Quantum Fourier Transform

Peter Young

(Dated: May 28, 2018)

The quantum Fourier Transform (QFT) acts on a set of $N = 2^n$ basis states to generate another set:

$$QFT_n\left(\,|j\rangle\,\right) = \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} e^{2\pi i jk/N}\,|k\rangle\,. \tag{1}$$

We use the binary notation

$$j = j_0 + 2j_1 + 2^2 j_2 + \cdots + 2^{n-1} j_{n-1}, \tag{2}$$

and similarly for k. The j_α take values 0 or 1 and are the bits of j.

We compute jk/N. Since this occurs as the power of $e^{2\pi i}$ in Eq. (1) an integer part will be irrelevant since $e^{2\pi i m} = 1$ for integer m.

$$\begin{aligned}\frac{jk}{N} = &\left(\frac{j_0}{2^n} + \frac{j_1}{2^{n-1}} + \cdots + \frac{j_{n-1}}{2}\right) k_0 + \\ &\left(\frac{j_0}{2^{n-1}} + \frac{j_1}{2^{n-2}} + \cdots + \frac{j_{n-2}}{2}\right) k_1 + \\ &\vdots \\ &+ \left(\frac{j_0}{2^2} + \frac{j_1}{2}\right) k_{n-2} + \left(\frac{j_0}{2}\right) k_{n-1} + \text{integers.}\end{aligned} \tag{3}$$

We write $|j\rangle \equiv |j_{n-1}\rangle \cdots |j_1\rangle |j_0\rangle$ and similarly for $|k\rangle$ and use the binary point notation

$$0.x_{n-1}x_{n-2}\cdots x_1 x_0 = \frac{x_{n-1}}{2} + \frac{x_{n-2}}{2^2} + \cdots + \frac{x_1}{2^{n-1}} + \frac{x_0}{2^n}\,. \tag{4}$$

From Eqs. (1) and (3) we can write the QFT as

$$\begin{aligned}QFT_n\left(\,|j_{n-1}\rangle \cdots |j_1\rangle |j_0\rangle\,\right) &= \frac{1}{2^{n/2}} \sum_{k=0}^{2^n-1} e^{2\pi i jk/N}\,|k_{n-1}\rangle \cdots |k_1\rangle |k_0\rangle \\ &= \frac{1}{2^{n/2}} \left(\sum_{k_0=0,1} e^{2\pi i k_0 (0.j_{n-1}j_{n-2}\cdots j_0)}\,|k_0\rangle\right) \\ &\quad \otimes \left(\sum_{k_1=0,1} e^{2\pi i k_1 (0.j_{n-2}j_{n-3}\cdots j_0)}\,|k_1\rangle\right) \\ &\quad \vdots \\ &\quad \otimes \left(\sum_{k_{n-1}=0,1} e^{2\pi i k_{n-1} (0.j_0)}\,|k_{n-1}\rangle\right)\end{aligned} \tag{5}$$

Writing explicitly the sums in the k_α in Eq. (5), we have

$$QFT_n\left(|j_{n-1}\rangle\cdots|j_1\rangle|j_0\rangle\right)=\left(\frac{|0\rangle+e^{2\pi i(0.j_{n-1}j_{n-2}\cdots j_0)}|1\rangle}{\sqrt{2}}\right)\quad\text{(sum over }k_0)$$
$$\otimes\left(\frac{|0\rangle+e^{2\pi i(0.j_{n-2}j_{n-3}\cdots j_0)}|1\rangle}{\sqrt{2}}\right)\quad\text{(sum over }k_1)$$
$$\vdots$$
$$\otimes\left(\frac{|0\rangle+e^{2\pi i(0.j_0)}|1\rangle}{\sqrt{2}}\right).\quad\text{(sum over }k_{n-1})\tag{6}$$

Let's make things concrete by considering the simplest case of $n=2$. Setting $n=2$ in Eq. (6) and considering the lines corresponding to $k=0$ and 1, we have

$$|k_0\rangle\rightarrow\left(\frac{|0\rangle+e^{2\pi i\left(\frac{j_1}{2}+\frac{j_0}{4}\right)}|1\rangle}{\sqrt{2}}\right),\tag{7a}$$

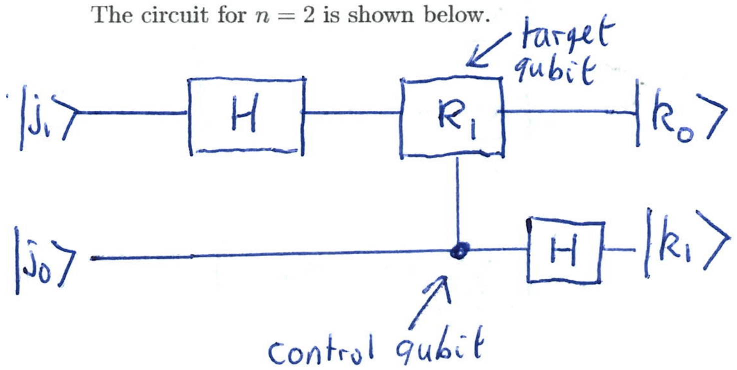
$$|k_1\rangle\rightarrow\left(\frac{|0\rangle+e^{2\pi i\left(\frac{j_0}{2}\right)}|1\rangle}{\sqrt{2}}\right).\tag{7b}$$

It is easy to construct $|k_1\rangle$. If $j_0=0$ then $|k_1\rangle=\frac{1}{\sqrt{2}}(|0\rangle+|1\rangle)$ and if $j_0=0$ then $|k_1\rangle=\frac{1}{\sqrt{2}}(|0\rangle-|1\rangle)$. This is just the transformation given by a Hadamard gate. For $|k_0\rangle$, the dependence on j_1 can similarly given by a Hadamard, but there is an additional phase change of $e^{i\pi/2}(=i)$ if $j_0=1$. Thus we will need *controlled phase gates*, R_d. As usual with a controlled gate there is a target qubit and a control qubit, and the control qubit remains unchanged during the action of the gate. If the control qubit is 1 then the amplitude of basis state $|1\rangle$ of the target qubit acquires a phase $\pi/2^d$. In other words, if the control qubit is 1 the target qubit is acted on by the operator

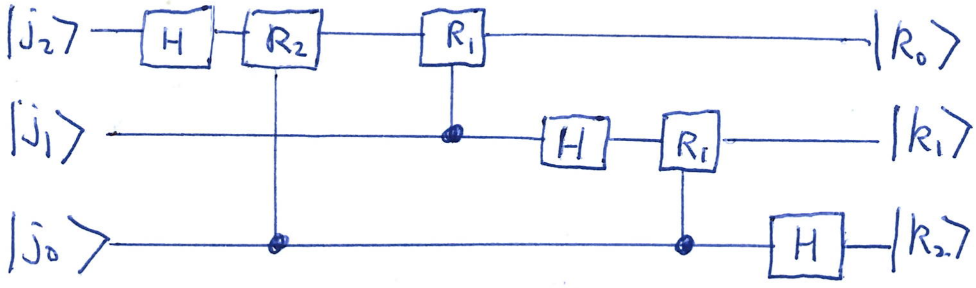
$$R_d=\begin{pmatrix}1&0\\0&e^{i\pi/2^d}\end{pmatrix}\tag{8}$$

and otherwise stays unchanged. R_0 is just the controlled-Z gate that we have mentioned earlier.

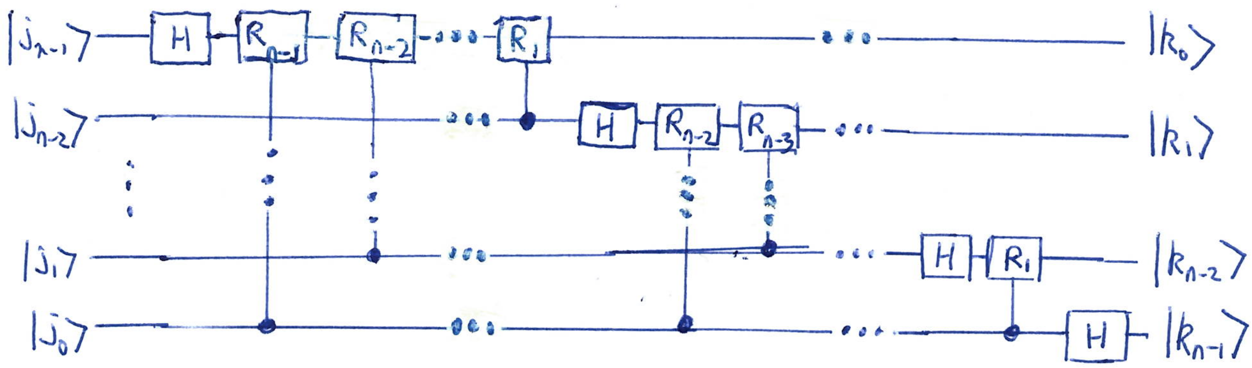
The circuit for $n=2$ is shown below.



The circuit for $n = 3$ is shown below.



It is not too difficult to see that the pattern for general n is as follows:



Note that on output the bits are in the reverse order. This can be compensated by adding $n/2$ swap gates. We see that the QFT also needs n H-gates and $n(n-1)/2$ Ctrl-R gates. These have to be used sequentially so the execution time is of order n^2, which is exponentially faster than the classical Fast Fourier Transform, FFT, which is of order $n2^n$. In fact the classical FFT can be derived from the breakup of the exponential in Eq. (5).