Exercise 4.13: (Circuit identities) It is useful to be able to simplify circuits by inspection, using well-known identities. Prove the following three identities:

$$HXH = Z; \quad HYH = -Y; \quad HZH = X. \tag{4.18}$$

Exercise 4.14: Use the previous exercise to show that $HTH = R_x(\pi/4)$, up to a global phase.

Exercise 4.15: (Composition of single qubit operations) The Bloch representation gives a nice way to visualize the effect of composing two rotations.

(1) Prove that if a rotation through an angle β_1 about the axis $\hat{n}_1$ is followed by a rotation through an angle β_2 about an axis $\hat{n}_2$, then the overall rotation is through an angle β_{12} about an axis $\hat{n}_{12}$ given by

$$c_{12} = c_1 c_2 - s_1 s_2 \, \hat{n}_1 \cdot \hat{n}_2 \tag{4.19}$$

$$s_{12} \hat{n}_{12} = s_1 c_2 \hat{n}_1 + c_1 s_2 \hat{n}_2 - s_1 s_2 \, \hat{n}_2 \times \hat{n}_1 \, , \tag{4.20}$$

where $c_i = \cos(\beta_i/2)$, $s_i = \sin(\beta_i/2)$, $c_{12} = \cos(\beta_{12}/2)$, and $s_{12} = \sin(\beta_{12}/2)$.

(2) Show that if $\beta_1 = \beta_2$ and $\hat{n}_1 = \hat{z}$ these equations simplify to

$$c_{12} = c^2 - s^2 \, \hat{z} \cdot \hat{n}_2 \tag{4.21}$$

$$s_{12} \hat{n}_{12} = sc(\hat{z} + \hat{n}_2) - s^2 \, \hat{n}_2 \times \hat{z} \, , \tag{4.22}$$

where $c = c_1$ and $s = s_1$.

Symbols for the common single qubit gates are shown in Figure 4.2. Recall the basic properties of quantum circuits: time proceeds from left to right; wires represent qubits, and a '/' may be used to indicate a bundle of qubits.

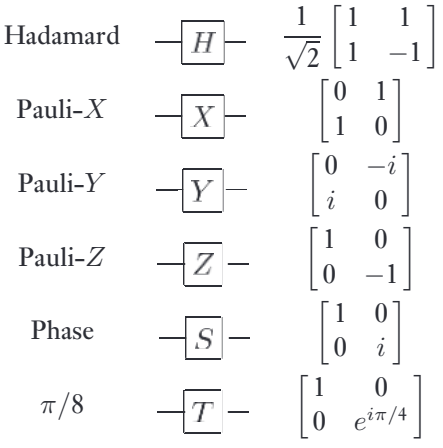


Figure 4.2. Names, symbols, and unitary matrices for the common single qubit gates.

4.3 Controlled operations

'If A is true, then do B'. This type of *controlled operation* is one of the most useful in computing, both classical and quantum. In this section we explain how complex controlled operations may be implemented using quantum circuits built from elementary operations.

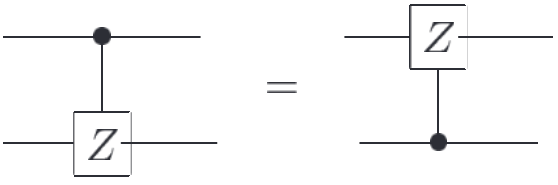
in the computational basis?

Exercise 4.17: (Building CNOT from controlled-Z gates) Construct a CNOT gate from one controlled-Z gate, that is, the gate whose action in the computational basis is specified by the unitary matrix

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & -1 \end{bmatrix},$$

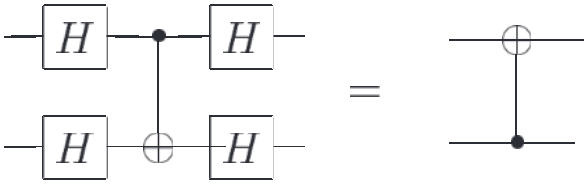
and two Hadamard gates, specifying the control and target qubits.

Exercise 4.18: Show that



Exercise 4.19: (CNOT action on density matrices) The CNOT gate is a simple permutation whose action on a density matrix ρ is to rearrange the elements in the matrix. Write out this action explicitly in the computational basis.

Exercise 4.20: (CNOT basis transformations) Unlike ideal classical gates, ideal quantum gates do not have (as electrical engineers say) 'high-impedance' inputs. In fact, the role of 'control' and 'target' are arbitrary – they depend on what basis you think of a device as operating in. We have described how the CNOT behaves with respect to the computational basis, and in this description the state of the control qubit is not changed. However, if we work in a different basis then the control qubit *does* change: we will show that its phase is flipped depending on the state of the 'target' qubit! Show that



Introducing basis states $|\pm\rangle \equiv (|0\rangle \pm |1\rangle)/\sqrt{2}$, use this circuit identity to show that the effect of a CNOT with the first qubit as control and the second qubit as target is as follows:

$$|+\rangle|+\rangle \rightarrow |+\rangle|+\rangle \tag{4.24}$$
$$|-\rangle|+\rangle \rightarrow |-\rangle|+\rangle \tag{4.25}$$
$$|+\rangle|-\rangle \rightarrow |-\rangle|-\rangle \tag{4.26}$$
$$|-\rangle|-\rangle \rightarrow |+\rangle|-\rangle. \tag{4.27}$$

Thus, with respect to this new basis, the state of the target qubit is not changed, while the state of the control qubit is flipped if the target starts as $|-\rangle$, otherwise

performed, conditional on the final work qubit being set to one. That is, U is applied if and only if all of c_1 through c_n are set. Finally, the last part of the circuit just reverses the steps of the first stage, returning all the work qubits to their initial state, $|0\rangle$. The combined result, therefore, is to apply the unitary operator U to the target qubit, if and only if all the control bits c_1 through c_n are set, as desired.

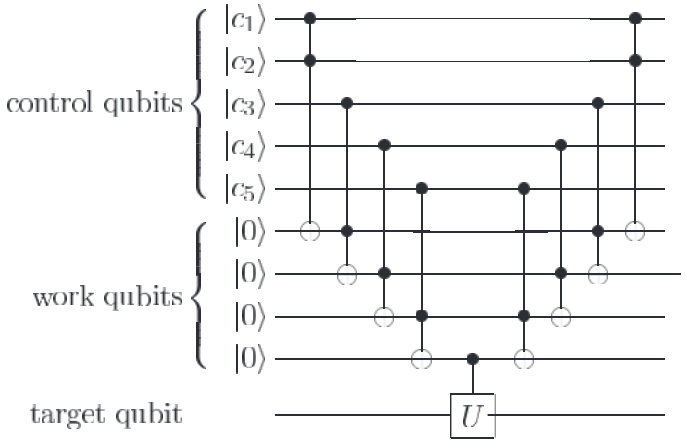


Figure 4.10. Network implementing the $C^n(U)$ operation, for the case $n = 5$.

Exercise 4.28: For $U = V^2$ with V unitary, construct a $C^5(U)$ gate analogous to that in Figure 4.10, but using no work qubits. You may use controlled-V and controlled-$V^\dagger$ gates.

Exercise 4.29: Find a circuit containing $O(n^2)$ Toffoli, CNOT and single qubit gates which implements a $C^n(X)$ gate (for $n > 3$), using no work qubits.

Exercise 4.30: Suppose U is a single qubit unitary operation. Find a circuit containing $O(n^2)$ Toffoli, CNOT and single qubit gates which implements a $C^n(U)$ gate (for $n > 3$), using no work qubits.

In the controlled gates we have been considering, conditional dynamics on the target qubit occurs if the control bits are set to *one*. Of course, there is nothing special about one, and it is often useful to consider dynamics which occur conditional on the control bit being set to zero. For instance, suppose we wish to implement a two qubit gate in which the second ('target') qubit is flipped, conditional on the first ('control') qubit being set to zero. In Figure 4.11 we introduce a circuit notation for this gate, together with an equivalent circuit in terms of the gates we have already introduced. Generically we shall use the open circle notation to indicate conditioning on the qubit being set to zero, while a closed circle indicates conditioning on the qubit being set to one.

A more elaborate example of this convention, involving three control qubits, is illustrated in Figure 4.12. The operation U is applied to the target qubit if the first and third qubits are set to zero, and the second qubit is set to one. It is easy to verify by inspection that the circuit on the right hand side of the figure implements the desired operation. More generally, it is easy to move between circuits which condition on qubits being set

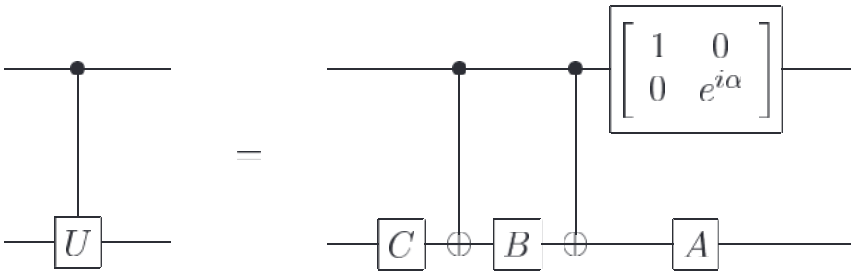


Figure 4.6. Circuit implementing the controlled-U operation for single qubit U. α, A, B and C satisfy $U = \exp(i\alpha)AXBXC$, $ABC = I$.

introduce a special circuit notation for them, illustrated in Figure 4.7. For the following we assume that $k = 1$, for simplicity. Larger k can be dealt with using essentially the same methods, however for $k \geq 2$ there is the added complication that we don't (yet) know how to perform arbitrary operations on k qubits.

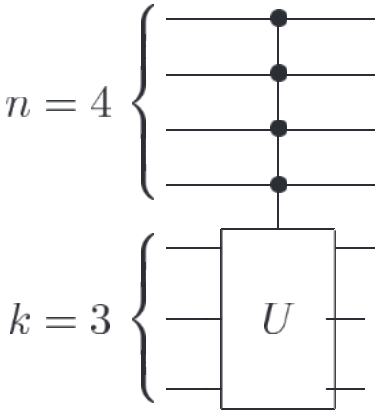


Figure 4.7. Sample circuit representation for the $C^n(U)$ operation, where U is a unitary operator on k qubits, for $n = 4$ and $k = 3$.

Suppose U is a single qubit unitary operator, and V is a unitary operator chosen so that $V^2 = U$. Then the operation $C^2(U)$ may be implemented using the circuit shown in Figure 4.8.

Exercise 4.21: Verify that Figure 4.8 implements the $C^2(U)$ operation.

Exercise 4.22: Prove that a $C^2(U)$ gate (for any single qubit unitary U) can be constructed using at most eight one-qubit gates, and six controlled-NOTs.

Exercise 4.23: Construct a $C^1(U)$ gate for $U = R_x(\theta)$ and $U = R_y(\theta)$, using only CNOT and single qubit gates. Can you reduce the number of single qubit gates needed in the construction from three to two?

The familiar Toffoli gate is an especially important special case of the $C^2(U)$ operation,

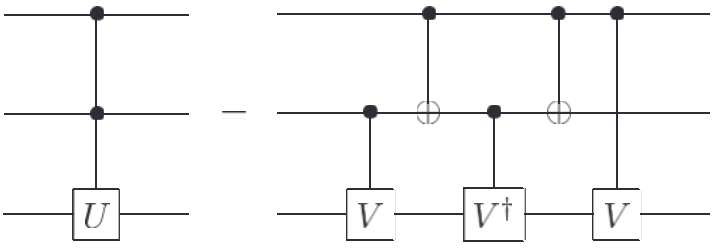


Figure 4.8. Circuit for the $C^2(U)$ gate. V is any unitary operator satisfying $V^2 = U$. The special case $V \equiv (1-i)(I+iX)/2$ corresponds to the Toffoli gate.

the case $C^2(X)$. Defining $V \equiv (1-i)(I+iX)/2$ and noting that $V^2 = X$, we see that Figure 4.8 gives an implementation of the Toffoli gate in terms of one and two qubit operations. From a classical viewpoint this is a remarkable result; recall from Problem 3.5 that one and two bit classical reversible gates are not sufficient to implement the Toffoli gate, or, more generally, universal computation. By contrast, in the quantum case we see that one and two qubit reversible gates are sufficient to implement the Toffoli gate, and will eventually prove that they suffice for universal computation.

Ultimately we will show that any unitary operation can be composed to an arbitrarily good approximation from just the Hadamard, phase, controlled-NOT and $\pi/8$ gates. Because of the great usefulness of the Toffoli gate it is interesting to see how it can be built from just this gate set. Figure 4.9 illustrates a simple circuit for the Toffoli gate made up of just Hadamard, phase, controlled-NOT and $\pi/8$ gates.

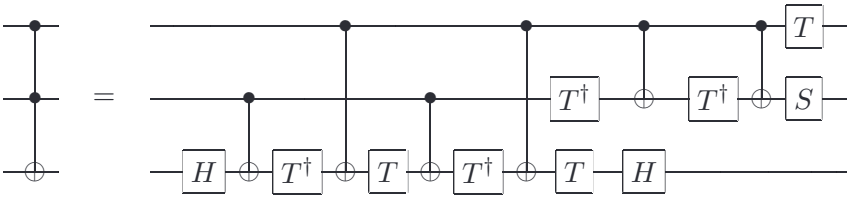


Figure 4.9. Implementation of the Toffoli gate using Hadamard, phase, controlled-NOT and $\pi/8$ gates.

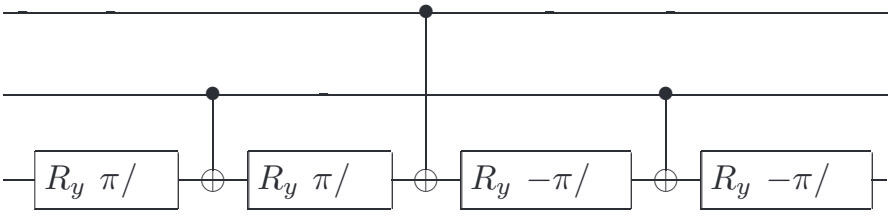
Exercise 4.24: Verify that Figure 4.9 implements the Toffoli gate.

Exercise 4.25: (Fredkin gate construction) Recall that the Fredkin (controlled-swap) gate performs the transform

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{4.30}$$

(1) Give a quantum circuit which uses three Toffoli gates to construct the Fredkin gate (*Hint*: think of the swap gate construction – you can control each gate, one at a time).
(2) Show that the first and last Toffoli gates can be replaced by CNOT gates.
(3) Now replace the middle Toffoli gate with the circuit in Figure 4.8 to obtain a Fredkin gate construction using only six two-qubit gates.
(4) Can you come up with an even simpler construction, with only five two-qubit gates?

Exercise 4.26: Show that the circuit:



differs from a Toffoli gate only by relative phases. That is, the circuit takes $|c_1, c_2, t\rangle$ to $e^{i\theta(c_1,c_2,t)}|c_1, c_2, t \oplus c_1 \cdot c_2\rangle$, where $e^{i\theta(c_1,c_2,t)}$ is some relative phase factor. Such gates can sometimes be useful in experimental implementations, where it may be much easier to implement a gate that is the same as the Toffoli up to relative phases than it is to do the Toffoli directly.

Exercise 4.27: Using just CNOTs and Toffoli gates, construct a quantum circuit to perform the transformation

$$\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix}. \tag{4.31}$$

This kind of partial cyclic permutation operation will be useful later, in Chapter 7.

How may we implement $C^n(U)$ gates using our existing repertoire of gates, where U is an arbitrary single qubit unitary operation? A particularly simple circuit for achieving this task is illustrated in Figure 4.10. The circuit divides up into three stages, and makes use of a small number $(n-1)$ of working qubits, which all start and end in the state $|0\rangle$. Suppose the control qubits are in the computational basis state $|c_1, c_2, \ldots, c_n\rangle$. The first stage of the circuit is to reversibly AND all the control bits $c_1, \ldots, c_n$ together to produce the product $c_1 \cdot c_2 \ldots c_n$. To do this, the first gate in the circuit ANDs c_1 and c_2 together, using a Toffoli gate, changing the state of the first work qubit to $|c_1 \cdot c_2\rangle$. The next Toffoli gate ANDs c_3 with the product $c_1 \cdot c_2$, changing the state of the second work qubit to $|c_1 \cdot c_2 \cdot c_3\rangle$. We continue applying Toffoli gates in this fashion, until the final work qubit is in the state $|c_1 \cdot c_2 \ldots c_n\rangle$. Next, a U operation on the target qubit is

it is left alone. That is, in this basis, the target and control have essentially interchanged roles!

Our immediate goal is to understand how to implement the controlled-U operation for arbitrary single qubit U, using only single qubit operations and the CNOT gate. Our strategy is a two-part procedure based upon the decomposition $U = e^{i\alpha}AXBXC$ given in Corollary 4.2 on page 176.

Our first step will be to apply the phase shift $\exp(i\alpha)$ on the target qubit, controlled by the control qubit. That is, if the control qubit is $|0\rangle$, then the target qubit is left alone, while if the control qubit is $|1\rangle$, a phase shift $\exp(i\alpha)$ is applied to the target. A circuit implementing this operation using just a single qubit unitary gate is depicted on the right hand side of Figure 4.5. To verify that this circuit works correctly, note that the effect of the circuit on the right hand side is

$$|00\rangle \rightarrow |00\rangle, \quad |01\rangle \rightarrow |01\rangle, \quad |10\rangle \rightarrow e^{i\alpha}|10\rangle, \quad |11\rangle \rightarrow e^{i\alpha}|11\rangle, \tag{4.28}$$

which is exactly what is required for the controlled operation on the left hand side.

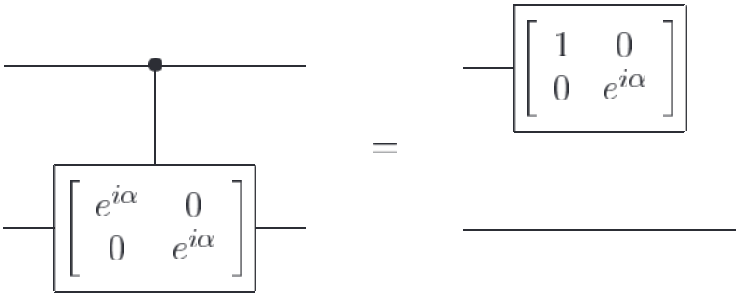


Figure 4.5. Controlled phase shift gate and an equivalent circuit for two qubits.

We may now complete the construction of the controlled-U operation, as shown in Figure 4.6. To understand why this circuit works, recall from Corollary 4.2 that U may be written in the form $U = e^{i\alpha}AXBXC$, where A, B and C are single qubit operations such that $ABC = I$. Suppose that the control qubit is set. Then the operation $e^{i\alpha}AXBXC = U$ is applied to the second qubit. If, on the other hand, the control qubit is not set, then the operation $ABC = I$ is applied to the second qubit; that is, no change is made. That is, this circuit implements the controlled-U operation.

Now that we know how to condition on a single qubit being set, what about conditioning on multiple qubits? We've already met one example of multiple qubit conditioning, the Toffoli gate, which flips the third qubit, the target qubit, conditioned on the first two qubits, the control qubits, being set to one. More generally, suppose we have $n + k$ qubits, and U is a k qubit unitary operator. Then we define the controlled operation $C^n(U)$ by the equation

$$C^n(U)|x_1x_2\ldots x_n\rangle|\psi\rangle = |x_1x_2\ldots x_n\rangle U^{x_1x_2\ldots x_n}|\psi\rangle\,, \tag{4.29}$$

where $x_1x_2\ldots x_n$ in the exponent of U means the *product* of the bits $x_1, x_2, \ldots, x_n$. That is, the operator U is applied to the last k qubits if the first n qubits are all equal to one, and otherwise, nothing is done. Such conditional operations are so useful that we

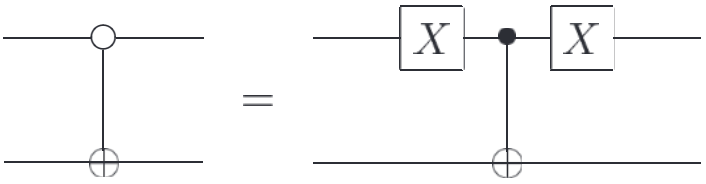


Figure 4.11. Controlled operation with a NOT gate being performed on the second qubit, conditional on the first qubit being set to zero.

to one and circuits which condition on qubits being set to zero, by insertion of X gates in appropriate locations, as illustrated in Figure 4.12.

Another convention which is sometimes useful is to allow controlled-NOT gates to have multiple targets, as shown in Figure 4.13. This natural notation means that when the control qubit is 1, then all the qubits marked with a $\oplus$ are flipped, and otherwise nothing happens. It is convenient to use, for example, in constructing classical functions such as permutations, or in encoders and decoders for quantum error-correction circuits, as we shall see in Chapter 10.

Exercise 4.31: (More circuit identities) Let subscripts denote which qubit an operator acts on, and let C be a CNOT with qubit 1 the control qubit and qubit 2 the target qubit. Prove the following identities:

$$CX_1C = X_1X_2 \tag{4.32}$$
$$CY_1C = Y_1X_2 \tag{4.33}$$
$$CZ_1C = Z_1 \tag{4.34}$$
$$CX_2C = X_2 \tag{4.35}$$
$$CY_2C = Z_1Y_2 \tag{4.36}$$
$$CZ_2C = Z_1Z_2 \tag{4.37}$$
$$R_{z,1}(\theta)C = CR_{z,1}(\theta) \tag{4.38}$$
$$R_{x,2}(\theta)C = CR_{x,2}(\theta). \tag{4.39}$$

4.4 Measurement

A final element used in quantum circuits, almost implicitly sometimes, is measurement. In our circuits, we shall denote a projective measurement in the computational basis (Section 2.2.5) using a 'meter' symbol, illustrated in Figure 4.14. In the theory of quantum circuits it is conventional to not use any special symbols to denote more general measurements, because, as explained in Chapter 2, they can always be represented by unitary transforms with ancilla qubits followed by projective measurements.

There are two important principles that it is worth bearing in mind about quantum circuits. Both principles are rather obvious; however, they are of such great utility that they are worth emphasizing early. The first principle is that classically conditioned operations can be replaced by quantum conditioned operations:

The prototypical controlled operation is the controlled-NOT, which we met in Section 1.2.1. Recall that this gate, which we'll often refer to as CNOT, is a quantum gate with two input qubits, known as the *control qubit* and *target qubit*, respectively. It is drawn as shown in Figure 4.3. In terms of the computational basis, the action of the CNOT is given by $|c\rangle|t\rangle \rightarrow |c\rangle|t \oplus c\rangle$; that is, if the control qubit is set to $|1\rangle$ then the target qubit is flipped, otherwise the target qubit is left alone. Thus, in the computational basis $|\text{control}, \text{target}\rangle$ the matrix representation of CNOT is

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}. \tag{4.23}$$

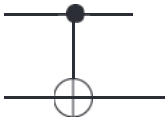


Figure 4.3. Circuit representation for the controlled-NOT gate. The top line represents the control qubit, the bottom line the target qubit.

More generally, suppose U is an arbitrary single qubit unitary operation. A *controlled-U* operation is a two qubit operation, again with a control and a target qubit. If the control qubit is set then U is applied to the target qubit, otherwise the target qubit is left alone; that is, $|c\rangle|t\rangle \rightarrow |c\rangle U^c|t\rangle$. The controlled-$U$ operation is represented by the circuit shown in Figure 4.4.

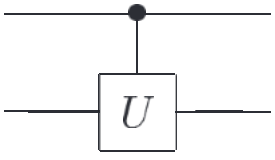
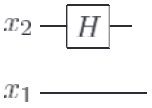


Figure 4.4. Controlled-U operation. The top line is the control qubit, and the bottom line is the target qubit. If the control qubit is set then U is applied to the target, otherwise it is left alone.

Exercise 4.16: (Matrix representation of multi-qubit gates) What is the 4×4 unitary matrix for the circuit



in the computational basis? What is the unitary matrix for the circuit