The Quantum Threat to Blockchains - 2026 Report

Alex Pruden / CEO, Project Eleven

Conor Deegan / CTO, Project Eleven

Executive Summary

Quantum computers represent a new era of computation, and the end of an era for classical cryptography. While current quantum computers are still relatively nascent in terms of capability, they are advancing rapidly. In light of current progress, they could make widely-deployed asymmetric cryptography obsolete within a decade, potentially even before this decade is out.

A quantum computer that breaks classical cryptography is called a “cryptographically relevant quantum computer” or CRQC. The moment a CRQC is realized is what we call “Q-Day”, and on that day, trillions of dollars currently secured under existing classical cryptographic schemes will be vulnerable. Only by migrating to cryptography that is secure against quantum attack can blockchains have any guarantee of being secure into the future.

The threat is accelerating as progress compounds across three dimensions: hardware improvements in physical qubit quality and scale, advances in quantum error correction efficiency, and algorithmic optimizations that reduce resource requirements. Recent developments over the last two years show this acceleration:

→ Google’s 105-qubit Willow processor experimentally demonstrated quantum error correction below-threshold; a key milestone for scaling quantum computers. [1]

→ Resource estimates for breaking the elliptic curve cryptography securing Bitcoin and the wider digital asset ecosystem have collapsed in parallel: a recent paper from Google Quantum AI and Stanford concludes that roughly 1,200 logical qubits and a runtime on the order of nine minutes on superconducting hardware would suffice, a runtime shorter than Bitcoin's ten-minute average block settlement. [2]

→ A 2026 neutral-atom proposal from researchers at Caltech and Oratomic, including John Preskill (Caltech) and Dolev Bluvstein (Oratomic), shows that Shor's algorithm can be executed at cryptographically relevant scales using as few as 10,000 reconfigurable atomic qubits, orders of magnitude below 2021 baseline estimates. [3]

This progress profile means quantum computing advancement may potentially follow a “nothing-and-then-all-at-once” exponential trajectory not unlike other emerging technologies such as AI. **Our analysis suggests that, based on current trends, Q-Day is more likely to occur than not by 2033, and potentially even as soon as 2030.**

This timeline is a consequence of the fact that small improvements in error correction efficiency, higher qubit connectivity, or better code design create potential feedback loops leading to order-of-magnitude reductions in the resources needed for cryptanalysis. What appears as incremental hardware progress today might rapidly converge to a CRQC with little warning. Waiting until that point is clearly on the horizon risks insufficient time for post-quantum cryptography to be selected, tested, and deployed.

Blockchain systems are especially vulnerable. Unlike traditional systems with ephemeral keys and regular rotation schedules, blockchain addresses often hold funds on static public keys for years or decades. Once compromised, these keys provide direct access to financial assets with no recovery mechanism. And the public key cryptography used in signature schemes is the primary mechanism for determining “ownership” of digital assets. “Not your keys, not your crypto” evaporates in a post-quantum world.

The window for the world to migrate to post-quantum cryptography is narrowing. The distributed nature of blockchain networks means that migration to post-quantum cryptography may take the better part of a decade, longer than other centralized systems. The risk that the migration timeline is not complete by Q-Day motivates the urgency to proactively add post-quantum cryptography to blockchains.

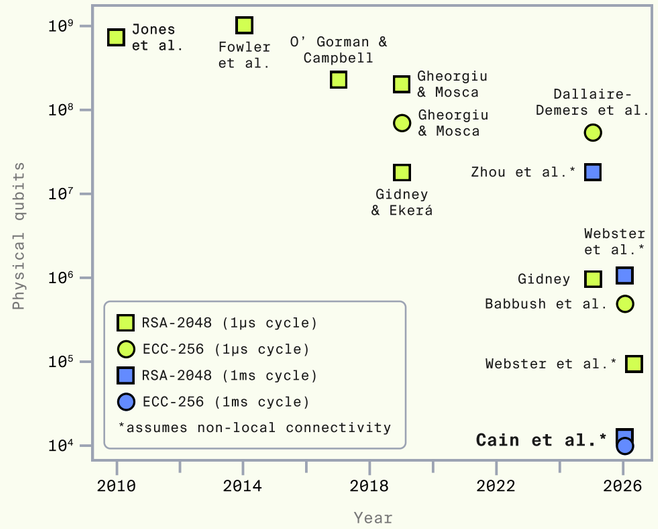


Figure 1: Estimated number of physical qubits to run Shor’s algorithm versus year of publication for prior resource estimates and the current work.

THIS REPORT COVERS

- What a quantum computer is, why it represents a threat, and where we stand on building a CRQC
- How existing cryptography in blockchains is vulnerable to a CRQC
- The state of post-quantum cryptography and how it needs to be applied to secure blockchains

"Migration to quantum-resistant cryptography is no longer optional but imperative for any blockchain system expected to be trusted and secure value into the future."

OVERVIEW

Table of Contents

01

State of Quantum Computing & Future Outlook

An assessment of the quantum computing landscape, the timeline to cryptographic relevance, and the engineering challenges that separate theory from reality.

SECTION ONE

What Is a Quantum Computer?

Quantum computing leverages quantum mechanics — the most validated physical theory ever created. While classical computers operate on straightforward logical concepts, quantum computers rely on principles of quantum mechanics that challenge everyday intuitions. Even though "computer" is the common term, quantum computers and classical computers differ in very fundamental ways.

Understanding the core tenets of quantum mechanics is critical to grasping both the potential power of a quantum computer and the challenges of building one. Here are the key facets that differentiate a quantum computer from a classical one.

KEY QUANTUM MECHANICAL CONCEPTS

Superposition

Qubits can exist in multiple states simultaneously, unlike classical bits that are strictly 0 or 1.

Entanglement

Particles become linked so the state of one instantly determines the other, regardless of distance.

Interference

Quantum algorithms exploit wave-like interference to amplify correct answers and cancel wrong ones.

QUANTUM CHALLENGES

No-Cloning Theorem

Quantum states cannot be copied — requiring fundamentally different approaches to computation.

Measurement Collapse

Measuring a quantum state destroys the **superposition**, making careful circuit design essential.

Probabilistic & Fragile

Quantum states are inherently probabilistic and fragile — any unintended interaction can collapse the **superposition**, demanding extremely controlled environments and fundamentally limiting scale.

SECTION ONE

Superposition & Entanglement

Superposition

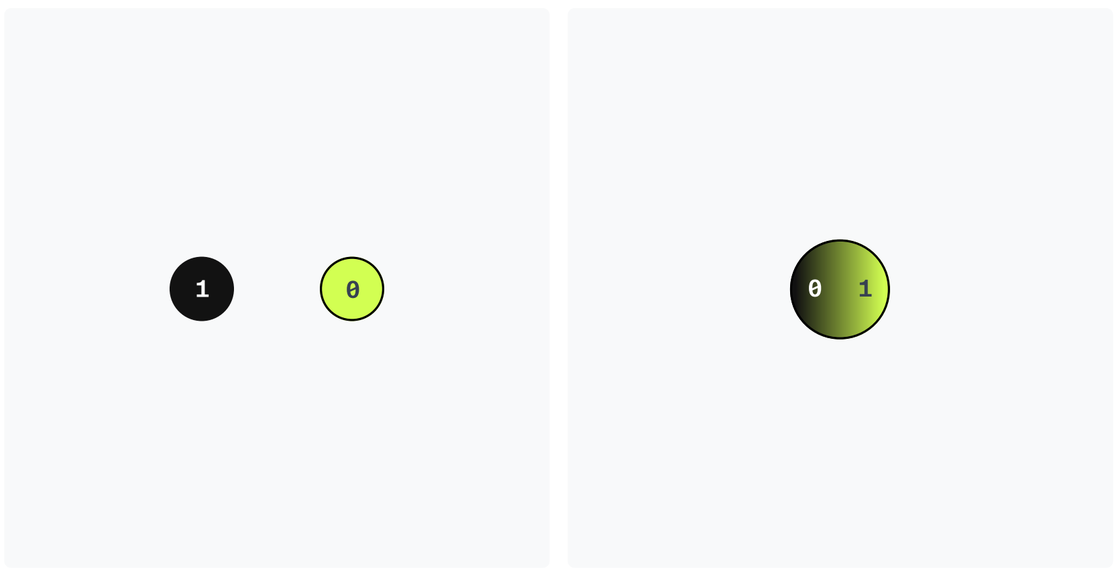
Qubits, the fundamental building blocks of quantum computers, don't necessarily occupy definite states like classical bits. Instead, due to the nature of quantum mechanics they can exist in a linear combination of possible states, described by a wavefunction. This linear combination is called **superposition** and represents "being" in multiple states at once.

Whereas a classical bit definitively represents either 0 or 1, a qubit can be in a **superposition** of both simultaneously. The outcome you get upon measurement depends on a probability distribution described by the wave function. This **superposition** allows a qubit to encode a much richer space of states than a classical bit, which is what gives quantum computing its exponential potential.

Entanglement

In quantum mechanics, particles can be entangled, meaning their states become linked in such a way that they must be described as a single system. Even when separated by large distances, the measurement outcome of one particle is correlated with (or even determined by) the state of the other, even without *any physical connection at all*.

QUANTUM **SUPERPOSITION**



CLASSICAL

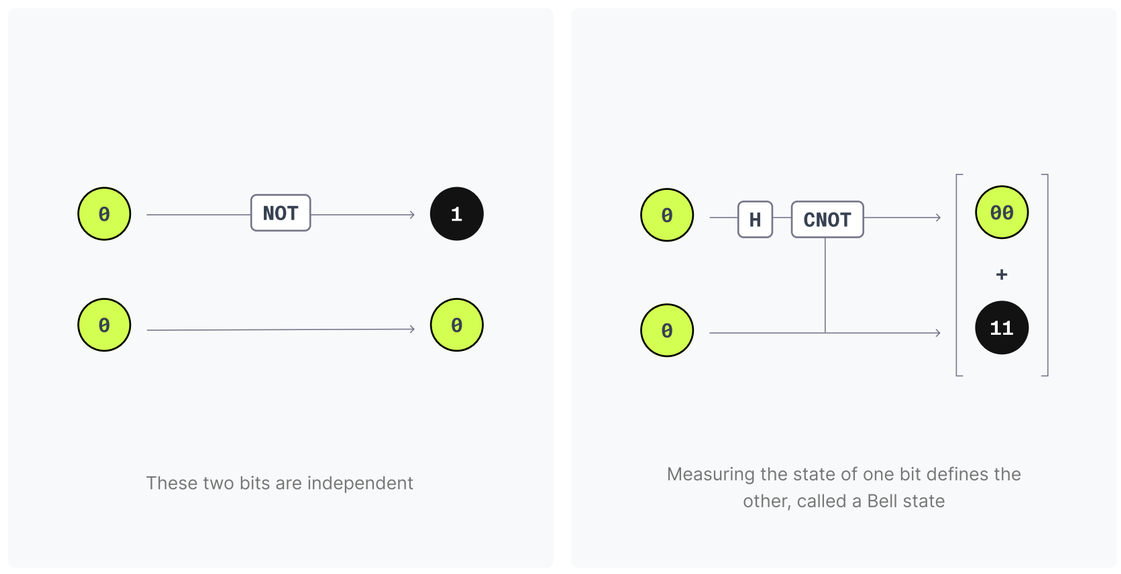
Classical bits may take the value of **EITHER** one **OR** zero

QUANTUM

Quantum **superposition** enables qubits to represent **MANY STATES AT ONCE**

Figure 2: Illustration of the quantum computing concept of superposition

QUANTUM ENTAGLEMENT



CLASSICAL

Classical bits **ARE LOGICALLY, NOT PHYSICALLY, DEPENDENT** on other bits

QUANTUM

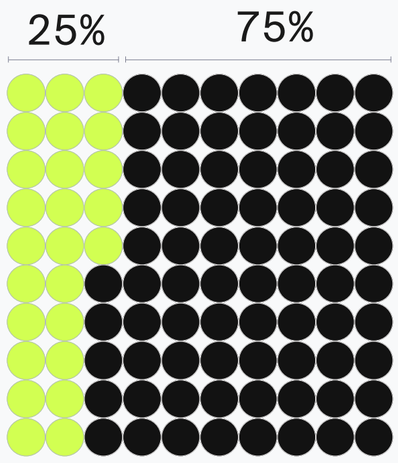
The **STATE OF ONE QUBIT** in an entangled pair **CAN AFFECT THE OTHER, EVEN SEPARATED BY DISTANCE**

Figure 3: Illustration of the quantum computing concept of entanglement

SECTION ONE

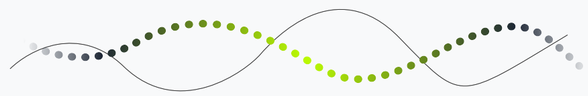
Interference

QUANTUM INTERFERENCE



CLASSICAL

Classical computation **CAN ONLY REPRESENT PROBABILITIES THAT SUM TO ONE**



QUANTUM

Quantum states are waveforms that have **POSITIVE** and **NEGATIVE** amplitudes that **CAN COMBINE TO AMPLIFY THE RIGHT ANSWER**

Figure 4: Illustration of the quantum computing concept of interference

In classical systems, outcome probabilities are positive, and sum to one. But in quantum mechanics, amplitudes (the components of the wavefunction) can be both positive AND negative, and thus interfere with each other before measurement. These amplitudes can reinforce (constructive interference) or cancel out (destructive interference), depending on their relative phases. Quantum computers exploit this phenomenon to “steer” a computation toward correct answers.

Instead of just exploring all paths in parallel, a quantum algorithm is designed so that wrong answers interfere destructively and cancel out, while desirable paths leading to right answers interfere constructively and dominate the final result, providing a unique advantage over classically randomized approaches.

The above properties are what enable quantum computers to outperform classical ones for certain tasks. By leveraging **superposition**, entanglement, and interference in just the right way, quantum computers can effectively “parallelize” what would otherwise be a serial classical computation[1].

[1] ‘Parallelization’ is a simplification meant to build intuition. Quantum computers do not literally evaluate all possible solutions simultaneously in the way a classical parallel computer would. Rather, quantum algorithms exploit interference to amplify the probability of correct answers and suppress incorrect ones. The computational advantage arises from the structure of the algorithm, not from brute-force parallel evaluation.

SECTION ONE

Quantum Challenges

However, certain quantum mechanical properties also present challenges to realizing a quantum computer in practice.

No-Cloning Theorem

Unlike a classical computer, it is impossible to “copy” quantum states. This no-cloning theorem makes the implementation of low-level primitives that we take for granted in classical computing (like memory registers) much more complex in practice. Instead, operations like quantum teleportation and entanglement swapping must be used to safely transmit or share quantum information across the system during computation.

Measurement Collapse

Critically, in classical computing, measurement is passive (reading memory doesn’t change it). But in quantum mechanics, the act of measuring a system collapses a **superposition** into a definite state. To gain meaningful advantage from a quantum computer, that **superposition** must be carefully preserved until the right moment.

Probabilistic & Fragile

The nature of the quantum mechanical wavefunction that describes reality at small scales is inherently probabilistic. Any unintended interaction (even an accidental subatomic interaction) can replicate the effect of a measurement, instantly destroying this fragile system of probabilities into definite states and removing the advantages of quantum computation described above. Thus, the theoretical potential of a fault-tolerant quantum computer is almost matched by the daunting engineering challenges involved in practically building one.

Despite the challenges, quantum computing remains an area of intense research interest because it enables a much more powerful computational paradigm. Whereas the resources required to represent a complex system (such as individual molecules in a fluid) might overwhelm even the most powerful classical hardware, quantum computers can harness **superposition** and entanglement to solve these otherwise intractable problems, including those that form the basis for modern cryptosystems.

SECTION ONE

Quantum Computers & Their Impact on Cryptography

When quantum computers were first theorized, they were mainly considered to be an intellectual curiosity. In fact, the entire motivation behind the concept was to enable more accurate physics simulations. At first, the field seemed to be forever destined to remain an academic curiosity.

Shor's Algorithm

That all changed in 1994, when mathematical physicist Peter Shor showed that a hypothetical quantum computer could be used to solve a math problem that no feasible classical computer ever could. Shor's algorithm describes a process for factoring large numbers.[2] Leveraging the unique properties of quantum mechanics (entanglement, interference), this algorithm can quickly recover the prime factors of a very large number where a classical computer would take billions of years. To this day, it remains one of the only known quantum algorithms with an exponential advantage over classical computing.

The problem of factoring large numbers, while seemingly also academic, is actually the basis for much of modern cryptography. It underlies our entire modern internet infrastructure, used in everything from banking transactions to military communications systems. Its security depends entirely on the computational difficulty of factoring, which evaporates in the face of Shor's algorithm.

Primary threat to blockchains

→ **Shor's algorithm** breaks public key cryptography (RSA, ECC), securing digital signatures and key exchange. This would allow private key recovery from a public key and signature forgery [4] — the primary quantum threat to blockchain systems.

Grover's Algorithm

A short time later, in 1996, computer scientist Lov Grover developed another quantum algorithm with cryptographic implications. Grover's algorithm provides a quadratic speedup for searching unsorted databases and, more importantly for cryptography, for reversing hash functions and brute-forcing encryption keys. While not as dramatic as Shor's exponential speedup, Grover's algorithm is still relevant to certain cryptographic systems.

Grover's algorithm weakens symmetric cryptography and hash functions, requiring larger key sizes and hash outputs to maintain equivalent security. While less immediately threatening than Shor's algorithm, at some point it will necessitate adjustments to cryptographic parameters to ensure the same level of security for hash functions and symmetric encryption [5].

Comparative threat to blockchains

→ **Grover's algorithm** weakens symmetric cryptography (AES) and hash functions (SHA-256). This is a secondary threat (manageable by doubling key sizes) unlike the existential threat posed by Shor's to public key cryptography.

[2] Technically it applies to an entire category known generally as the "hidden subgroup" problem. In addition to including the RSA assumption, it also includes the Discrete Logarithm Problem (the basis for ECDSA) and other problems that form the basis of nearly all classical asymmetric cryptography

SECTION ONE

Algorithmic Complexity Comparison: Shor's vs. Grover's

Although both algorithms are relevant, the cryptographic relevance of these algorithms (and the threat they pose) depends on how they compare to the best classical alternatives at scale.

For solving the discrete logarithm problem (the basis of elliptic curve cryptography like ECDSA):

- Best classical algorithm: Pollard's rho runs in time exponential to n in an n-element group, which for a 256-bit elliptic curve means approximately 2^{128} (or roughly 340 trillion trillion trillion) operations.
- Shor's algorithm: Runs in polynomial time, providing an exponential speedup, requiring on the order of tens of millions of operations to factor a 256-bit key.

For brute-force search problems (breaking symmetric encryption like AES):

- Best classical algorithm: Exhaustive search requires an exponential number of operations for an n-bit key.
- Grover's algorithm: Reduces the classical resources required by a modest quadratic factor.

Of course, describing an algorithm in theory and running it on a practical device are two different things. As discussed in the prior section, practically implementing a fault-tolerant quantum computer has proved to be a major challenge for the field over the last three decades.

Despite recent progress, current quantum computers are not capable of breaking cryptography with Shor's, Grover's, or any other known algorithm. Understanding whether and how that will change requires understanding the underlying challenges of building a fault-tolerant quantum computer.

To assess when that will change, we need to examine the core obstacle in quantum computing: maintaining coherence in the presence of noise while scaling up from proof-of-concept to a practical, fault-tolerant system.

CLASSICAL VS QUANTUM ALGORITHMIC COMPLEXITY

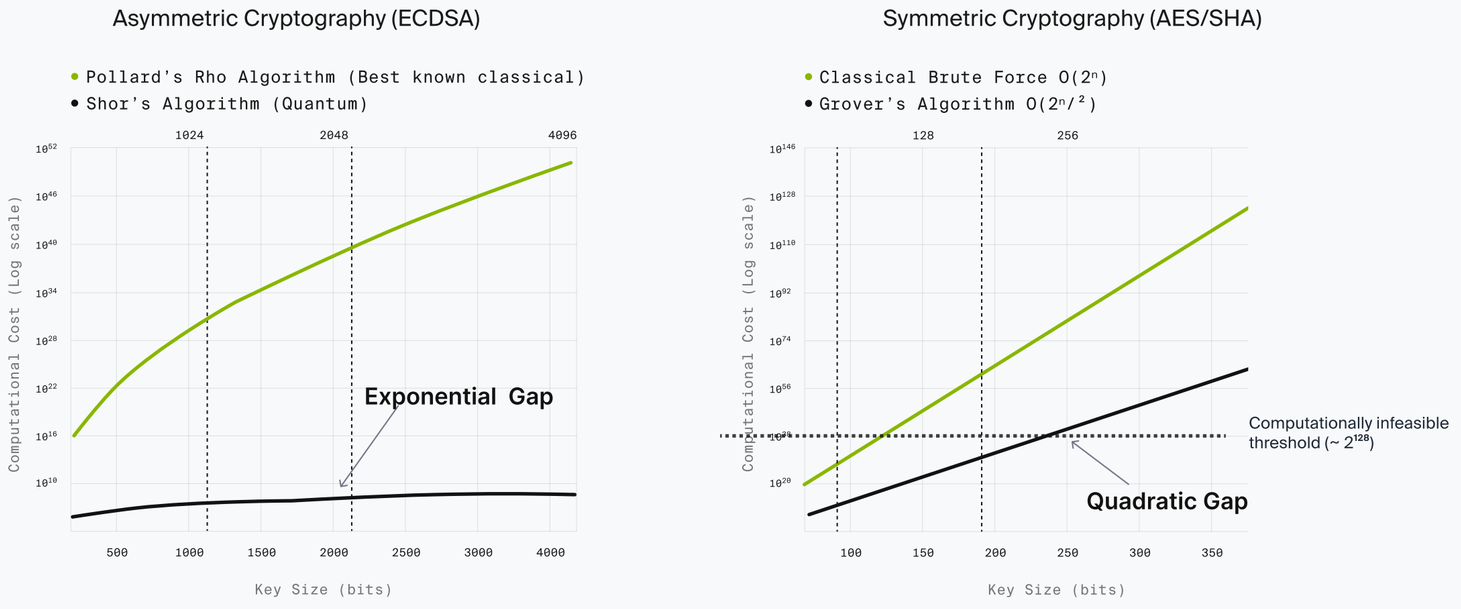


Figure 5: Relative Advantage of Shor's vs. Grover's Algorithm

SECTION ONE

How to Build a Practical Quantum Computer

A useful way to think about large-scale quantum computation is to imagine trying to fill a giant container with water. The problem is that the container is made of a material that leaks constantly.

If you pour water into it slowly, the water simply drains away before the container ever fills. The only way to succeed is to add water faster than it leaks out.

Quantum computation has the same problem. Physical qubits constantly accumulate errors, which slowly "leak" reliability out of the computation. Error correction acts like reinforcing the walls of the container, slowing the leak. But it also makes the container inherently larger and more complicated to fill.

A quantum computer becomes useful only when the system can execute enough reliable operations before errors drain away the computation.

More concretely, we want to predict how many physical qubits operating for how long are needed to run Shor's algorithm. But as the metaphor above illustrates, that answer depends not only on quantity, but on the *quality* of those qubits.

A single physical qubit is useless on its own, because it is too noisy and error-prone. It needs to be optimized for reliability in order to be useful. The threshold that we care about is the *logical error* rate per cycle at utility scale.

As quantum computation gets repeated over many cycles, the results get more and more unreliable. Error correction counteracts that, but adds additional computational overhead.

2025 Shor's algorithm resource requirements [5] (Google)

~1,000,000 physical qubits

A "logical" error rate of 1 in a quadrillion per cycle.

are needed to break RSA-2048. Note that newer estimates focused on ECDSA are even lower

To build a CRQC is not a single engineering challenge but a stack of interdependent capabilities, where each layer depends on the one below it and feeds into the one above. Assessing how close we are to a CRQC requires a structured framework for measuring progress across the four-layer quantum computing stack [7].

The 4-Layer Stack

- **Layer 1: Physics** : Quality of individual physical qubits; decoherence time and gate fidelity.
- **Layer 2: Error Correction** : Bundles physical qubits into logical qubits; determines physical-to-logical ratio, logical error rate, and logical cycle time.
- **Layer 3: System Integration** : Classical decoder, real-time error feed-forward, connectivity between components, and stability at scale.
- **Layer 4: Algorithm Demand** : Logical qubit count, circuit depth, and error rate requirements set by **Shor's algorithm** against RSA-2048.

A QUANTUM COMPUTING STACK

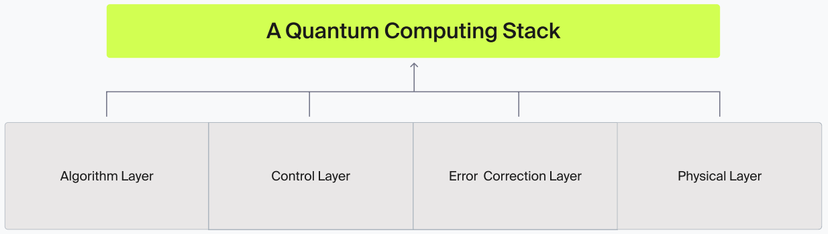


Figure 6: A simplified framework for understanding the quantum computing “stack”

SECTION ONE

Layer 1: Physics — Hardware Modalities

Everything begins with the quality of individual physical qubits. Two properties matter most: **decoherence time** (how long a qubit retains its quantum state before decaying) and **gate fidelity** (how accurately operations can be performed on it).

Because quantum mechanics governs the entire world around us, it is actually possible to realize a quantum computer in a number of different ways. These are sometimes referred to as “modalities”, which is the nomenclature we will use for the rest of this report. The choice of modality impacts the qubit quality, as well as dictates the engineering constraints for physically building the system. Higher coherence and fidelity mean fewer errors at the source, which reduces the burden on every layer above. Current state-of-the-art platforms achieve two-qubit gate fidelities between 99.9% and 99.99%. Measurement fidelity (how accurately you can read out a qubit’s state) and correlated noise (errors that affect multiple qubits simultaneously) are additional physical-layer constraints.

MODALITY	KEY STRENGTHS	PRIMARY CHALLENGES	EXAMPLE ARCHITECTURES
Superconducting	Fast gate speeds, mature fabrication pipelines, strong integration with classical control	Cryogenic infrastructure; wiring density; frequency crowding; correlated noise	Google Sycamore/Willow IBM Eagle/Heron
Trapped Ions	Very high gate fidelities; long coherence times; uniform qubits	Slower gate and measurement times; scaling trap arrays; parallelization limits	Quantinuum H-series IonQ Forte
Neutral Atoms	Reconfigurable geometry; programmable connectivity; natural compatibility with non-local codes	Measurement latency; laser stability; large-scale control complexity	QuEra Aquila Pasqal Fresnel Infleqtion Scorpius Oratomic
Silicon Spin	CMOS compatibility; potential for dense integration; leverages semiconductor industry	Two-qubit fidelity; variability; crosstalk; integration complexity	Intel Tunnel Falls Diraq Crossbar QuTech QARPET
Photonic	Room-temperature operation; low decoherence in transit; high-speed signal propagation	Probabilistic entangling gates; large resource overhead; complex fusion schemes	PsiQuantum Xanadu Borealis

A more detailed analysis of the different physical modalities can be found in Appendix B.

SECTION ONE

Layer 2: Quantum Error Correction

Raw physical qubits are too noisy for cryptographic computation. Error correction bundles groups of physical qubits into a single, more reliable logical qubit using specialized codes. The efficiency of this process determines the physical-to-logical qubit ratio (how much raw hardware is consumed per usable qubit), the logical error rate (how many operations can be chained before failure), and the logical cycle time (how quickly the error correction loop runs, which sets the clock speed of the overall system). In effect, it determines whether or not a given system can scale and still give a useful result.

In recent years, this aspect of the quantum computing stack has seen the greatest improvement, hallmarked by Google's "Willow" below-threshold demonstration in 2024. The significance of this was that, by increasing code distance (using more physical qubits to encode one logical qubit), the logical error rate turns into a "dial" that can be tuned by adding more physical qubits. Before that demonstration, adding more physical qubits increased rather than decreased the logical error rate, making achieving the scale of cryptographic relevance practically impossible.

Surface Code vs qLDPC

A 200× reduction in physical qubit requirements (from 20M to ~100K) occurred in just five years, driven almost entirely by improvements in error correction efficiency — not in physical hardware. qLDPC codes require fewer physical qubits per logical qubit than traditional surface codes, dramatically lowering the bar for cryptographic relevance.

PHYSICAL-TO-LOGICAL QUBIT OVERHEAD: SURFACE CODE VS QLDPC

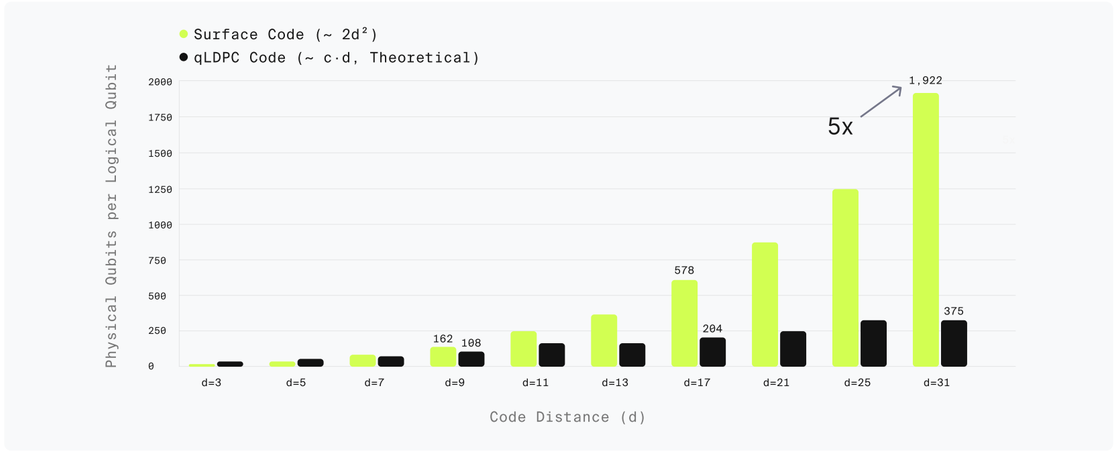


Figure 7: An asymptotic comparison of surface codes and qLDPC - two families of quantum error correcting codes

The term **"logical qubit"** does not imply that the error rate is zero. It means that the error rate has been reduced through error correction to a level where useful computation becomes possible. The acceptable error rate is not fixed by the hardware itself but by the **algorithm being executed.** Because each logical operation has a small probability of failure, errors accumulate as the circuit becomes deeper. As a result, the **longer the computation (the greater the circuit depth)**, the **lower the logical error rate per cycle must be** in order for the overall computation to succeed with high probability.

Key insight: Error suppression is exponential

Moreover, because error suppression is exponential, a relatively modest jump in the number of physical qubits results in a massive reduction of the logical error rate per cycle, as illustrated by the chart to the right. This means that once a system crosses the below-threshold point, every additional qubit compounds the reliability improvement dramatically.

Requirements for RSA-2048

LOGICAL QUBITS (LQC)	~1,000
LOGICAL GATE OPERATIONS (LOB)	~1 trillion (10^{12})
TARGET RUNTIME	~1 week (implies QOT ~1μs)
PHYSICAL QUBITS NEEDED	~100,000 (with qLDPC codes)

LOGICAL ERROR RATE COMPRESION (LOWER = BETTER)

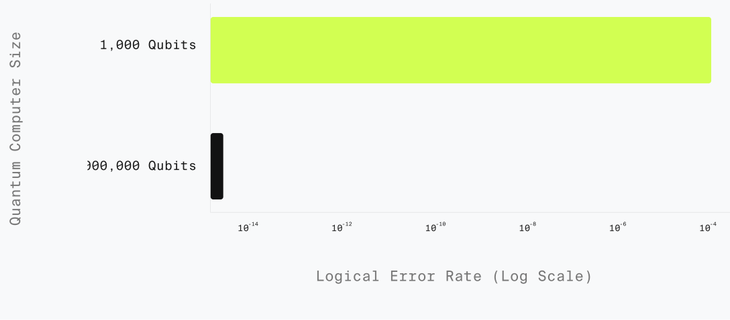


Figure 8: Illustration of below-threshold scalability – adding more qubits drives down logical error rates

SECTION ONE

Layer 3: System Integration

The layers above describe individual components; the system integration layer ties them together to enable continuous operation at scale. This includes the decoder, a classical computer that must:

- Measure or detect the outputs of the error correction inside the quantum machine (error “syndromes”)
- Decode those outputs, and issue corrections (“feed-forward”) in real time.

Decoding, measurement, and feed-forward in particular is handled by classical hardware or at the quantum-classical interface. It is a continuous operation that must maintain pace with the quantum system or errors will outrun the correction cycle. Moreover, the system must maintain coherence and stability not for microseconds but for hours or days. Critically, the system integration layer is where scaling penalties emerge most clearly: effects that are manageable at small scale (crosstalk, thermal load, control complexity) can become prohibitive as the system grows.

Thus, a good control layer is:

- **Efficient** - accurately performs measurement, decoding, and feed-forward with the minimum classical hardware
- **Reliable** - doesn’t introduce additional error that must also be corrected
- **Fast** - to ensure that the logical cycle time does not exceed the coherence constraint given by the physical layer

ONE “TRICK” OR LOGICAL CYCLE

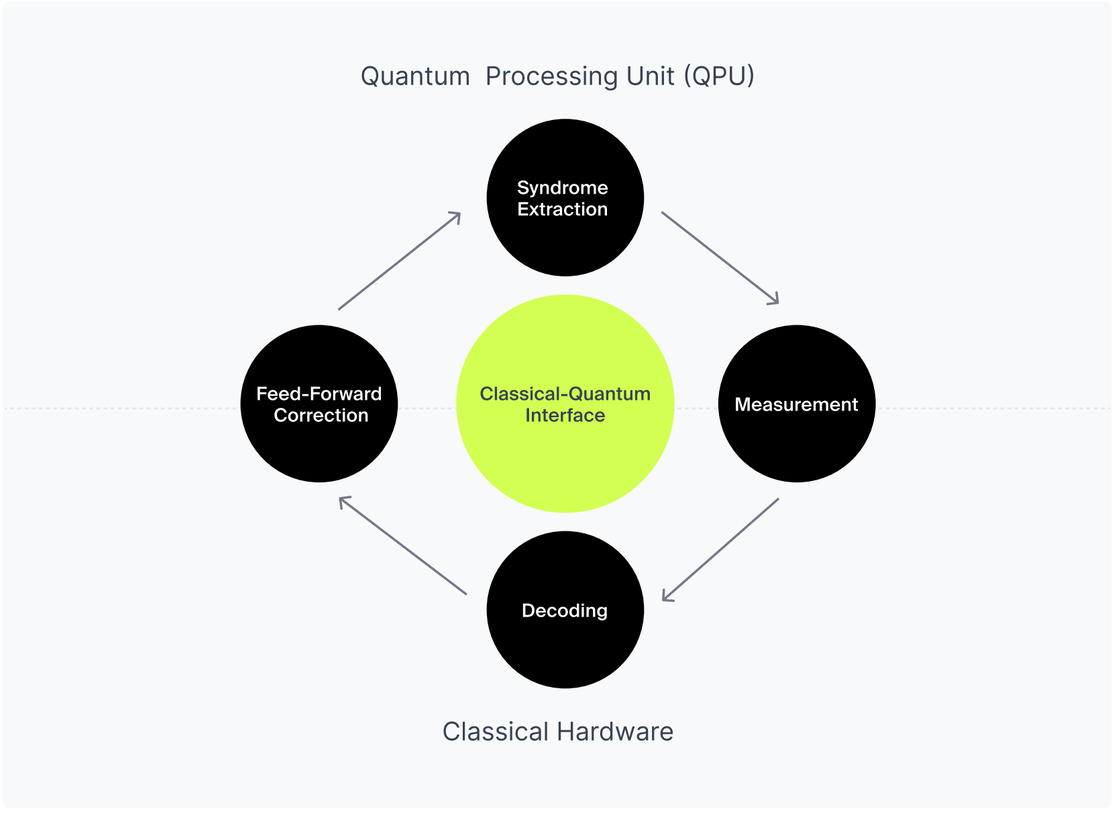


Figure 9: A diagram of the classical-quantum control interface during one logical cycle of a quantum operation

SECTION ONE

Layer 4: Algorithm Demand

The top of the stack defines the bar the lower levels must reach in order to achieve a useful result. For a CRQC, the target is running Shor's algorithm against cryptographic keys. This sets specific requirements: the number of logical qubits needed, the circuit depth, and the degree of parallelism the algorithm supports. Crucially, algorithmic optimizations lower the demand side of this equation: every improvement in how Shor's algorithm is implemented reduces the requirements that the hardware stack must meet. This is why the resource estimates in the table below have fallen so dramatically: better algorithms are pulling the target closer from above while better hardware pushes capability upward from below.

The algorithm choice and optimizations set the demand requirements, which must be matched with a sufficient "supply" of resources from the quantum computer, broken down into three composite metrics derived from a combination of factors across the quantum computing stack:

LQC Logical Qubit Capacity

The number of error-corrected logical qubits available simultaneously — the quantum equivalent of how much working memory the system has; or, equivalently, the "width" of the quantum circuit.

LOB Logical Operations Budget

The total number of reliable logical operations the system can execute before accumulated errors corrupt the computation; or, equivalently, the maximum "depth" of the quantum circuit.

QOT Quantum Operations Throughput

The speed at which logical operations execute, measured in operations per second — analogous to "clockspeed" on a classical processor. Determines whether a theoretically possible algorithm runs in a human-scale practical timeframe. Also called CLOPS [8].

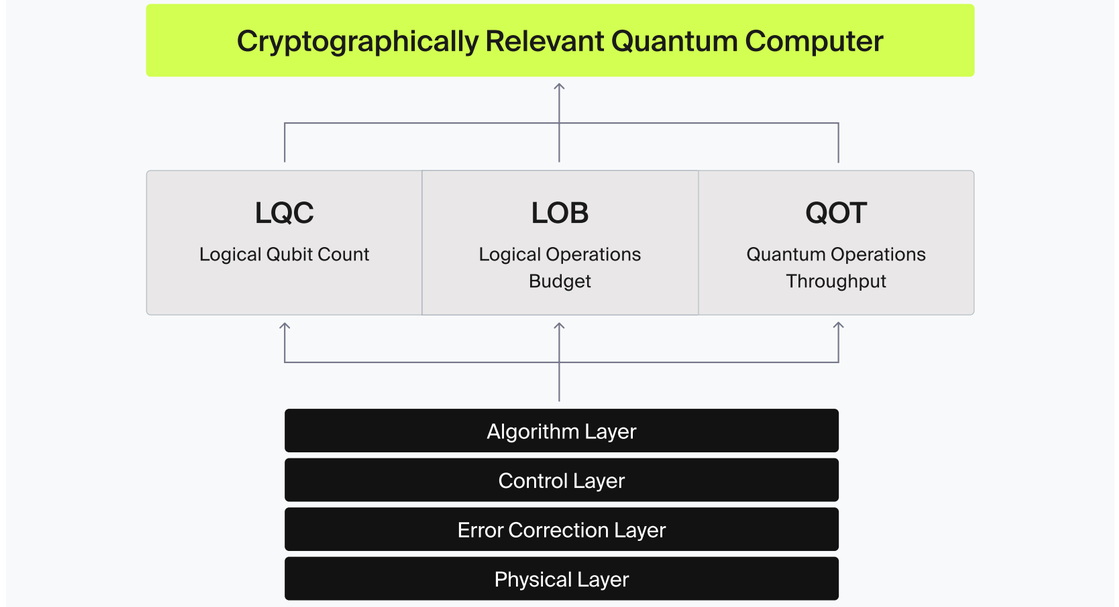


Figure 10: Distilling the quantum correcting stack into three metrics for measuring progress to utility scale hardware

SECTION ONE

Layer 4: Algorithm Demand (continued)

To put these metrics in context using Gidney 2025:

• Breaking RSA-2048 or ECDSA is estimated to require roughly 1,000 logical qubits (LQC = 1000). Today, the state of the art is a handful of logical qubits.

• The latest variants of Shor's algorithm require on the order of a trillion logical gate operations (LOB = 10^12). To date, no quantum computer has run more than a few thousand gate operations.

• A time target for RSA-2048 factoring is approximately one week. No quantum computer has yet demonstrated fault-tolerant logical operation at any meaningful duration required for such a computation, implying a QOT of one microsecond.

As Figure 10 shows, LQC and LOB have a geometric interpretation and can be combined into a single quantity sometimes called circuit “volume”.

Volume = LQC (width) × LOB (useable depth)

LCC = Volume × QOT

The choice of algorithm defines the threshold a quantum computer must reach. Variations of Shor's algorithm often exploit a time-space tradeoff: faster runtimes require greater LQC and larger LOB, while slower runtimes expand LOB further.

Even with sufficient LCC, a quantum computer is only cryptographically relevant if it operates on a human timescale. QOT is therefore a practical constraint, not just an implementation detail.

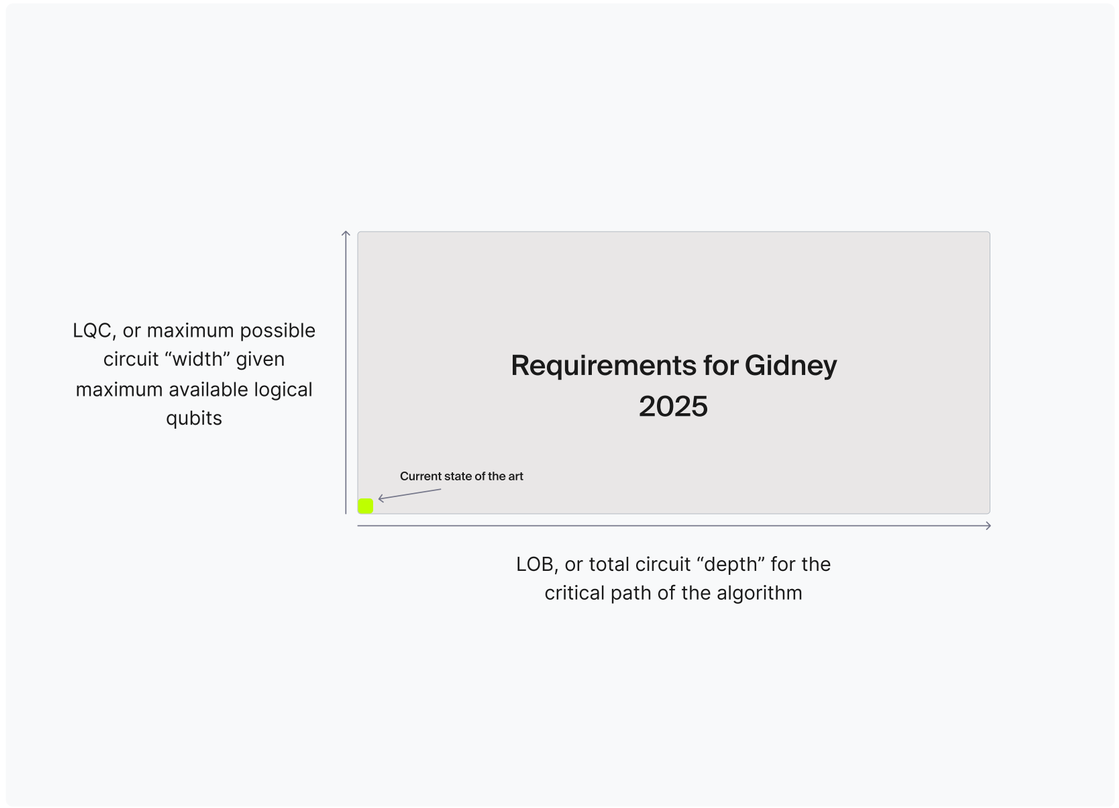


Figure 11: A simplified geometric interpretation of Shor’s algorithm requirements relative to current machines

SECTION ONE

Putting it All Together: Predicting Q-Day

Relying solely on physical qubit count to estimate progress towards Q-Day is insufficient. Today's quantum computers score far below any reasonable threshold of cryptographic relevance. But the predominant effort to develop a fault-tolerant quantum computer in recent years has not been focused on scaling the raw number of physical qubits. It's about creating the conditions (via better error correcting codes, control stack, and system integration) to enable scaling up the number of physical qubits trivially.

Thus, criticisms of quantum computing along the lines of "they can't factor 21 yet" are a red herring. Because once the foundation is solid on Layers 1–4, scaling up the physical qubits to factor cryptographically relevant numbers may potentially be the "easy" part. But while these scaling challenges are not fundamental barriers, they represent engineering unknowns that make extrapolating from today's small-scale demonstrations to fully fault-tolerant systems running thousands of qubits over days and weeks is genuinely uncertain.*

Fundamentally, the latest generation of quantum computers being designed today have the potential to run circuits of much greater volume and potentially reach the scale of cryptographic relevance by:

1. Fabrication and integration of an increasing number of physical qubits into the system.

2. Physical qubits that have lower baseline error rates, requiring less error correction, resulting in a higher **logical qubit capacity (LQC)**. See Appendix B for a detailed treatment of the various physical modalities/types of qubits.

3. More efficient quantum error correction (QEC) techniques to more powerfully and efficiently mitigate the negative impact of noise during the operation of a quantum computer that increase the **logical operations budget (LOB)**.

4. More robust and streamlined control planes to enable a more resilient quantum-classical interface that enables a higher **quantum operations throughput (QOT)**.

5. Better cryptanalytic algorithms specific to ECC as well as hybrid classical-quantum approaches that reduce the overall **logical compute capacity (LCC)** requirements to reach the threshold of cryptographic relevance.

Once someone has built a quantum computer with sufficient LCC to run a given algorithm, we've arrived at Q-Day.

* A note on scalability: the threshold theorem explicitly makes two critical assumptions: the error rate remains below threshold at scale, and the errors themselves are uncorrelated. However, the extent to which that's true is unknown. It may very well be that error rates increase with scale and that they are correlated. In that case, current resource estimates should be treated as lower bounds: the true qubit requirements for a CRQC may be higher than published figures assume

SECTION ONE

Estimating Q-Day: Three Approaches

To estimate exactly when that may occur, we take three different approaches:

1. Survey of Experts

Expert surveys of quantum computing researchers — tends to skew more pessimistic/uncertain, reflecting researchers' awareness of remaining engineering challenges.

2. Published Roadmaps

Publicly announced roadmaps from leading quantum computing companies across all major hardware modalities — tends to skew more optimistic.

3. Bottom-Up Analysis

A simplified model simulating progress for both "supply" (quantum computing capability in terms of LCC) and "demand" (algorithm parameters) — scenarios range from optimistic to pessimistic.

SECTION ONE

1. Survey of Experts

SURVEY OF EXPERTS

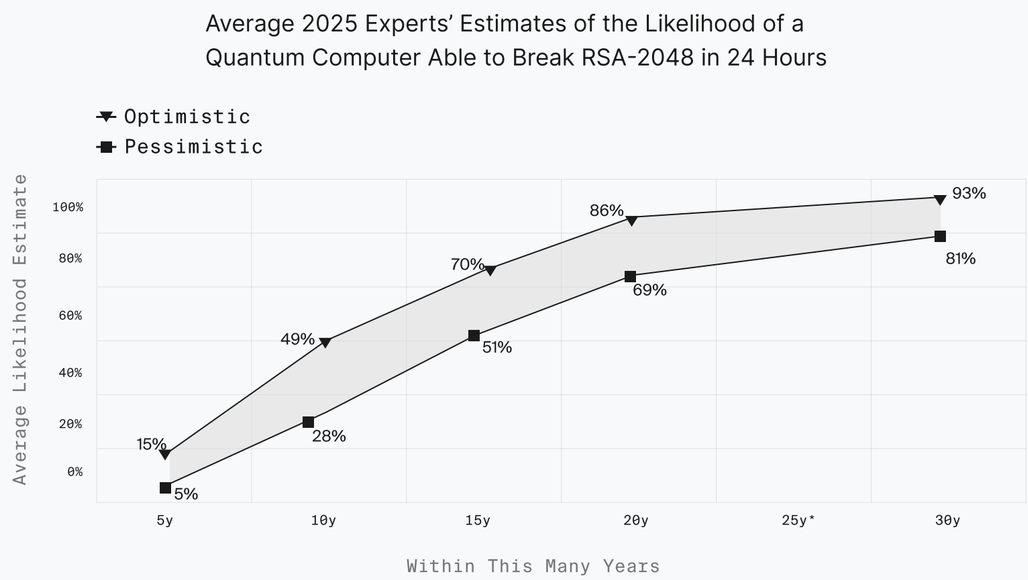


Figure 12: Survey of experts on the quantum computing timeline [8]

The **2025 Quantum Threat Timeline Report** by Michele Mosca and Marco Piani surveys leading experts in quantum computing to estimate when a **cryptographically relevant quantum computer (CRQC)** capable of breaking widely used public-key cryptography may emerge. The report gathers responses from researchers and industry leaders and analyzes their expectations about progress in hardware, error correction, and algorithmic improvements. Its central focus is the timeline for a quantum computer capable of running algorithms such as Shor's algorithm at a scale sufficient to break systems like RSA.

The survey results suggest that experts believe a CRQC is plausible within a few decades, though there remains significant uncertainty, with the report presenting probability estimates that such a machine could appear by roughly the **2030s–2040s**, with some respondents assigning non-negligible probability to earlier breakthroughs. Much of the uncertainty arises from open questions around scaling hardware, improving qubit fidelity, and implementing large-scale quantum error correction. Respondents also highlighted important milestones on the path to a CRQC, including demonstrating stable logical qubits, improving physical gate fidelities, and scaling systems to thousands or millions of qubits.

SECTION ONE

2. Published Roadmaps (& Physical Qubit Milestones)

PHYSICAL QUBITS VS RSA-2048 THRESHOLDS

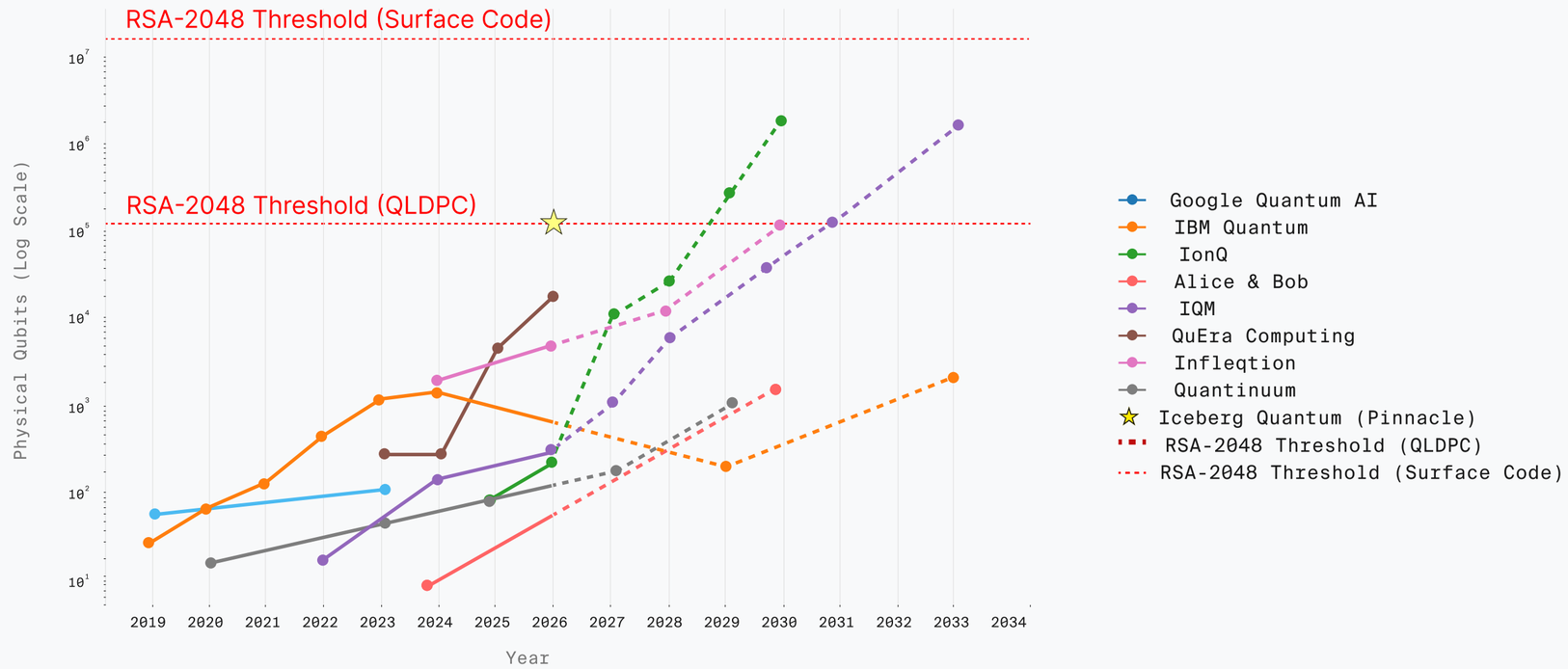


Figure 13: Road Map Chart

Published Company Roadmaps

The following table summarizes publicly announced roadmaps from the leading quantum computing companies across all major hardware modalities. This is probably the most optimistic estimate, as it comes from the companies building CRQCs themselves.

However, multiple companies project reaching the threshold of cryptographic relevance before the decade is out. And it should be noted that the latest resource estimates for elliptic curve algorithms (such as ECDSA over secp256k1 used in **Bitcoin**) are easier for a quantum computer to break than RSA, although significant engineering challenges remain to actually realize a real-time control stack that can accommodate the latest high-rate error correcting codes used in those constructions

Key observation

→ IBM's physical qubit count decreases post-2024 due to a focus on decreased error rates — fewer but more error-resistant qubits.

→ Companies targeting qLDPC codes project significantly lower physical qubit thresholds than those using surface codes.

→ ECC-256 (used in **Bitcoin**/**Ethereum**) has a lower resource threshold than RSA-2048, meaning Q-Day for blockchains may arrive earlier than RSA-focused estimates suggest.

SECTION ONE

3. Bottom-Up Analysis

Utilizing a simplified model to simulate progress for both the “supply” (quantum computing capability in terms of LCC) and “demand” (algorithm parameters), we can simulate various scenarios for how quantum computing scales. The full parameters for this model, assumptions, and definitions are detailed in Appendix E. Note that this is not a full resource estimate, but the parameters are based on (and the results are largely consistent with) Gidney’s 2025 resource estimate [6].

Parameter	2021 Historical	Baseline (2026)	Pessimistic → 2042	Moderate → 2033	Optimistic → 2030
Physical Qubit Count	1,000	3,000	+1.5× / year	+2× / year	+3× / year
Qubit Quality (2Q Gates)	99.5%	99.95%	+5% / year	+10% / year	+15% / year
Error Correction Efficiency	No below-threshold demonstrated	Surface code (x^2 overhead)	+2% / year	+5% / year	+8% / year
Control Overhead	N/A	80% LQC overhead; 1.5× circuit width; 3× depth penalty	+2% / year	+5% / year	+8% / year
Algorithm Requirements	~6T ops / 6,000 LQC min	~5T ops / 1,000 LQC	±0% reduction	−2.5% / year	−5% / year
Q-Day	—	—	**2042 (16 years)**	**2033 (7 years)**	**2030 (4 years)**

Based on known parameters, historical trends and scenario analysis with different future growth trajectories, we estimate that Q-Day is likely to occur within the next 4 (optimistic case) to 16 years (pessimistic case). Note that this model assumes no major breakthroughs, only relatively modest year-over-year improvement. If qubit quality were to suddenly drop by another order of magnitude, then it’s possible the world will face a Q-Day scenario even **before 2030**.[4]

[4] Note: This model assumes the threshold law for scaling; which states that correlated errors DO NOT increase with scale. In fact, in our model overhead goes down with scale to account for engineering improvements. This is a key assumption which should be taken into account when evaluating the model output.

SECTION ONE

3. Bottom-Up Analysis (continued)

SENSITIVITY ANALYSIS: QC COMPONENT IMPACT ON Q-DAY TIMELINE

Figure 14: Sensitivity analysis for the variables driving the Project Eleven Q-Day model

PROJECT ELEVEN Q-DAY FORECAST

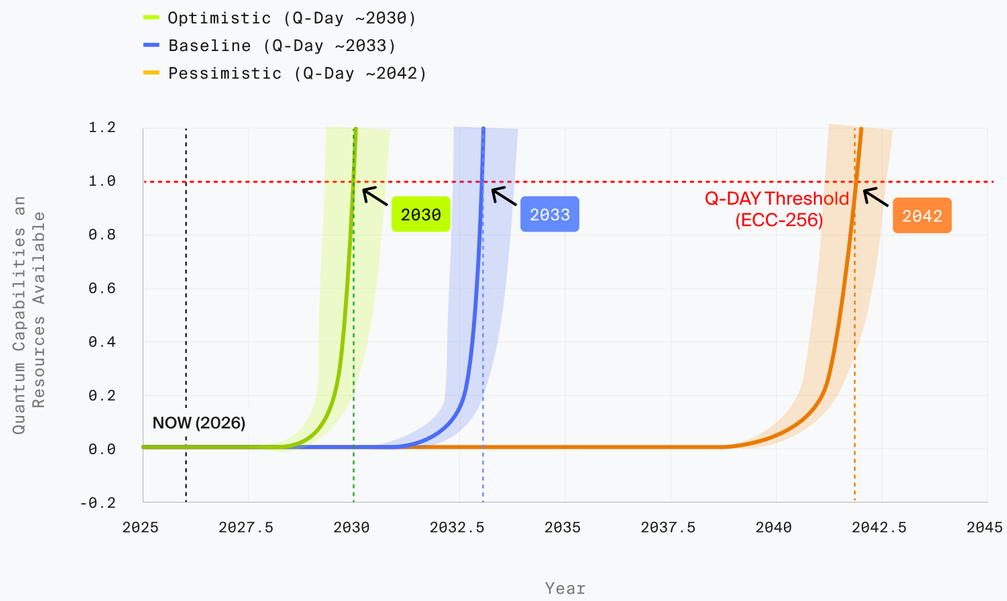


Figure 15: Q-Day model summary: exponential growth curves imply “nothing, then all at once” trajectories to Q-Day

In order to illuminate the relative weight of different factors in our model, below we present a sensitivity analysis based on our simplified four-layer quantum computing framework developed previously.

As we can see, in the simplified four-layer model, the dominant determinant of cryptographic relevance is scaling the physical layer: the number and quality of physical qubits. Improvements at the error-correction layer are the next most important, largely because they reduce physical-to-logical overhead. Algorithmic and control-layer improvements remain meaningful, and their relative importance depends on the exact architecture being used.

Note that this model is anchored to Gidney’s 2025 resource estimate, placing a width (LQC) requirement of 1000 logical qubits. As a result, achieving this width becomes the dominant bottleneck; and once logical qubits start to scale, the model rapidly converges on Q-Day.

SECTION ONE

Key Takeaways

- Using **three distinct methodologies** (expert assessment, quantum roadmaps, and our own analysis) we assess that **Q-Day is highly likely to occur in the next decade.**

- With below-threshold error correction demonstrated, the primary bottleneck to reaching cryptographic relevance is **logical circuit width (LQC)**, which itself is determined by physical qubit number, quality, physical-logical qubit ratio, and algorithmic overhead. Based on the best resource estimates/algorithm variants, we assess that **LQC is the dominant variable** affecting Q-Day timelines.

- Both expert surveys and our own model are anchored to algorithms targeting RSA-2048; a **major unknown factor is whether or not elliptic curve cryptography is as hard as breaking RSA. If not, Q-Day may arrive much earlier** than our model suggests.

While these projections offer a useful framework for thinking about progress, they remain highly uncertain for the reasons discussed below.

SECTION ONE

The Path Forward

Since the below-threshold demonstration by Google, quantum computing is no longer a challenge of science, but a challenge of engineering (albeit a significant one). The major open question is now whether the logical regime, demonstrated multiple times at small scales, can be scaled to cryptographic relevance. But the design space for how to address the various scaling bottlenecks is quite large. Moreover, the more quantum computers improve, the faster progress accelerates.

Progress toward a **cryptographically relevant quantum computer (CRQC)** is unlikely to be linear. Instead, advances will occur in discrete jumps as individual bottlenecks in the stack are removed. At any given time the system is constrained by its weakest component; once that constraint is lifted, the next bottleneck becomes dominant.

At the same time, improvements in different layers of the stack compound multiplicatively. Higher physical gate fidelity reduces the code distance required for fault tolerance, which lowers physical-to-logical overhead and increases the number of available logical qubits. Because these effects reinforce one another, solving a single bottleneck can dramatically expand the feasible computational volume.

Taken together, these dynamics mean that progress toward a CRQC can occur along several independent fronts simultaneously. Improvements in hardware, error correction, and algorithms all interact with one another, and advances in any one layer can dramatically expand the feasible computational regime. As a result, the trajectory toward cryptographic relevance is not defined by a single critical path, but by multiple avenues of parallel progress.

In this section

1. Multiple Physical Approaches

Superconducting, trapped-ion, neutral-atom, silicon-spin, and photonic modalities are all advancing in parallel — no single critical path to a CRQC.

2. Error Correction Compounds

Operating below threshold means each incremental improvement in code distance exponentially reduces logical error rates, amplifying gains across the entire stack.

3. Algorithms Lower the Bar

Circuit-level and arithmetic optimizations continually reduce the resource requirements for cryptanalysis — lowering the hardware threshold for cryptographic relevance.

SECTION ONE

1. Multiple Physical Approaches

Superconducting, trapped ions, neutral atoms, silicon-spin, and photonic-based modalities each have their own unique advantages and challenges. And the development/scaling of each approach is (to some extent) independent of the others.

As a result, quantum progress is not restricted to a single technology or “critical path”. In fact, a major motivation for developing alternatives to the superconducting regime (pioneered by companies like IBM, Rigetti, and Google) was the inherent challenge of scaling. This led to the development of trapped-ion and neutral atom-based approaches. Trapped ions, in particular, have the lowest baseline error rates of any modality, meaning the “workload” of the error correction stack can be somewhat lighter. Meanwhile, neutral atoms, also very stable, feature reconfigurable connectivity, making it easier to apply newer and more efficient error correction techniques. Both of these modalities also have their own control mechanisms, and both favor different error correction regimes.

Newer and more exotic modalities are pushing the boundaries of theoretical possibility even further. The silicon-spin modality takes advantage of existing CMOS fabrication techniques to ensure reliable fabrication of qubits at scale. Meanwhile, photonic-based systems have the potential to run at the speed of superconducting qubits, but at room temperature instead of inside a dilution-refrigeration system.

Each approach also has its challenges. But solving these challenges require distinct approaches, so the space for a potential breakthrough is quite large.

Moreover, these tech trees are not only progressing in parallel; they are increasingly reinforcing one another. Photonic technology, in addition to being a modality in its own right, is also being explored as a means of interconnecting distinct quantum systems, enabling modular heterogeneous architectures that combine the strengths of multiple physical modalities. Q-CTRL's recent work [12] illustrates how mixing different error-correcting code families and specialized processing units across modules can yield substantial efficiency gains, and DARPA has formalized this direction through its Heterogeneous Architectures for Quantum [HARQ] program. The practical consequence is that progress in any one modality may now compound into progress for the others, broadening the space of possible breakthroughs even further.

SECTION ONE

2. Error Correction Improvements Compound

PHYSICAL QUBITS REQUIRED FOR SHOR'S ALGORITHM

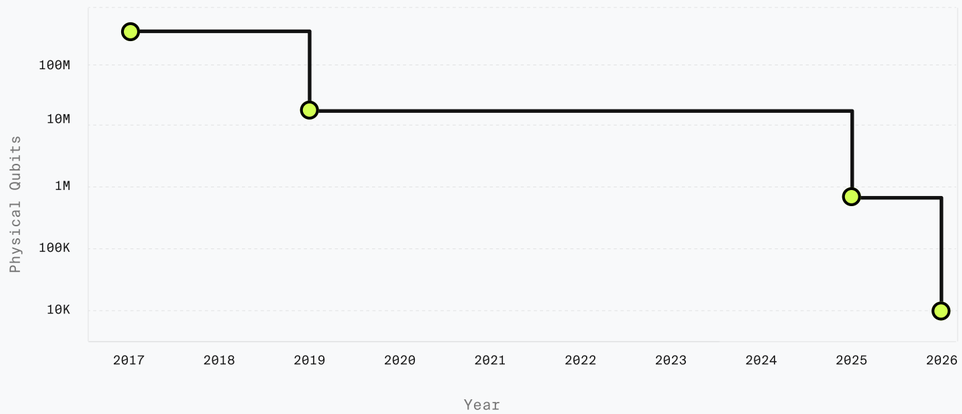


Figure 16: Summary of resource estimates for Shor's algorithm over the last ten years

Second, because reducing error rates is the fundamental challenge in scaling a cryptographically relevant quantum computer, even small improvements in physical fidelity or code efficiency can have massive downstream effects.

Operating below threshold means that doubling the code distance exponentially reduces logical error rates. But improving the underlying physical error rate also reduces the code distance needed to achieve target logical fidelity. And better error correcting codes that require less overhead have a compounding effect, because error correction is applied at every cycle of the computation. These effects multiply: better physical qubits enable smaller error-correcting codes AND smaller error correcting codes require fewer physical resources.

This feedback loop can massively reduce the cost to run Shor's algorithm.

This compounding is already evident in the trajectory of resource estimates for breaking RSA-2048 with Shor's algorithm: a 20× reduction from 20 million to approximately 1 million physical qubits occurred in just four years, driven almost entirely by improvements in error correction efficiency, not in physical hardware. The 2025 Gidney estimate assumes the same physical hardware parameters as 2021 (0.1% gate error rate, 1-microsecond surface code cycle time, nearest-neighbor superconducting grid), demonstrating that better algorithms and error correction techniques alone can collapse resource requirements dramatically. Further research [10] shows that the shift from surface codes to qLDPC codes can yield another order-of-magnitude reduction, bringing the total physical qubit count into a range that multiple hardware platforms are credibly targeting within the next several years.[5]

Subsequent research published in early 2026 reinforces this trajectory across multiple architectures and code families. In parallel to the Google work [2] cited throughout this report, several architecture-aware proposals using qLDPC and lifted-product codes converge on similar reductions: IonQ's "Walking Cat" design demonstrates 110 logical qubits in roughly 2,500 physical trapped-ion qubits with a streaming decoder, Q-CTRL's heterogeneous Q-NEXUS architecture reports a 138x physical-qubit reduction over surface-code baselines under detailed accounting [12], and a neutral-atom proposal from Caltech and Oratomic positions Shor's algorithm at cryptographically relevant scales using as few as 10,000 reconfigurable atomic qubits with all-to-all connectivity [3]. These works rely on different assumptions, but they describe a consistent picture: qLDPC codes and architecture-aware compilation are dragging the resource floor down across every modality, simultaneously.

Progress in classical machine learning is also now feeding back into the quantum stack itself. Real-time syndrome decoding has long been considered one of the practical bottlenecks for deploying qLDPC codes at scale, and recent work from Harvard demonstrates that a convolutional neural network decoder ("Cascade") can suppress logical error rates by more than an order of magnitude relative to existing decoders for qLDPC codes, while delivering three to five orders of magnitude higher throughput and meeting real-time latency budgets on several leading hardware platforms. [13] That a key decoder bottleneck appears to yield to a relatively standard deep learning architecture is a useful illustration of how AI infrastructure and the quantum computing stack can complement one another.

[5] Albeit at the cost of increased decoding complexity [11].

SECTION ONE

3. Algorithmic Optimizations Lower the Bar

Third, progress happens not just in hardware and error correction, but in algorithms themselves. There are multiple independent threads of algorithmic improvement, all of which lower the bar for cryptographic relevance:

- **Circuit-level optimizations**: Recent work targeting the Ed25519 curve exploits the isomorphism between Edwards and Weierstrass forms to reduce resource requirements by 75% on the key bottleneck metric and qubit requirements by 12% [14]. These optimizations also extend to NIST-standard prime fields, meaning the improvements apply broadly.

- **Arithmetic optimizations**: The introduction of approximate residue arithmetic reduced the number of logical qubits needed from ~6,000 to ~1,730 for RSA-2048 [6]. Subsequent work combined this with logical-layer improvements and improved code constructions to achieve a further 100× reduction in required resources [102].

- **Alternative factoring algorithms**: A 2023 proposal for a quantum factoring algorithm with reduced circuit size represents an asymptotic improvement over Shor's [15]. While the practical advantage over optimized Shor's remains unclear for current parameters, it represents an additional avenue for future improvement.

These algorithmic breakthroughs effectively lower the bar for what counts as "cryptographically relevant." While hardware capabilities are climbing, the target is simultaneously getting easier to hit. Progress can happen in both directions, and indeed, it has been.

SECTION ONE

Nothing, and Then All at Once

In 2019, Hartmut Neven, director of Google's Quantum AI lab, observed that improvements to Google's quantum processors were advancing at a doubly exponential rate — not just exponential like Moore's Law, but exponential on top of exponential. Under Moore's Law, computing power roughly doubles every two years. Under Neven's law, the growth rate itself accelerates: four improvement cycles yield not a 16× gain (as with simple exponential growth) but a 65,000× gain. Neven described the subjective experience of this trajectory: "It looks like nothing is happening, nothing is happening, and then whoops, suddenly you're in a different world."

For years, quantum computers may appear to make slow progress, factoring small numbers that classical computers can already handle. Then, as multiple breakthroughs converge (better physical fidelity crosses a threshold, higher-efficiency error correcting codes, an algorithmic insight cuts resource requirements), the gap suddenly closes. What seemed like a distant threat becomes imminent within months, not years.

Whether or not Neven's law holds precisely as a quantitative rule, or whether feedback loops from improving quantum computers result in truly non-linear growth, the qualitative pattern it describes matches the trajectory of quantum computing progress toward cryptographic relevance. Because improvements compound across multiple independent dimensions — hardware, error correction, and algorithms — progress will not be gradual and predictable. Instead, it will follow this "nothing-and-then-all-at-once" trajectory.

This sensitivity to initial conditions and assumption about rate of improvements makes precise timeline predictions extremely difficult. The majority of experts estimate Q-Day (the moment a CRQC is realized) as more likely than not in 10-12 years [9]. More aggressive projections, accounting for recent acceleration, suggest 3-5 years. Given the rapid progress and positive feedback loops leading to further acceleration, the difference between these estimates is not large in terms of technological capability; a few key breakthroughs separate them.

In this section

LOGICAL RESOURCES REQUIRED TO BREAK 256-BIT ECDLP

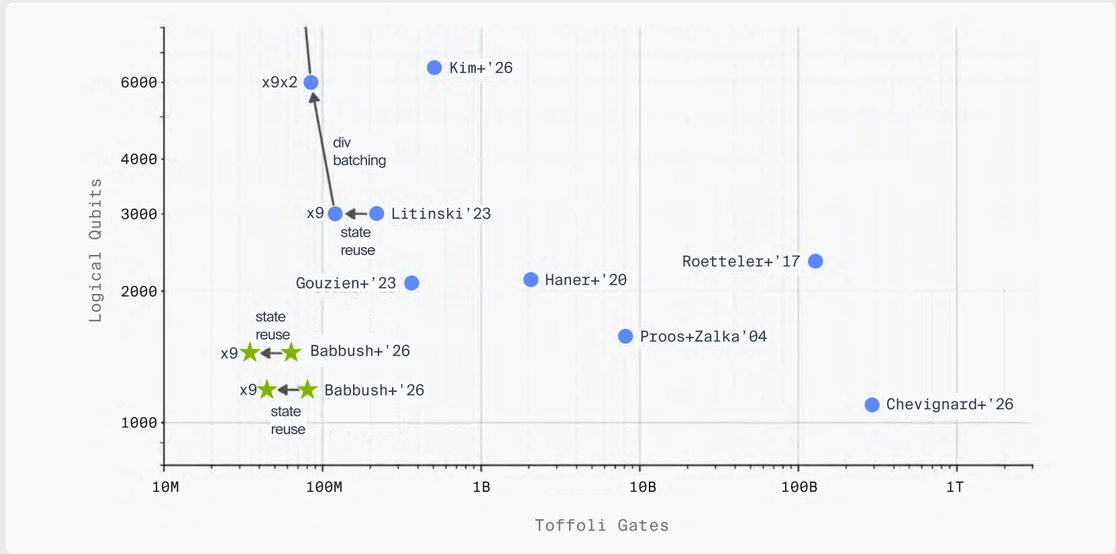


Figure 17: Comparison of logical quantum resources (number of logical qubits and Toffoli gates) required to break 256-bit ECDLP for the secp256k1 curve, as reported by various prior works.

SECTION ONE

1. Mosca’s Inequality: A Framework for Urgency

The proper way to reason about quantum risk under timeline uncertainty is Mosca’s inequality: a framework developed by cryptographer Michele Mosca [103] for evaluating when migration must begin. If the following inequality holds, migration should already be underway:

Migration Time + Data Shelf Life > Time to CRQC

- **Migration Time** - The total time required to transition the system to quantum-resistant cryptography.
- **Data Shelf Life** - How long the data (or in blockchain terms, the assets) must remain secure. For blockchain this term is effectively infinite: UTXOs never expire and assets must be protected indefinitely.
- **Time to CRQC** - The remaining time before a cryptographically relevant quantum computer arrives.

If this inequality holds (if the sum of the migration timeline and the data's required security horizon exceeds the time remaining before a CRQC exists), then migration should already be underway. Delay only widens the gap.

For blockchain systems, Mosca's inequality is even more unforgiving. The "data shelf life" term is effectively infinite: blockchain addresses hold economic value indefinitely, UTXOs never expire, and the public keys exposed in on-chain transactions persist permanently on an immutable ledger. Bitcoin (and other digital assets) have the property that, definitionally, shelf life should be infinite - a user should always have the ability to transact with a valid key. This means the inequality collapses to an even simpler condition:

If Migration Time > Time to CRQC, the system is already compromised in expectation.

Since blockchain data is public and permanently recorded, there is no separate “harvesting” step required; the adversary’s harvest is the blockchain itself. The only question is whether the migration can be completed before a CRQC arrives to exploit it.

Even optimistic blockchain migration timelines extend 4–6 years from initiation, with baseline estimates of 7–13 years. Against a CRQC arrival window of 2030–2033, Mosca’s inequality is already violated under nearly all reasonable assumptions.

This is why organizations from NIST to major browser vendors are already deploying post-quantum cryptography, despite the most optimistic quantum timelines being years away. The blockchain space must adopt the same precautionary stance.

For traditional internet infrastructure, the HNDL ("Harvest Now, Decrypt Later") threat provides the urgency: an adversary can harvest encrypted communications today and decrypt them retroactively once a CRQC arrives. This is why CISA, NSA, and NIST have jointly recommended immediate migration initiation regardless of timeline uncertainty. For data with confidentiality requirements extending beyond 2030, migration is already urgent.

MOSCA’S INEQUALITY APPLIED TO BLOCKCHAIN PQC MIGRATION

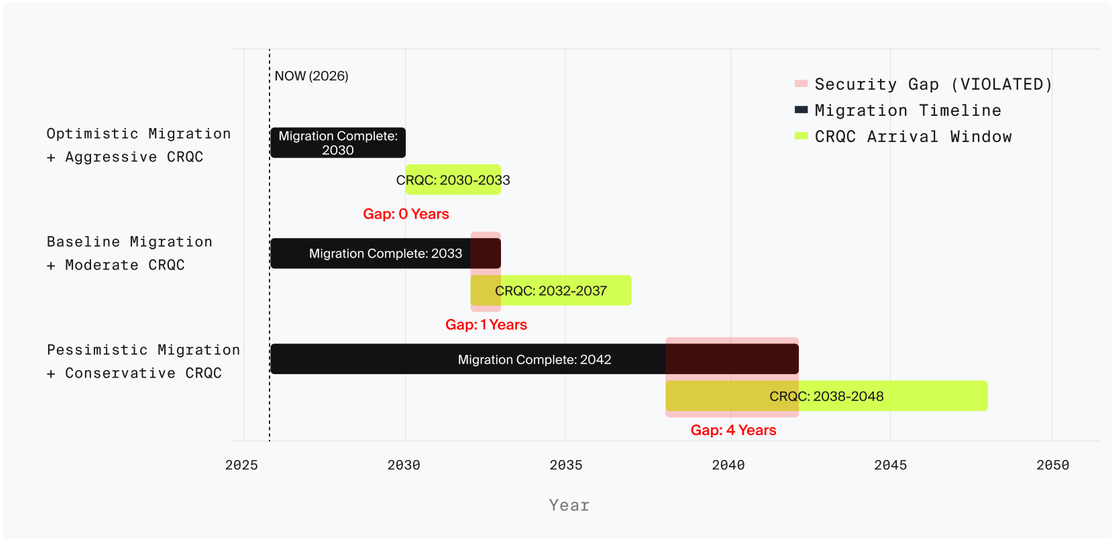


Figure 18: Illustration of Mosca’s Inequality: current migration timelines imply overlap with Q-Day

SECTION ONE

2. Timeline Uncertainty Demands Worst-Case Planning

Recent breakthroughs have drastically reduced the resource estimates for breaking 256-bit Elliptic Curve Cryptography, proving that the physical-to-logical qubit overhead is collapsing rapidly. On superconducting hardware, Google provided a zero-knowledge proof of a circuit to solve the ECDLP can be solved using between 1,200 and 1,450 logical qubits and under 90 million operations. Assuming physical error rates of 0.1%, this requires fewer than 500,000 physical qubits and takes just 18 to 23 minutes to execute. Meanwhile, neutral-atom architectures leveraging qLDPC codes can execute Shor's algorithm for ECC-256 using approximately 10,000 to 22,000 physical qubits in as little as 5 days.

This compressed timeline introduces a critical distinction in the threat landscape based on the physical clock speeds of different quantum modalities. "Fast-clock" architectures, such as superconducting or photonic systems operating with 1-microsecond cycles, can derive a private key from an exposed public key in roughly 9 minutes. This speed enables devastating "on-spend" attacks, allowing adversaries to hijack active transactions directly from the mempool before they are recorded on the blockchain. Conversely, "slow-clock" systems like neutral atoms and ion traps operate with millisecond cycle times and are restricted to "at-rest" attacks. These slower systems will instead target dormant wallets and long-lived smart contracts where the public key has been exposed for extended periods.

"Finally, the arrival of these capabilities will likely follow a "nothing, and then all at once" trajectory. Because the industry is shifting toward hiding the specific blueprints for advanced quantum attacks, the final milestones toward a CRQC will be obscured from public view — meaning Q-Day could arrive with virtually no warning."

Highlighting this shift, researchers are now actively withholding the exact mechanics of their most advanced cryptanalytic circuits to prevent malicious use, operating under coordinated vulnerability disclosure principles. To substantiate their latest resource estimates without arming adversaries, researchers utilized zero-knowledge proofs to cryptographically verify their claims.

These topics are the focus of the next section of this report.

02

Blockchain Vulnerabilities to a CRQC

An analysis of how a cryptographically relevant quantum computer threatens elliptic curve digital signatures, the exposure profile of major blockchain ecosystems, the state of post-quantum cryptography, and the path to migration.

SECTION TWO

Elliptic Curve Digital Signatures

The digital asset industry holds over $3 trillion in aggregate value, and virtually all of it is secured by the same class of cryptographic primitive: elliptic curve digital signatures. **Bitcoin** uses ECDSA over secp256k1. **Ethereum** uses ECDSA over secp256k1 for user accounts and BLS over BLS12-381 for consensus. **Solana**, Sui, Aptos, Near, and Stellar use EdDSA over Ed25519. Stablecoins, bridges, oracles, governance contracts, and custody systems all inherit the signature scheme of their host chain.

The quantum threat to digital assets is not a collection of independent, chain-specific problems. It is a single structural vulnerability — dependence on the hardness of the elliptic curve **discrete logarithm problem** — replicated across the entire ecosystem.

A CRQC running **Shor's algorithm** reduces this problem from computationally infeasible to efficiently solvable. The result is private key recovery from any known public key. The attacker produces valid signatures indistinguishable from the legitimate owner's, because mathematically, they hold the same key.

Two Properties That Make Blockchains Uniquely Vulnerable

1. Public Ledgers

Public keys, signatures, and transaction histories are permanently recorded and freely accessible. There is no "air gap" between cryptographic material and an attacker. The data a quantum attacker needs is already collected — it is the blockchain itself.

2. Bearer Instruments

Authorization is possession of a signing key, and signatures are final. There is no fraud department, no chargeback, no identity verification layer that can distinguish a legitimate owner from a quantum attacker holding the same private key. Once a forged signature is accepted by consensus, the transfer is irreversible.

"The quantum threat to digital assets is not primarily about future risk. It is about present-tense exposure to a future capability."

SECTION TWO

Elliptic Curve Digital Signatures (continued)

Every major blockchain uses some variant of elliptic curve cryptography (ECC) for transaction authorization. The underlying assumption is the same: given a public key (a point on an elliptic curve), it is computationally infeasible to derive the corresponding private key. **Shor's algorithm**, running on a sufficiently large quantum computer, solves this problem in polynomial time, breaking all elliptic curve schemes.

A common misconception is that hash functions like SHA-256 or Keccak-256 provide meaningful protection against quantum attacks. They do not, in the way that matters. **Grover's algorithm** provides only a quadratic speedup against hash preimages, reducing 256-bit security to an effective 128 bits — far beyond practical attack. However, hash functions are not the vulnerability. The vulnerability is the signature scheme itself: the elliptic curve operation that binds a public key to a private key. If an attacker can recover the private key, they can sign any transaction they want. No amount of hash strength changes that.

Some **Bitcoin** addresses do benefit from hash protection where the public key has never been revealed on-chain. But this is a micro-level protection that does nothing for the macro. If a quantum adversary drains the largest exposed wallets — early miner coins, exchange reserves, or any address that has ever broadcast a transaction — the systemic damage could collapse the value of the entire network.

Common Misconceptions

Multisig does not help
If each signer uses ECDSA or EdDSA, a quantum attacker simply recovers each signer's private key independently. A 3-of-5 multisig with five quantum-vulnerable keys requires three Shor executions instead of one — a linear increase in effort, not an exponential one. The security of a multisig is capped by the weakest underlying signature scheme.

Hardware wallets do not help
A hardware wallet secures how a private key is stored and used. A quantum attacker does not need access to the device. They need only the public key, which is already on-chain. The hardware wallet is defending against the wrong threat model entirely.

"The exposure varies by chain depending on when and how public keys become visible."

The following sections examine the specific vulnerability profiles of major blockchain ecosystems.

In this section

SECTION TWO

Bitcoin Exposure

Bitcoin's UTXO model provides a degree of quantum protection that is real but frequently overstated. When a user receives **Bitcoin** to a P2PKH or P2WPKH address, only a hash of the public key appears on-chain. The public key itself is not revealed until the owner spends from that address. For outputs that have never been spent, and whose addresses have never been reused, an attacker would need to reverse a hash function to obtain the public key — which **Grover's algorithm** does not make practical [16].

The protection fails under a set of conditions that cover a significant fraction of **Bitcoin**'s supply. P2PK and P2MS outputs embed the full public key directly in the locking script — these include the majority of early **Bitcoin** coinbase transactions from 2009 to 2011. P2TR (Taproot) outputs encode an x-only public key in the address itself, making every Taproot output quantum-vulnerable from the moment it receives funds [16]. Address reuse after spending is a third vector: when a user spends from a P2PKH or P2WPKH address, the public key is revealed in the spending transaction; any new UTXOs are then secured by an exposed key [16, 17].

In a standard BIP-32 HD wallet, exposing one leaf address does not compromise sibling addresses, because each child key is derived independently [17]. However, any individual address whose public key has been revealed — whether through spending, address type, or reuse — is independently vulnerable to quantum key recovery.

QUANTUM VULNERABLE **BITCOIN** - PROJECT ELEVEN

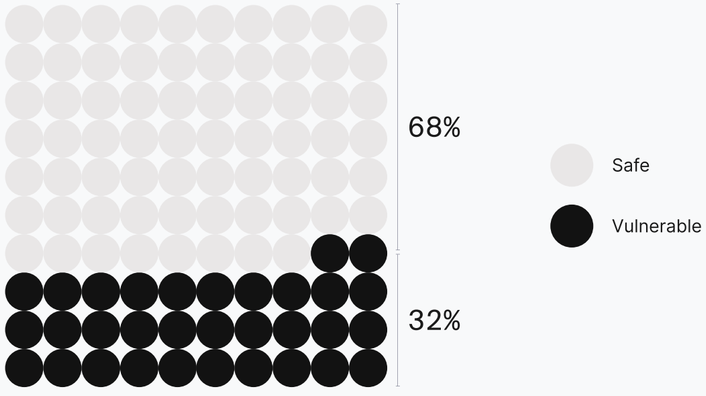


Figure 19

Project Eleven's **Bitcoin** Risq List [18] provides a continuously updated, address-level view of quantum-vulnerable **Bitcoin**, categorizing every quantum-vulnerable address by the specific mechanism of exposure. As of February 2026, approximately 6.9 million BTC — roughly 33% of the total circulating supply — sits in quantum-vulnerable addresses.

QUANTUM-VULNERABLE **BITCOIN** BY EXPOSURE MECHANISM

Exposure Mechanism	BTC Exposed	% Total	
Address Reuse	**4,994,044**	**72.3%**	
P2WPKH (SegWit) reuse	1,801,410	26.1%	Public key via prior spend
P2SH (Script Hash) reuse	1,299,221	18.8%	Public key via prior spend
P2PKH reuse	1,202,064	17.4%	Public key via prior spend
P2WSH (SegWit Script) reuse	691,349	10.0%	Public key via prior spend
Script Type	**1,915,211**	**27.7%**	
P2PK (Pay-to-Public-Key)	1,716,814	24.8%	Public key embedded in output
P2TR (Taproot)	198,106	2.9%	x-only public key in address
P2MS (Bare Multisig)	291	<0.01%	Public keys embedded in output
Total	**6,909,255 BTC**	**100%**	

Source: Project Eleven **Bitcoin** Risq List [18], block height 936,882 (February 2026). Total across 13.9 million addresses.

SECTION TWO

Ethereum Exposure

Ethereum's account model creates broader quantum exposure than **Bitcoin**'s UTXO model. When an externally owned account (EOA) sends a transaction, the sender's public key is recoverable from the ECDSA signature values (v, r, s) included in that transaction. **Bitcoin** can keep most value sitting behind fresh addresses; **Ethereum** ties value to a long-lived account, so the first outbound transaction from an address reveals the public key and that exposure persists for as long as the account remains relevant [19]. Analysis of the **Ethereum** blockchain has found that over 65% of all Ether is held in quantum-exposed addresses [20].

The exposure extends beyond user wallets. **Ethereum**'s Proof-of-Stake consensus layer depends on BLS signatures (BLS12-381), which are equally vulnerable to **Shor's algorithm** [21]. Every validator's BLS public key is published at the time of the 32 ETH deposit and remains visible in beacon chain state. A quantum attacker who recovers validator private keys could forge attestations, destabilize consensus, and trigger mass slashing. KZG commitments introduced with EIP-4844 (proto-danksharding) rely on elliptic curve pairing assumptions, adding a third quantum-vulnerable primitive to the protocol [21].

Smart contract governance introduces a distinct concentration of risk. Many of the most critical contracts in DeFi (lending protocols, DEXs, bridges, treasuries) are controlled by admin keys or small multisig wallets held by a limited set of signers. These admin keys can pause contracts, upgrade logic, change parameters, and move funds. If those keys use ECDSA and have ever signed a transaction, their public keys are recoverable. A quantum attacker does not need to find a bug in the contract code; they need only recover an admin key and call the contract's own privileged functions [19]. The damage from a single compromised admin key can far exceed the damage from draining any individual user wallet.

QUANTUM VULNERABLE **ETHEREUM** - DELOITTE

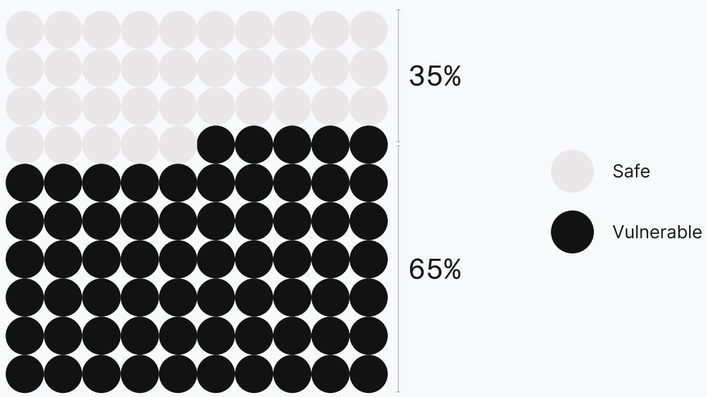


Figure 20

SECTION TWO

Stablecoins Exposure

Stablecoins warrant separate treatment because they combine high systemic importance with a concentrated, privilege-key-dependent architecture that creates a distinct quantum attack profile.

The stablecoin market exceeded $300 billion in total capitalization by early 2026, with USDT and USDC together accounting for over 80% of the market [22]. Monthly on-chain transaction volume reached $1 trillion by September 2025 [23]. The passage of the GENIUS Act in 2025 established a U.S. federal regulatory framework for payment stablecoins, and integration by major payment networks has extended stablecoin rails into traditional financial infrastructure [22].

The quantum attack surface of a treasury-backed stablecoin is not only the individual user's wallet. It is the contract's admin keys. A quantum adversary cannot touch the actual dollars held in reserve — instead, the attack surface is purely on the liability side of the ledger: the on-chain representation of who owns what, and the privileged roles that govern it. But that is more than enough to cause chaos.

Major stablecoins are implemented as upgradeable proxy contracts where a small number of privileged addresses control the entire system [24]. The typical role hierarchy includes: the proxy admin (able to replace the contract's entire logic), the owner (able to reassign other admin roles), a master minter role (able to authorize minters and set minting ceilings), a blacklister (able to freeze any account), and a pauser (able to halt all transfers globally) [26]. Providers most likely keep the highest-privilege roles behind multisig wallets, but nonetheless rely on ECDSA keys [26].

"This is a qualitatively different risk profile from attacking a base-layer protocol. In Bitcoin, a quantum attacker drains individual UTXOs, bounded by the exposed balance per address. In a stablecoin, a quantum attacker with admin key access can compromise the entire currency."

Attack Scenarios

Compromised minting authority
Allows the attacker to create unbacked tokens, collapsing the peg.

Compromised blacklister key
Enables selective freezing of exchange hot wallets or DeFi vaults.

Compromised proxy admin
Allows the attacker to deploy new contract logic, potentially rewriting balances across the entire token supply [26].

Blast Radius

The blast radius extends to every protocol that holds or prices against that stablecoin: DeFi lending pools, DEX liquidity, cross-chain bridges, and the traditional financial institutions that have integrated stablecoin settlement [25].

SECTION TWO

Other Networks

The vulnerability pattern described above is not unique to **Bitcoin** and **Ethereum**. Every major blockchain that relies on elliptic curve signatures shares the same fundamental exposure.

EdDSA chains — including **Solana**, Sui, Aptos, Near, and Stellar — use EdDSA with RFC-8032 key derivation, which computes the signing scalar from a seed via a hash function. A CRQC running **Shor's algorithm** recovers the signing scalar from the public key, but cannot reverse the hash to obtain the underlying seed. This structural property has significant implications for migration that we examine in detail later in this report.

ECDSA chains generally lack this property. ECDSA chains do not define a canonical function from seed to private scalar, and wallets typically sample a random scalar directly. The signing scalar is the private key, meaning **Shor's algorithm** recovers it completely with no hash-protected layer underneath.

Layer 2 networks, rollups, and application-specific chains inherit the quantum vulnerability of their parent chain's signature scheme and add additional exposure through their own sequencer keys, fraud-proof signer sets, and bridge contracts.

The pattern is consistent across the ecosystem: every layer of the digital asset stack, from Layer 1 consensus to Layer 2 rollups to the bridges and oracles connecting them, depends on the same class of quantum-vulnerable cryptographic primitive. The only path to securing any of it is migrating to cryptography that a quantum computer cannot break. That is the subject of the next section.

"The only path to securing any of it is migrating to cryptography that a quantum computer cannot break."

QUANTUM VULNERABLE **SOLANA** - PROJECT ELEVEN

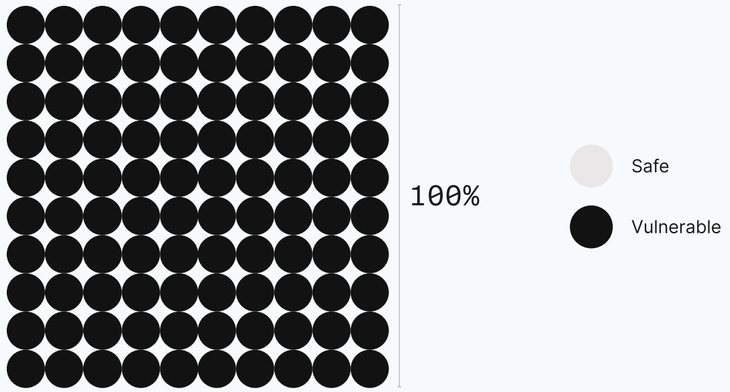


Figure 21: Solana exposes an X-only public key for addresses rendering all Solana quantum vulnerable

BTC AT RISK OVER TIME

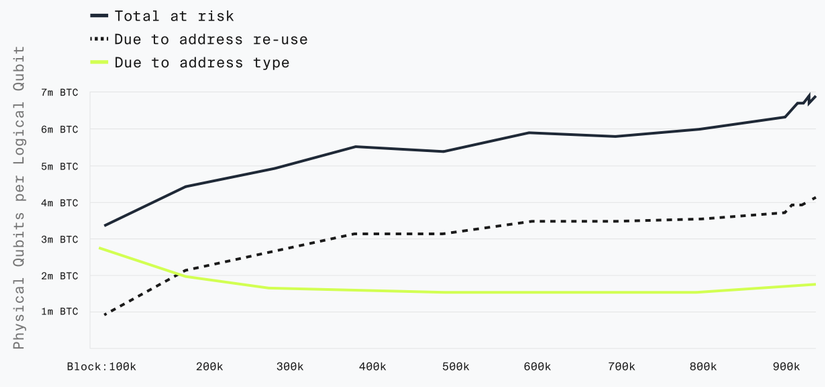


Figure 22: Source - Project Eleven **Bitcoin** Risq List

SECTION TWO

Post-Quantum Cryptography

Post-quantum cryptography (PQC) is the set of public-key encryption and signature algorithms believed to remain secure even if an attacker has access to a large, fault-tolerant quantum computer. PQC is no longer hypothetical research; the first PQC algorithms are standardized, shipping in widely used software, and already protecting significant fractions of real Internet traffic.

While much of the digital asset industry treats quantum risk as a future concern, the rest of the technology ecosystem is already deploying post-quantum protections at scale. PQC is already deployed at scale across the Internet. The gap between the Internet's pace of PQC adoption and the blockchain industry's pace of PQC adoption is not a gap in awareness. It is a gap in urgency.

Generally speaking, there are two different migration problems to address when talking about a PQC transition: key establishment (key exchange) for protocols like TLS and SSH, where "harvest now, decrypt later" (HNDL) risk shows up; and digital signatures for authentication and authorization, which Project Eleven calls a "harvest now, forge later" (HNFL) attack. If a CRQC can derive private keys from public keys for elliptic curve signatures, then it can forge signatures — especially relevant where signatures are meant to be long-lived and directly protect real value, which is precisely what blockchains do.

"Key establishment migration explains why the Internet moved quickly on PQ key exchange while signature migration — the part that matters most to digital assets — has yet to see wide adoption."

The Critical Asymmetry

The distinction matters because key establishment and digital signatures have fundamentally different migration properties.

Key Establishment (HNDL)
Can be upgraded transparently — update the client and server software, and users never notice. The Internet has already moved quickly on PQ key exchange.

Digital Signatures (HNFL)
Sit inside trust systems, transaction formats, and consensus rules that are difficult to change. In the case of blockchains, they require broad coordination across an entire ecosystem. This is the part the digital asset industry has barely started.

SECTION TWO

NIST PQC Algorithm Comparison

In August 2024, NIST published three finalized PQC standards.

ML-KEM (FIPS 203) [27]

The NIST-standard key encapsulation mechanism (KEM) for establishing shared symmetric keys over an insecure channel. Derived from CRYSTALS-Kyber, designed to replace RSA key transport and ECDH in protocols like TLS and SSH. Wire size: ~800–1,568 bytes for ciphertext.

ML-DSA (FIPS 204) [28]

The NIST-standard lattice-based digital signature scheme, derived from CRYSTALS-Dilithium. The primary standard intended to replace ECDSA and EdDSA for general-purpose signing. Signature size: 2,420–4,627 bytes — 38 to 72 times larger than a 64-byte Ed25519 or Schnorr signature.

SLH-DSA (FIPS 205) [29]

The NIST-standard stateless hash-based signature scheme, derived from SPHINCS+. Viewed as the most conservative option because its security relies on the well-understood properties of hash functions. The trade-off is size: signatures range from 7,856 to 49,856 bytes depending on variant and security level.

Additional Standards in Progress

FN-DSA (FIPS 206)

Based on FALCON, FN-DSA offers smaller signatures and public keys than ML-DSA — a meaningful advantage for bandwidth- and storage-constrained systems. The trade-off is implementation complexity: FALCON-style schemes require floating-point arithmetic during signing, raising engineering cost and the risk of subtle bugs across platforms.

HQC

Selected by NIST in March 2025, HQC is a second KEM intended to diversify underlying hardness assumptions. Unlike ML-KEM, it is not lattice-based, reducing ecosystem dependence on a single mathematical family. The trade-off is performance: HQC has larger messages and generally higher computational cost, so it is unlikely to replace ML-KEM as the default for high-volume protocols.

Practical Takeaway

ML-KEM solves key establishment with manageable wire sizes, around 1 KB for the commonly deployed ML-KEM-768 ciphertext. ML-DSA is the general-purpose signature standard, but signatures are roughly 2.4-4.6 KB, about 38-72x larger than a 64-byte Ed25519 or Schnorr signature. SLH-DSA offers a conservative hash-based option, but signature sizes are measured in tens of kilobytes. For systems where every byte is metered, priced, or propagated across a global network, especially blockchains, these numbers are the central engineering constraint.

ML-KEM (FIPS 203): KEY SIZE PARAMETERS BY SECURITY LEVEL

Parameter Set	Security Category	Public Key (bytes)	Private Key (bytes)	Ciphertext (bytes)
ML-KEM-512	1	800	1,632	768
ML-KEM-768	3	1,184	2,400	1,088
ML-KEM-1024	5	1,568	3,168	1,568

ML-DSA (FIPS 204): KEY SIZE PARAMETERS BY SECURITY LEVEL

Parameter Set	Security Category	Public Key (bytes)	Private Key (bytes)	Signature (bytes)
ML-DSA-44	2	1,312	2,560	2,420
ML-DSA-65	3	1,952	4,032	3,309
ML-DSA-87	5	2,592	4,896	4,627

SECTION TWO

Existing Deployments

PQC is no longer only in the standardization phase. It is being shipped into production systems at scale. The rollout pattern is consistent across the industry: deploy PQ key exchange first (often as a hybrid with classical ECDHE), then tackle signatures more slowly because PKI and code-signing ecosystems take longer to migrate. This pattern should be instructive for anyone planning a blockchain PQC transition: the easy part is done, and even the easy part took years.

Internet & Browsers (TLS)

Hybrid key exchange in TLS 1.3 is being standardized in the IETF TLS working group [30], including a general construction for combining classical and post-quantum key exchange so that the session remains secure if at least one component holds. In practice, this means combining X25519 with ML-KEM-768 into a single key agreement. Cloudflare enabled post-quantum hybrid key agreement across its network by default in October 2022 [31], and by December 2025, Cloudflare reported that 52% of human web traffic was post-quantum encrypted, nearly doubling from 29% at the start of the year [32]. On the client side, Google shipped support for ML-KEM hybrid post-quantum TLS in Chrome 131 [33]. Apple enabled hybrid post-quantum TLS support in iOS 26 [34], producing an immediate and measurable jump in PQ-encrypted traffic from mobile devices: within four days of the iOS 26 release, PQ support from iOS devices jumped from under 2% to 11%, and exceeded 25% by early December [32].

SSH (Key Establishment)

OpenSSH has offered post-quantum key agreement by default since OpenSSH 9.0 (April 2022). OpenSSH 10.0 (April 2025) adopted an ML-KEM-based hybrid (mlkem768×25519-sha256) as the new default [35]. For any organization running current SSH infrastructure, post-quantum key agreement is already the baseline, with no configuration changes required.

Cloud Platforms & OS Crypto

AWS and Google Cloud have both announced support for post-quantum KEMs in their KMS products [36, 37]. Apple has added PQC to CryptoKit as of iOS and macOS 26, including support for both ML-DSA and ML-KEM [34]. These are not experimental previews; they are production APIs that signal the expectation that downstream applications will begin integrating PQC into their own cryptographic workflows.

The State of Play

The Internet's key exchange infrastructure is already well into its post-quantum transition. Signatures are the unsolved problem. And signatures are exactly what blockchains depend on.

52%

of human web traffic post-quantum encrypted

(Cloudflare, Dec 2025)

SECTION TWO

Impact on Blockchains

Blockchains are disproportionately exposed to the signature side of the PQC transition. Every transaction is an authorization, and authorization is signature verification. That means the dominant costs are not KEMs or handshake overhead, but signature size, verification cost, mempool and block propagation, and block capacity. The key exchange migration already underway across the Internet does not address this risk. Blockchains need signature upgrades, which carry fundamentally different costs and coordination requirements.

The Wire Size Constraint

The first hard constraint is wire size. A typical Ed25519 or Schnorr signature is 64 bytes. An ML-DSA-44 signature is 2,420 bytes, roughly 38× larger. An ML-DSA-65 signature is 3,309 bytes. Moving from compact elliptic curve signatures to multi-kilobyte post-quantum signatures increases transaction size, bandwidth consumption, and storage requirements. In any system where transaction size affects fees, propagation latency, block capacity, and validator hardware requirements, that increase propagates into every layer of the economic and operational model.

The Validation Cost Constraint

The second hard constraint is validation cost. Benchmarks [38] show that ML-DSA signing is roughly 8 to 15× slower than ECDSA secp256k1, but ML-DSA verification is comparable or faster: ML-DSA-44 verification is approximately twice as fast as ECDSA. Since blockchains verify far more signatures than they produce, the verification profile of ML-DSA is surprisingly favorable for on-chain use. SLH-DSA tells a different story: signing ranges from milliseconds to over a second depending on the parameter set, and verification, while much faster than signing, is still 12 to 50× slower than ECDSA. ML-DSA is the fastest of the NIST PQ signature standards. If a network chooses SLH-DSA for its more conservative security assumptions, the compute overhead becomes a binding constraint alongside size. But for ML-DSA specifically, the dominant cost of post-quantum signatures on-chain is wire size, not verification time.

In practice, PQ signature adoption almost always implies protocol-level changes rather than a simple algorithm swap, because existing transaction formats, script limits, and fee models were designed around compact elliptic curve signatures.

SECTION TWO

Chain-Specific Engineering Constraints

Post-quantum signatures create different engineering constraints across chains, but the core problem is universal: transaction formats, consensus rules, and fee models were designed around compact elliptic curve signatures, not multi-kilobyte post-quantum alternatives.

Bitcoin

Bitcoin's scaling model is built around SegWit weight accounting: a 4,000,000-WU block limit where non-witness bytes cost 4 WU and witness bytes cost 1 WU. The witness discount helps, but it does not eliminate the cost of larger signatures — the bytes still flow through the network and must be stored by nodes. BIP-360 [39] introduces a new output type, P2MR (Pay-to-Merkle-Root), that functions like Taproot but with the quantum-vulnerable key-path spend removed. A companion proposal, QBIP [40], builds on BIP-360 by defining a phased sunset of legacy ECDSA and Schnorr signatures: first disallowing new spends to quantum-vulnerable address types, then, after a multi-year transition window, disallowing spends from them entirely.

A 2,420-byte ML-DSA-44 signature consumes roughly 605 vbytes of witness data alone, several times larger than a typical single-input P2WPKH spend today. Bitcoin Script also imposes a 520-byte maximum stack element size, retained by BIP 342, meaning a multi-kilobyte signature cannot be pushed and checked in the current model.

Adopting PQ signatures likely requires new witness program versions, new opcodes, and revised fee and propagation assumptions. BIP-360 introduces P2MR, a Taproot-style output type with the quantum-vulnerable key-path spend removed. QBIP builds on this with a phased sunset of legacy ECDSA and Schnorr signatures.

Ethereum

Ethereum's migration is more complex because it embeds three distinct quantum-vulnerable primitives: ECDSA for externally owned accounts, BLS signatures for Proof-of-Stake consensus, and KZG commitments for data availability.

Account abstraction provides a viable path for user-account migration by decoupling transaction validation from a hardcoded signature scheme. But the consensus layer is harder: BLS aggregation is what makes Ethereum's large validator set practical, and there is no production-ready post-quantum analogue today.

Lean Ethereum and the Strawmap are the main coordinated efforts, with research focused on replacing BLS with hash-based signatures, exploring zkVM-based aggregation, and migrating accounts through account abstraction. None of these paths are trivial or production-ready yet.

The most significant post-quantum effort for Ethereum is Lean Ethereum [41], a coordinated research and engineering program tackling both consensus-layer and account-layer migration. The overarching proposal of this process planning for post-quantum migration is the Strawmap [42].

Solana

Solana's transaction format is tightly optimized for Ed25519's 32-byte public keys and 64-byte signatures, with a default transaction size limit of 1,232 bytes. A single ML-DSA-44 signature is 2,420 bytes, and its public key is 1,312 bytes, exceeding the limit even for a basic transfer.

EdDSA with RFC-8032 key derivation computes the signing scalar from a seed via a hash function [43]. Recent research has shown that this property enables post-quantum-secure zero-knowledge proofs of seed knowledge, potentially allowing EdDSA accounts to bind their existing address to a new post-quantum key without moving funds [44].

Project Eleven's post-quantum Solana testnet confirmed that the bottleneck is not verification speed, but data movement. The roughly 16× increase in per-transaction size propagates through packet ingestion, deserialization, batching, and block production, reducing sustainable throughput proportionally. A production deployment would require a purpose-built transaction format, a new address primitive, redesigned packet handling, and updated fee models.

Other Chains

Every chain using elliptic curve signatures faces the same broad issue: PQ signatures are much larger, and every layer that touches signature data must be redesigned around that reality.

EdDSA HD wallets using SLIP-0010 derivation extend this advantage: every node in the hierarchy is a seed derived by HMAC from the parent seed, meaning the entire key tree remains quantum-safe even if individual signing scalars are compromised [43].

The migration path also depends on the signature scheme. EdDSA chains — including Solana, Sui, Aptos, Near, and Stellar — may have a cleaner path because RFC-8032 derives the signing scalar from a seed via a hash function. Even if a CRQC recovers the signing scalar, it cannot reverse the hash to recover the seed. This may allow accounts to prove seed ownership and bind an existing address to a new PQ key without moving funds.

ECDSA chains generally lack this property. ECDSA wallets typically derive or sample private scalars directly, so the same zero-knowledge migration path does not apply cleanly. Some BIP-32/BIP-85 wallets may support bespoke ZK approaches, but proving those derivations is expensive and does not help dormant accounts with unknown derivation paths.

SECTION TWO

Mitigating the Quantum Threat

Mitigation is not a single upgrade, because the quantum threat arrives through two different mechanisms that affect different parts of a system on different timelines.

Confidentiality Risk (HNDL)

If a system uses quantum-vulnerable key establishment today, an attacker can record encrypted traffic and attempt to decrypt it later once a CRQC exists. This is the harvest-now, decrypt-later attack, and it means the damage is being done now, even though the decryption happens in the future.

Authorization Risk (HNFL)

If a system relies on elliptic curve signatures for authorization, and an attacker can recover private keys from public keys, they can forge signatures and authorize transfers. For blockchains, this is the existential risk: not just theft, but a collapse of the ownership model itself — if anyone can produce a valid signature, signatures no longer prove ownership.

NIST guidance anticipates long transition periods and explicitly discusses hybrid approaches (running classical and post-quantum algorithms in parallel) as a pragmatic bridge. For digital assets, however, blockchains cannot typically deploy signature changes easily — strategies that require two full ecosystem migrations (first to hybrid, then to PQ-only) are often not viable. Most networks will need to aim for a single, well-planned migration.

SECTION TWO

Implementing Post-Quantum Cryptography in Protocols

Implementing PQC in Protocols: Three Layers

Layer 1: Off-Chain Key Establishment & Transport

Key establishment is not the dominant risk for blockchains, but it is a material risk surface for exchanges, custodians, wallets, validators, and infrastructure providers. Post-quantum key exchange should be treated as an immediate baseline requirement — not a future roadmap item.

Layer 2: Signature Ecosystems & Trust Roots

Post-quantum signatures are slower to deploy because they sit inside long-lived trust systems. High-value trust roots — release signing, firmware integrity, custody signing services — should be the immediate priority.

Layer 3: On-Chain Authorization & Consensus Validation

On-chain signatures are the primary quantum risk. Any change implies changes to transaction formats, relay policies, fee markets, block propagation, and developer tooling. This is where the hard work lives.

Layer 1: Off-Chain Key Establishment

Key establishment is not the dominant risk driver for blockchains themselves, but it is a material risk surface for the broader digital asset ecosystem. Exchanges, custodians, wallets, validators, sequencers, bridges, and infrastructure providers all operate private control planes that carry sensitive data and high-impact administrative actions. These channels are plausible targets for long-term interception.

This work is low-coordination, high-leverage, and can be rolled out without user migrations. For digital asset operators, upgrading TLS and SSH to support post-quantum key exchange should be treated as an immediate baseline requirement, not a future roadmap item. If your infrastructure is not already running post-quantum key exchange, you are behind the security baseline that the rest of the Internet established in 2025.

Layer 2: Signature Ecosystems & Trust Roots

Post-quantum signatures are slower to deploy than post-quantum key establishment because signatures sit inside long-lived trust systems. Certificates, code signing, firmware signing, secure boot, and internal service authentication all depend on signature formats and verification logic that is widely deployed and difficult to update.

For digital assets, this layer matters because it is a common path to catastrophic loss. If an attacker can subvert software distribution or signing, they do not need to break the chain directly. They can ship a malicious wallet update, a compromised hardware wallet firmware, or a tampered node binary. That is why post-quantum signatures need to be treated as part of the supply chain and operational security program, not only as a future on-chain upgrade.

The IETF's PQC engineering guidance makes the asymmetry clear: PQ key exchange is relatively self-contained, while PQ signature migration requires broader ecosystem changes across certificates, certificate authorities, HSMs, and trust anchors. The IETF LAMPS working group is actively defining how ML-DSA, SLH-DSA, and composite signatures are encoded in X.509 certificates. Adoption will be incremental: the first post-quantum certificates are expected in 2026, but broad availability and browser trust is unlikely before 2027. For digital asset organizations, high-value trust roots — release signing, firmware integrity, and custody signing services — should be the immediate priority, with broader PKI and application-layer migration following as standards and tooling mature.

SECTION TWO

Implementing Post-Quantum Cryptography in Protocols (continued)

Layer 3: On-Chain Authorization and Consensus Validation

On-chain signatures are the primary quantum risk for blockchains. Any change to the authorization mechanism implies changes to transaction formats, relay policies, fee markets, block propagation assumptions, and developer tooling.

Hybrid or dual-signature strategies — running classical and PQ signatures in parallel during a transition — are recommended by NIST and make sense in many contexts. On-chain, however, they are substantially harder to justify for three reasons.

Economic Overhead

Carrying and verifying multiple signatures per transaction increases bandwidth and validation cost, reducing throughput or raising fees during the transition period. On chains that are already capacity-constrained, this is not a marginal cost.

Operational Complexity

Wallets, exchanges, and custodians must support multiple signing paths and multiple address or account types, which increases failure modes during migration. Anyone who has lived through a major protocol upgrade knows that the number of things that can go wrong scales faster than the number of changes.

Ecosystem Coordination

Unlike browsers and web servers, blockchains cannot generally change authorization formats invisibly. A hybrid approach can imply two full migrations for users and infrastructure: first into a hybrid state, and later into a post-quantum-only state. Each migration requires wallet updates, exchange integration work, custodian re-certification, and user key rotation at scale.

This is a structural difference between blockchains and the Internet protocols that have already begun their PQ transition. In TLS, hybridization is deployed without end users doing anything. On a blockchain, signature schemes and validation rules are part of the protocol, and the migration burden is carried by every wallet and every actor that signs transactions.

The implication for planning is that many networks will aim for a single major signature migration on-chain, supported by a migration mechanism that can move users safely without requiring a second coordinated transition. Getting this right as one migration, not two, is the central design challenge of blockchain PQC.

Blockchain vs. Internet PQC Transition

In TLS, hybridization is deployed without end users doing anything.

On a blockchain, signature schemes and validation rules are part of the protocol, and the migration burden is carried by every wallet and every actor that signs transactions.

"Getting this right as one migration — not two — is the central design challenge of blockchain PQC."

SECTION TWO

The Blockchain Migration Challenge

A recent structured analysis of enterprise PQC migration timelines [46] found that even for traditional centralized organizations, migration takes far longer than commonly assumed: 5–7 years for small enterprises, 8–12 years for medium enterprises, and 12–15+ years for large enterprises, under baseline assumptions. These estimates decompose migration into sequential phases with hard dependencies between them:

- **Discovery & Inventory** (1–3 years): Identifying all cryptographic usage
- **Infrastructure Upgrade** (2–7 years): Replacing HSMs, PKI, network hardware
- **Application Migration** (3–10 years): Updating code, protocols, and integrations
- **Partner Synchronization** (1–5 years): Coordinating with external dependencies
- **Hybrid Operation** (ongoing): Maintaining dual classical/PQC systems during transition

Historical Precedents

Campbell's analysis draws on historical cryptographic transitions as benchmarks. The AES migration (DES to AES) took approximately 5 years. The SHA-1 deprecation (SHA-1 to SHA-2) took approximately 7 years and required browser vendors to force the transition by rejecting SHA-1 certificates. The TLS 1.3 rollout took 3–5 years despite offering clear performance improvements and backward compatibility.

Why PQC Is More Complex

Campbell notes that PQC migration is fundamentally more complex than any of these precedents due to larger parameter sizes (5–50× increase in signatures and certificates), hybrid operation requirements during transition, ecosystem-wide coordination needs, and deeper integration of cryptography into hardware.

These are the timelines for centralized organizations with dedicated security teams, executive authority to mandate changes, and established procurement processes. Blockchain protocols face a categorically different — and harder — migration challenge.

SECTION TWO

Why Blockchain Migration Is Structurally Harder

Blockchain PQC migration diverges from enterprise migration along almost every dimension that Campbell identifies as critical:

Decentralized Governance vs. Executive Authority

In Campbell's framework, enterprise migration is driven by executive sponsorship, program management offices, and regulatory compliance mandates. A CTO can mandate a cryptographic upgrade and direct resources accordingly. Blockchain protocols have no such authority. Changes require broad community consensus, typically through contentious governance processes (BIPs for Bitcoin, EIPs for Ethereum) that can take years to navigate.

The Bitcoin SegWit upgrade—a relatively modest change compared to PQC migration—took over two years from proposal to activation (2015-2017) and triggered a contentious chain split (Bitcoin Cash). Ethereum's merge from proof-of-work Ato proof-of-stake required approximately 6 years of research and development. A PQC migration, which touches the most fundamental cryptographic primitive in the protocol (the signature scheme securing all funds), will face at least comparable governance friction.

User-Initiated Asset Migration vs. IT-Managed Upgrades

In enterprise migration, the IT team migrates systems on behalf of users. Users may not even notice the cryptographic transition. In blockchain migration, individual users must actively migrate their own funds from classical-key addresses to post-quantum addresses. This is analogous to requiring every bank customer to personally visit a branch to upgrade their account—except there is no customer service, no help desk, and the penalty for failing to migrate could be total loss of funds.

Historical evidence suggests user migration rates will be slow. Bitcoin's SegWit adoption—which offered clear fee benefits to users—took over 3 years to reach 80% of transactions. Address format upgrades that require user action proceed even more slowly. Many users will have lost access to their keys entirely (estimated 3-4 million BTC are permanently lost), and some funds sit in multisig arrangements, smart contracts, or custodial structures that require coordinated action by multiple parties.

Three additional structural differences compound the challenge for blockchains: blockchain ledgers are immutable and public, so every exposed public key remains a permanent target; enterprise-style retroactive protection is impossible; and post-quantum signatures are significantly larger, meaning the migration fundamentally alters the throughput, fee economics, and state growth of the network.

SECTION TWO

A Blockchain PQC Migration Framework

Government and national cyber guidance consistently frames PQC migration as a multi-year program. NSA CNSA 2.0 sets milestones such as "support and prefer" for key establishment by 2025, with "exclusively use" targets between 2030 and 2033 [47]. The UK NCSC uses milestones of 2028, 2031, and 2035 for complete migration [48]. NIST IR 8547 states that 112-bit security public-key algorithms are planned to be deprecated after 2030, with a goal of disallowing all quantum-vulnerable public-key algorithms entirely after 2035 [49].

For digital assets, a credible migration plan should align with these external milestones, but must account for the coordination costs analyzed above. Because blockchain upgrades move more slowly than centralized system upgrades, not faster, networks need a more aggressive start date. A post-quantum-safe authorization path needs to be available early so that migration can begin well before it becomes urgent, because by the time it becomes urgent, it will be too late.

We recommend protocols approach this migration systematically in 3 separate phases:

Phase 1: Cryptographic Inventory, Operational Readiness, and Off-Chain Hardening

The first requirement is visibility. Every digital asset organization needs a cryptographic inventory, not just of public endpoints, but of embedded and third-party dependencies across the full stack: wallet SDKs, hardware wallets, custody systems (HSM and MPC), remote signers, validator and sequencer operations, bridges, RPC infrastructure, and software supply chain signing. Without this map, protocol and operational upgrades will miss critical components. You cannot migrate what you have not inventoried.

In parallel, deploy post-quantum key establishment in the off-chain control plane wherever it is operationally feasible. Hybrid TLS and the existing OpenSSH defaults are mitigations that can be rolled out without protocol changes or user migrations, and they reduce long-term confidentiality exposure while on-chain plans mature. For most organizations, this should be treated as an immediate action item.

Finally, start the on-chain program in Phase 1, even if the on-chain changes ship later. The longest lead-time dependency for blockchains is not the cryptography but ecosystem coordination. Wallets, exchanges, custodians, and users all have to adopt the new authorization path, and that adoption window cannot be compressed. Starting the design, specification, and ecosystem engagement work now is not premature; it is the only way to have enough runway.

SECTION TWO

A Blockchain PQC Migration Framework

Phase 2: Deploy an On-Chain Authorization Path Designed to be the Final Migration

This is the critical path for digital assets. The objective is not simply to support post-quantum signatures, but to deploy an authorization path that can realistically become the default without requiring two full user migrations.

In practice this implies:

Protocol support:

Deterministic verification in consensus clients, and transaction and script formats that can carry the new authorization data without hitting existing size or compute limits.

Economic sustainability:

Fee and resource pricing rules that reflect the real costs of larger authorization data and verification, so the network remains stable under load and incentives align with migration rather than against it.

Migration mechanics:

Safe key rotation and upgrade flows that can be automated by exchanges and custodians, supported by wallet UX that makes upgrading routine rather than exceptional, with clear failure handling for edge cases.

Testnets and rollout:

Public testnets, independent security audits, and performance benchmarking under realistic load, followed by a staged mainnet rollout with published integration timelines for major ecosystem actors.

Ecosystem readiness:

Reference implementations in major languages, migration tooling, and monitoring dashboards to measure adoption rates and identify cohorts that are not upgrading.

The intent is to give users a viable post-quantum-safe option early, then allow a long runway for voluntary migration before any attempt to deprecate legacy authorization.

Phase 3: Default Adoption and Deprecation of Quantum-Vulnerable Authorization

External guidance converges on the early 2030s for completing migration across major system categories. For blockchains, "complete" should be interpreted strictly: the default way to authorize value transfers no longer relies on elliptic curve signatures, and legacy authorization paths are possibly deprecated rather than left as permanent escape hatches.

In a blockchain context, deprecation is typically gradual. It usually means: defaulting new outputs and accounts to post-quantum-secure authorization; increasing policy and economic friction for legacy authorization (higher fees, reduced priority, warning messages in wallets); and eventually disabling legacy paths via consensus once measured adoption is high and the ecosystem has had sufficient time and tooling to migrate safely.

The uncomfortable truth is that this phase — complete deprecation — is where most blockchain PQC plans become vague. It is easy to propose a new signature scheme; it is hard to build the social and technical machinery to actually move an entire ecosystem off the old one. More importantly, it is not just a coordination problem, it is a throughput problem. Even if every user is willing to migrate, the system still needs enough blockspace to process that migration.

SECTION TWO

Migration Throughput Reality Check

A useful way to ground this is to ask a simple question: how long would it take for every **Bitcoin** UTXO to move to a PQ address type, if a fixed share of **Bitcoin** blockspace were dedicated purely to migration transactions?

Under an extreme assumption where 100% of blockspace is used for migration, recent research shows it will take ~76 days [50]. As soon as migration competes with normal economic activity, timelines extend significantly.

The implication is simple. Migration timelines are bounded by system capacity. If meaningful migration only begins once the quantum threat is widely accepted, there may not be enough time for the system to complete the transition.

"The networks that will be best positioned are the ones that design for the full lifecycle from the beginning: not just 'how do we support PQ signatures?' but 'how do we complete migration within real throughput limits before the threat materializes?'"

TIME TO MIGRATE ALL UTXOS (DAYS) VS % BLOCKSPACE USED FOR MIGRATION TXS

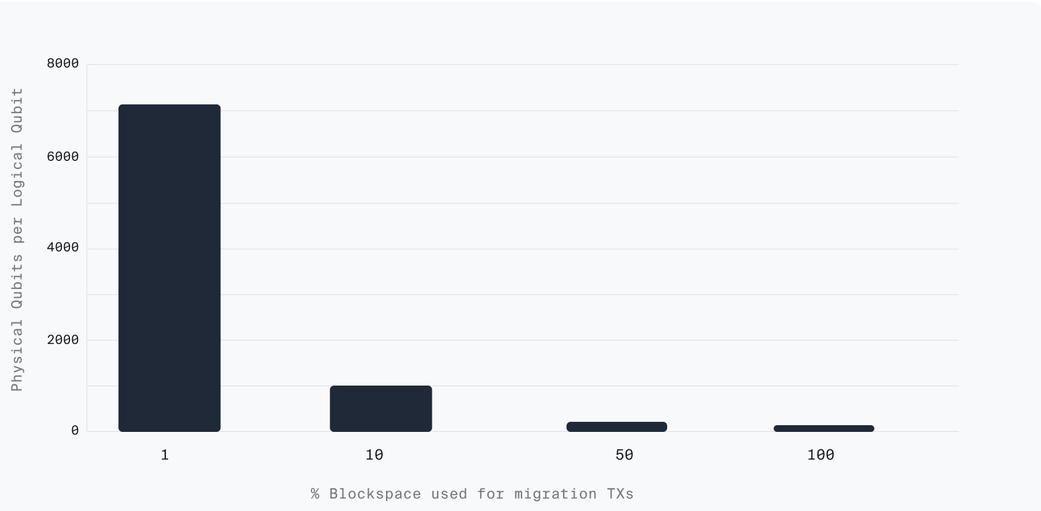


Figure 23

SECTION TWO

Conclusion

The post-quantum transition is not a future problem. It is a present-tense engineering and coordination challenge with a deadline that is set by physics and hardware progress, not by when the blockchain industry decides it is ready.

The Internet has already moved. Over half of web traffic is post-quantum encrypted. SSH defaults are post-quantum. Major cloud platforms and mobile operating systems ship PQC in production. The digital asset industry — which arguably has more at stake because blockchains directly protect bearer value with the exact cryptographic primitives that quantum computers threaten — has barely started.

The gap is not technical; the NIST standards exist, the algorithms work, and reference implementations are available. The gap is entirely coordination, urgency, and willingness to accept the costs of migration.

For Digital Asset Builders, Operators & Investors

Off-chain key establishment should be post-quantum today.
If it is not, you are behind the baseline that the rest of the Internet established in 2025.

On-chain PQC is a multi-year program that must start now.
Blockchain governance moves slowly, ecosystem migrations take years, and the threat timeline is compressing. Networks that have not begun serious design and specification work for post-quantum on-chain authorization are already at risk of running out of runway.

The signature problem is harder than the key exchange problem.
The Internet solved the easy half first. Blockchains face the hard half, and they face it with governance models that were not designed for urgent, ecosystem-wide cryptographic transitions.

"The networks that move early will define the post-quantum era of digital assets. The networks that wait will face emergency migrations under pressure, with all the risk, cost, and potential for value destruction that implies. The time to prepare is not when a CRQC is announced. It is now."

03

Appendices & Citations

Supplementary technical material and source references for the main report. Appendices A–C develop the underlying theory of Shor's algorithm, quantum hardware modalities, and quantum error correction. Appendix D defines NIST's PQC security categories. Appendix E details the Q-Day model. Appendix F proposes a faster PQC signature suite. Followed by a complete bibliography.

APPENDIX_A

Shor’s Algorithm and Its Variants

An analysis of Shor’s algorithm and its optimizations, examining the quantum period-finding technique that underpins attacks on classical cryptography, and surveying three decades of improvements that have progressively reduced the quantum resources required to factor RSA-2048 and break elliptic curve keys.

APPENDIX A

Shor's Algorithm and It's Variants

Shor's algorithm, developed by Peter Shor in 1994, is a quantum algorithm that efficiently factors large integers, a problem that is computationally intractable for classical computers.

The algorithm's brilliance lies in reducing the factoring problem to a problem of period finding, which quantum computers can solve exponentially faster than classical computers. Specifically, to factor a number N, Shor's algorithm finds the period r of the function f(x) = ax mod N for some randomly chosen a. This period r is the key to finding factors: if r is even and a(r/2) is not congruent to -1 mod N, then gcd(a^(r/2) ± 1, N) will yield a nontrivial factor of N with high probability. While the reduction from factoring to period finding is purely classical number theory, the quantum speedup comes from how efficiently a quantum computer can find this period.

The quantum advantage in period finding comes from the Quantum Fourier Transform (QFT), a quantum analogue of the classical discrete Fourier transform. The QFT can be implemented on a quantum computer using only O((log N)2) quantum gates, whereas a classical FFT requires O(N log N) operations (an exponential difference in scaling). In Shor's algorithm, the QFT is applied to a superposition of states that encodes the periodic function f(x). Through the properties of quantum interference discussed earlier, the QFT amplifies the amplitudes corresponding to multiples of the period while suppressing all other amplitudes. When the quantum state is measured after the QFT, the result reveals information about the period r with high probability. This entire quantum subroutine runs in polynomial time—specifically O((log N)3) operations using fast multiplication techniques—making it feasible to factor numbers that would be impossible for classical computers.

To understand the dramatic speedup, consider the best known classical factoring algorithm: the General Number Field Sieve (GNFS). GNFS has a sub-exponential runtime of approximately O(exp((64/9 * log N)(1/3) * (log log N)(2/3))), which, while better than purely exponential algorithms like trial division, still grows extremely rapidly with the size of N. For a 2048-bit number (commonly used in RSA encryption), GNFS would require roughly 2^112 operations—far beyond the reach of any conceivable classical computer, even with all the world's computational resources combined. In contrast, Shor's algorithm running on a quantum computer could factor the same 2048-bit number using only about 10 million quantum gates, executable in a matter of hours once a sufficiently large, fault-tolerant quantum computer exists. This exponential-to-polynomial reduction in complexity is what makes Shor's algorithm a fundamental threat to modern cryptography, transforming a problem that would take longer than the age of the universe into one solvable in an afternoon.

APPENDIX A

Quantum Period Finding

Shor's algorithm is the most famous instance of a more general quantum technique: quantum period finding. The core insight is that many hard number-theoretic problems—integer factoring, discrete logarithms over finite fields, and elliptic curve discrete logarithms—can each be reduced to the problem of finding the period of a function defined over a cyclic group. Classical computers have no efficient method for extracting such periods when the group is exponentially large, but a quantum computer can do so in polynomial time.

The quantum period-finding subroutine proceeds as follows. Two quantum registers are prepared: a phase register of approximately 2n qubits initialized in a uniform superposition, and a work register of n qubits. The function f(x)—for example, ax mod N in the case of factoring—is computed coherently into the work register, entangling the two registers such that each value in the work register is correlated with a set of phase register values spaced exactly r apart, where r is the unknown period. At this point, the Quantum Fourier Transform (QFT) is applied to the phase register. The QFT converts the periodic structure encoded in the amplitudes into sharp peaks at integer multiples of 2(2n) / r in the frequency domain. A measurement of the phase register then yields a value close to j * 2^(2n) / r for a random integer j, from which r can be extracted via the classical continued fractions algorithm. Repeating the procedure O(1) times succeeds with high probability.

The QFT itself is efficient—requiring only O(n2) gates, or O(n log n) using the approximate QFT of Coppersmith (1994)—and is never the computational bottleneck. The dominant cost is the modular exponentiation circuit, which must compute ax mod N for a superposition of exponentially many values of x. In a naive implementation this requires O(n^3) quantum gates, and virtually all algorithmic optimizations to Shor's algorithm focus on reducing the cost of this step through better arithmetic circuits, reformulations of the underlying number-theoretic problem, or space-time tradeoffs in the quantum circuit layout.

The generality of quantum period finding is what makes it so consequential: the same subroutine that factors integers also breaks Diffie-Hellman key exchange over finite fields and recovers private keys from elliptic curve public keys. Any cryptosystem whose security rests on the hardness of a hidden period problem is, in principle, vulnerable to a sufficiently large quantum computer running a variant of this technique.

APPENDIX A

Algorithmic Optimizations

Shor's 1994 paper [4] described a general approach, but the way it is implemented in practice can be tailored to the specific problem and hardware constraints. Over the past three decades, a series of algorithmic optimizations have substantially reduced the quantum resources required to carry out Shor's algorithm—lowering qubit counts, shrinking circuit depths, and reducing the number of expensive T-gates that dominate the cost of fault-tolerant quantum computation. These improvements matter because they effectively lower the threshold at which a quantum computer becomes cryptographically relevant: even without any hardware breakthroughs, better algorithms bring the threat closer.

The following subsections describe the most significant optimizations, followed by a summary comparison table benchmarked against RSA-2048 factoring.

Beauregard (2002) [51]

Stephane Beauregard introduced a circuit for Shor's algorithm requiring only 2n + 3 qubits, a dramatic reduction from the roughly 5n qubits needed by earlier constructions. The key technique is Draper's QFT-based addition circuit, which performs arithmetic directly in the Fourier (phase) domain rather than the computational basis, eliminating the need for classical carry propagation. Beauregard further reduced qubit overhead by using a "semi-classical" QFT that reuses qubits via sequential phase kickback, removing the need for a separate phase register entirely. For RSA-2048, this yields approximately 4,099 logical qubits. However, the space efficiency comes at the cost of circuit depth: the full algorithm requires O(n3) Toffoli gates, translating to on the order of 10^11 T-gates after decomposition—a deep, serial circuit that demands long coherence times.

Zalka (1998, 2006) [52, 53]

Christof Zalka explored the opposite end of the space-time tradeoff spectrum, showing how to parallelize modular exponentiation to reduce circuit depth at the cost of additional qubits. His 2006 result demonstrated that factoring can be performed with as few as n + o(n) qubits, approaching the information-theoretic minimum, while his depth-optimized variant uses O(n^2) qubits to achieve O(n) circuit depth. Zalka's work established the fundamental principle that space and time are exchangeable resources in quantum factoring—a principle that every subsequent optimization has exploited.

Ekera-Hastad (2017) [54]

Martin Ekera and Johan Hastad observed that integer factoring can be reformulated as a "short" discrete logarithm problem. When factoring N = pq, the factors p and q are each only n/2 bits long, yet standard Shor's algorithm uses a phase register of 2n qubits to find the full order r, which may be as large as n bits. Ekera and Hastad showed that because the secret (the factor) is much smaller than the group order, a phase register of only approximately n/2 bits suffices. This roughly halves the number of controlled modular multiplications—the most expensive component of the circuit—yielding an approximately 4x reduction in total quantum gate count for factoring. This reformulation was later adopted as a core component of the Gidney-Ekera construction, and Ekera's subsequent work extended the technique to provide constant-factor improvements for the elliptic curve discrete logarithm problem as well.

Gidney-Ekera (2021) [55]

Craig Gidney and Martin Ekera produced the most concrete and widely cited resource estimate for RSA-2048 factoring. Their construction combines multiple techniques: windowed arithmetic using classical precomputation to reduce quantum multiplication cost, oblivious carry runways to handle carry propagation without knowing intermediate values, the Ekera-Hastad short discrete log reformulation, and the approximate QFT. The result is a circuit requiring approximately 4,000 logical qubits and 2.4 ∗ 10^10 Toffoli gates (approximately 10^11 T-gates). Under a surface code error correction scheme with physical error rates of 10^-3, this translates to roughly 20 million physical qubits and a runtime of 8 hours. Gidney's subsequent work (2024) further refined these estimates, projecting that improved magic state distillation and the use of qLDPC error correcting codes could reduce the physical qubit requirement to approximately 4 million—a 5x improvement achieved purely through better classical and quantum compilation, with no changes to the underlying hardware.

APPENDIX A

Algorithmic Optimizations (continued)

Regev (2023) [15]

Oded Regev proposed a fundamentally different approach: multidimensional quantum period finding. Rather than computing a single modular exponentiation with an n-bit exponent, Regev's algorithm computes d independent modular exponentiations with smaller (approximately n/d)-bit exponents, then applies a d-dimensional QFT and uses classical lattice reduction (LLL) to recover the period from the multidimensional measurement results. Setting d = sqrt(n), the total gate count improves asymptotically from O(n2) to O(n(3/2)), a genuine reduction in circuit depth that is well suited to coherence-limited hardware. The tradeoff is space: the basic formulation requires O(n^(3/2)) qubits —roughly 90,000 for RSA-2048, far more than the 4,000 of Gidney-Ekera. Follow-up work by Ragavan and Vaikuntanathan (2023) showed that the algorithm can be made to work with O(n) qubits while preserving the O(n^(3/2)) gate count. Whether Regev's approach offers a practical advantage over highly optimized standard Shor for problem sizes like RSA-2048 remains an active area of investigation.

Gouzien-Sangouard (2021) [56]

Elie Gouzien and Nicolas Sangouard demonstrated an extreme space-time tradeoff using a multimode quantum memory architecture. Their scheme factors a 2048-bit RSA integer using approximately 13,436 logical qubits—far fewer than Gidney-Ekera—but requires 177 days of continuous quantum computation. The total space-time volume (qubits multiplied by time) is comparable to other approaches, confirming that the fundamental computational cost is roughly conserved across different tradeoff points. This result is significant because it shows that factoring may become feasible on machines with relatively few, high-quality qubits long before million-qubit processors are available, provided those machines can maintain coherence over extended periods.

Litinski (2023) and Huang et al. (2025) [57, 14]

While the preceding optimizations target RSA factoring, parallel work has focused on the elliptic curve discrete logarithm problem (ECDLP), which underpins the signature schemes used by virtually all blockchains. Daniel Litinski showed that a 256-bit elliptic curve private key can be recovered using approximately 2,330 logical qubits and only 50 million Toffoli gates—roughly two orders of magnitude cheaper than RSA-2048 factoring [57]. More recently, researchers exploited the isomorphism between Edwards and Weierstrass curve representations to reduce the T-count by a further 75%, the T-depth by 87%, and qubit requirements by 12% [14]. These results confirm that ECDLP is a significantly easier quantum target than RSA factoring, meaning that the elliptic curve cryptography used by Bitcoin, Ethereum, and most modern protocols will likely be the first to fall.

Webster et al. (2026) [10]

The Pinnacle Architecture, which utilizes quantum low-density parity check (qLDPC) codes to factor RSA-2048 with fewer than 100,000 physical qubits. By employing generalized bicycle codes, the architecture achieves an encoding density of roughly 101 physical qubits per logical qubit, a 10x to 40x improvement over previous surface code estimates. A key innovation is the "Magic Engine," a specialized module that distills and injects magic states within a single code block to maintain constant throughput. While this significantly lowers the qubit threshold, the scheme shifts the engineering challenge to the requirement for non-local connectivity and high-speed real-time qLDPC decoding.

APPENDIX A

Comparison Table

The following table summarizes the major optimizations benchmarked against RSA-2048 factoring. T-gate counts reflect the total number of T-gates required after Toffoli decomposition. Physical qubit estimates assume surface code error correction with physical error rates near 10^{-3}.

Algorithm / Technique	Year	Key Optimization	T-gates	Space-Time Tradeoff
Shor (original)	1994	Quantum period finding via QFT	~10^12	~6,000 logical qubits; deep serial circuit
Beauregard	2002	QFT-based arithmetic; 2n+3 qubits	~10^11	Minimal qubits (4,099); very deep circuit
Zalka	2006	Parallelized modular exponentiation	~10^11	Tunable: ~3,000 qubits (deep) to ~10,000+ qubits (shallow)
Ekera-Hastad	2017	Factoring as short discrete log; ~4x gate reduction	~10^10	~3n/2 logical qubits; halved circuit depth
Gidney-Ekera	2021	Windowed arithmetic, oblivious carry, short DLP	~10^11	~4,000 logical qubits / 20M physical; 8 hours
Gouzien-Sangouard	2023	Multimode memory; extreme space optimization	~10^10	~13,400 logical qubits; 177 days
Regev	2023	Multidimensional period finding; lattice reduction	~10^9–10^10	~90,000 qubits (basic) or O(n) (improved); shallow depth
Gidney (updated)	2024	Improved distillation + qLDPC codes	~10^10	~4,000 logical qubits / ~4M physical; ~10 hours
Webster et al.	2026	Pinnacle Architecture, GB qLDPC codes, Magic Engine	~10^10	< 100,000 physical qubits (~101:1 overhead); month-scale runtime

APPENDIX A

Implications

The trend across these optimizations is unambiguous: the quantum resources required to break classical cryptography have decreased steadily over the past three decades, and continue to decrease. Each new technique—whether it targets qubit count, circuit depth, T-gate cost, or error correction overhead—effectively lowers the hardware threshold at which a cryptographically relevant quantum computer (CRQC) becomes feasible. Importantly, these algorithmic improvements compose with advances in quantum hardware and error correction. A 2x improvement in qubit quality combined with a 2x reduction in algorithmic resource requirements yields a 4x reduction in the effective distance to a CRQC. The existence of multiple independent optimization axes—hardware, error correction, and algorithms—means that progress on any one front accelerates the overall timeline.

APPENDIX_B

Quantum Computing Modalities

An overview of leading quantum hardware modalities—superconducting circuits, trapped ions, neutral atoms, and photonics—examining their error models, connectivity constraints, control bottlenecks, and the path from physical qubits to fault-tolerant logical computation.

APPENDIX B

Introduction: The Physical vs. Theoretical Threshold

Quantum Error Correction (QEC) relies on the threshold theorem, which postulates that if the physical error rate p of quantum operations is below a certain threshold pth, arbitrarily long quantum computations can be performed by encoding information into logical qubits.

For the surface code, the theoretical threshold is relatively high, often cited at $p_{th} \approx 10^{-2}$ [58]. Operating precisely at this threshold represents a break-even point: the error correction process introduces errors at the exact rate it suppresses them.

However, to achieve sustainable error suppression, where the logical error rate pL is sufficiently low to run millions of gates, the physical hardware must achieve an operational requirement of $p \approx 10^{-4}$. The logical error rate scales according to the code distance d:

$$p_L \approx C\,(p / p_{th})^{\lfloor (d+1)/2 \rfloor}$$

where C is a constant related to the number of combinations of failure mechanisms. Reaching $p \approx 10^{-4}$ dramatically reduces the overhead required to construct stable logical qubits.

The Breakdown of i.i.d. Noise Models

Theoretical QEC models often assume independent and identically distributed (i.i.d.) noise. Under this assumption, the probability of simultaneous errors on two qubits, A and B, is simply $P(A \cap B) = P(A) \times P(B)$.

In practical hardware, particularly superconducting circuits, noise is heavily correlated. A primary source of highly correlated noise is ionizing radiation, such as cosmic rays [59]. When a high-energy particle strikes the silicon substrate, it generates a burst of phonons and quasiparticles that traverse the chip. Consequently, the i.i.d. assumption fails: $P(A \cap B) \gg P(A)P(B)$, fundamentally compromising the surface code's assumption of localized, independent errors. Additionally, noise and error from control lines, crosstalk, measurement, and other mechanisms add to this.

APPENDIX B

Superconducting Circuits: Transmons and Connectivity

The Transmon Hamiltonian

The transmon qubit mitigates charge noise by operating in a regime where the Josephson energy (E_J) far exceeds the charging energy (E_C). By shunting the Josephson junction with a large capacitance, the transmon becomes exponentially insensitive to charge noise n_g [60].

The system is modeled as a nonlinear oscillator. The transmon Hamiltonian is given by:

$$\hat{H} = 4E_C(\hat{n} - n_g)^2 - E_J \cos(\hat{\varphi})$$

where $\hat{n}$ is the Cooper pair number operator and $\hat{\varphi}$ is the superconducting phase difference.

Anharmonicity and Leakage

Because the potential well is a cosine rather than a perfect parabola, the energy levels are not equally spaced. The anharmonicity α is defined as the difference between the first and second transition energies:

$$\alpha = E_{12} - E_{01} \approx -E_C$$

In standard transmons, α / h ≈ −300 MHz. This weak anharmonicity imposes a fundamental speed limit on quantum gates to prevent spectral overlap and subsequent population leakage into the $|2\rangle$ state. Leakage is a change to a non-target state. Recent advances in fabrication, such as scaffold-assisted window junctions, aim to further refine these parameters for better coherence [61].

Lattice Constraints: Heavy-Hex vs. Square

To combat frequency crowding and crosstalk on 2D architectures, IBM introduced the Heavy-Hex lattice [62]. By reducing the connectivity degree from four to two or three, the Heavy-Hex topology minimizes spectator errors and frequency collisions. However, this restricts the native implementation of the standard surface code, requiring specialized Quantum Low-Density Parity-Check (qLDPC) codes, such as the Bivariate Bicycle code, to optimize the physical-to-logical qubit ratio [63, 64]. There has also been research into alternative schemes that have space-time tradeoffs [65].

Google's Willow chip demonstrated below-threshold for surface code error correction for the first time in 2024 [1]. With advances in fabrication, superconducting is developing.

APPENDIX B

Neutral Atoms: Rydberg Blockade and Reconfigurable Arrays

Neutral atoms trapped in optical tweezers present a highly scalable modality capable of dynamic on-the-fly reconfiguration [66].

The Rydberg Blockade

To perform a Controlled-Z (CZ) gate, atoms are excited to a highly energetic Rydberg state ($n \gg 1$). The dipole-dipole interaction potential V_{rr} between two atoms at distance R scales profoundly with the principal quantum number:

$$V_{rr} \propto n^{11} / R^{6}$$

This interaction creates the Rydberg Blockade radius R_b. If a control atom is excited to $|r\rangle$, the massive energy shift prevents a target atom within Rb from also being excited, conditionally mediating the phase shift required for universal entanglement [67].

High-Genus Topological Codes

Because optical tweezers can dynamically move atoms during a computation, neutral atom arrays can synthesize 3D lattices and high-genus topologies. This overcomes the planar restrictions of superconducting chips [65].

- Massive Scaling: Demonstration of a coherent 3,000-qubit system [68] and metasurfaces generating over 78,000 tweezers [69].
- Algorithmic Breakthroughs: The first logical execution of Shor's algorithm [70] and experimental logical magic state distillation [71].
- Fault Tolerance: Low-overhead transversal fault tolerance for universal computation [72].

APPENDIX B

Trapped Ions and the QCCD Architecture

Unlike superconducting qubits, Trapped Ions in a Quantum Charge-Coupled Device (QCCD) architecture feature physical mobility [73]. Ions are shuttled between memory zones and interaction zones.

All-to-All Connectivity

Ions can be physically transported across the potential landscape, thus the architecture boasts "all-to-all" connectivity. This allows for the direct execution of non-local stabilizers essential for high-efficiency qLDPC codes.

Sympathetic Cooling

During transport, ions accumulate motional heating, which drastically reduces the fidelity of Mølmer-Sørensen entangling gates. To circumvent the decoherence caused by direct laser cooling, researchers trap a secondary "refrigerant" species (e.g., $^{138}Ba^{+}$ with a $^{171}Yb^{+}$ qubit). Cooling the Barium ion sympathetically cools the Ytterbium ion via Coulomb interaction without disturbing the internal quantum state [74].

Photonic Modalities: Fusion-Based QEC

Photonic quantum computing circumvents decoherence limits by utilizing traveling photons. Rather than operating via a sequential circuit model, it leverages Measurement-Based Quantum Computing (MBQC) on massive entangled "cluster states."

In PsiQuantum's Fusion-Based Quantum Computing (FBQC) model, computation is driven by destructive multi-photon parity checks known as fusions [75]. Loss, the primary error channel in photonics, is managed not by active gate correction, but by treating lost photons as "erasures" within a highly redundant 3D topological defect network [76].

APPENDIX B

The Control System Bottleneck: Cryo-CMOS

Scaling to 1 million physical qubits introduces a catastrophic thermal bottleneck due to the heat load of millions of coaxial cables spanning from room temperature (300K) to the mixing chamber (≈ 15 mK) [77].

The solution lies in Cryo-CMOS technology, integrated controllers operating at the 4K stage. By multiplexing digital control signals and processing error syndromes locally within the cryostat, systems can overcome the I/O latency constraint, effectively bridging the final physical gap in the hardware-software stack.

The Decoding Bottleneck: Real-Time Error Arbitration

A critical, often overlooked component of the hardware-software gap is the classical processing power required to interpret error syndromes. For a quantum computer to be "fault-tolerant," it must identify and correct errors faster than they propagate.

Decoding Latency and Coherence Time

The decoding problem, mapping observed parity-check violations (syndromes) to the most likely physical errors, is typically solved using algorithms like Minimum Weight Perfect Matching (MWPM) or Union-Find.

• Superconducting Bottleneck: With gate times in the nanosecond range (10–100 ns), the classical decoder must resolve the error graph within microseconds. This necessitates hardware-level decoders (FPGA or ASIC) integrated directly into the Cryo-CMOS stack.

• Ion/Atom Advantage: These modalities have longer coherence times and slower gates (millisecond range), providing a more generous "time budget" for complex classical decoding algorithms.

APPENDIX B

Modality Comparison

Property	Superconducting	Trapped Ion	Neutral Atom	Photonic
Leading vendors	Google, IBM	Quantinuum, IonQ	QuEra, Pasqal, Infleqtion	PsiQuantum, Xanadu
Gate speed	10-100 ns	1-100 us	1-10 us	~ns (measurement-based)
2Q gate fidelity	99-99.5%	>99.99% [77]	~99.5%	Architecture-dependent
Connectivity	Nearest-neighbor (2D grid)	All-to-all (within chain)	Reconfigurable (mid-circuit)	Modular / networked
QEC cycle time	~1 us	~10-100 us	~1-10 ms	Architecture-dependent
Scalability path	Foundry fabrication	Modular linking	Optical tweezer arrays	Semiconductor photonics
QEC code	Surface code (below threshold)	Color code, various	Color code, hypercube code	Fusion-based (theoretical)
Key advantage	Speed, manufacturing maturity	High fidelity, connectivity	Reconfigurability, scale	Room temperature, networking
Key challenge	Cryogenics, limited connectivity	Speed, scaling beyond ~50 qubits	Atom loss, gate speed	Loss rates, determinism

APPENDIX B

Table N: Summary of Quantum Computing Modalities[1]

Modality	Gate Speed	Scalability	Below-Threshold QEC?	Example Architectures	Notes
Superconducting	Fastest (10-100 ns)	Challenging (requires cryogenic cooling; 2D grid limits connectivity)	Yes (Google Willow, 2024)	Google Sycamore/Willow, IBM Eagle/Heron	Most mature platform; benefits from existing semiconductor fabrication infrastructure
Trapped Ion	Slow (1-100 μs)	More challenging (chains limited to ~50 ions; modular linking adds engineering complexity)	Yes (Microsoft/ Quantinuum, 2024)	Quantinuum H-series, IonQ Forte	Highest gate fidelities (>99.5% 2Q); all-to-all connectivity within a chain eliminates routing overhead
Neutral Atom	Moderate (1-10 μs)	Less challenging (optical tweezer arrays scale to thousands; demonstrated 3,000+ qubits)	Yes (QuEra, 2024)	QuEra Aquila, Pasqal Fresnel, Infleqtion Sqorpius	Most rapidly advancing modality; reconfigurable connectivity; first logical Shor's execution (Infleqtion, 2025)
Photonic	Fast (~ns, measurement-based)	Less challenging (room temperature; naturally modular and networked)	No	PsiQuantum, Xanadu Borealis	Theoretical runtime advantages for certain cryptanalytic workloads (2-20x); least experimentally mature
Spin	Moderate (10ns – 1μs, depending on target)	Less challenging (1-4 Kelvin, Mature CMOS production)	No	Intel Tunnel Falls, Diraq Crossbar, QuTech QARPET	Fabricated on standard 300mm CMOS lines, diversity in realization (Si or NV center)

[1] Non-exhaustive, there are other types of quantum computing modalities, including silicon-spin qubits (used by Diraq) and Majorana zero-modes (Microsoft). Those approaches are less well developed in the scientific literature, however.

APPENDIX B

The Performance Layer: QCVV and QEM

For any hardware modality, the bridge between raw physical qubits and reliable computation is built on two pillars: Quantum Characterization, Verification, and Validation (QCVV) and Quantum Error Mitigation (QEM) [79].

QCVV: The Diagnostic Framework

QCVV is the rigorous scientific process of knowing what you have for hardware. Quantum states are fragile and cannot be directly observed without collapsing, QCVV uses indirect statistical methods to build a high-fidelity model of the system's behavior.

• **Characterization**: Identifying specific noise parameters, such as T_1 (relaxation) and T_2 (dephasing) times, or gate fidelities via Gate Set Tomography (GST) [80, 81].

• **Verification & Validation**: Ensuring the hardware is on the right path and double-checking itself. This involves benchmarks like Randomized Benchmarking (RB) [81] and Quantum Volume (QV) [82] to provide a holistic score of the system's operational capacity.

QEM: The Remedial Layer

While QCVV diagnoses the noise, Quantum Error Mitigation (QEM) works to suppress its impact on final results without the massive qubit overhead required for full Fault-Tolerant Quantum Error Correction (FTQEC). QEM is essential for the current NISQ (Noisy Intermediate-Scale Quantum) era [83].

• **Zero-Noise Extrapolation (ZNE)**: Intentionally scaling up noise in a circuit and then extrapolating back to the "zero-noise" limit to estimate the ideal result [84].

• **Probabilistic Error Cancellation (PEC)**: Using a known noise model (derived from QCVV) to apply a "quasi-probability" distribution that cancels out errors across an ensemble of circuit runs [85].

Key Distinction: QCVV provides the data, the identity and cause of errors, whereas QEM provides the action, the cleaning techniques. Together, they enable hardware modalities to reach "quantum utility" long before physical hardware is perfectly noise-free.

APPENDIX B

The Rise of qLDPC Codes: Beyond the Surface Code

Quantum Low-Density Parity-Check (qLDPC) codes, such as Hypergraph Product and Bivariate Bicycle codes, are emerging as a high-efficiency alternative to the traditional surface code by offering constant encoding rates that significantly reduce the physical qubit overhead. While the surface code's 2D planarity is hardware-friendly, its zero encoding rate in the thermodynamic limit necessitates a massive footprint, whereas Bivariate Bicycle codes can encode 12 logical qubits into just 144 physical qubits [63, 64].

However, implementing these codes requires non-local connectivity, making reconfigurable modalities like Neutral Atoms and Trapped Ions more suitable candidates than superconducting circuits [84].

Furthermore, the transition from hardware-intensive Magic State Distillation (MSD), which consumes considerable quantities of the system's qubits, to perform many algorithms.

Additionally, Magic State Cultivation (MSC) shifts the engineering challenge from raw qubit quantity to control complexity, adding nuance to quantum computing in practice [85]. This shift allows for the internal "growth" of non-Clifford resources but demands sophisticated real-time control and dynamic code-switching to maintain logical integrity.

APPENDIX_C

Quantum Error Correction

A rigorous treatment of the theory and engineering of quantum error correction—from the Lindblad master equation and stabilizer formalism to the surface code, fault-tolerant gate synthesis, and asymptotically good quantum LDPC codes.

APPENDIX C

Introduction & The Physics of Noise

In the following section, we'll zoom into the primary engineering challenge of practically implementing a CRQC

The Physics of Noise: Open Quantum Systems and Bosonic Baths

The fundamental challenge of quantum computing is the preservation of state purity in the presence of the macroscopic realm, where noise dominates [87, 88]. An isolated quantum system evolves unitarily. However, real physical qubits are inherently coupled to a surrounding environment, leading to decoherence, leakage, and dissipation.

Hamiltonian Dynamics of the Spin-Boson Model

Consider a composite Hilbert space $H = H_s \otimes H_e$, where H_s is the principal system (a qubit) and H_e is the environment. Simply modeled, the environment is a Bosonic bath, a continuum of non-interacting harmonic oscillators [88]. The total Hamiltonian is given by:

$$\mathbf{H_{tot} = H_s \otimes I_e + I_s \otimes H_e + H_{int}}$$

For a standard two-level system, the bare system Hamiltonian is $H_s = (\omega_q / 2)\, \sigma z$, and the bath Hamiltonian is $H_e = \Sigma_k\, \omega_k\, a_k^{\dagger}\, a_k$, where $a_k^{\dagger}$ and a_k are creation and annihilation operators that are orthonormal.

The interaction Hamiltonian H_{int} typically involves a linear coupling between the system's dipole moment and the bath modes [87]:

$$\mathbf{H_{int} = \Sigma_k\, (\, g_k\, \sigma_+ \otimes a_k + g_k^*\, \sigma_- \otimes a_k^{\dagger}\,)}$$

where g_k quantifies the coupling strength to the k-th mode.

APPENDIX C

Microscopic Derivation of the Master Equation

The exact evolution of the composite system is $\rho(t) = U(t)\,\rho(0)\,U^{\dagger}(t)$, where $U(t) = e^{(-i\,H_{tot}\,t)}$. With only access to the system S, the trace for environmental degrees of freedom to obtain the reduced density matrix is as such:

$$\rho_s(t) = Tr_e\,[\,U(t)\,(\rho_s(0) \otimes \rho_e(0))\,U^{\dagger}(t)\,]$$

To derive a tractable equation of motion, we apply the Born approximation (assuming weak coupling, $\rho(t) \approx \rho_s(t) \otimes \rho_e$) and the Markov approximation (assuming bath correlation time << system relaxation time) [86]. In the interaction picture, taking the trace over the bath yields an integro-differential equation.

Applying the rotating wave approximation averages out rapidly oscillating terms $\exp(i(\omega - \omega')t)$, yielding the Lindblad Master Equation [87, 88]:

$$d\rho_s / dt = -i[H_s', \rho_s] + \Sigma_j\,\gamma_j\,(\,L_j\,\rho_s\,L_j^{\dagger} - (1/2)\,\{L_j^{\dagger}\,L_j,\,\rho_s\}\,)$$

where H_s' is the Lamb-shifted Hamiltonian. The decay rates γ_j are determined by the Fourier transform of the bath correlation functions, directly tied to the spectral density $J(\omega) = \Sigma_k\,|g_k|^2\,\delta(\omega - \omega_k)$. For a qubit undergoing amplitude damping, $L = \sigma_-$; for pure dephasing, $L = \sigma z$.

Kraus Operator Sum Representation

The differential evolution described by the Lindblad equation can be integrated over a time interval Δt to formulate a Completely Positive Trace-Preserving (CPTP) map E. By Stinespring's Dilation Theorem, this map can be expressed via an Operator Sum Representation [86]:

$$E(\rho_s) = \Sigma_k\,M_k\,\rho_s\,M_k^{\dagger},\ \text{where}\ \Sigma_k\,M_k^{\dagger}\,M_k = I_s$$

The operators M_k are the Kraus operators. As the Pauli matrices {I, X, Y, Z} form a complete orthogonal basis for the space of 2 × 2 complex matrices under the trace inner product $Tr(A^{\dagger}\,B) / 2$, any continuous Kraus operator M_k can be uniquely expanded as a linear combination of Pauli matrices. This expansion is the bridge between continuous analog noise and discrete digital errors [87].

APPENDIX C

The Stabilizer Formalism

To protect against the continuum of errors generated by the Bosonic bath, encoded logical information is put into a larger, highly entangled physical Hilbert space. The Stabilizer Formalism provides a group-theoretic approach to manage this [89].

The n-Qubit Pauli Group and the Code Space

Let P_n denote the n-qubit Pauli group, consisting of all n-fold tensor products of the Pauli matrices, together with overall phase factors from the set $\{\pm 1, \pm i\}$. The group has order 4^{n+1}. A stabilizer code is generally defined by choosing an Abelian subgroup $S \subset P_n$ such that $-I \notin S$. If S is generated by n-k independent commuting operators $\{S_1, S_2, \ldots, S_{n-k}\}$, the simultaneous +1 eigenspace of all $S \in S$ defines the code space C [89]:

$$C = \{ |\psi\rangle \in (\mathbb{C}^2)^{\otimes n} : S_i |\psi\rangle = |\psi\rangle \ \forall\, i \}$$

This space has dimension 2^k, encoding k logical qubits. The projector onto the code space is given by:

$$P_C = (1 / 2^{n-k}) \prod_i (I + S_i) = (1 / |S|) \sum_{s} S$$

The Normalizer and Logical Operators

To manipulate the encoded information without leaving the code space, we must identify operators that commute with all stabilizers. We define the normalizer (or centralizer, as they coincide in P_n) of S:

$$N(S) = \{ P \in P_n : P S P^\dagger = S \ \forall\, S \in S \}$$

Any operator in N(S) preserves the code space. However, operators inside S act trivially as the identity on C. Therefore, the non-trivial logical operators are elements of the quotient group:

$$L = N(S) / S$$

This quotient group is isomorphic to the Pauli group on k qubits, P_k, providing the foundation for logical operations [89].

APPENDIX C

Vector Space Representation & Knill-Laflamme Conditions

Vector Space Representation

Manipulating operators in P_n directly is computationally inefficient. We construct a homomorphism to map the operators onto a binary vector space over the Galois Field F_2. We define the map $\varphi: P_n \to F_2^{2n}$ such that the global phase is ignored:

$$\varphi(i^\delta X^u Z^v) = (u \mid v) \in F_2^{2n}$$

where $u, v \in \{0,1\}^n$. Operator multiplication $g \cdot h$ translates directly to vector addition $\varphi(g) \oplus \varphi(h)$.

The commutation relation between two operators $g, h \in P_n$ is captured by the symplectic inner product. Two operators commute if and only if their symplectic inner product is zero. Therefore, a stabilizer code S is mathematically equivalent to a totally isotropic subspace of F_2^{2n} under the symplectic form Ω.

The Knill-Laflamme Conditions

The theoretical guarantee of quantum error correction is encapsulated by the Knill-Laflamme conditions [90].

Theorem (Knill-Laflamme): Let C be a quantum code with projector P_C, and let $E = \{E_a\}$ be a set of errors. The code can correct E if and only if there exists a Hermitian matrix α such that:

$$P_C E_a^\dagger E_b P_C = \alpha_{ab} P_C \; \forall \; E_a, Eb \in E$$

If α_{ab} is a diagonal matrix ($\alpha_{ab} = c_a \, \delta ab$), the condition implies that different errors map the code space to mutually orthogonal subspaces: $\mathrm{Tr}(P_C \, E_a\dagger \, Eb \, P_C) = 0$. Since the subspaces are orthogonal, a projective measurement can perfectly distinguish which error occurred without extracting any information about the logical state [90]. If α_{ab} is not diagonal, the matrix can be diagonalized via a unitary transformation of the error basis, resulting in a degenerate code where multiple distinct physical errors yield the exact same physical state deformation, thus requiring the same recovery operation.

APPENDIX C

Historical Baselines

The [[9,1,3]] Shor Code

The earliest constructive proof of the Knill-Laflamme conditions was the Shor code [91], which concatenates a 3-qubit bit-flip code with a 3-qubit phase-flip code [92]. The code space is spanned by the logical basis:

$$|0_l\rangle = (1 / 2\sqrt{2}) (|000\rangle + |111\rangle)^{\otimes 3}, \quad |1_l\rangle = (1 / 2\sqrt{2}) (|000\rangle - |111\rangle)^{\otimes 3}$$

While historically monumental, the Shor code possesses poor scaling properties and requires highly non-local interactions, making it unsuitable for physical hardware architectures.

The Surface Code, Homology, and Kitaev's Hamiltonian

To achieve practical realization, QEC must respect the local connectivity constraints of physical hardware (e.g., planar superconducting lattices). The Surface (or Toric) Code achieves this by mapping the stabilizer formalism onto the 2-dimensional homology of a manifold [94, 95].

Kitaev originally formulated this not just as an error-correcting code, but as an exactly solvable physical Hamiltonian [93]. Let X be a 2D cell complex. Qubits reside on the edges. The Hamiltonian is:

$$H_{toric} = -J_e \sum_v A_v - J_m \sum_p B_p$$

where $A_v = \prod_e X_e$ (star operators), and $B_p = \prod_e Z_e$ (plaquette operators). The fundamental theorem of homology states that the boundary of a boundary is trivial: $\partial_1 \partial_2 = 0$. This geometric property rigorously guarantees that the star and plaquette stabilizers commute ($[A_v, B_p] = 0$).

The ground state $|\psi_0\rangle$ satisfies $A_v|\psi_0\rangle = |\psi_0\rangle$ and $B_p|\psi_0\rangle = |\psi_0\rangle$, forming the code space C. Excitations above the ground state occur when $A_v = -1$ (an electric charge 'e') or $B_p = -1$ (a magnetic vortex 'm'). These excitations are anyons with mutual fractional statistics; moving an 'e' particle around an 'm' particle imparts a geometric phase of π, fundamentally linking error correction to topological quantum field theory [93].

Logical operators Z_l and X_l correspond to non-trivial cycles that wrap around the manifold, belonging to the homology group $H_1(X, F_2)$ and cohomology group $H^1(X, F_2)$. A logical error occurs only if a chain of physical errors spans the macroscopic lattice [94]. Innovations to this foundational model continue today, driving the development of yoked and dynamic variants to ease hardware constraints [96, 97].

APPENDIX C

The Decoding Problem and Fault-Tolerant Operations

The Decoding Problem and MWPM

When errors occur, they create pairs of anyonic defects. Syndrome measurement reveals the locations of these defects, but not the specific error chain that created them. This is the decoding problem.

For uncorrelated Pauli noise, the problem reduces to Minimum Weight Perfect Matching (MWPM) on a graph G = (V, E) [58]. The vertices V are the measured defects (syndromes equal to -1). A complete graph is constructed where edges e connect every pair of defects. The weight of an edge w_e is related to the probability p_e of an error chain connecting them:

$w_e = -\ln(p_e / (1 - p_e))$

The MWPM algorithm finds a perfect matching $M \subset E$ that minimizes the total weight $\Sigma_e w_e$. This corresponds to the most likely physical error chain, allowing the classical controller to apply the correct recovery operation [58].

Fault-Tolerant Operations and the Eastin-Knill Theorem

Defining a robust memory is only half the challenge; one must execute gates on the encoded data without causing small, correctable errors to cascade into catastrophic logical failures.

APPENDIX C

The Clifford Group and Eastin-Knill Theorem

The Clifford Group and Gottesman-Knill

A gate is transversal if its logical operation U_l can be implemented by applying independent physical gates U_i to each physical qubit: $U_l = \otimes U_i$. Transversal gates prevent error spreading.

We define the Clifford group C_n as the normalizer of the Pauli group:

$$\mathbf{C_n = \{ U \in U(2^n) : U P U^\dagger \in P_n \ \forall \ P \in P_n \}}$$

The Gottesman-Knill theorem states that any quantum circuit comprising only state preparations in the computational basis, Clifford group gates (Hadamard, Phase, CNOT), and Pauli basis measurements can be perfectly simulated efficiently on a classical computer [88]. Therefore, to achieve quantum advantage, we must introduce non-Clifford operations, such as the T-gate ($T = \text{diag}(1, e^{(i\pi/4)})$).

The Eastin-Knill Theorem and Magic States

The pursuit of entirely transversal logic is halted by the Eastin-Knill Theorem [93].

Theorem (Eastin-Knill): No quantum error-correcting code capable of correcting local errors can possess a universal gate set implemented via continuous transversal symmetries [97].

Because the T-gate does not map Pauli operators to Pauli operators ($T X T\dagger = (1/\sqrt{2})(X + Y) \notin P_n$), it is outside the Clifford group.

To circumvent Eastin-Knill, we employ state injection and distillation [98]. We initialize a highly noisy "magic state" $|T\rangle = (1/\sqrt{2})(|0\rangle + e^{(i\pi/4)}|1\rangle)$ by applying an unprotected physical T-gate. We then use a specialized auxiliary code, typically the [[15,1,3]] Reed-Muller code, to "distill" several noisy copies into a single high-fidelity copy. If the initial physical error rate is ε, the failure probability of the distilled output state drops cubically [98]:

$$\mathbf{\varepsilon_{out} = 35\varepsilon^3 + O(\varepsilon^4)}$$

The purified state is then consumed via a Gate Teleportation circuit, transferring the non-Clifford rotation onto the logical data qubit fault-tolerantly.

APPENDIX C

Lattice Surgery and Quantum LDPC Codes

Lattice Surgery

To perform multi-qubit logical operations (like CNOT) between distinct Surface Code patches without moving physical qubits, modern architectures utilize Lattice Surgery [99]. By ceasing the measurement of edge stabilizers and beginning the measurement of multi-body "merge" stabilizers across the gap between two patches, we project the joint system into an eigenstate of $Z_A \otimes Z_B$ or $X_A \otimes X_B$. After recording this parity syndrome, the patches are split back to their original configurations [99].

Quantum LDPC (qLDPC) Codes

While the Surface Code is technologically viable due to its 2D planar constraints, its encoding rate scales abysmally. For a Surface Code of distance d, we require $O(d^2)$ physical qubits to encode exactly 1 logical qubit. The rate $R = k/n \to 0$ as $n \to \infty$. To realize algorithms requiring thousands of logical qubits, we must explore Quantum Low-Density Parity-Check (QLDPC) codes [100].

Derivation of Hypergraph Product Codes

The quest for "asymptotically good" quantum codes was resolved by mapping QEC to high-dimensional expander graphs. The leading candidate is the Hypergraph Product (HGP) code [100].

The HGP code constructs a valid quantum CSS code from two arbitrary classical LDPC codes. Let C_1 be an $[n_1, k_1, d_1]$ classical code with parity check matrix H_1 of size $r_1 \times n_1$. Let C_2 be an $[n_2, k_2, d_2]$ code with matrix H_2 of size $r_2 \times n_2$. We construct the quantum check matrices via Kronecker products [98]:

$$\mathbf{HX = [\, H_1 \otimes I_{n2} \mid I_{r_1} \otimes H_2^T \,]} \quad \mathbf{HZ = [\, I_{n1} \otimes H_2 \mid H_1^T \otimes I_{r_2} \,]}$$

To satisfy the Knill-Laflamme commutativity requirement for CSS codes, we must have $HX\, HZ^T = 0 \pmod 2$. We verify this explicitly:

$$\mathbf{H_X H_Z^T = (H_1 \otimes I_{n2})(I_{n1} \otimes H_2^T) + (I_{r_1} \otimes H_2^T)(H_1^T \otimes I_{r2}) = (H_1 \otimes H_2^T) + (H_1 \otimes H_2^T) = 2(H_1 \otimes H_2^T) \equiv 0 \pmod 2}$$

APPENDIX C

Rate and Distance Bounds of HGP Codes

The total number of physical qubits in the HGP code is equal to the sum of the columns of the constituent matrices:

$$n_q = n_1 n_2 + r_1 r_2$$

The number of logical qubits kq is determined by the dimension of the kernel of the check matrices minus the image. By applying the rank-nullity theorem to the tensor product spaces, one can prove [100]:

$$kq = k_1 k_2 + (n_1 - r_1)(n_2 - r_2)$$

If C_1 and C_2 are identical classical codes with rate Rc = k/n, the quantum rate Rq ≈ Rc^2. Most importantly, this rate remains constant as n → ∞.

The minimum distance of the quantum code is strictly bounded by the minimum distances of the classical inputs [100]:

$$d_q = \min(d_1, d_2)$$

If we construct the HGP code using classical expander codes where $d_c \propto n_c$, the quantum distance scales as $d_q \propto \sqrt{n_q}$ [100]. While not fully linear O(n), the $\sqrt{n_q}$ distance paired with the O(1) constant encoding rate dramatically suppresses the required physical qubit overhead by several orders of magnitude compared to planar Surface Codes.

APPENDIX_D

NIST PQC Security Categories

An overview of the five NIST PQC security categories, defining robustness tiers in reference to AES key search and SHA collision attacks—the benchmarks used to classify ML-KEM and ML-DSA parameter sets.

APPENDIX D

NIST PQC Security Categories

Category	Reference Problem	Classical Security Level
1	Brute-force key search on AES-128	~128 bits
2	Collision search on SHA-256	~128 bits
3	Brute-force key search on AES-192	~192 bits
4	Collision search on SHA-384	~192 bits
5	Brute-force key search on AES-256	~256 bits

As part of the PQC standardization process, NIST introduced five security categories, labeled 1 through 5, to classify the robustness of each algorithm. Each category represents a minimum security level that a PQC algorithm's cryptanalysis must require, defined in reference to well-understood baselines in classical cryptography. This approach avoids over-reliance on precise bit estimates (which are uncertain in the quantum era) and instead uses broad tiers of strength:

Odd-numbered categories (1, 3, 5) define security against brute-force key search on symmetric ciphers. Even-numbered categories (2, 4) define security against hash collision attacks. In practice, most implementations target Category 1 (ML-KEM-512, ML-DSA-44), Category 3 (ML-KEM-768, ML-DSA-65), or Category 5 (ML-KEM-1024, ML-DSA-87).

APPENDIX_ **E**

Q-Day Model

A four-layer supply-and-demand framework for projecting when a cryptographically relevant quantum computer (CRQC) becomes feasible—modeling physical qubits, error correction, control overhead, and algorithm requirements.

APPENDIX E

Bottom-Up vs. Top-Down Predictive Models

BOTTOM-UP

Resource Estimates

- More accurate and rigorous
- Architecture-specific
- Hard to compare across approaches
- Forecasts requirements, not trends

Example **Gidney 2025**

VS

TOP-DOWN

Predictive Models

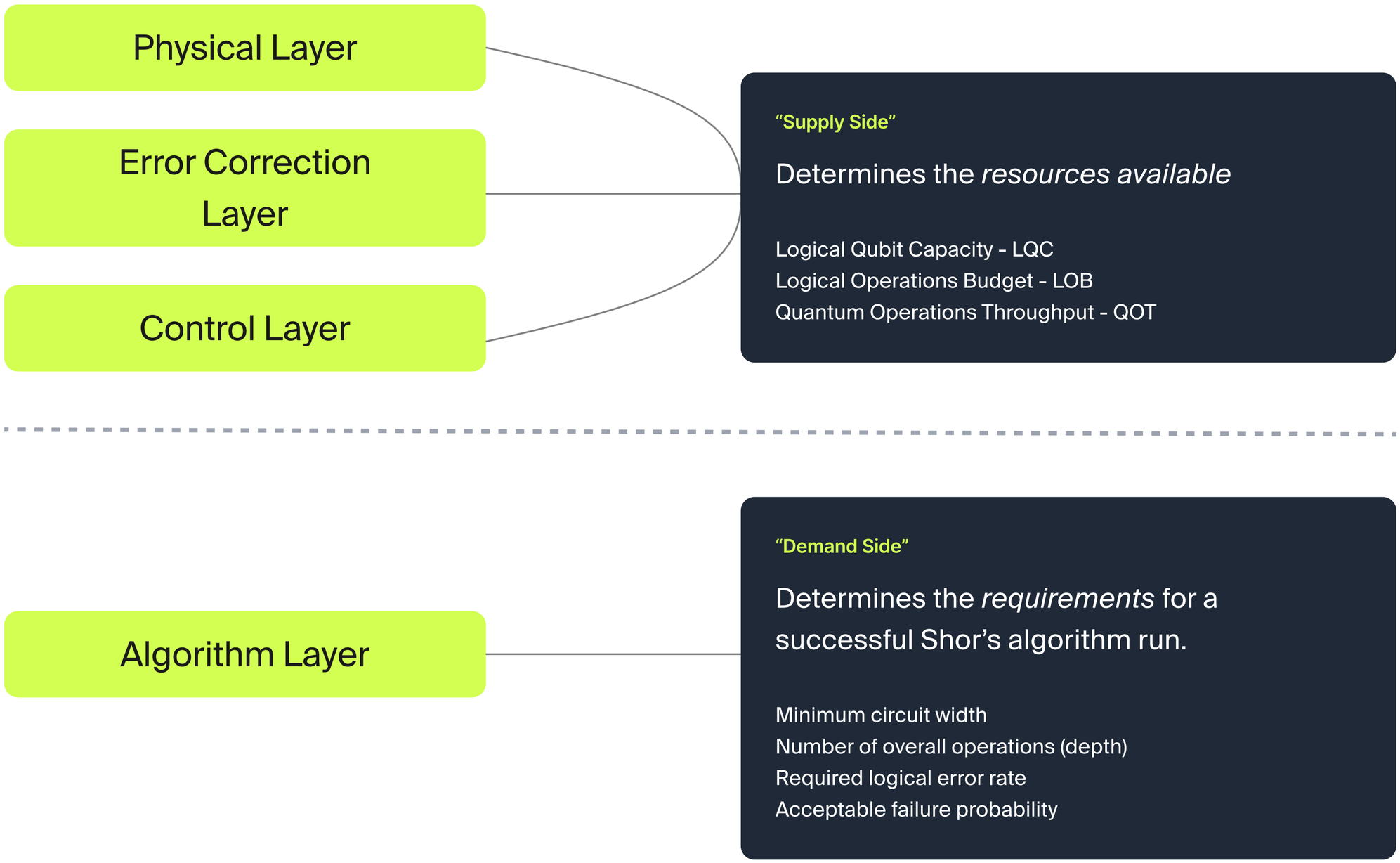
- General enough for multiple approaches
- Produces an explicit timeline forecast
- Additional assumptions reduce scientific precision

Example **This work — P11 Q-Day Model**

"All models are wrong, some are useful"

APPENDIX E

Q-Day Projection Model Framework



APPENDIX E

Three Constraints Define Q-Day

1 QUBIT CAPACITY

LQC ≥ Circuit Width

The number of logical qubits available must be at least as wide so as to accommodate the maximum number of logical operations at any step in the computation.

2 OPERATIONS BUDGET

LOB ≥ Circuit Depth

The number of logical operations available (above the failure threshold epsilon) must be at least as wide as the number of required parallel logical operations at the highest point.

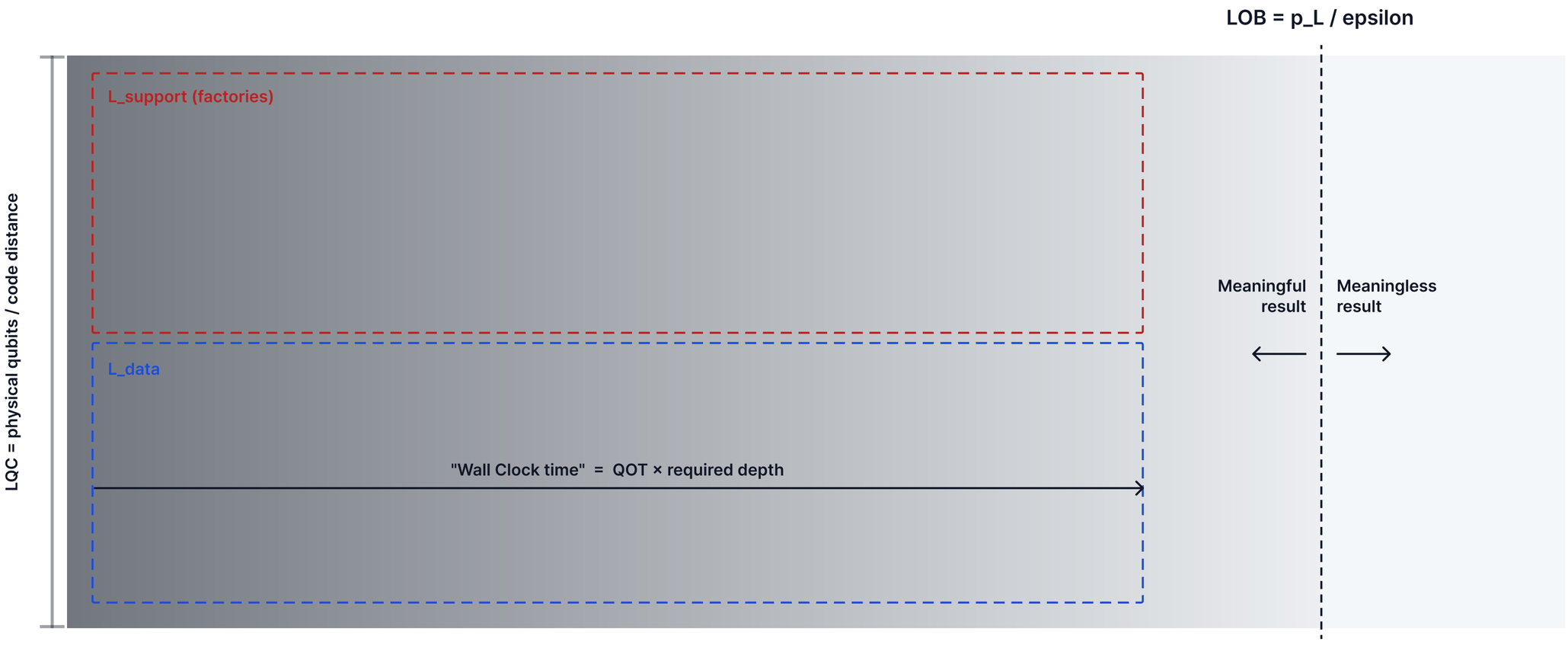
3 TIME CONSTRAINT

QOT × LOB ≤ 100 days

The total “wall clock time” (QOT x LOB) must be on a timescale that could be considered “cryptographically relevant. In this model, we define that scale as <100 days

APPENDIX E

A Simplified Model for Shor's Algorithm



Logical Compute Capacity (LCC) provided by your quantum computer — gradient defined by p_L

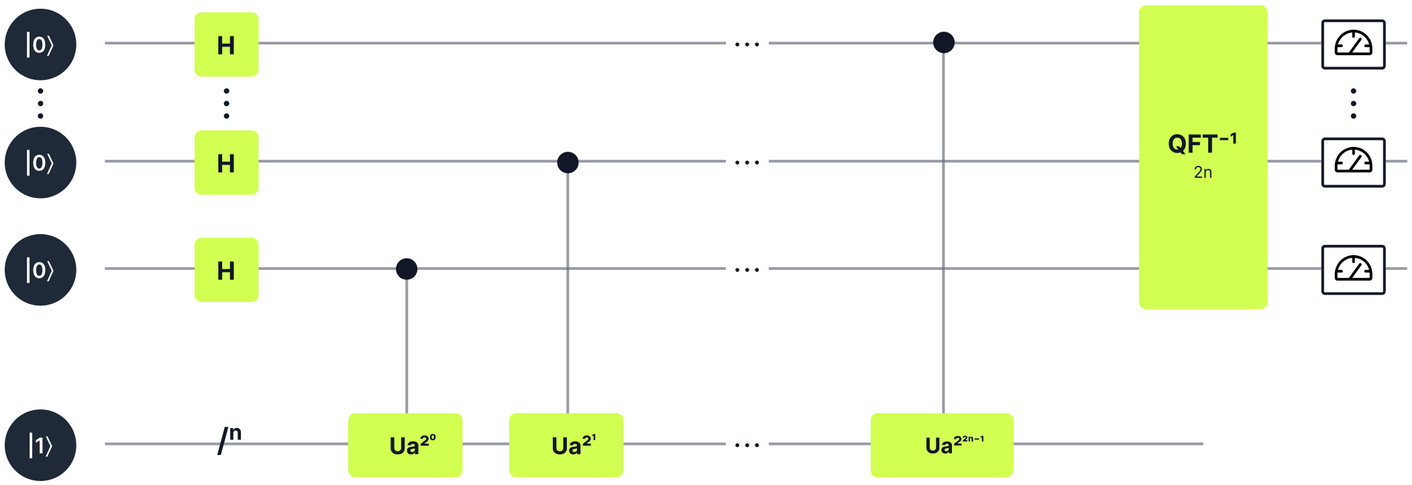
Algorithm Requirements (required width, depth, and factory overhead)

APPENDIX E

Reality

Quantum Phase Estimation

Shor's Algorithm Subroutine

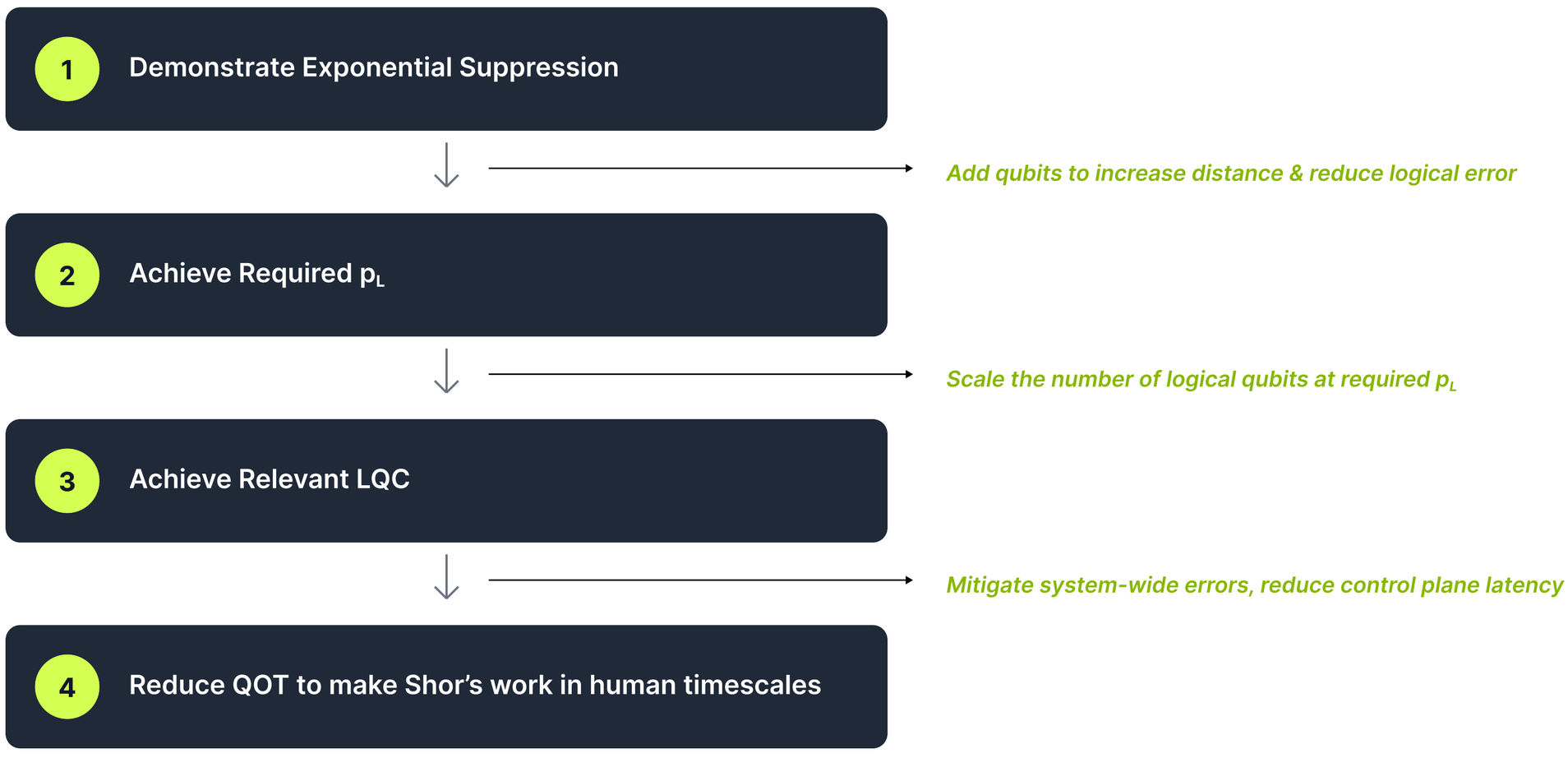


Reality

- The circuit consists of many heterogeneous operations
- Each operation has a different runtime, different overhead, different error rate
- The design of the algorithm is meant to minimize resources required for the **critical path**
- Pretty much analogous to modern semiconductor design

APPENDIX E

4 Steps to Fault-Tolerant Quantum Computing



APPENDIX E

Variable Definitions — Quantum Resources "Supply"

Experimental distance
the code distance used as a baseline.

Experimental logical error (p_L)
the logical error rate achieved at that distance

Qubits
the number of physical qubits available

Suppression Factor
the impact of increasing distance on the logical error rate at each step

C - constant
the constant factor penalty applied associated with with a given code

Gamma (γ)
the exponent "overhead" for the code. E.g. for surface code patches, **γ** is 2

Cycle Time
time required to complete one quantum error cycle (QOT)

APPENDIX E

Variable Definitions — Algorithm Overhead "Demand"

Failure Tolerance
the failure rate per run of a quantum algorithm

Factory Fraction
% of logical qubits dedicated to magic state production

Target Logical Error/Cycle
logical error rate

Routing Overhead
penalty applied to LQC to account for qubit routing reqs

Algorithm Reference Width
Minimum required width given by the algorithm

Algorithm Reference Volume
The total number of required operations for the given Shor's variant being targeted

Width Penalty
"effective" additional width required over the algorithm reference width to account for a variety of other failures/errors

APPENDIX E

Variable Relationships

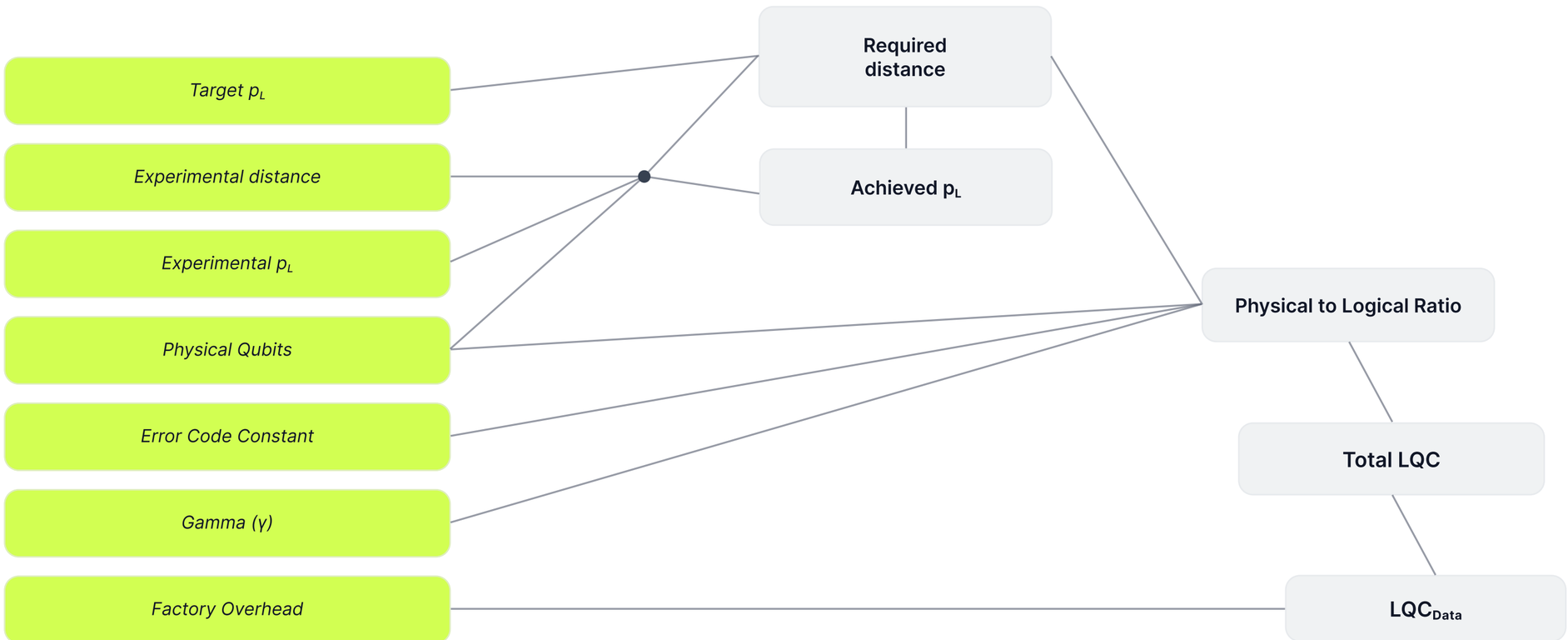
QUANTUM COMPUTING STACK				SIMPLIFIED CATEGORY · VARIABLE	
L1	L2	L3	L4	Category	Variable
■	■			● **Physical Qubit Quality**	Experimental logical error
■				● **Physical Qubit Count**	Qubits
■	■	■		● **Physical Qubit Quality**	Suppression factor (λ)
	■			● **Error Correction Eff.**	C — Constant
	■	■		● **Error Correction Eff.**	Gamma (γ)
	■	■		● **Control Overhead**	Factory fraction
	■	■		● **Control Overhead**	Routing overhead (≥1×)
		■		● **Control Overhead**	Width penalty β
			■	● **Algorithm Requirements**	Algorithm reference width
			■	● **Algorithm Requirements**	Algorithm reference volume

■ variable affects this layer

□ no impact on this layer

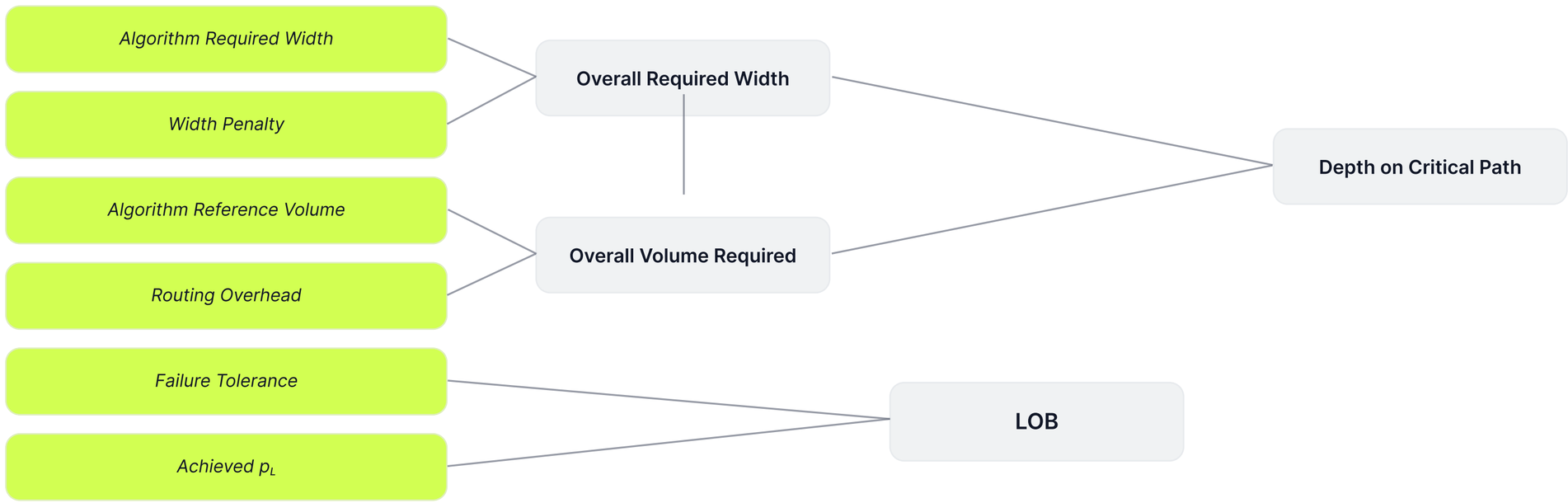
APPENDIX E

Model Mechanics — Supply Side



APPENDIX E

Model Mechanics — Demand Side



APPENDIX E

Notable Assumptions

01 Error Correction Anchored to Surface Codes

The physical-logical qubit overhead and error correction calculations are effectively anchored to surface codes for ease of modeling. Improvements stemming from more efficient codes like qLDPC codes are captured effectively by reducing the gamma (γ) parameter in the exponent

02 Cycle Time Fixed at 10 µs

Cycle time is 10 microseconds, roughly equivalent to superconducting qubits today. This is held as a constant; different scenarios could be considered with slower cycle times more in-line with modalities like neutral atom/trapped ions

03 Algorithm Target is RSA-2048 via Shor's

The algorithmic resources are based on RSA-2048. It’s entirely possible (in fact, likely) that elliptic curves require less width/depth, because Shor’s scales logarithmically based on key-length, and ECC keys are 256 bits (vs 2048 bits).

APPENDIX E

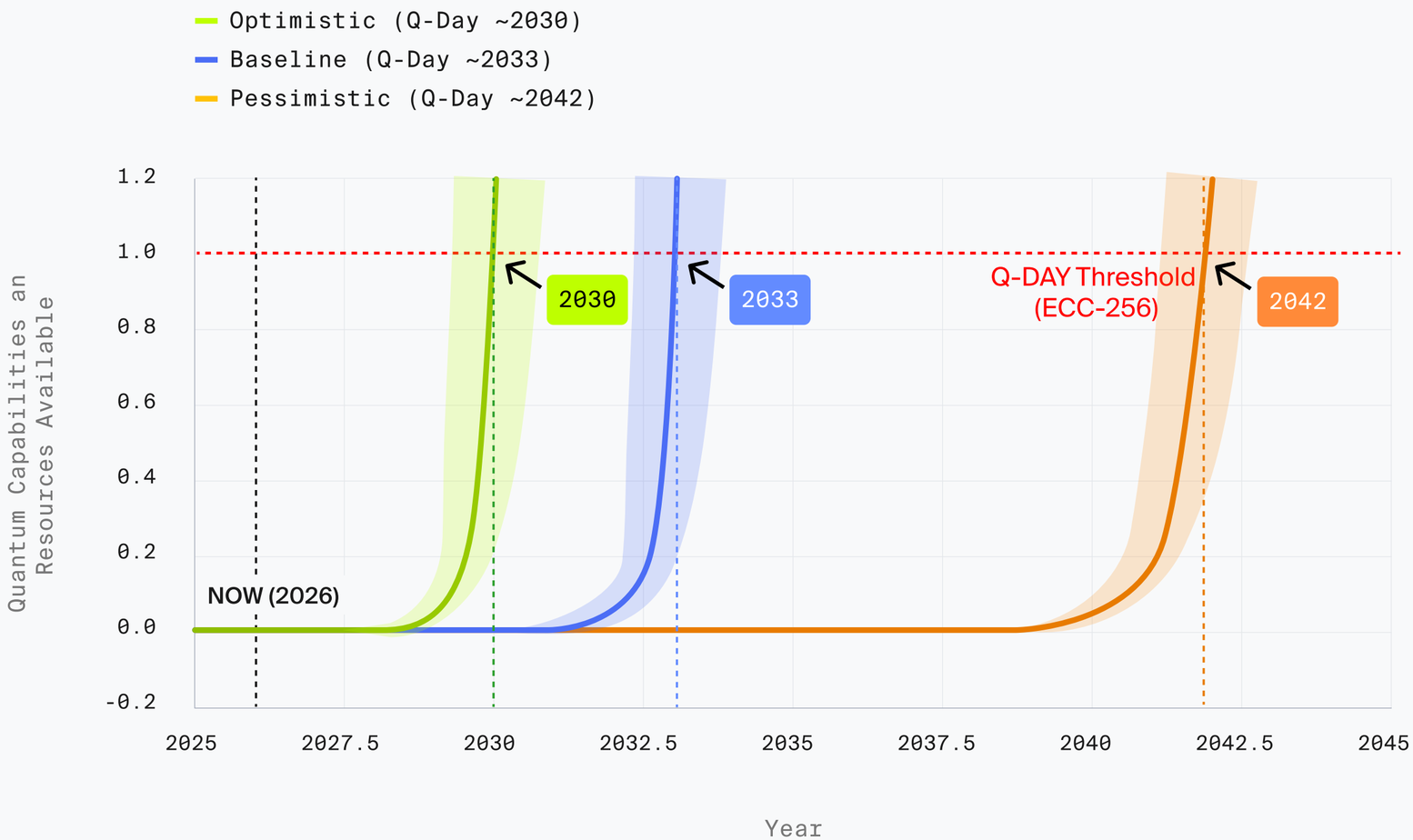
Variable Ranges

Variable	Bear		Base		Bull	
Experimental distance	7		7		7	
Experimental logical error	0.001	100%	0.001	125%	0.001	150%
Qubits	3,000	150%	3,000	200%	3,000	300%
Suppression factor (λ)	5	5%	7.5	10%	10	15%
C — Constant	10		10		10	
Gamma (γ)	2	−2%	2	−5%	2	−8%
Factory fraction	0.9	−2%	0.8	−5%	0.7	−8%
Routing overhead (≥1×)	3	−2%	3	−5%	3	−8%
Width penalty β	1.5	−2%	1.5	−5%	1.5	−8%
Algorithm reference width	1,000	0%	1,000	−5%	1,000	−10%
Algorithm reference volume	5×10^{12}	0%	5×10^{12}	−5%	5×10^{12}	−10%

APPENDIX E

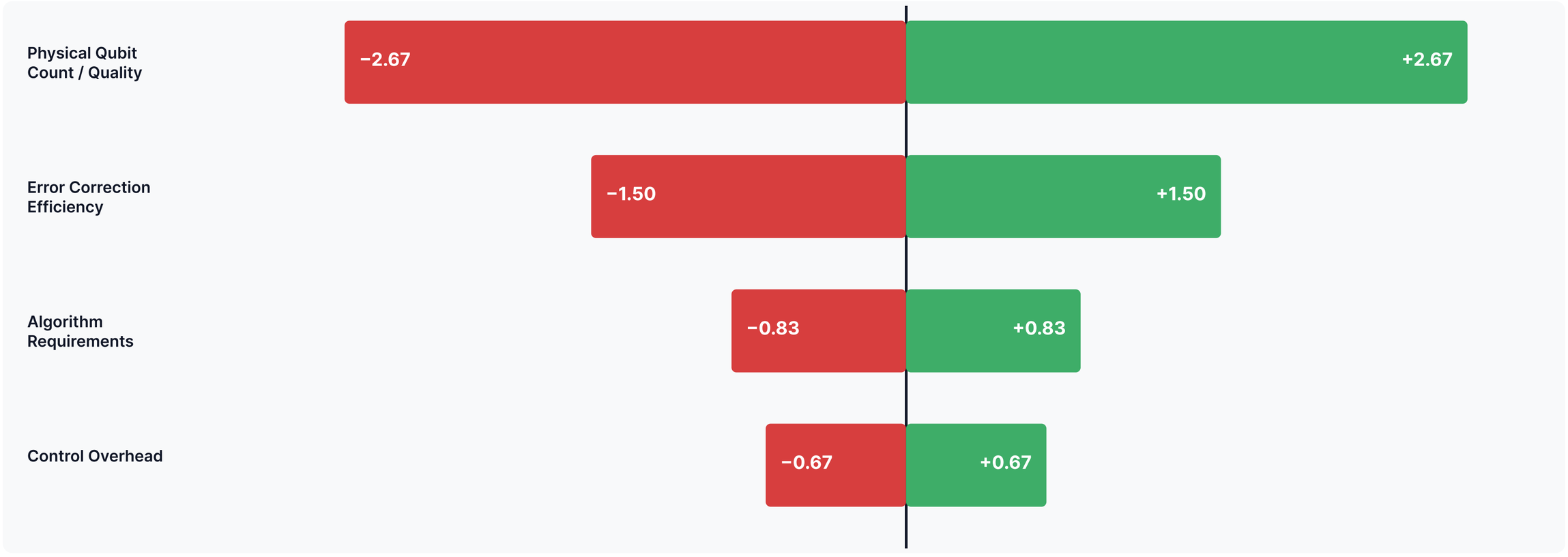
Model Output

PROJECT ELEVEN Q-DAY FORECAST



APPENDIX E

Sensitivity Analysis — Variable Layer Impact on Q-Day (yrs)



APPENDIX E

Model Idiosyncrasies

WHERE THE MODEL DEPARTS FROM PHYSICS

The point where the model probably breaks down from physics most clearly is the error suppression factor, α

The reason for that is that, in effect the model works “backwards” from what you would do in real life, which is define a distance and then calculate the logical error rate based on that. Instead, here we “hard code” the logical error rate based on the algorithm requirement (we hold it at 10^{-15}), and then calculate the distance based on that.

This massively inflates code distance relative to published resource estimates, so we crank up the suppression factor to compensate. Suppression factor is typically around 2 for recent Google, Quantinuum experiments. Our model assumes a range of 5–10 to start. Optimistic suppression factor estimates range to 25.

APPENDIX E

Key Takeaway — LQC is the Dominant Constraint

KEY TAKEAWAY 1 OF 3

Logical Qubit Capacity ≥ circuit width is the dominant constraint, assuming the threshold theorem holds and correlated errors do not dominate at scale

This is consistent with Gidney's 2025 resource estimate, and the Google Willow demonstration: once you have below-threshold error correction, you can effectively "buy" LOB by adding more physical qubits (increasing the distance of the code to lower the logical error rate).

The farther below threshold, the greater "headroom" for correlated errors at scale. So the physical error rate (defined by the physical qubit) and the threshold (defined by the error correction regime) still matter.

APPENDIX E

Key Takeaway — Reliability and Code Overhead

KEY TAKEAWAY 2 OF 3

Once below threshold, reliability becomes easier to scale, while code overhead becomes the main factor limiting LQC.

Surface codes are well studied, but they require a number of physical qubits that scale quadratically with distance. Newer qLDPC codes reduce that overhead making each physical qubit added count for more.

Better codes drive down the number of physical qubits needed to produce & connect, a clear example of a positive feedback loop.

APPENDIX E

Key Takeaway — Algorithm Design and Q-Day

KEY TAKEAWAY 3 OF 3

The algorithm design sets the “bar” for cryptographic relevance, and optimizations there potentially have a major impact on Q-Day timelines

Note that algorithm design is not independent from architecture design. In fact, this is a hardware/software co-design problem.

The fact that LQC is the bottleneck according to this model is largely a result of the algorithm. However, time–space tradeoffs are not “free”; below a certain width, the LOB tends to grow quickly.

APPENDIX E

Key Model Takeaways

POSTSCRIPT — DECADE BEYOND Q-DAY

Barring extremely fast cycle times, or massive algorithm optimizations, short-range attacks are going to be unfeasible for at least a decade beyond Q-Day

Cryptographic Relevance is Subjective

Cryptographic relevance is subjective, but the reality of the resource requirements for Shor's algorithm provide a lower bound required volume.

Even at Best-Case Hardware Speeds...

Even at superconducting-qubit speeds and LQC of over 10,000 qubits, the effective time it would take to run this attack is tens of minutes

APPENDIX E

Future Work

IMPROVEMENTS UNDER CONSIDERATION

01 **Refine input values**

Refinement of input values based on new/better data & trends

02 **Run a full Monte Carlo simulation**

A full “Monte Carlo” simulation with various parameters for a more robust sensitivity analysis

03 **Granular code-family breakdown**

A more granular breakdown and separate functions based on error correcting code type (surface codes vs. qLDPC)

CODE-FAMILY TRADE-OFFS (DETAIL ON ITEM 03)

Surface codes

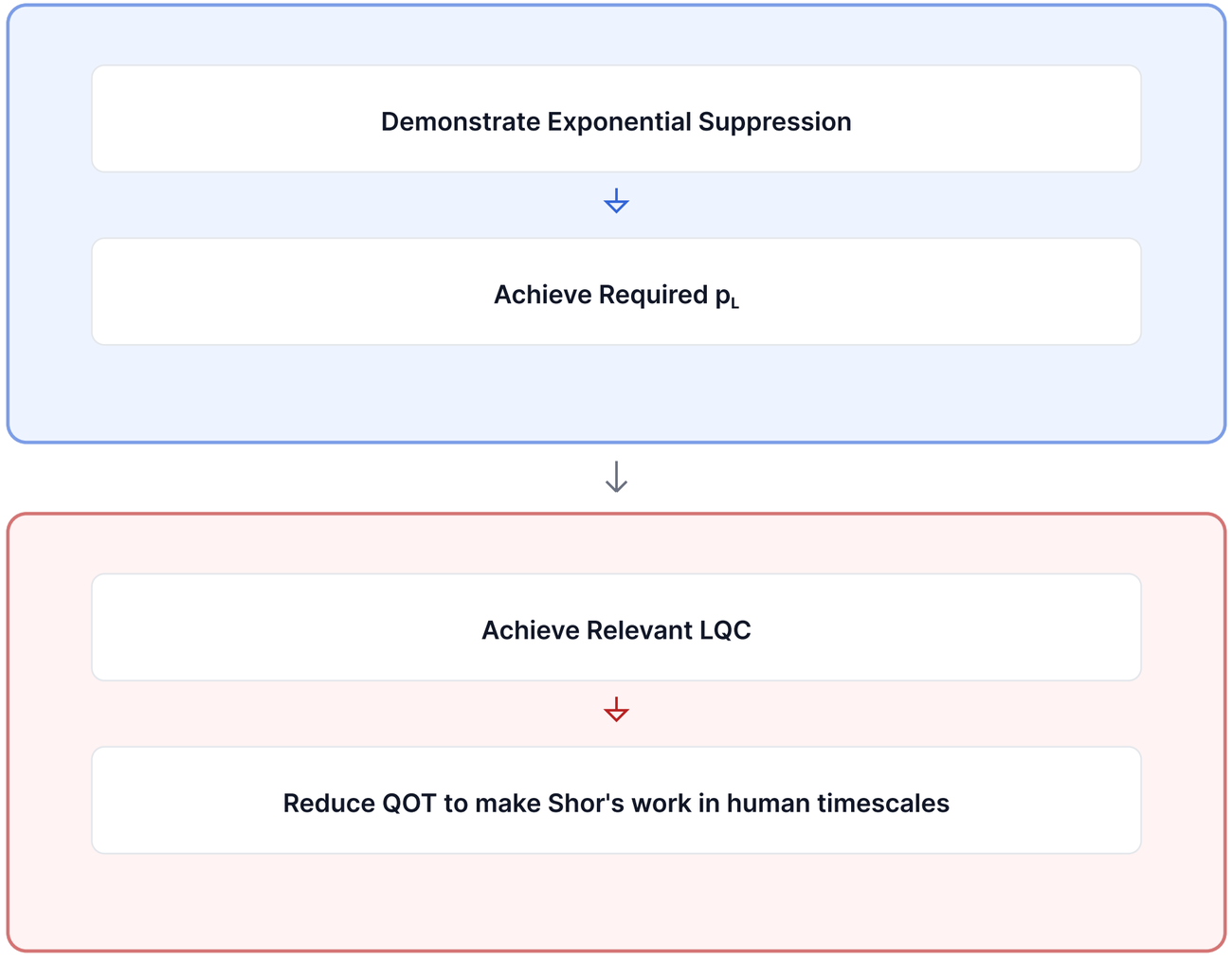
Surface codes have smaller constants, easier decoding, poor asymptotic scaling

qLDPC code families

qLDPC code families have larger constants, harder decoding, but much higher throughput (~k/n)

APPENDIX E

Postscript: How to tell if You're Bullish or Bearish Quantum



If you think this is the hard part...

You're bullish.

If you think this is the hard part...

You're bearish.

APPENDIX_ F

PQC Suite B: Faster Signatures with BLAKE3

A proposal to replace the internal hash functions in ML-DSA and SLH-DSA with BLAKE3—the fastest widely deployed cryptographic hash—delivering 20–30% signing and verification speed-ups with no change to key or signature sizes.

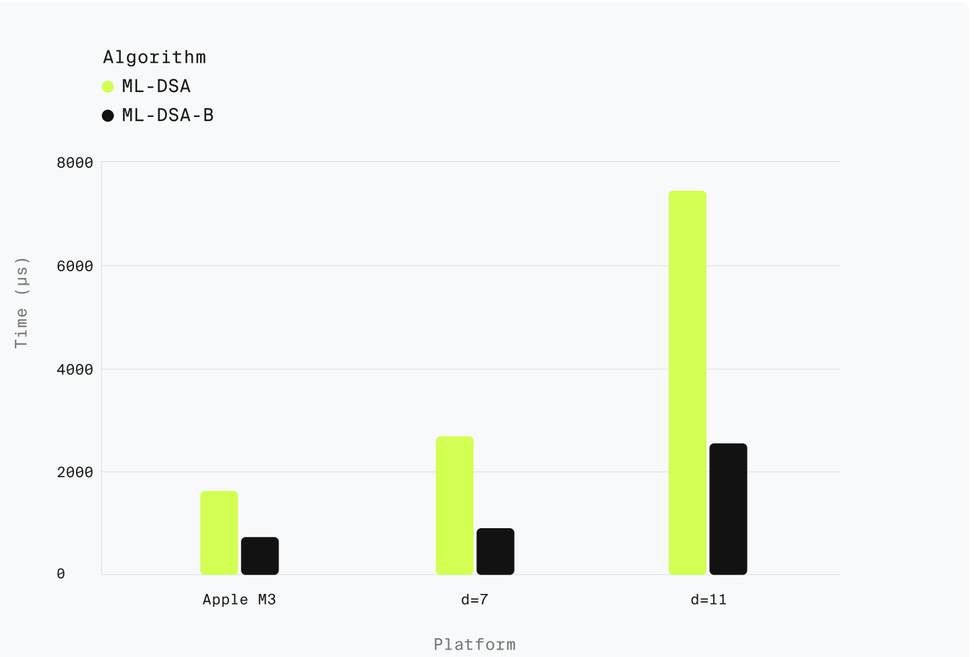
APPENDIX F

PQC Suite B: Faster Signatures with BLAKE3

Post-quantum signature schemes spend a substantial fraction of their runtime on hashing. ML-DSA uses SHAKE128 and SHAKE256; SLH-DSA offers variants built on SHA-256 or SHAKE. In both cases, the choice of hash function has a measurable impact on signing and verification performance.

PQC Suite-B [101] is a proposal, co-authored by JP Aumasson, Conor Deegan, Alex Pruden, and Zooko Wilcox-O'Hearn, to replace the internal hash functions in ML-DSA and SLH-DSA with BLAKE3, the fastest widely deployed cryptographic hash. The resulting schemes are called **ML-DSA-B** and **SLH-DSA-B.** BLAKE3 serves as a drop-in replacement for every hashing mode these schemes use.

SIGNATURE TIME FOR 1MB MESSAGES (LOWER = BETTER)



Experimental benchmarks modifying RustCrypto's implementations show meaningful speed-ups. For **ML-DSA-B**, message pre-hashing is up to 60 times faster, signing up to 20% faster, and verification up to 30% faster.

For **SLH-DSA-B**, the picture is more nuanced. SHAKE is consistently the slowest option, 4 to 7 times slower due to its higher per-bit hashing cost. BLAKE3 and SHA-2 perform in a similar range, with the winner determined by hardware: x86 platforms favor BLAKE3 (SIMD parallelism), while Apple silicon favors SHA-2 (dedicated hardware acceleration).

These results matter for blockchains. Every signature operation in a node's critical path, whether block validation, transaction relay, or consensus, is performed at scale and under latency pressure. A 20 to 30 percent verification improvement compounds across thousands of transactions per block. Importantly, PQC Suite-B does not change signature, public key, or private key sizes. It delivers a pure performance improvement with no trade-off against the size constraints that are already the central engineering challenge for post-quantum blockchains.

Citations

CITATIONS

Citations

[1] Google Quantum AI and Collaborators. Quantum error correction below the surface code threshold. Nature 638, 920–926 (2025). https://doi.org/10.1038/s41586-024-08449-y
[2] Babbush, Ryan, et al. "The Quantum Threat to Elliptic Curve Cryptocurrencies: Resource Estimates, Vulnerabilities, and Mitigations." Google Quantum AI, 25 Mar. 2026.
[3] Cain, Madelyn, et al. "Shor's Algorithm Is Possible with Approximately 10,000 Reconfigurable Atomic Qubits." Oratomic, 26 Mar. 2026.
[4] Shor, Peter W. "Algorithms for Quantum Computation: Discrete Logarithms and Factoring." Proceedings 35th Annual Symposium on Foundations of Computer Science, IEEE, 1994, pp. 124-134. DOI: 10.1109/SFCS.1994.365700.
[5] Grover, Lov K. 'A Fast Quantum Mechanical Algorithm for Database Search'. Proceedings of the Twenty-Eighth Annual ACM Symposium on Theory of Computing, Association for Computing Machinery, 1996, pp. 212–219, https://doi.org/10.1145/237814.237866. STOC '96.
[6] Gidney, Craig. "How to Factor 2048 Bit RSA Integers with Less than a Million Noisy Qubits." arXiv, 2025, arxiv.org/abs/2505.15917.
[7] Ivezic, Marin. "CRQC Quantum Capability Framework." Post-Quantum, postquantum.com/post-quantum/crqc-quantum-capability-framework/#75-how-each-capability-moves-lqc-lob-and-qot. Accessed 30 Mar. 2026.
[8] "Quantum Benchmark Zoo." Quantumbenchmarkzoo.org, 2021, quantumbenchmarkzoo.org/content/system-level-benchmark/others/clops.
[9] Mosca, Michele, and Marco Piani. Quantum Threat Timeline Report 2025. evolutionQ Inc. and Global Risk Institute, Mar. 2026. Global Risk Institute, www.globalriskinstitute.org.
[10] Webster, P., et al. "The Pinnacle Architecture: Reducing the Cost of Breaking RSA-2048 to 100,000 Physical Qubits Using Quantum LDPC Codes." arXiv, 2026, arxiv.org/abs/2602.11457.
[11] Ivezic, Marin. "No, the "Pinnacle Architecture" Is Not Bringing Q-Day Closer 2-5 Years (but It Is Credible Research)." PostQuantum - Quantum Computing, Quantum Security, PQC, 13 Feb. 2026, postquantum.com/security-pqc/pinnacle-architecture-q-day/. Accessed 30 Mar. 2026.
[12] Mundada, Pranav S., et al. "Heterogeneous Architectures Enable a 138x Reduction in Physical Qubit Requirements for Fault-Tolerant Quantum Computing under Detailed Accounting." arXiv, 13 Apr. 2026.
[13] Gu, Andi, et al. "Scalable Neural Decoders for Practical Fault-Tolerant Quantum Computation." arXiv, 9 Apr. 2026.
[14] Huang, H., et al. "Optimized Quantum Circuits for Ed25519 Elliptic Curve Discrete Logarithm." Quantum Information Processing, vol. 24, 2025.
[15] Regev, Oded. "An Efficient Quantum Factoring Algorithm." arXiv, 2024, arxiv.org/abs/2308.06572.
[16] Deegan, C. "Quantum Vulnerability of Bitcoin Addresses." Project Eleven Blog, July 2025, blog.projecteleven.com/posts/quantum-vulnerability-of-bitcoin-addresses.
[17] Deegan, C. "HD Wallets & Quantum Risk: Does Reusing One Address Endanger the Rest?" Project Eleven Blog, Aug. 2025, blog.projecteleven.com/posts/hd-wallets--quantum-risk-does-reusing-one-address-endanger-the-rest.
[18] "Bitcoin Risq List." Project Eleven, www.projecteleven.com/bitcoin-risq-list.
[19] "Quantum Attack Vectors in Ethereum." Project Eleven Blog, blog.projecteleven.com/posts/quantum-attack-vectors-in-ethereum.
[20] "Quantum Risk to the Ethereum Blockchain." Deloitte Netherlands, 2025, www.deloitte.com/nl/en/services/consulting-risk/perspectives/quantum-risk-to-the-ethereum-blockchain.html.
[21] "How Blockchains Will Evolve for the Quantum Era — Part 2." Fireblocks, July 2025, www.fireblocks.com/blog/how-blockchains-will-evolve-for-the-quantum-era.
[22] "Stablecoin Market Tops $317 Billion." MEXC News, Jan. 2026, www.mexc.co/news/421705; "Circle's USDC Outpaces Growth of Tether's USDT for Second Year Running." CoinDesk, Jan. 2026, www.coindesk.com/markets/2026/01/06/circle-s-usdc-outpaces-growth-of-tether-s-usdt-for-second-year-running.
[23] "Stablecoin Market Share and Transaction Volume — September 2025 Data." CoinLedger, coinledger.io/research/stablecoin-market-share-and-transaction-volume.
[24] "Stablecoin Market Tops $317 Billion." MEXC News, Jan. 2026, www.mexc.co/news/421705; "Circle's USDC Outpaces Growth of Tether's USDT for Second Year Running." CoinDesk, Jan. 2026, www.coindesk.com/markets/2026/01/06/circle-s-usdc-outpaces-growth-of-tether-s-usdt-for-second-year-running.
[25] Kearney, J., and C. Deegan. "Vulnerabilities of Stablecoins to Quantum Attacks." Project Eleven Blog, July 2025, blog.projecteleven.com/posts/vulnerabilities-of-stablecoins-to-quantum-attacks
[26] Deegan, C., and J. Kearney. "Quantum vs. USDC: A Threat Analysis of Circle's Smart-Contract." Project Eleven Blog, July 2025, blog.projecteleven.com/posts/quantum-vs-usdc-a-threat-analysis-of-circles-smart-contract.
[27] "FIPS 203: Module-Lattice-Based Key-Encapsulation Mechanism Standard." National Institute of Standards and Technology (NIST), csrc.nist.gov/pubs/fips/203/final.
[28] "FIPS 204: Module-Lattice-Based Digital Signature Standard." National Institute of Standards and Technology (NIST), csrc.nist.gov/pubs/fips/204/final.
[29] "FIPS 205: Stateless Hash-Based Digital Signature Standard." National Institute of Standards and Technology (NIST), csrc.nist.gov/pubs/fips/205/final.
[30] "Hybrid Key Exchange in TLS 1.3." Internet Engineering Task Force (IETF), datatracker.ietf.org/doc/draft-ietf-tls-hybrid-design/.
[31] "PQ 2025." Cloudflare Blog, blog.cloudflare.com/pq-2025/.
[32] "Radar 2025 Year in Review." Cloudflare Blog, blog.cloudflare.com/radar-2025-year-in-review/.
[33] "A New Path for Kyber on the Web." Google Security Blog, Sept. 2024, security.googleblog.com/2024/09/a-new-path-for-kyber-on-web.html.
[34] "Enhancing Your App's Privacy and Security with Quantum-Secure Workflows." Apple Developer Documentation, developer.apple.com/documentation/cryptokit/enhancing-your-app-s-privacy-and-security-with-quantum-secure-workflows.
[35] "Post-Quantum Cryptography." OpenSSH, www.openssh.org/pq.html.
[36] "Post-Quantum Cryptography." Amazon Web Services (AWS), aws.amazon.com/security/post-quantum-cryptography/.
[37] "Announcing Quantum-Safe Key Encapsulation Mechanisms in Cloud KMS." Google Cloud Blog, cloud.google.com/blog/products/identity-security/announcing-quantum-safe-key-encapsulation-mechanisms-in-cloud-kms.
[38] Deegan, C. "Benchmarking Post-Quantum Cryptography." GitHub, github.com/conor-deegan/benching-pq.
[39] Beast, H. "BIP 360." BIP 360, bip360.org/.
[40] Lopp, J. "QBIP." QBIP, qbip.org/.
[41] "Lean Roadmap." Ethereum, leanroadmap.org/.
[42] Strawmap, strawmap.org/.
[43] Deegan, C. "Post-Quantum Readiness for EdDSA Chains and a Possible Solution for Some ECDSA Wallets — Part 1." Project Eleven Blog, Aug. 2025, blog.projecteleven.com/posts/post-quantum-readiness-for-eddsa-chains-and-a-possible-solution-for-some-ecdsa-wallets---part-1.
[44] Baldimtsi, M., et al. "Post-Quantum Readiness in EdDSA Chains." IACR Cryptology ePrint Archive, 2025, eprint.iacr.org/2025/1368.pdf.
[45] "Post-Quantum Cryptography for Engineers." Internet Engineering Task Force (IETF), datatracker.ietf.org/doc/draft-ietf-pquip-pqc-engineers/.
[46] Campbell. "Enterprise Migration to Post-Quantum Cryptography." MDPI, www.mdpi.com/2073-431X/15/1/9.
[47] "CNSA 2.0 Algorithms." National Security Agency (NSA), May 2025, media.defense.gov/2025/May/30/2003728741/-1/-1/0/CSA_CNSA_2.0_ALGORITHMS.PDF.
[48] "PQC Migration Timelines." National Cyber Security Centre (NCSC), www.ncsc.gov.uk/guidance/pqc-migration-timelines.
[49] "IR 8547: Transition to Post-Quantum Cryptography Standards." National Institute of Standards and Technology (NIST), csrc.nist.gov/pubs/ir/8547/ipd.
[50] Pont, Jamie J., et al. 'Downtime Required for Bitcoin Quantum-Safety'. arXiv [Quant-Ph], 2024, arxiv.org/abs/2410.16965. arXiv.
[51] Beauregard, Stephane. "Circuit for Shor's Algorithm Using 2n+3 Qubits." arXiv, 2003, arxiv.org/abs/quant-ph/0205095.
[52] Zalka, Christof. "Fast Versions of Shor's Quantum Factoring Algorithm." arXiv, 1998, arxiv.org/abs/quant-ph/9806084.
[53] Zalka, Christof. "Shor's Algorithm with Fewer (Pure) Qubits." arXiv, 2006, arxiv.org/abs/quant-ph/0601097.
[54] Ekerå, Martin, and Johan Håstad. "Quantum Algorithms for Computing Short Discrete Logarithms and Factoring RSA Integers." IACR Cryptology ePrint Archive, 2017, eprint.iacr.org/2017/077.
[55] Gidney, Craig, and Martin Ekerå. "How to Factor 2048 Bit RSA Integers in 8 Hours Using 20 Million Noisy Qubits." Quantum, vol. 5, Apr. 2021, p. 433. DOI: 10.22331/q-2021-04-15-433.
[56] Gouzien, Élie, and Nicolas Sangouard. "Factoring 2048-Bit RSA Integers in 177 Days with 13,436 Qubits and a Multimode Memory." Physical Review Letters, vol. 127, Sept. 2021, p. 140503. DOI: 10.1103/PhysRevLett.127.140503.

Citations

[57] Litinski, Daniel. "How to Compute a 256-Bit Elliptic Curve Private Key with Only 50 Million Toffoli Gates." arXiv, 2023, arxiv.org/abs/2306.08585.
[58] Fowler, Austin G., et al. 'Surface Codes: Towards Practical Large-Scale Quantum Computation'. Phys. Rev. A, vol. 86, American Physical Society, Sept. 2012, p. 032324, https://doi.org/10.1103/PhysRevA.86.032324.
[59] McEwen, Matt, et al. 'Resolving Catastrophic Error Bursts from Cosmic Rays in Large Arrays of Superconducting Qubits'. Nature Physics, vol. 18, no. 1, Springer Science and Business Media LLC, Dec. 2021, pp. 107–111, https://doi.org/10.1038/s41567-021-01432-8.
[60] Koch, Jens, et al. 'Charge-Insensitive Qubit Design Derived from the Cooper Pair Box'. Phys. Rev. A, vol. 76, American Physical Society, Oct. 2007, p. 042319, https://doi.org/10.1103/PhysRevA.76.042319.
[61] Ke, Chung-Ting, et al. 'Scaffold-Assisted Window Junctions for Superconducting Qubit Fabrication'. arXiv [Physics.App-Ph], 2025, arxiv.org/abs/2503.11010. arXiv.
[62] Chamberland, Christopher, et al. 'Topological and Subsystem Codes on Low-Degree Graphs with Flag Qubits'. Phys. Rev. X, vol. 10, American Physical Society, Jan. 2020, p. 011022, https://doi.org/10.1103/PhysRevX.10.011022.
[63] Panteleev, Pavel, and Gleb Kalachev. 'Asymptotically Good Quantum and Locally Testable Classical LDPC Codes'. arXiv [Cs.IT], 2022, arxiv.org/abs/2111.03654. arXiv.
[64] Bravyi, Sergey, et al. "High-Threshold and Low-Overhead Fault-Tolerant Quantum Memory." Nature, vol. 627, no. 8005, 1 Mar. 2024, pp. 778–782, www.nature.com/articles/s41586-024-07107-7, https://doi.org/10.1038/s41586-024-07107-7. Accessed 26 Apr. 2024.
[65] Zhu, Guanyu, et al. 'Non-Abelian qLDPC: TQFT Formalism, Addressable Gauging Measurement and Application to Magic State Fountain on 2D Product Codes'. arXiv [Quant-Ph], 2026, arxiv.org/abs/2601.06736. arXiv.
[66] Dolev Bluvstein, et al. "A Fault-Tolerant Neutral-Atom Architecture for Universal Quantum Computation." Nature, 10 Nov. 2025, www.nature.com/articles/s41586-025-09848-5, https://doi.org/10.1038/s41586-025-09848-5.
[67] Saffman, M., et al. 'Quantum Information with Rydberg Atoms'. Rev. Mod. Phys., vol. 82, American Physical Society, Aug. 2010, pp. 2313–2363, https://doi.org/10.1103/RevModPhys.82.2313.
[68] Chiu, Neng-Chun, et al. "Continuous Operation of a Coherent 3,000-Qubit System." Nature, 15 Sept. 2025, pp. 1–3, www.nature.com/articles/s41586-025-09596-6, https://doi.org/10.1038/s41586-025-09596-6.
[69] Wang, Yuqing, et al. 'Direct Generation of an Array with 78400 Optical Tweezers Using a Single Metasurface'. Chinese Physics Letters, vol. 43, no. 1, IOP Publishing, Dec. 2025, p. 010606, https://doi.org/10.1088/0256-307x/43/1/010606.
[70] Rines, Rich, et al. 'Demonstration of a Logical Architecture Uniting Motion and In-Place Entanglement: Shor's Algorithm, Constant-Depth CNOT Ladder, and Many-Hypercube Code'. arXiv [Quant-Ph], 2025, arxiv.org/abs/2509.13247. arXiv.
[71] Rodriguez, Pedro Sales, et al. "Experimental Demonstration of Logical Magic State Distillation." Nature, vol. 645, no. 8081, 14 July 2025, pp. 620–625, https://doi.org/10.1038/s41586-025-09367-3.
[72] Zhou, Hengyun, et al. "Low-Overhead Transversal Fault Tolerance for Universal Quantum Computation." Nature, vol. 646, no. 8084, 24 Sept. 2025, pp. 303–308, www.nature.com/articles/s41586-025-09543-5, https://doi.org/10.1038/s41586-025-09543-5. Accessed 30 Mar. 2026.
[73] Pino, J M et al. "Demonstration of the trapped-ion quantum CCD computer architecture." Nature vol. 592,7853 (2021): 209-213. doi:10.1038/s41586-021-03318-4
[74] Sakrejda, Tomasz P., et al. 'Efficient Sympathetic Cooling in Mixed Barium and Ytterbium Ion Chains'. arXiv [Physics.Atom-Ph], 2021, arxiv.org/abs/1809.00240. arXiv.
[75] Bartolucci, Sara, et al. "Fusion-Based Quantum Computation." Nature Communications, vol. 14, no. 1, 17 Feb. 2023, https://doi.org/10.1038/s41467-023-36493-1.
[76] Garn, Michael, and Angus Kan. 'Quantum Resource Estimates for Computing Binary Elliptic Curve Discrete Logarithms'. IEEE Transactions on Quantum Engineering, vol. 6, Institute of Electrical and Electronics Engineers (IEEE), 2025, pp. 1–23, https://doi.org/10.1109/tqe.2025.3586541.
[77] Charbon, E., et al. 'Cryo-CMOS for Quantum Computing'. 2016 IEEE International Electron Devices Meeting (IEDM), 2016, p. 13.5.1-13.5.4, https://doi.org/10.1109/IEDM.2016.7838410.
[78] Hughes, A. C., et al. "Trapped-Ion Two-Qubit Gates with >99.99% Fidelity without Ground-State Cooling." arXiv, 2025, arxiv.org/abs/2510.17286.
[79] Blume-Kohout, Robin, et al. 'Quantum Characterization, Verification, and Validation'. arXiv [Quant-Ph], 2025, arxiv.org/abs/2503.16383. arXiv.
[80] Nielsen, Erik, et al. 'Gate Set Tomography'. Quantum, vol. 5, Verein zur Förderung des Open Access Publizierens in den Quantenwissenschaften, Oct. 2021, p. 557, https://doi.org/10.22331/q-2021-10-05-557.
[81] Magesan, Easwar, et al. 'Characterizing Quantum Gates via Randomized Benchmarking'. Phys. Rev. A, vol. 85, American Physical Society, Apr. 2012, p. 042311, https://doi.org/10.1103/PhysRevA.85.042311.
[82] Cross, Andrew W., et al. 'Validating Quantum Computers Using Randomized Model Circuits'. Physical Review A, vol. 100, no. 3, American Physical Society (APS), Sept. 2019, https://doi.org/10.1103/physreva.100.032328.
[83] Temme, Kristan, et al. 'Error Mitigation for Short-Depth Quantum Circuits'. Phys. Rev. Lett., vol. 119, American Physical Society, Nov. 2017, p. 180509, https://doi.org/10.1103/PhysRevLett.119.180509.
[84] Rosenfeld, Emma, et al. 'Magic State Cultivation on a Superconducting Quantum Processor'. arXiv [Quant-Ph], 2025, arxiv.org/abs/2512.13908. arXiv.
[85] Menon, Varun, et al. 'Magic Tricycles: Efficient Magic State Generation with Finite Block-Length Quantum LDPC Codes'. arXiv [Quant-Ph], 2025, arxiv.org/abs/2508.10714. arXiv.
[86] Breuer, Heinz-Peter, and Francesco Petruccione. The Theory of Open Quantum Systems. Oxford UP, 2006. DOI: 10.1093/acprof:oso/9780199213900.001.0001.
[87] Gardiner, C. W., and P. Zoller. Quantum Noise: A Handbook of Markovian and Non-Markovian Quantum Stochastic Methods with Applications to Quantum Optics. Springer, 2000.
[88] Leggett, A. J., et al. "Dynamics of the Dissipative Two-State System." Reviews of Modern Physics, vol. 59, Jan. 1987, pp. 1-85. DOI: 10.1103/RevModPhys.59.1.
[89] Gottesman, Daniel. "Stabilizer Codes and Quantum Error Correction." arXiv, 1997, arxiv.org/abs/quant-ph/9705052.
[90] Knill, Emanuel, and Raymond Laflamme. "Theory of Quantum Error-Correcting Codes." Physical Review A, vol. 55, Feb. 1997, pp. 900-911. DOI: 10.1103/PhysRevA.55.900.
[91] Shor, Peter W. "Scheme for Reducing Decoherence in Quantum Computer Memory." Physical Review A, vol. 52, Oct. 1995, pp. R2493-R2496. DOI: 10.1103/PhysRevA.52.R2493.
[92] Calderbank, A. R., and Peter W. Shor. "Good Quantum Error-Correcting Codes Exist." Physical Review A, vol. 54, Aug. 1996, pp. 1098-1105. DOI: 10.1103/PhysRevA.54.1098.
[93] Kitaev, A. Yu. "Fault-Tolerant Quantum Computation by Anyons." Annals of Physics, vol. 303, no. 1, Jan. 2003, pp. 2-30. DOI: 10.1016/s0003-4916(02)00018-0.
[94] Fowler, Austin G., et al. "Surface Codes: Towards Practical Large-Scale Quantum Computation." Physical Review A, vol. 86, Sept. 2012, p. 032324. DOI: 10.1103/PhysRevA.86.032324.
[95] Gidney, C., et al. "Yoked Surface Codes." Nature Communications, vol. 16, no. 4498, 2025. DOI: 10.1038/s41467-025-59714-1.
[96] Eickbusch, A., et al. "Demonstration of Dynamic Surface Codes." Nature Physics, vol. 21, 2025, pp. 1994-2001. DOI: 10.1038/s41567-025-03070-w.
[97] Eastin, Bryan, and Emanuel Knill. "Restrictions on Transversal Encoded Quantum Gate Sets." Physical Review Letters, vol. 102, Mar. 2009, p. 110502. DOI: 10.1103/PhysRevLett.102.110502.
[98] Bravyi, Sergey, and Alexei Kitaev. "Universal Quantum Computation with Ideal Clifford Gates and Noisy Ancillas." Physical Review A, vol. 71, Feb. 2005, p. 022316. DOI: 10.1103/PhysRevA.71.022316.
[99] Horsman, Dominic, et al. "Surface Code Quantum Computing by Lattice Surgery." New Journal of Physics, vol. 14, no. 12, Dec. 2012, p. 123011. DOI: 10.1088/1367-2630/14/12/123011.
[100] Tillich, Jean-Pierre, and Gilles Zemor. "Quantum LDPC Codes With Positive Rate and Minimum Distance Proportional to the Square Root of the Blocklength." IEEE Transactions on Information Theory, vol. 60, no. 2, Feb. 2014, pp. 1193-1202. DOI: 10.1109/tit.2013.2292061.
[101] "PQC Suite-B." GitHub, github.com/PQC-Suite-B/.
[102] Ransford, Anthony, et al. "Helios: A 98-Qubit Trapped-Ion Quantum Computer." arXiv, 2025, arxiv.org/abs/2511.05465.
[103] Mosca, M., and M. Piani. Quantum Threat Timeline Report 2023. Global Risk Institute and evolutionQ Inc., 2023.

וויא

PROJECT ELEVEN