Superconducting Quantum Computing for Electrical Engineers

From Josephson Junctions to Qubit Readout: A Signal Processing Perspective

Hans Johnson
SQMS & Illinois Institute of Technology

March 3, 2026

Abstract

This document provides those with an electrical engineering background a comprehensive introduction to quantum computing, with emphasis on superconducting quantum computing, in the context of the RF and signal processing aspects that dominate practical implementations. The transmon qubit, the basis of quantum computation in the superconducting regime, is presented as a nonlinear LC oscillator; dispersive readout is explained as an RF measurement problem; and the signal processing challenges that limit current quantum computer performance are described. The treatment prioritizes intuition and EE analogies while providing mathematical foundations for readers seeking deeper understanding. A comprehensive survey of qubit modalities (superconducting, trapped ion, silicon spin, neutral atom, and photonic) compares performance metrics, infrastructure requirements, and fabrication approaches. Adaptive filtering techniques for qubit readout optimization are contextualized within the broader goal of fault-tolerant quantum computation.

Contents

1 Introduction: Why Should an Electrical Engineer Care About Quantum Computing?

Imagine running a Fast Fourier Transform on a dataset so large that classical computers would take longer than the age of the universe to complete it, yet a quantum computer could solve it in seconds. Imagine EDA tools that can optimize billion-transistor chip layouts by exploring all possible configurations simultaneously, rather than relying on heuristics. Imagine protein folding simulations that unlock CRISPR-based cancer therapies, or physics simulations that reveal new materials for room-temperature superconductors. These are not science fiction; they are the promises of fault-tolerant quantum computing, and they are closer than most people realize.

The catch? We are not there yet. Today's quantum computers are fragile, error-prone, and limited to a few hundred qubits. The path from laboratory curiosities to world-changing machines requires solving numerous engineering challenges, and many of these challenges are fundamentally **electrical engineering problems**: RF pulse generation with sub-nanosecond timing, cryogenic low-noise amplification at the quantum limit, high-speed digital signal processing for real-time error correction, and feedback control systems that operate faster than qubits can decohere. This is where EEs come in. The quantum computing revolution will not be led by physicists alone; it needs engineers who understand signal integrity, noise floors, and system integration. The field is wide open, and the opportunities for impact are immense.

The Core Insight for EEs: A superconducting qubit is essentially a **nonlinear LC oscillator** operating at microwave frequencies ($\sim$5 GHz) and cryogenic temperatures ($\sim$20 mK). The "quantum" behavior emerges from the circuit's nonlinearity, which creates uneven energy level spacing. Everything else (control, readout, error correction) is RF engineering and digital signal processing.

This document presents quantum computing from an EE perspective, translating quantum mechanical concepts into familiar circuit and signal processing terminology. The goal is to provide sufficient background for EE researchers to contribute meaningfully to quantum computing hardware development without requiring an extensive physics background.

1.1 The Quantum Computing Stack

A superconducting quantum computer consists of several layers, most of which involve traditional EE disciplines (Table I).

Table I: The Quantum Computing Stack

Layer	**Function**	**EE Discipline**
Quantum Processor	Josephson junctions at 20 mK	Microwave circuit design, cryogenics
Control Electronics	Precise microwave pulses	RF/microwave engineering, DACs
Readout Chain	Amplify/digitize signals	Low-noise amplifiers, ADCs
Signal Processing	State discrimination	DSP, optimal filtering, ML
Real-time Control	Feedback for QEC	FPGA design, HDL

This document focuses on the physics necessary to understand the signal processing layer, with particular attention to the readout discrimination problem.

1.2 Where Does Quantum Speedup Come From?

Before diving into hardware details, it is worth understanding *why* quantum computers can outperform classical computers for certain problems. The answer is subtle and often misunderstood.

1.2.1 The Naive (Wrong) Explanation

A common misconception is: "A qubit can be 0 and 1 simultaneously, so n qubits can represent 2^n states at once, giving exponential parallelism." This is misleading because:

- While n qubits can exist in a superposition of 2^n basis states, **measurement only returns one outcome**
- If superposition alone provided speedup, we could just flip n coins and call it parallel computation
- The speedup comes from something more subtle: **interference**

Why the Misconception Seems Plausible The naive explanation feels right because quantum operations are **linear**. If you apply a function f to a superposition, you get the superposition of the outputs:

$$f(|0\rangle + |1\rangle) = f(|0\rangle) + f(|1\rangle)$$

This makes qubits look like "variables" in a program that can hold multiple values at once. If f is expensive to compute classically, it seems like we could evaluate it on all inputs in parallel. This is exactly what happens during the computation. The problem is that when we measure, we get *one* random output, not all of them. The "parallel computation" did happen, but we can only see one result. Without interference to bias which result we see, we have gained nothing over random guessing.

1.2.2 The Correct Explanation: Interference

Quantum algorithms exploit the wave-like nature of quantum states. In a superposition:

$$|\psi\rangle = \sum_{x=0}^{2^n-1} \alpha_x |x\rangle \tag{1}$$

Variable Definitions for Eq. (1):
$|\psi\rangle$ — quantum state of n qubits
$|x\rangle$ — computational basis state (e.g., $|0101...\rangle$ representing binary number x)
α_x — complex amplitude for state $|x\rangle$; probability of measuring x is $|\alpha_x|^2$
n — number of qubits; 2^n possible basis states

the amplitudes α_x are complex numbers that can interfere constructively or destructively when the quantum state evolves. A well-designed quantum algorithm:

1. **Prepares a superposition** over all possible inputs (easy; a few Hadamard gates)
2. **Applies operations** that encode the problem structure into the amplitudes
3. **Engineers interference** so that "wrong" answers have amplitudes that cancel out, and "right" answers have amplitudes that add up
4. **Measures** to obtain a right answer with high probability

EE Analogy: Think of a phased array antenna. Each element radiates in all directions (like a superposition), but the relative phases cause constructive interference in one direction and destructive interference in others. The array doesn't "try all directions at once"; it uses interference to concentrate energy where you want it. Quantum algorithms work similarly: they use interference to concentrate probability amplitude on correct answers.

Music Analogy: When two guitarists play the same note, the sound waves can reinforce each other (louder) or cancel out (quieter) depending on whether the waves are "in sync" or "out of sync." Quantum computers use this same wave-like behavior to make wrong answers cancel out and right answers get louder.

How Interference Is Engineered The key insight is that quantum gates manipulate *phases*, not just probabilities. Consider a simple example with two paths to the same answer:

- **Path A** contributes amplitude $+0.5$ to answer $|x\rangle$
- **Path B** contributes amplitude $+0.5$ to answer $|x\rangle$
- Total amplitude: $0.5 + 0.5 = 1.0$, so probability $= |1.0|^2 = 1$ (certain)

Now consider a wrong answer where the phases differ:

- **Path A** contributes amplitude $+0.5$ to answer $|y\rangle$
- **Path B** contributes amplitude -0.5 to answer $|y\rangle$
- Total amplitude: $0.5 + (-0.5) = 0$, so probability $= |0|^2 = 0$ (impossible)

The algorithm designer's job is to construct a sequence of gates such that, after all operations complete, the wrong answers have uniformly distributed phases (they cancel when summed) while correct answers are in phase (they add constructively). This is not magic; it requires deep mathematical structure in the problem. Factoring integers has this structure; sorting a list does not.

Key Insight: Quantum speedup is not "trying all answers at once." It is carefully arranging phases so that computational paths leading to wrong answers *destructively interfere* while paths leading to correct answers *constructively interfere*. The algorithm must encode problem structure into phase relationships. Problems without exploitable structure do not benefit from quantum computation.

Seeing Interference in Action For readers who want to visualize this process, interactive quantum circuit simulators provide immediate feedback:

- **Quirk** (`https://algassert.com/quirk`): Drag-and-drop circuit builder that shows amplitude magnitudes and phases in real time. Try building a 2-qubit Grover's algorithm to watch amplitudes redistribute.
- **IBM Quantum Composer** (`https://quantum.ibm.com/composer`): Build circuits and run them on real quantum hardware. Observe how measurement statistics match theoretical predictions.

1.2.3 Quantum Logic Gates: How Operations Work

Quantum algorithms are built from **quantum gates**, which transform qubit states. Unlike classical gates (AND, OR, NOT), quantum gates must be **unitary**: reversible and norm-preserving. This section introduces the essential gates.

Single-Qubit Gates A single-qubit gate is a 2×2 unitary matrix acting on the qubit state vector $[\alpha, \beta]^T$. The most important gates are:

- **X Gate (NOT):** Flips $|0\rangle \leftrightarrow |1\rangle$. Matrix: $X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}$
- **Z Gate (Phase Flip):** Leaves $|0\rangle$ unchanged, flips the sign of $|1\rangle$. Matrix: $Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$
- **Hadamard Gate (H):** Creates superposition from basis states. $H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$, $H|1\rangle = \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. Matrix: $H = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$

On the Bloch sphere (Section 2), gates correspond to rotations: X rotates about the x-axis, Z rotates about the z-axis, and H rotates to the equator.

EE Analogy: Think of quantum gates as signal processing blocks with complex-valued transfer functions. The input is a 2D complex vector; the output is a transformed 2D complex vector. The unitarity constraint ($U^\dagger U = I$) is analogous to requiring a lossless, reversible transformation. This is like a passive reciprocal network: all information that goes in must come out.

Two-Qubit Gates and Entanglement The **CNOT** (Controlled-NOT) gate is the essential two-qubit gate. It flips the "target" qubit if and only if the "control" qubit is $|1\rangle$:

Input		**Output**	
Control	Target	Control	Target
$\|0\rangle$	$\|0\rangle$	$\|0\rangle$	$\|0\rangle$
$\|0\rangle$	$\|1\rangle$	$\|0\rangle$	$\|1\rangle$
$\|1\rangle$	$\|0\rangle$	$\|1\rangle$	$\|1\rangle$
$\|1\rangle$	$\|1\rangle$	$\|1\rangle$	$\|0\rangle$

The CNOT gate creates **entanglement** when applied to a superposition. Starting from $|00\rangle$:

1. Apply H to qubit 1: $H|0\rangle \otimes |0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |0\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |10\rangle)$
2. Apply CNOT (qubit 1 controls qubit 2): $\frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$

The result is a **Bell state**: the two qubits are correlated such that measuring one instantly determines the other, regardless of distance. This is *not* classical correlation (the qubits were not "secretly" in a definite state). Entanglement is a uniquely quantum resource exploited by algorithms and error correction.

Reading Circuit Diagrams Quantum circuits are read **left to right**. Each horizontal line represents a qubit; boxes represent gates; vertical lines connect multi-qubit gates. Figure 1 shows a circuit that creates a Bell state.

Universal Gate Set: Any quantum computation can be built from just H, T (a phase gate), and CNOT. This is analogous to how any classical circuit can be built from NAND gates. In practice, quantum computers implement a small set of "native" gates efficiently, and compilers decompose arbitrary operations into these primitives.

1.2.4 Examples of Quantum Speedup

Well-known quantum algorithms and their speedups are summarized in Table II. Each algorithm exploits a different type of mathematical structure.

**HHL speedup comes with significant caveats about input/output encoding.*

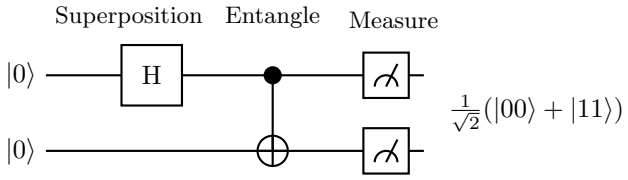

Figure 1: Quantum circuit for Bell state preparation and measurement. The Hadamard gate (H) creates superposition; the CNOT (dot = control, plus = target) creates entanglement. Measuring either qubit collapses both: if qubit 1 is $|0\rangle$, qubit 2 must be $|0\rangle$; if qubit 1 is $|1\rangle$, qubit 2 must be $|1\rangle$.

Table II: Examples of Quantum Speedup

Algorithm	**Problem**	**Classical**	**Quantum**	**Speedup**
Shor's [1]	Factor integer N	$O(e^{n^{1/3}})$	$O(n^3)$	**Exponential**
Grover's [2]	Unsorted search	$O(N)$	$O(\sqrt{N})$	Quadratic
Simulation	Quantum systems	$O(2^n)$	$O(\text{poly}(n))$	**Exponential**
HHL [3]	Linear systems	$O(N)$	$O(\log N)$	**Exponential***

What Each Algorithm Does:

Shor's Algorithm: Factors large integers by exploiting periodicity in modular exponentiation. This breaks RSA encryption, which relies on the difficulty of factoring products of large primes. A sufficiently large quantum computer could break most of today's public-key cryptography.

Grover's Algorithm: Searches an unsorted database of N items in $O(\sqrt{N})$ time instead of $O(N)$. This is "only" a quadratic speedup (not exponential), but it is provably optimal for unstructured search. It also applies to any NP problem: if verifying a solution takes time T, Grover's can find one in $O(\sqrt{N} \cdot T)$ time.

Quantum Simulation: Simulates quantum systems (molecules, materials, chemical reactions) efficiently. Classical computers struggle because simulating n quantum particles requires tracking 2^n amplitudes. Quantum computers represent these states natively. Applications include drug discovery, materials science, and understanding high-temperature superconductivity.

HHL Algorithm: Solves systems of linear equations $Ax = b$ exponentially faster than classical methods, with applications to machine learning and optimization. However, the speedup requires quantum-native input/output: preparing the input state and reading out the result both have overhead that can erase the advantage for many practical problems.

Important Caveat: Quantum computers do NOT speed up all problems. For most everyday computing tasks (web browsing, video encoding, databases), quantum computers offer no advantage. The speedup exists only for problems with specific mathematical structure that quantum interference can exploit.

1.2.5 What Quantum Computers Cannot Do

Quantum computers are **not** universal replacements for classical computers. Several fundamental limitations prevent QC from being "faster at everything":

1. **Measurement destroys information:** After measurement, superposition collapses to a single outcome. You cannot extract all 2^n computed values; you get one.

2. **No structure, no speedup:** Quantum algorithms require mathematical structure (periodicity, symmetry, phase patterns) to engineer interference. Most problems lack this structure.

3. **Input/output bottleneck:** Loading classical data into a quantum state and extracting results both require time proportional to data size. For problems dominated by I/O, quantum offers no advantage.

4. **Error correction overhead:** Fault-tolerant quantum computing requires 100–1000 physical qubits per logical qubit. A "1000-qubit" quantum computer may have only 1–10 logical qubits.

Table III: Where Quantum Computing Helps vs. Does Not Help

QC Provides Speedup	**QC Provides NO Speedup**
Factoring large numbers (Shor's)	Sorting, searching (only quadratic gain)
Simulating quantum systems	General database queries
Certain optimization problems	Video encoding/decoding
Breaking some cryptography (RSA, ECC)	Web browsing, file I/O
Some machine learning tasks (limited)	Most everyday computing
Quantum chemistry, drug discovery	Compression algorithms

Bottom Line: Quantum computers are specialized accelerators for problems with specific mathematical structure, not general-purpose replacements for classical computers. The laptop running your web browser will always be classical.

1.2.6 Why Readout Matters for Speedup

The interference patterns that encode the answer are fragile. They exist in the complex amplitudes α_x, which we cannot directly observe. We can only measure, which collapses the superposition and returns a single outcome. If the algorithm worked correctly, that outcome is (probably) the right answer. But there are failure modes:

- **Decoherence during computation:** If the coherence time T2 is too short (see Section 8 for formal definition), the interference pattern degrades before we finish the algorithm
- **Gate errors:** Imperfect gates corrupt the amplitudes, causing wrong answers to gain probability
- **Readout errors:** Even if the qubit is in the right state, misclassification reports the wrong answer

This is why high-fidelity readout is essential. The quantum computation may have produced the correct answer, but a readout error at the final step throws it away. For algorithms requiring many measurements (like variational algorithms), readout errors compound across shots. For quantum error correction, readout errors can be mistaken for data errors, triggering incorrect corrections.

Further Reading and Resources
For readers who want to explore quantum computing concepts in more depth before continuing with the hardware-focused sections of this document:
Video Lectures:

- "Quantum Computing for Computer Scientists" (Microsoft Research, YouTube): A rigorous 1-hour introduction that builds from linear algebra to Shor's algorithm.

Interactive Learning:

- IBM Quantum Learning (`https://learning.quantum.ibm.com/`): Structured courses from basics to advanced algorithms, with hands-on exercises on real quantum hardware.

- Qiskit Textbook (`https://qiskit.org/textbook/`): Comprehensive open-source textbook with executable Python code. Covers theory and implementation.

Deeper Reading:

- Scott Aaronson's blog (`https://scottaaronson.blog/`): A complexity theorist's perspective on quantum computing claims, hype, and genuine breakthroughs. Excellent for developing critical thinking about QC announcements.

2 The Transmon Qubit: A Nonlinear LC Oscillator

Quantum computation requires a physical system with exactly two accessible energy levels: a **qubit**. Why build one from an oscillator? Because quantum harmonic oscillators are among the best-understood quantum systems, they can be manufactured with existing microwave technology, and they couple naturally to electromagnetic control fields. The challenge is that a *linear* oscillator has infinitely many evenly-spaced energy levels, not two. The transmon solves this by introducing nonlinearity via a Josephson junction, creating unequal energy spacing that allows selective addressing of just the lowest two levels.

This section explains why linear oscillators fail as qubits, how the Josephson junction provides the necessary nonlinearity, and how the transmon circuit realizes a practical qubit. Understanding this circuit is essential for the readout discussion in later sections.

2.1 The Problem with Linear LC Circuits

Consider a standard LC oscillator. When quantized (treated quantum mechanically), its energy levels are:

$$E_n = \hbar\omega\left(n + \frac{1}{2}\right), \quad \omega = \frac{1}{\sqrt{LC}} \tag{2}$$

Variable Definitions for Eq. (2):
E_n — energy of the n-th quantum state (Joules)
$\hbar = h/2\pi \approx 1.055 \times 10^{-34}$ J·s — reduced Planck constant
ω — angular resonance frequency (rad/s)
$n = 0, 1, 2, \ldots$ — quantum number (state index)
L, C — inductance and capacitance of the LC circuit

This is the quantum harmonic oscillator [4], with **evenly spaced** energy levels separated by $\hbar\omega$. The transition frequency between any two adjacent levels is identical:

$$f_{01} = f_{12} = f_{23} = \cdots = \frac{\omega}{2\pi} \tag{3}$$

Variable Definitions for Eq. (3):
$f_{01}, f_{12}, f_{23}, \ldots$ — transition frequencies between adjacent energy levels (Hz)
ω — angular frequency of the oscillator (rad/s)
For a harmonic oscillator, all transition frequencies are identical.

The Problem: If all transitions have the same frequency, we cannot selectively drive just the $|0\rangle \leftrightarrow |1\rangle$ transition. Any pulse that excites $|0\rangle \to |1\rangle$ will also excite $|1\rangle \to |2\rangle$, $|2\rangle \to |3\rangle$, etc. We need a two-level system, but we have an infinite ladder.

2.2 The Josephson Junction Solution

The solution is to replace the linear inductor with a **Josephson junction** [5], which acts as a nonlinear inductor. The junction's inductance depends on the current flowing through it:

$$\boxed{L_J(\delta) = \frac{\Phi_0}{2\pi I_c \cos\delta}} \tag{4}$$

Variable Definitions:
$\Phi_0 = h/2e \approx 2.07 \times 10^{-15}$ Wb — magnetic flux quantum
I_c — critical current of the junction (typically ∼10–100 nA)
$\delta = \phi_L - \phi_R$ — superconducting phase difference across the junction

EE Analogy: Think of the Josephson junction as a **varactor** (voltage-dependent capacitor), but for inductance. Just as a varactor's capacitance varies with bias voltage, the Josephson junction's inductance varies with the current (or equivalently, the phase) across it.

This nonlinear inductance creates a **cosine potential** instead of the quadratic potential of a linear inductor. The energy stored in the junction is:

$$U_J = -E_J \cos\delta, \quad \text{where } E_J = \frac{\Phi_0 I_c}{2\pi} \tag{5}$$

Variable Definitions for Eq. (5):
U_J — potential energy stored in the Josephson junction (Joules)
E_J — Josephson energy (Joules); sets the depth of the cosine potential well
δ — superconducting phase difference across the junction (radians)
$\Phi_0 = h/2e \approx 2.07 \times 10^{-15}$ Wb — magnetic flux quantum
I_c — critical current of the junction (Amperes)

The cosine potential has a different curvature than a parabola, leading to **unequally spaced energy levels**:

$$\boxed{f_{12} = f_{01} - \alpha, \quad \alpha \approx \frac{E_C}{h} \approx 200\text{–}300 \text{ MHz}} \tag{6}$$

Variable Definitions for Eq. (6):
f_{01} — transition frequency between ground state $|0\rangle$ and first excited state $|1\rangle$ (Hz)
f_{12} — transition frequency between $|1\rangle$ and $|2\rangle$ (Hz)
α — anharmonicity; the frequency difference that enables selective addressing
$E_C = e^2/(2C)$ — charging energy of the qubit capacitance (Joules)
$h \approx 6.63 \times 10^{-34}$ J·s — Planck constant

The parameter α is called the **anharmonicity** [6]. It represents the frequency difference between adjacent transitions. With $\alpha \approx 200$ MHz, we can send a ∼20 ns pulse (bandwidth ∼50 MHz) that drives $|0\rangle \leftrightarrow |1\rangle$ without significantly exciting $|1\rangle \to |2\rangle$.

2.3 The Transmon Circuit

The **transmon** [6] (transmission-line shunted plasma oscillation qubit) is the most common superconducting qubit design. At its core, it consists of a Josephson junction shunted by a large capacitor—but physical implementations reveal a rich hierarchical structure spanning three orders of magnitude in scale, as shown in Fig. 2.

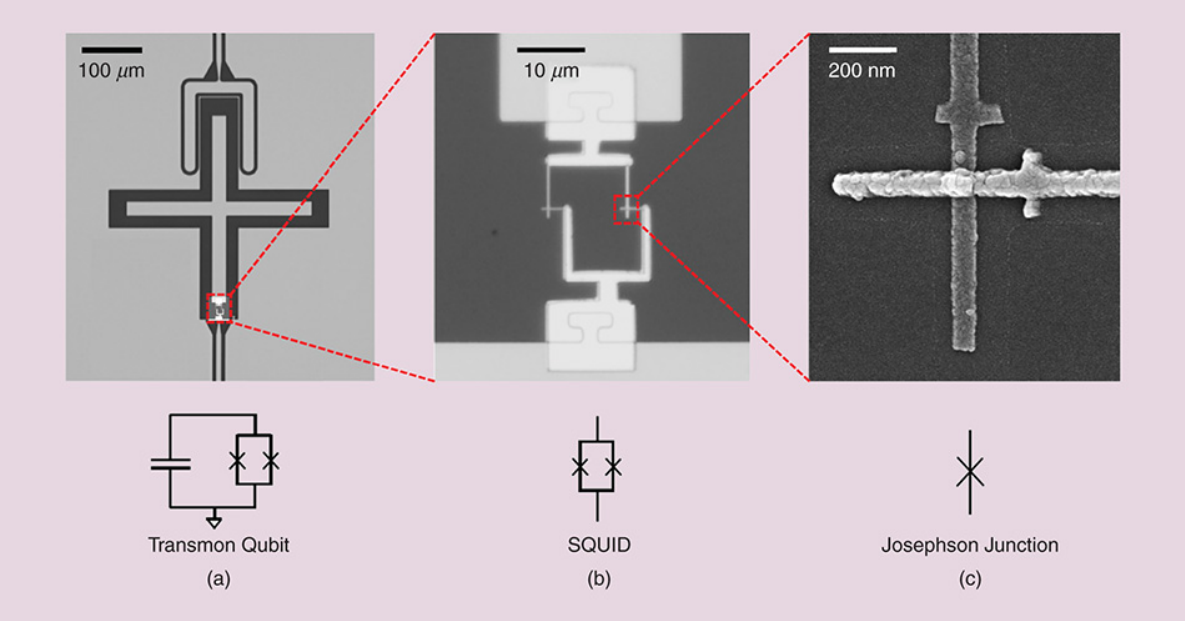

Figure 2: Multi-scale view of a transmon qubit: (a) Full transmon with cross-shaped capacitor pads and SQUID at center (100 μm scale), (b) SQUID detail showing two Josephson junctions in a loop (10 μm scale), (c) Single Josephson junction (200 nm scale). Circuit symbols shown below each image. Image from [7].

Modern transmons typically use a **SQUID** (Superconducting Quantum Interference Device)—two Josephson junctions in a superconducting loop—rather than a single junction. This allows *in-situ* tuning of the effective Josephson energy via an external magnetic flux threading the loop. The large cross-shaped pads visible in panel (a) form the shunt capacitor C_s; their geometry is carefully designed to minimize surface losses while providing the required capacitance (∼80 fF) for operation in the transmon regime.

Table IV: Transmon Qubit Parameters

Parameter	**Typical Value**	**Derived Quantity**
Shunt capacitance C_s	∼80 fF	$E_C/h \approx 240$ MHz
Critical current I_c	∼40 nA	$E_J/h \approx 20$ GHz
Josephson inductance L_{J0}	∼8 nH	At zero bias
Qubit frequency f_{01}	4–5 GHz	$\approx \sqrt{8E_J E_C}/h$
Anharmonicity α	200–300 MHz	$\approx E_C/h$

Key Insight: The transmon operates in the regime $E_J \gg E_C$ [6] (typically $E_J/E_C \approx 50$–100). As shown in Table IV, this makes the qubit insensitive to charge noise (a major source of decoherence in earlier designs) while maintaining sufficient anharmonicity for selective addressing.

Figure 3 shows the complete physical architecture of a real transmon qubit from an SQMS fabrication run, illustrating how the circuit elements described above are realized in practice.

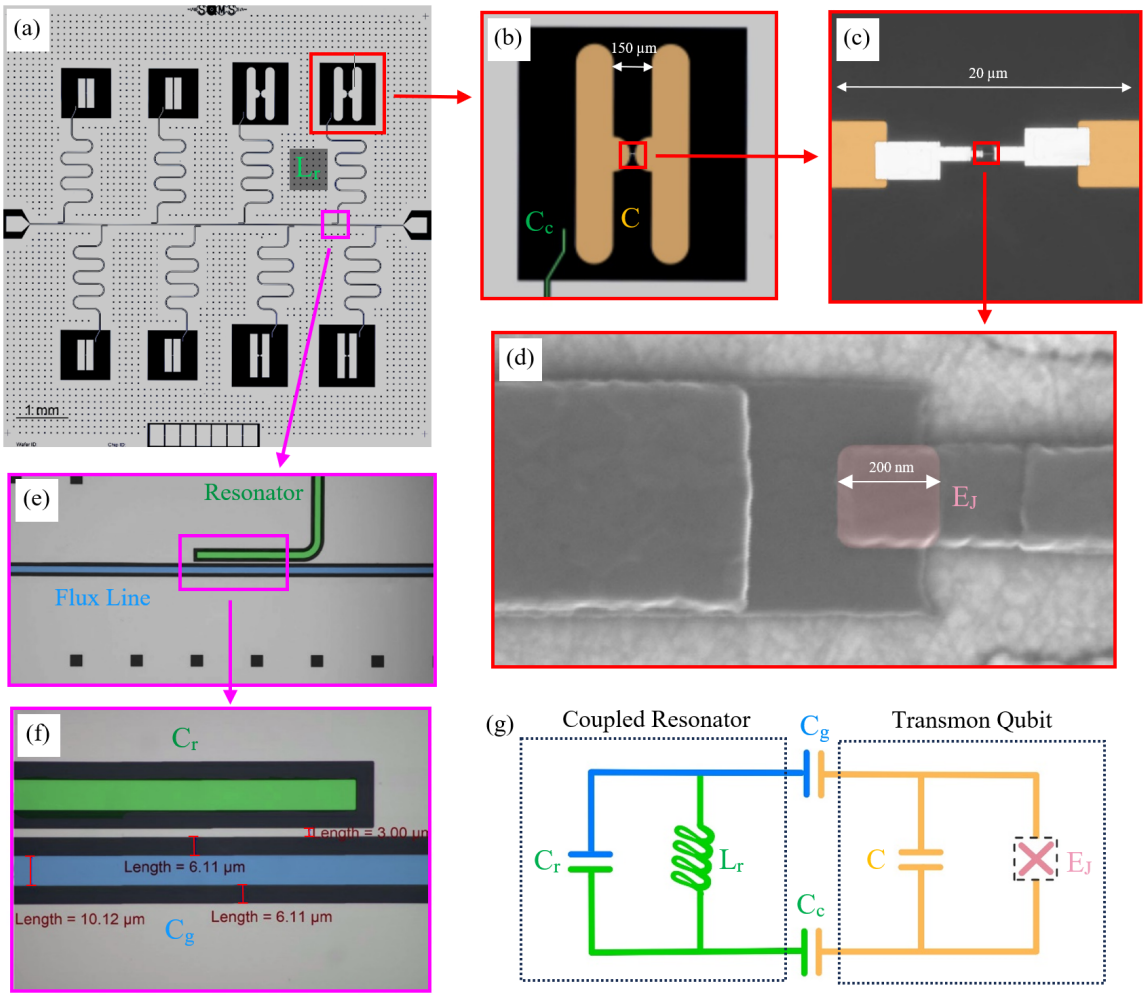

Figure 3: Physical qubit architecture and circuit representation (SQMS fabrication). (a) 8-qubit chip for testing three distinct qubit geometries; signals enter via 50 Ω launchers on flux line. (b) False-colored single qubit: niobium-tantalum encapsulated pads (orange) form capacitors for the Josephson junction on sapphire substrate. (c) Zoomed junction region (20 μm). (d) SEM image of Josephson junction: aluminum superconductors separated by aluminum oxide barrier (40/90 nm electrode thicknesses, ∼200 nm × 200 nm junction area). (e) Resonator capacitively coupled to flux line. (f) Coupling dimensions. Image reversed on vertical axis. (g) Circuit diagram: E_J (Josephson junction), C (qubit capacitance), C_c (coupling capacitor), C_r/L_r (resonator LC circuit), C_g (gate capacitor). Photos courtesy of Mustafa Bal and Nanofabrication Taskforce at SQMS.

Physical Architecture Details

The chip in Fig. 3(a) contains eight qubits arranged to test different capacitor geometries. Each qubit is coupled to a **coplanar waveguide (CPW) resonator**, visible as the meandering transmission line in panel (e), which serves as the primary interface for both control and readout.

50 Ω Launchers and Chip I/O: The rectangular pads at the left and right sides of the chip in Fig. 2(a) are **50 Ω launchers**: impedance-matched transitions that connect the on-chip transmission lines to external microwave electronics. Gold or aluminum wirebonds connect these launchers to a printed circuit board (PCB) mounted in the cryostat. The 50 Ω characteristic impedance matches standard microwave

test equipment, which minimizes signal reflections that would otherwise distort pulses and corrupt readout signals.

Feedline Architecture and Frequency Multiplexing: A central **feedline** (visible running horizontally across the chip) acts as a shared microwave highway. Multiple readout resonators, each tuned to a *different* frequency, couple capacitively to this feedline. This architecture enables **frequency-division multiplexing**: a single input/output line can address many qubits simultaneously by sending microwave tones at each resonator's unique frequency. Adjacent resonators are typically spaced 50–100 MHz apart to prevent crosstalk [8]. For an 8-qubit chip, resonator frequencies might span 6.0–6.7 GHz. Qubit frequencies are deliberately placed ∼1–2 GHz *below* their resonators (e.g., 4.5–5.5 GHz) to ensure operation in the dispersive regime, where the large detuning $|\Delta| = |\omega_q - \omega_r| \gg g$ prevents energy exchange between qubit and resonator [6].

Qubit-Resonator Coupling: Each qubit couples to its dedicated resonator through a small coupling capacitor C_g (panel (g) of Fig. 3). The coupling strength $g/2\pi \sim 50$–100 MHz is carefully engineered: strong enough for fast readout, but weak enough to operate in the **dispersive regime** where the qubit-resonator detuning $\Delta = \omega_q - \omega_r$ satisfies $|\Delta| \gg g$. In this regime, the resonator frequency shifts by $\pm\chi$ depending on whether the qubit is in $|0\rangle$ or $|1\rangle$, where $\chi \approx g^2/\Delta$. This state-dependent shift (typically 1–5 MHz) enables **quantum non-demolition (QND) measurement**, meaning the qubit state can be read without destroying it.

Signal Flow:

Readout: A microwave tone near the resonator frequency enters through one launcher, travels along the feedline, and interacts with each resonator it passes. The transmitted (or reflected) signal carries phase and amplitude information encoding the qubit states, then exits through the opposite launcher to be amplified and digitized.

Control: Qubit drive pulses at the qubit frequency f_{01} also enter through the feedline. Although primarily intended for readout, the resonator acts as a bandpass filter that couples control pulses to the qubit while providing **Purcell filtering**, which suppresses qubit energy decay into the feedline at frequencies far from the resonator.

Materials Engineering: Achieving long coherence times requires careful materials selection. The SQMS design uses:

- **Sapphire substrate**: Low-loss dielectric ($\tan\delta < 10^{-6}$) with excellent thermal conductivity at millikelvin temperatures
- **Niobium-tantalum encapsulation**: Protects niobium surfaces from native oxide formation, reducing two-level system (TLS) defects that limit T_1 [9]
- **Aluminum Josephson junctions**: Standard Al/AlO_x/Al trilayer fabricated via double-angle evaporation, with junction area ∼200 nm × 200 nm setting E_J

3 Cooper Pairs and the Josephson Effect

This section explains the physics underlying the Josephson junction. For readers primarily interested in signal processing applications, this section can be skimmed; the key result is Equation (2).

3.1 Why Superconductors Are Different

At temperatures below the critical temperature T_c (about 1.2 K for aluminum [10]), electrons in a metal form bound pairs called **Cooper pairs** [11]. This pairing is mediated by phonons (lattice vibrations):

1. An electron moving through the lattice attracts nearby positive ions
2. This creates a region of slightly higher positive charge density
3. A second electron is attracted to this region
4. The two electrons become weakly bound (binding energy ∼0.1–1 meV)

Table V: Single Electron vs. Cooper Pair Properties

Property	**Single Electron**	**Cooper Pair**
Spin	1/2 (fermion)	0 (boson)
Pauli exclusion	Yes	No
Charge	$-e$	$-2e$
Can occupy same state	No	Yes (all pairs)

The key difference, as shown in Table V, is that Cooper pairs are bosons. Because bosons are not subject to the Pauli exclusion principle, they can all condense into the **same quantum ground state** [10]. This Bose-Einstein condensate is described by a single macroscopic wavefunction:

$$\Psi = \sqrt{n_s} \cdot e^{i\phi} \tag{7}$$

Variable Definitions for Eq. (7):
Ψ — macroscopic wavefunction of the superconducting condensate
n_s — Cooper pair density (pairs per unit volume)
ϕ — macroscopic phase of the superconductor (radians); coherent across the entire material
$|\Psi|^2 = n_s$ — the squared magnitude gives the pair density

where n_s is the Cooper pair density and ϕ is the **macroscopic phase**, coherent across the entire superconductor.

EE Analogy: Think of the phase ϕ as a clock signal that is perfectly synchronized across the entire superconductor. All Cooper pairs oscillate in phase, like a distributed oscillator with zero phase noise.

Music Analogy: Imagine a massive choir where every singer holds the exact same note in perfect unison, with no one drifting sharp or flat. Cooper pairs in a superconductor are like this choir: millions of electron pairs all "singing" at exactly the same phase, which is why current flows without resistance.

3.2 Quantum Tunneling

Quantum tunneling [4] is not unique to Cooper pairs; it occurs for any quantum particle. The key physics is that a particle's wavefunction does not stop at a potential barrier but decays exponentially inside it:

$$\psi(x) \propto e^{-\kappa x}, \quad \kappa = \frac{\sqrt{2m(V_0 - E)}}{\hbar} \tag{8}$$

Variable Definitions for Eq. (8):
$\psi(x)$ — particle wavefunction amplitude inside the barrier
κ — decay constant inside the barrier (inverse meters)
m — particle mass (kg)
V_0 — barrier height (Joules)
E — particle energy (Joules); must have $E < V_0$ for tunneling
x — position inside the barrier (meters)

If the barrier is thin enough (∼1 nm for a Josephson junction), the wavefunction has non-zero amplitude on the other side, giving a finite tunneling probability:

$$T \approx e^{-2\kappa d} \tag{9}$$

Variable Definitions for Eq. (9):
T — tunneling probability (dimensionless, 0 to 1)
κ — decay constant inside the barrier (m^{-1})
d — barrier thickness (meters)
For a Josephson junction with ∼1 nm oxide barrier, T can be significant (∼0.01–0.1).

Single electrons tunnel through thin insulators routinely (this is how scanning tunneling microscopes and flash memory work). What makes **Josephson tunneling** special is that Cooper pairs tunnel **coherently** (preserving phase information) and **dissipationlessly** (no energy loss).

3.3 The Josephson Equations

When two superconductors are separated by a thin insulator, Cooper pairs can tunnel between them [5]. The tunneling current depends on the phase difference $\delta = \phi_L - \phi_R$:

$$\boxed{I = I_c \sin\delta} \quad \text{(First Josephson Equation)} \tag{10}$$

Variable Definitions for Eq. (10):
I — supercurrent flowing through the junction (Amperes)
I_c — critical current; maximum supercurrent the junction can carry (Amperes)
$\delta = \phi_L - \phi_R$ — phase difference between left and right superconductors (radians)

The phase evolves according to the voltage across the junction:

$$\boxed{\frac{d\delta}{dt} = \frac{2eV}{\hbar} = \frac{2\pi V}{\Phi_0}} \quad \text{(Second Josephson Equation)} \tag{11}$$

Variable Definitions for Eq. (11):
$d\delta/dt$ — rate of change of phase difference (rad/s)
V — voltage across the junction (Volts)
$e \approx 1.602 \times 10^{-19}$ C — electron charge
$\Phi_0 = h/2e \approx 2.07 \times 10^{-15}$ Wb — magnetic flux quantum
Note: A DC voltage causes the phase to wind continuously, producing an AC current at frequency $f = V/\Phi_0$.

3.4 Why Josephson Tunneling is Dissipationless

In a superconductor, there is an energy gap Δ [10] ($\sim$0.2 meV for aluminum). To create a quasiparticle excitation (break a Cooper pair), energy 2Δ is required. For voltages $V < 2\Delta/e \approx 0.4$ mV:

- Not enough energy to break Cooper pairs
- No quasiparticle current (which would be dissipative)
- Only coherent Cooper pair tunneling occurs
- **Zero resistance** (supercurrent)

Qubits always operate in this sub-gap regime, ensuring dissipationless dynamics. Figure 4 illustrates the Josephson junction structure and the resulting anharmonic energy levels that enable qubit operation.

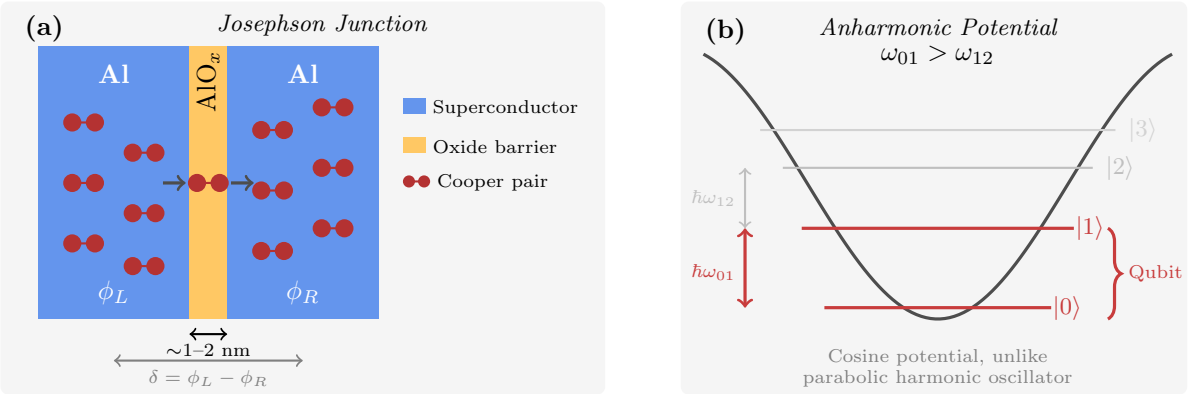

Figure 4: Josephson junction and qubit energy levels: (a) Two superconducting aluminum electrodes separated by a thin aluminum oxide (AlO_x) barrier. Cooper pairs tunnel coherently through the barrier, with tunneling current $I = I_c \sin\delta$ depending on the phase difference $\delta = \phi_L - \phi_R$. (b) The junction's nonlinear inductance creates a cosine potential $U \propto (1 - \cos\delta)$ with a characteristic flat bottom. Unlike a harmonic oscillator (parabolic potential), the energy levels are unequally spaced ($\omega_{01} > \omega_{12}$), enabling selective driving of only the $|0\rangle \leftrightarrow |1\rangle$ transition—the defining feature of a qubit.

3.5 Visualizing Qubit States: Bloch Sphere and Wigner Function

Before moving to measurement, it is worth briefly noting how qubit states are typically visualized. While not essential for understanding the signal processing focus of this document, these representations appear frequently in quantum computing literature and serve different purposes.

The Bloch Sphere Any pure state of a two-level quantum system (qubit) can be written as:

$$|\psi\rangle = \cos\left(\frac{\theta}{2}\right)|0\rangle + e^{i\phi}\sin\left(\frac{\theta}{2}\right)|1\rangle \tag{12}$$

Variable Definitions for Eq. (12):
θ — polar angle from the z-axis (0 to π)
ϕ — azimuthal angle in the x-y plane (0 to 2π)
$|0\rangle, |1\rangle$ — computational basis states

This parameterization maps qubit states to points on a unit sphere called the **Bloch sphere** [12]. The north pole represents $|0\rangle$, the south pole represents $|1\rangle$, and points on the equator represent equal superpositions with different phases. Qubit gates correspond to rotations on this sphere: an X gate (π-pulse) rotates 180° about the x-axis, flipping $|0\rangle \leftrightarrow |1\rangle$; a Hadamard gate rotates to the equator, creating a superposition.

EE Analogy: The Bloch sphere is a 3D extension of a phasor diagram. The z-axis represents the "amplitude" (population difference between $|0\rangle$ and $|1\rangle$), while the azimuthal angle represents phase, just as in a phasor. RF engineers visualize signals as phasors rotating in the complex plane; quantum engineers visualize qubit states as vectors on the Bloch sphere. A key difference: in RF, the phasor rotates at the carrier frequency; on the Bloch sphere, the "rotation" is removed by working in the rotating frame.

The Wigner Function The Wigner function $W(x, p)$ is a quasi-probability distribution that represents quantum states in phase space (position and momentum, or equivalently, the two quadratures I and Q). Unlike classical probability distributions, the Wigner function can take negative values, which is a signature of non-classical behavior. For Gaussian states (coherent states, thermal states), the Wigner function is positive everywhere; negativity indicates quantum features with no classical analog.

EE Analogy: If you know the **Wigner-Ville distribution** from signal processing, the quantum Wigner function is its direct analog. Both are phase space representations (time-frequency for signals, position-momentum for quantum states), and both can exhibit negative values due to interference between components. The key difference in quantum mechanics is that the negativity is not merely a mathematical artifact: it has physical significance.

Why Wigner Negativity Matters in Quantum Computing For readers familiar with time-frequency distributions from DSP, the quantum Wigner function offers a deeper insight: **negativity is the signature of genuine quantum behavior**.

- **Hudson's Theorem:** A pure quantum state has a non-negative Wigner function if and only if it is Gaussian (coherent states, squeezed states). Any non-Gaussian state (Fock states, cat states, GKP states) *must* have negative regions.
- **Quantum vs. Classical:** A positive Wigner function can be interpreted as a classical probability distribution over hidden variables. Negativity proves that *no* such classical description exists. This is not a technicality; it is a fundamental statement about the nature of the state.
- **Computational Resource:** In bosonic quantum computing (using microwave cavities as qubits), Wigner negativity is a *resource* for quantum advantage. The Gottesman-Kitaev-Preskill (GKP) code

and cat codes exploit non-Gaussian states with negative Wigner functions to achieve fault-tolerant quantum computation.

Connection to Readout The IQ plane used for qubit readout is essentially a **projection** of the Wigner function onto a specific quadrature axis. After heterodyne detection, what we measure is the marginal distribution of the Wigner function along the I and Q axes. The Wigner function contains the full quantum state information; the IQ plane shows a lossy classical projection of it. This is why repeated measurements are needed to reconstruct the full Wigner function (via quantum state tomography), but a single-shot IQ measurement suffices for state discrimination.

When to Use Each Representation Each representation serves different purposes (Table VI).

Table VI: When to Use Each Quantum State Representation

Representation	Best For	Limitations
Bloch Sphere	Single-qubit gates, visualizing rotations, understanding pulse sequences, tracking state evolution during control	Only works for single qubits (or single-qubit reduced states); does not capture multi-qubit entanglement or bosonic modes
Wigner Function	Cavity states, bosonic codes, visualizing non-classical features (cat states, Fock states), understanding measurement back-action	Requires 2D plotting; less intuitive for simple qubit operations; negativity requires careful interpretation
IQ Plane (Readout)	Measurement discrimination, signal processing, classifier design, SNR analysis	Classical representation; does not capture quantum coherence or state purity

For This Document: *The readout signal processing discussed in subsequent sections uses the IQ plane representation, where qubit states appear as two Gaussian clusters after amplification. This is a classical signal processing problem. The Bloch sphere and Wigner function are mentioned here for completeness but are not required for understanding the adaptive filtering application.*

4 Qubit State Measurement: Dispersive Readout

Measuring a qubit state is fundamentally an RF engineering problem. This section describes the standard approach used in nearly all superconducting quantum computers.

4.1 The Dispersive Regime

The qubit is capacitively coupled to a **readout resonator** (a linear microwave cavity, typically ∼7 GHz). When the qubit and resonator frequencies are far detuned ($|\omega_q - \omega_r| \gg g$, where g is the coupling strength), the system is in the **dispersive regime** [8, 13].

In this regime, the resonator frequency depends on the qubit state:

$$\boxed{f_r^{|0\rangle} = f_r + \chi, \quad f_r^{|1\rangle} = f_r - \chi} \tag{13}$$

Variable Definitions for Eq. (13):
f_r — bare resonator frequency (∼7 GHz)
χ — dispersive shift (typically 1–5 MHz) [8, 14]
The total shift between states is 2χ

EE Analogy: This is analogous to a voltage-controlled oscillator (VCO) where the "control voltage" is the qubit state. The qubit acts as a state-dependent reactive load that pulls the resonator frequency.

Music Analogy: Imagine gently touching a vibrating guitar string. Depending on where you touch it, the pitch shifts slightly. In dispersive readout, we "listen" to a resonator's pitch to figure out the qubit's state: the qubit shifts the resonator's frequency just like your finger shifts the string's pitch, without stopping the vibration entirely.

Intuitive Picture (Interference Again): The dispersive shift can be understood as interference, connecting back to Section 1. The qubit and resonator exchange **virtual photons**: the resonator briefly "lends" energy to the qubit, which returns it. Depending on the qubit state, these exchanges create constructive interference at slightly different frequencies. The $|0\rangle$ state "pushes" the resonator to higher frequency; the $|1\rangle$ state "pulls" it lower. Over many photon round-trips during a single measurement pulse, this small shift accumulates into a measurable phase difference.
Just as quantum algorithms use interference to amplify correct answers, dispersive readout uses the accumulated phase (constructive interference of resonator photons at a shifted frequency) to distinguish qubit states. The measurement is fundamentally a phase-sensitive detection of this interference pattern.

4.2 The Measurement Process

To measure the qubit, we probe the resonator with a microwave tone and detect the state-dependent phase/amplitude shift:

1. **Send probe tone** at frequency near f_r
2. **Signal reflects** off resonator with state-dependent phase:
   - Qubit in $|0\rangle$: resonator at $f_r + \chi \rightarrow$ phase ϕ_0
   - Qubit in $|1\rangle$: resonator at $f_r - \chi \rightarrow$ phase ϕ_1

3. **Amplify** using cryogenic amplifier (HEMT or JPA/TWPA)
4. **Downconvert** to intermediate frequency (∼100 MHz)
5. **Digitize** I and Q quadratures
6. **Classify** in IQ plane to determine state

4.3 The IQ Plane Representation

After digitization, each readout pulse produces a point in the IQ plane (Fig. 5). Due to the dispersive shift, $|0\rangle$ and $|1\rangle$ states produce points clustered in different regions.

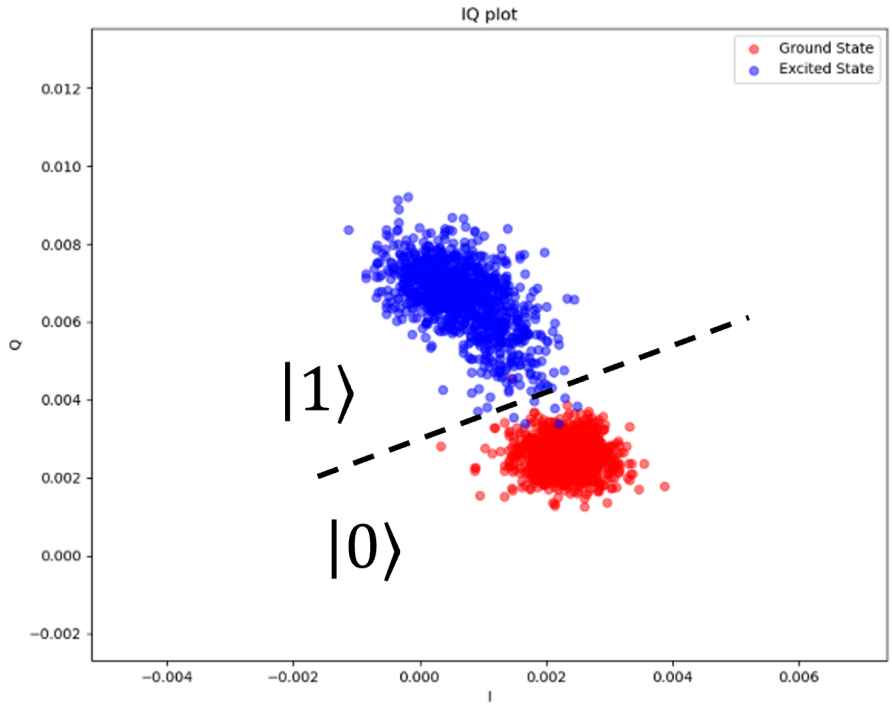

Figure 5: Qubit state discrimination in the IQ plane from actual experimental data. Each dot represents a single-shot readout measurement: red points correspond to the ground state $|0\rangle$, blue points to the excited state $|1\rangle$. The dashed line shows the optimal linear decision boundary for state classification. Cluster separation is proportional to $2\chi\tau$ (Eq. 13); cluster width reflects system noise; SNR determines classification accuracy.

Why the IQ Plane? The IQ (In-phase/Quadrature) representation is fundamental to dispersive readout because it captures both amplitude and phase information from the resonator response. When the readout tone reflects off the resonator, it acquires a state-dependent phase shift: if the qubit is in $|0\rangle$, the resonator sits at frequency $f_r + \chi$; if in $|1\rangle$, at $f_r - \chi$ (Eq. 13). After mixing with a local oscillator and low-pass filtering, this phase difference maps to different locations in the IQ plane.

What the Clusters Tell Us In Fig. 5, two key geometric features determine readout fidelity:

- **Cluster separation** is proportional to the dispersive shift χ and integration time τ. A larger χ (stronger qubit-resonator coupling in the dispersive regime) or longer integration pushes the clusters farther apart. The total separation scales as $2\chi\tau$.
- **Cluster width** reflects the total system noise, combining thermal fluctuations, amplifier noise, and any other broadening mechanisms. In most practical systems, the first-stage amplifier (typically a HEMT at 4 K with noise temperature 2–10 K) dominates this contribution.

The signal-to-noise ratio for state discrimination is simply:

$$\text{SNR} = \frac{\text{cluster separation}}{\text{cluster width}} \tag{14}$$

Higher SNR means less overlap between clusters and fewer classification errors. The dashed decision boundary in Fig. 5 shows where we draw the line: points on one side are classified as $|0\rangle$, points on the other as $|1\rangle$. Classification errors occur when noise pushes a point across this boundary.

Key Insight: The readout problem is fundamentally a **two-class Gaussian classification** problem in 2D. The clusters overlap due to noise; classification errors occur when a point from one class falls in the other class's region. Improving SNR (increasing separation or decreasing width) directly improves classification fidelity.

4.4 Sources of Readout Error

Among the error sources listed in Table VII, amplifier noise typically dominates in practical systems.

Table VII: Sources of Readout Error

Error Source	**Mechanism**	**Typical Contribution**
Thermal noise	Johnson noise from finite temperature	Negligible at 20 mK
Amplifier noise	Added noise from HEMT ($\sim$2–10 K noise temp)	Dominant for most systems
Qubit decay (T1)	$\|1\rangle \to \|0\rangle$ decay during measurement (see Section 8)	Grows with readout time
State preparation	Qubit not in intended initial state	$\sim$0.5–2%
Measurement-induced	High photon number kicks qubit to $\|2\rangle$	Limits max readout power

The fundamental tradeoff is between integration time and T1 decay (the energy relaxation time; see Section 8):

- **Longer integration** $\to$ more signal $\to$ higher SNR
- **But** longer integration $\to$ more time for $|1\rangle \to |0\rangle$ decay $\to$ measure wrong state
- **Optimal readout time** is typically 100–500 ns

4.5 From Superposition to Definite Outcomes: How Measurement Works

A fundamental question arises: if a qubit can exist in a superposition of $|0\rangle$ and $|1\rangle$, how do we ever get a definite answer? This section addresses the physics of quantum measurement and how we verify that our readout is correct.

The Measurement Postulate In quantum mechanics, measurement is fundamentally different from classical observation [12]. When we measure a qubit in the state:

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \quad |\alpha|^2 + |\beta|^2 = 1 \tag{15}$$

Variable Definitions for Eq. (14):
$|\psi\rangle$ — quantum state of the qubit
$|0\rangle, |1\rangle$ — computational basis states (ground and excited states)
α, β — complex probability amplitudes
$|\alpha|^2$ — probability of measuring state $|0\rangle$
$|\beta|^2$ — probability of measuring state $|1\rangle$
The normalization $|\alpha|^2 + |\beta|^2 = 1$ ensures total probability equals 1.

The measurement does not reveal "a little bit of $|0\rangle$ and a little bit of $|1\rangle$." Instead:

- We get **exactly** $|0\rangle$ with probability $|\alpha|^2$
- We get **exactly** $|1\rangle$ with probability $|\beta|^2$
- The superposition is **destroyed** (this is called "wavefunction collapse")
- After measurement, the qubit is in whichever state we measured

EE Analogy: Think of measurement like sampling a noisy signal with a 1-bit ADC. Before sampling, the signal could be anywhere. After sampling, you get either 0 or 1. The "superposition" is like the continuous voltage; the "measurement" is the quantization. The key difference: in quantum mechanics, this discretization is fundamental, not due to limited resolution. There is no "actual value" between 0 and 1 that we are simply failing to resolve.

Music Analogy: Think of a guitar string vibrating with multiple harmonics blended together. Before you record it, all those frequencies coexist. But the instant the microphone captures the sound, you get one specific waveform. Quantum measurement is similar: before measurement, the qubit exists in a blend of states; the act of measuring "records" one definite outcome.

What Physically Happens During Dispersive Readout When we send a probe tone into the readout resonator:

1. **Before measurement:** Qubit is in superposition $\alpha|0\rangle+\beta|1\rangle$. The resonator-qubit system is entangled.
2. **Photons enter resonator:** The probe tone populates the resonator with photons. These photons interact with the qubit via the dispersive coupling.
3. **Information leaks out:** The reflected/transmitted signal carries information about the qubit state (the phase shift encodes $|0\rangle$ vs $|1\rangle$).
4. **Amplification:** The HEMT or JPA amplifies this signal, but amplification is irreversible; it "copies" quantum information into many photons.
5. **Collapse occurs:** Once enough information has leaked into the environment (the amplifier, the measurement apparatus, ultimately the lab), the superposition collapses. The qubit is now in a definite state.
6. **We record the outcome:** The digitized IQ point tells us which state the qubit collapsed into.

Key Insight: The qubit does not "decide" its state when we look at the ADC output. The collapse happens earlier, when enough photons carrying state information have been irreversibly amplified. By the time we digitize, the qubit has already committed to $|0\rangle$ or $|1\rangle$. The IQ point we measure is simply our record of that outcome, corrupted by noise.

How Do We Know the Readout Is Correct? Since each measurement destroys the quantum state, we cannot measure the same qubit twice to "check our work." Instead, verification relies on statistical protocols (Table VIII).

Readout Fidelity Calculation:
Prepare $|0\rangle$ N_0 times $\rightarrow$ measure $|0\rangle$ n_{00} times, measure $|1\rangle$ n_{01} times
Prepare $|1\rangle$ N_1 times $\rightarrow$ measure $|0\rangle$ n_{10} times, measure $|1\rangle$ n_{11} times

$$F_{\text{readout}} = \frac{1}{2}\left(\frac{n_{00}}{N_0} + \frac{n_{11}}{N_1}\right) \tag{16}$$

Variable Definitions for Eq. (15):
F_{readout} — readout fidelity (0 to 1)

Table VIII: Readout Verification Methods

Verification Method	Procedure	What It Tests
Ground state prep.	Wait $\gg$ T1, then measure. Should get $\|0\rangle$ with probability $\sim$99%+.	Thermal population, readout bias
π-pulse calibration	Prepare $\|0\rangle$, apply π-pulse, measure. Should get $\|1\rangle$.	Pulse calibration + readout
Rabi oscillations	Vary pulse duration, measure. Should see sinusoidal oscillation.	Coherent control + readout consistency
Confusion matrix	Prepare known states many times, count misclassifications.	Readout fidelity directly
Repeated measurement	Measure, then immediately measure again (QND). Should agree.	Measurement is projective

N_0, N_1 — number of preparations in $|0\rangle$ and $|1\rangle$ states
n_{00} — correct measurements when prepared in $|0\rangle$
n_{11} — correct measurements when prepared in $|1\rangle$
Typical values: 95–99% for standard readout, 99%+ with optimal filtering [14].

The Role of Statistics Because each measurement gives a probabilistic outcome, quantum computing fundamentally requires statistics:

- **Single-shot readout:** One measurement, one bit of information. Used in error correction where we need real-time decisions.
- **Repeated measurements:** Run the same circuit many times, build a histogram of outcomes. Used to estimate probabilities $|\alpha|^2, |\beta|^2$.
- **Expectation values:** Quantum algorithms often compute averages (e.g., $\langle Z \rangle = |\alpha|^2 - |\beta|^2$), requiring many "shots."

The number of shots required depends on the desired precision. To estimate a probability p with standard error σ:

$$N_{\text{shots}} = \frac{p(1-p)}{\sigma^2} \tag{17}$$

Variable Definitions for Eq. (16):
N_{shots} — number of measurements (circuit executions) required
p — true probability being estimated (unknown; use $p = 0.5$ for worst case)
σ — desired standard error on the probability estimate
Example: For $\sigma = 0.01$ (1% precision) and $p = 0.5$: $N = 0.25/0.0001 = 2500$ shots.

For 1% precision on a 50/50 outcome, this requires $\sim$2500 shots. This is why quantum computers run circuits thousands or millions of times.

5 The Readout Signal Processing Challenge

5.1 The Ensemble Averaging Approach

The traditional approach to improving measurement SNR is **ensemble averaging**: repeat the experiment N times and average the results. The SNR improvement scales as:

$$\text{SNR}_{\text{avg}} = \text{SNR}_{\text{single}} \cdot \sqrt{N} \tag{18}$$

Variable Definitions for Eq. (17):
SNR_{avg} — signal-to-noise ratio after averaging N measurements
$\text{SNR}_{\text{single}}$ — signal-to-noise ratio of a single measurement
N — number of repeated measurements averaged together
Note: The $\sqrt{N}$ improvement comes from averaging uncorrelated noise samples.

For example, to improve SNR by 10× requires 100 repetitions.

The Problem with Ensemble Averaging:

- **Time cost:** 100 repetitions at 10 μs each = 1 ms per measurement point
- **Incompatible with QEC:** Quantum error correction requires **single-shot** readout; we cannot average because each measurement is unique
- **Destroys quantum information:** Averaging collapses superposition states

5.2 The Single-Shot Requirement

Many quantum computing applications require **single-shot readout**, where each individual measurement must correctly identify the qubit state (Table IX). In these applications, ensemble averaging is not an option.

Table IX: Applications Requiring Single-Shot Readout

Application	Why Single-Shot is Required
Quantum Error Correction	Syndrome measurements must complete before errors accumulate
Active Reset	Must know current state to apply correction pulse
Mid-circuit Measurement	Algorithm branches based on measurement outcome
Quantum Teleportation	Classical communication of measurement results

For quantum error correction using the surface code, single-shot readout fidelity must exceed ∼99% to remain below the error threshold [15].

5.3 Optimal Filtering: The Signal Processing Solution

Rather than averaging multiple experiments, we can apply **optimal filtering** to a single readout pulse. The key insight is that not all time samples are equally informative:

- Early samples contain mostly noise (resonator hasn't responded yet)
- Late samples may be corrupted by T1 decay
- The optimal filter weights samples according to their information content

The **Wiener filter** provides the theoretically optimal linear filter for minimizing mean-squared error in Gaussian noise. For a readout pulse processed through an N_t-tap Wiener filter:

$$\boxed{\text{SNR}_{\text{out}} = \text{SNR}_{\text{in}} + 10\log_{10}(N_t)\ \text{dB}} \tag{19}$$

Variable Definitions for Eq. (18):
SNR_{out} — signal-to-noise ratio after filtering (dB)
SNR_{in} — signal-to-noise ratio before filtering (dB)
N_t — number of filter taps (filter length)
Note: This assumes white Gaussian noise and optimal (Wiener) filter weights.

In RTL simulation, a 64-tap filter provides +9 dB improvement, equivalent to the SNR gain from 64× ensemble averaging, but achieved in a **single shot**. Hardware validation with actual qubit readout signals is currently in progress.

Key Insight: Optimal filtering extracts the same information as ensemble averaging but from a single measurement. This enables single-shot applications while providing the noise reduction benefits of averaging. The theoretical framework is well-established; ongoing work focuses on validating performance with real qubit data.

6 Control Electronics: The QICK Platform

The Quantum Instrumentation Control Kit (QICK) is an open-source qubit control platform developed at Fermilab [16]. It provides the classical electronics interface for superconducting quantum computers, representing a paradigm shift from traditional rack-mounted instrumentation to integrated FPGA-based control.

6.1 System Architecture

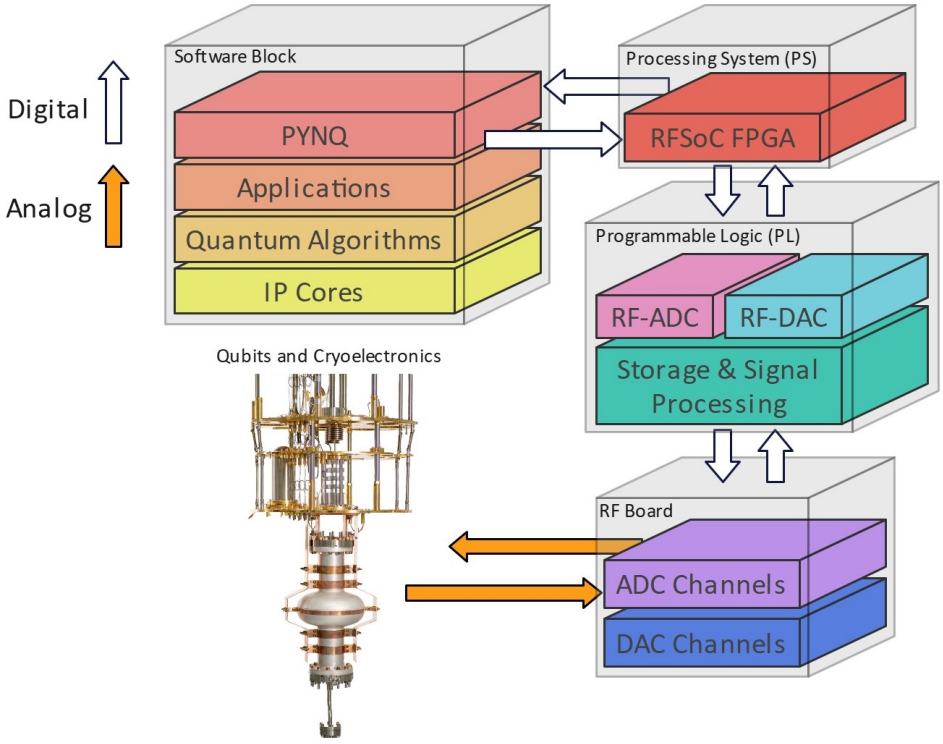

Figure 6: QICK system architecture showing the complete signal path from user software to qubits. **Software Block** (left): PYNQ provides the Python interface; applications and quantum algorithms are compiled into IP core instructions. **Processing System** (right, top): The RFSoC's ARM processors handle experiment orchestration. **Programmable Logic** (right, middle): RF-ADCs and RF-DACs interface directly with custom signal processing blocks. **RF Board and Cryogenics** (bottom): Analog signals travel to/from the dilution refrigerator containing the qubits. The orange arrows indicate the analog signal path; white arrows show digital control flow.

System Overview: Figure 6 illustrates the hierarchical structure of a QICK-based quantum control system, showing the flow from user-level Python code through the FPGA fabric to the quantum hardware.

In the QICK system, the readout process involves subsampling RF signals ranging from 4–8 GHz using a 2.5 GS/s ADC, effectively mapping the analog input spectrum into the first Nyquist zone (up to 1.25 GHz). The system employs a Numerically Controlled Oscillator (NCO) within the ADC for digital spectrum shifting, accommodating the phase noise increase inherent in subsampling (which remains below -70 dBc for 8 GHz signals at 100 Hz offset from carrier). The configuration and operational parameters—including timing, pulse lengths, and frequencies—are programmatically managed via Python through PYNQ, enabling flexible adaptation to experimental needs.

Hardware Components: QICK is built on AMD/Xilinx RFSoC (Radio Frequency System-on-Chip) FPGAs, which integrate high-speed DACs, ADCs, and programmable logic on a single chip (Table X).

Table X: QICK RFSoC Components

Component	**Specification**	**Function**
RF DACs	Up to 9.85 GSPS, 14-bit	Direct synthesis of control pulses up to 6 GHz
RF ADCs	Up to 4.096 GSPS, 12-bit	Digitize readout signals
Programmable Logic	∼900K logic cells	Real-time signal processing, tProcessor
ARM Processors	Quad-core Cortex-A53	Python interface, experiment control

Why FPGAs Instead of Traditional Instrumentation? Traditional qubit control systems rely on stacks of commercial arbitrary waveform generators (AWGs), local oscillators, mixers, and digitizers—often filling entire equipment racks for a single qubit [17]. This approach has several limitations:

- **Cost**: Commercial AWGs capable of direct microwave synthesis cost $50,000–$200,000 per channel
- **Synchronization**: Coordinating timing across multiple instruments introduces jitter and complexity
- **Latency**: Communication between rack instruments and control computers adds microseconds of delay, precluding real-time feedback
- **Scalability**: Adding qubits requires proportionally more equipment, cabling, and rack space

RFSoC FPGAs address these challenges by integrating high-speed data converters directly with programmable logic on a single chip. This enables:

- **Direct digital synthesis**: No external mixers or local oscillators needed for frequencies up to 6 GHz
- **Deterministic timing**: All operations locked to a single master clock with sub-nanosecond precision
- **Real-time processing**: Custom DSP blocks can process signals with nanosecond latency
- **Cost efficiency**: A complete multi-qubit control system fits on a single $5,000–$10,000 board

6.2 QICK Firmware Architecture

The QICK firmware is built around three core components: the **tProcessor** (tProc) for timing control, the **signal generator** (SG) blocks for pulse synthesis, and the **readout blocks** for signal acquisition and processing. Instructions are passed from the Zynq processor on the Processing System (PS) side to the Programmable Logic (PL) directly into the decentralized firmware. Signals travel from the DACs into the quantum system and are interpreted by the ADCs coming out of the dilution refrigerator.

6.2.1 QICK Version 1 Architecture

Figure 7 shows the tProcessor V1 block diagram and its associated signal paths.

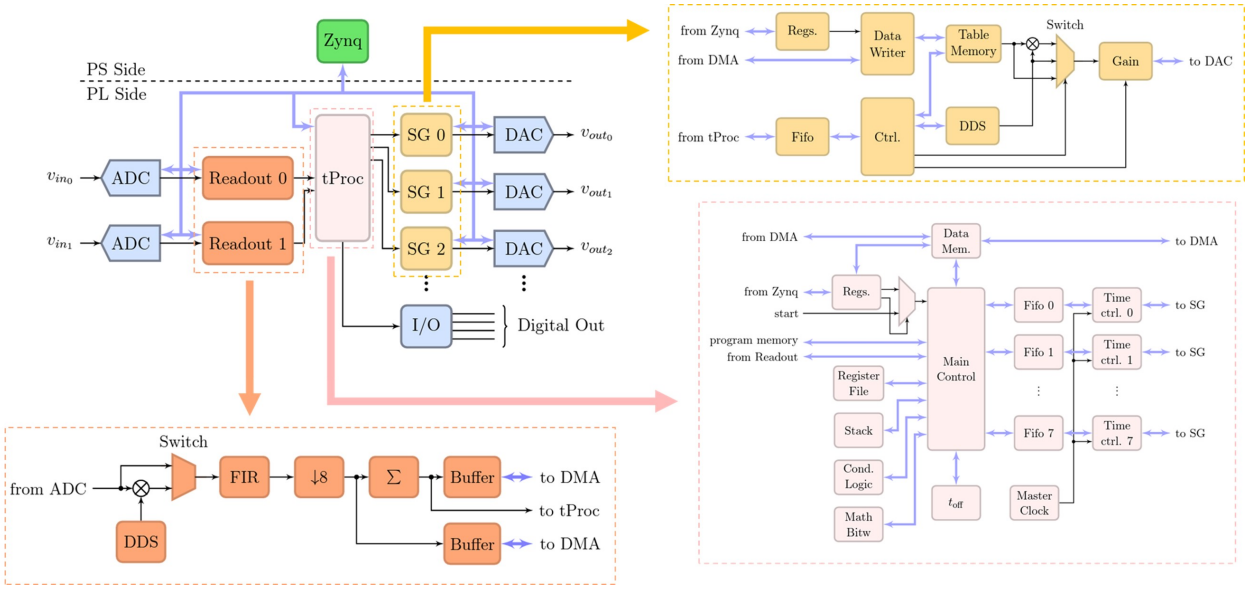

Figure 7: QICK Version 1 firmware architecture [16]. **Top left**: System overview showing the PS/PL boundary, with ADC inputs feeding readout blocks, the tProcessor coordinating timing, and signal generators (SG) driving DAC outputs. **Top right**: Single DAC output signal generator detail—user-specified waveforms are created in Python via PYNQ and sent into the signal generator block. Table memory stores IQ values characterizing the signal envelope; the fast DDS synthesizes the carrier tone for upconversion; a mux switch selects the upconversion mode; gain is applied before output to the DAC. **Bottom right**: 64-bit tProcessor V1 block diagram. The tProc handles differing time domains using t_{off} to align instructions. Signals are sent directly to signal generator blocks. Programs are written in a custom assembly language.

Readout Signal Processing (V1) The standard readout process in QICK V1 begins with high-speed samples received from the ADC. These samples are digitally downconverted using a rapid Direct Digital Synthesizer (DDS), with a switch to select the downconversion mode. Following downconversion, the signal undergoes low-pass filtering and is decimated by a factor of 8. The processed signal is then stored in buffers, which are managed and triggered by the tProcessor according to the specific needs of the quantum experiment.

The readout signal path utilizes a series of eight DDS channels for digital downconversion, covering the entire bandwidth of the ADC. This is followed by low-pass filtering and decimation tailored to specific requirements, with the data subsequently handled by buffering and averaging blocks optimized for rapid readout.

6.2.2 QICK Version 2 Architecture

Figure 8 shows the major improvements introduced in tProcessor V2, which was developed for improved co-design speedup and more sophisticated signal processing.

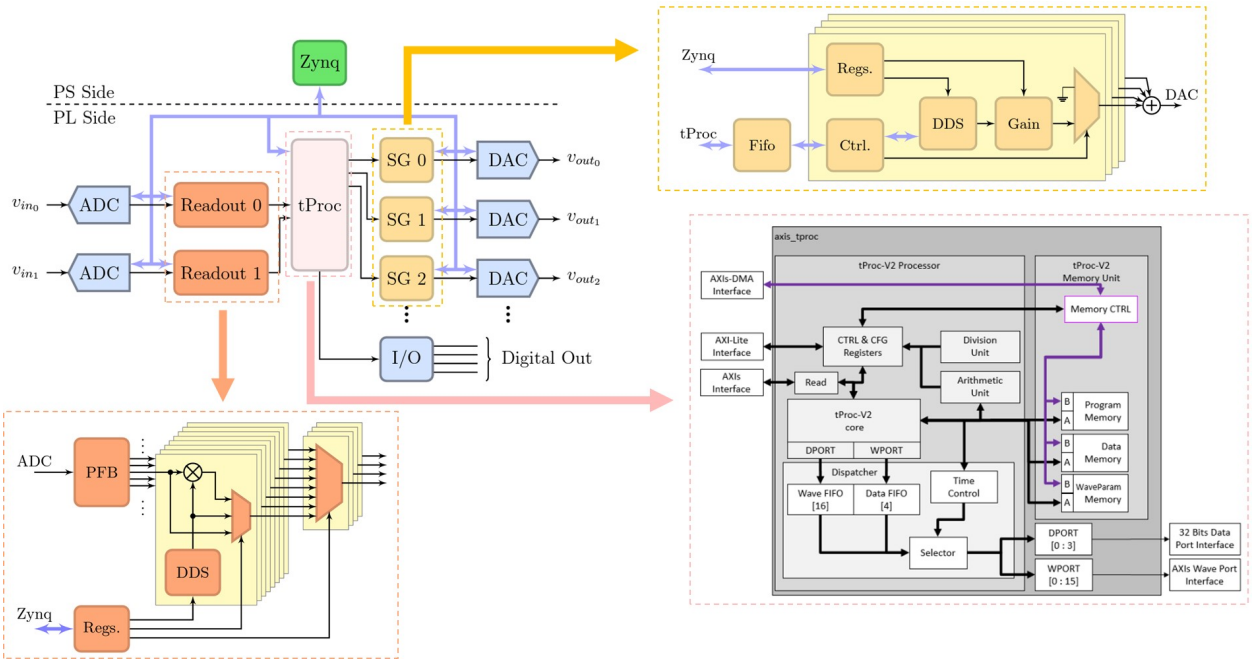

Figure 8: QICK Version 2 firmware architecture. **Top left**: System overview (same PS/PL structure as V1). **Top right**: Frequency-multiplexed DAC output—multiplexed signal generation on a single DAC output is achieved with independent DDS channels being digitally summed before transmission to the DAC. **Bottom left**: Multiplexed readout process—samples are processed through a polyphase filter bank (PFB) that demultiplexes incoming data into up to 16 channels. This setup uses 50% overlap between channels to maintain uniform gain across the entire bandwidth. The signal is then directed through DDS-based digital downconversion, followed by channel-specific processing. **Bottom right**: 72-bit tProcessor V2 block diagram built on Reduced Instruction Set Computer (RISC) architecture. Instructions are pipelined between processing functions and memory functions, enabling higher throughput.

Key Improvements in V2 The differences between Version 1 and Version 2 represent major improvements in co-design speedup:

- **RISC architecture**: The 72-bit tProc V2 uses a pipelined RISC design instead of the custom instruction set, improving instruction throughput
- **Polyphase filter bank**: Enables simultaneous processing of up to 16 frequency-multiplexed readout channels, critical for multi-qubit systems
- **Frequency-multiplexed output**: Multiple tones can be generated on a single DAC channel by digitally summing independent DDS outputs
- **Enhanced memory architecture**: Separate Program Memory, Data Memory, and WaveParam Memory with dual-port access for concurrent read/write

Key Insight: The evolution from QICK V1 to V2 reflects the scaling demands of quantum computing. V1's single-channel readout becomes a bottleneck when characterizing multi-qubit sys-

tems. V2's polyphase filter bank enables simultaneous readout of multiple qubits through frequency multiplexing—essential for systems with tens or hundreds of qubits sharing a single feedline.

6.3 The tProcessor: Timing and Synchronization

A common concern in qubit control is timing precision. The **tProcessor** (timed processor) is QICK's solution for deterministic, cycle-accurate control:

EE Analogy: The tProcessor is like a hardware sequencer with **absolute timestamping**. Every pulse and acquisition has a precise timestamp relative to a master clock, eliminating software-induced jitter.

Music Analogy: The tProcessor is like an orchestra conductor who keeps every musician in perfect sync. Without the conductor, the violins might rush, the brass might drag, and the piece falls apart. The tProcessor ensures every control pulse hits at exactly the right moment, keeping all the "instruments" (signal generators and digitizers) playing together.

Table XI: QICK tProcessor Timing Specifications

Parameter	Value	Significance
Timing jitter	**<2 ps RMS**	Negligible compared to qubit timescales
Total latency	184–211 ns	Deterministic, not variable
Instruction execution	Timed instructions	Events occur at absolute timestamps
Multi-board sync	<100 ps alignment	Enables scaling beyond single board

The key point, as shown in Table XI, is that **timing is deterministic**. When a readout pulse is triggered at time t_0:

- DAC output begins at t_0 (within clock cycle precision)
- ADC acquisition starts at $t_0 + \Delta t_{\text{propagation}}$
- All samples within a pulse are aligned to the same reference
- Pulse-to-pulse timing has no drift or jitter accumulation

Why Deterministic Timing Matters In software-based control systems (even real-time operating systems), timing has inherent variability. A command to "output a pulse now" may execute with microsecond-level jitter due to interrupt latency, cache misses, or OS scheduling. For qubit control, this is unacceptable for several reasons (Table XII).

EE Analogy: Consider a software-defined radio (SDR) where you need to correlate a received signal with a known template. If the sampling clock has jitter relative to the transmit clock, your correlation peak smears out and SNR degrades. The tProc eliminates this by ensuring that the "transmit" (DAC pulse) and "receive" (ADC acquisition) are locked to the same master clock with sub-nanosecond precision. It is analogous to a network analyzer where source and receiver are phase-locked.

How tProc Achieves Determinism Unlike a CPU that executes instructions "as fast as possible," the tProc executes **timed instructions**. Each instruction includes a timestamp specifying *when* it should execute, not just *what* it should do:

Table XII: Why Deterministic Timing Matters

Issue	Why It Matters	tProc Solution
Phase coherence	Qubit phase evolves at ∼5 GHz. A 1 ns timing error = 5 full rotations = completely wrong phase.	All pulses referenced to same clock; phase is computed, not measured
Pulse alignment	Multi-qubit gates require simultaneous pulses. Misalignment causes crosstalk and errors.	Absolute timestamps ensure pulses arrive simultaneously
Readout sync	ADC samples must align with the readout pulse envelope for coherent demodulation.	DAC and ADC share clock; acquisition window is precisely defined
Filter training	If timing varies between training and inference, the learned filter weights are invalid.	Every experiment has identical timing; training data matches inference conditions

Example tProc Instruction:
`PULSE(channel=0, freq=5.0 GHz, phase=0, amplitude=0.5, t_start=1000 ns, duration=20 ns)`
The pulse will begin at *exactly* 1000 ns after the sequence start, regardless of how long previous instructions took to process. The tProc pre-computes the waveform and loads it into the DAC buffer ahead of time; at t=1000 ns, it simply releases the trigger.

This architecture means that the latency from "decide to do something" to "thing happens" is fixed and known (184–211 ns depending on the operation). For feedback control (e.g., active qubit reset), this deterministic latency can be accounted for in the control algorithm. There is no variable delay that would require real-time compensation.

For adaptive filtering, deterministic timing ensures that:

- The signal template (qubit response shape) is identical across all training pulses
- Sample indices correspond to the same physical time relative to the pulse
- Phase relationships between I and Q channels are preserved
- Filter weights learned during training apply correctly during inference

7 The Current State of Quantum Computing

7.1 NISQ, FTQC, and FASQ

The quantum computing community uses several terms to describe the evolutionary stages of the technology (Table XIII) [18].

Table XIII: Quantum Computing Eras

Era	Full Name	Characteristics	Status
NISQ	Noisy Intermediate-Scale Quantum	50–1000 qubits, no QEC, limited depth	Current
FTQC	Fault-Tolerant QC	Error-corrected logical qubits	Early demos
FASQ	Fault-tolerant Application-Scale	Useful error-corrected computation	Future goal

John Preskill, who coined "NISQ" in 2018, recently introduced the concept of **"megaquop machines"** as milestones on the path from NISQ to FASQ. A megaquop machine can execute approximately one million quantum operations with sufficient fidelity [19]. Some researchers have expressed caution about overpromising quantum speedups in the near term [20].

7.2 The Error Budget

Current quantum computers are limited by cumulative errors. As shown in Table XIV, readout error is often the largest contributor to the typical error budget for a superconducting system.

Table XIV: Quantum Computing Error Budget

Error Source	Typical Rate	Trend
Two-qubit gates	0.1–1%	Improving with better calibration [21]
Single-qubit gates	0.01–0.1%	Approaching physical limits
Readout	**0.5–5%**	**Major bottleneck**
State preparation	0.1–1%	Limited by thermal population
Idle error (per μs)	0.01–0.1%	T1, T2 improvements ongoing

Unlike gate errors, readout error cannot be reduced by simply shortening the operation (shorter readout means lower SNR). and unlike gate errors, it cannot be reduced by simply shortening the operation (shorter readout means lower SNR).

7.3 Why Readout Matters for Error Correction

Quantum error correction (QEC) works by encoding a single **logical qubit** in many **physical qubits** and repeatedly measuring "syndrome" qubits to detect errors. For the surface code:

- Syndrome measurements must complete in $\sim$1 μs (before errors accumulate)
- Each syndrome measurement requires single-shot readout
- Readout errors can be mistaken for data errors, causing incorrect corrections
- The error correction threshold requires physical error rates below $\sim$1%

Improving readout fidelity and speed directly enables practical quantum error correction.

Quantitative QEC-Readiness: The surface code per-step error threshold is approximately 0.57–1.1%, depending on the noise model and decoder [15]. For readout specifically, assignment fidelity above 99.5% is needed to prevent readout from dominating the error budget. Using the Gambetta formula (Eq. 26), this requires a voltage-domain SNR of approximately 5.15 ($\sim$11 dB in power).

A useful design criterion is the **QEC-readiness condition**: the combined hardware SNR plus digital processing gain must exceed the QEC threshold SNR. If the hardware baseline is 0 dB (typical for HEMT-amplified systems without quantum-limited amplifiers), then approximately 11–13 dB of processing gain is needed, achievable with a 64-tap optimal filter ($10\log_{10}(64) = 18$ dB maximum). Systems with quantum-limited amplifiers (TWPA/JPA at +15 dB baseline) already exceed this threshold without additional digital filtering.

8 Qubit Performance Metrics and Platform Comparison

Understanding qubit performance requires familiarity with several key metrics. This section defines the critical parameters and compares current performance across different quantum computing platforms.

8.1 Key Qubit Metrics

T1: Energy Relaxation Time **T1** measures how long a qubit remains in the excited state $|1\rangle$ before spontaneously decaying to the ground state $|0\rangle$. This is analogous to the time constant of an RC discharge circuit.

$$P_{|1\rangle}(t) = P_{|1\rangle}(0) \cdot e^{-t/T_1} \tag{20}$$

Variable Definitions for Eq. (19):
$P_{|1\rangle}(t)$ — probability of finding qubit in excited state $|1\rangle$ at time t
$P_{|1\rangle}(0)$ — initial probability in $|1\rangle$ (typically 1 if prepared in $|1\rangle$)
T_1 — energy relaxation time constant (seconds)
t — elapsed time since state preparation (seconds)

EE Analogy: T1 is the quantum equivalent of a capacitor's discharge time constant. Just as a charged capacitor loses energy to its environment through resistive losses, an excited qubit loses energy to its environment through various dissipation mechanisms (photon emission, phonon coupling, etc.).

Music Analogy: T1 is like how long a guitar string keeps ringing after you pluck it. A high-quality guitar in a quiet room rings for a long time; a cheap guitar in a noisy room dies out quickly. Similarly, a well-isolated qubit holds its energy longer. When the "string" stops vibrating, the qubit has lost its excited state.

T2: Dephasing (Coherence) Time **T2** measures how long a qubit maintains phase coherence in a superposition state. Even without energy loss, environmental fluctuations can randomize the relative phase between $|0\rangle$ and $|1\rangle$.

$$T_2 \leq 2T_1 \quad \text{(fundamental limit)} \tag{21}$$

Variable Definitions for Eq. (20):
T_2 — dephasing (coherence) time; how long phase information is preserved (seconds)
T_1 — energy relaxation time (seconds)
The factor of 2 arises because energy decay also causes phase loss. If $T_2 < 2T_1$, additional "pure dephasing" processes are present.

In practice, T2 is often much shorter than 2T1 due to **pure dephasing** from low-frequency noise [22] (magnetic field fluctuations, charge noise, etc.). Several variants are commonly reported (Table XV).

Table XV: T2 Measurement Variants

Metric	Measurement	What It Captures
T2* (Ramsey)	Free precession after $\pi/2$ pulse	Total dephasing including slow drift
T2 (Echo)	Hahn echo ($\pi/2$ - π - $\pi/2$)	Dephasing with low-freq noise refocused
T2 (CPMG/DD)	Multiple refocusing pulses	Best-case coherence with dynamical decoupling

EE Analogy: T2 is analogous to **phase noise** in an oscillator. A qubit in superposition acts like a quantum oscillator; T2 measures how long the "clock" stays synchronized before environmental noise causes the phase to drift unpredictably. The echo techniques are similar to phase-locked loops that track out slow frequency drift.

Music Analogy: T2 is like how long an orchestra can play without a conductor before the musicians drift out of sync. At first, everyone is perfectly together. But over time, tiny timing differences accumulate until the ensemble falls apart. T2 measures how long the qubit's "rhythm" stays in sync before environmental noise throws it off beat.

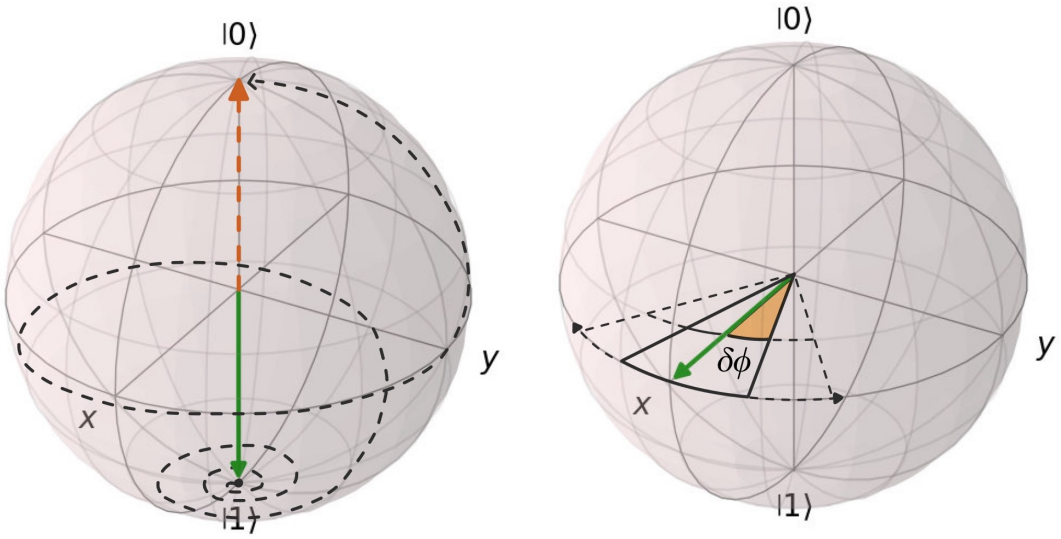

Figure 9: Bloch sphere visualization of decoherence mechanisms (see Section 2 for Bloch sphere introduction). (a) **T1 relaxation**: The qubit state decays along the z-axis from $|1\rangle$ (south pole) toward $|0\rangle$ (north pole). The green arrow shows a state initially in $|1\rangle$; over time $\sim T_1$, it relaxes to the ground state (orange arrow). (b) **T2 dephasing**: A superposition state on the equator (equal probability of $|0\rangle$ and $|1\rangle$) loses phase coherence. The angle $\delta\phi$ represents accumulated phase uncertainty; over time $\sim T_2$, the phase becomes randomized, collapsing the state toward the z-axis.

Rabi Oscillation and Gate Time When a qubit is driven by a resonant microwave pulse, its state oscillates between $|0\rangle$ and $|1\rangle$ at the **Rabi frequency** Ω_R, which is proportional to the drive amplitude.

$$|\psi(t)\rangle = \cos\left(\frac{\Omega_R t}{2}\right)|0\rangle - i\sin\left(\frac{\Omega_R t}{2}\right)|1\rangle \tag{22}$$

Variable Definitions for Eq. (21):
$|\psi(t)\rangle$ — qubit state at time t during a resonant drive
Ω_R — Rabi frequency (rad/s); proportional to drive amplitude
t — time since drive started (seconds)
$|0\rangle, |1\rangle$ — ground and excited states
The $-i$ phase factor arises from the rotating frame convention.

A π**-pulse** (duration $t_\pi = \pi/\Omega_R$) flips the qubit from $|0\rangle$ to $|1\rangle$. Typical single-qubit gate times:

- Superconducting qubits: 20–50 ns
- Trapped ions: 1–100 μs
- Neutral atoms: 0.1–1 μs

Gate Fidelity **Fidelity** measures how accurately a gate operation is performed. For a target operation U and actual operation $\mathcal{E}$:

$$F = \left\langle \psi \left| U^\dagger \mathcal{E}(|\psi\rangle\langle\psi|) U \right| \psi \right\rangle \tag{23}$$

Variable Definitions for Eq. (22):
F — gate fidelity (dimensionless, 0 to 1); often quoted as percentage
U — target (ideal) unitary gate operation
$\mathcal{E}$ — actual quantum channel implemented by the hardware
$|\psi\rangle$ — input quantum state
$U^\dagger$ — conjugate transpose of U
In practice, fidelity is averaged over input states or measured via randomized benchmarking.

Fidelity is often measured via **randomized benchmarking** [23], which extracts the average error per gate by running random sequences of gates. Current state-of-the-art:

- Single-qubit gates: 99.9–99.99% (error 0.01–0.1%)
- Two-qubit gates: 99–99.9% (error 0.1–1%)

The Quality Factor: Operations per Coherence Time

Key Insight: Raw coherence time is less important than the **number of operations achievable within T2**. A platform with T2 = 100 μs and 50 ns gates can perform ∼2000 operations before decoherence dominates. A platform with T2 = 1 ms but 10 μs gates can only perform ∼100 operations. The relevant figure of merit is:

$$N_{\text{ops}} = \frac{T_2}{t_{\text{gate}}} \tag{24}$$

Variable Definitions for Eq. (23):
N_{ops} — number of gate operations achievable before decoherence dominates
T_2 — dephasing (coherence) time
t_{gate} — duration of a single gate operation

Distinguishing Fidelity Types The term "fidelity" is used for several distinct metrics (Table XVI). Understanding the differences is critical for interpreting published results [24].

Table XVI: Distinguishing Fidelity Types

Fidelity Type	What It Measures	How Measured	Typical Values
Single-Qubit Gate	Rotation accuracy (X, Y, Z)	Randomized benchmarking [24]	99.9–99.99% [25]
Two-Qubit Gate	Entangling ops (CNOT, CZ)	Interleaved RB, XEB	99.0–99.9% [25, 26]
Readout	State discrimination	Confusion matrix [14]	95–99%
State Preparation	Initializing $\|0\rangle$	Measure after long wait	99–99.5%
SPAM	Prep + Measurement	Measure after reset	98–99%

Readout Fidelity vs. Gate Fidelity:
Gate fidelity measures how well we *control* the qubit (rotate it to the intended state).
Readout fidelity measures how well we *observe* the qubit (correctly identify its state).
Both contribute to the total error in a quantum circuit. A circuit with 100 gates and one final measurement has error contributions from all 100 gates AND the readout. Improving readout fidelity is especially important for:

- Quantum error correction (syndrome measurements must be accurate)
- Variational algorithms (many repeated measurements to estimate expectation values)
- Mid-circuit measurements (branching logic depends on correct classification)

Quantitative Scaling Example Consider a circuit with depth d (number of sequential gate layers) running on n qubits. The total circuit fidelity approximately follows:

$$F_{\text{circuit}} \approx (1-\epsilon_{1Q})^{n \cdot d_{1Q}} \cdot (1-\epsilon_{2Q})^{n_{\text{2Q gates}}} \cdot (1-\epsilon_{\text{readout}})^{n} \tag{25}$$

Variable Definitions for Eq. (24):
F_{circuit} — overall circuit fidelity (probability of no errors)
ϵ_{1Q} — single-qubit gate error rate
ϵ_{2Q} — two-qubit gate error rate
$\epsilon_{\text{readout}}$ — readout error rate
n — number of qubits
d_{1Q} — number of single-qubit gate layers

$n_{2Q\ \text{gates}}$ — total number of two-qubit gates
Note: This assumes independent errors; correlated errors (from crosstalk [27, 28]) make the situation worse.

Example: A 100-qubit circuit with 50 layers of 2Q gates, using qubits with $\epsilon_{2Q} = 0.5\%$ and $\epsilon_{\text{readout}} = 2\%$:

- 2Q gate contribution: $(0.995)^{2500} \approx 3.5 \times 10^{-6}$ (essentially zero)
- Readout contribution: $(0.98)^{100} \approx 0.13$
- Combined: Circuit success probability is negligible without error correction

The Scaling Wall: This exponential decay of fidelity with circuit size is why quantum error correction is essential. Without it, useful quantum computations on 100+ qubits are impossible with current error rates. The threshold theorem guarantees that if physical error rates are below ∼1%, error correction can suppress logical error rates arbitrarily low, but this requires 100–1000 physical qubits per logical qubit.

SNR-to-Fidelity (Gambetta Formula): For dispersive readout of superconducting qubits, the ideal assignment fidelity relates to the signal-to-noise ratio through:

$$F_{\text{assign}} = \frac{1}{2}\left[1 + \text{erf}\left(\frac{\text{SNR}}{2\sqrt{2}}\right)\right] \tag{26}$$

Variable Definitions for Eq. (26):
F_{assign} — assignment fidelity (probability of correct state classification)
SNR — voltage-domain signal-to-noise ratio (d/σ, full IQ blob separation divided by per-quadrature noise)
erf — the error function
This formula assumes two Gaussian-distributed IQ measurement outcomes with equal variance (Gambetta et al. 2006, 2007; Chen et al. 2023).

Table XVII shows representative values.

Table XVII: SNR vs. Assignment Fidelity

Voltage SNR (d/σ)	**Power SNR (dB)**	F_{assign}
3.29	7.34 dB	95%
4.65	10.35 dB	99%
5.15	11.23 dB	99.5%
6.18	12.81 dB	99.9%

This formula quantifies the gap between current hardware SNR and the fidelity requirements for quantum error correction. For EE readers, this is a standard result from detection theory: the probability of correctly classifying a signal in Gaussian noise via a threshold detector.

SNR Convention Warning: The quantum computing literature uses multiple SNR conventions. "Voltage SNR" ($20\log_{10}(d/\sigma)$) and "power SNR per quadrature" ($10\log_{10}(d^2/(2\sigma^2))$) differ by ∼3 dB at the same operating point. Always check which convention a paper uses before comparing numbers.

8.2 Quantum Computing Platform Comparison

Different physical implementations of qubits (also known as **modalities**) offer distinct tradeoffs. Table XVIII summarizes current state-of-the-art performance across major platforms.

Table XVIII: Quantum Computing Platform Comparison

Platform	T1	T2	1Q Gate	2Q Fidelity	Max Qubits
Supercond. (Industry) [25, 29, 30]	50–200 μs	50–150 μs	20–50 ns	99.0–99.5%	1,121 (IBM)
Supercond. (Lab best) [31]	**1.6 ms**	**~1 ms**	20–50 ns	99.5%	6 (Princeton)
SQMS (Fermilab) [9, 32]	**0.6 ms**	0.3 ms	20–50 ns	~99%	9 (Rigetti)
SQMS 3D SRF [33, 34]	**>2 s**	**>20 ms**	~100 ns	~99%	2 qudits
Trapped Ions [26, 35–37]	>10 s	1–10 s	1–10 μs	**99.9–99.99%**	56 (H2)
Neutral Atoms [38, 39]	**119 s**	12.6 s	0.1–1 μs	99.0–99.5%	6,100
Silicon Spin [40–42]	9.5 s	1.9 ms	1–10 μs	>99%	~10
Photonic [43, 44]	N/A	Limited	~ps	~99.2%	216+ modes

Modality Summary: Strengths and Drawbacks **Superconducting Qubits:** The dominant modality, likely to remain so for the foreseeable future.

- **Strengths:** Fastest gate times (20–50 ns) enable deep circuits; proven scalability to 1000+ qubits; decades of materials science investment from IBM, Google, and startups. Fabrication uses standard semiconductor fab *infrastructure* (cleanrooms, lithography tools, deposition systems).
- **Drawbacks:** Traditional Josephson junction fabrication uses specialized processes (double-angle evaporation, e-beam lithography, lift-off) that differ from high-volume CMOS process flows (though 2024 Imec work demonstrated CMOS-compatible methods [45]). Requires expensive dilution refrigerators [46] (~$0.5–3M, plus ^{3}He costs [47]) at 10–20 mK; short coherence times (50–200 μs typical) demand aggressive error correction.

Trapped Ions: Strong contender for early fault-tolerant demonstrations.

- **Strengths:** Highest gate fidelities (99.9–99.99%) and longest coherence times (seconds to minutes) with all-to-all connectivity via shared motional modes.
- **Drawbacks:** Slow gate operations (1–10 μs, ~100× slower than superconducting); scaling beyond ~50 qubits requires complex ion shuttling or photonic interconnects; precision laser systems and UHV chambers add cost and complexity; no path to standard semiconductor foundry fabrication.

Neutral Atoms: Strong candidate for early error-corrected systems.

- **Strengths:** Exceptional scalability (6,100 qubits demonstrated) with reconfigurable connectivity via optical tweezers; very long coherence (T1 > 100 s); no cryogenics required, as it operates near room temperature. Mid-circuit measurement and atom rearrangement capabilities.
- **Drawbacks:** Slow cycle times due to atom loading and rearrangement; finite Rydberg state lifetime (~150 μs) limits gate depth; atom loss during computation; requires precision laser systems and vacuum chambers; no semiconductor fab pathway.

Silicon Spin Qubits: Emerging contender with CMOS foundry compatibility.

- **Strengths:** Direct compatibility with 300mm CMOS foundries; exceptionally long T1 (seconds); potential for high-density integration using existing semiconductor infrastructure; demonstrated >99% gate fidelities.
- **Drawbacks:** Requires cryogenic operation (~100 mK, warmer than superconducting); slower gates than superconducting; two-qubit gate fidelities still improving; smaller demonstrated qubit counts than other modalities.

Why Superconducting Dominates: Superconducting qubits benefit from a 25+ year head start in academic research, billions of dollars in corporate R&D (IBM, Google, Rigetti, IQM), and the ability to leverage existing semiconductor fabrication infrastructure, even if not standard CMOS process flows. The combination of fast gates, proven multi-qubit integration, and mature control electronics has created a self-reinforcing ecosystem. While other modalities show promise in specific metrics (trapped ions in fidelity, neutral atoms in qubit count, silicon spin in foundry compatibility), superconducting platforms currently offer the best balance of speed, scale, and engineering maturity for near-term applications.

8.3 SQMS Contributions to Coherence

The Superconducting Quantum Materials and Systems (SQMS) Center at Fermilab has made significant contributions to extending qubit coherence through materials science innovations [9]:

Surface Encapsulation Technique SQMS researchers identified the niobium surface oxide as the primary source of energy loss in transmon qubits. Their **surface encapsulation** technique prevents the formation of lossy niobium oxide by capping with tantalum or gold during fabrication:

- Native Nb oxide: high loss, short T1
- Ta-capped Nb: 2–5× improvement in T1
- Best results: T1 up to 600 μs, median >300 μs

This technique is scalable and has been demonstrated in Rigetti's commercial foundry, enabling direct transfer to industry.

3D SRF Cavity Architecture SQMS leverages Fermilab's expertise in superconducting radio-frequency (SRF) cavities from particle accelerator development. By using ultra-high-Q SRF cavities ($\sim 10^{10}$) as quantum memories:

- Cavity photon lifetimes exceeding 2 seconds demonstrated
- Two-qudit QPU with >20 ms coherence (record for multimode superconducting system)
- Qudit encoding: single cavity stores multiple quantum levels, reducing hardware overhead

The computational advantage of qudits comes from the exponential scaling of Hilbert space dimension. For n qubits, the Hilbert space has dimension 2^n. For n qudits, each with d accessible levels, the dimension becomes:

$$\mathcal{D}_{\text{qudit}} = d^n \tag{27}$$

The equivalent number of qubits required to span the same Hilbert space is:

$$N_{\text{eff}} = n \cdot \log_2(d) \tag{28}$$

Variable Definitions for Eqs. (25)–(26):
$\mathcal{D}_{\text{qudit}}$ — Hilbert space dimension for a qudit-based system
d — Number of accessible levels per qudit (e.g., Fock states in a cavity)

n — Number of qudits (e.g., number of SRF cavities)
N_{eff} — Equivalent number of qubits to achieve the same Hilbert space dimension

For example, a single 10-level qudit ($d = 10$) provides $\log_2(10) \approx 3.3$ qubits worth of computational space. SQMS has demonstrated control of cavities with approximately 30 Fock states [33], equivalent to nearly 5 qubits per cavity. A system of 100 such cavities would span a Hilbert space equivalent to $\sim$500 qubits, but with dramatically reduced hardware complexity: 100 physical cavities versus 500 physical qubits, each requiring individual control lines, readout resonators, and calibration.

SQMS Vision: The Center aims to achieve 10 ms coherence in chip-based transmon qubits and deploy a 100+ qudit SRF quantum processor (equivalent to $\sim$500 qubits in computational space) within a single dilution refrigerator.

9 Application: Adaptive Filtering for Qubit Readout

This section describes how adaptive Wiener filtering [48, 49] can address the readout bottleneck described in previous sections. The detailed algorithm and implementation are covered in the companion document, "Block NLMS Adaptive Wiener Filter for Superconducting Qubit Readout."

9.1 The Filtering Approach

Instead of ensemble averaging (which requires multiple experiments) or fixed matched filtering (which assumes known signal/noise statistics), an **adaptive filter** learns the optimal weights from training data. As shown in Table XIX, adaptive filtering offers the best balance of optimality and flexibility.

Table XIX: Comparison of Filtering Approaches

Approach	Advantages	Disadvantages
Ensemble averaging	Simple, no training needed	Requires N experiments, incompatible with single-shot
Fixed matched filter	Optimal for known statistics	Must re-derive if system changes
Adaptive filter	Learns from data, adapts to drift	Requires training phase

9.2 Training and Inference

The adaptive filter operates in two phases:

Training Phase

1. Prepare qubit in $|0\rangle$ (ground state, by waiting for T1 decay)
2. Collect many readout pulses $\rightarrow$ label as class "0"
3. Apply π-pulse to prepare $|1\rangle$
4. Collect many readout pulses $\rightarrow$ label as class "1"
5. LMS algorithm iteratively adjusts filter weights to minimize classification error

Inference Phase

1. Freeze filter weights (no further adaptation)
2. For each readout pulse, apply fixed weights
3. Output is a single (I, Q) point with improved SNR
4. Classify based on decision boundary

9.3 Performance Summary

The key performance metrics (Table XX) demonstrate that adaptive filtering meets the requirements for QEC-compatible readout.

Table XX: Adaptive Filter Performance Summary

Metric	Value	Significance
SNR improvement	+9 dB (64 taps)	Equivalent to 64× averaging
Latency	<1 μs	Compatible with QEC timing
Training time	∼1000 pulses	Practical for calibration routines
Implementation	FPGA (QICK-compatible)	Real-time, no software latency

Validation Status: The performance metrics above have been verified through RTL simulation on FPGA using synthetic Gaussian noise models. Hardware validation with actual superconducting qubit readout signals is currently in progress.

9.4 Validation Strategy

Verification that the filter implementation is correct requires multiple complementary approaches (Table XXI).

Table XXI: Filter Validation Strategy

Validation Method	What It Checks
Bit-accurate simulation	HDL matches Python golden reference cycle-by-cycle
Convergence to Wiener solution	Weight sum matches theoretical $G_{\text{opt}} = \text{SNR}/(1+\text{SNR})$
SNR-dependent behavior	High SNR $\rightarrow$ unity gain; Low SNR $\rightarrow$ attenuation
MSE minimization	Error decreases monotonically during training

10 Mathematical Foundations (Reference)

This section collects the key mathematical results for readers seeking deeper understanding. It can be skipped without loss of continuity.

10.1 Transmon Hamiltonian

Mathematical Detail: Transmon Energy Levels
The transmon Hamiltonian in the phase basis:

$$\hat{H} = 4E_C(\hat{n} - n_g)^2 - E_J \cos\hat{\phi} \tag{27}$$

where $\hat{n}$ and $\hat{\phi}$ are conjugate variables: $[\hat{\phi}, \hat{n}] = i$.
In the transmon regime ($E_J/E_C \gg 1$), the energy levels are approximately:

$$E_m \approx -E_J + \sqrt{8E_J E_C}\left(m + \frac{1}{2}\right) - \frac{E_C}{12}(6m^2 + 6m + 3) \tag{28}$$

The transition frequencies are:

$$\hbar\omega_{01} \approx \sqrt{8E_J E_C} - E_C, \quad \alpha = \omega_{01} - \omega_{12} \approx -E_C/\hbar \tag{29}$$

10.2 Dispersive Readout

Mathematical Detail: Dispersive Hamiltonian
The qubit-resonator system in the dispersive regime:

$$\hat{H}_{\text{disp}} = \hbar\omega_r \hat{a}^\dagger\hat{a} + \frac{\hbar\omega_q}{2}\hat{\sigma}_z + \hbar\chi\hat{a}^\dagger\hat{a}\hat{\sigma}_z \tag{30}$$

The dispersive shift is:

$$\chi = \frac{g^2}{\Delta}\frac{\alpha}{\Delta + \alpha} \tag{31}$$

where g is the qubit-resonator coupling and $\Delta = \omega_q - \omega_r$ is the detuning.
The resonator frequency conditioned on qubit state:

$$\omega_r^{|0\rangle} = \omega_r + \chi, \quad \omega_r^{|1\rangle} = \omega_r - \chi \tag{32}$$

10.3 Wiener Filter Theory

Mathematical Detail: Optimal Wiener Filter
The Wiener filter minimizes mean-squared error:

$$\mathbf{w}_{\text{opt}} = \mathbf{R}_{xx}^{-1}\mathbf{r}_{xd} \tag{33}$$

where $\mathbf{R}_{xx}$ is the input autocorrelation matrix and $\mathbf{r}_{xd}$ is the cross-correlation between input and desired output.
For white noise and a known signal template, the optimal gain is:

$$G_{\text{opt}} = \frac{\text{SNR}}{1 + \text{SNR}} = \frac{P_{\text{signal}}}{P_{\text{signal}} + P_{\text{noise}}} \tag{34}$$

The SNR improvement from an N_t-tap filter:

$$\mathrm{SNR}_{\mathrm{out}} = \mathrm{SNR}_{\mathrm{in}} + 10\log_{10}(N_t)\ \mathrm{dB} \tag{35}$$

10.4 LMS Algorithm

Mathematical Detail: LMS Weight Update
The Least Mean Squares (LMS) algorithm [49] updates weights iteratively:

$$\mathbf{w}[n+1] = \mathbf{w}[n] + \mu \cdot e^*[n] \cdot \mathbf{x}[n] \tag{36}$$

where:

- μ = step size (learning rate)
- $e[n] = d[n] - y[n]$ = error signal
- $y[n] = \mathbf{w}^H[n]\mathbf{x}[n]$ = filter output

The Block NLMS variant normalizes by input power:

$$\mathbf{w}[n+1] = \mathbf{w}[n] + \frac{\mu}{\|\mathbf{x}\|^2 + \epsilon} \cdot e^*[n] \cdot \mathbf{x}[n] \tag{37}$$

This provides faster, more stable convergence across varying SNR conditions.

10.5 Qudit Hilbert Space Scaling

Mathematical Detail: Qudit Computational Advantage
The Hilbert space dimension for n qubits is 2^n. For qudits with d levels each:

$$\mathcal{D}_{\mathrm{qudit}} = d^n \tag{25}$$

The equivalent number of qubits required to achieve the same Hilbert space dimension:

$$N_{\mathrm{eff}} = n \cdot \log_2(d) \tag{26}$$

where:

- $\mathcal{D}_{\mathrm{qudit}}$ = Hilbert space dimension for a qudit-based system
- d = number of accessible levels per qudit (e.g., Fock states in a cavity)
- n = number of qudits (e.g., number of SRF cavities)
- N_{eff} = equivalent number of qubits

For SRF cavities with $d \approx 30$ controllable Fock states, each cavity provides $\log_2(30) \approx 4.9$ qubits of computational space. The hardware reduction factor is $d/\log_2(d)$, which equals $\sim 6\times$ for $d = 30$.

11 Conclusion

Superconducting quantum computing, despite its quantum mechanical foundations, is largely an electrical engineering endeavor. The qubit is a nonlinear LC oscillator; control and readout involve RF pulse generation and detection; and improving performance requires signal processing techniques familiar to any EE.

The readout bottleneck, in particular, is a classic signal processing problem: discriminating two Gaussian-distributed classes in the presence of noise. Adaptive filtering techniques, long used in communications and radar, can provide significant improvements in single-shot readout fidelity, directly enabling quantum error correction.

Summary of Key Points:

- A transmon qubit is a Josephson junction shunted by a capacitor, forming a nonlinear LC oscillator
- The Josephson junction acts as a current-dependent inductor: $L_J = \Phi_0/(2\pi I_c \cos\delta)$
- Nonlinearity creates anharmonic energy levels, enabling selective $|0\rangle \leftrightarrow |1\rangle$ addressing
- Dispersive readout measures state-dependent resonator frequency shift ($\sim$1–5 MHz)
- Single-shot readout is required for QEC; ensemble averaging is not sufficient
- Optimal filtering can achieve the same SNR improvement as averaging, in a single shot
- QICK provides deterministic timing ($<$2 ps jitter) for coherent control
- Readout error (0.5–5%) is currently a major bottleneck; improving it enables QEC

For detailed treatment of the adaptive Wiener filter implementation, see the companion document: "Block NLMS Adaptive Wiener Filter for Superconducting Qubit Readout."

References

[1] Peter W. Shor. Algorithms for quantum computation: Discrete logarithms and factoring. In *Proc. 35th Annual Symposium on Foundations of Computer Science*, pages 124–134, 1994. doi: 10.1109/SFCS.1994.365700.

[2] Lov K. Grover. A fast quantum mechanical algorithm for database search. In *Proc. 28th Annual ACM Symposium on Theory of Computing*, pages 212–219, 1996. doi: 10.1145/237814.237866.

[3] Aram W. Harrow, Avinatan Hassidim, and Seth Lloyd. Quantum algorithm for linear systems of equations. *Physical Review Letters*, 103:150502, 2009. doi: 10.1103/PhysRevLett.103.150502.

[4] David J. Griffiths. *Introduction to Quantum Mechanics*. Cambridge University Press, 3rd edition, 2018. ISBN 978-1107189638.

[5] Brian D. Josephson. Possible new effects in superconductive tunnelling. *Physics Letters*, 1(7):251–253, 1962. doi: 10.1016/0031-9163(62)91369-0.

[6] Jens Koch et al. Charge-insensitive qubit design derived from the Cooper pair box. *Physical Review A*, 76(4):042319, 2007.

[7] Thomas E. Roth, Ruichao Ma, and Weng C. Chew. The transmon qubit for electromagnetics engineers: An introduction. *IEEE Antennas and Propagation Magazine*, 65(2), April 2023. doi: 10.1109/MAP.2022.3176593.

[8] Alexandre Blais, Arne L. Grimsmo, Steven M. Girvin, and Andreas Wallraff. Circuit quantum electrodynamics. *Reviews of Modern Physics*, 93(2):025005, 2021.

[9] Murat Bal, Akshay A. Murthy, Shaojiang Zhu, Francesco Crisa, et al. Systematic improvements in transmon qubit coherence enabled by niobium surface encapsulation. *npj Quantum Information*, 10:43, 2024. SQMS surface encapsulation, T1 up to 0.6 ms.

[10] John Bardeen, Leon N. Cooper, and John Robert Schrieffer. Theory of superconductivity. *Physical Review*, 108(5):1175–1204, 1957. doi: 10.1103/PhysRev.108.1175.

[11] Leon N. Cooper. Bound electron pairs in a degenerate Fermi gas. *Physical Review*, 104(4):1189–1190, 1956. doi: 10.1103/PhysRev.104.1189.

[12] Michael A. Nielsen and Isaac L. Chuang. *Quantum Computation and Quantum Information*. Cambridge University Press, 10th anniversary edition, 2010. ISBN 978-1107002173.

[13] Michel H. Devoret and Robert J. Schoelkopf. Superconducting circuits for quantum information: An outlook. *Science*, 339:1169–1174, 2013. doi: 10.1126/science.1231930.

[14] Theodore Walter et al. Rapid high-fidelity single-shot dispersive readout of superconducting qubits. *Physical Review Applied*, 7(5):054020, 2017.

[15] Austin G. Fowler, Matteo Mariantoni, John M. Martinis, and Andrew N. Cleland. Surface codes: Towards practical large-scale quantum computation. *Physical Review A*, 86(3):032324, 2012.

[16] Leandro Stefanazzi et al. The QICK (Quantum Instrumentation Control Kit): Readout and control for qubits and detectors. *Review of Scientific Instruments*, 93(4):044709, 2022. arXiv:2110.00557.

[17] Philip Krantz, Morten Kjaergaard, Fei Yan, Terry P. Orlando, Simon Gustavsson, and William D. Oliver. A quantum engineer's guide to superconducting qubits. *Applied Physics Reviews*, 6(2):021318, 2019. doi: 10.1063/1.5089550. Comprehensive review of superconducting qubit instrumentation and control.

[18] John Preskill. Quantum computing in the NISQ era and beyond. *Quantum*, 2:79, 2018. arXiv:1801.00862.

[19] John Preskill. Beyond NISQ: The megaquop machine. Keynote address at Q2B 2024 Conference, December 2024. arXiv:2502.17368.

[20] Jens Eisert and John Preskill. The quantum computing bubble. arXiv:2411.03528, November 2025.

[21] M. A. Rol et al. Restless tuneup of high-fidelity qubit gates. *Physical Review Applied*, 7:041001, 2017. Gate calibration and drift characterization.

[22] Clemens Müller, Jared H. Cole, and Jürgen Lisenfeld. Towards understanding two-level-systems in amorphous solids: Insights from quantum circuits. *Reports on Progress in Physics*, 82:124501, 2019. TLS-induced coherence fluctuations and calibration drift.

[23] Sarah Sheldon et al. Characterizing errors on qubit operations via iterative randomized benchmarking. *Physical Review A*, 93:012301, 2016. Crosstalk and correlated error characterization.

[24] Jay M. Gambetta, Jerry M. Chow, and Matthias Steffen. Building logical qubits in a superconducting quantum computing system. *npj Quantum Information*, 3:2, 2017. Randomized benchmarking methodology for fidelity measurement.

[25] Google Quantum AI. Quantum error correction below the surface code threshold. *Nature*, December 2024. Willow processor: 105 qubits, 99.97% 1Q fidelity, 99.88% 2Q fidelity, T1 $\sim$100 μs.

[26] Quantinuum. Quantinuum extends its significant lead in quantum computing, achieving historic milestones for hardware fidelity and quantum volume. Quantinuum Blog, April 2024. H1: 99.914% 2Q gate fidelity, first "three nines" commercial system.

[27] Julian Kelly et al. State preservation by repetitive error detection in a superconducting quantum circuit. *Nature*, 519:66–69, 2015. Measurement crosstalk and readout-induced dephasing.

[28] David C. McKay et al. Universal gate for fixed-frequency qubits via a tunable bus. *Physical Review Applied*, 6:064007, 2016. ZZ crosstalk in fixed-frequency transmon architectures.

[29] IBM Quantum. IBM launches its most advanced quantum computers. IBM Newsroom, November 2024. Heron R2: 156 qubits, 5000 two-qubit gate circuits, tunable couplers.

[30] Youngseok Kim et al. Evidence for the utility of quantum computing before fault tolerance. *Nature*, 618:500–505, 2023. IBM 127-qubit utility demonstration.

[31] F. Bahrami, Nathalie P. de Leon, Andrew Houck, Robert J. Cava, et al. High-coherence transmon qubits on tantalum-on-silicon. *Nature*, November 2025. Princeton T1 $>$ 1.6 ms result.

[32] SQMS Center. Researchers achieve leading performance in transmon qubits. Fermilab News, May 2024. SQMS-Rigetti 9-qubit processor.

[33] Tanay Roy, Taeyoon Kim, Yubing Lu, Alexander Romanenko, and Anna Grassellino. Qudit-based quantum computing with SRF cavities at Fermilab. FERMILAB-CONF-24-0026-SQMS, 2024.

[34] SQMS Center. Fermilab's SQMS Center funded with $125 million. Fermilab News, November 2025. SQMS 2.0: 100+ qudit QPU roadmap, 10 ms coherence goal.

[35] Pengfei Wang et al. Single ion qubit with estimated coherence time exceeding one hour. *Nature Communications*, 12:233, 2021. Trapped ion T2 $\sim$5500 s.

[36] Christopher Monroe et al. Progress in trapped-ion quantum simulation. *Annual Review of Condensed Matter Physics*, 16:145–172, 2025.

[37] Quantinuum. Quantinuum's H-Series hits 56 physical qubits that are all-to-all connected. Quantinuum Blog, June 2024. H2-1: 56 qubits, maintained three-nines fidelity at scale.

[38] Harry Levine, Dolev Bluvstein, Marcin Kalinowski, et al. A tweezer array with 6,100 highly coherent atomic qubits. *Nature*, September 2025. Caltech/Harvard: T1 = 119 s, T2 = 12.6 s with DD, 99.97% single-qubit Clifford fidelity.

[39] Simon J. Evered, Dolev Bluvstein, et al. High-fidelity parallel entangling gates on a neutral-atom quantum computer. *Nature*, 622:268–272, 2023. Harvard/QuEra: 99.5% 2Q gate fidelity, logical qubit demonstrations.

[40] Nico Samkharadze, Amir Sammak, Giordano Scappucci, et al. Industry-compatible silicon spin-qubit unit cells exceeding 99% fidelity. *Nature*, 646:81–87, September 2025. Diraq/imec: T1 = 9.5 s, T2_Hahn = 1.9 ms, >99% 1Q/2Q fidelity, 300mm foundry.

[41] Bo Sun, Thomas Brecht, Brian H. Fong, et al. Full-permutation dynamical decoupling in triple-quantum-dot spin qubits. *PRX Quantum*, 5:020356, 2024. HRL: T2* = 2 μs $\rightarrow$ T2 = 720 μs with DD, exchange pulse error 2.8×10^{-5}.

[42] S. D. Ha, E. Acuna, K. Raach, et al. Two-dimensional Si spin qubit arrays with multilevel interconnects. *PRX Quantum*, 6:030327, 2025. HRL SLEDGE: >99.9% 1Q gate fidelity, 2D scalable arrays, spinQICK compatible.

[43] PsiQuantum. Omega: A manufacturable chipset for photonic quantum computing. *Nature*, February 2025. 99.98% SPAM fidelity, 99.5% two-photon interference, 99.72% chip-to-chip interconnect.

[44] Xanadu. On-chip generation of GKP states for fault-tolerant quantum computing. *Nature*, June 2025. First on-chip GKP qubit generation, room-temperature photonic platform.

[45] J. Van Damme et al. Advanced CMOS manufacturing of superconducting qubits on 300 mm wafers. *Nature*, September 2024. Imec: First demonstration of CMOS foundry-compatible superconducting transmon qubits with T1 > 100 μs, 98.25% yield across 400 qubits on 300mm wafers using optical lithography and reactive-ion etching.

[46] The Quantum Insider. What is the price of a quantum computer in 2025? The Quantum Insider, December 2025. Comprehensive cost analysis: system prices, cloud access, TCO by modality.

[47] Charles Day, M. Wozniak, et al. Helium-3 supply and operational costs. Physics Today (2011) and Energies (2020), 2020. ^{3}He pricing: \$200/L historical $\rightarrow$ \$2,000–\$3,000/L current; supply constraints from tritium decay.

[48] Simon Haykin. *Adaptive Filter Theory*. Pearson, Upper Saddle River, NJ, 5th edition, 2014.

[49] Bernard Widrow and Samuel D. Stearns. *Adaptive Signal Processing*. Prentice-Hall, Englewood Cliffs, NJ, 1985.