

# The Next 10 Years of IT Security: RFID, BMWs and Burglars



---

**Stanford University**  
**August 20, 2008**

Christof Paar  
University Bochum  
[www.crypto.rub.de](http://www.crypto.rub.de)

# Overview

- Embedded Systems and Security
- Case Study 1: Securing RFID
- Case Study 2: High Speed Signature Engine
- Case Study 3: Access Control and Physical Attacks
- Related Activities

# What are Embedded Systems?



- „Processor hidden in a product“, or
- „A computer that doesn't look like a computer“

# Characteristics of Embedded Systems

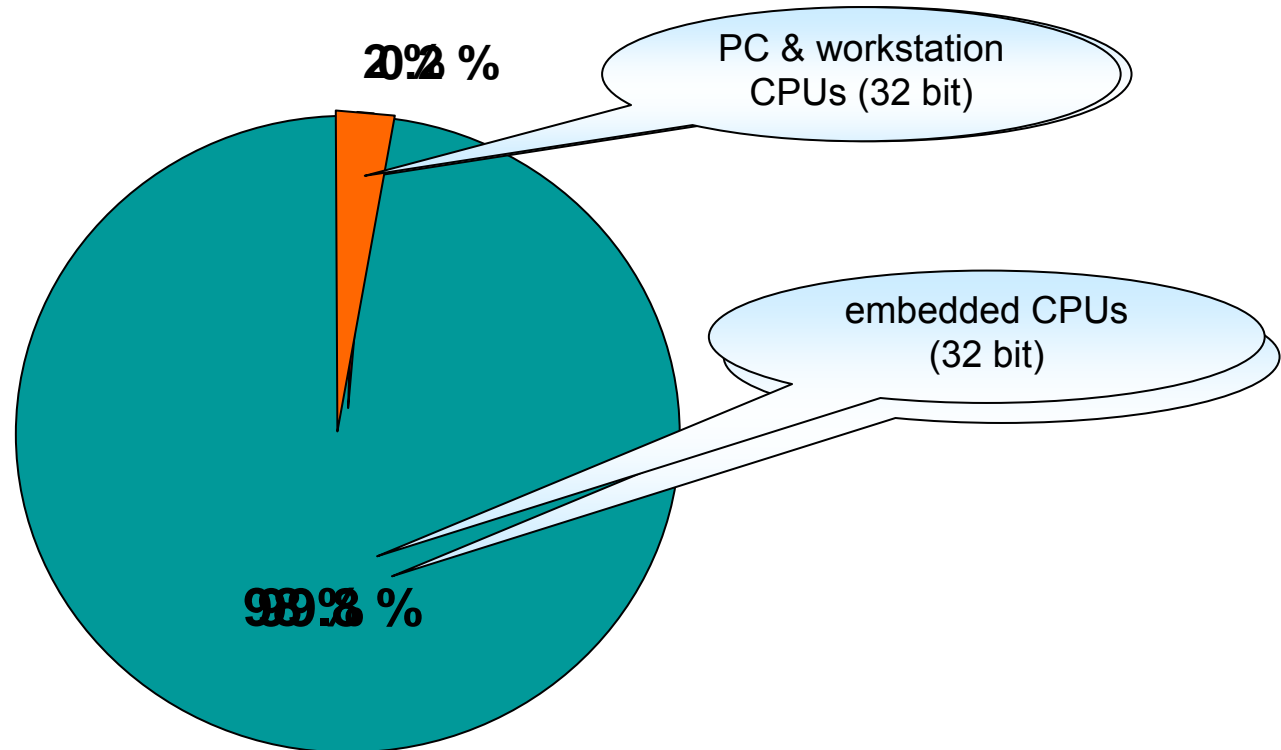
- Definition: „Device with processor“



- Single purpose
- Interacts with the world
- many, many applications

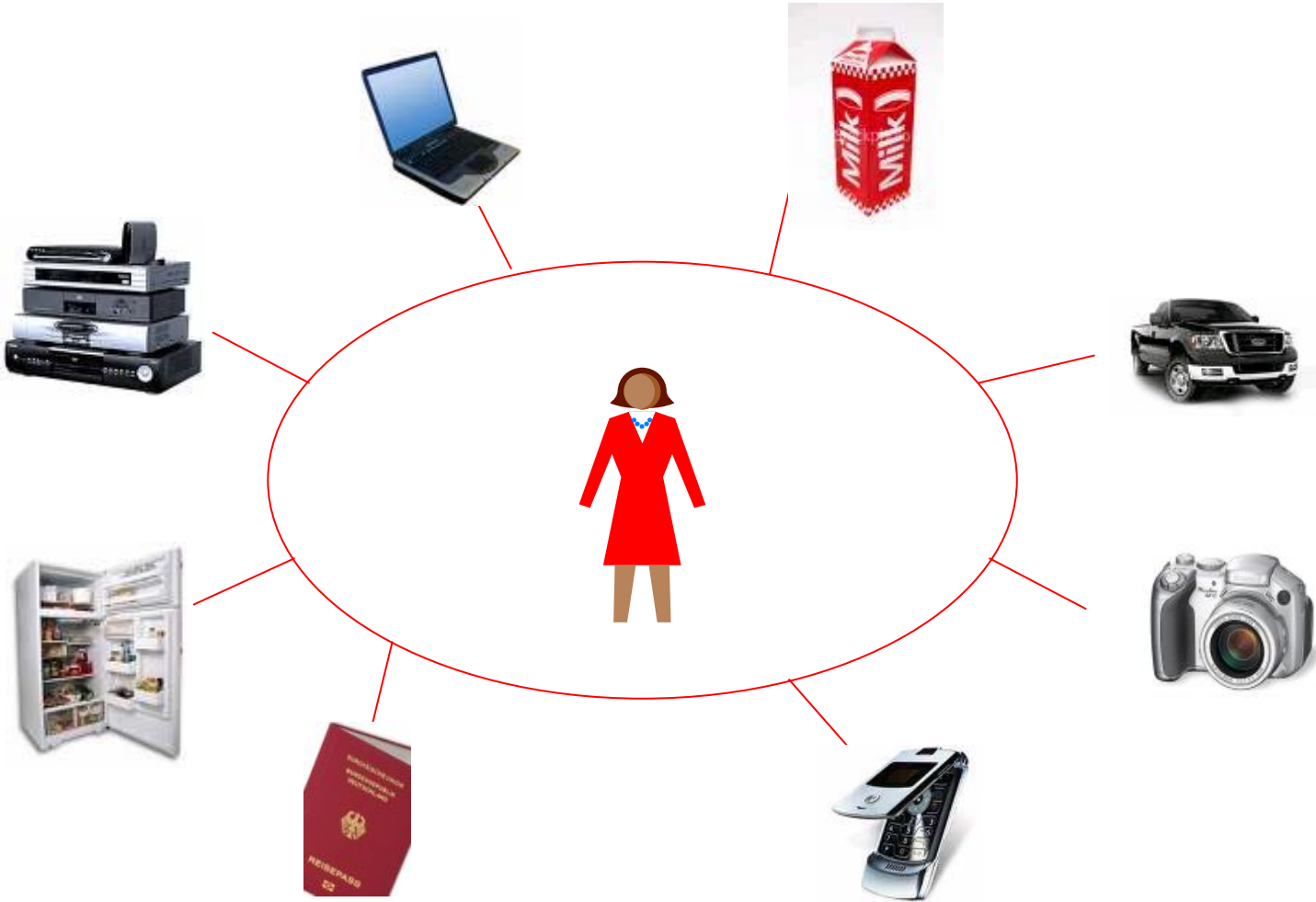
# Is this really important ?

- current CPU market
- by the numbers



- So, how does embedded technology affect the future IT landscape?

# Brave New Pervasive World



# Security Concerns in Embedded Applications

- Pervasive nature and **safety-critical** applications increase risk potential:  
*Hard disk crash vs. car crash*
- Often **wireless channels**  $\Rightarrow$  vulnerable
- **Contents protection** in many applications: iPod, navigation systems, XBox,, ...
- **Secure SW download**: engine control, cell phones, washing machine,...
- **Component protection**: original spare parts, product privacy protection, ...
- **Privacy issues**: biometrics (face recognition), geolocation, medical sensors, monitoring of home activities, etc.
- **Legislative requirements**: passports, road toll, data event recorders in machines, ...

# Overview

- Embedded Systems and Security
- **Case Study 1: Securing RFID**
- Case Study 2: High Speed Signature Engine
- Case Study 3: Access Control and Physical Attacks
- Related Activities

# Lightweight Cryptography

- “We need security with less than 2000 gates”  
Sanjay Sarma, AUTO-ID Labs, CHES 2002



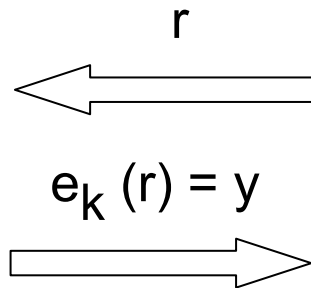
- \$3 trillions annually due to product piracy\* (> US budget '07)



\*Source: [www.bascap.com](http://www.bascap.com)

- ⇒ Authentication & identification problem: can both be fixed with cryptography
- ⇒ Q: How cheap can we make symmetric ciphers?

# Strong Identification (w/ symmetric crypto)



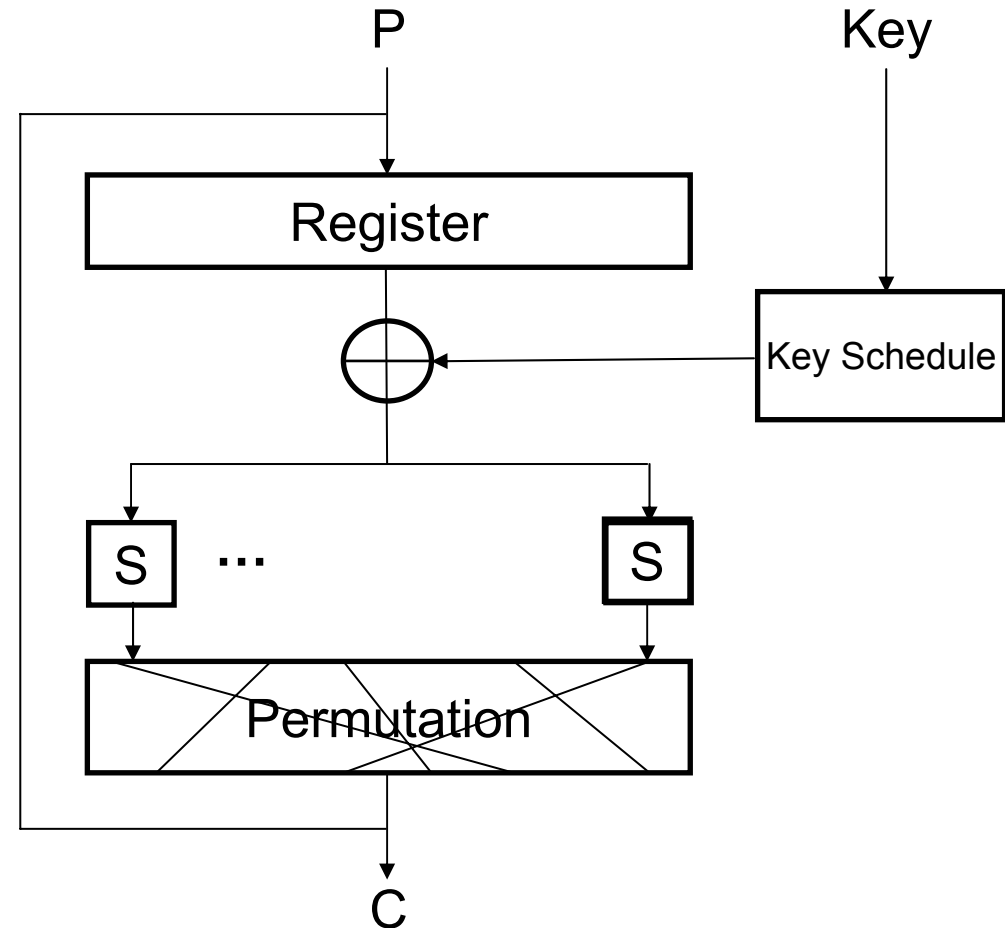
1. random challenge  $r$
2. encrypted response  $y$
3. verification  
 $e_k(r) = y'$   
 $y == y'$

Challenge: Encryption function  $e()$  at extremely low cost

→ almost all symmetric ciphers optimized with SW in mind

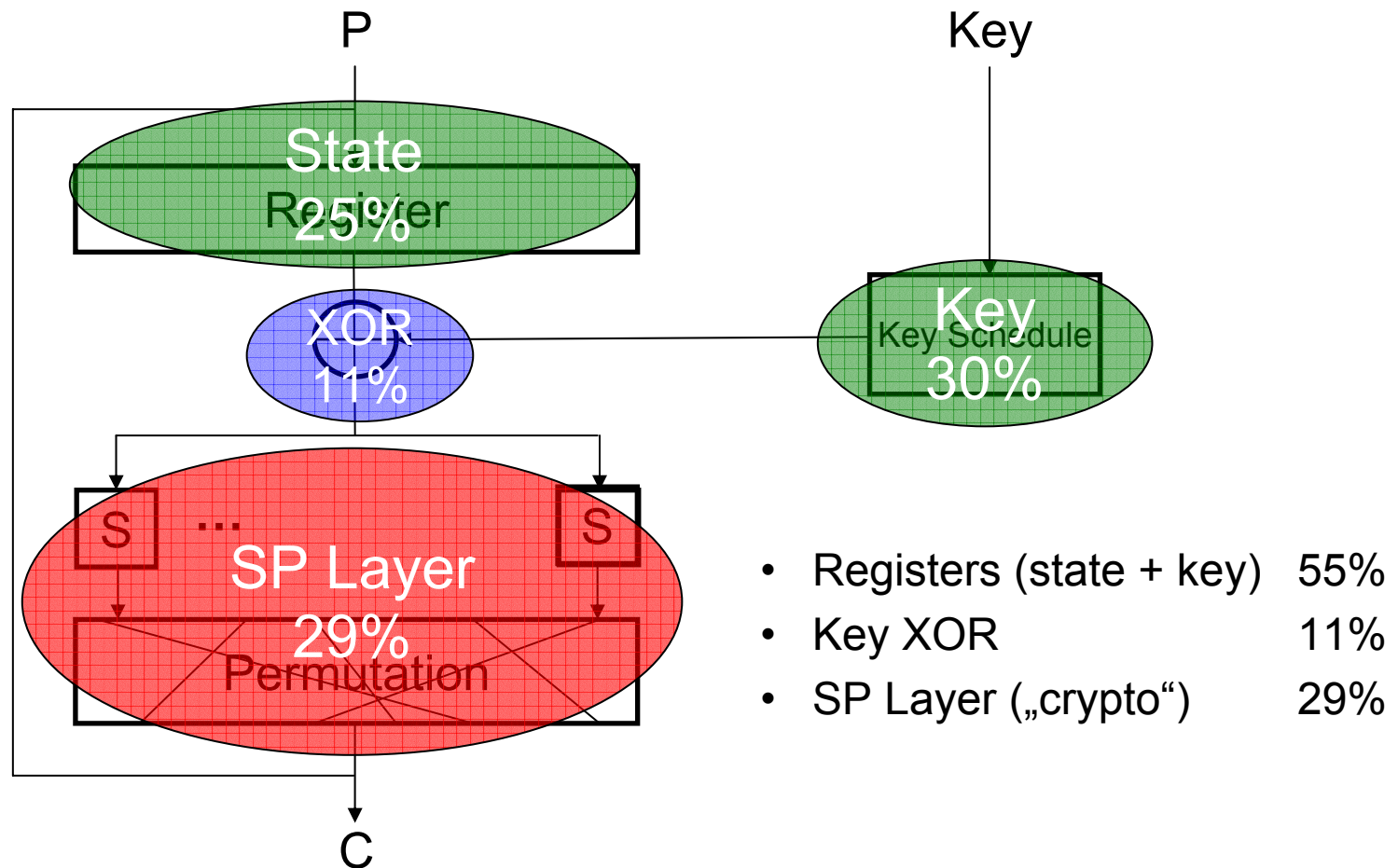
# PRESENT – An aggressively hardware optimized block cipher for RFID

- pure substitution-permutation network
- 64 bit block, 80/128 bit key
- 4-4 bit Sbox
- 31 round (32 clks)
- „provable secure“ against DC, LC
- joint work with Lars Knudsen, Matt Robshaw et al.
- no patents etc.

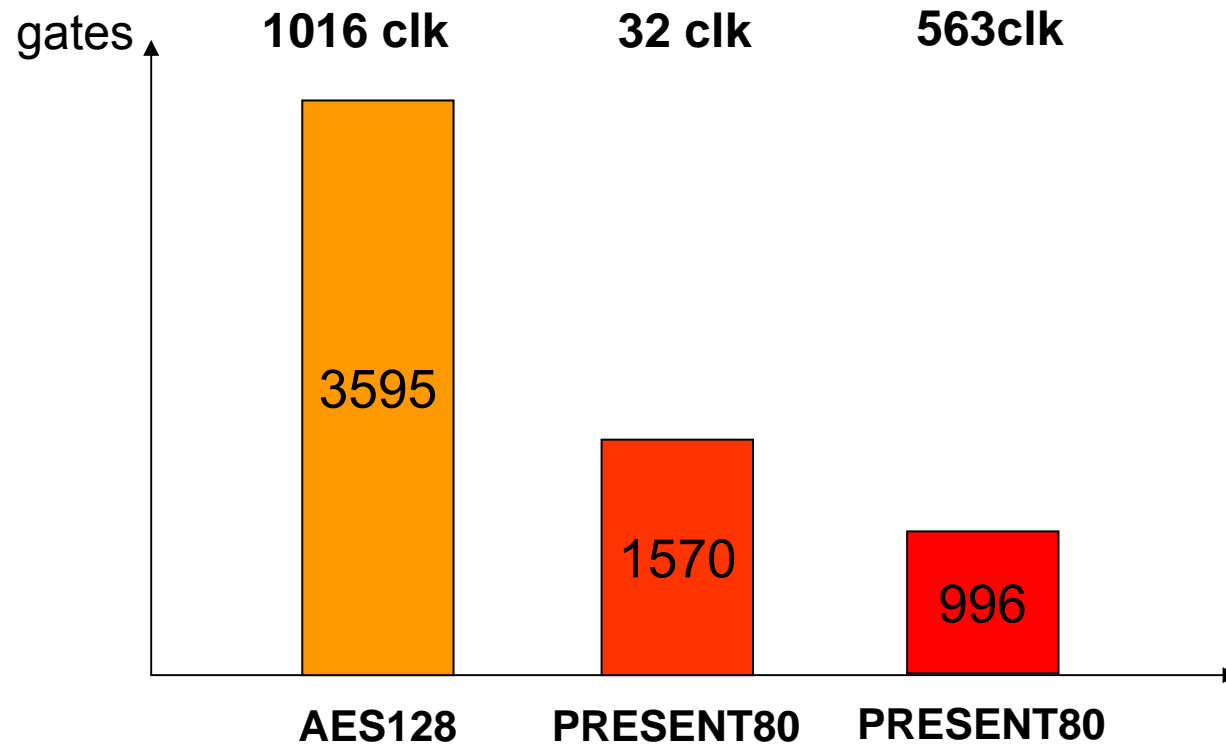


# Resource use within lightweight ciphers

## Round-parallel implementation of PRESENT (1570ge)



# Results – PRESENT



- TA product 1-2 orders of magnitude better than smallest AES architecture
- Serial implementation approaches theoretical complexity limit: almost all area is used for the 144 bit state (key + data path)
- smaller than all stream ciphers
- details: CHES '07 paper

- Embedded Systems and Security
- Case Study 1: Securing RFID
- **Case Study 2: High Speed Signature Engine**
- Case Study 3: Access Control and Physical Attacks
- Related Activities

# Case Study: High Speed Signature Engine

- USA: 42,000+ car fatalities per year (IIHS, 2002)
- 3.2m injuries (2000)
- est.: 90% driver errors



Video courtesy of Ken Labertaux,  
Toyota Research

- Mechanical safety (safety belt, air bag, ABS): great success but limits have been reached
- *Electronic driver assistance* will be key tool

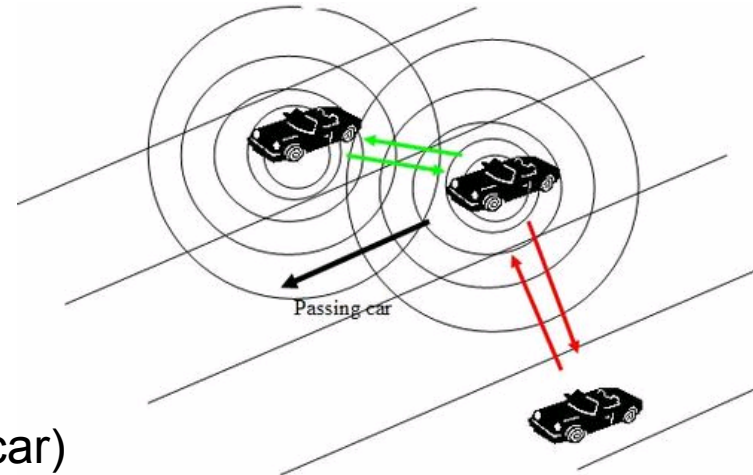
# VANET – Vehicular Ad-Hoc Networks

Broadcast position & direction information:

1. greatly improve safety
2. improve traffic management

Network characteristics

- small messages ( $\approx 100$  Bytes)
- medium frequency ( $\approx 10$  messages/sec per car)
- very ad-hoc (short lived, high dynamics)
- high number of incoming messages ( $> 1000$ msg/sec per car)
- IEEE P1609/DSRC standard



But messages must be authenticated!  
(safety-critical & legislative requirements)

# Elliptic Curve Primitive

- Given a Point  $P$  on an elliptic curve  $E$  over  $GF(p)$ :

$$E: y^2 = x^3 + ax + b \pmod{p}$$

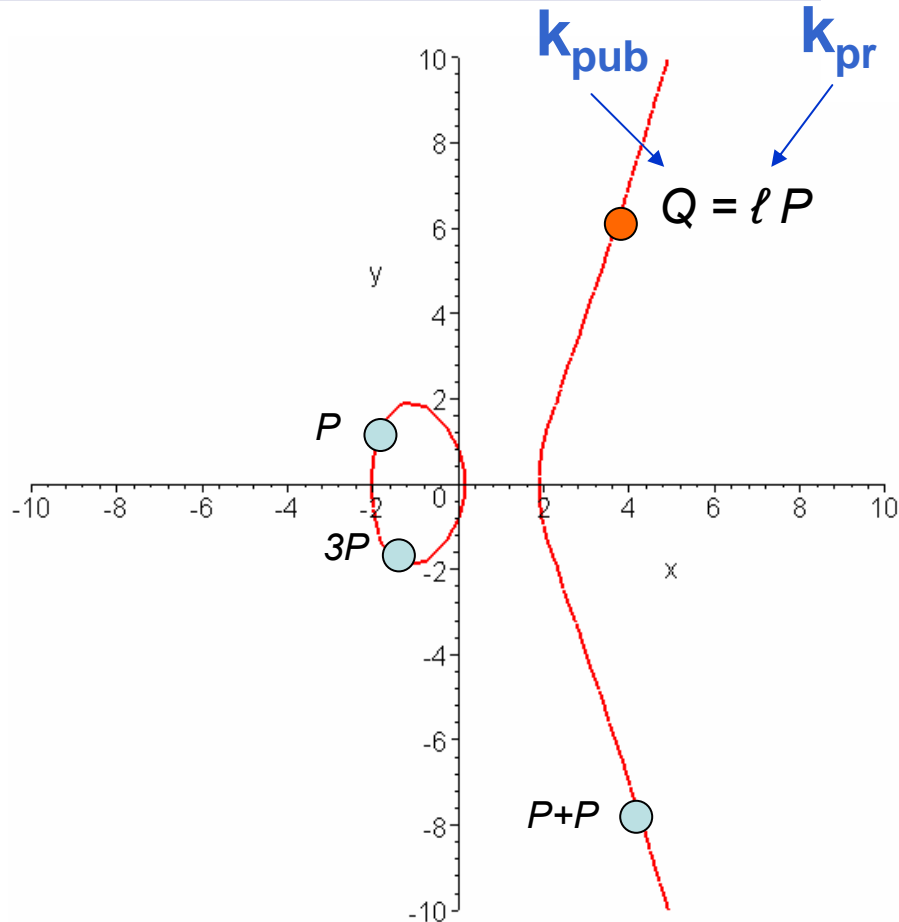
- Public key  $Q$  is multiple of base point  $P$

group  
operation

$$Q = P + P + \dots + P = \ell P$$

- EC discrete logarithm problem:

$$\ell = \text{dlog}_P(Q)$$



# Point Addition on EC

## Jacobian Coordinates over GF(p)

- **Point Addition**  $R = P + S$
- Input  $P = (X_1, Y_1, Z_1)$  ;  $S = (X_2, Y_2, Z_2)$
- Output  $R = (X_3, Y_3, Z_3)$

$$A = X_1 Z_2^2 \bmod p$$

$$B = X_2 Z_1^2 \bmod p$$

$$C = Y_1 Z_2^3 \bmod p$$

$$D = Y_2 Z_1^3 \bmod p$$

$$E = B - A \bmod p$$

$$F = D - C \bmod p$$

$$X_3 = -E^3 - 2AE^2 + F^2$$

$$Y_3 = -CE^3 + F(AE^2 - X_3)$$

$$Z_3 = Z_1 Z_2 E$$

$$1 \text{ Point Add} = 14 \text{ MUL}_{256\text{bit}} = 3584 \text{ MUL}_{16\text{bit}}$$

# Real-Time Signature Engine for VANETs

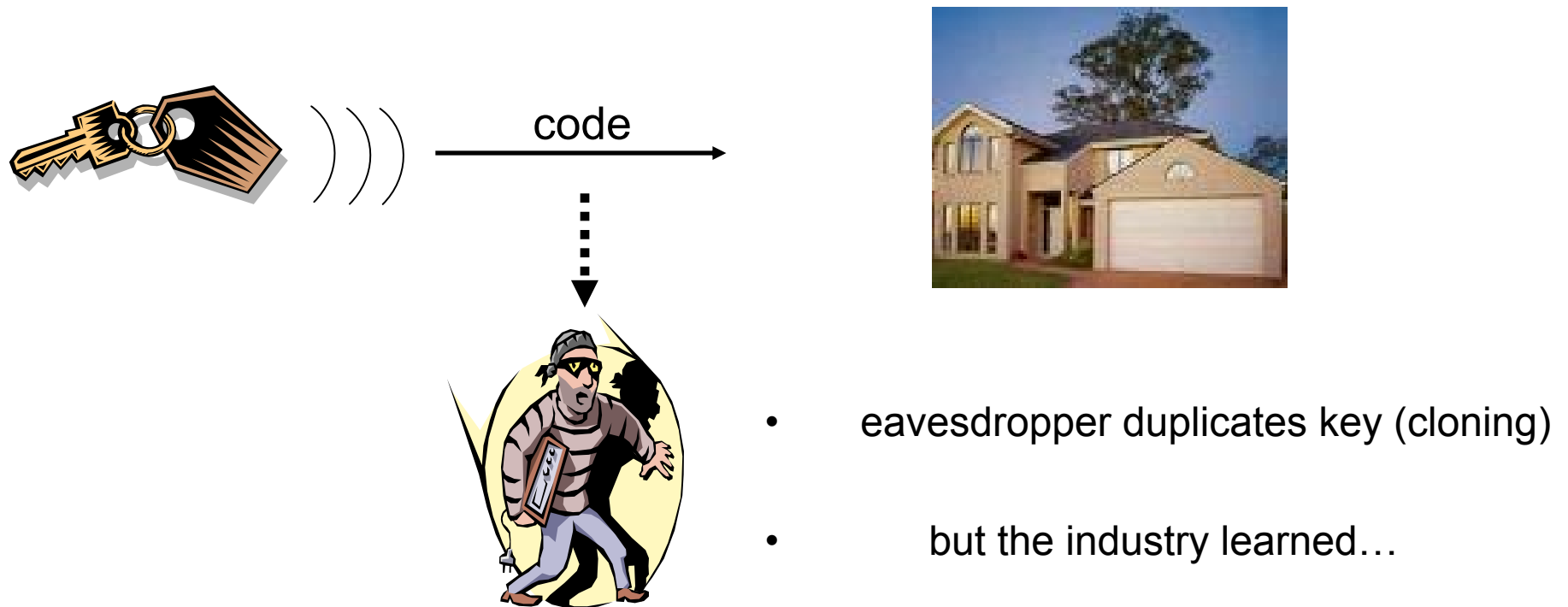
- *Requirements*
- 256bit ECC Engine (long-term security)
- 1000 sign./sec → 1,000,000,000  $Mul_{16}$ /sec
  
- *New VANET Signature Engine*
- 1  $Mul_{256}$  requires 63 cycles@500MHz
- **1 ECC VANET engine: > 1500 signatures/sec**
- performance and cost-performance record for **commercial hardware**
- patent pending

# Overview

- Embedded Systems and Security
- Case Study 1: Securing RFID
- Case Study 2: High Speed Signature Engine
- **Case Study 3: Access Control and Physical Attacks**
- Related Activities

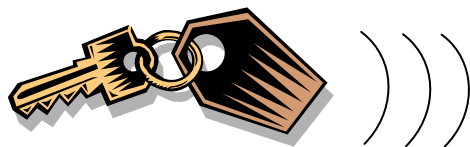
# Case Study Access Control

- Simple access controls: fixed code (“password”)



# Case Study Access Control

- advanced theft control: rolling code



$$\text{code} = e_k(n_i)$$

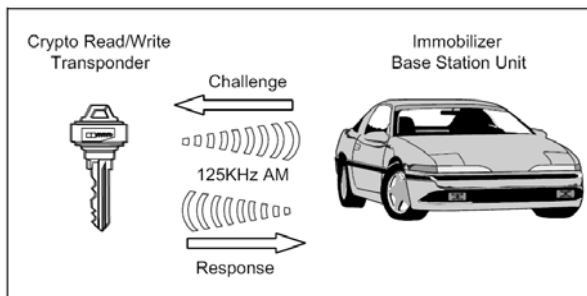


- rolling code (or hopping code)
- $\text{code} = e_k(n)$
- $\text{code} = e_k(n+1)$
- $\text{code} = e_k(n+2)$
- .....

$e_k()$  is often a  
block cipher

# Popular Rolling Code Cipher: KeeLoq

## HCS410 IMMOBILIZER TRANSPONDER

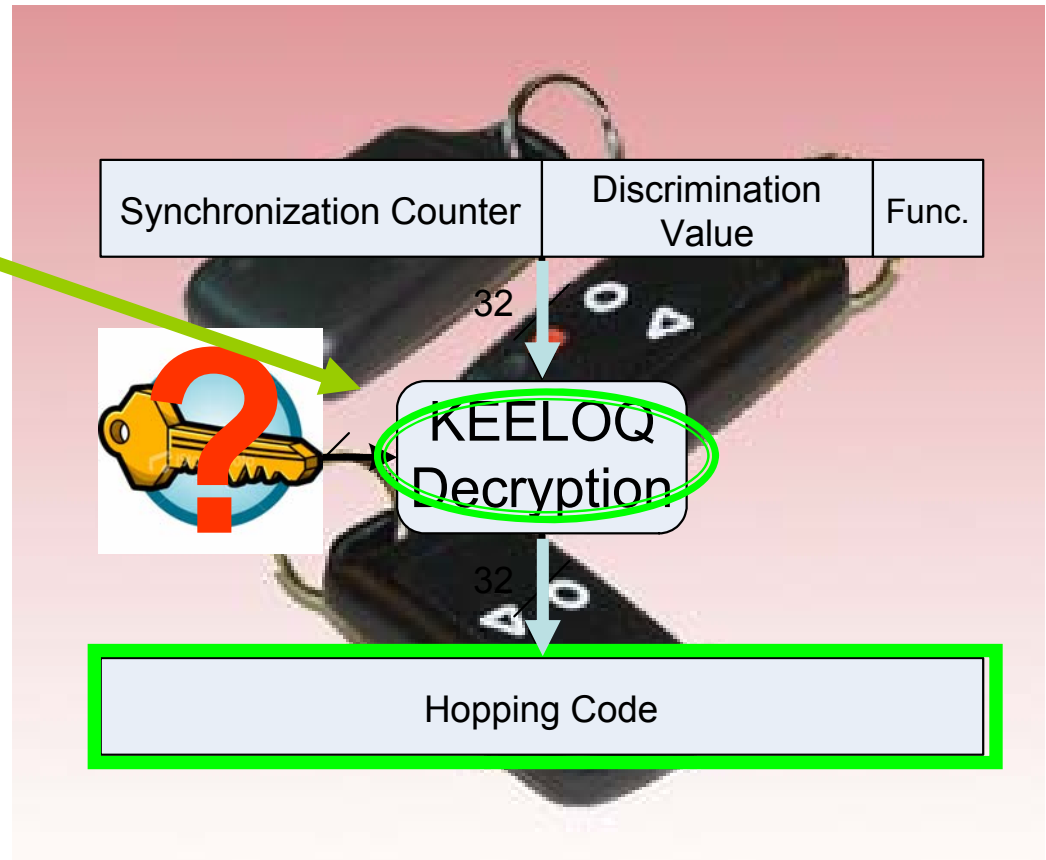
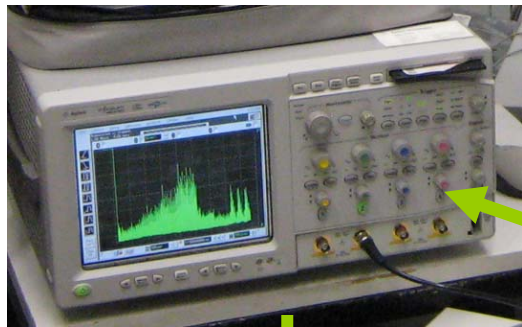


- Garage door access, car access, user authentication, ...
- KeeLoq chip embedded in passive or active RFID transponder („car key“)
- Wikipedia (?):  
Chrysler, Daewoo, Fiat, GM, Honda, Toyota, Volvo, Jaguar, ...

- Q: How secure is KeeLoq?
- Best known mathematical attack does not work for rolling code, requires 65,000 encryptions + plaintext + works only for certain (weak) key derivations
- but:

**? What about physical attacks ?**

# Side Channel Analysis



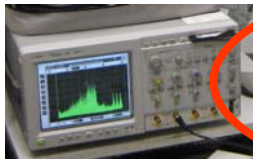
**secret key of remote control (HCS XXX Chip) !**

# Performing the Side-Channel Attack



Analyze cipher

- Find a suited predictable intermediate value in the cipher



Measurements

- Measure the power consumption



Post Processing

- Post-process acquired data

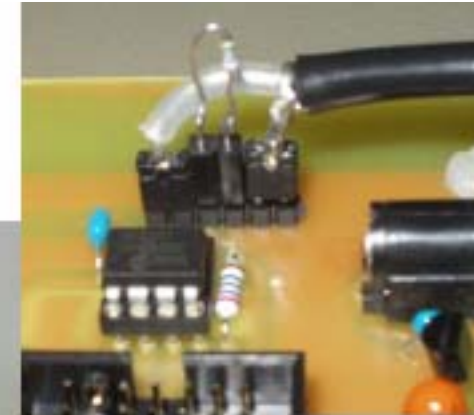
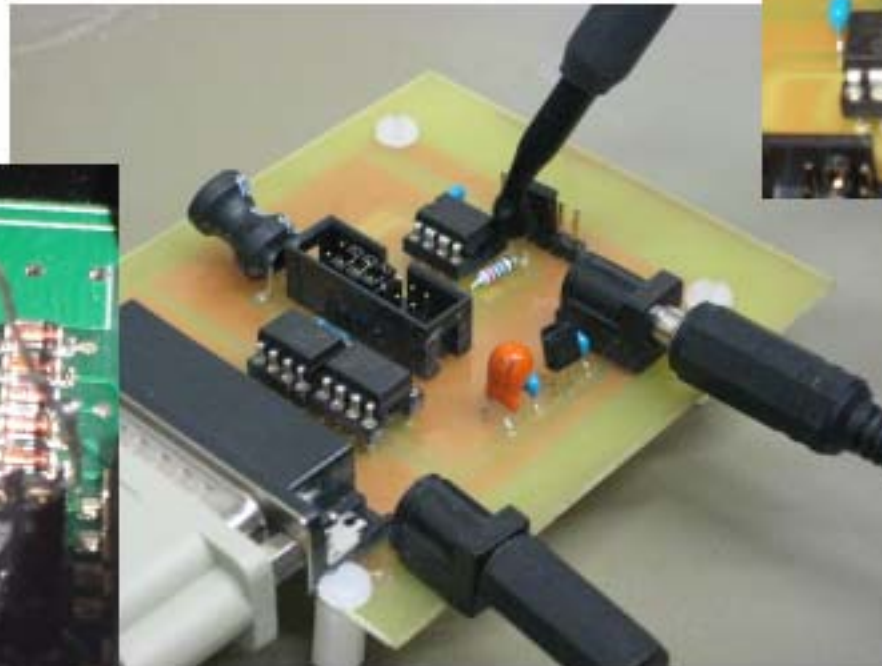


Key Recovery

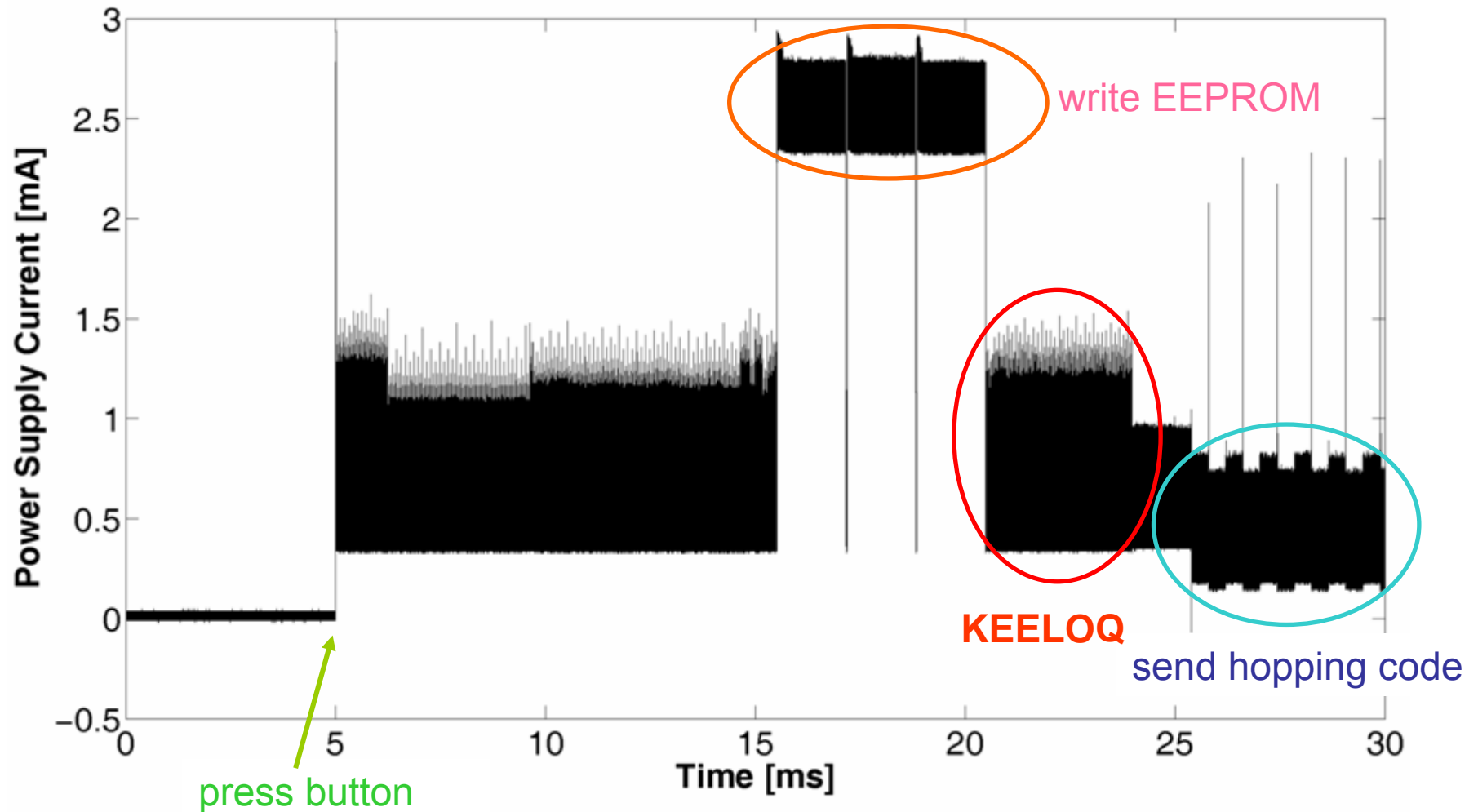
- Perform the attack to recover the key

# Measuring the Power Consumption

- digital oscilloscope (max. 1 GS/s sample rate)
- measure electromagnetic field or electric current



# Performing the Side-Channel Attack KeeLoq - Encryption

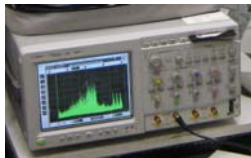


# Performing the Side-Channel Attack



Analyze cipher

- Find a suited predictable intermediate value in the cipher



Measurements

- Perform power measurements



Post Processing

- Post-process acquired data

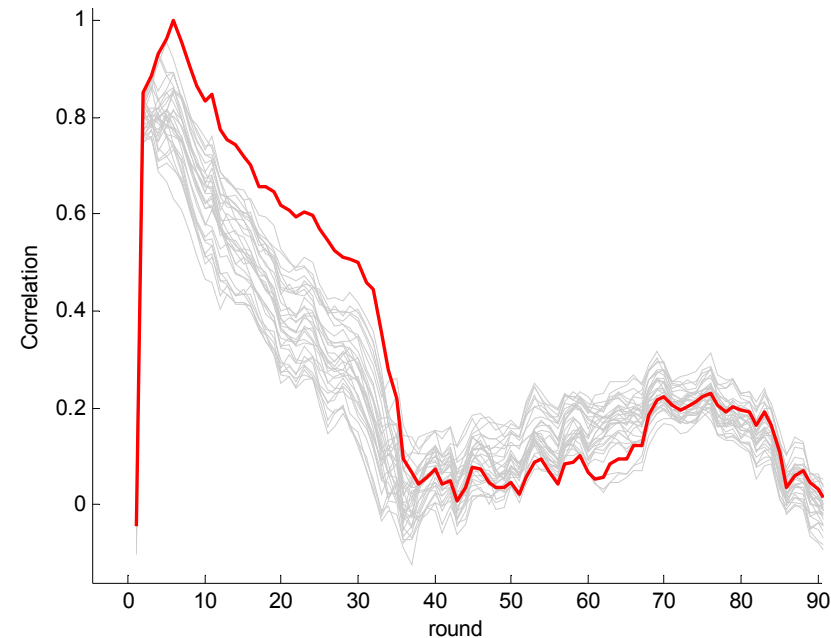


Key Recovery

- Perform the attack to recover the key

# Performing the Side-Channel Attack Key Recovery

- Correlate power consumption to predicted value  $y = f(x,k)$
- Divide and conquer approach
- Much off-line number crunching

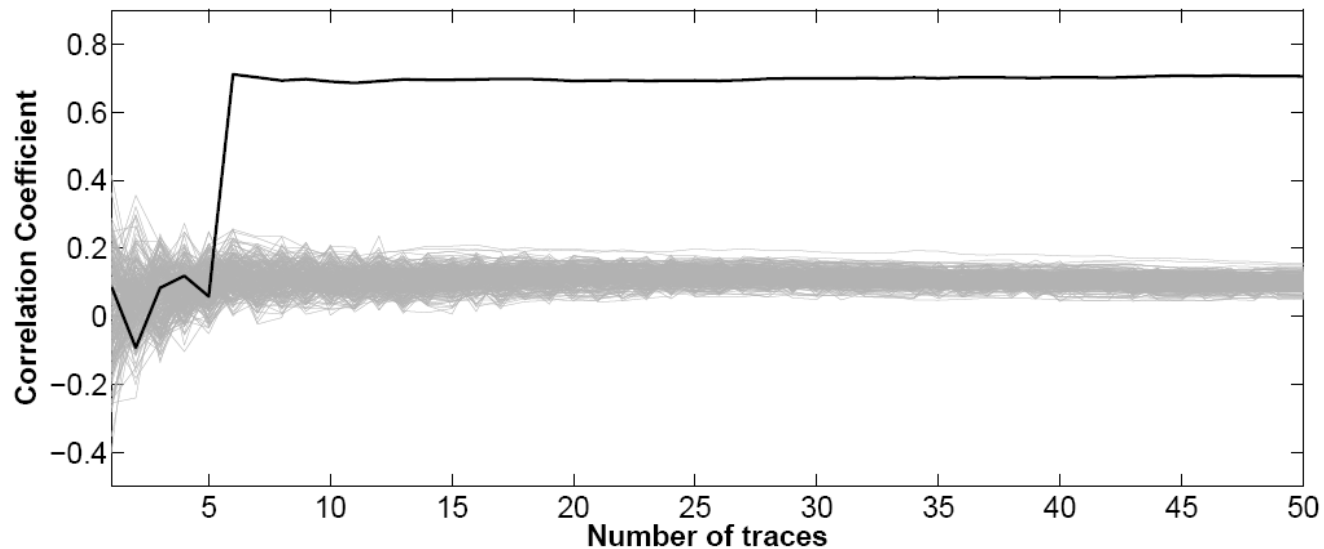


$$r(I_i(t), D(X_i, K_h)) = \frac{\sum_{i=1}^M I_i(t) \cdot D(X_i, K_h)}{\sqrt{\sum_{i=1}^M (I_i(t) - \overline{I_i(t)})^2 \cdot \sum_{i=1}^M (D(X_i, K_h) - \overline{D(X_i, K_h)})^2}}$$
$$= \frac{\frac{1}{M} \cdot \sum_{i=1}^M I_i(t) \cdot \sum_{i=1}^M D(X_i, K_h)}{\sqrt{\sum_{i=1}^M (I_i(t) - \overline{I_i(t)})^2 \cdot \sum_{i=1}^M (D(X_i, K_h) - \overline{D(X_i, K_h)})^2}}$$

# Side Channel Attack on transmitters

KeeLoq implemented in hardware

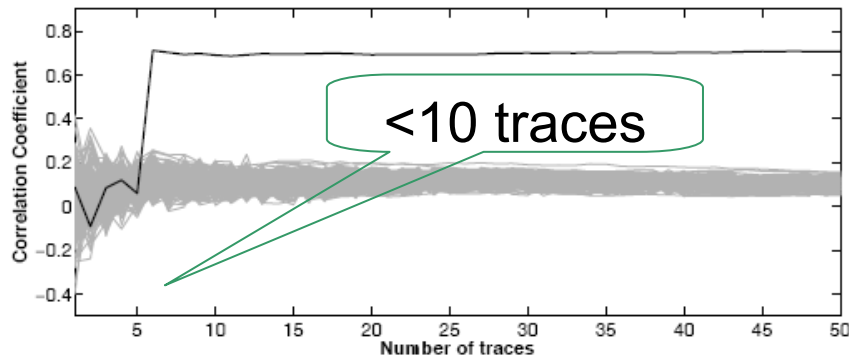
Total attack time (for known device family):  
5-30 traces,  $\approx$  minutes



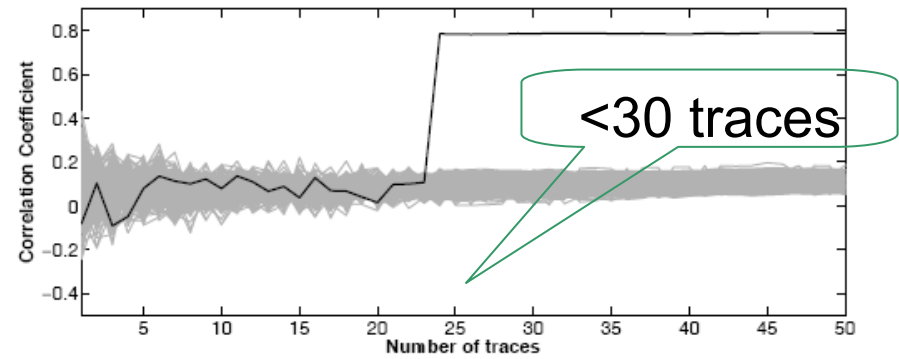
Convergence of correlation coefficient

Rem: low cost  
equipment suffices  
( $<$  \$1000)

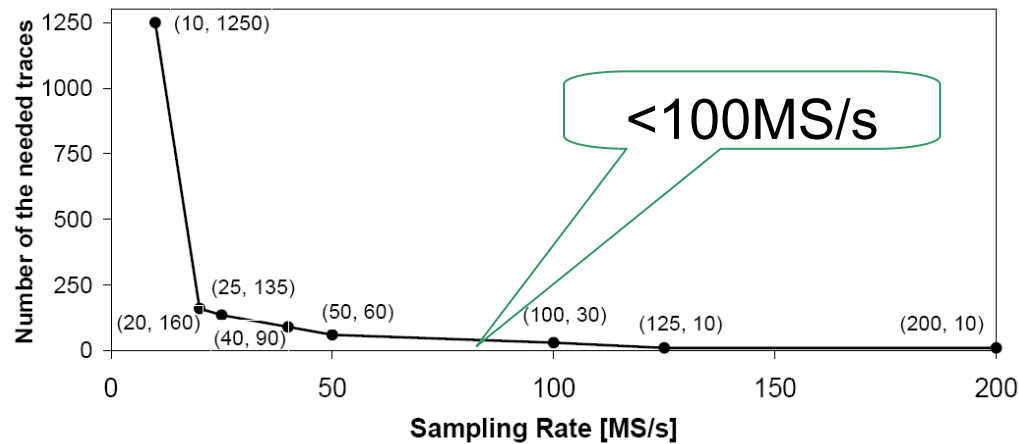
# Comparison of Packages & Sample Rates



(a) DIP



(b) SOIC



No expensive equipment needed !

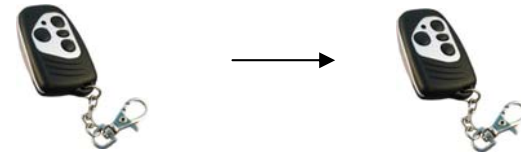
Rem: SCA on receivers (software) requires several 1000 traces

# So what can we do now?

---

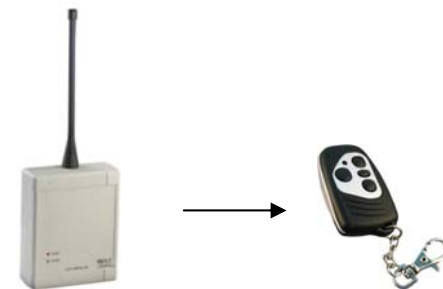
## If we have access to a remote

Recover device key and clone the device



## If we have access to a receiver

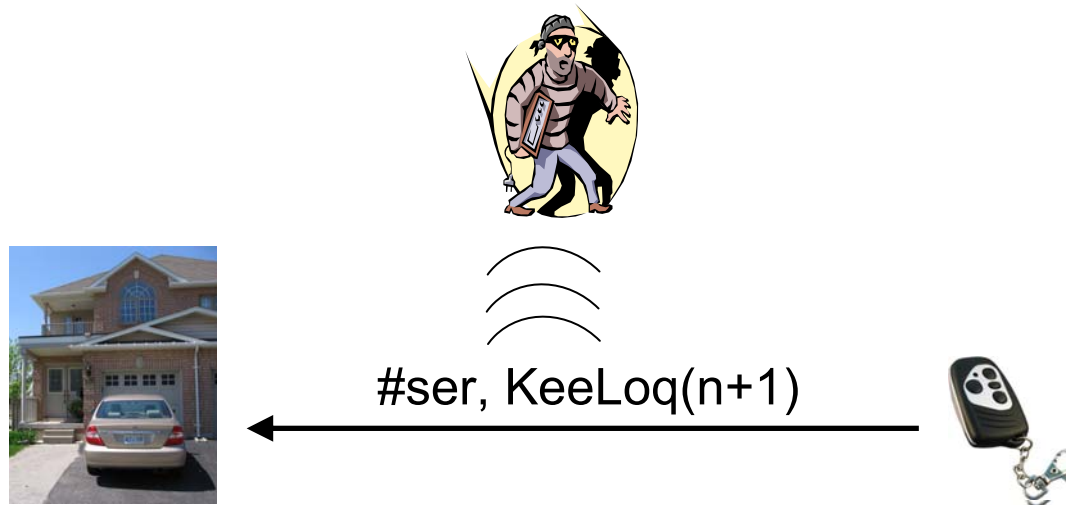
Recover manufacturer key and generate new remotes



# So what can we do now (2) ?

After extracting of manufacturing key:

**Remotely eavesdrop on 1-2 communications & clone key!**



- works for all key derivation schemes
- might require a few hours of computation  
(Rem: not necessary for any system we've analysed.)
- SCA attack is not specific to KeeLoq, e.g., unprotected AES is vulnerable too.

**! Side-channel step (recovery of manufacturer key, difficult)  
can be outsourced to criminal cryptographers !**

# Overview

- Embedded Systems and Security
- Case Study 1: Securing RFID
- Case Study 2: High Speed Signature Engine
- Case Study 3: High-Speed Vehicular Communication Engine
- **Related Activities**

# Related Workshops



**SECSI – Secure Component and Systems Identification**  
March 2008, Berlin

**RFIDSec 2008**  
July 2008, Budapest



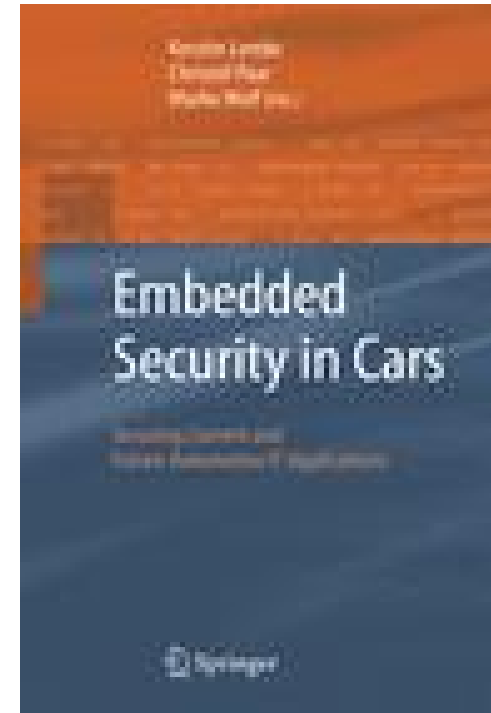
**CHES – Cryptographic Hardware and Embedded Systems**  
August 2008, Washington D.C.

**escar – Embedded Security in Cars**  
November 2008, Hamburg



## ... and a related book

1. Part  
**Embedded technologies in general**
2. Part  
**Security issues in cars**
3. Part  
**Business & security**



Lemke, Paar, Wolf: Embedded Security in Cars, by Springer

Thank you for your attention!

---

Christof Paar

[www.crypto.rub.de](http://www.crypto.rub.de)

[cpaar@crypto.rub.de](mailto:cpaar@crypto.rub.de)

