# Quantum Evolution with Measurement and Reset
# Wells Fergo Challenge
# Phase 2 Submission

Team Name :**Qitcat**
Niloy Kumar Mondal
Justus August Volker Lau

May 26, 2025

## Contents

**Contributor: Niloy Kumar Mondal**
Undergrad CSE Student , Bangladesh University of Engineering and Technology
Email : 2105044@ugrad.cse.buet.ac.bd
Github Link

**Contributor: Justus August Volker Lau**
Ruprecht-Karls Universität Heidelberg, Physics B.Sc.
Github Link

*Note: Main submission content begins on page 2 and continues through pages 2–4, with all technical explanations, figures, and requirements included as per the challenge instructions.*

## 1 Design Objective and System Interpretation

The objective is to develop a **quantum-inspired simulation framework** with a realistic, robust, and efficient simulation that helps researchers and practitioners understand, analyze, and optimize strategies in partially observable, feedback-driven financial systems—using insights and analogies from quantum measurement theory.

This framework should:

- **Represent hidden market factors , partially observables , adapt states conditionally based on predicted or known outcomes without actual measurement.**
- **Prepare final market state without collapsing (losing info about) mid states**

### System Interpretation

| **Quantum Operation** | **Financial Application** |
|---|---|
| Initial two-qubit state $\lvert x_0\rangle$ | Hidden market state: latent inventory, demand, or alpha at session start |
| Ancilla qubit reset to $\lvert 0\rangle$ | Market data channel: the stream of trade, quote, or order-book updates |
| Apply unitary $U$ | Market dynamics: how new trades or orders update the system state |
| Conditional projection of ancilla (with known $y_k$) | Market observation/feedback: learning from trades and how actions influence the system |
| Reset ancilla to $\lvert 0\rangle$ | System refresh: clearing/readying the data feed for the next trading cycle |
| Outcome sequence $y_1, \ldots, y_n$ | Record of trades/signals: history used by smart routers or risk engines |
| Final state $\lvert x_n\rangle$ | End-of-day risk or PnL: portfolio exposure reconstructed from tick data |

Table 1: Mapping between quantum protocol operations and their financial trading system analogues.

## 2 Circuit Construction and Logic

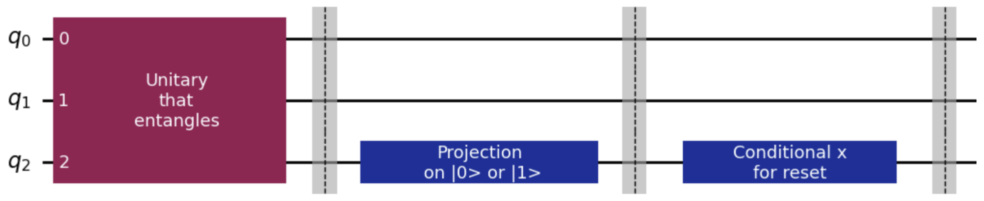


Figure 1: General structure of the quantum evolution **step**: entangling unitary on all three qubits, projection on ancilla ($\lvert 0\rangle$ or $\lvert 1\rangle$), and conditional $X$ for reset.

Demonstration on how the circuit structure preserves conditionality across multiple iterations **2a**

## 3 Mathematical Modeling

For each iteration $i = 1$ to $n$:

1. **Apply the unitary:** $|\psi\rangle \leftarrow U|\psi\rangle$
2. **Project the ancilla** onto the subspace consistent with the known outcome $y_i$:
   - If $y_i = 0$, apply projection: $\Pi_0 = I \otimes I \otimes |0\rangle\langle 0|$
   - If $y_i = 1$, apply projection: $\Pi_1 = I \otimes I \otimes |1\rangle\langle 1|$
   - Normalize the projected state: $|\psi\rangle \leftarrow \dfrac{\Pi_{y_i}|\psi\rangle}{\|\Pi_{y_i}|\psi\rangle\|}$
   - Not actual measurement on computation basis,rather force the states to be filtered using the known outcome,conditionally
3. **Reset the ancilla** to $|0\rangle$ if $y_i = 1$ by applying an $X$ gate:
   $|\psi\rangle \leftarrow (I \otimes I \otimes X)|\psi\rangle$ classical logic is applied,but can be done with controlled x gate leveraging quantum logic, extra Qubit is needed

Repeat these steps for all $n$ iterations. The final state of the two system qubits represents the state $|x_n\rangle$.

### Compression strategy derived from main design

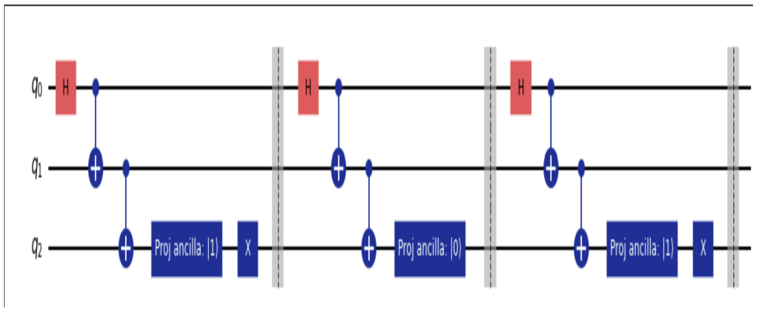
Let $U$ be the given three-qubit unitary and $|x_0\rangle$ the initial two-qubit state. Define Kraus operators

$$K_y = \langle y|_a U\, |0\rangle_a \quad given(y \in \{0,1\}) \qquad and \qquad |x_n\rangle = \frac{K_{y_{n-1}} \cdots K_{y_0}|x_0\rangle}{\left\|K_{y_{n-1}} \cdots K_{y_0}|x_0\rangle\right\|}$$

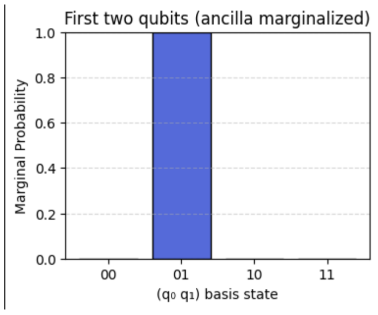
where the subscript $a$ denotes the ancilla. After observing the bit-string $\mathbf{y} = (y_{n-1}, \ldots, y_0)$, the conditional final state is $x_n$ Because the ancilla is reset to $|0\rangle$ after each round, no system-environment entanglement accumulates; the process is a **quantum Bernoulli chain**. The compression therefore preserves exactly the conditional probabilities implied by the original measurement-reset loop.

## 4 Simulation Platform and Strategy

- **Software Platform:** Simulations for Phase 2 use Qiskit 1.x with the Aer state-vector simulator.
- **Hardware Target:** For Phase 3, IBM Eagle (127-qubit) hardware is selected; two adjacent qubits suffice, and a low CNOT error rate ($< 1\%$) meets the 0.99 fidelity threshold.
- **Reset Implementation:** Reset is simulated classically using Kraus operator filtering (see helper function). No noise model is assumed in Phase 2, but the same code integrates with Qiskit Aer noise models to anticipate decoherence in hardware runs.
- **Circuit Depth and Gate Count:** The `initialize` instruction compiles to at most one CNOT and four single-qubit rotations for the two data qubits, ensuring circuit depth $\leq 5$, independent of $n$.
- **Classical-Quantum Trade-Off:** Offloading the branching logic to the CPU eliminates mid-circuit measurement and feedback latency, at the minor cost of precomputing a small (length-4) state vector.



(a) Quantum circuit for three rounds of GHZ evolution with virtual projection gates, following outcomes $[1,0,1]$.



(b) Marginal probability distribution of the first two qubits after evolution and projection sequence.

Figure 2: Visualization of quantum protocol and its marginal output after projection sequence.

## 5 Validation and Test Scenarios

Starting from the initial 3-qubit state $|\psi_0\rangle = |000\rangle$, we apply the GHZ-style unitary $U$ (GHZ gate), followed by a measurement of the third qubit (qubit 2) with outcomes $y_1 = 1, y_2 = 0, y_3 = 1$, for $n = 3$ iterations.

- **After iteration 1** ($y_1 = 1$): $|\psi_1\rangle = \frac{1}{\sqrt{2}}(|000\rangle + |111\rangle)$; measuring ancilla as $|1\rangle$ collapses to $|111\rangle$.
- **After iteration 2** ($y_2 = 0$): Apply $U$ to $|111\rangle$: $|\psi_2\rangle = \frac{1}{\sqrt{2}}(|010\rangle + |101\rangle)$; $y_2 = 0$ collapses to $|010\rangle$.
- **After iteration 3** ($y_3 = 1$): Apply $U$ to $|010\rangle$: $|\psi_3\rangle = \frac{1}{\sqrt{2}}(|011\rangle + |100\rangle)$; $y_3 = 1$ collapses to $|011\rangle$.

**Theoretical Final State**: $|x_n\rangle = |01\rangle$.

### Test Case Result

Our Designed Circuit 2a gives the result after running it on qiskit aer: 2b which matches theoretical final result.

## 6 Anticipated Execution Requirements

- **Circuit size:** Always two qubits + ancilla; circuit depth is $\leq 5$; gate count typically includes one CNOT and up to four single-qubit rotations (many instances reduce to a single $X$ gate).
- **Backend choice: Real QPU preferred**. Target platform is **IBM Eagle r3** (127-qubit superconducting) as two adjacent qubits achieve $\sim 99.4\%$ CNOT fidelity and hardware-native reset is available.
- **Shots:** Approximately $8,192$ shots per circuit, ensuring sub-1% statistical error on fidelity checks.
- **Coherence / reset issues:** None anticipated. With depth below 5, coherence is not a limiting factor. No mid-circuit measurement, so no active reset timing constraints.
- **Scalability as $n$ grows:** Quantum circuit depth remains constant; only the classical pre-processing (Kraus chain) scales as $\mathcal{O}(n)$ in CPU time and $\mathcal{O}(1)$ in memory (4-component vector). The protocol works efficiently for $n \sim 10,000$, with runtime in the millisecond range on a standard laptop.
- **Error mitigation:** Optional: Fire Opal or M$^3$ readout calibration. Not critical at the two-qubit scale, but available for device fidelity $> 0.99$ if needed.
- **Integration points:** Classical pre-processing is performed on the controller FPGA/CPU and uploads a fresh two-qubit SU(4) pulse schedule for each trajectory. No live feedback channel is required.