# PHY682 Special Topics in Solid-State Physics: Quantum Information Science

Lecture time: 2:40-4:00PM Monday & Wednesday

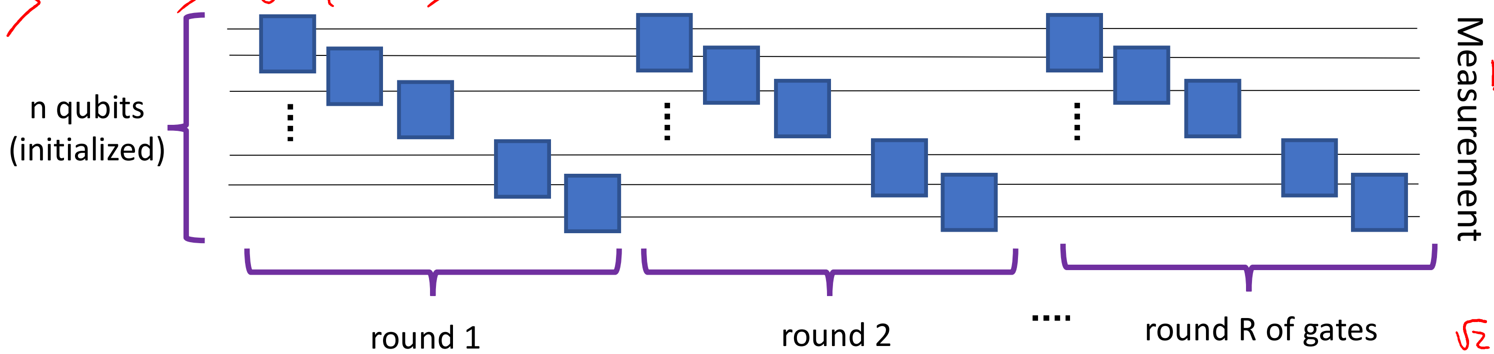
Today 9/16:

1. Reminder: Homework 2 due Sunday Sep. 20th
2. How to use Qiskit; Circuit model and Quantum gates

# Quantum computation: circuit model

$U$ $2^n \times 2^n$

$|0\cdots0\rangle \xrightarrow{U} U|0\cdots0\rangle \xrightarrow{\text{measurement}}$



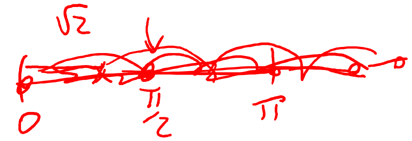

$D \sim 0/1$

$D \sim 0/1$

$\vdots$

$D \sim 0/1$

$\sqrt{2}$



- 3 steps: (1) Initialization, (2) Gate operations (3) Measurement

- Gates: a finite ("universal") set of unitary transformations are sufficient

$T^2 = S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$

$|0\rangle \to |0\rangle$

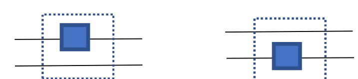
$|1\rangle \to e^{i\pi/4}|1\rangle$

approximate

  - 1-qubit gates: $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ $\quad T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$

  - 2-qubit gate: $\text{CNOT} = |0\rangle\langle 0| \otimes I + |1\rangle\langle 1| \otimes X = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix}$

Note 1-qubit gates are special case of 2-qubit gates:

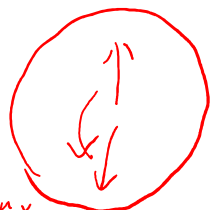


# Universal set of gates (and notations)

- Exact universality: able to decompose any unitary to a sequence of gates in the set

U = decomposed into a sequence of 1- & 2-qubit gates "exactly"

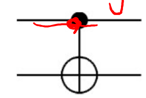
Bloch



any rotation

(i) arbitrary one-qubit rotations u3 and (ii) CNOT:

$$\mathrm{CNOT}_{12} = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{array}\right)$$



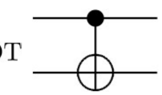
$$\boxed{u3(\theta,\phi,\lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i\lambda+i\phi}\cos(\theta/2) \end{pmatrix}}$$

$$\mathrm{CNOT}_{21} = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \end{array}\right)$$

2nd bit as control

- Approximate universality: able to approximate any unitary as close as possible

Example 1: $H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$ —[H]— $T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}$ —[T]— CNOT

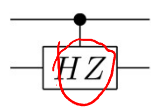


Note: S= T$^2$

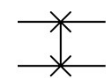
$$S = \begin{pmatrix} 1 & 0 \\ 0 & i \end{pmatrix}$$

Example 2:

$$\boxed{W} = \left(\begin{array}{cc|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & \frac{1}{\sqrt{2}} & \frac{-1}{\sqrt{2}} \\ 0 & 0 & \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{array}\right)$$



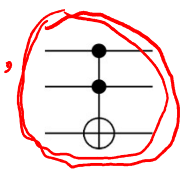
$$\boxed{\mathrm{SWAP}} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}$$



a b

b a

Example 3: $\boxed{H = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}}$, Toffoli $= \mathrm{C}^2 - X$ (Control-Control-NOT),



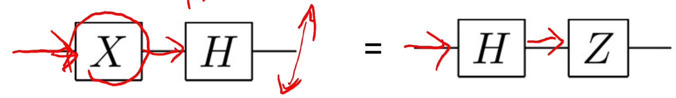
3-qubit gate

# Example gates and gate identity

- Basis transformation: X ←→ Z

$+/- \xleftrightarrow{H} 0/1$

$HX = ZH$



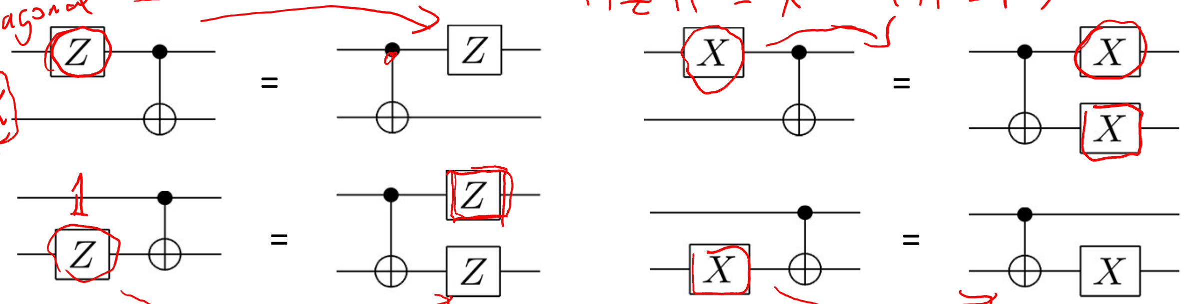
- Involving CNOT:

$HZH = X \quad (H^2 = 1)$

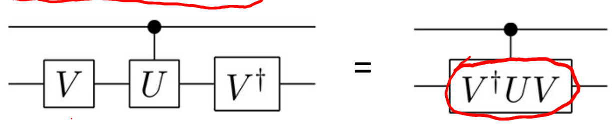
diagonal

$\mathrm{CNOT} = |0\rangle\langle 0| \otimes \mathbb{1} + |1\rangle\langle 1| \otimes X$

$\mathbb{1} \otimes Z$



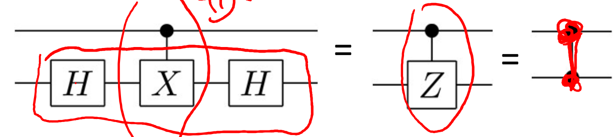
- Controlled gate:



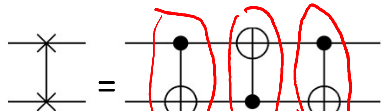
$(\mathbb{1} \otimes V^\dagger)(C\text{-}U)(\mathbb{1} \otimes V) = C\text{-}[V^\dagger U V]$

e.g.

$\mathrm{CNOT} \xrightarrow{H} CZ$

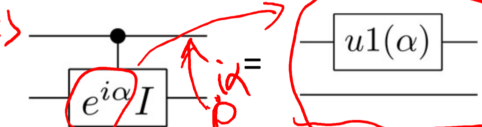


- Swap from CNOTs:



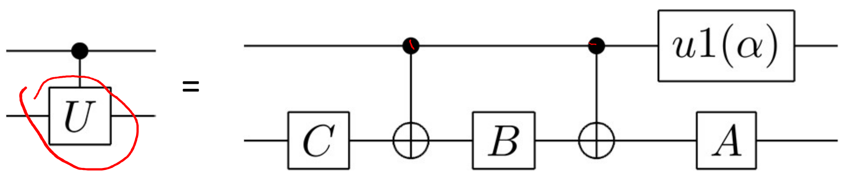
- Controlled constant phase (similar to phase kickback):

$|1\rangle$ $e^{i\alpha}$



$$u1(\alpha) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$$

# Multiqubit gates from standard set

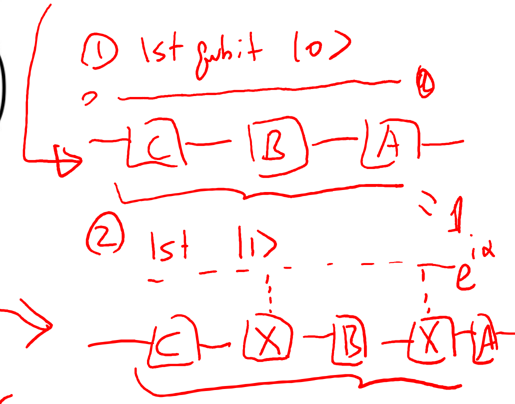



$$U = e^{i\alpha} AXBXC, \;\; ABC = I$$

$$u1(\alpha) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\alpha} \end{pmatrix}$$

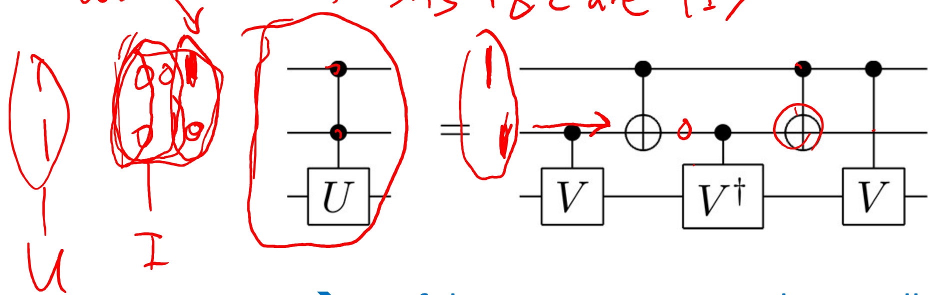
① 1st qubit |0>

② 1st |1>



➔ Useful to construct general controlled gates

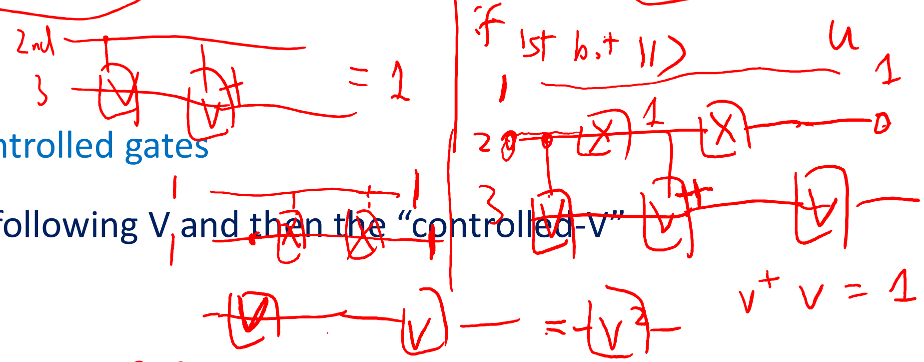
U's action if bits 1 & 2 are |1>




$$U = V^2$$

if 1st bit |0>

if 1st bit |1>



➔ Useful to construct general controlled-controlled gates

➢ For Toffoli gate: U=X, so can choose the following V and then the "controlled-V"

$$V = \frac{1-i}{2}(I + iX) \Rightarrow V^2 = X$$

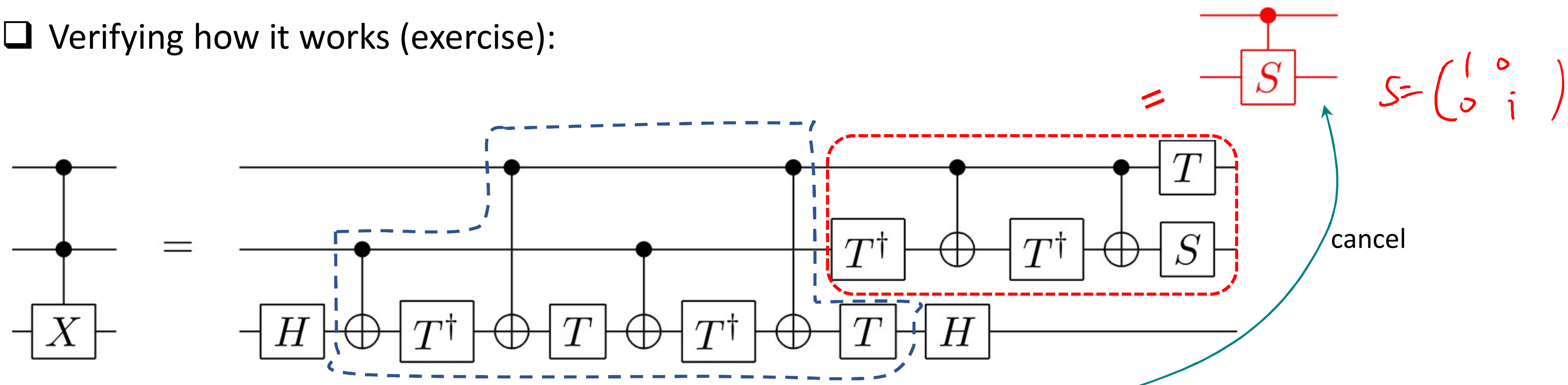
$V^{+} V = 1$

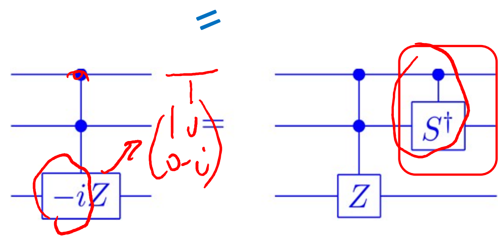
$(1+iX)^2 = 1 + 2iX + i^2 X^2 = 2iX$

This requires 8 CNOT gates. But there is a decomposition with 6 CNOT gates.

# Toffoli gate with 6 CNOTs*

- Verifying how it works (exercise):



As when first bit is zero, the action on 2& 3 is identity;
But when first bit is one, the action on 2&3 equals a controlled operation that the action on the third bit is
$(TXT^{\dagger})\ X\ (TXT^{\dagger})X = -i\ Z$

- Note that there is a proof that there must be at least 5 two-qubit gates for Toffoli

[Phys. Rev. A 88, 010304(R) (2013)]

# C^n(U) n-qubit controlled unitary*

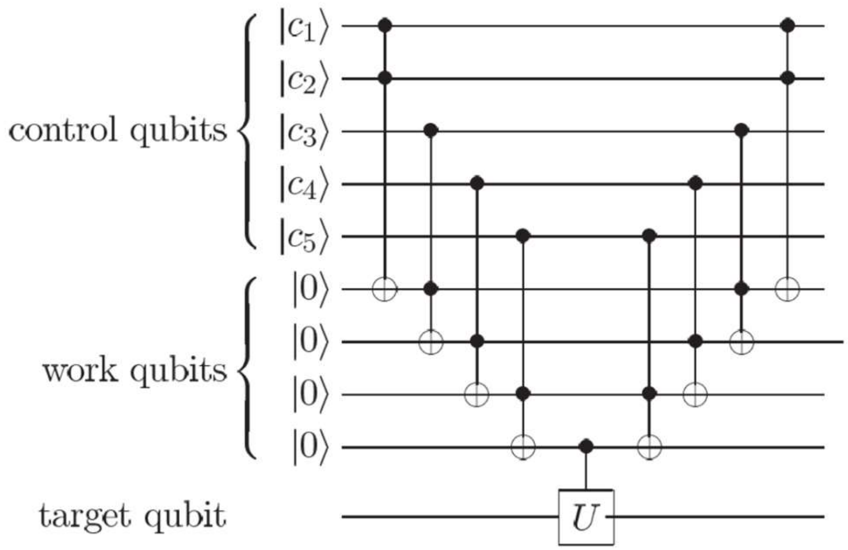



Figure 4.10. Network implementing the $C^n(U)$ operation, for the case $n = 5$.

# Qiskit gate set: single qubits

- u gates:

u3(angle1,angle2,angle3)

$$u3(\theta,\phi,\lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i\lambda+i\phi}\cos(\theta/2) \end{pmatrix}$$

u2(angle1,angle2)

$$u2(\phi,\lambda) = u3(\pi/2,\phi,\lambda) = \frac{1}{\sqrt{2}}\begin{pmatrix} 1 & -e^{i\lambda} \\ e^{i\phi} & e^{i(\phi+\lambda)} \end{pmatrix}$$

u1(angle1)

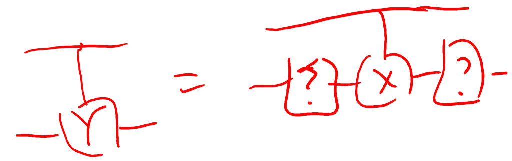
$$u1(\lambda) = u3(0,0,\lambda) = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\lambda} \end{pmatrix}$$

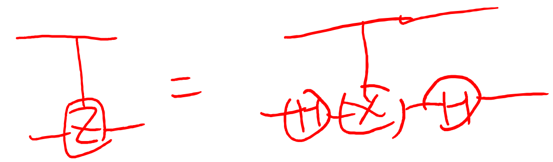
- Identity gate: iden = ~~u0(1)~~ u1(0)

$$u0(\delta) = u3(0,0,0)$$

- Pauli gates: x, y, z; Hadamard: h;
  phase gate: s and its inverse: sdg, ← $s^{+}$
  T gate: t and its inverse: tdg,
  X,Y,Z rotations: rx(angle, qubit), ry and rz

# Qiskit gate set: multiple qubits

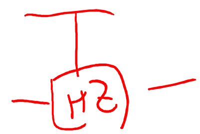



Controlled-NOT gate: cx(control, target)

Controlled-Y and -Z gates: cy(control, target), cz(control,target)

Controlled-Hadamard gate: ch(control, target)



Controlled-Rotation gate: crz(angle,control, target), and crx, cry

Controlled-U1 gate: cu1(angle,control, target) [useful in QFT]

Controlled-U3 gate: cu3(angle1,angle2,angle3,control, target)

Swap gate: swap(qubit1, qubit2)

Toffoli gate: ccx(control1,control2, target)

Controlled swap gate (Fredkin Gate): cswap(control, qubit2,qubit3)

# Other operations

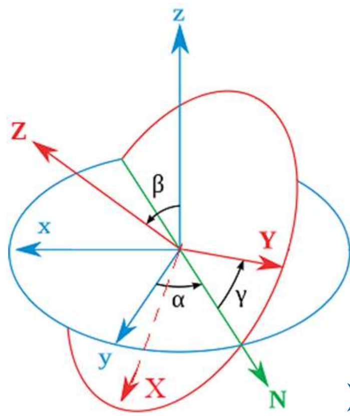
Measurement: measure(qubit, classical outcome)

Reset qubit to 0: reset(qubit)

Conditional operations (on classical outcome):

e.g.
```
qc.measure(q,c)
qc.x(q[0]).c_if(c,0) #apply X to q[0] if c is 0
```

# Comment: Euler rotation and u3 gate




$$U(\alpha,\beta,\gamma) = R_Z(\gamma) R_N(\beta) R_z(\alpha)$$

$$R_Z(\gamma) = R_N(\beta) R_z(\gamma) R_N(\beta)^{-1}$$
$$R_N(\beta) = R_z(\alpha) R_y(\beta) R_z(\alpha)^{-1}$$

$$U(\alpha,\beta,\gamma) = R_z(\alpha) R_y(\beta) R_z(\gamma)$$

- We also include an arbitrary overall phase in the unitary group and for one qubit: $e^{1\delta} U(\alpha,\beta,\gamma) = e^{i\delta} R_z(\alpha) R_y(\beta) R_z(\gamma)$

unitary group
$U(2)$ real

$$= e^{i\delta} \begin{pmatrix} e^{-i\alpha/2} & 0 \\ 0 & e^{i\alpha/2} \end{pmatrix} \begin{pmatrix} \cos(\beta/2) & -\sin(\beta/2) \\ \sin(\beta/2) & \cos(\beta/2) \end{pmatrix} \begin{pmatrix} e^{-i\gamma/2} & 0 \\ 0 & e^{i\gamma/2} \end{pmatrix} \rightarrow = \begin{pmatrix} + \end{pmatrix}$$

$e^{-i\beta \frac{\sigma_y}{2}}$ $\quad$ $e^{-i\gamma \frac{\sigma_z}{2}}$

- IBM's u3 gate by taking

$$\delta = \frac{\alpha+\gamma}{2}, \; \beta = \theta, \; \alpha = \phi, \; \gamma = \lambda$$

$$u3(\theta,\phi,\lambda) = \begin{pmatrix} \cos(\theta/2) & -e^{i\lambda}\sin(\theta/2) \\ e^{i\phi}\sin(\theta/2) & e^{i\lambda+i\phi}\cos(\theta/2) \end{pmatrix}$$

# Measurement

- In IBM, Google or Rigetti, the measurement is done in 0/1 basis, e.g.
  >>> qc.measure(q,c)

  In order to measure in arbitrary basis defined by ξ(0)/ ξ(1):

$$|\xi_+\rangle = U_\xi|0\rangle,\ |\xi_-\rangle = U_\xi|1\rangle$$

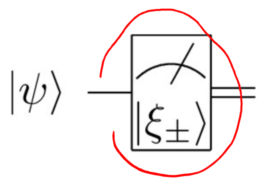
$P_0 = |\langle 0|\psi\rangle|^2$

$P_1 = |\langle 1|\psi\rangle|^2$

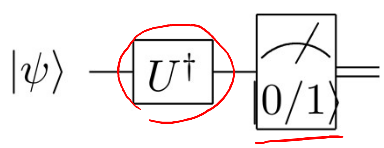
$\langle +|$

$\langle -|$

  - We first apply inverse of U (i.e. U†) before measuring in 0/1 basis




≡

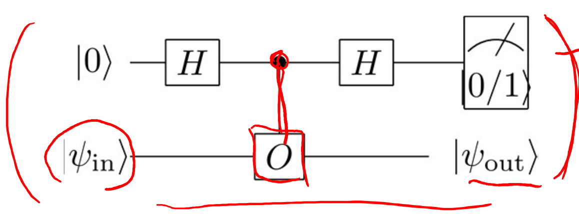



[If one cares about the exact post-measurement state being in the ξ basis, one should apply U after the measurement to undo the basis change U† earlier]

$\langle \xi_+| = \langle 0|U_\xi^\dagger$

$\langle \xi_-| = \langle 1|U_\xi^\dagger$

- How to measure a single-qubit operator O (unitary and Hermitian, thus $O^2=I$) and leave the output in the eigenstate?




leave exercise

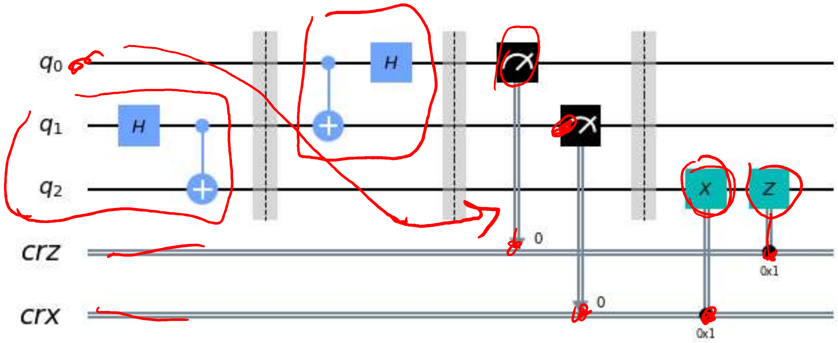
# Principle of Deferred Measurement

- ❑ Measurement can be moved to the end of the circuit; if measurement results are used to classically control some operation, it can be replaced by controlled operation

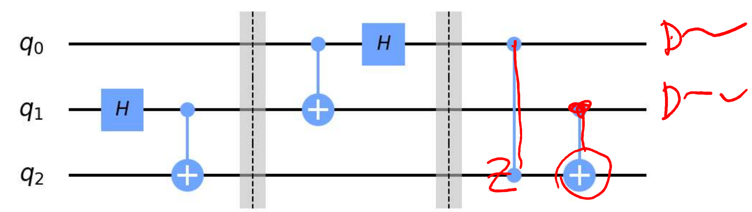
➔ Teleportation circuit is an example:

→ Bell-state measurement

Bell state




≡



# Comments: Clifford gates and Gottesman-Knill no go theorem*

- Clifford gates $U_C$ are those that transform a Pauli product σ to another Pauli product operator σ':

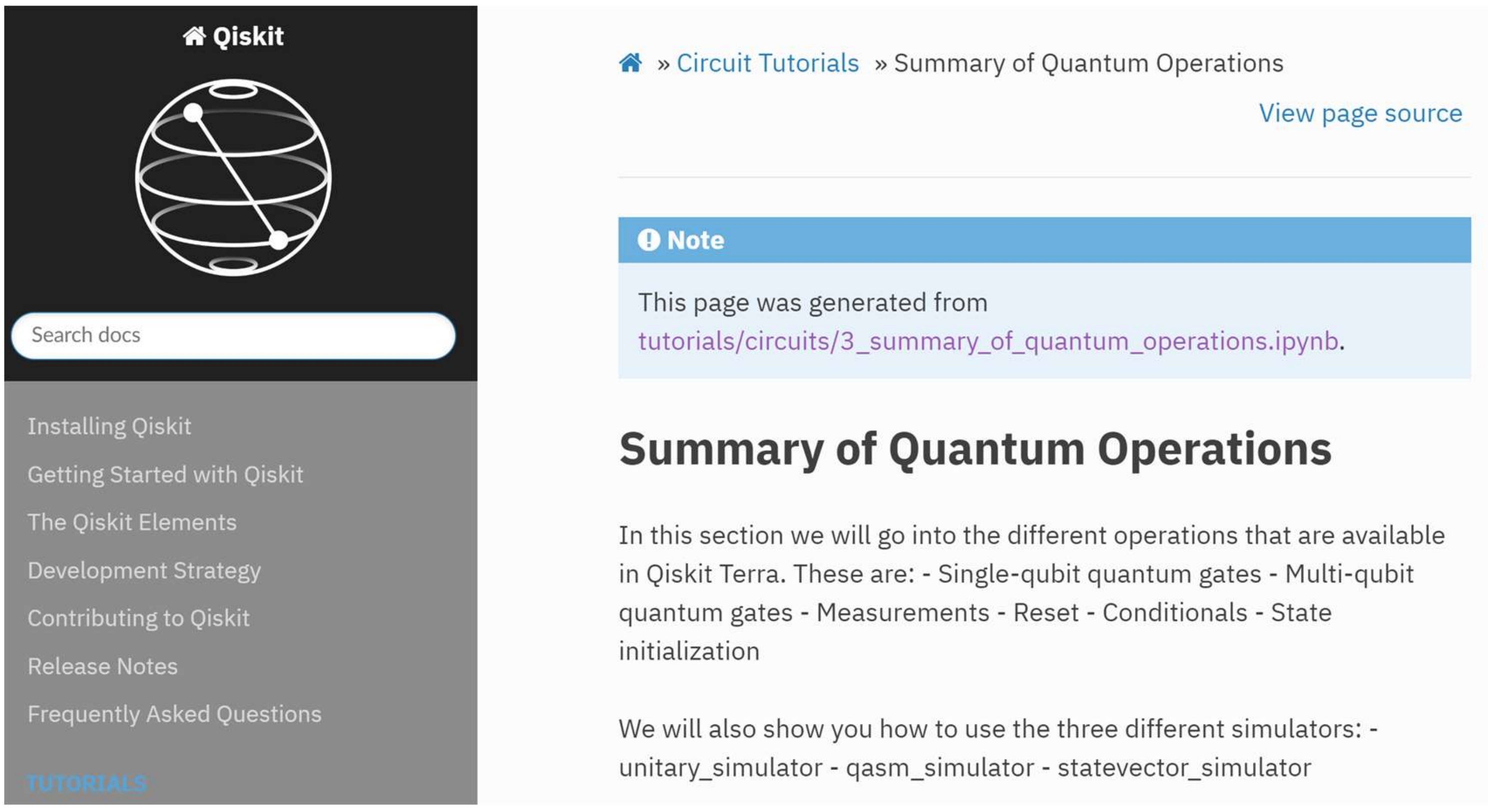
$$U_C \, \sigma \, U_C^{\dagger} = \sigma'$$

- Theorem 10.7: (Gottesman–Knill theorem) Suppose a quantum computation is performed which involves only the following elements: **state preparations in the computational basis, Hadamard gates, phase gates, controlled-NOT gates, Pauli gates, and measurements of observables in the Pauli group** (which includes measurement in the computational basis as a special case), together with the possibility of classical control conditioned on the outcome of such measurements.

➔ Such a computation may be efficiently simulated on a classical computer.

# Qiskit tutorial: summary of Q operations

https://qiskit.org/documentation/tutorials/circuits/3_summary_of_quantum_operations.html



Qiskit

Search docs

Installing Qiskit

Getting Started with Qiskit

The Qiskit Elements

Development Strategy

Contributing to Qiskit

Release Notes

Frequently Asked Questions

TUTORIALS

» Circuit Tutorials » Summary of Quantum Operations

View page source

**Note**

This page was generated from tutorials/circuits/3_summary_of_quantum_operations.ipynb.

## Summary of Quantum Operations

In this section we will go into the different operations that are available in Qiskit Terra. These are: - Single-qubit quantum gates - Multi-qubit quantum gates - Measurements - Reset - Conditionals - State initialization

We will also show you how to use the three different simulators: - unitary_simulator - qasm_simulator - statevector_simulator

Do Notebook on gates

Do Poll

Do you now feel comfortable with running the Ipython/Jupyter Notebook?

(a) Yes
(b) I may need to more time; but I am optimistic
(c) No

Which do you prefer when you need to run Qiskit notebooks?

(a) Install Python & Qiskit packages on my laptop/desktop
(b) Use Cocalc.com
(c) I am not yet sure