Fault attacks: yet another concern for the designer

Ingrid Verbauwhede
ingrid.verbauwhede-at-esat.kuleuven.be

K.U.Leuven, COSIC
Computer Security and Industrial Cryptography
www.esat.kuleuven.be/cosic

with input from:
Ph.D. students, former and current

Goal

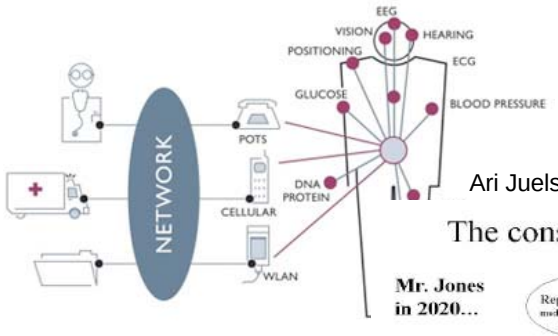
- VIEWPOINT of the designer
- GOAL: public-key crypto **within** budget

- Give designer insight into jungle of fault attacks and countermeasures
  - Extra design consideration
  - Trade-off with other design parameters

Outline

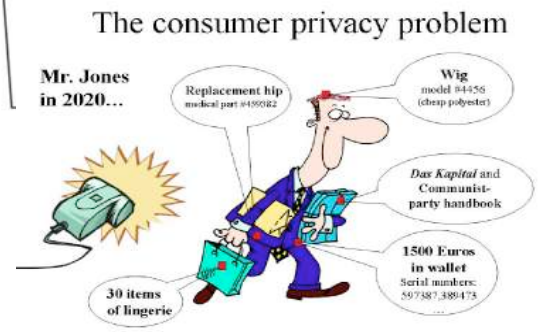
- Application domain
- How much public key in 1 microJoule?
- Attacks and more specific fault attacks
- Countermeasures
- Cost & Trade-offs
- Design methodology for security
- Conclusion

Embedded crypto everywhere

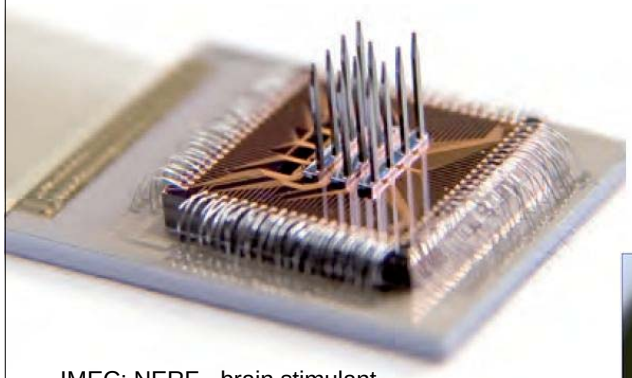


IMEC: Human++ project

Ari Juels: RFID tracking problem



Embedded crypto everywhere



IMEC: NERF - brain stimulant

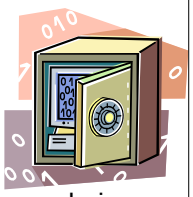
Deep Brain stimulation

[Sources: J. Rabaey, National Institutes of Health, Neurology journal]

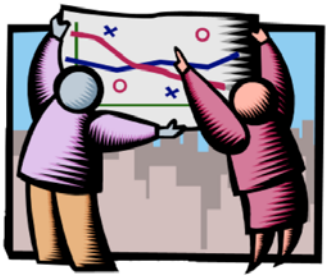
Embedded Security

NEED BOTH

- Efficient, light-weight Implementation
  - Within power, area, timing budgets
  - Public key: 1024 bits RSA on 8 bit μC and 100 μW
  - Public key on a passive RFID tag



- Trustworthy implementation
  - Active attacks: probing, power glitches, laser, JTAG scan chain
  - Passive attacks: side channel attacks, including power, timing and electromagnetic leaks
  - Combined attacks: SCA & FA



Design Parameters

Embedded security:

Area, delay, power, energy,
physical security

plus testability, NRE,
manufacturability, ….

Design parameters

- Speed or throughput:
  - Gbits/sec or Mbits/sec/slice
  - Cycles/byte
- Area:
  - mm2 (gate or transistor count)
  - Memory
- Power or energy consumption:
  - Power (Watts) for cooling or transmission (RFID)
  - Energy: battery operated devices
- Security:
  - How to measure attack resistance??
  - Passive: side channel resistance
  - Active: fault attacks, tampering

Power and Energy are not the same!

- Power = P = I x V (current x voltage) (= Watt)
  - instantaneous
  - Typically checked for cooling or for peak performance
- Energy = Power x execution time (= Joule)
  - Battery content is expressed in Joules
  - Gives idea of how much Joules to get the job done

Low power processor ≠ low energy solution !

- Low clock for low power does not necessarily result in low energy …

Medical implants

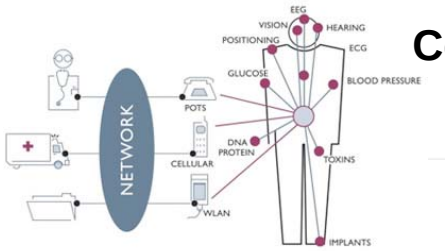
- Power is limited
  - Cooling!!
  - Implanted devices only temperature $\Delta < 1$ °C
- Battery is limited
  - Pace maker battery is not rechargeable
  - One AAA battery is 1300 … 5000 Joules
- **How much crypto in one micro Joule ?**

  and Public Key crypto in one microJoule?

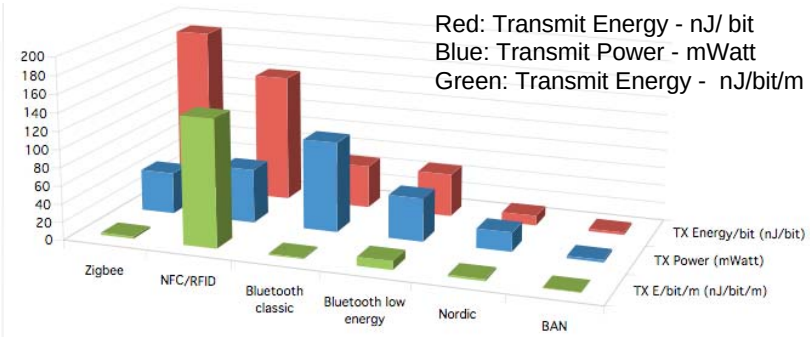
Example: Body Area Network

Computation - communication cost
first without countermeasures

BAN: Communication cost

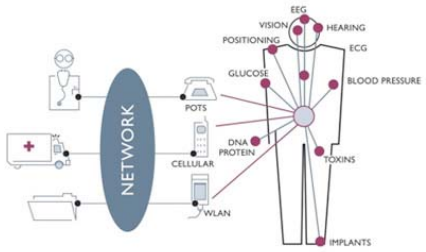


Red: Transmit Energy - nJ/ bit
Blue: Transmit Power - mWatt
Green: Transmit Energy - nJ/bit/m



[source: G. Dolmans IMEC NL]

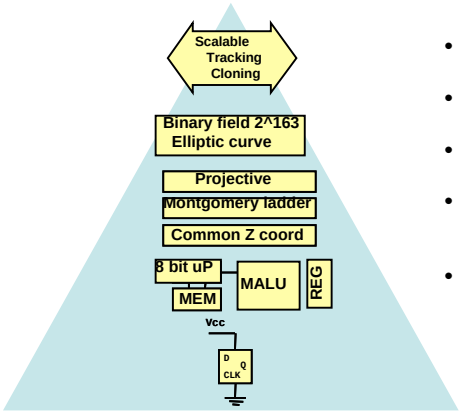
BAN: Computation cost



- Public key based - Elliptic Curve
- **Push** for lowest energy to fit budget of RFID ASIC - domain specific processor solution

ECC push for lowest energy

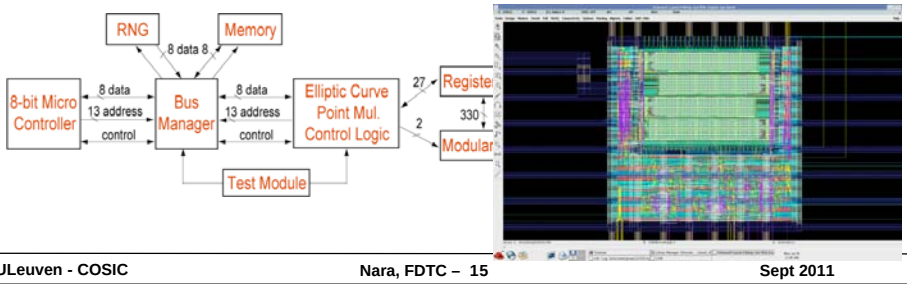
Address at all design abstraction levels!



- **Protocol** : asymmetric (most work for the reader)
- **Algorithm**: Elliptic curve (163 bits) instead of RSA (min 1024 bits)
- **Field Operation**: Binary and not Prime fields: easier field operations
- **Projective** coordinate system: (X, Y, Z) instead of (x,y): no field inversions
- **Special coordinate system**: no need to store Y coordinates (Lopez-Dahab) and common Z (only one Z coordinate)
- **Minimize storage**: Only 5 registers (with mult/add/square unit) or 6 registers (with mult/add-only unit) compared to 9+ registers before.

Results

- Results: ECC co-processor that can compute:
  - ECC point multiplications (163 by 4)
  - Scalar modular operations (8 bit processor with redundancy)
- Schnorr (secure ID transfer, but no tracking protection): **one** PM
- More advanced protocols: up to **four** PM on tag
- 14K gates, 79K cycles
- At 500 KHz, corresponds to 30 microWatt and 158 msec
- One point multiplication = **4.8 microJoule**



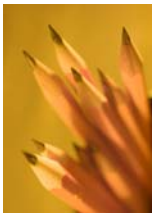
1 micro Joule

Transmission:

- 300 bits in BAN
- 11 bits Bluetooth
- 3 bits Zigbee

Encryption:

- 11000 bits AES
- 500 bits SHA3 hash
- 1/5 of one point multiplication



ECC based randomized Schnorr

Reader: $y, X = xP$

Tag: $x, Y = yP$

$r_1, r_2,$

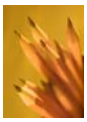
T_1, T_2 ← $T_1 = r_1P, T_2 = r_2Y$

c → c

$v = r_1 + r_2 + cx$

v ←

$c^{-1}[vP - T_1 - y^{-1}T_2] = ? X$



Tag: two point multiplications, two transmissions over BAN
Crypto dominates ≈ 10 microJoule + 1 microJoule

Now countermeasures

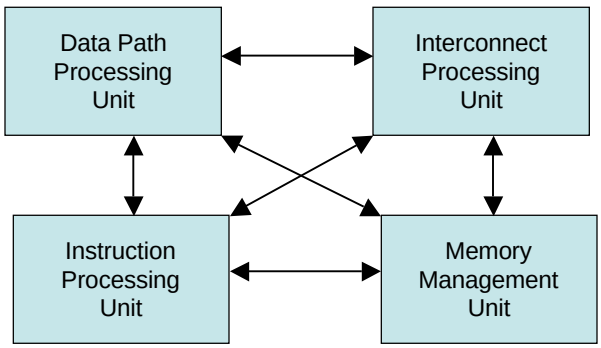
Limit to fault attacks

(Side-channel attack countermeasures is another story.)

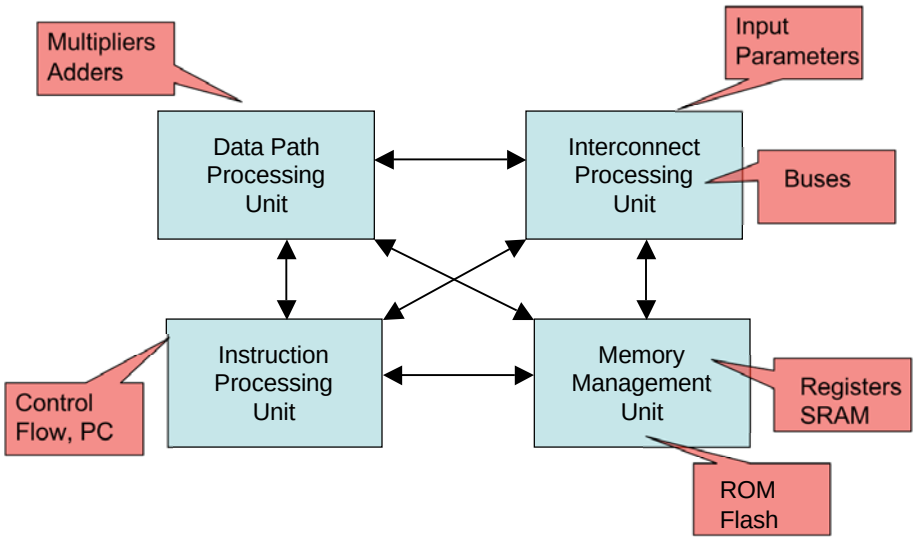
Fault attack classification

- Attack classification:
  - Target: Processor components
  - Precision
  - Duration
  - Exploitation
- Countermeasures:
  - Design abstraction level
  - Processor components

Processor components



Attacks on processor components

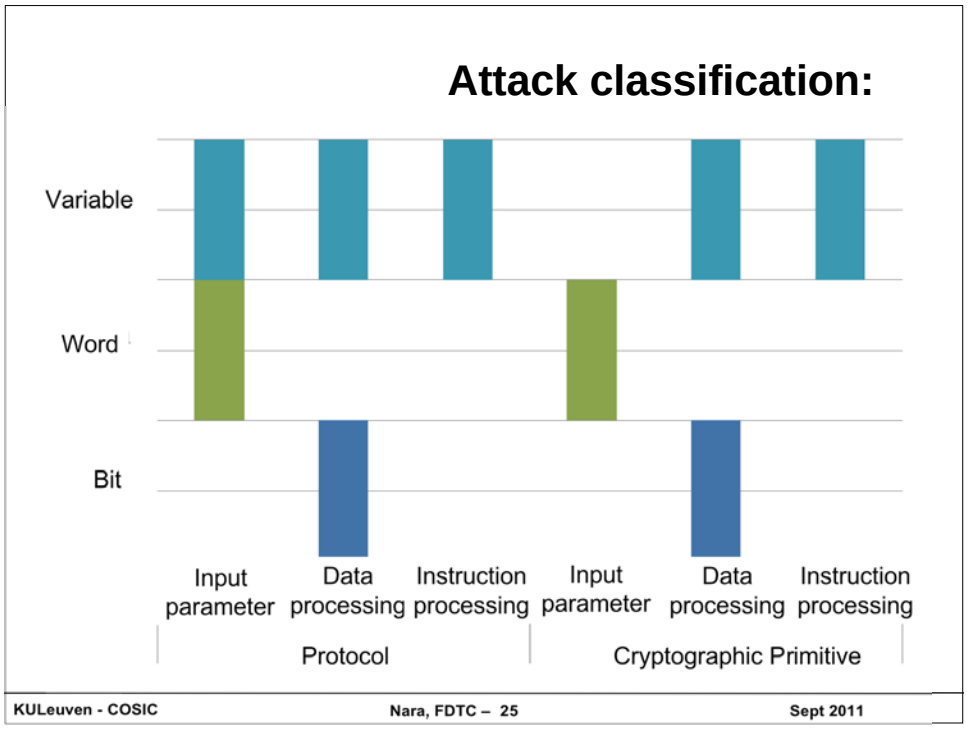


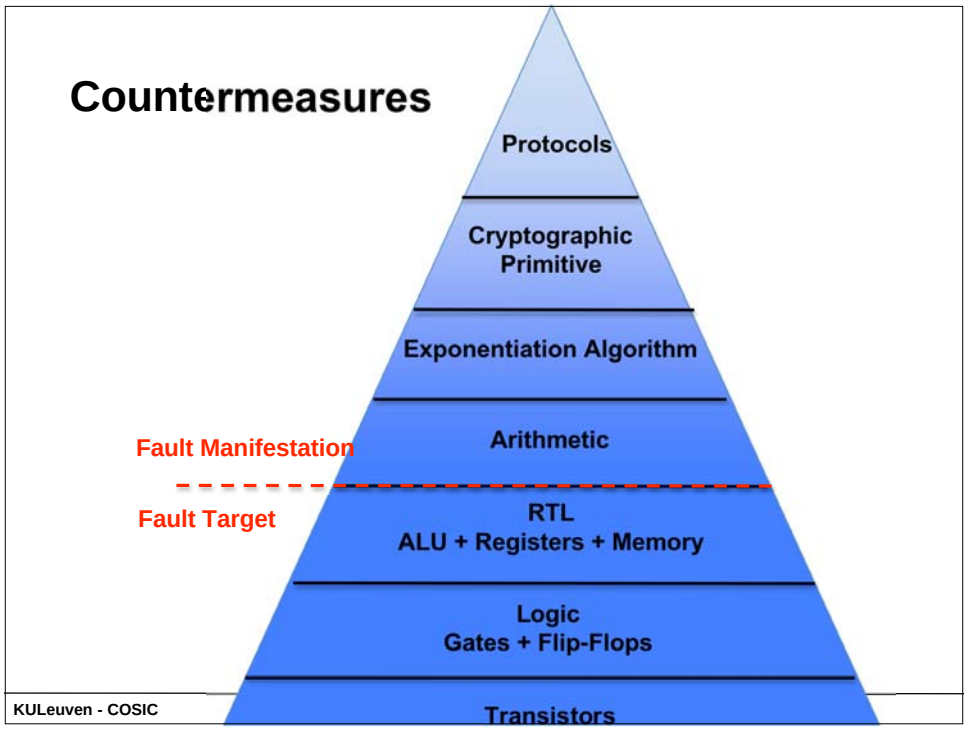
Precision

- Bit level:
  - One bit, any bit, very precise bit
- Word level
  - Word size of processor: 8, 16, 32, etc
  - Registers or busses
- Variable level:
  - Variable of algorithm
  - independent of machine word length

Time Classification

- Duration of fault attack
  - Permanent: laser or FIB
    effect over multiple cc
  - Transient attack: clock glitch
    time scale smaller than 1 cc
- Duration of existence of value/variable/target
  - Transient variable in multiplier or on bus
    lives for 1 cc or less
  - Permanent storage in register or memory
    lives for more than one cc
- Relation determines:
  - Effect: easy or difficult to mount attack
  - Detection: easy of difficult to detect attack


Attack classification:
Variable
Word
Bit
Input parameter
Data processing
Instruction processing
Input parameter
Data processing
Instruction processing
Protocol
Cryptographic Primitive
KULeuven - COSIC
Nara, FDTC – 25
Sept 2011


Countermeasures
Protocols
Cryptographic Primitive
Exponentiation Algorithm
Arithmetic
Fault Manifestation
Fault Target
RTL
ALU + Registers + Memory
Logic
Gates + Flip-Flops
Transistors
KULeuven - COSIC

Classification of countermeasures

- Processor part:
  - Interconnect: Check of input parameters
  - Data processing: Redundant and/or parallel computations
  - Control processing: Algorithm properties check
  - Storage: redundancy
- Abstraction level:
  - Protocol level
  - Cryptographic primitive level
  - Arithmetic level

Literature survey:

TABLE II
AN OVERVIEW OF COUNTERMEASURES AGAINST FAULT ATTACKS ON PUBLIC KEY CRYPTOGRAPHY

Mean of Protection	Check of Input Parameters [13]	Redundant Parallel Computations [14], [17], [37], [18], [6], [45], [31], [22], [33], [5], [15], [13], [29]	Algorithm Properties Check [10], [50], [19], [28], [40], [46], [36], [25], [32]
Abstraction Level	**Arithmetic Level** [14], [17], [37], [18], [6], [45], [31], [22], [33], [5], [15], [13], [29]	**Cryptographic Primitive Level** [10], [50], [19], [28], [40], [46], [36], [25], [32]	**Protocols Level** [34]

Goal: classify refs from literature

[see paper]

But problem is not solved

Passive SCA	Timing analysis
	Simple power analysis
	Differential power analysis
	Template attack
Active SCA	M safe-error
	C safe-error
	Invalid points
	Invalid curves
	Twist curves
	Sign-change attacks
	Differential faults

Attackers need only a single successful attack to win.

[source: Junfeng Fan]

Balanced PA/PD
Double-and-add-always
Montgomery Powering Ladder⊥
Base point blinding
Random projective coordinates
Randomized EC isomorphism
Randomized field isomorphism
Point validity check
Curve integrity check
Coherence check

Attacks vs. countermeasures

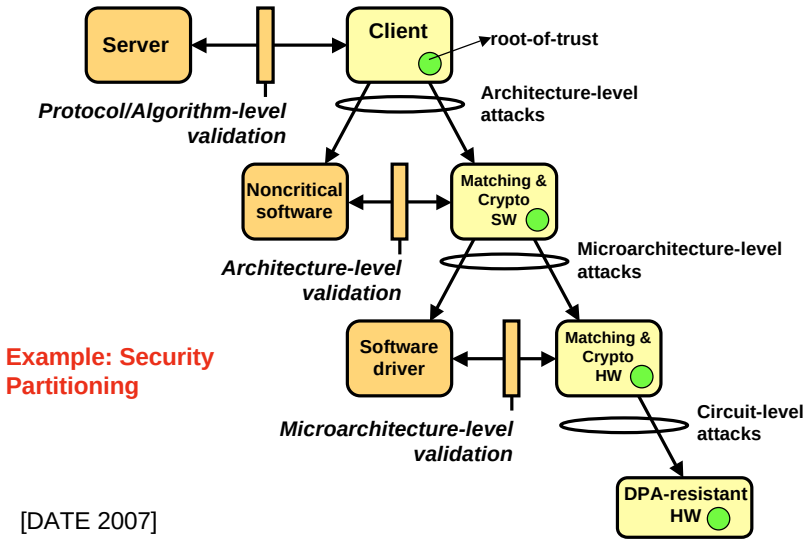
√ : Effective
× : Attacked
? : Unclear

-- : Irrelevant
H : helps the attack

[source: Junfeng Fan]

Countermeasures	Passive Attacks							Active Attacks						
								Safe-error		Weak curve			Differential	
	TA	SPA	Template	DPA	Comparative SCA	RPA/ZPA	Carry-based attack	M safe-error	C safe-error	Invalid point	Invalid curve	Twist curve	Sign change	Differential
Balanced PA/PD	√	√	--	--	?	--	--	--	--	--	--	--	--	--
Double-and-add-always	√	√	--	--	×	--	--	--	×H	--	--	--	--	--
Montgomery Powering Ladder⊥	√	√	--	--	×	×	--	√	√	--	--	H	√	--
Montgomery Powering Ladder⊤	√	√	--	--	×	×	--	√	√	--	--	√	--	--
Random scalar split	--	--	?	√	?	√	×	--	?	--	--	√	?	?
Scalar randomization	--	--	×	×	×	√	×	--	?	--	--	--	?	?
Base point blinding	--	--	×	×	×	√	--	--	--	?	--	--	--	?
Random projective coordinates	--	--	√	√	?	×	--	--	--	--	--	--	--	?
Randomized EC isomorphism	--	--	?	√	?	×	--	--	--	--	--	--	--	?
Randomized field isomorphism	--	--	?	√	?	×	--	--	--	--	--	--	--	?
Point validity check	--	--	--	--	--	--	--	--	H	√	?	√	H	√
Curve integrity check	--	--	--	--	--	--	--	--	--	?	√	--	--	--
Coherence check	--	--	--	--	--	--	--	--	H	--	?	--	√	√

Urgent Need for Design Methods



Example: Security Partitioning

[DATE 2007]

Conclusions

- Viewpoint from designer
- Protection against attacks is *only one* objective
- Extremely low power/low energy solutions are very hard to obtain
- First attempt at classification of fault attacks
- Design methods are needed:
  - Feasibility/cost of attacks
  - Cost vs. benefit of countermeasures
  - Integrate with rest of design flow