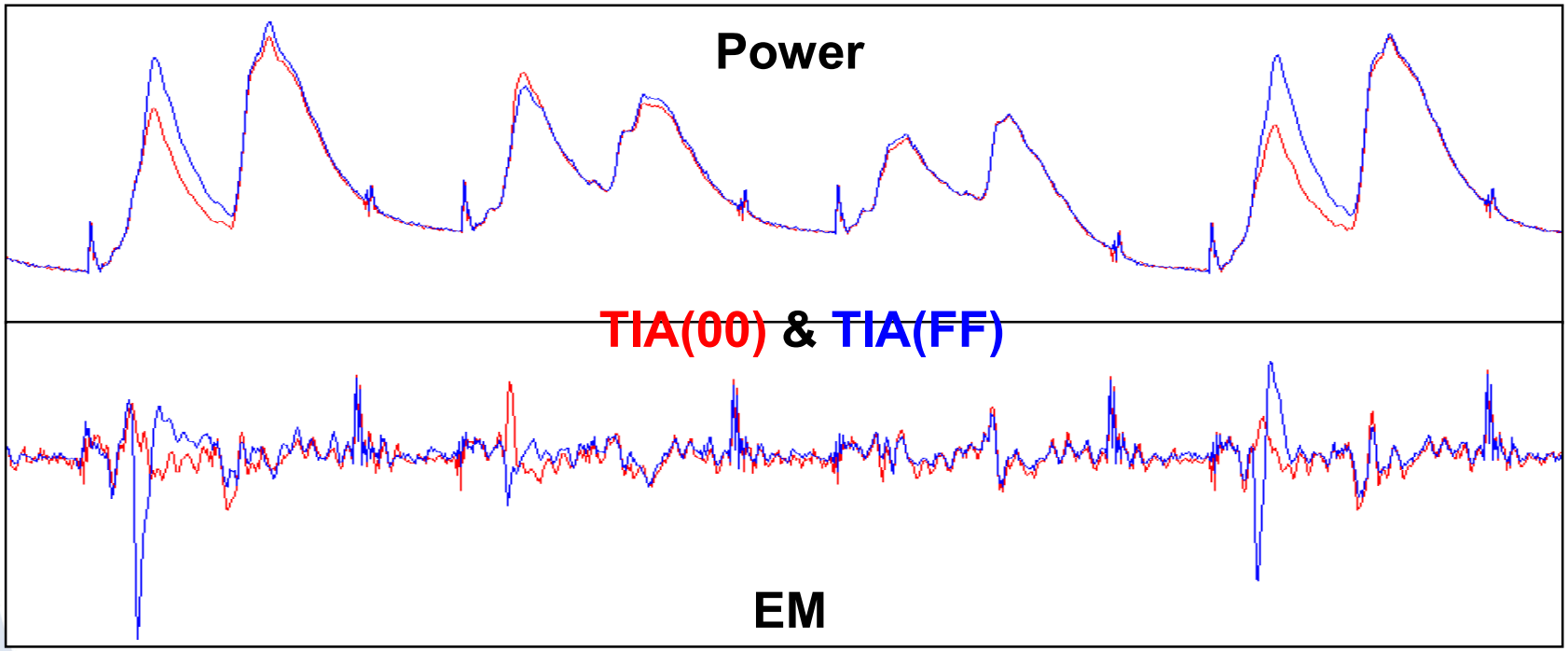
Probe design

- Hamming distance model for information leakage
  Correlated to the number of flipping bits (CMOS, VLSI)
- Electrical transitions disturb EM near field (and its flow ϕ)
- Captation by inductive probe
  - Handmade solenoid $V = -\frac{d\phi}{dt}$
    (Diameter = 150 to 500 µm)
  - Difficult to calibrate
    (Bandwidth > 100 MHz, low voltage, parasitic effects)
  - Good acquisition chain required, but no Faraday cage
    (Sampling at 1GHz)



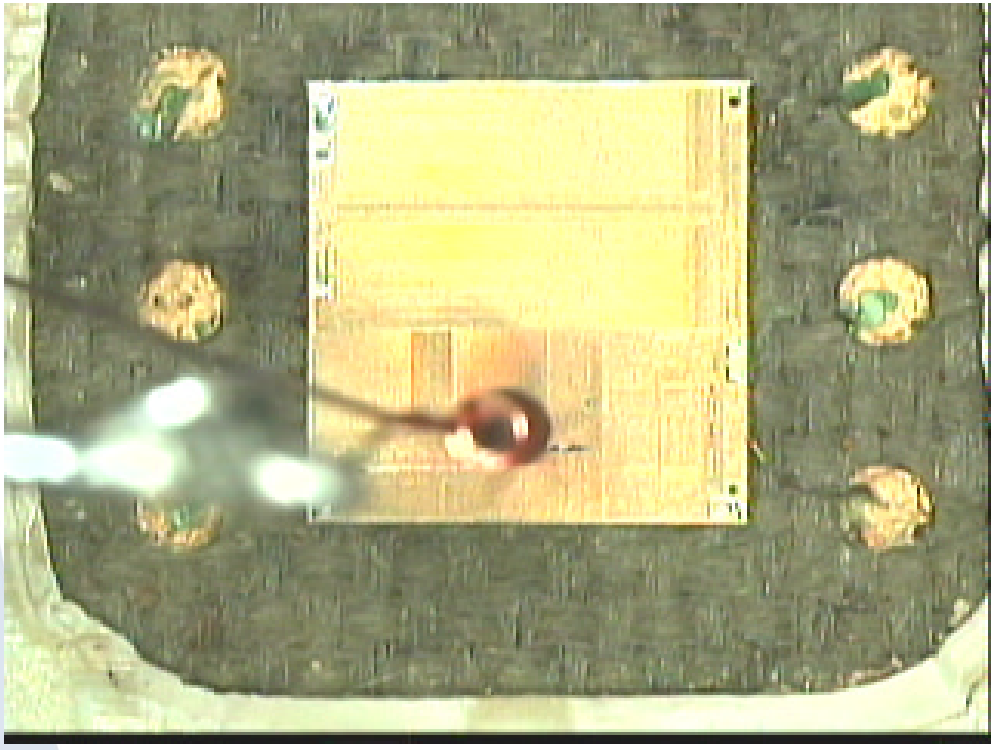
Electromagnetic Signals

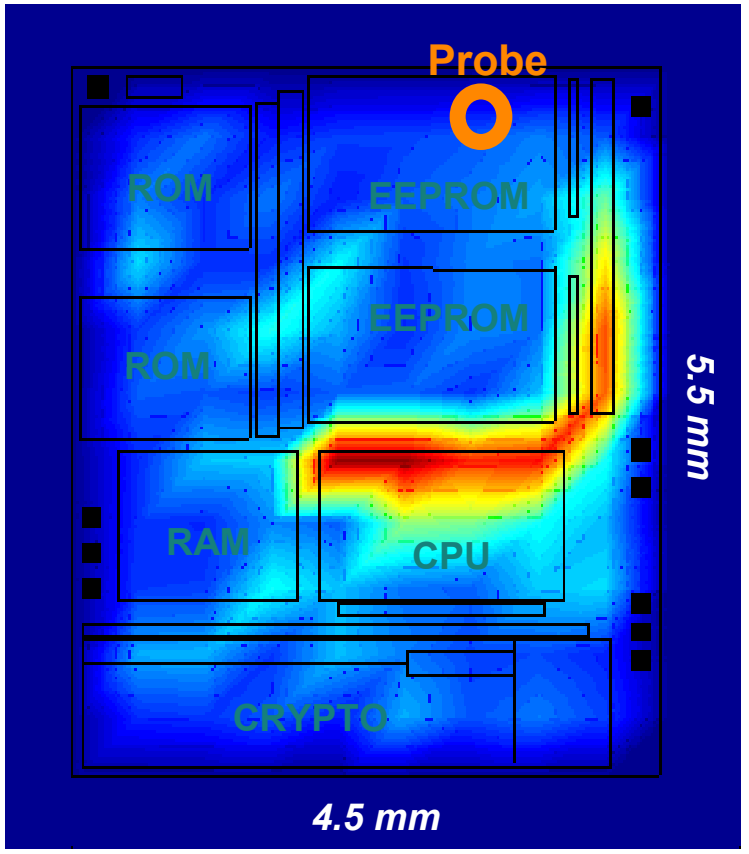
- Raw signals (TIA : transfer into accumulator instruction)
  - Power is less noisy
  - But EM signatures are sharper !



Spatial positioning

- Horizontal cartography (XY plane)
  - to pinpoint instruction related areas
  - better if automated





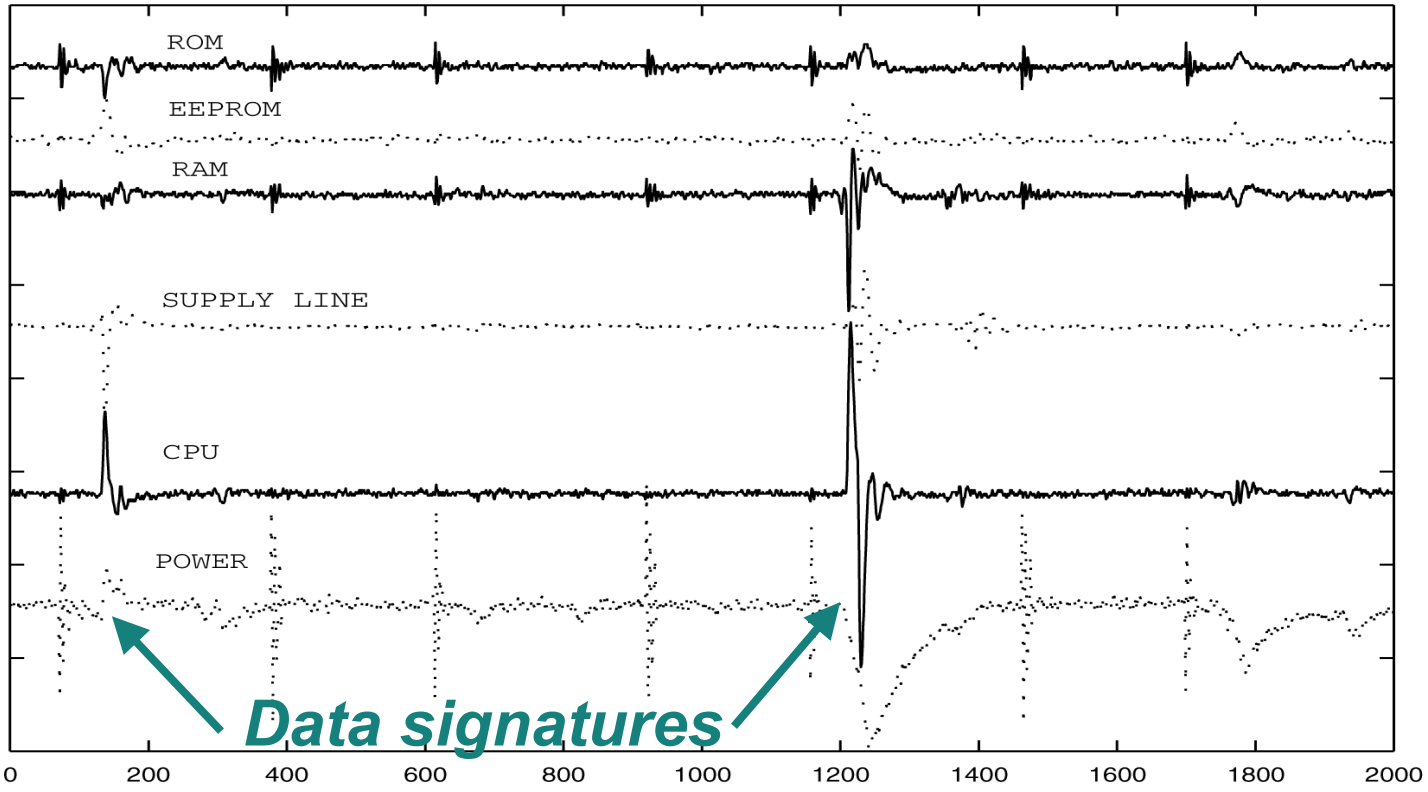
Spatial positioning

- EM signals versus XY probe position

Differential traces between (00h ⊕ 00h) and (FFh ⊕ 00h) picked up at different locations

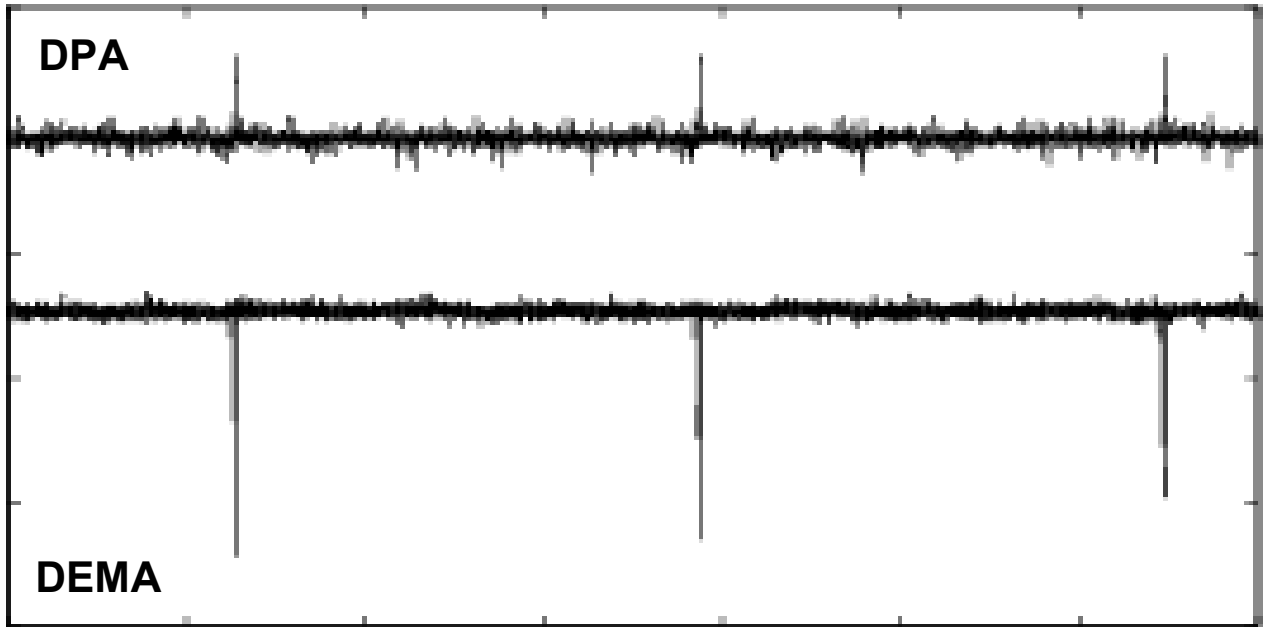
EM

Power



DEMA against the alleged COMP128

- First successful attack in Gemplus
- The DETECTION problem
  - better signal to noise on DEMA curves than on DPA
  - despite more noisy measurements !



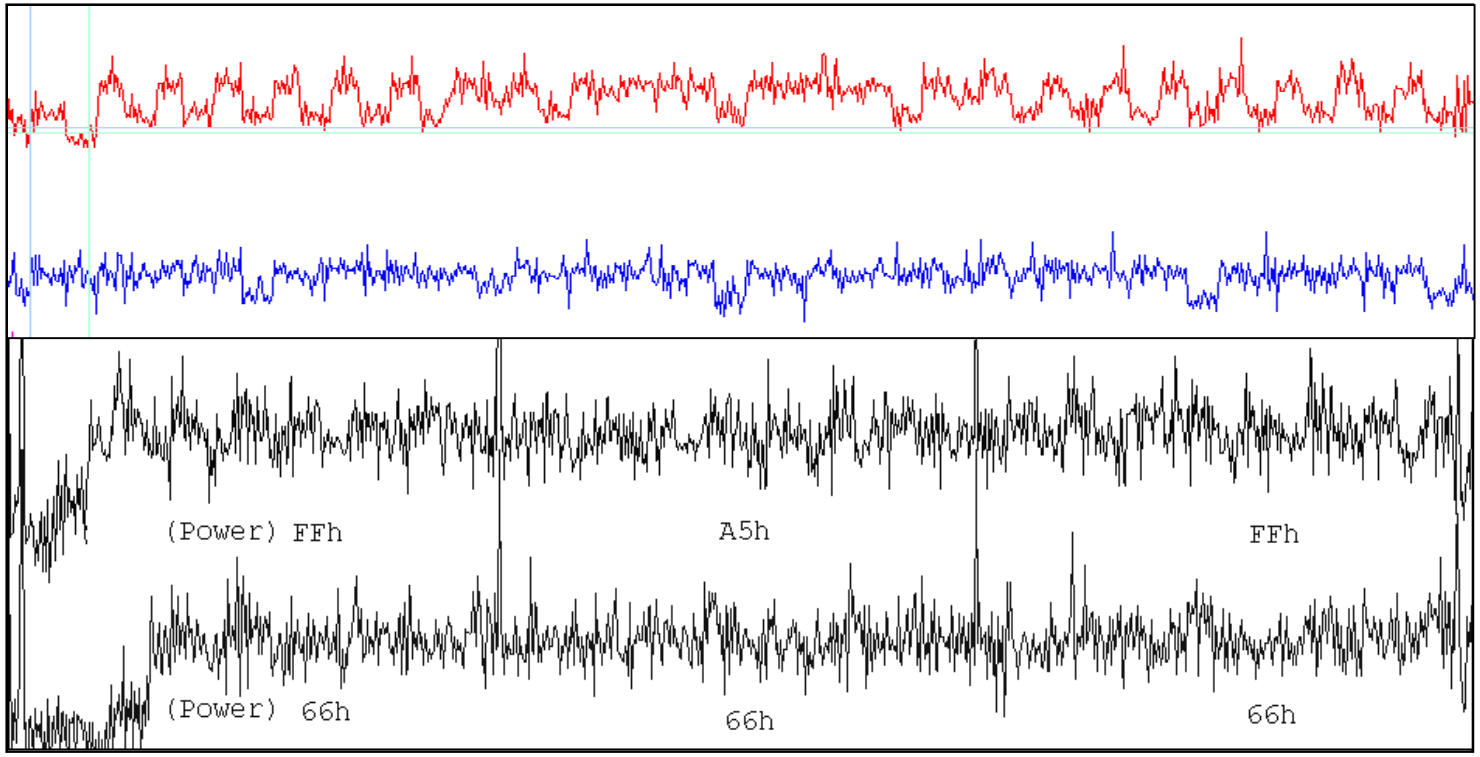
SEMA against RSA

- SEMA/SPA exploit larger scale patterns (single trace)
- Decapsulation (no statistical improvement for S/N)

  2 exponentiations involving 3 bytes of the private key : FFA5FFh and 666666h (same message and modulus).

EM patterns :

possible SEMA



Power

(no pattern : no SPA)

Electomagnetic Signals

- Advantage of EMA versus PA
  - Local information more "data correlated"
  - EMA bypasses current smoothers
  - EMA goes through HW countermeasures: shields, randomized logic
- Drawbacks
  - Experimentally more complicated
  - Geometrical scanning can be tedious
  - Low level and noisy signals (decapsulation required)

Countermeasures

- Software (crypto routines) :
  - coding techniques
  - same as anti DPA/SPA (data whitening…)
- Hardware (chip designers) :
  - confine the radiation (metal layer)
  - blur the radiation (e-g by an active emitting grid)
  - reduce the radiation (technology trends to shrinking)
  - cancel the radiation (dual logic)

Fault Induction (DFA)

- "Jolt" the smart card off its normal processing.
- Exploit any information that might be revealed
- Power glitches, flashes,...

Fault Attacks

- **Weights of coins:**
  - **a dollar** **5 grams**
  - **a cent** **3 grams**
- **How much money a 15 gram vault contains?**
- **Trivially, one of the following:**
  - **either** $5 \times 3 = 3$ **dollars**
  - **or** $5 \times 3 = 5$ **cents**
- **But how to make the difference?**

Fault Attacks

- **Assume that the owner of the vault has the habitude of counting each evening the amount in the vault.**
- **Have him drink some Vodka so that he mistakes a cent for a dollar or the other way around.**
- **Put the vault on the balance again:**
  - **either** $5 \times 2 + 3 = 13$ **grams**
  - **or** $4 \times 3 + 5 = 17$ **grams**
- **You can now tell exactly what was the amount in the vault.**

RSA using the Chinese Remainder Theorem

- ***a*** **and** ***b*** **are precumputed values, such that:**

$$\begin{cases} a \equiv 1 \pmod{p} \\ a \equiv 0 \pmod{q} \end{cases} \quad \text{and} \quad \begin{cases} b \equiv 0 \pmod{p} \\ b \equiv 1 \pmod{q} \end{cases}$$

and define:

$$\begin{aligned} d_p &= d \pmod{p-1} \\ d_q &= d \pmod{q-1} \end{aligned}$$

- **The two elements that replace** ***d*** **will be half the size (in bits) compared to** ***d*****.**

RSA using the Chinese Remainder Theorem

- **The secret key elements are used to calculate:**

$$\left.\begin{aligned} s_p &= m^{d_p} \pmod{p} \\ s_q &= m^{d_q} \pmod{q} \end{aligned}\right|$$

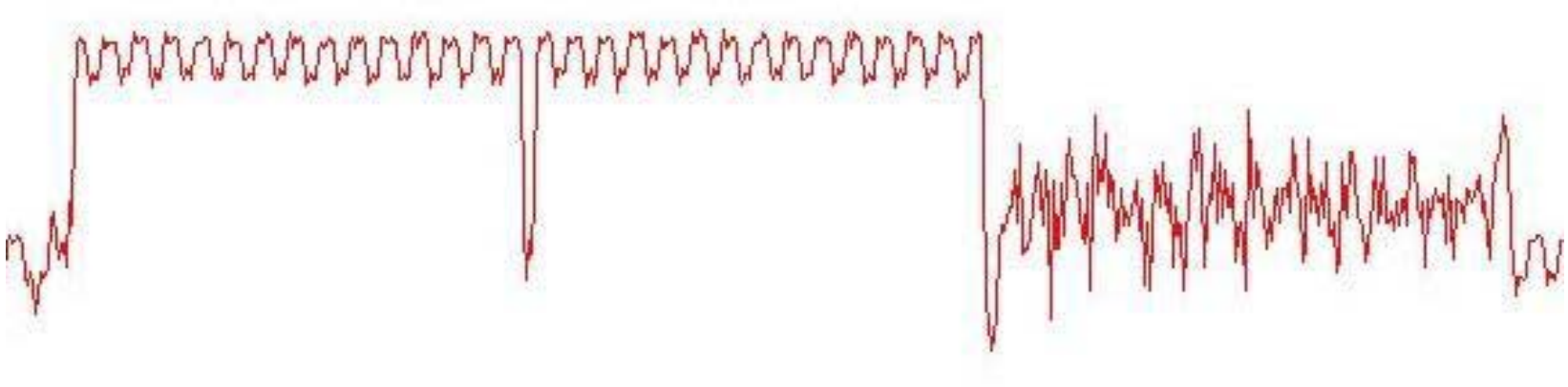
These two exponentiations will be computed four times faster than the usual m^d (mod n).

- **Which can be used to generate a signature:**

$$s = a \times s_p + b \times s_q \pmod{N}$$

Against an RSA using the Chinese Remainder Theorum

- **Two exponentiations to generate a RSA signature.**
- **A fault in one exponentation will provide an incorrect result that can leak information on the secret key used.**



If a fault occurs …

- **If a fault occurs during the calculation of S_q then:**

$$\begin{aligned}\Delta &\equiv s - \hat{s} \\ &\equiv (a \times s_p + b \times s_q) - (a \times s_p + b \times \hat{s}_q) \\ &\equiv b(s_q - \hat{s}_q) \pmod{N}\end{aligned}$$

and the secret prime numbers can be found by

$$GCD(\Delta \pmod{N}, N) = p$$

$$q = N/p.$$

Against an RSA using the Chinese Remainder Theorum

- **A correct signature S and an incorrect signature S' can be used to derive one of the prime numbers used in RSA.**
- **A GCD between the difference and n:**

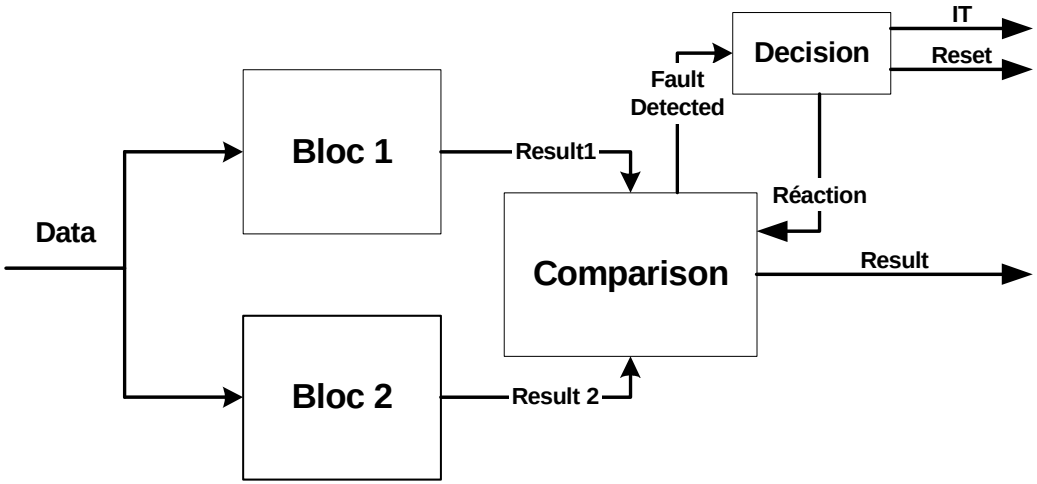
$$\begin{aligned}\gcd\left(S - S', n\right) &= \gcd\left(a\left(S - S'\right), n\right) \\ &= q\end{aligned}$$

Countermeasures

- **Software**
  - **Execution redundancy**
    - **repeating an algorithm**
    - **executing the inverse algorithm (ideal for RSA)**
  - **Checksums on data transfers**
  - **Randomised Execution**

Countermeasures

- **Hardware**
  - **Redundancy - hardware implemented twice with a comparison.**
  - **Better detectors**



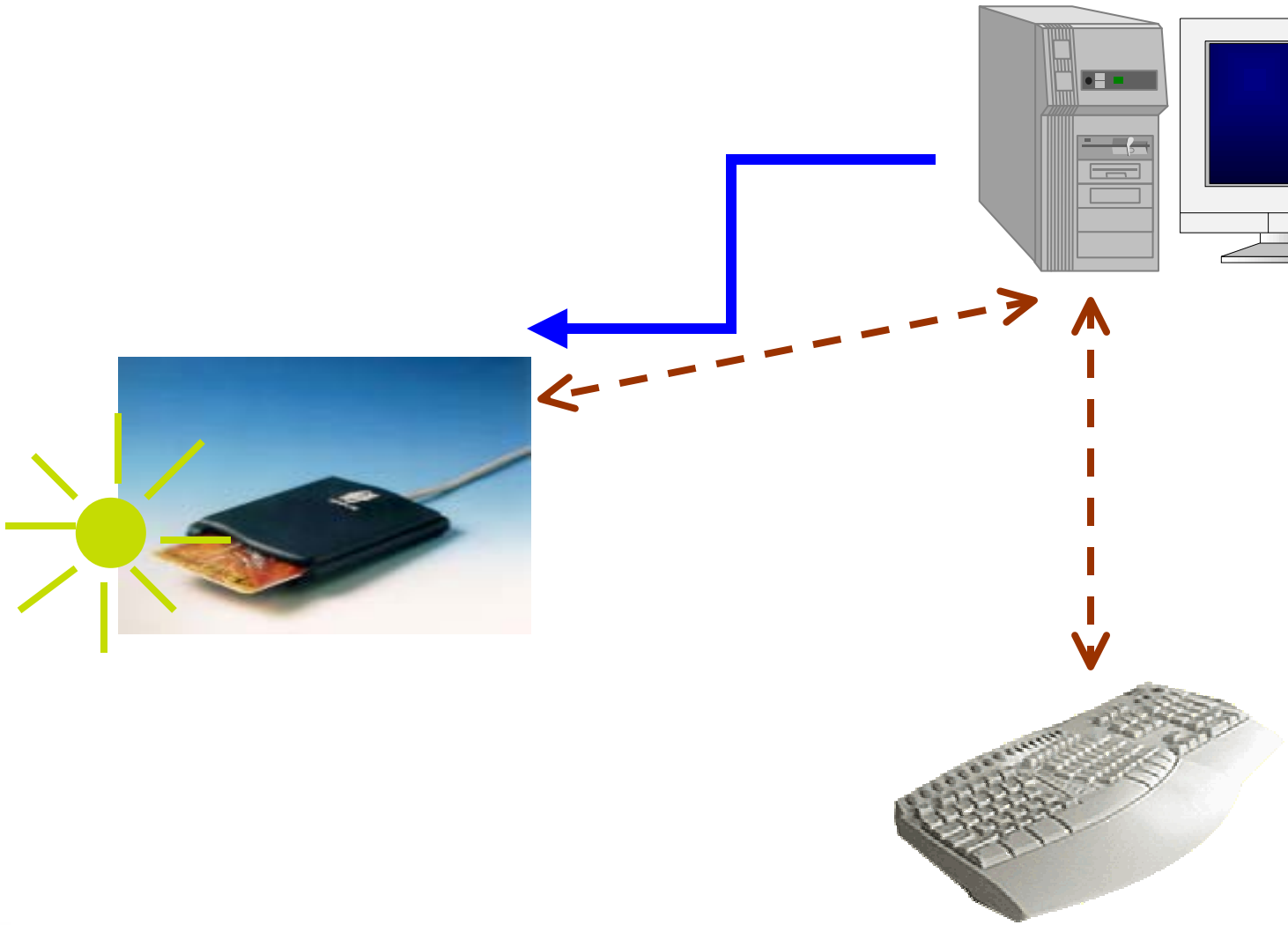
Outline

- **Smart Cards: What and Why**
- **Attacks on cards**
  - **Physical**
  - **Timing, spa, dpa, dfa**
- **Attacks on systems using Smart Cards**
- **Examples**

Systems Using Smart Cards

- **Smart cards can not interact directly with the card holder**
- **Smart cards are used in IT systems to store users credentials for authentication, signature or ciphering**
- **Classical IT security concepts apply to these systems**
  - **Trusted path**
  - **Security policies**
  - **Trojan horses**

Trusted Path: Normal PIN Verification on a PC



Trusted Path: PIN code verification

Trojan Horses: the Future

- **Stealing the PIN might not be interesting**
- **Placing calls on expensive numbers would...**

- **What about J2ME phones?**

Summary

- Bugs
  - Insecure Handling of data
  - Buffer Overflows
- Conceptions errors
  - Plain text or bad protocols
  - Reverse Engineering of programs
- Attack on the TCB
  - Trojan Horses
  - Viruses

Wrong Cryptographic Design

- Short keys
- Weak algorithms
- Broken protocols
- …

- Examples
  - French Credit Card
  - COMP128 for GSM


L'homme qui défie les cartes à puce

Passionné d'infor… à la justice pour a… de protection de…

10/SOCIETE

Génie incompris ou véritable escroc ?

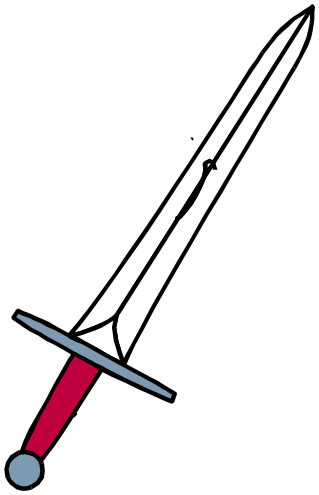
Carte bleue: l'homme qui a trouvé la faille

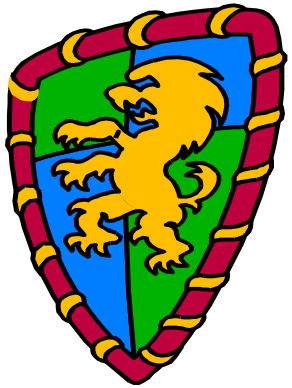
Serge Humpich a mis au point un système permettant de tromper les terminaux bancaires. Mal lui en a pris.

GEMPLUS

Conclusion

- **Smart cards are like any security devices: they have limitations**
- **A system should be designed with these limitations in mind**
- **A system should be upgradeable to deal with the (inevitable?) security breach or the aging of the technology**
- **The race goes on...**





Scenario #01

- **Symptom : Merchants complain that their fund-deposits are never credited to their accounts.**
- **Deposits are sealed in specific envelops at the end of each day and deposited by the merchants in the front-door mailboxes of their banks. Physical access to mailboxes is smart-card protected.**
- **Police investigation : card access control OK, mailbox OK.**

Scenario #01 : what happened

- **The criminal has opened an account at the attacked bank; pretending he was a merchant. He consequently got a smart-card granting him access to the mailbox.**
- **He then bought a heavy metal vault, made a rectangular hole in it and came to the bank just after office hour. Using his smart-card he accessed the mailbox's man-trap, taped shut the real mailbox's hole and placed the vault on the ground, securing it with a steel chain and an impressive padlock. Finally, he added a printed text (bearing the bank's logo) : "WORK IN PROGRESS, PLEASE PLACE YOUR DEPOSITS IN THE VAULT, THANKS".**

Scenario #02

- **Symptom : Users insert their cards to ATMs, enter their PINs but get no money. The ATM swallows the card and displays the message "INVALID CARD CONTACT YOUR BANK", money was however withdrawn with the card later.**

Scenario #02 : what happened

- **A false ATM…**

Scenario #03

- **Symptom : Same as scenario 2, using a smart-card with an EEPROM counter limited to 3. The card is always returned to the user but if its EEPROM counter contains 3 the card can not be used anymore.**
- **An audit of the ATM's log file showed that although the thief presented three false PIN codes, he could somehow try again and again. The correct PIN was found by exhaustive search after approximately 5000 attempts.**

Scenario #03 : what happened

- **In old cards, EEPROM programming was done using an external programming voltage (Vpp) supplied through a specific ISO contact. The thief had covered this specific card contact with a paper sticker (EEPROM programming made impossible).**

Scenario #04

- **Symptom : The ATM's log file and cash stock do not match; money is missing.**
- **An audit of the ATM's log file showed that the same user withdrew money several times. He always forgot his banknotes that were swallowed back by the ATM after a short time-out (a security feature).**

Scenario #04 : what happened

- **The thief would withdraw three banknotes but take only two of them. The remaining banknote was detected by the paper sensor and swallowed back by the ATM which automatically cancelled the transaction (no debit on user's account).**
- **The paper sensor could not distinguish between one, two or three banknotes...**

Scenario #05

- **Symptom : Although PIN-protected, stolen smart credit cards were successfully used to withdraw money.**
- **An audit of the ATM's log file shows that the correct PIN was used in the withdrawal operation.**

Scenario #05 : what happened

- **The fraud was technical : the smart-card's software was programmed to compare the presented PIN and if incorrect to increase the EEPROM counter.**
- **EEPROM programming is characterised by an increased power consumption and requires 5ms.**
- **The thief used a board that presented automatically all the PIN values (0000 to 9999) but detected the current consumption increase and powered off the card before the EEPROM error counter could be updated.**

Scenario #06

- **Symptom : Users complain that the ATM has swallowed their card but when employees look into the ATM there is no card.**

Scenario #06 : what happened

- A collar was put on the front of the hole, once the card was in it was impossible to get out.
- A thief told the victim to go in the bank to ask for the card and then get out the card and the collar and ran away.
- Unfortunately for us this technique was quite popular in Marseilles and was called “Le collet marseillais”