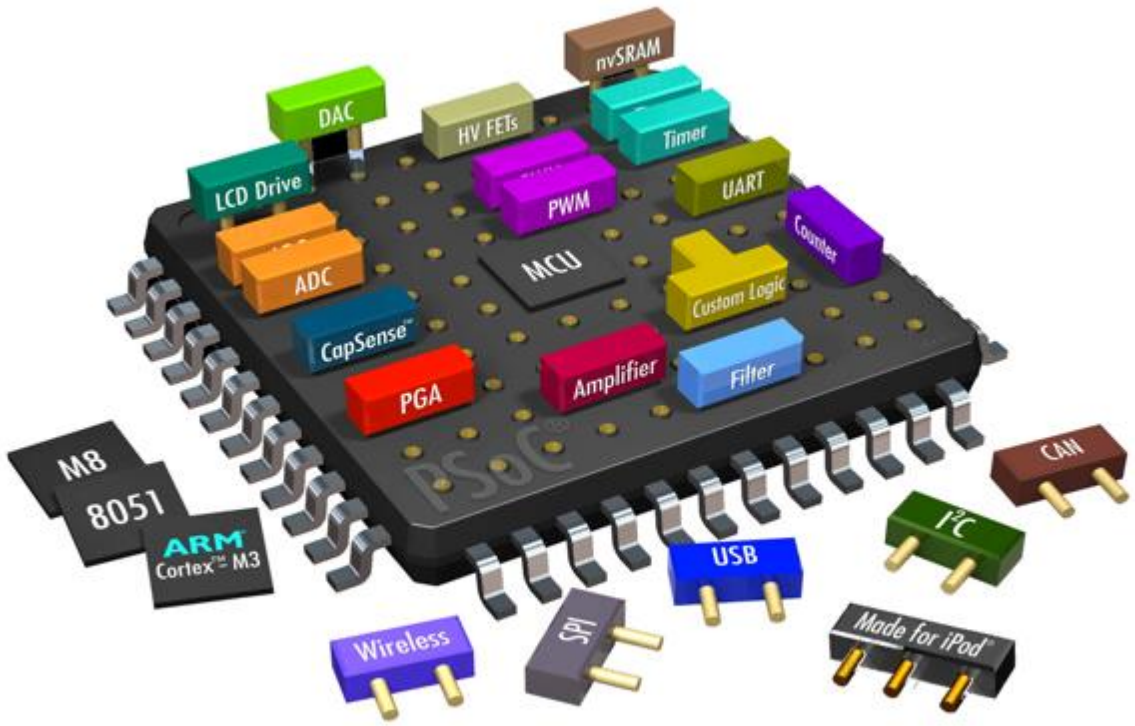
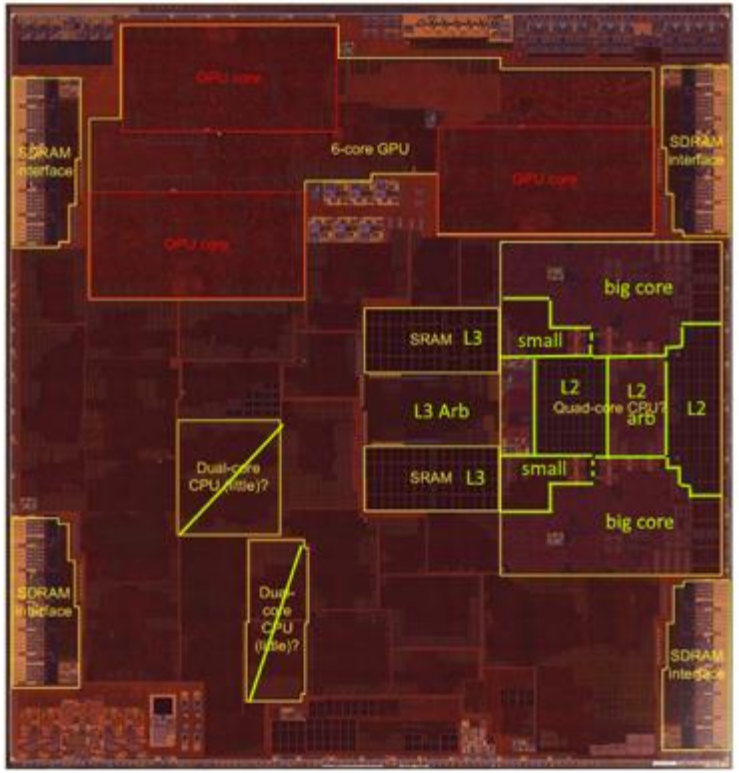
Security Verification & Validation:
Challenges and Solutions for Securing SoCs and SiPs

Mark Tehranipoor
Intel Charles E. Young Endowed Chair Professor
Chair, Electrical and Computer Engineering
University of Florida

Modern SoCs – Heterogeneous Architecture



Apple A10 Quad Core SoC

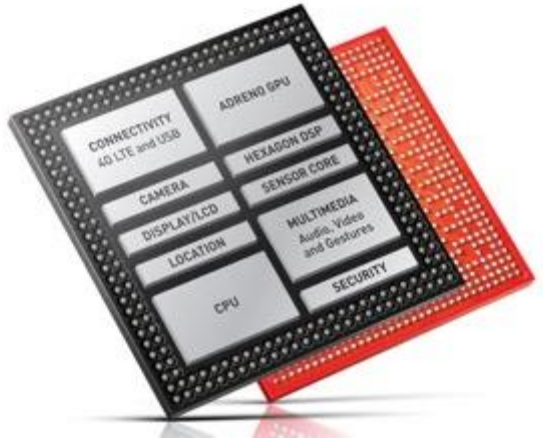


- TSMC's **16 nm** FinFET
- **3.3 billion** transistors
- Die size: 125 mm^2

@Chipworks

SoC Security: Why?

- **SoCs include highly sensitive assets that must be protected from unauthorized access**
  - Mobile devices / Smart Cards → our personal, financial and intimate physiological information (heart-rates)
- **Shortened time to market**, increased number of bugs and security vulnerabilities **Meltdown & Spectre**
- **Security compromise:**
  - Identity thefts, leakage of company trade secrets, even loss of human life



Tens of billions transistors

Designed around the globe

Many custom functionality

Tens of IPs from 3P vendors

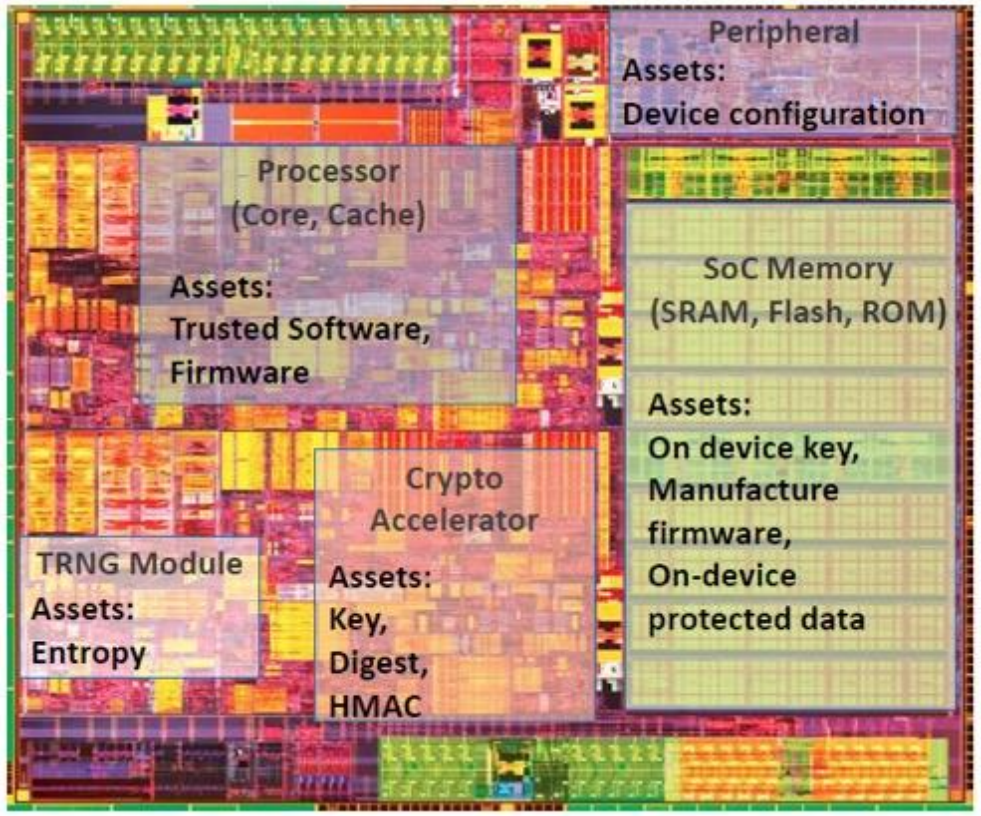
Ensuring security: extremely difficult

Security Assets

Asset: A resource of value worth protecting from an adversary

Security Assets in SoCs:

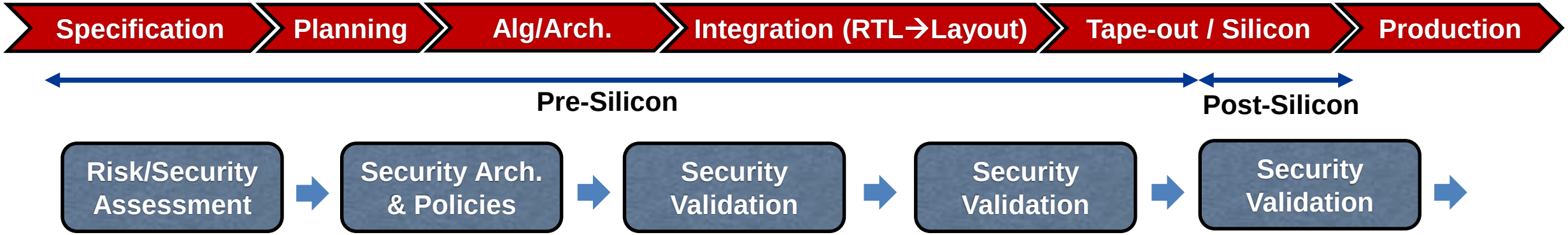
- On-device keys (developer/OEM)
- Device configuration
- Manufacturer Firmware
- Application software
- On-device sensitive data
- Communication credentials
- Random number or entropy
- E-fuse,
- PUF, and more…



Source: Intel

Security along SoC Design Life-cycle

ECE | Florida
Electrical & Computer Engineering



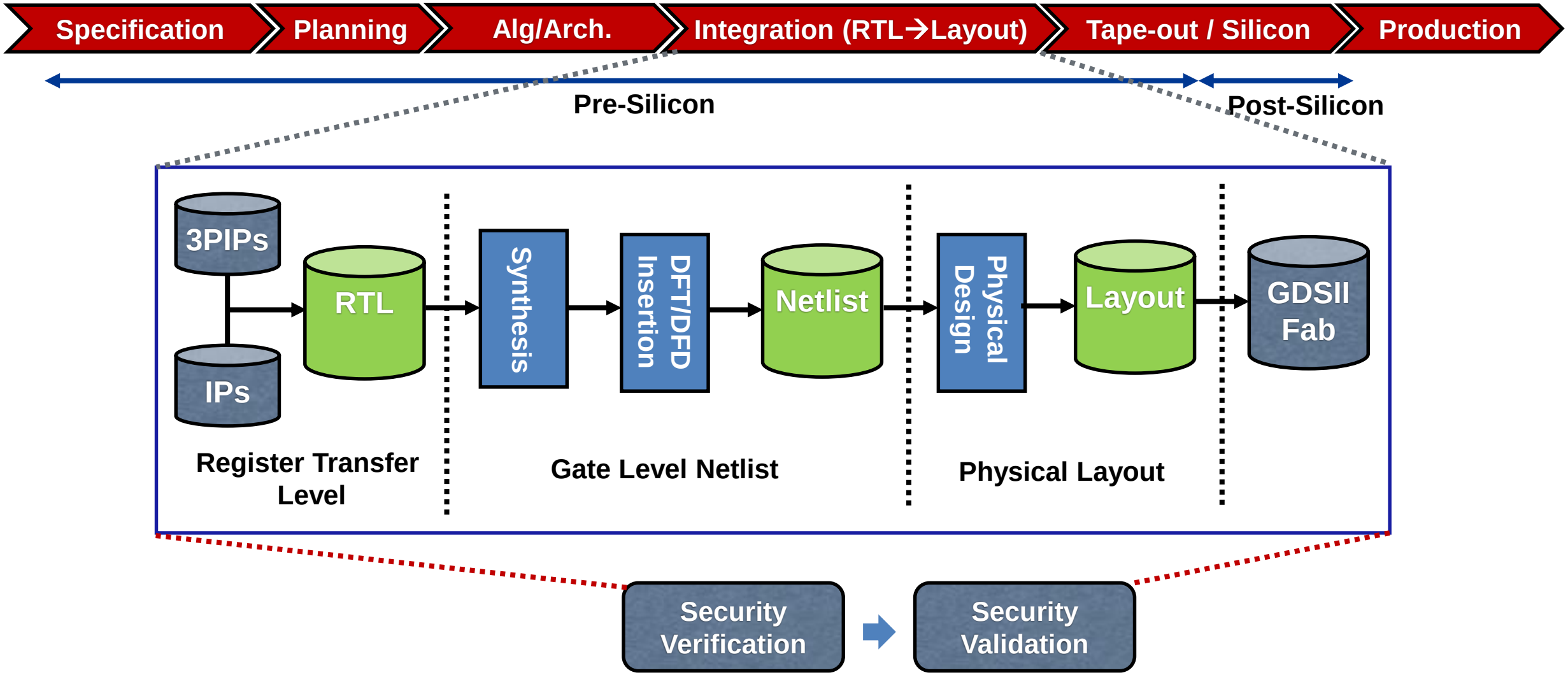
Risk Assessment
Includes definition of assets, threat model, adversaries, and security policies

Secure Architecture
Threat Modeling; Define architecture support for security; Review architecture level security Policies

Pre-silicon Security Verification:
Threat Modeling; Design review; Security verification against attacks at different stages of the design process;

Post-Silicon Security Verification:
Fuzzy test, Negative test, and penetration test

Security along SoC Design Life-cycle



Protect Assets: Strong Algorithms, Weak Implementation

ECE | Florida
Electrical & Computer Engineering

Strong Algorithm & Architecture

Weak Implementation & Execution

Algorithms, architectures, and policies could be impacted by design methods that do not understand Security!

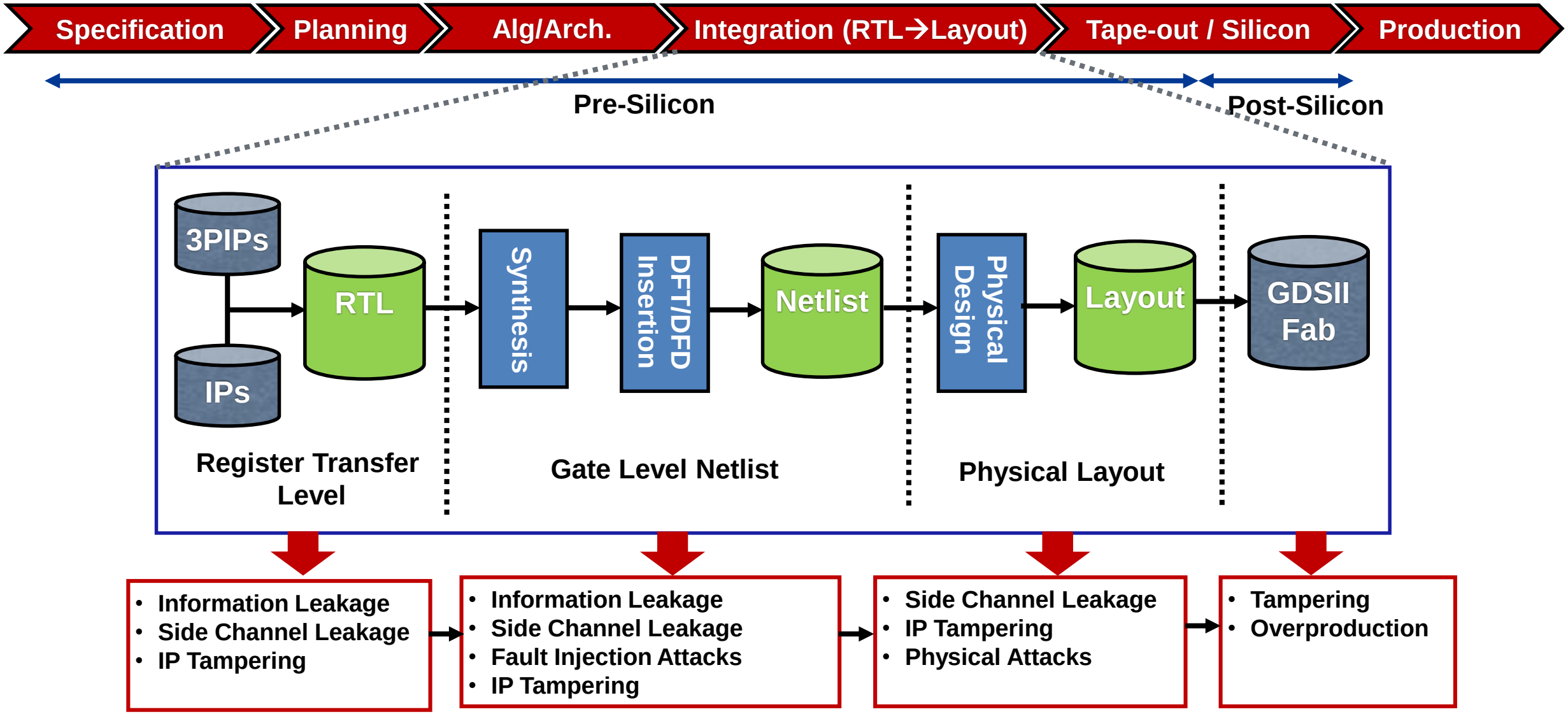
Vulnerabilities

- Information Leakage
- Side Channel Leakage
- Fault Injection
- IP Tampering, Trojan Insertion

Accesses/attack surfaces

- Remote Access (E.g., WiFi, Ethernet, Zigbee, etc.)
- PCB Access (E.g., JTAG and Debug ports)
- Physical Access

Security along SoC Design Life-cycle



Pre-silicon Security Validation

Information Leakage

Assets (secrets) being leaked to an unauthorized user or domain in the circuit; Untrusted IPs can obtain access to security assets

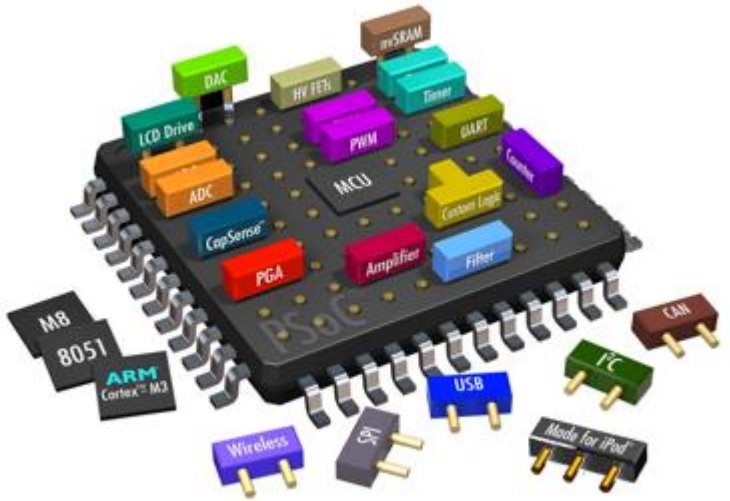
Side Channel Leakage

Involuntary signal emission providing opportunity to attackers to obtain access to secrets (timing, power, and EM)

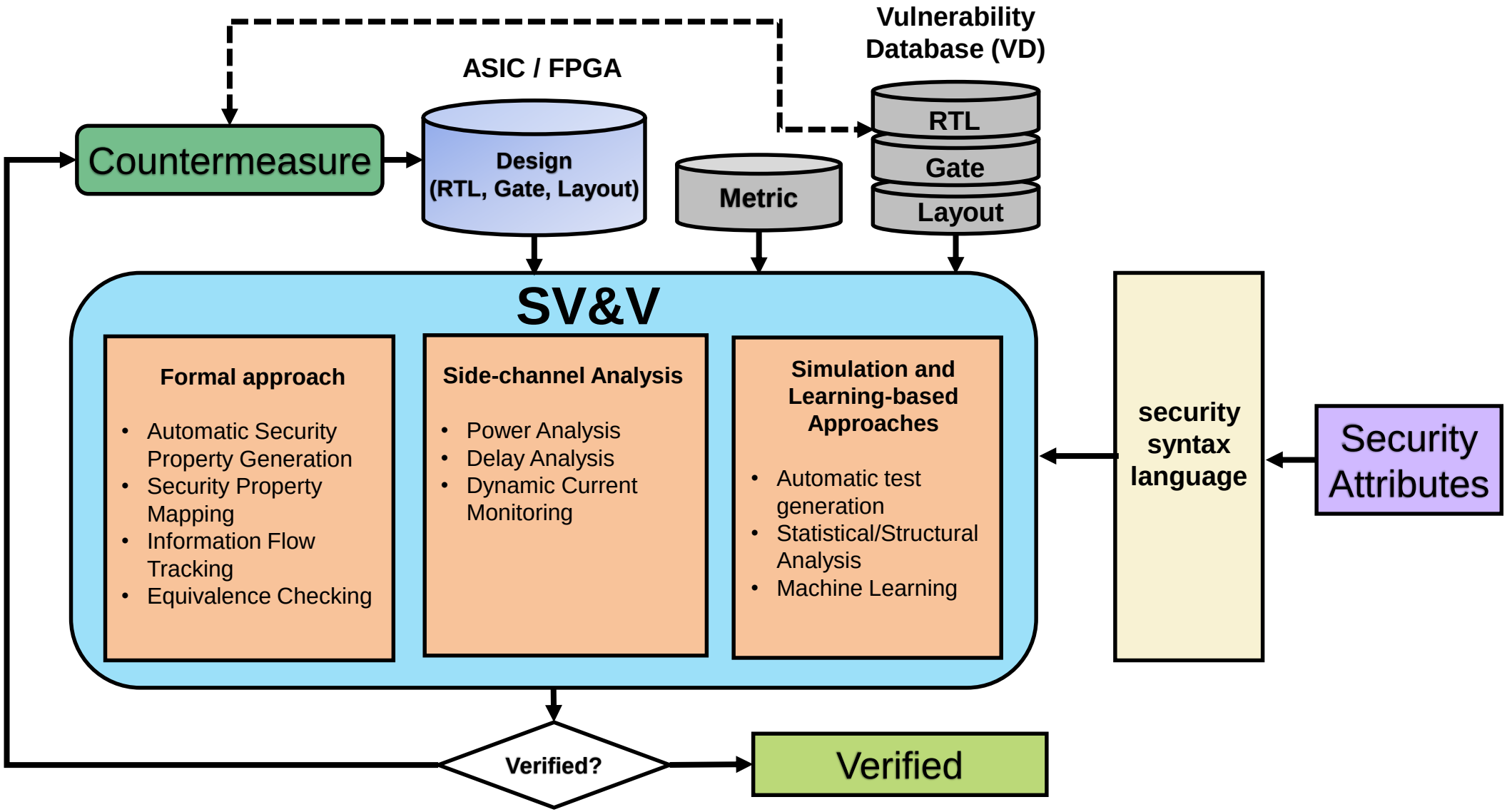
Tampering

The IPs used in SoCs are tampered by 3PIP vendors, SOC integrators, foundry, insider/designer; physically manipulating the content

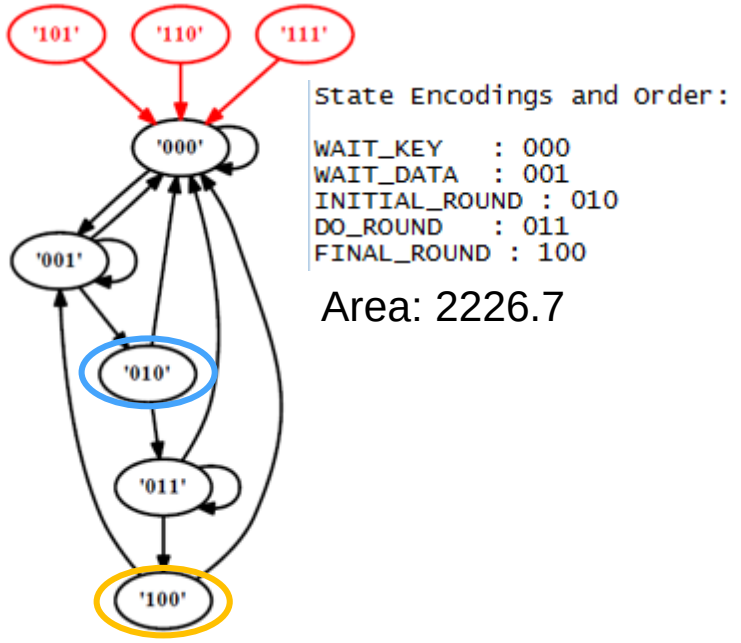
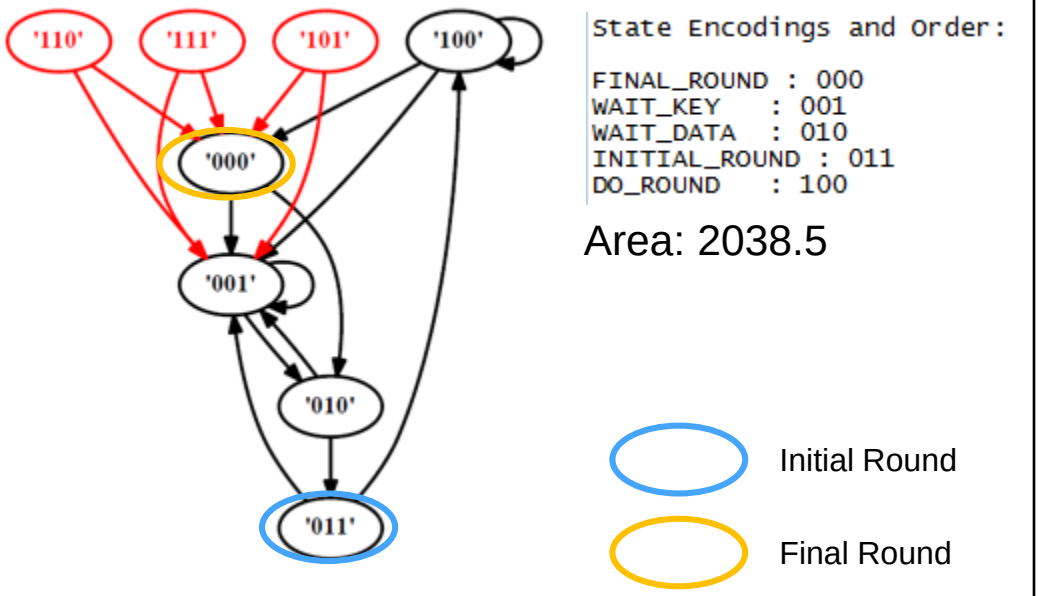
- **IP Level:** Vulnerabilities considered in modular basis at RTL, gate, and physical layout levels
- **SoC Level:** Vulnerabilities considered from system (e.g., SoC) level perspective – interaction between different cores



Security V&V


Vulnerability Database (VD)
ASIC / FPGA
RTL
Gate
Layout
Countermeasure
Design (RTL, Gate, Layout)
Metric
SV&V
Formal approach
• Automatic Security Property Generation
• Security Property Mapping
• Information Flow Tracking
• Equivalence Checking
Side-channel Analysis
• Power Analysis
• Delay Analysis
• Dynamic Current Monitoring
Simulation and Learning-based Approaches
• Automatic test generation
• Statistical/Structural Analysis
• Machine Learning
security syntax language
Security Attributes
Verified?
Verified

RTL -- Fault Injection Attack

Encoding Scheme 1	Encoding Scheme 2
	

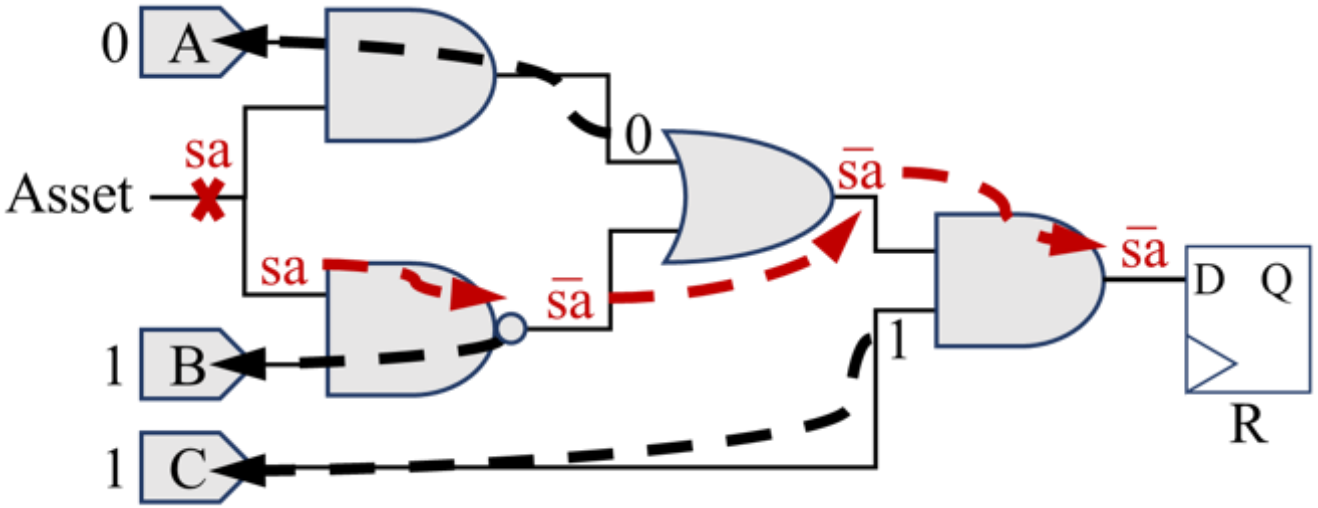
- **State encodings** impacts the vulnerabilities of a FSM

Vulnerability analysis of AES

	scheme 1	scheme 2
VF_{FI}	(0,0)	(58.9%,0.15)

Gate Level -- Information Leakage

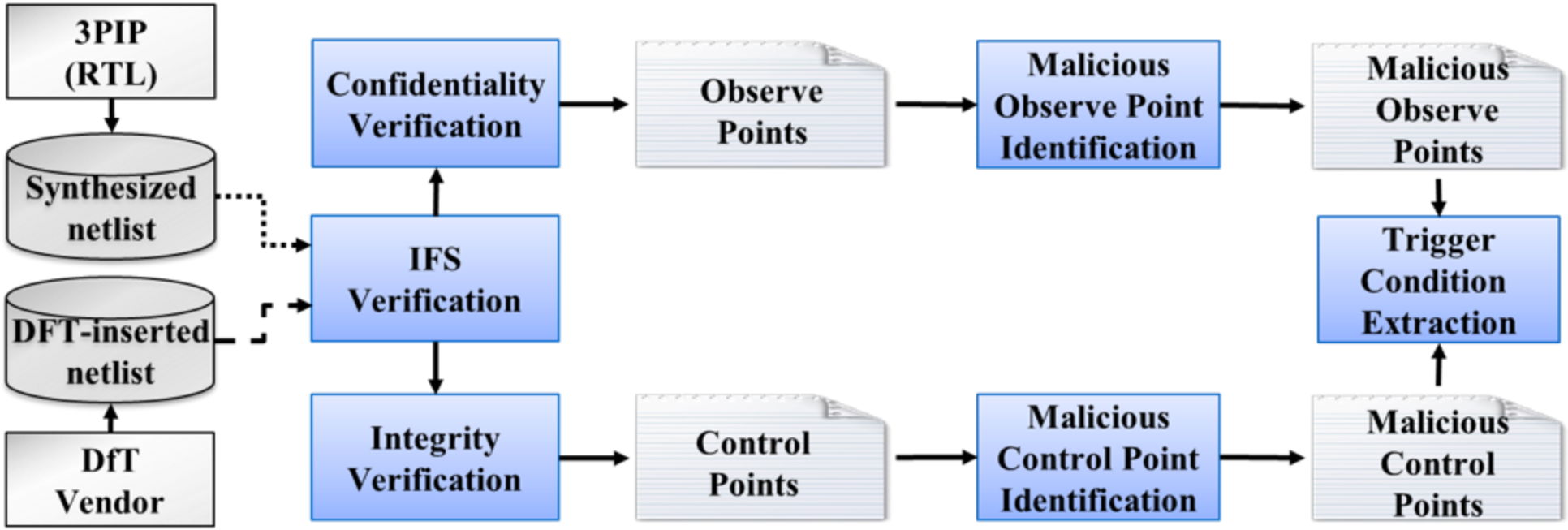
- Modeling an asset as **a stuck at fault**
- Utilize automatic test pattern generation algorithms to detect that fault
- A **successful** detection → Existence of information flow



**We need to identify all observe points →
Asset can be observed**

Malicious Leakage Detection

- **Information flow security (IFS)** verification → **Violation** of IFS policies due to malicious change/leak in the design
- **Observation:** A malicious change, however, small, will **alter** the normal information flow of a design and cause IFS violations



Confidentiality Analysis

ECE | Florida
Electrical & Computer Engineering

Benchmark	Payload	Ttrigger	# of Observe points	# of Malicious points	Time (s)
AES-T100	Leaks the key through covert CDMA	Always on	42	16	251.5
AES-T200	Leaks the key through covert CDMA	Always on	42	16	273.8
AES-T700	Leaks the key through covert CDMA	Specific plaintext	42	16	277.1
AES-T900	Leaks the key through covert CDMA	Counter	42	16	293.7
AES-T1100	Leaks the key through covert CDMA	Plaintext sequence	42	16	362.9
AES-T2000	Leaks the key through shift register	Specific plaintext	35	1	240.5
AES-T2100	Leaks the key through shift register	Plaintext sequence	35	1	350.5
RSA-T100	Leaks the key through output	Specific plaintext	37	2	19.7
RSA-T300	Leaks the key through output	Counter	37	2	20.4

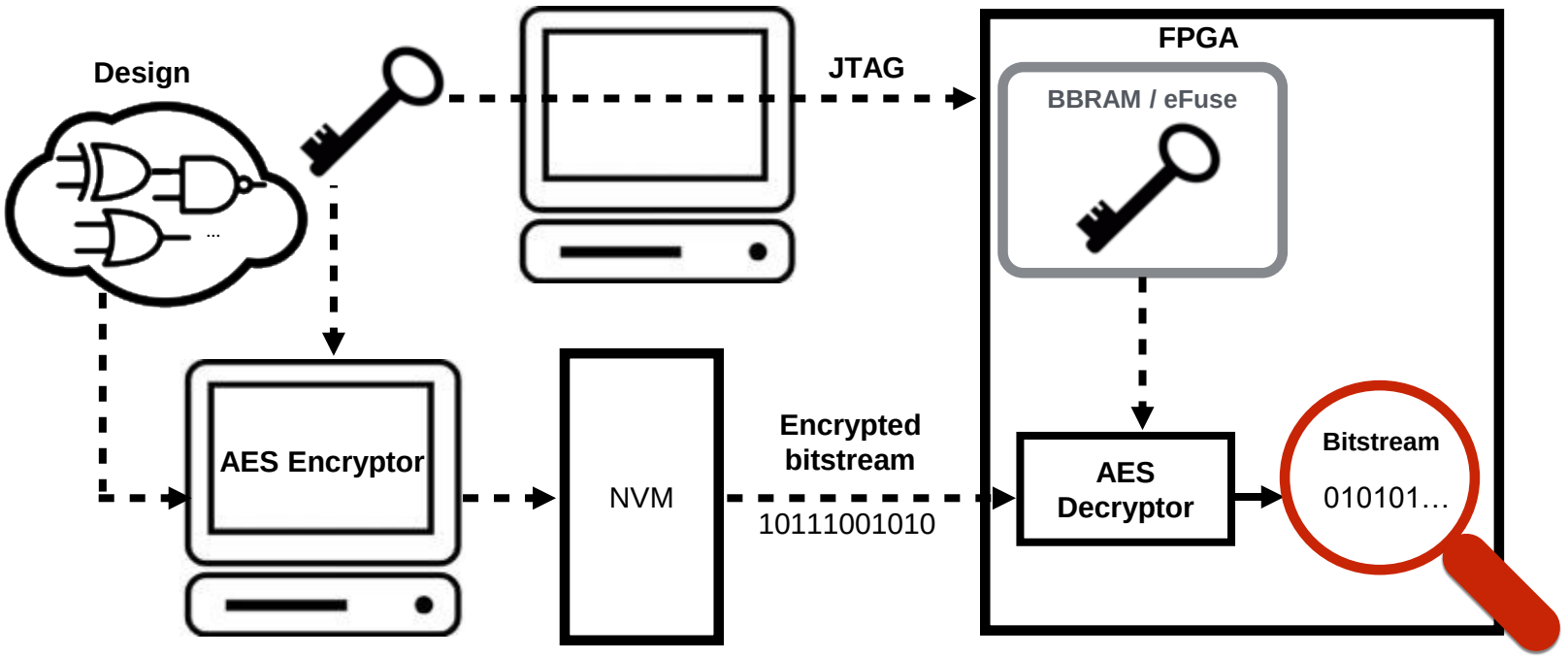
Integrity Analysis

ECE | Florida
Electrical & Computer Engineering

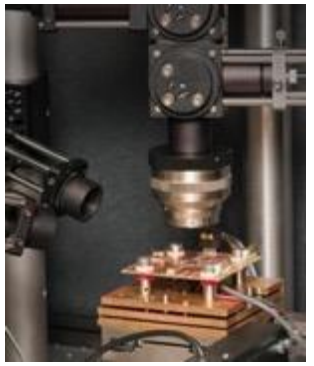
Benchmark	Payload	Trigger	# of Control points	# of Malicious points	time(s)
PIC-T100	Manipulates program execution flow	Counter	17	13	0.358
PIC-T200	Manipulates instruction register	Counter	41	14	100.5
b19-T500	Manipulates instruction register	FSM	193	2	211.6
RS232-T500	Manipulates a control signal	Counter	13	2	0.381
s35932-T100	Manipulates scan mode	Counter	23	23	1.905
RSA-T400	Replaces the key to leak plaintext	Counter	34	33	20.2

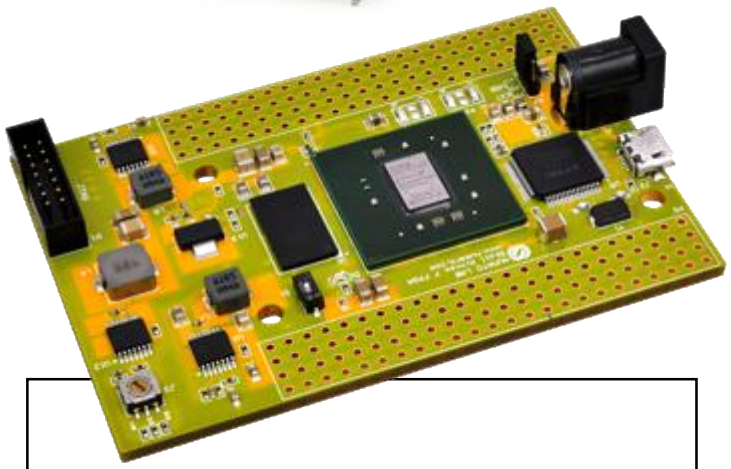
Attacking Bitstream Encryption of FPGAs

ECE | Florida
Electrical & Computer Engineering



Hamamatsu PHEMOS - 1000





- Device under Test (DUT): Xilinx Kintex 7 development board
  - Chip's technology: 28 nm
  - No chip preparation (e.g., depackaging, silicon polishing, etc.)
- Optical Setup: Hamamatsu PHEMOS-1000
  - Laser wavelength: 1.3 μm
  - Laser spot size: >1 μm

- Non-destructive
- Non-invasive
- No Footprint

Localizing the Configuration Logic

Xilinx Kintex 7 in flip-chip package

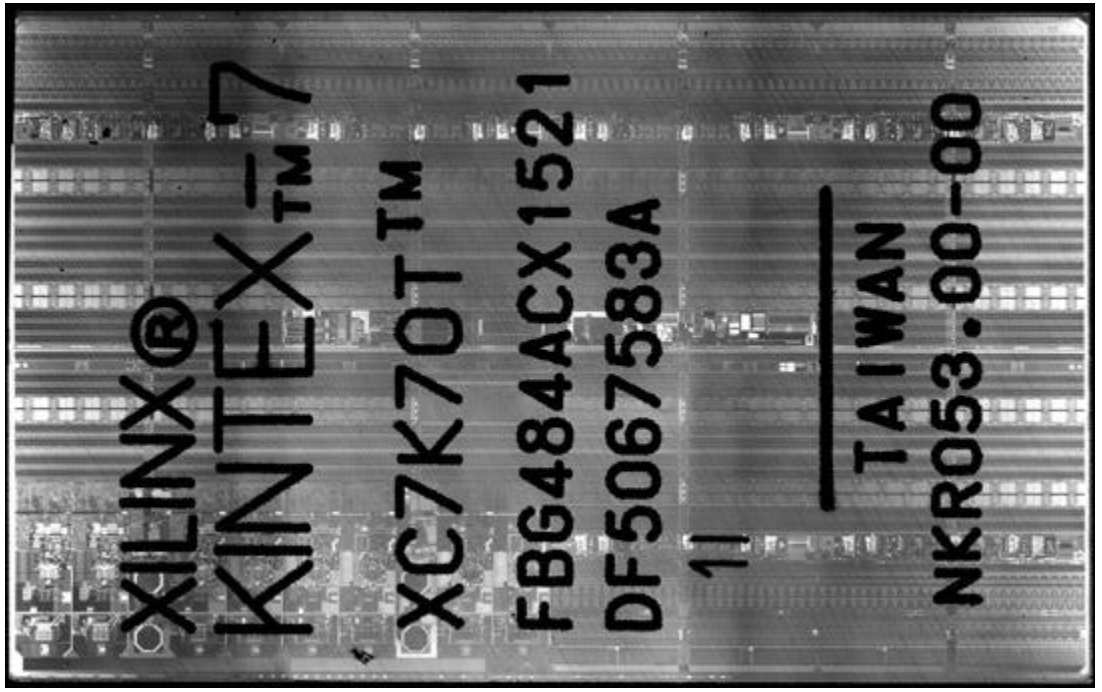
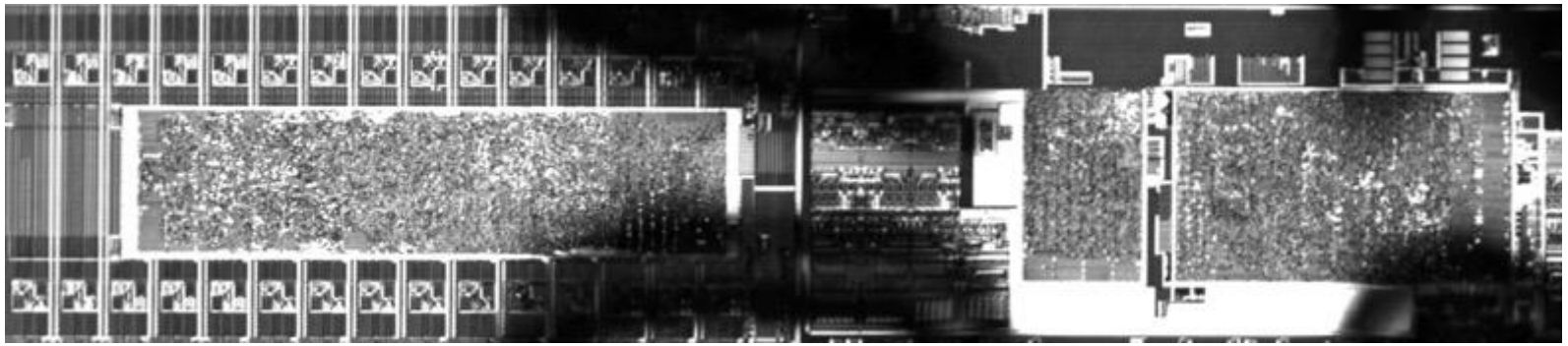


Image acquisition with a infra-red laser scanning microscope

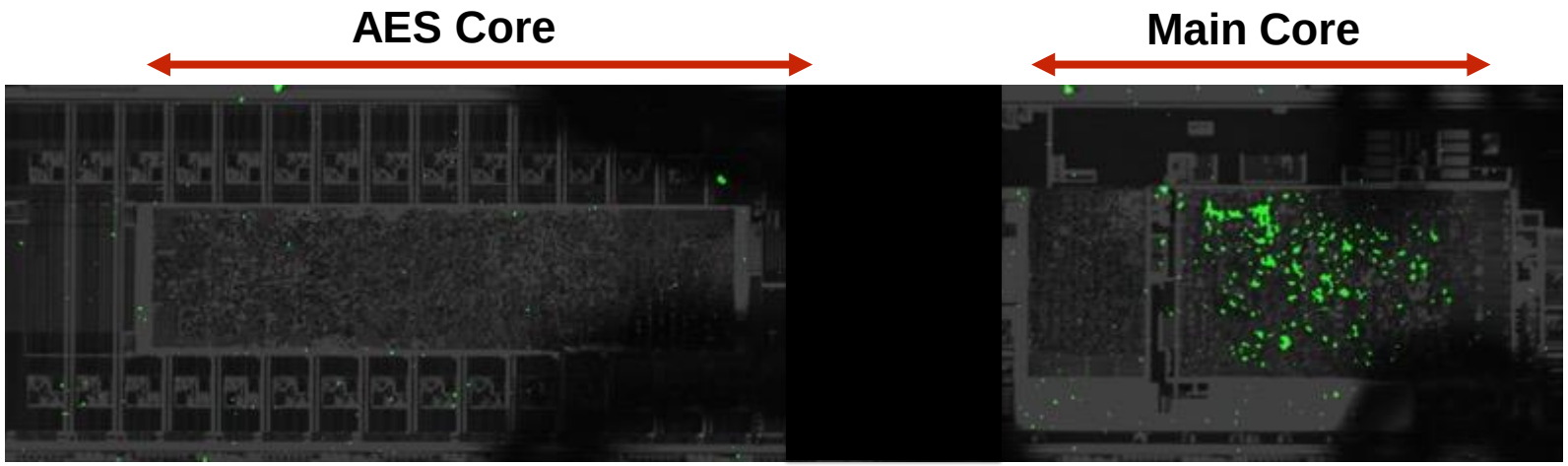
Tajik, S., Lohrke, H., Seifert, J. P., & Boit, C. "**On the Power of Optical Contactless Probing: Attacking Bitstream Encryption of FPGAs**," In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security.

Localizing the Configuration Logic



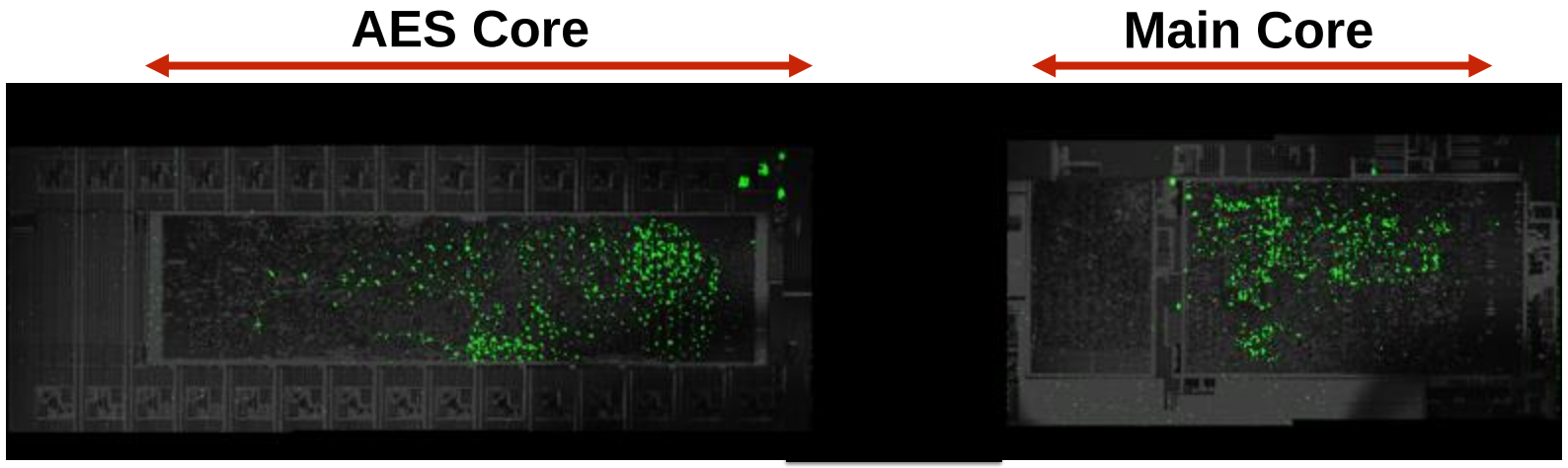
Random Logic

Localizing Decryption Core using EOFM



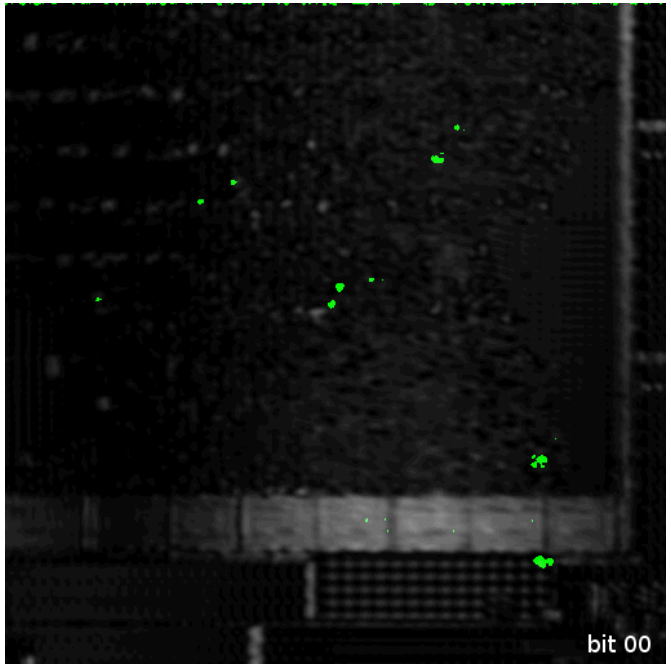
Clock activity for unencrypted bitstream

Localizing Decryption Core using EOFM



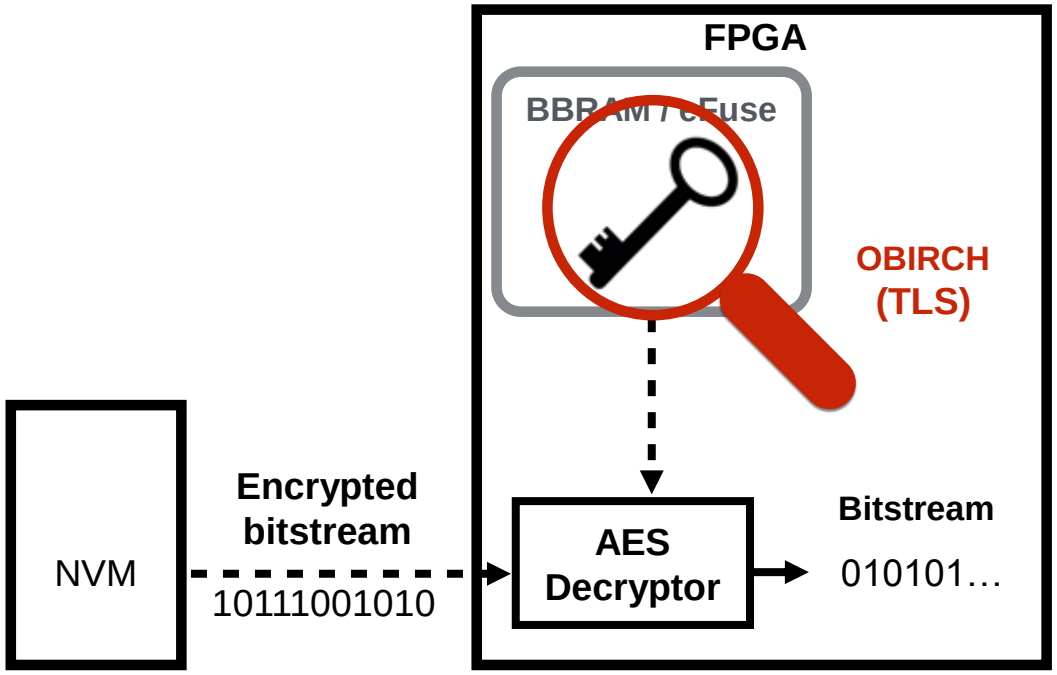
Clock activity for encrypted bitstream

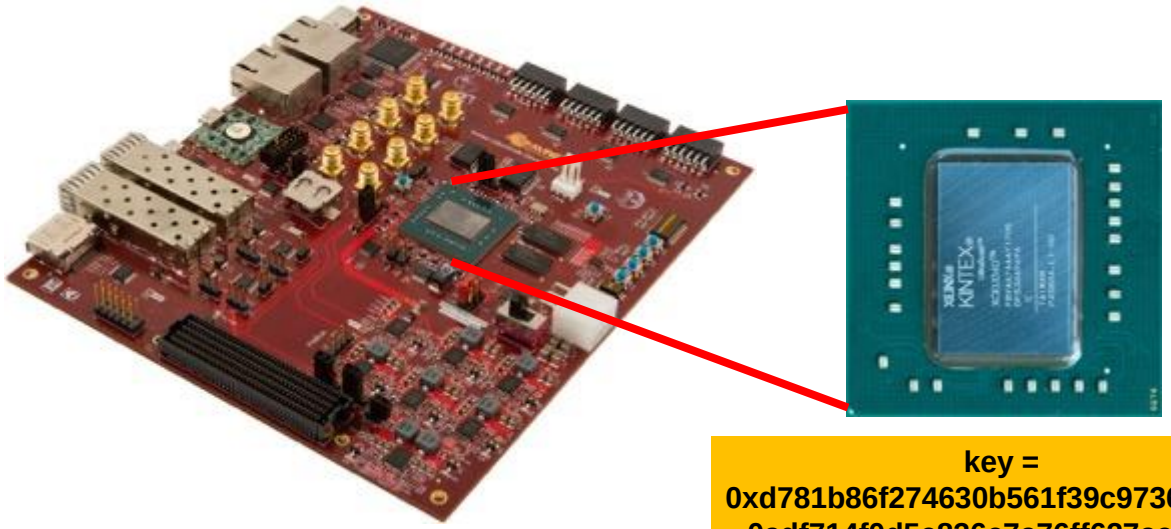
Locating the plaintext data



Locations in AES output port

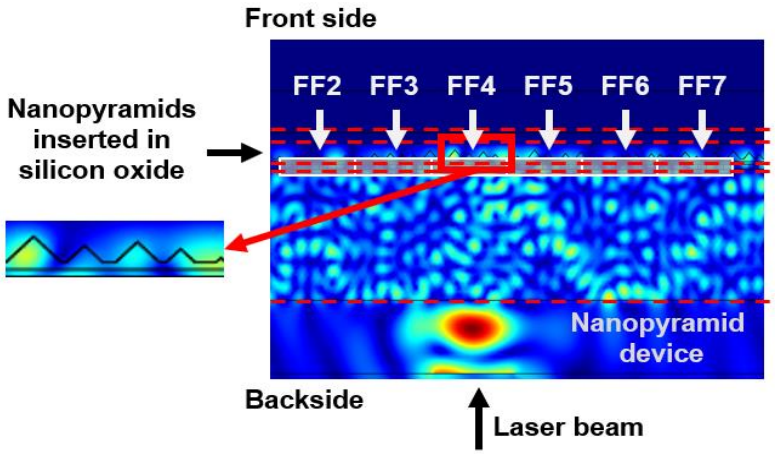
Key Extraction

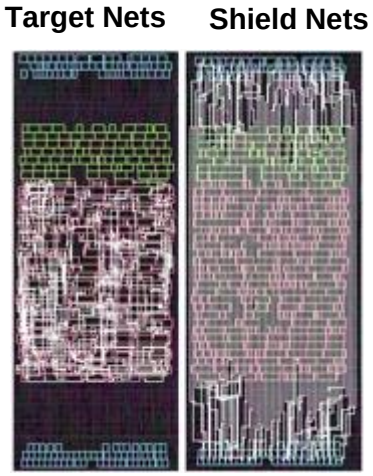




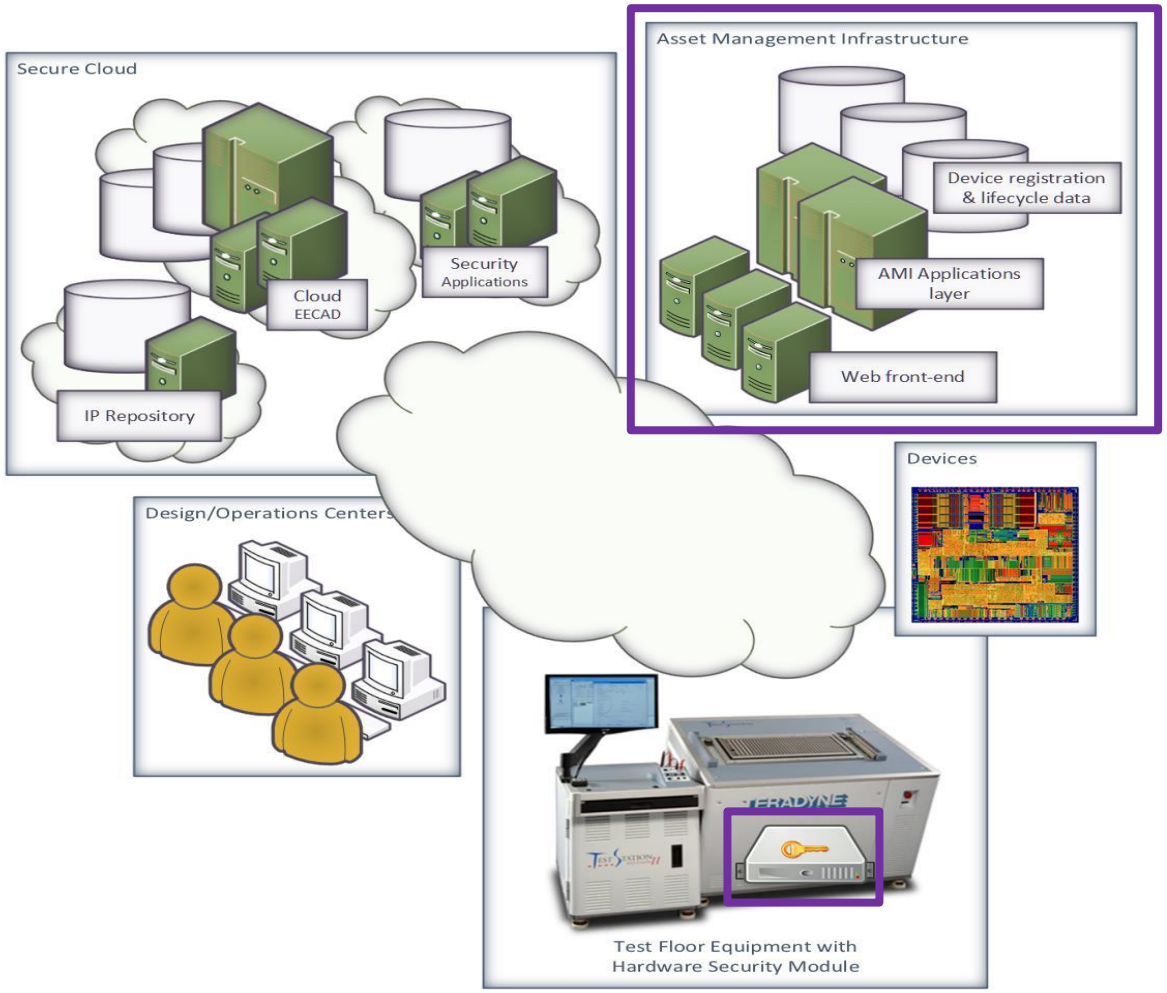
key =
0xd781b86f274630b561f39c9736f512eb
0adf714f0d5c836c7a76ff627aca4923

- **Protection**
  - **Circuit Level Solutions**
  - **Device Level solutions**
  - **Material Level Solutions**



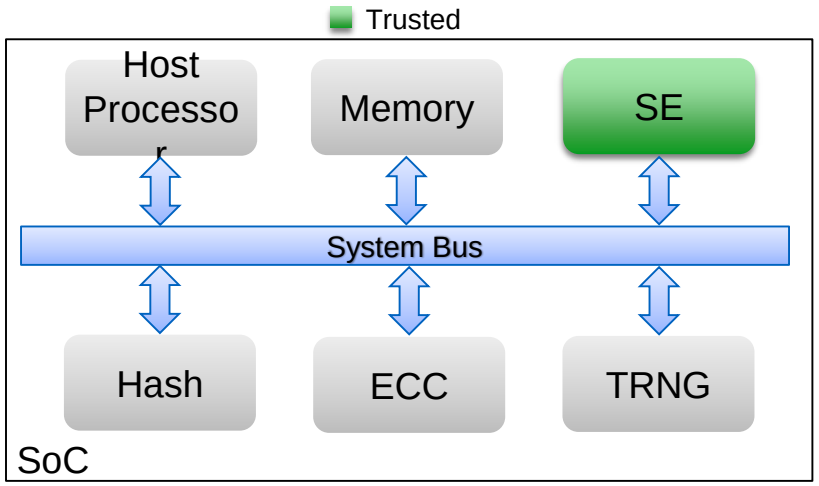


System Architecture – Root of Trust



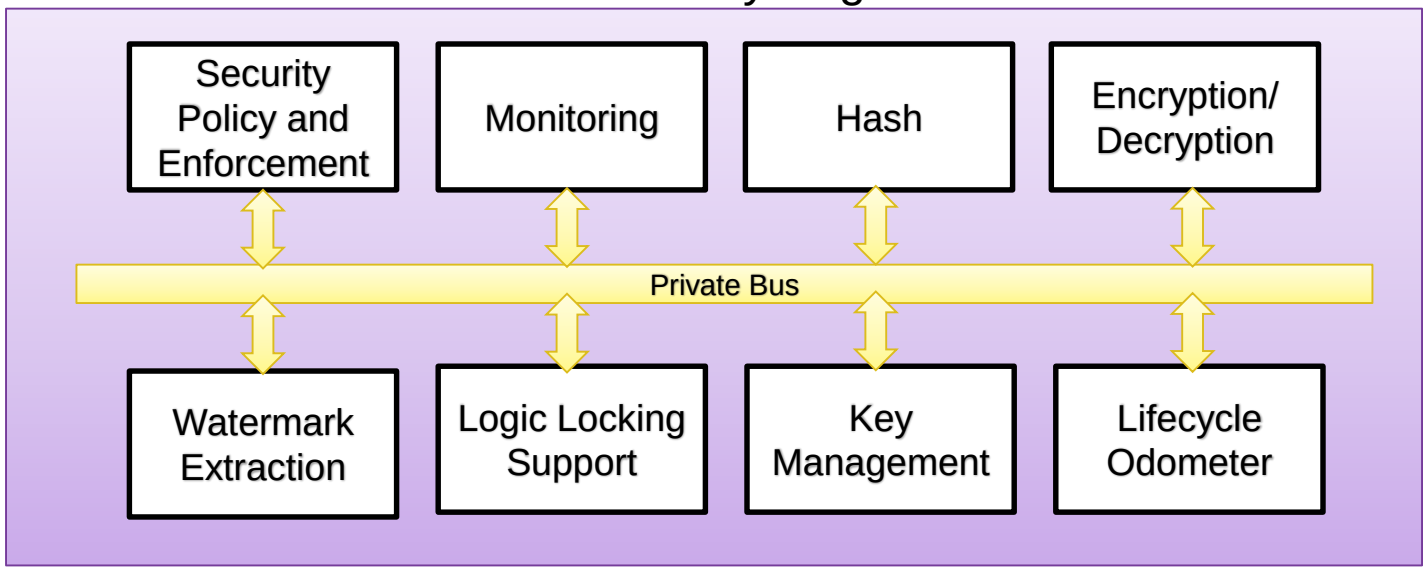
- **Asset Management Infrastructure**
  - Cloud-based database solution
  - Provides support for
    - Device Registration
    - Authentication
    - Security assets provisioning
    - Track device lifecycle
    - More.
- **Hardware Security Module**
  - HSM installed with test equipment.
  - Provide proxy for AMI and chip communication.
  - HSM generates security assets or transfers the assets from AMI to chip.

SE Components



SoC Architecture with SE

Security Engine



And More..

Secure Advanced Packaging and Heterogeneous Integration

Motivation & Problem Statement

ECE | Florida
Electrical & Computer Engineering

The semiconductor industry is moving towards rapid adoption of functionally disaggregated hardware

- New demanding server workloads and the slowing down of Moore's Law
- The significant performance/watt benefits of domain-specific accelerators
- The exponential cost of silicon development, especially at newer process nodes
- The economies of building chiplets instead of monolithic chips
- Availability of best-of-breed components as chiplets at optimum process nodes from multiple foundries

Impact

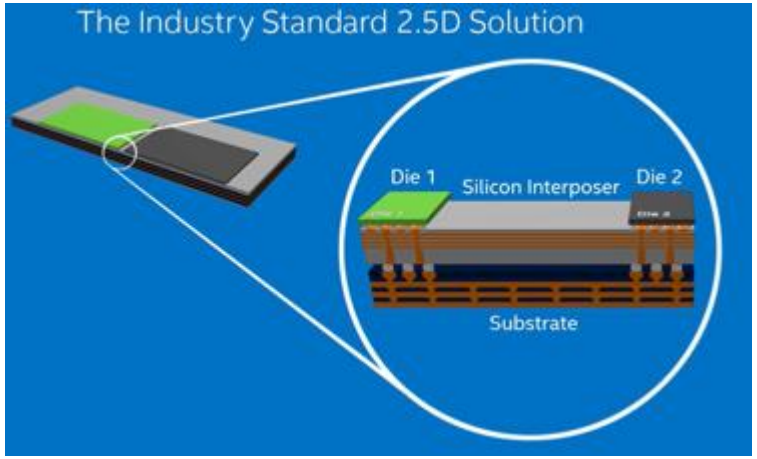
- Flexibility
- Scaling can continue
- Accessibility
- Reuse of expensive IPs
- Cost-efficient

Challenges

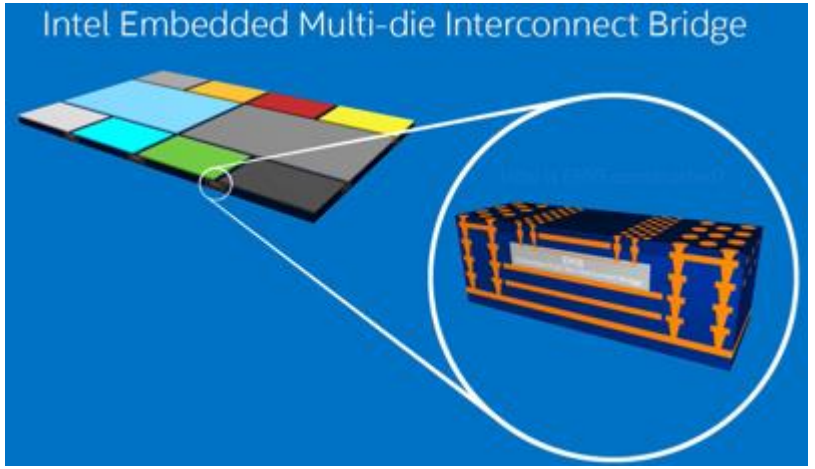
- New attack surfaces making it vulnerable to various existing and emerging threats

Heterogeneous Integration

Industry Heterogeneous Packaging

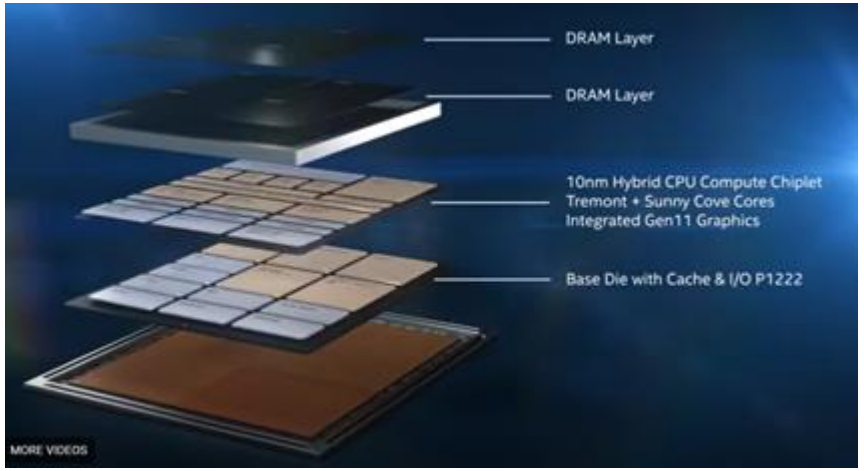


Intel Embedded Multi-die Interconnect Bridge EMIB (passive & active)



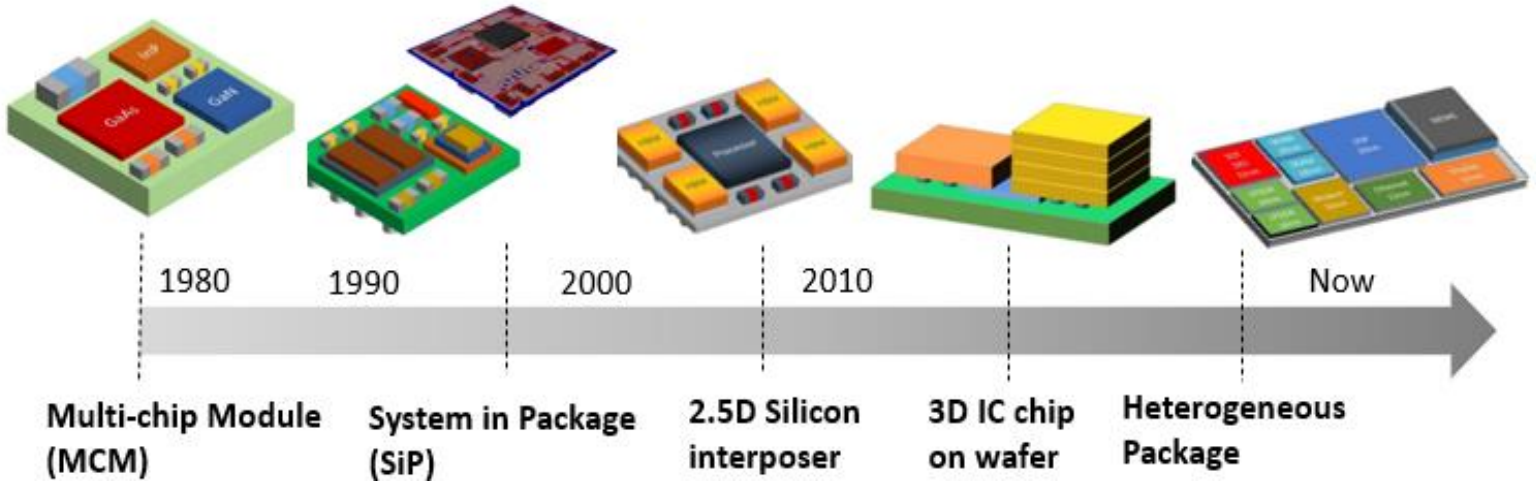
Source: https://www.intel.com/content/www/us/en/foundry/emib.html

Intel Foveros 3D Stacking Technology



Source: https://newsroom.intel.com/press-kits/lakefield/#gs.rdd753

Courtesy: Intel

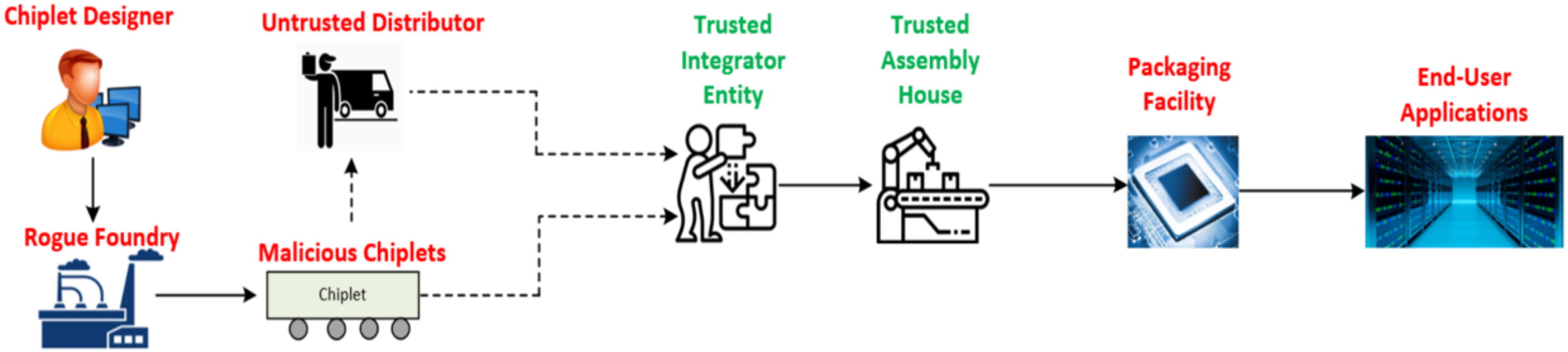


Assumptions

ECE | Florida
Electrical & Computer Engineering

- Some chiplets may be trusted, some may not
- Untrusted semiconductor fab
- Untrusted interposer layer
- Untrusted package substrates manufactured off-shore
- **Trusted facility for integration and assembly**

Supply Chain of Heterogenous Integration – Threat Model



Pre-Integration Chiplet Development and Fabrication

- IP piracy
- Hardware Trojans and malicious alterations
- Reverse engineering
- Counterfeit chiplets

Heterogeneous Integration and Assembly

- Information leakage
- Confidentiality and integrity
- Security policies

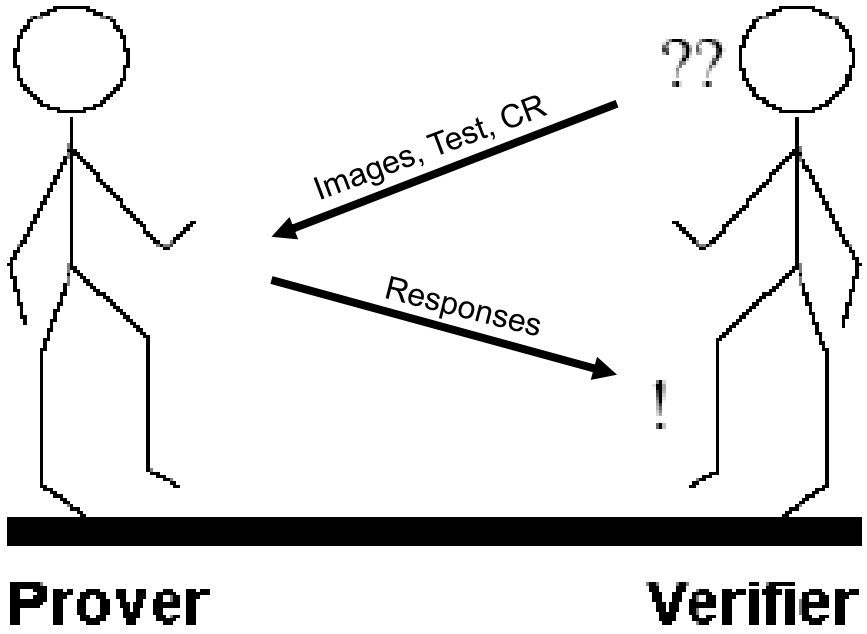
Packaging and In-field Operations

- Secure run-time operations throughout the lifetime
- Tamper detection
- Firmware security
- Supply chain integrity
- Physical attacks (side-channel and fault injection)
- Secure deployment and upgrade of chiplet firmware

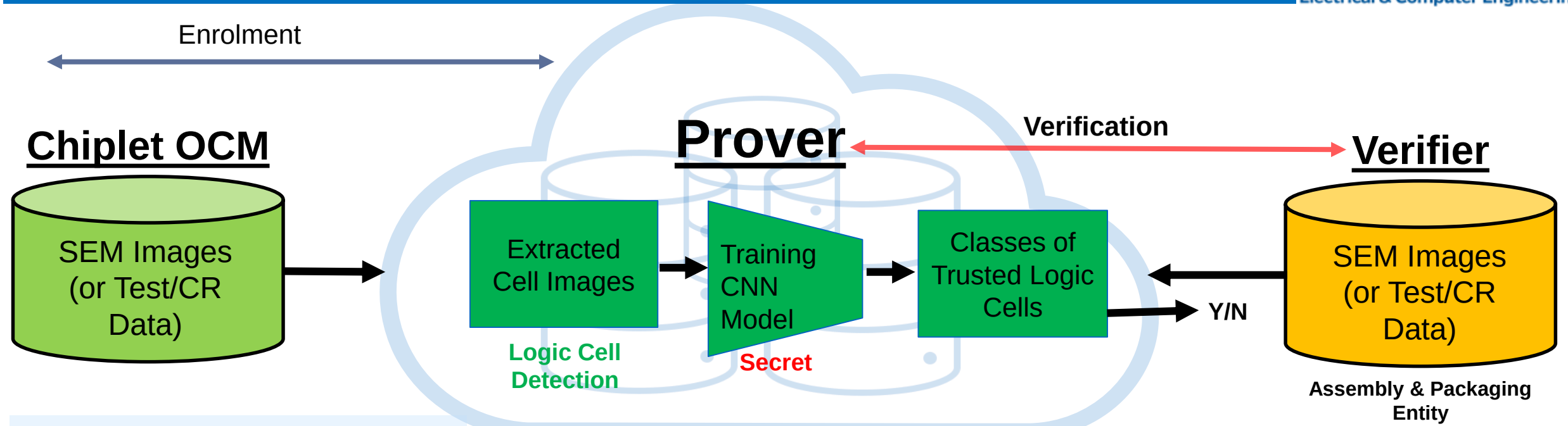
Chiplet Security

- **Logical Verification**
  - Attackers: Untrusted Chiplet OCM and foundry
  - Challenge-response (CR) based approach
    - Logical test, watermark, PUF, etc
    - *Insufficient to establish trust*

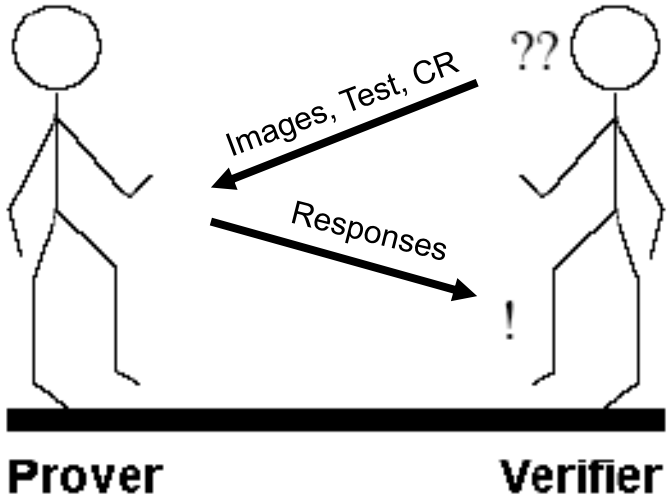
- **Physical Verification**
  - Attackers: Untrusted foundry
  - OCM is trusted
  - Imaging based approach to detect any change made by the untrusted entities



Enrolment & Verification



- Chiplet owner voluntarily participates in this exercise
- Increase in trust and market share



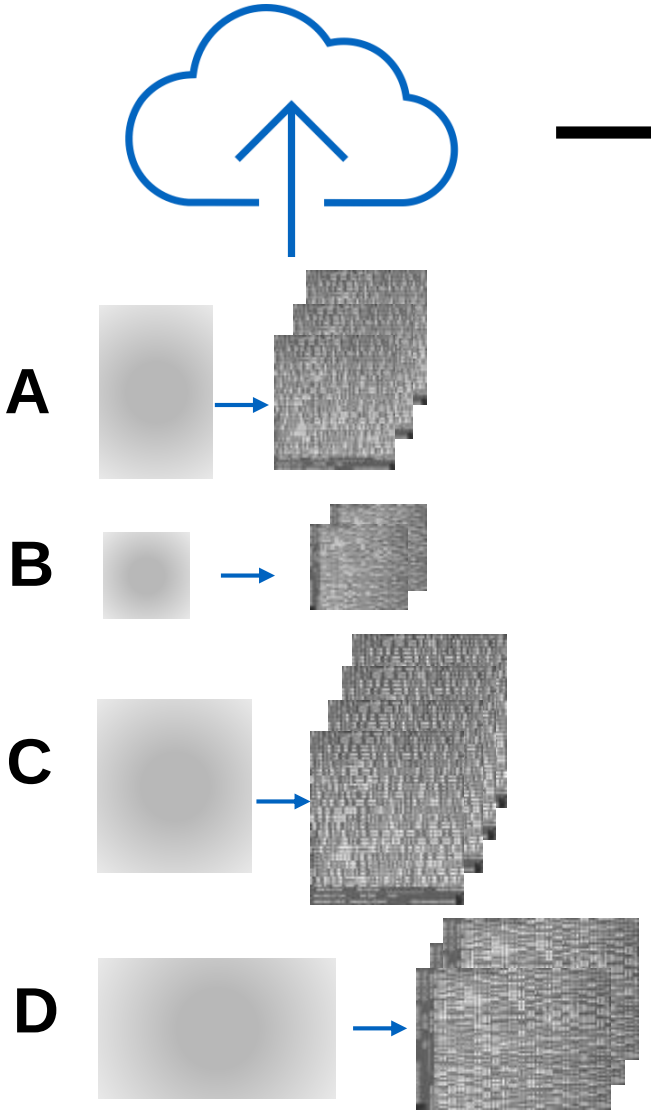
Physical Verification

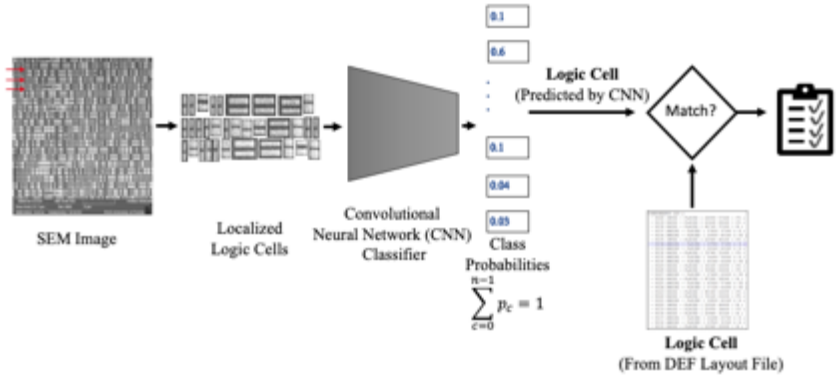
ECE | Florida
Electrical & Computer Engineering

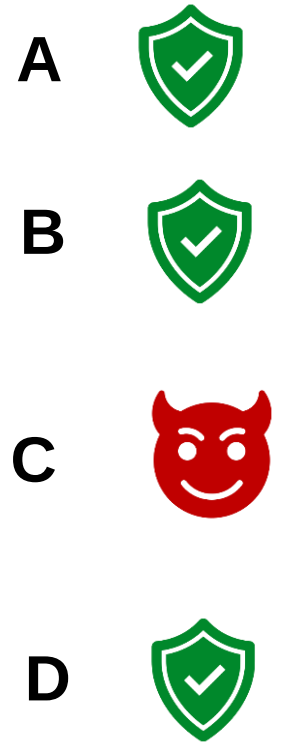
Verifier
Backside SEM Images

Prover
Malicious Change Detection

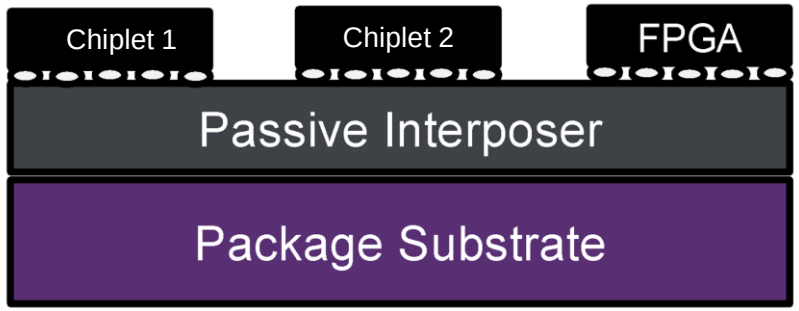
Outcome







Secure Integration and Lifecycle Assurance



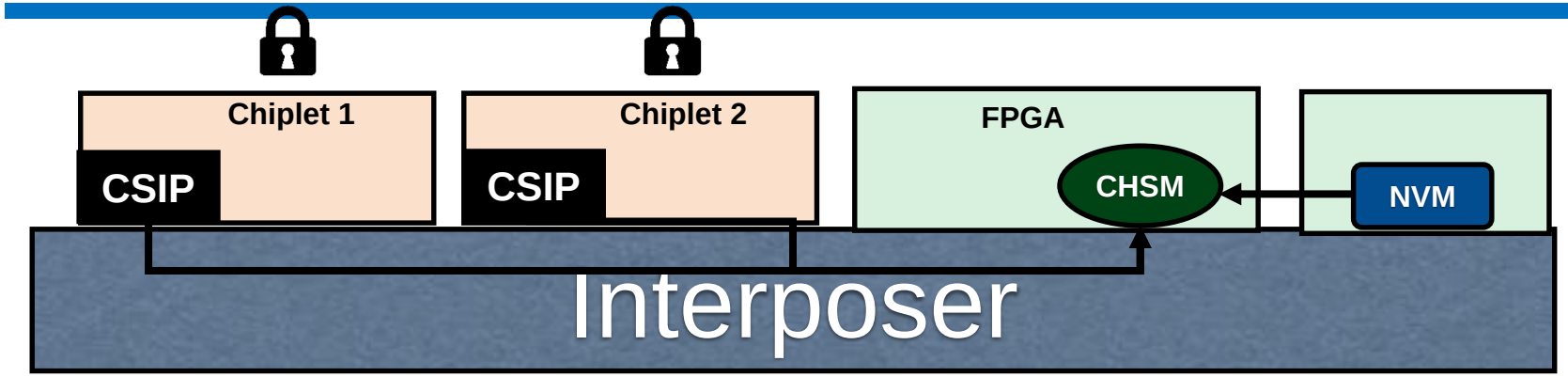
Brief Description:

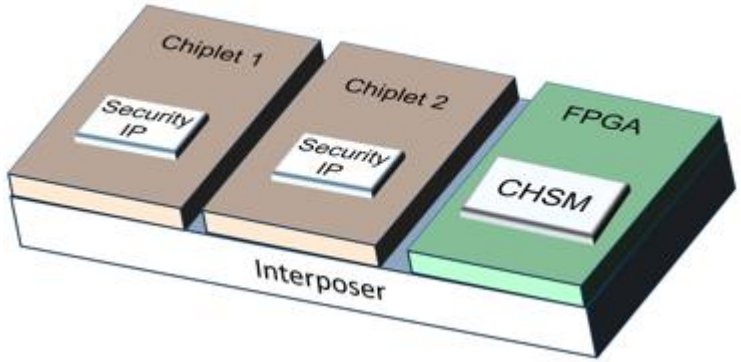
- Chiplets 1 and 2 fabricated using advanced technology node in untrusted foundry
  - Sensitive chiplets could be *locked* or have *stripped* functionality
- The FPGA is configured by the IC designer and the configuration data, i.e., bitstream, is unknown to the potential adversaries

Security Features:

Supply chain integrity: Enables end-to-end provenance and traceability for the package and each chiplet	**Locking/Unlocking and Obfuscation**: Enables secure key exchange between chiplets and FPGA
Runtime monitoring: Detect malicious attacks to device's firmware, malware, ransomware, Trojans, etc	**Tamper detection**: Detect any tampering including X-ray, optical, clock glitch, voltage glitch, Laser fault injection, etc.

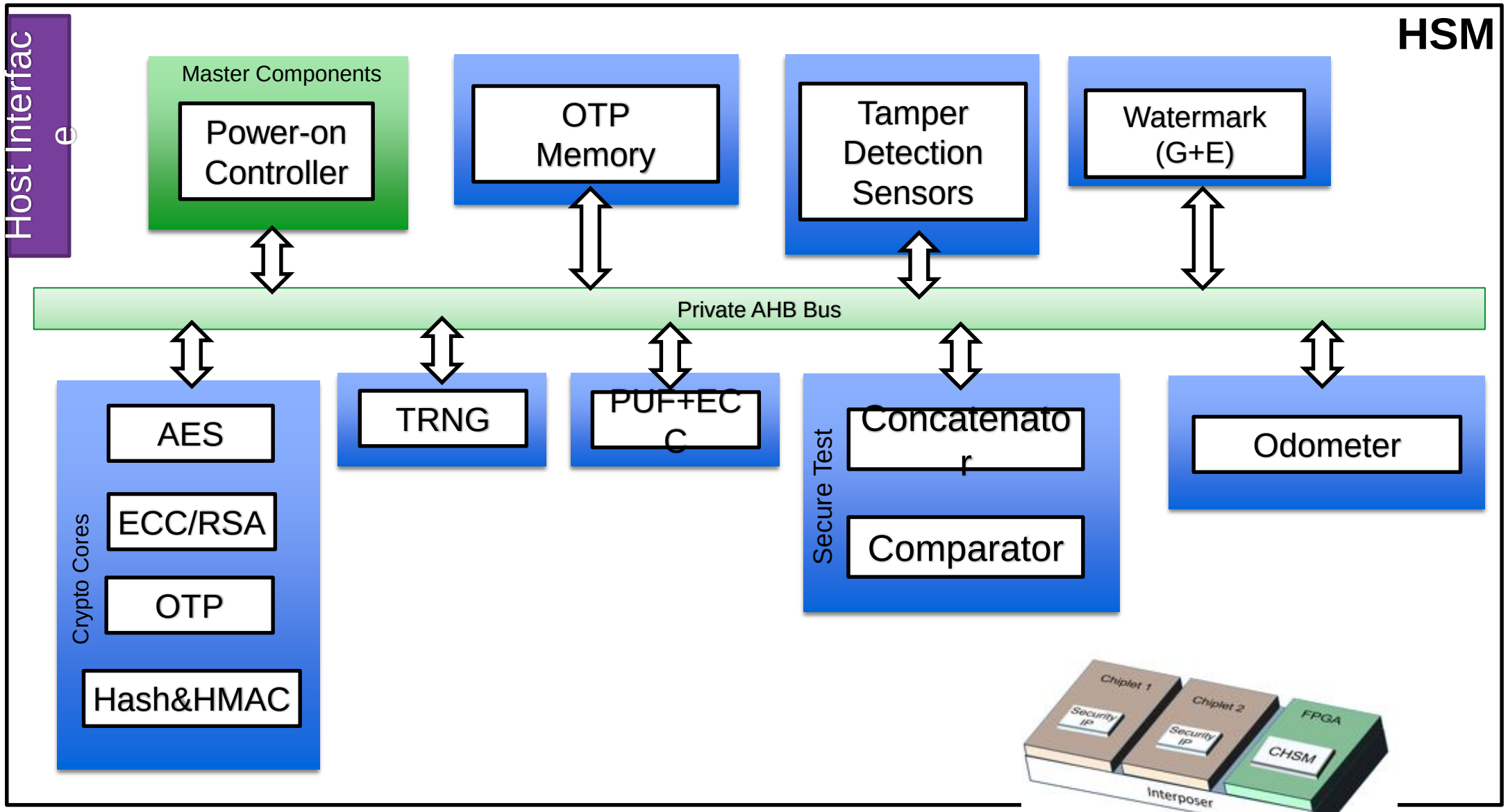
Secure Integration and Lifecycle Assurance





- Each chiplet must be **authenticated**
  - Challenge-response protocol
- Some chiplets may be **logic locked,** each requiring a separate key to unlock its functionality.
  - Logic locking keys should not be securely hard coded in the netlist or provisioned by the untrusted foundry.
  - The logic locking keys should not flow through the interposer in plaintext
- **Chiplet Security IP (CSIP)**
  - Some chiplets contain a CSIP
  - Securely obtains the key to unlock the chiplet, establishes key sharing, encryption, etc
- **Chiplet HSM (CHSM)**
  - implemented in the FPGA will send the unlocking keys to the chiplets using Diffie Hellman key ex change (DHKE) protocol, enables key sharing, encryption, Hash, etc
  - An NVM will store the encrypted bitstream of the CHSM.
  - Unlocking keys are stored inside the NVM accompanying the CHSM.

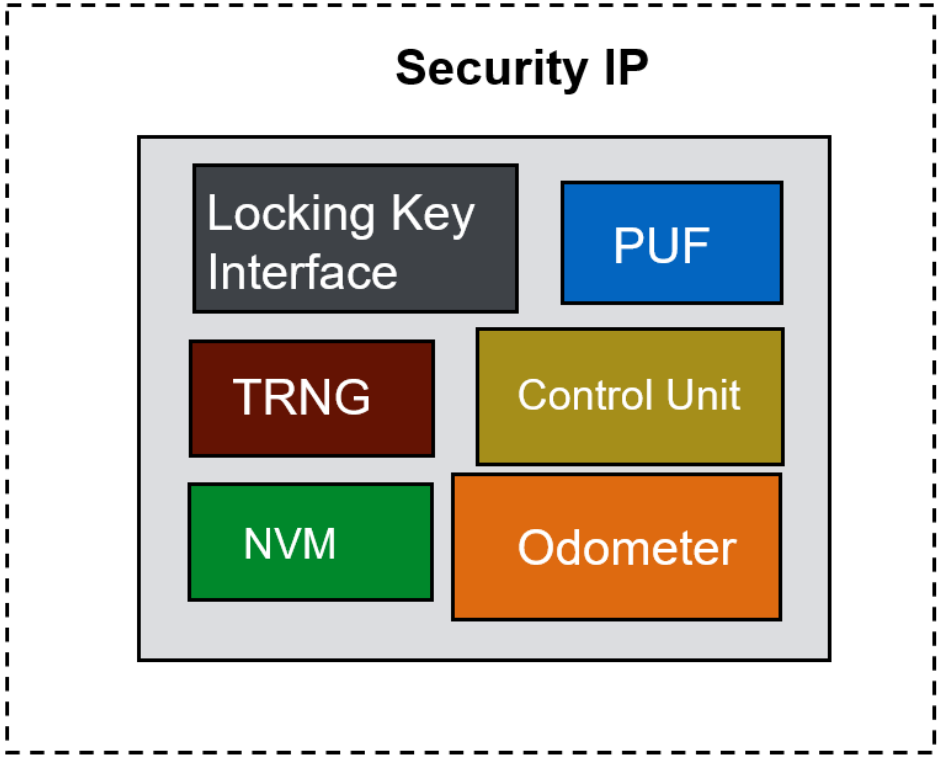
CHSM Design – Similar to SE

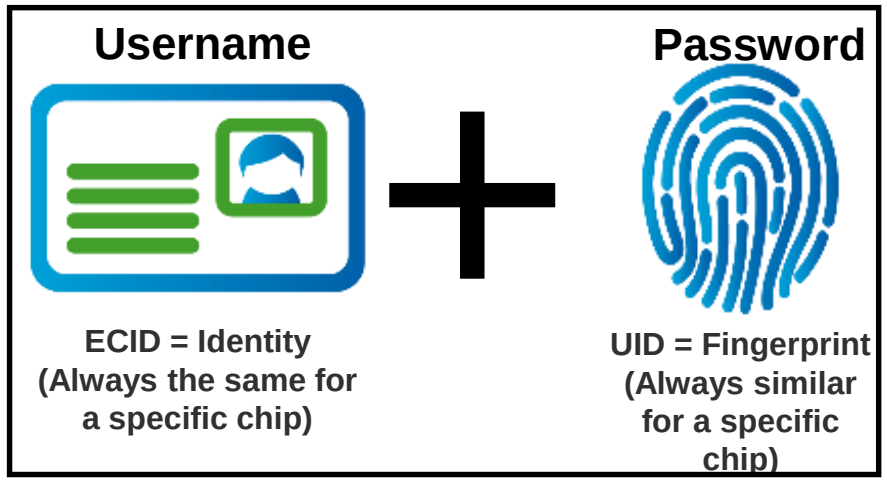


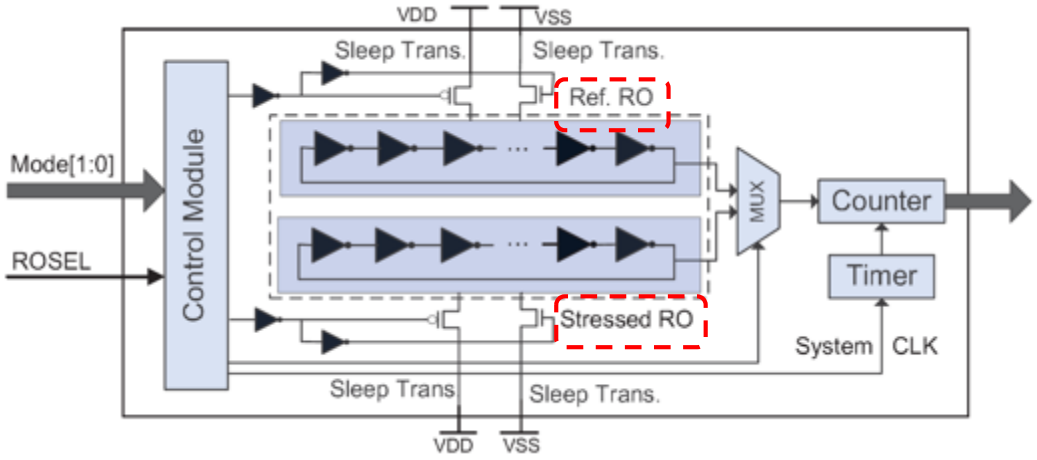
CSIP Design

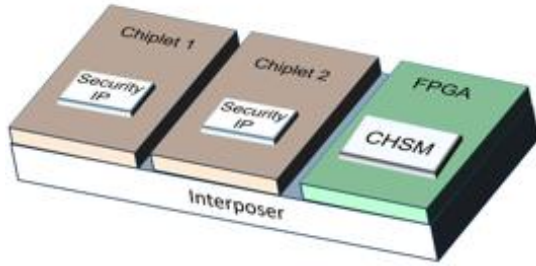
ECE | Florida
Electrical & Computer Engineering

- Chiplet Security IP (CSIP) securely unlocks the locked circuits inside each chiplet.
- Contains security primitives such as PUF, TRNG etc. to perform authentication and key generation.
- Ability to generate public keys and session keys.
- Interface to send and receive data to and from **root of trust**
- Performs **cryptographic** operations.
- Stores **ECID** or unique **chiplet ID** or other forms of identification (**Public or Private**).
- Keep track of the aging of the chip.



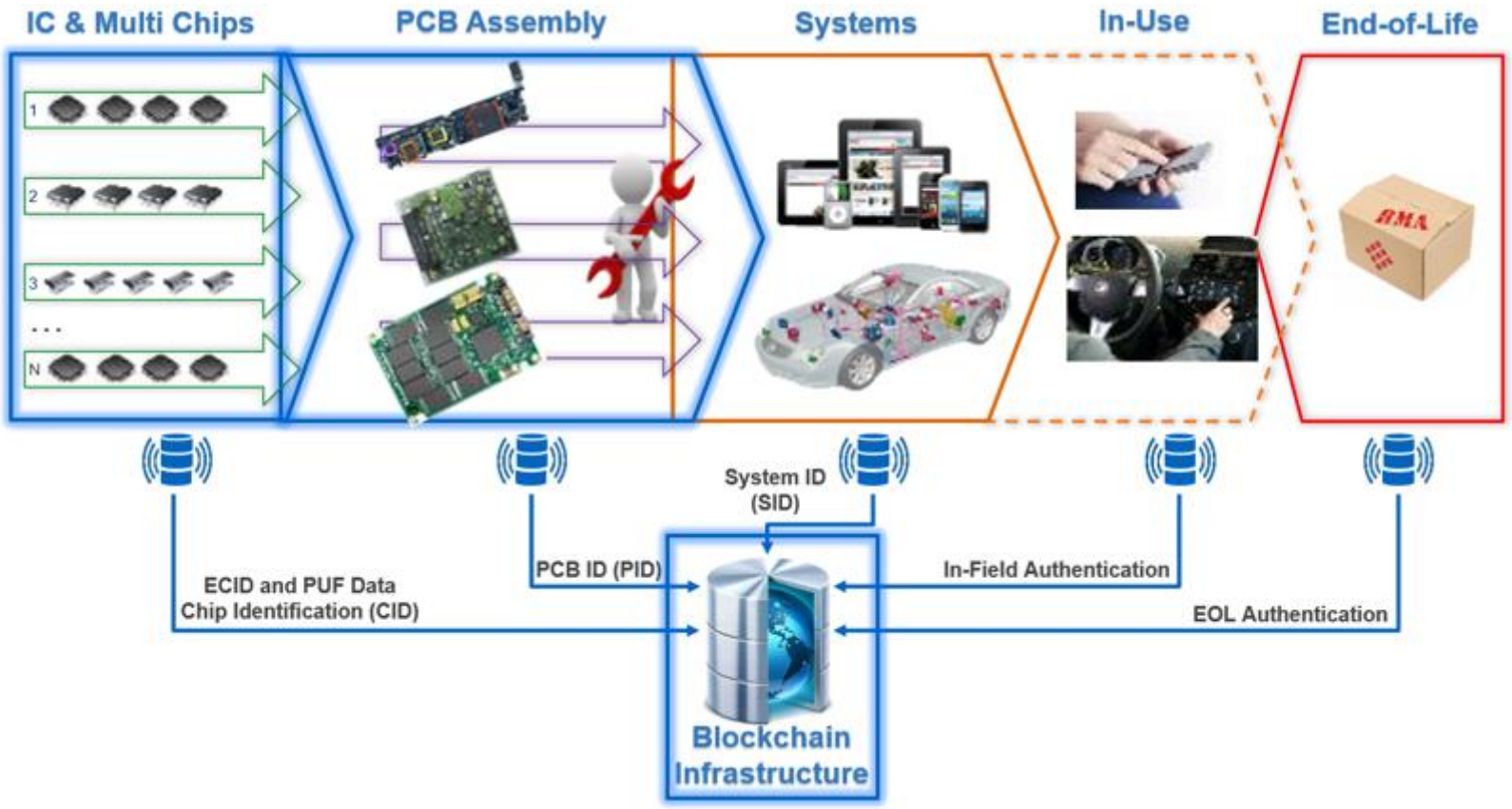


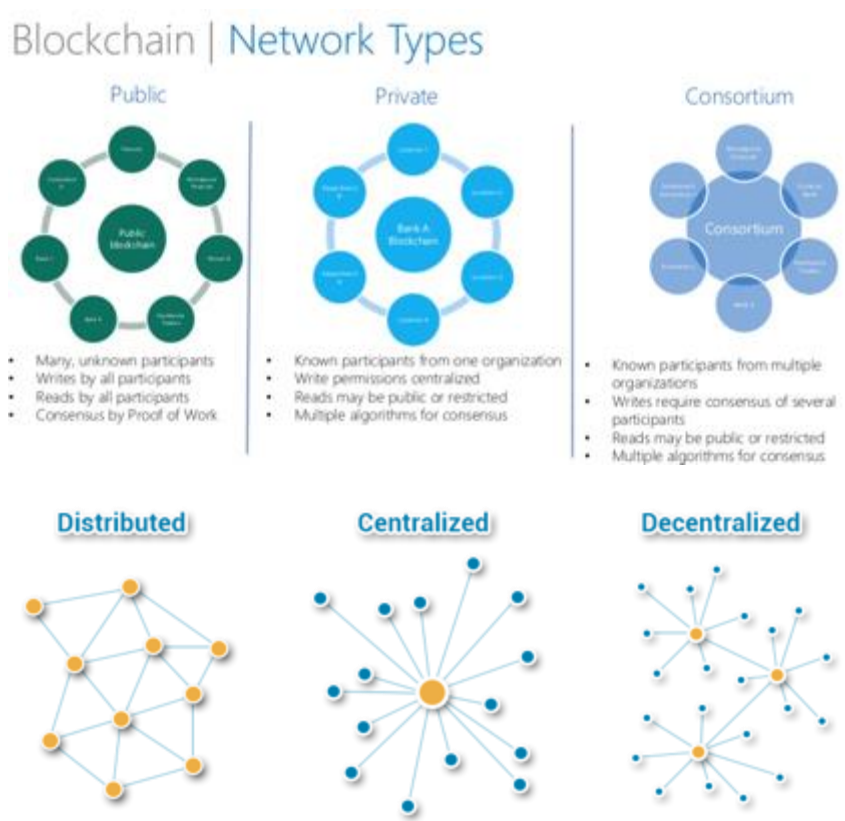




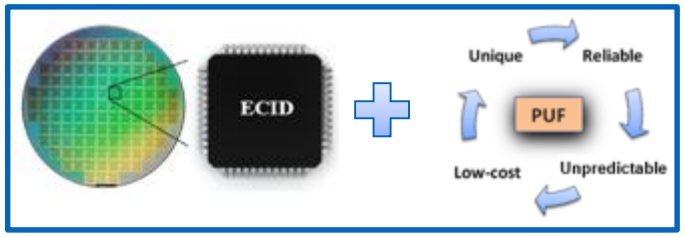
Device-to-System Authentication

ECE | Florida
Electrical & Computer Engineering

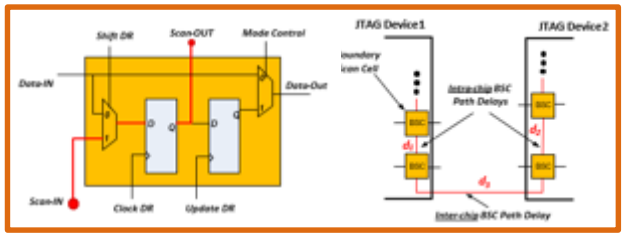




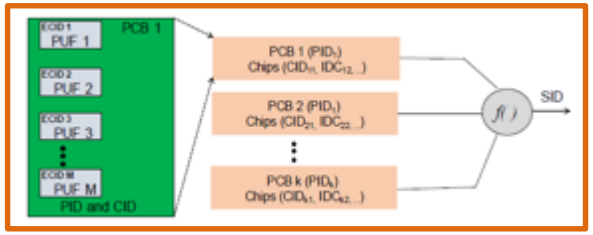
IC Authentication



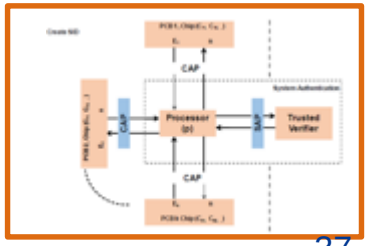
PCB Authentication



Subsystem Authentication



Hardware & Firmware Self Authentication



Secure Communication with the Chiplet Under Test using CSIP

Designer

1. Designer has already put in hooks in the design that can ensure non-functional operation if the correct key is not included in the chip
2. Detecting a non-functional chip is significantly easier than using PUF and dealing with process variations

1. Foundry will not be able to ship any functional chips to the market
2. Same for defective chips and out-of-spec chips; the chips are simply non-functional.

Foundry & Assembly

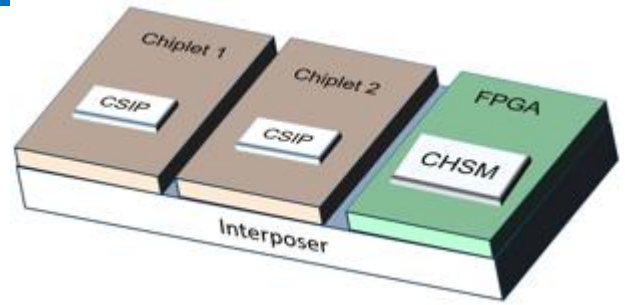
To prevent:

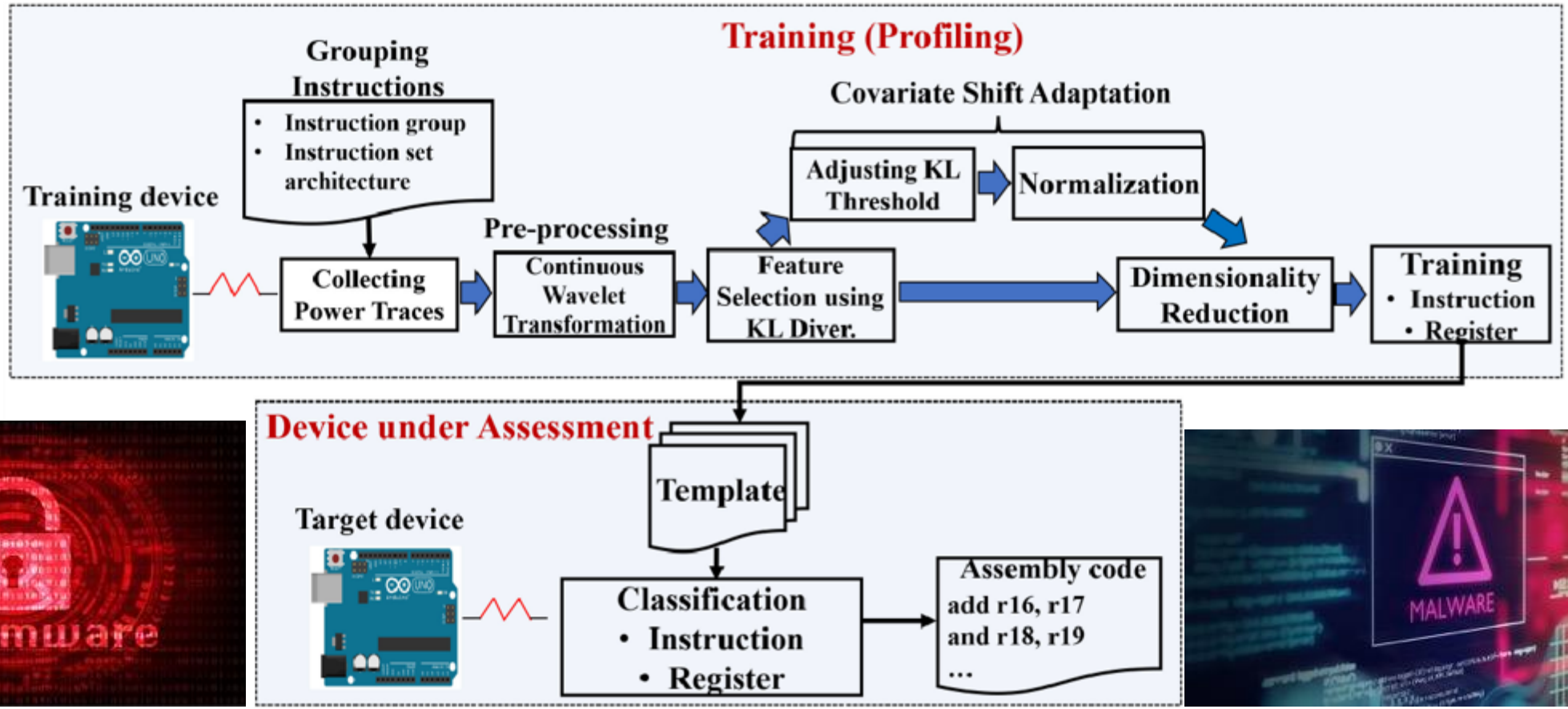
- Over-production
- Out-of-spec
- Defective
- Remarked
- Cloned

G. Contreras et. al., "**Secure Split-Test for preventing IC piracy by untrusted foundry and assembly**," *IEEE International Symposium Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT),* pp.196-203, 2013.
T. Rahman, D. Forte, Q. Shi, G. Contreras, and M. Tehranipoor, "**CSST: Preventing Distribution of Unlicensed and Rejected ICs by Untrusted Foundry and Assembly,"**IEEE Int. Symposium on Defect and Fault Tolerance Symposium (DFTS), Oct. 2014

Runtime Monitoring using CHSM

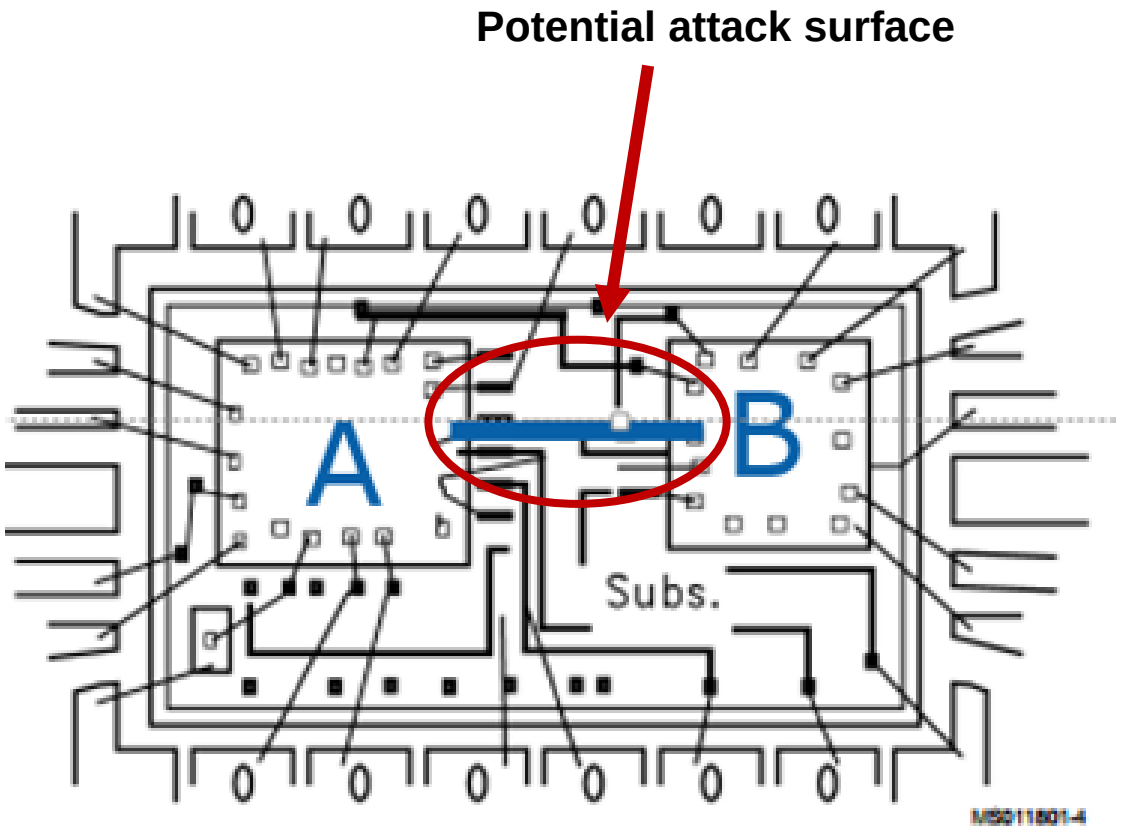
- **Runtime Security and Integrity Checker:** Equip FPGA with sensors to measures and perform side channel analysis
  - Enable detailed program analysis





Physical Assurance

- Signals with confidentiality and integrity requirements should pass between dies using an encryption protocol.
- But not all chiplets are equipped with encryption engine.
- **Approaches:**
  - ✓ Anti-tampering sensors
  - ✓ Active and passive shields
  - ✓ Watermarks on package
  - ✓ PUF based authentication



Conclusions

- **Complexity of modern SoCs and (re)emergence of SiPs**
- **Trust of third party IPs and chiplets remain a major challenge.**
- **Design for security and root of trust must be established at the hardware level.**
- **Security along design cycle requires effective verification solutions**
- **Security along life cycle can be accomplished via security engine (SoCs) or CHSM (SiPs)**

?