23. Quantum Hardware, Errors, and the Future

What you already know

You have never trusted hardware to be perfect. You use ECC memory because a cosmic ray can flip a RAM bit; you run RAID because disks die; TCP carries checksums because packets corrupt. Classical computing is built on a quiet foundation of error correction — you just rarely think about it, because classical errors are *rare*: a good memory cell flips maybe once in 10^{17} operations. The whole discipline is familiar: add redundancy, detect the error with a parity-like check, fix it.

Quantum hardware forces that discipline into the foreground, for two reasons you can already anticipate. First, the errors are not rare — today's best gates fail about once in a thousand operations, a rate that would make a classical computer unusable. Second, and stranger, the classical fix does not transfer: the no-cloning theorem of Chapter 20 forbids copying a qubit, so you cannot make the three redundant copies a classical repetition code relies on, and measuring a qubit to check it collapses it (Chapter 8). This chapter is about how physicists build qubits at all, why those qubits are so fragile, and the genuinely clever way — quantum error correction — that gets around both obstacles. It is the most fast-moving chapter in the book, so it leans hardest on the honest boundary.

What breaks a qubit

The enemy is *decoherence* (Chapter 8): a qubit's fragile superposition leaks into its environment and the quantum information is lost. Physicists split the damage into two timescales (Figure 23.1). T_1 is the *relaxation* time — how long until an excited $|1\rangle$ decays to $|0\rangle$, losing energy to the surroundings, exactly the kind of decay met in Chapter 11. T_2 is the *dephasing* time — how long the delicate *phase* relationship of a superposition (the very thing Chapter 20 showed distinguishes a qubit from a coin) survives before it randomizes. T_2 is usually the shorter and the crueller, because phase is where the quantum advantage lives. Every gate must run, and every algorithm must finish, before these clocks run out.

WORKED EXAMPLE · Why a deep circuit fails without help

Suppose each gate succeeds with probability $1 - p$ where $p = 10^{-3}$ (a good gate today). A circuit of D gates in sequence succeeds only if *all* of them do, roughly with probability $(1-p)^D \approx e^{-pD}$:

$$D = 100:\ e^{-0.1} \approx 0.90; \qquad D = 1000:\ e^{-1} \approx 0.37; \qquad D = 10^6:\ e^{-1000} \approx 0.$$

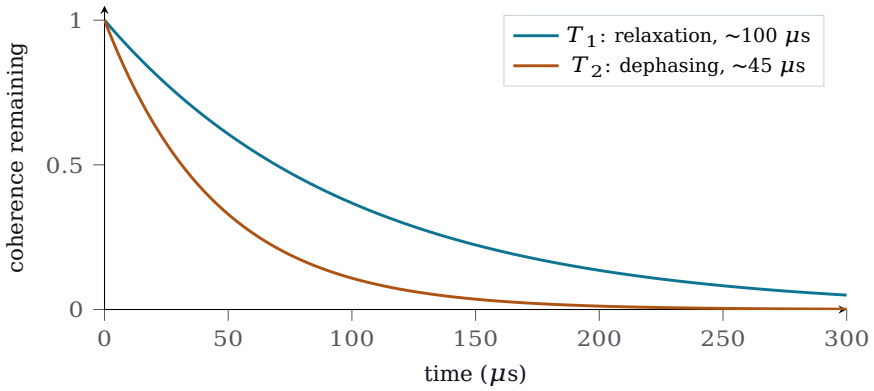

Figure 23.1 – The two clocks a qubit races. T_1 (teal) is how long an excited state survives before relaxing; T_2 (burnt) is how long the superposition's phase survives before randomizing. Both are exponential decays like Chapter 11's, and T_2 is typically the shorter — phase, the source of the quantum advantage, is the first thing lost. Representative superconducting-qubit values are shown; every gate must finish well inside them.

A hundred gates is fine; a thousand is a coin-flip; and Shor's algorithm on a real key (Chapter 22), which needs on the order of 10^9 gates, has essentially zero chance of finishing correctly on raw hardware. This single calculation is why quantum computing cannot work on physical qubits alone, and why the rest of the chapter exists.

The hardware zoo

There is no settled answer to what a qubit should be *made of*; several physical platforms are being pursued in parallel, each with a real strength and a real weakness (Figure 23.2). This is a healthy sign of an engineering field still finding its transistor — recall that early classical computing tried relays, tubes, and cores before silicon won.

Quantum error correction

Here is the idea that makes the whole enterprise possible, and it threads the needle between the two obstacles of the opening. You cannot copy a qubit, and you cannot look at it — but you *can* spread one *logical* qubit's information across many *physical* qubits, entangled together, and then measure only carefully chosen *joint* properties — parities of neighbouring qubits, called *syndromes* — that reveal *whether an error occurred and where*, without ever revealing (and so without collapsing) the encoded data itself. It is the classical parity check, bent around the no-cloning and no-peeking rules of quantum mechanics (Figure 23.3).

The remarkable guarantee is the *threshold theorem*: provided each physical component is below a certain error rate — roughly 1% for the leading *surface code* — you can make the logical error rate as small as you like by using more physical qubits per logical one. Quality above the threshold buys unlimited reliability through quantity. The catch is that "quantity": the overhead is enormous, often around a thousand physical qubits to protect a single logical one well enough for a serious algorithm. This is the true reason Chapter 22's Shor estimate ran to *millions* of physical qubits — a few thousand logical qubits, each paid for a thousand times over.

Platform	A qubit is…	Strength	Challenge
superconducting	an oscillation in a chilled circuit (transmon)	fast gates, chip fabrication	short coherence; needs milli-kelvin fridge
trapped ions	two levels of a held ion	long coherence, high fidelity	slower gates; scaling the trap
neutral atoms	atoms in optical tweezers	many qubits, flexible geometry	still maturing
photonic	states of light	room temperature, networks naturally	deterministic two-qubit gates are hard
topological	a nonlocal (anyonic) state	error-resistant in principle	largely unrealized so far

Figure 23.2 – The hardware zoo. No platform has clearly won. Superconducting and trapped-ion machines are the most advanced today; neutral atoms are rising fast; photonic and topological approaches make different bets. Each row trades coherence, speed, fidelity, and manufacturability differently.

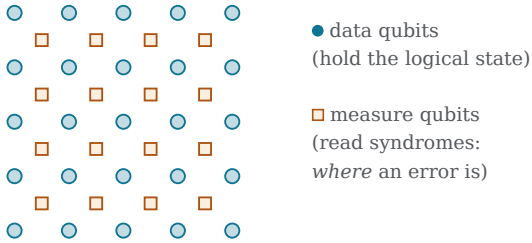

Figure 23.3 – A surface-code patch. One logical *qubit is stored across a grid of* physical *data qubits (teal); interleaved measure qubits (amber) repeatedly read parity "syndromes" that locate errors without measuring — or collapsing — the encoded data. The* threshold theorem *guarantees that if each physical gate is better than about* 1%*, enough such qubits give an arbitrarily reliable logical qubit; the price is steep, often around a thousand physical qubits per logical one.*

Where we are, and the future

Today's machines are what John Preskill named *NISQ* — Noisy Intermediate-Scale Quantum: tens to a few hundred physical qubits, too noisy and too few for full error correction. They can run shallow circuits, and a handful of demonstrations have performed some narrow, deliberately contrived sampling task faster than the best classical simulation — a real milestone, but not yet a *useful* one, and several such claims were later matched by improved classical methods. The honest present is a field of rapid, genuine engineering progress that has not yet crossed from demonstration to routine utility.

The path forward is not mysterious, only hard: better coherence, better gates, more qubits, and above all error correction at scale, until *fault-tolerant* machines with thousands of logical qubits exist. When that arrives — a decade away, or more, with a forecasting record no one should be proud of — the first genuinely useful payoff is most likely the quantum *simulation* of

Chapter 22, precisely because it plays to the machine's nature. And with that horizon set, the book turns in Chapter 24 to the one quantum computer entirely within reach today: a simulator, built in C# on the quantum context of Chapter 16.

HONEST BOUNDARY

Everything in this chapter dates fastest of anything in the book. The specific numbers — coherence times, gate fidelities, qubit counts — improve yearly, so treat them as of-their-moment illustrations, not fixed facts; by the time you read this the records will have moved. The error-correction overhead (~1000:1) is an estimate that varies widely with the algorithm, the code, and the noise model. "Quantum advantage" is a genuinely contested phrase: demonstrations have been real, narrow, and repeatedly challenged by better classical algorithms, and none so far solves a problem anyone needed solved. Topological qubits, potentially the most robust, remain largely a promise. What is *not* in doubt is the physics: decoherence is real (Chapter 8), the threshold theorem is proven, and the overhead is a mathematical consequence, not a temporary limitation. The engineering forecast is uncertain; the underlying science is not. Anyone offering confident dates is guessing — this book will not.