SCIENCES SORBONNE UNIVERSITÉ

THALES
Building a future we can all trust

Implementations of Post-Quantum Cryptography Algorithms Secured Against Physical Attacks

CALLE VIERA Andersson
Director : VERGNAUD Damien
Supervisor: BERZATI Alexandre
PhD. Session CARDIS 2023, 16 Nov. 2023

[1] Thales DIS, France
[2] Sorbonne Université, France

Context

Shor's **quantum algorithm** can **break** standard public key cryptosystems (based on **integer factorization** and **discrete logarithm**), in polynomial time

Context

Shor's **quantum algorithm** can **break** standard public key cryptosystems (based on **integer factorization** and **discrete logarithm**), in polynomial time

NIST: National Institute of Standards and Technology

- 2017: International competition to standardized PQC public-key algorithms
- 2024: First KEM and DSA Standards finalized

Context

Shor's **quantum algorithm** can **break** standard public key cryptosystems (based on **integer factorization** and **discrete logarithm**), in polynomial time

NIST: National Institute of Standards and Technology

- 2017: International competition to standardized PQC public-key algorithms
- 2024: First KEM and DSA Standards finalized

Importance: These algorithms will be implemented **securely** in a variety of use cases

Banking

Personal Data

Communication

PhD. Roadmap

- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)

Study PQC

PhD. Roadmap

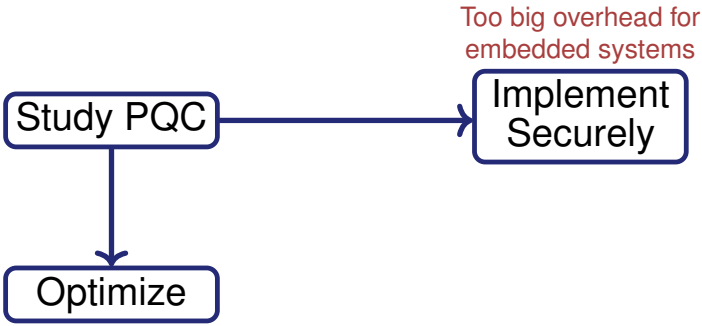
- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)

PhD. Roadmap

- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)

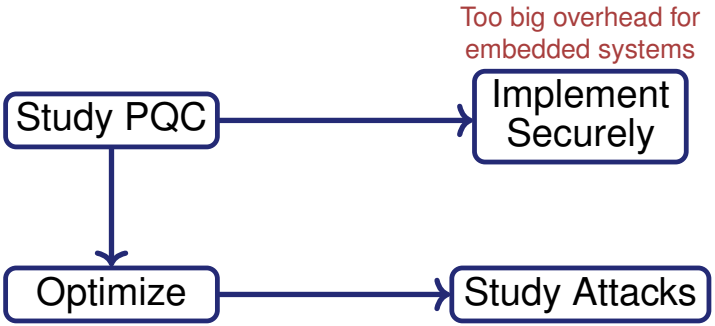
PhD. Roadmap

- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)



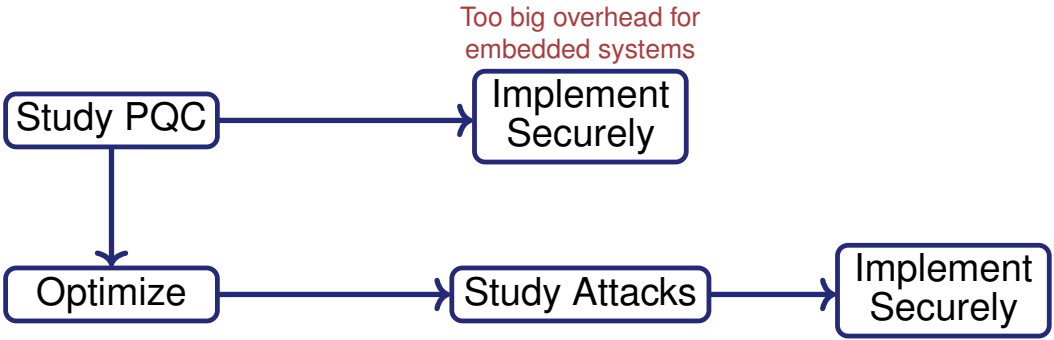
PhD. Roadmap

- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)



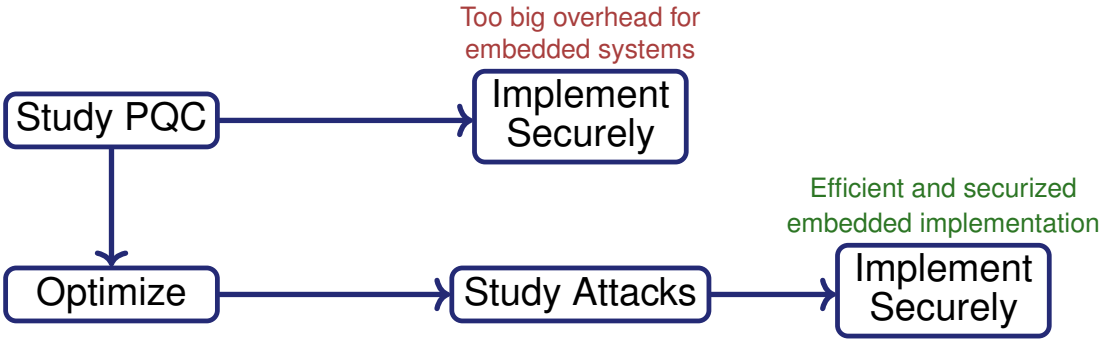
PhD. Roadmap

- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)



PhD. Roadmap

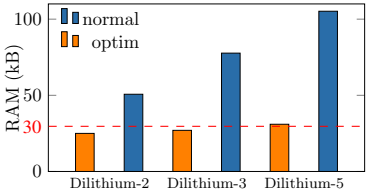
- CALLE VIERA Andersson
- PhD in cryptography from may 2022 to may 2025 (currently 2nd year)
- ALMASTY (Lip6, Sorbonne University) & THALES DIS (Meyreuil)



Optimizing Dilithium Signature Scheme

- Key size storage larger than secure element RAM size
- Reduce RAM consumption for the 3 security levels of Dilithium
- Up to 30% reduction for Dilithium-5
- Conform to standard Dilithium without fancy tricks

- Proprietary Implementation



A Practical Template Attack on Dilithium

Authors: BERZATI Alexandre, CALLE VIERA Andersson, CHARTOUNI Maya, MADEC Steven, VERGNAUD Damien, VIGILANT David

- Exploits zero value leakage during signature execution
- Allows to Recover (partial) secret key and forge signatures
- Confirms the need to protect this intermediate value
- Practical demonstration through Template Attack

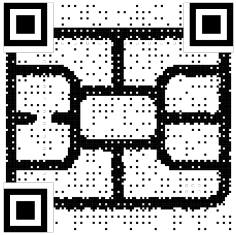
- Published at CHES 2023

ia.cr/2023/050

Fault Attacks sensitivity of Dilithium Verify

Authors: BERZATI Alexandre, CALLE VIERA Andersson, HEYDEMANN Karine

- Sensitivity Analysis of an implmentation of Verify
- Based on the idea to make $ct_1 2^d$ smaller than it is
- 4 faults models considered $\implies$ 3 main scenarios detailed
- Allow to accept false signatures



- Published at CARDIS 2023

sbd-research.nl/cardis-2023

Future Work

- Identify vulnerable operations within PQC schemes
  - SCA/FA on Dilithium/Kyber and NIST round 4 candidates
- Keep studying countermeasures for Dilithium and Kyber
  - Analyze the security of a potential efficient masking of the `Decompose` function
- Study novel approaches for implementing Dilithium and Kyber
  - Balance security and efficiency (changes in arithmetic used for example)

Future Work

- Identify vulnerable operations within PQC schemes
  - SCA/FA on Dilithium/Kyber and NIST round 4 candidates
- Keep studying countermeasures for Dilithium and Kyber
  - Analyze the security of a potential efficient masking of the `Decompose` function
- Study novel approaches for implementing Dilithium and Kyber
  - Balance security and efficiency (changes in arithmetic used for example)

Thank you

Questions?

Attacking Pair-Pointwise Multiplication in CRYSTALS-Kyber Using Deep Learning

Azade Rezaeezade

November 2023

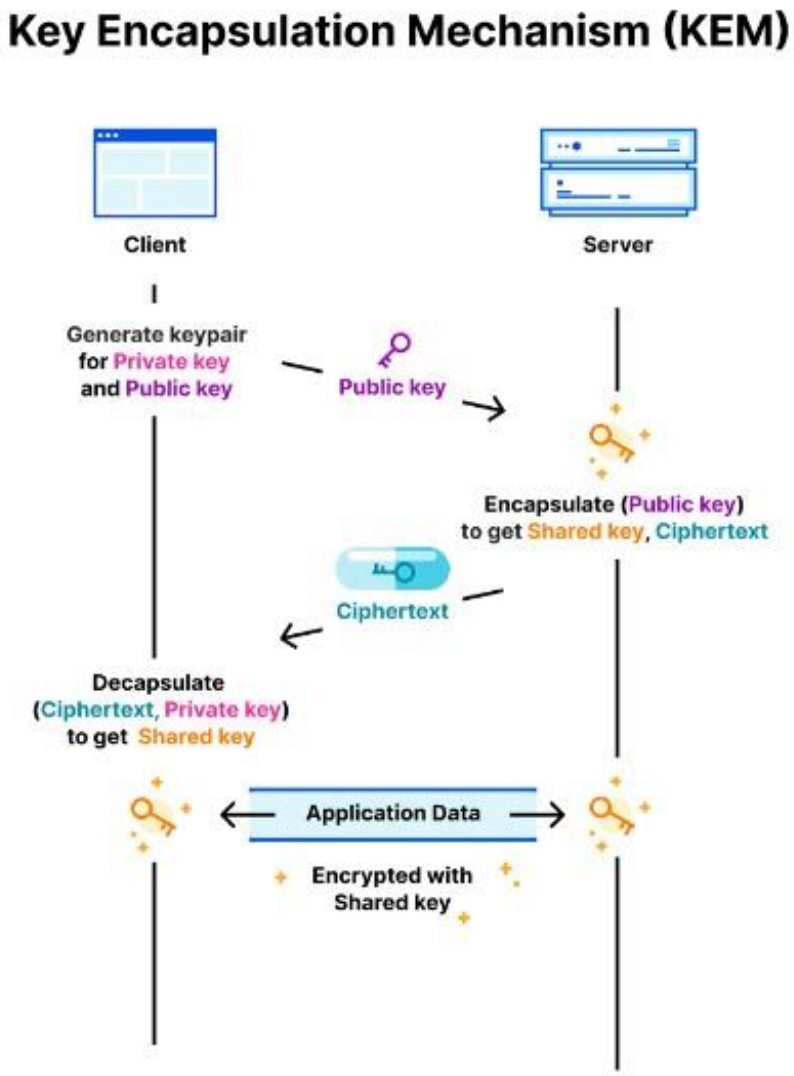
TU Delft

Why We Need Post-Quantum Crypto

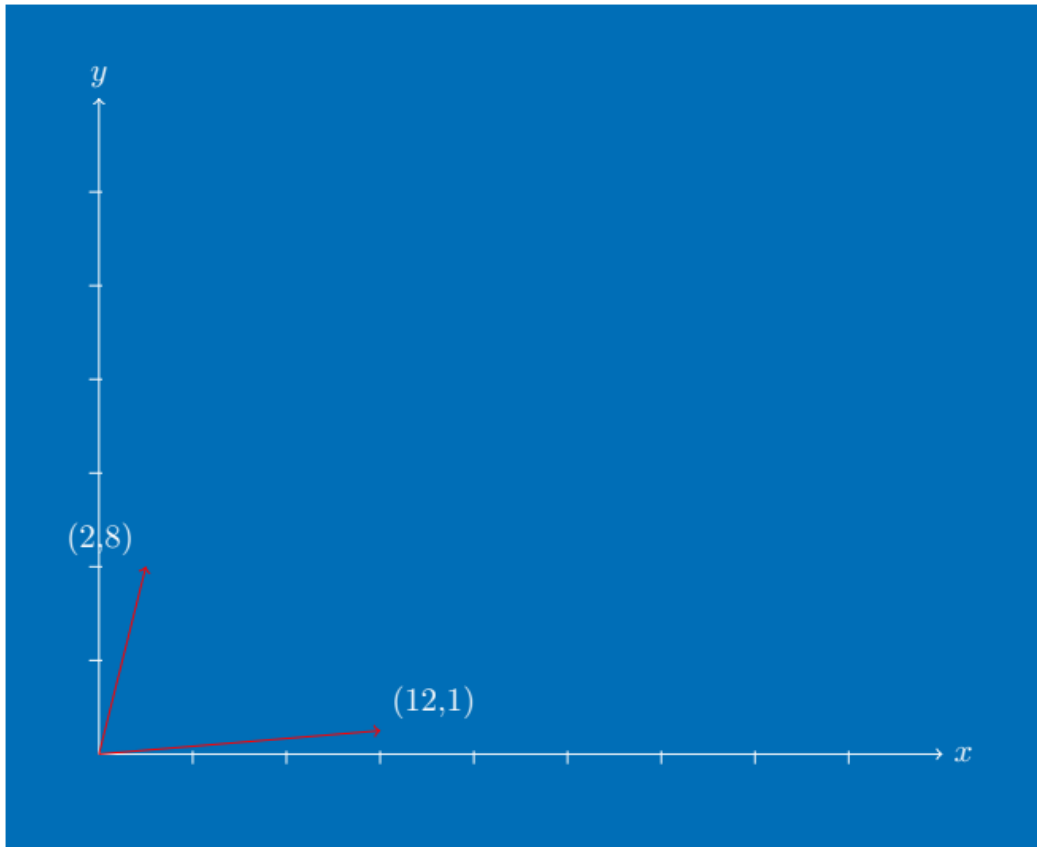
- The thread of large-scale quantum computers
- Shor’s algorithm breaks RSA and ECC
- CRYSTALS-Kyber is going to be standardized by NIST as a Key Exchange Mechanism

Key Encapsulation Mechanism

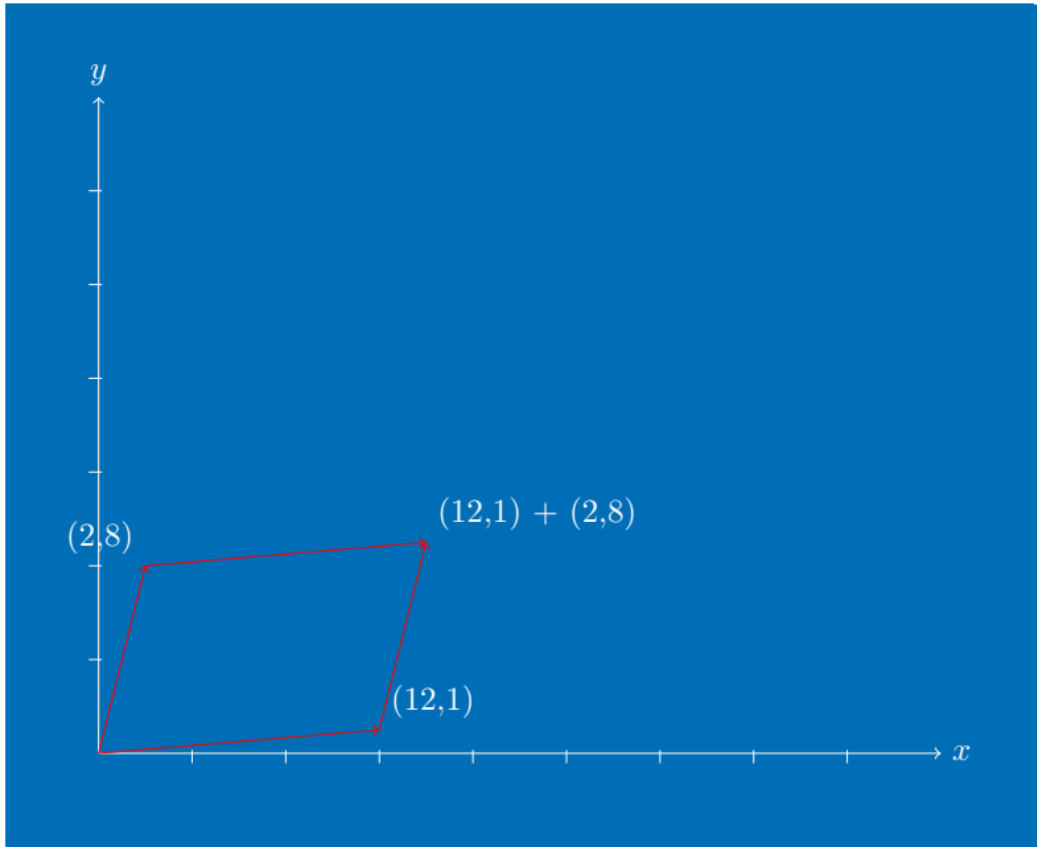
- Alice and Bob want to communicate
- The final goal is exchanging a shared key that can be used for symmetric cryptography.



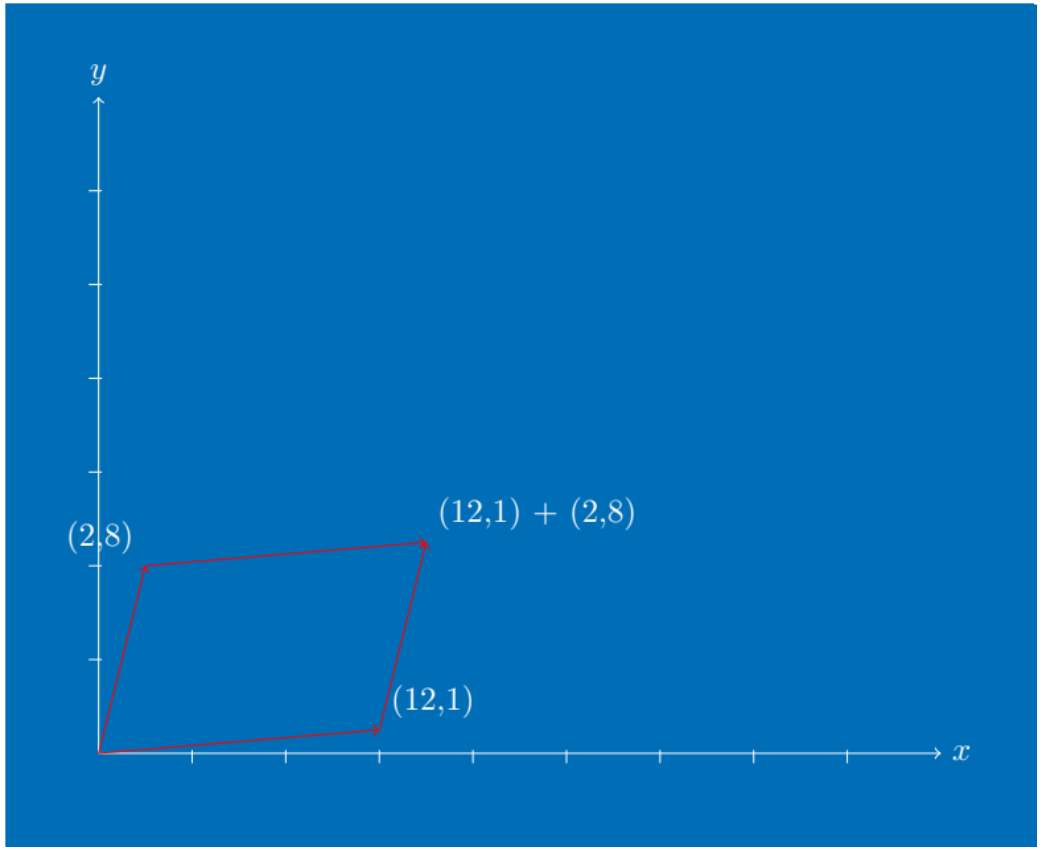
Get Deeper in Lattices

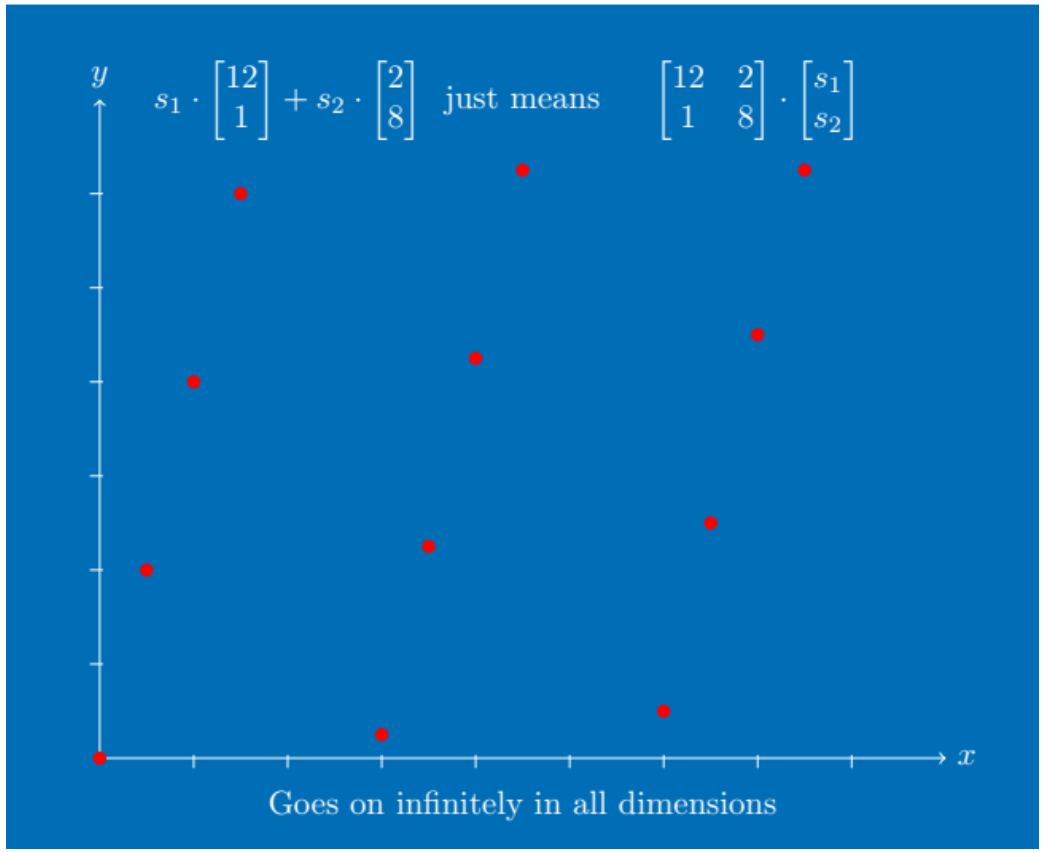


Get Deeper in Lattices

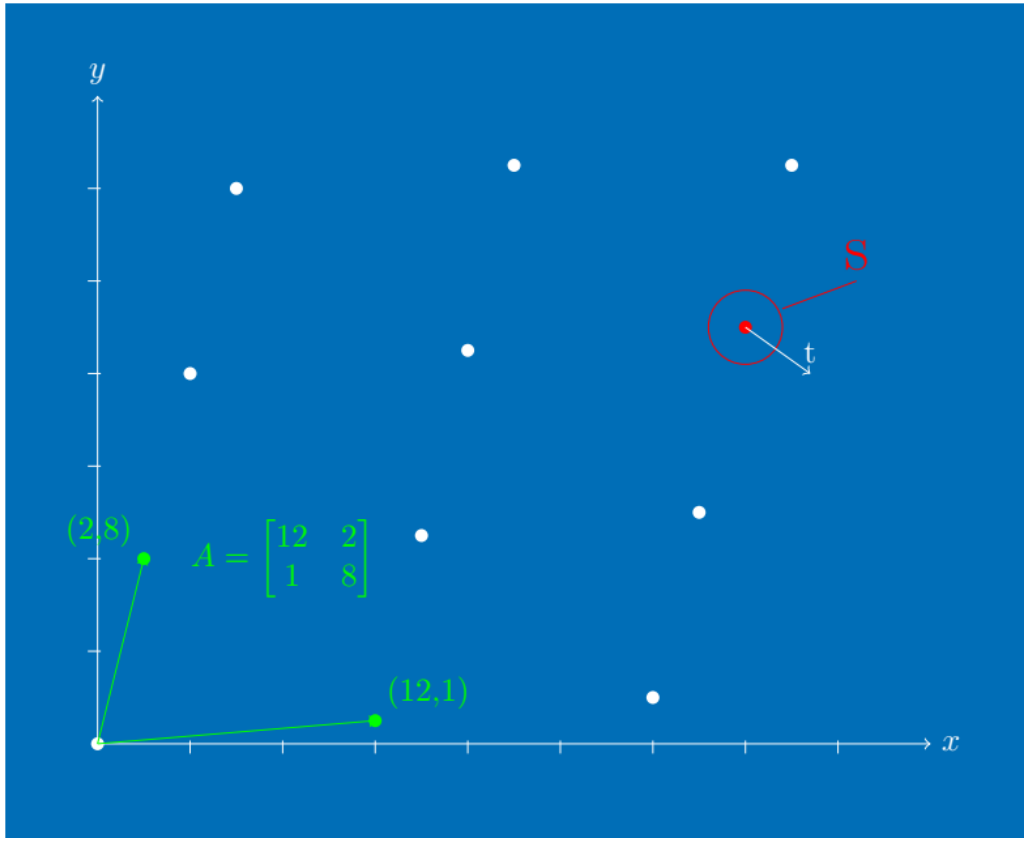


Get Deeper in Lattices





From Lattices to Public-Private Key Pair



- Final point is A.s+e = t
- Public key is:
  - Matrix A
  - Vector t
- Private key is:
  - Vector S hidden by error vector e
- This is Learning with Errors Problem

Kyber in Reality

- A is a matrix of polynomials with dimension 2, 3 or 4.
- s , e and t are vectors of polynomials with dimension 2, 3 or 4.
- The polynomials are from the ring R_q:

$$R_q := \frac{Z_{[X]}}{(X^n + 1)}, \qquad n = 256, \qquad q = 332$$

NTT Domain

- All the multiplication are happening in NTT domain
- NTT is a kind of Fourier transform
- Kyber polynomials in NTT domain are like:

$$NTT\,(a) = \; a_0 + a_1 x, \quad a_2 + a_3 x, \quad a_4 + a_5 x\,, \ldots, \quad a_{254} + a_{255} x$$

- Multiplication of two polynomials equals to:

$$c_1 = a_0 b_1 + a_1 b_0$$

$$c_0 = a_0 b_0 + a_1 b_1 \zeta$$

The Attack Point

- The pair-pointwise multiplication of secret key and a part of cipher!

Algorithm 1 KYBER.CPAPKE.Dec(sk, c): decryption

Input: Secret key $sk \in \mathcal{B}^{12 \cdot k \cdot n/8}$
Input: Ciphertext $c \in \mathcal{B}^{d_u \cdot k \cdot n/8 + d_v \cdot n/8}$
Output: Message $m \in \mathcal{B}^{32}$

1: $\mathbf{u} := \mathsf{Decompress}_q(\mathsf{Decode}_{d_u}(c), d_u)$
2: $v := \mathsf{Decompress}_q(\mathsf{Decode}_{d_v}(c + d_u \cdot k \cdot n/8), d_v)$
3: $\hat{\mathbf{s}} := \mathsf{Decode}_{12}(sk)$
4: $m := \mathsf{Encode}_1(\mathsf{Compress}_q(v - \mathsf{NTT}^{-1}(\hat{\mathbf{s}}^T \circ \mathsf{NTT}(\mathbf{u})), 1))$
5: **return** m

Assembly Implementation of Attack Point

- This assembly code repeats 64 time

```
1  macro doublebasemul_frombytes_asm_16_32
2  rptr_tmp, bptr, zeta, poly0, poly2, poly1,
3  poly3, tmp, q, qa, qinv
4
5    ldr \poly0, [\bptr], #4
6    ldr \poly2, [\bptr], #4
7
8    smulwt \tmp, \zeta, \poly1
9    smlabt \tmp, \tmp, \q, \qa
10   smultt \tmp, \poly0, \tmp
11   smlabb \tmp, \poly0, \poly1, \tmp
12   str \tmp, [\rptr_tmp], #4
13
14   smuadx \tmp, \poly0, \poly1
15   str \tmp, [\rptr_tmp], #4
16
17   neg \zeta, \zeta
18
19   smulwt \tmp, \zeta, \poly3
20   smlabt \tmp, \tmp, \q, \qa
21   smultt \tmp, \poly2, \tmp
22   smlabb \tmp, \poly2, \poly3, \tmp
23   str \tmp, [\rptr_tmp], #4
24
25   smuadx \tmp, \poly2, \poly3
26   str \tmp, [\rptr_tmp], #4
27 .endm
```

$a_1\zeta$

$a_1 b_1 \zeta$

$a_0 b_0 + a_1 b_1 \zeta$

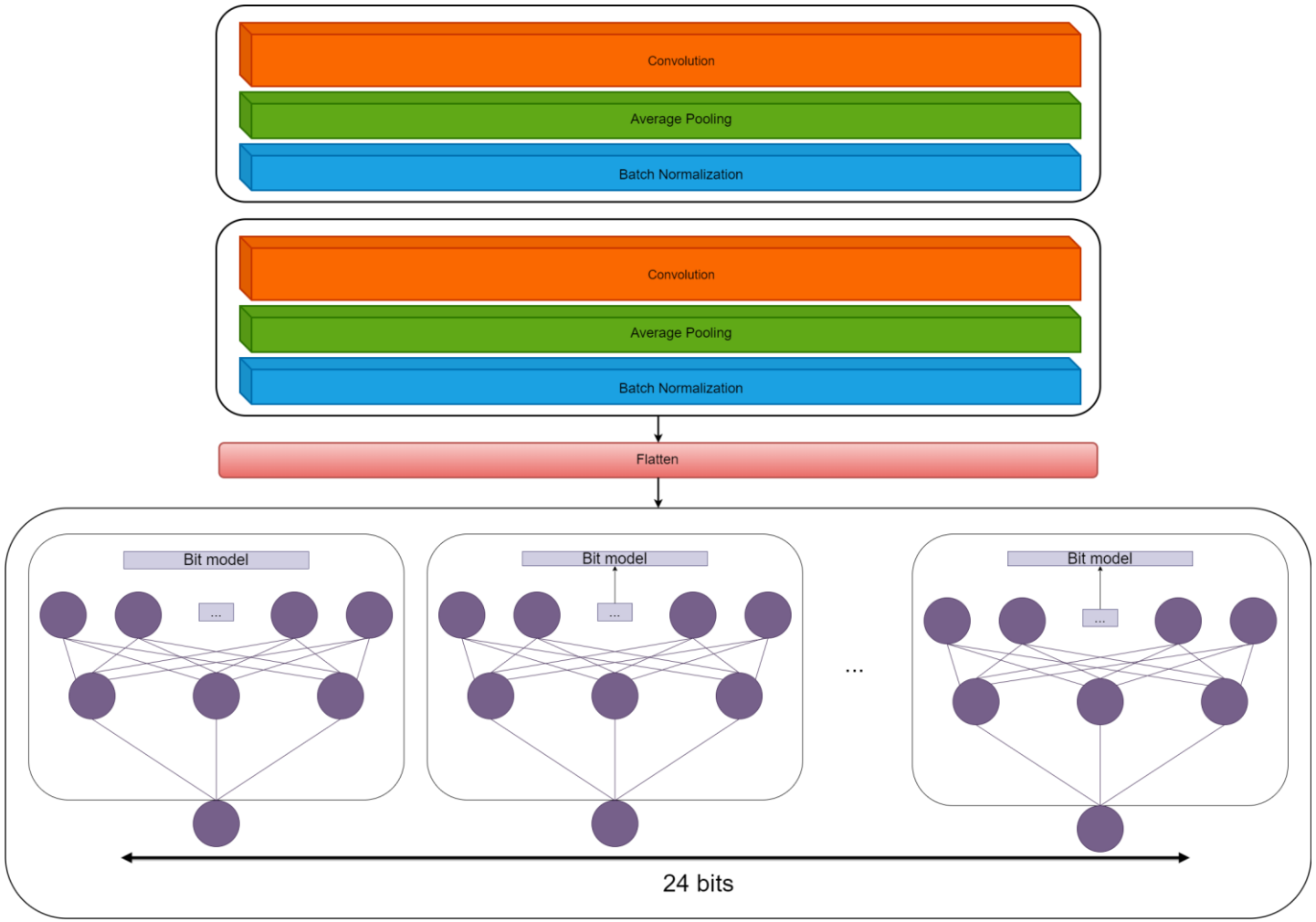
→ c_1

$a_0 b_1 + a_1 b_0$

→ c_0

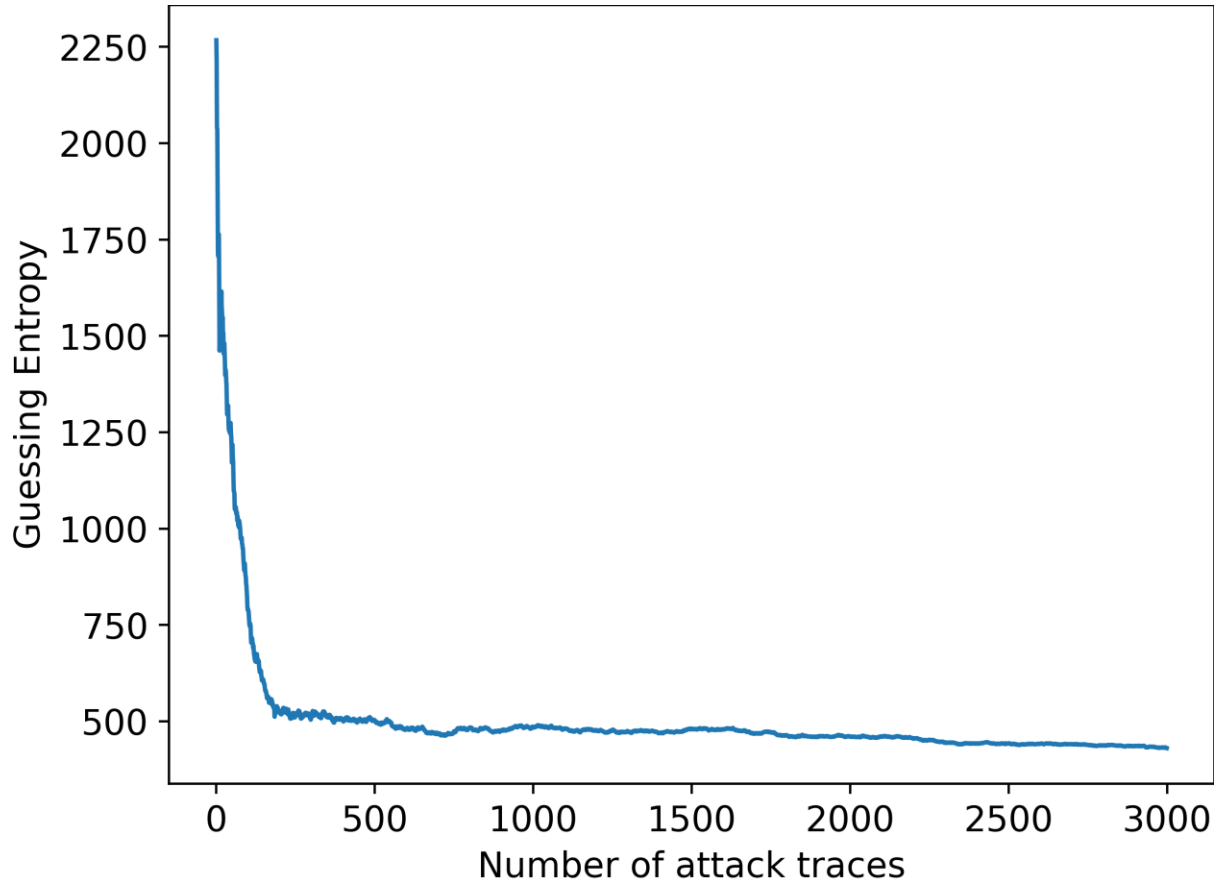
Leakage-Model-Free Deep Learning

- 2 layers CNN
- 24 separate MLP to learn 24 bits separately
- This is multitask learning



Results on Chipwhisperer

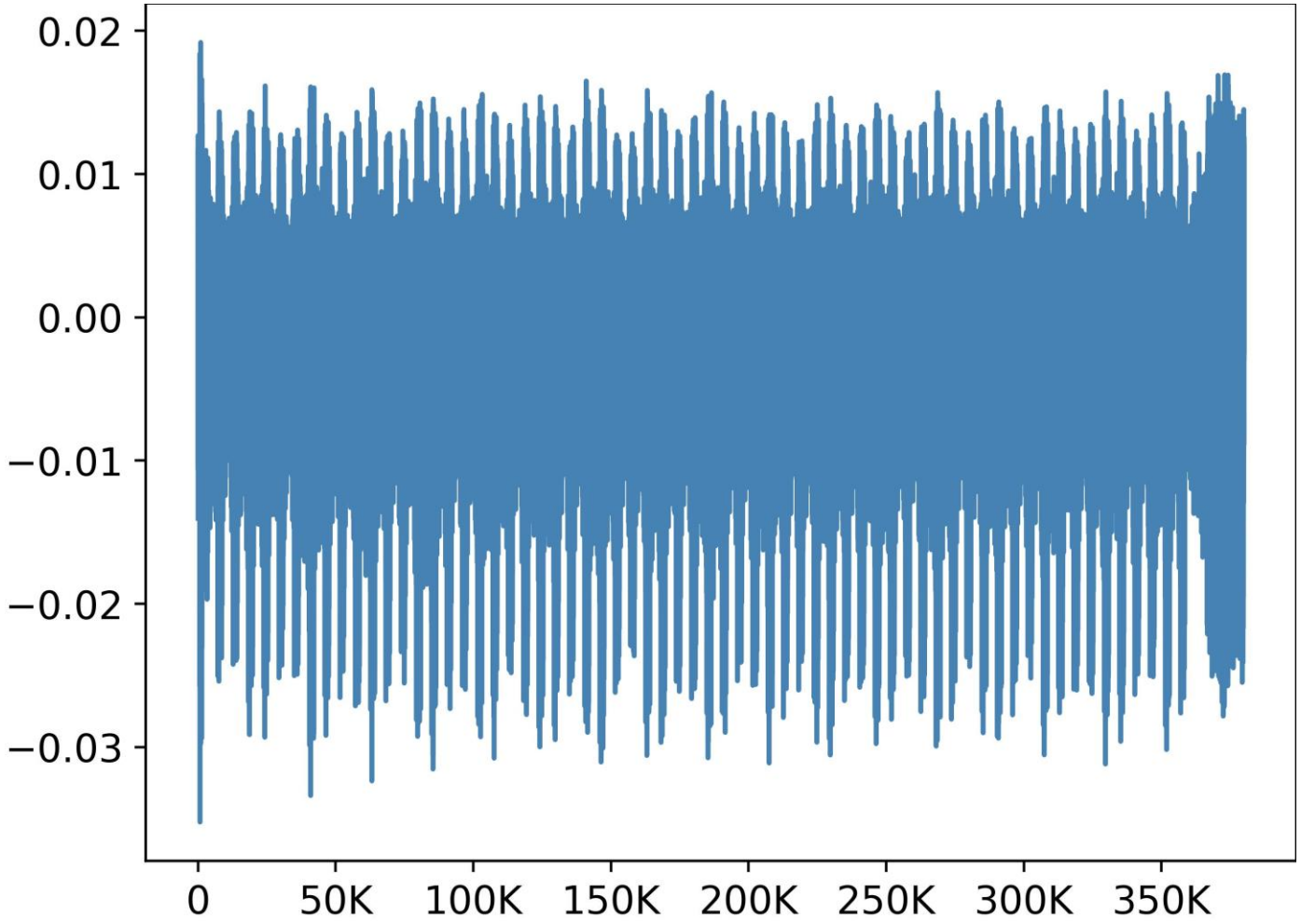
- This is the results on Chipwhisperer



Thank you!

Overview of The Kyber Algorithm

- At first the algorithm is Chosen Plaintext Secure (CPA Secure)



What Have We Done?

- Target implementation: unprotected kyber768 in pqm4
- Used platform: ARM Cortex M4
  - Power analysis: ChipWhisperer-Lite and CW308-STM32F4
  - Power analysis: Lecroy and CW308-STM32F4
- Target procedure: decapsulation
- The attack recovers secret key
- We are now collecting traces

Efficient Implementation of Kyber on RISC-V

PHAM Hoang Nguyen Hien
1st-year PhD Student
16/11/2023

EVIDEN

Content overview

EVIDEN

01 Modular Multiplications

Modular Multiplications

Signed Montgomery multiplication [7]

Input: a, b such that $-2^{n-1} q \le a \cdot b < -2^{n-1} q$ where $0 < q < 2^{n-1}$, $R = q^{-1} \bmod 2^n$

Output: $r = a \cdot b \cdot 2^{-n} \bmod q$ and $-q < r < q$

1. $m = [[a \cdot b]_n \cdot R]_{\pm n}$
2. $t = [m \cdot q]^n$
3. $r = [a \cdot b]^n - t$
4. **return** r

- **3 multiplications**
- **Montgomery representation**

Plantard multiplication [6]

Input: a, b such that $0 < a, b < q$, $q < 2^n/\phi$, $\phi = \frac{1+\sqrt{5}}{2}$, $R = q^{-1} \bmod 2^{2n}$

Output: $r = a \cdot b \cdot (-2^{-2n}) \bmod q$ and $0 \le r \le q$

1. $r = [([[a \cdot b \cdot R]_{2n}]^n + 1) \cdot q]^n$
2. **return** r

- **3 multiplications**
  => Only 2 multiplications
- **Plantard representation**

EVIDEN

02 Kyber on RISC-V

Kyber on RISC-V

General Ideas

- **RISQ-V: Tightly Coupled RISC-V Accelerators for Post-Quantum Cryptography [3]**
  Tim Fritzmann, Georg Sigl, Johanna Sepúlveda

  --- Tightly coupled accelerators for NewHope, Saber and Kyber:
  - ▸ 5 instructions for PQC: modular addition/subtraction/multiplication, CT butterfly, GS butterfly
  - ▸ 1 instruction for a complete round of Keccak
  - ▸ 5 instructions for binomial sampling

  --- PULPino: single-issued, 4-stage 32-bit RISC-V, SystemVerilog
  --- Kyber Round 2: $\eta = 2$

Kyber on RISC-V

General Ideas

- **Improved Plantard Arithmetic for Lattice-based Cryptography [4]**
  Ray C. C. Cheung, Junhao Huang, Jipeng Zhang, Haosong Zhao, Zhe Liu, Çetin Kaya Koç, Donglong Chen

  --- Signed Plantard modular multiplication
  --- Larger input range: $[-q2^{\ell}, q2^{\ell}]$ ---> $[-q^2 2^{2\ell}, q^2 2^{2\ell}]$ ====> **Lazy reduction**

Input: a, b such that $-q2^{\ell} \le a, b \le q2^{\ell}$, $q < 2^{n-\ell-1}$, $R = q^{-1} \bmod^{\pm} 2^{2n}$
Output: $r = a \cdot b \cdot (-2^{-2n}) \bmod q$ and $-q/2 < r < q/2$

---

1. $r = [([[a \cdot b \cdot R]_{2n}]^n + 2^{\ell}) \cdot q]^n$
2. **return** r

Kyber on RISC-V

General Ideas

- **Faster Kyber and Dilithium on the Cortex-M4 [1]**
  Amin Abdulrahman, Vincent Hwang, Matthias J. Kannwischer, Amber Sprenkels

  --- Use Cooley-Tukey (CT) butterfly for both NTT and INTT (with fp registers)
  --- Merge layers of (I)NTT 4-3 (instead of 3-3-1)

Kyber on RISC-V

General Ideas - Summary

Combining all previous approaches:

- 3 instructions for CT butterfly: gp registers/fp registers/light butterfly
- 1 instruction for modular multiplication
- 1 instruction for basemul
- 2 instructions for binomial sampling supporting Kyber Round 3
- Signed Plantard reduction → lazy reduction for (I)NTT
- 4-3 merging for (I)NTT

EVIDEN

03 Results

Results

Simulation using Questa. Results are in number of cycles: k = 1000.

Target	Keypair	Encap.	Decap.
Kyber512 [FSS20/21]	150k/116k	193k/176k	204k/186k
This work	72k (52% / 37,9%)	101k (47,7% / 42,6%)	117k (42,7% / 37,1%)
Kyber768 [FSS20/21]	273k/213k	325k/298k	340k/313k
This work	112k (59% / 47,4%)	148k (54,5% / 50,3%)	171k (49,7% / 45,4%)
Kyber1024 [FSS20/21]	349k/266k	405k/368k	424k/392k
This work	163k (53,3% / 38,7%)	208k (48,6% / 43,5%)	238k (43,9% / 39,3%)

EVIDEN

04 Future work

Future work

- Implement Dilithium with Plantard reduction on (64-bit) RISC-V
- Implement MASKED Kyber and Dilithium on RISC-V

EVIDEN

Conclusion

- Plantard reduction is interesting for Kyber and Dilithium.
- Specific instructions for butterflies help accelerating the implementation.
- 40-50% of improvement compared to [3].

hoang-nguyen-hien.pham@eviden.com
Hoang-nguyen-hien.pham@univ-grenoble-alpes.fr

EVIDEN

References

1. Abdulrahman, A., Hwang, V., Kannwischer, M. J., & Sprenkels, A. (2022, June). Faster kyber and dilithium on the cortex-M4. In *International Conference on Applied Cryptography and Network Security* (pp. 853-871). Cham: Springer International Publishing.
2. Avanzi, R., Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., ... & Stehlé, D. (2019). CRYSTALS-Kyber algorithm specifications and supporting documentation. *NIST PQC Round*, *2*(4), 1-43.
3. Fritzmann, T., Sigl, G., & Sepúlveda, J. (2020). RISQ-V: Tightly coupled RISC-V accelerators for post-quantum cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, 239-280.
4. Huang, J., Zhang, J., Zhao, H., Liu, Z., Cheung, R. C., Koç, Ç. K., & Chen, D. (2022). Improved Plantard arithmetic for lattice-based cryptography. *IACR Transactions on Cryptographic Hardware and Embedded Systems*, *2022*(4), 614-636.
5. Lyubashevsky, V., Ducas, L., Kiltz, E., Lepoint, T., Schwabe, P., Seiler, G., ... & Bai, S. (2020). Crystals-dilithium. *Algorithm Specifications and Supporting Documentation*.
6. Plantard, T. (2021). Efficient word size modular arithmetic. *IEEE Transactions on Emerging Topics in Computing*, *9*(3), 1506-1518.
7. Seiler, G. (2018). Faster AVX2 optimized NTT multiplication for Ring-LWE lattice cryptography. *Cryptology ePrint Archive*.

EVIDEN

Thank you for your attention !

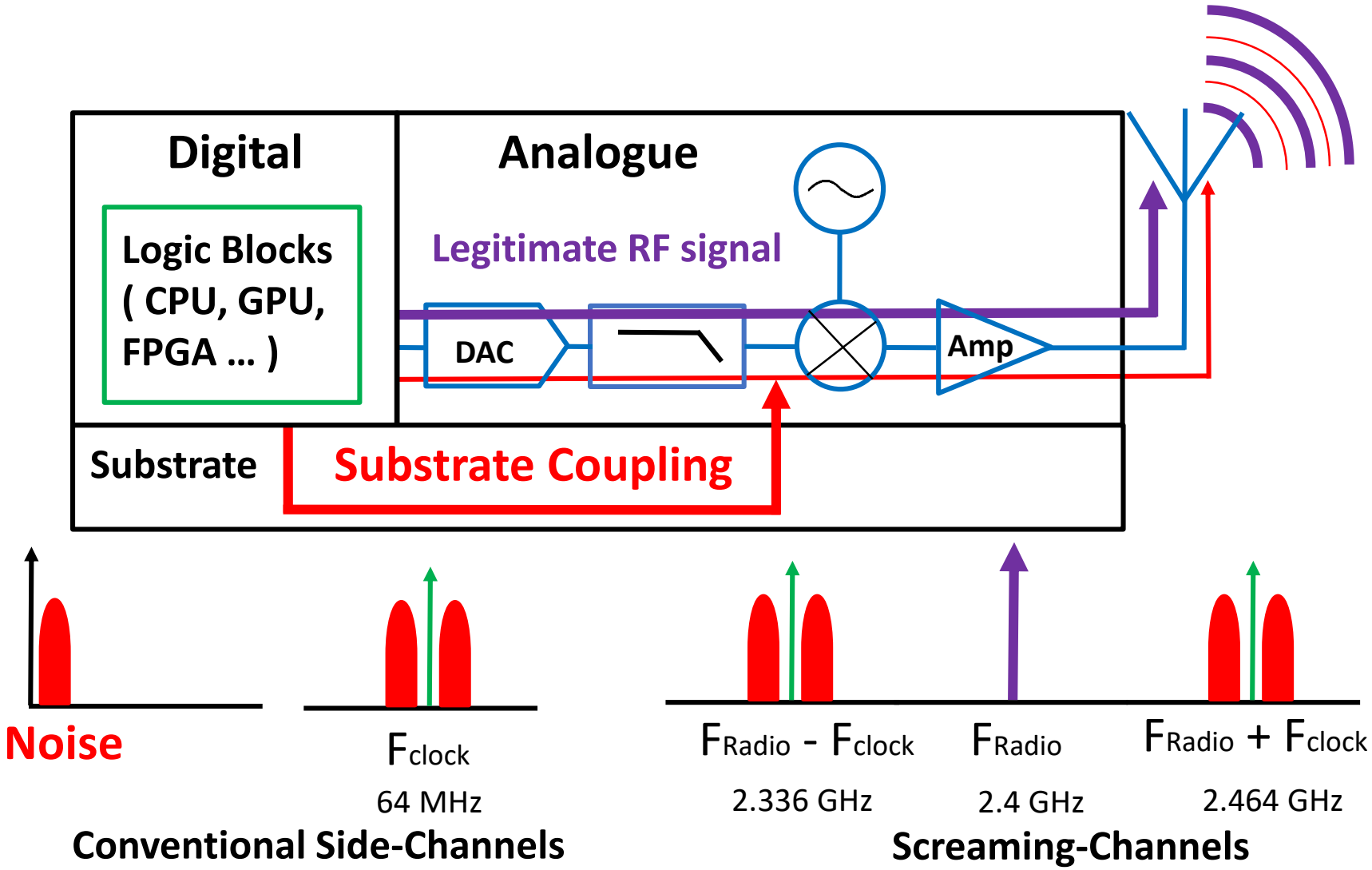
CNRS Images. © Jean-Claude MOSCHETTI / IETR / CNRS Photothèque

The Screaming Gate Array: Study and characterization of IP data leakages in mixed-signal FPGA SoC

Jeremy Guillaume[1]
Directeur de thèse: Maxime Pelcat[2]
Co-directeur: Amor Nafkha[1]
Encadrant: Ruben Salvador[3]

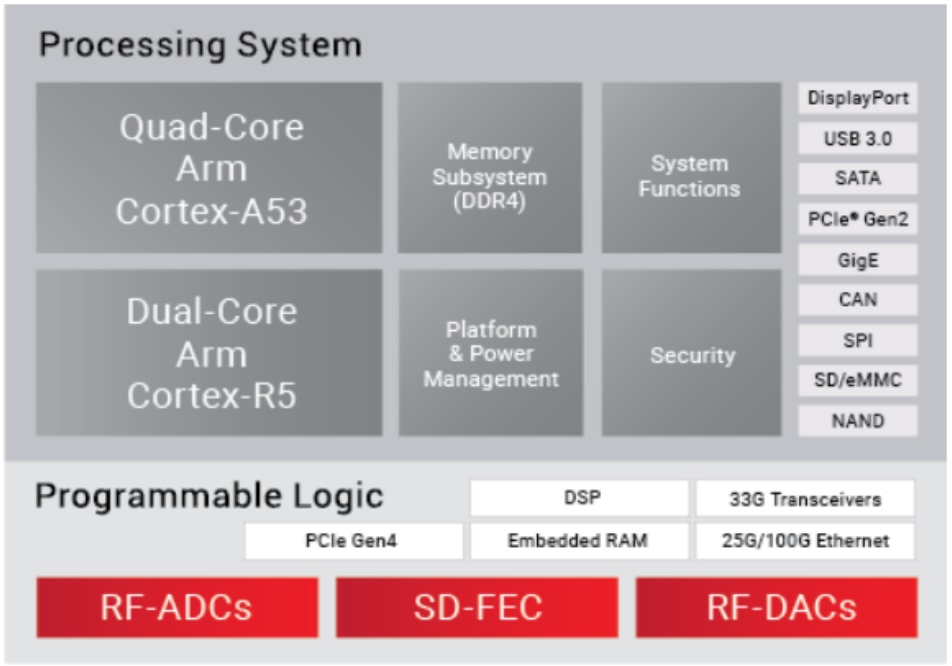
[1] Equipe ASIC, CentraleSupelec, IETR - UMR CNRS 6164
[2] Equipe VAADER, INSA Rennes, IETR - UMR CNRS 6164
[3] Equipe CIDRE, CentraleSupelec, IRISA - UMR CNRS 6074

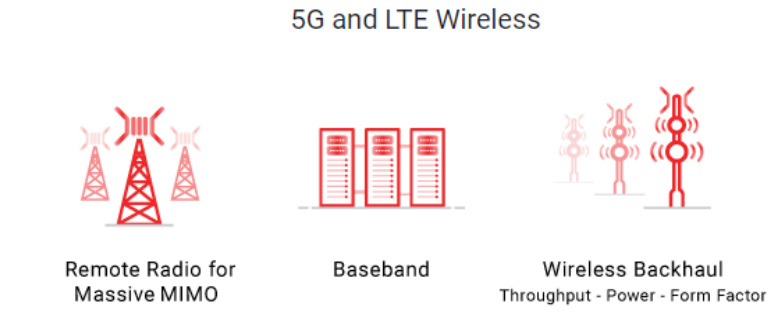
Screaming channel attacks

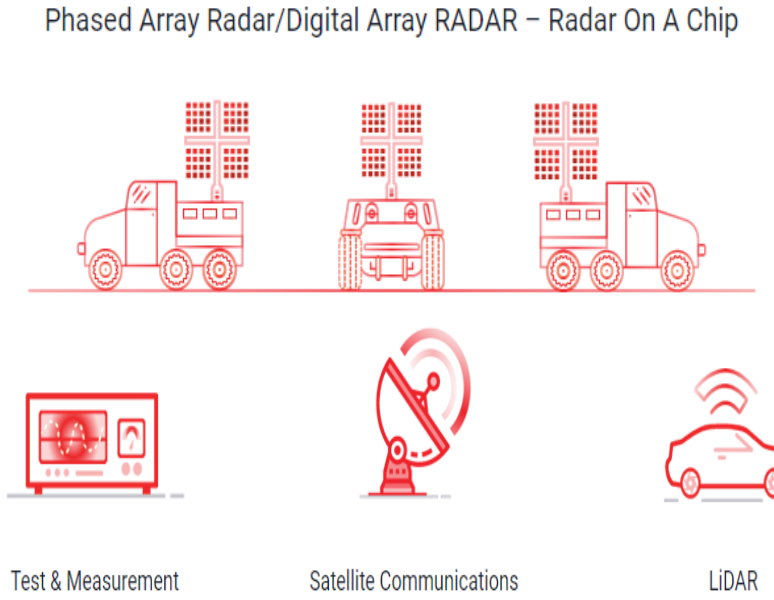


[1] G. Camurati, S. Poeplau, M. Muench, T. Hayes, and A. Francillon, "Screaming Channels: When Electromagnetic Side Channels Meet Radio Transceivers," ACM SIGSAC, 2018

RFSoC (Radio Frequency System on Chip)







Sources: [Zynq UltraScale+ RFSoC (xilinx.com)]

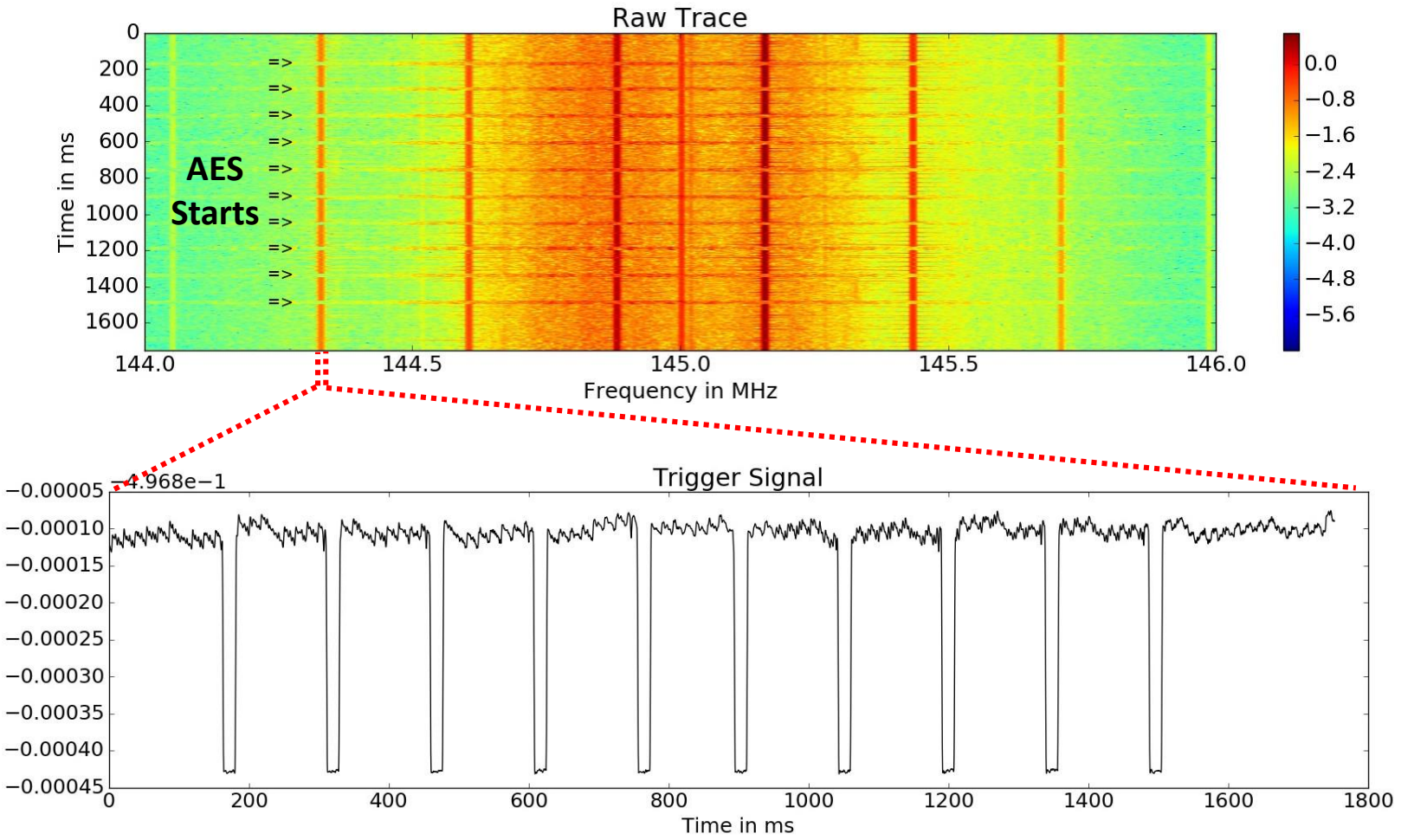
Trace synchronisation

- Problematic: How to **remove the need of a trigger signal** in context of screaming-channel attacks?

- Method used in all previous works on screaming-channel attacks:

  Frequency trigger mechanism

Frequency Trigger Mechanism



Frequency Trigger Mechanism

Advantage:

- Removes the need of a trigger signal

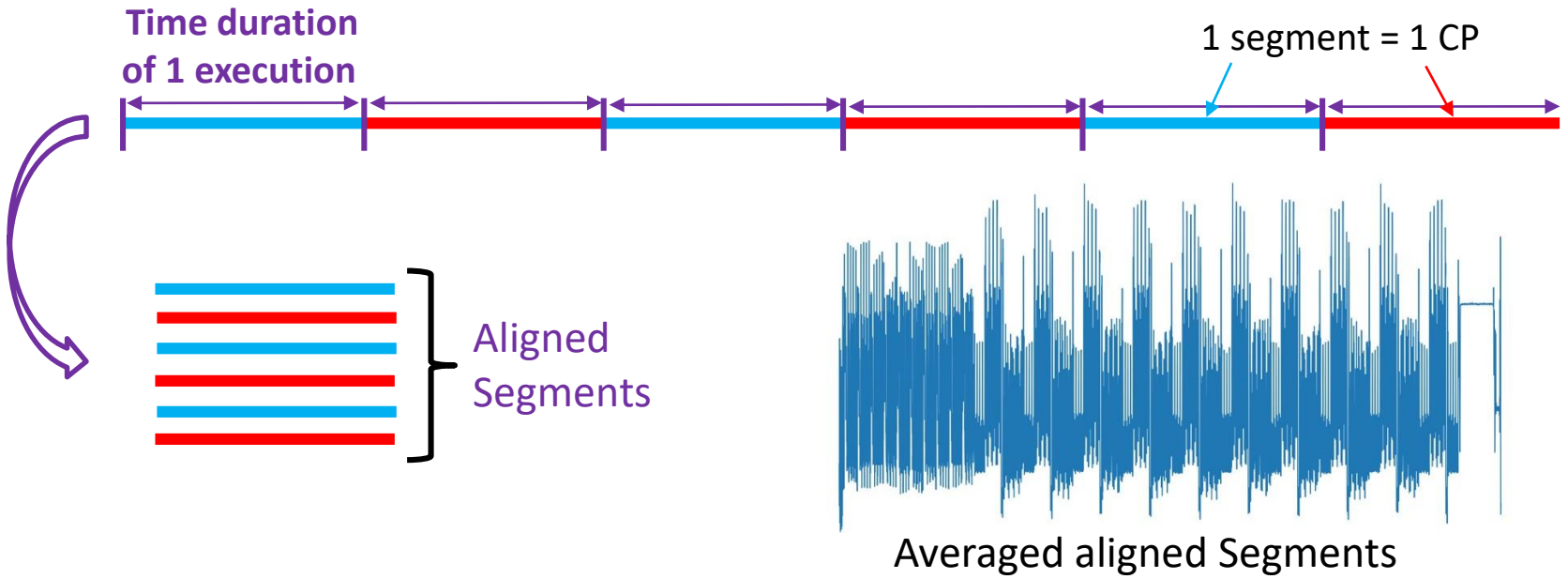
Limitations:

- Suppose that such a frequency component (FC) exists on the target device.
- More complex/costly setup to analyze the frequency spectrum to find the FC.
- Some parameters like the threshold are very sensitive and need to be well adjusted, making the method unstable and difficult in practice

Virtual trigger (VT) [3]

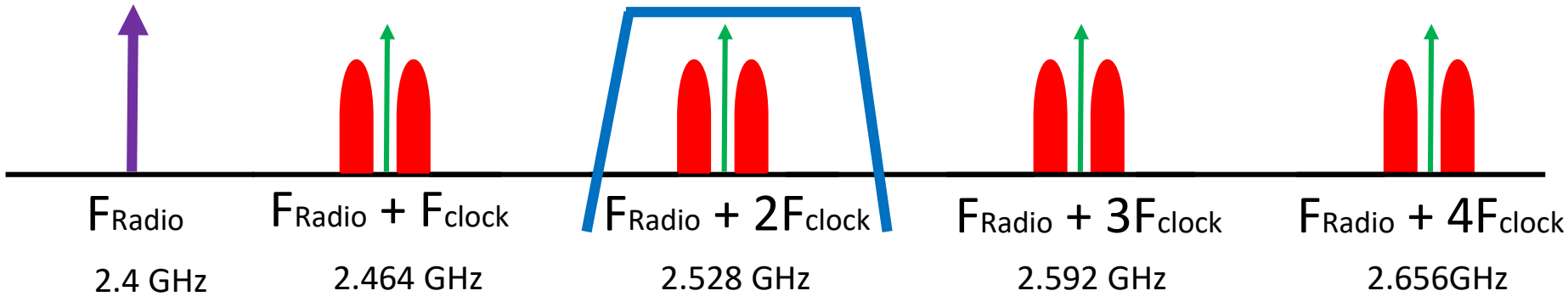
Concept

- A series of Cryptographic Process (CP) is executed recursively without interruptions
- By knowing precisely enough the CP execution time: possibility to create a Virtual Trigger (VT) that points to a common instant in each CP
- Trace segmentation: separate the CP segments using VT



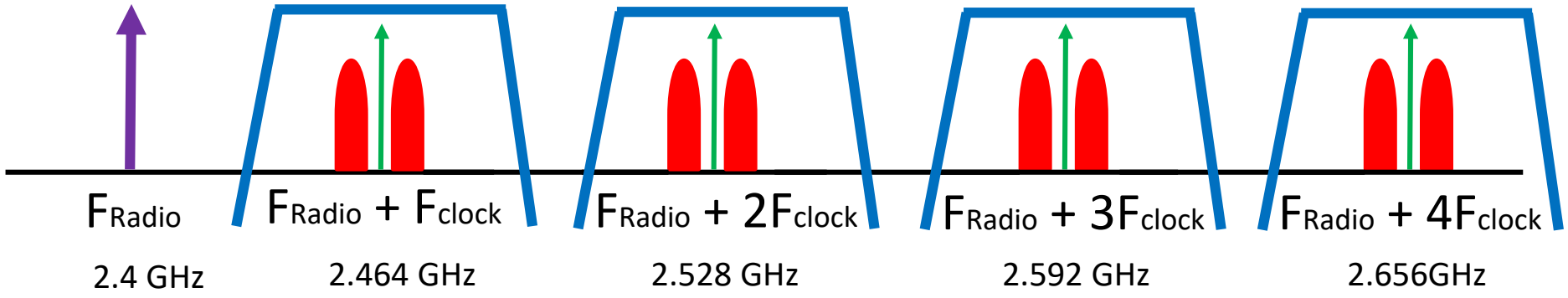
[2] J.Guillaume, M.Pelcat, A.Nafkha, R.Salvador, "Virtual Triggering: a Technique to Segment Cryptographic Processes in Side-Channel Traces," SIPS, 2022.

Frequency diversity



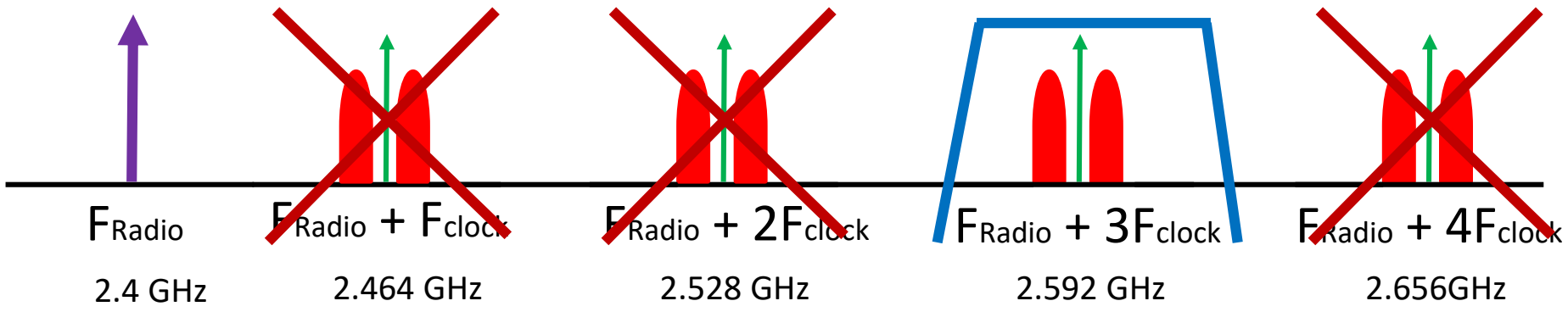
- The leakage is present at each harmonic of the digital clock frequency
- Initial works on screaming channel used only the second at 2,528 GHz

Frequency diversity



- Could the combination of multiple frequency improve the screaming channel attack?

Frequency diversity



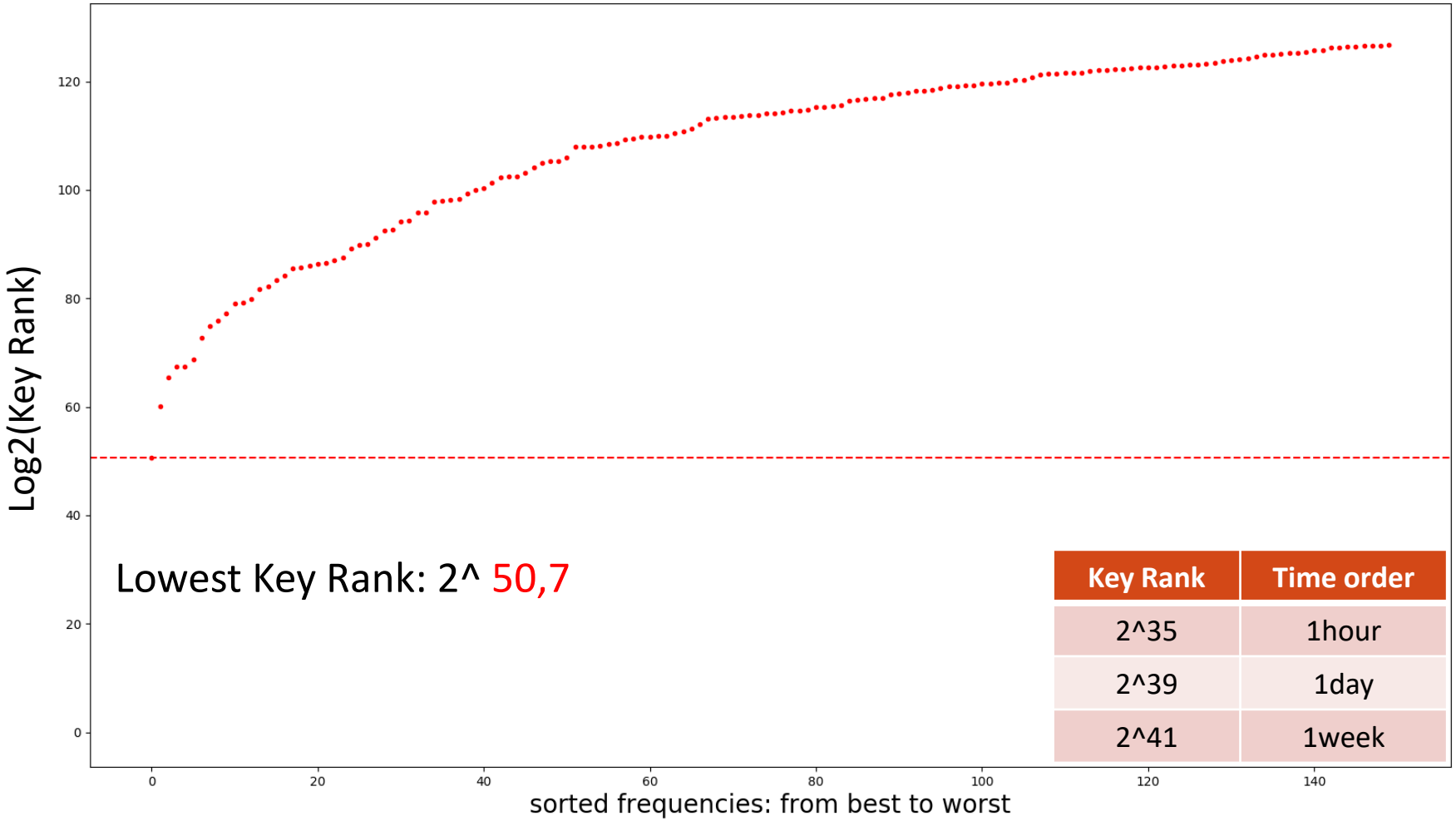
- The leakage is collected at the harmonics with a distance of 30 cm
- Only one harmonic is both unpolluted and sufficiently strong to mount a successful attack
- The harmonics are therefore not enough to provide frequency diversity

Frequency diversity

- Objective: **Keeping the attack feasible** in a polluted environment where all the harmonics are covered by noise.
- **Detection of frequencies** where leakage is present
- **Demontrating attacks** at other frequencies than the harmonics

Frequency combination

Combine the best frequency with all others

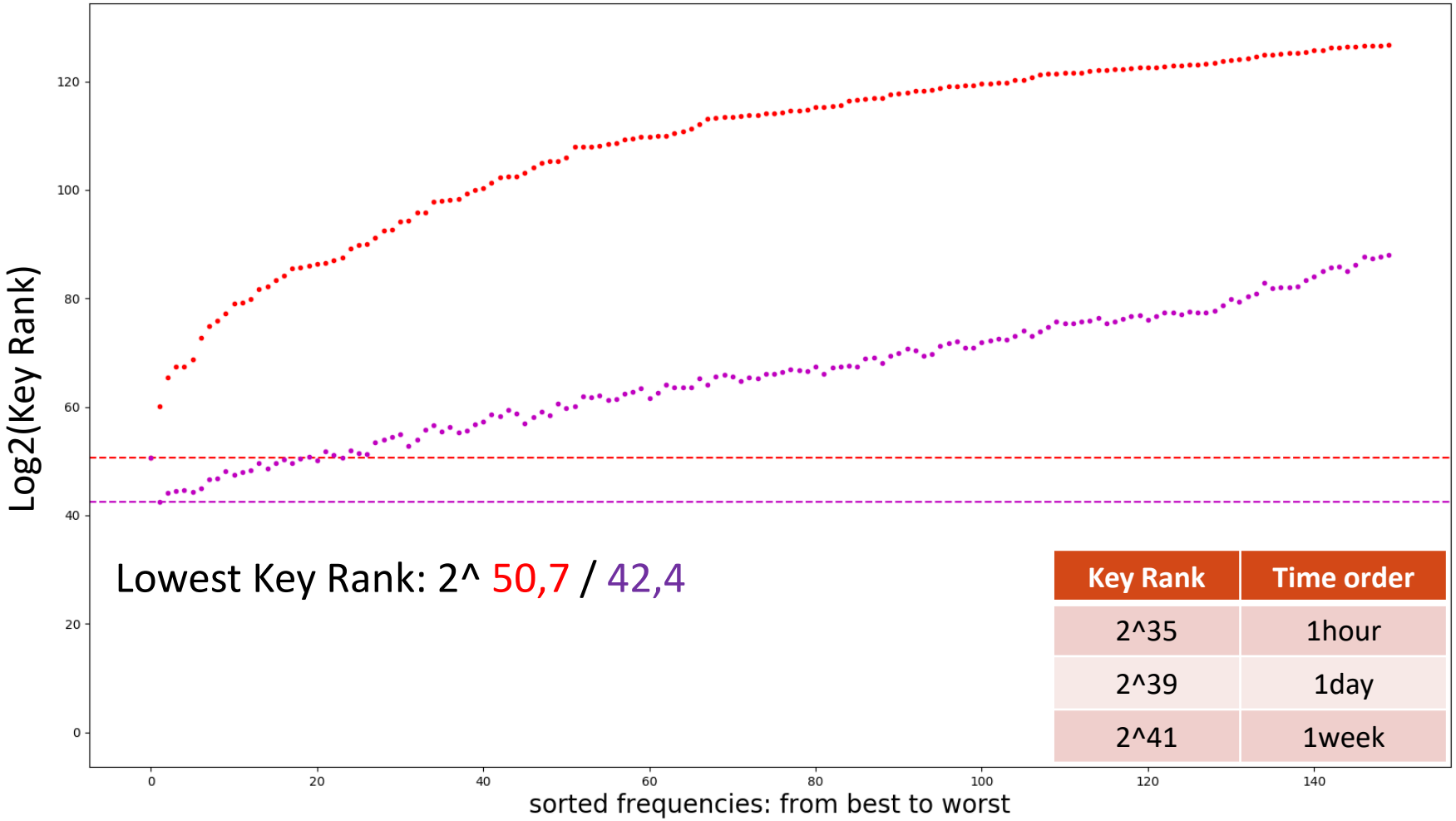


Lowest Key Rank: 2^ 50,7

Key Rank	Time order
2^35	1hour
2^39	1day
2^41	1week

Frequency combination

Combine the best frequency with all others

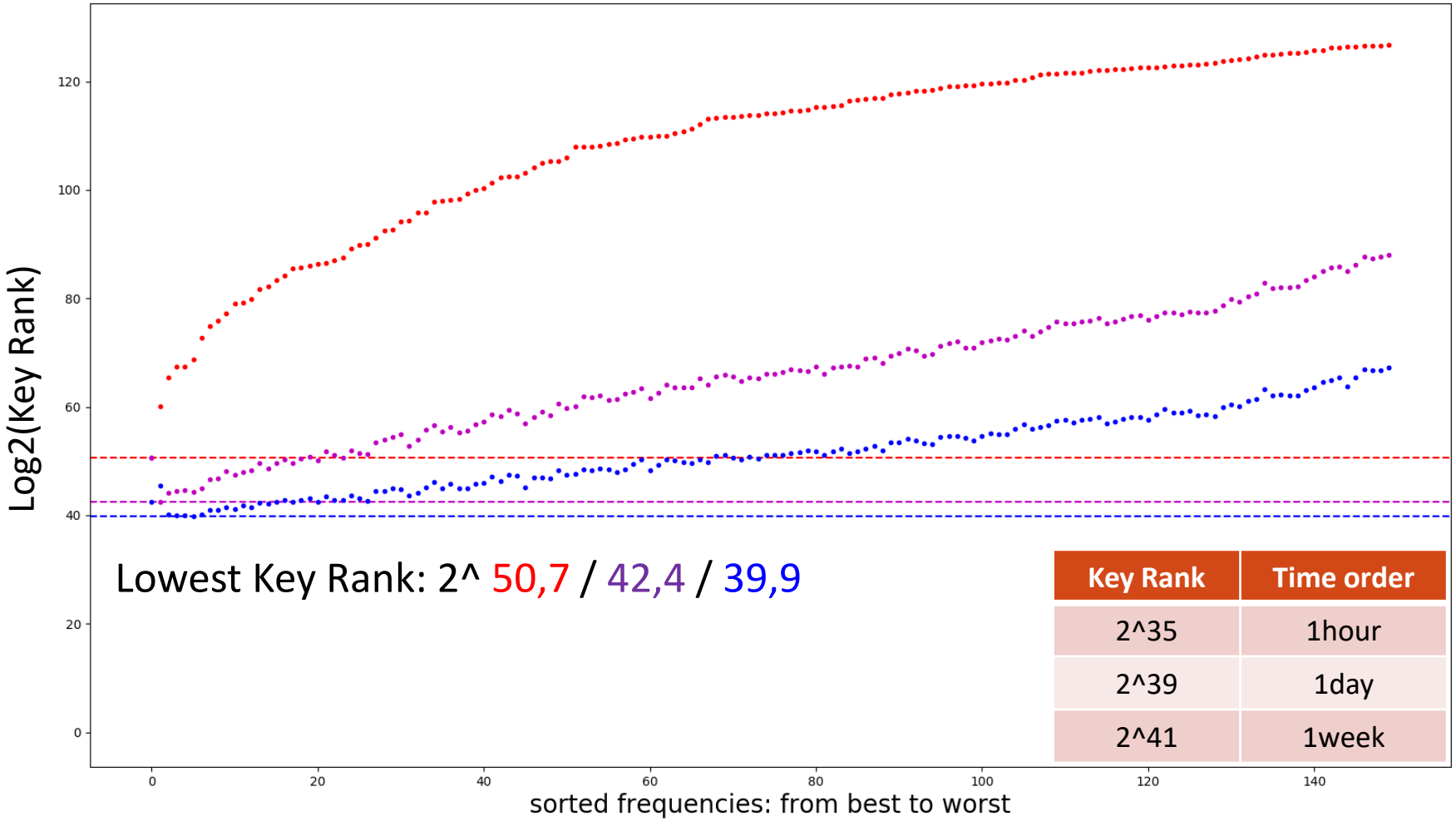


Lowest Key Rank: 2^ 50,7 / 42,4

Key Rank	Time order
2^35	1hour
2^39	1day
2^41	1week

Frequency combination

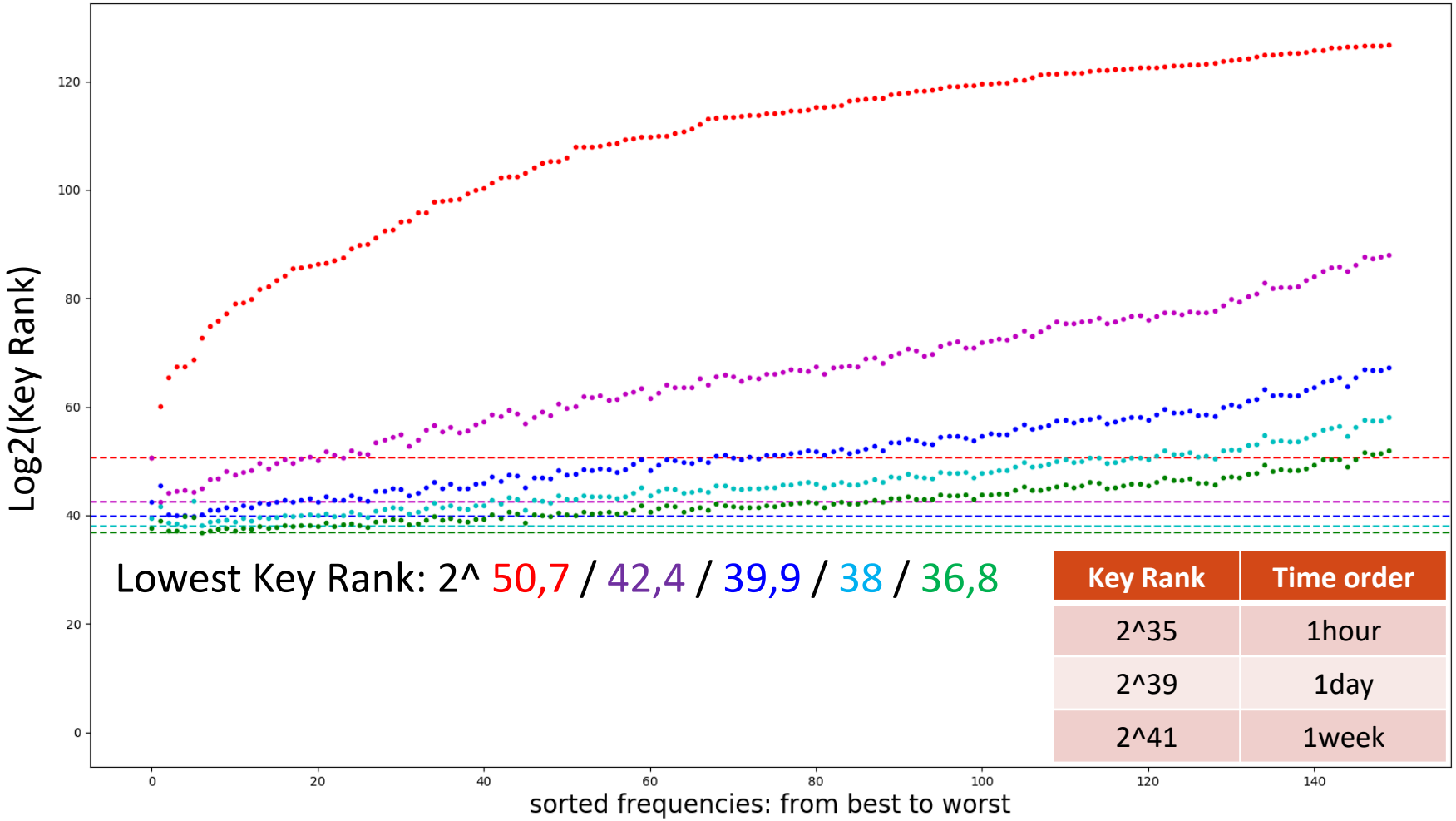
Combine the best frequency with all others



Key Rank	Time order
2^35	1hour
2^39	1day
2^41	1week

Frequency combination

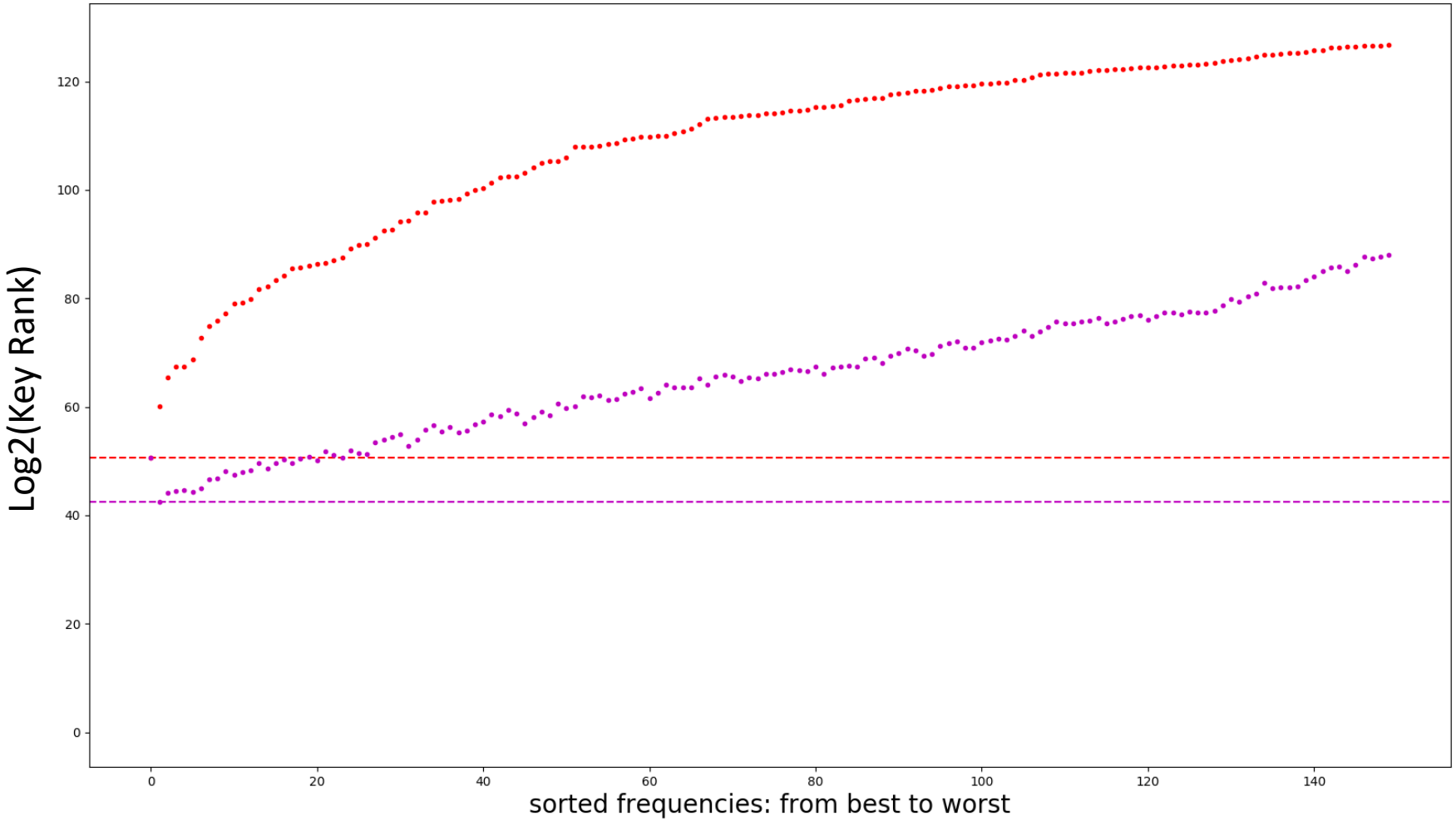
Combine the best frequency with all others



Key Rank	Time order
2^35	1hour
2^39	1day
2^41	1week

Frequency combination

Can weak frequencies be used to improve the attack?



CNRS Images. © Jean-Claude MOSCHETTI / IETR / CNRS Photothèque

IETR

Thank you !

E-mail:
Jeremy.guillaume@centralesupelec.fr

cnrs
CentraleSupélec
INSA RENNES
Nantes Université
Université de Rennes

Universiteit Leiden
The Netherlands

PROACT Project (Hardware)

University of Amsterdam
CARDIS Conference
NOV. 16, 2023

Abolfazl Sajadi

liacs

The Faculty of Science and the Leiden Institute of Advanced Computer Science

PROACT

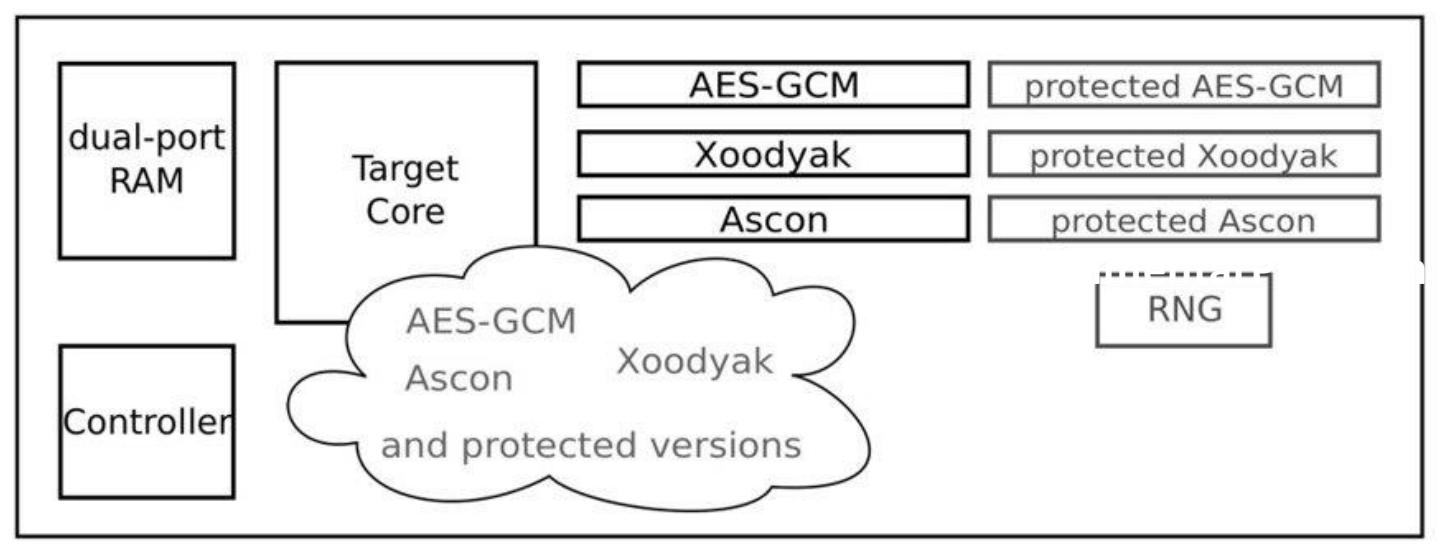
(Physical Attack Resistance of Cryptographic Algorithms and Circuits with Reduced Time to Market)

- **Increase [Physical] Security strength of Cryptographic Algorithms**
  - **Maximize the Probability of First-Time-Right Cryptographic Hardware Implementation**
  - **Accelerate Time-to-Market**
  - **Improve Design Quality**

- **Implement and Validate Pre-Silicon Leakage Analysis Tools (Simulators)**
  - **Leakage analysis (Power SCA) → Need for an Experimental Platform**
    - **[ASIC Tape-out]**

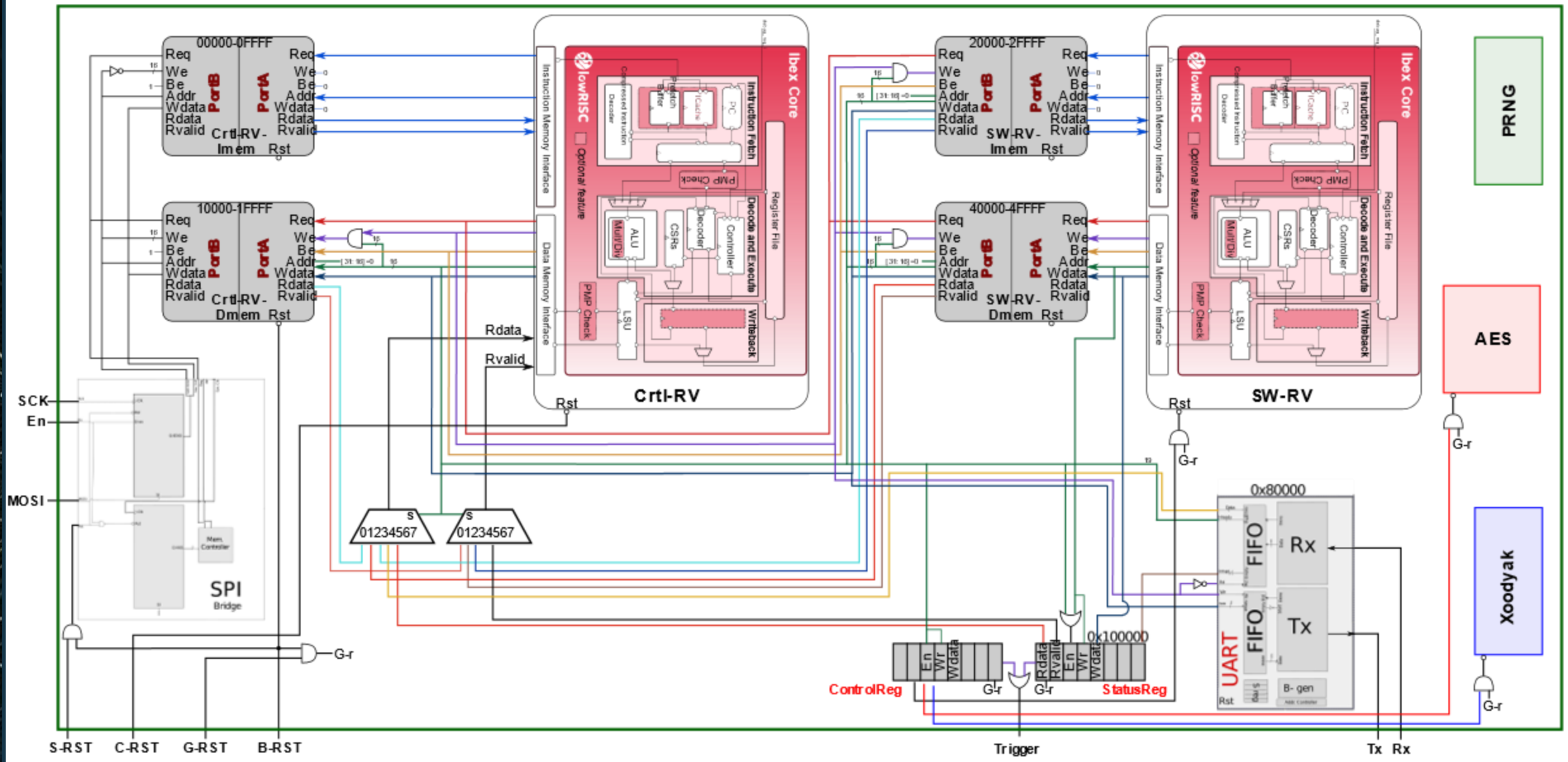
Funding from the Dutch Research Council (NWO) through the PROACT project (NWA.1215.18.014)

Experimental Platform



Leakage:

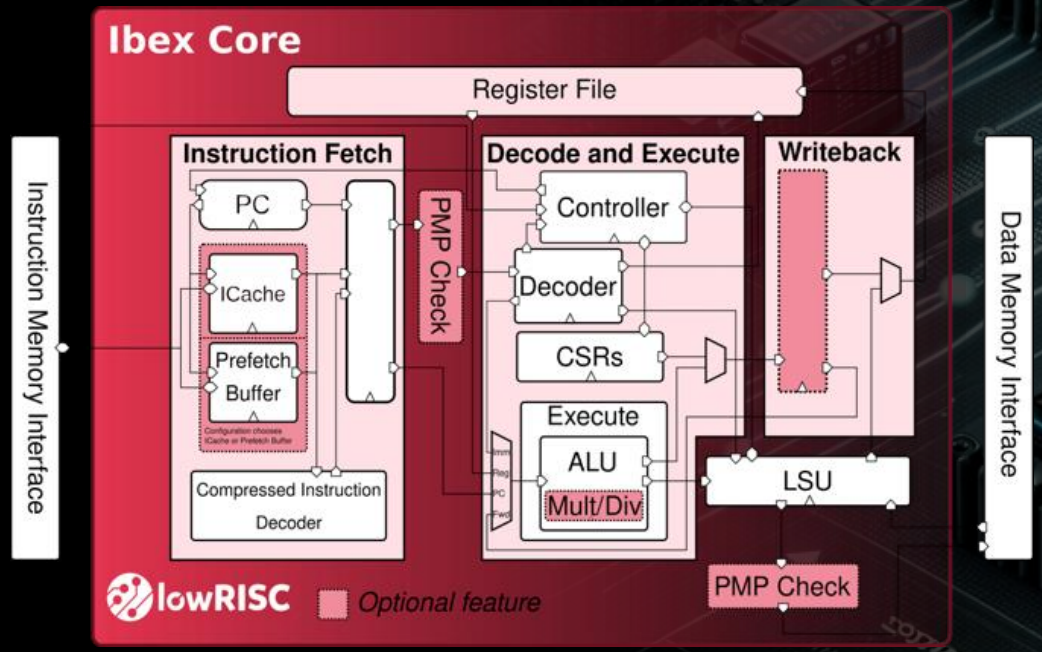
- Leakage for SW running on RISC-V
- Leakage for Co-processors
- Leakage for RISC-V ISA Extensions
- Leakage after adding countermeasures


00000-0FFFF
10000-1FFFF
20000-2FFFF
40000-4FFFF
Req
We
Be
Addr
Wdata
Rdata
Rvalid
PortA
PortB
Crtl-RV-Imem
Crtl-RV-Dmem
SW-RV-Imem
SW-RV-Dmem
Rst
Ibex Core
lowRISC
Optional feature
Instruction Memory Interface
Data Memory Interface
Instruction Fetch
Prefetch Buffer
Compressed Instruction Decoder
PC
ICache
PMP Check
Decode and Execute
Decoder
Controller
ALU
MultDiv
CSRs
LSU
Writeback
Register File
Crtl-RV
SW-RV
Rdata
Rvalid
PRNG
AES
G-r
SCK
En
MOSI
SPI
Bridge
Mem. Controller
S
01234567
0x80000
0x100000
UART
FIFO
Rx
Tx
B- gen
Xoodyak
ControlReg
StatusReg
Trigger
Tx
Rx
S-RST
C-RST
G-RST
B-RST

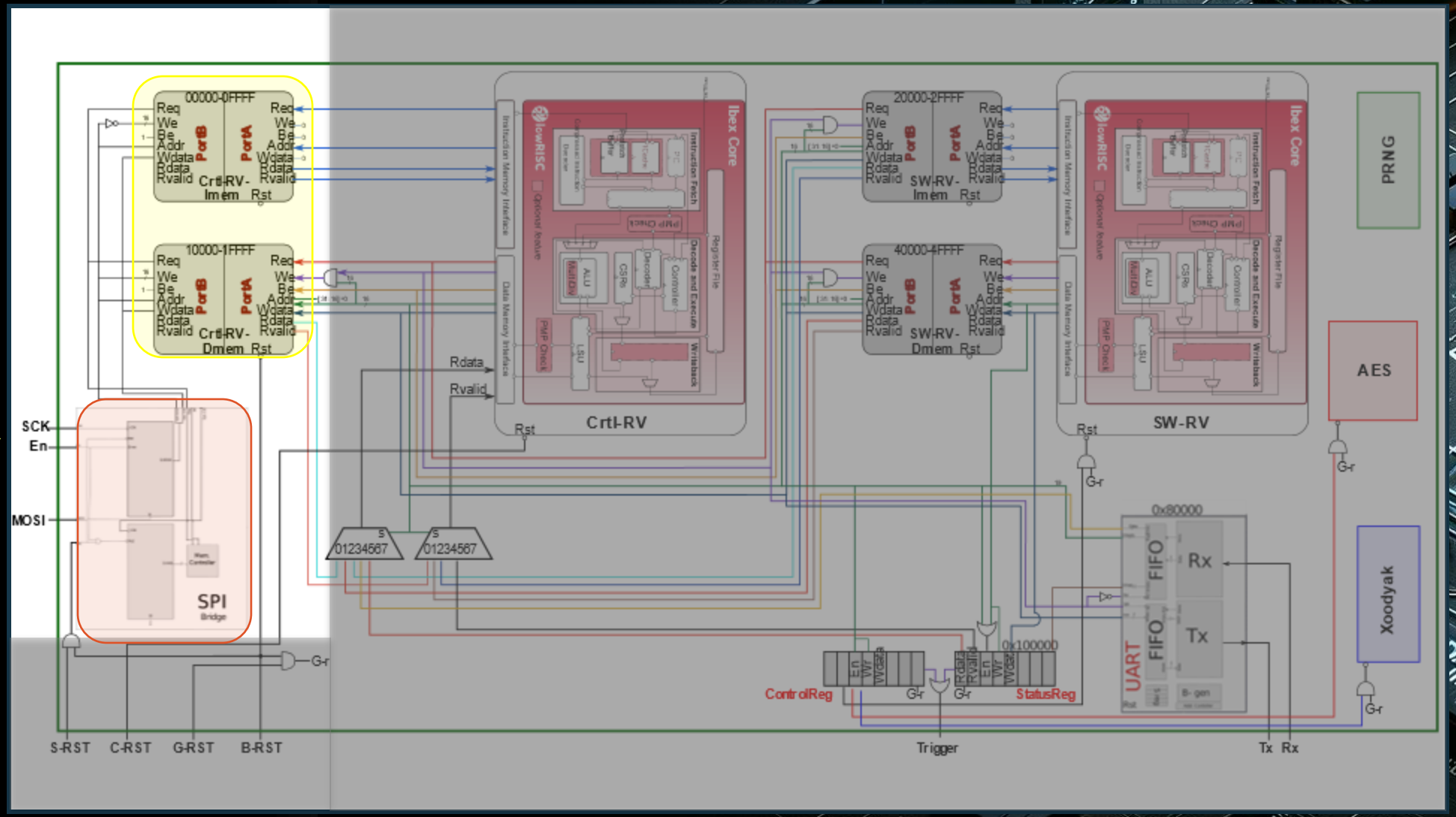
lowRISC

Two Ibex RISC-V Cores

- ✓ Controller (Ctrl-RV)
- ✓ Software Target (SW-RV)

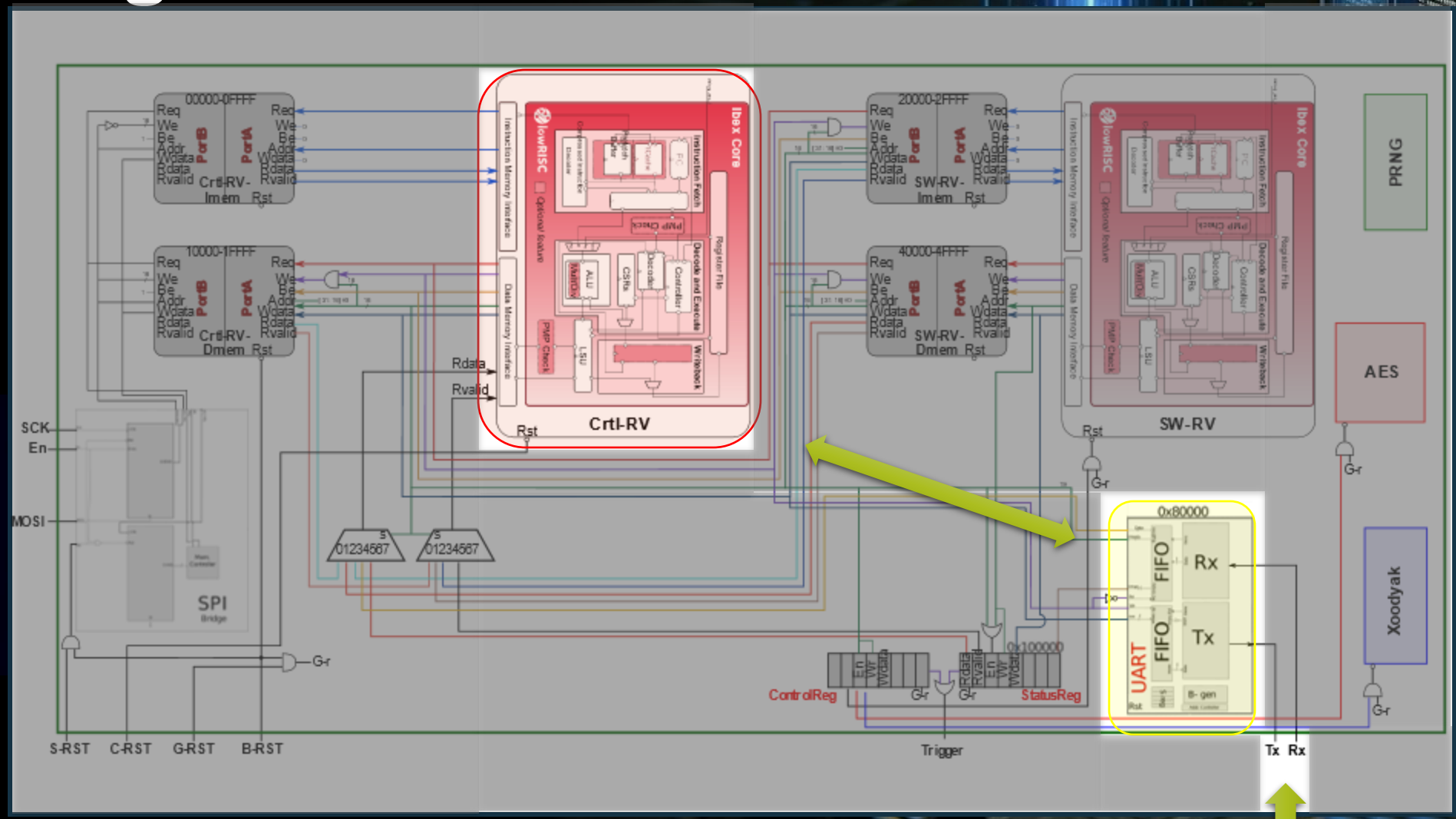


SPI Bridge


00000-0FFFF
10000-1FFFF
20000-2FFFF
40000-4FFFF
Req
We
Be
Addr
Wdata
Rdata
Rvalid
PortB
PortA
Crtl-RV-
Imem Rst
Crtl-RV-
Dmem Rst
SW-RV-
Imem Rst
SW-RV-
Dmem Rst
Ibex Core
lowRISC
Crtl-RV
SW-RV
Rst
Rdata
Rvalid
PRNG
AES
Xoodyak
SCK
En
MOSI
SPI
Bridge
S
01234567
0x80000
0x100000
FIFO
Rx
Tx
UART
ControlReg
StatusReg
Trigger
S-RST
C-RST
G-RST
B-RST
Tx
Rx

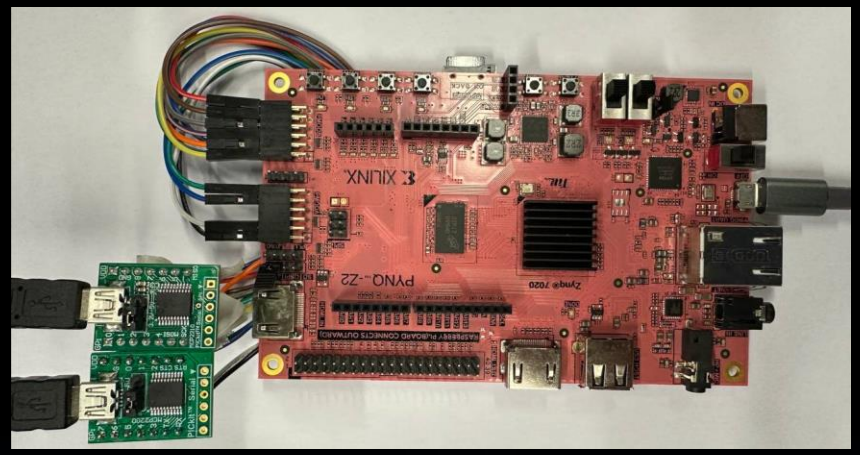

MICROCHIP
MCP2200

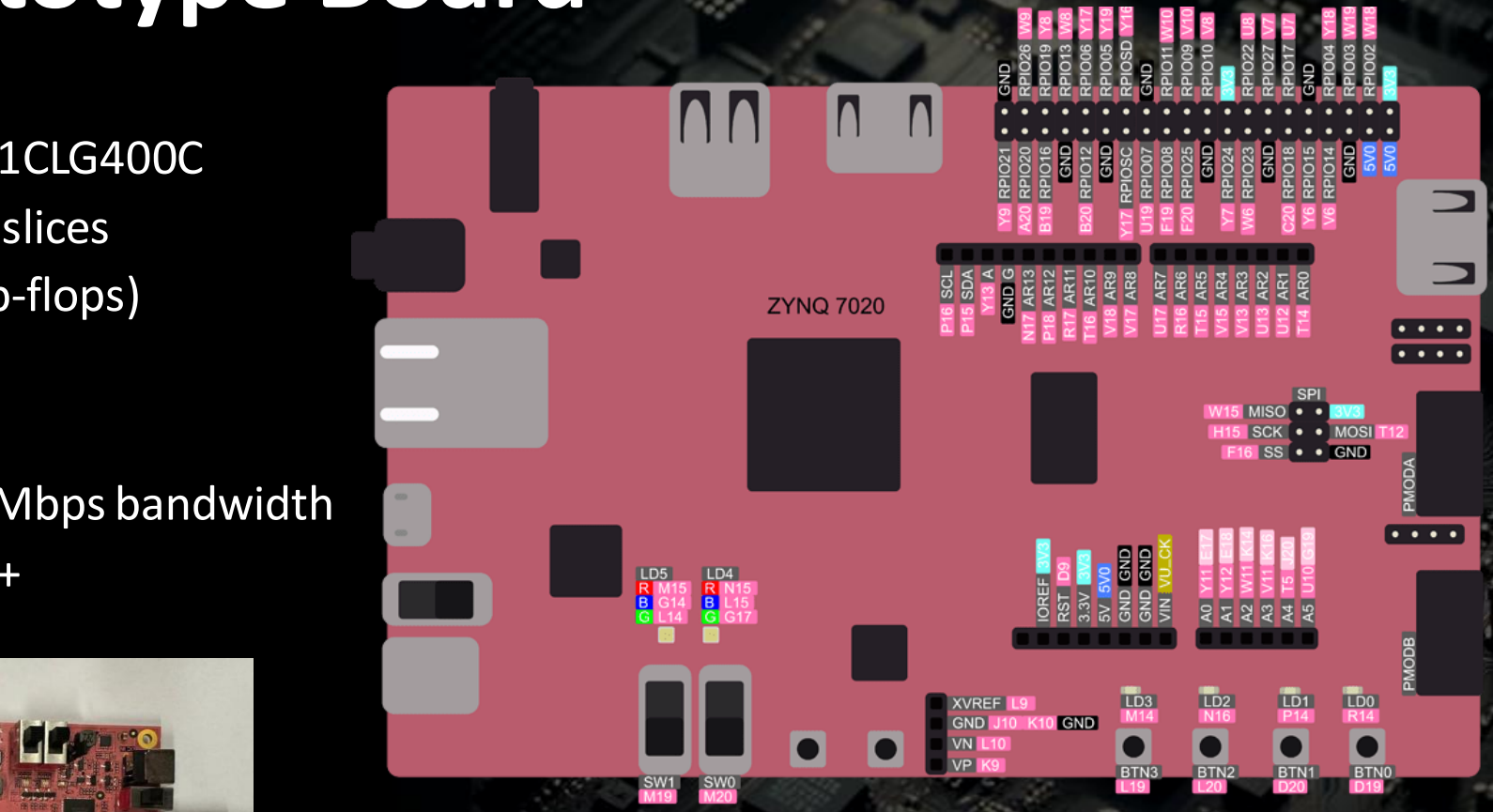
UART Bridge


00000-0FFFF
Req
We
Be
Addr
Wdata
Rdata
Rvalid
PortB
PortA
Ctrl-RV-
Imem
Rst
10000-1FFFF
Ctrl-RV-
Dmem
20000-2FFFF
SW-RV-
40000-4FFFF
lowRISC
Ibex Core
Instruction Fetch
Register File
Decode and Execute
Writeback
ALU
CSRs
Decoder
Controller
LSU
PMP Check
Instruction Memory Interface
Data Memory Interface
Rdata
Rvalid
Rst
Ctrl-RV
SW-RV
PRNG
AES
Xoodyak
SCK
En
MOSI
SPI
01234567
G-r
0x80000
UART
FIFO
Rx
Tx
B- gen
0x100000
ControlReg
StatusReg
Trigger
S-RST
C-RST
G-RST
B-RST
Tx
Rx
MCP2200
MICROCHIP
MCP2210

Current Prototype Board

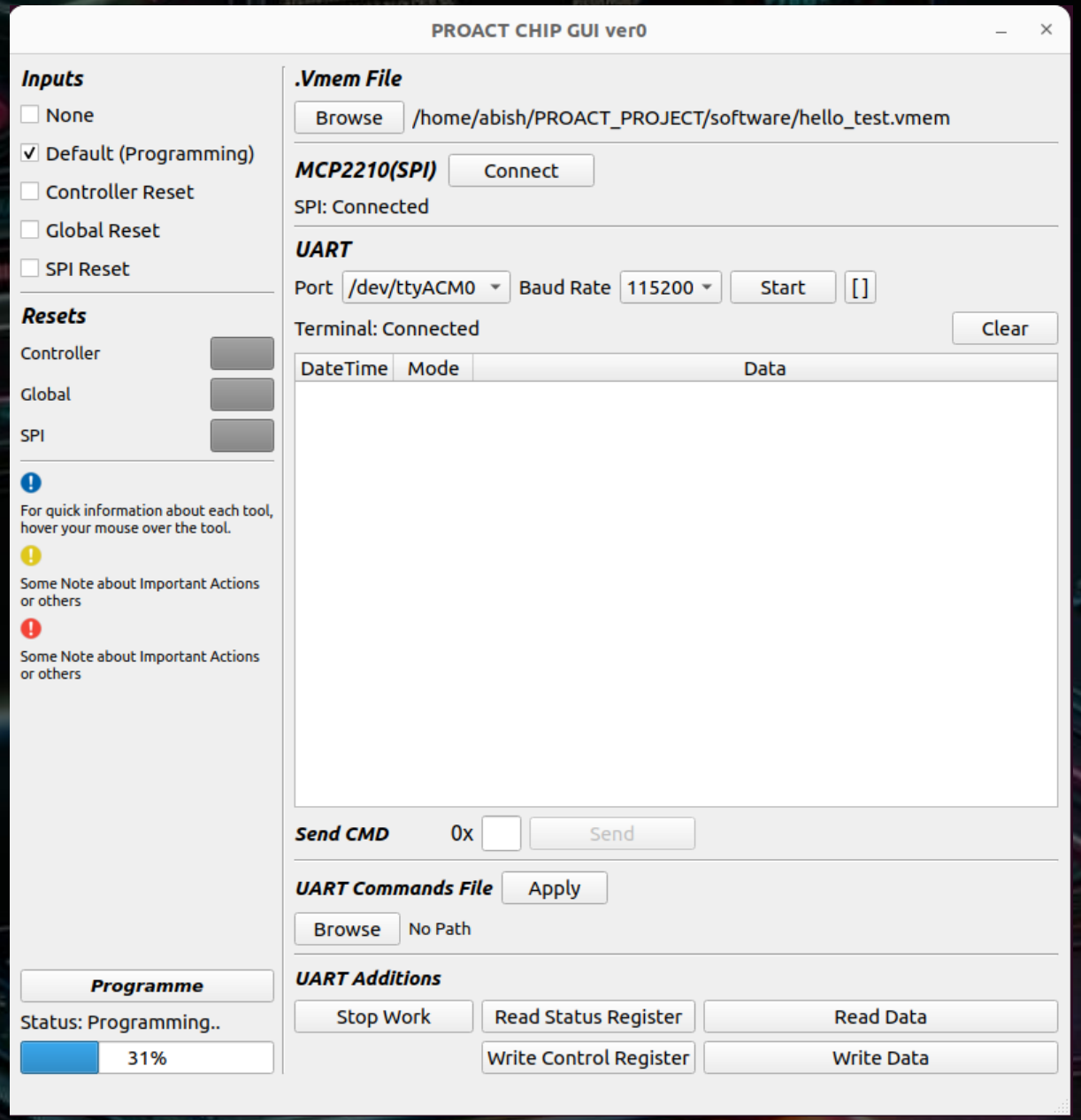
- **Part Number:** XC7Z020-1CLG400C
- **Logic Cells:** 13,300 logic slices
- (4 6-input LUTs and 8 flip-flops)
- **Block RAM:** 630 KB
- **DSP Slices:** 220
- **DDR3:** 512 MB w/ 1050Mbps bandwidth
- **Internal clock:** 650 MHz+





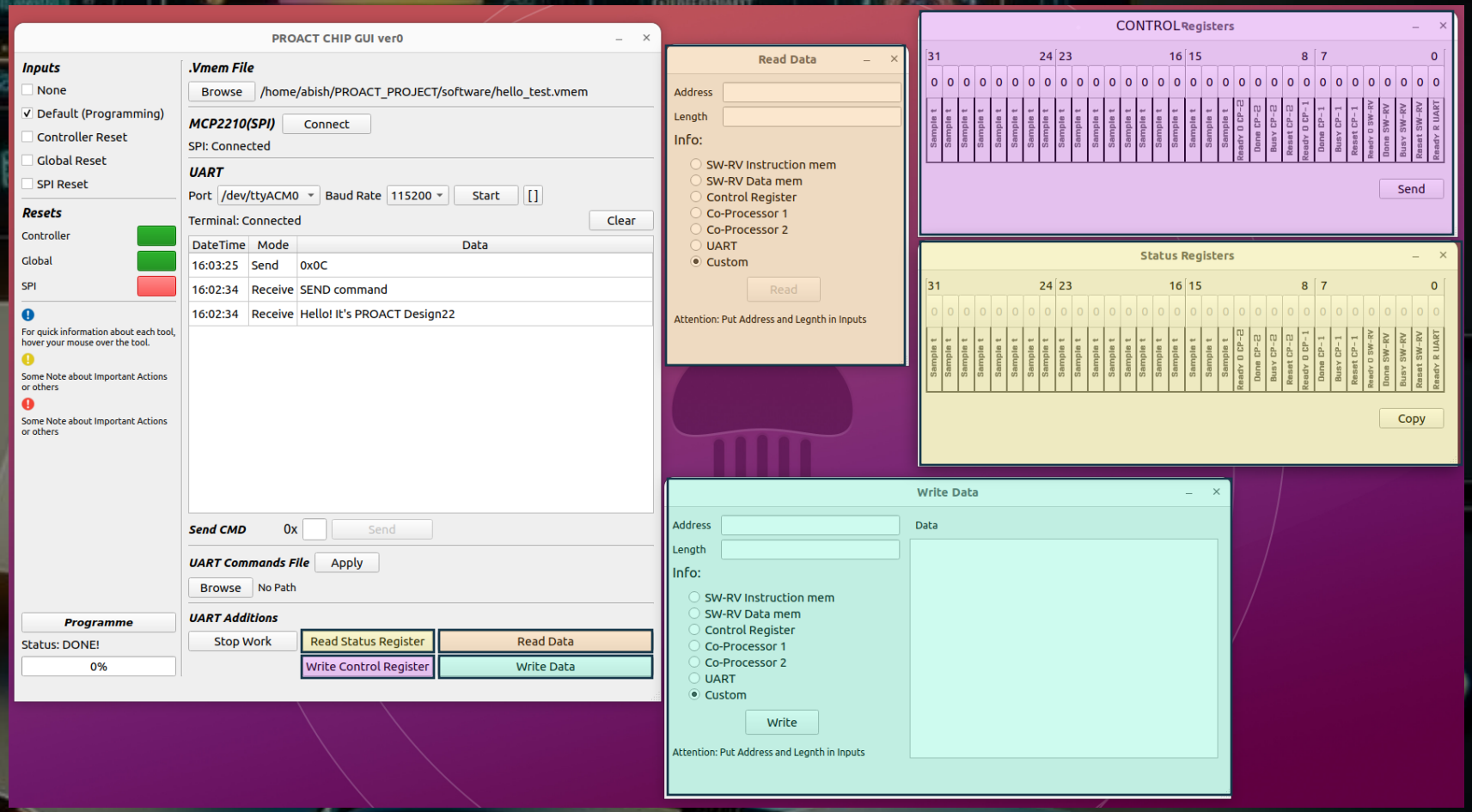
GUI

- Developed with PyQT
- Control Resets Signals
- Recognize and connect to SPI and UART Automatically
- Send Controller Program via SPI



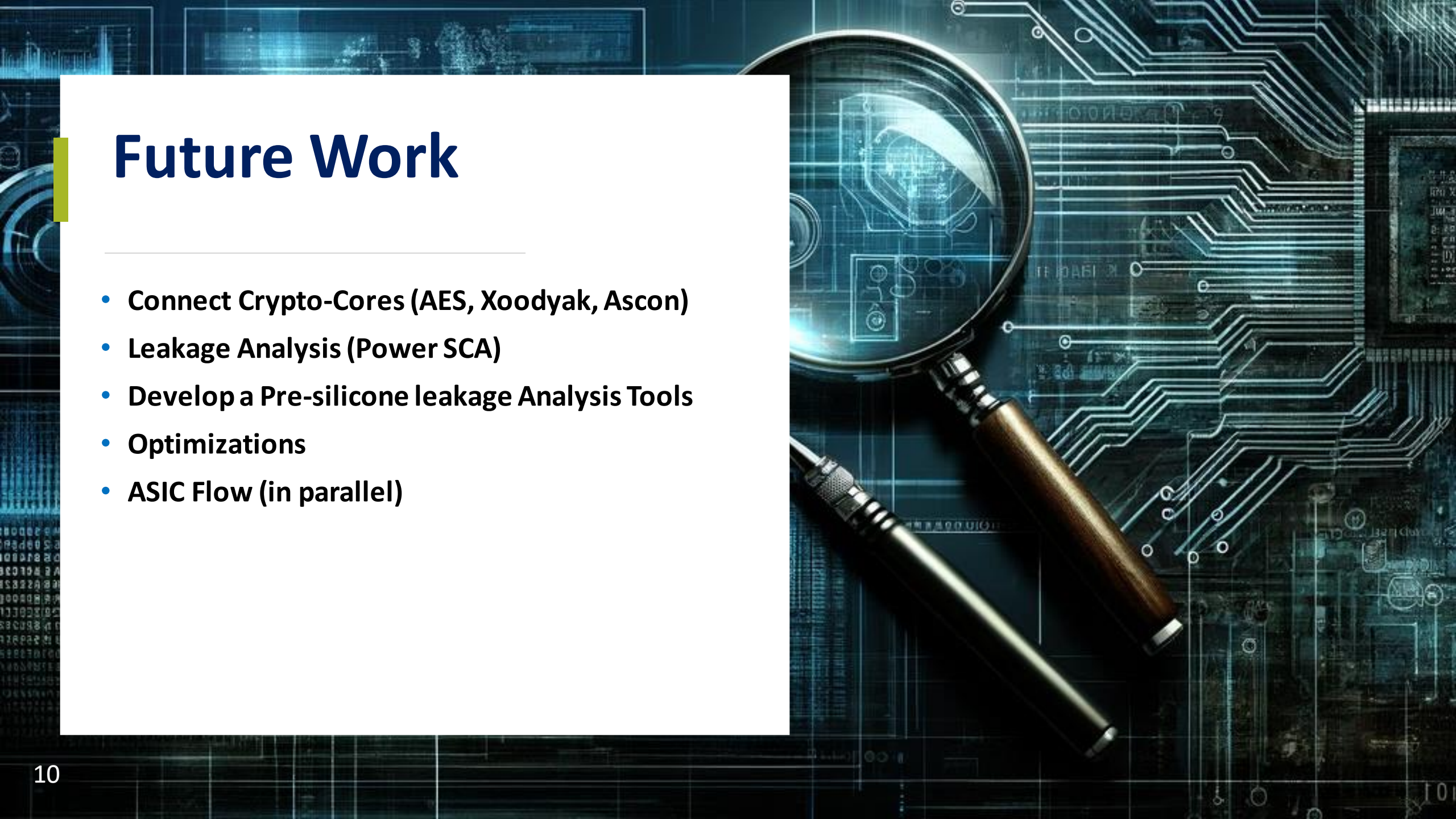
GUI- UART Additions

- Read Status Registers
- Write Control Registers
- Read from specific address
- Write to specific address
- Send program to SW-RV



Future Work

- Connect Crypto-Cores (AES, Xoodyak, Ascon)
- Leakage Analysis (Power SCA)
- Develop a Pre-silicone leakage Analysis Tools
- Optimizations
- ASIC Flow (in parallel)




Thank you for
your attenttion
PROACT