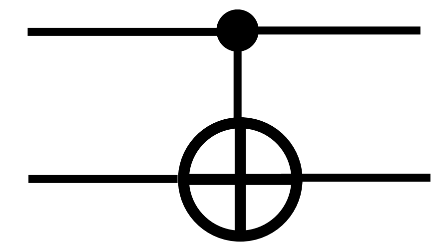
Day 1 Recap

- Introduction: ML and QML
  - ML: Universal approximation theorem
  - QML: parametrize the cost function with quantum algorithms and use classical optimizers
- Single qubit
  - Bloch sphere
  - Separable vs entangled states
  - Computational basis/Hadamard basis
  - Quantum circuits are expressed by unitary transformations and measurement
  - Measurement: inner product / projection
  - Single qubit gates: X, Y, Z, Hadamard, etc
- A system of two or more qubits
  - Tensor products

Day 2 Plan

- Two qubit gates
  - CNOT, SWAP
- No cloning
- Superdense coding
- Three qubit gates
  - Controlled CNOT, Controlled SWAP
- Teleportation
- A simple QA with two qubits: Deutsch Algorithm
- Deutsch-Jozsa algorithm
- Bernstein-Vazirani Algorithm and Simon's algorithm
- Quantum Fourier Transformation

Two Qubit Gates: CNOT and CU gates



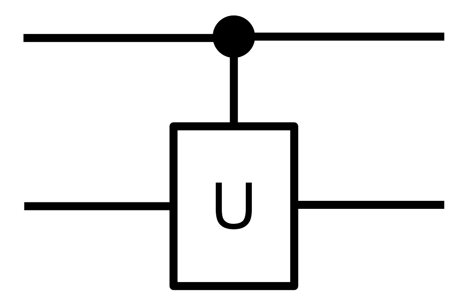
- CNOT gate = Controlled Not =Controlled X
- NOT operation is performed on 2nd qubit, when the 1st qubit is in state $|1\rangle$. Otherwise 2nd qubit is unchanged.

$$\begin{aligned} |00\rangle &\rightarrow |00\rangle \\ |01\rangle &\rightarrow |01\rangle \\ |10\rangle &\rightarrow |11\rangle \\ |11\rangle &\rightarrow |10\rangle \end{aligned} \qquad \begin{pmatrix} |00\rangle' \\ |01\rangle' \\ |10\rangle' \\ |11\rangle' \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \begin{pmatrix} |00\rangle \\ |01\rangle \\ |10\rangle \\ |11\rangle \end{pmatrix}$$

$$\begin{pmatrix} I & 0 \\ 0 & X \end{pmatrix} = \exp\left(i\frac{\pi}{4}(I - Z_1)(I - X_2) \right)$$

$$|ij\rangle \rightarrow |i\, i \oplus j\rangle \quad \text{(mod 2)}$$

- Generally, controlled U-gate



$$\begin{aligned} |00\rangle &\rightarrow |00\rangle \\ |01\rangle &\rightarrow |01\rangle \\ |10\rangle &\rightarrow |1\rangle \otimes U|0\rangle = |1\rangle \otimes \left(U_{00}|0\rangle + U_{01}|1\rangle\right) \\ |11\rangle &\rightarrow |1\rangle \otimes U|1\rangle = |1\rangle \otimes \left(U_{10}|0\rangle + U_{11}|1\rangle\right) \end{aligned}$$

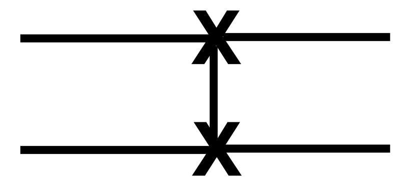
U: any arbitrary unitary matrix. U=X, Y, Z leads to CX, CY, CZ gates.

$$CU = \begin{pmatrix} I & 0 \\ 0 & U \end{pmatrix} = \exp\left(i\frac{1}{2}(I - Z_1)H_2 \right) \text{ for } U = e^{iH_2} = \begin{pmatrix} U_{00} & U_{01} \\ U_{10} & U_{11} \end{pmatrix}$$

$$e^{i\theta A} = \cos\theta + i\,A\,\sin\theta \text{ for } A^2 = I$$

Two Qubit Gates: SWAP and CPhase gates

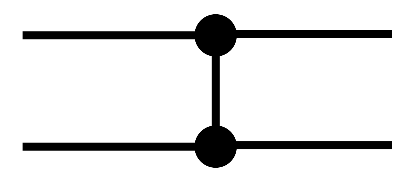
- SWAP gate: $|ab\rangle \rightarrow |ba\rangle$



$$|00\rangle \rightarrow |00\rangle$$
$$|01\rangle \rightarrow |10\rangle$$
$$|10\rangle \rightarrow |01\rangle$$
$$|11\rangle \rightarrow |11\rangle$$

$$\mathrm{SWAP} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} = \frac{1}{2}\left[I \otimes I + X \otimes X + Y \otimes Y + Z \otimes Z \right]$$

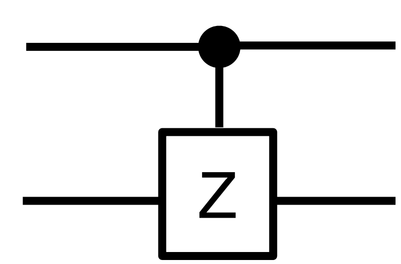
- CPhase gate = Controlled phase shift:
  shift phase by ϕ only if it acts on $|1\rangle$



$$|ab\rangle \rightarrow \begin{cases} |ab\rangle e^{i\phi} & \text{for } a = b = 1 \\ |ab\rangle & \text{otherwise} \end{cases}$$

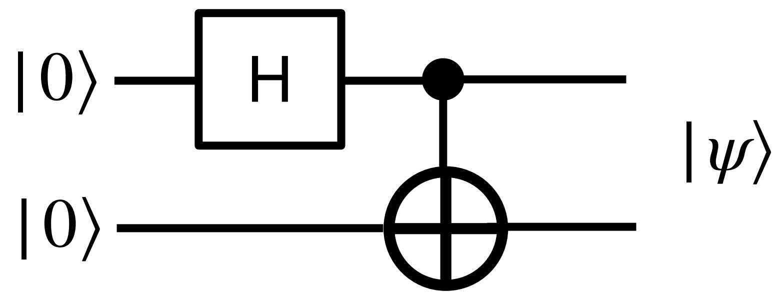
$$CPhase(\phi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & e^{i\phi} \end{pmatrix} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes P_\phi, \qquad P_\phi = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\phi} \end{pmatrix} = |0\rangle\langle 0| + |1\rangle\langle 1| e^{i\phi}$$

$$CPhase(\pi) = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{pmatrix} = \mathrm{CZ} = \text{Controlled Z}$$



Two Qubit Gates: Bell state

- Example: how to obtain Bell state.



$$
\begin{aligned}
|\psi\rangle &= \mathrm{CNOT}\,(H\otimes I)\left[|0\rangle\otimes|0\rangle\right]\\
&= \mathrm{CNOT}\left[\frac{1}{\sqrt{2}}\big(|0\rangle+|1\rangle\big)\otimes|0\rangle\right]\\
&= \mathrm{CNOT}\left[\frac{1}{\sqrt{2}}\big(|00\rangle+|10\rangle\big)\right]\\
&= \frac{1}{\sqrt{2}}\big(|00\rangle+|11\rangle\big)\\
&= \frac{1}{\sqrt{2}}\begin{pmatrix}1\\0\\0\\1\end{pmatrix}
\end{aligned}
$$

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

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix}1 & 1\\1 & -1\end{pmatrix} = \frac{1}{\sqrt{2}}\Big(|0\rangle\langle 0| + |0\rangle\langle 1| + |1\rangle\langle 0| - |1\rangle\langle 1|\Big)$$

$$H|0\rangle = |+\rangle \qquad H|+\rangle = |0\rangle$$

$$H|1\rangle = |-\rangle \qquad H|-\rangle = |1\rangle$$

No-cloning theorem

- Unknown quantum states can not be copied or cloned.
  - Suppose U is a unitary transformation that clones $U\left(|a\rangle|0\rangle\right) = |a\rangle|a\rangle$ for all quantum state $|a\rangle$
  - Let $|a\rangle$ and $|b\rangle$ be two orthogonal quantum states.

$$U\left(|a\rangle|0\rangle\right) = |a\rangle|a\rangle$$

$$U\left(|b\rangle|0\rangle\right) = |b\rangle|b\rangle$$

$$|c\rangle = \frac{1}{\sqrt{2}}\left(|a\rangle + |b\rangle\right)$$

$$\Rightarrow \quad U\left(|c\rangle|0\rangle\right) = \frac{1}{\sqrt{2}}\left[U|a\rangle|0\rangle + U|b\rangle|0\rangle\right]$$

$$= \frac{1}{\sqrt{2}}\left[|a\rangle|a\rangle + |b\rangle|b\rangle\right]$$

$$\neq$$

$$\Rightarrow \quad U|c\rangle|0\rangle = |c\rangle|c\rangle = \frac{1}{\sqrt{2}}\left(|a\rangle + |b\rangle\right)\frac{1}{\sqrt{2}}\left(|a\rangle + |b\rangle\right)$$

$$= \frac{1}{2}\left(|a\rangle|a\rangle + |a\rangle|b\rangle + |b\rangle|a\rangle + |b\rangle|b\rangle\right)$$

No-cloning theorem

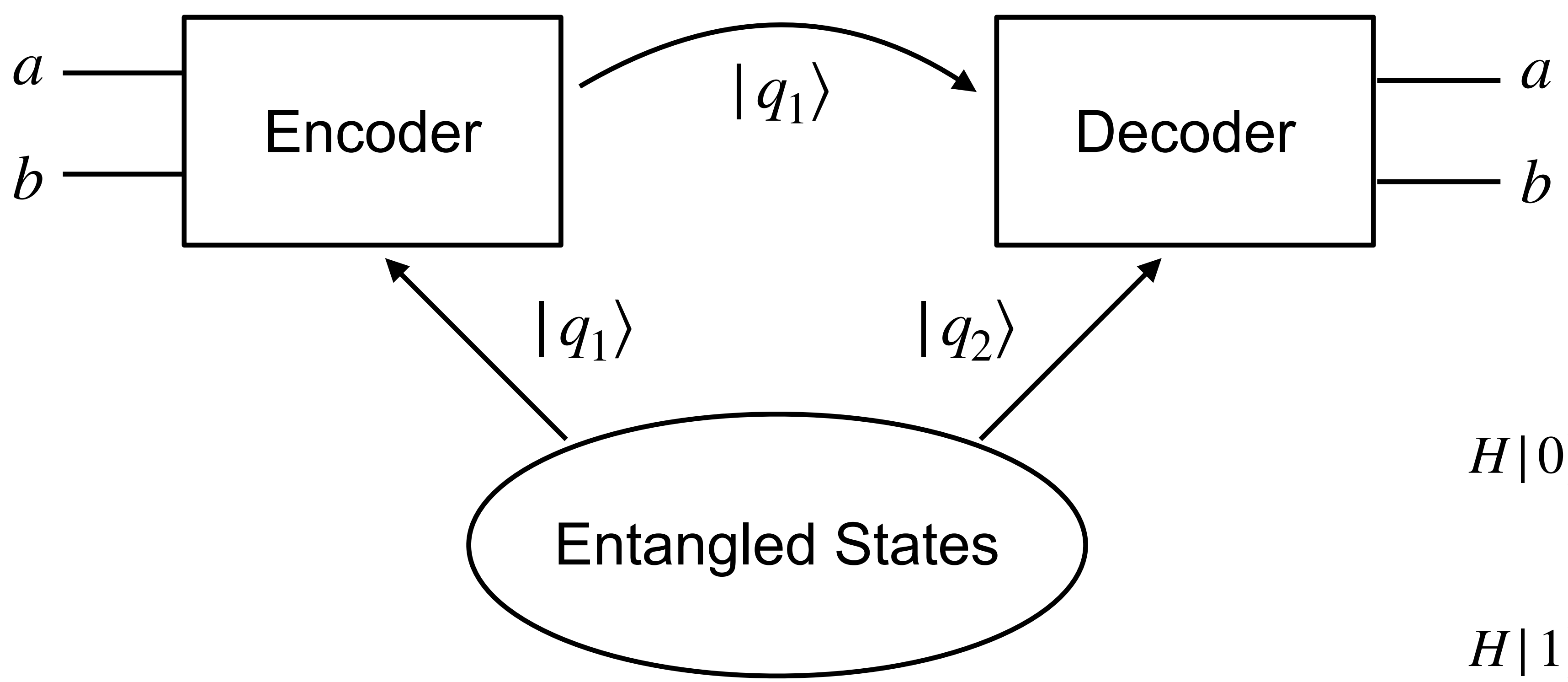
- No unitary operation that can clone all quantum states.
- However it is possible to construct a quantum state from a known quantum state.

- It is possible to obtain n particles in an entangled state $a|00\cdots0\rangle + b|11\cdots1\rangle$ from unknown state $a|0\rangle + b|1\rangle$.

- It is not possible to create n particle state $\left(a|0\rangle + b|1\rangle\right) \otimes \cdots \otimes \left(a|0\rangle + b|1\rangle\right)$ from an unknown state $a|0\rangle + b|1\rangle$.

- Profound implication in quantum information and error correction.

Superdense Coding



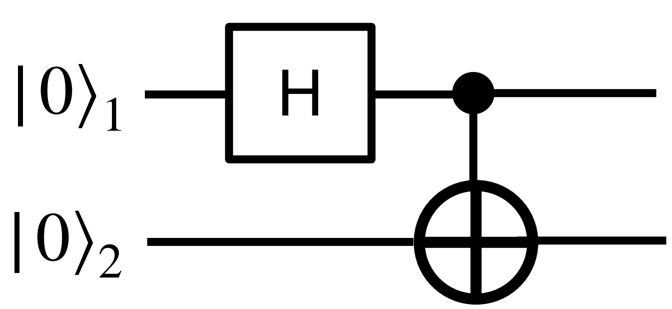
$$H|0\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big)$$

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

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

$$\text{CNOT}\,|a\,b\rangle = |a\,a \oplus b\rangle$$

- How to create two entangled states



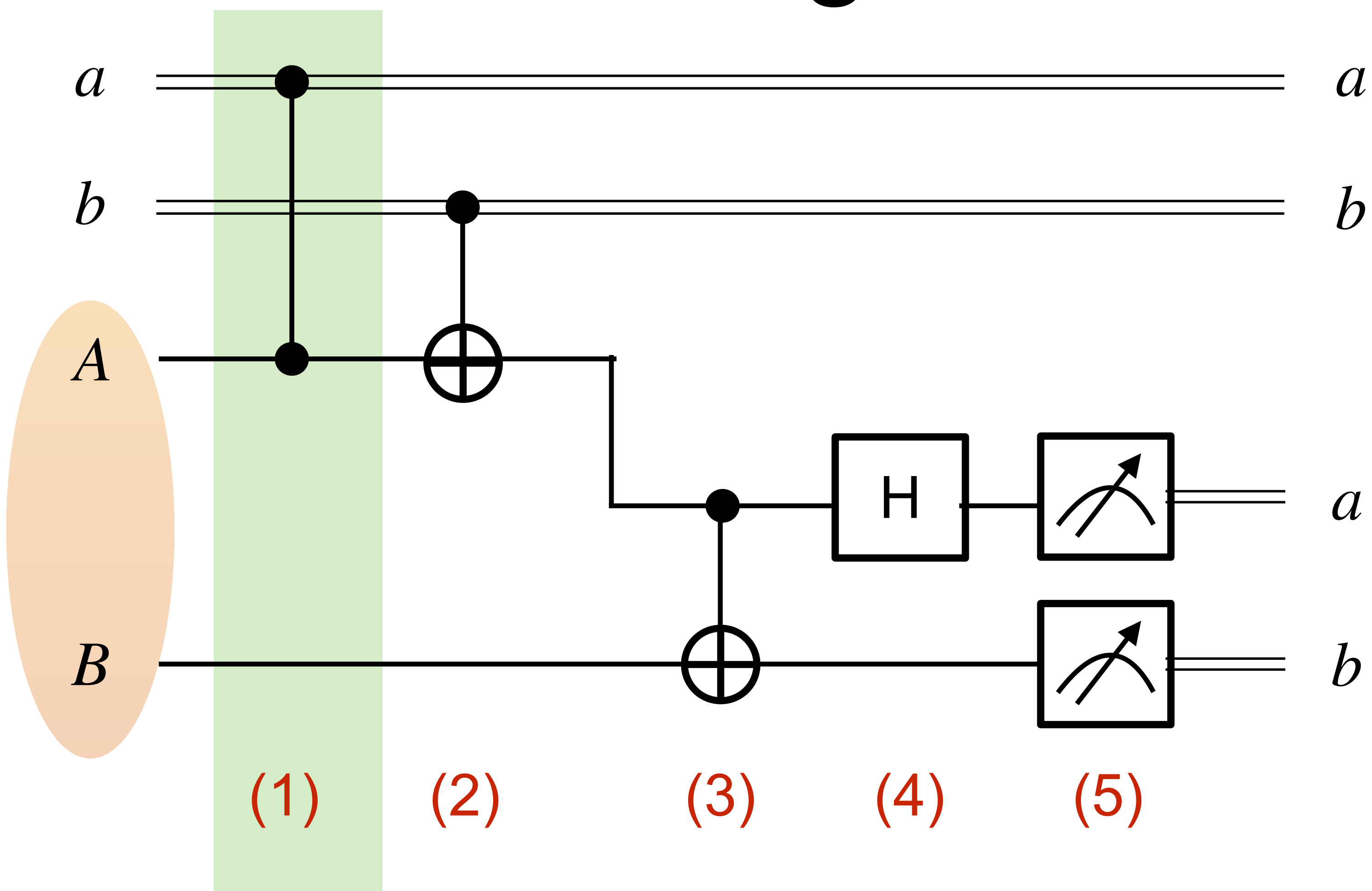
$$\text{CNOT}\Big(H \otimes I\Big)\Big(|0\rangle_1 \otimes |0\rangle_2\Big) = \text{CNOT}\,\frac{1}{\sqrt{2}}\Big(|0\rangle_1 + |1\rangle_1\Big) \otimes |0\rangle_2$$

$$= \text{CNOT}\,\frac{1}{\sqrt{2}}\Big(|00\rangle + |10\rangle\Big) = \frac{1}{\sqrt{2}}\Big(|00\rangle + |11\rangle\Big)$$

Superdense Coding

- Initial state of qubits A and B is the entangled Bell state.

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + |11\rangle\,\Big]$$



(1) $a, b \in \{0,1\}$ are classical bits.

Controlled phase gate = CZ ($\phi = \pi$)

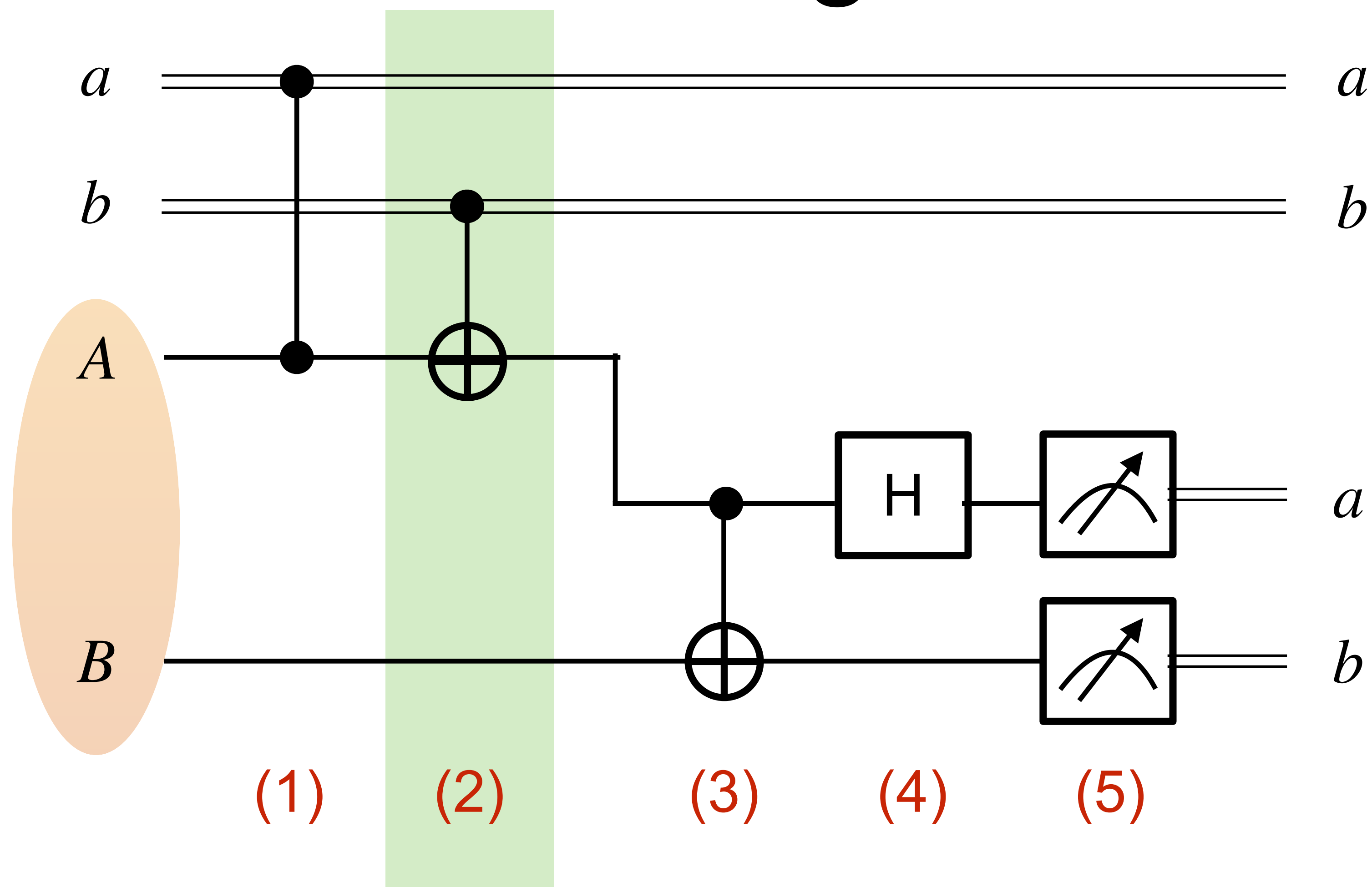
$$\text{if } a = 1,\ |1\rangle \longrightarrow -|1\rangle$$
$$|0\rangle \longrightarrow +|0\rangle$$
$$\text{if } a = 0,\ |0\rangle \longrightarrow +|0\rangle$$
$$|1\rangle \longrightarrow +|1\rangle$$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + (-1)^a\,|11\rangle\,\Big]$$

Superdense Coding

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + |11\rangle\,\Big]$$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + (-1)^a\,|11\rangle\,\Big]$$



(2) If b=0, the first qubit stays unchanged.

If b=1, the first qubit changes bit.

$$\text{CNOT}: \begin{array}{lcl} |00\rangle & \longrightarrow & |00\rangle \\ |01\rangle & \longrightarrow & |01\rangle \\ |10\rangle & \longrightarrow & |11\rangle \\ |11\rangle & \longrightarrow & |10\rangle \end{array}$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}\Big[\,|b0\rangle + (-1)^a\,|\bar{b}1\rangle\,\Big]$$

$$b = 0 \iff \bar{b} = 1$$

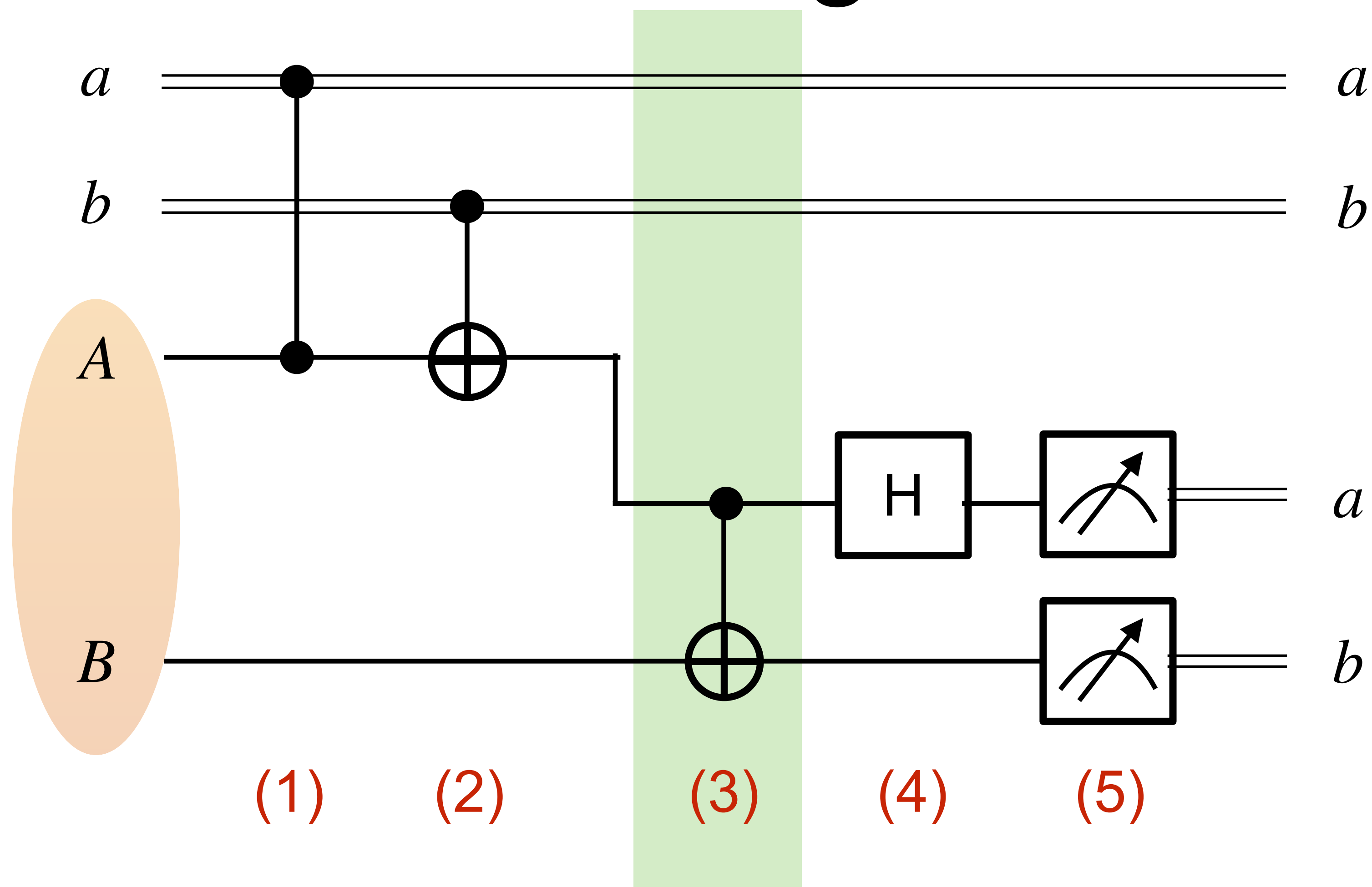
$$b = 1 \iff \bar{b} = 0$$

Superdense Coding

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + |11\rangle\,\Big]$$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle + (-1)^a|11\rangle\,\Big]$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}\Big[\,|b0\rangle + (-1)^a|\bar{b}1\rangle\,\Big]$$



Alice gives her qubit to Bob.

$$\mathrm{CNOT}\,|b0\rangle = |bb\rangle$$

$$\mathrm{CNOT}\,|\bar{b}1\rangle = |\bar{b}b\rangle$$

(3) Bob performs CNOT.

$$|\psi_3\rangle = \mathrm{CNOT}\,|\psi_2\rangle$$

$$= \mathrm{CNOT}\frac{1}{\sqrt{2}}\Big[|b0\rangle + (-1)^a|\bar{b}1\rangle\Big]$$

$$= \frac{1}{\sqrt{2}}\Big[|bb\rangle + (-1)^a|\bar{b}b\rangle\Big]$$

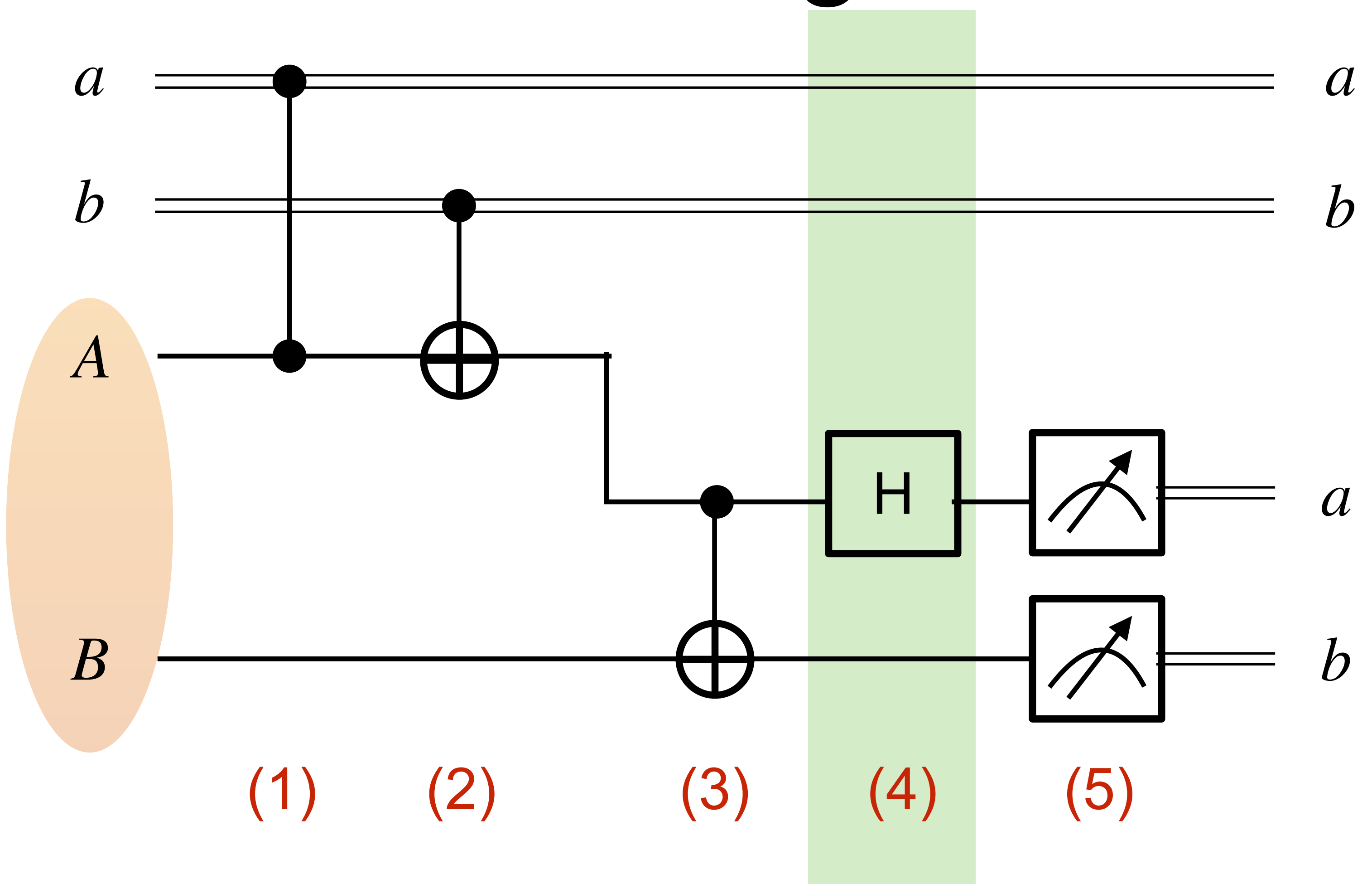
Superdense Coding

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle \;+\; |11\rangle\,\Big]$$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle \;+\; (-1)^a\,|11\rangle\,\Big]$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}\Big[\,|b0\rangle \;+\; (-1)^a\,|\bar{b}1\rangle\,\Big]$$

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}\Big[\,|bb\rangle + (-1)^a\,|\bar{b}b\rangle\Big]$$



(4) Bob applies Hadamard.

$$|\psi_4\rangle = \Big(H\otimes I\Big)|\psi_3\rangle = \Big(H\otimes I\Big)\frac{1}{\sqrt{2}}\Big[|bb\rangle + (-1)^a|\bar{b}b\rangle\Big]$$

$$= \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}\Big[|0b\rangle + (-1)^b|1b\rangle + (-1)^a\Big(|0b\rangle + (-1)^{\bar{b}}|1b\rangle\Big)\Big]$$

$$= \frac{1}{2}\Big[\Big(1 + (-1)^a\Big)|0b\rangle + \Big((-1)^b + (-1)^{a+\bar{b}}\Big)|1b\rangle\Big]$$

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

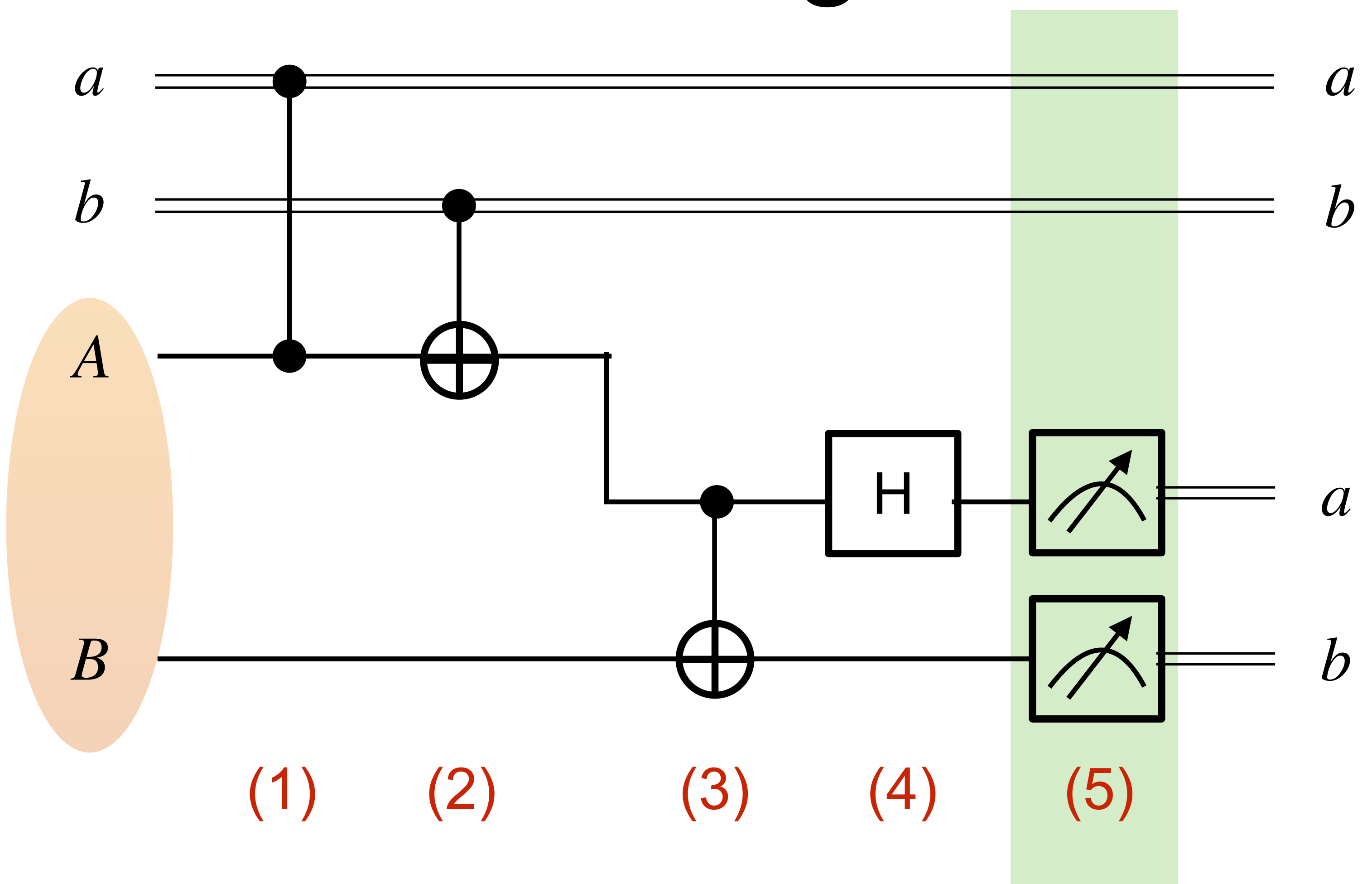
Superdense Coding

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle \;+\; |11\rangle\,\Big]$$

$$|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\,|00\rangle \;+\; (-1)^a\,|11\rangle\,\Big]$$

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}\Big[\,|b0\rangle \;+\; (-1)^a\,|\bar{b}1\rangle\,\Big]$$

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}\Big[\,|bb\rangle + (-1)^a\,|\bar{b}b\rangle\Big]$$



(4) Bob applies Hadamard.

$$|\psi_4\rangle = \frac{1}{2}\Big[\big(1+(-1)^a\big)\,|0\rangle + \big((-1)^b + (-1)^{a+\bar{b}}\big)\,|1\rangle\Big] \otimes |b\rangle$$

$$= \frac{1}{2}\Big[\big(1+(-1)^a\big)\,|0\rangle + (-1)^b\big(1-(-1)^a\big)\,|1\rangle\Big] \otimes |b\rangle$$

(5) Bob performs measurements.

Superdense Coding

a	b	$\bar{b}$	$a+\bar{b}$	$\lvert A\rangle$	$\lvert B\rangle$
0	0	1	1	$\lvert 0\rangle$	$\lvert 0\rangle$
0	1	0	0	$\lvert 0\rangle$	$\lvert 1\rangle$
1	0	1	0=2	$\lvert 1\rangle$	$\lvert 0\rangle$
1	1	0	1	$-\lvert 1\rangle$	$\lvert 1\rangle$

$$|\psi_4\rangle = |A\rangle \otimes |B\rangle = \frac{1}{2}\Big[\big(1+(-1)^a\big)|0\rangle + \big((-1)^b + (-1)^{a+\bar{b}}\big)|1\rangle\Big] \otimes |B\rangle$$

$$|\psi_4\rangle = (-1)^{ab}|ab\rangle = (-1)^{ab}|a\rangle \otimes |b\rangle$$

- Measurement of two qubits yield two classical bits a and b with 100% probability.
- By initially sharing some entanglement, one can send two bits of information by sending a single qubit.
- Shared entanglement → powerful resource for quantum cryptography

Superdense Coding

$$|\psi_0\rangle = \frac{1}{\sqrt{2}}\left(|00\rangle + |11\rangle \right)$$

a	b	Transformation (Alice)	New state
0	0	$I \otimes I \lvert\psi_0\rangle$	$\frac{1}{\sqrt{2}}\left(\lvert 00\rangle + \lvert 11\rangle \right)$
0	1	$X \otimes I \lvert\psi_0\rangle$	$\frac{1}{\sqrt{2}}\left(\lvert 10\rangle + \lvert 01\rangle \right)$
1	0	$Z \otimes I \lvert\psi_0\rangle$	$\frac{1}{\sqrt{2}}\left(\lvert 00\rangle - \lvert 11\rangle \right)$
1	1	$Y \otimes I \lvert\psi_0\rangle$	$\frac{1}{\sqrt{2}}\left(-\lvert 10\rangle + \lvert 01\rangle \right)$

- Bob measures two qubits in the standard basis to obtain two-bit binary encoding of the number that Alice wishes to send.

$\longrightarrow$

Alice gives her qubit to Bob.

CNOT (Bob)

$$\frac{1}{\sqrt{2}}\left(|00\rangle + |10\rangle \right) = \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle \right) \otimes |0\rangle$$

$$\frac{1}{\sqrt{2}}\left(|11\rangle + |01\rangle \right) = \frac{1}{\sqrt{2}}\left(|1\rangle + |0\rangle \right) \otimes |1\rangle$$

$$\frac{1}{\sqrt{2}}\left(|00\rangle - |10\rangle \right) = \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle \right) \otimes |0\rangle$$

$$\frac{1}{\sqrt{2}}\left(-|11\rangle + |01\rangle \right) = \frac{1}{\sqrt{2}}\left(-|1\rangle + |0\rangle \right) \otimes |1\rangle$$

$\xrightarrow{H \otimes I}$

$$|0\rangle \otimes |0\rangle$$

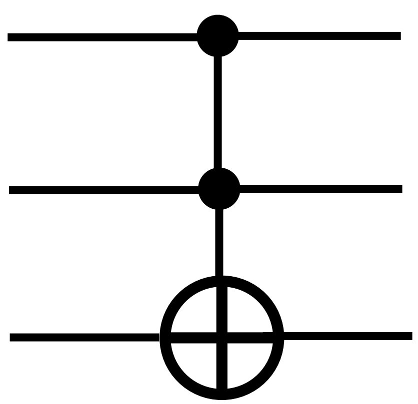
$$|0\rangle \otimes |1\rangle$$

$$|1\rangle \otimes |0\rangle$$

$$-|1\rangle \otimes |1\rangle$$

Three Qubit Gates

- Toffoli gate=Controlled CNOT=CCNOT=CCX=T
  - If 1st qubit is $|1\rangle$, perform CNOT on the second and third qubits.



$|000\rangle$
$|001\rangle$
$|010\rangle$
$|011\rangle$

$|100\rangle$
$|101\rangle$

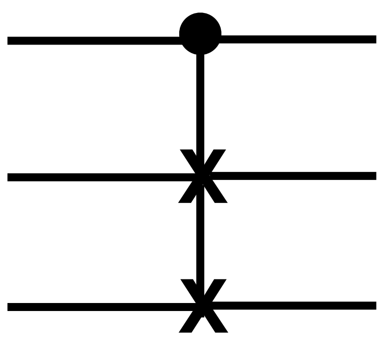
$|110\rangle \longrightarrow |111\rangle$
$|111\rangle \quad\quad\;\; |110\rangle$

$$T = \left(\begin{array}{cccc:cccc} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right) = \left(\begin{array}{c:c} I & 0 \\ \hdashline 0 & \text{CNOT} \end{array}\right)$$

$$T = \exp\left[i\frac{\pi}{8}(I - Z_1)(I - Z_2)(I - X_3) \right]$$

Three Qubit Gates

- Fredkin gate=Controlled SWAP=CSWAP gate
  - If 1st qubit is $|1\rangle$, swap the second and third qubits.

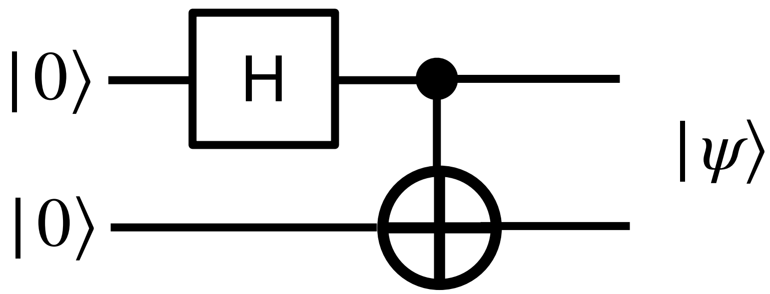


$$\begin{array}{ccc}
|000\rangle & & \\
|001\rangle & & \\
|010\rangle & & \\
|011\rangle & & \\
\hdashline
|100\rangle & \longrightarrow & |100\rangle \\
|101\rangle & & |110\rangle \\
|110\rangle & & |101\rangle \\
|111\rangle & & |111\rangle
\end{array}$$

$$T = \left(\begin{array}{cccc:cccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hdashline
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{array}\right) = \left(\begin{array}{c:c}
I & 0 \\
\hdashline
0 & \mathrm{SWAP}
\end{array}\right)$$

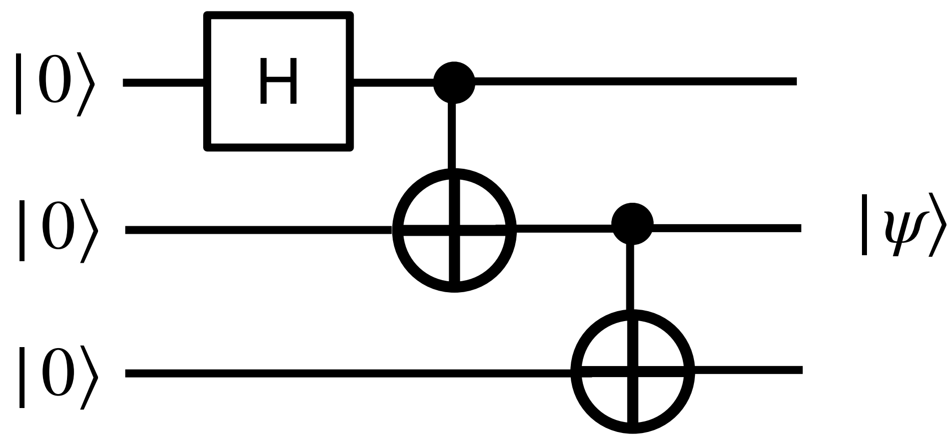
Two Qubit Gates: Bell state

- Example: how to obtain Bell state.



$$
\begin{aligned}
|\psi\rangle &= \mathrm{CNOT}\,(H \otimes I)\left[|0\rangle \otimes |0\rangle\right] \\
&= \mathrm{CNOT}\left[\frac{1}{\sqrt{2}}\big(|0\rangle + |1\rangle\big) \otimes |0\rangle\right] \\
&= \mathrm{CNOT}\left[\frac{1}{\sqrt{2}}\big(|00\rangle + |10\rangle\big)\right] = \frac{1}{\sqrt{2}}\big(|00\rangle + |11\rangle\big) \\
&= \frac{1}{\sqrt{2}}\big(|00\rangle + |11\rangle\big) \\
&= \frac{1}{\sqrt{2}}\begin{pmatrix} 1 \\ 0 \\ 0 \\ 1 \end{pmatrix}
\end{aligned}
$$

An example: GHZ state



$$|\psi\rangle = \frac{|000\rangle + |111\rangle}{\sqrt{2}}$$

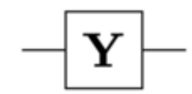
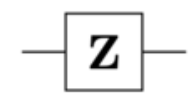
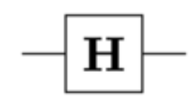
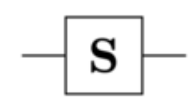
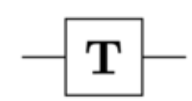
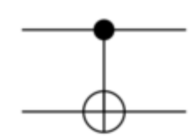
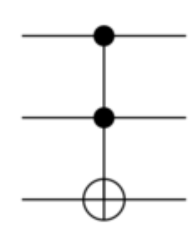
Greenberger-Horne-Zeilinger (GHZ) state, 1989

$$\begin{aligned}
|\psi\rangle &= \left(I_1 \otimes CNOT_{23}\right)\left(CNOT_{23} \otimes I_3\right)\left(H \otimes I_2 \otimes I_3\right)|0\rangle \otimes |0\rangle \otimes |0\rangle \\
&= \left(I_1 \otimes CNOT_{23}\right)\frac{1}{\sqrt{2}}\Big(|00\rangle + |11\rangle\Big) \otimes |0\rangle \\
&= \left(I_1 \otimes CNOT_{23}\right)\frac{1}{\sqrt{2}}\Big(|000\rangle + |110\rangle\Big) \\
&= \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes CNOT|00\rangle + |1\rangle \otimes CNOT|10\rangle\Big) \quad = \frac{|000\rangle + |111\rangle}{\sqrt{2}}
\end{aligned}$$

For N-qubit system: $$|GHZ\rangle = \frac{|0\rangle^{\otimes N} + |1\rangle^{\otimes N}}{\sqrt{2}} = \frac{|00\cdots0\rangle \; + \; |11\cdots1\rangle}{\sqrt{2}}$$

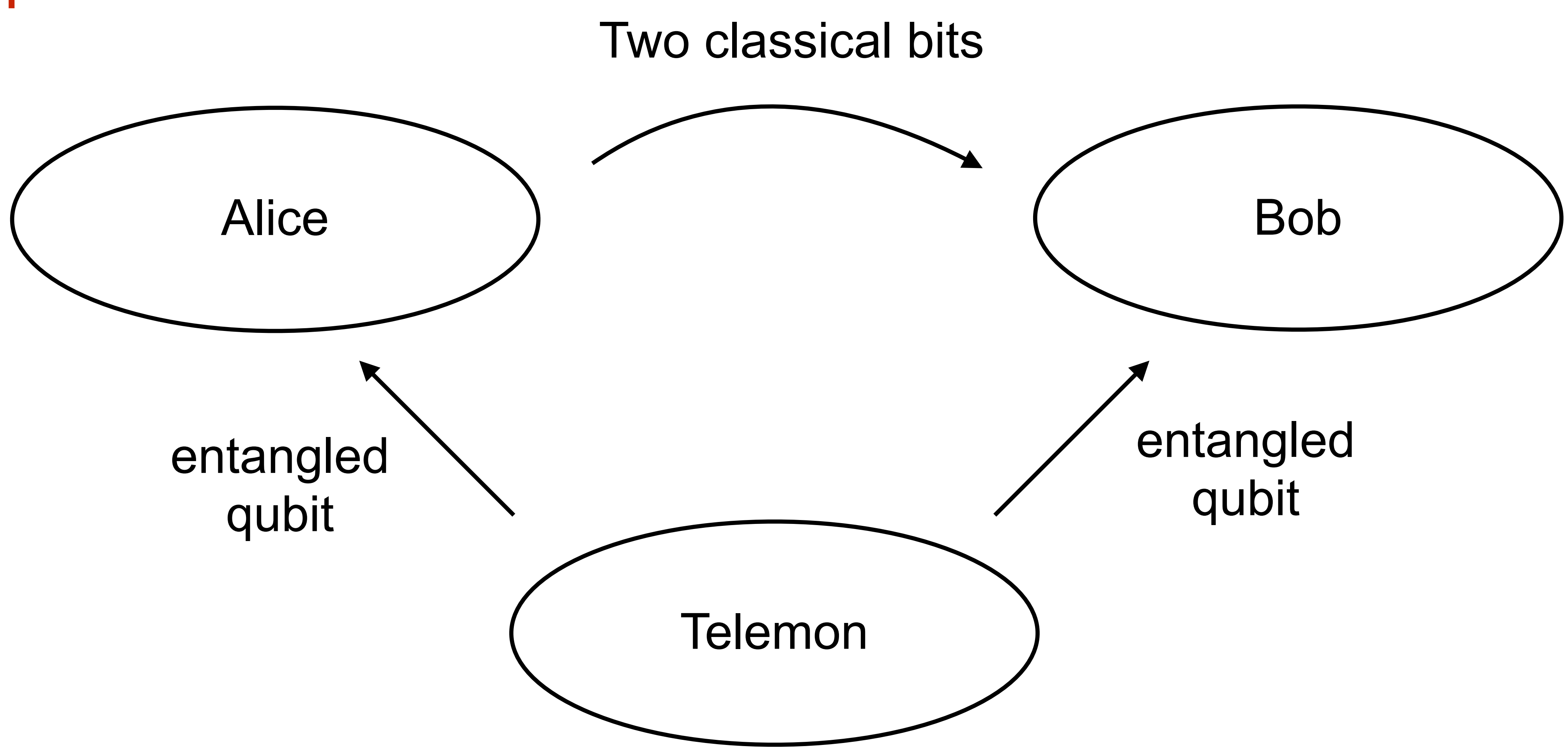
Maximally entangled quantum state

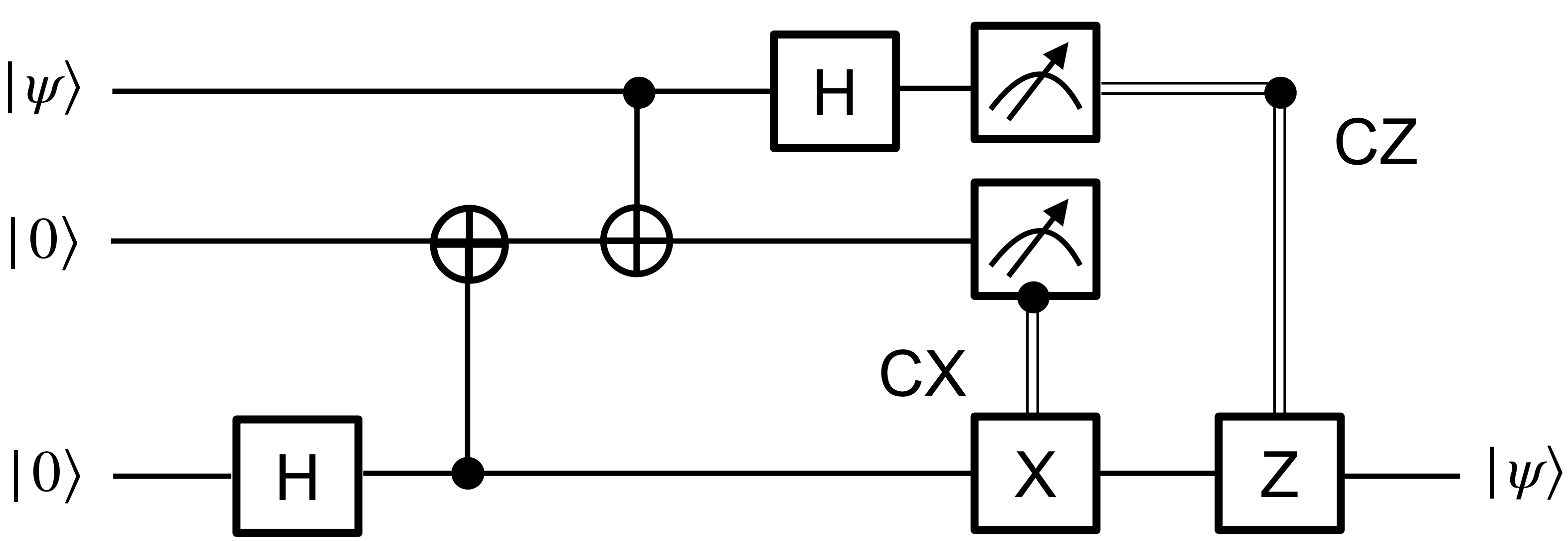
- IBMQ

Operator	Gate(s)	Matrix
Pauli-X (X)		$\begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$
Pauli-Y (Y)		$\begin{bmatrix} 0 & -i \\ i & 0 \end{bmatrix}$
Pauli-Z (Z)		$\begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$
Hadamard (H)		$\frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$
Phase (S, P)		$\begin{bmatrix} 1 & 0 \\ 0 & i \end{bmatrix}$
π/8 (T)		$\begin{bmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{bmatrix}$
Controlled Not (CNOT, CX)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$
Controlled Z (CZ)		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix}$
SWAP		$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$
Toffoli (CCNOT, CCX, TOFF)		$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}$

Teleportation

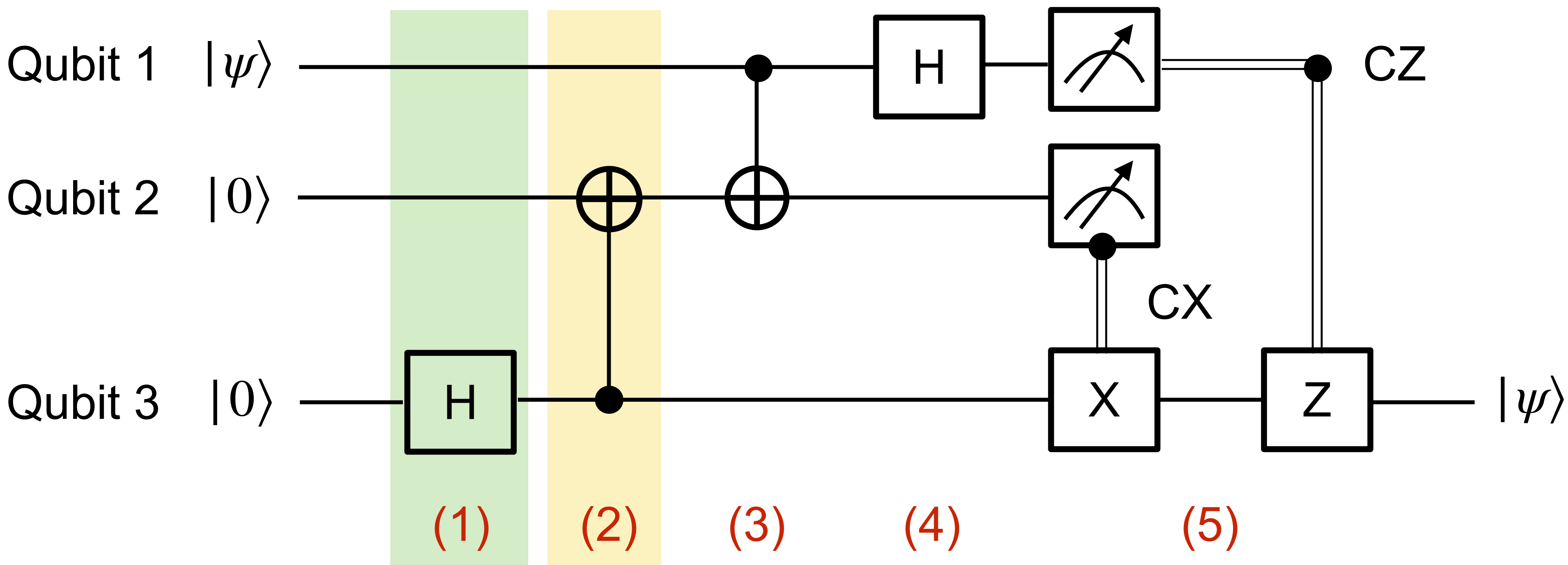
- Use two classical bits and one Bell pair to move a state from qubit 1 to qubit 3.





Teleportation

- Use two classical bits and one Bell pair to move a state from qubit 1 to qubit 3.



$$\text{initial state} = |\psi_0\rangle = |\psi\rangle_1 \otimes |0\rangle_2 \otimes |0\rangle_3$$

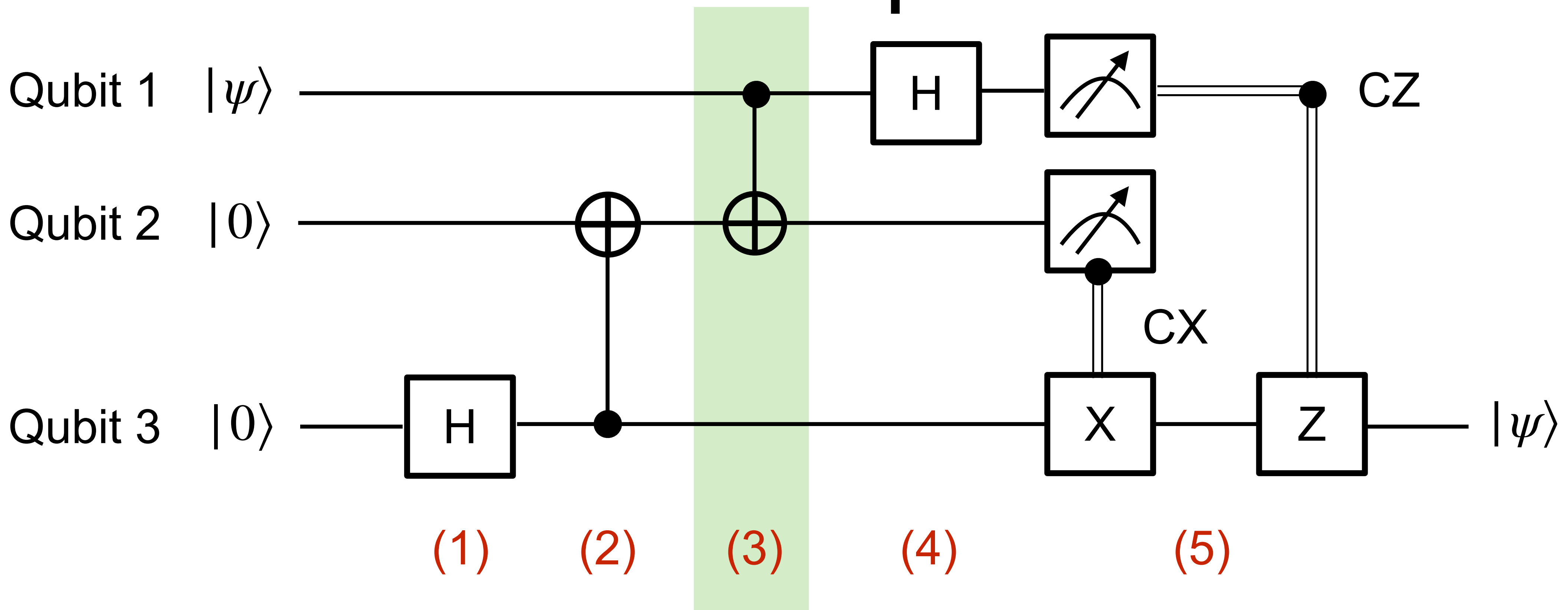
$$|\psi_1\rangle = H_3 |\psi\rangle_1 \otimes |0\rangle_2 \otimes |0\rangle_3 = |\psi\rangle_1 \otimes |0\rangle_2 \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle \Big)$$

$$|\psi_2\rangle = CNOT_3 |\psi\rangle_1 \otimes |0\rangle_2 \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle \Big)$$

$$= |\psi\rangle_1 \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle \Big)$$

conditioned on q3

Teleportation

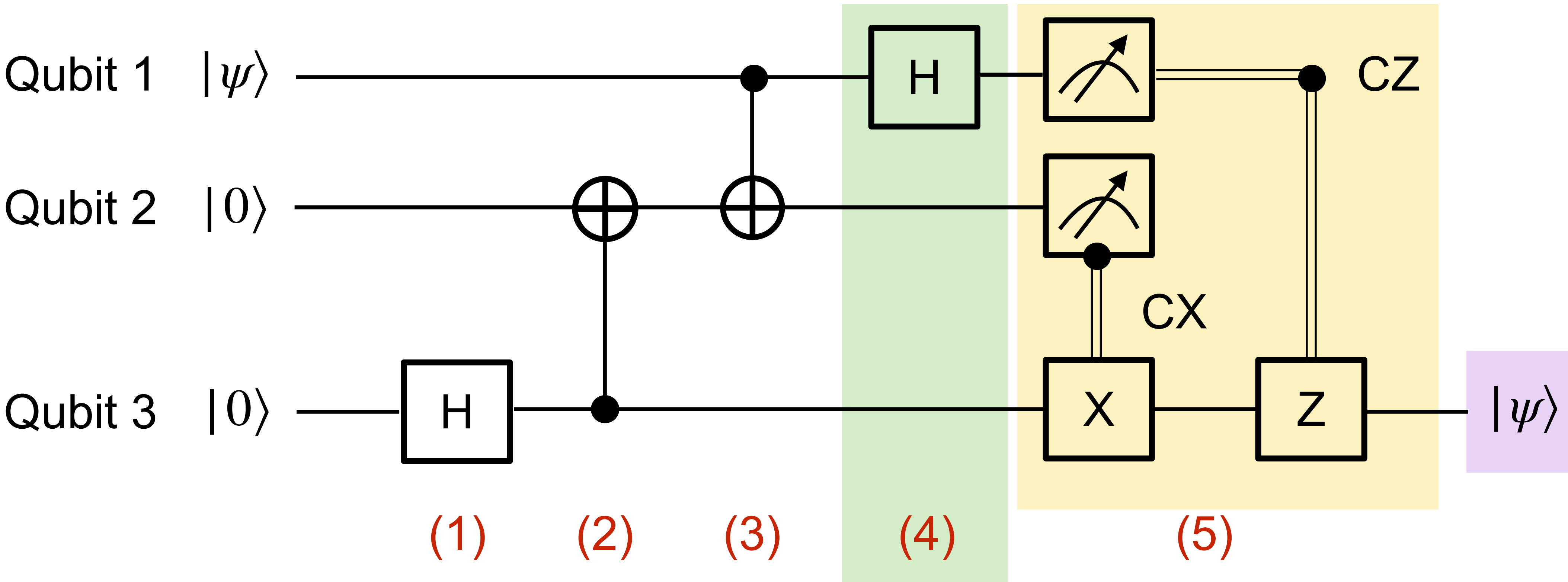


$$|\psi_2\rangle = |\psi\rangle_1 \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle \Big)$$

$$\begin{aligned}
|\psi_3\rangle &= CNOT_1 \; |\psi\rangle_1 \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle \Big) \\
&= CNOT_1 \; \big(\alpha|0\rangle + \beta|1\rangle\big) \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes |0\rangle + |1\rangle \otimes |1\rangle \Big) \\
&= CNOT_1 \; \frac{1}{\sqrt{2}}\Big(\alpha|000\rangle + \alpha|011\rangle + \beta|100\rangle + \beta|111\rangle \Big) \\
&= \frac{1}{\sqrt{2}}\Big(\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle \Big)
\end{aligned}$$

for $|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

Teleportation



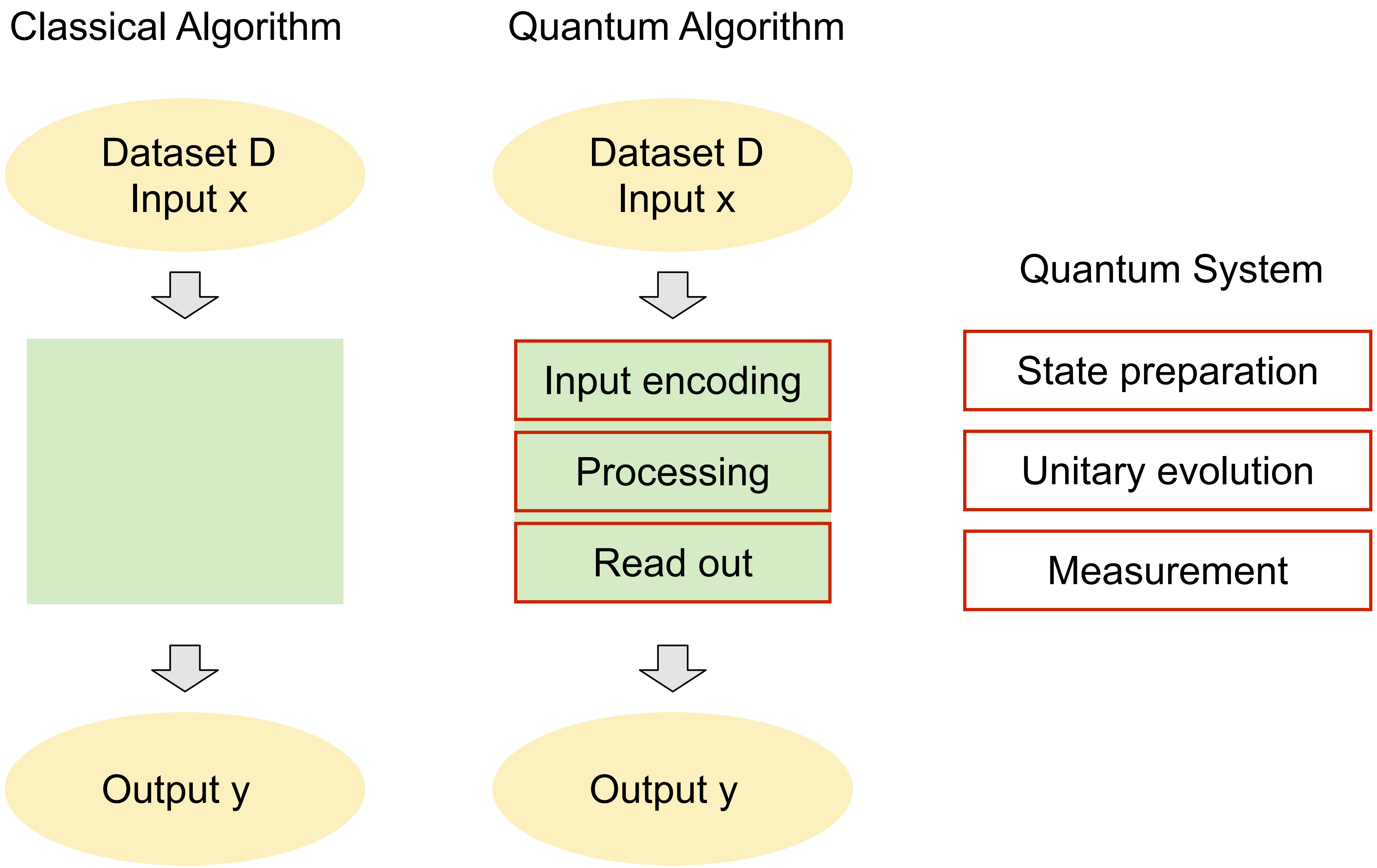
Initial state:
$|\psi\rangle = \alpha|0\rangle + \beta|1\rangle$

$$|\psi_3\rangle = \frac{1}{\sqrt{2}}\Big(\alpha|000\rangle + \alpha|011\rangle + \beta|110\rangle + \beta|101\rangle\Big)$$

$$|\psi_4\rangle = H_1|\psi_3\rangle = \frac{1}{2}\Big[\alpha\big(|000\rangle + |100\rangle\big) + \alpha\big(|011\rangle + |111\rangle\big) + \beta\big(|010\rangle - |110\rangle\big) + \beta\big(|001\rangle - |101\rangle\big)\Big]$$

qubit1	qubit2	qubit3	correction step	final state
0	0	$\alpha|0\rangle + \beta|1\rangle$	I	$\alpha|0\rangle + \beta|1\rangle$
0	1	$\beta|0\rangle + \alpha|1\rangle$	X	$\alpha|0\rangle + \beta|1\rangle$
1	0	$\alpha|0\rangle - \beta|1\rangle$	Z	$\alpha|0\rangle + \beta|1\rangle$
1	1	$-\beta|0\rangle + \alpha|1\rangle$	ZX	$\alpha|0\rangle + \beta|1\rangle$

Quantum Algorithms and Data Embedding



Quantum Algorithms and Data Embedding

	Classical data	Requirement	Quantum state
Basis Encoding	$\vec{x} \in \{0,1\}^{\otimes n}$ $\vec{x} = (x_1, x_2, \cdots, x_n) \in \{0,1\}$		$\lvert\psi\rangle = \lvert x_1 x_2 \cdots x_n\rangle$ $= \lvert x_1\rangle \otimes \lvert x_2\rangle \otimes \cdots \otimes \lvert x_n\rangle$
Amplitude Encoding	$\vec{x} \in \mathbb{R}^{2^n}$ $x_i \in \mathbb{R}$	$\sum_{i=1}^{2^n} \lvert x_i\rvert^2 = 1$	$\lvert\psi_x\rangle = \sum_{i=1}^{2^n} x_i \lvert i\rangle$
	$A \in \mathbb{R}^{2^n \times 2^m}$ $\quad i = 1,\cdots,2^n$ $A_{ij} \in \mathbb{R}$ $\quad j = 1,\cdots,2^m$	$\sum_{i,j} \lvert A_{ij}\rvert^2 = 1$	$\lvert\psi_A\rangle = \sum_{i,j} A_{ij} \lvert i\rangle \otimes \lvert j\rangle$
	$A \in \mathbb{R}^{2^n \times 2^n}$	$\sum_i A_{ii} = 1$ $\quad A^\dagger = A$ $A_{ij}^* = A_{ji}$	$\rho_A = \sum_{i,j} A_{ij} \lvert i\rangle\langle j\rvert$
Time-evolution Encoding	$x \in \mathbb{R}$	$x \in [0, 2\pi)$	$U(x) = e^{-ixH}$
Hamiltonian Encoding	$A \in \mathbb{R}^{2^n \times 2^n}$	$A^\dagger = A$	$H_A = A$
	$A \in \mathbb{R}^{2^n \times 2^n}$	$A^\dagger \neq A$ (in general)	$H_A = \begin{pmatrix} 0 & A \\ A^\dagger & 0 \end{pmatrix}$

Quantum versions of classical algorithms

- Any quantum computation is reversible prior to measurement. In contrast, classical computations are NOT in general reversible.
  - (ex) classical NOT operation is reversible while AND, OR NAND are not
  - Every classical computation does have a classical reversible analog (which takes slightly more computational resources)
  - The construction of efficient classical reversible versions of arbitrary Boolean circuits easily generalizes to construction of quantum circuits (that implement general classical circuits)
- Any classical reversible computation with n-input and n-ouput simply permutes $N = 2^n$ bit strings

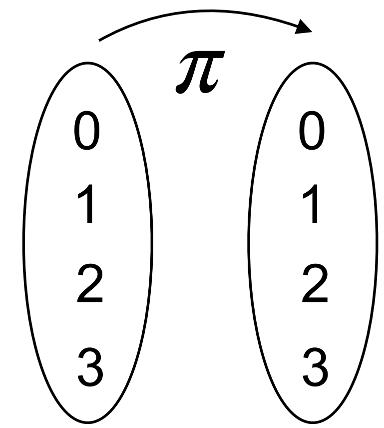
Classical computation: $\pi : \; Z_N \longrightarrow Z_N$

Quantum computation: $U_\pi : \; \sum_{x=0}^{N-1} a_x |x\rangle \longrightarrow \sum_{x=0}^{N-1} a_x |\pi(x)\rangle$

Quantum versions of classical algorithms

$n = 2,\ N = 2^2 = 4$

$$|0\rangle = |00\rangle$$
$$|1\rangle = |01\rangle$$
$$|2\rangle = |10\rangle$$
$$|3\rangle = |11\rangle$$



- Any classical computation n-inputs and m-outputs defines

$$\begin{array}{cccc} f: & Z_N & \longrightarrow & Z_M \\ & x & \longrightarrow & f(x) \end{array} \qquad N = 2^n \qquad M = 2^m$$

→ can be extended to a reversible function π_f acting on n+m bits

$$\begin{array}{cccc} \pi_f: & Z_L & \longrightarrow & Z_L \\ & (x, y) & \longrightarrow & (x, y \oplus f(x)) \end{array} \qquad L = 2^{n+m}$$

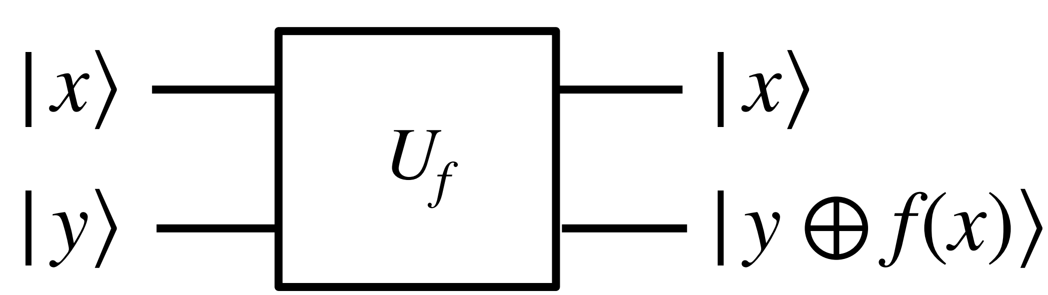
$\oplus$ = bitwise exclusive OR

x = n-bit string $\quad$ y = m-bit string $\quad$ L = n+m-bit string $\quad$ $f(x)$ = m-bit string

- For y=0, π acts like f: $(x,0) \longrightarrow (x, f(x))$

- π_f is reversible, there is a corresponding unitary transformation

$$U_f\Big(|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$



Quantum versions of simple classical gates

Let $b_0, b_1 \in \{0,1\}$ (binary variables)

NOT

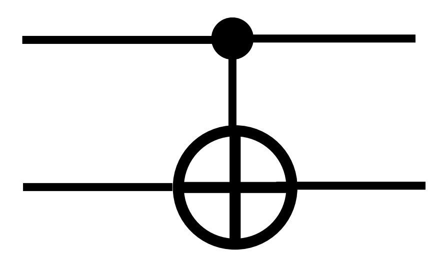
already reversible

$$X = |0\rangle\langle 1| + |1\rangle\langle 0|$$

XOR

b_1	b_0	b_1 XOR b_0
0	0	0
0	1	1
1	0	0
1	1	1

$|b_1\rangle$ $\qquad$ $|b_1\rangle$



$|b_1\rangle$ $\qquad$ $|b_1 \oplus b_0\rangle$

$$|00\rangle \longrightarrow |0\ 0 \oplus 0\rangle = |00\rangle$$
$$|01\rangle \longrightarrow |0\ 1 \oplus 0\rangle = |01\rangle$$
$$|10\rangle \longrightarrow |1\ 1 \oplus 0\rangle = |11\rangle$$
$$|11\rangle \longrightarrow |1\ 1 \oplus 1\rangle = |10\rangle$$

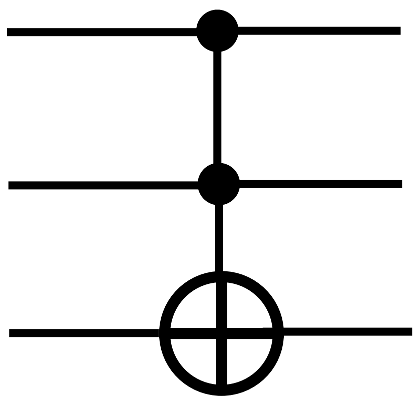
AND

b_1	b_0	b_1 AND b_0
0	0	0
0	1	0
1	0	0
1	1	1

Impossible to perform a reversible AND operation with two bits.

Quantum versions of simple classical gates

- Toffoli gate = T = CCX = CCNOT = Controlled-controlled NOT gate



$$
\begin{array}{l}
|000\rangle \longrightarrow |000\rangle \\
|001\rangle \\
|010\rangle \\
|011\rangle \\
\hline
|100\rangle \\
|101\rangle \\
\hline
|110\rangle \longrightarrow |111\rangle \\
|111\rangle \qquad\quad |110\rangle
\end{array}
$$

$$
T = \left(\begin{array}{cccc|cccc}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0
\end{array}\right) = \left(\begin{array}{c|c} I & 0 \\ \hline 0 & \mathrm{CNOT} \end{array}\right)
$$

$$T|b_1\, b_0\, 0\rangle = |b_1\, b_0\, b_1 \wedge b_0\rangle$$

$$T|b_1\, b_0\, 1\rangle = |b_1\, b_0\, 1 \oplus b_1 \wedge b_0\rangle$$

$\wedge$ = classical AND $\qquad \sim$ = NOT

- Toffoli gate T can be used to construct a complete set of Boolean connectives (NOT, AND, XOR, NAND)

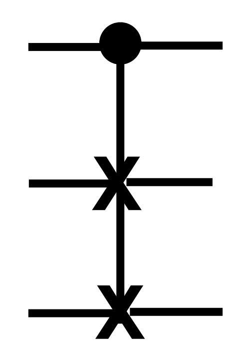
$$T|1\,1\,x\rangle = |1\,1\ \sim x\rangle$$

$$T|x\,y\,0\rangle = |x\,y\,x \wedge y\rangle$$

$$T|1\,x\,y\rangle = |1\,x\,x \oplus y\rangle$$

$$T|x\,y\,1\rangle = |x\,y\ \sim (x \wedge y)\rangle$$

- Alternative: Fredkin gate F=controlled SWAP

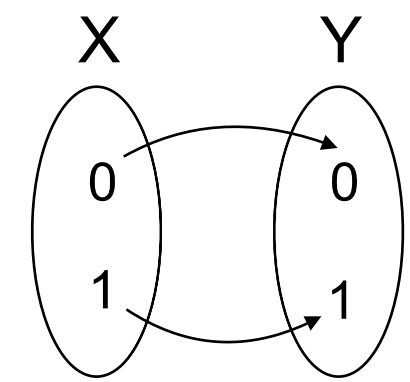


$$F|x\,0\,1\rangle = |x\,x\ \sim x\rangle$$

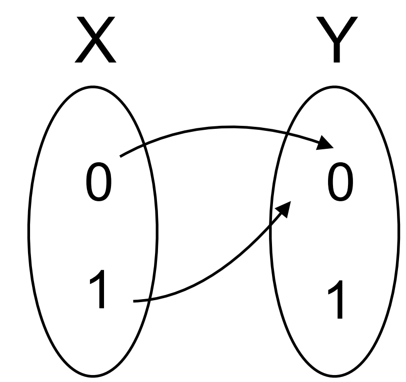
$$F|x\,y\,1\rangle = |x\,(y \vee x)\,y \vee (\sim x)\rangle$$

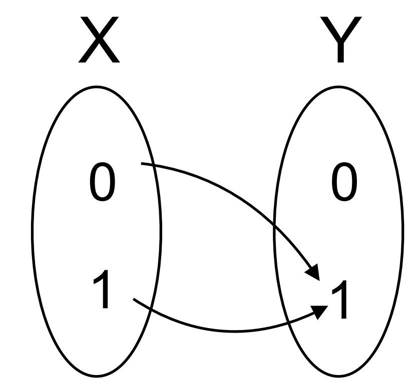
$$F|x\,0\,y\rangle = |x\,(y \wedge x)\,y \wedge (\sim x)\rangle$$

A simple QA with two qubits



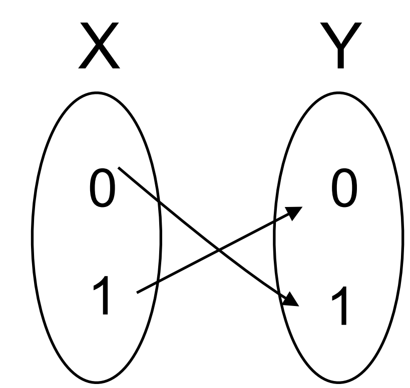
- Consider a simple function, $f(x) : \{0,1\} \longrightarrow \{0,1\}$
  one-bit domain   one-bit range
- For possible functions
  - Identity: $f(0) = 0$ and $f(1) = 1$
  - Bit-flip function: $f(0) = 1$ and $f(1) = 0$
  - Constant function: $f(x) = 0$ or $f(x) = 1$





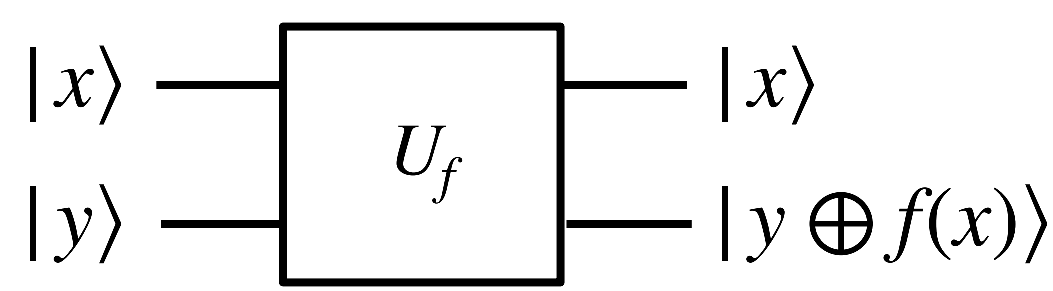
- Reconstruct a unitary transformation U_f such that $(x, y) \xrightarrow[U_f]{} (x, y \oplus f(x))$, which corresponds to

$$U_f\Big(\,|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$



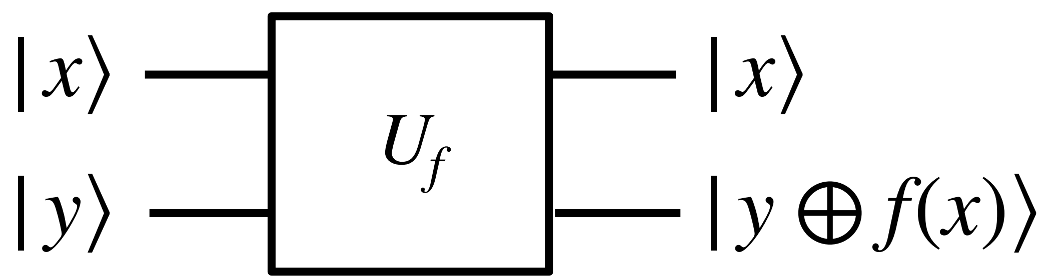
- $\oplus$ is mode 2 addition: $0 \oplus 0 = 0 = 1 \oplus 1$ and $0 \oplus 1 = 1 = 0 \oplus 1$.
- $x \longrightarrow f(x)$ is not suitable because $f(x)$ is not unitary in general.
- $(x, y) \xrightarrow{U_f} (x, y \oplus f(x)) \xrightarrow{U_f} (x, y \oplus f(x) \oplus f(x)) = (x, y)$

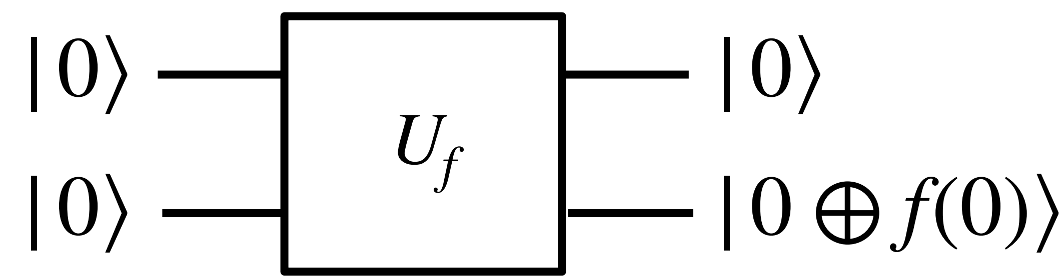
$$U_f\Big(\,|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$



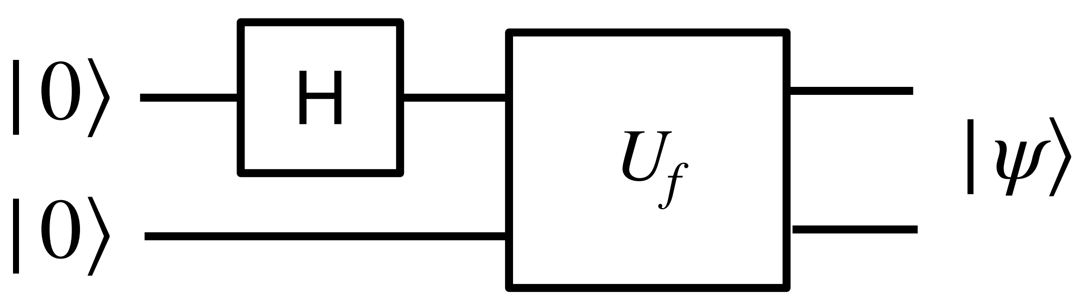
A simple QA with two qubits

- Take advantage of "quantum parallelism" (a qubit can have both $|0\rangle$ and $|1\rangle$)





- Apply Hadamard gate to the first qubit and then apply U.



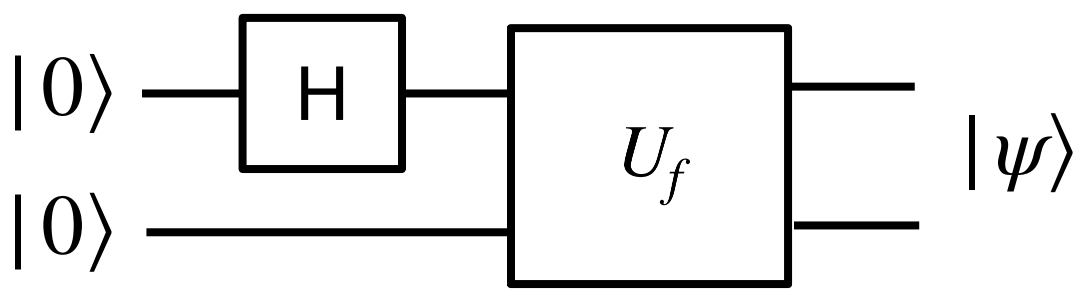
$$H|0\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big)$$

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

$$|\psi\rangle = \frac{1}{\sqrt{2}}U_f\Big(|0\rangle \otimes |f(0)\rangle + |1\rangle \otimes |f(1)\rangle\Big) = \sum_{x=0,1} \frac{1}{\sqrt{2}}|x\rangle \otimes |f(x)\rangle$$

A simple QA with two qubits

- $|\psi\rangle$ contains information on both f(0) and f(1)
  - Superposition of f(0) and f(1)
  - Need to perform measurement to access the info
  - However, measurement returns only one value of x and f(x)



$$|\psi\rangle = \frac{1}{\sqrt{2}} U_f \Big(|0\rangle \otimes |f(0)\rangle + |1\rangle \otimes |f(1)\rangle \Big) = \sum_{x=0,1} \frac{1}{\sqrt{2}} |x\rangle \otimes |f(x)\rangle$$

LAUREATES

Breakthrough Prize   Special Breakthrough Prize   New Horizons Prize   Physics Frontiers Prize

2023 2022 2021 2020 2019 2018 2017 2016 2015 2014 2013 2012


Charles H. Bennett

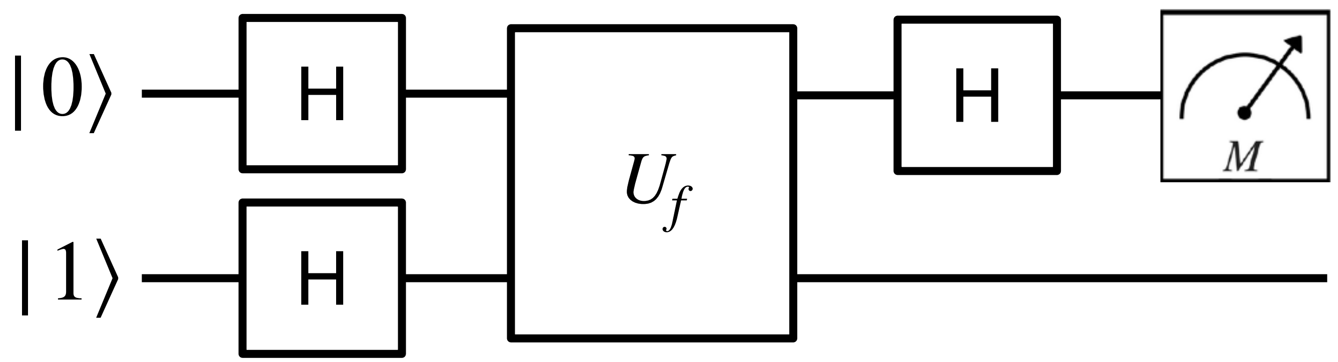

Gilles Brassard


David Deutsch


Peter W. Shor

Deutsch Algorithm

- Deutsch algorithm exploits QA to obtain information about global property of f(x).
- A function of a single qubit can be either constant $f(0) = f(1)$ or balanced $f(0) \neq f(1)$



(1) (2) (3)

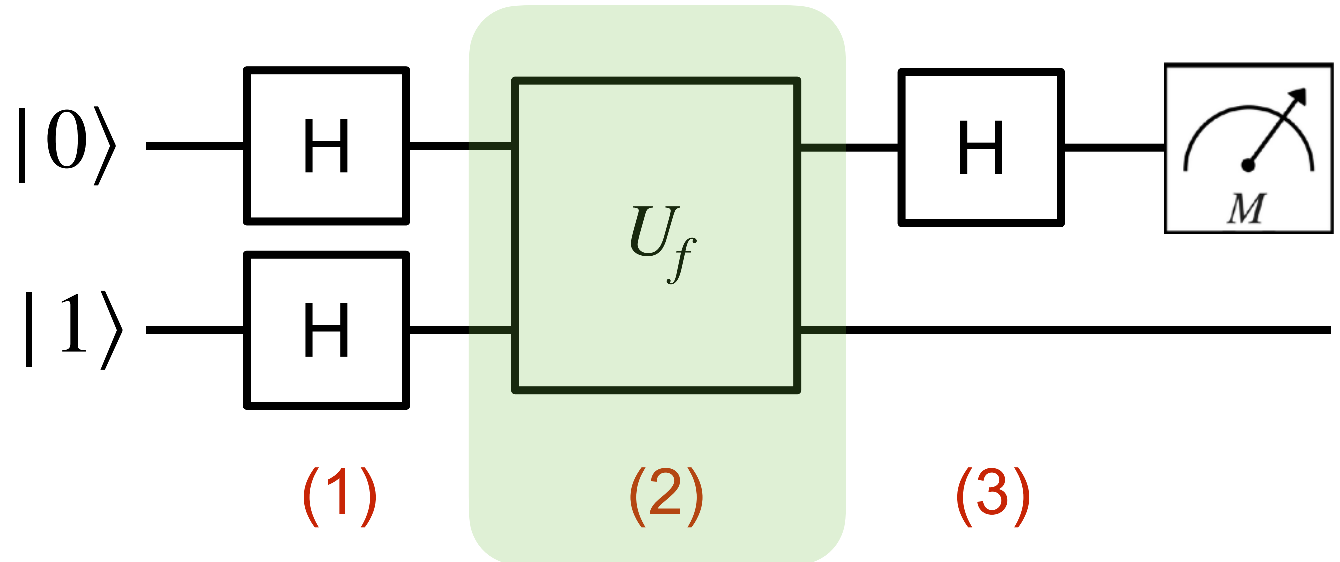
$$|\psi_0\rangle \xrightarrow{H \otimes H} |\psi_1\rangle \xrightarrow{U_f} |\psi_2\rangle \xrightarrow{H \otimes I} |\psi_3\rangle$$

$$|\psi_0\rangle \equiv |0\rangle \otimes |1\rangle = |01\rangle$$

(1) $$|\psi_1\rangle = H \otimes H |01\rangle = \frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big) \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)$$

$$= \frac{1}{2}\Big(|00\rangle - |01\rangle + |10\rangle - |11\rangle\Big) = \frac{1}{2}\Big(\sum_x |x\rangle\Big) \otimes \Big(|0\rangle - |1\rangle\Big)$$

Deutsch Algorithm



$$|\psi_0\rangle \equiv |0\rangle \otimes |1\rangle = |01\rangle$$

$$|\psi_1\rangle = \frac{1}{2}\Big(\sum_x |x\rangle\Big) \otimes \Big(|0\rangle - |1\rangle\Big)$$

$$U_f\Big(|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$

(2) $|\psi_2\rangle = U_f|\psi_1\rangle$

For $f(x) = 0$: $$U_f\Big[|x\rangle \otimes (|0\rangle - |1\rangle)\Big] = U_f\Big(|x\rangle \otimes |0\rangle\Big) - U_f\Big(|x\rangle \otimes |1\rangle\Big)$$

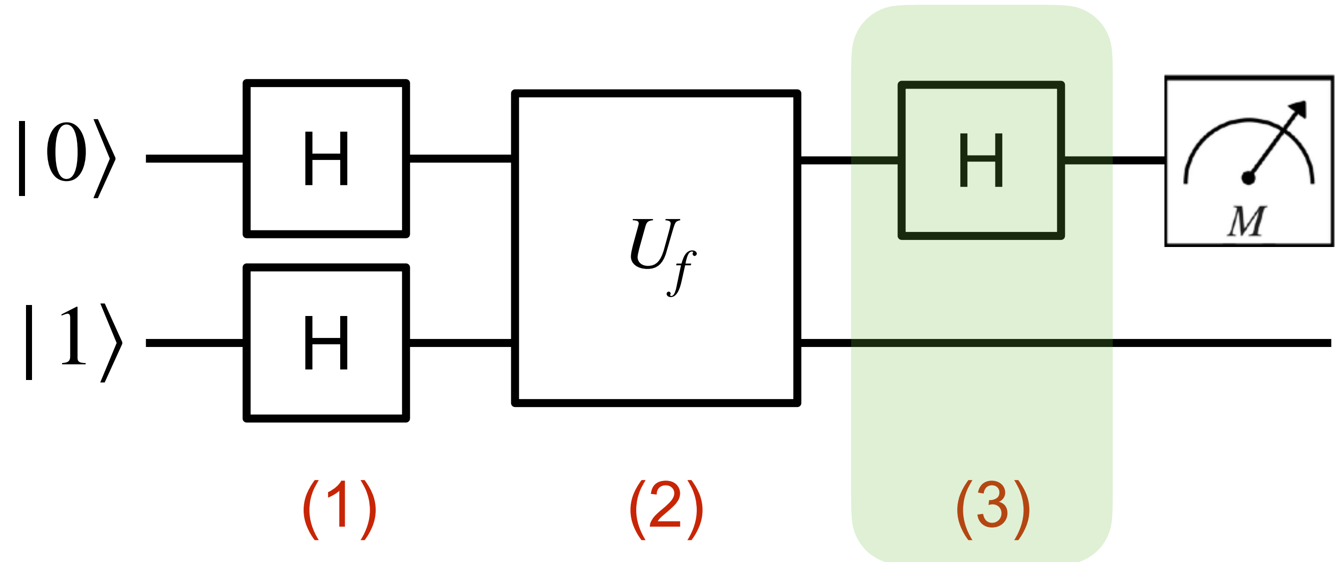
$$= |x\rangle \otimes |0 + f(x)\rangle - |x\rangle \otimes |1 + f(x)\rangle$$

$$= |x\rangle \otimes \Big(|0\rangle - |1\rangle\Big) = (-1)^{f(x)}|x\rangle \otimes \Big(|0\rangle - |1\rangle\Big)$$

For $f(x) = 1$: $$U_f\Big[|x\rangle \otimes (|0\rangle - |1\rangle)\Big] = |x\rangle \otimes \Big(|1\rangle - |0\rangle\Big) = (-1)^{f(x)}|x\rangle \otimes \Big(|0\rangle - |1\rangle\Big)$$

$$|\psi_2\rangle = U_f|\psi_1\rangle = \frac{1}{\sqrt{2}}\Big[\sum_x (-1)^{f(x)}|x\rangle\Big] \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)$$

Deutsch Algorithm



$$|\psi_0\rangle \equiv |0\rangle \otimes |1\rangle = |01\rangle$$

$$|\psi_1\rangle = \frac{1}{2}\Big(\sum_x |x\rangle\Big) \otimes \Big(|0\rangle - |1\rangle\Big)$$

$$U_f\Big(|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$

(3) $$|\psi_3\rangle = \big(H \otimes I\big)|\psi_2\rangle = \big(H \otimes I\big)\frac{1}{\sqrt{2}}\Big[\sum_x (-1)^{f(x)}|x\rangle\Big] \otimes \frac{1}{\sqrt{2}}\Big(|0\rangle - |1\rangle\Big)$$

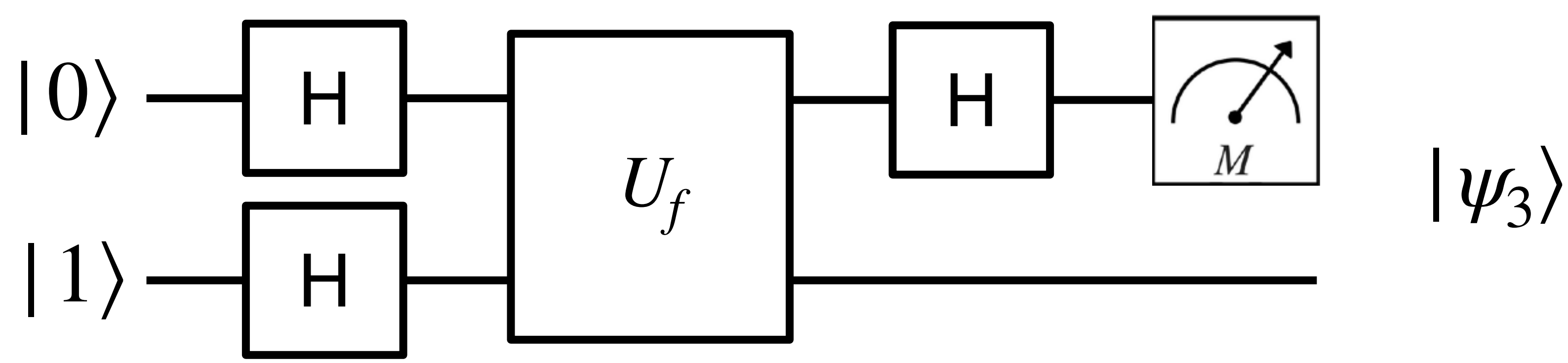
$$H\frac{1}{\sqrt{2}}\Big[\sum_x (-1)^{f(x)}|x\rangle\Big] = \frac{1}{\sqrt{2}}H\Big[(-1)^{f(0)}|0\rangle + (-1)^{f(1)}|1\rangle\Big]$$

$$= \frac{1}{\sqrt{2}}\Big[(-1)^{f(0)}\frac{|0\rangle + |1\rangle}{\sqrt{2}} + (-1)^{f(1)}\frac{|0\rangle - |1\rangle}{\sqrt{2}}\Big]$$

$$= \frac{1}{2}\Big[\Big((-1)^{f(0)} + (-1)^{f(1)}\Big)|0\rangle + \Big((-1)^{f(0)} - (-1)^{f(1)}\Big)|1\rangle\Big]$$

Deutsch Algorithm

- Deutsch algorithm exploits QA to obtain information about global property of f(x).
- A function of a single qubit can be either constant $f(0) = f(1)$ or balanced $f(0) \neq f(1)$



$$|\psi_0\rangle \xrightarrow{H \otimes H} |\psi_1\rangle \xrightarrow{U_f} |\psi_2\rangle \xrightarrow{H \otimes I} |\psi_3\rangle$$

$$|\psi_3\rangle = \frac{1}{2}\left[\left((-1)^{f(0)} + (-1)^{f(1)}\right)|0\rangle + \left((-1)^{f(0)} - (-1)^{f(1)}\right)|1\rangle\right]$$

- If measurement gives $|0\rangle$, $f(0) = f(1) \longrightarrow f(x) = \text{constant}.$
- If measurement gives $|1\rangle$, $f(0) \neq f(1) \longrightarrow f(x) = \text{balanced}.$
- Can be generalized to function with multiple input values, Deutsch-Josza algorithm

Basic operations with bit strings

- x and y are two n-bit strings:

$$|x\rangle = |x_{n-1}\, x_{n-2}\, \cdots\, x_1\, x_0\rangle$$

$$|y\rangle = |y_{n-1}\, y_{n-2}\, \cdots\, y_1\, y_0\rangle$$

$$x_i\,, y_i \in \{0,1\}$$

- Hamming distance = $d_H(x, y)$ = the number of bits in which the two strings differ.

$$|x\rangle = |\,10101\,\rangle$$

$$|y\rangle = |\,11100\,\rangle$$

$$d_H(x, y) = \,?$$

- Hamming weight = $d_H(x)$ = $d_H(x,0)$ = the number of 1-bit in x = the Hamming distance between x and 0.

- $x \cdot y$ = the number of common 1-bit in x and y = $d_H(x, y)$

- $x \oplus y$ = the bitwise exclusive OR = bitwise addition under mod 2

- $x \wedge y$ = the bitwise AND

- $\sim x$ = $x \oplus 111\cdots1$ = the bit string that flips 0 and 1

Useful Identities

- $x \cdot y = d_H(x, y)$
- $x \cdot y = \frac{1}{2}\left(1 - (-1)^{x \cdot y}\right) \bmod 2$
- $x \cdot y + x \cdot z = x \cdot (y \oplus z) \bmod 2$
- $d_H(x \oplus y) = d_H(x) + d_H(y) \bmod 2$
- $\sum_{x=0}^{2^n-1} (-1)^{x \cdot x} = 0$ b/c successive $2i$ and $2i+1$ terms cancel
- $\sum_{x=0}^{2^n-1} (-1)^{x \cdot y} = \begin{cases} 2^n, & \text{if y = 0} \\ 0\ , & \text{otherwise} \end{cases}$

Walsh-Hadamard Transformation

$$W \equiv H \otimes H \otimes \cdots \otimes H \equiv H^{\otimes n}$$ apply H to each qubit in an n-qubit system

$$W|0\rangle = \frac{1}{\sqrt{N}} \sum_{x=0}^{N-1} |x\rangle \qquad N = 2^n$$

$$|r\rangle = |r_{n-1}\, r_{n-2} \,\cdots\, r_1\, r_0\rangle$$
$$|s\rangle = |s_{n-1}\, s_{n-2} \,\cdots\, s_1\, s_0\rangle \qquad r_i, s_i \in \{0,1\}$$

- How does W act on $|r\rangle$? $\qquad W|r\rangle = \sum_s W_{rs} |s\rangle$

$$W|r\rangle = \big(H \otimes H \otimes \cdots \otimes H\big) |r_{n-1}\, r_{n-2} \,\cdots\, r_1\, r_0\rangle$$

$$= \frac{1}{\sqrt{2}^n} \Big[\underbrace{|0\rangle + (-1)^{r_{n-1}} |1\rangle}_{= \sum_{s_{n-1}=0}^{1} (-1)^{-s_{n-1}\cdot r_{n-1}} |s_{n-1}\rangle} \Big] \otimes \cdots \otimes \Big[\underbrace{|0\rangle + (-1)^{r_{n-1}} |1\rangle}_{= \sum_{s_0=0}^{1} (-1)^{-s_0\cdot r_0} |s_0\rangle} \Big]$$

$$= \frac{1}{2^n} \sum_{s=0}^{N-1} (-1)^{-s_{n-1}\cdot r_{n-1}} |s_{n-1}\rangle \otimes \cdots \otimes (-1)^{-s_1\cdot r_1} |s_1\rangle \otimes (-1)^{-s_0\cdot r_0} |s_0\rangle$$

$$W(|r\rangle) = \frac{1}{2^n} \sum_{s=0}^{2^n-1} (-1)^{s\cdot r} |r\rangle \qquad W_{rs} = W_{sr} = \frac{1}{\sqrt{2}^n} (-1)^{r\cdot s}$$

Deutsch-Jozsa Algorithm

- Given a function $f: Z_{2^n} \longrightarrow Z_2$ that is known to be either constant or balanced, and $U_f: |x\rangle \otimes |y\rangle \longrightarrow |x\rangle \otimes |x \oplus f(x)\rangle$, determine whether the function f is constant or balanced.

- Phase change for a subset of basis vectors

$$\text{Consider a superposition : } |\psi\rangle = \sum_i a_i |i\rangle$$

$$\text{Boolean function : } f: Z_{2^n} \longrightarrow Z_2 \text{ where } f(x) = \begin{cases} 1, & \text{if } x \in X \subset Z_{2^n} \\ 0, & \text{otherwise} \end{cases}$$

$$S_X^{\phi}: \sum_{x=0}^{N-1} a_x |x\rangle \longrightarrow \sum_{x \in X} a_x e^{i\phi} |x\rangle + \sum_{x \notin X} a_x |x\rangle \qquad \text{where } X = \{x \,|\, f(x) = 0\}$$

For $\phi = \pi$

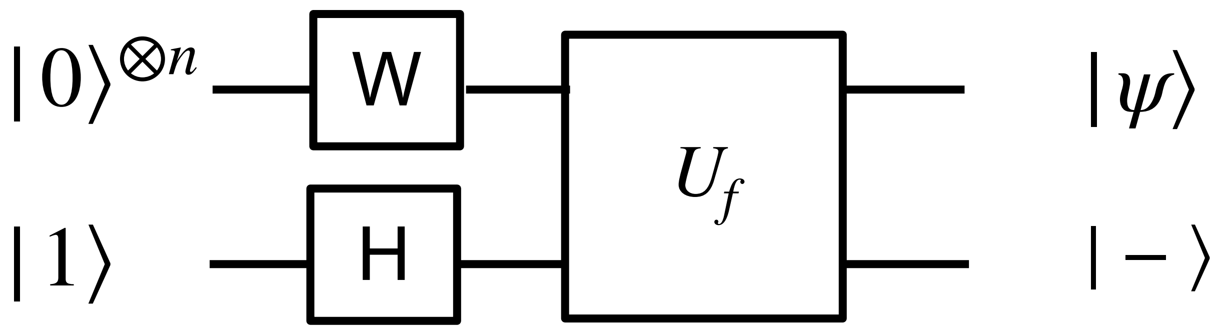
$$U_f\Big(|\psi\rangle \otimes |-\rangle\Big) = U_f\Big(\sum_{x \in X} a_x e^{i\phi} |x\rangle \otimes |-\rangle\Big) + U_f\Big(\sum_{x \notin X} a_x |x\rangle \otimes |-\rangle\Big)$$

$$= -\Big(\sum_{x \in X} a_x |x\rangle \otimes |-\rangle\Big) + \Big(\sum_{x \notin X} a_x |x\rangle \otimes |-\rangle\Big)$$

$$(-1)^{f(x)}$$

$$= \sum_x (-1)^{f(x)} |\psi\rangle \otimes |-\rangle$$

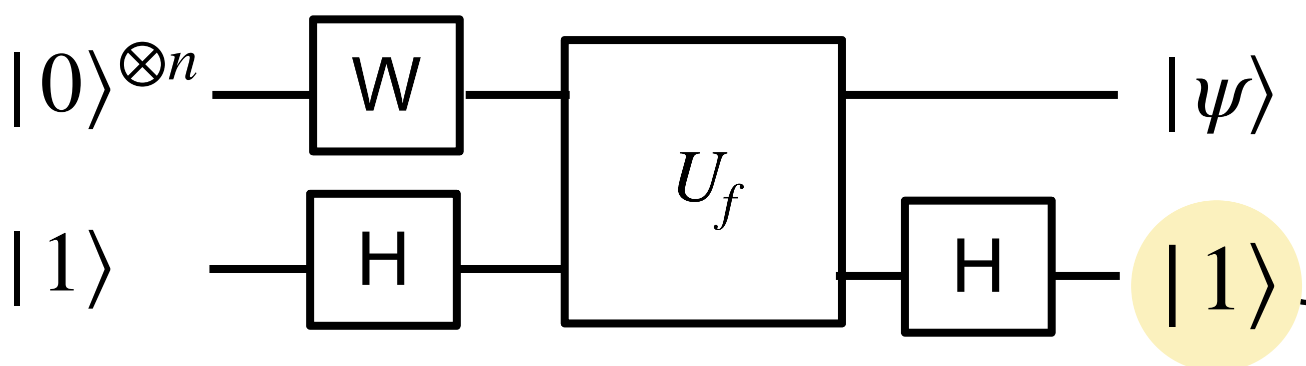
Deutsch-Jozsa Algorithm



n = number of qubits

$N = 2^n$ = dim of Hilbert space

$$|\psi_0\rangle = W|0\rangle = \frac{1}{\sqrt{N}}\sum_{x=0}^{N-1}|x\rangle$$



$$|\psi\rangle = \sum_x (-1)^{f(x)}|\psi_0\rangle$$

Can reuse the ancilla qubit

$$|\psi\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}(-1)^{f(i)}|\,i\,\rangle$$

$$|\phi\rangle = W|\psi\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}(-1)^{f(i)}W|\,i\,\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}(-1)^{f(i)}\sum_{j=0}^{N-1}\frac{1}{\sqrt{N}}(-1)^{i\cdot j}|\,j\,\rangle$$

For constant f, $(-1)^{f(i)} = (-1)^{f(0)}$ is a global phase.

$$|\phi\rangle = (-1)^{f(0)}\frac{1}{N}\sum_i\Bigg(\underbrace{\sum_i (-1)^{i\cdot j}}\Bigg)|j\rangle = (-1)^{f(0)}|0\rangle$$

only nonzero when $j = 0$

- $\displaystyle\sum_{x=0}^{2^n-1}(-1)^{x\cdot y} = \begin{cases} 2^n, & \text{if y = 0} \\ 0\ , & \text{otherwise} \end{cases}$

Deutsch-Jozsa Algorithm

$$|\phi\rangle = W|\psi\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}(-1)^{f(i)}\,W\,|\,i\,\rangle = \frac{1}{\sqrt{N}}\sum_{i=1}^{N-1}(-1)^{f(i)}\sum_{j=0}^{N-1}\frac{1}{\sqrt{N}}(-1)^{i\cdot j}\,|\,j\,\rangle$$

For balanced f, $\quad |\phi\rangle = \frac{1}{2^n}\sum_{j}\left(\sum_{i\in X}(-1)^{i\cdot j} - \sum_{i\notin X}(-1)^{i\cdot j}\right)|j\rangle \qquad$ where $X = \{x\,|\,f(x)=0\}$

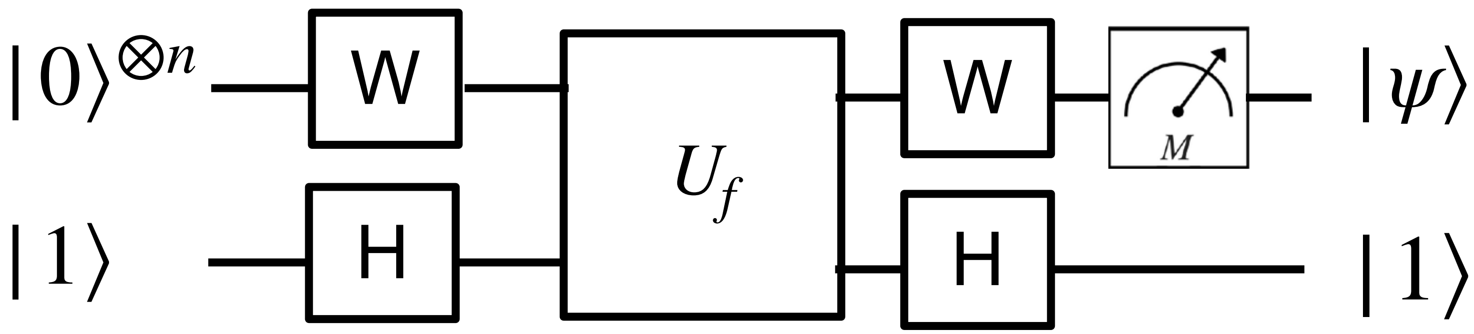
For $j=0$, amplitude is zero.

$\sum_{i\in X}(-1)^{i\cdot j} - \sum_{i\notin X}(-1)^{i\cdot j} = 0$ for $j=0$ $\quad\Longleftrightarrow\quad$ $|\phi\rangle$ does not contain $|0\rangle$.

$\Rightarrow$

- Measurement of state $|\phi\rangle$ (in the standard basis) will return $|0\rangle$ with probability 1, if f is constant, and will return a non-zero $|j\rangle$ with probability 1, if f is balanced.
- Classical algorithm must evaluate f at least $2^{n-1}+1$ times to solve the problem with certainty, while quantum algorithm solves with a single evaluation of U_f.
- There is an exponential separation between the query complexity of the QA and query complexity of any classical algorithm.
- There are classical algorithms that solve the problem in fewer evaluations but only with high probability of success (not 100% probability).

Deutsch-Jozsa Algorithm



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

$$|-\rangle = \frac{1}{\sqrt{2}}\big(|0\rangle - |1\rangle\big)$$

$$H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix} = \frac{1}{\sqrt{2}} \sum_{x,y\in\{0,1\}} (-1)^{xy}\,|y\rangle\langle x| \qquad H^2 = I$$

$$W \equiv H^{\otimes n} = \left(\frac{1}{\sqrt{2}} \sum_{x,y\in\{0,1\}} (-1)^{xy}\,|y\rangle\langle x| \right)^{\otimes n}$$

$$= \left(\frac{1}{\sqrt{2}} \sum_{x_0,y_0} (-1)^{x_0 y_0}\,|y_0\rangle\langle x_0| \right) \otimes \cdots \otimes \left(\frac{1}{\sqrt{2}} \sum_{x_0,y_0} (-1)^{x_{n-1},y_{n-1}}\,|y_{n-1}\rangle\langle x_{n-1}| \right)$$

$$= \frac{1}{\sqrt{2}^n} \sum_{x,y\in\{0,1\}^{\otimes n}} (-1)^{x\cdot y}\,|y\rangle\langle x| \qquad x\cdot y = x_o y_0 + x_1 y_1 + \cdots + x_{n-1}y_{n-1}$$

$$H^{\otimes n}\frac{1}{\sqrt{2}^n}\sum_x |x\rangle = 0 \qquad H^{\otimes n}|0\rangle = \frac{1}{\sqrt{2}^n}\sum_x |x\rangle$$

Bernstein-Vazirani Algorithm

- A n-bit function $f: \{0,1\}^{\otimes n} \longrightarrow \{0,1\}$, which outputs a singlet bit, is guaranteed to be of the form $f_s(x) = x \cdot s$, where s is an unknown n-bit string and $x \cdot s = x_0 s_0 + \cdots + x_{n-1} s_{n-1} = \sum_{i=0}^{n-1} x_i s_i \pmod 2$. Find the unknown string $s = (s_0 s_1 \cdots s_{n-1})$.
- Best classical algorithm uses $\mathcal{O}(n)$ calls to $f_s(x) = x \cdot s \mod 2$. Each query gives one bit of information of s (because f outputs one bit).

- How do we find s with less than n queries? → Use superposition (over all possible input bit strings)

1	3	5	7	9	11	13	15
17	19	21	23	25	27	29	31
33	35	37	39	41	43	45	47
49	51	53	55	57	59	61	63

2	3	6	7	10	11	14	15
18	19	22	23	26	27	30	31
34	35	38	39	42	43	46	47
50	51	54	55	58	59	62	63

4	5	6	7	12	13	14	15
20	21	22	23	28	29	30	31
36	37	38	39	44	45	46	47
52	53	54	55	60	61	62	63

8	9	10	11	12	13	14	15
24	25	26	27	28	29	30	31
40	41	42	43	44	45	46	47
56	57	58	59	60	61	62	63

16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

1	3	5	7	9	11	13	15
17	19	21	23	25	27	29	31
33	35	37	39	41	43	45	47
49	51	53	55	57	59	61	63

2	3	6	7	10	11	14	15
18	19	22	23	26	27	30	31
34	35	38	39	42	43	46	47
50	51	54	55	58	59	62	63

4	5	6	7	12	13	14	15
20	21	22	23	28	29	30	31
36	37	38	39	44	45	46	47
52	53	54	55	60	61	62	63

8	9	10	11	12	13	14	15
24	25	26	27	28	29	30	31
40	41	42	43	44	45	46	47
56	57	58	59	60	61	62	63

16	17	18	19	20	21	22	23
24	25	26	27	28	29	30	31
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

32	33	34	35	36	37	38	39
40	41	42	43	44	45	46	47
48	49	50	51	52	53	54	55
56	57	58	59	60	61	62	63

Bernstein-Vazirani Algorithm

- A n-bit function $f: \{0,1\}^{\otimes n} \longrightarrow \{0,1\}$, which outputs a singlet bit, is guaranteed to be of the form $f_s(x) = x \cdot s$, where s is an unknown n-bit string and $x \cdot s = x_0 s_0 + \cdots + x_{n-1} s_{n-1} = \sum_{i=0}^{n-1} x_i s_i \pmod 2$. Find the unknown string $s = (s_0 s_1 \cdots s_{n-1})$.
- Best classical algorithm uses $\mathcal{O}(n)$ calls to $f_s(x) = x \cdot s \mod 2$. Each query gives one bit of information of s (because f outputs one bit).

$$U_f\Big(|x\rangle \otimes |y\rangle\Big) = |x\rangle \otimes |y \oplus f(x)\rangle$$

$$f_s(x) = x \cdot s \mod 2$$

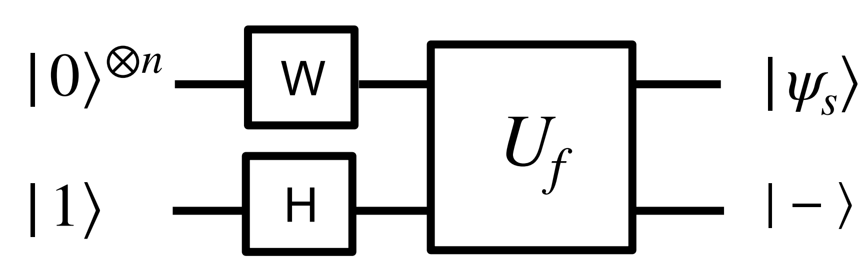
$$U_f = \sum_x \sum_y |x\rangle\langle x| \otimes |y \oplus f(x)\rangle\langle y|$$

$$U_f = \sum_{x \in \{0,1\}^{\otimes n}} \sum_{y \in \{0,1\}^{\otimes n}} |x\rangle\langle x| \otimes |y \oplus s \cdot x\rangle\langle y|$$

- How do we find s with less than n queries? → Use superposition (over all possible input bit strings)

$$|\psi_s\rangle = \frac{1}{\sqrt{2}^n} \sum_{x \in \{0,1\}^{\otimes n}} (-1)^{f(x)} |x\rangle = \frac{1}{\sqrt{2}^n} \sum_{x \in \{0,1\}^{\otimes n}} (-1)^{x \cdot s} |x\rangle$$

$$U_f\Big(|\psi\rangle \otimes |-\rangle\Big) = \sum_x (-1)^{f(x)} |\psi\rangle \otimes |-\rangle$$



Bernstein-Vazirani Algorithm

- $|\psi_s\rangle$ states are orthogonal! $\quad \langle\psi_s|\psi_t\rangle = \delta_{st}$

$$\langle\psi_s|\psi_t\rangle = \frac{1}{2^n}\sum_{x\in\{0,1\}^{\otimes n}}(-1)^{x\cdot s}\langle x|\sum_{y\in\{0,1\}^{\otimes n}}(-1)^{y\cdot t}|y\rangle = \frac{1}{2^n}\sum_{x,y}(-1)^{x\cdot s+y\cdot t}\langle x|y\rangle$$

$$= \frac{1}{2^n}\sum_{x\in\{0,1\}^{\otimes n}}(-1)^{x\cdot s+x\cdot t} = \frac{1}{2^n}\sum_{x\in\{0,1\}^{\otimes n}}(-1)^{x\cdot(s\oplus t)}$$

$$x\cdot s = x_0s_0 + \cdots + x_{n-1}s_{n-1}$$

$$x\cdot s + x\cdot t = x\cdot(s\oplus t) \pmod 2$$

$$\sum_{x\in\{0,1\}^{\otimes n}}(-1)^{x\cdot k} = \sum_{x\in\{0,1\}^{\otimes n}}(-1)^{x_0k_0+\cdots+x_{n-1}k_{n-1}} = \sum_{x_0\in\{0,1\}}(-1)^{x_0k_0}\sum_{x_1\in\{0,1\}}(-1)^{x_1k_1}\cdots\sum_{x_{n-1}\in\{0,1\}}(-1)^{x_{n-1}k_{n-1}}$$

$$= 2\delta_{k_00}\times 2\delta_{k_10}\cdots\times 2\delta_{k_{n-1}0} = 2^n\delta_{k0}$$

$$\sum_{x=0}^{2^n-1}(-1)^{x\cdot y} = \begin{cases} 2^n, & \text{if } y=0 \\ 0, & \text{otherwise}\end{cases}$$

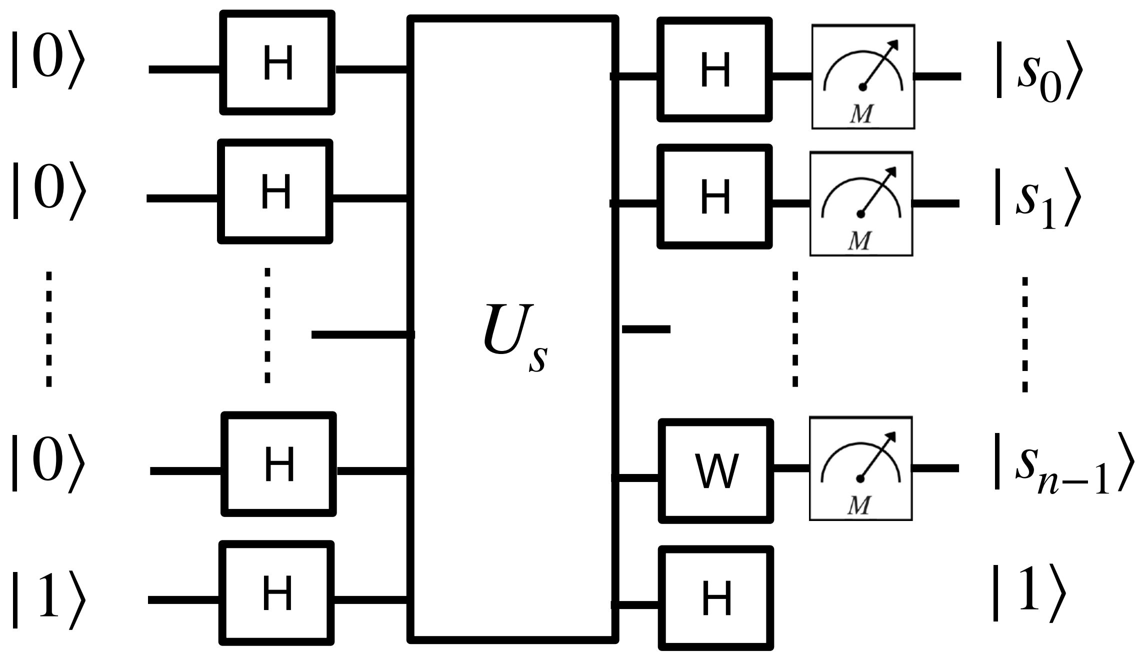
$$\langle\psi_s|\psi_t\rangle = \delta_{s\oplus t,0} = \delta_{s\,t}$$

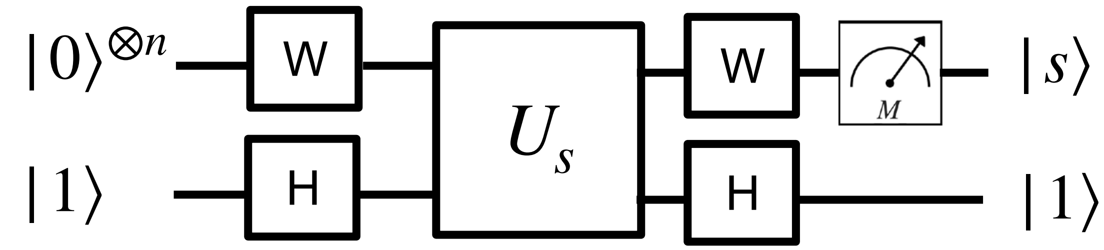
- Orthogonal set of vectors from a basis and we can “measure in this basis”.
- Equivalent to performing unitary transformation and measuring in the computational basis, from which we should be able to extract the string s.

$$W \equiv H^{\otimes n} = \frac{1}{\sqrt{2}^n}\sum_{x,y\in\{0,1\}^{\otimes n}}(-1)^{x\cdot y}|y\rangle\langle x| = \sum_{y\in\{0,1\}^{\otimes n}}|y\rangle\langle\psi_y|$$

Bernstein-Vazirani Algorithm

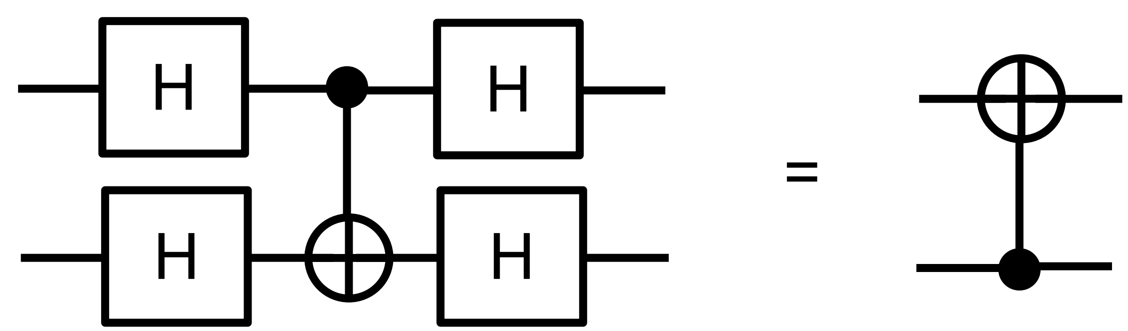
- Apply $H^{\otimes n}$ to $|\psi_s\rangle$: $$H^{\otimes n}|\psi_s\rangle = \sum_y |y\rangle\langle\psi_y|\psi_s\rangle = |s\rangle$$ in 100% probability





Circuit for Berstein-Vazirani algorithm

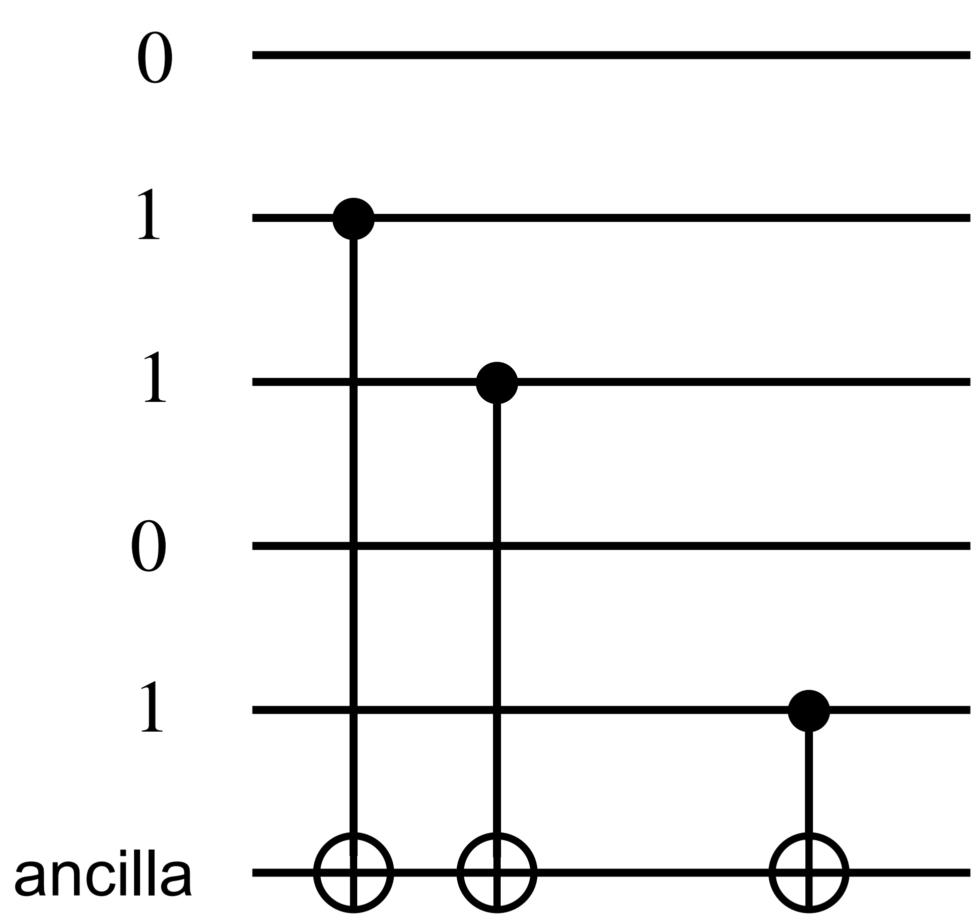
- Simpler explanation: Berstein-Vazirani algorithm behaves as if it were a circuit consisting only of CNOT operations from ancilla to the qubits corresponding to 1-bit of s.

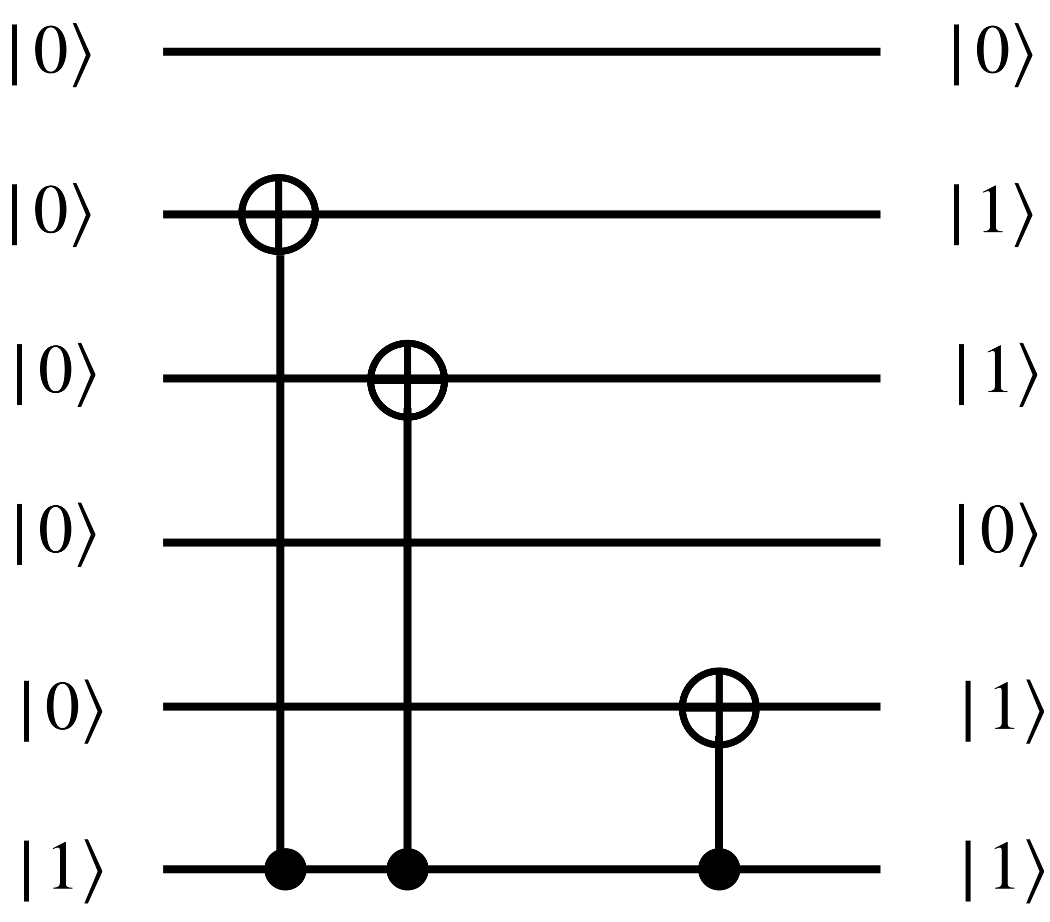


Bernstein-Vazirani Algorithm

- Berstein-Vazirani algorithm behaves as if it were a circuit consisting only of CNOT operations from ancilla to the qubits corresponding to 1-bit of s.

$s = 01101$





- For s=01101, the black box for U_s behaves as if it contained this circuit, consisting of CNOT gates for each 1-bit of s.

- BV algorithm behaves as if it were implemented by this simple circuit, consisting of a CNOT for each 1-bit of s.

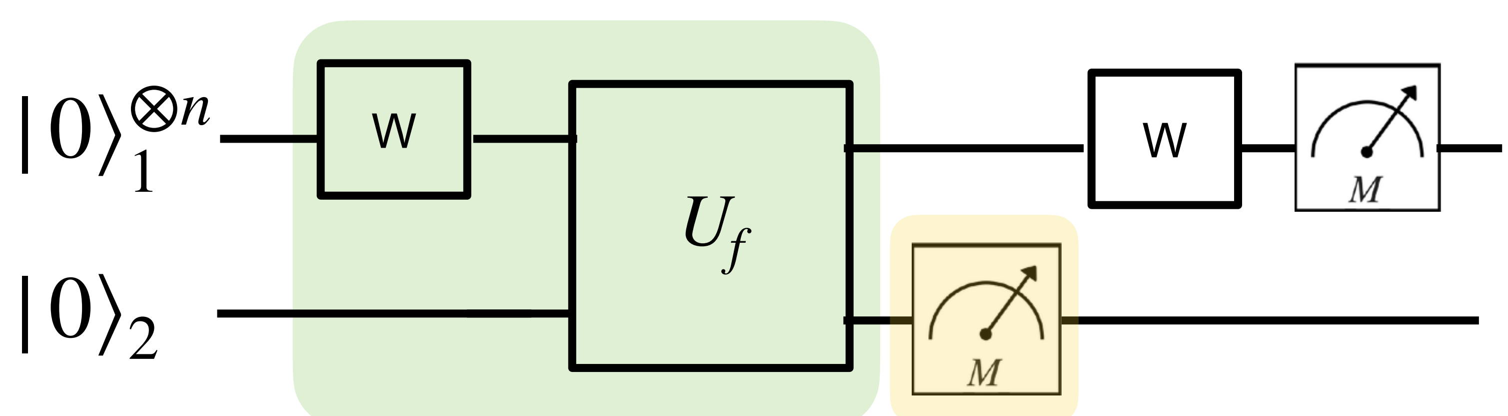
Simon's Algorithm

- Given a 2-to-1 function f such that $f(x) = f(x \oplus a)$ for all $x \in \mathbb{Z}_2^n$, find the hidden string $a \in \mathbb{Z}_2^n$. (Simon's algorithm shows structural similarities to Shor's algorithm)

$$U_f : |x\rangle \otimes |y\rangle \longrightarrow |x\rangle \otimes |y \oplus f(x)\rangle \qquad |x\rangle = |x_0 x_1 \cdots x_{n-1}\rangle$$

$$U_f \left[W|0\rangle^{\otimes n} \otimes |0\rangle \right] = U_f \frac{1}{\sqrt{N}} \sum_x |x\rangle \otimes |f(x)\rangle \qquad x_i \in \{0,1\} \qquad N = 2^n$$

- Suppose we perform a measurement on 2nd qubit and $f(x_0)$ is the measured value. Then the 1st qubit becomes $\frac{1}{\sqrt{2}}\left(|x_0\rangle + |f(x_0)\rangle \right)$.



Simon's Algorithm

- Apply Walsh-Hadamard:

$$W\left[\frac{1}{\sqrt{2}}\left(|x_0\rangle + |x_0 \oplus a\rangle\right)\right] = \frac{1}{\sqrt{2}}\left[\frac{1}{\sqrt{2}^n}\sum_y \left\{(-1)^{x_0\cdot y} + (-1)^{(x_0\oplus a)\cdot y}\right\}|y\rangle\right]$$

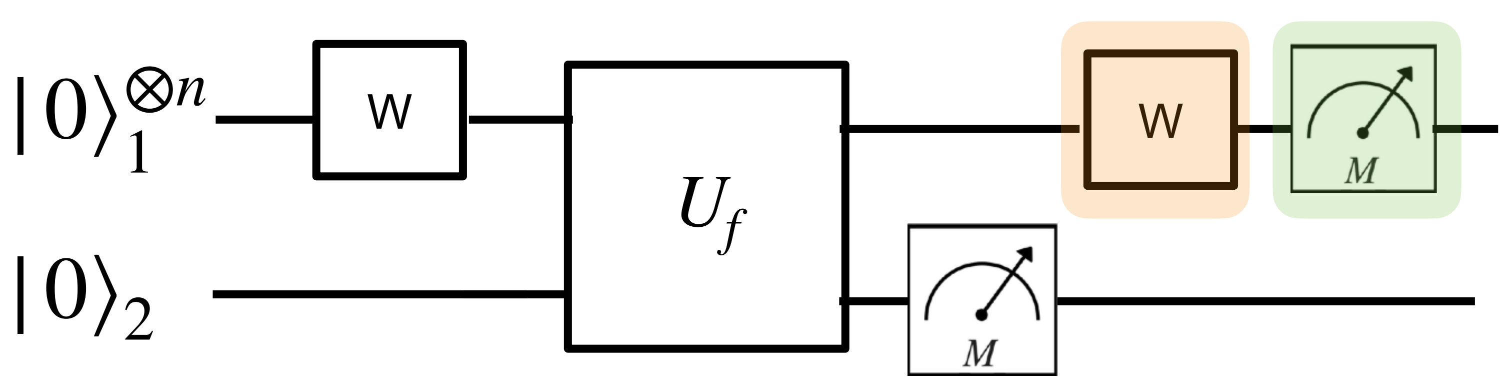
$$W(|r\rangle) = \frac{1}{2^n}\sum_{s=0}^{2^n-1}(-1)^{s\cdot r}|r\rangle$$

$$= \frac{1}{\sqrt{2}^{n+1}}\sum_y (-1)^{x_0\cdot y}\left(1 + (-1)^{a\cdot y}\right)|y\rangle$$

$$W_{rs} = W_{sr} = \frac{1}{\sqrt{2}^n}(-1)^{r\cdot s}$$

$$= \frac{1}{\sqrt{2}^{n+1}}\sum_{y\cdot a=even}(-1)^{x_0\cdot y}|y\rangle$$

- Measurement on the 1st qubit results in a random y such that $y \cdot a = 0 \bmod 2$.
- Unknown a_i must satisfy $y_0a_0 + y_1a_1 + \cdots y_{n-1}a_{n-1} = 0 \bmod 2$.



Simon's Algorithm

- Repeat the same procedure until n linearly independent equations have been found. Each time computation is repeated, at least 50% of the time, the resulting equation can be independent.
- Repeating 2n times, there is a 50% chance that n-linearly independent equations can be found.
- The equation can be solved to find the string a in $\mathcal{O}(n^2)$ steps.
- With high likelihood, the hidden string a will be found with $\mathcal{O}(n)$ calls to U_f, followed by $\mathcal{O}(n^2)$ steps to solve the resulting set of equations.
- Classical algorithm needs $\mathcal{O}(2^{n/2})$ calls to f.

Simon's Algorithm: probability of finding n-linearly independent equations

- Consider we have a string, $x = (x_1 x_2 x_3 \cdots x_n)$.
- 1st measurement: $P_1 = 1$
- After 1st measurement, what is the probability that next measurement will be linearly independent? $P_2 = 1 - 1/2^n$
- Probability that next measurement will be linearly independent: $P_2 = 1 - 2/2^n$
- Probability that next string x_{m+1} is linearly independent: $P_2 = 1 - 2^m/2^n$
- Probability of $n - 1$ being linearly independent:

$$P = \left(1 - \frac{1}{2^n}\right)\left(1 - \frac{2}{2^n}\right)\cdots\left(1 - \frac{1}{2^{n-2}}\right) \geq 1 - \sum_{k=2}^{n} \frac{1}{2^k} = 1 - \frac{\frac{1}{4}\left(1 - \frac{1}{2^{n-1}}\right)}{1 - \frac{1}{2}} \geq \frac{1}{2} + \frac{1}{2^n}$$

$$(1 - a)(1 - b) = 1 - a - b + ab \geq 1 - a - b \quad \text{for } 0 < a, b < 1$$

Discrete Fourier Transformation

- Simon's algorithm $\longrightarrow$ Shor's algorithm (factoring numbers) makes use of QFT.
- Discrete Fourier Transformation (DFT): signal processing, quantum theory (position $\leftrightarrow$ momentum).
- Assume a vector f of N complex numbers: $f_k\,,\ k = 0,1,\cdots,N-1$
- DFT is a mapping from N complex # to N complex #.

$$\text{DFT}:\ f_k \longrightarrow \tilde{f}_j = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} w^{-jk} f_k \qquad\qquad w = \exp\left(\frac{2\pi i}{N}\right)$$

$$\text{Inverse DFT}:\ \tilde{f}_k \longrightarrow \tilde{f}_j = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} w^{jk} \tilde{f}_k$$

nonzero only
when $j = \ell$

$$f_j = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} w^{jk}\tilde{f}_k = \frac{1}{\sqrt{N}}\sum_{k=0}^{N-1} w^{jk}\left(\frac{1}{\sqrt{N}}\sum_{\ell=0}^{N-1} w^{-\ell k} f_\ell\right) = \frac{1}{N}\sum_{\ell}^{N-1}\overbrace{\sum_{k=0}^{N-1} w^{(j-\ell)k}} f_\ell = \sum_{\ell}^{N-1} f_\ell\, \delta_{j\ell} = f_j$$

$$\boxed{\frac{1}{N}\sum_{k=0}^{N-1} w^{(j-\ell)k} = \delta_{j\ell}}$$

$$\frac{1}{N}\sum_{k=0}^{N-1} w^{(j-\ell)k} = \begin{cases} \dfrac{1}{N}\,\dfrac{1-\exp\left(\frac{2\pi i}{N}(j-\ell)N\right)}{1-\exp\left(\frac{2\pi i}{N}\right)} = 0\,, & \text{if } j \neq \ell \\ 1\,, & \text{if } j = \ell \end{cases}$$

Discrete Fourier Transformation

- Convolution (circular convolution, periodic convolution, cyclic convolution)

$$(f * g)_i = \sum_{j=0}^{N-1} f_i\, g_{i-j}\,, \qquad \text{where } g_{-m} = g_{N-m} \text{ (periodic condition)}$$

- DFT turns convolution into point wise vector multiplication.

$$\text{DFT of } f * g = \tilde{c}_k = \tilde{f}_k\, \tilde{g}_k$$

$$\tilde{c}_k = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{-jk} (f * g)_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{-jk} \left(\sum_{i=0}^{N-1} f_i\, g_{j-i} \right)$$

δ_{mk} δ_{ik}

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{-jk} \sum_{i=0}^{N-1} \left(\frac{1}{\sqrt{N}} \sum_{\ell} w^{i\ell} \tilde{f}_\ell \right) \left(\frac{1}{\sqrt{N}} \sum_{m} w^{(j-i)m} \tilde{g}_m \right) = \frac{1}{\sqrt{N}^3} \sum_{j,i,\ell,m} \tilde{f}_\ell\, \tilde{g}_m \underbrace{w^{-jk}\, w^{i\ell}\, w^{jm}\, w^{-im}}_{\delta_{\ell k}} = \tilde{f}_k\, \tilde{g}_k$$

$$\text{DFT}: \; f_k \longrightarrow \tilde{f}_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{-jk} f_k$$

$$\frac{1}{N} \sum_{k=0}^{N-1} w^{(j-\ell)k} = \delta_{j\ell} \qquad w = \exp\left(\frac{2\pi i}{N} \right)$$

$$\text{Inverse DFT}: \; \tilde{f}_k \longrightarrow \tilde{f}_j = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{jk} \tilde{f}_k$$

Fast Fourier Transformation

Quantum Fourier Transformation

- For classical discrete Fourier transformation

$$y_k = \frac{1}{\sqrt{2}^n} \sum_{j=0}^{2^n-1} w^{jk}\, x_j \qquad w = \exp\left(\frac{2\pi i}{2^n}\right) \qquad N = 2^n$$

- QFT is defined similarly $\quad F:\ |j\rangle \longrightarrow \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} w^{jk}\, x_k = F\,|j\rangle$

- For arbitrary quantum states, $\quad F:\ x\rangle = \frac{1}{\sqrt{2}^n} \sum_{j=0}^{2^n-1} x_j\,|j\rangle \longrightarrow |y\rangle = \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} y_k\,|k\rangle$

$$F\,|x\rangle = \frac{1}{\sqrt{2}^n} \sum_{j=0}^{2^n-1} x_j\, F\,|j\rangle = \frac{1}{\sqrt{2}^n} \sum_{j=0}^{2^n-1} \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} x_j\, w^{jk}\,|k\rangle$$

- For a single quantum state, $\quad F\,|j\rangle = \frac{1}{\sqrt{2}^n} \sum_{j=0}^{2^n-1} w^{jk}\,|k\rangle \qquad F\,|j'\rangle = \frac{1}{\sqrt{2}^n} \sum_{j'=0}^{2^n-1} w^{j'k'}\,|k'\rangle$

$$\langle j'\,|\,F^\dagger F\,|\,j\rangle = \frac{1}{2^n} \sum_{k=0}^{2^n-1} \sum_{k'=0}^{2^n-1} w^{-j'k'} w^{jk} \langle k'\,|\,k\rangle = \frac{1}{2^n} \sum_{k=0}^{2^n-1} w^{(j-j')k} = \delta_{jj'}$$

$$\boxed{\frac{1}{2^n} \sum_{k=0}^{2^n-1} w^{(j-\ell)k} = \delta_{j\ell}}$$

$F^\dagger F = 1$ and QFT is a unitary transformation.

Quantum Fourier Transformation

For $$j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_n 2^0 = \sum_{i=1}^{n} j_i\, 2^{n-i}$$

$$k = k_1 2^{n-1} + k_2 2^{n-2} + \cdots + k_n 2^0 = \sum_{i=1}^{n} k_i\, 2^{n-i}$$

$$\frac{1}{2^n} \sum_{k=0}^{2^n-1} w^{(j-\ell)k} = \delta_{j\ell}$$

$$F\,|\,j\rangle = \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} w^{jk}\,|\,k\rangle = \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} \exp\left(\frac{2\pi i j}{2^n} \sum_{\ell=1}^{n} k_\ell 2^{n-\ell}\right)|\,k\rangle$$

$$= \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} \exp\left(2\pi i j \sum_{\ell=1}^{n} k_\ell 2^{-\ell}\right)|\,k\rangle$$

$$= \frac{1}{\sqrt{2}^n} \sum_{k=0}^{2^n-1} \exp\left(2\pi i j k_1 2^{-1}\right) \exp\left(2\pi i j k_2 2^{-2}\right) \cdots \exp\left(2\pi i j k_n 2^{-n}\right)|\,k\rangle$$

$$= \frac{1}{\sqrt{2}^n} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} \exp\left(2\pi i j k_1 2^{-1}\right) \exp\left(2\pi i j k_2 2^{-2}\right) \cdots \underbrace{\exp\left(2\pi i j k_n 2^{-n}\right)}_{=\,|\,0\rangle + \exp\left(2\pi i j 2^{-n}\right)|\,1\rangle}|\,k_1\, k_2\, \cdots\, k_n\,\rangle$$

Quantum Fourier Transformation

$$F\,|j\rangle = \frac{1}{\sqrt{2}^n}\left(|0\rangle + \exp\left(\frac{2\pi ij}{2}\right)|1\rangle\right)\left(|0\rangle + \exp\left(\frac{2\pi ij}{2^2}\right)|1\rangle\right)\cdots\left(|0\rangle + \exp\left(\frac{2\pi ij}{2^n}\right)|1\rangle\right)$$

$$= \frac{1}{\sqrt{2}^n}\bigotimes_{k=1}^{n}\left(|0\rangle + \exp\left(\frac{2\pi ij}{2^k}\right)|1\rangle\right)$$

$j_i = 0\,,1$

$1 \le k \le n$

$0 \le j \le 2^n - 1$

- Binary fraction = expression in power of 1/2

In decimal form: $$0.j_\ell\, j_{\ell+1} \cdots j_m = \frac{j_\ell}{2} + \frac{j_{\ell+1}}{2^2} + \cdots + \frac{j_m}{2^{m-\ell+1}}$$

j is not necessarily an integer: $$\frac{j}{2^k} = j_1\, j_2 \cdots j_{n-k} \cdot j_{n-k+1} \cdots j_n = \sum_{\nu=1}^{n} j_\nu\, 2^{n-\nu-k}$$

If $n = 8$ and $k = 3$, $$j = j_1 2^7 + j_2 2^6 + j_3 2^5 + j_4 2^4 + j_5 2^3 + j_6 2^2 + j_7 2^1 + j_8 2^0$$

$$\frac{j}{2^3} = j_1 2^4 + j_2 2^3 + j_3 2^2 + j_4 2^1 + j_5 2^0 + \underbrace{j_6 2^{-1} + j_7 2^{-2} + j_8 2^{-3}}$$

$j_1 j_2 j_3 j_4 j_5 \cdot j_6 j_7 j_8$

binary fraction: $0\,.j_6 j_7 j_8$

Quantum Fourier Transformation

$$j = j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_{n-3} 2^3 + j_{n-2} 2^2 + j_{n-1} 2^1 + j_1 2^0 = \sum_{\nu=1}^{n} j_\nu 2^{n-\nu}$$

$$\frac{j}{2^k} = \frac{j_1 2^{n-1} + j_2 2^{n-2} + \cdots + j_{n-3} 2^3 + j_{n-2} 2^2 + j_{n-1} 2^1 + j_1 2^0}{2^k} = \sum_{\nu=1}^{n} \frac{j_\nu \, 2^{n-\nu}}{2^k} = \sum_{\nu=1}^{n} j_\nu \, 2^{n-\nu-k}$$

$$= j_1 j_2 \cdots j_{n-k} \, . j_{n-k+1} \cdots j_n$$

$$\exp\left(2\pi i \frac{j}{2^k}\right) = \exp\left(2\pi i \, 0\,. j_{n-k-1} \, \cdots \, j_n\right)$$

$$F\,|j\rangle = \frac{1}{\sqrt{2}^n} \left(|0\rangle + \exp\left(\frac{2\pi i j}{2}\right) |1\rangle \right) \left(|0\rangle + \exp\left(\frac{2\pi i j}{2^2}\right) |1\rangle \right) \cdots \left(|0\rangle + \exp\left(\frac{2\pi i j}{2^n}\right) |1\rangle \right)$$

$$= \frac{1}{\sqrt{2}^n} \bigotimes_{k=1}^{n} \left(|0\rangle + \exp\left(\frac{2\pi i j}{2^k}\right) |1\rangle \right) = \frac{1}{\sqrt{2}^n} \bigotimes_{k=1}^{n} \left(|0\rangle + \exp\left(2\pi i \, 0\,. j_{n-k-1} \, \cdots \, j_n\right) |1\rangle \right)$$

$$= \frac{1}{\sqrt{2}^n} \left(|0\rangle + \exp\left(2\pi i \, 0\,. j_n\right) |1\rangle \right) \left(|0\rangle + \exp\left(2\pi i \, 0\,. j_{n-1} j_{n-2}\right) |1\rangle \right)$$

$$\cdots \left(|0\rangle + \exp\left(2\pi i \, 0\,. j_1 j_2 \cdots j_n\right) |1\rangle \right)$$

Quantum Circuit for QFT

- $|j_\ell\rangle$ transforms into $\dfrac{1}{\sqrt{2}}\left[\,|0\rangle + \exp\left(2\pi i\, 0\,.j_\ell \cdots j_n\right)|1\rangle\right]$

$$= \frac{1}{\sqrt{2}}\left[\,|0\rangle + \underbrace{e^{2\pi i 0.j_\ell}}\,\underbrace{e^{2\pi i 0.j_{\ell+1}\cdots j_n/2}}|1\rangle\right]$$

$$\exp\left(2\pi i \frac{j_\ell}{2}\right) = \exp\left(\pi i j_\ell\right) = (-1)^{j_\ell}$$

use $R_k = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^k} \end{pmatrix}$

Controlled by the value of j_kth qubit.

$$\text{if } \begin{cases} j_k = 0\,, & R_k = 1 \\ j_k = 1\,, & R_k \end{cases}$$

$\boxed{\text{1st qubit:}}$ $|0\rangle + \exp\left(2\pi i\, 0\,.j_\ell \cdots j_n\right)|1\rangle$

Start with $|j\rangle = |j_2\rangle\,|j_2 j_3 \cdots j_n\rangle \xrightarrow{H_1} \dfrac{1}{\sqrt{2}}\left(\,|0\rangle + (-1)^{j_1}|1\rangle\right)|j_2 j_3 \cdots j_n\rangle$

$$= \frac{1}{\sqrt{2}}\left(\,|0\rangle + e^{2\pi i\, 0.j_1}|1\rangle\right)|j_2 j_3 \cdots j_n\rangle$$

$$\xrightarrow{\mathrm{R}_2 \text{ on } \mathrm{q}_1 \text{ with } \mathrm{q}_2 \text{ control}} \frac{1}{\sqrt{2}}\left(\,|0\rangle + e^{2\pi i\, 0.j_1}\, e^{2\pi i\, j_2/2^2}|1\rangle\right)|j_2 j_3 \cdots j_n\rangle$$

$$= \frac{1}{\sqrt{2}}\left(\,|0\rangle + e^{2\pi i\, 0.j_1 j_2}|1\rangle\right)|j_2 j_3 \cdots j_n\rangle$$

Quantum Circuit for QFT

$$\xrightarrow{\mathrm{R_3 \ on \ q_1 \ with \ q_3 \ control}} \frac{1}{\sqrt{2}}\Big(|0\rangle + e^{2\pi i\, 0.j_1 j_2 j_3} |1\rangle \Big) |j_2 j_3 \cdots j_n\rangle$$

$$\xrightarrow[\mathrm{to \ q_n}]{\mathrm{continue \ down}} \frac{1}{\sqrt{2}}\Big(|0\rangle + e^{2\pi i\, 0.j_1 j_2 j_3 \cdots j_n} |1\rangle \Big) |j_2 j_3 \cdots j_n\rangle$$

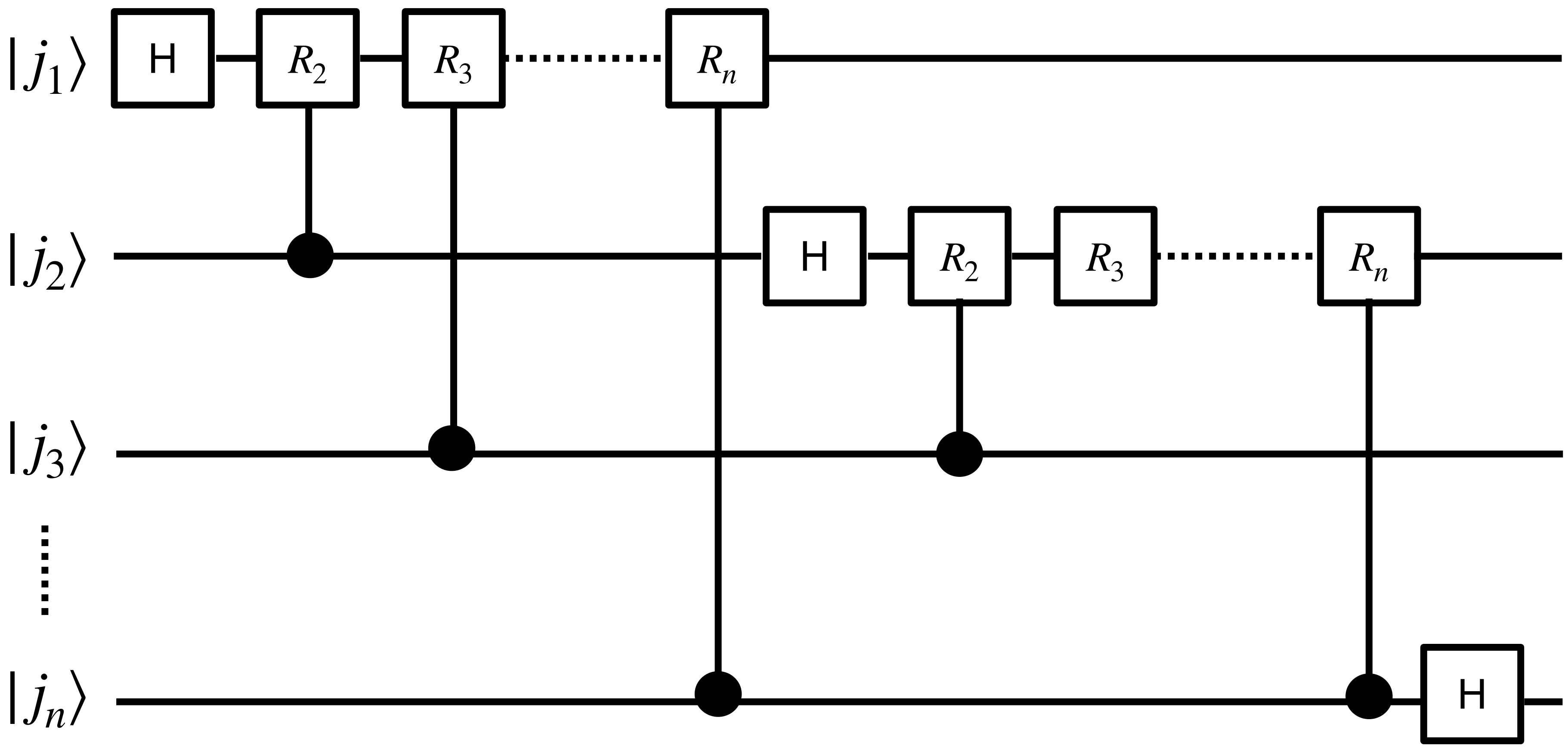
The entire procedure is repeated for all other qubits, $j_2, j_3, \cdots j_n$

$$\frac{1}{\sqrt{2}^n}\Big[|0\rangle + e^{2\pi i\, 0.j_1 \cdots j_n} |1\rangle \Big]\Big[|0\rangle + e^{2\pi i\, 0.j_2 \cdots j_n} |1\rangle \Big] \cdots \Big[|0\rangle + e^{2\pi i\, 0.j_n} |1\rangle \Big]$$

Use SWAP gate or relabel to obtain: $F|j\rangle = \dfrac{1}{\sqrt{2}^n} \displaystyle\bigotimes_{k=1}^{n} \left(|0\rangle + \exp\!\left(\frac{2\pi i j}{2^k}\right) |1\rangle \right)$

$$\frac{1}{\sqrt{2}^n}\Big[|0\rangle + e^{2\pi i\, 0.j_n} |1\rangle \Big]\Big[|0\rangle + e^{2\pi i\, 0.j_2 \cdots j_n} |1\rangle \Big] \cdots \Big[|0\rangle + e^{2\pi i\, 0.j_1 \cdots j_n} |1\rangle \Big]$$

Quantum Circuit for QFT



$$\frac{1}{\sqrt{2}}\Big[\,|0\rangle + e^{2\pi i\, 0.j_1\cdots j_n}\,|1\rangle\,\Big]$$

$$\frac{1}{\sqrt{2}}\Big[\,|0\rangle + e^{2\pi i\, 0.j_2\cdots j_n}\,|1\rangle\,\Big]$$

$$\vdots$$

$$\frac{1}{\sqrt{2}}\Big[\,|0\rangle + (-1)^{j_n}\,|1\rangle\,\Big] = \frac{1}{\sqrt{2}}\Big[\,|0\rangle + e^{2\pi i\, 0.j_1}\,|1\rangle\,\Big]$$

How many gates are required?

q_1: H + (n-1) controlled R gates $\longrightarrow$ n

q_2: H + (n-2) controlled R gates $\longrightarrow$ n-1

$\vdots$ $\qquad\qquad\vdots\qquad\qquad\vdots$

q_n: H + 0 controlled R gates $\longrightarrow$ 1

$$\left.\begin{matrix} n \\ n-1 \\ \vdots \\ 1 \end{matrix}\right\} \frac{n(n+1)}{2}$$

Also need $\mathcal{O}(n/2)$ SWAP gates

Overall scaling of QFT is $\mathcal{O}(n^2)$

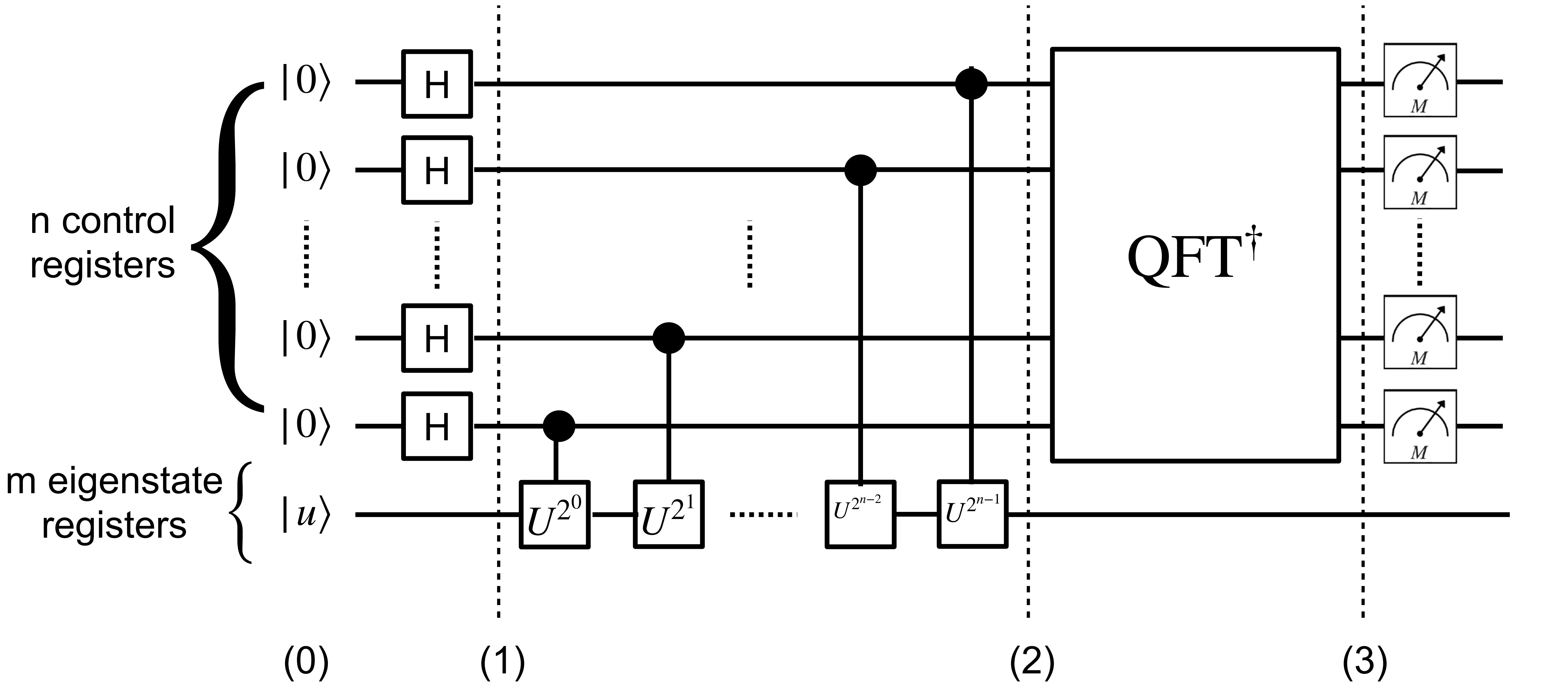
- Classical Fourier Transform scales as $\mathcal{O}(N^2) = \mathcal{O}((2^n)^2)$
- FFT: $\mathcal{O}(N ln(N))$ for $N = 2^n$

Quantum Phase Estimation and Finding Eigenvalues

- Good example of phase kickback and use of QFT
- Unitary operator $U: \; U|u\rangle = e^{i\phi}|u\rangle, \qquad 0 \le \phi < 2\pi$
- How to find eigenvalue? = How to measure the phase?
- How to find ϕ to a given level of precision?
- Find the best n-bit estimate of the phase ϕ

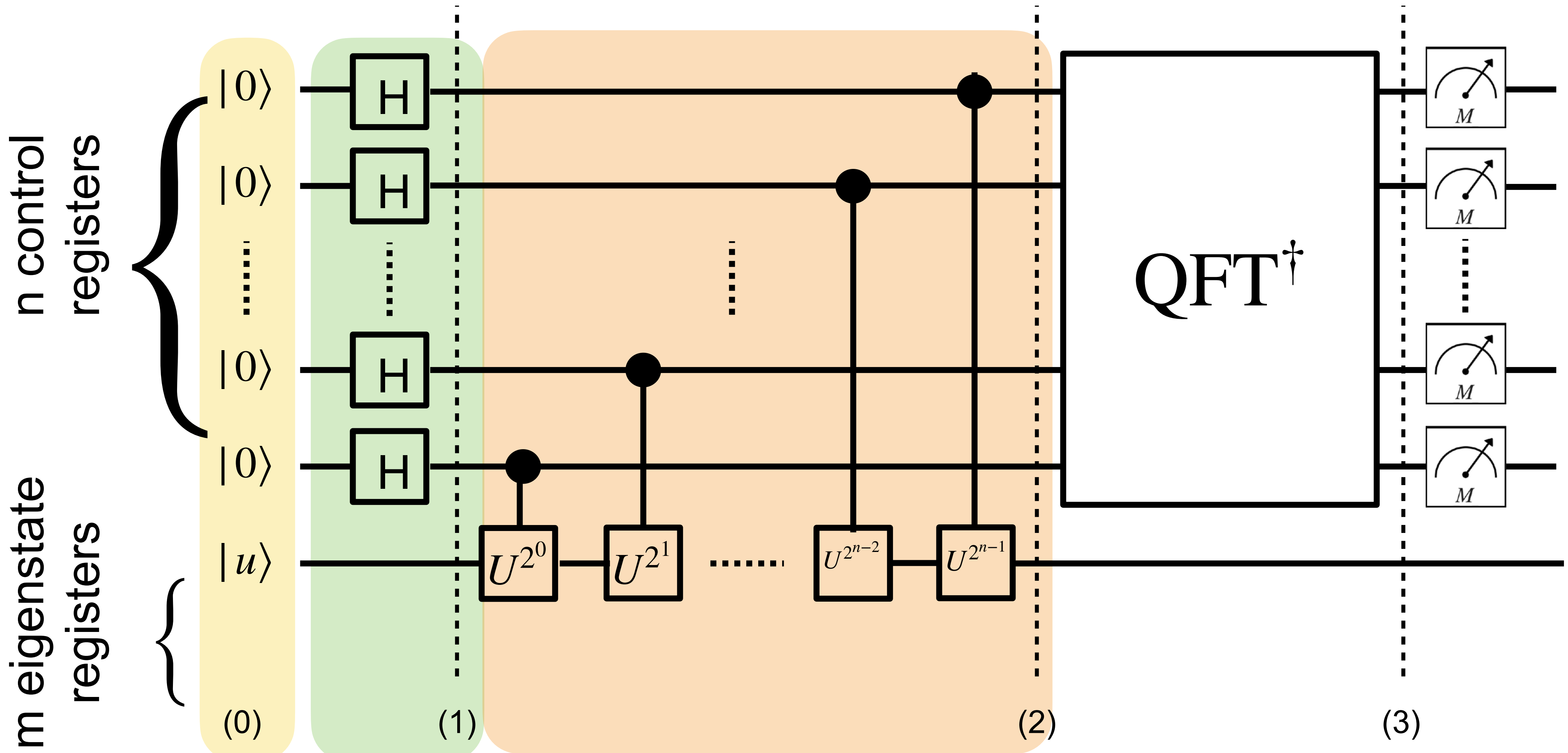
$$U^{2j}|u\rangle = \left(e^{i\phi}\right)^{2^j}|u\rangle = e^{i\phi\,2^j}|u\rangle$$

Quantum Circuit for QPE



$$\text{QPE} \ = H + \text{controlled} - U^{2^j} + \text{QFT}^\dagger$$

Quantum Circuit for QPE



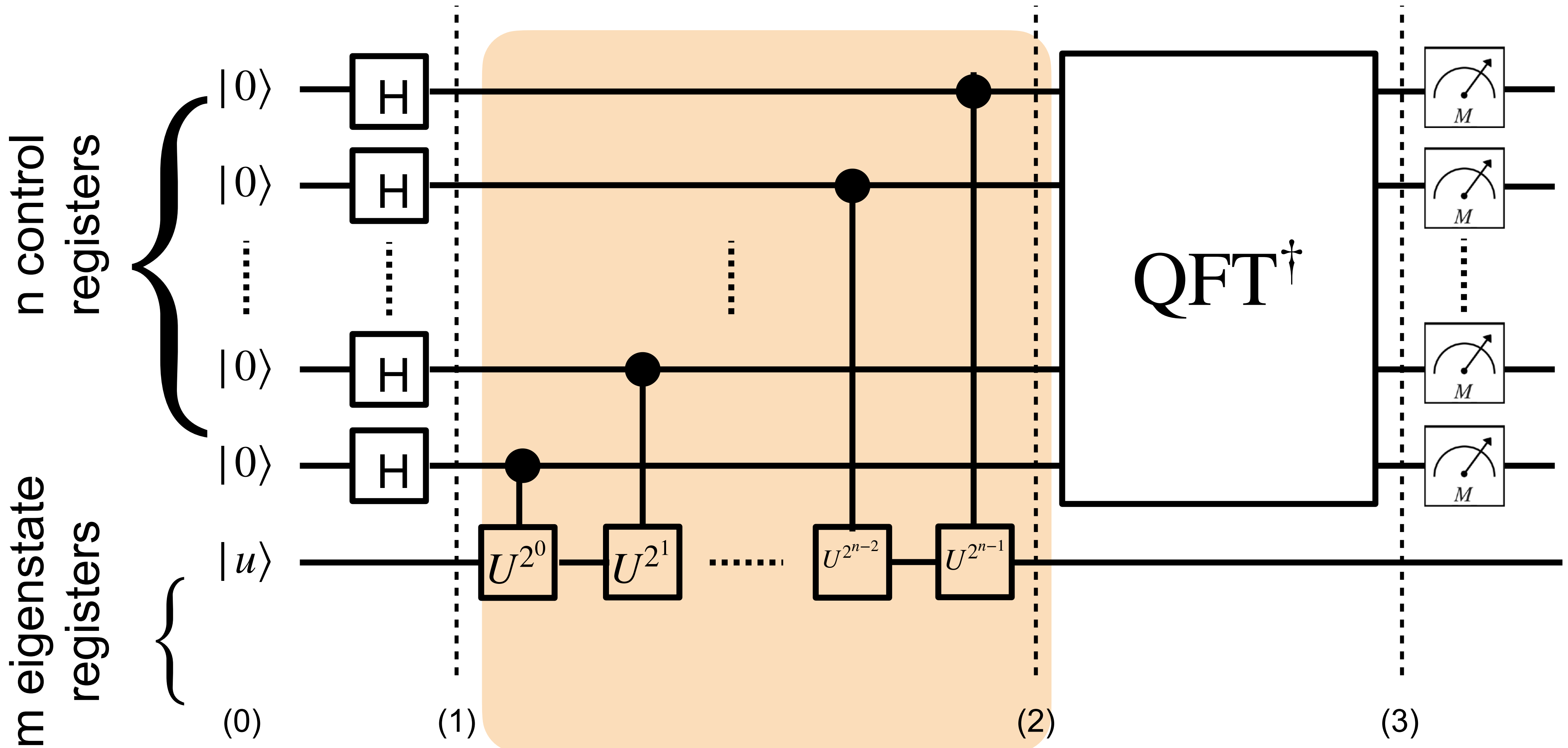
$$\text{QPE} \;= H + \text{controlled} - U^{2^j} + \text{QFT}^\dagger$$

$$|\psi_0\rangle = |0\rangle^{\otimes n} \otimes |u\rangle$$

$$|\psi_1\rangle = \Big(H|0\rangle\Big)^{\otimes n} \otimes |u\rangle = \frac{1}{\sqrt{2}^n}\Big(|0\rangle + |1\rangle\Big)^{\otimes n} \otimes |u\rangle$$

$$|\psi_2\rangle = \prod_{j=0}^{n-1} \text{CU}^{2^j} \frac{1}{\sqrt{2}^n}\Big(|0\rangle + |1\rangle\Big)^{\otimes n} \otimes |u\rangle$$

Quantum Circuit for QPE



$$|\psi_2\rangle = \prod_{j=0}^{n-1} \mathrm{CU}^{2^j} \frac{1}{\sqrt{2}^n}\Big(|0\rangle + |1\rangle\Big)^{\otimes n} \otimes |u\rangle$$

$$\frac{1}{\sqrt{2}}\Big(|0\rangle + |1\rangle\Big) \otimes |u\rangle \xrightarrow{\mathrm{CU}^{2^j}} \frac{1}{\sqrt{2}}\Big(|0\rangle \otimes |u\rangle + U^{2^j}|1\rangle \otimes |u\rangle\Big)$$

$$= \frac{1}{\sqrt{2}}\Big(|0\rangle + e^{i\phi\, 2^j}|1\rangle\Big) \otimes |u\rangle$$

Quantum Circuit for QPE

$$|\psi_2\rangle = \frac{1}{\sqrt{2}^n}\Big(|0\rangle + e^{i\phi\, 2^{n-1}}|1\rangle\Big)\Big(|0\rangle + e^{i\phi\, 2^{n-2}}|1\rangle\Big)\cdots\Big(|0\rangle + e^{i2\phi}|1\rangle\Big)\Big(|0\rangle + e^{i\phi}|1\rangle\Big)\otimes|u\rangle$$

$$= \frac{1}{\sqrt{2}^n}\sum_{y=0}^{2^n-1} \underbrace{e^{i\phi y}}|y\rangle \otimes |u\rangle$$

Phase kick-back: phase factor $e^{i\phi y}$ has been propagated back from the second eigenstate register to the first control register

$$\mathrm{QFT}\,|a\rangle = \frac{1}{\sqrt{2}^n}\sum_{k=0}^{2^n-1} e^{2\pi i a/2^n}|k\rangle \longrightarrow \frac{2\pi i a}{2^n} = i\phi \longrightarrow \boxed{\phi = 2\pi\Big(\frac{a}{2^n}+\delta\Big)}$$

$$a = a_{n-1}a_{n-2}\cdots a_0$$

- $\dfrac{2\pi a}{2^n}$ is the best n-bit binary approximation of ϕ .
- $0 \le |\delta| \le \dfrac{1}{2^{n+1}}$ is the associated error.

$$\mathrm{QFT}^{-1}|y\rangle = \frac{1}{\sqrt{2}^n}\sum_{x=0}^{2^n-1} e^{-2\pi i x y)/2^n}|x\rangle$$

$$|\psi_3\rangle = \mathrm{QFT}^{-1}|\psi_2\rangle = \frac{1}{2^n}\sum_{x=0}^{2^n-1}\sum_{y=0}^{2^n-1} e^{2\pi i(a-x)y/2^n}e^{2\pi i\delta y}|x\rangle\otimes|u\rangle$$

Operate only n control register.

Quantum Circuit for QPE

$$|\psi_3\rangle = \text{QFT}^{-1}|\psi_2\rangle = \frac{1}{2^n}\sum_{x=0}^{2^n-1}\sum_{y=0}^{2^n-1} e^{2\pi i(a-x)y/2^n} e^{2\pi i\delta y}|x\rangle \otimes |u\rangle$$

Operate only n control register.

(1) If $\delta = 0$, $\frac{1}{2^n}\sum_{y=0}^{2^n-1} \exp\left(\frac{2\pi i(a-x)y}{2^n}\right) = \delta_{ax} \longrightarrow |\psi_3\rangle = |a\rangle \otimes |u\rangle \longrightarrow \phi = \frac{2\pi a}{2^n}$

(2) If $\delta \neq 0$, Measuring 1st register and getting the state $|x\rangle = |a\rangle$ is the best n-bit estimate of ϕ. The corresponding probability is $P_a = |C_a|^2 \geq \frac{4}{\pi^2} \approx 0.405$

Quantum Circuit for QPE

$$|\psi_2\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} e^{2\pi i x\phi} |x\rangle \otimes |u\rangle$$

$$\mathrm{QFT}^{-1}|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{y=0}^{2^n-1} e^{-2\pi i xy/2^n} |y\rangle$$

$$|\psi_3\rangle = \mathrm{QFT}^{-1}|\psi_2\rangle = \frac{1}{2^n} \sum_{x=0}^{2^n-1} \sum_{y=0}^{2^n-1} e^{2\pi i x(\phi - y/2^n)} |y\rangle \otimes |u\rangle$$

Probability of observing $|y\rangle$ = $P(y) = \left| \frac{1}{2^n} \sum_{x=0}^{2^n-1} e^{2\pi i x(\phi - y/2^n)} \right|^2 = \frac{1}{2^{2n}} \left| \frac{1-r^{2^n}}{1-r} \right|^2, \quad r \equiv \exp\left[2\pi i\left(\phi - \frac{y}{2^n}\right)\right]$

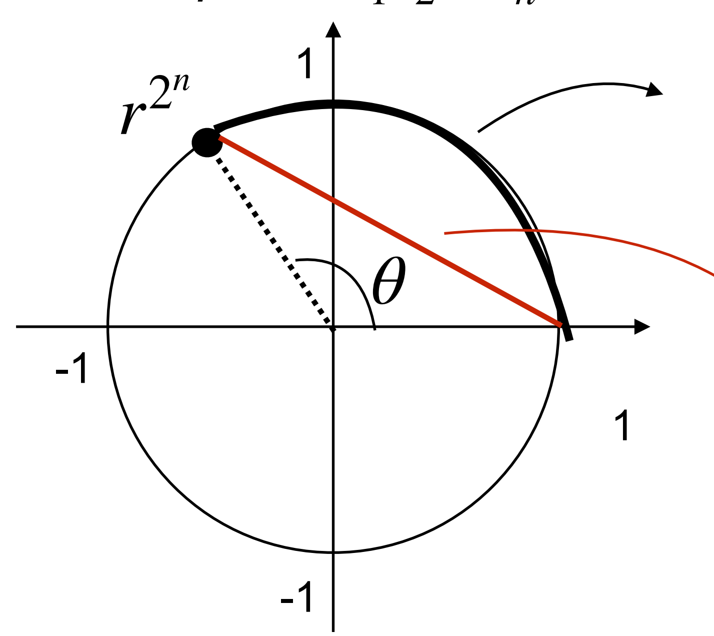
(1) If $\phi = \frac{y}{2^n}$, $\quad |\psi_3\rangle = |y\rangle \otimes |u\rangle \qquad P(\phi = \frac{y}{2^n}) = 100\,\%$

(2) If $\phi \neq \frac{y}{2^n}$, $\quad$ closest n − bit approximation to $\phi = 0.\nu_1\nu_2\cdots\nu_n = \equiv \nu \qquad \phi - \nu \equiv \delta, \quad 0 \le |\delta| \le \frac{1}{2^{n+1}}$

$$r \equiv \exp\left[2\pi i\left(\phi - \frac{y}{2^n}\right)\right] = \exp(2\pi i\delta)$$

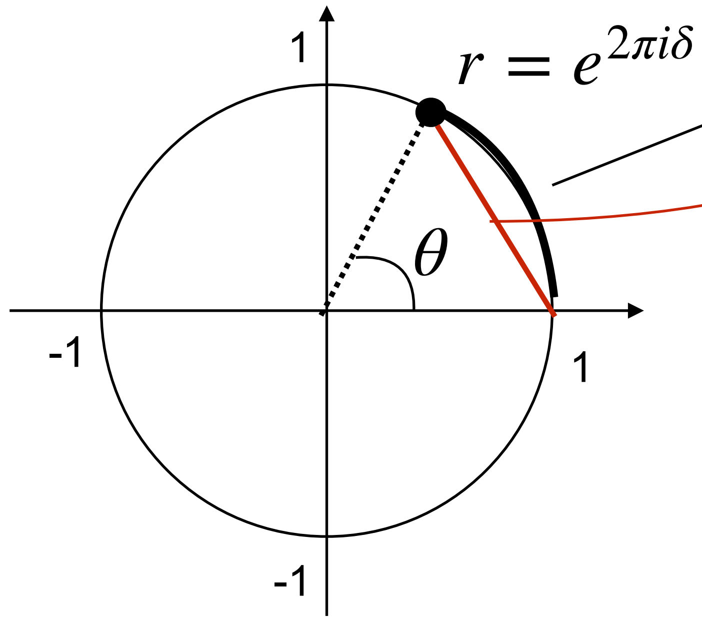
$$P(y) = \frac{1}{2^{2n}} \left| \frac{1-r^{2^n}}{1-r} \right|^2,$$

$$r^{2^n} = \left[\exp(2\pi i\delta)\right]^{2^n} = \exp(2\pi i\delta 2^n) = e^{i\theta}$$



$$\frac{\text{length of minor arc}}{\text{length of cord}} = \frac{2\pi\delta 2^n}{|1-r^{2^n}|} \le \frac{\text{half circumference}}{\text{diameter}} \le \frac{\pi R}{2R} = \frac{\pi}{2} \longrightarrow |1-r^{2^n}| \ge 4\delta 2^n$$

Quantum Circuit for QPE



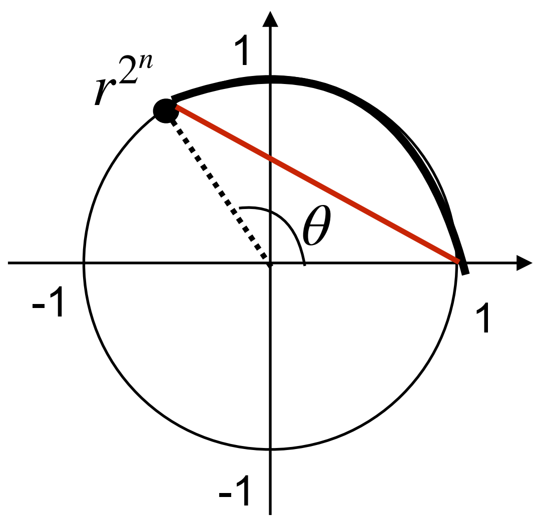
Length of minor arc = $\theta = 2\pi\delta 2^n$

Length of a cord from 1 to $r = |1 - r|$

$$\frac{\text{length of minor arc}}{\text{length of cord}} = \frac{2\pi\delta}{|1-r|} > 1\,, \qquad |1-r| < 2\pi\delta$$

$$P(y) = \frac{1}{2^{2n}}\left|\frac{1-r^{2^n}}{1-r}\right|^2 \geq \frac{1}{2^{2n}}\left(\frac{4\delta 2^n}{2\pi\delta}\right)^2 = \frac{4}{\pi^2} > 0.405$$

- We will get the correct answer with probability greater than a constant.
- Probability of getting incorrect outcome can be calculated using $|\delta| > \frac{1}{2^{n+1}}$



$$|1 - r^{2^n}| < 2 \qquad \frac{\text{length of minor arc}}{\text{length of cord}} = \frac{2\pi\delta}{|1-r|} < \frac{\pi}{2}\,, \qquad |1-r| > 4\pi\delta$$

$$P(y) = \frac{1}{2^{2n}}\left|\frac{1-r^{2^n}}{1-r}\right|^2 \leq \frac{1}{2^{2n}}\left(\frac{2}{4\delta}\right)^2 = \frac{1}{2^{2n}(2\delta)^2}$$

If $\delta = \frac{c}{2^n}$, $P(c) \leq \frac{1}{4c^2}$

- N-bit estimate of phase ϕ is obtained with a high probability.
- Need to repeat the calculation multiple times.
- Increasing n will increase the probability of success (not obvious but true).
- Increasing n (# of qubits) will improve the precision of the phase estimate.