Hardware Trojans: Taxonomy and Detection Methods

Julien Francq
julien.francq@cassidian.com

Cassidian CyberSecurity

2014, July the 15th
Outline

The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods

Design for Hardware Trust
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods
   Overview
   Logic Testing: Challenges & Solutions
   Side-Channels: Challenges & Solutions
   Some Subtleties
   Summary

Design for Hardware Trust
HINT Project Profile

- **HINT** = Holistic Approaches for Integrity of ICT-Systems
- **Project Number:** 317930
- **Project website:** www.hint-project.eu
- **Project start:** October 1, 2012
- **Project duration:** 3 years
- **Total Costs:** €5.103.893
- **EC-Contribuition:** €3.350.000
- **Project is co-financed by the European Commission under Seventh Framework Programme**
Missions

- Development of a common framework for system integrity checking
- Use developed technologies on real-time applications
- Prepare adoption by future security evaluation schemes
Motivation

- **Authenticity and integrity** of hardware components in modern ICT systems
- Security challenged by improving attacks
  - Recent trends:
    - Counterfeiting of hardware components
    - “Hardware Trojans”: Hidden functions in Integrated Circuits
- HINT proposal:
  - Novel technologies to support assurance of genuineness and integrity
Technical Approach

- Holistic Integrity Checking for Components in ICT-Systems
Objectives

- **Main objective:** Improve security of architectures and platforms based on tamper-resistant integrated circuits
- **Development of methods to:**
  - Perform **at-time-of-use integrated checking** of the global integrity of a system for hardware and embedded software
  - Check the “**genuineness**” of the secure integrated circuits by detecting functional clones or counterfeited circuits
  - Detect the presence of **Hardware Trojans**
- **Main technologies used:**
  - **Physically Unclonable Functions**, enabling to authenticate a hardware component using a physical, intrinsic and unique property of the device
  - **Side Channel based analysis** to monitor the behaviour of hardware components and to detect changes from their original specifications and implementations
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods
  Overview
  Logic Testing: Challenges & Solutions
  Side-Channels: Challenges & Solutions
  Some Subtleties
  Summary

Design for Hardware Trust
Hardware Trojan (HT)

- Malicious modifications of an Integrated Circuit (IC) during its design flow

**Fake** Counterfeiting has become a big problem for the U.S. military, and bogus packaging could disguise a questionable chip as a legitimate one. ... & **BAKE** Baking a chip for 24 hours after fabrication could shorten its lifespan from 15 years to a scant 6 months.

**Nick the Wire**
A notch in a few interconnects would be almost impossible to detect but would cause eventual mechanical failure as the wire became overloaded.

**Add Extra Transistors**
Adding just 1000 extra transistors during either the design or the fabrication process could create a kill switch or a trapdoor. Extra transistors could enable access for a hidden code that shuts off all or part of the chip.

**Add or Reconnect Wiring**
During the layout process, new circuit traces and wiring can be added to the circuit. A skilled engineer familiar with the chip’s blueprints could reconnect the wires that connect transistors, adding gates and hooking them up using a process called circuit editing.
Context

- **Outsourcing** of the fabrication of the ICs
- Difficult to ensure the **trust** in all the steps of the design flow
Hardware Trojans in Practice

- **2005**: US Department of Defense
- **2007**: DARPA “Trust in IC Program”
- **2009**: “Hot Topic” of CHES conference
- **After 2009**: other conferences (DATE, HOST, CARDIS, ReConFig, etc.)
- [Skorobogatov *et al.*: “Breaktrought Silicon Scanning Discovers Backdoor in Military Chip”, CHES 2012]
- [Becker *et al.*: “Stealthy Dopant-Level Hardware Trojans”, CHES 2013]
- Research projects like HINT (European funded)
- ⇒ HTs: real and emerging threat
Possible Payloads

- **Kill switch**
  - Fighters
- **Dysfonctional circuit**
  - Satellite which works only 6 months
- **Secret information leakage**
  - Ciphered communications
- Help a **malware** by providing a **backdoor**
  - Privilege escalation, automatic login, password theft
- Prevent from going to **sleep mode**
  - Autonomy
- etc.
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods
  Overview
  Logic Testing: Challenges & Solutions
  Side-Channels: Challenges & Solutions
  Some Subtleties
  Summary

Design for Hardware Trust
Hardware Trojan Taxonomy

- **Taxonomy**: tree where each branch defines a different property
- In the ideal case, a specific HT must be on only one leaf of the tree

Benefits of the taxonomy

- **Systematic study** of their characteristics
- **Specific detection methods** for each HT class
- **Benchmark circuits** for each class

- Best existing taxonomy: **Trust-Hub**
Factoring the Taxonomy

- $4 \times 5 \times 5 \times 6 \times 5 = 3000$ different HTs!
- Very rich taxonomy!
- Impossible to implement them all, and then detect them
- $\Rightarrow$ Factoring this taxonomy
- Total: $\sim 100$ HTs
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

**HT Detection Methods**
- Overview
- Logic Testing: Challenges & Solutions
- Side-Channels: Challenges & Solutions
- Some Subtleties
- Summary

Design for Hardware Trust
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

**HT Detection Methods**

**Overview**

Logic Testing: Challenges & Solutions
Side-Channels: Challenges & Solutions
Some Subtleties
Summary

Design for Hardware Trust
HT Detection Methods Overview

- Post Production Detection
  - Destructive
    - Optical
  - Non-Destructive
    - Run-Time
    - Test-Time
      - Side Channel Analysis
      - Logic Testing

- Prevention
  - Supportive Design
  - Secure Design
  - Trusted Production

▶ No method is 100% successful!
Detect HTs? Not so easy...

1. **Systems on Chip** are more and more **complex**, and detecting a small malicious modification is difficult
2. Reverse-engineering inspection is costly and difficult
   - No guarantee that the remaining ICs are HT-free
3. By nature, HTs are designed to be **stealthy**
   - Not easily **detectable** with conventional logic testing
4. By nature, HTs are **small** to be not easily detected by optical analysis
   - Difficult to detect them with **side-channel** (power consumption, electromagnetic radiations, *etc.*) analysis
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

**HT Detection Methods**

- Overview
- Logic Testing: Challenges & Solutions
- Side-Channels: Challenges & Solutions
- Some Subtleties
- Summary

Design for Hardware Trust
Test Generation (1/2)

- Conventional logic testing cannot be used to reliably detect HT.
- Manufacturing defects (stuck-at-faults) ≠ HT effects.
- Difficult to trigger a HT.
  - *Time-bombs*
- Some HTs have no impact on functional outputs (*Trojan Side-Channels*).
- Vast spectrum of possible HTs.
Test Generation (2/2)

- HTs are on low controllability and observability nodes for a rare triggering
- Extremely challenging to exhaustively generate test vectors for triggering a HT
Deterministic vs. Probabilistic Approach

- **Deterministic approach** difficult
  - Many possible HTs
  - Function of some IC nodes
  - ⇒ Exhaustive enumeration impossible

- **Statistic approach**:
  1. Find rare events in the circuit
  2. Get a list of HTs which can be inserted
  3. Generate test vectors and estimate their coverage
  4. ⇒ Set of high quality test vectors

- 85% reduction in testset length compared to a random approach, but less efficient with big triggers and takes a long time
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods
- Overview
- Logic Testing: Challenges & Solutions
- Side-Channels: Challenges & Solutions
- Some Subtleties
- Summary

Design for Hardware Trust
Side-Channel Analysis

- Any HT in the IC should modify its leakage current (IDDQ), dynamic power trace (IDDT), path-delay characteristic, ElectroMagnetic (EM) radiation.
- Don’t need to trigger a HT for measuring its effects
- Test vectors generation easier than for logic testing
- Needs HT-free circuits
  - Get side-channel measurements and then reverse-engineering to check if the IC is HT-free
- If so, the measurements become a reference, and we can then compare the side-channels of the other circuits
Global Side-Channel Analysis

- Green: RSA signal
- Red: Process noise (offset)
- Black: HT signal (offset)
Local Side-Channel Analysis

- Local Side-Channel Analysis more efficient than global ones
- Needs again HT-free circuits

- Maximize/Minimize the activity of some IC areas
Noise and Sensitivity

- Histogram showing the number of dles for Golden ICs, Trojan ICs, False Negatives, and False Positives.

- Graph depicting Sensitivity vs. Trojan Size, with data points for Low activity patterns and High activity patterns.

- Graph showing the change in trojan power ($\Delta P_{\text{dd}}/P_{\text{dd}}$) with respect to Trojan size (number of bits).
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

**HT Detection Methods**

*Overview*

Logic Testing: Challenges & Solutions
Side-Channels: Challenges & Solutions

**Some Subtleties**

Summary

Design for Hardware Trust
Some Subtleties

- Added circuitry for the HT detection must not be infected itself
  - At best, the added circuitry is disabled (e.g., fault countermeasure)
  - At worst, it can be turned into a backdoor (e.g., scan chain)
- A HT triggering logic can exploit the “Test/Scan Enable” control line to disable itself
- Parametric HTs very difficult to detect
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

**HT Detection Methods**
- Overview
- Logic Testing: Challenges & Solutions
- Side-Channels: Challenges & Solutions
- Some Subtleties
- Summary

Design for Hardware Trust
Summary

- Complementary methods
- Combine test-time and run-time methods
- Modify the IC for assistive and preventive methods
  - ⇒ Design for Hardware Trust

<table>
<thead>
<tr>
<th>Logic testing approach</th>
<th>Side-channel approach</th>
</tr>
</thead>
<tbody>
<tr>
<td>Pros</td>
<td>Pros</td>
</tr>
<tr>
<td>(a) Effective for small Trojans</td>
<td>(a) Effective for large Trojans</td>
</tr>
<tr>
<td>(b) Robust under process noise</td>
<td>(b) Test generation is easy</td>
</tr>
<tr>
<td>Cons</td>
<td>Cons</td>
</tr>
<tr>
<td>(a) Test generation is complex</td>
<td>(a) Vulnerable to process noise</td>
</tr>
<tr>
<td>(b) Large Trojan detection challenging</td>
<td>(b) Small Trojan detection challenging</td>
</tr>
</tbody>
</table>
The HINT Project

Introduction to Hardware Trojans

Hardware Trojan Taxonomy

HT Detection Methods

Overview
Logic Testing: Challenges & Solutions
Side-Channels: Challenges & Solutions
Some Subtleties
Summary

Design for Hardware Trust
Introduction

- To improve HT detection rate, modify the IC
- \(\Rightarrow\) Design for Hardware Trust
  - Prevent from the insertion of HT
  - Ease side-channel analysis and logic testing
- 4 main methods:
  - Delay-Based Methods
  - Rare Event Removal
  - Design for Trojan Test
  - Proof-Carrying Hardware
- Run-Time Detection Methods
Run-Time Methods

- Last line of defense
- On-line monitoring of the IC in real-time, for checks:
  - Critical operations,
  - Idle mode,
  - Security policies,
  - Performance or availability of some units,
  - etc.
- Costly
Run-Time Methods

- Disable one suspect block or force one operation
- SPN: Signal Probe Network
- SM: Security Monitor (∼ FSM)
- SECOPRO: Security and Control Processor
- Configurations ciphered and stored in secured Flash memory
- Overhead?
Conclusion

- Hardware Trojans are **real threats** for integrated circuits
- HT **taxonomy** is very rich
- No HT detection method of the state-of-the-art is **100% successful**
- **3 lines of defense:**
  - Design for Hardware Trust
  - Test-Time Methods
  - Run-Time Methods
- A European initiative: **HINT** project
  - Let’s talk about it during the coffee breaks!
- Very **encouraging** first results:
  - Infected benchmark circuits are available
  - Detection with **side-channel analysis**
  - Internal circuit delays extraction
Thanks! Questions?