Sapphire: A Configurable Crypto-Processor for Post-Quantum Lattice-based Protocols

Utsav Banerjee*, Tenzin S. Ukyab, Anantha P. Chandrakasan

*utsav@mit.edu

Massachusetts Institute of Technology
Post-Quantum Cryptography

- Current public key cryptography vulnerable to quantum attacks
- NIST post-quantum crypto standardization in progress
- Round 2 has 26 candidates:
  - **Lattice-based** (9 KEM + 3 Sign)
  - Code-based (7 KEM)
  - Hash-based (1 Sign)
  - Multivariate (4 Sign)
  - Supersingular isogeny (1 KEM)
  - Zero-knowledge proofs (1 Sign)
Learning with Errors

- Learning with Errors (LWE) and its variants:
  - LWE (Standard Lattices)
  - Ring-LWE (Ideal Lattices)
  - Module-LWE (Module Lattices)

- Computational requirements (apart from standard arithmetic):
  - Modular arithmetic over various small primes
  - Polynomial arithmetic for Ring-LWE and Module-LWE
  - Sampling of matrices and polynomials from discrete distributions
Sapphire Crypto-Processor

- Energy-efficient configurable lattice-crypto-processor
Outline

- Efficient Lattice-Crypto Hardware Implementation
  - Configurable Modular Multiplier
  - Area-Efficient NTT
  - Energy-Efficient Sampler
- Chip Architecture
- Measurement Results
- Side-Channel Analysis
Modular Multiplication

**Algorithm** Modular Multiplication with Barrett Reduction

Require: $x, y \in \mathbb{Z}_q$, $m$ and $k$ such that $m = [2^k/q]$

Ensure: $z = x \cdot y \mod q$

1: $z \leftarrow x \cdot y$
2: $t \leftarrow (z \cdot m) \gg k$
3: $z \leftarrow z - (t \cdot q)$
4: if $z \geq q$ then
5: $z \leftarrow z - q$
6: end if
7: return $z$

**Reduction with fully configurable modulus:**

- configurable parameters $m$, $k$, $q$
- $m$ and $q$ up to 24 bits
- $16 \leq k \leq 48$
- requires 2 explicit multipliers for reduction
Modular Multiplication

**Algorithm** Modular Multiplication with Barrett Reduction

Require: $x, y \in \mathbb{Z}_q$, $m$ and $k$ such that $m = [2^k/q]$

Ensure: $z = x \cdot y \mod q$

1: $z \leftarrow x \cdot y$
2: $t \leftarrow (z \cdot m) \gg k$
3: $z \leftarrow z - (t \cdot q)$
4: if $z \geq q$ then
5: $z \leftarrow z - q$
6: end if
7: return $z$

**Reduction with pseudo-configurable modulus:**

- choice of $q$ from a set of primes
- reduction coded in digital logic
- requires no explicit multiplier for reduction
- up to $6\times$ more energy-efficient
Number Theoretic Transform

- NTT memory banks using dual-port SRAMs have large area overheads
- Proposed single-port SRAM-based NTT
- Based on constant geometry FFT data-flow
  [Pease, J. ACM, 1968]
- Polynomials split among four single-port SRAMs based on address parity:
  - Achieves > 30% area savings compared to dual-port implementation (without loss in throughput)
NTT Data Flow

- One butterfly per cycle
- No read / write hazards
- No energy overheads
Energy-Efficient PRNG

Standard CS-PRNG:

- SHAKE-128 / 256
- AES-128 / 256
- ChaCha20

Keccak-based PRNG:

24-cycles and 2.33 nJ per round @ 1.1V
Discrete Distribution Sampler

- **Uniform Sampling**
  - Uniformly random
  - $-\eta, 0, +\eta$

- **Binomial & Gaussian Sampling**
  - $-\sigma, 0, +\sigma$

- **Trinary Sampling**
  - $-1, 0, +1$

- **Rejection Sampling**
  - $0, q, 2^{32}$

**PRNG Core**
- Seed Reg r0
- Seed Reg r1
- Keccak State
- Keccak $f[1600]$ Round
- $> 32$
- $32$
- PRNG Out

**Sampler**
- Uniform Sampler
- Binomial Sampler
- Gaussian Sampler
- Trinary Sampler
- Rejection Sampler

**Bit Mask**
Test Chip Overview

- Crypto core integrated with RISC-V processor

![Test Chip Overview Diagram](image-url)
Protocol Implementations

- Following NIST Round 2 protocols were implemented on our test chip:

<table>
<thead>
<tr>
<th>CCA-KEM</th>
<th>LWE</th>
<th>Frodo</th>
</tr>
</thead>
<tbody>
<tr>
<td>Ring-LWE</td>
<td>NewHope</td>
<td></td>
</tr>
<tr>
<td>Module-LWE</td>
<td>CRYSALS-Kyber</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Signature</th>
<th>Ring-LWE</th>
<th>qTesla</th>
</tr>
</thead>
<tbody>
<tr>
<td>Module-LWE</td>
<td>CRYSALS-Dilithium</td>
<td></td>
</tr>
</tbody>
</table>

- Computations shared between crypto core and RISC-V processor:

**PKE / KEM:**

- Encoding / Compression
- CCA-KEM
- CPA-PKE

**Sign:**

- Encoding / Compression
- Sign

- RISC-V S/W with SHA-3 H/W
- Lattice-Crypto H/W
Implementation of RLWE and MLWE

- Efficient utilization of 24 KB polynomial memory with 8192 elements
  - CRYSTALS-Kyber
  - CRYSTALS-Dilithium
  - NewHope-512
  - qTesla-I
  - NewHope-1024
  - qTesla-III

- Crypto core used to accelerate sampling and polynomial arithmetic
- Protocol scheduling, compression and encoding performed on RISC-V processor
Implementation of LWE

- Polynomial memory tiled to support non-power-of-two-size matrix manipulation
- Crypto core used to accelerate sampling and matrix arithmetic
- Protocol scheduling, compression and encoding performed on RISC-V processor

![Diagram showing Frodo-640 and Frodo-976 configurations with matrix tiling and dimension annotations.]
Protocol Evaluation Results

Order of magnitude improvement in energy-efficiency and performance

* Cycle counts for CCA-KEM-Encaps and Sign
Protocol Evaluation Results

CCA-KEM-Encaps

<table>
<thead>
<tr>
<th>Energy (µJ)</th>
<th>Post-Quantum Security (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>18</td>
<td>240</td>
</tr>
<tr>
<td>16</td>
<td>220</td>
</tr>
<tr>
<td>15</td>
<td>200</td>
</tr>
<tr>
<td>14</td>
<td>180</td>
</tr>
<tr>
<td>13</td>
<td>160</td>
</tr>
</tbody>
</table>

- NewHope-1024
- Kyber-1024
- Kyber-768
- NewHope-512
- Kyber-512

Sign

<table>
<thead>
<tr>
<th>Energy (µJ)</th>
<th>Post-Quantum Security (bits)</th>
</tr>
</thead>
<tbody>
<tr>
<td>85</td>
<td>180</td>
</tr>
<tr>
<td>80</td>
<td>160</td>
</tr>
<tr>
<td>75</td>
<td>140</td>
</tr>
<tr>
<td>70</td>
<td>120</td>
</tr>
<tr>
<td>65</td>
<td>100</td>
</tr>
</tbody>
</table>

- Dilithium-IV
- Dilithium-III
- Dilithium-II
- Dilithium-I
- qTESLA-III-size
- qTESLA-III-speed
- qTESLA-I

* Measured using test chip operating at 1.1 V and 72 MHz
# Performance Comparison

<table>
<thead>
<tr>
<th>Design</th>
<th>Platform</th>
<th>Tech (nm)</th>
<th>VDD (V)</th>
<th>Freq (MHz)</th>
<th>Protocol</th>
<th>Area (kGE)</th>
<th>Cycles</th>
<th>Energy (µJ)</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>This work</strong></td>
<td>ASIC</td>
<td>40</td>
<td>1.1</td>
<td>72</td>
<td>NewHope-512-CCA-KEM-Encaps&lt;br&gt;NewHope-1024-CPA-PKE-Encrypt&lt;br&gt;Kyber-512-CCA-KEM-Encaps&lt;br&gt;Kyber-768-CPA-PKE-Encrypt&lt;br&gt;Kyber-768-CCA-KEM-Encaps&lt;br&gt;Frodo-640-CCA-KEM-Encaps&lt;br&gt;Dilithium-II-Sign</td>
<td>106</td>
<td>136,077&lt;br&gt;106,611&lt;br&gt;131,698&lt;br&gt;94,440&lt;br&gt;177,540&lt;br&gt;11,609,668&lt;br&gt;514,246</td>
<td>10.02&lt;br&gt;12.00&lt;br&gt;9.37&lt;br&gt;10.31&lt;br&gt;12.80&lt;br&gt;1129.95&lt;br&gt;54.82</td>
</tr>
<tr>
<td>Basu et al. [BSNK19] †</td>
<td>ASIC</td>
<td>65</td>
<td>1.2</td>
<td>169&lt;br&gt;200&lt;br&gt;158</td>
<td>NewHope-512-CCA-KEM-Encaps&lt;br&gt;Kyber-512-CCA-KEM-Encaps&lt;br&gt;Dilithium-II-Sign</td>
<td>1273&lt;br&gt;1341&lt;br&gt;1603</td>
<td>307,847&lt;br&gt;31,669&lt;br&gt;155,166</td>
<td>69.42&lt;br&gt;6.21&lt;br&gt;50.42</td>
</tr>
<tr>
<td>Albrecht et al. [AHH+18]</td>
<td>SLE 78</td>
<td>-</td>
<td>-</td>
<td>50</td>
<td>Kyber-768-CPA-PKE-Encrypt&lt;br&gt;Kyber-768-CCA-KEM-Encaps&lt;br&gt;Dilithium-II-Sign</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>Oder et al. [OG17]</td>
<td>FPGA</td>
<td>-</td>
<td>-</td>
<td>117</td>
<td>NewHope-1024-Simple-Encrypt</td>
<td>-</td>
<td>179,292</td>
<td>-</td>
</tr>
<tr>
<td>Howe et al. [HOKG18]</td>
<td>FPGA</td>
<td>-</td>
<td>-</td>
<td>167</td>
<td>Frodo-640-CCA-KEM-Encaps</td>
<td>-</td>
<td>3,317,760</td>
<td>-</td>
</tr>
<tr>
<td>Fritzmann et al. [FSM+19]</td>
<td>FPGA</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>NewHope-1024-CPA-PKE-Encrypt</td>
<td>-</td>
<td>589,285</td>
<td>-</td>
</tr>
</tbody>
</table>

† Only post-synthesis area and energy consumption reported
Side-Channel Analysis Setup

- Oscilloscope
- Power Supply
- Diff. Amp.
- Test Board
- Test Chip
Timing and SPA Side-Channels

- All key building blocks constant-time by design
- Energy consumption of sampling and polynomial arithmetic follows a narrow distribution with coefficient of variation ≤ 0.5% (\(\sigma / \mu\))
- SPA attacks target polynomial arithmetic:
  - Number Theoretic Transform
  - Coefficient-wise Multiplication
  - Coefficient-wise Addition
- SPA resistance of polynomial arithmetic evaluated using difference-of-means test with 99.99% confidence interval
Masking for DPA Security

- Protocol evaluations without any DPA countermeasures
- Masked NewHope-CPA-PKE-Decrypt based on additively homomorphic property:
  1. Generate secret message $\mu_r$
  2. Encrypt $\mu_r$ to its corresponding ciphertext $c_r = (\hat{u}_r, v'_r)$
  3. Compute $c_m = (\hat{u} + \hat{u}_r, v' + v'_r)$ where $c = (\hat{u}, v')$ is the original ciphertext
  4. Decrypt $c_m$ to obtain $\mu_m = \mu \oplus \mu_r$ where $\mu$ is the original message
  5. Recover original message as $\mu = \mu_m \oplus \mu_r$

- Masked decryption using same hardware; 3× slower than unmasked version
- Masking increases decryption failure rate, which can be resolved by decreasing std. dev. $\sigma$ of error distribution (at the cost of slightly lower security level)
- Leakage tests and CCA-KEM masking – work in progress

[Reparaz et al, PQCrypto, 2016]
Conclusion

- Configurable crypto-processor for LWE, Ring-LWE and Module-LWE protocols
- Area-efficient NTT, energy-efficient sampler and flexible parameters
- ASIC demonstration of NIST Round 2 CCA-KEM and signature protocols: Frodo, NewHope, Kyber, qTesla, Dilithium
- Order of magnitude improvement in performance and energy-efficiency compared to state-of-the-art software and hardware
- Hardware building blocks constant-time and SPA-secure by design; masking can also be implemented for DPA security
Acknowledgements

- Texas Instruments for funding
- TSMC University Shuttle Program for chip fabrication
Questions

Quantum Adversary

Client

RSA, ECC, ...

Lattice Crypto

Server