Yale University

# Lecture 7 – Quantum Teleportation

CPSC 447/547

# Introduction to Quantum Computing

Instructor: Prof. **Yongshan Ding**

Computer Science, Applied Physics, Yale Quantum Institute

# Entanglement – non-local information

**Shared state over a distance**

## Classical Systems

Deterministic


Random


## Quantum Systems

If measured in an agreed basis, their outcomes will always be the same.


$$\frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle = \frac{1}{\sqrt{2}}|+\,+\rangle + \frac{1}{\sqrt{2}}|-\,-\rangle$$

Information is stored as “correlations” among the constituent subsystems.

# Entanglement as a Resource

**Quantum Teleportation:** Quantum information swapping at a distance.


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

**Direct quantum communication:**

Physically sending the qubit $|\psi\rangle$ from Alice to Bob: e.g., as a photon.

**Classical communication** without entanglement:

Even for a known $|\psi\rangle$, we need classical communication of 2 complex numbers to tell Bob what is $|\psi\rangle$.

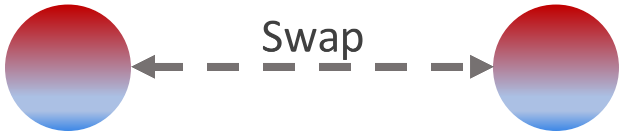
**Classical communication** with entanglement:

Classical communication of 2 classical bits and local gates by Alice and Bob.

Entanglement as a resource:

- Teleportation.
- Error-correcting code.
- Distributed computing.
- Certifiable randomness.
- Secure communication.
- Precise quantum sensors.
- Etc.

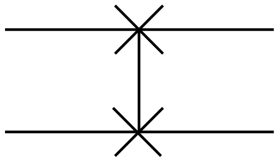
# Information Swapping




$|\psi_A\rangle = \alpha_A|0\rangle + \beta_A|1\rangle$ $\quad$ $|\psi_B\rangle = \alpha_B|0\rangle + \beta_B|1\rangle$

- Swapping Unentangled State: $|\psi_A\rangle \otimes |\psi_B\rangle$
- Swapping Entangled State: $|\psi_{AB}\rangle$

A quantum **swap gate** exchanges the amplitudes on $|01\rangle$ and $|10\rangle$.



$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Swap: $|\psi_A\rangle \otimes |\psi_B\rangle \rightarrow |\psi_B\rangle \otimes |\psi_A\rangle$

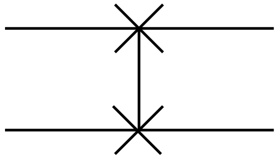
$$\begin{bmatrix} \alpha_A\alpha_B \\ \alpha_A\beta_B \\ \beta_A\alpha_B \\ \beta_A\beta_B \end{bmatrix} \quad \begin{bmatrix} \alpha_A\alpha_B \\ \beta_A\alpha_B \\ \alpha_A\beta_B \\ \beta_A\beta_B \end{bmatrix}$$

Swap: $|\psi_{AB}\rangle \rightarrow |\psi_{BA}\rangle$

$$\begin{bmatrix} \alpha_{00} \\ \alpha_{01} \\ \alpha_{10} \\ \alpha_{11} \end{bmatrix} \quad \begin{bmatrix} \alpha_{00} \\ \alpha_{10} \\ \alpha_{01} \\ \alpha_{11} \end{bmatrix}$$

# Implementing Local Swap

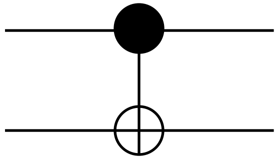
**Swap:**



$$\begin{bmatrix}1&0&0&0\\0&0&1&0\\0&1&0&0\\0&0&0&1\end{bmatrix}$$

Swap gate exchanges the amplitudes on $|01\rangle$ and $|10\rangle$.

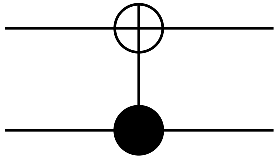
**CX(0,1)**



$$\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$$

CX(0,1) gate exchanges the amplitudes on $|10\rangle$ and $|11\rangle$.

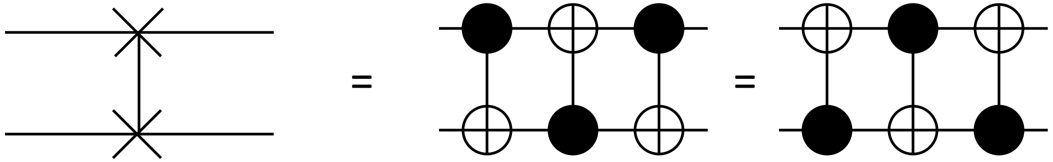
**CX(1,0)**



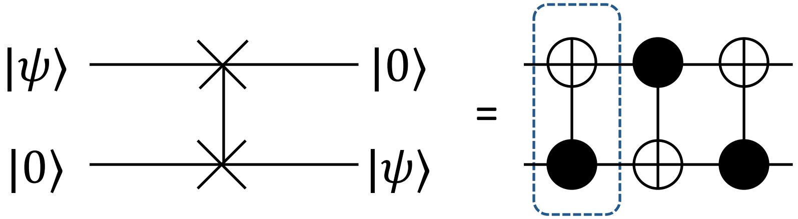
$$\begin{bmatrix}1&0&0&0\\0&0&0&1\\0&0&1&0\\0&1&0&0\end{bmatrix}$$

CX(1,0) gate exchanges the amplitudes on $|01\rangle$ and $|11\rangle$.

**Can we implement Swap using CX gates?**

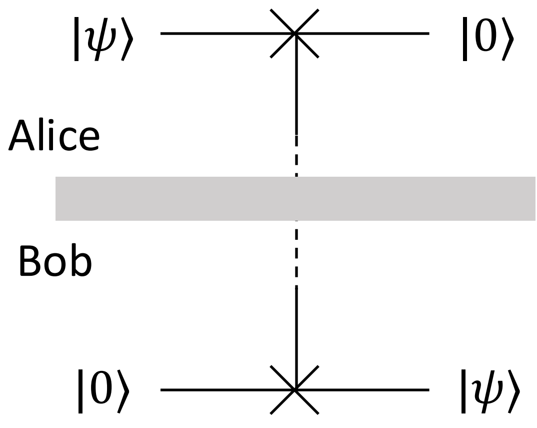



**What if we assume one of the qubits is $|0\rangle$?**



# Implementing Remote Swap

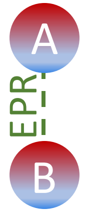
**Remote Swap:**



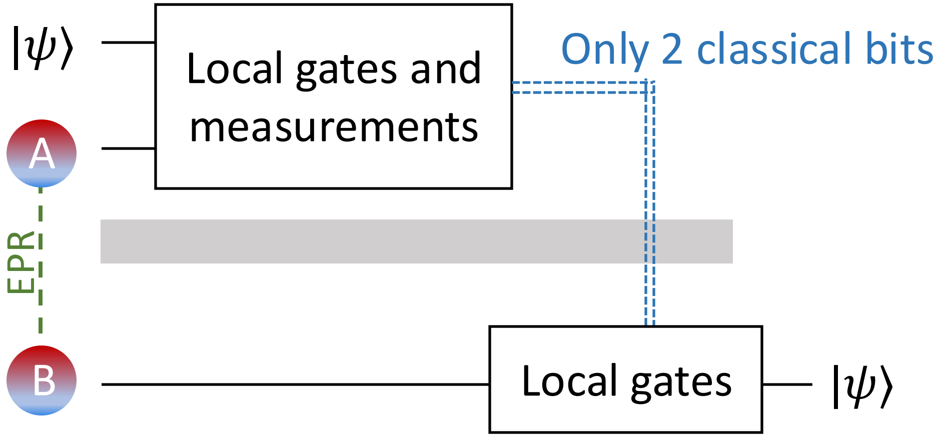

**Across the barrier:**

- Direct gates are **not** allowed between the qubits.
- Communicating classical bits are allowed.

**With the help of entanglement:**



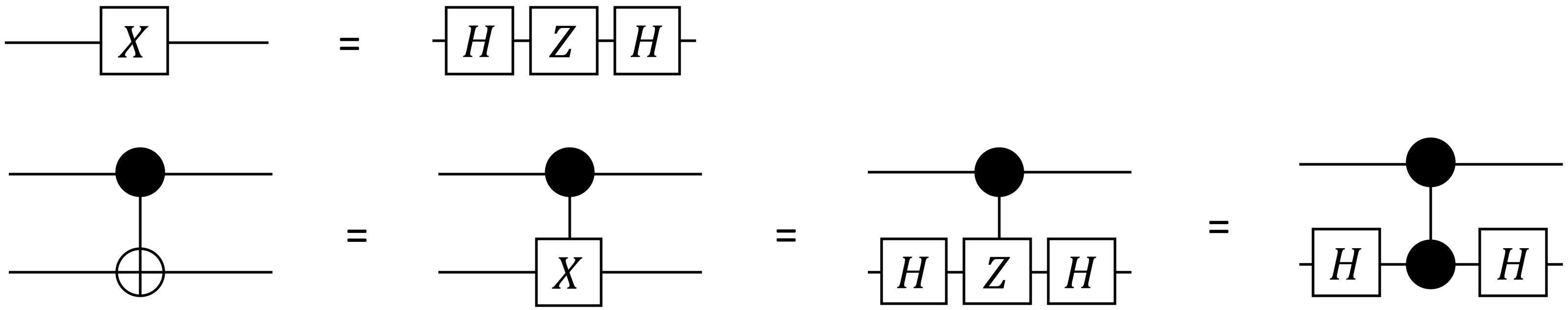

$= |\psi_{AB}\rangle = \frac{1}{\sqrt{2}}|00\rangle + \frac{1}{\sqrt{2}}|11\rangle$




Let's see how we can accomplish this next.

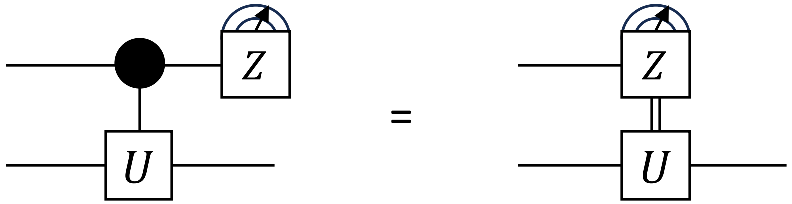
# Some Useful Facts about Quantum Circuits

**Conjugation Relations:** $X = HZH, Z = HXH$




**Measurement and Control:** $|0\rangle(\alpha_{00}|0\rangle + \alpha_{01}|1\rangle) + |1\rangle U(\alpha_{10}|0\rangle + \alpha_{11}|1\rangle)$

$$\begin{bmatrix} \alpha_{00} \\ \alpha_{01} \\ \alpha_{10} \\ \alpha_{11} \end{bmatrix}$$

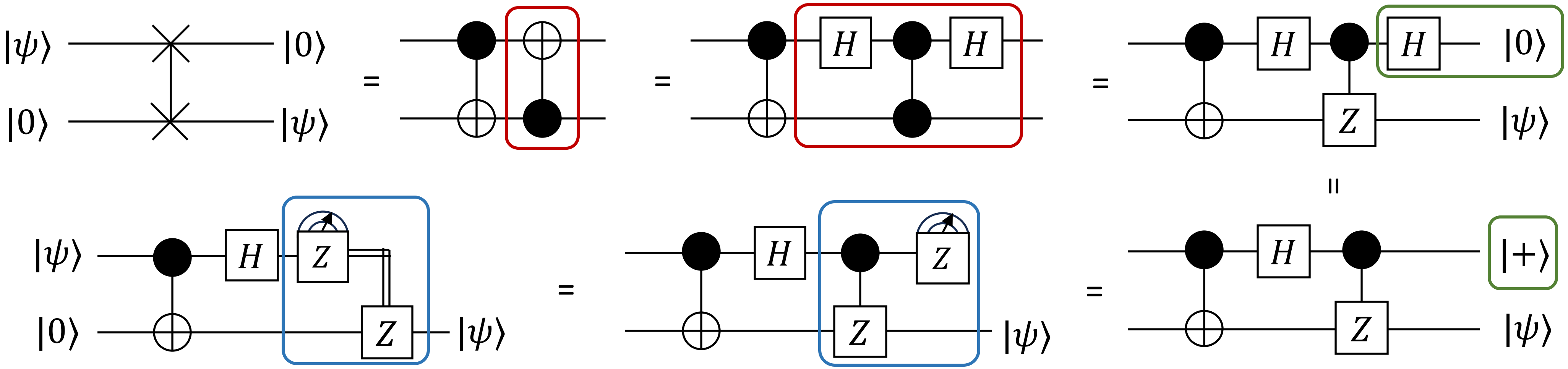


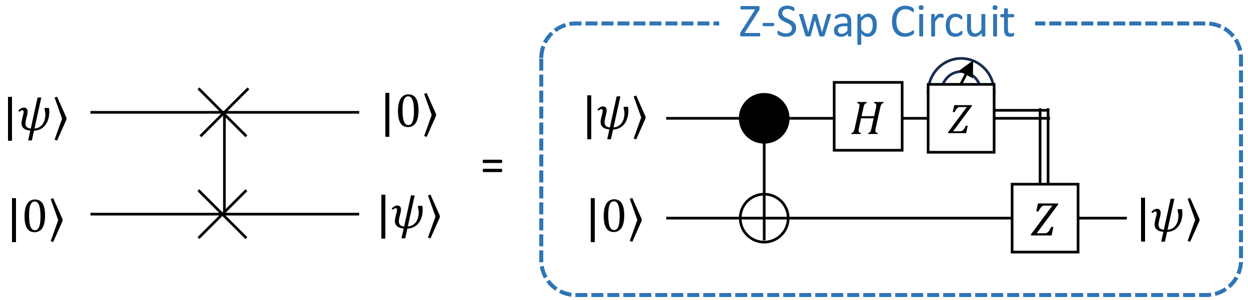

If measurement outcome is 1, then perform U gate.

Quantum controlled gate    Classical conditional gate

# Z-Swap Circuit

**One-Bit Teleportation**

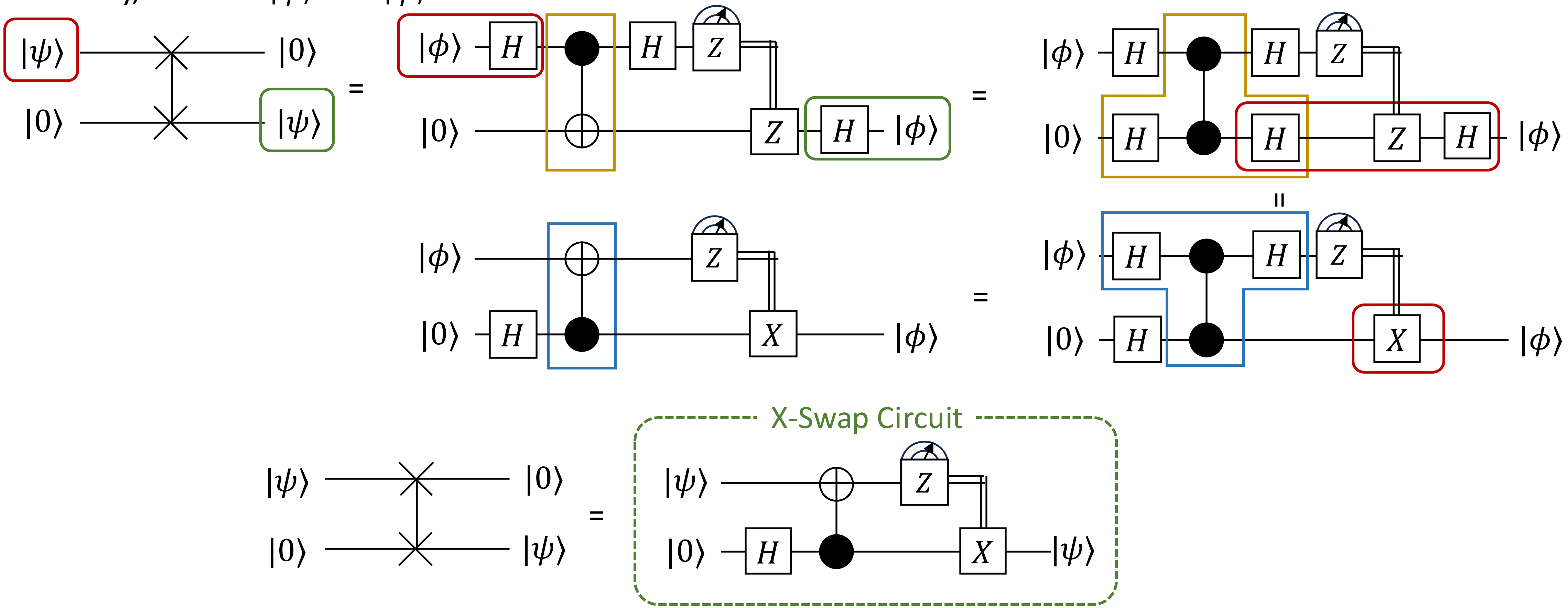


# X-Swap Circuit

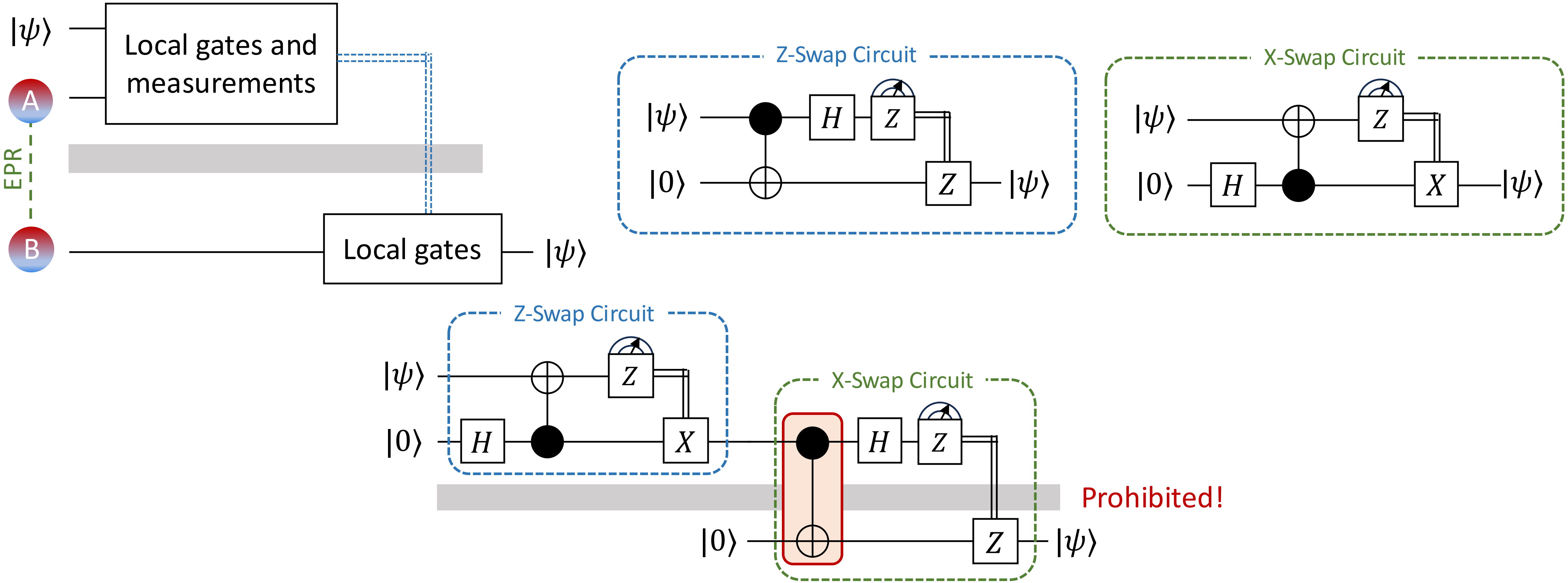
**One-Bit Teleportation**

Similarly, if we set $|\psi\rangle = H|\phi\rangle$.



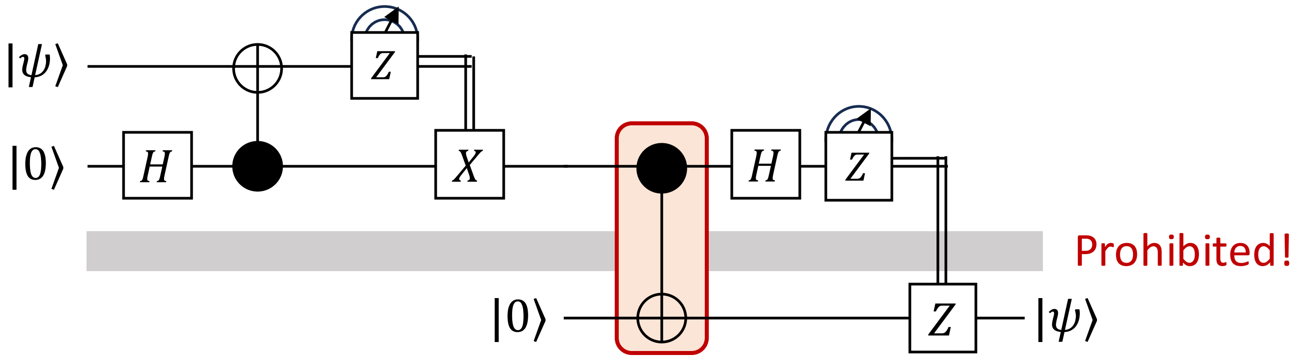
# (Two-Bit) Teleportation Circuit

**Putting it all together**

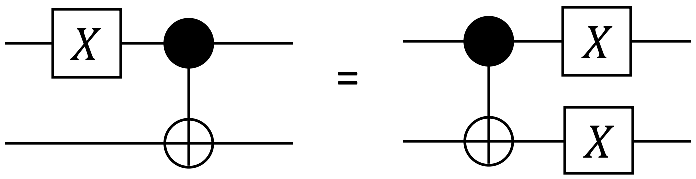


# (Two-Bit) Teleportation Circuit

**Putting it all together**




**More Circuit Equivalences:**

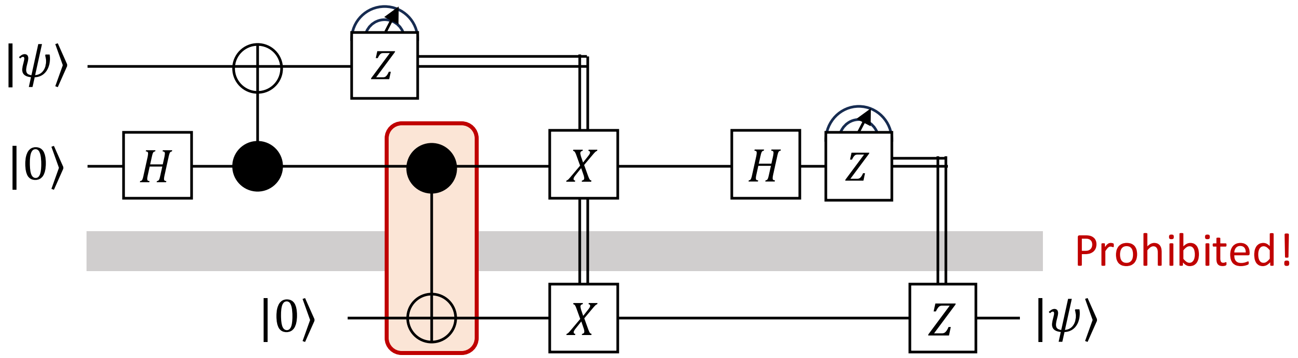



"Commute CX to the left."

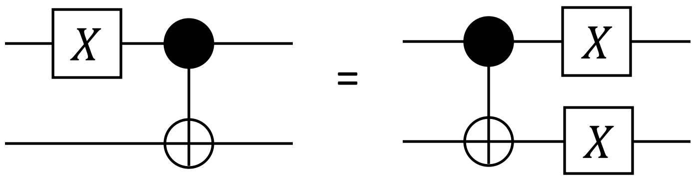
$$CX_{1,2}(X_1 \otimes I_2) = (X_1 \otimes X_2)CX_{1,2}$$

# (Two-Bit) Teleportation Circuit

**Putting it all together**

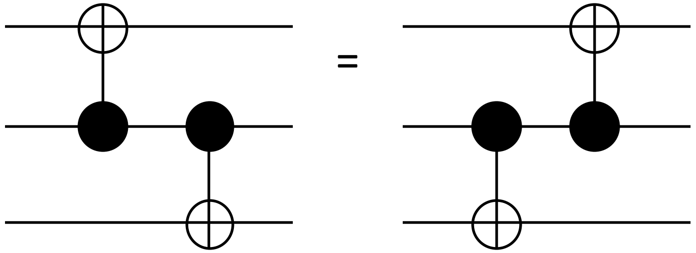



**More Circuit Equivalences:**




"Commute CX to the left."

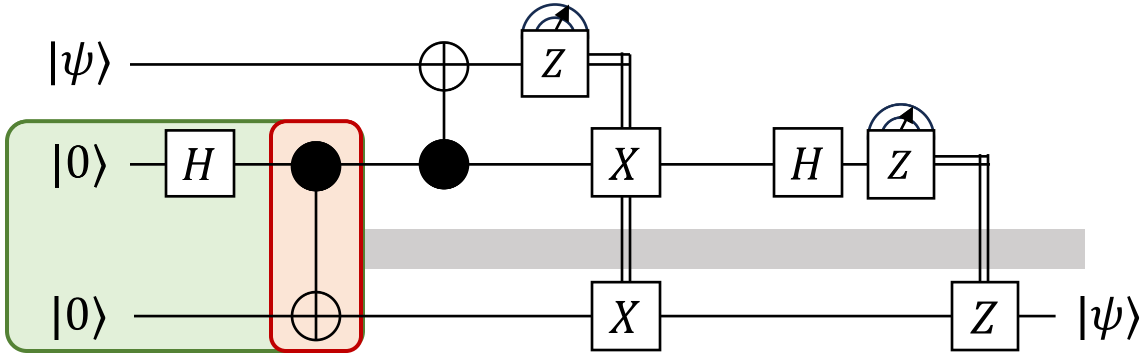
$$CX_{1,2}(X_1 \otimes I_2) = (X_1 \otimes X_2)CX_{1,2}$$




"Controls commute with each other."

# (Two-Bit) Teleportation Circuit

**Putting it all together**

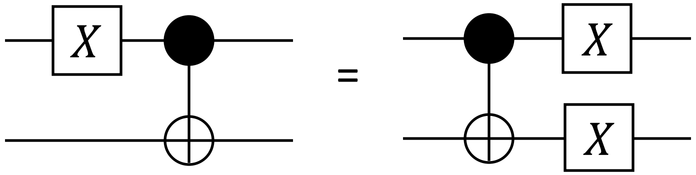



EPR State
(generate ahead of time)

This circuit works for **teleporting any arbitrary quantum state**!

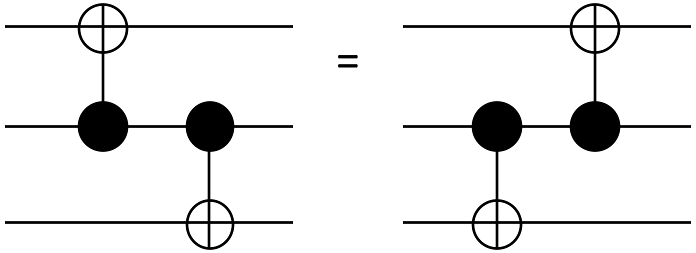
- Entanglement (EPR pair) shared prior to teleportation.
- No quantum gates between Alice and Bob.
- Classical communication (of 2 classical bits) is needed.

**More Circuit Equivalences:**



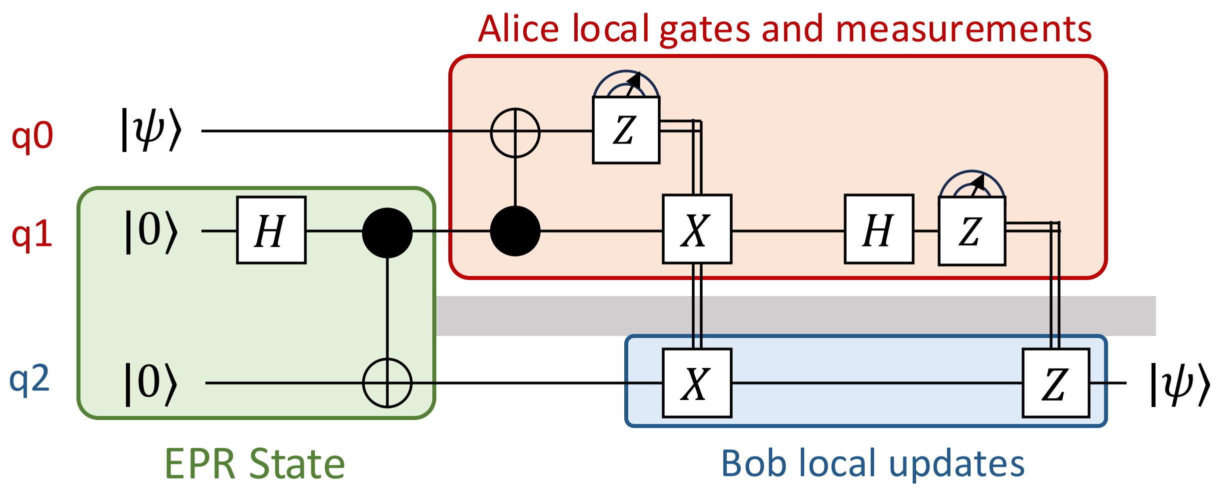

"Commute CX to the left."

$$CX_{1,2}(X_1 \otimes I_2) = (X_1 \otimes X_2)CX_{1,2}$$




"Controls commute with each other."

# Protocol Implementation




Alice's quantum register: {q0, q1}
Bob's quantum register: {q2}

Alice's classical register: {c0, c1}
Bob's classical register: {c0, c1, c2}

**Main program** for the teleportation protocol:

```
def teleportation():
    qc = QuantumCircuit(3, 3)

    # EPR generation on q1, q2
    qc.h(1)
    qc.cx(1, 2)
```

**SHARE** entangled qubits between Alice and Bob

```
    # Send q0, q1 to Alice and q2 to Bob
    alice_ids = [0, 1]
    bob_ids = [2]
    (alice_qreg, alice_creg) = qc.allocate(alice_ids, 0)
    (bob_qreg, bob_creg) = qc.allocate(bob_ids, 0)

    # Build subcircuits for Alice and Bob
    alice_subcircuit = QuantumCircuit(alice_qreg, alice_creg)
    share_creg = bob_creg + alice_creg
    shared_ids = bob_ids + alice_ids
    bob_subcircuit = QuantumCircuit(bob_qreg, share_creg)
```

**WAIT** for signal to start teleportation

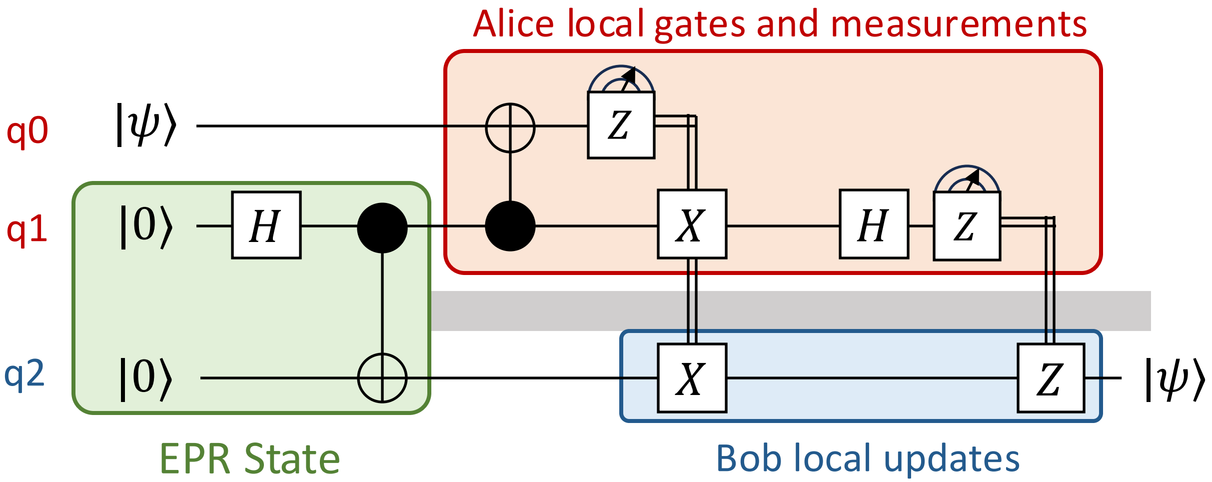
```
    # Alice performs her local operations
    alice_locals(alice_subcircuit)
    qc._append(alice_subcircuit, alice_ids, alice_ids)
```

**WAIT** for Alice's measurement signal

```
    # Bob performs his local update
    bob_updates(bob_subcircuit)
    qc._append(bob_subcircuit, bob_ids, shared_ids)

    return qc
```

# Protocol Implementation




**Classical conditional gate:**

```
# conditional gate: if all 'cbits' are 1, then perform 'gate' on 'qubits'
def conditional(self, cbits, gate, qubits):
    assert(len(cbits) > 0)
    assert(isinstance(gate, str))
    assert(gate in self.gateset)
    cond = '{}_if'.format(gate)
    conditionalGate = Gate(cond, len(qubits), None)
    self._append(conditionalGate, qubits, cbits)
    return
```

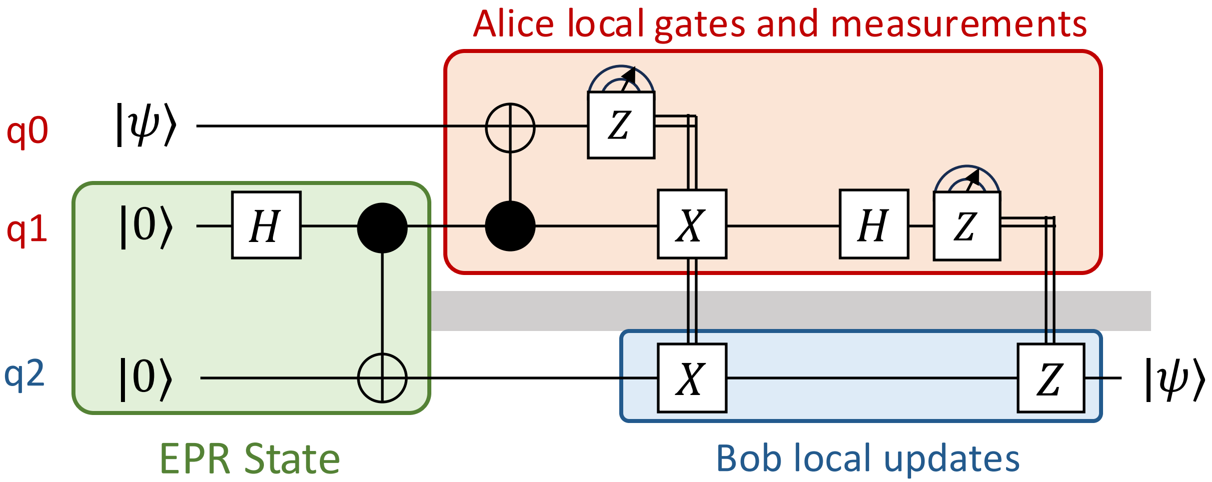
**Alice's local program:**

```
def alice_locals(alice_qc):
    # qreg: [0, 1] = [q0, q1]
    # creg: [0, 1] = [c0, c1]
    alice_qc.cx(1, 0)
    alice_qc.measure([0], [0])
    alice_qc.conditional([0], 'x', [1])
    alice_qc.h(1)
    alice_qc.measure([1], [1])
```

**Bob's local program:**

```
def bob_updates(bob_qc):
    # qreg: [0] = [q2]
    # creg: [0, 1, 2] = [c2, c0, c1]
    bob_qc.conditional([1], 'x', [0])
    bob_qc.conditional([2], 'z', [0])
```

# Protocol Implementation




**Modular program**
(with subroutines/subcircuits)

```
======<CPSC 447/547 QASM>======
Qreg: 3, Creg: 3
h qreg1 ,
cx qreg1 qreg2 ,
module qreg0 qreg1 , creg0 creg1
    Qreg: 2, Creg: 2
    cx qreg1 qreg0 ,
    measure qreg0 , creg0
    x_if qreg1 , creg0
    h qreg1 ,
    measure qreg1 , creg1
module qreg2 , creg2 creg0 creg1
    Qreg: 1, Creg: 3
    x_if qreg2 , creg1
    z_if qreg2 , creg2
===============================
```

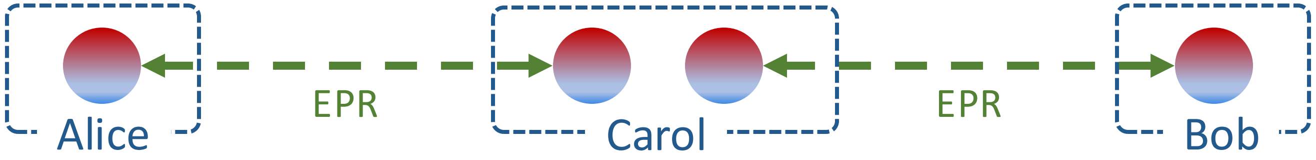
**Compiled program**

```
======<CPSC 447/547 QASM>======
Qreg: 3, Creg: 3
h qreg1 ,
cx qreg1 qreg2 ,
cx qreg1 qreg0 ,
measure qreg0 , creg0
x_if qreg1 , creg0
h qreg1 ,
measure qreg1 , creg1
x_if qreg2 , creg0
z_if qreg2 , creg1
===============================
```

**Flattened program**

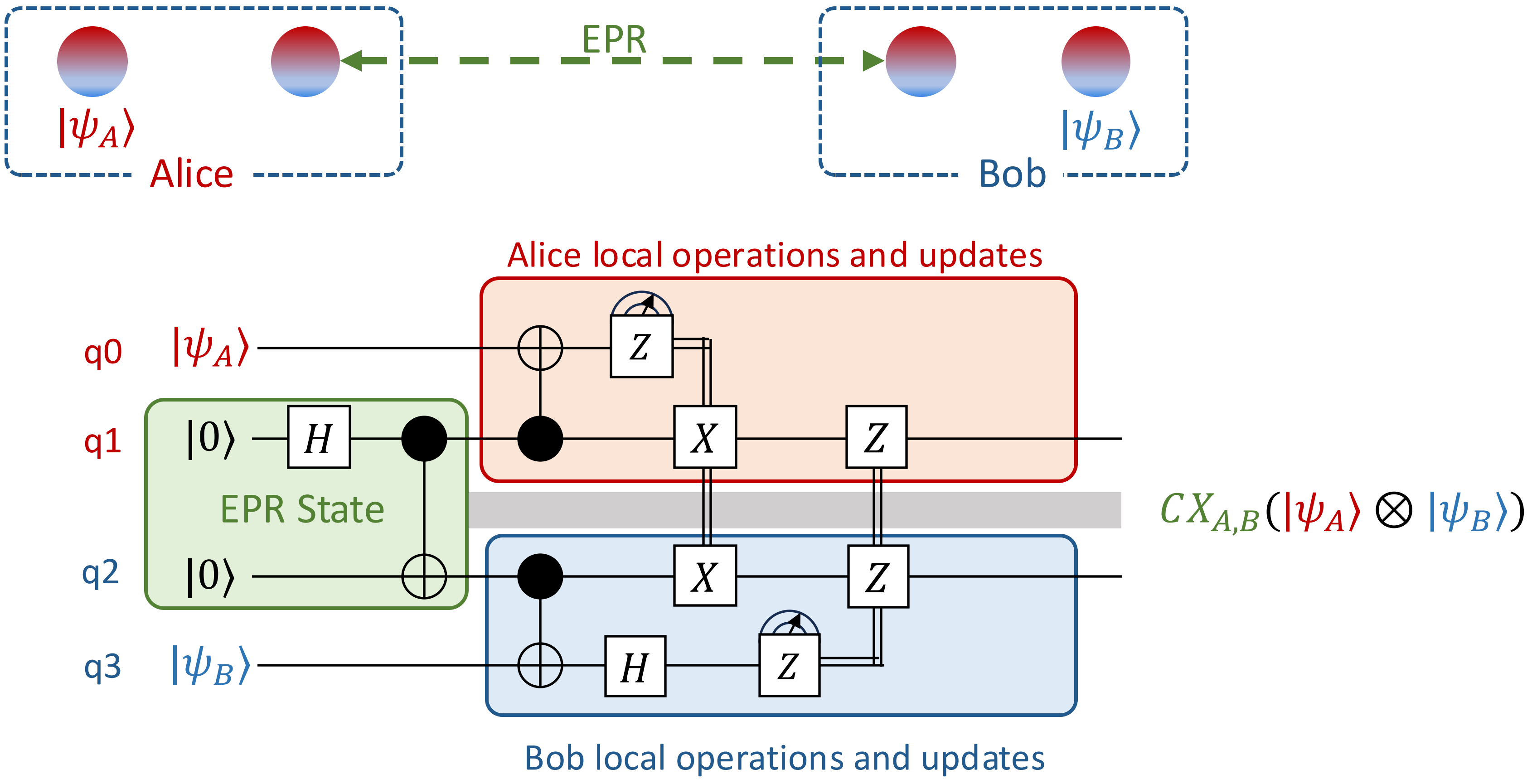
**Pre-generated EPR State:** How do Alice and Bob share entanglement in the first place, if they are far apart?
Can Alice and Bob share entanglement, despite never directly interacted with each other?

**Yes.** This is your homework problem on "entanglement swapping".



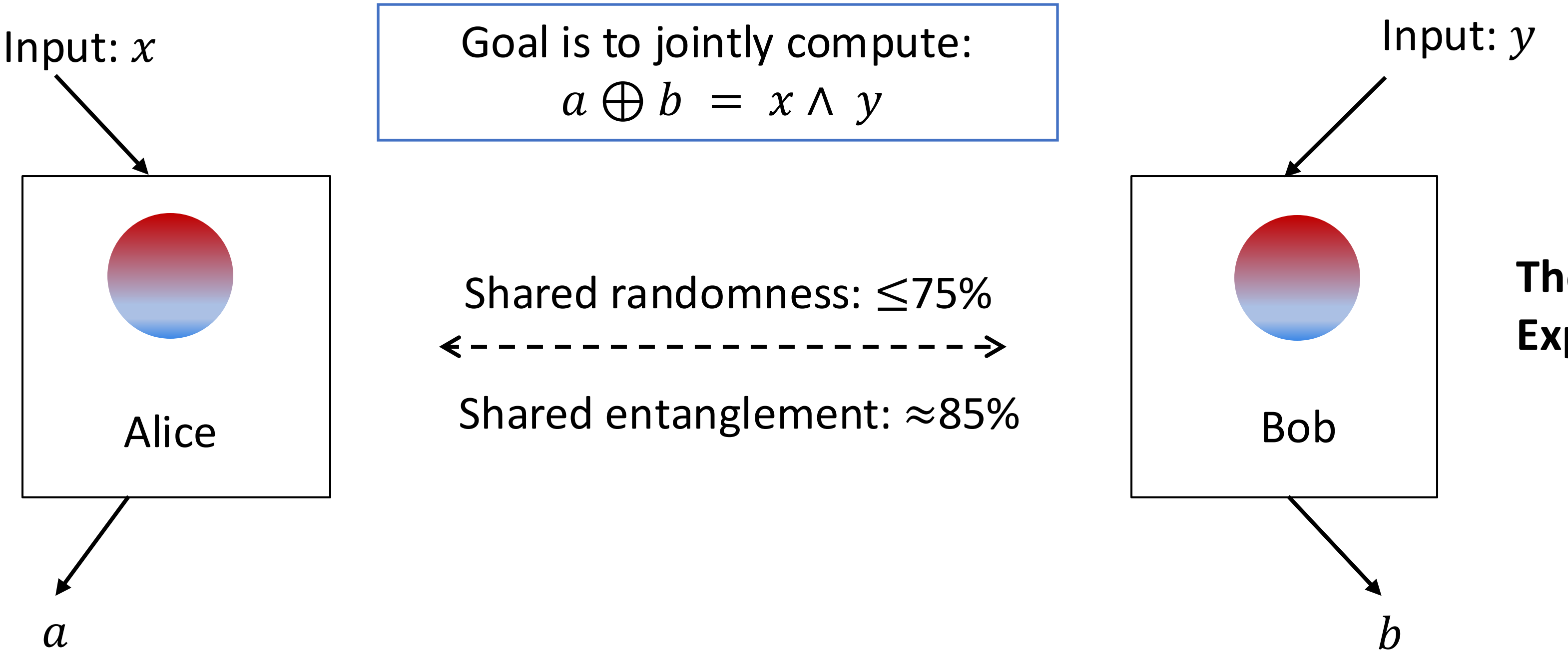

Yale

# Teleporting Gates
**Distributed Quantum Computing**



# Entanglement is stronger than classical correlation

Clauser–Horne–Shimony–Holt (CHSH) Game




**Theory**: J. Bell (1964)
**Experiment**: A. Aspect et al., early ’80s