Hardware Security for FPGAs using Cryptography

Jens Hüttemann, Microsemi Corporation
Senior Field Applications Engineer
Definitions: Design Security vs. Data Security

- FPGA Design/Device Security
  - Making sure that the **FPGA Design** is protected and the IP owner’s security intent is respected

- Data Security
  - The **Application** programmed into the device meets its security objectives (authenticity, confidentiality, integrity, etc.)
FPGA Technologies Compared

- FPGA reloaded after every power cycle
- Bitstream exposed at board level
- Application keys must be stored externally
- Bitstream keys (if encryption is even available) may also be volatile, requiring a long-term battery to maintain them

- FPGA NOT reloaded after every power cycle
- Bitstream NOT exposed
- All keys, for bitstream AND (optionally) for end application, are NON volatile and safely stored long-term inside the device
Design Security
Why is FPGA Design Security Important?

- **Cloning**
  - Someone copies your design without even necessarily having to understand how it works

- **Overbuilding**
  - Your contract manufacturer fills your order… then makes a few for himself. After all, he has all the data!

- **Reverse Engineering**
  - Someone figures out how your design works, then uses or improves on what he learned

- **Counterfeiting**
  - Illegal use of your brand name on a work-alike or cloned product

- **Data Security**
  - Without design/device security, it is virtually impossible to provide good data security

Potential Lost Revenue, Increased Costs, Unhappy Customers, Poor Reputation
Typical SRAM FPGA Hardware Security Architecture

- AES Decryption
- Cyclic Redundancy Check (CRC)
- AES Key
- Security Lock-bit(s)

Components:
- External Flash PROM
- SRAM FPGA Fabric Configuration
- OTP (anti-fuse)
- SRAM
- Flash
- Logic

Key

Microsemi
Typical Flash FPGA Hardware Security Architecture
Data Security
Some Electronic Security Applications

- Encryption/decryption of [network] data
- Enciphered storage of [hard or flash disk] data
- Identification/authentication of HW/SW/persons (e.g., biometrics)
- Secure tokens/one-time passwords
- Anti-cloning/overbuilding
- Anti-counterfeiting
- Licensing
- Key management
- Tethering (tying SW to a specific HW platform)
- Anti-piracy
- Anti-reverse engineering
- Digital rights management (DRM)
- Anti-tamper
- Secure/trusted software
- Watermarking/fingerprinting and forensics
- Gaming
- Traditional physical security (locks, surveillance, alarms, etc.)
Security Boundary

In general, there are two ways to secure a device:

- Put it in a secure location, such as a limited-access computer server room, and secure the communications interfaces that leave the room.
- If the unit is exposed to malicious physical attacks, then the device will have to be tamper-resistant, and protect itself.

Flash FPGAs can be used as a single-chip security module, or part of a larger system.
Hardware Root-of-Trust

- Fielded systems need a **HW Root-of-Trust** (RoT)
  - *SW by itself is not secure*

- The HW RoT can store keys and perform operations that extend the **trust zone** to cover other parts of the system.

- For example
  - Execute secure boot code
  - Check signatures on SW stored in external memory
  - Check that other boards in the system are not counterfeit
  - Prevent overbuilding or cloning of systems

*Nonvolatile flash-based key and code storage is an advantage*
HW Root-of-Trust Example: Set-Top Box

Smart Card
Threats and Attacks
Threats: Who, What, Why, How

- **Who**
  - Nation states
  - The competition
  - Contract manufacturers (CMs)
  - University researchers
  - Crooks (e.g., organized crime)
  - Kids

- **What**
  - Find keys
  - Steal bitstreams/IP
  - Insert Trojan Horse
  - Learn sensitive data
  - Deny service

- **Why**
  - Steal IP to use or sell
  - Steal IP to reverse engineer
  - Clone/overbuild for “profit”
    - From FPGAs to full systems
  - Counterfeit or 3\(^{rd}\)-party products/accessories
    - From ink cartridges to Cisco boards
  - Subvert security measures
    - Financial/identity fraud/theft
    - Military and state secrets
    - Wreak havoc (e.g., power grid)
  - Fun or fame

- **How**
  - Con humans
  - Old-fashioned theft
  - Bribery or blackmail
  - Technical attacks
Hierarchy of Technical Attacks

- **Protocol**
  - Eavesdropping
  - Man-in-the-middle
  - Replay
  - Relay
  - Other protocol design attacks

- **Side-Channel (non-invasive, passive)**
  - Timing
  - Simple power analysis
  - Differential power analysis
  - Electromagnetic emanations

- **Active (non-invasive)**
  - Fault injection (laser or power glitch)
  - Temperature (cold SRAM effect)
  - Clock manipulation

- **Semi-invasive and invasive**
  - Optical
  - Electron microscope
  - Focused ion beam (FIB)

- **Cryptographic**

---

General Cost Guideline

- High School Hacker
- Universities, companies, organized crime
- Well-funded adversaries, national labs

Increasing Cost
Side-Channel Analysis

Cryptographic Operation (e.g., Decryption)

Security Boundary

Input Data (e.g., Ciphertext)

Output Data (e.g., Plaintext)

“Main” Channel

Unintended “Side” Channel(s)

Side-Channel Analysis

Timing Information
Power Dissipation
Electromagnetic Fields
Light
Types of Side-Channel Attacks

- **Timing Analysis**: Observe differences in timing due to program branches, cache hits, etc (Differences in system response time, or combined with SPA-type power measurements)

- **Simple Power Analysis (SPA)**: Power supply currents may leak secret information directly

- **Differential Power Analysis (DPA)**: Statistical techniques to correlate power supply information leakage to recover critical information such as keys

- **Electromagnetic Analyses (EMA)**: Most Power Analyses have a radiated emissions analog
  
  - Note: “TEMPEST” is an NSA codename for standards to limit or analyze EM emissions, (or, more generally, Emissions Security, EMSEC) at a system level
Side-Channel Analogy – Mechanical Safe

- Six tumbler safe, 100 tick-marks
  \[100^6 = 1,000,000,000,000\] possible combinations
- Un-crackable, right?
- Wrong!
- Safe has a side-channel:
  - Tactile feedback from each tumbler
  - This allows cracking the safe one tumbler at a time
  - “Divide and conquer” approach (6 x 100) is much easier than “all at once” approach (100^6)
Typical Power and EM Analysis Set-Up

[ChangKyun Kim, Martin Schläffer, and SangJae Moon, ETRI Journal, April 2008]
Simple Power Analysis (SPA)

- IC power consumption depends on activity of transistors
- Measurements of device operation can directly reveal keys and other secrets

Plots courtesy of Cryptography Research, Inc.

Freescale MC908AZ60A password attack

Current traces for 5 different values of password byte 1

- wrong inputs: min/max measured currents
- wrong inputs: min/max difference to median
- correct input: current
- correct input: difference to median

Plot courtesy of Sergei Skorobogatov
Simple Electro-Magnetic Analysis (SEMA)

FPGA and antenna

EM Trace
Insecure ECC implementation

[E. Demulder EUROCON 2005]

ECC = Elliptic Curve Cryptography
Differential Power Analysis (DPA)

- Observe a series of cryptographic transactions
- Apply statistical tests to correlations in computational intermediates
- Results recover the key and other secrets
- First published in 1999 by Paul Kocher and his associates at Cryptography Research, Inc.
  - Major impact on development of the SmartCards used in banking, set-top boxes, etc.

Random low correlation
(x100 gain)

Correct key guess (an 8-bit portion of key)
Shows correlation peaks
(x100 gain)

Incorrect key guess
Random low correlation
(x100 gain)

[Paul Kocher et al, Proc. CRYPTO 1999]
DPA of a cipher implemented in an FPGA

1 correct key guess shown with **black** trace (an 8-bit portion of key guessed)
255 incorrect key guesses shown with **gray** traces

Correlation estimate time trace for correct key guess shows correlation peak well differentiated from traces for 255 incorrect key guesses (2000 measurements with random ciphertext used for correlation estimates)

Correlation estimates improve with the number of measurements used.

Correct key guess stands out from the 255 incorrect key guesses when 200 traces (or more) are used

[ChangKyun Kim, Martin Schläffer, and SangJae Moon, ETRI Journal, April 2008]
DPA Countermeasures Patent Situation

- Paul Kocher, et al, at Cryptography Research, Inc. (CRI) discovered Timing Analysis, SPA, and DPA in the late ‘90s
  - They then invented and patented many of the fundamental countermeasures, which CRI licenses to hardware manufacturers
  - Today, an estimated 5.5 billion ICs per year are manufactured under DPA patent portfolio licenses, comprised of over 50 CRI-owned patents

- Microsemi is, so far, the only major FPGA supplier to obtain a DPA patent portfolio license from CRI
  - Microsemi is licensed to use DPA countermeasures for improved FPGA design security, e.g., bitstream decryption
  - Also, Microsemi can pass-through a “lawyer-free” DPA license to end users by pre-paying the CRI royalty on any device it sells

- An ecosystem of IP suppliers is growing who, like ESCRYPPT, are licensed by CRI to develop and sell IP using the CRI patented techniques, so long as the IP buyers use it only in licensed (i.e., royalty-bearing) hardware
Conclusions

- SRAM and Flash FPGAs have fundamentally different security attributes

- Differential Power Analysis and related side-channel attacks are a serious threat...
  - Where systems are fielded and an attacker can make side-channel measurements
  - Where encryption is used to protect design IP, or
  - Where the system is performing calculations on secret data or keys
  - In most any electronic device: MCU, FPGA, ASIC, etc.

- Solutions are available today to mitigate DPA attacks
  - FPGAs that can be configured to be DPA-resistant for design IP
  - Systems designers can chose to make or buy solutions for data security applications involving symmetric or public-key cryptography
Thank You!
Questions?