Secure Heterogenous Integration: Challenges and Solutions

Farimah Farahmandi
Assistant Professor, Electrical and Computer Engineering, University of Florida
Associate Director, Florida Institute for Cybersecurity Research

Founder, Caspia Technologies

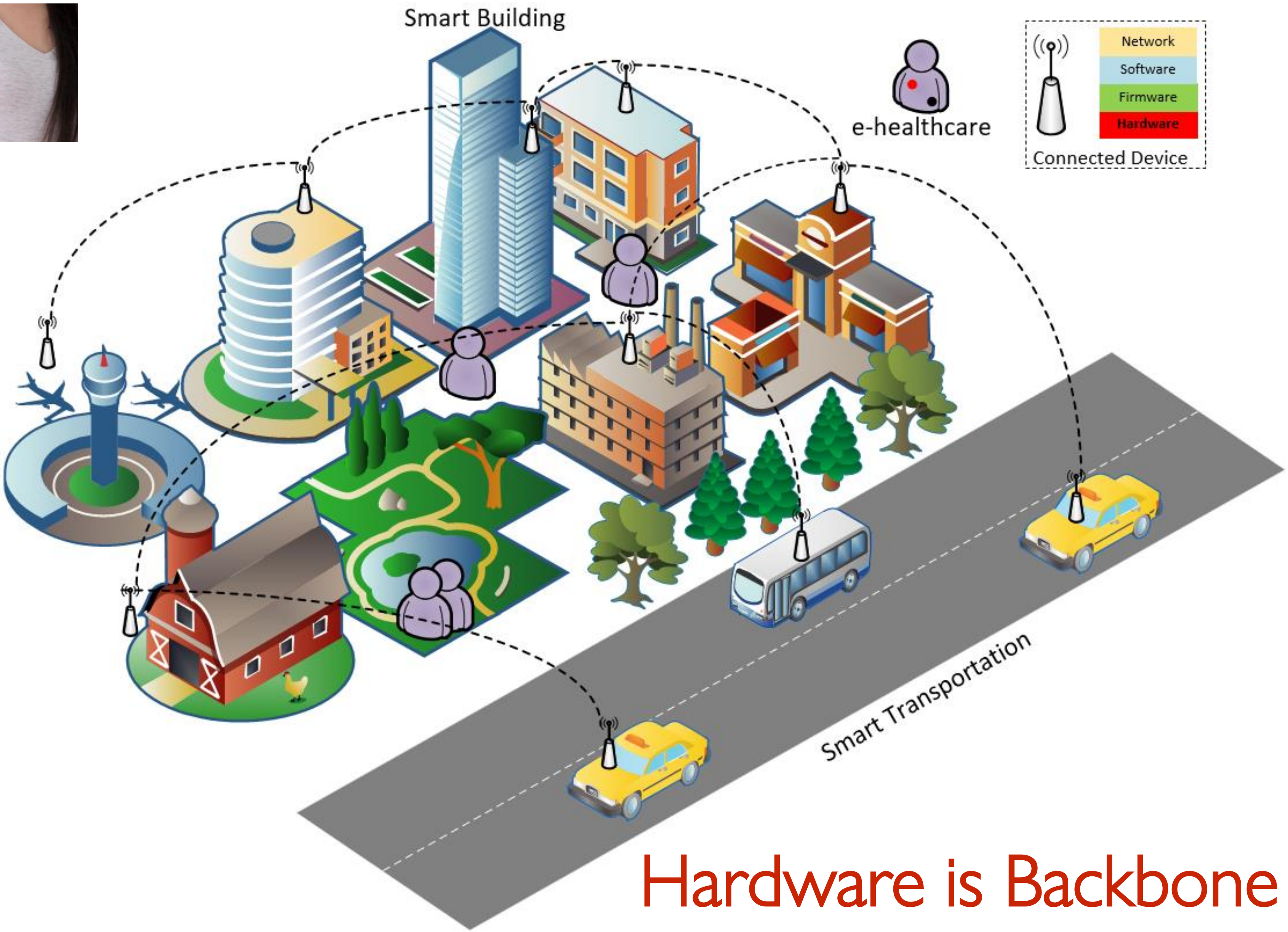
Caspia Technologies

FICS Research | UF UNIVERSITY of FLORIDA

Section 1:
Hardware Security Fundamentals

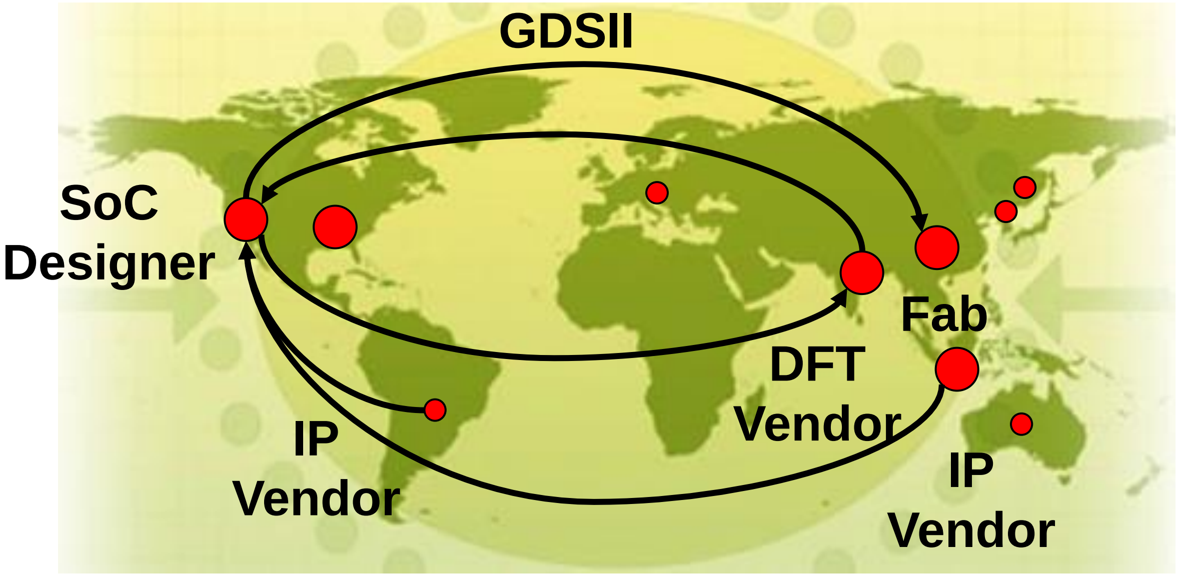
Internet of Thing Devices to Smart Cities

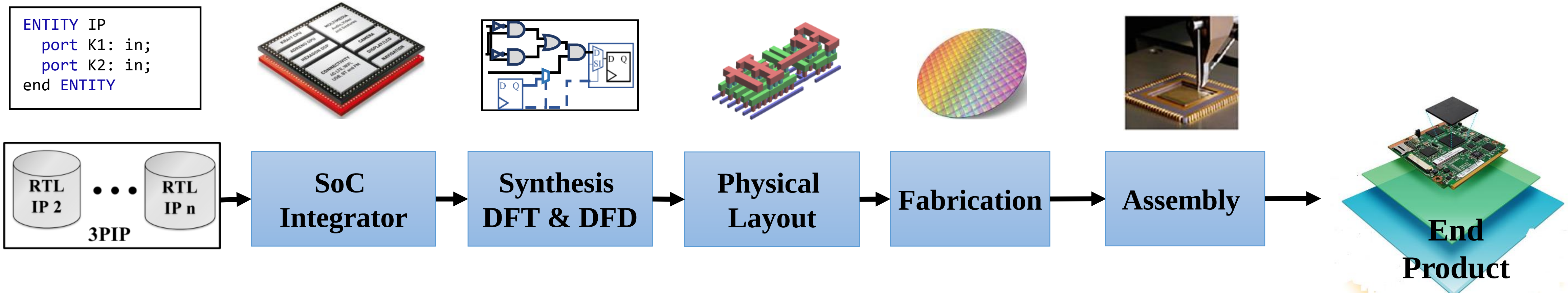
Things
+
Sense & Communication



Hardware is Backbone

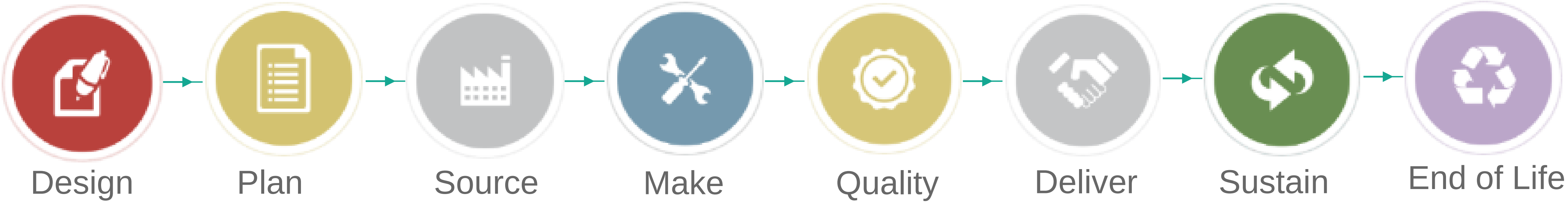
Design Flow





```
ENTITY IP
  port K1: in;
  port K2: in;
end ENTITY
```

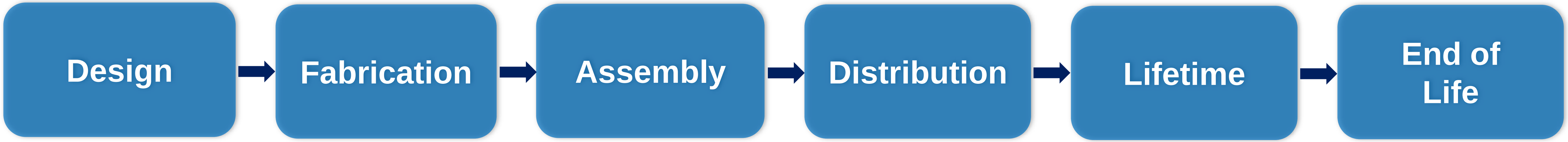
Supply Chain



Design → Plan → Source → Make → Quality → Deliver → Sustain → End of Life

And…

The Electronics Supply Chain Within It



Design → Fabrication → Assembly → Distribution → Lifetime → End of Life

Entities Involved in the Supply Chain

Open Source Software

Software Licensors

HW Component Suppliers

Cloud Service Providers

Logistics Partners

OEMs/ODMs

IOT Devices

Manufacturing Partners

Channel/Distributors

Repair /Refurbishment Partners

Scrap Partners

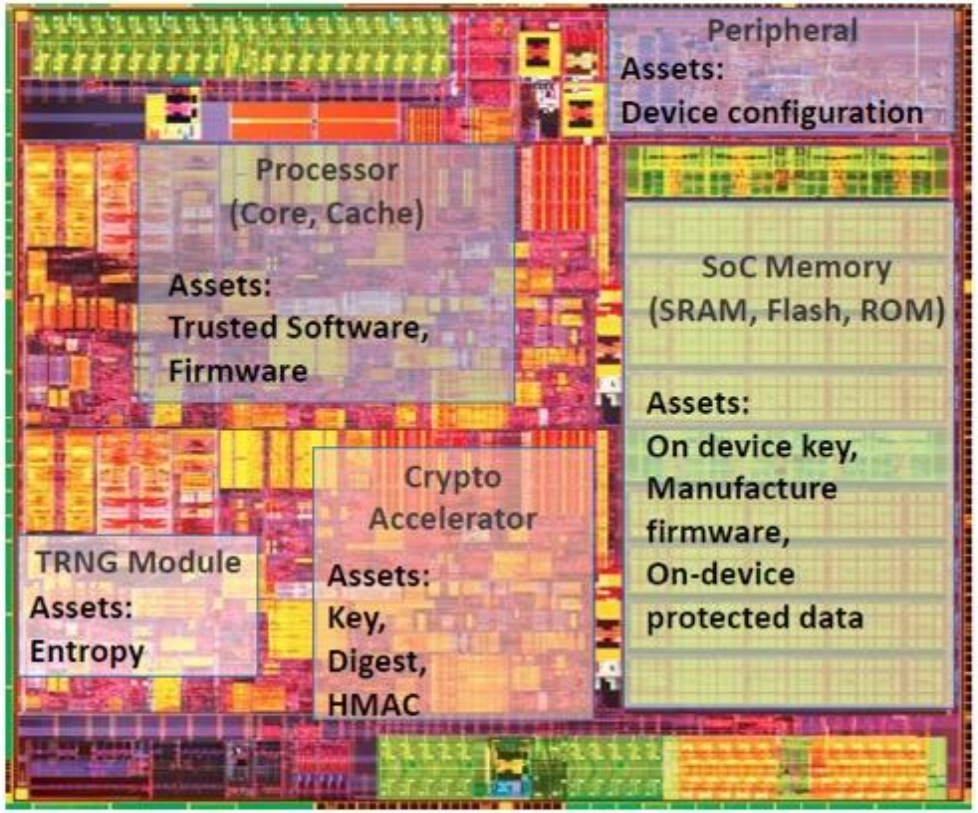
Recycling Partners

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
- Biometrics
- E-fuse
- And more…



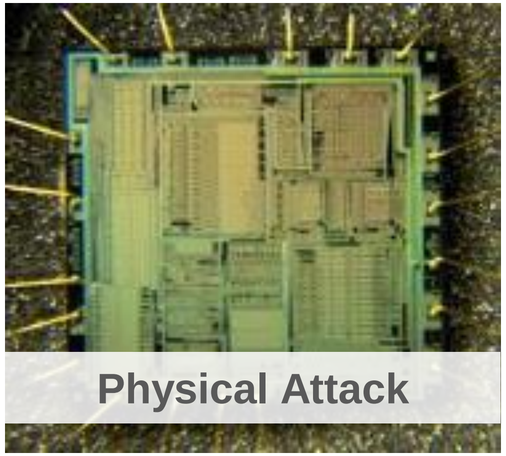
Source: Intel

More Attacks on Hardware

Trojans

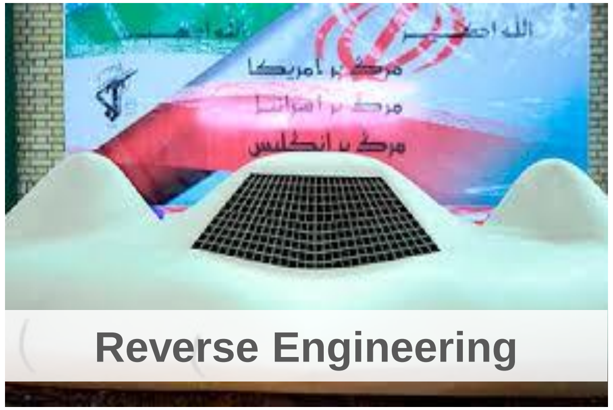
Untrusted Foundry

Counterfeit ICs


Physical Attack

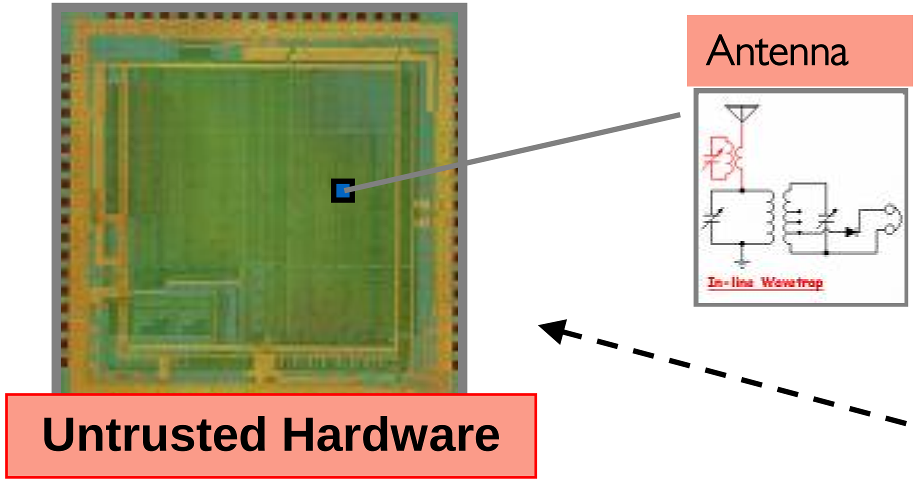
Side-channel

Fault Injection


Reverse Engineering

Fake Parts

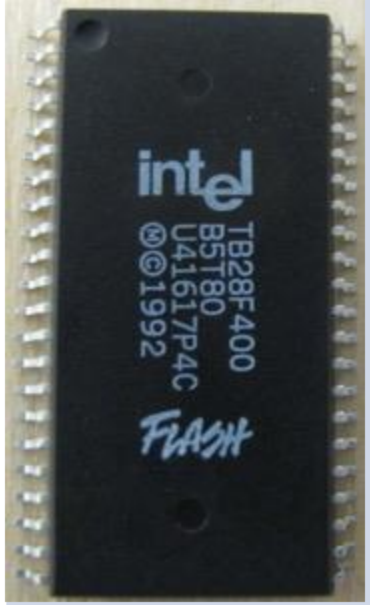
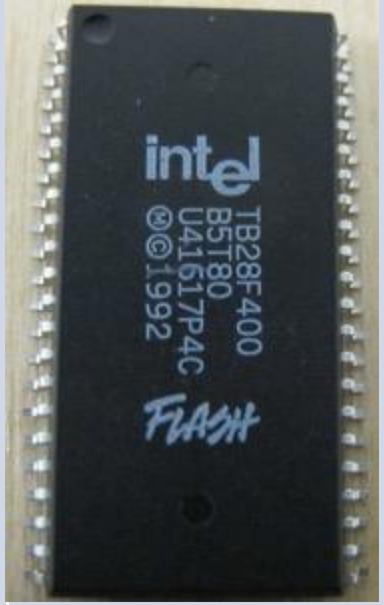
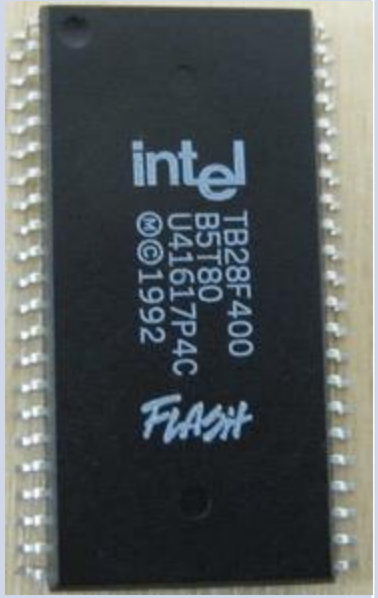
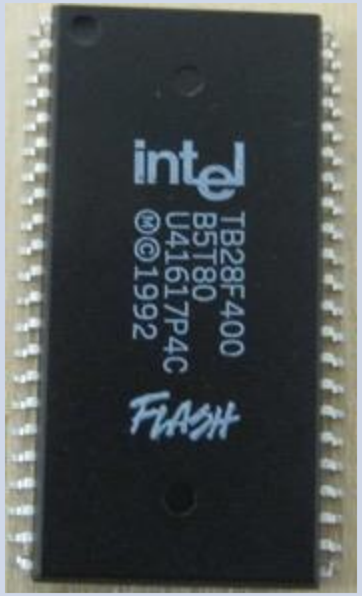
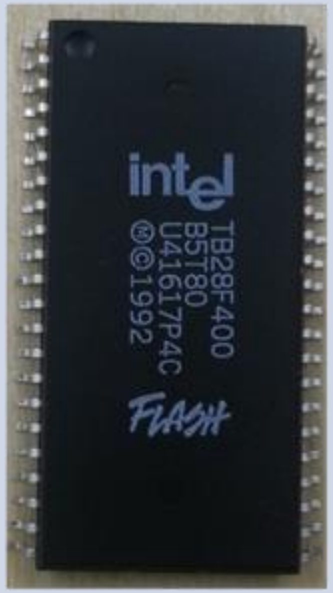
Hardware Trojan



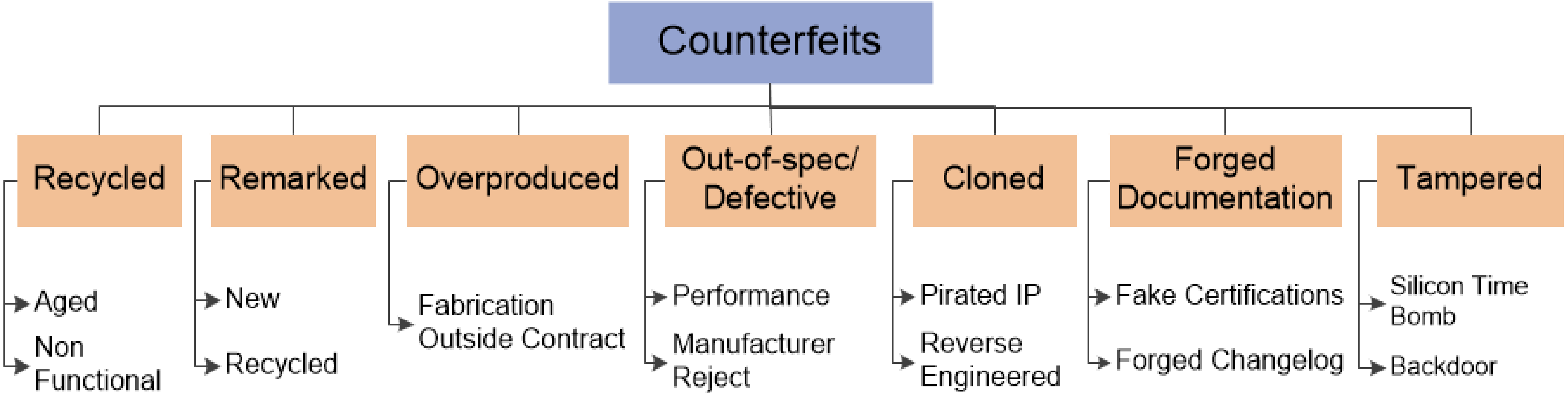
- ➢ Adversary can send and receive secret information.
- ➢ Adversary can disable the chip, send wrong processing data, impact circuit information, steal sensitive info, etc.

- ➢ Adversary can place an Antenna on the fabricated chip.
- ➢ Such Trojans cannot be detected since it does not change the functionality of the circuit.

Counterfeit Parts

Sample 1	Sample 2	Sample 3	Sample 4	Sample 5
				

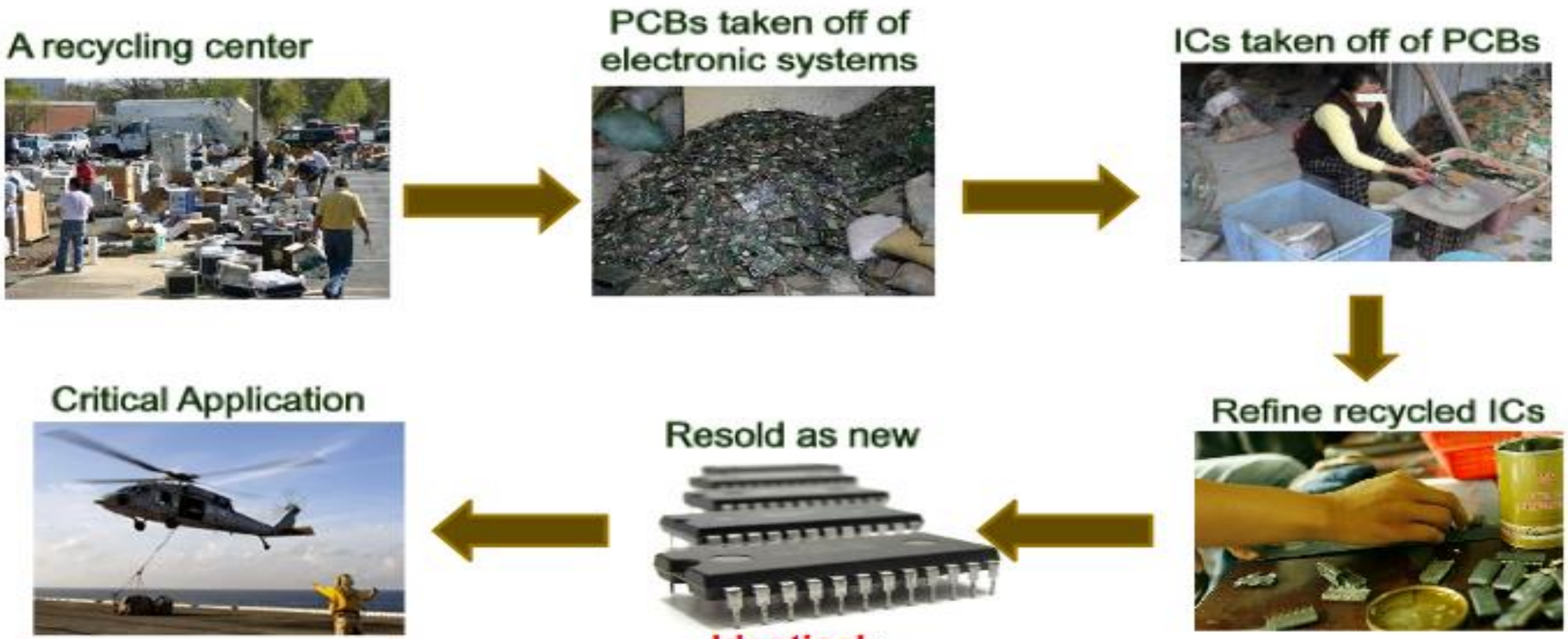
Counterfeit Parts



- **Recycled** and **remarked** types contribute to majority of counterfeit incidents.
- Untrusted foundry/assembly can introduce **overproduced** and **out-of-spec/defective** parts
- **Cloning** can be done by a wide variety of adversaries (a small entity to a large corporation)
- **Tampered** parts act as a backdoor where secret information from the chip or sabotage system functionality

U. Guin, D. DiMase, and M. Tehranipoor, "A Counterfeit Integrated Circuits: Detection, Avoidance, and the Challenges Ahead," Journal of Electronic Testing: Theory and Applications (JETTA), 2014.
U. Guin, D. DiMase, and M. Tehranipoor, "A Comprehensive Framework for Counterfeit Defect Coverage Analysis and Detection Assessment," Journal of Electronic Testing: Theory and Applications (JETTA), 2014.

Recycling Process



A recycling center

PCBs taken off of electronic systems

ICs taken off of PCBs

Refine recycled ICs

Resold as new

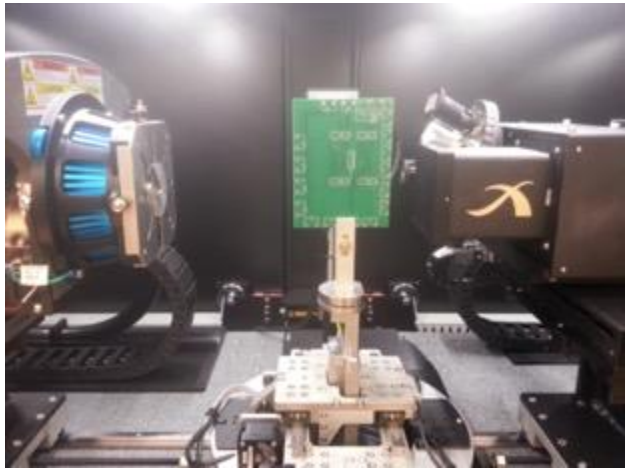
Critical Application

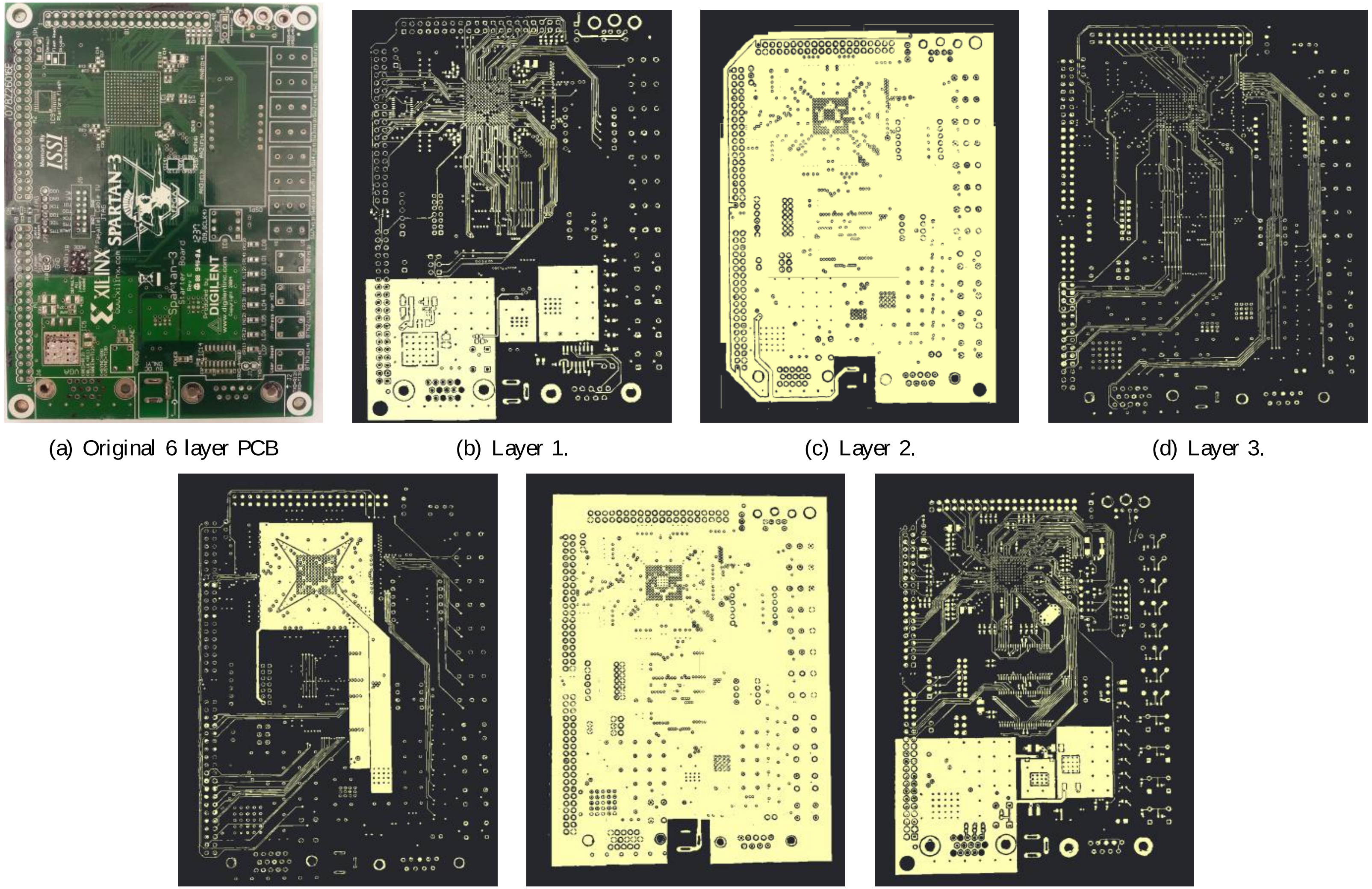
Identical:
Appearance, Function, Specification

Consumer trends suggest that more gadgets are used in much shorter time – more e-waste

Source: Images are taken from google

Reverse Engineering




(a) Original 6 layer PCB

(b) Layer 1.

(c) Layer 2.

(d) Layer 3.

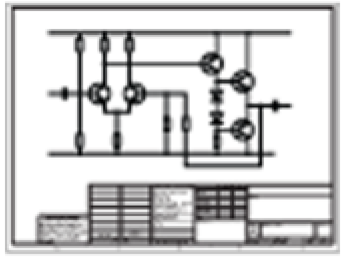
(e) Layer 4.

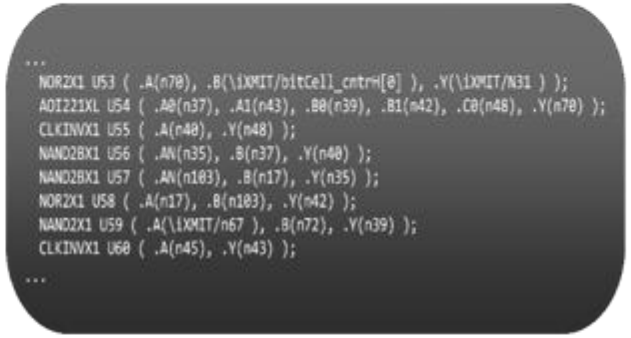
(f) Layer 5.

(g) Layer 6.

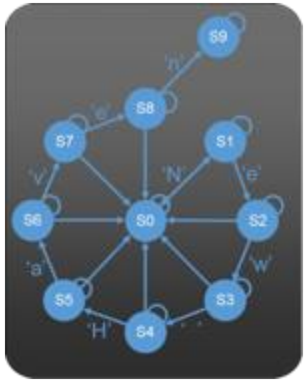
Reverse Engineering

Delayering and Imaging: Netlist recovery
Netlist Reverse Engineering: Function recovery





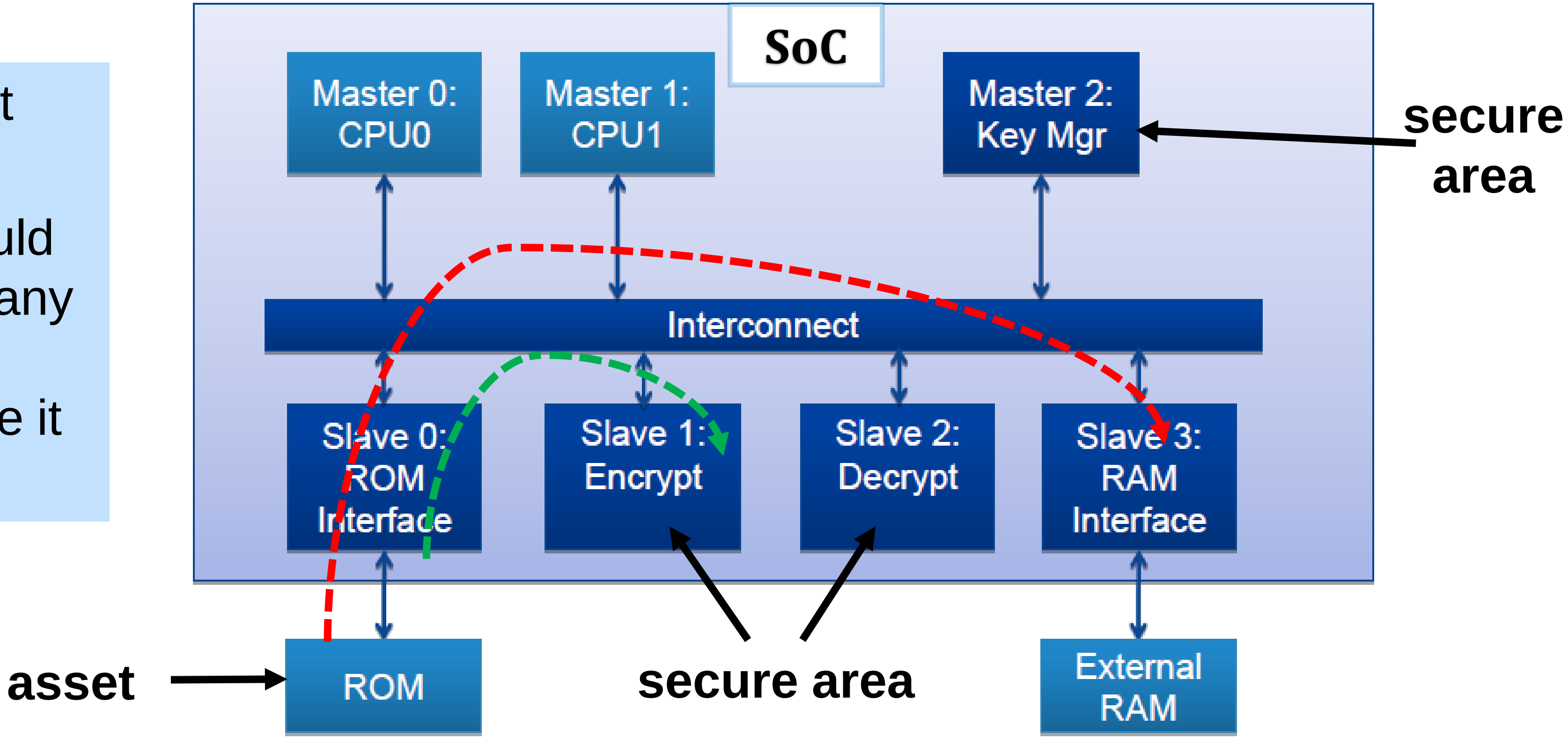
```
...
  NOR2X1 U53 ( .A(n70), .B(\iXMIT/bitCell_cntrH[0] ), .Y(\iXMIT/N31 ) );
  AOI221XL U54 ( .A0(n37), .A1(n43), .B0(n39), .B1(n42), .C0(n48), .Y(n70) );
  CLKINVX1 U55 ( .A(n40), .Y(n48) );
  NAND2BX1 U56 ( .AN(n35), .B(n37), .Y(n40) );
  NAND2BX1 U57 ( .AN(n103), .B(n17), .Y(n35) );
  NOR2X1 U58 ( .A(n17), .B(n103), .Y(n42) );
  NAND2X1 U59 ( .A(\iXMIT/n67 ), .B(n72), .Y(n39) );
  CLKINVX1 U60 ( .A(n45), .Y(n43) );
...
```



Information Leakage

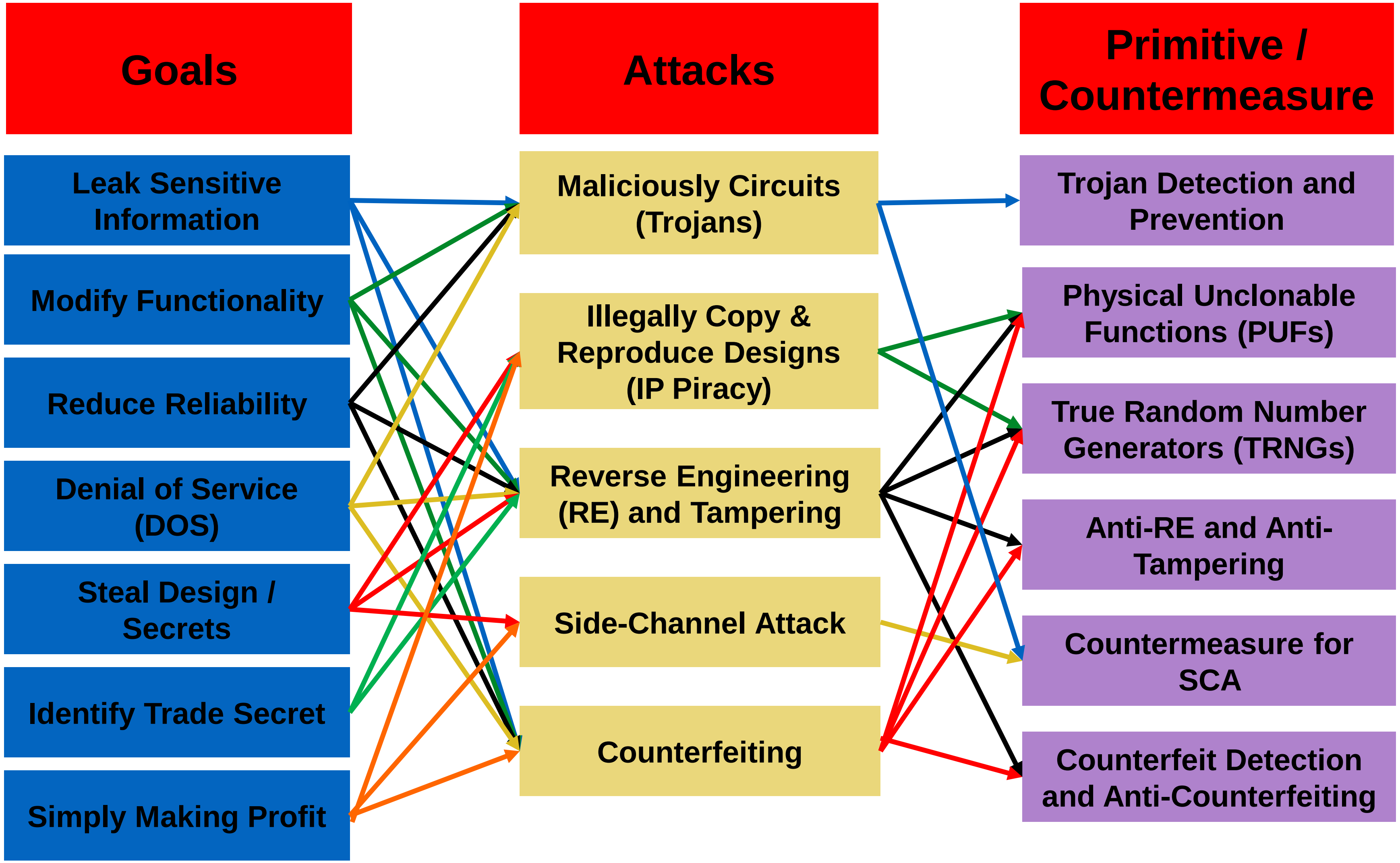
- **Validation of Deterministic Security Requirements**
  - Deterministic security requirements → can be directly derived from security policies
  - Includes **access control** restrictions, address translations, and more

- **Vulnerability:** Asset leakage
- **Rule:** An asset should never propagate to any location where an attacker can observe it



Source: Jasper

Security Goals and Attack Vectors



Goals

- Leak Sensitive Information
- Modify Functionality
- Reduce Reliability
- Denial of Service (DOS)
- Steal Design / Secrets
- Identify Trade Secret
- Simply Making Profit

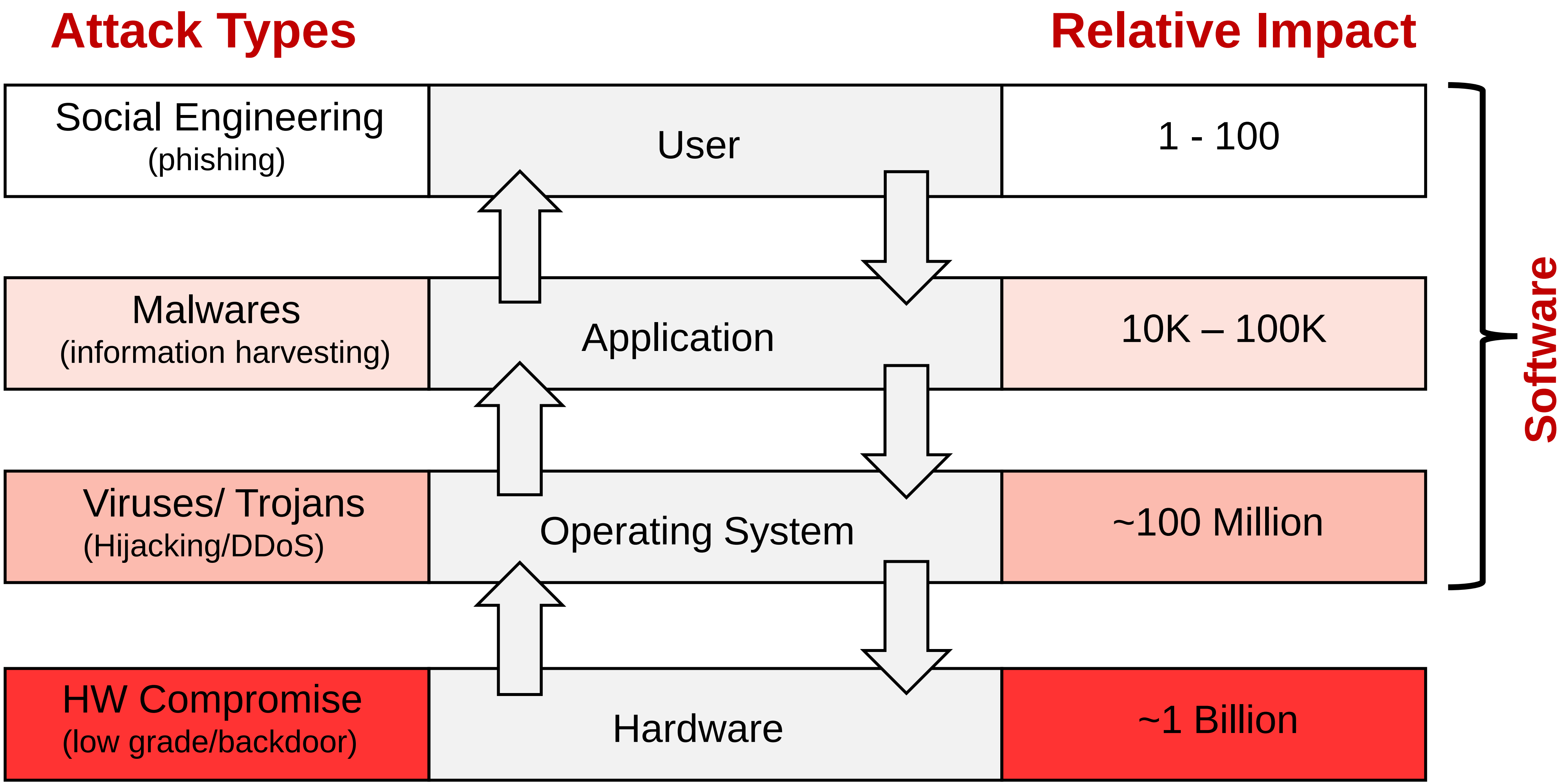
Attacks

- Maliciously Circuits (Trojans)
- Illegally Copy & Reproduce Designs (IP Piracy)
- Reverse Engineering (RE) and Tampering
- Side-Channel Attack
- Counterfeiting

Primitive / Countermeasure

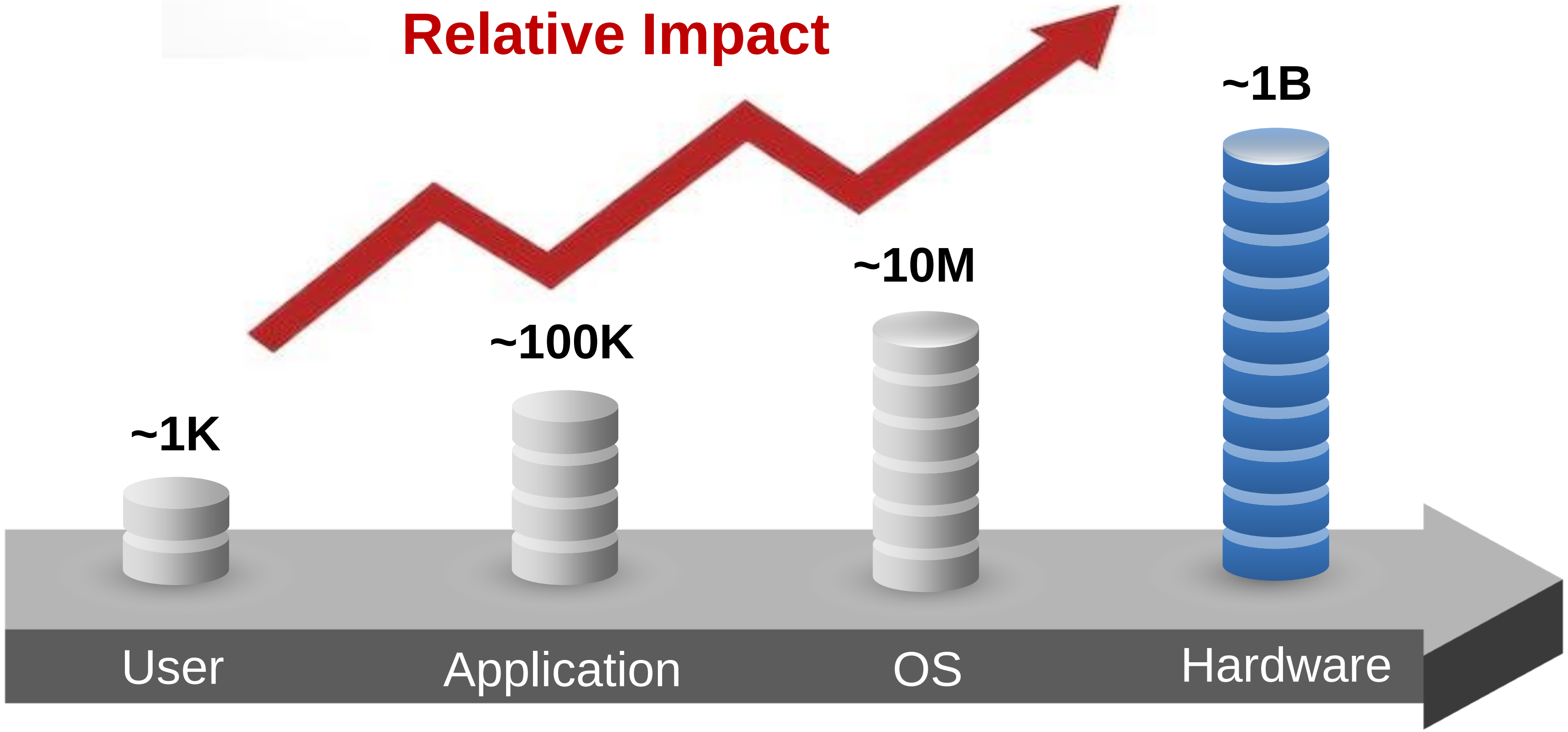
- Trojan Detection and Prevention
- Physical Unclonable Functions (PUFs)
- True Random Number Generators (TRNGs)
- Anti-RE and Anti-Tampering
- Countermeasure for SCA
- Counterfeit Detection and Anti-Counterfeiting

Impact of Hardware Compromise



Attack Types		Relative Impact
Social Engineering (phishing)	User	1 - 100
Malwares (information harvesting)	Application	10K – 100K
Viruses/ Trojans (Hijacking/DDoS)	Operating System	~100 Million
HW Compromise (low grade/backdoor)	Hardware	~1 Billion

Impact: HW Security Compromise

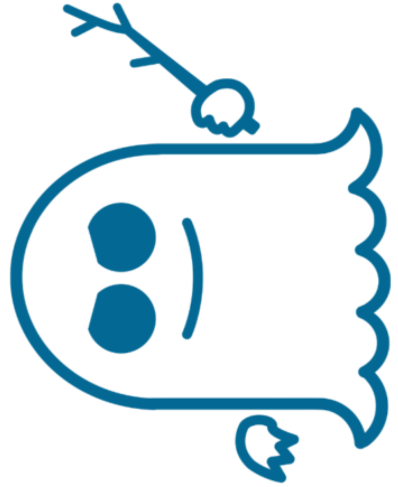


Social engineering (phishing)

Malwares (information harvesting)

Virus/ Trojan (Hijacking/ DDoS)

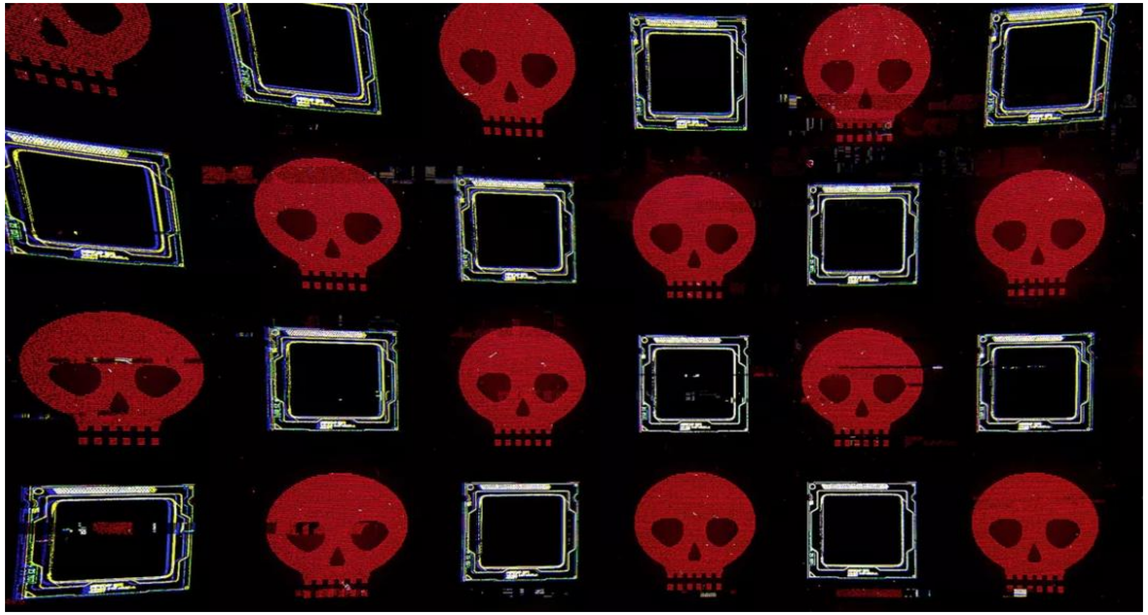
Hardware compromise (low grade/ backdoor)



Impact of Hardware Compromise

THE VERGE

Intel Facing 32 Lawsuits Over Meltdown and Spectre CPU Security Flaws



Jan 4, 2018

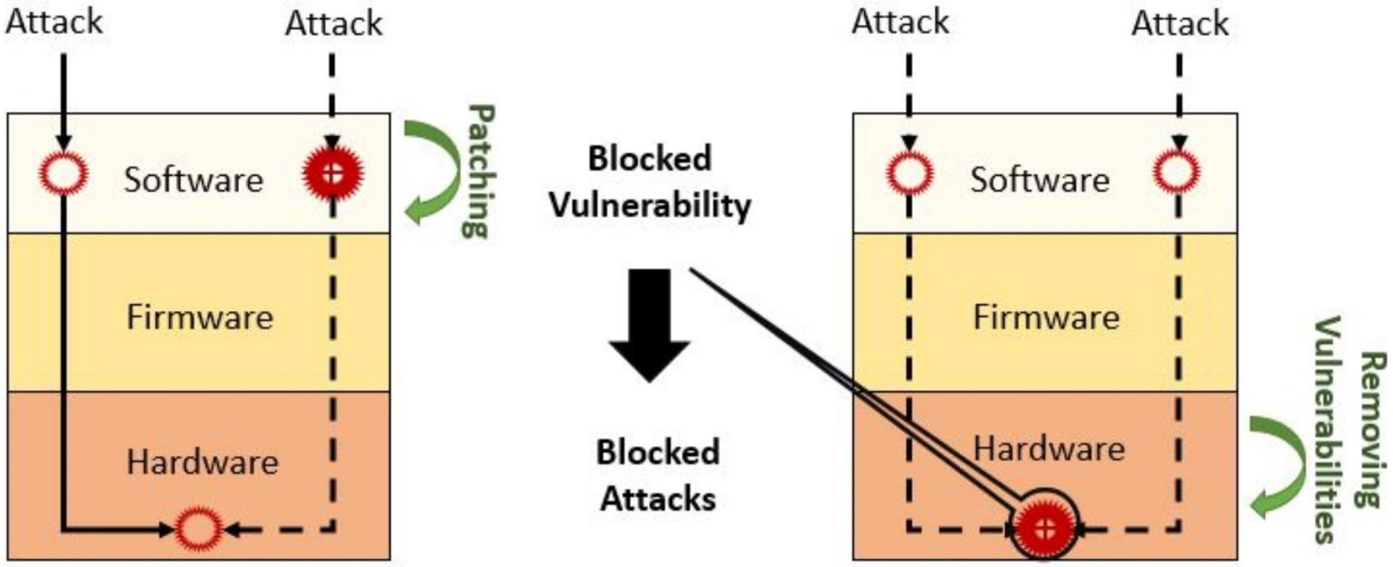
Intel sells off for a second day as massive security exploit shakes the stock

BUSINESS INSIDER

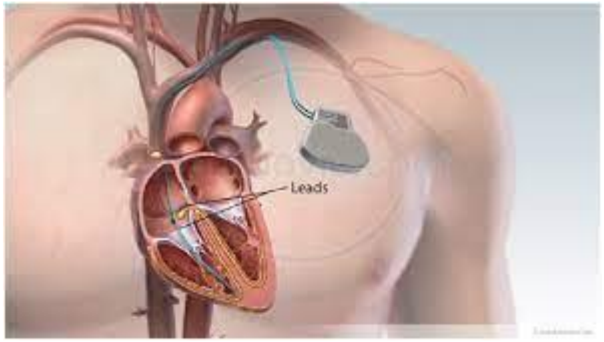
The company accused of selling Apple and Amazon data servers compromised by Chinese spies is getting crushed — it's lost half of its value today

Importance of Removing Hardware Vulnerabilities

- Removing hardware-level vulnerabilities will reduce system vulnerabilities by 43%



- If they remain undetected
  - Harm company’s reputation
  - Threaten user privacy
  - Endangers people’s life



Malfunctions in pacemakers will lead to patients’ death

Current Practices

- **Manual Security Assessment**
- **Certification Schemes**: Security verification by an independent official 3rd party
  - Example: payment Card Industry (PCI-DSS and PTS Finance industry)
- **Process overview:**
  -

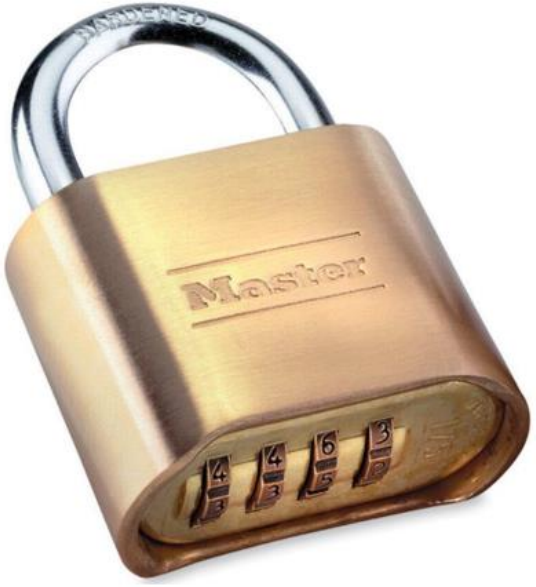
Security claims

3P Assessment

Final report

- **Suffer from various flaws**
  - Security review depends greatly on the experience
  - No proof that the design is secure against possible attack scenarios

Security Issues Detection and Prevention



Inspection

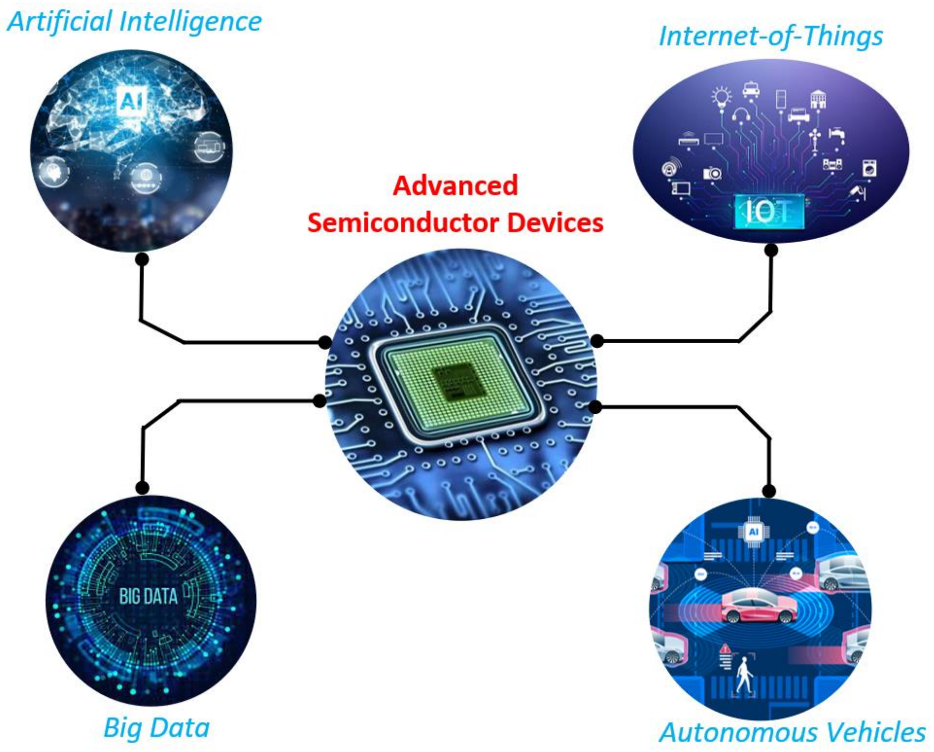
Protection

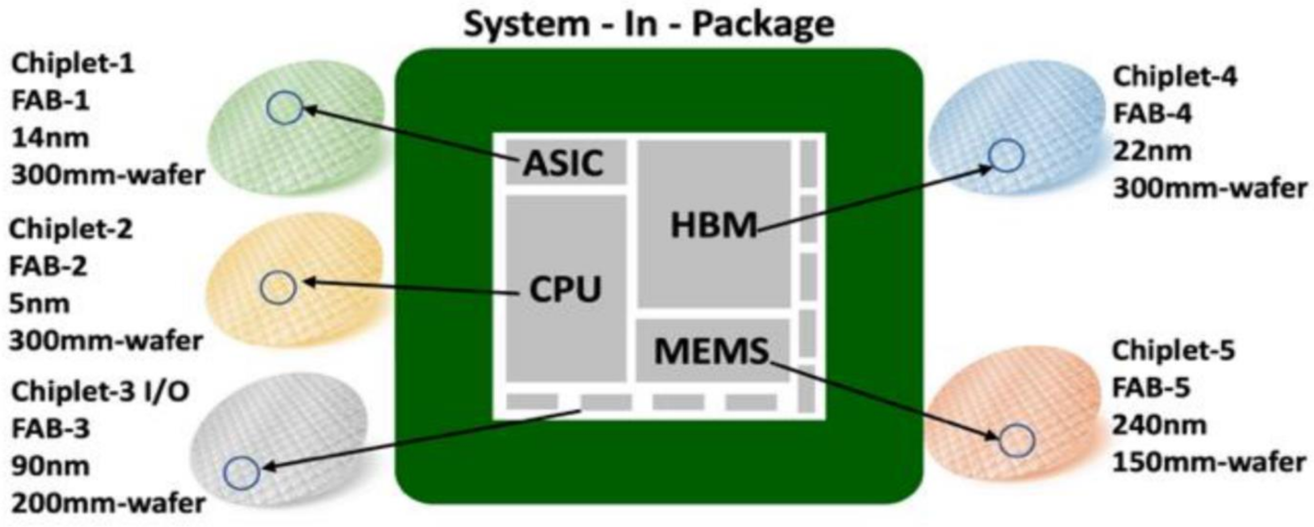
Automation

Section 2:
Secure Advanced Packaging

Heterogeneous Integration

- IoT, big data, and AI inspire unprecedent needs of powerful semiconductor
- Moore’s law is diming - ‘free lunch’ of technology node scaling is over
- *Beyond Moore* technology: **Heterogeneous Integration (HI)**
  - Fulfilling diversity of technology nodes, functionality, and materials





Motivation & Problem Statement

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

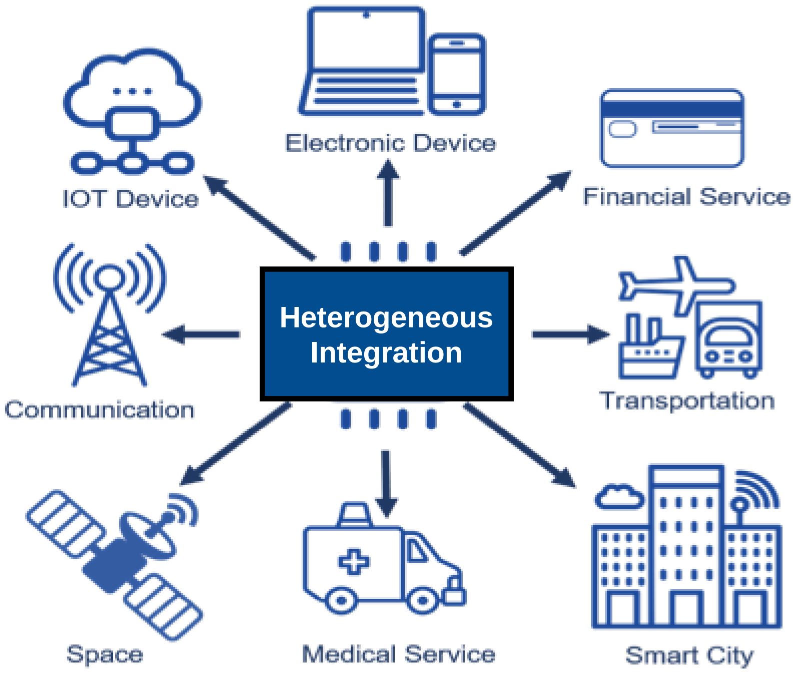
Impact

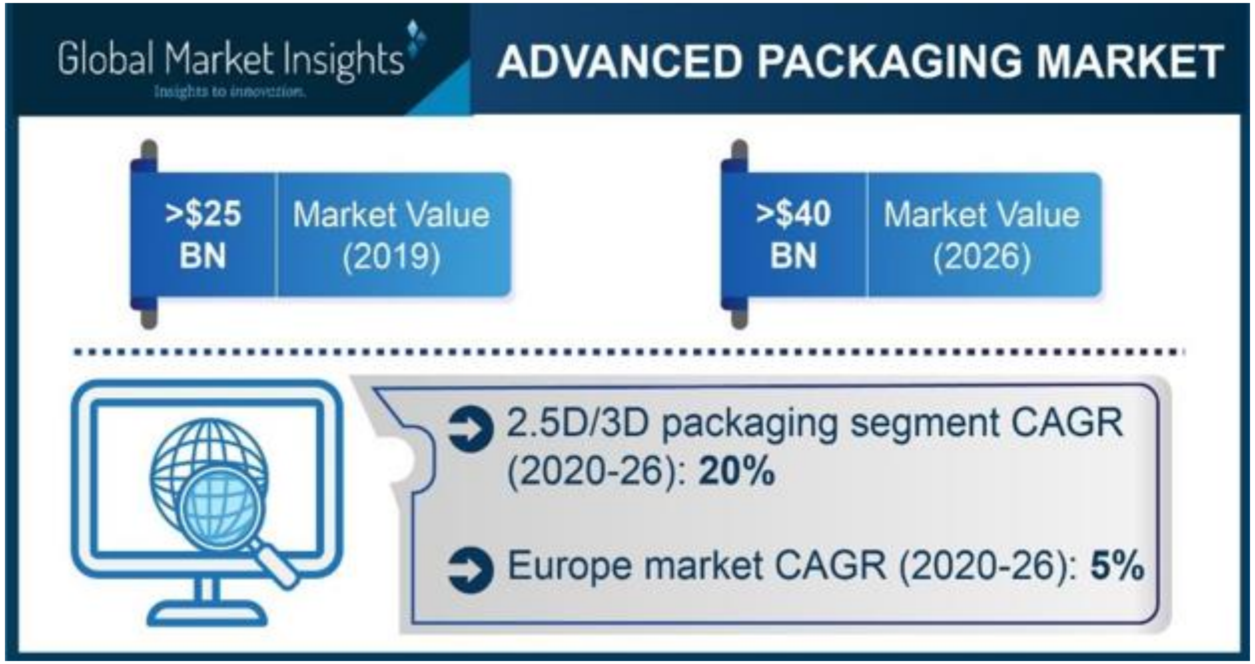
- Flexibility
- Scaling can continue
- Accessibility
- Reuse of expensive IPs
- Cost-efficient

Challenges

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

Heterogeneous Integration





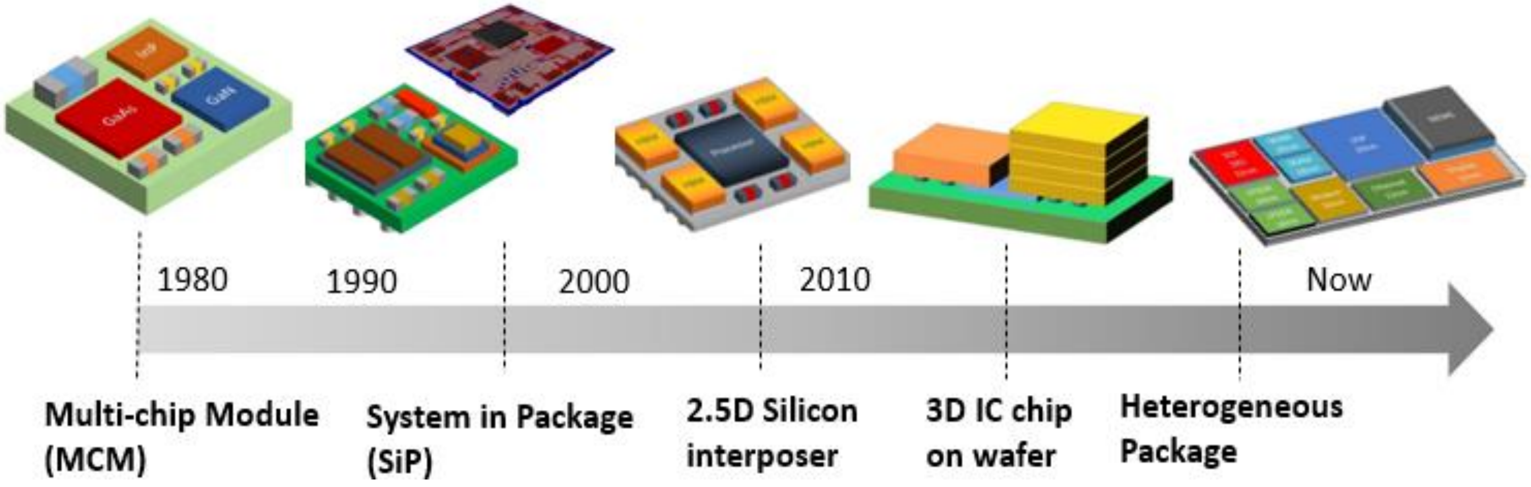
Global Market Insights
Insights to innovation.

ADVANCED PACKAGING MARKET

>$25 BN Market Value (2019)

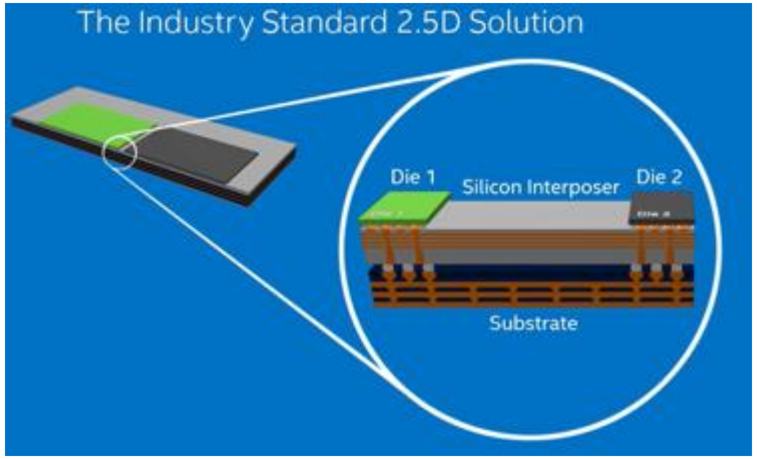
>$40 BN Market Value (2026)

- 2.5D/3D packaging segment CAGR (2020-26): **20%**
- Europe market CAGR (2020-26): **5%**

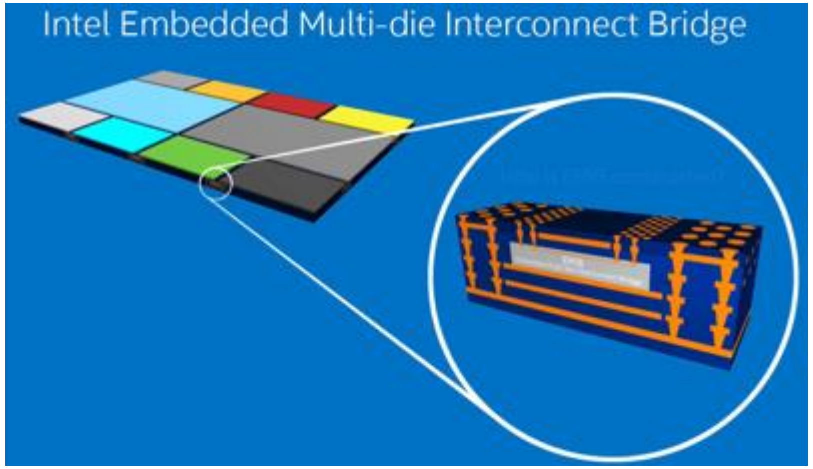


Heterogeneous Integration

Industry Heterogeneous Packaging

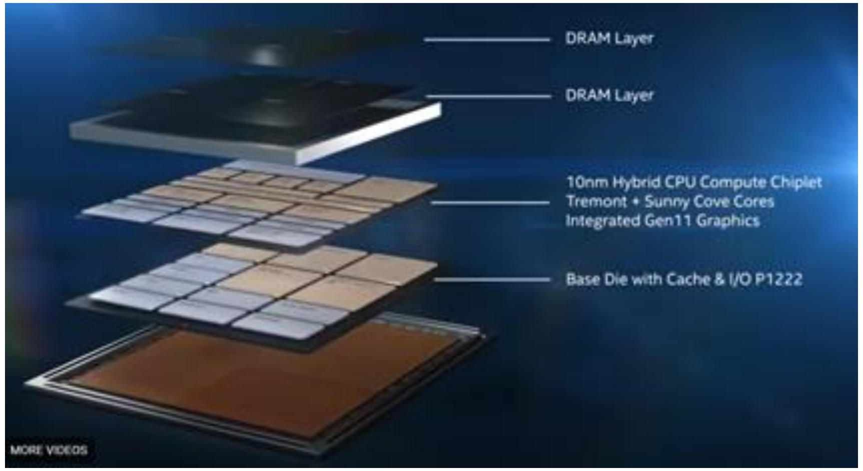


Intel Embedded Multi-die Interconnect Bridge EMIB (passive & active)



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

Intel Foveros 3D Stacking Technology

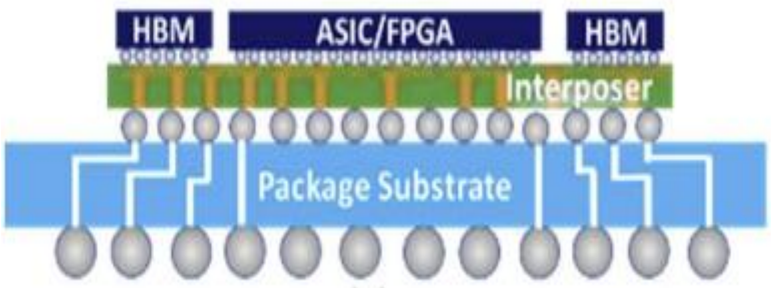

DRAM Layer
DRAM Layer
10nm Hybrid CPU Compute Chiplet
Tremont + Sunny Cove Cores
Integrated Gen11 Graphics
Base Die with Cache & I/O P1222
MORE VIDEOS

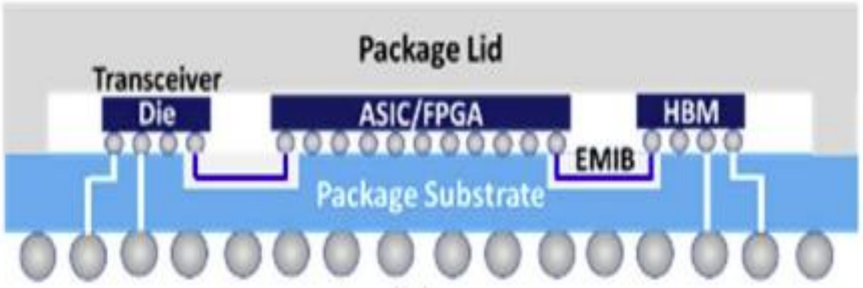
Source: https://newsroom.intel.com/press-kits/lakefield/#gs.rdd753

Courtesy: Intel

Co-design of Chiplets

- Design flow must consider chip-package co-design
- Common implementations of heterogeneous integration: Interposers, EMIBs



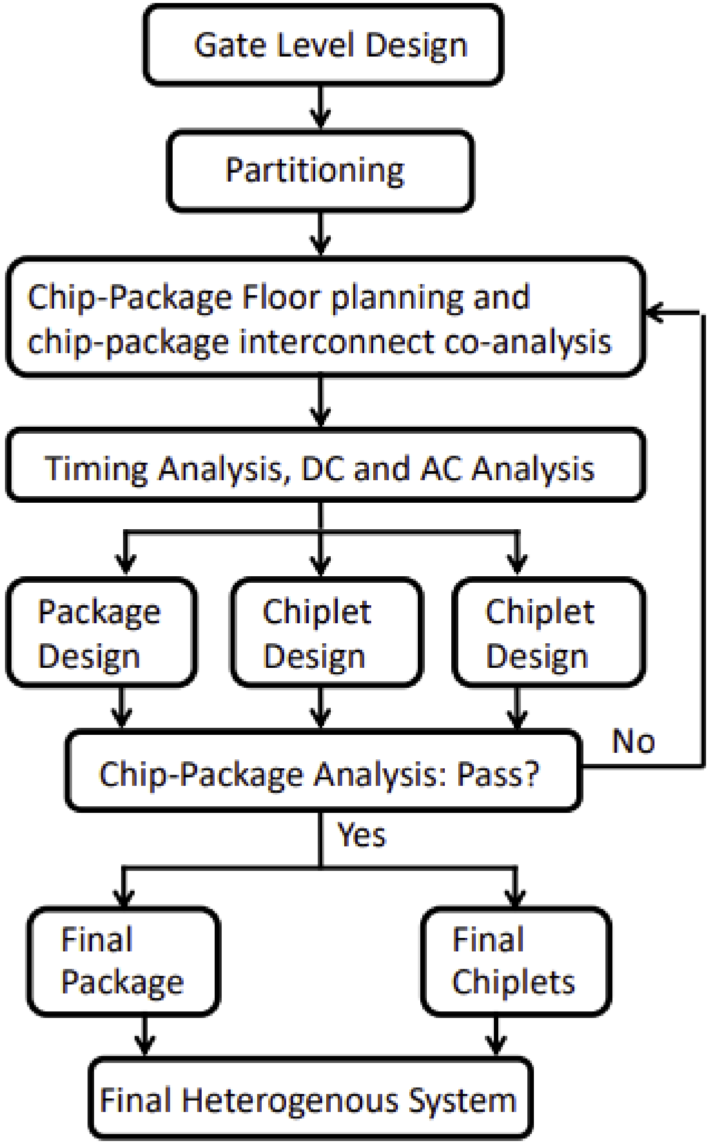


Interposer

XILINX AMD

EMIB

intel

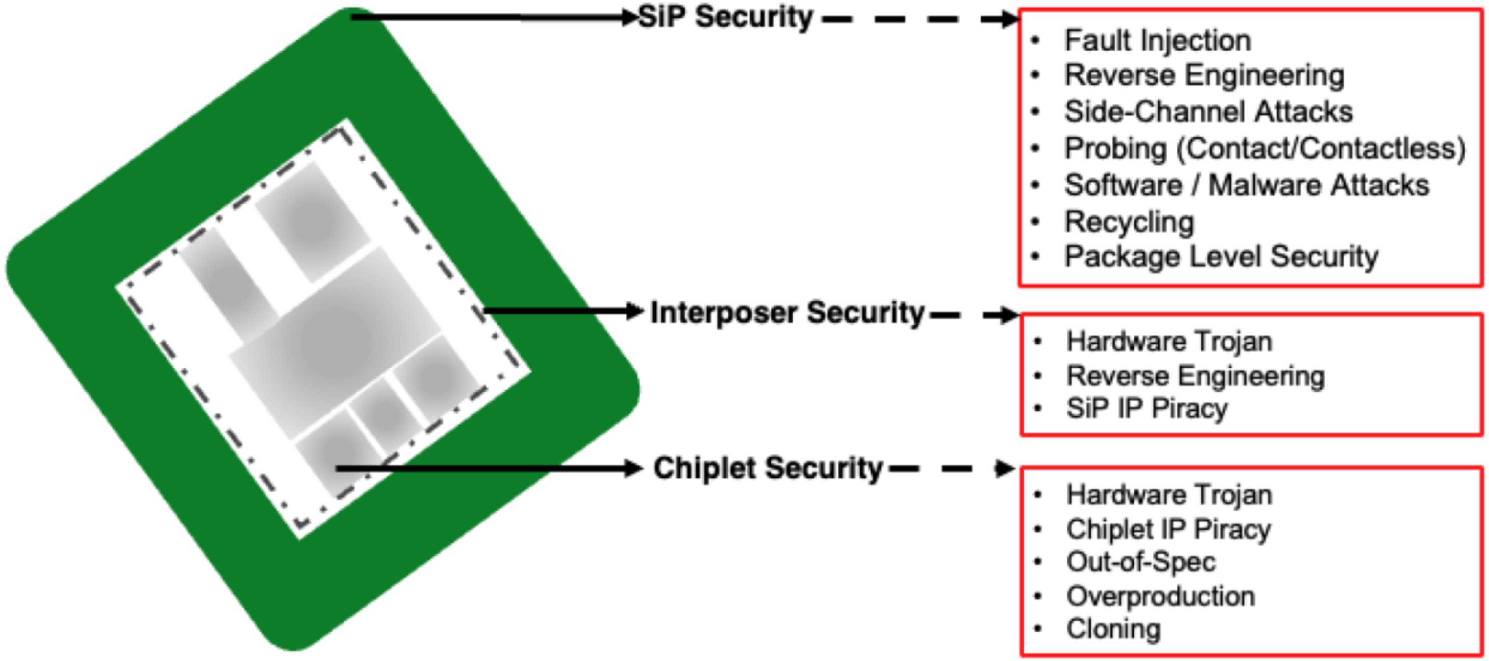

Gate Level Design
Partitioning
Chip-Package Floor planning and chip-package interconnect co-analysis
Timing Analysis, DC and AC Analysis
Package Design
Chiplet Design
Chiplet Design
Chip-Package Analysis: Pass?
No
Yes
Final Package
Final Chiplets
Final Heterogenous System

Assumptions

- 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**

Security Challenges

- Fundamental security risks of HI-based devices
  - Use of diverse, mostly untrusted and insecure, chiplets that might contain malicious functionality, counterfeits issues
  - Shared resources between chiplets introduces new attack surface, e.g., access control violations and information
  - Variety of in-field and physical attacks such as inter-chiplet interconnects
- New attack surfaces and vectors of system-in-package (SiP)



SiP Security

- Fault Injection
- Reverse Engineering
- Side-Channel Attacks
- Probing (Contact/Contactless)
- Software / Malware Attacks
- Recycling
- Package Level Security

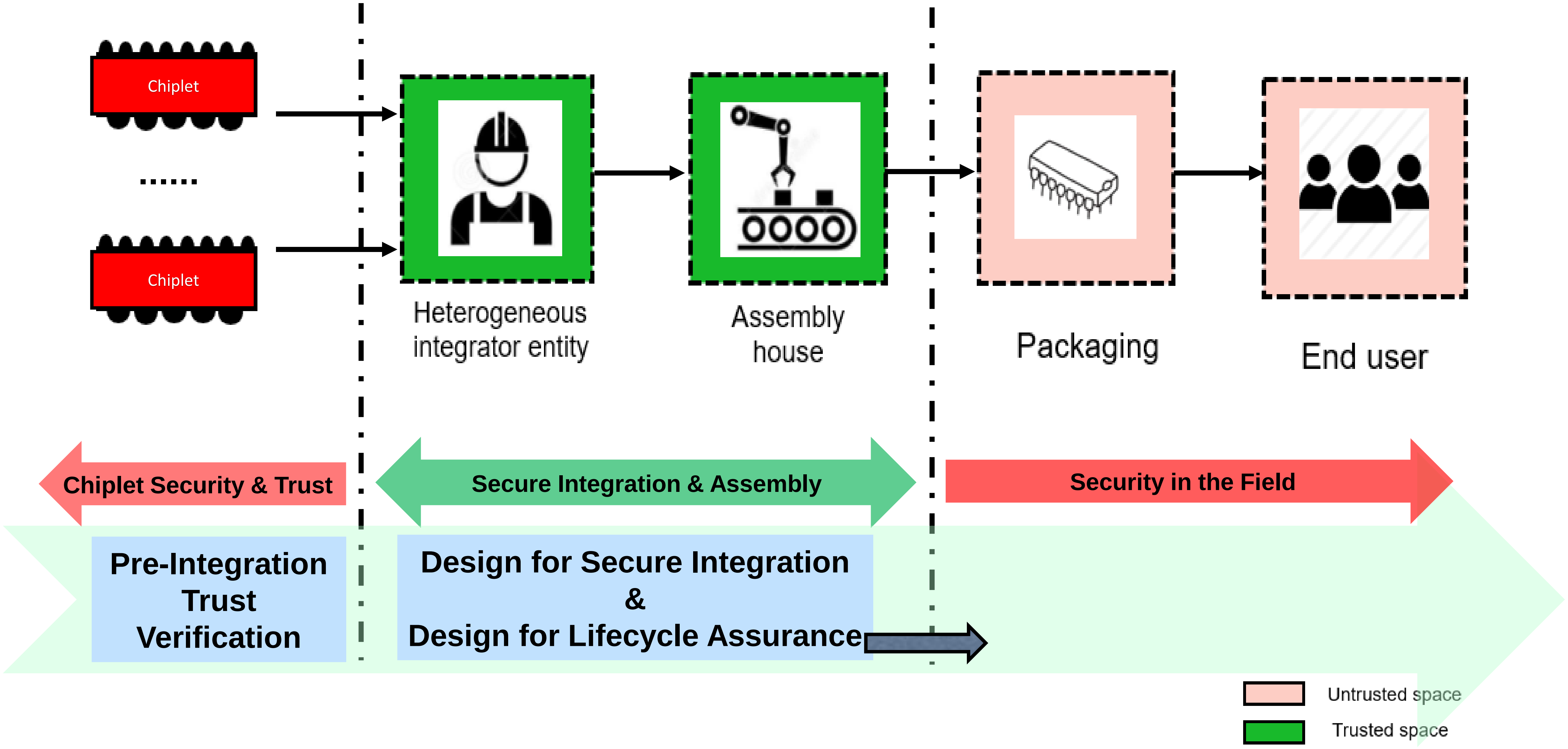
Interposer Security

- Hardware Trojan
- Reverse Engineering
- SiP IP Piracy

Chiplet Security

- Hardware Trojan
- Chiplet IP Piracy
- Out-of-Spec
- Overproduction
- Cloning

Supply Chain of Heterogenous Integration



Chiplet

......

Chiplet

Heterogeneous integrator entity

Assembly house

Packaging

End user

Chiplet Security & Trust

Secure Integration & Assembly

Security in the Field

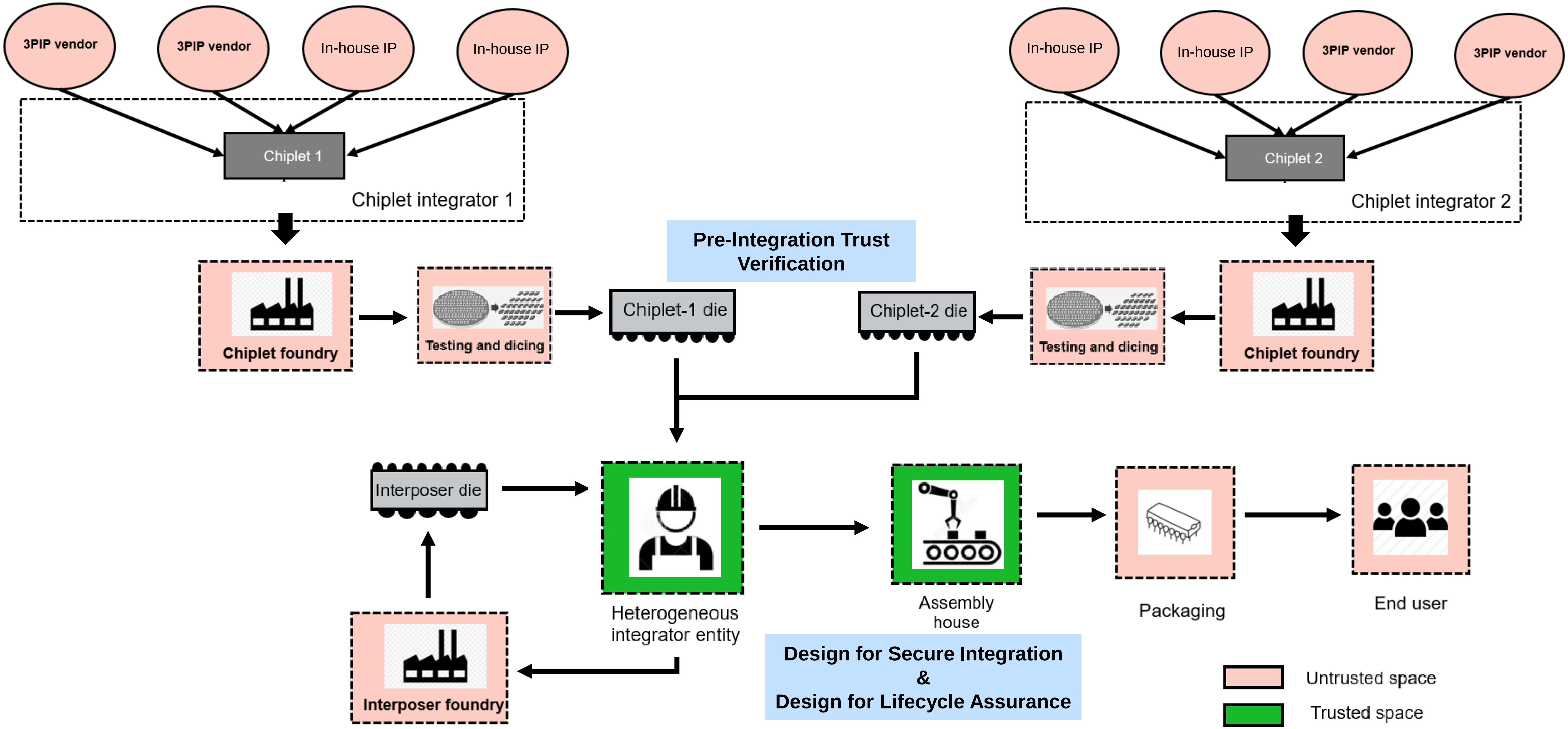
Pre-Integration Trust Verification

Design for Secure Integration & Design for Lifecycle Assurance

□ Untrusted space

🟩 Trusted space

Supply Chain of Heterogenous Integration



Pre-Integration Trust Verification

Design for Secure Integration
&
Design for Lifecycle Assurance

🟥 Untrusted space

🟩 Trusted space

Threat Model

- **Chiplet Security & Trust**
  - IP piracy
  - Hardware Trojans and malicious change
  - Reverse engineering
  - Counterfeit chiplets (Cloned, out of spec, recycled, etc)

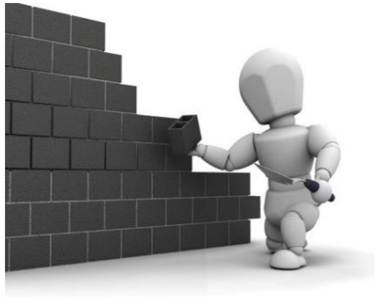
- **Design for Secure Integration**
  - Information leakage
  - Confidentiality
  - Integrity
  - Security policies

- **Design for Lifecycle Assurance**
  - Secure operation (run time) throughout lifetime
  - Tamper detection
  - Supply chain integrity
  - Over-production & Out of Spec
  - MIM and Impersonation
  - Physical Attacks

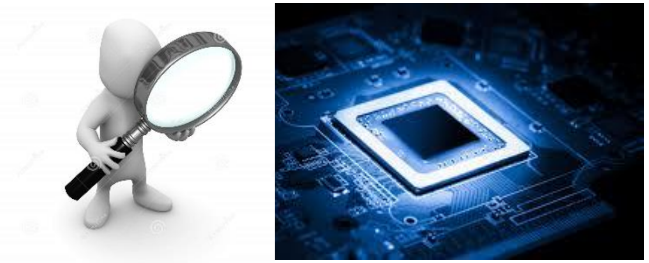
Aggressive threat model, but necessary

Building Secure Heterogenous Integration

- Consider security from very **beginning**
- Identify what needs to be protected (**assets, IPs, chiplets, operations**)
- Evaluate **right level** of security for each asset
- Identify potential **vulnerabilities and threat models**
  - During chiplet design time
  - Interposer and packaging vulnerability Assessment
  - Need to develop a vulnerability database
- **Analyze if vulnerabilities exists**
  - Need to develop metrics, standards, rules and properties
  - Need to develop CAD tools for security assessment



Security from the start



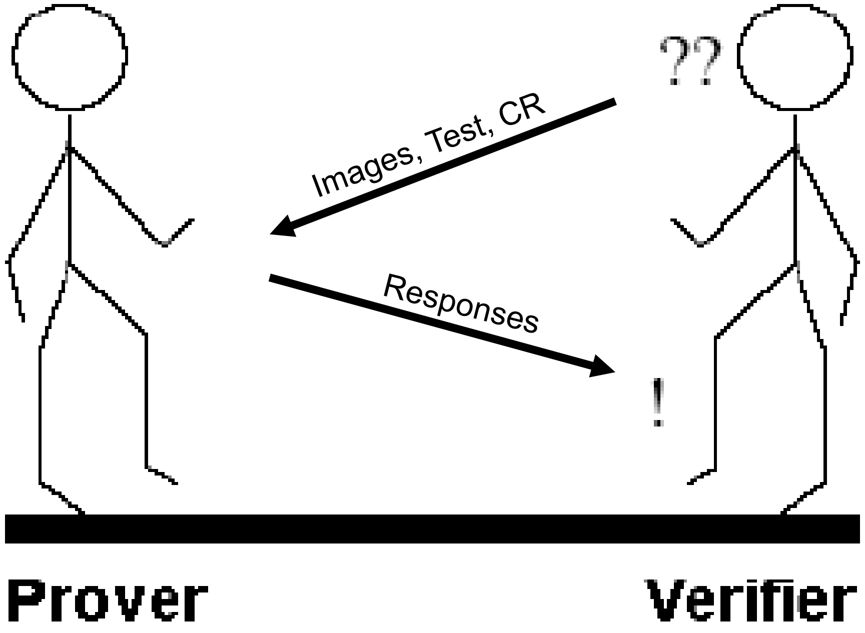
Security assessment

Chiplet Security & Trust

Chiplet Security

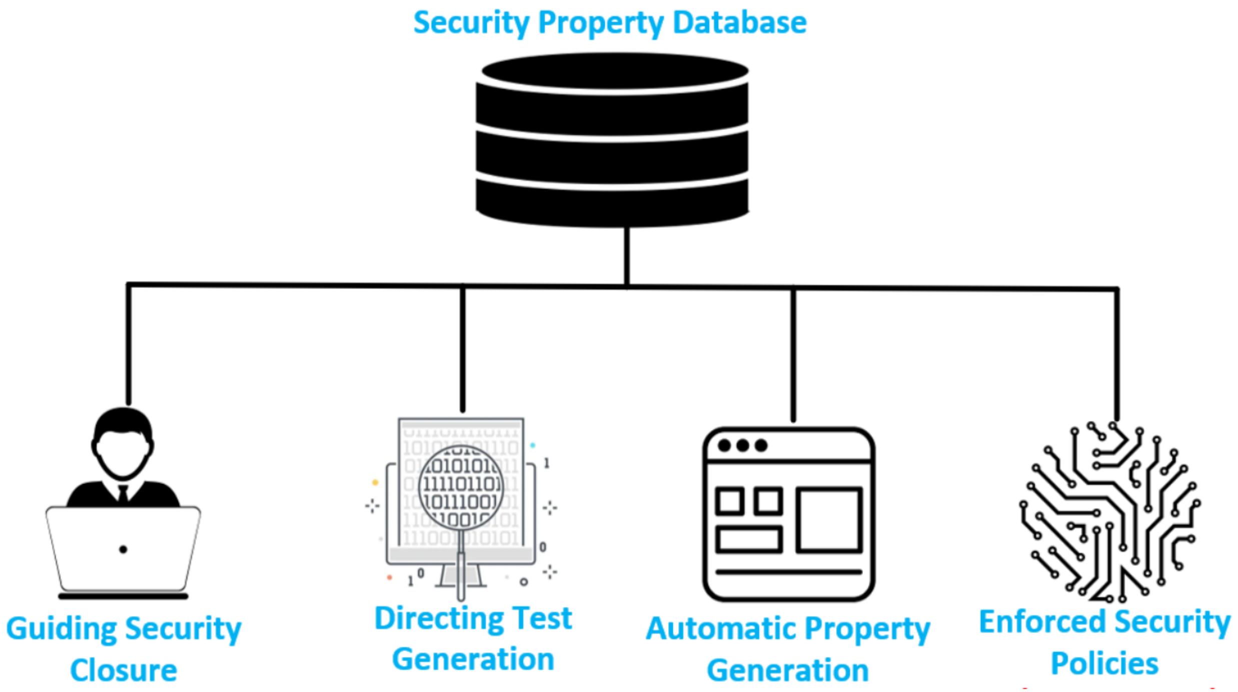
- **Logical Verification**
    - **Attackers: Untrusted Chiplet OCM and foundry**
    - **Challenge-response (CR) based approach**
        - **Logical test, watermark, PUF, etc**
        - ***Insufficient to establish trust***

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



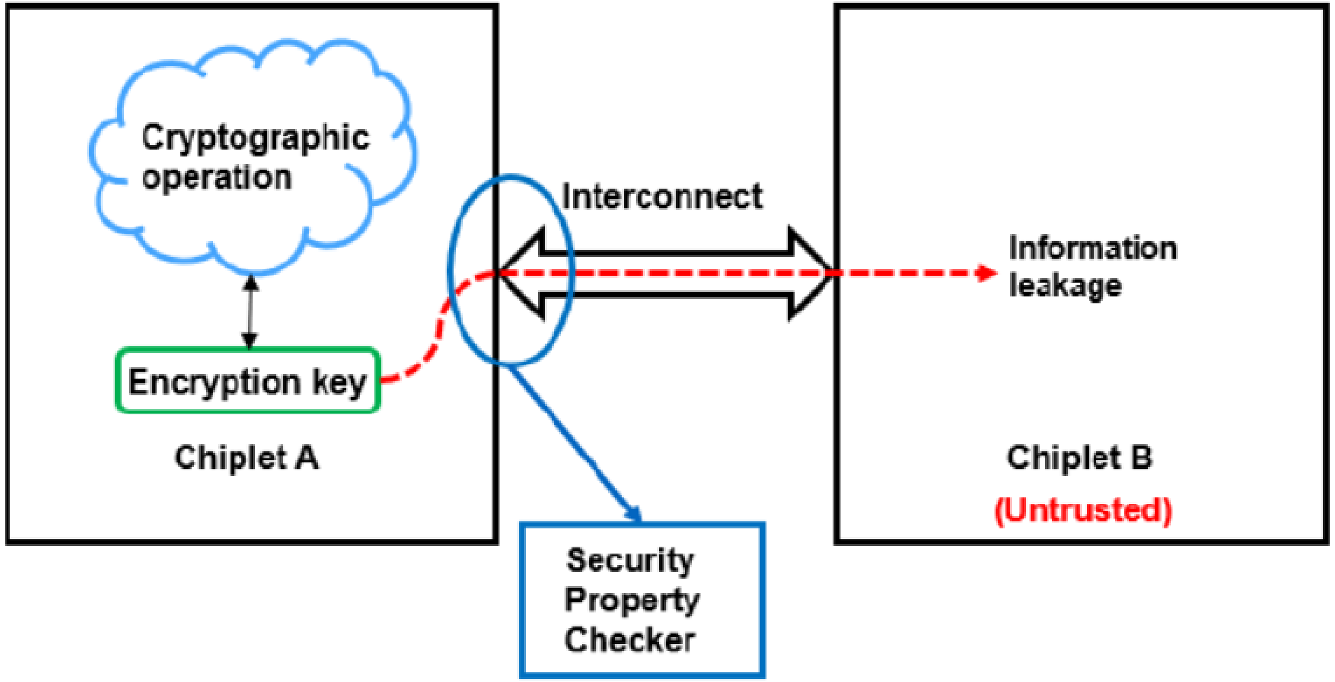
Possible Solutions: Static Security Verification

- Does my SiP behave in a secure manner? – Security Property and Rules
- No comprehensive set of security properties for heterogeneous system is available
- Our subtasks to resolve the challenges
  - Establish a comprehensive database of security properties
  - Automatic security property generation

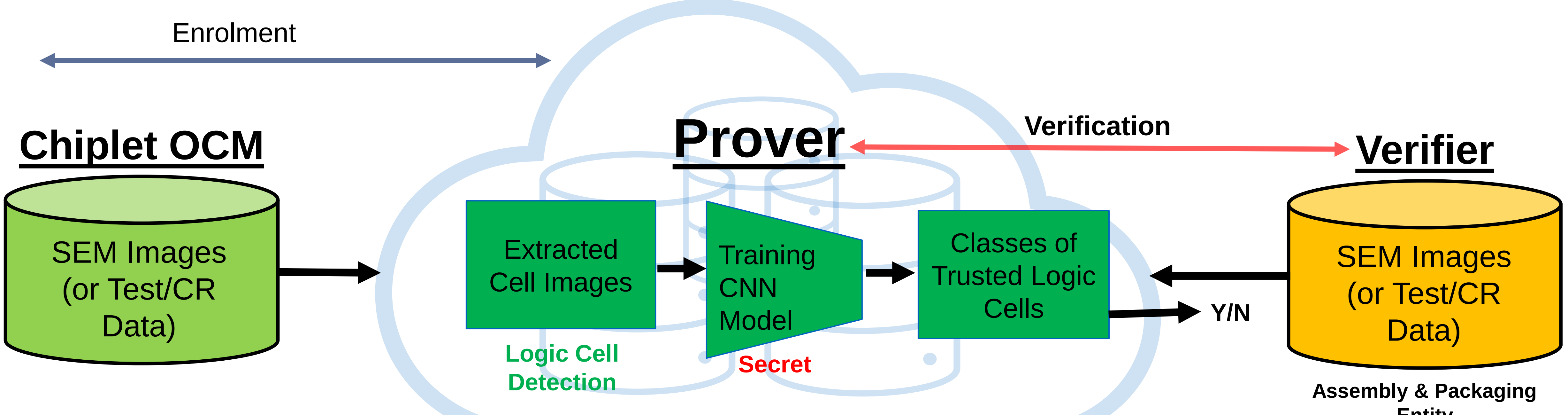

Security Property Database
Guiding Security Closure
Directing Test Generation
Automatic Property Generation
Enforced Security Policies

Possible Solutions: Security Property Database Generation

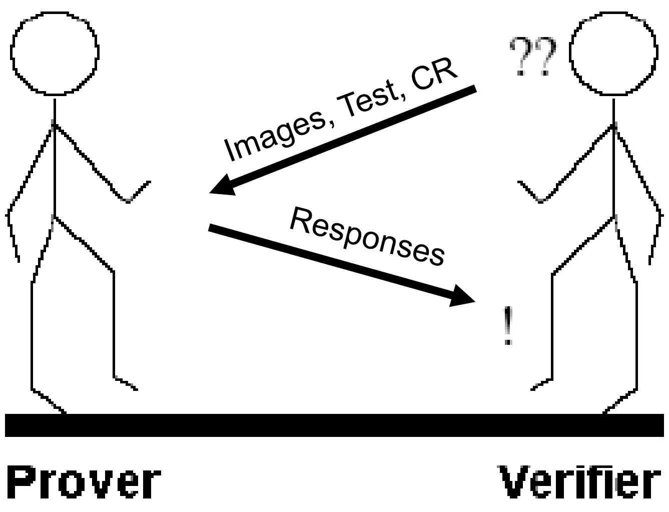
- Security property checking – presence/absence of vulnerabilities
- Characterizations of a SiP security property
- Property formalization when chiplet implementation details are unknown
  - Primary asset identification
  - Secondary asset identification
  - Vulnerability detection
  - Threat model development



Enrolment & Verification



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

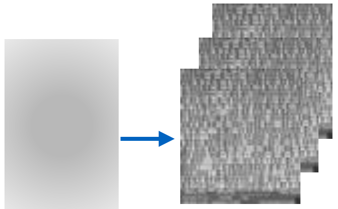


Physical Verification

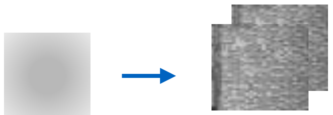
Verifier

Backside SEM Images

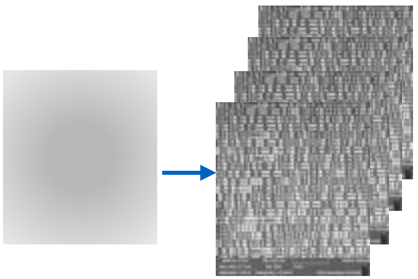
A



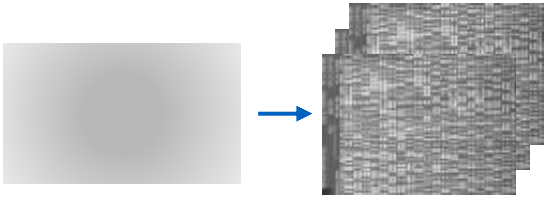
B



C

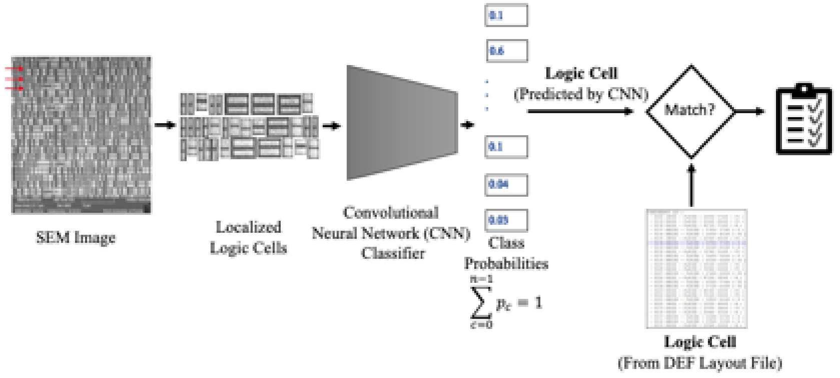


D



Prover

Malicious Change Detection


SEM Image
Localized Logic Cells
Convolutional Neural Network (CNN) Classifier
0.1
0.6
0.1
0.04
0.05
Class Probabilities
$\sum_{c=0}^{n-1} p_c = 1$
Logic Cell (Predicted by CNN)
Match?
Logic Cell (From DEF Layout File)

Outcome

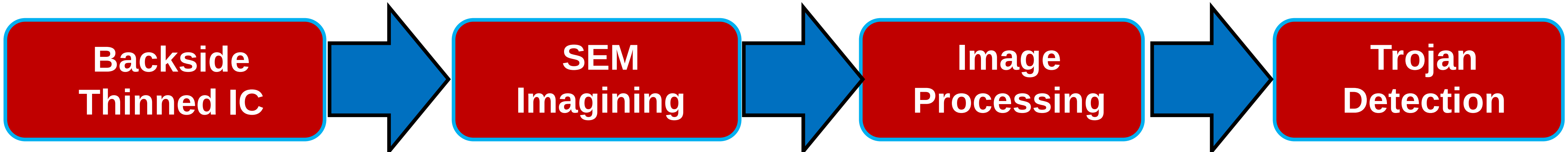
A ✓

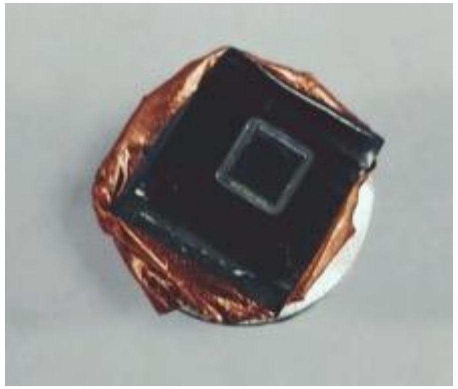
B ✓

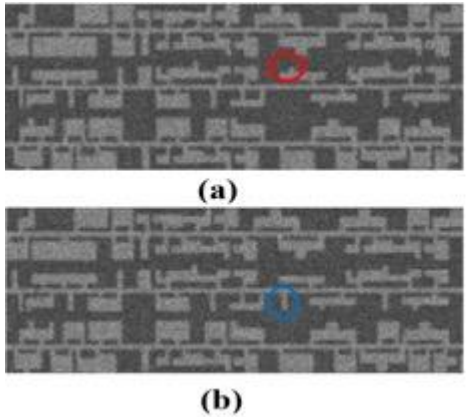
C 😈

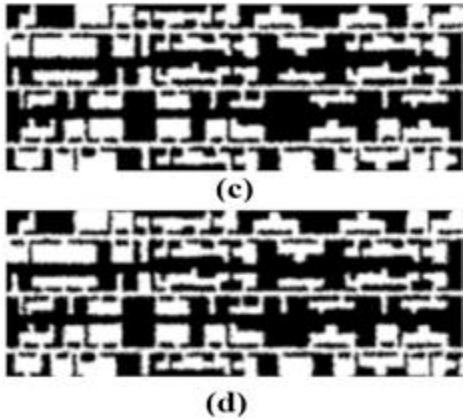
D ✓

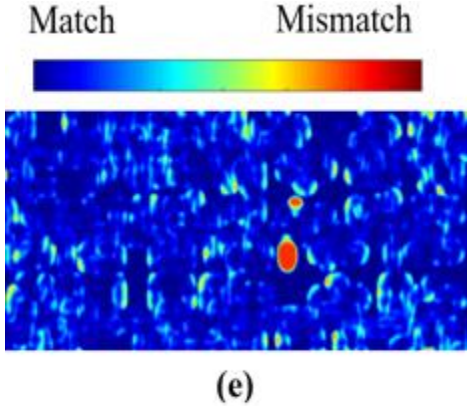
Physical Verification






(a)
(b)


(c)
(d)



(e)

Setting Parameters

i. High Voltage (HV)
ii. Dwelling time (Speed)
iii. Field of View (FoV) / (Magnification)
iv. Resolution

Capturing Images

(a) Chiplet Under Auth

Image Registration

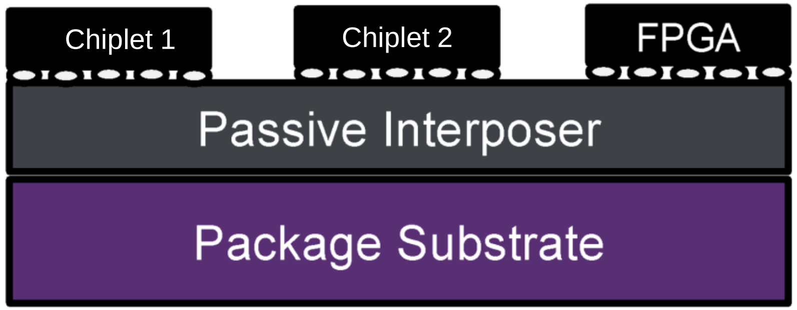
- Noise Removal - FFT BP filter
- Binarization - Adaptive Thresholding
- Smoothening - Gaussian Filter
- Flood Fill

Detection

- Optimized - **S**tructural **SIM**ilarity Index (SSIM) algorithm.
- Threshold based image labelling of suspicious areas of chiplet.

Design for Secure Integration &Lifecycle

ISILA: Secure Integration and Lifecycle Assurance



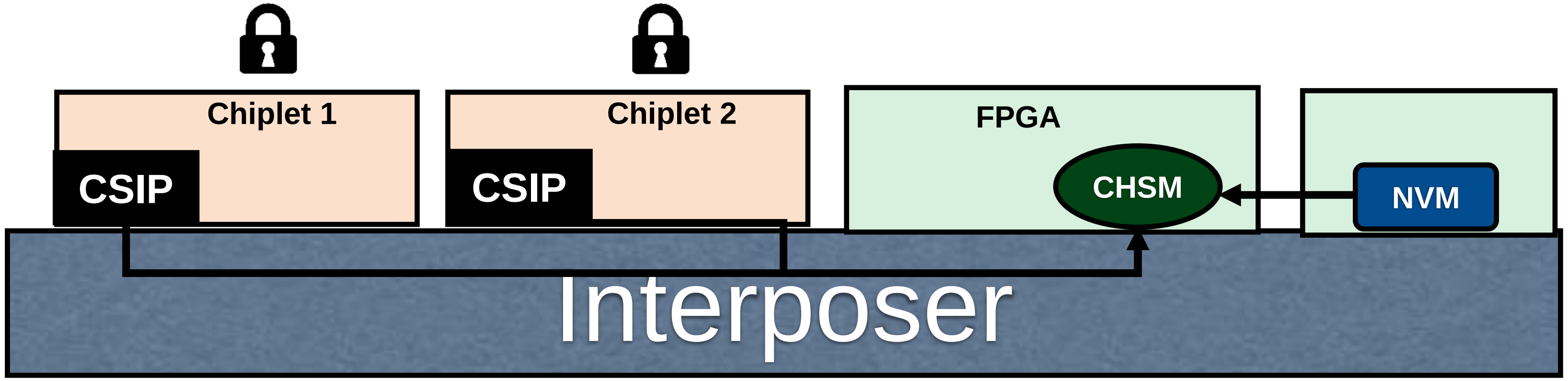
Brief Description:

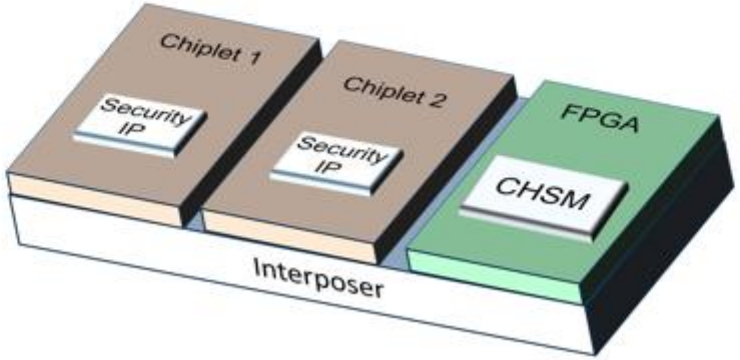
- Chiplets 1 and 2 fabricated using advanced technology node in untrusted foundry
  - Sensitive chiplets could be *locked* or have *stripped* functionality
- The FPGA is configured by the IC designer and the configuration data, i.e., bitstream, is unknown to the potential adversaries

ISILA's 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.

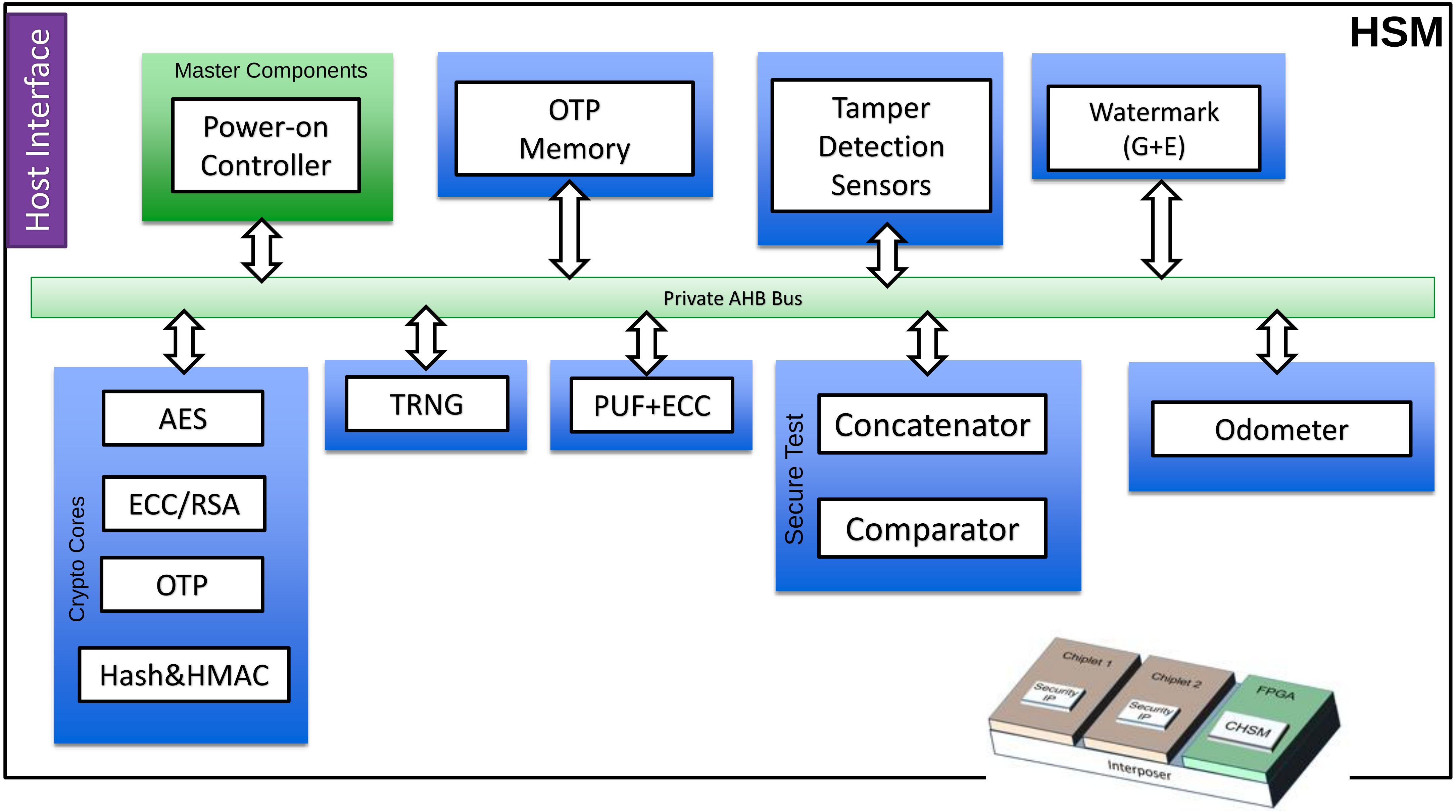
ISILA





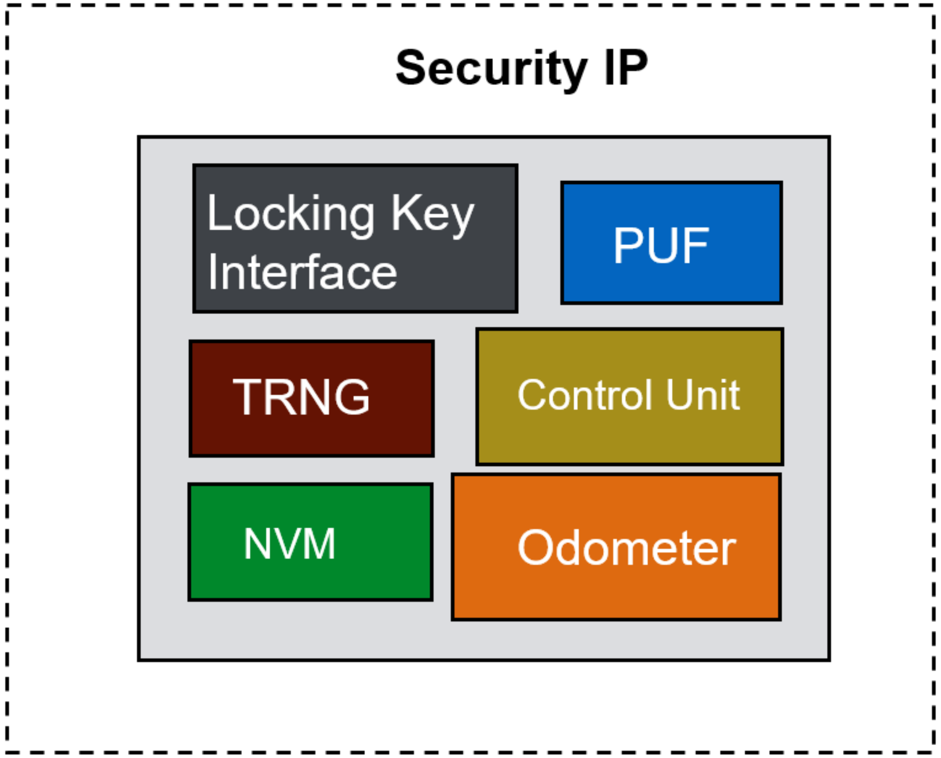
- Each chiplet must be **authenticated**
  - Challenge-response protocol
- Some chiplets may be **logic locked,** each requiring a separate key to unlock its functionality.
  - Logic locking keys should not be securely hard coded in the netlist or provisioned by the untrusted foundry.
  - The logic locking 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.

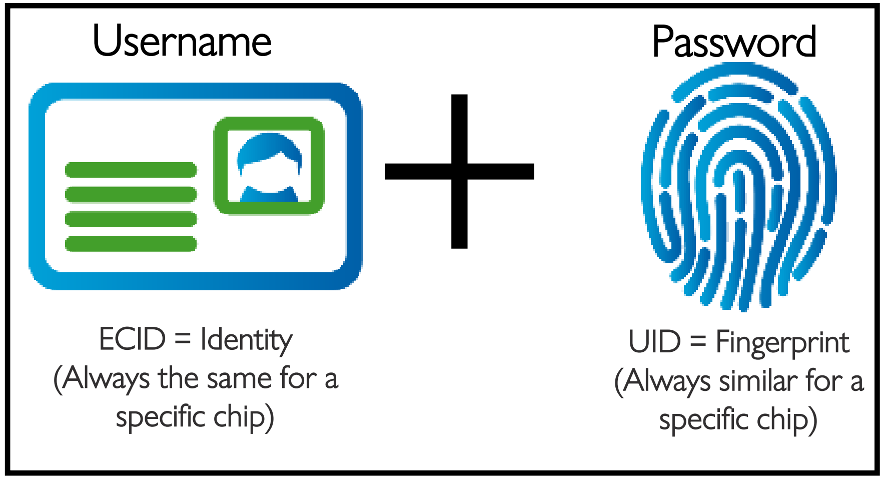
CHSM Design

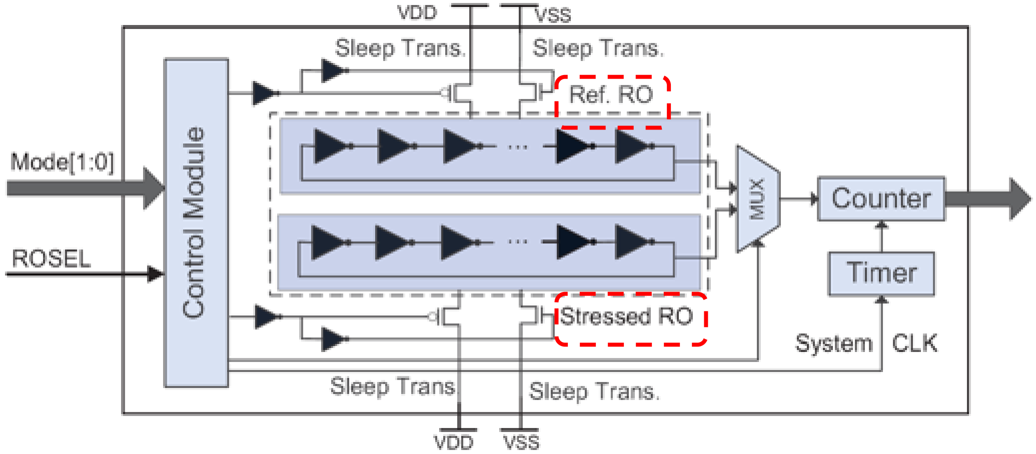

HSM
Host Interface
Master Components
Power-on Controller
OTP Memory
Tamper Detection Sensors
Watermark (G+E)
Private AHB Bus
Crypto Cores
AES
ECC/RSA
OTP
Hash&HMAC
TRNG
PUF+ECC
Secure Test
Concatenator
Comparator
Odometer
Chiplet 1
Security IP
Chiplet 2
Security IP
FPGA
CHSM
Interposer

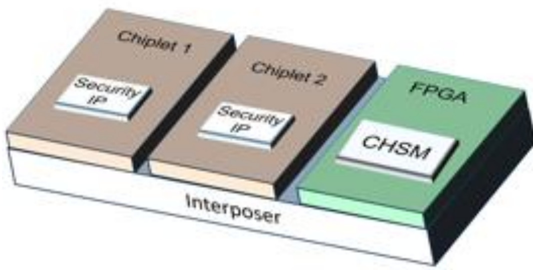
CSIP Design

- 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.

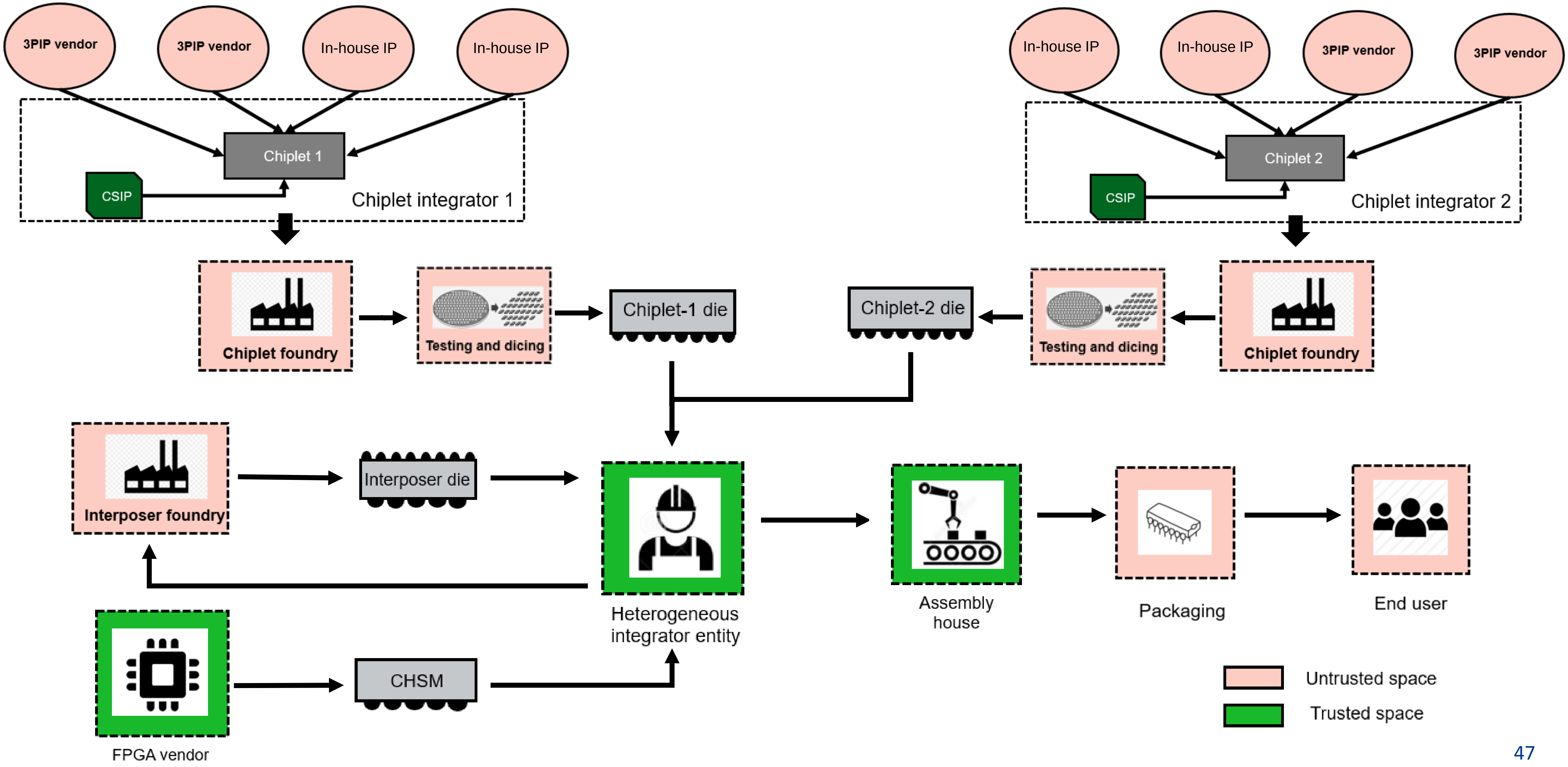



Username
Password
ECID = Identity
(Always the same for a specific chip)
UID = Fingerprint
(Always similar for a specific chip)



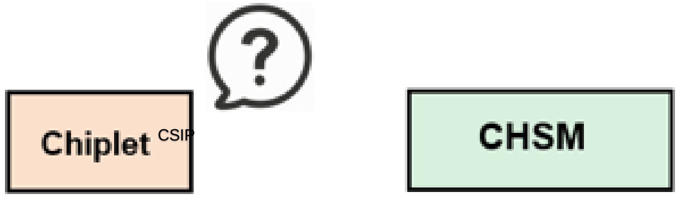


Secure Design Flow for HI

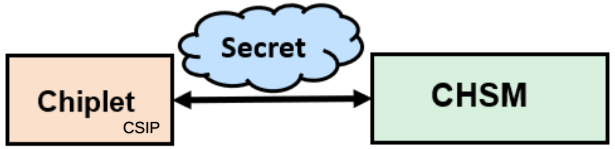


Design Requirements

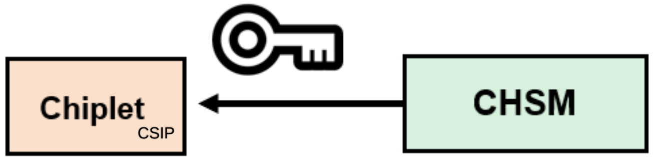
I. Authentication



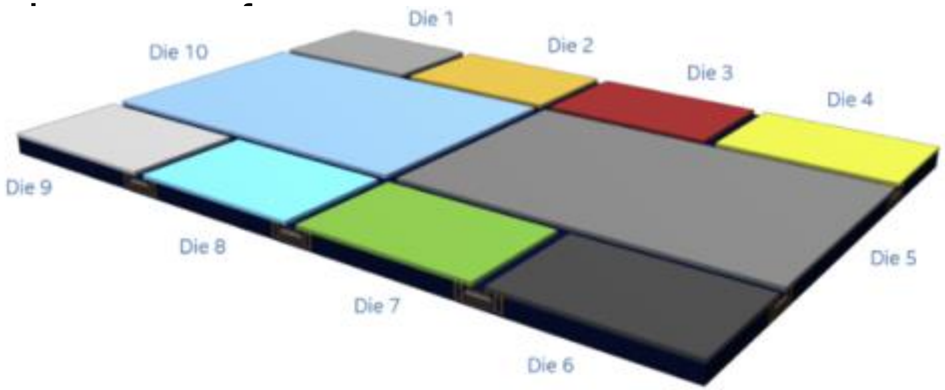
II. Key exchange



III. Locking key Transfer



- The CHSM is required to :
  - authenticate the chiplets in the package: Only enrolled chiplets identified by CHSM,
  - establish communication with the CSIPs,
  - securely deliver the keys to the respective chiplets through the CSIP.
- Key must **not** pass through the interposer layer in plaintext.
- The CSIP and CHSM design should be scalable and adaptal
  - size of locking key,
  - number of chiplets in the package,
  - type of locking scheme used.



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:

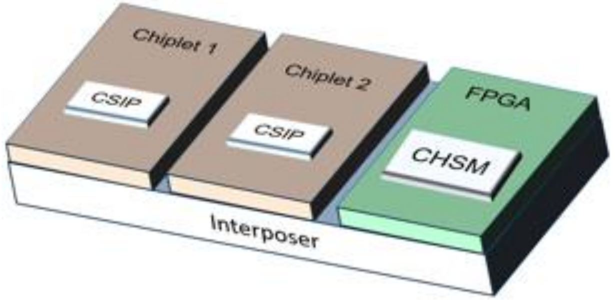
- Over-production
- Out-of-spec
- Defective
- Remarked
- Cloned

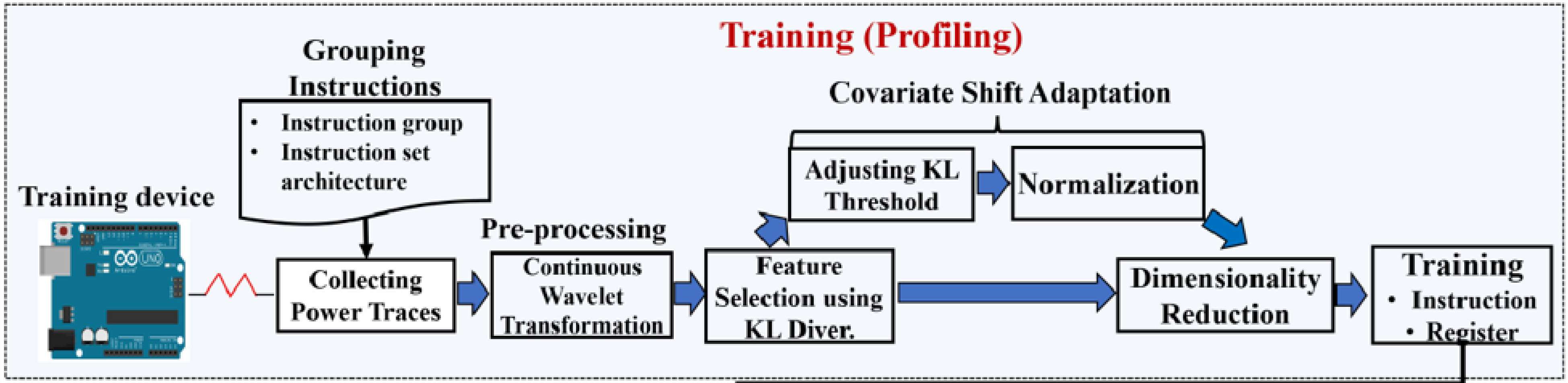
G. Contreras et. al., "Secure Split-Test for preventing IC piracy by untrusted foundry and assembly," *IEEE International Symposium Defect and Fault Tolerance in VLSI and Nanotechnology Systems (DFT),* pp.196-203, 2013.

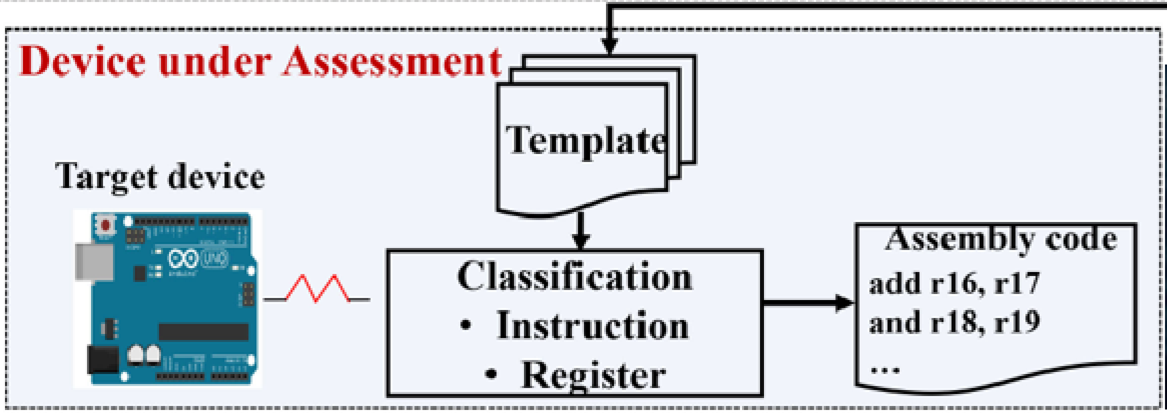
T. Rahman, D. Forte, Q. Shi, G. Contreras, and M. Tehranipoor, "CSST: Preventing Distribution of Unlicensed and Rejected ICs by Untrusted Foundry and Assembly,"IEEE Int. Symposium on Defect and Fault Tolerance Symposium (DFTS), Oct. 2014

Runtime Monitoring using CHSM

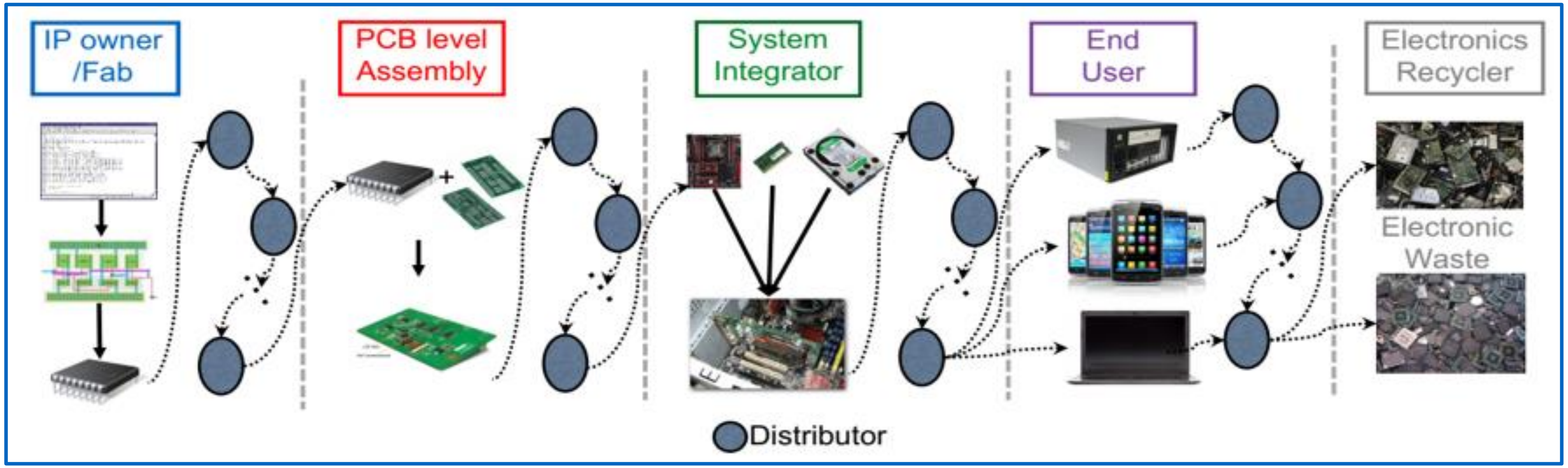
- **Runtime Security and Integrity Checker:** Equip FPGA with sensors to measures and perform side channel analysis
  - Enable detailed program analysis





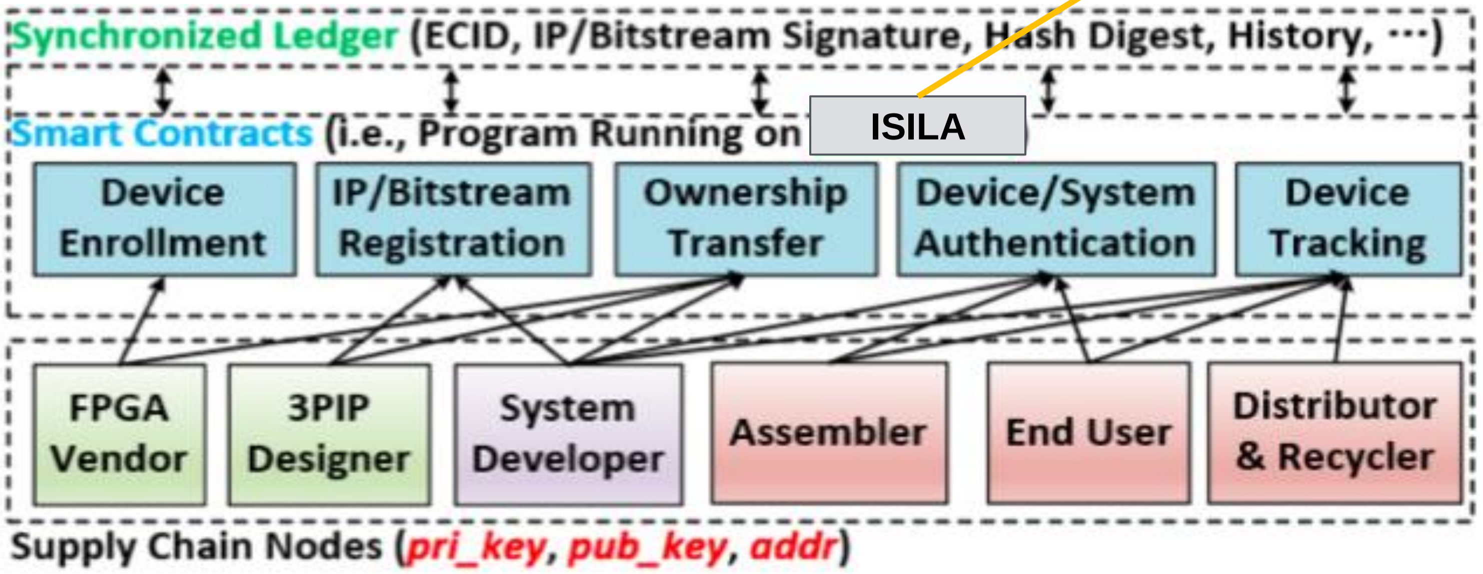


Supply Chain Integrity using CHSM/CSIP



Chiplet 1
Chiplet 2
FPGA
CSIP
CSIP
CHSM
Interposer

- **ISILA infrastructure offers end-to-end protection**
- **Available smart contracts**
  - Device enrollment
  - IP/Bitstream registration
  - Ownership transfer
  - Device/system auth.
  - Device tracking


Synchronized Ledger (ECID, IP/Bitstream Signature, Hash Digest, History, ⋯)

Smart Contracts (i.e., Program Running on ISILA

Device Enrollment | IP/Bitstream Registration | Ownership Transfer | Device/System Authentication | Device Tracking

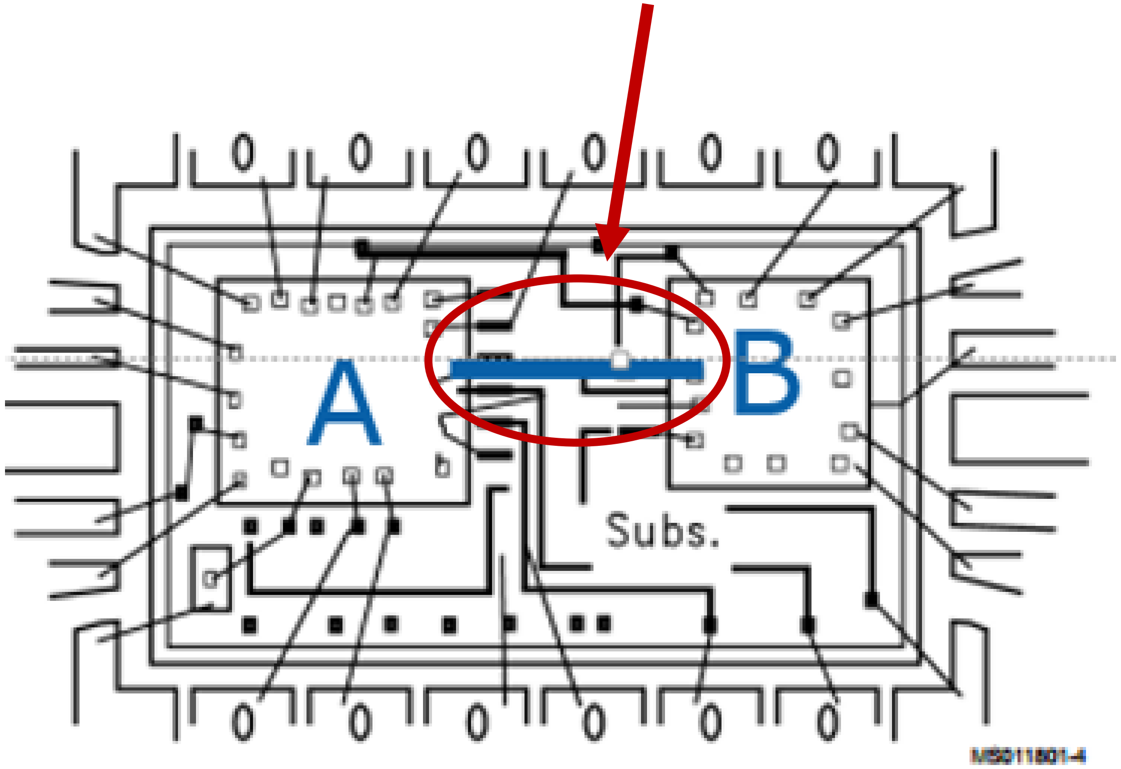
FPGA Vendor | 3PIP Designer | System Developer | Assembler | End User | Distributor & Recycler

Supply Chain Nodes (*pri_key*, *pub_key*, *addr*)

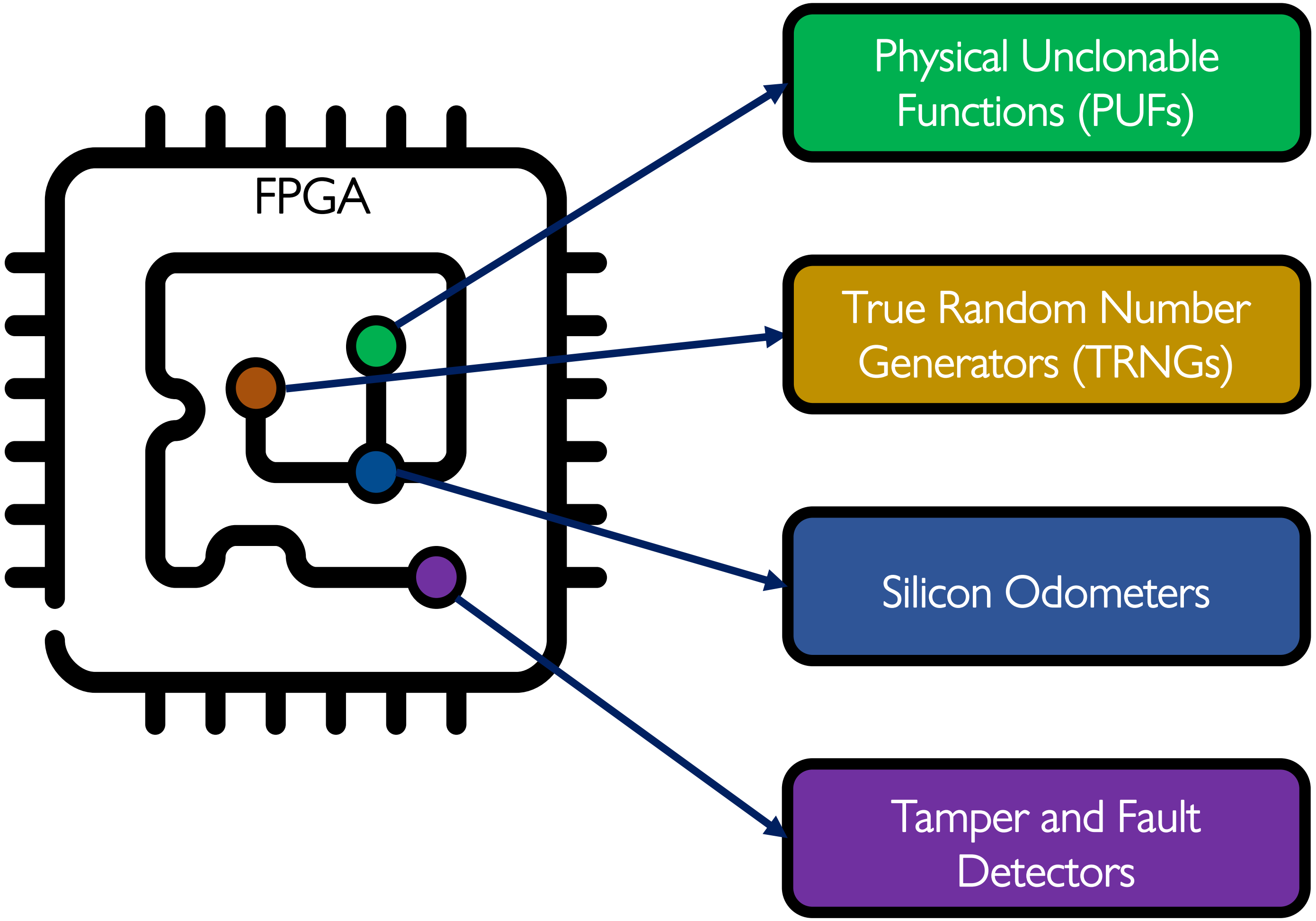
Physical Assurance

- Signals with confidentiality and integrity requirements should pass between dies using an authenticated encryption protocol.
- **Approaches:**
  - ✓ Anti-tampering sensors
  - ✓ Active and passive shields
  - ✓ Watermarks on package
  - ✓ PUF based authentication

Potential attack surface



ISILA – Tamper Detection



Physical Unclonable Functions (PUFs)

Generates unique silicon fingerprints
-- On-demand key generation
-- Device authentication

True Random Number Generators (TRNGs)

Generates robust random bitstream
-- Random seeds
-- Nonce and initialization vectors

Silicon Odometers

Estimates device lifecycle
-- Power up and boot sequence
-- Embedded usage log

Tamper and Fault Detectors

Detects external attacks and influences
-- Voltage/Power sensors
-- Clock glitch (timing) sensors
-- X-ray sensors
-- Optical detection

?