CPEN 400Q Lecture 14
QFT and quantum phase estimation

Wednesday 28 February 2024

Announcements

- Quiz 6 on Monday
- Assignment 3 coming on Monday (QPE and Shor)
- Fill out peer review survey by Friday (link in Piazza)

Last time

We introduced the quantum Fourier transform, and saw how it is the analog of the classical inverse discrete Fourier transform.

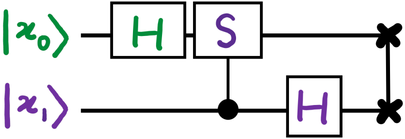
$$QFT|x\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle$$

$$QFT = \frac{1}{\sqrt{N}} \begin{pmatrix} 1 & 1 & 1 & \cdots & 1 \\ 1 & \omega & \omega^2 & \cdots & \omega^{N-1} \\ 1 & \omega^2 & \omega^4 & \cdots & \omega^{2(N-1)} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & \omega^{N-1} & \omega^{2(N-1)} & \cdots & \omega^{(N-1)(N-1)} \end{pmatrix}$$

where for n qubits, $N = 2^n$, and $\omega = e^{2\pi i/N}$

Last time

We saw the circuits for some special cases.



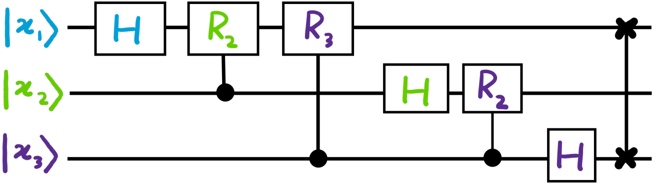


Image credit: Xanadu Quantum Codebook node F.2, F.3

Last time

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

The general form of the circuit is:

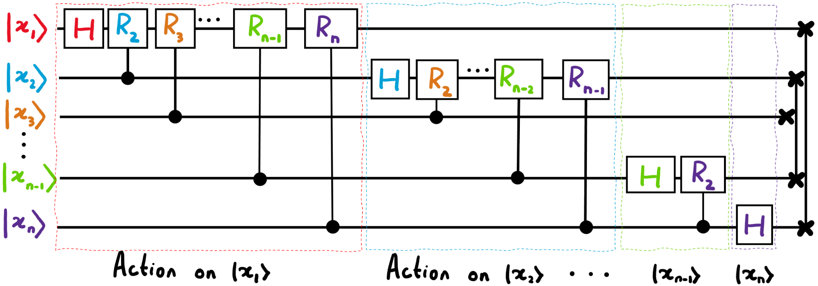


Image credit: Xanadu Quantum Codebook node F.3

Learning outcomes

- Implement the quantum Fourier transform in PennyLane
- Outline the steps of the quantum phase estimation (QPE) subroutine
- Use the QFT to implement QPE

$0.k_1 k_2 \cdots k_N = \sum_{l=1}^{n} k_l / 2^l$

We will reexpress k/N in fractional binary notation, then reshuffle and *factor* the output state to uncover the circuit structure.

$0.101 = \frac{1}{2} + \frac{1}{8} = 0.625$

$$|x\rangle \longrightarrow \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} w^{xk} |k\rangle$$

$|5\rangle = |101\rangle$

$N = 2^n \rightarrow$ # qubits

$w = e^{2\pi i/N}$

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i x \frac{k}{N}} |k\rangle$$

$= \sum_{l=1}^{n} \frac{k_l}{2^l}$

$$= \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} e^{2\pi i x \sum_{l=1}^{n} \frac{k_l}{2^l}} |k\rangle$$

$|k_1 k_2 \cdots k_n\rangle$

$\sum_{k_1=0}^{1} \sum_{k_2=0}^{1} \cdots$

$$= \frac{1}{\sqrt{N}} \sum_{k_1=0}^{1} \sum_{k_2=0}^{1} \cdots \sum_{k_n=0}^{1} e^{2\pi i x \sum_{l=1}^{n} \frac{k_l}{2^l}} |k_1 k_2 \cdots k_n\rangle$$

(keeping the last equation from the previous slide)

$$= \frac{1}{\sqrt{N}} \sum_{k_1=0}^{1} \sum_{k_2=0}^{1} \cdots \sum_{k_n=0}^{1} \underbrace{e^{2\pi i x \sum_{l=1}^{n} \frac{k_l}{2^l}}}_{e^{2\pi i x \frac{k_1}{2}} \, e^{2\pi i x \frac{k_2}{2^2}} \cdots} |k_1 k_2 \cdots k_n\rangle$$

$$|k_1\rangle |k_2\rangle \cdots |k_n\rangle$$

$$= \frac{1}{\sqrt{N}} \sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1} \left(e^{2\pi i x \frac{k_1}{2}} |k_1\rangle \right) \otimes \left(e^{2\pi i x \frac{k_2}{2^2}} |k_2\rangle \right) \otimes \cdots \otimes \left(e^{2\pi i x \frac{k_n}{2^n}} |k_n\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \underbrace{\sum_{k_1=0}^{1} \cdots \sum_{k_n=0}^{1}} \underbrace{\bigotimes_{l=1}^{n}} e^{2\pi i x \frac{k_l}{2^l}} |k_l\rangle$$

$$= \frac{1}{\sqrt{N}} \bigotimes_{l=1}^{n} \left(\sum_{k_l=0}^{1} e^{2\pi i x \frac{k_l}{2^l}} |k_l\rangle \right)$$

$\frac{x}{2^\ell} = 0.x_1 \cdots x_n$

(keeping the last equation from the previous slide)

$$= \frac{1}{\sqrt{N}} \bigotimes_{\ell=1}^{n} \left(\sum_{k_\ell=0}^{1} e^{2\pi i x \frac{k_\ell}{2^\ell}} |k_\ell\rangle \right)$$

$$= \frac{1}{\sqrt{N}} \bigotimes_{\ell=1}^{n} \left(|0\rangle + e^{2\pi i \frac{x}{2^\ell}} |1\rangle \right)$$

$x = 2^{n-1}x_1 + 2^{n-2}x_2 + \dots + x_n$

$$=$$

$$e^{2\pi i \frac{x}{2}} = e^{2\pi i \left(2^{n-2}x_1 + 2^{n-3}x_2 + \dots + x_{n-1} + \frac{x_n}{2}\right)} = e^{2\pi i 2^{n-2}x_1} e^{2\pi i 2^{n-3}x_2} \cdots e^{2\pi i \frac{x_n}{2}}$$

$$= \underbrace{e}_{1} \underbrace{e}_{1} \cdots \underbrace{e}_{e^{2\pi i\, 0.x_n}}$$

$\ell=2$

$$e^{2\pi i \cdot \frac{x}{2^\ell}} = e^{2\pi i \frac{x}{2^2}} = e^{2\pi i \left(2^{n-3}x_1 + 2^{n-4}x_2 + \dots + \frac{x_{n-1}}{2} + \frac{x_n}{4}\right)}$$

$$= e^{2\pi i \left(\frac{x_{n-1}}{2} + \frac{x_n}{4}\right)} = e^{2\pi i\, 0.x_{n-1}x_n}$$

$\ell=3$: $e^{2\pi i\, 0.x_{n-2}x_{n-1}x_n}$

So...

$$|x\rangle \underset{\text{QFT}}{\rightarrow} \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$

Believe it or not, this form reveals to us how we can design a circuit that creates this state!

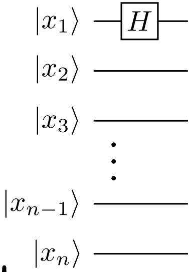
Starting with the state

$$|x\rangle = |x_1 \cdots x_n\rangle,$$

apply a Hadamard to qubit 1:

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\, 0.x_1}|1\rangle\right)|x_2 \cdots x_n\rangle$$

$$x_1 = 1: \; e^{2\pi i \cdot \frac{1}{2}} = e^{\pi i} = -1$$

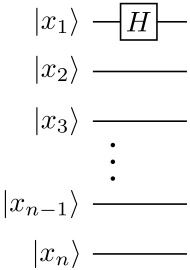


A circuit for the QFT

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 0.x_1}|1\rangle\right)|x_2 \cdots x_n\rangle$$

If $x_1 = 0$, $e^0 = 1$ and we get $|+\rangle$.

If $x_1 = 1$, $e^{2\pi i(1/2)} = e^{\pi i} = -1$ and we get $|-\rangle$.



A circuit for the QFT

We are trying to make a state that looks like this:

$$|x\rangle \to \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$

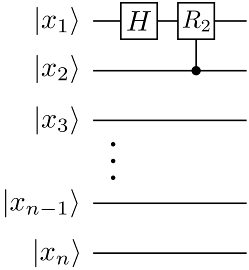
Every qubit has a different *phase* on the $|1\rangle$ state. Define

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

A circuit for the QFT

Apply controlled R_2 from qubit $2 \to 1$

$$R_2 = \begin{pmatrix} 1 & 0 \\ 0 & e^{\frac{2\pi i}{2^2}} \end{pmatrix}$$



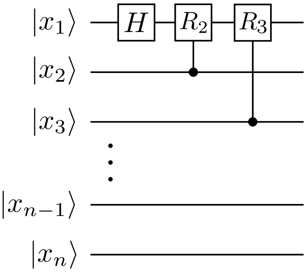
First qubit picks up a phase:

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\, 0.x_1}|1\rangle\right)|x_2 \cdots x_n\rangle$$

$$\to \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\, 0.x_1 x_2}|1\rangle\right)|x_2 \cdots x_n\rangle$$

A circuit for the QFT

Apply controlled R_3 from qubit $3 \to 1$

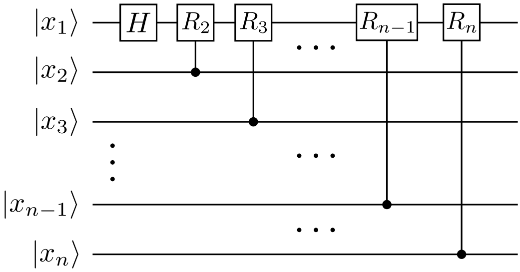


First qubit picks up another phase:

A circuit for the QFT

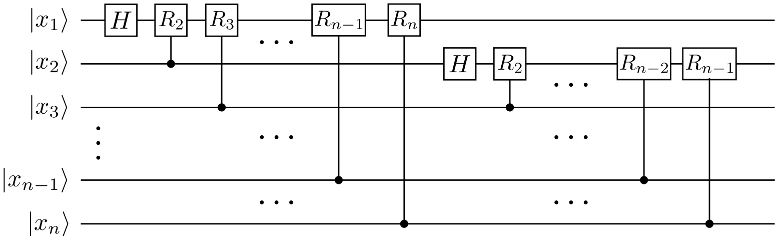
Apply a controlled R_4 from $4 \rightarrow 1$, etc. up to the n-th qubit to get

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i\, 0.x_1\cdots x_n}|1\rangle\right)|x_2\cdots x_n\rangle$$



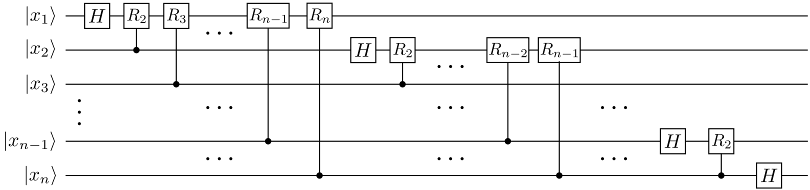
A circuit for the QFT

Next, do the same thing with the second qubit: apply H, and then controlled rotations from every qubit from 3 to n to get



Do this for all qubits to get that big ugly state from earlier:

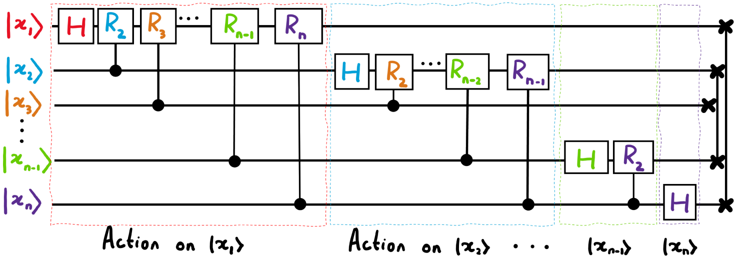
$$|x\rangle \to \frac{\left(|0\rangle + e^{2\pi i 0.x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_{n-1}x_n}|1\rangle\right)\cdots\left(|0\rangle + e^{2\pi i 0.x_1\cdots x_n}|1\rangle\right)}{\sqrt{N}}$$



(though note that the order of the qubits is backwards - this is easily fixed with some SWAP gates)

Gate counts:

- n Hadamard gates
- $n(n-1)/2$ controlled rotations
- $\lfloor n/2 \rfloor$ SWAP gates if you care about the order



The number of gates is *polynomial in n*!

Image credit: Xanadu Quantum Codebook node F.3

Reminder: where are we going?

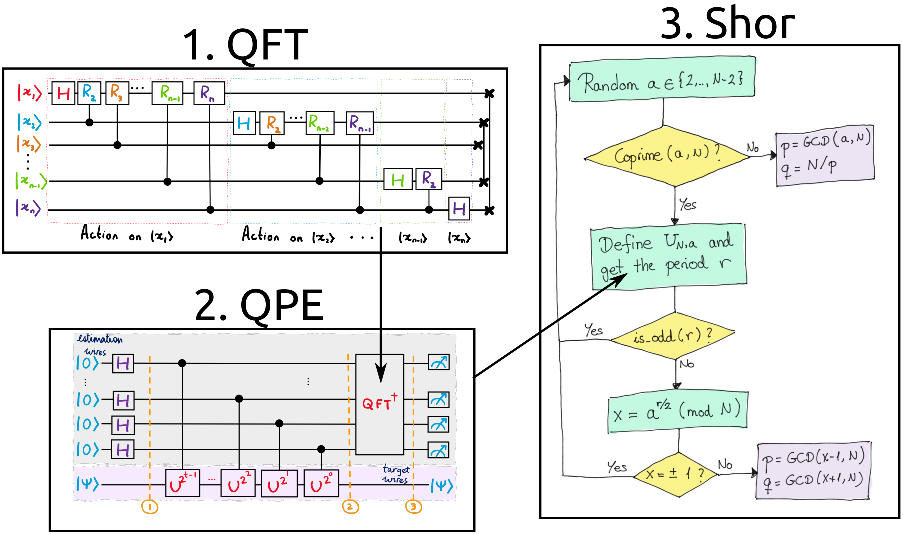


Image credit: Xanadu Quantum Codebook nodes F.3, P.3, S.4

Fun fact: eigenvalues of unitaries are complex numbers with magnitude 1.

Proof:

$$U|k\rangle = \lambda_k |k\rangle$$

$$\langle k| U^\dagger = \langle k| \lambda_k^*$$

$$\langle k| \underbrace{U^\dagger U}_{\mathbb{1}} |k\rangle = \langle k| \lambda_k^* \lambda_k |k\rangle$$

$$1 = \lambda_k^* \lambda_k = |\lambda_k|^2$$

Eigenvalues of unitary matrices

So we can write

$$\lambda_k = e^{2\pi i \theta_k}$$

!

where θ_k is some phase angle such that $|\theta_k| \leq 1$.

What if we want to *learn* an unknown θ_k?

Eigenvalues of unitary matrices

Idea: apply U to the relevant eigenvector, because that's "what makes the phase come out".

$$U|k\rangle = \lambda_k|k\rangle = e^{2\pi i\theta_k}|k\rangle$$

...but this is an unobservable *global* phase!

We have to do something different: eigenvalue estimation, or quantum phase estimation (QPE).

Quantum phase estimation

Given unitary U and eigenvector $|k\rangle$, estimate θ_k such that

$$U|k\rangle = e^{2\pi i\theta_k}|k\rangle$$

Must determine:

- How to design a circuit that extracts the θ_k
- To what precision can we estimate it
- What to do if we don't know a $|k\rangle$ in advance

(*You will explore the last two in your homework!*)

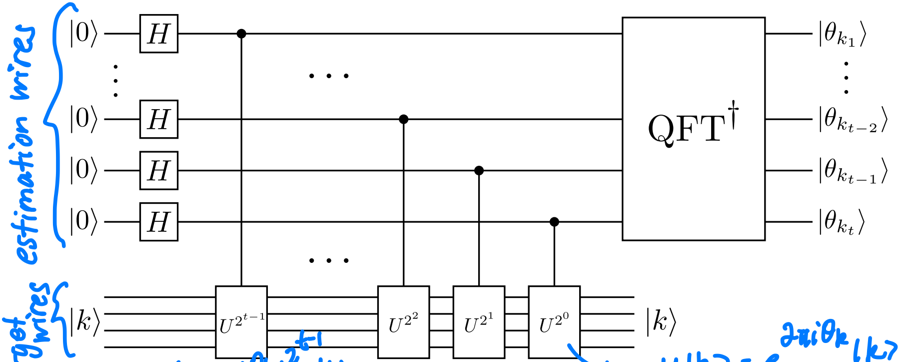
Quantum phase estimation

Let U be an n-qubit unitary; $|k\rangle$ is an n-qubit eigenstate.

Assume θ_k can be represented *exactly* using t bits:

$\lambda_k = e^{2\pi i \theta_k}$

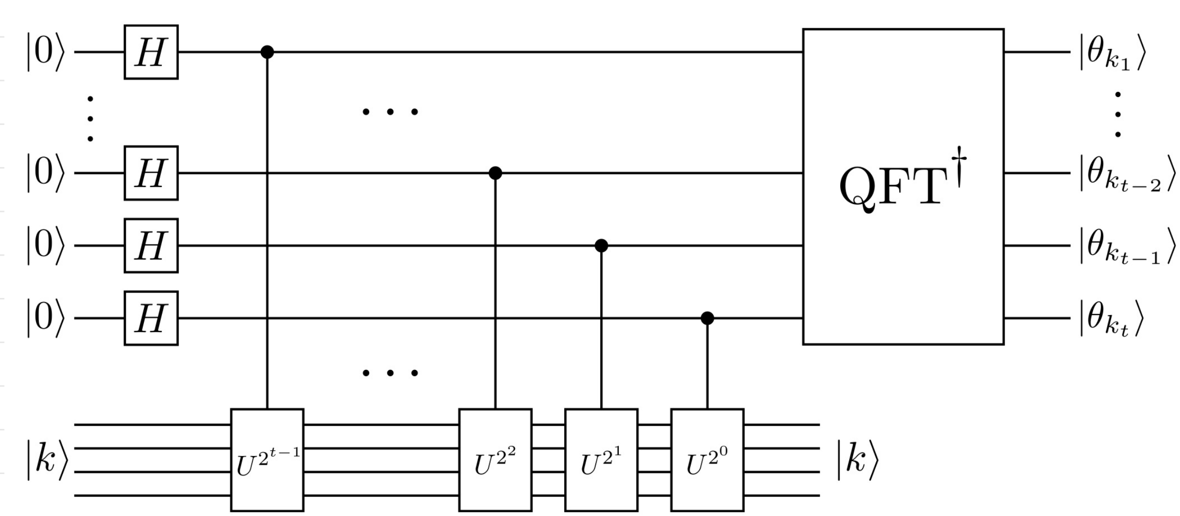
$$\theta_k = 0.\theta_{k_1} \cdots \theta_{k_t}$$



$(e^{2\pi i \theta_k})^{2^{t-1}} |k\rangle$ $\quad U|k\rangle = e^{2\pi i \theta_k}|k\rangle$

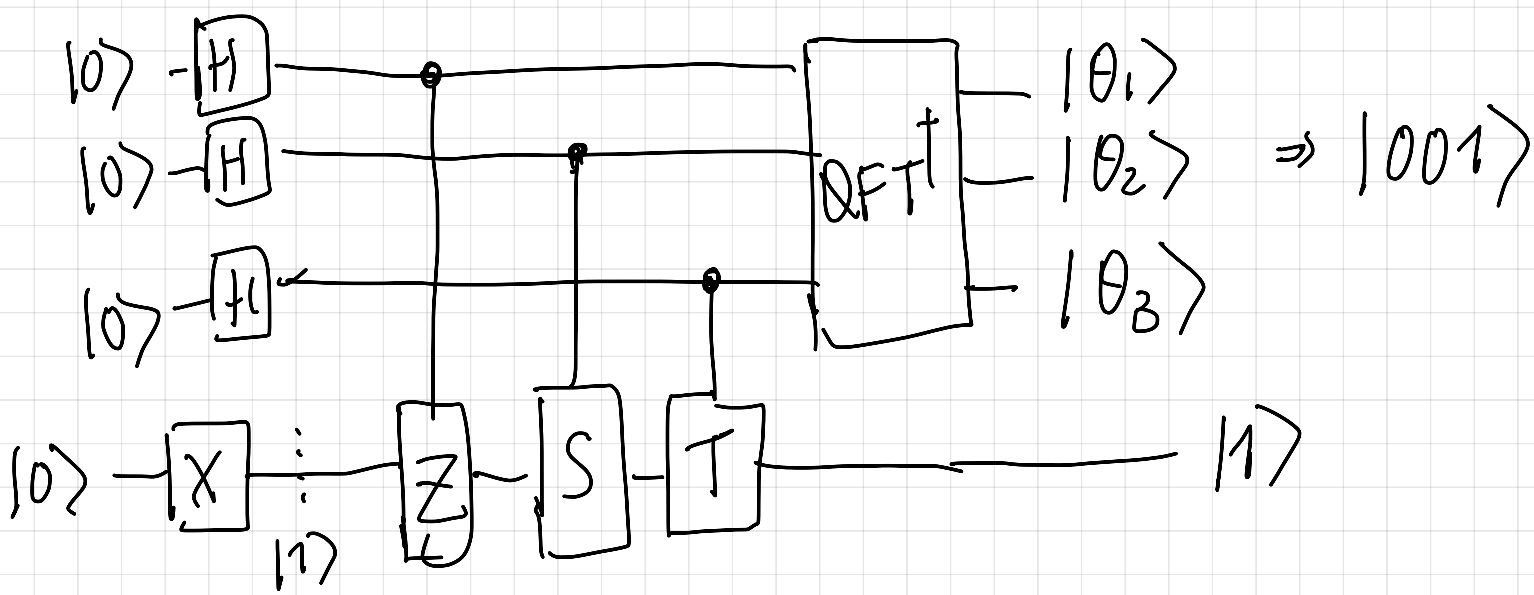
Exercise: QPE for

$$\underbrace{T = \begin{pmatrix} 1 & 0 \\ 0 & e^{i\pi/4} \end{pmatrix}}$$

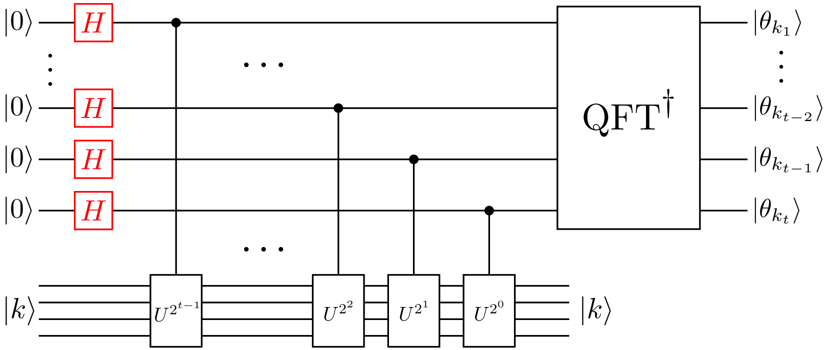


$$\lambda_1 = e^{i\pi/4} \qquad |k\rangle = \begin{pmatrix} 0 \\ 1 \end{pmatrix} = |1\rangle \qquad T|1\rangle = e^{i\pi/4}|1\rangle$$

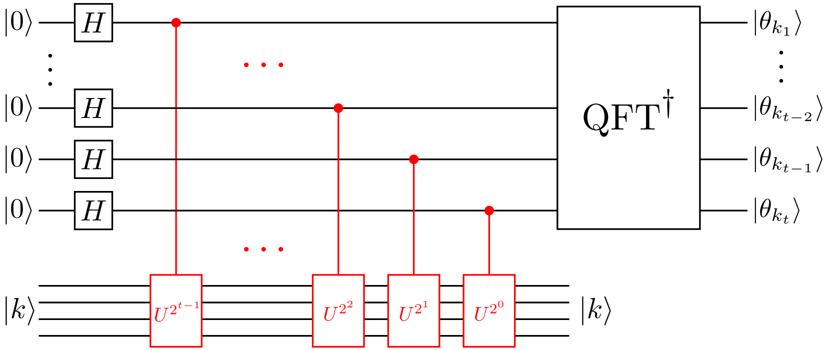
$$\lambda_k = e^{2\pi i \theta_k} \implies \theta_k = \frac{1}{8} = 0.001$$



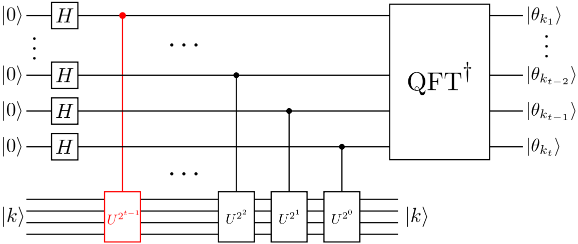
We will start here next time

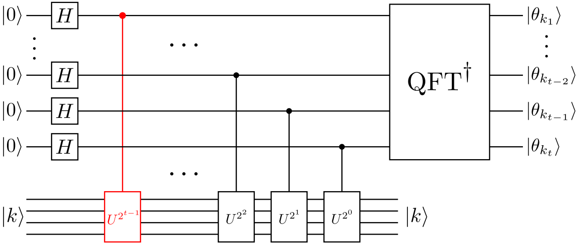


Quantum phase estimation: step 1



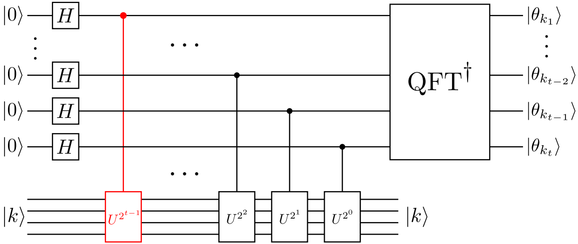
Quantum phase estimation: step 2





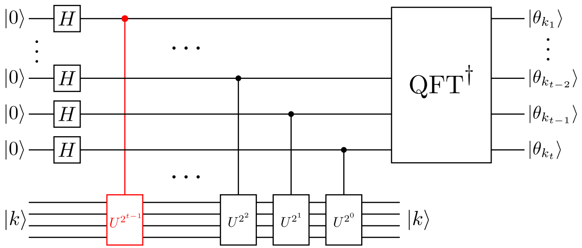
Use phase kickback

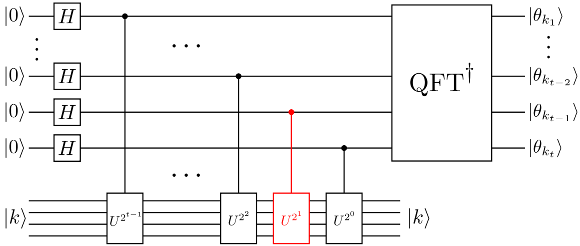
Quantum phase estimation: step 2



What is happening in the exponent?

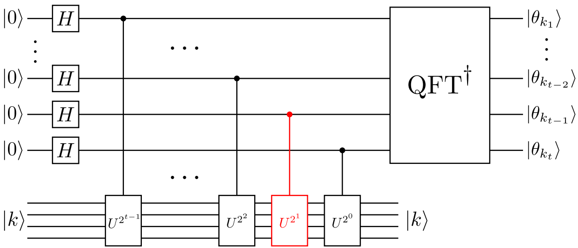
Quantum phase estimation: step 2





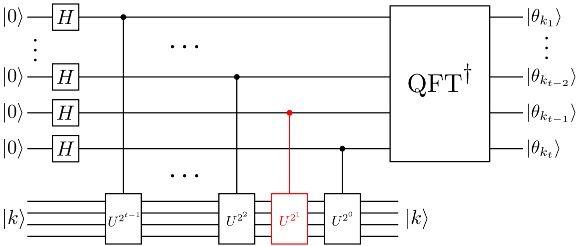
Check second-last qubit (ignore the others)

Quantum phase estimation: step 2

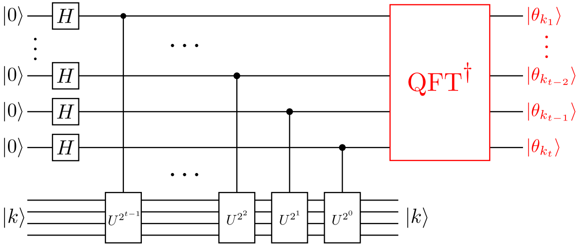


Again check the exponent...

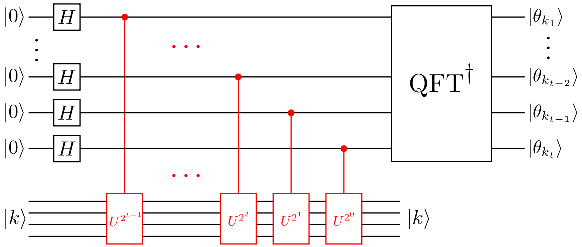
Quantum phase estimation: step 2



Quantum phase estimation: step 2



Can show in the same way for the last qubit (ignore others)

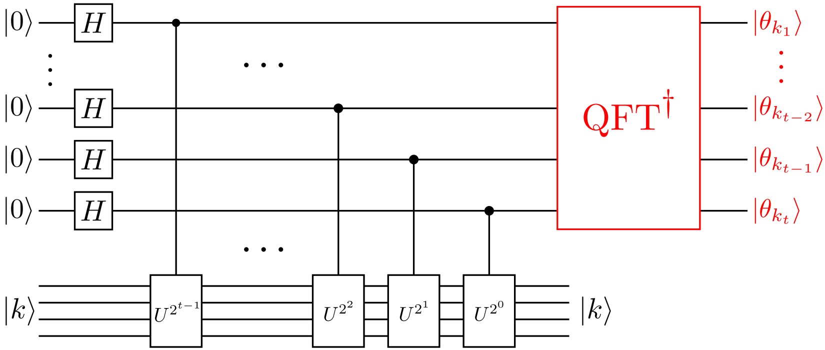


After step 2, we have the state

$$\frac{1}{\sqrt{2}}(|0\rangle+e^{2\pi i 0.\theta_{k_t}}|1\rangle)\cdots\frac{1}{\sqrt{2}}(|0\rangle+e^{2\pi i 0.\theta_{k_2}\cdots\theta_{k_t}}|1\rangle)\frac{1}{\sqrt{2}}(|0\rangle+e^{2\pi i 0.\theta_{k_1}\cdots\theta_{k_t}}|1\rangle)|k\rangle$$

Should look familiar!

Measure to learn the bits of θ_k.



Let's implement it.

Next time

Content:

- Quiz 6 on Monday
- Moving towards Shor's algorithm

Action items:

1. Work on project; midterm checkpoint next week
2. Fill out weekly peer assessment survey by Friday

Recommended reading:

- Codebook nodes F.1-F.3, P.1-P.4
- Nielsen & Chuang 5.1, 5.2