Quantum-Resilient IoT: Integrating Hardware-Based Post-Quantum Cryptography for Robust Device Security

Stephan Spitz[1], Alexander Lawall[1] and Michal Andrzejczak[2]
[1] *IU International University of Applied Science, Juri-Gagarin-Ring 152, 99084 Erfurt, Germany*
[2] *Resquant, Narutowicza 40 / 1, 90-135 Lodz, Poland*

Keywords: Silicon-Integrated Post-Quantum Cryptographic Cores, Applied Post-Quantum Cryptography, Internet of Things (IoT) Security, Industrial Internet of Things (IIoT) Security.

Abstract: The evolution of quantum computers necessitates the reevaluation of cryptographic standards, especially within the Internet of Things (IoT) infrastructures, where long-term security is critical. Current cryptographic algorithms, such as RSA, are vulnerable to quantum attacks, highlighting the need for post-quantum cryptographic (PQC) solutions. This paper explores the integration of PQC Cores into System-on-a-Chip (SoC) architectures to enhance the security of IoT devices. The foundation is a crypto-agile Root-of-Trust (RoT), these integrated PQC solutions provide robust lifecycle management, secure boot processes, and protection against quantum-based threats. The paper discusses the architectural considerations for integrating PQC, including secure boot, lifecycle management, and the role of RoT in ensuring device integrity and secure communications. The research findings emphasize the importance of PQC in safeguarding IoT infrastructures from emerging quantum threats and demonstrate how hardware-based PQC implementations offer superior security compared to software-based counterparts, particularly in the context of side-channel attack mitigation.

1 INTRODUCTION

The evolution of quantum computers will demand an update of security mechanisms, which are fundamental in communication protocols and IT processes. This is especially valid for mechanisms built on cryptographic primitives (Tiwari et al., 2024). There exists, for example, the Shor-Algorithm (Skosana, 2021) for the factorization of large primes that solves the foundational challenge on which the RSA (Rivest Shamir Adleman) algorithm (Rivest et al., 1978) has been built. Similarly, currently used asymmetric ciphers are affected once quantum computers offer sufficient computing capabilities that are based on qubits. For breaking the RSA cipher with the Shor-Algorithm, at least 4000 qubits are required for a 2048-bit key (Roetteler et al., 2017) whereas topical quantum computers have just surpassed the 1000 qubits (Quantum, 2023). For this reason, RSA ciphers with 512-bit key length can no longer be considered secure.

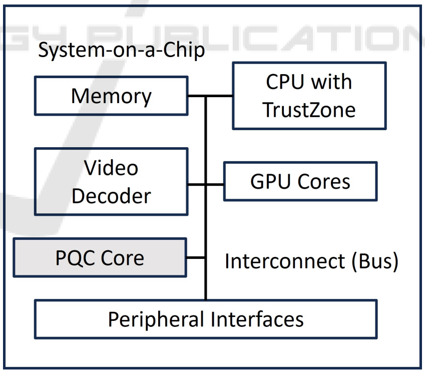

Figure 1: Integration of a Post-Quantum Cryptographic Core in a System-on-a-Chip.

2 THE TECHNOLOGY STACK

This section describes the foundation for linking security services on the application level to an integrated Post-Quantum Cryptographic (PQC) Core (Zagala and Andrzejczak, 2024) in a System-on-a-Chip (SoC). A dedicated technology stack is required linking the PQC Core via a processor bus to security services running in a TrustZone (Arm, 2018) or another

Spitz, S., Lawall, A. and Andrzejczak, M.
Quantum-Resilient IoT: Integrating Hardware-Based Post-Quantum Cryptography for Robust Device Security.
DOI: 10.5220/0013091100003899

In *Proceedings of the 11th International Conference on Information Systems Security and Privacy (ICISSP 2025) - Volume 2*, pages 419-424
ISBN: 978-989-758-735-1; ISSN: 2184-4356

Security Enclave on the main CPU, cf. Figure 1.

An integrated PQC Core needs to interact with secure software that is executed in a protected environment such as TrustZone. This ensures that only a higher privileged and protected process can use the PQC algorithms and that cryptographic keys are sufficiently protected.

2.1 Crypto Support of Secure Services

Concurrency of the security services is necessary because services executed in the richOS or Real-Time OS (RTOS) require parallel access to PQC crypto routines and cryptographic keys (Spitz, 2012). Consequently, the new security subsystem including the PQC Core must support high bandwidth communication, multitasking and multithreading. Secure concurrent communication over the Internet Protocol (IP), e.g. Transport Layer Security (TLS) demands the handshake asynchronous encryption and signature verification processes performed in the Secure OS. Large vendors such as IBM and Google are already working on PQC-enabled TLS Cipher Suites (Pursche et al., 2024), (Westerbaan, 2024) in their products. In contrast, the Internet Engineering Task Force (IETF) is currently working on standardizing TLS with hybrid modes (Stebila et al., 2024), where classical cryptography is used in combination with post-quantum algorithms.

2.2 Secure Boot

Another important aspect is the PQC Core integration in the boot process, cf. Figure 2. The SoC-integrated PQC Core is booted together with the SoC firmware before the richOS/RTOS starts execution. This results in a security-critical dependency of the entire boot process on the PQC Core. The authenticity of the firmware and communication stack must be validated during the SoC's Secure Boot process to ensure that no altered or compromised software or firmware is loaded by a potential attacker. Moreover, the verification procedure must be reliable and verified. Thus, a full hardware implementation (based only on dedicated circuits without any program) of PQC algorithms used in Secure Boot for signature verification must be performed. The circuit responsible for this part is verified with formal methods in the pre-silicon stage and is the security foundation for Secure Boot.

The involved keys and code require the following protection:

- Integrity protection of the PQC public keys for code integrity verification

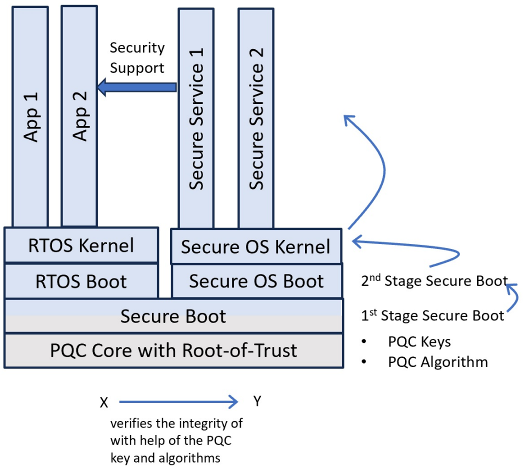

Figure 2: Staged Verification during the Boot with the PQC Core providing the Foundation.

- Integrity protection of the code doing the verification, which is part of the first stage bootloader
- Monotonic Counters which are used to invalidate old firmware and allow updates of the second-stage bootloader code

The entire SoC must exhibit tamper resistance, with particular emphasis on the cryptographic core. Similar to Smart Card semiconductors, security-critical components must be safeguarded through active countermeasures such as metal shielding, scrambled data buses, or sensors designed to detect attacks. These requirements introduce significant challenges to chip architecture and standard silicon fabrication processes.(Arm, 2018). A challenge for PQC is side-channel protection where additional leakage (e.g. thermal, timing and power) can be used to compromise used keys. To face this threat, several additional countermeasures are adopted, slowing down algorithms up to several times (Migliore et al., 2019). However, hardware implementations of PQC offer more ways to implement countermeasures compared to software implementations and are also more efficient. The generic SoC manufacturing processes must be enhanced to establish the necessary cryptographic and security infrastructure, particularly for the Secure Boot mechanism (IAR, 2018). The establishment of the Root-of-Trust (RoT) serves as the foundational security element for Secure Boot and all subsequent security processes, including the deployment and loading of the Secure OS, as well as individualization and penalization procedures. As personal devices like a wearable or smartphone get into the hands of users at a later stage, accounts are personalized in an insecure environment. The RoT ensures a reliable device identity and enables attestation services to bootstrap user identities and user authentication in the field.

3 ARCHITECTURAL CONCEPTS

3.1 Considerations on PQC Integration

SoC-integrated security solutions have an impact on the whole technology stack, which is built in bare metal (Spitz and Lawall, 2024). One important aspect is the reentrance of the PQC Core to support multiple cryptographic operations on the software layer. In the operating system, multitasking enables secure encrypted and authenticated asynchronous high-bandwidth communication (Safe, 2024). There are different possibilities to connect an IoT device securely to the Internet. The operating system architecture must account for the integration of the PQC Core, providing appropriate drivers and protocols to facilitate its interaction. At higher software layers, asynchronous communication is managed through a mailbox mechanism, enabling the exchange of large volumes of data between security-critical and non-critical processes within a designated memory space. Another security-critical piece of functions closely linked to the PQC Core is system calls, which are mapped to the cryptographic support in hardware. The entry point is the OS Kernel which contains the necessary routines for accessing the PQC Core e.g. for verifying the integrity of applications that are scheduled for execution by the OS Kernel. An example of such a service is the Samsung TIMA (Trusted Integrity Management Architecture) (Snyder, 2019) on Android smartphones. TIMA issues system calls to the services that are securely executed by the Secure OS running in the TrustZone of the SoC. One reason for such system calls is the run-time integrity protection of the OS Kernel itself and security-critical applications to avoid fraudulent modifications in the code, e.g. to bypass rights management. Typically, the Secure OS is loaded from flash memory in an early boot stage and initialized with the cryptographic keys stored in the PQC Core. Especially with Dilithium the key length of up to 4864 bytes for the private key has to be considered when storing keys in the secret memory of the PQC Core. Nevertheless, sufficient flash memory seems not to be an issue on modern SOCs whereas the SRAM (i.e. Cortex-M4) reaches in some cases its limitations during the execution of Dilithium with these key lengths. Large key sizes can be a challenge for more constrained devices. Long-term storage techniques for generating keys from master seed might be applied, but this won't solve issues with a lack of MCU's internal memory. In that case, specific implementation techniques or dedicated hardware circuits should be developed. The main OS Kernel is booted after the complete security subsystem consisting of the Secure OS, Secure Services and the PQC Core have been initialized. Thus, attacks on the OS Kernel can hardly circumvent the Secure OS or tamper the PQC Core in hardware. The essential master keys, integrity protection and verification mechanisms are part of the PQC Core because this offers protection in hardware (Zagala and Andrzejczak, 2024).

3.2 Overall SoC Security Architecture

A single PQC Core is insufficient for ensuring the security of an SoC. Strong hardware-based isolation mechanisms are crucial, such as Arm TrustZone or a dedicated secure processor core like the ARC SEM (Synopsys, 2016). Additionally, security-critical SoC components, including memory and I/O peripherals, can be isolated from standard processing elements and accessed exclusively in a secure execution mode.

Countermeasures against fault injection and side-channel attacks are essential in any hardware security solution. Since generic silicon fabrication processes offer limited support for such protections, the security features embedded in hardware and firmware become increasingly critical. Firmware plays a key role in safeguarding access to the PQC Core with security boundaries established during platform boot. Integrity checks must be performed for all low-level drivers and firmware components, particularly for the driver interfacing with the PQC Core. A staged secure boot process is needed to initialize all secure processing elements before activating standard components, ensuring that standard operations do not compromise security settings or access to critical cryptographic keys and functions.

4 LIFECYCLE MANAGEMENT OF INTEGRATED SECURITY SOLUTIONS

4.1 The Role of a Root-of-Trust

A robust RoT, integrated with the PQC Core, serves as the foundational security anchor for software lifecycle management and update processes (IAR, 2018). Lifecycle management encompasses the deployment, modification, and complete removal of security-critical code and data, particularly information related to device identity and access control roles. This identity-related data may pertain to the device itself or the roles governing access management.

Furthermore, a RoT is essential for establishing a secure communication channel with the device, enabling the provisioning of identities even when the device operates in an untrusted environment. Leveraging the RoT, an authenticated and secure channel can be established, facilitating the secure download of sensitive data, such as during maintenance operations, even when the device is already deployed in the field. The RoT provides the cryptographic foundation required for securely binding the device to an external trusted entity through a cryptographic handshake. A RoT can be based on the following asymmetric and symmetric keys:

- Crypto-Agility with a symmetric master key pre-seeded for exchange of the asymmetric keys and algorithms
- XMSS/LMS or SPHINCS+ public key for boot chain integrity verification
- CRYSTALS-Dilithium private key for Remote Attestation
- CRYSTALS-Dilithium public key for Remote Authentication. XMSS is best suited for BLOB updates.
- CRYSTALS-Kyber public key for a secure key exchange of a temporary symmetric key for Data Confidentiality

The RoT can be also located outside the PQC Core in the protected memory of the SoC. The RoT must be established in a secure process in a secure environment. At least, the AES256 (National Institute of Standards and Technology, 2001) secret must be pre-seeded with the first part of the bootloader. A staged secure bootloader incorporating the RoT is responsible for the integrity of the whole SoC and all the software executed on the device. Such a bootloader is typically organized in a first and second stage, cf. Figure 3.

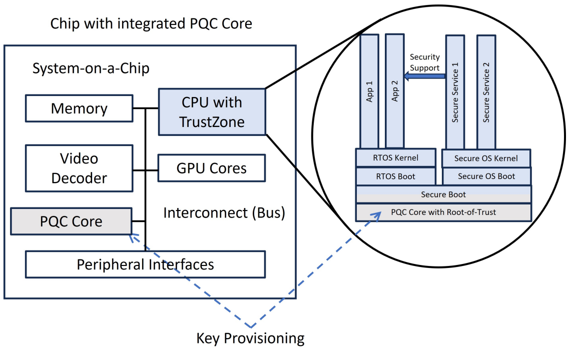

Figure 3: Key Provisioning for seeding a Root-of-Trust.

The PQC Cores have the necessary capabilities to protect the PQC master keys and at least one symmetric secret, preferably an AES256 symmetric key. This symmetric key has to be protected in the best possible way in the device and the infrastructure for device management. Furthermore, it should be device individual, because it allows the exchange of the other asymmetric PQC keys in the PQC Core. Lattice-base cryptography is relatively new in the field and first attacks on implementations have already been discovered(IAR, 2018). Utilizing a symmetric key allows for the exchange of a compromised key or a transition from lattice-based algorithms, such as CRYSTALS-Dilithium or Kyber, to SPHINCS+, a hash-based asymmetric algorithm. It is worth it to recap that the public PQC keys just require integrity protection because this information needs not to be kept confidential. Private and symmetric keys should be device individual and highly confidentiality protected.

4.2 Lifecycle Management Operations

For the secure lifecycle management of an IT system, the following aspects matter:

- Integrity verification including remote identity proof, cf. (Sundar et al., 2019)
- Protection of the code and data during loading and runtime, especially protection of the Secure OS, from fraudulent modifications cf. (Wang et al., 2019)
- User and Access Management including the option to assign individual rights to selected users. This starts with the establishment of initial user accounts with the necessary authentication methods.
- AA-functionality (Authentication/Authorization) of configuration changes, code updates, especially security-relevant settings
- Delegation of control to third parties, especially with the change of ownership
- Secure disabling of single accounts or the complete system e.g. for end-of-life, over-production control, and grey market prevention
- Initial setup of secure communication channels for different kinds kind of lifecycle operations, especially updates or configuration changes

The processes mentioned above contain all the following basic cryptographic routines:

- a) Identification of the IoT device
- b) Authorisation of a lifecycle action e.g. software update
- c) Authentication of a user or even an individual task

- d) Ensuring secrecy of code or data in transit and rest

4.2.1 Identification of a System-on-a-Chip

In scenario a), the RoT may include an identity and a private key, which are securely stored in the PQC Core during the manufacturing process, as illustrated in Figure 4. Subsequently, the Secure OS within the device receives a request to authenticate the device and attest its identity. This request is accompanied by a random number which the Secure OS signs using the private key stored in the RoT. The verifying entity external to the device can then decrypt the response utilizing the corresponding public key. The device successfully demonstrates its identity if the decrypted result matches the initial random challenge.

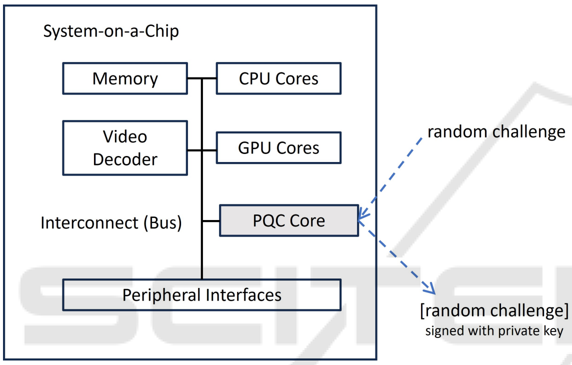

Figure 4: Device Identity Verification.

4.2.2 Authorisation and Authentication of an Administrative Action

In scenarios b) and c), the roles are reversed, with the RoT containing a public key, as depicted in Figure 5. In this case, the external entity possesses a private key, enabling it to authorize administrative actions, deploy code, or perform read/write operations within the Secure OS. The issuer of the administrative action is automatically identified in this scenario, as the private key can be uniquely associated with an individual, legal entity, or IT system. This association is established through a Public Key Infrastructure (PKI). It is important to note that, in this context, the public key on the device must be protected against modifications and unauthorized exchanges, although confidentiality protection is not a requirement. The RoT must ensure integrity protection, allowing only the Secure OS on the device to have read access to this public key.

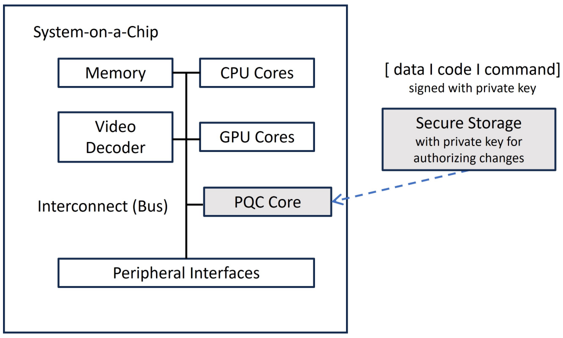

Figure 5: Authorisation of Remote Management.

4.2.3 Confidentiality Protection of Data

Scenario d) necessitates the use of a hybrid encryption scheme, which combines both symmetric and asymmetric encryption methods. This is due to the performance limitations of asymmetric algorithms, which are not well-suited for encrypting large volumes of data, as illustrated in Figure 6. In this hybrid approach, a symmetric key is temporarily generated and transmitted alongside the encrypted data to the device. The symmetric key itself is further encrypted using the public key of the device, ensuring its confidentiality during transmission. The Secure OS, which has access to the device's private key stored within the RoT, performs two critical operations: first, it decrypts the encrypted symmetric key using its private key, and second, it employs the decrypted symmetric key to encrypt and decrypt the data. This scenario is particularly pertinent when transmitting personalization data to the device, such as subscriber profiles for integrated SIM cards (ETSI, 2017) or payment credentials for Integrated Secure Elements (iSEs). The use of a hybrid encryption scheme enhances security while maintaining efficiency for scenarios requiring the secure handling of sensitive information. By leveraging the strengths of both encryption methods, this approach not only safeguards data confidentiality but also ensures that the process remains computationally feasible, thus enabling seamless operations in resource-constrained environments.

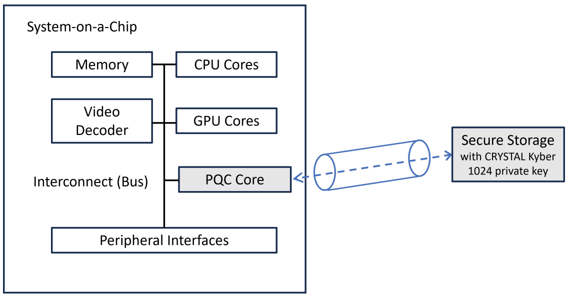

Figure 6: Confidentiality Protection.

5 CONCLUSION

Due to the advances of quantum computers, System-on-a-Chip manufacturers need to consider designing a Post-Quantum Cryptographic Core that offers long-term sustainability for the security of the lifecycle management of the whole IoT device. It is important to implement crypto agility with exchangeable PQC keys and algorithms which can be leveraged to protect for a long-term all relevant services running on the IoT device. The PQC Core forms the foundation of a modern Root-of-Trust which protects the IoT devices against many kinds of attacks. To fully leverage protection capabilities, it is essential to integrate the PQC Core across the entire chain, from hardware to the software services running on the IoT device's operating system and within the supporting infrastructure. There should be no weak element in this chain and integrity verification is an important aspect to ensure this. Last but not least, PQC is more vulnerable to sophisticated side-channel attacks than the outdated classical asymmetric ciphers such as RSA and ECC. However, recent developments show that it is possible to implement mechanisms to achieve tamper resistance also with PQC Cores (Zagala and Andrzejczak, 2024).

REFERENCES

Arm (2018). Cortex-m35p a tamper-resistant cortex-m processor with optional software isolation using trustzone for armv8-m. In *https://developer.arm.com/products/processors/cortex-m/cortex-m35p*. Arm.

ETSI (2017). iuicc poc group primary platform requirements, approved release. In *https://www.gsma.com/newsroom/wp-content/uploads/UIC.03_v1.0.pdf*. ETSI.

IAR (2018). Building a supply chain of trust: Understanding secure mastering. In *https://www.iar.com/support/resources/articles/secure-mastering*. IAR.

Migliore, V., Gérard, B., Tibouchi, M., and Fouque, P.-A. (2019). Masking dilithium. In Deng, R. H., Gauthier-Umaña, V., Ochoa, M., and Yung, M., editors, *Applied Cryptography and Network Security*, pages 344–362, Cham. Springer International Publishing.

National Institute of Standards and Technology (2001). Advanced encryption standard (aes). FIPS Publication 197, U.S. Department of Commerce.

Pursche, M., Puch, N., Peters, S. N., and Heinl, M. P. (2024). SoK: The engineer's guide to post-quantum cryptography for embedded devices. Cryptology ePrint Archive, Paper 2024/1345.

Quantum, I. (2023). The quantum decade: Ibm's quantum roadmap to 2033. Accessed: 2024-08-19.

Rivest, R. L., Shamir, A., and Adleman, L. (1978). A method for obtaining digital signatures and public-key cryptosystems. *Communications of the ACM*, 21(2):120–126.

Roetteler, M., Naehrig, M., Svore, K. M., and Lauter, K. (2017). Quantum resource estimates for computing elliptic curve discrete logarithms. In *International Conference on the Theory and Applications of Cryptology and Information Security (ASIACRYPT 2017)*, pages 241–270. Springer.

Safe, O. Q. (2024). Tls. In *https://openquantumsafe.org/applications/tls.html*. Resquant.

Skosana, T. (2021). Demonstration of shor's factoring algorithm for n=21 on ibm quantum processors. In *2021 Scientific Article number: 16599*, pages 1–4. Springer.

Snyder, J. (2019). Samsung trusted boot and trustzone integrity management explained. In *"https://insights.samsung.com/2019/09/04/samsung-trusted-boot-and-trustzone-integrity-management-explained/"*. Samsung.

Spitz, S. (2012). Mobicore® secure os for arm® trustzone® soc. In *https://prezi.com/rgrvv8vv-t4s/mobicore-secure-os-for-arm-trustzone-soc/*. Prezi.

Spitz, S. and Lawall, A. (2024). Silicon-integrated security solutions driving iot security. In *Proceedings of the 10th International Conference on Information Systems Security and Privacy - Volume 1: ICISSP*, pages 398–402. INSTICC, SciTePress.

Stebila, D., Fluhrer, S., and Gueron, S. (2024). Hybrid key exchange in TLS 1.3. Internet-Draft draft-ietf-tls-hybrid-design-10, Internet Engineering Task Force. Work in Progress.

Sundar, S., Yellai, P., Sanagapati, S. S. S., Pradhan, P. C., et al. (2019). Remote attestation based software integrity of iot devices. In *2019 IEEE International Conference on Advanced Networks and Telecommunications Systems (ANTS)*, pages 1–4. IEEE.

Synopsys (2016). Designware arc sem security processors. In *https://www.synopsys.com/dw/ipdir.php?ds=arc-sem*. Synopsys.

Tiwari, A., Chauhan, R., Joshi, N., Devliyal, S., Aluvala, S., and Kumar, A. (2024). The quantum threat: Implications for data security and the rise of post-quantum cryptography. *2024 IEEE 9th International Conference for Convergence in Technology (I2CT)*, pages 1–7.

Wang, W., Zhang, X., Hao, Q., Zhang, Z., Xu, B., Dong, H., Xia, T., and Wang, X. (2019). Hardware-enhanced protection for the runtime data security in embedded systems. *Electronics*, 8(1):52.

Westerbaan, B. (2024). The state of the post-quantum internet. Accessed: 2024-09-05.

Zagala, S. and Andrzejczak, M. (2024). Post-Quantum Cryptography IP Cores.