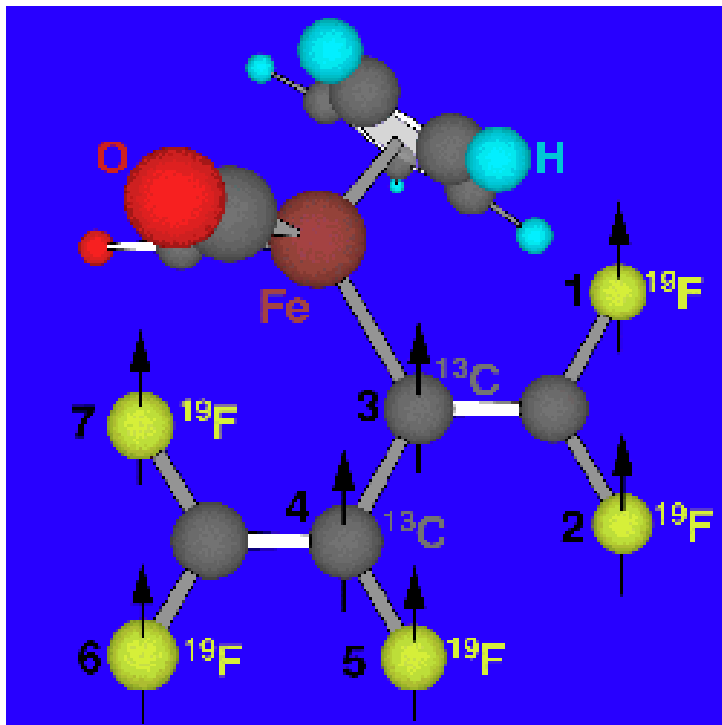
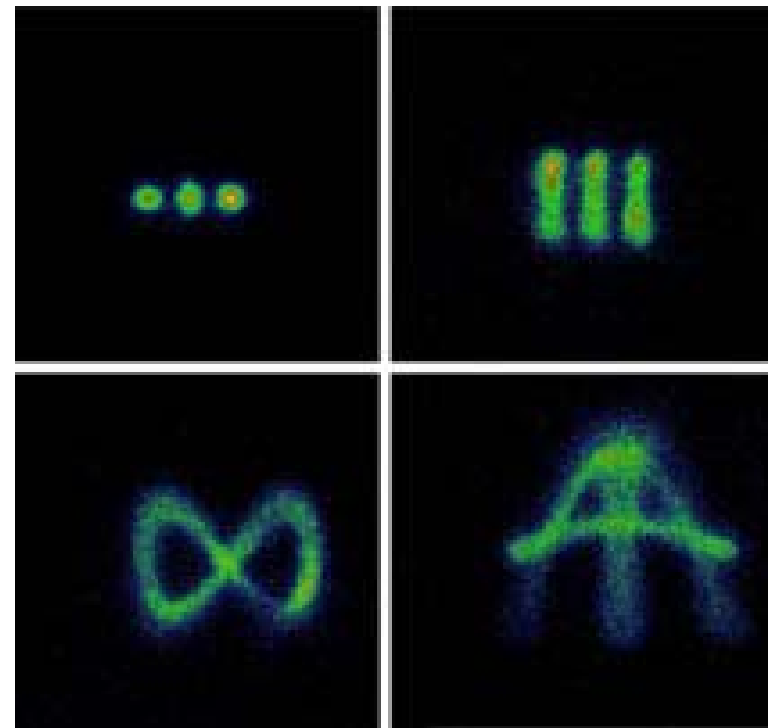


The Quantum Computer

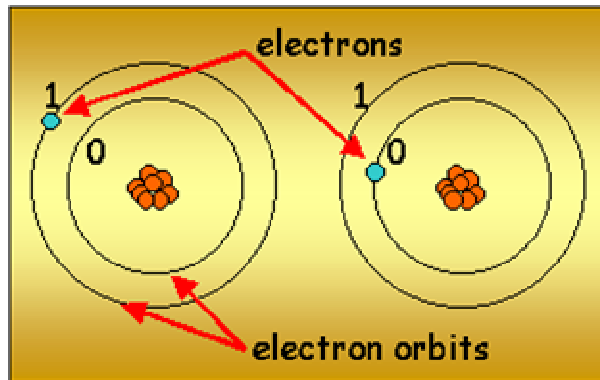


7-Qubit Quantum Computer

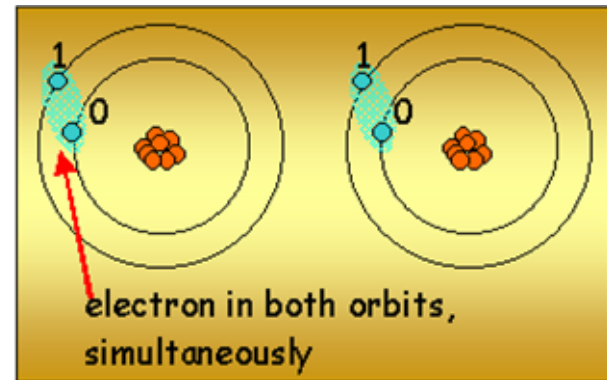


Typical Ion Oscillations in a Trap

Bits



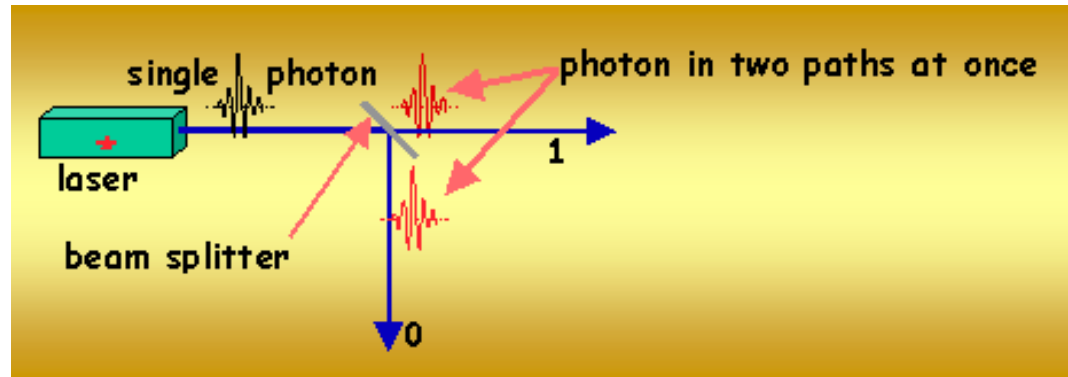
VS



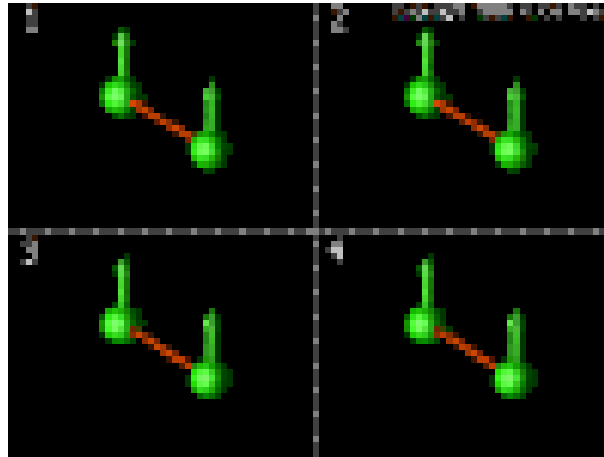
- Each qubit can represent both a 0 or 1 at the same time!
- This phenomenon is known as **Superposition**.
- It leads to **Quantum Parallelism**, which allows quantum computers to perform certain algorithms much, much faster than conventional computers.

Other Qubit-Representing Schemes

- Running photons through beam-splitters.



- Changing electron spin ($+1/2$ to $-1/2$)



Some Mathematical Notation

ket \longrightarrow $|abc\rangle = \begin{bmatrix} a \\ b \\ c \end{bmatrix}$, $\langle abc| = [\bar{a} \quad \bar{b} \quad \bar{c}]$ bra

- A bit is represented by either of two kets, known together as the **standard basis**. These kets are perpendicular and correspond to the X and Y axes.

$$|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}, |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

standard basis

- A quantum bit (qubit) is in a state of **superposition**, which is often represented by a linear combination of the standard basis.

$$q = c_0|0\rangle + c_1|1\rangle$$

Measuring Qubits

- c_0 and c_1 are complex, but they are scaled so that:

$$|c_0|^2 + |c_1|^2 = 1$$

- When a qubit is measured, it collapses into either of the two basis states with the following probabilities:

$$P(|0\rangle) = c_0^2, \quad P(|1\rangle) = c_1^2$$

- The state must collapse when measured in order to make sense. Otherwise you would be looking at both a 0 and a 1 at the same time!

Multiple Qubits: Tensor Product

- A computer with just a single qubit wouldn't do much good. So we must devise a mathematical scheme to represent attached qubits...we call this scheme the **tensor product**.
- Suppose we have 3 two-dimensional variables:

$$U = (u_0, u_1) \quad V = (v_0, v_1) \quad W = (w_0, w_1)$$

- The tensor product,

$$\begin{aligned} U \otimes V \otimes W &= \{u_0, u_1\} \otimes \{v_0, v_1\} \otimes \{w_0, w_1\} \\ &= \{(u_0, v_0, w_0), (u_0, v_0, w_1), (u_0, v_1, w_0), \\ &\quad (u_0, v_1, w_1), (u_1, v_0, w_0), (u_1, v_0, w_1), \\ &\quad (u_1, v_1, w_0), (u_1, v_1, w_1)\} \end{aligned}$$

Let's do an example involving matrices.

$$A = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix}, B = \begin{bmatrix} 5 \\ 6 \end{bmatrix}$$

$$A \otimes B = \begin{bmatrix} 1B & 2B \\ 3B & 4B \end{bmatrix} = \begin{bmatrix} 1 \begin{bmatrix} 5 \\ 6 \end{bmatrix} & 2 \begin{bmatrix} 5 \\ 6 \end{bmatrix} \\ 3 \begin{bmatrix} 5 \\ 6 \end{bmatrix} & 4 \begin{bmatrix} 5 \\ 6 \end{bmatrix} \end{bmatrix} = \begin{bmatrix} 5 & 10 \\ 6 & 12 \\ 15 & 20 \\ 18 & 24 \end{bmatrix}$$

Finally, let's apply tensor products to qubits to create *registers*.

A 2 qubit register:

$$|0\rangle \otimes |1\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

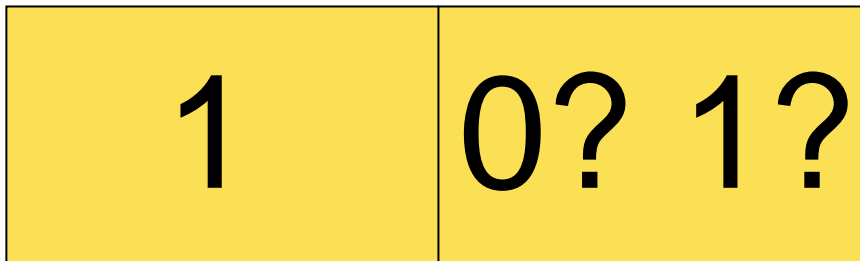
A 3 qubit register:

$$|0\rangle \otimes |1\rangle \otimes (a|0\rangle + b|1\rangle) = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ a \\ b \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The size of the register increases as 2^n , where n is the number of qubits!

Entanglement

- Suppose you have a two-bit register and you know the state of one of the bits. Can you know for certain the state of the second bit?



No! The bits are independent.

- What if you had a two-**q**ubit register?

- Let's say the 2-qubit register had an equal probability of being in two states.

$$\frac{1}{\sqrt{2}} (|00\rangle + |11\rangle)$$

- If you measure the first qubit and determine that it is a 1, you *know* that the second qubit must also be 1, and vice versa.
- The two qubits are in an **entangled** state.

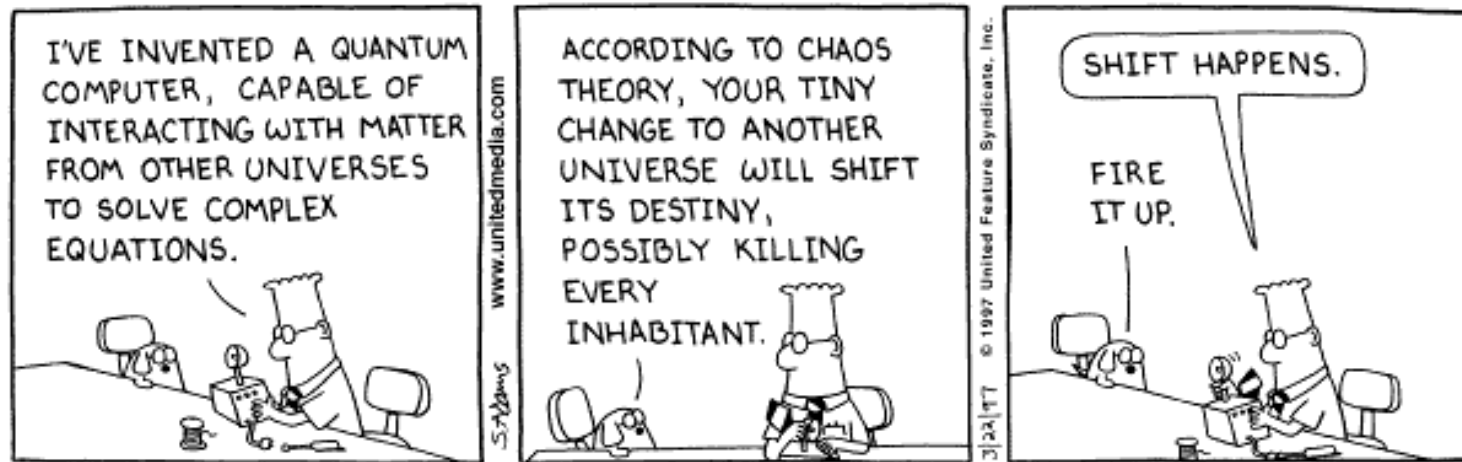
- Mathematically, this means that the register cannot be described by the tensor product of two qubits.
- To show this is the case, let's assume hypothetically that register *can* be written as a tensor product. Then:

$$\begin{aligned} \frac{1}{\sqrt{2}} (|00\rangle + |11\rangle) &= (p_0|0\rangle + p_1|1\rangle) \otimes (q_0|0\rangle + q_1|1\rangle) \\ &= p_0q_0|00\rangle + p_0q_1|01\rangle + p_1q_0|10\rangle + p_1q_1|11\rangle \end{aligned}$$

- This equation cannot be solved for p_0 , p_1 , q_0 , and q_1 . Therefore, the tensor product does not exist.

Consequences of Entanglement

Instantaneous Communication across the Universe.

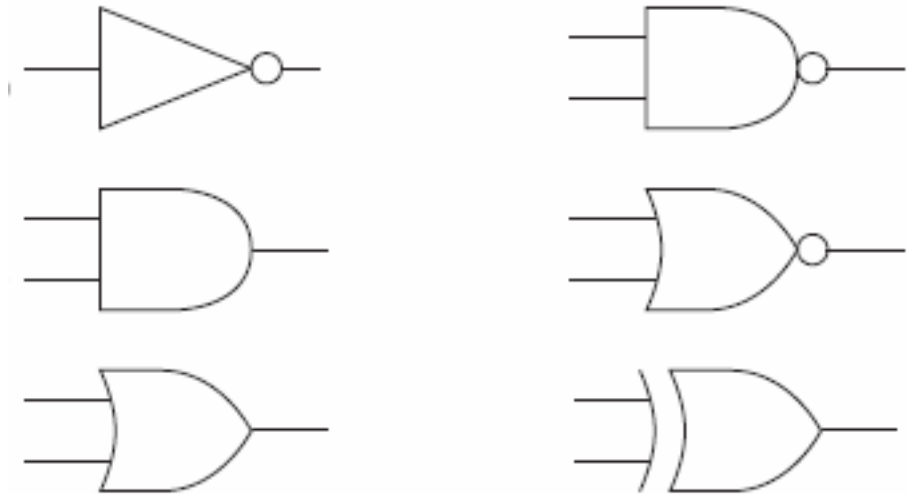


Copyright © 1997 United Feature Syndicate, Inc.
Redistribution in whole or in part prohibited

Not quite.

Quantum Gates

- Quantum computers use gates that are similar (in some respects) to the logic gates in classical computers.
- Many are based on the logical operators from Boolean Logic.



The classical NOT, AND, OR, NAND, NOR, and XOR gates.

Similarities

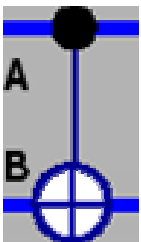
- Quantum gates take in a certain number of qubits as input and then output a certain number of qubits (it turns out the number of outputs and inputs must be the same, but more on this later).
- They follow consistent rules, such that a certain input will always result in the same output.

Similarities

- Gates can be represented by matrices.
- The column vectors of the matrix show the outputs.
- Here is the controlled-NOT gate:

$$|00\rangle = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad |01\rangle = \begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}, \quad |10\rangle = \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \end{bmatrix}, \quad |11\rangle = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}$$

$$C_{\text{not}}: \begin{array}{l} |00\rangle \rightarrow |00\rangle \\ |01\rangle \rightarrow |01\rangle \\ |10\rangle \rightarrow |11\rangle \\ |11\rangle \rightarrow |10\rangle \end{array} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$



The **cnot** gate acts similarly to a classical XOR, but with an extra output.

Differences

- However, there are quantum gates that could never exist in a classical computing environment.
- The simplest example of this is the $\sqrt{\text{NOT}}$ gate.
- This gate, when applied twice consecutively to a 0 yields a 1, and 1 when applied to a 0.

Differences

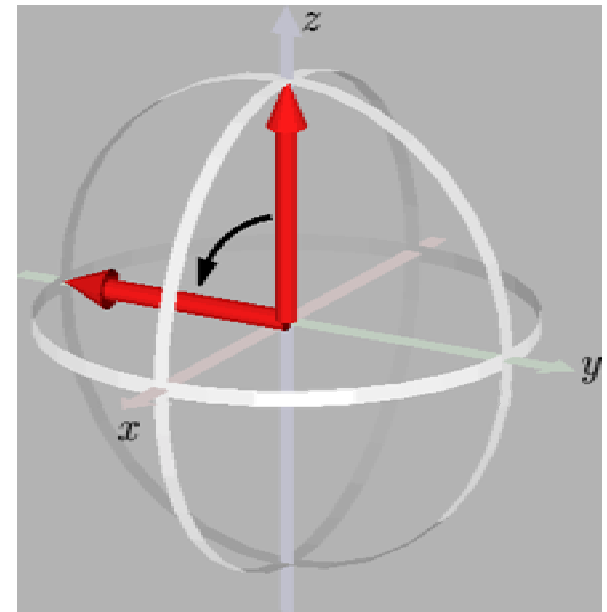
- When applied only once, the gate produces a result impossible in classical systems.

$$\sqrt{\text{NOT}}: \begin{array}{l} |0\rangle \rightarrow \frac{1}{\sqrt{2}} (|0\rangle - |1\rangle) \\ |1\rangle \rightarrow \frac{1}{\sqrt{2}} (|0\rangle + |1\rangle) \end{array} = \begin{bmatrix} \frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \\ \frac{-1}{\sqrt{2}} & \frac{1}{\sqrt{2}} \end{bmatrix}$$

- What does this mean?

Differences

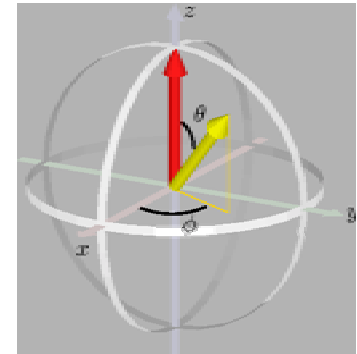
- The trick, so to speak, is that quantum gates are, most generally, not logical operators but rotational operators.
- Since we already have a vector representation of qubits and registers, let's treat a register as nothing more than a unit vector in \mathbb{R}^n where $n = 2^{\text{the number of qubits}}$.



Differences

- Now, a gate is simply a linear transformation with no scaling. The matrix representation becomes now even more appealing.
- Thus, there are gates that are just the rotation matrices.

$$\begin{pmatrix} e^{-i\phi/2} & 0 \\ 0 & e^{i\phi/2} \end{pmatrix} \begin{pmatrix} \cos(\theta/2) & -\sin(\theta/2) \\ \sin(\theta/2) & \cos(\theta/2) \end{pmatrix} \begin{pmatrix} \cos(\theta/2) & -i\sin(\theta/2) \\ -i\sin(\theta/2) & \cos(\theta/2) \end{pmatrix}$$



- One of the most important gates is the Walsh-Hadamard gate, which puts a qubit into a state of balanced superposition.

Differences

- The Walsh-Hadamard gate is represented by

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

- It is used to initialize registers, so that all qubits will be in superposition and thus able to represent both 0 and 1.

Differences

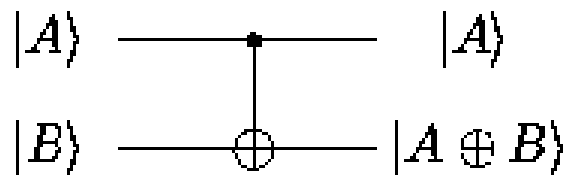
- The other primary difference is that quantum gates must be reversible.
- This a direct result of Schrödinger's equation.

$$H \psi = -i \hbar \frac{\partial \psi}{\partial t} = \frac{-\hbar^2}{2m} \nabla^2 \psi + V(0, X) \psi$$

- In any closed and isolate quantum system, there cannot be any loss of information such that any event cannot be reversed.

Differences

- Therefore, there must be as many outputs as inputs, so that the operation performed by the gate could be undone.
- This is why controlled-NOT gates are used instead of XOR gates.



Completeness

- It turns out that with the controlled-NOT gate and all one-qubit rotation gates, any possible operation can be carried out.
- These operators are referred to then as being complete.

Shor's Algorithm

- Factors large numbers in polynomial time
- Important applications in cryptanalysis

$n \neq p, p^m, 2m$



Choose $x \sum (x,n)=1$

If r is odd

Collapse quantum state and DFT

Quantum superposition

Compute order r of x



Compute $x^k \text{ mod } n$

If r is even



Compute $(n, x^{r/2} + 1)$ and $(n, x^{r/2} - 1)$

Requires quantum computer

Can be done on classical computer

Classical pre-processing

- Classically determine if n is a power of a prime or even
- Randomly select a number x and check to ensure that x and n are relatively prime using the Euclidean algorithm

Quantum processing

- Create an equal superposition of the integers 1 through k for some large k
- Compute $x^j \bmod n$ and store the result in a second register
- Measure the second register to collapse the first register into an equal superposition of integers spaced the period r apart

Register 1

1 through k



1 through k

$j+r*(1 \text{ through } m)$



$(1 \text{ through } m)/r$



s/r

Register 2

Processing

$x^{(1 \text{ through } k)} \bmod n$



Measurement

$x^{(j+r*(1 \text{ through } m))} \bmod n$



Discrete Fourier Transform

$m < k$

Measurement

Quantum processing

- Use the discrete Fourier transform to change the values in the first register to some superposition of the frequencies
- Measure the first register to obtain some multiple of $1/r$

Register 1

1 through k



1 through k

$j+r*(1 \text{ through } m)$



$(1 \text{ through } m)/r$



s/r

Register 2

Processing

$x^{(1 \text{ through } k)} \text{ mod } n$



Measurement

$x^{(j+r*(1 \text{ through } m))} \text{ mod } n$



Discrete Fourier Transform

$m < k$

$n \neq p, p^m, 2m$



Choose $x \sum (x,n)=1$

If r is odd

Collapse quantum state and DFT

Quantum superposition

Compute order r of x



Compute $x^k \text{ mod } n$

If r is even

Compute $(n, x^{r/2} + 1)$ and $(n, x^{r/2} - 1)$

Requires quantum computer

Can be done on classical computer

Classical post-processing

1. Determine r
2. If r is even

$$\left(x^{r/2} + 1\right)\left(x^{r/2} - 1\right) = x^r - 1 = 0 \pmod{n}$$

3. Compute

$$\left(x^{r/2} + 1, n\right) \quad \text{and} \quad \left(x^{r/2} - 1, n\right)$$

4. Repeat if necessary