2. Quantum Dense Coding

- **Goal**: To use one qubit to transmit two classical bits.
- **But**: One qubit (supposedly) only contains one classical bit's worth of information!
- **So**: How can we send 2 classical bits using just one qubit?
- **Answer**: Use entangled states!
Set-Up:

- Prepare two qubits Q1, Q2 in an entangled state $|\Psi^+\rangle = \sqrt{\frac{1}{2}} (|0\rangle_1|0\rangle_2 + |1\rangle_1|1\rangle_2)$.
- Alice gets Q1, Bob gets Q2.
- Alice manipulates her Q1 so that it steers Bob's Q2 into a state from which he can read off the 2 classical bits Alice desires to send. All he needs to do this is the post-manipulated Q1 that Alice sends to him.

$$|\Psi^+\rangle = \sqrt{\frac{1}{2}} (|0\rangle_1|0\rangle_2 + |1\rangle_1|1\rangle_2).$$
Protocol

1. Alice has a pair of classical bits: either 00, 01, 10, or 11. She first encodes it in Q1 by acting on Q1 with one of \{I, X, Y, Z\} according to:

| pair: \( (I_1 \otimes I_2) |\Psi^+\rangle \) | transform: \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2) \) | new state: | 
|---|---|---|---|
| 00 | \( (I_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2) = |\Psi^+\rangle \) | \( \bullet \) Let Q1 and Q2 be electrons in Hardness states. 
| 01 | \( (X_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|1\rangle_1 |0\rangle_2 + |0\rangle_1 |1\rangle_2) = |\Phi^+\rangle \) | 
| 10 | \( (Y_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (-|1\rangle_1 |0\rangle_2 + |0\rangle_1 |1\rangle_2) = |\Phi^-\rangle \) | 
| 11 | \( (Z_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 - |1\rangle_1 |1\rangle_2) = |\Psi^-\rangle \) | 

2. Alice now sends Q1 to Bob.

3. After reception of Q1, Bob first applies a \( C_{NOT} \) transformation to both Q1 and Q2:

| pair: \( (I_1 \otimes I_2) |\Psi^+\rangle \) | transform: \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2) \) | new state: | Apply \( C_{NOT} \): \( \frac{1}{\sqrt{2}} (|0\rangle_1 + |1\rangle_1) |0\rangle_2 \) |
|---|---|---|---|
| 00 | \( (I_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2) = |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 + |1\rangle_1) |0\rangle_2 \) |
| 01 | \( (X_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|1\rangle_1 |0\rangle_2 + |0\rangle_1 |1\rangle_2) = |\Phi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|1\rangle_1 + |0\rangle_1) |1\rangle_2 \) |
| 10 | \( (Y_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (-|1\rangle_1 |0\rangle_2 + |0\rangle_1 |1\rangle_2) = |\Phi^-\rangle \) | \( \frac{1}{\sqrt{2}} (-|1\rangle_1 + |0\rangle_1) |1\rangle_2 \) |
| 11 | \( (Z_1 \otimes I_2) |\Psi^+\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 - |1\rangle_1 |1\rangle_2) = |\Psi^-\rangle \) | \( \frac{1}{\sqrt{2}} (|0\rangle_1 - |1\rangle_1) |0\rangle_2 \) |

\( \bullet \) Note: According to the Eigenvalue-Eigenvector Rule, Q1 still has no definite value, but Q2 now does!
**Protocol**

4. Bob now applies a Hadamard transformation to Q1:

<table>
<thead>
<tr>
<th>pair:</th>
<th>transform:</th>
<th>new state:</th>
<th>Apply C(_{NOT}):</th>
<th>Now Apply H(_1):</th>
</tr>
</thead>
<tbody>
<tr>
<td>00</td>
<td>((I_1 \otimes I_2)</td>
<td>\Psi^+\rangle)</td>
<td>(\sqrt{\frac{1}{2}} (</td>
<td>0\rangle_1</td>
</tr>
<tr>
<td>01</td>
<td>((X_1 \otimes I_2)</td>
<td>\Psi^+\rangle)</td>
<td>(\sqrt{\frac{1}{2}} (</td>
<td>1\rangle_1</td>
</tr>
<tr>
<td>10</td>
<td>((Y_1 \otimes I_2)</td>
<td>\Psi^+\rangle)</td>
<td>(\sqrt{\frac{1}{2}} (-</td>
<td>1\rangle_1</td>
</tr>
<tr>
<td>11</td>
<td>((Z_1 \otimes I_2)</td>
<td>\Psi^+\rangle)</td>
<td>(\sqrt{\frac{1}{2}} (</td>
<td>0\rangle_1</td>
</tr>
</tbody>
</table>

- **Note:** According to the Eigenvalue-Eigenvector Rule, Q1 and Q2 now *both* have definite values.

5. Bob now measures Q1 and Q2 to determine the number Alice sent!

   (a) \((Q1 = 0, Q2 = 0) \Rightarrow 00\) \hspace{1cm} (c) \((Q1 = 1, Q2 = 0) \Rightarrow 10\)

   (b) \((Q1 = 0, Q2 = 1) \Rightarrow 01\) \hspace{1cm} (d) \((Q1 = 1, Q2 = 1) \Rightarrow 11\)
**Question:** How are the 2 classical bits transferred from Alice to Bob?

- *Not* transferred *via* the single qubit.
- Transferred by the *correlations* present in the 2-qubit entangled state $|\Psi^+\rangle$.
- In order to convey information between Alice and Bob, it need *not* be physically transported from Alice to Bob across the intervening spatial distance.
- The *only* thing required to convey information is to set up a correlation between the sender's data and the receiver's data.

$$|\Psi^+\rangle = \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2).$$
3. Quantum Teleportation

- **Goal:** To transmit an unknown quantum state using classical bits and to reconstruct the exact quantum state at the receiver.

- **But:** How can this avoid the No-Cloning Theorem?

- **Answer:** Use entangled states!
**Set-Up:**

- Alice has an unknown $Q_0$, $|Q_0\rangle = a|0\rangle + b|1\rangle$, and wants to send it to Bob.
- $Q_1$ and $Q_2$ are prepared in an entangled state $|\Psi^+\rangle = \sqrt{\frac{1}{2}} (|0\rangle_1|0\rangle_2 + |1\rangle_1|1\rangle_2)$. Alice gets $Q_1$, Bob gets $Q_2$.
- Alice manipulates $Q_0$ and $Q_1$ so that they steer Bob's $Q_2$ into the unknown state of $Q_0$. Bob then reconstructs it using the 2 classical bits sent by Alice.
**Protocol**

1. Alice starts with a 3-qubit system (Q0, Q1, Q2) in the state:

\[ |Q_0\rangle |\Psi^+\rangle = \sqrt{\frac{1}{2}} \left( a|0\rangle_0|0\rangle_1|0\rangle_2 + a|0\rangle_0|1\rangle_1|1\rangle_2 + b|1\rangle_0|0\rangle_1|0\rangle_2 + b|1\rangle_0|1\rangle_1|1\rangle_2 \right) \]

Alice now applies $C_{\text{NOT}}$ on Q0 & Q1, and then a Hadamard transformation on Q0:

*First $C_{\text{NOT}}$ on Q0 & Q1:*

\[ (C_{\text{NOT}} \otimes I_2)|Q_0\rangle |\Psi^+\rangle = \sqrt{\frac{1}{2}} \left( a|0\rangle_0|0\rangle_1|0\rangle_2 + a|0\rangle_0|1\rangle_1|1\rangle_2 + b|1\rangle_0|1\rangle_1|0\rangle_2 + b|1\rangle_0|0\rangle_1|1\rangle_2 \right) \]

*Then H on Q0:*

\[ (H_0 \otimes I_1 \otimes I_2)("\) = \frac{1}{2} |0\rangle_0|0\rangle_1 \left( a|0\rangle_2 + b|1\rangle_2 \right) + \frac{1}{2} |0\rangle_0|1\rangle_1 \left( a|1\rangle_2 + b|0\rangle_2 \right) + \frac{1}{2} |1\rangle_0|0\rangle_1 \left( a|0\rangle_2 - b|1\rangle_2 \right) + \frac{1}{2} |1\rangle_0|1\rangle_1 \left( a|1\rangle_2 - b|0\rangle_2 \right) \]

2. Alice now measures Q0 and Q1:

<table>
<thead>
<tr>
<th>Measurement outcome is:</th>
<th>...Q2 is now in state:</th>
</tr>
</thead>
<tbody>
<tr>
<td>$</td>
<td>0\rangle_0</td>
</tr>
<tr>
<td>$</td>
<td>0\rangle_0</td>
</tr>
<tr>
<td>$</td>
<td>1\rangle_0</td>
</tr>
<tr>
<td>$</td>
<td>1\rangle_0</td>
</tr>
</tbody>
</table>

**EE Rule:** Each of the terms represents a state in which Q0 and Q1 have definite values, but Q2 does not.
3. Alice sends the result of her measurement to Bob in the form of 2 classical bits: 00, 01, 10, or 11.

4. Depending on what he receives, Bob performs one of \{I, X, Y, Z\} on Q2. This allows him to turn it into (reconstruct) the unknown Q0.
• **Question 1:** Does Bob violate the *No-Cloning Theorem*? Doesn't he construct a copy of the unknown Q0?

• *No violation occurs.*

• Bob *does* construct a copy: Q2 has become an exact duplicate of Q0.

• **But:** After Alice is through transforming Q0 and Q1, the original Q0 has now collapsed to either $|0\rangle_0$ or $|1\rangle_0$! Alice destroys Q0 in the process of conveying the information contained in it to Bob!

\[
|\Psi^+\rangle = \frac{1}{\sqrt{2}} (|0\rangle_1 |0\rangle_2 + |1\rangle_1 |1\rangle_2).
\]
• **Question 2**: How does Bob reconstruct the unknown Q0 (that encodes an arbitrarily large amount of information) from just 2 classical bits?

• Information to reconstruct Q0 is transferred by the correlations present in the entangled state $|\Psi^+\rangle$, *in addition* to the 2 classical bits.

• The 2 classical bits are used simply to determine the appropriate transformation on Q2, *after* it has been "steered" into the appropriate state by Alice.

\[
|\Psi^+\rangle = \sqrt{\frac{1}{2}} (|0\rangle_1|0\rangle_2 + |1\rangle_1|1\rangle_2).
\]
4. Quantum Computation.

• *General Goal:* To use the inaccessible arbitrarily large amount of information encoded in qubits to perform computations in "quantum parallel" (*i.e.*, in record time!).

• *Initial (modest) Goal:* To compute all possible values of a function $f$ in a single computation.

• *First Question:* Can classical computations be done using qubits instead of classical bits? Can transformations on qubits be defined that reproduce the transformations on bits that are needed to implement a classical computer.
• **Classical Computation Using Bits:**

To implement a classical computer, it suffices to have an \textit{AND} transformation and a \textit{NOT} transformation on classical bits defined by the following:

\[
\begin{align*}
0 \text{ AND } 0 &= 0 & NOT \ 0 &= 1 \\
0 \text{ AND } 1 &= 0 & NOT \ 1 &= 0 \\
1 \text{ AND } 0 &= 0 \\
1 \text{ AND } 1 &= 1
\end{align*}
\]

• \textit{AND} takes two input bits and produces one output bit.

• \textit{NOT} takes one input bit and produces one output bit.

• **Initial problem:** Transformations on qubits are \textit{reversible}: the number of input qubits \textit{always} must equal the number of output qubits.

\textbf{Why?} Qubit transformations are operators on vector spaces. And an operator defined on an \textit{n}-dim vector space (e.g., \textit{n}-qubit space) that acts on \textit{n}-dim vectors (e.g., \textit{n} qubits) can only spit out \textit{n}-dim vectors.
• Solution: The "Controlled-controlled-\textit{NOT}" $CC_{\text{NOT}}$ operator.

• Changes the third target qubit if the first two control qubits are $|1\rangle|1\rangle$, and leaves it unchanged otherwise.

$$
CC_{\text{NOT}}|0\rangle|0\rangle|0\rangle = |0\rangle|0\rangle|0\rangle \quad CC_{\text{NOT}}|0\rangle|1\rangle|1\rangle = |0\rangle|1\rangle|1\rangle \quad CC_{\text{NOT}}|1\rangle|1\rangle|0\rangle = |1\rangle|1\rangle|1\rangle \\
CC_{\text{NOT}}|0\rangle|0\rangle|1\rangle = |0\rangle|0\rangle|1\rangle \quad CC_{\text{NOT}}|1\rangle|0\rangle|0\rangle = |1\rangle|0\rangle|0\rangle \quad CC_{\text{NOT}}|1\rangle|1\rangle|1\rangle = |1\rangle|1\rangle|0\rangle \\
CC_{\text{NOT}}|0\rangle|1\rangle|0\rangle = |0\rangle|1\rangle|0\rangle \quad CC_{\text{NOT}}|1\rangle|0\rangle|1\rangle = |1\rangle|0\rangle|1\rangle
$$

$$
CC_{\text{NOT}} = \begin{pmatrix}
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1
\end{pmatrix} \quad |0\rangle|0\rangle|0\rangle = \begin{pmatrix}1 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \end{pmatrix} \quad |0\rangle|0\rangle|1\rangle = \begin{pmatrix}0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \end{pmatrix} \quad \ldots \quad |1\rangle|1\rangle|1\rangle = \begin{pmatrix}0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
0 \\
1 \end{pmatrix}
$$

• Claim: $CC_{\text{NOT}}$ implements \textit{AND} and \textit{NOT} on qubits.

- To implement \textit{NOT}, act with $CC_{\text{NOT}}$ on a 3-qubit state in which the first two qubits are $|1\rangle|1\rangle$: $CC_{\text{NOT}}|1\rangle|1\rangle|x\rangle = |1\rangle|1\rangle|NOT \ x\rangle$.

- To implement \textit{AND}, act with $CC_{\text{NOT}}$ on a 3-qubit state in which the last qubit is $|0\rangle$: $CC_{\text{NOT}}|x\rangle|y\rangle|0\rangle = |x\rangle|y\rangle|x \ AND \ y\rangle$. 
• **So:** Any classical computation can be done using qubits instead of bits.

• **In particular:** Any classical function that takes \( n \) input bits and produces \( k \) output bits can be implemented using arrays of primitive \( CC_{NOT} \) "gates".

**How to Construct a Qubit-Based Function Calculator**

• Let \( |x\rangle_{(n)} \) represent \( n \) input qubits that encode the number \( x \).

• Let \( |0\rangle_{(k)} \) represent \( k \) qubits \( |0\rangle \) (the output register).

• Let \( |f(x)\rangle_{(k)} \) represent \( k \) output qubits that encode the number \( f(x) \).

• Define an operator \( U_f \) that acts on \((n+k)\) qubits in the following way:

\[
U_f|x\rangle_{(n)}|0\rangle_{(k)} = |x\rangle_{(n)}|f(x)\rangle_{(k)}.
\]

• **Now:** Feed \( U_f \) a superposition of all possible numbers \( x \) it can take as input.

• **Result:** A superposition of all possible values of the function in a single computation!
**Two Steps:**

1. Prepare as input a superposition of all possible numbers $x$ that can be encoded in $n$ bits:

   (i) Start with an $n$-qubit state $|0_1\rangle|0_2\rangle\cdots|0_n\rangle$

   (ii) Now apply a Hadamard transformation to each qubit:

   \[
   \left( H_1 \otimes H_2 \otimes \cdots \otimes H_n \right) |0_1\rangle|0_2\rangle\cdots|0_n\rangle
   \]

   \[
   = \left( \sqrt{\frac{1}{2}} \right)^n \left\{ (|0_1\rangle + |1_1\rangle)(|0_2\rangle + |1_2\rangle)\cdots(|0_n\rangle + |1_n\rangle) \right\}
   \]

   \[
   = \left( \sqrt{\frac{1}{2}} \right)^n \left\{ |0_1\rangle|0_2\rangle\cdots|0_n\rangle + |0_1\rangle|0_2\rangle\cdots|1_n\rangle + \cdots |1_1\rangle|1_2\rangle\cdots|1_n\rangle \right\} = \left( \sqrt{\frac{1}{2}} \right)^n \sum_{x=0}^{2^n-1} |x\rangle_{(n)}
   \]

   **The first term encodes the binary number for 0.**

   **Each term in between is the binary number for each number between 0 and $2^n - 1$.**

   **The last term encodes the binary number for $2^n - 1$.**

   So the entire sum is a superposition that encodes all numbers $x$ such that $0 \leq x < 2^n$. 
Two Steps:
2. Now attach a $k$-qubit output register $|0\rangle_{(k)}$ and apply $U_f$.

\[
U_f \left( \sqrt{\frac{1}{2}} \right)^n \sum |x\rangle_{(n)} |0\rangle_{(k)} = \left( \sqrt{\frac{1}{2}} \right)^n \sum U_f |x\rangle_{(n)} |0\rangle_{(k)} = \left( \sqrt{\frac{1}{2}} \right)^n \sum |x\rangle_{(n)} |f(x)\rangle_{(k)}
\]

A superposition of all possible values $f(x)$, for $0 \leq x < 2^n$, of the function $f$. And we've effectively calculated them all with just a single application of $U_f$.

- The Catch: None of these values of $f$ is accessible until we make a measurement!

The Task for Quantum Algorithm construction:
- Given a problem, first construct an appropriate superposition of solutions. Then manipulate the superposition so that the relevant terms acquire high probability.
**Example: Shor's Factorization Algorithm (1994)**

- Factors large integers into primes in *polynomial* time.

  - *Polynomial time* $\Rightarrow$ the time needed to factor an integer increases exponentially as the number of digits increases.
  - *Exponential time* $\Rightarrow$ the time needed to factor an integer increases as a power of the increase in number of digits of the integer.

- To factor integer $N$, current classical algorithms require $10^4(\log N)^{1/3}$ steps.
- The largest numbers capable of such factorization have ~150 (base 10) digits.

**Why is fast prime factorization important?**

- Classical RSA Encryption:
  - *public encryption key* = product $pq$ of two (very large) primes.
  - *private decryption key* = $p$, $q$ separately
  - *Thus*: Factorizing $pq$ (in your lifetime) would let you break RSA encryption (standard encryption for web transactions).