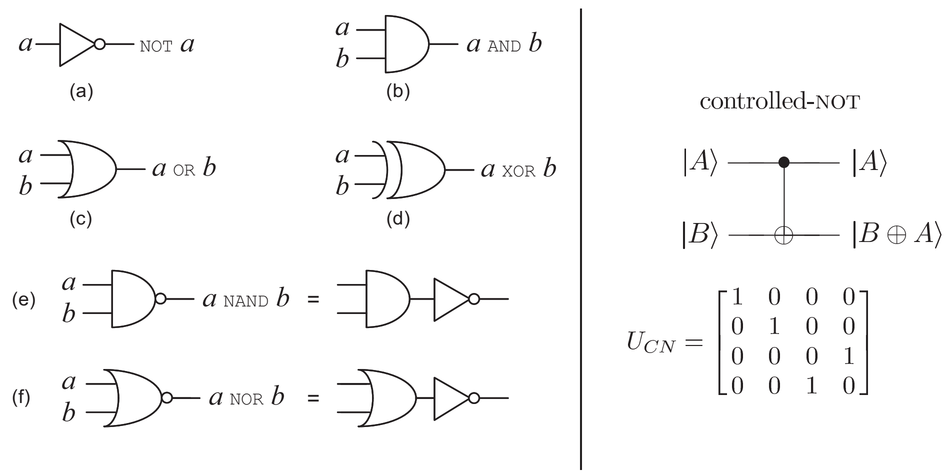

$$U_{CN} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

Figure III.9: Left: classical gates. Right: controlled-NOT gate. [Fig. 1.6 from NC]

C.2 Quantum gates

¶1. Quantum gates are analogous to ordinary logic gates (fundamental building blocks of circuits), but they must be unitary transformations. (See Fig. III.9, left.)
Fortunately, Bennett, Fredkin, and Toffoli have already shown how all the usual logic operations can be done reversibly.

C.2.a Single-qubit gates

¶1. **NOT:** The NOT gate is simple because it is reversible: $\text{NOT}|0\rangle = |1\rangle$, $\text{NOT}|1\rangle = |0\rangle$.

¶2. Its desired behavior can be represented:

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

Note that defining it on a basis defines it on all quantum states.

¶3. Therefore it can be represented

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

You can read this "return $|1\rangle$ if the input is $|0\rangle$, and return $|0\rangle$ if the input is $|1\rangle$."

¶4. In the standard basis:

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

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

¶5. **Superposition:** Although NOT is defined in terms of the computational basis vectors, it applies to any qubit:

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

¶6. **Pauli matrices:** In QM, 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:

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

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

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

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

¶7. We have seen that X is NOT, and I is obviously the identity gate. Z leaves $|0\rangle$ unchanged and maps $|1\rangle$ to $-|1\rangle$.

¶8. **Phase-flip operator:** Z 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.)

¶9. The Pauli matrices span the space of 2×2 complex self-adjoint unitary matrices (exercise).

¶10. 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.

¶11. What is the effect of Y on the computational basis vectors? (Exer. III.9)

¶12. **Alternative definition of Y:** 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 $90° = \pi/2$ counterclockwise rotation:
$Y(a|0\rangle + b|1\rangle) = b|0\rangle - a|1\rangle$.

¶13. Note that these operations apply to *any* state, not just basis states.

¶14. 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 I hope to skip.

¶15. Since they are reflections, they are Hermitian (their own inverses).

C.2.b Multiple-qubit gates

¶1. We know that any logic circuit can be built up from NAND gates. Can we do the same for quantum logic? We can't use NAND, because it's not reversible.

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

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

¶3. **Control and target:** Its first argument is called the *control* and its second is called the *target* or *data* bit.
This is a simple example of conditional quantum computation.

¶4. CNOT can be translated into a sum-of-outer-products or sum-of-dyads representation (Sec. A.2.j), which can be written in matrix form (Ex. III.16, p. 226).

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

¶5. 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 $\text{CNOT}|x, y\rangle = |x, x \oplus y\rangle$

¶6. CNOT is the only non-trivial 2-qubit reversible logic gate.

¶7. Note CNOT is unitary since obviously $\text{CNOT} = \text{CNOT}^\dagger$ (using the outer-product representation or its matrix representation, Ex. III.16, p. 226). See Fig. III.9 (right) for the matrix.

¶8. Note the diagram for CNOT in Fig. III.9 (right).

¶9. CNOT can be used to produce an entangled state:

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

¶10. **FAN-OUT:** Note that $\text{CNOT}|x, 0\rangle = |x, x\rangle$, i.e., FAN-OUT, which would seem to violate the No-cloning Theorem, but it works as expected only for $x \in \mathbf{2}$.
Note that in general $\text{CNOT}|\psi\rangle|0\rangle \neq |\psi\rangle|\psi\rangle$ (Exer. III.17).

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

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

¶12. All the Booleans operations can be implemented (reversibly!) by using Toffoli gates (Exer. III.19).

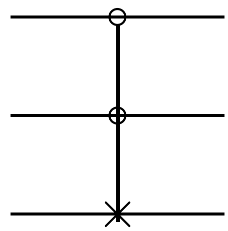

Figure III.10: Diagram for CCNOT or Toffoli gate [fig. from NC]. Sometimes the $\times$ is replaced by $\oplus$ because $\text{CCNOT}|xyz\rangle = |x, y, xy \oplus z\rangle$.

¶13. For example, $\text{CCNOT}|x, y, 0\rangle = |x, y, x \wedge y\rangle$.

¶14. **Quantum implementation:** In Jan. 2009 CCNOT was successfully implemented using trapped ions.[3]

C.2.c Walsh-Hadamard transformation

¶1. **Hadamard transformation:** The *Hadamard transformation* or *gate* is defined:

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

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

¶2. In sum-of-dyads form: $H \stackrel{\text{def}}{=} |+\rangle\langle 0| + |-\rangle\langle 1|$.

¶3. In matrix form (standard basis):

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

¶4. Applied to a $|0\rangle$, H generates an (equally-weighted) 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 superposition of possible inputs (described shortly).

[3] 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`.

¶5. $H^2 = I$ (since $H^\dagger = H$).

¶6. **Rotation of basis:** The H transform can be used to rotate the computational basis into the sign basis and back (Exer. III.24):

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

Alice and Bob could use this in QKD.

¶7. $H = (X + Z)/\sqrt{2}$ (Exer. III.25).

¶8. **Walsh(-Hadamard) transform:** The Walsh transform, a tensor power of H, can be applied to a quantum register to generate a superposition of all possible register values.

¶9. Consider the $n = 2$ case:

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

¶10. 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 a superposition of all possible values of the 2-bit register.

¶11. In general,

$$H^{\otimes n}|0\rangle^{\otimes n} = \frac{1}{\sqrt{2^n}} \overbrace{(|0\rangle + |1\rangle) \otimes (|0\rangle + |1\rangle) \otimes \cdots \otimes (|0\rangle + |1\rangle)}^{n}$$

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

Note that "$2^n - 1$" represents a string of n 1-bits, and $\mathbf{2} = \{0, 1\}$.

¶12. Hence, $H^{\otimes n}|0\rangle^{\otimes n}$ generates a superposition of all 2^n possible values of the n-qubit register.

¶13. **W:** We often write $W_n = H^{\otimes n}$ for the Walsh transformation.

¶14. **quantum parallelism:** An operation applied to such a superposition state in effect applies the operation simultaneously to all 2^n possible values. This is *exponential* quantum parallelism.

¶15. This suggests that QC might be able to solve exponential problems much more efficiently than classical computers.