

TrustSoC : A heterogeneous secure-by-design SoC architecture

Raphaële Milan

Thesis supervisor: Lilian Bossuet

Thesis co-supervisor: Loïc Lagadec

Why listen to this presentation ?

I don't use SoC everyday it does not concern me ..

Why listen to this presentation ?

~~I don't use SoC everyday it does not concern me ..~~

I use SoCs everyday it does concern me !



Smartphone



Military



Autonomous car



Cloud

Throwback to your phone of 2011 ...



ARM Cortex-A8 and PowerVR

512 MiB RAM

1 GHz

iPhone 4
2011

45 nm / 53,3 mm²
149 million transistors

CPU 6 cores / GPU 5 cores
Neural Engine 16 cores

8 Go RAM

4,04 GHz

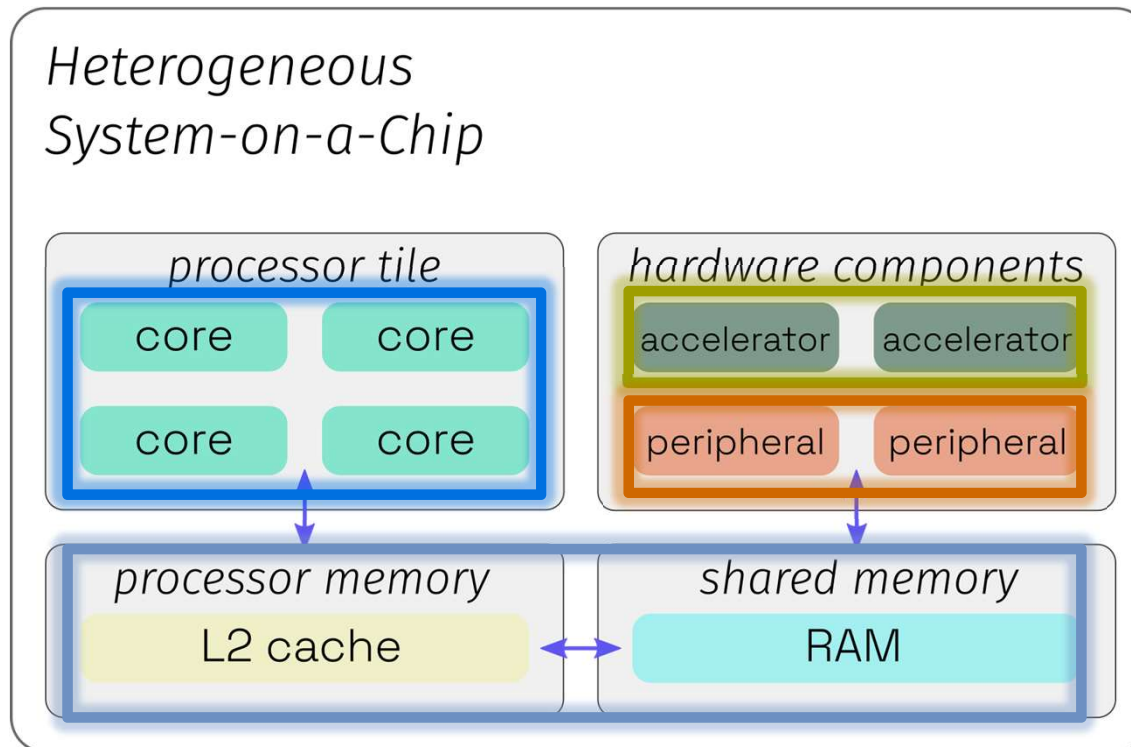
Face ID / Apple pay / Apple intelligence



iPhone 16
2024

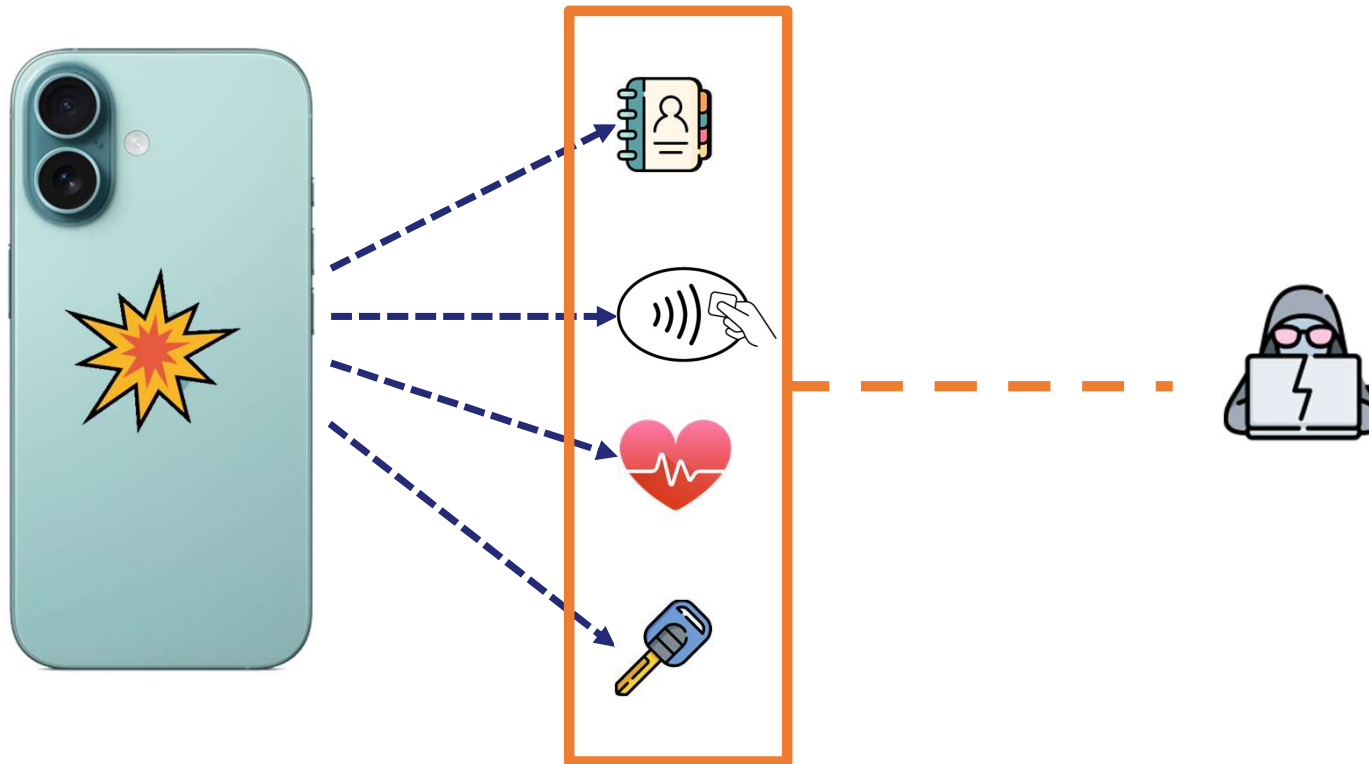
3nm / 90 mm²
~ 17 billion transistors

A solution to enhance performance: sharing !



- Simplify routing without incrementing access time
- Execution between processes quicker

Is the system secure ?

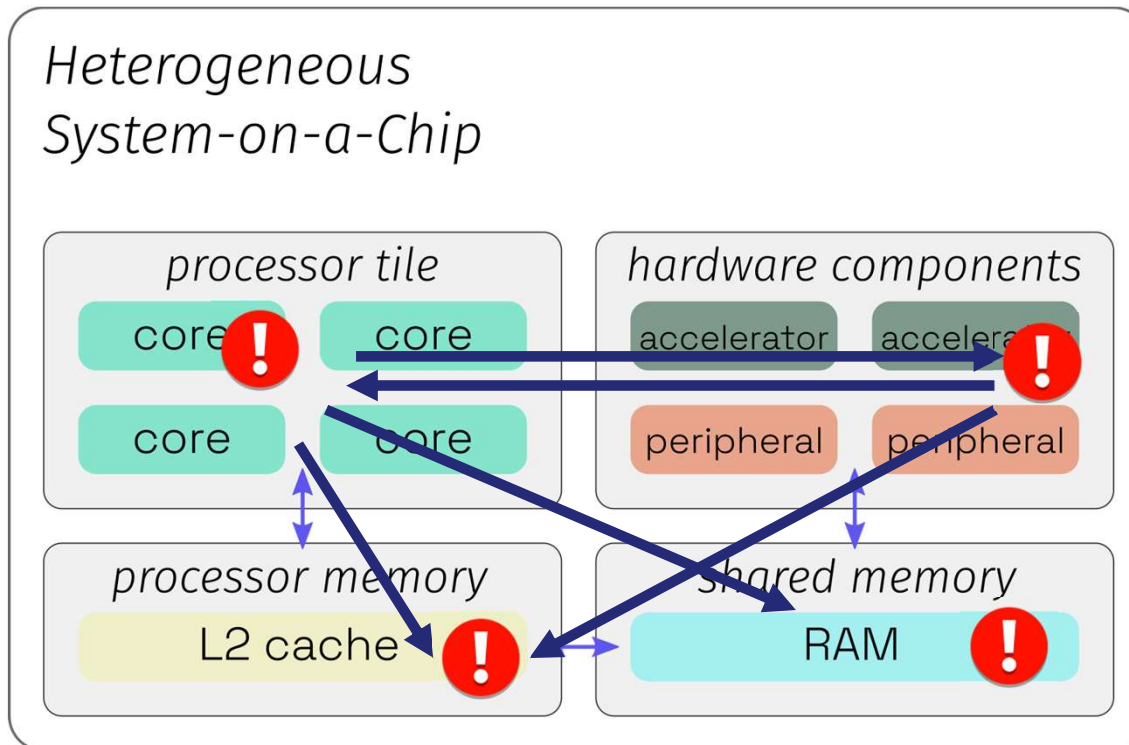


29/11/2024

The darkside of the performance race

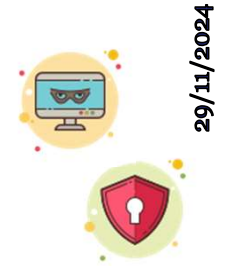


29/11/2024

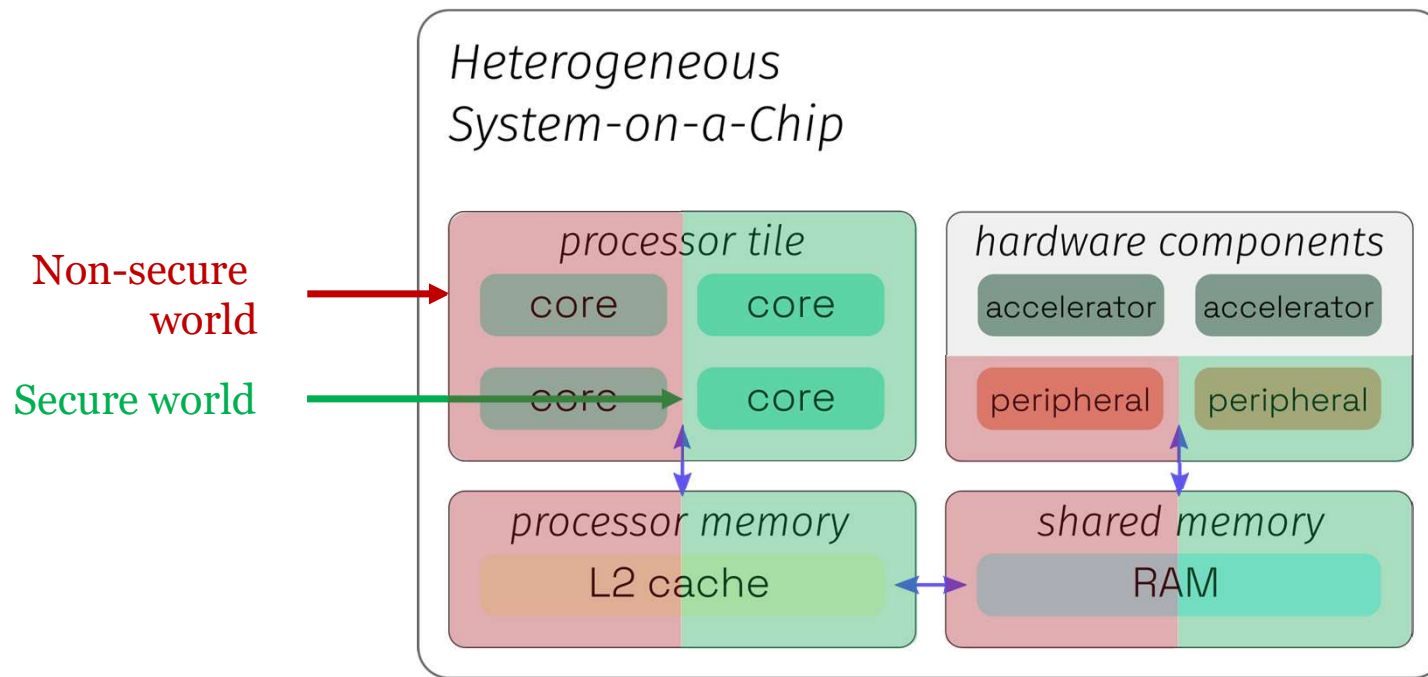


- Steal some information
- Introduce errors
- Hijack the system

Security solution : ARM TrustZone



29/11/2024

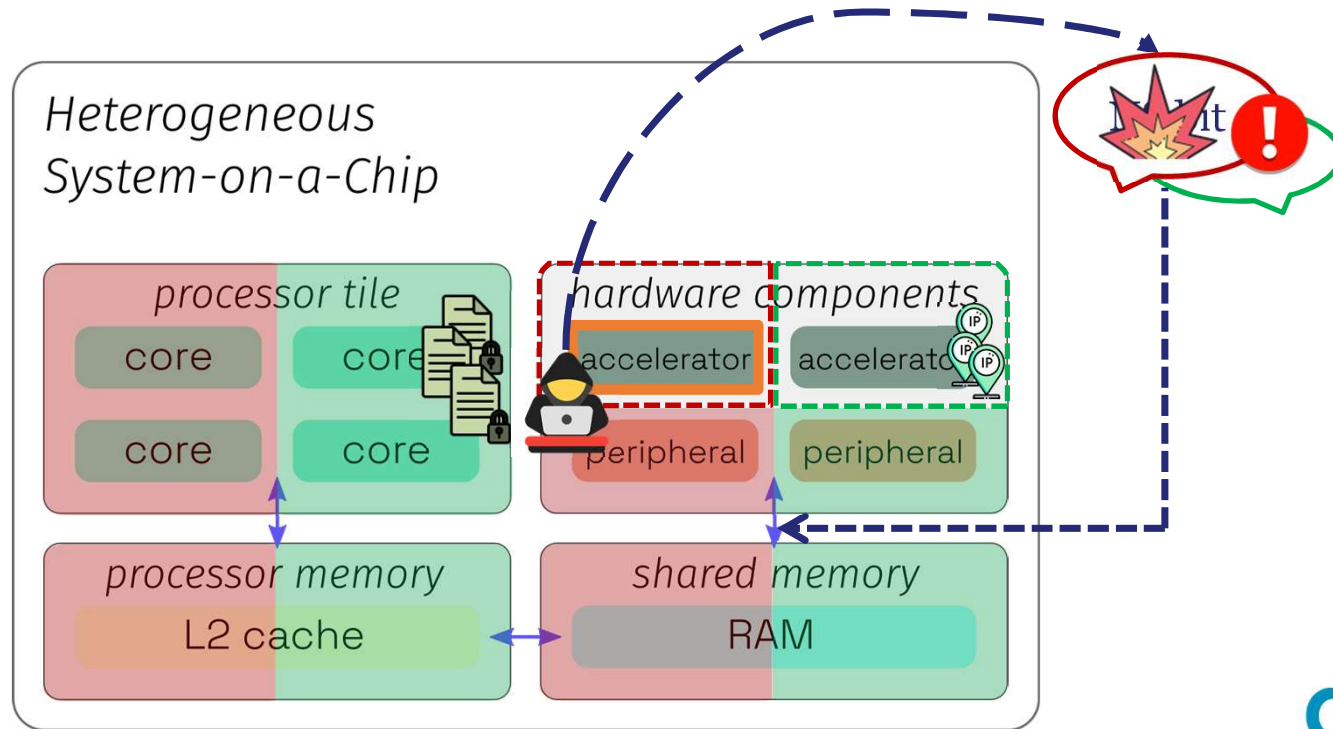


arm
TRUSTZONE

[1] A. Ltd, "TrustZone for Cortex-A – Arm®," Arm | The Architecture for the Digital World. <https://www.arm.com/technologies/trustzone-for-cortex-a>

[2] E. M. Benhani, L. Bossuet, and A. Aubert, "The security of arm trustzone in a fpga-based soc," IEEE Transactions on computers, vol. 68, no. 8, p. 1238–1248, 2019.

ARM TrustZone AMD Xilinx extension



AMD
XILINX

arm
TRUSTZONE

[1] A. Ltd, "TrustZone for Cortex-A – Arm®," Arm | The Architecture for the Digital World. <https://www.arm.com/technologies/trustzone-for-cortex-a>

[2] E. M. Benhani, L. Bossuet, and A. Aubert, "The security of arm trustzone in a fpga-based soc," IEEE Transactions on computers, vol. 68, no. 8, p. 1238–1248, 2019.

State of the art



Architecture	Type of processor	Threat model	Number of secure domains	Bus protections	Trusted hardware IPs	Protections against DoS attacks
ARM <i>TrustZone</i> [1]	ARM	Software only	1	○	○	○
<i>WorldGuard</i> SiFive [5]	SiFive RISC-V	Software only Privilege escalation attacks No protection inside the same world	N worlds	●	○	○
HECTOR-V [6]	RISC-V	Centered around the TEE	1	●	○	●
CURE [7]	RISC-V	Software attacks only targeting the software or OS	3 types of enclaves	●	○	○

[5] Inc. SiFive. SiFive WorldGuard Technical Paper. 2.4., Santa Clara,CA, July. 2021.

[6] Pascal Nasahl *et al.* « HECTOR-V : A Heterogeneous CPU Architecture for a Secure RISC-V Execution Environment ». In: AsiaCCS 2021. ACM.

[7] Raad Bahmani *et al.* « CURE : A Security Architecture with Customizable and Resilient Enclaves ». In: USENIX, August 2021.

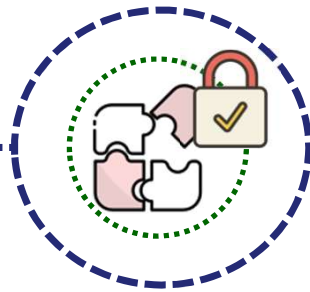
TrustSoC



TrustSoC: motivations and objectives



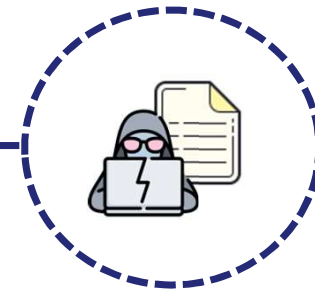
Minimal solution
centered
around the
communications



Include **all** system
components for
security



Protections for
memory system



Threat model
defined

TrustSoC: threat model



29/11/2024

Remote attacks, by intern system blocks

- **A corrupted software application** that tries to access sensitive information of other software applications or hardware IP
- **A corrupted hardware IP** that tries to access sensitive information of other software applications or hardware IP
- **Illegitimate accesses and modifications of the memory contents**



DoS attacks excluded



Compiler, foundry and CAD tool **trusted**

TrustSoC: security features

A secure-by-design architecture **must have security features** to ensure the SoC-FPGA security

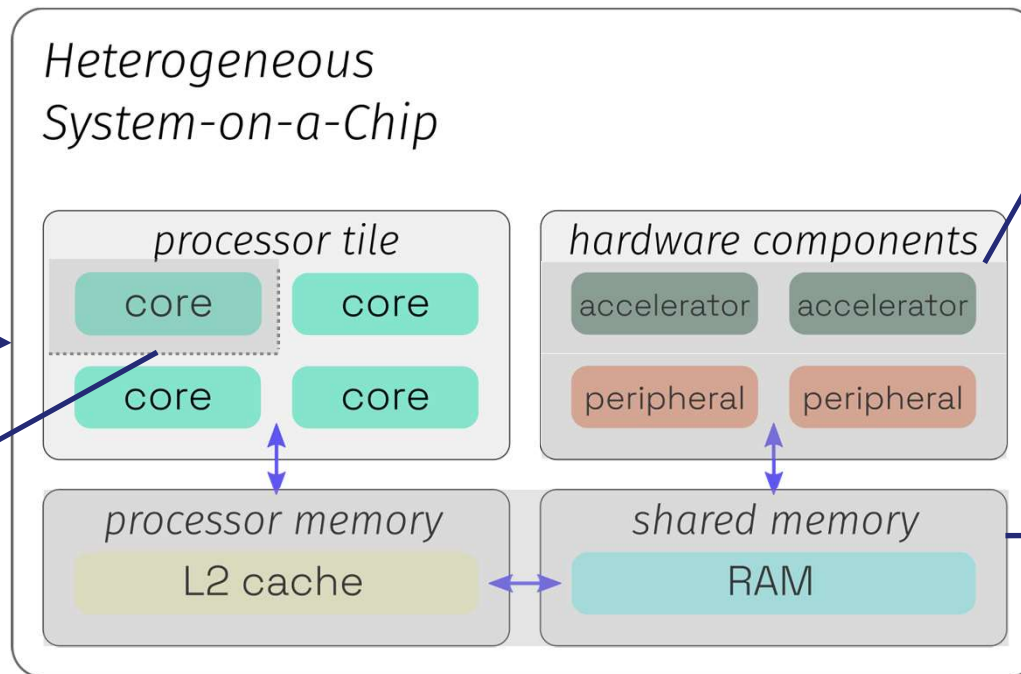
Operating rules





TrustSoC: security features

Operating rules



RS_CPU2: A non-secure application cannot interact with a trusted application without an authorization

RS_FPGA5: After each execution, a hardware IP is reset to its initial state

RS_MEMORY5: A secure or non-secure application cannot modify memory partitions belonging to hardware IPs



TrustSoC: security features

A secure-by-design architecture **must have security features** to ensure the SoC-FPGA security

The features we established:

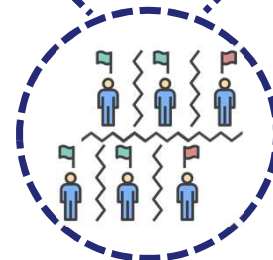
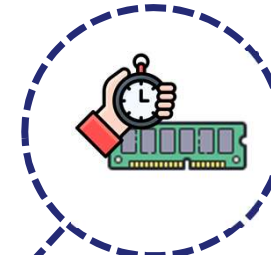
Operating rules



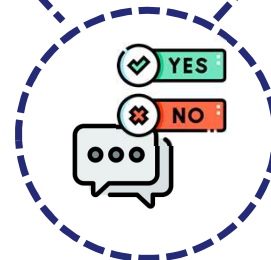
Secure FPGA



Resilience against side-channel attacks

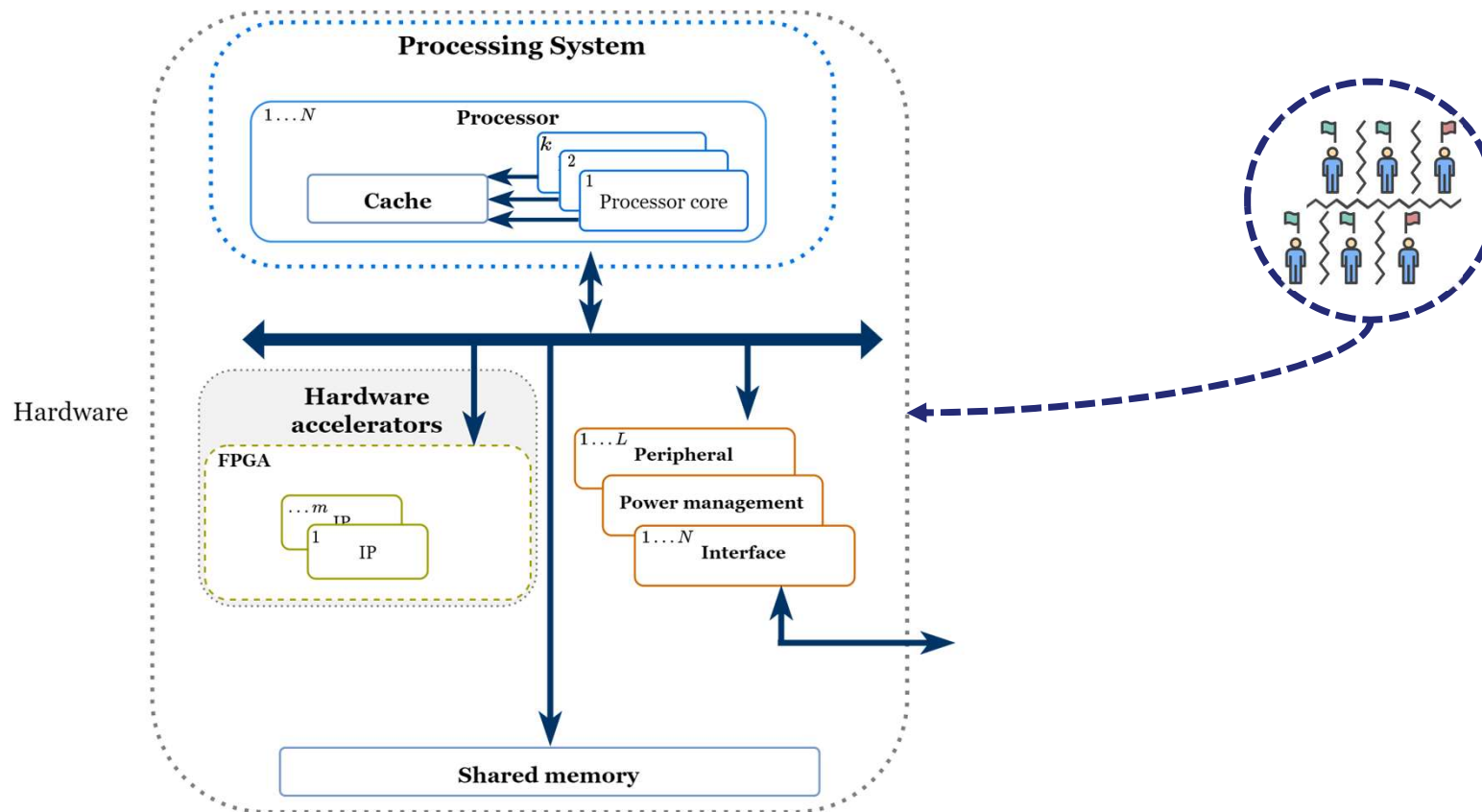


Multiple secure domains



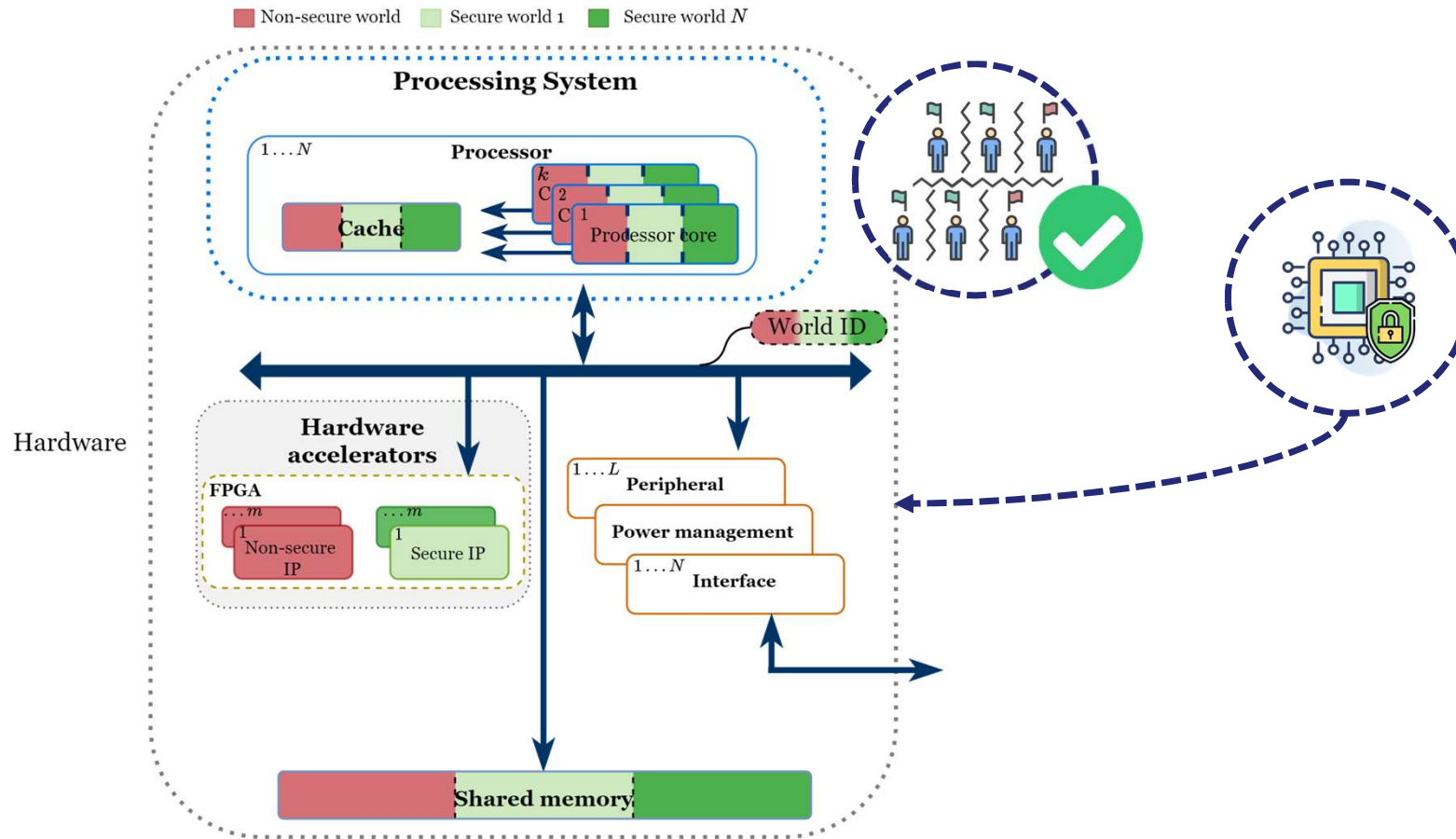
Trusted communications

TrustSoC: architecture prototype



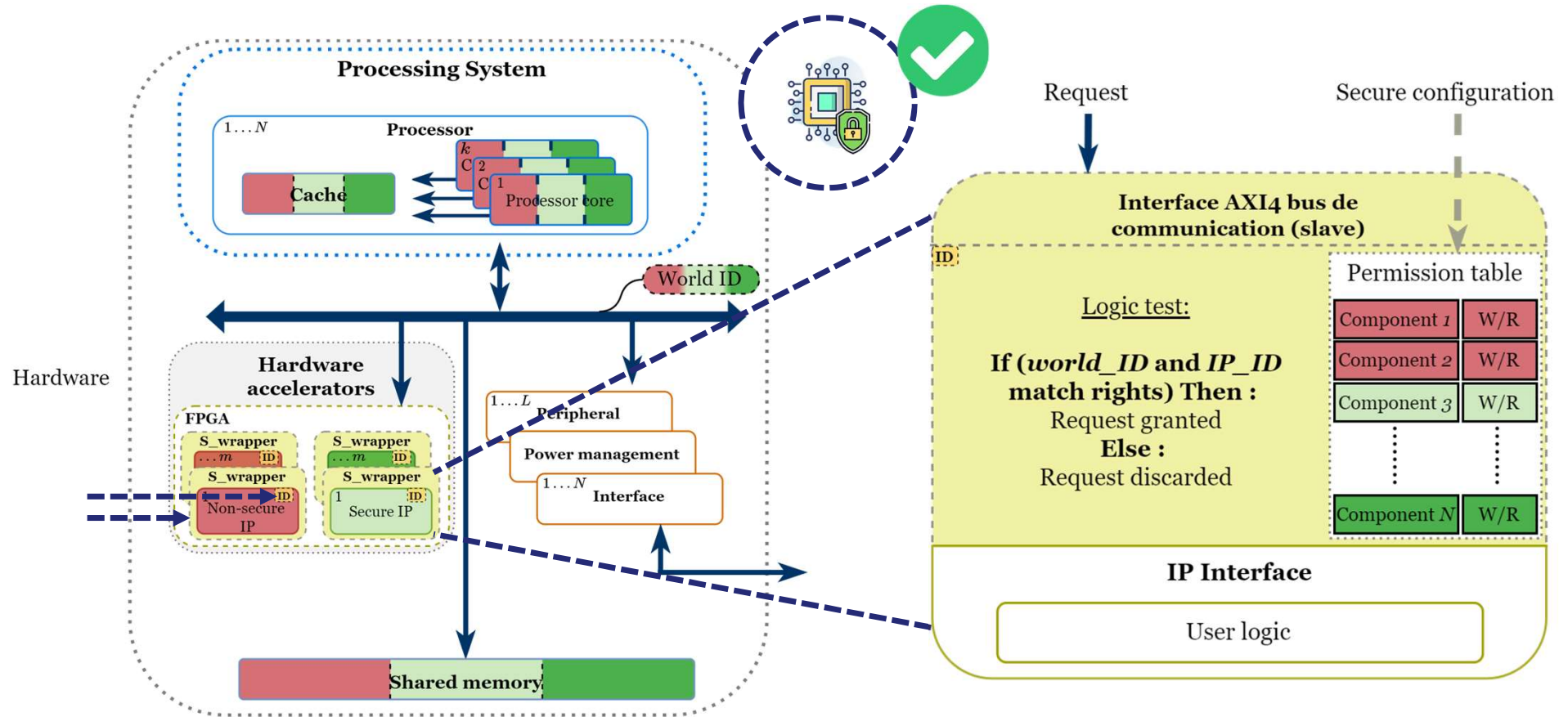


TrustSoC: architecture prototype



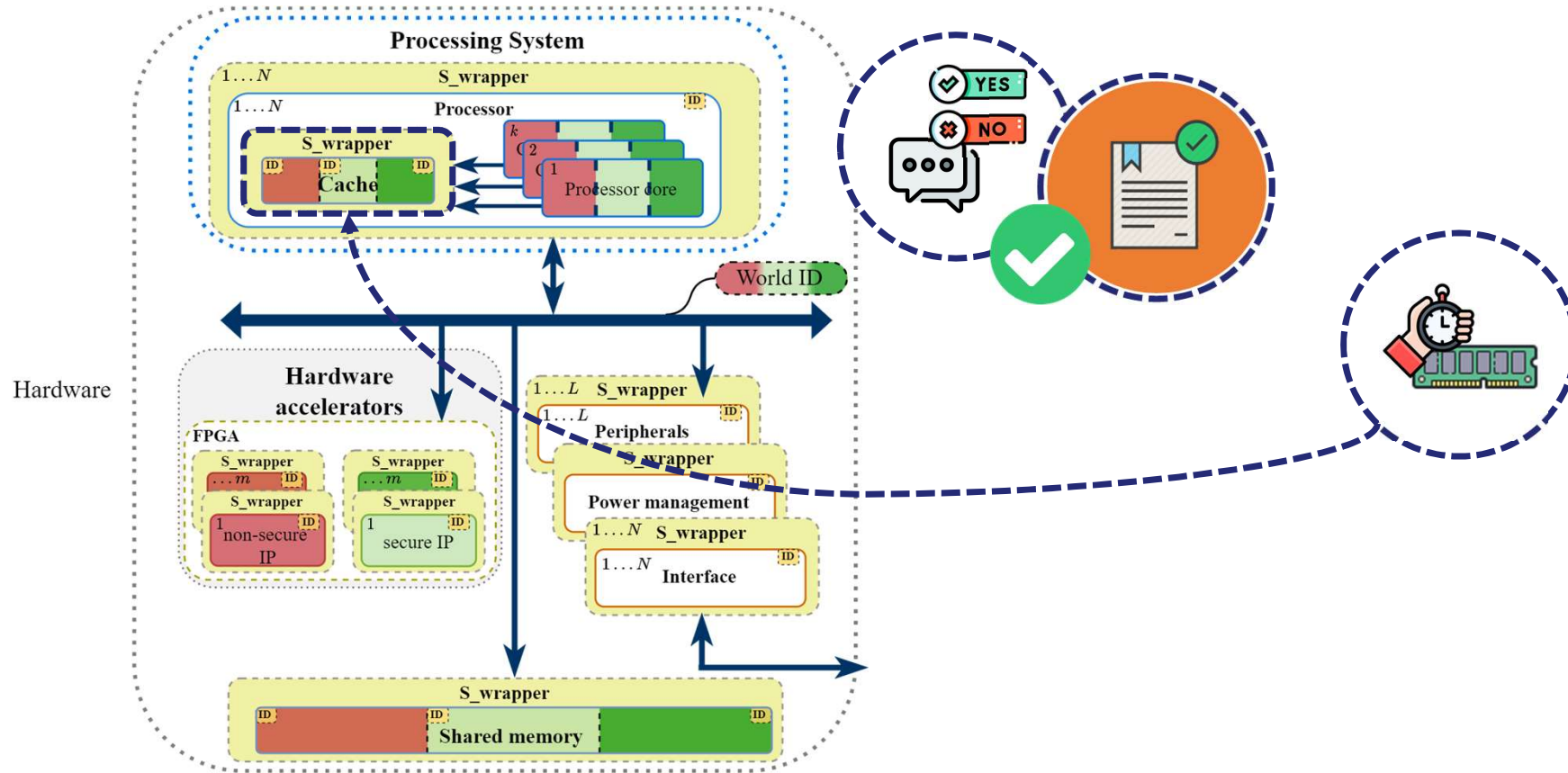


TrustSoC: architecture prototype





TrustSoC: architecture prototype



Implementation results

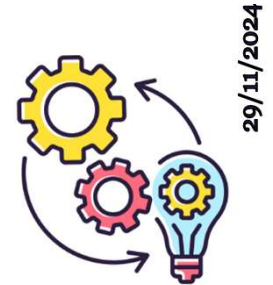
Protect six different hardware IPs from signal processing to cryptography

Mean overhead in LUTs: +0,34 %

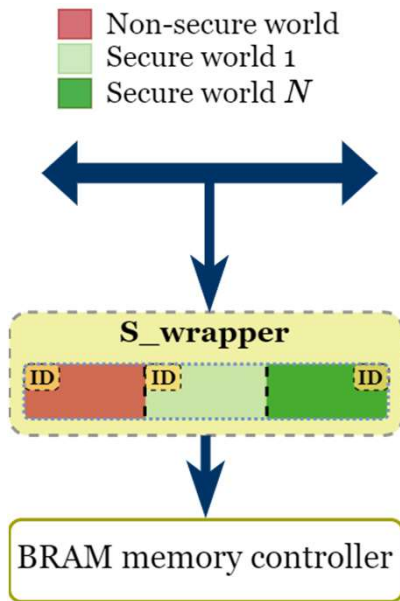
Mean overhead in FFs: +0,13 %

Mean maximum operating frequency : x

Our overheads are very small !



Implementation results on a BRAM



LUTs

Worlds \ Components	2	4	8	16
2	7	11	20	29
4	9	15	28	50
8	15	17	48	83
16	23	29	77	152
32	53	89	157	291

The overhead is directly linked to the permission tables

Contributions of *TrustSoC*



- **Software or hardware components** can be assigned to **different worlds** with **different privilege levels**
- A set of **distributed communication controllers** applies secure policies to have a **secure communication system inside the SoC**
- Introduction of the **trusted hardware IP** notion

Architecture	Type of processor	Threat model	Number of secure domains	Bus protections	Trusted hardware IPs	Protections against DoS attacks
<i>TrustSoC</i> [8]	ARM	Remote attacks only	N worlds	●	●	○

[8] Raphaële Milan, Lilian Bossuet, *et al.* "TrustSoC : Light and Efficient Heterogeneous SoC Architecture, Secure-by-design". AsianHOST 2023, Tianjin, China, December 2023.

RTrustSoC



RTrustSoC: threat model



29/11/2024

Remote attacks, by intern system blocks

- A corrupted software application that tries to access sensitive information of other software applications or hardware IP
- A corrupted hardware IP that tries to access sensitive information of other software applications or hardware IP
- Illegitimate accesses and modifications of the memory contents

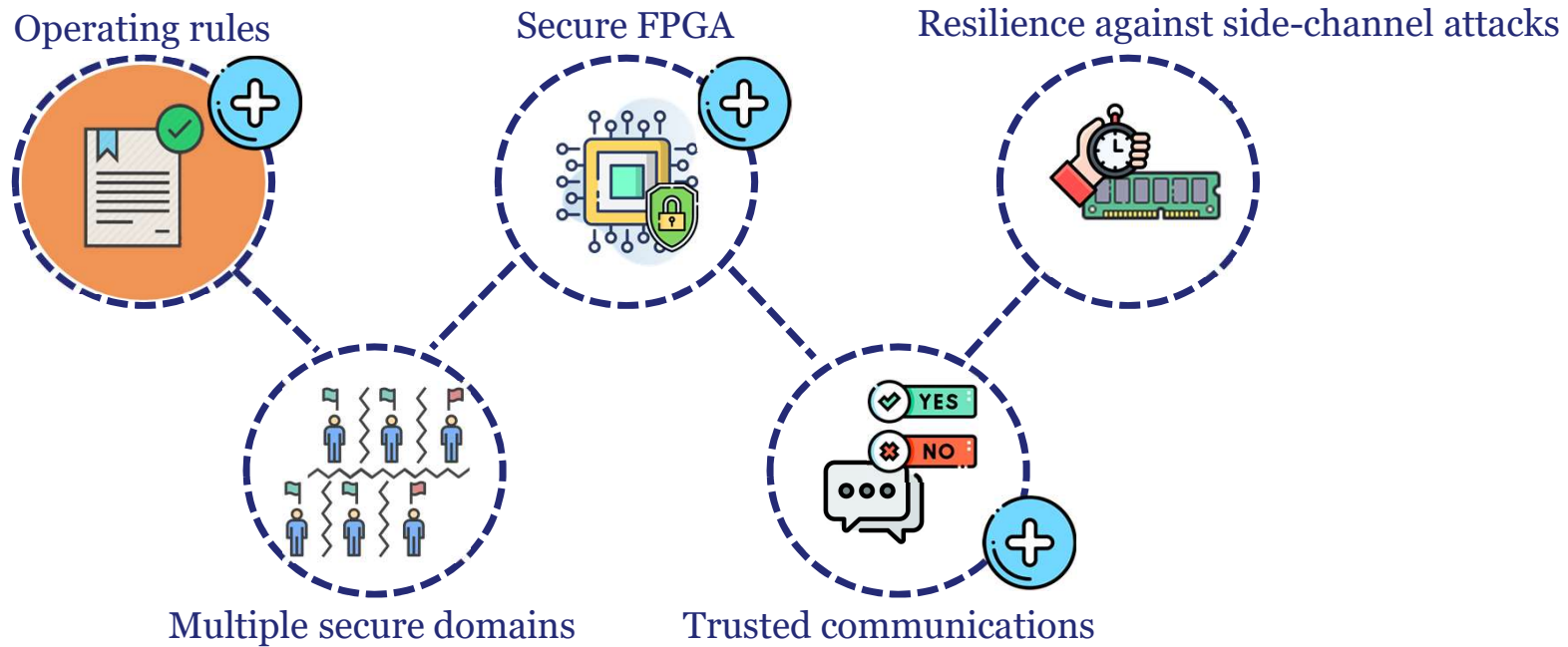


DoS attacks included

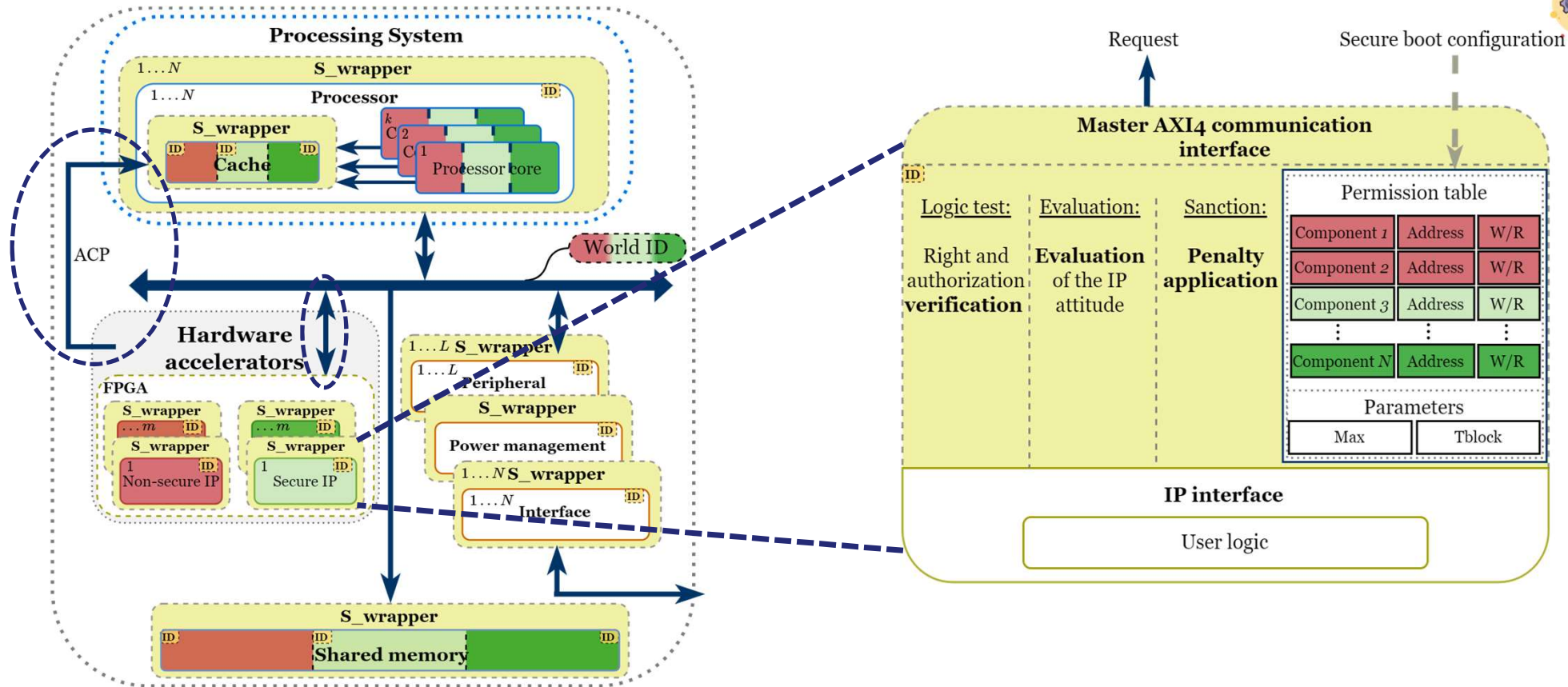


Compiler, foundry and CAD tool **trusted**

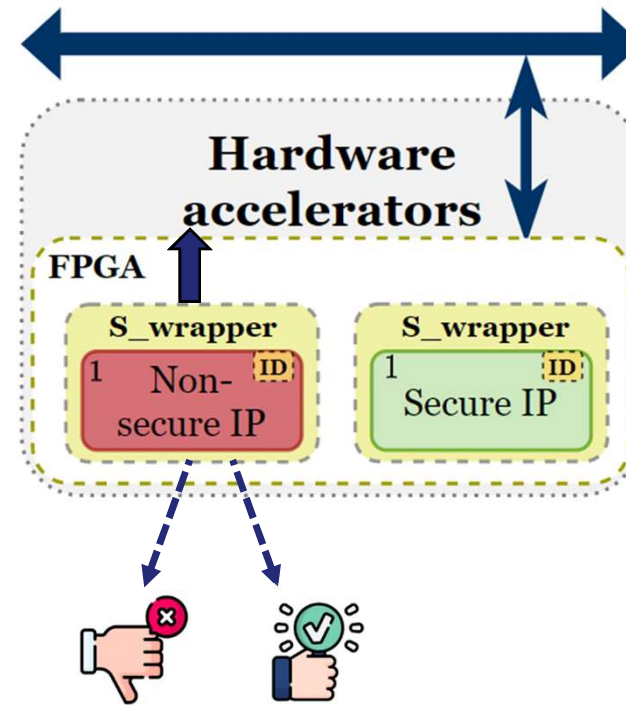
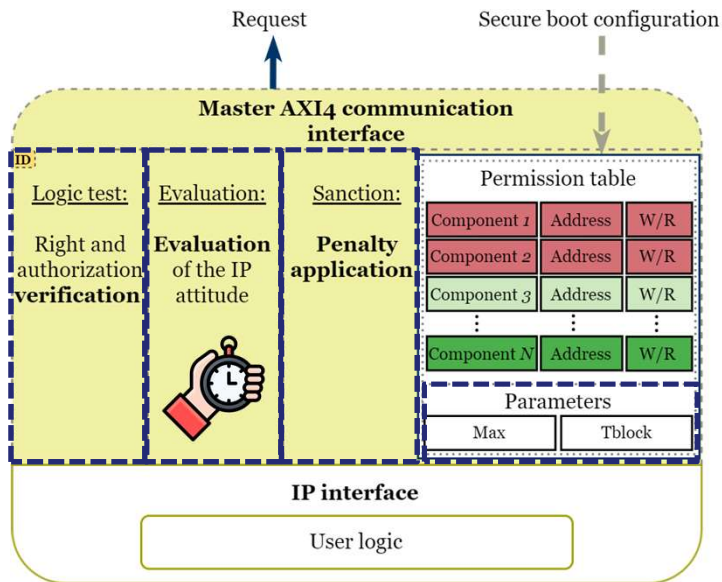
RTrustSoC: security features extended



RTrustSoC: architecture prototype



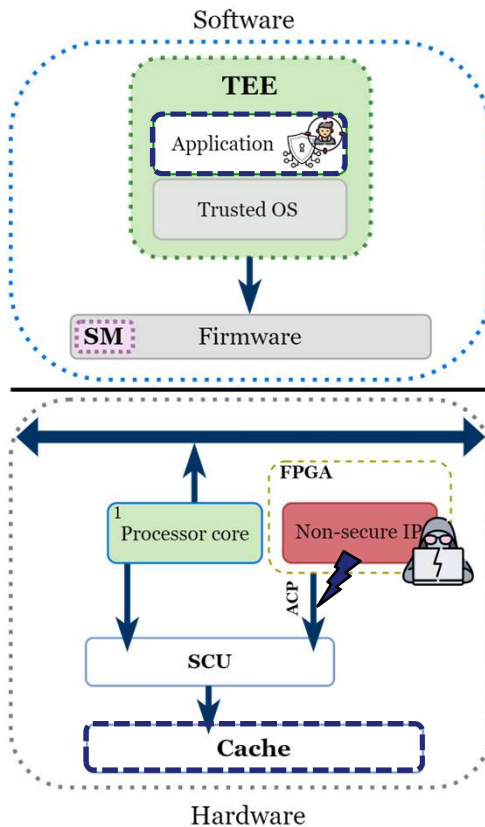
RTrustSoC: architecture prototype



RTrustSoC attack scenario



29/11/2024

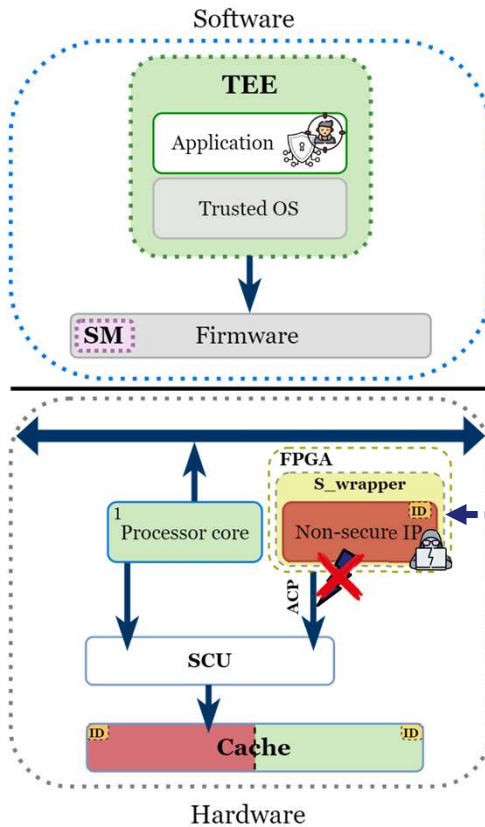


Without protection, the attacker [3] is able to recover the AES secret key

[3] L.Bossuet and E.M. Benhani., « Performing Cache Timing Attacks from the Reconfigurable Part of a Heterogeneous SoC—An Experimental Study ». In: Applied Sciences 11.14 (jan. 2021).



RTrustSoC attack scenario



Implementation results on SoC-FPGA AMD Zynq-7000

	LUT	FF	Fmax (Hz)
Controller ACP port	63	116	212
Utilization(%)	0.36	0.33	--

[4] R.Milan, L.Bossuet, *et al.*, «Efficient Adaptive Multi-level Privilege Partitioning with RTrustSoC». In: Transactions on Circuits And Systems I, 2024.

Contributions of *RTrustSoC*



- **Software or hardware** components can be **assigned to different worlds with different privilege levels**
- The **cache memory is protected** against attacks that exploit the shared access. **Demonstrated on a real-case scenario**
- **Extending the secure communication system** previously presented in *TrustSoC* inside the SoC **and the notion of trusted hardware IPs**

Architecture	Type of processor	Threat model	Number of secure domains	Bus protections	Trusted hardware IPs	Protections against DoS attacks
<i>TrustSoC</i> [8]	ARM	Remote attacks only	N worlds	●	●	○
<i>RTrustSoC</i> [4]	ARM	Remote attacks only	N worlds	●	●	●

TrustSoC-M

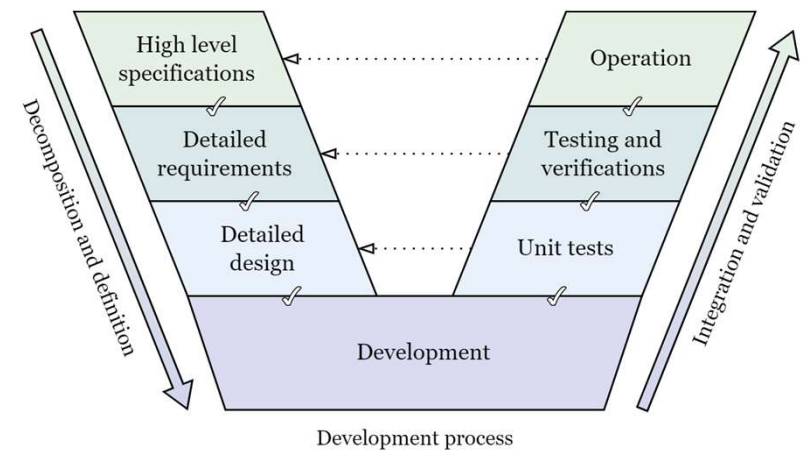


Model-Based System Engineering


MBSE puts models at the center of a system design

The model:

- Represents the system in a simpler way
- Eliminates some of its complexity with a more abstract representation
- Represents the same behavior and structure



Why MBSE method for *TrustSoC* ?

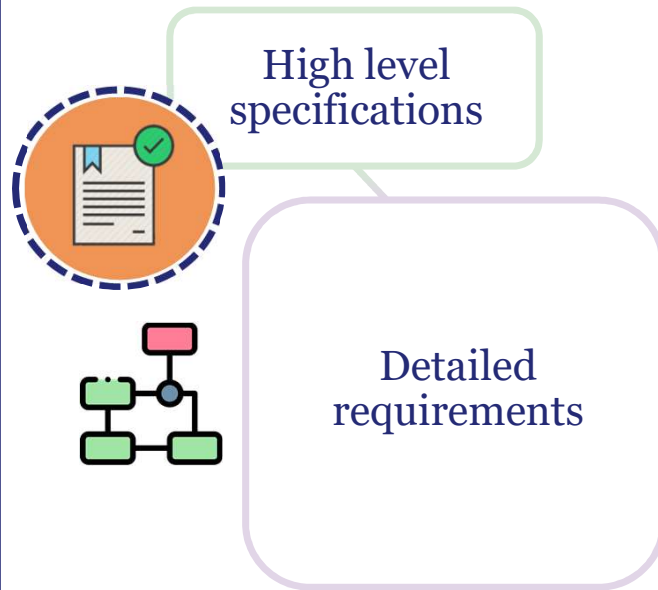
- To be able to **test different scenarios on the whole SoC architecture**, to **identify weaknesses** and have first cases to demonstrate the added value of our proposition 

- To **perform tests before implementing it on hardware** : quicker and very close to the hardware operation 

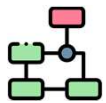
- **Explore the conception space** : be able to visualize the whole architecture and evaluate scenarios; **not restricted by the technology** 



TrustSoC-M approach

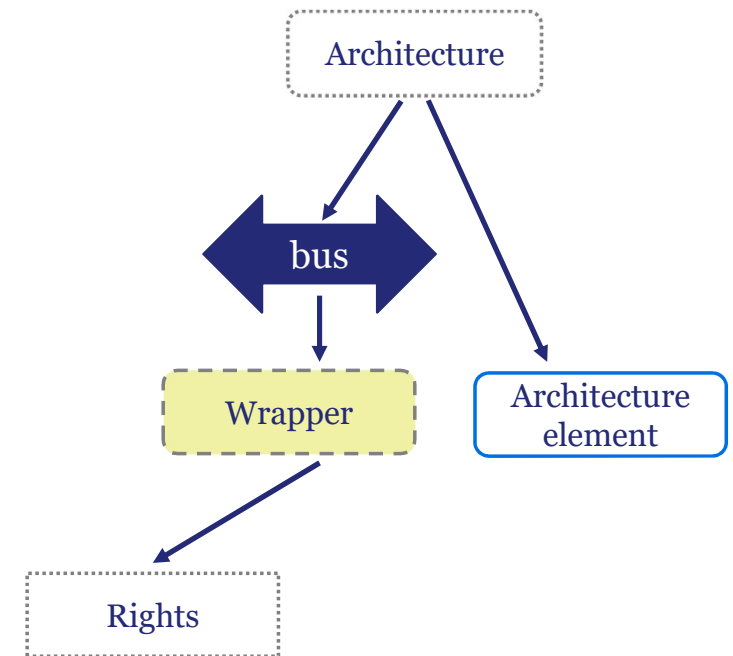


TrustSoC-M approach



Detailed requirements

Describe an **architecture prototype** as a model from **basic blocs**, **relationships** between these blocs and **behaviors**

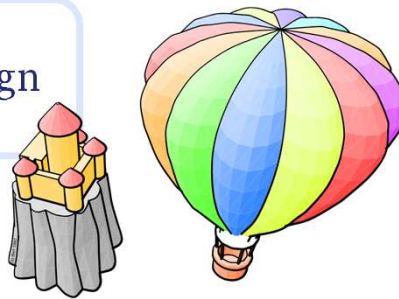


TrustSoC-M approach

High level specifications

Detailed requirements

Detailed design



Why **SmallTalk** ?

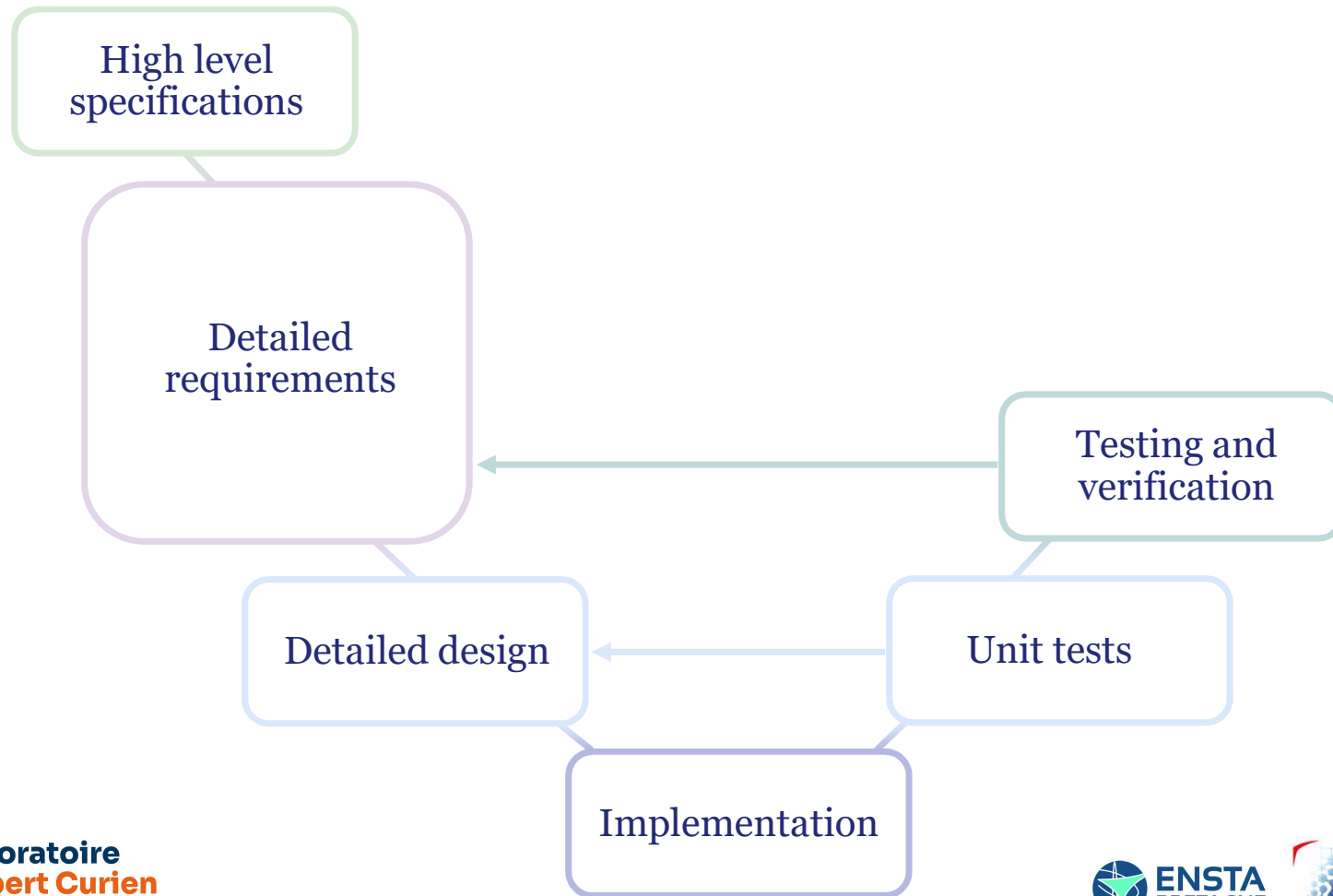
- Fast prototyping
- On-the-fly object modifications
- Inspect, modify and test in real-time during execution
- Add methods on-the-fly



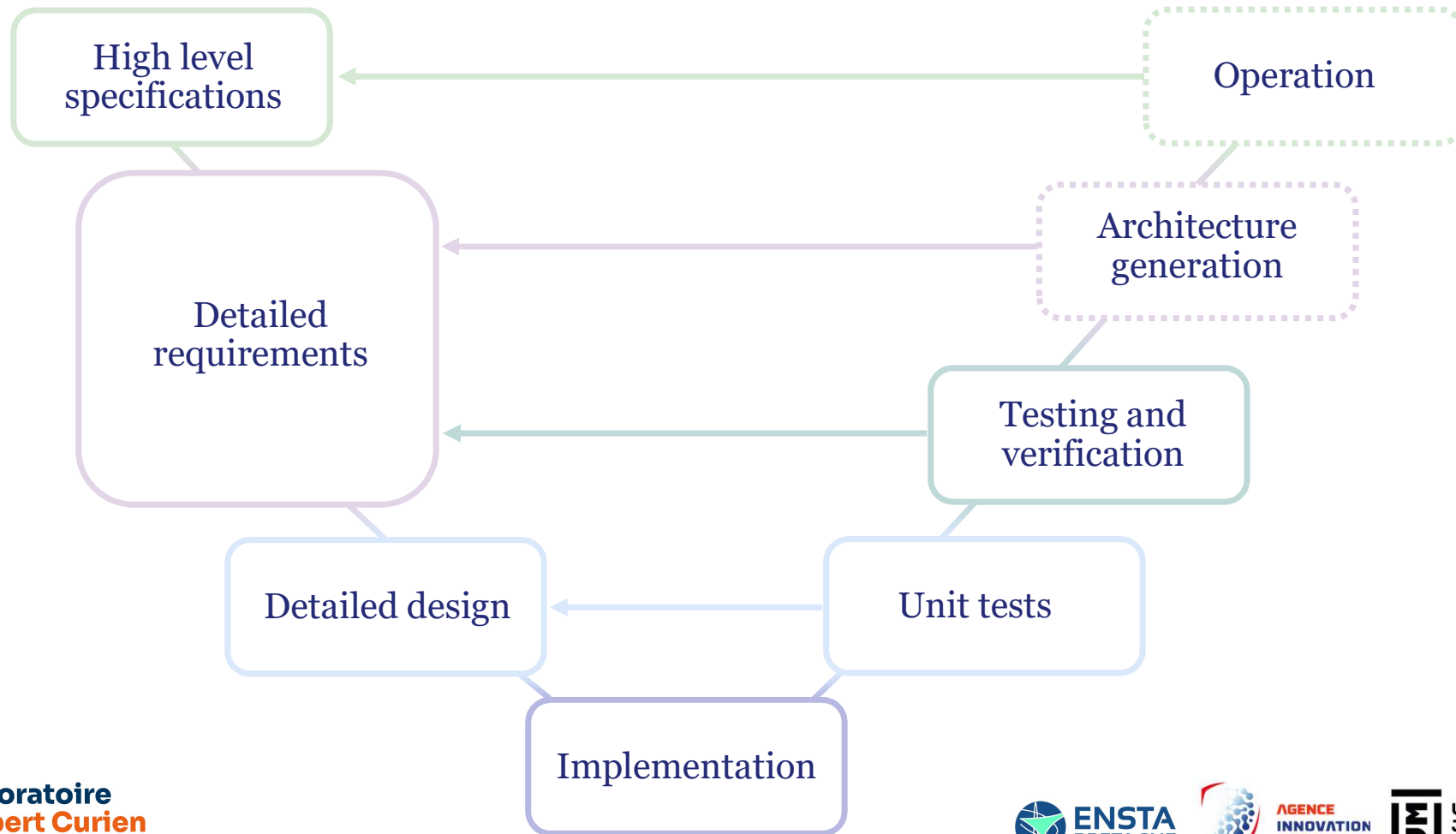
TrustSoC-M approach



29/11/2024

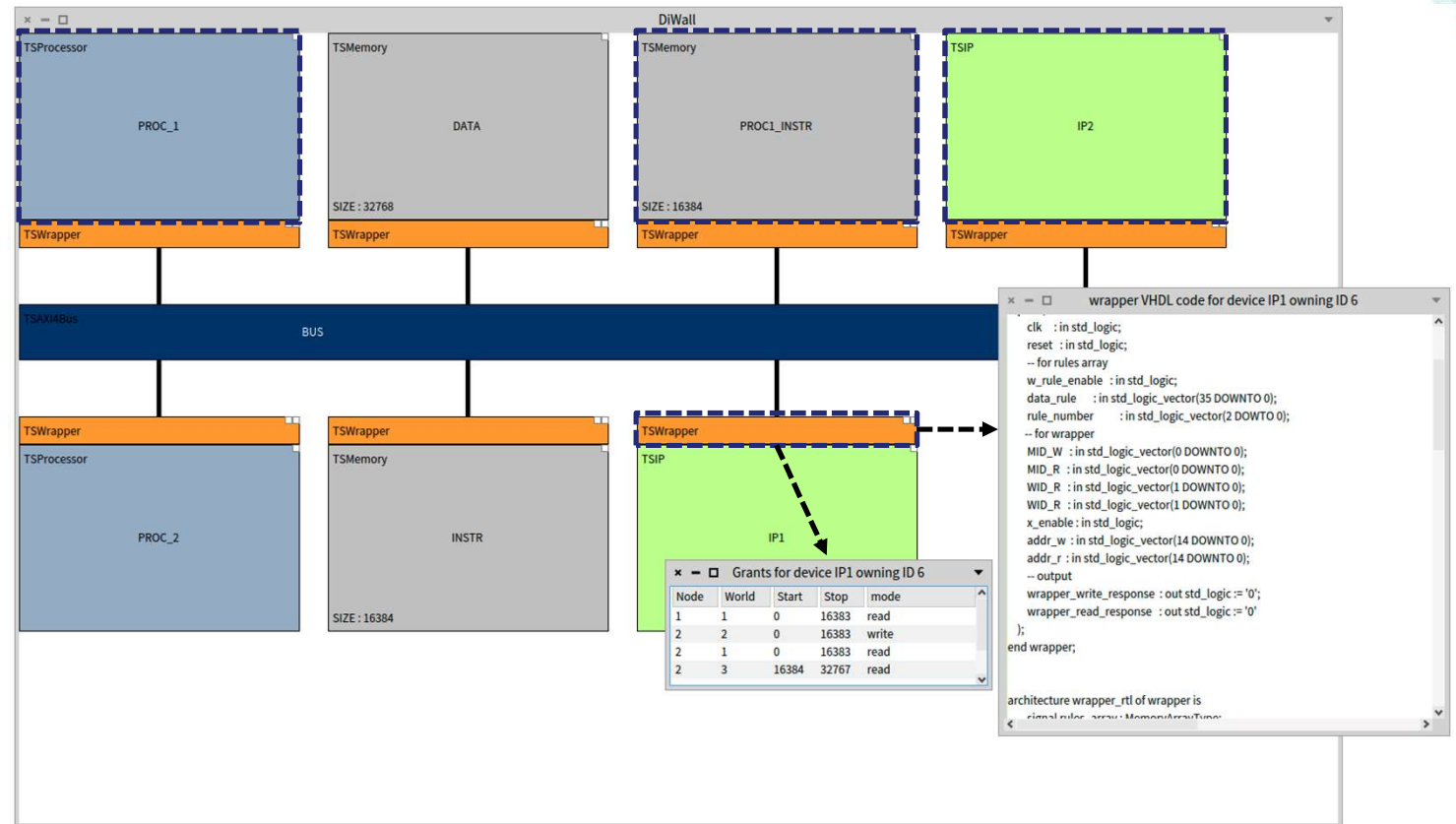


TrustSoC-M approach



TrustSoC-M DiWall

Architecture generation



Contributions of *TrustSoC-M*

- **Provide a modeling of the architecture *TrustSoC*** which removes the technological layer linked with the hardware implementation
- Restructuring *TrustSoC* architecture more quickly and more easily to **explore the conception space**
- Propose a tool to enable the designer to generate his own *TrustSoC* architecture





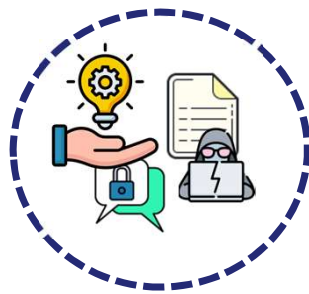
Conclusion and perspectives

Conclusion



29/11/2024

We have proposed an **architecture *TrustSoC*** that is **secured-by-design** and demonstrated our concept with an hardware prototype



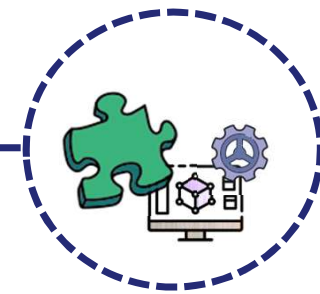
Security features and threat model defined



Provided with implementation results



Validation with attacks scenarios



Extension of the architecture application with modeling and generation

Perspectives



29/11/2024



Continue the modeling and the generation of the *TrustSoC* architecture and be able to provide a turnkey solution



Provide a modeling and the generation of the *TrustSoC* architecture with overlays [9][10] for Cloud applications



Provide a software prototype for *TrustSoC* architecture

[9] Théotime Bollengier, Loïc Lagadec *et al.* « Prototyping FPGA through overlays ». In : 2021 IEEE International Workshop on Rapid System Prototyping (RSP).

[10] Mohamad Najem *et al.* « Extended overlay architectures for heterogeneous FPGA cluster management ». In : Journal of Systems Architecture 78 (2017).

Communications



29/11/2024



Raphaële Milan, Lilian Bossuet, *et al.* "Efficient Adaptive Multi-level Privilege Partitioning with RTrustSoC". In IEEE Transactions on Circuits and Systems I : Regular Papers.



Raphaële Milan, Lilian Bossuet, *et al.* "Trust-SoC : Architecture SoC hétérogène légère et efficace sécurisée par conception". Conférence francophone d'informatique en Parallélisme, Architecture et Système (ComPAS 2023), Annecy, France, July 2023 + POSTER.

Raphaële Milan, Loïc Lagadec, *et al.* "Secured-by-design systems-on-chip : a MBSE Approach". RSP 2023, Hamburg, Germany, September 2023.

Raphaële Milan, Lilian Bossuet, *et al.* "TrustSoC : Light and Efficient Heterogeneous SoC Architecture, Secure-by-design". AsianHOST 2023, Tianjin, China, December 2023.



Raphaële Milan, Lilian Bossuet, *et al.* "TrustSoC-V: A RISC-V Heterogeneous SoC Architecture, Secure-by-Design". RISC-V Summit Europe 2024 (POSTER) + ACM SIGBED SRC 2023 ESWEEK.



Thank you



Appendix

Bibliography

- [1] A. Ltd, “TrustZone for Cortex-A – Arm®,” *Arm | The Architecture for the Digital World*. <https://www.arm.com/technologies/trustzone-for-cortex-a>
- [2] E. M. Benhani, L. Bossuet, and A. Aubert, “The security of arm trustzone in a fpga-based soc,” *IEEE Transactions on computers*, vol. 68, no. 8, 2019.
- [3] L.Bossuet and E.M. Benhani., « Performing Cache Timing Attacks from the Reconfigurable Part of a Heterogeneous SoC—An Experimental Study ». In: *Applied Sciences* 11.14 (Jan. 2021).
- [4] R.Milan, L.Bossuet, *et al.*, «Efficient Adaptive Multi-level Privilege Partitioning with RTrustSoC». In: *Transactions on Circuits And Systems I*, 2024.
- [5] Inc. SiFive. SiFive WorldGuard Technical Paper. 2.4., Santa Clara,CA, July 2021.
- [6] Pascal Nasahl et al. « HECTOR-V : A Heterogeneous CPU Architecture for a Secure RISC-V Execution Environment ». In : *AsiaCCS 2021*. ACM.
- [7] Raad Bahmani et al. « CURE : A Security Architecture with Customizable and Resilient Enclaves ». In: *USENIX*, August 2021.
- [8] Raphaële Milan, Lilian Bossuet, *et al.* "TrustSoC : Light and Efficient Heterogeneous SoC Architecture, Secure-by-design". *AsianHOST 2023*, Tianjin, China, December 2023.
- [9] Théotime Bollengier, Loïc Lagadec *et al.* « Prototyping FPGA through overlays ». In : *2021 IEEE International Workshop on Rapid System Prototyping (RSP)*.
- [10] Mohamad Najem *et al.* « Extended overlay architectures for heterogeneous FPGA cluster management ». In : *Journal of Systems Architecture* 78 (2017).

Icons

All the icons from the presentation are from icons8.com and flaticon.com