MALIK IMRAN

HARDWARE REALIZATION OF LATTICE-BASED POST-QUANTUM CRYPTOGRAPHY

Email: **malik.imran@taltech.ee**
Centre for Hardware Security
Dpt. of Computer Systems - School of IT
Tallinn University of Technology

CONTENTS

- Background
- Design Space Exploration (DSE)
  - Serial designs
  - Parallel designs
- Implementation Results
- Conclusions

Trends, Tech & Benchmarking

Background

BACKGROUND

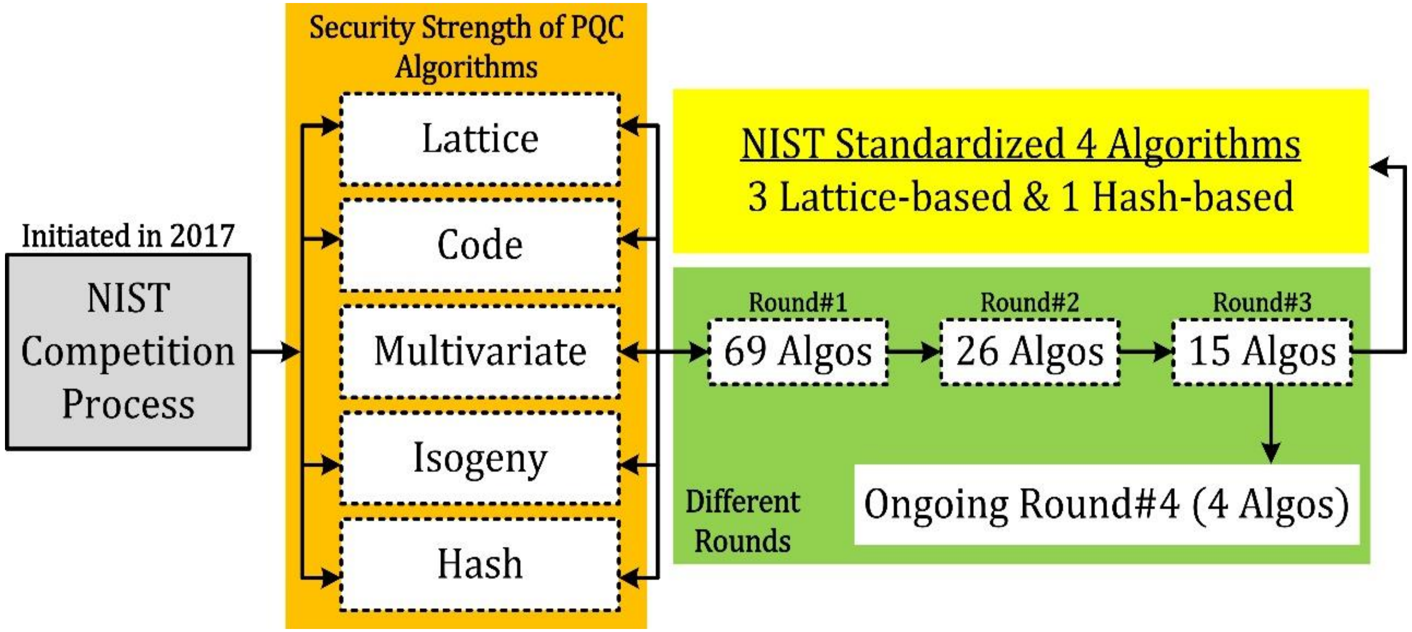
- ❑Cryptography mathematical based technology
  - ❑Symmetric
    - ❑Stream cipher (operates on one bit or byte at a time)
    - ❑Block cipher (block of plaintext is treated to produce ciphertext
  - ❑Asymmetric (public-key)
- ❑Current Standards AES (Symmetric), ECC and RSA (Asymmetric)
  - ❑Security hardness of ECC and RSA solving discrete logarithms and prime factorization problems
- ❑Can be broken using Shor’s algorithm on a quantum computer

Shor, Peter W. (1997), "Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer", SIAM J. Comput., 26 (5): 1484–1509

BACKGROUND

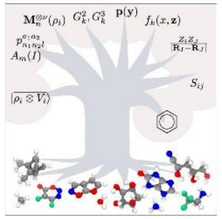
- ❑ **How to make future communications secure?**

  - ❑Post-quantum cryptography (mathematical based technology)



- ❑ **Lattice-based**
  - ❑ CRYSTALS-Kyber
  - ❑ CRYSTALS-Dilithium
  - ❑ FALCON
- ❑ **Hash-based**
  - ❑ SPHINCS+

https://csrc.nist.gov/News/2022/pqc-candidates-to-be-standardized-and-round-4


$\mathbf{M}_n^{\otimes\nu}(\rho_i)$
G_k^2, G_k^3
$\mathbf{p}(\mathbf{y})$
$f_k(x,\mathbf{z})$
$p_{n_1 n_2 l}^{a_1 a_2}$
$A_m(I)$
$\frac{Z_I Z_J}{|\mathbf{R}_I - \mathbf{R}_J|}$
S_{ij}
$|\overline{\rho_i \otimes V_i}\rangle$

Design Space Exploration (DSE)

determines the adaption in various architectural elements with an emphasis on optimizing the design for a specific 65nm ASIC technology

DESIGN SPACE EXPLORATION

State of the art (hardware accelerators)

Existing hardware accelerators on FPGA and ASIC platforms

Designs	Ref.#	Platform	Latency (µs)	Freq (MHz)	Area (LUT/FF or mm²)
SABER	[1]	UltraScale+	21.8/26.5/32.1	250	23.6K/9.8K
	[2]	40nm	2.66/3.64/4.25	400	0.38
	[3]	Artix-7	-/373.1/422.1	125	6.7K/7.3K
	[4]	Artix-7	3.2K/4.1K/3.8K	125	7.4K/7.3K
	[5]	UltraScale+	-/60/65	322	-/-

Is it possible to improve circuit frequency

DESIGN SPACE EXPLORATION

❑Baseline

- ❑ DP_1(1024×64)

❑ Converting the RTL code for ASIC

❑ RTL code: https://github.com/sujoyetc/SABER_HW

❑ Architecture type: “coprocessor”

❑ RTL is written in Verilog HDL for specific to “FPGAs”

❑ Serial designs

- ❑ DP_2(1024×32)
- ❑ DP_4(1024×16)
- ❑ DP_8(512×16)
- ❑ PIP_DP_4(1024×16)
- ❑ PIP_SP_4(256×64)

❑ Exploration of different types, numbers, and sizes of compiled memories in a `smart synthesis' fashion

❑ Logic sharing

❑ Pipelining

❑ Tapeout-ready design for SABER (in 65nm CMOS)

❑ RTL code for optimized SABER architecture[1]

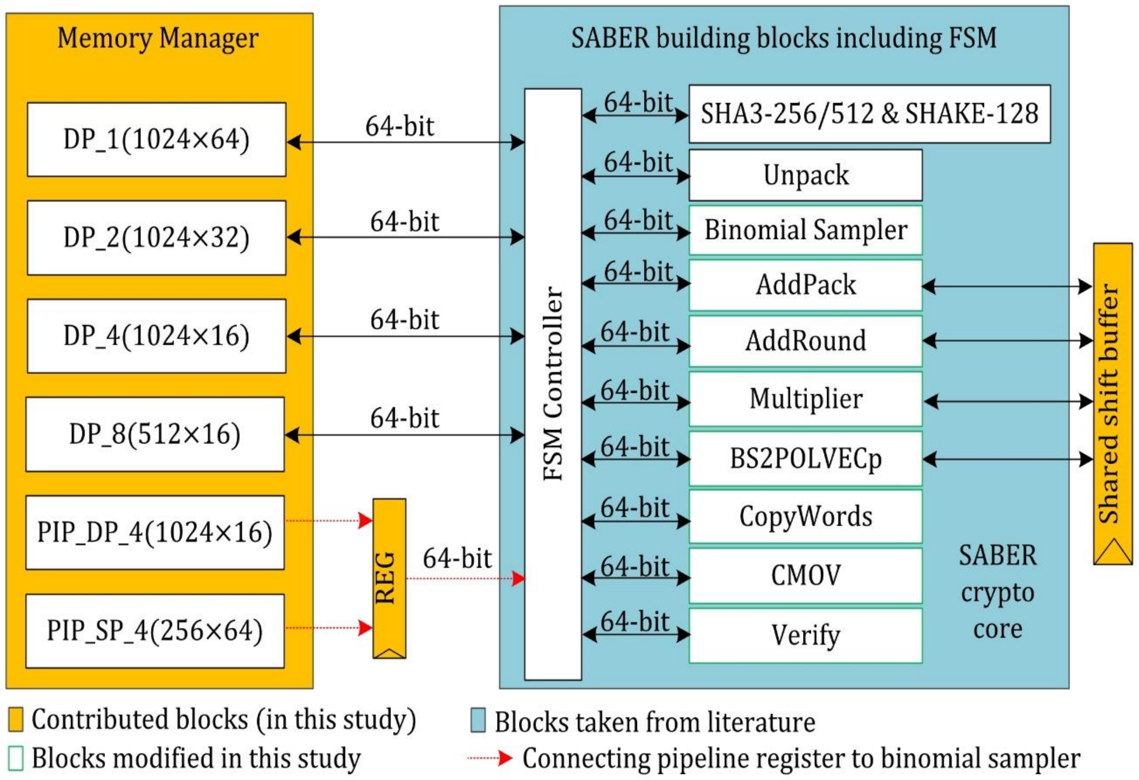
❑Parallel designs

- ❑SS_Parallel_SP_4(256×64)
- ❑DS_Parallel_SP_4(256×64)

❑ Datapath supporting 64 and 256 bit building blocks

[1] Malik Imran and Samuel Pagliarini. 2021. saber-chip. https://github.com/Centre-for-Hardware-Security/saber-chip.

DESIGN SPACE EXPLORATION



Serial SABER designs

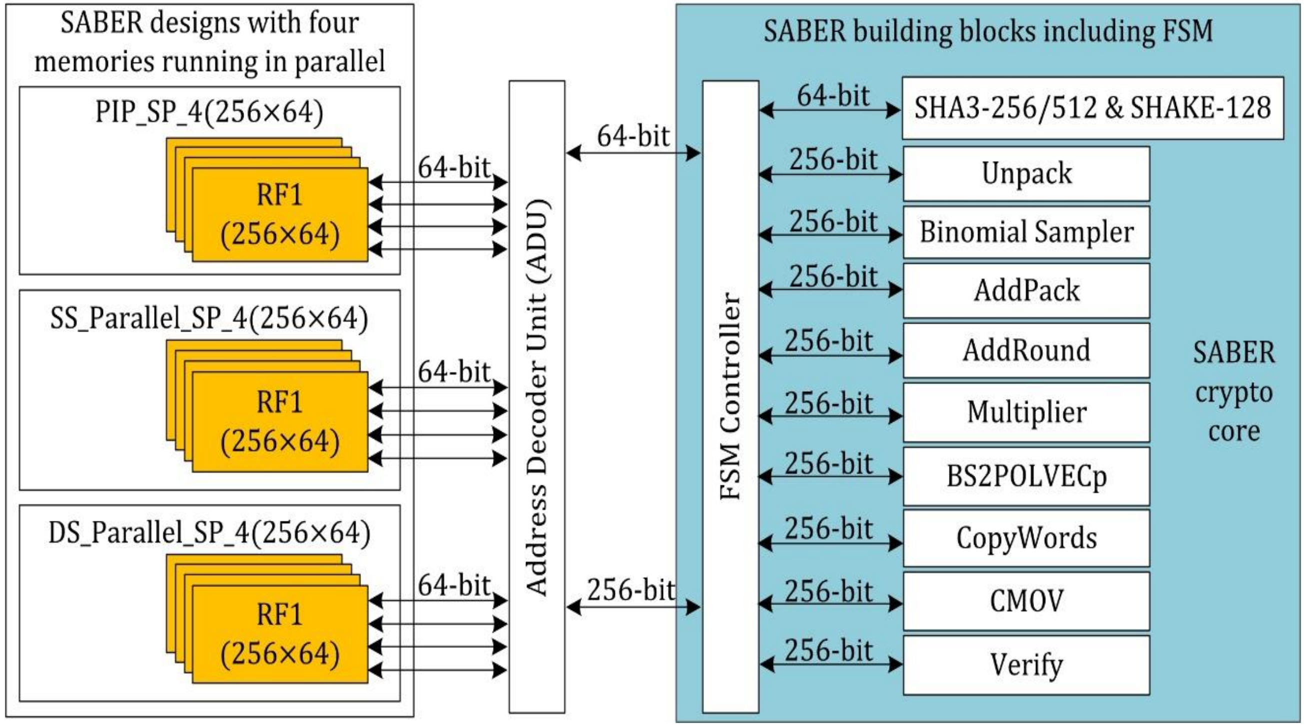
- ❑ **DP:** dual port memory
- ❑ **PIP:** pipelined
- ❑ **SP:** single port memory

- ❑ 64 bit architectures
- ❑ Memory instances operates serially

selected for parallel designs
PIP_SP_4(256×64)

M. Imran, F. Almeida, J. Raik, A. Basso, S. S. Roy, and S. Pagliarini, "Design Space Exploration of SABER in 65nm ASIC," *In Proceedings of the 5th Workshop on Attacks and Solutions in Hardware Security (ASHES '21)*, Republic of Korea, 2021, pp. 85–90. https://doi.org/10.1145/3474376.3487278

DESIGN SPACE EXPLORATION



Parallel SABER designs

- ❑ **PIP:** pipelined
- ❑ **SP:** single port memory
- ❑ **SS:** single sponge
- ❑ **DS:** double sponge

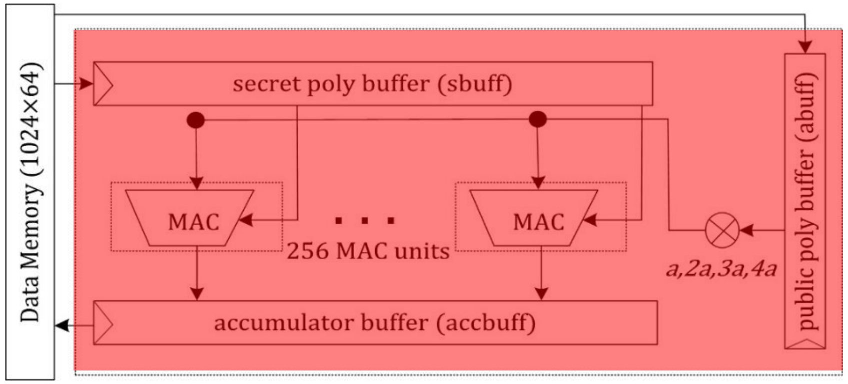
- ❑ 4 memories operates in parallel (one 256 bit word for datapath)
  - ❑ Each deals with 64 bit word

- ❑ 64 bit words for hash operations

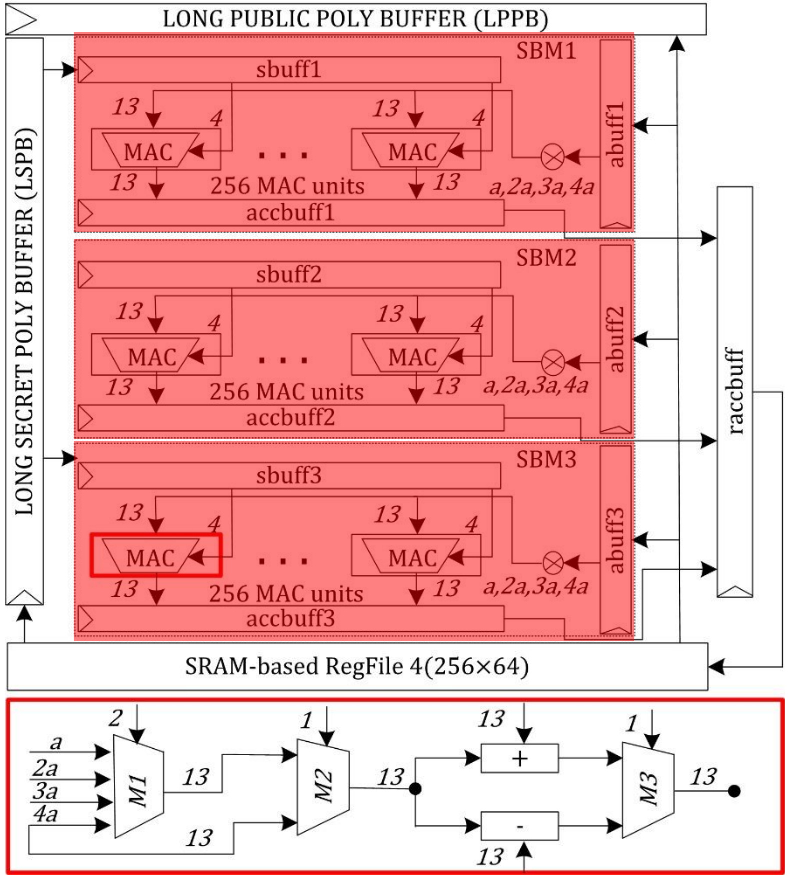
M. Imran, A. Aikata, S. S. Roy, and S. Pagliarini, “High-speed Design of Post Quantum Cryptography with Optimized Hashing and Multiplication,” *IEEE Trans. on Circuits and Systems: Express Briefs.* 10.1109/TCSII.2023.3273821

DESIGN SPACE EXPLORATION

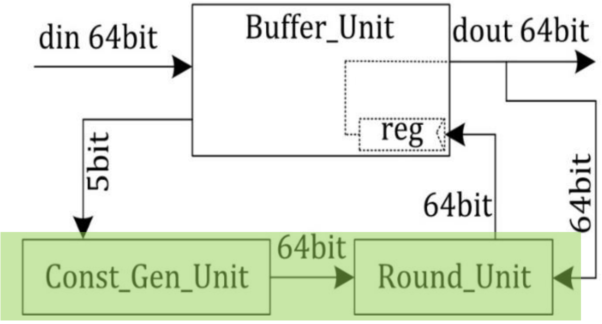
$$\begin{bmatrix} a_{(0,0)} & a_{(0,1)} & \cdots & a_{(0,255)} \\ a_{(1,0)} & a_{(1,1)} & \cdots & a_{(1,255)} \\ a_{(2,0)} & a_{(2,1)} & \cdots & a_{(2,255)} \end{bmatrix} \cdot \begin{bmatrix} s_0 \\ s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} r_0 \\ r_1 \\ r_2 \end{bmatrix}$$


Data Memory (1024×64)
secret poly buffer (sbuff)
MAC
MAC
256 MAC units
a,2a,3a,4a
public poly buffer (abuff)
accumulator buffer (accbuff)

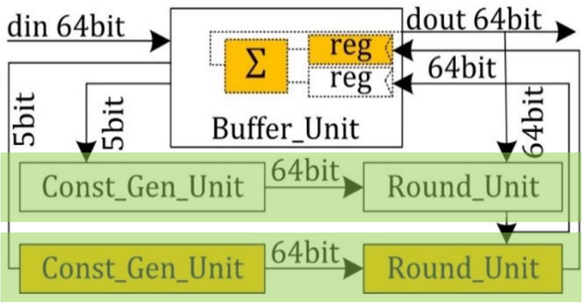
Serial/Iterative schoolbook multiplier



Parallel schoolbook multiplier



Single-sponge Keccak core



Double-sponge Keccak core

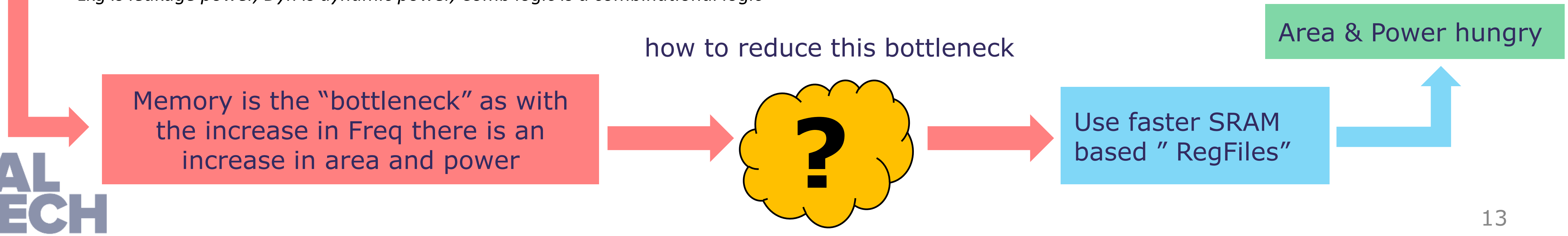
Implementation Results and Comparisons

DESIGN SPACE EXPLORATION

Area and power comparison of serial and parallel SABER designs on 65nm technology

Implemented Designs	Area Results		Timing Results		Power Information (mW)					
	Area (mm^2)	Gates	Clock period (ns)	Freq (MHz)	Crypto core		Combinational logic		Memory	
					Lkg	Dyn	Lkg	Dyn	Lkg	Dyn
Serial SABER designs with 64-bit datapath + single sponge										
DP_1(1024×64)	0.299	43336	2.000	500	0.090	86.844	0.059	16.235 (19%)	0.003	38.001 (44%)
DP_2(1024×32)	0.308	45319	1.718	582	0.091	104.835	0.059	18.499 (18%)	0.004	48.322 (46%)
DP_4(1024×16)	0.340	39981	1.638	610	0.082	135.342	0.051	18.762 (14%)	0.006	81.368 (60%)
DP_8(512×16)	0.478	45979	1.624	615	0.099	220.410	0.062	21.691 (10%)	0.010	157.490 (71%)
PIP_DP_4(1024×16)	0.365	46217	1.508	663	0.097	233.361	0.063	20.890 (10%)	0.006	168.476 (72%)
PIP_SP_4(256×64)	0.314	64230	0.998	1002	0.111	142.413	0.074	32.925 (23%)	0.006	39.060 (27%)
Parallel SABER designs with 256-bit datapath + single/double sponge										
SS_Parallel_SP_4(256×64)	0.944	199288	0.998	1002	0.412	646.880	0.241	106.457 (17%)	0.006	45.376 (7%)
DS_Parallel_SP_4(256×64)	1.026	237761	1.068	936	0.461	860.504	0.289	354.028 (41%)	0.006	43.020 (5%)

Lkg is leakage power, Dyn is dynamic power, Comb logic is a combinational logic



Memory is the “bottleneck” as with the increase in Freq there is an increase in area and power

how to reduce this bottleneck

Use faster SRAM based ” RegFiles”

Area & Power hungry

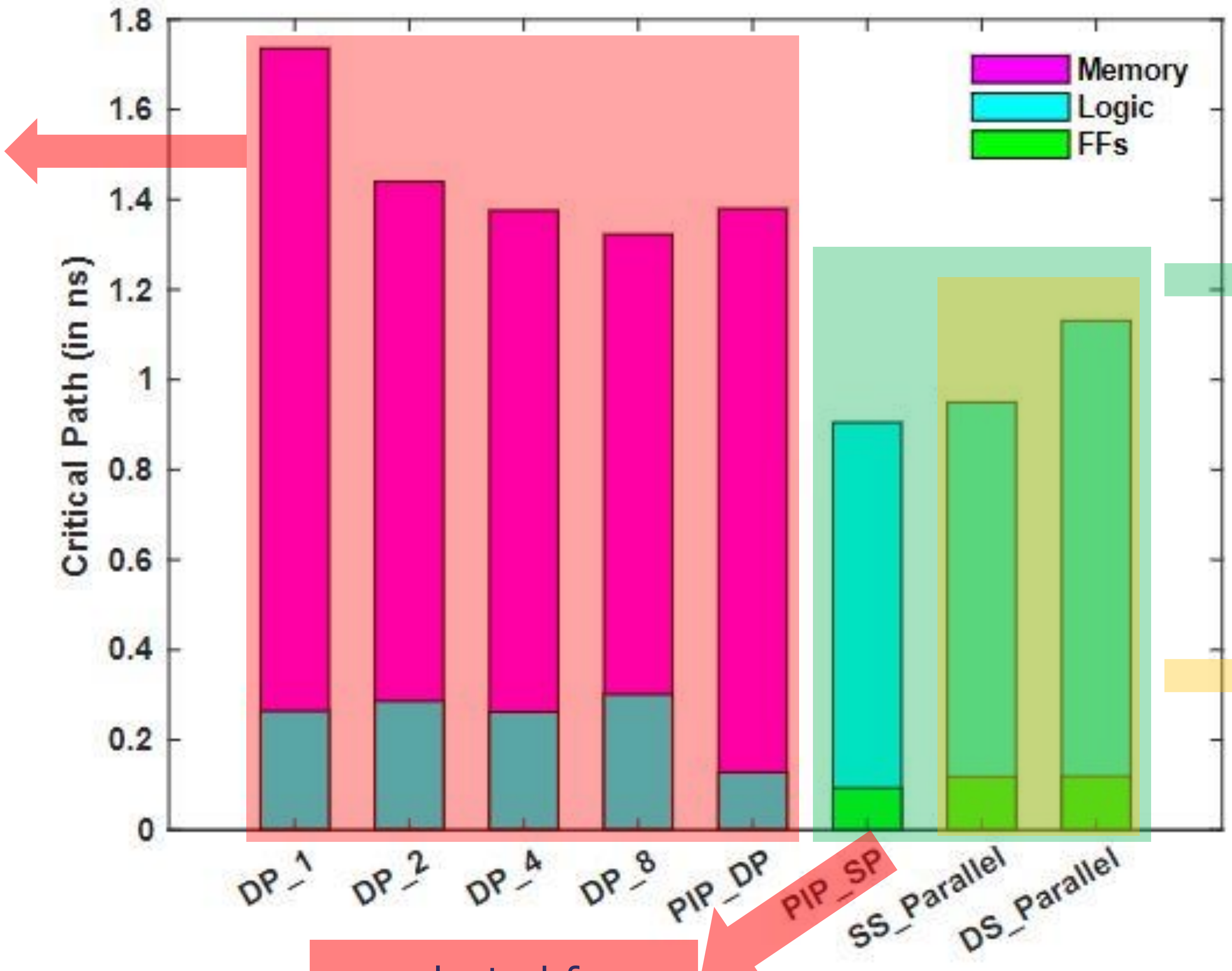
DESIGN SPACE EXPLORATION

Total clock cycles and latency for CCA-secure KEM SABER on a 65nm technology

Designs	Total clock cycles			Latency (µs)		
	KEYGEN	ENCAPS	DECAPS	KEYGEN	ENCAPS	DECAPS
DP_1	5644	6990	8664	11.2	13.9	17.3
DP_2	5644	6990	8664	9.6	12.0	14.8
DP_4	5644	6990	8664	9.2	11.4	14.2
DP_8	5644	6990	8664	9.1	11.3	14.0
PIP_DP	5741	7087	8761	8.6	10.6	13.1
PIP_SP	7154	7136	9359	7.1	7.1	9.3
SS_Parallel	4166	4917	5249	4.1	4.9	5.2
DS_Parallel	3836	4554	4908	4.0	4.8	5.2

Parallel designs are more efficient in clock cycles and computation time (latency)

DESIGN SPACE EXPLORATION



Memory is the "bottleneck"

Smaller memories are more efficient

Evaluation of wider datapath designs on 28nm technology

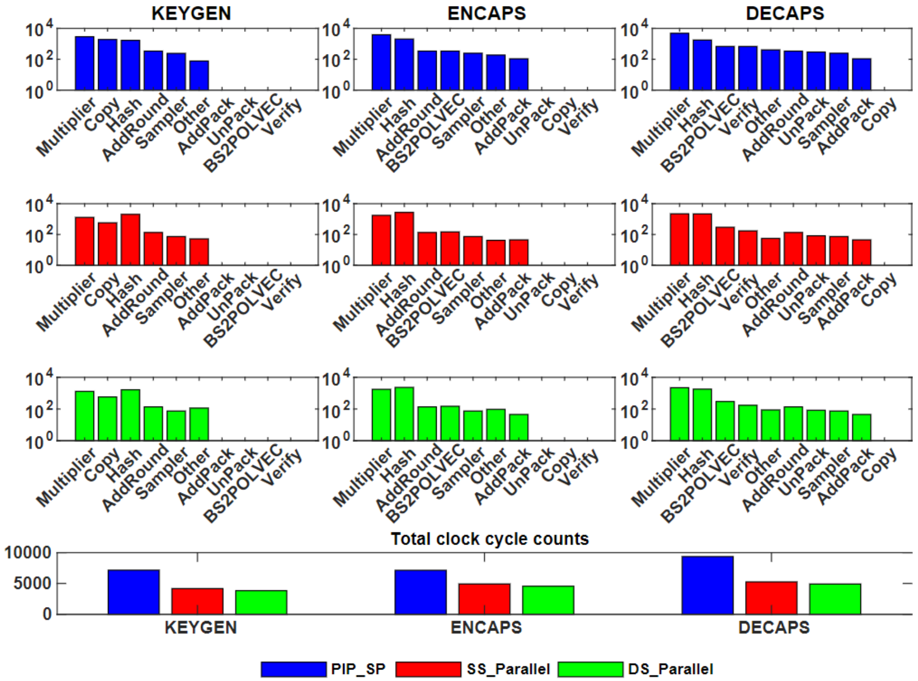
selected for "tapeout"

DESIGN SPACE EXPLORATION

Results of parallel designs on 28nm technology

Implementation Details	SS_Parallel	DS_Parallel
Maximum Freq (*MHz*)	2500	2500
Lat (KG/ENC/DEC) (*μs*)	1.66/1.96/2.09	1.53/1.82/1.96
Utilized Area (*mm*^{2})	0.251	0.255
Power (Lkg/Dyn) (*mW*)	10.96/556.25	11.49/597.05
Energy (μJ)	0.923/1.090/1.162	0.913/1.086/1.170

Lower computation time with "area" and "power" overhead



Clock cycle utilizations

DESIGN SPACE EXPLORATION

Comparison to state of the art hardware accelerators

Designs	Ref.#	Platform	Latency (μs)	Freq (MHz)	Area (LUT/FF or mm^2)
SABER	[1]	UltraScale+	21.8/26.5/32.1	250	23.6K/9.8K
	[2]	40nm	2.66/3.64/4.25	400	0.38
	[3]	Artix-7	-/373.1/422.1	125	6.7K/7.3K
	[4]	Artix-7	3.2K/4.1K/3.8K	125	7.4K/7.3K
	[5]	UltraScale+	-/60/65	322	-/-
This Work	PIP_SP	65nm	7.1/7.1/9.3	1000	0.314
	SS_Parallel	65nm	4.1/4.9/5.2	1002	0.944
	DS_Parallel	65nm	4.0/4.8/5.2	936	1.026
	SS_Parallel	40nm	2.4/2.9/3.0	1694	0.846
	DS_Parallel	40nm	3.4/4.1/4.4	1095	0.767
	SS_Parallel	28nm	1.6/1.9/2.0	2500	0.251
	DS_Parallel	28nm	1.5/1.8/1.9	2500	0.255

CONCLUSIONS

- ❑ Parallel use of several smaller memories
  - ❑ Beneficial to reduce frequent read/write access from the data memory
- ❑ Large data widths more beneficial to reduce clock cycles
- ❑ Efficient hash computations
  - ❑ Allow to optimize the circuit frequency and also help to minimize the cycle counts

The realized approaches (in this study) are practical to other lattice-based PQC algorithms to improve circuit performance for high-speed cryptographic applications

REFERENCES

[1] S. S. Roy and A. Basso, “High-speed Instruction-set Coprocessor for Lattice-based Key Encapsulation Mechanism: Saber in Hardware,” *IACR Transactions on Cryptographic Hardware and Embedded Systems 2020*, 4, 443–466.

[2] Y. Zhu, M. Zhu, B. Yang, W. Zhu, C. Deng, C. Chen, S. Wei and L. Liu, “LWRpro: An Energy-Efficient Configurable Crypto-Processor for Module-LWR,” *IEEE Transactions on Circuits and Systems I:Regular Papers*, 68, 3, 2021, 1146–1159.

[3] A. Abdulgadir, K. Mohajerani, V. B. Dang, J.-P. Kaps, and K. Gaj, “A lightweight implementation of saber resistant against side-channel attacks,” 2021. In: Adhikari, A., Küsters, R., Preneel, B. (eds) Progress in Cryptology – INDOCRYPT 2021. INDOCRYPT 2021. Lecture Notes in Computer Science (LNCS), vol 13143. Springer, Cham.

[4] J. M. B. Mera, F. Turan, A. Karmakar, S. S. Roy and I. Verbauwhede, “Compact domain-specific co-processor for accelerating module lattice-based KEM,” *In 2020 57th ACM/IEEE Design Automation Conference (DAC)*, 2020, 1–6.

[5] V. B. Dang, F. Farahmand, M. Andrzejczak and K. Gaj, “Implementing and Benchmarking Three Lattice-Based Post-Quantum Cryptography Algorithms Using Software/Hardware Codesign,” *In 2019 International Conference on Field-Programmable Technology (ICFPT)*, 2019, 206–214.

QAs

Thanks for your attention