CADT
IDT វិទ្យាស្ថានបច្ចេកវិទ្យាឌីជីថល
Institute of Digital Technology

Quantum Computing

From Theory to Practical

Gen 9 SE Y4 2025 (G1, G2, & G3)

Lecturer: **Ms. HENG Sovanmonynuth**

Software Engineering (SE), Faculty of Digital Engendering, Department of Computer Science

Cambodia Academy of Digital Technology (CADT)

Email: sovanmonynuth.heng@cadt.edu.kh

Mobile: (+855) 95 479 272

Course Outline

Chapter 1: Introduction to Quantum Computing

Chapter 2: Basic of Quantum Mechanics

Chapter 3: Quantum Algorithms

Chapter 4: Quantum Fourier Transform

Chapter 5: Quantum Errors & Noises

Chapter 6: Quantum Machine Learning

Chapter 7: Quantum Chemistry

Chapter 8: Quantum Cryptography

Chapter 9: Quantum Information Science

Chapter 10: Project Development & Project Report

Final Presentation

Chapter 5: Quantum Errors & Noises

- Introduction to Error Correction
- Quantum Errors and Noise
- Coherence and Decoherence
- Detecting Quantum Errors and Noise
- Quantum Error Correction (QEC)
- Quantum Error Mitigation (QEM)
- Quantum Error Suppression
- Practical Lab: Error Mitigation using MITIQ

Introduction to Errors Correction

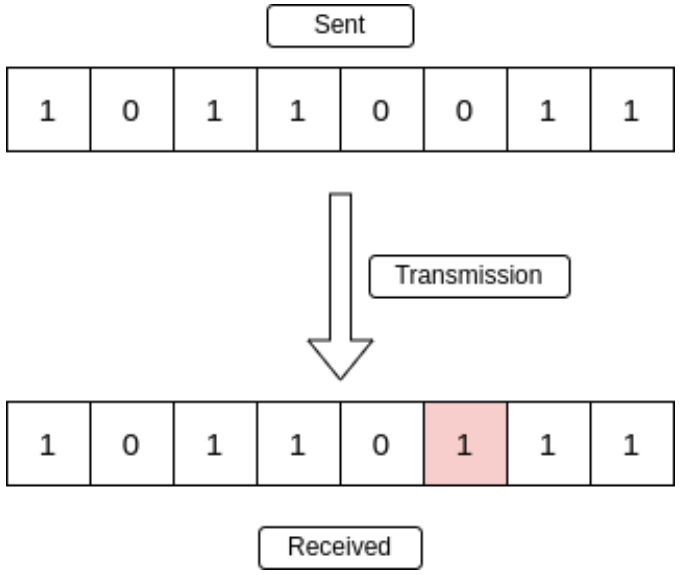
Classical Error Correction

- In classical computing, information is stored as **0 or 1**, and errors occur due to:
  - Electrical noise
  - Imperfect circuits
  - Signal transmission failure
- How to **know there is an error** in the classical computing?

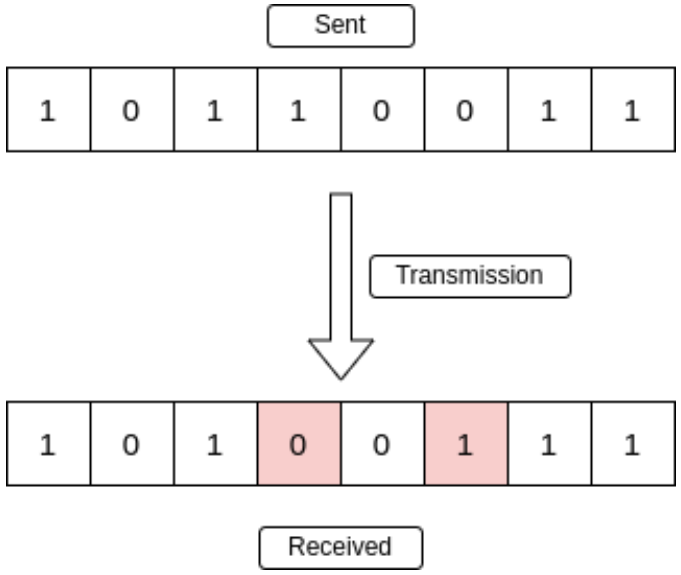
Introduction to Errors Correction

Classical Error Correction

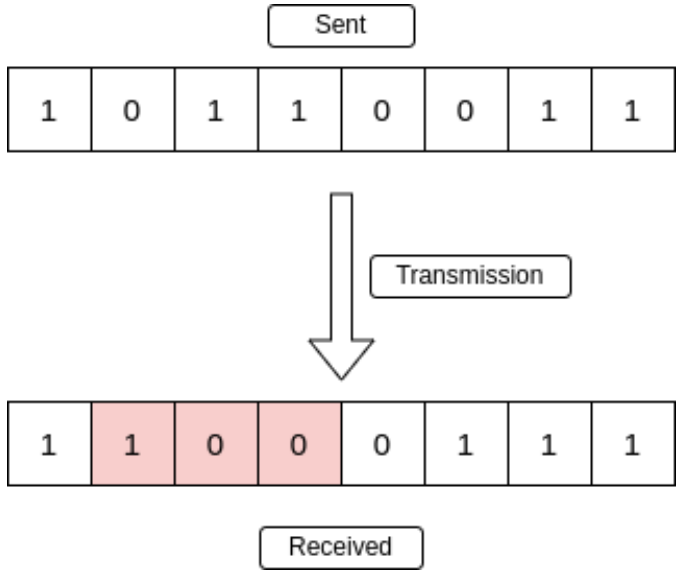
- How to **know there is an error** in the classical computing? Ans: **Detect**



Single Bit Error



Multiple Bit Error



Burst Error

Introduction to Errors Correction

Classical Error Correction

- How to **detect** the error in the classical computer:
- **3-Repertition Code**
  - Check the majority of the encoded bits
  - Can **detect** and **correct** single-bit errors only.
- **Parity-Check**
  - Adds **one parity bit** to detect whether the number of 1s is even or odd.
  - Can **detect** single-bit errors but **cannot correct** them.
- **Hamming codes**
  - Uses additional **redundant bits** placed at positions that are powers of 2.
  - Can **detect** and **correct** single-bit errors.
  - Identifies the error via a **syndrome**, then flips the incorrect bit back

Reading source: https://arthurpesah.me/blog/2022-05-21-classical-error-correction/

Introduction to Errors Correction

Classical Error Correction

- How to **detect** the error in the classical computer:
- **3-Repertition Code**

Error Types	Original bit	Encoded	Received	Majority vote	Decoded result
No Error	0	000	000	0	0
One Error	1	111	11**0**	1	1 (error fixed)
One Error	0	000	0**1**0	0	0 (error fixed)
One Error	0	000	**1**00	0	0 (error fixed)
Two Errors	1	111	**00**1	0	0 (Wrong)

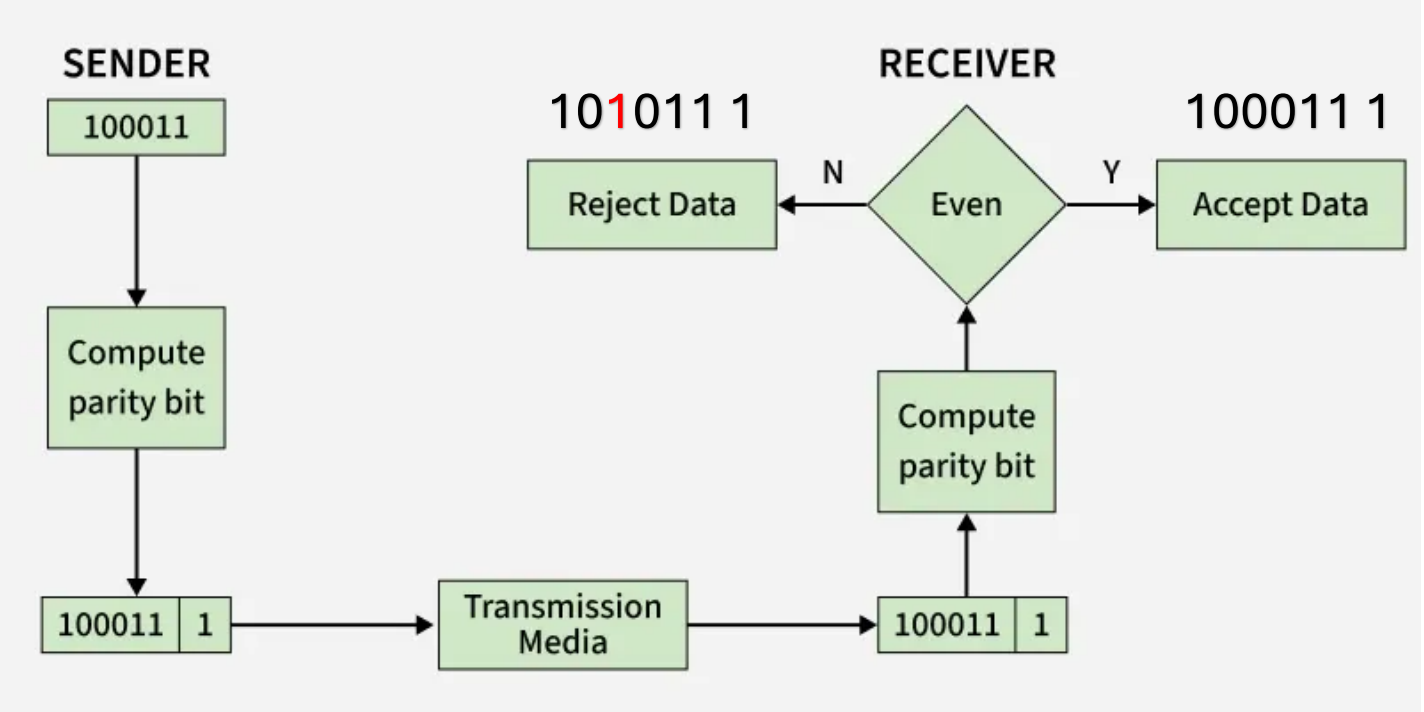
Reading source: https://arthurpesah.me/blog/2022-05-21-classical-error-correction/

Introduction to Errors Correction

Classical Error Correction

- How to **detect** the error in the classical computer:
- **Parity-Check**
  - 100011: r is 1 = 3 is odd, so apply 1
  - 110011: r is 1 = 4 is even, so apply 0
- **Hamming codes**
  - 100011~100011 = No Error
  - 100011~110011 = Error (2nd Bits)

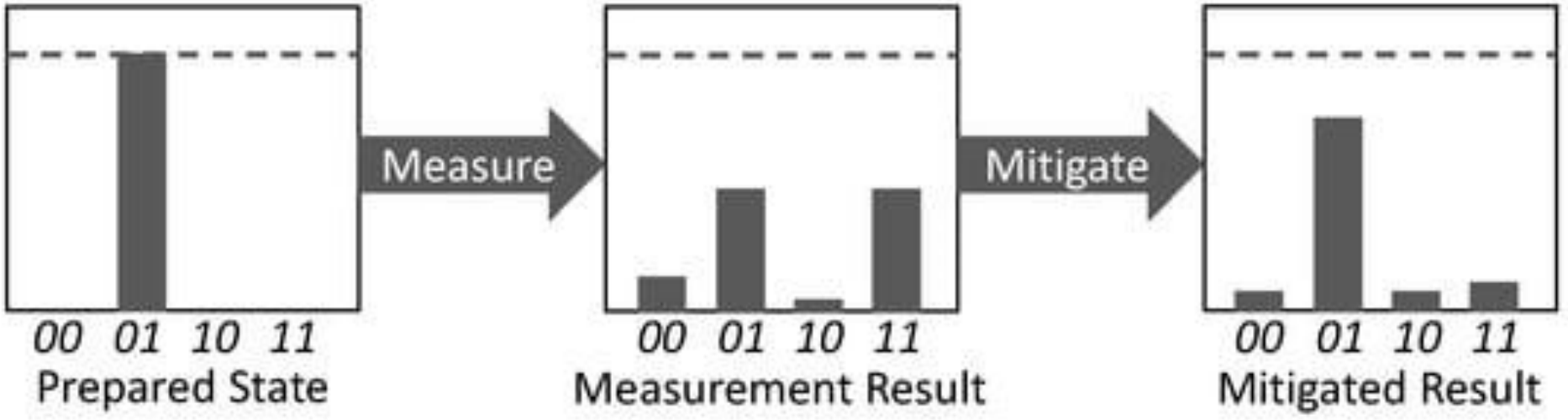
    100011~100011 = No Error (Flip)



Reading source: https://arthurpesah.me/blog/2022-05-21-classical-error-correction/

Quantum Errors and Noise

- **What are quantum errors?**
- Quantum computers operate on quantum bits (qubits), which exist in **superposition** and **entanglement**. These properties also make qubits **extremely sensitive to external environment**, making errors inevitable.
- To build reliable quantum computers, we must **detect**, **correct**, **mitigate**, and **suppress** these errors.



Introduction to Quantum Errors and Noise

- **Types of noise**

Error Type	Description	Example
Bit-Flip (X error)	Flips	$\|0\rangle \leftrightarrow \|1\rangle$
Phase-Flip (Z error)	Changes the phase	$\|1\rangle$ by -1 (add phase shift)
Bit-Phase Flip (Y error)	Combination of bit-flip and phase-flip simultaneously	Impacts both values & phase
Depolarizing Noise	Randomly replaces qubit state with maximally mixed state	State becomes uncertain
Amplitude Damping	Loss of energy from excited state to ground state (relaxation)	Qubit decays to ground state
Phase Damping	Loss of phase info (coherence) without energy loss	Superposition degrades

Quantum Code Visualizer: https://gui.quantumcodes.io/

Coherence and Decoherence

- **Coherence**
  - Ability of a qubit to **stay in superposition**.
  - Essential for quantum algorithms.
- **Decoherence**
  - Interaction with the environment **destroys quantum information**.
  - Caused by heat, vibration, radiation, magnetic fields.

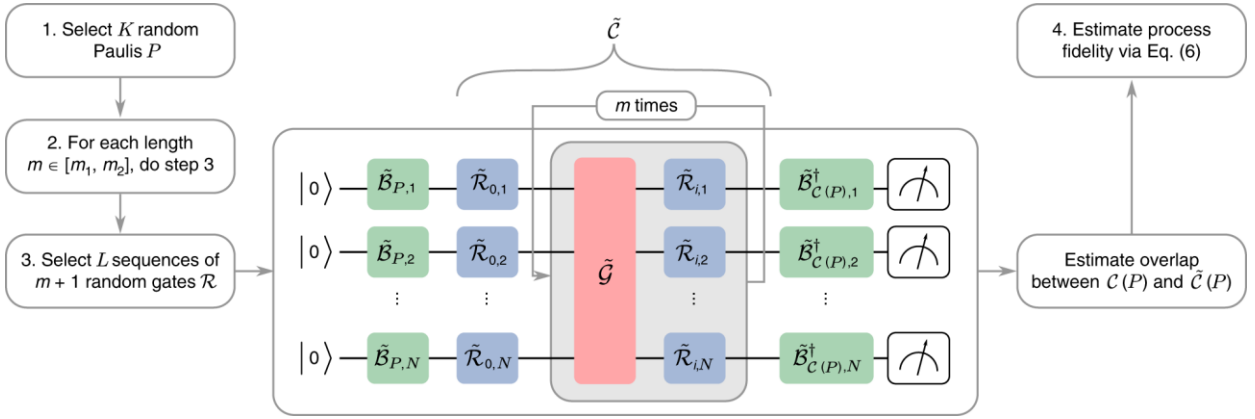
Parameter	Meaning
T_1 (relaxation time)	Time before amplitude damping to
T_2 (dephasing time)	Time before superposition is lost

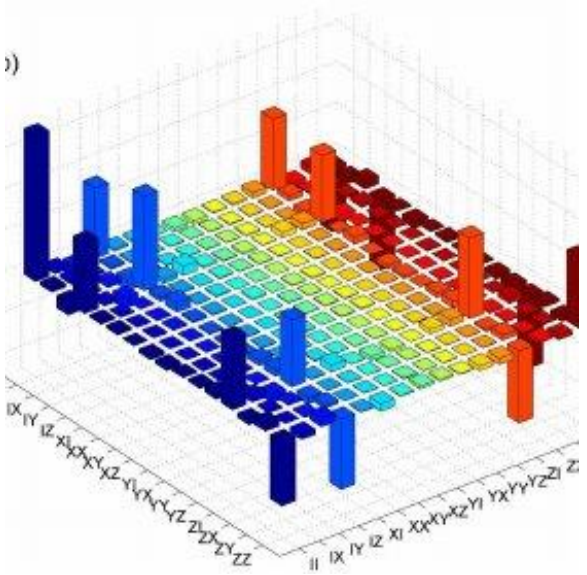
- **Note:** In current quantum hardware, T_1 and T_2 are only **tens to hundreds of microseconds**, limiting algorithm depth.

Detecting Quantum Errors and Noise

Techniques:

- **Measuring Fidelity**
  - Quantifies how close output states are to ideal results.
- **Randomized Benchmarking**
  - Measures average gate fidelity.
- **Quantum Process Tomography**
  - Reconstructs the quantum operation to identify imperfections.
- **Hardware Error Reports (e.g., IBM Quantum)**
  - Shows qubit error rate, CNOT error rate, T_1/T_2 time, readout error.

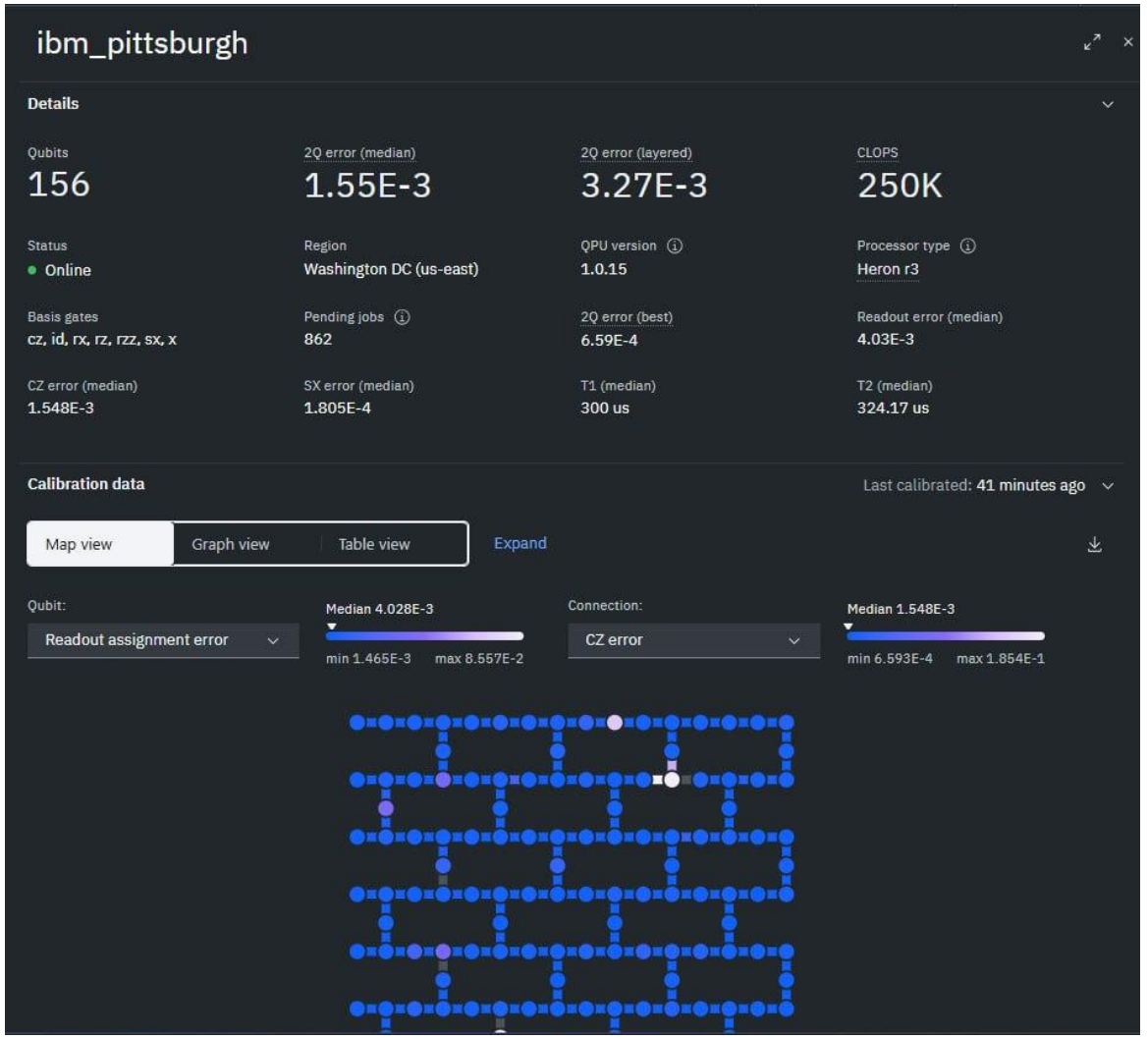




Detecting Quantum Errors and Noise

Techniques:

- **Hardware Error Reports (e.g., IBM Quantum)**
  - Shows qubit error rate, CNOT error rate, T_1/T_2 time, readout error.



Quantum Errors and Noise

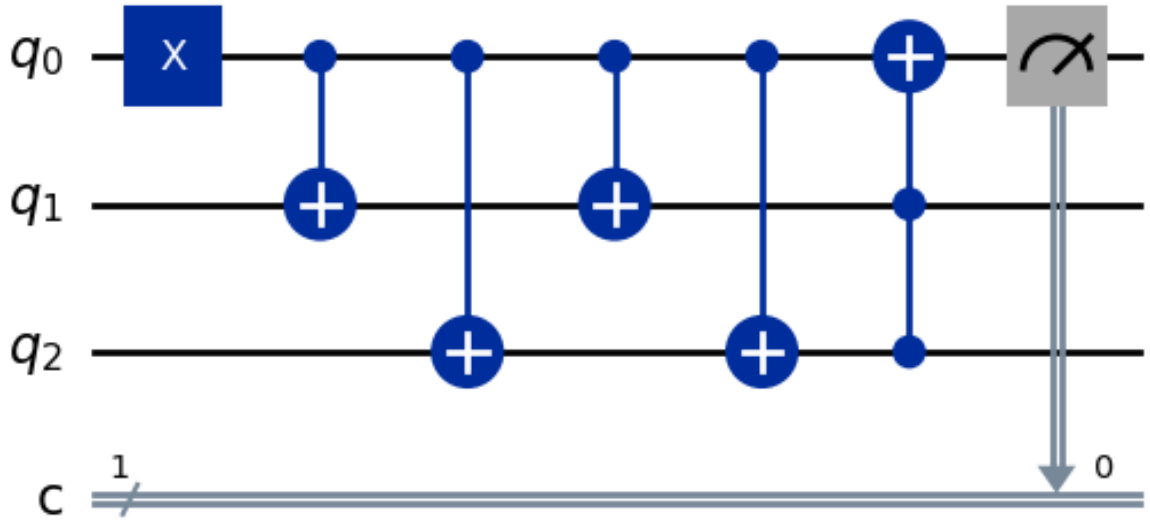
Why are they difficult to handle?

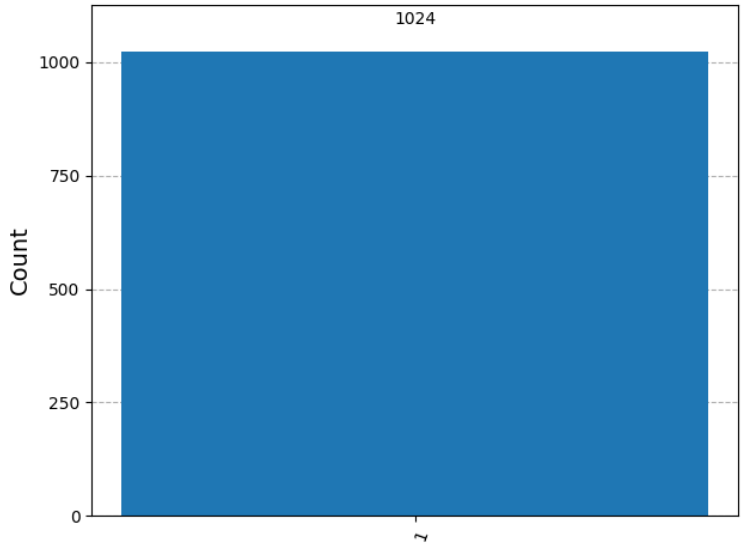
- **Quantum errors** are disturbances affecting the state of qubits due to interaction with the **surrounding environment**.

Classical Errors	**Quantum Errors**
Bits are discrete (0 or 1)	Qubits are continuous complex states
Errors easy to detect & clone	No-cloning theorem prevents copying states
Can observe bits directly	Measurement collapses quantum state

Lab Demo and Practice

- Build Circuit and draw the circuit
- Simulate the Circuit (Ideal: No Error)
- Simulate the Circuit With Error)
- Compare the Circuit Result
- Simulate the Circuit with Optimization Level [0, 1, 2, 3]

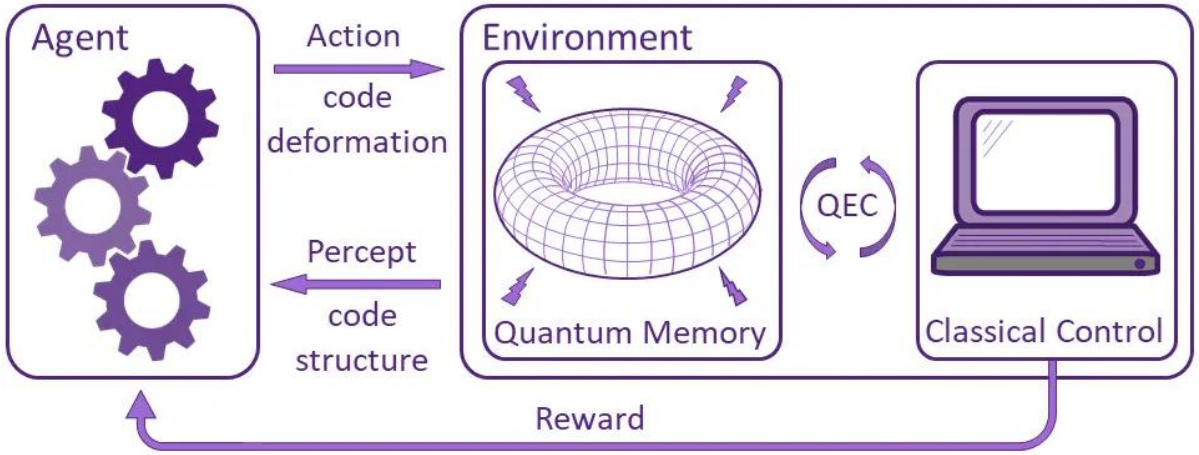




Quantum Error Correction (QEC)

- **Goal:** Protect quantum information by representing **one logical qubit using multiple physical qubits**.

Concept	Description
Redundancy	Encode logical qubit using several physical qubits
Stabilizers	Syndrome measurements to detect errors
Fault tolerance	Circuits operate correctly even with small errors



Quantum Error Mitigation (QEM)

What is quantum error mitigation?

- **QEM** is a collection of techniques used to **reduce the impact of noise on quantum computations without using extra qubits for full error correction**.

Techniques:

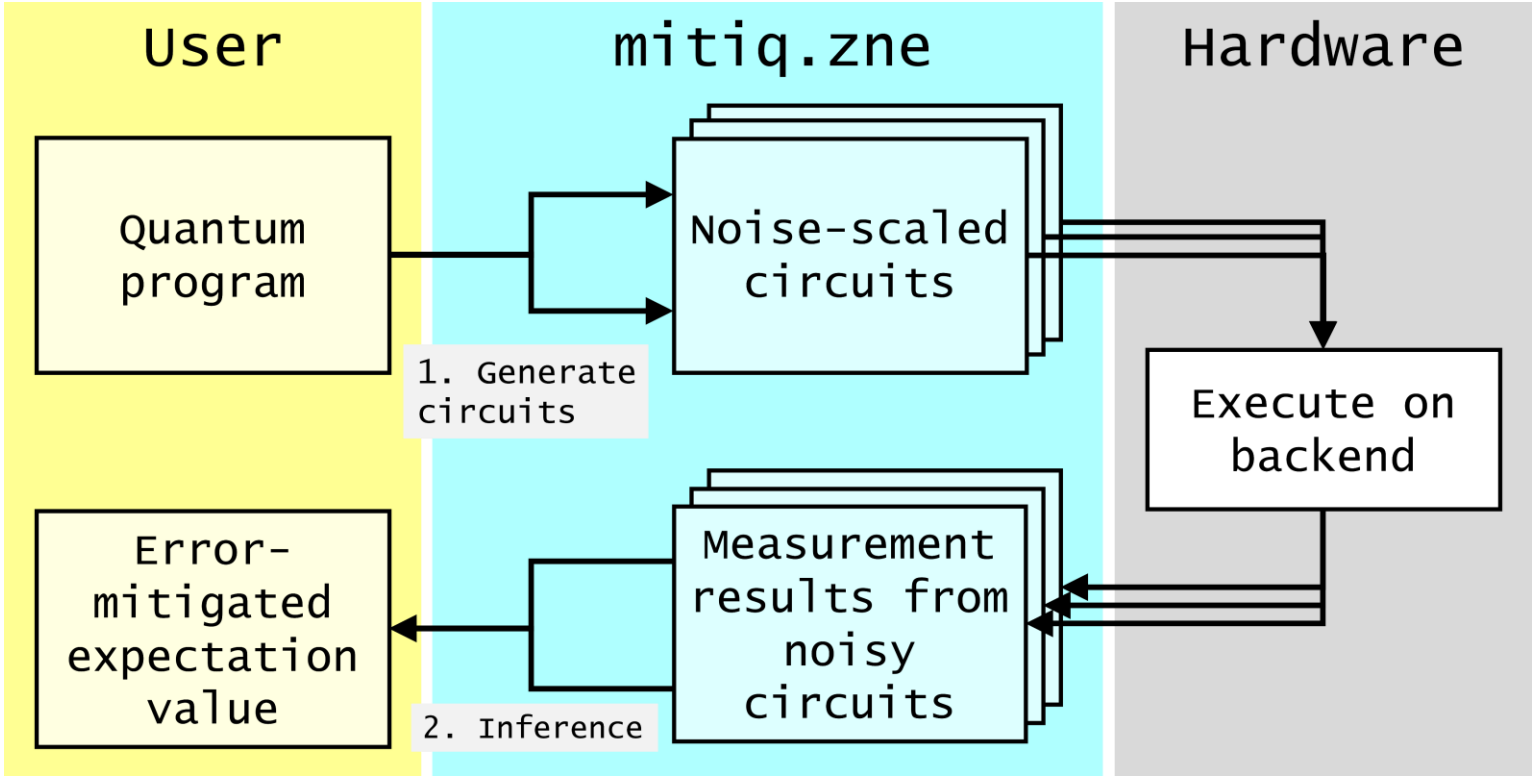
- Zero Noise Extrapolation (ZNE)
- Readout-Error Mitigation
- Probabilistic Error Cancellation (PEC)
- Pauli Twirling

Technique	Description
Zero-noise extrapolation (ZNE)	Run circuit with increasing noise and extrapolate backwards
Probabilistic Error Cancellation (PEC)	Reverse noise by applying inverse operators with weights

Quantum Error Mitigation (QEM)

Techniques:

- **Zero Noise Extrapolation (ZNE)**

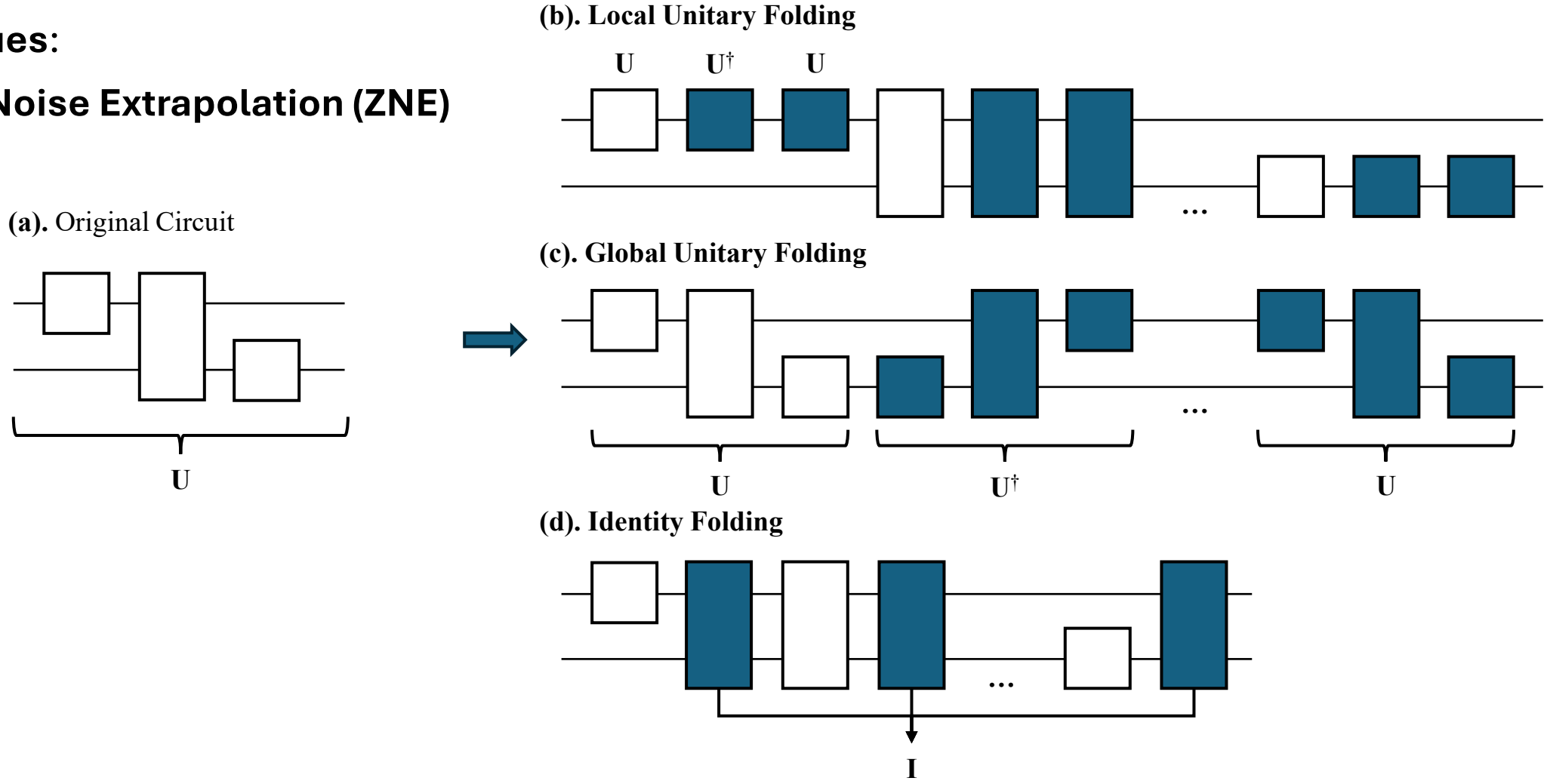


Reading Source: https://mitiq.readthedocs.io/en/stable/guide/zne.html

Quantum Error Mitigation (QEM)

Techniques:

- **Zero Noise Extrapolation (ZNE)**

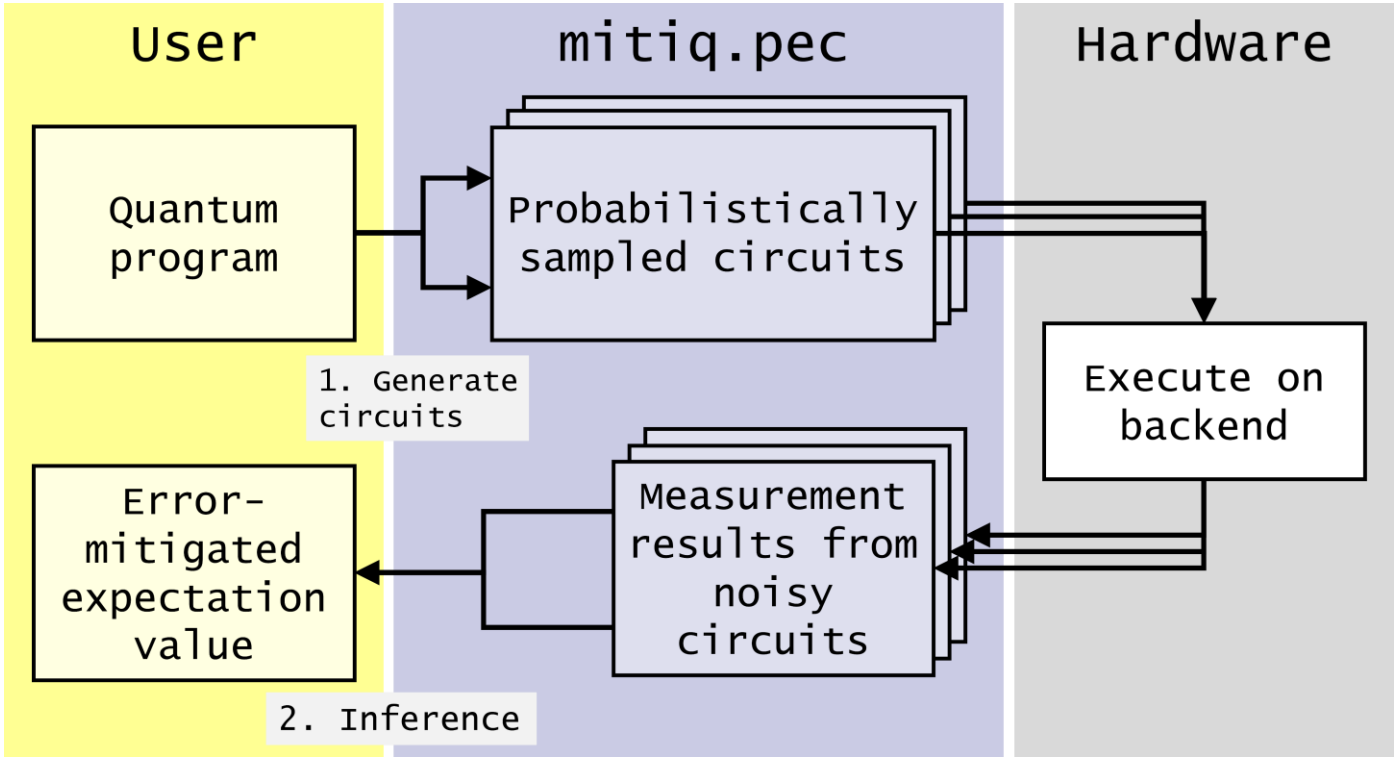


Reading Source: https://mitiq.readthedocs.io/en/stable/guide/zne.html

Quantum Error Mitigation (QEM)

Techniques:

- **Probabilistic Error Cancellation (PEC)**

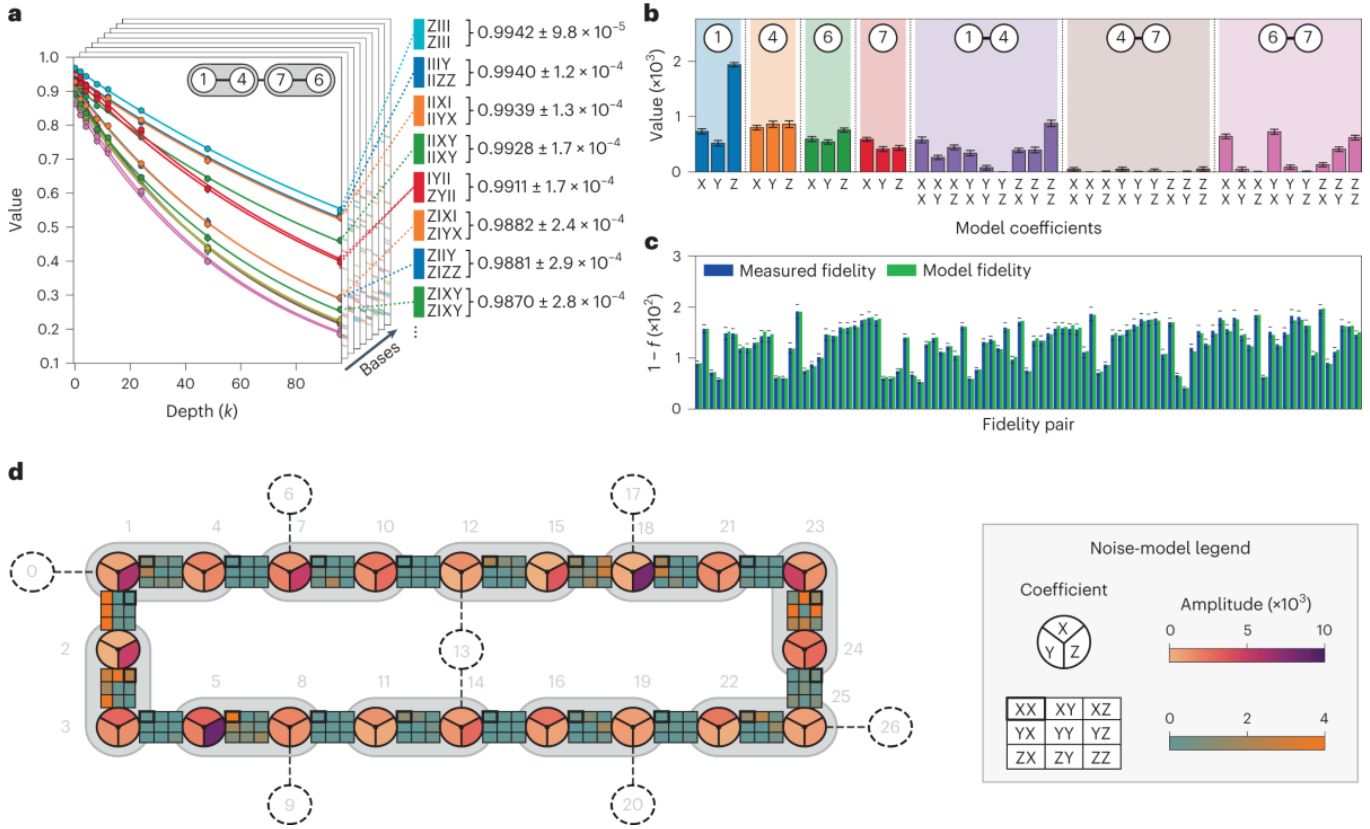


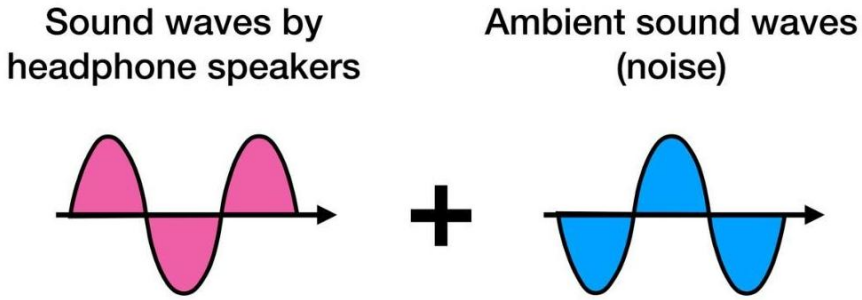
Reading Source: https://mitiq.readthedocs.io/en/stable/guide/pec.html
Paper: https://arxiv.org/pdf/2204.01471

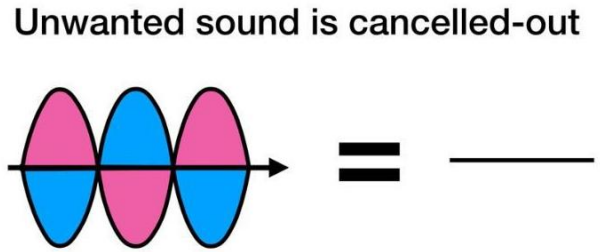
Quantum Error Mitigation (QEM)

Techniques:

- **Probabilistic Error Cancellation (PEC) ~ Noise Cancelation Headphone**



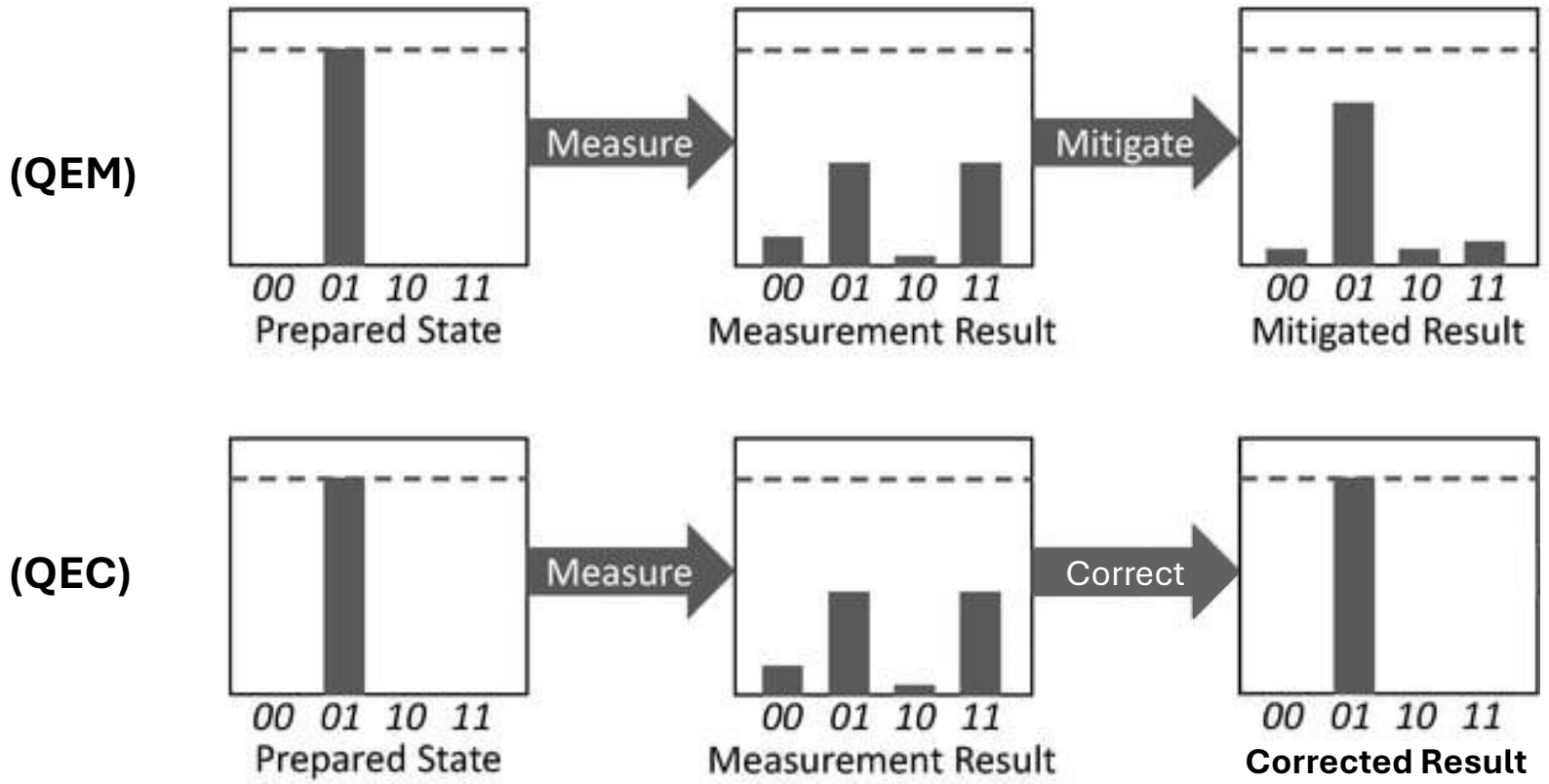




https://mitiq.readthedocs.io/en/stable/guide/pec.html

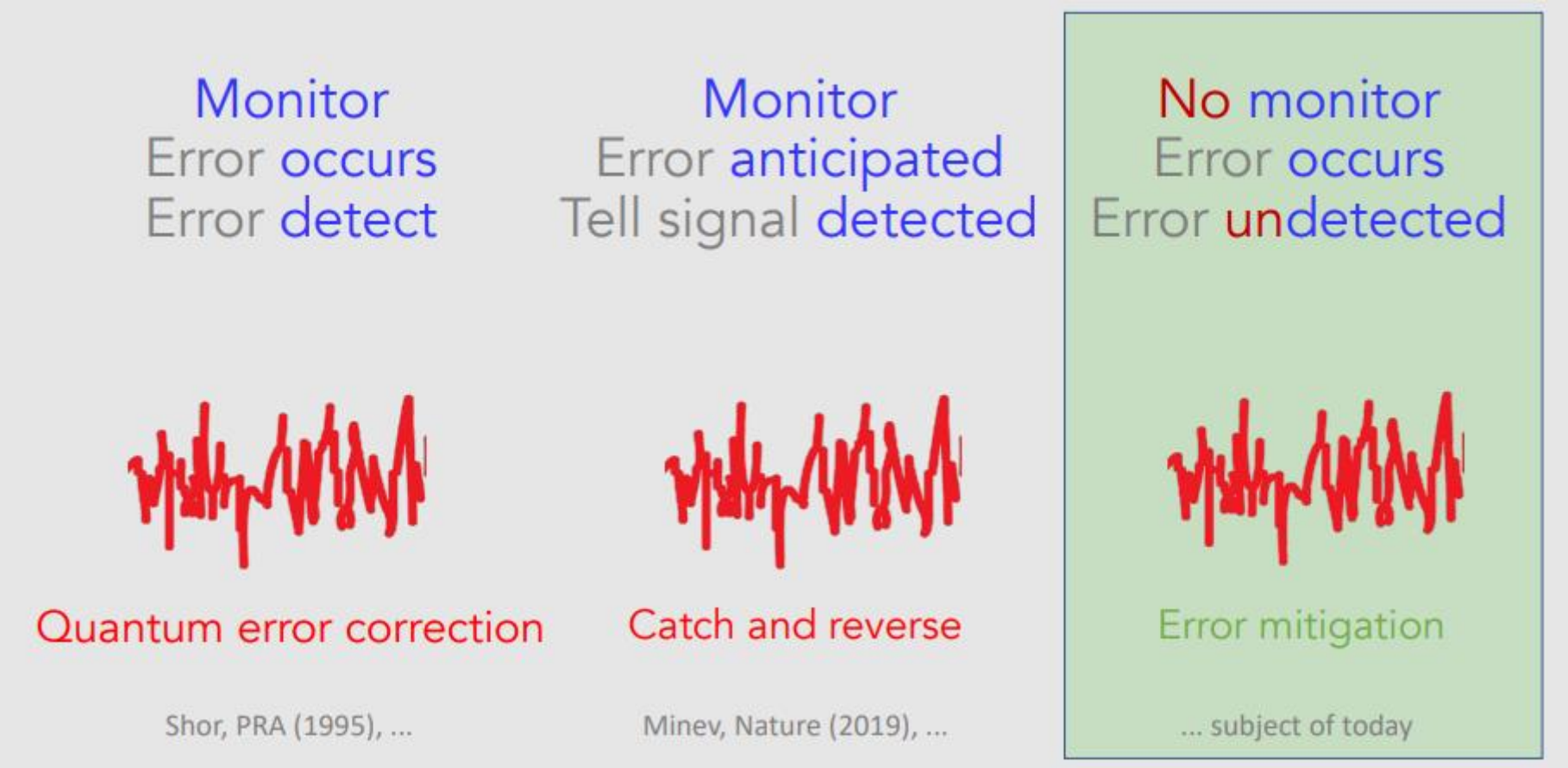
Quantum Error Mitigation (QEM)

- **Quantum Error Mitigation (QEM)** vs **Quantum Error Correction (QEC)**



Quantum Error Mitigation (QEM)

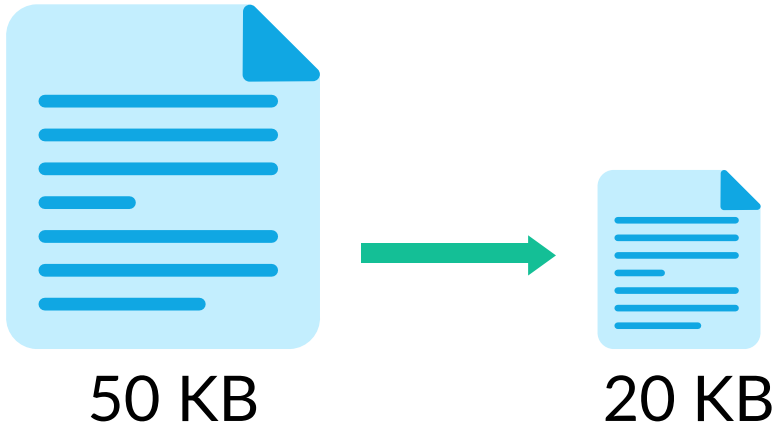
- **Quantum Error Mitigation (QEM)** vs **Quantum Error Correction (QEC)**



Quantum Error Suppression

What is quantum error suppression?

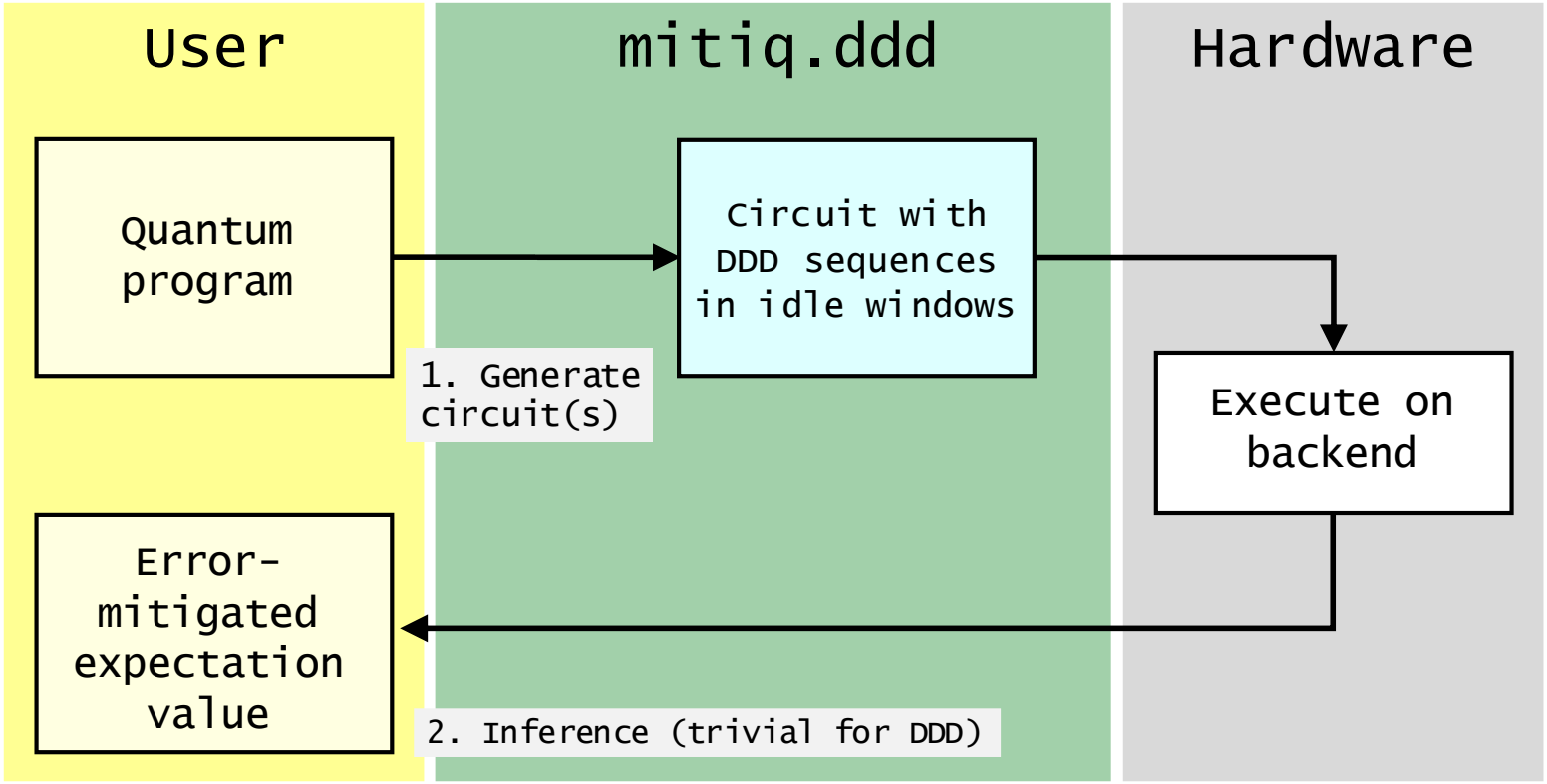
- **Quantum error suppression** is a set of techniques used to **minimize errors** in quantum computers by building resilience into the **hardware** and **quantum operations**. It works proactively by using **advanced control pulses** and **circuit designs**, such as dynamic decoupling, to anticipate and **prevent errors before they happen**, rather than detecting and correcting them after they have occurred.
- **Quantum Error suppression** is quite similar to **compression file**.



Quantum Error Mitigation (QEM)

Techniques:

- **Digital Dynamical Decoupling (DDD)**

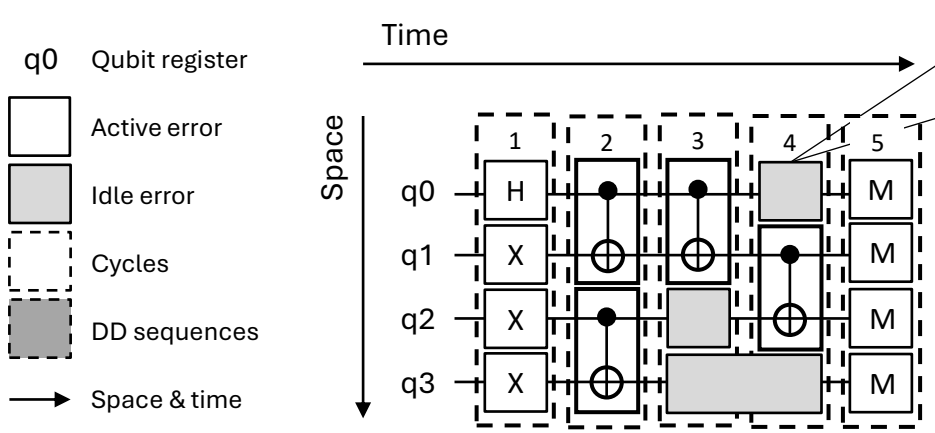


Reading Source: https://mitiq.readthedocs.io/en/stable/guide/zne.html

Quantum Error Mitigation (QEM)

Techniques:

- **Digital Dynamical Decoupling (DDD)**



DD Sequences	Pulse Implementation
CP	$(X)^n$
CPMG	$(Y)^n$
XY4	$(X-Y-X-Y)^n$
XY8	$(XY4-\overline{XY4})^n$

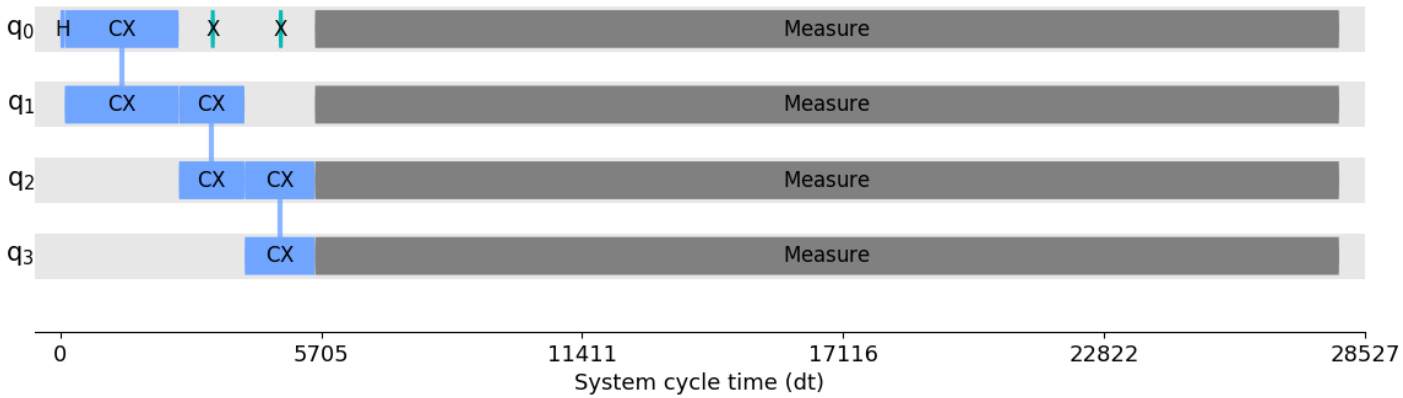


TABLE I: A summary of different DD sequences.

DD sequences	Pulse implementation
Hahn echo [19]	X or Y
CP/CPMG [20, 32]	$(X)^n$ or $(Y)^n$
XY4 [21]	$(X-Y-X-Y)^n$
XY8 [33]	$(XY4-\overline{XY4})^n$
XY16 [33]	$(XY8-\overline{XY8})^n$
UDD [34]	$(X)^n$ or $(Y)^n$ with space τ where $\tau=\sin^2(\frac{\pi j}{2n+2}), j\in\{1,2,...n\}$
KDD [22]	$(\pi)_{\frac{\pi}{6}+\phi}-(\pi)_{\phi}-(\pi)_{\frac{\pi}{2}+\phi}-(\pi)_{\phi}-(\pi)_{\frac{\pi}{6}+\phi}$ where $(\theta)_\phi=e^{\frac{i\theta}{2}[\cos(\theta)\sigma_x+\sin(\theta)\sigma_y]}$, $\phi=0,\frac{\pi}{2},0,\frac{\pi}{2}$

n: repetition time of the basic DD cycle.
$(\theta)_\phi$: a rotation of θ around the axis defined by ϕ.

Reading Source: https://mitiq.readthedocs.io/en/stable/guide/zne.html
Paper: https://arxiv.org/pdf/2204.01471

Comparison & Discussion

Quantum noise is the central challenge in building scalable quantum computers.

To overcome this challenge, we combine:

- **Quantum Error Correction - QEC** (long-term scalable)
- **Quantum Error Mitigation - QEM** (short-term practical)
- **Quantum Error suppression** (hardware-based optimization)
- **Benchmarking and tomography** for measurement

Together, these strategies make quantum computing more reliable.

Practical Lab: Error Mitigation using MITIQ

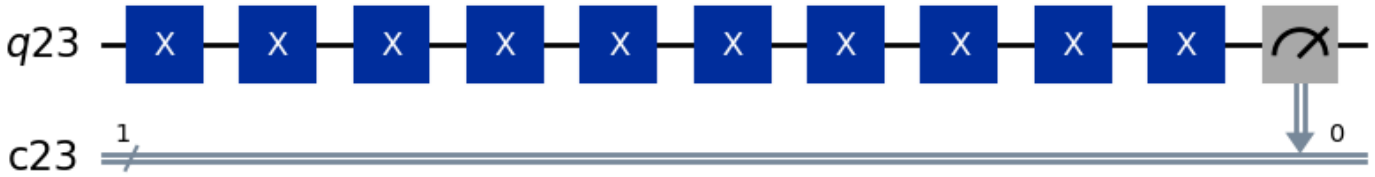
- MITIQ is a Python library for error mitigation.
- **Features**
  - Implements **ZNE** & **probabilistic cancellation**
  - Integrates with Qiskit, Cirq, Braket
- **Resources**
  - Research paper DOI: 10.48550/arXiv.2009.04417
  - GitHub repository & tutorials available
  - Used widely for **NISQ hardware experiments**

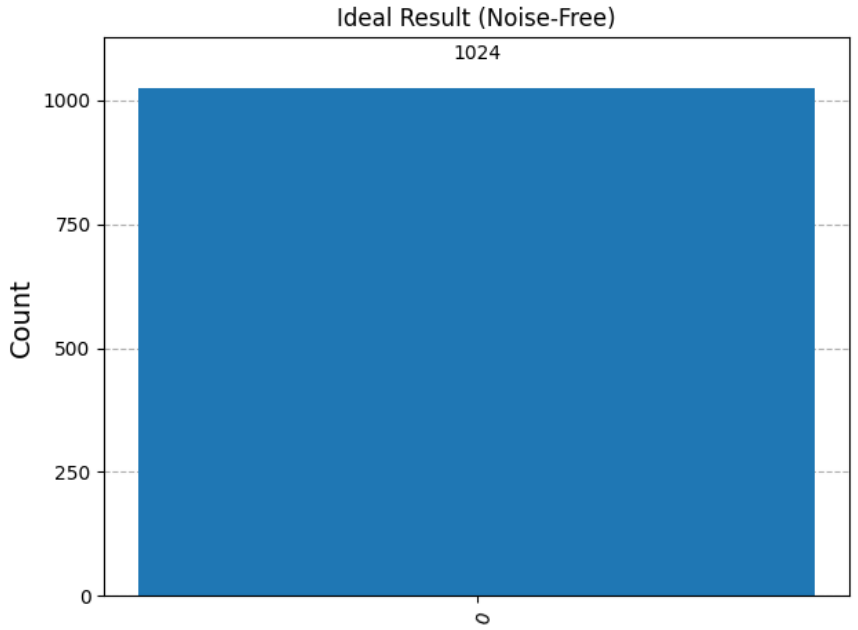


Paper Doi: https://doi.org/10.48550/arXiv.2009.04417 Website: https://unitary.foundation/research/mitiq/
GitHub: https://github.com/unitaryfoundation/mitiq

Lab Demo and Practice

- Build Circuit and draw the circuit
- Simulate the Circuit (Ideal: No Error)
- Simulate the Circuit with QEM of ZNE of Linear Factory
- Compare the Circuit Result of Ideal, Noisy (Before Mitigation), and ZNE (After Mitigation)






Thank you!