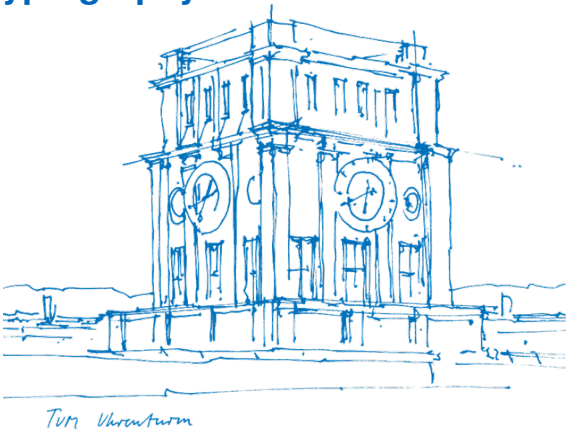
Chair of Security in Information Technology
Department of Electrical and Computer Engineering
Technical University of Munich

Masked Accelerators and Instruction Set Extensions for Post-Quantum Cryptography



Tim Fritzmann[1 3], Michiel Van Beirendonck[2], Debapriya Basu Roy[4], Patrick Karl[1], Thomas Schamberger[1], Ingrid Verbauwhede[2] and Georg Sigl[1]

[1] TU Munich, [2] KU Leuven, [3] Infineon, [4] IIT Kanpur

September 21, 2022

Content

- Towards Protected Post-Quantum Cryptography
- Masking SABER and KYBER
- Generic Ring Arithmetic Accelerator
- Masking Conversion Methods and Ciphertext Compression
- Masked Binomial Sampling
- System Integration and Results

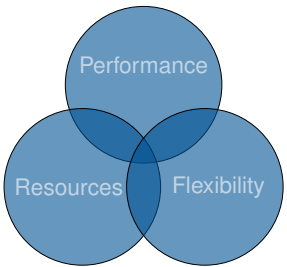
Towards Protected Post-Quantum Cryptography

- Cost analysis of side-channel countermeasures is still lacking
- Focus on **masking** as a countermeasure against differential power analysis

Towards Protected Post-Quantum Cryptography

- Cost analysis of side-channel countermeasures is still lacking
- Focus on **masking** as a countermeasure against differential power analysis

1) Masked HW/SW codesign of PQC finalists KYBER and SABER
2) Generic number theoretic transform multiplier
3) Novel masked ciphertext compression technique
4) Masked accelerators for critical non-linear operations

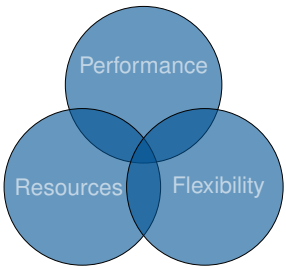


Masked hardware/software codesign of KYBER and SABER

Towards Protected Post-Quantum Cryptography

- Cost analysis of side-channel countermeasures is still lacking
- Focus on **masking** as a countermeasure against differential power analysis

1) Masked HW/SW codesign of PQC finalists KYBER and SABER
2) Generic number theoretic transform multiplier
3) Novel masked ciphertext compression technique
4) Masked accelerators for critical non-linear operations

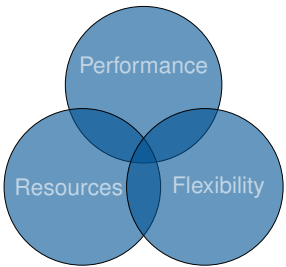


Masked hardware/software codesign of KYBER and SABER

Towards Protected Post-Quantum Cryptography

- Cost analysis of side-channel countermeasures is still lacking
- Focus on **masking** as a countermeasure against differential power analysis

1) Masked HW/SW codesign of PQC finalists KYBER and SABER
2) Generic number theoretic transform multiplier
3) Novel masked ciphertext compression technique
4) Masked accelerators for critical non-linear operations

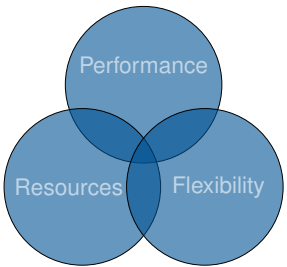


Masked hardware/software codesign of KYBER and SABER

Towards Protected Post-Quantum Cryptography

- Cost analysis of side-channel countermeasures is still lacking
- Focus on **masking** as a countermeasure against differential power analysis

1) Masked HW/SW codesign of PQC finalists KYBER and SABER
2) Generic number theoretic transform multiplier
3) Novel masked ciphertext compression technique
4) Masked accelerators for critical non-linear operations



Masked hardware/software codesign of KYBER and SABER

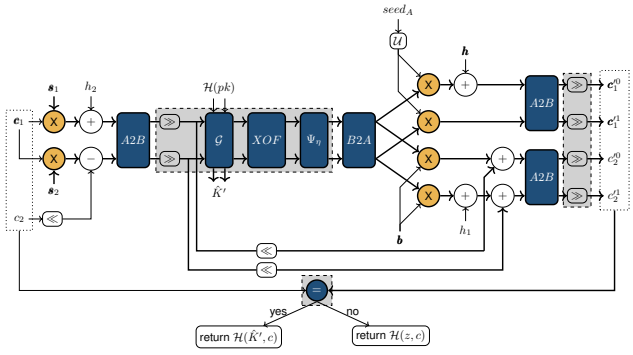
Towards Protected Post-Quantum Cryptography

- PKE/KEM schemes with KeyGen, Encaps, **Decaps**
- CCA-secure **Decaps** with re-encryption vulnerable against DPA
- Masking randomly splits secret variables into multiple shares to break the correlation between power consumption and the processed secret data
- Masking methods and complexity are different for KYBER and SABER

	KYBER	**SABER**
Method	MLWE	MLWR
Modulus q	3329	2^{13}

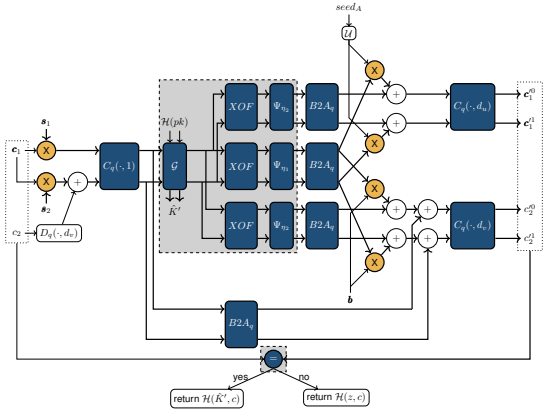
Content

Masked SABER.DECAPS



- Linear polynomial multiplication
- Hash function $\mathcal{G}$
- XOF (SHAKE)
- Binomial sampling ψ
- $A2B$ / $B2A$
- Masked ciphertext comparison

Masked KYBER.DECAPS

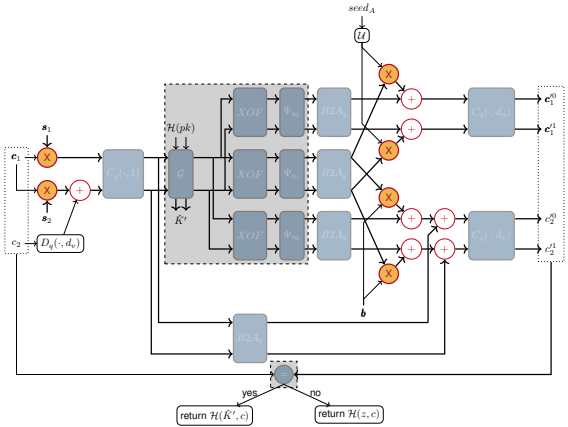


Main differences to SABER:

- Three sampling instances instead of one
- More complex $B2A$ conversion and ciphertext compression $C_q(\cdot, d_v)$

Content

Masked KYBER.DECAPS



Linear operations (ring arithmetic) are duplicated in a masked setting

$\rightarrow$ Fast multiplication important

Generic Ring Arithmetic Accelerator
Increasing Flexibility of NTT

Lattice-based algorithm	n	q	$\phi(x)$	NTT-based	$\lceil log_2(q') \rceil$
KYBER*	256	3329	$x^n + 1$	yes	12
Dilithium*	256	8380417	$x^n + 1$	yes	23
Falcon-512/1024*	512/1024	12289	$x^n + 1$	yes	14
SABER	256	8192	$x^n + 1$	no	34
ntruhps2048509	509	2048	$x^n - 1$	no	31
ntruhps2048677	677	2048	$x^n - 1$	no	32
ntruhps4096821	821	4096	$x^n - 1$	no	34
ntruhrss701	701	8192	$x^n - 1$	no	36

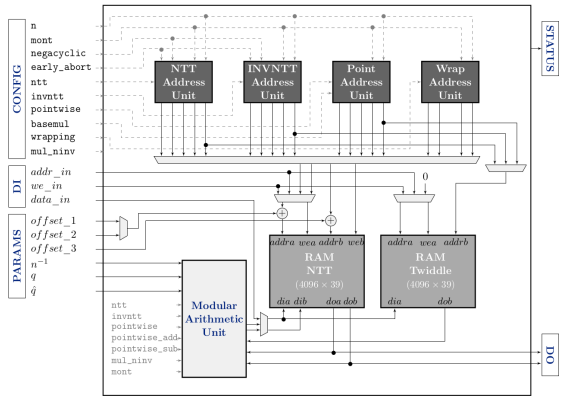
*** NIST PQC winners**

- NTT for polynomial multiplication is not always directly applicable
- Prime q can be lifted to any "NTT-friendly" prime, e.g., $q' > n \cdot q^2$

$\rightarrow$ Developed a generic NTT-based ring arithmetic accelerator with prime lift support

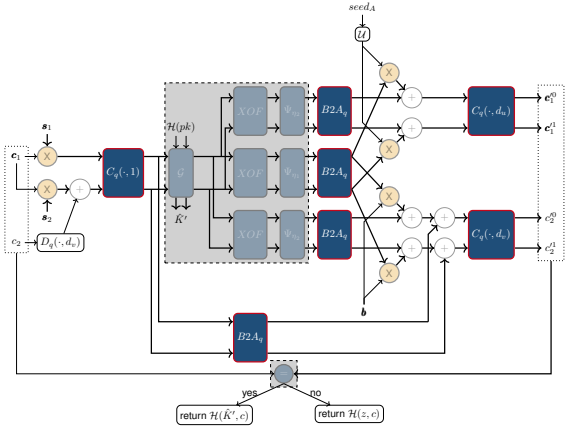
Generic Ring Arithmetic Accelerator
Increasing Flexibility of NTT

- Support for NTT operations
- Support for pointwise operations
- Support for positive and negative wrapped convolutions
- Support for early NTT abort (KYBER)



Content

Masked KYBER.DECAPS



Ring arithmetic requires **arithmetic sharing** and non-linear operations **Boolean sharing**
$\rightarrow$ Secure and efficient conversion methods required

Masking Conversion Methods

- Arithmetic sharing $X = A^0 + A^1$ or Boolean sharing $A = B^0 \oplus B^1$ with random mask $R = A^1 = B^1$
- Conversion is difficult to realize without recombining X

$$B^0 = (A^0 + R) \oplus R, \quad \text{or} \quad A^0 = (B^0 \oplus R) - R \tag{1}$$

Masking Conversion Methods

- Arithmetic sharing $X = A^0 + A^1$ or Boolean sharing $A = B^0 \oplus B^1$ with random mask $R = A^1 = B^1$
- Conversion is difficult to realize without recombining X

$$B^0 = (A^0 + R) \oplus R, \quad \text{or} \quad A^0 = (B^0 \oplus R) - R \tag{1}$$

- Generic conversion methods based on the secure masked addition SecAdd were presented in [CGV14] (suitable for hardware implementations)

KYBER $\texttt{MaskedCompress}_q(x, d)$

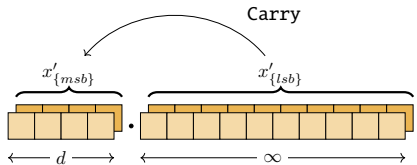
- Used to decrease the ciphertext size
- [OSPG18] proposed an interval comparison instead of the division

$$y = \texttt{Compress}_q(x, d) = \lfloor x' \rfloor \bmod 2^d \,, \quad x' = (2^d/q) \cdot x \tag{2}$$

KYBER $\texttt{MaskedCompress}_q(x, d)$

- Used to decrease the ciphertext size
- [OSPG18] proposed an interval comparison instead of the division

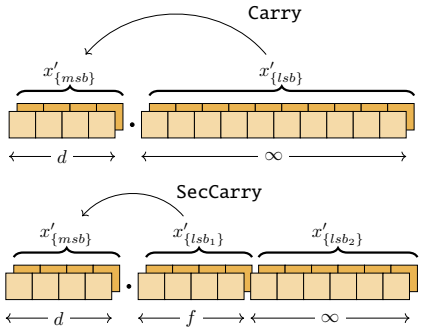
$$y = \texttt{Compress}_q(x, d) = \lfloor x' \rfloor \bmod 2^d \,, \quad x' = (2^d/q) \cdot x \tag{2}$$



KYBER $\texttt{MaskedCompress}_q(x, d)$

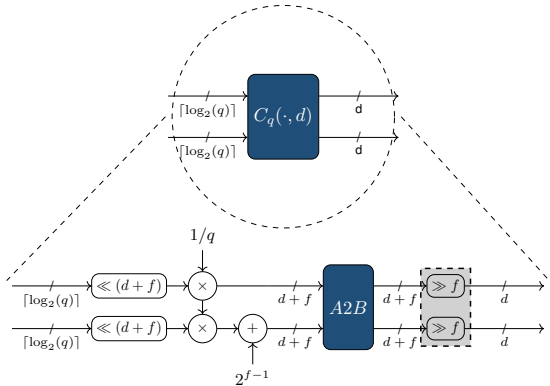
- Used to decrease the ciphertext size
- [OSPG18] proposed an interval comparison instead of the division

$$y = \texttt{Compress}_q(x, d) = \lfloor x' \rceil \bmod 2^d\,, \quad x' = (2^d/q) \cdot x \tag{2}$$



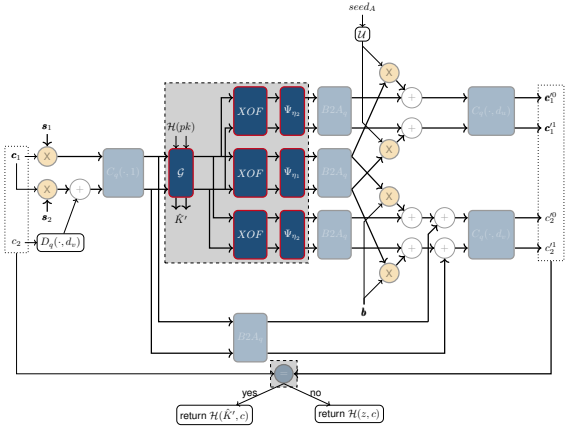
- Only f fractional bits to determine carry ($f = 13$ for KYBER)

KYBER MaskedCompress$_q(x, d)$



Content

Masked Binomial Sampling



→ Similar to conversion methods, binomial sampling requires attention (combines information of both shares)

Masked Binomial Sampling

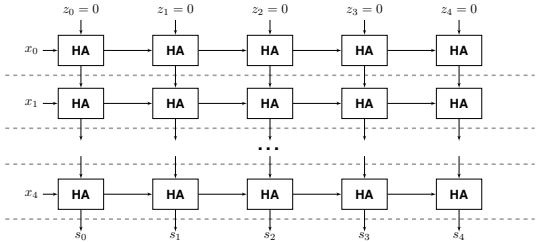
- Centered binomial distribution as approximation for the Gaussian distribution

$$\Psi_\eta = \sum_{i=0}^{\eta-1}(x_i - x_i') \mod q \quad \text{with } \eta \in [2, 5] \tag{3}$$

Masked Binomial Sampling

- Centered binomial distribution as approximation for the Gaussian distribution

$$\Psi_\eta = \sum_{i=0}^{\eta-1}(x_i - x_i') \mod q \quad \text{with } \eta \in [2,5] \tag{3}$$



Masked Binomial Sampling
Approach I (Threshold Implementation)

How does the adder tree look in a masked setting?

- TI is a function-based approach with incompleteness, correctness, and uniformity properties

Masked Binomial Sampling
Approach I (Threshold Implementation)

How does the adder tree look in a masked setting?

- TI is a function-based approach with incompleteness, correctness, and uniformity properties
- $f_1 : (x_0, z_0) \rightarrow (s_0)$ with $s_0 = x_0 \oplus z_0$
- $f_2 : (c_{i-1}, z_{i-1}, z_i) \rightarrow (c_i, s_i)$ with $c_i = c_{i-1} \wedge z_{i-1}$ and $s_i = z_i \oplus c_i$ for $i \neq 0$

Masked Binomial Sampling
Approach I (Threshold Implementation)

How does the adder tree look in a masked setting?

- TI is a function-based approach with incompleteness, correctness, and uniformity properties
- $f_1 : (x_0, z_0) \rightarrow (s_0)$ with $s_0 = x_0 \oplus z_0$
- $f_2 : (c_{i-1}, z_{i-1}, z_i) \rightarrow (c_i, s_i)$ with $c_i = c_{i-1} \wedge z_{i-1}$ and $s_i = z_i \oplus c_i$ for $i \neq 0$

$$f_1 : \qquad s_0^0 = x_0^0 \oplus z_0^0, \quad s_0^1 = x_0^1 \oplus z_0^1, \quad s_0^2 = x_0^2 \oplus z_0^2 \tag{4}$$

$$f_2 : \qquad c_i^0 = (c_{i-1}^1 \wedge z_{i-1}^1) \oplus (c_{i-1}^1 \wedge z_{i-1}^2) \oplus (c_{i-1}^2 \wedge z_{i-1}^1); \quad s_i^0 = z_i^0 \oplus c_i^0 \tag{5}$$

$$c_i^1 = (c_{i-1}^2 \wedge z_{i-1}^2) \oplus (c_{i-1}^0 \wedge z_{i-1}^2) \oplus (c_{i-1}^2 \wedge z_{i-1}^0); \quad s_i^1 = z_i^1 \oplus c_i^1 \tag{6}$$

$$c_i^2 = (c_{i-1}^0 \wedge z_{i-1}^0) \oplus (c_{i-1}^0 \wedge z_{i-1}^1) \oplus (c_{i-1}^1 \wedge z_{i-1}^0); \quad s_i^2 = z_i^2 \oplus c_i^2 \tag{7}$$

Masked Binomial Sampling
Approach I (Threshold Implementation)

How does the adder tree look in a masked setting?

- TI is a function-based approach with incompleteness, correctness, and uniformity properties
- $f_1 : (x_0, z_0) \rightarrow (s_0)$ with $s_0 = x_0 \oplus z_0$
- $f_2 : (c_{i-1}, z_{i-1}, z_i) \rightarrow (c_i, s_i)$ with $c_i = c_{i-1} \wedge z_{i-1}$ and $s_i = z_i \oplus c_i$ for $i \neq 0$

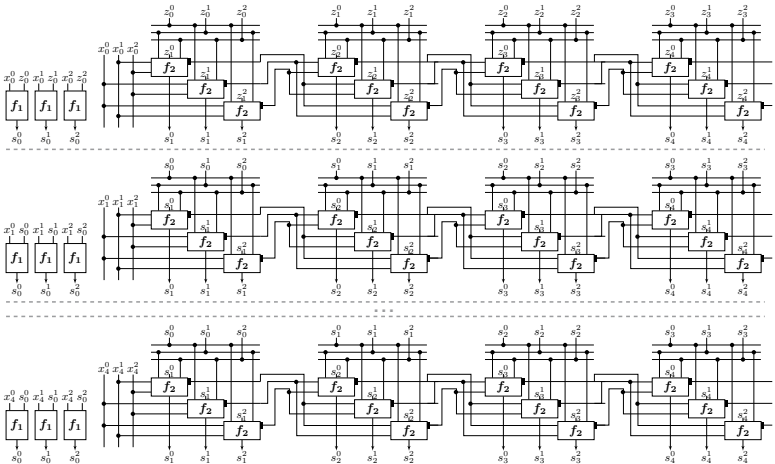
$$f_1 : \qquad s_0^0 = x_0^0 \oplus z_0^0, \quad s_0^1 = x_0^1 \oplus z_0^1, \quad s_0^2 = x_0^2 \oplus z_0^2 \tag{4}$$

$$f_2 : \qquad c_i^0 = (c_{i-1}^1 \wedge z_{i-1}^1) \oplus (c_{i-1}^1 \wedge z_{i-1}^2) \oplus (c_{i-1}^2 \wedge z_{i-1}^1); \quad s_i^0 = z_i^0 \oplus c_i^0 \tag{5}$$

$$c_i^1 = (c_{i-1}^2 \wedge z_{i-1}^2) \oplus (c_{i-1}^0 \wedge z_{i-1}^2) \oplus (c_{i-1}^2 \wedge z_{i-1}^0); \quad s_i^1 = z_i^1 \oplus c_i^1 \tag{6}$$

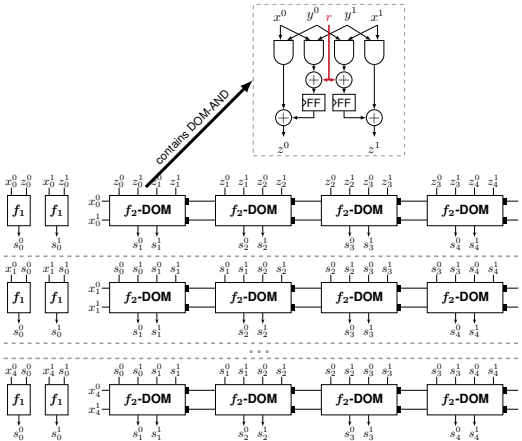
$$c_i^2 = (c_{i-1}^0 \wedge z_{i-1}^0) \oplus (c_{i-1}^0 \wedge z_{i-1}^1) \oplus (c_{i-1}^1 \wedge z_{i-1}^0); \quad s_i^2 = z_i^2 \oplus c_i^2 \tag{7}$$

Masked Binomial Sampling
Approach I (Threshold Implementation)



➡ High resource and randomness requirement

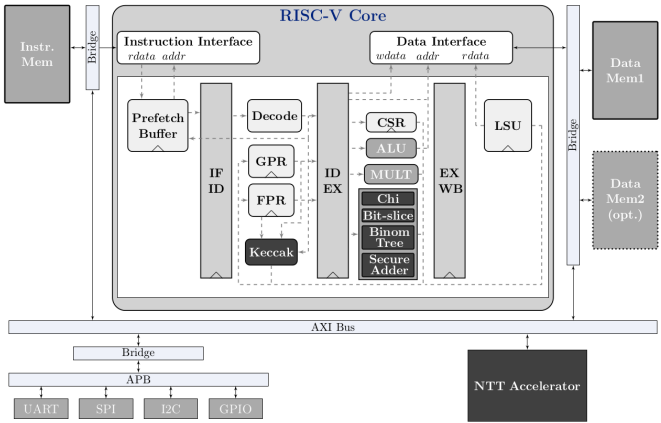
Masked Binomial Sampling
Approach II (Domain Oriented Masking)



- Compute non-linear carry computation $c_i = c_{i-1} \wedge z_{i-1}$ with DOM-AND
- Group shares into different domains
- Domain-crossing operations are refreshed with randomness
- \+ Reduces FPGA slices by $40\,\%$ and randomness requirement by $100\,\%$

Content

System Integration and Results
System Architecture



System Integration and Results
Performance Results (Clock Cycles)

Algorithm	Device	Decapsulation		Generate randomness
		unmasked	masked	
Kyber-768 [HKL+22]	ARM M4	–	$2,978,441$	$+0^{a)}$
Kyber-768 [BGR+21]	ARM M4	$882,000$	$3,116,000$	$+0^{a)}$
Kyber-768 [BGR+21]	ARM M0	$5,530,000$	$12,208,000$	–
Kyber-768 (this work)	**RISC-V**	$\mathbf{313,034}$	$\mathbf{1,235,460}$	$\mathbf{+167,190}$
Saber [BDK+21]	ARM M4	$1,123,280$	$2,833,348$	$+0^{a)}$
Saber (this work)	**RISC-V**	$\mathbf{351,370}$	$\mathbf{905,395}$	$\mathbf{+9,530}$

$^{a)}$ Onboard TRNG available.

- Masking performance overhead $\times 4.5$ for KYBER
- Masking performance overhead $\times 2.6$ for SABER

Thank you for your attention!

References I


Michiel Van Beirendonck, Jan-Pieter D'Anvers, Angshuman Karmakar, Josep Balasch, and Ingrid Verbauwhede.
A side-channel-resistant implementation of SABER.
J. Emerg. Technol. Comput. Syst., 17(2), April 2021.


Joppe W Bos, Marc Gourjon, Joost Renes, Tobias Schneider, and Christine van Vredendaal.
Masking Kyber: First-and higher-order implementations.
IACR Cryptol. ePrint Arch., 2021:483, 2021.


Jean-Sébastien Coron, Johann Großschädl, and Praveen Kumar Vadnala.
Secure conversion between Boolean and arithmetic masking of any order.
In *International Workshop on Cryptographic Hardware and Embedded Systems*, pages 188–205. Springer, 2014.


Daniel Heinz, Matthias J. Kannwischer, Georg Land, Thomas Pöppelmann, Peter Schwabe, and Daan Sprenkels.
First-order masked Kyber on ARM Cortex-M4.
2022.
https://eprint.iacr.org/2022/058.


Tobias Oder, Tobias Schneider, Thomas Pöppelmann, and Tim Güneysu.
Practical CCA2-secure and masked ring-LWE implementation.
IACR Transactions on Cryptographic Hardware and Embedded Systems, pages 142–174, 2018.