Quantum Phase Estimation

Bernice, Yun Ting, Zechu

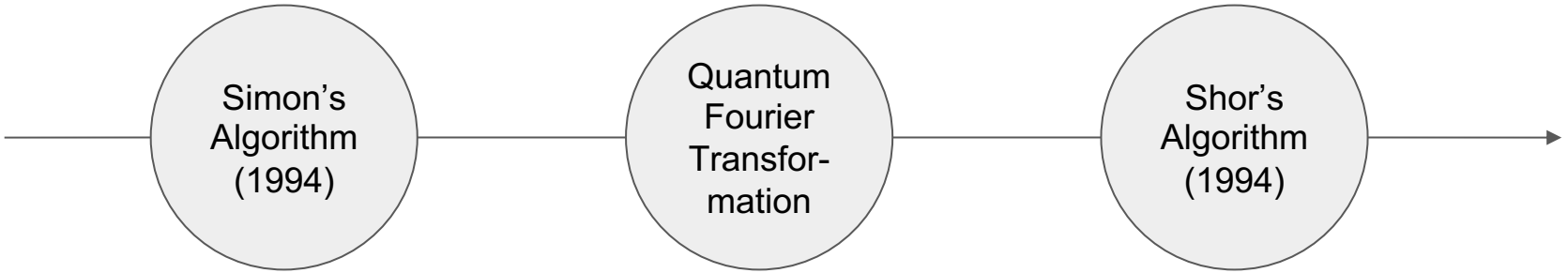
Table of Content

Our colour code (Slide background):

Concept check

Quantum Phase Estimation

Context


Simon's Algorithm (1994)
Quantum Fourier Transfor-mation
Shor's Algorithm (1994)

Goal

Problem:

Given a unitary operator U and its **eigenvector** $|u\rangle$, the algorithm estimates the value of the phase θ of the associated **eigenvalue**.

Significance:

Phase estimation

- **Subroutine** in other quantum algorithms
  - Shor's algorithm (prime factorisation)
  - Quantum algorithm for linear systems of equations.
- Runs efficiently on quantum computers
  - Currently infeasible on classical computers → **exponential speed-up**!

What is an eigenvalue and eigenvector?

Suppose A is a matrix, |v⟩ is a vector, and α is a scalar.

$$A|v\rangle = \alpha|v\rangle$$

Eigenvector of A

Eigenvalue of A

We want to find this!

Then |v⟩ is **an** eigenvector of A, and α is **an** eigenvalue of A.

Task

Similarly, let U be an unitary operator, |u⟩ be an eigenstate (eigenvector) of U:

$$U|u\rangle = c|u\rangle \text{ where } c \in \mathbb{C}$$

$$\begin{aligned} c &= |z|e^{i\theta} \\ &= e^{i\theta} \text{ where } \theta \in [0, 2\pi] \\ &= e^{2\pi i\phi} \text{ where } \phi \in [0, 1] \end{aligned}$$

(|z| is 1 since U is a **unitary** operator.)

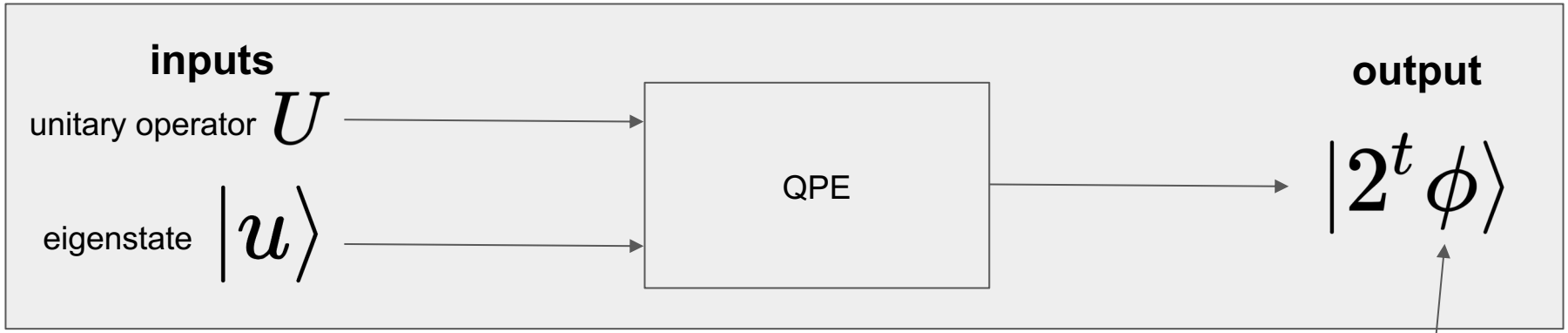
$$U|u\rangle = e^{2\pi i\phi}|u\rangle \text{ where } \phi \in [0, 1]$$

We want to find ϕ!

$U|u\rangle = e^{2\pi i\phi}|u\rangle$ where $\phi \in [0, 1]$

QPE Blackbox

TASK: Find ϕ



Assumption 1:
ϕ can be **perfectly/exactly expressed** using t bits as

$$\phi = 0.\phi_1\phi_2\phi_3\ldots\phi_t$$

Note that this notation is a binary integer:

$$2^t\phi = \phi_1\phi_2\phi_3\ldots\phi_t$$

In QPE, we receive output as such, and divide output by 2^t to attain our phase:

$$\phi = 0.\phi_1\phi_2\phi_3\ldots\phi_t$$

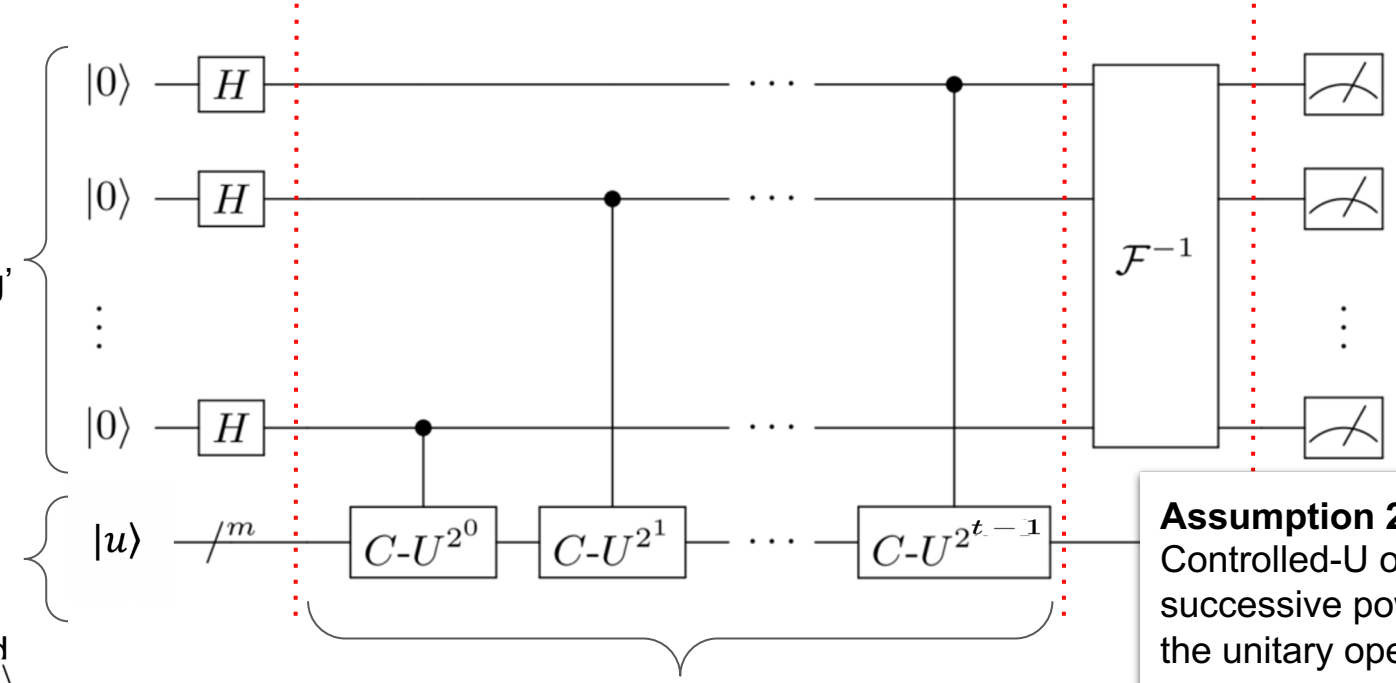
Quantum Circuit Implementation

Step 1:
Create superposition of counting qubits using Hadamard gates

Step 2:
Apply controlled-U operators

Step 3:
Apply Inverse Quantum Fourier Transform

Step 4:
Measurement of first register



First register:
t 'counting' qubits

Second register:
As many as needed to store $|u\rangle$

t number of C-U gates

Assumption 2:
Controlled-U operators raised to successive powers of 2 exist for the unitary operation U

Promise: Whole circuit is unitary

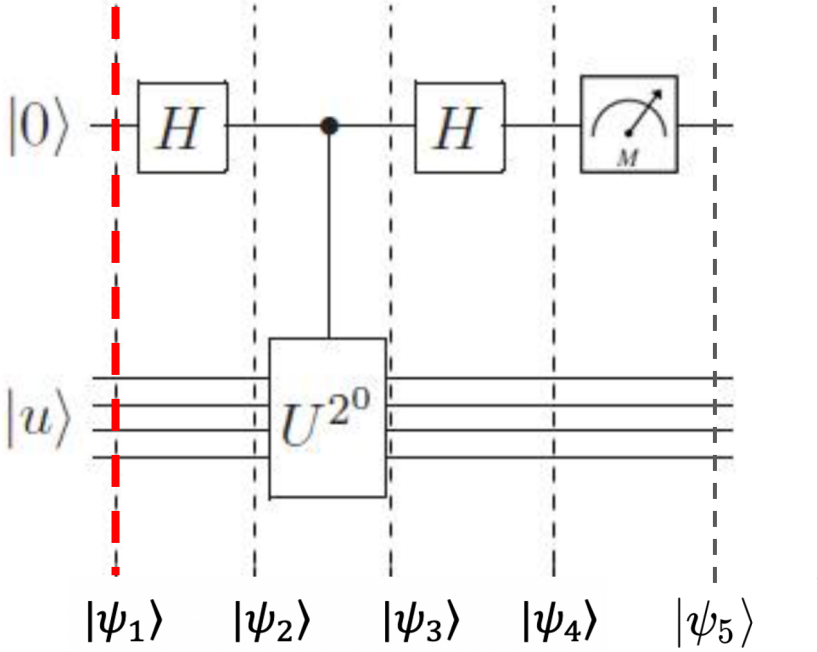
Let’s run the algorithm with an example!

Special case: single qubit (t = 1)

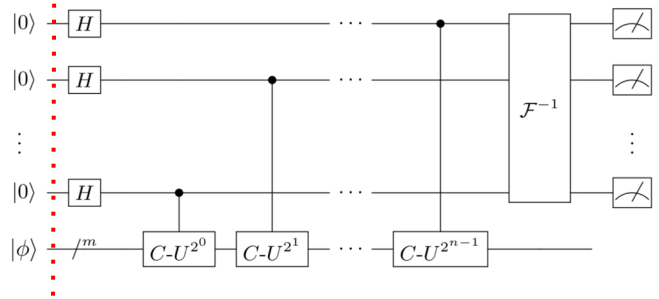
$$U|u\rangle = e^{2\pi i\phi}|u\rangle \text{ where } \phi \in [0, 1]$$

Suppose ϕ can be **perfectly** expressed using **one bit:**

$$\phi = 0.\phi_1$$



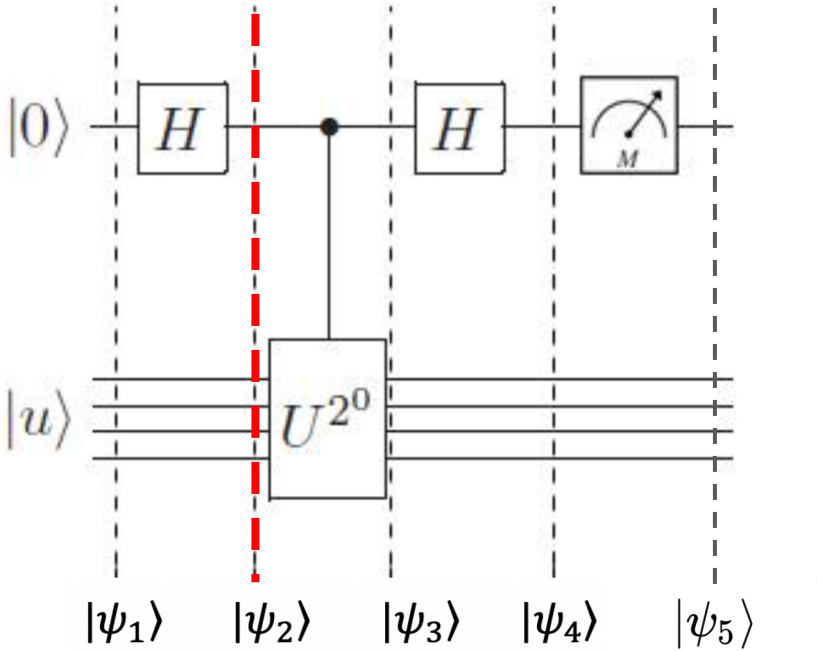
Full circuit:



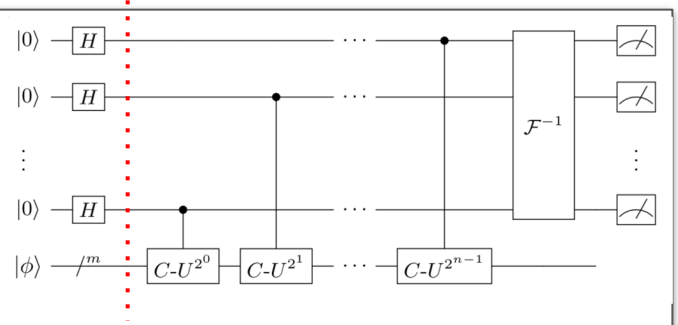
Step 0: Prepare initial state

Initial state:

$$|\psi_1\rangle = |0\rangle \otimes |u\rangle$$



Full circuit:



Step 1: Create superposition using Hadamard gate

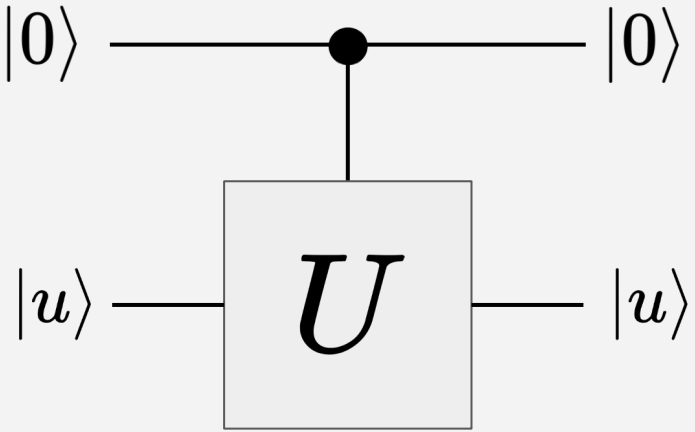
$$
\begin{aligned}
|\psi_2\rangle &= (H \otimes I)(|0\rangle \otimes |u\rangle) \\
&= (H \otimes |0\rangle) \otimes (I \otimes |u\rangle) \\
&= \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |u\rangle
\end{aligned}
$$

SUPERPOSITION!

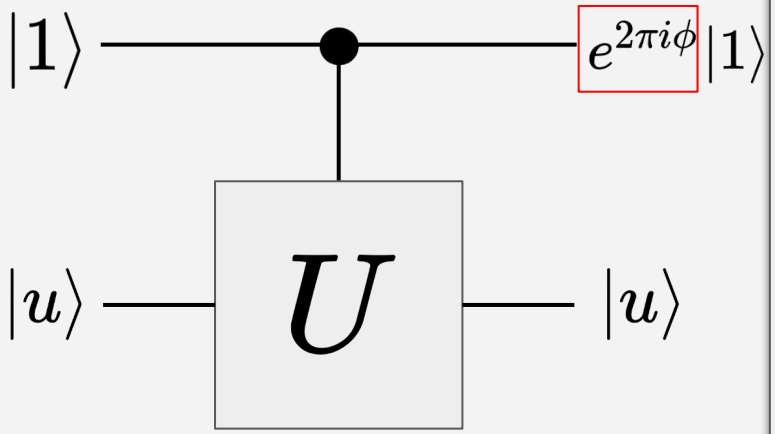
What is a C-U gate?

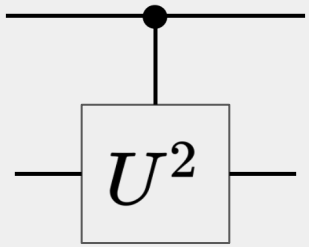
Controlled-U gate (an oracle)

$$|0\rangle \otimes |u\rangle \rightarrow |0\rangle \otimes |u\rangle$$

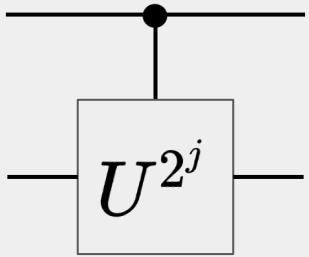


$$\begin{aligned} |1\rangle \otimes |u\rangle &\rightarrow |1\rangle \otimes U|u\rangle \\ &= |1\rangle \otimes e^{2\pi i\phi}|u\rangle \\ &= e^{2\pi i\phi}|1\rangle \otimes |u\rangle \end{aligned}$$

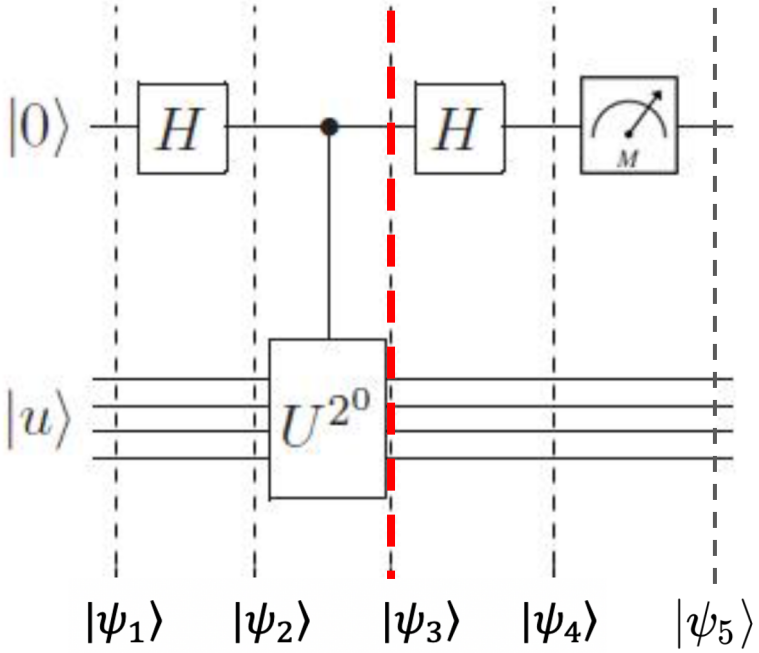




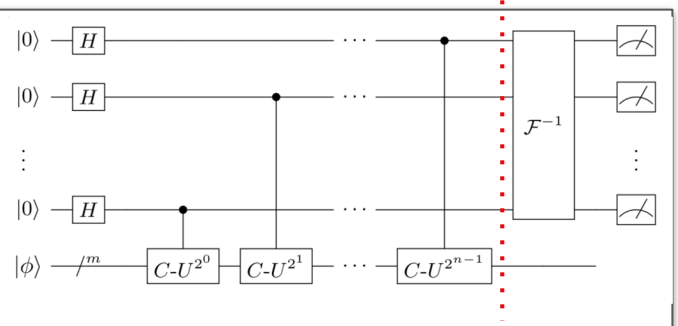
$$\begin{aligned} |1\rangle \otimes |u\rangle &\rightarrow |1\rangle \otimes U^2|u\rangle \\ &= |1\rangle \otimes e^{2\pi i 2\phi}|u\rangle \\ &= e^{2\pi i 2\phi}|1\rangle \otimes |u\rangle \end{aligned}$$



$$\begin{aligned} |1\rangle \otimes |u\rangle &\rightarrow |1\rangle \otimes U^{2^j}|u\rangle \\ &= |1\rangle \otimes e^{2\pi i 2^j \phi}|u\rangle \\ &= e^{2\pi i 2^j \phi}|1\rangle \otimes |u\rangle \end{aligned}$$



Full circuit:



Step 2: Apply controlled-U operator

$$
\begin{aligned}
|\psi_3\rangle &= CU|\psi_2\rangle \\
&= CU[\tfrac{1}{\sqrt{2}}(|0\rangle + |1\rangle) \otimes |u\rangle] \\
&= \tfrac{1}{\sqrt{2}} CU(|0\rangle \otimes |u\rangle + |1\rangle \otimes |u\rangle) \\
&= \tfrac{1}{\sqrt{2}} (CU|0\rangle \otimes |u\rangle + CU|1\rangle \otimes |u\rangle) \\
&= \tfrac{1}{\sqrt{2}} (|0\rangle \otimes |u\rangle + e^{2\pi i\phi}|1\rangle \otimes |u\rangle) \\
&= \tfrac{1}{\sqrt{2}} (|0\rangle + \boxed{e^{2\pi i\phi}}|1\rangle) \otimes |u\rangle
\end{aligned}
$$

What is Quantum Fourier Transform (QFT)?

Quantum Fourier Transform is a key component for quantum factoring and many quantum algorithms.

QFT: transform a single quantum state into a superposition of states with some phases that correspond to the input state.

In our algorithm, we use the **inverse** Quantum Fourier Transformation

Inverse-QFT: **we attain a single state from the superposition state for each counting qubit**

Quantum Fourier Transform Definition

QFT acting on quantum state |*x*> can be expressed as:

$$\mathrm{QFT} : |x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{xk} |k\rangle.$$

where vector $N = 2^t$, $\omega_N = e^{2\pi i / N}$

$$w_N^{xk} = e^{2\pi i x k / N}$$

$$\mathrm{QFT} : |x\rangle \mapsto \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega_N^{xk} |k\rangle.$$

Definition

QFT: single-qubit state

A QFT operator acting on a **single qubit state** is exactly the same result as applying the **Hadamard operator** on a single qubit state.

When t=1, N=2: $U_{QFT}|x\rangle = \frac{1}{\sqrt{2}}(\alpha e^{2\pi i(\frac{0\times 0}{2})}|0\rangle + (\beta e^{2\pi i(\frac{1\times 0}{2})}|1\rangle)$ where $|x\rangle = \alpha|0\rangle + \beta|1\rangle$

$$+ \frac{1}{\sqrt{2}}(\alpha e^{2\pi i(\frac{0\times 1}{2})}|0\rangle + (\beta e^{2\pi i(\frac{1\times 1}{2})}|1\rangle)$$

$$= \frac{1}{\sqrt{2}}(\alpha+\beta)|0\rangle + \frac{1}{\sqrt{2}}(\alpha-\beta)|1\rangle$$

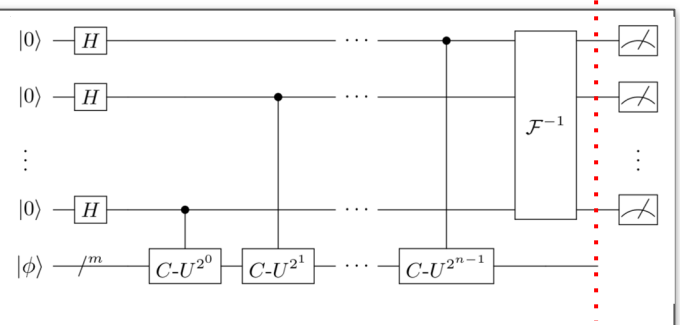
I.E. In Dimension 2

$$|0\rangle + |1\rangle \xrightarrow{F} |0\rangle + |1\rangle$$

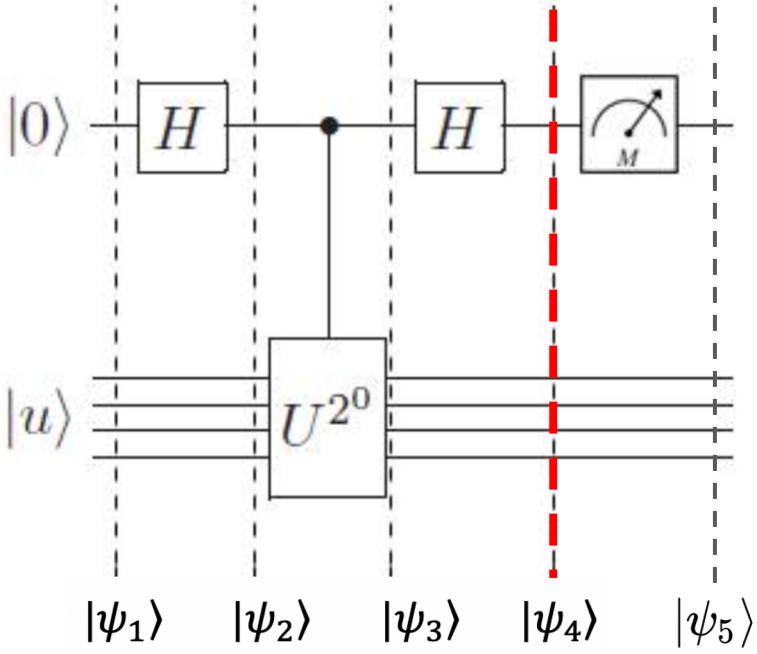
$$|0\rangle - |1\rangle \xrightarrow{F} |0\rangle - |1\rangle$$

$$\boldsymbol{H} = \frac{1}{\sqrt{2}}\begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

Full circuit:



Step 3: Apply Inverse QFT



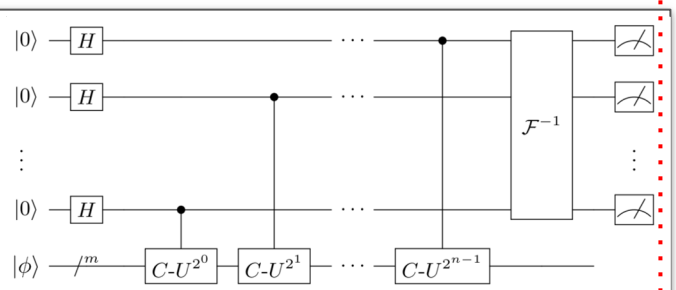
$$
\begin{aligned}
|\psi_4\rangle &= (H \otimes I)|\psi_3\rangle \\
&= \frac{1}{\sqrt{2}}(H \otimes I)[(|0\rangle + e^{2\pi i\phi}|1\rangle) \otimes |u\rangle] \\
&= \frac{1}{\sqrt{2}}[H \otimes (|0\rangle + e^{2\pi i\phi}|1\rangle)] \otimes (I \otimes |u\rangle) \\
&= \frac{1}{\sqrt{2}}\frac{1}{\sqrt{2}}[(1 + e^{2\pi i\phi})|0\rangle + (1 - e^{2\pi i\phi})|1\rangle] \otimes |u\rangle \\
&= \frac{1}{2}[(1 + e^{2\pi i\phi})|0\rangle + (1 - e^{2\pi i\phi})|1\rangle] \otimes |u\rangle
\end{aligned}
$$

$$\phi = 0.\phi_1 = \frac{\phi_1}{2^1}$$

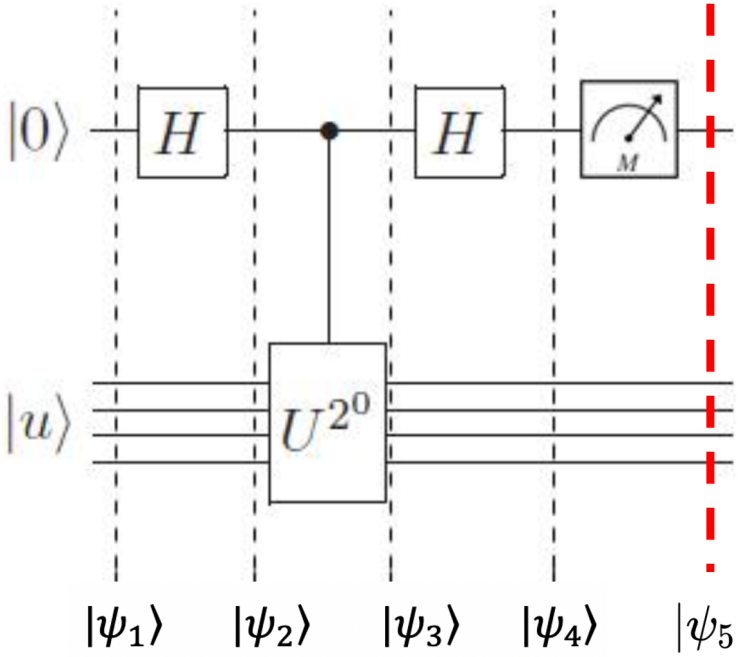
$$\frac{1}{2}(1 + e^{2\pi i\frac{0}{2}})|0\rangle = \frac{1}{2}(2)|0\rangle = |0\rangle$$

$$\frac{1}{2}(1 - e^{2\pi i\frac{1}{2}})|1\rangle = \frac{1}{2}(2)|1\rangle = |1\rangle$$

Full circuit:



Step 4: Take **measurement** from top register in **computational basis**

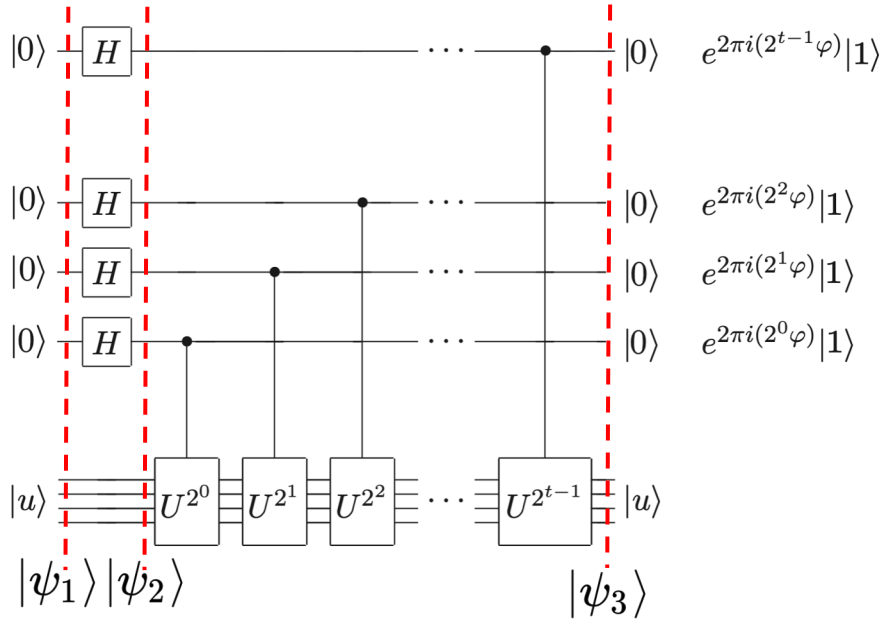


$$|\psi_{5,0}\rangle = \frac{A_0|\psi_4\rangle}{\sqrt{p(0|\psi_4)}} = \frac{|0\rangle\langle 0|\frac{1}{2}[(1+e^{2\pi i\phi})|0\rangle+(1-e^{2\pi i\phi})|1\rangle]}{\sqrt{p(0|\psi_4)}} = \frac{|0\rangle\frac{1}{2}(1+e^{2\pi i\phi})}{\sqrt{p(0|\psi_4)}} = \frac{|0\rangle\frac{1}{2}(2)}{1} = |0\rangle$$

$$|\psi_{5,1}\rangle = \frac{A_1|\psi_4\rangle}{\sqrt{p(1|\psi_4)}} = \frac{|1\rangle\langle 1|\frac{1}{2}[(1+e^{2\pi i\phi})|0\rangle+(1-e^{2\pi i\phi})|1\rangle]}{\sqrt{p(1|\psi_4)}} = \frac{|1\rangle\frac{1}{2}(1-e^{2\pi i\phi})}{\sqrt{p(1|\psi_4)}} = \frac{|1\rangle\frac{1}{2}(2)}{1} = |1\rangle$$

Updated state of knowledge

Generalise to t qubits



Initial state:

$$|\psi_1\rangle = |0\rangle^{\otimes t} \otimes |u\rangle$$

Step 1:

Hadamard gate is used to create a **superposition** of **all** the counting qubits with an equal probability of attaining |0> or |1>

$$|\psi_2\rangle = \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)^{\otimes t} \otimes |u\rangle$$

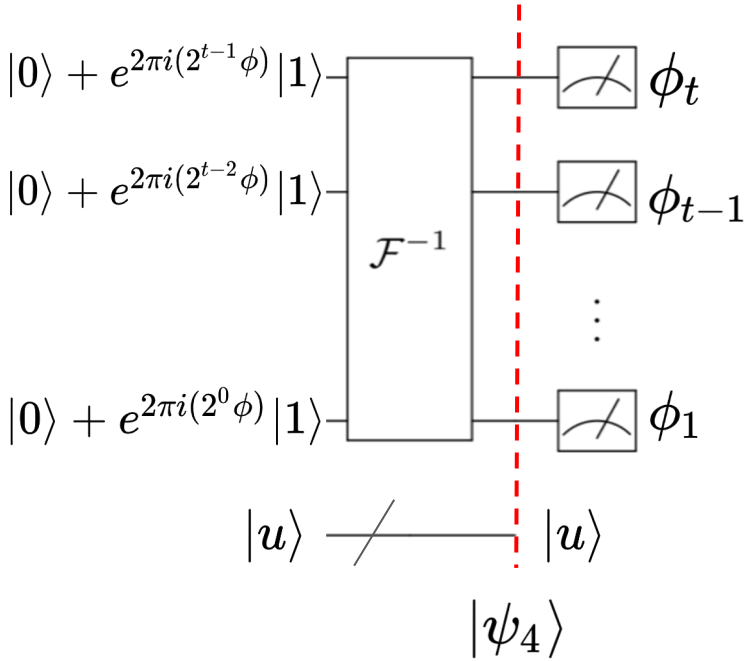
Step 2:

Controlled-U gates are used to move information to phase, we attain:

$$|\psi_3\rangle = \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} |j\rangle U^j |u\rangle$$

$$= \frac{1}{\sqrt{2^t}} \sum_{j=0}^{2^t-1} e^{2\pi i j \phi_u} |j\rangle |u\rangle$$

Generalise to t qubits



Step 3:

Apply the **inverse quantum Fourier transformation**, which takes the superposition states and returns a single state

$$|\psi_4\rangle = |2^t\phi\rangle \otimes |u\rangle$$

Only if Assumption 1 holds

*1-qubit example

$\frac{1}{2}[(1+e^{2\pi i\phi})|0\rangle + (1-e^{2\pi i\phi})|1\rangle] \otimes |u\rangle$

If we get 1, we obtain $\frac{1}{2}2|1\rangle = |1\rangle$

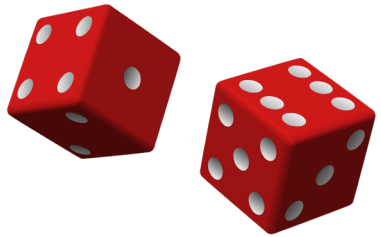
Step 4:

Measure the first register and we obtain:

Output $= \phi_1\phi_2\ldots\phi_t = 2^t\phi$

$$\phi = 0.\phi_1\phi_2\ldots\phi_t$$

Probabilistic or Deterministic?



We obtain a resulting state $|\psi_{5,outcome}\rangle$ with a probability associated with it.
It is the probability that the algorithm yields the correct estimation.

In our special case, we made **assumption 1:**
phase ϕ can be perfectly expressed using a **finite string of bits**, i.e. $\phi = 0.\phi_1\phi_2\ldots\phi_t$

Implication: measurement of the qubits yields the bits $\phi_1\phi_2\cdots\phi_t$
deterministically (probability = 1).

What about general cases where ϕ is arbitrary?

What about general cases where **ϕ is arbitrary**?
Inverse quantum Fourier transform produces a **superposition of t-bit strings**.

Implication: measurement of the output qubits yields a **probabilistic** outcome where **there is a high probability of obtaining the correct estimation**

$$\phi \approx 0.\phi_1 \phi_2 \ldots \phi_t$$

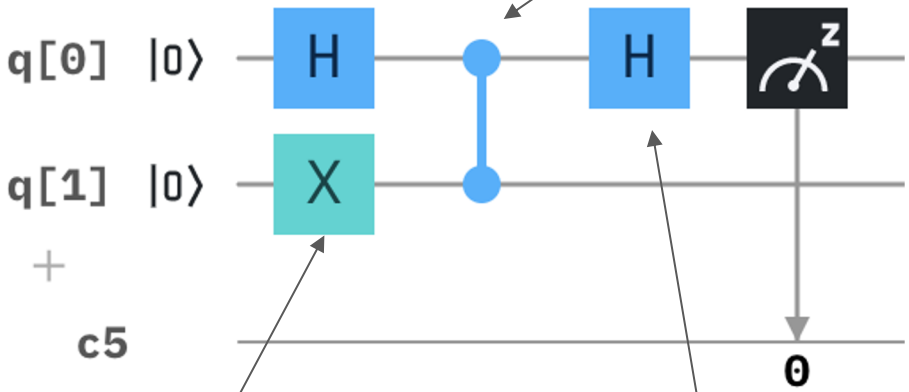
Number of $|0\rangle$ Qubits to Prepare as inputs

In the general case where **ϕ is arbitrary**, the number of qubits **t** decides the size of the top register to be used for phase estimation and determines the **accuracy** of the estimation.

Running on IBM Q

1-qubit example:

Controlled-Unitary operator chosen: **Controlled-Z gate**



To prepare |u>, we placed an X gate:

$$|u\rangle = X|0\rangle = |1\rangle$$

Earlier: QFT^{-1} on a single qubit = **Hadamard** gate

$$z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

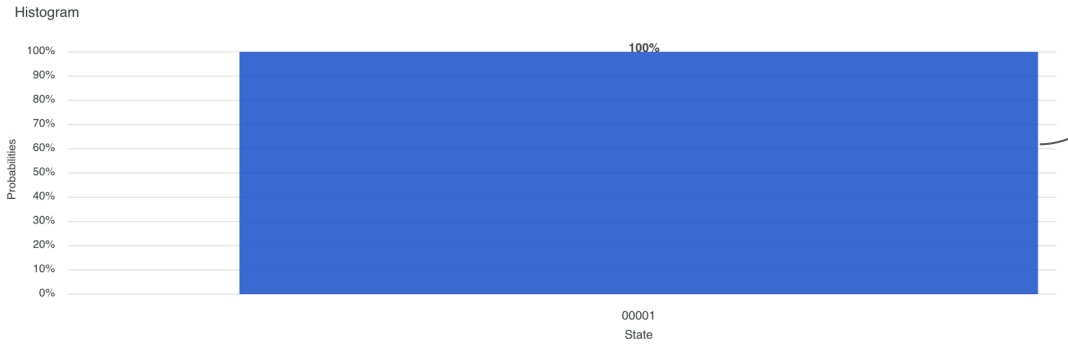
$$z\,|1\rangle = -|1\rangle = e^{\pi i}\,|1\rangle = e^{2\pi i\left(\frac{1}{2}\right)}|1\rangle$$

Φ = 0.1

eigenvector

eigenvalue

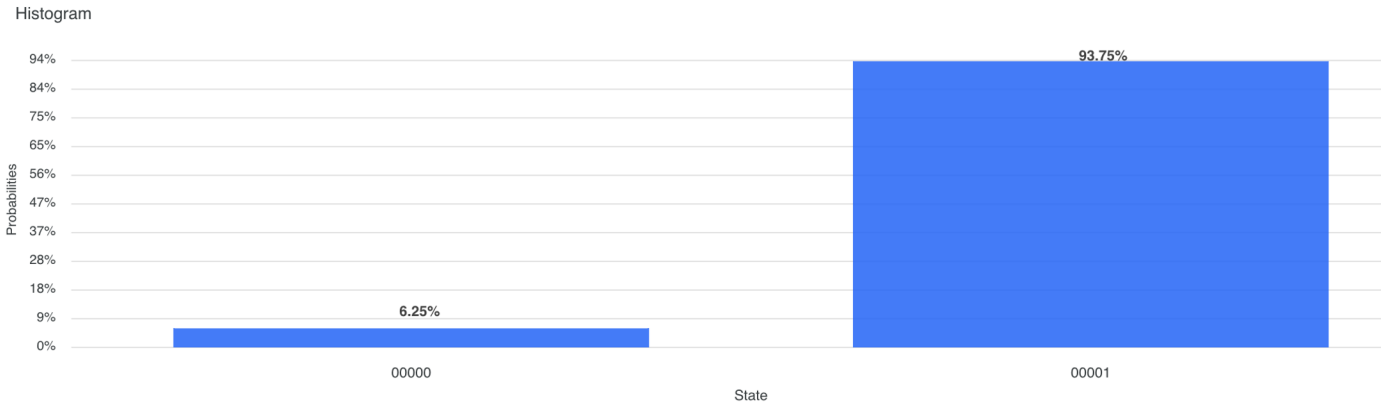
Result



Classical Simulator

Quantum Computer

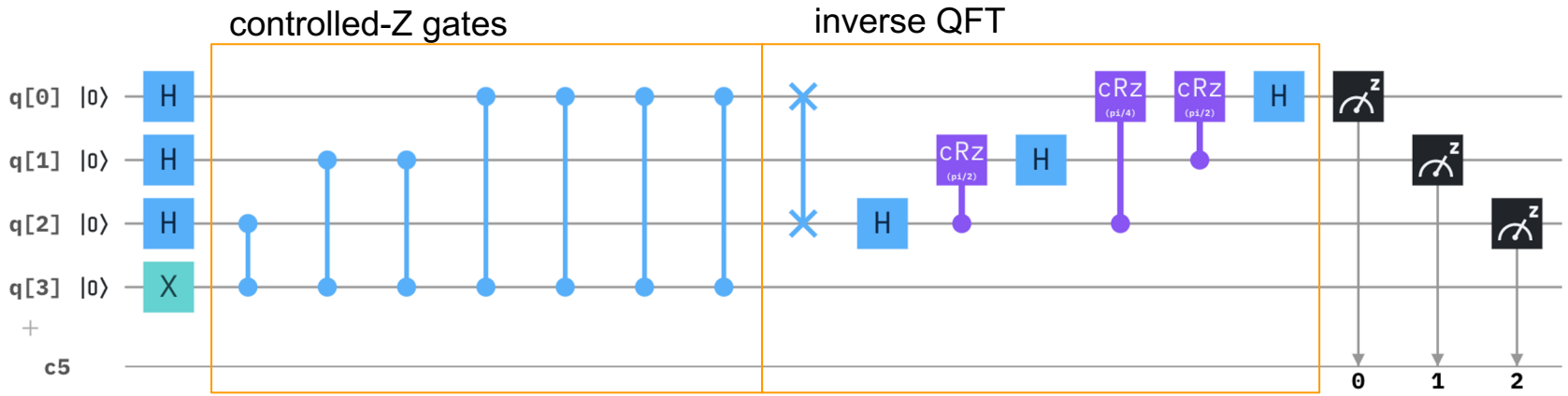
Result



Running on IBM Q

3-qubit example:

1) Create superposition using Hadamard gates
2) Apply controlled-Z operators
3) Apply Inverse Quantum Fourier Transform
4) Take measurement from top register

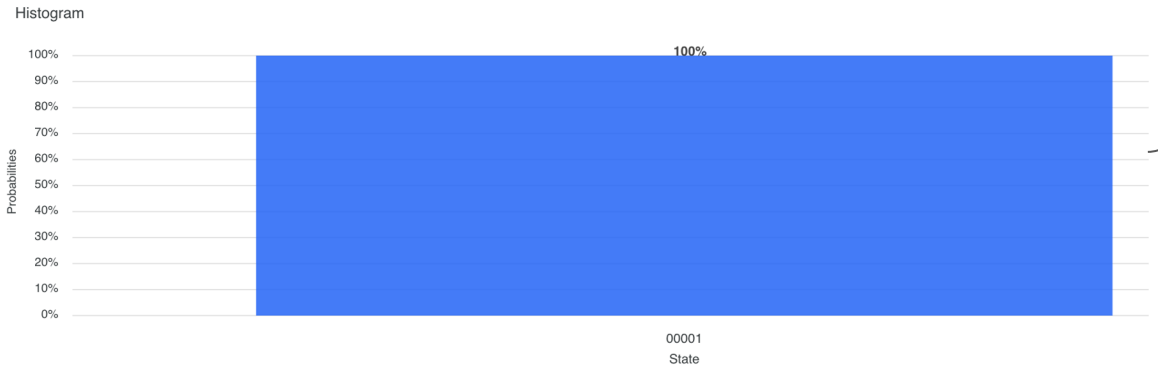


$$z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}$$

$$z\,|1\rangle = -|1\rangle = e^{\pi i}\,|1\rangle = e^{2\pi i\left(\frac{1}{2}\right)}|1\rangle$$

Φ = 0.100

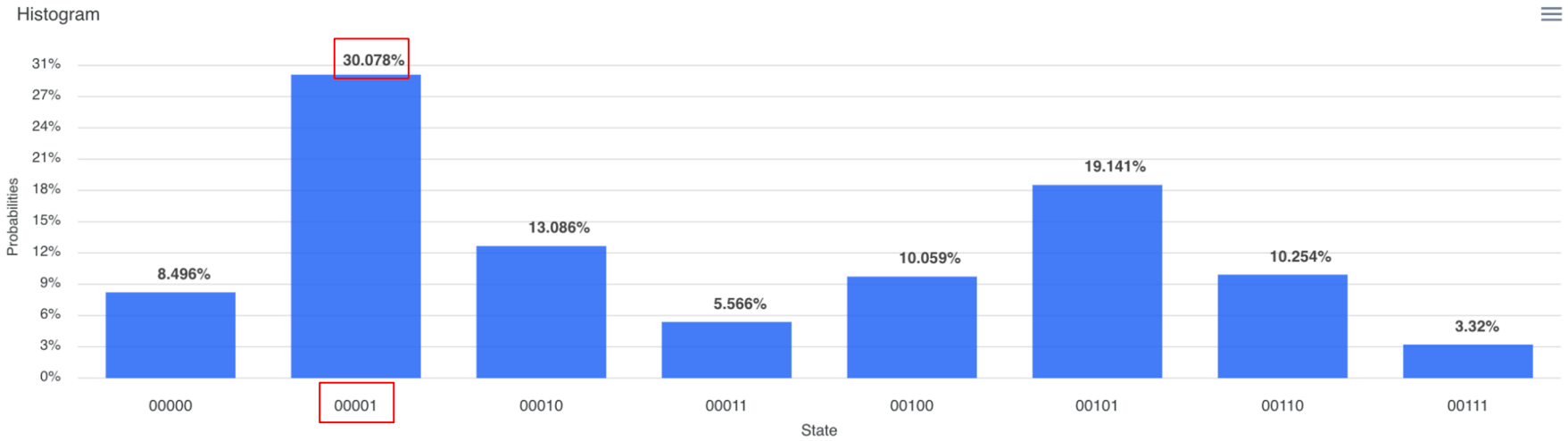
Result



Classical Simulator

Quantum Computer

Result

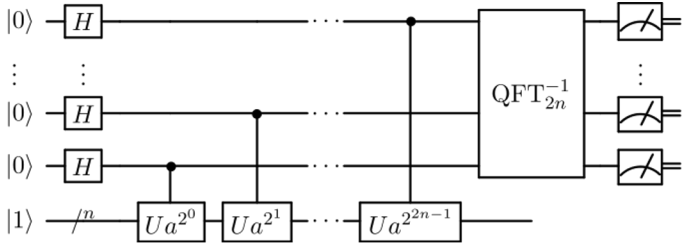


Strengths

1) Versatile: can be included in many algorithms and the unitary operator used can vary

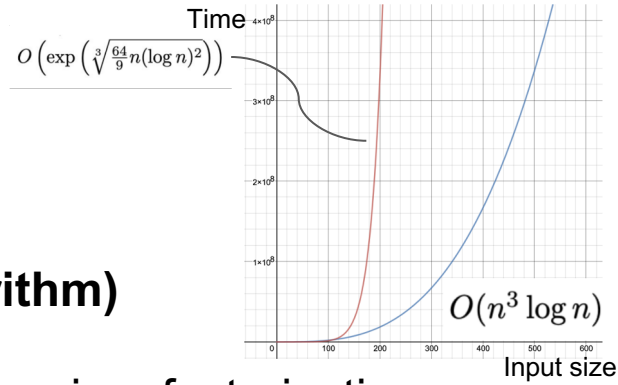
2 prominent algorithms:

1. Shor's algorithm - Prime factorisation of (extremely) large integers
    a. The algorithm is composed of two parts. The first part of the algorithm turns the factoring problem into the problem of finding the period of a function and may be implemented classically. The second part finds the period using the quantum Fourier transform and is responsible for the quantum speedup.



1. Quantum algorithm for linear systems of equations

Strengths



2) Exponential speed-up (Application on Shor's Algorithm)

Run time for the fastest known classical algorithm to solve prime factorisation:

$$O\left(\exp\left(\sqrt[3]{\frac{64}{9}n(\log n)^2}\right)\right)$$ **[Exponential☹]**

Run time for Shor's algorithm (with QPE):

$$O(n^3 \log n)$$ **[Polynomial😄]**

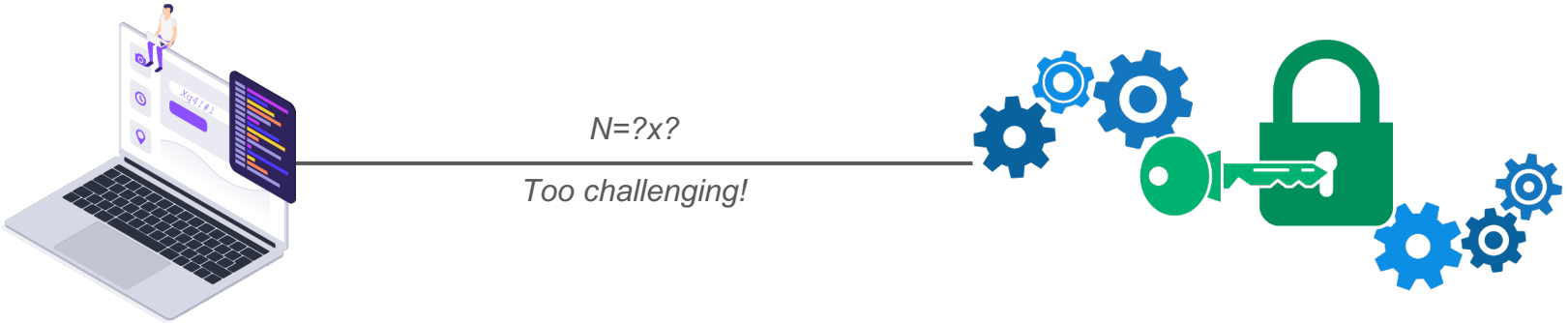
[n is size of the integer to factorise]

Exponential speed-up!

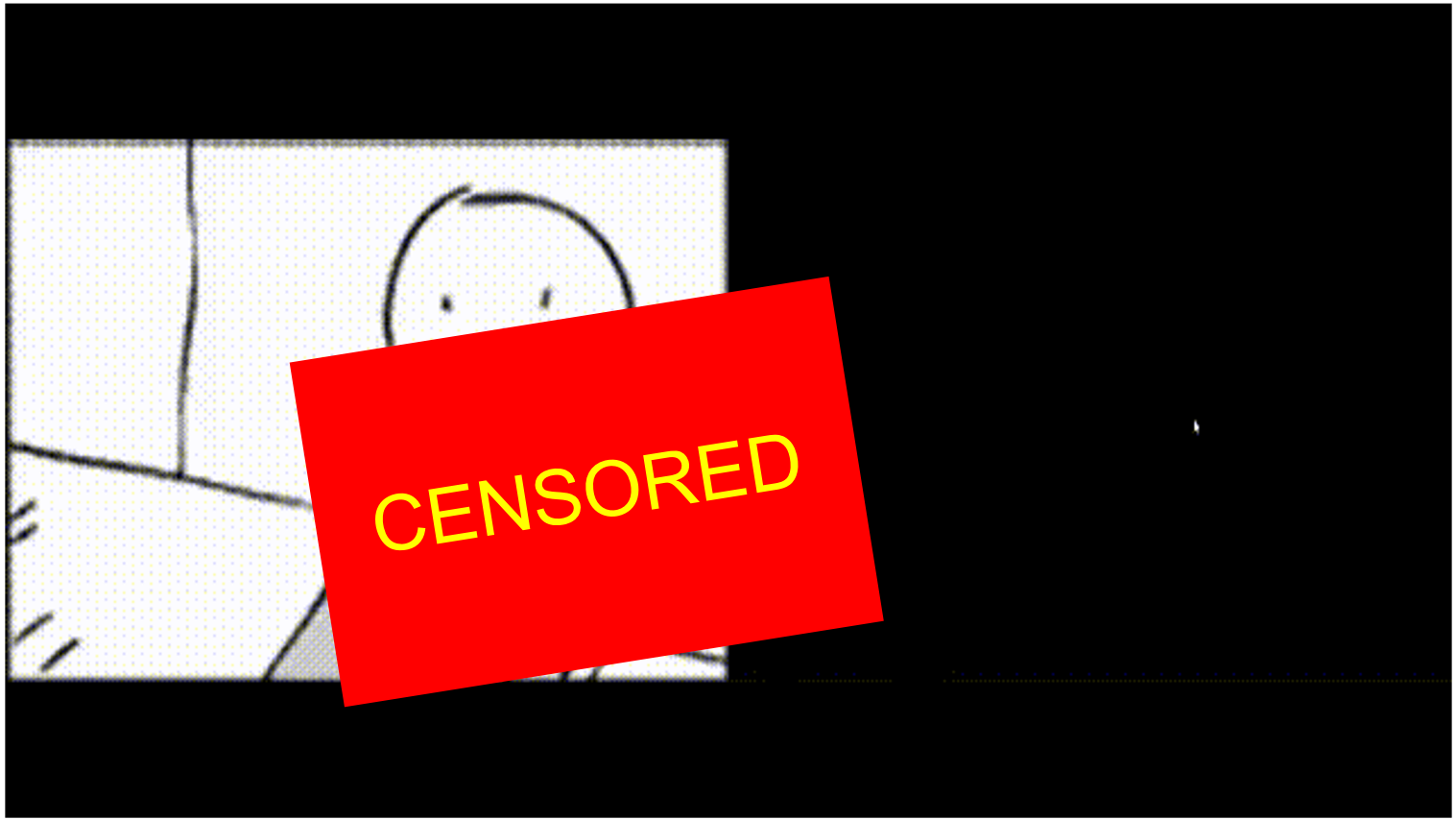
Application of QPE

RSA is a publicly known and widely used cryptographic algorithm on the Internet, often used for secure data transmission.
It is extremely hard to crack as it is based on the assumption that factoring large integers is computationally challenging.

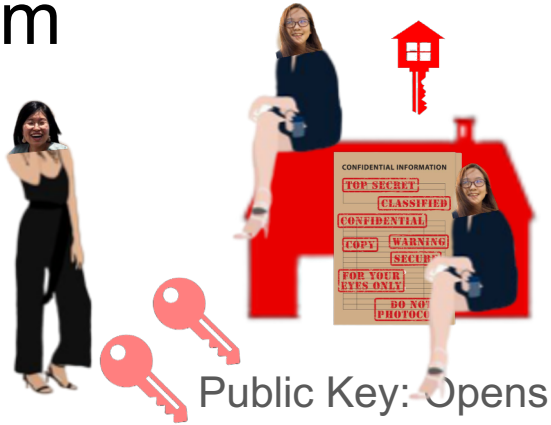
However, Shor's algorithm is able to crack the RSA scheme which is premised upon a key idea: **prime factorisation**.



Without Shor’s Algorithm...

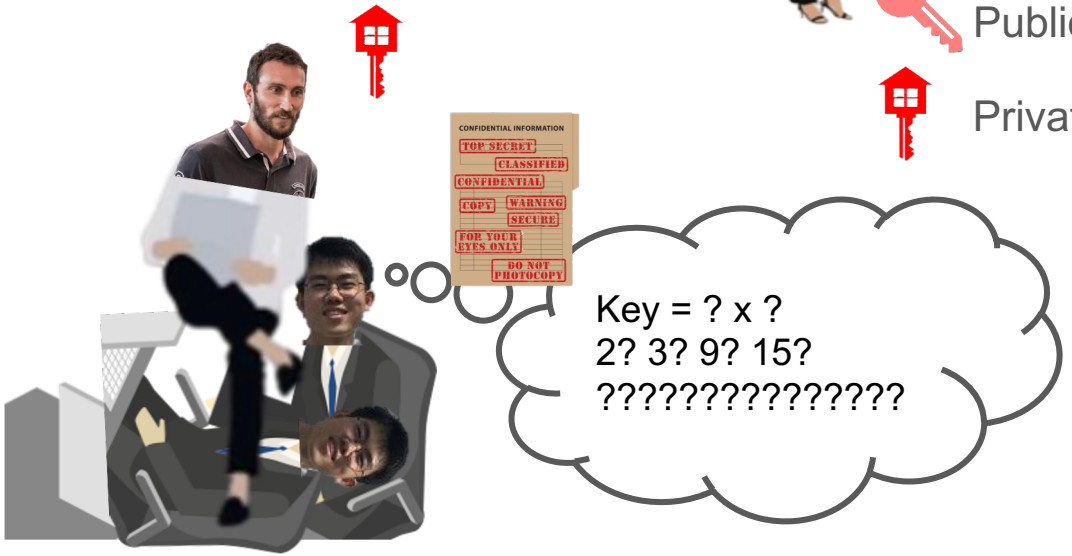


RSA encryption & Shor's algorithm



Public Key: Opens letterhole

Private Key: Opens house




YOU HAVE BEEN
HACKED !
+Dunken K Bliths

Physical Limitations

- Errors in estimations on quantum computers [seen in our IBM-Q]
- Larger input registers require tradeoffs between error rate (accuracy of estimation) and run time (circuit depth → computational resource usage)
- Increased running time for increasing higher powers of controlled-U gates

Implications:

- Impractical in current coherence-limited Noisy Intermediate-Scale Quantum (NISQ) devices without error correction / error mitigation
- Affects accuracy of estimating the phase

Me, after 20 minutes of quantum phase estimation:

Conclusion

QPE

- Estimates the phase of an eigenvalue of a unitary operator
- Moves phase information to control bits using controlled-U operators
- Uses inverse quantum Fourier transform to turn superposition into a single state → good estimator for the phase when measured
- Many useful applications including factoring large numbers
- However, practical limitations → long way to go

END

Thank you! Any questions?

References

Mohammadbagherpoor, H., Oh, Y. H., Singh, A., Yu, X., & Rindos, A. J. (2019). Experimental Challenges of Implementing Quantum Phase Estimation Algorithms on IBM Quantum Computer. *arXiv preprint arXiv:1903.07605*.

Nielsen, M. A., & Chuang, I. L. (2019). *Quantum Computation and Quantum Information*. Cambridge: Cambridge University Press.

O'Brien, T. E., Tarasinski, B., & Terhal, B. M. (2019). Quantum phase estimation of multiple eigenvalues for small-scale (noisy) experiments. *New Journal of Physics*, *21*(2), 023022. doi: 10.1088/1367-2630/aafb8e

Patil, S., Javadiabhari, A., Chiang, C.-F., Heckey, J., Martonosi, M., & Chong, F. T. (2014). Characterizing the performance effect of trials and rotations in applications that use Quantum Phase Estimation. *2014 IEEE International Symposium on Workload Characterization (IISWC)*. doi: 10.1109/iiswc.2014.6983057