Classical computing versus quantum computing

Start with a familiar bit, then see what changes when information becomes quantum.

17 min read Dr. Charith Jayasekara

By the end, you'll be able to

- Distinguish a classical bit from a pure qubit.
- Explain why quantum computing is not a faster replacement for every program.
- Identify preparation, transformation and measurement in a simple workflow.

A classical bit has one of two values: **0 or 1**. A pure qubit is described by two complex amplitudes. Those amplitudes let us predict measurement outcomes and, through their relative phase, how later operations will interfere.

What changes?

Write a qubit as

$$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle, \qquad |\alpha|^2 + |\beta|^2 = 1.$$

You do not need to know complex numbers yet. For now, read this as: "a state is built from two amplitudes, and the probabilities must add to one." Measuring in the computational, or Z, basis gives **one bit**: 0 with probability $|\alpha|^2$, or 1 with probability $|\beta|^2$.

A qubit does not give you two independently readable classical values. The state is richer than the outcome of one measurement.

A first experiment

The sphere below represents a single pure qubit. Its north pole is $|0\rangle$ and south pole is $|1\rangle$. Start at the north pole, apply H, then run repeated measurements. Every shot prepares the same state again.

A different workflow

1. **Prepare** the input state.
2. **Transform** it using gates.
3. **Measure** to obtain classical outcomes.
4. **Analyse** repeated outcomes and compare them with a prediction.

Classical computers still handle the surrounding application: data, user interfaces, networking and analysis. Quantum methods may help with particular structured problems. A large search space by itself does not prove that a quantum approach will be practically faster.

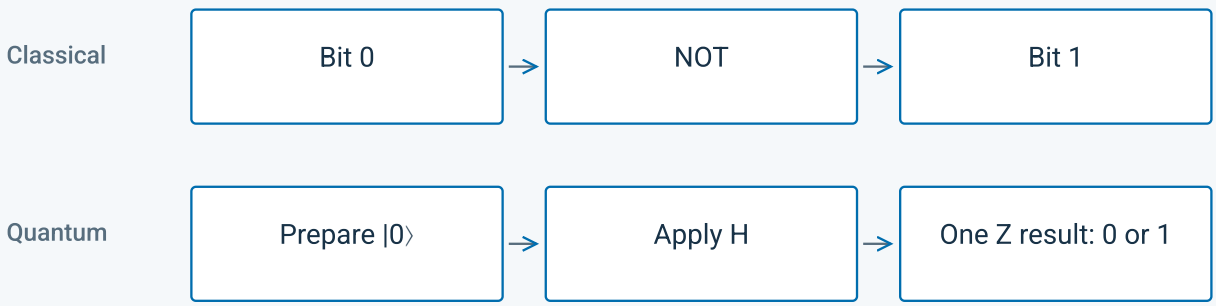
Connect it to the source

This lesson develops sections 2–4 of the author's introduction. Continue to quantum information or take the gentle route to qubits.

Read the diagram as an experiment

WORKED DIAGRAM · F01

Preparation, transformation and readout



A gate changes a quantum state; a Z measurement returns one classical bit. The lower lane describes independent repetitions, not two simultaneous readable answers.

Both rows end in a classical record. The distinction is in what can happen before that record is made. In the classical row, a NOT operation exchanges two definite labels. In the quantum row, H creates a coherent state from a known input, and a later measurement returns one of the labels. The drawing does not mean that a detector reads both amplitudes. It shows the stages that must be specified to make a prediction.

Imagine writing a test for a random-number function. A classical implementation might use a pseudorandom generator with an internal seed. An ideal quantum implementation might prepare a zero state and apply H before measurement. Both functions return ordinary bits to the calling program. Their internal explanations differ, and neither output sequence alone proves that its implementation is correct. In this book, the browser uses a classical simulator with a seed; it is a learning model, not a physical quantum random-number source.

Worked example: one H, then two

Start with amplitudes one for 0 and zero for 1. The H gate replaces them with equal positive amplitudes, each approximately 0.707107. To turn either amplitude into a probability, square its magnitude. Each square is one half. This is why freshly preparing this circuit many times produces an approximately even split. It is not a claim that every pair of shots contains exactly one zero and one one.

Now apply H a second time, without measuring between the gates. The new zero amplitude is the sum of the two old amplitudes divided by the square root of two. The new one amplitude is their difference divided by the same factor. The sum becomes one and the difference becomes zero. The output is therefore certainly zero in the ideal model. Adding the second gate has removed the random output by making amplitudes interfere.

This calculation gives an early reason to track amplitudes instead of treating a qubit as a hidden coin. If you replaced the intermediate state with a classical random choice between the two basis states, another H would still produce a half-and-half distribution. The coherent preparation retains a relationship between components that the random choice does not. Later lessons express that relationship using relative phase.

A precise first lab

The controls below open at polar angle zero: the north pole, representing the zero state. Reset restores that preparation. Before pressing a result button, decide which experiment you want to test. For the first trial, apply H once and write down your predicted amplitudes and probabilities in the prediction box. Then reveal the exact model. The sphere, histogram and statevector appear together so you can connect three representations of the same state.

For the second trial, apply H again to the current state. Changing the experiment clears the earlier prediction and hides its result. Record the new prediction, reveal it, and compare with the first trial. You should explain the change through addition and cancellation of amplitudes. Saying that the second H “measures the qubit” would give the wrong mechanism: H is a gate, and measurement happens when the experiment reads out the state.

Finally, reset and try X before measurement. X moves the prepared zero state to the one state, so repeated ideal measurements should all return one. Compare this deterministic change with H. If a run shows rounded probabilities of zero or one, remember that these values describe the chosen ideal model. Real devices need calibration and repeated evidence before such a reliability claim can be made.

L01 · INTERACTIVE LAB

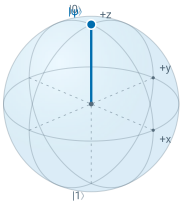
Runs on your device

Qubit explorer

Prepare a pure qubit, apply gates, then compare amplitudes, phases and Z-basis probabilities. Reset restores the initial preparation shown by the lesson’s controls.

MODEL · Ideal pure-state vector, complex double precision ORDER · q0 is the leftmost bit

Initial setup · reference state



Exact probability



Statevector before readout · ideal simulation

Basis	Amplitude a + bi	Magnitude	Phase (rad)	Probability
|0⟩	`1 + 0i`	1.000000	0.000000	100.0%
|1⟩	`0 + 0i`	0.000000	undefined	0.0%

Signs and complex phases are part of the state. Phase is undefined for a zero amplitude. A shared global phase changes no probabilities; relative phases can affect a later operation. Values are rounded to six decimal places.

Prediction to try: Predict the amplitudes and Z-basis probabilities for the preparation and gates you have chosen. If starting from $|0\rangle$, what will H do?

Challenge: Prepare $\theta = 120°$, $\varphi = 90°$. Predict both complex amplitudes. Then change φ without changing θ.

Open the live experiment studio

Decide when a quantum approach is relevant

A useful application needs a problem definition, an encoding, a transformation and a way to extract the desired answer. The fact that a statevector has many components does not let an application print those components in a single measurement. An algorithm must arrange interference so that a useful property influences outcomes that can be collected with reasonable effort. That requirement is the bridge between the physics and the software design.

For a software practitioner, the surrounding workflow remains familiar: validate inputs, document assumptions, choose a backend, collect results and check uncertainty. For a mathematics or research reader, the next question is which allowed transformations preserve the state rules. For a security practitioner, a further question is what resources an attack or protocol assumes. These perspectives can share the same foundation without assuming every workload benefits from a quantum device.

Try it on paper

Explain what changes if the circuit measures immediately after its first H, then applies another H to the measured qubit. Compare that experiment with the uninterrupted two-H circuit. State both the intermediate information available to an observer and the final distribution.

<details>
<summary>Check your reasoning</summary>

The intermediate measurement produces a classical zero or one and leaves the qubit in that basis state. H maps either basis input to a state with equal Z probabilities. The final result is therefore half zero and half one. Without the intermediate measurement, the two H gates compose to the identity and return the original zero state. Measurement changes the experiment; it cannot be inserted as a harmless debugging print.
</details>

Reference: IBM Quantum — Basics of quantum information.

Check your understanding

PAUSE & THINK

What changes if you measure between two H gates on a zero input?

- [ ] The second H still guarantees zero
- [ ] The intermediate measurement removes the coherence needed for deterministic return
- [ ] The measurement reads both amplitudes

Answer: The intermediate measurement removes the coherence needed for deterministic return. The measurement leaves a basis state. A later H gives equal Z probabilities, whereas two uninterrupted H gates restore zero.

PAUSE & THINK

What do you get from one Z-basis measurement of a qubit?

- [ ] Both amplitudes, read exactly
- [ ] One classical outcome: 0 or 1
- [ ] Every possible solution to a problem

Answer: One classical outcome: 0 or 1. A measurement gives one outcome. Repeated, freshly prepared states let us estimate a distribution; other measurement bases are needed to investigate phase.