An Introduction to Quantum Computing

Lecture 13: Quantum Error Correction

Paolo Zuliani

Dipartimento di Informatica
Università di Roma "La Sapienza", Rome, Italy

Agenda

- Error Correction
- The Repetition Code
- Quantum Error Correction

Error Correction

Errors in computing and communication cannot be avoided entirely:

- John von Neumann's work on the "synthesis of reliable organisms from unreliable components" (1952-56)
- Claude Shannon's work on the transmission of information from random sources (concept of *entropy* and *bit*; 1948)

Error Correction

Errors in computing and communication cannot be avoided entirely:

- John von Neumann's work on the "synthesis of reliable organisms from unreliable components" (1952-56)
- Claude Shannon's work on the transmission of information from random sources (concept of *entropy* and *bit*; 1948)

Computing and communication would be pretty much **impossible** without <u>error detection</u> and <u>error correction</u>:

Error Correction

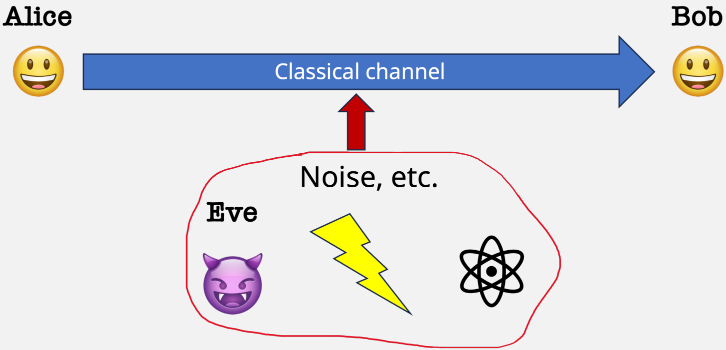
Errors in computing and communication cannot be avoided entirely:

- John von Neumann's work on the "synthesis of reliable organisms from unreliable components" (1952-56)
- Claude Shannon's work on the transmission of information from random sources (concept of *entropy* and *bit*; 1948)

Computing and communication would be pretty much **impossible** without <u>error detection</u> and <u>error correction</u>:

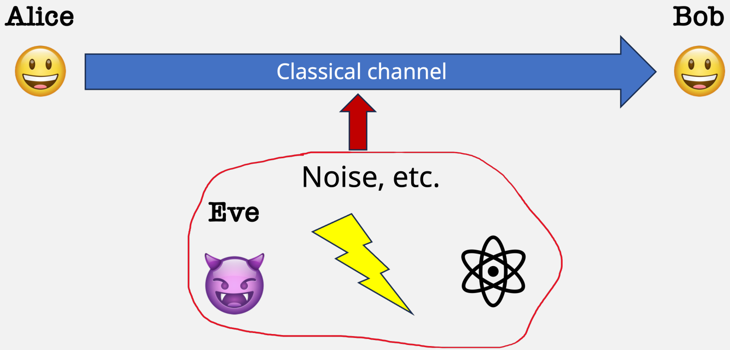
- error-correcting RAM (from servers up)
- data transmission over mobile phone networks
- compact discs (relics from the not-so-distant past!)
- TCP/IP (but UDP is *not* error-corrected)
- basically all transfers of digital information is error-corrected

Error Correction: Context and Problem



Alice wants to send classical bits to Bob, but the communication link is disturbed. Bob may end up with errors (**bit-flips**) in the data received.

Error Correction: Context and Problem



Alice wants to send classical bits to Bob, but the communication link is disturbed. Bob may end up with errors (**bit-flips**) in the data received.

Can we correct that? Assume single-bit errors occur independently.

Error Correction: The Repetition Code

Two-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	00
1	11

Error Correction: The Repetition Code

Two-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	00
1	11

If Bob receives sequences of 00 or 11, then he knows all is fine.

Error Correction: The Repetition Code

Two-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	00
1	11

If Bob receives sequences of 00 or 11, then he knows all is fine.

If Bob receives 01 or 10, then he knows an <u>error has occured</u>!

Error Correction: The Repetition Code

Two-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	00
1	11

If Bob receives sequences of 00 or 11, then he knows all is fine.

If Bob receives 01 or 10, then he knows an <u>error has occured</u>!

Can he correct it?

Error Correction: The Repetition Code

Two-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	00
1	11

If Bob receives sequences of 00 or 11, then he knows all is fine.

If Bob receives 01 or 10, then he knows an error has occured!

Can he correct it?

No! (Try it yourself.)

Error Correction: Repetition Code

Let's try with more bits:

Three-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	000
1	111

Error Correction: Repetition Code

Let's try with more bits:

Three-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	000
1	111

Bob receives	Bob decodes as
000, 001, 010, 100	0
111, 110, 101, 011	1

Error Correction: Repetition Code

Let's try with more bits:

Three-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	000
1	111

Bob receives	Bob decodes as
000, 001, 010, 100	0
111, 110, 101, 011	1

Essentially, Bob checks the parity of the triples received from Alice.

This simple code can detect and correct single-bit errors.

Error Correction: Repetition Code

Let's try with more bits:

Three-bit Repetion Code

Bit that Alice wants to send	Alice actually sends
0	000
1	111

Bob receives	Bob decodes as
000, 001, 010, 100	0
111, 110, 101, 011	1

Essentially, Bob checks the parity of the triples received from Alice.

This simple code can detect and correct single-bit errors.

Can Bob check parity in a way suitable for quantum?

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ $=$ "one of the two, but not both"

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this “offending” bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ $=$ “one of the two, but not both”

Bob receives three bits $b_0 b_1 b_2$.

He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases} \quad =$ "one of the two, but not both"

Bob receives three bits $b_0 b_1 b_2$.

He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

Three-bit Repetion Code

$b_0 b_1 b_2$	$b_0 \oplus b_1$	$b_0 \oplus b_2$	**Bob's corrective action**

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ = "one of the two, but not both"

Bob receives three bits $b_0 b_1 b_2$.
He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

Three-bit Repetion Code

$b_0 b_1 b_2$	$b_0 \oplus b_1$	$b_0 \oplus b_2$	**Bob's corrective action**
$b_0 = b_1 = b_2$	0	0	nothing

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ = "one of the two, but not both"

Bob receives three bits $b_0 b_1 b_2$.

He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

Three-bit Repetion Code

$b_0 b_1 b_2$	$b_0 \oplus b_1$	$b_0 \oplus b_2$	**Bob's corrective action**
$b_0 = b_1 = b_2$	0	0	nothing
$b_0 = b_1 \neq b_2$	0	1	flips b_2

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ = "one of the two, but not both"

Bob receives three bits $b_0 b_1 b_2$.

He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

Three-bit Repetion Code

$b_0 b_1 b_2$	$b_0 \oplus b_1$	$b_0 \oplus b_2$	**Bob's corrective action**
$b_0 = b_1 = b_2$	0	0	nothing
$b_0 = b_1 \neq b_2$	0	1	flips b_2
$b_0 = b_2 \neq b_1$	1	0	flips b_1

Error Correction: Repetion Code (Towards Quantum)

On three bits, if there is a single-bit error then two bits are equal and the third will differ – this "offending" bit must then be flipped to correct the error.

Recall that $a \oplus b = \begin{cases} 0 & a = b \\ 1 & a \neq b \end{cases}$ = "one of the two, but not both"

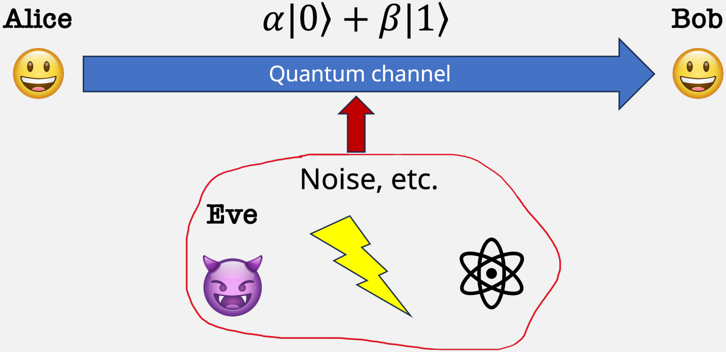
Bob receives three bits $b_0 b_1 b_2$.

He computes $b_0 \oplus b_1$ and $b_0 \oplus b_2$ to correct *any* single-bit error.

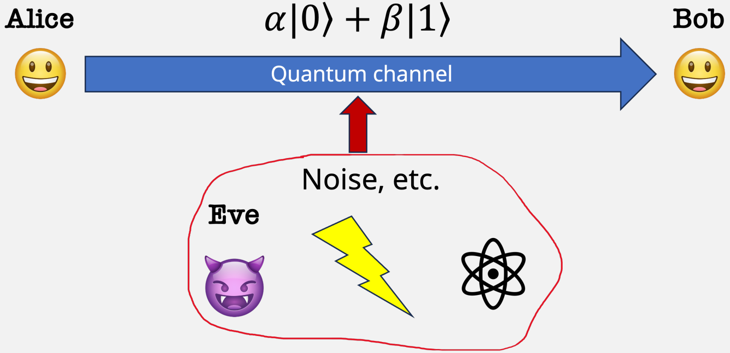
Three-bit Repetion Code

$b_0 b_1 b_2$	$b_0 \oplus b_1$	$b_0 \oplus b_2$	**Bob's corrective action**
$b_0 = b_1 = b_2$	0	0	nothing
$b_0 = b_1 \neq b_2$	0	1	flips b_2
$b_0 = b_2 \neq b_1$	1	0	flips b_1
$b_0 \neq b_1 = b_2$	1	1	flips b_0

Quantum Error Correction

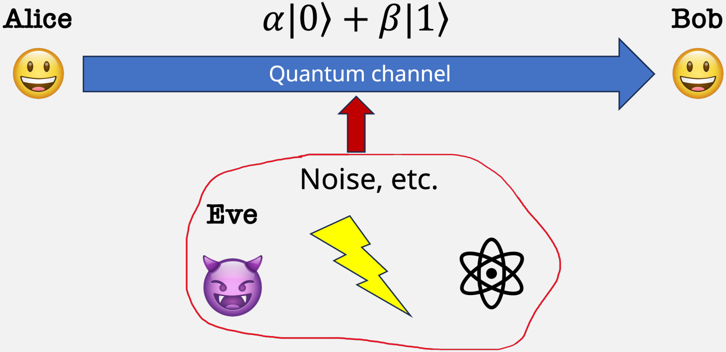


Quantum Error Correction



Assume single-bit errors, *i.e.*, Bob may receive $\alpha\,|1\rangle + \beta\,|0\rangle$ instead of $\alpha\,|0\rangle + \beta\,|1\rangle$.

Quantum Error Correction



Assume single-bit errors, *i.e.*, Bob may receive $\alpha|1\rangle + \beta|0\rangle$ instead of $\alpha|0\rangle + \beta|1\rangle$.

We again use a repetition code (beware of the no-cloning theorem!)

Quantum Error Correction

Alice wants to send $\alpha_0 |0\rangle + \alpha_1 |1\rangle$. We add two ancillary qubits initialized to $|0\rangle$, thus

$$(\alpha_0 |0\rangle + \alpha_1 |1\rangle) \otimes |00\rangle = \alpha_0 |000\rangle + \alpha_1 |100\rangle .$$

Quantum Error Correction

Alice wants to send $\alpha_0 |0\rangle + \alpha_1 |1\rangle$. We add two ancillary qubits initialized to $|0\rangle$, thus

$$(\alpha_0 |0\rangle + \alpha_1 |1\rangle) \otimes |00\rangle = \alpha_0 |000\rangle + \alpha_1 |100\rangle .$$

However, we would like to get $\alpha_0 |000\rangle + \alpha_1 |111\rangle$. How?

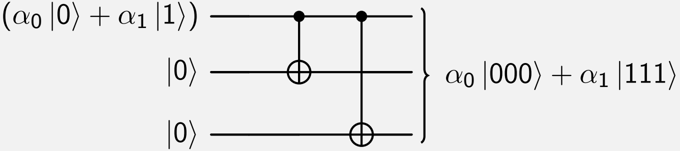
Quantum Error Correction

Alice wants to send $\alpha_0 |0\rangle + \alpha_1 |1\rangle$. We add two ancillary qubits initialized to $|0\rangle$, thus

$$(\alpha_0 |0\rangle + \alpha_1 |1\rangle) \otimes |00\rangle = \alpha_0 |000\rangle + \alpha_1 |100\rangle .$$

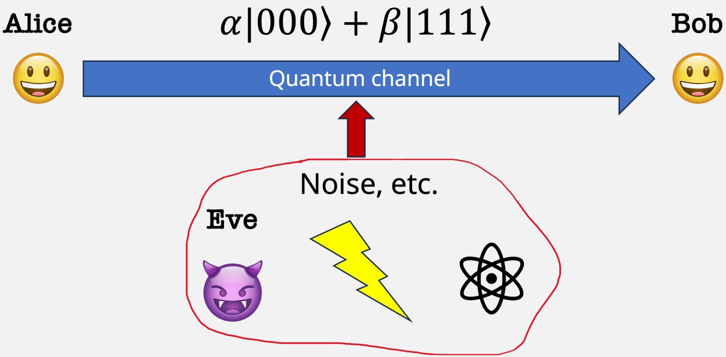
However, we would like to get $\alpha_0 |000\rangle + \alpha_1 |111\rangle$. How?

Apply two CNOTs to the ancillary qubits!

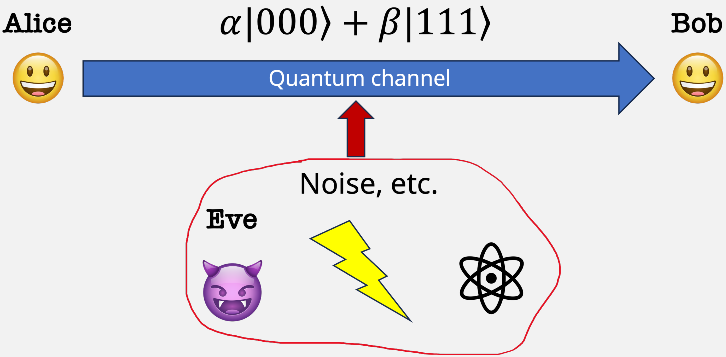


(This is easy to show.)

Quantum Error Correction



Quantum Error Correction

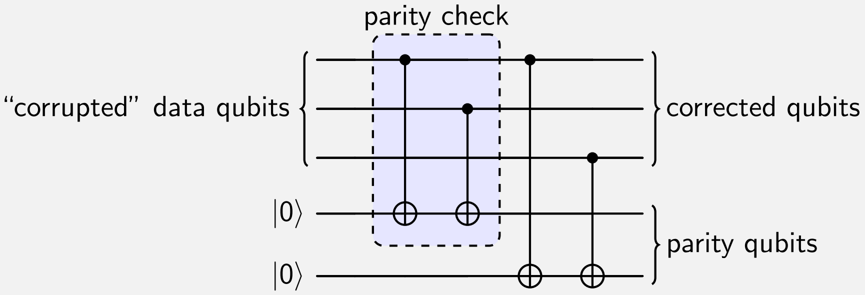


	Bob receives
No error	$\alpha_0 \|000\rangle + \alpha_1 \|111\rangle$
Single bit-flip	$\alpha_0 \|100\rangle + \alpha_1 \|011\rangle$ *or* $\alpha_0 \|010\rangle + \alpha_1 \|101\rangle$ *or* $\alpha_0 \|001\rangle + \alpha_1 \|110\rangle$

How can Bob correct a single bit-flip?

Quantum Error Correction

To correct single bit-flips, Bob uses this circuit:

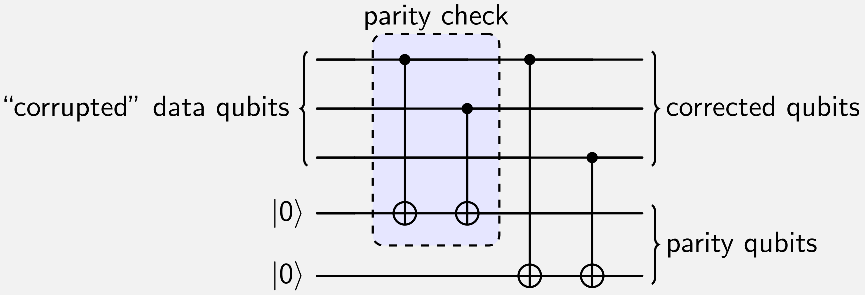


Let's consider the top four qubits (similar reasoning applies to the fifth qubit).

$$(\alpha_0 \left| b_0 b_1 b_2 \right\rangle + \alpha_1 \left| \bar{b}_0 \bar{b}_1 \bar{b}_2 \right\rangle) |0\rangle$$

Quantum Error Correction

To correct single bit-flips, Bob uses this circuit:

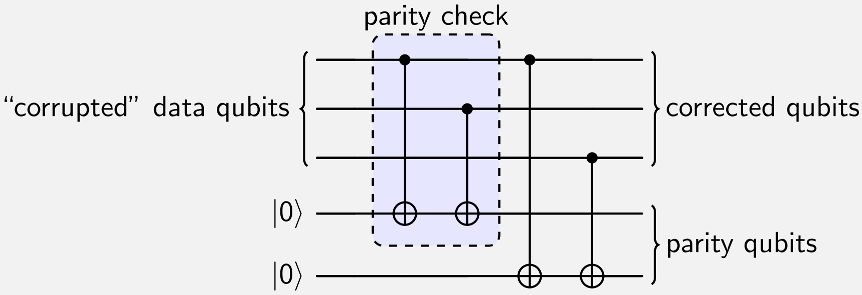


Let’s consider the top four qubits (similar reasoning applies to the fifth qubit).

$$(\alpha_0 |b_0 b_1 b_2\rangle + \alpha_1 |\bar{b_0} \bar{b_1} \bar{b_2}\rangle) |0\rangle = \begin{cases} \alpha_0 |b_0 b_1 b_2 0\rangle + \alpha_1 |\bar{b_0} \bar{b_1} \bar{b_2} 0\rangle & \text{if } b_0 \oplus b_1 = 0 \\ & \end{cases}$$

Quantum Error Correction

To correct single bit-flips, Bob uses this circuit:

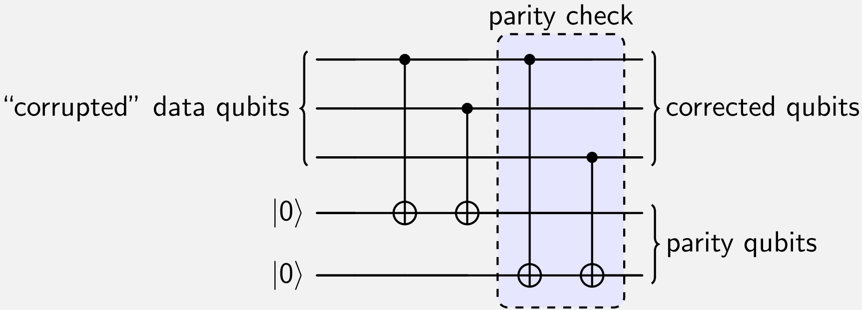


Let's consider the top four qubits (similar reasoning applies to the fifth qubit).

$$(\alpha_0 |b_0 b_1 b_2\rangle + \alpha_1 |\bar{b_0}\bar{b_1}\bar{b_2}\rangle)|0\rangle = \begin{cases} \alpha_0 |b_0 b_1 b_2 0\rangle + \alpha_1 |\bar{b_0}\bar{b_1}\bar{b_2}0\rangle & \text{if } b_0 \oplus b_1 = 0 \\ \alpha_0 |b_0 b_1 b_2 1\rangle + \alpha_1 |\bar{b_0}\bar{b_1}\bar{b_2}1\rangle & \text{if } b_0 \oplus b_1 = 1 \end{cases}$$

Thus, the *fourth* qubit is **not** entangled with the three "corrupted" data qubits.

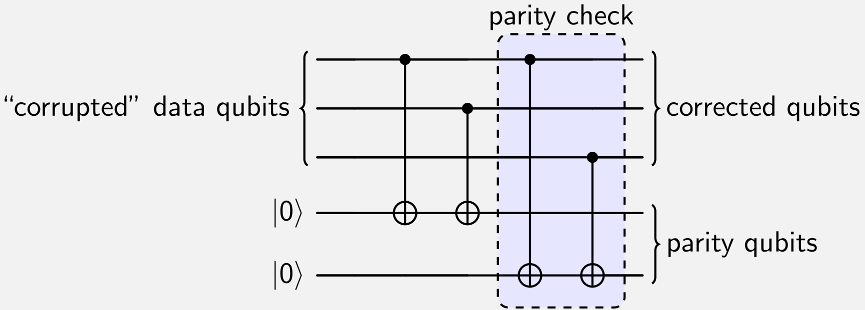
Quantum Error Correction



Let's now consider the top three qubits and the fifth qubit:

$$(\alpha_0 \, |b_0 b_1 b_2\rangle + \alpha_1 \, |\bar{b_0} \bar{b_1} \bar{b_2}\rangle) \, |0\rangle$$

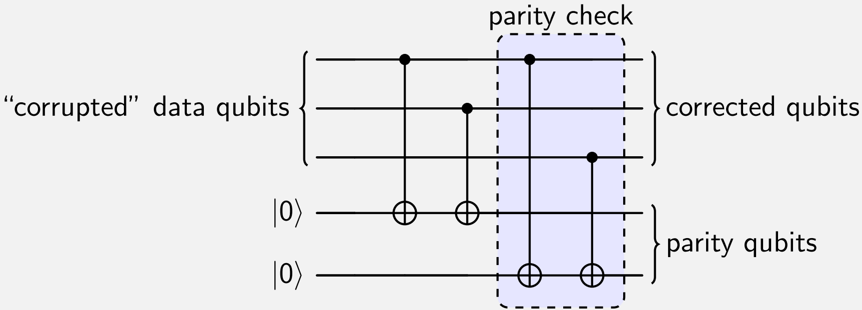
Quantum Error Correction



Let's now consider the top three qubits and the fifth qubit:

$$(\alpha_0 |b_0 b_1 b_2\rangle + \alpha_1 |\bar{b_0} \bar{b_1} \bar{b_2}\rangle) |0\rangle = \begin{cases} \alpha_0 |b_0 b_1 b_2 0\rangle + \alpha_1 |\bar{b_0} \bar{b_1} \bar{b_2} 0\rangle & \text{if } b_0 \oplus b_2 = 0 \\ \end{cases}$$

Quantum Error Correction



Let's now consider the top three qubits and the fifth qubit:

$$(\alpha_0 \left|b_0 b_1 b_2\right\rangle + \alpha_1 \left|\bar{b_0} \bar{b_1} \bar{b_2}\right\rangle) \left|0\right\rangle = \begin{cases} \alpha_0 \left|b_0 b_1 b_2 0\right\rangle + \alpha_1 \left|\bar{b_0} \bar{b_1} \bar{b_2} 0\right\rangle & \text{if } b_0 \oplus b_2 = 0 \\ \alpha_0 \left|b_0 b_1 b_2 1\right\rangle + \alpha_1 \left|\bar{b_0} \bar{b_1} \bar{b_2} 1\right\rangle & \text{if } b_0 \oplus b_2 = 1 \end{cases}$$

Thus, the *fifth* qubit is also **not** entangled with the three "corrupted" data qubits.

Quantum Error Correction

Bob measures the two parity qubits and takes a corrective action on the data qubits:

Three-qubit Repetion Code

Measured parity	**Bob's corrective action**
00	nothing
01	flips the <u>third</u> qubit by applying a NOT gate
10	flips the <u>second</u> qubit by applying a NOT gate
11	flips the <u>first</u> qubit by applying a NOT gate