# First complete quantum algorithm

Use an oracle, phase kickback and interference to answer a precisely defined question.

17 min read Dr. Charith Jayasekara

**By the end, you'll be able to**

- State Deutsch's input, promise and output.
- Derive its one-query result and connect it to Deutsch–Jozsa.
- Separate an ideal query result from a practical performance claim.

## The problem contract

Given oracle access to $f : \{0,1\} \to \{0,1\}$, decide whether f is **constant** ($f(0) = f(1)$) or **balanced** ($f(0) \neq f(1)$). There are four possible functions. A deterministic classical solution requires two evaluations in the worst case.

## Prepare and query

Start with $|0\rangle|1\rangle$ and apply H to both qubits. The state becomes $|+\rangle|-\rangle$. Apply $U_f$ once. Phase kickback produces

$$\frac{(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle}{\sqrt{2}} \otimes |-\rangle.$$

Apply H to the first qubit. If f is constant, the input becomes 0 up to global phase. If f is balanced, it becomes 1 up to global phase. Measuring the input therefore answers the question with one ideal query.

## Extend to more input bits

Deutsch–Jozsa uses an n-bit input under the promise that f is constant or balanced over all $2^n$ inputs. Its all-zero output has probability one for a constant function and zero for a balanced function in the ideal model.

The lab shows two- or three-bit input registers after phase kickback and factors out the unchanged ancilla. The balanced preset uses the last input bit. It is one valid balanced example, not an enumeration of every possible function.

## State the conclusion carefully

This is a query-model separation for a promised problem. It is not proof that a physical implementation outperforms a classical computer after oracle synthesis, noise, readout and total runtime are counted. Randomised classical comparisons also differ from the deterministic worst-case comparison above.

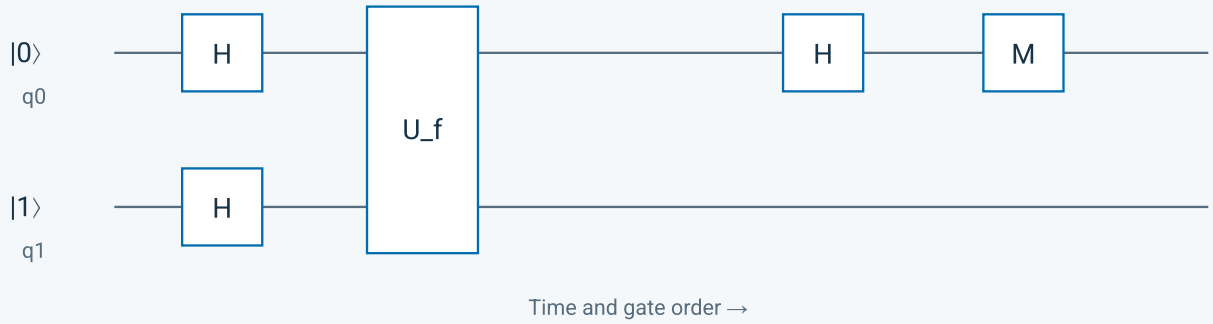
## Your next investigation

Complete P7 and P8 in the practice studio, then compare this phase-kickback pattern with Grover amplification. Keep the input/output contract, assumptions and excluded costs in your report.

## Read the complete circuit

WORKED DIAGRAM · F26

### Deutsch: one oracle call, then interference




The upper wire is the input and the lower is the ancilla. Prepare |0⟩|1⟩, apply H to both, call Uf once, then apply H and measure the input. Constant gives 0; balanced gives 1 in the ideal one-bit case.

The top wire is the input register and the bottom wire is the oracle target. H prepares plus on the input and minus on the target. The joint oracle box represents one invocation of the reversible function interface, not separate calls on the two wires. The last H acts only on the input, which is then measured to classify the promised function.

The target does not contain a readable copy of both function values. It is an eigenstate that enables phase kickback. Its state factors out after the query, while the input acquires a relative phase encoding whether the two function values agree. The final H turns that one-bit property into a measurement label.

## Work all four cases

For constant zero, the input after the query is plus and the final result is zero. For constant one, the input is negative plus and the final result is still zero. For f(x) equal to x, the input is minus and the result is one. For the complemented input function, the input is negative minus and the result is again one. The common minus signs do not change the classification.

You can express the same result algebraically. The final zero amplitude is the sum of the two phase signs divided by two, while the final one amplitude is their difference divided by two. Equal signs reinforce at zero; opposite signs reinforce at one. The algorithm asks one precise question and chooses interference so that its answer becomes deterministic in the ideal model.

## Compare the classical task fairly

A deterministic classical procedure evaluating the function at one input cannot always decide whether the unseen second value agrees. It needs two evaluations in the worst case for an exact answer. Deutsch's ideal quantum procedure uses one coherent oracle invocation. This is a query-complexity separation for a specified task, not a claim that the tiny circuit is faster than a classical computer in wall-clock time.

Allowing classical error changes the comparison. Guessing with one evaluation can have nonzero success under a chosen distribution, but it does not provide the same exact worst-case guarantee. Conversely, a noisy quantum implementation no longer achieves certainty without qualification. Match the success criterion, input promise and oracle access when stating the advantage.

## Generalise to Deutsch–Jozsa

With n input bits, the uniform input has equal amplitude on every bitstring. After phase kickback and final H gates, the all-zero amplitude is the average of the function's phase signs. A constant function makes every sign equal, so the all-zero probability is one. A balanced function has equal positive and negative signs, so that amplitude cancels to zero.

The promised algorithm separates these two classes. A non-promised function can produce an intermediate all-zero probability and does not fit the classification guarantee. A particular balanced function may produce one nonzero bitstring deterministically, but the theorem only needs the absence of all-zero. The explorer's balanced last-bit example is a transparent special case rather than a complete catalogue of balanced functions.

## What makes this a complete algorithm?

It has a defined input-access model, a promise, a preparation, a sequence of operations, a measurement rule and an interpretation of the result. It also has a correctness argument and a stated comparison resource. Those are the pieces to look for in later algorithms. A circuit picture alone is not enough if the oracle's cost or the output's meaning remains unspecified.

For a learning progression, explain the one-bit case without notation first: the two function outputs either agree or disagree; the oracle turns that relation into a phase pattern; H turns the pattern into one measured bit. Then use the amplitude equations to verify the story. An intuitive explanation and a calculation should support the same claim rather than tell different versions of the algorithm.

## Run the one-bit and larger cases

Choose one input qubit in the explorer. Test constant zero, constant one and balanced, recording a prediction for each before revealing. Check the amplitudes as well as the output label, and explain any global sign difference between the two constant cases. The minus target is factored out of the displayed register.

L05 · INTERACTIVE LAB

Runs on your device

### Deutsch–Jozsa explorer

A constant-or-balanced Boolean oracle acts by phase kickback, followed by H on each input qubit. The unchanged |−⟩ ancilla is factored out. One input qubit gives Deutsch's original case.

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 |
|---|---|---|---|---|
| \|00⟩ | `0 + 0i` | 0.000000 | undefined | 0.0% |
| \|01⟩ | `1 + 0i` | 1.000000 | 0.000000 | 100.0% |
| \|10⟩ | `0 + 0i` | 0.000000 | undefined | 0.0% |
| \|11⟩ | `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 whether the all-zero input-register outcome can occur for the selected promised oracle.

Challenge: Constant-zero and constant-one differ by a global sign before the final H gates. Explain why their measured probabilities agree.

Open the live experiment studio

Next select two or three input qubits. Predict the all-zero probability under each promised oracle. State what feature of the proof survives the larger register: it is cancellation of an equal number of positive and negative contributions, not the ability to read every function value. The experiment uses small ideal simulations and does not compile an arbitrary oracle into hardware gates.

## Practice: explain the algorithm to a beginner

Write a four-sentence account containing the question, preparation, interference mechanism and output meaning. Then identify one important implementation cost that a query-count comparison leaves out.

**Check your reasoning**

The question is whether the two Boolean outputs agree. Prepare plus on the input and minus on the oracle target. One oracle invocation encodes agreement or disagreement as equal or opposite input phases, and H converts those patterns into zero or one. Measuring zero means constant and one means balanced in the ideal one-bit task. Building the coherent reversible oracle, including workspace and cleanup, is an implementation cost outside the abstract query count.

Reference: IBM Quantum — Fundamentals of quantum algorithms.

## Check your understanding

PAUSE & THINK

### For a promised balanced Deutsch–Jozsa function, what is guaranteed?

- [ ] Every nonzero outcome is equally likely
- [ ] The all-zero input-register outcome has probability zero
- [ ] The whole truth table is returned

**Answer:** The all-zero input-register outcome has probability zero. Equal positive and negative phase contributions cancel the all-zero amplitude. Other output probabilities depend on the function.

PAUSE & THINK

### What is essential to the Deutsch–Jozsa conclusion?

- [ ] That f is promised constant or balanced
- [ ] That every function is balanced
- [ ] That a measurement reveals every f(x)

**Answer:** That f is promised constant or balanced. The promise defines the decision problem. The algorithm extracts a global property under that promise; it does not return the complete function table.