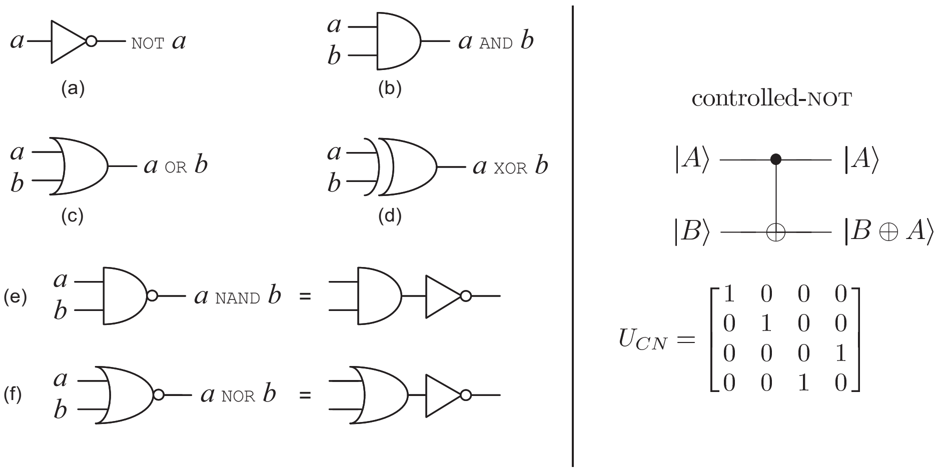

$$U_{CN} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Figure III.9: Left: classical gates. Right: controlled-Not gate. [from Nielsen & Chuang (2010, Fig. 1.6)]

C.2 Quantum gates

Quantum gates are analogous to ordinary logic gates (the fundamental building blocks of circuits), but they must be unitary transformations (see Fig. III.9, left, for ordinarty logic gates). Fortunately, Bennett, Fredkin, and Toffoli have already shown how all the usual logic operations can be done reversibly. In this section you will learn the most important quantum gates.

C.2.a Single-qubit gates

The NOT gate is simple because it is reversible: $\text{NOT}|0\rangle = |1\rangle$, $\text{NOT}|1\rangle = |0\rangle$. Its desired behavior can be represented:

$$\begin{array}{rrcl} \text{NOT}: & |0\rangle & \mapsto & |1\rangle \\ & |1\rangle & \mapsto & |0\rangle. \end{array}$$

Note that defining it on a basis defines it on all quantum states. Therefore it can be written as a sum of dyads (outer products):

$$\text{NOT} = |1\rangle\langle 0| + |0\rangle\langle 1|.$$

You can read this, "return $|1\rangle$ if the input is $|0\rangle$, and return $|0\rangle$ if the input is $|1\rangle$." Recall that in the standard basis $|0\rangle = (1\ 0)^{\mathrm{T}}$ and $|1\rangle = (0\ 1)^{\mathrm{T}}$.

Therefore NOT can be represented in the standard basis by computing the outer products:

$$\text{NOT} = \begin{pmatrix} 0 \\ 1 \end{pmatrix} (1\ 0) + \begin{pmatrix} 1 \\ 0 \end{pmatrix} (0\ 1) = \begin{pmatrix} 0 & 0 \\ 1 & 0 \end{pmatrix} + \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}.$$

The first column represents the result for $|0\rangle$, which is $|1\rangle$, and the second represents the result for $|1\rangle$, which is $|0\rangle$.

Although NOT is defined in terms of the computational basis vectors, it applies to any qubit, in particular to superpositions of $|0\rangle$ and $|1\rangle$:

$$\text{NOT}(a|0\rangle + b|1\rangle) = a\text{NOT}|0\rangle + b\text{NOT}|1\rangle = a|1\rangle + b|0\rangle = b|0\rangle + a|1\rangle.$$

Therefore, NOT exchanges the amplitudes of $|0\rangle$ and $|1\rangle$.

In quantum mechanics, the NOT transformation is usually called X. It is one of four useful unitary operations, called the *Pauli matrices*, which are worth remembering. In the standard basis:

$$I \stackrel{\text{def}}{=} \sigma_0 \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \quad \text{(III.10)}$$

$$X \stackrel{\text{def}}{=} \sigma_x \stackrel{\text{def}}{=} \sigma_1 \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \quad \text{(III.11)}$$

$$Y \stackrel{\text{def}}{=} \sigma_y \stackrel{\text{def}}{=} \sigma_2 \stackrel{\text{def}}{=} \begin{pmatrix} 0 & i \\ -i & 0 \end{pmatrix} \quad \text{(III.12)}$$

$$Z \stackrel{\text{def}}{=} \sigma_z \stackrel{\text{def}}{=} \sigma_3 \stackrel{\text{def}}{=} \begin{pmatrix} 1 & 0 \\ 0 & -1 \end{pmatrix} \quad \text{(III.13)}$$

We have seen that X is NOT, and I is obviously the identity gate. Z leaves $|0\rangle$ unchanged and maps $|1\rangle$ to $-|1\rangle$. It is called the phase-flip operator because it flips the phase of the $|1\rangle$ component by π relative to the $|0\rangle$ component. (Recall that global/absolute phase doesn't matter.) The Pauli matrices span the space of 2×2 complex matrices (Exer. III.15).

Note that $Z|+\rangle = |-\rangle$ and $Z|-\rangle = |+\rangle$. It is thus the analog in the sign basis of X (NOT) in the computational basis. What is the effect of Y on the computational basis vectors? (Exer. III.10)

Note that there is an alternative definition of Y that differs only in global phase:

$$Y \stackrel{\text{def}}{=} \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

This is a $90^\circ = \pi/2$ counterclockwise rotation: $Y(a|0\rangle + b|1\rangle) = b|0\rangle - a|1\rangle$. Draw a diagram to make sure you see this.

Note that the Pauli operations apply to *any* state, not just basis states. The X, Y, and Z operators get their names from the fact that they reflect state vectors along the x, y, z axes of the Bloch-sphere representation of a qubit, which we will not use in this book. Since they are reflections, they are Hermitian (their own inverses).

C.2.b Multiple-qubit gates

We know that any logic circuit can be built up from NAND gates. Can we do the same for quantum logic, that is, is there a universal quantum logic gate? We can't use NAND, because it's not reversible, but we will see that there are universal sets of quantum gates.

The *controlled-NOT* or CNOT gate has two inputs: the first determines what it does to the second (negate it or not).

$$\begin{array}{rlcl}
\text{CNOT}: & |00\rangle & \mapsto & |00\rangle \\
 & |01\rangle & \mapsto & |01\rangle \\
 & |10\rangle & \mapsto & |11\rangle \\
 & |11\rangle & \mapsto & |10\rangle.
\end{array}$$

Its first argument is called the *control* and its second is called the *target*, *controlled*, or *data* qubit. It is a simple example of conditional quantum computation. CNOT can be translated into a sum-of-dyads representation (Sec. A.2.d), which can be written in matrix form (Ex. III.18, p. 181):

$$\begin{array}{rcl}
\text{CNOT} & = & |00\rangle\langle 00| \\
 & + & |01\rangle\langle 01| \\
 & + & |11\rangle\langle 10| \\
 & + & |10\rangle\langle 11|
\end{array}$$

We can also define it (for $x, y \in \mathbf{2}$), $\text{CNOT}|xy\rangle = |xz\rangle$, where $z = x \oplus y$, the exclusive OR of x and y. That is, $\text{CNOT}|x, y\rangle = |x, x \oplus y\rangle$ CNOT is the only non-trivial 2-qubit reversible logic gate. Note that CNOT is unitary since obviously $\text{CNOT} = \text{CNOT}^\dagger$ (which you can show using its dyadic representation or its matrix representation, Ex. III.18, p. 181). See the right

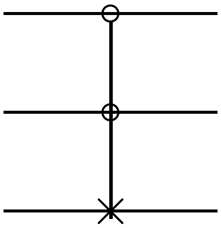

Figure III.10: Diagram for CCNOT or Toffoli gate [fig. from Nielsen & Chuang (2010)]. Sometimes the × is replaced by ⊕ because CCNOT$|xyz\rangle = |x, y, xy \oplus z\rangle$.

panel of Fig. III.9 (p. 104) for the matrix and note the diagram notation for CNOT.

CNOT can be used to produce an entangled state:

$$\text{CNOT}\left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right]|0\rangle = \text{CNOT}\frac{1}{\sqrt{2}}(|00\rangle + |10\rangle) = \frac{1}{\sqrt{2}}(|00\rangle + |11\rangle) = \beta_{00}.$$

Note also that CNOT$|x, 0\rangle = |x, x\rangle$, that is, FAN-OUT, which would seem to violate the No-cloning Theorem, but it works as expected only for $x \in \mathbf{2}$. In general CNOT$|\psi\rangle|0\rangle \neq |\psi\rangle|\psi\rangle$ (Exer. III.19).

Another useful gate is the three-input/output *Toffoli gate* or *controlled-controlled-NOT*. It negates the third qubit if and only if the first two qubits are both 1. For $x, y, z \in \mathbf{2}$,

$$\begin{aligned}\text{CCNOT}|1, 1, z\rangle &\overset{\text{def}}{=} |1, 1, \neg z\rangle,\\ \text{CCNOT}|x, y, z\rangle &\overset{\text{def}}{=} |x, y, z\rangle, \quad \text{otherwise.}\end{aligned}$$

All the Boolean operations can be implemented (reversibly!) by using Toffoli gates (Exer. III.21). For example, CCNOT$|x, y, 0\rangle = |x, y, x \wedge y\rangle$. Thus it is a universal gate for quantum logic.

In Jan. 2009 CCNOT was implemented successfully using trapped ions.[5]

[5] Monz, T.; Kim, K.; Hänsel, W.; Riebe, M.; Villar, A. S.; Schindler, P.; Chwalla, M.; Hennrich, M. et al. (Jan 2009). "Realization of the Quantum Toffoli Gate with Trapped Ions." *Phys. Rev. Lett.* **102** (4): 040501. `arXiv:0804.0082`.

C.2.c Walsh-Hadamard transformation

Recall that the sign basis is defined $|+\rangle \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)$ and $|-\rangle \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}}(|0\rangle - |1\rangle)$. The *Hadamard transformation* or *gate* is defined:

$$H|0\rangle \stackrel{\text{def}}{=} |+\rangle, \tag{III.14}$$

$$H|1\rangle \stackrel{\text{def}}{=} |-\rangle. \tag{III.15}$$

In sum-of-dyads form: $H \stackrel{\text{def}}{=} |+\rangle\langle 0| + |-\rangle\langle 1|$. In matrix form (with respect to the standard basis):

$$H \stackrel{\text{def}}{=} \frac{1}{\sqrt{2}} \begin{pmatrix} 1 & 1 \\ 1 & -1 \end{pmatrix}. \tag{III.16}$$

Note that H is self-adjoint, $H^2 = I$ (since $H^\dagger = H$). H can be defined also in terms of the Pauli matrices: $H = (X + Z)/\sqrt{2}$ (Exer. III.28).

The H transform can be used to rotate the computational basis into the sign basis and back (Exer. III.27):

$$\begin{aligned} H(a|0\rangle + b|1\rangle) &= a|+\rangle + b|-\rangle, \\ H(a|+\rangle + b|-\rangle) &= a|0\rangle + b|1\rangle. \end{aligned}$$

Alice and Bob could use this in QKD.

When applied to a $|0\rangle$, H generates an (equal-amplitude) superposition of the two-bit values, $H|0\rangle = \frac{1}{\sqrt{2}}|0\rangle + \frac{1}{\sqrt{2}}|1\rangle$. This is a useful way of generating a superposition of both possible input bits, and the Walsh transform, a tensor power of H, can be applied to a quantum register to generate a superposition of all possible register values. Consider the $n = 2$ case:

$$\begin{aligned} H^{\otimes 2}|\psi, \phi\rangle &= (H \otimes H)\,(|\psi\rangle \otimes |\phi\rangle) \\ &= (H|\psi\rangle) \otimes (H|\phi\rangle) \end{aligned}$$

In particular,

$$\begin{aligned} H^{\otimes 2}|00\rangle &= (H|0\rangle) \otimes (H|0\rangle) \\ &= |+\rangle^{\otimes 2} \\ &= \left[\frac{1}{\sqrt{2}}(|0\rangle + |1\rangle)\right]^{\otimes 2} \\ &= \left(\frac{1}{\sqrt{2}}\right)^2 (|0\rangle + |1\rangle)(|0\rangle + |1\rangle) \\ &= \frac{1}{\sqrt{2^2}}(|00\rangle + |01\rangle + |10\rangle + |11\rangle). \end{aligned}$$

Notice that this is an equal superposition of all possible values of the 2-qubit register. (I wrote the amplitude in a complicated way, $1/\sqrt{2^2}$, to help you see the general case.) In general,

$$\begin{aligned}
H^{\otimes n}|0\rangle^{\otimes n} &= \frac{1}{\sqrt{2^n}} \overbrace{(|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)}^{n} \\
&= \frac{1}{\sqrt{2^n}} (|0\rangle + |1\rangle)^{\otimes n} \\
&= \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x} \in \mathbf{2}^n} |\mathbf{x}\rangle \\
&= \frac{1}{\sqrt{2^n}} \sum_{\mathbf{x}=\mathbf{0}}^{2^n-1} |\mathbf{x}\rangle.
\end{aligned}$$

Note that "$2^n - 1$" represents a string of n 1-bits, and that $\mathbf{2} = \{0, 1\}$. Hence, $H^{\otimes n}|0\rangle^{\otimes n}$ generates an equal superposition of all the 2^n possible values of the n-qubit register. We often write $W_n = H^{\otimes n}$ for the Walsh transformation.

An linear operation applied to such a superposition state in effect applies the operation simultaneously to all 2^n possible input values. This is *exponential* quantum parallelism and suggests that quantum computation might be able to solve exponential problems much more efficiently than classical computers. To see this, suppose $U|x\rangle = |f(x)\rangle$. Then:

$$U(H^{\otimes n}|0\rangle^{\otimes n}) = U\left[\frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |x\rangle\right] = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} U|x\rangle = \frac{1}{\sqrt{2^n}} \sum_{x=0}^{2^n-1} |f(x)\rangle$$

This is a superposition of the function values $f(x)$ for all of the 2^n possible values of x; it is computed by one pass through the operator U.