The embedded security challenge: Protecting bits at rest

Patrick Schaumont
schaum@vt.edu

Acknowledgements: Eric Simpson, Pengyuan Yu

Secure Embedded Systems Group
ECE Department

VirginiaTech
Invent the Future

Secret bits-at-rest

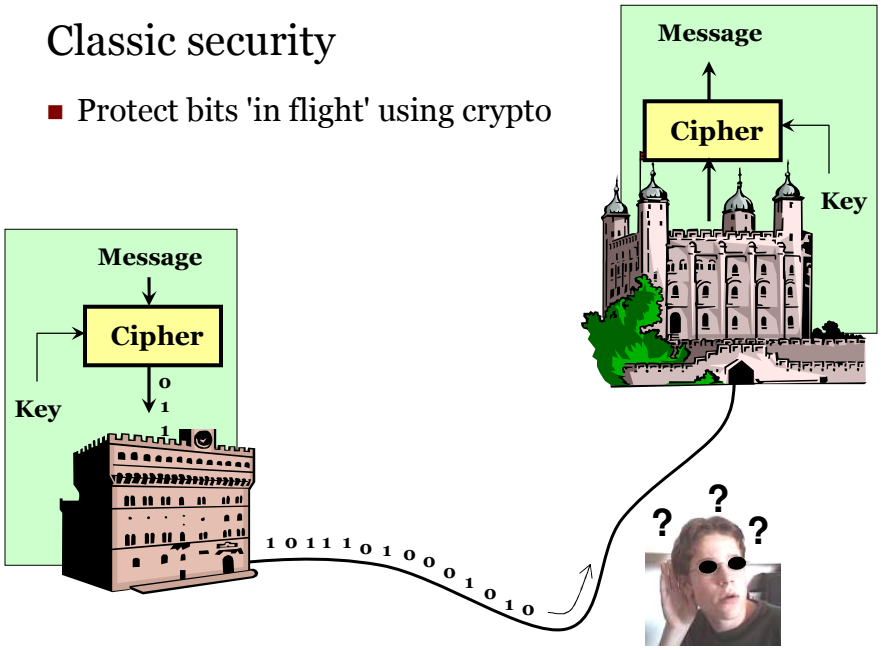
Hi-Res Digitized Signature

Car Unlock Code

AACS Keys (Blu-Ray DVD)

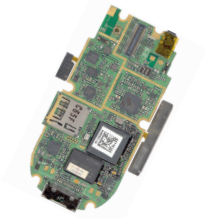
Classic security

- Protect bits 'in flight' using crypto



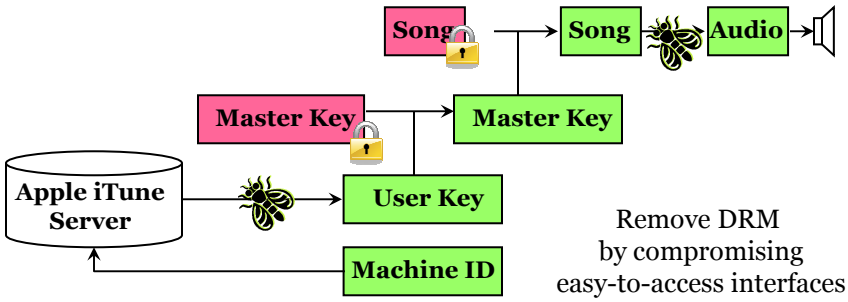
Embedded security

- Need more than crypto to protect bits 'at-rest'



Keeping secrets in software doesn't work

- Example - Fairplay encryption scheme of Apple

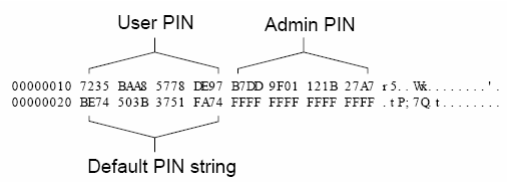


Remove DRM by compromising easy-to-access interfaces

- April 2007: Steve Jobs announces iTunes will sell DRM-free songs

Keeping secrets in hardware doesn't work

- Example: Aladdin eToken (2001) stores PIN in plaintext



[grandideastudios.com]

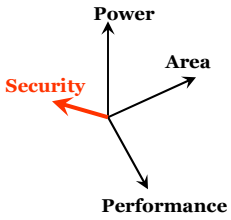
- April 2007: Secustick ('self-destruct thumbdrive') is broken by means of a trivial hack

[tweakers.net/reviews/683/1]

We need a secure design *methodology*

- Methodology - series of steps that can be learned and repeated.

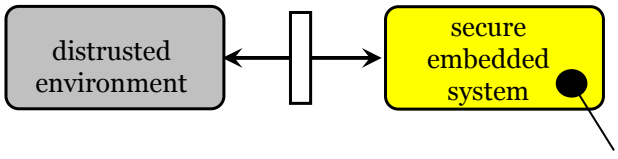
- *Zero-risk* security does not exist
  - *Zero-power* design does not exist either
- Low-risk security can be achieved
  - *Low-power* operation can be achieved



- Objective of secure design methodology: minimize spatial and temporal footprint of secret bits in embedded systems

The starting point: Root of Trust

- A secret in a box by itself is useless (useless like a key without a matching door-lock)
- So a secure system contains at least two parts



security policy

- authentication
- integrity
- confidentiality
- non-repudiation
- availability

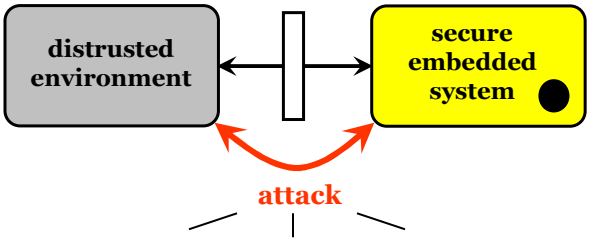
root-of-trust

The part that always works as expected (or so you think!)

Secret bits are inside of root-of-trust

Characteristics of attacks

- An 'attack' is an interface into the root-of-trust around the security policy

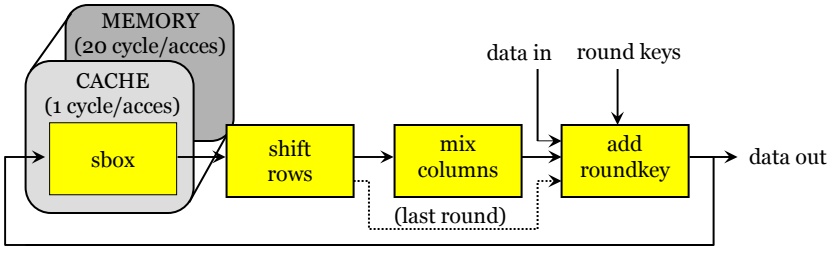


abstraction level
- logical
- side-channel
- physical

attacker proximity
- off-line
- connected
- close-range
- physical

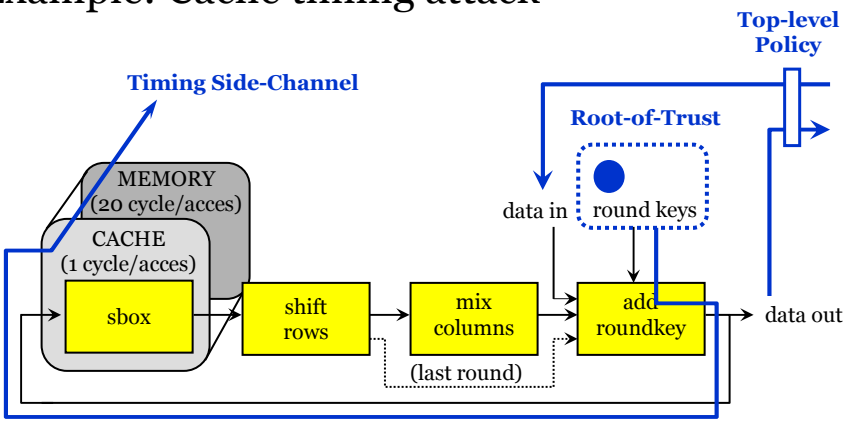
active/passive invasive/non-invasive
- side-channel
- fault-based
- spoof

Example: Cache timing attack



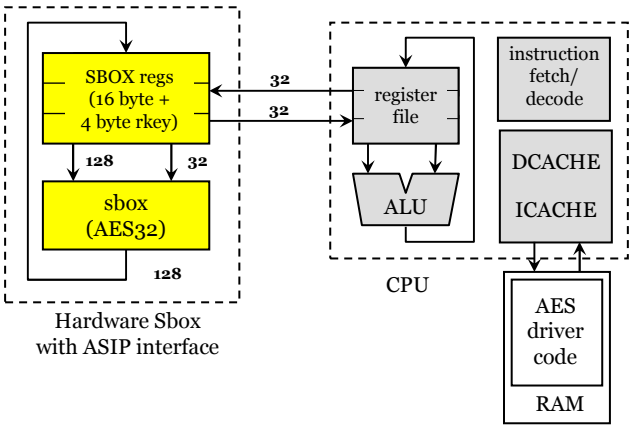
- Execution time dependency due to cache conflicts
- Software Solution: constant-time crypto (hard)

Example: Cache timing attack

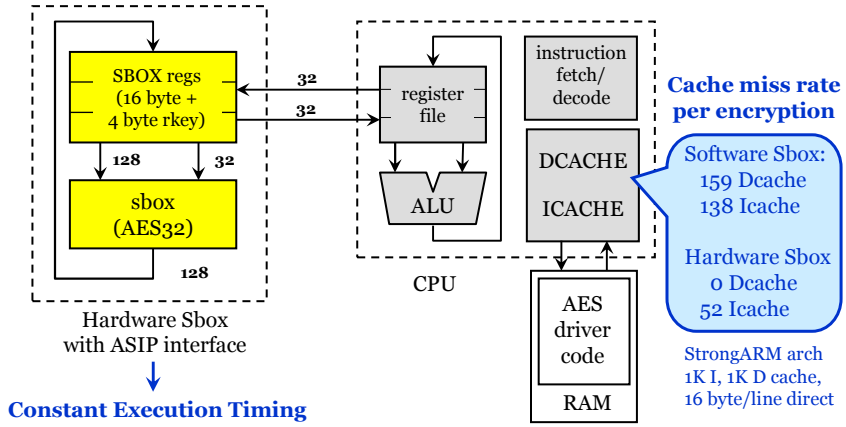


- Execution time dependency due to cache conflicts
- Software Solution: constant-time crypto (hard)

Moving Sbox into Hardware



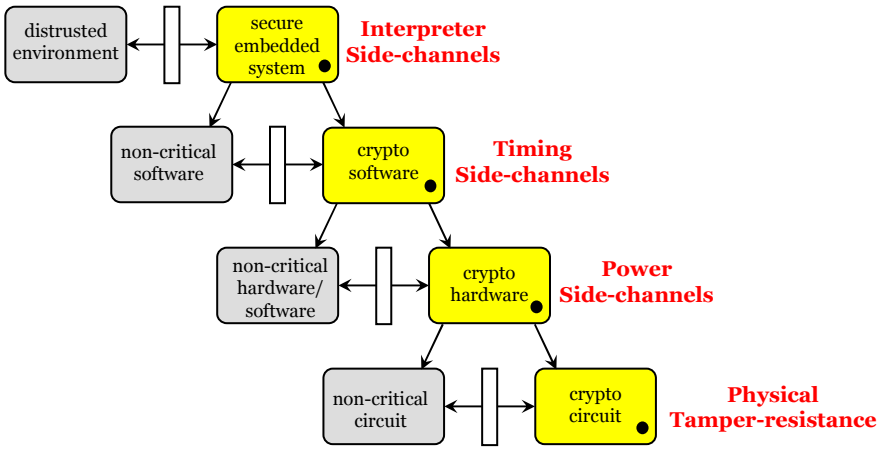
Moving Sbox into Hardware



- Timing Side-channel fixed, but others remain ..

Design Step: Secure partitioning

- Secure design over multiple abstraction levels (protocol, software, hardware, circuits)



Applications

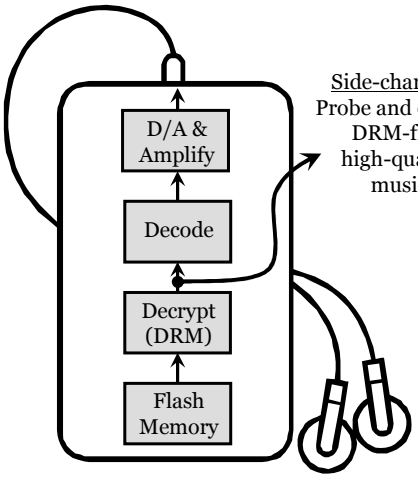
- Hardware Chain-of-Trust
  - Can we build a path in a device that is completely trusted, even as it extends into hardware?
- Side-channel resistant hardware in FPGA
  - Can we port side-channel resistant design styles for ASIC into FPGA while maintaining their properties?

Applications

- Hardware Chain-of-Trust
  - Can we build a path in a device that is completely trusted, even as it extends into hardware?
- Side-channel resistant hardware in FPGA
  - Can we port side-channel resistant design styles for ASIC into FPGA while maintaining their properties?

Hardware Chain of Trust for DRM - Why ?

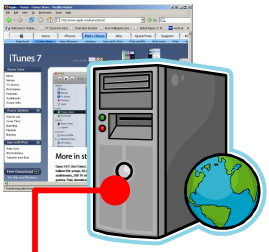
Download copy-righted multimedia into player



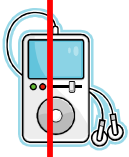
Side-channel:
Probe and extract DRM-free high-quality music

5/18/2007

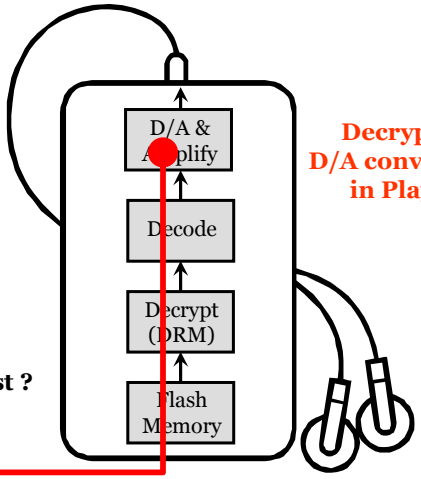
Hardware Chain of Trust for DRM - Principle



Encrypt at Server Hard-drive



Challenge: How to build this chain-of-trust ?



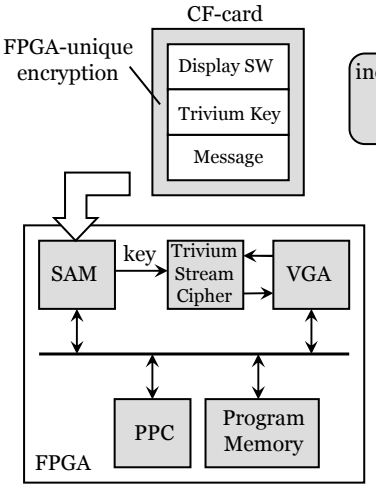
Decrypt at D/A conversion in Player

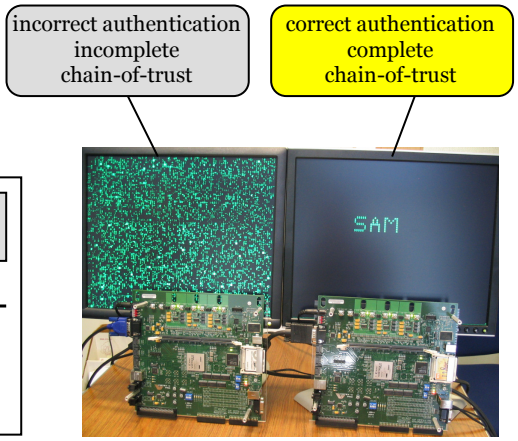
Hardware-specific Key-exchange

5/18/2007

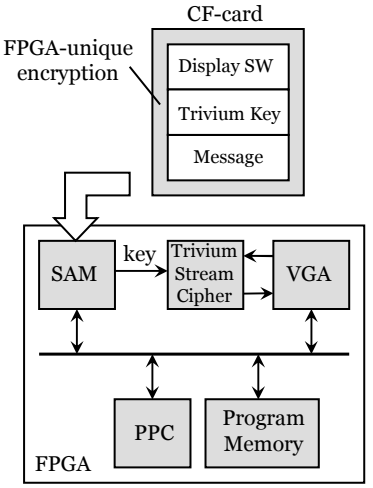
Chain-of-trust for Video Messaging

[Eric Simpson, VT]

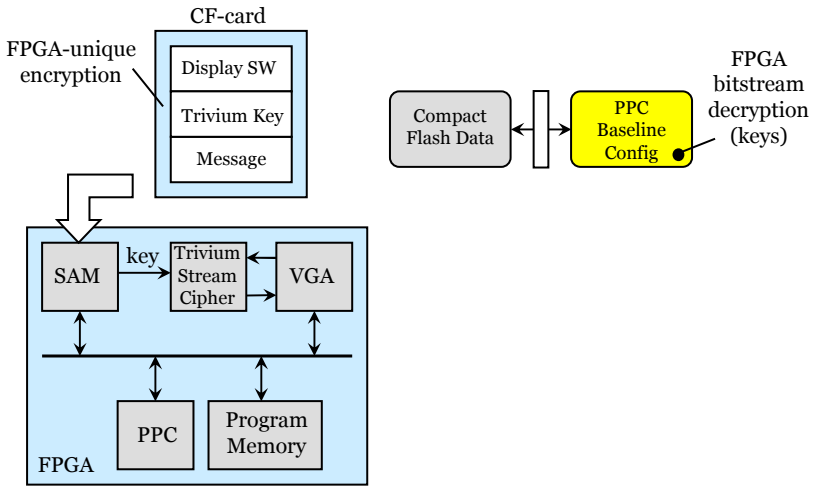




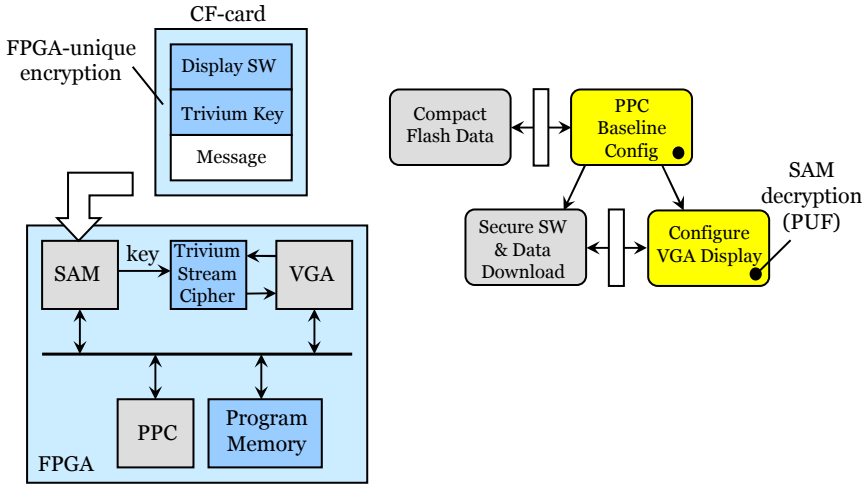
Chain-of-trust for Video Messaging



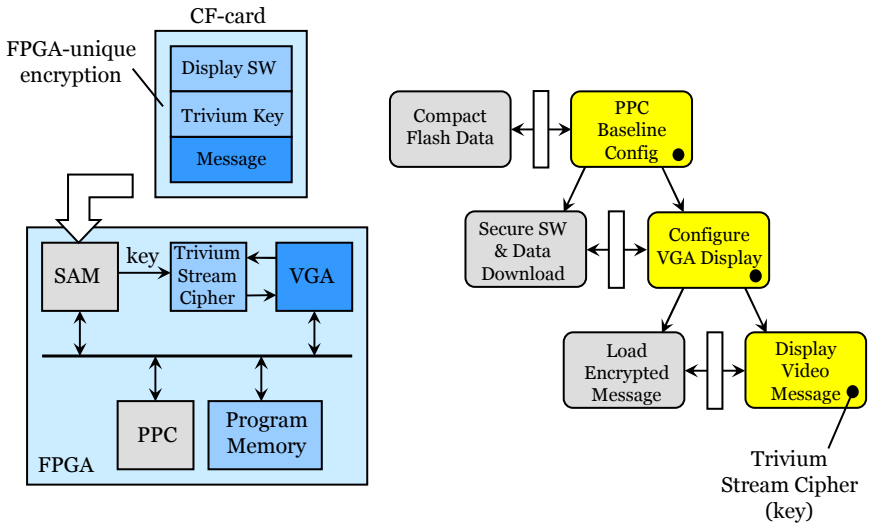
Chain-of-trust for Video Messaging



Chain-of-trust for Video Messaging



Chain-of-trust for Video Messaging

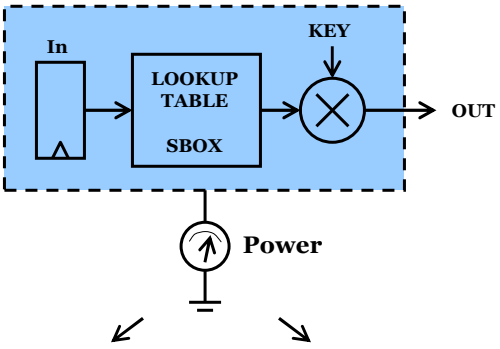


Applications

- Hardware Chain-of-Trust
  - Can we build a path in a device that is completely trusted, even as it extends into hardware?

- Side-channel resistant hardware in FPGA
  - Can we port side-channel resistant design styles for ASIC into FPGA while maintaining their properties?

Side-channel resistant design

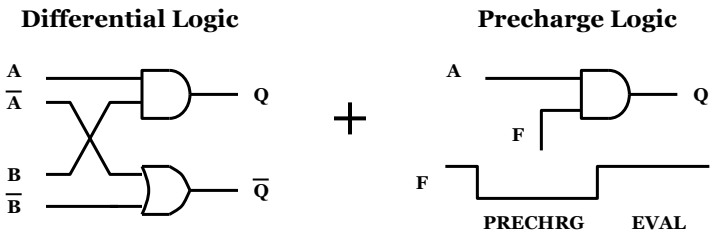


Reduce Power Variation
Constant-Power Design
Wave Dynamic Differential Logic

De-correlate Power Variation
Masking
Random Switching Logic

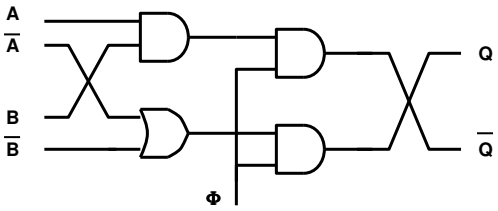
Reducing Power Variations: WDDL

- Wave Dynamic Differential Logic (Tiri 2003)
  Gates have exactly one 0->1 transition per clock cycle



- Differential output ensures each switch contains 0->1
- Precharge output guarantees switching each cycle

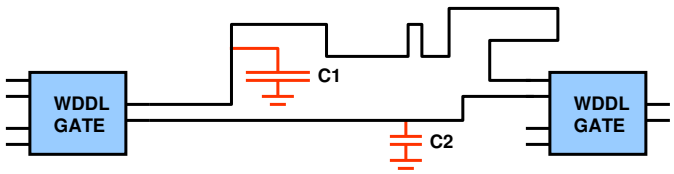
WDDL NAND Gate



F	PRE	EVAL	PRE	EVAL
A	0	1	0	1
B	0	0	0	1
Q	0	1	0	0
$\overline{Q}$	0	0	0	1

- Exactly one switching event per gate per cycle

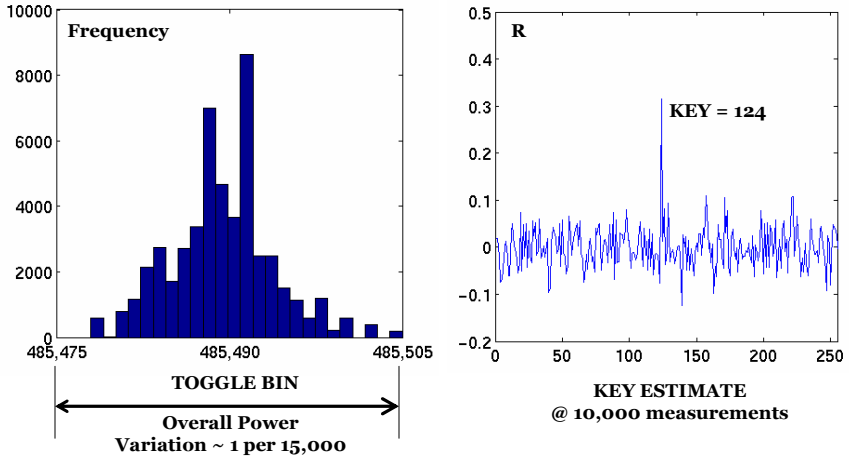
Matching Interconnect Capacitance



- Asymmetry in C-load gives residual power leakage
- Need symmetrical place-and-route technique, not easy in contemporary tools

Unbalanced WDDL easy to break

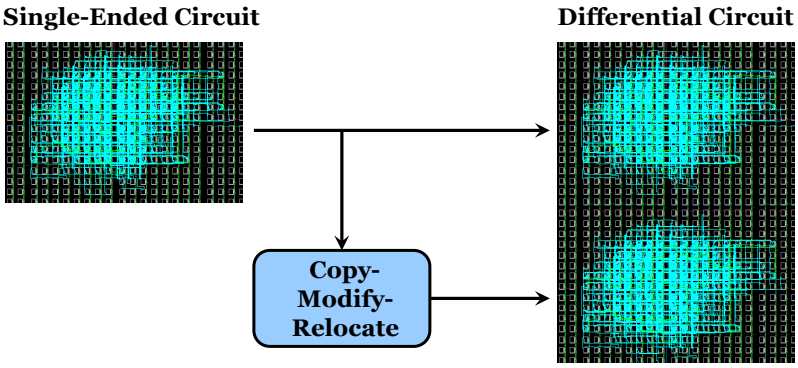
- Impact of imbalance of 1 part in 500 per WDDL net



WDDL in FPGA

[Pengyuan Yu, VT]

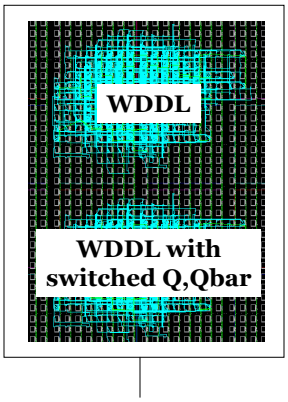
- Use FPGA fabric regularity to build a better WDDL



- 'Copy' + 'Relocate' results in identical routing pattern
- 'Modify' LUT content creates complementary logic function

Maping WDDL in FPGA

- Create complementary function starting from differential netlist by switching Q and Qbar



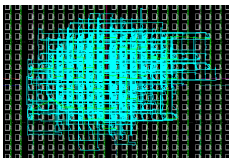
- 4X in area over single-ended

+ same advantages as WDDL with symmetrical routing

'DWDDL', Double Wave Dynamic Differential Logic

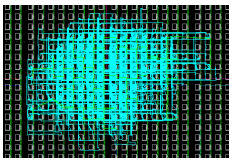
Test Setup considers 3 possible cases

Single-ended



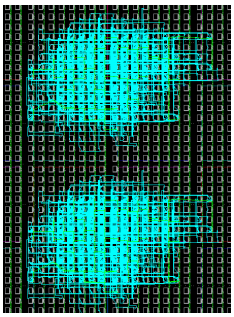
SE

WDDL



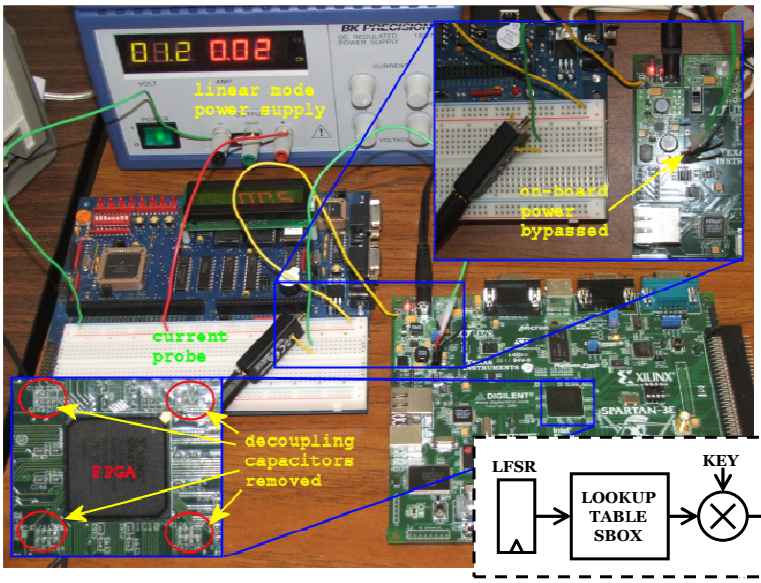
WDDL

WDDL + Complementary

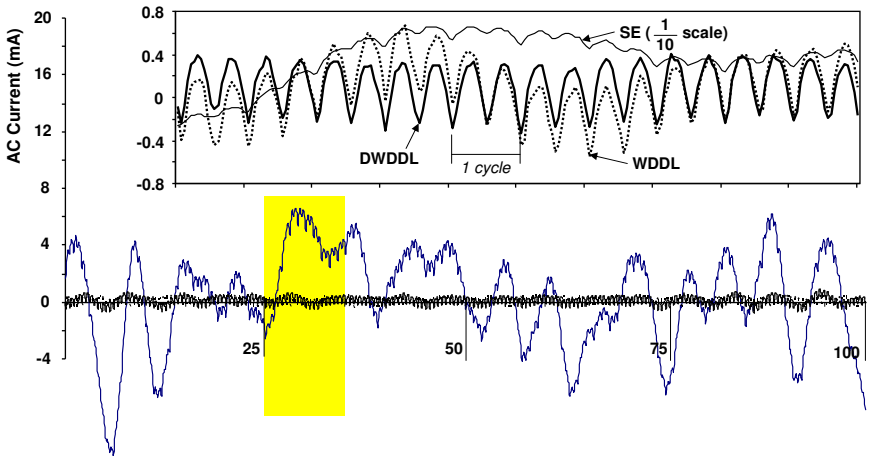


DWDDL

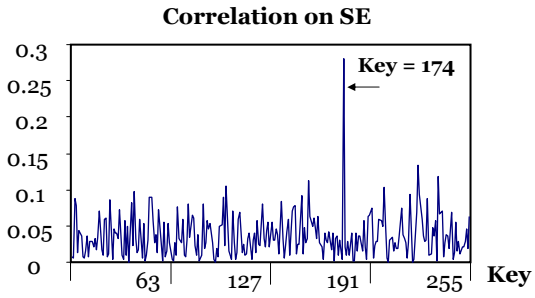
Test setup



Measurements



DPA on Measurements

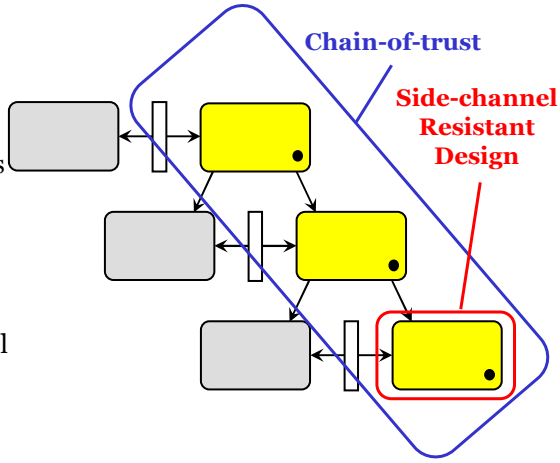


Correlation peak position on dual-rail design

Bit	0	1	2	3	4	5	6	7
WDDL	19	152	68	174	174	99	174	174
DWDDL	194	174	64	27	190	238	10	113

Conclusions

- In embedded systems, ZERO-risk security does NOT exist
- In embedded systems, LOW-risk security IS possible
- Methodology is essential
- Many open problems
  - How to **quantify** security? Number of measurements? Cost versus Security trade-offs?
  - Can we build **tools** to automate analysis and secure design ?



Thanks !

Patrick Schaumont, Eric Simpson, Pengyuan Yu

Secure Embedded Systems Group
ECE Department

VirginiaTech
Invent the Future