Classical and Quantum Gates

Peter Young

(Dated: April 25, 2020)

Now, finally, we get to computation!

The elementary circuit elements which acts on the data in a computer are called gates. In this handout we will first discuss classical gates and then go on to describe quantum gates.

I. CLASSICAL GATES

Data in a classical digital computer is in the form of bits, x, which take values 0 or 1. The only operation involving a single classical bit, i.e. the only 1-bit classical gate, is NOT which takes 0 to 1 and vice versa.

Of particular interest are 2 bit gates, the most common ones being

	In	Out	
AND	00	0	$x \wedge y$
	01	0	
	10	0	
	11	1	
OR	In	Out	$x \vee y$
	00	0	
	01	1	
	10	1	
	11	1	
XOR	In	Out	$x \oplus y$
	00	0	
	01	1	
	10	1	
	11	0	

(1)

These have two input bits and one output bit. For the AND gate the result is 0 unless both inputs are 1. For the OR gate the result is 0 unless one or both of the inputs are 1. The XOR gate only differs from the OR gate in giving zero if *both* the inputs are 1.

Note that AND gives the same results as multiplication of the bits xy. The XOR operation is equivalent to addition of the bits modulo 2, i.e. $x + y \,(\mathrm{mod}\ 2)$. To see this, note that the modulo

operation gives the remainder after integer division. For example, since $13 = (5 \times 2) + 3$ we have $13 (\mod 5) = 3$. Referring to the XOR gate consider the case $x = y = 1$, so we have $1 + 1 (\mod 2) = 0$, which is the value of XOR in this case. It is trivial to see that XOR is addition modulo 2 for the other values of x and y. For convenience of notation $x + y (\mod 2)$ is written as $x \oplus y$.

One can show that the AND, NOT and OR gates form a **universal set** which means that any logical operation on a arbitrary number of qubits can be expressed in terms of these gates. Thus, classically, we only need 1-bit and 2-bit gates to perform any operation.

However, we cannot directly take over gates like AND, OR and XOR to a quantum computer for the following reason. A gate in a quantum computer will be implemented by a unitary operator acting on a small number of qubits. A unitary operator has the property that $U^{-1} = U^\dagger$. Now U^{-1} performs the inverse operation, and since $U^\dagger$ is well defined the inverse operation must exist. **Thus, quantum gates must be reversible.**

However, AND, OR and XOR can not be reversible because they have a different number of outputs and inputs. Suppose, for example, we know that the output from an OR gate is 1, and want to know what is the input. We can't say because there are three possible inputs, 01, 10 and 11, which give this output.

Thus, a major change in going from classical to quantum computing will be having to deal with *reversible* computation. We will consider first reversible classical computation before doing the quantum case.

Clearly a necessary condition for a gate to be reversible is that it has the same number of input and output bits. The 1-bit NOT gate has one input and one output, and is reversible since acting twice gives back the original bit, i.e. $(\mathrm{NOT})^2 = \mathrm{IDENTITY}$, so $(\mathrm{NOT})^{-1} = \mathrm{NOT}$, i.e. NOT is its own inverse.

We will now consider a reversible, classical, 2-bit gate, the quantum analog of which will play an important role in quantum computing. This is the controlled-NOT, or CNOT gate. It is similar to XOR except that it has a second output bit which is one of the (unchanged) input qubits. As we shall see, this simple change suffices to make the gate reversible.

One way of representing the action of CNOT is

$$\begin{pmatrix} x \\ y \end{pmatrix} \longrightarrow \begin{pmatrix} x \\ x \oplus y \end{pmatrix}. \tag{2}$$

The first (upper) bit is called the control bit. This is unchanged by the action of CNOT. The second (lower) bit is called the target bit, and the effect of the XOR operation $x \oplus y$ is to flip y if

$x = 1$ and to leave y alone if $x = 0$. Hence, as far as the target bit is concerned, the gate is indeed a controlled NOT, since the NOT acts if x, the control bit, is 1, and does not act if $x = 0$. The truth table is as follows:

x	y	x'	y'
0	0	0	0
0	1	0	1
1	0	1	1
1	1	1	0

. (3)

It is useful to represent the CNOT gate by a diagram, as shown in Fig. 1. The input is on the left and the output on the right. The upper line is the control bit, and has value x on input, while the lower line is the target bit and has value y on input. On output, the control qubit is unchanged and the target qubit is the exclusive or (XOR) of x and y.

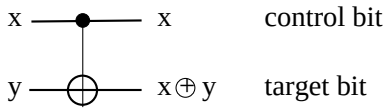


FIG. 1: The CNOT gate. The input is on the left and the output on the right.

It is easy to see that CNOT is reversible since, if we act twice, we get back the original input because

$$\begin{pmatrix} x \\ y \end{pmatrix} \overset{\text{CNOT}}{\longrightarrow} \begin{pmatrix} x \\ x \oplus y \end{pmatrix} \overset{\text{CNOT}}{\longrightarrow} \begin{pmatrix} x \\ x \oplus x \oplus y \end{pmatrix} = \begin{pmatrix} x \\ y \end{pmatrix}. \tag{4}$$

The last line follows because $x \oplus x = 0$ since $0 + 0 = 0$ and $1 + 1 = 0 \,(\bmod 2)$. Thus CNOT is its own inverse. It can therefore be regarded as a reversible version of XOR.

Note that to be reversible it is not required that the inverse operator is the same as the original operator, only that the inverse operator exists. However, it turns out that *most* quantum gates we consider will be their own inverse.

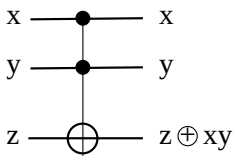


FIG. 2: The Toffoli gate. This has two control bits x and y and one target bit z. On output the control bits are unchanged and the target bit is flipped if both control bits are 1, so $z \to z \oplus xy$.

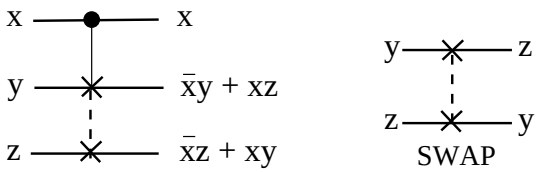


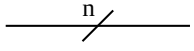
FIG. 3: Left: the Fredkin gate. This is a controlled-swap gate. If the upper (control) bit is 1 then the two lower (target) bits are swapped, and otherwise the target bits are unchanged. $\overline{x} \equiv 1 - x$ is the complement of x. Right: the elemental SWAP gate.

We mentioned above that the 1-bit (NOT) gate and a set of irreversible 2-bit gates (AND and OR) together form universal set, which means that any logical operation on an arbitrary number of bits can be constructed out of these gates. The question we now ask is whether 1-bit and 2-bit *reversible* gates are universal. The answer is no. Classically one also needs a 3-bit gate such as the Toffoli gate shown in Fig. 2 or the Fredkin gate shown in Fig. 3.

Amazingly we shall see that 3-qubit gates are *not* needed quantum mechanically. In fact we show see later in the course how to explicitly build the Toffoli gate out of 1-qubit and 2-qubit gates. It is not possible to do this classically. We shall see that quantum mechanics allows for a big range of 1-qubit gates, whereas we have already noted that classically the only 1-bit gate is NOT. It is this wide range of possibilities for 1-qubit gates that allows us to construct a quantum mechanical Toffoli gate out of 1-qubit and 2-qubit gates.

II. QUANTUM GATES

Following D. Deutsch we represent the action of quantum gates by a circuit. The circuit comprises a set of qubits in some initial state, acted on by gates and ending up in a final state. Each qubit is represented by a line in the circuit diagram and time runs from left to right, see e.g. Fig. 4.

Sometimes we will indicate a set of n qubits (called a register) compactly by a single line with a slash through it as follows: 

.

Quantum circuits have the following properties:

- There are no loops, because qubits can't go back in time.
- Lines can't splay out (fan out) because of the no-cloning theorem.
- Similarly lines can't merge.

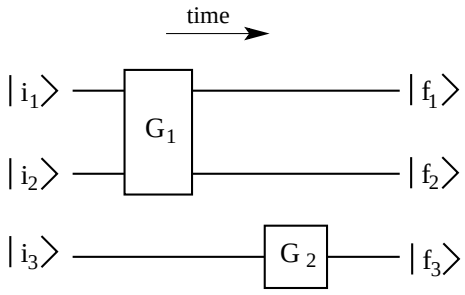
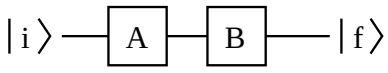


FIG. 4: A schematic circuit with three qubits and two gates. Time runs from left to right. The initial state of the qubits is $|i_1\rangle \otimes |i_2\rangle \otimes |i_3\rangle$ and the final state is $|f_1\rangle \otimes |f_2\rangle \otimes |f_3\rangle$.

- Gates and circuits are *linear*. We evaluate the effect of the circuit on an initial state which is a computational basis state. However, if the initial qubits are in a superposition of computational basis states, then the final qubits, after the circuit has acted, are easily computed since they are simply the corresponding linear superposition of outputs for each of the computational basis state inputs,

Circuits have several gates acting in succession on a qubit and it is important to understand the order in which they act. Unfortunately, this can be confusing. By convention, in diagrams time is from left to right, so in the diagram



A (the leftmost gate) acts first then B. However, when writing operator expressions, these work from right to left, so, the above diagram corresponds to

$$|f\rangle = BA|i\rangle, \tag{5}$$

in which A is on the *right*. You simply have to get used to this reversal of order when going from circuit diagrams to operator expressions.

Now we describe some commonly used quantum gates, recalling that quantum gates are unitary operators and so must be reversible.

Firstly we consider 1-qubit gates.

- NOT (corresponds to the Pauli X operator)

$$\begin{matrix} X|0\rangle = |1\rangle \\ X|1\rangle = |0\rangle \end{matrix}, \qquad X = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}. \tag{6}$$

- Phase flip (corresponds to the Pauli Z operator)

$$\begin{array}{l} Z|0\rangle = |0\rangle \\ Z|1\rangle = -|1\rangle \end{array}, \qquad Z = \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix}. \tag{7}$$

  In the physics literature X and Z are called Pauli spin matrices and are written $\sigma_x \equiv X$, and $\sigma_z \equiv Z$. There is also a third Pauli spin matrix

$$\sigma_y \equiv Y = iXZ = \begin{pmatrix} 0 & -i \\ i & 0 \end{pmatrix}. \tag{8}$$

- Hadamard

  The Hadamard gate H will be very important.

$$H = \frac{1}{\sqrt{2}}(X + Z) = \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{9}$$

  Note that $H^2 = \mathbb{1}$, and similarly $X^2 = Y^2 = Z^2 = \mathbb{1}$. Now a matrix which squares to the identity has eigenvalues ± 1. To see this note that if $\vec{x}$ is an eigenvector of A with eigenvalue λ then

$$A^2\vec{x} = A\left(A\vec{x}\right) = A\lambda\vec{x} = \lambda A\vec{x} = \lambda^2\vec{x}. \tag{10}$$

  But if $A^2 = \mathbb{1}$ then it follows that $\lambda^2 = 1$ and so $\lambda = \pm 1$.

  We need to become familiar with the action of H on computational basis states. This is:

$$\begin{aligned} H|0\rangle =& |+\rangle \equiv \frac{1}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) \\ H|1\rangle =& |-\rangle \equiv \frac{1}{\sqrt{2}}\left(|0\rangle - |1\rangle\right). \end{aligned} \tag{11}$$

  Combining these two equations, the action of H on a computational basis state $|x\rangle$ is seen to be

$$H|x\rangle = \frac{1}{\sqrt{2}}\left(\,|0\rangle + (-1)^x|1\rangle\,\right), \tag{12}$$

  for both values of x, 0 and 1.

A crucial point is that these gates are linear, and so they act in the same way on a superposition. For example:

$$H\left[\,\alpha|0\rangle + \beta|1\rangle\,\right] = \frac{\alpha}{\sqrt{2}}\left(|0\rangle + |1\rangle\right) + \frac{\beta}{\sqrt{2}}\left(|0\rangle - |1\rangle\right) = \left(\frac{\alpha+\beta}{\sqrt{2}}\right)|0\rangle + \left(\frac{\alpha-\beta}{\sqrt{2}}\right)|1\rangle. \tag{13}$$

What is the most general 1-qubit gate? To answer this question we first note that any 2×2 matrix can be expressed as a linear combination of the three Pauli matrices plus the identity. To see

this note that X, Y, Z and $\mathbb{1}$ are linearly independent (we can't write one as a linear combination of the others). Also a general 2×2 matrix

$$A = \begin{pmatrix} t & u \\ v & w \end{pmatrix} \tag{14}$$

has 4 complex elements, and so a total of 8 real parameters. If we write

$$A = a_0 \mathbb{1} + a_x X + a_y Y + a_z Z \tag{15}$$

then there are also 4 complex coefficients (8 real parameters). Hence there are just the right number of coefficients to specify any 2×2 matrix, so Equation (15) is a general expression for a 2×2 matrix.

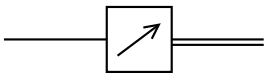
In the case where A is also unitary then $A^\dagger A = \mathbb{1}$. Since X, Y and Z are Hermitian, $X^2 = Y^2 = Z^2 = \mathbb{1}$, and different Pauli matrices anti-commute (e.g. $XZ + ZX = 0$), the unitarity condition gives

$$|a_0|^2 + |a_x|^2 + |a_y|^2 + |a_z|^3 = 1, \tag{16}$$

and so Eqs. (15) and (16) characterize a general 2×2 unitary matrix A, i.e. a general 1-qubit gate.

We also need to consider measurement gates, in which a classical measurement of a qubit takes place. By convention, measurements are made in the computational basis. The Pauli spin matrices are defined such that Z is diagonal, so we can also call the computational basis the Z-basis.

The result of the measurement is a classical bit. In the circuit diagrams we indicate a classical bit by a double line, and so a measurement gate is indicated as follows:



A measurement gate

Next we consider 2-qubit gates, the most important of which by far is the CNOT. We already met the classical CNOT gate in Fig. 1. In the quantum case, if initially the qubits are in a computational basis state, then the action of the CNOT is the same as classically, i.e. as shown in Fig. 5.

The CNOT gate has the matrix representation

$$U_{\text{CNOT}} = \begin{array}{c} \\ \langle 00| \\ \langle 01| \\ \langle 10| \\ \langle 11| \end{array} \begin{array}{c} \begin{array}{cccc} |00\rangle & |01\rangle & |10\rangle & |11\rangle \end{array} \\ \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{pmatrix} \end{array}. \tag{17}$$

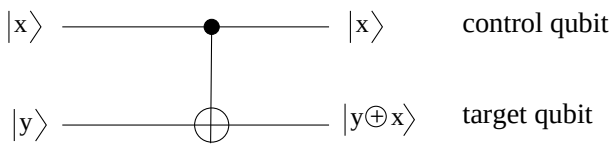


FIG. 5: A quantum CNOT gate. If the initial state of the qubits (on the left) is a computational basis state, then the action of the quantum CNOT gate is the same as that of the classical CNOT shown in Fig. 1. The upper line represents the control qubit and the lower line the target qubit.

In this tensor product the control qubit is to the left. The target qubit (to the right) is flipped if the control qubit is 1 (so, relative to the identity matrix, columns 3 and 4 are interchanged). We can also write this in terms of 2×2 blocks as follows

$$U_{CNOT} = \begin{pmatrix} \mathbb{1} & 0 \\ 0 & X \end{pmatrix}. \tag{18}$$

The quantum aspect appears if we input (on the left) a linear combination of basis states. Suppose we set the target (lower) qubit to $|0\rangle$. Then if the control qubit is initially $|0\rangle$ the final state of the 2-qubit system is $|00\rangle$, because the target qubit is not flipped (we take the control qubit to be the left one). If the control qubit is initially $|1\rangle$ then the final state of the 2-qubit system is $|11\rangle$ because the target qubit *is* flipped. Hence, by linearity, if the initial state of the control qubit is the superposition $\alpha|0\rangle + \beta|1\rangle$, then the final state of the 2-qubit system is $\alpha|00\rangle + \beta|11\rangle$, see Fig. 6.

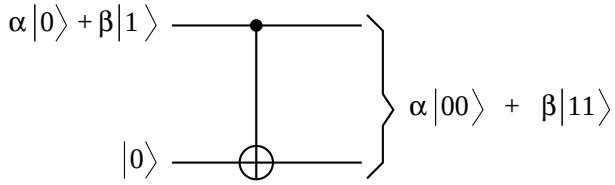


FIG. 6: The action of the CNOT gate when the upper (control) qubit is initially in a superposition $\alpha|0\rangle+\beta|1\rangle$, and the lower (target) qubit is initially $|0\rangle$. By linearity, the final state is α times the result of inputting $|0\rangle$ in the control qubit plus β times the result of inputting $|1\rangle$, i.e. $\alpha|00\rangle + \beta|11\rangle$.

Note that if $\alpha = 0$ (so $\beta = 1$ since $|\alpha|^2 + |\beta|^2 = 1$) or $\alpha = 1$ ($\beta = 0$), the final state is a clone of the initial state of the control qubit. However, for a general input state, the final state of the two qubits, $\alpha|00\rangle + \beta|11\rangle$, is not a clone of the initial state of the control qubit which would be $(\alpha|0\rangle + \beta|1\rangle) \otimes (\alpha|0\rangle + \beta|1\rangle) = \alpha^2|00\rangle + \alpha\beta(|01\rangle + |10\rangle) + \beta^2|11\rangle$. Hence there is no violation of the no-cloning theorem which states that a *general* quantum state can not be cloned.

In this course, we will specify the action of a gate by its action on an initial computational basis state. If we denote a qubit by a Latin letter, e.g. $|x\rangle$, we mean that this is a computational basis state and x takes values 0 or 1. General quantum states, i.e. superpositions of computational basis states, will be indicated by Greek letters, e.g. $|\psi\rangle$.

As already mentioned above, we do not need 3-qubit gates for quantum computing. More precisely, the statement is that one can generate an arbitrary unitary transformation (to a specified level of accuracy) on a arbitrary number of qubits, using only CNOT and single-qubit gates. I do not prove this result but refer interested students to a more advanced text [1]. It is fortunate that we don't need 3-qubit gates given the difficulty of making quantum circuits.

It is useful to mention here that one has to be careful when dealing with superpositions, and one's initial intuition as to the final result may be incorrect. As an example, consider the circuit in Fig. 7.

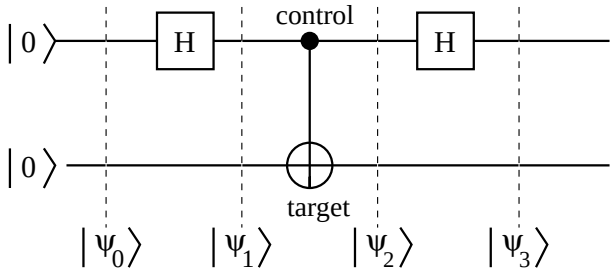


FIG. 7: The initial state of both qubits is $|0\rangle$. What is the final state $|\psi_3\rangle$? Equation (19) gives the state of the two qubits at each stage. The end result is that the two qubits are entangled and, in contrast to what one might have thought, the control qubit has a non-zero amplitude to be flipped relative to its initial state, i.e. to be in state $|1\rangle$.

Since $H^2 = \mathbb{1}$ and the CNOT gate doesn't change the control (upper) qubit, one might think that the final state of the control qubit would be the same as the initial state, i.e. $|0\rangle$. However this is not correct because the control and target qubits become entangled. Let's go through each stage of the circuit using the notation in Fig. 7, and taking the left-hand qubit in the formulae to

be the control qubit:

$$
\begin{aligned}
|\psi_0\rangle &= |00\rangle \\
|\psi_1\rangle &= \frac{1}{\sqrt{2}}\,(\,|00\rangle + |10\rangle\,) \\
|\psi_2\rangle &= \frac{1}{\sqrt{2}}\,(\,|00\rangle + |11\rangle\,) \\
|\psi_3\rangle &= \frac{1}{2}\,(\,|00\rangle + |10\rangle + |01\rangle - |11\rangle\,) \\
&= \left[\,|0\rangle_c \otimes \left(\frac{|0\rangle_t + |1\rangle_t}{\sqrt{2}}\right) + |1\rangle_c \otimes \left(\frac{|0\rangle_t - |1\rangle_t}{\sqrt{2}}\right)\right],
\end{aligned}
\tag{19}
$$

where in the last expression we indicate explicitly which qubit is the control qubit ("c"), and which the target qubit ("t"). We see that, contrary to what one might have initially guessed, there is an amplitude for the control qubit to be in state $|1\rangle$ because of its entanglement with the target qubit.

We have noted that the Pauli operators X, Y and Z, and the Hadamard operator have eigenvalues ± 1. Later in the course, when we consider the important topic of quantum error correction, we will encounter combinations of these operators on different qubits which also have ± 1 eigenvalues. We will now describe a convenient way of measuring such operators. Let us denote the operator by U. It will have an eigenvalue $+1$ with eigenvalue $|\psi_+\rangle$ and an eigenvalue -1 with eigenvector $|\psi_-\rangle$. We would like to investigate the qubit (or qubits) to determine which eigenstate of U it is in, or, if it is in a linear superposition, to project by measurement on to one of the eigenstates, and know which one.

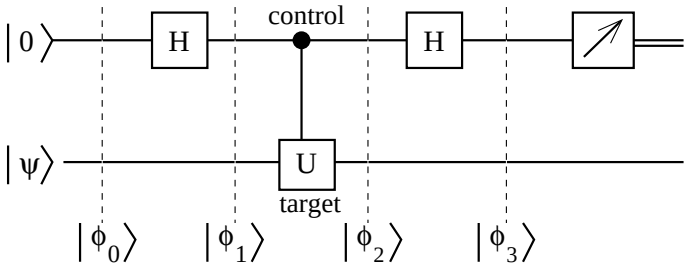


FIG. 8: A circuit with a control-U gate in which the control (upper) qubit is surrounded by Hadamards. U is an operator with eigenvalues ± 1 and corresponding eigenvectors $|\psi_+\rangle$ and $|\psi_-\rangle$. As shown in the text, if a measurement of the upper qubit gives $|0\rangle$ then the lower qubit will be in state $|\psi_+\rangle$, and if the measurement gives $|1\rangle$ then the lower qubit will be in state $|\psi_-\rangle$. The states $|\phi_i\rangle\,(i = 0, 1, 2, 3)$ are described in the text.

A convenient way is to use the circuit shown in Fig. 8, which has a control-U gate. If the control qubit is 1 the effect on the target qubit is

$$
U|\psi_+\rangle = |\psi_+\rangle, \quad U|\psi_-\rangle = -|\psi_-\rangle. \tag{20}
$$

If the control qubit is 0 then the target qubit is unchanged.

The lower (target) qubit is initially in state $|\psi\rangle$, which can be written as a linear combination of the two eigenvectors

$$|\psi\rangle = \alpha_+|\psi_+\rangle + \alpha_-|\psi_-\rangle\,, \tag{21}$$

and so, including the upper (control) qubit which is initially in state $|0\rangle$, the initial state of the circuit (on the left of Fig. 8) is

$$|\phi_0\rangle = \alpha_+|0\,\psi_+\rangle + \alpha_-|0\,\psi_-\rangle\,. \tag{22}$$

After the first Hadamard on the upper qubit the state is

$$|\phi_1\rangle = \frac{\alpha_+}{\sqrt{2}}\,(\,|0\,\psi_+\rangle + |1\,\psi_+\rangle\,) + \frac{\alpha_-}{\sqrt{2}}\,(\,|0\,\psi_-\rangle + |1\,\psi_-\rangle\,)\,. \tag{23}$$

The effect of the control-U gate on the target qubit is given by Eq. (20) when the control qubit is 1 and has no effect if the control qubit is 0. Hence, after the control-U gate, the state is

$$|\phi_2\rangle = \frac{\alpha_+}{\sqrt{2}}\,(\,|0\,\psi_+\rangle + |1\,\psi_+\rangle\,) + \frac{\alpha_-}{\sqrt{2}}\,(\,|0\,\psi_-\rangle - |1\,\psi_-\rangle\,)\,. \tag{24}$$

Applying the second (rightmost) Hadamard to the upper qubit we get

$$|\phi_3\rangle = \alpha_+|0\,\psi_+\rangle + \alpha_-|1\,\psi_-\rangle\,. \tag{25}$$

Hence if a measurement of the upper qubit gives $|0\rangle$ (which it does with probability $|\alpha_+|^2$) the lower qubit will be in state $|\psi_+\rangle$, and if the measurement gives $|1\rangle$ (probability is $|\alpha_-|^2$) the lower qubit will be in state $|\psi_-\rangle$. We see that measuring the control qubit projects the target qubit onto an eigenstate of U and tells us which one.

We will return to the circuit in Fig. 8 later in the course when we discuss quantum error correction.

---

[1] M. A. Nielsen and I. L. Chuang, *Quantum Computation and Quantum Information* (Cambridge University Press, Cambridge, England, 2000).