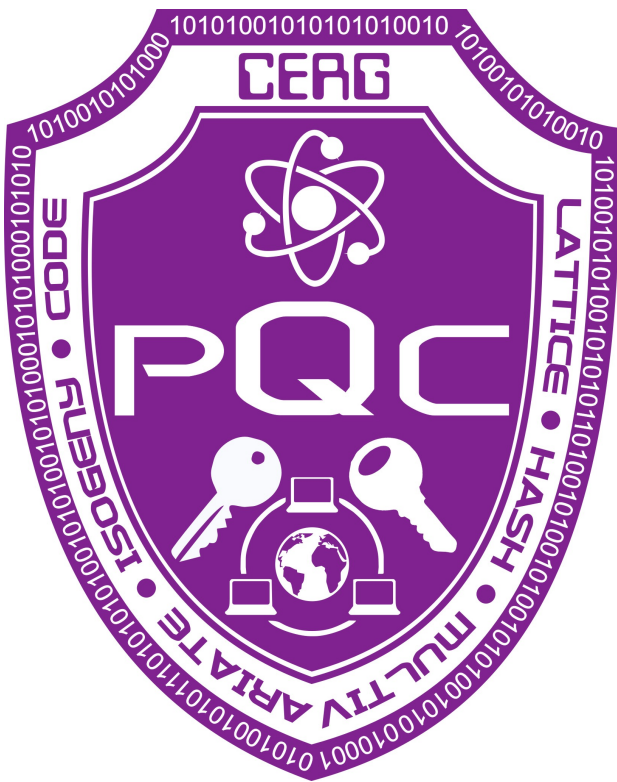


# Implementing and Benchmarking Three Lattice-based Post-Quantum Cryptography Algorithms Using Software/Hardware Codesign

Viet B. Dang,  
Farnoud Farahmand,  
Michał Andrzejczak,  
Kris Gaj



George Mason University

# Co-Authors

---

## GMU PhD Students



**Viet**  
Ba Dang



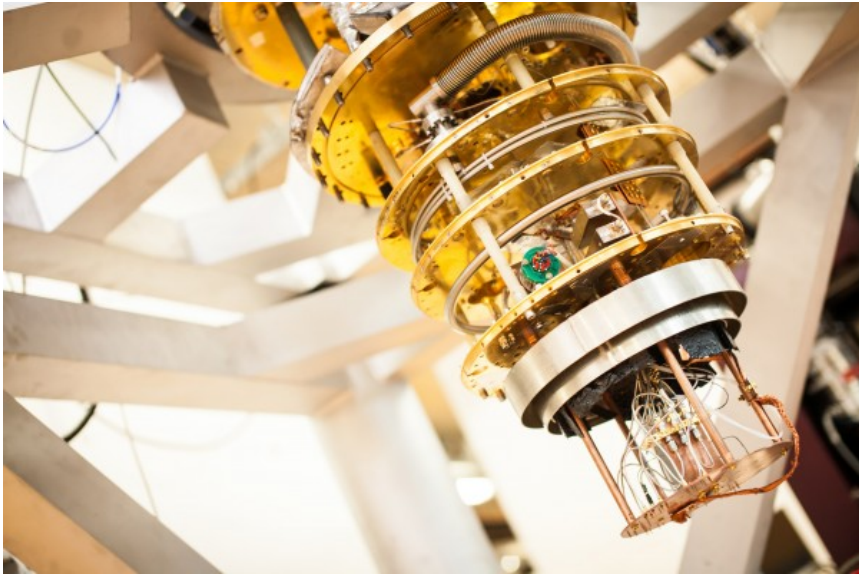
**Farnoud**  
Farahmand

## Visiting Scholar



**Michał**  
Andrzejczak  
Military University  
of Technology in  
Warsaw, Poland

# Quantum Computers



- Substantial investments by: Google, IBM, Intel, Microsoft, Alcatel-Lucent, NTT
- Quantum computers based on superconducting circuits operating in the temperature close to absolute 0 ( $\sim 0.01$  K)



- November 2017: IBM's 50-qubit chip
- January 2018: Intel's 49-qubit chip, "Tangle-Lake"
- March 2018: Google's 72-qubit chip "Bristlecone"
- October 2019: Quantum supremacy experiment made public by Google

# Quantum Computers & Cryptography

---

**1994: Shor's Algorithm**, breaks major public key cryptosystems based on

Factoring:

RSA

Discrete logarithm problem (DLP):

DSA, Diffie-Hellman

Elliptic Curve DLP:

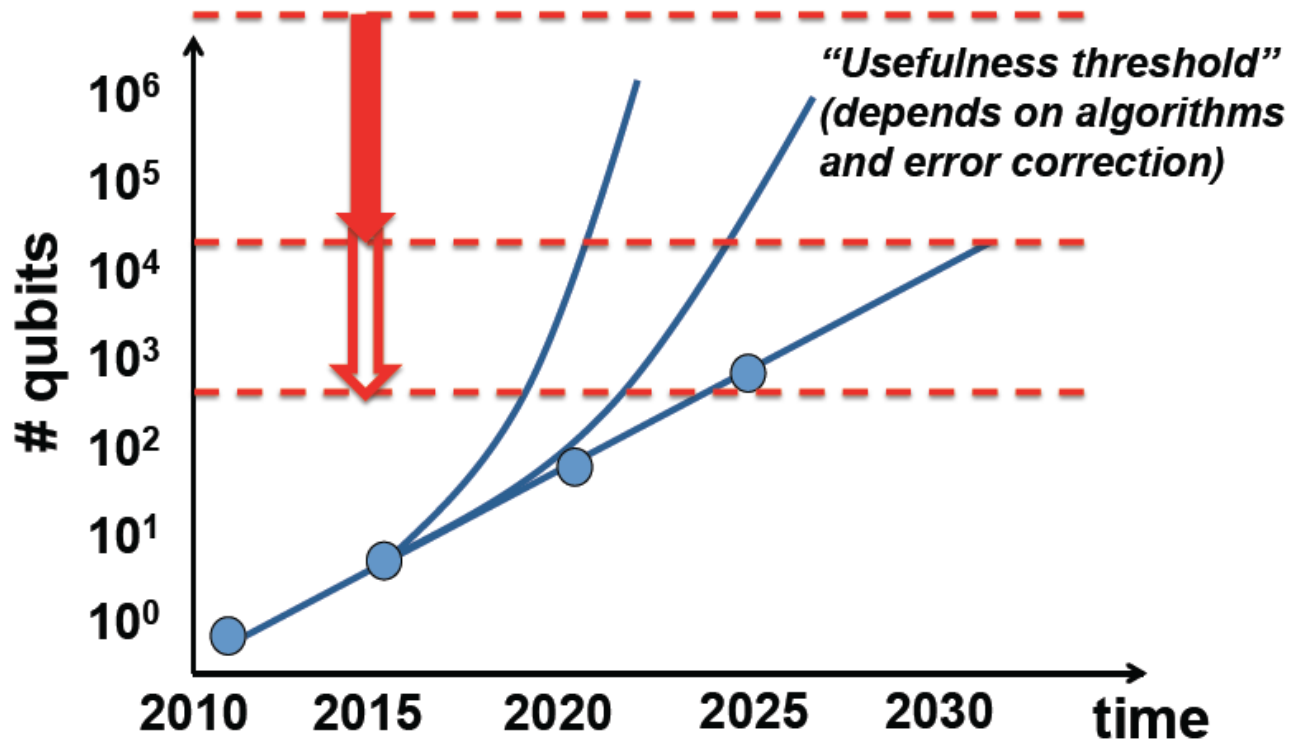
Elliptic Curve Cryptosystems

**independently of the key size**

**assuming**

**a sufficiently powerful and reliable quantum computer available**

# How Real Is the Danger?



*“There is a 1 in 7 chance that some fundamental public-key crypto will be broken by quantum by 2026, and a 1 in 2 chance of the same by 2031.”*

Dr. Michele Mosca

Deputy Director of the Institute for Quantum Computing, University of Waterloo

April 2015

# Post-Quantum Cryptography (PQC)

---

- Public-key cryptographic algorithms for which there are **no known attacks** using quantum computers
- Capable of being implemented using any **traditional methods**, including **software and hardware**
- Running efficiently on **any modern computing platforms**:  
smartphones, tablets, PCs, servers with FPGA accelerators, etc.
- **Based entirely on traditional semiconductor VLSI technology!**

# Cryptographic Contests 2007-Present

51 hash functions → 1 winner

X.2007

X.2012

SHA-3

57 authenticated ciphers  
→ multiple winners

I.2013

II.2019

CAESAR

69 Public-Key Post-Quantum  
Cryptography Schemes

XII.2016

TBD

Post-Quantum

07 08 09 10 11 12 13 14 15 16 17 18 19 20 21 22 Year

# Round 2 Candidates

26 Candidates announced on January 30, 2019

Family	Signature	Encryption/KEM	Overall
Lattice-based	3	9	12
Code-based		7	7
Multivariate	4		4
Symmetric-based	2		2
Isogeny-based		1	1
Total	9	17	26

Round 2 Evaluation until mid-2020

To be followed by Round 3, 12-18 months

# Evaluation Criteria

---

**Security**

**Software Efficiency**

$\mu$ Processors     $\mu$ Controllers

**Hardware Efficiency**

**FPGAs**    ASICs

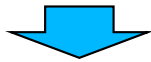
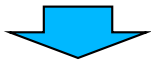

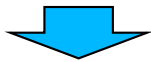
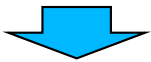
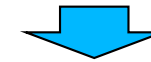


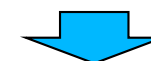
**Flexibility**

**Simplicity**

**Licensing**

# Round 2 Candidates in Hardware

---

	#Round 2 candidates	Implemented in hardware	Percentage
AES	5	5	100%
			
SHA-3	14	14	100%
			
CAESAR	29	28	97%
			
PQC	26	?	?

# Software/Hardware Codesign

---

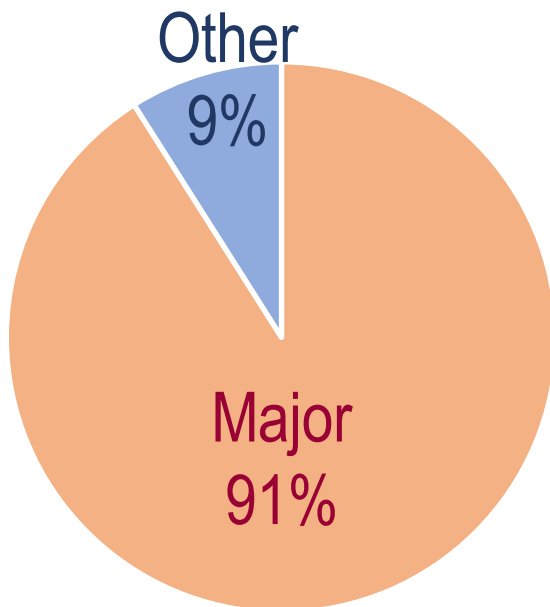
Software

Hardware

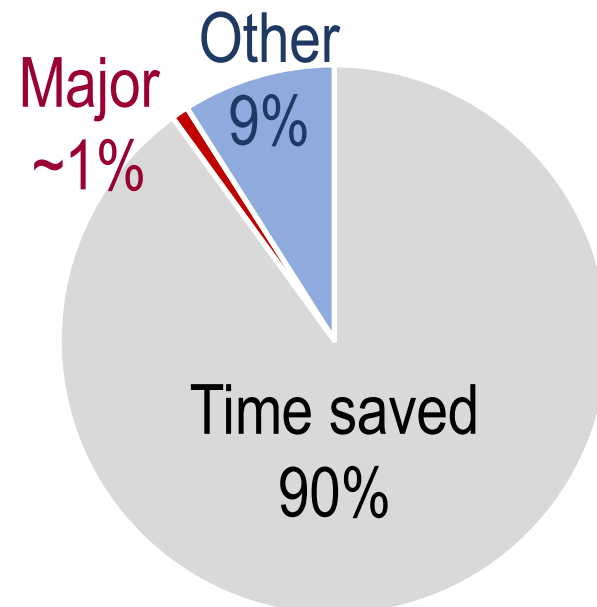
**Most time-critical  
operation**

# SW/HW Codesign: Motivational Example 1

## Software



## Software/Hardware



speed-up  $\geq 100$

91% major operation(s)  
9% other operations



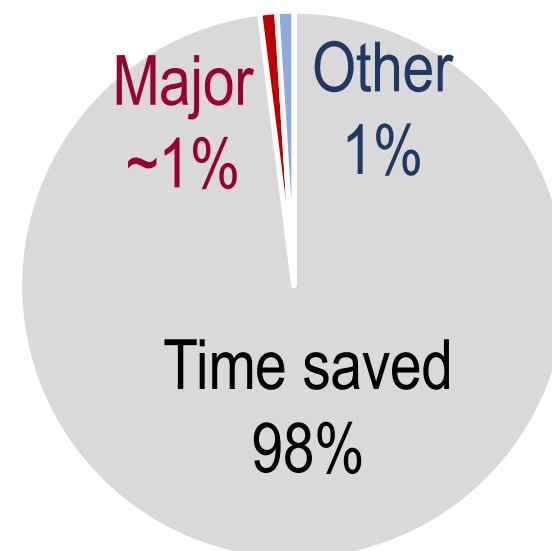
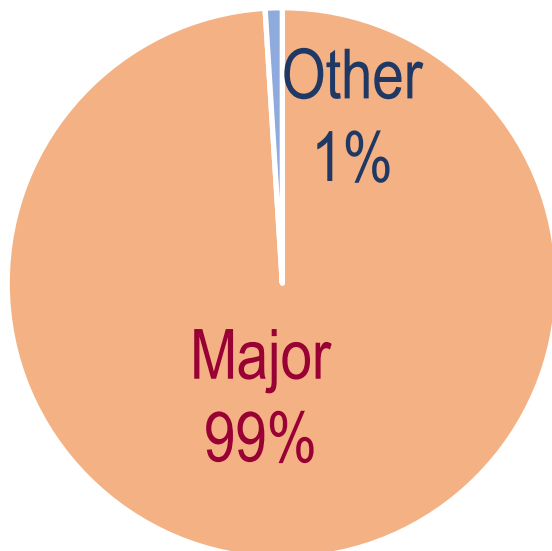
~1% major operation(s) in HW  
9% other operations in SW

**Total Speed-Up  $\geq 10$**

# SW/HW Codesign: Motivational Example 2

Software

Software/Hardware



speed-up  $\geq 100$

99% major operation(s)  
1% other operations

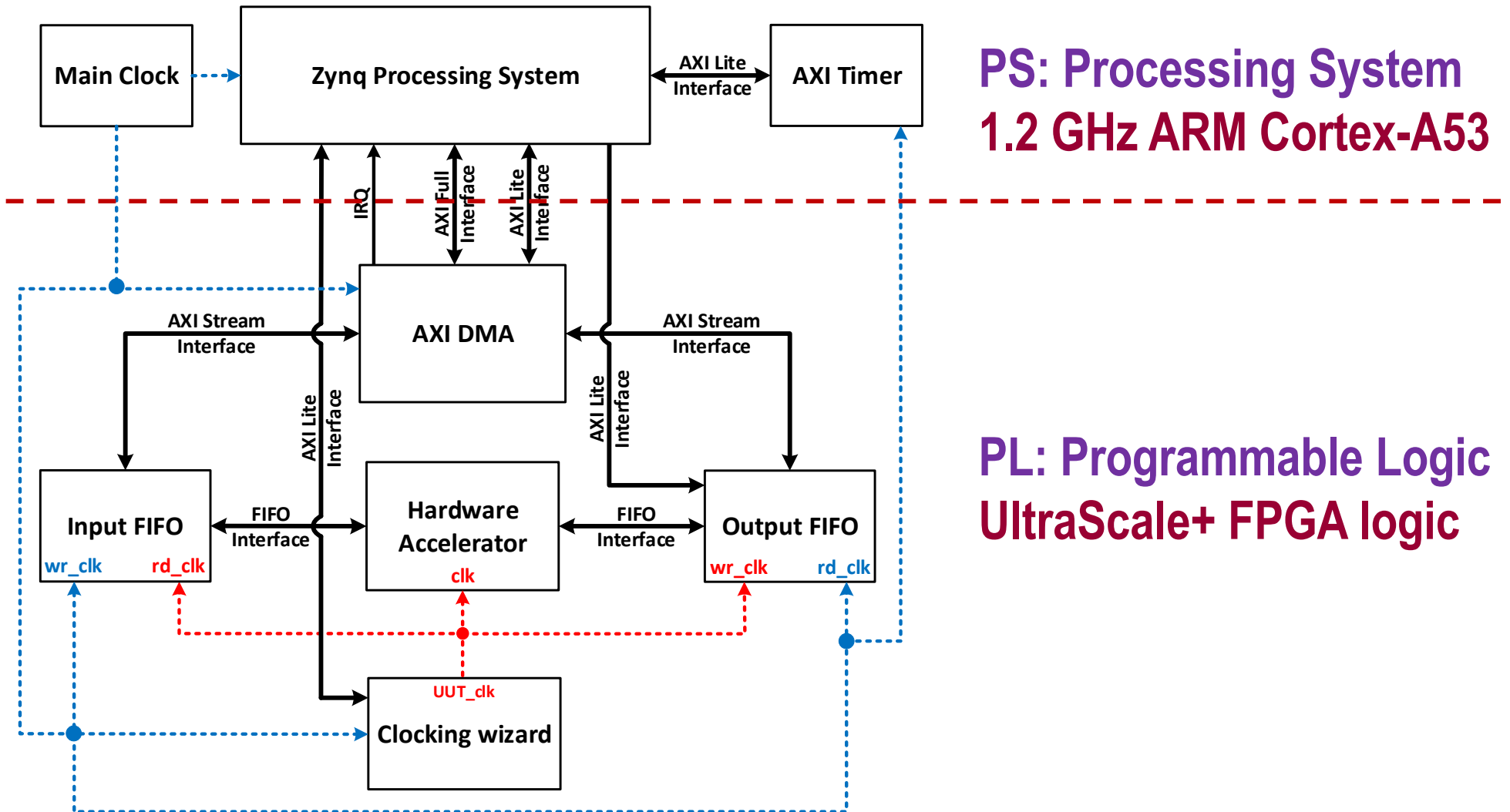


~1% major operation(s) in HW  
1% other operations in SW

Total Speed-Up  $\geq 50$

# Platform & Experimental Setup

## Xilinx Zynq UltraScale+ MPSoC



# SW/HW Codesign: Advantages

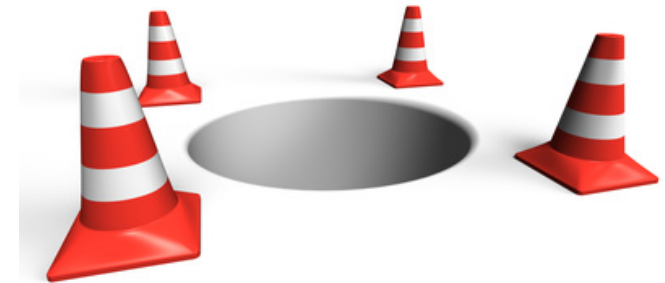
---

- ❖ Focus on a few major operations, known to be easily parallelizable
  - much shorter development time (at least by a factor of 10)
  - guaranteed substantial speed-up
  - high-flexibility to changes in other operations (such as candidate tweaks)
- ❖ Possibility of implementing multiple candidates by the same research group, eliminating the influence of different
  - assumptions
  - design skills
  - tools, etc.

# SW/HW Codesign: Potential Pitfalls

---

- ❖ Performance & ranking may strongly depend on
  - A. features of a particular platform
    - Software/hardware interface
    - Support for cache coherency
    - Differences in max. clock frequency
  - B. selected hardware/software partitioning
  - C. optimization of an underlying software implementation
- ❖ Limited insight on ranking of purely hardware implementations



**First step, not the ultimate solution!**

---



# Our Case Study

# Round 2 Candidates

26 Candidates announced on January 30, 2019

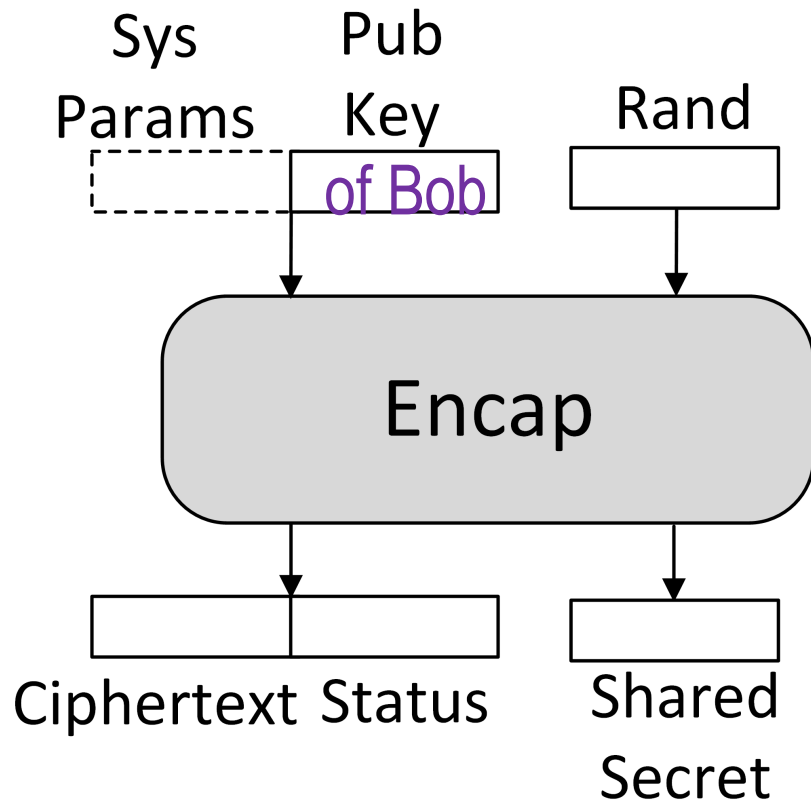
Family	Signature	Encryption/KEM	Overall
Lattice-based	3	9	12
Code-based		7	7
Multivariate	4		4
Symmetric-based	2		2
Isogeny-based		1	1
Total	9	17	26

Round 2 Evaluation until mid-2020

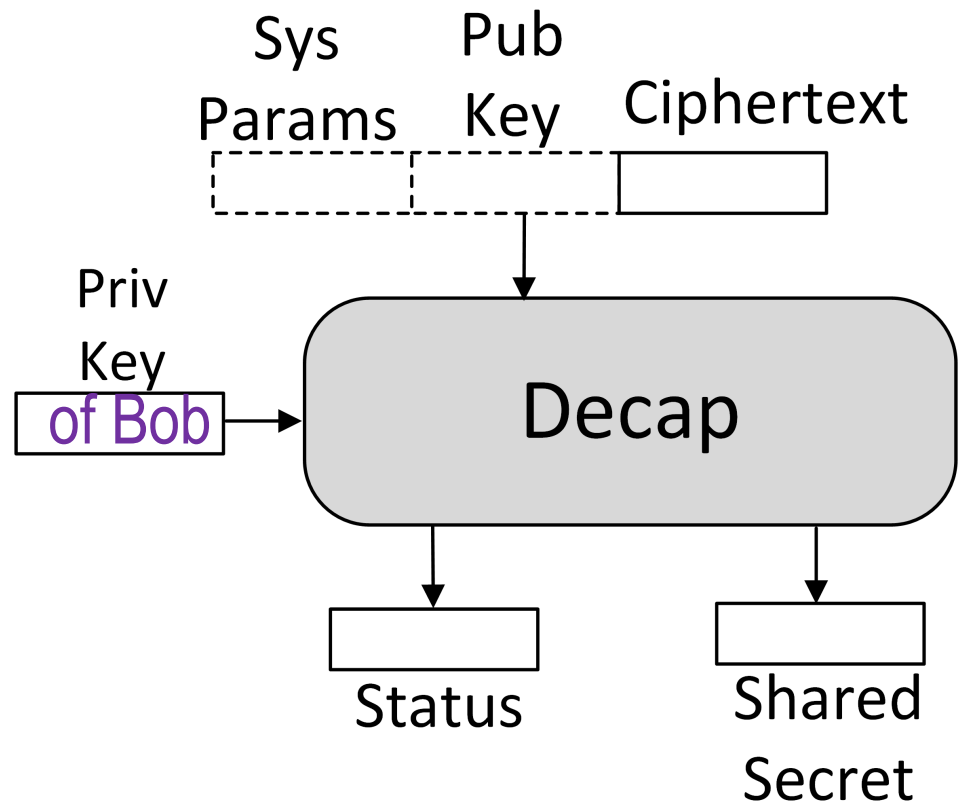
To be followed by Round 3, 12-18 months

# Key Encapsulation Mechanism (KEM)

Alice



Bob



**A way to agree on a common key for secret-key cryptography**

# SW/HW Codesign: Case Study

## 7 Lattice-Based Key Encapsulation Mechanisms representing 5 NIST PQC Round 2 Submissions

LWE (Learning with Error)-based:

FrodoKEM

RLWR (Ring Learning with Rounding)-based:

Round5

Module-LWR-based:

Saber

**3 schemes**

**with designs described in detail  
in the FPT paper**

NTRU-based:

NTRU

- NTRU-HPS
- NTRU-HRSS

NTRU Prime

- Streamlined NTRU Prime
- NTRU LPRime

**4 schemes**

**with results obtained  
after the paper submission.  
Presentation only!**

# Five Security Levels

---

Level	Security Description
I	At least as hard to break as <b>AES-128</b> using exhaustive key search
II	At least as hard to break as <b>SHA-256</b> using collision search
III	At least as hard to break as <b>AES-192</b> using exhaustive key search
IV	At least as hard to break as <b>SHA-384</b> using collision search
V	At least as hard to break as <b>AES-256</b> using exhaustive key search

# SW/HW Partitioning

---

## Top candidates for offloading to hardware

### From profiling:

- ❖ Large percentage of the execution time
- ❖ Small number of function calls

### From manual analysis of the code:

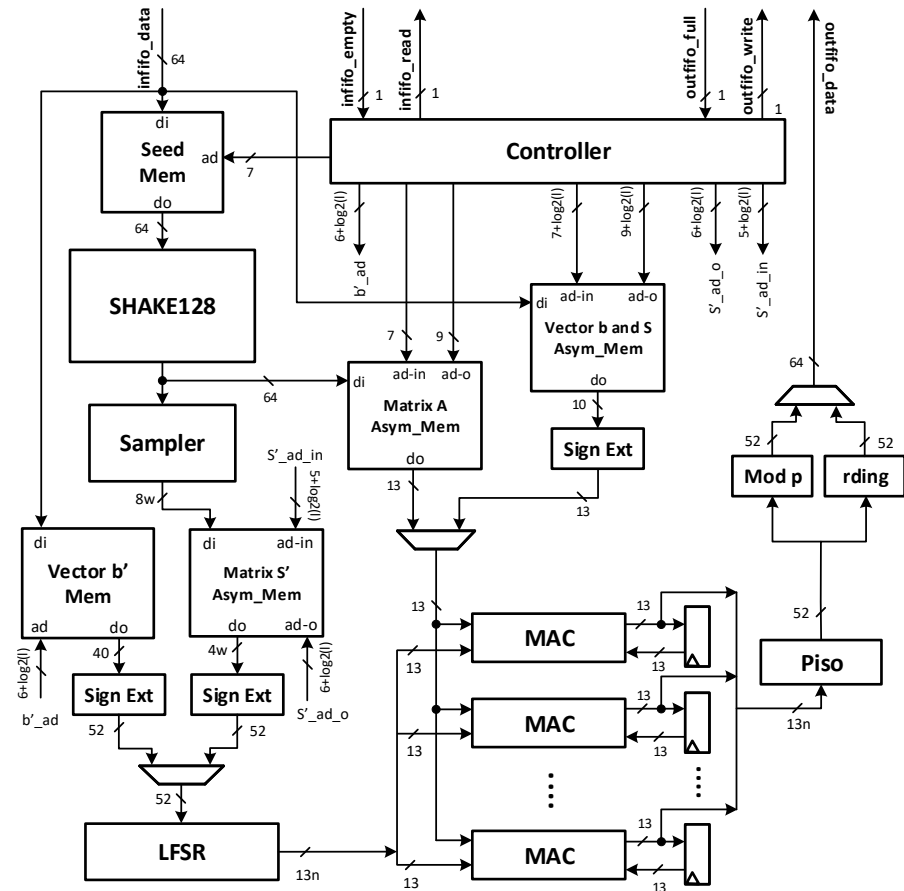
- ❖ Small size of inputs and outputs
- ❖ Potential for combining with neighboring functions

### From knowledge of operations and concurrent computing:

- ❖ High potential for parallelization

# Operations Offloaded to Hardware

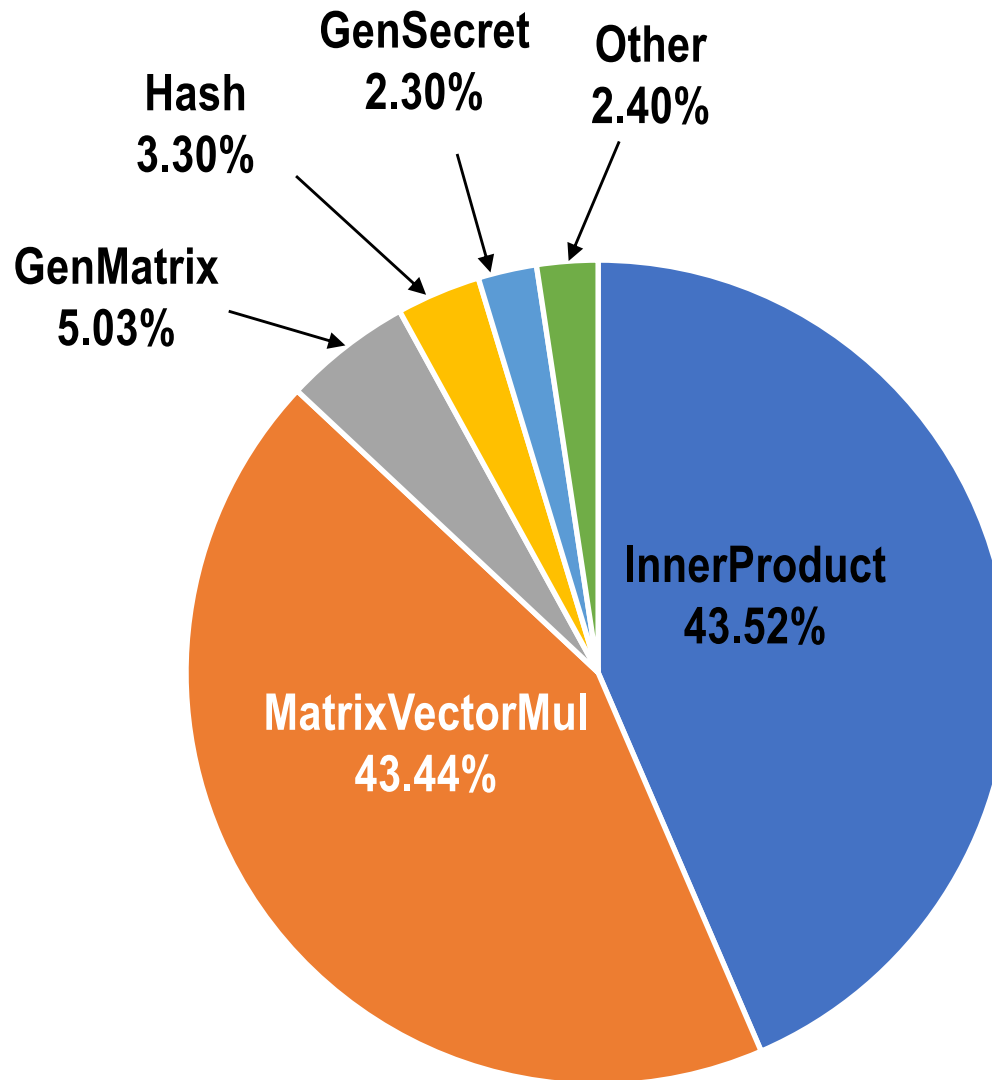
- Major arithmetic operations
  - Polynomial multiplications
  - Matrix-by-vector multiplications
  - Vector-by-vector multiplications
- All hash-based operations
  - (c)SHAKE128, (c)SHAKE256
  - SHA3-256, SHA3-512



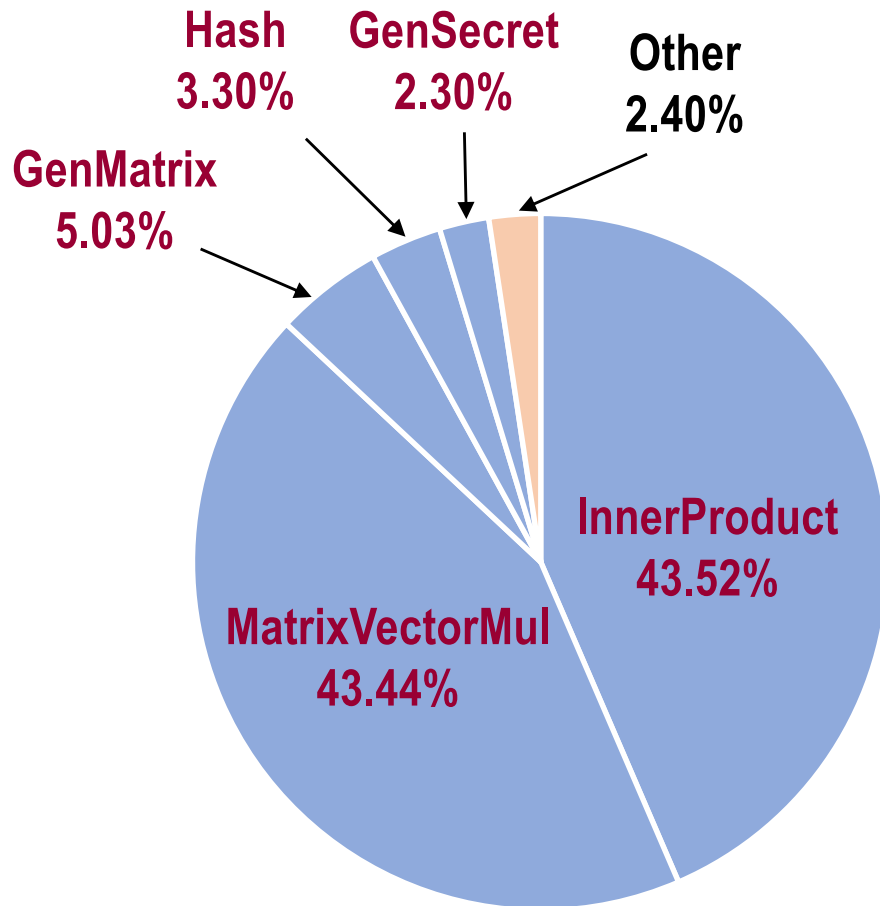
Hardware accelerator  
of Saber

# Example: LightSaber Decapsulation

---

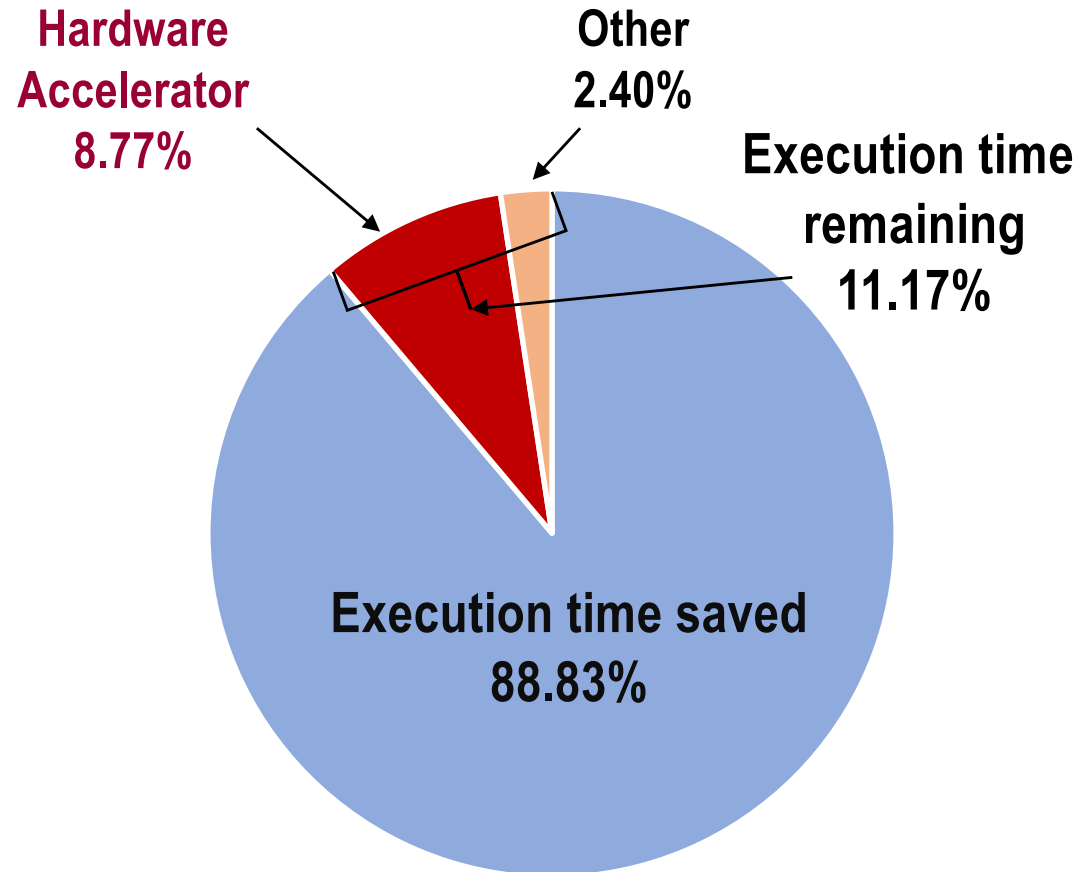


# LightSaber Decapsulation



Execution time of functions  
to be moved to hardware  
97.60%

Execution time of functions  
remaining in software  
2.40%



$$\text{Accelerator Speed-Up} = 97.60 / 8.77 = 11.1$$

$$\text{Total Speed-Up} = 100 / 11.17 = 9.0$$

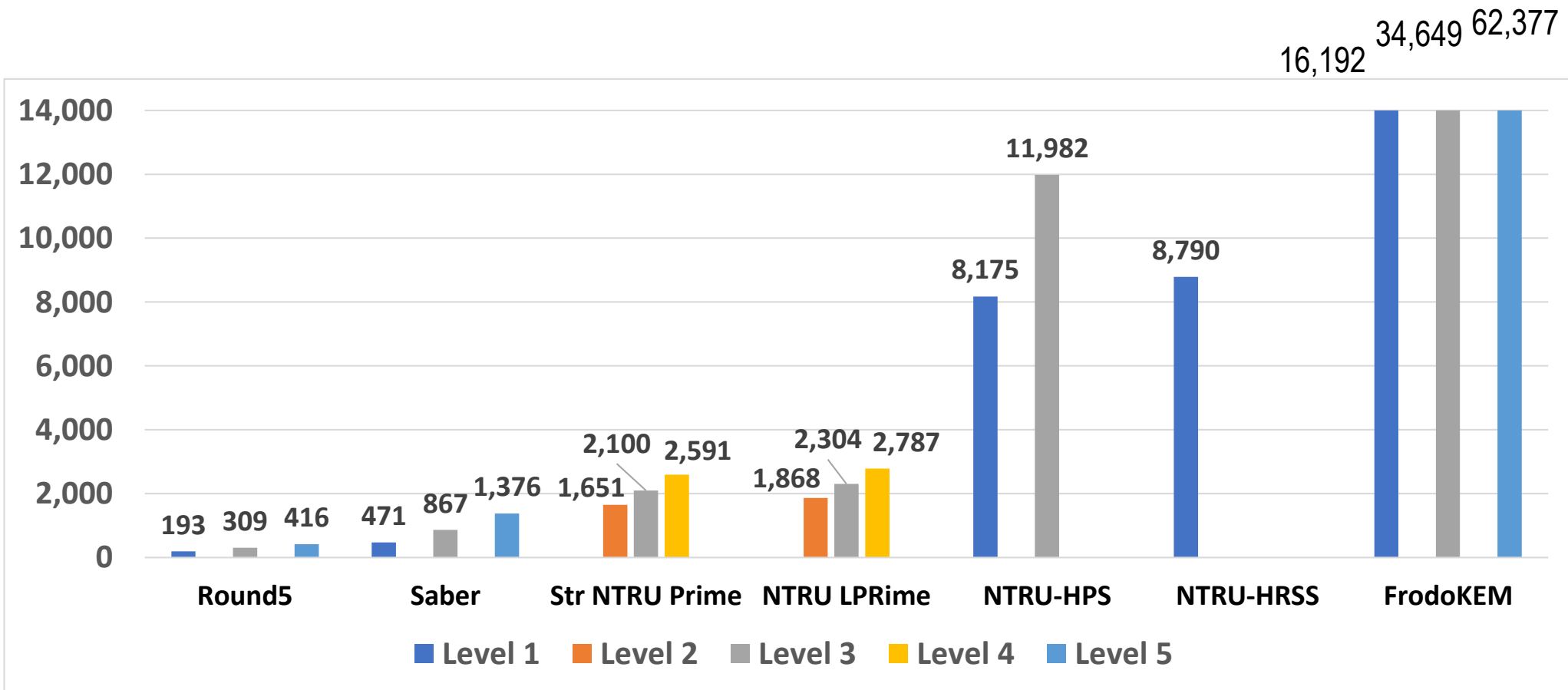
---



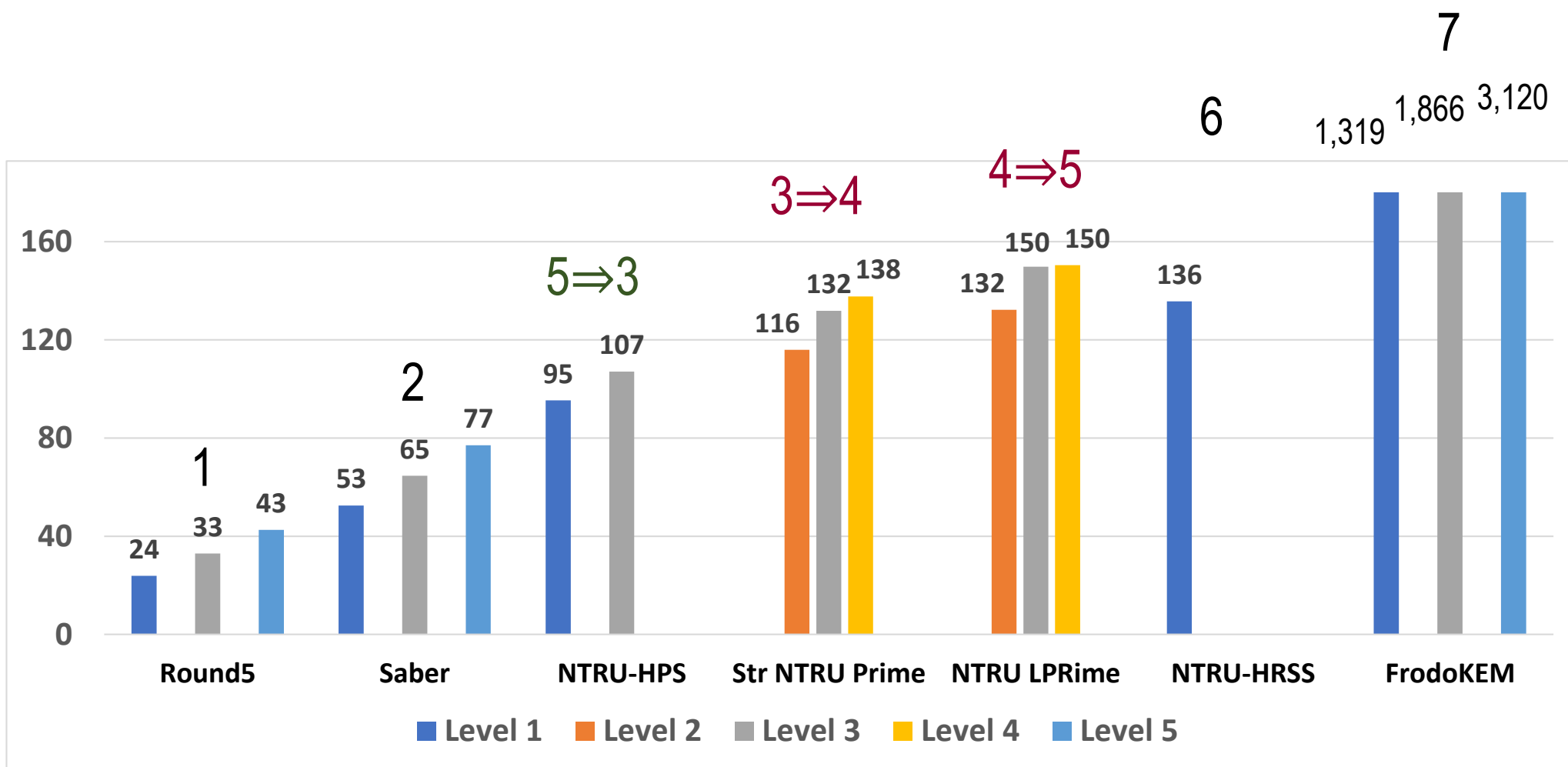
# Results

# Total Execution Time in Software [ $\mu$ s]

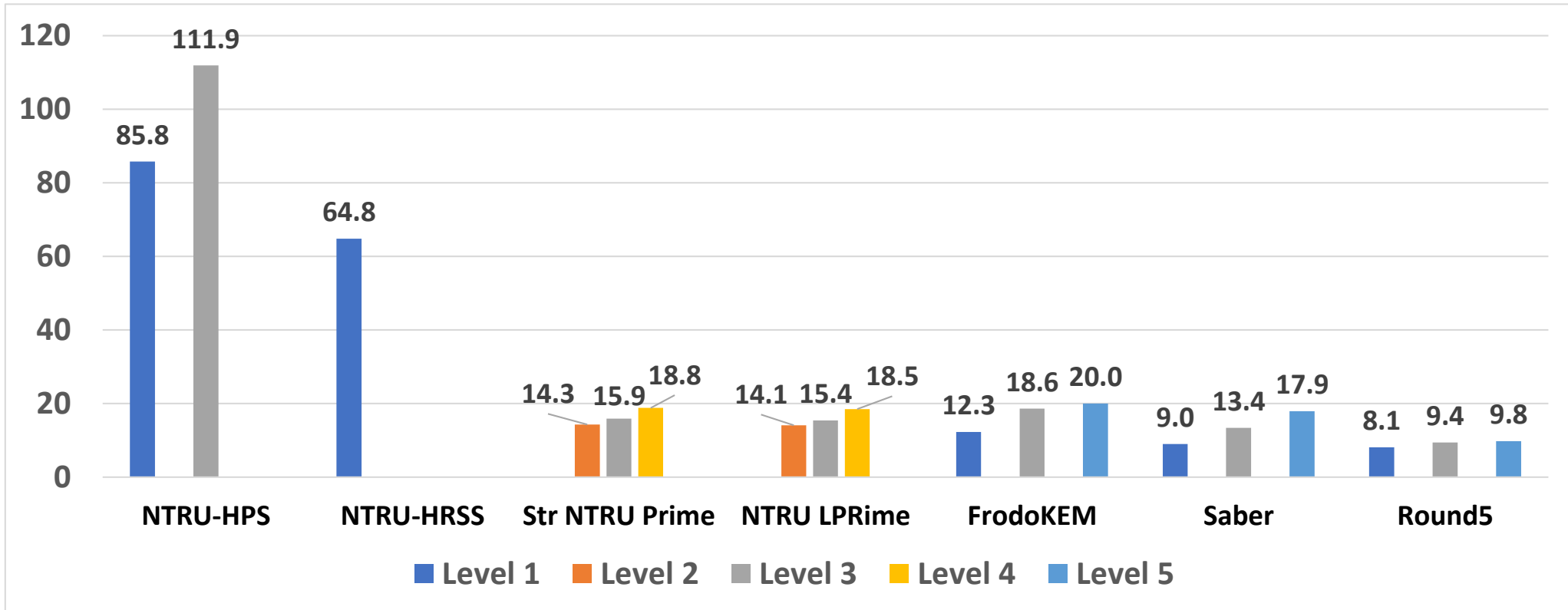
## Decapsulation



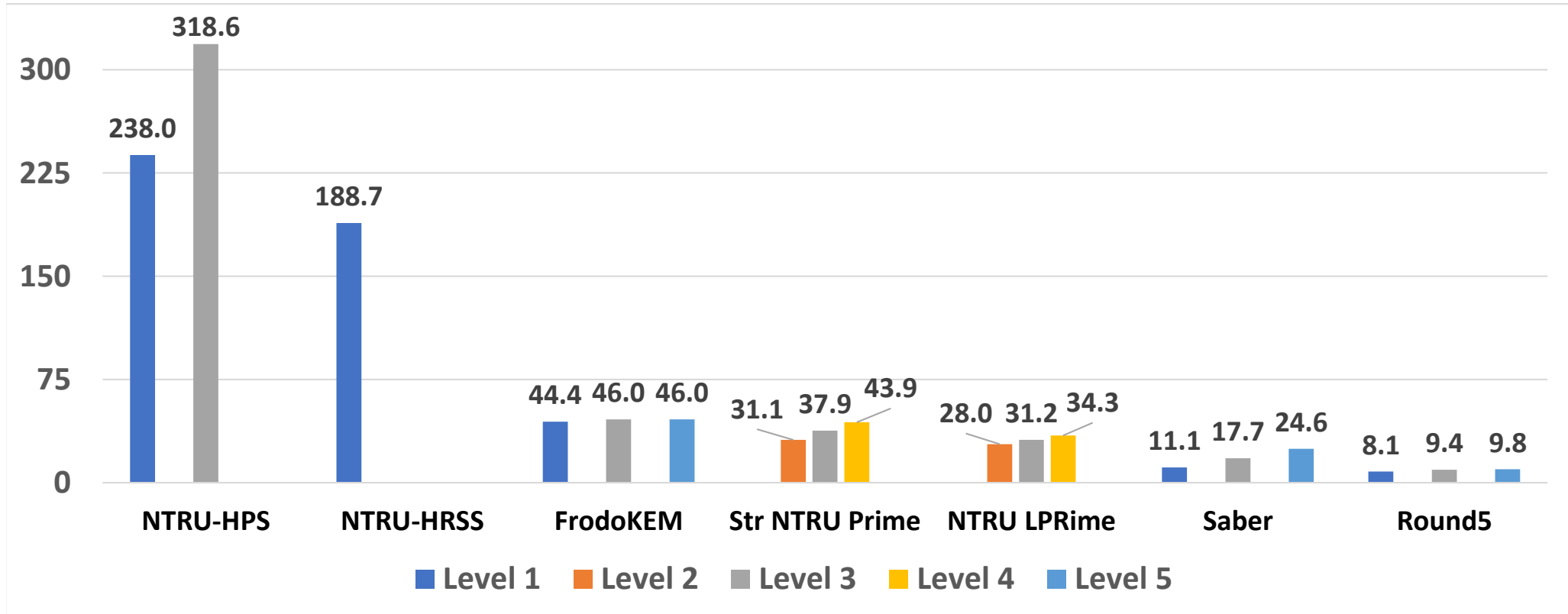
# Total Execution Time in Software/Hardware [ $\mu$ s]: Decapsulation



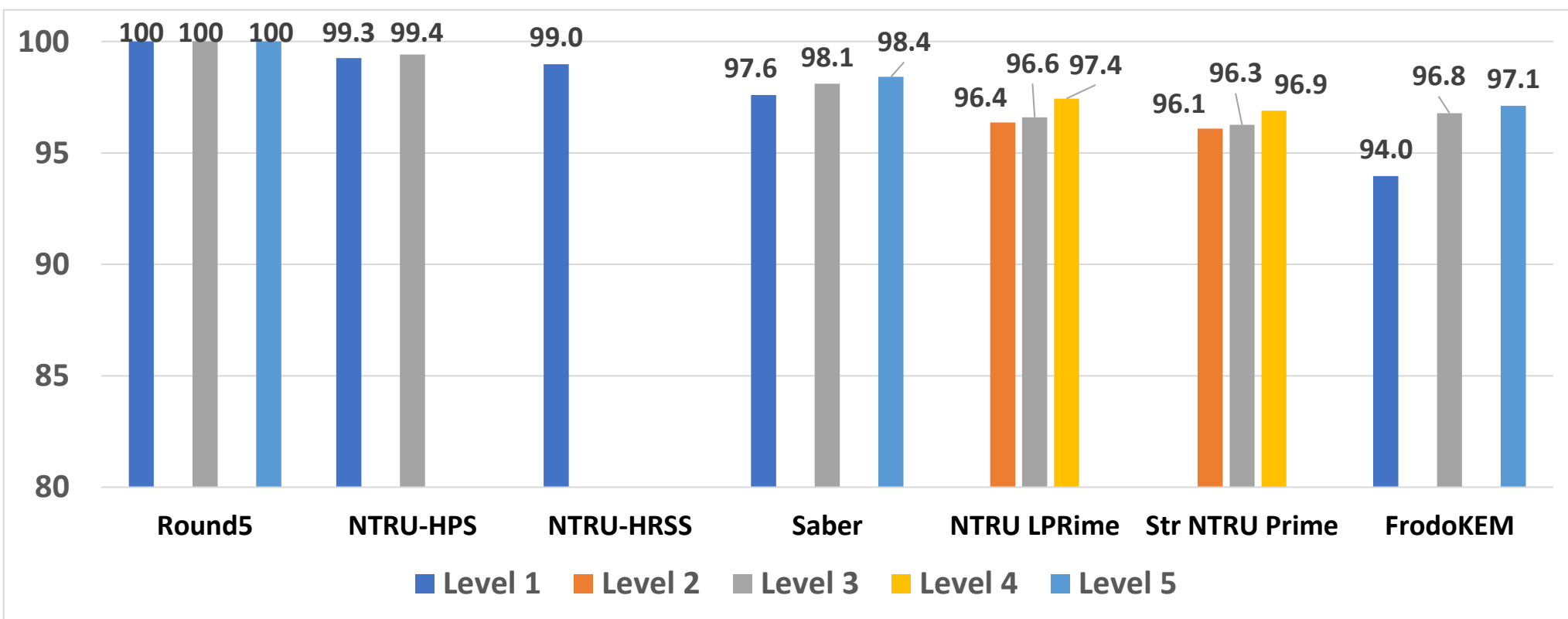
# Total Speed-ups: Decapsulation



# Accelerator Speed-ups: Decapsulation



# SW Part Sped up by HW[%]: Decapsulation

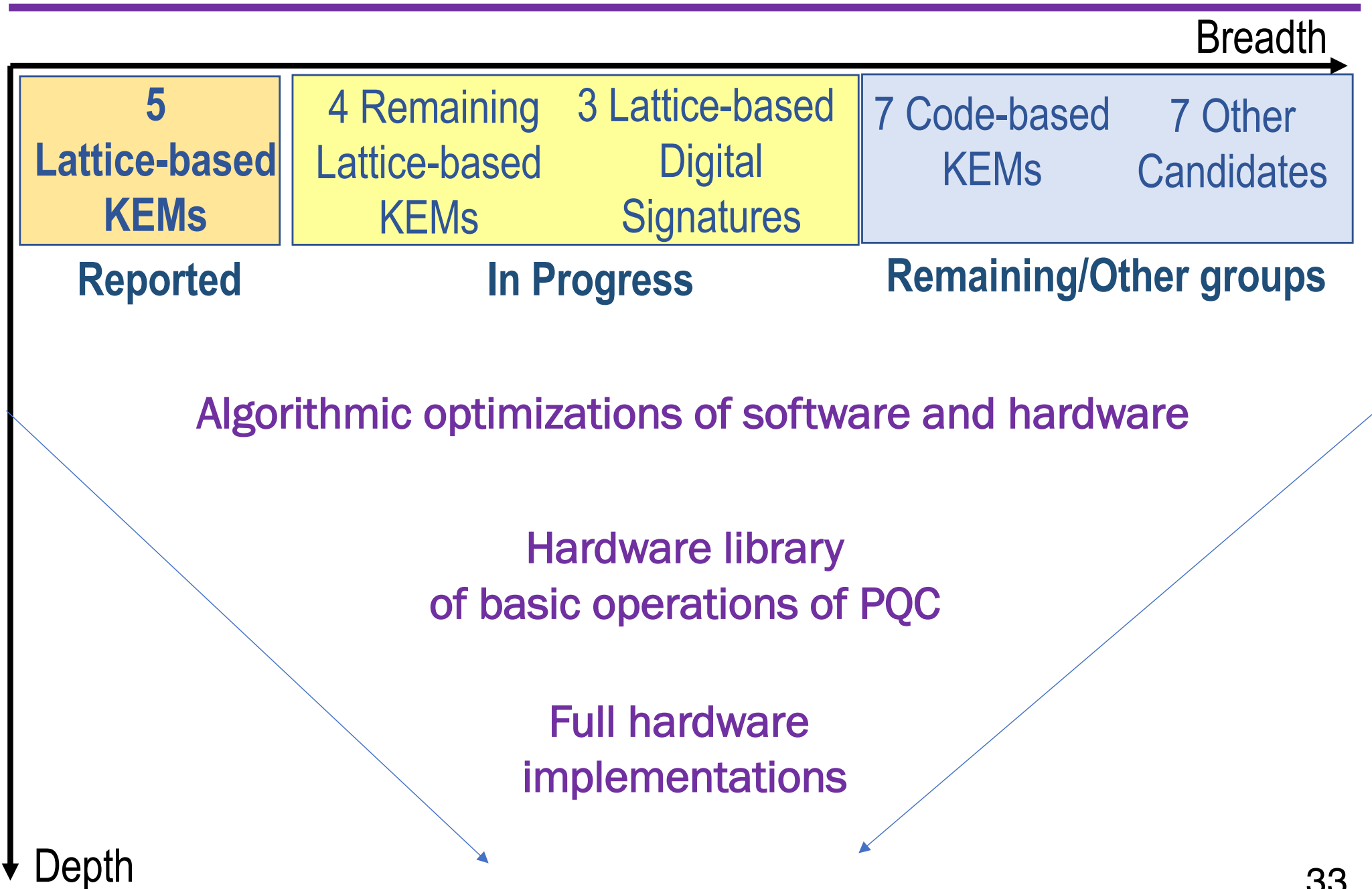


# Conclusions

---

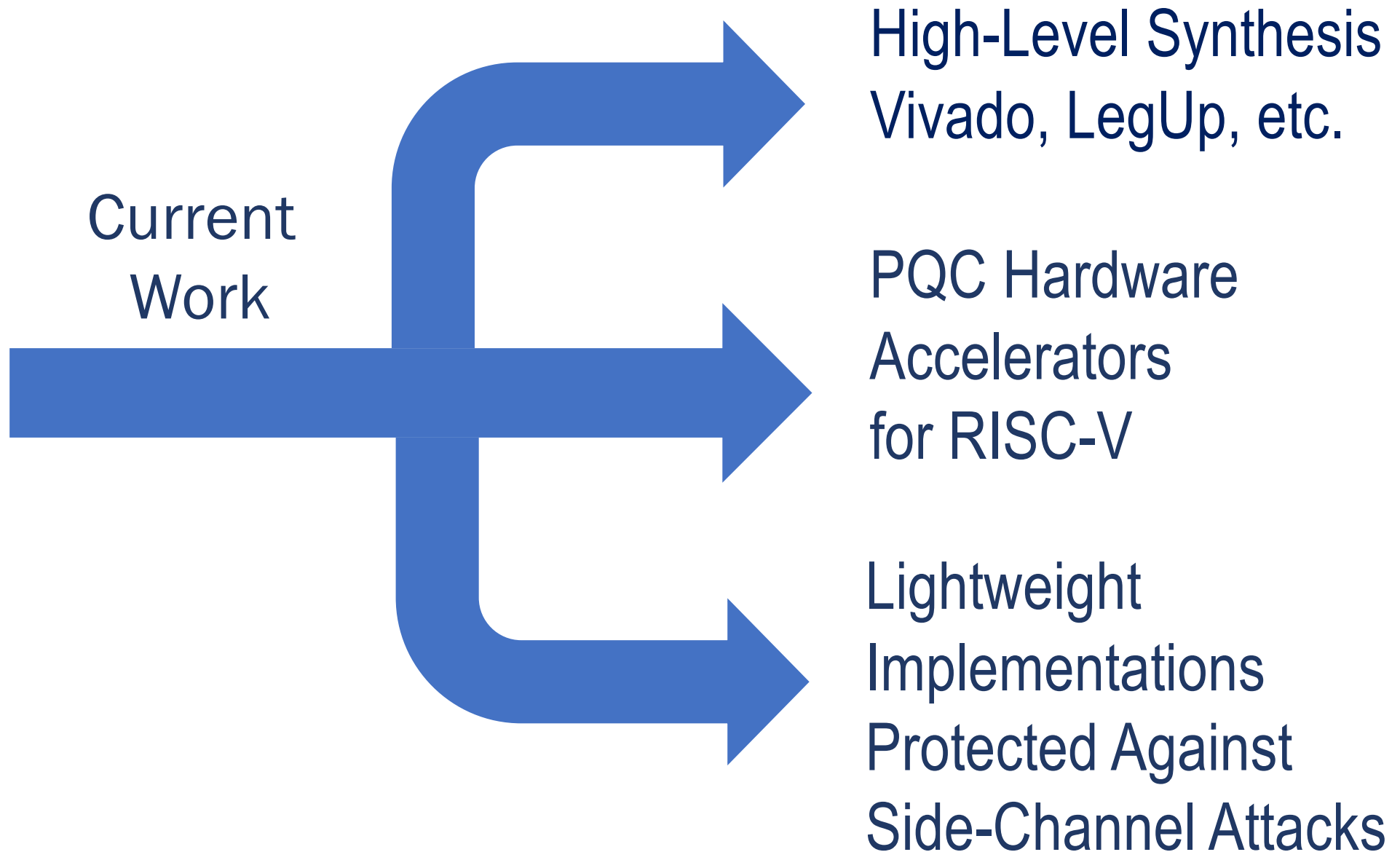
- ❖ For all 7 investigated PQC schemes, major operations offloaded to hardware amounted to at least 94% of the decapsulation time in software
- ❖ Ranking of the investigated candidates affected, but not dramatically changed, by hardware acceleration
- ❖ It might be possible (with participation of several groups) to complete similar designs for all Round 2 candidates within the evaluation period (12-18 months)
- ❖ Additional benefit: Comprehensive library of major operations in hardware

# Future Evaluation



# Future Research Directions

---



# Q&A

## Thank You!

Questions?



Comments?

## Suggestions?

CERG: <http://cryptography.gmu.edu>

ATHENa: <http://cryptography.gmu.edu/athena>