TECHNICAL DOSSIER // SECURITY & VPNS

Post-Quantum Cryptography Hardware Acceleration: Kyber & Dilithium

NTT Butterfly Pipelines, Modular Arithmetic Datapaths, and Side-Channel Silicon Hardening

Published: October 29, 2026 | Distribution: Open Engineering Community | Document ID: LT-2026-B15

EXECUTIVE SUMMARY & MICROARCHITECTURAL SCOPE

This cryptographic engineering dossier evaluates dedicated silicon hardware acceleration for NIST-standardized Post-Quantum Cryptography (PQC), focusing on ML-KEM (CRYSTALS-Kyber) and ML-DSA (CRYSTALS-Dilithium). We examine Number Theoretic Transform (NTT) butterfly pipelines, modular Montgomery polynomial multiplication, SHA-3/Keccak sponge coprocessors, and microarchitectural countermeasures against Differential Power Analysis (DPA).

Cryptographic Primitive	Classical ECC (ECDSA / X25519)	ML-KEM-768 (Kyber NIST Level 3)	ML-DSA-65 (Dilithium NIST Level 3)
Mathematical Hardness	Elliptic Curve Discrete Log (ECDLP)	Module Learning With Errors (M-LWE)	Module Short Integer Solution (M-SIS)
Public Key Size	32 to 64 Bytes	1,184 Bytes	1,952 Bytes
Ciphertext / Signature Size	32 to 64 Bytes	1,088 Bytes (Ciphertext)	3,293 Bytes (Signature)
Hardware Gate Count (ASIC)	35k to 55k NAND2 gates	180k to 260k NAND2 gates	280k to 420k NAND2 gates
Execution Cycles (HW Accelerated)	45,000 cycles (Point Mult)	12,500 cycles (Decapsulation)	28,000 cycles (Sign Generation)
Quantum Attack Resistance	Vulnerable to Shor's Algorithm	Quantum-Safe (Lattice-Based)	Quantum-Safe (Lattice-Based)

1. The Quantum Cryptanalytic Threat & NIST Standards

Shor's quantum algorithm solves the integer factorization and discrete logarithm problems in polynomial time, rendering RSA, Diffie-Hellman, and Elliptic Curve Cryptography (ECC) fundamentally obsolete upon the arrival of cryptanalytically relevant quantum computers (CRQCs). In response, NIST standardized FIPS 203 (ML-KEM / Kyber) for key encapsulation and FIPS 204 (ML-DSA / Dilithium) for digital signatures, rooted in lattice mathematics.

2. The Computational Core: Number Theoretic Transform (NTT)

Lattice-based cryptography relies heavily on polynomial ring multiplications ($R_q = Z_q[X] / (X^n + 1)$). Naive schoolbook polynomial multiplication requires $O(n^2)$ modular operations. By operating in prime modulus rings where $2n$-th roots of unity exist, the Number Theoretic Transform (NTT)—a discrete Fourier transform over finite fields—reduces multiplication complexity to $O(n \log n)$, executing via radix-2 Cooley-Tukey butterfly operations.

3. Hardware Pipeline Architecture & Memory Bank Conflicts

A dedicated PQC coprocessor implements parallel NTT butterfly pipelines coupled with Montgomery or Barrett modular reduction units. The primary architectural challenge is memory access: during butterfly stages, pairs of polynomial coefficients must be read and written simultaneously. Hardware cores utilize multi-bank SRAM layouts with conflict-free address mapping schemes to achieve continuous 1-cycle-per-butterfly throughput.

4. Keccak/SHAKE Coprocessing & Side-Channel Hardening

Over 60% of total ML-KEM/ML-DSA execution time in software is spent in symmetric pseudo-random generation (SHAKE-128/256 and SHA3). Dedicated silicon features tightly coupled Keccak hardware accelerators with 64-bit round pipelines. Furthermore, hardware architectures enforce constant-time execution to defeat timing attacks and incorporate first-order Boolean masking to prevent Differential Power Analysis (DPA) and electromagnetic leakage.

STRATEGIC ENGINEERING TAKEAWAYS & DEPLOYMENT CHECKLIST

• **Silicon Acceleration Imperative:** Hardware NTT and Keccak coprocessors accelerate ML-KEM and ML-DSA operations by 15x to 40x compared to software implementations on standard CPUs.
• **Bandwidth & Memory Impact:** PQC keys and ciphertexts are 30x to 100x larger than classical ECC, demanding expanded internal buffer SRAM and DMA transfer bandwidth.
• **Physical Security Hardening:** Hardware implementations must integrate algorithmic masking, noise injection, and shuffle techniques to protect against side-channel analysis.