Final Report: Design for Security Workshop

The views, opinions and/or findings contained in this report are those of the author(s) and should not contrived as an official Department of the Army position, policy or decision, unless so designated by other documentation.

Through sponsorship of the US Army Research Office, the University of Southern California Information Sciences Institute hosted a 1-day working meeting on the topic of Design for Security, July 23, 2014 at the Marina del Rey, CA location, where the primary focus was on electronics (ASICs, FPGAs, COTS, etc). In the past decade, more and more fabrication of advanced ICs has migrated offshore, largely because of global economic pressures. Fabrication facilities dedicated to supporting the Department of Defense can no longer provide the performance, variety, and volume of ICs at the cost needed. Such trends have raised concerns regarding the reliance of U.S. defense systems.
ABSTRACT

Through sponsorship of the US Army Research Office, the University of Southern California Information Sciences Institute hosted a 1-day working meeting on the topic of Design for Security, July 23, 2014 at the Marina del Rey, CA location, where the primary focus was on electronics (ASICs, FPGAs, COTS, etc). In the past decade, more and more fabrication of advanced ICs has migrated offshore, largely because of global economic pressures. Fabrication facilities dedicated to supporting the Department of Defense can no longer provide the performance, variety, and volume of ICs at the cost needed. Such trends have raised concerns regarding the reliance of U.S. defense systems on high-performance ICs and the potential vulnerabilities of these systems if fabricated and/or developed offshore. While previous programs, such as DARPA’s Trust in Integrated Circuits and Integrity and Reliability in Integrated Circuits, sought to address these concerns through hardware and design validation, the design perspective to explore what can be done during the design phase to increase the security of a system has not received equal attention. This workshop discussed how to incorporate security as a first-rate metric during the design flow, much like performance, area, and power and identified areas needing further investment.

Enter List of papers submitted or published that acknowledge ARO support from the start of the project to the date of this printing. List the papers, including journal references, in the following categories:

(a) Papers published in peer-reviewed journals (N/A for none)

<table>
<thead>
<tr>
<th>Received</th>
<th>Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOTAL:

Number of Papers published in peer-reviewed journals:

(b) Papers published in non-peer-reviewed journals (N/A for none)

<table>
<thead>
<tr>
<th>Received</th>
<th>Paper</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

TOTAL:

Number of Papers published in non peer-reviewed journals:

(c) Presentations
<table>
<thead>
<tr>
<th>Type</th>
<th>Received</th>
</tr>
</thead>
<tbody>
<tr>
<td>Non Peer-Reviewed Conference Proceeding publications (other than abstracts):</td>
<td></td>
</tr>
<tr>
<td>Peer-Reviewed Conference Proceeding publications (other than abstracts):</td>
<td></td>
</tr>
<tr>
<td>(d) Manuscripts</td>
<td></td>
</tr>
<tr>
<td>Books</td>
<td></td>
</tr>
</tbody>
</table>

**Number of Presentations:** 0.00

**TOTAL:**

**Number of Manuscripts:**

**TOTAL:**

**Number of Books:**
### Patents Submitted

### Patents Awarded

### Awards

### Graduate Students

<table>
<thead>
<tr>
<th>NAME</th>
<th>PERCENT_SUPPORTED</th>
<th>FTE Equivalent</th>
<th>Total Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.70</td>
<td>1</td>
</tr>
</tbody>
</table>

### Names of Post Doctorates

<table>
<thead>
<tr>
<th>NAME</th>
<th>PERCENT_SUPPORTED</th>
<th>FTE Equivalent</th>
<th>Total Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.70</td>
<td></td>
</tr>
</tbody>
</table>

### Names of Faculty Supported

<table>
<thead>
<tr>
<th>NAME</th>
<th>PERCENT_SUPPORTED</th>
<th>National Academy Member</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jeffrey Draper</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>FTE Equivalent:</td>
<td>0.70</td>
<td></td>
</tr>
<tr>
<td>Total Number:</td>
<td>1</td>
<td></td>
</tr>
</tbody>
</table>

### Names of Under Graduate students supported

<table>
<thead>
<tr>
<th>NAME</th>
<th>PERCENT_SUPPORTED</th>
<th>FTE Equivalent</th>
<th>Total Number</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
**Student Metrics**
This section only applies to graduating undergraduates supported by this agreement in this reporting period

The number of undergraduates funded by this agreement who graduated during this period: 0.00
The number of undergraduates funded by this agreement who graduated during this period with a degree in science, mathematics, engineering, or technology fields: 0.00
The number of undergraduates funded by your agreement who graduated during this period and will continue to pursue a graduate or Ph.D. degree in science, mathematics, engineering, or technology fields: 0.00
Number of graduating undergraduates who achieved a 3.5 GPA to 4.0 (4.0 max scale): 0.00
Number of graduating undergraduates funded by a DoD funded Center of Excellence grant for Education, Research and Engineering: 0.00
The number of undergraduates funded by your agreement who graduated during this period and intend to work for the Department of Defense: 0.00
The number of undergraduates funded by your agreement who graduated during this period and will receive scholarships or fellowships for further studies in science, mathematics, engineering or technology fields: 0.00

**Names of Personnel receiving masters degrees**

| NAME | Total Number:
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Names of personnel receiving PHDs**

| NAME | Total Number:
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Names of other research staff**

<table>
<thead>
<tr>
<th>NAME</th>
<th>PERCENT_SUPPORTED</th>
<th>FTE Equivalent:</th>
</tr>
</thead>
<tbody>
<tr>
<td>Larry Godinez</td>
<td>0.20</td>
<td>0.20</td>
</tr>
<tr>
<td>Total Number:</td>
<td></td>
<td>1</td>
</tr>
</tbody>
</table>

**Sub Contractors (DD882)**

**Inventions (DD882)**

**Scientific Progress**

See Attachment.

**Technology Transfer**

The workshop results were distributed to many DoD contractors and government agency representatives (see attendee list in the attached final report).
Final Report

Design for Security Workshop
September 30, 2014

Award Number: W911NF-13-1-0261

Submitted to:
Cliff Wang
Division Chief, Computing Sciences
Program Manager, Information Assurance
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211
Phone: (919) 549 – 4207
Fax: (919) 549 – 4248
Email: cliff.x.wang.civ@mail.mil

Submitted by:
University of Southern California – Information Sciences Institute
4676 Admiralty Way Suite 1001
Marina del Rey, CA 90292

<table>
<thead>
<tr>
<th>Technical Points of Contact</th>
<th>Administrative Point of Contact</th>
</tr>
</thead>
<tbody>
<tr>
<td>Jeff Draper, <a href="mailto:draper@isi.edu">draper@isi.edu</a></td>
<td>Ms. Brigidann Cooper, <a href="mailto:brigidann@usc.edu">brigidann@usc.edu</a></td>
</tr>
<tr>
<td>Tel: (310) 448-8750, FAX: (310) 823-6714</td>
<td>Tel: (310) 448-9161</td>
</tr>
</tbody>
</table>
Summary of Activities
Through sponsorship of the US Army Research Office, the University of Southern California Information Sciences Institute hosted a 1-day working meeting on the topic of Design for Security, July 23, 2014 at the Marina del Rey, CA location. While the primary focus was on electronics (ASICs, FPGAs, COTS, etc), some discussion of techniques at other system levels was also mentioned, especially in the invited talks. In the past decade, more and more fabrication of advanced ICs has migrated offshore, largely because of global economic pressures. Fabrication facilities dedicated to supporting the Department of Defense can no longer provide the performance, variety, and volume of ICs at the cost needed. Such trends have raised concerns regarding the reliance of U.S. defense systems on high-performance ICs and the potential vulnerabilities of these systems if fabricated and/or developed offshore. While previous programs, such as DARPA’s Trust in Integrated Circuits and Integrity and Reliability in Integrated Circuits, sought to address these concerns through hardware and design validation, the design perspective to explore what can be done during the design phase to increase the security of a system has not received equal attention. This workshop addressed/discussed how to incorporate security as a first-rate metric during the design flow, much like performance, area, and power. Some of the topics discussed include:

• Vulnerabilities in the current design flow of integrated circuits and embedded systems
• Potential holistic solutions to building in security during design
• Metrics for measuring security
• Defining the trade-off space between security and other design constraints such as cost, power, and reliability
• Defining the levels in the design flow where it makes sense to model threats and define appropriate defenses in response
• Security as it relates to 3rd-party IP and FPGAs
• Implications on test procedures

A wiki for distribution of workshop presentations, findings, and even videos was set up at https://uscisi.atlassian.net/wiki/display/DFSWM. Attendees and other approved users were given accounts for access to this material, and much of the material in this final report is taken directly from the postings.

Detailed Activities
The agenda for the workshop can be found in Appendix A. The day was organized into a number of invited talks, a panel session for questions and answers with the invited speakers as
well as to identify the main topics to be addressed during breakout sessions, and then two
breakout sessions. The invited speakers and talk titles are given below:

- Security in Mobile Systems - Rob Aitken, ARM
- Zynq Security Components and Capabilities - Steve Trimberger, Xilinx
- EDA Perspective on Tools for Hardware Trojan Detection and Supply Chain Security -
  Serge Leef, Mentor
- STARSS: Fundamental Design for Security Research Jointly Funded by Industry and
  Government - Celia Merzbacher, SRC

These presentations can be found in Appendix B. Following the invited presentations and panel
session, the attendees self-organized into roughly equally-sized groups between two breakout
sessions: one to address theory/metrics, and the other to address issues envisioned for reduction
to practice. Some of the issues related to these themes and presented to attendees were:

- Theory/Metrics:
  - Potential holistic solutions to building in security during design
  - Metrics for measuring security given known vulnerabilities in current design
    flow
    - How can metrics be defined so that security can be incorporated in design
      flow as constraint analogous to speed, area, power
  - Practice:
    - Defining the trade-off space between security and other design constraints such
      as cost, power, and reliability
    - Defining the levels in the design flow where it makes sense to model threats and
      define appropriate defenses in response
    - Security as it relates to 3rd-party IP and FPGAs
    - Implications on test procedures

Attendees were then given the following charge for their respective breakout sessions:

- Establish a research agenda that will solve the problem
  - Identify key aspects of the problem and a research plan for solving the problem
    - Identify key aspects of the problem that need investment
  - Identify key questions to be answered and a process for answering
  - Identify five central challenges that are worthy of pursuing and need investment

The findings of the breakout sessions are best summarized by the top 5 research area priorities
identified as needing investment:

- Methods to create verifiably secure, attack-resistant IP at all levels of design hierarchy,
  including definitions of metrics
- Methodologies/techniques for the behavioral modeling of the security of devices and
  systems
- Tools for secure interplay between hardware and software
- Design environment for modeling and simulating hardware attacks and actions for
  mitigation
- Extensions to HW description languages that capture security attributes
An outbrief presentation summarizing the motivation, issues, and findings of the workshop can be found in Appendix C. The list of workshop attendees along with their affiliations is given in Appendix D. The workshop attendance ended up being 34 with a mix of commercial industry, defense industry, academia, and government agency participants.
Appendix A – Design for Security Workshop Agenda
# Design for Security Working Meeting Agenda

University of Southern California  
Information Sciences Institute  
Marina del Rey, CA  
July 23, 2014

<table>
<thead>
<tr>
<th>Time</th>
<th>Topic</th>
<th>Presenter(s)</th>
<th>Room(s)</th>
</tr>
</thead>
<tbody>
<tr>
<td>08:30</td>
<td>Sign in / Continental breakfast</td>
<td></td>
<td>1137</td>
</tr>
<tr>
<td>08:45</td>
<td>Welcome/Logistics/Intro/Expectations</td>
<td>Jeff Draper, USC ISI</td>
<td>1135</td>
</tr>
<tr>
<td>09:00</td>
<td>Security in Mobile Systems</td>
<td>Rob Aitken, ARM</td>
<td>1135</td>
</tr>
<tr>
<td>09:20</td>
<td>Zynq Security Components and Capabilities</td>
<td>Steve Trimberger, Xilinx</td>
<td>1135</td>
</tr>
<tr>
<td>09:40</td>
<td>EDA Perspective on Tools for Hardware Trojan Detection and Supply Chain Security</td>
<td>Serge Leef, Mentor</td>
<td>1135</td>
</tr>
<tr>
<td>10:00</td>
<td>STARSS: Fundamental Design for Security Research Jointly Funded by Industry and Government</td>
<td>Celia Merzbacher, SRC</td>
<td>1135</td>
</tr>
<tr>
<td>10:20</td>
<td>Break</td>
<td></td>
<td></td>
</tr>
<tr>
<td>10:30</td>
<td>Panel - Q&amp;A from invited talks / Q&amp;A for setting up breakout sessions</td>
<td>Aitken, Trimberger, Leef, Merzbacher, Wang, Fazzari, Draper</td>
<td>1135</td>
</tr>
<tr>
<td>11:30</td>
<td>Lunch</td>
<td></td>
<td>1135</td>
</tr>
<tr>
<td>12:20</td>
<td>Report to breakout sessions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:30</td>
<td>Breakout Session 1 – Metrics Room 1135</td>
<td>Breakout Session 2 – Practice Room 689</td>
<td>1135, 689</td>
</tr>
<tr>
<td></td>
<td>Metrics for measuring security given known vulnerabilities in current design flow; how can security be incorporated in design flow as constraint analogous to speed, area, power</td>
<td>Security as it relates to IP/FPGA; impact on design flow including test procedures</td>
<td></td>
</tr>
<tr>
<td>14:30</td>
<td>Initial report out</td>
<td>Breakout Session Leaders</td>
<td>1135</td>
</tr>
<tr>
<td>15:00</td>
<td>Break / report back to breakout sessions</td>
<td></td>
<td></td>
</tr>
<tr>
<td>15:15</td>
<td>Breakout Session 1 – Metrics Room 1135</td>
<td>Breakout Session 2 – Practice Room 689</td>
<td>1135, 689</td>
</tr>
<tr>
<td></td>
<td>Follow-up session to address feedback from initial report out and finalize report</td>
<td>Follow-up session to address feedback from initial report out and finalize report</td>
<td></td>
</tr>
<tr>
<td>16:15</td>
<td>Final report out</td>
<td>Breakout Session Leaders</td>
<td>1135</td>
</tr>
<tr>
<td>17:15</td>
<td>Concluding remarks / Plan for report</td>
<td>Draper, Wang, Fazzari</td>
<td>1135</td>
</tr>
<tr>
<td>17:45</td>
<td>Adjourn</td>
<td></td>
<td></td>
</tr>
<tr>
<td>18:30</td>
<td>Dinner (stay tuned for more details)</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Appendix B – Invited Presentations
Mobile Security Systems

Rob Aitken
ARM Fellow
July 23 2014
About ARM…

- 50 Billion ARM chips
  (>10B /year)
- ~ $1.2B Revenue/year
- ~3000 Employees
- >95% Smartphone & Tablet market share
The Mobile Threat Environment

- Increasing risks
  - Social engineering – Trojans, phishing, APT
  - Malware
  - Physical loss or theft leading to risk to data – calendar, phonebook and email
  - Improperly secured devices – no PIN lock
  - User intervention – jailbreaking, unlocking
  - Mobile has become the enterprise security boundary

- Need to design in the right system-wide security (not just more security)
Whose Data Is Involved?

- User
  - Personal information, contacts, location, photos, etc.
- Carrier
  - Network interface
- Enterprise(s)?
  - BYOD
- Apps
  - Content providers
    - DRM for movies, songs, etc.
  - Finance companies
    - Account data, passwords
  - IOT
    - home automation, health, etc.
Security Profiles

- **Invasive HW Attacks**
  - Well resourced and funded
  - Unlimited time, money & equipment.

- **Non-invasive HW Attacks**
  - Side channels (DEMA, DPA)
  - Physical access to device – JTAG, Bus Probing, IO Pins, etc.

- **Software Attacks**
  - Malware & Viruses
  - Social engineering

- **TrustZone based TEE**

- **SmartCards / HSMs**

The diagram illustrates the cost/effort to attack and to secure various systems, with a focus on different types of attacks and the value to the attacker.
Mobile Solution Is Not PC Solution

- PC era security
  - Add layers of software security (SSO etc.)
  - Add hardware security (CVC, key fobs, etc.)
- Too unwieldy and confusing for mobile environment
Mobile Security Approach

- Hypervisor (with hardware support) separating large pieces of code
- Small, certifiable Trusted Execution Environment inside Application processor isolated using ARM TrustZone technology protecting against software attacks
- Secure Element for tamper proof security
Trusted Execution Environment

- Hardware root of trust
  - A basis for system integrity
- Integrity through Trusted Boot
- Secure peripheral access
  - Screen, keypad, fingerprint sensor etc.
- Secure application execution
- Trust established outwards
  - With normal world apps
  - With internet/cloud apps
Castle Analogy

- Layers of defense
- Reducing attack surface
- Increasing isolation
- Principle of least privilege
- Most precious assets protected by multiple layers of security
Castle Analogy

But…

- Modern OS/Framework is ~10GB + GBs of Apps
- So maybe we should think of a walled city & castle
- Attacks happen
- Everyone knows what the assets are and which room they are in
- Where to put high value assets such as keys?

Implementation details matter!
Castle Analogy with TrustZone Based TEE

- TrustZone based TEE creates a second (much smaller security boundary) castle with only one door, carefully designed entry/exit & APIs
- Keys only used in Secure World, Protected Crypto, Encrypted storage, Secure execution, Secure peripherals
- Offers:
  - Integrity (part of Trusted boot)
  - Confidentiality
- TrustZone TEE Castle is invisible to normal world
Castle Analogy with TrustZone Based TEE

Secure World

Isolated Trusted Apps

Trusted OS
e.g. Trustonic t-base300

Normal World

GlobalPlatform Client API
SMC calls
e.g. ARM Trusted Firmware

10-20 GB

1-2MB

Design For Security Workshop, July 23 2014
TrustZone: Two CPUs virtualized in one

- In pre-TrustZone Systems:
  - Rigid allocation of MHz/resources independent of the application.
  - Silicon costs with redundant hardware that is idle most of the time.
  - Complex control logic and deficient performance and power consumption.

![Diagram showing TrustZone architecture with two CPUs: one for normal OS code and one for trusted code only.](image-url)
TrustZone Basics

Key advantages over separate secure processor solutions:

- CPU MHz/resources are dynamically shared according to demands
- The two isolated domains are implemented in the same machine with no duplication of HW
  - Difficult to give precise “overhead” values since secure and non-secure tightly integrated from design standpoint
- Simpler and more flexible platform designs, lower costs and high power/performance efficiency
Attacking the TEE – Man In The Middle

Can then access memory used to communicate between Client App and Trusted App.

Malicious App can intercept traffic, replace it, modify it or eavesdrop.

Malicious App attacks OS/kernel.

Secure call to TEE.

Trusted Execution Environment

- Trusted App
- Trusted App

TEE Kernel

Rich OS

- Client App
- TEE Lib
- Malicious App

OS Kernel

TEE Monitor

ARM
Side-Channel Attacks

Rich OS

OS Kernel

Client App

Malicious App

TEE Lib

Secure call to TEE

Trusted Execution Environment

Trusted App

Trusted App

TEE Kernel

TrustZone® Monitor

TrustZone®

Trusted App

Trusted App

TEE Kernel

TrustedExecutionEnvironment

TrustedApp

TrustedApp

TEE Kernel

TrustZone® Monitor

TrustZone®
Defenses

- Normal World to Secure World communications are always exposed and vulnerable

Mitigation

- Don’t design systems that rely on secure communications between Normal World and Secure World
- Always use trustworthy components – crypto library, TEE and protocols

Rich OS

TrustZone®

Trusted Environment

Client App

Malicious App

OS Kernel

TEE Lib

Trusted App

secure call to TEE

Malicious App can intercept traffic, replace it, modify it or eavesdrop

TEE Kernel

Trusted App

ARM®
Propagating System Security

NS : NOT Secure, treated like an address line
## TrustZone Controllers – Vital Statistics

<table>
<thead>
<tr>
<th>Code</th>
<th>Product</th>
<th>Main Function</th>
<th>Key Features</th>
<th>Size</th>
</tr>
</thead>
<tbody>
<tr>
<td>TZC-380</td>
<td>TrustZone Address Space Controller</td>
<td>Partition external DRAM into secure and non-secure regions</td>
<td>Configurable up to 16 regions of size 32K to 4G, each with 8 sub-regions (down to 4K). Configurable registering to meet timing constraints with minimum latency. AXI interface for compatibility with NIC-301 and DMC-34x.</td>
<td>10-100k gates</td>
</tr>
<tr>
<td>BP141</td>
<td>TrustZone Internal Memory Wrapper</td>
<td>Protects internal SRAM</td>
<td>Manages a single secure region within the SRAM. AXI interface.</td>
<td>&lt;1k gates</td>
</tr>
<tr>
<td>BP147</td>
<td>TrustZone Protection Controller</td>
<td>Prevents non-secure accesses to peripherals</td>
<td>Allows peripherals to be safely shared by the Secure and Non-Secure worlds. APB interface.</td>
<td>&lt;1k gates</td>
</tr>
</tbody>
</table>
Application of Hypervisor for BYOD

Two Personas

- Mutual Distrust model between OSs
- Ensuring Enterprise OS Security, while protecting Consumer OS Privacy.
- Enabling Enterprises to have control of their own assets in case of loss.
Secure Content Path: SoC Requirements

Firmware protected against tampering
- Any SW component directly used in setting up protected memory path
- Decoders, mixers, renderers, DRM
- Critical components placed in secure processing space
- Integrity checked at boot time

Unencrypted content protected
- After DRM protection removed
- Unencrypted content never accessible to processes running in HLOS
- Unencrypted content only ever written to protected memory

Memory buffers protected by HW control
- All memory used in processing, decoding, mixing and rendering
- Sufficient memory for video bitstream and frame buffer
- Not accessible by HLOS or unauthorised HW or SW
- Output only to internal display or via protected export clients such as HDCP and DTCP
Secure Implementation Example

**Low cost and complexity**

- Secure CPU, bus fabric and Video from a single source
- System IP designed to work together
- Simple SW integration – create a secure session then manage scheduling/control as normal

**Minimal memory fragmentation**

- Major issue for HD content
- Video MMU can be used for secure sessions by TEE
- No need to assign large, contiguous secure buffers

**Increased flexibility and protection**

- Simultaneous protected and unprotected video streams
- Additional protection of video firmware (read-only) and data (non-executable)
Zynq Security Components and Capabilities

Steve Trimberger, Xilinx
Agenda

- Security Features
  Inherited from FPGAs
- Zynq Secure Boot
- TrustZone Integration
Zynq All-Programmable SoC

**Processor System (PS)**
- 2x ARM9 866MHz-1GHz 32K/32K I/D Caches
- 512KB shared L2 Cache
- 256KB On-chip memory
- Memory controller
- Bus interfaces, timers
- Libraries, OSs, middleware

**Programmable Logic (PL)**
- 28K – 440K LCs
- 240K – 3MB RAM
- 80 – 2020 DSP blocks
- I/O, Transceivers, PCIe, Ethernet...

**Programmable ADC**
- Inputs from Voltage, Temp sensors

**AMBA AXI bus fabric**

© Copyright 2014 Xilinx
Agenda

- Security Features
  Inherited from FPGAs
- Zynq Secure Boot
- TrustZone Integration
Passive Security Features: Device Identification

Device DNA and User eFUSE field
- Uniquely identify the chip
- An application can be tied to exactly that chip and no other

User eFUSE bits disable unencrypted bitstreams
- FPGA rejects unencrypted bitstreams
- Restrict system usage to authorized applications only
Active Security Features: Monitors

- DETECT if there is activity on JTAG chain and DISABLE the JTAG chain – JTAG is arguably the #1 vulnerability in every integrated circuit.

- ADC can monitor user-specified temperature and voltage limit

- SEU checker: detects and repairs configuration bit flips. Detects attempts to subvert operation with focused radiation
Active Security Features: Actors

- Clear the **design**, **data** and **key** from inside the FPGA

- GTS macros immediately **tri-state** pins

- PROG_B intercept: user application prevents reconfiguration until cleanup done
Agenda

Security Features
Inherited from FPGAs
Zynq Secure Boot
TrustZone Integration
Dual Authentication in Zynq Devices

Symmetric (AES-HMAC)
- High-speed for fast configuration
- Inherited from FPGA

Asymmetric (RSA)
- Provides authentication without using secret data
- Key in silicon is “public” - does not have to be secret
Zynq Key Loading

READY TO BOOT!

Secret “Red” AES Key
BBRAM or eFuse

Programmable Logic

© Copyright 2014 Xilinx
Zynq Secure Boot

- Trust starts with boot ROM
- In secure boot, FSBL is authenticated before execution
  - RSA-2048, user chooses the key
- FSBL is just (authenticated) code. It can do anything securely
  - Partition into pieces to fit into OCM
  - RSA authentication for each partition
  - AES-HMAC for each partition
  - New authentication or decryption algorithms
  - Key rolling
- Single-entity model
  - All secure boot starts with PS boot
  - Secured PS boot manages PL boot
Agenda

- Security Features Inherited from FPGAs
- Zynq Secure Boot
- TrustZone Integration
ARM TrustZone: separates Secure World processes and components from Normal World

Secure World may access all components. Normal World may not access secure world components.

Trust tags added to AXI bus transactions: AWPROT, ARPROT

Mapping of components to Secure World is done in the PS system build
**TrustZone in Programmable Logic**

- **AXI Switches handle TrustZone protection bits**
- **TrustZone pushed to AXI bus endpoints in PL**
  - These are firmware, built from FPGA fabric
  - They operate just like the corresponding hard logic in PS
  - They are marked by the user at compile time as Secure World or Normal World
  - Check ARPROT, AWPROT during operation
What Does This Mean?

› Chain of Trust lets you build what you want in code and fabric
› Control: JTAG, configuration
› Monitoring
› Defensive Actions
› Algorithm choice
› DPA resistance
› HW/SW, it’s all good

EDA Perspective on Hardware Cybersecurity

Serge Leef

Vice President and General Manager

- New Ventures
- System Level Engineering Division
Cybersecurity Is A ‘Big’ Topic

Source: Search conducted in Factiva. Duplicates removed

© 2012 Mentor Graphics Corp. Company Confidential
www.mentor.com
Need to Fill Up Your Calendar?

July 2014

August 2014

September 2014

October 2014

November 2013

December 2014

- Internet of Things
- Cloud Computing
- Cybersecurity


© 2012 Mentor Graphics Corp. Company Confidential
www.mentor.com
Internet of Things Dramatically Expands the Threats to Cyber Security

**ATTACK TYPES**

- **Social Engineering**
  - (Phishing/bating)
  - Impact: 1 - 100

- **Malware / Macros**
  - (Information harvesting)
  - Impact: 10,000 – 100,000

- **Viruses/ Trojans**
  - (Hijacking, DDoS, etc...)
  - Impact: ~100 Million

- **IoT**
  - Impact: ~100 Billion

Embedded Threats Moving Down the Stack
Stealth & System Control Increase

Traditional target for disabling security tools

OS can harbor ‘advanced persistent threats’ for a specific target

Embedded Firmware malware is beyond the reach of current tools

Ultimate threat resides in the hardware blocks
Are EDA companies Ultimately Responsible for Solving the Security Problem?

- Traditional verification role
  - Verifying a chip does what it is supposed to do

- Emerging new role
  - Verifying a chip does nothing it is NOT supposed to do
EDA as the first line of defense

What to attack first?
1. Problems that have measurable impact
2. Appear to be solvable
3. Customers are willing pay for solutions

Side-channel Attacks - solutions exist
— Attempt to leak out keys via differential power analysis (and the like)
— Current targets are mainly in smart-card and set-top box areas

Counterfeiting - problem apparent, no solutions yet
— Over-produced, cloned re-marked, recycled or otherwise unauthorized ICs provided by uninformed or untrustworthy suppliers and distributors for economic or adversarial reasons

Hardware Trojans - a theoretical threat?
— Malicious circuits put inside a chip which are harmless in normal operation until triggered by a preset internal or external condition(s)
CHARACTERIZING THREAT VECTORS
But it’s not just design, the whole supply chain has evolved:
Evolution of IC supply chain (past)
Evolution of IC supply chain (present)
Attacking IC design flow

- Lots of Third party IP and Code Reuse
- Complicated Scripts, Third Party tools
- Physical Placement of Gates, Heavy use of tcl scripts

Diagram:
- Requirements
- Design Specification
- RTL Coding
- Functional Verification
- Logic Synthesis
- Gate Level Simulation
- Place and Route
- Layout Verification
Manufacturing stages → many attack vectors

~200 processing steps in IC fabrication

- Lithography processes present opportunities to print additional circuitry and devices.
- Need to replace glass masks.
- Masks are automatically loaded into litho tools.
- No physical access to target tool required.
- Other process and measurement/metrology steps present opportunities for causing scraps.
- Trojan circuitry may be inserted in different layers of circuitry within the chip.
- Long manufacturing lines ~200 processing steps.
- Many opportunities for malicious insiders.
- Targeting processes at the BEOL (Back End Of the Line) causes higher damages to the IC manufacturer.
COUNTERFEITING

Over-produced, cloned re-marked, recycled or otherwise unauthorized ICs provided by uninformed or untrustworthy suppliers and distributors for economic or adversarial reasons.
Electronics Supply Chain is Global

Global nature of supply chain makes chain-of-custody approach unworkable

Lifecycle for a single JSF (Joint Strike Fighter) IC – Component changed hands 15 times before final install

Source: IDC Manufacturing Insights & Booz Allen analysis
How can we build trusted silicon in an untrusted environment? VPN for ICs?

Secure Tunnel (VPN)

Data

IC Design

Supply Chain Protection Solution

Untrusted

network

Untrusted

supply chain

Users 😊

“VPN” for Trusted Silicon

Users 😊
Authentication?
Authentication alone is not enough: Additional mechanisms to be considered for higher security levels

- A more comprehensive EDA tool is needed

- On-chip **odometers** can address recycling threat
  - On chip structures that count some physical events like power cycles, memory accesses or other inexpensively measurable values
  - Data in the odometer is encrypted; reset to ‘0’ indicates an attack
  - Can be accessed at the authentication time

- **Activation** – chips do not work as manufactured
  - Only the IP rights holder would have the keys needed to activate chips
  - Different degrees of activation need to be offered to enable the customer to make trade-offs between security and costs
  - Pre-existing test methods should be accommodated
  - Power-up, event-based or periodic activation offers highest security
Chip Activation by Logic Encryption

- Inject “security gates” into the design
- Security gates are no-ops if the key value is correct
- Security gates break functionality if the key value is wrong
- At least one gate per key bit
  - More gates (up to a point) can be used to “couple” key bits and increase variance of the outputs
- Key strength increases with key register size
Managing Keys Securely

- Design house may never see the actual keys if they are managed by PaaS (Platform as a Service) technology.

- A secure key management platform could be operated by:
  - Government agency (ex: DoD)
  - Commercial entity (ex: Amazon)
  - The Design House’s own cloud.

- PaaS would expose a Web Services API that all EDA tools in the chain would access to deposit or access key via encrypted communications.
Different types of solution are needed

- **Offline authentication**
  - Valid chip contain a unique key that can be verified offline
  - Unauthorized chips work but can be identified (if in possession)

- **One-time activation using global key**
  - As-manufactured chips do not work
  - Key needs to be entered to activate the chip

- **One-time activation using unique key**
  - Same as above but key is unique for each chip

- **Power-up activation using unique key**
  - Chip is not activated permanently, each use must be activated
Solution Space for IP Protection

- Chip activated on every use
- Unique key for each chip
- Chip disabled until activated
- Offline authentication
- Key insertion
- Per-part key management
- Per-chip key management
- Activation infrastructure
- Testing impact

Cost vs. Security
Markets for Supply Chain Security Solutions

IC price

Power-up unique key

One-time unique key

One-time global key

Offline authentication

Ignore the problem

security

volume
Needed: scalable platform that can support multiple contributions from many parties

- Need to raise the bar to deter financial incentive, can’t solve Nation-State Attack
- New, digital design is target (not discrete or existing design)
- Following traditional EDA methods, crypto, security gates, registers insertion, access can be automated, verification performed
- User assisted selection of crypto, activation block, # of registers
- EDA contribution: Standard insertion methods and interface
TROJANS

Malicious circuits put inside a chip which are harmless in normal operation until triggered by a preset internal or external condition(s)
Threat Example: Compromised Router

- Unpublished control message travels around the internet and is unrecognized and ignore by most routers.
- When a router containing a hardware Trojan in the control plane sees such message, it takes action to re-direct data.
Why is Trojan Detection Difficult?

- **Low probability of triggering during test**
  - Even a small IC today has millions of nodes
  - There are billions of states
  - Tests are for known use cases
  - Test time is expensive
  - Consider testing a million chips per production batch
  - Very difficult to test for **Unknown Unknowns**

- **Large number of gates in modern chips**
  - Exhaustive simulations are extremely computation and memory intensive
  - Obfuscation occurs during synthesis
  - No signature in Trojan circuits - they look just like normal hardware
  - Low probability triggers are finite state machines that can change states when time or input data changes
  - Nano-scale devices and high system complexity make detection through physical inspection almost impossible
IP as a Trojan carrier

- In a typical IP-based design, each block can originate from different sources.
- Incoming IP blocks are verified to confirm promised functionality.
- Additional verification may be done to confirm proper interaction with other IP blocks operating in the system context.

- A key question that does NOT get asked in this process is: “does this block do anything ELSE?”
- Possible countermeasures:
  - Scan incoming IP for Trojan signatures - HARD
  - Insert run-time detection mechanisms.
Possible Trojan Solution

- Design time detection - not viable
  - Expanded test benches
  - Formal methods

- Solution: Run-time Trojan detection
  - Using declarative form, describe rules governing bus-based communications
  - Synthesize bus-interface logic & co-processor
  - Generate encrypted microcode containing detection mechanisms for known attack profile as well as system architecture specific ones
  - Include co-processor in the design
STARSS:
Fundamental Design-for-Security Research

Dr. Celia Merzbacher
VP for Innovative Partnerships & Government Relations
Director, Trustworthy and Secure Semiconductors and Systems
I just want to say one word to you. Just one word. Are you listening? Cybersecurity.
Semiconductor Industry Trends & Challenges

- More pervasive, embedded, and networked, including in critical infrastructure systems
- More complex (SoC, NoC, SoS)
- More 3rd party IP
- Supply chain more global

★ **More vulnerabilities**
★ **Greater impact if chip fails**
★ **More attractive to adversaries**
Semiconductor Research Corporation:
A Family of Distinct, Related Program Entities

Trustworthy and Secure Semiconductors and Systems (T3S): A New Thrust in the SRC Portfolio

Global Research Collaboration
Ensuring vitality of current industry

Targeted Research

ESH

T3S

SemiSynBio

Advanced Connectivity

STARnet

Focus Center Research Program Phase VI
- STARnet
- Early research engagement of key long horizon semiconductor challenges

NRI

Nanoelectronics Research Initiative
- Beyond CMOS – the next switch and associated architectures

Education Alliance
Attracting and educating the next generation of innovators and technology leaders

Bringing industry together to identify and support – in collaboration with government – fundamental research for hardware assurance.
Essential Features of SRC Programs

✓ Highly leveraged investment in research
✓ Needs-driven, consensus-based goals
✓ World-class researchers (faculty & students)
✓ Interaction among academic and industry experts
✓ All members have rights to resulting IP
✓ Facilitated tech transfer via liaisons, online tools for access to project information, student resumes, etc.; webinars and in person reviews
✓ Nimble and adaptable (does not fund “bricks & mortar”)
✓ Accountable; value-driven; efficient; effective
**Goal:** Provide maximum assurance that IP/chips/ systems will perform only as intended without impacting time to market, cost & performance and are resistant to attack/theft.

**Objectives:**

- Develop cost-effective strategies, techniques and tools to increase security, trust, and assurance in chip-based components and systems.
- Form public-private partnerships that leverage investment.
- Grow/tap into the university research enterprise.

**Initial participants**

- Intel
- Freescale
- Mentor Graphics
- NSF
Step 1: Define Research Needs

SRC-NSF sponsored workshop in January 2013*, with experts from industry, academia and government, identified the following areas:

- Design and Manufacture for Security and Assurance, including properties, principles, architecture, specification, verification (internal and 3rd party IP) and validation
- Metrics for evaluating security and trustworthiness
- Vulnerability and threat assessment and frameworks
- Anti-counterfeiting strategies/techniques, e.g., authentication of semiconductor provenance tamper resistance, and securing the supply chain

Survey of Hardware Security Threats

- Via email in May 2014
- Sent to ~200 individuals in industry, academia and government; received 60 responses
- Summary available via the SRC website at file:///C:/Users/merzbacher/Downloads/starss-survey-results.pdf
Q1 What are the top three current threats.
Q2 What will be the top three threats in 10-20 years.
Other Threats

• Reverse engineering
• Hardware features that enable software and data attack
• The primary challenge in 10 – 20 years will likely be something we are not aware of today.
Current Threat Ranking by Sector

- IP/data theft
- Bad (3rd party) IP
- HW-related breach
- Fake legacy parts
- Fake current parts
- Tamper/Trojan
- Emerging tech vuln.
- Other

Legend:
- Govt
- Acad
- Ind
Future Threat Ranking by Sector

- IP/data theft
- Bad (3rd party) IP
- HW-related breach
- Fake legacy parts
- Fake current parts
- Tamper/Trojan
- Emerging tech vuln.
- Other

Legend:
- Govt
- Acad
- Ind
Q3: Rank the following threat agents in order of concern to you/your organization today.
Threat Agent Ranking by Sector

- Hacker
- Political attack
- Econ attack
- Competitor
- Insider
- Other

Legend:
- Govt
- Acad
- Ind
Q4: What are the top three research challenges that you feel can and should be addressed by university research in the next 3-5 years?
Other Research Needs

• Design for resilience against security attacks (similar to fault tolerance against operational defects)
• Role of humans in building and operating assured systems
Research Needs Ranking by Sector

- Design
- Verification
- Features (e.g., PUFs)
- Metrics
- Threat assessment
- Run-time monitor
- Roots of Trust
- Est. assurance
- Sec. composition
- IoT, distrib nets
- HW/SW co-design
- SC assur/provenance
- Other

Govt  Acad  Ind
• **Engaging/recruiting additional members & partners**
  - In discussion with other semiconductor companies; network & other system developers/integrators; and critical infrastructure companies
  - Workshop held in May 2014—31 companies participated—to discuss drivers, capabilities, gaps and research needs. [https://www.src.org/calendar/e005440/](https://www.src.org/calendar/e005440/)
  - Engaging additional government partners with interests/investments in university research and education

• **National Science Foundation (NSF) and T3S co-funding a multi-million program** on Secure, Trustworthy, Assured, and Resilient Semiconductors and Systems (STARSS).
  - First round of projects have been selected and will start in Q4 2014
NSF-T3S STARSS Solicitation Topics

- **Architecture & Design**: Architectural and design approaches, models, and frameworks for reasoning about and specifying hardware-specific security properties.
- **Properties, Principles & Metrics**: Development of a set of hardware security design principles and semiconductor-specific properties.
- **Security Verification & Analysis**: Tools, techniques, and methodologies for verifying hardware-specific security properties and enforcing the security design principles.
- **Threat Assessment**: Frameworks for analyzing and sharing information about security threats due to unintended vulnerabilities or malicious attack during design or manufacture.
- **Authentication & Attestation**: Models for the insertion of artifacts and/or design elements that are verifiable during design and manufacture.
- **Tools and Frameworks**: Develop security engineering models for implementation of research results and for use in education and training of engineers.

1st Round of STARSS Projects

- Secure chip odometers for measuring use and age
- Trojan detection and diagnosis
- PUF-based authentication
- Design of low-cost memory-based security primitives and techniques
- Design & Metrics for resistance to differential power analysis attack
- Understanding and detection of fault-based attacks
- IP integrity validation
- Hierarchical approach to design of secure IC’s using authentication and obfuscation
- Invariant carrying machine for hardware assurance
Increasingly recognized as important…

But remains a challenge.

Google Project Ara modular phone
Appendix C – Design for Security Workshop Outbrief
Design for Security Working Meeting
Final Report

Jeff Draper
Project Leader, Information Sciences Institute
Research Associate Professor, Ming Hsieh Department of EE
ISI, Marina del Rey, CA
September 29, 2014
- U. S. Army Research Office Award No. W911NF-13-1-0261
  POC: Dr. Cliff Wang

- The views, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.
— Motivation

- Protection of Intellectual Property (IP), critical components of all DoD designs
  - Additionally, DoD has a legacy IP protection issue
    » Designs spanning the last 30 years with thousands of different IP blocks
  - Intersection of security, reliability, safety
  - Mission sustaining capability
    » Infrastructure for combatting unknown unknowns

— IP Protection

- Privacy
- Copying/counterfeiting
- Sabotage

— Challenges (Areas needing investment)

- Metrics for quantifying design security
- Tools for measuring design security and developing/evaluating mitigation approaches
Motivation

• Intellectual Property (IP) intensive industries account for 20% of the US workforce and a third of GDP

• Many reported IP compromises involve chip-based platforms

For DoD, chips are core to modern weapon systems, including airplane, missile, C4ISR, etc.
Motivation

• Chip-level IP protection strategies must consider various threats
  — Privacy
    ▪ Preventing reverse engineering
  — Theft
    ▪ Copying / counterfeiting
  — Sabotage
    ▪ Preventing denial of service or more insidious attacks
  — Relationships between threats
    ▪ Threats are not necessarily mutually exclusive or hierarchical, i.e., copying can be done without any knowledge of how a design really works

• DoD IP protection involves critical factors that are not as dominant in consumer market and therefore not likely to be addressed by commercial ventures alone
  — Legacy IP protection
    ▪ Theory/tools needed for assessing vulnerabilities in legacy systems
  — Mission-sustaining capability
    ▪ Approaches must consider long deployment lifetimes
  — Information dominance
    ▪ DoD’s C4ISR must be protected and trusted
• All facets of IC industry have a stake in the game, for example
  — ARM
    ▪ Embedded core ecosystem where customer demands protection
    ▪ Trustzone approach only beginning to tackle the problem
  — Xilinx
    ▪ FPGA protection, especially configuration scan chain
    ▪ Zynq secure boot addresses only part of the problem
  — Mentor
    ▪ Trustworthy CAD tool flow for generating/verifying chip designs
• Will markets really be willing to pay the cost for added protection?
• Developing a holistic approach that enables security to be quantified so that it can be treated as a design constraint
  — If successful, should be able to easily extend current design flow using security as another constraint (similar to area, energy, speed) in multi-objective optimizations
  — Implies development of an “algebra” for quantifiable assessments

• Reducing such a paradigm to practice
  — Techniques
  — Tools
  — Indirect effect on other parts of chip design flow, like testing
    - Must incorporate intelligent targeted testing accounting for attack types; Monte Carlo-style testing alone will not suffice
• Success of an extensive design for security approach hinges on quantifiable measures of security, or metrics

• Prior work in this area has been largely theoretical without a means to reduce to practice

• Potential tiered approach may enable traction
  — Capture design statistics at various levels that could contribute to a measure of security
    • e.g., layer density at layout level, complete state machine transition specification at RTL level, complexity figures (number of gates, number of nets, fanout averages, etc)
  — Combine design statistics with attack-specific information
    • While design statistics are static with respect to a specific design, the contribution of the attack-specific information to a security measure is dynamic, changing with the added knowledge of future attack types
Mulit-Level Security Metrics

- Different levels of views of security and associated metrics (from each specific attack to general resilience issues against the unknown unknowns)
- Establish composite security metrics
- Account for:
  - Dynamic risk/reward model
  - Cost to implement
  - Cost to detect
  - Cost of attack
  - Attribution of attack (designer, foundry, etc)
  - Impact of attack
  - Resilience to attacks
    - Side channel exposure, reversability
- Approach should not be dominated by any specific problem/attack and should envision the presence of future unknown unknowns
• Metrics and tools for design security must consider nuances of targeted domains
  — Analog / mixed-signal
  — Digital
    ▪ Control blocks versus datapath structures
    ▪ Pipelined structures
    ▪ State machine structures

• Must consider each level of design flow and identify overlapping, orthogonal, or even conflicting issues between various levels of design
• Overarching theme: need for systematic approach to HW security
  — Methods to create verifiably secure, attack-resistant IP at all levels of design hierarchy, including definitions of metrics
  — Methodologies/techniques for the behavioral modeling of the security of devices and systems
  — Tools for secure interplay between hardware and software
  — Design environment for modeling and simulating hardware attacks and actions for mitigation
  — Extensions to HW description languages that capture security attributes
Appendix D – Design for Security Workshop Attendee List

1. Rob Aitken, ARM
2. John Chandy, Uconn
3. Michael Chen, Mentor
4. Aravind Dasu, USC ISI
5. Antonio de la Serna, Draper Laboratory
6. Jeff Draper, USC ISI
7. Ben Epstein, iSW
8. Josh Etzkin, Sandia National Laboratories
9. Saerio Fazzari, Booz Allen Hamilton
10. Paul Franzon, NCSU
11. Matt French, USC ISI
12. Mike Fritze, USC ISI
13. Denis Garagic, BAE Systems
14. Krishna Gopalakrishnan, BAE Systems
15. Chana M. Greene, Capt. USAF
16. Kenneth Heffner, Honeywell
17. Jonathan Heiner, USAF
18. Fouad Kiamilev, U of Delaware
19. Farinaz Koushanfar, Rice University
20. Serg Leef, Mentor
21. Daniel Marrujo, DMEA
22. Carl McCants, IARPA
23. Celia Merzbacher, SRC
24. Matt Noell, Raytheon
25. Jon Osborn, Aerospace Corporation
26. Jim Plusquellic, University of New Mexico
27. Miodrag Potkonjak, UCLA
28. Gang Qu, U of Maryland
29. Matt Sale, NSWC Crane
30. Shaker Sarwary, Atrenta
31. Peilin Song, IBM
32. Steve Trimberger, Xilinx
33. Ingrid Verbauwhede, UCLA
34. Cliff Wang, US Army Research Office