How Formal Analysis Proves the Security of Your FPGA Design

BUDAPEST 2019

Rachid LAARIS

Product Manager

Cadlog Srl

RACHID LAARIS

CADLOG

Design and Manufacturing Innovation

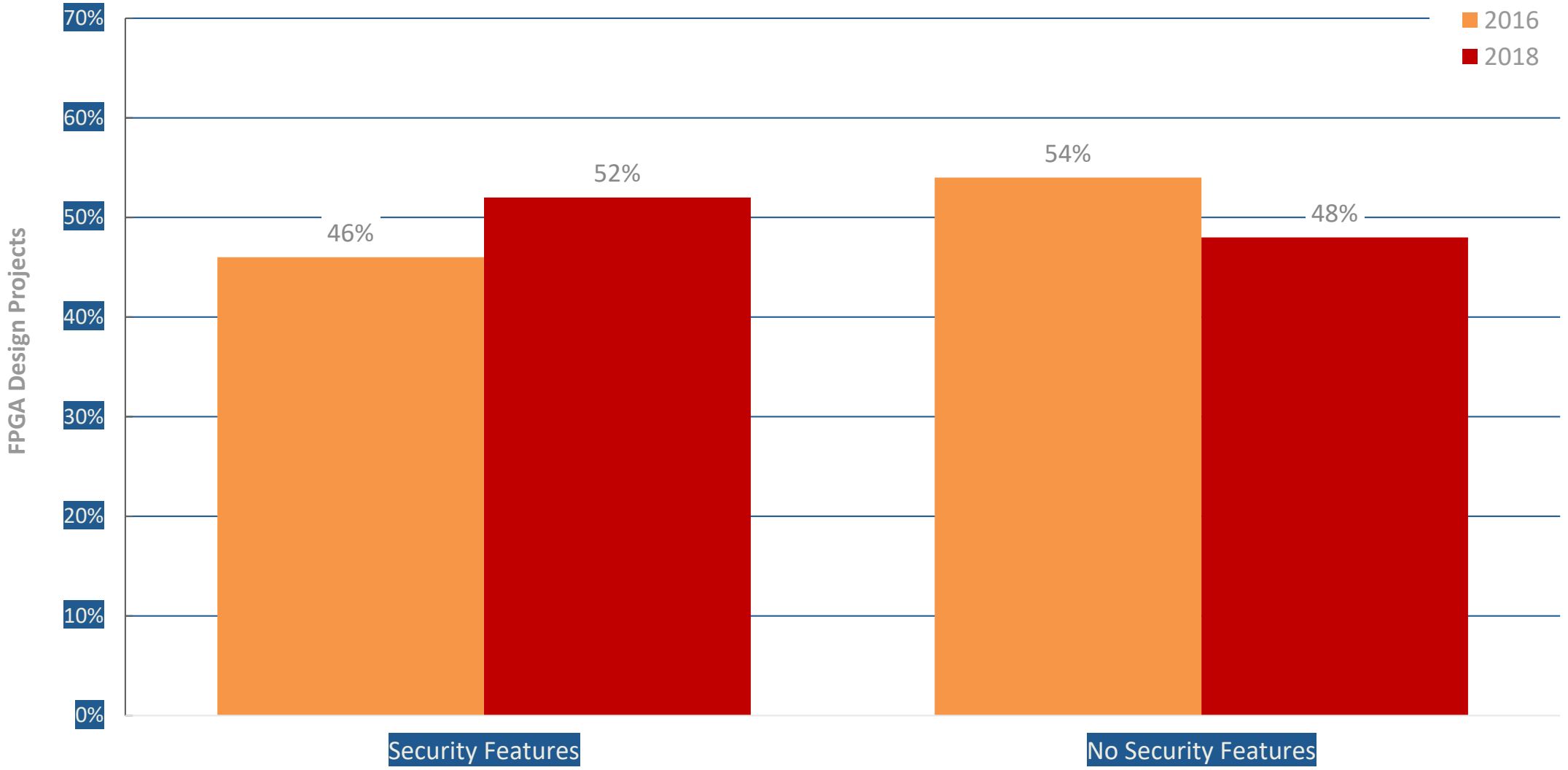
Mentor®

A Siemens Business

Presentation Overview

- The Security Challenge
- The Need for Security Verification
- Why Normal Verification is Inadequate
- Data Integrity and Data Confidentiality

Projects Working on Designs With Security Features



Source: Wilson Research Group and Mentor Graphics, 2018 Functional Verification Study


MOTHERBOARD
SINGAPORE

The Need For Secure Medical Hardware

WIRED

THREAT LEVEL

It's Insanely Easy to Hack Hospital Equipment

BY KIM ZETTER 04.25.14 | 6:30 AM | PERMALINK

When Scott Erven was given free rein to roam through all of the medical equipment used at a large chain of Midwest health care facilities, he knew he would find security problems–but he wasn't prepared for just how bad it would be.

Forbes

Hacking Insulin Pumps And Other Medical Devices From Black Hat

One of the briefings at Black Hat this year was a session on how vulnerable medical devices are to cyber attack, given by Jay Radcliff. This was part of our company's coverage of Black Hat and Defcon highlights sent via social media. Although the actual demonstration of an insulin pump being induced to given an insulin overdose as a result of switching batteries was interesting, the more disturbing take away from the briefing was the lack of controls and incentives on medical device manufacturers to ensure that their devices are secured from tampering. Some key points

- Most medical devices are surprisingly open
- Fixed equipment as well as wearables are vulnerable to unauthorized digital data tampering

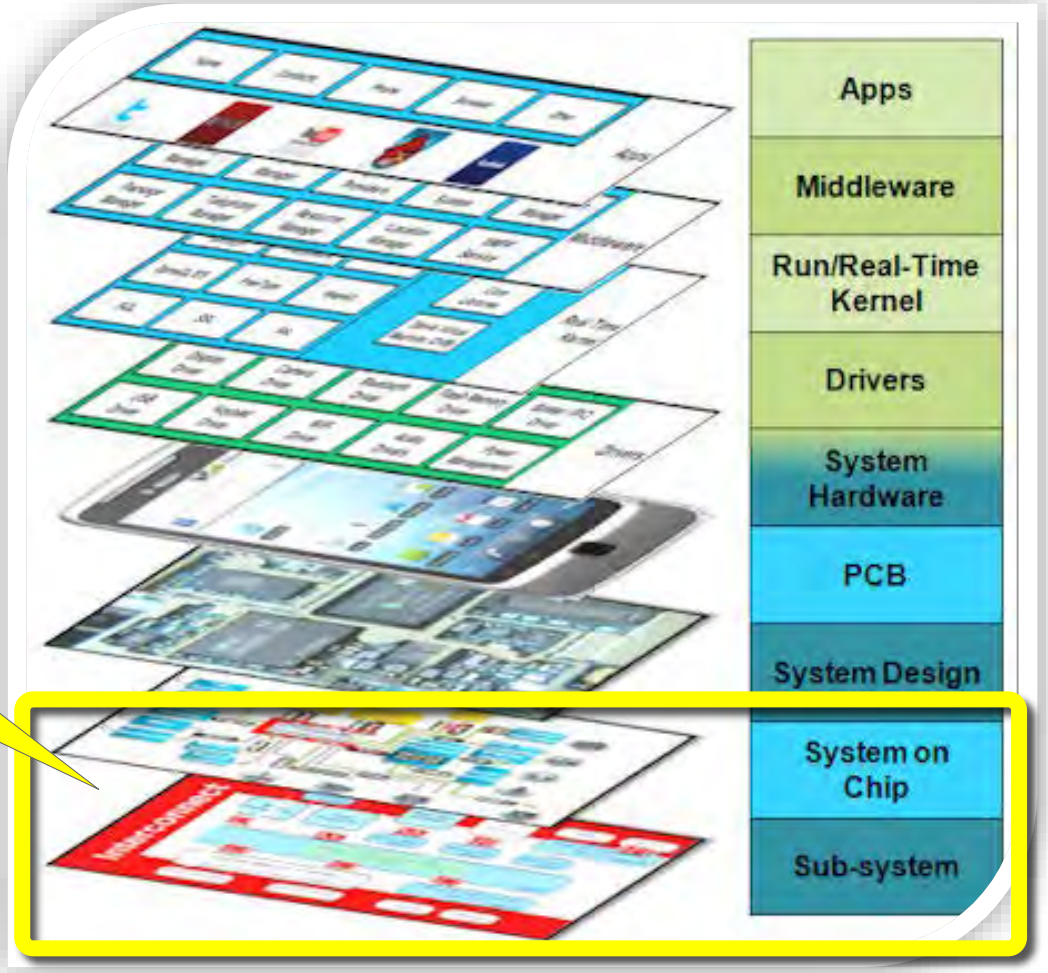
How Secure Is Your NPP?

- NPP I&C systems generally use closed data and communication networks.
- However, recent cases of APT attacks demonstrate that NPP I&C systems may also be infected by malware
- These connection points are usually related to the plant maintenance and test task

APT: Advanced Persistent Threat is a stealthy computer attack … get unauthorized access to a network and remain undetected …

Root of Trust Begins At The “Bottom of the Stack”

Today’s Focus:
Chip-level Design & Verification



The HW Security Challenge In A Nutshell

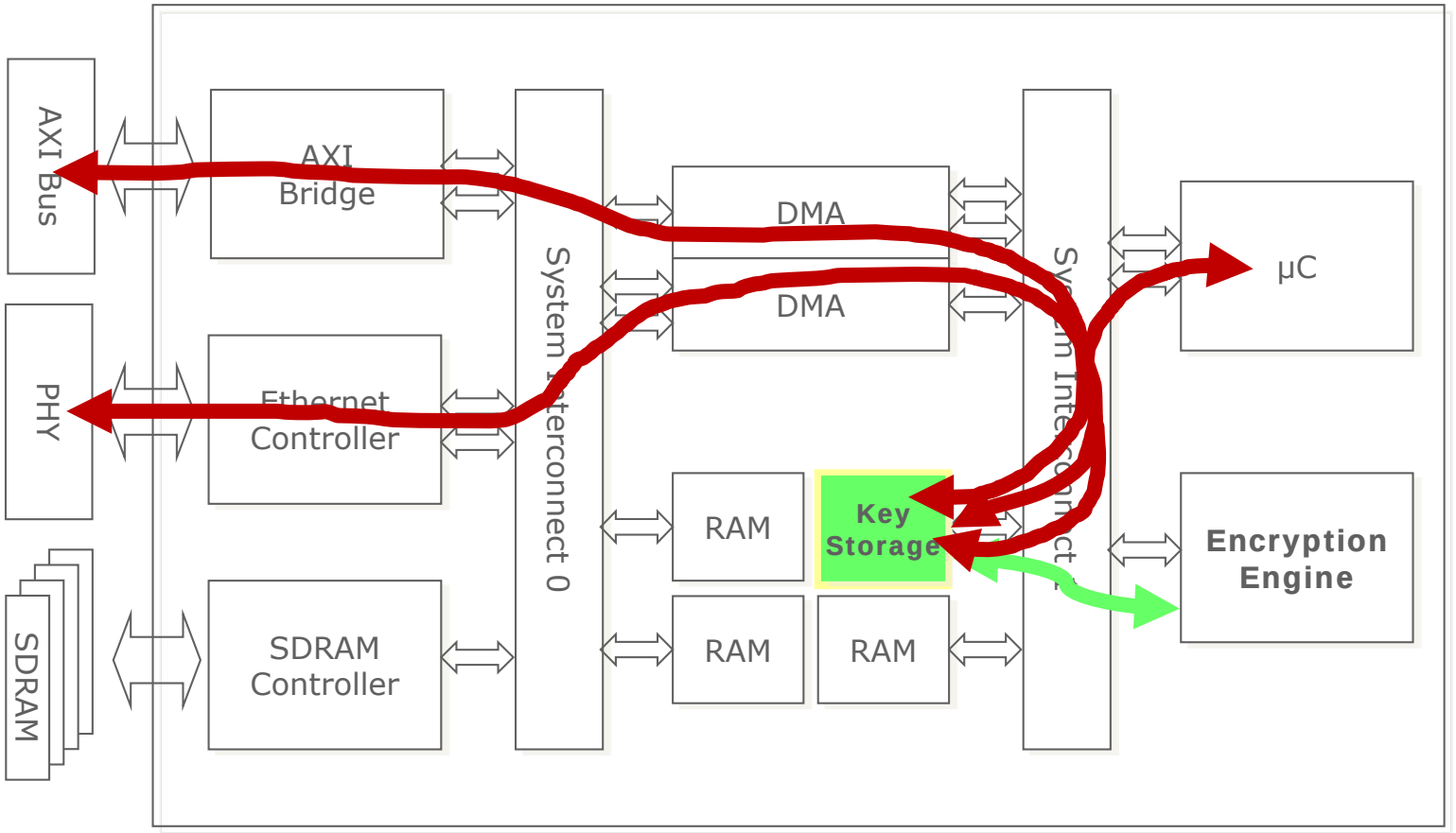
Packets in all systems – automotive, medical, NPPs, etc. – will need to be "signed" by private keys

Thus,

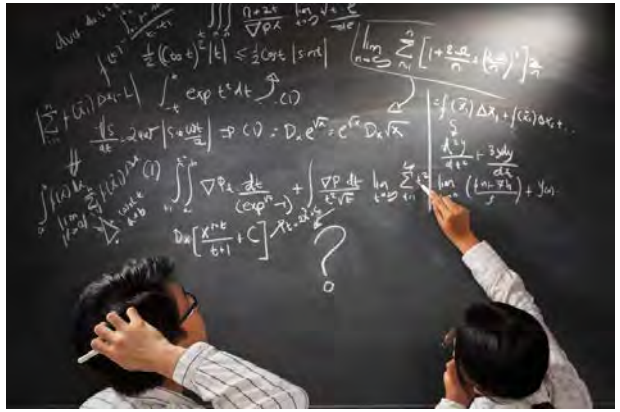
If the private key storage hardware is not secure, the whole system is vulnerable.

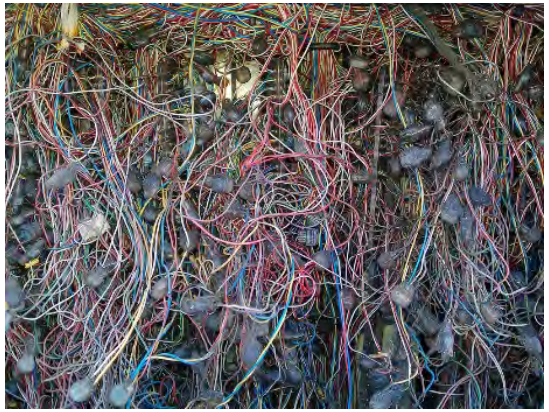
Protect your private security keys!!!

SoC controller: Is The Desired Path The ONLY Path?



Bad News: Manual & Sim. Methodologies Don’t Scale





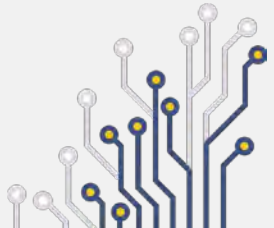
- Expert inspection of critical / open paths decreases in effectiveness as circuit complexity increases
- Even a really well designed constrained-random simulation environment is not exhaustive

Good News: Formal is Exhaustive & Scales!

CADLOG
Design and Manufacturing Innovation

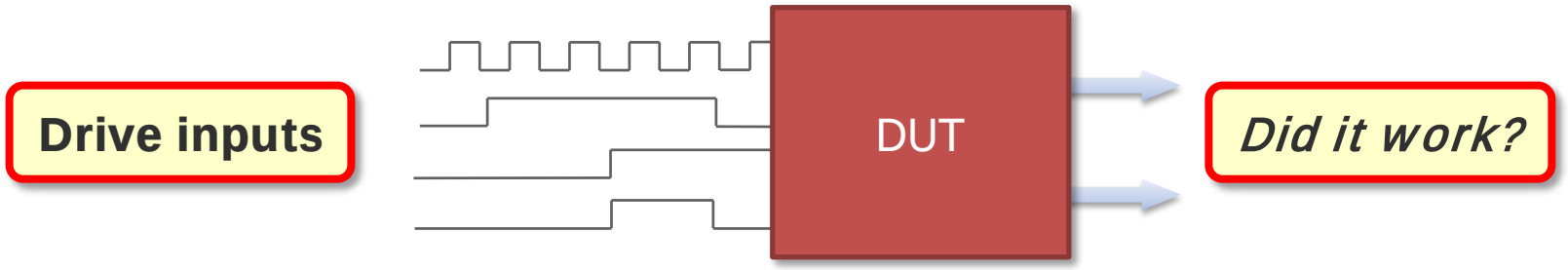
> *"Formal verification uses mathematical formal methods to prove or disprove the correctness of a system's design with respect to formal specifications expressed as properties…."*
>
> [*Using Formal Methods to Verify Complex Designs*, IBM Haifa Research Lab]

- Mathematical and algorithmic - > **exhaustive**
- Ensures implementation meets requirements
- Requires no testbench or stimulus

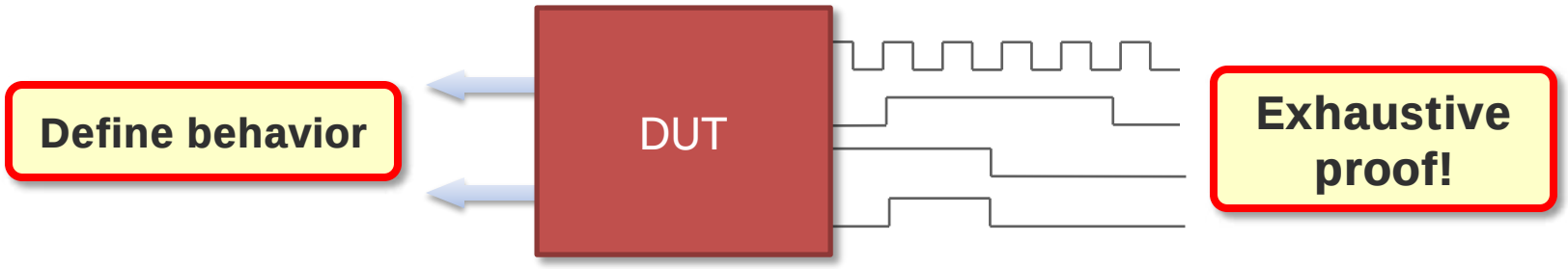


Simulation vs. Formal

Simulation – tests the design

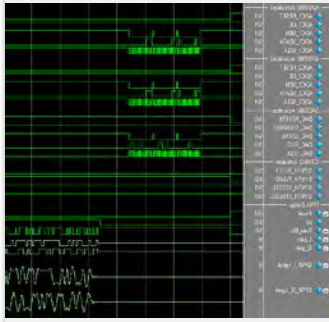


Formal – proves the design

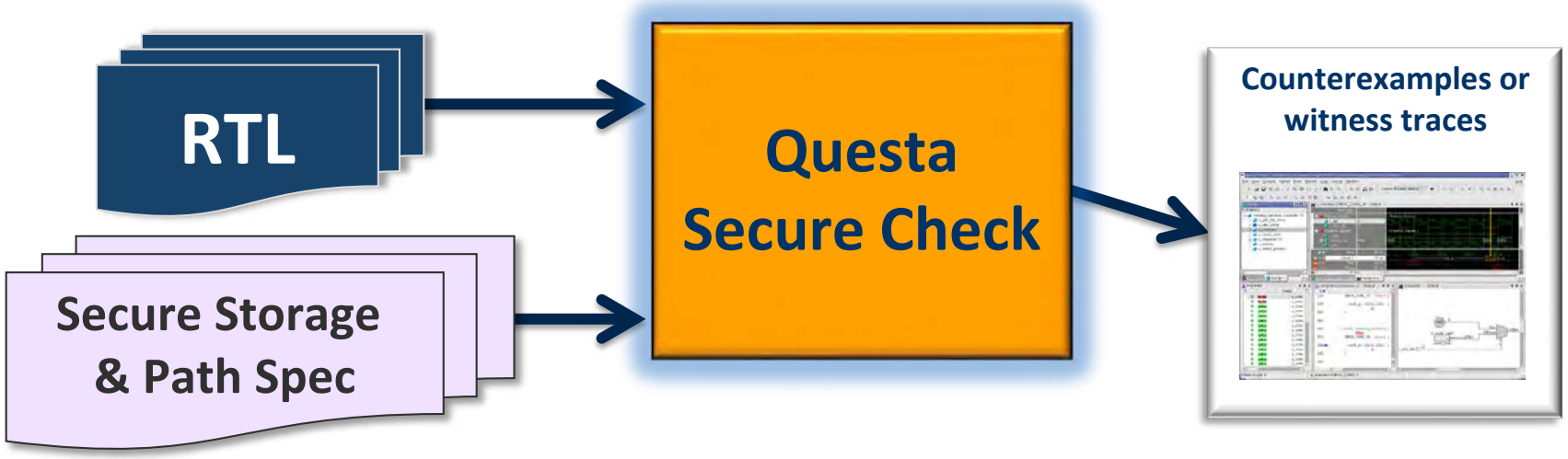


Caveat for Formal-based Security Path Analysis

- Standard formal approaches are tedious to manually configure for this problem
  - Would need hand-written assertions for each path (1,000s!)
  - Connectivity type checks can help but are incomplete
  - Checking for absence of a connection is not tractable with normal assertions and formal methods
- A specialized, formal-based app is needed for verifying paths to secure storage

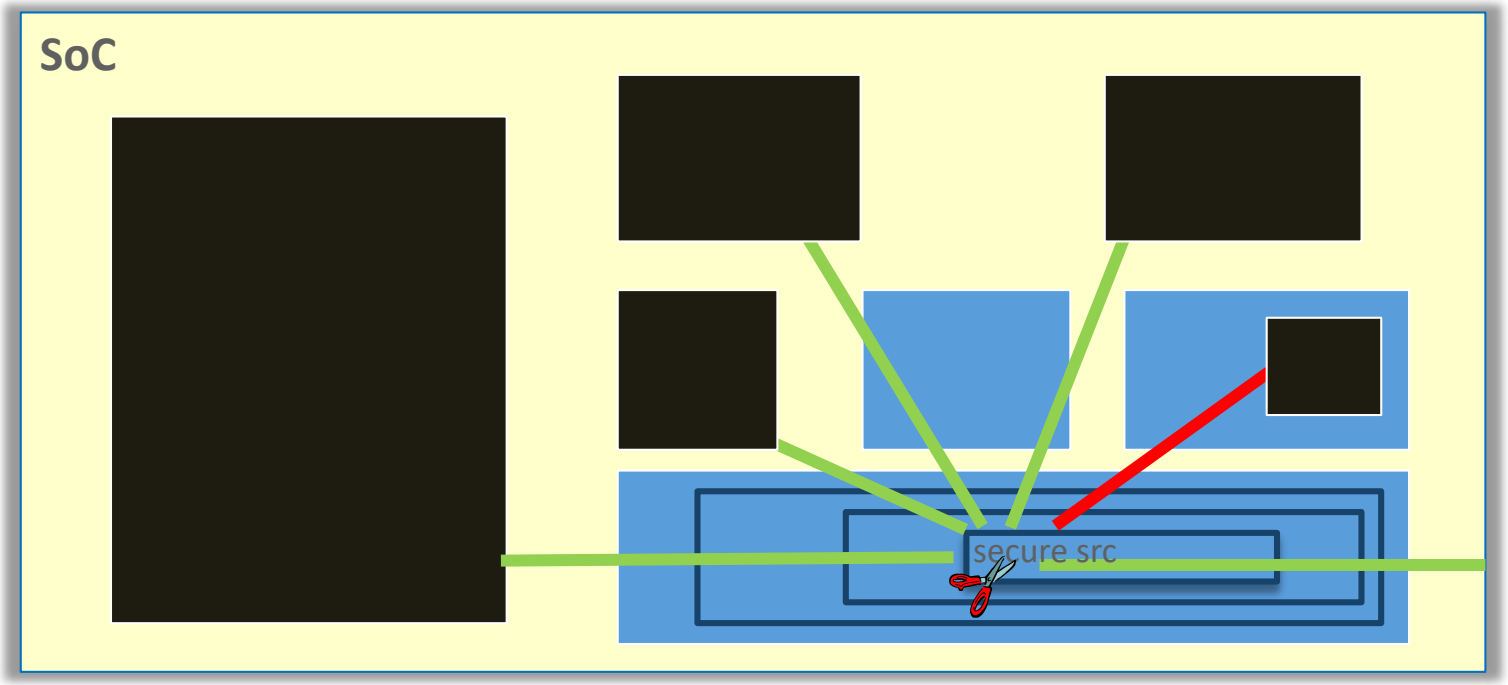


Mentor's Solution: Questa Secure Check



- Inputs: RTL & Spec. of the secure storage & signals
- Secure Check automatically finds ports/black box inputs and generates properties for confidentiality & integrity
- App automatically formally verifies these properties
- Counter example waveforms show vulnerable paths

“Black box” To Significantly Increase Scalability



Black box as much logic as you can

- Improves compile and run performance at SoC/chip level
- Can BB any module which doesn’t have path through it
- Note: make sure your path spec doesn’t enter a BB

Secure Check: Two Main Types of Checks

Context: 2 sides of the data path verification coin

- ✓ “Insecure” = path(s) exist between two points
- ✓ “Secure” = no path exists between two points

Corresponding Secure Check Analysis

1. Confidentiality
   - ◉ Checking “from” some internal secure point/interface
   - ◉ *Can the key make it off chip somehow?*

2. Integrity
   - ◉ Checking “to” some internal secure point/interface
   - ◉ *Is there some way to affect this secure item?*

Questa Verify (log/securecheck_verify.db)

File View Compile Verify Tools Reports Logs Layout Window Help

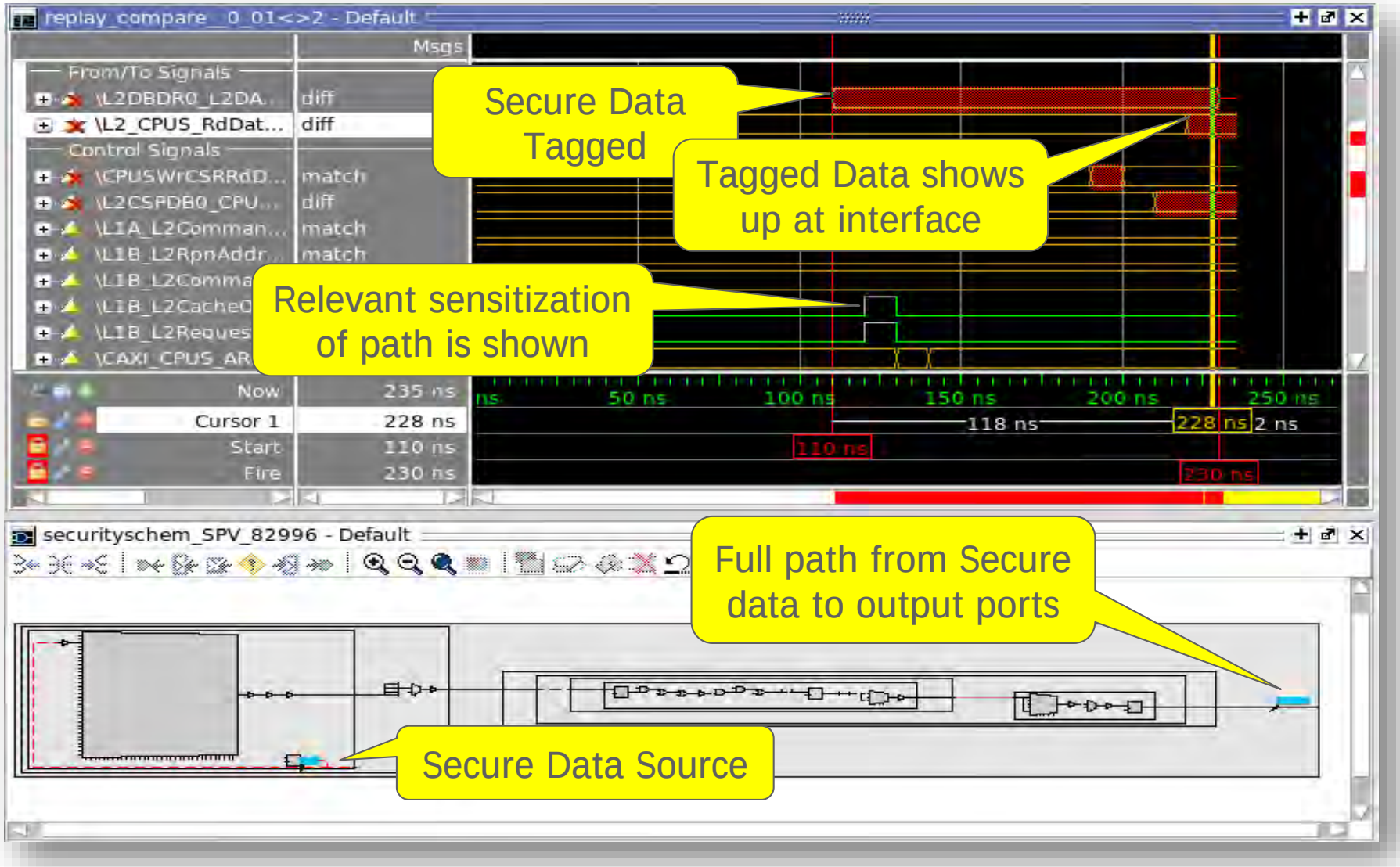
SecureCheck Layout Last Settings

Design

Instance	Design Unit	Design Unit Type	State Bits	Memory Bits
axi4lite_to_apb4 (4)	axi4lite_to_apb4	Top Module	584	684
u_axi4lite_slave (8)	axi4lite_slave	Module	13	0
u_csr_interface_apb (3)	csr_interface_apb	Module	170	128
u_fifos (4)	fifos	Module	243	556
u_master_interface (3)	master_interface	Module	158	0

Project Library Design Modules Directives Resets Clocks

Waveforms & Schematics to Follow Path of Insecure Data



Case Study: WW Consumer Electronics Maker

- Background
  - DUT: High-end entertainment systems that are under constant attack
  - Users want to hack it to work around perceived limitations
- Prior solutions
  - Manual methodologies → didn't scale
  - Software solutions incomplete → insecure paths were missed
  - Competitor formal solution → ran slow
- Secure Check App Results
  - Verified all expected secure paths OK
  - Reproduced expected insecure paths
  - Users were impressed with the speed of the analysis
  - Analysis workflow and quality of results exceeded expectations

Summary

- Security is a growing concern for many applications
  - Private key and other secure data storage
- Simulation is not nearly complete enough to verify all possible scenarios
- Normal formal methods, while more complete then simulation, also have challenges
  - Some checks can not be accomplished with traditional methods
- Questa Secure Check is the answer to these challenges
  - Built on state of the art formal technology
  - Tells whether a path is secure or insecure
  - Easy debug with waveforms and schematics to quickly get to the root cause of any insecure path

Thank You • Grazie • Merci • Gracias