RUHR-UNIVERSITÄT BOCHUM

FPGA-based Implementation Attacks with GIAnT

9th CryptArchi Workshop, Bochum

June 17th, 2011

David Oswald, Timo Kasper, Stephen Markhoff, Christof Paar

Chair for Embedded Security, Ruhr-University Bochum

hgi Horst-Görtz Institut für IT Sicherheit

EMSEC EmbeddedSecurity

Acknowledgements

- Timo Kasper
- Stephen Markhoff
- Christof Paar

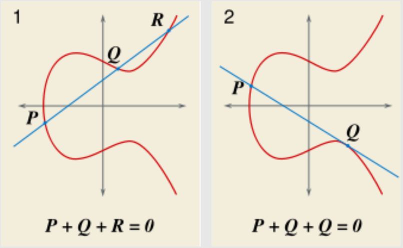
Outline of this talk

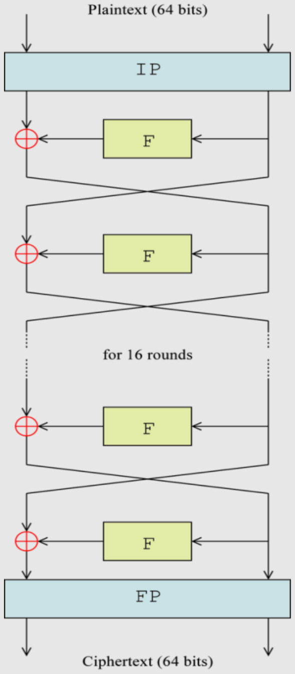
- Motivation
- GIAnT: Architecture and Features
- Practical Results
  - 3DES on ATXMega
  - RSA-CRT and AES on ATMega
- Live Demo
- Conclusion

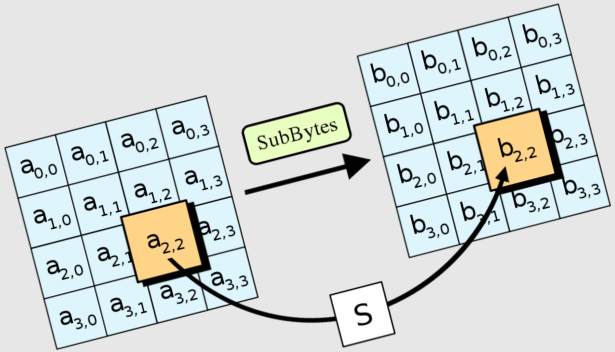
Motivation

Modern Cryptography

- E.g. AES, 3DES, RSA, ECC, …
- Mathematically secure
  ⇒ **No analytical attacks**
- Large key size
  ⇒ **No brute-force attacks**
- All problems solved?
- **No!** Crypto has to be implemented somewhere

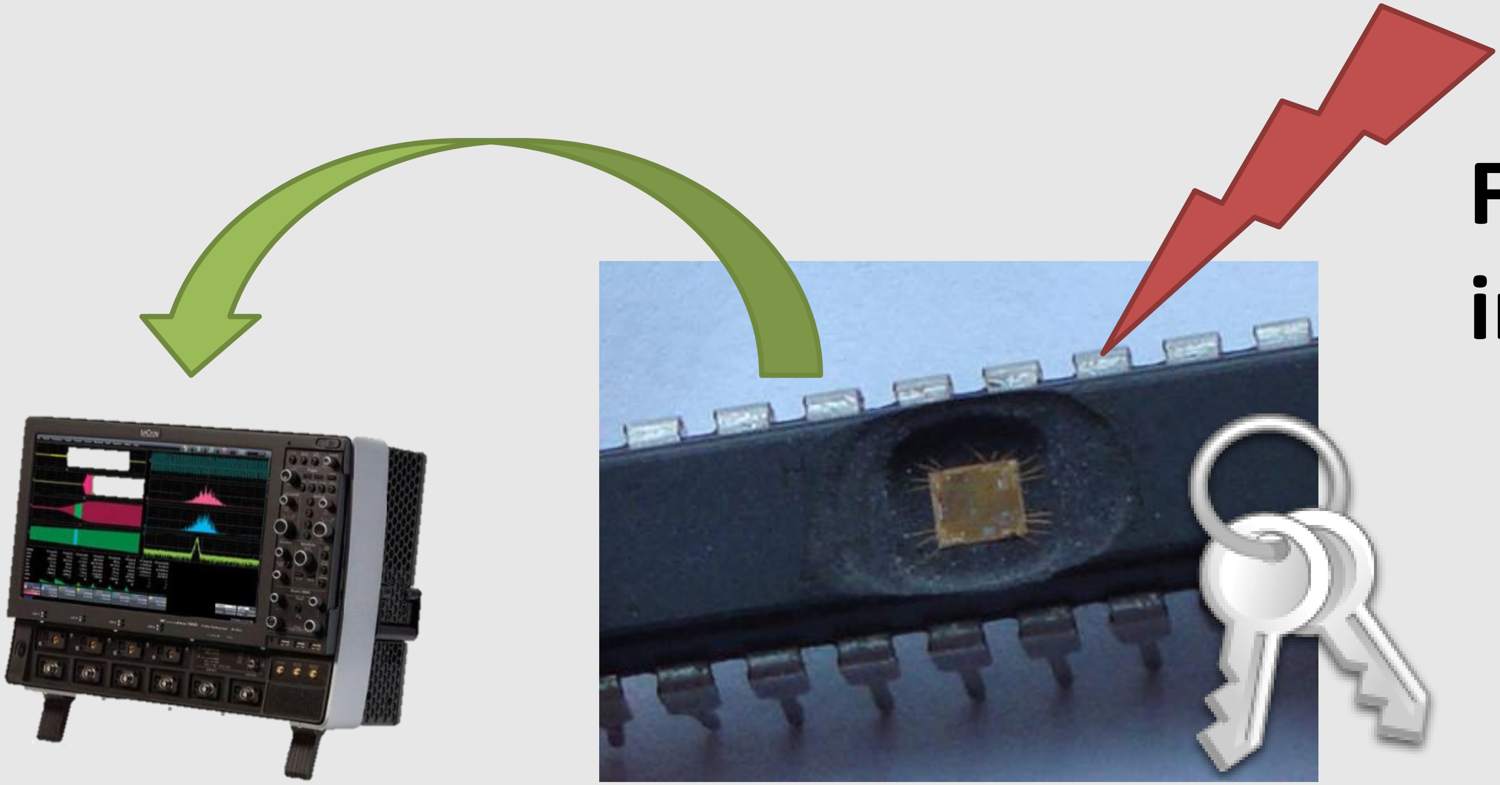





$a_{0,0}$ $a_{0,1}$ $a_{0,2}$ $a_{0,3}$
$a_{1,0}$ $a_{1,1}$ $a_{1,2}$ $a_{1,3}$
$a_{2,0}$ $a_{2,1}$ $a_{2,2}$ $a_{2,3}$
$a_{3,0}$ $a_{3,1}$ $a_{3,2}$ $a_{3,3}$
SubBytes
S
$b_{0,0}$ $b_{0,1}$ $b_{0,2}$ $b_{0,3}$
$b_{1,0}$ $b_{1,1}$ $b_{1,2}$ $b_{1,3}$
$b_{2,0}$ $b_{2,1}$ $b_{2,2}$ $b_{2,3}$
$b_{3,0}$ $b_{3,1}$ $b_{3,2}$ $b_{3,3}$

Source: Wikipedia

Implementation Attacks



Fault injection

Side-Channel Analysis

Cryptographic IC

Off-the-shelf Equipment

- Digital Oscilloscope: **2000 – 50000 USD**

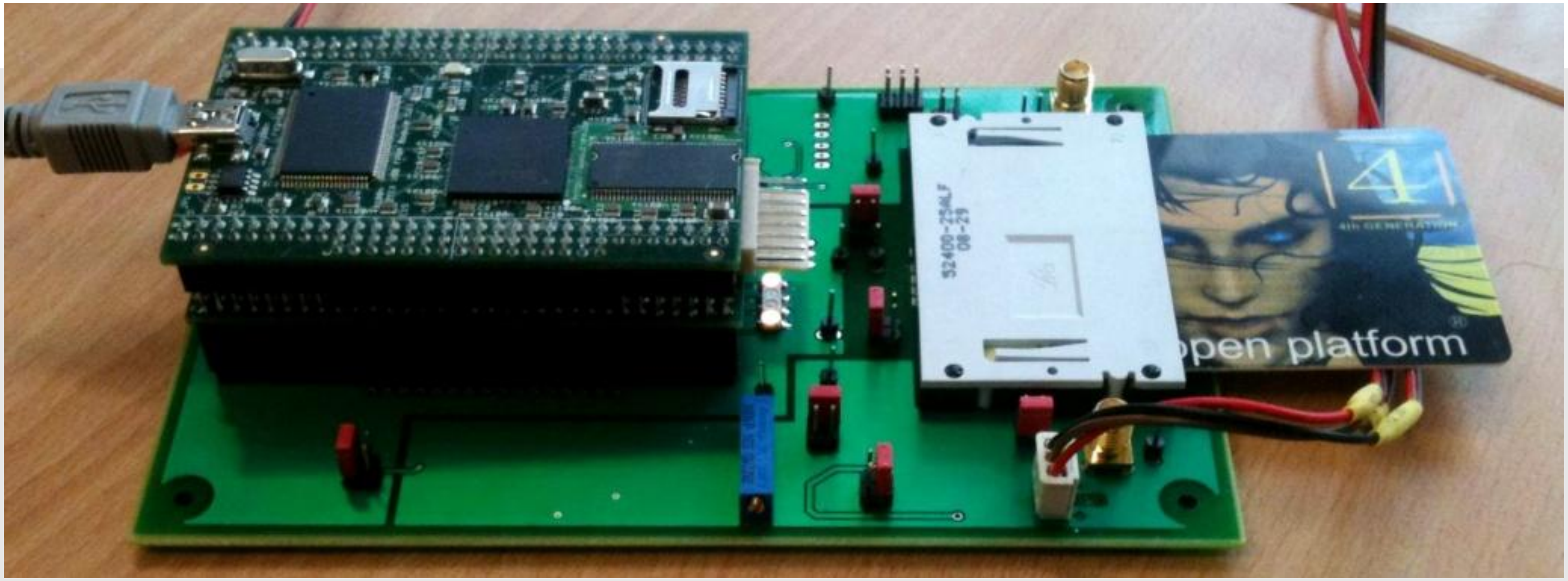
- Signal Generator: **2000 – 10000 USD**

- Specialized Devices:
  - E.g. by Riscure

Sources: LeCroy, Agilent, Riscure

- **Expensive**
- Usually **not fully open / extendable**

Our contribution: The GIAnT

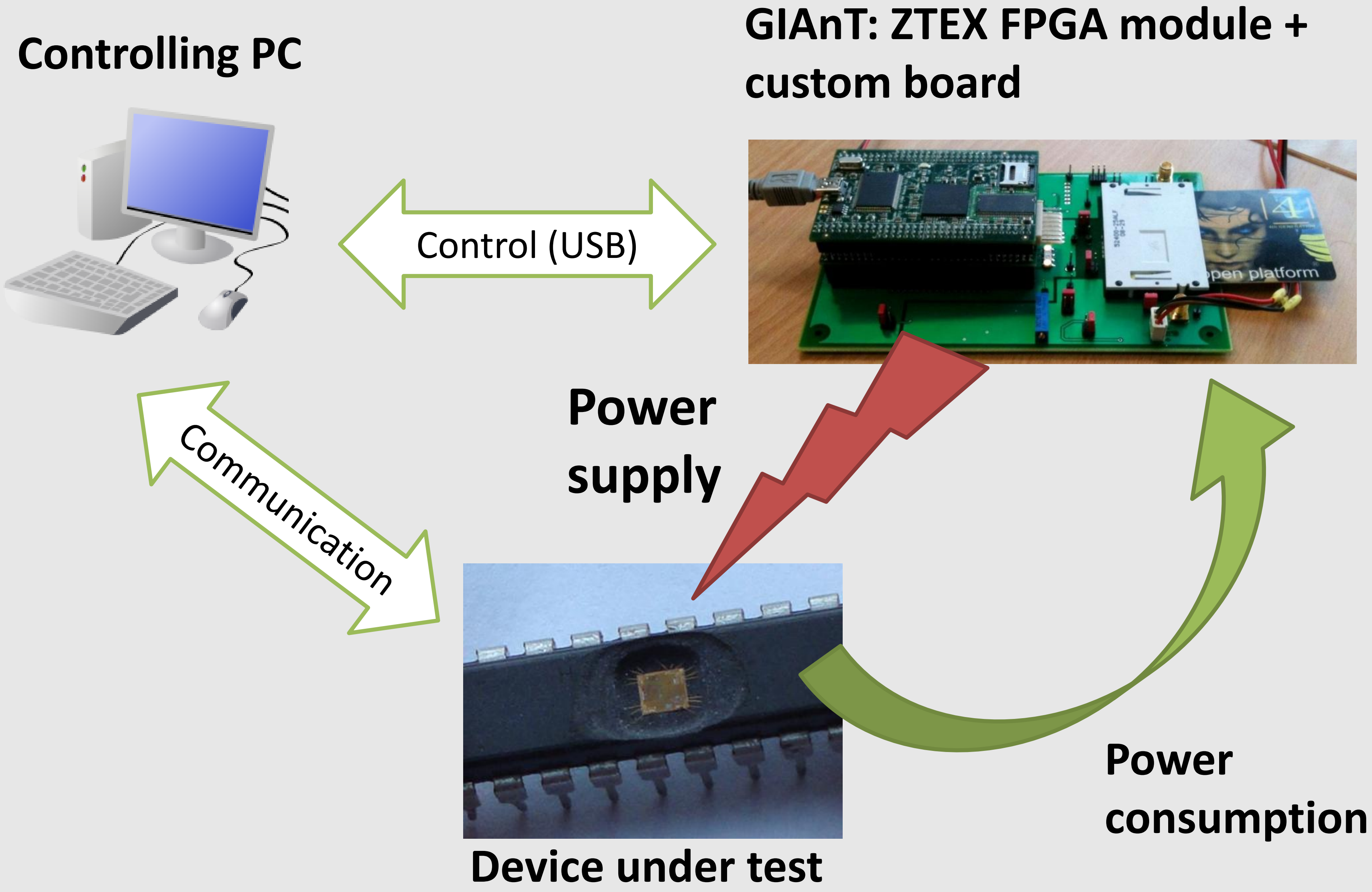


- **G**eneric **I**mplementation **An**alysis **T**oolkit
- **Low-cost**: < 300 USD
- **FPGA-based** (Spartan 6)
- **Open-source**: sf.net/projects/giant
- Support for fault injection and side-channel analysis

Architecture and Features

GIAnT

Typical Setup: Overview



Controlling PC

GIAnT: ZTEX FPGA module + custom board

Control (USB)

Communication

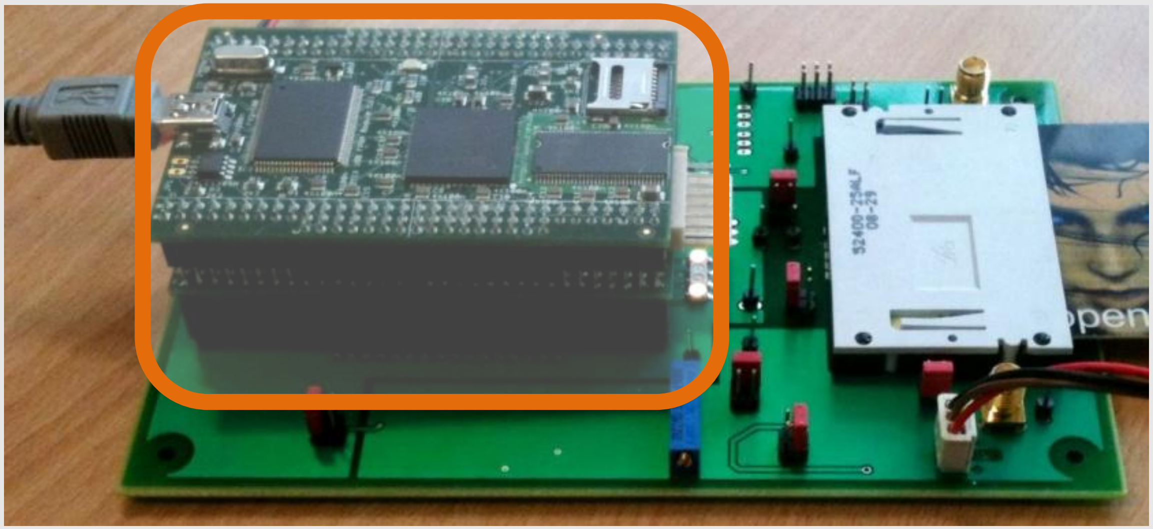
Power supply

Power consumption

Device under test

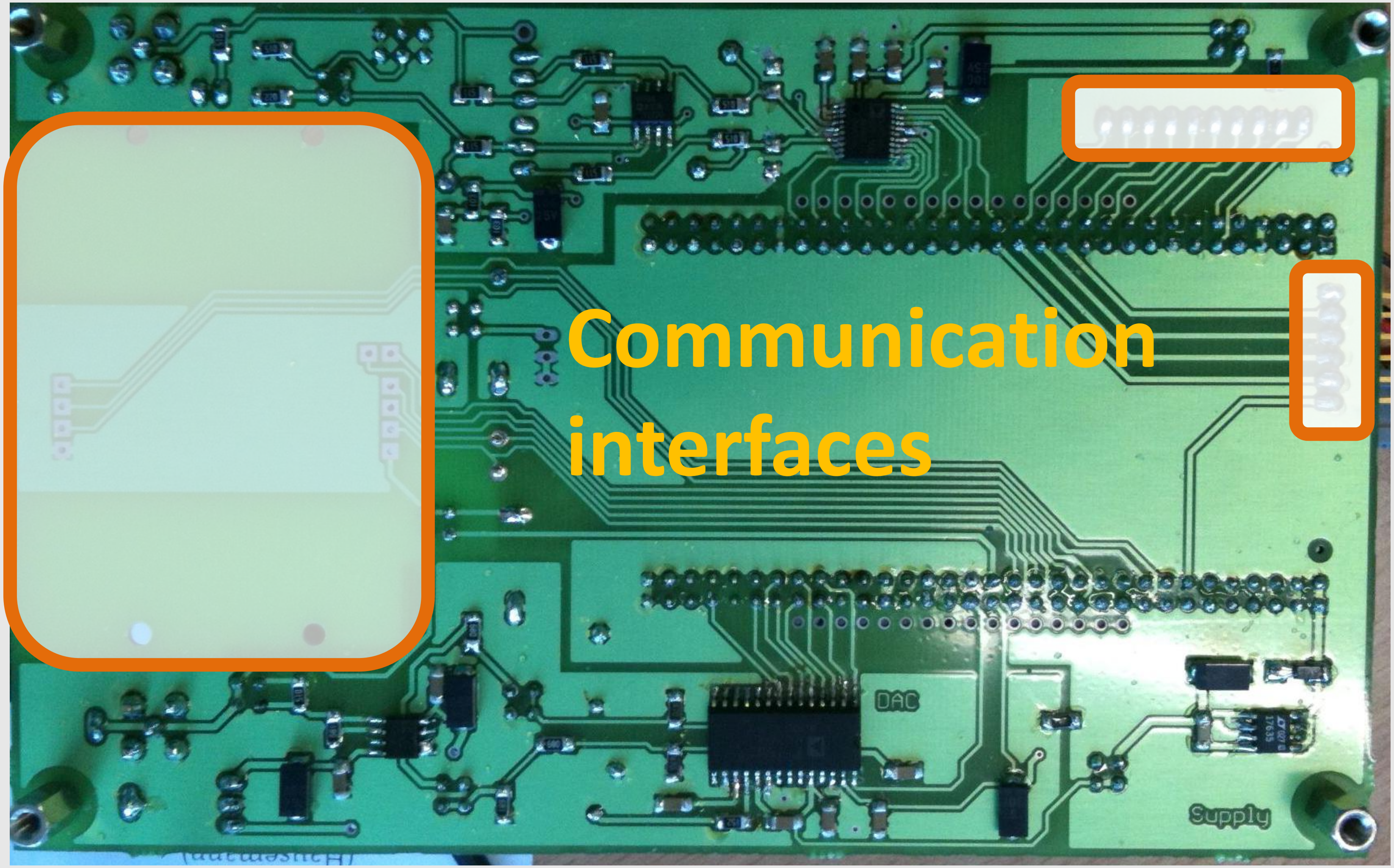
GIAnT: Structure

- **GIAnT** = ZTEX Spartan 6 module + custom board



- ZTEX Spartan 6 module: www.ztex.de
  - Additional µC for USB 2.0 link
  - FPGA power supply
  - 64 MB SRAM

GIAnT: Hardware Overview

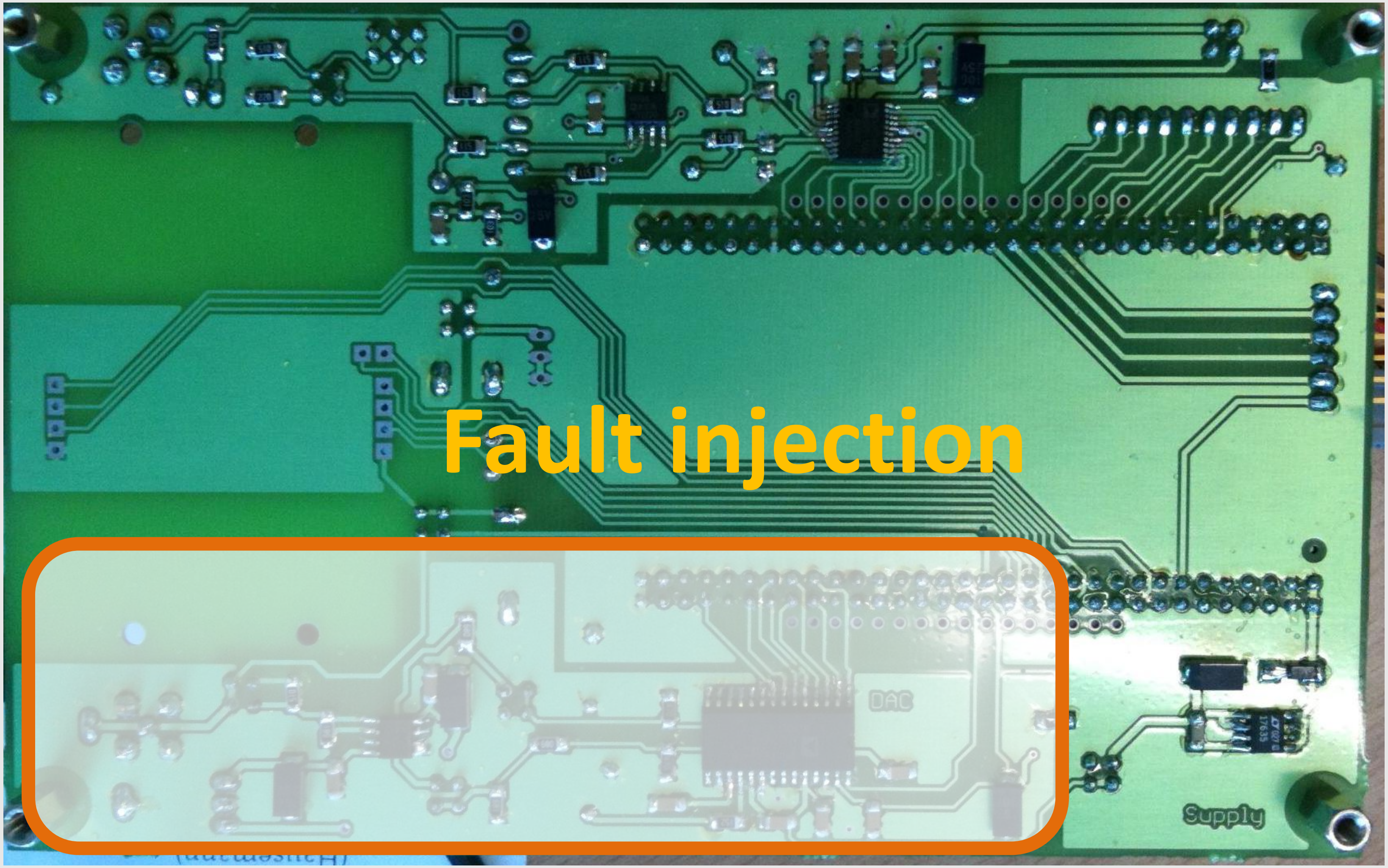


Communication interfaces

Communication Interfaces

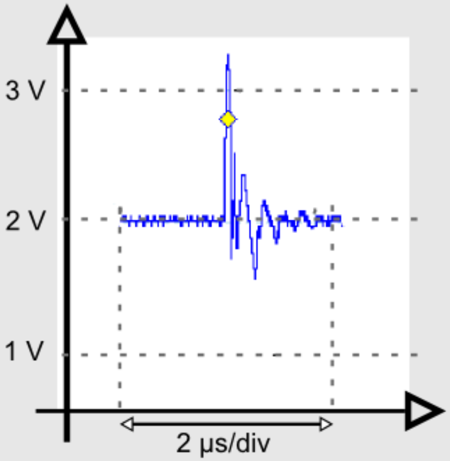
- Controlled and programmed via USB 2.0
- Interfaces to DUT
  - General-purpose I/O
  - Serial links (SPI, TWI, …)
  - ISO 7816 (Contact-based smartcards)
  - ISO 14443 (Contactless smartcards)
  - …

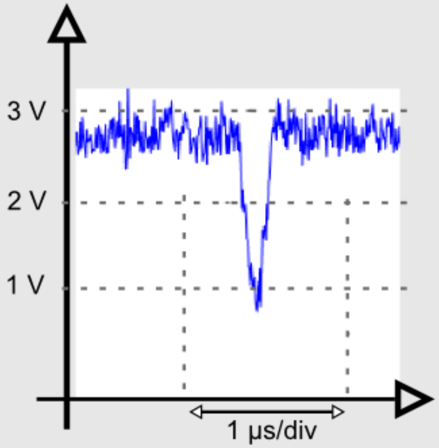
GIAnT: Hardware Overview

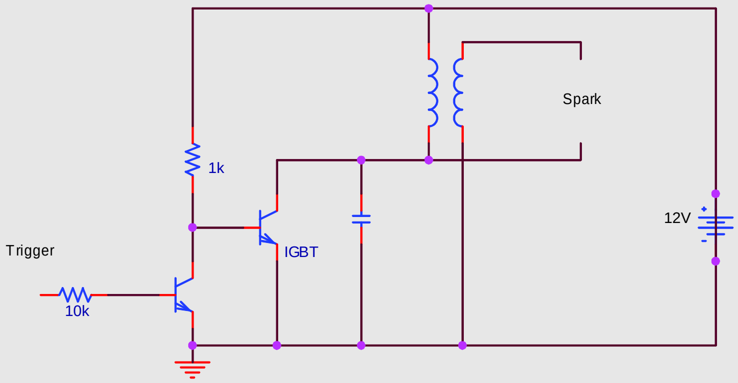


Fault Injection

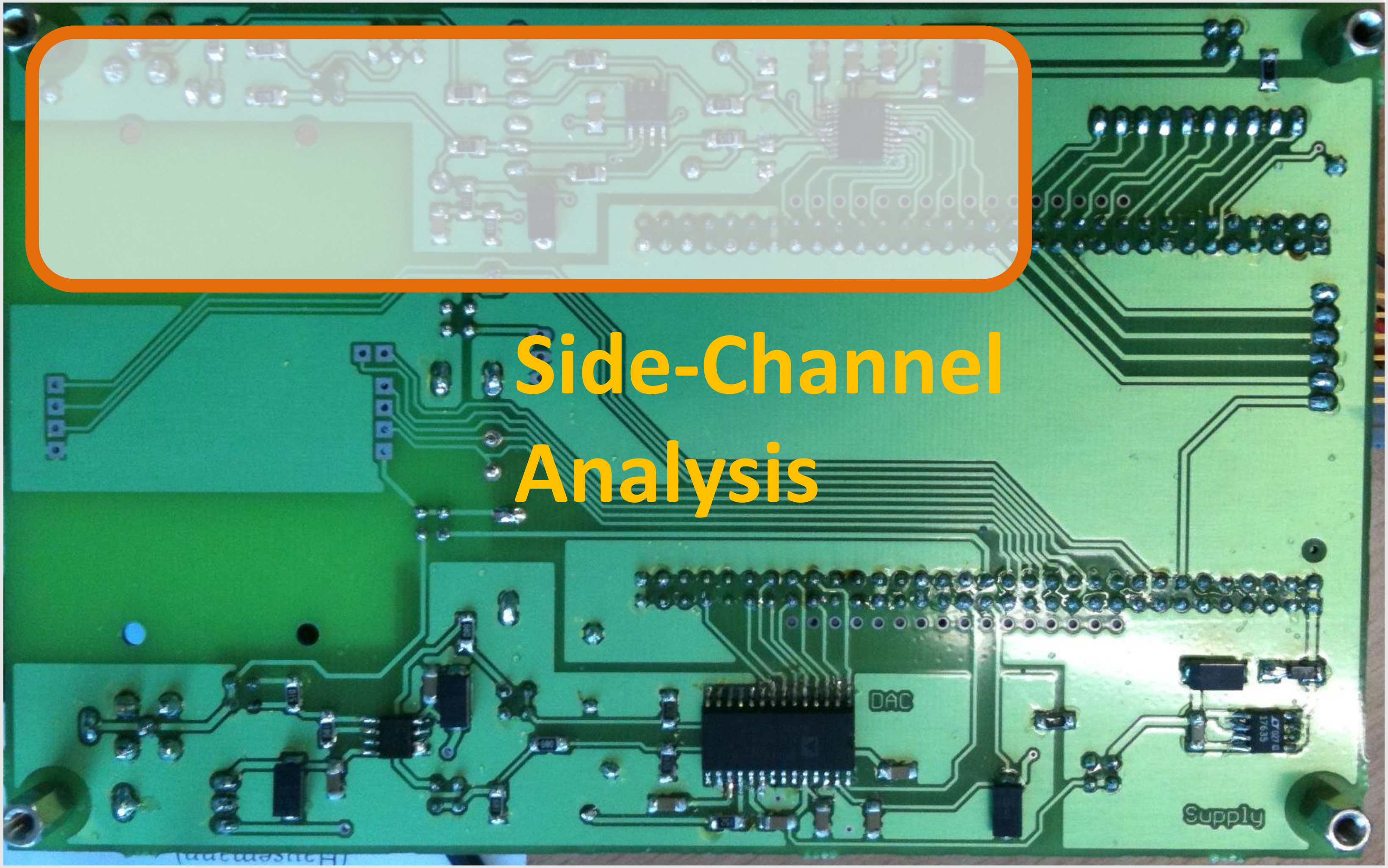
- **Digital-Analog Converter** AD9283
  - Up to 100 MHz (Resolution 10 ns)
  - Amplifier: -10 V … +10 V
  - Arbitrary waveform possible
- Extendable with external circuitry
  - Clock glitches
  - EM pulses
  - Laser





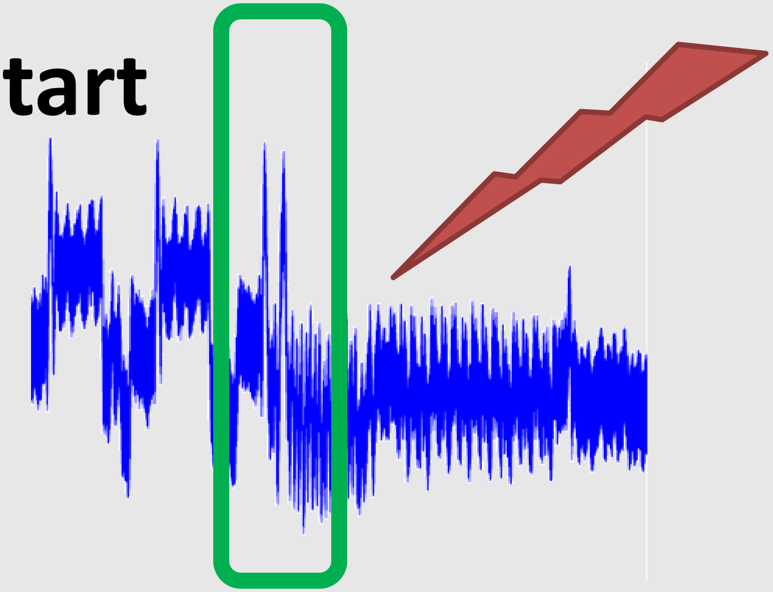


GIAnT: Hardware Overview



Side-Channel Analysis

- **Analog-Digital Converter** AD9283
  - Up to 100 MHz
  - 64 MB SRAM on FPGA module
- Record analog signals for side-channel analysis
- Pattern-detection for dynamic triggering



Detect start pattern

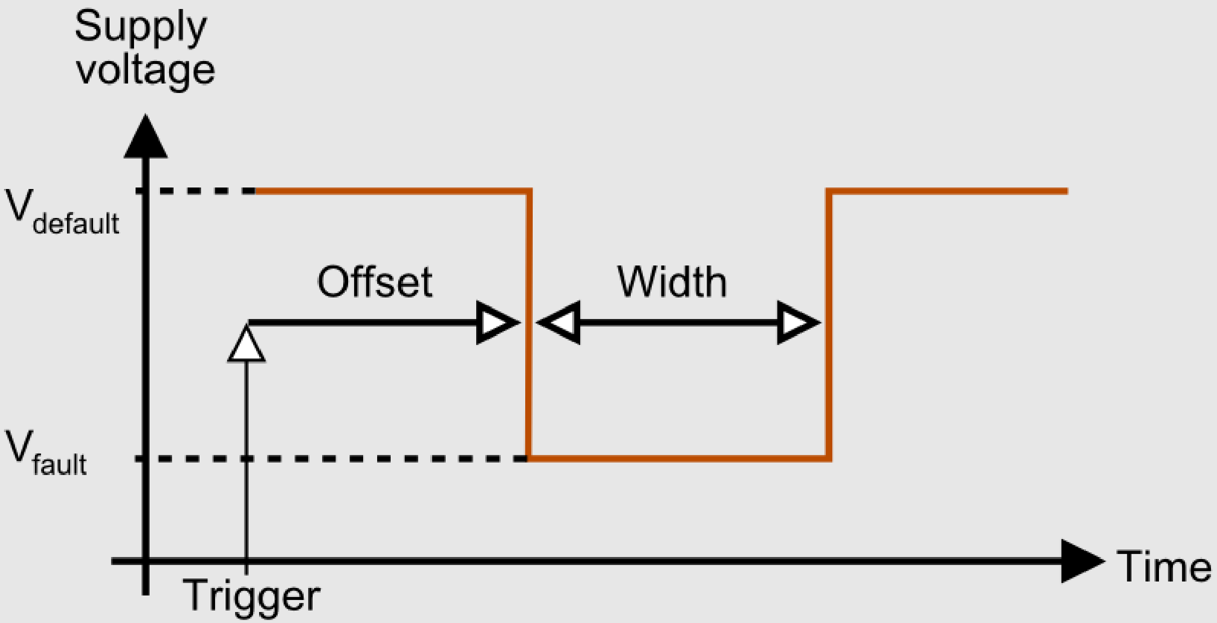
Dynamically inject fault

Fault injection

Practical Results

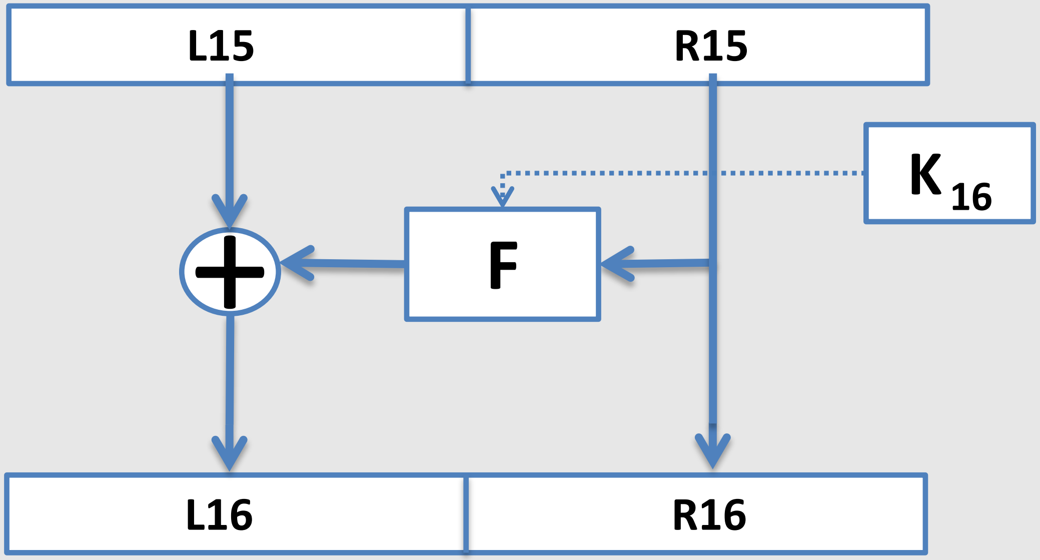
Practical Results: Basics

- **Aim**: Demonstrate basic functionality
- **Test devices:** Popular 8-bit µC
- **Fault type:** Voltage glitch/pulse
- **Fault effect:** Skip instruction(s)



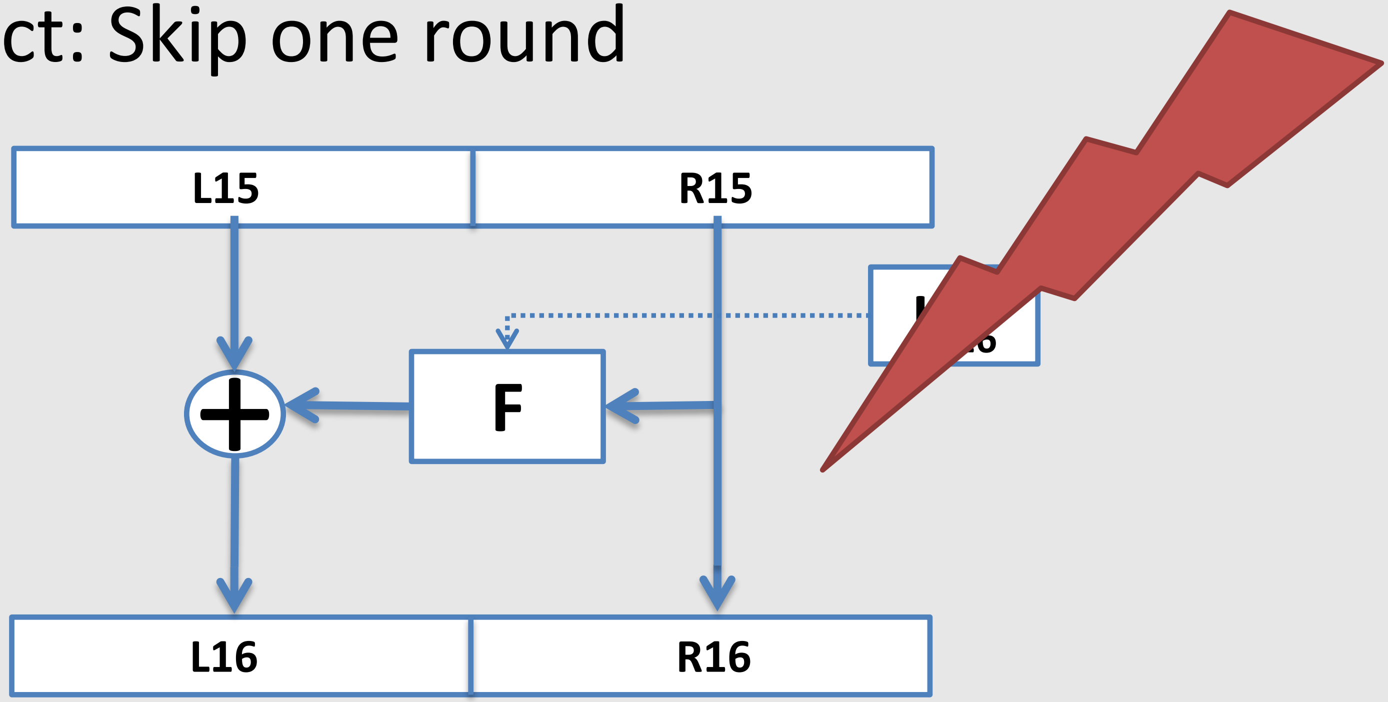
Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



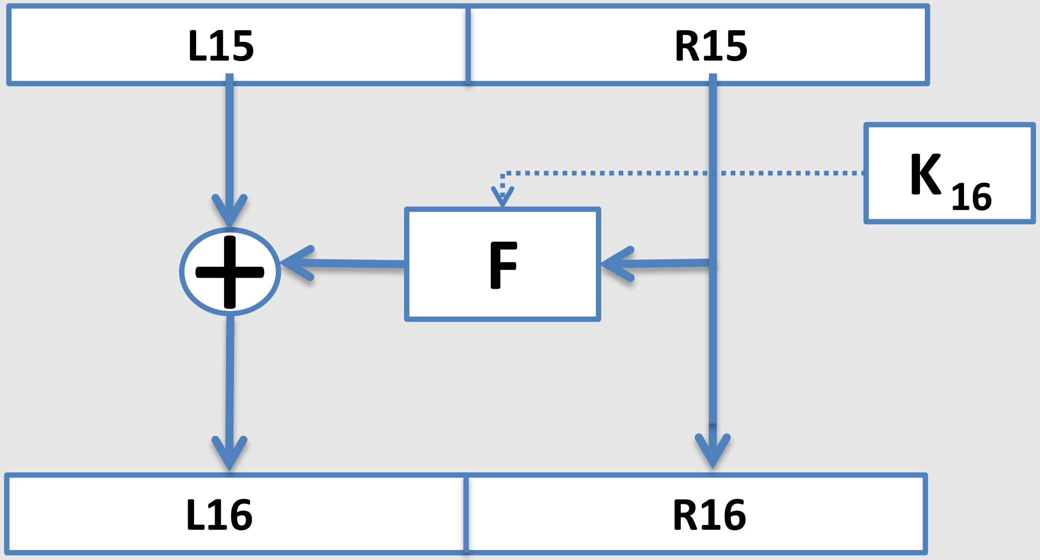
Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



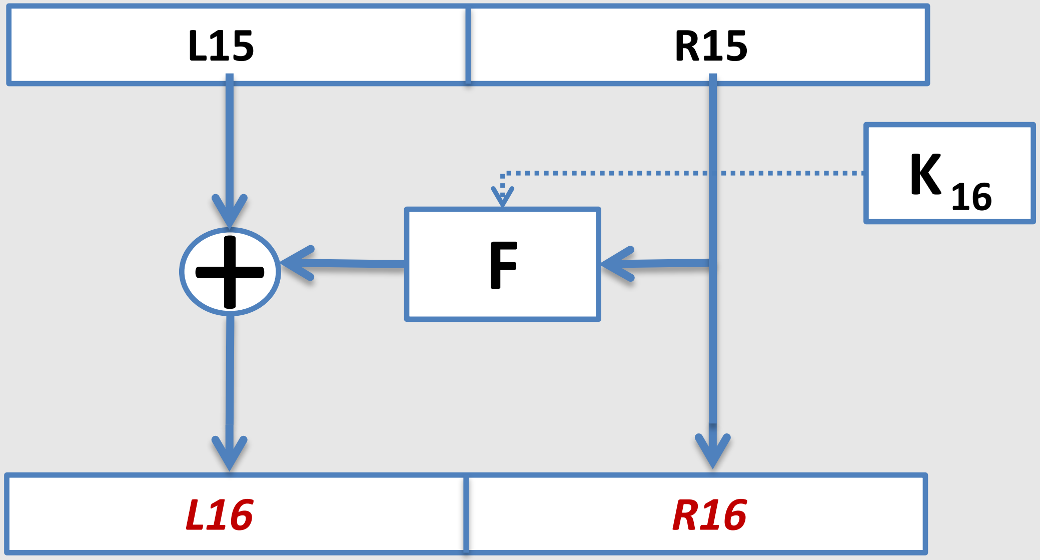
Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



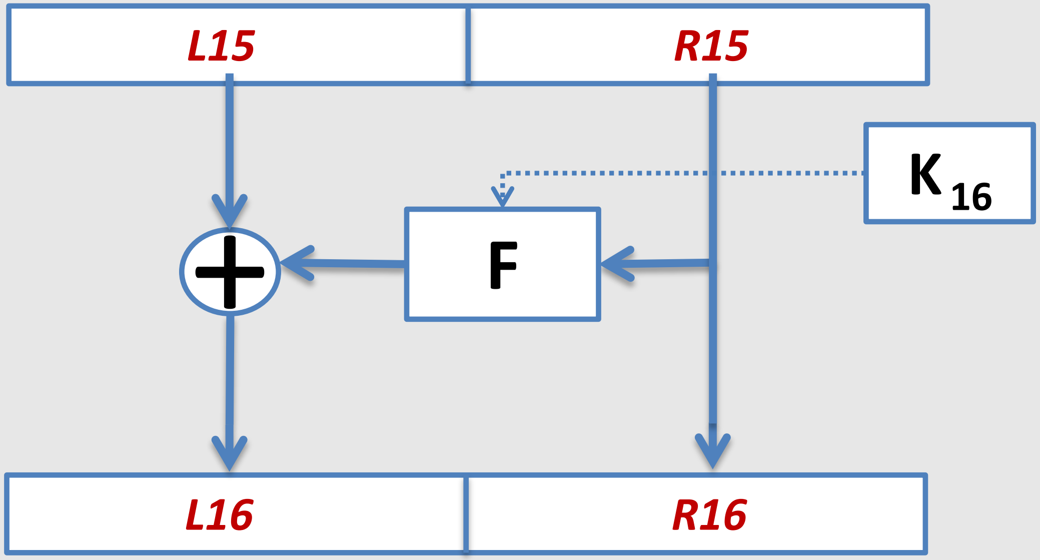
Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



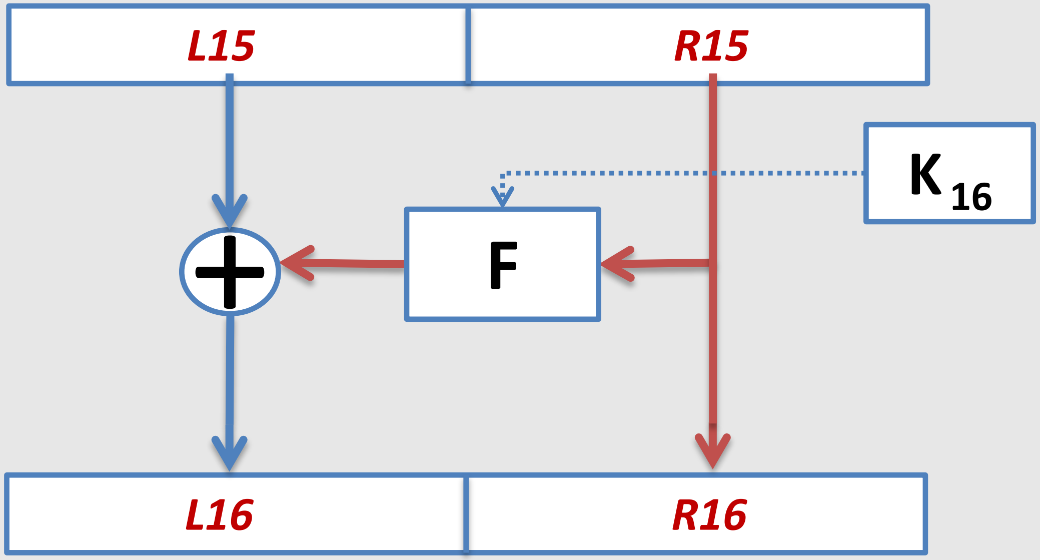
Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



Practical Results: 3DES on ATXMega

- Atmel ATXMega: Hardware **DES** engine
- Execute `DES` instruction 16 times
- Fault effect: Skip one round



- Recover K_{16}, iterate for full key

Practical Results: CRT-RSA and AES on ATMega

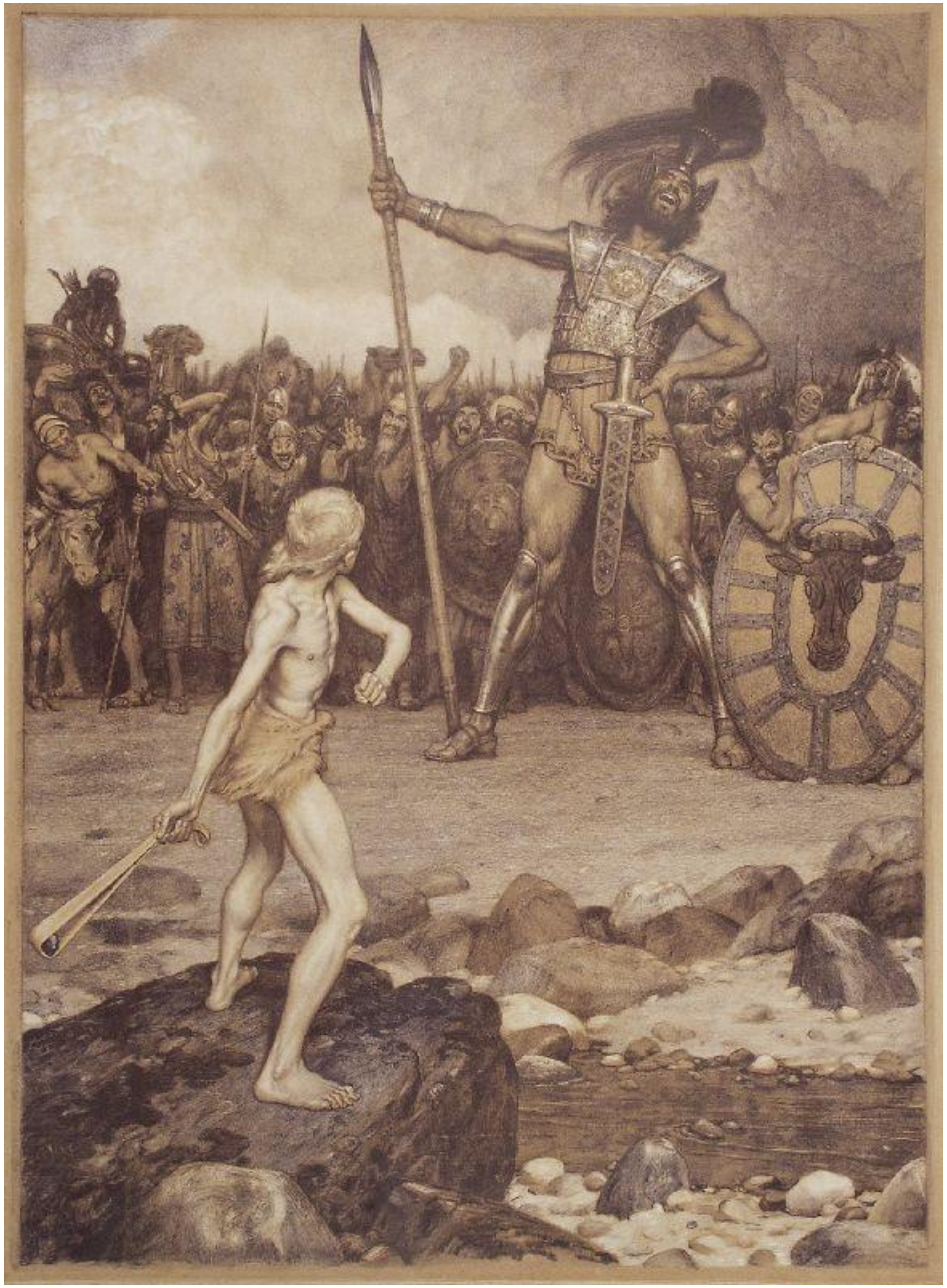
- Atmel ATMega: Software **CRT-RSA** on „smartcard“

  - Obtain faulty signature *c‘* on *x*
  - Lenstra: $d = gcd(x - (c')^e, n)$
- Atmel ATMega: Software **AES**
  - Fault causes modification of internal states
  - Used for live demo

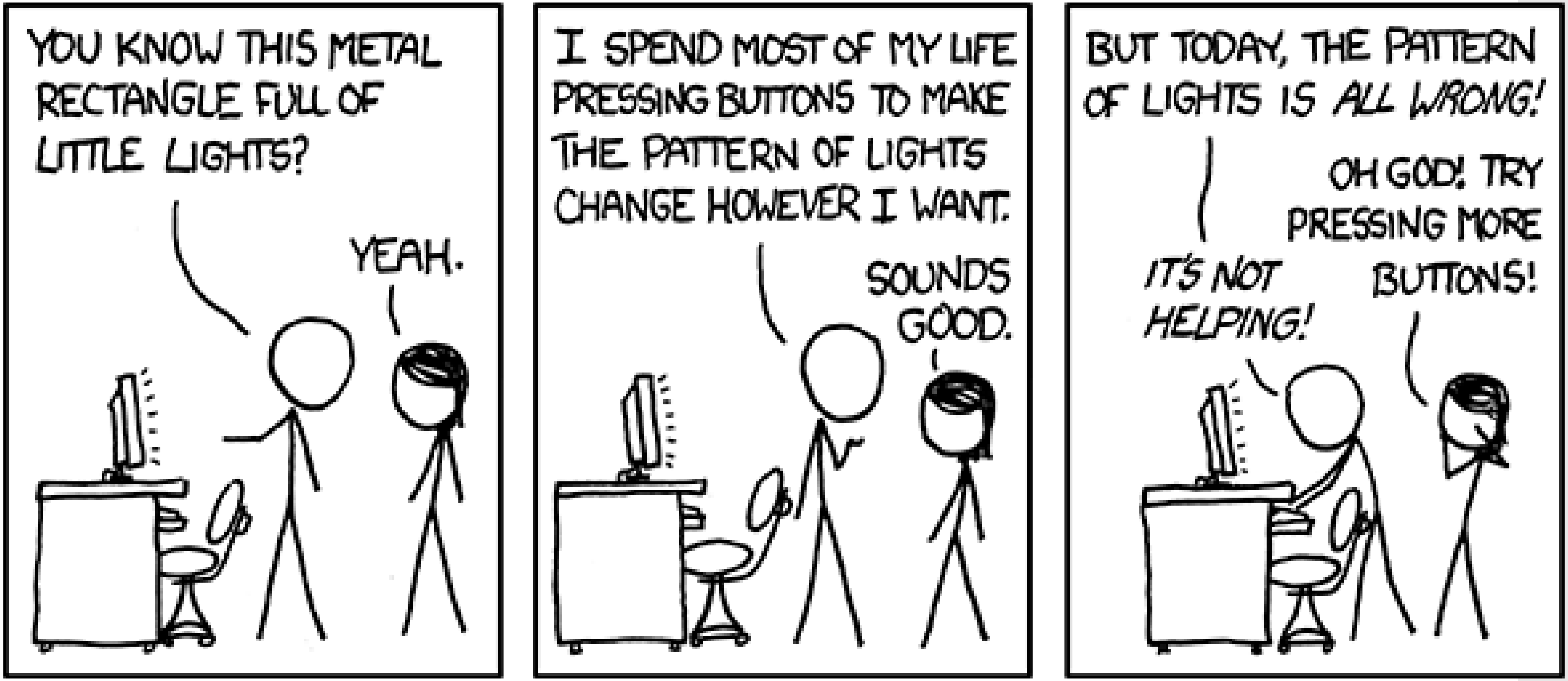
Let‘s hope the best and expect the worst

Live demonstration

Sometimes, testing and debugging feels like this...



In case it is not working ....


YOU KNOW THIS METAL RECTANGLE FULL OF LITTLE LIGHTS?
YEAH.
I SPEND MOST OF MY LIFE PRESSING BUTTONS TO MAKE THE PATTERN OF LIGHTS CHANGE HOWEVER I WANT.
SOUNDS GOOD.
BUT TODAY, THE PATTERN OF LIGHTS IS ALL WRONG!
OH GOD! TRY PRESSING MORE BUTTONS!
IT'S NOT HELPING!

xkcd: http://xkcd.com/722/

Live Demonstration: Software AES on ATMega

1. **Normal operation:**

   After first key addition and S-Box layer @ V_{dd} = 2.5V

   ```
   63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 76
   ```

2. **Effect of fault voltage:** 2V vs. 1V

   ```
   63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 6b
   ```

3. **Effect of pulse duration:** 10ns ... 100ns

   ```
   w = 10: 63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 76
   w = 20: 63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 63
   w = 30: 63 7c 77 7b f2 6b 6f c5 30 01 67 2b fe d7 ab 6b
   w = 40: 3b bb 11 00 91 81 31 46 15 2a 53 6d 34 72 74 43
           34 72 64 2a b5 (reset, ATR)
   ...
   ```

Conclusion

GIAnT: Conclusion

- **Fault injection** and **side-channel analysis** in-a-box
- **Low-cost**
- **Open source**
- Tested with various devices
- Continously being improved
  - RFID
  - Different pulse shapes
  - Other fault injection methods
  - ...
- Contributions are welcome, visit sf.net/projects/giant

Thanks!

Questions?

David Oswald, Timo Kasper, Stephen Markhoff, Christof Paar

Chair for Embedded Security, Ruhr-University Bochum