IEEE Hot Chips Symposium (HCS) 36, 25-27 Aug. 2024

A Trusted Execution Environment RISC-V System on Chip

Binh Kieu-Do-Nguyen, Khai-Duy Nguyen, Tuan-Kiet Dang, Cong-Kha Pham, and Trong-Thuc Hoang
University of Electro-Communications (UEC), Tokyo, Japan

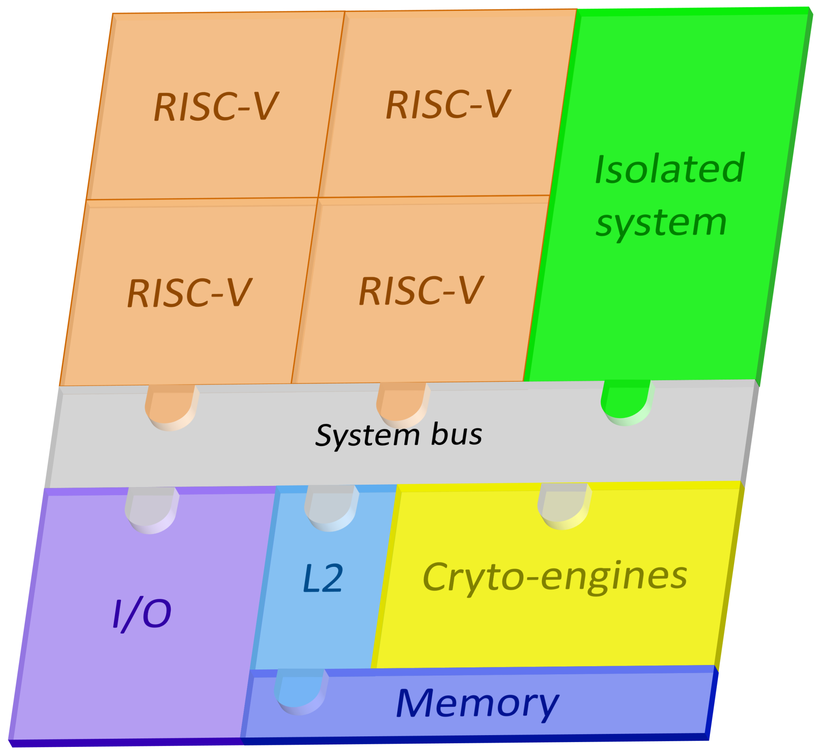


Figure 1 RISC-V system with Trusted Exectuion Environments (TEEs) hardware

INTRODUCTION

This work proposes a new open-source hardware framework for Trusted Execution Environments (TEEs) on RISC-V systems. The framework is designed to be secure, flexible, and easily upgradable. It includes various cryptographic accelerators and an isolated microcontroller to improve boot performance. The design was implemented and tested on VLSI platforms to demonstrate its feasibility and effectiveness.

Proposed system

- Trusted Execution Environment (TEE) includes optional RISC-V cores: Rocket, BOOM, Ibext, CVA6, etc.
- Proposed Isolated system includes RISC-V 32-bit IMC: Boot on reset and perform initial authentication and maintain Root of Trust (RoT).
- Proposed crypto-acclerators include: ECDSA, EdDSA, AES-GCM, SHA3-512, HMAC-SHA2, RSA, AEAD, Poly1305, ChaCha20, TRNG.
- Compatible with TLS1.3

SECURED BOOT FLOW

- **(M)** acts as root Certificate Authority and sign its certificate ($M_{Cert.}$) by itself. Using high-bit RSA (1024/2048-bits).
- **(R)** generates Elliptic Curve (EC) key pairs. Root certificate ($R_{Cert.}$) is signed using (M)'s secret key (S_M).
- **(D)** generates EC key pairs. Using ECDSA to sign for Device certificate ($D_{Cert.}$) based on (R)'s secret key.
- **(K)** generates EC key pairs. Using (D)'s secret key to sign for Program certificate ($K_{Cert.}$). (S_K) will be used to sign for TEE programs.

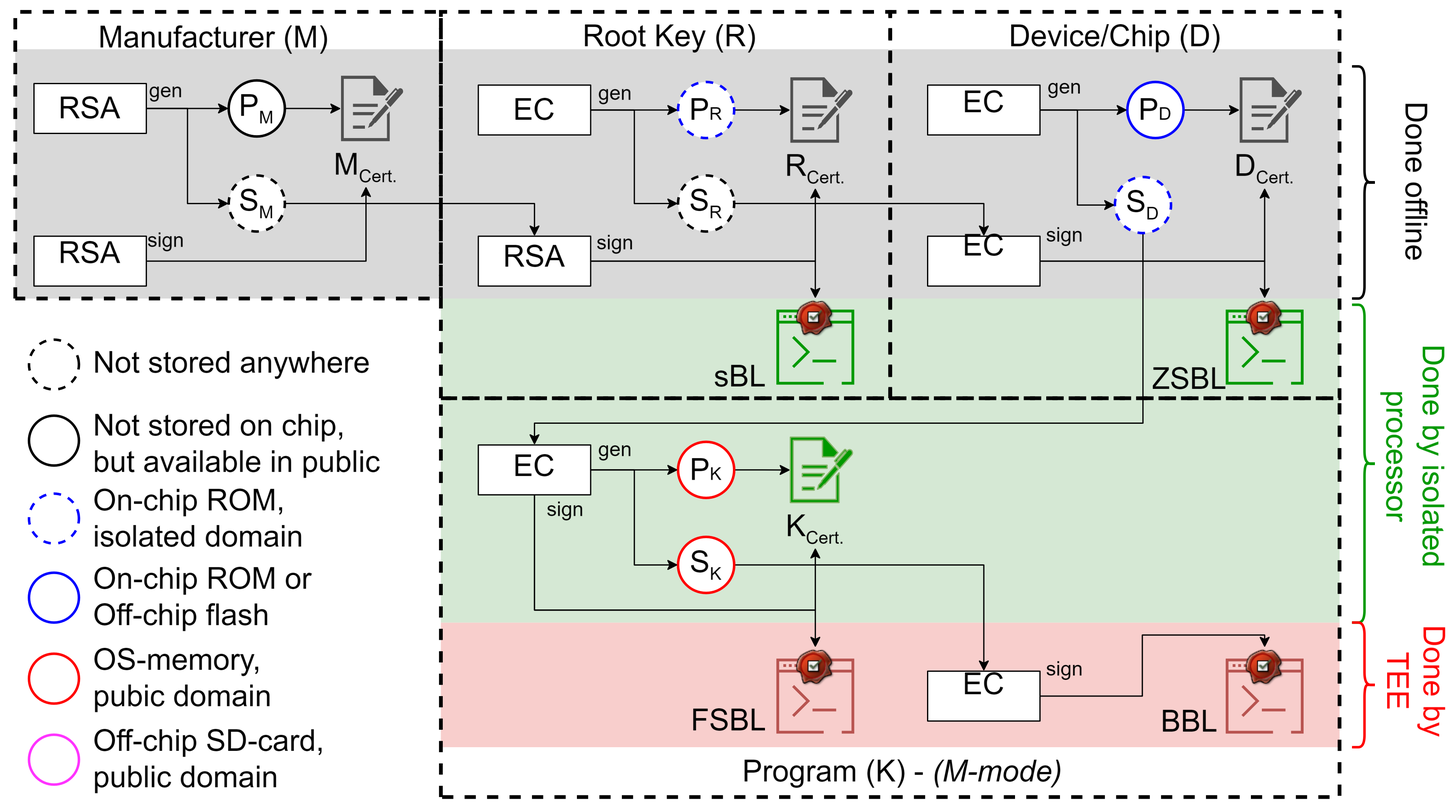


Figure 2 Key management in the secured boot procedure

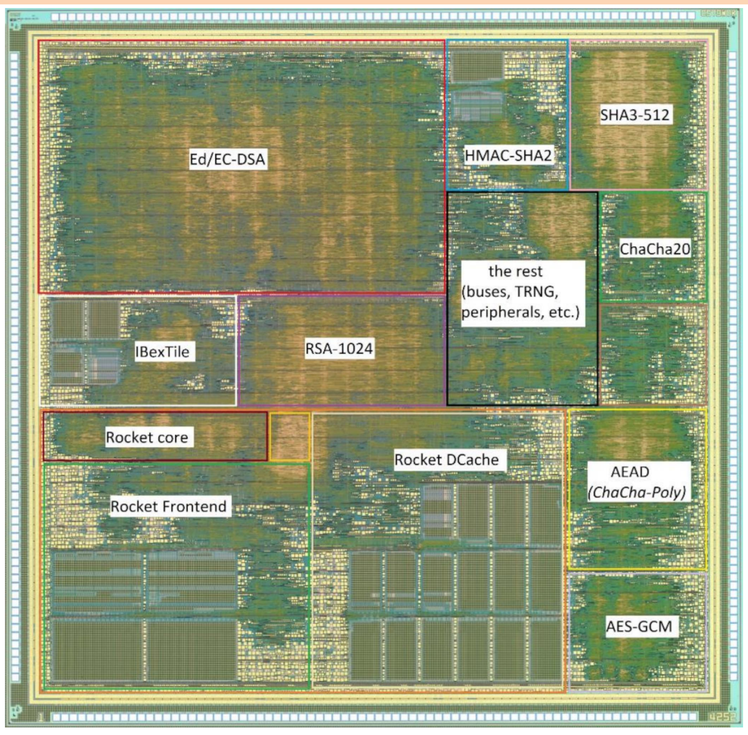
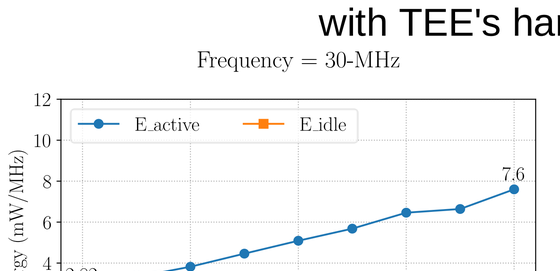


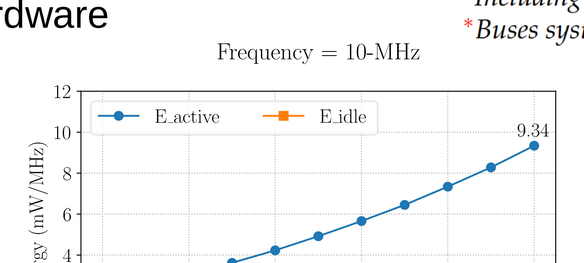
Figure 3 Die photo of dual-core RISC-V system with TEE's hardware

EVALUATION

ROHM 180-nm	Cell-count (NAND2)	Cell-area		Power			
		μm^2	%	Leakage (nW)	Dynamic (mW)	Total (mW)	%
Total system	460,195	14,744,115	100.00	5,487	3,075	3,075	100
Rocket	75,030	5,100,826	34.59	1,213	425	425	13.82
core	15,337	372,392	2.53	177	182	182	5.92
dcache	25,398	2,509,375	17.02	456	154	154	5.01
icache	32,127	2,169,710	14.72	555	77	77	2.50
IBex[1]	17,681	737,478	5.00	201	69	69	2.24
BootROM	4,272	70,672	0.48	21	11	11	0.36
ECED	166,720	3,638,115	24.68	1,664	1,311	1,311	42.63
RSA	35,754	827,563	5.61	385	226	226	7.35
AEAD	30,345	783,675	5.32	349	223	223	7.25
Chacha	16,723	402,309	2.73	178	85	85	2.76
Poly	10,966	308,602	2.09	136	89	89	2.89
SHA3	26,873	669,773	4.54	292	156	156	5.07
AES_GCM	20,753	532,594	3.61	266	80	80	2.60
HMAC-SHA2	13,155	529,278	3.58	176	92	92	2.99
TRNG	268	3,983	0.03	1	0.15	0.15	0.01
Other*	41,655	1,139,247	7.74	605	308	308	10.03

[1] *Including the isolated sub-system.*
* *Buses system, debug module, peripherals, interrupt.*





	CURE [1]	HECTOR-V [2]	WorldGuard [3]	ITUS [4,5]	This work
Open-source	○	○	◐	○	◐
Secure boot	◐	●	◐	●	●
Flexible boot process	●	●	●	○	●
TEE & secure boot iso.	○	○	○	●	●
Exclusive TEE processor	◐	●	◐	○	○
Exclusive secure storage	○	●	○	●	●
Secure I/O paths	●	●	◐	○	○
Crypto. accel.	○	○	◐	●	●
SCA resilience	●	●	◐	○	○
Hardware cost	●	◐	●	○	◐
High expressiveness	◐	●	◐	○	◐
Low porting efforts	○	○	◐	●	●

[1] R. Bahmani et al.: USENIX Security, Aug. 2021. **[2]** P. Nasahl, et al.: ASIA CCS, Jun. 2021.
[3] SiFive, Inc.. Securing The RISC-V Revolution. **[4]** V. B. Y. Kumar, et al.: SOCC, Sep. 2019.
[5] J. H.-Yahya, et al.: ISQED, Mar. 2019.

Corresponding author: Binh Kieu-Do-Nguyen (binh@vlsilab.ee.uec.ac.jp)