Quantum noise and hardware limitations

Useful computation depends on reliability, connectivity and total resources, not just a qubit count.

16 min read Dr. Charith Jayasekara

By the end, you'll be able to

- Distinguish finite-shot fluctuations from physical noise.
- Explain physical versus logical qubits without assuming a fixed overhead.

Noise is more than randomness

Even an ideal $|+\rangle$ preparation gives random Z outcomes. That is measurement statistics. Physical noise changes the preparation, evolution or readout relative to the ideal model.

Possible effects include decoherence, imperfect gates and readout errors. A single number rarely characterises all of them. A noise model must state which processes it approximates and which it excludes.

Connectivity changes the circuit

If a device only supports interactions between neighbouring qubits, an abstract two-qubit gate may require routing operations. SWAP gates can increase depth and the number of imperfect operations. Compilation quality therefore matters alongside the original algorithm.

Physical and logical qubits

A physical qubit is a hardware system. A logical qubit encodes information across physical resources using an error-correcting scheme. The overhead depends on the code, physical error rates, target logical reliability and the computation. There is no universal "one logical qubit equals 1,000 physical qubits" conversion.

A simple evidence exercise

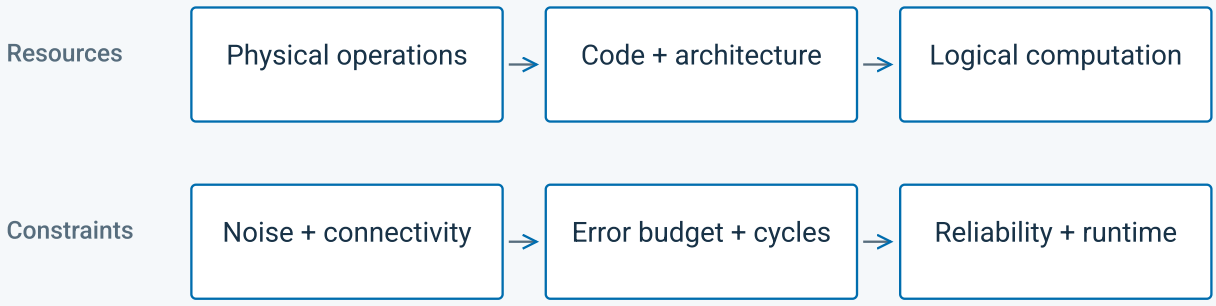
Open BB84 in the experiment studio. With no adversary, increase the channel bit-flip probability. Then set it to zero and enable intercept–resend. Both can introduce disagreements. The toy model demonstrates why an observed error rate alone need not identify its cause.

This BB84 channel is not a general decoherence simulator. The foundation statevector tools remain ideal. A realistic noise study needs an explicit channel model, independent checks and careful interpretation.

Think in terms of an error budget

WORKED DIAGRAM · F22

Reliability depends on more than a physical-qubit total



The same physical count can support different logical computations depending on the code, error rates, connectivity, control speed and required failure probability. No fixed multiplication gives a universal resource estimate.

A useful computation needs its total failure probability below a chosen tolerance. That budget includes more than the number of qubits. Gate errors, idle errors, measurement errors, connectivity, decoder performance and the computation's duration all matter. Error correction redistributes information across a larger physical system so that suitable error syndromes can be detected and corrected without directly reading the protected logical state.

Finite-shot variation is different. Repeated ideal measurements can be random even when every gate is perfect. Physical noise changes the experiment relative to that ideal prediction. Increasing the number of shots reduces uncertainty about a noisy distribution; it does not necessarily move that distribution toward the ideal one. More data cannot by itself remove a systematic calibration error.

Worked example: repeated exposure to error

As a deliberately simplified classical reliability model, suppose each of one thousand operations independently fails with probability one in one thousand, and any failure ruins the computation. The probability of no failures is approximately the product of one thousand copies of 0.999, about 0.368. A seemingly small per-operation error can therefore produce a poor overall success rate in a long calculation.

This multiplication is not a full quantum noise calculation. Errors can cancel, propagate, correlate or be corrected, and the effect of an error depends on the algorithm and measurement. The example teaches why a per-gate number cannot be interpreted without circuit size and error handling. A resource estimate must translate physical error assumptions into a logical failure budget for the full computation.

Updated RSA-2048 resource examples

There is no fixed physical-to-logical qubit conversion. Gidney's 2025 study estimates RSA-2048 factoring with fewer than one million noisy physical qubits in less than a week, assuming a nearest-neighbour square grid, a 0.1-percent gate error rate, one-microsecond surface-code cycles and a ten-microsecond control reaction time. This is a conditional resource estimate. Gidney, 2025.

Webster and colleagues' Pinnacle architecture study, revised in May 2026, estimates fewer than one hundred thousand physical qubits using quantum LDPC codes under stated physical-error and timing assumptions. Its architecture differs from the surface-code design, so the headline qubit counts are not interchangeable hardware specifications. Webster et al., 2026.

Neither paper reports a completed RSA-2048 factoring experiment. Compare the full assumptions: connectivity, code operations, control and decoding, runtime, resource allocation and target success probability. A lower qubit count may involve a different architecture or a different space–time trade-off. These estimates update the reading edition's earlier simplified example; they are not dates for the arrival of a working attack machine.

Logical protection is an engineering system

A logical qubit is more than a group of physical qubits with a ratio attached. It requires repeated syndrome extraction, decoding and corrective control or frame tracking. Logical gates can need additional ancillas, routing and specialised resource states. The resources used while storing a state can differ from those needed while performing a difficult non-Clifford operation.

Connectivity also affects code feasibility. An architecture that assumes certain long-range interactions cannot be compared with a nearest-neighbour layout by counting qubits alone. Physical cycle times and classical reaction times constrain scheduling. A scientifically useful comparison states the whole operational model so readers can see which improvements are algorithmic, code-related or hardware-dependent.

Use a limited noise model honestly

The BB84 lab below offers independent channel bit flips and an intercept–resend option. It illustrates how different causes can produce errors in a protocol record. It is not a density-matrix simulation of energy relaxation or dephasing, and it does not implement error correction. Use it to practise identifying the scope of a model rather than treating one slider as a universal physical noise strength.

With no adversary, predict the sifted error rate for a chosen channel-flip probability, then run. Next remove channel flips and enable intercept–resend. Compare the resulting uncertainty intervals. Similar observed error rates can arise from different mechanisms; one summary statistic does not identify the cause.

L31 · INTERACTIVE LAB

Runs on your device

BB84 & the evidence of errors

A toy prepare-and-measure protocol with random Z/X bases, optional intercept–resend and independent channel bit flips. It produces error statistics, not a secure key.

MODEL · Prepare & measure, seeded sampling ORDER · q0 is the leftmost bit

Initial setup · reference state

Alice (random bits) → Channel (bit flips) → Bob (random bases)

Matching bases occur with probability 1/2. Ideal expected sifted QBER: 0.0%.

All sifted bits are used for this illustration. No secret key is produced. Authentication, reconciliation, privacy amplification and a finite-key security proof are outside the model. Noise and an adversary can both create errors.

Prediction to try: Predict the sifted fraction and error rate for these settings. What uncertainty should finite sampling introduce?

Challenge: Find an attack case and a noise-only case with similar QBER. Explain why that statistic alone cannot identify the cause.

Open the live experiment studio

For a security reader, the resource and uncertainty questions lead to different decisions. A factoring estimate concerns the feasibility of a future attack under assumptions. A protocol error estimate concerns evidence collected in a particular experiment. Keep those arguments separate when deciding what a result says about a deployed system.

Practice: compare two resource claims

One design uses fewer physical qubits but takes longer and assumes more demanding connectivity. Another uses more qubits with local interactions. What additional information would you request before declaring the first design better?

Check your reasoning

Ask for the same target problem, success probability, physical error model, timing, connectivity feasibility, decoder and control costs, and total runtime. Include ancillas and resource-state factories in the count. “Better” depends on the engineering constraints and objective, such as minimum time, minimum hardware footprint or a space–time budget. A headline qubit count alone cannot settle the comparison.

Reference: Preskill — Quantum computing in the NISQ era and beyond.

Check your understanding

PAUSE & THINK

Can two RSA resource estimates be compared using only their physical-qubit headline?

- [ ] Yes
- [ ] No; architecture, runtime, error assumptions and success target also matter
- [ ] Only if both use the word logical

Answer: No; architecture, runtime, error assumptions and success target also matter. There is no universal physical-to-logical ratio, and different designs have different space–time and engineering assumptions.