Quantum Gates

Common Quantum Circuits in Quantum Computing

Quantum circuits are the foundation of quantum algorithms and computations, relying on specific combinations of quantum gates to achieve tasks like creating superpositions, entangling qubits, and measuring results. Here are some of the most commonly used quantum circuits in quantum computing, along with explanations of their purposes and functions.

1. Superposition Circuit (Hadamard Circuit)

A superposition circuit puts qubits into a superposition state, where each qubit exists simultaneously in both $|0\rangle$ and $|1\rangle$ states. This circuit typically uses the Hadamard gate.

Purpose:
Superposition allows quantum computers to evaluate multiple possibilities simultaneously, a fundamental aspect of quantum parallelism.

Circuit:

- Apply the Hadamard gate to a qubit initially in state $|0\rangle$.
- The qubit will be transformed into a state where it has an equal probability of being measured as $|0\rangle$ or $|1\rangle$.

Example:
Applying the Hadamard gate to a qubit:

$$H|0\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$$

This state has a 50% chance of being $|0\rangle$ and a 50% chance of being $|1\rangle$ when measured.

2. Bell State Circuit (Entanglement Circuit)

The Bell state circuit creates an entangled state between two qubits. Entanglement is a unique quantum property where the state of one qubit is directly correlated with the state of another, no matter the distance between them.

Purpose:
Bell states are foundational for quantum communication, quantum cryptography, and quantum teleportation.

Circuit:

- Apply a Hadamard gate to qubit 0 to put it in superposition.
- Apply a CNOT gate, with qubit 0 as the control and qubit 1 as the target, to entangle the qubits.

  Resulting State:

$$|\psi\rangle = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle)$$

This Bell state means that if one qubit is measured as $|0\rangle$, the other will also be $|0\rangle$; if one is $|1\rangle$, the other will be $|1\rangle$.

3. Quantum Teleportation Circuit

Quantum teleportation allows the transfer of a qubit's quantum state from one qubit to another distant qubit, without directly transferring the qubit itself. This circuit demonstrates quantum entanglement and the use of classical communication.

Purpose:

Quantum teleportation is important for secure quantum communication and distributed quantum computing.

Circuit:

- Entangle qubits 1 and 2 by creating a Bell state.
- Prepare qubit 0 in the state to be teleported.
- Apply a CNOT gate and a Hadamard gate to qubit 0 and measure the qubits, transmitting the results classically.
- Apply conditional operations (Pauli gates) to qubit 2 to complete the teleportation.

Result:

After running this circuit and applying corrections, qubit 2 will have the same state as the original state of qubit 0.

4. Quantum Fourier Transform (QFT) Circuit

The Quantum Fourier Transform (QFT) is the quantum equivalent of the classical Fourier transform. It maps a quantum state to the frequency domain, which can help detect periodicities in data.

Purpose:

QFT is a crucial component in algorithms like Shor's algorithm for factoring and phase estimation algorithms in quantum chemistry.

Circuit:

- The QFT circuit consists of Hadamard gates and controlled phase rotation gates applied in a specific sequence to each qubit.

- The QFT is often followed by a swap operation to rearrange qubit order.

Resulting State:
The QFT transforms the input quantum state into a state that represents the frequency components of the data.

5. Grover's Search Algorithm Circuit

Grover's algorithm is designed to search an unsorted database for a marked item with quadratic speedup compared to classical search algorithms.
Purpose:
Grover's algorithm is used in quantum search problems and optimization tasks.
Circuit:

- Initialize all qubits in a superposition using Hadamard gates.
- Apply the Oracle: This phase flips the amplitude of the solution state.
- Apply the Diffusion Operator: This operator amplifies the probability of the solution state.
- Repeat the Oracle and Diffusion steps until the probability of the correct solution is maximized.

Result:
After the final iteration, measuring the circuit should yield the solution state with high probability.

6. Phase Estimation Circuit

Phase estimation is a quantum algorithm for estimating the eigenvalue (or phase) of a unitary operator. This circuit is used in applications such as finding the energy levels of molecules in quantum chemistry.
Purpose:
The phase estimation circuit is foundational in quantum chemistry and number-theoretic algorithms like factoring.
Circuit:

- Prepare a register of qubits in superposition.
- Apply controlled-unitary gates to the target qubit, introducing phase shifts.
- Apply the Quantum Fourier Transform (QFT) to the register qubits.
- Measure the result, which reveals the phase information.

Result:
The phase estimation algorithm outputs an approximation of the phase of the eigenstate, which is useful for tasks like finding energy levels of molecules.

7. Quantum Half-Adder Circuit

A quantum half-adder circuit performs the addition of two single-bit binary numbers, outputting a sum and a carry.

Purpose:
Used as a basic building block in quantum arithmetic operations.

Circuit:

- Input two qubits representing binary numbers.
- Apply a CNOT gate for the sum bit and a Toffoli (CCX) gate for the carry bit.

Result:
Outputs the sum and carry of the binary addition on two qubits.

8. Swap Test Circuit

The swap test circuit is used to determine the similarity between two quantum states without directly measuring them. It's often used to measure the fidelity or overlap of two quantum states.

Purpose:
Commonly used in quantum machine learning and quantum data comparison tasks.

Circuit:

- Add an ancilla qubit in the $|+\rangle$ state.
- Apply controlled-SWAP (Fredkin) gates between the two states.
- Measure the ancilla qubit.

Result:
If the ancilla qubit is measured as $|0\rangle$, it indicates similarity between the states with a probability based on the overlap.

9. Deutsch-Jozsa Algorithm Circuit

The Deutsch-Jozsa algorithm determines whether a given function is constant or balanced in a single query. This algorithm showcases quantum speedup over classical deterministic methods.

Purpose:
This circuit is an early example of quantum algorithms providing exponential speedup.

Circuit:

- Prepare the input qubits in superposition using Hadamard gates.
- Apply the oracle function to determine if the function is constant or balanced.

- Apply Hadamard gates again and measure.

Result:
If the result is all zeros, the function is constant; otherwise, it is balanced.

Summary of Common Quantum Circuits

Circuit	**Purpose**	**Key Gates Used**	**Applications**
Superposition	Creates superposition in qubits	Hadamard	Quantum parallelism
Bell State	Entangles two qubits	Hadamard, CNOT	Quantum cryptography, teleportation
Teleportation	Transfers a qubit state to another qubit	Hadamard, CNOT, Pauli gates	Quantum communication
QFT	Maps quantum states to frequency domain	Hadamard, Controlled Phase	Shor's algorithm, phase estimation
Grover's Search	Searches an unsorted database	Hadamard, Oracle, Diffusion	Search and optimization
Phase Estimation	Finds phase/eigenvalues of unitary operators	Controlled-unitary, QFT	Quantum chemistry, factoring
Quantum Adder	Adds binary numbers	CNOT, Toffoli	Quantum arithmetic
Swap Test	Compares similarity of two quantum states	Hadamard, Fredkin (controlled-SWAP)	Quantum machine learning
Deutsch-Jozsa	Determines if function is constant or balanced	Hadamard, Oracle	Demonstration of quantum speedup

These quantum circuits are foundational in quantum computing, enabling quantum computers to handle complex tasks more efficiently than classical computers for specific applications. Each circuit exploits quantum phenomena like superposition, entanglement, and interference to achieve functionality that is impractical in classical systems.