



Beyond Post-Quantum Security

Physical Attacks and Countermeasures

Dr. Melissa Azouaoui

Principal Cryptographer
Competence Center Cryptography & Security
NXP Semiconductors

Introduction

- Melissa Azouaoui
- Principal Cryptographer
- Technical Lead for PQC Long Term Innovation



Oct 2017

April 2021

Jan 2026

PhD

@ NXP, Germany & UCLouvain, Belgium

Elliptic Curve Cryptography

Physical Attacks & Countermeasures

Cryptographer

@ NXP, Germany

Many aspects of **PQC**

Mostly **secure implementations**

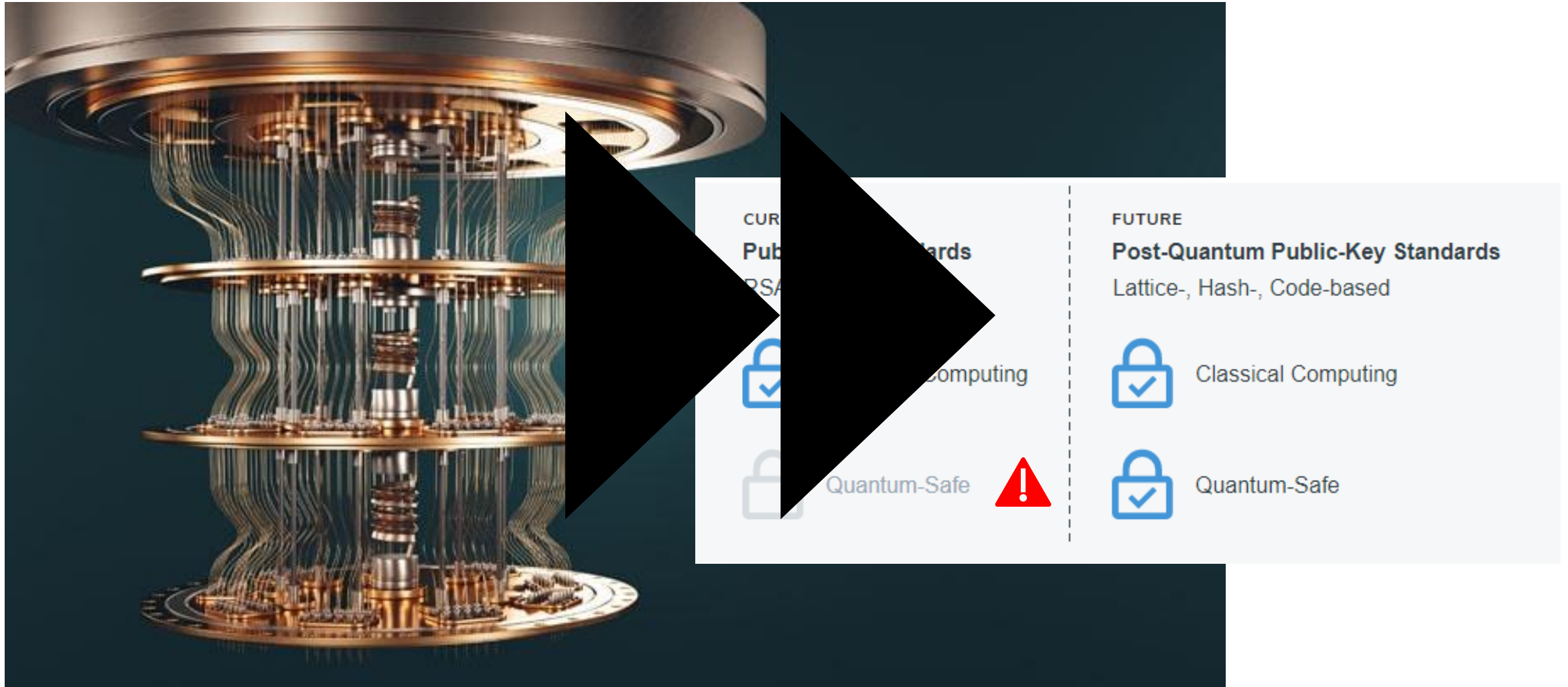
Technical Lead for PQC Long Term Innovation

@ NXP, Germany

Notable Inventor (10+ patents)

Recently, **Secure AI** too

Post-Quantum Cryptography





Automotive



Industrial & IoT



Mobile



**Communication
Infrastructure**



Automotive



eGovernment



Bank cards



**Smart mobility
(MIFARE) cards**



**Tags &
Authentication**



Readers



Mobile

What is the impact of PQC on the billions of embedded devices?

The PQC migration path



Standards and regulations



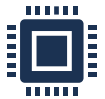
Digital infrastructures



Protocols



Software



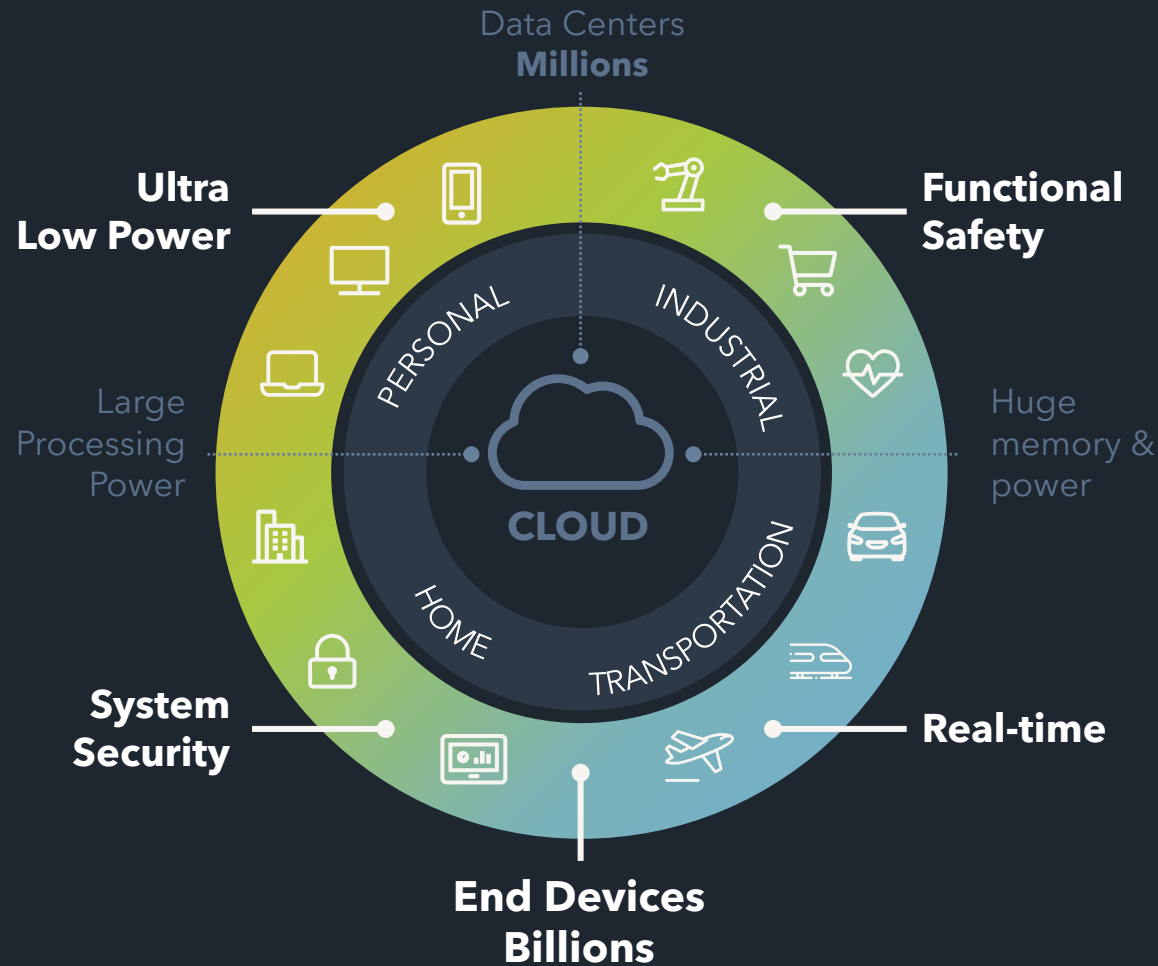
Hardware



Cryptography



Implementation Challenges for Embedded Devices



Data collection, processing and decisions at the edge
Devices securely connected to the cloud

No Silver Bullet

No one-size-fits-all solution. Some schemes are not suited for particular use cases & device constraints.

Cryptographic Keys

Orders of magnitude larger.

Bandwidth & Power

Larger signatures (up to 49.8KB).

Performance

Varies: some faster, some significantly slower.

Memory

Orders of magnitude more:
up to 100KB memory of RAM when executing.

Security

Implementation attacks: side-channel and fault injection.
Countermeasures can imply significant performance overheads.

New algorithms and standards



FIPS 203
Federal Information Processing Standards Publication

**Module-Lattice-Based
Key-Encapsulation Mechanism Standard**

Category: Computer Security Subcategory: Cryptography

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.FIPS.203>

Published August 13, 2024

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

Key Exchange / Encapsulation

FIPS 204
Federal Information Processing Standards Publication

**Module-Lattice-Based Digital
Signature Standard**

Category: Computer Security Subcategory: Cryptography

Information Technology Laboratory
National Institute of Standards and Technology
Gaithersburg, MD 20899-8900

This publication is available free of charge from:
<https://doi.org/10.6028/NIST.FIPS.204>

Published August 13, 2024

U.S. Department of Commerce
Gina M. Raimondo, Secretary

National Institute of Standards and Technology
Laurie E. Locascio, NIST Director and Under Secretary of Commerce for Standards and Technology

Digital Signatures
(generic)

FIPS 205
Federal Information Processing Standards Publication

**Stateless Hash-Based Digital Signature
Standard**

NIST Special Publication 800-208

**Recommendation for Stateful
Hash-Based Signature Schemes**

David A. Cooper
Daniel C. Apon
Quynh H. Dang
Michael S. Davidson
Morris J. Dworkin
Carl A. Miller

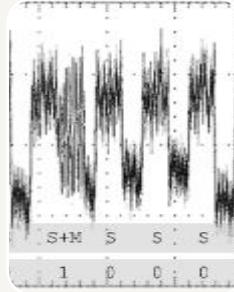
This publication is available free of charge from:
<https://doi.org/10.6028/NIST.SP.800-208>

Digital Signatures
(software / firmware signing)

More ongoing and upcoming! FIPS 206, Round 4, On-Ramp, ISO, etc..

- [1] ML-KEM, <https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.203.pdf>
- [2] ML-DSA, <https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.204.pdf>
- [3] SLH-DSA, <https://nvlpubs.nist.gov/nistpubs/fips/nist.fips.205.pdf>
- [4] LMS / XMSS, <https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-208.pdf>

Resistance against physical attacks



Side-channel attacks

- Power analysis (SPA, DPA)
- Electromagnetic analysis (SEMA, DEMA)
- Timing Analysis
- Photo-emission microscopy (high-end)
- Profiled, unprofiled and ML-assisted variants



Fault injection attacks

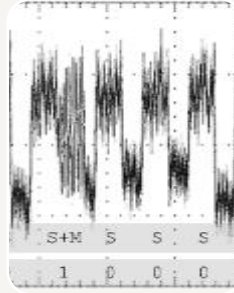
- Voltage or clock glitching
- Electromagnetic fault injection (EMFI)
- Body bias injection
- Laser fault injection
- Single and multi-shot scenarios



Invasive attack

- Focused Ion Beam (FIB) modifications
- Micro/Nano-probing of internal signals
- Signal forcing
- Delaying
- Reverse-engineering

Resistance against physical attacks



Side-channel attacks

- Power analysis (SPA, DPA)
- Electromagnetic analysis (SEMA, DEMA)
- Timing Analysis
- Photo-emission microscopy (high-end)
- Profiled, unprofiled and ML-assisted variants



Fault injection attacks

- Voltage or clock glitching
- Electromagnetic fault injection (EMFI)
- Body bias injection
- Laser fault injection
- Single and multi-shot scenarios



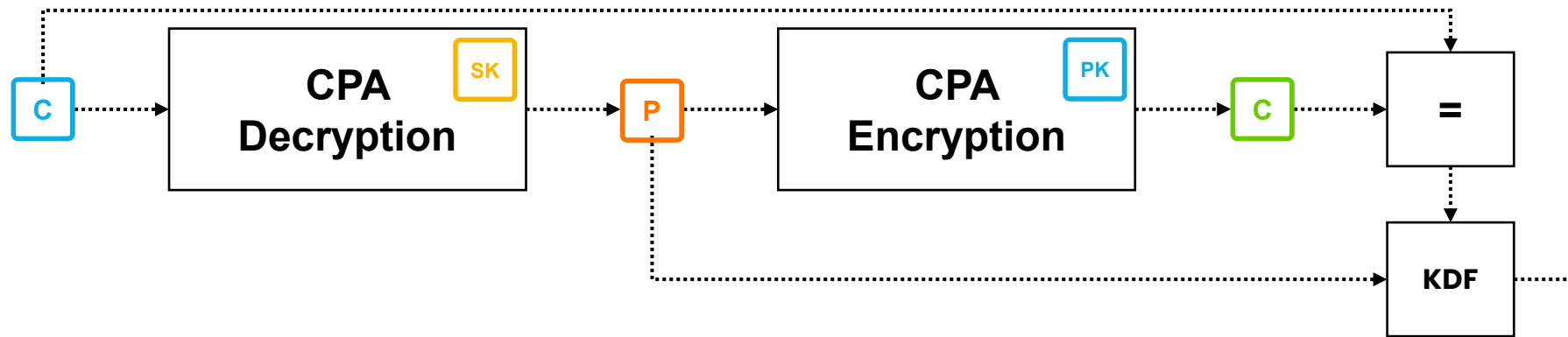
Invasive attack

- Focused Ion Beam (FIB) modifications
- Micro/Nano-probing of internal signals
- Signal forcing
- Delayering
- Reverse-engineering

The FO-calypse



Fujisaki Okamoto transform



Transform a scheme which achieves **IND-CPA**
("chosen plaintext attack") security to reach **IND-CCA**
("indistinguishability against chosen-ciphertext attacks") security

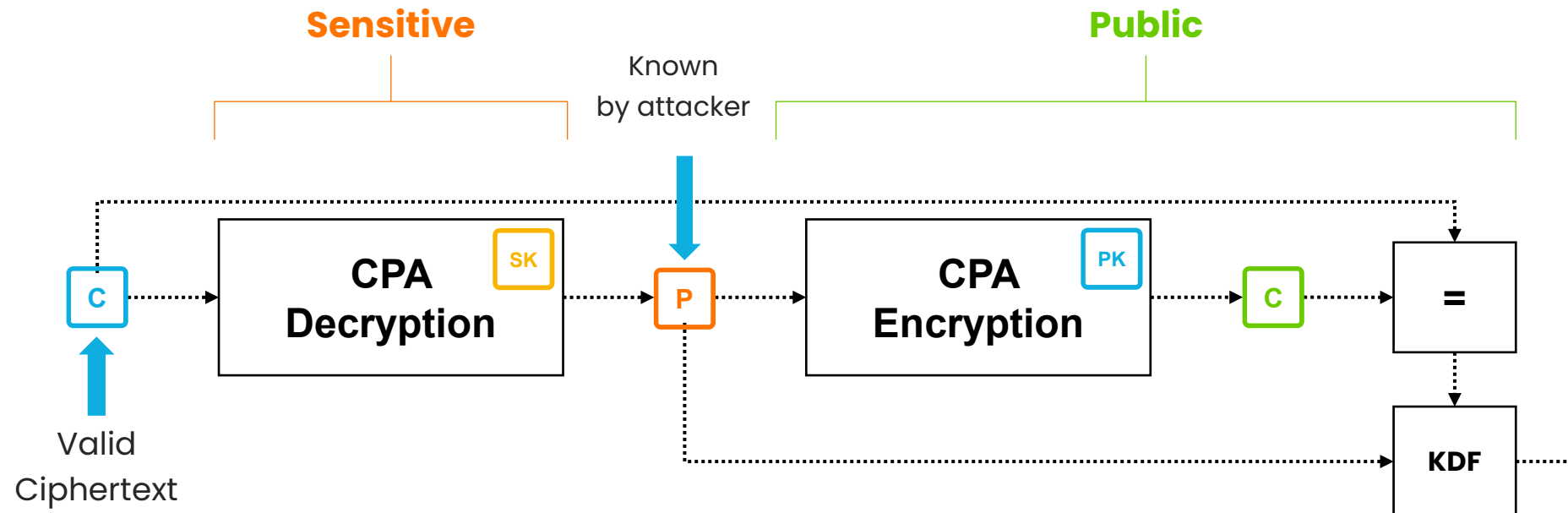
Fujisaki, E. and Okamoto
T., Secure integration of
asymmetric and symmetric
encryption schemes, CRYPTO
1999 and JoC 2013

The Side-Channel Attack Problem of the FO-Transform



Attack 1: Chosen Plaintext

- Attacker inputs only valid ciphertexts
- Attack focuses on **CPA Decryption**, everything after (and including) **P** is public
- Only need to protect **CPA Decryption**

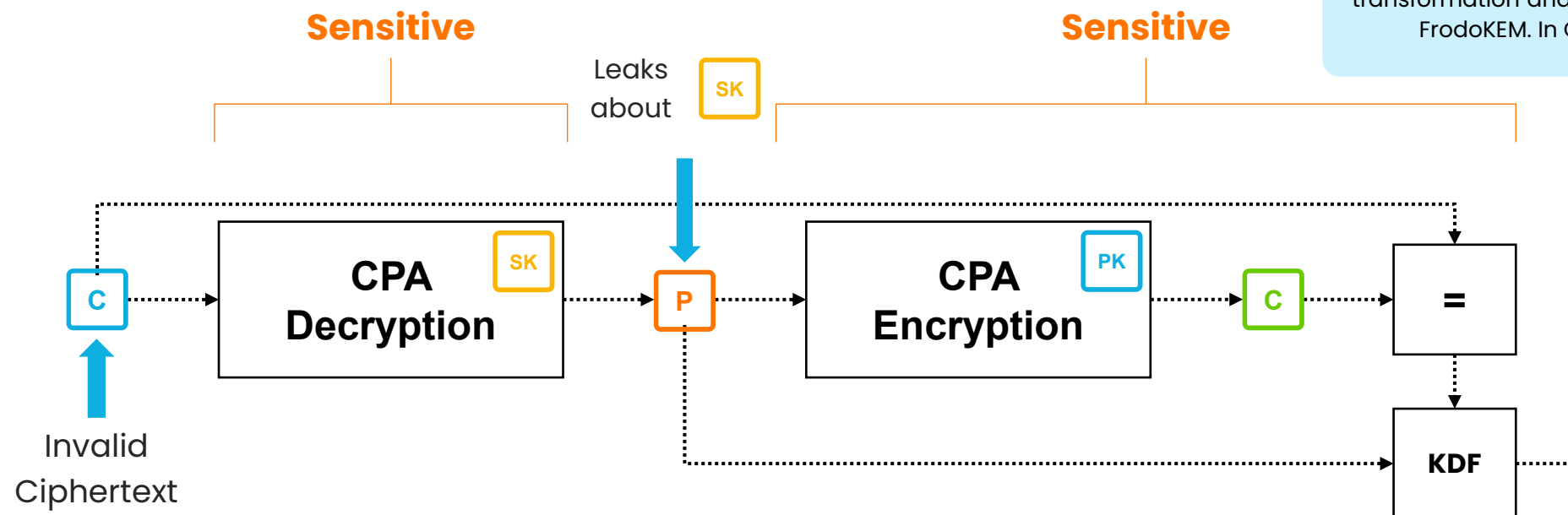


The Side-Channel Attack Problem of the FO-Transform



Attack 2: Chosen Ciphertext

- Attacker inputs specially-crafted invalid ciphertexts
- Attack focuses on **CPA Decryption** + everything after (and including) **P** is potentially sensitive
- Potentially all (or most) modules need to be hardened

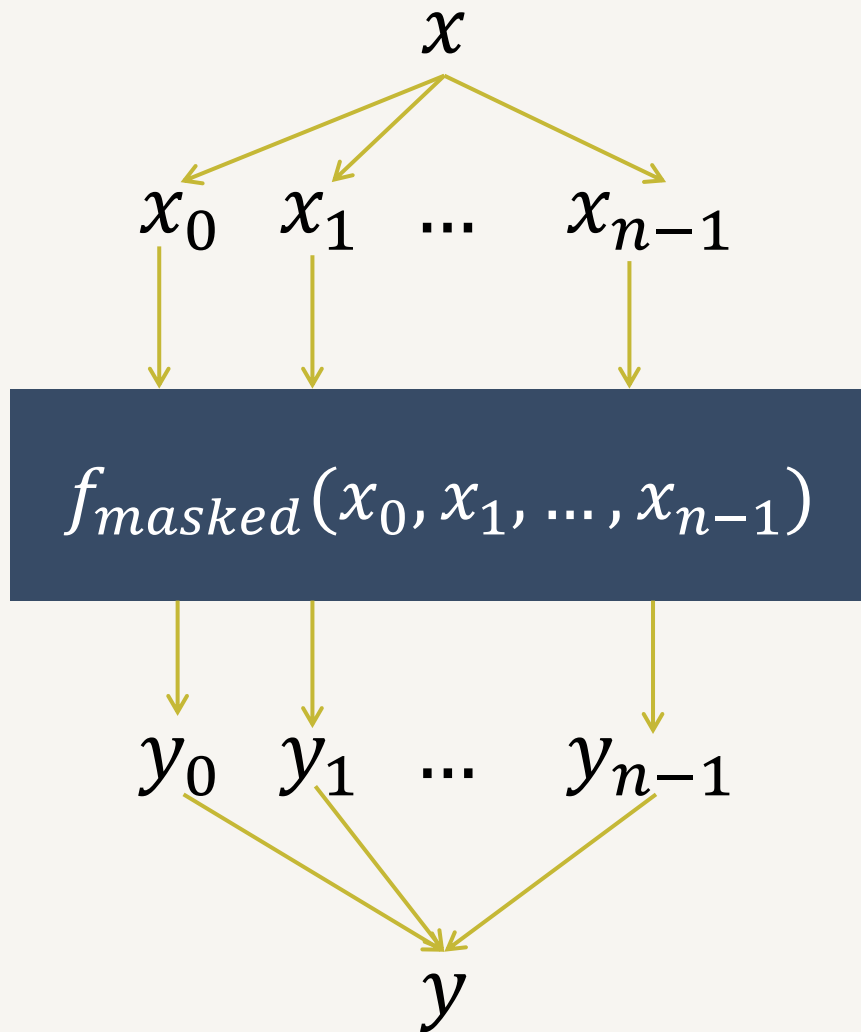


Guo, Johansson, Nilsson. A key-recovery timing attack on post-quantum primitives using the Fujisaki-Okamoto transformation and its application on FrodoKEM. In Crypto 2020.

Surviving the FO-calypse



Masking against SCA



- Encode secrets and sensitive variables into **shares**
- Compute securely on shares by **“rewriting” the function f**
- Decode at the end to recover result
- Arithmetic and/or Boolean masking

Masking if implemented **correctly** increases the attack complexity **exponentially** in the number of shares.
(assuming sufficient noise)

Masking ML-KEM (Kyber)

Performance Number for masked Kyber (kCycles) on ARM Cortex-M4

Operation	Number of shares					
	2	3	4	5	6	7
Kyber.CCAKEM.Decaps	3 178	57 141	97 294	174 220	258 437	350 529
Kyber.CPAPKE.Dec	200	4 203	7 047	13 542	20 323	27 230
Kyber.CPAPKE.Enc	2 024	18 879	32 594	53 298	75 692	104 191
comparison ($c = c'$)	693	32 293	54 725	102 922	156 075	210 518
$\mathcal{G}$	98	1 639	2 801	4 489	6 456	8 794
$\mathcal{H}$	113	113	113	113	113	113
$\mathcal{H}'$	13	13	13	13	13	13



- Masked decryption is <10% of the cost of masked decapsulation
- Cost of masked decapsulation is dominated by the masked FO

Azouaoui, M., Kuzovkova, Y., Schneider, T., van Vredendaal, C. "Post-Quantum Authenticated Encryption against Chosen-Ciphertext Side-Channel Attacks". TCHES 2022.

Avoiding the FO-calypse*

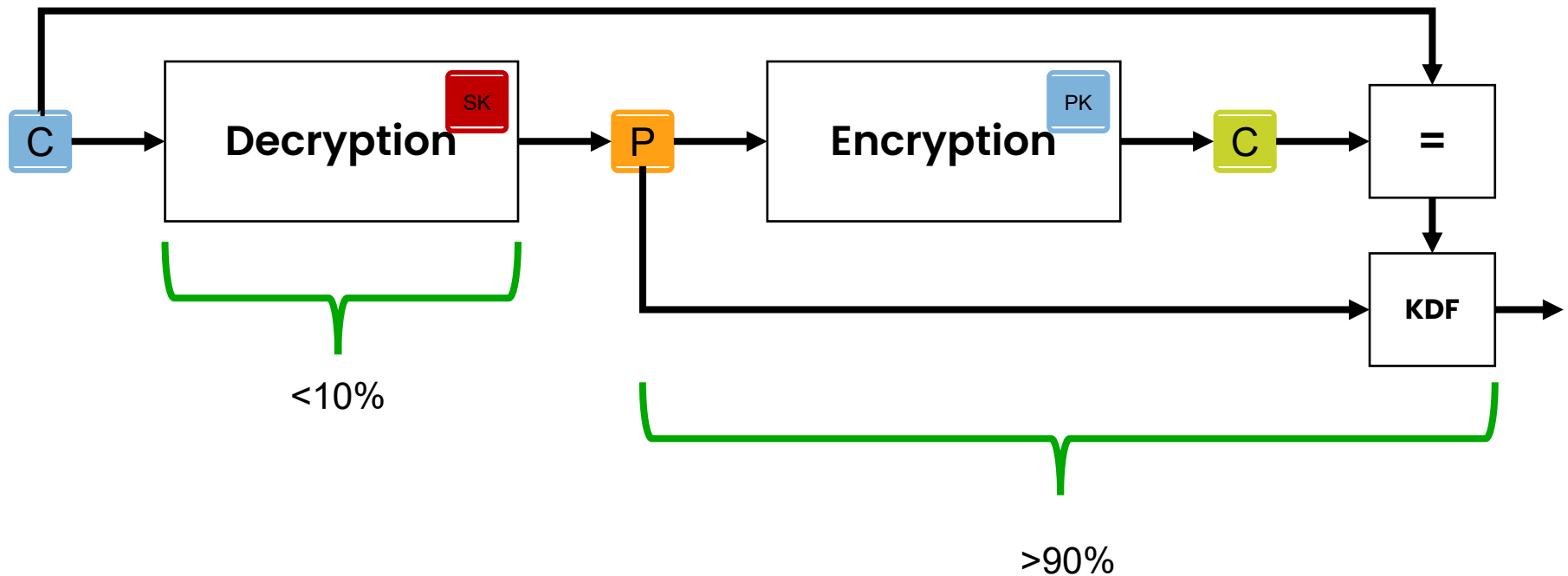
*for some use cases



FO KEM Alternative: Encrypt-then-Sign KEM

Observations:

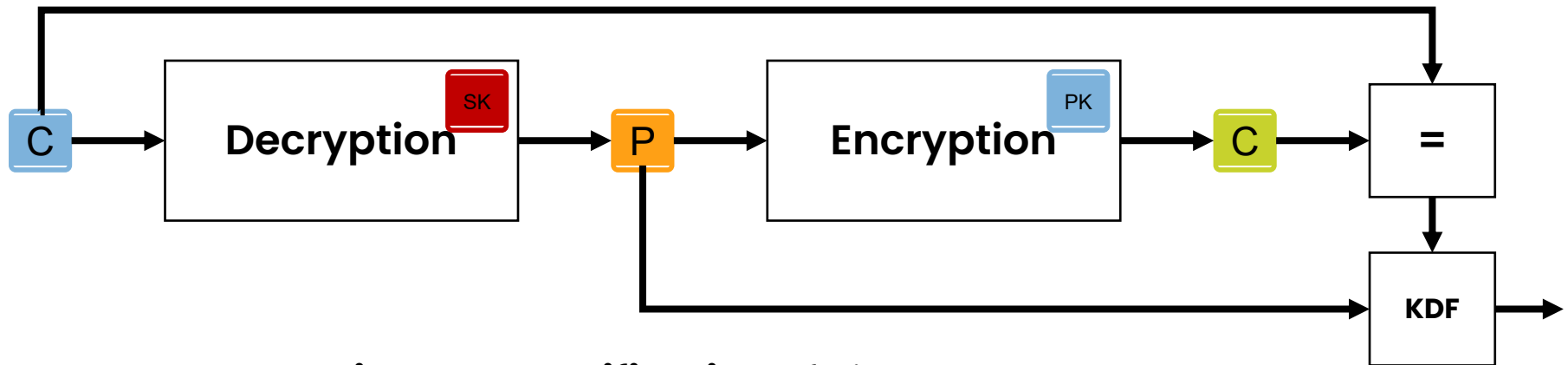
- Decryption is <10% of the runtime of the **protected** decapsulation
- Cost of masked decapsulation is dominated by the masked FO



FO KEM Alternative: Encrypt-then-Sign KEM

Observations:

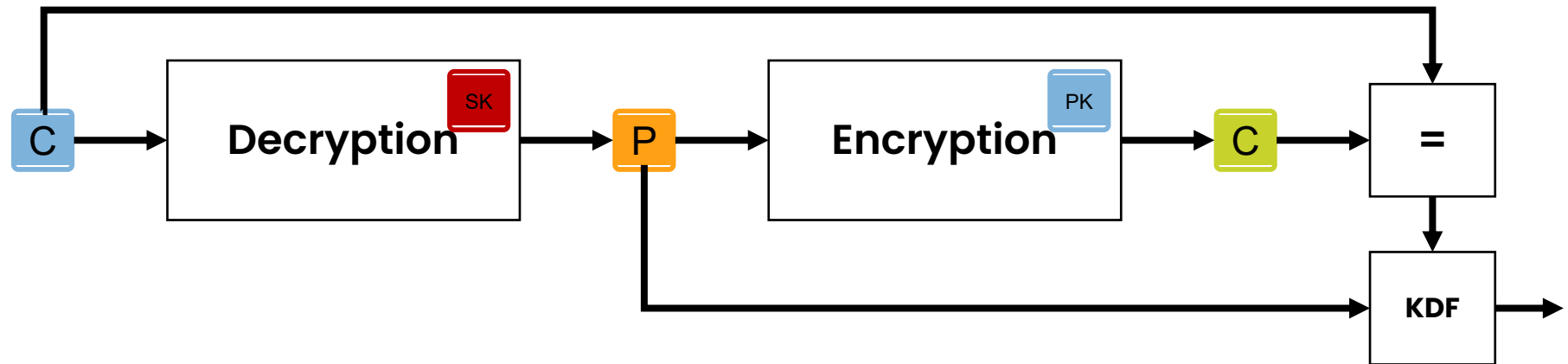
- Decryption is <10% of the runtime of the **protected** decapsulation
- Cost of masked decapsulation is dominated by the masked FO



Proposal: Replace FO check by **signature verification** of ciphertext

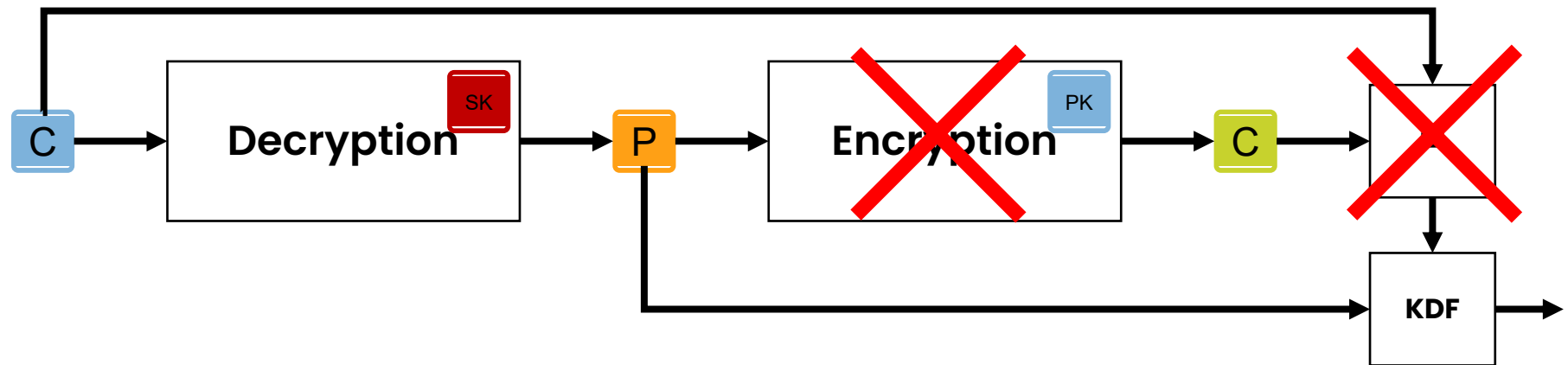
- Based on the Encrypt-then-Sign ($\mathcal{EtS}$) paradigm
- Signature verification only uses public data and does not require SCA protection
- Decapsulation requires less protection overall because no FO leakage

FO KEM Alternative: Encrypt-then-Sign KEM



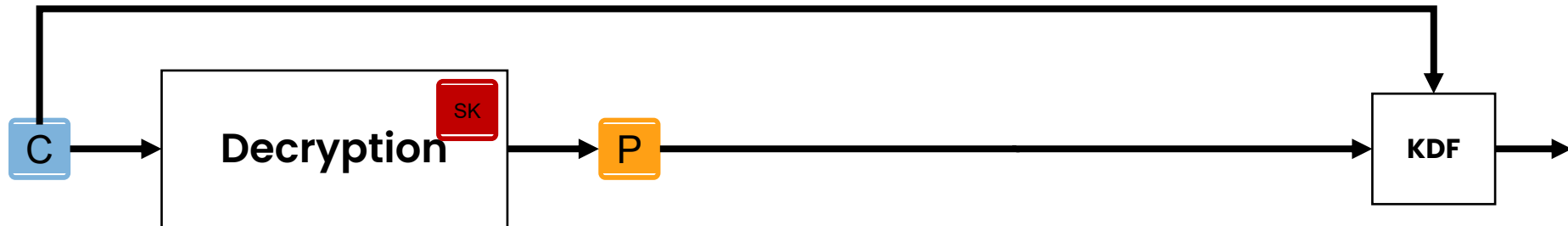
CCA Security

FO KEM Alternative: Encrypt-then-Sign KEM



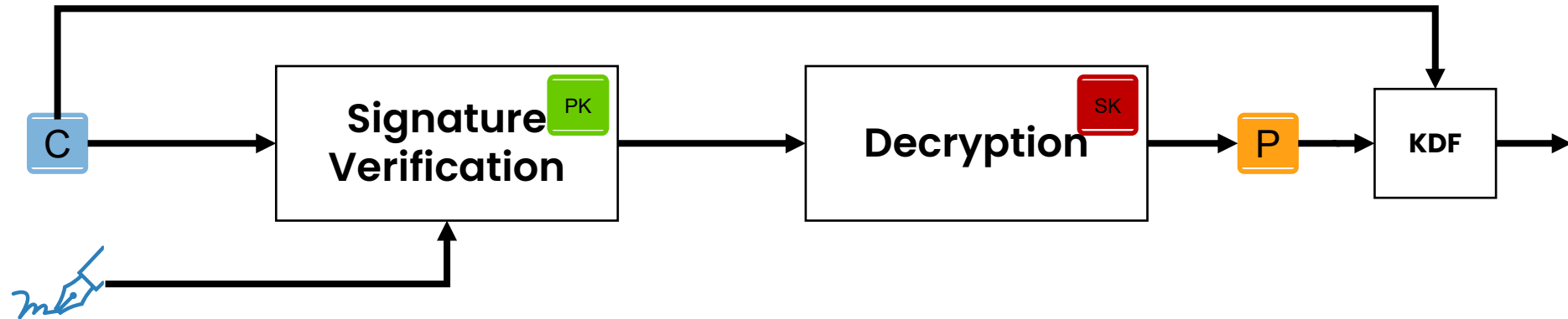
CCA Security

FO KEM Alternative: Encrypt-then-Sign KEM



CCA Security

FO KEM Alternative: Encrypt-then-Sign KEM



CCA Security*

***In the outsider model**

FO KEM Alternative: Encrypt-then-Sign KEM

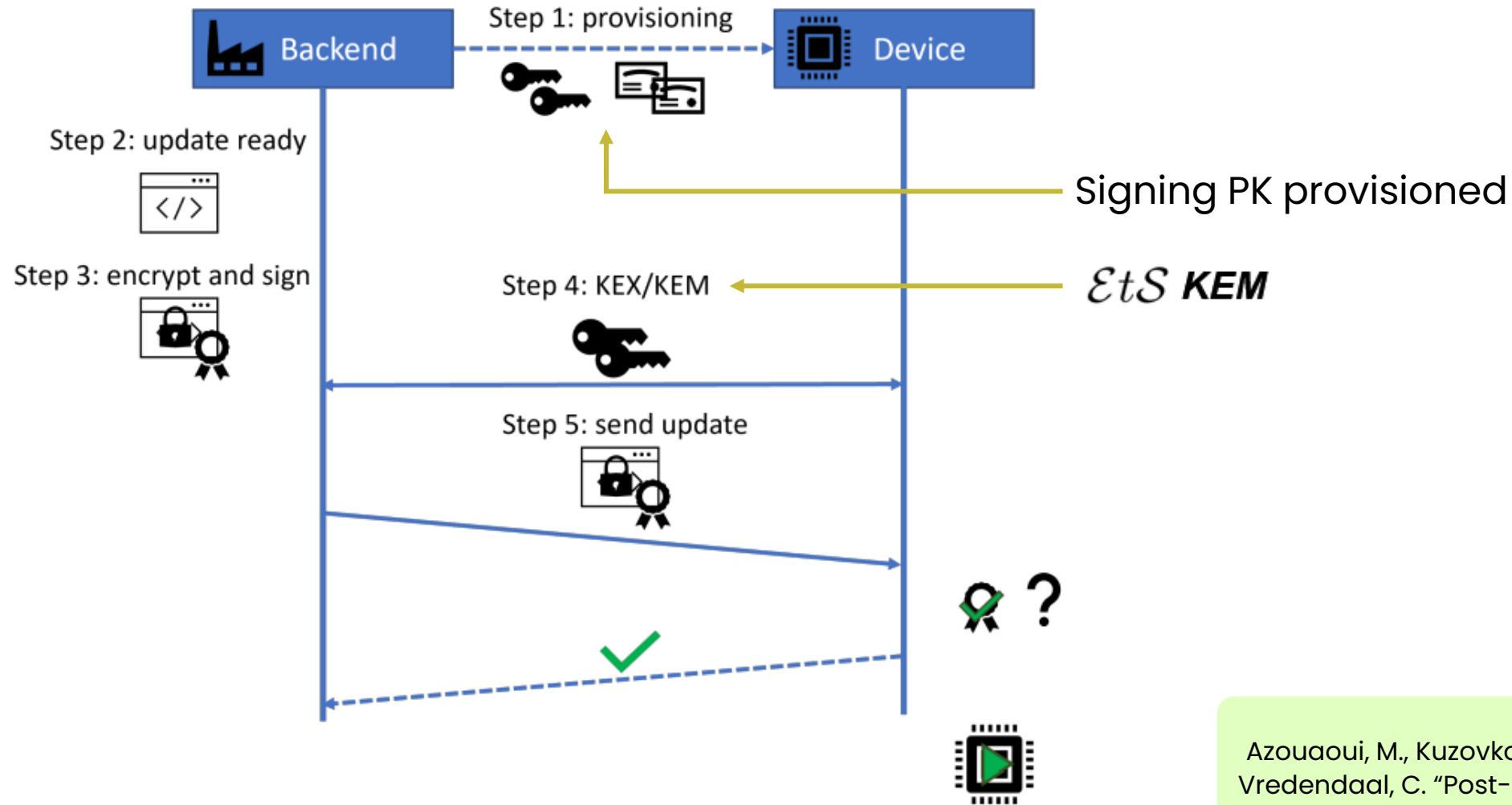
- **Outcome:** Never decrypt untrusted, potentially malicious, ciphertexts
- Standard application of **Signcryption / Encrypt-then-Sign $\mathcal{EtS}$** paradigm
- CCA security shown by An et al. in the **outsider security model**
 - Adversary has only access to public material
 - It is neither the sender nor the receiver
 - Adversary does not hold a trusted signing key
- Post-quantum CCA security shown by Chatterjee et al.

Y. Zheng. Signcryption and its applications in efficient public key solutions. ISW 1997.

An, JH., Dodis, Y., Rabin, R. On the Security of Joint Signature and Encryption. EUROCRYPT 2002.

Chatterjee, S., Pandit, T., Puria, SKP., Shah, A. Signcryption in a Quantum World. IACR ePrint Arch., 2020.

The $\mathcal{EtS}$ KEM for secure update mechanism



Azouaoui, M., Kuzovkova, Y., Schneider, T., van Vredendaal, C. "Post-Quantum Authenticated Encryption against Chosen-Ciphertext Side-Channel Attacks". TCHES 2022.

Performance improvement, pros & cons

- No Side-Channel CCAs / FO leakage
- Significant performance improvement



# of shares	Kyber.Decaps	EtS Kyber + Dilithium Verify	Improvement
2	3 178	2 568	× 1.24
3	57 141	6 571	× 8.7
4	97 294	9 415	× 10.33
5	174 220	15 910	× 10.95
6	258 437	22 691	× 11.39

Performance improvement, pros & cons



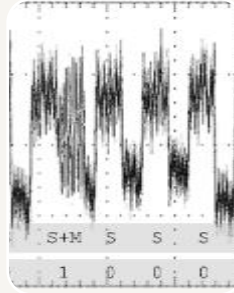
- Increased “ciphertext” size

	Kyber.Decaps	EtS Kyber + Dilithium Verify	Size increase
Ciphertext size	1088 bytes	4381 bytes	× 4

- Not FIPS compliant

FIPS 203: “K-PKE is only a component. The **public-key encryption scheme** K-PKE described in Section 5 **shall not be used as a stand-alone cryptographic scheme**. Instead, the algorithms that comprise K-PKE **may only be used as subroutines in the algorithms of ML-KEM**. In particular, the algorithms K-PKE.KeyGen (Algorithm 13), K-PKE.Encrypt (Algorithm 14), and K-PKE.Decrypt (Algorithm 15) are **not approved for use as a public-key encryption scheme**.”

Resistance against physical attacks



Side-channel attacks

- Power analysis (SPA, DPA)
- Electromagnetic analysis (SEMA, DEMA)
- Timing Analysis
- Photo-emission microscopy (high-end)
- Profiled, unprofiled and ML-assisted variants



Fault injection attacks

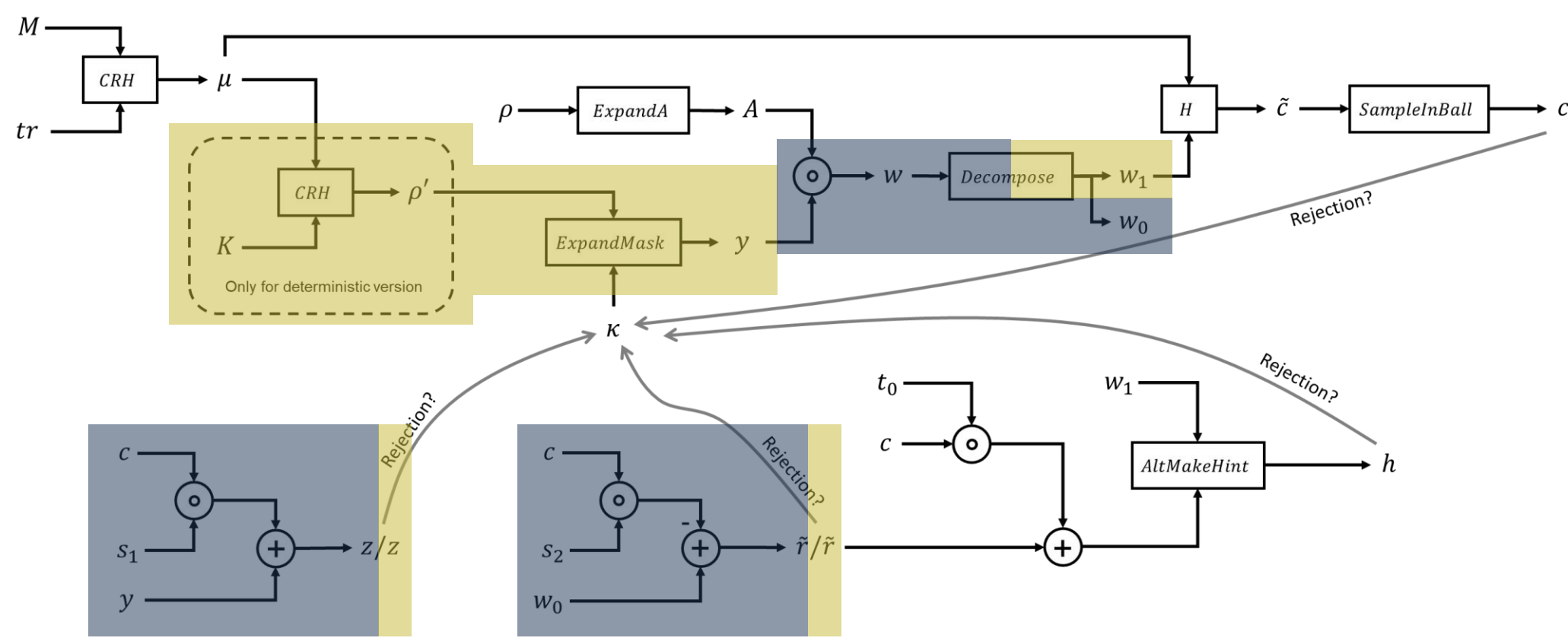
- Voltage or clock glitching
- Electromagnetic fault injection (EMFI)
- Body bias injection
- Laser fault injection
- Single and multi-shot scenarios



Invasive attack

- Focused Ion Beam (FIB) modifications
- Micro/Nano-probing of internal signals
- Signal forcing
- Delayering
- Reverse-engineering

Side-Channel Hardened ML-DSA



Boolean

Arithmetic

Side-Channel Hardened ML-DSA



Full SW implementation

ARM Cortex-M4

Based on the PQM4 library

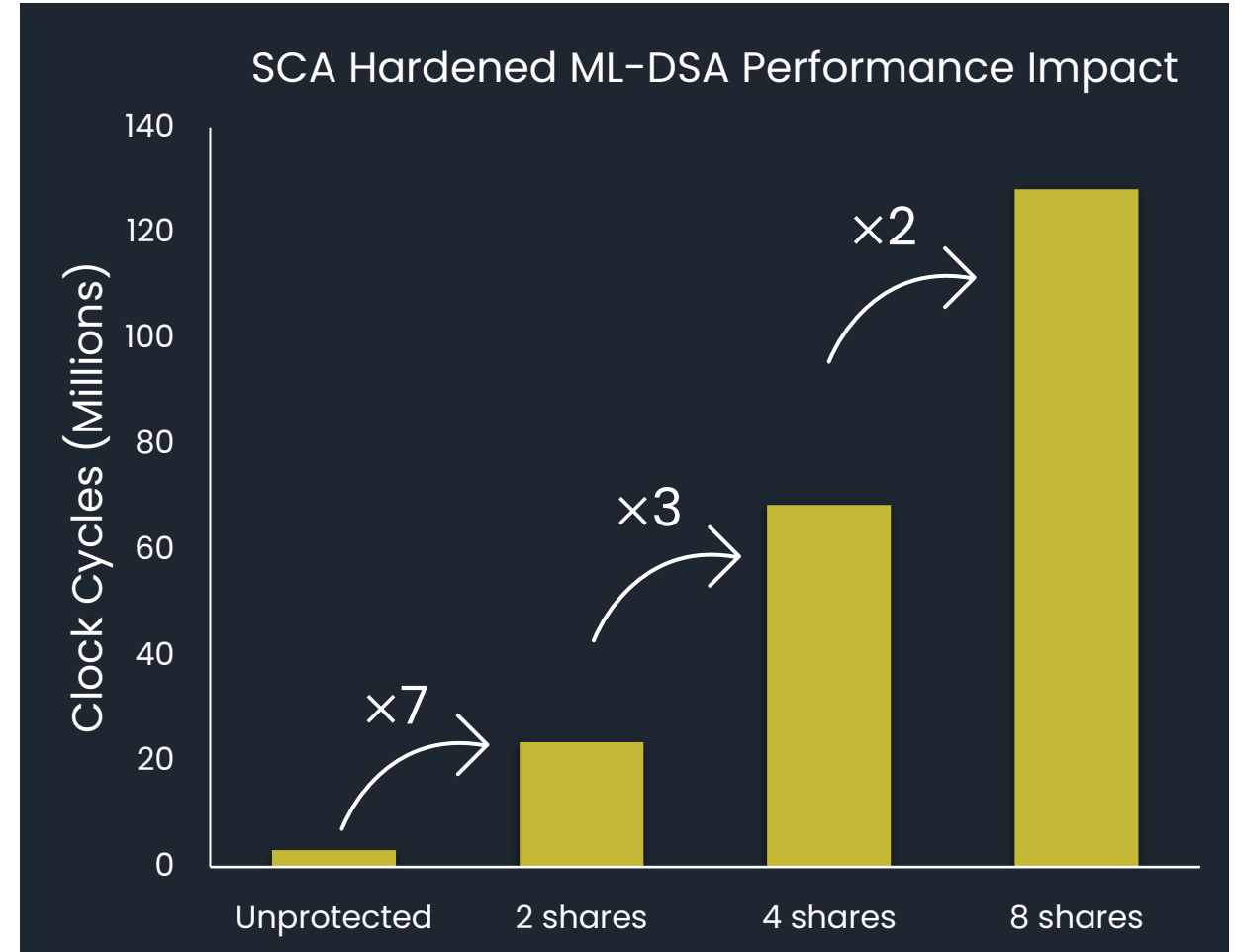


Hardening can be expensive but *necessary*

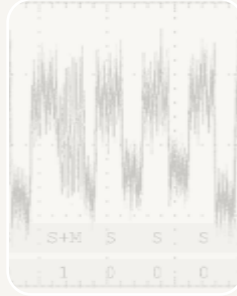
Post-Quantum security is *pointless* if embedded implementations are not protected against a few \$ physical attacks



SCA-protected hashing $\approx$ 40% of runtime $\rightarrow$ HW acceleration



Resistance against physical attacks



Side-channel attacks

- Power analysis (SPA, DPA)
- Electromagnetic analysis (SEMA, DEMA)
- Timing Analysis
- Photo-emission microscopy (high-end)
- Profiled, unprofiled and ML-assisted variants



Fault injection attacks

- Voltage or clock glitching
- Electromagnetic fault injection (EMFI)
- Body bias injection
- Laser fault injection
- Single and multi-shot scenarios



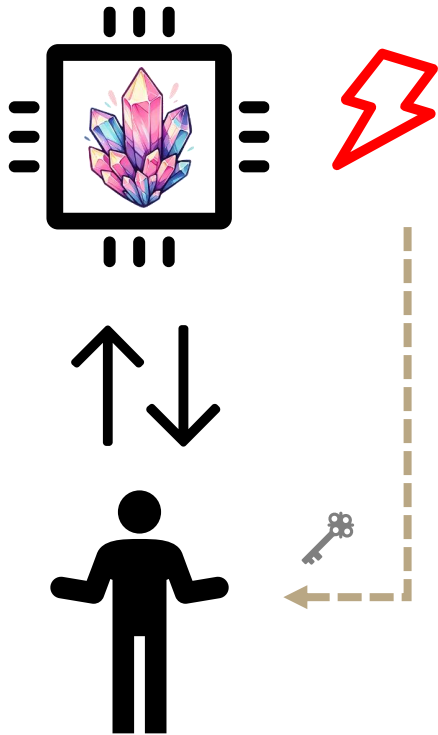
Invasive attack

- Focused Ion Beam (FIB) modifications
- Micro/Nano-probing of internal signals
- Signal forcing
- Delaying
- Reverse-engineering

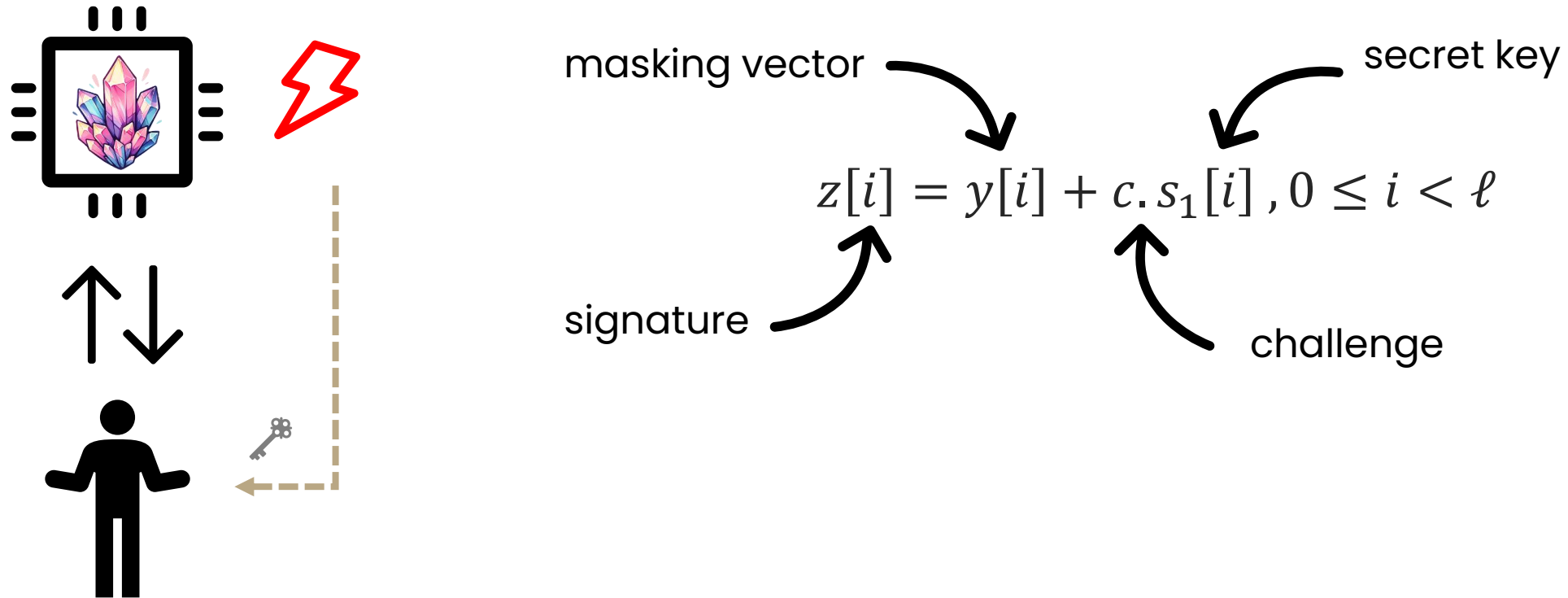
From MLWE to RLWE: DFA on randomized ML-DSA



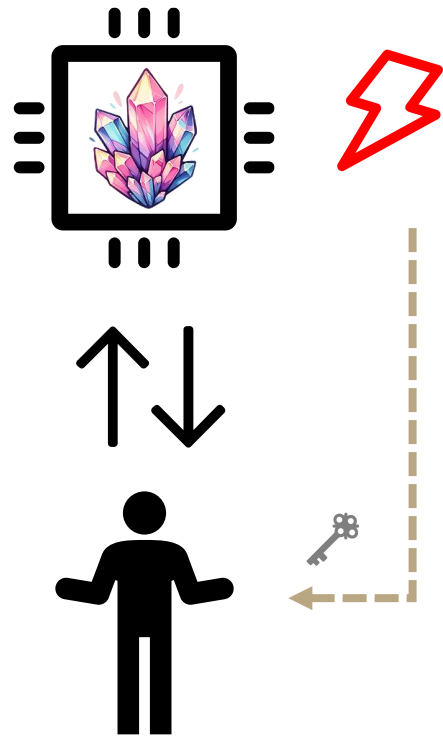
Instruction Skip Attack on ML-DSA



Instruction Skip Attack on ML-DSA



Instruction Skip Attack on ML-DSA



masking vector

secret key

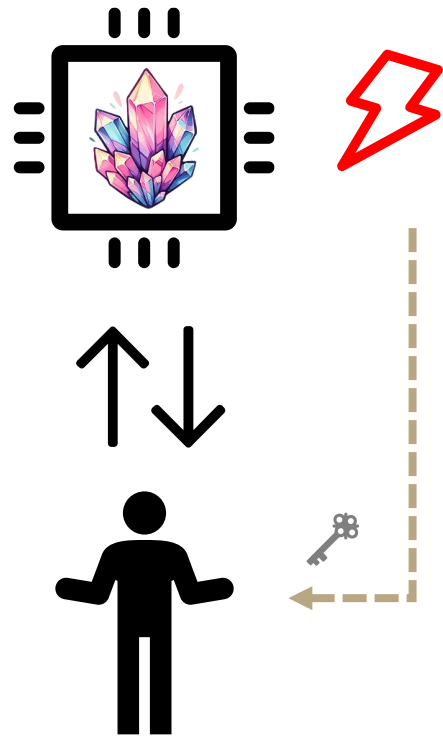
$$z[i] = y[i] + c.s_1[i], 0 \leq i < \ell$$

signature

challenge

```
void polyvec1_uniform_gamma1(polyvec1 *v, const uint8_t seed[CRHBYTES], uint16_t nonce) {  
    unsigned int i;  
  
    for(i = 0; i < L; ++i)  
        poly_uniform_gamma1(&v->vec[i], seed, L*nonce + i);  
}
```

Instruction Skip Attack on ML-DSA



masking vector

secret key

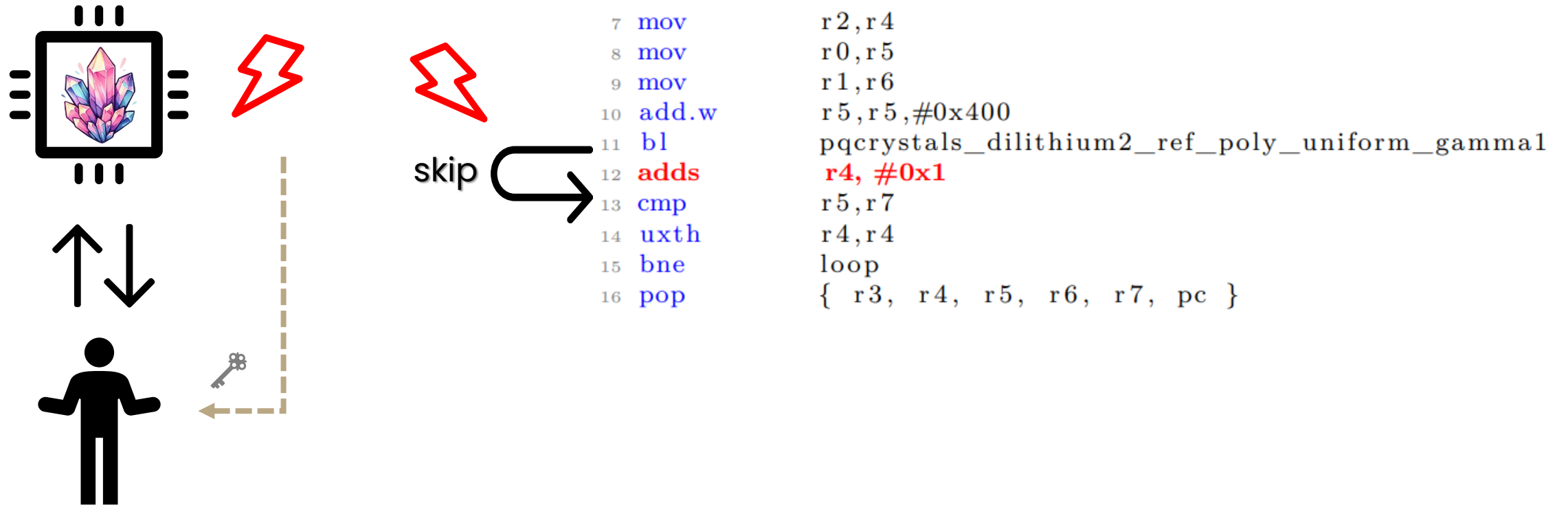
$$z[i] = y[i] + c.s_1[i], 0 \leq i < \ell$$

signature

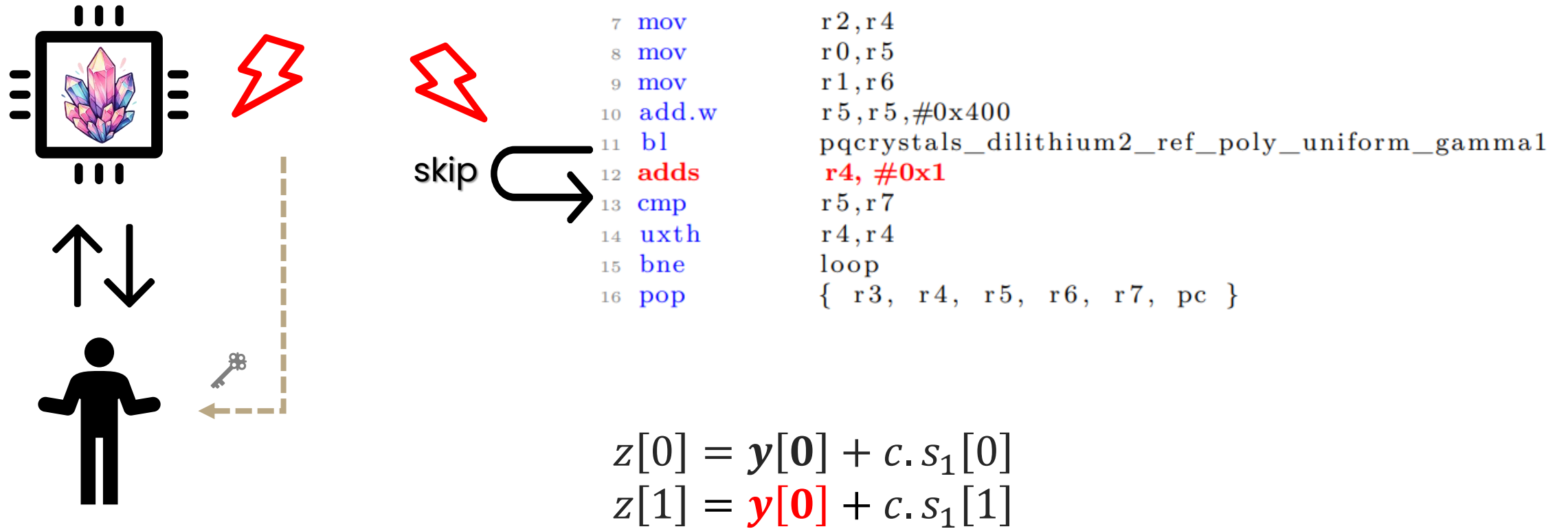
challenge

```
void polyvec1_uniform_gamma1(polyvec1 *v, const uint8_t seed[CRHBYTES], uint16_t nonce) {  
    unsigned int i;  
  
    for(i = 0; i < L; ++i)  
        poly_uniform_gamma1(&v->vec[i], seed, L*nonce + i);  
}
```

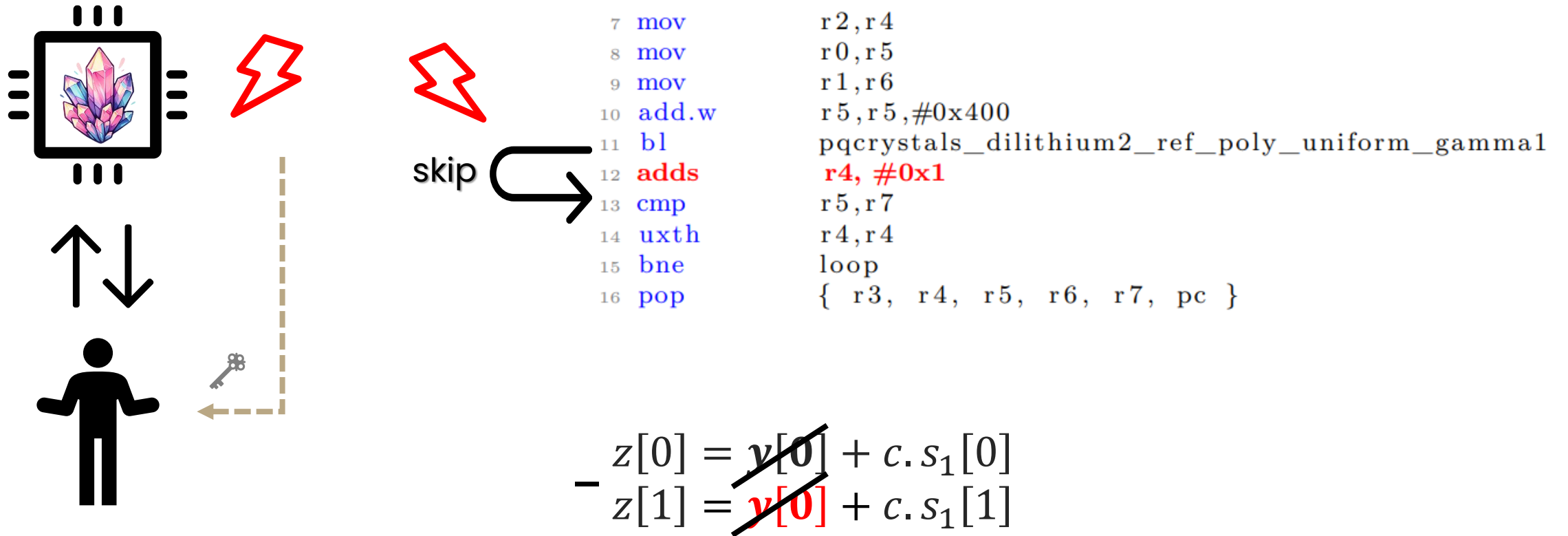
Differential Fault Attack on ML-DSA



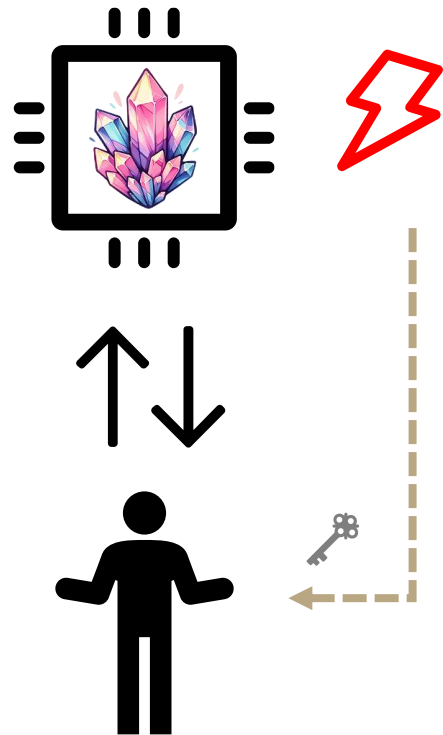
Differential Fault Attack on ML-DSA



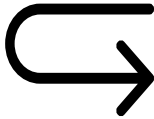
Differential Fault Attack on ML-DSA



From MLWE to RLWE



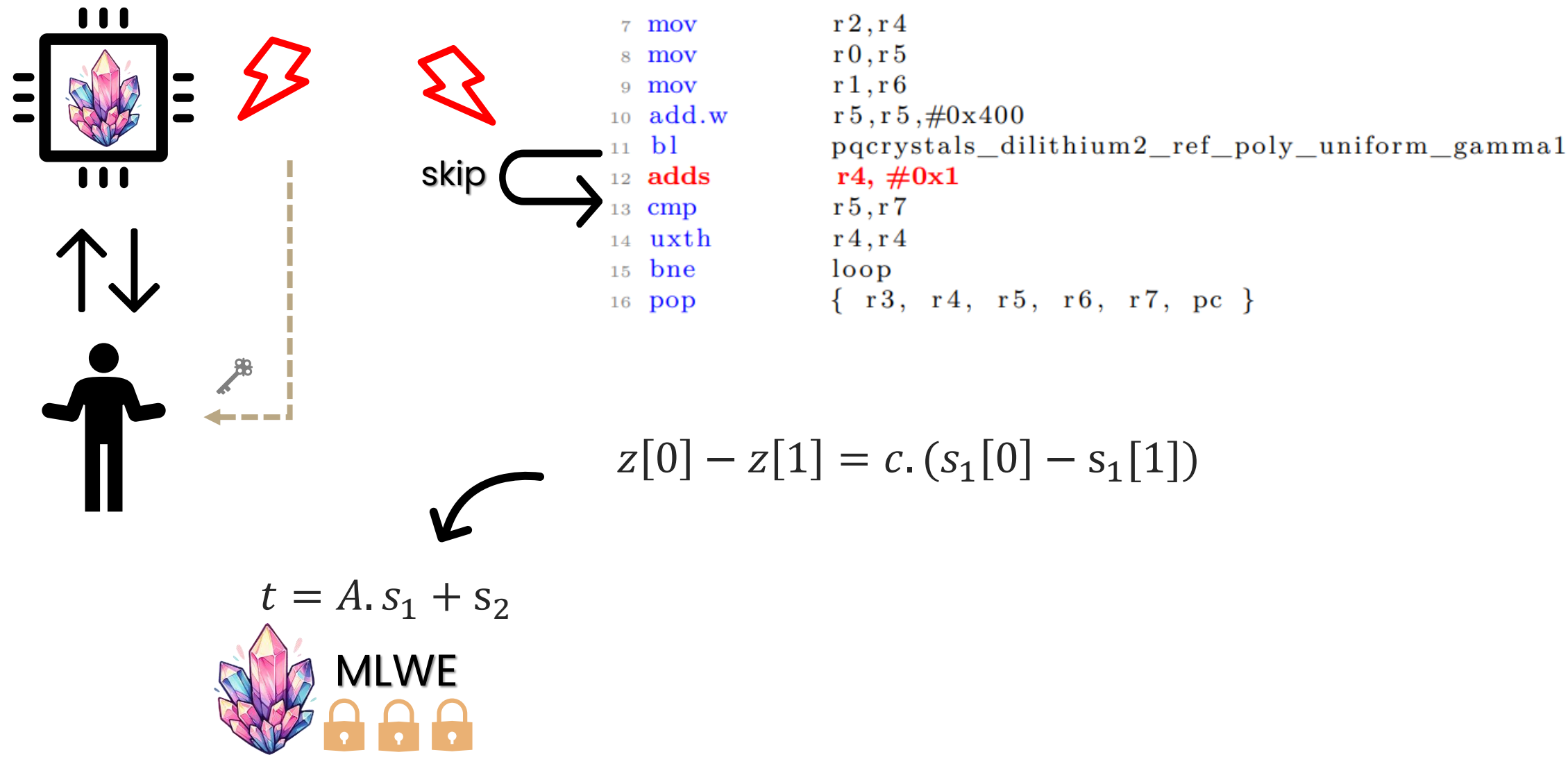
skip



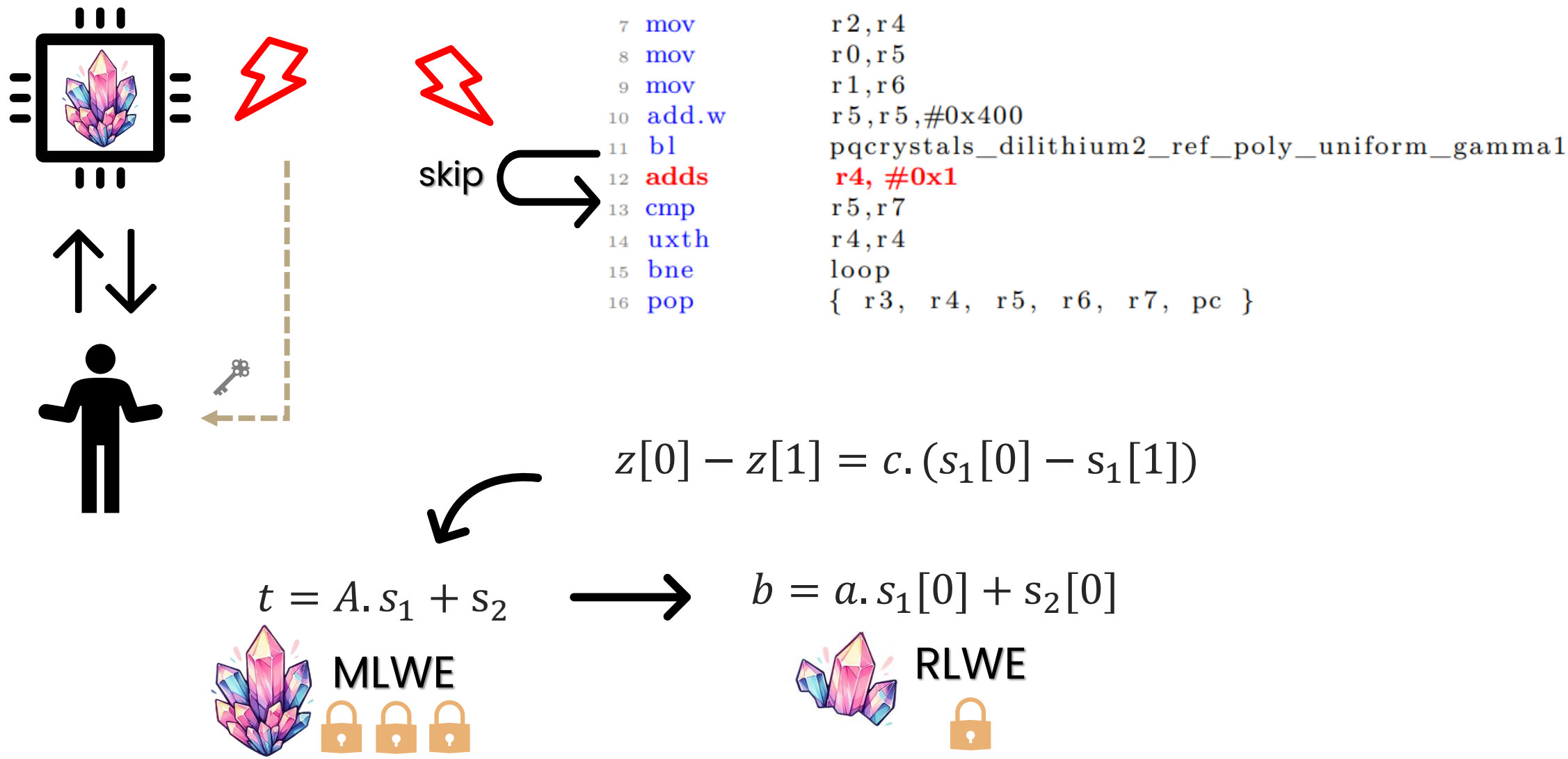
```
7  mov      r2,r4
8  mov      r0,r5
9  mov      r1,r6
10 add.w    r5,r5,#0x400
11 bl       pqcrystals_dilithium2_ref_poly_uniform_gamma1
12 adds    r4, #0x1
13 cmp      r5,r7
14 uxth     r4,r4
15 bne      loop
16 pop      { r3, r4, r5, r6, r7, pc }
```

$$z[0] - z[1] = c. (s_1[0] - s_1[1])$$

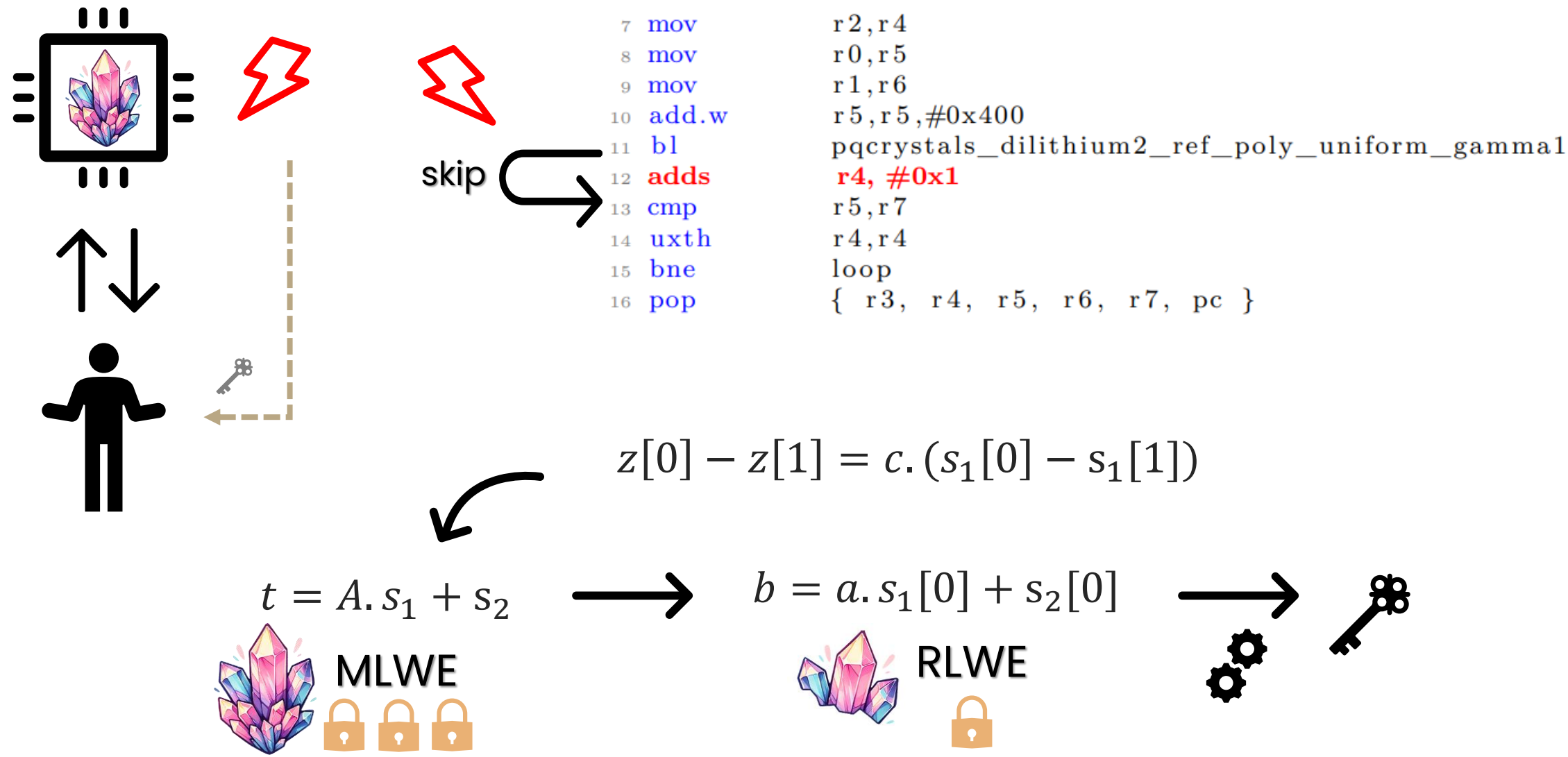
From MLWE to RLWE



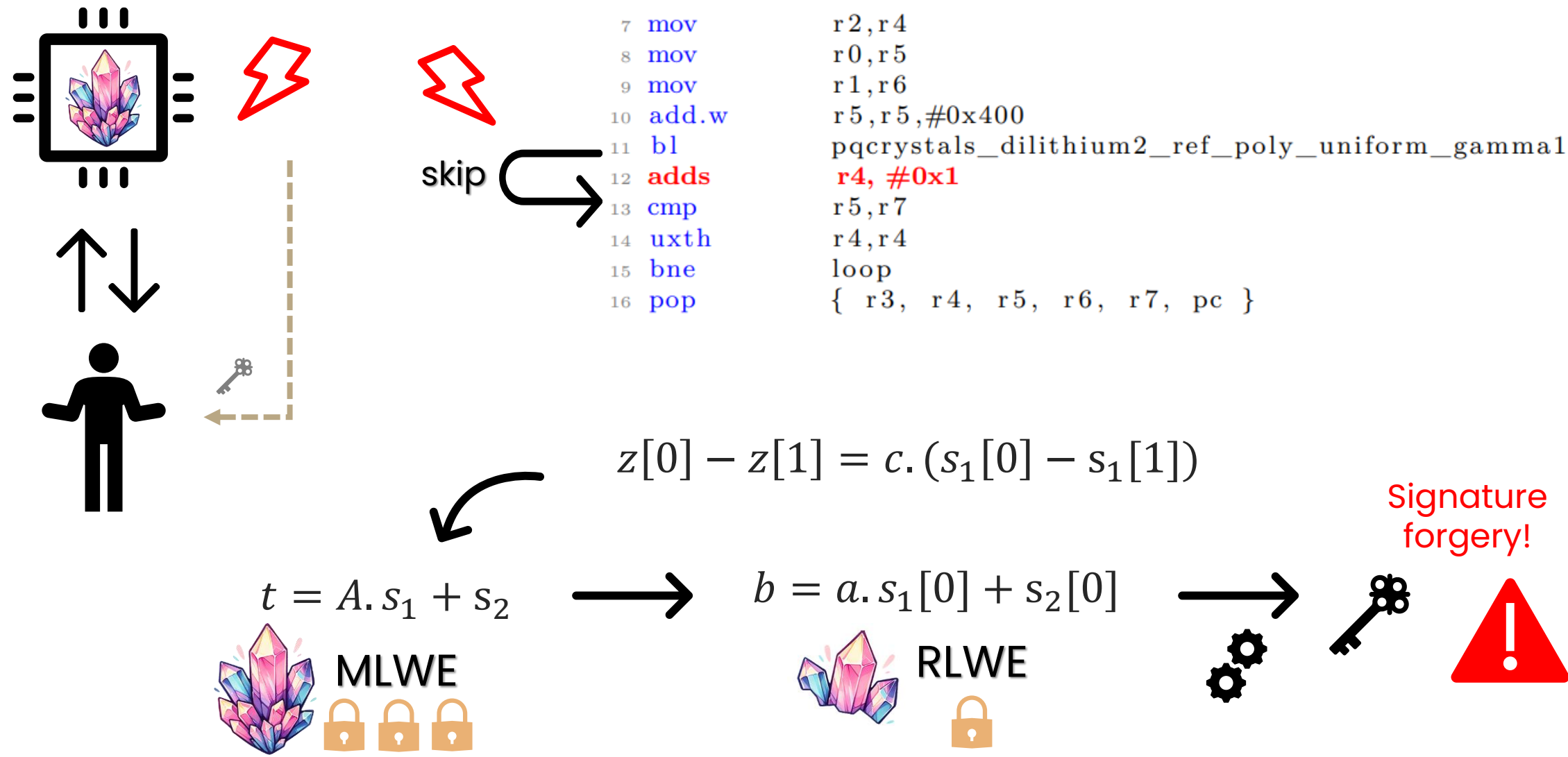
From MLWE to RLWE



From MLWE to RLWE



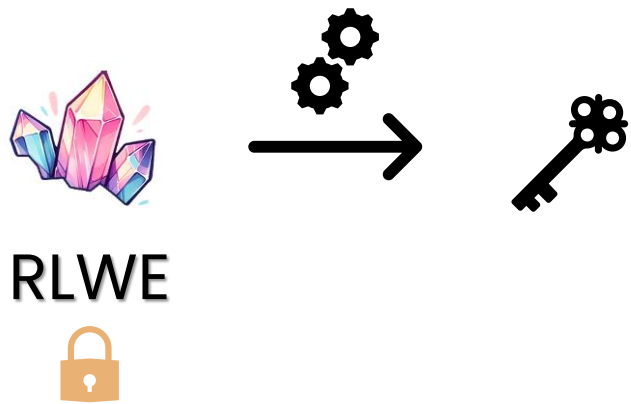
From MLWE to RLWE



From MLWE to RLWE



Solving the Resulting RLWE



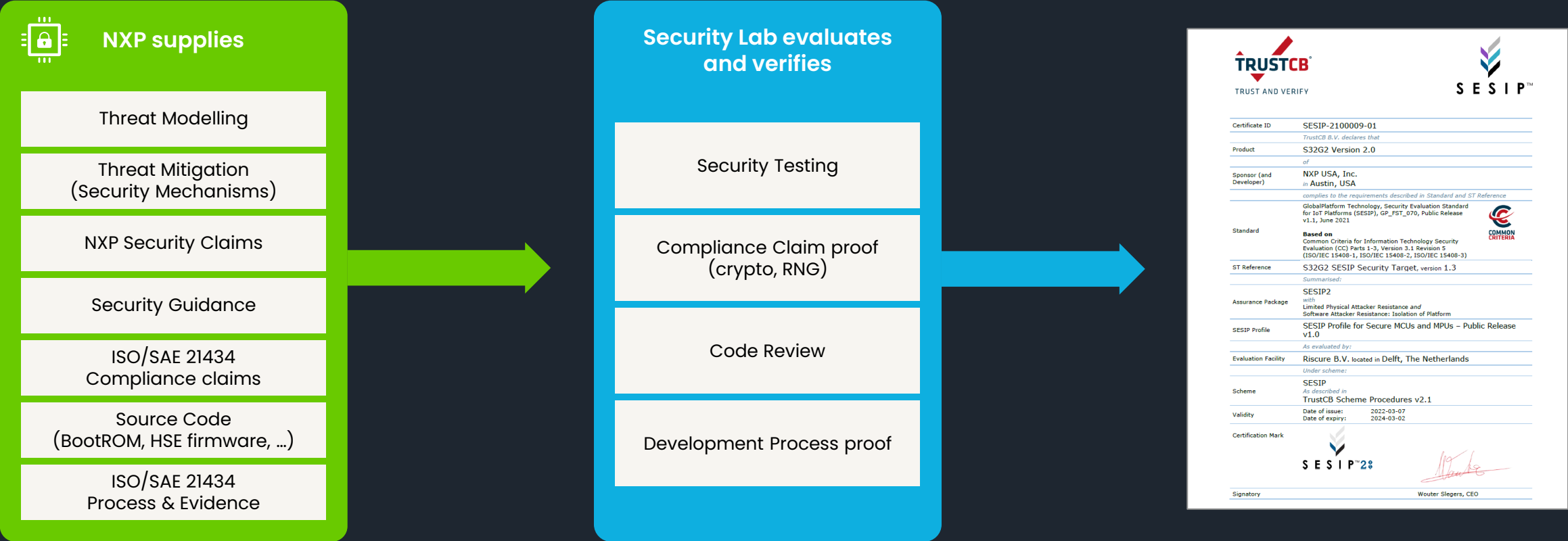
	NIST Security Level	II	III	V
 MLWE	β before attack	434	641	890
 RLWE	β after attack (average)	29	48	2
	Attack runtime (hours)	~15	~17	~23

[DDGR20] Dana Dachman-Soled, Léo Ducas, Huijing Gong, Mélissa Rossi. LWE with Side Information: Attacks and Concrete Security Estimation. CRYPTO 2020.

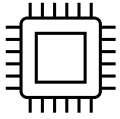
**What does it
mean in practice?**



From Attacks & Countermeasures to achieving Security & Safety



Summary



Embedded challenges of PQC

- Migration spans many layers
- Diverse platforms, use cases & applications
- Implementation challenges: memory, sizes, speed, physical security



Embedded security **with & for** PQC

- PQC alone doesn't guarantee security
- Physical attack resistance is essential
- Protocol design can reduce attack surface



Attacks & Countermeasures

- Understanding attacks is key to understanding what must be protected
- Countermeasures provide resistance and protection
- Together they enable high-assurance implementations





Thank you!

Melissa Azouaoui

melissa.azouaoui@nxp.com

nxp.com