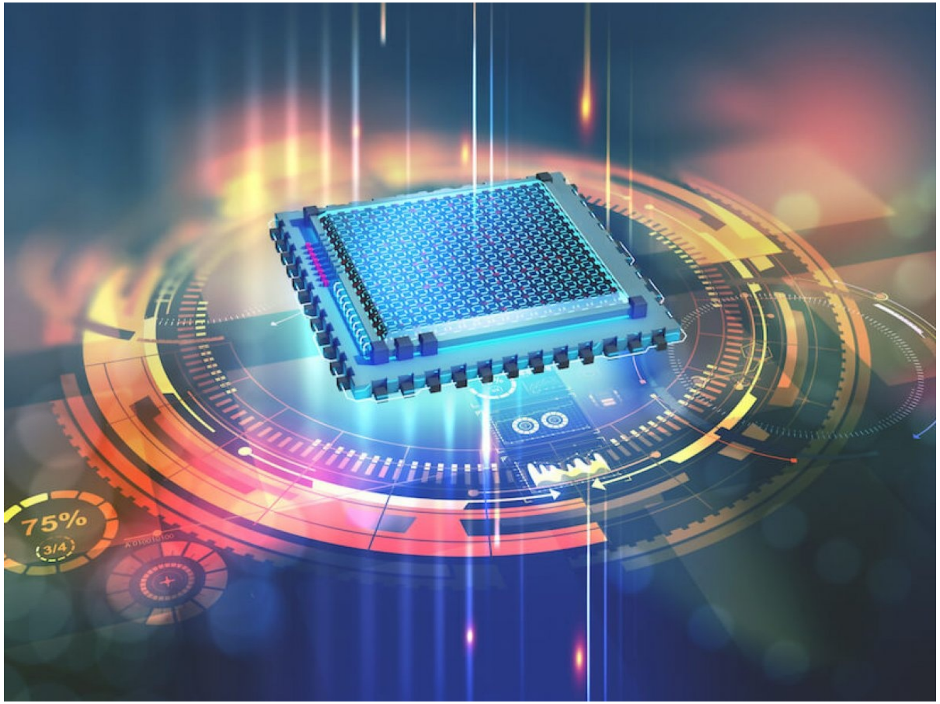
Introduction to Quantum Computing



We start from bits

- (but things will get more complex quickly)
- A bit is anything which can be in one of two states
- We denote these two states as 0 and 1
- How many possibilities if we have 3 bits? And if we have n bits?

Representing bit-strings as vectors

- One bit: 0 or 1
- We will represent them as vectors since this easily generalizes to qubits
- $|0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$ $\qquad |1\rangle = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$
- With 2 bits we have 4 possibilities:

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

Tensor products

- All the possible combinations, with the product of values

$$\begin{bmatrix} x_0 \\ x_1 \end{bmatrix} \otimes \begin{bmatrix} y_0 \\ y_1 \end{bmatrix} = \begin{bmatrix} x_0 y_0 \\ x_0 y_1 \\ x_1 y_0 \\ x_1 y_1 \end{bmatrix} \qquad |0\,0\rangle = |0\rangle \otimes |0\rangle = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \otimes \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

- For now, values can be just 0 and 1
- Observation: exactly one 1 in each vector on the LHS implies exactly one 1 in the result
- The operator can be generalized to n vectors
- Tensor product corresponds to compose states

Manipulating bits

- One bit operation: NOT
- NOT(x) = 1 if x=0
  0 if x=1
- NOT just flips the bit
- Notation for NOT(x): $\bar{x}$

Manipulating bits as vectors

- x can be either $\begin{bmatrix} 0 \\ 1 \end{bmatrix}$ or $\begin{bmatrix} 1 \\ 0 \end{bmatrix}$
- More in general $x = \begin{bmatrix} x_0 \\ x_1 \end{bmatrix}$
- We use matrices to represent operators, and matrix multiplication to apply them
- NOT is denoted as X, and represented by the matrix

$$X = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$$

Matrix multiplication

- We can multiply a matrix *nxm* and a matrix *mxp* and obtain a matrix *nxp*

$$A = \begin{bmatrix} a_{1,1} & a_{1,2} & \cdots & a_{1,m} \\ a_{2,1} & a_{2,2} & \cdots & a_{2,m} \\ \vdots & \vdots & \ddots & \vdots \\ a_{n,1} & a_{n,2} & \cdots & a_{n,m} \end{bmatrix} \qquad B = \begin{bmatrix} b_{1,1} & b_{1,2} & \cdots & b_{1,p} \\ b_{2,1} & b_{2,2} & \cdots & b_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ b_{m,1} & b_{m,2} & \cdots & b_{m,p} \end{bmatrix}$$

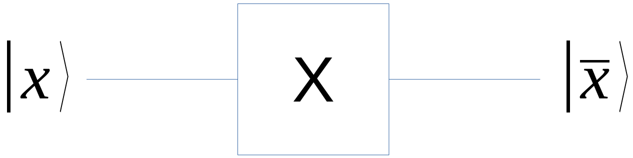
$$C = \begin{bmatrix} c_{1,1} & c_{1,2} & \cdots & c_{1,p} \\ c_{2,1} & c_{2,2} & \cdots & c_{2,p} \\ \vdots & \vdots & \ddots & \vdots \\ c_{n,1} & c_{n,2} & \cdots & c_{n,p} \end{bmatrix}$$

$$c_{i,j} = a_{i,1}b_{1,j} + a_{i,2}b_{2,j} + \cdots + a_{i,m}b_{m,j} = \\ = \sum_{k=1}^{m} a_{i,k}b_{k,j}$$

Element obtained by multiplying row by column

Flipping a bit

- Circuit diagram:



X is the matrix we defined before

- Ket notation:

$$|\bar{x}\rangle = X|x\rangle$$

Manipulating one bit

- If someone hands you a bit, there are two ways you can process it to obtain another bit:

  1) Keep it as it is (identity)

  => those are actually the only operation I can do on a single bit

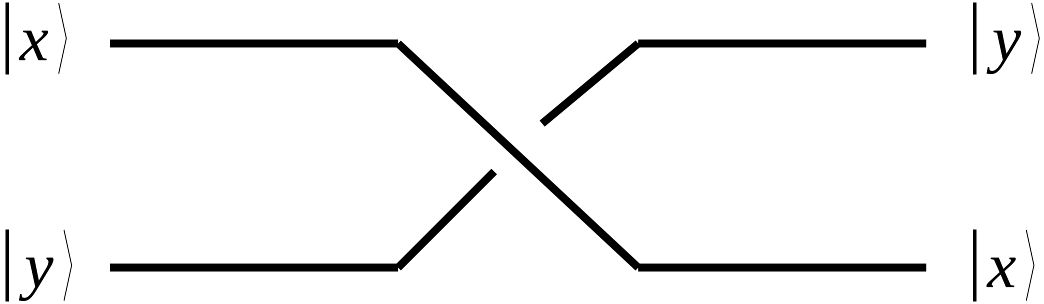
  2) Flip it (NOT)

- That's it. If we want more interesting computations, we'd better have more bits

- We could also turn it into constant 0 or 1, but these are not *reversible*

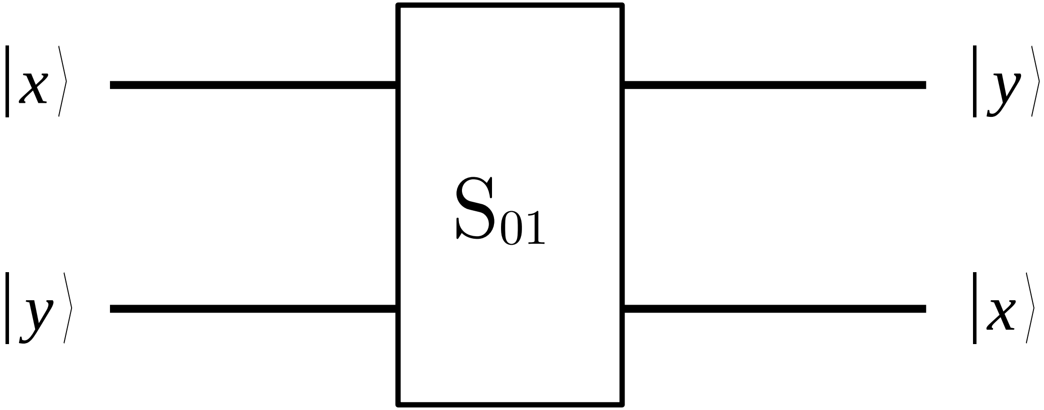
Swapping two bits

- Given two bits, what can we do?
  For instance, swap them!



Swapping two bits

- Given two bits, what can we do?
  For instance, swap them!



Swapping two bits

- Given two bits, what can we do?
  For instance, swap them!

$$S_{01}|x\rangle|y\rangle = |y\rangle|x\rangle$$

We assume tensor product between vectors, and matrix multiplication between matrices and vectors

- As a matrix:

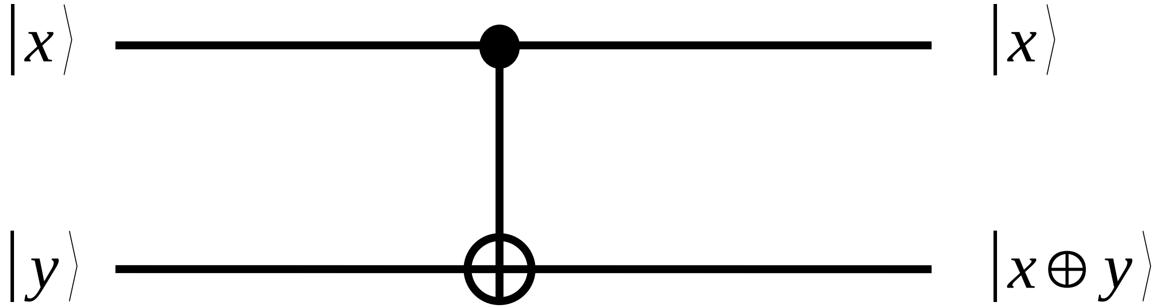
$$S_{01} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Controlled NOT

- NOT on the second bit, if the first is 1, identity otherwise

$$CNOT|x\rangle|y\rangle = |x\rangle|x \oplus y\rangle$$

$$where\ x \oplus y = (x + y)\, mod\, 2$$



Controlled NOT: test

- What matrix represents CNOT?
  $CNOT|x\rangle|y\rangle=|x\rangle|x\oplus y\rangle$

$$(A)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (B)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$(C)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix} \qquad (D)\quad \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Controlled NOT: test solution

- What matrix represents CNOT?
  $CNOT|x\rangle|y\rangle = |x\rangle|x \oplus y\rangle$

$$(A)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (B)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\boxed{(C)\quad \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}} \qquad (D)\quad \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

Controlled NOT: test explanation

$$CNOT\left|x\right\rangle\left|y\right\rangle=\left|x\right\rangle\left|x\oplus y\right\rangle$$

$$\begin{array}{cl}\begin{bmatrix} \; \\ \; \\ \; \\ \; \end{bmatrix} & \begin{matrix}00\\01\\10\\11\end{matrix}\end{array}$$

$$CNOT\ swaps\ 10\ \text{and}\ 11$$

$$00\rightarrow 00\,,01\rightarrow 01\,,10\rightarrow 11\,,11\rightarrow 10$$

- The matrix doing this is $\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix}$