Final Project: Implementing the W State on Superconducting Qubits Using Pulse-Level Coding

Syed Emad Uddin Shubha
sshubh1@lsu.edu

Course: PHYS 4750 – Quantum Technology
Instructor: Dr. Justin Wilson
Louisiana State University

Abstract

This project explores the implementation of a three-qubit W state on a superconducting quantum processor using low-level pulse control. Using calibrated device parameters extracted from IBM's `ibm_sherbrooke` backend, custom pulse schedules were constructed for single- and two-qubit operations, including RY, X, and echoed Cross-Resonance (CNOT) gates. A Rabi-based calibration procedure was used to determine π-pulse amplitudes, and the full pulse-level schedule was assembled and simulated with realistic noise. The work demonstrates fine-grained quantum control and serves as a step toward hardware-level optimization and fidelity-aware quantum programming.

1 Introduction

The W state is a three-qubit entangled state defined as:

$$|W\rangle = \frac{1}{\sqrt{3}}\left(|001\rangle + |010\rangle + |100\rangle\right).$$

Unlike the GHZ state, the W state retains bipartite entanglement even when one qubit is lost, making it more robust in noisy environments. It plays a significant role in quantum communication tasks such as entanglement-based quantum secret sharing, quantum teleportation, and superdense coding.

This project implements the W state using pulse-level coding on IBM's superconducting qubit platform. Rather than relying on high-level gate abstractions, pulse-level coding provides direct control over the underlying microwave pulses, allowing more precise timing and phase adjustments tailored to device-specific calibration parameters.

2 Preparing the W State

The W state can be constructed from the initial state $|000\rangle$ using a carefully chosen sequence of single- and two-qubit gates as shown in 1

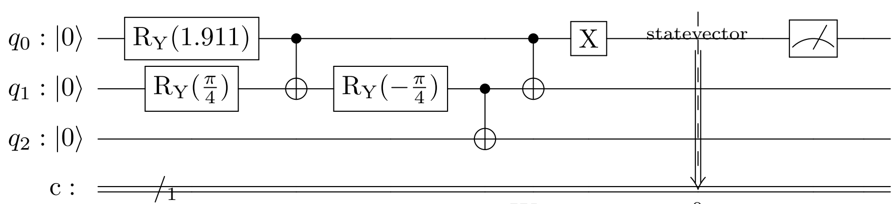

Figure 1: W state preparation

The implemented sequence proceeds as follows:

1. Apply an $RY(\theta)$ rotation to qubit q_0, where $\theta = 2\cos^{-1}(1/\sqrt{3}) \approx 1.911$. This prepares:

$$|\psi_1\rangle = \sqrt{\frac{1}{3}}\,|000\rangle + \sqrt{\frac{2}{3}}\,|100\rangle\,.$$

2. Apply $R_Y(\pi/4)$ to qubit q_1, followed by $CNOT$ with q_0 as control and q_1 as target, and finally $R_Y(-\pi/4)$ to qubit q_1. Together they apply controlled hadamard to q_1 with q_0 as control. Hence the state becomes,

$$|\psi_2\rangle = \sqrt{\frac{1}{3}}\,(|000\rangle + |100\rangle + |110\rangle)$$

3. Apply a $CNOT$ with q_1 as control and q_2 as target. This entangles qubit 2 with qubit 1:

$$|\psi_3\rangle = \sqrt{\frac{1}{3}}\,(|000\rangle + |100\rangle + |111\rangle)$$

4. Apply a $CNOT$ with q_0 as control and q_1 as target. This flips q_1 when q_0 is in $|1\rangle$, introducing the third term:

$$|\psi_4\rangle = \sqrt{\frac{1}{3}}\,(|000\rangle + |110\rangle + |101\rangle)$$

5. Finally, apply an X gate on q_0 to flip its logical value:

$$|\psi\rangle = \sqrt{\frac{1}{3}}\,(|100\rangle + |010\rangle + |001\rangle)$$

Using the calibrated pulse schedules, this sequence is implemented at the pulse level, providing precise control over the system's dynamics.

3 Superconducting Qubits

Superconducting qubits, particularly the transmon variant, are one of the leading physical implementations of qubits due to their scalability, controllability, and compatibility with existing fabrication technologies. The transmon is a weakly anharmonic oscillator realized by a Josephson junction shunted by a capacitor. Its Hamiltonian is approximately:

$$H = \omega a^\dagger a + \frac{\alpha}{2} a^\dagger a^\dagger a a,$$

where ω is the transition frequency between the ground state $|0\rangle$ and the first excited state $|1\rangle$, α is the anharmonicity, and $a^\dagger, a$ are the ladder operators. The anharmonicity ensures that $|1\rangle \to |2\rangle$ transitions are off-resonant, allowing the two-level approximation to hold.

Control and Measurement: Transmon qubits are manipulated using resonant microwave pulses applied via drive channels. The frequency of the pulse determines the addressed qubit, while its amplitude and duration define the rotation angle on the Bloch sphere. The phase of the pulse controls the rotation axis in the XY-plane. Measurement is performed via a dispersively coupled readout resonator, where the qubit state shifts the resonator frequency, and this shift is detected via a probe signal.

Device-Specific Parameters: We used IBM's `ibm_sherbrooke` backend to extract realistic qubit parameters, saved to a JSON file for later reuse in simulation. The physical qubits Q1, Q2, and Q3 were selected based on coherence time and connectivity. These parameters directly informed both our pulse construction and noise model generation.
Parameter Descriptions:

- T_1: Energy relaxation time — average time a qubit remains in $|1\rangle$ before decaying to $|0\rangle$.
- T_2: Coherence (dephasing) time — how long phase information is preserved in a superposition.
- **Readout Error**: Probability that a qubit state is incorrectly measured (e.g., reporting $|1\rangle$ as $|0\rangle$).

- **Drive Frequency**: The calibrated resonance frequency of the qubit for microwave driving (in GHz).
- **Schedule Index**: Corresponds to the index used in drive and measure channels in the pulse schedule.
- **Time resolution (dt)**: The backend's time granularity, used in pulse sampling. For Sherbrooke, $dt = 2.222 \times 10^{-10}$ seconds.

Table 1: Calibrated parameters from `ibm_sherbrooke` used for pulse-level simulation

Qubit	T_1 (μs)	T_2 (μs)	Readout Error	Drive Freq (GHz)	Schedule Index
Q1	306.36	306.98	0.0222	4.7363	0
Q2	260.32	200.50	0.0127	4.8192	1
Q3	396.67	386.45	0.0249	4.7472	2

This table forms the foundation for all subsequent pulse-level construction and noise-aware simulation in this project.

4 Calibration via Rabi Experiment

4.1 Physics of the Rabi Experiment

The Rabi experiment is a cornerstone technique for calibrating microwave pulse amplitudes in superconducting qubit systems. It involves applying a series of pulses with varying amplitudes to a qubit and measuring the probability of the qubit being in the excited state $|1\rangle$. This probability exhibits oscillatory behavior, known as Rabi oscillations, as a function of the pulse amplitude.

For a transmon qubit under a resonant microwave drive, the Hamiltonian in the rotating frame is approximated as:

$$H = \frac{\Omega}{2}\sigma_x, \tag{1}$$

where Ω is the Rabi frequency (proportional to the drive amplitude), and σ_x is the Pauli-X operator. The Rabi frequency Ω governs the rate of qubit rotation around the X-axis on the Bloch sphere.

Applying a pulse of duration t_{pulse} induces a rotation angle θ:

$$\theta = \Omega t_{\text{pulse}}. \tag{2}$$

For a π-pulse, which fully inverts the qubit from $|0\rangle$ to $|1\rangle$, $\theta = \pi$. The probability of measuring $|1\rangle$ post-pulse is:

$$P(|1\rangle) = \sin^2\left(\frac{\theta}{2}\right) = \sin^2\left(\frac{\Omega t_{\text{pulse}}}{2}\right). \tag{3}$$

When $\theta = \pi$, $P(|1\rangle) = 1$ ideally. However, practical measurements are influenced by noise factors such as relaxation (T_1), dephasing (T_2), and readout errors.

4.2 Calibration Process

The Rabi experiment was conducted on qubits 1, 2, and 3 of the IBM Sherbrooke backend to determine the π-pulse amplitude. The process included:

1. **Amplitude Sweep**: Tested amplitudes from 0.05 to 1.0 across 50 points.
2. **Pulse Application**: For each amplitude A, applied an $R_x(\theta)$ rotation, where $\theta = A \times \pi$.
3. **Noise Model**: Incorporated backend-specific noise parameters shown in table 1.
4. **Simulation**: Simulated single-qubit circuits with 1024 shots using AerSimulator, including thermal relaxation and readout errors.
5. **Probability Calculation**: Computed $P(|1\rangle)$ from measurement counts.

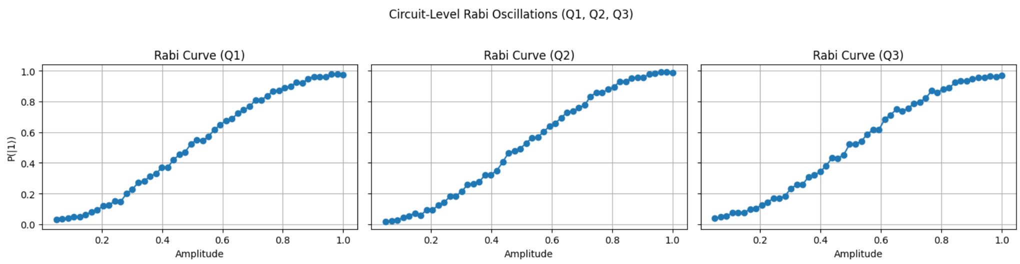

Figure 2: Rabi oscillations for Q1, Q2, and Q3 using simulated backend noise. The peak of each curve corresponds to the π-amplitude.

6. **π-Amplitude Identification**: Selected the amplitude maximizing $P(|1\rangle)$.

The calibrated π-amplitudes were:

$$\text{Qubit 1: } \pi\text{-amp} \approx 0.9806$$
$$\text{Qubit 2: } \pi\text{-amp} \approx 0.9612$$
$$\text{Qubit 3: } \pi\text{-amp} \approx 1.0000$$

We have attached the calibration graph in figure 2. These calibrated amplitudes are used throughout the project to scale single-qubit pulses for X and RY rotations, ensuring accurate qubit state preparation and control at the pulse level.

4.3 Pseudocode for Calibration

Algorithm 1 Calibrating π-pulse amplitude via Rabi oscillation

```
for each qubit q in {Q1, Q2, Q3} do
    Load T1, T2, RO from backend data
    Initialize a noise model:
    Add thermal relaxation error using T1, T2, gate time
    Add readout error using RO
    for each amplitude A in range [0.05, 1.0] do
        Apply R_X(θ = Aπ) to qubit q
        Measure the qubit with 1024 shots
        Record probability P(|1⟩)
    end for
    Find amplitude A* that maximizes P(|1⟩)
    Store A* as π-amplitude for qubit q
end for
```

5 Pulse Schedules for Quantum Gates

5.1 Single-Qubit Gates: RY and X

Single-qubit gates are implemented using microwave Gaussian pulses with carefully calibrated amplitudes. These pulses are defined and scheduled using Qiskit Pulse.

5.1.1 RY Gate

The RY gate performs a rotation around the Y-axis by an angle θ, implemented using a phase shift trick:

- Apply a phase shift of $\frac{\pi}{2}$ to the drive channel to convert an X-rotation into a Y-rotation.
- Play a Gaussian pulse of amplitude $A = \frac{\theta}{\pi} \times A_\pi$, where A_π is the calibrated amplitude for a π-pulse.
- Undo the phase shift with $-\frac{\pi}{2}$.

Algorithm 2 Schedule for $RY(\theta)$ on qubit q

1: $A \leftarrow A_\pi \cdot (\theta/\pi)$
2: Apply phase shift $+\frac{\pi}{2}$ to drive channel of qubit q
3: Play Gaussian pulse with amplitude A and duration τ on drive channel of q
4: Apply phase shift $-\frac{\pi}{2}$ to drive channel of qubit q

5.1.2 X Gate

The X gate is implemented directly as a calibrated Gaussian pulse of amplitude A_π, corresponding to a π-rotation.

Algorithm 3 Schedule for X gate on qubit q

1: Play Gaussian pulse with amplitude A_π and duration τ on drive channel of qubit q

5.2 Two-Qubit Gate: CNOT via Cross-Resonance

The CNOT gate is implemented using the Cross-Resonance (CR) effect between two capacitively coupled transmon qubits. This is done by applying a microwave drive on the control qubit at the frequency of the target qubit. An echoed version is used to cancel unwanted terms.

5.2.1 Physics of Cross-Resonance

The effective Hamiltonian is given by:

$$H_{\mathrm{CR}} = g_{\mathrm{CR}} \left(X_c \otimes Z_t \right),$$

where X_c is the Pauli-X operator on the control and Z_t is the Pauli-Z operator on the target.

5.2.2 Echoed CR Protocol

- Play a CR pulse on the control at the target's frequency (duration τ, amplitude A).
- Apply an X pulse on the target to echo out unwanted Z terms.
- Play a CR pulse on the control at negative amplitude $-A$.

Algorithm 4 Schedule for CNOT(control, target)

1: Set drive frequency of control qubit to match target's frequency
2: Play Gaussian CR pulse with amplitude A on control qubit
3: Play Gaussian X pulse on target qubit using its A_π
4: Play Gaussian CR pulse with amplitude $-A$ on control qubit

5.3 Measurement

Measurement is done using a constant (square) pulse applied to the measure channel of each qubit. The reflected signal is digitized and classified based on predefined IQ discrimination.

Algorithm 5 Measurement schedule for qubit q

1: Play constant pulse of amplitude A_{meas} and duration τ_{meas} on measure channel of q
2: Acquire the reflected signal over τ_{meas}
3: Classify the measurement result using IQ thresholds

5.4 Full W-State Pulse Schedule

The full pulse schedule constructed from the primitive gates described above is shown in Figure 3. This schedule includes all single- and two-qubit gates needed to implement the W-state, along with final measurement pulses.

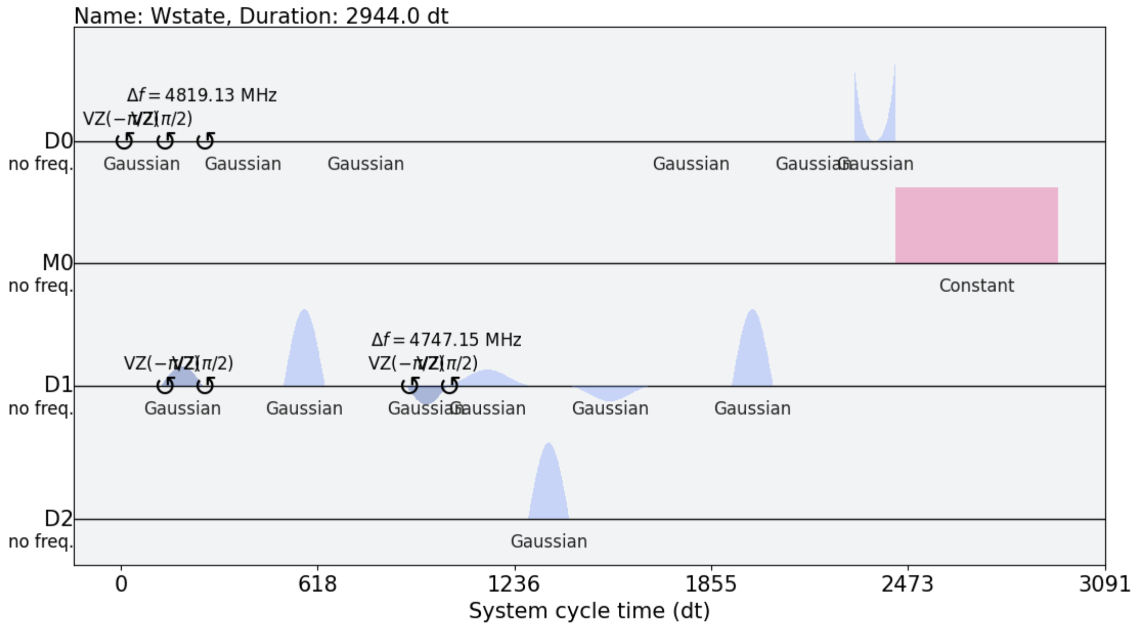

Figure 3: Full pulse schedule for W-state preparation and measurement

6 Discussion and Conclusion

This project demonstrated the complete construction of a three-qubit W state using pulse-level control on a superconducting qubit architecture. The implementation involved extracting device-specific parameters from IBM's `ibm_sherbrooke` backend, performing Rabi-based calibration of π-pulses, and constructing custom pulse schedules for single- and two-qubit gates using the Qiskit Pulse framework.

Unlike traditional gate-level circuits, this approach required low-level manipulation of hardware-native parameters such as drive frequencies, relaxation and dephasing times, and scheduling resolution (dt). A noise-aware Rabi experiment was performed in simulation to determine amplitude calibrations, which were then reused in constructing Gaussian pulses for X, RY, and Cross-Resonance (CNOT) gates. The final pulse schedule was built step-by-step to match the logic of a W-state preparation circuit and visualized using backend time resolution.

Future Work

- **CR Calibration:** Currently, the Cross-Resonance gates were constructed using approximated amplitudes due to missing calibration data between selected qubit pairs on Sherbrooke. In future work, real calibration could be performed using echo CR calibration routines or backend-calibrated instruction maps (if available).
- **Device Execution:** The current implementation was simulated using Aer and backend noise parameters. Execution on real hardware—if pulse access becomes available—could validate fidelity under real decoherence and cross-talk.
- **Pulse Optimization:** Future extensions could include drag correction, optimal control pulse shapes, or closed-loop learning-based calibration to suppress leakage and improve fidelity.
- **Error Characterization:** Quantitative metrics such as state fidelity, average gate fidelity, or quantum process tomography could be used to assess how well the pulse-level W-state preparation matches the ideal case.

The full codebase, including pulse definitions, Rabi experiments, and schedule construction, is publicly available at: https://github.com/syedshubha/PulseSCQ/tree/main.