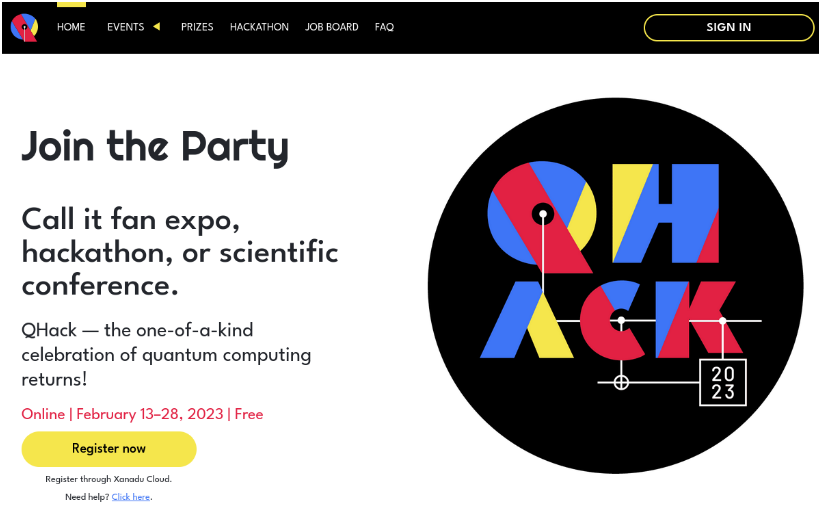
CPEN 400Q Lecture 09
The quantum Fourier transform (QFT)

Monday 6 February 2023

Announcements

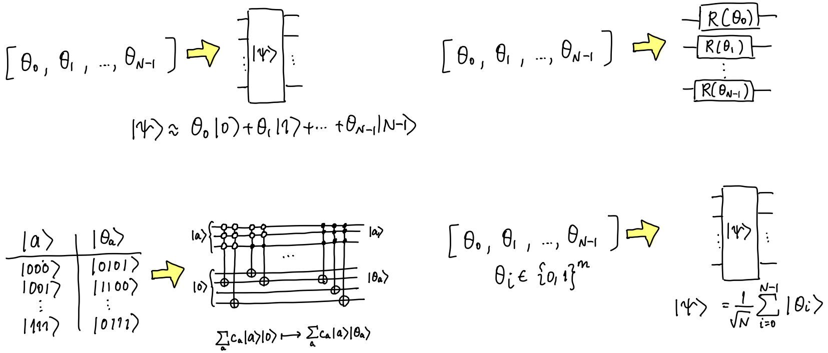
- Quiz 4 today
- Assignment 1 due tonight
- Next literacy assignment coming this week

Announcements: QHACK 2023

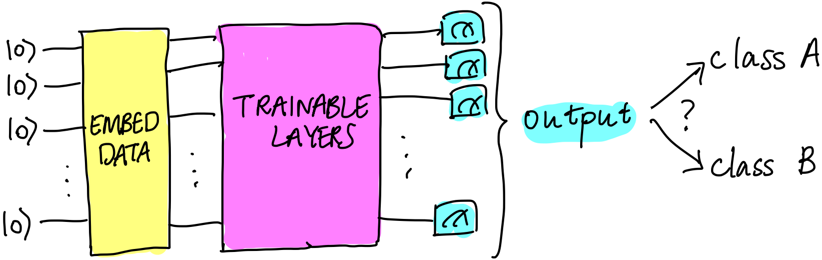


Visit `qhack.ai` for more details.

We explored different ways of encoding data into quantum circuits.



We successfully implemented a variational classifier in PennyLane.



Where are we going?

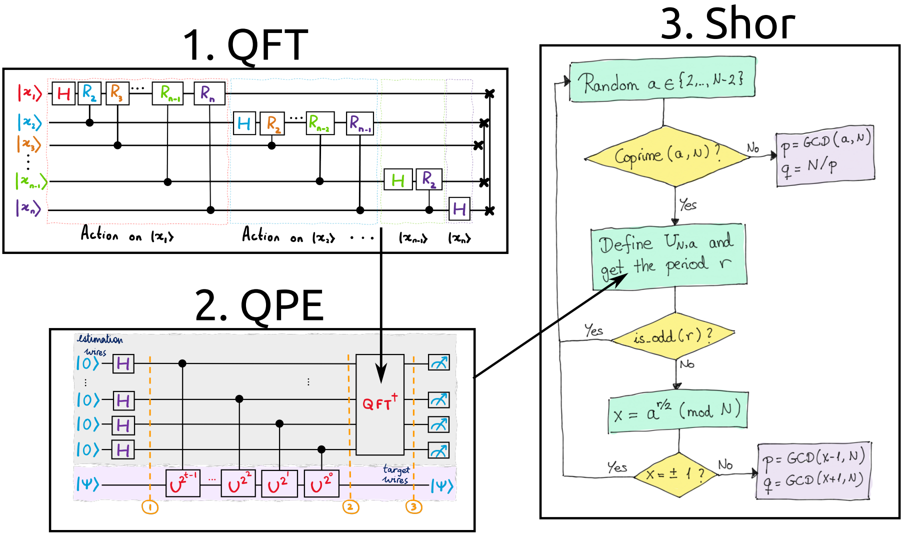


Image credit: Xanadu Quantum Codebook nodes F.3, P.3, S.4

Learning outcomes

- Express floating-point values in fractional binary representation
- Describe the behaviour of the quantum Fourier transform
- Implement the quantum Fourier transform in PennyLane

Today there will be lots of MATH.

The discrete Fourier transform

From ELEC 221[1]:

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

where $\bar{\omega} = e^{-2\pi i/N}$.

Note: sometimes there a prefactor, depends how inverse is defined.

[1] See Lecture 13: `https://github.com/glassnotes/ELEC-221`

The DFT (and the fast Fourier transform which implements it efficiently) are standard tools in digital signal processing to convert between time and frequency domain.

Given a signal $x[n]$ in the time domain, the DFT computes

$$\tilde{X}[k] = \sum_{n=0}^{N-1} e^{-\frac{2\pi i k n}{N}} x[n] = \sum_{n=0}^{N-1} \bar{\omega}^{nk} x[n]$$

The inverse DFT computes

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} e^{2\pi i k n/N} X[k] = \frac{1}{N}\sum_{k=0}^{N-1} \omega^{nk} X[k]$$

$$x[n] = \frac{1}{N}\sum_{k=0}^{N-1} e^{\frac{2\pi i k n}{N}} \tilde{X}[k] = \frac{1}{N}\sum_{k=0}^{N-1} \omega^{nk} \tilde{X}[k]$$

where $\omega = e^{2\pi i/N} = \bar{\omega}^{-1}$

The DFT invertible; its matrix is unitary (up to a prefactor).
Seems like a good candidate for a quantum computer...

The quantum Fourier transform (QFT) is the quantum analog of the **inverse DFT**.

Let $|x\rangle$ be an n-qubit computational basis state, $N = 2^n$.

The QFT sends

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

$$e^{2\pi i \frac{xk}{N}}$$

We are sending individual computational basis states to another basis, which is made up of linear combinations of computational basis states with complex exponential coefficients.

The QFT has the following action on the basis states:

$$QFT = \frac{1}{\sqrt{N}} \sum_{j=0}^{N-1} \sum_{k=0}^{N-1} \omega^{jk} |k\rangle \langle j|$$

Check that this works...

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

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

n qubits: $N=2^n$

As a matrix, it looks a lot like the DFT:

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

But... can we implement this unitary efficiently? How do we *synthesize* a circuit for it?

$$\omega = e^{\frac{2\pi i}{N}} = e^{\frac{2\pi i}{2}} = e^{\pi i}$$

Let's start with some special cases... suppose $n = 1$ ($N = 2$).

Here, $e^{2\pi i/2} = e^{i\pi} = -1$, so

$$QFT = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & \omega \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}$$

Hadamard

Look familiar?

Suppose $n = 2$ ($N = 4$).

$$QFT = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & \omega & \omega^2 & \omega^3 \\ 1 & \omega^2 & \omega^4 & \omega^6 \\ 1 & \omega^3 & \omega^6 & \omega^9 \end{pmatrix}$$

$$\omega = e^{\frac{2\pi i}{4}} = e^{\frac{\pi i}{2}} = i$$

Here $\omega = i$, and $\omega^2 = -1$, $\omega^4 = 1$. So

$$QFT = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & i & -1 & -i \\ 1 & -1 & 1 & -1 \\ 1 & -i & -1 & i \end{pmatrix}$$

H,

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

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

If we apply a SWAP, familiar things show up...

$$\mathrm{SWAP}(QFT) = \frac{1}{2}\begin{pmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & i & -1 & -i \\ 1 & -i & -1 & i \end{pmatrix}$$

Top blocks are H, bottom are HS. Can show that the following circuit implements this QFT:

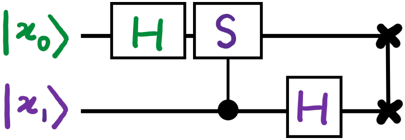
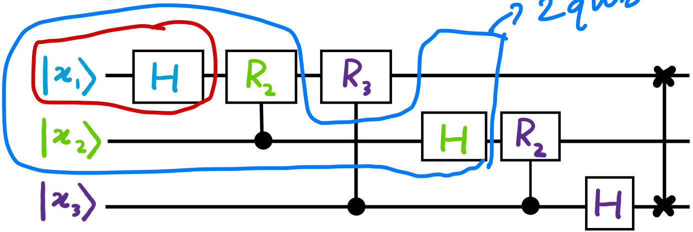


Image credit: Xanadu Quantum Codebook node F.2

Do the same for $n = 3$ ($N = 8$) but things get nasty... can show that the structure of the circuit that implements it is

2qubit QFT



Here, $R_2 = S$ and $R_3 = T$.

Image credit: Xanadu Quantum Codebook node F.3

Quantum Fourier transform

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

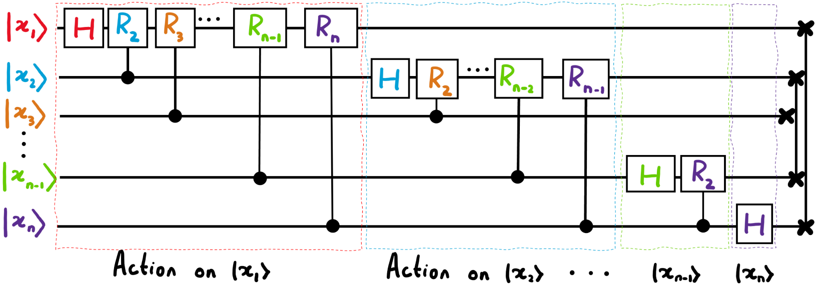


Image credit: Xanadu Quantum Codebook node F.3

A circuit for the QFT

$$|110\rangle \leftrightarrow |6\rangle$$

Consider the expression

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

Here x and k are represented as integers.

They are n-qubit computational basis states so they also have binary equivalents $|x\rangle = |x_1 \cdots x_n\rangle$, $|k\rangle = |k_1 \cdots k_n\rangle$:

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

and similarly for k.

Recall that $\omega = e^{2\pi i/N}$.

We are working with

$$\omega^{xk} = e^{2\pi i x(k/N)}$$

with $N = 2^n$.

We can write a fraction $k/2^n$ in a 'decimal version' of binary:

$$\begin{aligned} \frac{k}{2^n} &= 0.k_1 k_2 \cdots k_n \\ &= 2^{-1}k_1 + 2^{-2}k_2 + \cdots + 2^{-n}k_n \\ &= \frac{k_1}{2} + \frac{k_2}{2^2} + \cdots + \frac{k_n}{2^n} \\ &= \sum_{l=1}^{n} k_l \cdot 2^{-l} \end{aligned}$$

$$\frac{1}{2} + \frac{1}{2^2} + \frac{0}{2^3} + \frac{1}{2^4} = \frac{13}{16}$$

Exercise: let $k = 0.11010$. What is the numerical value of k?

$$\frac{}{2^n}$$

$$\frac{1}{2} \quad \frac{1}{4} \quad \frac{1}{16} = \frac{13}{16}$$

$$\begin{aligned} 0.11010 &= \frac{1}{2} + \frac{1}{2^2} + \frac{0}{2^3} + \frac{1}{2^4} + \frac{0}{2^5} \\ &= \frac{1}{2} + \frac{1}{4} + \frac{1}{16} \\ &= 0.8125 \end{aligned}$$

$$= 0.8125$$

Using the fractional decimal expression for k/N, we will work through the specification of the Fourier transform and see how we can reshuffle and *factor* the output state to get something that will make clear a circuit. Brace yourselves.

$$|x\rangle = \frac{1}{\sqrt{N}} \sum_{k=0}^{N-1} \omega^{xk} |k\rangle \sim (\ .\ ) \otimes (\ .\ ) \otimes \cdots \otimes (\ .\ )$$

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

$$\frac{k}{N} \to \sum_{\ell=1}^{n} k_\ell \cdot 2^{-\ell}$$

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

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

A circuit for the QFT

$\alpha|0\rangle \otimes \beta|1\rangle = (\alpha\beta)|0\rangle \otimes |1\rangle$

(keeping the last equation from the previous slide)

$\left(e^{2\pi i x \cdot \frac{k_1}{2}}|k_1\rangle\right) \otimes \cdots \left(e^{2\pi i x \frac{k_n}{2^n}}|k_n\rangle\right)$

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

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

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

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

$$\frac{x}{2} = \frac{2^{n-1}}{2}x_1 + \frac{2^{n-2}}{2}x_2 + \cdots + \frac{2}{2}x_{n-1} + \frac{x_n}{2}$$

$$= 2^{n-2}x_1 + 2^{n-3}x_2 + \cdots + x_{n-1} + \frac{x_n}{2}$$

(keeping the last equation from the previous slide)

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

$$e^{\frac{2\pi i x}{2}} = e^{\frac{2\pi i x_n}{2}} = e^{2\pi i\, 0.x_n}$$

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

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

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

A circuit for the QFT

We will start here on Friday!

So...

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

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

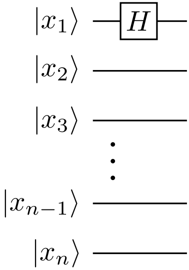
A circuit for the QFT

Starting with the state

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

apply a Hadamard to qubit 1:

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

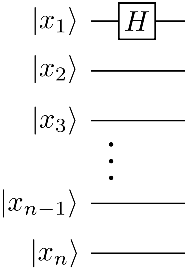


A circuit for the QFT

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

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

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



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. We are going to need some way of creating this.

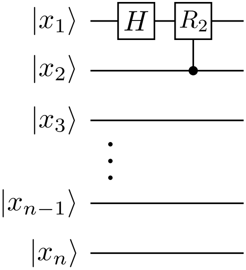
We define the gate:

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

A circuit for the QFT

Apply controlled R_2 from qubit $2 \rightarrow 1$

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

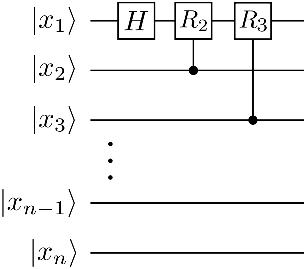


First qubit picks up a phase:

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

Apply controlled R_3 from qubit $3 \to 1$

$$R_3 = \begin{pmatrix} 1 & 0 \\ 0 & e^{2\pi i/2^3} \end{pmatrix}$$



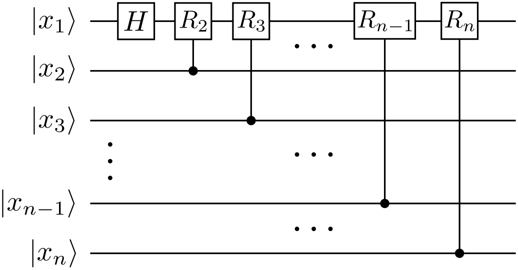
First qubit picks up another phase:

$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 0.x_1x_2}|1\rangle\right)|x_2\cdots x_n\rangle \to \frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 0.x_1x_2x_3}|1\rangle\right)|x_2\cdots x_n\rangle$$

A circuit for the QFT

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

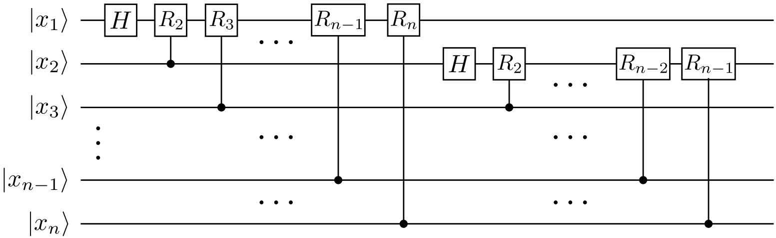
$$\frac{1}{\sqrt{2}}\left(|0\rangle + e^{2\pi i 0.x_1 x_2 \cdots x_n}|1\rangle\right)|x_2 \cdots x_n\rangle$$



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

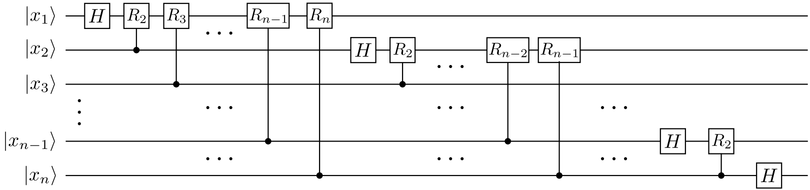
$$\frac{1}{\sqrt{2}^2}\left(|0\rangle + e^{2\pi i 0.x_1x_2\cdots x_n}|1\rangle\right)\left(|0\rangle + e^{2\pi i 0.x_2\cdots x_n}|1\rangle\right)|x_3\cdots x_n\rangle$$



A circuit for the QFT

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

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

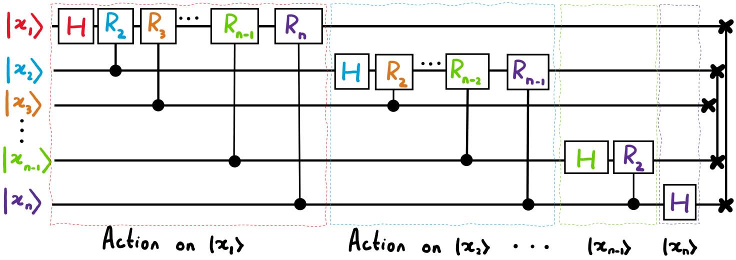


(though note that the order of the qubits is backwards - this is easily fixed with some SWAP gates)

Quantum Fourier transform

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

Next time

Content:

- Quantum phase estimation

Action items:

1. Finish Assignment 1

Recommended reading:

- Codebook module F
- Nielsen & Chuang 5.1
- Codebook module P (for next class)