Side Channel Analysis on Embedded Systems

Job de Haas
Riscure

July 2, 2010

Who the hell is…

Job de Haas

- Electro engineer: 1990
- First exploit: 1991
- ITSX: 1998
- Riscure: 2006

Currently at Riscure

- Director Embedded Technology
- Testing security on: Set-top-boxes, mobile phones, smart cards, payment terminals, ADSL routers, VoIP modems, smart meters, airbag controllers, USB tokens, …
- BTW: Riscure is hiring!

Agenda

Scope

What is SCA on embedded?

How do you test for it in practice?

How to assess the strength of a product?

Embedded systems to consider

Microcontroller based

- USB sticks
- Car locks
- Remote access tokens

‘Complex’ processor based

- Mobile devices
- Game consoles
- Multi-media chipsets for pay-TV

Scope

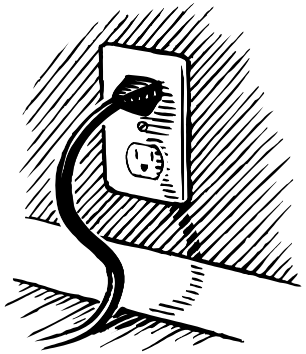
- Device: embedded systems with security functions
- Focus on passive side channels
- **Why?**
  - What is the threat from side channel analysis to embedded systems?
  - How does it compare with attacks on smart cards?
  - What are future developments?
  - Demonstrate side channel analysis.

What is SCA on embedded?

Attacking Side Channels

- Time

- Power consumption



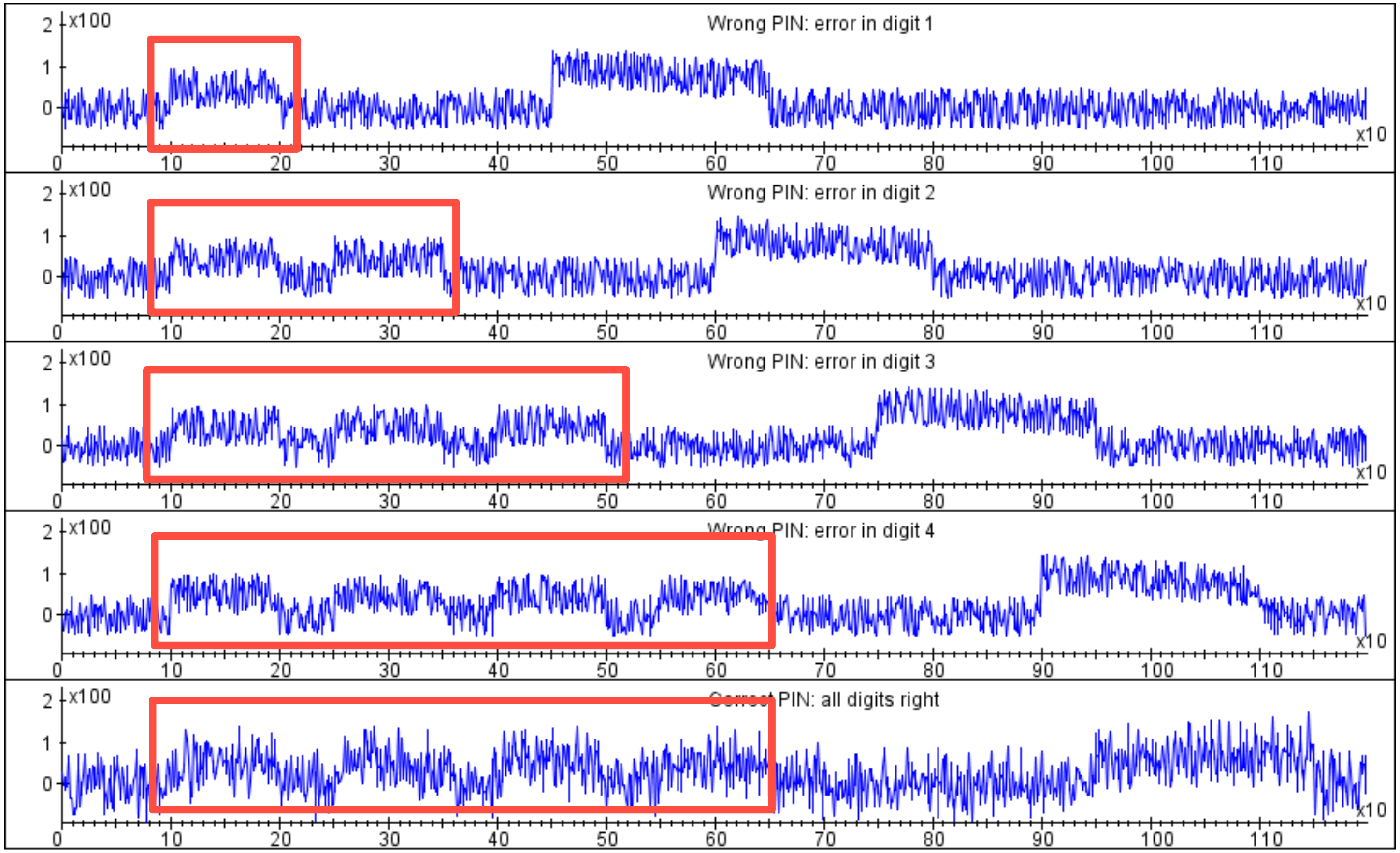
- Electro-Magnetic radiation
- Light

- Sound

Power / EM traces

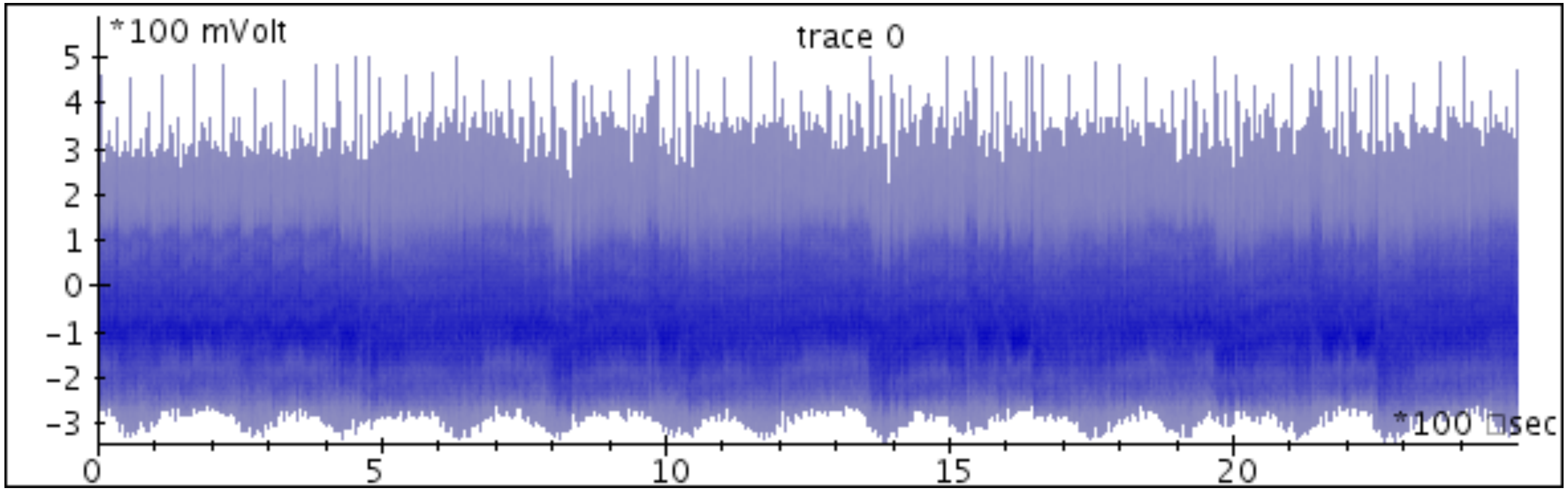
- Signal leakage from busses, registers, ALUs, etc

PIN verification attempts



Statistical data detection

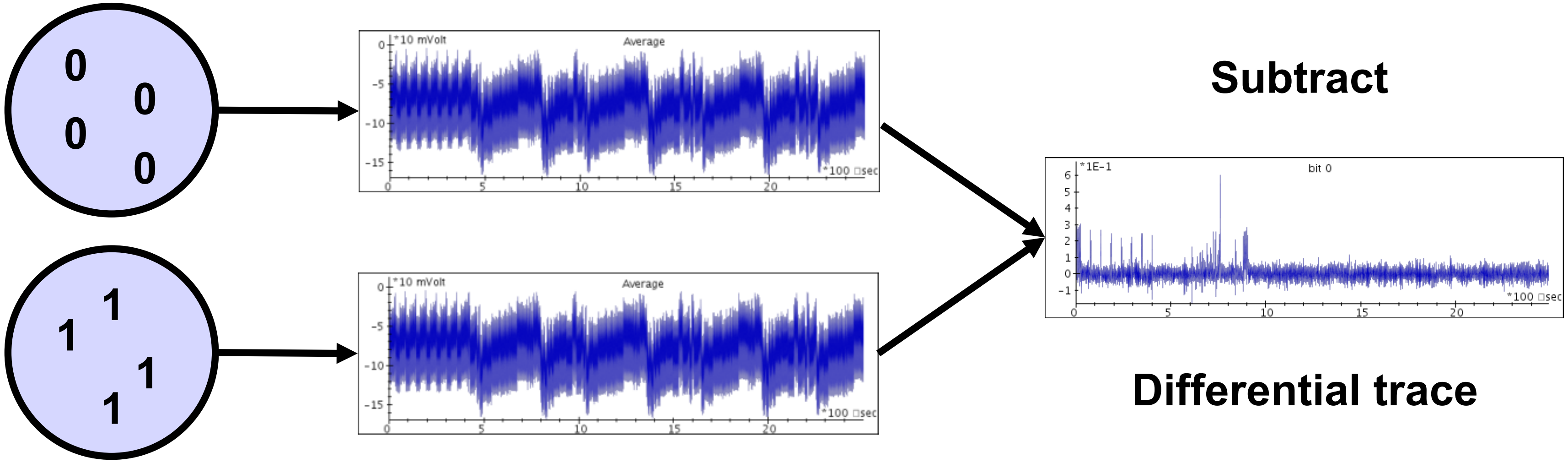
- Where is data processed in presence of noise?



Statistical data detection

- Where is data processed in presence of noise?
- Collect many traces with different data (n > 1000)
- Assume data values are:
  - known (e.g. algorithm input or output)
  - uniformly random (typical for crypto)
- We focus on one bit of one variable in the process

Group by known data **Average trace**

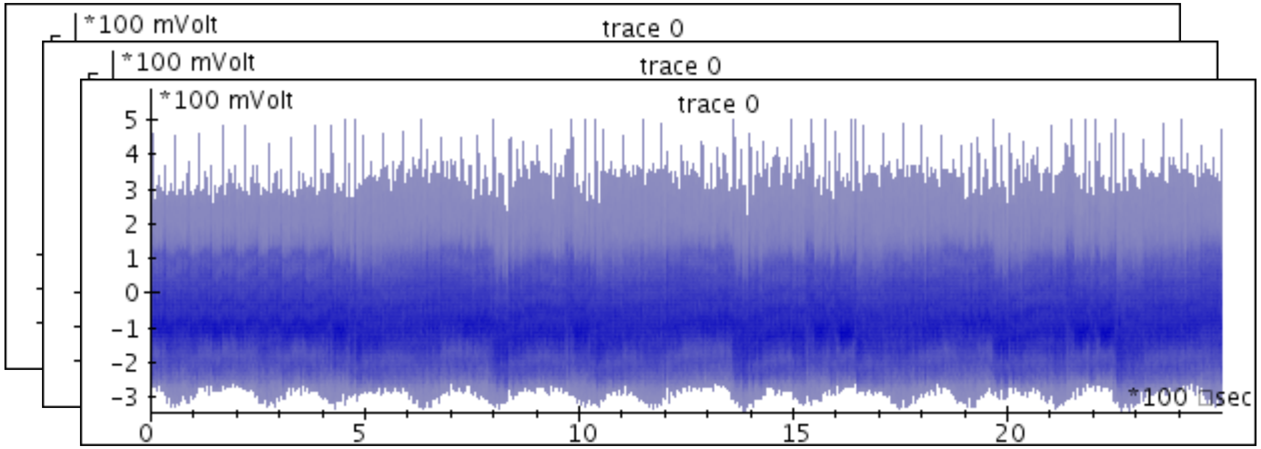


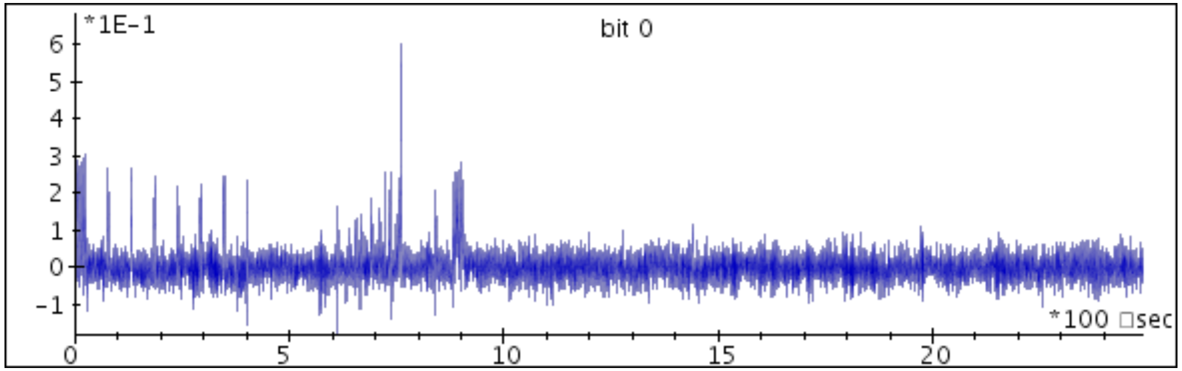
Subtract

Differential trace

Differential trace

- Input: *n* traces with known variable (e.g. input or output)
- Output: 1 trace with indication where bit causes trace differences





Purpose of SCA on embedded

Retrieve secrets

- Key
- PIN
- Unlock code

Reverse engineer

- Program flow
- Crypto protocol
- Algorithm

Not much changes …

When does SCA become interesting?

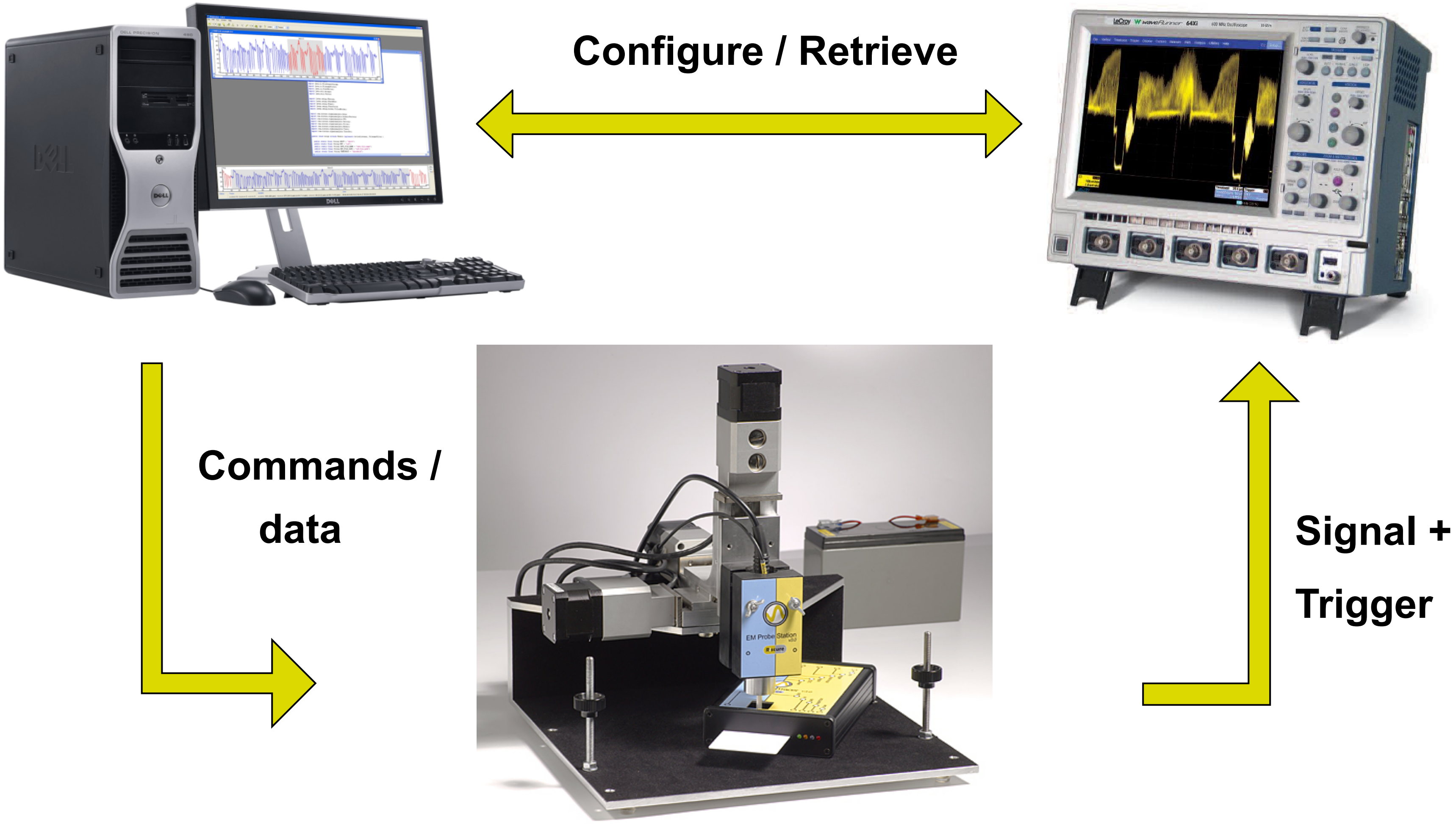
If side channel threats apply, depends on

- Physical access?
- Access time window?
- Interfacing and control?
- Exploitation equipment $?

A device becomes interesting when

- It contains a secret
- It contains a feature that can be unlocked
- Logical or physical access to internals is hard

Typical SCA set up



Configure / Retrieve

Commands / data

Signal + Trigger

Typical prerequisites

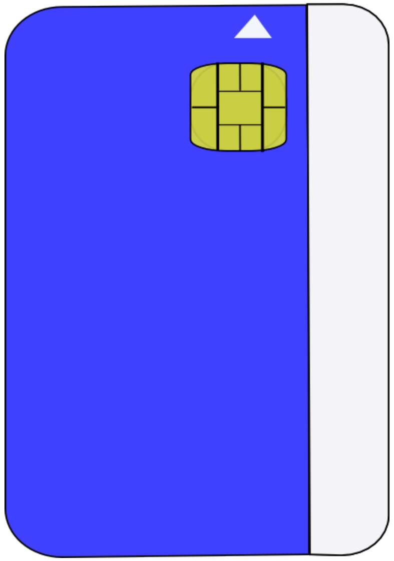
- ✓ Access to side channel
- ✓ Access to input or output data
- ✓ Minimize noise in side channel
- ✓ Time measurement of operation (trigger)
- ✓ Link data to operation

Comparing to smart cards

So far SCA testing centered on smart cards

A smart card:

- Standardized device
- Focus of SCA since its conception
- The benchmark of how SCA is rated



A smart card is an embedded system

... **But a very well defined one**

Processor comparison

	Smart card	Embedded
Processor complexity	Simple CPU next to crypto core	Complex processor with lots of peripheral next to crypto core(s)
Crypto core size	Significant compared to overall chip	tiny compared to overall chip
No. of crypto engines	One core per crypto operation	>10 cores for different purposes
SW or HW engine	Few SW implementations	Both HW and SW implementations
Countermeasures	Hardware and software countermeasures against leaking of both CPU and crypto core	No countermeasures and CPU leaks significantly

Acquisition comparison

	Smart card	Embedded
Power interface	Standard interface	Implemented on PCB with dedicated power supply
Triggering of acquisition	Standard interface allows controlled trigger	Trigger may be difficult without control over CPU
Flexibility of interfacing	Interface restricted	Control over CPU can often be gained through reverse engineering
Power consumption	Low power device (few mA)	Low to High power device (0.5A to 4A)
Clocks	Moderate clocks speeds (<50MHz), limited number	Moderate to high clock speeds, single or multiple clock domains
Sample preparation	Attacks are often noninvasive	Attacks mostly require invasive action

How do you test for it in practice?

Test versus attack

An attacker needs to turn a vulnerability into an exploit

A tester needs to gain insight in attacker cost efficiently

- How to create the optimal environment to discover a vulnerability?

General aspects

Controlling the crypto

Linking data with measurements

Efficiency of acquisition

Increased speed versus increased complexity

Timing analysis

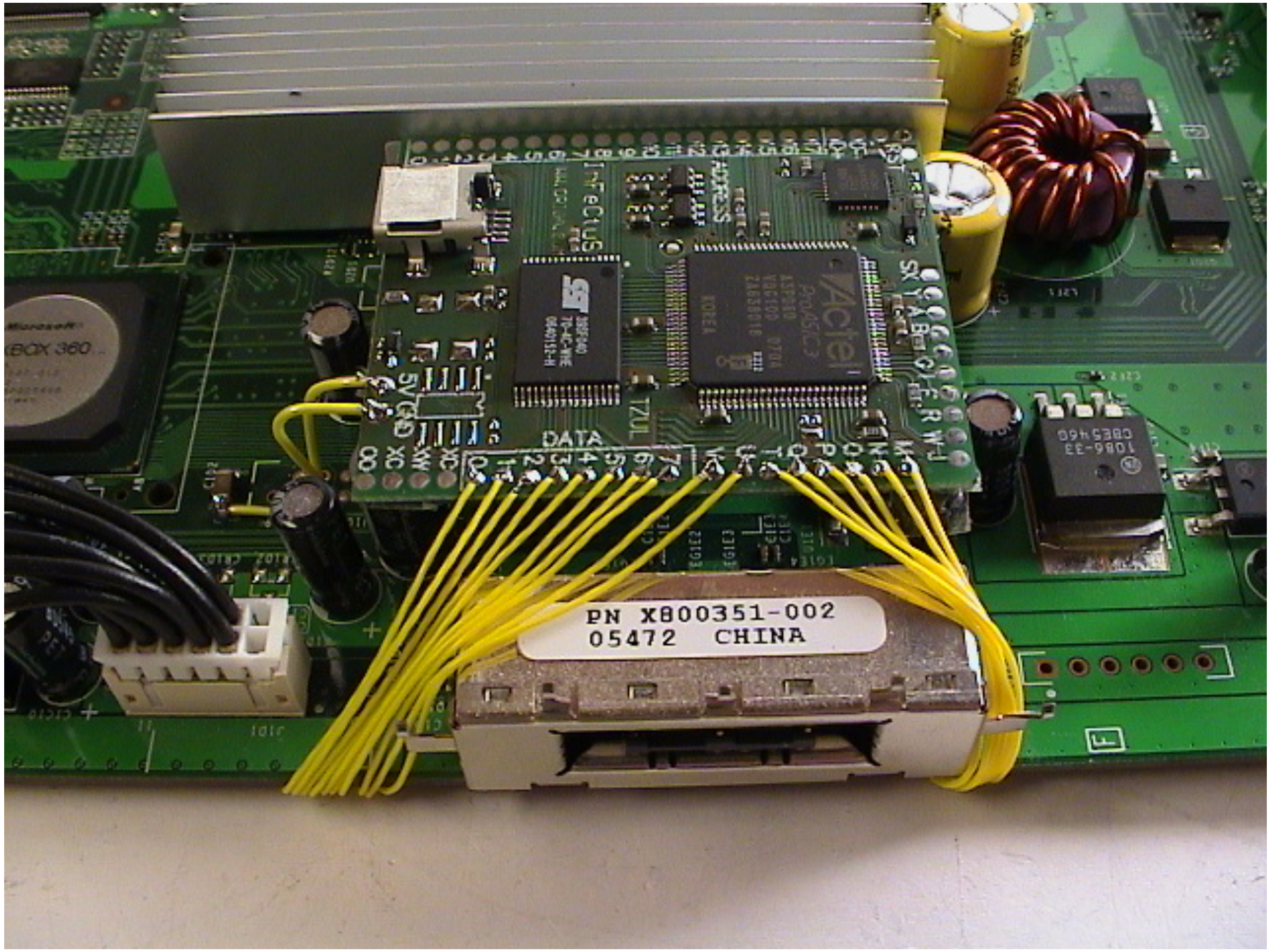
Peripheral outputs assist (example XBOX 360)

Exploiting runtime access (cache)

Increasing accuracy with EM and power

- Timing is a risk in many software implementations: both crypto and comparisons

XBOX 360 with Infectus board



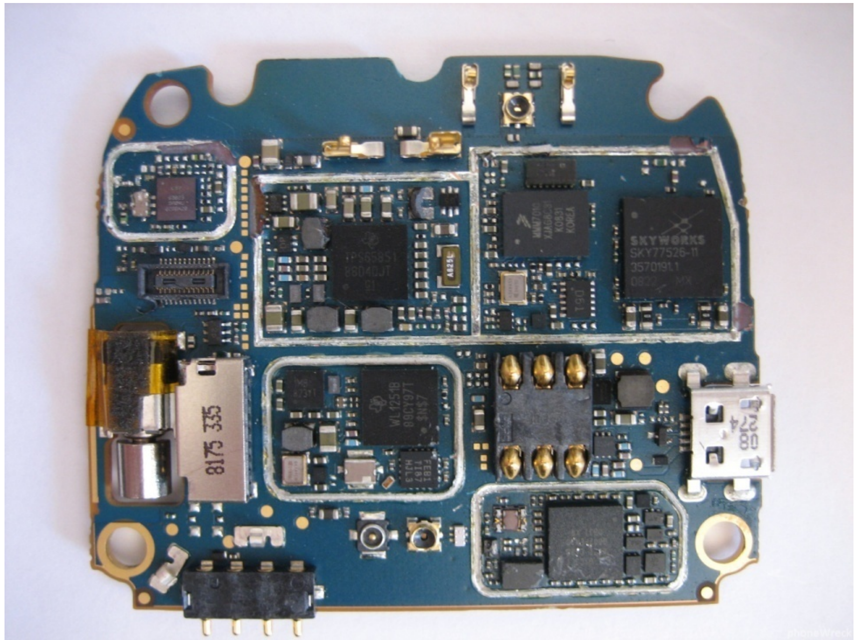
source: http://beta.ivancover.com

XBOX 360 timing attack

- XBOX 360 has a **secure boot** chain
- First boot loader security implemented with a HMAC-SHA1
- Sequence:
  - Hash secret key + boot loader with SHA1
  - Compare 16 bytes result with stored 16 bytes
- Comparison is per byte → **timing attack**
- Implementation in Infectus board:
  - It can modify stored HMAC-SHA1 value in NAND flash
  - Observes **timing** of diagnostic POST byte on PCB
  - Reset CPU with nTRST
- Brute forcing 16*128 = 2048 values on average takes **about 2 hrs**

source: http://www.xboxhacker.net

Power analysis



http://www.phonewreck.com

Tapping power or supplying it

Reaching rails

Identifying the correct supply rail

Disabling power domains

Disabling peripherals

- ➢ All require (more detailed) knowledge on target

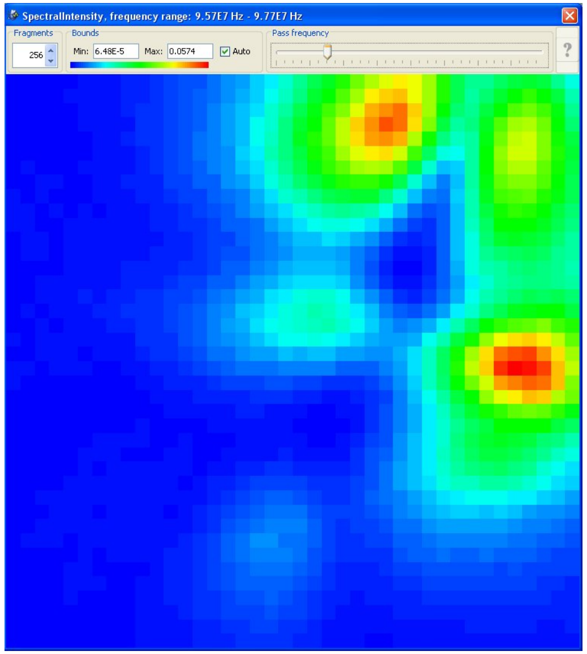
EM Analysis

EM signal adds dimension

How to locate?

When can EM be better?

EMA is an active research topic



SpectralIntensity, frequency range: 9.57E7 Hz - 9.77E7 Hz

Fragments: 256

Bounds: Min: 6.48E-5 Max: 0.0574 ☑ Auto

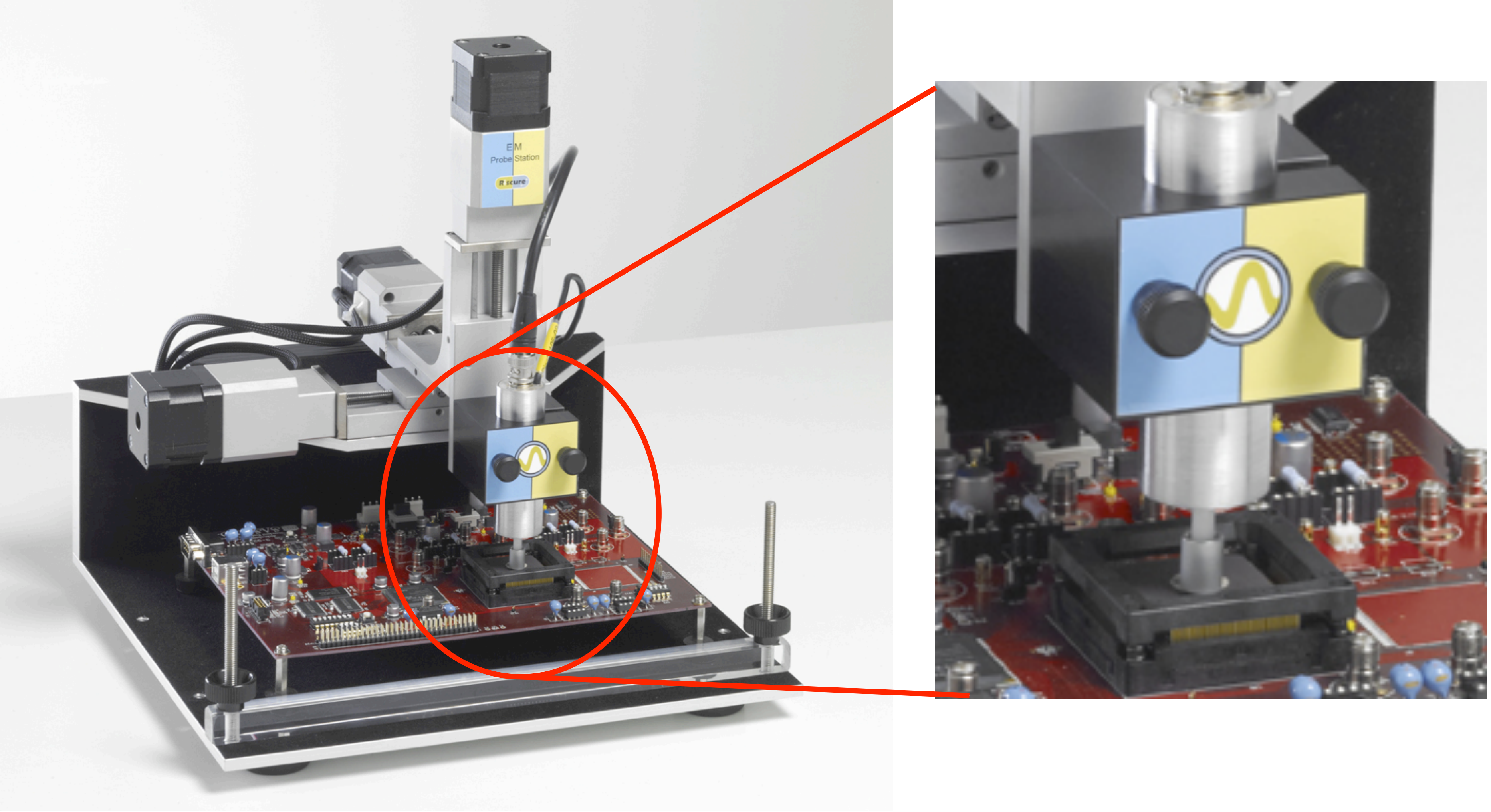
Pass frequency

- ➢ EM seems to add most when target operation is small relative to overall chip

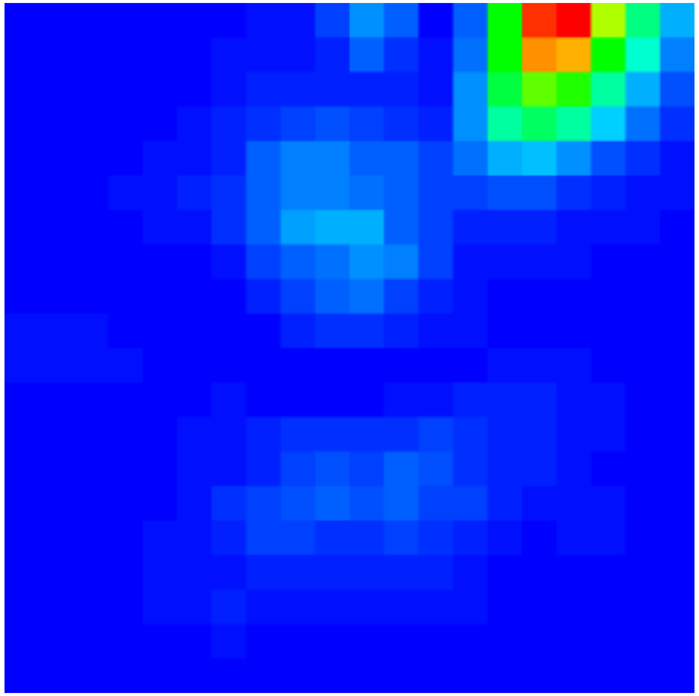
EM probe

- Probe is a **coil** for magnetic field
- Generally the **near field** (distance << λ) is most suitable

EM scanning DEMO

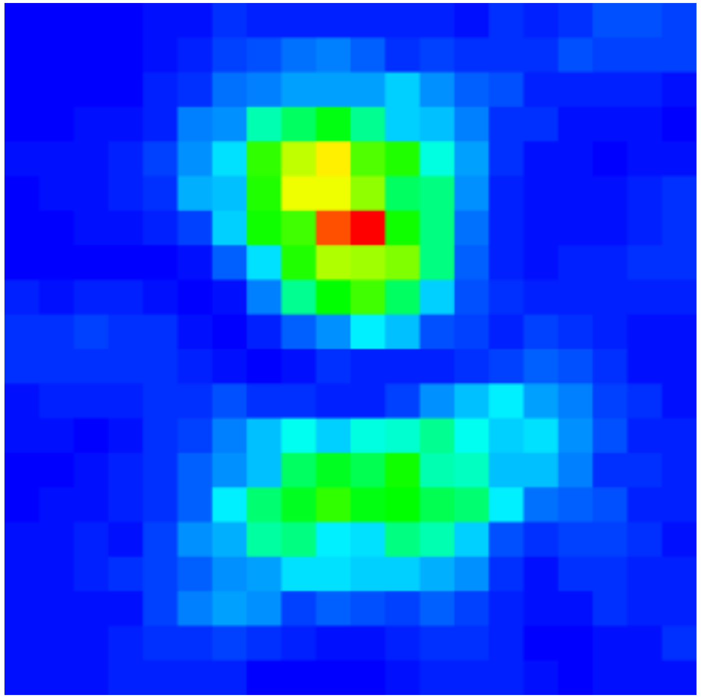


XY scan examples



Clock pin! (20MHz)

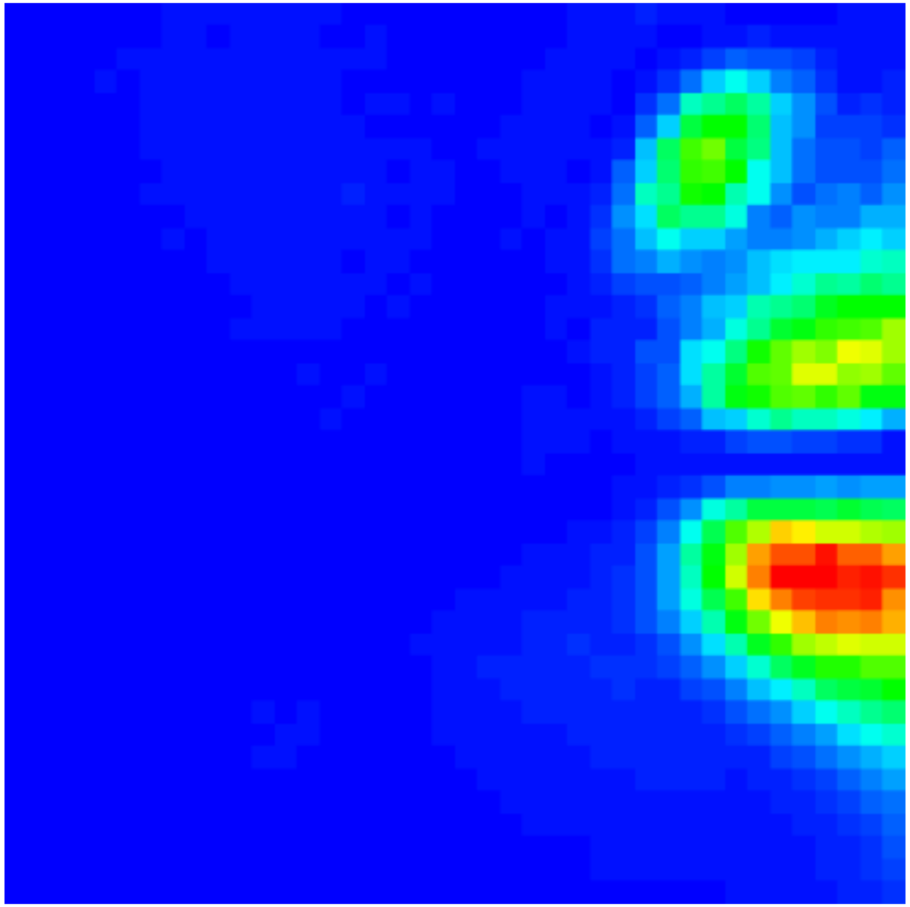
Full spectrum



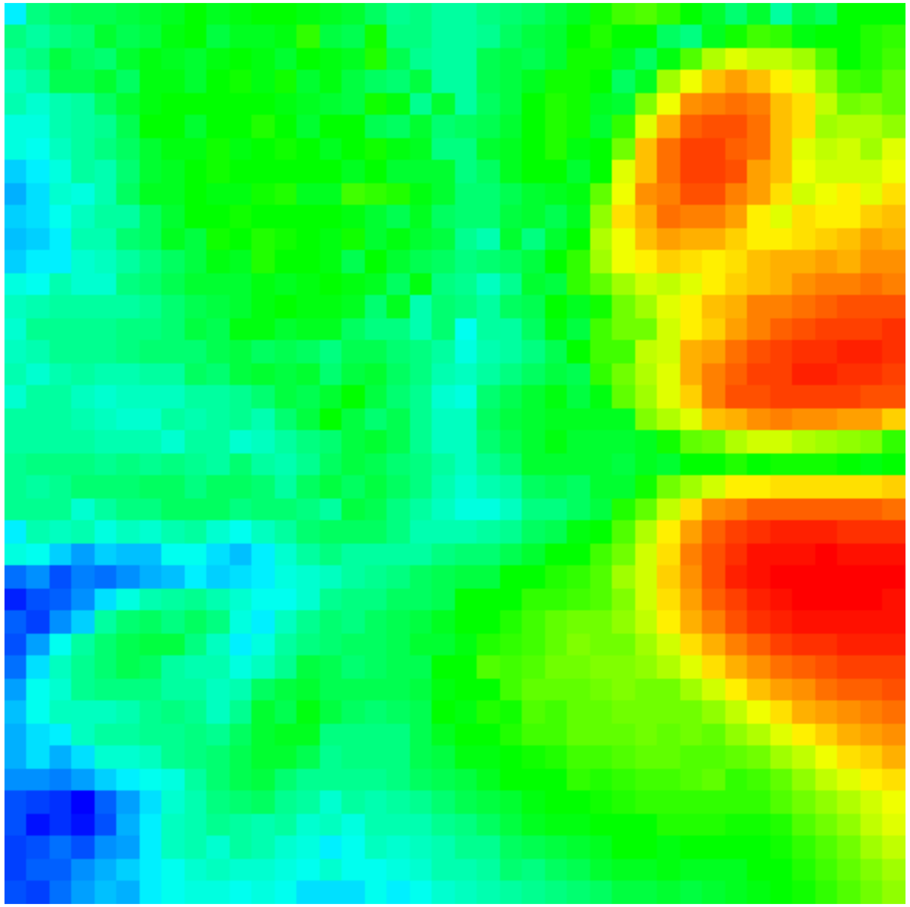
Around 40MHz

Scans above same chip running at 20MHz

Bonding wire effects

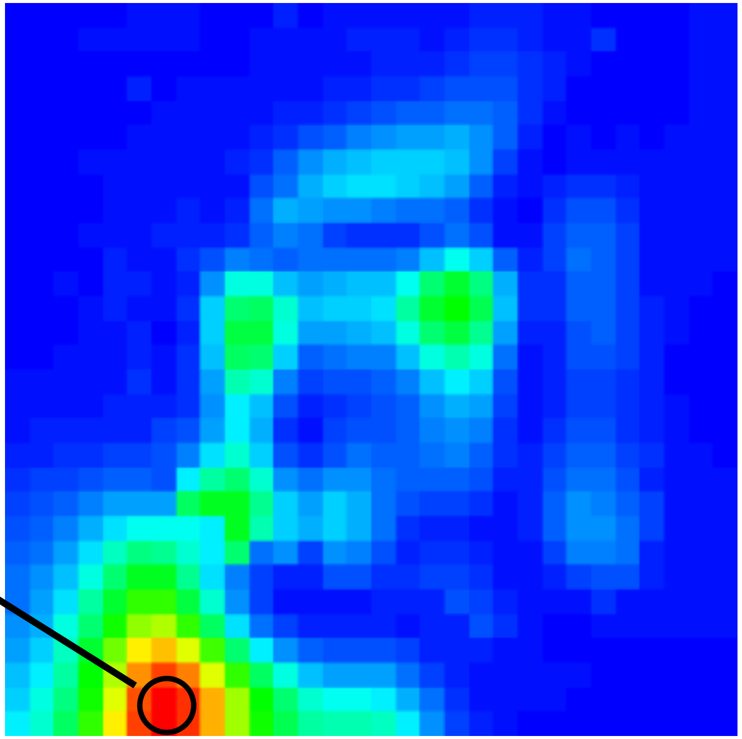


Full spectrum

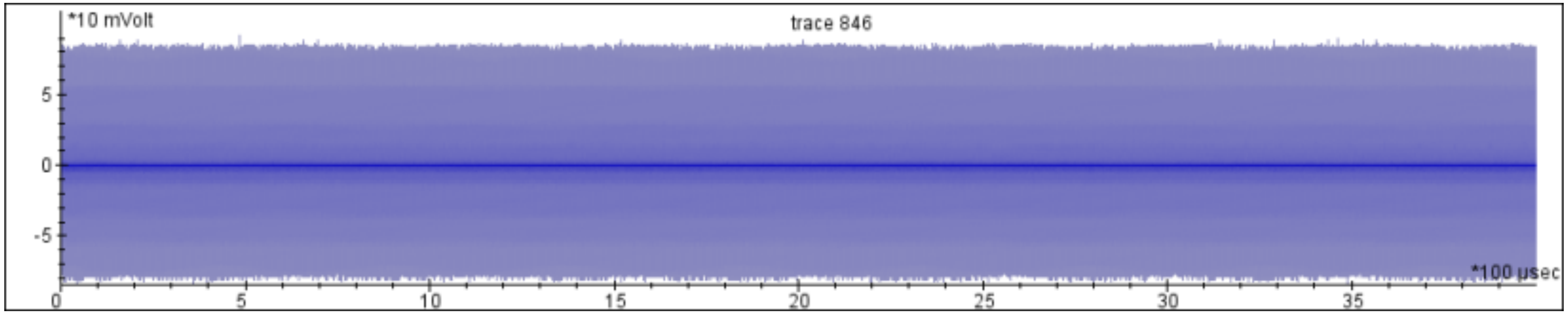


Full spectrum logarithmic

Hotspot is clock line



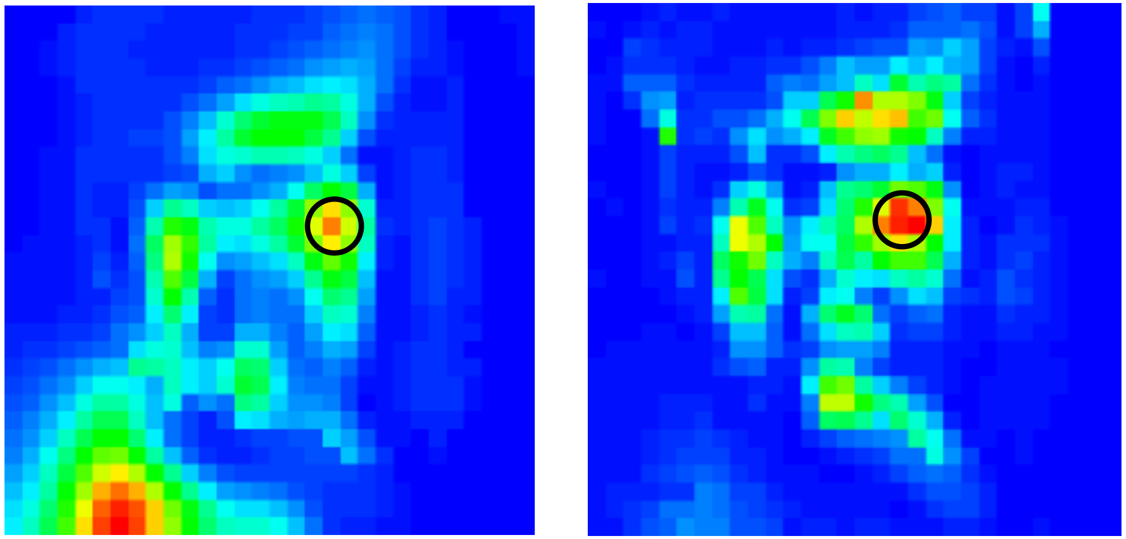
Corresponding trace shows no pattern: base clock

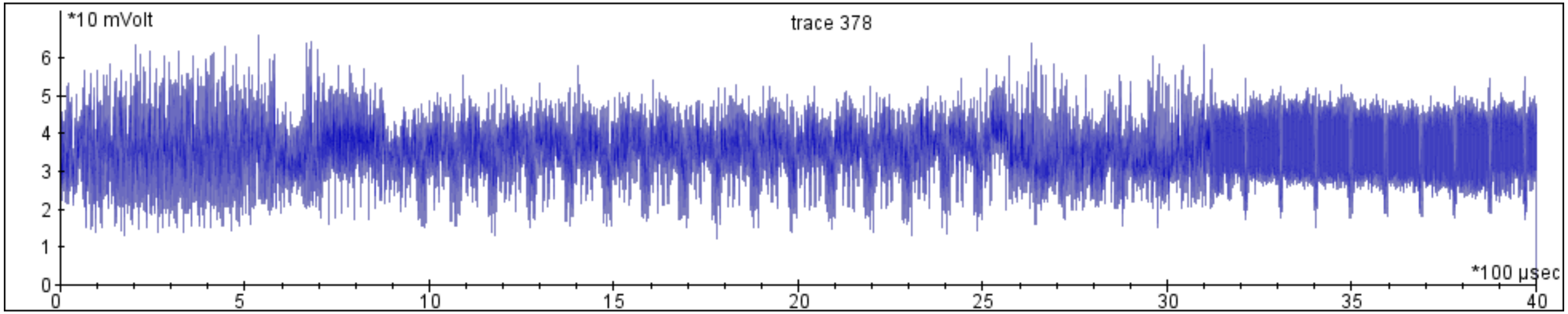


Hotspot after resampling

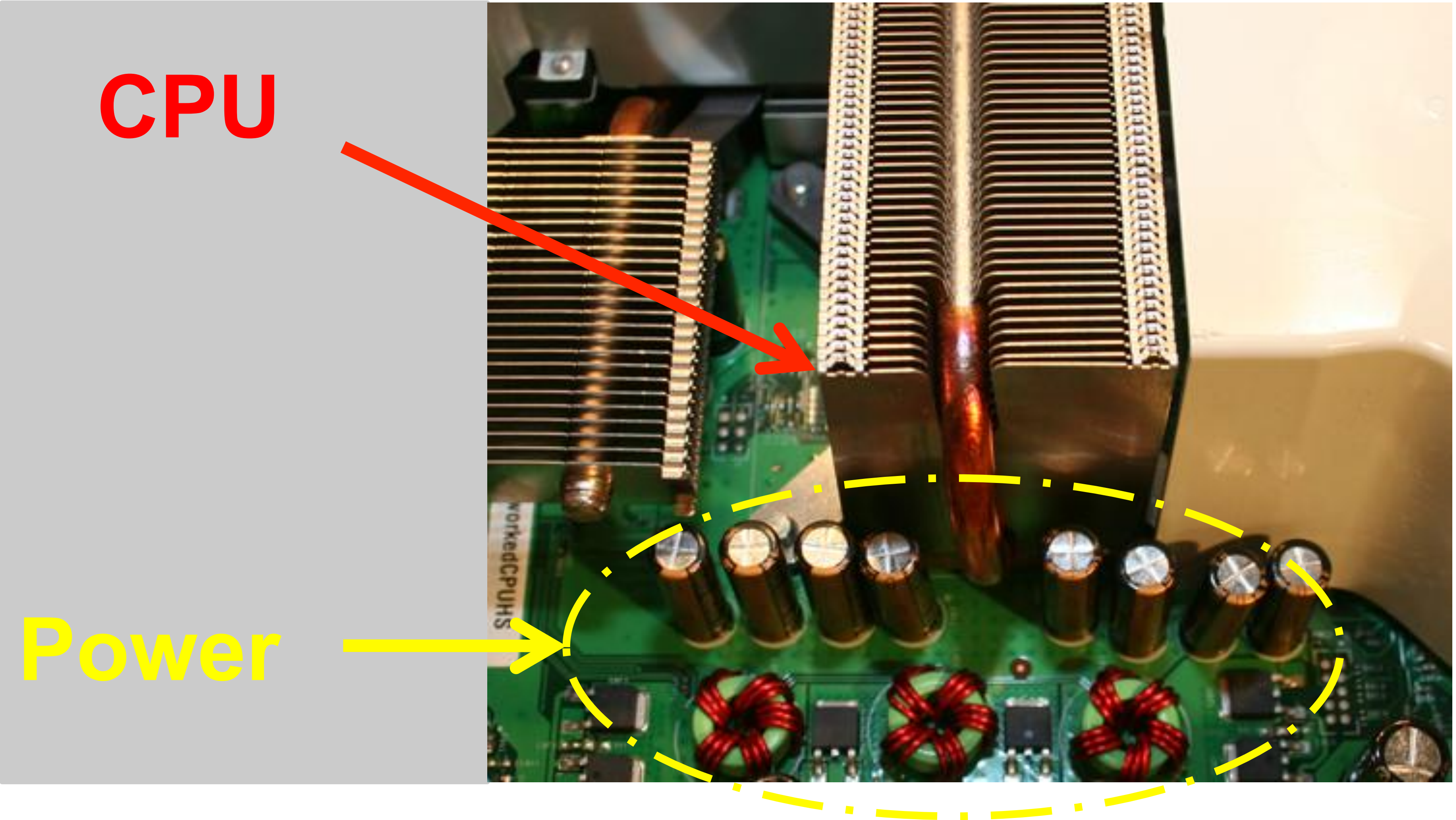
XY plot full spectrum (left); selected higher harmonic (right)

Trace shows pattern



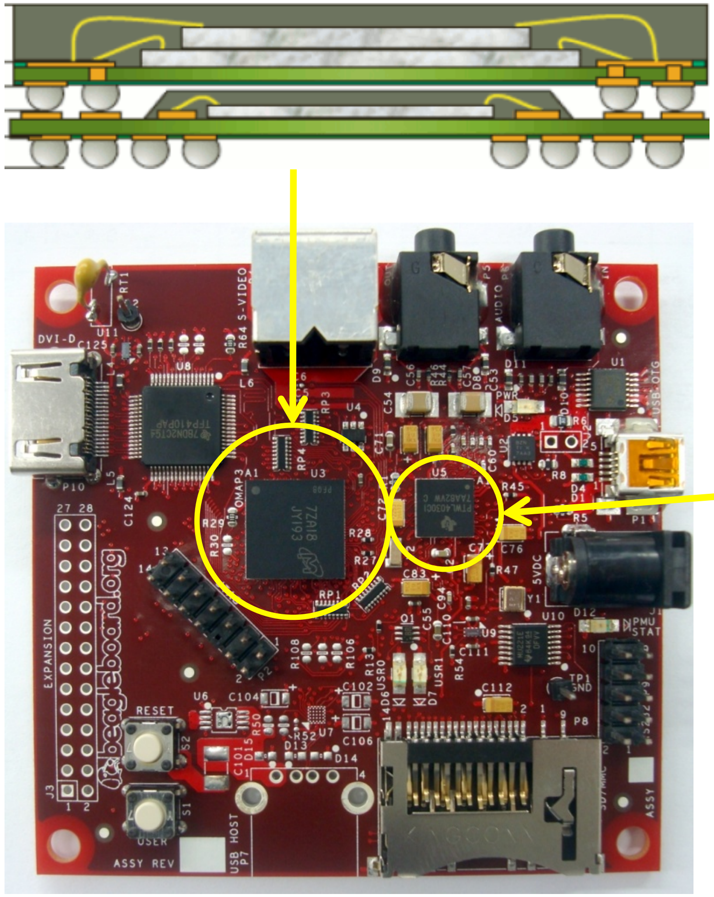


Practical encounters (1)

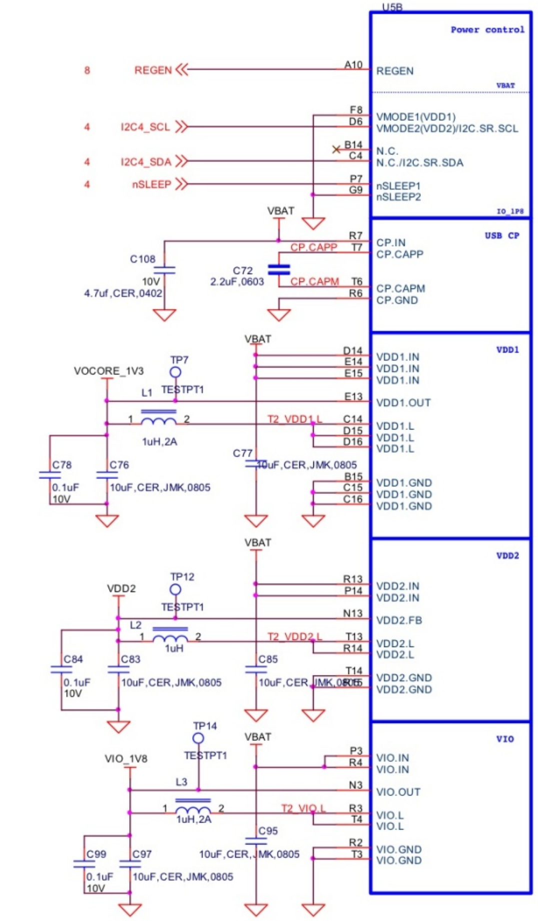


Practical encounters (2)

Package-on-package



Main power rails



How to assess the security strength of a product?

Threat and impact

SCA can break security functions, because:

- Few countermeasures
- Significant leakage
- Fast acquisition
- Examples in the field: Keeloq, …

However…

Effort can be considerable

Required level of control

Attacks needed to achieve control

High noise level, increased acquisition times

- ➢ Even without countermeasures,
  but countermeasures do improve this!

Countermeasures

Hardware

- Random Interrupts
- Data / key masking
- Shielding
- Balancing

Software

- Randomizing flow
- Blinding / masking
- Algorithm
- Protocol design

- ➢ Patented by Cryptography Research Inc (CRI)
- ➢ Licenses required and taken by major vendors (Infineon, NXP, Renesas, Samsung, …)
- ➢ Check with CRI

Side channel resistance

CPU type	Counter-measure	Effort (inc setup)	Skills	Strength
Basic microcontroller	No	1-2 weeks	SPA/DPA	0
Basic microcontroller	Basic	2-6 weeks	+ Adv sig proc	1
Complex processor	No	2-6 weeks	+ Adv sig proc	1
Complex processor	Basic	1-3 months	+ Adv sig proc	2
Both	Strong	>3 months	+ High order DPA	3

Note:
A complex processor with a bad RSA can still break in less than a week!
These are only indicators.

Developments

Side channel analysis related

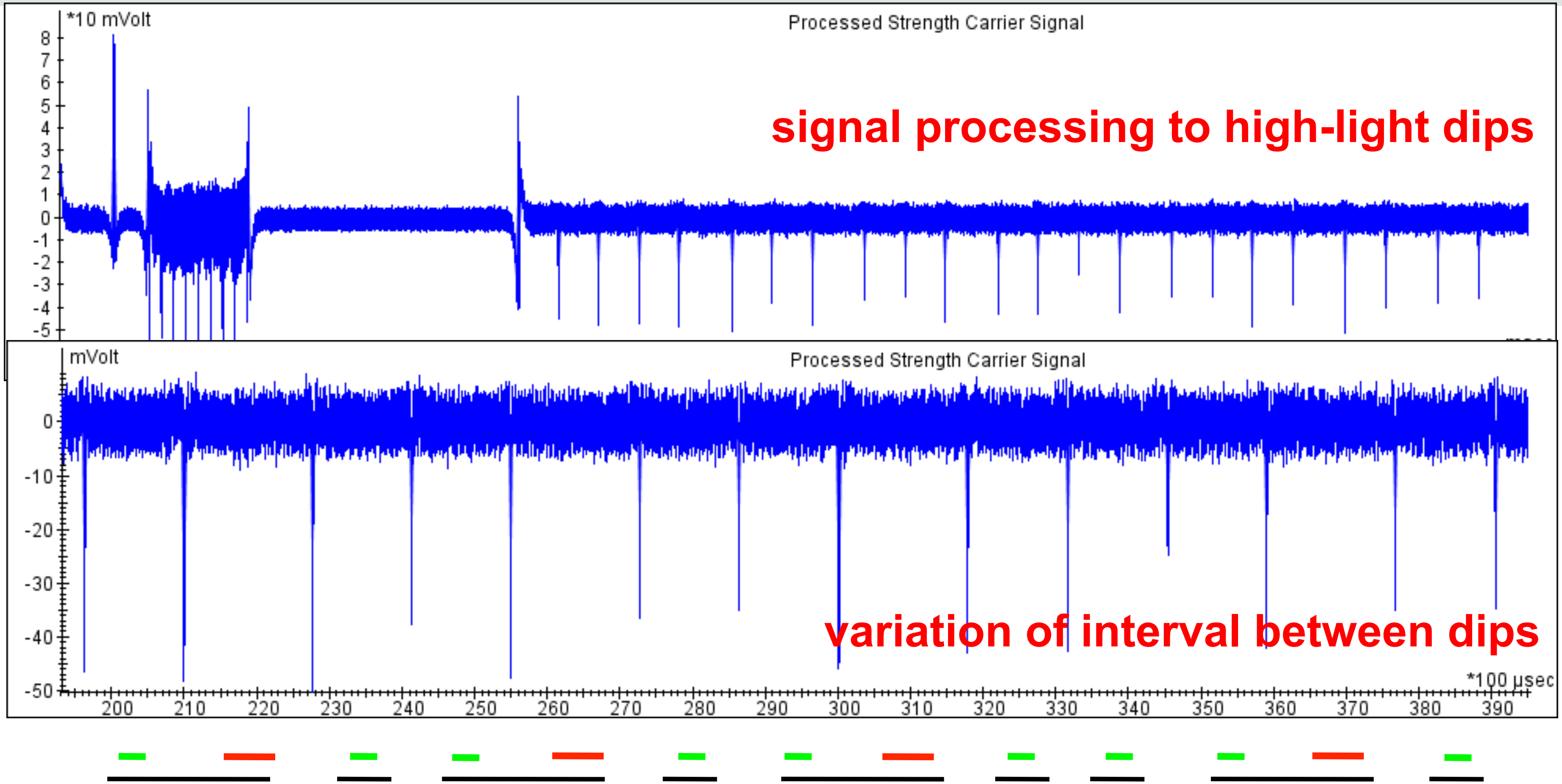
- Increasingly high speed acquisition
- Combined analysis of EM and power
- SCA becomes more mainstream
  - Tools
  - Techniques

Processor related

- More security features everywhere
- Basic countermeasure introduced

http://opensca.sf.net
http://www.dpacontest.org

Side Channel on RSA - DEMO



signal processing to high-light dips

variation of interval between dips

1 0 1 0 1 0 0 1 0

RSA implementation

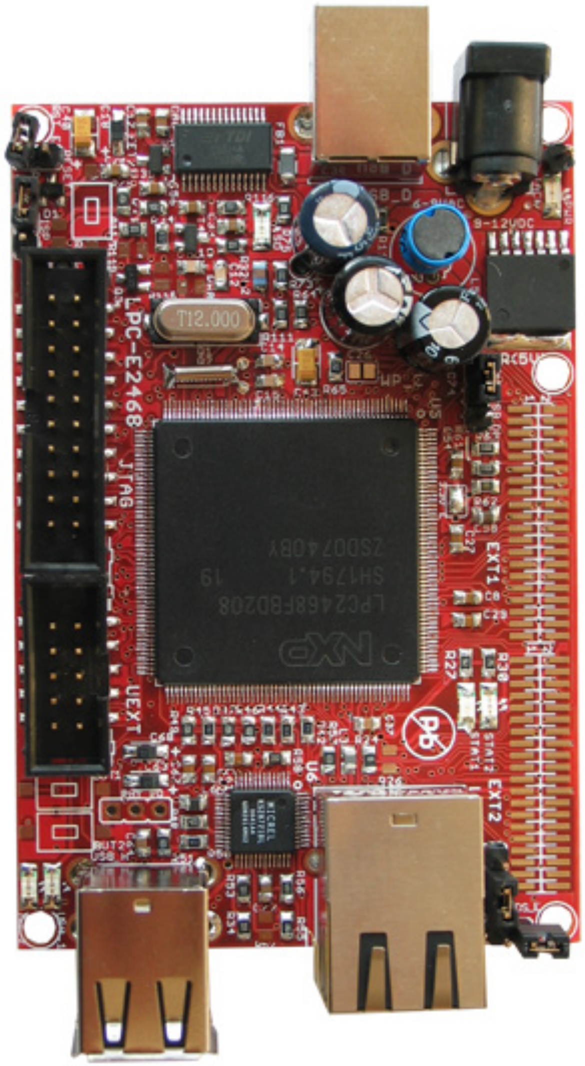
- Algorithm for $M=c^d$, with d_i is exponent bits ($0 \leq i \leq t$)

```
– M := 1
– For i from t down to 0 do:
    • M := M * M
    • If d_i = 1, then M := M*C
```

Demo Target

- Dev board with LPC2468
- ISP + JTAG can be locked
- Internal Flash for boot and storage
- Internal SRAM

- Can be moderately secured
- Running RSA with internal key



Conclusion

Findings

Embedded systems provide a different environment for SCA

- New obstacles for attackers: interfacing, noise, triggering
- Potential exposure due to: limited/no countermeasures, speed of acquisition, software implementations

Side channel is primarily a threat to

- Devices with basic microcontrollers
- High security devices that protect something very valuable

Recommendations

- To achieve strong protection against SCA, strong countermeasures must be added
- Demand countermeasures from manufacturers if you need the security level
- Do not rely solely on the hardware for protection
- Verify SCA protection if you need that security level

Looking for a job?

Riscure is Hiring!

Discussion

Thank you

Job de Haas

Director Embedded Technology

dehaas@riscure.com

Riscure B.V.
Frontier Building
Delftechpark 49
2628 XJ Delft
The Netherlands

Phone: +31 (0)15 251 4090
www.riscure.com

hack in the box
Keeping Knowledge Free

HITBSECCONF2010
AMSTERDAM
29th June - 2nd July 2010 :: http://conference.hackinthebox.nl/